International Society for Photogrammetry and Remote Sensing
Société Internationale de Photogrammétrie et de Télédétection
Internationale Gesellschaft für Photogrammetrie und Fernerkundung

THE INTERNATIONAL ARCHIVES OF THE PHOTOGRAMMETRY, REMOTE SENSING AND SPATIAL INFORMATION SCIENCES
ARCHIVES INTERNATIONALES DE PHOTOGRAMMÉTRIE, DE TÉLÉDÉTECTION ET DE SCIENCES DE L'INFORMATION SPATIALE - INTERNATIONALES ARCHIV FÜR PHOTOGRAMMETRIE, FERNERKUNDUNG UND RAUMBEZOGENE INFORMATIONSWISSENSCHAFTEN

VOLUME
VOLUME
BAND
XXXVI

PART
TOME
TEIL
3

Symposium of ISPRS Commission III
Photogrammetric Computer Vision
PCV ' 06

Bonn, Germany
20 – 22 September 2006

Editors
Wolfgang Förstner, Richard Steffen

Organisers
ISPRS Commission III - Photogrammetric Computer Vision and Image Analysis

Sponsors
DGPF – German Association for Photogrammetry and Remote Sensing

ISSN 1682-1750

Published by
ISPRS Commission III

ISPRS Headquarters 2004-2008
c/o ORHAN ALTAN, ISPRS Secretary General
Division of Photogrammetry, Faculty of Civil Engineering
Istanbul Technical University
Ayazaga, 34469 Istanbul, Turkey
Phone: +90 212 285 38 10
FAX: +90 212 285 65 87
Email: oaltan@itu.edu.tr
ISPRS WEB Homepage: http://www.isprs.org

Available from
GITC bv
P.O.Box 112
8530 AC Lemmer
The Netherlands
Tel: +31 (0) 514 56 18 54
Fax: +31 (0) 514 56 38 98
E-mail: mailbox@gitc.nl
Website: www.gitc.nl

Table of Content

Photogrammetric Computer Vision --- PCV ' 06

Foreword ______ i

Committees ______ ii

Program Committees ______ iii

Reviewers ______ iii

Session 1: Image Analysis ***Chair Helmut Mayer***

Matthias Butenuth
Segmentation of Imagery Using Network Snakes ______ 1

Session 2: Building Extraction I ***Chair George Vosselman***

Martin Drauschke, Hanns-Florian Schuster, Wolfgang Förstner
Detectability of Buildings in Aerial Images over Scale Space ______ 7

Franz Rottensteiner
Consistent Estimation of Building Parameters Considering Geometric
Regularities by Soft Constraints ______ 13

Norbert Haala, Susanne Becker, Martin Kada
Cell Decomposition for the Generation of Building Models at Multiple Scales ______ 19

Poster session 1

Fayez Tarsha-Kurdi, Tania Landes, Pierre Grussenmeyer, Eddie Smigiel
New Approach for Automatic Detection of Buildings in Airborne Laser Scanner
Data Using first Echo only ______ 25

Lukas Zebedin, Andreas Klaus, Barbara Gruber, Konrad Karner
Facade Reconstruction from Aerial Images by Multi-View Plane Sweeping ______ 31

Olaf Kähler, Joachim Denzler
Detection of Planar Patches in Handheld Image Sequences ______ 37

Matthias Heinrichs, Volker Rodehorst
Trinocular Rectification for Various Camera Setups ______ 43

Fabio Remondino
Detectors and Descriptors for Photogrammetric Applications ______ 49

Karin Hedman, Stefan Hinz, Uwe Stilla
A Probabilistic Fusion Strategy Applied to Road Extraction from
Multi-Aspect SAR Data ______ 55

Carles Matabosch, Elisabet Batlle, David Fofi, Joaquim Salvi
A Variant of Point-to-Plane Registration including Cycle Minimization ______ 61

Miriam Rehor, Hans-Peter Bähr
Segmentation of Damaged Buildings from Laser Scanning Data ______ 67

Olivier Tournaire, Nicolas Paparoditis, Franck Jung, Bernard Cervelle
3D Road-Mark Reconstruction from Multiple Calibrated Aerial Images ______ 73

Jens Göpfert, Christian Heipke
Assessment of Lidar DTM Accuracy in Coastal Vegetated Areas ______ 79

Session 3: Laser Range Data Analysis ***Chair Claus Brenner***

Hermann Gross, Ulrich Thoennessen
Extraction of Lines from Laser Point Clouds ______ 86

Sander Oude Elberink, George Vosselman
Adding the Third Dimension to a Topographic Database Using Airborne Laser Scanner Data ______ 92

Camillo Ressl, Alexander Haring, Christian Briese, Franz Rottensteiner
A Concept For Adaptive Mono-Plotting Using Images and Laserscanner Data ______ 98

Wolfgang von Hansen
Robust Automatic Marker-free Registration of Terrestrial Scan Data ______ 105

Session 4: Surface Reconstruction and Analysis ***Chair Peter Sturm***

Nicolas Champion, Didier Boldo
A Robust Algorithm for Estimating Digital Terrain Models from Digital Surface Models in Dense Urban Areas ______ 111

Fabio Remondino, Li Zhang
Surface Reconstruction Algorithms for Detailed Close-Range Object Modeling ______ 117

Pablo d'Angelo, Christian Wöhler
Image-based 3D Surface Reconstruction by Combination of Sparse Depth Data with Shape from Shading and Polarisation ______ 124

Session 5: Building Extraction II ***Chair Franz Rottensteiner***

Helmut Mayer, Sergiy Reznik
MCMC Linked With Implicit Shape Models and Plane Sweeping for 3D Building Facade Interpretation in Image Sequences ______ 130

Laurence Boudet, Nicolas Paparoditis, Franck Jung, Gilles Martinoty, Marc Pierrot-Deseilligny
A Supervised Classification Approach Towards Quality Self-Diagnosis of 3D Building Models Using Digital Aerial Imagery ______ 136

Mélanie Durupt, Franck Taillandier
Automatic Building Reconstruction from a Digital Elevation Model and Cadastral Data : An Operational Approach______ 142

Poster session 2

Jihye Park, Impyeong Lee, Yunsoo Choi, Young Jin Lee
Automatic Extraction of Large Complex Buildings Using Lidar Data and Digital Maps ____________ 148

Claus Brenner, Nora Ripperda
Extraction of Facades using rjMCMC and Constraint Equations ____________ 155

Eckart Michaelsen, Wolfgang von Hansen, Michael Kirchhof, Jochen Meidow, Uwe Stilla
Estimating the Essential Matrix: GOODSAC versus RANSAC ____________ 161

László Havasi, Tamás Szirányi
Extraction of Horizontal Vanishing Line Using Shapes and Statistical Error Propagation ____________ 167

Janet Heuwold
Verification of a Methodology for the Automatic Scale-Dependent Adaptation of Object Models ____ 173

Yeh Fen Yang, Peter Lohmann, Christian Heipke
Genetic Algorithms for the Unsupervised Classification of Satellite Images ____________ 179

Uwe Soergel, Hermann Gross, Antje Thiele, Ulrich Thoennessen
Extraction of Bridges over Water in Multi-Aspect High-Resolution InSAR Data ____________ 185

Ilkka Korpela
3D Treetop Positioning by Multiple Image Matching of Aerial Images
in a 3D Search Volume bounded by LIDAR Surface Models ____________ 191

Alexander Brzank, Christian Heipke
Classification of Lidar Data into Water and Land Points in Coastal Areas ____________ 197

Dominik Lenhart, Stefan Hinz
Automatic Vehicle Tracking in Low Frame Rate Aerial Image Sequences ____________ 203

Session 6: Road and Vehicle Extraction ***Chair Stephan Heuel***

Helmut Mayer, Stefan Hinz, Uwe Bacher, Emmanuel Baltsavias
A Test of Automatic Road Extraction Approaches ____________ 209

Aluir Porfirio Dal Poz, Rodrigo de Araujo Gallis, João Fernando Custodio da Silva
Semiautomatic Road Extraction by Dynamic Programming Optimisation
in the Object Space: Single Image Case ____________ 215

Jens Leitloff, Stefan Hinz, Uwe Stilla
Automatic Vehicle Detection in Satellite Images ____________ 221

Session 7: Lidar and SAR Processing ***Chair Christian Heipke***

Josef Reitberger, Peter Krzystek, Uwe Stilla
Analysis of Full Waveform Lidar Data for Tree Species Classification ____________ 228

Boris Jutzi, Uwe Stilla
Precise Range Estimation on Known Surfaces
by Analysis of Full-Waveform Laser ____________ 234

Franz Meyer, Stefan Hinz, Andreas Laika, Steffen Suchandt, Richard Bamler
Performance Analysis of Spaceborne SAR Vehicle Detection and Velocity Estimation ____________ 240

Session 8: Image Orientation **Chair Volker Rodehorst**

Fabio Remondino, Camillo Ressl
Overview and Experiences in Automated Markerless Image Orientation ________________________ 248

Ilias Kalisperakis, George Karras, Lazaros Grammatikopoulos
3D Aspects of 2D Epipolar Geometry ____ 255

Justin Domke, Yiannis Aloimonos
A Probabilistic Notion of Camera Geometry: Calibrated vs. Uncalibrated _______________________ 260

Chris Engels, Henrik Stewénius, David Nistér
Bundle Adjustment Rules ____ 266

Foreword

We are proud to present the proceedings of this year's ISPRS Symposium of Commission III „Photogrammetric Computer Vision and Image Analysis“. It takes place 20 – 22 September 2006 in Bonn and is sponsored by the German Society for Photogrammetry and Remote Sensing (DGPF).

This second symposium entitled „Photogrammetric Computer Vision“ reflects the ongoing and increasing interaction between photogrammetry and computer vision.

We decided to select the papers in a double blind review process. This is new for ISPRS Symposia, but follows the trend of many ISPRS workshops in the last years and of course closes the gap in the reviewing procedure with conferences in the computer vision community.

From 70 submitted papers the program committee carefully selected 44 papers, 24 being presented orally and 20 being presented as posters. We thank the program committee and all reviewers for their excellent work, guaranteeing a high standard for this conference.

We are indebted to the organizing committee for their efficient handling of the reviewing process and the publication of these proceeding.

Wolfgang Förstner
Helmut Mayer
Richard Steffen

PCV ' 06 Committees

Program Chairs:

Wolfgang Förstner, University of Bonn
Helmut Mayer, Bundeswehr University Munich

Financial Chair:

Heiko Ellenbeck, University of Bonn

Local Arrangements:

Heidi Hollander, University of Bonn
Monika Tüttenberg, University of Bonn

Technical Committee:

Thomas Läbe, University of Bonn
Bernhard Weber, University of Bonn

Publication Chair:

Richard Steffen, University of Bonn

Program Committee

Claus Brenner, Germany
Norbert Haala, Germany
Olaf Hellwich, Germany
Juha Hyppää, Finland
Theo Moons, Belgium
Ilkka Niini, Finland
David Nister, USA
Marc Pollefeys, USA
Camillo Ressl, Austria
Franz Rottensteiner, Austria
Daniel Scharstein, USA
Uwe Stilla, Germany
Peter Sturm, France
George Vosselman, The Netherlands
Chunsun Zhang, Australia

Reviewers

Claus Brenner
Norbert Haala
Olaf Hellwich
Juha Matti Hyppää
Theo Moons
Peggy Agouris
Arie Croitoru
Richard Hartley
Stefan Hinz
Amnon Krupnik
Franz J. Meyer
Tomas Pajdla
Frank A. van den Heuvel
Volker Paelke
Ilkka Niini
David Nister
Marc Pollefeys
Camillo Ressl
Franz Rottensteiner
Meir Barzohar
Jan-Michael Frahm
Matthias Heinrichs
Volker Rodehorst
Anselmo Lastra
Changchang Wu
Nicolas Paparoditis
Uwe Weidner
Daniel Scharstein
Uwe Stilla
Peter Sturm
George Vosselman
Chunsun Zhang
Christian Briese
Nora Ripperda
Stephan Heuel
Marc Jäger
Chris McGlone
Ulas Yilmaz
Andreas Reigber
Christoph Dold

Notes

SEGMENTATION OF IMAGERY USING NETWORK SNAKES

Matthias Butenuth

Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover
Nienburger Str. 1, 30167 Hannover, Germany
butenuth@ipi.uni-hannover.de

KEY WORDS: Segmentation, Imagery, Snakes, Network, Topology, Shape

ABSTRACT:

A new methodology for the segmentation of imagery using network snakes is presented in this paper. Snakes are a well known technique, but usually are limited to closed object boundaries. Enhancing traditional snakes the focus is on objects forming a network respectively being adjacent with only one boundary in between. In addition, the focus is on linear objects with open non-fixed endings. The internal energy controlling the shape of the object contours during the energy minimization process is defined for nodes with different degrees to enable the exploitation of the object topology. Exemplary results of two different applications demonstrate the functionality and transferability of the proposed methodology: First, field boundaries are extracted from high resolution satellite imagery. The second example from the medical sector deals with the delineation of adjacent cells in microscopic cell imagery. Concluding remarks are given at the end to point out further investigations.

1. INTRODUCTION

The segmentation of imagery is a well known problem in image processing and computer vision. One important methodology to delineate objects precisely are active contours, first introduced by (Kass et al., 1988). Active contours are a sophisticated image processing technique combining image features with shape constraints in an energy minimization process. Parametric active contours, often called snakes (Kass et al., 1988; Blake and Isard, 1998), have a rigid topology, in contrary to geometric active contours (Malladi et al., 1995; Caselles et al., 1997), which are able to change their topology due to flexible level sets and thus allow for extracting foreground objects without prior knowledge about their shape. Numerous approaches using snakes have been presented to detect different objects in many kinds of imagery, for example refer to medical image segmentation (McInerney and Terzopoulos, 1996; Suri et al., 2002), 3D deformable surface models (Cohen and Cohen, 1993), the extraction of roads using scale space and snakes (Laptev et al., 2000) or the displacement of lines in cartographic generalization tasks (Burghardt and Meier, 1997).

Most of these approaches require closed contours, which describe the boundary of an object separately – a limitation concerning linear objects with open non-fixed endings and a limitation regarding objects, which form a network and thus interact with each other during the optimization process. A new methodology of parametric active contours to overcome these restrictions is presented in this paper, called *network snakes*. In the literature, only a limited amount of work can be found regarding active contours beyond explicitly or implicitly represented closed object boundaries. Trihedral corners imposing constraints of 90 degree angles between the three edges ending at the corner are used to extract buildings in (Fua et al., 2000). An extension of parametric active contours, which combines the ability to handle transiently touching objects and exerts topological control is given in (Zimmer and Olivio-Marin, 2005). An adaptive adjacency graph consisting of a network of active contours was firstly introduced in (Jasiobedzki, 1993) and afterwards utilized in (Dickinson et al., 1994) to track 3D objects. The authors introduce constraints in the form of springs to connect the contour ends, but the approach does not enable the definition of a unique nodal point including a geometrical control of the contours up to the nodal point.

In contrary, the methodology presented in this paper is able to handle objects with a given network topology, but without the necessity of introducing any particular constraints. Possible applications using network snakes are, for example, the extraction of road networks, field boundaries as well as adjacent cells. For these purposes parametric active contours are more applicable than level set techniques, since the image-dependent energy terms of parametric active contours are defined specifically to individual objects. Multiple level sets as well as multiple parametric active contours are not suitable, because they can intersect or overlay each other losing the correct topology. However, parametric active contours need an initialization to start the energy minimization process. The required information can be taken from an initial segmentation or from a GIS, as long as a correct topology can be assumed.

The next section outlines traditional snakes, while Section 3 focuses on the enhancements concerning network snakes. In Section 4 exemplary results of two different applications are presented to demonstrate the potential and the transferability of the proposed methodology: At first, the extraction of field boundaries from high resolution satellite imagery is depicted. Field boundaries have become objects of increasing interest during the last few years. Application areas are geo-scientific questions such as the derivation of potential wind erosion risk fields and applications in the agricultural sector, for instance precision farming or the monitoring of subsidies. Work on the extraction of field boundaries exists but is limited, for example refer to (Torre and Radeva, 2000; Aplin and Atkinson, 2004), but neither a fully automatic solution nor the exploitation of the topology is used. The second exemplary result highlighted in Section 4 is the extraction of adjacent cells in microscopic cell imagery. Medical image segmentation has received a large attention, in particular the delineation of cells for high-throughout biological research and drug discovery (Suri et al., 2002; Jones et al., 2005). However, the focus is mostly on single cells not taking into account the neighborhood. Finally, concluding remarks are given and further investigations are discussed in Section 5.

2. TRADITIONAL SNAKES

In this section, parametric active contours are summarized in order to provide a basis for a discussion of their pros and cons concerning the enhancements contained in Section 3. Traditional snakes (Kass et al., 1988) are defined as a parametric curve

$$v(s) = (x(s), y(s)) \ , \tag{1}$$

where s is the arc length and x and y are the image coordinates of the 2D-curve. In the simplest way, the image energy can be written as the image intensity itself with

$$E_{img}(v(s)) = I(v(s)) \ , \tag{2}$$

where I represents the image. In the literature, the image energy is often defined as

$$E_{img}(v(s)) = -|\nabla I(v(s))|^2 \ . \tag{3}$$

$|\nabla I(v(s))|$ is the norm or magnitude of the gradient image at the coordinates $x(s)$ and $y(s)$. In practice, the image energy $E_{img}(v(s))$ is computed by integrating the values $|\nabla I(v(s))|$, taken from precomputed gradient magnitude images along the line segments, that connect the vertices of the contour. The internal energy is defined as

$$E_{int}(v(s)) = \tfrac{1}{2}\left(\alpha(s)\cdot|v_s(s)|^2 + \beta(s)\cdot|v_{ss}(s)|^2\right) \ , \tag{4}$$

where v_s and v_{ss} are the first and second derivative of v with respect to s. The function $\alpha(s)$ controls the first-order term of the internal energy: the elasticity. When the aim is to minimize $E_{int}(v(s))$ and $v(s)$ is allowed to move, large values of $\alpha(s)$ let the contour become very straight between two points. The function $\beta(s)$ controls the second-order term: the rigidity. Large values of $\beta(s)$ let the contour become smooth, small values allow the generation of corners. $\alpha(s)$ and $\beta(s)$ need to be predefined based on the modeled shape characteristics of the object of interest.

The total energy of the snake E^*_{snake}, to be minimized, is defined as

$$\begin{aligned} E^*_{snake} &= \int_0^1 E_{snake}(v(s))\,ds \\ &= \int_0^1 \left[E_{img}(v(s)) + E_{int}(v(s)) + E_{con}(v(s))\right] ds \ . \end{aligned} \tag{5}$$

The additional external energy $E_{con}(v(s))$ is introduced in (Kass et al., 1988) as an external constrained force, which provides the opportunity for individual forces at particular parts or points of the contour. With constant weight parameters $\alpha(s) = \alpha$ and $\beta(s) = \beta$ a minimum of the total energy in Equation 5 can be derived by solving the Euler equation:

$$\frac{\partial E_{img}(v(s))}{\partial v(s)} - \alpha v_{ss}(s) + \beta v_{ssss}(s) = 0 \ . \tag{6}$$

The derivatives are approximated with finite differences since they can not be computed analytically. Converted to vector notation with $v_i = (x_i, y_i)$ and with $\partial E_{img}(v(s)) / \partial v(s) = f_v(v)$ the Euler equations read

$$\begin{aligned} &\alpha_i(v_i - v_{i-1}) - \alpha_{i+1}(v_{i+1} - v_i) \\ &+ \beta_{i-1}(v_{i-2} - 2v_{i-1} + v_i) - 2\beta_i(v_{i-1} - 2v_i + v_{i+1}) \\ &+ \beta_{i+1}(v_i - 2v_{i+1} + v_{i+2}) \\ &+ f_v(v) = 0 \end{aligned} \tag{7}$$

and can be rewritten in matrix form as

$$Av + f_v(v) = 0 \ . \tag{8}$$

A is a pentadiagonal matrix, which depends only on the functions α and β. Equation 8 can be solved iteratively by introducing a step size γ multiplied with the negative time derivatives $\partial v / \partial t$, which are discretized by $v_t - v_{t-1}$. It is assumed that $f_v(v)$ is constant during a time step, i.e. $f_v(v_t) \approx f_v(v_{t-1})$, yielding an explicit Euler step regarding the image energy. In contrast, the internal energy is an implicit Euler step due to their specification by the banded matrix A. The resulting equation is

$$Av_t + f_v(v_{t-1}) = -\gamma(v_t - v_{t-1}) \ . \tag{9}$$

The time derivatives vanish at the equilibrium ending up in Equation 8. Finally, a solution can be derived by matrix inversion:

$$v_t = (A + \gamma \mathrm{I})^{-1}(\gamma v_{t-1} - \kappa f_v(v_{t-1})) \ , \tag{10}$$

where I is the identity matrix and κ is an additional parameter in order to control the weight between internal and image energy.

A requirement of traditional snakes is the necessity to have an initialization close to the true object boundary. Additional terms to increase the capture range of the image forces and thus bridge larger gaps between initialization and true object boundary, for example the balloon model (Cohen, 1991), are only applicable, when the background is relatively homogeneous and no disturbing structures hinder the movement of the snake. Since these conditions can not be guaranteed in general, a solution without such additional terms is preferred in this work. Instead, strong internal energies are used containing the modeled shape characteristics of the object of interest to be relatively independent of the initialization and those parts of the image energies, which represent disturbing structures.

3. NETWORK SNAKES

In order to enhance traditional snakes to be able to deal with network topologies and open endings of contours, a closer look to the internal energy $E_{int}(v(s))$ controlling the shape part of the curve is required. The minimization of the internal energy during the optimization process is only defined for closed object boundaries, i.e. $v_0 = v_n$ (Kass et al., 1988), because the derivatives are approximated with finite differences (cf. Equation 7). Most of the approaches to be found in the literature use closed contours or define fixed end points when using open contours. This process requires correct end points before starting the snake optimization, which often can not be guaranteed. Similarly, network topologies represented by single contours ending in common nodal points require predefined correct nodal points. In this work a new definition of snakes is given, achieving a solution using image features and shape constraints without fixed end or nodal points.

At first, the topology of the initial contour has to be derived. In addition to the nodes with a degree $\rho(v) = 2$ of the preliminary contour $v(s)$ each node with a degree $\rho(v) \neq 2$ has to be set up. Nodes with a degree $\rho(v) = 1$ define the end points and nodes with a degree $\rho(v) \geq 3$ define the nodal points of the contour (cf. Fig. 1 for an example).

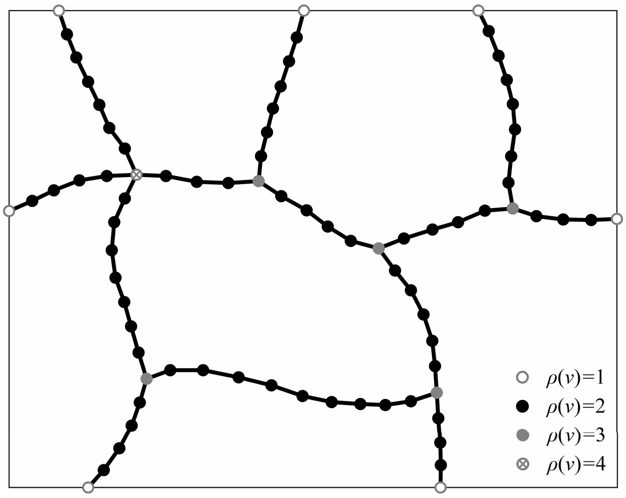


Figure 1. Topology of a network snake

Imposing the network topology in the energy minimization process causes a problem when solving Equations 7 and 8: the derivatives approximated by finite differences are not defined for nodes with a degree $\rho(v) = 1$ or $\rho(v) \geq 3$, because the required neighboring nodes are either not available ($\rho(v) = 1$) or existing multiple times ($\rho(v) \geq 3$). Thus, the shape control can not be accomplished at these parts of the contour in a traditional way.

Let v_a, v_b and v_c represent three contours, each ending in a common *nodal point* v_n with a degree $\rho(v) = 3$. Regarding Equation 7, the first term, weighted by the parameter α, can not support the control of the internal energy during the energy minimization process in the vicinity of v_n when using network snakes: the finite differences of the first term approximating the derivatives are only available for the two nodes v_{n-1} and v_n, but not for v_{n+1}. Thus, no shape control is possible and the first term is not considered. The second term of the internal energy, weighted by the parameter β, is rewritten using the available finite differences controlling the curvature of the contour. Consequently, the control of the total energy at the common nodal point $v_n = v_{a_n} = v_{b_n} = v_{c_n}$ is defined for network snakes as follows:

$$\begin{aligned}\beta\left(v_{a_n} - v_{a_{n-1}}\right) - \beta\left(v_{a_{n-1}} - v_{a_{n-2}}\right) + f_{v_a}(v_a) = 0\\ \beta\left(v_{b_n} - v_{b_{n-1}}\right) - \beta\left(v_{b_{n-1}} - v_{b_{n-2}}\right) + f_{v_b}(v_b) = 0\\ \beta\left(v_{c_n} - v_{c_{n-1}}\right) - \beta\left(v_{c_{n-1}} - v_{c_{n-2}}\right) + f_{v_c}(v_c) = 0\end{aligned} \qquad (11)$$

All three contours intersect in the common nodal point without interacting concerning their particular shape. The energy definition of Equation 11 allows for a minimization process controlling the shape of each contour separately, though ending in one common point exploiting the network topology. The matrix A of Equation 8 is adapted accordingly at the nodal points and their neighbors to fulfill the new definition of the internal energy, i.e. omitting some parts of the banded structure and/or filling up some additional parts to build further connections between different parts of the contour.

The definition of the internal energy for nodes with a degree $\rho(v) > 3$ is straightforward to the proposed method above, i.e. adding further parts to Equation 11. Similarly, the new internal energy is defined at the *end points* of a contour: only one part of Equation 11 is needed, because only one contour without connection to other parts of the contour is available. The adaptation of the matrix A is analogous compared to nodal points. Thus, the control of the shape is feasible by the end of the contour without fixing the end points.

Snakes have the tendency to shorten during the energy minimization due to the first term (α-term) of Equation 4. A shortening of contours with an open ending can be avoided by chaining the end points at the image border allowing for movement only along the image borders (cf. Fig. 1). Alternatively, the contour can be chained at a topologically neighbored object allowing for movement only along the object border. When there are no neighbored objects to allow for chaining the open endings of the contour, a possible idea could be the introduction of an external constraint force regarding a constant length of the contour.

4. EXEMPLARY RESULTS

Results concerning the functionality and capability of network snakes are presented in this section. Two different applications are shown to point out the transferability of the methodology: the extraction of field boundaries from high resolution satellite imagery and the extraction of cells from microscopic cell imagery.

4.1 Extraction of Field Boundaries

The delineation of field boundaries within the complex environment of vegetation is accomplished with color satellite images having a resolution of two meters. The strategy for extracting the objects of interest is divided into two parts: First, a segmentation is carried out in a coarse scale to derive the topology taking into account a somewhat inaccurate geometrical position. The topology of the segmentation, however, is assumed to be correct. In a second step, network snakes are used to improve the preliminary results exploiting the local image features and the object topology.

The initial segmentation of the imagery is briefly outlined below, for details refer to (Butenuth and Heipke, 2005). The use of prior knowledge from a GIS enables a partition of the image scene: Field boundaries are only located within the open landscape and, in addition, the road network describes already fixed field boundaries, because fields naturally end at these objects. Within these regions of interest a multi-channel region growing is performed using the RGB- and IR-channels of the image resulting in an initial segmentation of the image (cf. Fig. 2a, black lines). Note, that the geometrical correctness of the segmentation has been artificially degraded to emphasize the following steps in a better way.

The result of the segmentation is used to derive the topology (cf. Fig. 2b) and to initialize the network snake. Since the objects to be extracted are rather straight, the parameter β is set to a large value compared to α. Thus, image noise and small disturbances have relatively small effects and, in addition, relatively coarse initial values of the contour can be used. The open endings of the contour are chained to the image borders, and are allowed for movement only along the borderline. In Figure 2c the capability of the network snake is highlighted: the contours and in particular the nodal points with a degree $\rho(v) \geq 3$ and the end points move to the correct result, although the initialization is rather poor. The underlying image consists of the standard

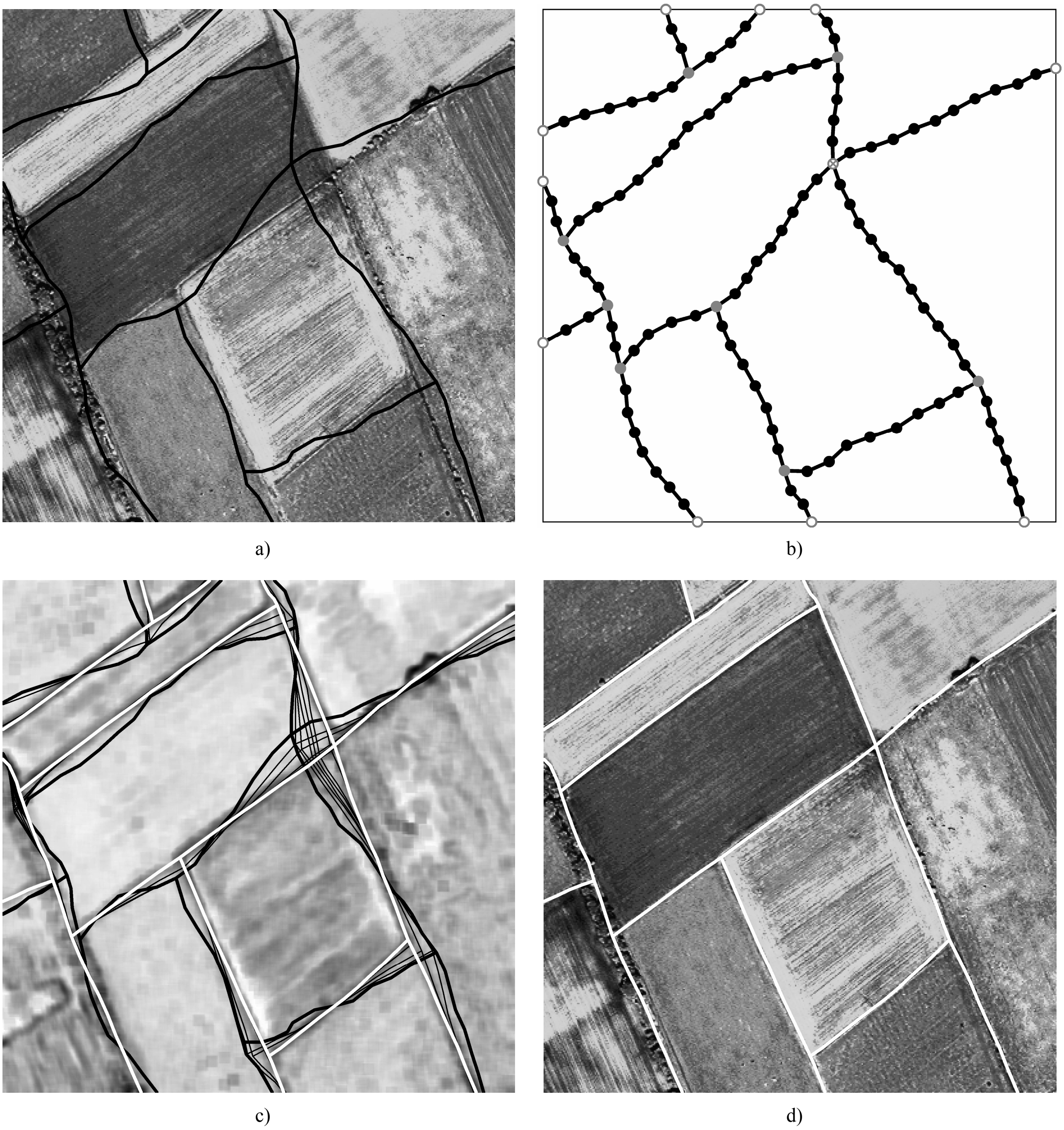


Figure 2. Extraction of field boundaries from high resolution satellite imagery (400 × 400 pixels): a) initialization of the network snake (black); b) topology; c) initialization (black), movement (thin black) and result of the snake (white); d) result superimposed on the intensity channel of the CIR-image

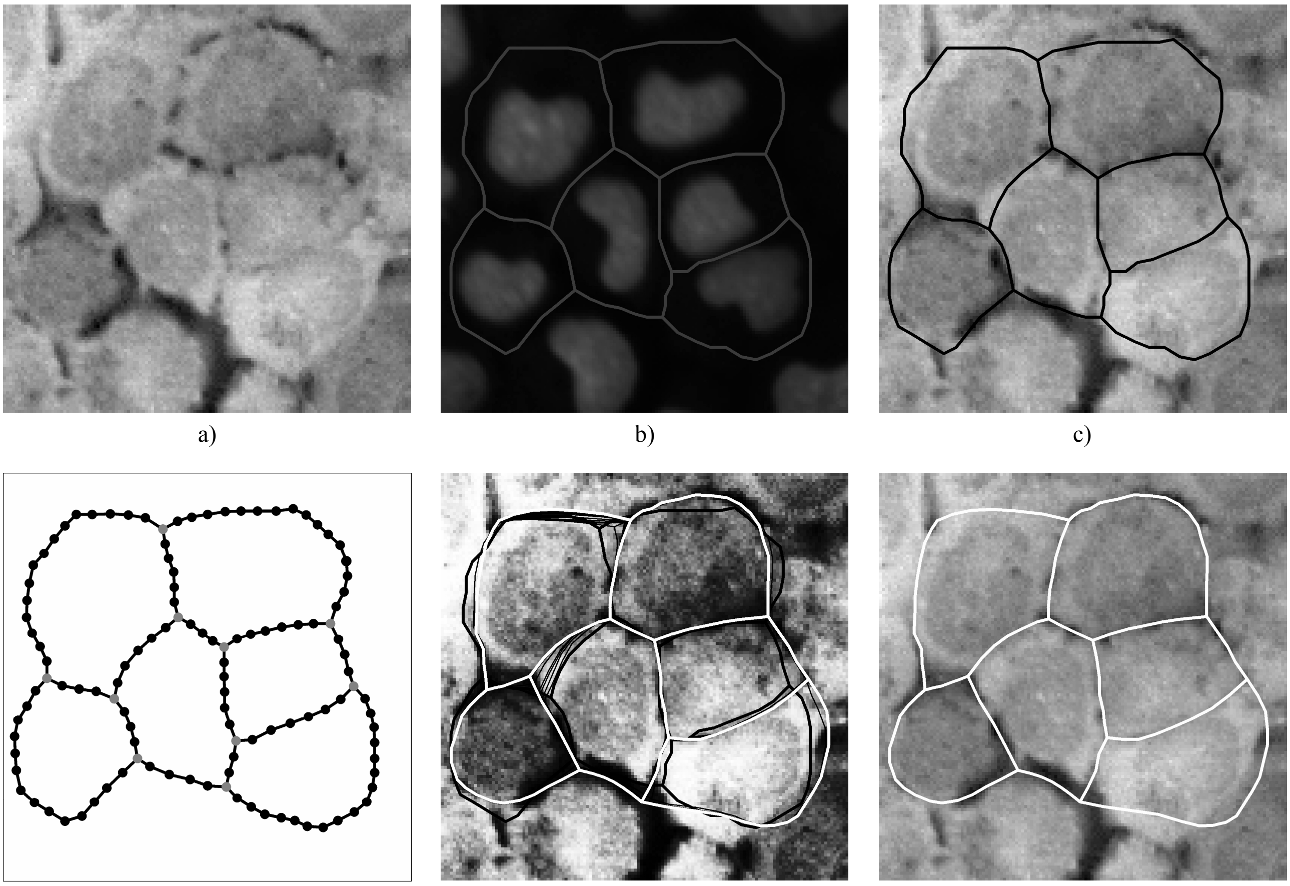


Figure 3. Extraction of cells from microscopic cell imagery (135 × 135 pixels): a) cell image; b) cell nuclei and derived initialization (gray); c) initialization of the network snake (black); d) topology; e) initialization (black), movement (thin black) and result of the snake (white); f) result superimposed on cell image [imagery provided by Evotec Technologies]

deviation of the image intensities of the CIR-image within a quadratic mask, because high values typically belong to field boundaries. Regarding the area around the nodal points, in particular the point with a degree $\rho(v)$ = 4, the internal energy and the exploitation of the topology specify the movement of the contour during the first iteration steps, because the values of the image energy are without effect. Step by step, the inverted image energy helps with small values pushing the contour respectively the nodal points to the correct solution during the minimization process. The final result is depicted in Figure 2d superimposed on the satellite image. The geometrical correctness is in most parts convincing, only the tree rows on the left side prevent a clearly defined field boundary and, thus, the image energy can not support the energy minimization process in an optimal manner. However, the example demonstrates, network snakes are a powerful methodology to delineate objects precisely exploiting their topology.

4.2 Extraction of Cells

The delineation of adjacent cells in microscopic cell imagery is the second example presented in this section. Figure 3a shows a microscopic image of stained cytoplasm, which fluoresced during the data capture. The depicted image has a size of about 20 × 20 micrometers. Again, the strategy for extracting the objects of interest is divided into two parts: At first, a coarse object contour is needed to initialize the processing and to derive the topology. In a subsequent step the geometrical correctness of the initial object contour is optimized using network snakes.

The cell nuclei are much easier to detect than the boundaries of the cytoplasm (cell membrane), because they are well defined against the background (cf. Fig. 3b). The background of the cell nuclei image is segmented followed by a calculation of the skeleton (cf. Fig. 3b, gray line). Due to the fact, that each cell nucleus is located within the associated cell membrane, the skeleton can be used to initialize the network snake having a correct topology even though the geometrical correctness is not very high. Before starting the optimization process, the object contour is thinned out taking into account an equal distance between each node (cf. Fig. 3c, black line).

The topology of the object contour is derived to set up the network snake (cf. Fig 3d). Since the objects to be extracted have a specific curvature, the parameter β controlling the internal energy of the energy minimization process is not set to such a large value compared to the extraction of mostly straight field boundaries (cf. Section 4.1), yet is set again larger than α. A histogram linearization of the original image is accomplished to ease the optimization, because the enhanced contrast of the image helps to push the contour to the correct solution. In Figure 3e the optimization process is highlighted: Starting from the initialization (black line) the movement of the network snake (thin black line) is depicted resulting in extracted cell boundaries (white line). In Figure 3f the final result is superimposed on the cell image. The optimization process works well, although there is strong image noise and in parts the image intensities can not help yielding the correct result. In the lower right part of the image one cell is not delineated completely, because the initialization is too far away and the image energy is not able to push the contour to the true object boundary.

5. CONCLUSIONS

A new segmentation methodology to delineate objects precisely from imagery using network snakes is presented in this paper. Using traditional snakes, adjacent objects, which influence each other and objects forming a network or having open endings are not defined due to the representation of the internal energy. In contrary, network snakes exploit the topology of the objects of interest during the energy minimization process comprising a complete shape control of the contours. The exploitation of the topology turns out to be a powerful method to deal with noise and disturbances in the imagery. The obtained object contours represent a superior geometrical solution when interacting with each other compared to traditional snakes.

Different results concerning the extraction of field boundaries from high resolution satellite imagery and the extraction of cells from microscopic cell imagery demonstrate the potential and the transferability of the proposed methodology. In addition, the two examples emphasize, the requirement of a given correct topology can be achieved in particular applications. However, when the assumption of a given correct topology can not be guaranteed, an additional intervention comprising the global view of the optimized network has to be considered to insert or delete parts of the contour. Possible further applications are the delineation of other objects such as road networks, and other topics such as the update of GIS-data with an already given topology.

In addition, the control of the internal energy can be improved choosing varying values of the parameters α and β, if the modeled object shape has these characteristics. A further interesting question is the behavior of the iteration process: are there dependencies of the initialization, internal energy and image characteristics and can they be exploited? For example, if the object of interest has other shape characteristics than the image disturbances or noise, the control of the internal energy and the exploitation of the topology can allow for a relatively coarse initialization. Investigations regarding this problem can give specific answers about the required quality of the image data and the initialization.

REFERENCES

Aplin, P. and Atkinson, P. M., 2004. Predicting Missing Field Boundaries to Increase Per-Field Classification Accuracy. *Photogrammetric Engineering & Remote Sensing*, Vol. 70, No. 1, pp. 141-149.

Blake, A. and Isard, M., 1998. *Active Contours*. Springer, Berlin Heidelberg New York, 351 p.

Burghardt, D. and Meier, S., 1997. Cartographic Displacement Using the Snake Concept. In: Förstner, Plümer (eds.), *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, Basel, Birkhäuser Verlag, pp. 59-71.

Butenuth, M. and Heipke, C., 2005. Network Snakes-Supported Extraction of Field Boundaries from Imagery. In: Kropatsch, Sablatnig, Hanbury (eds.), *Lecture Notes in Computer Science*, Vol. 3663, Springer Verlag, pp. 417-424.

Caselles, V., Kimmel, R. and Sapiro, G., 1997. Geodesic Active Contours. *International Journal of Computer Vision*, Vol. 22, No. 1, pp. 61-79.

Cohen, L. D. and Cohen, I., 1993. Finite Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 11, pp. 1131-1147.

Cohen, L. D., 1991. On Active Contour Models and Balloons. *CVGIP: Image Understanding*, Vol. 53, No. 2, pp. 211-218.

Dickinson, S. J., Jasiobedzki, P., Olofsson, G. and Christensen, H. I., 1994. Qualitative Tracking of 3-D Objects using Active Contour Networks. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, pp. 812-817.

Fua, P., Grün, A. and Li, H., 2000. Optimization-Based Approaches to Feature Extraction from Aerial Images. In: Dermanis, Grün, Sanso (eds.), *Geomatic Methods for the Analysis of Data in the Earth Sciences*, Lecture Notes in Earth Science, Vol. 95, Springer, pp. 190-228.

Jasiobedzki, P., 1993. Adaptive Adjacency Graphs. *SPIE*, Vol. 2031 Geometric Methods in Computer Vision II, San Diego, pp. 294-303.

Jones, T. R., Carpenter, A. and Golland, P., 2005. Voronoi-Based Segmentation of Cells on Image Manifolds. In: Liu, Jiang, Zhang (eds.), *Lecture Notes in Computer Science*, Vol. 3765, Springer, pp. 535-543.

Kass, M., Witkin, A. and Terzopoulus, D., 1988. Snakes: Active Contour Models. *International Journal of Computer Vision*, Vol. 1, pp. 321-331.

Laptev, I., Mayer, H., Lindeberg, T., Eckstein, W., Steger, C. and Baumgartner, A., 2000. Automatic Extraction of Roads from Aerial Images Based on Scale Space and Snakes. *Machine Vision and Applications*, No. 12, pp. 23-31.

Malladi, R., Sethian, J. A. and Vemuri, B. C., 1995. Shape Modeling with Front Propagation: A Level Set Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, pp. 158-175.

McInerney, T. and Terzopoulos, D., 1996. Deformable Models in Medical Image Analysis: A Survey. *Medical Image Analysis*, Vol. 1(2), pp. 91-108.

Suri, J. S., Kamaledin Setarehdan, S. and Singh, S., 2002. *Advanced Algorithmic Approaches to Medical Image Segmentation: State-of-the-Art Applications in Cardiology, Neurology, Mammography and Pathology*. Springer Verlag, 668 p.

Torre, M. and Radeva, P., 2000. Agricultural Field Extraction from Aerial Images Using a Region Competition Algorithm. *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXIII, Amsterdam, No. B2, pp. 889-896.

Zimmer, C. and Olivio-Marin, J. C., 2005. Coupled Parametric Active Contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 11, pp. 1838-1842.

DETECTABILITY OF BUILDINGS IN AERIAL IMAGES OVER SCALE SPACE

Martin Drauschke, Hanns-Florian Schuster,Wolfgang Förstner

Institute of Photogrammetry, University of Bonn

KEY WORDS: Building Detection, Scale Space, Feature Extraction, Stable Regions

ABSTRACT:

Automatic scene interpretation of aerial images is a major purpose of photogrammetry. Therefore, we want to improve building detection by exploring the "life-time" of stable and relevant image features in scale space. We use watersheds for feature extraction to gain a topologically consistent map. We will show that characteristic features for building detection can be found in all considered scales, so that no optimal scale can be selected for building recognition. Nevertheless, many of these features "live" in a wide scale interval, so that a combination of a small number of scales can be used for automatic building detection.

1 INTRODUCTION

Building detection from aerial images is a very active research area in photogrammetry, cf. the review in (Mayer, 1999). Early attempts go back into the eighties (Nevatia and Price, 1982, Herman and Kanade, 1987, Huertas and Nevatia, 1988). In most cases roof edges or roof parts have been used to identify complex buildings, as facades usually are more difficult to extract. Though some approaches concentrate on simple building types, such as gabled or hipped roof type buildings, they are not generic enough to deal with the great variety of building structures.

Interestingly, all approaches use image features extracted at a single scale, which however, is either given by the resolution of the images, or in some reasonable way selected by a human interpretor. Of course, in general at resolutions of 5 to 30 cm pixel size at ground level building easily can be detected by humans. They obviously exploit the rich context on top of a building and around it. When building an automatic interpretation system, modeling context is one of the most difficult tasks. We want to reduce the demands for modeling context by automatically selecting the optimal scale for image feature extraction.

Technically, detection is inferring existence from observable image features using some classification procedure; localization is only a side effect, precise boundaries are not of primary concern at this stage, cf. the approach of (Brunn and Weidner, 1998) using an image pyramid. Detection may be based on Bayes' rule $P(B|f(R)) \propto P(f(R)|B)P(B)$, where the posterior probability $P(B|f(R))$ for detecting a building or building part given some features $f(R)$ of a region R requires the likelihood $P(f(R)|B)$ and some a priori information about the occurrence of a building. Training a classifier essentially consists of determining parameters p of an adequate likelihood function $l(p) = P(f(R)|B,p)$. This approach assumes the type of features $f(R)$ to be known.

We want to explore the observability of image features, which may be relevant for building detection. Especially, we want to investigate the suitability of regions extracted over scale space. Small changes of scale often do not affect most of the regions however, may lead to extinction of certain regions by merging with neighbored regions, cf. fig. 1.

Such investigations are important for evaluation the mapping potential in the context of human image interpretation, cf. (Jacobsen, 1997) or for evaluating the observability of objects in images in the context of automatic interpretation, e. g. for road mapping procedures (Mayer et al., 1998, Baumgartner et al., 1999, Pakzad and Heller, 2004).

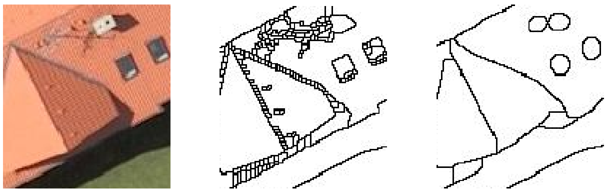


Figure 1: Effect of a scale change onto segmentation. Left: image section of roof part with dormer, chimney, windows and antenna. Middle: segmentation with scale $\sigma = 1$. Right: segmentation with scale $\sigma = 4$. The regions belonging to minor roof parts, e. g. small parts of the dormer's roof or smaller shadows, do not live over a larger range of scales. Most of them merge with other regions with increasing scale.

In our context we need to consider the complexity of roof structures when deciding on the type of image features. Whereas *reconstruction* implicitly aims at a geometric description, and therefore uses features based on edges, cf. e. g. (Nevatia and Price, 1982), or edge-aggregates such as corners, cf. e. g. (Lang and Förstner, 1996), *detection* appears to better rely on features based on regions, especially their form.

Scale space for region extraction has already been investigated, cf. the review (Harvey et al., 1997). In contrast to the blob detection approach of (Lindeberg, 1993), we are interested in a complete partitioning of color images, not restricting to local maxima of intensity or a certain color. Therefore, we propose to use the watersheds of the gradient image, cf. also (Olsen and Nielsen, 1997). Additionally, we adopt the idea of finding maximally stable regions over scale, which are regions whose area does not change over scale, similar to the approach of MSER (Matas et al., 2002) which searches for region which are stable over intensity level sets. At the moment we do not exploit the hierarchical structure of the regions, as e. g. (Bangham et al., 1999) and (Kuiper et al., 2003).

The goal of this paper is to investigate the suitability of such regions for building detection. We will derive a statistics about the scale occurrence of certain roof parts, such as triangular or rectangular roof planes, which is a first attempt to derive the likelihood function for building part detection. Using only a single scale will turn out not to be sufficient for capturing the region information contained in an image, as we will show in our empirical investigations.

The paper is organized as follows: Section 2 describes the segmentation procedure. Sect. 3 presents our approach to measure the stability of regions. Sect. 4 investigates the suitability of the regions over scale space. Sect. 5 discusses the results and gives an outlook on future research.

2 SEGMENTATION

Our segmentation is based on the watershed boundaries derived from a sequence of images in the Gaussian scale space of the gradient magnitude.

The Gaussian scale space is built with logarithmicly ranged scales $2^{i/n}\sigma_0, i = -N_1, ..., 0, ..., N_2$ starting at $\sigma_{-N_1} = 2^{-N_1/n}\sigma_0$ and leading to $\sigma_{N_2} = 2^{N_2/n}\sigma_0$. In our experiments we use $\sigma_0 = 1$, $n = 10$, and $N_2 = 30$, thus scales between 1 and 8 with steps of a ten'th octave. The $N_1 = 17$ scales between 0 and 1 continue the arrangement of the larger scales into the scale between 0.3 and 1. Smaller scales are useless to compute, because the smoothing has nearly no effect.

For each scale $\sigma = \sigma_i$ the three band image $\boldsymbol{f} = [f_c], c = 1, 2, 3$ is convolved with a Gaussian filter $G(x, y, \sigma)$:

$$\boldsymbol{f}(x, y, \sigma) = \boldsymbol{f}(x, y) * G_\sigma(x, y). \tag{1}$$

As input function for the watershed algorithm we use the total gradient of the color images $\boldsymbol{f}(x, y, \sigma)$ as homogeneity measure: For each channel $f_c(x, y, \sigma)$ we compute the squared gradients $||\nabla f_c(x, y, \sigma)||^2$. In order to compensate for the different noise characteristics in the three color channels the homogeneity then is the sum of the squared gradients over all channels c weighted with the inverse of the variance σ_{n_c} of the noise

$$g(x, y, \sigma) = \sqrt{\sum_{c=1}^{3} \frac{||\nabla f_c(x, y, \sigma)||^2}{\sigma_{n_c}^2}} \tag{2}$$

in each channel. (Brügelmann and Förstner, 1992) have shown that the median of the squared gradients, except for a factor, is a good estimation for the noise variance. Therefore, we apply this approach and get the channel specific noise variance by

$$\sigma_{n_c}^2 = \mathrm{med}_{x,y}(||\nabla f(x, y, \sigma)||^2). \tag{3}$$

In order to eliminate noise effects we use as input function for the watershed algorithm

$$h(x, y, \sigma) = \max(g(x, y, \sigma), m_g) \tag{4}$$

where

$$m_g = \mathrm{med}_{x,y}(g(x, y, \sigma)). \tag{5}$$

The watershed algorithm takes the local minima of the input function as seed points and performs a region growing. This gives us a complete partitioning of the image. The result is a label image

$$l(x, y, \sigma) = \mathrm{WS}\,[h(x, y, \sigma)] \tag{6}$$

that has the same labels at the catchment region of the local minima. It can be thought as flooding the basins, if the input function is seen as height values of a virtual landscape. All border pixels of watershed regions are labeled 0.

3 STABILITY OF REGIONS OVER SCALE SPACE

Regions which show only little variation over a certain scale range can be termed stable. There are various metric and topological criterions for measuring stability of regions, but nevertheless the area is the most important one: The region size changes dramatically when regions merge or split. Other region's properties do not change that much over scale.

For obtaining the stability of the L_σ regions $R(l, \sigma), l = 1, ..., L_\sigma$ at scale σ , we compute the area $|R(l, \sigma)|$ of each region from the histogram of $l(x, y, \sigma)$ in (6). We build a set of images where each region is labeled with its area

$$a(x, y, \sigma) = |R(l(x, y, \sigma), \sigma)| \tag{7}$$

We then analyse the *area function*

$$a(\sigma|x, y) \tag{8}$$

for manually selected points (x, y) over the scales. Taking points in regions with a selected content allows to investigate the stability of these regions, thus their usefulness for detection.

In order to evaluate the stability of the area from the area function $a(\sigma|x, y)$, we have to consider the uncertainty of the area of regions. Areas can be categorized as stable in case their area lies in the error band over a large enough range of scales. We require stability over at least 10 scale space layers, i. e. over at least one octave.

The uncertainty of the area A of a polygon $[\boldsymbol{p}_j]$ with J chord lengths d_j between the two neigboring points $\boldsymbol{p}_{j-1}$ and $\boldsymbol{p}_j$ can be shown to be

$$\sigma_A^2 = \frac{1}{4} \sum_j^J d_j^2\, \sigma_p^2 \tag{9}$$

if all points have the same standard deviation σ_p and taking the indices j cyclically, cf. (Förstner, 1999). In case of dense points and a smooth boundary the standard deviation of the area

$$\sigma_A = \frac{U}{\sqrt{J}} \sigma_p \tag{10}$$

reveals to be only dependent on the length U of the boundary, the number of border points J and the standard deviation σ_p of the points. We use a 3 σ_A-error bands assuming a positional error of $\sigma_p = 0.5$ [pixel] to estimate the accuracy of the region's boundary.

4 EMPIRICAL INVESTIGATION

The empirical investigation aims at exploring the usefulness of regions over scale space for building detection. A region is useful, if we can expect that features, which are distinctive for separating building parts from non-building parts, can be derived automatically. Therefore we select roof regions in a supervise mode by picking a point in the region, identify the scale range for stable regions from the area function $a(\sigma|x, y)$ and evaluate these regions visually with respect to their usefulness. Though this is subjective, it gives a clear indication whether there is a chance at all, that stable and relevant regions may be found. We also want to find out whether there are characteristic scales for different classes of roof regions.

4.1 Basic categories of roof planes

Since roofs are the most distinctive parts of a building seen in an aerial image, there seems to be no limit for the complexity of urban roof structures. We model our roof prototypes by examining their roof planes.

Although there exist various catalogs of basic roof styles due to roof construction, we define roof prototypes in a different way. This is due to the high complexity of roof structures, which can be only partially categorized by classical roof styles. Therefore our detection scheme does not aim at such a categorization of complete roofs, but only on categories of roof planes.

Fig. 2 shows some basic roof styles of suburban buildings. Each of these roofs can be modeled by triangles and tetragons. Some planes of half-hipped roofs are hexagons. Of course, more complex buildings, e. g. L-shaped buildings, show other shapes, e. g. parallelograms or skew trapezoids. However, taking roof regions as key-features for triggering building detection does not require distinguishing between planes of major roof and those of dormers (see fig. 3).

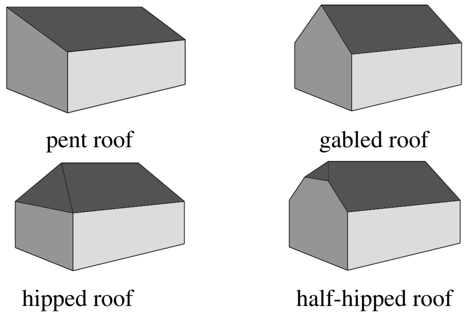


Figure 2: Common types of Middle European roofs

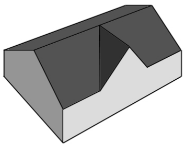


Figure 3: Example for building with a dormer

We therefore represent each roof by the roof planes together with their geometric traits and by their adjacency graph, possibly including attributes of the type of neighborhood.

Besides dormers the roof planes can get disturbed by other objects fixed on the roof. Examples of these objects are chimneys, windows and solar cells (see fig. 1, left). Furthermore, roofs can be occluded by trees or other buildings and their shadows. These disturbances may affect the region detection. Whereas some of these, e. g. antennas and their shadows, will not be visible at lower resolution, thus at other scales, others such as occlusions will change the form of the extractable regions, however, be visible over a larger range of scales.

At this stage of our investigation we are only interested in the observability of stable regions, which show roof-type structures.

4.2 Test Data

Our experiments are based on aerial image data, showing suburban buildings in the cities of Bonn (Germany), Graz (Austria), and Toyonaka (Japan).

Bonn: We consider 13 aerial images taken over Bonn, having a ground resolution of 10 cm, cf. figs. 4 and 5. The first example image shows a scene of a shopping area with bigger flat and gabled roofs. In the other example image, there is a suburban scene with gabled roofs often having additional roof parts as dormers or windows. Due to the time of image acquisition in winter, the vegetation around the buildings does not show a strong contrast. Additionally, as the position of the sun is quite low, the shadows often reach to the neighboring building.

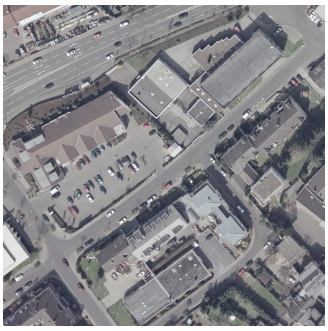


Figure 4: Image section from Bonn, shopping area, 10 cm ground resolution

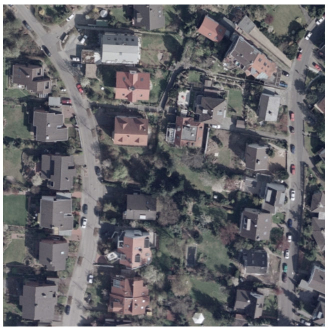


Figure 5: Image section from Bonn, residential area, 10 cm ground resolution

Graz: Our 14 test images of Graz have a ground resolution of 8 cm (cf. fig. 6). Most of the roofs are covered with red tiles, the buildings are surrounded by fresh vegetation. There are only small shadows in the picture, the roof planes are only disturbed by other objects, such as chimneys or solar cells. The images show many gabled and cross gabled roofs.

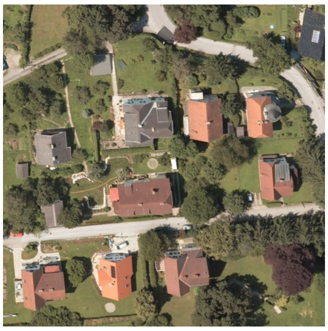


Figure 6: Image section Graz, residential area, 8 cm ground resolution, kindly provided by Vexcel Imaging GmbH in Graz

Toyonaka: Our 9 test images of Toyonaka have a ground resolution of 7 cm (cf. fig. 7). The concentration of buildings differs strongly from the other test images. Roofs with colorful tiles are detectable by eye very well. Due to the low position of the sun, shadows make it difficult to distinguish between dark covered roofs. There are no other objects on the roofs, but most of the houses are extended by additional building parts. Our test image does not show any vegetation and is weak in contrast. Most of buildings have hipped or pyramid roofs.

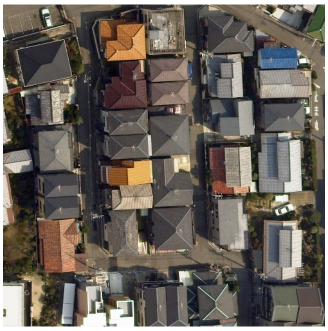


Figure 7: Image section Toyonaka, dense residential area, 7 cm ground resolution, kindly provided by Vexcel Imaging GmbH in Graz

4.3 Experimental Results

We selected roof planes manually to observe the stability of their area in scale space. Tab. 1 shows the number of stable regions we found in all images. We distinguished the regions by their shape. The row of more complex shaped regions refers to those regions, which have melted together with other regions of the roof still forming characteristic roof shapes. Less than 20% of the selected regions were not stable at all, these regions are not taken into account any further.

Shape	Bonn	Graz	Toyonaka	Σ
triangle	20	71	151	242
square	55	95	16	166
rectangle	205	373	152	730
trapezoid	39	60	115	214
more complex	56	60	176	292
Σ	375	659	610	1644

Table 1: Statistics of selected regions which turned out to be stable.

Fig. 9, an extract of fig. 5, demonstrates that at various scales relevant roof areas can be detected. We obtain a thin recangular shaped roof plane, which merges with another one at $\sigma \approx 3$ [pixel]. As long as the balconies form bays at the bottom of the region, it is not considered to be stable. In contrast to the balcony bays, the hole of the region belonging to the window is very stable.

Fig. 10, an extract of fig. 4, demonstrates that even smaller roof parts as dormer roofs can be stable, too. From scale $\sigma = 4$ [pixel] on, the region merges with a vegetation area in front of the building.

Fig. 11, an extract of fig. 5, shows the smoothing deforms roof part with increasing scale. The region is stable over various scale space layers, but the shape of the original region changes from a triangle to a circle within the last 10 layers, resp. starting at $\sigma \approx 4$ [pixel].

Fig. 12, an extract of fig. 7, shows a problem of our manual region selection: The Japanese roofs are often strong textured in opposite to the most European roofs. In this case, one almost always selects a border point of region at at least one scale space layer between $\sigma = 0$ and 1 [pixel]. So, the area function $a(\sigma|x, y)$ is not determined from a region but from all border pixels in the image (which have the same label: 0). From $\sigma = 1$ [pixel] on, the roof planes are well observable.

The results of our investigation are shown in fig. 8. It is organized in a max-min-diagram that shows the maximal versus the minimal level of scale for the rectangular roof parts, measuring the scales in dm at ground level. The other roof parts show similar results, the graphics would present nearly the same range of positions, the density of the drawn dots would only be less.

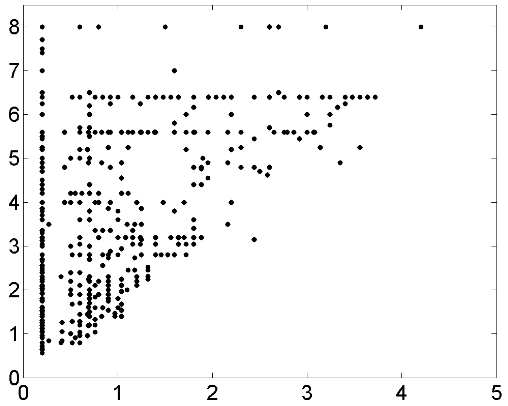


Figure 8: The results of the investigation on rectangular regions are drawn in a max-min-interval diagram. The results of the different shaped regions are similar to those as tab. 2 shows.

Obviously, we have very stable regions, where the minimum scale is small and the maximum scale is large, namely those in the upper left of the diagram. We also find regions which only live in large scales as those few in the upper right of the diagram. We finally find regions which only exist at small scales, i. e. those in the lower left.

There is no certain scale where uniquely formed regions can be found. Also, choosing a certain scale for finding regions, say 3 dm, would only allow to detect those regions which are in the upper left rectangular having its lower right corner at (3,3), thus missing quite some relevant regions, in the lower left and the upper right of the diagram.

The relevance of the extracted regions has only been evaluated in general: over appr. 80% of all regions are stable over at least an octave in the investigated scale range.

Tab. 2 also contains the range of the minimal and maximal scales in [dm]. The different forms appear in all scale ranges. As the minimum and maximum ranges of the scales almost totally overlap, selecting a single scale in this overlap would lead to a loss in region detection, e. g. when choosing $\sigma = 3$ dm and searching for rectangular roof regions.

Type	$\sigma_{\min}$ [dm]	$\overline{\sigma_{\min}}$	$\sigma_{\max}$ [dm]	$\overline{\sigma_{\max}}$
	min – max	[dm]	min – max	[dm]
triangle	0.21 – 3.56	0.58	0.64 – 8.00	3.13
square	0.21 – 4.00	0.61	0.56 – 8.00	3.37
rectangle	0.21 – 4.15	0.74	0.56 – 8.00	3.95
trapezoid	0.21 – 4.15	0.59	0.56 – 8.00	3.48
others	0.21 – 4.25	0.86	0.72 – 8.00	3.64

Table 2: Range of minimal and maximal scales over all stable regions, additionally their means, distinguishing the shapes of regions.

5 CONCLUSION

This paper is a first investigation into the detectability of building roofs via regions which are stable in scale space. The stability of a region can be measured by the scale range where the region's area does not change significantly.

We used the watershed algorithm on the averaged and weighted gradient magnitude image for image partitioning. The weights are the inverse of the noise variance in the different channels. These regions turned out to be quite stable over scale.

We have shown, that regions that represent roofs and roof parts in aerial images can only be extracted in certain intervals of scale. However, there is no optimal scale for the extraction of roof parts in aerial images. It is necessary to automatically choose the scale for each region.

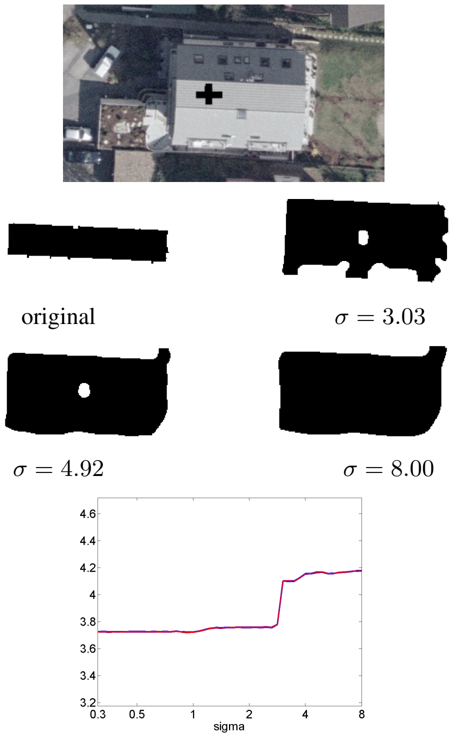
The usefulness of stable regions was explored. Over appr. 80% of the roof regions, which were selected manually, lead to image regions which were stable and promised to have attributes for reliable detection.

We are currently investigating the automatic extraction of regions which are stable over scale space. We are setting up an annotated image database, which makes it possible to train our building detectors.

The approach should be easily transferred to other types of images, as such region detectors exploiting scale space can be expected to play at least a prominent role as point type detectors.

Acknowledgements

This work was done within the project "Ontological scales for automated detection, efficient processing and fast visualisation of landscape models" which is supported by the German Research Foundation (DFG). The authors would also like to thank Ribana Roscher and Susanne Wenzel for assisting at the evaluation.


original

$\sigma = 3.03$

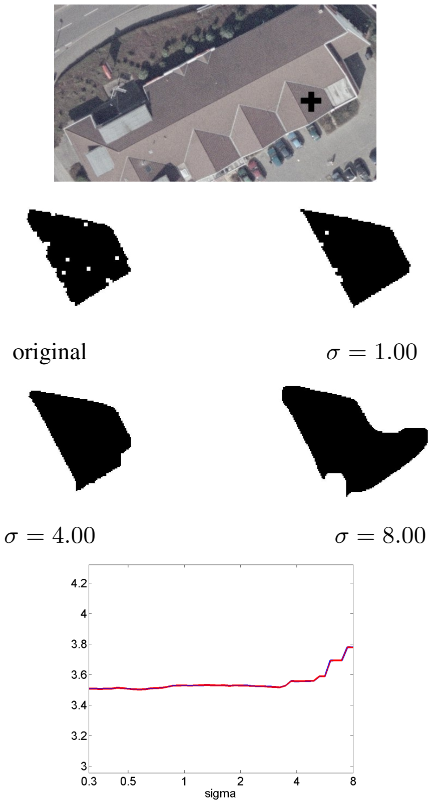
$\sigma = 4.92$

$\sigma = 8.00$

Figure 9: Example 1 of the development of a region: $\log(a(\sigma|x, y))$. Starting with a thin rectangular roof plane, the region merges at higher levels with other roof planes and roof objects. Graph: Relation between smoothing scale and region sizes together with the error bands. Stable Regions allude to concrete scales.

REFERENCES

Bangham, J. A., Moravec, K., Harvey, R. and Fisher, M., 1999. Scale-space trees and applications as filters for stereo vision and image retrieval. In: BMVC, pp. 113–143.


original

$\sigma = 1.00$

$\sigma = 4.00$

$\sigma = 8.00$

Figure 10: Example 2 of the development of a region: $\log(a(\sigma|x, y))$. Starting with a trapezoid part of a dormer, the region merges at higher levels with a vegetation area in front of the building. Graph: Relation between smoothing scale and region sizes together with the error bands.

Baumgartner, A., Steger, C., Mayer, H., Eckstein, W. and Ebner, H., 1999. Automatic road extraction based on multi-scale, grouping, and context. Photogrammetric Engineering & Remote Sensing 65, pp. 777–786.

Brügelmann, R. and Förstner, W., 1992. Noise estimation for color edge extraction. In: W. Förstner and S. Ruwiedel (eds), Robust Computer Vision, Wichmann, Karlsruhe, pp. 90–107.

Brunn, A. and Weidner, U., 1998. Hierarchical bayesian nets for building extraction using dense digital surface models. Journal for Photogrammetry & Remote Sensing 53(5), pp. 296–307.

Förstner, W., 1999. Areas and their uncertainty. Technical report, Institute of Photogrammetry, University of Bonn.

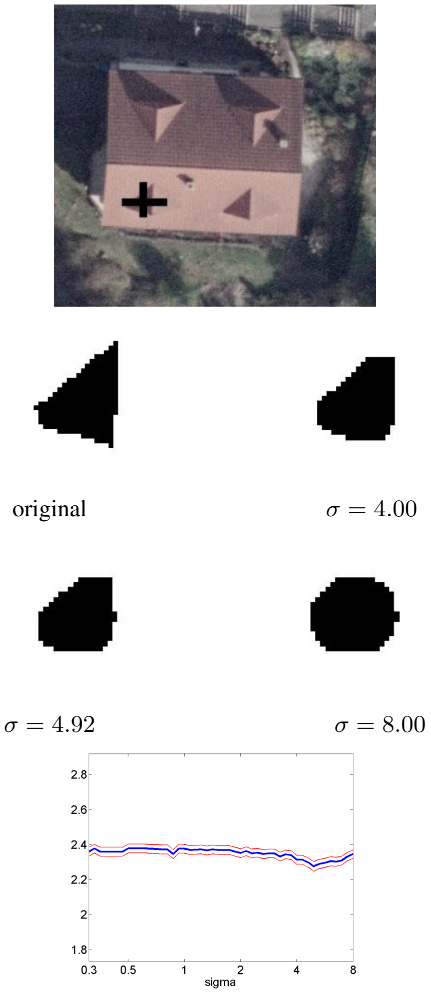
Harvey, R., Bangham, J. A. and Bosson, A., 1997. Scale-Space Filters and Their Robustness. Vol. Lecture Notes in Computer Science 1252, pp. 341–344.

Herman, M. and Kanade, T., 1987. The 3d mosaic scene understanding system: Incremental recognition of 3d scenes from complex images. In: Fischler/Firschein (ed.), Readings in Computer Vision, Kaufmann, pp. 471–482.

Huertas, A. and Nevatia, R., 1988. Detecting building in aerial images. CVGIP 41, pp. 131–152.

Jacobsen, K., 1997. Comparison of information contents of different space images. In: Joint Workshop "Sensors and Mapping from Space".

Kuiper, A., Florack, L. and Viergever, M., 2003. Scale space hierarchy. JMIV 18, pp. 169–189.


original

$\sigma = 4.00$

$\sigma = 4.92$

$\sigma = 8.00$

Figure 11: Example 3 of the development of a region: $\log(a(\sigma|x, y))$. Deformation of a region's shape by image smoothing. Graph: Relation between smoothing scale and region sizes together with the error bands.

Lang, F. and Förstner, W., 1996. Surface reconstruction of man-made objects using polymorphic mid-level features and generic scene knowledge. In: International Archives of Photogrammetry and Remote Sensing, Part B3, Vol. 31, pp. 415–420.

Lindeberg, T., 1993. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. IJCV 11(3), pp. 283–318.

Matas, J., O.Chum, M.Urban and T.Pajdla, 2002. Robust wide baseline stereo from maximally stable extremal regions. In: BMVC, Vol. 1, pp. 384–393.

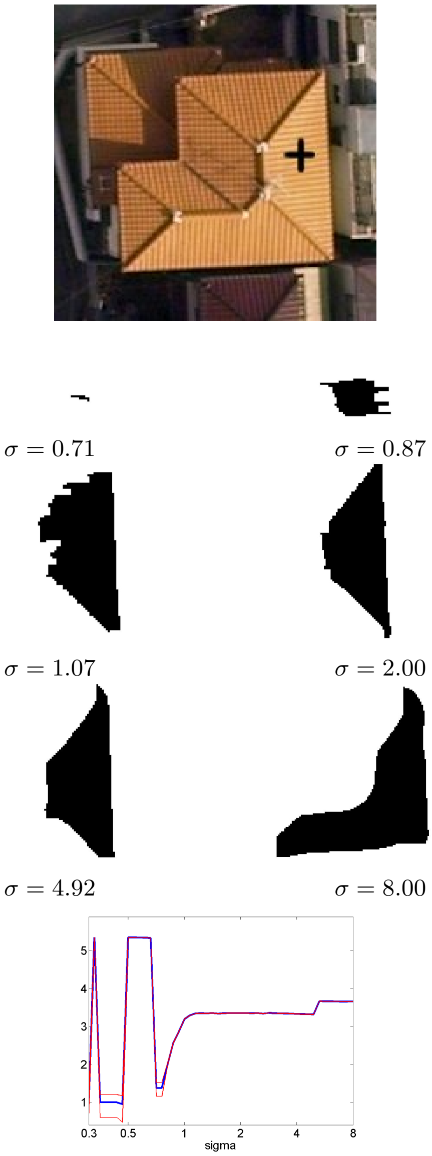
Mayer, H., 1999. Automatic object extraction from aerial imagery - a survey focusing on buildings. Computer Vision and Image Understanding 74, pp. 138–149.

Mayer, H., Laptev, I. and Baumgartner, A., 1998. Multi-scale and snakes for automatic road extraction. In: Fifth European Conference on Computer Vision (2), pp. 720–733.

Nevatia, R. and Price, K. E., 1982. Locating Structures in Aerial Images. IEEE Transactions on Pattern Analysis and Machine Intelligence (5), pp. 476–484.

Olsen, O. F. and Nielsen, M., 1997. Multiscale gradient magnitude watershed segmentation. In: ICIAP'97 - 9th International Conference on Image Analysis and Processing, Vol. Lecture Notes in Computer Science 1310, pp. 9–13.

Pakzad, K. and Heller, J., 2004. Automatic scale adaptation of semantic nets. Publikationen der DGPF 13, pp. 67–76.


$\sigma = 0.71$

$\sigma = 0.87$

$\sigma = 1.07$

$\sigma = 2.00$

$\sigma = 4.92$

$\sigma = 8.00$

Figure 12: Example 4 of the development of a region: $\log(a(\sigma|x, y))$. The strong oscillation of the area refers to wrongly selected regions. Graph: Relation between smoothing scale and region sizes together with the error bands.

CONSISTENT ESTIMATION OF BUILDING PARAMETERS CONSIDERING GEOMETRIC REGULARITIES BY SOFT CONSTRAINTS

Franz Rottensteiner

Institute of Photogrammetry and Remote Sensing, Vienna University of Technology,
Gußhausstraße 27-29, A-1040 Vienna, Austria - fr@ipf.tuwien.ac.at

Cooperative Research Centre for Spatial Information,
723 Swanston Street, The University of Melbourne, VIC 3010, Australia – franzr@unimelb.edu.au

Commission III, WG III/4

KEY WORDS: Building Reconstruction, Parameter Estimation, Constraints

ABSTRACT:

This paper describes a model for the consistent estimation of building parameters that is a part of a method for automatic building reconstruction from airborne laser scanner (ALS) data. The adjustment model considers the building topology by GESTALT observations, i.e. observations of points being situated in planes. Geometric regularities are considered by "soft constraints" linking neighbouring vertices or planes. Robust estimation can be used to eliminate false hypotheses about such geometric regularities. Sensor data provide the observations to determine the parameters of the building planes. The adjustment model can handle a variety of sensor data and is shown to be also applicable for semi-automatic building reconstruction from image and/or ALS data. A test project is presented in order to evaluate the accuracy that can be achieved using our technique for building reconstruction from ALS data, along with the improvement caused by adjustment and regularisation. The planimetric accuracy of the building walls is in the range of or better than the ALS point distance, whereas the height accuracy is in the range of a few centimetres. Regularisation was found to improve the planimetric accuracy by 5- 45%.

1. INTRODUCTION

The shapes of buildings and other man-made objects, despite being very complex in realistic scenes, are often characterised by certain geometrical regularities. At a level of detail typical for topographic mapping (mapping scales 1:500 to 1:1000) most buildings can be modelled by polyhedrons. This implies that all vertices belonging to a face must be situated on a plane in object space. Apart from that, other geometrical regularities include perpendicular walls, horizontal roof edges, or symmetry between roof faces.

It is the goal of automatic building reconstruction to generate 3D building models from sensor data in previously detected regions of interest. In this context, model regularisation by considering geometric constraints is essential for achieving high quality building models. Besides resulting in a more regular visual appearance, considering geometric regularities helps to improve the geometric accuracy of the models, especially if the sensor geometry is weak. There are two general strategies for building reconstruction, differing in the way buildings are represented in the reconstruction process and thus also in the way geometric regularities are considered. The first strategy is based on a bottom-up process. The sensor data are segmented in order to obtain 3D features such as edges and planes. These features are combined to obtain a polyhedral model, e.g. (Rottensteiner et al., 2005). Alternatively, buildings can be reconstructed by parametric primitives in a top-down process, e.g. (Brenner, 2000). In the first case, assumptions on geometric regularities may or may not be used in order to select the 3D features and group them; they can and should be considered as additional information in a final parameter estimation process yielding consistent and regularised building models. In the second case, assumptions about regularities, e.g. rectangular footprints, are an implicit part of the description of the primitives. Using parametric primitives reduces the level of detail that can be achieved as the number of primitives is usually small and most have a rectangular footprint. This can be avoided by using "adaptive primitives" (Rottensteiner & Schulze, 2003), i.e. primitives having an adaptive parameterisation. However, the bottom-up strategy seems to be more flexible with respect to handling geometric regularities. They are not an implicit part of the building model, but rather are added as additional information to the estimation of the building parameters and thus only have to be considered where enough evidence is found in the data. From the point of view of parameter estimation, this can be handled in two ways. First, geometric regularities can be considered in the adjustment by constraint equations. This strategy will result in models precisely fulfilling these "hard" constraints. Brenner (2005) has given an overview about the ways such constraints can be handled in object modelling. The alternative is to add "soft constraints", i.e. direct observations for entities describing a geometric regularity, to the adjustment of the sensor-based observations. In this case, the constraints will not be fulfilled exactly, but there will be residuals to the observations. The degree to which the constraints are fulfilled depends on the stochastic model. Using the second strategy, robust estimation techniques can be applied to the soft constraints to determine whether a hypothesis about a geometric regularity fits to the sensor data or not.

Vosselman (1999) proposed an algorithm for building reconstruction from airborne laser scanner (ALS) data that determined building outlines under the assumption of all neighbouring walls intersecting at right angles. He addressed the necessity of adding constraints to the estimation of the model parameters without doing so himself. Vögtle and Steinle (2000) reconstruct buildings from ALS and spectral data. The coordinates of their building vertices are estimated by local adjustment only, and no geometric regularities are considered. Alharty and Bethel (2004) describe two methods for roof outline detection. The first method relies on the existence of a dominant roof direction and the neighbouring walls being orthogonal. The second does not require such assumptions, but no overall adjustment is carried out, and no geometric regularities are considered. Ameri (2000) describes a general adjustment model for building reconstruction from image data. Geometric constraints are considered. For instance, for two orthogonal building edges a direct observation of the inner product of the directional vectors is introduced. The weighting of such an algebraic observation seems to be somewhat critical. A method for fitting building models to multiple aerial images using “hard” constraints was presented in (Vallet & Taillandier, 2005). McGlone (1996) describes the mathematical basis for handling geometrical constraints both as (“hard”) condition equations and as “soft” constraints, using this basis for improving the results of multiple-image point matching under the assumption of certain object regularities.

In (Rottensteiner et al., 2005) we have presented a method for automatic building reconstruction from ALS data that is based on the detection and combination of roof planes. The final step of building reconstruction is an overall adjustment of all observations to determine the model parameters consistently. The adjustment model was originally presented in (Rottensteiner, 2003), but implemented only recently. It is the first goal of this paper to present this adjustment model in its improved and revised form and to show how it can be used as a tool for consistent estimation of building parameters for different types of available sensor data. Special emphasis is laid on the way geometric regularities can be considered. The second goal of this paper is to evaluate the results of building reconstruction from ALS data by comparing automatically derived building models to reference data. This comparison should also show how effective the overall adjustment is in improving the geometric quality of building models.

2. WORKFLOW FOR BUILDING RECONSTRUCTION

Our algorithm for building reconstruction requires ALS points and a coarse approximation of the building outlines. The ALS data are sampled into a Digital Surface Model (DSM) in the form of a regular grid of width Δ by linear prediction. The work flow consists of three steps (Rottensteiner et al., 2005):

1. **Detection of roof planes** based on a segmentation of the DSM. These planes are expanded by region growing.
2. **Grouping of roof planes and roof plane delineation:** Co-planar roof planes are merged, and hypotheses for intersection lines and/or step edges are created based on an analysis of the neighbourhood relations of the roof planes.
3. **Consistent estimation of the building parameters** to improve these parameters using all available sensor data and considering geometric constraints.

In step 2, the boundary polygons of the roof planes are determined as a combination of roof plane intersections and step edges, the step edges being located in the DSM by an edge extraction technique taking into account specific information about buildings. Decisions in the determination of the shapes of the roof polygons are based on hypotheses tests and robust estimation. We use the concept of uncertain projective geometry (Heuel, 2004) for consistent modelling of the stochastic properties of all geometric entities. In this paper, we will focus on the final step of the reconstruction process.

3. THE ADJUSTMENT MODEL

The adjustment problem we want to solve can be described as follows. We assume to have given a polyhedral building model in boundary representation (B-rep). The model consists of planar faces, loops, edges, and vertices. Each edge is the intersection of two neighbouring faces, and each vertex is the intersection of at least three planes of the model. All vertices belonging to the boundary of a face have to lie in the face’s plane. The faces of the model are labelled as being a roof face, a wall, or the floor. Walls are modelled to be strictly vertical. The topology of the model and some meaningful initial values for its parameters are assumed to be known. The initial model can be the outcome of the bottom-up strategy for building reconstruction (cf. section 2). In this case it is an approximate version of the final model, and its initial parameters are already derived in some way from the sensor data. The coarse model has to be analysed for geometric regularities, which can be done automatically or based on the interaction of a human operator, and the model parameters have to be estimated. For that purpose, we use five categories of observations:

1. Observations representing the topology of the model
2. Observations corresponding to geometric regularities
3. Sensor and sensor-derived observations
4. Observations linking the sensor observations to the model
5. Direct observations for unknowns to avoid singularities.

They are used to determine four categories of unknowns:

1. The co-ordinates of the model vertices
2. The parameters of the model planes
3. Transformation parameters, e.g. the unknown angle for each pair of perpendicular walls (cf. section 3.2)
4. Additional unknowns, e.g. unknown object co-ordinates for each ALS point (cf. section 3.3.2).

Our method for handling the model topology and geometric regularities is independent not only from the types of sensor data that are used, but also from the way in which the original model was created. The adjustment model is based on the program ORIENT for hybrid photogrammetric adjustment, especially on its concept of handling object space constraints by “GESTALT” observations (Kager, 2000).

3.1 Observations Representing Model Topology

It is the idea of our method to find a mapping between the B-rep of the polyhedral model and a system of GESTALT observations representing the model topology in adjustment. GESTALT observations are observations of a point **P** being situated on a polynomial surface (Kager, 2000). The polynomial is parameterised in an observation co-ordinate system (u, v, w) related to the object co-ordinate system by a shift $\mathbf{P_0}$ and three rotations $\mathbf{\Theta} = (\varpi, \phi, \kappa)^T$. The actual observation is **P**‘s distance from the surface which has to be 0. Using $(u_R, v_R, w_R)^T = \mathbf{R}^T(\mathbf{\Theta}) \cdot (\mathbf{P} - \mathbf{P_0})$, with $\mathbf{R}^T(\mathbf{\Theta})$ being a transposed rotational matrix parameterised by $\mathbf{\Theta}$, and restricting ourselves to vertical planes for walls and tilted planes for roofs, there are three possible formulations of GESTALT observation equations:

$$
\begin{aligned}
r_u &= \frac{m_u \cdot u_R + a_{00} + a_{01} \cdot m_v \cdot v_R}{\sqrt{1 + a_{01}^2}} \\
r_v &= \frac{m_v \cdot v_R + b_{00} + b_{10} \cdot m_u \cdot u_R}{\sqrt{1 + b_{10}^2}} \\
r_w &= \frac{m_w \cdot w_R + c_{00} + c_{10} \cdot (m_u \cdot u_R) + c_{01} \cdot (m_v \cdot v_R)}{\sqrt{1 + c_{10}^2 + c_{01}^2}}
\end{aligned}
\tag{1}
$$

In equation 1, r_i are the corrections of the fictitious observations of co-ordinate i and $m_i \in \{-1, 1\}$ are mirror coefficients. An application is free to decide which of the parameters (**P**, $\mathbf{P_0}$, **Θ**, a_{jk}, b_{ik}, c_{ij}) are to be determined in adjustment and how to parameterise a surface. In addition, different GESTALTs can refer to identical transformation or surface parameters, which will be used to handle geometric regularities (cf. section 3.2). Here, we declare the rotations to be 0 and constant. $\mathbf{P_0}$ is a point situated inside the building and constant. For each face of the B-rep of the building model, we define a set of GESTALT observations, taking one of the first two equations 1 for walls and the third one for roofs. The unknowns to be determined are the object co-ordinates of each vertex **P** and the plane parameters *(a_{jk}, b_{ik}, c_{ij})*. As each vertex is neighboured by at least three faces, the co-ordinates of the vertices are determined from these GESTALT observations and thus need not be observed directly in the sensor data. Further, these observations link the vertex co-ordinates to the surface parameters and thus represent the building topology in the adjustment. They do already enforce geometric constraints by modelling walls as being strictly vertical and by declaring all vertices of a face to lie in the same plane. The stochastic model of these GESTALT observations is described by the a priori standard deviation σ_T of the fictitious distance between a point and the plane.

3.2 Observations Representing Geometric Regularities

Geometric regularities are considered by additional GESTALT equations, taking advantage of specific definitions of the observation co-ordinate system and specific parameterisations of the planes. Geometric regularities can occur between two planes or between two vertices of the model. In the current implementation, we restrict ourselves to regularities involving planes or vertices being neighbours of one edge. In all cases, the observation co-ordinate system is centred in one vertex $\mathbf{P_1}$ of that edge and the *w*-axis is vertical, thus $\varpi = \phi = 0 = const$. Four types of geometric regularities are considered (Figure 1). The first type, a horizontal roof edge, involves the edge's end points: Its two vertices $\mathbf{P_1}$ and $\mathbf{P_2}$ must have identical heights. The two points are declared to be in a horizontal plane ε_h that is identical to the *(u,v)* – plane of the observation co-ordinate system. One observation is inserted for $\mathbf{P_2}$: $r_w = w_R = Z_2\text{-}Z_1$.

The other cases involve the two neighbouring planes of an edge. One of the axes of the observation coordinate system is defined to be the intersection of these two planes ε_1 and ε_2. There is one additional unknown rotational angle κ describing the direction of the *u*-axis. For each vertex $\mathbf{P}_i$ of the planes, GESTALT observations are added for ε_1 or ε_2. For the edge's second vertex $\mathbf{P}_2$ two observations (one per plane) are added. The GESTALT observations for ε_1 and ε_2 are parameterised in a specific way:

- The edge is the intersection of two horizontal and symmetric roof planes ε_1 and ε_2. There is only one tilt parameter c^1_{01}. Symmetry is enforced by selecting $m_v = -1$ for ε_2:

$$
\varepsilon_1 : r_w = \frac{w_R + c_{01}^1 \cdot v_R}{\sqrt{1 + \left(c_{01}^1\right)^2}}; \qquad \varepsilon_2 : r_w = \frac{w_R - c_{01}^1 \cdot v_R}{\sqrt{1 + \left(c_{01}^1\right)^2}} \tag{2}
$$

- The edge is the intersection of two perpendicular walls: ε_1: $r_u = u_R$, ε_2: $r_v = v_R$. There is no additional surface parameter to be determined.
- Two walls are identical and the edge does not really exist in the object: ε_1: $r_v = v_R$, ε_2: $r_v = v_R$. There is no additional surface parameter. $\mathbf{P}_1$ and/or $\mathbf{P}_2$ might become undetermined, so that direct observations for one of the co-ordinates of these vertices have to be generated.

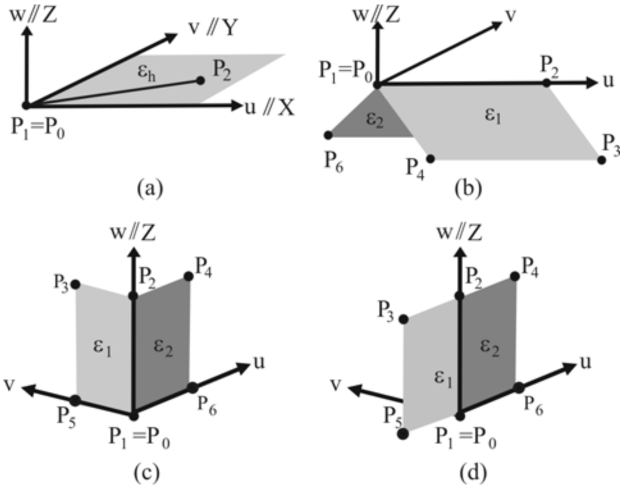


Figure 1. (a) Horizontal edge; (b) horizontal and symmetric edge; (c) perpendicular walls; (d) Identical walls.

The stochastic model of these GESTALT observations is described by their a priori standard deviations σ_C. The "soft constraints" thus modelled will only be fulfilled up to a degree depending on σ_C. The GESTALT observations corresponding to the geometrical constraints can be subject to robust estimation for gross error detection. If the sensor observations contradict the constraints, the respective GESTALT observations should receive large residuals, which can be used to modulate the weights in an iterative robust estimation procedure (Kager, 2000). Thus, if the GESTALT observations describing a geometric constraint are eliminated in adjustment, this means that the hypothesis about a constraint was wrong.

Whether or not a hypothesis about a constraint is introduced can be decided in several ways. For instance, the coarse model can be analysed whether the angles between neighbouring walls differ from 90° by less than a threshold ε_α, and a constraint about perpendicular walls can be inserted if this is the case. More sophisticated methods can take into account the stochastic properties of the coarse model. In a semi-automatic working environment, geometric constraints can be inserted (and enforced) by the user. The principle can be expanded to the definition of parametric primitives by generating more complex systems of constraints between the planes of a building (Rottensteiner & Schulze, 2003).

3.3 Sensor Observations and Observations Linking the Sensor Data to the Model

The observations described so far link the plane parameters to the vertices or to the parameters of other planes. In order to determine the surface parameters, observations derived from the sensor data are necessary. ORIENT can handle a large variety of sensor models. Any of these sensors or any combination of them can be used in adjustment. Here we will restrict ourselves to image and ALS data.

3.3.1 Image co-ordinates: Points measured in images are related to object space by the perspective equations. We assume the orientation parameters of the images to be known and constant. An observed image point has to be assigned to an entity of the object model to contribute to the determination of the model parameters. Two cases can be distinguished. First, an image point can be assigned to a building vertex, which yields two perspective observation equations for that vertex. Second, the image point can be assigned to a model edge. As such a point is not a part of the model, its object co-ordinates have to be determined as additional unknowns; however, each point assigned to an object edge yields four additional observations: its two image co-ordinates and two GESTALT observations (one for each object plane intersecting at the object edge). The stochastic model of an image co-ordinate is described by its standard deviation σ_I. Depending on the way the image points were determined, σ_I can describe the accuracy of manual measurement, or it can be the result of a feature extraction process.

3.3.2 ALS data: ALS points give support to the determination of the roof plane parameters. As an ALS point is not a part of the model, its object co-ordinates have to be determined as unknowns. Each ALS point gives four observations, namely its three co-ordinates and one GESTALT observation for the roof plane the point is assigned to. As the walls only receive few laser hits, their parameters have to be determined from other observations. Walls correspond to sections of step edges in the DSM (Rottensteiner et al., 2005). Each step edge section is derived from "edge points" in the DSM (e.g. points of maximum height gradient). In order to determine the walls, these edge points have to be used as observations in a way similar to the original ALS points: Each edge point gives three observations (its X and Y co-ordinates and 1 GESTALT), but two additional unknowns (again X and Y). The ALS observations can be modelled in two different ways: They can be introduced as "control point" observations, i.e. as direct observations for the object co-ordinates, or they can be introduced as "model points". In the latter case, the ALS points are linked to the object co-ordinate system by a rigid motion, and the six parameters of that rigid motion are estimated in the adjustment. Using this variant, local shifts and rotations of the ALS co-ordinate system with respect to the object co-ordinate system that might be the result of systematic GPS and INS errors of the ALS system can be compensated. This only makes sense if additional data, e.g. aerial images, are available. Otherwise, the ALS and the object co-ordinate systems are assumed to be identical. The stochastic model of an ALS point is described by two standard deviations: σ_{XY} for its planimetric co-ordinates and σ_Z for its height. The edge point co-ordinates are introduced with a standard deviation σ_E.

3.4 Overall Adjustment

All observations are used in an overall adjustment process. The weights of the observations are determined from their a priori standard deviations. Correlations between the observations (e.g. between the x and y image co-ordinates of an image point) are not considered. Robust estimation is carried out by iteratively re-weighting the observations depending on their normalised residuals in the previous adjustment (Kager, 2000). The re-weighting scheme is only applied to the sensor observations and to the observations modelling geometric constraints, in order to eliminate gross observation errors and wrong hypotheses about geometric regularities. The surface parameters and the vertex co-ordinates determined in the adjustment are used to derive the final building model.

4. EVALUATION

4.1 The Test Data

For our test, we selected 8 buildings of different size and complexity out of a larger test area in Fairfield (NSW). They were chosen to highlight the method's potential to handle buildings of both regular and irregular shapes. Both ALS and image data were available for that test site. The ALS data were captured using an Optech ALTM 3025 laser scanner with a nominal average point distance of 1.25 m. As our test buildings were at the edge of a swath, there was a relatively irregular point density, with point distances of about 0.5 m in flight direction and 1.5 - 2 m across flight direction. The aerial images were a stereo pair taken at a scale of 1:11000 (focal length f = 30 cm). They were scanned at a resolution 15 µm, which corresponds to a ground sampling distance of 0.17 m.

4.2 Generating Reference Data

The aerial images were used to determine the reference data for the test. In a semi-automatic working environment, the roof polygons were digitised in the images and hypotheses about geometric regularities were introduced by the human operator. The adjustment model described in section 3 was used to determine the parameters of the reference buildings, taking into account the GESTALT observations, the image co-ordinates of the building vertices, and ALS points to improve the height accuracy of the reference models. The ALS points were necessary because of the weak configuration of the images. Figure 2 shows an upright projection of a reference building resembling a hip roof and the ALS points. Three variants are shown: the results of photogrammetric plotting with and without geometric constraints and the results achieved by combining photogrammetric plotting with geometric constraints and ALS data. For the variant without geometric constraints the RMS values of the height differences of the horizontal eaves is ±0.25 m. In the constrained version, the eaves are horizontal, but the figure reveals that the heights of the eaves derived from the ALS data are about 50 cm lower. The ALS points were introduced as model co-ordinates; the shift was about 15 cm in X and Y and about 5 cm in Z. The precision of the building vertices was about ±17 cm in X and Y, and about ±5 cm in Z.

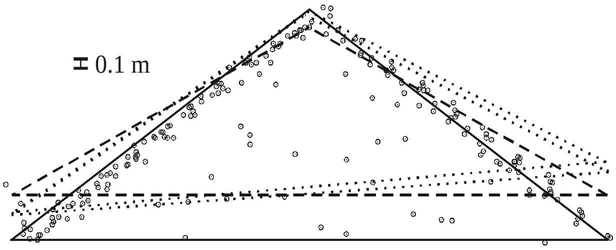


Figure 2. Upright projection of a hip roof (heights enlarged by a factor 2) generated from images without constraints (dotted lines); images with constraints (broken lines); images with constraints and ALS points (full lines). Circles: ALS points.

4.3 Results and Discussion

From the ALS data, a DSM with a grid width of $\Delta = 0.5$ m was generated. From the DSM, roof planes were extracted, and the roof boundary polygons were determined as a combination of intersection lines and step edges in the way described in (Rottensteiner et al., 2005). These initial roof boundary polygons are shown super-imposed to the DSM in Figure 3.

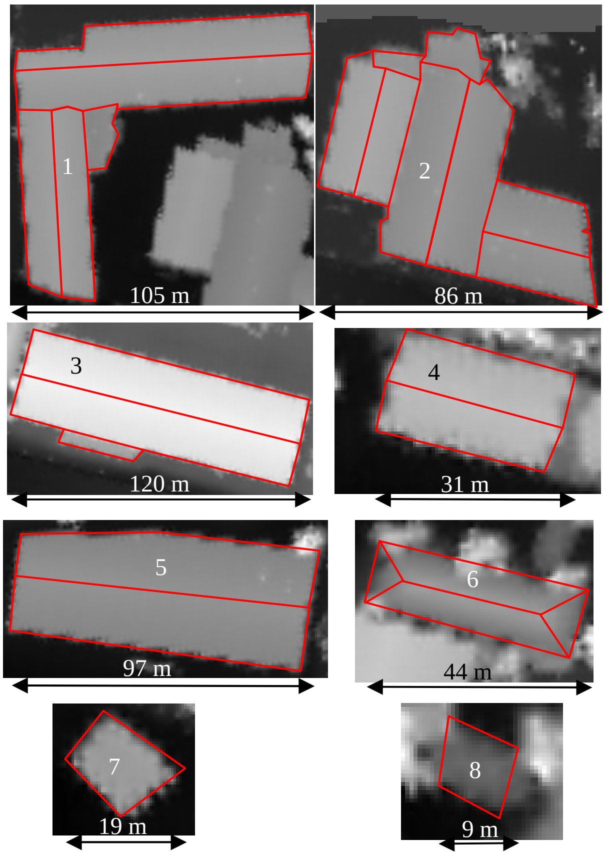


Figure 3. Initial roof boundary polygons for the eight buildings superimposed to the DSM. The buildings are shown in different scales, according to the extents shown in the figure.

In general the models look quite good except for building 8, which is partly occluded by trees. There is some noise in the outlines of buildings 1 and 2. Buildings 4, 6, 7, and 8 and the main part of building 3 should have a rectangular footprint, which is not entirely preserved in the initial models; geometric constraint should help to overcome this situation. The initial models, the original ALS points, and the step edge points provide the input for the overall adjustment. Soft constraints were introduced just on the basis of a comparison of angles/height differences to thresholds. Table 1 gives an overview about the stochastic model for the individual groups of observations in adjustment. Robust estimation was applied to the soft constraints and to the ALS and step edge points. In the current implementation this had to be done in a supervised way. It turned out that with some larger buildings the stochastic model had to be changed to make false hypotheses on geometric constraints detectable. Using σ_C = ±0.05 m and σ_E = ±0.25 m turned out to be a good choice. However, the final adjustment without the eliminated observations was carried out using the values given in Table 1. They were confirmed by a variance component analysis.

Topology σ_T [m]	Constraints σ_C [m]	ALS XY σ_{XY} [m]	ALS Z σ_Z [m]	Step Edge σ_E [m]
±0.01	±0.015	±0.25	±0.075	±0.5

Table 1. A priori standard deviations of the observations.

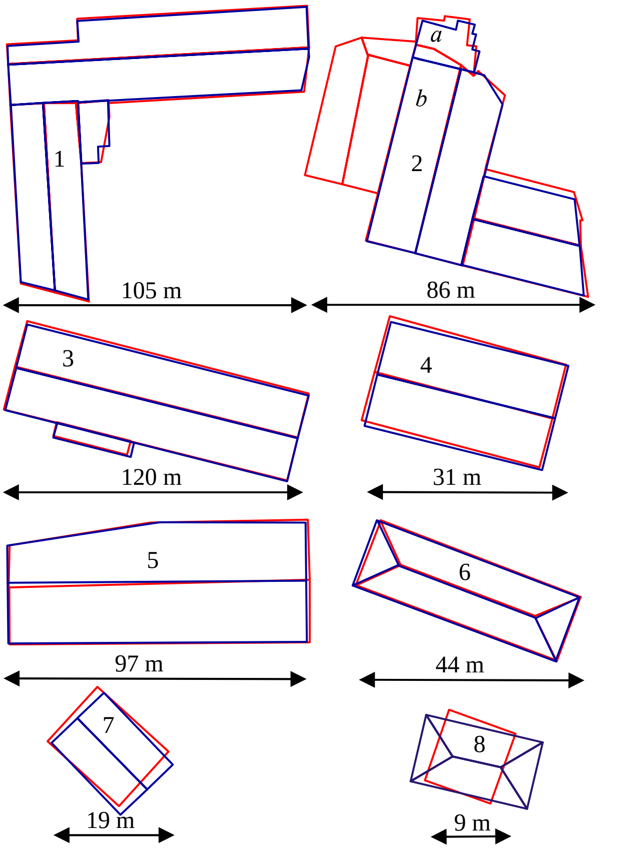


Figure 4. Final roof boundary polygons (red) and reference data (blue). A part of building 2 is missing in the reference data since it only occurs in the ALS data.

Figure 4 gives the final results of building reconstruction and a comparison to the reference data. Compared to figure 3, the building models appear to be more regular. For buildings 1-6 the number of extracted roof planes was correct. The intersection lines are very accurate, and step edges are in general determined quite well, too. Some small roof structures are generalised, e.g. the outline of the smallest roof plane of building 1 or of roof plane *a* of building 2. The step edge between that plane and its neighbouring plane *b* was also not very precisely determined. The problem was that roof plane *a* was horizontal, its western vertex being higher and its eastern vertex lower than the corresponding vertices of roof plane *b*; the maximum height difference was only 0.3 m, so that the step edge was poorly defined. Building 7 was reconstructed as being flat. The intersection of the two roof planes is only 0.15 m lower than the eaves, which is the reason why the two planes were merged. Building 8 was also reconstructed as a flat roof. It was the smallest building in the sample with only a few ALS points on the roof planes, and both ends occluded by trees. The outlines at the occluded ends are not very well detected either. Apart from the visual inspection of the building models, a numerical evaluation of these results was carried out. RMS values of the co-ordinate differences of corresponding vertices in the reconstruction results and the reference data were computed for each roof plane:

$$RMS_{XY} = \sqrt{\frac{\sum\left(\Delta X^2 + \Delta Y^2\right)}{N}} \text{ and } RMS_Z = \sqrt{\frac{\sum \Delta Z^2}{N}} \qquad (3)$$

In Equation 3, *N* is the number of corresponding points in the respective roof plane. If no matching vertex was found, the closest point on the corresponding roof boundary polygon was used instead. For buildings 7 and 8 only the outlines were evaluated. Figure 5 shows a graph of RMS_{XY} and RMS_Z depending on the roof area. RMS_{XY} is smaller than 3.1 m for all roof planes. For most roofs it is in the range between ±0.5 m and ±1.5 m, which is better than the point density across the flight direction. The largest values occur for roof planes smaller than 100 m^2, with the exception of roof planes *a* and *b* of building 2, for reasons discussed above. RMS_Z is much smaller than RMS_{XY} because heights are better defined in ALS data than step edges. RMS_Z becomes smaller with increasing area roof planes because more ALS points give support to large planes. Intersections are more accurately determined than step edges. RMS values computed for intersection lines are only ±0.35 m in planimetry and ±0.07 m in height.

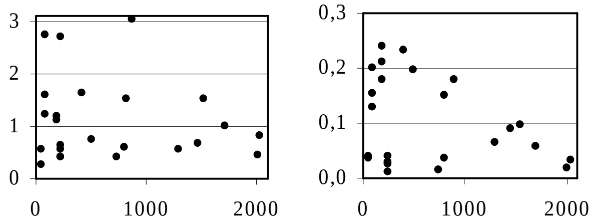


Figure 5. Left: RMS_{XY} [m], right: RMS_Z [m], both depending on the roof area [m^2].

B	*P*	RMS_{XY} [m]	RMS_Z [m]	Δ_{XY} [m]	Δ_Z [m]
1	5	0.76	0.12	0.24	0.01
2	5	2.27	0.20	0.00	-0.02
3	3	0.82	0.10	0.07	0.16
4	2	0.60	0.02	0.13	0.03
5	2	1.31	0.08	-0.08	-0.02
6	4	0.48	0.09	0.36	0.17
7	2	1.43	0.14	0.44	0.03
8	O	2.74	-	-0.02	-

Table 2. *B*: Building; *P*: Number of planes; RMS_{XY}, RMS_Z: Combined RMS values in planimetry / height; Δ_{XY}, Δ_Z: improvement of RMS_{XY} / RMS_Z.

Table 2 gives combined RMS values for all the test buildings. The large value for RMS_{XY} for building 2 of ±2.27 m is caused by the erroneous step edge; the combined value without that edge would be ±1.43 m. For most buildings, RMS_{XY} is better than the average point distance across flight direction. Apart from problems with low step edges, errors occurred at the outlines of some of the larger building due to occlusions: as the test area was at the edge of the swath, the positions of the step edges were very uncertain there. The height accuracy is good, with the largest value of ±0.20 m occurring at building 2, again at the problematic step edge. Table 2 also gives the impact of the overall adjustment to the RMS values. With building 5, the RMS values get worse by a small value after adjustment, but in most cases the RMS values are improved by the overall adjustment. The improvement can be up to 45% (building 6).

5. CONCLUSION

In this paper we have described a model for the consistent estimation of building parameters that is part of a method for the automatic reconstruction of buildings from ALS data. The adjustment model can consider geometric regularities by “soft constraints”, and it can handle different sensor data. It was used not only in the reconstruction process, but also for the generation of reference data for a test project. In the test project, the roof boundary polygons extracted from the ALS data were compared to the reference data. The accuracy was determined to be in the range of or better than the average point distance in planimetry, and about ±0.1 - ±0.2 m in height. The improvement of the model co-ordinates caused by the geometric constraints can be up to 45 %.

ACKNOWLEDGEMENT

The data set used in this study was provided by AAM Hatch (www.aamhatch.com.au).

REFERENCES

Alharty, A., Bethel, J., 2004. Detailed building reconstruction from airborne laser data using a moving surface method. In: *IAPRSIS* XXXV - B3, pp. 213-218.

Ameri, B., 2000. Feature based model verification (FBMV): A new concept for hypotheses validation in building reconstruction. In: *IAPRS* XXXIII-B3A, pp. 24-35.

Brenner, C., 2000. Dreidimensionale Gebäuderekonstruktion aus digitalen Oberflächenmodellen und Grundrissen. PhD thesis, University of Stuttgart. DGK-C 530.

Brenner, C., 2005. Constraints for modelling complex objects. In: *IAPRSIS* XXXIII-3/W24, pp. 49 - 54.

Heuel, S., 2004. *Uncertain Projective Geometry. Statistical Reasoning for Polyhedral Object Reconstruction*. Springer-Verlag, Berlin Heidelberg, Germany.

Kager, H., 2000. Adjustment of Algebraic Surfaces by Least Squared Distances. In: *IAPRS,* Vol. XXXIII-B3, pp. 472–479.

McGlone, C., 1996. Bundle adjustment with geometric constraints for hypothesis evaluation. *IAPRS*, Vol. XXXI-B3, pp. 529–534.

Rottensteiner, F., 2003. Automatic generation of high-quality building models from Lidar data. *IEEE CG&A* 23(6), pp. 42-51.

Rottensteiner, F. and Schulze, M, 2003. Performance evaluation of a system for semi-automatic building extraction using adaptable primitives. In: *IAPRSIS* XXXIV / 3-W8, pp. 47-52.

Rottensteiner, F., Trinder, J., Clode, S., and Kubik, K., 2005. Automated delineation of roof planes in LIDAR data. In: *IAPRSIS XXXVI – 3/W19,* pp. 221-226.

Vallet, B., Taillandier, F., 2005. Fitting Constrained 3D Models in Multiple Aerial Images. In: *BMVC,* accessed 19/07/2006: http://www.bmva.ac.uk/bmvc/2005/papers/paper-57-176.html

Vögtle, T., Steinle, E., 2000. 3D modelling of buildings using laser scanning and spectral information. In: *IAPRS* XXXIII-B3B, pp. 927-933.

Vosselman, G., 1999. Building reconstruction using planar faces in very high density height data. In: *IAPRS* XXXII/3-2W5, pp. 87–92.

CELL DECOMPOSITION FOR THE GENERATION OF BUILDING MODELS AT MULTIPLE SCALES

Norbert Haala, Susanne Becker, Martin Kada

Institute for Photogrammetry, Universitaet Stuttgart
Geschwister-Scholl-Str. 24D, D-70174 Stuttgart, Germany
Forename.Lastname@ifp.uni-stuttgart.de

KEY WORDS: CAD, Data Structures, Representation, Three-dimensional, Point Cloud, Urban, LIDAR, Modelling

ABSTRACT:

Existing tools for 3D building reconstruction usually apply approaches, which are either based on constructive solid geometry (CSG) or boundary representation (B-Rep). After a brief discussion of their respective advantages and disadvantages, the paper will present an alternative approach based on cell decomposition. This type of representation is also well known in solid modelling, and can be used efficiently for building reconstruction. Firstly, topological correct representations of building polyhedrons can be constructed easily from planar surface patches as they can for example be extracted from airborne LIDAR data. Furthermore, constraints between different object parts like co-planarity or right angles can be integrated relatively easy. The approach will be demonstrated exemplarily by a building reconstruction based on airborne LIDAR data and given outlines of the respective buildings. In principle, different levels of generalisation can be defined during reconstruction. This also allows a refinement of an already given building model based on terrestrial LIDAR data as it will be demonstrated in the final part of the paper.

1. INTRODUCTION

Since the acquisition of 3D urban data has become a topic of major interest, a number of algorithms have been made available both for the automatic and semiautomatic collection of building models. Usually, these tools for the generation of polyhedral building models are either based on a constructive solid geometry (CSG) or a boundary representation (B-Rep) approach. In the following, the pros and cons of both approaches will be discussed briefly. This will motivate our new approach for building reconstruction, which is based on cell decomposition as an alternative form of solid modelling.

Within B-Rep approaches, the planar surface boundaries of the reconstructed building are directly generated from measured vertices, edges or faces. If the reconstruction is for example based on 3D point clouds from airborne laser scanning, a triangulation can in principle be directly applied to generate an appropriate surface model. For this purpose, a number of algorithms are available from computer graphics, which automatically compute geometric surface representations from polygonal or triangular meshes. These algorithms include surface simplification processes for reduction or smoothing of the originally measured dense 3D point clouds. By these means, discrete and continuous representations can be generated at different levels of detail, while optimization criteria are used to preserve the original shape. However, while these approaches are suitable for free-form objects, they are usually not adequate for modelling man-made objects such as buildings. Building architecture features special characteristics like right angles or parallel lines. If these constraints are not maintained during surface simplification, the visual impression of the resulting building model will be limited significantly. The human eye is very sensitive to deviations between piecewise flat building objects and their approximation by the meshed surface. Thus, an adequate visualisation will not be feasible for a number of scenarios even when the great computational load of dense meshes from directly triangulated original LIDAR points is accepted.

These deviations are inevitable at least to a certain degree due to limitations in point sampling distance and accuracy of LIDAR sensors. In order to reduce their influence to the final result, a number of B-Rep based approaches first extract planar regions of appropriate size from the LIDAR data. Based on this segmentation, polyhedral building models are then generated from these regions by intersection and step edge generation. However, while numerous approaches are available for the extraction of such building fragments, the combination of these segments to generate topological correct boundary representations is difficult to implement (Rottensteiner 2001). This task is additionally aggravated if geometric constraints, such as meeting surfaces, parallelism and rectangularity have to be guaranteed for respective segments, which have been extracted from measured and thus error-prone data.

Such regularization conditions can be met easier, if object representations based on CSG are used (Brenner 2004). Within CSG based modelling, simple primitives are combined by means of regularized Boolean set operators. An object is then stored as a tree with simple primitives as the leaves and operators at internal nodes. Some nodes represent Boolean operators like union or intersection, or set difference, whereas others perform translation, rotation and scaling. Since modelling using primitives and Boolean operations is much more intuitive than specifying B-rep surfaces directly, CSG is used widely in computer aided design (Mäntylä 1988). CSG representations are also always valid since the simple primitives are topologically correct and this correctness is preserved during their combination by the Boolean operations. Additionally, the implicit geometrical constraints of these primitives like parallel or normal faces of a box type object allows for the quite robust parameter estimation. This is especially important for reconstructions based on error prone measurements. However, semi-automatic reconstruction requires the availability of an appropriate set of primitives. This can be difficult for complex buildings. Additionally, the derivation of a boundary representation from the collected CSG model is required for most visualization and simulation applications. While this so-called boundary evaluation is not difficult conceptually, its correct and efficient implementation can be difficult. Error-prone measurements, problems of numerical precision and unstable calculation of intersections can considerably hinder the robust generation of a valid object topology. These difficulties to provide robust implementations in this context seem to be rooted in the interaction of approximate numerical and exact symbolic data (Hoffmann 1989).

As it will be demonstrated in the paper, the application of cell decomposition can help to facilitate these problems of CSG and B-Rep based 3D building reconstruction. Cell decomposition is a special type of decomposition models, which subdivides the 3D space into relatively simple solids. Similar to CSG, these spatial-partitioning representations describe complex solids by a combination of simple, basic objects in a bottom up fashion. In contrast to CSG, decomposition models are limited to adjoining primitives, which must not intersect. The basic primitives are thus 'glued' together, which can be interpreted as a restricted form of a spatial union operation. The simplest type of spatial-partitioning representations is exhaustive enumeration. There the object space is subdivided by non overlapping cubes of uniform size and orientation, which allows for very simple algorithms. However, due to large memory consumption and the restricted accuracy of the object representation the applicability of exhaustive enumeration is usually limited. These problems can be alleviated while preserving these nice properties of spatial-occupancy enumeration, if other basic elements than just cubes are used. Cell decompositions are therefore based on a variety of basic cells, which may be any objects that are topologically equivalent to a sphere i.e. do not contain holes.

In solid modelling, cell decomposition is mainly used as auxiliary representation for specific computations (Mäntylä 1988). As it will be demonstrated, by a reconstruction algorithm using LIDAR data and given ground plans, cell decomposition can be applied efficiently for the automatic reconstruction of topological correct building models at different levels of detail. In the following section, the basic idea of cell decomposition is introduced by the decomposition of 2D building outlines. As it is demonstrated in section 3, this process can be extended to a 3D building reconstruction by the additional integration of a point cloud from airborne LIDAR. Section 4 will present the refinement of the building models using terrestrial LIDAR, while a concluding discussion will be given in section 5.

2. CELL DECOMPOSITION FOR BUILDING BLOCK APPROXIMATION

Within our cell decomposition approach, the reconstruction of polyhedral 3D building models is based on a subdivision of space into 3D primitives. As input data a 2D building ground plan and a triangulated 2.5D point cloud from airborne laser scanning are used. As first step, a set of space dividing planes is derived from the input data. This subdivision generates a set of primitives that organize the infinite space into building and non-building parts. After these building primitives are glued together, the resulting 3D building model is a good approximation of the real world object. This process is similar to a generalisation approach presented in (Kada 2006) that simplifies the 3D geometry of existing polygonal building blocks. In contrast, now the 3D shape of the building is generated from scratch by a combination of a 2D ground plan and 2.5D airborne LIDAR data as previously mentioned. Afterwards, we use this approximation as the basic building block for further refinement of the façade structure by an additional integration of terrestrial laser scanning data.

2.1 2.5D reconstruction based on ground plans

A first approximation of the 3D building can be generated by an extrusion of the 2D ground plan. The vertical extension conforms to the minimum height value of all LIDAR points within that region. Within the extrusion process each segment of the ground plan generates a polygonal face perpendicular to the horizontal ground plane. However, for easier understanding within the following figures, only the horizontal footprints of these 3D objects are depicted. Thus, within Figure 1 to Figure 3 the planes perpendicular to the horizontal ground plan are depicted as straight lines. However, in addition to these 2D sketches, the reconstruction process will be demonstrated by a sequence of 3D visualisations at the end of this section in Figure 4 and Figure 5.

In order to generate the cell decomposition from the extruded ground plan, a set of subdivision planes is computed in an iterative approach. Since an approximating 3D model is aspired, planar buffers are used to minimize the required number of planes. In doing so, protrusions and other small structural elements that are included in the buffer can be optionally eliminated before any further reconstruction. This process is depicted exemplarily in Figure 1.

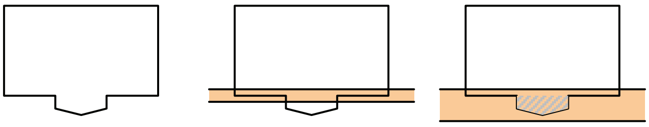


Figure 1: Buffer operation for the generation of approximating planes.

At the beginning of an iteration step, a buffer is created for each polygonal face, which is depicted exemplarily by the highlighted area in Figure 1. Buffers are delimited by two parallel planes, which initially coincide with the plane equation of its defining polygon. As it is shown in the middle and right image of Figure 1 within an iterative process the size of the buffer is adapted to keep track of a set of polygons that are completely inside this buffer region. For new buffers, this set consists solely of their single defining polygon. As a buffer grows, more polygons are inserted.

The total area of the polygons inside this set also denotes the importance of the buffer. The buffers are then tested pair wise against each other. If the orientations of their delimiting planes are approximately the same, the pair is a candidate for a merge. Though, the distance between the delimiting planes of the merged buffers might be higher then a maximum threshold. Therefore, a new buffer is created that contains both sets of polygons and the delimiting planes are adjusted to this set accordingly. A candidate pair is valid if the distance of the delimiting planes of this merge is under the aforementioned threshold. Only valid candidate pairs that create new buffer of high importance are created. The algorithm stops when no more buffers can be merged and the buffer of highest importance is returned for that iteration step. From the set of polygons, an averaged plane equation is calculated in order to create a subdivision plane. The set of polygons inside the buffer is discarded from further processing. By this iterative process, a set of subdivision planes is detected in descending order of importance. In order to preserve right angles and parallelism in the final building model, the orientation of the subdivision planes can further be analyzed and adjusted accordingly. Figure 2 (top) exemplarily shows a given ground plan, while the six vertical subdivision planes are represented by the red lines in Figure 2 (bottom). Additionally, a buffer area for one subdivision plane is depicted by the grey area.

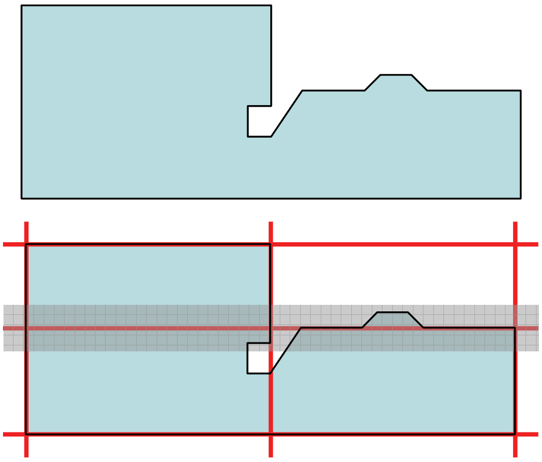


Figure 2: Approximation of 2D building ground plan by six subdivision planes represented by red lines.

2.2 Construction of decomposition cells

Once the subdivision planes have been determined, they are used to create a decomposition of the infinite space. In practice an infinite space is unsuitable, so a solid two times the size of the building's bounding box is used as a substitute. The result is a set of solid blocks. Until now, there is no information available, what subset needs to be glued together to form the final shape. Therefore, the solids are differentiated in building and non-building primitives in a subsequent step. For this purpose, a percentage value is calculated for each primitive that denotes the volume of the original building model inside the respective block (see Figure 3 top). All solids with a percentage value under a given threshold value are then denoted as non-building primitives and are discarded from further processing. Because the primitives are rather coarse, a threshold value of around 50% is suitable in most cases.

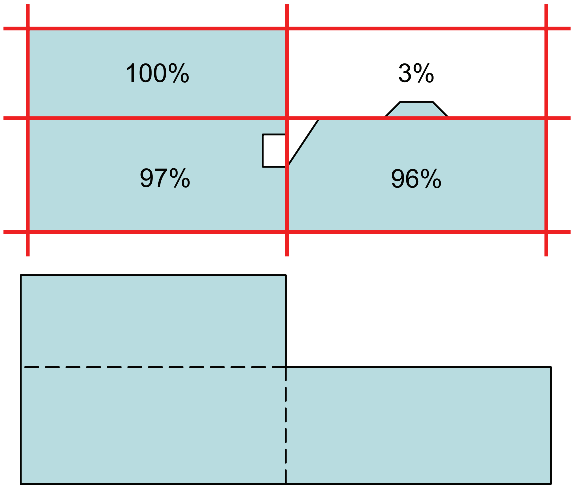


Figure 3: Building fragments with computed overlap to the original ground plan (top) and combination of building cells (bottom).

When glued together as depicted in Figure 3 (bottom) the building blocks form a flat-top approximation that is shaped after the original ground plan. However, the cell decomposition simplifies the reconstruction of the roof structure from airborne laser scanning data. This comes from the fact that the roof can be reconstructed per cell and not per building

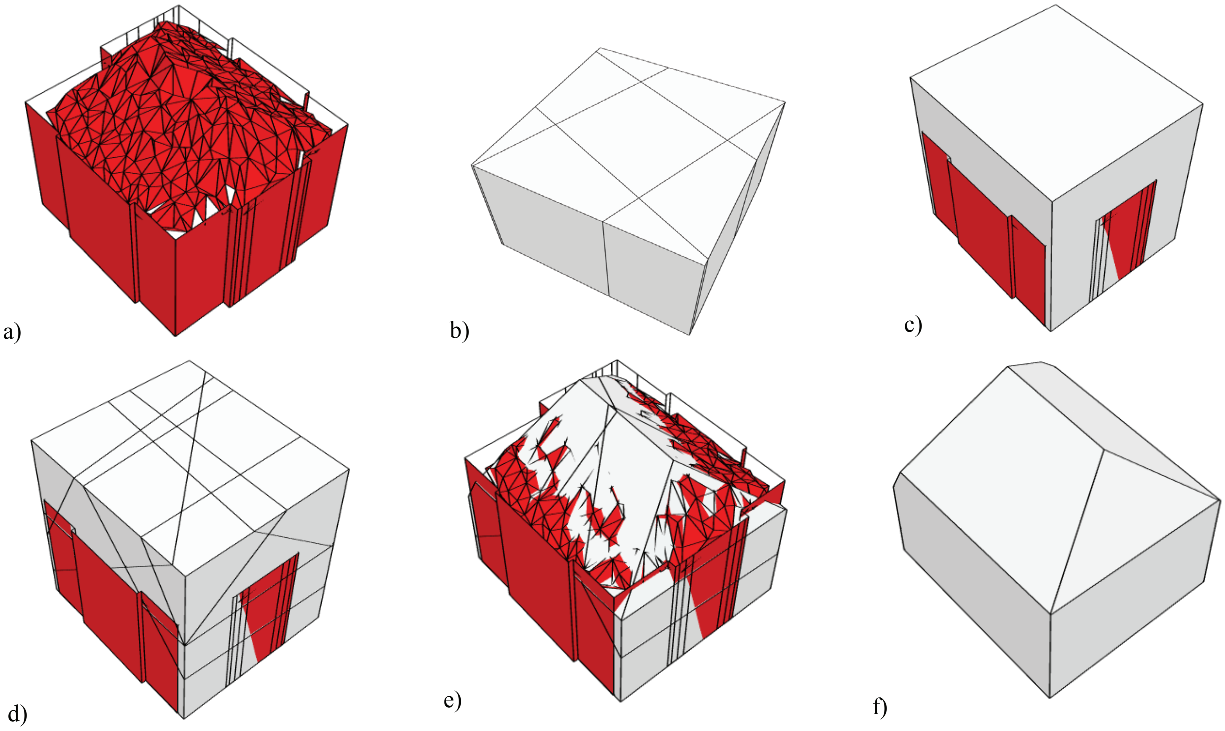


Figure 4: 3D building reconstruction by cell decomposition a) extruded 2D ground plan and meshed LIDAR points, b) subdivision planes and 3D cells from ground plan analysis, c) selected 3D cell and extruded ground plan, d) additional 3D cells for roof approximation from meshed LIDAR points e) selected 3D building cells overlaid to extruded ground plan and meshed LIDAR points, f) final 3D model after gluing.

2.3 Roof reconstruction from airborne LIDAR

The sequence in Figure 4 exemplarily shows the reconstruction of a building including the roof structure. For this process only the triangulated points from laser scanning, which are completely inside the extruded ground plan are used (Figure 4a). Due to the buffer operation discussed in section 2.1, the ground plan analysis results in four perpendicular planes, which subdivide the 3D space (Figure 4b). By intersection of these planes, the 3D cell shown in Figure 4c is generated. In addition to the selected 3D cell, the extruded ground plan is depicted again in red.

In principle, the iterative generation of subdivision planes for roof reconstruction, which is demonstrated in the bottom row of Figure 4, is similar to the process already described in sections 2.1 and 2.2. There, the initial planes were provided from the ground plan segments, while now the meshed triangles of the LIDAR points are used. Similar to a process described in (Gorte 2002), each TIN mesh defines a planar surface at the start of a merging process. Within this process, coplanar surfaces are iteratively merged and the plane equation is updated until there no more similar surfaces can be found. Although the first subdivision process based on the extruded ground plan will generate individual building blocks, the planes for roof reconstruction are determined globally from the triangulated 2.5D point cloud. This ensures that the resulting roof polygons still fit against each other at neighbouring blocks. Subsequent to the decomposition (Figure 4d), the building cells have to be identified. For this purpose, the coverage of potential roof cells by the meshed surface from LIDAR measurement is computed. In Figure 4e, this surface is depicted in red, while the selected 3D cells are shown in grey. As a final step, these cells are glued together to shape the 3D building model (Figure 4f).

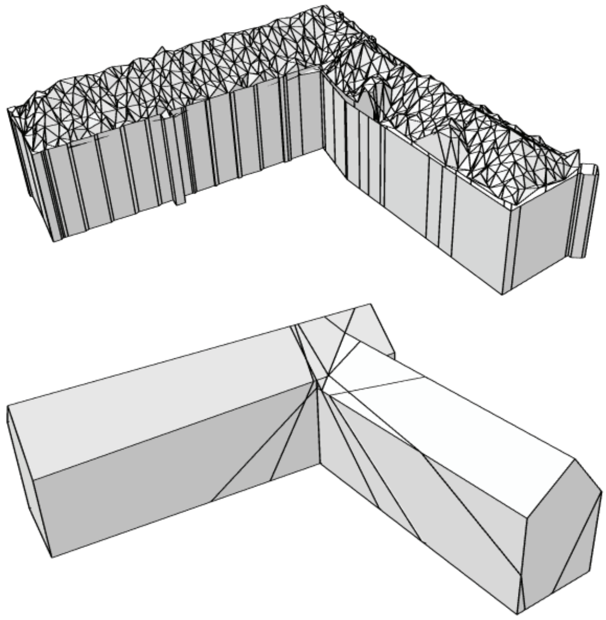


Figure 5: 3D building reconstruction from 2D ground plan and triangulated LIDAR points.

An additional result of the algorithm is given in Figure 5. The top image again shows the extruded ground plan and the meshed LIDAR points while the bottom image contains the selected 3D cells, which were generated from ground plan analysis and mesh merging. These cells are then glued together in a final step. The airborne LIDAR data used in the examples given in Figure 4 and Figure 5 were collected at a mean point distance of 1,5m, which prevents a very detailed reconstruction of the respective roof. However, the general structure of the building is captured successfully.

3. FAÇADE REFINEMENT BY TERRESTRIAL LIDAR

Urban models extracted from airborne data are sufficient for a number of applications. However, some tasks like the generation of realistic visualisations from pedestrian viewpoints require an increased quality and amount of detail for the respective 3D building models. This can be achieved by terrestrial images mapped against the facades of the buildings. However, this substitution of geometric modelling by real world imagery is only feasible to a certain degree. Thus, for a number of applications a geometric refinement of the building facades will be necessary. As an example, protrusions at balconies and ledges, or indentations at windows will disturb the visual impression if oblique views are generated. As it will be demonstrated by the integration of window objects, our approach based on cell decomposition is also well suited for such a geometric refinement of an existing 3D model.

3.1 Data pre-processing

In contrast to an image based detection of windows (Mayer & S. Reznik 2005), we use densely sampled point clouds from terrestrial laserscanning, which contain a considerable amount of geometric detail. Usually, such data are collected from multiple viewpoints to allow a complete coverage of the scene while avoiding occlusions. This requires a co-registration and geocoding of the different scans as a first processing step. Traditionally, control point information from specially designed targets is used for this purpose. Alternatively, an approximate direct georeferencing can be combined with an automatic alignment to existing 3D building models (Böhm & Haala 2005). After this step, the 3D point cloud and the building models are available in a common reference system. Thus, relevant 3D point measurements can be selected for each façade by a simple buffer operation.

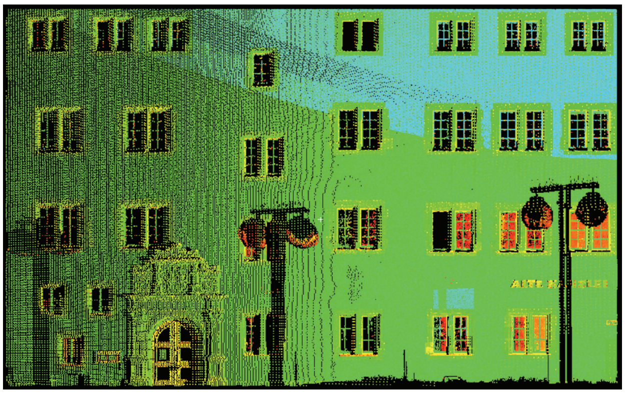


Figure 6: 3D point cloud as used for the geometric refinement of the corresponding building façade.

As an example, for the 3D building in Figure 5, terrestrial LIDAR data was selected by such a buffer operation. For data collection, LIDAR points were measured by a HDS 3000 scanner at an approximate spacing of 4cm. Figure 6 shows this point cloud after transformation to a local coordinate system as defined by the façade plane. Since the LIDAR measurements are more accurate than the available 3D building model, this plane is determined from the 3D points by a robust estimation process. After mapping of the 3D points to this reference plane, further processing can be simplified to a 2.5D problem. Thus algorithms originally developed for filtering of airborne LIDAR (Sithole & Vosselman 2004) can be applied.

3.2 Generation of façade cells

For the refinement of the façade geometry, 3D cells are generated from LIDAR points which have been measured at the window borders. These cells either represent a homogenous part of the façade or an empty space in case of a window. Therefore, they have to be differentiated based on the availability of measured LIDAR points. After this classification step empty cells are eliminated, while the remaining façade cells are glued together to generate the refined 3D building model.

3.2.1 Point cloud segmentation

As it is visible for the façade in Figure 6, usually no 3D points are measured at window areas. Either no measurement is feasible at all due to specular reflections of LIDAR pulses at the glass, or the available points refer to the inner parts of the building and are eliminated due to their large distance to the façade. Thus, in our point cloud segmentation algorithm, window edges are given by no data areas. In principle, such holes can also result from occlusions. However, this is avoided by using point clouds from different viewpoints. In that case, occluding objects only reduce the number of LIDAR points since a number of measurements are still available from the other viewpoints.

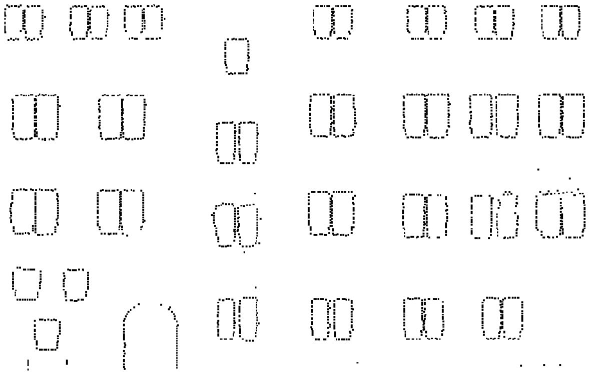


Figure 7: Detected edge points at horizontal and vertical window structures.

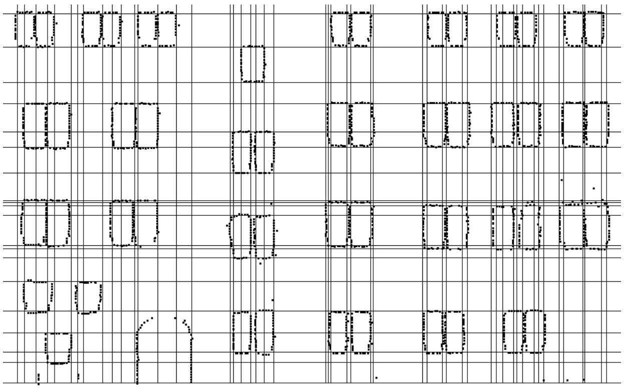


Figure 8: Detected horizontal and vertical window lines.

During segmentation of edge points, four different types of window borders are distinguished: horizontal structures at the top and the bottom of the window, and two vertical structures that define the left and the right side. As an example, to extract edge points at the left border of a window, points with no neighbour measurements to the right have to be detected. In this way, four different types of structures are detected in the LIDAR points of Figure 6. These extracted points are shown in Figure 7. Figure 8 then depicts horizontal and vertical lines, which can be estimated from non-isolated edge points in Figure 7. Each line depicted in Figure 8 can be used to define a plane, which is perpendicular to the building façade. Thus, similar to the ground plan fragmentation in section 2.1, these planes provide the basic structure of the 3D cells to be generated. For this purpose, these planes are intersected with the façade plane and an additional plane behind the façade at window depth. This depth is available from LIDAR measurements at window cross bars. The points are detected by searching a plane parallel to the façade, which is shifted in its normal direction.

3.2.2 Classification of 3D cells

According to the general outline of our algorithm, all the generated 3D cells have to be separated into building and non-building fragments. For this purpose, a binary 'point-availability-map' is generated.

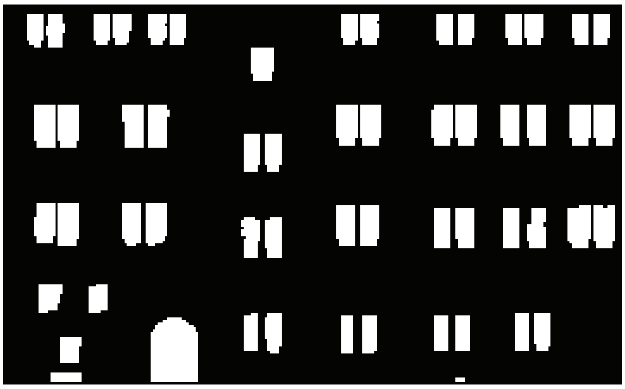


Figure 9: Point-availability-map.

Within this image, which is depicted in Figure 9, black pixels are grid elements where LIDAR points are available. In contrast, white pixels define raster elements with no 3D point measurements. Of course, the already extracted edge points in Figure 7 and the resulting structures in Figure 8 are more accurate than this rasterized image. However, this limited accuracy is acceptable since the binary image is only used for classification of the 3D cells as they are already created from the detected horizontal and vertical window lines. This is realised by computing the ratio of façade to non-façade pixels for each generated 3D cell. This process corresponds to the separation of building and non building cells by ground plan analysis as shown in Figure 3.

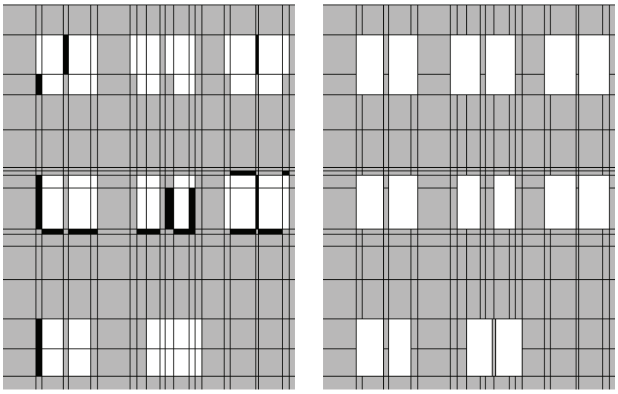


Figure 10: Classification of 3D cells before (left) and after enhancement (right).

As a consequence of the relative coarse rasterization and the limited accuracy of the edge detection, the 3D cells usually do not comprise of facade pixels or window pixels, exclusively. Within the classification, 3D cells including more than 70% façade pixels are defined as façade solids, whereas 3D cells with less than 10% façade pixels are assumed to be window cells. These segments are depicted in Figure 10 as grey and white cells, respectively.

While most of the 3D cells can be classified reliably, the result is uncertain especially at window borders or in areas with little point coverage. Such cells with a relative coverage between 10% and 70% are represented by the black segments in the left image of Figure 10. For the final classification of these cells depicted in the right image of Figure 10, neighbourhood relationships as well as constraints concerning the simplicity of the resulting window objects are used. As an example, elements between two window cells are assumed to belong to the façade, so two small windows are reconstructed instead of one large window. This is justified by the fact that façade points have actually been measured in this area. Additionally, the alignment as well as the size of proximate windows is ensured. For this purpose the classification of uncertain cells is defined depending on their neighbours in horizontal and vertical direction. Within this process it is also guaranteed that the merge of window cells will result in convex window objects.

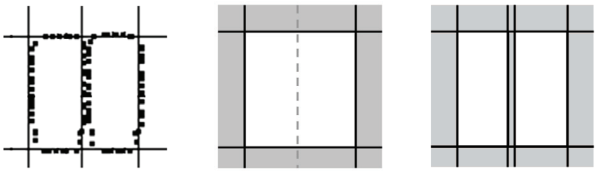


Figure 11: Integration of additional façade cell.

As it is depicted in Figure 11, additional façade cells can be integrated easily if necessary. Figure 11a shows the LIDAR measurement for two closely neighboured windows. Since in this situation only one vertical line was detected, a single window is reconstructed (Figure 11b). To overcome this problem, a window objects is separated into two smaller cells by an additional façade cell. This configuration is kept, if it can be verified as a valid assumption if façade points were actually measured at this position (Figure 11c).

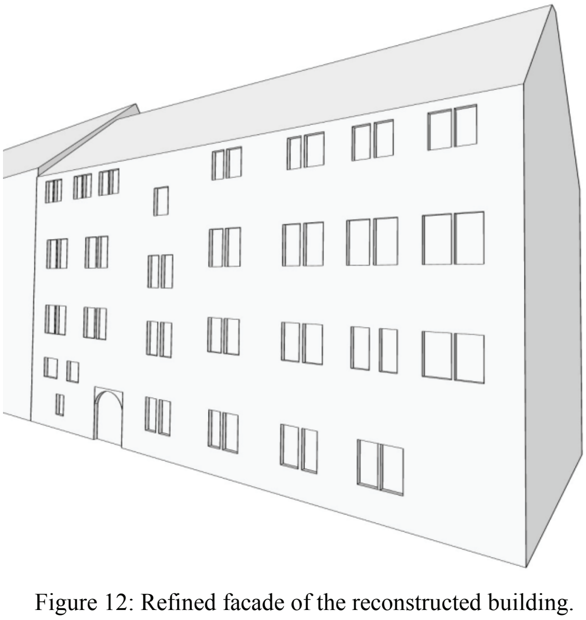


Figure 12: Refined facade of the reconstructed building.

The final result of the building façade reconstruction from terrestrial LIDAR is depicted in Figure 12. For this example window areas were cut out from the coarse model depicted in Figure 5. While the windows are represented by polyhedral cells, also curved primitives can be integrated in the reconstruction process. This is demonstrated exemplarily by the round-headed door of the building.

4. CONCLUSION

Within the paper, an approach for 3D building reconstruction based on cell decomposition was presented. As input data, 2D ground plans and 3D point clouds from airborne and terrestrial LIDAR were used. During the generation of intersecting planes by the combination of ground plan segments, buffer operations are used. By these means the aspired level of generation is defined. Thus, the extruded ground plan can be simplified according to the point density available from airborne LIDAR, which is used for roof reconstruction. Additionally, symmetry relations like coplanarity can be detected during the generation of the planes also for larger distances between different building parts since the extension of these planes is only limited by the subsequent intersection step. The cell decomposition also showed to be very flexible if additional detail has to be integrated. While in our approach windows are represented by indentations, a reconstruction based on cell decomposition can also be used to efficiently subtract such rooms from an existing 3D model if measurements in the interior of the building are available.

Still there is enough room for further algorithmic improvement. However, in our opinion the concept of generating 3D cells by the mutual intersection of planes already proved to be very promising and has a great potential for processes aiming at the reconstruction of building models at different scales.

5. REFERENCES

Brenner, C. [2004]. Modelling 3D Objects Using Weak CSG Primitives. IAPRS Vol. 35.

Böhm, J. & Haala, N. [2005]. Efficient Integration of Aerial and Terrestrial Laser Data for Virtual City Modeling Using LASERMAPS. IAPRS Vol. 36 Part 3/W19 ISPRS Workshop Laser scanning 2005 , pp.192-197.

Gorte, B. [2002]. Segmentation of TIN-Structured Surface Models. Proceedings Joint International Symposium on Geospatial Theory, Processing and Applications, on CDROM, 5p.

Hoffmann, C.M. [1989]. *Geometric & Solid Modelling*. Morgan Kaufmann Punblishers, Inc., San Mateo, CA.

Mäntylä, M. [1988]. *An Introduction to Solid Modeling*. Computer Science Press, Maryland, U.S.A.

Mayer, H. & S. Reznik [2005]. Building Façade Interpretation from Image Sequences. IAPRS Vol. 36-3/W24.

Rottensteiner, F. [2001] Semi-automatic extraction of buildings based on hybrid adjustment using 3D surface models and management of building data in a TIS. PhD. thesis TU Wien .

Sithole, G. & Vosselman, G. [2004]. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* **59**(1-2), pp.85-101.

NEW APPROACH FOR AUTOMATIC DETECTION OF BUILDINGS IN AIRBORNE LASER SCANNER DATA USING FIRST ECHO ONLY

F. Tarsha-Kurdi, T. Landes*, P. Grussenmeyer, E. Smigiel

Photogrammetry and Geomatics Group, MAP-PAGE UMR 694 - INSA de Strasbourg, 67000 Strasbourg, France
(fayez.tarshakurdi|tania.landes|pierre.grussenmeyer|eddie.smigiel@insa-strasbourg.fr)

Commission III, WG III/3

KEY WORDS: LIDAR, Urban, Processing, Detection, Building, Segmentation, Classification

ABSTRACT:

Airborne laser scanning has become a significant 3D data acquisition technique in the field of surveying. By measuring point clouds defined in three-dimensional coordinates, this technique provides almost automatically Digital Surface Models (DSMs). But for 3D city modelling, the discrimination between terrain and elevated objects based on this surface model is still a challenging task, since fully automatic extractions are not operational. Moreover, some of the available methods combine several echoes although echo separation is not always obvious and sometimes last echo is not reliable. In this context, the aim of this study is to develop a general automatic segmentation method of Lidar point clouds focussing exclusively on the first echo and without any external data. The result of the proposed methodology is the automatic discrimination of the buildings and the terrain, by excluding vegetated areas. In the first step, terrain and off-terrain clouds are discriminated, based mainly on threshold features as proposed in the literature, but improved and generalized to the case of brutal terrain discontinuities. In the second step, buildings and vegetation are categorized as subclasses of the off-terrain class. The innovation of the exposed approach lies in the exploitation of the whole analysis levels combining points, pixel, segment and spatial information. Thus, the processing chain fully benefits from the planimetric and altimetric information of a point cloud. The complete workflow is presented, as well as its limitations. At last, the satisfying results obtained for three different test sites covered by two cloud densities validate our processing chain.

1. INTRODUCTION

1.1. Motivation and goals

Most GIS applications need digital terrain models (DTMs) or digital 3D building models as reference layers for subsequent processes. Automatic extraction of man-made objects, particularly 3D building models is a coveted topic (Baltsavias et al., 2001). Currently available DTMs or DEMs covering wide areas come rather from the processing of stereo pairs acquired by optical or radar satellite sensors. Nevertheless, the resolution and accuracy of derived products do not yet match the requirement standards in urban surveying. This is why, before the Lidar technology, photogrammetric techniques applied to aerial photos were the best issue.

The suitability of airborne laser scanning techniques for 3D object reconstruction has been proved over the last decade (Maas, 2005). Nevertheless, although DTM as well as building models are inherent to DSMs, their extraction was never completely or automatically carried out. When using multiple and reliable echoes, results are often very satisfying (Wotruba et al., 2005), since the second echo helps to distinguish points captured on the top of the canopy from those captured on the ground. But in most cases, either the second echo is less accurate than the first one (Yu et al., 2005) or it is not always separable from the first one (Pfeifer et al., 1999; Wotruba et al., 2005).

In this context, the goal of present project is to develop a general segmentation method for the automatic extraction of buildings using the first echo only.

1.2. Related work

The first goal in the processing of laser scanning data is the segmentation of acquired points into terrain and off-terrain classes. In this paper, segmentation means an extraction of point clusters describing a specific class. As summarized in (Maas and Vosselman, 1999), such segmentation may be obtained using additional sources of data, such as 2D GIS information or reflectance information; other processes analyse the local histograms or use filtering techniques considering exclusively Lidar data. Despite the difficulty to categorize complex processes, the latter family could be subdivided into: (a) approaches where the support is mainly an image produced by interpolation and/or segmentation. In this case, segmentation means mainly generation of objects composed of similar pixels; (b) approaches trying to concentrate processing on point level and where segmentation means the discrimination of several clusters in a point cloud. In the category (a), digital image processing techniques are employed, e.g. remote sensing classification methods (Maas, 1999; Tóvári and Vögtle, 2004; Lohmann and Jacobsen, 2004); digital filters related to morphological filtering methods (Lohmann et al., 2000; Vosselman, 2000; Sithole, 2001) or to Fast Fourier Transformations (Marmol and Jachimski, 2004); theory of active shape models (Elmqvist, 2001; Weinacker et al., 2004). In the category (b), the procedures try to stay or return at the Lidar point cloud level, sometimes in an iterative way. One can cite the use of interpolation methods such as the linear prediction method (Kraus and Pfeifer, 1998; Rottensteiner and Briese, 2002), the 3D surface detection (Lee and Schenk, 2002) or the octree structure based segmentation (Whang and Tseng, 2004).

1.3. Position of proposed approach

For the first segmentation of terrain an off-terrain points, the algorithm developed in this paper finds its place in the category (a), because a raster DSM and image processing procedures are used. Reasons explaining this choice are on the one hand the high computing time required when processing the cloud on a point level with adaptative methods, and on the other hand, the availability of well-known digital image processing functions. Thus, based on the analysis of height features on a previously interpolated DSM, successive operations such as thresholding, gradient and morphological filtering on height features are achieved. Results are improved in order to make the proposed procedure reliable even in steep and discontinuous terrains.

The second segmentation chain performs a discrimination of the off-terrain points into two classes: buildings and vegetation. Our solution joins the category (b) in the sense that it uses the relevant information provided by the original Lidar 3D points. Using jointly the DSM and the initial point cloud, the presented algorithm exploits the fact that one cell or pixel in a DSM may contain one or more points of the cloud. Considering building characteristics, this observation leads to the generation of features, which are specific to buildings and based on 3D topological information. Thus, a large part of vegetation can be removed and buildings can be isolated properly. In this way, proposed approach combines three levels of analysis usually used separately: processing based on a pixel level (Maas, 1999; Rottensteiner and Briese, 2002), processing based on a segment level (Lohmann and Jacobsen, 2004; Tóvári and Vögtle, 2004) and that based on a spatial level (Wang and Tseng, 2004).

2. DATA

In order to test our approach on different point densities, two types of data covering three areas are used (Table 1).

Test sites	**“Hermanni”**	**“Victoire boulevard”/ ”Strasbourg centre”**
Acquisition	End of June 2002	Begin September 2004
Sensor	TopoEye	TopScan (Optech ALTM 1225)
Points density	7-9 points / m²	1.3 points / m²
Flight height	200 m	1440 m
Pulse frequency	7 kHz	25 kHz
Field of view	± 20 degrees	±26 degrees
Points/dataset	410 497	450 000 and 400 000

Table 1. Characteristics of the three datasets used in this study

The first test site “Hermanni” is a residential area in periphery of Helsinki, where large and spaced storey houses are surrounded by vegetation. This point cloud belongs to the building extraction project of EuroSDR (www.eurosdr.org). The second test site called “Victoire boulevard” is located in the campus district of Strasbourg, along a road where trees are near to large buildings. Finally, the third cloud “Strasbourg centre” covers the centre of Strasbourg city, known for its tangled up houses. TopScan has acquired the last two clouds during the same campaign. Unfortunately, only the first echo is reliable.

3. WORK FLOW FOR OFF-TERRAIN DETECTION

The first segmentation step consists in separating the off-terrain points (building, vegetation, trees) from the ground. The workflow is presented in the following paragraphs.

3.1 Interpolation of a DSM

A Lidar point cloud is represented by 3D points, not always regularly spaced. The use of digital filters requires transforming this dataset into a uniform 2D grid. In order to preserve the real measured altitudes (the fitting surface should follow the Lidar points), a nearest neighbour interpolation technique is used. The well-known advantages of this technique are the low interpolation calculation time and the conservation of the original altitude values. This means, that the topological original relationships between points -in the sense of relative height variations between neighbouring points- can also be preserved by this interpolation. Of course, a determining criterion is the definition of the sampling value (resolution) of the DSM. Under the assumption that the distribution of points is regular and that one pixel must contain at least one point, the average cloud density can be calculated. Thus the sampling interval *SI* can be deduced (1).

$$SI = \frac{1}{\sqrt{density}} \qquad (1)$$

Obviously, the detectable object is directly dependent on the available density, whatever the method of interpolation.

3.2 Detection of the off-terrain segment edges

To detect the off-terrain segment edges, as suggested by (Maas, 1999), directional gradient filters are applied on the DSM with 3x3 kernels under eight different rotations (*k.π/4; k=1…8*). The first matrix contains value *1* in the upper left cell and value *-1* in the lower right cell. Thus, eight bands are generated in which the grey values represent height differences. Then, the maximum gradient for each pixel is searched over the k bands and assigned to a matrix ΔZ (equation 2).

$$\Delta Z_{i,j} = \max\,(G_{i,j})_k \qquad (2)$$

where i, j : pixel position in line and column
$(G_{i,j})_k$: kth band of filtered image
k : gradient band number (k=1,…,8)

Comparing the pixel values of the ΔZ matrix to a defined threshold S_1, the detection of the edge pixels is possible. The maximum absolute gradient shows behaviour similar to that of the slope around each pixel. Thus if $\Delta Z_{i,j} > S_1$ the pixel describes an off-terrain edge (buildings or vegetated area) and takes the value 1 in a binary matrix A. The threshold S_1 is defined according to the smallest detectable building. Generally, in the countries we are concerned with, the smallest foreseeable height for a building is 5 to 6 meters. Figure 2a presents the binary matrix A obtained for the Hermanni test site.

At this stage, two operations occur: the first one consists in filling the body of the off-terrain segment borders already detected. The second one consists, in parallel, in assigning a neighbouring ground value to the off-terrain pixels that facilitates the generation of a DTM.

3.3. Detection of the whole off-terrain pixels

In order to fill the body of the segment borders created previously, the neighbourhood of each pixel has to be considered. For this purpose, the matrix A and the DSM are analyzed. Firstly, if the central pixel in a 3x3 moving window (moving over A) belongs to an edge, the lowest neighbouring altitude is assigned to it (Fig. 3 [1]). Then, the height difference between the pixel and its neighbours is calculated in matrix $\Delta Z_{(i,j),(k,l)}$ as expressed in equation 3.

$$\Delta Z_{(i,j),(k,l)} = (Z_{k,l} - Z_{i,j\_ground}) \qquad (3)$$

were $\quad$ i, j : central pixel coordinates
$\quad$ k, l : coordinates of the 8 neighbours; $(k,l)\neq(i, j)$

The moving window takes into account the results obtained by the last position before continuing its progression. This leads to the distinction between the neighbouring pixels in which the altitude has been changed by this step, i.e. projected on the ground $Z_{k,l\ ground}$ and the neighbouring pixels in which the altitude remains the same, i.e. $Z_{k,l\ orig}$ (Fig. 3 [2]).

Pixels describing off-terrain objects will present high values for equation 3. Thus, if $\Delta Z_{(i,j),(k,l)} > S_1$ and simultaneously the neighbouring pixel (k,l) does not describe an edge in matrix A, the pixel (i,j) is assigned to the off-terrain class (Fig. 3 [3]) and its altitude becomes $Z_{i,j\ ground}$. The threshold S_1 is the same as previously, since maximum height differences are also compared here. Figure 2b presents the resulting binary image.

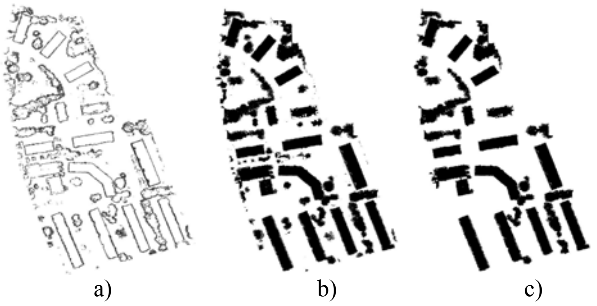


Figure 2. a) Off-terrain segment edges (matrix A). b) Off-terrain class. c) Off-terrain after morphological filtering

3.4. Assigning ground altitudes to the off-terrain pixels in a later building extraction purpose

As already mentioned, while off-terrain pixels are assigned to their class, their altitude is replaced, in parallel, by a ground level altitude $Z_{i,j\ ground}$. This modification is directly dependent on the surrounding altitude values and on their history ($Z_{k,l\ ground}$ or $Z_{k,l\ orig}$). So, each off-terrain pixel gets a ground level altitude. Thus, three matrices are output from the workflow summarized in Figure 3: matrix A, the normalized DSM (nDSM) and a matrix called Test_ground. Matrix A is a binary mask where the off-terrain pixels are non-zero. The nDSM contains the whole pixels belonging to the off-terrain class. The matrix Test_ground contains three values: 1, 2 and 0. A pixel with a value of 1 means that its altitude in nDSM has been taken from the original dataset; a pixel with value 2 means that its altitude comes from previously modified altitudes, whereas pixels with value 0 belong to the “ground” class (Figure 4). The value 2 occurs generally for pixels located inside the body of the buildings. This is due to the fact, that inside a vegetated area, the laser beam may reach the ground. Whereas this situation is rare in a building segment, except in the case of interior courts. This characteristic (although insufficient) is of crucial importance for the next segmentation step, i.e. for the distinction between buildings and vegetation.

3.5. Classification generalization

As suggested by (Vosselman and Maas, 2004), mathematical morphology operations may help to clean the classification from remaining segment residuals**.** Two successive operations are applied here. Firstly, a morphological opening allows erasing the punctual segments remaining on the ground. Then, a morphological closing enables filling last gaps occurring in the off-terrain segments (Figure 2c).

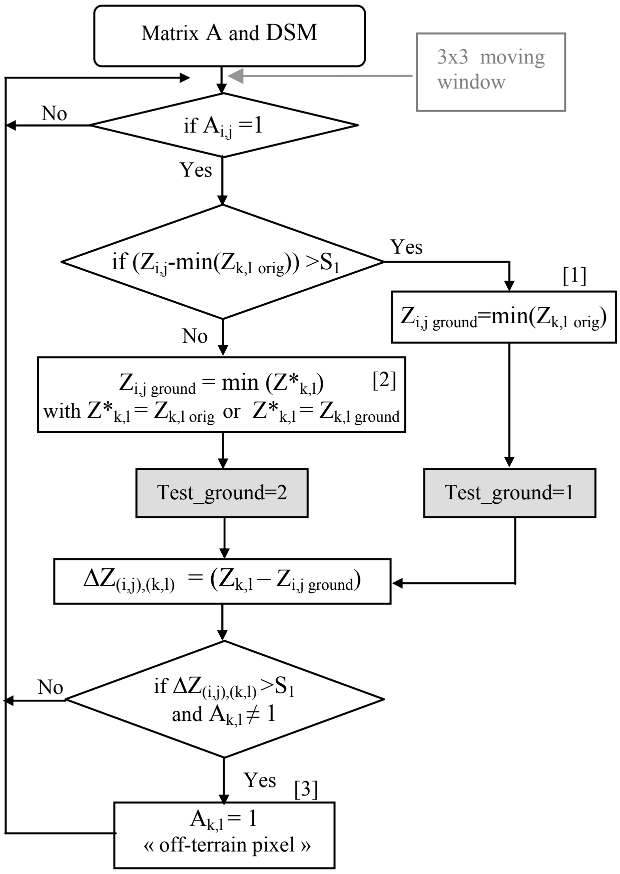
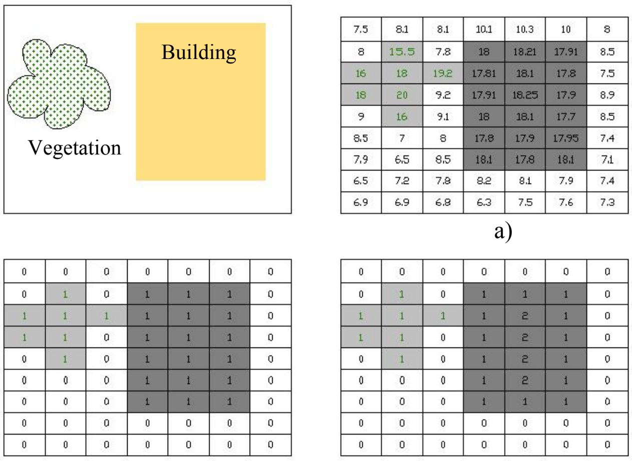

Matrix A and DSM
3x3 moving window
No
if $A_{i,j}$ =1
Yes
if $(Z_{i,j}\text{-}\min(Z_{k,l\ orig})) > S_1$
Yes
[1]
$Z_{i,j\ ground}=\min(Z_{k,l\ orig})$
No
$Z_{i,j\ ground} = \min(Z^*_{k,l})$
[2]
with $Z^*_{k,l} = Z_{k,l\ orig}$ or $Z^*_{k,l} = Z_{k,l\ ground}$
Test_ground=2
Test_ground=1
$\Delta Z_{(i,j),(k,l)} = (Z_{k,l} - Z_{i,j\ ground})$
No
if $\Delta Z_{(i,j),(k,l)} > S_1$
and $A_{k,l} \neq 1$
Yes
[3]
$A_{k,l} = 1$
« off-terrain pixel »

Figure 3. Workflow for the detection of off-terrain pixels


Building
Vegetation

7.5	8.1	8.1	10.1	10.3	10	8
8	15.5	7.8	18	18.21	17.91	8.5
16	18	19.2	17.81	18.1	17.8	7.5
18	20	9.2	17.91	18.25	17.9	8.9
9	16	9.1	18	18.1	17.7	8.5
8.5	7	8	17.8	17.9	17.95	7.4
7.9	6.5	8.5	18.1	17.8	18.1	7.1
6.5	7.2	7.8	8.2	8.1	7.9	7.4
6.9	6.9	6.8	6.3	7.5	7.6	7.3

a)

0	0	0	0	0	0	0
0	1	0	1	1	1	0
1	1	1	1	1	1	0
1	1	0	1	1	1	0
0	1	0	1	1	1	0
0	0	0	1	1	1	0
0	0	0	1	1	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

b)

0	0	0	0	0	0	0
0	1	0	1	1	1	0
1	1	1	1	2	1	0
1	1	0	1	2	1	0
0	1	0	1	2	1	0
0	0	0	1	2	1	0
0	0	0	1	1	1	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

c)

Figure 4. Off-terrain class detection. a) DSM. b) Matrix A. c) Test_ground matrix.

3.6. Improvement of the processing chain to the case of terrain discontinuities

By analysing the results, it becomes clear that the present segmentation based on a sequence of thresholds and morphological filtering needs improvement in the case of large local terrain discontinuities caused by holes, ditches, noise, etc. Indeed, the cutting surface passes over the terrain surface with a height threshold S_1. Therefore, in the case of brutal discontinuities, the algorithm misclassifies the pixels behind the ditch as “off-terrain” pixels (Figure 5).

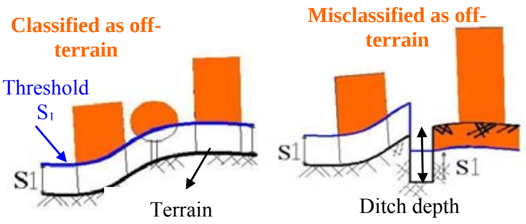
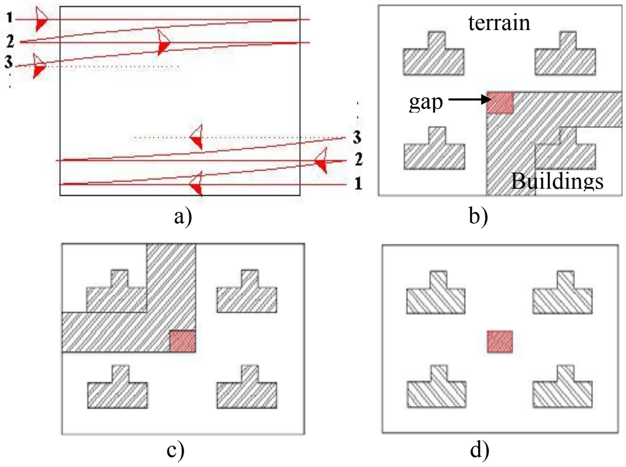


Figure 5. Processing error in the case of terrain discontinuities.

In the case of discontinuities, (Sithole and Vosselman, 2003) showed deficits of all main filtering methods even when the algorithm has some special rules to avoid misclassification next to breaklines. To cope with this problem, we analyzed the reaction of the algorithm on different test sites and in several discontinuities conditions. It becomes clear that the error generated and illustrated in Figure 5 is directly dependent on the direction passage of the moving window. The solution consists simply in achieving the filtering along different directions over the image. The intersection of the intermediate products therefore solves the problem.

Figure 6 gives an example. In step (a), the moving window detecting the off-terrain pixels moves over the DSM following the usual and then the opposite direction. It provides detection of the shaded segments in (b) and in (c) respectively, where the lower right part of (b) and the upper left area of (c) are misclassified. The intersection of (b) and (c) cancels the misclassified areas in the step (d). The detected ditch is automatically assigned to the ground class.



a) b)

c) d)

Figure 6. Misclassification correction in the case of terrain discontinuities. a) Moving window movements. b), c) Respective results. d) Ditch cancelling.

By this mean, two classes of points are efficiently generated, i.e. the terrain class (ground) and the off-terrain class (buildings, vegetation, noise, etc.), even in the case of abrupt terrain. The next part will analyse the off-terrain class in order to extract the buildings from it.

4. DETECTION OF BUILDINGS

Various algorithms have been suggested and applied to laser scanning data with the aim of separating buildings from other elevated objects. As mentioned in the introduction, these procedures are mainly based on previously interpolated grids, while often neglecting invaluable information contained in the initial irregular cloud of acquired points.

4.1. Contribution of 3D points contained in one cell

The approach proposed in this paper exploits the fact that one pixel in the nDSM may contain one or more points of the cloud. The 3D position of these cell-points will play an important role in the discrimination procedure of the subclasses vegetation and buildings. Thus, the nDSM and the initial point cloud are used jointly in a pixel level, segment level and spatial analysis level.

Considering one cell, three types of altitude values Z may occur as illustrated in Figure 7:

1. Z_build : the real topographic altitudes (terrain or off-terrain points);
2. Z_DSM: the raw DSM values obtained by interpolation of the cloud points;
3. Z_points: the point altitudes as acquired by the sensor. Other interesting features are the extreme values of Z among the points of a cell (min_Z_points or max_Z_points), etc.

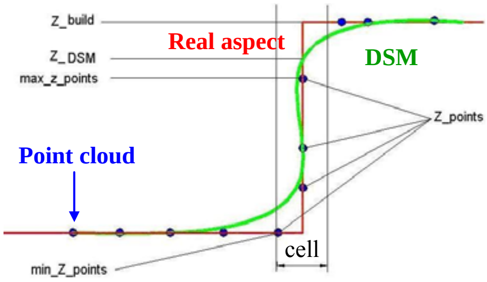


Figure 7. Different types of altitude values occurring in one cell.

4.2. Segmentation methodology developed

The approach for detecting buildings among trees or other objects is based on the assumption that the building roofs are normally composed of flat planes or more generally of surfaces. So, the methodology developed here is based on the search for planes composing the roofs of the buildings, as suggested by (Elaksher and Bethel, 2002). The main advantages lie in the fact that it is possible to adapt the concept to relatively low point densities. Moreover, only the first echo is used and, at last, the segments representing a mixture of buildings and vegetation can be removed.

For the moment, the off-terrain class contains mainly trees and buildings. If we succeed in adjusting a group of points by a plane using the least square method (regarding small residues) a large quantity of points representing vegetation can be eliminated. The implementation of this principle emphasizes limits and constraints. The mathematical detection of the points composing a plane requires a great deal of processing time. Moreover, because of low point density and/or details on the roofs (e.g. chimneys), the distribution of the points on a roof cannot always be adjusted by a plane. Finally, it occurs that vegetation points can be adjusted by an average plane while presenting negligible residues!

To avoid these disadvantages, our approach replaces the mathematical test by topological relationships. As mentioned above, it will benefit from a paramount element in the interpretation of the cloud: the points inside a cell of the DSM. Because of the irregular distribution of Lidar points and the existence of vertical elements, the number of points per pixel in X,Y as well as in Z dimension is variable. So the three topological features we can derive for every cell are:

- The number of points per cell
- The maximum vertical distance between points
- The maximum slope angle of the best fitting plane

These three criterions allow the elaboration of classification rules with the specific aim of building detection. No assumption can easily be made based on the first criterion (number of pixels per cell), since this criterion is very dependant on the point cloud density and regularity. If a cell covers entirely a portion of a roof, then one can put forth the following assumptions:

- Assumption relating to second criterion: the maximum height difference $\Delta h_{\max}$ between points in a cell will be lower than a certain threshold. The value of the threshold will depend on 3 factors: the sampling of the DSM grid, the roofs geometry (horizontal plane, tilted plan or spherical surface) and the altimetric measurement accuracy.

So $\Delta h_{\max} = h_2 - h_1$ and if we consider $\sigma_{h_2} = \sigma_{h_1} = \sigma_h$, the transmission of errors delivers: $\sigma_{\Delta h\max} = \sqrt{2}.\sigma_h = \pm 21cm$, since Lidar data relative accuracy in Z is about ± 15cm.

- Assumption relating to third criterion: the maximum slope of a plan adjusting locally a roof is equal to 60 degrees. That allows expressing $\Delta h_{\max}$ as a function of the sampling interval *p* and maximum slope angle, so: $\tan(60°) = \Delta h / p$.

Thus, from the last two assumptions we find the threshold of maximum height difference (equation (4)). Under this threshold, the corresponding pixel is assigned to the building class.

$$\Delta h_{\max} \leq \tan(60°).p + \sqrt{2}.\sigma_h \qquad (4)$$

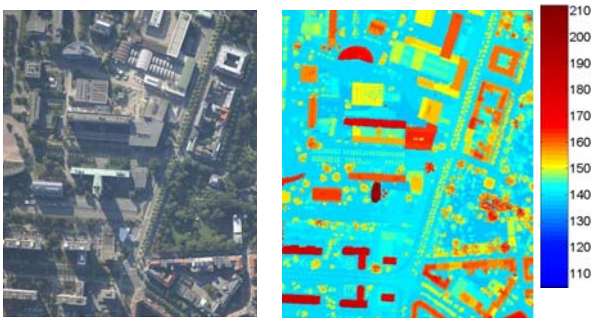
In addition to the last test, the classification rule also takes into account the values 2 representing the body of building segments in the matrix Test_ground (Figures 3 and 4).

The detected segments represent at first the kernel of building roof planes. In order to complete this kernel with the surrounding pixels, a specific region-growing algorithm has been developed, working on the 8 neighbouring height differences. A last filter erases the remaining segments by regarding the smallest foreseeable building segment. Results obtained through this workflow are very satisfying, since the major part of buildings is well classified. Only a few pixels of vegetation are misclassified and easily rejected by mathematical morphology. Indeed, it happens that a group of points, within the vegetation class, accepts an average plane with negligible residues and respects the whole topological assumptions.

5. RESULTS AND DISCUSSION

We applied the proposed algorithm on the three datasets available. The computing time required for extracting buildings is negligible. Figure 9 presents the buildings extracted from the DSM (Figure 8) of the "Victoire boulevard" test site.

In order to evaluate the precision of the building classification, an estimation method suggested by (Sithole and Vosselman, 2003) and based on a confusion matrix has been applied. The classes of interest are "buildings" and "not-buildings". The reference images have been conceived by digitizing the buildings in the DSM with the help of aerial images. Three errors characterize the precision of the obtained segmentations and are reported in Table 10. Error I shows the proportion of building-pixels misclassified; Error II the proportion of not-building pixels misclassified and Total Error the proportion of misclassified pixels. The last column presents the mention of the influence of neighbouring points filtering. The different mentions obtained are explained by the different urban typology of the city centre compared to that of areas located in periphery. However, the proposed algorithm provides very good results, for both point densities.



a) b)

Figure 8. " Victoire boulevard" site. a) Aerial image. b) Pseudocolor coded DSM.

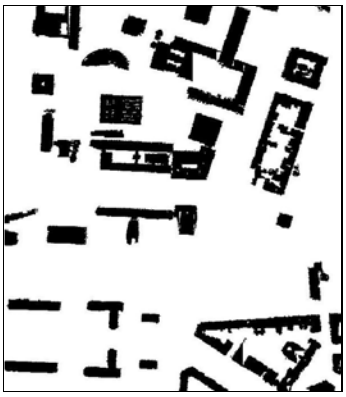


Figure 9. Building detection for "Victoire boulevard" site.

Test site	Error I (%)	Error II (%)	Total Error (TE) in %	Mention
Hermanni	0.22	0.01	0.01	Excellent (TE<1%)
Victoire boulevard	1.57	0.34	0.55	Excellent (TE<1%)
Strasbourg centre	1.46	0.97	1.12	Very good (1<TE<5%)

Table 10: Precision of building/not-building segmentation

At last, more independent quantitative evaluation consists in counting the detected buildings among the existing buildings. Table 11 proves that the detection rate is very satisfying and validates definitely our method.

Test site	Number of detected buildings	Number of non detected buildings	Total number of buildings	Detection rate (%)
Hermanni	15	0	15	100
Victoire boulevard	60	4	64	94
Strasbourg center	64	3	67	95

Table 11: Rate of correctly detected buildings.

Nevertheless, this approach shows limitations in three main aspects. On the one hand, the classification precision is function of the point density and decreases with it. On the other hand, when trees and buildings are simultaneously close to each other and of the same height, the distinction becomes difficult. Furthermore, if a bloc of buildings is composed of several buildings with similar heights (falling under the threshold S_1) the algorithm will misclassify the “off-terrain” pixels as “terrain”. So, the use of shape or geometric criterions has to be considered. At last, a methodology leading to deduce the most appropriate threshold according to the urban typology has to be developed.

6. CONCLUSION

This paper presents a new approach for detecting buildings in a Lidar point cloud, using exclusively the first echo. The most relevant idea is to benefit from the original point locations at strategic moments of the segmentation. While eliminating automatically the misclassifications caused by terrain discontinuities, the developed algorithm takes advantage of the topology of points belonging to one cell and produces a building segmentation image with high accuracy. Thus, the Lidar data are considered on a point level, pixel level, segment level and global level during the processing chain. Nevertheless, this methodology needs to be improved since in particular cases, some small vegetation segments may remain at the end of the process. Further investigations should also allow to predefine the optimal threshold referring to the urban typology. At this stage, the reconstruction of the building geometry in the forthcoming modelling phase can be considered.

REFERENCES

Baltsavias, E., Gruen, A., Van Gool, L., 2001. (Editors): *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*, A.A. Balkema Publishers, ISBN 9058092526, 415p.

Elaksher, A. F., and Bethel., J. S., 2002. Reconstructing 3D buildings from Lidar data. *Int. Archives of Photogrammetry and Remote Sensing*, Vol. XXXIV, part 3A/B, ISSN 1682-1750, pp102-107.

Elmqvist, M., 2001. Ground Estimation of Laser Radar Data using active shape Models. *OEEPE workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Elevation Models*, Stockholm, Sweden.

Kraus, K., and Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing,* 53, pp. 193–203.

Lee., I, Schenk., T, 2002. Perceptual organization of 3d surface points. *Photogrammetric computer vision. ISPRS Commission III*, Graz, Austria. Vol. XXXIV, part 3A/B, ISSN 1682-1750.

Lohmann, P., Koch, A., Schaeffer, M., 2000. Approaches to the filtering of laser scanner data. Vol. 33, *Int. Archives of Photogrammetry and Remote Sensing*, Amsterdam, pp. 540–547

Lohmann, P., and Jacobsen, K., 2004. Filterung segmentierter Oberflächenmodelle aus Laserscannerdaten. In: *PFG* (2004), Nr. 4, S. 279-287.

Maas, H.-G., 1999. The potential of height texture measures for the segmentation of airborne laserscanner data. *Proceedings of the 4th International Airborne Remote Sensing Conference*, Ottawa, Vol. I, pp. 154-161.

Maas, H.-G., 2005. Akquisition von 3D-GIS Daten durch Flugzeuglaserscanning. *Kartographische Nachrichten*, Vol. 55, Heft 1, S. 3-11.

Maas, H.-G., Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. ISPRS *Journal of Photogrammetry & Remote Sensing* Vol. 54, No. 2/3,

Marmol, U., Jachimski, J., 2004. A FFT based method of filtering airborne laser scanner data. *Int. Archives of Photogrammetry and Remote Sensing*, ISSN 1682-1750, Vol. XXXV, part B3.

Pfeifer, N., Reiter, T., Briese, C., Rieger, W., 1999. Interpolation of high quality ground models from laser scanner data in forested areas. *Joint Workshop of the ISPRS working groups III/5 and III/2* , La Jolla, CA, USA, Nov. 9 – 11 1999.

Rottensteiner, F., Briese, Ch., 2002. A new method for bulding extraction urban areas from high-resolution LIDAR data. *Int. Archives of Photogrammetry and Remote Sensing*, Vol XXXIV / 3A (2002), ISSN 1682-1750; 295 - 301.

Sithole, G., Vosselmann, G., 2003. Automatic Structure Detection in a Point-Cloud of an Urban Landscape, *2nd Joint Workshop on Remote Sensing and Data Fusion over Urban Areas* (URBAN2003), May 22-23, Berlin, Germany.

Sithole, G., 2001 . Filtering of laser altimetry data using a slope adaptative filter. *Int. Archives of Photogrammetry and Remote Sensing*, Annapolts, Vol. XXXIV – 3/W4

Tóvári, D., Vögtle, T., 2004. Classification methods for 3D objects in laserscanning data. *Int. Archives of Photogrammetry and Remote Sensing*, ISSN 1682-1750, Vol. XXXV, part B3.

Vosselman, G., 2000. Slope based filtering of laser altimetry data. Vol. XXXIII, *Int. Archives of Photogrammetry and Remote Sensing*, Amsterdam, pp. 935–942, Part B3.

Vosselman, G. and Maas, H., 2004. Airborne Laser Altimetry: DEM production and Automatic Feature Extraction. *Tutorial TU6 held at the XXth ISPRS Congress* in Istanbul, July 14, 2004.

Wang, M., Tseng, Y.-H., 2004. Lidar data segmentation and classification based on octree structure. *Int. Archives of Photogrammetry and Remote Sensing*, ISSN 1682-1750, Vol. XXXV, part B3.

Weinacker, H., Koch, B., Heyder, U., Weinacker, R., 2004. Development of filtering, segmentation and modelling modules for LIDAR and multispectral data as a fundament of an automatic forest inventory system. *Int. Archives of Photogrammetry and Remote Sensing*. Freiburg, Germany, Volume XXXVI, Part 8/W2. ISSN 1682-1750.

Wotruba, L., Morsdorf, F., Meier, E., Nüesch, N., 2005. Assessment of sensor characteristics of an airborne laser scanning using geometric reference targets. *Proceedings of the ISPRS Workshop Laser scanning 2005*. Enschede, The Netherlands, ISSN 1682-1777.

Yu., X, Hyyppä., H, Kaartinen., H, Hyyppä., J, Ahokas., E, Kaasalainen., S. 2005. Applicability of first pulse derived digital terrain models for boreal forest studies factors affecting the quality of DTM generation in forested areas. *Proceedings of the ISPRS Workshop Laser scanning 2005*. Enschede, The Netherlands, ISSN 1682-1777.

FACADE RECONSTRUCTION FROM AERIAL IMAGES BY MULTI-VIEW PLANE SWEEPING

Lukas Zebedin, Andreas Klaus, Barbara Gruber and Konrad Karner

VRVis Research Center
Inffeldgasse 16/2, Graz, AUSTRIA
{zebedin, klaus, gruber, karner}@vrvis.at

KEY WORDS: Building Reconstruction, Aerial Images, Plane Sweeping, Information Fusion, Multi-View Matching

ABSTRACT:

This papers describes an algorithm to estimate the precise position of facade planes in digital surface models (DSM) reconstructed from aerial images using an image-based optimization method which exploits the redundancy of the data set (along and across track overlap). This approach assumes that a facade is a vertical plane and that the heightfield is precise enough to generate hypotheses for the initialization of the optimization algorithm. The initialization is first roughly oriented using the principal line directions of its texture, afterwards a hierarchical algorithm performs a finer optimization to maximize the correlation across different views. The proposed method is applied to real world imagery and its results are shown.

1 INTRODUCTION AND MOTIVATION

Reconstruction of buildings in urban areas from aerial images is a challenging task. Many applications like virtual tourism, urban planning and cultural documentation benefit from a realistic, high-quality city model. There already exist methods to create a dense point cloud of urban scenes using LIDAR scans or dense image matching ((Berthod et al., 1995), (Cord et al., 1998)) which can be used to create a polygonal roof model ((Samadzadegan et al., 2005)), (Vosselman and Dijkman, 2001)), however the estimation of facades poses a separate problem because of the oblique angle at which they are viewed during aerial data acquistion. The optimization employed by the proposed algorithm is image-based.

One critical aspect of building reconstruction is the estimation of the contours of buildings. Many workflows on urban scene reconstruction rely on additional information like a ground-plan ((Brenner, 2000) and (Haala et al., 1998) for example) to delineate the contours of buildings. However, this information is not always available or has to be manually created which is a major drawback if a fully automatic workflow is desirable.

The other possibility is to infer the outlines of buildings by segmenting the DSM into building blocks. This has been done by (Weidner, 1996) and (Vosselman, 1999). The drawback of this approach is obviously the flawed, jaggy nature of the obtained contours. (H. Gross, 2005) tried to alleviate this by fitting rectangles to the outline. Such improvements however can only guess the position of the facades. If the resulting model is afterwards textured, any error in the placement results in skewed and misaligned textures.

This drawback of automatic deduction of outlines can be alleviated by optimizing the position of the outlines as proposed in this paper.

(Coorg and Teller, 1999) presented a similar algorithm which operated on close-range imagery. They, however, relied strongly on horizontal lines in building facades to even initialize their estimates.

The basic idea of plane sweeping was also used in (T. Werner, 2002), but there only a translational plane sweep is considered in terrestial imagery. Also the initialization of the plane sweep is quite different from our approach where vanishing points are being exploited.

(C. Vestri, 2000) discusses a very similar algorithm to the one proposed in this paper, but is based on pointwise reconstruction of a facade. The main difference however is that they use vertical planes which are rotated in 20 degree intervals around the vertical axis to obtain the facade points whereas our algorithm optimizes the rotational and translational component of each facade independently therefore increasing the estimation accuracy. Additionally the pointwise reconstruction performed by them does not exploit the knowledge that the facade is a plane.

This contribution is based on images from the UltraCamD camera from Vexcel Corporation with its multispectral capability. The UltraCamD camera features a multi-head design. It delivers large format panchromatic images composed from nine CCD sensors (11500 pixels across-track and 7500 pixels along-track) and simultaneously recorded four additional channels (red, green, blue and NIR) at a frame size of 3680 by 2400 pixels. The image data used comprise the panchromatic high resolution images as well as the low resolution multispectral images.

The data set used in this paper to compute the depicted results was acquired in Summer 2005 over the inner city of Graz, Austria. It consists of 155 images flown in 5 strips. The along-track-overlap of this data set is 80%, the across-track overlap is approximately 60%. The ground sampling distance is around 8cm.

2 FACADE OPTIMIZATION

The algorithm for obtaining optimized facades can be decomposed into three distinct steps: first some hypotheses have to be found. Those estimated facades are then refined in such a way, that they are parallel to the true facade. In the last step the fine-grained optimization using multi-view correlation is performed.

2.1 Input Data

The optimization algorithm is image-based, therefore a precise orientation of the imagery is of utmost importance. The average back projection error is of utmost importance to enable convergence of the optimization. Theoretically two views of a plane are enough to calculate the correlation score, however in case of occlusions and in order to increase stability more views can be used. Therefore the data acquisition is also critical to the success of the optimization because only views are usable where the facade lies near the border of the image. The reason for this is the fact that aerial images have a very limited visibility of vertical planes as in the center of each image the perspective projection is comparable to a orthographic projection which hides all vertical planes . This assumption requires that flight altitude, velocity, focal length and along/across-track overlap are carefully chosen to provide also data redundancy for facades.

Another prerequisite is the DSM which is used to initialize the hypothesis for facades. For the experiments conducted for this paper, a plane sweeping approach was chosen which is improved and densified by applying an iterative and hierarchical multi-view matching algorithm based on homographies. A more detailed description of this algorithm implemented on graphics hardware can be found in (Zach et al., 2003).

The building block layer is based on a land use classification and describes the position of buildings within the scene. The land use classification used for this data set is a supervised classification that includes a training phase and that runs automatically afterwards. The classification results comprise classes like buildings, streets or other solid objects with low height, water, grass, tree or wood, as well as soil or bare earth. The classification is based on support vector machines and is described in detail in (Gruber-Geymayer et al., 2005).

2.2 Initialization

The initial estimates of the position of facades is obtained by applying a Canny edge detector to the heightfield. Those edgels are afterwards chained together to form lines. One important parameter of this line extraction is the minimum length of each line, as longer lines tend to be more stable in the optimization performed in a later phase.

The line extraction is aided by the land use classification which assigns a label to each pixel in the heightfield. These labels are used to restrict line extraction to regions near buildings.

The result of this procedure is illustrated in Figure 1. Note that only lines near the building are extracted whereas there are no lines near the tree in the inner courtyard of the building.

These lines in 2D are then extended to 3D planes by estimating the minimum and maximum height from the surrounding area in the heightfield. A small margin is subtracted from the top and bottom of the plane to account for possible occlusions near the roof (protrusion of the eave line) and the ground.

2.3 Line Direction Optimization

The first optimization applied to the facade planes tries to align the orientation of real facades and their hypothesis. As a result the plane will be almost parallel to the real facade. The algorithm relies on the fact that facades mainly contain structures which are horizontally or vertically aligned with the facade itself (windows, balconies, signs and alike).

For each facade plane the algorithm first makes a ranking of all available cameras and assigns each one a score. This score is calculated with the following equation:

$$score = normal \cdot (origin - anchor)$$

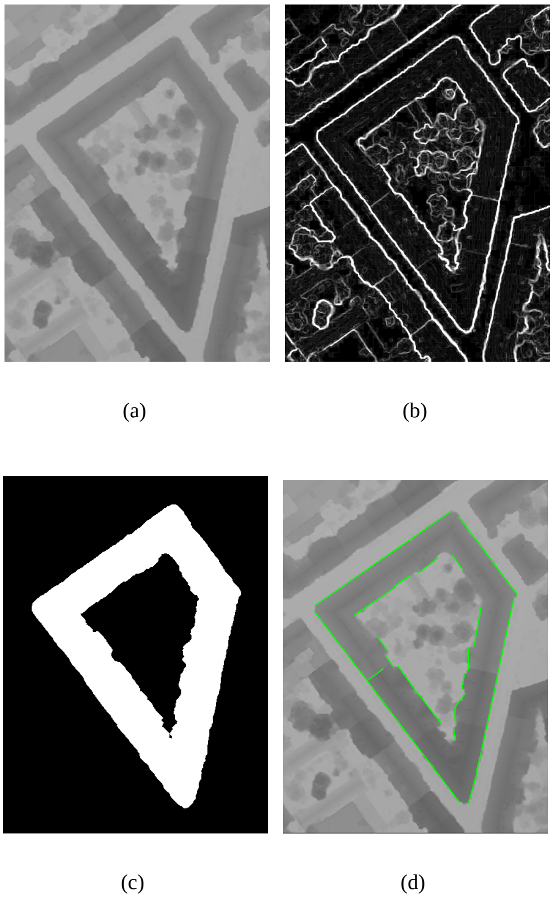
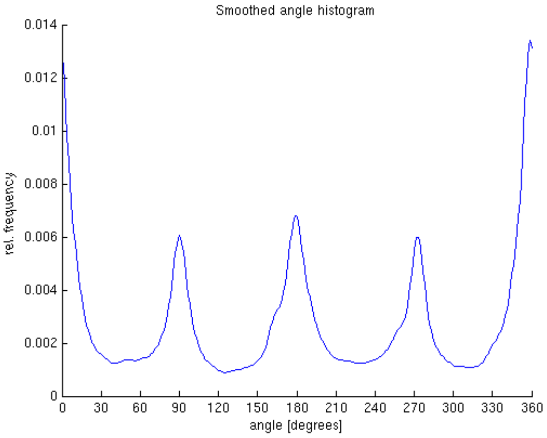


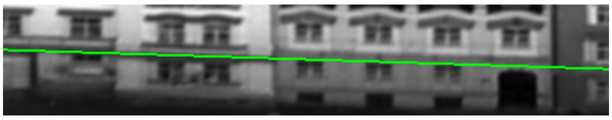
Figure 1: This figure illustrates the line extraction process in the heightfield. (a) shows the original heightfield, (b) depicts the gradient image (Sobel), (c) is the building-layer of the classification for the test area and (d) overlays the extracted lines (green) with the heightfield.

where $normal$ is the normal vector of the facade plane, $origin$ is the position of the camera and $anchor$ is the center of the facade plane.

Once the optimal camera has been determined, the corresponding image is perspectively correctly resampled. A Gaussian filter is then applied to remove small artifacts. For each pixel in the smoothed image the x and y derivative is calculated and stored in a $(\phi, magnitude)$ vector, where ϕ gives the angle of the derivative vector and $magnitude$ its Euclidean length. Subsequently all pairs with a small $magnitude$ are removed. The remaining members of the vector are used to construct an orientation histogram. Each peak in that histogram corresponds to one strong line direction in the texture. This peak estimation is more stable if the histogram is smoothed beforehand. Because of our assumption that a facade contains horizontally and vertically aligned structures, we conclude that the peak closest to zero should in fact be exactly at zero to make the facade plane parallel to the real facade. Figure 2 shows an orientation histogram and the corresponding warped texture. The green line is the estimated principal horizontal line. There are four peaks clearly visible, each accounts for the principal directions (up, down, left, right) of the gradients. To have a parallel facade those four peaks should be at exactly 0, 90, 180 and 270 degrees respectively. The angle histogram enables us to calculate an orientation change which compensates this deviation of the peaks. Figure 3 illustrates this intersection procedure. The detected line direction is used to create a plane which contains the camera center and a line on the facade with this direction. This plane is intersected with a horizontal plane to give the new orientation of the facade estimation.



(a)



(b)

Figure 2: (a) A smoothed orientation histogram with its four distinct peaks in horizontal and vertical direction. (b) shows a part of the corresponding texture with the principal horizontal line direction marked with green.

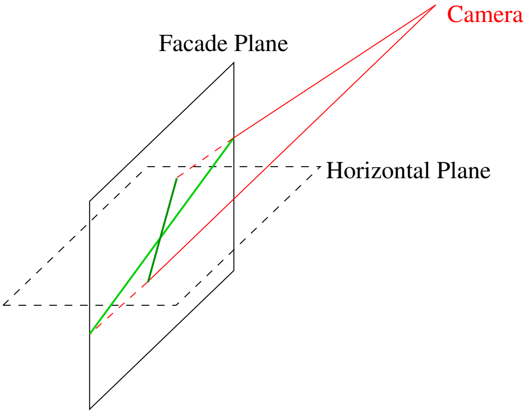


Figure 3: The lines from camera center to the endpoints of the detected line are intersected with the horizontal plane. The new plane defined by this horizontal line is parallel to the real facade.

2.4 Correlation Optimization

In the third and last step the facade plane is further refined to increase the correlation of warped textures from different views. At the beginning the facade plane can not be used to correlate the views at the full resolution level because even an offset of a few pixels may cause a very bad correlation value. Therefore a hierarchical approach is used to overcome this problem. Each warped texture is turned into an image pyramid and starting with the coarsest level the correlation optimization is performed until the highest resolution level is reached. The algorithm is detailed in Algorithm 1. Figure 4 illustrates the process of generating new hypotheses starting with an initial facade plane. The illustration is a top view because it is assumed that facades are always vertical. Figure 6 shows how the optimization on different resolution levels converges to the final position.

The correlation score is calculated using the normalized cross correlation with an adaptive window size depending on the resolution level - on the highest level a smaller window is used as on lower resolution levels. Because of the different resolution the correlation window always covers approximately the same region. Also a correlation truncation (lower boundary) at 0.8 is used to improve the stability of the correlation as explained in (Scharstein and Szeliski, 2002).

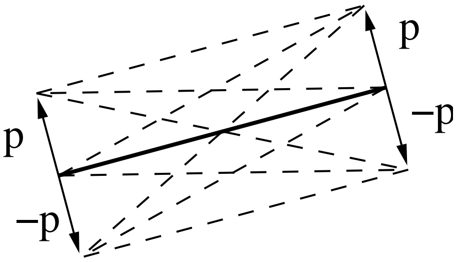


Figure 4: For a given facade plane a translation vector p is calculated which shifts each end of the facade plane and generates therefore eight new hypotheses. New hypotheses are marked with dashed lines.

Algorithm 1 Correlation Optimization

Require: At least two views for a facade

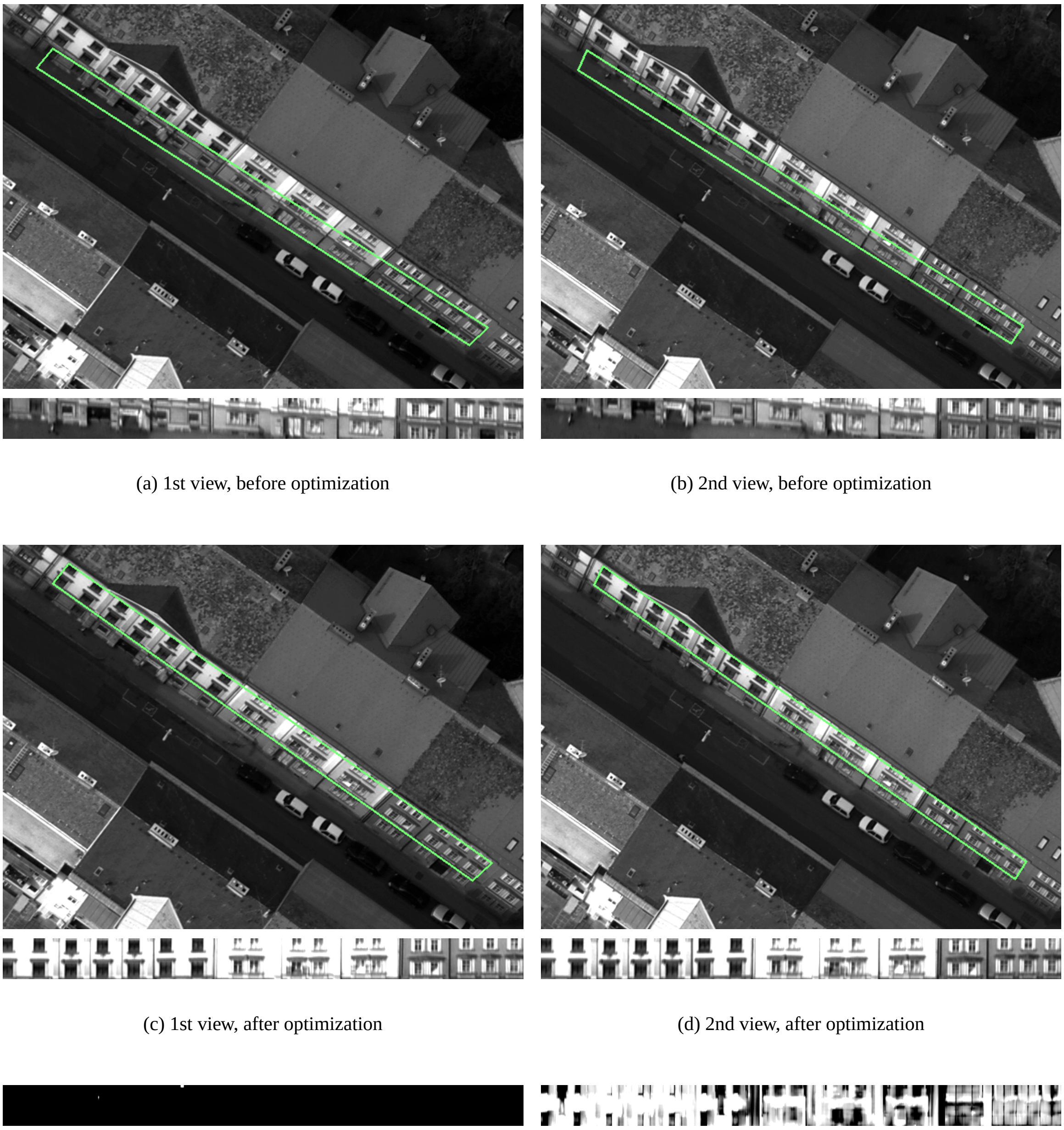
```
repeat
    calculate a translation vector $p$ normal to the facade plane
    such that the length of the projection at the current resolution level is approximately one pixel.
    create new hypotheses by moving each end of the facade plane independently back and forth along the translation vector.
    if no higher correlation can be obtained by any hypothesis, switch to a higher resolution level.
until highest resolution level is reached
```

The quality of the optimization can be judged by the correlation factor. Values of above approximately 0.8 indicate that the estimate snapped to the real facade, whereas lower values may either be due to the fact, that there are occlusions (trees are very disturbing especially in inner courtyards) in the images or that the facade can not be satisfyingly be approximated with one plane because of balconies or depth jumps in the real facade. Figure 5 illustrates an optimization of one facade. Looking at the warped patches one can observe the improvement in positioning the facade.

3 RESULTS AND DISCUSSION

Figure 7 illustrates the result of the optimization on one corner of the building. One can see that the initialization of the facade is in fact the eave line of the roof, whereas the optimization results in the correct position which is slightly translated inwards.

A rendering of the complete building block is depicted in Figure 8. It consists of 21 facades planes and 46 roof planes. The 3D model creation is subject of current research and therefore does not exploit all of the information available. As mentioned in the paragraph above the gap between facade and eave line can be



(a) 1st view, before optimization

(b) 2nd view, before optimization

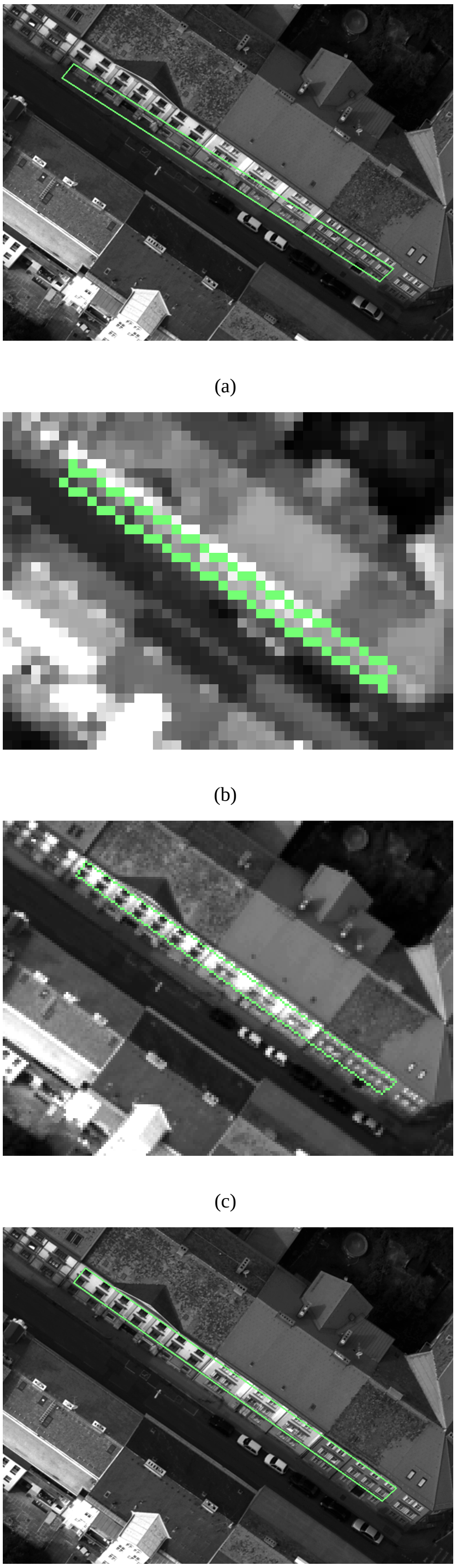
(c) 1st view, after optimization

(d) 2nd view, after optimization

(e) correlation before optimization

(f) correlation after optimization

Figure 5: Facade estimation before and after optimization. Two out of three views are shown (left and right). The top two rows represent the initial estimate, the regions marked with the green quadrangle are rectified and shown in the next row. It is clearly visible that the initial estimate deviates from the real facade. After the optimization (third and fourth row) the correct placement can be observed in the rectified images which are nearly identical. This is confirmed by the correlation images (bottom row): the left correlation image shows the correlation for the initial estimate, the right image is calculated after the optimization. The final correlation score is about 0.87.



(a)

(b)

(c)

(d)

Figure 6: Four steps in the correlation optimization process: the green lines delineate the estimation after (a) initialization, (b) optimization on the lowest level, (c) medium resolution level and (d) highest resolution level.

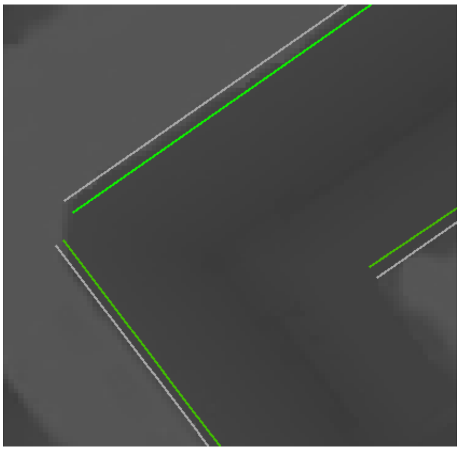


Figure 7: A zoom onto a corner of the building: the gray line denotes the initialization, whereas the green line indicates the position with the optimized correlation. The difference of these positions accounts for the offset between eave line and real facade.

reconstructed (either by comparing the initial estimate and optimized facade or by looking at the correlation image because the correlation will drop where the facade is occluded by the roof) and included in the 3D model. The depicted model lacks this improvement and therefore the roof gets projected onto the facade at the top where in fact the eave line should extend.

4 CONCLUSIONS AND FUTURE WORK

This paper presents a novel approach to improve the location of facade planes using two image-based optimization techniques. The success of such optimizations can easily be judged using the correlation score. The algorithms are outlined and their results are demonstrated using a real world example.

The preliminary results are visually appealing, but further research is required. Especially the exact reconstruction of the offset between eave line and real facade is very promising. The fusion of optimized facade planes, roof planes and offset of the eave lines into a three dimensional model is subject of future research and presents a major step towards fully automated city modelling.

ACKNOWLEDGEMENTS

This work has been done in the VRVis research center, Graz/Austria (http://www.vrvis.at), which is partly funded by the Austrian government research program Kplus. We would also like to thank Vexcel Corporation (http://www.vexcel.com) for supporting this project.

REFERENCES

Berthod, M., Gabet, L., Giraudon, G. and Lotti, J., 1995. High resolution stereo for the detection of buildings. In: A. Grun, O. Kubler and P. Agouris (eds), Automatic Extraction of Man-Made Objects from Aerial and Space Images, Birkhäuser, pp. 135–144.

Brenner, C., 2000. Towards fully automatic generation of city models. In: International Archives of Photogrammetry and Remote Sensing, Commission III, Vol. 33, pp. 85–92.

C. Vestri, F. D., 2000. Improving correlation-based dems by image warping and facade correlation. In: In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), p. 1438 ff.

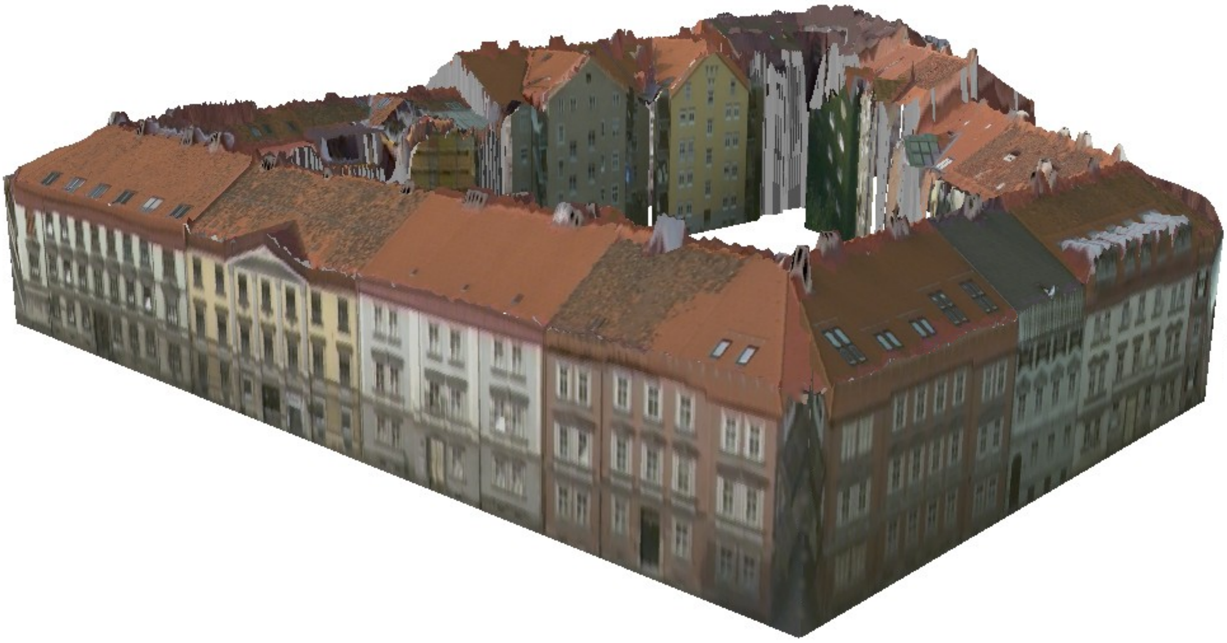


Figure 8: A 3D rendering of one building with optimized facades.

Coorg, S. and Teller, S., 1999. Extracting textured vertical facades from controlled close-range imagery. In: In Proceedings IEEE Conference on Computer Vision and Pattern Recognition, pp. 625–632.

Cord, M., Paparoditis, N. and Jordan, M., 1998. Dense, reliable, and depth discontinuity preserving dem computation from very high resolution urban stereopairs. In: ISPRS Symposium, Cambridge (England).

Gruber-Geymayer, B. C., Klaus, A. and Karner, K., 2005. Data fusion for classification and object extraction. In: Proceedings of CMRT05, Joint Workshop of ISPRS and DAGM, pp. 125–130.

H. Gross, U. Thoennessen, W. v. H., 2005. 3d-modeling of urban structures. In: Proceedings of the ISPRS Workshop CMRT 2005, pp. 137–142.

Haala, N., Brenner, C. and Statter, C., 1998. An integrated system for urban model generation. In: ISPRS Commission II Symposium, Cambridge, England.

Samadzadegan, F., Azizi, A., Hahn, M. and Lucas, C., 2005. Automatic 3d object recognition and reconstruction based on neuro-fuzzy modelling. In: ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 59, pp. 255–277.

Scharstein, D. and Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In: International Journal of Computer Vision, Vol. 47, pp. 7–42.

T. Werner, A. Z., 2002. New technique for automated architectural reconstruction from photographs. In: In Proceedings of the European Conference on Computer Vision (ECCV), pp. 541–555.

Vosselman, G., 1999. Building reconstruction using planar faces in very high density height data. In: Proceedings of the ISPRS Automatic Extraction of GIS Objects from Digital Imagery, pp. 87–92.

Vosselman, G. and Dijkman, S., 2001. 3d building model reconstruction from point clouds and ground plans. In: International Archives of Photogrammetry and Remote Sensing, Vol. 34, pp. 37–43.

Weidner, U., 1996. An approach to building extraction from digital surface models. In: Proceedings of the 18th ISPRS Congress, Commission III, pp. 924–929.

Zach, C., Klaus, A. and Karner, K., 2003. Accurate dense stereo reconstruction using 3d graphics hardware. Eurographics 2003 pp. 227–234.

DETECTION OF PLANAR PATCHES IN HANDHELD IMAGE SEQUENCES

Olaf Kähler, Joachim Denzler

Friedrich-Schiller-University, Dept. Mathematics and Computer Science, 07743 Jena, Germany
{kaehler,denzler}@informatik.uni-jena.de

KEY WORDS: planar patches, homography, feature detection, degeneracy

ABSTRACT:

Coplanarity of points can be exploited in many ways for 3D reconstruction. Automatic detection of coplanarity is not a simple task however. We present methods to detect physically present 3D planes in scenes imaged with a handheld camera. Such planes induce homographies, which provides a necessary, but not a sufficient criterion to detect them. Especially in handheld image sequences degenerate cases are abundant, where the whole image underlies the same homography. We provide methods to verify, that a homography does carry information about coplanarity and the 3D scene structure. This allows deciding, whether planes can be detected from the images or not. Different methods for both known and unknown intrinsic camera parameters are compared experimentally.

1 INTRODUCTION

The detection and tracking of features is one of the preliminaries for many applications, ranging from motion analysis to 3D reconstruction. Depending on the complexity of features, more or less knowledge can be gained directly from them. The typical approach is to match corresponding point features over an image sequence, which is solved for many applications (Shi and Tomasi, 1994). Inferring information about the 3D structure of the scene can benefit however from additional constraints, e.g. coplanarity of points (Bartoli and Sturm, 2003). In fact planes are relatively easy to handle as features and do have many useful geometric properties.

Planes have caught the interest of research before. Linear subspace constraints on the motion of planes have been elaborated and used for separating independently moving objects (Zelnik-Manor and Irani, 1999). For the representation of video there are many applications related to planes or so called layers, either for efficient coding exploiting the 2D object motion (Baker et al., 1998, Odone et al., 2002), or aimed towards an interpretation of the 3D scene structure (Gorges et al., 2004). The benefits of incorporating coplanarity constraints (Bartoli and Sturm, 2003) or of explicitly using planes for 3D reconstruction (Rother, 2003) have been investigated, too. Also efficient auto-calibration algorithms in planar scenes are possible (Triggs, 1998). More recently many of the above results have been combined to allow explicit tracking of 3D camera motion from image intensities (Cobzas and Sturm, 2005).

Despite many applications, the automatic extraction of planar regions is still a difficult task. The work of Baker (Baker et al., 1998) was one of the first setting the trend to use homographies for finding planes. Later algorithms made use of random sampling to automatically detect points mapped under a common homography (Schindler, 2003). Using a sparse set of tracked point features, random sampling was also applied for a Least Median of Squares regression to detect a *dominant* homography in the scene (Odone et al., 2002). The dominant homography is defined as the one transferring all known points with the least median transfer error. The extraction of dominant homographies is iterated to find smaller and smaller planar patches. A very similar algorithm was given in (Gorges et al., 2004). The dominant homography in that case is defined as the one transferring most points correctly.

The mentioned works concentrate on finding point features or image regions underlying a common homography. This is a necessary condition for the points to reside on the same 3D plane, it is not a sufficient one however. A very simple case is a camera not moving at all between two frames. All points are then transferred with the same homography, the identity matrix. Yet the points may reside in many different 3D scene planes. A similar well known situation occurs, if the camera undergoes a pure rotation. Especially when processing image sequences from handheld or head-mounted cameras, both of these cases are abundant and ignoring them leads to erroneous planes being detected. Detection of planar patches in a scene should therefore not only find image regions under a homography, but also decide, whether coplanarity can be inferred from the detected homographies.

The detection of related degenerate cases is an important issue in many different computer vision tasks, yet rarely addressed in research. A seminal work on the topic (Torr et al., 1999) is considering the case of degeneracy for the estimation of the fundamental matrix. The basic task in that work is to find a guidance for feature matching, either the epipolar geometry or a homography warp on the whole image. This is highly related to our problem and we will develop similar methods in our work.

The rest of the paper is organized as follows. In section 2 we will shortly introduce the notation and present a useful decomposition of homography matrices. Finding homographies from known point correspondences is reviewed in section 3. The task of deciding on coplanarity from given homographies is elaborated in section 4. In section 5 an experimental evaluation of the developed methods is given. Some final remarks on further work and conclusions will sum up the results in the end.

2 PRELIMINARIES

Throughout the work we will use the standard projective camera model projecting world points $\boldsymbol{X}$ onto image points $\boldsymbol{x}$ with $\boldsymbol{x} = \alpha K(R\boldsymbol{X} + \boldsymbol{t})$. The matrix K is an upper triangular matrix containing intrinsic parameters, R is a rotation matrix and $\boldsymbol{t}$ the translation vector. We typically need two camera frames and two sets of camera parameters, which are then denoted with a index, e.g. K_1 and K_2. Restricting to two frames it is sufficient to know the relative motion, and hence we set $R_1 = \mathrm{Id}$, $\boldsymbol{t}_1 = \boldsymbol{0}$ and $R_2 = R$, $\boldsymbol{t}_2 = \boldsymbol{t}$.

A world plane is defined by the inner product $\boldsymbol{n}^T\boldsymbol{X} = d$, with inhomogeneous 3D vectors $\boldsymbol{n}$ and $\boldsymbol{X}$, and a scalar d. Every world plane projected into two images induces a homography between the two images, a 2D-2D projective transformation H. This H maps the projections $\boldsymbol{x}_1$ of world points on the plane onto corresponding points $\boldsymbol{x}_2$ in the second projection. This is easily shown using the relative motions:

$$\begin{aligned}
\boldsymbol{x}_2 =& \alpha_2 K_2(R_2\boldsymbol{X} + \boldsymbol{t}_2) = \alpha_2 K_2(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T)\boldsymbol{X} \\
\boldsymbol{X} =& \frac{1}{\alpha_1} R_1^{-1} K_1^{-1}\boldsymbol{x}_1 - R_1^{-1}\boldsymbol{t}_1 = \frac{1}{\alpha_1} K_1^{-1}\boldsymbol{x}_1 \\
\boldsymbol{x}_2 =& \alpha \underbrace{K_2(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T)K_1^{-1}}_{=H} \boldsymbol{x}_1 \qquad (1)
\end{aligned}$$

The homography matrix H therefore is only defined up to an unknown scale.

3 FROM POINTS TO HOMOGRAPHIES

To detect planar patches we first establish point correspondences between consecutive image frames in the sequence using KLT-tracking (Shi and Tomasi, 1994). As points on planar regions underlie a homography, the first step in finding these planar regions is to establish groups of point features correctly transformed by a common 2D-2D projective transformation. Different approaches to do this mainly use the two concepts of random sampling and iterative dominant homography estimation. Before going into their details in sections 3.2 and 3.3, we will shortly review the computation of homographies.

3.1 Computation of Homographies

In an usual approach the 2D homography can be estimated from 4 point correspondences by solving the following linear equation system for the entries of H:

$$\boldsymbol{x}_2 = \alpha H \boldsymbol{x}_1 \qquad (2)$$

With equality up to scale, each pair of corresponding points leads to 2 independent equations in the entries of H. As the matrix H can only be computed up to scale, it has 8 degrees of freedom. Hence four points determine the entries of the matrix H.

With known epipolar geometry however, even three points are a sufficient minimum parameterization of planes. There are several ways of exploiting this (Hartley and Zisserman, 2003). This will basically enforce the computation of homographies compatible with the epipolar geometry, in the sense that the single globally rigid scene motion stored in the epipolar geometry is enforced to be also valid for all points of the homographies. This will fail however, if there are multiple independent motions. In general also the computation of epipolar geometry frequently gives rise to numerical problems.

For our work we do not use the epipolar constraints, but compute homographies directly from equation 2. We typically use more than 4 points and solve the overdetermined system using SVD techniques.

3.2 Random Sampling

The RANSAC approach was used e.g. in (Schindler, 2003) to detect points underlying the same homography. Basically the idea of model fitting with random sampling is very intuitive. Starting with a minimal set of random samples, which define an instance of the model, the support for this instance among the other available samples is measured. In the end we keep the hypotheses with highest support.

For our homography problem, the algorithm has to randomly select points from all known correspondences, so that the parameters of the homography can be determined. This means three random points with known epipolar geometry or four points in the more general case. The errors for transferring the remaining point correspondences with this homography can be computed. Each point correctly transferred up to e.g. 2 pixels difference can be counted as supporting the hypothesis that the points are coplanar. If only the initially selected points support the hypothesis, these points are most likely not coplanar and the computed homography does not have any physical meaning.

This idea of extending an initial homography H_i to more point correspondences can be applied iteratively. After an extension step, a new homography H_{i+1} can be computed with the additional points included. The new homography matrix H_{i+1} can again be extended to all other points correctly transferred. The iteration ends, if no more points are added to the computations. With this approach the result is more robust against small matching inaccuracies in the initially selected points.

3.3 Iterative Dominant Homography Estimation

In various works (Odone et al., 2002, Gorges et al., 2004) the homography explaining most point correspondences is called the *dominant* homography. To find this dominant homography, first the RANSAC algorithm is applied as above. From all sampled candidates only the single best one is kept however. This is defined to be either the one with Least Median overall transfer error (Odone et al., 2002), or the one transferring the largest number of points correctly (Gorges et al., 2004). This dominant homography of the scene is accepted as a planar region, the covered points are removed and another iteration step is started to find the dominant homography of the remaining points.

For the least median error method, the breakdown point is at 50% outliers. If there are many small planes in the scene each covering only a small portion of the image, the homographies found will thus explain only a small portion of all point correspondences. The homography with least median transfer error is then almost arbitrary, and will not necessarily be exactly valid for any but the initially sampled points used to construct it. We therefore decided not to use the least median error method, but to count the points correctly transferred up to e.g. 2 pixels tolerance instead.

3.4 Locality Constraints

If the mentioned homography detection algorithms are applied as described above, they will mostly detect *virtual* homographies. These are induced by *virtual* planes, i.e. geometrically valid 3D planes with many observable points on them, but without any corresponding physical plane. An example can be seen in Figure 1. Note that from geometry and the computed homographies alone, these virtual planes do well represent sets of coplanar points and there is no way to detect them. Additional constraints to prevent the virtual planes therefore can not result from pure photogrammetry. Two basic approaches occur in the literature.

In the work of (Gorges et al., 2004) an explicit locality criterion is used. Only points in a certain neighborhood region are sampled to compute the initial hypotheses in the RANSAC algorithm. In the extension steps, points outside the boundaries of the local neighborhood can be taken into account as well. This might seem like a

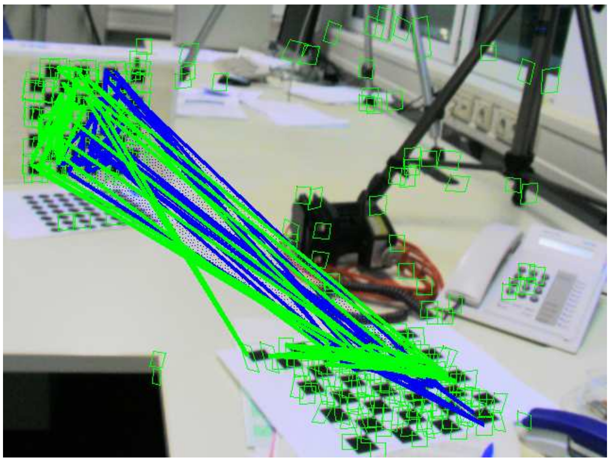


Figure 1: The points connected by the green and blue lines are lying on two virtual planes, which represent coplanar points on planes that do not correspond to any physical object plane

heuristic at first, however it directly facilitates the detection of *locally* planar structures. Starting from the locally planar neighborhood, the iterative extension of the homography to more points still allows the detection of larger planes with arbitrary shape. In our experiments this method practically eliminated the detection of virtual planes.

A more complex but in essence quiet similar criterion was used in (Schindler, 2003). There the plane detection is initialized with equilateral triangles selected by random sampling. All points inside the triangles have to match the same homography, and only then a region growing is started. This is basically an extension of the mentioned locality constraint above, first from an arbitrary shaped neighborhood to the convex interior of a triangle and second from sparse point correspondences to a dense constraint on all image points. Due to the higher complexity with basically the same effect, we have not investigated this method further.

4 FROM HOMOGRAPHIES TO PLANES

Detecting image regions underlying one common homography is only the first step for finding planar patches in an image sequence. All planar image regions will underlie a homography, but not all image regions underlying a homography are necessary coplanar. We will first show that these cases occur exactly if there is no translational motion between the two frames under consideration. Further we will present several methods for detecting these cases in different scenarios, like known or unknown intrinsic camera parameters.

The mentioned problematic cases are directly apparent from the homography decomposition given in equation 1:

$$H = K_2(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T)K_1^{-1}$$

If the term $\boldsymbol{t}\boldsymbol{n}^T$ vanishes for planes with arbitrary normals $\boldsymbol{n}$, the homographies do not contain any information about the planes, but only consist of $K_2RK_1^{-1}$. On the other hand any homography matrix H containing the second term, has one unique plane with normal $\boldsymbol{n}$ inducing it.

The term vanishes for arbitrary $\boldsymbol{n}$ if and only if $\boldsymbol{t} = 0$. In that case we have a pure rotational motion or change of intrinsic parameters and can not infer anything on the 3D structure. To ensure, a homography does contain relevant information about a 3D plane, we therefore have to test for a translation $\boldsymbol{t} \neq 0$. A first class of testing methods is to analyze a single homography matrix and check it for a particular form. A different class is taking into account additional information from other correspondences.

Algorithms in the first class are testing, whether a given H is of the form $K_2RK_1^{-1}$. Note these methods will always fail to identify the plane at infinity. This is the plane containing all the vanishing points, and it has the normal $\boldsymbol{n} = 0$. So the homography of this plane is always of the form of a pure camera rotation. Only once a translational part is detected in the homography of any other plane, it could be inferred that $\boldsymbol{t} \neq 0$ and hence the homography $H = K_2RK_1^{-1}$ must be induced by the plane with normal $\boldsymbol{n} = 0$.

This inference, like the approaches using knowledge from other correspondences, can only be used in case of a globally rigid motion of the scene however, and not in case of independently moving objects in the scene. This becomes apparent for the example of an object moving in front of a static camera. The plane induced homographies of the object do have a translational motion part, and the whole static background is underlying the same homography. But the background does not necessarily consist of one single plane. If a static scene is assumed on the other hand, the additional information will ease the task of detecting motions without translations.

4.1 Known Intrinsic Parameters

If the intrinsic camera parameters are known, a simple and straight forward test for the translational part in a homography is possible. Multiplying the homography matrix H with the intrinsic parameter matrices K_1 and K_2^{-1} from left and right we get:

$$\begin{aligned} H' = K_2^{-1}HK_1 =& \alpha K_2^{-1}K_2(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T)K_1^{-1}K_1 \\ =& \alpha(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T) \end{aligned}$$

It is obvious that the term $\frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T$ vanishes if $\boldsymbol{t} = 0$, i.e. there is no translational part in the camera motion. The larger $\boldsymbol{t}$, the more is H' dominated by a rank-1 part and deviating from the pure rotation matrix R.

A test for H' to be a rotation matrix is given by the singular value decomposition. For the rotation matrix R, all singular values are equal to 1. Taking into account the unknown scale factor α, the ratio of largest to smallest singular value of H' will therefore be 1 if $\boldsymbol{t} = 0$ or $\boldsymbol{n} = 0$. For our experiments we used a slightly less restrictive threshold of 1.2 for the ratio.

4.2 Unknown but Constant Intrinsic Parameters

Needing knowledge of the intrinsic parameters clearly is a shortcoming of the method above. We will consider the next simple case, where the intrinsic camera parameters are unknown, but known to be constant. This scenario is of great practical relevance and has been studied before (Triggs, 1998). Many and especially cheap cameras are not equipped with a zoom-lense and hence fulfill the requirement.

In the case of a constant intrinsic parameter matrix $K = K_1 = K_2$, the homography matrix H is similar (i.e. conjugate) to the matrix $R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T$. This means the two matrices do have the same determinant, eigenvalues and some more properties which are not relevant here, although the singular values might differ.

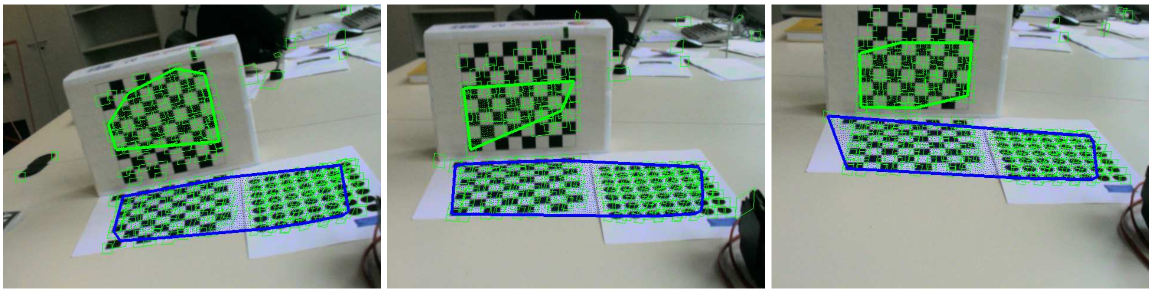


Figure 2: Excerpts of a calibration pattern scene with planar patches detected in the individual frames shown as polygons with thick boundary lines.

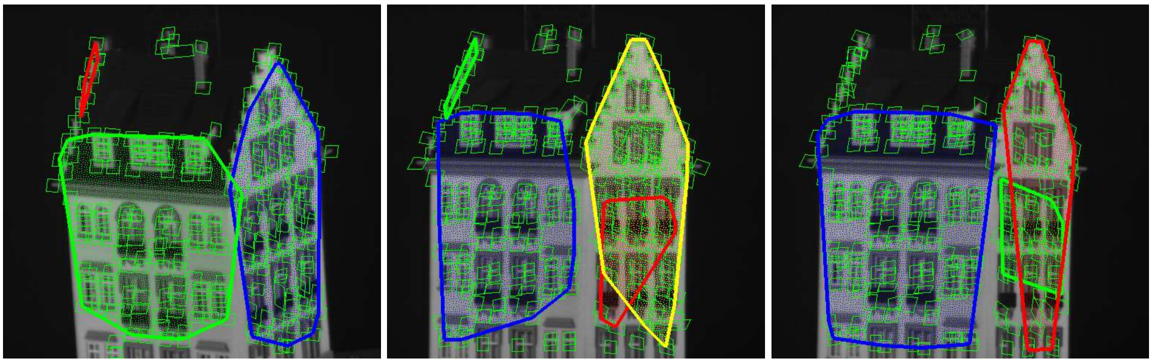


Figure 3: Excerpts of an architectural scene with the thick polygons delineating planar patches found from point correspondences.

The equivalence of eigenvalues is derived from:

$$\begin{aligned}\det(\frac{1}{\alpha}H - \lambda \mathrm{Id}) &= \det(KRK^{-1} + \frac{1}{d}K\boldsymbol{t}\boldsymbol{n}^T K^{-1} - \lambda KK^{-1}) \\ &= \det(K)\det(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T - \lambda \mathrm{Id})\frac{1}{\det(K)} \\ &= \det(R + \frac{1}{d}\boldsymbol{t}\boldsymbol{n}^T - \lambda \mathrm{Id})\end{aligned}$$

The eigenvalues are given as the roots of this characteristic polynomial and are hence identical for the two matrices. Using this result and the equality $\det(A + \boldsymbol{x}\boldsymbol{y}^T) = (1 + \boldsymbol{y}^T A^{-1}\boldsymbol{x})\det(A)$ it follows, that H has the same eigenvalues up to scale with the rotation matrix R, if and only if $\boldsymbol{n}^T R^T \boldsymbol{t} = 0$. All three eigenvalues of the rotation matrix R do have the same absolute value 1. So do the eigenvalues of the homography matrix H up to the common scale α, if the intrinsic parameters are constant. The ratio of largest to smallest absolute eigenvalue hence provides a means of detecting cases with $\boldsymbol{n}^T R^T \boldsymbol{t} = 0$. In our experiments we again used a ratio of 1.2 as a threshold, to tolerate the effects of slight noise.

The condition tested by this criterion is either met for $\boldsymbol{t} = 0$ or $\boldsymbol{n} = 0$ or if the vectors $R\boldsymbol{n}$ and $\boldsymbol{t}$ are orthogonal. This provides a slightly over-sensitive test for the detection of translations. The case where this measure generates false alarm is a translation in a plane parallel to the plane inducing H.

As mentioned before, these two tests can be extended to a global measure, if we assume a globally rigid motion. Detecting a translational part in any homography matrix, we can assume the whole scene has undergone a translation, and hence every observed homography H carries information about coplanarity. This way the cases where the test is oversensitive can be avoided as well, unless the camera motion is parallel to all planes in the scene.

4.3 Global Homography

Another very intuitive idea exploiting the rigid motion constraint is to simply count, how many points are not correctly transferred between the frames using the homography H. In the case of no translation between the frames, the homography matrix for any plane will be the same. The second, parallax term will vanish and $H = H_\infty = K_2 R K_1^{-1}$. Therefore if all points are transferred with the homography H_∞, the motion of the points was most likely caused by a camera movement without translation. For practical purposes a small portion of outliers should be allowed, depending on the quality of the point correspondences found. In our experiments we considered a homography as global, if it transferred more than 80% of all points with a small transfer error.

However, again there are cases where this test will fail, e.g. if only one plane is visible in the scene. This plane is not necessarily the plane at infinity with $\boldsymbol{n} = 0$, but could as well be a real object plane filling the whole view. Knowledge of the intrinsic parameters and one of the tests above could decide upon this ambiguity.

4.4 Epipolar Geometry

Another way of explicitly using points not residing in the potential plane is to take into account the epipolar geometry. Note with the usual 8-point-algorithm the fundamental matrix F can only be determined up to a two-parameter family of matrices in the case of all points residing in the same 3D plane or no translation occuring between the frames (Torr et al., 1999). Testing for these rank-deficiencies when computing the epipolar geometry will therefore allow the detection of cases without translation.

This test basically has the same restrictions as for the global homography computation before. In fact the same condition that all points underly a common homography is only tested differently here. But again the numerically problematic epipolar geometry is needed, and a small portion of incorrect point correspondences could severely affect this method.

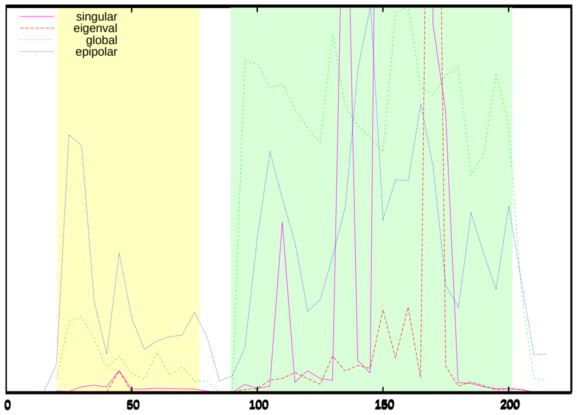


Figure 4: Confidence of different criteria that a translational camera motion was present in the individual frames of a sequence. The yellow background indicates ground-truth frames with pure camera rotation, the green background indicates general motion

5 EXPERIMENTS

We have presented a method for detecting homographies and several different methods for checking the information on planarity contained in a homography. For the experimental evaluation we follow a similar structure. First the results from homography detection are shown qualitatively, as this part of the work can hardly be evaluated quantitatively. For the different methods of detecting planes from homographies a detailed evaluation is given in section 5.2.

5.1 Detection of Homographies

Detailed error analysis of the decomposition of image sequences into planes is difficult. First of all real video sequences do not provide a ground truth segmentation that could be used for numerical error analysis. But even more important such a decomposition into planar patches is not unique. Planar patches detected from sparse point correspondences are in fact typically smaller than the physical planes they represent, and finding the exact delineations of planar regions is a different issue not covered here.

We have performed experiments with different scenes and environments. In some rather artificial sequences, checkerboard calibration patterns were placed on a table and recorded with a handheld camera. The checkerboards provide high contrasts and sharp corners, that can be tracked well and provide good point correspondences over the image sequence. Another set of images was taken from publicly available sequences of architectural scenes showing model buildings. These kind of scenes are a typical application scenario for planar patch detection.

Example planes found with our algorithms are shown in Figure 2 for a calibration pattern scene and in Figure 3 for an architectural scene. Note the detected planes do represent planar image areas and correspond to physically present planes in the scene, no virtual planes are detected. As it was expected, the detected planes are typically smaller than the physical planes due to the sparseness of the point correspondences used to find them. Points assigned to a plane were not removed and therefore some planes are detected several times and do overlap. On the other hand this allows correct handling of points on the delineation of two planar patches. Note that point correspondences not lying in any of the planes are correctly identified, so if the observed objects are not planar, no false planes are detected.

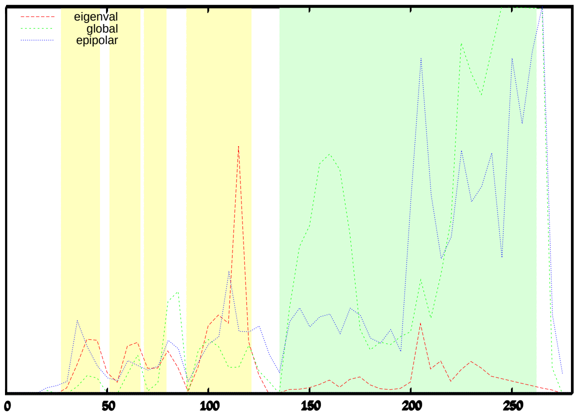


Figure 5: Confidence of different criteria that a translational camera motion was present in the individual frames of a sequence. The yellow background indicates ground-truth frames with purely zooming camera, the green background indicates general motion

In these example images, the planar patches are detected only and not kept from one frame to the next. Depending on the application, this temporary knowledge of coplanarity might be sufficient. Otherwise a homography tracking can be applied and simple methods to prevent overlapping planes from being detected over and over again could be thought of.

5.2 Detection of Cases Without Translation

In section 4 we have presented various ways of detecting camera motions without translational part. In these cases the homographies do not give us any information on coplanarity of points and hence no planes can be detected using the homographies.

To evaluate the performance of the individual methods, some video sequences with controlled camera motion were recorded. Mounted on a tripod, a camera captured a motion sequence with at least approximately a pure rotational motion. With a motorized zoom it was further possible to take influence on the intrinsic camera parameters without any other camera motion. So it was possible to acquire a ground truth classification of the camera motion and to compare the detected motion classes of “translation” and “no translation” with that ground truth.

In Figure 4 the different criteria from section 4 were compared for a sequence with pure camera rotation. The ground truth information is shown as a background coloring, where the white parts indicate no camera motion, yellow parts a camera rotation and green parts a sequence of images with non-zero camera translation. For each image frame the tests computed one value per detected homography, e.g. one ratio of eigenvalues. For the figure these values were averaged over several such tests (e.g. over the 5 planes detected in this frame). Note that due to constant and known intrinsic camera parameters, all criteria could be applied for the sequence with pure rotation. The short times with completely static camera were clearly identified by all criteria. The translational movement can also be clearly identified from the global homography criterion (line “global”). Also the singular value and eigenvalue criteria allow a classification of the camera movement, with some small false alarms around frame 45. The epipolar criterion seems to be severely affected by incorrect point matches however.

A similar comparison is shown in Figure 5 for variable intrinsic parameters, i.e. a zooming camera. Note we do not have accurate knowledge of the intrinsic parameters in this case and hence skip the singular value criterion. To allow comparison we did test the eigenvalue criterion however. It can be seen that the criterion incorrectly classifies the zooming camera, as expected. As described in section 4.2 the criterion needs constant intrinsic parameters to be valid. Both the epipolar and especially the global homography criteria allow a relatively good identification of the translational camera motion, however the results are far less clear compared to the sequence with a rotating camera.

Overall if the intrinsic calibration is known or constant, this knowledge should be used, as was seen in the test with pure camera rotation. In other cases the global homography criterion seems to perform sufficiently good as well. This was also confirmed in further qualitative tests with different sequences. The epipolar geometry most likely suffers from numerical instabilities and outliers of the point matching. Skipping the tests for a camera translation, one "plane" is detected covering all point correspondences in the image, unless a translational motion is present.

6 FURTHER WORK

The criterion derived from epipolar geometry currently does not provide a useful measure for the translational part, most likely due to the numerical instability of computing epipolar geometry. The normalized eight-point-algorithm used in this work already performs better than using unnormalized pixel coordinates, but still it is not robust against incorrect point matches. Using an improved algorithm could also render the epipolar geometry useful for homography estimation, as described in section 3.1.

Having found the coplanar point sets, the exact delineations of the planes are still unknown. A pixel-wise assignment of image points to physical planes is needed for various applications like exact scene representation or image based rendering. This can be solved with region growing algorithms, as was done e.g. in (Gorges et al., 2004) or with graph-cut related techniques. Both do need initial seed regions that can be generated robustly from the image data with our algorithms. And both have to be made aware of cases where it can not be inferred on coplanarity from homographies.

7 CONCLUSION

The aim of this work was to automatically detect planar features in image streams from handheld cameras. Various applications were mentioned in the introduction. In most of these a manual selection of planes is used. The few works dealing with the automatic detection of planes concentrated of finding image regions under homography. We have given a brief overview and presented a similar algorithm based on random sampling and iterative estimation of the dominant plane.

As we have shown, finding homographies between the frames of a sequence can not be enough for the detection of planes however. For camera movements without 3D translational part the common homography is not a sufficient criterion for the coplanarity of points. We have presented various methods to detect such cases and to prevent planes from being detected in case of no camera translation. These methods made use of known or constant intrinsic camera parameters or of the static-scene assumption, and hence can be applied to many different application scenarios.

In the experiments we have first shown that physically meaningful planes can be detected with the suggested approach. Also a comparison of the various methods for plane extraction from the homographies was given. Especially the cases of pure camera rotation and varying intrinsic parameters were investigated, exactly the cases where a homography does not contain information about the coplanarity of points. The sequences with a pure rotation could be identified clearly. It was more difficult to separate a change of intrinsic parameters from general camera motion. But using the appropriate methods it was possible as well.

REFERENCES

Baker, S., Szeliski, R. and Anandan, P., 1998. A layered approach to stereo reconstruction. In: Proc. Computer Vision and Pattern Recognition, Santa Barbara, CA, pp. 434–441.

Bartoli, A. and Sturm, P., 2003. Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene. International Journal of Computer Vision 52(1), pp. 45–64.

Cobzas, D. and Sturm, P., 2005. 3d ssd tracking with estimated 3d planes. In: Proc. Second Canadian Conference on Computer and Robot Vision, Victoria, Canada, pp. 129–134.

Gorges, N., Hanheide, M., Christmas, W., Bauckhage, C., Sagerer, G. and Kittler, J., 2004. Mosaics from arbitrary stereo video sequences. In: Lecture Notes in Computer Science, Vol. 3175, 26th DAGM Symposium, Springer-Verlag, Heidelberg, Germany, pp. 342–349.

Hartley, R. and Zisserman, A., 2003. Multiple View Geometry in Computer Vision. 2nd edition edn, Camebridge University Press.

Odone, F., Fusiello, A. and Trucco, E., 2002. Layered representation of a video shot with mosaicing. Pattern Analysis and Applications 5(3), pp. 296–305.

Rother, C., 2003. Linear multi-view reconstruction of points, lines, planes and cameras using a reference plane. In: Proc. International Conference on Computer Vision 2003, Nice, France, pp. 1210–1217.

Schindler, K., 2003. Generalized use of homographies for piecewise planar reconstruction. In: Proc. 13th Scandinavian Conference on Image Analysis (SCIA2003), Springer-Verlag GmbH, Gotenborg, pp. 470–476.

Shi, J. and Tomasi, C., 1994. Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition CVPR, pp. 593–600.

Torr, P. H. S., Fitzgibbon, A. W. and Zisserman, A., 1999. The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. International Journal of Computer Vision 32(1), pp. 27 – 44.

Triggs, B., 1998. Autocalibration from planar scenes. In: Proc. 5th European Conference on Computer Vision, Vol. 1, Springer-Verlag, pp. 89 – 105.

Zelnik-Manor, L. and Irani, M., 1999. Multi-view subspace constraints on homographies. In: Proc. 7th International Conference on Computer Vision, Vol. 2, IEEE Computer Society, Kerkyra, Corfu, Greece, pp. 710–715.

TRINOCULAR RECTIFICATION FOR VARIOUS CAMERA SETUPS

Matthias Heinrichs* and Volker Rodehorst

Computer Vision & Remote Sensing, Berlin University of Technology, Franklinstr. 28/29, FR 3-1,
D-10587 Berlin, Germany – (matzeh, vr)@cs.tu-berlin.de

KEY WORDS: Photogrammetry, Trifocal Geometry, Planar Rectification, Uncalibrated Images, Normal Images

ABSTRACT:

Convergent stereo images can be rectified such that they correspond to the stereo normal case. The important advantage is that the correspondence analysis is simplified to a linear search along one of the image axes. The proposed method describes a rectification method for three uncalibrated cameras, where the configuration is not generally known. The algorithm automatically determines the best positioning of the cameras and corrects mirroring effects to fit the desired camera setup. Since there is no direct solution for the rectification problem for three cameras, the rectification homographies are linearly determined to within six degrees of freedom from three compatible fundamental matrices. The remaining six parameters are then obtained by enforcing additional constraints.

1. INTRODUCTION

The epipolar geometry of a pinhole camera model implies that the correspondent of a given point lies on its epipolar line in the second image of a stereo pair. The pencil of all epipolar lines passes through a point called the epipole **e**, which is the projection of the camera center onto the corresponding image plane. In case of convergent stereo setups, the image matching task is fairly complex and thus inefficient. Rectification determines a transformation of each image plane such that the epipolar lines become parallel to one of the image axes (see Figure 1). This configuration corresponds to the stereo normal case.

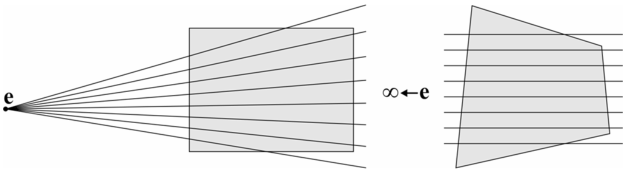


Figure 1. Geometry of normal images

A discussion of *binocular rectification* methods can be found in Hartley (Hartley, 1999), where the normal images are generated using a single linear transformation, which is often referred to as *planar rectification*. In principle, the images are reprojected onto a plane parallel to the baseline between the optical centers. This technique is relatively simple, fast and retains image characteristics, i.e. straight lines. The reported methods minimize image distortion and maximize the computational efficiency. Matoušek (Matoušek et al., 2004) proposed a data-optimal rectification procedure that minimizes the loss of discriminability in the resulting images.

The linear approach however is not general and fails when the epipole lies within the image. The transformation for an epipole close to the image borders leads to an extremely large and strongly distorted image. This problem can be avoided by using stereo configurations with almost parallel camera alignment. For general image sequences with arbitrary camera orientations, an improved method must be used.

In order to solve this problem, Roy (Roy et al., 1997) suggested a *cylindrical rectification* method with a separate transformation for each epipolar line. The basic idea lies in the use of polar coordinates with the epipole at the origin. Pollefeys (Pollefeys et al., 1999) adapted this non-linear approach for applications in projective geometry. However, the use of different non-linear transformations leads to irregular, distorted images, which makes the following correspondence analysis more difficult. To avoid this effect, a *hybrid* procedure was proposed by Oram (Oram, 2001). Here, the epipoles are first overlaid with a compatible homography and after which the same non-linear transformation is used for both images.

It has been shown that multi-view matching can considerably improve the quality of spatial reconstruction, in which rectification remains of interest. In case of a *trinocular rectification*, the images are reprojected onto a plane, which lies parallel to the optical centers (see Figure 2).

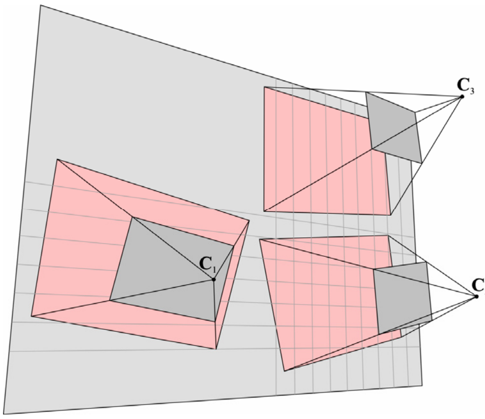


Figure 2. Trinocular Rectification

Ayache and Hansen (Ayache et. al., 1988) proposed a technique for rectifying image triplets that works with calibrated cameras. Loop and Zhang (Loop and Zhang, 1999) presented a stratified method to decompose each transformation and formulate geometric criteria to minimize image distortion during rectification. For a review of various trinocular rectification methods, see (Sun, 2003).

Zhang (Zhang et al., 2002/03) proposed rectification homographies in a closed form and introduced stronger, geometrically meaningful constraints. An (An et al., 2004) reported an efficient trinocular rectification method using the geometric camera model instead of the relative image orientation. However, this method is only applicable when well known control points are available to calibrate and orientate the cameras.

The proposed trinocular rectification method requires an uncalibrated image triple with more or less parallel camera alignment. The camera configuration is arbitrary, but each projection center must be invisible in all other images. This condition is necessary, since otherwise the epipoles lie in the image and mapping them to infinity will lead to unacceptable distortion of the images. Furthermore, we assume non-degenerate camera positions, where the camera centers are not collinear, because collinear setups can be rectified by chaining a classical binocular rectification approach. Additionally, a common overlapping area and at least six homologous image points are necessary, so that the trifocal tensor, the fundamental matrices and the epipoles can be determined (Hartley and Zisserman, 2000). The result consists of three geometrically transformed images, in which the epipolar lines run parallel to the image axes.

2. CAMERA SETUP

A given image triplet consists of the original images *b* (base), *h* (horizontal) and *v* (vertical). Subsequently, we denote the rectified images $\tilde{b}$, $\tilde{h}$ and $\tilde{v}$. The rectification tries to fit any image triple to a configuration shown in Figure 3.

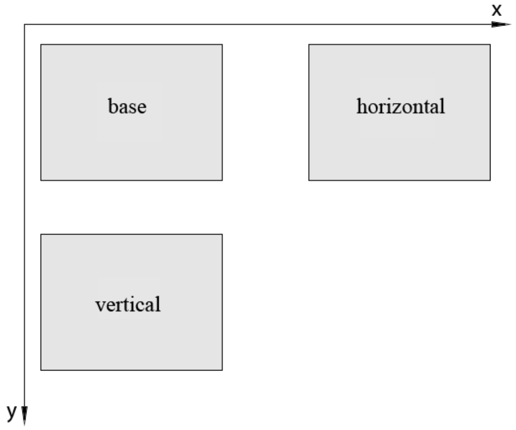


Figure 3. Image arrangement

This setup has the following properties:

- The epipolar lines of image *b* and image *h* correspond with their image rows.
- The epipolar lines of image *b* and image *v* correspond with their image columns.
- The epipolar lines of image *h* and image *v* have a slope of minus unity.

The last property has the advantage, that the disparities between corresponding points in $\tilde{b} \leftrightarrow \tilde{h}$ and $\tilde{b} \leftrightarrow \tilde{v}$ are equal.

For rectification, the epipoles between the images *b*, *h* and *v* should be mapped to infinity:

$$\begin{aligned}\tilde{\mathbf{e}}_{bh} &= \tilde{\mathbf{e}}_{hb} = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^{\mathrm{T}} \\ \tilde{\mathbf{e}}_{bv} &= \tilde{\mathbf{e}}_{vb} = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^{\mathrm{T}} \qquad (1) \\ \tilde{\mathbf{e}}_{hv} &= \tilde{\mathbf{e}}_{vh} = \begin{bmatrix}-1 & 1 & 0\end{bmatrix}^{\mathrm{T}}\end{aligned}$$

The relative image orientation for this setup is quite simple. The fundamental matrices between the rectified images are given by

$$\tilde{\mathbf{F}}_{bh} = \left[\tilde{\mathbf{e}}_{bh}\right]_{\times} = \begin{bmatrix}0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0\end{bmatrix},$$

$$\tilde{\mathbf{F}}_{bv} = \left[\tilde{\mathbf{e}}_{bv}\right]_{\times} = \begin{bmatrix}0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0\end{bmatrix} \text{ and} \qquad (2)$$

$$\tilde{\mathbf{F}}_{hv} = \left[\tilde{\mathbf{e}}_{hv}\right]_{\times} = \begin{bmatrix}0 & 0 & 1 \\ 0 & 0 & 1 \\ -1 & -1 & 0\end{bmatrix}.$$

2.1 Computation of the Projective Image Orientation

This section describes a method to assign each image in a given triplet to a suitable position *b*, *h* and *v*. At least six homologous image points are used to compute the trifocal tensor $\mathcal{T}$ by the minimal 6-point-algorithm (Torr and Zisserman, 1997) with the robust estimator GASAC (Rodehorst and Hellwich, 2006). The projection matrix $\mathbf{P}_1$ of the reference camera is set to a canonical form and the projection matrices $\mathbf{P}_2$ and $\mathbf{P}_3$ for the other two cameras can be obtained from $\mathcal{T}$. The projection center $\mathbf{C}_1 = (C_{11}, C_{12}, C_{13}, C_{14})^{\mathrm{T}}$ of the first camera is placed in the origin and the two remaining projection centers $\mathbf{C}_2$ and $\mathbf{C}_3$ can be estimated from the projective $\mathbf{P}_2$ and $\mathbf{P}_3$.

$\mathbf{P}_1$ is defined to point in z-direction and all cameras have a common overlapping area and are not visible in the other cameras. Therefore $\mathbf{P}_2$ and $\mathbf{P}_3$ can not have a significant translation in z-direction and the camera alignment can be computed in the x/y-plane. Under these conditions the absolute angles β_1, β_2 and β_3 between the cameras in the x/y-plane can be computed:

$$\begin{aligned}\beta_1 &= \begin{cases}\left|\arctan\left(C_{22}/C_{21}\right)\right| & \text{for } C_{21} \neq 0 \\ \pi/2 & \text{otherwise}\end{cases} \\ \beta_2 &= \begin{cases}\left|\arctan\left(C_{32}/C_{31}\right)\right| & \text{for } C_{31} \neq 0 \\ \pi/2 & \text{otherwise}\end{cases} \qquad (3) \\ \beta_3 &= \begin{cases}\left|\arctan\left(\left(C_{32}-C_{22}\right)/\left(C_{31}-C_{21}\right)\right)\right| & \text{for}\left(C_{31}-C_{21}\right) \neq 0 \\ \pi/2 & \text{otherwise}\end{cases}\end{aligned}$$

Using the absolute value is necessary, since camera configurations, which are different from the setup assumed in Figure 3 produce mirror effects. The compensation of these effects is discussed in section 3.3.3. Figure 3 shows that the camera pair with the highest angle value is the pair $b \leftrightarrow v$ and the camera pair with the lowest angle value is the pair $b \leftrightarrow h$. Now the images can be aligned in a suitable fashion and the trifocal tensor $\mathcal{T}$ must be adapted for this enhanced setup.

3. RECTIFICATION

The initial task is to determine the relative image orientation. The fundamental matrices of the original images can be obtained uniquely by the trifocal tensor (see section 2.1). Note that the fundamental matrices are not independent and have only 18 significant parameters in total (Hartley and Zisserman, 2000):

$$\mathbf{e}_{hv}^{\mathrm{T}}\mathbf{F}_{hb}\mathbf{e}_{bv} = \mathbf{e}_{vb}^{\mathrm{T}}\mathbf{F}_{vh}\mathbf{e}_{hb} = \mathbf{e}_{vh}^{\mathrm{T}}\mathbf{F}_{vb}\mathbf{e}_{bh} = 0$$

3.1 Mapping Epipoles to Infinity

Let $\mathbf{H}_b$, $\mathbf{H}_h$ and $\mathbf{H}_v$ be the unknown homographies between the original and rectified images. The rows of these homographies will be abbreviated by three vectors $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$:

$$\mathbf{H}_i = \begin{bmatrix} \mathbf{u}_i^{\mathrm{T}} \\ \mathbf{v}_i^{\mathrm{T}} \\ \mathbf{w}_i^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} u_{i1} & u_{i2} & u_{i3} \\ v_{i1} & v_{i2} & v_{i3} \\ w_{i1} & w_{i2} & w_{i3} \end{bmatrix} \quad \text{for} \quad i \in \{b,h,v\} \tag{4}$$

For a correspondence $\mathbf{x}_b \leftrightarrow \mathbf{x}_h \leftrightarrow \mathbf{x}_v$, the projective transformation between the image coordinates can be written as

$$\tilde{\mathbf{x}}_i = \mathbf{H}_i \mathbf{x}_i \quad \text{for} \quad i \in \{b,h,v\}. \tag{5}$$

The fundamental matrices, which are calculated from the original images, satisfy the epipolar constraints:

$$\begin{aligned} \mathbf{x}_h^{\mathrm{T}} \mathbf{F}_{bh} \mathbf{x}_b &= 0 \\ \mathbf{x}_v^{\mathrm{T}} \mathbf{F}_{bv} \mathbf{x}_b &= 0 \\ \mathbf{x}_v^{\mathrm{T}} \mathbf{F}_{hv} \mathbf{x}_h &= 0 \end{aligned} \tag{6}$$

Similar conditions apply for the rectified images:

$$\begin{aligned} \tilde{\mathbf{x}}_h^{\mathrm{T}} \tilde{\mathbf{F}}_{bh} \tilde{\mathbf{x}}_b &= 0 \\ \tilde{\mathbf{x}}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{bv} \tilde{\mathbf{x}}_b &= 0 \\ \tilde{\mathbf{x}}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{hv} \tilde{\mathbf{x}}_h &= 0 \end{aligned} \tag{7}$$

Combining equations (6) and (7), one obtains

$$\begin{aligned} \mathbf{x}_h^{\mathrm{T}} \mathbf{H}_h^{\mathrm{T}} \tilde{\mathbf{F}}_{bh} \mathbf{H}_b \mathbf{x}_b &= \mathbf{x}_h^{\mathrm{T}} \mathbf{F}_{bh} \mathbf{x}_b = 0 \\ \mathbf{x}_v^{\mathrm{T}} \mathbf{H}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{bv} \mathbf{H}_b \mathbf{x}_b &= \mathbf{x}_v^{\mathrm{T}} \mathbf{F}_{bv} \mathbf{x}_b = 0 \\ \mathbf{x}_v^{\mathrm{T}} \mathbf{H}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{hv} \mathbf{H}_h \mathbf{x}_h &= \mathbf{x}_v^{\mathrm{T}} \mathbf{F}_{hv} \mathbf{x}_h = 0 \end{aligned} \tag{8}$$

and comparing the result with (7), it follows that

$$\begin{aligned} \mathbf{H}_h^{\mathrm{T}} \tilde{\mathbf{F}}_{bh} \mathbf{H}_b &= \lambda_1 \mathbf{F}_{bh} \\ \mathbf{H}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{bv} \mathbf{H}_b &= \lambda_2 \mathbf{F}_{bv} \\ \mathbf{H}_v^{\mathrm{T}} \tilde{\mathbf{F}}_{hv} \mathbf{H}_h &= \lambda_3 \mathbf{F}_{hv} \end{aligned} \tag{9}$$

where λ_i are scale factors. The rectified fundamental matrices $\tilde{\mathbf{F}}_{bh}$, $\tilde{\mathbf{F}}_{bv}$ and $\tilde{\mathbf{F}}_{hv}$ contain many zeros (see Eq. 2). Hence, equations (9) can be simplified to give:

$$\begin{aligned} \mathbf{w}_h \mathbf{v}_b{}^{\mathrm{T}} - \mathbf{v}_h \mathbf{w}_b{}^{\mathrm{T}} &= \lambda_1 \mathbf{F}_{bh} \\ \mathbf{u}_v \mathbf{w}_b{}^{\mathrm{T}} - \mathbf{w}_v \mathbf{u}_b{}^{\mathrm{T}} &= \lambda_2 \mathbf{F}_{bv} \\ (\mathbf{u}_v + \mathbf{v}_v) \mathbf{w}_h{}^{\mathrm{T}} - \mathbf{w}_v (\mathbf{u}_h + \mathbf{v}_h)^{\mathrm{T}} &= \lambda_3 \mathbf{F}_{hv} \end{aligned} \tag{10}$$

3.2 Computation of Rectifying Homographies

Since the fundamental matrices are of rank 2, the equations (10) can not be solved directly. However, knowing the vectors $\mathbf{w}_b$, $\mathbf{w}_h$ and $\mathbf{w}_v$, gives a solution for (10) with six degrees of freedom (DOF). The **w**-vectors have a convenient property: Since the epipoles in the original images should be mapped to infinity, the scalar product of $\mathbf{w}_i$ with both epipoles of an image is zero. That means the **w**-vector is perpendicular to both epipoles of an image and can be calculated (up to a scale factor) by the cross product of the two epipoles:

$$\begin{aligned} \mathbf{w}_b &= \mathbf{e}_{bh} \times \mathbf{e}_{bv} \\ \mathbf{w}_h &= \mathbf{e}_{hb} \times \mathbf{e}_{hv} \\ \mathbf{w}_v &= \mathbf{e}_{vb} \times \mathbf{e}_{vh} \end{aligned} \tag{11}$$

The epipoles are the left and right null-vectors of the **F**-matrices and can be determined by singular value decomposition (Hartley and Zisserman, 2000). Since the projection centers are not collinear, all epipole pairs are linearly independent. Hence their cross products, especially the third components, are non zero and the third component of each **w**-vector can be scaled to unity to simplify the equations (10).

Six variables have to be defined for a direct solution. Depending on the variables chosen, the equations become very simple. We recommend setting

$$u_{b3} = v_{h3} = v_{v3} = 0 \quad \text{and} \quad \lambda_1 = \lambda_2 = \lambda_3 = 1 \tag{12}$$

which yields the solutions

$$\mathbf{H}_b^* = \begin{bmatrix} w_{b1} F_{33}^{bv} - F_{31}^{bv} & w_{b2} F_{33}^{bv} - F_{32}^{bv} & 0 \\ F_{31}^{bh} & F_{32}^{bh} & F_{33}^{bh} \\ w_{b1} & w_{b2} & 1 \end{bmatrix} \tag{13}$$

$$\mathbf{H}_h^* = \begin{bmatrix} w_{h1}(F_{33}^{bv} - F_{33}^{bh}) + F_{13}^{bh} - F_{31}^{hv} & w_{h2}(F_{33}^{bv} - F_{33}^{bh}) + F_{23}^{bh} - F_{32}^{hv} & F_{33}^{bv} - F_{33}^{hv} \\ w_{h1} F_{33}^{bh} - F_{13}^{bh} & w_{h2} F_{33}^{bh} - F_{23}^{bh} & 0 \\ w_{h1} & w_{h2} & 1 \end{bmatrix}$$

$$\mathbf{H}_v^* = \begin{bmatrix} F_{13}^{bv} & F_{23}^{bv} & F_{33}^{bv} \\ w_{v1}(F_{33}^{bv} - F_{33}^{hv}) + F_{13}^{hv} - F_{13}^{bv} & w_{v2}(F_{33}^{bv} - F_{33}^{hv}) + F_{23}^{hv} - F_{23}^{bv} & 0 \\ w_{v1} & w_{v2} & 1 \end{bmatrix}.$$

They look similar to those proposed by Zhang (Zhang et al., 2002), but satisfy the common image representation with the origin in the upper left corner. These equations already describe primitive rectifying homographies for the given image triplet, but generally produce undesirable shearing and scaling. A detailed derivation of (13) from (10) is available online (Heinrichs and Rodehorst, 2006).

3.3 Imposing Geometric Constraints

The assumptions in equation (12) can be generalized once more by comparing equations (13) and (10). The missing variables will be split into different components with the geometric meaning of translation, mirroring, scaling and shearing. The mirroring component is necessary because the computation of the camera setup ignores mirror effects in (3). These two parameters are introduced for convenience, to maintain the order of the image content, but do not influence the correctness of the rectification. To clarify the meaning of the remaining DOF, we define:

$$\begin{aligned} &s_1 = u_{b3}, \quad s_2 = v_{h3}, \quad s_3 = v_{v3} - v_{h3} \quad \text{and} \\ &\alpha_1 = \lambda_1 / \lambda_3, \quad \alpha_2 = \lambda_2 / \lambda_3, \quad \alpha_3 = \lambda_3 \end{aligned} \tag{14}$$

The mirroring components can be written as follows:

$$m_x, m_y \in \{-1, 1\} \tag{15}$$

The choice of correct signs allows a better visual interpretation. Mirror compensation is necessary for camera setups, in which the original images are flipped over one or two axes: *v* below *b* or *h* left of *b*. If m_x and m_y have different signs the slope of the epipolar lines between $\tilde{h}$ and $\tilde{v}$ becomes positive.

The general solution for the homographies is given by:

$$\mathbf{H}_b = \begin{bmatrix} 1 & 0 & s_1 \\ 0 & 1 & s_2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_b^* \tag{16}$$

$$\mathbf{H}_h = \begin{bmatrix} 1 & 0 & s_1 + s_3 \\ 0 & 1 & s_2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1-\alpha_1 & F_{33}^{bv}(\alpha_2 - 1) \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_h^*$$

$$\mathbf{H}_v = \begin{bmatrix} 1 & 0 & s_1 \\ 0 & 1 & s_2 + s_3 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha_2 & 0 & 0 \\ 1-\alpha_2 & 1 & F_{33}^{bv}(\alpha_2 - 1) \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_v^*$$

where the free parameters can be interpreted as:

- s_1 is the global shift value in x-direction of all images
- s_2 defines the global shift value in y-direction
- s_3 is the shift value in x-direction for image h and the shift value in y-direction of image v relative to image b
- α_1 is the scale of the y-component of images b and h, which affects the shearing in y-direction of image v
- α_2 is the scale of the x-component of images b and v, which affects the shearing in x-direction of image h
- α_3 defines the global scaling factor to keep the images at a suitable resolution

The two convenience parameters are:

- m_x is a mirroring factor in the x-direction of all images
- m_y defines the mirroring factor in the y-direction

The challenge is to estimate optimal values for these parameters. Since the first six parameters are independent of each other, one can deal with them separately. The shift parameters depend on the mirroring parameters, thus we have to correct the mirror parameters first. The values should be calculated in the following order:

1. Finding proper shearing values for α_1 and α_2
2. Finding a global scale value α_3
3. Compensate potential mirroring using m_x and m_y
4. Finding right offset values for s_1, s_2 and s_3

The factor $F_{33}^{bv}(\alpha_2 - 1)$ in the shearing matrices of equation (16) of image h and v is needed to compensate the loss of information in (10) by applying (12).

3.3.1 Shearing Correction

Following the approach of Loop and Zhang (Loop and Zhang, 1999), the shearing of images h and v can be minimized by keeping two perpendicular vectors in the middle of the original image perpendicular in the rectified one. Let W be the width and H be the height of the original images. Two perpendicular direction vectors $\mathbf{x}$ and $\mathbf{y}$ are defined by computing the center lines of the image.

$$\mathbf{x} = \mathbf{a} - \mathbf{b}, \quad \mathbf{y} = \mathbf{c} - \mathbf{d} \quad \text{with}$$

$$\mathbf{a} = \begin{bmatrix} W/2 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} W/2 \\ H \\ 1 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 0 \\ H/2 \\ 1 \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} W \\ H/2 \\ 1 \end{bmatrix} \tag{17}$$

For the horizontal rectified image $\tilde{h}$ the vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ can be calculated as follows:

$$\begin{gathered} \tilde{\mathbf{x}} = \tilde{\mathbf{a}} - \tilde{\mathbf{b}}, \quad \tilde{\mathbf{y}} = \tilde{\mathbf{c}} - \tilde{\mathbf{d}} \\ \tilde{\mathbf{a}} = \mathbf{H}_h^* \cdot \mathbf{a}, \quad \tilde{\mathbf{b}} = \mathbf{H}_h^* \cdot \mathbf{b}, \quad \tilde{\mathbf{c}} = \mathbf{H}_h^* \cdot \mathbf{c}, \quad \tilde{\mathbf{d}} = \mathbf{H}_h^* \cdot \mathbf{d} \end{gathered} \tag{18}$$

We apply the shearing matrix to $\mathbf{H}_h$, which can be derived from (16):

$$\mathbf{S}_h = \begin{bmatrix} 1 & 1-\alpha_1 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{19}$$

Note that the factor $F_{33}^{bv}(\alpha_2 - 1)$ in (16) is ignored, since the last (homogeneous) component of the vectors $\tilde{\mathbf{a}}$, $\tilde{\mathbf{b}}$, $\tilde{\mathbf{c}}$ and $\tilde{\mathbf{d}}$ are the same and therefore their difference equals zero.
The vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ are perpendicular, when the constraint

$$(\mathbf{S}_h \tilde{\mathbf{x}})^{\mathrm{T}} (\mathbf{S}_h \tilde{\mathbf{y}}) = 0 \tag{20}$$

is satisfied. The quadratic equation

$$\alpha_1^2 + \alpha_1 \left(\frac{-\tilde{x}_x \tilde{y}_y - \tilde{x}_y \tilde{y}_x - 2\tilde{x}_y \tilde{y}_y}{2\tilde{x}_y \tilde{y}_y} \right) + \left(\frac{\tilde{x}_x \tilde{y}_x + \tilde{x}_x \tilde{y}_y + \tilde{x}_y \tilde{y}_x + \tilde{x}_y \tilde{y}_y}{2\tilde{x}_y \tilde{y}_y} \right) = 0 \tag{21}$$

has two solutions, where the solution with the smaller absolute value $|\alpha_1|$ is to be preferred.
For the vertical rectified image $\tilde{v}$, the vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ can be calculated as follows:

$$\begin{gathered} \tilde{\mathbf{x}} = \tilde{\mathbf{a}} - \tilde{\mathbf{b}}, \quad \tilde{\mathbf{y}} = \tilde{\mathbf{c}} - \tilde{\mathbf{d}} \\ \tilde{\mathbf{a}} = \mathbf{H}_v^* \cdot \mathbf{a}, \quad \tilde{\mathbf{b}} = \mathbf{H}_v^* \cdot \mathbf{b}, \quad \tilde{\mathbf{c}} = \mathbf{H}_v^* \cdot \mathbf{c}, \quad \tilde{\mathbf{d}} = \mathbf{H}_v^* \cdot \mathbf{d} \end{gathered} \tag{22}$$

The shearing matrix can be written as

$$\mathbf{S}_v = \begin{bmatrix} \alpha_2 & 0 & 0 \\ 1-\alpha_2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{23}$$

Once again, the factor $F_{33}^{bv}(\alpha_2 - 1)$ can be ignored, because the third component of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is zero. The perpendicularity condition results in

$$(\mathbf{S}_v \tilde{\mathbf{x}})^{\mathrm{T}} (\mathbf{S}_v \tilde{\mathbf{y}}) = 0 \cdot \tag{24}$$

Again, after solving the quadratic equation

$$\alpha_2^2 + \alpha_2 \left(\frac{-\tilde{x}_x \tilde{y}_y - \tilde{x}_y \tilde{y}_x - 2\tilde{x}_x \tilde{y}_x}{2\tilde{x}_x \tilde{y}_x} \right) + \left(\frac{\tilde{x}_x \tilde{y}_x + \tilde{x}_x \tilde{y}_y + \tilde{x}_y \tilde{y}_x + \tilde{x}_y \tilde{y}_y}{2\tilde{x}_x \tilde{y}_x} \right) = 0 \tag{25}$$

the result with the smaller absolute value $|\alpha_2|$ is selected.

3.3.2 Scale Correction

Once the shearing parameters have been obtained, the global scale can be chosen. To preserve as much information as possible, the number of pixels in image b and $\tilde{b}$ should be equal. The resolutions can be estimated from the length of the diagonal line through b and its projection in $\tilde{b}$. Using

$$\begin{gathered} \mathbf{a} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}}, \quad \mathbf{b} = \begin{bmatrix} W & H & 1 \end{bmatrix}^{\mathrm{T}} \\ \tilde{\mathbf{a}} = \begin{bmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_b^* \cdot \mathbf{a}, \quad \tilde{\mathbf{b}} = \begin{bmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_b^* \cdot \mathbf{b} \end{gathered} \tag{26}$$

one obtains the common scaling factor:

$$\alpha_3 = \frac{|\tilde{\mathbf{b}} - \tilde{\mathbf{a}}|}{|\mathbf{b} - \mathbf{a}|} \tag{27}$$

3.3.3 Mirroring Correction

To correct potential mirror effects, the order of the point correspondences are examined. First, the four corresponding triplets with the smallest and highest x- and y-value in image b are selected, to avoid flips of points due to perspective projection. If the order of the transformed values switches in one dimension for all three images, the rectified images have to be mirrored in that direction by setting m_x or m_y to -1.

3.3.4 Offset Estimation

The offsets s_1, s_2 and s_3 depend on the origin of the coordinate system. The estimate for s_1 is calculated from the rectification of the origins in image b and v. The parameter s_2 can be determined from the origin in image b and h. First, the homographies with shearing-, mirroring- and scaling correction are applied to the origins:

$$\mathbf{x}_b^* = \begin{bmatrix} x_{b1}^* \\ x_{b2}^* \\ x_{b3}^* \end{bmatrix} = \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_b^* \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{x}_h^* = \begin{bmatrix} x_{h1}^* \\ x_{h2}^* \\ x_{h3}^* \end{bmatrix} = \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1-\alpha_1 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_h^* \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad (28)$$

$$\mathbf{x}_v^* = \begin{bmatrix} x_{v1}^* \\ x_{v2}^* \\ x_{v3}^* \end{bmatrix} = \begin{bmatrix} m_x\alpha_3 & 0 & 0 \\ 0 & m_y\alpha_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha_2 & 0 & 0 \\ 1-\alpha_2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{H}_v^* \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Now, the horizontal and vertical offsets are defined by the negation of the minimal coordinates:

$$s_1 = -\min(x_{b1}^*, x_{v1}^*)$$
$$s_2 = -\min(x_{b2}^*, x_{h2}^*) \qquad (29)$$

The remaining parameter s_3 is calculated from the origin of image h and v. Since the parameters s_1 and s_2 are already known, we can use

$$\mathbf{y}_h^* = \begin{bmatrix} 1 & 0 & s_1 \\ 0 & 1 & s_2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{h1}^* \\ x_{h2}^* \\ x_{h3}^* \end{bmatrix}$$
$$\mathbf{y}_v^* = \begin{bmatrix} 1 & 0 & s_1 \\ 0 & 1 & s_2 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{v1}^* \\ x_{v2}^* \\ x_{v3}^* \end{bmatrix} \qquad (30)$$

to determine the missing parameter

$$s_3 = -\min(y_{v2}^*, y_{h1}^*)\,. \qquad (31)$$

3.3.5 Finding the Common Region

Finally, the computation of the common image regions minimizes the image sizes and speed up image matching. Therefore, we cut off regions, which are not visible in all three images. The required clipping lines are shown in Figure 4.

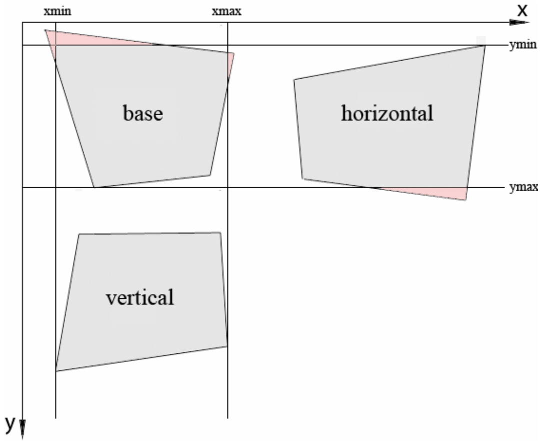


Figure 4. Definition of the common region

The vertical clipping line $\mathbf{x}_{min}$ can be derived from the corners of the images b and v using

$$\mathbf{a} = \mathbf{H}_b \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{b} = \mathbf{H}_b \cdot \begin{bmatrix} 0 \\ H \\ 1 \end{bmatrix}, \quad \mathbf{c} = \mathbf{H}_v \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{d} = \mathbf{H}_v \cdot \begin{bmatrix} 0 \\ H \\ 1 \end{bmatrix} \qquad (32)$$
$$\mathbf{x}_{\min} = \left[1 \quad 0 \quad -\max(\min(\mathbf{a},\mathbf{b}), \min(\mathbf{c},\mathbf{d}))\right]^{\mathrm{T}}$$

The line $\mathbf{x}_{max}$ is given by

$$\mathbf{a} = \mathbf{H}_b \cdot \begin{bmatrix} W \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{b} = \mathbf{H}_b \cdot \begin{bmatrix} W \\ H \\ 1 \end{bmatrix}, \quad \mathbf{c} = \mathbf{H}_v \cdot \begin{bmatrix} W \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{d} = \mathbf{H}_v \cdot \begin{bmatrix} W \\ H \\ 1 \end{bmatrix} \qquad (33)$$
$$\mathbf{x}_{\max} = \left[1 \quad 0 \quad -\min(\max(\mathbf{a},\mathbf{b}), \max(\mathbf{c},\mathbf{d}))\right]^{\mathrm{T}}$$

The horizontal clipping lines $\mathbf{y}_{min}$ and $\mathbf{y}_{max}$ are determined from the corners of the images b and h in an analogous manner.

3.4 Applying the Rectifying Transformation

To achieve optimal accuracy, the images b, h and v are resampled only once, using equation (16), by the indirect method. For every integer position in the rectified image, the non-integer position in the original image is determined using the inverse projective transformation $\mathbf{H}^{-1}$. The sub-pixel intensities are then computed by bicubic interpolation, which determines the intensity value from 16 pixels in a 4×4 neighborhood (see Figure 5). The computation

$$f(i+dx, j+dy) = \sum_{m=-1}^{2}\sum_{n=-1}^{2} f(i+m, j+n)\cdot r(m-dx)\cdot r(dy-n) \qquad (34)$$

using the cubic weighting function

$$r(k) = \frac{1}{6}\left[p(k+2)^3 - 4p(k+1)^3 + 6p(k)^3 - 4p(k-1)^3\right]$$
$$\text{with} \quad p(k) = \begin{cases} k & \text{for } k > 0 \\ 0 & \text{otherwise} \end{cases} \qquad (35)$$

defines the interpolated intensity value and must be computed separately for each color channel.

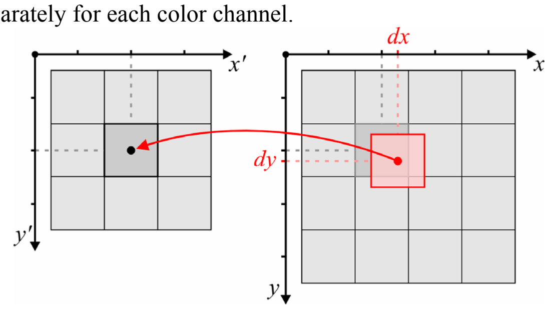


Figure 5. Resampling using the indirect transformation

4. EXPERIMENTAL RESULTS

The proposed method is verified by rectifying some real images of a statue, which were acquired by hand with an uncalibrated digital camera (see Figure 6). The original image size is 1024x768 pixels. For each image, 16 corresponding points were measured and the trifocal tensor was computed. The results are illustrated in Figure 7. To verify the correction of mirroring effects and the image alignment, several image triples were permutated and passed to the algorithm. The images were always positioned reasonably and no global mirror artefacts were observed.

The mean error and variance of the epipolar distance in the rectified images were computed for each image pair separately and for all three images. The results are shown in Table 1.

Distance	b↔h	b↔v	h↔v	Total
Mean error	0.378	0.285	0.573	0.398
Variance	0.374	0.163	0.688	0.393

Table 1: Epipolar distance of the rectified images [in pixels]

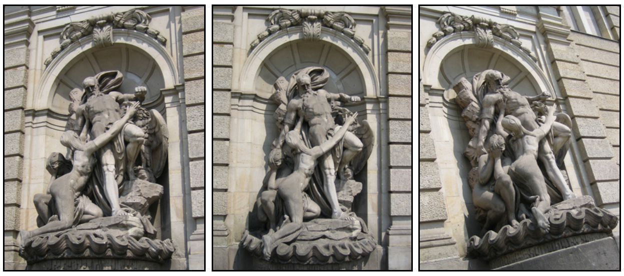


Figure 6. Image triplet before rectification

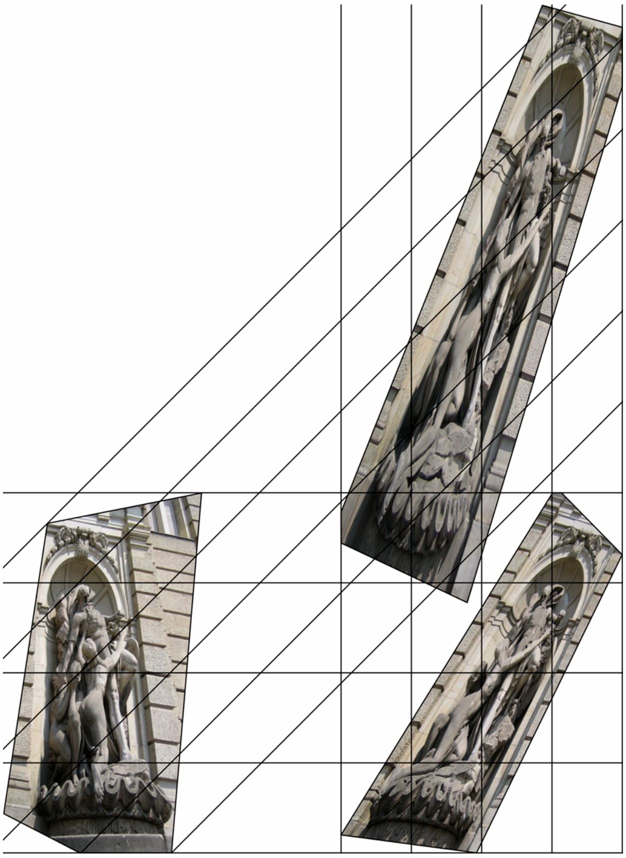


Figure 7. Image triplet after rectification

5. CONCLUSIONS

In this paper, a linear method for trinocular rectification of uncalibrated images, in closed form with 6 degrees of freedom, was proposed. In a post processing stage, proper geometric constraints are selected to minimize the projective distortion.

The proposed mirror correction eases the interpretation of the rectified images and makes it possible to apply this approach to various camera setups. Furthermore, the automated image alignment allows more convenient image acquisition, because the images can be shot without minding the relative image order. Finally, the computation of the common image regions minimizes the image sizes and speeds up image matching. However, the quality of the rectification depends on the robust estimation of the fundamental matrix. Therefore, the correspondence sets should be carefully chosen and well distributed over the scene setup.

6. ACKNOWLEDGEMENTS

This work was partially supported by grants from the German Research Foundation DFG.

7. REFERENCES

An, L., Jia, Y., Wang, J., Zhang, X. and Li, M., 2004: An efficient rectification method for trinocular stereovision, *Proc. Int. Conf. on Pattern Recognition*, vol. 4, Cambridge, pp. 56-59.

Ayache, N. and Hansen, C., 1988: Rectification of Images for Binocular and Trinocular Stereovision, *Proc. 9th International Conference on Pattern Recognition*, Rome, pp. 11-16.

Hartley, R.I., 1999: Theory and practice of projective rectification, *Int. Journal of Computer Vision*, vol. 35, no. 2, pp. 115-127.

Hartley, R.I. and Zisserman, A., 2000: *Multiple view geometry in computer vision*, Cambridge University Press, 607 p.

Heinrichs, M. and Rodehorst, V., 2006: Technical Report on Trinocular Rectification, Berlin University of Technology, http://www.cv.tu-berlin.de/publications/pdf/RectifyTR06.pdf

Loop, C. and Zhang, Z., 1999: Computing rectifying homographies for stereo vision, *Proc. Computer Vision and Pattern Recognition*, vol. 1, Fort Collins, pp. 125–131.

Matoušek, M., Šára, R. and Hlaváč, V., 2004: Data-optimal rectification for fast and accurate stereovision, *Proc. 3rd Int. Conf. on Image and Graphics*, Hong Kong, pp. 212-215.

Oram, D.T., 2001: Rectification for any epipolar geometry, *Proc. British Machine Vision Conf.*, London, pp. 653-662.

Rodehorst, V. and Hellwich, O., 2006: Genetic Algorithm SAmple Consensus (GASAC) - A Parallel Strategy for Robust Parameter Estimation, Int. Workshop "25 Years of RANSAC" in conjunction with CVPR '06, IEEE Computer Society, New York, 8 p.

Pollefeys, M., Koch, R. and Van Gool, L., 1999: A simple and efficient rectification method for general motion, *Proc. 7th Int. Conf. on Computer Vision*, vol. 1, Copenhagen, pp. 496–501.

Roy, S., Meunier, J. and Cox, I., 1997: Cylindrical rectification to minimize epipolar distortion, *Proc. Computer Vision and Pattern Recognition*, San Juan, pp. 393–399.

Sun, C., 2003: Uncalibrated Three-View Image Rectification, *Image and Vision Computing*, vol. 21, no. 3, pp. 259-269.

Torr, P.H.S. and Zisserman, A., 1997: Robust parameterization and computation of the trifocal tensor, *Image and Vision Computing*, vol. 15, no. 8, pp. 591-605

Zhang, H. and Šára, R. 2002: A linear method for trinocular rectification, Research Report No. 9, Center for Machine Perception, Czech Technical University Prague, 12 p.

Zhang, H., Čech, J., Šára, R., Wu, F. and Hu, Z., 2003: A Linear Method for Trinocular Rectification, *Proc. British Machine Vision Conference*, vol. 1, London, pp. 281-290.

DETECTORS AND DESCRIPTORS FOR PHOTOGRAMMETRIC APPLICATIONS

Fabio Remondino

Institute for Geodesy and Photogrammetry, ETH Zurich, Switzerland
E-mail: fabio@geod.baug.ethz.ch
http://www.photogrammetry.ethz.ch

Commission III

KEY WORDS: Features Detection, Orientation, Precision

ABSTRACT

This paper reports about interest operators, region detectors and region descriptors for photogrammetric applications. Features are the primary input for many applications like registration, 3D reconstruction, motion tracking, robot navigation, etc. Nowadays many detectors and descriptors algorithms are available, providing corners, edges and regions of interest together with n-dimensional vectors useful in matching procedures. The main algorithms are here described and analyzed, together with their proprieties. Experiments concerning the repeatability, localization accuracy and quantitative analysis are performed and reported. Details on how improve to location accuracy of region detectors are also reported.

1. INTRODUCTION

Many photogrammetric and computer vision tasks rely on features extraction as primary input for further processing and analysis. Features are mainly used for images registration, 3D reconstruction, motion tracking, robot navigation, object detection and recognition, etc. Markerless automated orientation procedures based on image features assume the camera (images) to be in any possible orientation: therefore the features should be invariant under different transformations to be re-detectable and useful in the automated matching procedures.

[Haralick and Shapiro, 1992] report these characteristics for a distinctive matching feature: distinctness (clearly distinguished from the background), invariance (independent from radiometric and geometric distortions), interpretability (the associated interest values should have a meaning and possibly usable for further operations), stability (robustness against image noise) and uniqueness (distinguishable from other points).

We should primarily distinguish between feature detectors and descriptors. *Detectors* are operators which search 2D locations in the images (i.e. a point or a region) geometrically stable under different transformations and containing high information content. The results are generally called ‘interest points’ or ‘corners’ or ‘affine regions’ or ‘invariant regions’. *Descriptors* instead analyze the image providing, for certain positions (e.g. an interest point), a 2D vector of pixel information. This information can be used to classify the extracted points or in a matching process.

In photogrammetry, interest points are mainly employed for image orientation or 3D reconstruction applications. In vision applications, regions have been recently also employed, for object detection, recognition and categorization as well as automated wide-baseline image orientation.

In the literature different detectors and descriptors have been presented. The achieved results vary, according to the used images and parameters, therefore assesses of the performances are required. Previous works comparing feature point detectors have been reported in [Schmid et al., 1998; Zuliani et al., 2004; Rodehorst and Koschan, 2006]. [Mikolajczyk et al., 2005] compared affine regions detectors while [Mikolajczyk & Schmid, 2003] reported about local descriptors evaluation.

Usually different measures and criterion are used to assess performance evaluations of interest points or regions detectors: for example, given a ground-truth, the geometrical stability of the detected interest points is compared between different images of a given (planar) scene taken under varying viewing conditions.

Selecting the best procedure to compare the operators is very difficult. In our work, the evaluation is performed calculating the number of correct points detected, their correct localization, the density and analyzing the relative orientation results between stereo-pairs. In all the experiments, the results are checked by visual inspection and statistical evaluations. No comparison of the detection speed is performed as difficult to achieve and as the efficiency of a detector (or descriptor) strongly depends on its implementation.

In the context of this work, we only consider points and regions, excluding edges. An overview and comparison of edge detectors is presented in [Heath et al., 1997; Ziou & Tabbone, 1998].

2. POINT AND REGION DETECTORS

2.1 Point detectors

Many interest point detectors exist in the literature and they are generally divided in contour based methods, signal based methods and methods based on template fitting. Contour based detectors search for maximal curvature or inflexion points along the contour chains. Signal based detectors analyze the image signal and derive a measure which indicates the presence of an interest point. Methods based on template fitting try to fit the image signal to a parametric model of a specific type of interest point (e.g. a corner). The main properties of a point detector are: (1) accuracy, i.e. the ability to detect a pattern at its correct pixel location; (2) stability, i.e. the ability to detect the same feature after that the image undergoes some geometrical transformation (e.g. rotation or scale), or illumination changes; (3) sensitivity, i.e. the ability to detect feature points in low contrast conditions; (4) controllability and speed, i.e. the number of parameters controlling the operator and the time required to identify features.

Among the different interest point detectors presented in the literature, the most used operators are afterwards shortly described:

- Hessian detector [Beaudet, 1978]: it calculates the corner strength as the determinant of the Hessian matrix ($I_{xx}I_{yy}$-I^2_{xy}). The local maxima of the corner strength denote the corners in the image. The determinant is related to the Gaussian curvature of the signal and this measure is invariant to rotation. An extended version, called Hessian-Laplace [Mikolajczyk & Schmid, 2004] detects points which are invariant to rotation and scale (local maxima of the Laplacian-of-Gaussian).

- Moravec detector [Moravec, 1979]: it computes an un-normalized local autocorrelation function of the image in four directions and takes the lowest result as the measure of interest. Therefore it detects point where there are large intensity variations in every direction. Moravec was the first one to introduce the idea of 'point of interest'.
- Förstner detector [Förstner, W. & Guelch, E., 1987]: it uses also the auto-correlation function to classify the pixels into categories (interest points, edges or region); the detection and localization stages are separated, into the selection of windows, in which features are known to reside, and feature location within selected windows. Further statistics performed locally allow estimating automatically the thresholds for the classification. The algorithm requires a complicate implementation and is generally slower compared to other detectors.
- Harris detector [Harris & Stephens, 1988]: similar to [Moravec, 1979], it computes a matrix related to the auto-correlation function of the image. The squared first derivatives of the image signal are averaged over a window and the eigenvalues of the resulting matrix are the principal curvatures of the auto-correlation function. An interest point is detected if the found two curvatures are high. Harris points are invariant to rotation. Extended versions of the Harris detector have been presented in [Mikolajczyk & Schmid, 2001; Brown et al., 2005] where the detected points are invariant to scale and rotation.
- Tomasi and Kanade detector [Tomasi & Kanade, 1991]: they developed a features tracker based on a previous work of [Lucas & Kanade, 1981]. Defining a good feature 'the one that can be tracked well', a feature is detected if the two eigenvalues of an image patch are smaller that an empirically computed threshold.
- Haralick operator [Haralick & Shapiro, 1992]: it first extracts windows of interest from the image and then computes the precise position of the point of interest inside the selected windows. The windows of interest are computed with a gradient operator and the normal matrix; the point of interest is determined as the weighted centre of gravity of all points inside the window.
- Heitger detector [Heitger et al., 1992]: derived from biological visual system experiments, it uses Gabor filters to derive 1D directional characteristic in different directions. Afterwards the first and second derivatives are computed and combined to get 2D interest locations (called 'keypoints'). It requires a lot of CPU processing.
- Susan detector [Smith & Brady, 1997]: it analyzes different regions separately, using direct local measurements and finding places where individual region boundaries have high curvature. The brightness of each pixel in a circular mask is compared to the central pixel to define an area that has a similar brightness to the centre. Computing the size, centroid and second moment of this area, 2D interest features are detected.

2.2 Region detectors

The detection of image regions invariant under certain transformations has received great interest, in particular in the vision community. The main requirements are that the detected regions should have a shape which is function of the image transformation and automatically adapted to cover always the same object surface. Under a generic camera movement (e.g. translation), the most common transformation is an affinity, but also scale-invariant detectors have been developed. Generally an interest point detector is used to localize the points and afterwards an elliptical invariant region is extracted around each point.

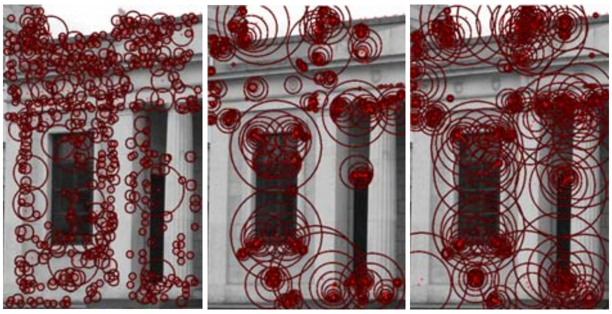


Figure 1: Scale-invariant regions extracted with DoG detector (left) [Lowe, 2004] and affine-invariant regions extracted with Harris-affine (center) and Hessian-affine detector (right) [Mikolajczyk and Schmid, 2002].

Methods for detecting *scale-invariant regions* were presented in [Lindeberg, 1998; Kadir & Brady, 2001; Jurie & Schmid, 2004; Lowe, 2004; Leibe & Schiele, 2004]. Generally these techniques assume that the scale change is constant in every direction and search for local extrema in the 3D scale-space representation of an image (x, y and scale). In particular, the DoG (Difference of Gaussian) detector [Lowe, 2004] showed high repeatability under different tests: it selects blob-like structures by searching for scale-space maxima of a DoG (FIG). On the other hand, *affine-invariant region* detector can be seen as a generalization of the scale-invariant detector, because with an affinity, the scale can be different in each direction. Therefore shapes are adaptively deformed with respect to affinities, assuming that the object surface is locally planar and that perspective effects are neglected. A comparison of the state of the art of affine region detectors is presented in [Mikolajczyk et al., 2005]. The most common affine region detectors are:

- the Harris-affine detector [Mikolajczyk & Schmid, 2002]: the Harris-Laplace detector is used to determine localization and scale while the second moment matrix of the intensity gradient determines the affine neighbourhood.
- the Hessian-affine detector [Mikolajczyk & Schmid, 2002]: points are detected with the Hessian matrix and the scale-selection based on the Laplacian; the elliptical regions are estimated with the eigenvalues of the second moment matrix of the intensity gradient.
- the MSER (Maximally Stable Extremal Region) detector [Matas et al., 2002]: it extracts regions closed under continuous transformation of the image coordinates and under monotonic transformation of the image intensities.
- the Salient Regions detector [Kadir et al., 2004]: regions are detected measuring the entropy of pixel intensity histograms.
- the EBR (Edge-Based Region) detector [Tuytelaars & Van Gool, 2004]: regions are extracted combining interest points (detected with the Harris operator) and image edges (extracted with a Canny operator).
- the IBR (Intensity extrema-Based Region) detector [Tuytelaars & Van Gool, 2004]: it extracts affine-invariant regions studying the image intensity function and its local extremum.

3. DESCRIPTORS

Once image regions (invariant to a class of transformations) have been extracted, (invariant) descriptors can be computed to characterize the regions. The region descriptors have proved to successfully allow (or simplify) complex operations like wide baseline matching, object recognition, robot localization, etc. Common used descriptors are:

- the SIFT descriptors [Lowe, 2004]: the regions extracted with DoG detector are described with a vector of dimension 128 and the descriptor vector is divided by the square root of the sum of the squared components to get illumination invariance. The descriptor is a 3D histogram of gradient location and orientation. It was demonstrated with different measures that the SIFT descriptors are superior to others [Mikolajczyk & Schmid, 2003]. An extended SIFT descriptor was presented in [Mikolajczyk, K. & Schmid, C., 2005]: it is based on a gradient location and orientation histogram (GLOH) and the size of the descriptor is reduced using PCA (Principal Component Analysis).
- Generalized moment invariant descriptors [Van Gool et al., 1996]: given a region, the central moments M^{a}_{pq} (with order $p+q$ and degree a) are computed and combined to get invariant descriptors. The moments are independent, but for high order and degree, they are sensitive to geometric and photometric distortion. These descriptors are suitable for color images.
- Complex filters descriptors [Schaffalitzky & Zissermann, 2002]: regions are firstly detected with Harris-affine or MSER detector. Then descriptors are computed using a bank of linear filters (similar to derivates of a Gaussian) and deriving the invariant from the filter responses. A similar approach was presented in [Baumberg, 2000].

Matching procedures can be afterwards applied between couple of images, exploiting the information provided by the descriptors. A typical strategy is the computation of the Euclidean or Mahalanobis distance between the descriptor elements. If the distance is below a predefined threshold, the match is potentially correct. Furthermore, cross-correlation or Least Squares Matching (LSM) [Gruen, 1985] could also be applied to match the regions (see Section 5) while robust estimators can be employed to remove outliers in the estimation of the epipolar geometry.

4. EXPERIMENTAL SETUP AND EVALUATION RESULTS

Five interest point detectors (Förstner, Heitger, Susan, Harris and Hessian) have been firstly compared with different tests, as described in Section 4.1 and Section 4.2 while in Section 4.3 and 4.4 two region detectors/descriptors (Harris-affine and Lowe) are also considered.

In our work, the evaluation is performed calculating the number of correct corners detected (Section 4.1), their correct localization (Section 4.2), the density of detected points/regions (Section 4.3) and analyzing the relative orientation results between stereo-pairs (Section 4.4). The operators used in the comparison have been implemented at the Institute of Geodesy and Photogrammetry (ETH Zurich), except Harris-affine [Mikolajczyk & Schmid, 2002] and [Lowe, 2004] operators, available on the Internet.

4.1 Corner detection under different transformations

A synthetic image containing 160 corners is created and afterwards rotated, distorted and blurred (Figure 2). Corners are firstly detected with the mentioned operators and then compared with the ground-truth (160).

In Table 1 the numbers of detected corners are presented. Förstner and Heitger performed always better than the other detectors in all the analyzed images.

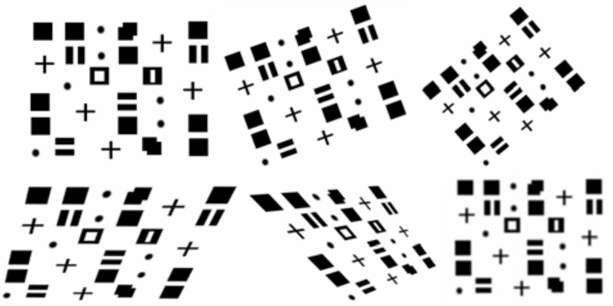


Figure 2: Synthetic images used for the corners detection. The images are numbered left to right from the top-left (1).

	IMAGE 1	IMAGE 2	IMAGE 3	IMAGE 4	IMAGE 5	IMAGE 6 (blur)
Förstner	160/160	159/160	154/160	149/160	145/160	145/160
Heitger	160/160	157/160	158/160	148/160	145/160	148160
Susan	150/160	139/160	118/160	90/160	121/160	141/160
Harris	140/160	139/160	136/160	140/160	121/160	144/160
Hessian	150/160	144/160	142/160	149/160	145/160	140/160

Table 1: Results of the interest point detection on the synthetic images of Figure 1.

4.2 Localization accuracy

The localization accuracy is a widely used criterion to evaluate interest points. It measures whether an interest point is accurately located at a specific location (ground truth). The evaluation requires the knowledge of precise camera and 3D information or simply requires the knowledge of the precise 2D localization of the feature in image space. This criterion is very important in many photogrammetric applications like camera calibration or 3D object reconstruction.

In our experiment, performed on Figure 3 (upper left), the correct corner localizations are achieved with manual measurements. The detected corners obtained from the different operators are afterwards compared with the manual measurements and the differences plotted, as shown in Figure 3. Heitger detector presents only 2 times one-pixel shifts while Harris and Hessian detectors have always a constant shift of one pixel. This might be an implementation problem, but tests performed with other detectors available on the Internet reported the same results.

4.3 Quantitative analysis based on relative orientation between image pairs

Interest points and regions detectors are also used to automatically compute the relative orientation of image pairs. Firstly points (regions) are detected, then matched and finally the coplanarity condition is applied. The correspondences are double-checked, by means of visual inspection and blunder detection (Baarda test and RANSAC estimator), therefore no outliers are present in the data. The extracted points are also well distributed in the images, providing a good input for a relative orientation problem. For each image pair, the same interior orientation parameters are used.

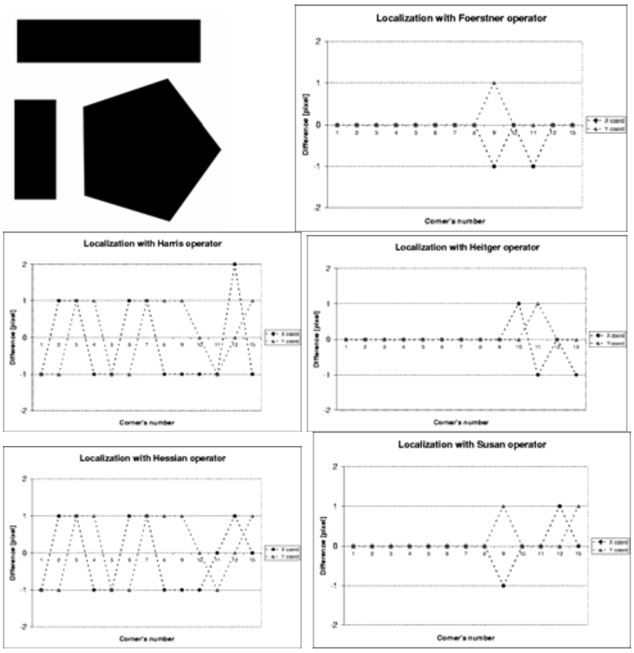


Figure 3: Synthetic image used to evaluate the localization accuracy of the point detectors (upper left). Results of the localization analysis expressed as differences between manual measurements (reference) and automatically detected points.

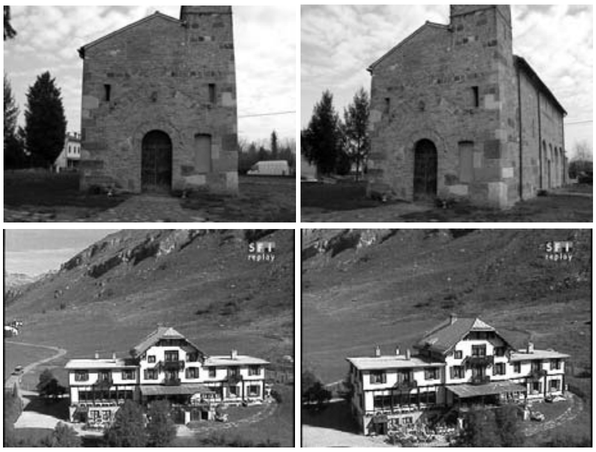


Figure 4: Two stereo-pairs used for the automated relative orientation computation. Church (1024x768 pixel), Hotel (720x576 pixel).

		CHURCH	HOTEL
Förstner	matched	145	89
	sigma$_0$	0.0183	0.0201
Heitger	matched	133	106
	sigma$_0$	0.0217	0.0207
Susan	matched	127	122
	sigma$_0$	0.0174	0.0217
Harris	matched	184	85
	sigma$_0$	0.0256	0.0425
Hessian	matched	93	91
	sigma$_0$	0.0259	0.0290
Lowe	matched	269	135
	sigma$_0$	0.0341	0.0471
Harris-Affine	matched	139	94
	sigma$_0$	0.0321	0.0402

Table 2: Results of the relative orientation between stereo-pairs in terms of matched points and sigma naught [mm] of the adjustment.

In Table 2 the results of the experiments are reported. To notice the fact that with region detectors (Lowe and Harris-affine operators), the number of matched correspondences is maybe higher but the accuracy of the relative orientation is almost two time worst than with an interest points detector.

5. ACCURACY IMPROVEMENT OF DETECTOR AND DESCRIPTOR LOCATIONS

As shown in section 4.4, region detectors and descriptors provide worst accuracy compared to corners in orientation procedures. The reason might be explained as follow (Figure 5): regions are localized with their centroid and generally matched using the extracted descriptor feature vectors. But, due to perspective effects between the images, the centre of the regions might be slightly shifted, leading to lower accuracy in the relative orientation.

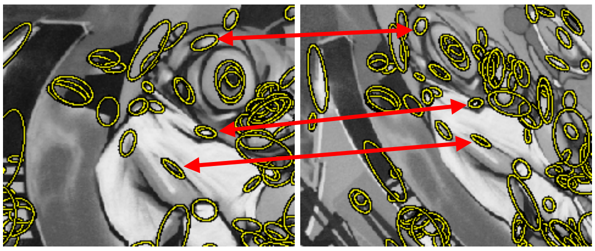


Figure 5: Affine regions detected with Harris detector [Mikolajczyk et al., 2004] with homologues regions. Due to perspective effects, the centre of the regions might be slightly shifted (red arrows).

Affine invariant regions are generally drawn as ellipses, using the parameters derived from the eigenvalues of the second moment matrix of the intensity gradient [Lindeberg, T., 1998; Mikolajczyk, K. and Schmid, C., 2002]. The location accuracy of the region centers can be improved using a LSM algorithm. The use of cross-correlation would fail in case of big rotations around the optical axis and big scale changes, both typical situations in wide baseline images. The ellipse parameters of the regions (major and minor axis and inclination) can be used to derive the approximations for the affine parameters transformation of the LSM. Indeed LSM can cope with different image scale (up to 30%) and significant camera rotation (up to 20 degrees), if good and weighted approximations are used to constraint the estimation in the least squares adjustment.

An example is shown in Figure 6. Given a detected affine region and its ellipse parameters in the template and search image, LSM is computed without and with initial approximations (provided by the region detector), leading to wrong convergence and correct matching results.

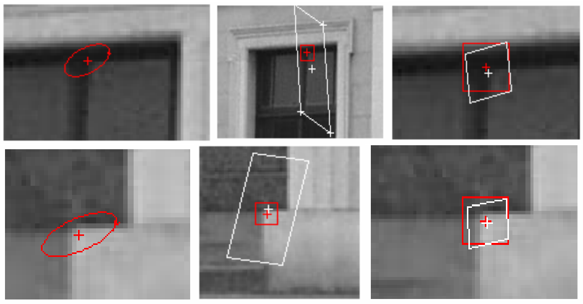


Figure 6: Detected affine region (left). Wrong LSM results with strongly deformed image patch in the search image, initialized with the centroid of the region (centre). LSM result (right) obtained using the approximations derived by the region detector algorithm.

For the church example of Section 4.3, all the extracted Lowe points (regions) were re-located, as previously described, by means of LSM algorithm. The final precision of the relative orientation decreased to 0.0259 mm.

6. CONCLUSIONS

An evaluation and comparison of interest point and region detectors and descriptors has been presented. As the selection of comparison criteria is quite difficult, we tried to used measures and procedures which are typical in photogrammetric applications. Moreover, we showed how to improve to location accuracy of region detectors using a classical least squares measurement algorithm.

From all our tests and results, [Förstner & Guelch, 1987] and [Heitger et al., 1992] operators showed better results than the others examined algorithms. Compared to other evaluation papers, we performed a quantitative analysis of the analyzed point detectors, based on the relative orientation. On the other hand, region detectors and descriptors, as they detect an area and not a single point, reported worst accuracy in the relative orientation problem. In fact they might detect the same region, but the centroid of the region (i.e. the point used to solve for the image orientation) might be shifted due to perspective effects. Nevertheless, they generally provide for affinity invariant parameters, which can be used as approximations for a least squares matching measurement algorithm, which would not converge without good approximations due to the large camera rotations or scale change. Therefore regions could also be good image features for precise and automated orientation procedures, in particular with images acquired under a wide baseline.

As final remark, we should mention that each operator has its own set of parameters which are generally used fix and constant for the entire image. An adaptive parameter selection could help in the optimization of the point selection and distribution.

REFERENCES

Baumberg, A., 2002: Reliable feature matching across widely separated views. Proc. of CVPR, pp. 774-781

Beaudet, P., 1978: Rotationally invariant image operators. Proc. 4th Int. Joint Conference on Pattern Recognition, pp. 579-583

Brown, M. Szeliski, R. and Winder, S., 2005: Multi-image matching using multi-scale oriented patches. IEEE CVPR'2005 Proceedings, Vol. I, pp. 510-517

Förstner, W. and Guelch, E., 1987: A fast operator for detection and precise location of distinct points, corners and center of circular features. ISPRS Conference on Fast Processing of Photogrammetric Data, Interlaken, Switzerland, pp. 281-305

Gruen, A., 1985: Adaptive least square correlation: a powerful image matching technique. South African Journal of PRS and Cartography, Vol. 14(3), pp. 175-187

Haralik, R.M., 1985: Second directional derivative zero-crossing detector using the cubic facet model. Proceedings of 4th Scandinavian Conference on Image Analysis, pp.17–30

Harris, C. and Stephens, M., 1988: A combined edge and corner detector. Proc. of Alvey Vision Conference, pp. 147-151

Haralick, R.M. and Shapiro, L.G., 1992: Computer and Robot Vision, Addison-Wesley, 630 pp.

Heath, M., Sarkar, S., Sanocki, T. and Bowyer, K.W., 1997: A Robust Visual Method for Assessing the Relative Performance of Edge-Detection Algorithms. IEEE Transactions on PAMI, Vol. 19(12), pp. 1338-1359

Heitger, F., Rosenthalter, L., von der Heydt, R., Peterhans, E. And Kuebler, O., 1992: Simulation of neural contour mechanism: from simple to end-stopped cells. Vision Research, Vol. 32(5), pp. 963-981

Kadir, T. and Brady, M., 2001. Scale, saliency and image description. International Journal of Computer Vision, Vol. 45(2), pp. 83–105

Kadir, T., Zissermann, A. and Brady, M., 2004: An affine invariant salient region detector. Proc. of 8th ECCV

Ji, Q. and Haralick, R.M., 1997: Corner detection with covariance propagation, Proc IEEE Conf Computer Vision and Pattern Recognition, pages 362—367

Jurie, F. and Schmid, C., 2004: Scale-Invariant shape features for recognition of object categories. Proc. of CVPR, vol. 02, pp. 90-96

Leibe, B. and Schiele, B., 2004: Scale-Invariant Object Categorization using a Scale-Adaptive Mean-Shift Search. DAGM Pattern Recognition Symposium, Springer LNCS, Vol. 3175, pp. 145-153

Lindeberg, T., 1998: Feature detection with automatic scale selection. International Journal of Computer Vision, Vol. 30(2), pp. 79-116

Lowe, D., 2004: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, Vol. 60(2), pp. 91-110

Lucas, D. and Kanade, T., 1981: An Iterative Image Registration Technique with an Application to Stereo Vision. Int. Joint Conference on Artificial Intelligence, pp. 674-679

Matas, J., Chum, O., Urban, M. and Pajdla, T., 2002: Robust wide baseline stereo from maximally stable extremal regions. Proc. of British Machine Vision Conference, pp. 384-393

Mikolajczyk, K. and Schmid, C., 2001: Indexing based on scale invariant interest points. Proc. of 8th ICCV, pp. 525-531

Mikolajczyk, K. and Schmid, C., 2002: An affine invariant interest point detector. Proc. of 7th ECCV, pp. 128-142

Mikolajczyk, K. and Schmid, C., 2003: A performance evaluation of local descriptors. Proc. of CVPR

Mikolajczyk, K. and Schmid, C., 2004: Scale and Affine Invariant Interest Point Detectors. Int. Journal Computer Vision, Vol. 60(1), pp. 63-86

Mikolajczyk, K. and Schmid, C., 2005: A performance evaluation of local descriptors. Accepted for PAMI

Mikolajczyk, K. Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Van Gool, L. 2005: A comparison of affine region detectors. Int. Journal of Computer Vision

Moravec, H.P., 1979: Visual mapping by a robot rover. Proc. 6th International Joint Conference on Artificial Intelligence, pp. 598-600

Rodehorst, V. and Koschan, A., 2006: Comparison and Evaluation of Feature Point Detectors. Proc. of the 5th International Symposium Turkish-German Joint Geodetic Days

Schaffalitzky, F. and Zisserman, A., 2002: Multi-view matching for unordered image sets. Proc. of ECCV

Smith, S.M. and Brady, J.M., 1997: SUSAN – a new approach to low level image processing. Int. Journal Computer Vision, Vol. 23(1), pp. 45-78

Schmid, C., Mohr, R. and Bauckhage, C., 1998: Comparing and Evaluating Interest Points. Proc. of ICCV'98

Tuytelaars, T. and Van Gool, L., 2004: Matching widely separated views based on affine invariant regions. International Journal of Computer Vision, Vol. 59(1), pp. 61-85

Tomasi, C. and Kanade, T, 1991: Detection and Tracking of Point Features. Carnegie Mellon University Technical Report CMU-CS-91-132

Van Gool, L., Moons, T. and Ungureanu, D., 1996: Affine / photometric invariants for planar intensity pattern. Proc. of 4th ECCV, pp. 642-651

Ziou, D. and Tabbone, S., 1998: Edge Detection Techniques - An Overview. Journal of Pattern Recognition and Image Analysis. Vol. 8, pp. 537-559

Zuliani, M., Kenney, C, and Manjunath, B.S., 2004: A mathematical comparison of point detectors. Proc. of CVPR 'Image and Video Registration Workshop'

A PROBABILISTIC FUSION STRATEGY APPLIED TO ROAD EXTRACTION FROM MULTI-ASPECT SAR DATA

Karin Hedman [a, *], Stefan Hinz [b], Uwe Stilla [a]

[a] Photogrammetry and Remote Sensing, [b] Remote Sensing Technology,
Technische Universitaet Muenchen,
Arcisstrasse 21, 80333 Muenchen, Germany

{karin.hedman,stefan.hinz,stilla}@bv.tum.de

KEY WORDS: SAR, road extraction, multi-aspect images, fusion

ABSTRACT:

In this paper, we describe an extension of an automatic road extraction procedure developed for single SAR images towards multi-aspect SAR images. Extracted information from multi-aspect SAR images is not only redundant and complementary, in some cases even contradictory. Hence, multi-aspect SAR images require a careful selection within the fusion step. In this work, a fusion step based on probability theory is proposed. Before fusion, the uncertainty of each extracted line segment is assessed by means of Bayesian probability theory. The assessment is performed on attribute-level and is based on predefined probability density functions learned from training data. The prior probability varies with global context. In the first part the fusion concept is introduced in a theoretical way. The importance of local context information and the benefit of incorporating sensor geometry are discussed. The second part concentrates on the analysis of the uncertainty assessment of the line segments. Finally, some intermediate results regarding the uncertainty assessment of the line segments using real SAR images are presented.

1. INTRODUCTION

Synthetic aperture radar (SAR) holds some advantages against optical image acquisition. SAR is an active system, which can operate during day and night. It is also nearly weather-independent and, moreover, during bad weather conditions, SAR is the only operational system available today. Road extraction from SAR images therefore offers a suitable complement or alternative to road extraction from optical images [Bacher & Mayer, 2005]. The recent development of new high resolution SAR systems offers new potential for automatic road extraction. Satellite SAR images up to 1 m resolution will soon be available by the launch of the German satellite TerraSAR-X [Roth, 2003]. Airborne images already provide resolution up to 1 decimetre [Ender & Brenner, 2003]. However, the improved resolution does not automatically make automatic road extraction easier, yet it faces new challenges. Especially in urban areas, the complexity arises through dominant scattering caused by building structures, traffic signs and metallic objects in cities. These bright features hinder important road information. In order to fully exploit the information of the SAR scene, bright features and their contextual relationships can be incorporated into the road extraction procedure. Detected vehicles and rows of building layover as well as metallic scattering caused by road signs are indicators of roads [Wessel & Hinz, 2004], [Amberg, et al. 2005].

The inevitable consequences of the side-looking geometry of SAR, occlusions caused by shadow- and layover effects, is present in forestry areas as well as in built-up areas. In urban areas, the best results for the visibility of roads are obtained, when the illumination direction coincide with the main road orientations [Stilla et al., 2004]. Preliminary work has shown that the usage of SAR images illuminated from different directions (i.e. multi-aspect images) improves the road extraction results. This has been tested both for real and simulated SAR scenes [Tupin et al. 2002], [Dell'Acqua et al., 2003]. Multi-aspect SAR images contain different information, which is both redundant and complementary. A correct fusion step has the ability to combine information from different sensors, which in the end is more accurate and better than the information acquired from one sensor alone.

In this article we present a fusion concept based on a Bayesian statistical approach, which incorporates both global context and sensor geometry. A short overview of the road extraction procedure will be given in Sect. 2. The main focus of this paper is the proposed fusion module, which is explained in Sect. 3. Some intermediate results of an uncertainty assessment of line segments based on a training step and global context are discussed in Sect 4.

2. ROAD EXTRACTION SYSTEM

The extraction of roads from SAR images is based on an already existing road extraction approach [Wessel & Wiedemann, 2003], which was originally designed for optical images with a ground pixel size of about 2m [Wiedemann & Hinz, 1999]. The first step consists of line extraction using Steger's differential geometry approach [Steger, 1998], which is followed by a smoothening and splitting step. By applying explicit knowledge about roads, the line segments are evaluated according to their attributes such as width, length, curvature, etc. The evaluation is performed within the fuzzy theory. A

[*] Corresponding author.

weighted graph of the evaluated road segments is constructed. For the extraction of the roads from the graph, supplementary road segments are introduced and seed points are defined. Best-valued road segments serve as seed points, which are connected by an optimal path search through the graph. The approach is illustrated in Fig. 1.

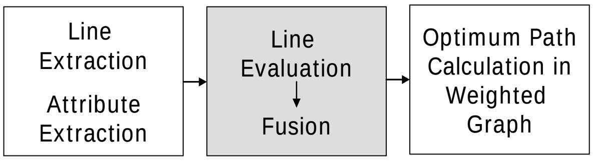


Figure 1. Automatic road extraction process

The novelty presented in this paper refers on one hand to the adoption of the fusion module to multi-aspect SAR images and on the other hand to a probabilistic formulation of the fusion problem instead of using fuzzy-functions (marked in gray in Fig. 1).

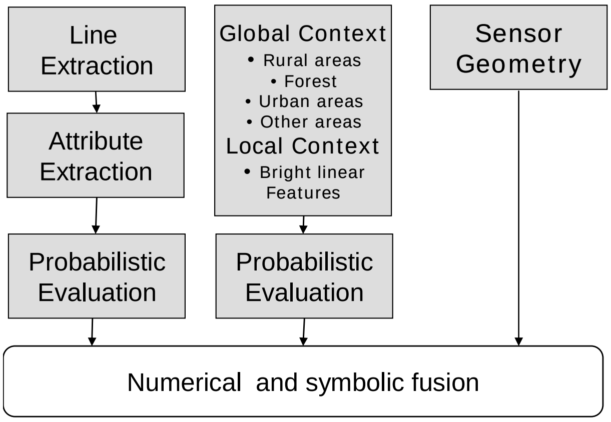


Figure 2. Fusion module and its input data

3. PROBABILISTIC FUSION CONCEPT

Line extraction from SAR images often delivers partly fragmented and erroneous results. Especially in forestry and in urban areas over-segmentation occurs frequently. Attributes describing geometrical and radiometric properties of the line segments can be helpful in the selection and especially for sorting out the most probable false alarms. However, these attributes may be ambiguous and are not considered to be reliable enough when used alone. Furthermore occlusions due to surrounding objects may cause gaps, which are hard to compensate. One step to a solution is the use of multi-aspect SAR images. If line extraction fails to detect a road in one SAR view, it might succeed in another view illuminated from a more favourable direction. Therefore multi-aspect images supply the interpreter with both complementary and redundant information. But due to the over-segmented line extraction, the information is often contradicting as well. To be able to solve possible conflicts, the uncertainty of the incoming information must be considered.

Many methods, both numerical and symbolic, can be applied for the fusion process. Some frameworks worth to mention, are evidence theory, fuzzy-set theory, and the probability theory. The last one is, regarding its theoretical foundations, the best understood framework to deal with uncertainties. In this chapter we will discuss a fusion process accommodating for these aspects.

3.1 Features, Attributes and Evaluation

Man-made objects in general tend to have regular geometrical shapes with distinct boundaries. The main feature involved in the road extraction process is the line segment, which can either belong to the class ROADS or to the class FALSE_ALARMS. The selection of attributes of the line segments is based on the knowledge about roads. Roads in SAR images appear as dark lines since the smooth surface of a road acts like a mirror. Therefore radiometric attributes such as *mean* and *constant intensity,* and *contrast* of a line as well as geometrical attributes like *length* and *straightness* should be representative attributes for roads.

Other features of interest are linked to global and local context. Bright linear features (BRIGHT_LINES) represent the local context in this work. The global region features applied in this work are URBAN, FOREST, FIELDS and OTHER_AREAS. These regions are of interest, since road attributes may have varying importance depending on the global context region. For example, length becomes more significant for roads in rural areas, but may be of less importance in urban areas.

By means of an attribute vector ***x***, the probability that a line segment belongs to the class ω_i (i.e. ROADS or FALSE_ALARMS) is estimated by the well-known Bayesian formula,

$$p(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)\ p(\omega_i)}{\sum_{j=1}^{M} p(\mathbf{x}|\omega_j)\ p(\omega_j)}. \qquad (1)$$

If there is no correlation between the attributes, the likelihood $p(\mathbf{x}|\omega_i)$ can be assumed equal to the product of the separate likelihoods for each attribute

$$\begin{aligned} p(\mathbf{x}|\omega_i) &= p(x_1, x_2,..x_n|w_i) \\ &= p(x_1|w_i)\ p(x_2|w_i)\ \ldots\ p(x_n|w_i) \end{aligned}. \qquad (2)$$

It is important to show that this simplification is valid for the data used. Furthermore, it should be noted that this is not a definite classification; instead each line segment obtains an assessment, which is necessary for the subsequent fusion of multi-aspect SAR images.

3.2 Definition and Validation of Probability Density Functions

Each separate likelihood $p(x_j|\omega_i)$ is approximated by a probability density function learned from training data. Learning from training data means that the extracted line segments are sorted manually into two groups, ROADS and FALSE_ALARMS. The global context (URBAN, FOREST, FIELDS and OTHER_AREAS) is specified for each line segment as well. A global context term will be helpful by the latter estimation of the prior term $p(\omega_i)$. The training data used is X-band, multi-looked, ground range SAR data with a resolution of about 0.75 m. The small test area is located near the airport of DLR in Oberpfaffenhofen, southern Germany.

The independence condition has been empirically proved by a correlation test using the training data. Only two attributes, *mean intensity* and *constant intensity,* showed any correlation, which in fact can be expected due to the speckle characteristics of SAR data. As a conclusion, the factorized likelihoods can not be applied for these two attributes. The rest of the attributes did not indicate any dependence. Figure 3 exemplifies this for the two attributes length and intensity.

A careful visual inspection indicated that the histograms might follow a lognormal distribution, i.e.

$$p(\omega_i|x) = \frac{1}{S\sqrt{2\pi}\;x}\, e^{-\frac{(\ln x - M)^2}{2S^2}} \quad . \qquad (3)$$

A reasonable way to test the match of histograms and parameterized distributions is to apply the Lilliefors test [Conover, 1999]. This test evaluates the hypothesis that *x* has a normal distribution with unspecified mean and variance against the alternative hypothesis that *x* does not have a normal distribution. However, the Lilliefors test tends to deliver negative results, when applied to histograms of manually selected training data, since the number of samples is naturally limited. To accommodate for this fact, the probability density functions have been fitted to the histograms by a least square adjustment of *S* and *M* since it allows to introducing a-priori variances. Figs. 4 and 5 show the histogram of the attribute *length* and its fitted lognormal distributed curve. A fitting carried out in a histogram with one dimension is relatively uncomplicated, but as soon as the dimensions increase, the task of fitting becomes more complicated. Since mean intensity and constant intensity tend to be correlated, fitting of a bivariate lognormal distribution shall be carried out. This is under development and until than, only the one-dimensional fitting of *mean intensity* is applied.

Please note that the estimated probability density functions should represent a degree of belief rather than a frequency of the behaviour of the training data. The obtained probability assessment shall correspond to our knowledge about roads. At a first glance, the histograms in Figs. 4 and 5 seem to overlap. However, Fig. 6 exemplifies for the attribute *length* that the discriminant function

$$g(x) = \ln\big(p(x|ROADS)\big) - \ln\big(p(x|FALSE\_ALARMS)\big) \quad (4)$$

increases as the length of the line segment increases. The behaviour of the discriminant function corresponds to the belief of a human interpreter. The behaviour of the discriminant function was tested for all attributes. All are illustrated in Fig 6a-d.

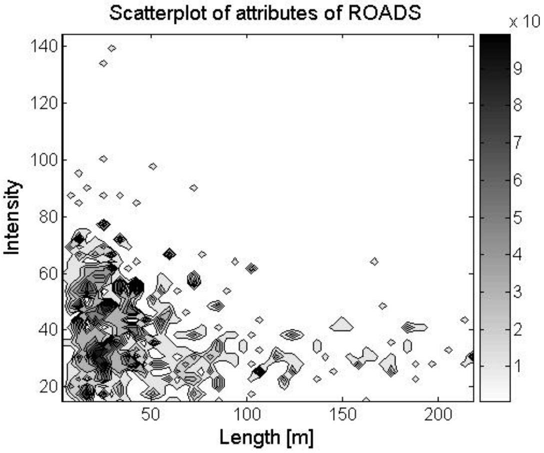


Figure 3. Scatter plot of attributes intensity and length

It should be kept in mind that statistical attributes addressing deviation and mean are not reliable for short line segments of only a few pixels length. Since these line segments are considered unreliable with respect to their short length, they can simply be sorted out. It should also be pointed out that more attributes does not necessarily mean better results, instead rather the opposite occur. A selection including a few, but significant attributes is recommended.

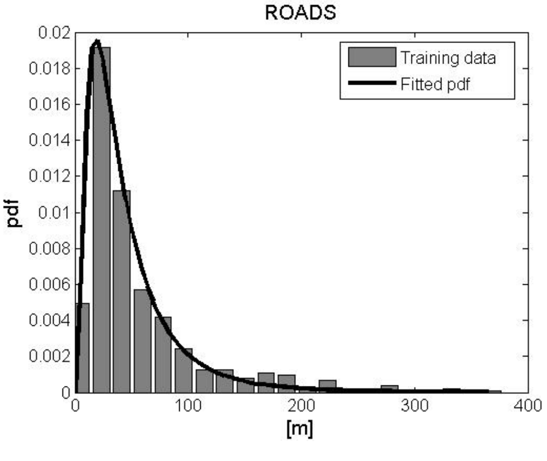


Figure 4. A lognormal distribution is fitted to a histogram of the attribute length (ROADS).

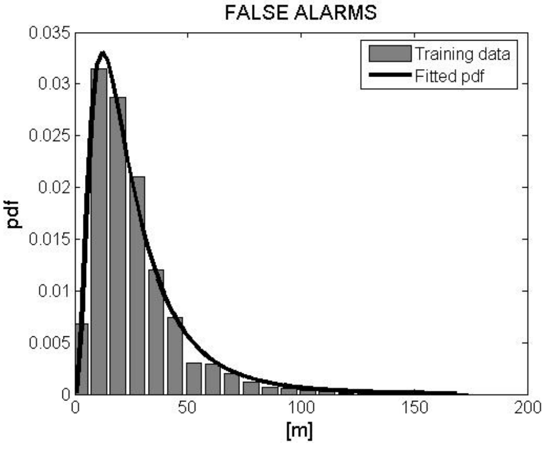
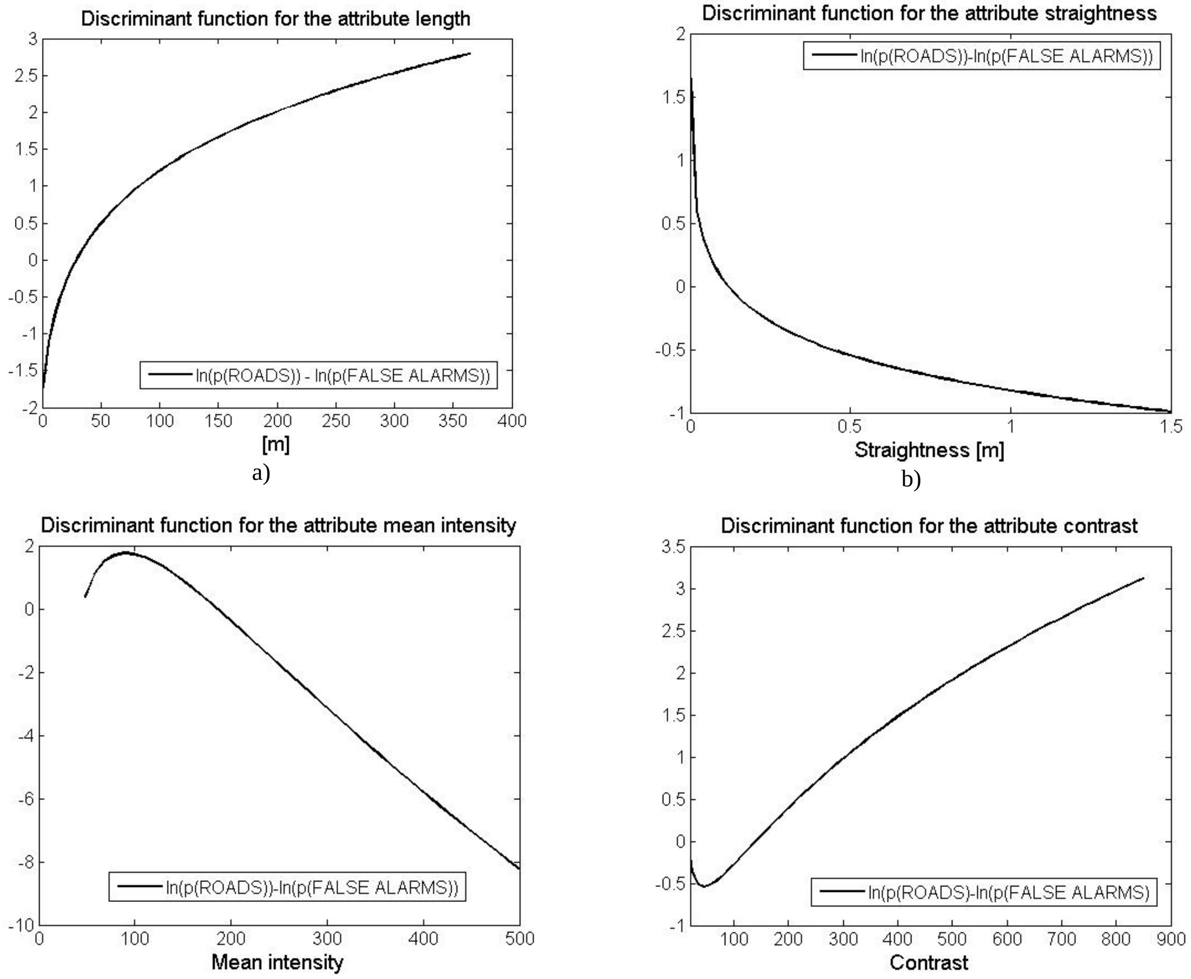


Figure 5. A lognormal distribution is fitted to a histogram of the attribute length (FALSE_ALARMS).



a) b)

c) d)

Figure 6 a-d. Discriminant function for the attributes a) Length, b) Straightness, c) Inner intensity and d) Contrast.

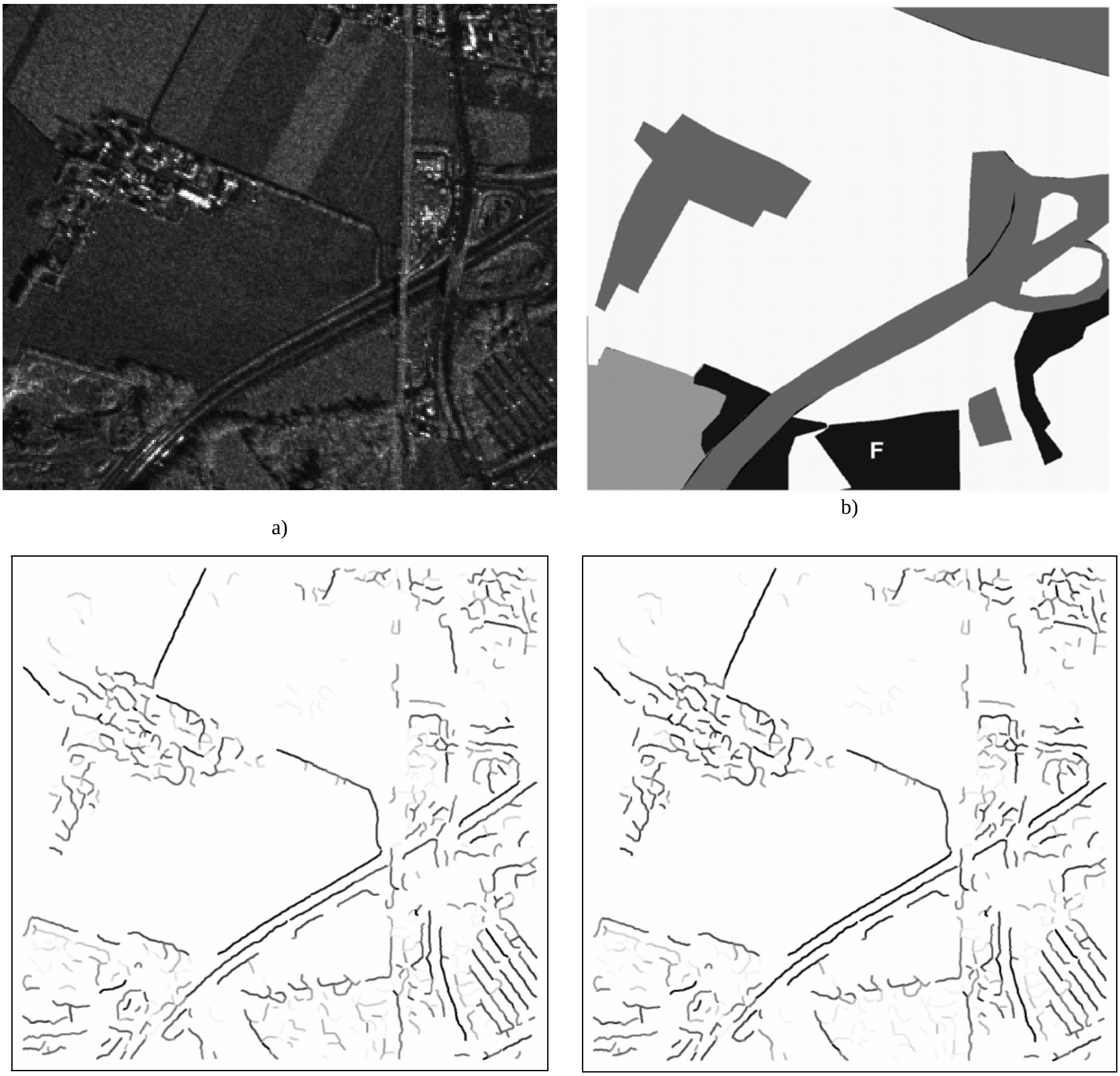
3.3 Global and Local Context

Since even a very sophisticated feature extractor delivers generally results with ambiguous semantics, additional information of global and local context is helpful to support or reject certain hypotheses during fusion. Assume, for instance that two SAR images with perpendicular view direction contain a road flanked by high buildings. The road is oriented across-track in one scene and along-track in the other scene. While in the first image, the true road surface is visible, in the second image, merely the elongated shadow of the fore-buildings and the bright, elongated layover area of the buildings across the road are detectable. The parallel appearance of bi-polar linear features (dark/light) would stand for local context, while the whole urban area would represent the global context region. Hence, a correct fusion of both views must involve a reasoning step, which is based on the sensor geometry and its influence on the relations between the extracted features. Relations between features, which appear due to local context, usually need to be detected during the extraction process. Consequently also the features involved in local context relations should be attached with confidence values.

Global context regions are derived from maps or GIS before road extraction, or can be segmented automatically by a texture analysis. As a start, global context (URBAN, FOREST, FIELDS and OTHER_AREAS) is extracted manually (see Fig. 7b). Global context plays an important role for the reasoning step within the fusion module as well as for the definition of the priori term. The frequency of roads is proportionately low in some context areas, for instance in forestry regions. The a-priori probability must be different in these areas. In this work the user specifies the priors (see Tab. 1). Therefore the priors represent the belief of the user to a certain degree. In future work, these values will be compared with values learned from training data.

Global context	*p(*ROADS*)*	*p(*FALSE_ALARMS*)*
FIELDS	0.4	0.6
URBAN AREAS	0.5	0.5
FOREST	0.1	0.9
OTHER AREAS	0.3	0.7

Table 1. Prior terms for different global context areas



a)

b)

c)

d)

Figure 7. a) SAR image analysed in this work b) Manual extraction of global context from previous SAR scene c) Results of discriminant function neglecting global context d) Results of discriminant function incorporating global context

4. RESULTS AND DISCUSSION

A cross-validation was carried out in order to examine if the assessment of a sample of the training data (1220 line segments) delivers a correct result. 83.5% of the line segments belonging to the class ROADS were correctly classified and 76.0% of the FALSE_ALARMS were correctly classified. An assessment ignoring global context did not change the number of correctly classified road segments, but deteriorated the classification of FALSE_ALARMS. As much as 54.3% of the FALSE_ALARMS are falsely classified as road segments. The prior terms of each classes were assumed to be $p($ROADS$)=0.3$ and $p($FALSE_ALARMS$)=0.7$.

The assessment was also tested on a line extraction carried out in a scene taken by the same sensor as the training data but now performed with different parameter settings. In order to test the derived likelihood functions in terms of sensitivity and ability to discern roads from false alarms, we allowed a significant over-segmentation. Results of this test are illustrated in Fig. 7c). The derived discriminant value $g(x)$ of each line segment is coded in gray, i.e. the darker the line the better the evaluation. Two assessments are carried out, one incorporating global context and one containing the same priori terms for all context areas.

A fact that comes clear from the comparison of Figs..7c) and d) is the importance of using global context for the evaluation, in particular for determining the Bayesian priors. Incorporating global context reduces the number of false alarms in forest regions (marked black in Fig. 7b). Still many line segments are falsely classified in urban regions, which indicates the need of additional local context information and a different assessment in these regions. The attribute *length*, for instance, should have less influence on the final evaluation since short line segments may also correspond to roads.

As can also be seen from Fig. 7, most line segments that correspond to roads still got a good evaluation. On the other hand, many of the false alarms in the urban and forest area are rated worse, even though also some correct segments got a bad rating. However, keeping in mind that this evaluation is only an intermediate step before fusion and network-based grouping (see flow charts in Figs. 1 and 2) the learned likelihood functions seem indeed being robust enough to be applied to different parameter settings as well as different images – of course under the condition that the image characteristics do not differ too heavily.

The results achieved so far are promising in terms that the evaluation of the lines is on one hand statistically sound and, on the other hand, it closely matches the assumptions on the significance of different attributes with respect to their distinctiveness. However, the fusion of evaluated lines from different views and thereby taking into account local context needs still to be done and analysed in depth.

ACKNOWLEDGEMENTS

The authors would like to thank the Microwaves and Radar Institute, German Aerospace Center (DLR) for providing the SAR data of Oberpfaffenhofen.

This work was done within the TUM-DLR Joint Research Lab (JRL) [http://www.ipk.bv.tum.de/jrl], which is funded by Helmholtz Gemeinschaft.

REFERENCES

Amberg, V., Coulon M., Marthon P., Spigai M., 2005. Improvement of road extraction in hihg.resolution SAR data by a context-based approach, *Geoscience and Remote Sensing Symposium, 2005. IGARSS '05.* Vol. 1, pp. 490-493.

Bacher U, Mayer H 2005 Automatic road extraction from multispectral high resolution satellite images. In: Stilla U, Rottensteiner F, Hinz S (eds) Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring - Concepts, Algorithms, and Evaluation (CMRT05). International Archives of Photogrammetry and Remote Sensing. Vol 36, Part 3 W24 : 29-34.

Conover, W. J., 1999. *Practical nonparametric statistics.* New York, Wiley.

Dell'Acqua, F., Gamba, P., Lisini, G., 2003. Improvements to Urban Area Characterization Using Multitemporal and Multiangle SAR Images. *IEEE Transactions on Geoscience and Remote Sensing*, 41(9), pp. 1996-2004-

Ender JHG, Brenner AR., 2003 PAMIR - a wideband phased array SAR/MTI system. IEE Proceedings - Radar, Sensor, Navigation, vol 150(3): 165-172.

Roth, A., 2003. TerraSAR-X: A new perspective for scientific use of high resolution spaceborne SAR data. *2nd GRSS/ISPRS Joint workshop on remote sensing and data fusion on urban areas, URBAN 2003*. IEEE, pp. 4-7.

Steger, C., 1998. An unbiased detector of curvilinear structures, *IEEE Trans. Pattern Anal. Machine Intell.,* 20(2), pp. 549-556.

Stilla, U., Michaelsen, E., Soergel, U., Hinz, S., Ender, H.J., 2004. Airborne Monitoring of vehicle activity in urban areas. In: *Altan MO (ed) International Archives of Photogrammetry and Remote Sensing*, 35(B3), pp. 973-979.

Tupin, F., Houshmand, B., Datcu, M., 2002. Road Detection in Dense Urban Areas Using SAR Imagery and the Usefulness of Multiple Views, *IEEE Transactions on Geoscience and Remote Sensing*, 40(11), pp. 2405-2414.

Wessel, B., Hinz, S., 2004. Context-supported road extraction from SAR imagery: transition from rural to built-up areas. In: *Proc. EUSAR 2004*, Ulm, Germany, pp. 399-402.

Wessel, B., Wiedemann, C., 2003. Analysis of Automatic Road Extraction Results from Airborne SAR Imagery. In: *Proceedings of the ISPRS Conference "Photogrammetric Image Analysis" (PIA'03), International Archieves of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Munich 2003, 34(3/W8), pp. 105-110.

Wiedemann, C., Hinz, S., 1999. Automatic extraction and evaluation of road networks from satellite imagery, *International Archives of Photogrammetry and Remote Sensing*, 32(3-2W5), pp. 95-100.

A VARIANT OF POINT-TO-PLANE REGISTRATION INCLUDING CYCLE MINIMIZATION

Carles Matabosch^a, Elisabet Batlle^a, David Fofi^b and Joaquim Salvi^a

^{a} Institut d'Informatica i Aplicacions, University of Girona Campus Montilivi, 17017
Girona, Spain -(cmatabos,bbatlle,qsalvi)@eia.udg.es
^{b} Le2i UMR CNRS 5158, Université de Bourgogne, rue de la fonderie 12, 71200
Le Creusot, France - d.fofi@iutlecreusot.u-bourgogne.fr

KEY WORDS: Vision, Registration, Surface, Large, Three-dimensional

ABSTRACT:

3D models are very important in many industrial and scientific applications. Most part of commercial sensors obtain only a partial acquisition of the object, so that a set of views are required to build a complete model of the object. Although the motion between these views is usually unknown, it can be computed by means of registration algorithms. A survey of most important techniques is presented in this paper, in which they have been classified into coarse and fine registration and compared in terms of the number of views aligned at every step, the accuracy and the robustness against outliers. The second part of the article presents an improvement of point-to-plane registration, which includes the determination of cycles in a sequence of views with the aim of minimizing the propagation error or drift.

1 INTRODUCTION

The acquisition and representation of 3D information is a very important topic in Computer Vision. The main steps involved in this problem are: a) Surface acquisition; b) Registration and c) Integration. Surface acquisition is focused on the search of the depth usually by means of laser scanning (Forest and Salvi, 2002) or coded structured light (Salvi et al., 2004) among others such as stereovision (Matabosch et al., 2003) or structure from motion (Armangué et al., 2003). Registration is the process to determine the Euclidean motion between two or more views of a given surface that permits to align them with respect to the same reference (Besl and McKay, 1992). Integration consists of representing the set of views in a continuous and homogeneous surface (Curless and Levoy, 1996).

Although there are many papers focused on surface acquisition, only a few of them obtain a complete reconstruction. Most papers are based on one-shot acquisition, so that only a partial view of the surface is obtained(Salvi et al., 2004). Besides, other papers take advantage of some sort of mechanical system such as robot arms or rotating tables to obtain a set of views with respect to the same reference (Levoy et al., 2000). However, the reconstruction of the surface is still incomplete due to surface occlusions depending on the shape of the surface itself and the number of degrees of freedom of the mechanics. Finally, the accuracy of these kind of systems highly depends on the accuracy of the mechanics.

Range Image Registration is a sort of techniques that computes the motion between 3D views with the aim of aligning them with respect to the same reference without any prior knowledge of the pose from where such views where acquired. Most part of techniques are centered on pair-wise registration so that only two different views are aligned in every registration. Hence, a registration error is accumulated when we are aligning a sequence of views making necessary to use a further process (multi-view) to reduce the drift once all the views are already aligned. In summary, a complete reconstruction of objects is not a trivial problem in computer vision.

In this paper a survey of the most important registration methods is presented in section 2. Furthermore, a summary of the state-of-art is given in section 3. Then, section 4 details the new method proposed to reduce the propagation errors. Experimental results are provided in section 5. The article ends with conclusions.

2 REGISTRATION ALGORITHMS

Registration is defined as the set of techniques used to determine the Euclidean motion between two or more sets of points. There are several varieties of registration: a) 2D/2D Registration; b) 2D/3D Registration and c) 3D/3D Registration. The surveyed techniques differ as to whether initial information is required, so that a *Coarse Registration* can only be estimated without an initial guess. If an estimated motion between views is available, a *Fine Registration* can then be computed.

2.1 Coarse Registration

Coarse Registration techniques can be defined as the group of techniques that estimates the motion between two views without any prior information.

There exists lots of methods to obtain a coarse estimation of the motion. Some of them are especially used in registration applications, while others are adaptations of recognition algorithms. The main idea of them is to characterize some features (points, lines, etc) in both surfaces in order to find correspondences. Although points are the most used correspondences (Chen et al., 1998) (Johnson, 1997), other characteristics can be used such as lines (Stamos and Leordeanu, 2003) or principal axis (Kim et al., 2003). For instance, Johnson (Johnson, 1997) characterizes points by using the Spin Image. This image is a 2D representation of the neighborhood of one point on the surface. Comparing Spin-images from two different surfaces, point correspondences between them can be established. Another point descriptor is the point signature (Chua, 1997). This algorithm describes a point by using all the points located in a constant distance from it obtaining a vector descriptor of the point which is then compared with all points in the second surface to find matchings.

Some authors propose to use lines to find pairs of correspondences. Examples are the straight line-based method proposed by

Stamos (Stamos and Leordeanu, 2003) and the curved line-based method proposed by Wyngaerd (Wyngaerd, 2002). The first one is only applied in structured objects, not in free-form shapes. The second one is based on extracting curves from free-form shapes to find matchings between pairs of segments.

The main problem of most of these algorithms is the large computing time involved in obtaining a solution. This is because once some points on the first surface are characterized, they must be compared with all points in the second surface in order to find correspondences. In general, Spin Image presents the best ratio accuracy/time, and the solution obtained is good enough to be used as an initial guess in a further fine registration.

Although coarse registration techniques are used in many applications, in others the motion is provided by mechanical or manual alignment. Despite the method used to estimate the initial guess, usually the registration error is minimized using next a fine registration technique.

2.2 Fine Registration

Fine registration refers to the set of techniques that obtain the Euclidean motion between two or more surfaces by an iterative minimization. The main drawback of these techniques is the requirement of an initial guess to start the process which may be quite close to the solution to guarantee the convergence. Hereafter, the most known fine registration techniques are discussed

2.2.1 Iterative Closest Point (ICP) The ICP method was presented by Besl (Besl and McKay, 1992). The goal of this method is to obtain an accurate solution by minimizing the distance between point correspondences, known as closest point. When an initial estimation is known, all the points are transformed to a reference system by applying the Euclidean motion. Then, every point in the first image is taken into consideration to search for its closest point in the second image. A new motion is estimated by the minimization of the distances between these correspondences, and the process is iterated until convergence.

ICP obtains good results even in the presence of Gaussian noise. However, the main drawback is that the method can not cope with non-overlapping regions because outliers are never removed. Moreover, when starting from a rough estimation of the motion, the convergence is not guaranteed.

Some modifications of ICP have been presented in recent years. Greenspan (Greenspan and Godin, 2001) applied the Nearest Neighbor Problem to facilitate the search of closest points. The first range image is considered as a reference set of points, which is preprocessed in order to find for every point the neighborhood of points in the second view located at a certain distance. The points of the neighborhood are sorted according to that distance. The use of this pretreatment leads to consider the closest point of the previous iteration as an estimation of the correspondence in the current iteration. If this estimation satisfies the spherical constraint, the current closest point is considered to belong to the neighborhood of the estimate. This pretreatment decreases the computing time drastically. A year later, Jost (Jost and Hugli, 2002) presented the Multi-resolution Scheme ICP algorithm, which is a modification of ICP for fast registration. The main idea of the algorithm is to solve the first few iterations using down sampled points and to progressively increase the resolution by increasing the number of points considered. The author divides the number of points by a factor in each resolution step. The number of iterations in each resolution step is not fixed, so that the algorithm goes to the next resolution when the distance between correspondences falls below a threshold.

Some other approaches (Godin et al., 2001) (Sharp et al., 2002) are presented with the aim of incorporating features in the points to increase the efficiency in the matching. In addition, other authors (Trucco et al., 1999) (Zinsser and Schnidt, 2003) proposed some improvements to increase the robustness of ICP by removing correspondences whose distances are higher than a threshold.

Overall, ICP is the most common registration method used and the results provided by authors are very good. However, this method usually presents problems of convergence, lots of iterations are required, and in some cases the algorithm converges to a local minimum. Moreover, unless a robust implementation is used, the algorithm can only be used in surface-to-model registration.

2.2.2 Method of Chen The algorithm proposed by Chen (Chen and Medioni, 1991) is an alternative to the Iterative Closest Point. The main difference between both algorithms is in the matching algorithm. While ICP uses point-to-point matchings, Chen's approach is based on point-to-plane matchings. Concretely, considering a point in the first image, the intersection of the normal vector at this point with the second surface determines a second point in which the tangent plane is computed. The distance between this plane and the initial point is the function to minimize.

Although of most part of this paper is focused on pair-wise registration, at the end, the author proposed to fuse consecutive views in a single metaview, avoiding propagation errors. This approach can be considered as the beginning of the multiview approach.

Despite of the difficulty to determine the cross point between a line and a plane in a point of clouds, some techniques are presented to speed up this process (Gagnon et al., 1994) (Park and Subbarao, 2003).

Compared to ICP, this method is more robust to local minima and, in general, better results are obtained. The method is less influenced by the presence of non-overlapping regions. The reason is that only the control points whose normal vector intersects the second view are considered in the matching, deferring from ICP, where all points in the first cloud are used in the registration. Moreover, Chen's approach usually requires less iterations compared to ICP.

2.2.3 Matching Signed Distance Fields Masuda (Masuda, 2001) (Masuda, 2002) presented a new registration algorithm based on the Matching Signed Distance Fields. The main idea of a signed distance field is to store the distance to the nearest surface for each point in space. The method is robust so that outliers are automatically removed. Another advantage of this algorithm is that all the views of a given object are registered at the same time, which means a *multi-view registration*. Hence, the propagation error problem is drastically reduced.

Summarizing, all views are first transformed to a reference coordinate system using the initial estimations of the motion. A set of key points are then generated on a fixed-size 3D grid of buckets. Finally, the closest point from every key point is searched in every surface to establish correspondences.

The algorithm presents the advantage of a multi-view registration and the fact that an integration solution is directly given. Besides, this algorithm can not be used in real time applications such as simultaneous localization and mapping because it requires the knowledge of the complete set of views to start the minimization process.

2.2.4 Genetic Algorithms Chow (Chow et al., 2004) presented a dynamic genetic algorithm to solve the registration problem. The goal of this method is to find a chromosome composed of the 6 parameters of the motion that aligns a pair of range images accurately. The chromosome is composed of the three components of the translation vector and the three angles of the rotation matrix. In order to minimize the registration error, the median of distances between correspondences is chosen as the fitness function.

Therefore, only a sample of points of the first image are used to compute the error with the aim of decreasing the computing time. New chromosomes (potential solutions) are generated by crossover and mutation operators. The cross-over operation consists in combining genes made by two chromosomes to create a new chromosome. The number of genes to be swapped is randomly selected in each iteration. The cross-over operation works well when the chromosome is far from the final solution but it is useless for improving the solution in a situation close to convergence. Therefore, the mutation operation was defined as follows: a gene is randomly selected and a value randomly obtained between the limits $[-MV, +MV]$ is added. The limits are very wide at the beginning and become narrower at every step in order to guarantee the convergence in the final steps.

A similar method was proposed the same year by Silva (Silva et al., 2003). The main advantage of this work is that a more robust fitness function is used and no initial guess is required. The author defined the Surface Interpenetration Measure (SIM) as a new robust measurement that quantifies visual registration errors. Another advantage compared to Chow's method is the multi-view registration approach. Finally, the hillclimbing strategy was used to speed up the convergence.

Overall, the use of genetic algorithms has the advantage of avoiding local minima, which is a common problem in registration, especially when the initial motion is not provided or it is given with low precision. This algorithm also works well in the presence of noise and outliers given by non overlapping regions. The main drawback of this algorithm is the time required to converge.

3 SUMMARY OF THE STATE-OF-ART

Referring to Pair-wise registration, Chen's approach presents the best results in terms of accuracy and convergence. Although, the fact of computing the normal vectors may be considered a drawback, most of the commercial sensors directly provide this information during the acquisition step. Otherwise, normal vectors can be estimated by local planar approximation. Another important aspect is that Chen's approach obtains the best results in case of low sampling data. The reason is that ICP needs point-to-point correspondences, so that in the presence of a low resolution it is very difficult to ensure that the same 3D point is present in both views. Besides, point-to-plane distances let us to establish correspondences between points in the second image that are not present but estimated by a local planar approximation. So, it is easier to find fine correspondences in a point-to-plane approach.

Another important aspect in registration techniques is the percentage of overlapping area. Although original ICP can not cope with non-overlapping area, robust variants presented by Trucco and Zinsser obtain good results because of the removal of outliers (Trucco et al., 1999) (Zinsser and Schnidt, 2003). In Chen's approach, as only correspondences are considered if the normal vector intersect with the other surface, some outliers are removed avoiding convergence problems. The method of Chow is also very robust against outliers, however the high computing time is an important drawback in genetic algorithms.

Most part of algorithms presented are based on Pair-wise registration, so that only two views are registered simultaneously. This fact implies that in the presence of more views, a sequence of pair-wise registration must be computed. As every registration presents errors in the computation, this error is accumulated through all the views producing a drift in the alignment. In order to solve this problem, a refinement step is required. There are several possibilities to apply this refinement. A solution is to apply a multi-view algorithm (Pulli, 1999) (Masuda, 2001). Although this is probably the most accurate solution, it presents some problems when lots of views are used. First, the time involved in the registration is very high. Second, due to propagation errors, initial guess of the multi-view algorithm can be far from the solution, producing errors in the convergence. Finally, it can only be used once all views are already acquired.

With the aim of solving these problems, some other proposals have been recently presented. The main idea is to determine loops between the views. A cycle is considered when the actual acquisition contains significant overlapping area with a previous surface. A minimum number of views is required in order to avoid loops in consecutive acquisitions. The idea of a loop is similar to robot navigation where a cycle is considered when the same place is revisited by a robot. When a cycle is determined, the accumulated registration error associated is computed by forcing the product of all matrices to be the identity. Some authors (Sharp et al., 2004) distributes the error through all the views of the cycle. However, some rules are required to distribute the error between views properly. Another important step is the way a cycle is determined. Registration errors can increase dramatically if a cycle is estimated between views that do not really form a cycle.

Although, the method proposed by Sharp solves the drift problem between the initial and the final view in a cycle, the propagation error is not always correctly distributed through the rest of views. The final view is forced to be well registered to the initial view, and the transformation involved in this motion is distributed through the rest of views depending on the weight associated to each view. Hence, the selection of the weights of every view is crucial to obtain good results. If these weights are not very accurate, the error is badly distributed, obtaining misalignments inside the loop. Views near the endings are good located, but not the views far from them. In order to solve this problem, we propose to analyze simultaneously all the views belonging to the loop, as explaining next section.

4 REFINEMENT STEP

In order to solve the problem of the propagation error without using all views in the minimization, we propose to minimize the error in a loop by only considering the views that have common information. Note that in large sequence of views, when views are registered simultaneously, a lot of time in general is wasted in searching potential correspondences between views that do not even contain overlapping area.

Our algorithm is based on Pair-wise registration of consecutive views until a cycle is determined reducing the search of correspondences to only the views with overlapping area. Then, all the views of the cycle are minimized simultaneously to remove propagation errors. Finally, the algorithm follows until another cycle is found or no more surfaces are acquired.

The goal of our application is to develop an algorithm to register surfaces acquired by a 3D hand-sensor. Our refinement approach

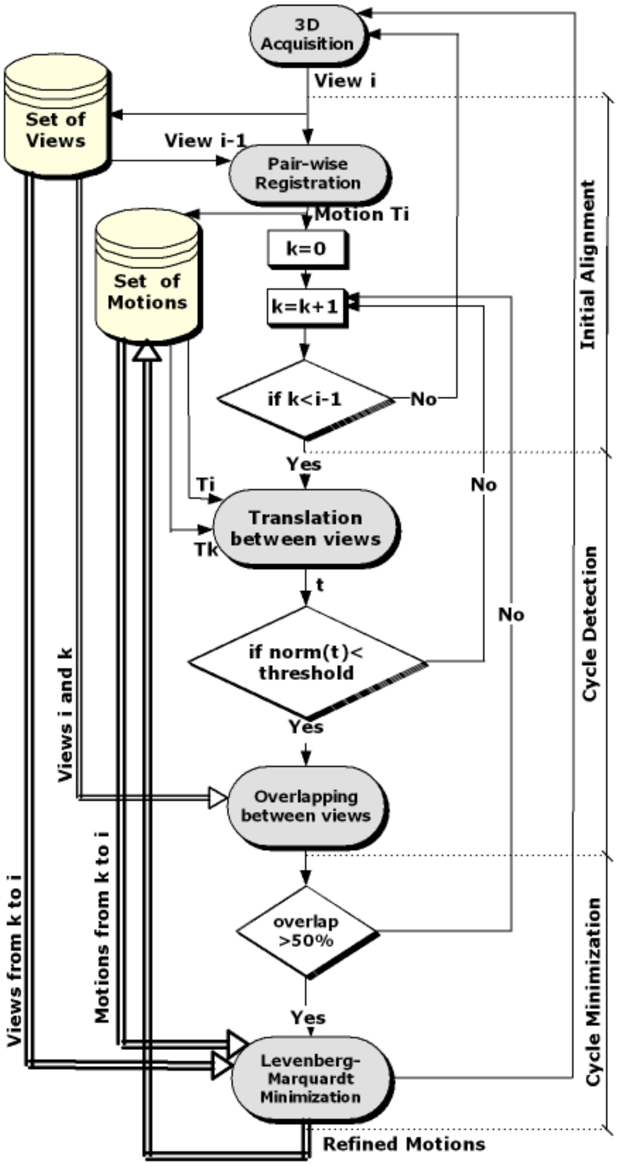


Figure 1: Flow diagram of the proposed method

is composed of three main parts: a) Initial alignment; b) Cycle detection; and c) Cycle Minimization. All the steps are shown in Figure 1 and detailed in the following section.

4.1 Initial alignment

The first part of the algorithm is focused on obtaining an initial alignment. As views are acquired consecutively, we assume that two consecutive views are close one to the other. This assumption only fails when we analyse two views that have not been acquired consecutively but they belong to a sequence. In this case, the motion between both views is computed by the product of all the motions in the sequence.

The algorithm selected is based on the method of Chen. However, some modifications have been done to increase the accuracy. The Normal Space Sampling defined by Rusinkiewicz (Rusinkiewicz and Levoy, 2001) is added in order to select the most representative points in the first image. Furthermore, the proposal of Park (Park and Subbarao, 2003) is used to speed up the process. An example of registration is presented in Figure 2.

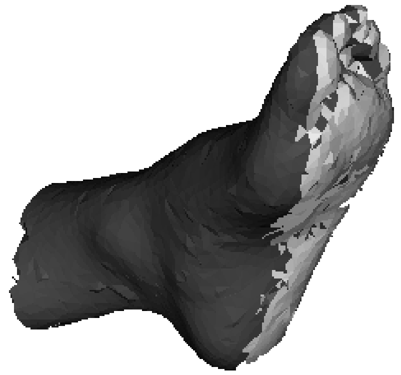


Figure 2: Result of pair-wise registration between two consecutive views

4.2 Cycle detection

A cycle is defined as a set of views that forms a sequence and the initial and final views shares a large overlapping area. The cycle determination step consists in searching for surfaces whose overlapping region is significant. As two consecutive views contain lots of points in common but do not form a cycle, a minimum number of views in a sequence is required to check if they form a cycle.

In order to determine if two views are close enough, the motion between them is computed by using pair-wise registration. The motion (${}^{j}T_{i}$) between any view (i) and the last view acquired (j) is estimated by the product of all consecutive motions (${}^{k}T_{k-1}$) from i to j as shown in equation 1.

$$ {}^{j}T_{i} = \prod_{k=i+1}^{j} {}^{k}T_{k-1} \tag{1} $$

Then, the translation is given by the fourth column of ${}^{j}T_{i}$. Finally, both views are considered close one to the other if the norm of the translation vector is smaller than a threshold.

In order to validate this result, the overlapping percentage between both views is computed. First, as the computation of the overlapping region is hard consuming, an approximation is applied. Hence, the 3D bounding box of both surfaces is computed. Then, the overlapping is analyzed in 2D by projecting both bounding boxes on the planes X-Y, X-Z and Y-Z. Then, the percentage of overlapping area is computed by means of the overlapping of the bounding boxes in such planes. If this overlapping percentage is higher than a threshold (50% in our case), a loop is considered between these views. Second, in order to speed up the process and assuming that the real overlapping area is not necessary but just a percentage, an approximative but very fast computation is proposed. Hence, a nxn matrix is defined whose elements are increase by 1 if they belong to any box, and unset to 0 otherwise. Then, an approximation of the overlapping area is obtained by counting the number of 2 divided to the area formed by both boxes.

4.3 Cycle minimization

When a cycle is found, a multi-view minimization must be applied to decrement the propagation errors. In order to take into account all the views of the cycle, corresponding pairs are simultaneously searched for in all views. For each view i, the translation vector with respects to the other views j is computed. If the distance is small enough to guarantee an overlapping region, point-to-plane correspondences are searched for, obtaining two sets of points P_{ik} and P_{jk}, where P_{ik} and P_{jk} are the points from the view i and j, respectively. Then, the function f to minimize is the following:

$$f = \sum_{i=1}^{N-1} \sum_{j=2}^{N} \sum_{k=1}^{N_p} P_{ik} - (T_i^o \times)^{-1} T_j^o \times P_{jk} \qquad (2)$$

where N is the number of views in the cycle, N_p is the number of point correspondences between views i and j and T_i^o is the transformation matrix than aligns view i with respect to the first view in the cycle. This function is minimized by using Levenberg-Marquardt algorithm.

5 RESULTS AND DISCUSSION

In order to test our approach, real images are acquired with the 3D sensor developed in our laboratory (Matabosch et al., 2006). The goal of this sensor is to acquire 3D surfaces by means of a on-the-self camera and a stripe laser composed of 19 slits. The set-up lets us to acquire views from moving objects or acquire consecutive views while the sensor is manually displaced around the object, without any prior information about the pose.

As the goal of the experiments is to evaluate the accuracy of the registration process, the sensor is placed on a XYZ-translation table (see Figure 3). In this experiment the object of Figure 5a is used and 27 consecutive views are acquired.

Determining the transformation matrix that relates the coordinate system of the sensor with respects to the coordinate system of the table, the motion between consecutive views can be computed and compared to the motion obtained by the registration process.

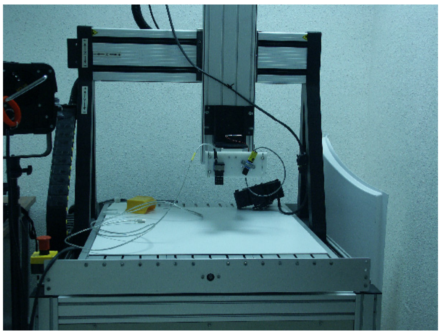
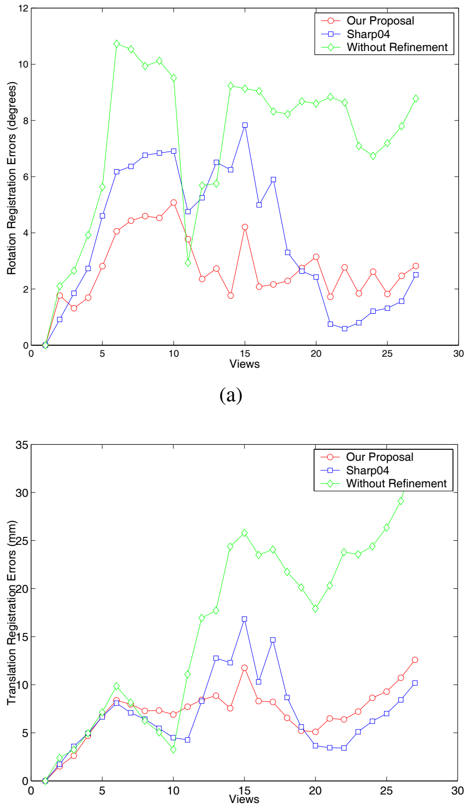


Figure 3: Set-up used in the experiments

Both translation and rotation errors are represented in Figure 4. Translation errors are obtained as the discrepancy between the real translation (given by XYZ-Table) and the estimated one (obtained by registration). Rotation errors can be analysed by comparing the angle between both real and estimated rotation axis and the discrepancy between the norm of both axis of rotation. Figure 4 shows that our method is suitable to reduce the propagation error in the presence of cycles. Although Sharp's method obtains better results at the end of the cycle (view 21), the error is worse distribute inside the view with respect to our approach. After this view, the error increases because no other cycle is found.

The complete reconstruction is shown in Figure 5, where an integration algorithm is applied to obtain a continuous surface without redundant information. The algorithm used is based on the Volumetric Integration method of Curless (Curless and Levoy, 1996).



(a)

(b)

Figure 4: Evolution of the registration errors: a) Rotation Errors; b) Translation Errors

6 CONCLUSIONS

In this paper, a survey of registration techniques is presented discussing the pros and cons among them. Furthermore, as most part of registration algorithms do not solve the problem of error propagation, some approaches are discussed and a new proposal is presented.

Our proposal is based on minimizing the registration errors between all views contained in a loop. A loop is detected by computing the translation vector between views. Then, in order to prove that a real loop exists, the overlapping between the first and the last view in the loop is computed. An approximation of the overlapping area is computed by means of the projections onto planes X-Y, X-Z and Y-Z with the aim of reducing the computing time.

When a loop is found, global error is minimized by using a multi-view registration algorithm based on Levenberg-Marquardt and point-to-plane correspondences.

Results show that errors are less important compared to the ones obtained by using traditional Pair-wise approach. Furthermore, as only views of the same cycle are simultaneously minimized, our approach obtains better accuracy in less computing time compared to a classic multi-view.

These experiments also show than our method obtain better results than the proposal of Sharp. This is because our proposal minimize the global registration error whereas Sharp's algorithm

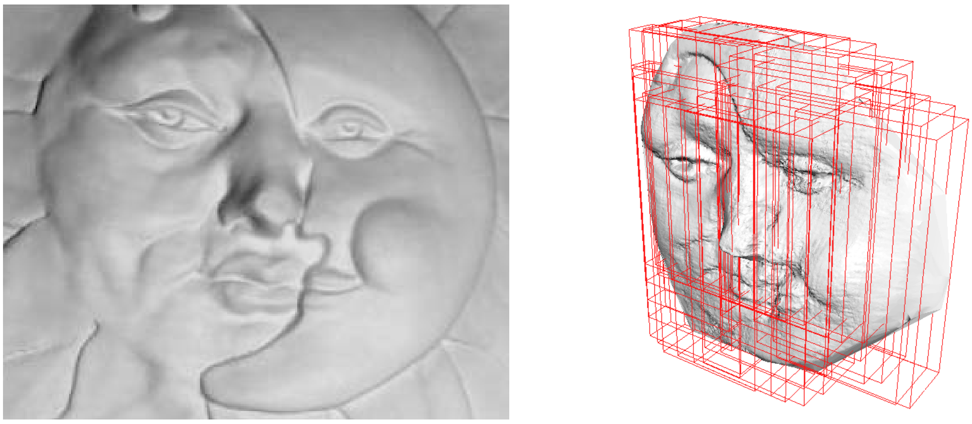


Figure 5: Complete registration of a real object: a) Picture of the object b) Final registration including bounding boxes of all 27 views acquired to obtain the final model

only force that the error between initial and final view of the cycle must be zero, then the error is distributed through the views of the cycle. On the other hand, this distribution does not require significant computation, obtaining final results in less time than our proposal.

Experimental results are done with real objects, obtaining both visual and quantitative good results.

REFERENCES

Armangué, X., Araújo, H. and Salvi, J., 2003. A review on egomotion by means of differential epipolar geomety applied to the movement of a mobile robot. Pattern Recognition 21(12), pp. 2927–2944.

Besl, P. and McKay, N., 1992. A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), pp. 239–256.

Chen, C.-S., Hung, Y.-P. and Cheng, J.-B., 1998. A fast automatic method for registration of partially overlapping range images. In: International Conference on Computer Vision, Bombay, pp. 242–248.

Chen, Y. and Medioni, G., 1991. Object modeling by registration of multiple range images. In: IEEE International Conference on Robotics and Automation, pp. 2724 –2729.

Chow, C., Tsui, H. and Lee, T., 2004. Surface registration using a dynamic genetic allgorithm. Pattern recognition 37(1), pp. 105–117.

Chua, C. J. R., 1997. Point signatures: A new representation for 3d object recognition. International Journal of Computer Vision 25(1), pp. 63–85.

Curless, B. and Levoy, M., 1996. A volumetric method for building complex models from range images. In: SIGGraph-96, pp. 303–312.

Forest, J. and Salvi, J., 2002. An overview of laser slit 3d digitasers. In: International Conference on Robots and Systems, Lausanne, pp. 73–78.

Gagnon, H., Soucy, M., Bergevin, R. and Laurendeau, D., 1994. Registration of multiple range views for automatic 3-d model building. In: Computer Vision and Pattern Recognition, pp. 581–586.

Godin, G., Laurendeau, D. and Bergevin, R., 2001. A method for the registration of attributed range images. In: 3DIM01 Third International Conference on 3D Digital Imaging and Modeling, Québec, Canada, pp. 179–186.

Greenspan, M. and Godin, G., 2001. A nearest neighbor method for efficient icp. In: Third International Conference on 3-D Digital Imaging and Modeling, Quebec City, Canada, pp. 161–168.

Johnson, A., 1997. Spin-images: A Representation for 3-D Surface Matching. PhD thesis, Carnegie Mellon University, USA.

Jost, T. and Hugli, H., 2002. A multi-resolution scheme icp algorithm for fast shape registration. In: First International Symposium on 3D Data Processing Visualization and Transmission, pp. 540– 543.

Kim, S., Jho, C. and Hong, H., 2003. Automatic registration of 3d data sets from unknown viewpoints. In: Worshop on Frontiers of Computer Vision, pp. 155–159.

Levoy, ., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J. and Fulk, D., 2000. The digital michelangelo project: 3D scanning of large statues. In: Siggraph 2000, Computer Graphics Proceedings, ACM Press / ACM SIGGRAPH / Addison Wesley Longman, pp. 131–144.

Masuda, T., 2001. Generation of geometric model by registration and integration of multiple range images. In: Third International Conference on 3-D Digital Imaging and Modeling, pp. 254 –261.

Masuda, T., 2002. Object shape modelling from multiple range images by matching signed distance fields. In: First International Symposium on 3D Data Processing Visualization and Transmission, pp. 439–448.

Matabosch, C., Salvi, J. and Forest, J., 2003. Stereo rig geometry determination by fundamental matrix decomposition. In: Workshop on European Scientific and Industrial Collaboration, pp. 405–412.

Matabosch, C., Salvi, J., Fofi, D. and Meriaudeau, F., 2006. A refined range image registration technique for multi-stripe laser scanner. In: Proceedings of SPIE - The International Society for Optical Engineering, Vol. 6090, San Jose, California, USA, pp. 246–253.

Park, S.-Y. and Subbarao, M., 2003. A fast point-to-tangent plane technique for multi-view registration. In: 3DIM, 4th International Conference on 3D Digital Imaging and Modeling, pp. 276–284.

Pulli, K., 1999. Multiview registration for large data sets. In: Second International Conference of 3-D Digital Imaging and Modeling, pp. 160–168.

Rusinkiewicz, S. and Levoy, M., 2001. Efficient variant of the icp algorithm. In: 3rd International Conference on 3-D Digital Imaging and Modeling, pp. 145–152.

Salvi, J., Pagès, J. and Batlle, J., 2004. Pattern codification strategies in structured light systems. Pattern Recognition 37(4), pp. 827–849.

Sharp, G., Lee, S. and Wehe, D., 2002. Icp registration using invariant features. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(1), pp. 90–102.

Sharp, G., Lee, S. and Wehe, D., 2004. Multiview registration of 3d scenes by minimizing error between coordinate frames. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(8), pp. 1037–1050.

Silva, L., Bellon, O. and Boyer, K., 2003. Enhanced, robust genetic algorithms for multiview range image registration. In: 3DIM03. Fourth International Conference on 3-D Digital Imaging and Modeling, pp. 268–275.

Stamos, I. and Leordeanu, M., 2003. Automated feature-based range registration of urban scenes of large scale. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 555 –561.

Trucco, E., Fusiello, A. and Roberto, V., 1999. Robust motion and correspondences of noisy 3-d point sets with missing data. Pattern Recognition Letters 20(9), pp. 889–898.

Wyngaerd, J. V., 2002. Automatic crude patch registration: Toward automatic 3d model building. Computer Vision and Image Understanding 87, pp. 8–26.

Zinsser, T. and Schnidt, J. Niermann, H., 2003. A refined icp algorithm for robust 3-d correspondences estimation. In: International Conference on Image Processing, pp. 695–698.

SEGMENTION OF DAMAGED BUILDINGS FROM LASER SCANNING DATA

Miriam Rehor and Hans-Peter Bähr

Institute of Photogrammetry and Remote Sensing (IPF), Karlsruhe University, Germany
rehor@ipf.uni-karlsruhe.de, baehr@ipf.uni-karlsruhe.de

Commission III/3

KEY WORDS: LIDAR, Segmentation, Triangulation, Statistics, Classification, Damaged Buildings

ABSTRACT:

Laser scanning- or ”LIDAR”-based systems show perfect performance during time-critical events, like data collection in disaster management. To this end the research reports from classification of damaged buildings, a special challenge, for which first results are given. Firstly, mathematical foundations compile some new insights which are specific for the mentioned task, like Bolzano’s theorem and statistical tests based on an extended Gauss-Markov model. Secondly, it is presented how these tools support the segmentation of planar surfaces and the classification of TIN segments into undamaged and damaged elements as well as into connecting triangles. Results are presented for real data from a training area of the Swiss Military Disaster Relief. The present status of the investigation shows that clear assumptions can be made for damaged buildings. Further steps will fuse additional knowledge in terms of data and algorithms.

1 INTRODUCTION

Disaster Management issues, unfortunately, are of growing importance worldwide. In any case of disaster, spatial data are the backbone for adequate decisions.

This is particularly true in case of time-critical situations, where the responsible experts have to make their decisions very fast with respect to save as many lives as possible. Therefore, image-based data acquisition including automatic image analysis procedures proves to be an excellent tool in a disaster environment.

More precisely, laser scanning (”LIDAR”) shows an ideal performance for such environments due to fast and geometrically precise data. At the IPF the technique has been analysed in the context of strong earthquakes since 1997 (Steinle & Bähr 1999; Vögtle & Steinle 2000). However, a general problem is the lack of ”real” laser scanning data from earthquakes or similar occasions: synthetic simulations of destroyed or damaged buildings would never fully reveal what might happen in reality.

This problem has overcome by a laser scanning flight of a camp from the Swiss Military Disaster Relief. The camp contains a complete collection of different types of destroyed or damaged building ensembles. The aim of this publication is to show the performance of laser scanning data for detection and classification of such an environment. This is a challenging new task which starts from the results for modelling undamaged buildings, which have been broadly published (see e.g. Kaartinen et al. 2005; Schwalbe et al. 2005; Steinle 2005).

Before starting, some basic terms have to be clarified. The overall aim is *classification* of damaged buildings recorded by laser scanning in the context of disasters, like earthquakes. Classification means to assign unknown patterns to a priori given classes. The classes are expressed by names (concepts). This is a very important observation, since concepts are by nature ambiguous (Bähr & Müller 2004; Bähr 2005).

The patterns to be classified are the result of a *segmentation* process of the LIDAR point clouds. Therefore, segmentation is a necessary step with respect to the following classification and means division of the point cloud into homogenous features. *Homogeneity* may be very diverse, like patches of similar colour, shape or orientation, like edges of similar length, width and mutual position and even like point clusters of given distribution. The features extracted in the segmentation process are, nota bene, without any semantics.

Finally, the term *model* needs some comments, since its use is often vague and not clearly defined. In the context of this work the model contains the knowledge (i.e. facts and rules) necessary for segmentation of the point cloud. Subsequently, *modelling* means formalising the physical world in order to make the data fit for reasoning.

2 SOME MATHEMATICAL FOUNDATIONS

2.1 Theory of Model Error Detection in Gauss-Markov Models

In order to check a Gauss-Markov model for model errors, the initial model given by

$$l + v = A\,\hat{x} \qquad \text{and} \qquad C_{ll} = \sigma_0^2\,Q_{ll} = \sigma_0^2\,P^{-1} \tag{1}$$

may be extended. To do this, p new unknowns y are introduced, which compensate gross errors from single observations or from groups of observations (see Baarda 1967; Baarda 1968; Heck 1985; Niemeier 2002).

While the stochastic model remains unchanged, the extended functional model is given by

$$l + \bar{v} = A\,\hat{\bar{x}} + B\,\hat{y}. \tag{2}$$

The matrix B describes the influence of the new parameters y on the observations. Since the residuals and the estimates of the unknowns change in relation to the initial model, these values are written as $\bar{v}$ and $\hat{\bar{x}}$. The estimates of the new unknowns are collected in the vector $\hat{y}$. The extension $B\,\hat{y}$ may be regarded as an improvement of the initial model. The redundancy $\bar{r}$ of the extended model is given by

$$\bar{r} = \dim(l) - \big(\dim(\hat{\bar{x}}) + \dim(\hat{y})\big) = r - p \tag{3}$$

where r is the redundancy of the initial model.

If B is column-regular, the weighted square sum of the residuals $\bar{\Omega} = \bar{v}^T P \bar{v}$ of the extended model follows from the weighted square sum of the residuals $\Omega = v^T P v$ of the initial model:

$$\bar{\Omega} = \Omega - \hat{y}^T Q_{yy}^{-1} \hat{y} = \Omega - \Delta\Omega \tag{4}$$

with

$$\hat{y} = -Q_{yy} B^T P v \qquad \text{and} \qquad Q_{yy} = (B^T P Q_{vv} PB)^{-1}$$

This equation shows clearly that the model extension leads to a reduction of the weighted sum of the squares of the residuals. The extension makes sense only if the square sum $\bar{\Omega}$ of the extended model becomes significantly smaller than the respective value from the initial model (Ω). This is precisely the case if $\Delta\Omega$ is significantly larger than zero.

This case may be checked by means of a parameter test. The belief that the model errors are not significant (i.e. the initial model does fit to the physical reality) corresponds to the null hypothesis H_0. The alternative hypothesis H_A, on the other hand, assumes that model errors do exist and therefore the extended model has to be accepted. In order to test whether the model errors are significant, the weighted square sum $\Delta\Omega$ may be compared to the a priori variance σ_0^2 or to the a posteriori variance $\hat{\bar{\sigma}}^2$ of the extended model (Niemeier 2002).

The corresponding test statistics are

$$T_1 = \frac{\hat{y}^T Q_{yy}^{-1} \hat{y}}{\sigma_0^2} \quad \sim \chi^2_{(p,\lambda)} = p \cdot F_{(p,\infty,\lambda)} \tag{5}$$

and

$$T_2 = \frac{\hat{y}^T Q_{yy}^{-1} \hat{y}}{\hat{\bar{\sigma}}^2} \quad \sim p \cdot F_{(p,r-p,\lambda)} \tag{6}$$

respectively. If $y = E(\hat{y})$ is the expectation of $\hat{y}$,

$$\lambda = \frac{y^T Q_{yy}^{-1} y}{\sigma_0^2} \tag{7}$$

is the non-centrality parameter of the (non-central) Fisher distribution and vanishes if H_0 is valid.

2.2 Segmentation of Planar Surfaces

The selected approach for modelling buildings is based on the assumption that undamaged buildings may be represented by planar surfaces. For the extraction of planar surface elements a region growing algorithm is used, taking 2.5D raster data. The starting point for any surface segment is a seed region which fulfils the condition that the n assigned points are approximately located in a plane. The parameters of this plane are determined by least squares adjustment. Owing to only 3 unknowns ($\hat{a}_0, \hat{a}_1, \hat{a}_2$), the solution of the adjustment may be given straightforward by the well known expressions (x_i, y_i: position coordinates; $g(x_i, y_i)$: height):

$$g(x_i, y_i) + v_i = \hat{a}_0 + x_i\,\hat{a}_1 + y_i\,\hat{a}_2 \tag{8}$$

$$\begin{pmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \hat{a}_2 \end{pmatrix} = \begin{pmatrix} n & \sum_{i=1}^n x_i & \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i & \sum_{i=1}^n x_i y_i & \sum_{i=1}^n y_i^2 \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n \; g(x_i, y_i) \\ \sum_{i=1}^n \; g(x_i, y_i)\; x_i \\ \sum_{i=1}^n \; g(x_i, y_i)\; y_i \end{pmatrix} \tag{9}$$

After determination of the seed region any of the adjacent pixels which were not yet assigned to a surface segment is tested whether it fulfils the planar equation under concern. To this end, in a first step the plane is computed again, based on the enlarged set of points. In a second step it is decided by a global test, involving

$$T_{glob} = \frac{v^T P v}{\sigma_0^2} \;\sim\; \chi_r^2 \;, \tag{10}$$

if the null hypothesis can be accepted. v is the column vector of least squares residuals resulting from the enlarged set of points.

Moreover, the model error detection method described in chapter 2.1 is taken in order to check whether the tested point might show a gross error.

In case of rejection of the null hypothesis by either the global test or the model error detection method, the model contains an unacceptable error. As the model was ok before adding the point, the conclusion is that this point does not fit to the plane. The results of the segmentation procedure are stored in a so-called *label image* from where they are taken for further processing.

2.3 Bolzano's Theorem for Retrieving Vertical Planes

By the segmentation of planar surfaces vertical planes cannot be extracted. Therefore, the neighbourhood between the planes may only be determined, for a first step, in relation to the projection of the segments onto the ground plane. By this procedure it is impossible to recognise if adjacent planes do really intersect at their border lines. Due to this situation for any contour line recovered from the ground plane it is necessary to test if a vertical break exists between the two neighbouring segments. In case of detection of such a vertical break, a vertical plane has to be inserted in the course of the border line in order to form a consistent building model.

Based on this procedure it is tested if adjacent planes do really intersect in the vicinity of a common border line. Because of linear height change along the border line it is sufficient to test in the endpoints if there is a significant height difference between the two planes.

The methodology is based on Bolzano's theorem (see Bronstein et al. 2001 (p.61)):

> If a function $f(x)$ is defined and continuous on a closed interval $[a, b\,]$ and the values of the function in the endpoints of the interval $f(a)$ and $f(b)$ have different signs, then there exists at least one value c, where $f(x)$ is zero:
>
> $$f(c) = 0 \qquad \text{for} \qquad a < c < b.$$
>
> Geometrically spoken, the curve of a continuous function intersects the x-axis at least one time at the transition from one side of the x-axis to the other.

This mathematical theorem is graphically explained and subsumed to the case of segmentation of planes in Figure 1. For any border line of the two planes A and B exist exactly two vertical planes, which respectively contain one of the two endpoints P_A and P_E of the border line and which are orthogonal to that line. By intersecting the two vertical planes with the planes A and B, the four straight lines g_{A_A}, g_{B_A}, g_{A_E} and g_{B_E} are created. More precisely, the lines g_{A_A} and g_{B_A} result from the intersection with the vertical plane which contains P_A, whereas the lines g_{A_E} and g_{B_E} result from the intersection with the vertical plane which contains P_E.

This height checking test investigates whether the straight lines which result from intersection with one of the vertical planes do intersect within a given interval. The value k that defines the interval may be chosen arbitrarily, but it controls the velocity of the procedure and the quality of the final results.

As a straight line is a continuous function, the distance between two straight lines again is continuous. Therefore, instead of testing intersection of two lines within a particular interval, it may be asked whether the function which results from the distance between the two lines shows a zero point within this interval.

In order to determine if the lines g_{A_A} and g_{B_A} do meet in the interval which is given by P_{A1} and P_{A2} it has to be tested after Bolzano's theorem if the differences d_{A1} and d_{A2} from the functions of both lines show the same sign. If so, the two lines do not meet within the given interval and vice versa (see Figure 1 (b) and (c)).

If both the lines g_{A_A}, g_{B_A} and g_{A_E}, g_{B_E} do intersect within the intervals defined by the points P_{A1}, P_{A2} and P_{E1}, P_{E2}, respectively, the trace of the resulting line in the ground plane is within the red domain of Figure 1 (a). This means that it nearly matches the estimated border line. Consequently, the intersection of this line with other edges will lead to points in the estimated locations. Therefore, in such cases no vertical plane has to be inserted. However, if only one of the two line pairs intersects in the given interval or if no intersection can be determined at all, the vertical gap between the segments A and B in the vicinity of the border line is too large and the intersection is rejected. To make the model consistent, a vertical plane has to be introduced. The equation of this plane may easily be determined by the coordinates of the points P_A and P_E together with the vertical condition.

3 FROM SEGMENTS TO CAD MODELS FOR UNDAMAGED BUILDINGS

In order to do the step from single segments to CAD models after completion of the surface segmentation, an analysis of the neighbourhood conditions of the surfaces has to be performed. For this reason by means of morphological operators and starting from the *label image* (see par. 2.2), it is checked in a 2D space which surface segments are adjacent (according to (Steinle 2005)). In a next step the border lines between neighbouring segments are determined. For the itinerary of the border lines it is tested if breaks between adjacent segments do occur, a case which would demand the insertion of a vertical plane. For this procedure the described approach based on Bolzano's theorem is taken (see par. 2.3).

After the topological relations between the single surfaces are known, the lines which describe the edges of the buildings may be determined from intersections of the planes. The topology of the edges is derived indirectly from the neighbourhood conditions of the surfaces, since the edges are not determined directly from

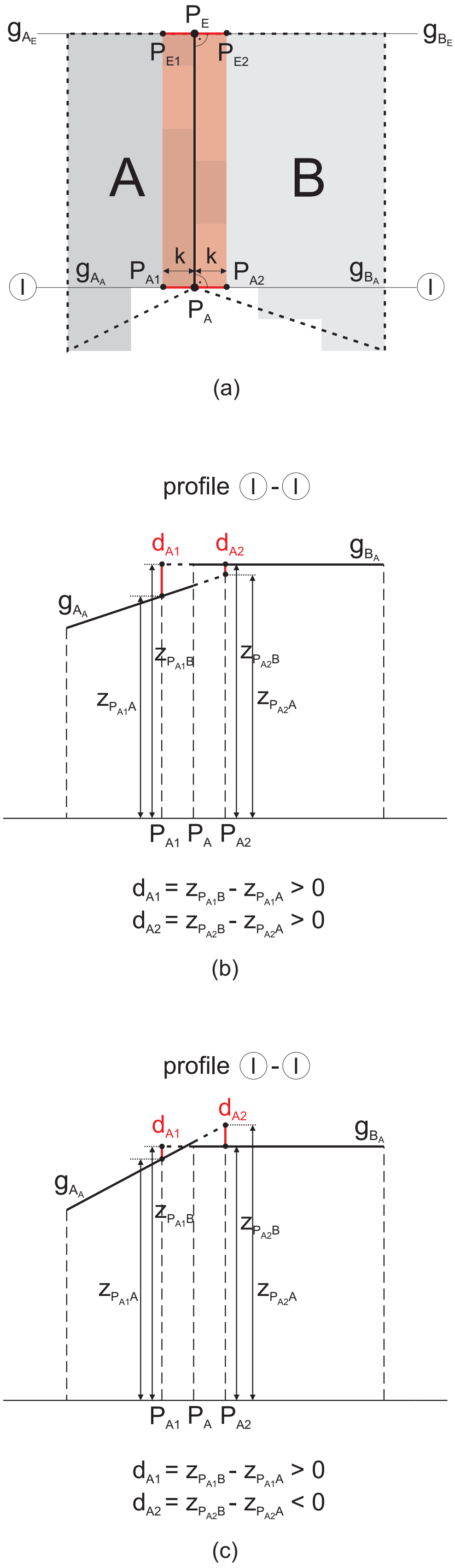


Figure 1: Height checking based on Bolzano's theorem; (a) Ground plane; (b) Intersection along profile I - I in case that the straight lines g_{A_A} and g_{B_A} do not intersect in the given interval; (c) Intersection along profile I - I in case that the straight lines g_{A_A} and g_{B_A} do intersect in the given interval

the original data but from plane intersections. The corner points of the buildings are the result of intersecting edges.

Since the geometrical primitives and the corresponding topology is known, it is possible to construct a wireframe model automatically. Commercial visualisation software shows certain constraints and therefore does not fully allow presenting such a wireframe model. One of the limitations is that surfaces can be displayed only if they are composed of 3 or at maximum of 4 points. Since the surfaces of buildings are generally made of more than 4 points, it is required to cut the surfaces into subsurfaces (e.g. triangles). To this end in a first step the border polygons of single building surfaces are determined. Afterwards it is tested if the polygons are ”simple polygons” what means that ”non adjacent edges” do not contain common points. For all ”non simple polygons” a reduction to simple polygons is mandatory, e.g. after Sunday's method (Sunday 2005).

The polygons then have to be cut into triangles, what in our case is performed by a Constrained Delaunay Triangulation. It must be pointed out that a bordering polygon of a roof surface may contain another polygon of the same surface completely. This e.g. happens in case of garrets which appear within a roof surface. The central polygon then has to be excluded from the triangulation. An example for an automatically generated building model is shown in Figure 2.

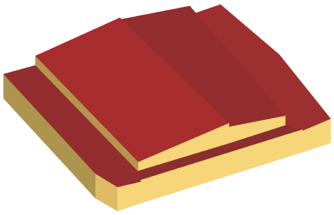


Figure 2: Automatically generated building model

4 CLASSIFICATION OF DAMAGED BUILDINGS

Classification of damages in buildings affected by earthquakes presents a key research topic at Karlsruhe University since a decade (Steinle & Bähr 1999). The solutions all have to be based on comparing pre- and post event structures of the buildings under investigation. In a first step appropriate models for describing damages from the laser scanning data have to be set up.

Damaged buildings may show very different damage types. The types to be discriminated are summarised in a damage catalogue as shown in Figure 3. In detail, for each damage type descriptions and geometrical characteristics are assigned. As geometry is concerned, features are e.g. differences in height and volume as well as changes of the inclinations of the building's surfaces (Schweier & Markus 2004).

Therefore, modelling damaged buildings has to take into account such geometrical features which characterise the respective damage types well. The sequence of the approach is given in Figure 4. In a first step planar surfaces are segmented (see par. 2.2) in order to answer questions about change of inclinations and about size of the registered surfaces. In case of strong damages the segmentation results in many small surface elements and many non-segmented pixels. If a reference model of the undamaged building structure is available, an estimation is possible, whether damage occurred or not. This may be done by comparing number and size of the surfaces under concern. If no reference model is available, speculation must be done most carefully (e.g. many small surfaces might represent a dome).

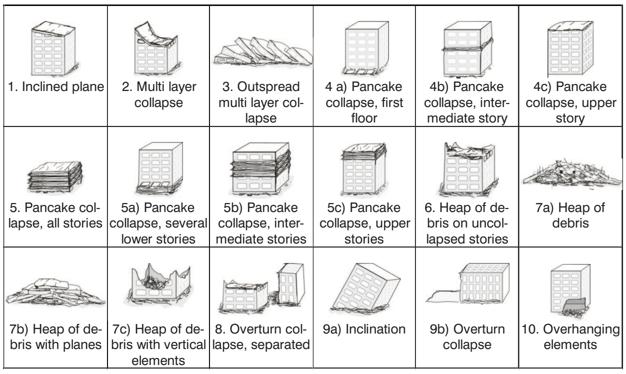

1. Inclined plane
2. Multi layer collapse
3. Outspread multi layer collapse
4 a) Pancake collapse, first floor
4b) Pancake collapse, intermediate story
4c) Pancake collapse, upper story
5. Pancake collapse, all stories
5a) Pancake collapse, several lower stories
5b) Pancake collapse, intermediate stories
5c) Pancake collapse, upper stories
6. Heap of debris on uncollapsed stories
7a) Heap of debris
7b) Heap of debris with planes
7c) Heap of debris with vertical elements
8. Overturn collapse, separated
9a) Inclination
9b) Overturn collapse
10. Overhanging elements

Figure 3: Compilation of the damage types (Schweier & Markus 2004)

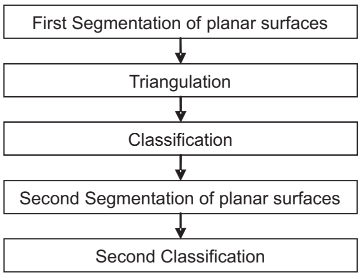

First Segmentation of planar surfaces
Triangulation
Classification
Second Segmentation of planar surfaces
Second Classification

Figure 4: Workflow of the modelling of damaged buildings

After the surface segmentation a planar Delaunay-based TIN in 2.5D is produced. For the definition of the mesh points the results from the surface segmentation have to be taken into account. Therefore, points have to be selected which guarantee that the produced triangles match the adjusted planes sufficiently in segmented areas. This is the case if, for any segmented pixel, a grid point is produced with position coordinates x_P and y_P equal to the pixel coordinates and with height z_P computed in such a way that the point exactly matches the extracted plane. This height may be derived from the general equation of a plane:

$$z_P = (d - a\,x_P - b\,y_P)/c \qquad (11)$$

As the non segmented points of course have to be integrated into the triangulation, too, for each pixel which was not assigned to a surface segment a point is added to the number of the mesh points, whereas its coordinates are taken from the respective pixel. For reducing the number of the created triangles, only the non segmented points and the border points of the surface segments are accepted as mesh points. Figure 5 shows the TIN of an area with damaged buildings.

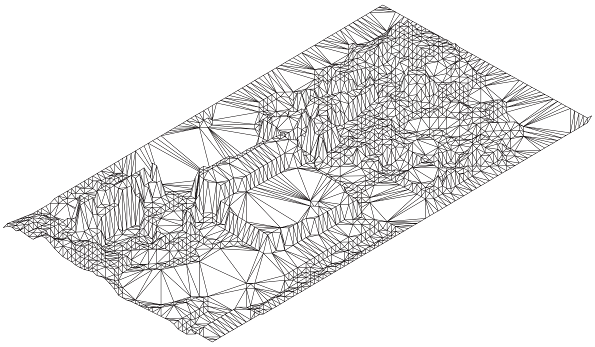


Figure 5: Produced TIN of an area with damaged buildings

Because of the ambiguity of the 2D Delaunay Triangulation in a raster domain, the TIN produced this way is not fully clear as it may lead to different results in case of regular distribution of the grid points. A 3D approach would improve this situation.

After creating the TIN the triangles have to be classified according to Figure 4. For each triangle it is tested, whether its corners were assigned by the surface segmentation process to the same segment i.e. whether the triangle is located in the associated plane. If this is true, the triangle represents a part of the assigned plane and is added to the class of *plane triangles*. It may happen that the corners of a triangle are not part of a single plane. For this type of triangle the term *planes connecting triangle* is used because it is representing a connection between two or three planar surfaces. If a triangle contains just one point which was not segmented it is classified as *debris triangle*. For such triangles the probability exists that they represent strongly damaged building parts.

Narrow shaped planar surfaces are not registered, because in the surface segmentation process a new surface segment may be created only if a seed region is found which shows a minimum area (e.g. 3 x 3 pixels) and fulfils a certain precondition (see par. 2.2). Therefore it may happen e.g. that side roofs or parts of ton-shaped roofs are represented by *debris triangles* (see par. 5).

To avoid this problem, a second segmentation is executed starting from the *debris triangles* (Figure 4). In this process it is looked for several neighbouring *debris triangles* lying approximately in a plane. The used approach is similar to the first segmentation based on raster data (see par. 2.2). The starting point is built by a *debris triangle*. First of all the parameters of the plane defined by the three points of this triangle are calculated. Afterwards it is tested for any of the neighbouring triangles if it concerns a *debris triangle*. If this is the case a regression plane is calculated through the points of the initial triangle and the points of the currently examined triangle. In order to check the correctness of the used model a global test (eq. (10)) and a test for blunders (eq. (5) and (6)) are carried out. Is the model accepted by both tests the triangles are lying approximately in a plane. So the examined triangle is assigned to the new segment. Is the assumption rejected the triangle is not assigned to the new segment and the plane parameters are reset. In both cases the next adjacent triangle is looked at. In the further steps the regression plane is calculated through the points of all triangles that have already been assigned to the new segment and the point of the momentarily considered triangle that does not belong to one of the other triangles. If no further adjacent triangle can be found that fulfils the requirements it is tested how many triangles have been assigned to the segment. If the number is less than a given number the area of the segment is regarded as too small and therefore the segment is deleted.

After the second segmentation, the triangles of course have to be classified once more. Now, two new classes are introduced: *segment triangles* and *segment/planes connecting triangles*. The first mentioned class represents newly detected surface segments. The second class contains triangles which connect new detected segments or new and old ones.

5 RESULTS

It has to be highlighted that the classification approach for damaged buildings was tested by real laser scanning data in an area of physically damaged buildings (i.e. no simulation!). The test area is a training field from the Swiss Military Disaster Relief (Figure 6). It has an extension of about 500 m × 800 m and is used for training rescue and support during catastrophic events. The original data were acquired by TopoSys Company in 2004 and transformed into digital surface models of 1 m raster width. The precision of these models is in the order of ± 0.5 m in position and ± 0.15 m in height.

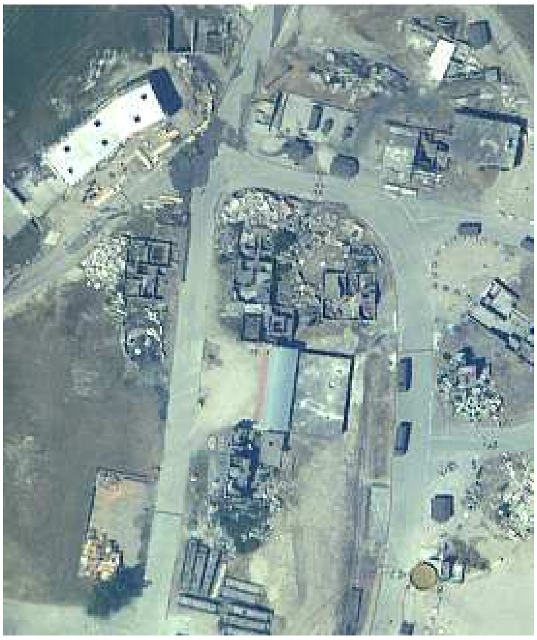


Figure 6: Aerial photograph of the test area

Figure 7 displays how a side roof of a building and a ton-shaped roof (a) look before (b) and after (c) the second surface segmentation step. The *debris triangles* are shown in red, the *planes connecting triangles* in dark and the *segment/planes connecting triangles* in light grey. Each of the extracted segments is displayed in a different colour. It is obvious, that by the first step neither the side roof nor parts of the ton-shaped roof were segmented correctly. Consequently, they are shown as debris in Figure 7 (b). After the second segmentation step the corresponding surfaces are assigned to the new surface segments.

Figure 8 shows the model of a larger area, where each surface segment is displayed by a different colour. The area contains both the building from Figure 7 of pancake collapse type and some heaps of debris. Besides, two trucks are imaged and marked by black circles. The discrimination between debris on the one hand and obsolete information like the trucks (”perturbations”) on the other hand plays an important role in classification of damage types. Elements like the trucks, taken as building components, do inevitably lead to misclassifications. The example in Figure 8 clarifies, that the trucks may not be discriminated from the further debris structures without introducing additional knowledge.

A CAD model for the area shown in Figure 6 is given in Figure 9. This example shows that, supported by such a model, estimations at high probability are feasible for areas where strong damages exist and for areas where the buildings will probably not show major damages.

6 CONCLUSIONS

Modelling undamaged buildings by laser scanning is nearly operational (Kaartinen et al. 2005), whereas segmentation and classification of damaged buildings is a new challenging task. Laser scanning obviously is an ideal tool for developing fast automatic real-time procedures, e.g. in the context of rescue in a disaster environment. First results are presented which use extended clas-

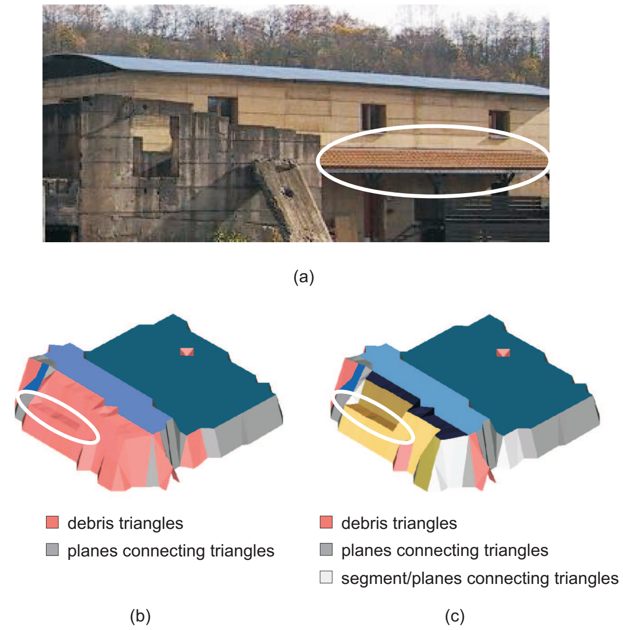

(a)
🟥 debris triangles
▨ planes connecting triangles
(b)
🟥 debris triangles
▨ planes connecting triangles
□ segment/planes connecting triangles
(c)

Figure 7: (a) A building with a side roof and a ton-shaped roof in a classified TIN (b) before and (c) after the second segmentation step. Each of the extracted segments is displayed in a different colour.

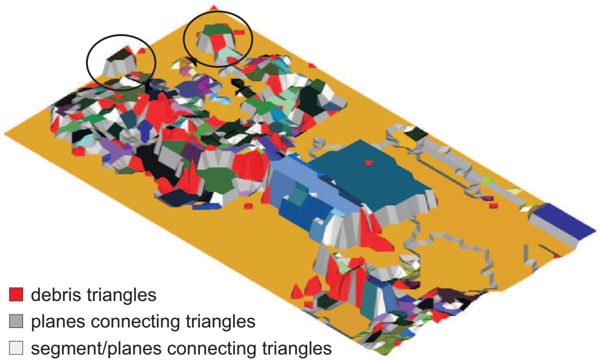

🟥 debris triangles
▨ planes connecting triangles
□ segment/planes connecting triangles

Figure 8: Model containing damaged buildings and ”perturbations” (trucks marked by black circles). Each segment is shown by another colour.

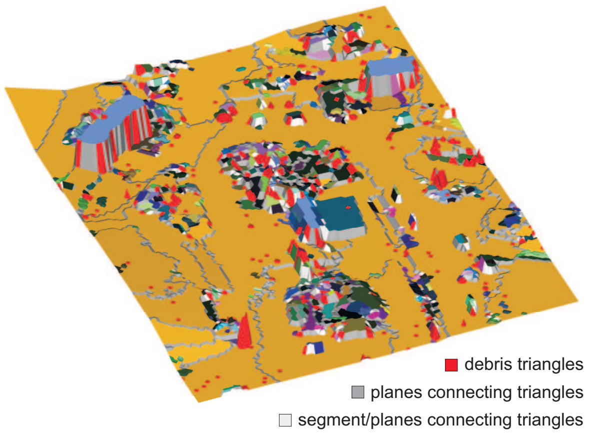

🟥 debris triangles
▨ planes connecting triangles
□ segment/planes connecting triangles

Figure 9: Model of the test area containing damaged buildings. The segments are shown in different colours.

sical approaches known from modelling of undamaged buildings and which make clear that damaged parts may be discriminated from undamaged parts of buildings.

The results are encouraging to further extend the approach: As far as the algorithms are concerned, much more should be possible to extract from the TIN than just geometrical parameters, especially when modelling in 3D instead of 2D. For instance, neighbourhood and shape are important additional features to take into consideration. Besides refinement of the algorithms additional data is expected to strongly improve the results. The step to fuse multispectral scanner and laser scanning data suggests itself as today the flights do provide both.

ACKNOWLEDGEMENTS

The authors would like to thank the German Research Foundation, Collaborative Research Centre 461 (DFG SFB 461) for financial support and the Swiss Disaster Relief Coordination and Control Center DDPS for providing their facilities.

REFERENCES

Baarda, W., 1967. Statistical concepts in geodesy. Publications on geodesy 2(4), Netherlands Geodetic Commission.

Baarda, W., 1968. A testing procedure for use in geodetic networks. Publications on geodesy 2(5), Netherlands Geodetic Commission.

Bähr, H.-P. and Müller, M., 2004. Graphics and Language as Complementary Formal Representations for Geospatial Descriptions. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences XXXV, Part B, Comm. 4, Istanbul, Turkey, pp. 216-221.

Bähr, H.-P., 2005. Sprache - ein Datentyp der Bildanalyse. In: H.-P. Bähr and T. Vögtle (eds), Digitale Bildverarbeitung - Anwendungen in Photogrammetrie, Fernerkundung und GIS, Wichmann Verlag, Heidelberg, 4. edn., pp. 211-228.

Bronstein, I. N., Semendjajew, K. A., Musiol, G. and Mühlig, H., 2001. Taschenbuch der Mathematik. Unchanged reprint of the 5. edn., Verlag Harri Deutsch, Thun and Frankfurt/Main.

Heck, B., 1985. Ein- und zweidimensionale Ausreißertests bei der ebenen Helmert-Transformation. Zeitschrift für Vermessungswesen 110(10), pp. 461-471.

Kaartinen, H., Hyyppä, J., Gülch, E., Vosselman, G., Hyyppä, H., Matikainen, L., Hofmann, A.D., Mäder, U., Persson, Å., Söderman, U., Elmqvist, M., Ruiz, A., Dragoja, M., Flamanc, D., Maillet, G., Kersten, T., Carl, J., Hau, R., Wild, E., Frederiksen, L., Holmgaard, J. and Vester, K., 2005. Accuracy of 3D city models: EuroSDR comparison. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVI, Part 3/W19, Enschede, The Netherlands, pp. 227-232.

Niemeier, W., 2002. Ausgleichungsrechnung. Walter de Gruyter, Berlin.

Schwalbe, E., Maas, H.-G., Seidel, F., 2005. 3D building model generation from airborne laserscanner data using 2D GIS data and orthogonal point cloud projections. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVI, Part 3/W19, Enschede, The Netherlands, pp. 209-214.

Schweier C. and Markus M., 2004. Assessment of the search and rescue demand for individual buildings. In: Proceedings of the 13th World Conference on Earthquake Engineering, Vancouver, Canada.

Steinle E. and Bähr H.-P., 1999. Laserscanning for change detection in urban environment. In: O. Altan and L. Gründig (eds), Proceedings of the Third Turkish-German Joint Geodetic Days - Towards A Digital Age, Volume I, Istanbul, Turkey, pp. 147-156.

Steinle, E., 2005. Gebäudemodellierung und -änderungserkennung aus multitemporalen Laserscanningdaten. PhD dissertation, Deutsche Geodätische Kommission, Reihe C, Heft Nr. 594, Verlag der Bayrischen Akademie der Wissenschaften, München, http://129.187.165.2/typo3_dgk/docs/c-594.pdf (accessed 28 Feb. 2006).

Sunday, D., 2005. Intersections for a 2D Set of Segments. http://www.geometryalgorithms.com/Archive/algorithm_0108/algorithm_0108.htm (accessed 19 Dec. 2005).

Vögtle, T. and Steinle, E., 2000. 3D Modelling of Buildings Using Laser Scanning and Spectral Information. International Archives of Photogrammetry and Remote Sensing XXXIII, Part B3, Amsterdam, The Netherlands, pp. 927-934.

3D ROAD-MARK RECONSTRUCTION FROM MULTIPLE CALIBRATED AERIAL IMAGES

Olivier Tournaire[1,2], Nicolas Paparoditis[1], Franck Jung[3], Bernard Cervelle[2]

[1]IGN / MATIS, 2-4, Ave Pasteur, 94165 S^t-Mandé - France
[2]UMLV, 5, Bd Descartes, Champs-sur-Marne 77454 Marne-la-Vallée CEDEX 2 - France
[3]ESGT, 1, Bd Pythagore, 72000 Le Mans - France

{olivier.tournaire;nicolas.paparoditis}@ign.fr franck.jung@esgt.cnam.fr bernard.cervelle@univ-mlv.fr

KEY WORDS: 3D reconstruction, Road-marks, Object extraction, Template matching, Correlation, Hierarchical estimation

ABSTRACT:

A method for accurate 3D reconstruction of road features from multiple calibrated aerial images of urban areas is proposed in this paper. We here focus on road-marks and in particular on zebra-crossings and discontinuous road-marks separating circulation lanes. The approaches used here are generic and based on a-priori external knowledge and thus constrain the extraction of image features. As we will explain, two strategies are adopted depending on the object size. For zebra-crossings, we first build 3D segments representing stripes' borders by 2D segments matching. For discontinuous lanes, we build a graph describing the network in each image and then match nodes in order to obtain 3D position of stripes' centers. This provides in both cases an initial solution in 3D space. Using geometric and radiometric modeling to obtain a set of plausible models, e then look for an optimal solution. The last step yields us to choose the best one in adequacy with images data. A correlation based energy and template matching strategy achieve this in a hierarchical frame. The algorithm is finally evaluated with ground control points surveyed with a millimetric precision.

1 INTRODUCTION

Most of the photogrammetric research on object extraction from aerial images in the last years has focused on building reconstruction. However, the road network is extremely structuring for urban scene analysis and for defining possible building ROIs. In addition, in 3D city models, roads and pavements should need to be described as well as buildings, thus needing a surfacic representation and a decimetric and geometric accuracy instead of the classical linear spaghetti model encountered in most of Road GIS databases. In this scope, (Vosselman, 2003) proposed a 3D road reconstruction from LASER points cloud and a cadastral map. For these applications, road-marks are very interesting descriptors of the road surface architecture. Semantic and functional informations can be derived from them: way of circulation, number of lanes, special lanes (public transport, ...). They can be used in numerous applications such as cartographic road databases updating (Zhang, 2003), road extraction (Hinz and Baumgartner, 2002; Steger et al., 1997) or creation of visual landmarks used in autonomous navigation systems (Royer et al., 2006).

Concerning ground-based imagery, many papers were published and various approaches are used. (Se and Brady, 2003) detect zebra-crossings for outdoor aid navigation for the partially sighted using vanishing lines. (Rebut et al., 2004) proposed a method for road marks analysis with mathematical morphology and a training database. For an automatic road marking repainting tool, (Charbonnier et al., 1997) designed an algorithm analysing segments by pairs. In real time driver assistance (Enkelmann et al., 1995) introduced a method using parallel segments and radiometric features in order to detect marking lanes.

The link between aerial and terrestrial imagery has become more and more important in the last years. It is crucial for instance for urban environments reconstruction problematics such as georeferencing and / or matching of images produced by mobile mapping systems (MMS) or to texture 3D models obtained from aerial imagery (Pénard et al., 2006). Most of the problems encountered by MMS lies in the fine and robust absolute localisation of the vehicle. Direct georeferencing methods such as GPS combined with INS and / or other sensors (odometers, gyroscopes, ...) are often used. However, in dense urban areas, GPS masks, multi-path errors and bad satellites configurations are extremely frequent. These errors cannot be fully corrected with an INS due to its relative drift on long distances providing an absolute accuracy from 0.5 m to 1 m. Thus, to provide an accurate georeferencing, we have to deal with external data to introduce constraints on the position. A strategy is to integrate in the system aerial images georeferenced with a bundle adjustement. Images then become the key-frame for obtaining absolute localisation by matching shapes detected from the two points of view.

In France, zebra-crossings, and more generally road marks are (in most cases) governed by careful specifications[1]. Moreover, these kinds of objects can be considered as invariants with a simple shape not suffering from generalisation, e.g to match aerial and ground based images or for the generation of landmarks databases for autonomous navigation.

This paper describes robust and accurate road-mark detection and reconstruction experts that can be helpfull for all previously described applications. We will not at all describe the reconstruction of the road network topology which could be in any case be extracted from medium-scale existing databases (at least in Europe and North America) but only describe two road-mark experts that could be helpfull to derive higher level information in a more complete system. The paper is organised as follows. A first part presents the algorithm for 3D zebra-crossing reconstruction. A second one is focused on the 3D reconstruction of dashed lanes. We then present in a third section a hierarchical method for refining the 3D position of the detected objects. Finally, we present briefly numerical results and evaluations.

2 ZEBRA-CROSSING RECONSTRUCTION

We first choose to reconstruct zebra-crossings because they strongly structure the road network in urban areas. Moreover, they are the

[1] Source: Ministère de l'Intérieur et Ministère de l'Equipement, de l'Aménagement du Territoire et des Transports: *Instruction interministérielle sur la signalisation routière*. 1988.

objects covering the greatest surface.

2.1 Zebra-crossing specifications

The specifications show that pedestrian walkaways have a fixed width of 0.5 m. The length of each stripe is only described in urban areas by a minimal size of 2.5 m. Two consecutive stripes are separated by a distance in the range $[0.5m; 0.8m]$, but is regular for a zebra-crossing. Finally, the stripes are white on a black background, but in special cases, like pedestrian areas, the hue can be inversed, or the background can be colored. Zebra-crossings have most of the time four to twenty stripes, and their maximum length is around 6 m.

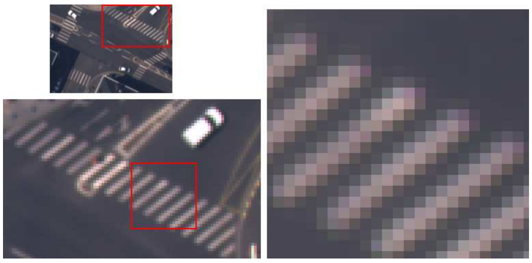


Figure 1: Extracts of a 4000×4000 digital image in Amiens (25 cm ground pixel)

2.2 Zebra-crossing extraction

Our strategy relies on 2D segments lines image features. We use the Canny-Deriche edge detector (Deriche, 1987). The images are oversampled by a factor 2 to have a better sampling of the convolution filter, and α is set to 1.5 to handle a compromise between localisation and sensitivity to noise. A hysteresis threshold is then processed, followed by subpixelar localisation of each contour point. Finally, chaining of contour points and polygonalisation is performed by the Douglas-Peucker algorithm (Douglas and Peucker, 1973). We now have 2D segments, with the knowledge of their covariance matrix in (ρ, θ) polar coordinates (Deriche et al., 1991).

In order to find zebra-crossings' segments, we analyse their relative organisation, and use specifications. First, segments are filtered on their length, taking in account a tolerance error. After this, we search for parallel groups of segments (with a tolerance taking in account the angular variance) respecting stripes size and distance between stripes. The homogeneity of length is equally computed, thus following again specifications. Finally, we retain objects that have at least six segments.

2.3 3D segments reconstruction

This 2D processing provides a set of segments belonging to zebra-crossings. We now build 3D segments with the detected structures in the images. For 3D segments reconstruction, we choose a true multi-image matching algorithm of sweep-planes (Collins, 1996) (more details can be found in (Taillandier, 2004)). Here, we introduce an external data - a DSM computed by image matching (Pierrot-Deseilligny and Paparoditis, 1996) - to limit search space to cut down combinatory. The DSM is morphologically dilated (to define an upper and lower bounding surface) and the object space is discretised in voxels. The sweep step and the cells' size are defined with respect to the flight parameters.

With this sweep-plane technique, we obtain for each voxel segments correspondences between each images. To reconstruct a 3D segment from a match, we use a two step minimisation procedure. We first construct a set of 3D segments by intersecting two by two all the pairs of planes within the set (see Figure 2 and 3). Each plane is defined by the center of projection of the camera and goes through the image straight line. Each segment of a set defines a $(P_i, \vec{u_i})$ 3D line. The final 3D segment lies on the line whose direction $\vec{v}$ minimises in a robust way the sum of angular difference with all the segments of a given set (see Equation 1). Using a least squares minimisation, it leads to find the vector $X = \vec{v} = (x, y, z)^t$ by solving the system $A^tAX = 0$ where A^tA is the 3×3 matrix defined in Equation 2. Vector X is finally obtained by extracting the eigenvector corresponding to the smallest eigenvalue of A^tA. Note that the normalisation of the $\vec{u_i}$ leads to the constraint $\|X\| = 1$.

$$\underset{\vec{v}}{argmin} \sum_i \sin^2(\vec{v}, \vec{u_i}) \equiv \underset{\vec{v}}{argmin} \sum_i \left\| \vec{v} \wedge \vec{u_i} \right\| \tag{1}$$

$$A^tA = \begin{pmatrix} \sum\limits_i (u_{i_y}^2 + u_{i_z}^2) & -\sum\limits_i u_{i_x} u_{i_y} & -\sum\limits_i u_{i_x} u_{i_z} \\ -\sum\limits_i u_{i_x} u_{i_y} & \sum\limits_i (u_{i_x}^2 + u_{i_z}^2) & -\sum\limits_i u_{i_y} u_{i_z} \\ -\sum\limits_i u_{i_x} u_{i_z} & -\sum\limits_i u_{i_y} u_{i_z} & \sum\limits_i (u_{i_x}^2 + u_{i_y}^2) \end{pmatrix} \tag{2}$$

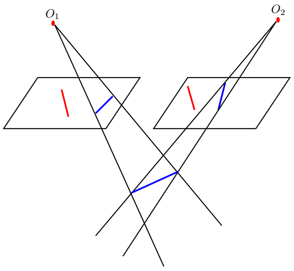


Figure 2: One possible two by two planes intersection

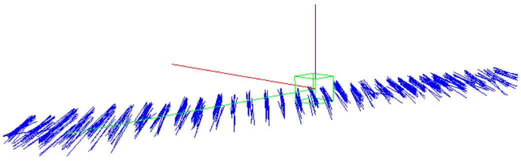


Figure 3: Zebra-crossing of Figure 1 3D segments corresponding to the two by two intersection of pairs of planes

Once we have the direction, we have to find the point which the 3D line goes through. It is defined as the one which minimises the sum of distances to each 3D segment of a given set. Using the same techniques, we have to solve the system $A^tAX = B$ where B is a 3×1 vector. Finally, the end-points of the reconstructed 3D segment are given by projecting orthogonally the extremities of each segment of the considered set and computing the union (see Figure 4 - (Xu and Z.Zhang, 1996)).

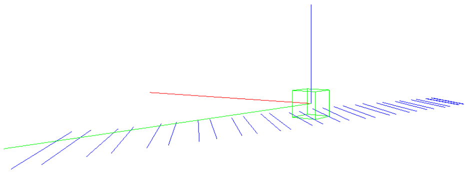


Figure 4: Final 3D segments

This process only reconstructs the long sides of the stripes. We now need to find the transversal axis , i.e the stripes' small side. Thus, we have to find two 3D lines, each of this corresponding to one transversal side of the zebra-crossing. On each side of it, we use a robust least squares minimisation on the long side segment extremities to find those 3D lines. The small sides are then obtained by projecting those lines on the stripes' borders segments. To find a stripe, and thus know which borders we have to link two by two, we use the gradient direction and distance between two consecutive segments (distances constraints from specifications are introduced). Result is shown on Figure 5.

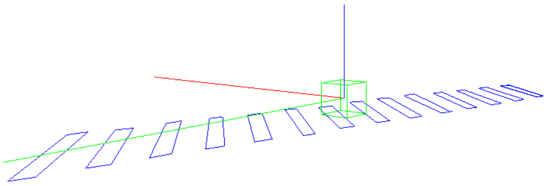


Figure 5: Final zebra-crossing stripes of Figure 1

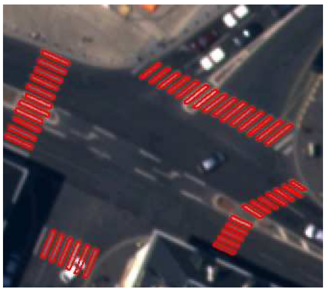


Figure 6: Final zebra-crossings projected in image space

Each stripe of a zebra-crossing is now modeled by a 3D parallelogram and is considered as an initial solution for a fine position refinement described in section 4

3 DISCONTINUOUS ROAD-MARKS RECONSTRUCTION

The other road-mark feature extremely structuring for the road network is the discontinuous line. We now present our strategy for its reconstruction.

3.1 Discontinuous road-marks specifications

Many kind of Discontinuous Road-Marks (DRM) can be found in urban environments. They depend on the road functionality, or on the road type, and the stripes they are composed of are defined by three characteristics: the length, the width and the distance between consecutive stripes. Table 1 and Figure 7 give an overview of the discontinuous road-marks available in the French towns.

Type	Stripes length (m.)	Distance between stripes (m.)
T3	3	1.33
T2	3	3.5
T'2	1.33	5

Table 1: Specifications for discontinuous road-marks

3.2 Monocular extraction

We do not use the protocol presented for zebra-crossing. DRM are objects whose size is under the ground pixel size. Indeed, their stripes are at most 12 cm width. So, working directly with segments in 3D space is not possible because these image features at this resolution are highly miss located: the stripes' borders are stretch toward the exterior, and because of their small length, segments lines have also a very noisy direction. So, the protocol described in 2.3 will be inefficient for reconstructing 3D segments describing stripes' borders. A graph representation - which provides the neighbors of an object - is for this purpose more robust, because predecessor and successor of a stripe will provide a fine orientation needed for the 3D reconstruction of stripes' borders.

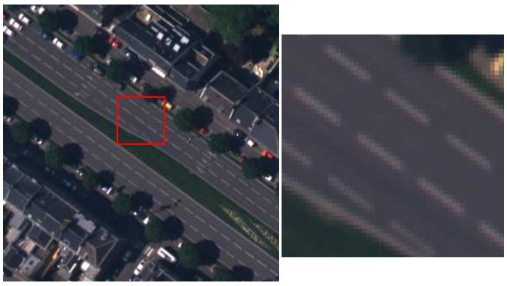


Figure 7: Extracts of a 4000×4000 digital image in Amiens (25 cm ground pixel)

The strategy for DRM detection is based on graph theory. The graph construction of the DRM in an image consists in finding arrangements of segments who best fit the external geometric knowledge from specifications. As for zebra-crossings, segments are extracted and we only keep the ones belonging to a specific length interval defined by the type of DRM we want to extract (see Table 1). We then have segments that potentially belong to DRM. We now have to describe arrangements between those road-marks. So, we build numerical potentials describing the strength of the interactions between pairs of segments. Three potentials described below are used in our application: a connection potential, an alignment potential and a potential for the the length homogeneity. The value for each potential is given by a set of parameters and takes a value thanks to a function.

3.2.1 Potential function

The potential function ζ is generic and has the same general shape for each potential. Two parameters describe it (c and e). However, this function must respect a set of constraints:

- its values must be in $[0;1]$
- it must be symmetric
- it must be increasing on $[-1;0]$
- $\begin{cases} \zeta(c)=\zeta(-c)=0 \\ \zeta(0)=1 \\ \forall x \in [-e;e], \zeta(x)=1 \end{cases}$

The symmetry is important because angles are computed on $[0;2\pi]$. The parameter c allows to choose the extension of the potential function. e is used to have a "plateau" defining a set of values for which the potential function takes its maximum value. Finally, we choose to define the function ζ as:

$$\begin{array}{ccccc} \zeta & : & \mathbb{R}^3 & \rightarrow & [0;1] \\ & & \begin{pmatrix} x \\ c \\ e \end{pmatrix} & \mapsto & \begin{cases} 1 & \text{if} \quad |x| \leq e \\ 0 & \text{if} \quad |x| \geq c \\ \frac{c^2-x^2}{c^2-e^2} & \text{else} \end{cases} \end{array} \tag{3}$$

3.2.2 Potentials definitions

Connection potential

This is the first potential to be computed because if it is null, the others are undefined. Around a given segment s_i, we define a region of interest $ROI_{s_i} = ROI_{s_i}^{(1)} \cup ROI_{s_i}^{(2)}$. Given an angular tolerance θ_c, $ROI_{s_i}^{(j)}$ is an union of discs of radii r_c located at a given distance from $\overline{s_i}$ the middle of s_i in the direction of s_i. This surface is approximated by a trapeze (see Figure 8).

We then look for segments s_j whose middle $\overline{s_j}$ belongs to ROI_{s_i}. If such segments exist, the connection potential is:

$$\mathcal{P}(s_i \underset{c}{\sim} s_j) = \zeta\big(d(\overline{s_i},\overline{s_j}) - d_{th}, c_c, e_c\big) \tag{4}$$

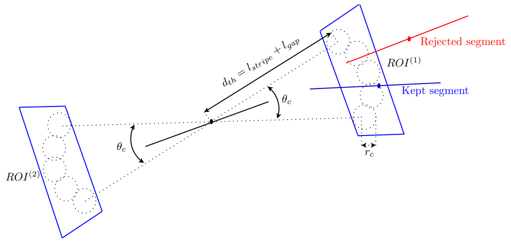


Figure 8: Connection potential description

where d_{th} is the distance between two consecutive stripes. c_c can be defined as a fraction of d_{th} and e_c allows to take into account segment detection accuracy.

Alignment potential

After the connection potential, we compute an alignment potential. It is an angular difference between the two segments s_i and s_j we are studying. The angular difference θ_i^j is then $\theta_i^j = \theta_i - \theta_j$ (see Figure 9). As we want to penalise pairs of segments having a high angular difference, the alignment potential is:

$$\mathcal{P}(s_i \underset{a}{\sim} s_j) = \zeta\big(\theta_i^j, c_e, e_a\big) \tag{5}$$

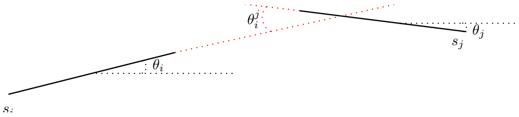


Figure 9: Alignment potential description

In our application, we use $c_e = \pi/2$ because when segments are perpendicular, the potential must be null. e_a is set to avoid penalising curved roads, and can take into account the segment's variance, i.e uncertainties on their angular parameter.

Length potential

This potential is useful to know the length homogeneity of two segments s_i and s_j. We assign a higher potential to pairs of segments of the same length - in a DRM network stripes have the same length (see Figure 10). Thus, we compute the norms' ratio:

$$\mathcal{P}(s_i \underset{l}{\sim} s_j) = \zeta\Big(1 - \min\big(\frac{\|s_i\|}{\|s_j\|}, \frac{\|s_j\|}{\|s_i\|}\big), c_l, e_l\Big) \tag{6}$$

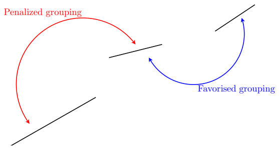


Figure 10: Length homogeneity potential description

e_l allows to have a tolerance on the length. Indeed, the edge detector is very sensitive and often, segments are truncated at their extremities. This parameter is then set to take this observation into account, and so on to avoid penalising grouping of pairs of segments having a small length difference. In addition, we use $c_l = 1$.

Global potential

Finally, once we have computed the three individual potentials, we use a global potential to summarise existing relations between pairs of segments. The global potential is simply a weighted sum:

$$\begin{cases} \mathcal{P}(s_i \underset{G}{\sim} s_j) = \sum\limits_{k=c,a,l} \alpha_k \mathcal{P}(s_i \underset{k}{\sim} s_j) \\ \forall k, \alpha_k \geq 0, \sum\limits_k \alpha_k = 1 \end{cases} \tag{7}$$

As we know, there is a high incertitude on segments norms, so α_l is the smallest coefficient. In addition, α_c and α_a are high, and can be equivalent, but most of the time, we will have $\alpha_c > \alpha_a$. In our application, we often use $\alpha_c = 0.45$, $\alpha_a = 0.35$ and $\alpha_l = 0.2$

To be sure to find the objects relations we are looking for, we use a threshold δ_k on each individual potential and also on the global one. Thus, two segments s_i and s_j are considered to be in interaction, only if the following conditions are respected:

$$\begin{cases} \forall k \in \{c, a, l\}, \mathcal{P}(s_i \underset{k}{\sim} s_j) > \delta_k \\ \mathcal{P}(s_i \underset{G}{\sim} s_j) > \delta_G \end{cases} \tag{8}$$

It is efficient to obtain good results and also in terms of time consuming. Interactions are stored in an $n \times n$ adjacency matrix, where n is the number of selected segments. A segment is selected only if it interacts with another one. The matrix fully describes our DRM network, but we need some simplifications in order to obtain a graph composed of nodes and edges.

Note that some tests show that α_k and δ_k values are not critical.

3.3 Graph creation

As we used a segment detector for our modeling of DRM network, a stripe is most of the time composed of two parallel segments. We want to have a node representing each stripe, and a valued edge (modeling interaction's strength) linking two adjacent stripes.

Thus, a node is created with the following rules:

- if there is only one segment for a stripe, the node is its middle,
- if there are two segments for a stripe, the node is the barycenter of the four extremities (a stripe is composed of two segments if two segments having the same direction and a high recovering are found in a small neighborhood)

The valuations between two edges are computed using the interactions values between pairs of segments composing each stripe. Thus, if we consider two stripes (i.e two nodes), the valuation of the edge linking them is the maximum of the interaction between their segments.

As we use 2D noisy segments lines, the center of a stripe as computed above can only be considered as an estimation of the real position. To obtain a best solution, we build a 2D radiometric template (see section 4) with the known geometry and find the best location of the center by moving the template in the vicinity of the node and optimising a similarity score.

3.4 Chaining road-marks

The graph created in 3.3 is used to extract DRM chains. This is done recursively on its adjacency matrix. We search for long paths and validate them with geometric characteristics. We first look for regularity, i.e a path must not be auto-intersecting and its curvature must vary slowly. In addition, some structures are found on the roofs (false alarms). We filter them using a DTM generated from a DSM. Results are shown on Figure 11.

3.5 3D Reconstruction

A graph of the DRM is created as described in the previous sections for each images. The last step of the reconstruction process consists in matching nodes across the different views. We use here a simple algorithm consisting in making each image being successively the master one. For each stereopairs and epipolar constraints, we search for candidates for matching. The graph structure allows introducing topological, i.e neighborhood constraints. We can thus create a set of possible matches.

From each matching possibility, a 3D point is reconstructed by intersecting the rays (a ray is a 3D line going through the camera's center of projection and the image point). The resulting 3D point

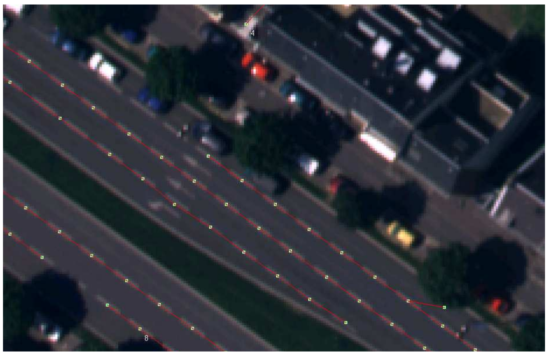


Figure 11: Road marks chaining

is the one which minimises the sum of distances to the rays. To decide between concurrent matches, we use a DSM and check for the Z difference between the reconstructed point and the height given by the DSM. A multi-image similarity score is also used to validate or not the 3D point.

We thus obtain 3D points describing the center of DRM's stripes. Note that if an object (car, tree, ...) hides a DRM element in an image, the multi-image frame allows to obtain with this robust 3D reconstruction the missing element if it is at least not occluded in two images. A 3D reconstruction is given on Figure 12.

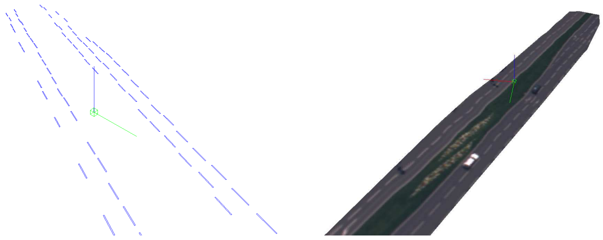


Figure 12: 3D DRM reconstruction and textured triangulation on the 3D stripes' centers

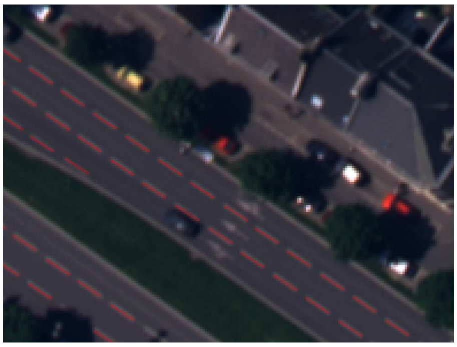


Figure 13: Final DRM of Figure 12 projected on image space

4 3D OBJECTS POSITION REFINEMENT

The strategies presented in 2.2 and 3.2 provides us a robust initial solution that needs to be refined. So, we model a stripe as a parallelogram in 3D space and try to find its optimal position using multiple images (Baltsavias, 1991) in a hierarchical frame. The idea is to distort the base model (the initial solution) in 3D space and to correlate a derivated 2D signal with images data. (Jain et al., 1996) uses this principle in 2D space with a grid transformation. An other modeling of this strategy is proposed in (Chen et al., 2003).

4.1 Notations and definitions

$\mathcal{M}_r^{(n)}$: the model of reference at level n (see 4.3),
$\mathcal{M}_b$: the best model,
$\mathcal{T}$: a class of transformations,
T_i: a transformation ($\mathcal{T} = \bigcup_i T_i$),
$\mathcal{M}_i$: a transformed model ($\mathcal{M}_i = T_i \mathcal{M}_r^{(n)}$).

An object's model is represented with a set of four points. So, a model is defined by the central point $p_i = (x_i, y_i)^t$ of the stripe, its length L_i and direction d_i^1, its width l_i and direction d_i^2.

A transformation is the set of operations used for the generation of the model hypothesis. Both for zebra-crossing and DRM stripe, it is composed of two rotations $\widetilde{r_1}$ and $\widetilde{r_2}$ along the directions vectors d_i^1 and d_i^2, and of translations t_1, t_2 and t_3 along each 3D axis. Specially for zebra-crossings, the transformations also have to take into account the length and width variations of the object. Finally, a model is composed of five parameters for a DRM stripe and of six parameters for a zebra-crossing stripe (see Equation 9).

$$\begin{cases} T_i^{Zebra}(\widetilde{r_1}, \widetilde{r_2}, t_1\,\vec{X}, t_2\,\vec{Y}, t_3\,\vec{Z}, \alpha L_i) = T_i^{Zebra}. \\ T_i^{DRM}(\widetilde{r_1}, \widetilde{r_2}, t_1\,\vec{X}, t_2\,\vec{Y}, t_3\,\vec{Z}) = T_i^{DRM}. \end{cases} \tag{9}$$

The vectors of parameters to be estimated are then defined by:

$$\begin{cases} \Theta^{Zebra} = (\widetilde{r_1}, \widetilde{r_2}, t_1, t_2, t_3, \alpha) \\ \Theta^{DRM} = (\widetilde{r_1}, \widetilde{r_2}, t_1, t_2, t_3) \end{cases} \tag{10}$$

4.2 Model choice

To choose the best 3D position for a stripe, our strategy is to compare the image signal with a perfect simulated signal. For each model in 3D space $\mathcal{M}_i$ we have four points making a parallelogram. The knowledge of the projection geometry allows to project this shape in all the images I_j. We thus obtain for each vertex of $\mathcal{M}_i$ its subpixellar position in 2D images spaces. We then simulate a signal SS_{ij} (j stands for the image number) with this positions for each images, i.e a white anti-aliased 2D shape on black background. Finally, the best 3D model $\mathcal{M}_b$ is chosen by maximising the following energy:

$$\mathcal{M}_b = \underset{i}{max} \sum_j Corr_{\mathcal{M}_i}(SS_{ij}, I_j) \tag{11}$$

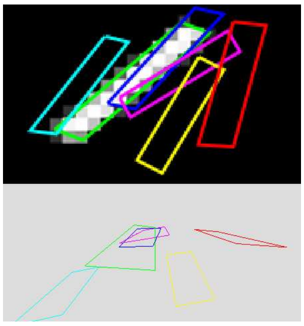


Figure 14: Projections in image space of 3D models (lower image) on a simulated signal (upper image). Each color corresponds to a different model $\mathcal{M}_i$. The found solution is in green.

4.3 Hierarchical models generation

As we have six parameters to estimate a zebra stripe and five for a DRM stripe, the computational search space is huge (because all parameters are estimated simultaneously) and need to be reduced. That is why we adopt an iterative multi-scale frame (Kropatsch, 1991; Hummel, 1988). For each level of the hierarchy, we set search spaces and steps. This idea has already been used in different context (Gharavi-Alkhansari, 2001; Stefano et al., 2005).
The system is initialised with $\mathcal{M}_r^{(n)}$ (the initial solution). At this level n of the hierarchy, the search spaces and sampling distances on the parameters are the biggest. From this reference model and with a class of transformations $\mathcal{T}$, we build several models $\mathcal{M}_i$ and the simulated SS_{ij} signals in the images. Then, the best model $\mathcal{M}_b^{(n)}$ at this level is given by Equation 11. We go down a level of the hierarchy and repeat this process with initialising $\mathcal{M}_r^{(n-1)}$ with $\mathcal{M}_b^{(n)}$.
Each time we go down a level, the search spaces and sampling distances are reduced. Here, for both we use a dyadic factor. This protocol is iterated while $n > 0$ or convergence is reached.
The number n of levels of the hierarchy, the search spaces and sampling distances are chose to be in adequation with the wished accuracy for the final stripe position.

5 RESULTS

To test the robustness of our algorithms and their ability to detect and reconstruct road marks, we have a reference database of points surveyed with a millimetric accuracy on the town of Amiens. It is composed of both zebra-crossings and DRM stripes' corners, and were acquired with classical topometric techniques. The evaluations were done only for the zebra-crossings, but give clear information about the algorithm's accuracy. $\frac{B}{H}$ ratio is in the range $[0.2; 0.6]$ and reconstructions were performed using from 3 to 9 images. We first measure absolute planimetric and altimetric accuracies on a set of 112 stripes. The RMS is about 15 cm for the first one, and less than 20 cm for the second one mainly to the quality of the aerial triangulation. In terms of relative accuracy, the algorithm shows its ability to be very fine. Indeed, it's about only a few cm, meaning that the global structure of a zebra-crossing is preserved by our algorithm. We can also note that the geometric refining presented in section 4 gives good results. The accuracy gain is about 5 cm. For both zebra-crossing and DRM, there are only a few false positives alarms because there are no ground structures having the same radiometric and geometric properties as the objects we want to reconstruct. In addition, the false positives detected for DRM are located on the buildings' roof and can easily be filtered with a focalisation mask. However, the detection rate is higher than 90% for zebra-crossing stripes. The missing stripes are the small ones located near the pavement, the ones hidden by a car or the old ones degraded (thus loosing their geometric and radiometric properties).

6 CONCLUSION AND FUTURE WORKS

As we have shown on examples, our modeling and detection of the road-marks is very efficient for road detection and characterisation in an urban environment. In can also be extended to suburban areas or motorways.
To obtain a tool able to give more complete informations on the road network, we now have to detect other road-marks (specialised lanes, bus stops, traffic informations, ...).
An other key point to take advantage of our systems (aerial and terrestrial) is to have a full collaboration between them, e.g to search for missing objects in the images from the other viewpoint. We have presented 3D reconstructing experts for road marks which are a structuring features of the road network e.g to separate lanes and estimate their width.

REFERENCES

Baltsavias, E., 1991. Multiphoto geometrically constrained matching. PhD thesis, Institute for Geodesy and Photogrammetry, ETH Zurich.

Charbonnier, P., Diebolt, F., Guillard, Y. and Peyret, F., 1997. Road markings recognition using image processing. In: IEEE Conference on Intelligent Transportation System, Vol. 1, pp. 912–917.

Chen, J. H., Chen, C. S. and Chen, Y. S., 2003. Fast algorithm for robust template matching with M-estimators. In: IEEE Transactions on Signal Processing, Vol. 51-1, pp. 230–243.

Collins, R., 1996. A sweep-space approach to true multi-image matching. In: Proceedings of the 15^{th} Conference on Computer Vision and Pattern Recognition, San Francisco - USA.

Deriche, R., 1987. Using Canny's criteria to derive a recursively implemented optimal edge detector. International Journal of Computer Vision 1(2), pp. 167–187.

Deriche, R., Vaillant, O. and Faugeras, O., 1991. From noisy edge points to 3D reconstruction of a scene: a robust approach and its uncertainty analysis. In: Proceedings of the 7^{th} Scandinavian Conference on Image Analysis, Alborg - Danmark, pp. 225–232.

Douglas, D. H. and Peucker, T. K., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer 10(2), pp. 112–122.

Enkelmann, W., Struck, G. and Geisler, J., 1995. ROMA - a system for model-based analysis of road markings. In: IEEE Proceedings of the Intelligent Vehicle '95 Symposium, Vol. 1, Graz - Austria, pp. 356–360.

Gharavi-Alkhansari, M., 2001. A fast globally optimal algorithm for template matching using low-resolution pruning. IEEE Transactions on Image Processing 10(4), pp. 526–533.

Hinz, S. and Baumgartner, A., 2002. Urban road net extraction integrating internal evaluation model. In: ISPRS Commission III Symposium on Photogrammetric Computer Vision, Vol. XXXIV - Part 3A, Graz, Austria, pp. 163–168.

Hummel, R., 1988. The scale-space formulation of pyramid data structures. Parallel Computer Vision pp. 107–123.

Jain, A. K., Zhong, Y. and Lakshmanan, S., 1996. Object matching using deformable templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(3), pp. 267–278.

Kropatsch, W., 1991. Image pyramids and curves, an overview.

Pénard, L., Paparoditis, N. and Pierrot-Deseilligny, M., 2006. Reconstruction 3D automatique de façades de bâtiments en multi-vues. In: RFIA, Tours - France.

Pierrot-Deseilligny, M. and Paparoditis, N., 1996. A multiresolution and optimization-based image matching approach: an application to surface reconstruction from SPOT5-HRS stereo imagery. In: WG I/5-6 Workshop on Topographic Mapping from Space, Vol. XXXVI, Ankara - Turkey.

Rebut, J., Bensrhair, A. and Toulminet, G., 2004. Image segmentation and pattern recognition for road marking analysis. In: IEEE International Symposium on Industrial Electronics, Vol. 1, Graz - Austria, pp. 727–732.

Royer, E., Lhuillier, M., Dhome, M. and Lavest, J.-M., 2006. Localisation par vision monoculaire pour la navigation autonome: précision and stabilité de la méthode. In: RFIA, Tours - France.

Se, S. and Brady, M., 2003. Road feature detection and estimation. Machine Vision and Applications 14(3), pp. 157–165.

Stefano, L. D., Mattoccia, S. and Mola, M., 2005. An efficient algorithm for exhaustive template matching based on normalized cross correlation. In: Proceedings of the 12^{th} International Conference on Image Analysis and Processing.

Steger, C., Mayer, H. and Radig, B., 1997. The role of grouping for road extraction. In: A. Gruen, E. Baltsavias and O. Henricsson (eds), Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhäuser Verlag, Basel, Switzerland, pp. 245–256.

Taillandier, F., 2004. Reconstruction du bâti en milieu urbain: une approche multi-vues. PhD thesis, Ecole Polytechnique - Paris.

Vosselman, G., 2003. 3D reconstruction of roads and trees for city modeling. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences XXXIV(part 3/W13), pp. 231–236.

Xu, G. and Z.Zhang, 1996. Epipolar geometry in stereo, motion and object recognition. Kluwer Academic Publishers.

Zhang, C., 2003. Updating of cartographic road databases by image analysis. PhD thesis, Institute of Geodesy and Photogrammetry, Zurich.

ASSESSMENT OF LIDAR DTM ACCURACY IN COASTAL VEGETATED AREAS

J. Göpfert, C. Heipke

Institute of Photogrammetry and GeoInformation
University of Hannover
(goepfert, heipke)@ipi.uni-hannover.de

Commission III, WG III/3

KEY WORDS: lidar, vegetation, accuracy, feature extraction

ABSTRACT:

Digital terrain models (DTM's) are widely used in coastal engineering. Reliable height information is necessary for different purposes such as calculating flood risk scenarios, change detection of morphological objects and hydrographic numeric modelling. In this specific field light detection and ranging (lidar) replaces step by step other methods such as terrestrial surveying. However, some new problems are associated with lidar technology. For instance, in vegetated areas the accuracy of the lidar DTM decreases.
In this paper the influence of different types of coastal vegetation on the accuracy of the lidar height information is investigated. For that purpose this research starts with a comparison of terrestrial control measurements and the lidar data in order to detect problematic areas with respect to the accuracy of the DTM. Based on the resulting height differences the influence of different attributes of the vegetation, i.e. type, height, density, is analysed. In the next step typical features, which are able to describe the attributes, are extracted from the available remote sensing and GIS data (ranging from laser heights and intensity information to multispectral images and biotope mapping). These features were used to perform a classification of the lidar data in different categories of accuracy. Finally, first results for two test areas are presented.

1. INTRODUCTION

Various agencies operating in the field of coastal management require reliable area-wide height information for the transition zone between land and water, in dunes and for their protection facilities, in order to detect important changes with regard to the safety of the coastal area. In former times terrestrial surveying was used to collect this information. However, these methods are very time and cost consuming as well as difficult to perform in coastal areas with dense vegetation and frequently flooded terrain. Therefore, the lidar technique replaces more and more the traditional methods. However, new problems related to the application of the lidar data have to be solved. The influence of vegetation on the quality of the lidar DTM is one of these problems. The laser beam is not able to fully penetrate dense vegetation surfaces such as shrubberies in dune valleys. Thus, the laser pulse is often reflected before hitting the bare ground or a mixed signal (surface as volume scatterer) generates a certain height off-set. Common filter algorithms are able to remove points reflected from higher vegetation. If there are only a few or no ground points in the analysed area caused by the dense vegetation the filter methods fail. Additionally, low vegetation which is not significantly higher than the surrounding bare ground is difficult to detect. Figure 1 demonstrates the circumstances on the basis of a valley in the dunes (East Frisian Island Juist) with dense standings of Japanese rose and creeping willow. The digital surface model (DSM) from the unfiltered lidar data is illustrated on the left side (a), whereas the figure in the middle depicts the lidar DTM (b) and the DTM of the control points is displayed on the right (c). Obviously, some vegetation points are still present in the dataset after the filtering process. This paper investigates the described influence of the vegetation on the accuracy of the lidar DTM for typical plant population in the coastal area of Northern Germany.

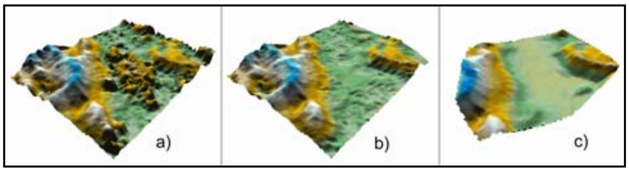


Figure 1: a) lidar DSM, b) lidar DTM, c) DTM of control points

Additionally the potential of different attributes (vegetation height and density) for the description of the height discrepancies caused by the vegetation are analysed. These attributes have to be connected to features extracted from the available remote sensing data in order to perform a classification of the lidar data in different accuracy levels. Finally, the presented approach generates a lidar quality map depending on the vegetation.

2. STATUS OF RESEARCH

Before choosing attributes which influence the lidar quality in vegetated areas, it is necessary to understand the basic principles of the interaction between the laser beam and the reflecting surface. Wagner et al. (2004) discussed physical concepts for understanding how distributed targets such as trees or inclined surfaces transform the emitted laser pulse by using the radar equation. Additionally, they pointed out the advantages of full-waveform scanners in the analysis of the backscattered laser pulse. Pfeifer et al. (2004) considered the influence of different parameters such as flying height, footprint size, echo detection and selection as well as pulse width on the laser measurement over vegetation.

After understanding the basic principle an analysis can be performed by using ground truth measurements in comparison to the lidar height. In this manner several studies investigated the influence of different vegetation types on the quality of the lidar DTM. Elberink and Crombaghs (2004) found a systematic upwards shift of up to 15cm for low vegetated areas (creeping red fescue). Ahokas et al. (2003) evaluated the lidar accuracy for asphalt (standard deviation 10 cm), gravel (4cm), grass (11cm) and forest ground (17cm). Pfeifer et al. (2004) investigated the influence of long dense grass (+ 7.3cm), young forest (+ 9.4cm) and old willow forest (+ 11.6cm) on the accuracy of lidar data. Hodgson and Bresnahan (2004) used the horizontal coordinates of the irregularly distributed lidar points for the measurement of the ground truth in order to avoid an interpolation influence during the calculation of an error budget for a lidar data set. They found a standard deviation of 17cm for evergreen and 26cm for deciduous forest, however in contrast to other studies only low shifts (-4,6 cm for evergreen, + 1,0cm for deciduous) occurred.

Only a few researchers investigated object or data driven parameters with an influence of the laser measurement except the vegetation type. Hopkinson et al. (2004) presented a method to identify the relationship between the standard deviation of pre-processed laser heights (the ground elevation was subtracted from the first and last pulse measurement) and vegetation height itself for low vegetation (<1,3m). They found the following expression,

vegetation height = 2.7 * standard deviation,

and determined the r.m.s.e. of the predicted vegetation heights with 15cm. Pfeifer et al. (2004) and Gorte et al. (2005) used also the variation of the laser heights in order to detect relations to the height shift in low vegetated areas. Instead of the standard deviation they defined texture parameters and showed their potential for correction.

In (Moffiet et al., 2005) the capabilities of classified returns (ground and vegetation, first, last and single pulse) as well as the returned intensity were investigated to distinguish different tree types. The authors pointed out that the average and the standard deviation of the intensity values are affected by the forest structure as well as the reflective properties of the vegetation, whereas the information content of a single intensity value is difficult to interpret.

In different studies a combination of height and multispectral data is used in order to detect and classify vegetation types. For example, Mundt et al. (2006) explored the potential of this combination for mapping sagebrush distribution; and Straub and Heipke (2001) determine tree hypothesis using geometric and radiometric features from height and image data.

3. DATA

This research is mainly based on two test flights. Most of the investigations were carried out using data collected by the company Milan-Flug GmbH covering the region of the East Frisian Island "Langeoog" in the leaf-off period (April 2005). During the campaign a LMS Q560 system of the company Riegl was used. Flying at a height of 600m the system provided an average point density of 2.9 points/m^2. The following data were collected:

- RGB – Orthophotos (resolution: 0,2 m)
- maximum of three pulses per laser beam
- unfiltered raw data (x, y, z, intensity)
- points (x, y, z), separated into ground and vegetation

Supported by biologists various control areas for typical vegetation types were defined. Within a few days of the flight campaign ground truth data for these regions were collected including the height of the ground and the vegetation as well as a verbal description of the vegetation. For each of the following vegetation types two test fields were chosen:

- Japanese rose (Rosa rugosa) (vegetation heights up to 1,3 m)
- Beach grass (Ammophila arenaria) (<1,0m)
- Crowberry (Empetrum nigrum) (<0,4m)
- Creeping willow (Salix repens) (<1,6m)
- Common seabuckthorn (Hippophaë rhamnoides) (<1,4m)
- Common reed (Phragmites australis) (<2,2m)
- Sand couch grass (Agropyron pungens) (<0,5m)

Furthermore, two mixed habitats (rose/seabuckthorn (<2,6m) and seabuckthorn/willow (<1,6m) were investigated. Additionally, some fisheye photos taken from the ground to the zenith were acquired in order to quantify the vegetation density (Figure 3). Four bare ground areas in the immediate vicinity of the vegetated test region were surveyed to check the general quality of the data.

The data for the second test flight were collected during a measurement campaign of the company TopScan with an ALTM 2050 scanner from Optech covering the East Frisian island Juist (March 2004). The flying altitude was 1000m and the system provided an average point density of 2 points/m^2. The following data were used for the analysis:

- CIR – Orthophotos (resolution: 0,2 m)
- last pulse data
- unfiltered raw data (x, y, z, intensity)
- points (x, y, z), separated into ground and vegetation

A test area called "Dunes" consisting of 696 control points within a mixed population of Japanese rose and willow was surveyed in the same way as described above. Vegetation heights of up to 2,8m occurred.

Finally, a biotope mapping performed on aerial photos taken in 2002 and 2003 with a HRSC-AX and a DMC camera was used as input for the distinction of different predominant vegetation types.

4. METHODS

In this paper we analyse the relationship between different object as well as data driven features and the accuracy of the lidar DTM in vegetated areas. We combine image, lidar and GIS data to accomplish this task. In contrast to the above mentioned work our aim is not to do vegetation classification

Initially, section 4.1 investigates the characteristics of vegetation with respect to the lidar measurement. The next section connects the analysed vegetation attributes to features generated from remote sensing data. Finally, in 4.3 the workflow to classify the lidar data into different accuracy levels using the extracted features is discussed.

4.1 Characteristics of vegetation in lidar data

For an assessment of the influence of vegetation attributes on the quality of the lidar height information we use ground truth measurements. In order to compare directly terrestrial and lidar data it is necessary to interpolate the heights obtained by one of the methods from the surrounding measurements. One argument for the interpolation of the lidar data is the higher point density (~3 points/m^2 in comparison to ~0.5 points/m^2 for the ground truth measurements). Additionally, the topographic features of the surface are not reflected in the point distribution of the terrestrial data due to the difficult measurement conditions, i.e. dense vegetation and rough terrain. For these reasons we compute a DTM from the filtered lidar data and interpolate the lidar height information for the control points using their x- and y-coordinates. Then the mean height discrepancies (lidar-DTM minus reference height) and their standard deviation were determined.

Initially the height discrepancies (lidar DTM minus reference height) depending on one parameter are analysed. The influence of the following parameters is investigated:

- vegetation type
- vegetation height
- vegetation density

Many related studies found that the surface type is one of the crucial factors for the accuracy of the DTM derived from lidar data (see also status of research). For that reason the first parameter which has to be analysed is the vegetation type.
Laser pulses do not penetrate every layer of the vegetation in a similar way. Therefore, the laser beam is very often reflected above the bare ground or a mixed signal from ground and vegetation returns to the scanner. Thus, an upwards shift for the lidar heights is expected in vegetated areas. Caused by the different structure the influence of several vegetation types on the lidar accuracy should vary.
The research focuses on typical vegetation for coastal areas, beginning with layers of biomass covering the ground in spring, produced by felted mulch or bear leaves during winter times. For example dense standings of beach grass and shrubberies in dunes as well as common reed and sand couch grass in the transition zone between land and water belong to the monitored vegetation types.

The vegetation height is the next analysed parameter. With higher vegetation the distance of the laser beam through the different layers of organic material becomes longer. Therefore, assuming a uniform vegetation structure in every layer the probability that a part of the laser energy is absorbed or reflected before reaching the ground is higher.
In order to investigate the influence of the vegetation height the parameter is divided into regular intervals. The height discrepancies at the control points are assigned to the related interval. For every related interval the mean and the standard deviation of the height shift are determined and plotted over the vegetation height.

Subsequent, the influence of the vegetation density is studied. The vegetation density can not be measured directly. Therefore, suitable values which are able to describe the characteristics of the vegetation density must be defined. One method to quantify this parameter determines the ratio of the classified ground points to all lidar points in the analysed test field. This idea assumes that in dense vegetation less laser pulses penetrate the canopy and more vegetation points are filtered. For that reason a larger ratio implies a lower vegetation density and can potentially act as an indicator for higher accuracies of the lidar DTM. However, the filter result may be wrong, and thus the definition of vegetation density breaks down in very dense vegetation surfaces which are hardly penetrated by the Laser beam.
The analysis of the fish eye photos offers another method to define the vegetation density (figure 2). An algorithm to calculate the coverage with organic material depending on the zenith angle was developed based on the rectified image (for the rectification process see r. g. Schwalbe, 2005). The images are segmented into vegetation and background using simple thresholds which are calculated from the minima of the grey value histogram. Finally, the correlation between the degree of coverage and the height discrepancies is investigated.

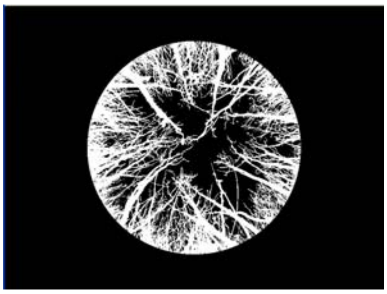


Figure 2: Segmented fish eye photo (rosa rugosa)

Note that the term “dense vegetation” is also related to the size of the lidar footprint: lasers with smaller footprints are able to penetrate vegetation with higher density in a more undisturbed way. The impact of the size of the lidar footprint on the penetration rate, however, is not subject of this research.

Finally, the parameters vegetation height and density are also analysed with respect to only one vegetation type, in order to describe a more complex model of the influence of the vegetation on laser heights.

4.2 Features for classification

In the next step features have to be determined which are able to represent the vegetation attributes in the remote sensing data. The mean value and the standard deviation of the multispectral, lidar height and intensity channels and some texture parameters derived from the co-occurrence matrix (i.e. contrast and homogeneity) are investigated. The co-occurrence matrix contains the spatial dependencies of the grey values for certain directions and distances (see e. g. Haralick, 1979).
Considering one vegetation type we assume that the mean values of the Normalized Difference Vegetation Index (NDVI) correspond to the vegetation density: The leaf area index (LAI) is one of the most important parameters for characterising the structure of canopy. Many studies such as (Pandya, 2004) found a strong positive correlation between the LAI and the NDVI calculated from remote sensing data. Therefore, if the NDVI increases, the amount of active organic material and the vegetation density in the pixel should be higher. For the test flight in the area of the island Langeoog the near infrared channel is not available. Thus, the Degree of Artificiality (DoA) as defined in (Niederöst, 2000) is used instead of the NDVI.
Next, we relate lidar intensity to vegetation density: Every layer of the vegetation where the laser pulse is reflected decrease the intensity value for the following echoes. For that reason a lower intensity indicates a higher vegetation density under the assumption of a similar reflectivity observing only one kind of vegetation and the same beam direction. However, very dense vegetation surfaces which can not be penetrated by the laser pulse yield higher intensity values. But the cross section of the illuminated area is not as homogeneous as the footprint hitting bare ground. Therefore, the average returned intensity in vegetated areas should be lower. Thus, for pre-defined neighbourhoods we compute the mean and the average of the lidar intensity (also motivated by Moffiet et al. 2005).
Furthermore, features have to be defined for the vegetation height. On one side we can use the contrast of the height image

derived from the co-occurrence matrix to describe height differences in the local neighbourhood of a pixel. A higher contrast is equivalent to larger differences of grey values, and we assume a correlation with higher vegetation. On the other side for vegetation heights larger than 0.5m different pulses can be detected by the Riegl scanner. In this case we use the differences between the first and the last pulse of lidar raw data to define the vegetation height.

Finally, instead of extracting features for the distinction of different vegetation types we use the biotope mapping in order to limit the research area to one predominant plant population.

The capabilities of extracted features to describe the characteristics of the vegetation and their influence on the accuracy of the lidar DTM are tested using the height discrepancies at the control points.

4.3 Classification

Based on the different data sources (multispectral image, lidar data, biotope mapping) a supervised classification is performed in order to divide the lidar data into different levels of accuracy depending on the predominant vegetation. Figure 3 depicts the workflow of the classification.

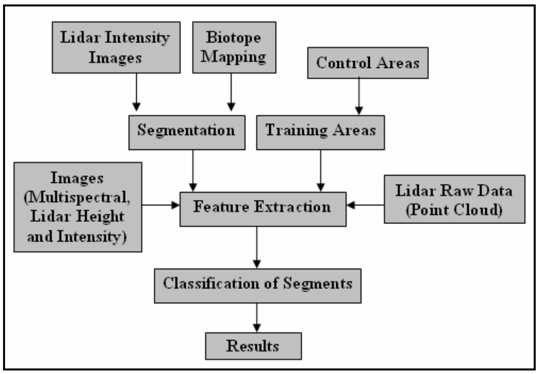

Lidar Intensity Images
Biotope Mapping
Control Areas
Segmentation
Training Areas
Images (Multispectral, Lidar Height and Intensity)
Feature Extraction
Lidar Raw Data (Point Cloud)
Classification of Segments
Results

Figure 3: Classification workflow

A segment based approach for the classification was chosen in order to consider the local neighbourhood of the laser pulse and to define mean values and standard deviation as well as other texture parameters.

Initially, a DSM is calculated using the unfiltered lidar data in order to preserve texture information stemming from the vegetation. Subsequently, this DSM is transformed to a greyscale image in order to use the data in combination with the multispectral images for an image based classification. The same procedure is accomplished for the intensity values of the returned laser pulses.

The segmentation is performed using a region growing method applied to the low pass filtered lidar intensity image. Starting with the local grey value minima as seed regions (corresponds to areas with low lidar accuracy), the analysed pixel is assigned to the current segment if the difference of the average grey value of the segment and the grey value of the pixel is smaller than a certain threshold.

Previous work indicates that the vegetation type is an important factor for the accuracy of the lidar DTM and for the applicability of the discussed features. Thus, the extension of the segments and, consequently, the area of the following classification are limited to one predominant vegetation type using the borderlines of the biotope mapping.

Training areas are generated by using the height discrepancies from the control points. For that purpose a difference model is calculated and transformed into an image, so that the grey values correspond to the height discrepancies. This image is segmented into different accuracy levels. These segments are used as training areas for the classification.

In the last step the feature vectors derived for the training areas and the segmentation are used to classify the lidar height data into different levels of accuracy. In this paper the Euclidian distance between the feature vectors is used to classify the current segment. For this method the features are normalised to the same overall value in order to weight the features equally.

5. RESULTS

5.1 Characteristics of vegetation in lidar data

The influence of the vegetation type on the lidar accuracy is illustrated in figure 4. Obviously, the lidar DTM is higher than the related control points for each vegetation type (8 – 24 cm). This finding corresponds to the theoretical consideration that laser pulses do not penetrate all vegetation. For each type the standard deviation of a single measurement is only in the range from 5 up to 15cm. The highest height shift was detected for beach grass (+19.3cm), seabuckthorn (+18.4cm), sand couch grass (+20.1cm) and the mixed area seabuckthorn/willow (+23.2cm).

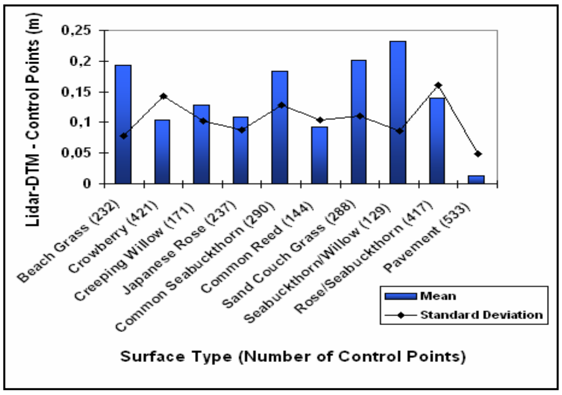


Figure 4: Height shift for different surface types
(Langeoog, Riegl scanner)

Figure 5 illustrates the accuracy of the lidar DTM depending on the vegetation height. The relatively large discrepancies for the vegetation heights of 0.5 – 1.0m are caused by the beach grass belonging to this range. Various standings of beach grass produce a height shift up to 0.38m. In contrast many values obtained by the control area in the reed lead to lower discrepancies in the diagram for vegetation heights between 1.7 and 2.0m. Due to the vertical plant structure without ramifications in the leaf-off period the influence of the reed on the quality of the lidar DTM is low. In summary, the diagram demonstrates that the vegetation height without considering other parameters does not suffice in order to describe the height discrepancies in the vegetated areas.

However, considering only one vegetation type some plant heights show a strong correlation with the height discrepancies (see figure 6). Obviously, the filtering process influences these dependencies. If some points of the higher vegetation are filtered, the accuracy for the related interval increases.

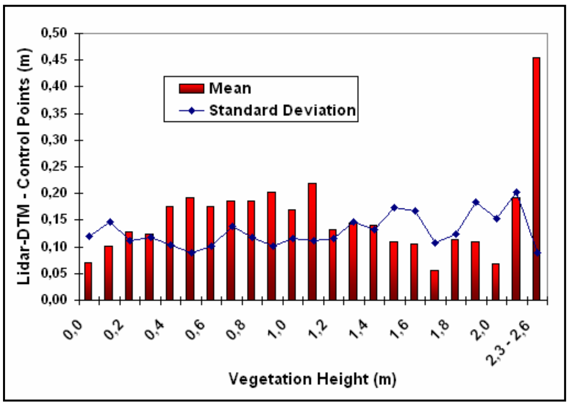


Figure 5: Height shift plotted over vegetation heights for all vegetation types (Langeoog, Riegl scanner)

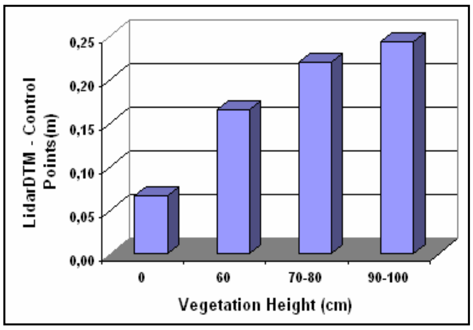


Figure 6: Height shift plotted over vegetation heights for beach grass (Langeoog, Riegl scanner)

In figure 7 the correlation between the vegetation density calculated from the fisheye images and the height discrepancies is visualised. Only a low correlation (0.19) was found considering all vegetation types. However, for several lower vegetation types a high correlation was detected, e.g. for beach grass and Japanese rose. Therefore, for certain vegetation types the defined vegetation density seems to correspond to the height shift. The two negative values are caused by some outliers which occurred due to the filtering process, as was determined by a detailed analysis of the data.

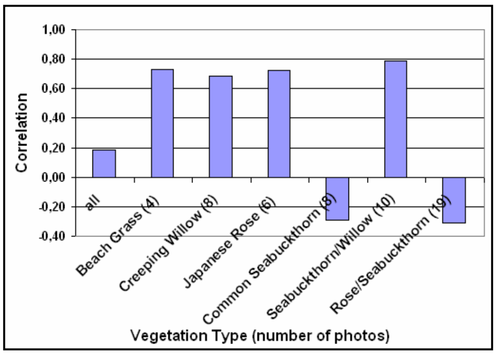


Figure 7: Correlation between the degree of coverage of the fisheye photos and the height shift (zenith angle up to 40°)

5.2 Features for classification

Figures 8 and 9 show the height discrepancies depending on the lidar intensity for some vegetation types. A high negative correlation can be detected for the intensity values (-0.51 for beach grass, see figure 8), and especially for the lidar DSM (-0.92 instead of -0.6 for the DTM, see figure 9). Thus, whereas the filtering process eliminates higher vegetation and therefore increases the accuracy of the lidar result, it unfortunately also renders lidar intensity less useful as an indicator for the lidar DTM accuracy, because the darker points potentially belonging to the upper vegetation are filtered out.

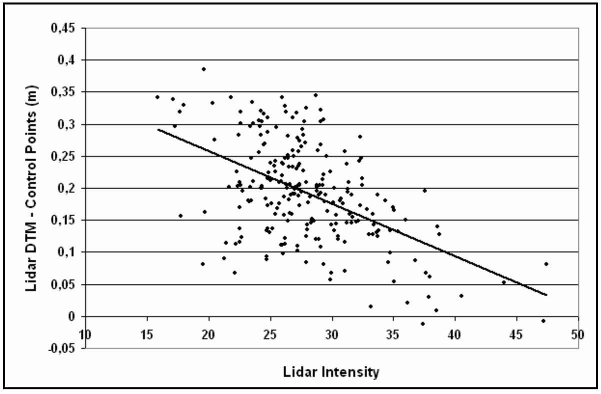


Figure 8: Height shift plotted over lidar intensity for beach grass (Langeoog, Riegl scanner)

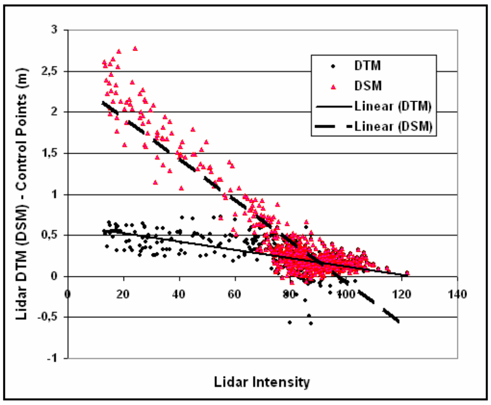


Figure 9: Height shift plotted over lidar intensity for Japanese rose/willow (Juist, ALTM scanner)

Figures 10 and 11 illustrate the height discrepancies depending on the DoA and NDVI. Only a low correlation (0.39 for beach grass (DoA), 0.23 for Dunes (NDVI)) between the height shift and the indices could be identified. Due to the fact that the measurement campaign was conducted in spring, the plants in the test area Dunes (rose, willow) had just started their activity. Therefore, in general only low NDVI values occurred. Obviously, the correlation for active vegetation such as beach grass is higher.

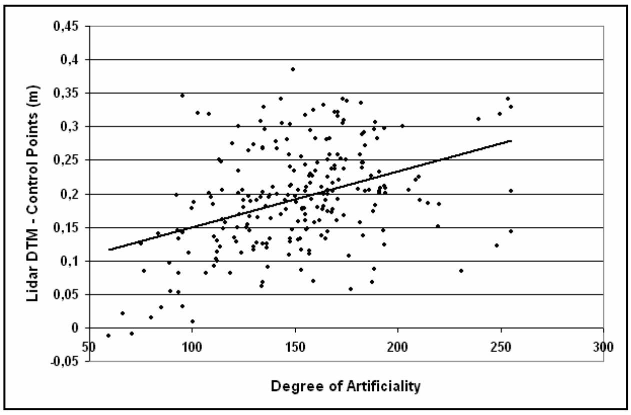


Figure 10: Height shift plotted over DoA for beach grass (island Langeoog, Riegl scanner)

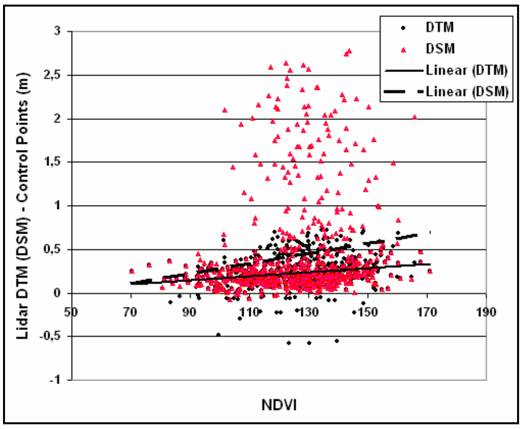


Figure 11: Height shift plotted over NDVI for the test area dunes (Japanese rose/willow, island Juist, ALTM scanner)

In summary, lidar intensity as well as vegetation indices show potential for the distinction of different accuracy levels in the lidar DTM. Additionally, a negative correlation between the height shift and the values of the channels in the visible spectrum was also detected. However, all values vary strongly in a single accuracy interval. Therefore, mean values for segments are more suitable for classification purposes.

5.3 Classification

Finally, first classification results for beach grass and the test area Dunes are presented. Tables 1 and 2 summarise the extracted features for different accuracy levels, while figures 12 and 13 illustrate the results graphically. In addition to the features discussed in 5.2 the contrast of the height image was also used (see also chapter 4.2).

In the first example two different training areas of beach grass are used to classify the same region. If the algorithm works correctly, the extracted features for the accuracy intervals in table 1 and the classifications on the left and the right side of figure 12 have to be the same. However, comparing these features and classifications some differences can be identified. The problems are associated with the different range of the height shift (table 1 second row) and the varying size of the control areas related to the accuracy levels for the two test fields. The latter one has a strong influence on the extracted height contrasts.

Table 1: Extracted features for the accuracy levels of beach grass (not normalised)

Parameter	Beach Grass 1			Beach Grass 2		
Area Height Shift (cm)	(+5,7) – (+34,5)			(-1,2) – (+38,5)		
Class (Height Shift)	<+13	<+26	>+26	<+13	<+26	>+26
Mean Blue	88,5	83,6	82,5	86,4	83,0	80,6
Mean DoA	79,5	89,0	92,7	76,1	89,2	94,5
Mean Intensity	70,0	69,6	65,1	83,6	67,7	60,8
Height Contrast	0,20	0,21	0,24	0,13	0,30	0,33

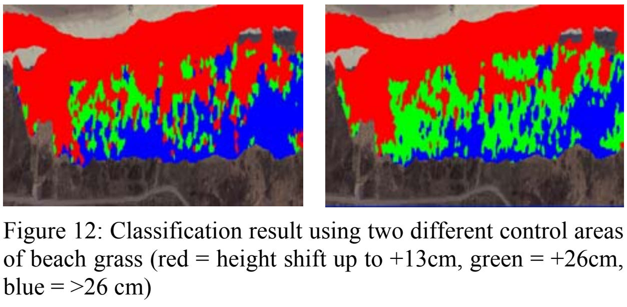


Figure 12: Classification result using two different control areas of beach grass (red = height shift up to +13cm, green = +26cm, blue = >26 cm)

Due to the completely different size and form of the training areas and the segments it is also possible to compare the classification result within the training areas in order to assess the applicability of the segmentation process and the practicability of the classification using the extracted features. A good match can be detected in the second example for the test area Dunes (figure 13).

Table 2: Extracted features for the accuracy levels of the test area Dunes (not normalised)

Parameter	Test Area Dunes				
Area Height Shift (cm)	(-7,7) – (+72,0)				
Class (Height Shift)	<+15	<+30	<+45	<+60	>+60
Mean Green	84,7	81,1	77,3	76,7	76,5
Mean NDVI	129,0	136,9	146,0	145,6	149,2
Mean Intensity	88,5	81,6	62,0	40,5	38,3
Height Contrast	0,28	0,34	0,68	0,62	0,50

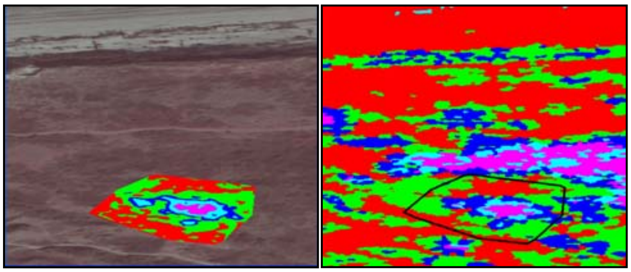


Figure 13: Left: training area from control points, Right: Classification result for a part of the island Juist (red = height shift up to +15cm, green = +30cm, blue = +45 cm, cyan = +60cm, pink = >+60cm)

6. CONCLUSION AND OUTLOOK

This paper discusses an approach for mapping the quality of lidar heights in vegetated areas using a combination of various data sources. Some features (lidar intensity, height contrast, vegetation indices) show capabilities in order to classify lidar data in vegetated areas into different accuracy levels. However, attributes and features are strongly correlated to the vegetation type. Therefore, a biotope mapping or a multispectral classification of the vegetation has to be used in conjunction with the lidar data.

The vegetation attributes, such as the height and density, as well as some extracted features, such as lidar intensity, show a better correlation to the height discrepancies using unfiltered lidar data. Only the height contrast is related to the filtering process. Therefore, the presented method should be applied to unfiltered lidar data in order to assess the quality of the height information in vegetated areas, and the filtering process has to be modelled as well. For example, a simple method can be designed using a difference model between the unfiltered and the filtered lidar heights. This difference model can then be subtracted from the predicted height shift.

The transferability of the features extracted from the lidar data to other scanners, flight conditions and regions seems to be difficult. For instance, lidar intensity values depend on many parameters (i.e. the scanner type, echo detection methods and intensity determination, characteristics of the emitted pulse, flight date, surface type etc.). For the general transferability of the method this approach uses ground truth data which adapt the features to the current conditions.

ACKNOWLEDGMENTS

This research has been financed by the Federal Ministry of Education and Research (BMBF) under project no. 03KIS050.

We acknowledge the support of our project partners: Department of Rural Area Husum (ALR), Federal Waterways Directorate (WSD) and the Lower Saxony Water Management, Coastal Defence and Nature Conservation Agency Division Norden-Norderney (NLWKN).

REFERENCES

Ahokas, E., Kaartinen, H., Hyyppä, J.(2003): A quality assessment of airborne laser scanner data. The International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV, 3/W13, Dresden, Germany.

Gorte, B., Pfeifer, N., Oude Elberink, S. (2005): Height texture of low vegetation in airborne laser scanner data and its potential for DTM correction. The International archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 3/W19, pp. 150-155.

Haralick, R. M. (1979). Statistical and structural approaches to texture. Proceedings of the IEEE 67(5), 786–804.

Hodgson, M. and P. Bresnahan (2004). Accuracy of Airborne Lidar-Derived Elevation: Empirical Assessment and Error Budget. Photogrammetric Engineering & Remote Sensing, Vol. 70, No. 3, March 2004, pp. 331-339.

Hopkinson, C., Lim, K., Chasmer, L.E., Treitz, P., Creed, I.F., Gynan, C. (2004): Wetland grass to plantation forest – estimating vegetation height from the standard deviation of lidar frequency distributions. The International archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 8/W2, pp. 288-294.

Moffiet, T., Mengerson, K., Witte, C., King, R., Denham, R. (2005): Airborne laser scanning: Exploratory data analysis indcates variables for classification of individual trees or forest stands according to species. ISPRS Journal of Photogrammetry & Remote Sensing 59 (2005), pp. 289-309.

Mundt, J. T., Streutker D. R., Glenn N.F. (2006): Mapping Sagebrush Distribution Using Fusion of Hyperspectral and Lidar Classifications. In Photogrammetric Engineering & Remote Sensing, Vol. 72, No.1, January 2006, pp. 47-54.

Niederöst, M. (2000): Reliable Reconstruction of Buildings for Digital Map Revision. The International Archives of Photogrammetry and Remote Sensing, ISPRS, Amsterdam, Netherlands, Vol. XXXIII, Part B3, pp. 635-642.

Oude Elberink, S. and M. Crombaghs (2004). Laseraltimetrie voor de hoogtemetingen van de kwelders Waddenzee. Technical Report AGI-GAP-2003-50 (in Dutch), AGI, RWS, The Netherlands.

Pandya, M.R. (2004): Leaf Area Index Retrieval Using Irs Liss-iii Sensor Data and validation of Modis Lai Product Over India. The International Archives of Photogrammetry and Remote Sensing, Vol. XXXV, Part B7, pp. 144-149.

Pfeifer, N., Gorte, B., Oude Elberink, S. (2004): Influences of Vegetation on Laser Altimetry – Analysis and Correction Approaches. The International archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part 8/W2, pp. 283-287.

Straub, B., Heipke, C. (2001): Automatic Extraction of Trees for 3D-City Models from Images and Height Data. Automatic Extraction of Man-Made Objects from Aerial and Space Images. Vol. 3., eds. Baltsavias, Gruen, van Gool, A.A.Balkema Publishers. Lisse Abingdon Exton(PA) Tokio. pp. 267-277.

Schwalbe, E. (2005): Geometric Modelling and Calibration of Fisheye Lens Camera Systems. The International archives of Photogrammetry, Remote Sensing and Spatial Information Sciences.Vol. XXXVI, Part 5/W8.

Wagner, W., Ullrich, A., Melzer, T., Briese, C., Kraus, K.(2004): From single-pulse to full-waveform airborne laser scanners: Potential and practical challenges. The International Archives of Photogrammetry and Remote Sensing, Vol. XXXV, Part B3, pp. 201-206.

EXTRACTION OF LINES FROM LASER POINT CLOUDS

Hermann Gross *, Ulrich Thoennessen

FGAN-FOM, Research Institute for Optronics and Pattern Recognition
76275 Ettlingen, Germany (gross,thoe)@fom.fgan.de

Commission III, WG III/3

KEY WORDS: Laser data, 3D point clouds, covariance of points, edge detection, line generation, eigenvalues, eigenvectors, momentum of inertia, segmentation

ABSTRACT:

Three dimensional building models have become important during the past for various applications like urban planning, enhanced navigation or visualization of touristy or historic objects. 3D models can increase the understanding and explanation of complex urban scenes and support decision processes. A 3D model of the urban environment gives the possibility for simulation and rehearsal, to "fly through" the local urban terrain on different paths, and to visualize the scene from different viewpoints. The automatic generation of 3D models using Laser height data is one challenge for actual research.

In many proposals for 3D model generation the process is starting by extraction of the border lines of man made objects. In our paper we are presenting an automatic generation method for lines based on the analysis of the 3D point clouds in the Laser height data. For each 3D point additional features considering the neighborhood are calculated. Invariance with respect to position, scale and rotation is achieved. Investigations concerning the required point density to get reliable results are accomplished. Comparing the new features with analytical results of typical point configurations provide discriminating features to select points which may belong to a line. Assembling these points to lines the borders of the objects were achieved. First results are presented.

Possibilities for the enhancement of the calculation of the covariance matrix by including the intensity of the Laser signal and a refined consideration of the neighborhood are discussed.

1. INTRODUCTION

Three-dimensional building models have become important during the past for various applications like urban planning, enhanced navigation or visualization of touristy or historic objects (Brenner et al., 2001). They can increase the understanding and explanation of complex scenes and support the decision process. The benefit for several applications like urban planning or the virtual sightseeing walk was demonstrated utilizing LIDAR data.

For decision support and operation planning the real urban environment should be available. In most cases the necessary object models are not present in the simulation data base. Especially in time critical situations the 3D models must be generated as fast as possible to be available for the simulation process.

Different approaches to generate the necessary models of the urban scene are discussed in the literature. Building models are typically acquired by (semi-) automatic processing of Laser scanner elevation data or aerial imagery (Baillard et al., 1999; Geibel & Stilla, 2000). For large urban scenes LIDAR data can be utilized (Gross & Thoennessen, 2005). Pollefeys (1999) uses projective geometry for a 3D reconstruction from image sequences. Fraser et al. (2002) use stereo approaches for 3D building reconstruction. Vosselman et al. (2004) describes a scan line segmentation method grouping points in a 3D proximity.

Airborne systems are widely used but also terrestrial Laser scanners are increasingly available. The latter ones provide a much higher geometrical resolution and accuracy (mm vs. dm) and they are able to acquire building facade details which are a requirement for realistic virtual worlds. Whereas in the orthogonal Nadir view of an airborne system the data can be interpreted as 2D image this is not possible for terrestrial Laser scanners.

We are presenting an approach for the segmentation of building parts like 3D edges. Analytical considerations give hints to extract these characteristic objects. We have realized and tested the detection of 3D edges as well as their approximation by lines. Also quality measures for the lines are determined. The capability of the algorithm is additionally demonstrated on the detection of overhead wires of a tram.

In chapter 2 the calculation of additional point features is described. The features are normalized with respect to translation, scale and rotation. The dependencies between covariance matrix and the tensor of momentum of inertia are discussed. Investigations on the sensitivity of the specified features deliver constraints concerning their usage.

In chapter 3 typical constellations of points are discussed and discriminating features are presented. Examples for the combination of eigenvalues and structure tensor are shown. For typical situations analytical feature values are derived.

The importance of a precise registration of Laser point clouds if different data sets have to be fused is illustrated in chapter 4.

The generation of lines is described in chapter 5. Points with the same eigenvectors are assembled and approximated by lines. Resulting 3D boundaries of objects are shown for different data sets.

In chapter 6 the possibilities using additional features are summarized. Outstanding topics and aspects of the realized method are discussed.

2. ADDITIONAL POINT FEATURES

A Laser scanner delivers 3D point measurements in an Euclidian coordinate system. For airborne systems mostly the height information is stored in a raster grid with a predefined resolution. Image cells without a measurement are interpolated by considering their neighborhood.

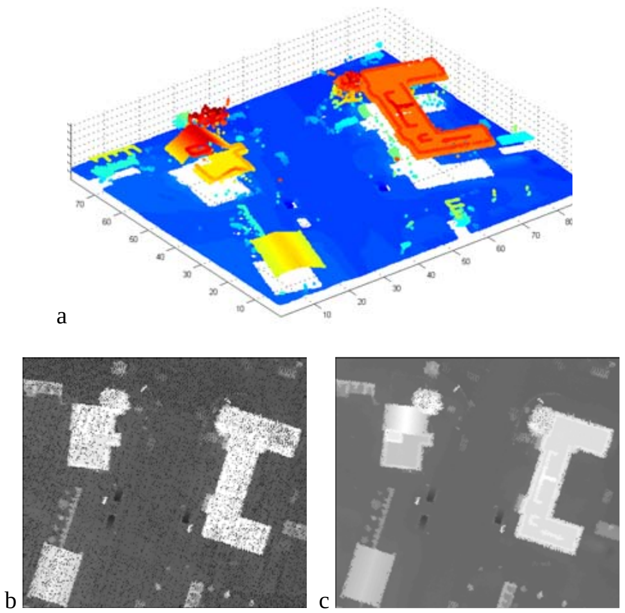


Figure 1. Point clouds from Toposys® Laser scanner
a) colored by height
Raster image based on point clouds:
b) without, c) with interpolated values

An example data set gathered by an airborne Laser scanner system as 3D points is shown in Figure 1a. The color corresponds to the height. A transformation to a raster image selecting the highest value for each pixel yields the Figure 1b. After filling missing pixels we are able to detect more details in Figure 1c. Due to the preprocessing steps the image does not represent the original 3D information anymore. The horizontal position is slightly different and some of the height values are calculated not measured. Additionally, sometimes more than one measurement for a resolution cell exists considering first and last echo or combining data of several measurement campaigns.

An example for a dense point cloud of a terrestrial Laser scanner is shown in Figure 2 representing the intensity of the signal.

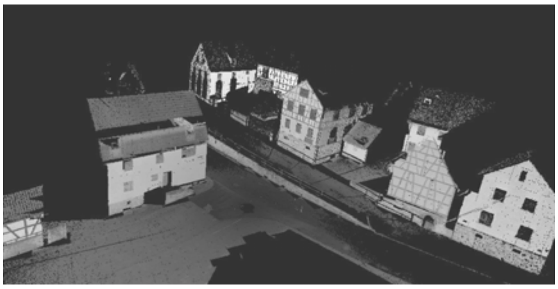


Figure 2. Point clouds colored by intensity

In contrary to the airborne data the projection of terrestrial Laser data along any direction is not very reasonable. Especially the combination of airborne (Figure. 1) and terrestrial (Figure. 2) Laser scanning data requires directly the analysis in the 3D data.

2.1 Moments

A 3D spherical volume cell with radius R is assigned to each point of the cloud. All points in a spherical cell will be analyzed. 3D moments as described by Maas & Vosselman (1999) are discussed and improved.

In a continuous domain, moments are defined by:

$$m_{ijk} = \int_V x^i y^j z^k f(x,y,z)\,dv\,, \tag{1}$$

where $i,j,k \in \mathbb{N}$, and $i+j+k$ is the order of the moment integrated over a predefined volume weighted by $f(x,y,z)$.

As weighting function the mass density can be used. It reduces to a constant value if homogeneous material is assumed.

Another possibility is to use the intensity of the reflected Laser beam (s. Figure 2, Figure 11) as weighting function. Some aspects of using the intensity signal were discussed in (Jutzi et al., 2005).

We restrict the order of moments to $i+j+k \le 2$. This delivers the weight, the center of gravity and the matrix of covariance. To be invariant against translation we calculate the center of gravity

$$\bar{x} = \frac{m_{100}}{m_{000}},\quad \bar{y} = \frac{m_{010}}{m_{000}},\quad \bar{z} = \frac{m_{001}}{m_{000}} \tag{2}$$

and the centralized moments

$$\bar{m}_{ijk} = \int_V \left(x-\bar{x}\right)^i \left(y-\bar{y}\right)^j \left(z-\bar{z}\right)^k f(x,y,z)\,dv \tag{3}$$

with $\bar{m}_{000} = m_{000}$. Scale invariance may be achieved by

$$\tilde{m}_{ijk} = \frac{\bar{m}_{ijk}}{R^{i+j+k}\,\bar{m}_{000}} \tag{4}$$

We need two normalizations because $f(x,y,z)$ can take a different physical unit (other than length).

In the discrete case the integral (3) is approximated by the sum

$$\bar{m}_{ijk}\left(x_a,y_a,z_a\right) = \sum_{l=1}^{N}\left(x_l-\bar{x}\right)^i \left(y_l-\bar{y}\right)^j \left(z_l-\bar{z}\right)^k f\left(x_l,y_l,z_l\right)\Delta v \tag{5}$$

including all points inside the sphere with radius R centered at an actual point $\begin{pmatrix} x_a & y_a & z_a \end{pmatrix}$ with the constraint

$$\left\| \begin{pmatrix} x_l & y_l & z_l \end{pmatrix} - \begin{pmatrix} x_a & y_a & z_a \end{pmatrix} \right\| \le R \tag{6}$$

Under the assumption that the incremental volume Δv is constant and due to the dependency of the moments from the number of points inside the sphere and the selected radius R we get the normalized moments

$$\tilde{m}_{ijk} = \frac{\bar{m}_{ijk}}{R^{i+j+k}\,\bar{m}_{000}} = \frac{\sum_{l=1}^{N}\left(x_l-\bar{x}\right)^i \left(y_l-\bar{y}\right)^j \left(z_l-\bar{z}\right)^k f\left(x_l,y_l,z_l\right)}{R^{i+j+k}\sum_{l=1}^{N} f\left(x_l,y_l,z_l\right)} \tag{7}$$

For constant weighting function $f(x,y,z)$ as used in many cases we get

$$\tilde{m}_{ijk} = \frac{\sum_{l=1}^{N}\left(x_l-\bar{x}\right)^i \left(y_l-\bar{y}\right)^j \left(z_l-\bar{z}\right)^k}{R^{i+j+k}N} \tag{8}$$

Neither the number of points nor the chosen physical unit for the coordinates, the radius and the weighting factor influences the values of the moments.

Finally we calculate for each point of the whole data set a symmetrical covariance matrix

$$M = \begin{pmatrix} \tilde{m}_{200} & \tilde{m}_{110} & \tilde{m}_{101} \\ \tilde{m}_{110} & \tilde{m}_{020} & \tilde{m}_{011} \\ \tilde{m}_{101} & \tilde{m}_{011} & \tilde{m}_{002} \end{pmatrix} \tag{9}$$

The calculation of the eigenvalues λ_i and eigenvectors $\vec{e}_i$ with $i = 1,2,3$ delivers features for each point. The eigenvalues are invariant concerning rotation.
If we calculate the tensor of momentum of inertia by

$$T = \left(\tilde{m}_{200} + \tilde{m}_{020} + \tilde{m}_{002}\right)\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - M \qquad (10)$$

instead of the moments M of order two we will get the same eigenvectors. The sum of the eigenvalues belonging to the same eigenvector is constant for each eigenvector.

$$\lambda_i(M) + \lambda_i(T) = \tilde{m}_{200} + \tilde{m}_{020} + \tilde{m}_{002} = const \quad \forall i = 1,2,3 \; (11)$$

Due to the non contiguous (discrete) calculation of the moments the quality of the resulting numerical invariants can be discussed in a statistical (as moments M) or a physical (as moments of inertia T) way considering each point not only as a point but as a representative physical part of its surrounding.

2.2 Point distribution in 3D space

In this section we discuss the influence of the distribution of point measurements concerning the proposed features.

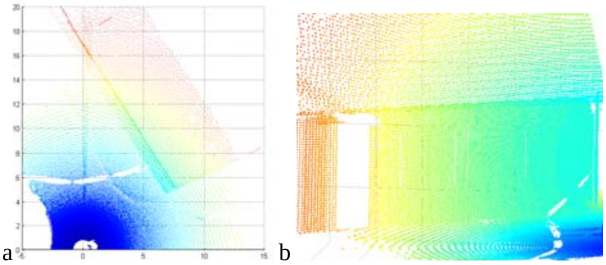


Figure 3. Point clouds of a terrestrial Laser scanner: a) vertical view, b) horizontal view; color indicates the distance to the sensor (blue=near, red=far away)

Figure 3 shows as an example for the dependency of the point density of the Zoller+Fröhlich Laser scanner concerning the distance to the sensor.
The comparable scan pattern of the Toposys sensor is shown in Figure 4a for a regular pattern and in Figure 4b for a wavy pattern. The point density in flight direction is usually much higher than in the perpendicular direction. In both cases there is no uniform distribution of the measured points.

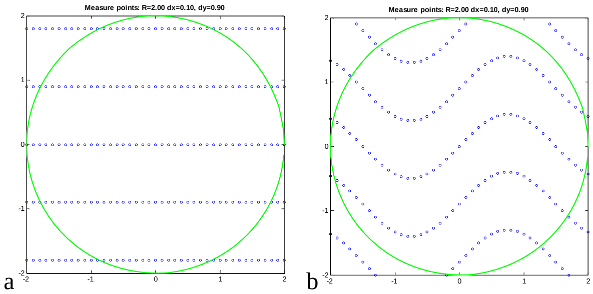


Figure 4. Scan pattern similar to the Toposys Laser scanner: a) regular pattern, b) wavy pattern

For non uniform distribution equations (1) and (5) imply to weight each point by the volume around this point without other points like inside a cell of a Voronoi diagram (Aurenhammer, 2000) or to correct the moments by integration over each cell of the diagram separately. To avoid such a time consuming but more precise calculation we have discussed the behavior of the eigenvalues of M dependent on the radius of the sphere and the density of the points. To investigate the behavior of the eigenvalues we have generated synthetically regular scans and also wavy scans (Figure 4) for a plane. After calculating covariance and eigenvalues taking all points inside the green circle we consider the ratio λ_2/λ_1 of the second and the greatest eigenvalue. The third eigenvalue is $\lambda_3 = 0$.

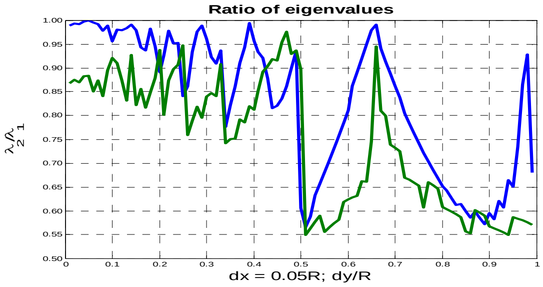


Figure 5. Ratio of λ_2/λ_1 dependent on the smaller point density dy/R : blue: regular pattern; green: wavy pattern

Figure 5 shows the ratio of the non zero eigenvalues dependent on the density of the points in the y-direction. Nearly the same behavior is calculated for both the regular and the wavy scan. The ratio for the regular pattern (blue) is slightly greater than for the wave pattern (green). The variations of the function are caused by the digitalization (Figure 4). For $dy/R < 0.5$, dy point distance in y-direction, we got acceptable results. Weighting each point by the same factor we have to select the radius of the sphere as $R > 2dy$ (two times of the largest point distance.) Under this constraint λ_2/λ_1 is greater than 0.75 (e.g. $dx = 0.1m \;\; dy = 0.5m \;\; \Rightarrow \;\; R > 1m$).

3. FILTERING OF POINTS

After calculation of the covariance matrix for each point in the data set considering a local environment defined by a sphere we have additional features for each point.

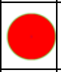
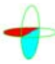
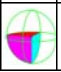
S	Type	λ_1	λ_2	λ_3
	Isolated point	0	0	0
	End of a line	$\frac{1}{12}$	0	0
	Line	$\frac{1}{3}$	0	0
	Half plane	$\frac{1}{4}$	$\frac{1}{4}\left(1-\frac{64}{9\pi^2}\right)=0.07$	0
	Plane	$\frac{1}{4}$	$\frac{1}{4}$	0
	Quarter plane	$\frac{1}{4}\left(1-\frac{2}{\pi}\right)=0.09$	$\frac{1}{4}+\frac{1}{2\pi}-\frac{32}{9\pi^2}=0.05$	0
	Two planes	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}-\frac{8}{9\pi^2}=0.03$
	Three planes	$\frac{1}{6}\left(1-\frac{1}{\pi}\right)=0.11$	$\frac{1}{6}\left(1-\frac{1}{\pi}\right)=0.11$	$\frac{1}{6}\left(1+\frac{2}{\pi}\right)-\frac{2^6}{3^3\pi^2}=0.03$

Table 1. Eigenvalues for some typical situations

These features are the center of gravity, the distance between center of gravity to the point, the eigenvectors, the eigenvalues and the number of points inside the sphere. They can be used for determination of object characteristics.

Table 1 shows the eigenvalues of the covariance matrix of some special point configurations. The first six rows present 2D cases the last two 3D ones.

The ratios are based on typical situations and analytically calculated. For an ideal line two eigenvalues are zero and one of it is greater than zero. For straight edges at the border of a half-plane one eigenvalue is zero and the ratio of $\frac{\lambda_2}{\lambda_1} = \frac{9\pi^2 - 64}{9\pi^2} = 0.28$ shows a significant difference between the both non zero eigenvalues. If we are looking for points inside a plane we have to compare the eigenvalues $\lambda_1 = \lambda_2 = 0.25 \quad \wedge \quad \lambda_3 = 0$ with the values for a plane. For the edge points at the intersection line of two orthogonal planes the ratios are $\frac{\lambda_2}{\lambda_1} = 0.5$ and $\frac{\lambda_3}{\lambda_1} = 0.5 - \frac{32}{9\pi^2} = 0.14$.

Figure 6a shows all points with eigenvalues satisfying the criteria for planes. The color indicates the object height. In Figure 6b only the edge points are drawn corresponding to Table 1 row 4.

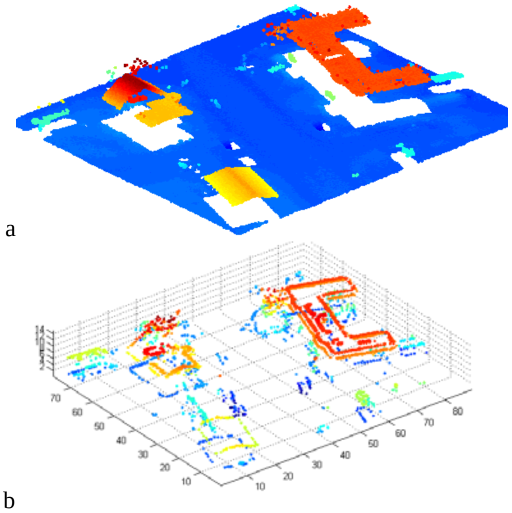


Figure 6. a) Points identified as plane points
b) Points with one high and two small eigenvalues

For object classification especially for region growing West (2004) uses the following features which depends on the eigenvalues:

$$\text{Structure Tensor Omnivariance} \quad = \sqrt[3]{\prod_{i=1}^{3} \lambda_i} \qquad (12)$$

$$\text{Structure Tensor Anisotropy} \quad = \frac{\lambda_1 - \lambda_3}{\lambda_1} \qquad (13)$$

$$\text{Structure Tensor Planarity} \quad = \frac{\lambda_2 - \lambda_3}{\lambda_1} \qquad (14)$$

$$\text{Structure Tensor Sphericity} \quad = 1 - \text{Anisotropy} = \frac{\lambda_3}{\lambda_1} \qquad (15)$$

$$\text{Structure Tensor "Eigenentropy"} \quad = -\sum_{i=1}^{3} \lambda_i \log(\lambda_i) \qquad (16)$$

$$\text{Structure Tensor Linearity} \quad = \frac{\lambda_1 - \lambda_2}{\lambda_1} \qquad (17)$$

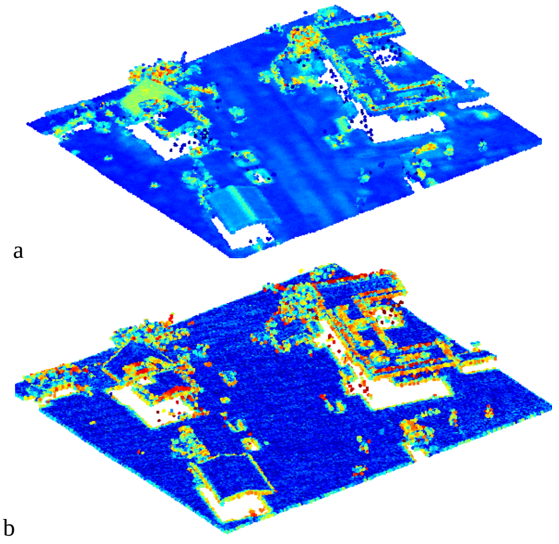


Figure 7. Points marked by a) Omnivariance b) Linearity

Figure 7 shows the points classified and colored by the features a) Omnivariance and b) Linearity. A detailed analysis of these features for point classification is under investigation.

4. REQUIREMENT FOR REGISTRATION

The enhancement of resolution is possible combining multiple scans of the same scene. We have investigated this approach for airborne Laser scan data (Toposys). Especially the reconstruction of gabled roofs was considered. A precise registration of the data sets is necessary.

The application of the filter process mentioned before delivered the result shown in Figure 8a. A detailed analysis shows some discrepancy in the registration of different scan data. Viewing along the ridge of the gabled roof, Figure 8a, demonstrates the gap between two flights.

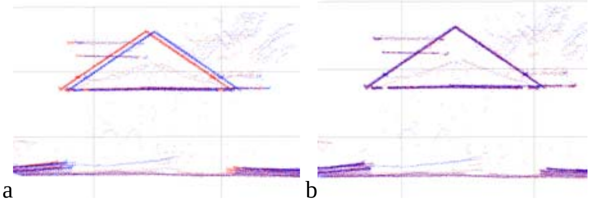


Figure 8. Gabled roof a) before and b) after fine registration

Using the Iterative Closest Point (ICP) algorithms (Besl 1992, Fitzgibbon, 2001) the registration was refined (Figure 8b). This method uses data of two point clouds inside a common region and determines translation, rotation and scaling to minimize the distance between the point clouds. Based on the transformed data acceptable eigenvalues for the classification of the planes of the gabled roof are achieved.

5. LINE GENERATION

All points marked as edge point may belong to a line. These points are assembled to lines by a grouping process. We consider the greatest eigenvalue λ_1 and its eigenvector $\overline{e}_1$. Consecutive points with a similar eigenvector, lying inside a small cylinder are grouped together and approximated by a line. Let Cl be the set of all points of the cloud. Starting with any point $\overline{p} \in Cl$ with eigenvector $\overline{e_1^p}$ as feature. This point is called the trigger point. Now we are looking for all points $\bar{c}$ and determine the set

$$C = \left\{ \bar{c} \in Cl \left\| \overline{e_1^p} \circ \overline{e_1^c} \right| > \text{min\_cos} \right\} . \tag{18}$$

This set contains all points with nearly the same or opposite direction for the first eigenvector tested comparing the inner product of two vectors against a given threshold min_cos . We construct a line through the trigger point along its first eigenvector:

$$\bar{g} = \bar{p} + \mu \overline{e_1^p} \tag{19}$$

The scalar components for $\bar{c} \in C$ to each eigenvector are

$$\mu_i\left(\bar{c}, \bar{p}\right) = \left(\bar{c} - \bar{p}\right) \circ \overline{e_i^p} \; . \tag{20}$$

Due to the normalization of the eigenvectors these components describe the distances along each direction. The distance of the point $\bar{c}$ to the line is

$$d\left(\bar{c}, \bar{p}\right) = \sqrt{\mu_2^2\left(\bar{c}, \bar{p}\right) + \mu_3^2\left(\bar{c}, \bar{p}\right)} \tag{21}$$

Let $D = \left\{ \bar{c} \in Cl \middle| d\left(\bar{c}, \bar{p}\right) \le \text{max\_d} \right\}$ be the set of edge points inside the cylinder given by $\bar{g}$ with the given radius max_d . The intersection $GP = C \cap D$ includes all edge points with nearly the same first eigenvector as the trigger point and not far away from the straight line given by the trigger point and its first eigenvector.

Collinear edges of different buildings in a row may belong to GP ($\bar{p}$). Therefore we examine the contiguity of the points in the neighborhood of $\bar{p}$. The scalar values $\mu_1\left(\bar{c}, \bar{p}\right)$ describe the projection of the points onto the straight line. Let $\mu s\left(\bar{c}, \bar{p}\right)$ a sorted list of the $\mu_1\left(\bar{c}, \bar{p}\right)$. Because $\mu s\left(\bar{p}, \bar{p}\right) = 0$, we have to search for gaps defined by an acceptable value max_gap on the left and the right side of zero. $\mu s_L \le 0$ is the left boundary and $\mu s_R \ge 0$ is the right boundary if

$$\begin{aligned} &\mu s_{L-1} + \text{max\_gap} < \mu s_L \;\; \wedge \;\; \mu s_R + \text{max\_gap} < \mu s_{R+1} \\ &\quad \wedge \;\; \mu s_{j-1} + \text{max\_gap} \ge \mu s_j \;\; \forall \text{L} < \text{j} \le \text{R} \end{aligned} \tag{22}$$

Let $GPs = \left\{ \bar{c} \in GP \middle| \mu s_L \le \mu\left(\bar{c}, \bar{p}\right) \le \mu s_R \right\}$ the set of points along the straight line without gap with respect to $\bar{p}$. For determination of the line we calculate the mean values $\overline{cm} = \frac{1}{n} \sum_{\bar{c} \in GPs} \bar{c}$ where n is the number of points in GPs . The direction of the line is given by the eigenvector $\bar{e}_1$ belonging to the greatest eigenvalue of the covariance matrix CM . The elements of the matrix are

$$cm_{ij} = \frac{1}{n} \sum_{\bar{c} \in GPs} \left(x - xm\right)^i \left(y - ym\right)^j \left(z - zm\right)^k \tag{23}$$

where

$$\left(x \quad y \quad z\right) = \bar{c} \quad \text{and} \quad \left(xm \quad ym \quad zm\right) = \overline{cm} \tag{24}$$

The straight line is described by $\overline{xl} = \overline{cm} + \mu \bar{e}_1$. Start point and endpoint are given by

$$\overline{xa} = \overline{cm} + \min_{\bar{c} \in GPs}\left(\bar{c} \circ \bar{e}_1\right)\bar{e}_1 \quad \text{and} \quad \overline{xe} = \overline{cm} + \max_{\bar{c} \in GPs}\left(\bar{c} \circ \bar{e}_1\right)\bar{e}_1 \tag{25}$$

The length of the line is

$$L = \left\| \overline{xe} - \overline{xa} \right\| \tag{26}$$

The eigenvalues of CM can be normalized by $\nu_i = \frac{\lambda_i}{L^2}$ to be independent from length. These normalized eigenvalues are reasonable for a quality assessment of the lines. The same process is repeated for all points not assigned to a line until each point belongs to a line or can not generate an acceptable line.

Figure 9 shows the results of the line generation for the data set shown in Figure 1. The color indicates the height of the lines. The eaves as well as the ground plan of the buildings are approximated by lines. For the detection of the ridge of the saddle roof we have to use other thresholds for the eigenvalues especially for roofs with small inclination.

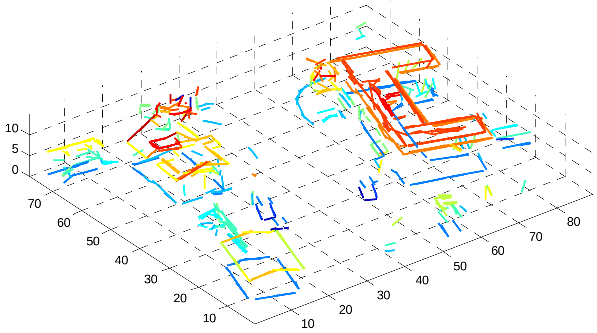


Figure 9. Lines generated from edge points

For the scene from Figure 3 we got the approximation lines shown in Figure 10. The ridge line, the contour lines at the bottom of the building and the boundary lines of the door are detected.

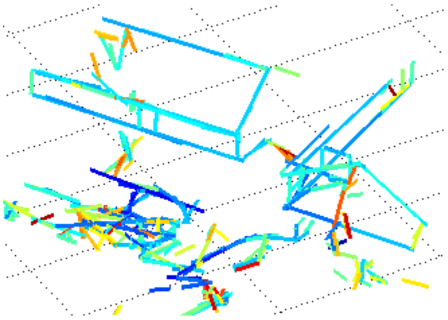


Figure 10. Lines generated from edge points for the point clouds of a terrestrial Laser scanner (s. Figure 3) colored by the 1. eigenvalue

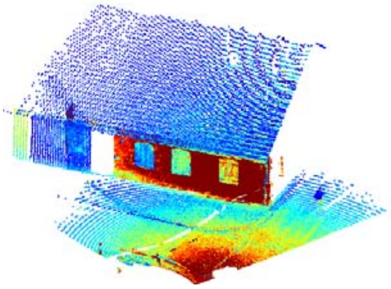


Figure 11. Building of Figure 3 colored by intensity

Considering the intensity of the Laser scanner signal of the same scene (Figure 11) we will investigate the reconstruction of windows. More tests have to be accomplished to stabilize the method.

The proposed method delivers not only edges of buildings but also the overhead wires of tramways in a city. For data from the Toposys sensor Figure 12 displays the Last- and First-Echo and Figure 13 shows the generated lines of the power lines and the support wires.

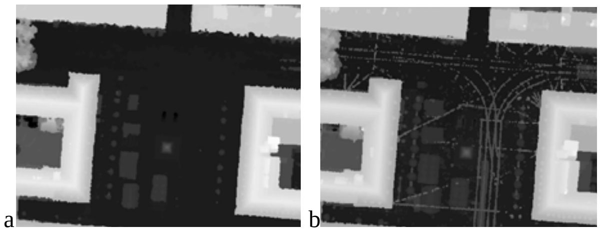


Figure 12. LastEcho and FirstEcho of a city scene

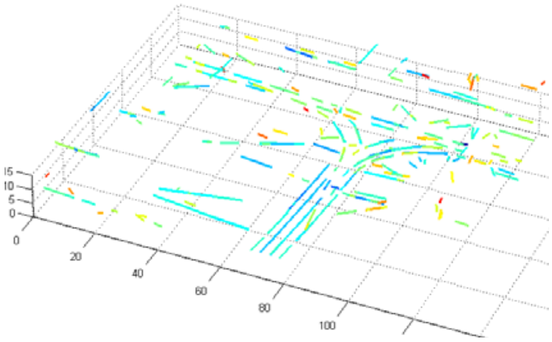


Figure 13. Lines generated from edge points for overhead wires

6. CONCLUSION AND OUTLOOK

Laser scanner systems gather directly 3D information. For data reduction and visualization the data sets are transformed often to a raster grid interpolating gaps. Due to this step the original 3D data is tampered.

For terrestrial Laser scan data this method is more difficult to apply and tampering error may be larger. Additional problems will appear if we want to fuse airborne and terrestrial data sets. We propose the exploitation of the original 3D point clouds.

Additional features for each point of the cloud are calculated from the covariance matrix including all neighbor points. The neighborhood is defined by a sphere. The quality of the resulting eigenvalues and the eigenvectors of the matrix depends on the resolution and the number of points inside the sphere. For different resolutions of different scan directions these values are discussed. Based on this investigation the radius of the sphere can be calculated by a function of the resolution. The new features are invariant with respect to position, rotation and scale.

The additional features are appropriate for classification of the points as edge, corner, plane or tree points. For some typical situations analytically determined eigenvalues are opposed to calculated eigenvalues of real data for comparison. The greatest eigenvalue is used for filtering edge like points.

The described method for generation of lines combines consecutive points with the same eigenvector inside a small cylinder without any gap. The presented results are promising.

Further investigations are planned concerning the fusion of the data on basis of the point clouds and/or on a higher level of lines. For the filtering process features derived from the eigenvalues (12)-(17) should be tested on different kind of data to get a robust point classification.

A further topic is the construction of planes assembling plane like points.

A calculation of the covariance matrix which is adapted to the resolution should be investigated and may deliver better results. This process is expensive and should be tested on several data sets.

LITERATURE

Aurenhammer F., Klein R., 2000, Voronoi Diagrams, Ch. 5 in *Handbook of Computational geometry*, Ed. J.-R. Sack and J. Urrutia, Amsterdam Netherland: North-Holland, pp. 201-290

Baillard, C., Schmid, C., Zisserman, A. and Fitzgibbon A., 1999. Automatic line matching and 3D reconstruction from multiple views. In: *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery*, Vol. 32.

Besl P.J., McKay N.D. 1992. A Method for registration of 3D shapes, *IEEE Trans. Pattern Anal. and Machine Intell.* 14 (2), 239-256

Brenner, C., Haala, N. and Fritsch, D., 2001. Towards fully automated 3D city model generation. In: *E. Baltsavias, A. Grün and L. van Gool (eds), Proc. 3rd Int. Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*.

Fitzgibbon A. W., 2001. Robust Registration of 2D and 3D Point Sets, *Proceedings of the British Machine Vision Conference*, 2001, 662-670

Fraser C. S., Baltsavias E., Gruen A., 2002. Processing of IKONOS Imagery for Submetre 3D Positioning and Building Extraction. *ISPRS Journal of Photogrammetry & Remote Sensing* 56, 177-194

Geibel R., Stilla U., 2000. Segmentation of Laser-altimeter data for building reconstruction: Comparison of different procedures. *International Archives of Photogrammetry and Remote Sensing*. Vol. 33, Part B3, 326-334

Gross H., Thoennessen U., 2005. 3D Modeling of Urban Structures. Joint Workshop of ISPRS/DAGM Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring CMRT05, *Int. Archives of Photogrammetry and Remote Sensing*, Vol. 36, Part 3/W24, pp. 137-142

Jutzi B., Neulist J., Stilla U., 2005. Sub-Pixel Edge Localization Based on Laser Waveform Analysis, ISPRS WG III/3, III/4, V/3 *Workshop "Laser scanning 2005", Enschede, the Netherlands*, pp. 109-114

Maas H., Vosselman G., 1999. Two algorithms for extracting building models from raw Laser altimetry data, *ISPRS Journal of Photogrammetry & Remote Sensing 54*, pp. 153-163

Pollefeys M., 1999. Self-Calibration and Metric 3D-Reconstruction from Uncalibrated Image Sequences, PhD-Thesis, K. U. Leuven

Vosselman, G. et al., 2004. Recognizing structure in Laser scanner point clouds. Int. *Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 46, part 8/W2, Freiburg, Germany, pp. 33-38.

West, K. F. et al., 2004. Context-driven automated target detection in 3-D data. *Automatic Target Recognition XIV, edited by Firooz A. Sadjadi, Proceedings of SPIE* Vol. 5426, pp. 133-143.

ADDING THE THIRD DIMENSION TO A TOPOGRAPHIC DATABASE USING AIRBORNE LASER SCANNER DATA

Sander Oude Elberink and George Vosselman
Department of Earth Observation Science, International Institute for Geo-Information Science and Earth Observation – ITC Enschede, The Netherlands
{oudeelberink, vosselman}@itc.nl

Commission III, WG III/3

KEY WORDS: 3D Reconstruction, topographic features, data fusion, laser scanner data, segmentation, modelling.

ABSTRACT:

Laser altimetry provides reliable and detailed 3D data, which to certain extent, can be processed (semi-)automatically into 3D information. The use of an additional source of information, like 2D GIS data, can improve the reconstruction process, especially in terms of time and reliability. This paper describes the reconstruction of 3D topographic objects by fusing medium scale map data with the national height model, acquired by airborne laser altimetry. We assume that the topographic objects can all be described by smooth surface patches. We therefore first process the laser data to extract the larger smooth surfaces. Discontinuities are, however, preserved. The resulting set of laser points is used to first assign heights to the lines of the 2D GIS data and later on to reconstruct the surfaces of the objects. A set of processing rules is used in the first step to obtain the most likely heights of the object outlines. A constraint Delaunay triangulation of combined 3D outline points and laser points is used for the surface reconstruction. The developed method is demonstrated with a 3D reconstruction of a complex motorway interchange.

1. INTRODUCTION

With the growing demand for 3D topographic data the need for automated 3D data acquisition also grows. Over the past 10 years several researchers proposed methods to acquire 3D topographic data. Many of them focussed on 3D reconstruction of man-made objects, (Haala et al., 1998; Rottensteiner and Briese, 2002; Vosselman, 1999). Automated methods for reliable and accurate 3-D reconstruction of man-made objects are essential to many users and providers of 3-D city data, including urban planners, architects, and telecommunication and environmental engineers (Henricsson and Baltsavias, 1997).

Laser altimetry provides reliable and detailed 3D data, which to certain extent, can be processed (semi-)automatically into 3D information. The use of an additional source of information, like 2D GIS data, can improve the reconstruction process, especially in terms of time and reliability.

This paper describes the reconstruction of 3D topographic objects by fusing medium scale map data with the national height model, acquired by airborne laser altimetry. This topic is part of a larger research project handling the data modelling, acquisition and analysis of national 3D topographic databases.

In section 2 we first describe related work on 3D reconstruction from laser scanner data. The datasets, advantages of merging information and the properties of an extension of a topographic database to 3D are discussed in section 3. In section 4 we describe the approach to derive the 3D topographic information. Adding height to a 2D topographic database not only requires assigning heights to the object boundaries, but also needs the introduction of surface descriptions. Results of the 3D reconstruction of a complex highway interchange are shown and discussed in section 5.

2. RELATED WORK

Over the past ten years airborne laser scanning has broadened its application fields from a suitable technique for the acquisition of digital terrain models, to more detailed reconstruction tasks like the acquisition and modelling of 3D (topographic) objects (Maas, 2001). When used for the 3D reconstruction of buildings the increasing amounts of points contain more and more information about the shape of buildings. Therefore methods for 3D reconstruction can be more data driven and need less specific object models (Vosselman, 1999).

There are several papers concerning the reconstruction of objects from laser data without using additional information sources like 2D maps or aerial images. Most of them discuss the geometric reconstruction of *buildings* in dense laser scan data, (Vosselman, 1999), (Maas and Vosselman, 1999), (Rottensteiner and Briese, 2002), (Elaksher and Bethel, 2002). (Maas and Vosselman, 1999) suggest when using laser altimetry data with a point density of 0.1 point / m^2 or less, the use of GIS data is necessary to successfully reconstruct building roofs. (Rottensteiner and Briese, 2002) also suggest to use image edges for matching roof edges, to improve their building extraction results. In (Rottensteiner and Briese, 2003) they present the use of image segments to find planar regions and use image edges to fit wire frames.

The use of an additional source of information can improve the reconstruction process, especially in terms of time and reliability. Several papers describe the advantage of using both laser data and 2D maps. 2D maps provide outlines, classified polygons and topologic and 2D semantic information. Although most of the papers in this field discuss the reconstruction of *buildings*, (Haala et al., 1998), (Brenner, 2000), (Vosselman and Dijkman, 2001), (Overby et al., 2004), (Hofmann, 2004) and (Schwalbe et al., 2005), there are some authors handling the reconstruction of *other topographic objects*, like roads in (Vosselman, 2003) and (Hatger and Brenner, 2003), roads and lakes in (Koch, 2004), or unclassified break lines (Briese, 2004). The purposes for integrating map data and laser data vary from improving the filtering process for DTM generation by explicitly modelling 3D breaklines (Briese, 2004) to rapid acquisition of 3D city models for virtual reality applications (Haala et al., 1998).

In this research we recognise and model height discontinuities between objects that are adjacent in a 2D topographic database. For modelling the surfaces of the 3D topographic object a point cloud segmentation algorithm is used. This algorithm preserves height discontinuities, but eliminates small objects like cars and traffic signs that should not be included in the 3D topographic database. Filtering algorithms are also used to select the correct laser points for modelling the object surfaces.

3. DATA PROPERTIES

3.1 Data sources

This research is a part of a project to develop methods for acquiring, storing, and querying 3D topographic data as a feasibility study for a future national 3D topographic database. Usage is therefore made of the current national 2D topographic database TOP10vector and the national elevation model AHN.

TOP10vector is a digital 2D topographic database for usage at a scale around 1:10.000. It has been built up in a fully coded object structure. The database is acquired from photographs in a 1:18.000 scale and has an accuracy of 1-2 m. Small buildings like houses, are stored in a different layer and are not shown in figure 1.

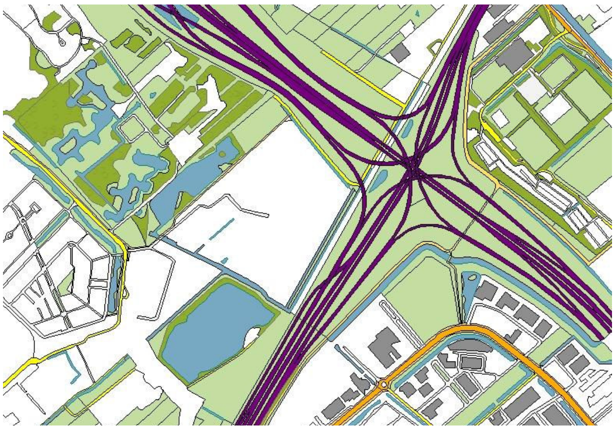


Figure 1: The study area in the TOP10vector database.

The national Height model of the Netherlands (AHN) has an average point density of 1 point per 16 m² or better and a height precision of about 15 cm standard deviation per point. In the standard production process the laser data has been filtered, removing buildings, trees and outliers. This filtered dataset will normally be interpolated to a regular grid, and delivered in grid sizes of 5, 25 and 100 meter. However, in this project the original, unfiltered irregular point cloud has been used in order to use as much information from the point cloud as possible (Figure 2).

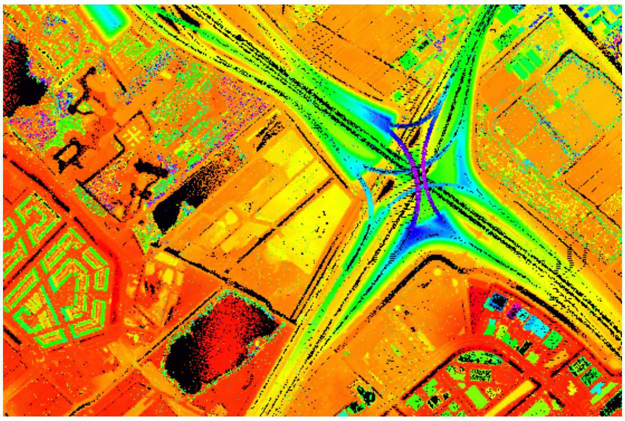


Figure 2: Colour coded AHN elevation data of the study area.

3.2 Data fusion

The existing topographic data delivers a large amount of topological and semantical information. Objects in topographic maps have been classified by human interpretation of aerial images. In this step the outlines, classification and semantics of topographical features are being stored for every object. We describe four different examples, showing how 2D map data can be used to better process the laser data:

1. Outlines. Although there might be small planimetric discrepancies between map data and laser altimetry data, the map data delivers information at object edges where there might be a change of class, resulting in break lines in the height data. Outlines can also be used as input for partitioning the 2D object (Haala et al., 1998), (Vosselman and Dijkman, 2001).
2. Classification in relation to individual laser points: Because the ground structure at the earth surface has influence on the characteristics of the returned laser pulse (Jutzi and Stilla, 2003), (Pfeifer et al., 2004), this class information will be used as input knowledge to further process the laser data.
3. Classification in relation to groups of laser points. Where the previous step focussed on the behaviour of individual laser pulses, the class information can be extended to groups of laser points. Lakes should be horizontal, roads should be smooth, and vegetated areas can show varying heights. Using the information that roads should be smooth in 3D, helps to determine filter parameters for road polygons, filtering out laser points reflected on small objects like cars, containers, traffic lights etc.
4. Semantics. One step further is the implementation of knowledge about an object in relation to its neighbouring objects. A good example is given in [Koch, 2004] where the object 'lake' has not only to fulfil internal constraints (the lake should be horizontal), but it also has to lie below its neighbouring objects. To give another example, reconstructing two intersecting roads should result in a smooth surface at the junction.

3.3 Features & representation

In the 2D map used in this project, road segments are represented by closed polygons. Its geometry has been defined by the coordinates of vertices and the topology. In the map implicit height information can be stored by adding 'hidden' objects classifications to polygons covering locations with multiple land use. Figure 3a shows that the middle polygon has two classification attributes: 'visible road 1' and 'hidden road 2'. Figure 3b clarifies that adding height to 2D vertices is not enough to get a 3D model. At a certain point the terrain will connect the upper road with the lower road; part of the edges between terrain and road, which were connected in 2D do not connect to each other in 3D. This means that additional 3D edges have to be created for overlapping objects. Our task is to derive a method which automatically determines the location and shape of the interchange by adding laser data to map data. In the next chapter we describe a method, which integrates object knowledge into the reconstruction of 3D infrastructural objects.

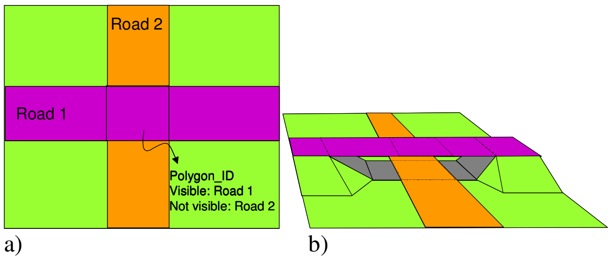


Figure 3: Fly-over in a 2D (a) and 3D representation (b).

4. APPROACH

4.1 Pre-processing 2D map

As shown in figure 3b, edges that are straight in the 2D map do not need to be straight in the 3D model. To correctly capture the shape of the infrastructural objects, the edges therefore need to be described by more points. For this purpose, points were inserted into the edges of the polygons at every 10 m. For all these points and the original map points the height needs to be determined from the laser data.

4.2 Segmentation

We assume that the topographic objects can all be described by smooth surface patches. The purpose of the point cloud segmentation is therefore to find piece-wise continuous surfaces that can be used to infer the heights of the topographic objects. Traditional filter algorithms that are used to produce digital elevation models often completely or partially remove objects like bridges and road crossings (Sithole and Vosselman, 2004). By segmenting a scene into piece-wise continuous patches and further classifying the segments this problem can be avoided (Sithole and Vosselman, 2005); (Tóvári and Pfeifer, 2005).

In our case, we do not perform a classification of the segments, but just use the segmentation results to eliminate laser points on small objects like cars, light poles, traffic signs, and trees. By requiring a minimum segment size, all these points will be left without a segment number after the segmentation step and can be easily removed.

For the segmentation of the point cloud a surface growing algorithm is used with some modifications that allow a fast processing of large datasets (Vosselman et al., 2004). The surface growing method consists of a seed surface detection followed by the actual growing of the seed surface. For the detection of seed surfaces we employ the 3D Hough transform. This transform is applied to the k nearest points of some arbitrary point. If the Hough transform reveals that a minimum number of points in this set is located in a plane, the parameters of this plane are improved by a least squares fit and the points in this plane constitute the seed surface. To speed up the seed detection, we do not search for the optimal seed (with most points in a plane and the lowest residual RMS of the plane fit), but start with the growing once an acceptable seed surface is found.

In the growing phase we add a point to the surface if the distance of the point to a locally estimated plane is below some threshold. This threshold is set such that some amount of noise is accepted. At the same time is also serves to allow for a small curvature in the surface. For a faster processing, the normal vectors of points are not computed and checked. The distance of a point to the local plane is the only criterion. If a point is accepted as an expansion of the surface, a local plane needs to be assigned to this point. In case the distance computed for this point was very small, no new local plane is estimated, but the plane parameters of the neighbouring surface point is copied to the new point. This strategy again serves a faster processing of the point cloud. Once no more points can be added to a surface, the seed detection is repeated. This process continues until no more seed surfaces are found.

4.3 3D reconstruction method

The first step of adding the third dimension to the map is to assign heights to the boundaries of all map objects. In many cases, two objects that are adjacent in 2D are also adjacent in 3D. In some cases, however, there will be a clear height difference for (a part of) the boundary that the objects share in 2D. Assigning the proper heights to the object outlines then requires the introduction of additional lines in the database (cf. section 3.3).

For each point in the map lines after the densification (section 4.1), the objects with boundaries containing this point are selected. For each of the objects around a point the height is derived from the laser points inside the object outline. For this purpose the segmentation results are used. First the k laser points that are nearest to the map point are selected. Next it is determined which segment number is most frequent among the selected laser points. A plane is fitted through the laser points of the most frequent segment number and the height of this plane at the location of the map point is taken as the boundary height. The usage of the most frequent segment number has proven useful in cases of a slight misregistration between the map and the laser data. In this case points of a high object may be located inside the boundaries of an adjacent low object or vice versa. A straightforward fitting of a surface to all laser points near the map point would then lead to errors. The selection of the points of with the most frequent segment number makes the height assignment more robust.

Once a height has been estimated for all objects around a point, it needs to be determined whether objects with similar heights should share the same 3D boundary point. A series of processing rules is used to make this decision:

- If a water and a meadow object are adjacent, the height of the meadow boundary point is set to the height of the water level. This ensures that the shores of water areas are horizontal (Koch, 2004).
- If there is a small height difference between two objects of the same type, a common 3D boundary point is used with the average height of the two objects.
- If there is a small height difference between a road object and another object, the height estimated for the road object is taken as the height of a common 3D boundary point. This

rule is used because the heights on (the very smooth) road surfaces can be estimated more accurately.

Figure 4 shows an example of a few road and meadow objects of a road junction. At the locations where the road surface is above the ground level, additional object lines are introduced to model the height difference.

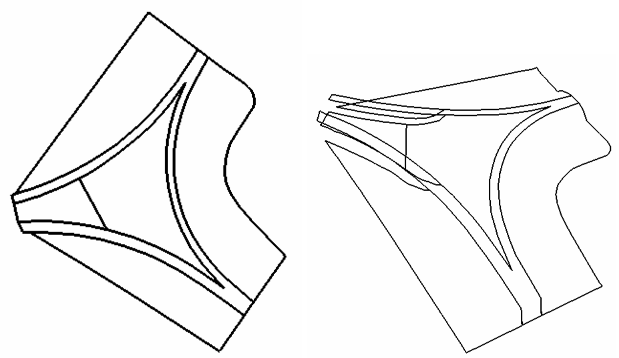


Figure 4: 2D map lines of a few road and meadow objects (left) and perspective view on the reconstructed 3D object boundaries (right).

4.4 Surface modelling

In the previous section laser data has been used to assign heights to the dense map points, which are situated on the object boundaries. Adding height to a 2D object not only means giving height to the boundaries of this object, but also to the surface of the object. Most of the objects show some relief at its surface, like structures on the roof of a building and height differences in grasslands.

To obtain a realistic surface model, a Delaunay triangulation was performed with the set of dense 3D map points combined with the set of laser points. However, road and water objects are triangulated without using the laser points. The motivation is that the resulting 3D road will be smoother, which can be seen as a generalization choice in 3D. Implicitly the laser points on the road segments already gave their height information to the map points, as described in section 4.3. In all triangulations the object boundaries have been added as constraints.

Morphological filtering has been applied to prevent unwanted spikes near edges between roads and meadow. These spikes are caused by misregistrations between the laser and map data, e.g. when laser points are located within meadow polygons but actually lie on upper roads of the interchange. These mistakes did not influence the height determination of the map points (in section 4.3), because a plane was fitted through a dominant segment of laser points. However, when adding individual laser points to the surface these errors show up as steep triangles in the TIN, and have to be removed. This filtering is performed for each object separately.

In 3D, road objects can be modelled as volume objects, instead of surface objects. At this moment we have added a fixed, predefined thickness of 1 meter, underneath the road surface to improve the visualisation at interchanges and flyovers. In the future terrestrial laser data will be integrated to be able to model the object parts which can not be seen from aerial laser and image data. For visualisation purposes the boundary representation has been converted to VRML 2.0 format.

5. RESULTS

Figure 5 shows the result of an important preprocessing step on the laser data: removing small segments from the point cloud. It can be seen that many small features like cars and bushes are being removed in this step.

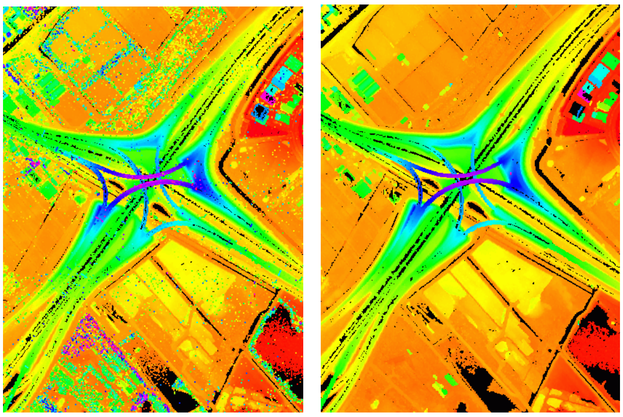


Figure 5: Laser scanner data before (left) and after (right) the removal of small segments. Black areas contain no laser points.

Note that on some parts of the roads even in the unfiltered data set only a few laser points return from the surface. This type of asphalt partly absorbs the laser pulse, resulting in lower point density on road objects. Only for small 2D road objects the low point density results in unreliable 3D reconstruction (cf. figure 10).

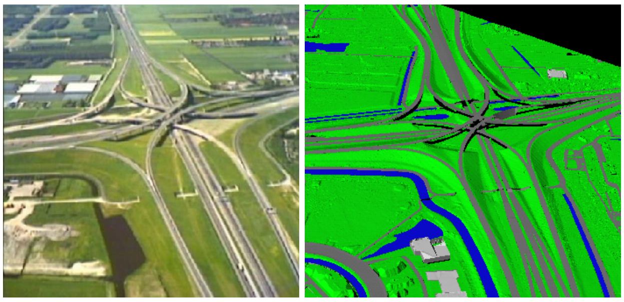


Figure 6: Aerial photograph of the motorway interchange (© Picture archive of the Ministry of Transport, Water Management and Public Works) and reconstructed model.

Figure 6 illustrates the motorway interchange on an oblique photograph (left) and as reconstructed model (right). As the picture is taken in 1983, a few objects have changed over time. In figure 7 the reconstructed model of the test region is shown. All objects have kept their classification type of the 2D map (cf. figure 1). For simplicity reasons, we choose to assign all objects to four classes: road (grey), meadow (green), water (blue) and building (light grey). The focus is on the reconstruction of infrastructural objects and the connections to the terrain. In the upper left part of the scene two large spikes show up. The selection of suitable laser points for plane fitting for the height determination of the map points has failed there. The reason is that the laser data ends a few meters behind those map points.

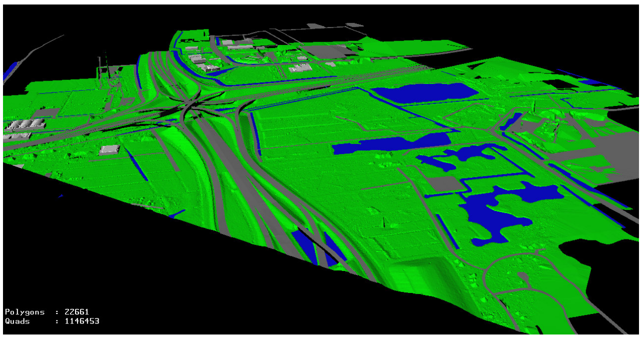


Figure 7: Overview of reconstructed scene with complex infrastructural objects.

In the next figures we will discuss this result in detail. Figure 8 & 9 show results for our reconstruction method. Water objects are horizontal and the neighbouring meadow objects connect to the water boundaries. The upper road in figure 8 is reconstructed above the water and the other road and connects to terrain at the correct position. Note that the black objects underneath the flyovers are still holes in the model. These holes will be filled up in a later stage, either in an integration process with terrestrial laser scanner data or by adding other information to the model. This information can be in the form of object knowledge: most of the holes can be filled up by interpolation between the two neighbouring objects.

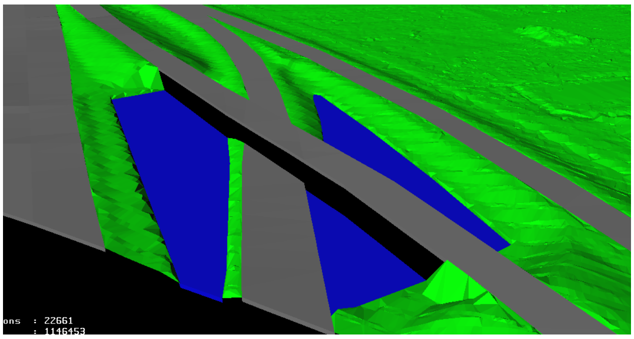


Figure 8: Reconstructed interchange, together with water and meadow objects.

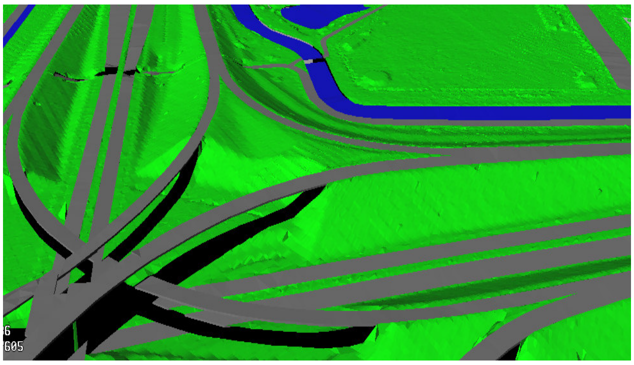


Figure 9: Result for the reconstruction of the body of the flyover, and the flying roads.

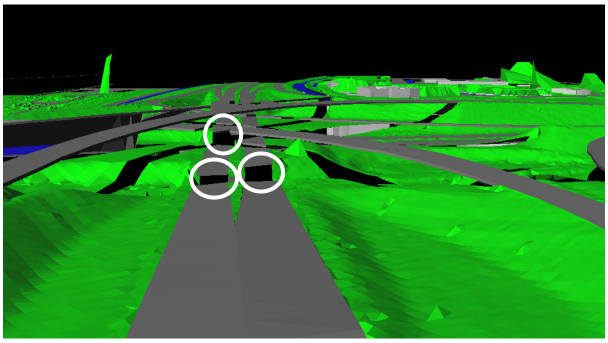


Figure 10: Holes due to hidden object parts and lack of suitable laser points. The white circles show the locations of three holes.

Figure 10 shows that some road object parts are missing on the lower region of the flyover. For some parts the reason is that there is a reconstructed road object on an upper level of the flyover, resulting in gaps at all lower levels. Another reason for missing parts is that the number of laser points may become too small to reliably fit a plane through these laser points, as we already have seen in figure 5. This means that the boundaries of these object parts cannot be determined in 3D. We decided not to add those unreliable parts in the model. Additional knowledge has to be put into the reconstruction process to constrain the connectivity between object parts, which represent the same real world object.

6. CONCLUSION & OUTLOOK

We have presented a method that recognises and models height discontinuities between objects that are adjacent in a 2D topographic database. A segmentation algorithm has successfully been used to connect laser points on smooth surfaces and remove small segments. First, the 3D boundaries have been determined by fitting planes to neighbouring dominant laser segments. Several connection rules have been applied to get a tight model at object boundaries. Several conditions have been applied to get horizontal lakes and smooth roads. At interchanges and flyovers additional boundaries have automatically been reconstructed to allow the reconstruction of 3D objects.

In the near future we will focus on how to add missing polygons to hidden objects. Knowledge about semantics and topology will be integrated with reconstruction method in order to overcome the lack of laser points on hidden objects. Together with other research partners we are working on the modelling of volume objects in a TEN data structure. This gives the opportunity to reconstruct 3D models with 3D primitives instead of with 2D surfaces. Next, focus will be on the detailed reconstruction of buildings, by fusing higher point density laser data with large scale topographic maps.

ACKNOWLEDGEMENT

This research is partially funded by the Dutch BSIK research programme Space for Geo-Information, project 3D Topography. The authors would also like to thank the Topographic Service of the Dutch Cadastre as well as the Steering Committee AHN for providing the data.

REFERENCES

Brenner, C., 2000. Towards Fully Automatic Generation of City Models, XIX ISPRS Congress. IAPRS, Amsterdam, The Netherlands, pp. 85-92.

Briese, C., 2004. Three-Dimensional Modelling of Breaklines From Airborne Laser Scanner Data, XXth ISPRS Congress: Geo-Imagery Bridging Continents. IAPRS, Istanbul, Turkey.

Elaksher, A. and Bethel, J., 2002. Reconstructing 3D Buildings From Lidar Data, Symposium 2002 Photogrammetric Computer Vision. IAPRS, Graz, Austria.

Haala, N., C. Brenner and Anders, K.-H., 1998. 3D Urban GIS From Laser Altimeter and 2D Map Data, ISPRS Commission IV – GIS Between Visions and Applications. IAPRS, Ohio, USA.

Hatger, C. and Brenner, C., 2003. Extraction of Road Geometry Parameters From Laser Scanning and Existing Databases, ISPRS working group III/3 workshop `3-D reconstruction from airborne laserscanner and InSAR data'. IAPRS, Dresden, Germany.

Henricsson, O. and Baltsavias, E., 1997. 3-d building reconstruction with aruba: A qualitative and quantitative evaluation. In: Gruen, Baltsavias and Henricsson (Editors), Automatic Extraction of Man-Made Objects from Aerial and Space Images (II). Birkhauser, Ascona, pp. 65-76.

Hofmann, A., 2004. Analysis of TIN-Structure Parameter Spaces in Airborne Laser Scanner Data for 3-D Building Model Generation, XXth ISPRS Congress: Geo-Imagery Bridging Continents. IAPRS, Istanbul, Turkey.

Jutzi, B. and Stilla, U., 2003. Laser Pulses Analysis for Reconstruction and Classification of Urban Objects, Photogrammetric Image Analysis. IAPRS, Munich, Germany,, pp. pp. 151–156.

Koch, A., 2004. An Approach for the Semantically Correct Integration of a DTM and 2D GIS Vector Data, XXth ISPRS Congress: Geo-Imagery Bridging Continents. IAPRS, Istanbul, Turkey.

Maas, H.-G., 2001. The suitability of Airborne Laser Scanner Data for Automatic 3D Object Reconstruction, Third International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images, Ascona, Switzerland.

Maas, H.-G. and Vosselman, G., 1999. Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 54(no. 2-3): 153-163.

Overby, J., L. Bodum, E. Kjems and Ilsøe, P., 2004. Automatic 3D Building Reconstruction from Airborne Laser Scanning and Cadastral Data Using Hough Transform, XXth ISPRS Congress: Geo-Imagery Bridging Continents. IAPRS, Istanbul, Turkey.

Pfeifer, N., B. Gorte and Elberink, S.O., 2004. Influences of Vegetation on Laser Altimetry Analysis and Correction Approaches, International Conference NATSCAN "Laser-Scanners for Fores and Landscpae Assessment – Instruments, Processing Methods and Applications, ISPRS working group VIII/2, Freiburg im Breisgau, Germany.

Rottensteiner, F. and Briese, C., 2002. A New Method for Building Extraction in Urban Areas from High-Resolution Lidar Data, Symposium 2002 Photogrammetric Computer Vision. IAPRS, Graz, Austria, pp. 295-301.

Rottensteiner, F. and Briese, C., 2003. Automatic Generation of Building Models from Lidar Data and the Integration of Aerial Images, ISPRS working group III/3 workshop `3-D reconstruction from airborne laserscanner and InSAR data'. IAPRS, Dresden, Germany.

Schwalbe, E., H.-G. Maas and Seidel, F., 2005. 3D Building Model Generation from Airborne Laser Scanner Data Using 2D GIS Data and Orthogonal Point Cloud Projections, Workshop "Laser scanning 2005". IAPRS, Enschede, The Netherlands.

Sithole, G. and Vosselman, G., 2004. Experimental Comparison of Filter Algorithms for Bare Earth Extraction From Airborne Laser Scanning Point Clouds. ISPRS Journal of Photogrammetry and Remote Sensing, 59((1-2)): 85-101.

Sithole, G. and Vosselman, G., 2005. Filtering of airborne laser scanner data based on segmented point clouds, Workshop Laserscanning 2005. IAPRS, Enschede, the Netherlands.

Tóvári, D. and Pfeifer, N., 2005. Segmentation based robust interpolation - a new approach to laser data filtering, Laserscanning 2005. IAPRS, Enschede, the Netherlands.

Vosselman, G., 1999. Building Reconstruction Using Planar Faces in Very High Density Height Data, International Archives of Photogrammetry and Remote Sensing, Munich, Germany, pp. 87-92.

Vosselman, G., 2003. 3D Reconstruction of Roads and Trees for City Modelling, ISPRS working group III/3 workshop `3-D reconstruction from airborne laserscanner and InSAR data'. IAPRS, Dresden, Germany, pp. 231-236.

Vosselman, G., B. Gorte, G. Sithole and Rabbani, T., 2004. Recognising Structure in Laser Scanner Point Clouds, International Conference NATSCAN "Laser-Scanners for Fores and Landscpae Assessment – Instruments, Processing Methods and Applications, ISPRS working group VIII/2, Freiburg im Breisgau, Germany.

Vosselman, G. and Dijkman, S., 2001. 3D Building Model Reconstruction from Point Clouds and Ground Plans, ISPRS Workshop Land Surface Mapping and Characterization Using Laser Altimetry. IAPRS, Annapolis, USA.

A CONCEPT FOR ADAPTIVE MONO-PLOTTING USING IMAGES AND LASERSCANNER DATA

C. Ressl[a], A. Haring[a], Ch. Briese[a], F. Rottensteiner[b]

[a] Christian Doppler Laboratory “Spatial Data from Laser Scanning and Remote Sensing”, TU Vienna, {car,ah,cb}@ipf.tuwien.ac.at
[b] Institute of Photogrammetry and Remote Sensing, TU Vienna, fr@ipf.tuwien.ac.at

KEY WORDS: Photogrammetry, LIDAR, Surface, Reconstruction, Semi-automation, Extraction, Fusion, Combination

ABSTRACT:

The combination of photogrammetry (with its high geometric and radiometric resolution) and terrestrial laser scanning (allowing direct 3D measurement) is a very promising technique for object reconstruction, and has been applied for some time now, e.g. in the system Riegl LMS Z-420i. Nevertheless, the results presented from the combined laser-image-data very often are only coloured point clouds or textured meshes. Both object representations usually have erroneous representations of edges and corners (due to the characteristics of the laser measurement) and furthermore the amount of data to be handled in these “models” is typically enormous. In contrast to these object representations a surface model using a polyhedral compound would use only the relevant object points. However, the extraction of these *modelling points* from laser-image-data has not yet been fully automated. Especially the necessary generalization can only be accomplished by a human operator. Therefore, our aim is to support the operator in his work by speeding up the measurement of these modelling points. For this aim, this article presents a simple mono-plotting method that allows the human operator to identify each modelling point (on corners and edges) in the high-resolution images by a single mouse click. Subsequently, for this selected image ray, the missing distance is automatically determined from the associated laser data. This procedure starts by extracting the laser points in a cone around the image ray. Then these extracted points are tested for locally smooth surface patches (e.g. planar regions). Finally, the image ray is intersected with the foremost or hindmost of the extracted plane surface patches. Within this procedure the influence of erroneous laser measurements close to edges and corners can be avoided and furthermore, the distance from the scanner centre to the intersection point is determined with a better accuracy than the single laser point.

1. MOTIVATION

3D objects need to be represented for many applications, e.g. for visualization purposes, or for object analyses in order to derive certain object properties. The representation of a 3D object can be any of the following types (Rottensteiner 2001):

- *point cloud:* the object is just described by the vertices
- *wire frame model:* the object is described by vertices and edges
- *surface model:* the object is described by vertices, edges and faces
- *volumetric model:* the object is described by vertices, edges, faces and volumes, e.g. a set of volumetric primitives.

The representation using a point cloud may only serve for visualization purposes, with the visualization quality depending on the point density. However, mathematical analyses such as computing the volume or the area of an object are very difficult to accomplish when using only a point cloud representation. Such analyses require a model representation.

Of the three model representations, the surface model is the most applicable both for visualization and for mathematical analyses. Compared with wire frame models, surface models add the important definitions of faces, and compared with the volumetric models, surface models allow the representation of irregularly shaped objects in a much easier way.

Independent of the sensor (digital camera or terrestrial laser scanner) used for surveying of an object, the whole modelling process can be divided into three phases: data *acquisition*, data *orientation*, and the actual *modelling*.

In the context of this work we consider terrestrial objects, whose surface can be very well approximated by a *polyhedral compound*, e.g. facades of buildings, like the oriel window shown in fig. 1. Because of the large quantity of different object types that may appear in terrestrial scenes and the associated level of detail, it is very difficult to model such objects automatically – at least in a practical way. Usually the available data, e.g. images, provide a much higher resolution than required for the reconstruction of the object. The necessary generalization can only be accomplished by a human operator. Therefore, in the context of this work we consider the selection of the relevant object information for an adequate representation to be performed manually. A human operator digitizes the important object points, which make up the vertices of the surface model to be generated. These modelling points are placed at distinct positions of the object – usually on edges and at corners of the object.

Terrestrial photogrammetry always has been a prime source for deriving surface models of 3D objects. This is due to the capability of very fast data acquisition and high image resolution both in a geo-metrical and a radio-metrical sense. The orientation and modelling phases, however, are much more time consuming. This is mainly due to the fact that images only record directions to the object points, thus the 3D reconstruction has to be done indirectly by spatial intersection using at least two images taken from different view points. In addition, terrestrial images are usually not taken in a regular pattern, so that the orientation of such images can only be automated in a limited way (at least up to now no commercial software package provides a fully automated orientation procedure for general image arrangements), or special time consuming actions are required to take place on site (such as sticking markers on the object resp. providing many control points). Due to the indirect nature of photogrammetric object reconstruction, the modelling phase is slowed down as the human operator has to identify the modelling points in at least two images – very often only monoscopically. Overall, for the reasons mentioned above surface reconstruction from images is generally a rather time-consuming process.

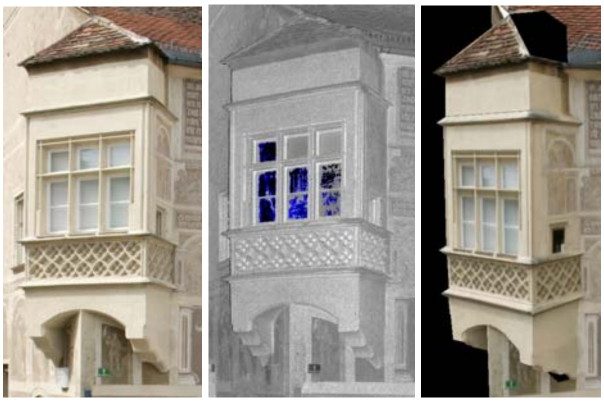


Fig. 1. a) Section of the original photo (pixel spacing 6mm), b) Section of the intensity image of the laser scanner (pixel spacing 1cm), c) Reconstructed object covered with the original texture from a). Data acquired with Riegl LMS Z420i and mounted Canon digital camera EOS 1Ds with a 20mm lens; mean object distance 14.5m.

With the advent of terrestrial laser scanning, it seemed that the procedure of deriving surface models would be sped up and terrestrial photogrammetry would become less important. This expectation was due to the following features of laser scanning, which outperform photogrammetry: (i) It is an active and direct 3D measuring principle, thus a 'single' measurement is sufficient to derive the 3D coordinates of an object point. (ii) The orientation of several overlapping laser scans can be automated to a very high degree. These tremendous advantages also compensate a slightly longer acquisition time on site (compared to taking images). However, it also became evident that laser scanning has its drawbacks: (i) Distances measured close to edges and corners are very unreliable; e.g. (Böhler et al. 2003). (ii) Compared with digital photogrammetry the laser's object resolution is generally a little worse from a geometric viewpoint and dramatically worse from a radiometric viewpoint; cf. fig. 1b and 1a. Due to these drawbacks, modelling from terrestrial laser scanner data is not yet completely satisfactory: The human operator has problems in identifying the "important" points only from the laser intensity data (due to the bad geometric and radiometric resolution), and furthermore these important points are on edges and corners of the object – spots where the laser might return erroneous distances. Consequently, point clouds rather than surface models are presented usually as the result of terrestrial laser scanning.

It also became clear that a combination of both photogrammetry (with its high geometric and radiometric resolution) and terrestrial laser scanning (allowing highly automated direct 3D measurement) would be promising. Additionally by mounting a digital camera directly on top of the laserscanner, e.g. the system Riegl LMS Z-420i (Riegl 2006) shown in fig. 2a, the orientation of the whole system can be determined very fast. Nevertheless, the results presented from the combined laser-image-data very often are still only coloured point clouds or textured meshes – both with erroneous representations of edges and corners. Although meshes are specific surface models, they are not the best choice for representing objects with polyhedral compounds from a storage point of view. Further, they usually have no object interpretation and generalisation.

In order to speed up manual modelling of objects by polyhedral compounds based on oriented image-laser-data, in this article we present a simple method that allows the human operator to identify each modelling point (at corners or edges) in the high-resolution images by a single mouse click. With this selected image ray, the missing distance is determined from the associated laser-data automatically. This procedure starts by extracting the laser points in a certain cone around the image ray. The extracted points are tested for the occurrence of planes. Then, the intersection points between the image ray and the detected planes are calculated yielding candidates for the required object point. Candidates being too far away from the laser points defining its object plane are eliminated. Finally, one of the remaining candidate points is chosen as result according to an intersection option selected by the user (i.e. the foremost or the hindmost point). In this way, the erroneous laser measurements close to edges and corners are avoided and furthermore, the distance from the image centre to the intersection point is determined with a better accuracy than the single laser point. This technique works best for images from mounted cameras, such as for the Riegl LMS Z420i, but can also be applied to other images.

The paper is structured in the following way: Section 2 gives an overview on related work. The detailed explanation of the proposed method is given in section 3, followed by two examples in section 4. An outlook in section 5 concludes the paper.

2. RELATED WORK

Our approach is closely related to mono-plotting in aerial photogrammetry, e.g. (Kraus 1996): From a single image the 3D coordinates of object points are determined by intersecting the respective projection rays with a given surface model; i.e. the points are 'plotted' on the surface. In recent years related work on applying mono-plotting to combined laser-image-data was published in different papers.

Perhaps one of the first approaches was the so-called '3D-orthophoto' (Forkert and Gaisecker 2002), later renamed to 'Z-coded true orthophoto' (ZOP) (Jansa et al. 2004). Here the image-laser-data is used to derive a true orthophoto with respect to a predefined object plane and with a certain ground resolution. The transition from this usual orthophoto to the ZOP is established by also computing the depth values of the orthophoto pixels with respect to the predefined object plane and adding this information as fourth layer to the three colour layers (red, green, blue).

Other authors apply the mono-plotting to the original images by mapping all laser points into the image and interpolating the object distance for all image pixels from these mapped points. Again, this distance information is stored as a fourth channel with the image. In (Bornaz and Dequal 2004) this resulting 4-dimensional image is termed 'solid image', and in (Abdelhafiz et al. 2005) this result is termed '3D image'.

The idea behind these 4-dimensional images and the ZOP is that the human operator just views the respective image, clicks on the points of interest and immediately gets the corresponding 3D coordinates. However, since here the original laser points are used for interpolating the distance of each image pixel either by nearest neighbour or by a simple average weighted method, the mentioned erroneous laser measurements close to edges and corners will to some extent remain in the results and may lead to unwanted smoothing effects.

The approach presented by (Becker et al. 2004) is closer to our method. Here also the original images and the associated laser data are used and only selected image points are determined in 3D space by intersecting image rays with 3D planes. The difference to our approach is that in (Becker et al. 2004) the operator first has to manually define the respective 3D plane in a view of the original image superimposed with the respective laser point cloud by selecting a certain area of supporting laser points. Afterwards the adjusting plane through that point set is determined, and from then on all further selected points in the original image will be mono-plotted with respect to this pre-defined plane. In our approach the respective object plane is determined automatically for each selected point, thus a higher degree of automation and a better adaptation to the shape of the object is achieved.

3. THE PROPOSED METHOD

Of the three phases mentioned in section 1 we only deal with the modeling phase in the context of this paper. Thus we assume the acquisition and orientation phase to be accomplished in advance. Therefore we know the camera's interior orientation, its relative orientation with respect to the scanner, and further the scanner's absolute orientation.

Consequently our problem is the following: Given the measured image co-ordinates of a point, we want to determine its 3D object coordinates using a raw, i.e. in no way pre-processed, laser scanner point cloud that covers the area of interest. With the known orientation, the image ray can be transformed to the co-ordinate system of the scanner.

Since the direction to the object point is already very precisely determined by the image ray, only the distance information is missing. The simplest approach would be to use the measured distance d_{meas} of the laser point P_{close} that is situated closest to the image ray and to intersect the image ray with the sphere with radius d_{meas} centred in the scanner's origin. This approach, however, has two drawbacks:

(i) It is not robust and thus not reliable: If P_{close} is near a depth discontinuity (i.e. close to an edge) the measured laser distance can be *systematically* wrong. A laser scan of an object of interest generally contains also points on non-interesting objects e.g. points on vegetation, on humans or cars passing by, etc. A scan may also contain blunders caused by failures of the measurement device. Consequently, if P_{close} is accidentally on one of these mentioned objects or a blunder, the selected distance will be *grossly* wrong.

(ii) It neglects possible accuracy improvements. Even if P_{close} is a valid laser measurement, its distance is still affected by random errors.

Both drawbacks can be eliminated if not only one point is considered but also its neighborhood. Consequently the task is to "intersect" the image ray with the point cloud. For this the following facts have to be considered:

- The laser point cloud is discrete. Therefore the covered region in the neighbourhood of the object point has to be approximated by a proper surface in order to compute the intersection with the image ray. The simplest approximation is by a plane, although in principle surfaces of higher order are also applicable.
- The laser measurements contain *random*, *systematic* and *gross errors*. In order to deal with the random errors the surface approximation has to be done using an adjustment and in order to deal with the systematic errors close to edges and gross errors in general this has to be done in a robust way.
- If the point of interest is situated on an edge or in a corner the respective image ray will in general intersect more than one object plane. Consider e.g. the planes of the oriel in fig. 1, where the image ray of a point on an oriel's edge will also intersect the plane of the façade of the house. The searched object point may be situated at the oriel's edge as well as in the façade's plane, since both possible 3D points are mapped to one and the same image point. In order to get a unique solution, the user has to specify which part of the object (the foremost or the hindmost) he or she is interested in.

The proposed method consists of two steps. In the first step, we extract the laser points from the point cloud that are situated within a certain neighbourhood of the image ray. More exactly, we define a "cone of interest". The axis of that cone coincides with the image ray, and its apex is the camera's projection centre. Its apex angle is chosen depending on the scan's angular step width. Only points inside this cone are considered for further analysis.

In the second step, the 3D co-ordinates of the point of interest are determined by intersecting the respective image ray with an object plane. For this at first, we set up plane hypotheses using an extension of the RANSAC (random sample consensus) framework (Fischler and Bolles 1981). Then, the intersection points between the image ray and the detected planes are calculated yielding candidates for the required object point. Candidates being too far away from the laser points defining its object plane are eliminated. Finally, one of the remaining candidate points is chosen as result according to an intersection option selected by the user (i.e. the foremost or the hindmost point).

3.1 Determination of the points inside the cone of interest

Solving this task is simplified by using the laser points' topological information, which is provided by most laser scanner systems. In case of the system Riegl LMS-Z420i, the measurements are arranged in a *measurement matrix*, where the column-row-index-space (c, r) and the direction-space (α, ζ) are related in the following way:

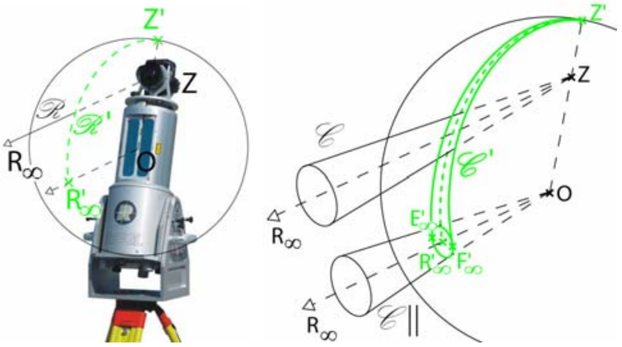
$$\alpha = \Delta_\alpha \cdot c + \alpha_0 \quad \text{and} \quad \zeta = \Delta_\zeta \cdot r + \zeta_0 \qquad (1)$$

where α is the horizontal direction, ζ the vertical direction, Δ_α the horizontal angle step width and Δ_ζ the vertical angle step width (usually $\Delta_\alpha = \Delta_\zeta = \Delta$), c the column index, r the row index, α_0 the horizontal direction at $c = 0$, and ζ_0 the vertical direction at $r = 0$.

Note that the directions α and ζ are only scheduled values. The actual direction measurement values (α_{meas}, ζ_{meas}) may slightly differ from the scheduled ones. These measurements (α_{meas}, ζ_{meas}) together with the measured distance and the intensity of the returned laser-pulse are stored in this matrix. Thus the measurement matrix actually contains 4 layers. Using only the intensity layer as grey value image, also called "intensity image" (cf. fig. 1b), the laser scanner data can be visualized in a simple way. Note, however, that only those cells in the data matrix are valid for which a distance measurement has been carried out successfully (cf. fig. 1b, where non-valid points appear blue).

In order to have enough points inside the cone of interest (which we denote by $\mathscr{C}$) we select a rather large apex angle of 20 times the angle step width Δ. For determining the points inside $\mathscr{C}$ we map $\mathscr{C}$ into the measurement matrix. Therefore, we have to project $\mathscr{C}$ first onto the unit sphere centred in the scanner's origin O, and afterwards transform it from the direction space to the column-row-index space using equations (1).

Apart from the cases where the image ray (which we denote by $\mathscr{R}$) contains the scanner's centre, its projection onto unit sphere is a part of a great circle (fig. 2a). Hence, this great circle arc is also the projection of the cone's axis. In order to get the whole spherical area of interest, we need the projection of the cone's contour (as seen from the scanner's centre O). The projections of the cone's contour generators are also great circle arcs. The set $\mathscr{G}$ of all generators' points at infinity corresponds to the intersection curve of the unit sphere with the parallel congruent cone $\mathscr{C}\|$ having its apex in the scanner's centre O. Hence, the image curve of $\mathscr{G}$ on the unit sphere is a small circle (fig. 2b).



(a) (b)

Fig. 2. (a): The laser scanner Riegl LMS-Z420i with mounted camera. An image ray $\mathscr{R}$ starting at the camera's projection centre Z is mapped onto unit sphere centered in the origin O of the scanner's co-ordinate system. The resulting image of the ray is a great circle arc between the image Z' of the projection centre and the vanishing point R_∞' of the ray's direction. (b): Cone of interest $\mathscr{C}$ around the ray $\mathscr{R}$ and its image on the unit sphere. E_∞' and F_∞' are the vanishing points of the cone's contour generators seen from the scanner's origin O.

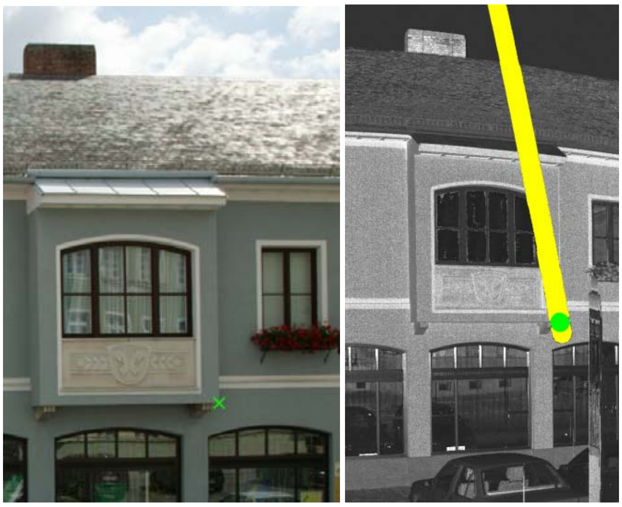
Using the equations of the two great circle arcs through E_∞' and F_∞' in fig. 2b and the small circle, and by applying the transformation from direction space to column-row-index space, we can determine the window of interest in the measurement matrix. Afterwards, we check for each pixel within this window, if it has a valid distance and if the respective laser point is actually inside the cone of interest ("point-inside-cone test"). Fig. 3 shows an example for a measured image point and the projection of the respective cone of interest in the scan's intensity image.

As result of this first step, we obtain a set of points near the image ray (represented by the green pixels in fig. 3b), which is the basis for further analyses.

3.2 Detection of object planes and determination of the 3D co-ordinates of the point of interest

In order to determine the 3D co-ordinates of the point of interest, we have to estimate a laser distance for the digitised image point using the obtained set of points inside the cone of interest. It was already argued in the beginning of section 3 that a reliable determination of such a distance by intersection with the respective image ray requires a surface approximation in the vicinity of the point of interest. The simplest approximation is by a plane, although in principle surfaces of higher order are also applicable.

We assume that up to i_{max} (e.g. $i_{max} = 5$) planes are to be found in the vicinity of the point of interest (i.e. in the cone of interest). Our approach for detecting them is based on the RANSAC framework. At the beginning, all points are unclassified, i.e. none of them is assigned to any plane. Plane detection is done iteratively. In each step ($i = 1, \dots i_{max}$), that plane π_i is detected which has the highest support from the unclassified points. The supporting points are then classified as belonging to the detected plane.



(a) (b)

Fig. 3. (a): Section of a photo with measured image point (green cross). (b): Section of the scan's intensity image. The pixels inside the spherical image of the respective cone of interest are marked in yellow; those also fulfilling the point-inside-cone test are marked in green.

Although in each step only one plane (the one with largest support) is detected, for finding this plane several plane hypotheses γ_k ($k = 1, \dots k_{max}$) are tested based on an adapted RANSAC approach. Finding planes using RANSAC in its original form would mean that we would have to randomly select the minimum number of three points and test the remaining points for incidence with this plane. However, we only select one seed point for each plane hypothesis. Then, we select all neighbouring points within a sphere of radius ε_1, which is chosen dependent both on the angular step width Δ of the laser scan and the laser distance measured at the seed point d_S as $\varepsilon_1 = 3d_S\Delta$. Thus, 20-30 points will be selected. A plane hypothesis γ_k is generated by calculating a least-squares plane through the seed point and the selected neighbours. This plane can also be considered as a tangential approximation to the laser points in the seed point. Summarizing, our modified RANSAC method differs from the classical one by the following:

- Each of the *n* points inside the cone of interest could be used to create a plane hypothesis. Thus, the maximum number of possible hypotheses is *n* compared with "*n* choose 3" in classical RANSAC. k_{max}, the number of hypotheses that have to be checked in order to find at least one seed point in one of the up to i_{max} possible planes with a probability of 99.9% is given by $\ln(0.001)/\ln(1-1/i_{max}) \sim 31$.
- The plane hypotheses are set up locally by least-squares adjustment compared to a direct solution through 3 points at classical RANSAC.
- This modified approach tends to suppress the generation of hypotheses merging two or more slightly different planes to a single one and therefore obtaining unjustifiably high support. In other words, our modified method provides more robustness against unjustified merging of planes.

A point supports a plane hypothesis if its orthogonal distance to the plane is within $\pm\varepsilon_2$, which is the standard deviation (in direction of the plane-normal) derived from the seed point's co-variance matrix. The latter is derived from the accuracies of the laser scanner. Of all the plane hypotheses γ_k the plane π_i having the highest support, i.e. the plane explaining most unclassified data points, is accepted. Afterwards, π_i is re-adjusted using all its supporting points.

The points in this supporting set, however, are not yet finally assigned to π_i. The assignment of an unclassified data point to the plane π_i is based on a statistical test (significance level e.g. 5%). This test considers the points' orthogonal distance to π_i and the covariance matrices of π_i and of the points (derived from the laser scanner accuracies). This is favoured over a simple (non-statistical) distance threshold criterion, because the accuracy of a laser point in Cartesian space may be rather anisotropic. Especially in case of short distances (< 10m), a point is significantly better determined perpendicularly to the laser beam than in radial direction. In other words, the noise perpendicular to the plane heavily depends on the angle between plane normal and laser beam.

For the remaining unclassified points the next plane with maximum support π_{i+1} is detected, and so on. The process will stop after the maximum number of "best" planes π_i ($i_{max} = 5$) has been detected or if only a small percentage (e.g. 10%) of points is still unclassified. As result, we get a set of planes together with their associated data points.

Each plane π_i is intersected with the image ray yielding candidates S_i for the desired intersection point. However, we can immediately reject those candidate-points that are situated far away from any data point assigned to the respective plane π_i. Therefore, for each candidate point S_i, we determine the closest data point P_i belonging to its underlying plane, and calculate the distance between S_i and P_i. A candidate point is rejected if this distance exceeds a distance ε_3, which depends on the distance d_S of S_i to the scanner's origin and the scan's angular step width Δ as $\varepsilon_3 = 2d_S\Delta$.

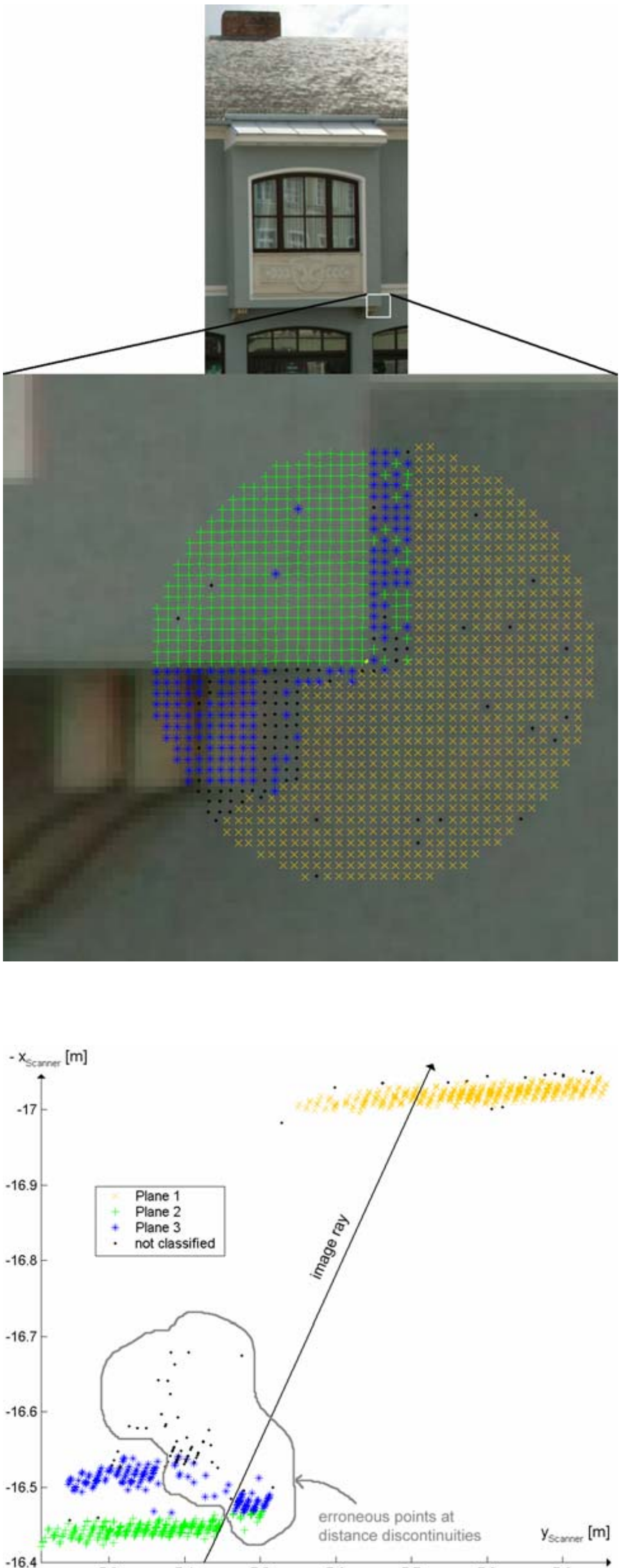
Finally, one of the remaining candidate-points is accepted according to the selected user option, which may be intersection with either the foremost or the hindmost plane. In this way we obtain the 3D co-ordinates of the object point measured in the photo in the beginning. If there are additional observations to the same point (e.g. in other photos), its calculated laser distance together with its image co-ordinates may be introduced as observations in a subsequent adjustment.

4. EXAMPLES

In this section, we give two typical examples in order to demonstrate our approach. In case of the first example, three planes were detected (fig. 4). Depending on the user option, either plane 2 or plane 1 is intersected with the ray. In this case plane 2 is used (i.e. the foremost plane) since the user is interested in the oriel's corner. Note that our approach delivers a reasonable result, although the image ray runs through an area of erroneous laser points near distance discontinuities.

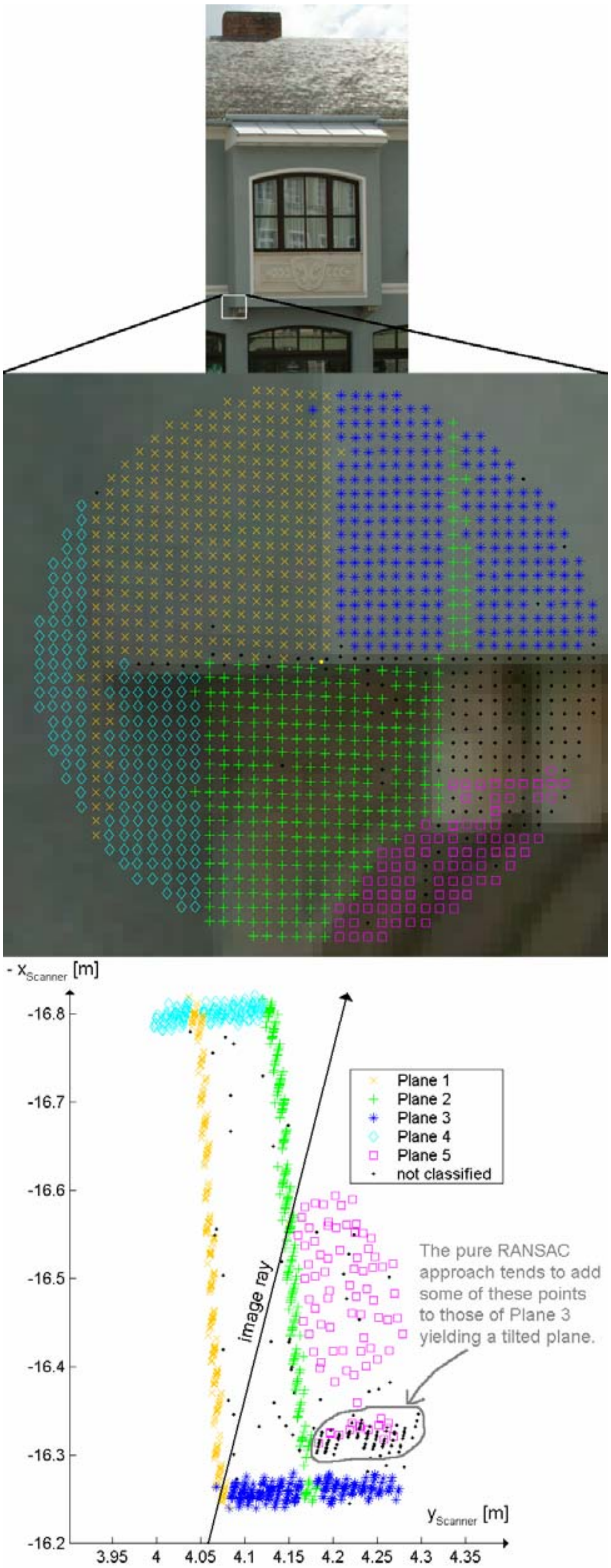
Fig. 5 shows another example, where the maximum number of planes ($N_{max} = 5$) was detected. Compared with the previous example, the proportion of erroneous points is relatively small. However, this is a good example in order to argue why we do not use the original RANSAC approach (generation of hypothesis by 3 random points) for plane detection: Due to the poor extension/noise ratio of plane 3, the original RANSAC approach tends to merge the points of plane 3 with some of those situated on the (parallel) front plane of the oriel's corbel, which is about 5cm behind. Hence, the hypothesis having the highest support would deliver a tilted plane – and therefore a wrong intersection point. However, our adapted RANSAC approach is able to separate those two different planes, since it is more robust against merging of noisy similar planes. The desired point (the oriel's corner) is obtained by intersecting the image ray either with plane 1 or plane 3. Note that the distance between the respective intersection points is only 5mm (compared with the distance measurement accuracy of ±1cm). Thus the error of choosing a wrong neighbouring plane is smaller than the original measurement accuracy. However, as this example shows a further improvement of the proposed method could be achieved by determining the object point of interest not only by intersecting the image ray with one plane but to include (if present in the laser data) up to two intersecting planes in case the point lies on an edge or up to three planes in case the point is a corner. Fig. 5 also shows an unwanted property of the current implementation of the RANSAC approach, that due to the "first come first serve" classification points, which would better fit to plane 4 are classified to the more dominant plane 1. Same holds for the planes 3 and 2. We will adapt this by region-based analyses in the next implementation.

Anyway, the two examples show that our approach is able to deal with blunders, systematic errors and measurement noise.



erroneous points at distance discontinuities

Fig. 4. Laser points near image ray after classification (cf. fig. 3). Top: Projection into photo. Bottom: Ground view.



The pure RANSAC approach tends to add some of these points to those of Plane 3 yielding a tilted plane.

Fig. 5. Another example after classification of the laser points. Top: Projection into photo. Bottom: Ground view.

5. SUMMARY AND OUTLOOK

We presented a concept for mono-plotting using combined image and laser data. The idea is that a human operator first selects the relevant modelling points manually in the high resolution images, thereby performing the necessary generalization. Then the 3D coordinates of the respective object points are obtained by intersecting the image ray with a surface patch. The latter is extracted by analyzing the laser points in a cone of interest around the image ray. We search for planes with the highest support from the laser points, which is motivated by considering primarily objects that can be represented by polyhedral compounds. Surface patches of higher order, however, could also be applied for other applications.

The main properties of this approach are: (i) it is *adaptive*, in the sense that for each selected image point a well suited plane from the laser data is searched for, (ii) gross and systematic *errors in the laser data are removed* and due to the adjusting surface patch the distance to the object points is derived with a better accuracy than the single laser point measurement.

Future work should be directed in two ways:
(i) Increase the automation of surface modelling using combined image and laser data. From a geometric point of view the redundancy in the image and laser data, especially concerning edges, which can be extracted automatically to a high degree in both data sets, is promising. From a semantic point of view this task, however, is rather challenging, as the rate and method of generalization is difficult to automate and will involve many aspects from artificial intelligence. Therefore this task will remain relevant within the respective communities (photogrammetry, computer vision, cartography …) for the coming years, perhaps even decades.

(ii) In the meantime the proposed mono-plotting method is a promising tool to speed up object modelling. Therefore it is worth investigating the amount of time that can be saved using our method, e.g. by comparing the time required to model a certain large object by this mono-plotting method and by other methods. Also the accuracy achieved by the proposed method needs to be analyzed, although the term *accuracy* in the context of surface modelling also involves aspects of generalization. Further the method for deriving the (planar) patch of highest support from the laser data may have room for improvement (cf. sec. 4). An alternative to the already implemented RANSAC approach would be a Hough-transform-like approach (e.g. (Pottmann et al. 2002)), where for each laser point in the cone of interest its tangential plane is estimated using the neighbouring points. Afterwards in the parameter-room of these planes the clusters of planes are analyzed. We will work on theses issues in the future.

6. ACKNOWLEDGEMENT

Part of this project was supported by the innovative project „The Introduction of ILScan technology into University Research and Education“ of the University of Technology, Vienna.

7. REFERENCES

Abdelhafiz A., Riedel B., Niemeier W., 2005. “3D Image” As A Result From The Combination Between The Laser Scanner Point Cloud And The Digital Photogrammetry, *Optical 3-D Measurement Techniques VII*, (Eds. A. Grün and H. Kahmen), Volume 1: 204-213

Becker R., Benning W., Effkemann, C., 2004. Kombinierte Auswertung von Laserscannerdaten und photogrammetrischen Aufnahmen, *Zeitschrift für Vermessungswesen*, Heft 5: 347-355

Böhler W., Bordas Vicent M., Marbs A., 2003. Investigating Laser Scanner Accuracy, *The International Archives of* Photogrammetry*, Remote Sensing and Spatial Information Sciences*, Antalya , Vol. XXXIV, Part 5/C15: 696-701

Bornaz L., Dequal S., 2003. The Solid Image: An Easy And Complete Way To Describe 3D Objects, *The International Archives of* Photogrammetry*, Remote Sensing and Spatial Information Sciences*, Antalya , Vol. XXXIV, Part 5/C15.

Fischler M. A., Bolles R. C. Random Sample Consensus, 1981: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Comm. of the ACM*, Vol 24, pp 381-395.

Forkert G., Gaisecker Th., 2002. *3D Rekonstruktion von Kulturgütern mit Laserscanning und Photogrammetrie*, Oral presentation at the CULTH2 congress “Die Zukunft des Digitalen Kulturellen Erbes” at the MUMOK, Vienna, Jan. 13 – 14 2002.

Jansa J., Studnicka N., Forkert G., Haring A., Kager H., 2004. Terrestrial Laserscanning and Photogrammetry - Acquisition Techniques Complementing One Another; in: O. Altan (Ed.); *Proceedings of the ISPRS* XXth Congress, Vol XXXV, Part B/7, Istanbul, July 12 – 23 2004; ISSN 1682-1750; 948 - 953.

Kraus K, 1996. *Photogrammetry*, Volume 2: Advanced Methods and Applications, Third edition, Dümmler/Bonn.

Pottmann H., Leopoldseder S., Wallner J., Peternell M., 2002. Recognition and reconstruction of special surfaces from point clouds. *International Archives of Photogrammetry and Remote Sensing*, Graz, Austria, Vol. XXXIV, Part 3A, pp. 271-276.

Riegl, 2006. http://www.riegl.co.at/ [Accessed: 2006-03-24]

Rottensteiner F., 2001. Semi-automatic extraction of buildings based on hybrid adjustment using 3D surface models and management of building data in a TIS, PhD Thesis, Institute of Photogrammetry and Remote Sensing, Vienna, 189 pages

ROBUST AUTOMATIC MARKER-FREE REGISTRATION OF TERRESTRIAL SCAN DATA

Wolfgang von Hansen

FGAN-FOM, Gutleuthausstr. 1, 76275 Ettlingen, Germany
wvhansen@fom.fgan.de

KEY WORDS: Urban, Terrestrial, Laser scanning, Point Cloud, Segmentation, Automation, Registration, Algorithms

ABSTRACT:

Terrestrial laser scanning systems have become widely available during the past years. Raw data acquired by such systems typically consists of separate overlapping datasets – each in its own local coordinate system. Applications that need data from more than a single scan position therefore must be preceded by a registration of all scans into a common geometric reference frame.
In this paper, a novel method for the automatic and marker-free coarse registration of terrestrial laser scan data is presented. It is based on matching planes in object space and is thus especially suitable for scenarios that are dominated by planar structures such as built-up areas. First, suitable planes are extracted from the raw point cloud in a robust way. Then, the automatic coarse registration is carried out based on correspondences of single plane pairs. Results are shown for test data constisting of 26 datasets of a small village.

1 INTRODUCTION

1.1 Motivation

In addition to airborne laser scanning, terrestrial LIDAR systems have become widely available. While products from airborne scanners cover larger areas and are often delivered as *one* georeferenced dataset, terrestrial systems are typically operated by end-users and capture separate overlapping datasets – each in its own local coordinate system. Any application that needs data from more than a single scan position therefore must be preceded by a registration of all scans into a common geometric reference frame.

The state of the art in registration of terrestrial scan data is to place artificial markers – either 2D or 3D targets – into the scene before data acquisition. Registration software for (semi-) automatic matching of the targets is commercially available. In contrast to this, an automatic coarse registration of terrestrial scan data in the absence of markers still is a topic of research (Dold, 2005).

In this paper, a novel method for the automatic and marker-free coarse registration of laser scan data is presented. It is based on matching planes in object space and is thus especially suitable for scenarios that are dominated by planar structures such as built-up areas. A robust generation of planes from the 3D point cloud is used as preprocessing. The registration algorithm comprises a complete search that generates all possible solutions for single plane matches and then chooses the best ones based on inlier counts. The implementation turned out very fast for our test data which is a set of 26 datasets of a small village. Results show that a reliable coarse registration is possible even for such complex scenarios and thereby proves the applicability of our algorithm to real world tasks.

1.2 Related work

The basic algorithm often cited for registration of point clouds is the ICP *(iterative closest point)* algorithm (Besl and McKay, 1992): Given an initial transformation, feature correspondences are found and new transformation parameters are estimated through a least squares adjustment. This procedure is iterated until convergence. Extensions exist to enhance the radius of convergence but ICP is mainly suitable for fine registration. One example of an ICP derived method is presented in (Bae and Lichti, 2004). Matching is based on geometric curvature and change of normal vector within a given neighborhood.

(Dold and Brenner, 2004) describe the principle of registration based on three plane matches. A region growing method for the estimation of planes from point clouds with known scan geometry is presented. Subsequently, the unknown rotation is recovered through *extended gaussian images* (Dold, 2005): The normal vectors are all projeted onto a unit sphere and then clustered through its tesselation. Matching the spheres at multiple resolution levels yields the rotation matrix but not the translation vector.

The adaption of the *normal distribution transform* (NDT) from 2D laser scanners used in robotics applications to the registration of 3D point clouds is proposed by (Ripperda and Brenner, 2005). Basically, the 3D data is sliced horizontally and then processed as 2D data. Although the method yields good results, one has to cope with convergence issues as well as some loss of information through the reduction of dimensionality.

The complete sequence of segmentation, coarse and fine registration is also shown by (Liu and Hirzinger, 2005), introducing the *matching tree* as a new search structure. The scene is segmented based on changes of the normal vectors and stored in a special graph structure which is then exploited for registration.

The methods reviewed here are all steps towards a generic solution, but each approach is improvable. Results for coarse registration presented are mainly – with the exception of (Ripperda and Brenner, 2005) – applied to simple scenarios only, where a single object dominates the scene and the overlap between the datasets is large. The applicability to large and complex scenarios had not been proven yet. Correspondence search algorithms similar to the one presented in this paper have been applied to 2D matching problems in computer vision for a long time (Ballard and Brown, 1982, Grimson, 1990).

2 METHODOLOGY

2.1 Overview

We will use the following terminology: The small and localized planes generated directly from the point clouds are called *surface elements*. Groups of coplanar and neighboring surface elements are *planes*. The term *matching* will be used to denote the establishment of a logical link between two planes of two different datasets while the transformation of one dataset into the geometric reference frame of the other is called *registration*. The algorithms frequently require some thresholds for decisions – these are always denoted by a Θ with the referenced entity as index.

This section describes the processing chain from the raw point clouds to the final transformation parameters. The registration of point clouds can be subdivided into two tasks. The first is a preprocessing and feature generation step that converts the raw point cloud into a representation suitable for the second task, which is the matching of features in order to estimate the yet unknown transformation parameters of the registration. The registration can again be subdivided into a coarse and a fine registration. This distinction is necessary as precise algorithms usually require good initialization values for the transformation parameters, while robust methods that can handle large displacements usually do not return a statistically optimal result. Here is a summary of the steps of the method proposed in this paper:

Generation of surface elements Each point cloud is split into 3D raster cells. For each cell, the dominant plane is estimated through a RANSAC scheme.

Grouping to planes Neighboring coplanar surface elements are grouped to planes. These typically coincide with planar object surfaces.

Coarse registration This step is the most difficult in the processing chain and its solution is the main contribution of this paper. An exhaustive search for matching planes of two datasets is carried out. For each possible match, initial transformation parameters are computed and the number of inliers is counted. Those matches with a high inlier count are returned as correct matches for the fine registration.

Fine registration As the coarse registration returns both a set of plane matches and initial transformation parameters, the fine registration is a least squares adjustment over all scan positions to compute optimal parameters. A statistical test allows detection and removal of outliers that may have remained in the data. The fine registration is outside the scope of this paper.

2.2 Generation of surface elements

The planes that the registration algorithm requires as input will be generated in a two step process. Surface elements will be generated in a robust way from the point cloud and then are grouped to planes (see Sec. 2.3). We utilize a method that is described in (von Hansen et al., 2006) and is shown here as Alg. 1.

The set of 3D points $\mathcal{X}$ is partitioned and assigned to 3D volume cells using a Cartesian raster. All points in one of the raster cells are denoted by $\mathcal{X}_i$. For each cell, the dominant plane $p_i = (\mathbf{n}_i, d_i)$ – i. e. the one that has the biggest support from the 3D points $\xi \in \mathcal{X}_i$ – is robustly estimated. This has been implemented using the well known RANSAC strategy (Fischler and Bolles, 1981), yielding a set of inlier points $\widehat{\mathcal{X}}_i$.

```
Input: 3D point cloud $\mathcal{X}$.
Output: 3D raster $\mathcal{S}$ with one surface element $s_i$ per cell.
  Divide $\mathcal{X}$ into regular raster cells $\mathcal{X}_i$.
  for all $\mathcal{X}_i$ do
    Robustly estimate dominant plane $p_i = (\mathbf{n}_i, d_i)$
        from all points $\xi \in \mathcal{X}_i$.        {E. g. via RANSAC.}
    Compute barycenter $\mathbf{x}_i$ from those $\hat{\xi}$ that support $p_i$.
    Add $s_i := (\mathbf{n}_i, \mathbf{x}_i)$ to output $\mathcal{S}$.
  end for
```

Algorithm 1: Segment point cloud into surface elements.

```
Input: Surface elements $\mathcal{S}$ as output from Alg. 1.
Output: A set of planes $\mathcal{P}$ grouped from $\mathcal{S}$.
  {Build graph structure.}
  Create empty graph $\mathcal{G}$.
  for all $s \in \mathcal{S}$ do
    Insert $s$ as vertex into $\mathcal{G}$.
    for all $t \in \mathcal{G}, t \neq s$ do
      if $s, t$ neighbors in 3D raster and $s, t$ coplanar then
        Insert (undirected) edge between $s$ and $t$ into $\mathcal{G}$.
      end if
    end for
  end for
  {Extract planes from graph.}
  {Connected components of $\mathcal{G}$ are groups of $\mathcal{S}$.}
  for all connected components $\mathcal{C} \subseteq \mathcal{G}$ do
    if $|\mathcal{C}| < \Theta_\mathcal{C}$ then
      {Omit small structures.}
    else
      Estimate $p := (\bar{\mathbf{n}}, \bar{\mathbf{x}})$ from all $s_i = (\mathbf{n}_i, \mathbf{x}_i) \in \mathcal{C}$.
      Add $p$ to $\mathcal{P}$.
    end if
  end for
```

Algorithm 2: Group surface elements to planes.

A localization of the plane in space is also needed in order to be able to recover the translation vector $\mathbf{t}$ with only one plane match. The plane represented by the Hesse normal form

$$ax + by + cz + d = \mathbf{n}^\top \mathbf{x} + d = 0 \qquad (1)$$

has an infinite extent. We are interested in a small and delimited plane representing the points $\widehat{\mathcal{X}}_i$ only. Therefore, in addition to the normal vector $\mathbf{n}_i$, the barycenter $\mathbf{x}_i$ – i. e. the mean – of the point cloud $\widehat{\mathcal{X}}_i$ is stored as well. The distance d_i of the plane to the origin need not be stored because it is determined by

$$d_i = -\mathbf{n}_i^\top \mathbf{x}_i. \qquad (2)$$

The surface elements $\mathcal{S}$ are used for visualization instead of the raw points. Their shape can be recognized easily in all figures showing 3D data.

2.3 Grouping to planes

The input is a regular 3D raster $\mathcal{S}$ with each cell containing *one* surface element $s_i = (\mathbf{n}_i, \mathbf{x}_i)$ that rather precisely represents a small planar region of an object surface. Obviously, many neighboring surface elements describe exactly the same plane. The grouping collects them into a single plane based on adjacency and coplanarity as described in Alg. 2.

The basic structure used for this is a graph $\mathcal{G}$. All surface elements $s \in \mathcal{S}$ are entered as nodes and then compared to all of their 26 neighboring cells of the 3D raster. If two such surface elements are coplanar, then an undirected edge is inserted between the respective graph nodes. Since the order of these operations does not matter, the resulting graph is determined uniquely.

The connected components of $\mathcal{G}$ are planes composed from the surface elements. They are simply extracted from the graph by computing a mean normal vector $\bar{\mathbf{n}}$ and a mean barycenter $\bar{\mathbf{x}}$ from each connected component and storing it as one plane in the output set $\mathcal{P}$. A threshold $\Theta_\mathcal{C}$ is applied to remove planes that do not have enough support by the surface elements. This is mainly done to reject planes induced by noise and to reduce the amount of data in favor of larger and better planes.

2.4 Coarse registration

The coarse registration of two datasets is a typical chicken and egg problem. In order to compute transformation parameters, matching entities must be identified first. On the other hand, matching usually requires some knowlegde about the transformation parameters. We will solve the dilemma through a complete search that generates all possible matches from which the correct ones will be extracted based on inlier counts.

It is required that the scenario contains planar object surfaces as these will be used for matching. The purely mathematical solution such as proposed in (Dold and Brenner, 2004) needs two plane matches for rotation and three for translation. Neither a random nor a systematic generation of matches seems feasible when only about 20–30$\%$ of the planes are in the overlapping area. However, the situation can be improved when ancillary knowledge is taken into account. For terrestrial laser scanners, the zenith direction is usually known from restrictions in the sensor setup. Hence, each dataset implicitly contains the horizontal ground plane so that only one plane match is required to solve for rotation.

This single plane match can already be exploited to get an approximate translation vector $\mathbf{t} = \mathbf{x}_j - \mathbf{x}_i$ via the known barycenters. This will not yield a precise solution – because there might be systematic shifts when different parts of a surface have been visible in the two datasets – but this error will cancel out when multiple matches are regarded.

The complete strategy for the coarse registration is presented in Alg. 3. First, a complete search over all possible single plane matches is carried out. As co-aligned zenith directions are assumed, this knowledge can be applied to narrow the search. The 3D normal vector $\mathbf{n}$ of each plane is expressed in a spherical coordinate system with inclination φ and azimuth α. Two planes can only match when they have the same inclination. Then, the transformation parameters rotation $\mathbf{R}$ and translation $\mathbf{t}$ from $\mathcal{P}_2$ to $\mathcal{P}_1$ are computed through the difference of azimuth and barycenter respectively. The planes of $\mathcal{P}_2$ are transformed into the geometric reference frame of $\mathcal{P}_1$ and denoted $\mathcal{P}_2''$.

The next step is the count of inliers n_{ij} – the number of planes matching for a particular set of parameters. If the transformation is correct, then two matching planes must have similar parameters. Each plane of the first dataset $\mathcal{P}_1$ is compared to all planes of the second dataset $\mathcal{P}_2''$ and the number of matches with similar inclination φ, azimuth α and barycenter $\mathbf{x}$ are counted. Finally, the triggering match is entered into a list along with its transformation parameters and inlier count (Tab. 1).

Each generated match thereby is supported by other matches that verify it. The list is sorted with respect to the inlier count n_{ij} and the m best ones are picked for computation of the parameters. There are several complementary possibilities to choose m:

1. The maximum number of inliers found $n_{\max}$ – i. e. the first row of the sorted list – is an upper bound for m.
2. The median of the first $n_{\max}$ inlier counts is a robust estimation for the inlier rate.
3. A large difference in n from one row to the next (e. g. $n_{i+1} - n_i > \Theta_n = 1$) indicates a possible end of the inlier list.

We have used the minimum of all three possibilities as m. The transformation parameters are then estimated as a (robust) mean from all inlier matches.

```
Input: Two sets of planes $\mathcal{P}_1, \mathcal{P}_2$ generated from two point
  clouds as output from Alg. 2.
Output: Transformation parameters $\mathbf{R}, \mathbf{t}$ from $\mathcal{P}_2$ to $\mathcal{P}_1$. List of
  plane matches $\widehat{\mathcal{C}} = (n_{ij}, p_i, p_j, \mathbf{R}_{ij}, \mathbf{t}_{ij}), p_i \in \mathcal{P}_1, p_j \in \mathcal{P}_2$.
  {Compute additional plane attributes.}
  for all $p = (\mathbf{n}, \mathbf{x}) \in \mathcal{P}_1 \cup \mathcal{P}_2$ do
    $\alpha \leftarrow \arctan(\mathbf{n}_y / \mathbf{n}_x)$ {Azimuth}
    $\varphi \leftarrow \arcsin \mathbf{n}_z$ {Inclination}
    Add attributes $\alpha, \varphi$ to $p$.
  end for
  {Iterate through all possible correspondences.}
  Create empty list of correspondences $\mathcal{C}$.
  for all $p_i \in \mathcal{P}_1$ do
    for all $p_j \in \mathcal{P}_2$ do
      if $|\varphi_j - \varphi_i| < \Theta_\varphi$ then {New correspondence.}
        {Transform $\mathcal{P}_2$ according to match $(p_i, p_j)$.}
        $\mathbf{R}_{ij} \leftarrow$ Rotation by angle $\alpha_j - \alpha_i$ around $z$-axis.
        $\mathcal{P}_2' \leftarrow$ Apply $\mathbf{R}_{ij}$ to $\mathcal{P}_2$.
        $\mathbf{t}_{ij} \leftarrow \mathbf{x}_j' - \mathbf{x}_i'$
        $\mathcal{P}_2'' \leftarrow$ Translate $\mathcal{P}_2'$ by $\mathbf{t}_{ij}$.
        {Count inliers.}
        $n_{ij} \leftarrow 0$
        for all $p_k \in \mathcal{P}_1$ do
          for all $p_\ell'' \in \mathcal{P}_2''$ do
            if $|\varphi_\ell'' - \varphi_k| < \Theta_\varphi \wedge |\alpha_\ell'' - \alpha_k| < \Theta_\alpha$
                $\wedge \|\mathbf{x}_\ell'' - \mathbf{x}_k\| < \Theta_\mathbf{x}$ then
              $n_{ij} \leftarrow n_{ij} + 1$
            end if
          end for
        end for
        Insert $(n_{ij}, p_i, p_j, \mathbf{R}_{ij}, \mathbf{t}_{ij})$ into $\mathcal{C}$.
      end if
    end for
  end for
  Sort $\mathcal{C}$ with respect to $n$.
  Pick $m$ correspondences $\widehat{\mathcal{C}}$ with most inliers from $\mathcal{C}$.
  Compute output $\mathbf{R}$ and $\mathbf{t}$ from all $c \in \widehat{\mathcal{C}}$.
```

Algorithm 3: Automatic coarse registration.

3 EXPERIMENTS AND RESULTS

3.1 Available datasets

We dispose of 26 overlapping datasets from a Z+F Imager 5003 terrestrial laser scanner. It has an operation range of 50 m and captured about 100 million valid 3D points per dataset. For each 3D point, the amount of reflected light is recorded and available for surface textures.

The imaged scenario is a farming village, containing moderately complex arrangements of small houses around many courtyards. Buildings are typically two stories high and have inclined roofs. The global layout of all scan positions is shown in Fig. 1.

3.2 Generation of surface elements

The raster size for the generation of surface elements has been chosen as 1 m in order to describe façade and roof surfaces through several surface elements but also to ignore small structures. Results for three selected positions are shown in Fig. 2. The grid like texture of the plane boundaries originally stems from unintended border effects in the visualization, but clearly shows how the result is composed. The number of surface elements generated for each of the positions ranges from 2600 to 10000.

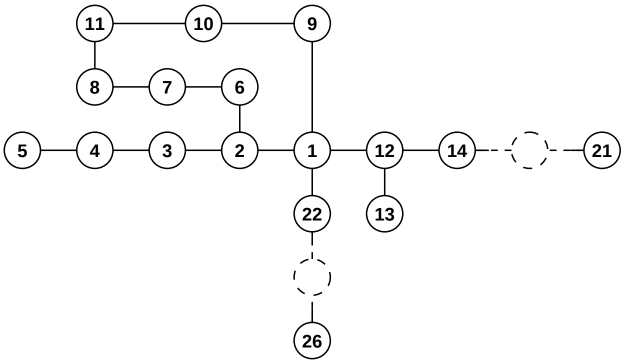


Figure 1: Graph-like layout of all scanner positions, roughly in their correct geometric place. Dashed lines denote rows of consecutive positions that have been left out in this illustration. Directly neighboring positions are connected, but distant datasets may also overlap.

			Rotation	Translation		
#Inliers	p_i	p_j	α/rad	$\mathbf{t}_x$/m	$\mathbf{t}_y$/m	$\mathbf{t}_z$/m
17	88	41	−0.2419	20.76	4.43	−1.71
16	95	55	−0.2527	20.56	2.85	−1.22
16	94	52	−0.2467	20.78	3.93	−0.54
16	91	45	−0.2406	21.00	2.99	−1.31
16	90	40	−0.2424	21.19	2.81	−1.68
16	89	43	−0.2435	20.38	2.85	−1.12
16	87	42	−0.2340	20.92	3.42	−1.27
16	86	33	−0.2282	21.14	2.85	−2.08
16	85	34	−0.2571	20.50	2.81	−1.30
16	67	26	−0.2512	21.89	3.17	−1.53
16	35	3	−0.2587	21.00	2.72	−1.97
16	31	5	−0.2411	21.01	3.18	−1.67
16	27	2	−0.2521	21.39	2.94	−1.25
15	79	29	−0.2408	21.96	2.57	−1.02
15	76	30	−0.2310	20.81	3.01	−0.75
13	93	52	−0.2481	20.80	3.52	−3.90
8	76	31	−0.2276	19.19	2.27	0.93
5	94	62	−0.2304	−3.70	−0.62	−0.80
5	49	55	1.2278	−3.44	−15.16	−2.69
5	28	59	1.2675	−3.14	−12.69	−3.01

Table 1: Twenty best matches from position 2 to 3. The horizontal line indicates the end of the automatically chosen inlier set.

3.3 Grouping to planes

The grouping is a deterministic procedure that can be guided through two thresholds – one for coplanarity and one to reject too small planes. One result is shown in Fig. 3, where all planes recovered from position 2 are shown in uniform colors. The number of planes for each position ranges from 30 to 100.

Since only local comparisons are used for the creation of the graph, it may happen that large resulting regions are not exactly planar. While the surface elements are an oversegmentation of object space, the planes are an undersegmentation for which the streets in an outdoor scenario would be typical examples. For the coarse registration this poses no real problem, since undersegmentation results in only a few planes that are easily ignored by robust algorithms.

3.4 Coarse registration

As the maximum deviation from the true zenith direction was $\leq$30 mrad, no prior rotation of the datasets was necessary. As a typical example for the output of Alg. 3, the list of matches from position 2 to 3 is shown in Tab. 1. Only the top twenty matches are given – the actual list is much longer (cf. column “#Tests” of Tab. 2). Column “#Inliers” contains the number of inliers for the match of the two planes listed in the columns p_i and p_j. The last four columns show the transformation parameters valid for this particular match. A horizontal line marks the automatically defined end of the inlier set.

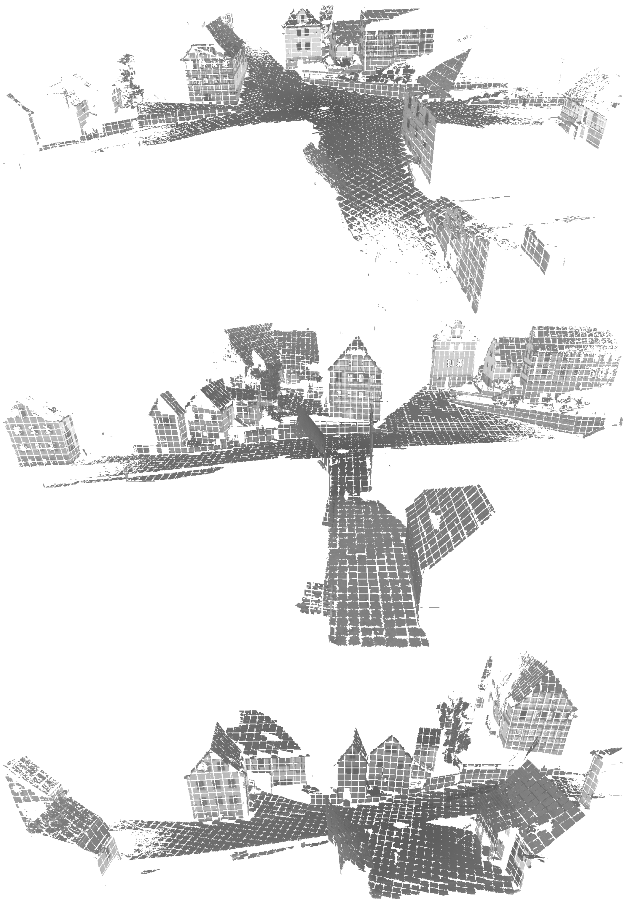


Figure 2: Datasets from positions 1 to 3 (from top to bottom). The small square structures are the surface elements.

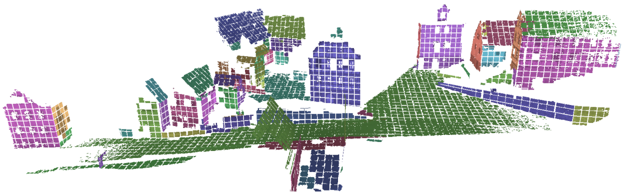


Figure 3: Result of the grouping for position 2.

Results for all neighboring positions of the test data (cf. Fig. 1) are shown in Tab. 2. Columns $\mathcal{P}_1$ and $\mathcal{P}_2$ are the position numbers, column $n_1 n_2$ is the total number of tests that are possible, while the next two columns show the number and percentage of tests actually carried out because the inclination indicated a possible match. At least half of the generated correspondences could be rejected early through this criteria.

The following columns are the results of the inlier tests. The absolute number of inliers is given along with the inlier rate with respect to $\min(n_1, n_2)$. The inlier rate of only 25% is low mainly because of the limited overlap between the datasets. Not all registrations have been successful. Filled circles mark successful registrations while empty circles indicate failures.

In order to illustrate the results of the coarse registration, a color-coded fusion of the individual datasets from positions 1 to 3 is shown in Fig. 4.

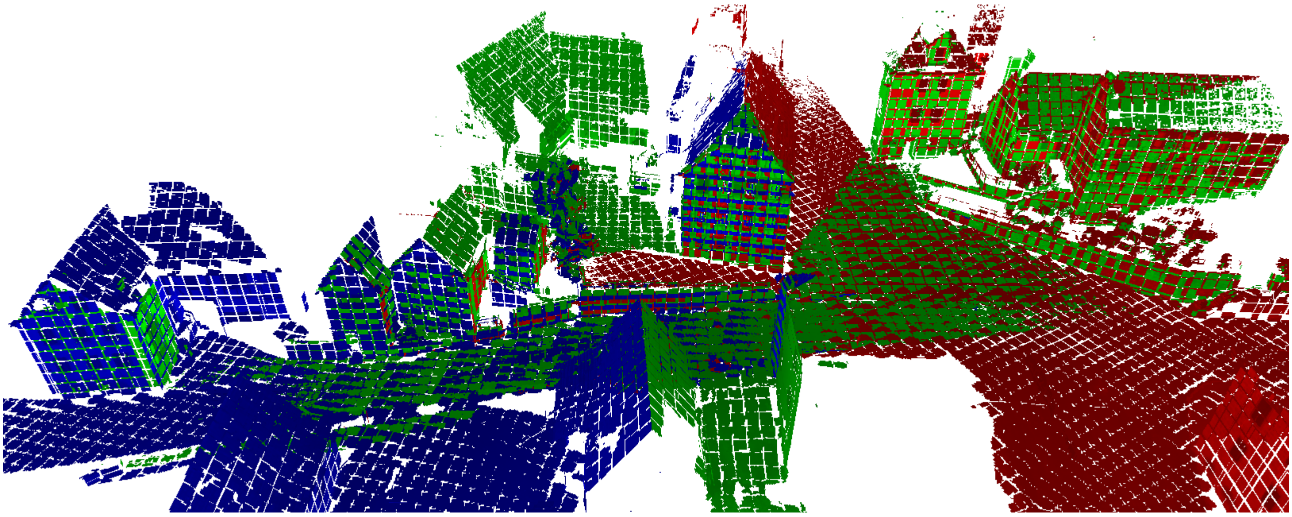


Figure 4: Resulting coarse registration of positions 1 (red), 2 (green), 3 (blue).

$\mathcal{P}_1$	$\mathcal{P}_2$	$n_1 n_2$	#Tests		#Inliers		
1	2	7776	3191	41%	21	26%	●
2	3	6240	1993	32%	15	23%	●
3	4	3705	1431	39%	17	30%	●
4	5	2337	980	42%	11	27%	●
2	6	4512	1479	33%	13	28%	●
6	7	2021	623	31%	4	9%	○
7	8	2064	550	27%	12	28%	●
8	11	2928	672	23%	6	12%	●
11	10	2501	646	26%	4	10%	○
10	9	2255	774	34%	8	20%	●
9	1	4455	2042	46%	6	11%	●
1	12	5346	2354	44%	17	26%	●
12	13	5610	2300	41%	18	27%	●
12	14	4620	1742	38%	5	8%	●
14	15	6160	2364	38%	18	26%	●
15	16	6336	2520	40%	19	26%	●
16	17	6120	2377	39%	16	22%	●
17	18	7395	2588	35%	20	24%	●
18	19	7743	2531	33%	16	18%	●
19	20	4272	1276	30%	10	21%	●
20	21	1632	364	22%	3	9%	○
1	22	6075	2910	48%	22	29%	●
22	23	4575	1864	41%	10	16%	●
23	24	3111	1162	37%	16	31%	●
24	25	2091	727	35%	12	29%	●
25	26	1681	555	33%	10	24%	●

Table 2: The number of tests actually needed compared to the total number of correspondences.

4 DISCUSSION

Robustness The robustness – i. e. outliers do not have an impact on the result – of our method is achieved at multiple levels:

- Alg. 3 is superior to RANSAC: It is also a generate-and-test scheme that uses the inlier count as quality measure, but with the distinction that the random sampling has been replaced by a complete search. RANSAC would return only one of the first m rows of Tab. 1 as its result whereas we get m valid rows.
- There exist several methods to determine m (Sec. 2.4). Hence, the estimated size of the inlier set can be checked.
- Estimation of the transformation parameters from the m rows can again be done through a robust scheme such as RANSAC or a least squares adjustment with outlier detection.

Complexity Alg. 3 has a complexity of $O(n^4)$ with n being the average number of planes in $\mathcal{P}_i$: Both the generate loop and the test loop nested inside compare all planes from $\mathcal{P}_1$ and $\mathcal{P}_2$ and thus each have complexity $O(n^2)$. Despite this, the implementation turned out very fast and finished within seconds even on a slow computer (Pentium III Mobile CPU @ 750 MHz) for multiple reasons:

- In practical applications, n is small. For the 26 datasets we have $n < 100$ and therefore can expect less than $100^4 = 100$ million runs of the innermost loop.
- The test of matching inclinations reduces the number of calls to the test loops. According to Tab. 2, less than half of the generated matches actually have to be checked.
- The innermost loop only contains comparisons so that it does not need much processing power. In fact, since there exist only $2n$ data elements, it is likely that the innermost loop will run entirely on the CPU cache.

Alternatively to the count of inliers one could also look for clusters in parameter space. The complexity for the matching is reduced to $O(n^2)$ as only the generation loop is needed, but a Hough like clustering would require a four dimensional accumulator for the parameters.

Execution times for the different stages are shown in Tab. 3. The first two stages take considerably longer because they process the complete cloud of about 100 million points so that I/O performance is an issue as well. In contrast to this, grouping and registration run very fast as these steps operate only on plane representations of the data.

Results Compared to the 100 million points of the raw point cloud, the surface elements are a significant data reduction – especially near the laser scanner, where the point density of the raw data is very high. As can be seen from Fig. 2, they describe the scene very well. The results of the grouping are good, because planes typically coincide with object surfaces.

As can be seen from the registration result in Tab. 1, every match from the inlier list (and in fact also the next two rows) contain similar transformation parameters. The final parameters for the coarse registration can easily be obtained from this list – e. g. by averaging over the automatically chosen inlier set. Additionally,

Stage	min	avg	max
Split point cloud into raster	249	307	343
Generate surface elements	136	215	563
Group to large planes	0.18	0.59	2.18
Coarse registration	0.17	0.57	1.29

Table 3: Minimum, average and maximum execution time in seconds for all datasets and the different stages on a Pentium M processor @ 1.7 GHz.

we thereby already dispose of a list of plane matches that could be used as starting point for the fine registration.

In all cases of a failure in Tab. 2, the datasets did not have enough matching planes in the overlapping area to get the inlier rate above the noise. During data acquisition, the positions have only been chosen to produce overlapping datasets that cover the complete scenario, so that the failures are due to bad sensor positioning. However, it is possible to work around such cases by first joining some datasets and then to try to match with this instead.

In Tab. 4, the ten best matches are shown for one of the failures. It can be seen that the first m rows do not necessarily contain only inliers. In order to detect a failure caused by a weak configuration, the parameters of the first rows must be checked for this.

Object features that belong together are overlayed very well in Fig. 4. The quality of the fusion can be seen best from the two nicely fitting blue-green and blue-red roofs in the middle of the figure. Close inspection of the dataset reveals that the registration is not perfect. While some surfaces actually coincide (these can be identified by their multi-color pattern) others – like the blue façade on the very left – are up to 1 m apart. A fine registration can easily fix this, as valid plane correspondences are available.

Errors in zenith direction We do not explicitly take into account the errors in the zenith direction. From analysis of the data we know that the absolute error is less than 30 mrad. However, the rotational component is also compensated via a z-shift. Actually this should even decrease bending of large models because the small rotations are not summed up.

5 CONCLUSIONS

We have presented a novel algorithm for the automatic marker-free coarse registration of two point clouds from terrestrial laser scanner – a task that is still considered difficult. The key idea was to recover initial parameters for rotation and translation from single plane correspondences only. As prerequisites it was required that the scene contains planar surfaces and that the laser scanner is set up with the local z-direction pointing upward. These conditions are easily fulfilled for built-up areas and available systems.

We have shown results for 26 datasets covering large parts of a moderately complex farming village. It has been shown that a reliable coarse registration of such real world data with an overlap of only 20–30% between neighboring positions is possible with our method. Some failures occured, but these were always due to an insufficient number of common planes in both datasets, caused by bad sensor placement. It is possible to automatically detect such bad configurations by analysis of the correspondence lists.

Current work is focussed on the extension of the approach to the automatic determination of topology of multiple datasets. The input shall be a number of datasets without any additional information and the output a neighborhood graph similar to Fig. 1 along with the complete set of transformation matrices for all datasets. We also plan to test the applicability of this algorithm to the mixed registration of terrestrial and airborne LIDAR data.

			Rotation	Translation		
#Inliers	p_i	p_j	α/rad	$\mathbf{t}_x$/m	$\mathbf{t}_y$/m	$\mathbf{t}_z$/m
4	29	13	0.1898	−11.31	18.57	0.11
3	44	16	−2.9469	9.24	−5.62	−2.37
3	28	27	−1.3910	−11.21	−24.28	−2.65
3	18	19	−1.3633	18.64	−24.13	−1.79
3	18	3	0.2133	−12.23	−11.49	−2.58
3	11	3	0.1621	−12.66	18.36	0.21
3	4	13	0.2979	−32.38	30.89	3.82
3	4	3	−1.2788	−13.93	17.33	1.30
3	2	16	0.2951	−33.09	34.87	5.29
3	2	2	−1.2707	−14.62	20.61	1.54

Table 4: Ten best matches from position 20 to 21. The horizontal line indicates the end of the automatically chosen inlier set.

REFERENCES

Bae, K.-H. and Lichti, D. D., 2004. Automated Registration of Unorganised Point Clouds from Terrestrial Laser Scanners. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B5. URL: www.isprs.org/istanbul2004/comm5/papers/553.pdf.

Ballard, D. H. and Brown, C. M., 1982. Computer Vision. Prentice-Hall. URL: homepages.inf.ed.ac.uk/rbf/BOOKS/BANDB.

Besl, P. J. and McKay, N., 1992. A Method for Registration of 3-D Shapes. PAMI 14(2), pp. 239–256.

Dold, C., 2005. Extended Gaussian Images for the Registration of Terrestrial Data. In: G. Vosselman and C. Brenner (eds), Laser scanning 2005, IAPRS, Vol. XXXVI-3/W19. URL: www.commission3.isprs.org/laserscanning2005/papers/180.pdf.

Dold, C. and Brenner, C., 2004. Automatic Matching of Terrestrial Scan Data as a Basis for the Generation of Detailed 3D City Models. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B3. URL: www.isprs.org/istanbul2004/comm3/papers/429.pdf.

Fischler, M. A. and Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM 24(6), pp. 381–395.

Grimson, W. E. L., 1990. Object Recognition by Computer: The Role of Geometric Constraints. The MIT Press.

Liu, R. and Hirzinger, G., 2005. Marker-free Automatic Matching of Range Data. In: R. Reulke and U. Knauer (eds), Panoramic Photogrammetry Workshop, IAPRS, Vol. XXXVI-5/W8. URL: www.informatik.hu-berlin.de/sv/pr/PanoramicPhotogrammetryWorkshop2005/Paper/PanoWS_Berlin2005_Rui.pdf.

Ripperda, N. and Brenner, C., 2005. Marker-free Registration of Terrestrial Laser Scans Using the Normal Distribution Transform. In: S. El-Hakim, F. Remondino and L. Gonzo (eds), 3D-ARCH 2005, IAPRS, Vol. XXXVI-5/W17. URL: www.commission5.isprs.org/3darch05/pdf/33.pdf.

von Hansen, W., Michaelsen, E. and Thönnessen, U., 2006. Cluster analysis and priority sorting in huge point clouds for building reconstruction. In: The 18th International Conference of Pattern Recognition.

A ROBUST ALGORITHM FOR ESTIMATING DIGITAL TERRAIN MODELS FROM DIGITAL SURFACE MODELS IN DENSE URBAN AREAS

Nicolas champion Didier Boldo

MATIS Laboratory, Institut Geographique National
2, Avenue Pasteur. 94165 SAINT-MANDE Cedex - FRANCE
Firstname.Lastname@ign.fr

Commission III/2

KEY WORDS: DSM, DTM, Elastic Grid, Outliers, Robust Statistics

ABSTRACT:

This paper describes an algorithm in order to derive DTMs (Digital Terrain Models) from correlation DSMs (Digital Surface Models) and above-ground (buildings and vegetation) masks in dense urban areas. Among all the methods found in literature, the Elastic Grid method shows a good capability to reconstruct the topographic surface. This method consists in interpolating height values under above-ground masks by minimizing an energy. Nevertheless, this method is ill-adapted to outliers in input data (above-ground points out of above-ground masks). The main contribution of our study is the use of a method based on robust statistics in order to reject outliers from calculation so that the final DTM fits the true topographic surface for the best. For that purpose, the initial Elastic Grid has been noticeably changed. The results of the new method for 2 test sites with a pixel ground size of 20 cm (the first one is relatively flat and the second one is hilly) show the quality of the final DTM and the robustness of our method. Tests have been carried out with lower resolution DSMs and without any mask and show the feasability of extending the method to a more general context.

1 INTRODUCTION

1.1 Background

In the past few years, DTMs (Digital Terrain Models) have increasingly been used as an important tool for engineering works or environmental applications (water overflowing control for example).
In urban areas, especially in a change detection process, a DTM can be very useful. As a matter of fact, using only the radiometric and texture information from RGB images or orthophotos are generally not sufficient to perform a good detection of buildings. The buildings height, calculated by making the difference between a DSM and the corresponding DTM, is often necessary. A lot of techniques exist to calculate DSMs (lidar scanning, stereo-matching algorithms) but few techniques are available to calculate a reliable DTM. It is sometimes possible to use a reference DTM (generally built manually or semi-manually) but, especially when working on high resolution data, such a reference is often not as accurate as the corresponding DSM: that leads to classical detection problems, typically a high underdetection rate (False Negative rate) and a high overdetection rate (False Positive rate). In order to make the underdetection rate tend towards 0 and to have the overdetection rate as small as possible, a good DTM i.e a good approximation of the topographic surface is necessary.
In this paper, a method for deriving a reliable DTM from a DSM, a building mask (derived from a database) and a vegetation mask is presented and evaluated.
In a DSM generation context, 2 families of techniques can be distinguished: lidar scanning and stereo-matching techniques. Lidar scanning methods have an undeniable advantage in rural areas, where they generally provide both DTMs and DSMs. In dense urban areas, the DTM can not be so easily obtained. Image-based DSM generation has then some advantages over lidar techniques. On the one hand, as images are most of time necessary for photogrammetric projects, generating a DSM with stereo-matching techniques does not implie additionnal costs. On the other hand, images provide a higher degree of internal geometric quality. The main challenge when deriving a DTM from a stereo-matching DSM is to filter and to discard outliers (blunders), i.e points that have too high an elevation compared with their surroundings (See Subsection 2.4 for a list of several kinds of outliers that can be found in a DSM). Almost all the methods found in literature try to deal with this problem, as shown in the following subsection.

1.2 Related Works

Several methods to derive a DTM from a DSM have been considered.
The first method for estimating DTMs is based on morphological operators. A description can be found in (Weidner, 1996). This method is not robust when DSMs contain outliers. To solve this problem, (Eckstein and Munkelt, 1995) introduces the Dual Rank Filter . Unfortunately, the structuring element remains difficult to define without any a priori knowledge about the study area (urban / industrial . . .). Moreover, the method can fail because of big aggregations of vegetation or big buildings (typically a cathedral) in city centres. Eventually, such a tool generally relocates ridges and thalwegs.
An other strategy consists in using parametric methods in order to reconstruct the topographic surface. The final DTM is supposed to belong to a family of parameterized surfaces and these parameters have to be derived from observations . Unfortunately, as shown in (Jordan and Cord, 2004), all the kinds of surface can not be reconstructed and the reconstruction is all the more difficult and inaccurate as the study area is big.
A large set of methods based on triangulation have been found in literature (See (Baillard, 2003) for an example). The main challenge here is to choose ground points and then to triangulate them in order to interpolate height in the whole scene. As the final surface depends on this choice, finding good criteria to select true ground points (and not outliers!) is determinant. An other weak point is that the final surface is not regular (i.e not differentiable) and so not natural .
A good method that gives regular surfaces is the Elastic Grid. Former works when producing the French Elevation Database have shown its capability to represent the topographic surface naturally and correctly (Masson d'Autume, 1978).

1.3 Presentation

The Elastic Grid method has always been used in order to derive a DTM from a set of extracted points (for example, contour lines). There are no ouliers in input data in that case. Tests have shown the limits of the algorithm in presence of outliers: when applied too roughly, the algorithm creates artificial blobs (See figure 3 in Section 4). The main goal of this study is also to show the feasability of adapting the method to such a context.

In Section 2, input data are first described. In Section 3, our method is detailed. In Section 4, the results of our method are presented and qualitative and quantitative results are given. Eventually, forthcoming research axes are given in concluding remarks

2 INPUT DATA

The algorithm presented in Section 3 uses a DSM, a building mask and a vegetation mask to estimate the final DTM.

2.1 DSM

In our study, 2 stereo-matching algorithms are used to compute the initial DSM. The first one is described in (Baillard and Dissart, 2000) and is based on cost minimization along epipolar lines. This cost takes discontinuities in heights and radiometric similarities into account. The second one is based on a multi-resolution implementation of Cox and Roy optimal flow image matching algorithm. More details are given in (Pierrot-Deseilligny and Paparoditis, 2006). DSMs have a resolution of 20 cm.

2.2 Buildings Masks

The buildings mask is directly derived from a cadastral database. This database is a vector database where buildings ground footprints are represented in 2D. As it is produced manually, it has a good precision but contains some discrepancies (for example demolished buildings), as shown in Subsection 2.4.

2.3 Vegetation Masks

The vegetation mask is produced by applying a threshold on NDVI images (Normalized Difference Vegetation Index). This index is high for vegetation due to the fact most of the visible light is absorbed and nearly all infrared light is reflected.

$$NDVI = \frac{NIR - Red}{NIR + Red} \tag{1}$$

This index is computed on orthophotos so that vegetation masks can be easily superimposed on buildings masks and DSMs. RGB and IR images used for orthophotos are calibrated but, to avoid problems linked to the hot-spot phenomenon, source images are corrected with an algorithm that performs a radiometric equalization. The model used for this correction is a parameterized and semi-empirical BRDF model. More details can be found in (Paparoditis et al., 2006).

2.4 Comments

Three types of outliers can be distinguished in input data. Firstly, some buildings are not represented in masks as the database is not necessarily up-to-date (Case 1 in Figure 1). Secondly, some above-ground points are out of them: cars, street furniture, newspapers kiosks...(Case 2 in Figure 1). Eventually, a lot of outliers are located at the edges of buildings (Case 3 in Figure 1). That comes from the fact that buildings footprints are given by the outlines of the walls in the cadastral database and that the walls limits do not necessarily fit the roof limits given in a DSM.

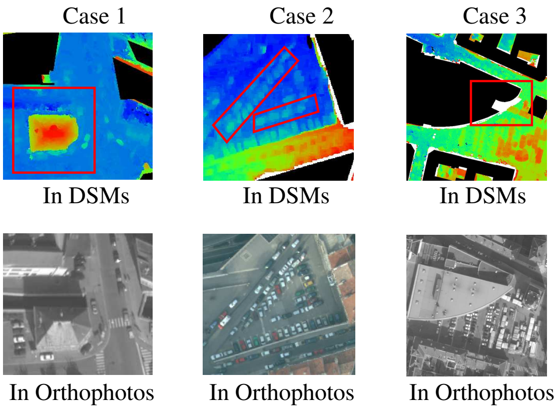


Figure 1: Problems in buildings masks. Outliers (highlighted in red boxes) in above-ground masks sumperimposed on DSMs (upper images) and corresponding orthophotos (bottom images)

3 METHOD

In this section, a short mathematical description of the Elastic Grid method is given. This method is based on a functional (that contains a regularization term and a data term) to minimize. Firstly, the importance of the norm ρ to use in the data term is shown. Secondly, the 3 parameters to be tuned (a tuning constant c intrinsic to the norm ρ, the standard deviation σ and the smoothing coefficient λ) are introduced and justified. Eventually, the general strategy used for the process is detailed.

3.1 Theoretical Aspects

The Elastic Grid method estimates the reconstructed topographic surface by fitting an elastic surface to a finite sample of observation points (i.e points considered as ground points in the DSM i.e points out of above-ground masks). Mathematically, this is equivalent to the minimization of this functionnal:

$$E(z) = K(z) + \lambda G(z, \sigma) \tag{2}$$

- the Regularization Term $K(z)$ corresponds to the discrete approximation of the second derivative of the surface to reconstruct. This term minimizes the mean quadratic curvature (i.e height variations) of the final DTM.

$$K(z) = \sum_{l=1}^{M}\sum_{c=1}^{N}\left(\frac{\partial^2 z_{c,l}}{\partial c^2}\right)^2 + \sum_{l=1}^{M}\sum_{c=1}^{N}\left(\frac{\partial^2 z_{c,l}}{\partial l^2}\right)^2 \tag{3}$$

$$= \sum_{l=1}^{M}\sum_{c=2}^{N-1}\Big(z_{c-1,l} - 2z_{c,l} + z_{c+1,l}\Big)^2 + \sum_{l=2}^{M-1}\sum_{c=1}^{N}\Big(z_{c,l-1} - 2z_{c,l} + z_{c,l+1}\Big)^2 \tag{4}$$

- the Data Term $G(z,\sigma)$ corresponds to the distance between the model to estimate and observations.

$$G(z,\sigma) = \sum_{l=1}^{M}\sum_{c=1}^{N}\rho\Big(\frac{z_{c,l} - obs_{c,l}}{\sigma}\Big) \tag{5}$$

  where $z_{c,l}$ is the value of the estimated model at the (c,l) pixel, $obs_{c,l}$ the corresponding observation value and ρ, the norm used in order to calculate the distance.

- the factor λ is used in order to balance both terms. The higher λ is, the better the model fits observations. The smaller λ is, the smoother the model is.

3.2 Description of the grid parameters

The initial Elastic Grid method uses the Least-Squares method to minimize the difference between the model to estimate and observations. Therefore, a classical euclidean norm is introduced in the data term.

$$\sum_{l=1}^{M}\sum_{c=1}^{N}\left(z_{c,l} - obs_{c,l}\right)^2$$

Such an approach is not robust to outliers in input data and can become very unstable. The method used in our work in order to reject outliers from calculation is derived from the M-estimator technique. This technique reduces the effect of outliers by replacing the sum of squared differences (residuals) by a certain function ρ that is symmetric, positive-definite, with a minimum at zero and less increasing than square.

$$\sum_{l=1}^{M}\sum_{c=1}^{N}\rho(z_{c,l} - obs_{c,l})$$

Name of the Tested Norm	Tested Norm $\rho(x)$
L1L2	$2 \times (\sqrt{1 + \frac{x^2}{2}} - 1)$
Cauchy	$\frac{c^2}{2} \times \log\left(1 + (\frac{x}{c})^2\right)$
Geman-McLure	$\frac{\frac{x^2}{2}}{1+x^2}$
Huber $\begin{cases} if\|x\| < c \\ if\|x\| \geq c \end{cases}$	$\frac{x^2}{2}$ $c \times (\|x\| - \frac{c}{2})$
Tukey $\begin{cases} if\|x\| < c \\ if\|x\| \geq c \end{cases}$	$\frac{c^2}{6} \times (1 - (1 - (\frac{x}{c})^2)^3)$ $\frac{c^2}{6}$

Table 1: Robust norms tested in our study

All the norms tested in our study are listed in Table 1. In most norms, there is a tuning constant c. It is all the more important as it determines points whose influence will be reduced in the process. In (Zhang, 1997), the author considers that noise follows a gaussian law $\mathcal{N}(0,\,1)$ and gives, for each norm, the value for the tuning constant c in order to reach the 95 percent asymptotic efficiency on the standard normal distribution. (Zhang, 1997) shows that $c = 4.6851$ for the Tukey's norm for instance.

In our work, the difference $z_{c,l} - obs_{c,l}$ is assumed to follow a gaussian law but is not standardized $(z_{c,l} - obs_{c,l}) \sim \mathcal{N}(0,\,\sigma)$, what prevents us to apply the previously mentioned values directly. Therefore, a standard deviation σ must be calculated. It is calculated with the classical estimator, in a clean and horizontal area (typically a square without any tree, car…) so that it is not biased because of the presence of outliers.

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(r_i - \overline{r}_i)^2} \qquad (6)$$

where n is the number of pixels in clean areas, $r_i = z_i - obs_i$ is the difference between the estimated model and correponding observations and $\overline{r}_i = \frac{1}{n} \times \sum_{i=1} r_i$ is the mean value of differences in clean areas.

3.3 General Strategy

As can be seen in Figure 2, several steps are necessary in order to compute the final DTM. The process is divided into 4 steps: Initialization, Paving, Elastic Grid and Mosaicking.

As the process to minimize $E(z)$ is iterative, a good way to decrease the number of iterations is to calculate an initial solution. This approximate solution is given by a method based on a dual rank filter. This tool has some imperfections mentioned in Subsection 1.2 but is fast and easy to implement. Moreover, as the convergence is all the more long as the study area is big (the ratio calculation time / study area size is not linear), a paving strategy (with a 1000 × 1000 tile) has been set up. A mosaicking process is consequently necessary.

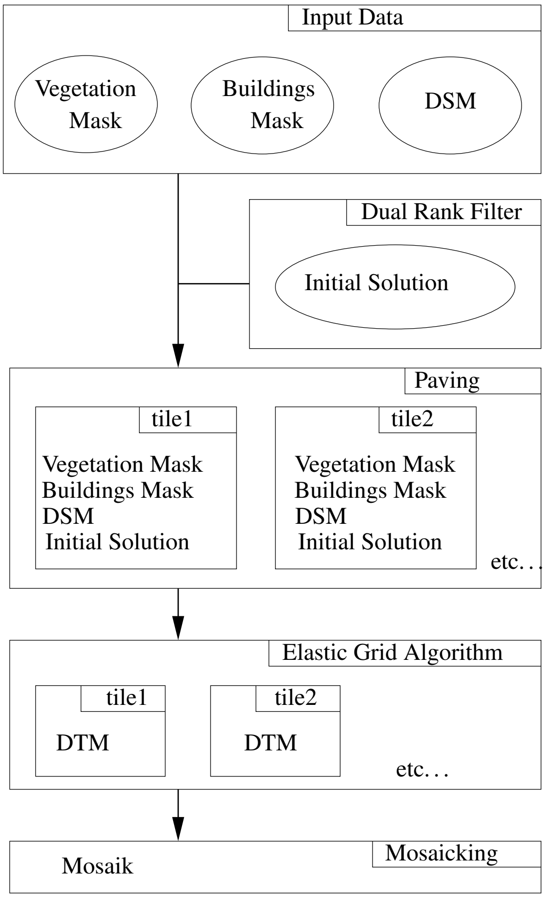


Figure 2: General Strategy

4 RESULTS AND DISCUSSION

4.1 Test areas and data

2 test sites are presented:

- Amiens City Centre, France
  - Pixel Ground Size = 20cm
  - Area = 800m × 800m $\simeq 0.64km^2$
  - Terrain Type: relatively flat
  - Land Cover Type: dense urban area
  - Matching Algorithm: (Baillard and Dissart, 2000)
- Marseille City Centre, France
  - Pixel Ground Size = 20cm
  - Area = 950m × 950m $\simeq 0.90km^2$
  - Terrain Type: hilly

- Land Cover Type: dense urban area
- Matching Algorithm: (Pierrot-Deseilligny and Paparoditis, 2006)

Several norms have been tested in our work. For each norm, the tuning constant c is firstly found in literature. Secondly, as our data are not standardized, a standard deviation must be calculated in a clean area to standardize them and to be able to apply the value of c found in literature. Once these 2 factors fixed, a sensitivity study is carried out in a small area (typically, a 300 × 300 area) to determine the best value to give to the smoothing coefficient λ. Results are assessed by visual inspection (difference between the DSM and the DTM) and by editing profiles along lines in the DSM and corresponding lines in the DTM. Experiments have shown the terrain is best reconstructed with a Tukey's norm. This norm has also been used with the 3 factors (c, σ and λ) previously determined to process the whole area. The corresponding results are given in the next subsection.

4.2 Results

As the process to minimize $E(z)$ is long, an optimized numerical library (GNU Scientific Library) is used. The calculation time with a 1.8 GHz PC is about 30 hours in Amiens (size of the whole scene: 4000 × 4000 / 36 tiles) and 40 hours in Marseille (size of the whole scene: 4600 × 4600, 49 tiles). The qualitative and quantitative results of our algorithm are now given.

4.2.1 Qualitative Results The benefit of introducing a robust norm (instead of the classical euclidean norm) is clearly shown in Figure 3 where results in a small test area are presented. The initial DSM is displayed on the upper left image. Above-ground masks are sumperimposed on the initial DSM and are displayed on the upper right image. Final DTMs are displayed on bottom images (on the left, the one processed with the classical Elastic Grid, on the right, with our algorithm). All the figures are displayed by using the same scale in height as the bottom right DTM. In this way, readers can have a first visual idea of the quality of DTMs. In the bottom left DTM, artificial blobs are created. That comes from outliers present in input data: as they are considered by the classical Elastic Grid algorithm as true ground points, they have the same influence in the process and deviate the reconstructed topographic surface upwards. As the new algorithm introduces a robust norm in order to reject outliers, such a bad effect does not occur.

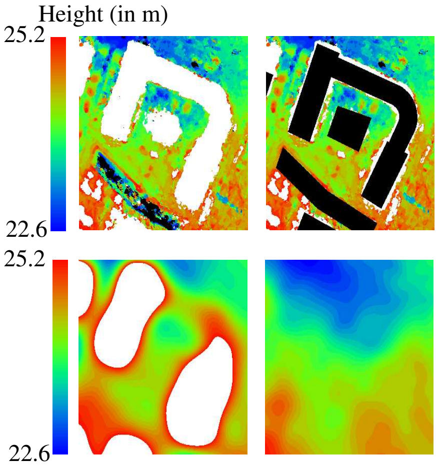


Figure 3: Comparison between the classical Elastic Grid algorithm and the new algorithm. In white, points higher than 25.2m

The qualitative results when applying our algorithm are now given in Figures 5 - 8 (Amiens) and Figures 9 - 12 (Marseille). As detailed in the introduction, a good detection of above-ground points implies a good reconstruction of the topographic surface. Therefore, the difference $DSM - DTM$ is a good indicator for assessing the quality of final products and is given in Figure 8 (Amiens) and Figure 12 (Marseille).

4.2.2 Quantitative Results In order to assess results quantitatively, a statistical analysis is firstly performed by manually extracting ground points from initial DSMs and by comparing them with corresponding points extracted from DTMs. A bias, a standard deviation and a RMS are then calculated and are shown in Table 2. Secondly, profiles along arrows in DSMs and corresponding arrows in DTMs are edited (See Figures 4 - 13 - 14).

Area	Bias	σ	RMS	Nb Pts
Amiens City Centre	0.424	0.447	0.616	1104
Marseille City Centre	0.307	2.606	2.604	886

Table 2: Stastistical Analysis

The topographic surface is well reconstructed in Amiens. The bias is slightly positive. That means the reconstructed surface is slightly disturbed by the presence of outliers. Nevertheless, as shown in the profile (Figure 4) along the green arrow (Figures 5 and 7), the final DTM perfectly clings to points in streets and courtyards and reconstructs all the small undulations of the terrain.

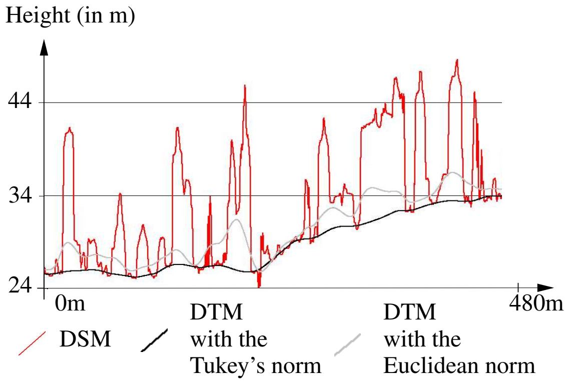


Figure 4: Profiles in Amiens along the green arrow in the DSM and DTM (See Figures 5 and 7). In red, the initial DSM. In black, the result with the new algorithm. In light grey, the result of the classical Elastic Grid. The blobs that correspond to buildings and vegetation in the DSM are filtered with the new method. Artificial blobs are created when using the euclidean norm.

Some problems occur in Marseille. The bias is small and proves that the computed DTM is a good approximation of the topographic surface. As shown in the profile (Figure 13) along the green arrow in the DSM and DTM (Figures 9 and 11), the final DTM generally clings to true ground points and filters blobs corresponding to buildings. In that case, results are similar to Amiens. Nevertheless, a high RMS outlines problems in specific areas, especially in breaklines areas. For example, some problems occur in the profile along the red arrow (Figure 14). The left breakline is well reconstructed, which proves the capability of our algorithm to reconstruct such terrain types. Nevertheless, the right breakline is completely eroded. This problem firstly comes from the terrain type (a 50m high cliff difficult to reconstruct), secondly from the combination of using a robust norm and a coarse initial solution: using a robust norm is an efficient means

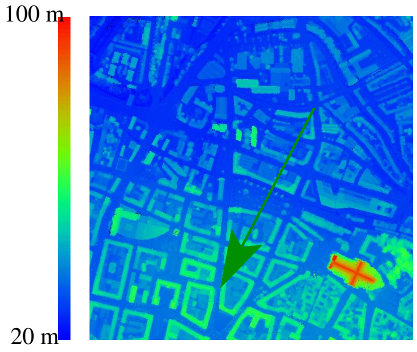


Figure 5: Amiens - Initial DSM (Top View)

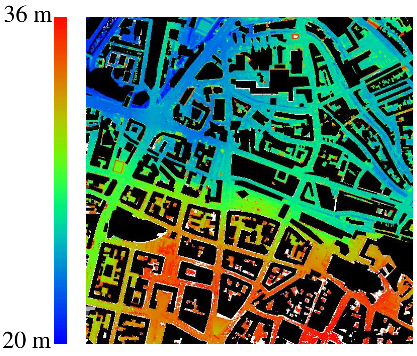


Figure 6: Amiens - Mask over DSM (Top View)

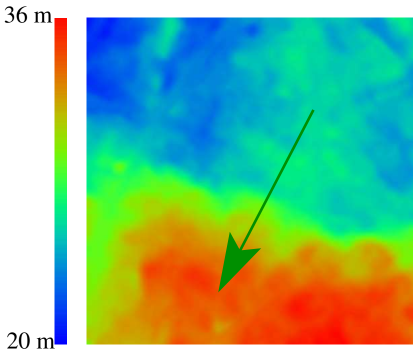


Figure 7: Results in Amiens - DTM (Top View)

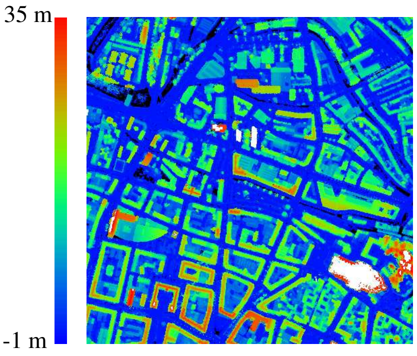


Figure 8: Results in Amiens - $DSM - DTM$ (Top View)

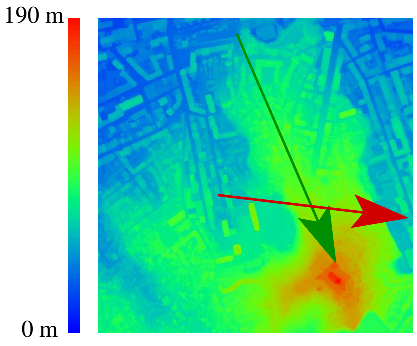


Figure 9: Marseille - Initial DSM (Top View)

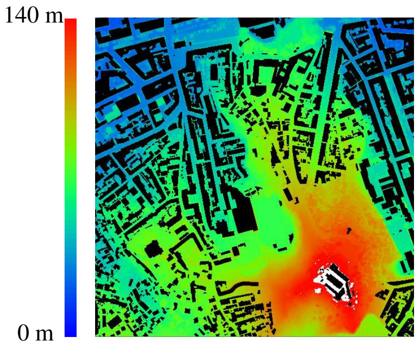


Figure 10: Marseille - Mask over DSM (Top View)

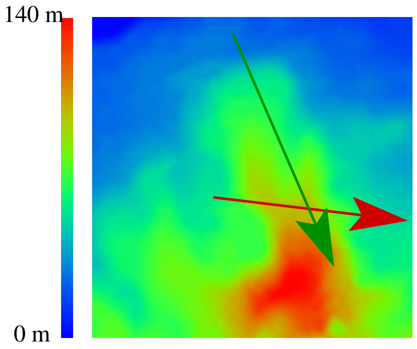


Figure 11: Results in Marseille - DTM (Top View)

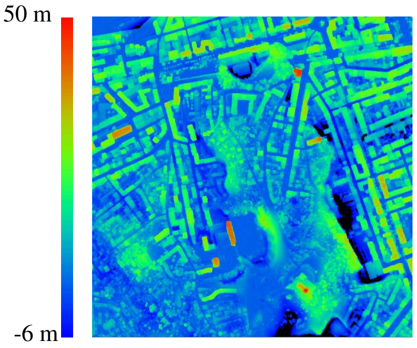


Figure 12: Results in Marseille - $DSM - DTM$ (Top View)

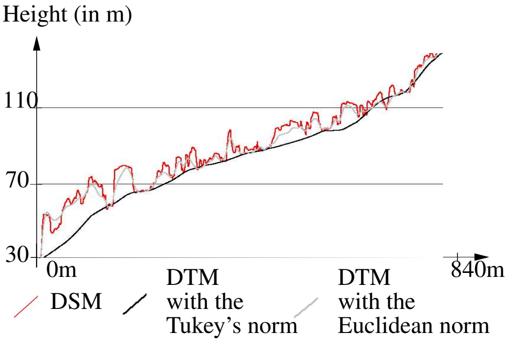


Figure 13: Profiles in Marseille along the green arrow. In red, the initial DSM. In black, the result of the new algorithm. In light grey, the result of the classical Elastic Grid. The blobs that correspond to buildings and vegetation in the DSM are filtered with the new method.

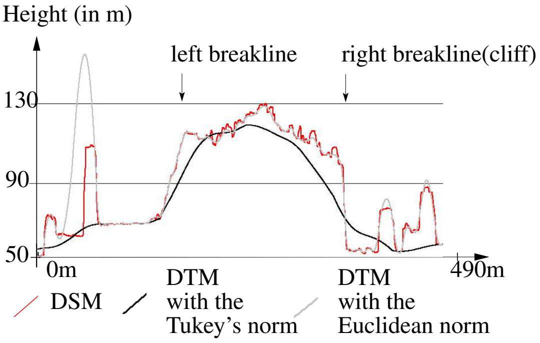


Figure 14: Profiles in Marseille along the red arrow. In red, the initial DSM. In black, the result of the new algorithm. In light grey, the result of the classical Elastic Grid. The left breakline is well reconstructed. The right breakline is too eroded (Problems in the initialization step).

to reject definitely outliers; problems occur when rejected points are inliers (true ground points). In Marseille, the right breakline is eroded in the initialization step (because of the use of a dual rank filter, see Subsection 1.2 for more explanations). As the difference $z_{c,l} - obs_{c,l}$ is then too big in that area, all the corresponding points (even inliers) are considered outliers: the process does not use these points to reconstruct the topographic surface and the breakline is not modelled very well in the end. A multi-resolution coarse-to-fine approach is being considered to give our algorithm a more precise initial solution.

5 CONCLUSIONS AND FUTURE WORK

Our goal was to compute a DTM from a DSM and above-ground masks, in dense urban areas and in a difficult context (presence of outliers in input data). The initial Elastic Grid has been revised by introducing a robust norm (instead of the classical euclidean norm) and by setting the grid parameters (the tuning constant c, the standard deviation σ and the smoothing coefficient λ) suitably. The results presented in this paper and corresponding to different configurations (high resolution DSM, relatively flat / hilly city centres) show the robustness of our approach. In order to make our method as generic as possible, tests have been carried out with lower resolution DSMs (pixel ground size = 70cm and 5m). First results are promising and show the feasability of extending our method to such a resolution. An other research axis is to adapt our method so that the initial above-ground mask becomes optional. The challenge here is to reject above-ground points rapidly. On the one hand, the points where the difference between the model and observations is negative and that are also closer to the model to estimate (typically ground points) must have their influence in the calculation increased. On the other hand, points with a positive difference (typically above-ground points) must be rejected. In (Jordan et al., 2002), a dissymetric norm is used for that purpose: the euclidean norm is used wherever the difference $z_{c,l} - obs_{c,l}$ is negative and the Tukey's norm is used wherever it is positive. As the euclidean norm is more increasing than the Tukey's norm, lowest points are advantaged. The main idea is to introduce such a dissymetric norm (by replacing the non robust euclidean norm with the more robust L1L2 norm) in our grid data term.

6 ACKNOWLEDGEMENTS

The first author would like to thank his colleagues for useful and valuable comments, suggestions and discussions.

REFERENCES

Baillard, C., 2003. Production of urban DSMs combining 3D vector data and stereo aerial imagery. In: ISPRS Archives, Vol. XXXIV, Part3/W8.

Baillard, C. and Dissart, O., 2000. A stereo matching algorithm for urban Digital Elevation Models. ASPRS Photogrammetric Engineering and Remote Sensing 66(9), pp. 1119–1128.

Eckstein, W. and Munkelt, O., 1995. Extracting objects from digital terrain models. In: Remote Sensing and Reconstruction for Three-Dimensional Objects and Scenes, Vol. 2572, pp. 43–51.

Jordan, M. and Cord, M., 2004. Etude portant sur l'état de l'art, l'évaluation et la comparaison de méthodes de segmentation sol / sursol à partir de modèles numériques d'élévation. Technical report, ETIS, Equipe Image, ENSEA.

Jordan, M., Cord, M. and Belli, T., 2002. Building detection from high resolution digital elevation models in urban areas. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV-3B, Graz, Austria, pp. 96–99.

Masson d'Autume, G., 1978. Construction du modèle numérique d'une surface par approximations successives: application aux modèles numériques de terrain. Bulletin de la Société Française de Photogrammétrie et Télédétection 71, pp. 33–41.

McKeown, D., Bulwindle, T., Cochran, S., Harvey, W., McGlone, J. and Shufelt, J., 2000. Performance evaluation for automatic feature extraction. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII-B2, Amsterdam, The Netherlands, pp. 379–394.

Paparoditis, N., Souchon, J., Martinoty, G. and Pierrot-Desseiligny, M., 2006. High-end aerial digital cameras and their impact on the automation and quality of the production workflow. IJPRS. To Appear.

Pierrot-Deseilligny, M. and Paparoditis, N., 2006. A multiresolution and optimization-based image matching approach: An application to surface reconstruction from spot5-hrs stereo imagery. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXVI, Ankara, Turkey.

Weidner, U., 1996. An approach to building extraction from digital surface models. In: Proceedings of the 18th ISPRS Congress, Comm. III, WG 2, Vol. 43 - Building Detection from a Single Image, pp. 924–929.

Zhang, Z., 1997. Parameter estimation techniques: A tutorial with application to conic fitting. Image and Vision Computing Journal 15(1), pp. 59–76.

SURFACE RECONSTRUCTION ALGORITHMS FOR DETAILED CLOSE-RANGE OBJECT MODELING

Fabio Remondino, Li Zhang

Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland
E-mail: fabio@geod.baug.ethz.ch
Web: http://www.photogrammetry.ethz.ch

KEY WORDS: Surface Reconstruction, Modeling, Precision, Matching

ABSTRACT

Nowadays 3D modeling is generally performed using image or range data. Range sensors are getting a quite common source of data for modeling purposes due to their speed and ability to capture millions of points. In this paper we report about two surface measurement algorithms for precise and detailed object reconstruction from terrestrial images. Photogrammetry has all the potentialities to retrieve the same details of an object that range sensors can achieve. Using advanced measurement techniques, which combine area-based and feature-based matching algorithms we are able to generate dense point clouds of complex and free-form objects, imaged in closely or widely separated images. Different examples are reported to show the potentiality of the methods and their applicability to different close-range data sets.

1. INTRODUCTION

Three-dimensional modeling from images is a great topic of investigation in the research community, even if range sensors are becoming more and more a common source and a good alternative for generating 3D information quickly and precisely. 3D modeling of a scene should be meant as the complete process that starts with the data acquisition and ends with a virtual model in three dimensions visible interactively on a computer. The interest in 3D modeling is motivated by a wide spectrum of applications, such as animation, navigation of autonomous vehicles, object recognition, surveillance, visualization and documentation.

In the last years different solutions for image-based 3D modeling have been developed. Most of the current reliable and precise approaches are based on semi-automated procedures, therefore the introduction of automated algorithms is a key goal in the photogrammetric and vision communities. 3D modeling methods can be classified according to the level of automation or the required input data while their strength is reflected by the variety of scene that can be processed and the level of detail that can be reconstructed.

The common fully automated 'shape from video' framework [e.g. Fitzibbon & Zisserman, 1998; Nister, 2001; Pollefeys et al., 2004] requires good features in the images, very short baseline and large overlap between consecutive frames, requirements which are not always satisfied in practical situations, due to occlusions, illumination changes and lack of texture. So far, automated surface reconstruction methods, even if able to recover complete 3D geometry of an object, reported errors between 3% and 5% [Pollefeys et al., 1999], limiting their use for applications requiring only nice-looking 3D models. Furthermore, post-processing operations are generally required, which means that user interaction is still needed. Indeed the most impressive results are achieved with interactive methods and taking advantage of the environment constraints, in particular for architectural objects. For different applications, such as cultural heritage documentation, semi-automated methods are still preferred as smoothed results, missing details or lack of accuracy are not accepted.

In this article we report about two surface matching algorithms developed for the precise and detailed measurement and 3D modeling of complex and free-form terrestrial objects, like pots, reliefs, statues, façades, etc. Commercial photogrammetric stations generally fail with tilted close-range images, therefore the topic still need some developments. We will concentrate only on the measurement of the object surface, assuming the calibration and orientation of the images already performed. As the network configurations that allow full and precise camera calibration are usually very different from those used for scene reconstruction, we generally first calibrate the camera using the most appropriate set of images and afterwards recover the orientation parameters of the scene's images using the calibration results. The orientation is generally performed by means of a photogrammetric bundle adjustment, extracting the required tie points with automated approaches [Remondino & Ressl, 2006] or manual measurements.

The first surface measurement algorithm presented afterwards matches the points in image-pairs, the second one works simultaneously with many images. Both methods require some seed points between the images at the beginning of the process, to initialize it and improve the performances near surface discontinuities. The seed points can be provided manually (stereo or monocular measurements) or extracted automatically, leading to a fully automated surface reconstruction method. The number of seed points depends on the set of images, their disparity and texture content. Starting from these seeds points, a dense and robust set of correspondences covering the area of interest is generated.

Our research aims to combine area-based and feature-based matching techniques to recover complete and detailed 3D surfaces. The methods can cope with depth discontinuity, wide baselines, repeated pattern, occlusions and illumination changes.

In the next sections, after an overview of image-based modeling works and matching strategies, the two matching strategies are described in details. Then some examples demonstrating the potentialities of the algorithms and their applicability to different close-range data sets are reported and discussed. Results in form of 3D point clouds, shaded and textured 3D models are shown.

2. 3D MODELING FROM IMAGES

Recovering a complete, detailed, accurate and realistic 3D model from images is still a difficult task, in particular if uncalibrated or widely separated images are used. Firstly because the wrong camera parameters lead to inaccurate or deformed results. Secondly because a wide baseline between the images generally requires the user interaction in the measurement phase.

The research activities in terrestrial image-based modeling can be generally divided in area-based [e.g. Pollefeys et al., 2004] and feature-based [e.g. Schmid & Zisserman, 2000] methods. A more detailed classification of point-based methods is:

1. *Approaches that try to get automatically a 3D model of the scene from uncalibrated images* (also called 'shape from video' or 'VHS to VRML' or 'Video-to-3D'). The fully automated procedure widely reported in the vision community [Fitzibbon & Zisserman, 1998; Pollefeys et al., 1999; Nister 2001; Mayer, 2003] starts with a sequence of images taken with an uncalibrated camera. The system then extract interest points, sequentially match them across the view-pairs and compute the camera parameters as well as the 3D coordinates of the matched points using robust techniques. This is done in a projective geometry framework and is usually followed by a bundle adjustment. A self-calibration, to compute the interior camera parameters, is afterwards performed in order to obtain a metric reconstruction, up to a scale, from the projective one. The 3D surface model is then automatically generated by means of dense depth maps on image pairs. See [Scharstein & Szeliski, 2002] for a recent overview of dense stereo correspondence algorithms. The key to the success of these automated approaches is the very short interval between consecutive images. Some approaches have been also presented for the registration of widely separated views [Pritchett & Zisserman, 1998; Matas et al., 2002; Xiao & Shah, 2003; Lowe 2004] but their reliability and applicability for automated image-based modeling of complex objects is still not satisfactory, as they yield mainly a sparse set of matched feature points. Dense matching results under wide baseline conditions were instead reported in [Megyesi & Chetverikov, 2004; Strecha et al., 2004].
2. *Approaches that perform an automated 3D reconstruction of the scene from oriented images*. The automated 3D reconstruction is generally based on object constraints, like verticality and perpendicularity [Werner & Zisserman, 2002; Van den Heuvel, 2003; Wilczkowiak et al., 2003] or using the geometric epipolar constraint [Gruen et al., 2001].
3. *Approaches that perform a semi-automated 3D reconstruction of the scene from oriented images*. The semi-automated modeling rely on the human operator and produced so far the most impressive results, in particular for architectural objects [Debevec et al., 1996; El-Hakim, 2000, 2002; Gibson et al., 2002]. The interactive work consists of the topology definition, segmentation, editing and 3D data post-processing. The degree of automation increases when certain assumptions about the object, such as perpendicularity or parallel surfaces, can be introduced.

Manual measurements are also performed in some projects, generally for complex architectural objects or in cultural heritage documentations where highly precise and detailed results are required [Gruen et al., 2004]. Manual measurements are time consuming and provide for less dense 3D point clouds, but have higher reliability compared to automated procedures.

3. MATCHING FOR SURFACE MEASUREMENTS

Image matching represents the establishment of correspondences between primitives extracted from two or more images. In its oldest form, image matching involved 4 transformation parameters (cross-correlation) and could already provide for successful results [Foerstner, 1982]. Further extensions considered a 6- and 8-parameters transformation, leading to the well known non-linear Least Squares Matching (LSM) estimation procedure [Gruen, 1985; Foerstner, 1986]. Gruen [1985] and Gruen & Baltsavias [1986] introduced the Multi-Photo Geometrical Constraints into the image matching procedure (MPGC) and integrated also the surface reconstruction into the process. Then from image space, the matching procedure was generalized to object space, introducing the concept of 'groundel' or 'surfel' [Wrobel, 1987; Helava, 1988].

Even if more than three decades have been devoted to the image matching problem, nowadays some important limiting factors still remain. A fully automated, precise and reliable image matching method, adaptable to different image sets and scene contents is not available, in particular for close-range images. The limits stay in the insufficient understanding and modeling of the undergoing processes (human stereo vision) and the lack of appropriate theoretical measures for self-tuning and quality control. The design of an image matcher should take into account the topology of the object, the primitives used in the process, the constraint used to restrict the search space, a strategy to control the matching results and finally optimization procedures to combine the image processing with the used constraints. The correspondences between images are matched starting from primitives (features and image intensity patterns) and using similarity measures. Ideally we would like to find the correspondences of every image pixel. But, in practice, coherent collection of pixels and features are generally matched.

A part from simple points, the extraction of feature lines (see [Dhond & Aggarwal, 1989; Ziou & Tabbone, 1998] for a review) is also a crucial step in the surface generation procedure. Lines (edgel) provide more geometric information than single points and are also useful in the surface reconstruction (e.g. as breaklines) to avoid smoothing effects on the object edges. Edge matching [Vosselman, 1992; Gruen & Li, 1996; Schmid & Zisserman, 2000] establishes edge correspondences over images acquired at different standpoints. Similarity measures from the edges attributes (like length, orientation and absolute gradient magnitude) are a key point for the matching procedure. Unfortunately in close-range photogrammetry, the viewpoints might change consistently; therefore similarity measures are not always useful for edge matching.

4. STEREO-PAIR SURFACE MEASUREMENT

The first developed algorithm is a stereo matcher with the additional epipolar geometric constraint. The method was firstly developed for the measurement of human body parts [D'Apuzzo, 2003] and afterwards also applied to full human body reconstruction [Remondino, 2004] and rock slopes retrieval [Roncella et al., 2005]. It has been now extended to include also edge matching. The main steps of the process are:

1. *Image pre-processing:* the images are processed with the Wallis filter [Wallis, 1976] for radiometric equalization and especially contrast enhancement. The filter enables a strong enhancement of the local contrast by retaining edge details

and removing low-frequency information in the image. The filter parameters are automatically selected analyzing the image histogram.

2. *Point matching:* the goal is to produce a dense and robust set of corresponding points between image-pairs. Starting from few seed points well distributed in the images, the automated process establishes correspondences by means of LSM. The images are divided in polygonal regions according to which of the seed point is closest. Starting from the seed points, the automated process produce a dense set of image correspondences in each polygonal region by sequential horizontal and vertical shifts. One image is used as template and the other as search image. The algorithm matches correspondences in the neighborhood of a seed point in the search image (approximation point) by minimizing the sum of the squares differences of the gray value between the two image patches. If the orientation parameters of the cameras are available, the epipolar geometric constraints between the images can also be used in the matching process. Generally two stereo-pairs (i.e. a triplet) are used: the matcher searches the corresponding points in the two search images independently and at the end of the process, the data sets are merged to become triplets of matched 2D points.
3. *Edge matching:* the approach extracts line features based on the edge detection and linking proposed in [Canny, 1986] and [Henricsson & Heitger, 1994]. For each image, only the edges longer than a certain threshold are kept. Afterwards an edge matching is performed for each image pair of the set. Firstly the middle points of the edges are matched, providing a preliminary list of edge correspondences. Afterwards, starting from the matched middle point, the other points lying on the edge are matched in a propagative way.
4. *3D Point cloud generation:* the 2D matched points and edges are transformed in 3D data by forward ray intersection, using the camera exterior orientation parameters.

The developed matching process works on image pairs and integrates the epipolar constraint in the least squares estimation, limiting the patch in the search image to move along the epipolar line. To evaluate the quality of the matching results, different indicators are used: a posteriori standard deviation of the least squares adjustment, standard deviation of the shift in x and y directions and displacement from the start position in x and y direction. Thresholds for these values are defined manually for different cases, according to the level of texture in image and to the type of template. The definition of the seed points is generally crucial, in particular if discontinuities are present on the surface. The matcher, working only with stereo-pairs, is less robust than a multi-image strategy which takes into account all the available and overlapping images at the same time, but it is still able to provide for accurate and detailed 3D surfaces.

5. MULTI-IMAGE SURFACE MEASUREMENT

The multi-image matching approach was originally developed for the processing of the very high-resolution TLS Linear Array images [Gruen & Zhang, 2003] and afterwards modified to accommodate any linear array sensor [Zhang & Gruen, 2004; Zhang, 2005]. Now it has been extended to process other image data such as the traditional aerial photos or close-range images. The multi-image approach uses a coarse-to-fine hierarchical solution with an effective combination of several image matching algorithms and automatic quality control. Starting from the known calibration and orientation parameters, the approach (Figure 1) essentially performs three mutually connected steps:

1. *Image pre-processing:* the set of available images is processed combining an adaptive smoothing filter and the Wallis filter [Wallis, 1976], in order to reduce the effects of the radiometric problems such as strong bright and dark regions and optimizes the images for subsequent feature extraction and image matching. Furthermore image pyramids are generated.
2. *Multiple Primitive Multi-Image (MPM) matching:* this part is the core of the all strategy for accurate and robust surface reconstruction, utilizing a coarse-to-fine hierarchical matching strategy. Starting from the low-density features in the lowest resolution level of the image pyramid, the MPM matching is performed with the aid of multiple images (two or more), incorporating multiple matching primitives (feature points, grid points and edges) and integrating local and global image information. The MPM approach consists of 3 integrated subsystems (Figure 1): the feature point extraction and matching, the edge extraction and matching (based on edge geometric and photometric attributes) and the relaxation based relational matching procedure. Within the pyramid levels, the matching is performed with an extension of the standard cross-correlation technique (Geometrically Constrained Cross-Correlation -GC^3-). The MPM matching part exploits the concept of multi-image matching guided from object space and allows reconstruction of 3D objects by matching all available images simultaneously, without having to match all individual stereo-pairs and merge the results. Moreover, at each pyramid level, a TIN is reconstructed from the matched features using the constrained Delauney triangulation method. The TIN is used in the subsequent pyramid level for derivation of approximations and adaptive computation of some matching parameters.
3. *Refined matching:* a modified Multi-Photo Geometrically Constrained Matching (*MPGC*) and the Least Squares B-Spline Snakes (*LSB-Snakes*) methods are used to achieve potentially sub-pixel accuracy matches and identify some inaccurate and possibly false matches. This is applied only at the original image resolution level. The surface derived from the previous MPM step provides well enough approximations for the two matching methods and increases the convergence rate.

The main characteristics of the multi-image-based matching procedure are:

- Truly multiple image matching: the approach does not aim at pure image-to-image matching but it directly seeks for image-to-object correspondences. A point is matched simultaneously in all the images where it is visible and exploiting the collinearity constraint, the 3D coordinates are directly computed, together with their accuracy values.
- Matching with multiple primitives: the method is a robust hybrid image matching algorithms which takes advantage of both area-based matching and feature-based matching techniques and uses both local and global image information. In particular, it combines an edge matching method with a point matching method through a probability relaxation based relational matching process.
- Self-tuning matching parameters: they are automatically determined by analyzing the results of the higher-level image pyramid matching and using them at the current pyramid level. These parameters include the size of the correlation

window, the search distance and the threshold values. The adaptive determination of the matching parameters results in higher success rate and less mismatches.

- High matching redundancy: exploiting the multi-image concept, highly redundant matching results are obtained. The high redundancy also allows automatic blunder detection. Mismatches can be detected and deleted through the analysis and consistency checking within a small neighbourhood.

More details of the matching approach are reported in Zhang [2005].

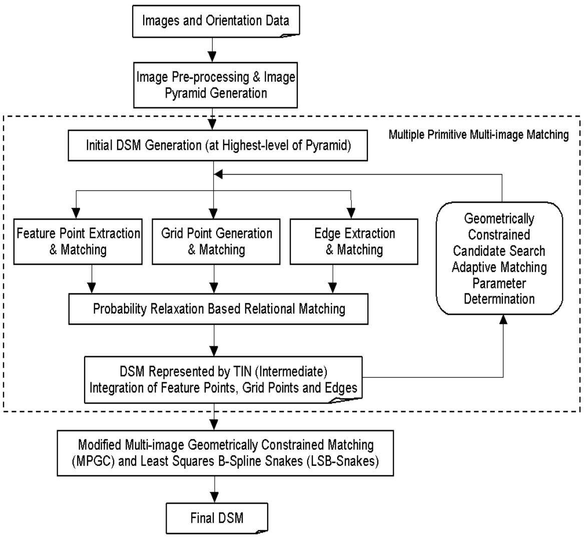

Images and Orientation Data
Image Pre-processing & Image Pyramid Generation
Multiple Primitive Multi-image Matching
Initial DSM Generation (at Highest-level of Pyramid)
Feature Point Extraction & Matching
Grid Point Generation & Matching
Edge Extraction & Matching
Geometrically Constrained Candidate Search Adaptive Matching Parameter Determination
Probability Relaxation Based Relational Matching
DSM Represented by TIN (Intermediate) Integration of Feature Points, Grid Points and Edges
Modified Multi-image Geometrically Constrained Matching (MPGC) and Least Squares B-Spline Snakes (LSB-Snakes)
Final DSM

Figure 1: Workflow of the automated DSM generation approach. The approach consists of 3 mutually connected components: the image preprocessing, the multiple primitive multi-image (MPM) matching and the refined matching procedure.

6. EXPERIMENTS

We have performed many tests on different close-range data sets with the two surface reconstruction approaches. So far the results are checked just visually, as no reference is available. In the future an accuracy test should be performed. In the next sections we report results from widely separated images, untextured surfaces and detailed heritage objects. More examples are reported in our homepage.

Test1. Three images of the main door of the S. Marco church in Venice (Italy) are used. The image size is 2560x1920 pixels. The triplet is acquired under a wide baseline (base-to-distance ratio ~ 1:1.4) and very fine details are present on the object. Both methods could correctly retrieve the surface details, as shown in Figure 3. The stereo-pair strategy matched approximately 590 000 points between the two pairs while the multi-image matching recovered ca 700 000 points.

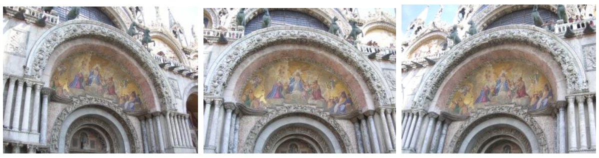


Figure 2: The three images of the church's façade acquired under a large baseline.

Test2. A very small pot (ca 3x4 cm) is modeled with the two presented matching strategies. Six images, with a size of 1856 x 1392 pixels are used. The detailed results are shown in Figure 4 as textured and shaded surface model.

Test 3. The data set consists of 6 images of a Maya relief in Edzna, Mexico. The object is approximately 4 meters long and 2 meters high. The images have different light conditions and scales. Due to the frontal acquisition, the upper horizontal part of the relief is not visible in the images, leading to some gaps in the matching point results and some stretching effects in the meshed model. Both methods could reconstruct all the details of the heritage. The stereo-pairs approach (performed on 4 pairs) generated ca 860 000 points and 7 900 edges while with the multi-image strategy a cloud of 1 940 000 points and 23 000 edges was produced. The results are shown in Figure 5.

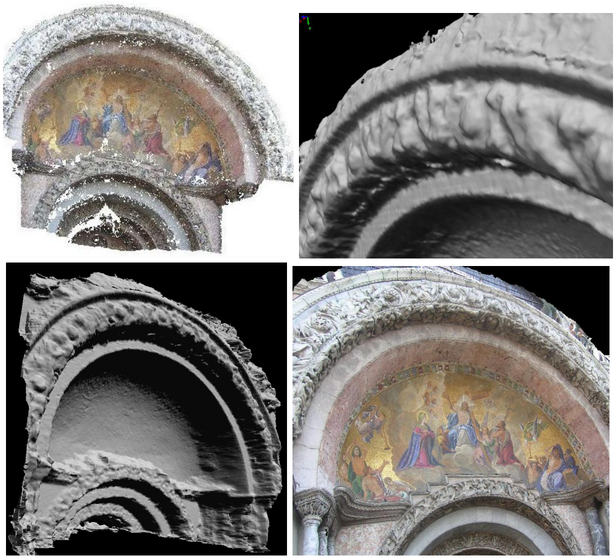


Figure 3: Results of the stereo-pair matching method (above): recovered 3D point cloud, displayed with pixel intensity values and a particular of the generated shaded model. Views of the shaded and textured surface generated with the multi-image method (below).

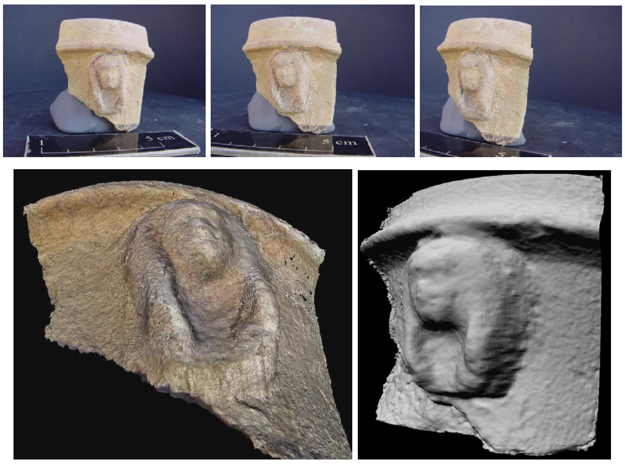


Figure 4: Three (out of 6) image of the small pot (above). The surface reconstructed with the stereo-pair matching and with the multi-photo approach.

7. CONCLUSIONS AND OUTLOOK

We have presented two matching strategies for the precise surface measurement and 3D reconstruction of complex and detailed terrestrial objects. The stereo-pair approach constraints the search of correspondences along the epipolar line while the 3D coordinates of points and matched edges are computed in a second phase, using rejection criteria for the forward ray intersection. The multi-image approach is more reliable and precise but requires very accurate image orientation parameters to exploit the collinearity constraint within the least squares matching estimation. The maximum orientation errors in image space should be less than 2-3 pixels.

The two approaches use points and edges to retrieve all the surface details and they have both advantages and disadvantages. They can be applied to short or wide baseline images and can cope with scale changes, different illumination conditions or repeated pattern. Employing the precise LSM algorithm, they can recover sub-pixel accuracy matches. They both need some seed points to initialize the matching procedure and the number of seed points is strictly related to the image texture and surface discontinuities.

The results so far achieved are promising but more tests have to be performed as well as an accuracy assessment of the two strategies. Photogrammetry has all the potentiality to retrieve the same results (details) than range sensors. But to asses the accuracy of the systems is not an easy task. Assessment on the whole measured surface would require the two models to be in the same reference systems or to set one model as reference and transform the second one into the first reference system.

REFERENCES

Baltsavias, E.P., 1991: Multi-Photo geometrically constrained matching. PhD Thesis, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, 221 pages

D'Apuzzo, N., 2003: Surface Measurement and Tracking of Human Body Parts from Multi Station Video Sequences. PhD Thesis Nr. 15271, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, 147 pages

Debevec, P., Taylor, C. and Malik, J., 1996: Modelling and rendering architecture from photographs: a hybrid geometry and image-based approach. ACM Proceedings of SIGGRAPH '96, pp. 11-20

Dhond, U.R. and Aggarwal, J.K., 1989: Structure from Stereo. IEEE Transaction on System, Man and Cybernetics, Vol. 19(6), pp. 1489-1510

El-Hakim, S., 2000: A practical approach to creating precise and detailed 3D models from single and multiple views. IAPRS, 33(B5A), pp 122-129

El-Hakim, S., 2002: Semi-automated 3D reconstruction of occluded and unmarked surfaces from widely separated views. IAPRS, 34(5), pp. 143-148, Corfu, Greece

Fitzgibbon, A and Zisserman, A., 1998: Automatic 3D model acquisition and generation of new images from video sequence. Proceedings of European Signal Processing Conference, pp. 1261-1269

Foerstner, W., 1982: On the geometric precision of digital correlation. International Archives of Photogrammetry, Vol. 24(3), pp.176-189

Foerstner, W., 1986: A feature based correspondence algorithm for image matching. International Archives of Photogrammetry, Vol. 26(3), Rovaniemi

Gibson, S., Cook, J. and Hubbold, R., 2002: ICARUS: Interactive reconstruction from uncalibrated image sequence. ACM Proceedings of SIGGRAPH'02, Sketches & Applications

Gruen, A., 1985: Adaptive least square correlation: a powerful image matching technique. South African Journal of PRS and Cartography, Vol. 14(3), pp. 175-187

Gruen, A. and Baltsavias, E., 1986: Adaptive least squares correlations with geometrical constraints. Proc. of SPIE, Vol. 595, pp. 72-82

Gruen, A., and Li, H., 1996: Linear feature extraction with LSB-Snakes from multiple images. IAPRS, Vol. 31(B3), pp. 266-272

Gruen, A., Zhang, L. and Visnovcova, J., 2001: Automatic reconstruction and visualization of a complex Buddha Tower of Bayon, Angkor, Cambodia. Proceedings 21.Wissenschaftlich-Technische Jahrestagung der DGPF, pp. 289-301

Gruen, A., Zhang L., 2003. Automatic DTM Generation from TLS data. In: Gruen/Kahman (Eds.), Optical 3-D Measurement Techniques VI, Vol. I, ISBN: 3-906467-43-0, pp. 93-105.

Gruen, A., Remondino,F., Zhang, L., 2004: Photogrammetric Reconstruction of the Great Buddha of Bamiyan, Afghanistan. The Photogrammetric Record, Vol. 19(107)

Helava, U.V., 1988: Object-space least-squares correlation. PE&RS, Vol. 54(6), pp. 711-714

Henricsson O. and Heitger F., 1994: The role of key-points in finding contours. ECCV'94, Vol. 2, pp. 371-383

Lowe, D., 2004: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, Vol. 60(2), pp. 91-110

Matas, J., Chum, O., Urban, M. and Pajdla, T., 2002: Robust wide baseline stereo from maximally stable extremal regions. Proceedings of BMVC, pp. 384-393

Mayer, H., 2003: Robust orientation, calibration, and disparity estimation of image triplets. 25th DAGM Pattern Recognition Symposium (DAGM03), Number 2781, series LNCS, Michaelis/Krell (Eds.), Magdeburg, Germany

Megyesi, Z. and Chetverikov, D., 2004: Affine propagation for surface reconstruction in wide baseline stereo. Proc. ICPR 2004, Cambridge, UK

Nister, D., 2001: Automatic dense reconstruction from uncalibrated video sequences. PhD Thesis, Computational Vision and Active Perception Lab, NADA-KHT, Stockholm, 226 pages

Pollefeys, M., Koch, R. and Van Gool, L., 1999: Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. IJCV, Vol. 32(1), pp. 7-25

Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J. and Koch, R., 2004: Visual modeling with a hand-held camera. IJCV, Vol. 59(3), pp. 207-232

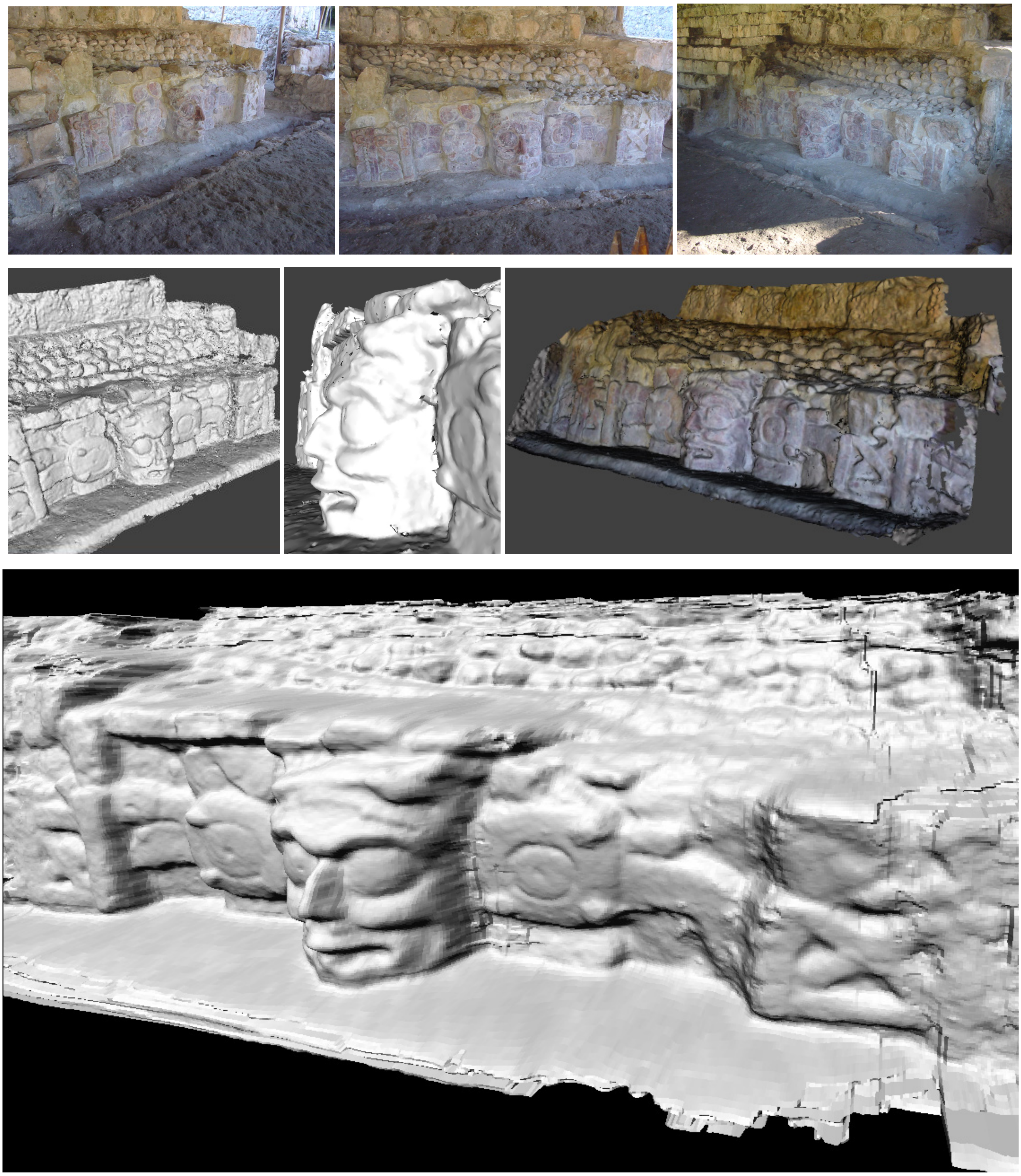


Figure 5: Three (out of six) images of a Maya relief (above). Results of the stereo-pair matching (4 pairs) displayed as shaded model (with a close view of the mesh) and as textured model (middle). Shaded model obtained with the multi-image matcher simultaneously run on 6 images (bottom).

Pritchett, P. and Zisserman, A. 1998: Matching and reconstruction from widely separated views. 3D Structure from Multiple Images of Large-Scale Environments, LNCS 1506

Remondino, F., 2004: 3D Reconstruction of Static Human Body Shape from Image Sequence. Computer Vision and Image Understanding, Vol. 93(1), pp. 65-85

Remondino, F. and Ressl, C., 2006: Overview and experiences in automated markerless image orientation. IAPRS, Comm. III Symposium, Bonn, Germany. In press.

Roncella, R., Forlani, G. and Remondino, F., 2005: Photogrammetry for geological applications: automatic retrieval of discontinuity in rock slopes. Videometrics VIII - Beraldin, El-Hakim, Gruen, Walton (Eds), SPIE-IS&T Electronic Imaging Vol. 5665, pp. 17-27

Scharstein, D. and Szeliski, R., 2002:. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV, 47(1/2/3): 7-42

Schmid, C. and Zisserman, A., 2000: The geometry and matching of lines and curves over multiple views. IJCV, Vol. 40(3), pp. 199-233

Strecha, C., Tuytelaars, T. and Van Gool, L., 2003: Dense Matching of Multiple Wide-baseline Views. IEEE Proceedings of ICCV'03, Vol.2, pp. 1194-1201

Van den Heuvel, F., 2003: Automation in architectural photogrammetry. PhD Thesis, Publication on Geodesy 54, Netherlands Geodetic Commission

Vosselman, G., 1992: Relational matching. Lecture Notes in Computer Science, No. 628, Springer Berlin, 190 pages

Wallis, R., 1976: An approach to the space variant restoration and enhancement of images. Proc. of Symposium on Current Mathematical Problems in Image Science, Naval Postgraduate School, Monterey, CA

Werner, T. and Zisserman, A., 2002: New technique for automated architectural reconstruction from photographs. Proceedings 7th ECCV, Vol.2, pp. 541-555

Wilczkowiak, M., Trombettoni, G., Jermann, C., Sturm, P. and Boyer, E., 2003: Scene modeling based on constraint system decomposition techniques. IEEE Proceedings 9th ICCV, pp. 1004-1010

Xiao, J. and Shah, M. 2003: Two-frame wide baseline matching. IEEE Proceeding of 9th ICCV, Vol.1, pp. 603-610

Wrobel, B., 1987: Facet Stereo Vison (FAST Vision) – A new approach to computer stereo vision and to digital photogrammetry. Proc. of ISPRS Intercommission Conference on 'Fast Processing of Photogrammetric Data', Interlaken, Switzerland, pp. 231-258

Zhang, L., Gruen, A., 2004. Automatic DSM Generation from Linear Array Imagery Data. IAPRS, Vol. 35(B3), pp. 128-133

Zhang, L., 2005: Automatic Digital Surface Model (DSM) generation from linear array images. PhD Thesis Nr. 16078, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, 199 pages

Ziou, D. and Tabbone, S., 1998: Edge Detection Techniques - An Overview. Journal of Pattern Recognition and Image Analysis. Vol. 8, pp. 537-559

IMAGE-BASED 3D SURFACE RECONSTRUCTION BY COMBINATION OF SPARSE DEPTH DATA WITH SHAPE FROM SHADING AND POLARISATION

Pablo d'Angelo and Christian Wöhler

DaimlerChrysler Group Research, Machine Perception
P. O. Box 2360, D-89013 Ulm, Germany

KEY WORDS: Industry, Metrology, Application, Polarization, Three-dimensional Reconstruction, Close Range Photogrammetry

ABSTRACT:

In this contribution we describe an image-based framework for 3D surface reconstruction by a combined analysis of reflectance, polarisation, and sparse depth data. An error functional consisting of several error terms related to the measured reflectance and polarisation properties and the depth data is minimised in order to compute a dense surface gradient field and in a subsequent step a dense 3D surface profile. The error terms related to reflectance and polarisation directly depend on the surface gradients, while the depth-related error term describes the deviation between the 3D surface profile implied by the surface gradient field and the measured depth points. Hence, we suggest an optimisation scheme that simultaneously adapts the surface gradients to the measured reflectance and polarisation data and to the surface slopes implied by depth differences between pairs of depth points. To increase the robustness of the optimisation scheme it is implemented as a multi-scale approach, thus providing a result largely independent of the provided initialisation. In our system the sparse depth data are provided by a correlation-based stereo vision algorithm, but in principle arbitrary sources of depth data are possible. We evaluate the algorithm based on synthetic ground truth data, demonstrating that the combined approach increases the accuracy of 3D surface reconstruction, compared to the result obtained by applying either of the techniques alone. Furthermore, we report 3D reconstruction results for a raw forged iron surface and compare them to ground truth depth data obtained by means of a laser focus profilometer. This evaluation yields a depth accuracy (root-mean-square deviation) of our approach of 62 μm, which is of the same order of magnitude as the intrinsic roughness of the metallic surface.

1 INTRODUCTION

Three-dimensional surface reconstruction is an important topic in various application areas, such as quality inspection and reverse engineering. Many image-based reconstruction methods have been proposed, based on photometric as well as geometric principles. Well known geometric approaches include stereo and structure from motion (Faugeras, 1993), and projection of structured light (Batlle et al., 1998). In practice, even passive methods such as stereo and structure from motion often require structured illumination to artificially produce texture required for a dense reconstruction of the surface (Calow et al., 2002). Reconstruction algorithms based on photometric methods include shape from shading (SfS) and polarisation (Horn and Brooks, 1989; d'Angelo and Wöhler, 2005a; Miyazaki et al., 2003). In contrast to the geometric approaches, they can be used to reconstruct smooth, textureless surfaces without structured illumination.

A combined reconstruction based on geometric and photometric reconstruction methods is desirable, since both approaches complement each other. A number of approaches to combine stereo and shape from shading have been proposed in the literature. Cryer et al. (1995) fuse low-pass filtered stereo depth data and high-pass filtered shape from shading depth data. Samaras et al. (2000) introduce a surface reconstruction algorithm that performs stereo analysis of a scene and uses a minimum description length metric to selectively apply SfS to regions with weak texture. A related approach (Fassold et al., 2004) integrates stereo depth measurements into a variational SfS algorithm and estimates surface shape, light source direction, and diffuse reflectance map.

In this paper we propose a combination of shape from photopolarimetric reflectance (SfPR) with 3D depth measurements from arbitrary sources. Our approach extends a variational SfPR framework (d'Angelo and Wöhler, 2005a) by adding an additional depth error term to the error function. A multi-scale approach is applied to reconstruct the surface gradient field. In this framework we assume known reflectance functions and light source positions.

2 SHAPE FROM PHOTOPOLARIMETRIC REFLECTANCE

In our scenario, we will assume that the surface $z(x, y)$ to be reconstructed is illuminated by a point light source and viewed by a camera, both situated at infinite distance in the directions $\vec{s}$ and $\vec{v}$, respectively. The xy plane is parallel to the image plane. Parallel unpolarised incident light and an orthographic projection model are assumed. For each pixel location (u, v) of the image we intend to derive a depth value $z(u, v)$. The surface normal is given in the so-called gradient space by the vector $\vec{n} = (-p, -q, 1)^T$ with $p = \partial z/\partial x$ and $q = \partial z/\partial y$. The incidence angle θ_i is defined as the angle between surface normal $\vec{n}$ and illumination direction $\vec{s}$, the emission angle θ_e as the angle between surface normal $\vec{n}$ and viewing direction $\vec{v}$, and the phase angle α as the angle between illumination direction $\vec{s}$ and viewing direction $\vec{v}$. A measure for the intrinsic reflectivity of the surface is given by the surface albedo $\rho(u, v)$.

In the framework of shape from photopolarimetric reflectance (SfPR), the light reflected from a surface point located at the world coordinates (x, y, z) with corresponding image coordinates (u, v) is described by the observed pixel intensity $I(u, v)$, the polarisation angle $\Phi(u, v)$ (i. e. the direction in which the light is linearly polarised), and the polarisation degree $D(u, v)$. Measurement of polarisation properties is thus limited to linear polarisation while circular or elliptic polarisation is not taken into account. It is assumed that models are available that express these photopolarimetric properties in terms of the surface orientation $\vec{n}$, illumination direction $\vec{s}$, and viewing direction $\vec{v}$. These models may either be physically motivated or empirical (cf. Section 2.2) and are denoted in this paper by R (intensity reflectance), R_Φ

(polarisation angle reflectance), and R_D (polarisation degree reflectance). The aim of surface reconstruction in the presented framework is to determine for each pixel (u, v) the surface gradients $p(u, v)$ and $q(u, v)$, given the illumination direction $\vec{s}$ and the viewing direction $\vec{v}$, such that the modelled photopolarimetric properties of the pixel correspond to the measured values:

$$\begin{aligned} I(u,v) &= R\left(p(u,v), q(u,v), \vec{s}, \vec{v}\right) && (1)\\ \Phi(u,v) &= R_\Phi\left(p(u,v), q(u,v), \vec{s}, \vec{v}\right) && (2)\\ D(u,v) &= R_D\left(p(u,v), q(u,v), \vec{s}, \vec{v}\right) && (3) \end{aligned}$$

The reflectance functions (1)–(3) may depend on further, e. g. material-specific, parameters which possibly in turn depend on the pixel coordinates (u, v), such as the surface albedo $\rho(u, v)$ which influences the intensity reflectance R. A local approach to obtaining the surface gradients $p(u, v)$ and $q(u, v)$ consists of solving the nonlinear system of Eqs. (1)–(3) individually for each pixel location (u, v) either exactly or in the least-mean-squares sense (d'Angelo and Wöhler, 2005b). For integration of large-scale depth information, however, a global optimisation scheme for determining the surface gradient field is more favourable, since it is not straightforward to include global depth constraints into the local approach to estimate $p(u, v)$ and $q(u, v)$.

2.1 Global optimisation scheme

In this section we describe a global approach to adapt the surface gradients $p(u, v)$ and $q(u, v)$ to the observed photopolarimetric properties $I(u, v)$, $\Phi(u, v)$, and $D(u, v)$ by solving the system of equations (1)–(3) (d'Angelo and Wöhler, 2005a). The 3D surface profile $z(u, v)$ is then obtained by integration of the surface gradient field by solving the Poisson equation $\Delta z = p_x + p_y$ (Simchony et al., 1991).

2.1.1 Determination of surface gradients and relative depth

The solving technique is based on the optimisation of a global error function (Horn, 1989; Jiang and Bunke, 1997; d'Angelo and Wöhler, 2005a). One part of this error function is the intensity error term

$$e_I = \sum_{l=1}^{L} \sum_{u,v} \Big[I^{(l)}(u,v) - R\left(\rho(u,v), p(u,v), q(u,v), \vec{s}^{(l)}, \vec{v}\right) \Big]^2 . \quad (4)$$

The number of light sources and thus of acquired images is given by L. We assume orthographic projection, hence $\vec{s}^{(l)}$ and $\vec{v}$ are constants.

As the pixel intensity information alone is not necessarily sufficient to provide an unambiguous solution for the surface gradients $p(u, v)$ and $q(u, v)$, a regularisation constraint e_s is introduced which requires smoothness of the surface, i. e. for example small absolute values of the directional derivatives of the surface gradients. We will therefore make use of the error term

$$e_s = \sum_{u,v} \left[p_x^2 + p_y^2 + q_x^2 + q_y^2 \right] . \quad (5)$$

(Horn, 1989; Jiang and Bunke, 1997). In the scenarios regarded in this paper, the assumption of a smooth surface is realistic. For wrinkled surfaces, where using Eq. (5) leads to an unsatisfactory result, it can be replaced by the departure from integrability error term described in detail by Horn (1989).

To integrate polarisation angle and degree into the 3D surface reconstruction framework, we define two error terms e_Φ and e_D which denote the deviations between the measured values and those computed using the corresponding phenomenological model:

$$e_\Phi = \sum_{l=1}^{L} \sum_{u,v} \Big[\Phi^{(l)}(u,v) - R_\Phi\left(p(u,v), q(u,v), \vec{s}^{(l)}, \vec{v}\right) \Big]^2 \quad (6)$$

$$e_D = \sum_{l=1}^{L} \sum_{u,v} \Big[D^{(l)}(u,v) - R_D\left(p(u,v), q(u,v), \vec{s}^{(l)}, \vec{v}\right) \Big]^2 . \quad (7)$$

Based on the feature-specific error terms e_I, e_Φ, and e_D, a combined error term e is defined which takes into account the reflectance and polarisation properties:

$$e = e_s + \lambda e_I + \mu e_\Phi + \nu e_D . \quad (8)$$

Minimising error term (8) yields the surface gradients $p(u, v)$ and $q(u, v)$ that optimally correspond to the observed reflectance and polarisation properties, where the Lagrange parameters λ, μ, and ν denote the relative weights of the individual reflectance-specific and polarisation-specific error terms. With the discrete approximations $p_x(u, v) = \left[p(u + 1, v) - p(u - 1, v)\right]/2$ and $p_y(u, v) = \left[p(u, v + 1) - p(u, v - 1)\right]/2$ for the second derivatives of the surface and $\bar{p}(u, v)$ as the local average over the four nearest neighbours of pixel (u, v) we obtain an iterative update rule for the surface gradients by setting the derivatives of the error term e with respect to them to zero:

$$\begin{aligned} p_{n+1} = \bar{p}_n &+ \lambda \sum_{l=1}^{L} \left(I - R(\bar{p}_n, \bar{q}_n)\right) \left.\frac{\partial R}{\partial p}\right|_{\bar{p}_n, \bar{q}_n} \\ &+ \mu \sum_{l=1}^{L} (\Phi - R_\Phi(\bar{p}_n, \bar{q}_n)) \left.\frac{\partial R_\Phi}{\partial p}\right|_{\bar{p}_n, \bar{q}_n} \\ &+ \nu \sum_{l=1}^{L} (D - R_D(\bar{p}_n, \bar{q}_n)) \left.\frac{\partial R_D}{\partial p}\right|_{\bar{p}_n, \bar{q}_n} . \end{aligned} \quad (9)$$

A corresponding expression for q is obtained in an analogous manner. This derivation is described in more detail in Jiang and Bunke (1997). The initial values $p_0(u, v)$ and $q_0(u, v)$ must be provided based on a-priori knowledge about the surface or on independently obtained depth data (cf. Section 3). The surface profile $z(u, v)$ is then derived from the resulting gradients $p(u, v)$ and $q(u, v)$ by means of numerical integration of the gradient field (Simchony et al., 1991).

The reconstruction is done in a multi-scale approach to speed up convergence and avoid getting stuck in local minima. Reconstruction of the gradient field starts at a low resolution and is repeated on the next pyramid level, using the gradients estimated at the previous level as initial gradient values.

2.2 Determination of empirical photopolarimetric models

For the purpose of determination of empirical reflectance and polarisation models for the surface material the surface normal $\vec{n}$ of a flat sample is adjusted by means of a goniometer, while the illumination direction $\vec{s}$ and the viewing direction $\vec{v}$ are constant over the image. Over a wide range of surface normals $\vec{n}$, five images are acquired through a linear polarisation filter at orientation angles ω of 0°, 45°, 90°, 135°, and 180°. For each filter orientation ω, an average pixel intensity over an image area containing a flat part of the sample surface is computed. To the measured pixel intensities we fit a sinusoidal function of the form

$$I(\omega) = I_c + I_v \cos(\omega - \Phi) \quad (10)$$

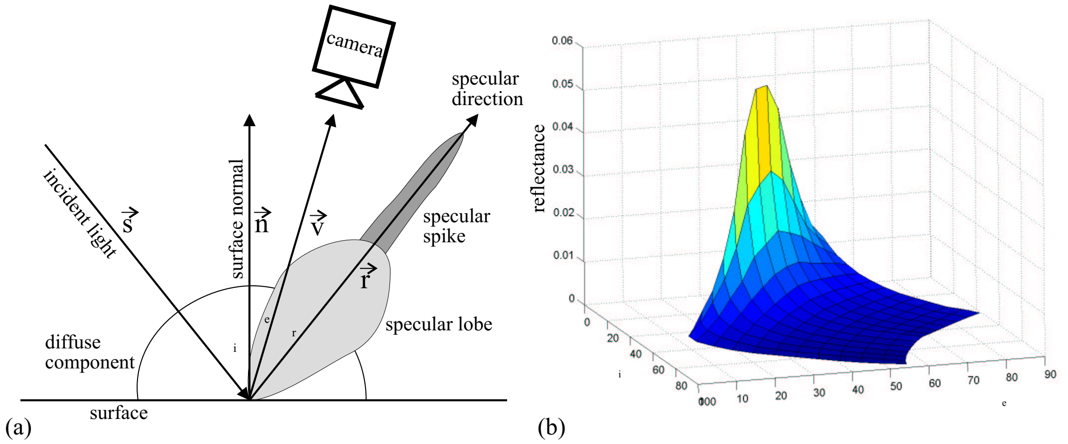


Figure 1: (a) Plot of the three reflectance components. (b) Measured reflectance of a raw forged iron surface for $\alpha = 75^\circ$.

using the linear method described by Rahmann (1999). The filter orientation Φ for which the maximum intensity $I_c + I_v$ is observed corresponds to the polarisation angle. The polarisation degree amounts to $D = I_v/I_c$. In principle, three measurements would be sufficient to determine the three parameters I_c, I_v, and Φ, but the fit becomes less noise-sensitive and thus more accurate when more measurements are used. The parameter I_c corresponds to the intensity reflectance R of the surface.

According to Nayar et al. (1991), the reflectance of a typical rough metallic surface consists of three components: a diffuse (Lambertian) component, the specular lobe, and the specular spike. We model these components by the phenomenological approach

$$R(\theta_i, \theta_e, \alpha) = \rho \left[\cos\theta_i + \sum_{n=1}^{N} \sigma_n \cdot (\cos\theta_r)^{m_n} \right] \qquad (11)$$

with $\cos\theta_r = 2\cos\theta_i\cos\theta_e - \cos\alpha$ describing the angle between the specular direction $\vec{r}$ and the viewing direction $\vec{v}$ (cf. Fig. 1a). For $\theta_r > 90^\circ$ only the diffuse component proportional to $\cos\theta_i$ is considered. The albedo ρ is assumed to be constant over the image. The shapes of the two specular components are expressed by $N = 2$ terms proportional to powers of $\cos\theta_r$, where the coefficients $\{\sigma_n\}$ denote the strength of the specular components relative to the diffuse component and the parameters $\{m_n\}$ their widths.

The polarisation angle Φ is phenomenologically modelled by an incomplete third-degree polynomial in p and q according to

$$R_\Phi(p,q) = a_\Phi pq + b_\Phi q + c_\Phi p^2 q + d_\Phi q^3. \qquad (12)$$

Without loss of generality we assume illumination in the xz plane (zero y component of $\vec{s}$) and a view along the z axis ($\vec{v} = (0,0,1)^T$). Eq. (12) is antisymmetric in q, and $R_\Phi(p,q) = 0$ for $q = 0$, i. e. coplanar vectors $\vec{n}$, $\vec{s}$, and $\vec{v}$. These properties are required for geometrical symmetry reasons as long as an isotropic interaction between the incident light and the surface material can be assumed. The polarisation degree D is modelled by an incomplete second-degree polynomial in p and q according to

$$R_D(p,q) = a_D + b_D p + c_D p^2 + d_D q^2. \qquad (13)$$

For rough metallic surfaces, R_D is maximum near the direction of specular reflection. Symmetry in q is imposed to account for isotropic light-surface interaction.

3 INTEGRATION OF DEPTH INFORMATION

Since the obtained solution of SfS and, to a lesser extent, SfPR may be ambiguous as long as single images are regarded, integrating additional information into the surface reconstruction process improves the reconstruction result. For example, a sparse set of 3D points of the object surface can be reconstructed by stereo vision, laser triangulation, or shadow analysis. Previous approaches either merge the results of stereo and SfS (Cryer et al., 1995) or embed the SfS algorithm into stereo (Samaras et al., 2000) or structure from motion algorithms (Lim et al., 2005). For the examples in this paper, a stereo algorithm was used to extract sparse depth information.

3.1 Description of the employed stereo algorithm

A block matching stereo algorithm is used in this paper. We assume that the images are rectified to standard stereo geometry with epipolar lines parallel to the horizontal image axis. The proposed approach is not restricted to this choice since any other source of relative depth information can be used instead.

For each pixel i at position (u,v) in the left image, a corresponding point is searched along the epipolar line in the right image. We use the normalized cross correlation coefficient (**normxcorr**) as similarity measure. A square region of 7 by 7 pixels of the left image (L) is correlated with regions on the corresponding epipolar line in the right image (R) for all candidate disparities d, resulting in an array of correlation coefficients $c_i(d) = \textbf{normxcorr}(L_{u,v}, R_{u-d,v})$. The disparity with the maximum correlation coefficient $d_i = \text{argmax}_d c_i(d)$ is determined, and a parabola $P(d) = ad^2 + bd + e$ is fitted to the local neighbourhood of the maxima. The dispartiy d_i is estimated at subpixel accuracy according to $d_i = -b/(2a)$. Only fits with $c_i(d_i) > 0.9$ and $a_i < -0.1$ are used. This ensures that only well localised correspondences are considered for further processing. The coordinates of a point (u_i, v_i) in the left stereo camera coordinate system are then given by $Z_i = bf/d_i$, $X_i = u_i b/d_i$, and $Y_i = v_i b/d_i$. The focal length f and base distance b between the cameras are determined by binocular camera calibration (Krüger et al., 2004).

3.2 Fusion of sparse depth information with SfPR

To incorporate the depth information into the global optimisation scheme presented in Section 2.1, we define a depth error term based on the depth difference between the sparse 3D points and the integrated gradient field. The depth difference between two 3D points i and j is given by

$$(\Delta z)^{ij} = Z^j - Z^i. \tag{14}$$

The corresponding depth difference of the reconstructed surface gradient field is calculated by integration along a path C^{ij} between the coordinates (u^j, v^j) and (u^i, v^i):

$$(\Delta z)^{ij}_{\mathrm{surf}} = \int_{C^{ij}} (pdx + qdy). \tag{15}$$

In our implementation the path C^{ij} is approximated by a list of K discrete pixel positions (u_k, v_k) with $k = 1, \ldots, K$. While in principle any path C^{ij} between the points i and j is possible, the shortest integration path, a straight line between i and j, is used here. Longer paths tend to produce larger depth difference errors because the gradient field is not guaranteed to be integrable.

Using these depth differences, it is possible to extend the global optimisation scheme introduced in Section 2.1 by adding an error term which minimises the squared distance between all N depth points:

$$e_z = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{\left((\Delta z)^{ij} - (\Delta z)^{ij}_{\mathrm{surf}}\right)^2}{\|(u_i, v_i) - (u_j, v_j)\|_2} \tag{16}$$

The iterative update rule Eq. (9) then becomes

$$\begin{aligned} p_{n+1}(u,v) = \bar{p}_n(u,v) + \lambda \frac{\partial e_I}{\partial p} + \mu \frac{\partial e_\Phi}{\partial p} + \nu \frac{\partial e_D}{\partial p} \\ + 2\chi \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left[\frac{(\Delta z)^{ij} - (\Delta z)^{ij}_{surf}}{\|(u_i, v_i) - (u_j, v_j)\|_2} \right] \left. \frac{\partial (\Delta z)^{ij}_{\mathrm{surf}}}{\partial p} \right|_{u,v}. \end{aligned} \tag{17}$$

An analogous expression is obtained for q. The derivatives of $(\Delta z)^{ij}_{\mathrm{surf}}$ with respect to p and q may only be nonzero if the pixel (u_k, v_k) belongs to the path C^{ij} and are zero otherwise. They are computed based on the discrete gradient field. The derivative depends on the direction (d_u, d_v) of the integration path at pixel location (u_k, v_k) with $d_u = u_{k+1} - u_k$ and $d_v = v_{k+1} - v_k$:

$$\begin{aligned} \left. \frac{\partial (\Delta z)^{ij}_{\mathrm{surf}}}{\partial p} \right|_{u_k, v_k} &= d_u p(u_k, v_k) \\ \left. \frac{\partial (\Delta z)^{ij}_{\mathrm{surf}}}{\partial q} \right|_{u_k, v_k} &= d_v q(u_k, v_k) \end{aligned} \tag{18}$$

The update of the surface gradient at location (u, v) is then normalised with the number of paths to which the corresponding pixel belongs. Error term (16) will lead to the evaluation of $N(N-1)/2$ lines at each update step and becomes prohibitively expensive for a large number of depth measurements. Therefore only a limited number of randomly chosen lines is used during each update step.

An earlier approach by Wöhler and Hafezi (2005) fuses SfS and shadow analysis using a similar depth difference error term. It is, however, restricted to depth differences along the light source direction. In contrast to the method by Fassold et al. (2004), which directly imposes depth constraints selectively on the sparse set of surface locations with known depth, our approach establishes large-scale surface gradients by computing differences between depth points. Effectively, our method transforms sparse depth data into dense depth difference data as long as a sufficiently large number of paths C^{ij} is taken into account. The influence of the depth error term is thus extended to a large number of pixels.

4 EXPERIMENTAL EVALUATION

To examine the accuracy of 3D reconstruction using the techniques described in Section 3.2, we apply them to synthetically generated surfaces in Section 4.1. In Section 4.2 we regard real-world scenarios of 3D surface reconstruction of metallic surfaces in the domain of industrial quality inspection.

4.1 Synthetic examples

To examine the behaviour of the global optimisation scheme described in Section 3.2, we apply the developed algorithms to the synthetically generated surface shown in Fig. 2a. We assume a perpendicular view on the surface along the z axis, corresponding to $\vec{v} = (0, 0, 1)^T$. The scene is illuminated by a single light source from the positive x direction under an angle of 15° with respect to the horizontal plane. This setting results in a phase angle $\alpha = 75^\circ$. A set of 100 random points has been extracted from the ground truth and is used as depth data Z for the reconstruction.

The reflectance functions of the rough metallic surface measured according to Section 2.2 were used to render the synthetic images shown in Fig. 2c. The reconstruction was performed with synthetic noisy data, where we used Gaussian noise with a standard deviation of 0.001 for I (maximum grey value ~ 0.06), 1° for Φ and 1 pixel for the depth values. Only intensity I, polarisation angle Φ and depth Z have been used during the reconstruction. In the case of rough metallic surfaces, the polarisation degree D contains similar information as the intensity I, has a higher measurement error, and is strongly affected by small-scale variations of the surface roughness (d'Angelo and Wöhler, 2005b), and is therefore not used for reconstruction.

The weights for the error terms according to Eq. (17) were set to $\lambda = 50$ (I in arbitrary units, with a maximum of ~ 0.06), $\mu = 14$ (Φ in radian), $\nu = 0$, and $\chi = 0.5$ (z between 0 and 50 pixels). The surface gradients were initialised with zero values. The 3D reconstruction results obtained with various combinations of error terms are shown in Fig. 2d-f. The reconstruction errors are listed in Table 4.1. It is apparent that the shape from shading reconstruction fails to reconstruct the surface (Fig. 2d), while the surface shape can be reconstructed approximately using intensity and polarisation degree (Fig. 2e). The combined approach (Fig. 2f) shows the smallest error. Table 4.1 also indicates that using intensity, polarisation, and depth leads to better results than either feature alone.

4.2 Real-world example: raw forged iron surface

We have applied our surface reconstruction algorithm to a raw forged iron surface. For the stereo reconstruction of the surface (cf. Section 3.1), we used a vergent stereo setup of two cameras (1032 × 776 pixels image size, 10° horizontal field of view, 320 mm base distance, average object distance 450 mm). Stereo calibration and image rectification to standard epipolar geometry were performed using the method by Krüger et al. (2004). The disparity values at object distance thus amount to approximately 4000 pixels. Experiments with synthetic data have shown that the standard deviation of the disparity is 0.3 pixels, resulting in an estimated standard error of 30 μm of the determined depth values. One of the stereo cameras is equipped with a rotating linear polarisation filter and is used to acquire the images required for SfPR (cf. Section 2.2). Fig. 3a shows the intensity and polarisation angle image, and Fig. 3b shows the triangulated stereo reconstruction result. The stereo reconstruction is very sparse due to the highly non-Lambertian metallic surface and does not extend across the complete surface to be reconstructed.

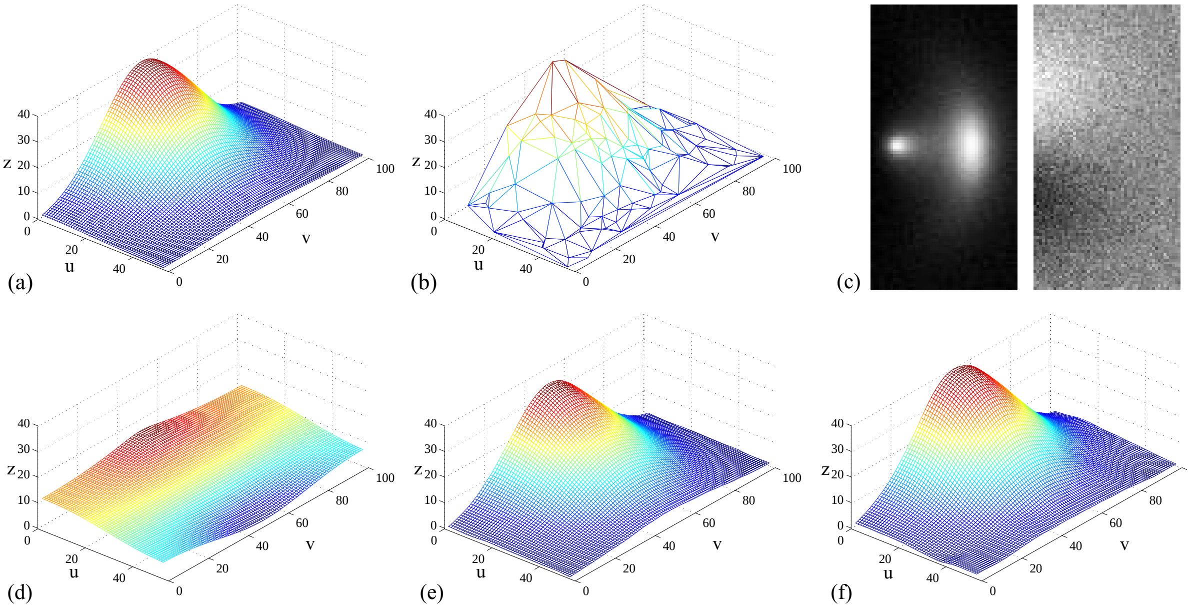


Figure 2: 3D reconstruction of a synthetically generated surface based on a photopolarimetric image and sparse depth values. (a) Ground truth. (b) Noisy 3D data. (c) From the left: noisy intensity and polarisation angle images, based on measured reflectance functions of a raw forged iron surface. The reconstruction result for noisy images of a surface with uniform albedo is shown in (d) using intensity only and in (e) using intensity and polarisation angle. (f) Reconstruction result obtained using the combined SfPR and depth approach.

Table 1: Results on the synthetic ground truth example shown in Fig. 2.

Method	RMS error (without noise)			RMS error (with noise)		
	z [pixels]	p	q	z [pixels]	p	q
I	8.19	0.267	0.508	8.19	0.267	0.508
I, Φ	2.07	0.186	0.039	2.12	0.189	0.058
Z	1.20	0.137	0.102	1.16	0.135	0.136
I, Z	0.80	0.070	0.076	0.79	0.083	0.115
I, Φ, Z	0.46	0.050	0.026	0.50	0.075	0.063

The unknown albedo ρ was computed based on the specular reflections, which appear as regions of maximum intensity I_{spec} and for which we have $\theta_r = 0°$ and $\theta_i = \alpha/2$. Eq. (11) then directly yields the albedo ρ. The reconstructed surface shown in Fig. 3c was computed using $\lambda = 50$, $\mu = 8$, $\nu = 0$, and $\chi = 1$ as error term weights, cf. Eq. (17). A cross-section of the surface was measured with a scanning laser focus profilometer and compared to the corresponding cross-section extracted from the reconstructed 3D profile (Fig. 3d). Although the triangulated depth data RMSE of 80 μm along the inspected profile of 14 mm length is already quite low, no small-scale detail of the surface is revealed. When all available photopolarimetric and depth information is used, the RMSE amounts to 62 μm. Without depth information the SfPR method yields a RMSE of 65 μm, while intensity information alone results in a much higher RMSE of 300 μm. If no polarimetric information is available (e.g. when incident light is not polarised by reflection at the surface), the combination between intensity and sparse depth data yields a RMSE of 70 μm.

5 SUMMARY AND CONCLUSION

In this paper we have presented an image-based method for 3D surface reconstruction relying on the simultaneous evaluation of reflectance, polarisation, and sparse depth data. The reflectance and polarisation properties of the surface material have been obtained by means of a series of images acquired through a linear polarisation filter under different orientations. SfPR and depth difference error terms are minimised using a variational approach, resulting in a surface gradient field. A dense depth map is obtained by numerical integration of the gradient field. A multi-scale approach has been used to improve the convergence behaviour. The proposed method transforms sparse depth data into dense depth difference data. In contrast to previous methods, the influence of the corresponding error term does not remain restricted to a small number of pixels. The presented method has been evaluated based on a synthetically generated surface, and a high accuracy of surface reconstruction has been demonstrated. Furthermore, we have successfully applied our method to the difficult real-world scenario of 3D reconstruction of a surface section of a raw forged iron part, yielding a very reasonable accuracy of 62 μm along the inspected profile of 14 mm length. A somewhat lower accuracy of 70 μm is obtained when polarisation information is neglected. These measurement errors are of the same order of magnitude as the intrinsic roughness of the metallic surface. We conclude that the suggested approach is a favourable technique for industrial surface inspection systems.

REFERENCES

Batlle, J., Mouaddib, E., Salvi, J., 1998. Recent progress in coded structured light as a technique to solve the correspondence problem: a survey. *Pattern Recognition*, 31(7), pp. 963-982.

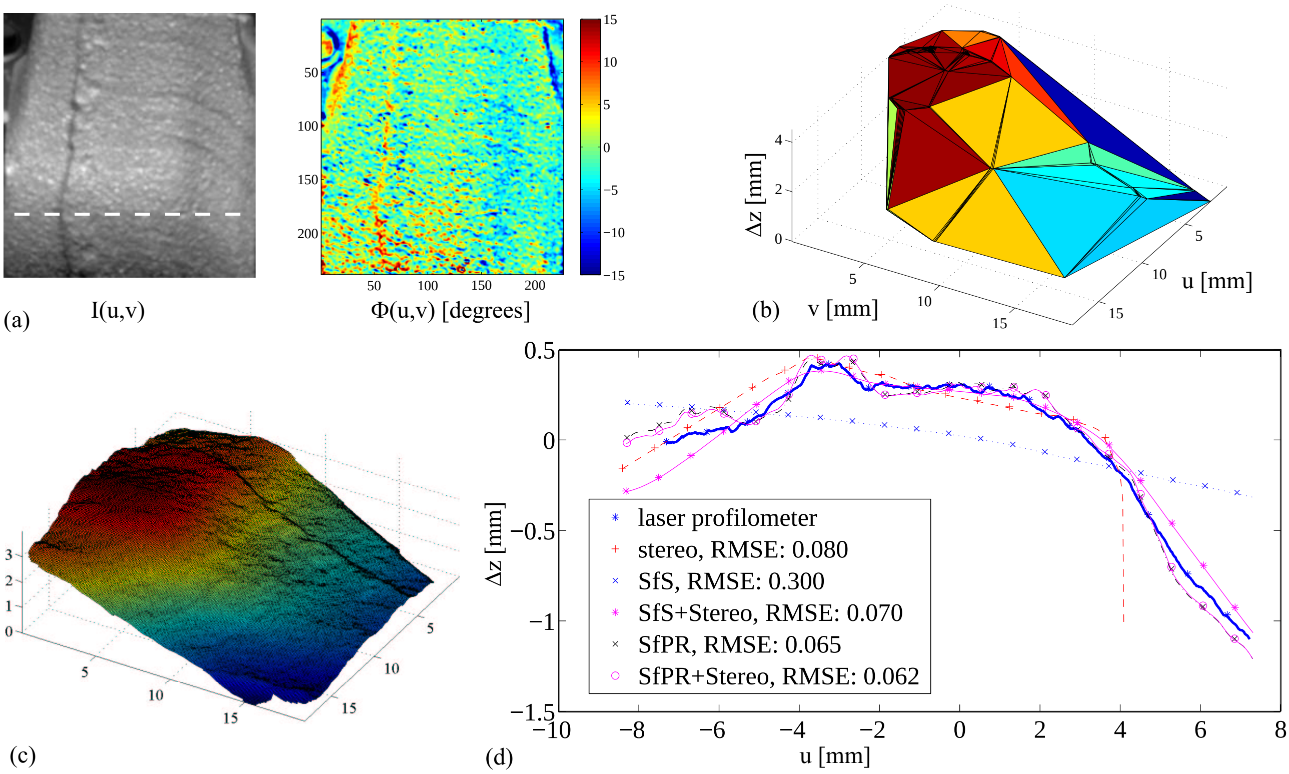


Figure 3: 3D reconstruction of a raw forged iron surface. (a) Reflectance and polarisation angle images. The size of the region of interest is 240 × 240 pixels. (b) Triangulated stereo reconstruction result. (c) Reconstruction by combined stereo and SfPR. (d) Comparison of the cross-section indicated by the dashed line in (a) to the ground truth measurement obtained with a laser focus profilometer.

Calow R., Gademann G., Krell G., Mecke R., Michaelis B., Riefenstahl N., Walke M., 2002. Photogrammetric measurement of patients in radiotherapy. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(5-6), pp. 347-359.

Cryer, J.E., Tsai, P.-S., Shah, M., 1995. Integration of shape from shading and stereo. *Pattern Recognition*, 28(7), pp. 1033-1043.

D'Angelo, P., Wöhler, C., 2005a. 3D Reconstruction of Metallic Surfaces by Photopolarimetric Analysis. In: H. Kalviainen et al. (Eds.), *Proc. 14th Scand. Conf. on Image Analysis*, LNCS 3540, Springer-Verlag Berlin Heidelberg, pp. 689-698.

D'Angelo, P., Wöhler, C., 2005b. 3D surface reconstruction based on combined analysis of reflectance and polarisation properties: a local approach. *ISPRS Workshop Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images*, Beijing, China.

Fassold, H., Danzl, R., Schindler, K., Bischof, H. 2004. Reconstruction of Archaeological Finds using Shape from Stereo and Shape from Shading. *9th Computer Vision Winter Workshop*, Piran, Slovenia, pp. 21-30.

Faugeras, O., 1993. *Three-Dimensional Computer Vision*. MIT Press, Cambridge, Massachusetts.

Horn, B. K. P., Brooks, M. J., 1989. *Shape from Shading*. MIT Press, Cambridge, Massachusetts.

Horn, B. K. P., 1989. Height and Gradient from Shading. MIT technical report 1105A. http://people.csail.mit.edu/people/bkph/AIM/AIM-1105A-TEX.pdf

Jiang, X., Bunke, H., 1997. *Dreidimensionales Computersehen*. Springer-Verlag, Berlin.

Krüger, L., Wöhler, C., Würz-Wessel, A., Stein, F., 2004. Infactory calibration of multiocular camera systems. *SPIE Photonics Europe (Optical Metrology in Production Engineering)*, Strasbourg, pp. 126-137.

Lim, J., Jeffrey, H., Yang, M., Kriegman, D., 2005. Passive Photometric Stereo from Motion. *IEEE Int. Conf. on Computer Vision*, Beijing, China, vol. II, pp. 1635-1642.

Miyazaki, D., Tan, R. T., Hara, K., Ikeuchi, K., 2003. Polarization-based Inverse Rendering from a Single View. *IEEE Int. Conf. on Computer Vision*, Nice, France, vol. II, pp. 982-987.

Nayar, S. K., Ikeuchi, K., Kanade, T., 1991. Surface Reflection: Physical and Geometrical Perspectives. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(7), pp. 611-634.

Rahmann, S., 1999. Inferring 3D scene structure from a single polarization image. *Conf. on Polarization and Color Techniques in Industrial Inspection*, SPIE Vol. 3826, Munich, Germany, pp. 22-33.

Samaras, D., Metaxas, D., Fua, P., Leclerc, Y.G. Variable Albedo Surface Reconstruction from Stereo and Shape from Shading. *Proc. CVPR 2000*, vol I, pp. 480-487.

Simchony, T., Chellappa, R., Shao, M., 1991. Direct Analytic Methods for Solving Poisson Equations in Computer Vision Problems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(5), pp. 435-556.

Wöhler, C., Hafezi, K., 2005. A general framework for three-dimensional surface reconstruction by self-consistent fusion of shading and shadow features. *Pattern Recognition*, 38(7), pp. 965-983.

MCMC LINKED WITH IMPLICIT SHAPE MODELS AND PLANE SWEEPING FOR 3D BUILDING FACADE INTERPRETATION IN IMAGE SEQUENCES

Helmut Mayer and Sergiy Reznik

Institute of Photogrammetry and Cartography, Bundeswehr University Munich, D-85577 Neubiberg, Germany
{Helmut.Mayer|Sergiy.Reznik}@unibw.de

KEY WORDS: Markov Chain Monte Carlo, Implicit Shape Models, Plane Sweeping, Facade Interpretation

ABSTRACT:

In this paper we propose to link Markov Chain Monte Carlo – MCMC in the spirit of (Dick, Torr, and Cipolla, 2004) with information from Implicit Shape Models – ISM (Leibe and Schiele, 2004) and with Plane Sweeping (Werner and Zisserman, 2002) for the 3D interpretation of building facades, particularly for determining windows and their 3D extent. The approach starts with a (possibly uncalibrated) image sequence, from which the 3D structure and especially the vertical facades are determined. Windows are then detected via ISM. The main novelty of our work lies in using the information learned by the ISM also to delineate the window extent. Additionally, we determine the 3D position of the windows by plane sweeping in multiple images. Results show potentials and problems of the proposed approach.

1. INTRODUCTION

Recently, there are – among others – two yet not contradicting important directions in object extraction: Appearance based and generative models. Prominent examples for the former are, e.g., (Lowe, 2004) and (Agarwal, Awan, and Roth, 2004). The basic idea of these two and similar approaches is that an object is modeled by features computed from small characteristic image patches and their spatial arrangement, both being learned more or less automatically from given training data, i.e., images. While this can also be seen as a discriminative model where a hypothesis for an object is created bottom-up from the data, generative models go the other way, i.e., top-down: From a given hypothesis they generate a plausible instance of the data generatively, i.e., via computer graphics, and compare it with the given image data. Usually this is done in a Bayesian framework. There are priors for the parameters, the comparison with the data results into a likelihood, and both are combined into the posterior. One particularly impressive example for an approach linking discriminative and generative modeling tightly in a statistically sound manner is (Tu, Chen, Yuille, and Zhu, 2005). In (Fei-Fei, Fergus, and Perona, 2004) a generative appearance based model is employed to learn 101 object categories from only a few training examples for each class via incremental Bayesian learning.

We are aiming at the interpretation of building facades from image sequences, particularly inspired by the generative model based on Markov Chain Monte Carlo – MCMC, e.g., (Neal, 1993), put forward in (Dick, Torr, and Cipolla, 2004). To detect objects, in our case windows, we follow (Mayer and Reznik, 2005) who use appearance based modeling in the form of an Implicit Shape Model – ISM, as introduced by (Leibe and Schiele, 2004). Yet, and this is the main novelty of our approach, we additionally link ISM to MCMC for the determination of the window extent. By this means we partly avoid the tedious manual generation of a model for the in our case sometimes complex structures of windows and also robustify the approach. Additionally, we compute the three-dimensional (3D) extent of the windows by means of plane sweeping proposed in (Werner and Zisserman, 2002). Opposed to (Werner and Zisserman, 2002) as well as (Bauer, Karner, Schindler, Klaus, and Zach, 2003) we do not detect windows as objects which are situated behind the facade plane which makes us independent from the fact if the windows are behind, on, or in even in front of the facade.

The basic idea of the generative model of (Dick, Torr, and Cipolla, 2004), which is our main inspiration, is to construct the building from parts, such as the facades and the windows, for which parameters, e.g., the width, brightness, are changed statistically to produce an appearance resembling the images after perspectively projecting the model with the given parameters. The difference between the given and the generated image determines the likelihood that the data fits to the model and is combined with prior information describing typical characteristics of buildings.

Other work on facades is, e.g., (Früh and Zakhor, 2003), where a laser-scanner and a camera mounted on a car are employed to generate 3D models of facades (yet without information about objects such as windows or doors) and together with aerial images and aerial laser-scanner data realistic models of areas of cities. In photogrammetry as well as in computer vision semi-automatic approaches have been proposed (van den Heuvel, 2001; Wilczkowiak, Sturm, and Boyer, 2005), where the latter exploits special geometrical constraints of buildings for camera calibration. (Böhm, 2004) shows how to eliminate visual artifacts from facades by mapping images from different view points on the facade plane employing the robust median. The determination of fine 3D structure on facades via disparity estimation is presented by (von Hansen, Thönnessen, and Stilla, 2004). (Wang, Totaro, Taillandier, Hanson, and Teller, 2002) take into account the grid, i.e., row / column, structure of the windows on many facades. (Alegre and Dallaert, 2004) propose a more sophisticated approach, where a stochastic context-free grammar is employed to represent recursive regular structures of the windows. Both papers only give results for one or two very regular high-rising buildings.

In Section 2. we sketch our approach to generate a vertically oriented Euclidean 3D model consisting of cameras and points from (possibly uncalibrated) image sequences, from which we determine vertical facade planes. Section 3. describes the ISM and as main contribution of this paper how we learn and use the segmentation information to help delineate the windows via MCMC. Finally, in Section 4. we show how the 3D extent of the windows can be determined based on plane sweeping. The paper ends up with conclusions.

2. 3D RECONSTRUCTION

Our approach is based on wide-baseline image sequences. After projective reconstruction using fundamental matrices and trifocal tensors (Hartley and Zisserman, 2003) employing Random Sample Consensus – RANSAC (Fischler and Bolles, 1981) based on Förstner points (Förstner and Gülch, 1987) which we match via least squares matching, we calibrate the camera employing the approach proposed in (Pollefeys, Van Gool, Vergauwen, Verbiest, Cornelis, and Tops, 2004). If calibration information is available, we use (Nistér, 2004) to determine the Euclidean 3D structure for image pairs. Our approach deals efficiently with large images by using image pyramids and we obtain full covariance matrices for the projection matrices and the 3D points by means of bundle adjustment taking into account the covariance matrices of the least squares matching of all employed images.

Having generated a 3D Euclidean model we orient it vertically based on the vertical vanishing point derived from the vertical lines on the facade and the given calibration parameters. The vertical vanishing point is detected robustly again using RANSAC, the user only providing the information if the camera has been been very approximately held horizontally or vertically.

The vertically oriented model is the basis for the determination of the facade planes using once again RANSAC. To make the determination more robust and precise, we employ the covariance information of the 3D points from the bundle adjustment by testing the distances to a hypothesized plane based on the geometric robust information criterion – GRIC (Torr, 1997). Additionally, we check, if the planes are vertical and we allow only a limited overlap of about five percent between the planes. The latter is needed, because of the points possibly situated on intersection lines between the planes.

Finally, as the position of the facade planes is often determined in-between the plane defined by the real facade and the plane defined by the windows, its depth is optimized via plane sweeping (Baillard and Zisserman, 1999; Werner and Zisserman, 2002). From the parameters for the facade planes as well as the projection matrices we compute homographies between the plane and the images. We project all images a facade can be seen from (this can be derived via the points that lead to the plane and from which images they were determined) onto the facade plane and compute an average image as well as the bias in brightness for each projected image to it. Then, we move the facade plane in its normal direction and determine for a larger number of distances the squared differences of gray values to the average image for all images after subtracting the bias in brightness determined above. We finally take the position, where this difference is minimum.

The result of this step are projection matrices, 3D points, and optimized facade planes all in a vertically oriented Euclidean system. The only additional information the user has to provide for the further processing is the approximate scaling of the model so that the images can be projected on the facade with a normalized pixel-size. Therefore, for the next step of the delineation of windows on the facade we can assume vertically oriented facade planes with a standardized pixel size.

3. DETECTION AND DELINEATION OF WINDOWS BASED ON MCMC AND ISM

An Implicit Shape Model – ISM (Leibe and Schiele, 2004) describes an object in the form of the spatial arrangement of characteristic parts. As (Agarwal, Awan, and Roth, 2004) we use as parts image patches (here of the empirically determined size 9×9 pixels) around Förstner points. Training patches and patches in an image to be analyzed are compared via the (normalized) cross correlation coefficient (CCC). For the arrangement of the points we employ as (Leibe and Schiele, 2004) the generalized Hough transform.

Similarly as (Mayer and Reznik, 2005), we "learn" the model for a window in a way that can be seen as a simplified version of (Leibe and Schiele, 2004): We manually cut out image parts containing training windows using in the range of about 100 windows. In these (cf. Figure 1 for an example) we extract Förstner points with a fixed set of parameters. Opposed to (Mayer and Reznik, 2005), we manually mark the extent of the whole window including the frame and compute from it the center.

We "learn" only salient points at the corners of the manually marked window extent (small yellow squares in Figure 1). For these we store the gray values in the patches around the points, their relation to the window center in the form of the difference vector, and particularly their relation to the window extent. This is done in the form of images of the edges of the window extent. The latter gives information which we use for the segmentation, i.e., the delineation of the window, the main novelty of our approach. Figure 2 shows examples for image patches (left) together with the edges derived from the manually given window extent (right). Please note that for many of our (training) windows the window extent does not fit too well to the Förstner points as they tend to be situated at the salient image corner between glass and window frame.

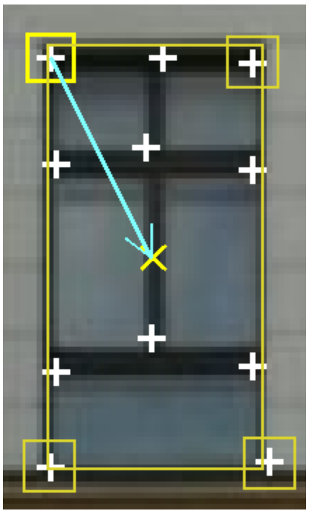


Figure 1. Image part containing training window with Förstner points (white crosses), manually marked window extent (yellow rectangle), window center (yellow diagonal cross), patches around salient points at the corners of the window extent (small yellow squares), and one of four difference vectors to center (blue arrow)

To detect windows on a facade, we extract Förstner points with the same set of parameters as above (cf., e.g., Figure 3, left) and compare the patches of size 9×9 centered at them with all salient points learned above by means of CCC. If CCC is above an empirically determined threshold of 0.9, we write out the difference vector learned for the corresponding point into an initially empty evidence image, incrementing the corresponding pixel by one. By this means, each match votes for the position of the window center. The Förstner points as well as the evidence for the position of the window centers are given for our running example in Figure 3, right.

Figure 3, right, shows that the hypothesized window centers are widely spread, because parts of windows can vote for different

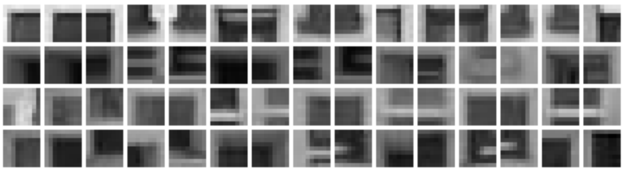
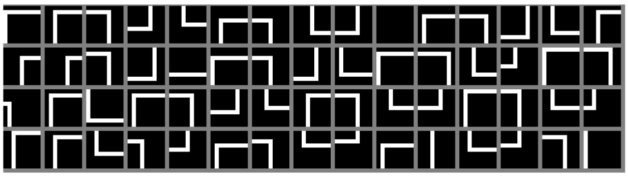



Figure 2. Set of patches (left) and set of edges (right) for window corners

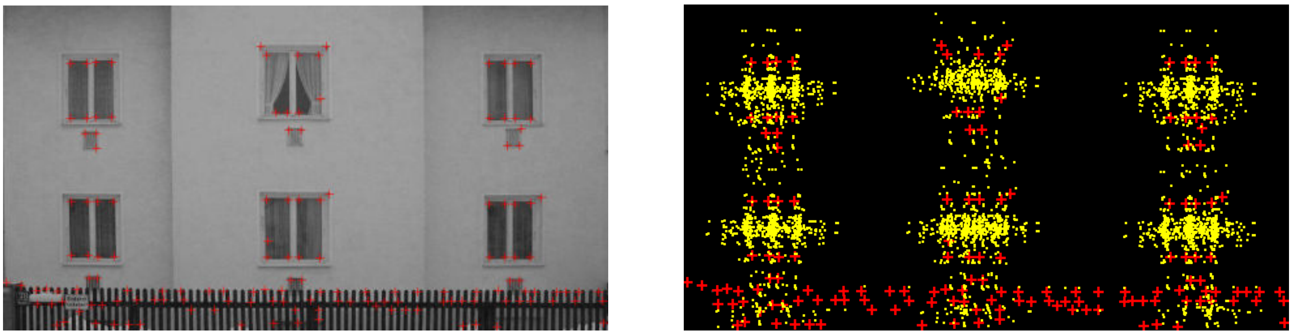


Figure 3. Facade (left) and evidence for window centers (yellow dots, right) both with Förstner points (red crosses)

positions. A patch can look, e.g., similar to an upper right corner of a whole window, but is actually situated at a transom (horizontal bar) at the center of the window. To generate meaningful hypotheses for window centers, we, therefore, integrate the evidence by smoothing them with a Gaussian and then determine all local maxima above a given threshold. The result for this is shown in Figure 5, left. Please note that none of the windows used for training stems from this scene as well as any of our examples presented in this paper.

The information from the ISM is used for segmentation by inserting it into the generative modeling based on MCMC. For this, the patches voting for the respective centers need to be determined. In Figure 5, right, all hypotheses and their difference vectors for the areas around the local maxima for the window centers, where the evidence is beyond 0.9 of the local maximum value, are shown. From these vectors only the vectors pointing diagonally are retained. Only they provide information about the window extent, because windows are assumed not to be extremely narrow or low. The average vectors of these patches pointing to the center are shown in Figure 4 together with the areas where the evidence is locally above 0.9 of its maximum.

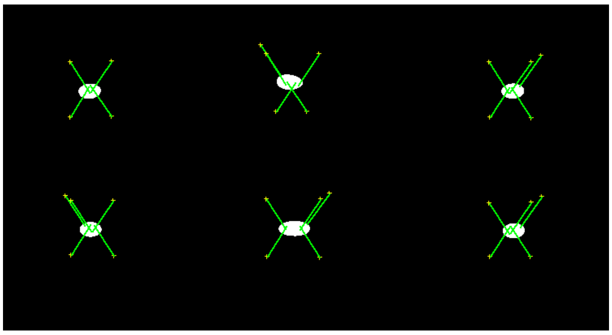


Figure 4. Areas with a value beyond 0.9 times of the local maxima (white) and average vectors of all hypotheses for corners pointing diagonally to the maxima, i.e., hypotheses for the window centers (green lines)

Once the potential patches at the window corners are known, the corresponding edges (cf. Figure 2, right) are summed up (cf. Figure 6, left). For guiding MCMC, the edges are thinned, normalized and then blurred to extend the area of convergence. As the likelihood is normalized in the MCMC process, it is important that the ends of the straight segments are cut and not blurred in the direction of the edge. The result is the window corner image (cf. Figure 6, right).

To delineate the windows, we start with hypotheses constructed from the centers of the diagonally most distant patches voting for a particular window and a small inward offset of 8 pixels in horizontal and vertical direction to avoid that the random search starts outside the window extent. We then take up the basic idea of (Dick, Torr, and Cipolla, 2004), i.e., we try to generate an image which is similar to the actual image. Our basic model is very simple, namely a rectangle brighter or darker than the background, i.e., with an edge to the background. The corresponding edges for the windows are projected into the window corner image and the normalized strength of all pixels above zero gives the likelihood. As we found that for bright facades it is very helpful that windows are in most cases darker than the facade plane, we follow for them (Mayer and Reznik, 2005) and correlate a model consisting of noisy dark rectangles on a bright background with the facade image abstracted by gray-scale morphology. The result for this is then combined with the result based on ISM on a half and half basis.

Figure 7, left, shows a hypothesis for the window extent, i.e., the start position, and right the final position. Please note that we have employed the half and half combination of correlation and ISM for the running example with its bright facade. Therefore, the final position in Figure 7, right, does not fit perfectly to the distribution given by the ISM.

The parameters for the window extent are disturbed by Gaussian noise taking into account the prior that the ratio of height to width of a window lies in the majority of cases between 0.25 to 5 modeled by a mixture of Gaussians. For each iteration of MCMC, we either change the width, the height, or the position of the rectangle representing the window extent. For robustification we use simulated annealing. I.e., the higher the number of iteration becomes, the lower becomes the probability to accept results which

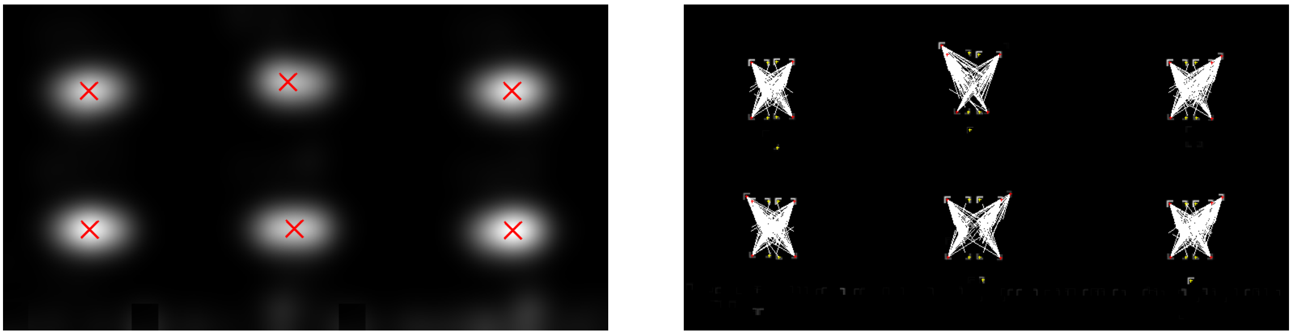


Figure 5. Evidence for window centers integrated with Gaussian together with maxima (diagonal red crosses – left) and hypotheses for window corners pointing to the maxima (right)

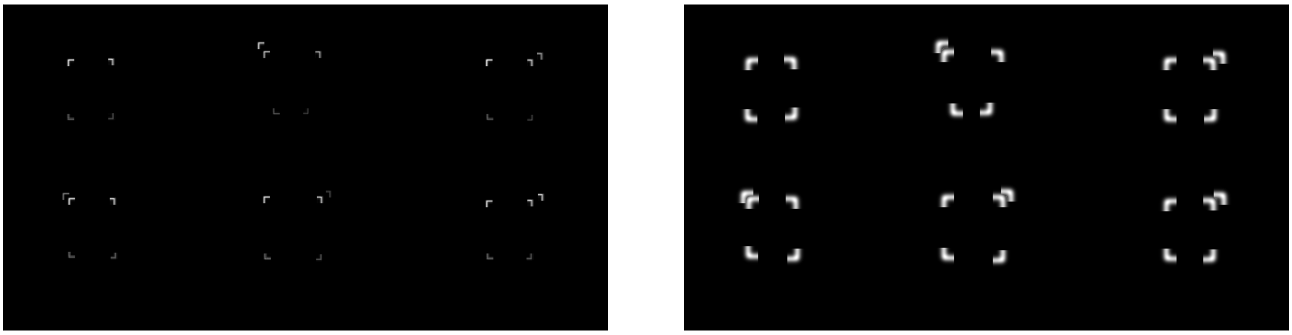


Figure 6. Sum of edges describing window corners (left) and derived distribution to guide MCMC (window corner image, right)

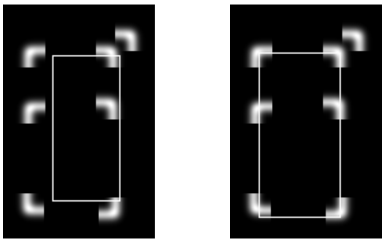


Figure 7. Distribution from ISM used to guide MCMC with hypothesis for window extent, i.e., start position (left) and final position (right).

are worse than for the preceding iteration. Figure 8 shows the hypotheses in white and the final result in green.

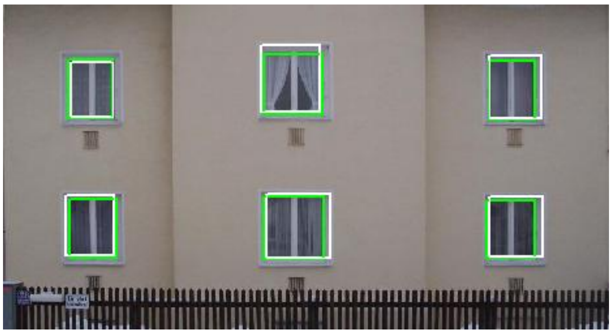


Figure 8. Hypotheses for the window extent (white) and final outcome (green)

4. DETERMINATION OF THE 3D EXTENT OF WINDOWS VIA PLANE SWEEPING

As windows are often not lying on the facade plane, but mostly behind it, their 3D position needs to be determined. This is done again by means of plane sweeping, cf. Section 2., employing the 3D Euclidean reconstruction result by computing homographies between planes and images. The bias in brightness of the images to an average image is computed for the whole facades as it is too unreliable for the individual windows. To determine the depth of a particular window, we move the rectangular part of the facade plane determined above to correspond to a window in the direction of the normal of the facade plane. We compute for a larger number of reasonable distances from the facade plane the squared differences of gray values from the individual images it can be seen from to the average image and take the position, where the difference is minimum.

Results for this are given in Figures 10 and 13. The first result, the input images for which are given in Figure 9, shows that we are actually dealing with a 3D setup where not only images of facade planes, but also there 3D position and relations to the cameras are known. For this bright facade again ISM and correlation have been used on a half by half basis leading to a meaningful delineation of the windows after detecting all windows on the facade. Also plane sweeping was successful for all windows as can be seen from the nearly constant offset. With our approach we are able to determine different depths for individual windows as we do not employ 3D information in the form of local maxima of the whole plane to determine possible window hypotheses such as (Werner and Zisserman, 2002). Yet, we have to note that a combination of both ideas might be the best way to proceed to deal with more complex situations.

For the second building in Figure 13 (input images cf. Figure 12,

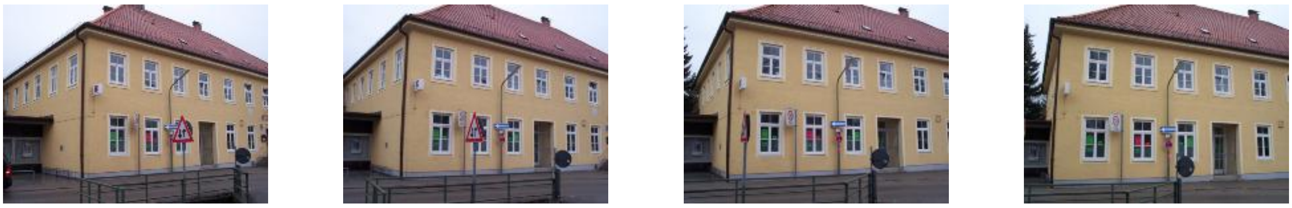


Figure 9. Four images used to generate the model given in Figure 10

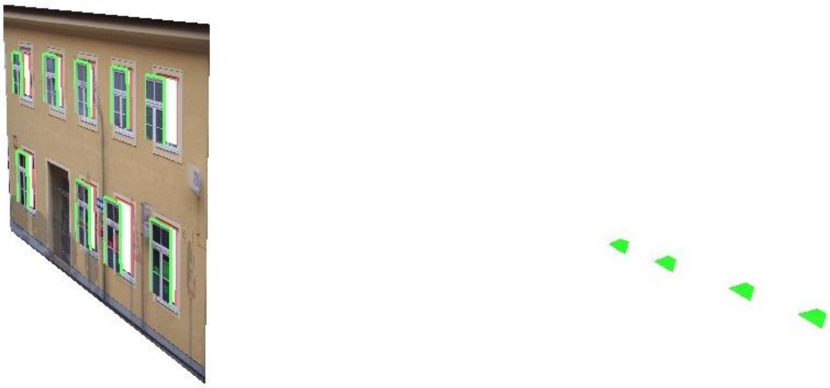


Figure 10. Bright building seen from the back – the windows are marked in red on the facade and in green behind the facade; cameras are given as green pyramids with the tip defining the projection center and the base the viewing direction

3D points and cameras, cf. Figure 11) the facades are rather dark. Therefore, we could only use ISM for the delineation of the windows. One can see from Figure 13, left, that for it all windows have been detected, except for the upper left, where the resolution of the image is not good and which is disturbed by a bird house. As we have not yet modeled doors, the door on the right facade is interpreted as a window. Figure 13, right, shows that in most cases there was a correct and consistent determination of the depth of the windows. Here one has to note, that these are mostly windows without mullions and transoms, where a determination of the depth is rather difficult, also because the windows are partly reflecting the surroundings.

5. CONCLUSIONS

We have shown how by combining appearance based and generative modeling employing MCMC and ISM the extent of objects, particularly windows, can be determined robustly based on automatically learned models even if the structure of the object varies or the contrast is weak. This can be seen as an extension of approaches such as (Dick, Torr, and Cipolla, 2004), where a less adaptive object-wise modeling of the texture was employed. We have also demonstrated how based on plane sweeping employing homographies between the facade plane and the images it is possible to determine the 3D position of the planar hypotheses for the windows.

Windows, but also other objects on the facade, can have substructures of different sizes, e.g., mullions and transoms. To model them and also other objects such as doors and architectural details, we plan to integrate scale into ISM.

The homographies employed in the 3D determination could on one hand help to identify 3D details not lying on the facade, but could also be used to compute the 3D position of (partly) planar objects far off, but parallel to the facade plane such as balconies. For handling problems with different reflectivity we plan to introduce a robust estimator.

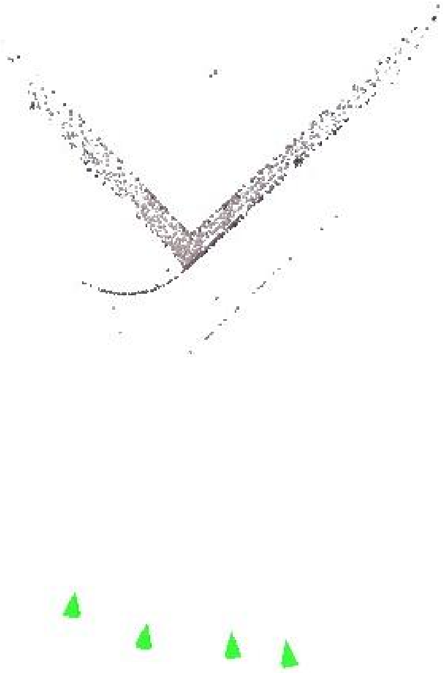


Figure 11. 3D points and cameras (green pyramids) for dark building

To be able to model rows or columns of windows or architectural details and grids made up of them, it is essential that one can deal with models of changing complexity. A means for this is Reversible Jump Markov Chain Monte Carlo – RJMCMC (Green, 1995), used, e.g., by (Dick, Torr, and Cipolla, 2004). It allows to change the number of objects during processing, i.e., to include new windows, etc. To model rows, columns, and grids in a principled way we want to employ a (context free) stochastical grammar describing the hierarchy of the objects on the facade as well as of the different facades of a building in the spirit of (Alegre and Dallaert, 2004).

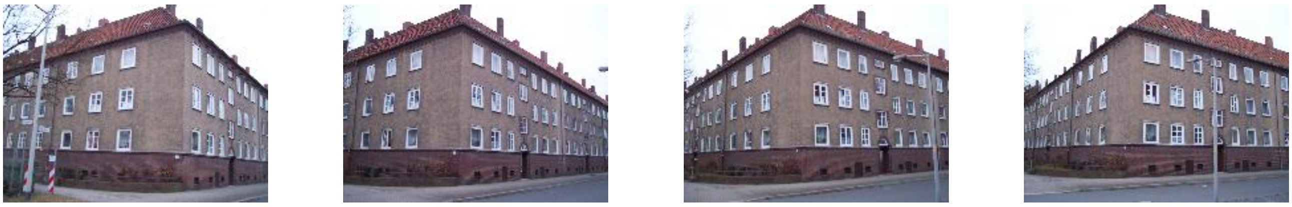


Figure 12. Four images used to generate the model given in Figures 13 and 11

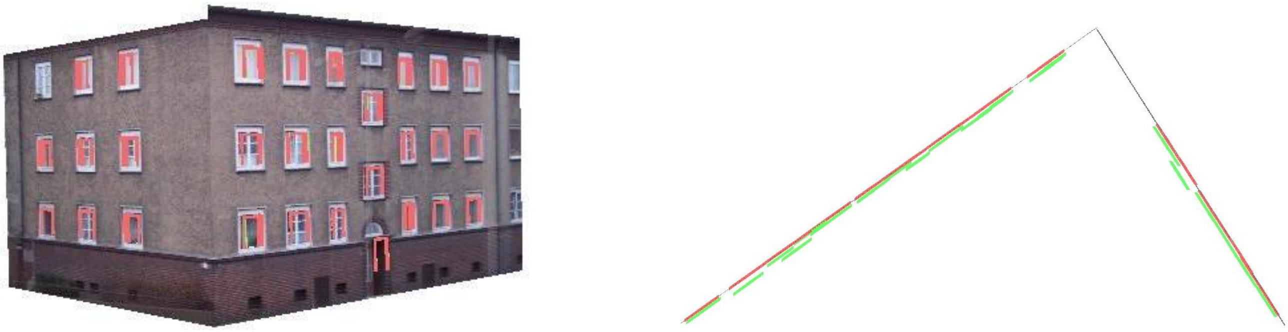


Figure 13. Dark building seen from the outside (left) and from the top (right) – colors and cameras cf. Figure 10

ACKNOWLEDGMENTS

Sergiy Reznik is funded by Deutsche Forschungsgemeinschaft under grant MA 1651/10. We thank the anonymous reviewers for their helpful comments.

REFERENCES

Agarwal, S., Awan, A., and Roth, D., 2004. Learning to Detect Objects in Images via a Sparse, Part-Based Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(11), 1475–1490.

Alegre, F. and Dallaert, F., 2004. A Probalistic Approach to the Semantic Interpretation of Building Facades. In *International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres*, pp. 1–12.

Baillard, C. and Zisserman, A., 1999. Automatic Reconstruction of Piecewise Planar Models from Multiple Views. In *Computer Vision and Pattern Recognition*, Volume II, pp. 559–565.

Bauer, J., Karner, K., Schindler, K., Klaus, A., and Zach, C., 2003. Segmentation of Building Models from Dense 3D Point-Clouds. In *27th Workshop of the Austrian Association for Pattern Recognition*.

Böhm, J., 2004. Multi Image Fusion for Occlusion-Free Façade Texturing. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume (35) B5, pp. 867–872.

Dick, A., Torr, P., and Cipolla, R., 2004. Modelling and Interpretation of Architecture from Several Images. *International Journal of Computer Vision* 60(2), 111–134.

Fei-Fei, L., Fergus, R., and Perona, P., 2004. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In *IEEE Workshop on Generative-Model Based Vision*.

Fischler, M. and Bolles, R., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM* 24(6), 381–395.

Förstner, W. and Gülch, E., 1987. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features. In *ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, Interlaken, Switzerland, pp. 281–305.

Früh, C. and Zakhor, A., 2003. Constructing 3D City Models by Merging Aerial and Ground Views. *IEEE Computer Graphics and Applications* 23(6), 52–61.

Green, P., 1995. Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. *Biometrika* 82, 711–732.

Hartley, R. and Zisserman, A., 2003. *Multiple View Geometry in Computer Vision – Second Edition*. Cambridge, UK: Cambridge University Press.

Leibe, B. and Schiele, B., 2004. Scale-Invariant Object Categorization Using a Scale-Adaptive Mean-Shift Search. In *Pattern Recognition – DAGM 2004*, Berlin, Germany, pp. 145–153. Springer-Verlag.

Lowe, D., 2004. Distintive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60(2), 91–110.

Mayer, H. and Reznik, S., 2005. Building Façade Interpretation from Image Sequences. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume (36) 3/W24, pp. 55–60.

Neal, R., 1993. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto.

Nistér, D., 2004. An Efficient Solution to the Five-Point Relative Pose Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(6), 756–770.

Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., and Tops, J., 2004. Visual Modeling with a Hand-Held Camera. *International Journal of Computer Vision* 59(3), 207–232.

Torr, P., 1997. An Assessment of Information Criteria for Motion Model Selection. In *Computer Vision and Pattern Recognition*, pp. 47–53.

Tu, Z., Chen, X., Yuille, A., and Zhu, S.-C., 2005. Image Parsing: Unifying Segmentation Detection and Recognition. *International Journal of Computer Vision* 63(2), 113–140.

van den Heuvel, F. A., 2001. Object Reconstruction from a Single Architectural Image Taken with an Uncalibrated Camera. *Photogrammetrie – Fernerkundung – Geoinformation* 4/01, 247–260.

von Hansen, W., Thönnessen, U., and Stilla, U., 2004. Detailed Relief Modeling of Building Facades From Video Sequences. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume (35) B3, pp. 967–972.

Wang, X., Totaro, S., Taillandier, F., Hanson, A., and Teller, S., 2002. Recovering Facade Texture and Microstructure from Real-World Images. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume (34) 3A, pp. 381–386.

Werner, T. and Zisserman, A., 2002. New Techniques for Automated Architectural Reconstruction from Photographs. In *Seventh European Conference on Computer Vision*, Volume II, pp. 541–555.

Wilczkowiak, M., Sturm, P., and Boyer, E., 2005. Using Geometric Constraints through Parallelepipeds for Calibration and 3D Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27(2), 194–207.

A SUPERVISED CLASSIFICATION APPROACH TOWARDS QUALITY SELF-DIAGNOSIS OF 3D BUILDING MODELS USING DIGITAL AERIAL IMAGERY

Laurence Boudet[1] Nicolas Paparoditis[1] Franck Jung[2] Gilles Martinoty[1] Marc Pierrot-Deseilligny[1]

[1] Institut Géographique National - Laboratoire MATIS. 2-4 avenue Pasteur, 94165 Saint Mandé
[2] Ecole Supérieure des Géomètres et Topographes. 1, boulevard Pythagore - Campus Universitaire, 72000 Le Mans

{laurence.boudet; nicolas.paparoditis; gilles.martinoty; marc.pierrot-deseilligny}@ign.fr
franck.jung@esgt.cnam.fr

KEY WORDS: Quality Self-diagnosis, Consistency, Image-based measures, Performance evaluation, Classification, 3D City Models

ABSTRACT:

In the context of 3D building model production or updating, the models have to be manually checked one by one by a human operator in order to ensure their quality. In this paper, we investigate a new approach to perform a quality self-diagnosis of building models in dense urban areas from high resolution aerial images. Hence, we aim at reliably identifying roof facets that do not comply with quality specifications. The self-diagnosis process will highlight potential incorrect facets for their inspection by a human operator. A set of calibrated aerial images enable us to collect positive or negative evidences of roof facet existence and consistency. A particular attention has been paid to the definition of a set of low-level, complementary, robust and consistent image processing measures. Four quality classes have been defined and are used to classify roof facet quality. A supervised classifier and robust decision rules are then applied to perform an effective self-diagnosis according to the traffic light paradigm. Finally, the work in progress leads to a promising quantitative and qualitative evaluation in the context of dense urban areas.

1. INTRODUCTION

1.1 Motivation

Many applications use 3D building models, such as urban environment planning, telecommunications and natural disaster simulations. Automation of 3D building reconstruction from aerial images has been a very active field of research for the two last decades, leading to a large number of automated or semi-automated systems. Automated production of 3D building models is now conceivable over entire cities, especially when 2D building footprints are available, from cadastral maps for instance. Nevertheless, a verification stage is necessary to control the quality of produced data, including shape description correctness, topological consistency, geometrical accuracy and completeness. This quality control is now a key issue to a greater use and an easier maintenance of 3D building models, since it has been done manually so far.

In this paper, we focus on the quality self-diagnosis of individual roof facets, as a first step of the 3D building model assessment in the context of data production, update or verification. In order to produce useful information on this diagnosis, results should be presented according to the traffic light paradigm (Förstner, 1996). It is based on three qualitative identified classes, namely accepted (high quality verified facets), rejected (poor quality verified facets) and undecided (intermediate quality facets). Then, a verification stage completes the self-diagnosis process, in which a human operator only checks the undecided and rejected facets, in order to confirm, edit or delete them. The self-diagnosis process of 3D roof facet quality is based on aerial images and does not depend on the level of automation involved in the reconstruction stage (none, semi or complete) or on the specific algorithm used to produce the building models.

1.2 Related Work

Since intensive researches have been carried out on 3D building model reconstruction from aerial imagery, quantitative and qualitative evaluations have also been achieved (Henricsson and Baltsavias, 1997, Rottensteiner and Schulze, 2003, Durupt and Taillandier, 2006) using visual inspection and/or a high quality ground truth reference. Avoiding the reference need, (Schuster and Weidner, 2003, Meidow and Schuster, 2005) proposed to use another reconstructed scene to compute either absolute or relative quality measures. Quality criteria are based on completeness, robustness, geometric accuracy, and shape similarity according to the reference, in addition to those proposed in (McKeown et al., 2000). These empirical evaluations showed the capabilities of semi-automated and automated systems for the production of 3D building models.

Another approach of evaluation in computer vision is the algorithm performance characterisation in terms of internal evaluation and error propagation. (Förstner, 1994, Förstner, 1996, Thacker et al., 2005) give useful guidelines on this topic. Nevertheless, the presented self-diagnosis process aims at assessing data quality independently from the reconstruction techniques or algorithms. Thus, self-diagnosis is based on observations of the reality and requires the definition of image-based measures. Some examples can be found in the ”hypothesize and verify” approach of 3D model reconstruction, such as (Suveg and Vosselman, 2002, Jibrini et al., 2004, Taillandier and Deriche, 2004) where the best building model is selected among plausible ones, or such as (Kim and Nevatia, 2004, Ameri, 2000) where the building models are confirmed or discarded during a verification stage. The authors generally take advantage of evidences provided either by a Digital Elevation Model (DEM), correlation scores, 3D feature extraction or shadow detection, according to the initial hypothesis generating method. Finally, the decision is taken by thresholding according to a prior knowledge, by maximizing posterior probabilities or by using a supervised classifier (Kim and Nevatia, 2003).

1.3 Overview

In this paper, quality self-diagnosis of 3D roof facets is performed by using overlapping aerial images. The problem of discriminating facets that comply or not with a set of quality specifications results in a three-class solution, namely an accepted, an undecided and a rejected class. Hence, our problem is expressed as a classification problem. First, an overview of building modelling errors is introduced in the section 2. Then, a set of image coherence measures is defined (section 3.) in order to prove the roof facet existence and to characterize its consistency. Attention is paid to their robustness and their complementarity. Their combination is performed in a supervised classification stage (section 4.). Finally, the algorithm is applied on two datasets in dense urban areas (section 5.), and evaluated by comparing its results with manual labels of the roof facets.

2. 3D ROOF FACET QUALITY ANALYSIS

In this section, we introduce the input dataset which is constituted of 3D roof facets and aerial images. A succinct overview of building modelling errors is provided.

2.1 Data

3D building models are described by a set of 3D planar polygons (the facets) which represents building roofs without small structure elements, such as chimneys or dormer-windows. Each roof facet is described by geometrical properties (a set of 3D vertices, 3D edges and a normal direction) and topological relations (3D connexity and 2D planimetric connexity between the facets). Their quality evaluation is performed by using multiple 25 cm resolution aerial images. They are acquired by a high quality digital frame camera (SNR=300). Each roof is viewed by 8 to 11 images.

2.2 Building Model Error Causes

In dense urban areas, errors in building modelling may occur because of the complexity of roof shapes, the presence of occlusions and vegetation. Besides, a lack of texture (along the roofs or inside the shadow areas) or a low contrast (along the building ridges) may mislead the reconstruction process. Additional external data which are often used, such as cadastral maps, are also error prone. Moreover, as regards to building reconstruction, some robust approaches do not manage some roof shapes while other more general approaches, based on feature detection, produce less robust and unpredictable results. Finally, buildings may have been destroyed, modified or extended between the database production and new image acquisition.

2.3 Building Model Errors

We may consider three kinds of errors in building modelling:

- the non-existence of the corresponding building,
- the shape description incorrectness which corresponds to under-modelling (Fig. 1) and over-modelling errors. It affects the topological relations and the geometrical characteristics of roof facets,
- the geometrical inaccuracy of a 3D facet, either in slope, altimetric location, and/or planimetric delimitation.

In the following, we focus on the verification of individual roof facet consistency including their existence, their shape description correctness and their geometrical accuracy.

3. CHARACTERIZATION OF THE COHERENCE BETWEEN THE IMAGES

In this section, overlapping images are used in order to collect positive or negative evidences of roof facet consistency. Among several image coherence characterization techniques, we use multi-image correlation and feature detection in order to define robust and complementary measures.

3.1 A Texture Coherence Analysis

Multi-image correlation techniques measure the similarity of textures over image-windows in order to get an estimation of the elevation such as in DEMs. Both the correlation scores and the estimated elevations bring an evidence of facet consistency, or on the contrary, find out a better solution.

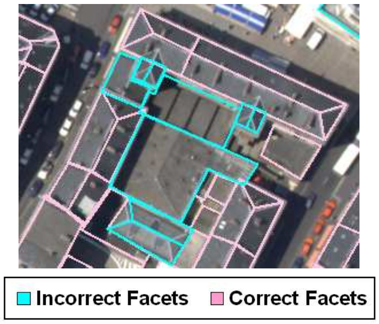


Figure 1: An example of correct and uncorrect (under-modelled) roof facets.

The multi-image correlation function defined in (Paparoditis et al., 2000) has been selected because it permits to compute efficiently DEMs in a multi-image context with a very low-level analysis (3×3 window size). The image similarity is estimated along the roof facet in the object space. The most probable elevation is estimated by maximizing the correlation function on a scan of a tolerance bound of $[-2m, 2m]$ along the vertical axis. Calling $\mathbf{v_i}$ the vector of intensity values, computed thanks to the implicit homography defined between the images, the multi-image correlation function (MIC) is defined by :

$$MIC = \frac{\text{Var}\ \sum_{i=1}^{n} \mathbf{v_i}}{\sum_{i=1}^{n} (\text{Var}\,(\mathbf{v_i}))} \quad \in [0, n] \tag{1}$$

where Var is the variance and n the image number. A preliminary image-window selection stage is performed in order to take into account the occlusions predicted by the building model dataset.

Facet Elevation Consistency Analysis A first clue of roof facet consistency is obtained by measuring the discrepancy between the expected elevation -predicted by the facet- and the estimated one. This difference is shown in Fig. 2.

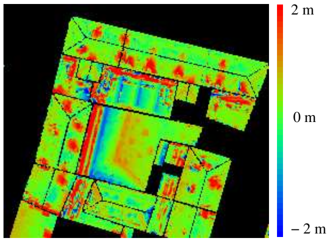


Figure 2: Vertical axis difference between the expected elevation (predicted by the facets) and the estimated one (estimated by maximizing the multi-image correlation function).

Although the under-modelled buildings can easily be identified, it should be noted that occlusions still disturb the elevation estimation. Indeed, occlusion prediction intrinsically depends on the geometric accuracy of the occluding buildings. Besides, elevation estimation is disturbed by the unmodelled roof structures such as chimneys or dormer-windows. Hence, a robust estimator such as the following pseudo-median function is required :

$$\text{med}(Y) = Y\left(\frac{min\,(\mathcal{S}_{\mathcal{F}}, \mathcal{S}_0)}{2}\right) \tag{2}$$

where Y is a ranged vector, $\mathcal{S}_{\mathcal{F}}$ is the facet area and $\mathcal{S}_0$ is an area threshold (500 m^2) used to cope with large facets. Hence, a first measure of facet consistency is based on the robust estimation of the distance between the estimated elevation points $\hat{P}(x, y, \hat{z})$ of each ground pixel (x, y) and their projection onto the facet plane $\mathcal{P}_{\mathcal{F}}$. It leads us to define the Correlation Distance ($\mathbf{CD}$) value by :

$$\mathbf{CD} = \underset{(x,y)\in\mathcal{F}}{\text{med}} \ \mathcal{D}_{\mathbb{R}^3}(\hat{P}(x, y, \hat{z}), \mathcal{P}_{\mathcal{F}}) \tag{3}$$

where $\mathcal{D}_{\mathbb{R}^3}$ is the euclidean distance in $\mathbb{R}^3$.

Correlation Function Profile Analysis Another clue assessing the roof facet consistency is provided by the correlation function profile. Correlation scores along the facets (Fig. 3, on the left) are expected to be high for correct facets and higher than those obtained along the vertical scan of the object space.

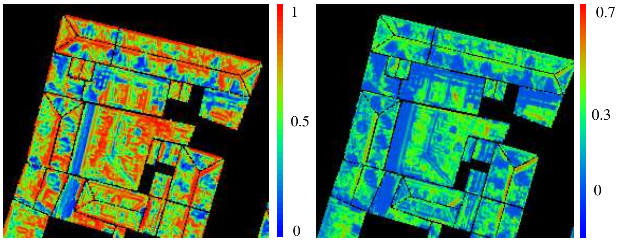


Figure 3: Multi-image correlation function (on the left) and correlation profile function (on the right) applied on the facets.

Two issues linked to the correlation function have to be handled. Firstly, homogeneous or periodic textures result in smoothed correlation profiles or in local extrema. Such profile should not be considered as reliable even if correlation scores are high. Secondly, since an image-window selection stage is carried out to cope with occlusions, the number of images varies from one pixel to another. Thus, a normalisation is required but linearity is not fulfilled. Both of these issues have been getting through by defining a new consistency measure based on the shape of the correlation profile (Fig. 3, on the right). It takes into account the correlation score $s_{\mathcal{F}}(x, y)$ obtained near the facet and its relative differences with the scores $s(x, y, z)$ obtained along the profile, where (x, y) is a ground pixel. Applying the pseudo-mediane function, we define the Correlation Profile ($\mathbf{CP}$) value by:

$$\mathbf{CP} = \underset{(x,y)\in\mathcal{F}}{\text{med}} \left({s_{\mathcal{F}}}^2(x, y) \sum_{z=-M_z}^{M_z} (s_{\mathcal{F}}(x, y) - s(x, y, z)) \right) \tag{4}$$

$$s_{\mathcal{F}}(x, y) = \max_{-\delta_z \le z \le \delta_z} (s(x, y, z)) \tag{5}$$

where M_z is the tolerance bound ($2m$) and δ_z the z-step ($0.25m$). Finally, the $\mathbf{CP}$-value is higher when the correlation profile has got a high (because of the square function) and unique peak nearby the facet (because of the sum of the relative differences).

3.2 A Structured Feature Coherence Analysis

A complementary way to assess the roof facet consistency is to take advantage of the high level of structuration of urban areas. Extracting these structures from the images allows us to verify the facet geometric characteristics and the shape description correctness. Hence, 3D segments detected from images provide positive clues when they overlap the facet edges or lay onto its plane, but also negative ones when they are found in a corner, or far away.

Edgel Extraction First, the detected contours are matched in order to produce robust, accurate and very low-level linear feature elements (Fig. 4) by using the reconstruction technique proposed in (Jung et al., 2002). These features, called “edgels”, are 3D points with a 3D tangent direction. Here, the facet is only used to determine the regions of interest in the images and the matching process is performed with photogrammetrical constraints. The corresponding contours are searched in a reliable and adaptative tolerance bound estimated thanks to a DEM. This bound is larger when the features are closer to the DEM discontinuities. Thus, the main structures of the scene can be extracted even if the facet is not correct.

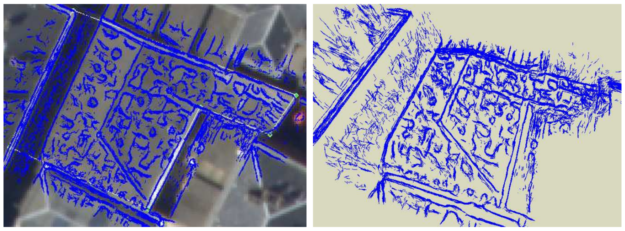


Figure 4: Reconstruction of edgels applied on an under-modelled facet. On the left, edgels projected on one image. On the right, a 3D view of the edgel set. The main structures of the roof building are well reconstructed wherever image contours have been extracted.

3D Segment Detection A set of relevant 3D segments are extracted from the edgel set in order to compare them to the facet. A filtering method enables to recover the segment direction and location, applied firstly in planimmetry and secondly in altimetry. Geometrical and filtering thresholds are required in order to get robust segments, the main ones being a required minimum number of edgels accumulated along the segment (linked to the image number). The 3D segments are detected inside three specific zones which are defined according to the facet (Fig. 5). A segment coherence value is defined for each zone.

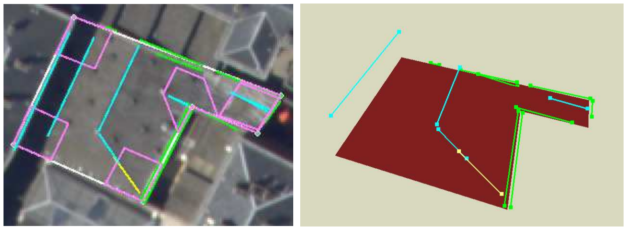


Figure 5: 3D segments detected for the under-modelled facet within three specific zones: near the edges (green), in the corners (yellow) and inside (cyan). Notice that a 3D inner segment belongs to a neighbouring facet (on its left). The corner zones are outlined in pink (on the left). In the 3D view (on the right), a 3D segment corresponding to a shadow boundary is occluded by the 3D facet. Even if many segments are detected near the facet edges, negative clues are collected by those detected in the corner and inside the facet.

Facet Edge Analysis An edge zone is defined for each facet edge with a distance and an angular deviation thresholds. The detection of 3D segments overlapping the facet edges allow to verify the facet boundary consistency and to detect over-modelling errors. An Edge Segment ($\mathbf{ES}$) value is defined by the weigthed coverage rate of the 3D segments $\{s_{j_0}, .., s_{j_n}\}$ projected onto their corresponding facet edge e_j:

$$\mathbf{ES} = \frac{\sum_{j=0}^{n} \alpha_j r(e_j, \{s_{j_0}, .., s_{j_n}\})}{\sum_{j=0}^{n} \alpha_j \|e_j\|} \tag{6}$$

where the function r computes the coverage length, α_j is a weight parameter (1/2 for edges belonging to several facets and 1 otherwise) and $\|.\|$ is the euclidean distance between the segment end-points.

Facet Corner Analysis A corner zone is defined for each facet vertice (pink polygons in Fig. 5, left) with a window width ($5m$)

and an angular deviation threshold (15°). The corner segments allow to verify the facet shape correctness and to detect under-modelled roof (a missing hip roof structure for instance). A Corner Segment ($\mathbf{CS}$) value is defined by the maximum of the summed length of the segment set $\{s_0, .., s_{j_n}\}$ detected in each corner zone j:

$$\mathbf{CS} = \max_j \sum_{i=0}^{j_n} \|s_i\|. \qquad (7)$$

Inner Facet Analysis The remaining edgels, that do not match with the neighbouring facet edges and that are inside the ground facet boundary, are selected in order to detect inner segments. They allow to assess the facet shape correctness. Finding a segment onto the facet plane may indicate a well localisation (the matched image contours may come from a two-material roof for instance). On the contrary, finding a segment far away from the facet plane, $2m$ above it for instance, may outline under-modelling errors, such a saw-tooth roof modelled by a flat facet. For each inner segment s_i, the area $\mathcal{A}(s_i, \mathcal{P}_\mathcal{F})$ defined between its end-points and their projection onto the facet plane is computed. This area is normalised by the length of the facet in the segment direction ($\|\mathcal{F}_{\vec{s_i}}\|$) in order to take into account the facet shape variability. Then, a Inner Segment ($\mathbf{IS}$) value is defined by the sum of the normalised areas of all inner segments:

$$\mathbf{IS} = \sum_{i=0}^{n} \frac{\mathcal{A}(s_i, \mathcal{P}_\mathcal{F})}{\|\mathcal{F}_{\vec{s_i}}\|}. \qquad (8)$$

4. FACET QUALITY SELF-DIAGNOSIS

We introduce in this section how image coherence characterization is used to classify the roof facet quality. First, four levels of quality are defined. Afterwards, a learning and a supervised classification are performed in order to associate and predict facet quality classes from the image coherence parameters.

4.1 Definition of quality classes

Four quality classes have been defined in order to value their level of adequacy with reality from false to correct:

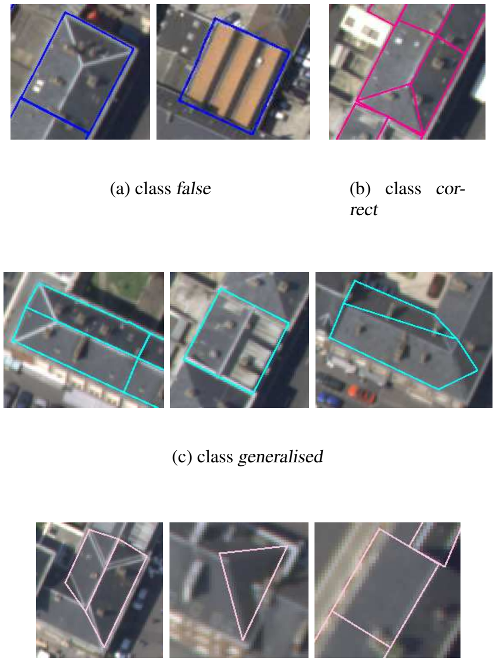
- *false*: the roof facet does not fit with the reality (Fig. 6(a));
- *generalised*: a part of the roof is not correctly modelled or geometric deviations are observed (Fig. 6(c));
- *acceptable*: the roof is quite well modelled, but either unmodelled hip roof ridge without geometric deviation or small geometric deviations are observed (Fig. 6(d));
- *correct*: the roof is correctly modelled by the facet (Fig. 6(b)).

The self-diagnosis process should alert the *false* and *generalised* facets and validate the *acceptable* and *correct* facets.

4.2 A Supervised Classification

The problem of self-diagnosing the quality of roof facets is expressed as a classification problem, whose inputs are the quality classes and a parameter vector $\mathbf{V} = \{\mathbf{CD}, \mathbf{CP}, \mathbf{ES}, \mathbf{CS}, \mathbf{IS}\}$. A simple classifier, the k-Nearest Neighbour (Duda et al., 2000), has been used to evaluate the efficiency of the image coherence measures. Each parameter is normalised by its standard deviation computed on the training instances. The euclidean distance between two normalised parameter vectors has been used.

Practically, 60 instances of each quality class have been learnt. Fig. 7 shows parameter mean and standard deviation for each quality class. Firstly, it shows that image coherence mean values are compliant with the quality classes, as expected. Secondly, even if the class *false* is quite well disjoined from the other ones, the classes *generalised* and *acceptable* are really close from each other. Indeed, these two labels are assigned whether a facet is acceptable or not, based on its shape correctness and geometrical accuracy. Finally, it shows that no measure alone is able to reliably classify each quality class.


(a) class *false*

(b) class *correct*

(c) class *generalised*

(d) class *acceptable*

Figure 6: Some facet instances of each quality class. The *false* and *generalised* ones should be identified as not acceptable by the self-diagnosis algorithm.

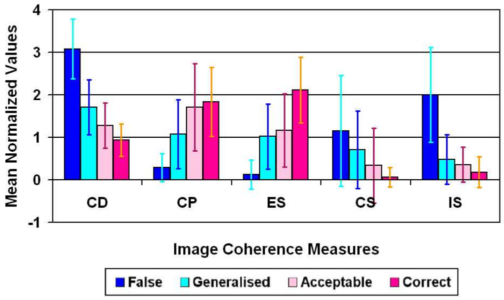


Figure 7: Image coherence parameter mean and standard deviation for each quality class when applied on the training instances.

4.3 Robust Decision Rules

In the following, the neighbour number k has been fixed to 15 which is a good trade-off between overfitting and generalisation. As the majority vote rule is not robust enough and does not reveal ambiguous classifications, the final decision is taken in order to translate the self-diagnosis results into the three classes of the traffic light paradigm. The decision is based on the number of neighbours N_F, N_G, N_A, N_C belonging to each class, the distance d of the k neighbours, and follows selective rules for acceptance:

- if $(N_F + N_G \geq \frac{k}{2}$ or $\max(N_F, N_G) \geq \frac{k}{3})$:
  if the majority vote says F or G, decide *Rejected*;
  otherwise, decide *Undecided*,
- if $(N_F + N_G \geq \frac{k}{3}$ or $d \geq \beta k)$: decide *Undecided*,
- otherwise: decide *Accepted*.

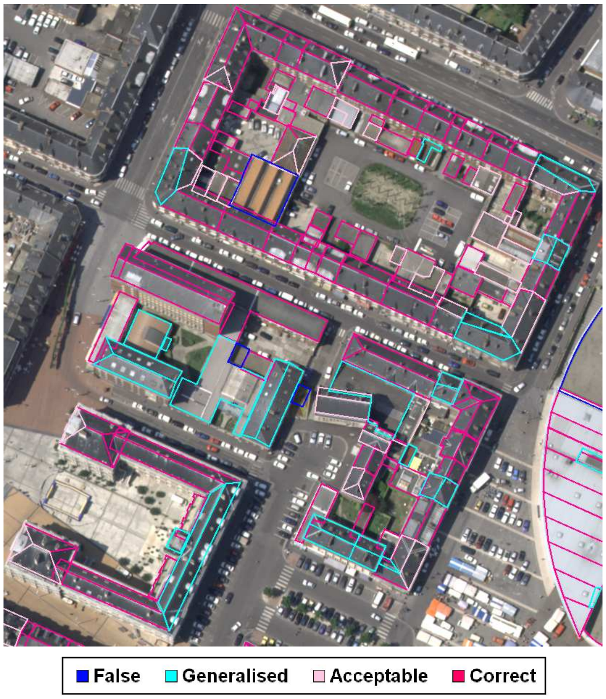


Figure 8: Manually labelled roof facets of the realistic dataset.

Here, the maximum distance of all the neighbours has been fixed with $\beta = 1.2$. Moreover, neighbours at a distance null have been excluded, enabling to merge the results of the training and testing examples in the next section.

5. RESULTS

5.1 3D Building Model Datasets

We have chosen two dense urban areas of Amiens, France. The first one is composed of many similar buildings, mainly with gable roofs, hip roofs and low slope garage roofs within courtyards. The second area is composed of many different roof materials and shapes, with a mix of industrial and very small buildings.

Two building model datasets have been used for the self-diagnosis evaluation. The first one, called realistic (Fig. 8), has been produced semi-automatically by a platform containing several algorithms (Flamanc and Maillet, 2005). The main modelling errors (nearly 20% on 862 facets) are hip roof with missed structures, industrial buildings simply modelled by a flat roof and small buildings poorly modelled. In order to get enough modelling errors for statistical evaluation, we complete the first dataset with a second one simulating systematic errors and containing only flat roof facets (80% incorrect facets on 251). They have been delimited by 2D vectorial building footprints and are located at the median altitude given by a DEM. As buildings have many different shapes, the simulated discrepancies between the reality and the models are widespread. All roof facets that are flat in the realistic dataset have been removed from the flat roof dataset. This one provides all the training instances of the class *false* and a few ones of the class *generalised*, while the realistic dataset provides all the other ones.

5.2 Quantitative Results

As regards to the semi-automatic building model verification, an operator will inspect the *rejected* and *undecided* facets. Thus, the self-diagnosis process makes two erroneous decisions : a False Acceptance (FA) error when a *false* or *generalised* facet is classified as *accepted*, and a False Rejectance (FR) error when an *acceptable* or *correct* facet is classified as *rejected* or *undecided*. It should be emphasis that minimizing the FA errors is the most important because the *accepted* facets will not be inspected anymore. FR errors only correspond to time lost for an operator to inspect facets while ideally it would not have been required.

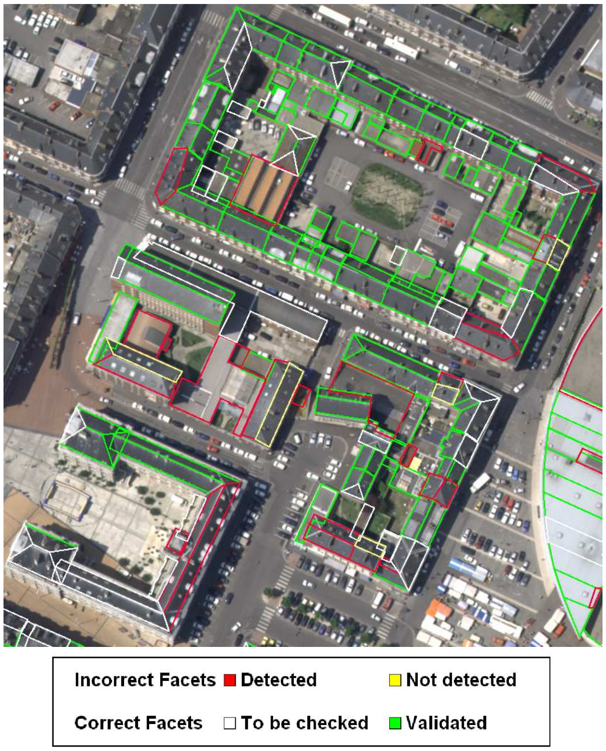


Figure 9: Correct (detected/validated) and incorrect (not detected/to be checked) self-diagnosis decisions of the realistic dataset.

The results of the self-diagnosis algorithm are provided in Table 1 merging all the facets of both datasets. The percentages are computed according to the facet number of each quality class. The self-diagnosis algorithm detects almost all the *false* modelling errors (0.5% FA rate), but the results are mixed with the class *generalised* (nearly 20% FA rate) which is confused with the classes *acceptable* (11%) and *correct* (9%). As the decision rule is selective for acceptance, only 52% and 80% of correct acceptance rates are reached for the classes *acceptable* and *correct* respectively. Globally, the rates of correct rejectance (91%) and correct acceptance (73.7%) are quite satisfying, especially considering that an overall rate of 79.6% of correct decisions is reached with only 3% of false acceptance errors on the whole datasets (containing 1113 roof facets).

Quality class	Decision			Facet number
	Rejected	*Undecided*	*Accepted*	
	Correct R.		FA error	
false	96.2%	3.3%	0.5%	209
generalised	63.5%	17.1%	19.4%	170
	FR error		Correct A.	
acceptable	28.9%	19.1%	52%	173
correct	9.4%	10.2%	80.4%	561

Table 1: The statistics of the self-diagnosis decisions according to the quality class on the whole datasets.

5.3 Qualitative Results

An overview of the correct and incorrect self-diagnosis decisions is provided on the realistic dataset (Fig. 9). It shows that many facets are correctly classified. As regards to the false rejectance decisions (Fig. 10), the facet quality estimation is generally mislead either by occlusions, by the presence of shadows, dormer-windows or a hip roof ridge without geometric deviation. Even if these facets have been manually labelled as *acceptable* or *correct*, their inspection by an operator may be well-founded - for some of them at least-.

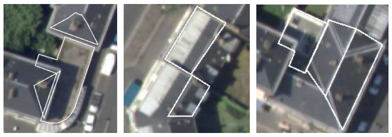
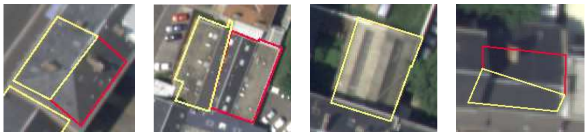


Figure 10: Examples of false rejectance decisions from the realistic dataset.

Let's now focus on the analysis of the false acceptance decisions. Considering the results presented in Fig. 9, only 4 false acceptance decisions have been made on 33 incorrect facets. In figure 11(a), the facet is geometrically accurate but a hip roof structure is not modelled. This is not detected by corner segments because the unmodelled roof ridge is not contrasted enough. Therefore, no evidence of incorrectness shape description has been proved (as for 13 FA errors on 34). In figure 11(b), a part of an overhanging roof is modelled by the facet of interest. The roof slope and location are correct, its boundary is covered at 44% by edge segments. As the corner segment directions do not fit with the roof ridge, no corner segment has been detected. Therefore, using the pseudo-median function and based on the detected segments, this facet has been erroneously validated. In figure 11(c), a double side roof is modelled by an horizontal facet. Even if the middle ridge is detected by inner segments (IS = 2.6), fair geometric deviation measurement (CD = $50cm$) and edge segment coverage (ES = 50%) lead to its validation. In figure 11(d), the roof slope is deviated by the neighbouring roof. While the CD value is quite the same, smaller CP value and edge coverage (ES = 28%) are balanced by a very small inner segment value (IS = 0.2). Finally, based on 10 neighbours belonging to the class *acceptable*, this facet has been validated.



(a) a missing hip roof structure

(b) an extended roof

(c) an under-modelled roof

(d) a deviated slope roof

Figure 11: Examples of false acceptance decisions (yellow outlined) from the realistic dataset.

6. CONCLUSION AND PERSPECTIVES

We have introduced a new approach for a quality self-diagnosis of roof facets in dense urban areas. It is based on the definition of robust and meaningful image-driven measures that aims at characterizing individual roof facet existence and consistency. The originality of our work is to take advantage of a set of very low-level image observations and of a supervised learning in order to classify roof facet quality. Although a simple classifier is used, it has shown very promising results in the difficult context of dense urban areas with the detection of almost all the *false* modelling errors (99.5%). Considering also the *generalised* modelling errors, which should be alerted, the evaluation shows very satisfyingly that only 3% of false acceptance decision is made on the whole datasets. Our efforts will be focused on the improvement of the *generalised* facet detection (only 81%) while keeping a good correct acceptance rate.

Future works will be carried out on the completion of the image coherence measures by considering the radiometric changes between the images and the change consistency with the facet specular direction. It will allow us to assess the slope of the roof facets when the roof material is not lambertian. Besides, others classifiers could be used, as linear separation or neural networks for instance, in order to improve the classification stage.

7. ACKNOWLEDGEMENTS

The author would like to acknowledge the French Defence Agency (DGA) for supporting this research.

REFERENCES

Ameri, B., 2000. Feature based model verification (FBMV): A new concept for hypothesis validation in building reconstruction. In: Proceedings of the XIXth ISPRS Congress, IAPRS, Vol. 33, B3, Amsterdam.

Duda, R., Hart, P. and Stork, D., 2000. Pattern Classification. Wiley Interscience.

Durupt, M. and Taillandier, F., 2006. Automatic building reconstruction from a digital elevation model and cadastral data : An operational approach. In: Proceedings of the ISPRS Commission 3 Symposium on Photogrammetric Computer Vision, Bonn, Germany.

Flamanc, D. and Maillet, G., 2005. Evaluation of 3D city model production from pleiades-HR satellite images and 2D ground plans. In: 3rd International Symposium Remote Sensing and Data Fusion over Urban Areas, Tempe, USA.

Förstner, W., 1994. Diagnotics and performance evaluation in computer vision. In: Performance versus Methodology in Computer Vision, NSF/ARPA Workshop, IEEE Computer Society, Seattle.

Förstner, W., 1996. 10 pros and cons against performance characterization of vision algorithms. In: Workshop on Performance Characteristics of Vision Algorithms, Cambridge.

Henricsson, O. and Baltsavias, E., 1997. 3-D building reconstruction with ARUBA: a qualitative and quantitative evaluation. In: Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Ascona, pp. 65–76.

Jibrini, H., Pierrot-Deseilligny, M., Paparoditis, N. and Matre., H., 2004. Détermination d'une surface polyédrique continue optimale à partir d'un fouillis de plans. In: RFIA' 04, Vol. 1, AFRIF-AFIA, Toulouse, France.

Jung, F., Tollu, V. and Paparoditis, N., 2002. Extracting 3d edgel hypotheses from multiple calibrated images: a step towards the reconstruction of curved and straight object boundary lines. In: Proceedings of the ISPRS Photogrammetric Computer Vision, IAPRS, Vol. 34, Graz, Austria, pp. B100–104.

Kim, Z. and Nevatia, R., 2003. Expandable bayesian networks for 3d object description from multiple views and multiple mode inputs. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(6), pp. 769–774.

Kim, Z. and Nevatia, R., 2004. Automatic description of complex buildings from multiple images. Computer Vision and Image Understanding 96(1), pp. 60–95.

McKeown, D., Bulwinkle, T., Cochran, S., Harvey, W., McGlone, C. and Shufelt, J., 2000. Performance evaluation for automatic feature extraction. In: Proceedings of the XIXth ISPRS Congress, IAPRS, Vol. 33, Amsterdam, pp. 379–394.

Meidow, J. and Schuster, H., 2005. Voxel-based quality evaluation of photogrammetic building acquisitions. In: CMRT'05, Vienna.

Paparoditis, N., Thom, C. and Jibrini, H., 2000. Surface reconstruction in urban areas from multiple views of aerial digital frames. In: Proceedings of the XIXth ISPRS Congress, IAPRS, Vol. 33, B3, Amsterdam.

Rottensteiner, F. and Schulze, M., 2003. Performance evaluation of a system for semi-automatic building extraction using adaptable primitives. In: IAPRS, Vol. 34, Munich.

Schuster, H.-F. and Weidner, U., 2003. A new approach towards quantitative quality evaluation of 3d building models. In: J. Schiewe, L. Hahn, M. Madden and M. Sester (eds), ISPRS com IV, Workshop "Challenges in Geospatial Analysis, Integration and Visualization II", Stuttgart, Germany, pp. 156–163.

Suveg, I. and Vosselman, M., 2002. Mutual information based evaluation of 3D building models. In: 16th International Conference on Pattern Recognition (ICPR 02), Vol. 3, IAPR/IEEE Computer Society, Quebec city, pp. 557–560.

Taillandier, F. and Deriche, R., 2004. Automatic buildings reconstruction from aerial images: a generic bayesian framework. In: Proceedings of the XXth ISPRS Congress, IAPRS, Istanbul.

Thacker, N., Clark, A., Barron, J., Beveridge, R., Clark, C., Courtney, P., Crum, W. and Rameh, V., 2005. Performance characterization in computer vision : A guide to best practices.

AUTOMATIC BUILDING RECONSTRUCTION FROM A DIGITAL ELEVATION MODEL AND CADASTRAL DATA : AN OPERATIONAL APPROACH

Mélanie Durupt Franck Taillandier

Institut Géographique National - Laboratoire MATIS. 2, Avenue Pasteur. 94165 SAINT-MANDE Cedex - FRANCE
melanie.durupt@ign.fr - franck.taillandier@ign.fr

KEY WORDS: Digital Elevation Model, Cadastral Maps, RANSAC, Plane Extraction, Building, Polyhedral Surface, Three-Dimensional Modeling.

ABSTRACT:

In this paper, we tackle the problem of automatic building reconstruction using digital elevation model and cadastral data. We aim at massive production of 3D urban models and present thus an algorithm, that is an adaptation of a more general and semi-automatic strategy to an operational context where robustness is essential. We present two approaches relying on two different techniques for non vertical planes extraction using constraints inferred by cadastral limits. The first one consists in inferring planar primitives by estimating only two parameters for each building : the height of gutter and the slope of roofs. The other idea is to extract planar primitives directly from the cadastral limits and from the DEM, using a robust RANSAC estimation algorithm. The results of an evaluation carried out on 620 buildings on a dense urban centre are promising and enables to compare both approaches.

1 INTRODUCTION

1.1 Context and objectives

In this article, we deal with automatic building reconstruction from aerial images to define a production line for massive production of 3D urban models. Real time, robustness and automation are then essential criteria.

We propose here to adapt a generic reconstruction algorithm (Taillandier and Deriche, 2004) to an operational context. The general algorithm implements a hypothesize-and-verify strategy where buildings are modeled in a very general way as any polyhedral shapes with no overhang. This algorithm only uses aerial images but some limitations prevent its direct use in a context of massive production of 3D models where robustness of building reconstructions is more important than generality. Especially, to overcome the weakness of primitives detector, we now propose to use cadastral limits where polygones define buildings outlines and a digital elevation model. We will present the necessary adaptations to implement this algorithm using these data in the context of an operational production line where real-time and automation are key issues.

1.2 State of the art

Automatic building reconstruction has interested the community for more than ten years and numerous works have focussed on this subject. Several strategies in various context have appeared : data-based or model-based approaches, in a stereoscopic context or from multiple aerial images, with or without external data.

Stereoscopy allows to obtain a reliable 3D information but there can be occlusions problems in a dense urban environment. Thus, in this context, in order to overcome the lack of information, methods often implement model-based approaches ((Cord et al., 2001), (Paparoditis et al., 1998)). They consequently suffer from a lack of generality.

Multiscopy allows to avoid occlusions problems, therefore, methods are more general and implements data-based strategy. Most of the developed approaches use only one kind of primitives (corners for example for (Heuel et al., 2000)). The major drawback of these methods is their lack of robustness. In (Baillard and Zisserman, 1999), for instance, the method described allows to produce generic models from aerial images. It is based on the detection of 3D segments and then on facets detection around these segments by correlation. Planes intersection allows to define roofs. The main drawback of this method is the absence of under-detection handling and its lack of robustness making it not adapted to a massive production environment.

Cadastral limits allow to add strong information on structures and have been studied for building reconstruction. (Flamanc et al., 2003) developed a model approach using cadastral limits to deduce possible skeleton of the building and then a possible models library. The principal disadvantage of this approach is its lack of generality. In (Vosselman and Suveg, 2001), authors propose to segment the cadastral parcel in elementar rectangles. Each rectangle can represent an elementar form among three possible shapes. The set of possible models is built from the collection of possible segmentations of the parcel. This method can provide robust models but it is not adapted to our context due to the high number of generated hypotheses and therefore the induced computing time for construction and evaluation of these hypothesis.

The approach of (Jibrini et al., 2000) utilizes cadastral limits so as to constraint planes search by a Hough transform technique. The enumeration algorithm is very interesting, the general strategy of this article is an extension of it. However, planes extraction with Hough transform gives a lot of over detections and leads to a combinatory explosion and then to a lack of robustness of the reconstructed models, which penalises this algorithm.

1.3 Structure of the article

We first describe a general algorithm of building reconstruction (part 2.1). We then detail the adaptations for its use with cadastral maps : on the one hand by simulating planar primitives (part 2.2.1), on the other hand, by extracting planar primitives with RANSAC algorithm (part 2.2.2).

We will present the results of an evaluation of these two methods led on the urban center of Amiens. Finally, we conclude and present future work.

2 BUILDING RECONSTRUCTION

2.1 Original algorithm : polyedral model without overhang

The complete description of this general algorithm can be found in (Taillandier and Deriche, 2004).

Reconstruction is performed solely from aerial images without any cadastral information. However as it is shown afterwards, user-interaction is still needed in the focusing step. A building is modeled as a polyhedral form, without overhang and whose outline is constituted with vertical planes. This very generic modeling allows to represent almost all of the buildings in urban area.

We briefly sum up the general methodology. The main steps are shown on an example on figure 1. For each building or group of buildings, the operator manually selects a focusing area and a ground altitude (the maximum altitude is automatically deduced), therefore delineating a volume in which reconstruction should be performed. Reconstruction is then achieved in a four-step algorithm.

First, planar primitives (plane facets and oriented facades) and 3D segments are automatically detected in the volume.

In the second step, a 3D graph is generated from the intersection of all the detected planar primitives (facades and non-vertical planes). After simplifications in this graph, it is proven that the search for all possible shapes of buildings is equivalent to the search for maximal cliques in an appropriate graph. All possible models of buildings are thus enumerated in this second step leading to a set Γ of possible solutions.

In the third step, the best model $\widehat{M}$ is then chosen in the entire set of possible solutions Γ by bayesian modeling (equation 1) by taking into account the adequation of the model with observations D ($P(D/M)$ term) and simplicity of the form ($P(M)$ term).

$$\widehat{M} = \arg\max_{M\in\Gamma}(P(M/D)) = \arg\max_{M\in\Gamma}\{P(D/M)\cdot P(M)\} \tag{1}$$

The term related to the adequacy of the model with the data, $P(D/M)$, allows to take into account the adequation of the model with external data (detected segments, over-ground mask, images ...). The term related to complexity form probability, $P(M)$, is inspired by the Shannon relation and is linked to the description length $\mathcal{L}(M)$ of the model (that takes into account for example topological and geometrical informations).

$$P(M) = C \cdot \exp^{-\frac{\mathcal{L}(M)}{\beta}} \tag{2}$$

The parameter β allows to adjust the data term and the complexity term, C is a normalization term common to all models and then omitted afterwards.

The final step of the algorithm consists in automatic application of geometrical constraints on the chosen model.

This general algorithm allows to reconstruct very complex buildings, buildings with internal facades and even several buildings on the same focusing area. However, in an operational context and in dense urban environment, some limitations are very restrictive : focusing on an area is a manual action and the exhaustive exploration of all possible models can involve combinatory problems since the maximal clique exploration is a NP problem. Finally, even if the algorithm can manage errors of the primitives detector, this primitives extraction is only made from images : quality of primitives geometry strongly depends on images quality and errors of primitives extraction are mostly the cause of false reconstructions.

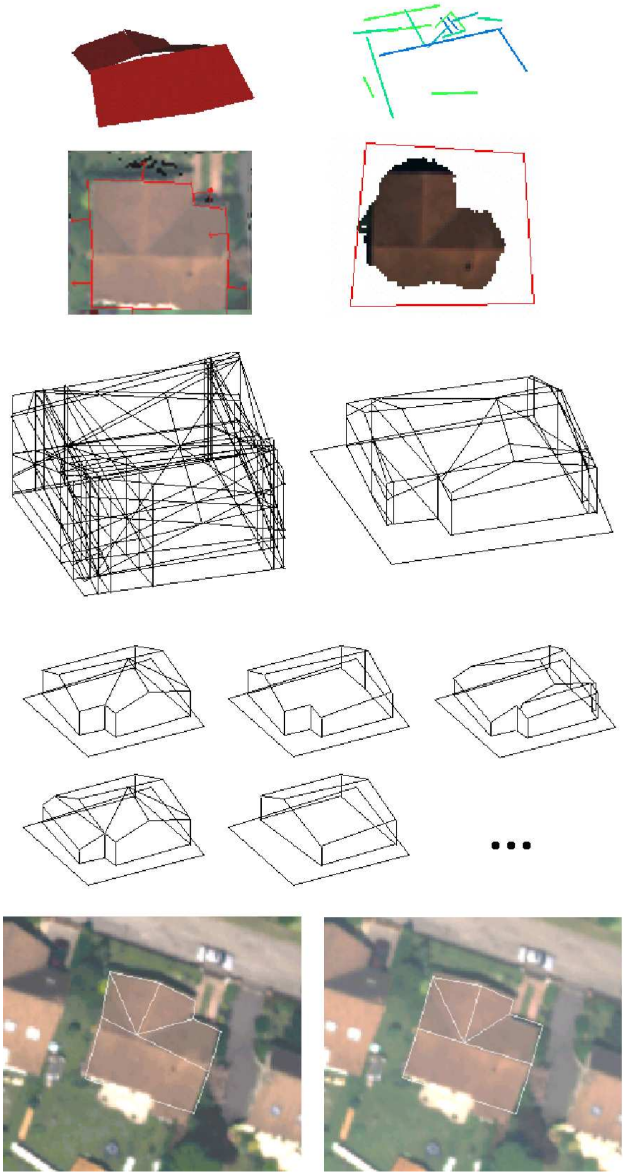


Figure 1: Each step of the algorithm applied to an example. 1st level : primitives detection (non vertical planes, 3D segments, facades, over-ground mask) ; 2nd level : resulting 3D graph before and after simplifications ; 3rd level : enumeration of possible solutions ; 4th level : superposition of the model chosen on a true orthophoto, before and after constraints application.

2.2 Adaptation for an operational context

The objective is to adapt the former algorithm to integrate it in an operational software. Whereas generality was previously favoured, we now aim at more robustness with a real-time constraint.

In this context, cadastral data bring useful information. The outlines of the buildings are indeed essential to solve some problems: focusing area delineation is automatic and primitives detection is easier. Indeed, we have directly facades hypotheses and planar primitives detection can be restricted to some directions orthogonal to principal directions given by the building outlines. As the number of primitives is therefore reduced, we do not have any combinatory problems for the maximal cliques enumeration. In the following, we present two techniques for planes extraction using cadastral outlines, the first one using strong constraints, the second one relaxing these constraints.

2.2.1 Planes simulation The objective of planes simulation is to introduce strong constraints on planar primitives in order to improve robustness. This method is described in details in (Taillandier, 2005). From the cadastral maps, non vertical planes are inferred with the following rules :

- From each segment of the cadastral outline a gutter segment of z_g m is deduced. Initially, z_g is arbitrarily fixed at 0m.
- One plane is extracted for each gutter segment, orthogonally to this segment.
- All planar primitives have a given slope p initially fixed at 45 ˚ .

From these plane primitives, we can then enumerate every possible reconstructions with the maximal cliques enumeration technique recalled in part 2.1. Some models are not however likely to represent building roofs (figure 2). A pruning step is thus necessary, in order to make the search for solutions more reliable. We constrain each facet to pose on the segment that has generated it, we impose a minimum angle between two edges (10 ˚) and a minimum surface of the facets (1 m^2).

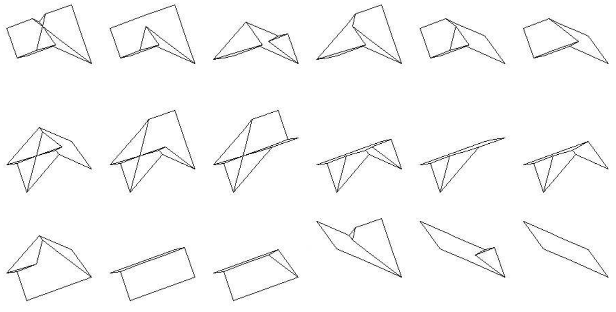


Figure 2: 18 possible solutions on a total of 83

The resulting models are enough simple to consider them equiprobable (figure 3) and discard the complexity term in the choice process. The solution is then chosen only on a criterion of adequacy to the data. In our case, since we initially fixed arbitrary altitude of gutter and slope, we use centered correlation on DEM (figure 4) as adequacy term to be independent of these arbitrary values.

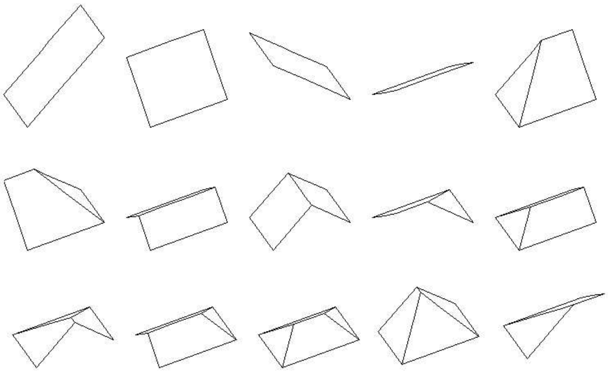


Figure 3: The 15 remaining solutions after simplifications

The last step consists in estimating altitude of gutter and slope of the planes. They are computed by minimization on the correlation DEM. A point M on a plane generated by a segment of gutter S and at the distance D_M from S is at the altitude z_M :

$$z_M = z_G + p \cdot D_M \tag{3}$$

where z_G is the altitude of gutter, p the slope of planes and D_M the orthogonal distance from the point P to the segment S. We minimize with L1 norm the difference between this model and the correlation DEM. This norm has been chosen because of its robustness : it is useful to overcome errors of correlation in the DEM and the non modeled superstructures of the roof.

Results obtained with this method are very good : 85% of the reconstructions are acceptable (see part 3.3). The real time constraint is also respected : in the very large majority of the cases, a result is obtained in less than 1 second.

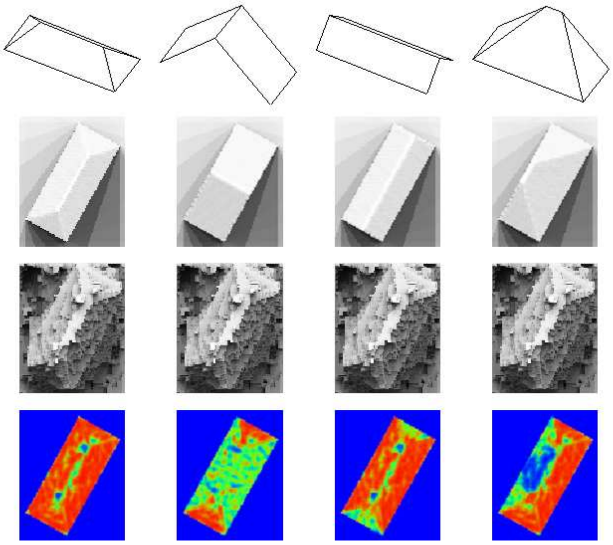


Figure 4: We calculate an altitude map corresponding to each model (2$^{\text{nd}}$ row) that is compared with the reference DEM (3$^{\text{rd}}$ row). The last row is the result of centered correlation between the 2 images (red : high correlation scores ; blue : low correlation scores)

2.2.2 Direct extraction of planes In the method previously described, constraints are very strong : there is only one slope of roof and one height of gutter per building. The objective of this second technique is to relax contraints in order to try to obtain more generality while maintaining a high level of robustness and then have more realistic reconstructions on buildings with irregular forms. In this case, to extract planes, we now exploit cadastral limits and DEM.

Cadastral limits utilisation Outlines of the buildings allow us to limit planes extraction. Indeed, most of the gutter being horizontal, we impose to a plane extracted from a gutter G that the horizontal component of its normal vector is perpendicular to G (figure 5). This implies that only 2 3D points and one 2D direction allow to define a plane. The first step of our approach is therefore to extract these particular directions from the outline of the building. The use of principal directions rather than the original segments from the outline allows for example to impose constraints of symmetry on the planes.

These principal directions allow to restrain the space of search of the planes in the DEM. The strategy used for extracting planes in the DEM is the robust algorithm of RANSAC.

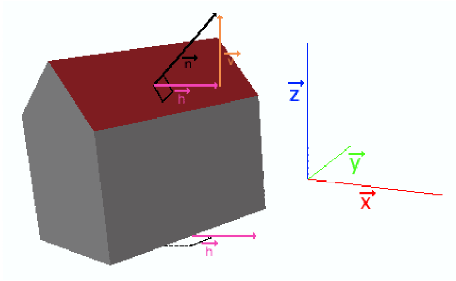


Figure 5: The horizontal component of a normale to a plane is perpendicular to the segment that generated this plane

RANSAC algorithm The principle of RANSAC algorithm (Fischler and Bolles, 1981) is to estimate parameters with the minimum necessary observations. These observations are selected randomly among the set of observations and we count the number of observations compatible with the model deduced. These steps are reiterated and the model chosen is the one that maximizes the consensus.
Two parameters have to be estimated : the error tolerance to determine whether or not an observation is compatible with a model and the number of tests to realize. The number of tests to realize k is (see (Fischler and Bolles, 1981)) :

$$k = \frac{\log(1-p)}{\log(1-w^m)} \tag{4}$$

where p is the probability that at least one subset of observations is correct, m is the number of necessary observations to estimate the model and w the probability that any observation is compatible with the model.
In our particular case, we want a plane equation and the set of observations is the 3D points of the DEM included in the cadastral parcel. We fix p at 99% and w is $\frac{n}{N}$, where n is the minimal number of observations to insure a plane presence (therefore n is linked to a minimal surface of planes that we want to detect) and N is the total number of points to consider. The other parameter is the error tolerance, in our case, it is linked with the intrinsic quality of the DEM used : σ_z. As we know the ratio $\frac{B}{H}$ of the aerial images that have been used to compute the DEM, and the resolution of these images, we can deduce σ_z :

$$\sigma_z = \frac{\text{resolution}}{B/H} \tag{5}$$

In our case, we do not want to find only one model, but several planes. Therefore we apply a few times this algorithm and remove at each iteration the points that are compatible with the plane detected. We stop the processus when there are not enough points remaining (less than 5% of the total of points). We explicitely introduce the knowledge of the principal directions so as to constraint the normal vector of the planes to extract but also to reduce the number of attempts. Indeed, in formula 4, m is the number of observations to define a plane. In general case, $m = 3$ (3 points define a plane), but since we have this additional data, $m = 2$. Therefore, we will randomly choose k couple of observations for each principal direction and finally choose the planes that maximize the set of consensus. For instance, for the building in figure 6 (6387 points of the DEM are inside the parcel and it is 27m large), the number of attempts is near 22 millions. With the pincipal directions, this number reduces to 260000 (130000 per direction). The phase of tests is critical for the complexity of our algorithm. So as to reduce it again, we make a realistic hypothesis : we suppose that each plane is in contact with a gutter. This will allow us to choose each couple of observations near a gutter segment (and inside the cadastral parcel). This method seems less precise, but we overcome this drawback by taking into account all points inside the cadastral outline to compute the set of consensus. For the same example, if the attempts are made in a 2m large belt around each segment, the number of attempts is reduced to 6200.

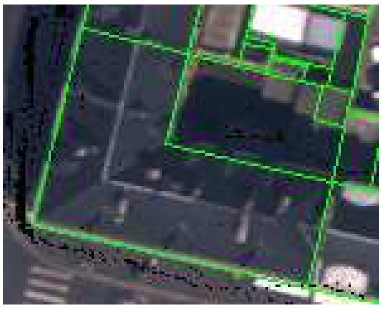


Figure 6: Example of a building

Choice of the best model After planes extraction, the 3D graph is built and the possible models are then enumerated according to the general algorithm. In order to choose the best model, we use the bayesian formulation described in part 2.1. Indeed, we have in general more models than with the method using simulated planes, it is then essential to reintroduce the complexity term. The adequacy to the data term is only computed in relation to the reference DEM. For each facet f of a model, we compute a score with the formula 6 :

$$\text{score} = \frac{\text{surf}(f)}{\underset{P\in f}{\text{card}(P)}} \cdot \frac{1}{\sigma_z} \sum_{P\in f} |\text{DEM}_{\text{ref}}(P) - \text{DEM}_f(P)| \tag{6}$$

where $\text{surf}(f)$ is the surface of the projected of the facet f in 2D, $\underset{P\in f}{\text{card}(P)}$ is the number of points of the DEM that belong to the facet f, $\text{DEM}_{\text{ref}}(P)$ is the altitude of the point P in the original correlation DEM and $\text{DEM}_f(P)$ is altitude of the point P projected vertically on the facet f. The score of a model is the sum of the score for each facet. Eventually, to link this quantity to equation 1 :

$$\sum_{f\in M} \text{score}(f) = -\ln(P(D/M)) \tag{7}$$

hence (see equation 1) :

$$\widehat{M} = \underset{M\in\Gamma}{\arg\min} \left\{ \sum_{f\in M} \text{score}(f) + \frac{\mathcal{L}(M)}{\beta} \right\} \tag{8}$$

We can adjust the adequacy term and the complexity term with the parameter β.

Results After all these adaptations, 89% of the reconstructions are acceptable (part 3.3). However, a few seconds are necessary for planes extraction. For instance, for the building on figure 6, the result is given in 4 seconds.

3 METHODS EVALUATION

3.1 Data

The evaluation was performed on 620 buildings of the urban center of Amiens (France). We have a correlation DEM of 25cm resolution (Pierrot-Deseilligny and Paparoditis, 2006) and the cadastral maps, preliminarily corrected so that each parcel corresponds to a building.

3.2 Some results

We present here some results (figure 7 and figure 8) and differences that can appear between the methods exposed.

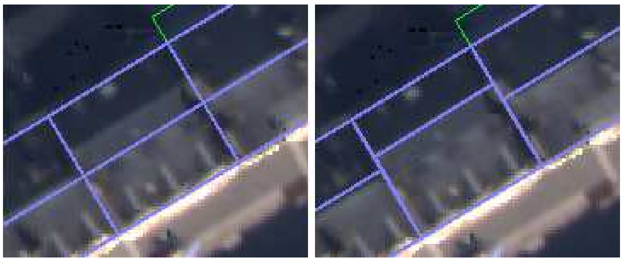


Figure 7: Comparison of the methods in the particular case of an asymmetrical building. The solution given by the method using simulated planes represents a symmetrical roof (on the left), the one given by the method using RANSAC planes extracted can represent this asymmetrical case.

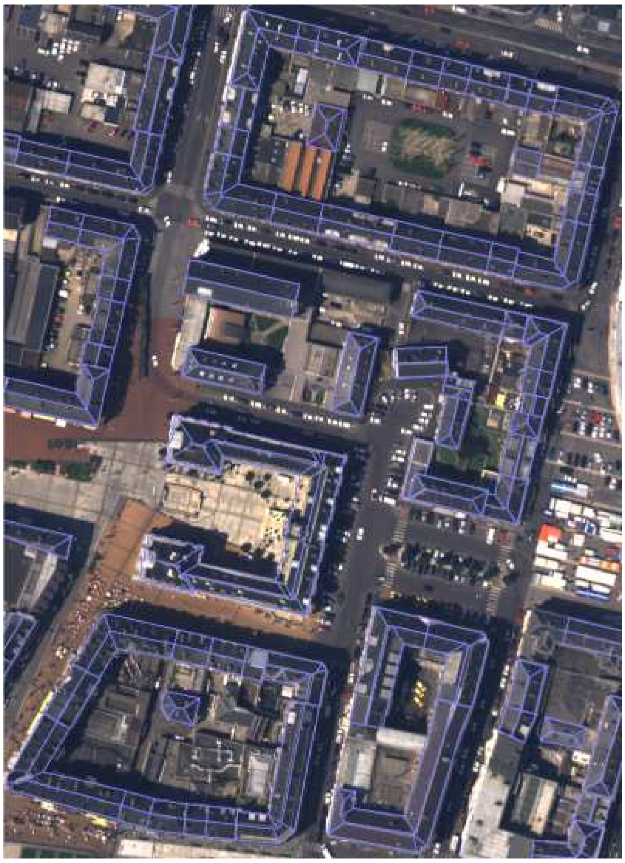


Figure 8: Results obtained by the method using the first technique on a part of the urban center of Amiens

3.3 Visual evaluation

We want here to determine for each reconstruction "up to what point it corresponds to reality" ; only topological structure is observed. In this test, we do not take into account geometrical precision of the reconstructions. Each reconstruction of the evaluated area has been classified in one of these categories :

- *correct :* the reconstruction is in conformity with reality (we still tolerate oversights like fanlights and chimneys).
- *generalized :* the reconstruction is an acceptable caricature of the reality. We fix a limit that is the maximal size of details that can be forgotten (level of generalization of 1,5m).
- *surgeneralized :* it deals with reconstruction that could have been classified in the category "generalized" for a superior level of generalization.
- *false :* the reconstruction cannot be accepted, whatever the level of generalization chosen.

We present a synthesis of the results on the 620 buildings studied (table 1). The column "simulation" sums up the results obtained using simulated planes. The columns "β = 1000", "β = 500", "β = 100" synthesize the results obtained with RANSAC extracted planes and with different values for the parameter β. The best value for the parameter β, in term of rate of acceptable reconstructions (correct and generalized) among the three values tested is β = 500.

	Simulation	β = 1000	β = 500	β = 100
correct	76.61%	73.71%	73.23%	61.61%
generalized	9.71%	13.71%	16.13 %	25.48%
surgen.	0.16%	3.06%	6.55%	5.97%
false	14.03%	8.87%	6.77%	6.29%
failure	0.48%	0.65%	0.32%	0.65%
Total	100%	100%	100%	100%

Table 1: Results of the evaluation expressed as percentages

The rate of acceptable reconstructions using simulated planes is over 85%. Using RANSAC extracted planes and the parameter β = 500, the rate of acceptable reconstructions is 89%. However, we can notice that in term of exact reconstructions, the method using simulated planes is more effective.

It is interesting to cross the results of these two methods to know if a false reconstruction with one method is correct with the other one (table 2). For RANSAC extracted planes, we consider the parameter β = 500, the one that gives the best results.

We can read in table 2 that for 91 non acceptable reconstructions with simulated planes, 66 are acceptable with the other method (75%). We can then hope by stringing both methods together to correctly reconstruct more than 95% of the buildings.

	correct	gener.	surgen.	false/fail.	Total
correct	397	2	1	54	454
gener.	47	42	0	11	100
surgen.	12	5	0	5	22
false/fail.	19	5	0	20	44
Total	475	54	1	90	620

Table 2: The results for RANSAC extracted planes are in lines, crossed with results obtained with simulated planes in colomns. For instance, 54 false reconstructions with simulated planes are correct with RANSAC extracted planes

4 CONCLUSION AND PERSPECTIVES

4.1 Conclusion

We have presented in this article two methods producing automatically 3D models of buildings. The method using simulated planes gives acceptable results in 85% of the cases. The method using RANSAC planes extracted gives acceptable results in 89% of the cases. In the case of planes simulation, the execution is real time (less than 1 second) whereas a few seconds are necessary in the case of RANSAC planes extraction.

4.2 Perspectives

Three developments are envisaged. To be operational in a context with data of lower resolution (50-70cm), it can be useful that an operator can lead the algorithm to choose a general form of the solution. For example the operator could impose that the final reconstruction has a saddleback roof or a hip-roof. This is only valid for the method using simulated planes.

Then we will carry on the relaxation of contraints and introduce internal facades of buildings, this extension being possible in the original general algorithm.

At last, we will implement an alert system. Indeed, in order to make the process even faster, we can consider that in a massive production context of 3D database, an operator launches the reconstruction algorithm using simulated planes on a large area and only verifies the buildings whose reconstruction have given an alert by the system. For these buildings, either he confirms the reconstruction or he lauches the method using RANSAC planes extracted. By this way, we could hope to semi-automatically reconstruct 95% of the buildings.

REFERENCES

Baillard, C. and Zisserman, A., 1999. Automatic reconstruction of piecewise planar models from multiple views. In: Proceedings of 18th the Conference on Computer Vision and Pattern Recognition (CVPR'99), IEEE Computer Society, Fort Collins, CO.

Cord, M., Jordan, M. and Coquerez, J., 2001. Accurate building structure recovery from high resolution aerial imagery. Computer Vision and Image Understanding 82(2), pp. 138–173.

Fischler, M. and Bolles, R., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Graphics and Image Processing 24(6), pp. 381–395.

Flamanc, D., Maillet, G. and Jibrini, H., 2003. 3D city models: an operational approach using aerial images and cadastral maps. In: H. Ebner, C. Heipke, H. Mayer and K. Pakzad (eds), Proceedings of the ISPRS Conference Photogrammetric on Image Analysis (PIA'03), The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34, Institute for Photogrammetry and GeoInformation University of Hannover, Germany, Münich, Germany, pp. 53–58. ISSN: 1682-1750.

Heuel, S., Lang, F. and Forstner, W., 2000. Topological and geometrical reasoning in 3D grouping for reconstructing polyhedral surfaces. In: Proceedings of the XIXth ISPRS Congress, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 33, ISPRS, Amsterdam.

Jibrini, H., Paparoditis, N., Pierrot-Deseilligny, M. and Maitre, H., 2000. Automatic building reconstruction from very high resolution aerial stereopairs using cadastral ground plans. In: Proceedings of the XIXth ISPRS Congress, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 33, ISPRS, Amsterdam.

Paparoditis, N., Cord, M., Jordan, M. and Coquerez, J.-P., 1998. Building detection and reconstruction from mid-and high resolution aerial imagery. Computer Vision and Image Understanding 72(2), pp. 122–142.

Pierrot-Deseilligny, M. and Paparoditis, N., 2006. A multiresolution and optimization-based image matching approach: An application to surface reconstruction from SPOT5-HRS stereo imagery. In: Topographic Mapping From Space (With Special Emphasis on Small Satellites), ISPRS, Ankara, Turkey.

Taillandier, F., 2005. Automatic building reconstruction from cadastral maps and aerial images. In: U.Stilla, F.Rottensteiner and S.Hinz (eds), Proceedings of the ISPRS Workshop CMRT 2005: Object Extraction for 3D City Models, Road Databases and Traffic Monitoring - Concepts, Algorithms and Evaluation, Vol. 36/3/W24, Vienna, Austria. ISSN:1682-1777.

Taillandier, F. and Deriche, R., 2004. Automatic buildings reconstruction from aerial images: a generic bayesian framework. In: Proceedings of the XXth ISPRS Congress, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS, Istanbul, Turkey.

Vosselman, G. and Suveg, I., 2001. Map based building reconstruction from laser data and images. In: E. Baltsavias, A. Gruen and L. Gool (eds), Automatic Extraction of Man-Made Objects from Aerial and Space Images (III), A.A. Balkema Publishers, Centro Stefano Franscini, Monte Verità, Ascona, pp. 231–239.

AUTOMATIC EXTRACTION OF LARGE COMPLEX BUILDINGS USING LIDAR DATA AND DIGITAL MAPS

Jihye Park [a], Impyeong Lee [a, *], Yunsoo Choi [a], Young Jin Lee [b]

[a] Dept. of Geoinformatics, The University of Seoul, 90 Jeonnong-dong, Dongdaemun-gu, Seoul, Korea -
(jihye, iplee, choiys,)@uos.ac.kr
[b] Telematics ·USN Research Division, Electronics and Telecommunications Research Institute, 161 Gajeong-dong,
Yuseong-gu, Daejeon, Korea - yjinlee@etri.re.kr

Commission Ⅲ, WG Ⅲ/3

KEY WORDS: LIDAR data, Digital Map, Building Model, Primitives, Patches

ABSTRACT:

As the use of building models are rapidly increased for various applications, many studies have been performed to develop a practical and nearly automatic method to extract such models from various sensory and GIS data. Nevertheless, it is still a difficult problem to extract the models of large-complex buildings in particular. The purpose of this study is thus to develop a fully automatic method to extract the detail models of buildings from LIDAR data and a digital map. This extraction consisting of primitive extraction and modeling is mainly based on robust segmentation of planar patches from numerous LIDAR points. These primary primitives are used as the references to generate secondary primitives such as edges and corners and then refined based on these secondary primitives to form a complete polyhedral model. The proposed method was successfully applied to extracting large-complex buildings from real data in the test site. It can be a promising time- and cost-effective solution for a country to enhance their traditional map to include 3D models of buildings.

1. INTRODUCTION

The need of detail and realistic building models is rapidly increasing because of their intensive uses for various applications not limited to urban planning and redevelopment, three-dimensional car navigation systems, video games, and others areas.

To reconstruct 3D buildings, many studies based on various sensory data have been performed. For examples, using aerial images, Baillard and Zisserman [1] reconstructed polyhedral models using the edges between planar roof patches. The main idea is to obtain the half planes to the left and right of a detected dihedral line segment. The advantage is that only relatively local information is exploited (Brenner, 2003). In recent research, Suveg and Vosselman (2004) reconstructed buildings using aerial images and 2D ground plans. They generated the 3D volumetric primitives using the 3D corners extracted from 2D digital map and filtered by images. 75% of all objects were extracted using this method.

Building reconstruction from LIDAR data are very active these days. Rottensteiner and Briese (2003) extracted roof faces from DSM and derived the intersection and step edges from the regularized DSM. Additionally, images were used to detect small buildings. In recent studies for extracting the roof faces, Lodha and Kumar (2005) applied K-Mean algorithm to refine LIDAR points and to detect the planar roof faces. Since users should assign the number K indicating the number of point clusters, this approach is a semi-automatic method. Vosselman (1999) extracted roof faces from LIDAR points using a Hough transform. Vosselman and Dijkman (2001) improved this method by using ground plans in addition to LIDAR data. Brenner (1998) generated building models from LIDAR data and 2D ground plan using a heuristic algorithm.

Although many researchers have proposed semi-automatic (or automatic) methods, it is not yet solved to extract the detail models of large-complex buildings in particular in a fully automatic manner. In most cases, the modeling processes still have involved intensive manual editing steps and thus been thought to be time and money consuming.

The purpose of this study is thus to develop a fully automatic method to extract the detail models of buildings in particular with large-complex roof structure from LIDAR data and a digital map. This extraction includes two stages, *primitive extraction* and *modeling*. The most important step of this process is segmentation of planar patches. These primary primitives are used as the references to derive the secondary primitives such as edges and corners and refined to form each facet of the complex roof structure.

2. OVERVIEW AND PREPROCESSING

2.1 Overview of the Proposed Method

As the framework shown in Figure 1, the proposed building modeling approach include three main stages, that is, preprocessing input data, extracting building primitives, and generating building models. The inputs are airborne LIDAR data and the building layers of a digital map covering the same area. During the preprocessing stage, the LIDAR data are

[*] Corresponding author

registered to the digital map and the LIDAR points around and inside each building boundary are extracted. From these points are extracted the building primitives such as surface patches, edges, and corners. These primitives are then refined and grouped into a complete polyhedral building model.

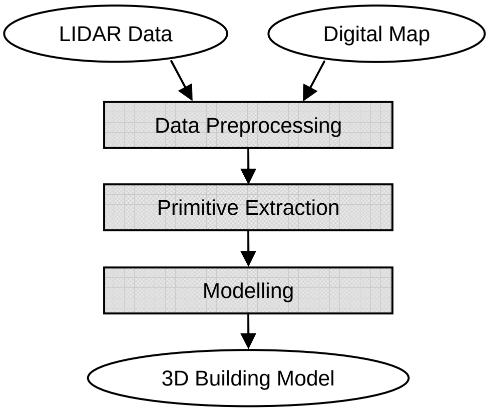


Figure 1. Framework for the building modelling approach

2.2 Input Data

LIDAR data consist of numerous three-dimensional points sampled from the terrain. Although these data provide very accurate elevations of the surfaces, they hardly retain the exact locations of corners and edges of objects in general because of relatively lower sampling density than images. This is a main reason why we also use the building boundary of a large scale map.

The test site is hilly district in Daejun Metropolitan city, Korea. As shown in an aerial image of Figure 2, the site includes many large buildings of various shapes and complex roof structures. The input LIDAR data of this site is shown in Figure 3. The point density is about 5.4 points/m^2. The input digital map is published by National Geographic Information Institute, Korea and the scale is 1/5,000. Figure 4 presents the building layer of this map. The aerial image in Figure 2 has not been used for the input but only for the verification of the modeling results.

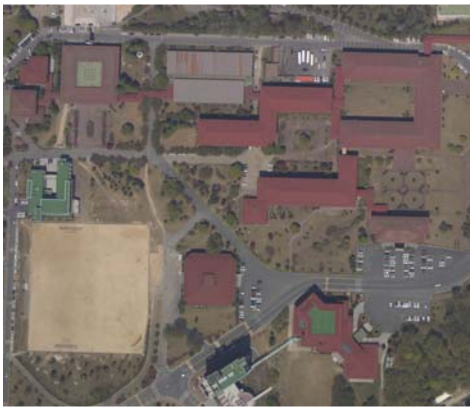


Figure 2. Aerial image of the test site

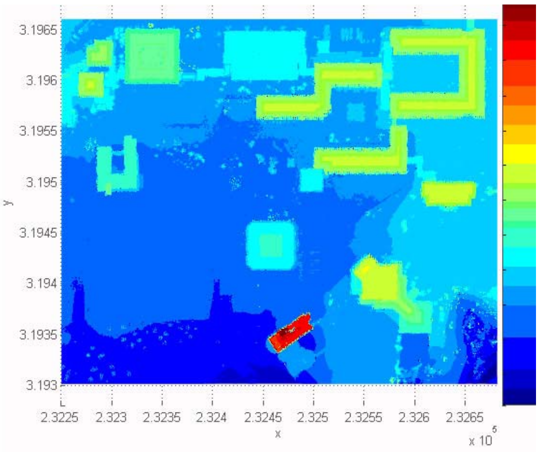


Figure 3. LIDAR data

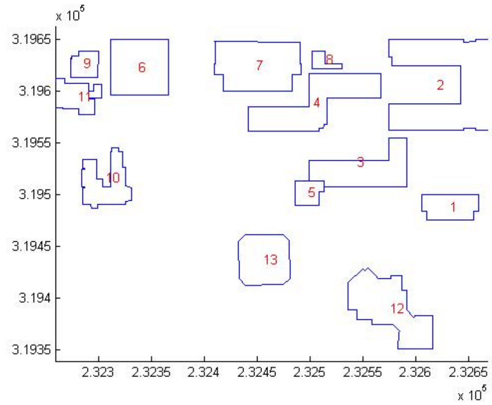


Figure 4. Digital map

2.3 Data Preprocessing

2.3.1 Calibration and outlier elimination: The LIDAR data were calibrated and verified to have better than the accuracy of ± 50 cm and ± 20 cm in horizontal and vertical positions, respectively. Some outliers in the data were detected and removed using the method based on the point density proposed by Moon et. al. (2005). The outlier ratio was found to be about 1 %.

2.3.2 Registration: The LIDAR data were then geometrically registered with the digital map. This registration was performed using the tie points manually selected from both data sets. These tie points mostly locate at the corners of buildings. Since the digital map provides only horizontal coordinates of the corners, only horizontal registration was possible using the 2D similarity transformation. Both data were registered with the precision of about ± 50 cm.

2.3.3 Points Extraction: Two sets of LIDAR points are extracted for each building boundary. One includes the points inside the boundary. We use these inside points to model the roof structure. The other includes the points locating outside the boundary with a distance of less than 5 m from it. The ground along the building can be derived from these outside points. Figure 5 shows the points extracted for all the buildings, where the blue and magenta dots indicates the inside and outside points, respectively.

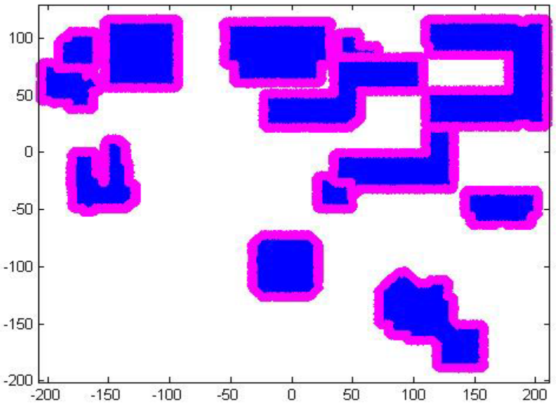


Figure 5. Result of points extraction

3. PRIMITIVE EXTRACTION

The primitives such as planar patches, edges and corners are extracted from the inside and outside points for each building. Planar patches are the *primary primitives* and the others are the *secondary primitives* that can be derived using the primary ones.

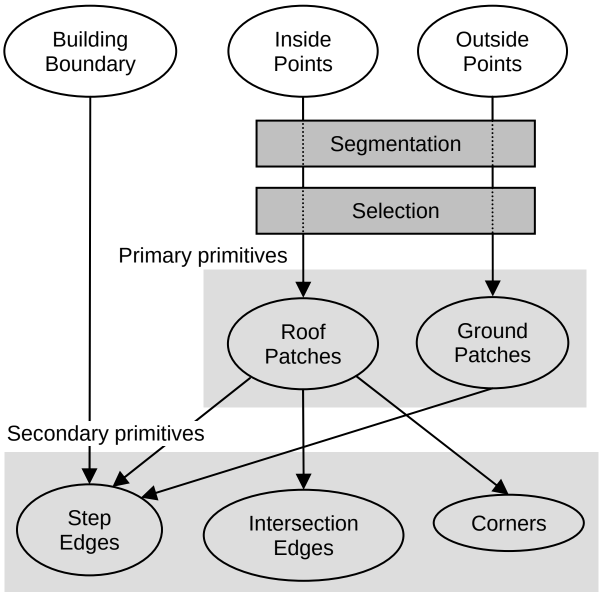


Figure 6. Primitive extraction process

3.1 Patch Adjacency and Connectivity Graph (PACG)

The adjacency and connectivity between all the primitives are examined and the results are stored into a graph structure called *patch adjacency and connectivity graph* (PACG). This graph incorporates each primitive into a node and any identified adjacency between two primitives is stored into an arc between them. Three types of adjacency are defined. The first one is *2D adjacency* created between two primitives if the horizontal distance between their boundaries is less than a given threshold, which is set to 2 m in this study. If the 2D adjacent primitives are also sufficiently close in 3D, *3D adjacency* is established between them. Otherwise, *2D only adjacency* is established. If any two adjacent primitives are verified to be actually connected, *connectivity* is additionally assigned. For example, if a patch is the nearest primitive adjacent to another patch and the intersection edge between them is adjacent to their boundary, they are confirmed to be connected.

3.2 Segmentation of Planar Patches

Planar patches are segmented from the inside and outside LIDAR points, respectively. This segmentation can be performed using the algorithm based on perceptual organization of numerous 3D points. This algorithm was originally proposed by Lee (2002) and summarized as follows:

The segmentation process starts with establishing the adjacency among the LIDAR point irregularly distributed in 3D space. A point cluster is constructed for every point by gathering a small number of points adjacent to the point. Each cluster is approximated to a plane. The clusters with relatively small fitting errors are selected as seed clusters, from each of which a planar patch is then growing with the adjacent points added to the cluster.

During the growing step, every adjacent point to the growing patch is tested about whether the point is statistically consistent with the patch. This growing process for a patch continues until no more adjacent point can pass this test. This grown point cluster called a patch is then verified by checking the size of the cluster and the fitting errors. For the verified patch, its boundaries are computed by determining the outlines of the point cluster using the alpha-shape algorithm (Edelsbrunner, 1983).

After segmentation, the set of points is converted into a set of patches, where each patch is expressed with the plane parameters, the boundary, and the fitting error considered as the roughness.

3.3 Selection of Planar Patches

Some of the patches from the inside points may be unreasonable to be parts of the roof structures. We thus selected as the roof patches only the patches satisfied with the following conditions:

1. The patch size is enough large.
2. Roughness of the patch is relatively low.
3. The shape of the patch is geometrically suitable.

In a similar way, we also select the ground patch among the patches segmented from the outside patches. The selection criteria are as follows:

1. The size of the patch is larger than any other patches.
2. The height of the patch is lower than any other patches.
3. The roughness of the patch is smaller than any other patches.

The adjacency between all the selected roof and ground patches are then examined by checking the distance between any pair of patches. All the patches with the identified adjacency are stored into the PACG.

For example, Figure 7 shows the roof and ground patches selected from the segmented patches for a building (ID: 2). As compared with its representation given in the aerial image and the LIDAR plots in Figure 2 and 3, the segmented and selected patches reasonably describe the roof structure and the ground around the building.

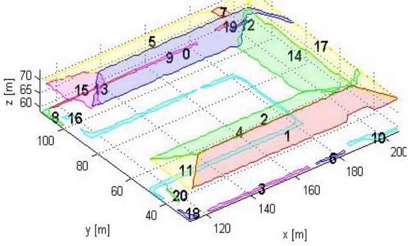


Figure 7. Roof and ground patches

3.4 Intersection Edges

Intersection edge can be derived from two 3D adjacent patches using the algorithm as follows:

1. Select a pair of adjacent patches identified from PACG.
2. Compute a straight line intersected by the planes.
3. Compute the distance between this line and the boundary of each patch.
4. If the distances become greater than a given threshold, discard the line.
5. Otherwise, determine the two ending points limiting the straight line.

To determine the ending points, we first identify the parts of the boundaries that are adjacent to the straight line and then project them to the straight line. The extreme two points of the range on the straight line in which the parts of boundaries are actually projected are selected as the ending points. The straight segment limited by these points is called intersection edge. This edge is also stored into the PACG with the connectivity assigned to the two adjacent patches. Examples of these edges are shown in Figure 8.

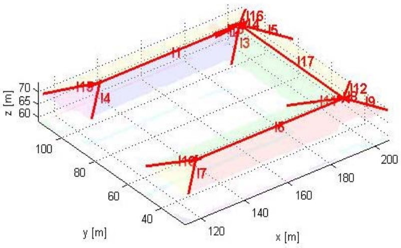


Figure 8. Intersection edges

3.5 Corners

If three patches are identified to be adjacent each other based on the PACG, a corner can be derived from them. With the three planes, an intersection point is computed. If this point is also adjacent to the three patches, it is confirmed as a corner. This corner is also stored into the PACG with the connectivity assigned to the three adjacent patches. Figure 9 shows the derived corners.

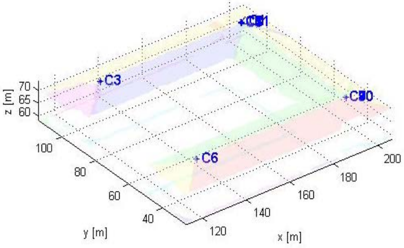


Figure 9. Corners

3.6 Step Edges

Step edges mean the parts of building outlines showing abrupt change in elevation across the edges, for examples, the outlines of vertical walls. These edges are mainly observed along the building boundary provided by the digital map but sometimes inside the roof structure within the boundary. The edges along the boundary are derived by projecting the 2D building boundary to the nearest adjacent roof patches and ground patches. In addition, we derive the edges within the roof structure from any pair of 2D only adjacent patches identified from PACG. The derived step edges are also stored into the PACG with proper connectivity and adjacency assigned. Figure 10 shows examples of the derived step edges.

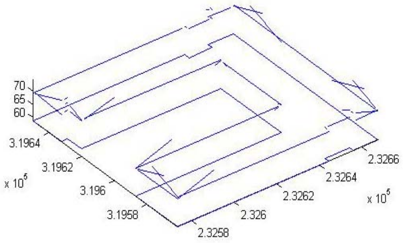


Figure 10. Step edges

4. MODELING

The derived roof patches as the primary primitives has also important roles in the modeling process shown in Figure 11. Their boundaries are refined using the secondary primitives such as the corners, the intersection edges, and the step edges. The refined roof patches are then called *roof facets*. The space between each pair of the step edges connected in 2D can be filled with a vertical planar patch, dedicated to *wall facets*. The roof and wall facets constitute the final polyhedral model.

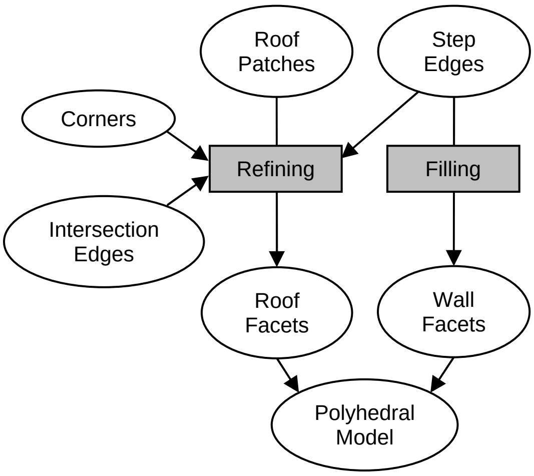


Figure 11. Modelling process

4.1 Roof Facets

Since the boundary of each roof patch is formed by the outer points of the patch, it is so rough that the patch could not be used as a roof facet directly. It is thus necessary to refine the boundaries of roof patches using the secondary primitives located with better accuracy in general.

This refining algorithm is illustrated with a simple example in Figure 12 and summarized as follows:

1. Select a roof patch.
2. Select an edge forming the boundary called a boundary edge.
3. Find the nearest one among the secondary primitives to be connected to this edge from the PACG.
4. If this is a corner, then the nearest point on the boundary edge is changed to the corner.
5. If this is an intersection or step edge, then project the boundary edge to the edge.
6. Repeat 2 to 5 until all the boundary edges will be refined.
7. Repeat 1 to 6 until all the patches will be refined.

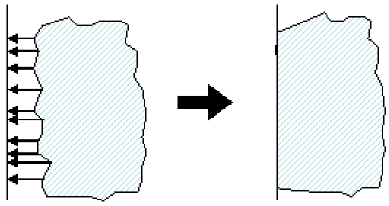


Figure 12. Refining roof patch with an edge

4.2 Wall Facets

Wall facets should be generated hypothetically because the vertical faces of an object are hardly observed from LIDAR data. Only the horizontal locations of the building outlines are accurately provided by the digital map. The step edges regardless of being derived from the building boundary of a map or from the patches connected in 2D indicates the existence of vertical facets of themselves. We thus derive a vertical patch between a pair of step edges to fill the gap between them with this patch. Figure 13 shows examples of the wall facets generated based on this method.

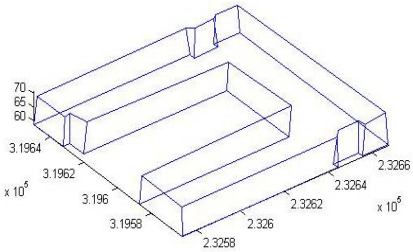


Figure 13. Wall facets

4.3 Polyhedral Model

The derived roof and wall facets are grouped into a polyhedral model. Among these facets, those connected to each other in particular share some edges and corners. Such redundant shared edges and corners are unified. Finally, the polyhedral model is examined with a topological test to check the completeness of the model. Inconsistency found among the facet, edges, and corners of a model indicates the existence of gaps in the model. These gaps are just identified so that they can be later edited if necessary. Figure 14 shows an example of the final polyhedral model (building 2).

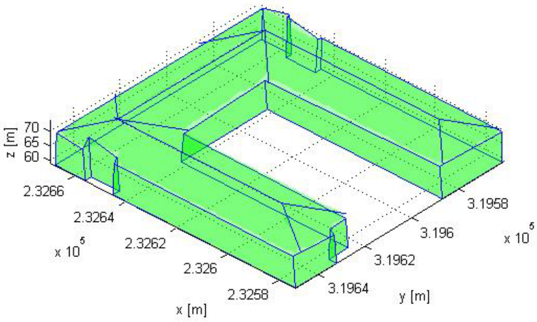


Figure 14. Polyhedral model

5. EXPERIMENTAL RESULTS

The proposed extraction approach was implemented as a program coded using C++ with standard template library. This program was applied to extracting building models from the input data of the test site mentioned in Section 2.2. The modeling results from the 13 buildings existing in this site are presented in Figure 15. They are the results from fully automatic processes without any manual intervention during this process or any manual editing after it. All the buildings presented from the map mainly retaining large and complex roof structures were verified to be reasonably modeled with visual inspection.

We inspected each building model based on its appearance in the aerial image, the LIDAR point plot, and the digital map. The inspection results of three buildings (ID: 4, 12, 10) are presented as follows.

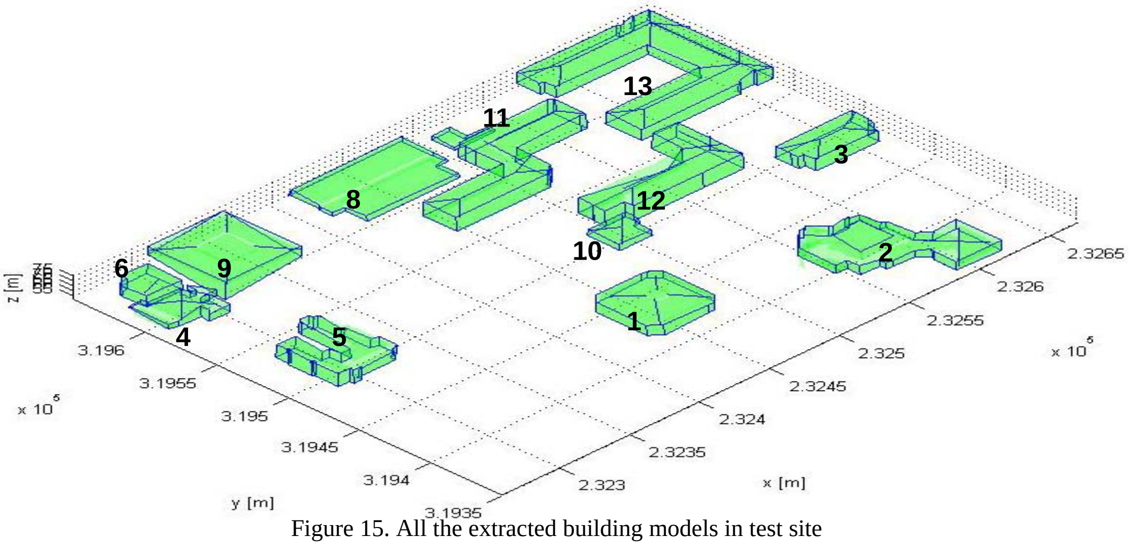


Figure 15. All the extracted building models in test site

Figure 16 shows the model of building 4 retaining the width of about 125 m, the length of about 56 m, and the height of 18 m. The generated model represents the winding of the wall around position A, which indicates the modeling process recovers narrow vertical facets.

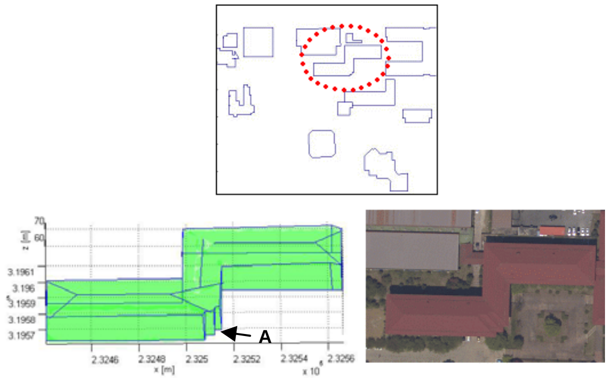


Figure 16. Extracted model of building (ID: 11)

The next building (ID: 2) is the most complicated shape. This building has many step edges and intersection edges on the roof. In addition, the ground plan is also irregular. Although the shape of the building is so peculiar that it might be difficult to apply a traditional model-based approach, the extracted model completely describes the shape in detail, as shown in Figure 17.

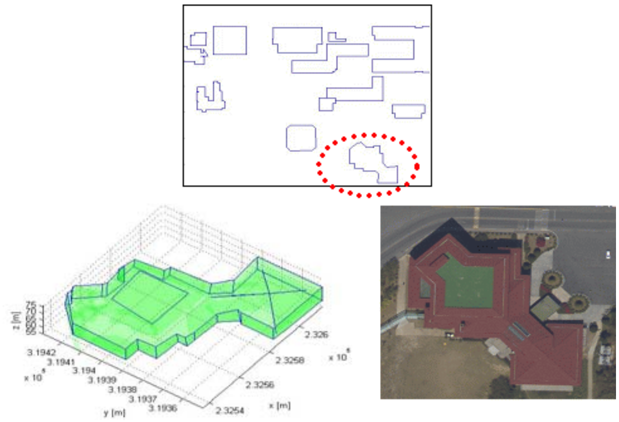


Figure 17. Extracted model of building (ID: 2)

Building 5 also has very complex roof structure in which many step edges are observed. The roof structure is modeled with 16 patches of various shape and size. The size of the smallest one is just 4 m^2, indicating how detail the proposed process can model a building.

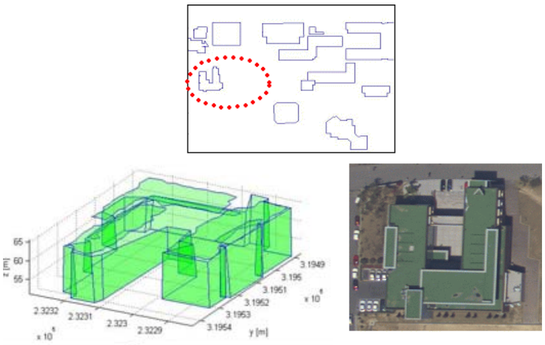


Figure 18. Extracted model of building (ID: 5)

6. CONCLUSION

We proposed an automatic method to extract three-dimensional detail models of buildings in particular with large-complex roof structure from LIDAR data and a digital map. From the modeling results of 13 buildings in the test site, the proposed method is verified to successfully extract the detail polyhedral models.

Most countries have already constructed large-scale maps including building layers. If they improve these maps to include 3D models of buildings for various applications such as 3D car navigation, the proposed method can be a time- and cost-effective solution.

Buildings newly built after a map being generated cannot be identified from the map. In order to model them in addition, we are studying a method to identify the existence of building and generate the step edges along the building only from LIDAR data.

REFERENCE

Baillard, C. and A. Zisserman, Automatic reconstruction of piecewise planar models from multiple views. In: *Proc. IEEE Conference on Computer Vision Pattern Recognition*, pp. 559–565.

Brenner, C. 2003. Building reconstruction from images and laser scanning. In: *Proc. ITC Workshop on Data Quality in Earth Observation Techniques*, Enschede, The Netherlands.

Brenner, C., 1998. Rapid acquisition of virtual reality city models from multiple data sources. In: *Int. Arch. Photogramm. Remote Sensing*, Vol. 32, Part 5, pp. 323-330.

Edelsbrunner, H., D. G. Kirkpatrick, and R. Seidel, 1983. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, Vol. 29, No. 4, pp. 551-559.

Lee, I., 2002. *Perceptual Organization of Surfaces*, Ph. D. Dissertation, The Ohio State University, Columbus, Ohio, USA.

Lodha S. K., K. Kumar, and A. Kuma, 2005. Semi-automatic roof reconstruction from aerial LIDAR data using K-means with refined seeding, In: *ASPRS Conference*, Baltimore, Maryland.

Moon J., I. Lee, S. Kim, and K. Kim, 2005. Outlier Detection from LIDAR Data based on the Point Density. *KSCE journal*, 25(6D), pp. 891-897.

Rottensteiner, F and C. Briese, 2003. Automatic Generation of Building Models from LIDAR Data and the Integration of aerial image, In: *Int. Arch. Photogramm. Remote Sensing*, Vol. 34.

Suveg, I. and G. Vosselman, 2004. Reconstruction of 3D models from aerial images and maps. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3-4), pp. 202-224.

Vosselman, G. and Dijkman, 2001. 3D building model reconstruction from point clouds and ground plan, In: *Proceedings of the ISPRS Workshop on Land Surface Mapping* *and Characterization Using Laser Altimetry*, Annapolis, MD, pp. 37-44.

EXTRACTION OF FAÇADES USING RJMCMC AND CONSTRAINT EQUATIONS

Claus Brenner and Nora Ripperda

Institute of Cartography and Geoinformatics, University of Hannover, Germany
{Claus.Brenner, Nora.Ripperda}@ikg.uni-hannover.de

KEY WORDS: Markov Chain, constraint equations, façade modelling, building extraction, least squares adjustment.

ABSTRACT:

Today's processes to extract man-made objects from measurement data are quite traditional. Often, they are still point based, with the exception of a few systems which allow to automatically fit simple primitives to measurement data. At the same time, demands on the data are steadily growing. The need to be able to automatically transform object representations, for example, in order to generalize their geometry, enforces a structurally rich object description. Likewise, the trend towards more and more detailed representations requires to exploit structurally repetitive and symmetric patterns present in man-made objects, in order to make extraction cost-effective. In this paper, we address the extraction of building façades in terms of a structural description. As has been described previously by other authors, we use a formal grammar to derive a structural façade description in the form of a derivation tree. We introduce two new concepts. First, we use a process based on reversible jump Markov Chain Monte Carlo (rjMCMC) to guide the application of derivation steps during the construction of the tree. Second, we attach variables and constraint equations to the symbols of the grammar, so that the derivation tree automatically leads to a constraint equation system. This equation system can then be used to optimally fit the entire façade description to given measurement data.

1 INTRODUCTION

The extraction of man-made objects from sensor data has a long history in research (Baltsavias, 2004). Especially for the modelling of 3D buildings, numerous approaches have been reported, based on monoscopic, stereoscopic, multi-image, and laser scan techniques (Brenner, 2005). While most of the effort has gone into sensor-specific extraction procedures, very little work has been done on the structural description of objects.

Nowadays, in extraction systems, one can choose between boundary representation (BRep) and constructive solid geometry (CSG) modelling. BRep modelling is inspired by traditional photogrammetric point measurement, with subsequent topology definition to obtain lines, surfaces, and volumes. CSG, on the other hand, models objects by combining predefined volumetric primitives using Boolean operations. Thus, it has the intrinsic potential to attach meaning to the primitives and to obtain a structural description in terms of a CSG modelling tree. However, primitives usually reduce to simple geometric shapes such as planar patches, cylinders and spheres, and the CSG tree is often derived according to the modelling process and the desired 3D shape rather than with a semantic modelling of the building structure in mind.

Modelling structure though is very important for downstream usability of the data, especially for the automatic derivation of coarser levels of detail (LoD) from detailed models (a process called generalization). Being able to deliver different LoDs tailored to different customers needs, to context-adapted visualizations, such as on mobile displays, or simply to cut down rendering time of large models is essential for 3D models to enter the market. The Sig3D group has defined five levels of detail for building models (Kolbe et al., 2005). However, the definition of discrete LoDs alone does not imply any path to derive one level from the other in an automated way. Experience from 2D map generalization in cartography shows that generalization purely based on geometric information is indeed a hard problem, which becomes even worse in 3D.

Representing structure is not only important for the later usability of the derived data, but also as a means to support the extraction process itself. A fixed set of structural patterns allows to span a certain subspace of all possible object patterns, thus forms the model required to interpret the scene. Patterns can also guide the measurement process (taking place after the interpretation). Especially for man-made structures such as building façades, a large number of regularity conditions hold, which can be introduced into the measurement process as constraints. In interactive measurement processes, introducing structural descriptions can cut down acquisition time, since repeated or mirrored parts can be introduced in one step.

This paper elaborates on the grammar-based extraction of façade descriptions. The grammar is used in two places. First, it guides the generation of possible façade layouts using a reversible jump Markov Chain Monte Carlo (rjMCMC) process to explore solution space. Second, the obtained derivation tree is used for the automatic setup of constraint equation systems during the fine matching of the generated façade layout to measurement data.

2 RELATED WORK

2.1 Extraction of objects using constraints

The extraction of objects from measurement data is different from computer aided design (CAD) construction. In CAD, the general problem is to derive an instance (a geometrical instantiation) of an object, given a sketch (or just an idea), annotated with dimensional information. Algebraically, sketch annotations are constraints and the sketch defines a constraint graph, out of which a constraint equation system

$$\boldsymbol{f}(\boldsymbol{x}) = \mathbf{0} \qquad (1)$$

results, where $\boldsymbol{x}$ is the parameter vector describing the (geometry of the) solution. Finding $\boldsymbol{x}$, given (1), is termed *geometric constraint solving*. In order to obtain a finite set of solutions, $\boldsymbol{f}$ must be well constrained or consistently overconstrained. In contrast, when objects are reconstructed using measurements, the task can typically be formulated as

$$\begin{aligned} \|\boldsymbol{g}(\boldsymbol{b}, \boldsymbol{x})\| &\overset{!}{=} \min, \quad \text{subject to} \\ \boldsymbol{f}(\boldsymbol{x}) &= \boldsymbol{0}, \end{aligned} \tag{2}$$

where $\boldsymbol{g}$ subsumes the (possibly contradictory) constraints imposed by some measurement data $\boldsymbol{b}$, whereas $\boldsymbol{f}$ represents the "hard" constraints imposed by the model. As opposed to the case in CAD, $\boldsymbol{f}$ will be normally underconstrained (as else the measurements will have no effect on the solution), whereas $\boldsymbol{g}$ will be typically overconstrained (since redundant measurement data is used), which leads to a system which is both locally overconstrained and globally well- or underconstrained.

There are no extraction tools which implement (2) rigorously. For example, modelling of objects from close range scan data is usually carried out using CAD-based systems which combine CAD modelling functionality with the ability to fit CAD objects to point clouds (e.g., (Leica Geosystems, 2006)). In this case, the first part of (2) is implemented, but not the second one. For a practical example, assume that four best-fit planar patches have been extracted from laser scan data. Then, it is not possible to make them meet in a single point except by manual modification (usually, a "snap" operation) of one of the planes – which destroys the initial best-fit property.

The need to introduce constraints into the reconstruction process of man-made objects has been recognized early. For example, Weidner extracts roof faces using a DSM segmentation and proposes to automatically derive mutual relationships between the extracted faces, such as 'same slope', 'symmetry', and 'antisymmetry', in order to insert them as constraints into a global robust adjustment (Weidner, 1997). Although this has been proposed by several authors, constraint-based extraction does not play a role nowadays, except for research systems (Ermes, 2000).

The major problems with constraint-based modelling are *(i)* to insert the constraints in a meaningful manner, *(ii)* to manage, introspect, and debug large constraint equation systems, and *(iii)* to solve constraint equation systems. As opposed to the classical geometric constraint solving problem, which attempts to build a solution "from scratch", in reconstruction, initial values are usually available, so that linearization and iterative estimation can be used for solving the equation system. Thus, the main task lies in the structured insertion and management of constraints. To facilitate this in interactive environments, "weak primitives" have been proposed in (Brenner, 2004). The concept has been extended later to include hierarchical structures using containers (Brenner and Sester, 2005).

2.2 Generalization and incremental modelling

Automation of (manual) map generalization procedures has been a topic in cartography for several decades. There are now first operational systems available, which usually start from a scene description in form of 2D primitives like polygons or polylines. From this, implicit relationships are discovered, such as adjacency, parallel and rectangular structures, distances, protrusions, etc., which are to be modified or preserved during generalization. The final outcome is again a description of the objects in terms of their geometry only. Since the discovered structures are not being made explicit, they cannot be modified, which frequently leads to the need to check and correct the outcome of the automatic generalization step manually.

Recently, in cartography methods are being investigated and developed which aim at the recognition of important structures that are needed as a basis for generalization, e.g. parallelism, linear arrangement, clusters (Christophe and Ruas, 2002, Anders and Sester, 2000). Furthermore, there are approaches which try to separate generalization processes related to different objects in different hierarchical levels, e.g. when defining generalization modules that can be handled independently (Kilpeläinen and Sarjakoski, 1995). A first attempt to explicitly model these structures has been done in the AGENT project, where different hierarchical levels of objects have been specified that can act independently with a specific dedicated behavior (Lamy et al., 1999).

In (Brenner and Sester, 2005), the previously mentioned approach of primitives and containers has been extended to include discrete behavior. Primitives are defined as the combination of geometric description (e.g., polygons), sets of constraints (e.g., all line segments aligned horizontally or vertically), and discrete behavior (e.g., boundary simplification rules). Containers provide the ability to spatially layout primitives, with dedicated interface slots which allow to connect primitives to containers. This leads to a simple hierarchical description scheme, which is extended in this paper to a grammar-based description.

2.3 Modelling of architectural patterns

Grammars have been extensively used to model structures. For modelling plants, Lindenmayer systems were developed by the biologist Aristid Lindenmayer (Prusinkiewicz and Lindenmayer, 1990). They have also been used for modelling streets and buildings (Parish and Müller, 2001, Marvie et al., 2005). However, Lindenmayer systems are not necessarily appropriate for modelling façades. Façades differ in structure from plants and streets, since they don't grow in free space and modelling is more a partition of space than a growth-like process.

For this reason, other types of grammars have been proposed for architectural objects. Stiny introduced shape grammars which operate on shapes directly (Stiny and Gips, 1972). The rules replace patterns at a point marked by a special symbol. Mitchell describes how grammars are used in architecture (Mitchell, 1990). The derivation is usually done manually, which is why the grammars are not readily applicable for automatic modelling tools.

Wonka et al. developed a method for automatic modelling which allows to reconstruct different kinds of buildings using one rule set (Wonka et al., 2003). The approach is composed of a split grammar, a large set of rules which divide the building in parts, and a control grammar which guides the propagation and distribution of attributes. During construction, a stochastic process selects among all applicable rules.

Dick et al. introduce a method which generates building models from measured data, i.e. several images (Dick et al., 2004). This approach is also based on the rjMCMC method. In a stochastic process, 3D models with semantic information are built.

3 GRAMMAR-BASED FAÇADE RECONSTRUCTION

In this section, the basic concept of our method is described. As in the approaches outlined in the previous section, we use a grammar to define façade layout. However, we do not want to generate artificial façade descriptions, but rather derivation trees which correspond to measurement data. Two major tasks can be identified:

1. the recognition of the façade structure, i.e., building of a structural description in the form of a derivation tree, together with a first instantiation of all (geometric) parameters, and

2. measurement, i.e., fine-matching the geometry of this initial structure to the measurement data.

The first task is the interpretation step, for which we describe an approach that uses rjMCMC to explore different derivation trees. As for the second task, we propose to attach constraint equation systems to the derivation rules such that a complete derivation tree not only defines the structure and initial layout, but also a set of constraints which allow to precisely match the structure to the measurement data.

For our experiments, we use terrestrial laser scan data and images. For the moment, we concentrate on façades, i.e., the measurement data consists of point clouds and orthorectified images of single façades.

3.1 Façade grammar

The façade model is described in terms of a recursive partition of space. Each part is represented by one of the symbols listed in table 1 and 2. There are two kinds of symbols, the first one being nonterminals (table 1). Geometrically, nonterminals do not represent façade geometry directly but serve as containers which hold other objects, represented in the derivation tree by nonterminal or terminal children. The second group contains the terminal symbols, which represent façade geometry and cannot be subdivided further (table 2).

AboveDoor	IdenticalFaçadeArray
AboveWindow	PartFaçade
Façade	StaircaseColumn
FaçadeArray	SymmetricPartFaçade
FaçadeColumn	SymmetricPartFaçadeMiddle
FaçadeElement	SymmetricPartFaçadeSide
FaçadeRow	SymmetricFaçade
Gable	SymmetricFaçadeMiddle
GroundFloor	SymmetricFaçadeSide

Table 1: Nonterminal symbols corresponding to containers.

Door	Wall
DoorArch	Window
StaircaseWindow	WindowArch

Table 2: Terminal symbols corresponding to façade geometry.

The start symbol is the symbol Façade. Starting from it, the model can be expressed as a derivation tree with Façade as root. The subdivision is made by rules similar to the ones introduced by (Wonka et al., 2003). Figure 1 shows an example façade. The Façade can be partitioned into GroundFloor and upper parts of the building, modelled as PartFaçade. PartFaçade shows symmetry and therefore only one side is modelled as SymmetricPartFaçadeSide. In this part the windows are arranged in a regular grid modelled by an IdenticalFaçadeArray. This array can be instantiated with a single Window which is placed at each grid position. The GroundFloor doesn't show any regularities which is why it is subdivided into FaçadeElements which can contain Windows or Doors. Each rule has a left side which consists of one symbol and a right side which may comprise several symbols in a certain spatial layout. The result of the method is a derivation tree which describes the model of the façade.

3.2 Exploration of the derivation tree using rjMCMC

We use rjMCMC for the construction of the derivation tree. The tree is encoded in a vector θ, which holds all parameters which are present in the derivation tree, e.g. positions and sizes of terminal symbols. The task is to find the optimum value for θ, given measurement data. In terms of a distribution, we are therefore looking for the maximum (mode) of the distribution $P(\theta|D_S D_I)$, i.e., the conditional distribution of θ, given scan data D_S and image data D_I. Finding this maximum by an exhaustive search is not feasible, due to the dimension of θ. Therefore, we use a stochastic method to instantiate the value of θ randomly. The overall approach is thus of the type hypothesize-and-test, where the hypotheses are generated randomly and tested afterwards, using measurement (scan and image) data. In order to be feasible, the samples θ are drawn from the distribution $P(\theta|D_S D_I)$, so that more samples are in the vicinity of high distribution values (i.e., close to probable façade layouts). The problem with this is that first, $P(\theta|D_S D_I)$ usually has a highly complex shape, far from a standard distribution, so drawing samples is nontrivial. Second, $P(\theta|D_S D_I)$ is not analytically available. The first problem is solved using Markov Chain Monte Carlo (MCMC, see e.g. (Gilks et al., 1996)). Basically, using the algorithm of Metropolis-Hastings, a Markov chain is obtained which converges to the desired distribution. Thus, after an initial phase, the algorithm delivers samples drawn from the distribution $P(\theta|D_S D_I)$. As for the second problem, using Bayes' law, $P(\theta|D_S D_I) \propto P(\theta)P(D_S D_I|\theta)$. The first term (prior) is evaluated using plausibility functions, which are set up manually. For example, one part of $P(\theta)$ describes assumptions about window sizes (by assuming a distribution). The second term (likelihood function) is evaluated by a score function based on the model (defined by θ) and scan and image data. The realization of both terms is described in more detail below. Thus, to summarize, the method explores the solution space by drawing samples from a (posterior) distribution, without the need to know this distribution analytically. Since the derivation tree changes during the process, the dimension of θ changes as well, and MCMC is not directly applicable. To resolve this, rjMCMC is used, which allows jumps between spaces of different dimension (Green, 1995). Our approach is described in more detail in (Ripperda and Brenner, 2006).

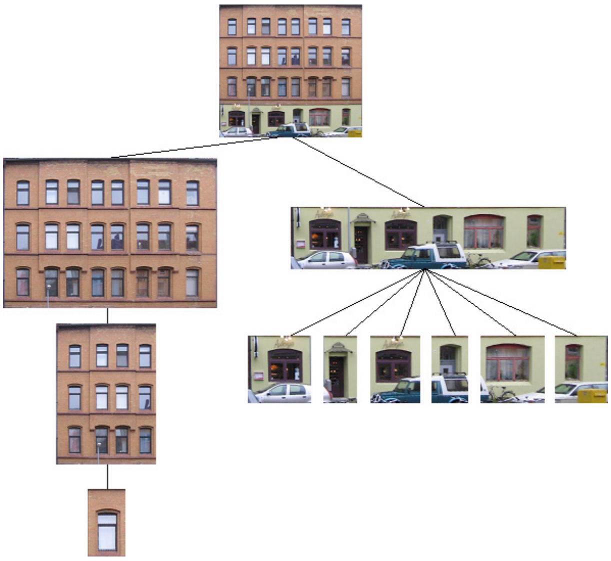


Figure 1: Example partition of a façade.

During the exploration of the derivation tree, any state change can be assigned to one of the following categories:

- Application of a split rule from the grammar. Façade elements are divided horizontally, vertically or in both directions and each part becomes a new symbol (see Fig. 2). In fact, one grammar rule comprises a set of changes to the parameter vector θ, since the associated attributes have to be chosen, such as the number and size of children. Figure 3 shows an example where one rule splits the symbol Façade into FaçadeColumns. The number of columns and their width is determined randomly. If a Façade can be divided into several FaçadeColumns the general rule stands for all rules of this kind with different number of columns and different positions.

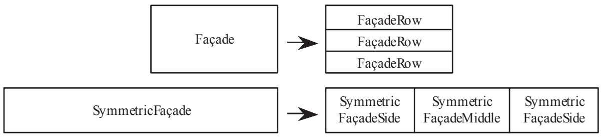


Figure 2: Split rules.

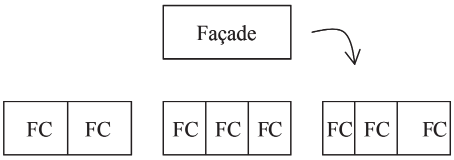


Figure 3: Different applications of a split rule.

- Changes in structure. Even after derivation of new containers according to the previous step, a second set of state changes allows to modify parameters, e.g. the number of columns or the position of the parting lines between columns (see Fig. 4). The same can be done starting from a child symbol. In this case, the neighbor symbols which are involved in the change have to be changed as well.

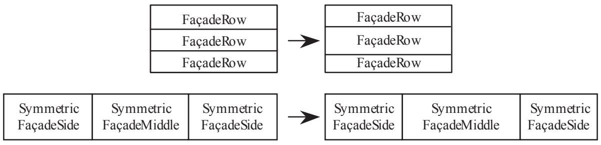


Figure 4: Changes which modify splits.

- Replacement of symbols. This allows to interchange one symbol in the derivation tree by another symbol. In this case, the geometry stays the same, but the denotation changes. This is especially used in the case of the symbols AboveDoor and AboveWindow. For example, the space above a window is modelled by the symbol AboveWindow. The rules

$$\text{AboveWindow} \rightarrow \text{WindowArch}$$
$$\text{AboveWindow} \rightarrow \text{Wall}$$

allow to replace this symbol.

The control is done by the rjMCMC method. To ensure the reversibility, each change can be applied from left to right and vice versa. This is a difference to the way split grammars are used, but is a requirement for the rjMCMC approach. A change is proposed depending on the jumping distribution $J_t(\theta_t|\theta_{t-1})$ which expresses the likelihood for each change.

For the evaluation of changes, we use different methods which can be divided into two groups. The first group contains methods which test the general plausibility of the model of the façade. In the group there are methods which test how good the model fits the data. This group subdivides in methods working with range data and methods working with image data. In any case, the evaluation functions return a probability which is used to decide if the change is accepted or rejected.

The general plausibility depends on the alignment, the extent and the position of the façade elements. Windows are usually arranged in rows and columns. Therefore, such layouts are assigned a high acceptance probability. We consider the size and the aspect ratio of façade elements to rate their probability. We also use the size for the rating of the subdivision into rows, columns or arrays. A row which is five meters high is not very likely and thus has a low acceptance probability. The last general criterion is the position of the elements. A door in the third floor is not very likely, so only doors in the ground floor are assigned a high probability.

To evaluate the match of the data to the model, scan and image data are used. In the first case, the fact that window points typically lie behind the façade is exploited. In the second case, color difference has been used since windows typically appear darker than the surrounding façade. In both cases, the information is used for the subdivision into rows, columns, and arrays as well. For example, upon division into rows, the resulting row strips are correlated to obtain an acceptance probability. Additionally, in image data a color change may indicate a changeover of ground floor and first floor.

Fig. 5 and 6 show the partition of a symmetric façade and the corresponding derivation tree. The symbol Façade is replaced by SymmetricFaçade. SymmetricFaçade is split into SymmetricFaçadeSide and SymmetricFaçadeMiddle. Each one is further subdivided into IdenticalFaçadeArray and FaçadeElements, respectively. Window and Door are on the leaf level.

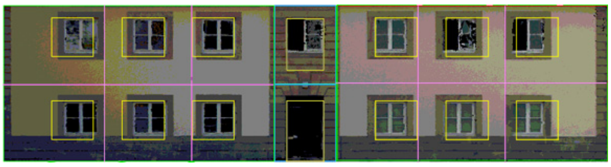


Figure 5: Resulting partition of a façade.

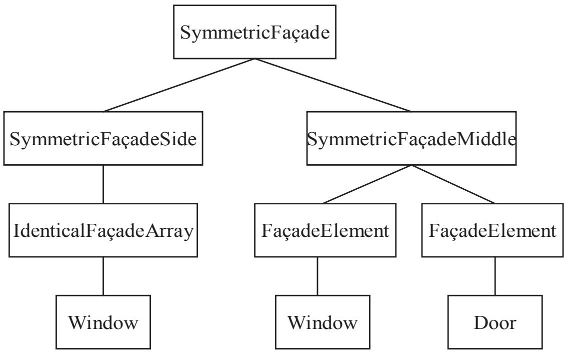


Figure 6: Derivation tree of the façade shown in figure 5.

3.3 Introduction of constraints

In 2D, with points represented by $\boldsymbol{p} = (x_1, y_1)^{\mathbf{T}}, \boldsymbol{q} = (x_2, y_2)^{\mathbf{T}} \in \mathbb{R}^2$ and lines by $\boldsymbol{l} = (a_1, b_1, c_1)^{\mathbf{T}}, \boldsymbol{m} = (a_2, b_2, c_2)^{\mathbf{T}}$ (in Hesse normal form $ax + by + c = 0$), typical logic constraint equations are $a_1^2 + b_1^2 - 1 = 0$ ($\boldsymbol{l}$ having a unit length normal vector), $a_1 x_1 + b_1 y_1 + c_1 = 0$ ($\boldsymbol{p}$ incident $\boldsymbol{l}$), $a_1 a_2 + b_1 b_2 = 0$ ($\boldsymbol{l}$ perpendicular $\boldsymbol{m}$), $a_1 b_2 - a_2 b_1 = 0$

($\boldsymbol{l}$ parallel $\boldsymbol{m}$), whereas dimensional equations include $a_1x_1 + b_1y_1 + c_1 - d = 0$ ($\boldsymbol{p}$ having (signed) distance d from $\boldsymbol{l}$), $(x_1 - x_2)^2 + (y_1 - y_2)^2 - d = 0$ ($\boldsymbol{p}$ having Euclidean distance d from $\boldsymbol{q}$), $a_1a_2 + b_1b_2 - \cos\varrho = 0$ and $a_1b_2 - a_2b_1 - \sin\varrho = 0$ (two oriented lines $\boldsymbol{l}$ and $\boldsymbol{m}$ enclosing the fixed angle ϱ). Thus, constraints between objects often result in bilinear equations. For solving those constraints, linearization and least squares estimation can be used. As noted earlier, the main problem is to introduce constraints in a sensible way so that they are manageable and constraint dependencies are minimized.

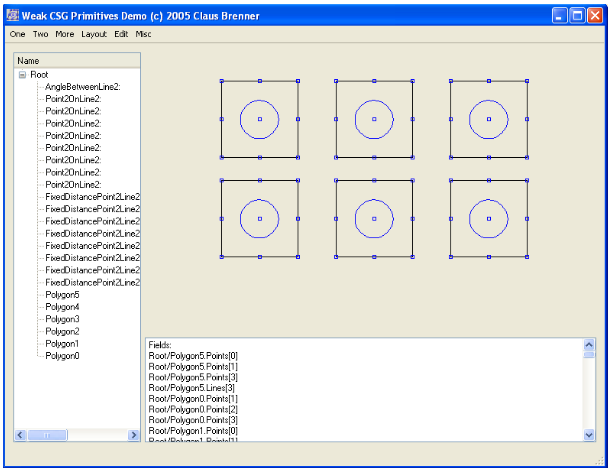
We use the derivation tree to define the set of constraints automatically. Two types of constraints can be generated from this tree. Terminal symbols represent geometry, which is fitted to measurement data. Thus, terminal symbols can generate fitting constraints, depending on the measurement data type, e.g. least squares fitting of surfaces to laser scanner data, or fitting of edges to the orthorectified image. Nonterminal symbols, on the other hand, can introduce constraints between their children, such as alignment, size, or orientation.

As an example, Fig. 7 shows a derivation tree (as obtained by the grammar), the corresponding geometric representation, and the generated unknowns and constraints. IDENTICALFAÇADEARRAY, as seen by the grammar, subdivides space into a regular array (depicted here as 2x3 array). From a unknowns/ constraints viewpoint, IDENTICALFAÇADEARRAY introduces column alignment lines at x_1, x_2, x_3 and row alignment lines at y_1, y_2. As IDENTICALFAÇADEARRAY enforces a regular column spacing, a constant distance Δx together with constraint equations $x_{i+1} - x_i = \Delta x$ is introduced. Since IDENTICALFAÇADEARRAY enforces identical sizes as well, width w and height h variables are introduced. All variables are inherited, i.e., the FAÇADEELEMENT shown in the figure receives the relevant alignment variables x_3 and y_1 as well as w and h. WINDOW is a weak primitive p and thus consists of geometry and internal constraints. To the outside, it offers variables $p.c_x$, $p.c_y$ (the center), $p.w$ (width), $p.h$ (height) in the form of fields (slots). Those fields are connected to the inherited variables x_3, y_1, w, h by the addition of four constraints. Being a terminal symbol, WINDOW represents a "real" geometry. Thus, additional constraints are added which match the geometry of WINDOW to the measurement data.

In contrast to the approach in (Wonka et al., 2003), the distinctive feature of our approach is that we do not "copy" attribute values down the derivation tree, but rather distribute (symbolic) variables. These variables can be used by children in arbitrary complex ways by introducing constraint equations. By the distribution of variables and the link by constraints, the geometric representation of the tree is "alive" in the sense that changes in one place can propagate across the entire tree. Finally, mapping the tree to a constraint equation system and subsequent solution of that system in the least squares sense allows a mathematically thorough, well-defined solution, which seamlessly integrates observations and constraints. To experiment with constraint equation systems in 2D, we have developed an environment which allows the interactive modification of geometric items while geometric constraints are enforced using least squares estimation (Fig. 8).

4 CONCLUSIONS AND OUTLOOK

In this paper, we have proposed to use grammars for the extraction of façade descriptions from measurement data. We introduced two major concepts. First, the use of rjMCMC to guide the construction of the derivation tree, in conjunction with evaluation functions which rate possible changes based on measurement data. Second, the use of the hierarchic derivation tree structure as a means to automatically establish constraint equations for a subsequent least-squares fitting of the façade description to the measurement data. For the future, we plan to enlarge our set of derivation rules as well as to improve our evaluation functions.



Root
- AngleBetweenLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- Point2OnLine2:
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- FixedDistancePoint2Line2
- Polygon5
- Polygon4
- Polygon3
- Polygon2
- Polygon1
- Polygon0

Fields:
Root/Polygon5.Points[0]
Root/Polygon5.Points[1]
Root/Polygon5.Points[3]
Root/Polygon5.Lines[3]
Root/Polygon0.Points[1]
Root/Polygon0.Points[2]
Root/Polygon0.Points[3]
Root/Polygon1.Points[0]
Root/Polygon1.Points[1]

Figure 8: Snapshot of the interactive tool for evaluation of constraint equations.

REFERENCES

Anders, K.-H. and Sester, M., 2000. Parameter-free cluster detection in spatial databases and its application to typification. In: IAPRS Vol. 33 Part A4, Amsterdam.

Baltsavias, E. P., 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. ISPRS Journal of Photogrammetry and Remote Sensing 58, pp. 129–151.

Brenner, C., 2004. Modelling 3D objects using weak primitives. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXV, Istanbul, 2004.

Brenner, C., 2005. Building reconstruction from images and laser scanning. Int. Journal of Applied Earth Observation and Geoinformation 6(3-4), pp. 187–198.

Brenner, C. and Sester, M., 2005. Cartographic generalization using primives and constraints. In: Proc. 22nd International Cartographic Conference, July 9-16, La Coruna, Spain.

Christophe, S. and Ruas, A., 2002. Detecting building alignments for generalisation purposes. In: International Symposium on Spatial Data Handling SDH 2002, Ottawa, Canada, pp. 419–432.

Dick, A., Torr, P., Cipolla, R. and Ribarsky, W., 2004. Modelling and interpretation of architecture from several images. International Journal of Computer Vision 60(2), pp. 111–134.

Ermes, P., 2000. Constraints in CAD models for reverse engineering using photogrammetry. In: IAPRS Vol. 33 Part B5, Amsterdam.

Gilks, W. R., Richardson, S. and Spiegelhalter, D. J. (eds), 1996. Markov Chain Monte Carlo in Practice. Chapman & Hall.

Green, P. J., 1995. Reversible jump markov chain monte carlo computation and bayesian model determination. Biometrika 82(4), pp. 711–732.

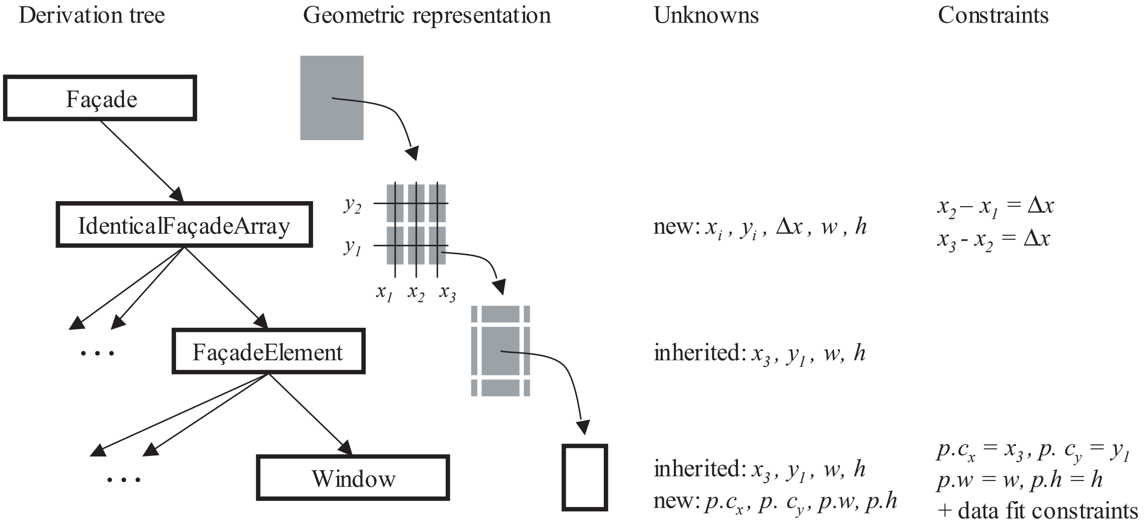


Figure 7: Derivation tree, corresponding geometry, and generated unknowns and constraints.

Kilpeläinen, T. and Sarjakoski, T., 1995. Incremental generalization for multiple representations of geographic objects. In: J. Muller, J. P. Lagrange and R. Weibel (eds), GIS and Generalization: Methodology and Practise, Taylor & Francis, London, pp. 209–218.

Kolbe, T. H., Gröger, G. and Plümer, L., 2005. CityGML – interoperable access to 3D city models. In: P. van Oosterom, S. Zlatanova and E. Fendel (eds), Proc. of the Int. Symposium on Geo-information for Disaster Management, Delft, March 21.-23., Springer.

Lamy, S., Ruas, A., Demazeau, Y., Jackson, M., Mackaness, W. and Weibel, R., 1999. The application of agents in automated map generalization. In: Proceedings of the 19th International Cartographic Conference of the ICA, Ottawa, Canada.

Leica Geosystems, 2006. Cyclone software. http://www.leica-geosystems.com/hds, last accessed March 2006.

Marvie, J.-E., Perret, J. and Bouatouch, K., 2005. The FL-system: a functional L-system for procedural geometric modeling. The Visual Computer 21(5), pp. 329 – 339.

Mitchell, W. J., 1990. The Logic of Architecture : Design, Computation, and Cognition. Cambridge, Mass.: The MIT Press.

Parish, Y. and Müller, P., 2001. Procedural modeling of cities. In: E. Fiume (ed.), ACM SIGGRAPH, ACM Press.

Prusinkiewicz, P. and Lindenmayer, A., 1990. The algorithmic beauty of plants. New York, NY: Springer.

Ripperda, N. and Brenner, C., 2006. Reconstruction of facade structures using a formal grammar and rjMCMC. In: 28th Annual Symposium of the German Association for Pattern Recognition (accepted).

Stiny, G. and Gips, J., 1972. Shape Grammars and the Generative Specification of Painting and Sculpture. Auerbach, Philadelphia, pp. 125–135.

Weidner, U., 1997. Digital surface models for building extraction. In: A. Grün, E. Baltsavias and O. Henricsson (eds), Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhäuser, Basel, pp. 193–202.

Wonka, P., Wimmer, M., Sillion, F. and Ribarsky, W., 2003. Instant architecture. ACM Transaction on Graphics 22(3), pp. 669–677.

ACKNOWLEDGEMENT

This work has been carried out within the scope of the junior research group “Automatic methods for the fusion, reduction and consistent combination of complex, heterogeneous geoinformation”, funded by the VolkswagenStiftung, Germany.

ESTIMATING THE ESSENTIAL MATRIX: GOODSAC VERSUS RANSAC

Eckart Michaelsen[a], Wolfgang von Hansen[a], Michael Kirchhof[a], Jochen Meidow[a], Uwe Stilla[b]

[a]FGAN-FOM, Gutleuthausstr. 1, 76275 Ettlingen, Germany, mich@fom.fgan.de
[b]Photogrammetry and Remote Sensing, Technische Universität München, Arcisstr. 21, 80280 München, Germany

KEY WORDS: essential matrix, robust estimation, RANSAC, structure from motion

ABSTRACT:

GOODSAC is a paradigm for estimation of model parameters given measurements that are contaminated by outliers. Thus, it is an alternative to the well known RANSAC strategy. GOODSAC's search for a proper set of inliers does not only maximize the sheer size of this set, but also takes other assessments for the utility into account. Assessments can be used on many levels of the process to control the search and foster precision and proper utilization of the computational resources. This contribution discusses and compares the two methods. In particular, the estimation of essential matrices is used as example. The comparison is performed on synthetic and real data and is based on standard statistical methods, where GOODSAC achieves higher precision than RANSAC.

1 INTRODUCTION

1.1 Motivation

One of the basic tasks for many computer vision applications is to describe a set of measurements by a mathematical model. Often this model is overdetermined because the number of measurements is much higher than the number of unknown sought model parameters. Different methods to find an optimal solution despite the presence of noise and outliers have been devised during the years. The techniques used for robust estimation include random sampling (RANSAC), a complete search to test all possible inlier sets, clustering such as the Hough transform and maximum-likelihood-type estimation (McGlone et al., 2004, p. 103 ff). Even though they vary greatly in detail, their common property is to reduce the influence of outliers on initial solutions thus allowing their detection and removal.

RANSAC is often used when the outlier rate is high. The approach to its solution is to detect an inlier set of maximal size, while the precision of the resulting parameters is not taken into account. Better results can be achieved, when some of the often redundant inlier samples are traded in for others that have more impact on the parameters.

For real-time applications the computation time constraints are hard. Methods where the overall processing time is data-dependent and not guaranteed to be within a given bound cannot be accepted. In particular, it is desirable to have a method that can be terminated by external requirements. On the other hand, it should utilize the given resources properly, i. e. it should also not be ready long before the requirement for a result arrives and keep the resources idling for the rest of the time. For the experiments in this contribution, methods are chosen that have this anytime capability.

1.2 Our approach

In this paper, we propose the GOODSAC *(good sample consensus)* paradigm as an alternative to RANSAC *(random sample consensus)* (Fischler and Bolles, 1981). RANSAC uses a blind generate-and-test strategy that treats all samples equally regardless of their quality and requires a large number of tests to find an optimal solution. Furthermore, RANSAC is nondeterministic so that two runs on the same dataset will return different results.

In contrast to this, GOODSAC replaces the random sampling with an assessment driven selection of *good* samples. Good samples are those that possess a high degree of confidence and advantageous geometry so that the model parameters computed are well defined. Appropriate utility functions can usually be derived from the mathematical, geometric model in order to produce a sorted list of samples which features the most promising ones in the leading positions. Multiple abstraction stages from the measurements to the samples evade the complete search through all combinatorial possibilities.

A second enhancement is the replacement of tedious inlier tests by a clustering in parameter space. While RANSAC defines the best result as the one which has the largest inlier set, GOODSAC looks for solutions that turn up often. The concept of the inlier set is still present in GOODSAC through the set of predecessors – those samples who lead to the cluster in parameter space.

This contribution focuses on the application of GOODSAC to the estimation of essential matrix constraints between two images and in particular on the special case of nonuniform distributed interest points. Its performance is evaluated by comparing it to RANSAC with respect to robustness and accuracy. The experimental setup is chosen to resemble a situation frequently encountered in forward looking aerial thermal videos.

1.3 Related work

RANSAC (Fischler and Bolles, 1981) had been introduced to the scientific community 25 years ago and is widely used for its robustness in the presence of many outliers (25 Years of RANSAC, 2006). Documented enhancements of RANSAC based estimation mainly focus on the reduction of the required number of random samples in order to decrease processing time. (Matas et al., 2002) replaced the seven point correspondences required for the estimation of a fundamental matrix with three matching affine regions. Many implementations contain additions to the original RANSAC – typically constraints that bail out early on bad random samples.

A major recent improvement on the method itself is the preemptive RANSAC scheme (Nistér, 2005). While the original algorithm generates and tests one hypothesis at a time, preemptive RANSAC first generates a number of hypotheses and then tests them in parallel, discarding bad hypotheses early in order to speed up processing.

Another variant more along the lines of our work is PROSAC *(progressive* RANSAC*)* (Chum and Matas, 2005). PROSAC introduces an assessment component to narrow the set of samples to draw from. If successful, this achieves a higher inlier rate so that fewer runs are needed. In contrast, GOODSAC eliminates the random sampling process completely and introduces additional assessment components to achieve robustness.

GOODSAC has already been used for completely different recognition tasks in the past (Michaelsen et al., 2006).

2 METHOD AND TEST SETUP

2.1 The GOODSAC Paradigm

The good sample consensus principle has been designed for tasks where an assessment on the quality of the samples and of their parts can be provided (Michaelsen and Stilla, 2003). In this case the random search for a sample that maximizes consensus can be replaced by a controlled search. It is intended to capture sensible heuristics or proven utilities in a more systematic way and use them to prevent waste of computational resources and foster precision.

Let $F(\mathbf{m}, \mathbf{x}) = 0$ denote an implicitly defined functional model, where $\mathbf{m}$ is a k-tuple of parameters and $\mathbf{x}$ is an n-tuple of measurements. Only a subset $\{x_i\}$ with $i \in \mathcal{I} \subseteq \{1, \dots, n\}$ fulfills the model. The task is to estimate $\mathbf{m}$ from $\mathbf{x}$. If the set $\mathcal{I}$ were known, the solution would be found by minimizing $\sum_{i\in\mathcal{I}} F(\mathbf{m}, x_i)^2$ under variation of $\mathbf{m}$. Systematical complete search in the power set 2^n for an optimal set $\mathcal{I}$ is usually not feasible. It is assumed that there exists $\ell < n$ minimally such that $\mathbf{m}$ can be determined from an ℓ-sample $\{x_{i_1}, \dots, x_{i_\ell}\}$.

Whereas the RANSAC method draws samples at random, GOODSAC regards them as objects. Each object is assessed according to its presumed utility for the estimation task at hand. Furthermore, it exploits part-of hierarchies: intermediate objects are introduced between the single measurements and the ℓ-samples, which are smaller sub-sample objects (e. g. pairs or triples). Thus, the way is open for a better control on the search. Badly composed sub-sample objects can be neglected, while presumably well suited ones can be preferred – ℓ-sample objects vote for specific settings of the parameters $\mathbf{m}$ of the model. ℓ-sample objects with consistent votes – according to a metric and threshold in $\mathcal{M} \ni \mathbf{m}$ – are parts of a cluster object, which represents the highest level of the object hierarchy. The best cluster object is the result.

2.2 Assessment Driven Control

GOODSAC uses a general control approach. A part-of hierarchy is formulated as finite production system:

$$\{p_\iota; p_\iota = o_\kappa \leftarrow (o_\lambda, o_\mu) \vee o_\kappa \leftarrow \{o_\lambda, \dots, o_\lambda\}\}, \qquad (1)$$

where p_ι denotes the productions and o_κ, o_λ or o_μ respectively denote object types. Note that the productions may be of two forms – either a more complex object is formed from an ordered pair of simpler objects of possibly different kinds or it is formed from a set of objects of the same kind. Associated with each production p_ι there is a predicate π_ι that the right side must fulfill for the production to be appropriate, a function that determines the object instances attribute values on the left side from the values found in the right side and in particular an assessment function α_ι for the newly constructed object instance. A proper assessment driven control cycle for production systems has already been given by (Stilla et al., 1995):

1. Form working elements $(\alpha_0, o_\lambda, \xi, \text{nil})$ from a given set of primitive object instances, where ξ is a pointer to the object instance and α_0 its initial assessment.

2. Sort the set of working elements according to the assessments α.

3. Pick a fixed number of elements from the "good end" of the sorted working set and proceed with all of them.

4. Let $(\alpha_0, o_\lambda, \xi, \chi)$:

   (a) If $\chi = \text{nil}$, then clone $(\alpha_0, o_\lambda, \xi, \chi)$ with $\chi = \iota$ for each production p_ι in which o_λ occurs on the right side.

   (b) Else query the database for partner object instances that fulfill π_ι together with the object o_λ to which ξ points; generate all new objects o_κ that are obtained using these combinations and insert new working elements $(\alpha_\iota, o_\kappa, \eta, \text{nil})$ for each of these with η pointing to them.

5. If the set of working elements is still not empty and no external break criterion holds continue at step 2.

After breaking the control cycle the best object of the highest hierarchical type is chosen as result. Using this control scheme for GOODSAC is achieved by taking the measurements $\mathbf{x}$ as primitive objects and larger samples as intermediate non-primitive objects up to the minimal ℓ-samples required for calculating F. Thus these objects can be attributed with parameter estimations for F. A single cluster production for these minimal ℓ-sample objects is added (of the type $o_\kappa \leftarrow \{o_\lambda, \dots, o_\lambda\}$) that demands adjacency in the parameter space of F. It constructs non-minimal sample objects that are the result of the process.

The assessment functions used in this process must not only be capable of comparing the presumed utility of objects of the same type and hierarchy level, they must also be valid between objects of all different types, because all these objects constantly compete for the same computational resources. There are no random choices in a GOODSAC search run. It is completely determined by F, $\mathbf{x}$, the object hierarchy and the assessment functions. Its success depends on the care of the designer of the latter structures. It is particularly appropriate where utility assessment criteria can be given in a mathematically sound way. Sect. 2.3 gives an example for the estimation of essential matrix constraints.

2.3 An Example: Estimating Essential Matrices

GOODSAC is particularly suitable for essential matrix estimation. The minimal sample for this task is five correspondences $(x, y, x', y')^\top$ giving one constraint each, so that the five parameters of an essential matrix $\mathbf{E}$ can be obtained by evaluating the roots of a polynomial of 10th degree (Nistér, 2004). GOODSAC clusters each of the up to ten hypotheses computed from one sample. This independent treatment of the hypotheses is similar to a straightforward RANSAC implementation.

The hierarchy of the corresponding GOODSAC system consists of five object types: Correspondences $\boldsymbol{K}$, pairs $\boldsymbol{P}$, quadruples $\boldsymbol{Q}$, quintuples $\boldsymbol{R}$ and essential matrix clusters $\boldsymbol{C}$. The following attributes and assessment functions are assigned to these objects:

1. Objects $\boldsymbol{K}$ are obtained from image pairs taken from a video stream. Therefore they are attributed with the locations of the corresponding item in the first and second image

$(x, y, x', y')^\top$. It will be most useful if objects $\boldsymbol{K}$ with high assessment have a small expected error or outlier probability. There are extraction methods that provide these measures, and they have proven very useful for RANSAC acceleration (Chum and Matas, 2005). In our comparison uniform distributed random assessments have been used in the experiments in sect. 3 because emphasis is on exploitation of the geometric configurations. Random assessments are the worst possible choice apart from using the sequence in which the objects have been generated.

2. Objects $\boldsymbol{P} \leftarrow \boldsymbol{KK}$ are pairs of correspondence pairs. A similarity transform with four degrees of freedom may be calculated from an object $\boldsymbol{P}$. The expected deviation of this transform from the correct value would inversely depend on the Euclidean distance d between the objects $\boldsymbol{K}$ preceding it (in one or the other image). This motivates heuristically that the assessment is obtained from this distance. It is directly and linearly transformed to the assessment interval $[0, 1]$ by $a(\boldsymbol{P}) = d/d_{\max}$ where $d_{\max}$ is the maximal possible distance in the image.

3. Objects $\boldsymbol{Q} \leftarrow \boldsymbol{PP}$ are quadruple of correspondence objects. For their assessment the area a of the smallest of the four triangles formed from the four locations is used. This motivation is based on the fact that from an object $\boldsymbol{Q}$, a planar projective transform with eight degrees of freedom may be calculated. The expected deviation of this transform from the correct value would be highly correlated to this assessment value $a/a_{\max}$, where $a_{\max}$ is the area of the largest possible triangle in the image bounds. It can at most be equal to one, but usually it is much smaller. In order to balance between different object types, $(a/a_{\max})^e$ is used for the assessments of objects $\boldsymbol{Q}$ with an appropriate value $0 < e < 1$.

4. Objects $\boldsymbol{R} \leftarrow \boldsymbol{QK}$ are a quintuple of correspondence objects. Given an object $\boldsymbol{Q}$, partners $\boldsymbol{K}$ that are not co-linear with two of the four locations of this object are searched. The assessment is again obtained from the area of the smallest triangle. Also this assessment must be properly normed to be bounded by one and balanced with the other assessments as described above.

5. Objects $\boldsymbol{C} \leftarrow \{\boldsymbol{R}; \text{mutually consistent}\}$ are clusters of essential matrices very similar to each other. All objects $\boldsymbol{K}$ preceding them are again entered into the same procedure – this time in its overdetermined version – resulting in a new more precise estimation of $\mathbf{E}$. For assessment, the convex hull of all preceding objects $\boldsymbol{K}$ in the image is determined. The assessment value is formed as product of the number of correspondences k and the area of the convex hull. This assessment is neither balanced with respect to the other assessments nor bounded by one. It is not used for control purposes. Objects $\boldsymbol{C}$ do not compete for computational resources. This assessment is only used for picking the best result after termination of the GOODSAC search run.

If a very precise result is needed, a concluding consensus set may be formed from all objects $\boldsymbol{K}$ being consistent with the result, followed by least squares optimization. This last step is also proposed for RANSAC search. It is well known as "guided matching" (Hartley and Zisserman, 2000, p. 125).

2.4 Performance Evaluation

In this section we evaluate the performance of the proposed estimations with respect to the robustness of the procedures and the accuracies of the individual results. Whereas the ability of detecting outliers is specified by an error rate in terms of a binary classification, the results of the parameter estimations will be evaluated using statistical tests and results from adjustment theory, cf. (Mikhail, 1976, Förstner, 1994) for instance. For the real data set we do not have the true projection matrices and we are also not sure of the presence of possible non-projective distortions. Therefore we will describe qualitatively the result on a particular example.

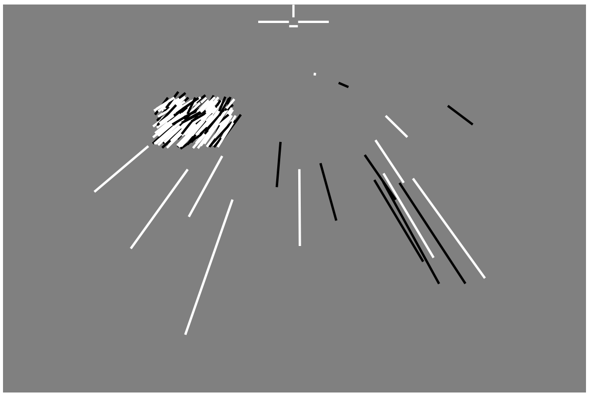


Figure 1: White displacements are inlier correspondences, black displacements outliers, the white aircraft symbol indicates the epipole.

2.4.1 Robustness

Outlier detection: Error rate. We consider the procedures to be a binary classifier indicating inliers and outliers with the help of a threshold. Competing classifiers can be evaluated based on their empirical confusion matrices. The rate of missed outlier detections is of interest beside the error rate being the ultimate measure of the classification performance (Jain et al., 2000). Since these measures are random variables, they have an associated distribution permitting hypothesis testing.

Self-diagnosis. In automatic analysis there is a demand for reliable self-diagnostics. Concerning the detectability of errors evaluation quantities can be derived from the stochastic model within general least squares adjustment models:

An initial covariance matrix $\mathbf{Q}_{xx}$ of the observations $\mathbf{x}$ is assumed to be known and related to the true covariance matrix $\mathbf{C}_{xx}$ by $\mathbf{C}_{xx} = \sigma_0^2 \mathbf{Q}_{xx}$ with the possibly unknown scale factor σ_0^2, also called variance factor. If the initial covariance matrix correctly reflects the uncertainties of the observations, this factor is $\sigma_0^2 = 1$. The estimated parameters are independent with respect to scaling of this covariance matrix, therefore only the ratios of the variances and covariances have to be known in advance.

The variance factor can be estimated from the estimated corrections $\hat{\mathbf{v}}$ for the observations $\mathbf{x}$ via

$$\hat{\sigma}_0^2 = \frac{\hat{\mathbf{v}}^\top \mathbf{Q}_{xx}^{-1} \hat{\mathbf{v}}}{R} \qquad (2)$$

with the redundancy R of the system.

If the mathematical model actually holds and the observations are normally distributed, the estimated variance factor will be Fisher distributed with R and ∞ degrees of freedom (McGlone et al., 2004)

$$T_1 = \frac{\hat{\sigma}_0^2}{\sigma_0^2} \sim F_{R,\infty} \qquad (3)$$

with the test statistic T_1 having the expectation value one.

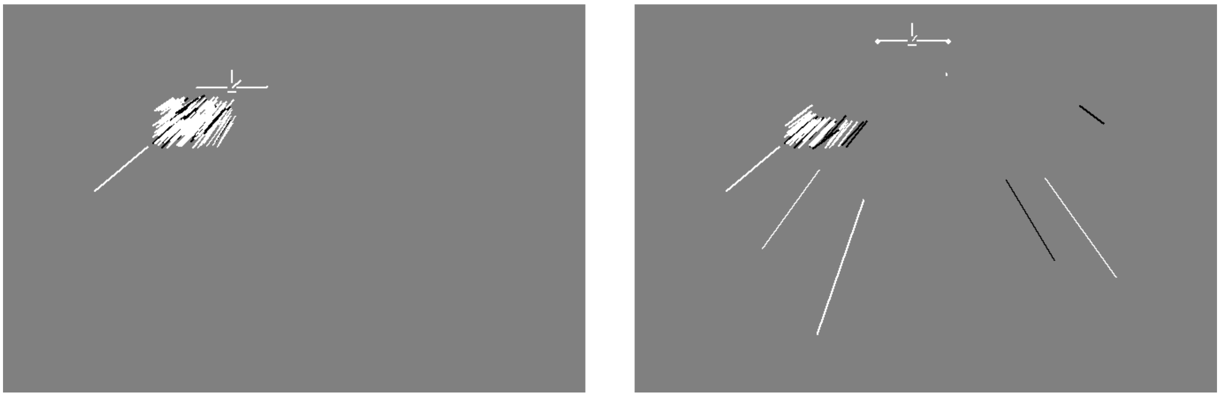


Figure 2: Left: Typical RANSAC result, right: Typical GOODSAC result. RANSAC maximizes the size of the inlier set, while GOODSAC returns correspondences that allow better precision of the estimated parameters.

If the test (3) is accepted, the data and model will fit. In the case of deviations there is no evidence for the reasons. In particular, small errors in the assumptions concerning the precision of the observations lead to a rejection of the test. But, for *synthetic data* σ_0^2 is known and the mathematical model holds. Therefore this test checks indirectly the robustness, since gross errors or blunders lead to a rejection.

2.4.2 Precision and Accuracy

Acceptability of the empirical precision. The empirical precision indicates the effect of random errors onto the estimated parameters, taking the estimated variance factor $\hat{\sigma}_0^2$ into account. The empirically estimated covariance matrix for the estimated parameters $\mathbf{m}$ is

$$\hat{\mathbf{C}}_{\hat{m}\hat{m}} = \hat{\sigma}_0^2 \mathbf{Q}_{\hat{m}\hat{m}} \quad (4)$$

where the covariance matrix of the estimated parameters results from $\mathbf{Q}_{\hat{m}\hat{m}} = (\mathbf{J}^\top \mathbf{Q}_{xx}^{-1} \mathbf{J})^{-1}$ if the observations can be expressed explicit in terms of the parameters, where $\mathbf{J}$ denotes the Jacobian of the equations with respect to the parameters.

If a certain precision of the parameters is required, the individual values can be compared with some pre-specified tolerances for the specific application.

Empirical Accuracy. The evaluation of the covariance matrices, as discussed so far, is only an internal evaluation relying on the internal redundancy of the observation process. Systematic errors, which may not have an influence on the residuals but may deteriorate the estimated parameters, are not taken into account. Evaluating the empirical accuracy of the estimated parameters therefore requires reference values $\mathbf{m}_r$ for the parameters.

The Mahalanobis distance is useful for checking the complete set

$$T_2 = (\hat{\mathbf{m}} - \mathbf{m}_r)^\top \left(\mathbf{C}_{rr} + \hat{\mathbf{C}}_{\hat{m}\hat{m}}\right)^{-1} (\hat{\mathbf{m}} - \mathbf{m}_r) \sim \chi_u^2 \quad (5)$$

within a combined statistical test with u degrees of freedom being the number of parameters. If the test (5) has been rejected it can be concluded that the accuracy potential of the observations is not exploited, provided that the reference data $\mathbf{m}_r$ actually are correct and thus $\mathbf{m} \sim N(\mathbf{m}_r, \mathbf{C}_{rr})$.

3 EXPERIMENTS

3.1 Experiments with Synthetic Data

Optimal statistical analysis can only be accomplished with synthetic data because exact measure of the noise is required. A time constraint has been introduced by limiting the number of sample objects $\boldsymbol{R}$ to 2,000. The same number of samples was then permitted to the RANSAC search runs. This is almost two orders of magnitude larger than the standard textbook literature recommends for quintuple samples at 95% probability for an inlier-only sample with our 33% outlier rate (Hartley and Zisserman, 2000). For correspondences uniform distributed all over the image both methods will yield robust results. In order to elaborate the difference in the behavior a critical situation was simulated in the following way: 90% of the correspondences are located within a small region of the image. Only 10% are uniform distributed over the entire image (Fig. 1). For the synthetic data the scene is assumed to be flat. Therefore the correspondences result from a planar projective homography constraint. While a planar scene cannot be used for fundamental matrix estimation it should not pose a problem to essential matrix estimation following (Nistér, 2004).

Fig. 1 shows the generated frame-to-frame point correspondences. The camera motion has no rotational component. Thus the epipole – sketched as aircraft symbol – is a fixed point giving the flight direction and the horizon is a straight line of fixed points. 67% of the data are disturbed by an additive normally distributed shift error on the position in the second image. 33% of the data are disturbed by a much larger additive shift error on the position in the second image. This error is distributed uniformly within a squared search window eight times larger than the standard deviation of the inliers. GOODSAC requires assessment values for the correspondences. In this example, they were chosen randomly and independent of the outlier property and displacement error. Because the result of the search depends on these initial assessments the outcome is nondeterministic (as it normally would be with real assessments), allowing a statistical evaluation. Therefore the GOODSAC run was repeated 20 times with independently drawn assessments.

One GOODSAC result on the particular data set given in Fig. 1 is displayed in Fig. 2, left. The GOODSAC estimation is based on a fairly small number of objects $\boldsymbol{K}$ with some outliers included. These correspondences, however, are well spread over the image, so that the resulting estimation fits the ground truth neatly. The epipole is again displayed as aircraft symbol. Almost no rotation is left. Yaw and pitch rotations are indicated by a line inside the center of the aircraft symbol showing the resulting displacement and the roll is indicated by fins on the wingtips. RANSAC is a non-deterministic method. Therefore we repeated the experiment 20 times for each particular setting of correspondences. Among these runs there were also examples, where the outcome was superior to the GOODSAC result. To make our point clearly we decided to show an example result in Fig. 2 which comes up with

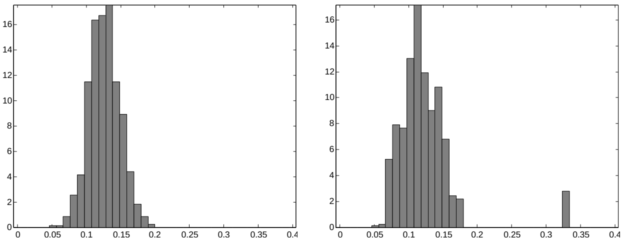


Figure 3: Empirical distributions for the false positives rates. Left: RANSAC, right: GOODSAC. For the peak at 33% see text.

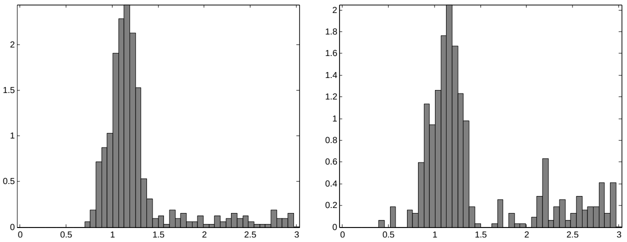


Figure 4: Empirical distribution of the variance ratios (3). Left: RANSAC, right: GOODSAC.

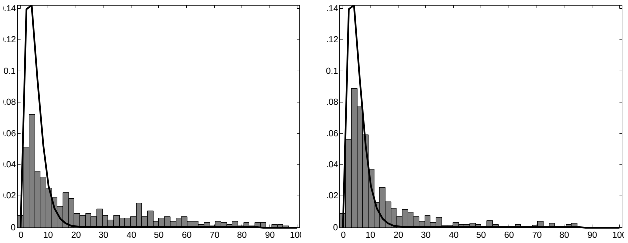


Figure 5: Empirical distribution of the Mahalanobis distances with the χ^2_5 distribution. Left: RANSAC, right: GOODSAC.

a fairly low false positive rate but with an unpleasant deviation of the essential matrix. There are only few black lines visible. But the estimation is based on a small image region. Thus the epipole can be displaced considerably from the true position. To compensate for this a considerable rotation – even in roll angle – is "invented".

The quantitative evaluation is based on 40 different settings of correspondences with 20 GOODSAC searches and 20 RANSAC searches performed on each. RANSAC always finds the vast majority of the consensus set inside densely populated areas. GOODSAC typically spreads the hypothesis generating samples across the entire image. Fig. 3 shows the distributions of the false positives rates for the GOODSAC and the RANSAC approach with a similar shape. The expectation value of 0.12 is caused by the fact that even in the optimal case about 30% of the outliers fit to the essential matrix and therefore are not detectable at this stage. There are rare situations, where almost all correspondence objects $\boldsymbol{K}$ closest to the image margin are actually outliers. This may cause the GOODSAC search to fail completely. Even after 2,000 quintuple objects $\boldsymbol{R}$ have been constructed no cluster may be found at all. Then the procedure falls back on using all correspondences as inliers. These cases lead to a small peak at 33% false positive rate for the GOODSAC method.

The empirical distribution of test statistics (Eq. (3)) is plotted in Fig. 4. Note that these values stem from different Fisher distributions since the degrees of freedom are varying with the number of inliers. The values obviously do not exceed the expectation value one for both estimation methods. Thus, the outliers have been removed successfully.

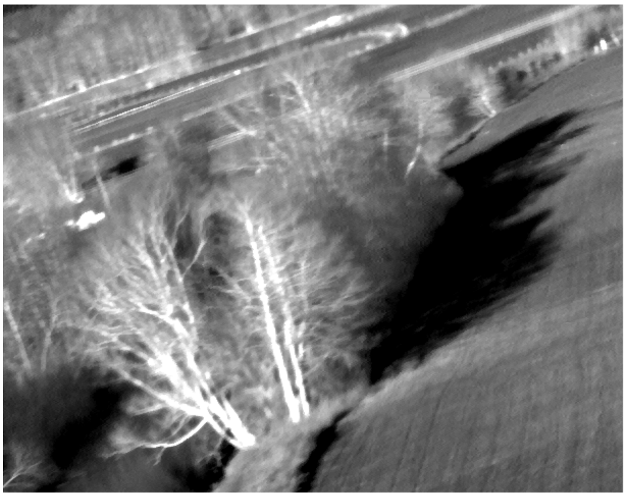


Figure 6: Frame from a forward-looking thermal video captured from a helicopter.

The empirical distributions of the Mahalanobis distances (5) shown in Fig. 5 reveal some deviation from the expected (analytical) probability density function. It can be seen that GOODSAC is closer to the theoretical distribution than RANSAC. This is because for a given sample and a corresponding essential matrix, the essential matrix is extrapolated outside the convex hull of this sample. While RANSAC maximizes the size of the sample as expected, this extrapolation leads to lower accuracy in the essential matrix.

3.2 An Experiment with Real Data

GOODSAC has been designed for applications where non-uniform distributed features are a common phenomenon. In particular, aerial forward-looking thermal videos often exhibit large uniform areas and strongly textured or structured regions often are quite sparse and small. An example frame is presented in Fig. 6.

Correspondences were obtained from a pair of frames with a sufficient baseline length, so that the displacements allow essential matrix estimation. Then GOODSAC and RANSAC were applied to these data under the same conditions that were also used for the synthetic setup. Quantitative evaluation of this experiment would need manual labeling of outliers, acquisition of ground truth, e. g. by an inertial navigation system, and repetition with a considerable number of image pairs. This has not yet been undertaken. Instead, in this contribution only the tendency of the outcome can be shown by example results in Fig. 7.

Note that while the sample found as consensus set by RANSAC has a larger size than the consensus set found by GOODSAC. However, some correspondences on the margin of the correspondence point cloud are missing in the RANSAC set, but appear in the set found by GOODSAC. This confirms the tendency found by the investigations with the synthetic data.

4 DISCUSSION AND OUTLOOK

Concerning the false positive rates on the synthetic dataset GOODSAC is only a little better than RANSAC. However, the Mahalanobis distance plots of the essential matrices resulting from the same experiments indicate that higher accuracy can be expected from GOODSAC. This can be explained by the fact that RANSAC simply tries to maximize the number of inliers which will only be directly related to the accuracy, if they are uniformly

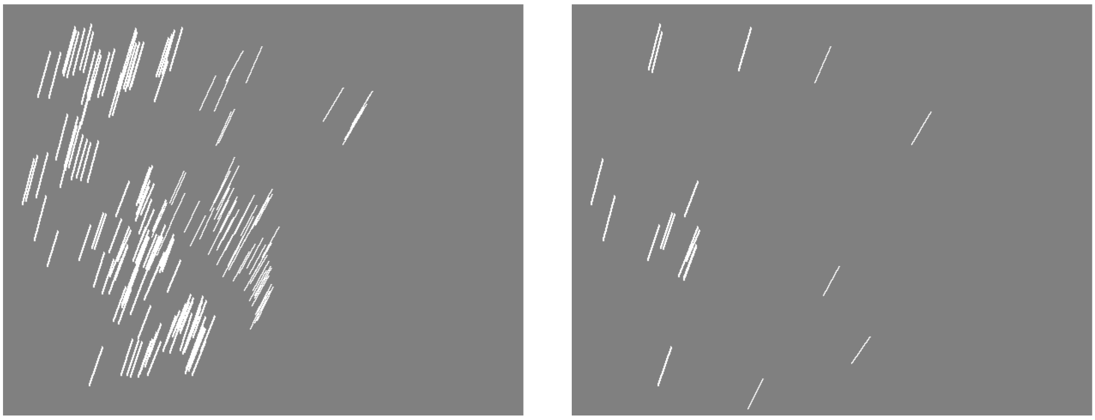


Figure 7: Typical result of the generated samples. Left: RANSAC, right: GOODSAC.

distributed over the entire image. GOODSAC tries to back the estimation by a stable geometric base and trades the sheer number of measurements for it.

The part-of hierarchy used for essential matrix estimation overlaps highly with that suitable for planar homography estimation (Michaelsen and Stilla, 2003). We may just add attributes to the quadruple objects $\boldsymbol{Q}$, add a clustering production and balance the assessment functions accordingly. If the scene is planar, the homography results will usually be more reliable, else the essential matrix solution will be better, while both calculations may be based on the same partial sub-calculations.

An open research problem with respect to this multiple use of intermediate objects is the choice of the assessment functions. We did not use any meaningful assessments on the elementary correspondence objects $\boldsymbol{K}$ here – for reason of fair competition. But in a real application we would of course use something reasonable: The quality of the match between the first and second image gives a good criterion related to both the outlier probability and the expected displacement error of an inlier correspondence. Or the length of displacement between the two images, because this estimation will fail on a set of stationary correspondences.

Further research is also needed to compare the two methods on real data with real inliers and outliers. The used assumptions on which the distributions of both kinds of correspondences are based must be verified. Here, we presume the normal distribution of the inliers to be the smaller problem. The distribution of real inliers may be deviating due to the pixel structure of the detector or properties of the matching algorithm, but the deviation may well be tolerable. The distribution of outliers, however, is probably a more severe problem. Real outliers do not occur randomly with equal probability anywhere. They are caused by unpredictable clutter effects. Some of these (e. g. moving objects, partial occlusions) may be foreseeable but a quantitative prediction is hard. They will, however, have a bias, and the influence of outliers on either method remains to be studied. However, the presented statistics based on the simplified assumption on the outliers still indicate potential usefulness of the presented method.

REFERENCES

25 Years of RANSAC, 2006. Workshop in conjunction with CVPR 2006, New York, June 18.

Chum, O. and Matas, J., 2005. Matching with PROSAC – Progressive Sample Consensus. In: Proc. of Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, pp. 220–226.

Fischler, M. A. and Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the Association for Computing Machinery 24(6), pp. 381–395.

Förstner, W., 1994. Diagnostics and Performance Evaluation in Computer Vision. In: Performance versus Methodology in Computer Vision, NSF/ARPA Workshop, IEEE Computer Society, Seattle, pp. 11–25.

Hartley, R. and Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge.

Jain, A. K., Duin, R. P. W. and Mao, J., 2000. Statistical Pattern Recognition: A Review. IEEE Transactions on Pattern Recognition and Machine Intelligence 22(1), pp. 4–37.

Matas, J. et al., 2002. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In: Proceedings of the British Machine Vision Conference, Vol. 1, pp. 384–393.

McGlone, J. C., Mikhail, E. M. and Bethel, J. (eds), 2004. Manual of Photogrammetry. 5th edn, American Society of Photogrammetry and Remote Sensing.

Michaelsen, E. and Stilla, U., 2003. Good Sample Consensus Estimation of 2D-Homographies for Vehicle Movement Detection from Thermal Videos. In: H. Ebner, C. Heipke, H. Mayer and K. Pakzad (eds), Photogrammetric Image Analysis PIA03, International Archives of Photogrammetry and Remote Sensing, Vol. 34, Part 3/W8, pp. 125–130.

Michaelsen, E., Soergel, U. and Thoennessen, U., 2006. Perceptual Grouping in Automatic Detection of Man-Made Structure in High Resolution SAR Data. Pattern Recognition Letters 27(4), pp. 218–225.

Mikhail, E. M., 1976. Observations and Least Squares. With Contributions by F. Ackerman. University Press of America, Lanham.

Nistér, D., 2004. An Efficient Solution to the Five Point Relative Pose Problem. IEEE Transactions on Pattern Recognition and Machine Intelligence 26(6), pp. 756–769.

Nistér, D., 2005. Preemptive RANSAC for Live Structure and Motion Estimation. Machine Vision and Applications 16(5), pp. 321–329.

Stilla, U., Michaelsen, E. and Lütjen, K., 1995. Structural 3D-Analysis of Aerial Images with a Blackboard-based Production System. In: A. Grün and O. Kübler (eds), Automatic Extraction of Man-made Objects from Aerial and Space Images, Birkhäuser, Basel, pp. 53–62.

EXTRACTION OF HORIZONTAL VANISHING LINE USING SHAPES AND STATISTICAL ERROR PROPAGATION

L. Havasi *, T. Szirányi

Hungarian Academy of Science, H-1111 Kende utca 13-17 Budapest, Hungary - (havasi, sziranyi)@sztaki.hu

Commission III

KEY WORDS: geometrical scene analysis, error propagation, video processing, Hough transformation

ABSTRACT:

In this paper we address the problem of estimating the horizontal vanishing line, making use of motion statistics derived from a video sequence. The computation requires the satisfying of a number of corresponding object's height measurement; and in our approach these are extracted using motion statistics. These easy-to-compute statistics enable accurate determination of average shape of every point in the image. Thanks to the use of motion statistics and error propagation formulas in the intermediate steps, our approach gives robust results. The outcomes show that our approach gives accurate results in the context of different environments.

1. INTRODUCTION

In recent years there has been a dramatic increase in the number of video surveillance systems in use; and these have in turn generated a large quantity of archived video recordings, which are usually stored without any image-processing. In most cases for such recordings one does not know the relative and global geometrical properties of the surveillance cameras.

Despite this, there is a striking lack of publications concerning the extraction of geometric characteristics from images contained in video recordings. We may note that this task is much simplified in the case where some known test object is used during system calibration. In this paper however a statistical framework is introduced which allows us, without such calibration to derive the horizontal vanishing line (VL). The vanishing line is useful for camera orientation and extrinsic parameter determination (Lu et al., 2002). For still images (Criminisi et al., 1999), it can be successfully determined only when there are detectable parallel lines; and in image-sequences, only when certain assumptions are satisfied which enable us to detect and track known objects (Lu et al., 2002). In summary, most of the published still-image based methods are unsuitable for processing the images of a typical surveillance scene. Furthermore, in typical surveillance scenes of public places the assumptions on which the video-based methods are posited are not satisfied.

The main practical advantage of our proposed method is that there is no need for any time-consuming processing steps. Furthermore because of the statistical method employed it is a robust procedure, and sub-pixel accuracy may be achieved. These properties are especially important in the analysis of outdoor surveillance videos. In videos captured by analog surveillance cameras the contrast and focus are often badly adjusted, and thus precise measurements are not possible in individual frames. This consideration led to our concept of summarizing the information from a sequence of a number of frames (as many as possible) in order to achieve higher accuracy in the averaged retrieved information. The paper introduces a parameter-optimization approach which is appropriate to this statistical feature-extraction method; and thus we establish a framework for estimation of the parameters in a slightly different way to other parameter optimization methods (Nguyen et al., 2005; Ji and Xie, 2003).

The parameter estimation method we introduce here applies simple outlier rejection step prior to the nonlinear optimisation. Another advantage of the method is that it is capable of working on low frame-rate videos, since the relevant parameter for the statistical information extraction is not the refresh rate itself, but rather the total frame-count of the processed sequence.

2. PROBLEM DESCRIPTION

Parallel planes in a 3-dimensional space intersect a plane at infinity in a common line, and the image of this line is the horizontal vanishing line, or horizon. The vanishing line (VL) depends only on the orientation of the camera (Hartley and Zisserman, 2000). In the paper we describe the VL with the parameters of the line.

In summary, the determination of the vanishing line is possible with knowledge of at least two vanishing points (these lie in the VL); thus three corresponding line segments (e.g. derived from the height of a given person in the image), or else known parallel lines in the same plane, are necessary.

These line segments can be computed from the apparent height of the same object as seen at different positions (depths) on the ground-plane. The objects may for instance be pedestrians (Lu et al., 2002), and the line segments denote their height. However, the precise detection of such non-rigid objects is a challenging task in outdoor images.

3. PROPOSED METHOD

In this section we introduce a novel solution based on our previous results on motion statistics to derive shape information from videos. The statistical properties of uncertainty of extracted information have been successfully used for accurate parameter estimation.

* Corresponding author.

3.1 Extraction of Shape Properties

Our feature-detection method is based on the use of so-called co-motion statistics (Szlávik et al., 2004). These statistics have been successfully used for image registration in the case of wide-baseline camera pairs and for vanishing point determination in camera-mirror settings (Havasi and Szirányi, 2006). Briefly, these statistics are a numerical estimation of the concurrent motion probability of different pixels in the camera plane

A straightforward step is that the input is some motion mask which is extracted using a suitable algorithm, which may be any one of the several methods available (Cucchiara et al., 2003).

The temporal collection of 2D masks provides useful information about the parts of the image where spatially-concurrent motion occurs, and thereby about the scene geometry. These statistics come from the temporal summation of the binarized masks; these mask are written $m(t,\mathbf{x})$ where *t* is the time and the 2D vector $\mathbf{x}$ is the position in the image. Thus, these masks comprise a set of elements signifying motion ("1") or no-motion ("0"). The co-motion statistics in *local* sense can be summarized with the following two equations. First, we define the global motion statistics which determines the motion probability in every pixel (because of the discrete time-steps, Δt denotes the frame count):

$$P_g(\mathbf{x}) = \frac{\sum_t m(t,\mathbf{x})}{\Delta t} \tag{1}$$

In general, the concurrent-motion probability of an arbitrary image-point $\mathbf{u}$ with another image-point $\mathbf{x}$ may be defined with the following conditional-probability formula:

$$P_{co}(\mathbf{u}|\mathbf{x}) = \frac{\sum_t m(t,\mathbf{u})\, m(t,\mathbf{x})}{\sum_t m(t,\mathbf{x})} \tag{2}$$

For a detailed description of the implementation issues, we refer to (Szlávik et al., 2004). After normalization

$$\sum_{\mathbf{u}} P_{co}(\mathbf{u}|\mathbf{x}) = 1 \tag{3}$$

the $P_{co}(.)$ is assigned to every pixel in the image, the 2D discrete PDF (probability distribution function) will provide useful information about the shapes: the average shape can be determined, because $P_{co}(\mathbf{u}|\mathbf{x})$ collects information about objects which pass through the point $\mathbf{x}$. These PDFs may be approximated by normal distributions because the *central limit theorem* says that the cumulative distribution function of independent random variables (each have an arbitrary probability distribution with mean and finite variance) approaches a normal distribution (Kallenberg, 1997).

Thus,

$$P_{co}(\mathbf{u}|\mathbf{x}) = \mathrm{N}(\mathbf{u},\boldsymbol{\mu}_{\mathbf{x}},\boldsymbol{\Sigma}_{\mathbf{x}}) \tag{4}$$

where the normal distribution is defined as

$$\mathrm{N}(\mathbf{x},\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^2|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{5}$$

The following figure illustrates the results of motion statistics in both indoor and outdoor sequences.

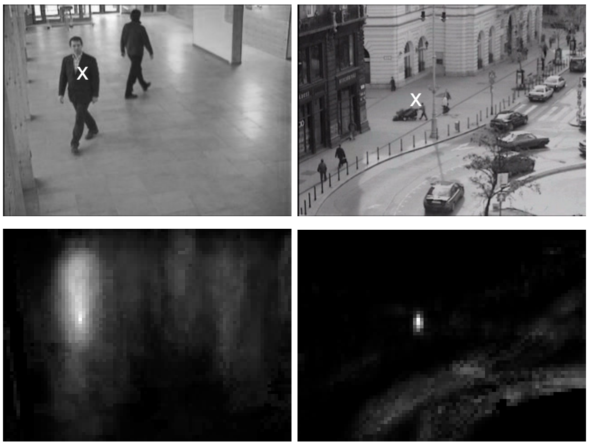


Figure 1. Sample frames in upper row, and raw motion statistics (defined by (2)) in the bottom row. The corresponding point is marked by 'x'

From this feature extraction the input for the further processing steps is the parameters of the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{x}}$ in point $\mathbf{x}$. The dimensions and orientation of the average shape come from the eigen-value decomposition of the covariance matrix:

$$\boldsymbol{\Sigma}_{\mathbf{x}} \mathbf{v}_{\mathbf{x},i} = \lambda_{\mathbf{x},i} \mathbf{v}_{\mathbf{x},i} \quad i = 1,2 \tag{6}$$

These statistical characteristics are displayed in figure 2.

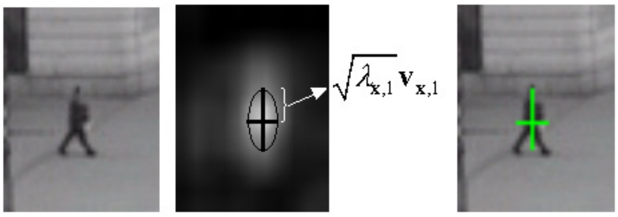


Figure 2: Example to shape properties: axes of normal distributions, derived from the eigen-value decomposition of the covariance matrix.

Finally, the height measurement comes from the projection (vertical component) of the most vertical eigenvector:

$$(\lambda_{\mathbf{x},\max}, \mathbf{v}_{\mathbf{x},\max}) = \arg\max_{(\lambda,\mathbf{v})} \left(\lambda_{\mathbf{x},1}\langle \mathbf{e},\mathbf{v}_{\mathbf{x},1}\rangle, \lambda_{\mathbf{x},2}\langle \mathbf{e},\mathbf{v}_{\mathbf{x},2}\rangle\right)$$

$$h_j = h_{\mathbf{x}} = \sqrt{\lambda_{\mathbf{x},\max}}\langle \mathbf{e},\mathbf{v}_{\mathbf{x},\max}\rangle \tag{7}$$

where $\mathbf{e}$ denotes the vertical unit vector: $\mathbf{e} = [0 \quad 1]$ and $\langle . \rangle$ is the dot product, respectively. These height estimations are displayed in figure 3. For the sake of later simplification we transform the indices from vector (coordinate) form e.g. $\mathbf{x}$ to a simple scalar index *j*. Henceforward, *j* denotes a point in the image; viz. h_j is the height measurement in image-point *j* and vector $\mathbf{p}_j$ determines the coordinates of point *j* in the image. Because the scheme (7) utilizes information extracted from statistics, a more sophisticated form may be given for the height estimation which takes into account the uncertainty:

$$P\left(\hat{h}_j \middle| h_j\right) = \mathrm{N}\left(\hat{h}_j, h_j, \Sigma_{\Delta h_j}\right) \qquad (8)$$

where

$$\Sigma_{\Delta h_j} = \sigma^2_{\Delta h_j} = \left(\sqrt{\lambda_{j,\max}} - h_j\right)^2 \qquad (9)$$

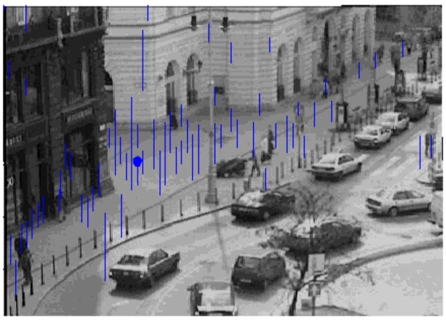


Figure 3: Odd sample from height estimations in outdoor environment.

3.2 Outlier Rejection

In summary, the determination of the vanishing line is possible with knowledge of at least three corresponding line segments. In our framework the necessary height information can be easily determined from the *local* statistics. The information derived from statistics is valid only if the following assumption is satisfied: there are regions where the same objects are moving with equivalent probability (e.g. pathway or road).

In general, without making any prior assumptions about the scene every point may be paired to every other point. But the practical processing of this huge data-set requires that we have an effective way to drop “outlier” points and extract information for VL estimation.

First, we describe simple conditions which can be used to reduce the size of the data-set. The outlier rejection in this case is similar to dropping points where two objects are moving but are not the same size. Let *j* represents an arbitrary point in the image and *k* denotes another (corresponding) point: $j \neq k$

We reckon two points as corresponding points (which is probable, where same-sized objects are concerned) if

$$\sigma_1 < \frac{\lambda_{j,1}}{\lambda_{j,2}} / \frac{\lambda_{k,1}}{\lambda_{k,2}} < \sigma_2 \qquad (10)$$

and

$$\Phi\left(\mathbf{v}_{j,1}, \mathbf{v}_{k,1}\right) < \alpha \qquad (11)$$

where the notations come from the eigenvalue decomposition of the covariance matrices of two points, see (6), and $\Phi\left(.\right)$ denotes the angle between two vectors (the deviation of eigenvectors in our case). These simple conditions lead to a set of points where the objects have similar orientation and aspect ratio. Figure 4 demonstrates the point set (marked by circles) corresponding to a point (marked by x).

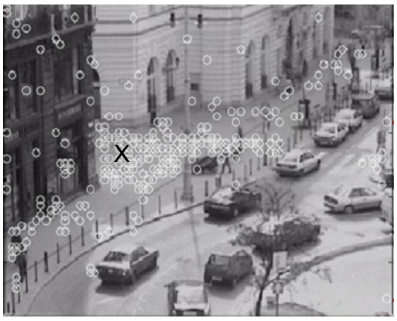


Figure 4: Corresponding points (marked with circles) are related to an arbitrary image point (marked by large ‘x’).

After this preprocessing every point will have several probable corresponding point-pairs. However, several outliers remain, thus we have to use all points to determine vanishing points and an estimation about horizon.

3.3 Error Propagation

An initial guess about the horizon can be computed using the height information of corresponding points to an arbitrary point *j*:

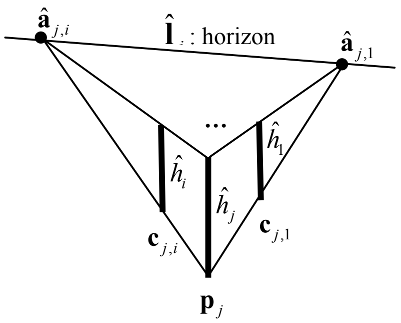


Figure 5. Using vertical size information to get the horizon ($\hat{\mathbf{l}}_j$) and vanishing points $\hat{\mathbf{a}}_{j,i}$. The 2D point $\mathbf{p}_j$ is an arbitrary image point, while $\mathbf{c}_{j,1}$ and $\mathbf{c}_{j,i}$ are two samples for corresponding points determined in the previous section.

To simplify the further computations the transformation between height information and the 2D image plane is necessary. We have to compute point coordinates in the ground-plane, as it is demonstrated in the following figure.

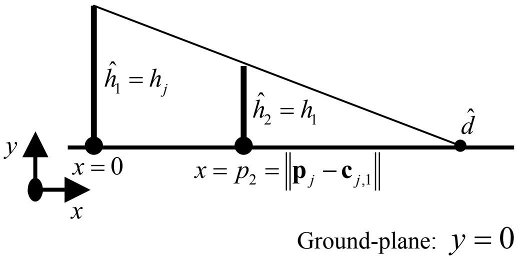


Figure 6. Determination of a vanishing point, which in ideal case lies in the horizontal vanishing line (horizon). The task may be summarized as the computation of $\hat{d}$ taking into account the inaccuracy of height measurements.

The determination of $\hat{d}$ without uncertainty comes from elementary algebra:

$$\hat{d} = \hat{h}_1 \frac{p_2}{\hat{h}_1 - \hat{h}_2} \tag{12}$$

To derive a formula which contains the uncertainty – based on the method described in (Ji and Xie, 2003) – we define the relationship between the input and the output quantity in an implicit form. For this scheme we define the ideal input vector $\mathbf{X}$ and the observed vector $\hat{\mathbf{X}}$. The ideal parameter vector $\mathbf{\Theta}$ and the observed $\hat{\mathbf{\Theta}}$, respectively. The $\hat{\mathbf{\Theta}}$ and $\hat{\mathbf{X}}$ are related through an optimisation function $F(.)$, and $\hat{\mathbf{\Theta}}$ is determined by minimising $F\left(\hat{\mathbf{X}},\hat{\mathbf{\Theta}}\right)$. In this phase of our method the input measurements are height information about the objects, while the output is the estimated position of the intersection of ground plane and the line through points $\left(0,\hat{h}_1\right)$ and $\left(p_2,\hat{h}_2\right)$, see figure 6. This line-plane intersection determines one point, accordingly the input vector is

$$\hat{\mathbf{X}} = \left[\hat{h}_1, \hat{h}_2\right] \tag{13}$$

and the observation is

$$\hat{\mathbf{\Theta}} = \left[\hat{d}\right] \tag{14}$$

The analytic curve function expressed as

$$F\left(\hat{\mathbf{X}},\hat{\mathbf{\Theta}}\right) = \hat{h}_1\left(p_2 - \hat{d}\right) - \hat{h}_2\left(p_1 - \hat{d}\right) = 0 \tag{15}$$

Error propagation relates the uncertainty of input measurements to the perturbation of $\hat{\mathbf{\Theta}}$. Let $\mathbf{\Sigma}_{\Delta X}$ be the covariance matrix of measurements:

$$\mathbf{\Sigma}_{\Delta X} = \begin{bmatrix} \sigma^2_{\Delta X} & 0 \\ 0 & \sigma^2_{\Delta X} \end{bmatrix} \tag{16}$$

where

$$\sigma^2_{\Delta X} = \frac{\sigma^2_{\Delta h_j} + \sigma^2_{\Delta h_k}}{2} \tag{17}$$

Based on the covariance propagation theory (Haralick, 1994), we have

$$\mathbf{\Sigma}_{\Delta\Theta} = 2\sigma^2_{\Delta X}\left[\left(\frac{\partial g}{\partial \mathbf{\Theta}}\right)^T\right]^{-1} \tag{18}$$

where $\dfrac{\partial g\left(\mathbf{X},\mathbf{\Theta}\right)}{\partial \mathbf{\Theta}}$ is defined as

$$\frac{\partial g}{\partial \mathbf{\Theta}} = 2\frac{\left(\dfrac{\partial F}{\partial \mathbf{\Theta}}\right)^2}{\left(\dfrac{\partial F}{\partial h_1}\right)^2 + \left(\dfrac{\partial F}{\partial h_2}\right)^2} \tag{19}$$

Thus, we have

$$\mathbf{\Sigma}_{\Delta\Theta} = \sigma^2_{\Delta\Theta} = \sigma^2_{\Delta X}\frac{\left(p_2 - \hat{d}\right)^2 + \hat{d}^2}{\left(\hat{h}_2 - \hat{h}_1\right)^4} \tag{20}$$

The result is illustrated in the following figure.

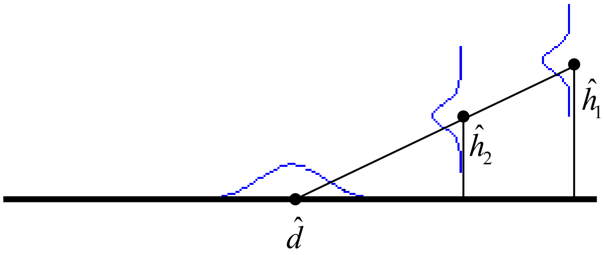


Figure 7. Simulation of error propagation from input data (height estimations) into 1D position coordinate. The two uncertainty heights are used to determine the intersection of line through these points and the x axis. The formula for uncertainty of this intersection was expressed by (20).

Finally, we have to convert the result of (20) into the 2D image plane. This conversion can be accomplished by constructing a 2D covariance matrix:

$$\mathbf{\Sigma}_{\Delta VP_{j,i}} = \mathbf{U}^T \begin{bmatrix} \sigma^2_{\Delta\Theta} & 0 \\ 0 & 0 \end{bmatrix} \mathbf{U} \tag{21}$$

where $\mathbf{U}$ is the matrix of eigen-vectors (Note that, $\mathbf{U}\mathbf{U}^T = \mathbf{I}$.):

$$\mathbf{U} = \begin{bmatrix} \mathbf{v}_i \\ \tilde{\mathbf{v}}_i \end{bmatrix} \text{ and } \left\langle \mathbf{v}_i, \tilde{\mathbf{v}}_i \right\rangle = 0 \tag{22}$$

with eigen-vectors formed from the unit length vector through points $\mathbf{p}_j$ and $\mathbf{c}_{j,i}$:

$$\mathbf{v}_i = \frac{\mathbf{c}_{j,i} - \mathbf{p}_j}{\left\|\mathbf{c}_{j,i} - \mathbf{p}_j\right\|} \tag{23}$$

While the centroid (position of vanishing point defined by points j and i) is determined from the estimated distance $\hat{d}$ along the line with direction $\mathbf{v}_i$:

$$\hat{\mathbf{a}}_{j,i} = \mathbf{p}_j + \mathbf{v}_i\hat{d} \tag{24}$$

Thus, we have the formula for probability density of measurement noise:

$$P\left(\hat{\mathbf{a}}_{j,i} \middle| \mathbf{a}_{j,i}\right) = \mathrm{N}\left(\hat{\mathbf{a}}_{j,i}, \mathbf{a}_{j,i}, \mathbf{\Sigma}_{\Delta VP_{j,i}}\right) \tag{25}$$

3.4 Conversion into Hough-space

After the evaluation of the error propagation formula to every corresponding point pair we will have several uncertain 2D point coordinates. These estimations represent an initial guess about horizon, since the inliers of the data-set lie in the horizon. This line estimation problem is well known and there are several approaches to solve it in various cases: e.g. least squares (LS), total least squares (TLS) and Hough transformation (Kiryati and Bruckstein, 2000; Nguyen et al., 2005).

In short, our case has the following special properties:

1. Error in both coordinates in the 2D plane (x and y).
2. There is correlation between the noise in the two coordinates.
3. The noise covariance matrices are different for different data points (heteroscedastic noise).
4. Notable amount of outliers can be found in the data-set.

Because of these specific characteristics the line fitting is viewed as a global optimisation procedure. Generally, there is no analytic solution for the cases of heteroscedastic and correlated noise, where we assume that the noise in x is correlated to the noise in y, furthermore, the variance of the noise is not identical for all data points. Heteroscedastic regression problem in computer vision is studied in (Kiryati and Bruckstein, 2000). Both LS and TLS methods fail when the data-set contains outliers. Line fitting on such data-set needs a robust estimator, for survey see (Nguyen et al., 2005). The Hough transform is an effective and popular way for line-fitting (Duda and Hart, 1972). In the standard version, an accumulator array is used to collect the points which lie along the same line. The line is parametrized by (θ, ρ):

$$\rho = x\cos(\theta) + y\sin(\theta) \qquad (26)$$

In this section the error propagation will be continued, and an optimal line parameter has been determined by using non-linear optimisation procedure. Because the Hough transformation generates sinusoidal voting patterns in the parameter space we will not use the same error propagation formula as in the previous section. In the end of the section a simple formula for the error estimation in the parameter space will be given.

Let 2D point $\mathbf{a}_{j,i} = (x_i, y_i)$ be the unknown accurate position of the i$^{\text{th}}$ vanishing point introduced in the previous section. The measurements are $\hat{\mathbf{a}}_{j,i} = (\hat{x}_i, \hat{y}_i)$ based on the error propagation formula. Due to noise, $\mathbf{a}_{j,i} \neq \hat{\mathbf{a}}_{j,i}$. The probability density of measurement noise is modelled as a 2D heteroscedastic Gaussian, with correlated noise in formula (25). We define the line-fitting task as finding the maximum of the objective function:

$$\hat{\mathbf{l}}_j = \arg\max_{\mathbf{l}\in(\theta,\rho)} \sum_i P_g(\mathbf{p}_i) C_{j,i}(\mathbf{l}) \qquad (27)$$

where the maximum value of probability (25) along the line $\mathbf{l}$ is defined by:

$$C_{j,i}(\mathbf{l}) = \max_{\mathbf{u}\in\mathbf{l}} P(\mathbf{u} \mid \mathbf{a}_{j,i}) \qquad (28)$$

(This line is also parametrized by $\hat{\mathbf{l}}_j = (\hat{\theta}_j, \hat{\rho}_j)$.)

The optimum value is determined by *unconstraint non-linear optimisation* of (27), the initial estimate is given by LMS method. The introduced formula handles the outliers, thus there is no need for robust M-estimator, where the error expressions are replaced by some saturation function (Nguyen et al., 2005). We note that, the computation of (28) is simple; it has analytic solution, see (Kiryati and Bruckstein, 2000) for details. Since the residual outliers cause an error in line-fitting, we define the error in line-fitting with a 2D Gaussian:

$$P(\hat{\mathbf{l}}_j \mid \mathbf{l}_j) = \mathrm{N}(\hat{\mathbf{l}}_j, \mathbf{l}_j, \Sigma_{\Delta VL_j}) \qquad (29)$$

where the covariance matrix is defined as

$$\boldsymbol{\Sigma}_{\Delta VL_j} = \begin{bmatrix} \tan^{-1}\left(\dfrac{\Delta l_Y}{\Delta l_X}\right)^2 & 0 \\ 0 & \Delta l_X{}^2 \end{bmatrix} \qquad (30)$$

The notations are detailed in the following figure.

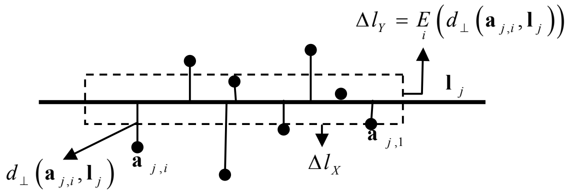

$\Delta l_Y = E_i(d_\perp(\mathbf{a}_{j,i}, \mathbf{l}_j))$
$\mathbf{l}_j$
$\mathbf{a}_{j,1}$
$d_\perp(\mathbf{a}_{j,i}, \mathbf{l}_j)$
$\mathbf{a}_{j,i}$
Δl_X

Figure 8. Demonstrating the parameters for the expression of line-fitting error (see (30)) in parameter space.

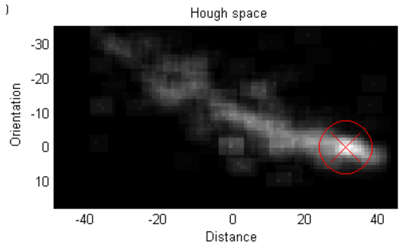
The function $d_\perp(.)$ computes the distance between the line $\mathbf{l}_j$ and point $\mathbf{a}_{j,i}$, while the expected value denoted by $E(.)$. Thus, the guess about the horizon at point j is determined by (29) which describes uncertainty in the parameter space (2D Hough-space).

3.5 Final optimization procedure

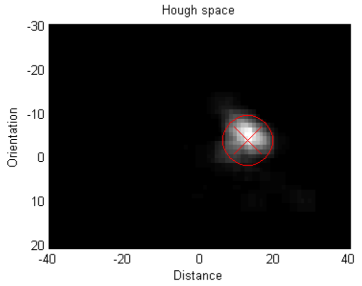
The estimation about the horizon and the estimation error are attached to several points in the image, as we have introduced it in the previous sections. The accurate determination of horizon is carried out in parameter space using all estimations:

$$\mathbf{l}_h = \arg\max_{\mathbf{l}\in(\theta,\rho)} \sum_i P_g(\mathbf{p}_i) P(\mathbf{l} \mid \mathbf{l}_i) \qquad (31)$$

It has been fulfilled with the same optimisation technique as in previous section. The following figure displays the 2D parameter space which has been filled with numerically computed values of (31).



a)



b)

Figure 9: The lower picture in b) depicts the Hough space of indoor, while the upper relates to outdoor scene, respectively.

4. EXPERIMENTAL RESULTS

We performed a practical evaluation of the method in which both indoor and outdoor videos were used as input. The parameters introduced in the previous sections are assigned the following values in empirical fashion: $\sigma_1 = 0.8$ and $\sigma_2 = 1.25$ in (10), while $\alpha = 10^\circ$ in (11). To determine the binary motion mask ($m(t, \mathbf{x})$) a motion-detection method was used which is based on the background model introduced by Stauffer (Stauffer et al., 2000).

The manual extrapolation of the vanishing line is a difficult task, because: i) there are not enough static features for accurate alignment; and ii) the objects are usually too small in case of outdoor images. The outdoor video used for testing shows not only pedestrians, but cars as well; this is why the parameter configurations (distance and orientation of the horizon line) in Hough space show scatter, see figure 9a. The deviation from optimal parameter values is much smaller in indoor case, see figure 9b.

The results demonstrated by straight line in 2D coordinate space after final optimisation of Hough space are displayed in figure 10.

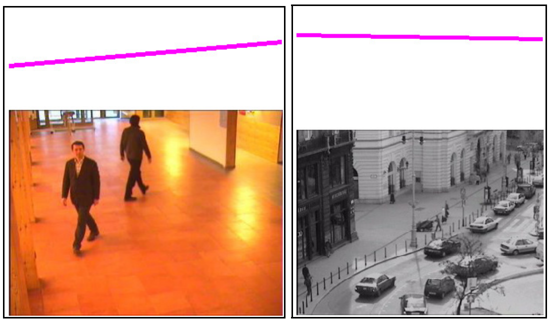


Figure 10: Horizon computation in indoor and outdoor videos.

5. DISCUSSION

An approach for the determination of geometric scene property, namely the horizontal vanishing line (horizon) in image sequences has been introduced. The simple motion statistics are the novel feature used as a basis for estimation of average shape for every point in the image, which provides the necessary height information for the estimation of VPs and finally for the determination of horizon.

We have shown that using the proposed algorithm it is feasible to compute the horizon with good accuracy even from a real-life noisy data set which contains several outliers. The proposed approach executes two statistical parameter optimization steps by using the benefits of error propagation formula.

In future work, we intend to investigate the estimation of vertical vanishing point to accomplish camera calibration task.

6. REFERENCES

Lu, F., Zhao, T. and Nevatia, R, 2000 Self-Calibration of a camera from video of a walking human. *in Proc. of ICPR*

Criminisi, A., Reid, I. and Zisserman, A., 1999 Single view metrology. *in Proc. of ICCV*, pp. 434-442

Hartley, R. and Zisserman, A., 2000 *Multiple View Geometry in Computer Vision*, University Press, Cambridge.

Nguyen, V., Martinelli, A., Tomatis, N. and Siegwart, R., 2005 A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics. *in Proc. of IROS*

Ji, Q. and Xie, Y., 2003 Randomised hough transform with error propagation for line and circle detection. *Pattern Analysis and Applications*, vol. 6, pp. 55-64.

Szlávik, Z., Havasi, L. and Szirányi, T., 2004 Estimation of common groundplane based on co-motion statistics. *in Proc. of Image Analysis and Recognition*, LNCS Springer, New York, pp. 347-353.

Havasi, L. and Szirányi, T., 2006 Estimation of Vanishing Point in Camera-Mirror Scenes Using Video. *Optics Letters*, In Press. http://digitus.itk.ppke.hu/~havasi/pcv/ol.pdf

Kallenberg, O., 1997 *Foundations of Modern Probability*. New York: Springer-Verlag.

Haralick, R.M., 1994 Propagating covariance in computer vision. *in Proc. of ICPR*, pp. 493-498.

Kiryati, N. and Bruckstein, A. M., 2000 Heteroscedastic Hough Transform (HtHT): An Effective Method for Robust Line Fitting in the 'Errors in the Variables' Problem. *Computer Vision and Image Understanding*, vol. 78, pp. 69-83.

Duda, R. O. and Hart, P. E., 1972 Use of the Hough transform to detect lines and curves in pictures. *Comm. ACM*, pp. 11-15.

Stauffer C., Eric W. and Grimson L., 2000 Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 747-757.

Cucchiara R., Grana C., Piccardi M. and Prati A., 2003 Detecting Moving Objects, Ghosts and Shadows in Video Streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1337-1342.

VERIFICATION OF A METHODOLOGY FOR THE AUTOMATIC SCALE-DEPENDENT ADAPTATION OF OBJECT MODELS

Janet Heuwold

Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover
Nienburger Str. 1, 30167 Hannover, Germany – heuwold@ipi.uni-hannover.de

Commission III, WG III/5

KEY WORDS: Interpretation, Model, Scale, Knowledge Base, Multiresolution, Object, Extraction

ABSTRACT:

A methodology for the automatic adaptation of object models consisting of parallel line-type objects parts to a lower image resolution was developed previously. This paper aims at the verification of this algorithm and describes the verification process. The verification is supposed to allow a statement whether the automatically adapted object models produce satisfying object extraction results and are as useful for image analysis in the lower resolution as the original model is in high resolution.

For this purpose, an example system was created comprising the automatic adaptation of a given object model for road extraction to several lower image resolutions as well as the implementation of the original and the adapted object models in a knowledge-based image interpretation system. The paper illustrates the results of the object extraction with the adapted object models and comments on the comparison of these results. At the end of the paper, conclusions concerning the success of the automatic scale-dependent adaptation algorithm are drawn from the verification results.

1. INTRODUCTION

Due to the varying appearance of landscape objects in different image resolutions, an already existing model for image analysis can usually not be used for the extraction of the same object in another resolution. Hence, several models need to be created for the extraction of a landscape object, although the information, how that object looks like in a lower resolution image is already implicitly contained in the model for the highest spatial resolution. This can be assumed, as objects can loose some details in lower resolution images, but usually no new details are added.

The automatic generation of image analysis models for the extraction of landscape objects in aerial and satellite images is a crucial issue of research, as it can reduce tedious manual work [Mayer04]. Methods for the automatic adaptation of image analysis models consisting of parallel line-type object parts to a lower image resolution were developed in order to facilitate the redundant work of object model creation for lower resolutions [Heller&Pakzad05]. A similar algorithm is known in cartography as model generalisation [Sester01]. Generalisation is carried out according to cartographic rules in order to adapt a symbolic appearance of objects in maps in different scales; the algorithm for model adaptation to be verified here, however, requires the prediction of the object's geometric and radiometric appearance in images of reduced resolution.

In the remainder of this paper, the image analysis models to be adapted are called "object models", while they not only describe the relations of the object parts among each other, but also their appearance in the image. According to the categorisation of models given in [Förstner93], the models adapted here integrate both the object model and the image model. The processed object models use the explicit type of representation of semantic networks. In order to enable an automatic adaptation, the object model to be adapted needs to fulfil certain requirements [Pakzad&Heller04]. The developed methods use for the prediction of the appearance of the object in a lower image resolution the concepts of linear scale-space theory, e.g. [Witkin86], [Lindeberg94]. The automatic algorithm for the scale-dependent adaptation of object models represents a new approach for the automatic creation of models in image analysis. Up to now, these methods have not been tested extensively on aerial image data and therefore the new adaptation algorithm could not yet be approved sufficiently. The work presented in this paper strives for the verification of the developed methodology.

In an example system an object model for the extraction of a dual carriageway in very high-resolution images is implemented in the knowledge-based image interpretation system GeoAIDA [Bückner02], [Pahl03]. The model is automatically adapted with the developed methods to three lower spatial resolutions. The adapted object models are also implemented in GeoAIDA and its extraction results are compared to the results, which were gained with the given object model for the high resolution. For the comparison aerial images of a suburban region are used.

Section 2 gives a short summary of the strategy and the methodology of the adaptation algorithm. The concept used here for the verification is described in section 3. The example system including the implementation of the example object model in GeoAIDA and three automatically adapted object models to lower resolutions is presented in section 4. Section 5 compares the extraction results of the original object model and the adapted object models. Conclusions from the results of the verification are derived in section 6.

2. STRATEGY AND METHODOLOGY FOR SCALE-DEPENDENT ADAPTATION

2.1 Strategy

The general strategy for the automatic adaptation of object models can be divided into three main steps that enable the separate scale-space analysis of object parts for the prediction of their scale behaviour while scale changes (cf. Fig.1).

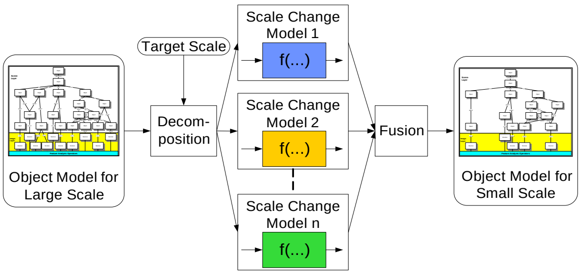


Figure 1. Strategy for Scale-Dependent Adaptation

With knowledge of the target scale, the original object model for high spatial resolution is at first decomposed into object parts with similar scale change behaviour and in neighbouring object parts that interfere each other's appearance in the coarser scale. These groups of object parts are then analyzed separately regarding their scale behaviour. Their appearance in the lower target resolution is predicted by so-called scale change models. At last, all predicted objects are composed back to a complete object model, suitable for the extraction of that object in images of the lower target resolution.

2.2 Methodology

The methodology to be verified here carries out the adaptation of a given object model created for a certain image resolution to a coarser resolution in an automatic algorithm. The automatic methods are based on linear scale-space theory, as the reduction of spatial resolution is a matter of scale change. The analysis is undertaken in scale-space to examine the appearance of object parts in the target resolution. The adaptation process takes into account discrete scale events, which may appear during scale change and affect the structure of the resulting semantic net. For parallel line-type object parts two scale events are relevant: Annihilation (disappearance of objects) and Merging (one or more objects merge into a single object). Besides the scale events, the scale change models automatically predict the resulting attribute values of the object parts in the target resolution as well, thereby adapting the description of the appearance of the object parts in the lower resolution image. The adapted attribute values serve then as new adjusted parameters for the feature extraction operators in the target resolution. For a detailed insight into the developed methods, please see [Heller&Pakzad05].

3. VERIFICATION CONCEPT

A verification of the new methods can decide on the success and usefulness of the developed adaptation algorithm. Thus, the verification method used here not only has to allow a statement on whether the extraction of the object in the respective lower resolution utilising the adapted object model is possible at all, but also on how well the prediction of the objects' appearance in the target resolution is done with the developed algorithm. As the adaptation process naturally changes the object model, a direct comparison of the given model with the adapted model is not reasonable. Rather the extraction results of several adapted object models gained in the respective lower resolutions with the extraction result of the original object model are considered here for verification.

The concept of the verification method applied here is depicted in Fig.2. With both the original model for the high resolution and the adapted object model for the lower resolution the image analysis is carried out on image data with corresponding spatial resolution. In order to ensure comparability of extraction results, it is reasonable to derive the image data utilised for the extraction in lower resolution from the same image scene in high resolution, which is simultaneously used for the extraction of the object with the given high-resolution object model. For this purpose, the image data of the high resolution are at first filtered with a Gaussian low-pass filter in order to avoid aliasing and subsequently down-sampled to the desired spatial resolution by bilinear transformation. The obtained extraction results in both resolutions are then compared to each other. To gain better insight about possible insufficiencies of the automatic adaptation process, the verification is here carried out by incorporating both the whole object and the object part results.

Completeness and correctness regarding the extraction output are used here in the comparison process as a measure for the success of the adaptation methodology. The result of the extraction applying the given object model in the high resolution serves as reference data set, i.e. this extraction outcome represents 100% for both completeness and correctness. By comparing the results of the extraction with the automatically adapted object models in the corresponding image data to the reference data, only the quality of the adaptation algorithm is evaluated. In contrast, the image analysis capability of the adapted object models in regard to an extraction reference set created manually from an aerial image is not subject of this study. The quality of the target object extraction, however, is clearly specified by the high resolution object model itself, which is not verified here.

Because the structure of the object model can change in the adaptation algorithm due to the occurrence of scale-space events, the comparison including object parts is not straight forward. The comparison method of the extraction results in different resolutions needs to consider possible scale events. Generally, the occurred difference can have three main origins. The first is the occurrence of scale events, which can easily be explained by a difference in the structure of the object models, as the scale event should also have been predicted in the adaptation process and therefore be inherent in the adapted object model for the low resolution. Another reason for an

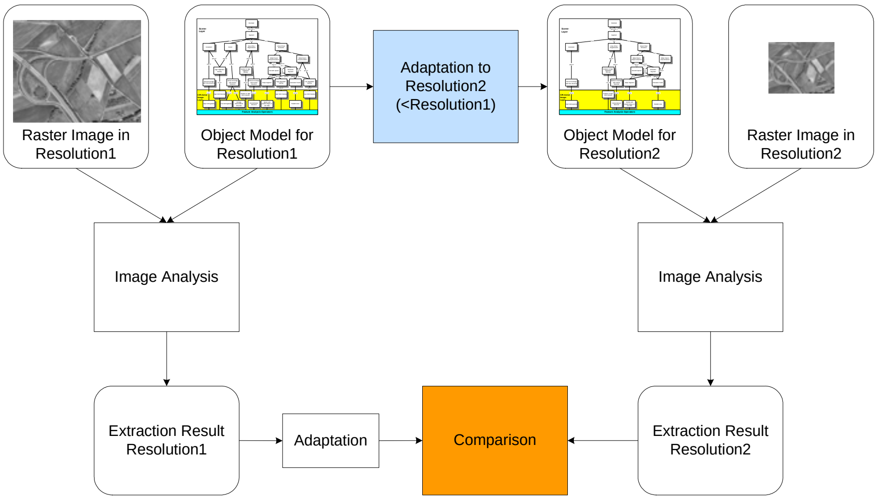

Raster Image in Resolution1
Object Model for Resolution1
Adaptation to Resolution2 (<Resolution1)
Object Model for Resolution2
Raster Image in Resolution2
Image Analysis
Image Analysis
Extraction Result Resolution1
Adaptation
Comparison
Extraction Result Resolution2

Figure 2. Concept of Verification

extraction difference could be the inconsistent performance of the feature extraction operators that are assigned to the object parts. The feature extraction operators carry out the extraction of the object parts and could prove less successful or even fail completely in the lower resolution. In a third scenario, the adaptation of the object model to the coarser scale is incorrect. In this case, the automatic adaptation methodology is erroneous. It is then tried to enhance the quality of the adaptation algorithm by searching for the problem in the methodology and resolve it to obtain a sufficient adaptation result. The verification is then repeated. With this loop the adaptation algorithm is improved.

4. EXAMPLE SYSTEM

4.1 Input Data

4.1.1 Image Data

For the verification process high-resolution aerial image data of a suburban region near Hanover, Germany were used. The images were digitised to 0.033m spatial resolution. In order to ease the verification process and to make its documentation more clear, the images were transformed from colour (RGB) to grey value images. Fig.3 displays the three test images that have been used for the verification. Whereas the first image is relatively simple, the other two images display a curved road and contain disturbances that hinder the extraction of the object parts, e.g. shadows and a non-permanent road work marking, which is not contained in the given example object model as a neighbouring line in the vicinity.

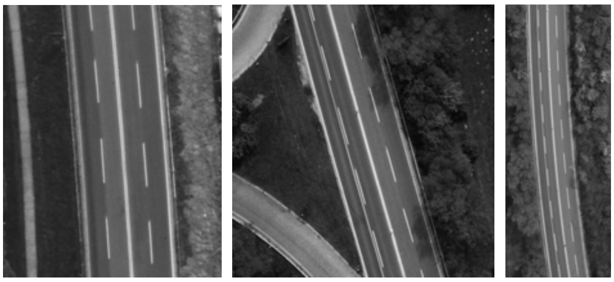


Figure 3. Example Images in 0.033m/pel

4.1.2 Example Road Model

The example object model for the high resolution was created manually for a dual carriageway in images of 0.03-0.04m resolution. Fig.4 displays the given original object model for the high resolution, serving as a starting point for the automatic adaptation. The semantic net is composed of the roadway itself and the road markings, forming nodes, which are part of the road. The uppermost node “roadway” is modelled here as a continuous stripe with a certain grey value and extent, i.e. width of the line-type object. The road markings are either of object type periodic stripe or continuous stripe. A periodic object describes lane markings, which appear as dashed lines in the image. The nodes not only contain the respective object type, but also values for the attributes grey value, extent and periodicity. The specification of the spatial relations and the distances between the object parts are essential for the scale-dependent adaptation process. The distance *d* corresponds to the width of a single lane. All nodes of the net are connected to appropriate feature extraction operators.

The original example object model was adapted with the automatic algorithm to be verified to a spatial resolution of 0.10m. This scale change corresponds to a scale parameter σ=1.0. In the adaptation a scale event was predicted – the Merging of the two central continuous line markings to a single line. Although in the grey value profile there are still two

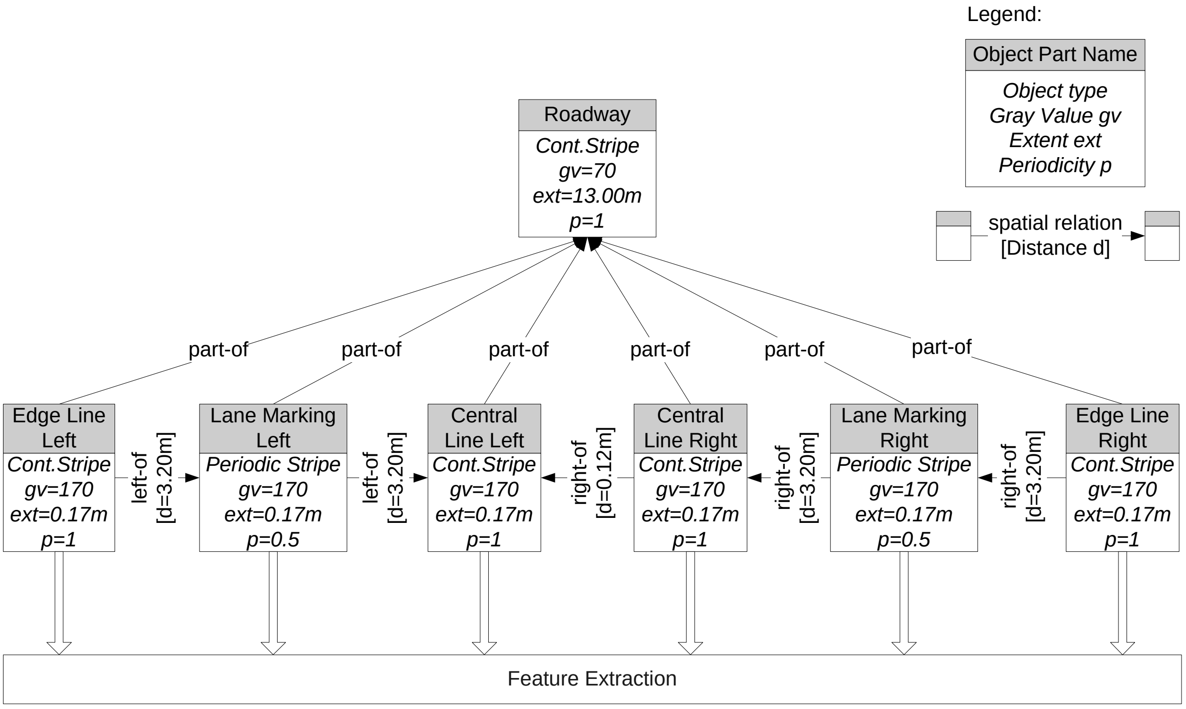


Figure 4. Original Object Model for Dual Carriageway in 0.03m/pel

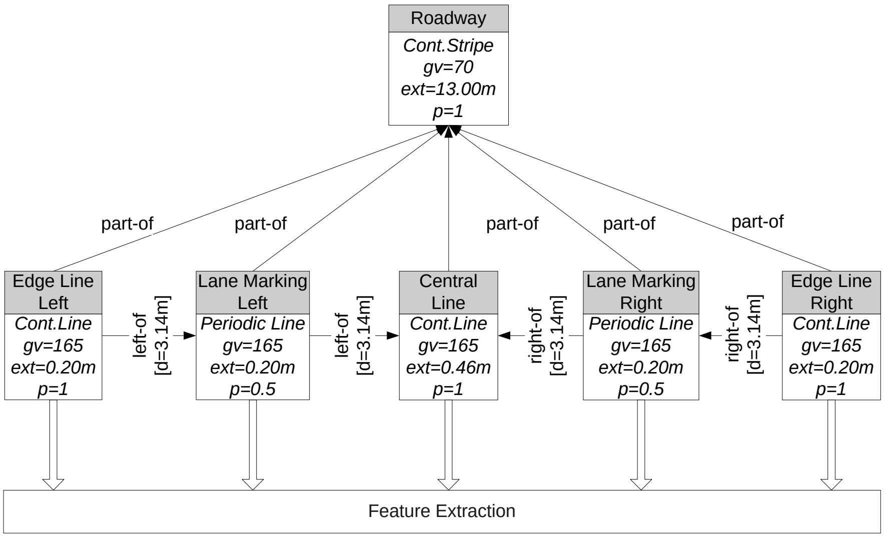
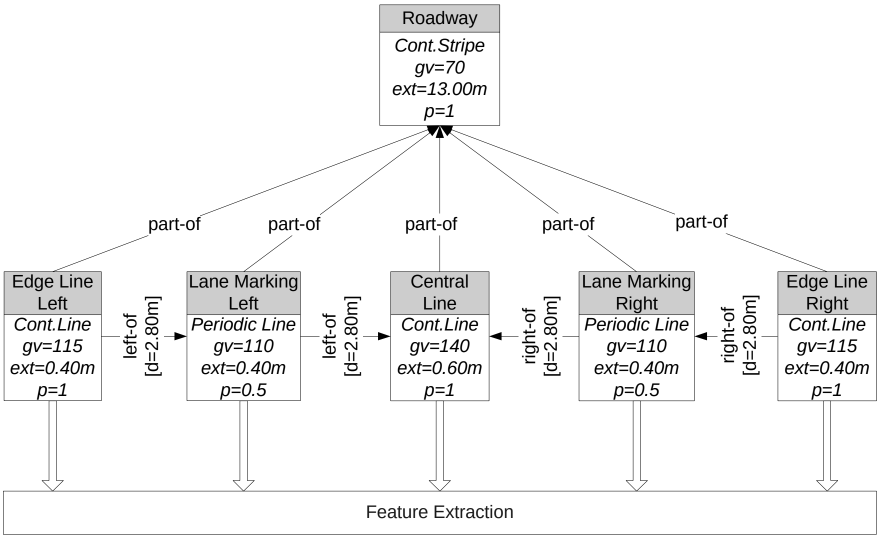


Figure 5. Adapted Object Model for Dual Carriageway 0.10m/pel



distinct maxima present, these two adjacent lines cannot be distinguished reliably from each other anymore in an image resolution of 0.10m by the line extraction operator. Furthermore, the values for the attributes are adjusted due to the slightly different appearance of the object parts (road markings) whose type now changed from stripes to lines in the lower resolution image. The lines appear wider and with less contrast in the images of the lower resolution. The resulting object model for the extraction of the example road in 0.10m resolution images is depicted in Fig.5.

In the scale-dependent adaptation to 0.20m no further scale event were confirmed. However, the central lines now exhibit a definite Merging with only a single maximum in the grey value profile left. The attribute values for grey value and extent of the object parts are adjusted here as well (cf. Fig.6).

As a last target resolution for the verification 1.00m was chosen. For this resolution the adaptation algorithm predicted the failure of the operator for lane markings, resulting in another scale event – the Annihilation of the lane markings. The structure of the semantic net has been altered here significantly with only three extractable road markings left, as can be seen in Fig.7.

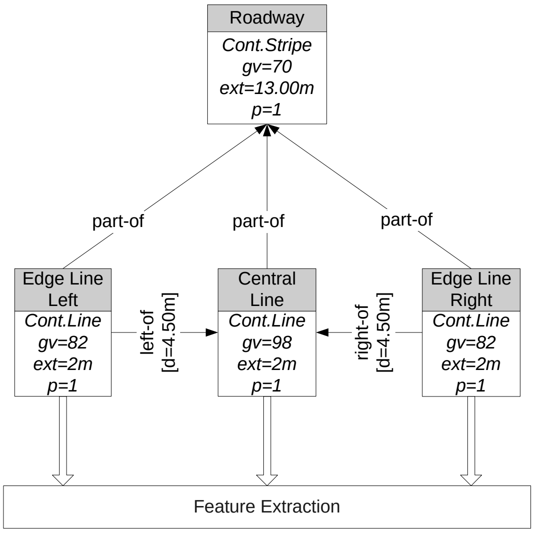


Figure 7. Adapted Object Model for Dual Carriageway 1.00m/pel

4.2 Concept of GeoAIDA

The knowledge-based image interpretation system GeoAIDA was developed at the Institute of Communication Theory and Signal Processing (TNT) at the University of Hannover and represents a tool for image analysis incorporating a priori knowledge in form of semantic nets [Bückner02]. Hypotheses for the existence of the object parts in the semantic net are generated and evaluated in the extraction process. GeoAIDA applies Top-Down- and Bottom-Up-operators. After generating the hypotheses, for each object part the corresponding Top-Down-operator is called, extracting from the input image data the respective object part by image processing algorithms. The output of the Top-Down-operators is then evaluated and grouped to superior objects by the Bottom-Up-operators. For verified hypotheses an instance net with label images for the corresponding instance nodes is created.

4.3 Implementation of the example object model

In the example system the Top-Down-operators carry out the extraction of the road markings, which are modelled as object parts in the example road model. The operators extract edge lines and central lines as continuous lines and lane markings as dashed lines. The operators use the line extraction algorithm of Steger [Steger98], followed by the evaluation and fusion of lines according to [Wiedemann02]. The algorithm of Wiedemann was adapted to the special requirements of the extraction of road markings in high-resolution images [Schramm05]. Ingoing parameters for the road markings extraction are width and contrast of the lines in the image. The operators were designed very flexible in regard to the setting of the parameters, allowing the adjustment of parameters in accordance with varying width and contrast of the markings in different image resolutions.

The Bottom-Up-operators group the extracted lines and evaluate the instances concerning the hypotheses from the semantic net. At first the operators select from the results of the Top-Down-operators those lines with the appropriate attribute values that fit to the ones assigned in the nodes of the semantic net. The lines are then tested for their spatial relations, also considering their distances to each other. Instances fulfilling all the conditions of the relations of the object parts determined in the semantic net are subsequently grouped to a superior object. This superior object is grouped again with appropriate line instances, if the spatial relation to that line is satisfying. This grouping is repeated until all hypotheses for the object parts (road markings) are evaluated. If all hypotheses were accepted, the extraction of the road in the examined image is successful and GeoAIDA creates the label images with the instance nodes.

5. RESULTS

5.1 Reference Data Set 0.033m

The extraction result obtained with the original object model in the cut-out of the example image set serves as reference for the verification of the adaptation algorithm in the example system.
The extraction of all relevant road markings with the Top-Down operators in the example image was successful. All road markings were grouped by the Bottom-Up operators according to the spatial relations assigned in the given object model for 0.033m image resolution. Fig.8 depicts the result of the object extraction with the road markings operators.

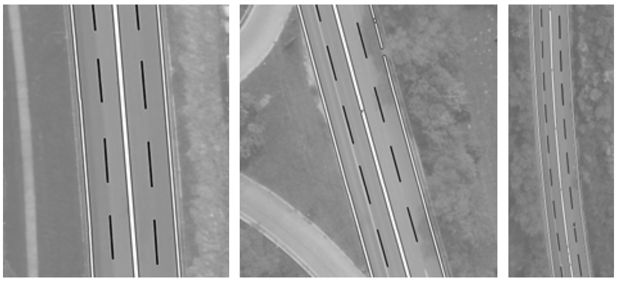


Figure 8. Extraction Results in 0.033m/pel – Reference (white: edge lines, black: lane markings)

5.2 Extraction Results 0.10m

The extraction of the road markings in 0.10m resolution was carried out with adjusted parameter values for contrast and line width taken from the adapted object model. All relevant road markings were found and the grouping was successful. In comparison to the reference data set all object parts were extracted correctly, taking into account the scale event Merging of the central line. The central line was extracted with the edge line operator, but with a different line width parameter taken from the adapted model. The correctness achieved 100% for this target resolution. In contrast, the extraction of the right edge line was not complete in the second image in shadowed image regions.

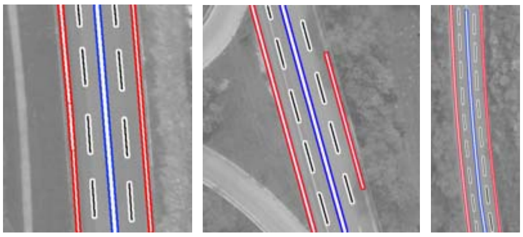


Figure 9. Extraction Results for 0.10m/pel (white/blue: central line, white/red: edge lines, black: lane markings)

5.3 Extraction Results 0.20m

In 0.20m resolution not all the object parts could be extracted 100% completely and correctly with the predicted attributes for contrast and width. The operators had problems in shadow regions and for the left dashed lane marking with the adjacent continuous road work marking in the second and third test image. However, the grouping of the road markings was still successful with the adapted distances between the object parts.

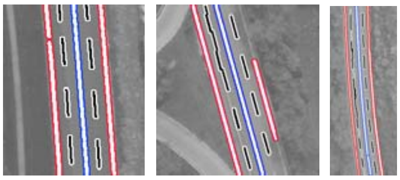


Figure 10. Extraction Results for 0.20m/pel

5.4 Extraction Results 1.00m

For a resolution of 1.00m the predicted entire failure of the lane marking operator is confirmed, although there is still a dashed line with small contrast in the image existent. This Annihilation was predicted correctly by the adaptation algorithm. Due to shadows and low contrast the operator for continuous lines is not successful for all the edge lines in the first two example images (cf. Fig.11). Therefore, not all hypotheses could be verified by the Bottom-Up operators and subsequently the object dual carriageway could not be extracted successfully in these two example images with the feature extraction operators used for a resolution of 1.00m/pel. In the third image all remaining lines were extracted, thereby proving the adaptation algorithm also for 1.00m to be correct.

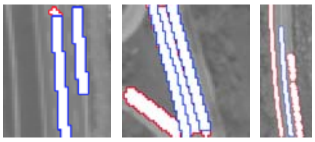


Figure 11. Extraction Results for 1.00m/pel (enlarged)

5.5 Comparison to high resolution extraction results

For the comparison of the extraction results of the lower image resolutions the difference to the reference data set is of interest. In order to estimate the quality of the automatic adaptation the percentage of the difference to the reference set is determined. Table 1 reflects the completeness and correctness values for all object parts in the adapted object model for the three target resolutions. Due to degraded contrast and context objects the completeness suffered in some image regions. The insufficient extraction result regarding completeness for the third examined target resolution of 1.00m can be accounted to the limit of the usability range of the applied feature extraction operators, which can be reduced by disturbing influences, such as shadows or insufficient contrast. This usability range therefore simultaneously defines the scale change limit for the adaptability of the given object model with its assigned feature extraction operators.

	0.10m	**0.20m**	**1.00m**
Completeness	97%	96%	60%
Correctness	100%	90%	100%

Table 1. Completeness and Correctness of extraction results for object parts with adapted object models in target resolutions

6. CONCLUSIONS AND OUTLOOK

A method for the verification of a previously developed algorithm for the automatic adaptation of object models was presented, enabling the assessment of the success of the developed algorithm. The results of the verification lead to the conclusion that an automatic scale-dependent adaptation exploiting linear scale-space theory is generally possible. The prediction of scale events of object parts occurring during scale change could be confirmed being correct by the verification process. In the verification process the tested algorithm can be improved, correcting unforeseen shortcomings.

The verification results also revealed the sensitivity of the adaptation algorithm to the assigned feature extraction operators. The assigned feature extraction operators in the original object model should be easily adaptable to another resolution by parameters corresponding to the attributes in the nodes of the adapted object model. Otherwise, the operators might fail already for a relatively small change in image resolution. This flexibility is desirable in order to enlarge the range of image resolution, for which the adaptation with the examined methodology will be successful. The performance of the operators can also be degraded or even lead to a complete failure to extract the object due to disturbances in the images, such as shadow or local context objects. This limitation could be overcome by extending the adaptation algorithm in regard to the incorporation of local context in the adaptation process. This algorithm extension is therefore a goal for the near future.

For further future tasks, a test of the algorithm for satellite image resolution (up to 5m) is intended by using the feature extraction operators of the continuous road markings for the extraction of the roadway, as roads possess the same object type in satellite images.

7. ACKNOWLEDGEMENTS

The development of the methodology for the adaptation of object models is funded by the German Research Foundation (DFG) under grant HE 1822/13-1.

The implementation of the object models in the knowledge-based image interpretation system GeoAIDA was substantially supported by M. Gerke, Institute of Photogrammetry and GeoInformation (IPI). This help is gratefully acknowledged.

8. REFERENCES

Bückner, J., Pahl, M., Stahlhut, O. and Liedtke, C.-E., 2002. A Knowledge-Based System for Context Dependent Evaluation of Remote Sensing Data. In: *LNCS*, Vol. 2449, DAGM2002, Springer Verlag, Berlin, pp. 58-65.

Förstner, W., 1993. A future of photogrammetric research. *NGT Geodesia*, Vol. 93, No. 8, pp. 372-383.

Heller, J. and Pakzad, K., 2005. Scale-Dependent Adaptation of Object Models for Road Extraction. In: *IntArchPRS*, Vol. XXXVI, Part 3/W24, Vienna, pp. 23-28, CMRT05.

Lindeberg, T., 1994. *Scale-Space Theory in* Computer *Vision.* Kluwer Academic Publishers, The Netherlands, 423 p.

Mayer, H., 2004.Object Extraction for Digital Photogrammetric Workstations. In: *IntArchPRS*, Vol. XXXV, Part B2, Istanbul, pp. 414-422.

Pahl, M., 2003. Architektur eines wissensbasierten Systems zur Interpretation multisensorieller Fernerkundungsdaten. *Schriftenreihe des TNT der Universität Hannover*, Vol. 3 ibidem-Verlag, Stuttgart, 145 p.

Pakzad, K. and Heller, J., 2004. Automatic Scale Adaptation of Semantic Nets. In: *IntArchPRS*, Vol. XXXV, Part B3, Istanbul, pp. 325-330.

Schramm, M., 2005. *Untersuchungen zum Skalenverhalten von Bildanalyse-Operatoren zur automatischen Extraktion von Fahrbahnmarkierungen*. Diploma Thesis, Institute of Photogrammetry and GeoInformation, University of Hannover, 66 p.

Sester, M., 2001. Maßstabsabhängige Darstellungen in digitalen räumlichen Datenbeständen. Habilitation, *DGK, Series C*, No. 544, Munich, 114p.

Steger, C., 1998. An Unbiased Detector of Curvilinear Structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2), pp.113-125.

Wiedemann, C., 2002. Extraktion von Straßennetzen aus optischen Satellitenbilddaten. Dissertation, *DGK, Series C*, No. 551, Munich, 94 p.

Witkin, A., 1986. Scale space filtering. In: Pentland, A. (Ed.), *From Pixels to Predicates.* Ablex Publishing Corporation, New Jersey, pp. 5-19.

GENETIC ALGORITHMS FOR THE UNSUPERVISED CLASSIFICATION OF SATELLITE IMAGES

Y. F. Yang [a]*, P. Lohmann [b], C. Heipke [b]

[a] Dept. of Civil Engineering, National Chung Hsing University, 250 Kuokuang Road Taichung, Taiwan 402, R.O.C - d9062503@mail.nchu.edu.tw
[b] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Nienburger Str. 1, D-30167 Hannover, Germany - (lohmann, heipke)@ipi.uni-hannover.de

Commission III

KEY WORDS: Remote Sensing Analysis, Unsupervised Classification, Genetic Algorithm, Davies-Bouldin's Index, Heuristic Algorithm, ISODATA

ABSTRACT:

Traditionally, an unsupervised classification divides all pixels within an image into a corresponding class pixel by pixel; the number of clusters usually needs to be fixed a priori by a human analyst. In general, the spectral properties of specific information classes change with the seasons, and therefore, the relation between object class and spectral cluster is not constant over time. In addition, relations for one image can in general not be extended to others. Thus, even if the number of clusters is correctly fixed for one image at one instance in time, the results cannot be transferred to other areas or epochs.

In this study, a heuristic method based on Genetic Algorithms (GA) is adopted to automatically determine the number of cluster centroids during unsupervised classification. The optimization is based on the Davies-Bouldin Index (DBI). A software programme was developed in MATLAB, - and the GA unsupervised classifier was tested on an IKONOS satellite image. The classification results were compared to conventional ISODATA results, and to ground truth information derived from a topographic map for the estimation of classification accuracy.

1. INTRODUCTION

1.1 Background on unsupervised classification

Image classification, including supervised and unsupervised classification, is an established analytical procedure of digital image processing (Lillesand and Kiefer, 2000). Supervised classification procedures require a human analyst to provide training areas, which form a group of pixels with known class label, so as to assemble groups of similar pixels into the correct classes (Avery and Berlin, 1992). In comparison, unsupervised classification proceeds with only minimal input. An unsupervised classification divides all pixels within an image into a corresponding class pixel by pixel. Typically, the only input an unsupervised classification needs is the number of classes of the scene. However, this value is usually not known a priori. Moreover, the spectral properties of specific classes within the images can change frequently and the relationships between the object classes and the spectral information are not always constant, and once defined for one image cannot necessarily be extended to others. Supervised and unsupervised classification suffers from these drawbacks.

Heuristic unsupervised classification works by establishing some mathematical model and then optimising a predefined index to determine the cluster numbers and centroids automatically. Heuristic optimization processes, therefore, are seen as a repeatable, accurate, and time-effective method to classify remote sensing imagery automatically, which is the main objective of this research. Genetic algorithms (GA) constitute one possibility for heuristic unsupervised classification. GA -have already been adopted successfully in image processing (Kawaguchi, et al., 1997), and image recognition for some special purposes such as medical treatment or criminal offence investigations (Caldwell and Johnston, 1991; Yang, et al., 2000). In this study, GA is adopted to determine number of cluster centroids of an image for use in unsupervised classification.

1.2 Status of research applying genetic algorithms

Genetic algorithms, introduced by John Holland in 1975 (Coley, 1999; Pham and Karaboga, 2000), are numerical optimisation algorithms inspired by the nature evolution process and directed random search techniques. In many fields, such as the analysis of time series, water networks, work scheduling, and facial recognition, GA have been successfully applied (Coley, 1999; Rothlauf, 2006). In 1975, De Jong (1975) executed a number of tests to study the effect of the various control parameters concerning the performance of GA. In this research, suitable values were defined, such as population size, crossover probability, and the mutation probability (Pham and Karaboga, 2000). In 2001, Bandyopadhyay and Maulik (2001) applied GA to cluster different man-made experimental point data sets and obtained very good results.

* Corresponding author. The research reported here was carried out in 2005 and 2006 while Y.F. Yang was with the Institute of Photogrammetry and GeoInformation, Leibniz University of Hannover.

2. BASES OF GENETIC ALGORITHM

The genetic algorithm is a method, which is suitable for solving an extremely wide range of problems (Coley, 1999). Recently, GA has been widely and successfully applied to optimization problems specifically in unsupervised classification of digital data sets (Ross, 1995; Bandyopadhyay and Maulik, 2002). The following sections describe the general operation of GA.

2.1 Chromosome representation

In GA applications, the unknown parameters are encoded in the form of strings, so-called chromosomes. A chromosome is encoded with binary, integer or real numbers. Since multi-spectral image data are usually represented by positive integers, in this research a chromosome is encoded with a unit (tuple) of positive integer numbers. Each unit represents a combination of brightness values, one for each band, and thus a potential cluster centroid.

The length of the chromosome, *K*, is equivalent to the number of clusters in the classification problem. *K* is selected from the range [K_{min}, K_{max}], where K_{min} is usually assigned to 2 unless special cases are considered (Bandyopadhyay and Maulik, 2002), and K_{max} describes the maximum chromosome length, which means the maximum number of possible cluster centroids. K_{max} must be selected according to experience.

Without assigning the number of clusters in advance, a variable string length is used. Invalid (non-existing) clusters are represented with negative integer "-1". The values of the chromosomes are changed in an iterative process to determine the correct number of clusters (the number of valid units in the chromosomes) and the actual cluster centroids for a given classification problem.

2.2 Chromosome initialization

A population is the set of chromosomes. The typical size of the population can range from 20 to 1000 (Coley, 1999). In the following an example is given to explain the creation of an initial population: we assume to have a satellite image with three bands, K_{min} is set to 2 and K_{max} to 8. At the beginning, for each chromosome *i* (*i* =1, 2,….*P*, where *P* is the size of population) all values are chosen randomly from the data space (universal data set; here: positive integers). Such a chromosome belongs to the so-called parent generation. One (arbitrary) chromosomes of the parent generation is given here:

-1 (110, 88, 246) (150, 78, 226) -1 (11, 104, 8) (50, 100, 114) -1 (227, 250, 192)

2.3 Crossover and Mutation

2.3.1 Crossover: The purpose of the crossover operation is to create two new individual chromosomes from two existing chromosomes selected randomly from the current population. Typical crossover operations are one-point crossover, two-point crossover, cycle crossover and uniform crossover. In this research, only the simplest one, the one-point crossover was adopted; the following example illustrates this operation (the point for crossover is after the 4th position):

Parent1 : ***-1 (110, 88, 246) (150, 78, 226) -1 (11, 104, 8) (50, 100, 114) -1 (227, 250, 192)***

Parent2 : (210, 188, 127) (110, 88, 246) -1 -1 (122, 98, 45) -1 (98, 174, 222) (125, 101, 233)

Child1 : ***-1 (110, 88, 246) (150, 78, 226) -1*** (122, 98, 45) -1 (98, 174, 222) (125, 101, 233)

Child2 : (210, 188, 127) (110, 88, 246) -1 -1 ***(11, 104, 8) (50, 100, 114) -1 (227, 250, 192)***

2.3.2 Mutation: During mutation, all the chromosomes in the population are checked unit by unit and according to a pre-defined probability all values of a specific unit may be randomly changed. An example explains this procedure; the bold-faced and italic units represent the result of the mutation.

Old string: (210, 188, 127) (***110, 88, 246***) -1 -1 (122, 98, 45) -1 (98, 174, 222) (125, 101, 233)

New string: (210, 188, 127) (***97, 22, 143***) -1 -1 (122, 98, 45) -1 (98, 174, 222) (125, 101, 233)

2.4 Indices identification

Based on crossover and mutation the chromosomes, once initialised, iteratively evolve from one generation to the next. In order to be able to stop this iterative process, a so-called fitness function needs to be defined to measure the fitness or adaptability of each chromosome in the population. The population then evolves over generations in the attempt to maximize the value of fitness, also called *index*.

Previous research used different indices, such as distance, separation index, Fuzzy C-Means, K-means, Davies-Bouldin Index (DBI), and Xie-Beni Index (XBI), as criteria to determine the best clustering (Ross, 1995; Bandyopadhyay and Maulik, 2002). Here, the DBI was adopted, because it is not as complex as fuzzy C-Means and one can obtain better results than with some other indices as shown using simulated data (Bandyopadhyay and Maulik, 2002; Yang and Wu, 2001). For the reasons of comparison, we also used the ISODATA algorithm.

3. METHODOLOGY

3.1 GA application of unsupervised classification

In the following paragraphs we explain the application of GA within unsupervised classification of satellite imagery. In particular, each GA procedure (such as reproduction, crossover, and mutation) is described.

3.1.1 Parent generation and population size: This procedure is an operation to produce the cluster centroids including the initial cluster centroids, which are selected randomly. This step is identical to the example given above. The range [K_{min}, K_{max}] equals [2, 8]. Two population sizes were used in or research: 40 and 100.

3.1.2 Crossover: Crossover is considered, according to the crossover probability, for example, if there are 100 chromosomes (population size 100), and the crossover probability is 0.8, the best 80 chromosomes (according to some index) are chosen for the crossover pool. The next generation (the new 100 chromosomes) are then only produced from the 80 old chromosomes of this pool.

3.1.3 Mutation: Mutation is a parameter for extending the search space; therefore, the time to reach a convergent solution increase with an increase of the mutation probability. According to the suggestion of Schaffer et al., in 1989 (Pham and Karaboga, 2000), the mutation probability is set to 0.005 here.

3.2 The Davies-Bouldin's Index

In this research, the Davies-Bouldin index (DBI) is used to represent the fitness of a chromosome (Xie and Beni, 1991; Bezdek and Pal, 1998; Swanepoel, 1999; Martini and Schöbel, 2001; Yang and Wu, 2001; Groenen and Jajuga, 2001).
First, each pixel x_n of the whole image is assigned to the nearest cluster centroid of the given chromosome, see Eq. (1):

$$\mu_{kn} = \begin{cases} 1; & \|x_n - u_k\| \leq \|x_n - u_j\| \ , \\ 0; & otherwise \end{cases} \quad 1 \leq k, j \leq K; \ \ j \neq k; \ \ 1 \leq n \leq N \tag{1}$$

where x_n = pixel n with grey values x (one for each band)
N = total number of pixels
u_k = grey values of k^{th} cluster centroid of the previous iteration (=generation)
K = total number of clusters
μ_{kn} = membership function of each pixel x_n belonging to the k^{th} cluster

Next, the average and the standard deviation for each cluster and for the current iteration are computed (Eq. (2) and (3), followed by determining the Minkowski distance between the clusters (Eq. (4))):

$$v_k = \frac{\sum_{n=1}^{M} (\mu_{kn}) x_n}{\sum_{n=1}^{M} (\mu_{kn})} = \frac{\sum_{x_n \in X_k} x_n}{M_k} \qquad 1 \leq k \leq K \tag{2}$$

where v_k = average value of k^{th} cluster in the current iteration
M_k = the number of pixels belonging to the k^{th} cluster

$$S_k = \Big(\frac{1}{|X_k|} \sum_{x \subset X_k} \|x - v_k\|^2\Big)^{1/2} \tag{3}$$

where S_k = standard deviation of the pixels in the k^{th} cluster

$$d_{kj,t} = \|v_k - v_j\|_t \tag{4}$$

where $d_{kj,t}$ = Minkowski distance of order t between the k^{th} and j^{th} centroids. Here 2 has been chosen for t.

Subsequently, the value $R_{k,t}$ of the k^{th} cluster can be computed as Eq. (5):

$$R_{k,t} = \max_{j, j \neq k} \left\{ \frac{S_k + S_j}{d_{kj,t}} \right\} \tag{5}$$

The *DB* value is then defined as the average of *R* for all clusters in the chromosome (Eq. (6)):

$$DB = \frac{1}{K} \sum_{k=1}^{K} R_{k,t} \tag{6}$$

$$Min \qquad DBI = 1/DB \tag{7}$$

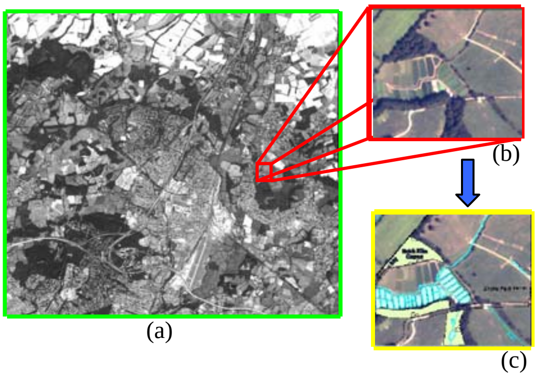
The goal for achieving a proper clustering is to minimize the DBI (Eq. (7)). Thus, the fitness function for chromosome *j* is defined as $1/DB_j$, which is equivalent to the clustering with the smallest inner-cluster scatter and the largest cluster separation.
After calculating the DBI of each chromosome of a given population, the best chromosome is compared to the best one of the previous generation (iteration). The termination condition for the iterations is that the difference between these two values lies below a pre-defined threshold. If this condition is not met, the best chromosomes are selected into the crossover pool (see above) and a new iteration is started. The computations are also stopped once a maximum number of generations is reached.

3.3 Influence of crossover and mutation probabilities

There are five factors that influence the result of a GA algorithm: the encoding form (binary, real number and so on), the size of the initial population, the fitness function, the genetic operations (such as the one-point crossover, two-points crossover, etc.), and the probabilities for crossover and mutation (Pham and Karaboga, 2000). In this research, variations of the initial population size and the crossover probability are discussed.

3.4 Image data, ground truth and error matrices

For our research we used a multi-spectral IKONOS image. The image depicts Chandlers Ford in the U.K. and, was taken on 2000/08/25 with 4 meters pixel size and 11 bits per pixel (see Figure 1). We used a subset with a total of 18330 pixels. A higher resolution map served as a reference for obtaining ground truth information.
We measure classification success using the well-known criteria *producer's accuracy or completeness* (the number of pixels that are correctly assigned to a certain class divided by the total number of pixels of that class in the reference data) and *user's accuracy or correctness* (the number of pixels correctly assigned to a certain class divided by the total numbers of pixels automatically assigned to that class).


(a)

(b)

(c)

Figure 1. (a) The original IKONOS image; (b) The extracted IKONOS subset image; (c) Ground truth map superimposed on the subset image.

4. ANALYSIS RESULTS

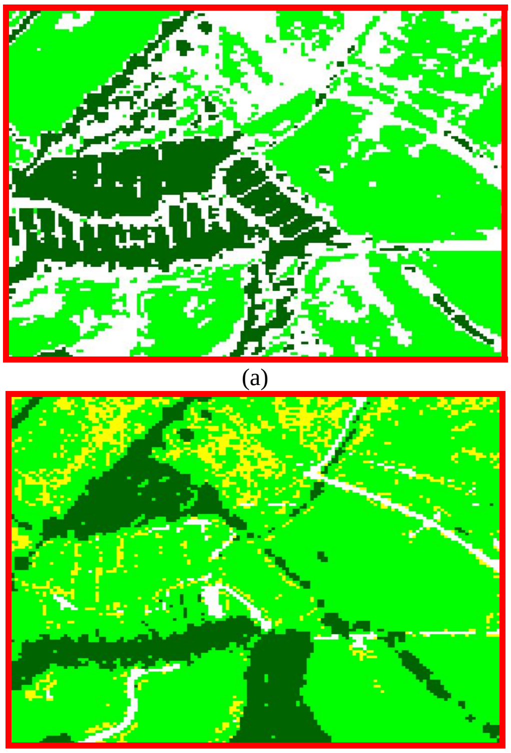
In this section, we present the results of our research. The following parameters of the GA classifier were set:

1.	chromosome length	8
2.	single point crossover	
3.	crossover probability	0.4 and 0.8
4.	population size	40 and 100
5.	mutation probability	0.005

In the ground truth data four distinct classes can be found: *road*, *farmland*, *forest*, and *others*. Figures 2 and 3 show the results with one colour per class: *road* in white, *farmland* in light green, *forest* in dark green, and *others* in yellow. The error matrices of the four experiments are shown in Tables 1 to 4.

Compare Figure 2 (a) with Figure 2 (b) and Table 1 (a) and (b), when the population size increases, the overall accuracy increases from 49.1% to 69.8% and four instead of only three classes are found. The same effects are evident from Figure 3 (a) and 3 (b) and Table 2: the overall accuracy increases from 54.4% to 71.1% and again four classes can be detected with a population size of 100. When comparing the effect of the two investigated parameters, it is clear that the population size is significantly more important than the mutation probability. With a few exceptions, most notably the completeness of roads, the producer's and the user's accuracy all increase when increasing the population size.

As a reference, Figure 4 and Table 3 depict the results of the traditional ISODATA with four classes as prior information. The results of the GA are better (taking the higher population size) than the ISOADATA results; it should be mentioned, however, that the computational expense for GA is significantly larger than that for the ISODATA algorithm.


(a)

(b)

Figure 2. Results with (a) population size 40, and crossover probability 0.4; (b) population size 100 and crossover probability 0.4

		Reference Data			
		Road	Farmland	Forest	Other
Classification	Road	77.7%	31%	49.7%	52%
	Farmland	7%	52.6%	12.6%	48%
	Forest	15.3%	16.4%	37.7%	0%
	Other	0	0%	0%	0%

Producer's Accuracy (Completeness)	**User's Accuracy (Correctness)**
Road=77.7%	Road=14%
Farmland=52.6%	Farmland=87.4%
Forest=37.7%	Forest=33.5%
Other=0%	Other=0%

Overall accuracy=49.1%

(a)

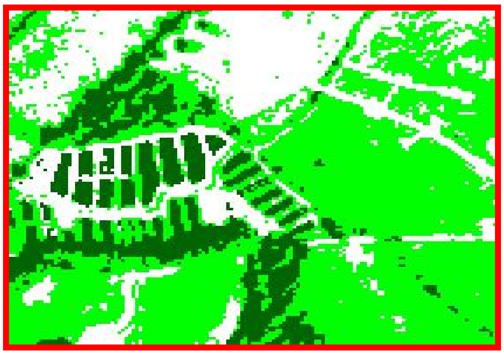
		Reference Data			
		Road	Farmland	Forest	Other
Classification	Road	33.3%	5.5%	1.4%	38.6%
	Farmland	50.2%	76.5%	22.9%	54.7%
	Forest	5.3%	6.9%	74.7%	0.4%
	Other	11.2%	11.1%	1%	6.3%

Producer's Accuracy (Completeness)	**User's Accuracy (Correctness)**
Road=33.3%	Road=35.9%
Farmland=76.5%	Farmland=81.4%
Forest=74.7%	Forest=79.3%
Other=6.3%	Other=0.8%

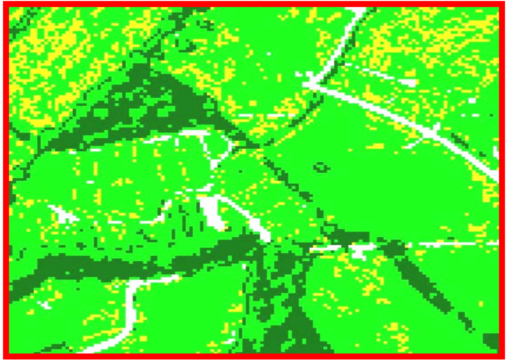
Overall accuracy= 69.8%

(b)

Table 1. (a) and (b). Error matrices for results depicted in Figure 2



(a)



(b)

Figure 3. Results with (a) population size 40, and crossover probability 0.8; (b) population size 100 and crossover probability 0.8

		Reference Data			
		Road	Farmland	Forest	Other
Classification	Road	77.1%	33.2%	5%	88.8%
	Farmland	19.2%	57.6%	42.3%	7.2%
	Forest	3.7%	9.2%	52.6%	0%
	Other	0%	0%	0%	0%

Producer's Accuracy (Completeness)	**User's Accuracy (Correctness)**
Road=77.1%	Road=16.4%
Farmland=57.6%	Farmland=82.9%
Forest=52.6%	Forest=56.7%
Other=0%	Other=0%

Overall accuracy=54.4%

(a)

		Reference Data			
		Road	Farmland	Forest	Other
Classification	Road	38.4%	3.2%	1.9%	45.3%
	Farmland	49.6%	79.5%	39.2%	48.9%
	Forest	7.3%	4.5%	57.9%	0%
	Other	4.7%	12.7	1%	5.8%

Producer's Accuracy (Completeness)	**User's Accuracy (Correctness)**
Road=38.4%	Road=44.4%
Farmland=79.5%	Farmland=84.5%
Forest=57.9%	Forest=72.8%
Other=5.8%	Other=0.6%

Overall accuracy= 71.1％

(b)

Table 2. (a) and (b). Error matrices for results depicted in Figure 3

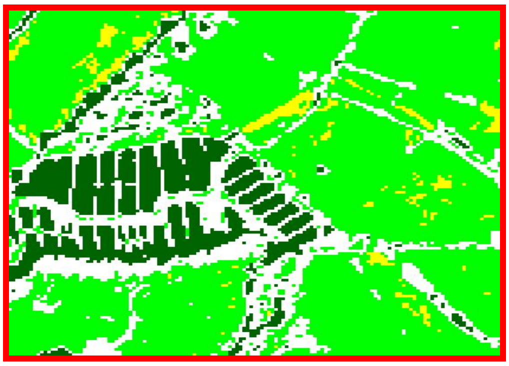


Figure 3. Results of ISODATA algorithm (4 clusters)

		Reference Data			
		Road	Farmland	Forest	Other
Classification	Road	64%	13.9%	52.2%	69.1%
	Farmland	27.2%	70%	26.4%	9%
	Forest	8.4%	12.5%	20.7%	21.5%
	Other	0.4%	3.6%	0.7%	0.4%

Producer's Accuracy (Completeness)	**User's Accuracy (Correctness)**
Road=64%	Road=17.5%
Farmland=70%	Farmland=88.7%
Forest=20.7%	Forest=33.6%
Other=0.4%	Other=0.2%

Overall accuracy= 65.1%

Table 3. Error matrix from ISODATA results

5. CONCLUSION

One of the a priori inputs traditionally needed for unsupervised classification is the number of clusters in the data set. In many cases, however, this number of classes is not available. This research describes a procedure for unsupervised classification based on genetic algorithms, which is able to estimate the required number of clusters as part of the procedure. In order to evaluate the individual results we used the Davues-Bouldin's Index (DBI).

The effectiveness of the new technique was evaluated using examples of IKONOS satellite image data. Based on independent ground truth an overall accuracy of 71.1% was reached as compared to 65.1% when using the ISODATA algorithm. For a number of applications this accuracy is still acceptable.

GA has a number of free parameters. Two of them, namely population size and the crossover probability were considered in this research. In our results the population size proofed to be significantly more important than the crossover probability. In future research we will further investigate the potential influence of the other parameters and also consolidate our results using more test data and alternative indices for measuring the chromosome fitness.

6. REFERENCES

Avery, T.E. and Berlin, G.L., 1992. *Fundamentals of remote sensing and airphoto interpretation.* MacMillan Publishing Company, New York, 472 p.

Bandyopadhyay, S., and Maulik, U., 2001. Nonparametric genetic clustering: comparison of validity index. *IEEE Transactions on systems man, and cybernetics-part C: Applications and reviews*, 31(1), pp.120-125.

Bandyopadhyay, S., and Maulik, U., 2002. Genetic clustering for automatic evolution of clusters and application to image classification. *IEEE pattern recognition*, Vol.35, pp.1197-1208.

Bezdak, J.C., and Pal, N.R., 1998. Some new indexes of cluster validity. *IEEE Transactions on systems, man, and cybernetics*, part B, 28(3), pp.301-315.

Caldwell, C., and Johnston, V.S., 1991. Tracking a Criminal Suspect Through "Face-Space" with a Genetic Algorithm. *Proceedings of the 4th International Conference on Genetic Algorithms*, Morgan Kaufmann, pp.416-412.

Coley, A D., 1999. *An Introduction to Genetic Algorithms for Scientists and Engineers.* World Scientific, Singapore, 188p.

De Jong, K.A., 1975. *An analysis of the behavior of a class of genetic adaptive systems*. Ph.D dissertation, University of Michigan, Ann Arbor, Michigan.

Groenen, P.J.F., Jajuga, K., 2001. Fuzzy clustering with squared Minkowski distances. *Fuzzy Sets and Systems*, Vol.120, pp.227-237.

Kawaguchi, T., Baba, T., Nagata, R., 1997. 3-D object recognition using a genetic algorithm-based search scheme, *IEICE transactions on information and systems*, E80D(11), pp.1064-1073.

Lillesand, T.M. and Kiefer, R.W., 2000. *Remote Sensing and Image Interpretation.* John Wiley & Sons, New York. 724 p.

Martini, H., and Schöbel, A., 2001. Median and center hyperplanes in Minkowski spaces -- a unified approach. *Discrete Mathematics*, Vol.241(1), pp.407-426.

Pham, D.T., and Karaboga, D., 2000. *Intelligent Optimisation Techniques*. Springer, London, Great Britain, 261p.

Ross, T.J., 1995. *Fuzzy logic with engineering applications.* Mc Graw-hill, Singapore, 592p.

Rothlauf, F., 2006. *Representations for Genetic and Evolutionary Algorithms*. Springer, Netherlands, 314p.

Swanepoel, K.J., 1999. Cardinalities of k-distance sets in Minkowski spaces. *Discrete Mathematics*, 197(198), pp.759-767.

Xie, X.L., and Beni, G., 1991. A Validity Measure for Fuzzy Clustering. *IEEE Transaction on Pattern Analysis and Machine Inteligence*, 13(8), pp. 841-847.

Yang, G., Reinstein, L.E., Pai, S., Xu, Z., Carroll, and D.L., 2000. A new genetic algorithm technique in optimization of prostate implants. *Medical Physics*, 35(5), pp.104-112.

Yang, M.S., and Wu, K.L., 2001. A new validity index for fuzzy clustering. *IEEE International Fuzzy Systems Conference*, pp.89-92.

7. ACKNOWLEDGEMENTS

We would like to thank the Ordnance Survey of United Kingdom to offer the IKONOS image data within the OEEPE test and National Scientific council (NSC) of Taiwan Taipei for subsidizing this research.

EXTRACTION OF BRIDGES OVER WATER IN MULTI-ASPECT HIGH-RESOLUTION INSAR DATA

U. Soergel[1], H. Gross[2], A. Thiele[2], U. Thoennessen[2]

[1]Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, 30167 Hannover, Germany
[2]FGAN-FOM Research Institute for Optronics and Pattern Recognition, 76275 Ettlingen, Germany

soe@fom.fgan.de

Commission III

KEY WORDS: SAR, Infrastructure, Building, Detection, Reconstruction, Urban

ABSTRACT:

Modern airborne SAR sensors provide spatial resolution in the order well below half a meter. In such data many features of urban objects are visible, which were beyond the scope of radar remote sensing only a few years ago. Core elements of urban infrastructure are bridges. In high-resolution InSAR data even small bridges are mapped to extended data regions covering large numbers of pixels. Therefore, in data of this quality the identification of bridge structure details is possible at least by visual interpretation. The special appearance of bridges over water in high-resolution InSAR data is discussed. Geometric constraints for the mapping of certain bridge elements in interferometric SAR imagery are given. An approach for detection of such bridges is proposed. Information about the bridge structure is extracted in subsequent fine analysis. First results of the approach are demonstrated using orthogonal InSAR single-pass data sets of spatial resolution better than 40cm.

1. INTRODUCTION

In time critical events SAR can be the most appropriate remote sensing technique for gathering useful actual data under certain circumstances such as bad weather or at night-time. For example, satellite SAR has proven being suitable for flooded area detection and damage assessment purposes [Bach et al., 2005]. Due to climate change, flooding events of unfortunately even increasing devastation capability are more frequently observed in many places of the earth [International Charter, 2006]. Given the rather coarse resolution of operational SAR satellite systems, up to now the SAR analysis was mostly restricted to medium scale products, such as flood maps (e.g. Elbe flooding, 2006, [International Charter, 2006]). With the advent of high resolution SAR satellite systems and commercial airborne systems more detailed analysis at the object level becomes feasible. This was already studied for example in the context of building recognition [Soergel et al., 2003] and road extraction [Hedman et al., 2005] from SAR imagery.

Bridges are key elements of man-made infrastructure. Monitoring of these important connecting parts of the traffic network is vital for applications such as disaster management or in the context of political crisis, e.g. to evacuate inhabitants and to deliver goods and equipment.

In this paper, first results of a long-term project are presented, which aims at automatic detection and reconstruction of bridges in Interferometric SAR (InSAR) data of fine spatial resolution. Here, the focus is on bridges over water. In later project phases the investigation shall be expanded to other bridge types too. Compared to coarser SAR images in high-resolution SAR data of modern sensors many additional bridge structure features are observable, allowing better discrimination from other urban objects and higher level of detail in object recognition. Urban analysis does not only benefit from higher resolution of conventional amplitude SAR imagery. In addition, the capability of SAR to measure the 3D shape of scene topography by interferometric processing offers valuable features to distinguish man-made objects of different kinds [Soergel et al., 2003]. For example, bridges are naturally higher than surrounding ground and they coincide with an orthogonal orientated stripe of low signal amplitude and poor coherence, if they span a river. Additionally, the strong aspect dependency of SAR, due to the oblique scene illumination principle, leads to very interesting effects at bridges over water.

Under certain viewing conditions different types of scattering events lead to the appearance of several bridge images at different range locations [Raney, 1983; Raney, 1998; Robalo & Lichtenegger, 1999]. These images are mainly caused by direct backscatter, double-bounce reflection, and triple-bounce reflection involving bridge structure and water surface. The location of such scattering events is predictable from the given SAR viewing geometry and the bridge structure. On the one hand, such features are useful to extract information about the 3D structure of bridges from InSAR data.

On the other hand, SAR phenomena such as layover and occlusion burden the analysis. Hence, in order to achieve higher detection probability a multi-aspect analysis is advantageous. In this paper, a methodology for bridge detection in large multi-aspect InSAR data sets is proposed and demonstrated. Based on detection results information about the bridge structure is derived in subsequent fine analysis.

The paper is organized as follows. In Section 2 the typical appearance of bridges in high-resolution InSAR data is discussed. Geometric constraints for the mapping of bridge structures into the SAR imagery are given. The methodology for bridge detection and geometry extraction is presented in Section 3. This structural image analysis approach is demonstrated for two InSAR data sets of the same urban scene, which have been taken from orthogonal viewing directions. The data have spatial resolution better than 40 cm in range and even finer in azimuth direction.

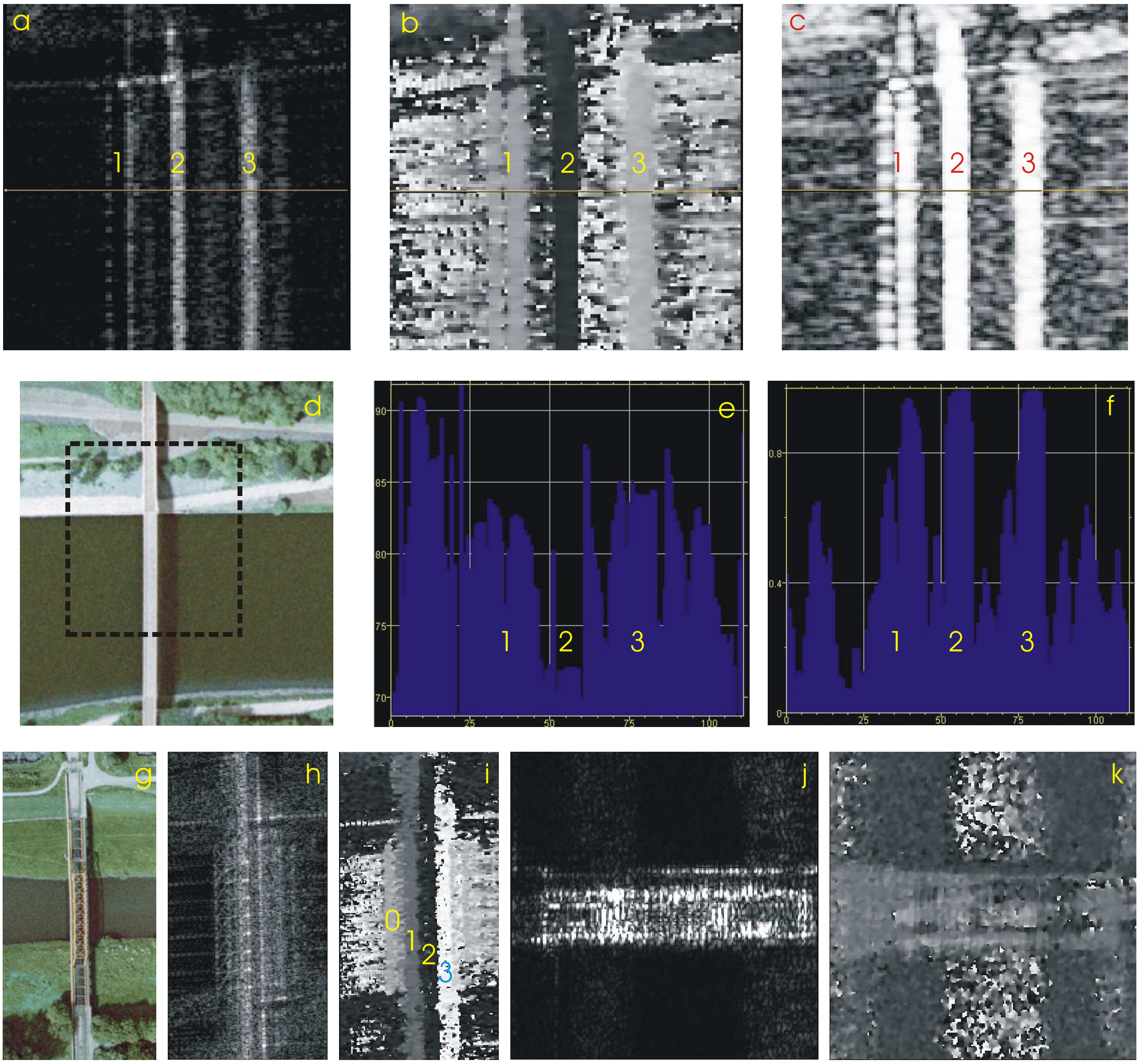


Figure 1 InSAR data sets with spatial resolution approximately 38 cm in range and 18 cm in azimuth, off nadir angle 43 degree, range is always from left to right: a-c) magnitude, elevation (DEM), and coherence images of an interferogram showing part of a narrow bridge over a river in slant range geometry; d) aerial image of same bridge (dashed area corresponds to SAR data); e,f) elevation and coherence values along the horizontal profile in b-c); g-h) railway bridge: g) aerial image; h,i) amplitude and elevation data of same aspect as a-c), number 0 corresponds to layover from bride superstructure; j,k) amplitude and elevation of railway bridge in orthogonal view.

2. APPEARANCE OF BRIDGES IN HIGH-RESOLUTION INSAR DATA

Bridges over water illuminated orthogonal to their orientation (i.e. along the river direction) may cause multiple images in SAR data. Usually three parallel structures are observed at increasing range locations: first direct backscatter from the bridge (more precise: layover of bridge and water signal), followed by double-bounce reflection between bridge and water or vice versa, and finally triple reflection (water, lower parts of the bridge and water again). Sometimes additionally superstructure elements and piles are also visible. This was already shown in the literature for SAR satellite amplitude imagery [Raney, 1998]. In SAR data of coarser resolution usually the structures show up as salient bright lines in sharp contrast to surrounding water surface. From the ground range distance Δg_s of first to second or second to third stripe and off nadir angle θ the bridge height h can be estimated [Raney, 1983; Robalo & Lichtenegger, 1999] according to:

$$h = \Delta g_s / \tan(\theta). \qquad (1)$$

In SAR data of finer spatial sampling however the structures are not line-like anymore but appear as stripes of considerable width, which has to be considered for geometric analysis. Additionally, in the case of InSAR data further information is available in form of interferometric elevation and coherence.

In the following, the appearance of bridges in high-resolution multi-aspect InSAR data is discussed and geometric constraints are given. The test site is located in the city area of Dorsten, Germany. It contains several water canals. The single-pass X-band SAR data shown in Figure 1 were acquired by the AeS sensor of Intermap Technologies [Schwaebisch & Moreira, 1999]. Spatial data resolution is 38.5 cm in range and 18 cm in azimuth. After co-registration and further pre-processing, interferograms have been calculated from the given SAR imagery. From the interferogram the coherence is obtained and, after phase-unwrapping, the InSAR elevation (DEM). The image chips depicted in Figure 1a-c cover part of a narrow bridge spanning water, illumination direction is from left to right, off nadir angle θ is approximately 43 degree. The mentioned triple stripe structure shows up again in the magnitude, elevation, and coherence images. In the magnitude image (Figure 1a) however the bridge's layover signal (structure *1*) is only partly visible, probably due to scattering away from sensor at railing elements and mirror reflection on the smooth paving. The former hypothesis is supported from the dashed structure of the related coherence (Figure 1c). Both in elevation and coherence images (Figure 1b,c) the layover stripe structure is better visible compared to the magnitude data. The entire width of the layover stripe Δs was estimated manually from the InSAR images to be approximately 5m in slant geometry that project to distance Δg of 7.3m in ground range according to:

$$\Delta g = \Delta s / \sin(\theta), \qquad (2)$$

with the difference Δs between first s_{lf} and last layover point s_{ll} (Figure 2a). This is well above the ground truth bridge width of 4m taken from the aerial image shown in Figure 1d. But considering the sketch in Figure 2a**,** this is not surprising, since layover on the water body is caused both by vertical and horizontal bridge structure elements. If additionally the identification of the backscatter of point s_{lc} located at the lower bridge corner is possible, at least the vertical bridge dimension h_b can be derived from the data by:

$$h_b = (s_{lc} - s_{lf}) / \cos(\theta). \qquad (3)$$

Assuming s_{lc} to coincide with the border between dashed and solid layover parts, vertical height h_b is estimated to 2.6m, which seems to be plausible for such small bridge.

Reason for the second bright stripe (structure *2*) is double-bounce reflection s_{db} occurring at the corner reflector that is spanned from smooth vertical bridge facets facing the sensor and the water surface. The signal propagation according to this effect is sketched in Figure 2b. By theory all these double-bounce signal contributions s_{db} should integrate into the range cell that coincides with the direct reflection or single-bounce backscatter path length s_{sb} from the nadir projection of the vertical bridge elements on the water surface:

$$s_{sb} = s_{db}. \qquad (4)$$

But, due to additional different scattering events (e.g. at small bridge structures) and non-perfect smoothness of bridge and water surface, the double-bounce signal is usually spread out around the slant range value s_{sb} of a direct signal from the bridge footprint [Robalo & Lichtenegger, 1999]. The width of this stripe seems therefore to be hardly predictable without very detailed 3D information of bridge geometry and material properties.

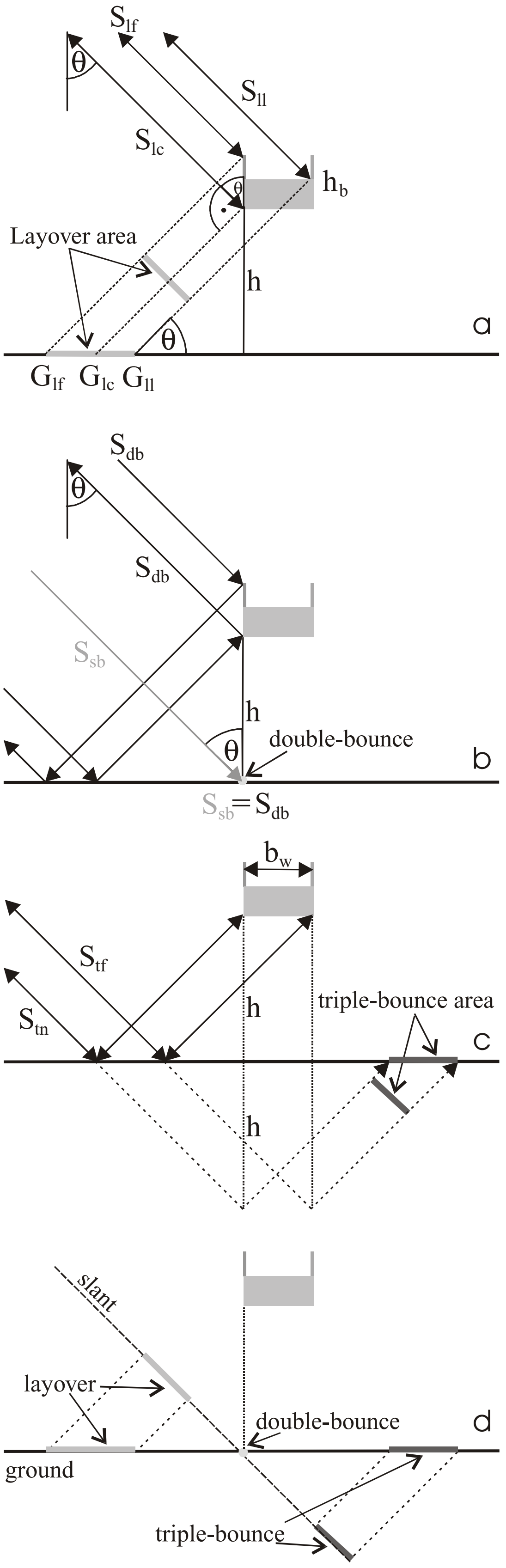


Figure 2 SAR Phenomena arising from viewing geometry at a bridge (grey) over water: a) layover, b) corner reflector double-bounce, c) triple-bounce, d) location of these effects in slant and ground geometry.

Analogous to Equation 3, the bridge height h can be estimated from the difference s_{db} - s_{lc}:

$$h = (s_{db} - s_{lc}) / \cos(\theta). \quad (5)$$

Such estimate of height h of course can also be derived from the InSAR elevation data. At first glance, the most straightforward way for this task seems to use the interferometric elevation value difference between bridge and surrounding water. However, elevation values coinciding with water surface were not useful for this purpose, because almost specular signal reflection led to negative SNR of about -3dB, resulting in elevation data approximately evenly distributed over the possible unambiguous elevation span of 20m. But, it turned out that the mean elevation value over the entire second stripe was a very good estimate of the water surface height. The elevation data standard deviation over this stripe was also very low. This observation is supported by the related mean coherence magnitude $|\gamma|$ of 0.98 (Figure 1f). According to

$$SNR = \frac{|\gamma|}{1-|\gamma|}, \quad (6)$$

this coherence value translates to SNR of 49 or approximately 17dB. The bridge height h over water was estimated using elevation values taken from the layover stripe (structure *1*). The difference of both estimates giving the distance between bridge deck and water was in this case 11m compared to 10,8m from ground truth (LIDAR DEM).

Very interesting is also the third bridge image (structure *3*) resulting from triple-bounce reflection between water, the lower bridge part, and water again. Figure 2c illustrates this effect: because of the longer path length the signal is mapped to a position behind the true bridge location in range direction. Geometrically the signal seems to stem from a virtual bridge replica produced by mirroring the real bridge at the water surface. Assuming the absence of substructures below the bridge's core, the width of the bridge b_w can be estimated exploiting this type of signal. Analogous to Equation 2, bridge width b_w is given by the difference of near and far triple-bounce stripe borders, here called s_{tn} and s_{tf} respectively:

$$b_w = (s_{tf} - s_{tn}) / \sin(\theta). \quad (7)$$

This estimate yields 4.5m for the width of this bridge b_w that is close to 4m according to the aerial image.

Interestingly, the interferometric elevation values of such stripes were in some cases far too high in the final DEM product, possibly due to erroneous treatment during phase unwrapping processing, because initial phase values indicate to elevation well below water level. This behaviour is object of further studies.

In Figure 2d the mentioned effects are summarized and their location in slant and ground range SAR images is given. From the sketch and the image examples discussed above it becomes clear that in high-resolution InSAR data the stripes are not evenly spaced in range and show different spatial extension. Hence, simple height determination according to Equation 1, which yields good results for data of coarser resolution, seems not to be appropriate for data of finer spatial sampling.

In the same InSAR dataset a railway bridge spanning water is almost perfectly orientated in same direction. This object features typical construction structures often observed at railway bridges, such as superstructures made of connected metal bars crossing in vertical and horizontal directions. The horizontal structures are directly visible in the aerial image (Figure 1g) taken from nadir view and the vertical ones can at least be guessed from sun shadow on water and shore. Similarly, but caused by totally different mapping processes, the superstructure pattern appears in the InSAR data. Especially from the magnitude image (Figure 1h, illumination direction from left again) the human observer may extract details of these structures. Despite layover, which causes signal mixture of vertical and horizontal metal bridge structures, the horizontal X-structures are at least partly visible in the layover signal, not only in the water region but also in the grassland area. However, the interpretation of the amplitude data is not straightforward, mainly because of dominant scattering events (e.g. at metal bars of the superstructure) superimposing adjacent areas even far apart the origin of such strong backscatter. The mentioned superstructures cause in the InSAR elevation data a fourth salient signal stripe, appearing as bright or elevated zone on the left in Figure 1i (structure *0*, structures *1,2,3* same as in Figure 1a-c). Then the sequence follows as described for the other bridge: layover of bridge's trackway, double-bounce area, and triple-bounce area. The very same bridge is shown again in Figure 1j,k, this time illuminated from orthogonal aspect along trackway direction. Even though the single-pass acquisition of both tracks was only 10 minutes apart, the bridge mapping is now totally different only due to the altered aspect angle. Superstructures orientated perpendicular to the illumination direction lead to strong scattering, revealing some insight in the bridge's geometry.

3. BRIDGE DETECTION APPROACH

Bridge detection and feature extraction are carried out in two subsequent modules. Knowledge about the typical size of bridges is coded a priori in a bridge model that can be further specified according to information of features of the scene of interest. In the general case, including bridges over roads and bridges crossing valleys of the landscape, both steps base on structural image analysis. For example, bridge hypotheses are detected using crossing stripe-like objects (one for the bridge and the other for the bridged obstacle), which fulfill certain model requirements and have been built hierarchically from edge or line primitives [Soergel et al., 2006]. But, for the special case of bridges over water such approach is not appropriate for two reasons. Firstly, despite man-made river regulation, rivers exhibit often rather curved structure together with sometimes remarkable and abrupt change of contours, e.g. due to natural riverbank variation. An example is given in Figure 3a on the right depicting an amplitude image of a canal of different shape on both bridge sides. Furthermore, river shape may change significantly because of seasonal effects influencing the water level. Secondly, the sharp bridge contrast to the water background allows a simplified detection strategy.

The strategy applied here is described using the images shown in Figure 3. First step is segmentation of dark amplitude image areas based on a threshold that can be estimated from histogram analysis. Of course besides the desired water area all other dark areas (e.g. caused from smooth surfaces such as asphalt) are extracted in this manner and the bright bridge structures are still missing (Figure 3b). By a sequence of morphological erosion and dilatation steps undesired small objects are removed and bridge gaps are closed. The remaining image region (white in Figure 3c) is now the expectation area for bridges over rivers. The morphological operation sequence has to be parameterized according to a given river and bridge model (i.e. search for narrow or broad rivers or bridge, respectively). Here, in general it is assumed that the bridge is narrower compared to the river or canal. The expectation area can further be scaled down by a logical “exclusive or” operation with the initial threshold result (i.e. Figure 3b ⊕ Figure 3c) and subsequent morphological noise reduction.

The aim of the algorithm described so far is screening of large data sets for potential bridge locations. For reasons of robustness and computational load the described procedure is carried out in sub-sampled data. Subsequent analysis is based on high-resolution data.

The next step consists of the detection of possible bridge structures in the InSAR data restricted to the segmented expectation area. As discussed in the previous section, depending on viewing aspect and river orientation the very same bridge might appear as single or multiple stripe structure in the imagery. In the remainder of this paper the more interesting latter case is focused on. Compared to magnitude and elevation data the coherence image is most suitable for detection of the triple stripe structure (Figure 1a-c). For the detection of individual stripes the Steger operator [Steger, 1998] is used. This operator requires the stripe width as parameter. The admissible range of this parameter is adjusted according to the given bridge model. Furthermore, the expected bridge orientation can at least be roughly estimated from the main direction of the detected water body in proximity of the bridge (assuming preferred orthogonal crossing of bridges over water). The extracted stripe structures for the two bridge examples are shown in Figure 3d. The analysis up to now was carried out separately for every viewing direction of the given InSAR data. The individual results can be fused to improve evidence. This topic shall be investigated thoroughly in future work.

4. FINE ANALYSIS

The fine analysis is based on the geometric constraints discussed in Section 2. The first step is to decide whether neighboring stripes belong to the same bridge or not. This is sometimes hardly possible, if bridges are located close to each other. In the test area this problem does not arise. According to the given bridge model, plausible minimum and maximum values of the separation in range of the stripes can be roughly estimated from Equation 1.

As discussed before, the bridge’s height over water can be estimated in different ways from SAR and InSAR data. Results are shown in Table 1 and compared to LIDAR data as ground truth. B1 refers to the right column and B2 to the left column in Figure 3. The SAR and LIDAR data have not been collected at the same time, but since the scene contains canals and not rivers the water height is expected to be kept quite constant from authorities in order to ensure smooth shipping traffic. Except for the LIDAR reference data all estimates are rounded to integer values.

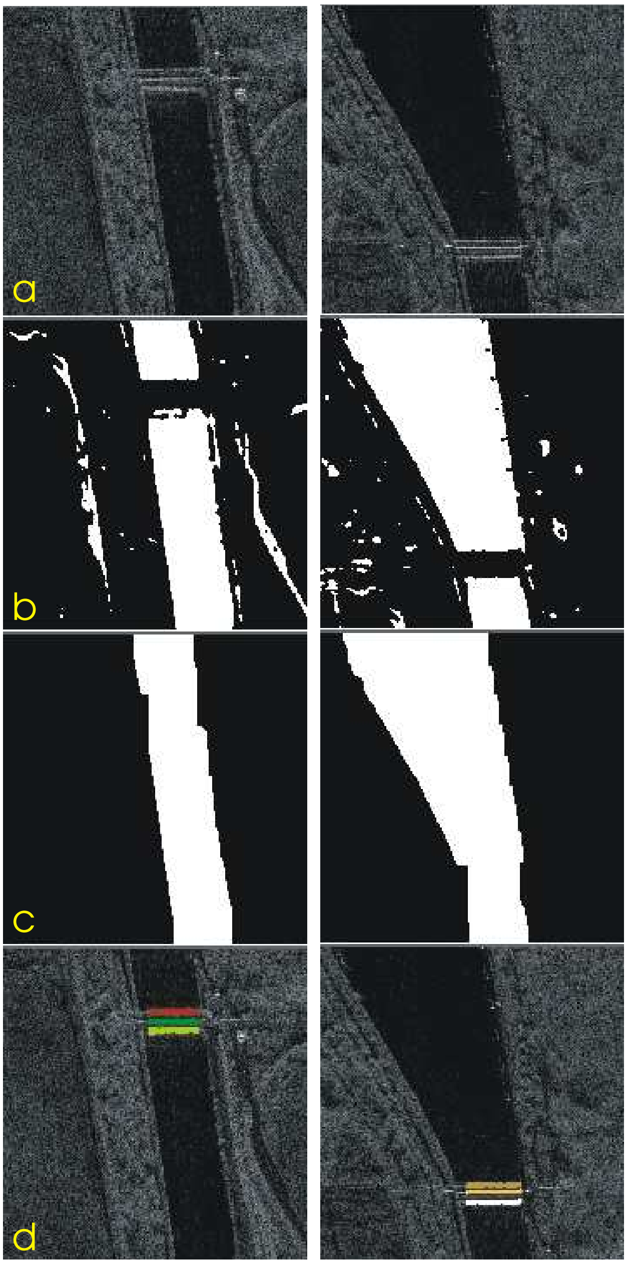


Figure 3 Detection of two parallel bridges in InSAR data, range direction top-down. From top: a) amplitude images, b) Result of threshold operation (dark regions shown in white), c) result after morphological operations, d) detected typical triple stripe image structure of both bridges.

	Ground truth	Height from amplitude		Height from elevation	
		1 2	2 3	manual	automatic
B1	10.8	9	11	11	11
B2	11	11	9	11	11

Table 1 Results of bridge height over water estimation.

First, bridge height extraction from amplitude data is discussed. With respect to Equation 1, the problem arises to choose the correct range locations for the estimate. Here, manually the middle stripe range positions have been used and two estimates were carried out for stripes one to two and two to three. The accuracy of results varies with up to 2m error.

Another possibility to determine the bridge height is the elevation data. At the beginning of the investigations it was assumed, that the average elevation of the water would match its real height, despite the lower SNR compared to other objects. However, this was not the case, probably due to absence of wind leading to almost mirror-like water surface resulting in dominant noise influence and an elevation mean only slightly below bridge level. Therefore, the height was estimated from the difference of the layover and the double-bounce signals. This was done twice: manually and from the automatically detected stripe structures. The results are close to the reference values. In the case of the exploitation of the automatically detected stripes only elevation pixels coinciding with coherence values larger than 0.9 were used to calculate the mean over the extracted stripe. Furthermore, the coherence is used as weight in the averaging process [Soergel et al., 2003] to increase the accuracy of this estimate. Without consideration of the coherence results would be severely degraded.

Of course from such few examples as presented here no statistical sound overall assessment of the methodology is possible. However, the achieved accuracy encourages further investigation in this direction in future studies.

5. CONCLUSION AND FUTURE WORK

Modern SAR sensors achieve such high spatial resolution that even rather small bridges are mapped with considerable level of detail. Therefore, more comprehensive analysis of such objects is now possible. Interferometric processing even reveals many additional object features supporting bridge extraction. However, the constraints arising from the sometimes multiple appearance of bridge structures in the data have to be considered carefully. Height estimate based on InSAR elevation data seems to be more robust compared to analysis of amplitude SAR data alone. First results of the proposed approach for bridge detection and geometry extraction are promising.

In this paper the focus was on bridges over water. The morphological water segmentation might fail in the case of narrow rivers or creeks. In order to detect such thin water bodies a line based approach will be developed. In further investigations other types of bridges shall also be considered (e.g. spanning roads, railway tracks, or valleys).

At present, the detection is carried out independently in each InSAR data set. In the future the image analysis shall be combined in earlier recognition stages to enhance results by mutual evidence support and the elimination of blunders. Furthermore, context information given for example by a road network extraction [Hedman et al., 2005] will be incorporated to support the analysis.Finally, automatic reconstruction of bridge extensions in terms of length and width will be investigated.

REFERENCES

Bach, H., Appel, F., Fellah, K., de Fraipont, P.: Application of flood monitoring from satellite for insurances, Proc. of IGARSS, 2005, CD, 4p.

Hedman, K. Wessel, B., Soergel, U., Stilla U.: Automatic Road Extraction by Fusion of Multiple SAR Views. In: M. Moeller, E. Wentz (eds). 3rd International Symposium: Remote sensing and data fusion on urban areas, URBAN 2005. International Archives of Photogrammetry and Remote Sensing, Vol. 36, Part 8 W27, 2005, CD, 5 p.

International Charter Space and Major Disasters: list of recent floodings: http://www.disasterscharter.org/new_e.html flooding of river Elbe, Germany and Czechia, April 2006: http://www.disasterscharter.org/disasters/CALLID_116_e.html

Raney, R. K.: "The Canadian SAR Experience", Chapter 13 of Satellite Microwave Remote Sensing, edited by T.D. Allan, Ellis Horwood Ltd., Chichester, 1983, pp 223-234. (see also: http://ccrs.nrcan.gc.ca/radar/ana/confed_e.php).

Raney, R. K.: "Radar Fundamentals: Technical Perspective." In Henderson, Floyd M., and Anthony J. Lewis, ed., Manual of Remote Sens-ing 3rd Edition: Principles and Applications of Imaging Radar, Vol 2. American Society for Photogrammetry and Remote Sensing, 1998, pp 9-130.

Robalo, J. and Lichtenegger, J.: "ERS-SAR Images a bridge", ESA, Earth Oberservation Quarterly, December 1999, pp. 7-10 (see also: http://esapub.esrin.esa.it/eoq/eoq64/bridge.pdf).

Schwaebisch, M. and Moreira, J.: "The High Resolution Airborne Interferometric SAR AeS-1". In: Proceedings of the Fourth International Airborne Remote Sensing Conference and Exhi-bition, Ottawa, Canada, 1999, pp. 540-547.

Soergel, U., Cadario, E., Gross, H., Thiele, A., Thoennessen U.: "Bridge Detection in multi-aspect high-resolution Interferometric SAR Data". Proc. of 6th European conference on synthetic aperture radar, EUSAR, 2006, CD, 4p.

Soergel, U., Thoennessen, U., and Stilla, U.: "Reconstruction of Buildings from Interferometric SAR Data of built-up Areas." Proc. of PIA, International Archives of Photogrammetry and Remote Sensing, Vol. 34, Part 3/W8, 2003, pp. 59-64.

Steger, C.: "An Unbiased Detector of Curvilinear Structures." IEEE Trans. Pattern Analysis Machine Intelligence, Vol. 20, No. 2, 1998, pp. 113-125.

3D TREETOP POSITIONING BY MULTIPLE IMAGE MATCHING OF AERIAL IMAGES IN A 3D SEARCH VOLUME BOUNDED BY LIDAR SURFACE MODELS

I. Korpela

[a] Department of Forest Resource Management, POB 27, 00014 University of Helsinki, Finland – ilkka.korpela@helsinki.fi

Commission III

KEY WORDS: Forest Inventory, Remote Sensing, Single Tree, Mapping, Height Estimation, Allometric Modelling

ABSTRACT:

This paper presents a method for semi-automatic 3D positioning of tree tops that can be used for obtaining tree maps of the photo-visible trees and tree heights. Such spatio-temporal, detailed information is usable for many applications in e.g. forestry and landscape management. The method incorporates the use of passive, high-resolution optical images with co-existent low-resolution airborne lidar data. The latter is used for confining the search space of image matching to agree with the volume of photo-visible trees in the upper canopy and for obtaining an accurate elevation model, which is paramount for reliable tree height estimation. The method is presented here and tested with restricted image and field material.

1. INTRODUCTION

Remote sensing is applied currently in almost all forest data acquisition. Orthoimages and stereopairs of aerial photographs are used for stratifying the forest into stands, satellite images are employed in the assessment of large areas and airborne laser scanning is used for the mapping of topography and canopies. Advances in the sensor technologies and analysis methods continuously widen the potential scenarios of new forest inventory methods that put to use remote sensing (Leckie 1990, Baltsavias 1999, Petrie 2003, Naesset et al. 2004). Single-tree remote sensing (STRS) that is based on the idea of substituting the field measurements and mapping of individual trees with cost-efficient airborne observations is an example of a field made possible by the development. Digital and automatic, image- and/or lidar-based STRS is a topical domain (See references in Culvenor, 2003; Korpela, 2004; Pouliot et al., 2005), although the concept of STRS is not entirely novel (Worley and Landis, 1954; Talts, 1977).

STRS aims at a detailed description of the growing stock that is crucial in most applications of forest inventory. Ideally, it provides the size-distribution of the standing trees per species with the two- or three-dimensional map of trees. Korpela and Tokola (2006) examined the potential of image-based, 2D and 3D STRS. The DBH (stem diameter at 1.3 m height) and volume of individual trees cannot be estimated as accurately with STRS as is it possible in the field. The main reason is the indirect estimation phase with allometric models that results in both random and systematic, tree and stand level errors. The model inaccuracies are coupled with photogrammetric measurement errors in species, tree height and/or crown width. Random errors cancel out effectively, but the aggregate results of STRS at the stand level are liable to systematic offsets. Inclusion of tree heights, i.e. the use of 3D STRS was found to improve the estimation accuracy of both DBH and stem volume considerably in comparison to 2D STRS, in which trees are measured for species and crown dimensions only. In addition, in STRS the growing stock is inherently underestimated since some trees always remain unseen – at least by optical sensors.

In STRS, field calibration is needed for avoiding the systematic errors of the allometric equations. Thus, some field visits seem inevitable if very accurate data is wanted. Because of the inferior accuracy in comparison to field measurements, an applicable STRS system has to provide the measurements and estimates with much lower costs, which calls for automatic procedures. A complete 3D STRS system solves all of the following tasks: (a) tree or crown positioning in 3D, measurements of (b) crown dimensions and (c) tree height, (d) species recognition and (e) allometric estimation of stem size (Figure 1).

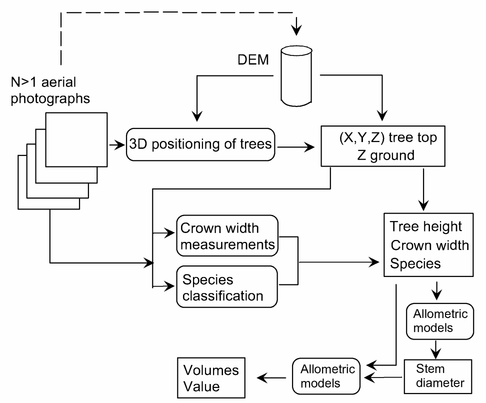


Figure 1. An example of the data, tasks and output of a 3D image-based STRS-system for stand cruising.

1.1 Hypotheses and objectives

This paper addresses the question of using remote sensing for 3D treetop positioning and height estimation and extends the work by Korpela (2000, 2004), in which a semi-automatic method for treetop positioning was introduced. It was based on the use of multiple image-matching of digitized aerial photographs for the purpose of finding treetops inside a predefined 3D search space in the canopy volume (Figure 2). The algorithm applies template matching for processing the aerial images into correlation images, where local maxima correspond to 2D image positions of treetops (cf. Pollock, 1996; Larsen and Rudemo, 1998). The predefined 3D search space is processed into a point mesh. The points in the mesh are back-projected into the correlation images and aggregated for volumetric correlation, which is further processed into 3D maxima that correspond to candidate treetop positions. The algorithm resembles that of Tarp-Johansen (2001), who positioned tree bases of oaks in 3D using multiple leaf-off aerial images. Here, it is further assumed that correct 3D treetop positions will help in solving the other image-based tasks of the measurement of the crown dimensions and the interpretation of species (cf. Figure 1).

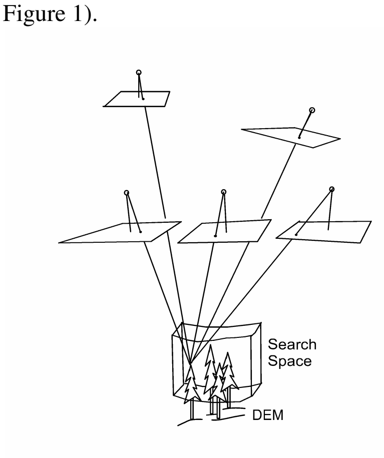


Figure 2. Image matching for 3D tree top positioning (Korpela, 2004). The search is restricted to a predefined volume in the canopy. A DEM/DTM is used for height estimation. The scale of the images (N>1) is not restricted as such, but the full orientation of the images has to be established reliably.

The discernibility of treetops is a major restriction of optical STRS. Only the dominant, co-dominant and intermediate trees are visible with a high likelihood. The probability of discernibility is an exponential function of the relative height of the tree; such probability-of-discernibility curves vary between stands according to the density of the stand (Korpela, 2004). In most cases trees with a relative height of below 50% are not seen at all in the images. The 50% relative height constitutes thus a lower limit for the volume from where to conduct the manual or automatic search of treetops – at least in closed canopies. Respectively, the upper limit is at the maximal height of trees. These two parameters vary spatially and it is necessary to obtain reasonably accurate estimates of them to avoid commission errors by the treetop positioning algorithm (Korpela, 2000; 2004) as the locally restricted depth of the 3D search space is the geometric (epipolar) constraint that is used for the solution of the mathematically ill-posed correspondence problem for tree tops. The results can only be optimal if the search is set to cover the upper canopy volume (Korpela, 2004; p. 35, 65–66).

The estimation of tree height is straightforward once the treetop is positioned in 3D. A DTM gives the elevation of the butt. The error of the height estimate consists thus from possible treetop positioning errors and DTM errors. A DTM is also needed for defining the lower limit of search space at the app. 50% relative height level below which treetops cannot be expected to be measurable. Korpela (2004) suggested that an accurate DTM obtained by means of low-resolution laser scanning could be incorporated in the algorithm for the delineation of the search space and for accurate tree height estimation. Similarly, laser scanning was proposed for the estimation of the local, maximal height of trees by a canopy height model (CHM). These proposals/theses are put to test here with real field, image and lidar data. By combining aerial photographs with lidar this paper exploits the principle of the photo-lidar approach presented by St-Onge et al. (2004). A low sampling rate airborne lidar is used to keep the material costs to a minimum. The proposal in this article is that low-resolution lidar can be combined with multiple image-matching of aerial images for accurate and cost-efficient, semi-automatic tree top positioning and tree height estimation.

2. METHOD FOR SEMI-AUTOMATIC 3D TREETOP POSITIONING USING AERIAL IMAGES AND LIDAR BASED SURFACE MODELS

The method consists of the steps 1–9 given below. Automation of steps 2 and computations in step 5 have been developed most in comparison to the algorithm presented in (Korpela, 2004).

1) Delineation of the area of interest. Here, the tree tops were positioned inside circular plots with a radius ranging from 15 to 20 m. In general, the geometry of the area of interest can vary and a homogenous stand would be a natural choice in practice.

2) Delineation of the 3D search space in the upper canopy. This is done by analyzing the lidar-DTM and the lidar-CHM such that the search space is filled by a 3D point mesh with 0.5 m spacing. The maximal elevation or local dominant height in a given XY point is given by the CHM, which is multiplied by parameter $f_{HDOM} \in [1, 1.3]$ to reduce the inherent underestimation. Parameter *HDepth* ∈ [0, 1] defines the depth of the search space with respect to the local dominant height of trees (*HDepth* = 1) and the terrain elevation (*HDepth* = 0).

3) Selection of a sample tree and the measurement of its 3D treetop position using manual image-matching. The capture of elliptic templates representing the tree in all images (Figure 3).

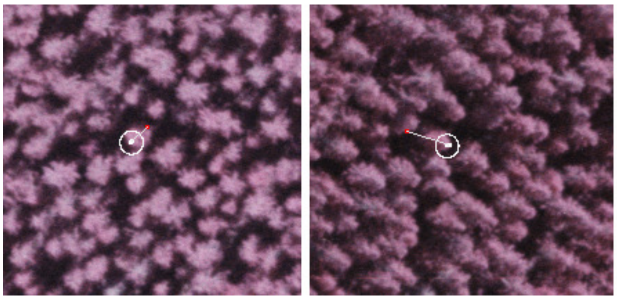


Figure 3. Template-boundaries of a selected and manually positioned sample spruce tree with parameters: *EllipseHeight* = 3.0 m, *EllipseWidth* = 2.6 m and *EllipseShift* = −1.0 m. The shift downwards by *EllipseShift* is seen in the image on the right: the image position of the hot-spot i.e. the tree top and the template centre deviate. The vertical lines connect the measured 3D tree top position and the DTM. This photo-lidar height estimate was 15.53 m and the field measurement was 15.7 m.

Object space parameters *EllipseShift*, *EllipseHeight*, and *EllipseWidth* define the position, size, and shape of the ellipse in the images. *EllipseShift* shifts the center of the template in the Z direction. Using this parameter, the templates are typically moved down to capture more of the crown than the background. *Ellipseheight* defines the major axis of the elliptic template, which in the images is made parallel to the direction of the Z axis (trunks). *EllipseWidth* defines the length of the shorter axis.

The shape is conditioned to circular i.e. the templates are allowed to be elliptic for oblique views only and in the direction of the radial displacement (i.e. Z axis, tree trunk). These 3 parameters take metric values. The actual template images are rectangular copies of the aerial images. Pixels that fall outside the ellipse are masked out. The location of the treetop inside the template, the so called hot-spot, is stored for each template and is accounted for in cross-correlation computations that follow.

4) Template matching. Template matching with normalized cross-correlation is carried out for each image using the template of that aerial image. This procedure maps the aerial images into cross-correlation images ρ (x,y) ∈ [-1,1], in which high values of ρ indicate good match at image location x,y (Figure 4). Ideally ρ (x,y) would consist of very sharp peaks at the correct positions of the treetops.

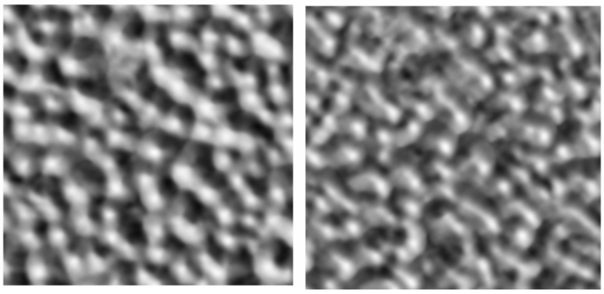


Figure 4. Cross-correlation images computed using the captured templates and aerial images of Figure 4. High correlation is displayed in white.

5) Aggregation of 3D correlation, *ρ3D*. Each point in the search space is back-projected to the cross-correlation images using collinear equations and an affine fiducial mark transformation with pixel accuracy. *ρ3D* is computed for each point in the 3D search space seen as a geometric mean of the images resulting in *ρ3D*∈ [0, 2].

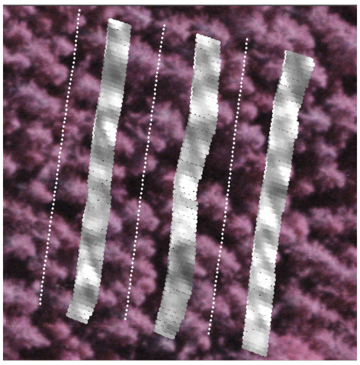


Figure 5. Illustration of the volumetric, discrete *ρ3D* data in the search space with three transects (slices) superimposed in an oblique aerial view. The brightness of the points denotes *ρ3D*. The undulation is due to changes in terrain elevation and local dominant height of trees. The white dots that form lines are the terrain points.

6) Clustering of the *ρ3D* data into 3D treetop candidate positions. The point set is first sorted in the ascending order of *ρ3D*. Clusters are formed from points with *ρ3D* above a limit, *Rlimit*. Points are merged into existing clusters while the sorted list is processed. Merging is controlled by a planimetric distance parameter, *XYthin*. Points closer than the set value are merged into existing clusters and do not form a new cluster. The 3D position of the cluster is the mean of the 3D points that belong to the cluster and *ρ3D* is used in linear weighting of the coordinates. *Rlimit* is a parameter that controls the quality of the clusters. Only the best clusters are accepted as tree top candidates, if *Rlimit* is set to a high value. In such cases, omission errors are few assuming that the search space is set correctly. A low value of *Rlimit* brings about new clusters at the cost of commission errors. The merge-parameter *XYthin* controls the density of the clusters. A value that is too large causes neighbouring trees to be merged. Similarly, if *XYthin* is set too low it can result in several clusters originating from the actual *ρ3D* response of a single tree.

The description of the steps 7–9 below applies to any practical implementation of the algorithm in situations where no ground truth exists. In the experiments of this study step 7 was replaced by a numerical quality assessment, and steps 8 and 9 were not performed.

7) Visual quality assessment of the treetop positioning. The visual evaluation of the matching results is based on visual examination of the candidates that are superimposed either on monoscopic or stereoscopic views. If necessary, the clustering algorithm is re-run by adjusting the parameters *XYthin* and *Rlimit*. Sometimes the procedures have to be repeated from the start by selecting and positioning a new model tree. As all subsequent steps need to be re-computed it is important to have good approximate values for the parameters to avoid unnecessary iteration.

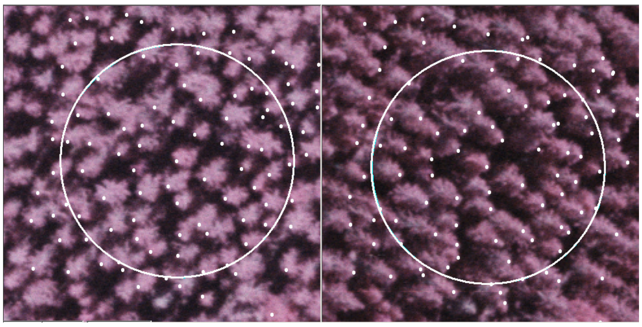


Figure 6. Candidate positions and the borders of circular photo-plot (r=15 m) superimposed in an image pair. The circle is drawn at the elevation of the treetop of the model tree.

8) Manual correction of the semi-automatic matching results. In it, the bad candidates are removed or corrected for position. The unrecognized tree tops are completed manually using stereo interpretation (for operators with a good stereo vision) or using manual image matching with monocular observations and epipolar constraining (Korpela, 2004)

9) Height estimation using the existing DTM.

3. EXPERIMENTS

3.1 Data

The field data in Hyytiälä, southern Finland (61°50'N, 24°20'E) consists of fully mapped and measured stands (Korpela 2004). The field measurement errors for tree positions and the basic tree variables are known through repeated observations. The positions of the field trees have been established with tacheometer and VRS™-GPS observations and field levelling. The image data consisted of digitized aerial photographs, which have been orientated in one large multi temporal (1946-2004) image block (Korpela 2006). Here, leaf-on images from summers of 2002 and 2004 were used in the experiments. These were taken using standard metric cameras with 15 cm and 21 cm lenses and the images have a 14- or 15-micron pixel size. The experiment allowed for testing the following nominal scales: 1:6000, 1:8000, 1:12000, 1:14000, 1:16000 and 1:30000. The images have forward and side overlaps that vary from 60 % to 80 %. Lidar data was from August 2004 with an Optech ALTM2033 sensor from a flying height of 900 m. The pulses had a footprint diameter of 0.3 m and the pulse density was 1.1 m by 1.3 m, on average. The instrument recorded 1 or 2 returns. The full geometry of each pulse was available: time stamp, position and orientation of the lidar, ranges, intensities and positions of the 1 or 2 returns. A raster DTM was processed from the lidar returns using a simple gradient-based method and a RMSE of 0.30 m was obtained in a test set of 10947 tacheometer points representing terrain of wooded areas. A raster CHM was constructed from lidar maxima in 5 m by 5 m cells.

3.2 Performance of tree top positioning

A treetop was considered to be correctly found (hit) if a candidate was inside a 2.4-meter wide and a 6-meter high test-cylinder. The dimensions of the test-cylinder affect the performance measures. The field errors in tree positioning using tacheometer, in height measurements, errors made in updating heights to the time of the photography, possible tree slant and sway as well as the stand density of the test sites were considered. The test-cylinders can have overlap in dense forests and excessive candidates in a test-cylinder or in intersecting cylinders were considered as commission errors and trees without a candidate were considered as errors of omission. A buffer around circular test plots (Figure 7) was used as trees can be hit by a candidate from the buffer and vice versa.

Hit-rate was the ratio between the number of hits and the total number of trees. An accuracy index was computed based on the numbers of omission (o) and commission (c) errors and the number of trees (n) (cf. Pouliot et al. 2005): AI = $[(n - o - c) / n] \times 100$. The 3D-positioning accuracy was evaluated with the RMSE that were computed separately for the XY and Z although the positioning is entirely 3D. The RMSEs include the imprecision of the ground truth and therefore overestimate the true inaccuracy. The positioning error-vector [ΔX, ΔY, ΔZ] was defined as **field–candidate**; thus a positive ΔZ indicates underestimation. Mean differences of ΔX, ΔY and ΔZ measure systematic offsets. To evaluate the averaging effect of tree heights, a regression line was fitted in the ΔZ×tree height distribution and the slope coefficient (trend) and its standard error were computed. The set of field trees was confined to those that were discernible to the operator. This tree set represents the potential trees to be found. In some stands such a criterion can leave out 50% or more of the trees; however, the proportion of the total volume in the non-discernible trees is normally small, from 0 % in managed stands to 12 % in natural forests (Korpela, 2004).

3.3 Tests in a spruce stand

Treetop positioning was tried out using image sets in scales 1:8000-1:16000 (Table 1) in one managed spruce stand. Images in the scale of 1:6000 were left out because of the computational burden of template matching and scale 1:30000 was omitted because individual treetops were not well measurable in that scale anymore.

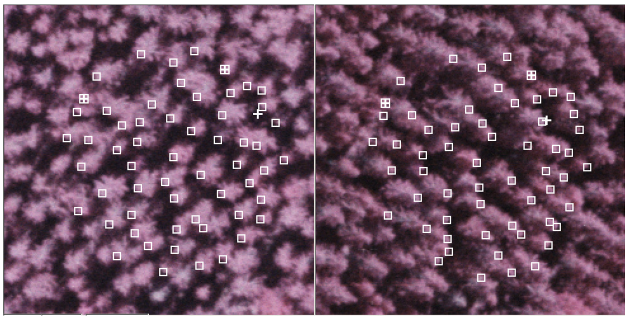


Figure 7. Results of treetop positioning for a circular test plot. Unfilled squares depict the candidate positions for correct hits (56), squares with a cross depict missed treetop positions (o = 2), and the crosses depict the commission errors (c = 1). The AI was [(58-2-1)/58]×100 = 94.8%. The hit-rate in total stem volume was 97.1%, RMSE of ΔXY was 0.55 m, RMSE of ΔZ was 0.67 m with a slope coefficient of 0.055 m per m of tree height. The errors in the DTM elevations had an RMS of 0.27 m.

One model tree was used in all trials, and the parameters defining the shape and position of the elliptic templates were kept fixed. The exact 3D position of the treetop was measured separately for each set of images using manual, monoscopic multi-image matching. It varied in Z because of the temporal mismatch of the May 2002 and June 2004 images and because of small orientation and observation errors. The search space was kept fixed with parameters f_{HDOM} and *HDepth*. Tree heights from May 2002 were simply added +0.7 m, which corresponded to the average height growth of three summers. Parameters *Rlimit* and *XYthin* were tuned for obtaining optimal results in the AI-measure.

Number of images, scale, overlaps (%), focal length (cm)	AI-%	c	Mean ΔZ, m	RMS ΔZ, m	RMS ΔXY, m
2 1:8000 60/60 21	61.1	13	−0.04	1.29	0.70
2 1:8000 60/60 21	77.9	9	−0.39	1.24	0.73
4 1:8000 60/60 21	85.3	7	+0.06	0.76	0.68
4 1:8000 60/60 21	88.4	5	−0.21	0.99	0.67
6 1:8000 60/60 21	85.3	2	−0.08	0.72	0.61
2 1:12000 70/60 15	82.1	10	+0.22	0.80	0.60
3 1:12000 70/60 15	91.6	5	+0.13	0.70	0.56
4 1:12000 70/60 15	88.4	4	+0.31	0.85	0.57
3 1:14000 80/60 21	87.4	4	−0.09	0.83	0.68
4 1:14000 80/60 21	87.4	7	−0.20	0.93	0.65
6 1:14000 80/60 21	94.7	3	−0.15	0.87	0.60
7 1:14000 80/60 21	93.7	2	−0.12	0.94	0.62
2 1:16000 60/60 15	85.3	5	−0.25	0.87	0.66
3 1:16000 60/60 15	88.4	3	+0.15	0.93	0.62
4 1:16000 60/60 15	74.7	4	+0.42	0.98	0.58

Table 1. Results of treetop positioning using different number of images in different scales. Plot S6 with 95 photo-visible trees in a circular plot with radius of 20 m. f_{HDOM} = 1.15, *HDepth* = 0.65.

Increasing the number of images usually improved the performance in the AI measure; however there were images in which the crown of the model tree was not seen against a clear background, which resulted in a poor cross-correlation image that deteriorated treetop positioning. The imaging geometry affects treetop positioning; best results were obtained with an image set that consisted of six images taken with normal-angle cameras at the scale of 1:14000. These images had even large overlaps and the elevation of the sun was higher (45^o) during the photography. These factors affect occlusion and shading in aerial views that can impede image matching. It seems that the optimal scale for the type of spruce trees in plot S6 (heights from 12 to 22 m) is somewhere between 1:10000 and 1:15000. The images in 1:8000 had details that did not help in treetop positioning, but may be needed for example in species recognition with texture measures.

Parameter f_{HDOM} corrects the local dominant height given by the CHM and thus defines the upper limit of search space. Similarly, the parameter *HDepth* defines the lower height and depth of the search space. Treetop positioning was tried at different values of these parameters. The optimal values for f_{HDOM} were from 1.1 to 1.3, when *HDepth* was kept at 0.65. All performance measures showed best performance in this range. Here, the CHM was calculated using a 5 m grid, which may be too coarse in sparse stands. Similarly, the density of the lidar data will most likely affect the quality of the CHM, which needs to be considered in setting the value for f_{HDOM}.

f_{HDOM}	AI-%	c	Mean ΔZ, m	RMS ΔZ, m	RMS ΔXY, m	Trend ΔZ×h, m/m
0.95	46.3	21	+1.03	1.35	0.71	0.33
1.00	78.9	11	+0.88	1.28	0.69	0.31
1.05	88.4	7	+0.48	0.92	0.70	0.24
1.10	94.7	4	+0.21	0.74	0.69	0.18
1.15	93.7	5	+0.06	0.74	0.70	0.16
1.20	89.5	7	–0.09	0.80	0.70	0.17
1.25	90.5	5	–0.18	0.86	0.71	0.16
1.30	88.4	5	–0.21	0.85	0.71	0.14
1.35	85.3	6	–0.32	0.91	0.71	0.13
1.40	73.7	15	–0.39	0.93	0.72	0.13

Table 2. Performance of the 3D tree top positioning algorithm for different values of the parameter f_{HDOM}. Plot S6 with 95 trees in a circular plot with radius of 20 m. *Rlimit* = 1.41, *XYthin* = 1.5 and *HDepth* = 0.65. Four images in scale 1:12000.

Parameter *HDepth* gives the lower height of the search space, and this parameter should be adjusted according to stand density since in dense stands only the tallest trees remain photo-visible The dominant height of plot S6 was 20.6 m and the shortest discernible tree had a plot-level relative height of 0.53. However, the neighboring trees of this 10.6-m high tree had heights from 15 to 18 m, which means that the local relative height of this tree is approximately 0.6. Best results in AI-% were obtained with *HDepth* at 0.65. Commission errors ("short ghost trees") start to appear, if the search space is started from a too low height. If the search space is not deep enough, the heights of the short trees are overestimated and the averaging effect increases. These effects are seen in Table 3.

HDepth	AI-%	c	Mean ΔZ, m	RMS ΔZ, m	RMS ΔXY, m	Trend ΔZ×h, m/m
0.45	64.2	31	+0.28	0.84	0.72	0.11
0.50	75.8	22	+0.19	0.77	0.71	0.11
0.55	88.4	10	+0.12	0.72	0.70	0.12
0.60	90.5	8	+0.06	0.71	0.70	0.13
0.65	93.7	5	+0.06	0.74	0.70	0.16
0.70	92.6	3	+0.01	0.75	0.70	0.19
0.75	89.5	3	–0.06	0.75	0.71	0.21
0.80	89.5	2	–0.21	0.77	0.72	0.22
0.85	82.1	3	–0.42	0.82	0.73	0.22
0.90	69.5	2	–0.70	0.94	0.74	0.20

Table3. Performance of the 3D tree top positioning algorithm for different values of the parameter *HDepth*. Plot S6 with 95 trees in a circular plot with a radius of 20 m. *Rlimit* = 1.41, *XYthin* = 1.5 m, f_{HDOM} = 1.15. Four images in scale 1:12000.

4. DISCUSSION

Semi-automatic 3D tree top positioning of individual trees using image-matching is an alternative or complement to lidar-based techniques in which trees are found by processing very high-resolution lidar data with from 5 to 30 points per m^2. The method presented here combines optical images and low-cost lidar with emphasis on the use of images. The lidar-based surface models that approximate the canopy elevation and give the terrain relief accurately are a necessity for accurate height estimation, since the ground is seldom seen in images taken under leaf-on conditions. If the image-matching strategy here is compared with common techniques of stereo matching for surface modelling, it can be said that the lidar CHM and DTM provided a short-cut and gave a good approximation for the possible space of solutions, which normally are obtained by hierarchical image matching techniques and the coarse-to-fine strategy (Schenk, 1999). The results of the experiments gave support to the thesis that low-resolution lidar data can be used for delineating and bounding the search space in the canopy semi-automatically by adjusting the parameters that define the relative underestimation of the lidar-CHM (f_{HDOM}) and the lowest relative height of the trees that expected to be visible in the aerial views (*HDepth*).

The implementation described here is not very robust against the variation in the size of tree crowns and the results presented here were good mainly because the test stand represented a rather regular forest. In stands with a large species mixture and variation in crown sizes, the results have been found inferior. It may be possible to incorporate the use several sample trees (or synthetic images of crowns; see Larsen, 1997) in image matching to improve the possibilities to detect and position trees of varying size. Similarly, it would be desirable, if the feature detector, template matching in this case, would yield not only the 2D image positions of tree tops but also symbolic information similar to what is utilized by an operator when the task is performed manually (species, crown size). It would then be possible to rule out automatically some of the unpreventable commission errors.

A semi-automatic approach seems to be the only solution to 3D tree top positioning using aerial views because of the nature of the problem. Occlusion and shading are inherently present in aerial views and trees vary in size, shape and radiometric properties. In the development of the methods presented here, the strategy has been to provide a system for measuring as many tree tops as possible automatically with a high positioning accuracy and a low commission error rate. After manual amendment the 3D tree tops provide tree heights and 2D image positions that can be used as seed points for the remaining tasks of species identification and measurement of crown dimensions, which can possibly be solved in the 2D image domain.

5. REFERENCES

Baltsavias, E.P. 1999. A comparison between photogrammetry and laser scanning. *ISPRS, JPRS* 54(2-3), pp. 83-94.

Culvenor, D. S., 2003. Extracting individual tree information: a survey of techniques forhigh spatial resolution imagery. In: *Remote Sensing of Forest Environments: Concepts and Case Studies.* Edited by M. A. Wulder and S. E. Franklin (Boston: KluwerAcademic), pp. 255-277

Korpela, I. 2000. 3-d matching of tree tops using digitized panchromatic aerial photos. University of Helsinki. Department of Forest Resource Management. Licentiate Thesis. 109 p.

Korpela, I. 2004. Individual tree measurements by means of digital aerial photogrammetry. *Silva Fennica Monographs*, 3, pp. 1-93.

Korpela, I. and Tokola T. 2006. Potential of aerial image-based monoscopic and multiview single-tree forest inventory - a simulation approach. *Forest Science*, 52(3), pp. 136-147

Korpela I. 2006. Geometrically accurate time series of archived aerial images and airborne lidar data in a forest environment. *Silva Fennica,* 40(1), pp. 109-126.

Larsen, M. 1997. Crown modeling to find tree top positions in aerial photographs. In: *Proceedings of the third International Airborne Remote Sensing Conference and Exhibition*, Copenhagen, Denmark. ERIM international. Vol 2, pp. 428-435.

Larsen, M. and Rudemo, M. 1998. Optimizing templates for finding trees in aerial photographs. *Pattern Recognition Letters*, 19(12), pp. 1153-1162.

Leckie, D.G. 1990. Advances in remote sensing technologies for forest surveys and management. *Canadian Journal of Forest Research,* 20(4), pp. 464-483.

Naesset, E., Gobakken, T., Holmgren, J., Hyyppä, H., Hyyppä, J., Maltamo, M., Nilsson, M., Olsson, H., Persson, Å. and Söderman, U. 2004. Laser scanning of forest resources: The Nordic experience. *Scandinavian Journal of Forest Research*, 19(6), pp. 482-499.

Pouliot, D.A., King, D.J., and Pitt, D.G., 2005. Development and evaluation of an automated tree detection-delineation algorithm for monitoring regenerating coniferous forests. *Canadian Journal of Forest Research,* 35(10), pp. 2332-2345.

Petrie, G. 2003. Airborne digital frame cameras. The technology is really improving. *GEOInformatics*, 7(6), October/November 2003, pp.18-27.

Pollock, R.J. 1996. The automatic recognition of individual trees in aerial images of forests based on a synthetic tree crown model. PhD-thesis in computer science. The University of British Columbia. 158 p.

Schenk, T. (1999). *Digital photogrammetry*. Vol I. TerraScience, Laurelville, Ohio, USA.

St-Onge, B., Jumelet, J., Cobello, M. and Véga, C.2004 Measuring individual tree height using acombination of stereophotogrammetry and lidar. *Canadian Journal of Forest Research*, 34(10), pp. 2122–2130.

Talts, J. 1977. Mätning i storskaliga flygbilder för beståndsdatainsamling. Summary: Photogrammetric measurements for stand cruising. Royal College of Forestry. Department of Forest mensuration and management. Research notes NR 6 – 1977. 102 p. (In Swedish).

Tarp-Johanssen, M.J. 2001. Locating Individual Trees in Even-aged Oak Stands by Digital Image processing of Aerial Photographs. PhD Thesis. Dept. of Mathematics and Physics. KVL. Copenhagen, Denmark. 158 p.

Worley, D.P. and Landis, G.H. 1954. The accuracy of height measurements with parallax instruments on 1:12000 photographs. *Photogrammetric Engineering*, 20(1): 823-829.

Classification of lidar data into water and land points in coastal areas

A. Brzank, C. Heipke

Institute of Photogrammetry and GeoInformation
University of Hanover
(brzank, heipke)@ipi.uni-hannover.de

KEY WORDS: lidar, laser scanning, classification, fuzzy logic, filtering

ABSTRACT:

Over the last years lidar has become one of the major techniques to obtain spatial data in coastal areas. Due to the fact that lidar systems can provide several 3D points per square meter and high height accuracy, lidar data is suitable for several applications in the field of coastmonitoring and coastprotection. Generally, a digital terrain model (DTM) is used as basic spatial information for applications like morphologic change detection and hydrological modelling. In order to generate a DTM in coastal areas from lidar data, a classification process has to be performed to separate the lidar points into water and land points. Only land points, representing the coastsurface, are used to calculate the DTM.

In this paper, we present a new method to classify lidar data in water points and land points. The original points of each flight strip are classified scan line by scan line. Several parameters which are directly related to each point as well as the point distribution within one scan line are used for the classification method. A fuzzy logic concept is applied to determine a membership value for every point belonging to the class water. Then, a threshold method is employed to classify the points of every scan line. Afterwards, classification discrepancies are detected and corrected by comparing height differences between neighboured water and non-water points. In order to achieve a more realistic classification result small isolated point groups of a certain class are removed. To illustrate the ability of the algorithm two examples with different characteristics (lidar scanner system, point density, point distribution etc.) are presented. The results are promising and constitute a proof-of-concept for the suggested method.

1. INTRODUCTION

Lidar has become one of the major techniques to obtain high accurate spatial data in coastal areas. The method delivers, depending on the lidar system and flight parameters, several laser points per square meter with high height accuracy. Large areas can be registered fast (e.g. Brügelmann and Bollweg 2002) and digital terrain models (DTMs) can then be interpolated from the individual 3D points. Generally, a DTM is used as basic spatial information for applications like morphologic change detection and hydrologic modelling. In order to calculate a DTM, a filtering process has to be performed to separate lidar points into terrain points and off terrain points (Sithole and Vosselman, 2004).

Within coastal areas, several regions are covered by water. Typically the lidar beam does not penetrate water. Hence, lidar points measured in water regions describe the water surface but not the DTM lying underneath. In order to obtain a DTM of high accuracy, another process must be included to identify water points and exclude them from the DTM calculation.

Depending on the available data sources different approaches are possible. Two general cases can be distinguished. In the first case simultaneous acquisition of lidar and multispectral data is assumed. In this case, the images can be used to classify water with common classification methods. Lecki et al. (2005) pointed out that high-resolution multispectral imagery and appropriate automatic classification technique offer a viable tool for stream mapping. Within their analysis, especially water was classified accurately. Mundt et al. (2006) demonstrated that the accuracy of classification significantly increases by combining images and height data. However, multicspectral images are not always acquired during lidar data capture. Thus, in the second case, only the lidar data is assumed to be available. Typically, lidar data providers deliver the original 3D points and an intensity value, which corresponds with the strength of the backscattered beam echo. Up to now, only a few approaches to use the intensity of lidar data for classification were published. Katzenbeisser and Kurz (2004) emphasized the fact that classification methods used for remote sensing images need to be adapted to intensity data. They pointed out that the intensity has only a useful information value within open areas where only one echo was detected. Hence, other criteria's have to be considered in order to filter water points from lidar data.

In this paper, we first summarize important physical characteristics of lidar data and previous approaches, which were carried out to separate water points in lidar data. Then, a new method is presented to classify lidar data into water and land points. Starting from original irregularly distributed lidar points, several parameters are derived and rated using a fuzzy logic concept. Several steps are taken after classification in order to detect discrepancies and enhance the classification result.

To illustrate the ability of the algorithm, two examples with different characteristics (lidar scanner system, point density, point distribution etc) are presented. Finally, this paper concludes with a summary and an outlook on further development issues.

2. STATUS OF RESEARCH

2.1 Physical characteristics of lidar data within coastal areas

In order to develop a suitable algorithm which is capable of classifying the lidar data (raw 3D-lidar points and intensity values) the physical characteristics of common lidar systems as well as the reflection of water and land areas have to be considered. Generally, lidar systems operate in the near infrared range. Wolfe and Zissis (1989) describe the absorption of infrared radiation depending on the illuminated surface material and the wavelength. They pointed out that the absorption for water is significantly higher than the absorption for soil. This leads to the fact that the intensity of water points is regularly lower than the intensity of land points.

Additionally, as a result of the Rayleigh Criteria, calm water surfaces behave like a mirror. Thus, specular reflexion occurs. Depending on the spatial orientation of the aircraft, the emitted laser pulse and the water surface with respect to each other, in general only a small part of the emitted radiation returns to the detector. Often, a distance measurement can not accomplished successfully because the received radiation energy is not distinguishable from background noise. This leads to the fact that the point density of lidar data within water areas is often significantly lower than within land areas.

2.2 Filtering off terrain points and filtering water points respectively

The filtering of off terrain points from lidar data is a common and necessary step in order to derive a DTM. Many different approaches (i.e. Sithole and Vosselman, 2005 or Tóvári and Pfeifer, 2005) were published and provide accurate results (Sithole and Vosselman, 2004). Neglecting differences of the approaches it can be stated that high points (or segments respectively) in the vicinity of lower points are generally labelled as off terrain points.

In order to calculate an accurate DTM in coastal areas a filtering of water points is performed. Analogous to off terrain points water points do not belong to the surface and have to be removed from the data set. Water points have a lower height than the surrounding land points. Theoretically, an inverse strategy of filtering off terrain points is able to classify likely water points. However, the overall correctness of a classification using such an inverse filtering strategy is not satisfying due to the fact that common filter techniques use only geometrical relationships of neighboured lidar points or segments respectively. Hence, local minima like tidal trenches are detected, but they may be dry and thus the detected points actually belong to the DTM. Furthermore, completely filled tidal trenches or swales can not be detected because the water level height is nearly equal to the surrounding flat coastal area.

2.3 Previous approaches to extract water areas from lidar data

Brockmann and Mandlburger (2001) developed a technique to extract the boundary between land and water of rivers, and applied it to data from the German river “Oder”. Based on lidar data, the planimetric location of the river centre line as well as bathymetric measurements of the riverbed, the boundary was obtained within a two-stage approach. First, the height level of the water area was derived by averaging the lidar points in the vicinity of the river centre line. Afterwards, the DTM of all lidar points (including also points of the water surface) was calculated. Then, the 0 m contour line of the difference model of the lidar DTM and the water height level was derived. This contour line is called the preliminary borderline. Within step two, the bathymetric points of the preliminary water area are combined with all lidar points outside the preliminary water area. Then, a DTM representing the riverbeds instead of waterlevel was calculated. Afterwards, the final borderline was obtained by intersecting the DTM including the riverbeds and the height level of water area.

Brzank and Lohmann (2004) (see also Brzank et al., 2005) proposed another algorithm which separates water regions from non water regions based on a DSM calculated from lidar data. The main idea was to detect reliable water regions and expand them with the use of height and intensity. For that purpose local height minima were extracted from the DSM, which represent the potential seed zones of the searched water areas. This was followed by region growing procedure using height and intensity data of the grid points.

2.4 Evaluation of previous approaches

In order to classify water points within lidar data, only height information is not sufficient. At least one additional data source is necessary. Brockmann and Mandlburger (2001) used the 2D position of the river as prior information. Hence they knew approximately where water occurs. Assuming that a water area has lower height than the surrounding land, the border can be detected. Next to the 2D position and the lidar data, also bathymetric measurements are prior information of this method. Thus, this algorithm needs additional information which is not always available in our application, taking into account that form and position of tidal creeks are changing fast.

Brzank and Lohmann (2004) tried to use the intensity as additional criteria to classify water. The algorithm provides accurate results if the intensity of water points differs significantly from land points. However, due to the fact that the intensity is generally very noisy and strongly influenced by the lidar scanner type and used wavelength, type and water ratio of the illuminated area, the classification accuracy can be unpredictable. Thus, at least one criterion has to be implemented in a new algorithm. Furthermore, this method does not work with the original lidar data but uses grid data. This is a crucial disadvantage because lidar data is obtained strip wise and generally, parts of several flight strips are combined in order to calculate a certain grid. Depending on the flight planning, a time shift occurs between neighboured flight strips. Taking into account that the water level in coastal area varies with time due to the tide, several water levels of the same water area may thus occur in a grid.

2.5 Requirements of the algorithm to classify water points from lidar data

Based on the physical characteristics of lidar data and the evaluation of the previous approaches, the following requirements for a successful algorithm are defined:

1. The algorithm uses the original lidar data.
2. No additional data sources such as images or vector GIS data are permissible.
3. The point density is used as additional criterion.
4. The classification is done for every flight strip.

3. CLASSIFICATION OF WATER POINTS USING 1D-LIDAR PROFILES

The new classification method is based on the analysis of 1D-lidar profiles of the original raw data in combination with fuzzy logic. Each lidar strip is classified separately followed by a check across the scan lines. At first the lidar points of a strip are grouped into single scan lines. Then a membership value of class water (see equation 1) is calculated for each point of every scan line. The membership value depends on the parameters height, slope, intensity, segment length, point distribution and missed points (see section 3.1), the membership function and weight for every used parameter. Afterwards, the classification is done using a hysteresis-threshold-method. Finally, in order to detect and remove discrepancies, several steps are applied. They use the classification results of neighbouring scan lines to overcome the limitation of 1D profile classification. All classification steps are described in more detail in the following.

$$\mu(x) = \sum_{i=1}^{m} \delta_i \mu_i(x) / \sum_{i=1}^{m} \delta_i \qquad (1)$$

δ_i:	weight parameter i
$\mu(x)$	entire membership of class water for point x
$\mu_i(x)$	membership value water point x depending on parameter i

3.1 Employed parameters and membership function

For classification several parameters are used. The parameters are:

Height: The higher a lidar point is situated the higher is the assumption that this point is not a water point. Thus with increasing height the membership value for class water decreases.

Slope: The more the slope within the profile direction increases the more the assumption holds that the following point is not a water point. Thus, with increasing slope the membership value for class water decreases.

Intensity: As pointed out earlier a low intensity value is an indication for a water point. Thus, with decreasing intensity the membership value for class water increases.

Missed points: If holes occur from one profile point to next within the scan line, discrete point(s) are not measured. The appearance of holes is an indication for a water region. The bigger a hole between two neighboured points the higher is the assumption for the occurrence of water. In order to deal with points which are close to the border line between land and water the number of missed points is checked in both direction for every profile point. Only the membership value related to the smaller number of missed points is used further.

Segment length: Based on the determination of the missed points the number of contiguous points within a profile can be derived. Thus, every profile point is a member of a certain segment with a certain segment length. With increasing segment length the indication increases that the segment points are land points.

Point density: For every point the number of previous and following profile points within a certain distance s can be determined. The higher number is divided by the distance s. Thus, with increasing point density the membership value for class water decreases.

It has to be pointed out, that the parameters missed points, segment length and point density are related to the fact that generally the number of points within the water area is smaller than within the land area. The usage of all parameters is possible, but existing correlation should be considered.

In order to calculate the membership value for a certain parameter a membership function and thresholds are needed. Basically, every function which increases strictly monotonic (or decreases strictly monotonic) can be used. In our algorithm, a straight line is applied. The two resulting thresholds limit the application range of the membership function. Outside the application range the membership value is set to 0 or 1 depending on the parameter. Figure 1 illustrates the calculation of the membership value of the parameter height for a scan line. After selecting the two thresholds the membership value can be calculated.

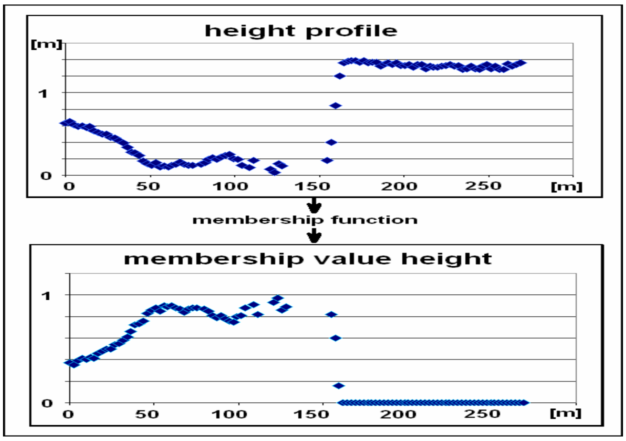


Figure 1: Deriving the membership value of the parameter height for a 1D-profile

After the calculation of the membership value for every scan line point using equation 1 the classification is done with a hysteresis-threshold-method. A low and a high threshold have to be defined. The classification of the actual point depends on the classification result of the previous point. If the previous point was classified as land the membership value of the actual point has to be higher than the high threshold to be classified as water. If the previous point was classified as water the membership value of the actual point has to be only higher than the low threshold to be classified as water.

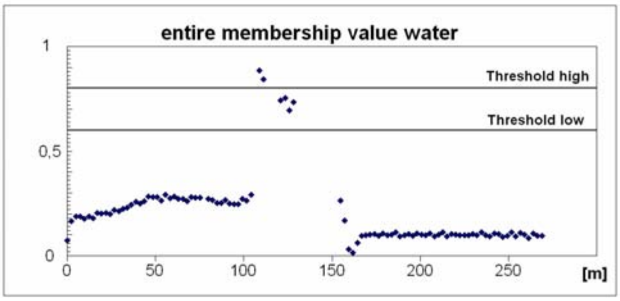


Figure 2: Classification of a 1D-profile with hysteresis-threshold-method

Figure 2 illustrates the classification process. The classification starts from the beginning (left side) of the profile. All of the first points have a membership value below the low threshold. They are classified as land points. Then, two points next to each other have a membership value above the high threshold, thus they are classified as water. The next four points of the profile are in between both thresholds. These four points are also classified as water points, because the previous point was classified as water and the membership value is higher than the low threshold. Thus, six points of the illustrated profile are classified as water points. It has to be mentioned that this classification depends on the direction, in which the profile is processed. If the classification starts from the other side (the end of the profile) the result may be not the same. In case of Figure 2 only the two points above the high threshold are considered to be water points if the classification starts from the right side.

3.2 Elimination of classification discrepancies and classification enhancement

Typically, classification techniques do not output error-free classification results. In order to obtain a suitable result classification discrepancies have to be removed. To detect and remove these discrepancies several steps are performed. They are all based on the fact that a water point next to a land point must have a lower height. At first, every individual profile is checked. If a water point next to a land point is found, the mean height of all water points within a certain distance is compared with the height of the land point. This mean height of several neighboured water points is used to suppress the influence of occurring waves. If the mean water height is equal or higher than the land height, a classification discrepancy occurs. Then, the average of the mean membership value of the water points and the membership value of the land point is calculated and compared to the average of the two thresholds used for the hysteresis-threshold-classification (see Figure 2). All points are labelled as water/land if the average membership value is higher/lower than the average of both thresholds.

Due to the fact that the algorithm is limited to 1D-profiles, the classification does not take neighboured points of the previous and next scan line into account. Therefore, in order to improve the result the second dimension is considered in the next step. Every scan line is compared to its left and right neighbour scan line. It is assumed that every correctly classified segment continues in the previous as well as the next scan line. A simple example may illustrate this assumption. Assuming a tidal trench which is filled with water is present in the lidar data. Several scan lines cross the water area. Assuming further that all scan lines are classified correctly, the classified water segment of the tidal trench for a certain line can be found next to this segment in the previous and the next scan line.

To check all classified segments of every scan line we use the following approach. First, every scan line is split into classified segments of the same class (see figure 3). Then, a rectangle with a width of three scan lines is generated, which is limited by the first and last point of the considered segment. Afterwards, all points from the previous and next scan line which are inside the rectangle are extracted. If no point of the extracted previous scan line and also no point of the next scan line have the same classification as the considered segment, the classification is defined to be wrong. Then, the classification of the considered segment is changed. Figure 3 shows an example of the check. The segment in the centre of the figure is detected as an isolated segment and the classification is changed while the segment in the lower right remains.

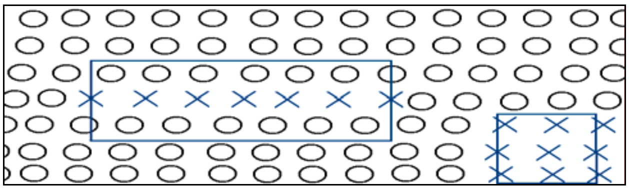


Figure 3: Check for isolated classified segments, crosses represent classified water points – circles represent classified land points, scan lines run from left to right

Subsequently, another classification check is performed. Again we use the assumption that if the height of a water point is equal or higher than a neighboured land point a classification discrepancy occurs. At first a certain number of neighboured scan lines is selected (e.g. 10). Then, a cross section is created for every point of each scan line perpendicular to the azimuth of the scan line. For every scan line the point with the smallest distance to the cross section is determined. The point becomes a member of the cross section if the distance is smaller than a predefined distance. Then, every cross section is checked analogous to the control of every individual scan line (see above).

After performing these checks the number of classification errors decreases. However, small classified segments may remain. Thus, the classification results may appear to be noisy. In order to enhance the classification further, small classified segments (of one scan line as well as perpendicular using several scan line) which are surrounded by classified segments of the other class are detected and removed. Finally, an almost consistent and smooth classification result can be obtained.

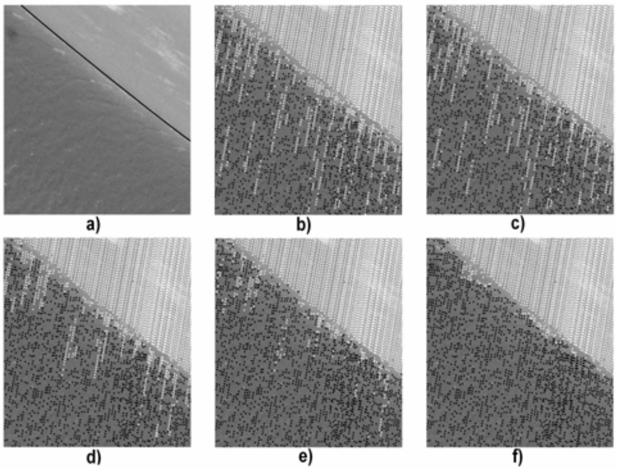


Figure 4: Elimination of classification discrepancies and enhancement, bright points represent land, dark points represent water. a) Orthophoto with digitized water-land-border, b) Classification result without further checks for discrepancies, c) Discrepancies within every scan line removed, d) Segments removed, which only occur in one scan line, e) discrepancies removed within perpendicular cross section, f) Small isolated segments removed

Figure 4 illustrates the process of removing discrepancies and enhancing the classification. Figure 4 a) shows a small part of the coast line of the East Frisian Island Langeoog. The added black line represents the border between water and beach. Figure 4 b) shows the classification result without checking for discrepancies. Bright points are classified as land. Dark points are classified as water. Within the water area some points are classified incorrectly due to the fact that they are part of long segments, which leads to a low water membership value. Furthermore, waves are present. Points on waves are higher, thus they have a low water membership value. The following images show the stepwise process of enhancement and removing discrepancies: 4 c) – discrepancies within every scan line removed, 4 d) segments removed, which only occur in one scan line, 4 e) – discrepancies removed within perpendicular cross section, 4 f) – small isolated segments removed. Finally, a smooth classification result without isolated points is obtained.

3.3 Automated Determination of used parameters in training areas

It is obvious that the selection of the used parameters, the membership function, the weights as well as the thresholds have a crucial impact on the classification result. Depending on the data (lidar scanner type, weather conditions etc.) only parameters which differ between land and water should be used. Because the user has to make these selections, he has to know the data rather well. In order to assist the user with his choice, at least one training area for water and for land is selected interactively. In our approach, the mean value of every parameter within the training area is determined. Based on these values, the user can better decide, which parameters are suitable for the classification.

3.4 Classic Fuzzy classification concepts vs. suggested approach

The classification algorithm uses fuzzy logic. Based on the fundamentals introduced by Zadeh (1965) also classification algorithms containing fuzzy concepts were developed and widely used (Traeger, 1993). Although these fuzzy classification concepts deliver suitable results we adapt the classic concept to overcome some difficulties.

In classic fuzzy classification concepts fuzzy sets (for example: low, medium, high) for every used parameter are defined. Based on membership functions exact values for certain parameters can be transformed into membership values for all defined fuzzy sets. Then, a rule base is defined which decribes how to combine all possible combinations of fuzzy sets of all used parameters. Finally, a defuzzification process is performed in order to allocate the result to a certain class. In our method we do not define fuzzy sets for the used parameters (for example: low height, medium height, high height) but transform sharp values of every used parameter in a membership value for the output class water by using two thresholds as well as the membership function. Thus, we do not have to define a rule base, which is a rather complex task. Assuming that we define three fuzzy sets for every used parameter (6) a total of $3^6 = 729$ rules have to be defined. Furthermore, the membership function of every fuzzy set has to be defined, too. According to the data, the membership functions have to be changed either in an automated process or by a human operator. Furthermore, practical tests with various lidar data pointed out, that the benefit of a parameter also depends on the used lidar scanner system. Thus, the rule base has to be designed taking the used lidar system into account. In our approach, it is easier to classify water areas, because the needed parameter values (thresholds and weights) can be derived by using training areas.

4. EXAMPLES

To show the capability of this approach two different examples are presented. The first example is taken from the lidar campaign "Langeoog 2005". The East Frisian Island "Langeoog" was flown by the German company Milan using the LMS Q560 system of the company Riegl. The example contains a large see water area, mainly dry coast region and some water puddles. The second example contains a certain part of a flight strip of the campaign "Friedrichskoog 2005" which is situated at the coast of the North Sea next to the estuary of the river Elbe. The flight was carried out by the German company Toposys using their own lidar system Falcon II.

Within the first example 361.280 points were classified (see Figure 5). The classification was mainly based on the fact that the point density of the lidar points within water was significantly lower than over land. Additionally, the height also had a major impact on the classification result.

The second example (see figure 6) contains 998.029 lidar points. Due to the fact that the point density did not differ significantly between water and land, the classification was based on the parameters height, slope and intensity.

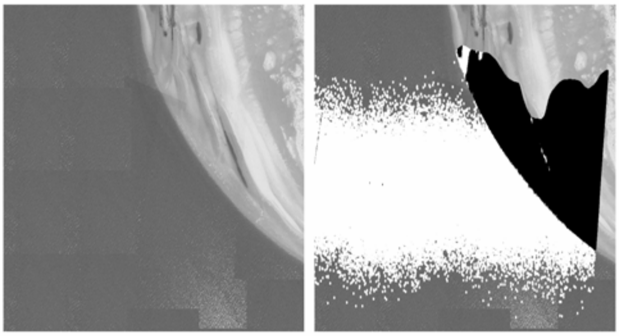


Figure 5: Classification of a part of a flight strip of the campaign "Langeoog 2005" – left: lidar DTM, right: classified water points = white dots, classified land points = black dots

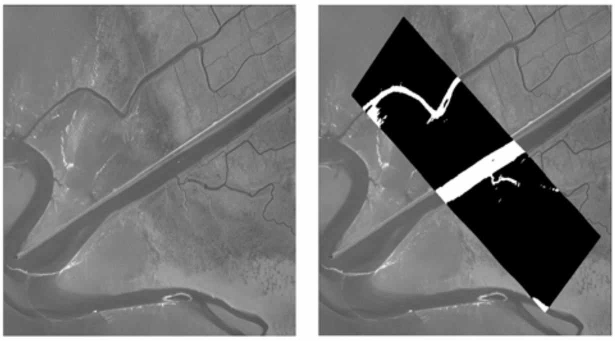


Figure 6: Classification of a part of a flight strip of the campaign "Friedrichskoog 2005" – left: lidar DTM, right: classified water points = white dots, classified land points = black dots

The used parameters, thresholds and weights are listed in Table 1. The thresholds of the parameters were obtained from training areas (see section 3.3) while the weights and the hysteresis-threshold-values were defined manually based on experience with the data set.

Table 1: Classification parameter of Example "Langeoog 2005" and "Friedrichskoog 2005"

	Langeoog 2005			Friedrichskoog 2005		
	Threshold		Weight	Threshold		Weight
	Water	Land		Water	Land	
Height [m]	-0.8	-0.4	2	1.4	2	3
Slope [°]	-10	10	1	-10	10	1
Intensity	---	---	0	22	40	1
Missed points	4	0	2	---	---	0
Segment length	2	10	2	---	---	0
Point density [point/m]	0.722	1.5	5	---	---	0
	low	high		low	high	
Water Threshold	(35%)	(50%)		(40%)	(50%)	

Table 2: Classification result of Example “Langeoog 2005” and “Friedrichskoog 2005”

	Langeoog 2005		Friedrichskoog 2005	
Number classified points	361.280		998.029	
Classified water points	121.399		86.991	
Classified land points	239.881		911.038	
	Water	Land	Water	Land
Classified water points	119.253	2.146	79.803	19.695
Classified land points	990	238.891	7188	891.343
Correctness [%]	99,2	99,1	91.7	97.8

To check the reached correctness the simultaneously acquired image data was merged into an orthophoto mosaic. Based on this mosaic the water and land area was digitized and intersected with the classified points. The results of the check are listed in Table 2. It can be seen that for “Langeoog 2005”, the rate of correctly classified points within the land as well as the water is higher than 99%. The main border line between the sea and coast was nearly completely extracted. Only in the upper centre part of the flight strip the classification is not very accurate due to the fact that this part contains wet sand only slightly higher than the sea water level. The point density within the wet sand is significantly lower than in the neighboured dry sand area, thus the classification provides high water membership values for this part.

Also for “Friedrichskoog 2005”, the results were very promising. 91.7% of the classified water points and 97.8% of the classified land points are correct. Analogous to the first example the algorithm has problems to classify wet land areas. Their intensity values are generally low and their height is only slightly higher than the neighboured water area.

5. CONCLUSION AND OUTLOOK

An approach to separate lidar points into the classes water and land based on 1D profile analysis of the raw lidar data has been introduced. The classification is based on the original lidar data and classifies for every flight strip. The algorithm uses several parameters which are derived from the lidar data. The classification is based on the fuzzy logic concept. Two different examples are shown to illustrate the capability of this algorithm. They point out that the classification algorithm is able to deliver accurate results for different lidar scanner types. However the classification lacks in accuracy if wet land area of low height occur.

In order to increase the automation rate it will be part of the future work to determine meaningful weights of the used parameter as well as the two final water thresholds from training areas.

ACKNOWLEDGMENTS

This research has been financed by the Federal Ministry of Education and Research (BMBF) under project no. 03KIS050. We acknowledge the support of our project partners: Department of Rural Area Husum (ALR), Federal Waterways Directorate (WSD) and the Lower Saxony Water Management, Coastal Defence and Nature Conservation Agency Division Norden-Norderney (NLWKN).

REFERENCES

Brockmann H., Mandlburger G. (2001). Aufbau eines Digitalen Geländemodells vom Wasserlauf der Grenzoder. In *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformationen*, Band 10, 2001, pp. 199 – 208.

Brügelmann, R. and Bollweg, A.E. (2004). Laser altimetry for river management. In *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXV, B2, Istanbul, Turkey, pp. 234 – 239.

Brzank, A., Göpfert, J., Lohmann, P. (2005). Aspects of Lidar Processing in Coastal Areas. In *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXVI PartI/W3, Hanover, Germany, 6 pages, CD.

Brzank, A., Lohmann, P. (2004). Steigerung der Genauigkeit von Digitalen Geländemodellen im Küstenbereich aus Laserscannermessungen, *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformationen*, Band 13, 2004, pp. 203 – 210.

Katzenbeisser, R. and Kurz, S. (2004). Airborne Laser-Scanning, ein Vergleich mit terrestrischer Vermessung und Photogrammetrie. In *Photogrammetrie Fernerkundung Geoinformation*, Heft 8 (2004), pp. 179-187.

Leckie, D., Cloney, E., Jay, C. and Paradine, D. (2005). Automated Mapping of Stream Features with High-Resolution Multispectral Imagery: An Example of the Capabilities. In *Photogrammetric Engineering & Remote Sensing*, Vol. 71, No. 2, February 2005, pp. 145 – 155.

Mundt, J. T., Streutker, D. R., Glenn, N. F.(2006): Mapping Sagebrush Distribution Using Fusion of Hyperspectral and Lidar Classifications. In *Photogrammetric Engineering & Remote Sensing*, Vol. 72, No.1, January 2006, pp. 47 – 54.

Sithole, G., Vosselman, G. (2004). Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds. In *ISPRS Journal of Photogrammetry and Remote Sensing*, 59 (1-2), pp. 85 – 101.

Sithole, G., Vosselman, G. (2005). Filtering of airborne laser scanner data based on segmented point clouds. In *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXVI, 3/W19, Enschede, Netherlands, pp. 66 – 71.

D. Tóvári, D., Pfeifer N. (2005). Segmentation based robust interpolation – a new approach to laser data filtering. In *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXVI, 3/W19, Enschede, Netherlands, pp. 79 – 84.

Wolfe, W. and Zissis, G.J. 1989. The infrared handbook. The Infrared Information Analysis Center. Enviromental Research Institut of Michigan, Detroit, 1700 pages.

AUTOMATIC VEHICLE TRACKING IN LOW FRAME RATE AERIAL IMAGE SEQUENCES

D. Lenhart, S. Hinz
Remote Sensing Technology, Technische Universitaet Muenchen, 80290 Muenchen, Germany
{Dominik.Lenhart|Stefan.Hinz}@bv.tu-muenchen.de

KEY WORDS: Vehicle Tracking, Traffic Monitoring, Traffic Parameters

ABSTRACT:

Traffic monitoring requires mobile and flexible systems that are able to extract densely sampled spatial and temporal traffic data in large areas in near-real time. Video-based systems mounted on aerial platforms meet these requirements, however, at the expense of a limited field of view. To overcome this limitation of video cameras, we developed a concept for automatic derivation of traffic flow data which is designed for commercial medium format cameras with a resolution of 25-40 cm and a rather low frame rate of only 1-3 Hz. Since resolution and frame rate are the most limiting factors, the focus of the first implementations and evaluations lies on the approach for automatic tracking of vehicles in image sequences of such type in near real-time. The tracking procedure relies on two basic components: a simple motion model to predict possible locations of previously detected vehicles in the succeeding images and an adaptive shape-based matching algorithm in order to match, i.e. recognize, the detected vehicles in the other images. To incorporate internal evaluations and consistency checks on which the decision of a correct track can be based, the matching is done over image triplets. The evaluation of the results shows the applicability and the potentials of this approach.

1. INTRODUCTION

1.1 Traffic Monitoring

Traffic monitoring is a very important task in today's traffic control and flow management. The acquisition of traffic data in almost real-time is essential to swiftly react to current situations. Stationary data collectors such as induction loops and video cameras mounted on bridges or traffic lights are matured methods. However, they only deliver local data and are not able to observe the global traffic situation. Space borne sensors do cover very large areas. Because of their relatively short acquisition time and their long revisit period, such systems contribute to the periodic collection of statistical traffic data to validate and improve certain traffic models. However, often, monitoring on demand is necessary. Especially for major public events, mobile and flexible systems are desired, which are able to gather data about traffic density, average velocity, and traffic flow, in particular, origin-destination flow. Systems based medium or large format cameras mounted on airborne platforms meet the demands of flexibility and mobility. While they have the capability of covering large areas, they can deliver both temporally and spatially densely sampled data. Yet, in contrast to video cameras, approaches relying on these types of cameras have to cope with a much lower frame rate.

A more extensive overview on the potential of airborne vehicle monitoring systems is given in (Stilla et al., 2004), while the use of aerial image sequences to derive traffic dynamics is studied in (Toth et al., 2003). There, it is also shown that the knowledge about traffic income and outgo directions allows a more precise and effective handling of traffic flow management.

1.2 Related Work

In the last decades, a variety of approaches for automatic tracking and velocity calculation have been developed. Starting with the pioneering work of Nagel and co-workers based on optical flow (Dreschler and Nagel 1982; Haag and Nagel, 1999), the usage of stationary cameras for traffic applications has been thoroughly studied. Further examples for this category of approaches are (Dubuisson-Jolly et al., 1996; Tan et al., 1998, Rajgopalan et al., 1999; Meffert et al., 2005). Some of the ideas incorporated in these approaches have influenced our work. Though, a straigtforward adoption is hardly possible since these approaches exploit oblique views on vehicles as well as a higher frame rate – both, however, at the expense of a limited field-of-view. Another group of approaches uses images taken by a photogrammetric camera with a high resolution of 5-15cm on ground (e.g., (Hinz, 2004)). Also, these approaches are hardly applicable since the vehicle's substructures which are necessary for matching a wire-frame model are no more dominant in images of lower resolution.

In (Ernst et al., 2005), a matured monitoring system for real time traffic data acquisition is presented. Here, a camera system consisting of an infrared and an optical sensor is mounted on slowly moving air vehicles like an airship or a helicopter, but also tests with aircrafts have been conducted. Traffic parameter estimation is based on vehicle tracking in consecutive image frames collected with a frame rate of 5 Hz and more. While the results are promising, a major limitation of this system is the narrow field of view (the width of one single road) due to the low flying altitude that is necessary to obtain a reasonable resolution on ground.

Considering the data characteristics, the most related approaches are (Reinartz et al. 2005) and (Lachaise, 2005). Like us, they use aerial image sequences taken with a frame rate of 1-3 Hz and having a resolution of 25-40cm. Vehicle detection is done by analyzing difference images of two consecutive frames. This method is quite robust to detect moving objects and to quickly find possible locations for car tracking. Yet, with this approach, it is not possible to detect cars that are not moving, which often also happens for active vehicles if they are stuck in a traffic jam or waiting at a traffic light or stop sign. Furthermore, tracking of detected vehicles includes an interactive component at the current state of implementation.

The boundary conditions of our work are primarily defined by the use of medium format cameras of moderate cost. They allow a large coverage and still yield a resolution of roughly

25cm. However, due to the high amount of data for each image, the frame rate must be kept rather low, i.e. 1 up to a maximum of 3 Hz. In the following, we will outline a concept to automatically detect and track vehicles which is designed to deal with these constraints. The main contribution presented here relates to the tracking procedure rather than the detection of the vehicles. We focused on this point first since the low frame rate is the most influencing factor of the overall concept, and the benefits and limitations of this module should be clearly analyzed. In addition, also some first results of automatic detection will be given.

2. OVERALL CONCEPT

The underlying goal of the concept outlined in the following is the fulfillment of near real time requirements for vehicle tracking and derivation of traffic parameters from image sequences. The general work flow is depicted in Fig. 1.

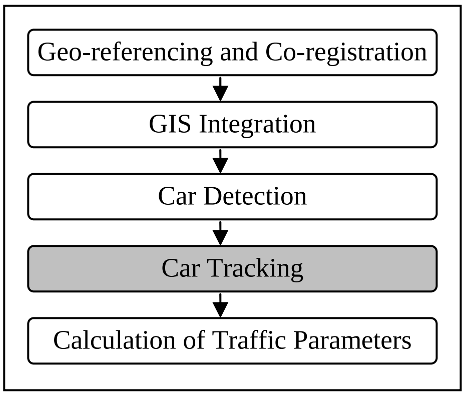

Geo-referencing and Co-registration
GIS Integration
Car Detection
Car Tracking
Calculation of Traffic Parameters

Figure 1. Work flow of online vehicle tracking

The images are co-registered and approximately geo-referenced after acquisition. This process is commonly supported by simultaneously recorded navigation data of an INS-/GPS-System. GIS road data, e.g. stemming from NAVTEQ or ATKIS data bases, are mapped onto the geo-referenced images and approximate regions of interest (RoI) are delineated(so-called road sections). Thus, the search area for the following automatic vehicle detection can be significantly reduced. For further processing, it is helpful to extract the road as well as their lanes in addition, since geo-referencing might not be accurate enough and GIS data rarely includes the position of individual lanes. An example for the automatic determination of lane sections using a slightly modified version of the road extraction system of Hinz & Baumgartner (2003) is shown in Fig 2. This example is generated by a stand-alone module and not yet incorporated into the automatic processing chain.

A car detection algorithm is supposed to deliver positions and, optionally, additional attributes such as boundary and direction constrained to the lanes within the RoI. Tests with matching wire frame models of cars showed only limited success due to the moderate ground resolution of 25-40cm. More promising results were obtained by a differential geometric blob detection algorithm similar to (Hinz, 2005), which has to be trimmed for colored blobs yet. Results of blob detection are shown in Fig. 3.

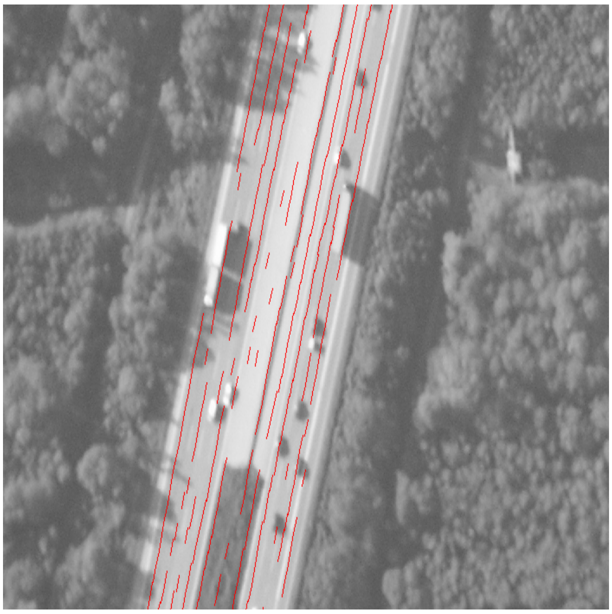


Figure 2. Intermediate result of lane extraction

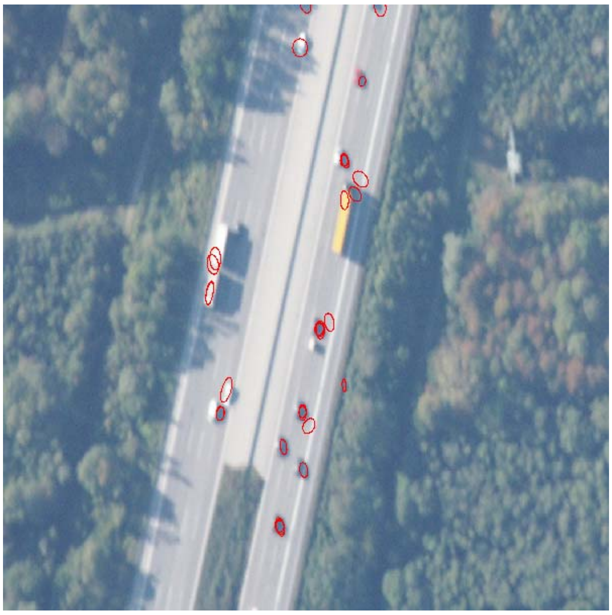


Figure 3. Results of a blob detection

After their detection in the first image, the cars are tracked by matching them within the next two images. To this end, an adaptive shape-based matching algorithm is employed including internal evaluation and consistency checks (see details in Sect. 3). From the results of car tracking, various traffic parameters are calculated. These are most importantly vehicle speed, vehicle density per road segment, as well as traffic flow, i.e. the product of traffic density and average speed, eventually yielding the number of cars passing a point in a certain time interval.

In our tests of vehicle tracking, the first three parts are simulated, thereby accounting for potential impreciseness and uncertainty of the data. Their implementation is due to future work: *i*) The co-registration between image pairs is done by an affine 2D-transformation using least-squares optimization. This approximation seems reasonable, since our focus is on roads, which are generally almost planar objects. *ii*) GIS data have been simulated by digitizing road lines for each carriage way of a road. These lines will be referred to as "road polygons" in the sequel. They consist of "polygon points" *P*, while two of these enclose a "polygon segment" *L*. For each segment, the length as well as the orientation angle *ang(L)* are determined. *iii*) Cars are selected manually by digitizing the approximate center of the car including the shadow region since the shadow is an important indicator in detection and tracking a vehicle. However, since this step will be replaced by an automatic procedure in the near future (see Fig. 3), we will call them "detected vehicles" or "detected cars" in the following.

3. VEHICLE TRACKING

Before outlining algorithmic details of the tracking procedure in Sect. 3.2., we will first sketch the underlying vehicle motion model.

3.1 Vehicle Motion Model

The frame rate of the image sequences dictates the change of locations of a car, i.e. the possible maneuvers a car has undergone in the inter-frame time interval. Cars possibly move sideways and forward quite far within a period of half a second or more. Therefore, a motion model for predicting a vehicle's position in the next image is necessary.

3.1.1 Motion Model for Single Vehicles: We suppose that cars generally move in a controlled way, i.e. certain criteria describing speed, motion direction and acceleration should be met. To better incorporate the continuity of motion direction, we consider also a third image of the sequence. Figure 4 illustrates some of these cases. For instance, there should be no abrupt change of direction and change of speed, i.e. abnormal acceleration, from one image to the others. In general, the correlation length of motion continuity is modelled depending on the respective speed of a car, i.e., for fast cars, the motion is expected to be straighter and almost parallel to the road axis. Slow cars may move forward between two consecutive images but cannot move perpendicular to the road axis or backwards in the next image. These model criteria are incorporated in our tracking evaluation described in section 3.2.

3.1.2 Motion Model for Vehicle queues: In more complex traffic situations, the motion model can be extended to consider also vehicle queues. For images taken with a frame rate of 1-3 Hz, the car topology within a queue changes very rarely from one image to the other, although one could think of more complex queue motion models that describe the interaction of cars in a Markov-Chain manner. Hence, we currently analyze only pairs of cars as shown in Fig. 5. The distance of two cars following each other might increase or decrease, of course with a lower bound depending on the vehicles' speed. The trailing car may start to pass the leading car and change lanes. However, the cars cannot switch their relative positions.

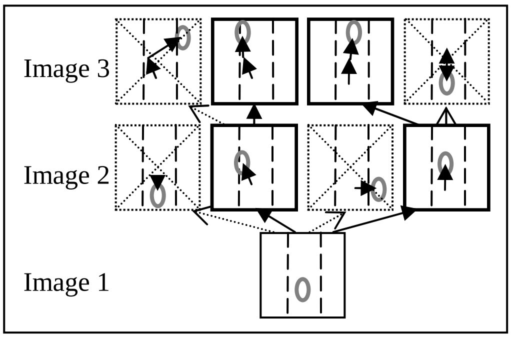


Figure 4. Examples for possible and impossible car movement

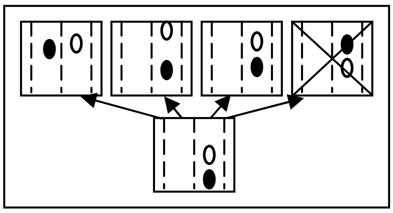


Figure 5. Vehicle queue behavior

3.2 Tracking procedure

In the current implementation, we focus on single car tracking in three consecutive images. Figure 6 shows the workflow of our tracking algorithm. As it can be seen, image triplets are used in order to gain a certain redundancy allowing an internal evaluation of the results. Of course, one could use more than three images for tracking. However, vehicles that move towards the flying direction only appear in few images so that the algorithm should also deliver reliable results for a low number of frames.

We start with the co-registration of the three images I1, I2, and I3, followed by car detection in I1 and the determination of a number of vehicle parameters which describe the actual state of a car, i.e. the distance to the road side polygon and the approximate motion direction (Sect. 3.2.1). Then, we create a vehicle image model C_1 by selecting a rectangle around the car. By using a shape-based matching algorithm, we try to find the car in the other images. In order to reduce the search, we select a RoI for the matching procedure based on the motion model (Sect. 3.2.2). The matching procedure delivers matches M_{12} in image I2 and the matches M_{13} in image I3. It should be mentioned, that both M_{12} and M_{13} contain multiple match results also including some wrong matches (see Fig. 7). As output of the matching algorithm, we receive the position of the match center.

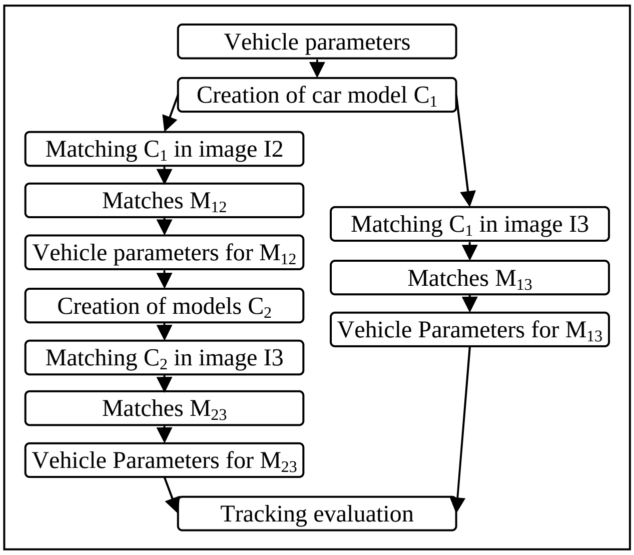

Vehicle parameters
Creation of car model C_1
Matching C_1 in image I2
Matches M_{12}
Vehicle parameters for M_{12}
Creation of models C_2
Matching C_2 in image I3
Matches M_{23}
Vehicle Parameters for M_{23}
Matching C_1 in image I3
Matches M_{13}
Vehicle Parameters for M_{13}
Tracking evaluation

Figure 6. Workflow for the vehicle tracking algorithm

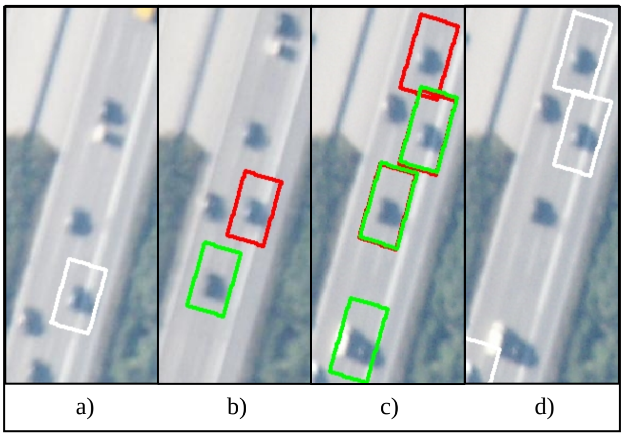


Figure 7. a) First image with detected car; b) second image with two matches M_{12} for C_1; c) third image with three matches M_{23} for each C_2 (corresponding matches are indicated by the same color; note the overlapping rectangles); d) third image with matches M_{13}

For each match M_{12}, vehicle parameters are calculated and new vehicle image models are created based on the match positions of M_{12}. These models are searched in image I3, eventually resulting in matches M_{23}, for which vehicle parameters are determined again. Finally, the results are evaluated and checked for consistency to determine the correct track combination of the matches (see Sect. 3.2.3).

3.2.1 Vehicle Parameters: The vehicle parameters are defined and determined as follows:

Distance to road polygon: The road polygon closest to a given vehicle is searched, and root point P_F is determined. This point is needed to approximate the direction of the car's motion.

Direction: A given vehicle's motion direction *dir(Car)* is approximated as a weighted direction derived from the three adjacent polygon segments, thus also considering curved road segments. The situation is illustrated in Fig. 8. The distances d_0 and d_1 between P_F and the end points of the central line segment L_n are determined. The weight of the angle of L_n is set to 1. The weight of the adjacent line segments' angles is computed using the relative distances d_0 and d_1. Note that d_0 is used to determine the weight of $ang(L_{n+1})$ while d_1 contributes to the weight of $ang(L_{n-1})$. This results in a higher weight for the angle of the closer adjacent line segment. The weights for both $ang(L_{n+1})$ and $ang(L_{n-1})$ add up to 1. Therefore, the overall weight sum is 2. The formula for *dir(Car)* is

$$dir(Car) = \frac{1}{2} \cdot \left(ang(L_n) + \frac{d_1}{d_0 + d_1} \cdot ang(L_{n-1}) + \frac{d_0}{d_0 + d_1} \cdot ang(L_{n+1}) \right).$$

3.2.2 Matching: For finding possible locations of a car in another image, we are using the shape-based matching algorithm proposed by (Steger, 2001) and (Ulrich, 2003). The core of this algorithm is visualized in Fig. 9. First, a model image has to be created. This is simply done by cutting out a rectangle of the first image around the car's center. The size of the rectangle is selected in such a way that both car and shadow as well as a part of the surrounding background (usually road) is covered by the area of the rectangle.

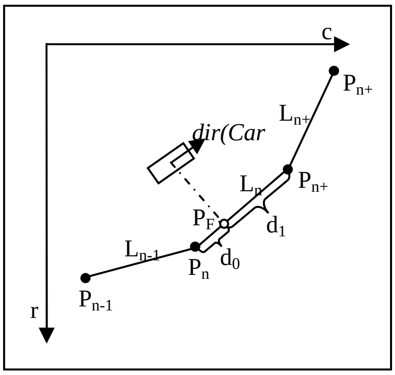


Figure 8. Approximation of the car's motion direction

Still, no other cars or distracting objects such as neighboring meadows should be within the rectangle. The rectangle is oriented in the approximate motion direction that has been calculated before.

A gradient filter is applied to the model image and the gradient directions of each pixel are determined. For run time reasons, only those pixels with salient gradient amplitudes are selected and defined as model edge pixels, in the following also referred as model points. In the RoI of the search image, the gradient filter is also applied. Finally, the model image is matched to the search image by comparing the gradient directions. In particular, a similarity measure is calculated representing the average vector product of the gradient directions of the transformed model and the search image. This similarity measure is invariant against noise and illumination changes but not against rotations and scale. Hence the search must be extended to a predefined range of rotations and scales, which can be easily derived from the motion model and the navigation data. To fulfill real-time requirements also for multiple matches, the whole matching procedure is done using image pyramids. For more details about the shape-based matching algorithm, see (Ulrich, 2003) and (Steger, 2001).

A match is found whenever the similarity measure is above a certain threshold. As a result, we receive the coordinates of the center, the rotation angle, and the similarity measure of the found match. To avoid multiple match responses close to each other, we limited the maximum overlap of two matches to 20%.

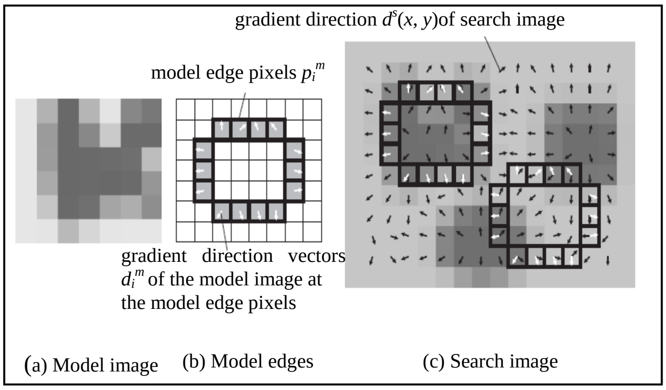

gradient direction $d^s(x, y)$of search image
model edge pixels p_i^m
gradient direction vectors d_i^m of the model image at the model edge pixels
(a) Model image
(b) Model edges
(c) Search image

Figure 9. Principle of the shape-based matching method, taken from (Ulrich, 2003), p. 70

3.2.3 Tracking Evaluation

The matching process delivers a number of match positions for M_{12}, M_{23}, and M_{13}. In our tests, we used a maximum number of the 6 best matches for each run. This means that we may receive up to 6 match positions for M_{12} and 36 match positions for M_{23} for each C_1. Also having 6 match positions for M_{13}, we need to evaluate 216 possible tracking combinations for one car. At a first glance, this seems quite cost intensive. Yet, many incorrect matches can be rejected through simple thresholds and consistency criteria so that the computational load can be controlled easily.

Evaluation scheme: As depicted in figure 10, we employ a variety of intermediate weights that are finally aggregated to an overall tracking score. Basically, these weights can be separated into three different categories, each derived from different criteria: *i*) First, a weight for the individual matching runs is calculated (weights w_{12}, w_{23}, and w_{13} in Fig. 10). Here, we consider the single car motion model and the similarity measure as output of the matching algorithm which is also referred to as matching score. *ii*) Based on these weights, a combined weight w_{123} for the combination of the matching runs M_{12} and M_{23} is determined. In this case, the motion consistency is the underlying criterion. *iii*) Finally, weights w_{33} are calculated for the combination of the match positions M_{23} and M_{13}. For a correct match combination, it is essential that the positions of M_{13} and M_{23} are identical within a small tolerance buffer.

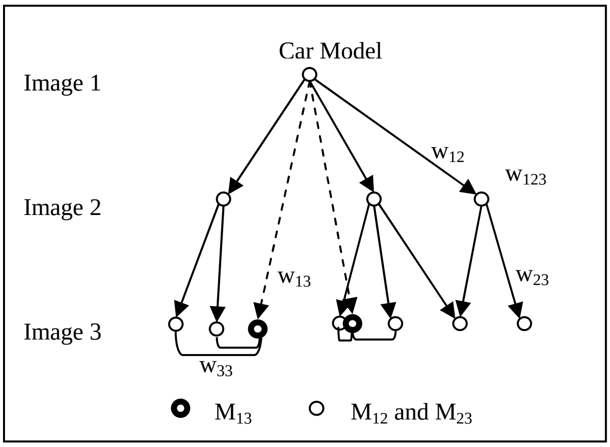


Figure 10. Diagram of the match evaluation process for one car

To avoid crisp thresholds and to allow for the handling of uncertainties, each criterion is mathematically represented as a Gaussian function

$$w(c,\mu,\sigma) = e^{-\frac{(c-\mu)^2}{2\sigma^2}}$$

with parameters mean μ and standard deviation σ evaluating the quality of an observation with respect to the criterion. By this, the weights are also normalized.

In the following, we will outline the calculation and combination of the different weights.

Single Tracking Run: The score w_{match} of the shape-based matching is already normalized (see (Ulrich, 2003) for details). In order to take into account the continuity criterion of a single car's motion, the difference between the motion direction in the first image (say of model C_1) and its conjugate in the next image (say match M_{12}) is considered. In addition, a displacement angle ang_{12} is also included, that essentially reflects the direction difference between the orientation of the trajectory from C_1 to M_{12} and the motion directions in C_1 and M_{12}. From this, the criteria value $Dcross_{12}$ is derived, penalizing across displacements regarding the expected direction $dirS_{12}$. To accommodate the fact that fast cars should move almost straight, $Dcross_{12}$ is multiplied by the distance vel_{12} between M_0 and M_{12}.

$$dirS_{12} = dir(C_1) + \frac{1}{2}\left(dir(C_1) - dir(M_{12})\right)$$

$$Dcross_{12} = \sin\left(dirS_{12} - ang_{12}\right) \cdot vel_{12}$$

The final weight w_{dir} for this criterion is obtained again by measuring its fit with the expected values represented by a Gaussian function. The combined weight w_{12} then calculates to

$$w_{12} = w_{match} \cdot w_{dir} \,.$$

Please note that the formulae above also hold for very slow or even parking cars, since a very small motion distance vel_{12} will scale down $Dcross_{12}$ and thereby allowing for nearly arbitrary direction differences.

Motion consistency: In order to exclude implausible combinations of matches, we examine the consistency of a car's trajectory over image triplets. The first criterion of this category is the change of velocity, i.e. the difference between vel_{12} and vel_{23}.

$$dvel_{13} = \left| vel_{12} - vel_{23} \right| .$$

In typical traffic scenarios accelerations of more than 1.5m/s^2 rarely happen, while a (nearly) constant speed is common. Again, such values are used to parameterize a Gaussian function resulting in weights w_{vel}.

In order to address the continuity of the trajectory, we carry out the very same calculations as for the single tracking run, now using C_1 and M_{23}, and compare it with the sum of $Dcross_{12}$ and $Dcross_{23}$ of the single displacements. If no difference appears, a car moves totally straight. Deviations from it are again modeled with a Gaussian function, eventually yielding weight w_{dis}. The weights w_{vel} and w_{dis} are combined to w_{123} by multiplication.

$$w_{123} = w_{vel} \cdot w_{dis}$$

Identity of M_{13} and M_{23}: As a last criterion, the identity of Matches M_{13} and M_{23} is checked (see Fig. 10). Weight w_{33} is simply the distance between the match positions of M_{13} and M_{23} put into a Gaussian function.

Final Weight: Assuming that the five individual measurements w_{12}, w_{23}, w_{13}, w_{123}, and w_{33} reflect statistically nearly independent criteria (which, in fact, does not perfectly hold), the final evaluation score W is computed as the product of the five weights:

$$W = w_{12} \cdot w_{23} \cdot w_{13} \cdot w_{33} \cdot w_{123}$$

The correct track is selected as that particular one yielding the best evaluation, however, as long as it passes a lower rejection threshold. Otherwise, it is decided that there is no proper match for a particular car. This may happen when a car is occluded by shadow or another object, but also when it leaves the field-of-view of the images. The latter case can of course be predicted based on a car's previous trajectory. Please note that the track evaluation allows a straightforward extension to more frames or even the tracking of multiple hypotheses if, e.g., the second best track reaches nearly the score of the best track. This option will be included in future work.

4. RESULTS AND DISCUSSION

We tested our algorithm on image triplets. These images have been acquired with a Minolta DiMAGE 7i 5Mpixel camera at a frame rate of 2 Hz. The focal length was approximately 50mm. The approximate flight altitude was between 2000 and 3000 m, therefore we have a ground pixel size of roughly 25-40 cm.

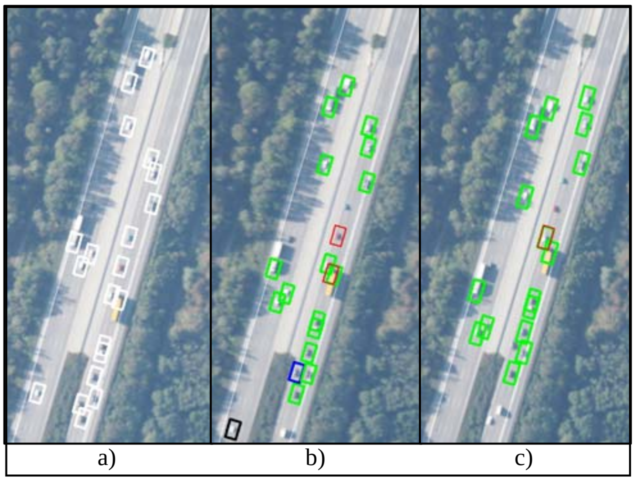


Figure 11. Results of the tracking in the test image.
a) Detected vehicles in the first image; b) associated cars in the second image; c) final track positions in the third image; see text for explanation of the color coding.

Figure 11 shows the tracking results for one cut-out of an image triplet. It depicts a quite busy highway with cars traveling with different velocities. What makes it also challenging is the presence of the severe shadows on the left carriage way of the highway.

Correctly tracked cars are marked green while incorrect track results are marked red. Black rectangles mark cars which were correctly matched in the second image but moved out of the field-of-view of the third image. Blue marked vehicles are correctly matched in the second image but could not be tracked in the third image even though they were present. In this triplet, 16 out of 20 cars could be correctly tracked. One car moved out of sight in the third image, therefore the comparison with the third image failed. One car was incorrectly tracked. Two cars couldn't be found in the third image although they were present, one of those was at least correctly found in the second image. Note that it is possible that correct and incorrect tracks overlap in the third image. This is the case for the car in front of the yellow bus in Fig. 11. The car itself was tracked correctly, but was also falsely assigned to another car.

In other image triplets with less dominant shadows, correct tracks were found for roughly 90% of the vehicles. However, more testing especially with larger and more variable scenes is still essential. The results reached so far are nonetheless very promising and show the potential of our approach.

The total computation time for all tracks was approximately 5-6 seconds on a 1.8 GHz standard computer. The tracking time for the fourth and following images will further decrease since prior knowledge from the first image triplet can be introduced to better restrain the regions of interest. In addition, we have to mention that the current C++ implementation is by far not yet optimized.

5. FUTURE WORK

As mentioned in Sect. 2 , we want to integrate the tracking approach with an automatic vehicle detection module including lane extraction in the near future. Concerning the tracking, it is planned to apply our approach not only to individual image triplets but – sequentially – also to longer image sequences in order to recover the whole trajectory of each car. Furthermore, when tracking vehicles in longer image sequences, we are planning to extent the motion model by an adaptive component so that besides evaluating the speed and acceleration of a car, the relations to neighboring cars can also be integrated into the evaluation. This would allow a more strict limitation of the search area and deliver a much more precise measure for tracking evaluation. Another area of research would be the detection and integration of context information such as large shadow areas or partial occlusions to be able to also track vehicles that were partially lost during the tracking.

REFERENCES

Dubuisson-Jolly, M.-P., Lakshmanan, S. and Jain, A. (1996): Vehicle Segmentation and Classification Using Deformable Templates. IEEE Trans. on Pattern Analysis and Machine Intelligence 18 (3): 293–308.

Dreschler, L., Nagel, H-H. (1982): Volumetric model and trajectory of a moving car derived from monocular TV frame sequence of a street scene. CGIP, 20: 199-228.

Ernst, I., Hetscher, M., Thiessenhusen, K., Ruhé, M., Börner, A., and Zuev, S. (2005): New approaches for real time traffic data acquisition with airborne systems. Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 3/W24, 69-73.

Haag, M. and Nagel, H.-H. (1999): Combination of Edge Element and Optical Flow Estimates for 3D-Model-Based Vehicle Tracking in Traffic Sequences. Int. Journal of Computer Vision 35 (3): 295–319.

Hinz, S. (2005): Fast and Subpixel Precise Blob Detection and Attribution . Proceedings of ICIP 05, Sept. 11-14 2005, Genua.

Hinz, S. (2004): Detection of vehicles and vehicle queues in high resolution aerial images. Photogrammetrie-Fernerkundung-Geoinformation, 3/04: 201-213.

Hinz, S., Baumgartner, A. (2003): Automatic Extraction of Urban Road Nets from Multi-View Aerial Imagery. ISPRS Journal of Photogrammetry and Remote Sensing 58/1-2: 83–98.

Lachaise, M. (2005): Automatic detection of vehicles and velocities in aerial digital image series. Diploma Thesis, Universitee Lyon.

Meffert B, Blaschek R, Knauer U, Reulke R, Schischmanow A, Winkler F (2005): Monitoring traffic by optical sensors. Proc. of Second International Conference on Intelligent Computing and Information Systems (ICICIS 2005): 9-14.

Rajagopalan, A., Burlina, P. and Chellappa, R. (1999): Higher-order statistical learning for vehicle detection in images. Proceedings of the International Conference on Computer Vision, 1999.

Reinartz P., Krauss T., Pötzsch M., Runge H., Zuev S. (2005): Traffic Monitoring with Serial Images from Airborne Cameras. Proc. of Workshop on High-Resolution Earth Imaging for Geospatial Information, Hannover, 2005.

Steger, C. (2001): Similarity measures for occlusion, clutter, and illumination invariant object recognition. In: B. Radig and S. Florczyk (eds.) *Pattern Recognition*, DAGM 2001, LNCS 2191, Springer Verlag, 148–154.

Stilla, U., Michaelsen, E., Soergel, U., Hinz, S., and Ender, J., 2004. Airborne monitoring of vehicle activity in urban areas. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXV, Part B3, 973-979.

Tan, T., Sullivan, G. and Baker, K. (1998): Model-Based Localisation and Recognition of Road Vehicles – International Journal of Computer Vision 27 (1): 5–25.

Toth, C. K., Grejner –Brezinska, D. and Merry, C., 2003. Supporting traffic flow management with high-definition imagery. Proceedings of the Joint ISPRS Workshop on High Resolution Mapping from Space 2003. Hannover, Germany. 6-8 October, (on CDROM).

Ulrich, M., 2003. Hierarchical Real-Time Recognition of Compound Objects in Images. Dissertation, German Geodetic Commission (DGK), Vol. C.

A TEST OF AUTOMATIC ROAD EXTRACTION APPROACHES

Helmut Mayer[†], Stefan Hinz[‡], Uwe Bacher[*], Emmanuel Baltsavias[⋆]

[†]Chair of Photogrammetry and Remote Sensing, Bundeswehr University Munich, Germany; Helmut.Mayer@unibw.de
[‡]Remote Sensing Technology, Technische Universität (TU) München, Germany; Stefan.Hinz@bv.tu-muenchen.de
[*]Geosystems GmbH, Germering, Germany; u.bacher@geosystems.de
[⋆]Chair of Photogrammetry, ETH Zurich, Switzerland; manos@geod.baug.ethz.ch

KEY WORDS: Automatic Road Extraction, Evaluation, Test

ABSTRACT:

Roads are important objects for many applications of topographic data. They are often acquired manually and as this entails significant effort, automation is highly desirable. Deficits in the automatic extraction hindering a wide-scale practical use have led to the idea of setting-up a EuroSDR test comparing different approaches for automatic road extraction. The goal is to show the potential of the state-of-the-art approaches as well as to identify promising directions for research and development. After describing the data and the evaluation criteria used, we present the approaches of a number of groups which have submitted results and give a detailed discussion of the outcome of the evaluation of the submitted results. We finally present a summary and conclusions.

1. MOTIVATION AND BACKGROUND

The need for accurate, up-to-date, and detailed information for roads is rapidly increasing. They are used in a variety of applications ranging from the provision of basic topographic infrastructure, over transportation planning, traffic and fleet management, car navigation systems, location based services (LBS), and tourism, to web-based applications. While road extraction has been performed by digitizing maps, the update and refinement of the road geometry is often based on aerial imagery or high resolution satellite imagery such as Ikonos or Quickbird. Additionally, terrestrial methods, particularly mobile mapping are of significant importance for determining attributes for navigational purposes.

Because road extraction from imagery, on which we focus for the remainder of this paper, entails large efforts in terms of time and money, automation of the extraction is of high potential interest. Full automation of the extraction of topographic objects is currently practically impossible for almost all applications and thus a combination with human interaction is necessary. An important factor hindering the practical use of automated procedures is the lack of reliable measures indicating the quality and accuracy of the results, making manual editing lengthy and cumbersome. Manufacturers of commercial systems have developed very few tools for semi-automated extraction and their cooperation with academia has been minimal. Thus, users and producers of such data, including national mapping agencies (NMAs) and large private photogrammetric firms, have been left with many wishes to be fulfilled.

NMAs increasingly plan to update their data in shorter cycles. Their customers have increasing demands regarding the level of accuracy and object modeling detailedness, and often request additional attributes for the objects, e.g., the number of lanes for roads. The insufficient research output and the increasing user needs, necessitate appropriate actions. Practically oriented research, e.g., the ATOMI project at the ETH Zurich (Zhang, 2004), has shown that an automation of road extraction and update is feasible to an extent that is practically very relevant. Companies that have developed semi-automated tools for building extraction and other firms too, could very well offer similar tools for roads.

These considerations led to the idea of setting-up a road extraction test under the umbrella of EuroSDR (European Spatial Data Research – www.eurosdr.net). An important inspiration for it was the highly successful 3D reconstruction test of (Scharstein and Szeliski, 2002) which has become a standard in the field. The emphasis of our test is put on the thorough evaluation of the current status of research (including models, strategies, methods and data used). Through testing and comparing existing semi- or fully automated methods based on various datasets and high quality reference data extracted manually by an experienced operator from the image data used for the test, weak points as well as promising directions should be identified and, strategies and methods that lead to a fast implementation of operational procedures for road extraction, update, and refinement should be proposed. However, since most of the participating groups focus on road extraction rather than on refinement or update, the scope of this test has been limited purely on road extraction for the time being.

2. DATA AND TEST SET-UP

Initially, eight test images were prepared from different aerial and satellite sensors. All images have a size of at least 4,000 × 4,000 pixels. Unfortunately, this was found to be insurmountable by nearly all approaches and, therefore, the limiting factor of the test. Reasons for an inability to process the larger scenes were apparently twofold: First, because of missing functionality for processing the whole image in patches which are then combined into one solution, intermediate results just exceeded the available memory. Second, even if this had not been the case, the time it takes to process the images together with the need to adapt the parameters to all variations in the larger scenes, meant these images required too much effort for most people. Hence we decided eventually to cut out three smaller parts with 1,600 × 1,600 pixels of Ikonos images where we found the largest interest.

In the following, only those images are listed, for which at least three extraction results were submitted:

- 3 scanned aerial images from the Federal Office of Topography, Bern, Switzerland (image scale 1 : 16 000, focal length 0.3 m, RGB, 0.5 m ground resolution, 4 000 × 4 000 pixels – see Fig. 1))
  - Aerial1: suburban area in hilly terrain
  - Aerial2: hilly rural scene with medium complexity
  - Aerial3: hilly rural scene with low complexity
- 3 IKONOS images (Geo) from Kosovo, provided by Bundeswehr Geoinformation Office (AGeoBw), Euskirchen, Germany, given as pan-sharpened images in red, green, blue, and infrared (1 600 × 1 600 pixels – see Fig. 1 and 2)
  - Ikonos1-Sub1: urban/suburban area in hilly terrain

- Ikonos3-Sub1 and -Sub2: rural hilly scenes with medium complexity

For evaluation we use criteria put forward by (Wiedemann et al., 1998). The basic assumption is that reference data is available in the form of the center lines of the roads. Additionally, it is assumed that only roads within a buffer of a certain width, usually the average width of the roads, around the road, here 5 pixels on both sides, i.e., 10 m for the Ikonos data, are correct. The extracted roads which are inside the buffer of the given reference roads and vice versa are determined via matching of the respective vector data. The most important criteria defined by (Wiedemann et al., 1998) based on these matching results to which we have restricted the analysis are:

Completeness: This is the percentage of the reference data which is explained by the extracted data, i.e., the part of the reference network which lies within the buffer around the extracted data. The optimum value for completeness is 1.

Correctness: It represents the percentage of correctly extracted road data, i.e., the part of the extracted data which lie within the buffer around the reference network. The optimum value for correctness is 1.

RMS (root mean square): The RMS error expresses the geometrical accuracy of the extracted road data around the reference network. In the given evaluation framework its value depends on the buffer width w. If an equal distribution of the extracted road data within the buffer around the reference network is assumed, it can be shown that $RMS = w/\sqrt{3}$. The optimum value is $RMS = 0$. As RMS mainly depends on the resolution of the image, it is given in pixels in this paper.

The reference data has an estimated precision of half a pixel. It comprises major and secondary roads, but no paths or short driveways. The reference data has not been made available to the participants. The participants usually asked only once or twice for an evaluation, i.e., no optimization in terms of the reference data was pursued. Opposed to (Scharstein and Szeliski, 2002) we allowed people to optimize their parameters for each and every image, as constant parameters were seen as too challenging.

3. ROAD EXTRACTION APPROACHES

We will shortly introduce the approaches of the participating groups (alphabetical ordering according to corresponding author):

Uwe Bacher and Helmut Mayer, Institute for Photogrammetry and Cartography, Bundeswehr University Munich, Germany: The approach is only suitable for the Ikonos images and is focusing on rural areas where roads are mostly homogeneous and are not disturbed by shadows or occlusions. It is based on earlier work from TU München of (Wiedemann and Hinz, 1999) and partially (Baumgartner et al., 1999). The approach of (Wiedemann and Hinz, 1999) starts with line extraction in all spectral bands using the sub-pixel precise Steger line extractor (Steger, 1998) based on differential geometry and scale-space including a thorough analysis and linking of the topology at intersections. The lines are smoothed and split at high-curvature points. The resulting line segments are evaluated according to their width, length, curvature, etc. Lines from different channels or extracted at different scales, i.e., line widths, are then fused on a best first basis. From the resulting lines a graph is constructed, supplemented by hypotheses bridging gaps. After defining seed lines in the form of the best evaluated lines, optimal paths are computed in the graph and from it gaps to be closed are derived. Bacher has extended this by several means (Bacher and Mayer, 2005). The central idea is to take into account the spectral information by means of a (fuzzy) classification approach based on fully automatically created training areas. For the latter parallel edges are extracted in the spirit of (Baumgartner et al., 1999) in a buffer around the lines and checked if the area in-between them is homogeneous. The information from the classification approach is used to evaluate the lines. Additionally, it is the image information when optimizing snakes to obtain a more geometrically precise, but also more reliable basis for bridging larger gaps in the network, which is another novel feature of Bacher's approach.

Charles Beumier and Vinciane Lacroix, Signal and Image Center, Royal Military Academy, Brussels, Belgium: The approach for Ikonos images rests on the line detector of (Lacroix and Acheroy 1998) which assumes that the gradient vectors on both sides of a line are pointing in opposite directions. Bright lines are extracted from the green channel with a slight Gaussian smoothing employing non-maximum suppression. Lines are tracked with limited direction difference until a minimum strength is reached. Lines are only kept if they are at least 30 pixels long and are straight enough when checked based on the square root of the inertial moment. For each of the line points the Normalized Difference Vegetation Index (NDVI) is computed from the red and the infrared channel and if it is below zero, the point is supposed to be vegetation and is rejected. Finally, the rest of the points are again tracked and checked to see if they are still long and straight enough.

Markus Gerke and Christian Heipke, Institute for Photogrammetry and Geoinformation (IPI), Hannover University, Germany: They use two approaches suitable for aerial images as well as for Ikonos data, both designed primarily for rural areas as Bacher above. Gerke_W is the approach of (Wiedemann and Hinz, 1999) – see Bacher above. Gerke_WB consists of a combination of Gerke_W with the approach of (Baumgartner et al., 1999). The latter is based on extracting parallel edges with an area homogeneous in the direction of the road in between in the original high resolution image and fusing this information with lines extracted at a lower resolution. Herewith it combines the high reliability of high resolution with the robustness against disturbances particularly for the topology of the lower resolution. Quadrangles are constructed from the parallel edges and, from them, in turn longer road objects taking also local context information into account. Gerke_WB in essence substitutes the Steger line extractor of the original Baumgartner approach by the full-fledged (Wiedemann and Hinz, 1999) approach and additionally puts less weight on the homogeneity in the direction of the road. Gerke notes that there is still room for improvement as he has not at all optimized the snakes used to bridge gaps.

Jose Malpica and Jose Mena, Subdirección de Geodesia y Cartografía, Escuela Politécnica, Campus Universitario, Alcalá de Henares, Spain: This approach (Mena and Malpica, 2003; Mena and Malpica, 2005) makes heavy use of the spectral and color characteristics of roads learned from training data. The latter is usually generated based on (possibly outdated) GIS data from the given image data. The basic image analysis is done on three statistical levels. On the first level, only color information is employed using Mahalonobis distance. On the so-called ”one and a half order” level, the color distribution is determined for a pixel and its 5 x 5 neighborhood and compared to the learned distribution via Bhattacharyya distance. Bhattacharyya distance is also used on the ”second order” statistical level where, for six different cross-sections of a 3 x 3 neighborhood of a pixel, co-occurrence matrices and from them 24 Haralick features are computed. The three statistical levels are normalized and combined employing the Dempster-Shafer Theory of Evidence. After thresholding and cleaning the derived plausibility image for roads is the basis for deriving the main axes of the roads. A standard skeleton showing all, including the usually unwanted details, is combined with a coarse skeleton to obtain a graph with precise road segments without too many wrong short road segments. The segments in the graph are finally subject to a geometrical as well as topological adjustment.

Karin Hedman and Stefan Hinz, Institute for Photogrammetry and Cartography, Technische Universität München, Germany: Like Bacher's and Gerke's approaches it again rests on (Wiedemann and Hinz, 1999) and is particularly suitable for rural areas. It has only been used for the smaller pieces cut from the Ikonos images. As Hedman and Hinz found that line extraction is the critical point, they have optimized it: First, they noted that the blue channel gives the best results, with the NDVI adding little, but complementary, information particularly for rural areas. For Ikonos3-Sub1 they found that it was advantageous to use two different scales for line extraction in the blue channel. They also note that it is surprising and needs further investigations that the blue channel delivers the best results for line extraction, since this spectral range is supposed to be significantly affected by atmospheric attenuation.

Qiaoping Zhang and Isabelle Couloigner, Department of Geomatics Engineering, University of Calgary, Canada: The approach is used with minor modifications for all test images. The two Aerial images were re-scaled by a factor of two. At the core of the approach of Zhang and Couloigner is K-means clustering with the number of classes set to an empirically found value of six. For most of the images three channels were used. The infrared channel was only employed for Ikonos1-Sub1; for the other two Ikonos sub-images it was regarded as too noisy. From one or more clusters the road cluster is constructed by a fuzzy logic classifier with predefined membership functions. The road cluster is refined by removing big open areas, i.e., buildings, parking lots, fields, etc., again by means of a fuzzy classification based on a shape descriptor using the Angular Texture Signature (Zhang and Couloigner, 2006a; Zhang and Couloigner, 2006b). Road segments are found from the refined clusters via a localized and iterative Radon transform with window size 31×31 pixels with improved peak selection for thick lines. The segments are grouped bridging gaps smaller than five pixels and forming intersections. Finally, only segments longer than twenty pixels are retained.

4. RESULTS AND DISCUSSION

The results of the evaluation are summarized in Table 1 linking particularly good results to Figures 1 and 2. The table is ordered in the first instance according to the test areas (from aerial to satellite data) and in the second instance alphabetically according to the group and possibly its approaches. For each test area the best result in terms of the geometric mean of completeness and correctness is marked in bold. In addition, all values for completeness or correctness which are beyond a value of 0.6 or 0.75 respectively, are marked in bold. These numbers can be seen as a lowest needed limit so that the results become practically useful. The value for correctness was set to a higher value as experience shows that it is much harder to manually improve given faulty results than to acquire roads from scratch. To be of real practical importance, in many cases both values probably need to be even higher, e.g., for correctness around 0.85 and for completeness around 0.7, but we have chosen the lower values, to distinguish 'the probably useful' for the obtained results from the rest.

We focus the analysis on the details from the Ikonos images, as it is only for these smaller images that we have received a larger number of results. We comment on the images, discuss the individual approaches and give important overall findings. For the different images, we observed the following:

Aerial1–3: All three images have only been processed by Gerke and Zhang. The latter performs best for Aerial1 (Fig. 1a), which is the most difficult of the three images showing a suburban area.

No	Name **(best)**	Completeness (≥ **0.6)**	Correctness (≥ **0.75)**	RMS [pix]
	Aerial1			
1	Gerke_W	0.46	0.47	3.74
2	Gerke_WB	0.31	0.56	1.53
3	**Zhang** (Fig. 1a)	0.51	0.49	1.92
	Aerial2			
4	Gerke_W	**0.76**	0.66	2.87
5	**Gerke_WB** (Fig. 1b)	**0.65**	**0.82**	1.14
6	Zhang	**0.67**	0.49	1.72
	Aerial3			
7	Gerke_W	**0.81**	0.63	3.14
8	**Gerke_WB** (Fig. 1c)	**0.72**	**0.77**	1.3
9	Zhang	**0.72**	0.63	1.66
	Ikonos1_Sub1			
10	Bacher	0.34	0.66	1.29
11	**Beumier** (Fig. 1d)	0.48	0.69	1.3
12	Gerke_W	0.27	0.41	1.89
13	Gerke_WB	0.19	0.49	1.91
14	Hedman	0.31	0.51	1.25
15	Malpica	0.25	0.74	1.13
16	Zhang	0.56	0.41	1.52
	Ikonos3_Sub1			
17	**Bacher** (Fig. 2a)	**0.81**	**0.87**	0.97
18	Gerke_W	**0.8**	0.65	1.53
19	Gerke_WB	**0.68**	**0.75**	1.99
20	Hedman (Fig. 2b)	**0.77**	**0.78**	1.16
21	Malpica (Fig. 2c)	**0.6**	**0.79**	1.41
22	Zhang	**0.72**	0.35	1.22
	Ikonos3_Sub2			
23	Bacher (Fig. 2d)	**0.86**	**0.89**	1.
24	Gerke_W	**0.75**	0.52	1.35
25	Gerke_WB (Fig. 2e)	**0.71**	**0.84**	1.7
26	**Hedman** (Fig. 2f)	**0.85**	**0.91**	1.19
27	Malpica	**0.6**	**0.89**	1.59
28	Zhang	**0.7**	0.34	1.18

Table 1. Results of the evaluation. Bold names represent the best result for a test area in terms of the geometric mean of completeness and correctness. Bold numbers are beyond 0.6 or 0.75 for completeness or correctness, respectively.

It seems that for it the loss of information by down-sampling by a factor of two by Zhang is more than made up by employing color information via classification, a feature Gerke is lacking. Gerke_WB gives the best results in terms of completeness and correctness for images 2 and 3 (Fig. 1b and c) showing rural areas for which it was designed. Particularly the result for Aerial3 is on a level which could be a viable basis for a practical application. Finally, comparing the results for Gerke_W and Gerke_WB one can see nicely how for Gerke_WB completeness is still sacrificed for correctness even when introducing additional information in the form of the homogeneity in the road direction for the original high resolution imagery.

Ikonos1-Sub1: This image shows an urban/suburban scene and has been processed by six approaches, none giving a practically useful result. It seems to be too hard a challenge for the current approaches. Beumier has only submitted this one result (Fig. 1d), but it is the best for this scene. It shows that a good line extractor combined with spectral information (NDVI) and well chosen constraints on the geometry can produce a pretty good result. In terms of a trade-off between completeness and correctness Bacher, Malpica, and Zhang are similarly good. Looking at the individual results, however, one can very well see the different balance. Yet, for practical applications, the high correctness values for Bacher and Malpica would probably be preferred compared to Zhang.

Ikonos3-Sub1 and -Sub2: These two images depicting rural scenes of medium complexity are the only ones for which a larger number of approaches, namely six, was applied and for which at least some of the results (see also Fig. 2) are in a range, which could be suitable for a practical application. This is particularly true for the approaches of Hedman and Bacher. Both rest on the approach of (Wiedemann and Hinz, 1999) and both make use of color information, Bacher in a more sophisticated way than Hedman. While Bacher has a clear edge for Sub1, Hedman is slightly better on Sub2. Gerke's approaches also rest on (Wiedemann and Hinz, 1999), but are less sophisticated in the way they make use of the color information, which seems to be a clear disadvantage here. While for Sub1 Gerke_W and _WB perform very similarly, Gerke_WB taking into account the homogeneity of the road in the original resolution is markedly better on Sub2. For the approaches based on color and texture Malpica achieves a higher quality particularly in terms of correctness than Zhang. This is especially true for Sub1. Finally, a comparison of the results for Bacher and Malpica shows the benefits of global network optimization inherent in all approaches based on (Wiedemann and Hinz, 1999) together with snakes for bridging gaps. It is clearly visible that in Bacher's result many smaller gaps are bridged meaningfully.

We next comment on distinct characteristics of the individual approaches, if they have not been discussed already with the images:

Bacher, Gerke_W and _WB, and Hedman: All three follow (Wiedemann and Hinz, 1999), the difference being which additional information is used. Bacher, with classification based on automatically generated training data is the most sophisticated and achieves the best results, but also Hedman's suitable selection of channels and scales as well as the use of the NDVI is sufficient to outperform Gerke_W and _WB, which do not make explicit use of color.

Gerke_W versus _WB: Gerke_WB can be seen as an extension of Gerke_W, taking into account higher resolution information in the form of parallel edges enclosing a region homogeneous in the direction of the road. Gerke_WB enforces more detailed constraints and, thus, as expected, the results for it show a lower completeness, but a higher correctness. This is true in all cases, but Gerke_WB seems to be particularly well suited for open rural scenes, where the roads mostly match its model of homogeneous areas.

Malpica and Zhang: Both employ a classification approach using color or multispectral information, though in a different way. While Malpica also includes textures and learns the characteristics from given GIS data, Zhang uses an unsupervised classification. Malpica outperforms Zhang for the Ikonos data, but Zhang is more flexible with the unsupervised classification producing results for most images, also the ones not reported here.

Finally, we want to note some important **overall findings**:

The approaches based on line extraction, i.e., Bacher, Gerke_W, Gerke_WB and Hedman based on the Steger extractor as well as Beumier built on top of Lacroix's work give better results for the more line-like high resolution Ikonos data than approaches based on pixel-wise or local classification, i.e., Zhang and Malpica. It would be interesting to see how the latter perform on higher resolution aerial images where the line structure of the image is less marked and the spectral information should be of higher quality. Please note that the Ikonos data available for the test are pan-sharpened with a physical resolution for the color of only about 4 m. One issue still to be investigated is the use of the original low resolution images.

There is a trend to use color / multispectral information particularly for high resolution satellite data. This is done either as simple as the NDVI (Beumier and Hedman) or based on a more or less sophisticated classification (Bacher, Malpica, and Zhang). For the latter, training is done using given GIS data (Malpica), training areas are automatically generated from characteristic homogeneous road parts with parallel road sides (Bacher), or the classification is done unsupervised (Zhang). Network optimization and bridging gaps, e.g., by means of snakes, only seems to become important when a certain level of quality has been reached as, for example, by Bacher for Ikonos-Sub1 and -Sub2.

5. SUMMARY AND CONCLUSION

In summary, the results show that it is possible to extract roads with a quality in terms of completeness and correctness which should be useful for practical applications, although only for scenes with limited complexity, namely up to medium complex rural scenes. This is true for aerial as well as high resolution satellite data. The test has also demonstrated that most approaches cannot deal with images larger than about 2,000 × 2,000 pixels. This is probably due to missing functionality to process images in patches and shows that the approaches focus on furthering the understanding of the basic problems rather than on practical development, where robustness to all possible situations would be the central issue. With the advent of digital aerial cameras and high resolution satellite data making high quality color and spectral information available, there is a recent focus to employ this information for road extraction and the results show its usefulness. However, particularly for the high resolution data, there is still much to be done.

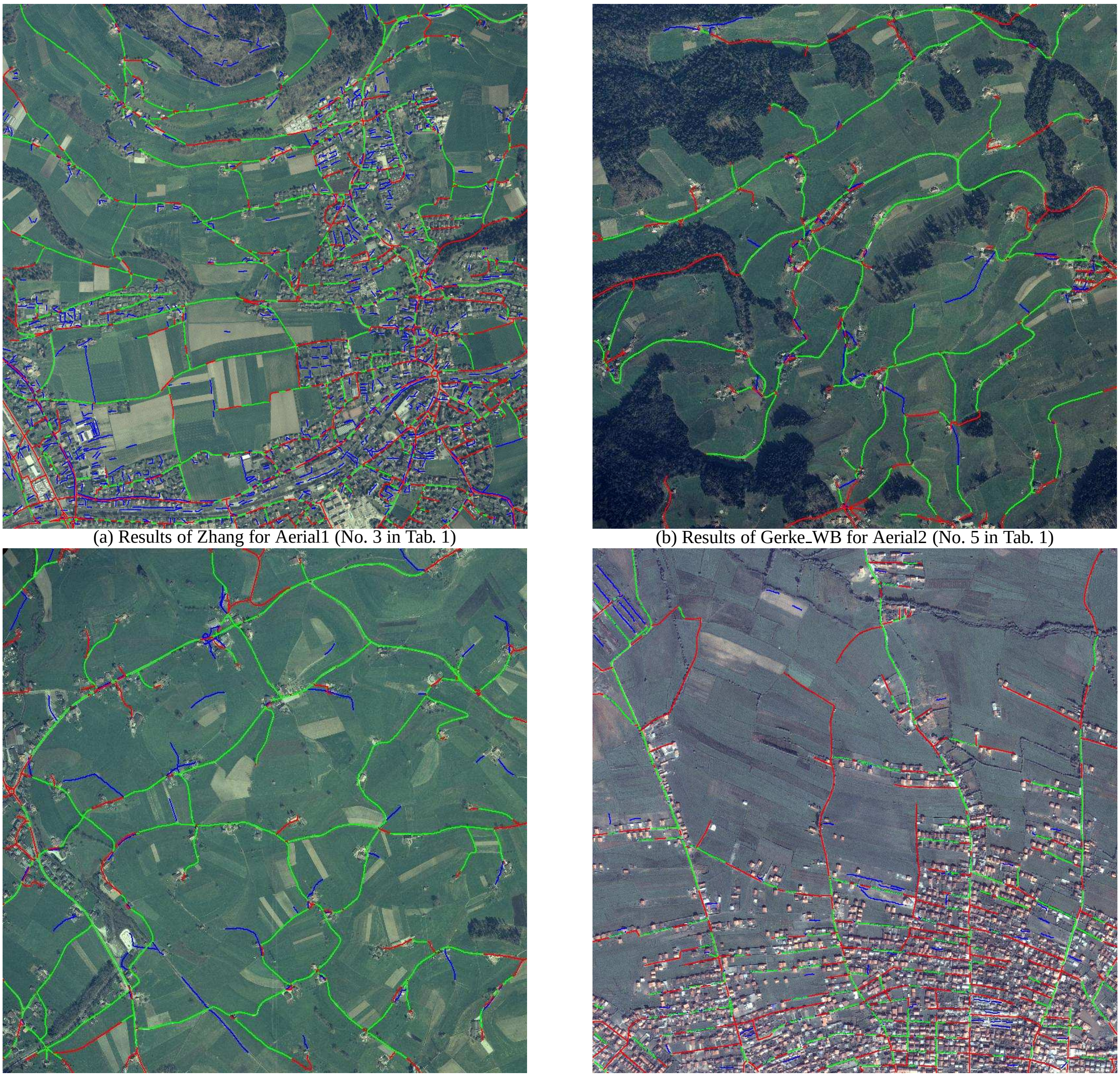
As it took nearly two years to obtain the results presented here, we learned the hard way that it is extremely important to see as very long term the evaluation of the results for different approaches based on the same data. Experience for similar tests, such as the highly successful 3D reconstruction test of (Scharstein and Szeliski, 2002) only gained momentum after some time. The goal has to be, that after a while, papers proposing a new approach only get accepted for higher level conferences when they show comparable or improved results on the test data compared to the state of the art. It is, thus, very important to continue this work.

Although there are only scarce resources both in academia and practice, we hope that this EuroSDR test will help to create a nucleus of interested researchers, who with the cooperation of NMAs, and if possible manufacturers, could form a well-coordinated and focused research network which can speed up the development of operational (or quasi-operational) systems for road extraction. Here, a focus should be on using a priori data. Though, we note that a fair test of these systems is a difficult issue due to the complexity of practical environments.

Promising directions for future research comprise statistical generative modeling. A particularly impressive instance is (Stoica et al., 2004) who have employed this kind of modeling for roads. To our knowledge this is the first time that the natural variability of the road network has been modeled in a realistic way.

ACKNOWLEDGMENTS

We thank the anonymous reviewers as well as Kevin Mooney and Nicolas Paparoditis for their helpful comments and gratefully acknowledge the funding of Uwe Bacher by Bundeswehr Geoinformation Office while he was with Bundeswehr University Munich. We are grateful to EuroSDR for giving us the possibility to conduct this test under their umbrella.



(a) Results of Zhang for Aerial1 (No. 3 in Tab. 1)

(b) Results of Gerke_WB for Aerial2 (No. 5 in Tab. 1)

(c) Results of Gerke_WB for Aerial3 (No. 8 in Tab. 1)

(d) Results of Beumier for Ikonos1-Sub1 (No. 11 in Tab. 1)

Figure 1. Results of EuroSDR Road Extraction Test: Correctly extracted roads are given in green, incorrectly extracted roads in blue, and missing roads in red.

REFERENCES

Bacher, U. and Mayer, H., 2005. Automatic Road Extraction from Multispectral High Resolution Satellite Images. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 36, 3/W24, pp. 29–34.

Baumgartner, A., Steger, C., Mayer, H., Eckstein, W. and Ebner, H., 1999. Automatic Road Extraction Based on Multi-Scale, Grouping, and Context. Photogrammetric Engineering and Remote Sensing 65(7), pp. 777–785.

Lacroix, V. and Acheroy, M., 1998. Feature-extraction Using the Constrained Gradient. ISPRS Journal of Photogrammetry and Remote Sensing 53, pp. 85–94.

Mena, J. and Malpica, J., 2003. Color Image Segmentation Using the Dempster-Shafer Theory of Evidence for the Fusion of Texture. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34 3/W8, pp. 139–144.

Mena, J. and Malpica, J., 2005. An Automatic Method for Road Extraction in Rural and Semi-Urban Areas Starting from High Resolution Satellite Imagery. Pattern Recognition Letters 26, pp. 1201–1220.

Scharstein, D. and Szeliski, R., 2002. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. International Journal of Computer Vision 47(1), pp. 7–42.

Steger, C., 1998. An Unbiased Detector of Curvilinear Structures. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(2), pp. 113–125.

Stoica, R., Descombes, X. and Zerubia, J., 2004. A Gibbs Point Process for Road Extraction from Remotely Sensed Images. International Journal of Computer Vision 57(2), pp. 121–136.

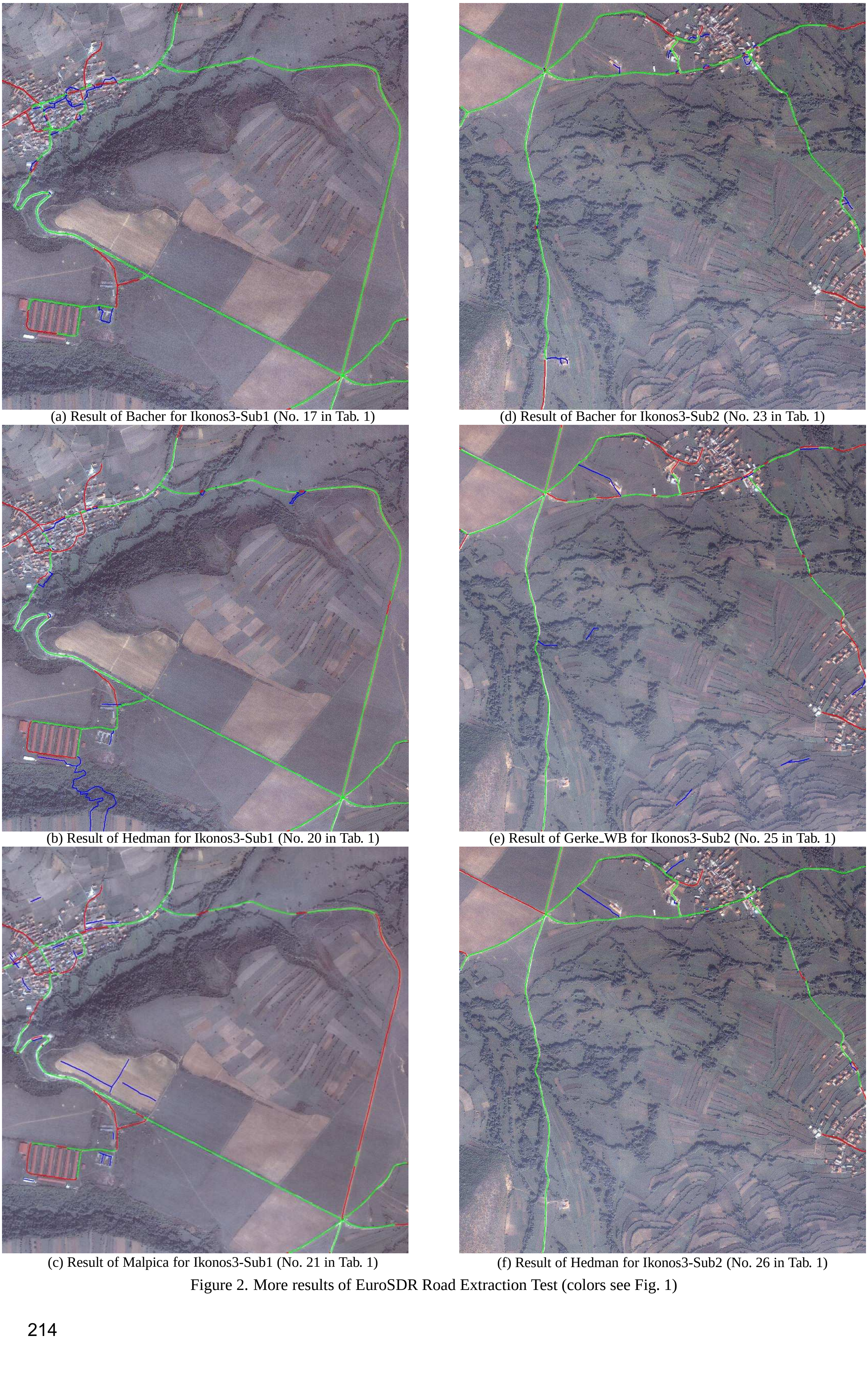
Wiedemann, C. and Hinz, S., 1999. Automatic Extraction and Evaluation of Road Networks from Satellite Imagery. In: International Archives of Photogrammetry and Remote Sensing, Vol. 32(3-2W5), pp. 95–100.

Wiedemann, C., Heipke, C., Mayer, H., and Jamet, O., 1998. Empirical evaluation of automatically extracted road axes. In: Empirical Evaluation Methods in Computer Vision, IEEE Computer Society Press, Los Alamitos, CA, pp. 172–187.

Zhang, C., 2004. Towards an Operational System for Automated Updating of Road Databases by Integration of Imagery and Geodata. ISPRS Journal of Photogrammetry and Remote Sensing 58, pp. 166–186.

Zhang, Q. and Couloigner, I., 2006a. Automated Road Network Extraction from High Resolution Multi-Spectral Imagery. In: ASPRS 2006 Annual Conference, Reno, Nevada, 10 pages.

Zhang, Q. and Couloigner, I., 2006b. Benefit of the Angular Texture Signature for the Separation of Parking Lots and Roads on High Resolution Multi-spectral Imagery. Pattern Recognition Letters 27(9), pp. 937–946.



(a) Result of Bacher for Ikonos3-Sub1 (No. 17 in Tab. 1)

(d) Result of Bacher for Ikonos3-Sub2 (No. 23 in Tab. 1)

(b) Result of Hedman for Ikonos3-Sub1 (No. 20 in Tab. 1)

(e) Result of Gerke_WB for Ikonos3-Sub2 (No. 25 in Tab. 1)

(c) Result of Malpica for Ikonos3-Sub1 (No. 21 in Tab. 1)

(f) Result of Hedman for Ikonos3-Sub2 (No. 26 in Tab. 1)

Figure 2. More results of EuroSDR Road Extraction Test (colors see Fig. 1)

SEMIAUTOMATIC ROAD EXTRACTION BY DYNAMIC PROGRAMMING OPTIMISATION IN THE OBJECT SPACE: SINGLE IMAGE CASE

A. P. Dal Poz[a, *], R. A. Gallis[b], J. F. C. da Silva[a]

[a] Dept. of Cartography, São Paulo State University, R. Roberto Simonsen, 305, Presidente Prudente-SP, Brazil, (aluir, jfcsilva@fct.unesp.br)

[b] Ph.D. Student in Cartographic Sciences, São Paulo State University, R. Roberto Simonsen, 305, Presidente Prudente-SP, Brazil, rodrigogallis@pos.prudente.unesp.br

Commission III

KEY WORDS: Dynamic Programming, road model, feature extraction, road extraction, aerial images.

ABSTRACT:

This article proposes a novel road extraction methodology from digital images. The innovation is based on the dynamic programming (DP) algorithm to carry out the optimisation process in the object space, instead of doing it in the image space such as the DP traditional methodologies. Road features are traced in the object space, which implies that a rigorous mathematical model is necessary to be established between image and object space points. It is required that the operator measures a few seed points in the image space to describe sparsely and coarsely the roads, which must be transformed into the object space to make possible the initialisation of the DP optimisation process. Although the methodology can operate in different modes (mono-plotting or stereo-plotting), and with several image types, including multisensor images, this paper presents details of our single image methodology, along with the experimental results.

1. INTRODUCTION

Road extraction from aerial and satellite imagery is of fundamental importance in the context of spatial data capturing and updating for GIS applications. Substantial work on road extraction has been accomplished since the 70's in computer vision and digital photogrammetry, with pioneering works by, e. g., Bajcsy and Tavakoli (1976) and Quam (1978). This research topic is still challenging, what is demonstrate, for example, by the fact that vendors of commercial photogrammetric system have not provided useful toolkits for automated road extraction, including practical semiautomatic ones.

The mentioned classification of the road extraction methods is related to the amount of automation incorporated by them. Semiautomatic methods depend on the intervention of an operator for identifying the road object and supplying a little information about it, as e.g. seed points. These methods include road-follower (McKeown and Denlinger, 1988; Vosselman and de Knecht, 1995; Dal Poz and Silva, 2002; Kim et al., 2004) and some kind of simultaneous curve fitting (Kass et al., 1987; Grüen and Li, 1997; Agouris et al., 2000; Hu et al., 2004; Merlet and Zerubia, 1996; Dal Poz and Vale, 2003). Automated methods try to completely circumvent human intervention during the extraction process. A sophisticated example is found in Baumgartner et al. (1999), in which different resolutions, grouping, and context are used to extract road networks from high-resolution images. Stoica et al. (2004) modelled the road network in remote sensed images as connected line segments, resulting in a probabilistic model to be solved by the Maximum a Posteriori (MAP) estimation. As a final example, Zhu et al. (2004) extracted linear features from laser data and used them to guide the road extraction from aerial images.

While such fully, or at least close to fully, automated processes have not reached a mature state, semiautomatic methods need to be developed or improved to allow the rapid, reliable and accurate provision of data for GIS systems. This article proposes a novel road extraction methodology from digital images. The innovation is based on the dynamic programming optimisation (DP) algorithm to carry out the optimisation process in the object space, instead of doing it in the image space such as the DP traditional methodologies. This paper is organised in four main sections. Section 2 presents our object space road extraction methodology using a single aerial image and a DTM. Results are presented and discussed in Section 3. Finally, conclusions are provided in Section 4.

2. OBJECT SPACE ROAD EXTRACTION USING A SINGLE IMAGE

2.1 Image space road models

Photometric and geometric road properties (as e.g. road is elongated and lighter than the background, road grey levels do not change much within a short distance, road is smooth etc.) are used to formulate a generic road model considering that the road can be represented by an image space polygon $P^i = \{p_1, ..., p_n\}$, where p_i is its i^{th} vertex. The generic road model can be formulated by the merit function (equation 1) and an inequality constraint (equation 2), as follows (Gruen and Li, 1997),

$$E = \sum_{i=1}^{n-1}((Ep_1(p_i, p_{i+1}) - \beta.Ep_2(p_i, p_{i+1}) + \gamma.Ep_3(p_i, p_{i+1})).[1+\cos(\alpha_i - \alpha_{i+1})]/|\Delta S_i|) \quad (1)$$

$$C_i = |\alpha_i - \alpha_{i+1}| < T \quad (2)$$

[*] Corresponding author.

where, $Ep_1(p_i, p_{i+1})$, $Ep_2(p_i, p_{i+1})$, and $Ep_3(p_i, p_{i+1})$ are functions describing geometric and radiometric road properties and depending on consecutive points p_i and p_{i+1}; α_i is the direction of the vector defined by points p_{i-1} and p_i; β and γ are positive constants; $|\Delta S_i|$ is the distance between points p_{i-1} and p_i; and T is a user-defined threshold for direction change between two adjacent vectors.

Analysing the merit function (equation 1), it is easily concluded that the function E is a sum of sub-functions E_i depending only on three consecutive points (p_{i-1}, p_i, p_{i+1}) of the polygon P^i, i.e,

$$E = \sum_{i=1}^{n-1} E_i(p_{i-1}, p_i, p_{i+1}) \qquad (3)$$

Due to the structure of equation 3, where only six variables are interrelated simultaneously, the DP algorithm can be used to efficiently solve the optimisation process, which is transformed into a sequential decision-making process (Gruen and Li, 1997). The solution of this problem is a 2D polygon $P^i = \{p_1, ..., p_n\}$ representing a user-selected road and corresponding to the maximum of merit function E. This function is appropriate to be used in semiautomatic road extraction process from low-resolution images (road widths ranging from 1 to 3 pixels). Mainly in high-resolution images roads usually manifest as wide and homogeneous ribbons. As a result, the extracted polygons would hardly represent the corresponding road axes. In order to avoid this problem, Dal Poz and Vale (2003) proposed an improvement in equation 3. Basically, an edge constraint term was added to the original merit function, resulting in a function (equation 4) that is a sum of sub-functions E_i^t. Each sub-function depends only on three consecutive points (p_{i-1}, p_i, p_{i+1}) of the polygon P^i and respective road widths (w_{i-1}, w_i, w_{i+1}).

$$E^m = \sum_{i-1}^{n-1} E_i^t(p_{i-1}, p_i, p_{i+1}, w_{i-1}, w_i, w_{i+1}) \qquad (4)$$

Equation 4 shows that the goal of the optimisation process by the DP algorithm is similar to the one based on equation 3. The basic difference is that the optimisation process should provide the polygon $P^i = \{p_1, ..., p_n\}$ representing the road centreline and the road widths at the respective vertices. Equation 4 also shows that 9 variable are interrelated simultaneously.

2.2 Object space road model

As shown before, the image space road model has as unknowns the image co-ordinates (L, C) of polygon vertices representing the road and, in case of high-resolution images, the road width for every polygon vertex as well. Equation 3 or 4 can be modified in order to express roads in function of ground co-ordinates. The resulting equation will be the basis for an optimisation problem by the DP algorithm. As a result, it needs to have an appropriate structure, like one of equation 3 or 4, thus allowing the DP algorithm to be advantageous for solving the optimisation problem. We show below that a merit function with these characteristics can be easily derived.

We start below with the modification of the equation 3. The basic prerequisite to work in this direction is the selection of the object space reference system in which the 3D vertices of polygons representing roads are referred to. UTM (Universe Transverse Macerator) co-ordinates (E, N) plus the ellipsoidal height (h) is adopted as the ground reference system. Although this reference system (E, N, h) is not cartesian, it is well-known that a mathematical relation between an object point (P(E, N, h)) referred to it and the corresponding point in the image reference system (p(L, C)) can be established rigorously. In order to establish this mathematical relation, many parameters are needed to be known, like the interior and exterior orientation parameters of the sensor, the datum and UTM map projection parameters, beside others.

For frame camera images, the relation between an image space point $p_i(L_i, C_i)$ and the corresponding object space point $P_i(E_i, N_i, h_i)$ can be established in function of known parameters, such as usual ones listed below:

- λ_{cm} is the longitude of the central meridian of a UTM fuse;
- a and f_e are respectively the semi major axis and the flattening of the ellipsoid;
- ϕ_0 ,λ_0, and h_0 are the geodetic co-ordinates (ϕ_0 ,λ_0) and the ellipsoidal height (h_0) of the origin of the local vertical reference system;
- κ, φ, ω, X_0, Y_0, and Z_0 are the exterior orientation parameters of the camera, previously computed by an orientation procedure, taking as reference the local vertical reference system;
- f is the focal length of the camera;
- x_0 and y_0 are the co-ordinates of the principal point;
- K_1, K_2, and K_3 are the parameters of radial lens distortions;
- P_1 and P_2 are the parameters of decentering lens distortions;
- ε_{45} is a refraction coefficient for a standard atmosphere, depending upon the flying height above mean sea level and the orthometric height of the object point P_i.

The mathematical relation that allows the transformation from object point $P_i(E_i, N_i, h_i)$ into image point $p_i(L_i, C_i)$ is too complex to be presented here. In fact, it involves object space reference system transformations, projective transformation by the collinearity equations, image space reference system transformations, and introduction of systematic errors to the computed image space points. In addition, mathematical concepts and formulae are well-known. Assuming that L_i and C_i image co-ordinates can be obtained from E_i, N_i, and h_i object co-ordinates by f_1 and f_2 equations, respectively, we have,

$$\begin{aligned} L_i &= f_1(\text{Par}, V_i) \\ C_i &= f_2(\text{Par}, V_i) \end{aligned} \qquad (5)$$

where Par = (λ_{mc}, a, f_e, ϕ_0, λ_0, h_0, κ, φ, ω, X_0, Y_0, Z_0, f, x_0, y_0, K_1, K_2, K_3, P_1, P_2, ε_{45}) and V_i= (E_i, N_i, h_i). Since Par is known, an image space point can be expressed as a function of only V_i, i. e.:

$$p_i(L_i, C_i) = p_i(f_1(V_i), f_2(V_i)) = p_i(f_1(E_i, N_i, h_i), f_2(E_i, N_i, h_i)) \qquad (6)$$

Expression 6 allows equation 3 to be rewritten as follows,

$$E = \sum_{i=1}^{n-1} E_i(p_{i-1}(L_{i-1}, C_{i-1}), p_i(L_i, C_i), p_{i+1}(L_{i+1}, C_{i+1})) =$$

$$\begin{aligned} &\sum_{i=1}^{n-1} E_i(p_{i-1}(f_1(E_{i-1}, N_{i-1}, h_{i-1}), f_2(E_{i-1}, N_{i-1}, h_{i-1})), \\ &\qquad p_i(f_1(E_i, N_i, h_i), f_2(E_i, N_i, h_i)), \\ &\qquad p_{i+1}(f_1(E_{i+1}, N_{i+1}, h_{i+1}), f_2(E_{i+1}, N_{i+1}, h_{i+1}))) \end{aligned} \qquad (7)$$

Equation 7 shows that it depends simultaneously on coordinates of three successive object points of a polygon representing a road in the object space, i.e.: $P_{i-1}(E_{i-1}, N_{i-1}, h_{i-1})$, $P_i(E_i, N_i\ h_i)$ e $P_{i+1}(E_{i+1}, N_{i+1}, h_{i+1})$. This means that equation 7 may be rewritten in the following way,

$$E = \sum_{i=1}^{n-1} E_i\,(P_{i-1}(E_{i-1}, N_{i-1}, h_{i-1}), P_i(E_i, N_i, h_i), P_{i+1}(E_{i+1}, N_{i+1}, h_{i+1})) \quad (8)$$

Equation 8 is ambiguous as in principle the energy value E can be the same for infinite object space polygons $P^o = \{P_1, ..., P_n\}$. This is a direct consequence of the well-known characteristic of the transformation given by equation 5, by which one can select infinite object points that map to the same image point. Consequently, equation 8 can not be used for extracting roads without imposing constraints to remove its ambiguity. This ambiguity can be removed if a DTM is available as roads can be enforced to lay on the DTM. This constraint takes the form $h_i=f(E_i,N_i)$, allowing equation 8 to be rewritten as follows,

$$E = \sum_{i=1}^{n-1} E_1(P_{i-1}(E_{i-1}, N_{i-1}), P_i(E_i, N_i), P_{i+1}(E_{i+1}, N_{i+1})) \quad (9)$$

The maximum of the energy E corresponds to a road runs on the DTM (figure 1), which can be efficiently found by the DP algorithm. Equation 9 shows that only six variables are interrelated simultaneously, implying a similar computational complexity when compared to the corresponding image space equation (equation 3). Equation 9 is the basis for road centreline extraction in the object space using a single low-resolution aerial image. In case of medium- and high-resolution aerial images, it is easily demonstrated that, starting from equation 4, the following equation form can be obtained,

$$E = \sum_{i=1}^{n-1} E_1(P_{i-1}(E_{i-1}, N_{i-1}), P_i(E_i, N_i), P_{i+1}(E_{i+1}, N_{i+1}), W_{i-1}, W_i, W_{i+1}) \quad (10)$$

where, W_{i-1}, W_i, and W_{i+1} are the road widths at points P_{i-1}, P_i, and P_{i+1}, respectively. During the optimisation procedure by DP, object space road widths are sampled and the corresponding values in the image space are required for enforcing the edge constraint. This means that a mathematical relation between image and object space widths is necessary. This mathematical relation can be approximately stated by means of the relation between the local image scale and road widths in both spaces. A rigorous mathematical relation can be also stated, but the approximate relation is efficient and very attractive under the computational viewpoint. This relation is efficient because it needs only to discretize the image space road widths within intervals that contain the optimal road widths. However, a rigorous computation of the object space road widths from the corresponding ones optimised in the image space would require the rigorous projection of road width segments onto the DTM, which is a well-known procedure.

Equations 9 and 10 are the bases for road extraction in the object space using a single aerial image. As commented before, these equations can be modified for other sensors. In this case, it is necessary to adapt equation 5 in accordance with sensor geometry. The solution of equation 9 or 10 by the DP optimisation can be compared to a mono-plotting process proposed by Makarovic (1973). This approach is based on two basic steps: 1- feature digitalisation in the image space; and 2- feature projection onto the DTM by the so-called inverse collinearity equations. In the proposed approach the road features are directly extracted in the object space. As illustrated in figure 1, the road is tracked on the DTM while useful photometric information are searched along roads in the image space. While in Makarovic's approach two well-defined mapping steps (i.e., feature extraction from an image and feature projection onto DTM) can be identified, in our approach both steps are accomplished simultaneously.

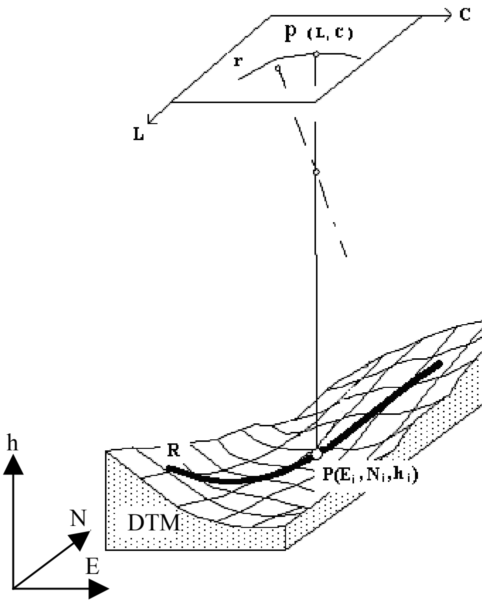


Figure 1. Principle for road extraction in object space

2.3 Strategy for DP optimisation

When in an optimisation problem the variables of the merit function are not interrelated simultaneously, the efficient manner for solving the problem is by applying the DP technique (Ballard and Brown, 1982). Equations 9 and 10 show a structure that meets that requirement, since they have respectively six and nine variables interrelated simultaneously. Below we present only the general strategy developed for extracting 3D polygons representing road centrelines in the object space using the DP algorithm. Mathematical foundations and algorithm aspects can be found in an extensive literature, as e.g. in Ballard and Brown (1982).

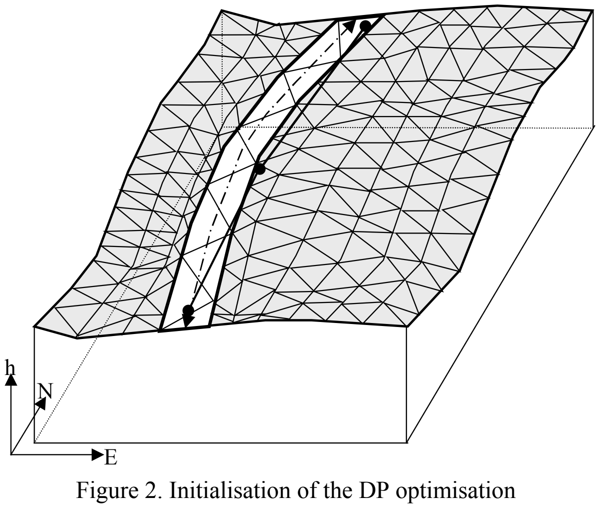


Figure 2. Initialisation of the DP optimisation

No matter what the strategy is used to initialise the proposed methodology, the necessary information are seed points in the reference system where roads will be extracted. Figure 2 shows an illustrative example where three seed points in E, N, and h co-ordinates are used to coarsely describe a road centreline segment. The terrain surface is modelled by a polyhedral model obtained from a regular or irregular mesh. The seed points are collected on or transformed onto the polyhedron, just because the roads will be traced on it.

Figure 3 shows the principle of DP optimisation. Basically, it starts with an initial 3D polygon defined by the user-supplied seed points (black dots in figure 3), which is progressively made dense and refined. In order to accomplish the first iteration, new equidistant vertices are linearly interpolated between every adjacent seed points. In illustrative example of figure 3 two new vertices (marked as circumference) are added. The resulting initial polygon is the reference for generating a search space composed by candidate polygons to accurately represent the road centreline. During the DP optimisation, every vertex may move around its initial position, generating a finite number of polygons. If m is the number of positions each vertex can take and n is the number of vertices defining each polygon, there will be m^n polygons in search space. This means that m should be as small as possible to avoid a prohibitive search space. In order to meet this requirement, the candidates for each best vertex are searched on the polyhedron in direction perpendicular to the actual polygon at each vertex. At the beginning of the optimisation process (i.e., at first iteration) the actual polygon is the initial one. For other iterations, the actual polygon is the one optimised at the last iteration. Figure 3 shows how the search windows are defined at the actual polygon vertices. Each search window is defined as the intersection between the polyhedron and the vertical plane that is perpendicular to the actual polygon at a given vertex. The intersection between each vertical plane and the plane h= 0 is a straight line segment, along which N and E co-ordinates are mathematically interrelated according to equation N= a.E + b, where a and b are the angular and linear coefficient of the straight line, respectively. Now remember that the constraint needed to remove the ambiguity of equation 8 takes the form h= f(E, N) and that the adopted terrain surface model is a polyhedron, the h co-ordinate can be mathematically expressed only in function of E co-ordinate, i.e.,

$$h= A.E + B.(a.E + b) + C \qquad (11)$$

where, A, B, and C are plane coefficients of a polyhedron face.

Equation 11 shows that the search windows' points can be sampled by only sampling their E co-ordinate. Two other components are computed internally. In our scheme E co-ordinates are sampled on the plane h= 0 in a such way that the resulting points (E, N= a.E+b) are equally spaced. If the selected distance between sampled points (E, N) is d, E co-ordinates must be sampled such that $|E_{i+1}-E_i|= d/\sqrt{a^2+1}$ or $|E_{i+1}-E_i|$= d for vertical straight lines. Corresponding points (E, N, h) in the search window are not equally spaced due to the varying slope. The advantage of using this strategy is the elimination of the variables N and h, remaining only 3 simultaneous variable (E_{i-1}, E_i, E_{i+1}) in equation 9 or 6 (E_{i-1}, E_i, E_{i+1}, W_{i-1}, W_i, W_{i+1}) in equation 10. The drawback is that it does not work for horizontal road segments. However, a small rotation of the search window showed to be efficient to overcome that problem. The value of d is directly related to the resolution, size, and number of elements of the search window. Larger pull-in-range can be obtained by using low-resolution (larger d) and large-sized search windows at first iterations of the optimisation process. An iteration consists of adding new interpolated points to the polygon optimised at the last iteration and of applying to the resulting polygon a new DP optimisation. At last iterations high-resolution and small-sized search windows are used. This strategy allows the computational effort to be reduced properly. During each iteration, two curvature constraints are used to additionally reduce the computational effort. They consist in checking if the horizontal and vertical angles at each polygon vertices are below a given threshold. In order words, only smooth polygons are evaluated by the DP optimisation algorithm. Figure 3 shows that after first iteration the initial polygon is geometrically refined, but it does not accurately described the road centreline. Final result is obtained after checking the convergence of the optimisation process and it is expected to accurately represent the road centreline. Convergence checking consists in verifying after each iteration if all added points are collinear to neighbour points. When this condition is verified, the optimisation process is stopped.

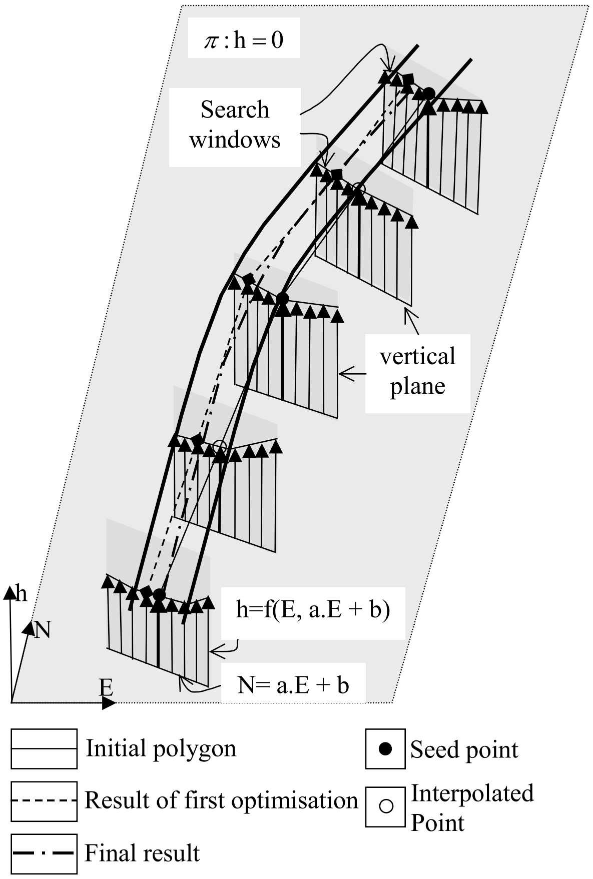


Figure 3. Optimisation strategy

3. RESULTS

The proposed methodology was implemented using Borland C++ Builder 5 compiler for Windows XP. An image (9286 x 9496 pixels) at the approximate scale of 1:9200 is used in experiments (figure 4). This is a high-resolution image, since the pixel footprint is about 26 cm. This image is from a region of Switzerland and is available in the LPS (Leica Photogrammetry Suite®) system, along with the interior and exterior orientation parameters. DTM used in our experiments has a resolution of 5 m.

In order to initialise the road extraction methodology, a few seed points are measured along roads on the image and backprojected onto DTM. Resulting points are seed points needed to start the extraction of each road. Extracted roads in 3D by the DP algorithm are projected into the image reference system (L, C) and overlaid on the input image. This allows a visual analysis of the geometric quality of the extracted road centrelines. Since the methodology depends on some information measured on the image, it is also possible to analyse its robustness against irregularities in the image content, like road obstructions from trees and shadows. Due to the high resolution of the test image, we present below only three selected windows (figures 5, 6, and 7) of it.

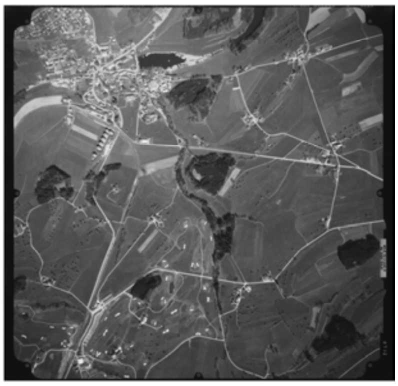


Figure 4. Aerial image used in the experiments

Figure 5 shows the first window where segments of a main road and secondary roads are presented. The results show that the extracted polygons are accurately positioned along the road axes. This is observed even at the road crossings along the main road, where some deficiencies could be expected. An important factor to extract accurate road centrelines is the existence of well-defined road edges, because it allows the edge constraint to be very effective. Places like road crossings, where one or both edges are not presented, can be extracted without deficiencies due to the global curvature constraint enforced in the merit function used in the DP optimisation. This geometric constraint imposes that the accumulate curvature along a road is minimum, implying smoothness to the extracted road centreline. As a result, a short segment of a road affected by some small anomalies (e.g., missing edges at road crossing or obstructions) tend to be represented after extraction by a road centreline segment that has similar curvature to adjacent ones. Although the secondary roads do not have a good contrast with adjacent regions, the road centrelines are accurately extracted.

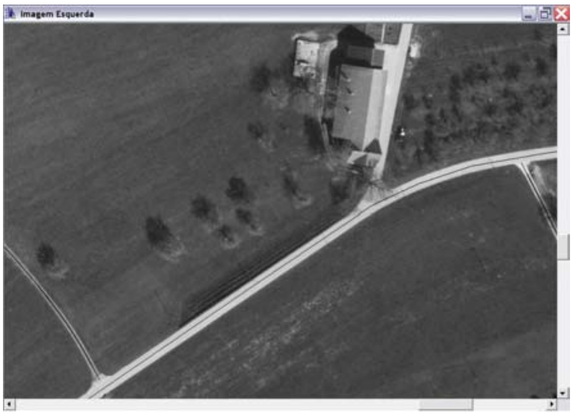


Figure 5. Results in the first window

Figure 6 shows the results in the second window selected in the original image. This example shows two concurrent segments of road, one being straight and another being curved. Both roads have good contrast with adjacent regions and well-defined edges. As in previous example, no significative displacement of road centrelines is observed along the road crossing region. Since for this type of road crossing both edges of each road are missed, the edge constraint almost vanishes. The effect can be better observed on the curved road, where the road centreline segment going through the road crossing region is straight. In general, both road centrelines are accurate, but they show a slight displacement in some parts of road centrelines.

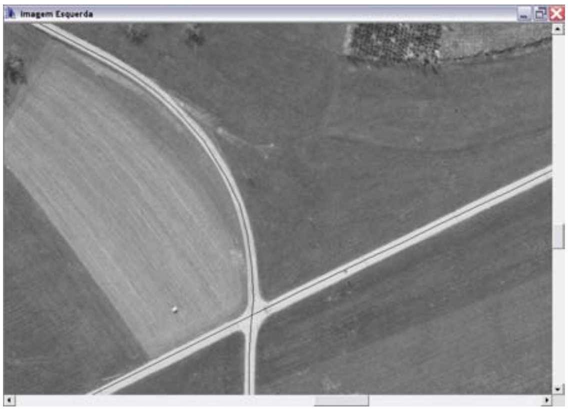


Figure 6. Results in the second window

Figure 7 shows the third window where a curly segment of a road is presented. Even though the contrast of the road with backgrounds is good, some irregularities are visible along and close to the left road curve. The perspective obstruction from the house is very small, but it perturbed a little the extraction process. It is clearly observed that the road centreline changes direction slightly where the house is located. Consequently, along this road segment the extracted road centreline forms a corner and is closer to a road edge. The perspective obstruction caused by trees seems to be not critical to the extraction process. It is also possible to observe the tendency of the road centreline in approaching the internal road edge along road curves. This effect has been commonly observed in other tests. The possible cause is the global curvature constraint enforced in the merit function, by which the accumulate curvature along whole road centreline is minimum. In other words, that global constraint tends to slightly dominate local constraints (e.g., edge constraint). However, irregularities along road curves can disturb the result.

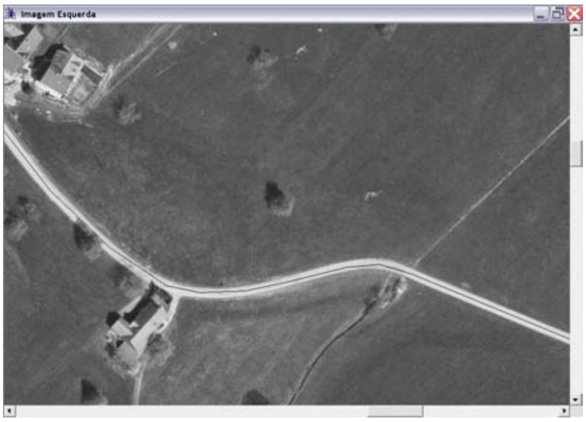


Figure 7. Results in the third window

In order to access the accuracy of the proposed methodology, road centrelines were manually extracted and compared to corresponding ones extracted by the road extraction algorithm. The node positions of road centrelines were determined to be about 1.3 m in average from the manually extracted road centrelines. This accuracy corresponds to approximately one-sixth of the main roads' mean width.

4. CONCLUSIONS

In this paper was proposed an object space road extraction methodology from a single image. It allowed the integration of two basic steps of data capturing for GIS system, i.e., the road extraction in the image space and the transformation of road features into a map projection. Different resolution aerial images can be handled by the proposed methodology. Terrain information in form of a polyhedron is also necessary to allow the solution of the extraction problem. In order to initialise the extraction procedure, a few seed points is necessary to be supplied on the polyhedron. We identify these points on the image and project them onto polyhedron.

In order to evaluate the methodology one experiment was carried out using a high-resolution aerial image. The results obtained were projected into image space and overlaid on the input image. Three image windows are selected in the input image to analyse the performance of the methodology. In general, the methodology proved to be robust, since it handled irregularities like obstructions. The accuracy of extracted road centrelines is good, although some slight displacements are observed.

Our future works on this subject may include the improvement of the proposed methodology, the development of the multiple image mode, and the development of new application. A possible improvement of the proposed methodology can be accomplished in the merit function. For example, a property not modelled in the merit function (equation 9 or 10) is the smoothness of the road centreline profile in the object space. It is also well-known that asphalt material has very low radiometric responses for laser scanner sensor, property that could be also modelled in the merit function. Consequently, these modifications would allow the integration of laser scanner data with image data for road extraction using DP optimisation. This integration could be carried out both in single image and multiple image modes. An interesting application of the single image methodology is in the refinement of pre-existing road database at smaller scale. In this case, the methodology can be initialised automatically by using pre-existing 3D roads as the first approximation for the DP optimisation.

ACKNOWLEDGES

This work was supported by FAPESP (Research Foundation of the State of São Paulo, Brazil) grant number 01/01168-5 and CAPES (Brazilian Foundation for Support of Graduate People).

REFERENCES

Agouris, P., Gyftakis, S., Stefanidis, A., 2000. Uncertainty in image-based change detection (snakes). In: *Accuracy 2000*, Amsterdam, pp. 1-8.

Bajcsy, R., Tavakoli, M., 1976. Computer Recognition of Roads from Satellite Pictures. *IEEE Transaction on System, Man and Cybernetic*, 6(9), pp. 76-84.

Ballard, D. H., Brown, C. M., 1982. *Computer Vision*. Englewood Cliffs, New Jersey: Prentice Hall, 523p.

Baumgartner, A., Steger, C. Mayer, H., Eckstein, W., Ebner, H., 1999. Automatic Road Extraction Based on Multi-Scale, Grouping, and Context. *Photogrammetric Engineering and Remote Sensing*, 66(7), pp. 777-785.

Dal Poz, A. P., Silva, M. A. O., 2002. Active Testing and Edge Analysis for Road Centreline Extraction. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Graz, Vol. 34, pp. 44-47.

Dal Poz, A. P., Vale, G. M., 2003. Dynamic programming approach for semi-automated road extraction from medium- and high- resolution images. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Munich, Vol. 34, pp. 87-91, 2003.

Gruen, A., Li, H., 1997. Semi-automatic linear feature extraction by dynamic programming and LSB-snakes. *Photogrammetric Engineering and Remote Sensing*, 63(8), pp. 985-995.

HU, X.; ZHANG, Z.; TAO, C. V. A Robust Method for Semi-Automatic Extraction of Road Centerlines Using a Piecewise Parabolic Model and Least Square Template Matching. Photogrammetric Engineering and Remote Sensing, v. 70, n. 12, 2004, pp. 1393-1398.

Kass, M., Witkin, A. Terzopoulos, D., 1987. Snakes: Active contour models. In: *First International Conference on Computer Vision*, London, UK, pp. 259-268.

Kim, T., Park, S. R., Kim, M. G., Jeong, S., Kim, K. O., 2004. Tracking Road Centerlines from High Resolution Remote Sensing Images by Least Squares Correlation Matching. *Photogrammetric Engineering and Remote Sensing*, 70(12), pp. 1417-1422.

Makarovic, B., 1973. Digital mono-plotters. *ITC Journal*, pp. 583-599.

McKeown, D. M., Denlinger, J. L., 1988. Cooperative methods for road tracking in aerial imagery. In: *Workshop of Computer Vision and Pattern Recognition*, pp. 662-672.

Merlet, N., Zerubia, J., 1996. New prospects in line detection by dynamic programming. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 18(4), pp. 426-431.

Quam, L. H., 1978. Road Tracking and Anomaly Detection in Aerial Imagery. In: *Image Understanding Workshop*, London, pp. 51-55.

Stoica, R., Descombes, X., Zerubia, J., 2004. *A* Gibbs Point Process for Road Extraction from Remotely Sensed Images. *International Journal of Computer Vision*, 57(2), pp. 121-136.

Vosselman, G., de Knecht, J., 1995. Road tracing by profile matching and Kalman filtering. In: *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Birkhaeuser Verlag, pp. 265-274.

Zhu, P., Lu, Z., Chen, X., Honda, K., Eiumnoh, A., 2004. Extraction of City Roads Through Shadow Path Reconstruction Using Laser Data. *Photogrammetric Engineering and Remote Sensing*, 70(12), pp. 1433-1440.

AUTOMATIC VEHICLE DETECTION IN SATELLITE IMAGES

J. Leitloff [1], S. Hinz[2], U. Stilla[1]

[1] Photogrammetry and Remote Sensing, [2] Remote Sensing Technology
Technische Universitaet Muenchen, Arcisstrasse 21, 80333 Muenchen, Germany
{jens.leitloff, stefan.hinz, stilla}@bv.tum.de

KEY WORDS: Urban, Feature extraction, Edge detection, Optical Satellite Imagery, Quickbird

ABSTRACT:
Vehicle detection is motivated by different fields of application, e.g. traffic flow management, road planning or estimation of air and noise pollution. Therefore, an algorithm that automatically detects and counts vehicles in air- or space-borne images would effectively support these traffic-related analyses in urban planning. Due to the small vehicle size in satellite images detection of single vehicles would deliver ambiguous results. Hence, our scheme focuses primarily on the extraction of vehicle queues, as the pattern of a queue makes it better distinguishable (as a whole) from similar objects. Hypotheses for queues are generated by sophisticated extraction of ribbons. Within these ribbons single vehicles are searched for by least-squares fitting of Gaussian kernels to the width and contrast function of a ribbon. Based on the resulting parameter values, false and correct hypotheses are discerned. The results show that the analysis of width and contrast information using least square optimization is able to extract single vehicles from queues with high correctness. Still, the completeness of the overall extraction is relatively low, since only queues can be extracted but no isolated vehicles. The results clearly show that the approach is promising but further improvements are necessary to achieve a higher completeness.

1. INTRODUCTION

1.1 Motivation

There is an increasing demand for traffic monitoring of densely populated areas. The traffic flow on main roads can partially be measured by fixed installed sensors like induction loops, bridge sensors and stationary cameras. Traffic on smaller roads – which represent the main part of urban road networks – is scarcely monitored and information about on-road parked vehicles is not collected. Wide-area images of the entire road network can complement these selectively acquired data. New optical sensor systems on satellites, which provide images of 1-meter resolution or better, e.g. Ikonos and QuickBird, make this kind of imagery available. Hence new applications like traffic monitoring and vehicle detection from these images have achieved considerable attention on international conferences, e.g. (Bamler and Chiu, 2005; Heipke et al., 2005; Stilla et al., 2005). The presented approach focuses on the detection of single vehicles by extracting of vehicle queues from satellite imagery.

1.2 Related work

Depending on the used sensors and the resolution of the imagery different approaches (Stilla et al., 2004) have been developed in the past. The extraction of vehicles from images with a resolution of about 0.15 m has already been comprehensively tested and delivers good results in many situations. Available approaches either use implicit or explicit vehicle models (Hinz, 2003). The appearance-based, implicit model uses example images of vehicles to derive gray-value or texture features and their statistics, which are assembled in vectors. These vectors are used as reference to test computed feature vectors from image regions. Since the implicit model classification uses example images the extraction results depend strongly on the choice of representative images.

Approaches using an explicit model describe vehicles in 2 or 3 dimensions by filter or wire-frame representations. The model is then either matched "top-down" to the image or extracted image features are grouped "bottom-up" to create structures similar to the model. A vehicle will be declared as detected, whenever there is sufficient support of the model found in the image. These approaches deliver comparable or even better results than approaches using implicit models but are hardly applicable to satellite imagery since there vehicles only appear as blobs without any prominent sub-structures (see Fig. 1).

Three different methods for vehicle detection from simulated satellite imagery of highway scenes are tested in (Sharma, 2002). The gradient based method and the method using Bayesian Background Transformation (BBT) deliver the best number of vehicle counts compared to ground truth. Since the number of false detections is lower using BBT, this method is more reliable. The performance of the third method using Principal Component Analysis (PCA) varies significantly with the noise level of the image. Furthermore, the method gives the lowest vehicle count. A manually created background image is mandatory for the PCA and BBT method, which requires extensive interactive work. Consequently, the approaches can hardly be generalized and are limited to images of the same scene.

In (Gerhardinger et al., 2005) the commercial software *Features Analyst®* is used to implement an iterative learning approach by analyzing the spectral signature and the spatial context. The authors report that good results can be achieved if a very

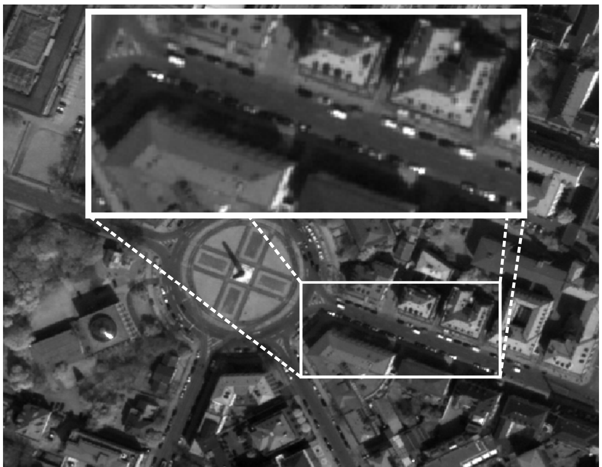


Figure 1: Appearance of vehicles in optical high-resolution satellite imagery (Quickbird), GSD = 0.6 m

accurate road GIS is available. However, this had to be derived by manual digitalization in their case.

An encouraging approach for single vehicle detection is presented in (Jin and Davis, 2004). First, they use morphological filtering for a rough distinction between vehicle pixels and non-target pixels, which are similar to vehicles. Then a morphological shared-weight neural network is used for extraction. The approach achieves good performance values under the condition that vehicles appear isolated. However, the approach is not designed for vehicle queues or traffic jams (Jin and Davis, 2004).

The last mentioned approaches are designed for a resolution coarser than 0.5 m and limit their search space to roads and parking lots using GIS information. By this, the number of false alarms is significantly decreased.

In dense traffic situations, traffic jams or parking lots, car groupings show quite evident regularities (see e.g. Fig. 1). Exploiting the knowledge about these repeating occurrences and the fact that cars rarely occur isolated is also referred to as global modeling in the filed of vehicle detection. Vehicle hypotheses extracted by a neural network classifier (Ruskoné et al., 1996) or a "spot detector" (Michaelsen & Stilla, 2001) are collinearly grouped into queues while isolated vehicle hypotheses are rejected. Hinz & Stilla (2006) use a differential geometric blob detector for an initial extraction of car candidates followed by a modified Hough transform for accumulating global evidence for car hypotheses. Since these grouping schemes select hypotheses but do not add new hypotheses, these approaches need an over-segmentation as initial input. They are designed for medium resolution images of approximately 0.5m ground sampling distance (GSD).

When high resolution imagery is available a more promising strategy is to focus on reliable hypotheses for single vehicles first and complete the extraction afterwards by searching for missing vehicles in gaps of a queue using a less constrained vehicle model (Hinz, 2003). By this, not only queues but also isolated cars can be extracted as long as they belong to the set of reliable hypotheses.

One of the few approaches focusing directly on vehicle queues – in particular military convoys – is presented in Burlina et al. (1997). They extract repetitive, regular object configurations based on their spectral signature. In their approach the search space is limited to roads and parking lots using accurate GIS-information. This seems necessary since the spectrum will be heavily distorted, if adjacent objects gain much in influence – even if the spectrum is computed for quite small images patches.

1.3 Overview

Figure 3 shows the overall structure of our approach which is separated into three processing stages. In the pre-processing step (Fig. 2 I), GIS data is used to determine Regions of Interest (ROI). Afterwards we use a differential geometric approach followed by some post-processing to extract linear features as hypotheses of the queues (Sect. 2.2 and 2.3; Fig. 2 II). Finally, we determine single vehicles from these hypotheses by analyzing the width and contrast function using a least squares optimization (Sect.2.4; Fig. 2 III).

2. QUEUE DETECTION

In Sect. 2.1 the used model will be presented. Sect. 2.2 describes the extraction of vehicle queues using sophisticated line extraction. Then a number of attributes are calculated (Sect. 2.3). The attributes are analyzed and checked for consistency to verify or falsify single vehicle hypotheses. This is done by a least square adjustment (discussed in Sect. 2.4) and by an iterative constrained search (see Sect. 2.5).

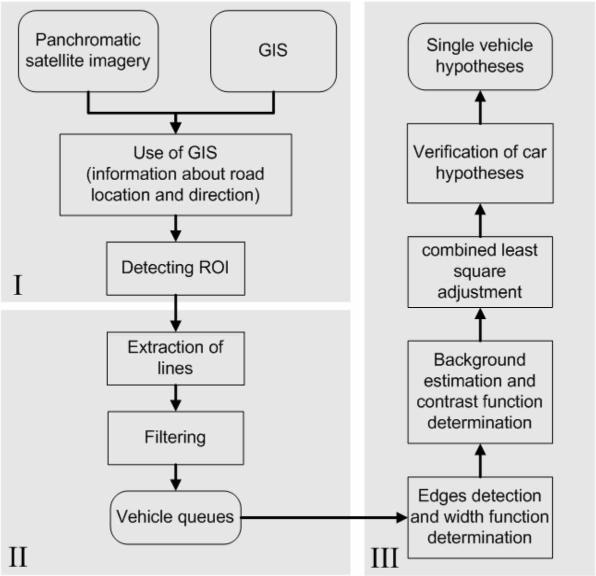

Panchromatic satellite imagery
GIS
Use of GIS (information about road location and direction)
Detecting ROI
I
Extraction of lines
Filtering
Vehicle queues
II
Edges detection and width function determination
Background estimation and contrast function determination
combined least square adjustment
Verification of car hypotheses
Single vehicle hypotheses
III

Figure 2: Processing Scheme

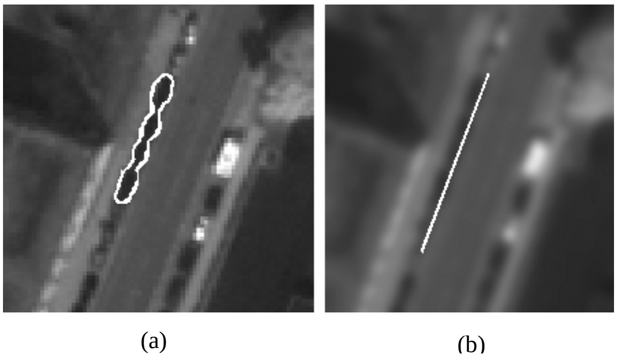
2.1 Model of vehicle queues

Generally, a vehicle queue is defined as ribbon with distinct symmetries along and across its local orientation. Basically, the model is similar to that defined in (Hinz, 2003); though, since this model is originally designed for aerial images, a number of modifications regarding the significance of different features have been applied:

A vehicle queue

- must have sufficient length, bounded width and low curvature;
- shows a repetitive pattern along the centerline, both in contrast and width (Fig. 3a), while length and width of the individual replica correspond to vehicle dimensions;
- collapses to a line in Gaussian scale space, i.e. when smoothing the image accordingly (Fig. 3b).

Please note that this queue model differs from the above mentioned approaches in a way that – in particular through the scale-space description – the queue is modeled as a unique structure and not just as a composite of its underlying, smaller elements. At first glance, this seems of less importance. Still, it provides the basis for detecting a queue hypothesis as a whole



(a) (b)

Figure 3: Queue model. a) original image, b) smoothed image

(even though at a coarser scale) rather than constructing it from smaller elements. Thereby global knowledge can be incorporated from the very beginning of the extraction.

2.2 Extraction of vehicle queues

Figure 4 illustrates the a priori information about road location and direction taken from a national core database. The positional accuracy is known to be approximately 2m. Neither the road sides are contained in the database nor the position of the individual lanes. Hence, the road width needs to be estimated from attributes like the number of lanes or the average width per road segment. Thus the generated regions of interest (ROI) can only be regarded as an approximation of the true road area.

Line extraction is carried out by applying the differential geometric approach of Steger (1998). This algorithm is primarily based on the computation of the second image derivatives, i.e. the local curvatures of the image function. Parameters for the line extraction are chosen corresponding to the vehicle geometry (vehicle width: w) and radiometry (expected contrast to road: c).

Thus, the necessary input parameters for line extraction σ,.t_L and t_H can be calculated as follows:

$$\sigma = \frac{w}{2\sqrt{3}} \qquad t = c\frac{-w}{\sqrt{2\pi}\cdot\sigma^3}e^{-\frac{1}{2}\left(\frac{w}{2\sigma}\right)^2} = c\frac{-\sqrt{6}\cdot 12}{\sqrt{\pi}\cdot w^2}e^{-\frac{3}{2}} = c\cdot a$$

$$t_L = c_L \cdot a \qquad t_H = c_H \cdot a$$

where σ defines the preliminary smoothing factor, calculated from the maximum expected width (e.g. 2.5 meter). t_L and t_H define the hysteresis thresholds for the second partial derivative of the image at each point. If the value exceeds t_H a point is immediately accepted as line point. All points where the second derivative is smaller than t_L are rejected. Points with a second derivative between t_H and t_L are accepted if they can be connected to already accept points. In order to achieve initial hypotheses, the parameters for c_L (minimum contrast to be accepted) and c_H (contrast for queues definitely to accept) are chosen quite relaxed.

Additionally, the line extraction algorithm is supported by morphologically filtering the image with a directional rectangular structuring element oriented along the particular road segment. In doing so the queues are enhanced and disturbing substructures in bright cars are almost completely removed. The relaxed parameter settings lead to a huge number of false hypotheses but also return most of the promising hypotheses for vehicle queues. However, since the line extraction requires a minimum contrast between vehicles and the road surface, gray vehicles can not be extracted reliably, as they hardly emerge from their surroundings.

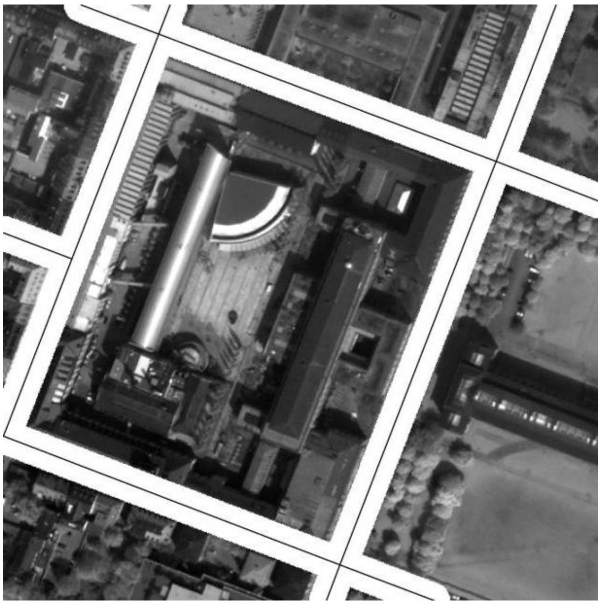


Figure 4: Regions of Interest

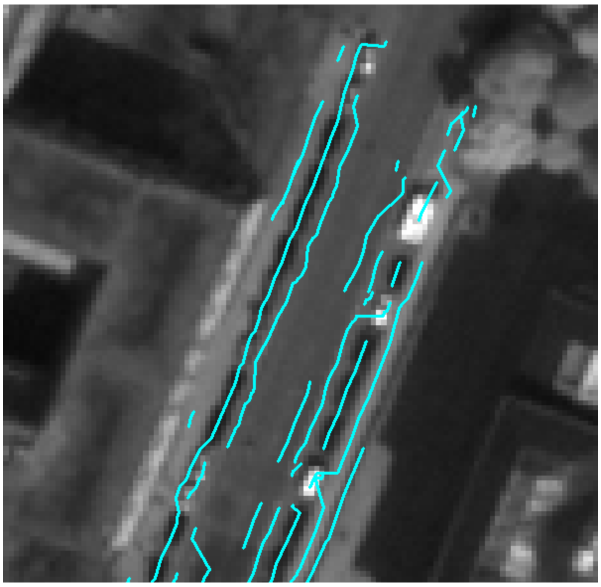


Figure 5: Resulting lines after merging, smoothing and filtering.

Bright and dark lines are extracted separately. They are connected if they fulfill some distance and collinearity criteria. In our case a maximum distance of one vehicle length must not be exceeded. Additionally, one has to keep in mind that the merging of parallel lines would lead to significant positional errors and is therefore prevented. The final processing steps consist of geometrical smoothing by polygonal approximation, resampling (Ramer, 1972) and testing all resulting lines against a minimum length threshold and an upper limit for direction differences to the road. Results of the merging and filtering steps are illustrated in Fig. 5.

2.3 Determining queue width and contrast

After extracting lines as medial axes of a ribbon, width and contrast functions are determined. The algorithm to find the ribbon width in each line point is based on profiles spanned perpendicular to the local line direction, and determining each profile’s gray values by bilinear interpolation. Then, for each profile, local maxima are determined with sub-pixel precision by fitting a second-order polynomial to the first derivative of the gray value profile in each profile point. The first maximum value found on either side of centerline is supposed to correspond to the vehicle boundary, i.e., the distance between the two maxima yields the queue width. If no maximum is found, gaps in the width function are closed afterwards by linear inter- or extrapolation.

Results of width determination are illustrated in Fig. 6. It can be seen that most edges correspond to vehicle sides. Because of weak contrast between vehicles and road surface a number of outliers are present, which are to remove by median filtering the width function.

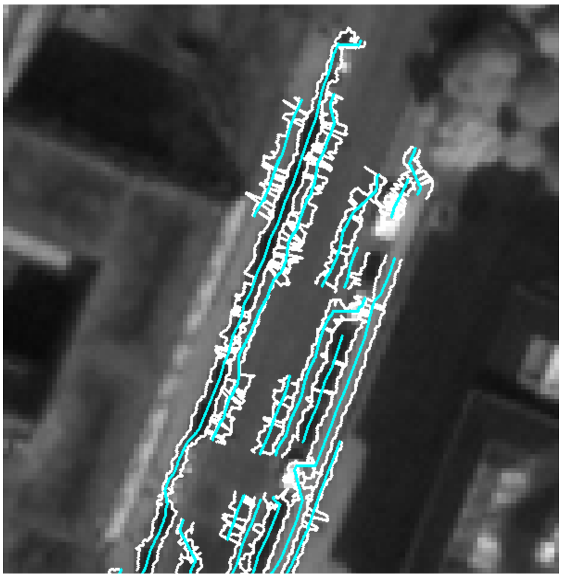


Figure 6: Extracted ribbons: medial axis (cyan) and width function (white).

One can see that most edges correspond to vehicle sides. However, since the gradient image has quite weak contrast, the edges extraction results show also some irregularities, i.e. noisy boundaries. Therefore smoothing of the extracted edges is useful to reduce the number of outliers.

Usually the irregularities are caused by other edges nearby the vehicle queue. In future implementations we intend to detect such outliers by a more sophisticated shape analysis of the boundary functions.

To determine the contrast function of a ribbon, a reference gray value outside the vehicle regions must be defined. The actual gray values in the direct neighborhood of a vehicle, however, are often influenced by adjacent objects or shadows and are therefore no reliable estimates of the reference gray value. A better way to determine the contrast function is to estimate the median road surface brightness in the neighborhood of a vehicle queue and use this estimate as reference gray value. Assuming that – despite of the presence of some vehicles – the most frequent gray values in the RoI represent the road surface, and further assuming that in the center of a road less disturbances by vehicles and shadows occurs than at the road sides, the following simple procedure has been implemented to compute the road surface brightness:

- project the start and end point of each extracted centerline onto the GIS road axis, thereby defining the relevant road section
- dilate this section by approximately the width of one lane
- calculate the median gray value of this image region in order to estimate the road surface brightness

Since the gray values along the medial axes have already been extracted, the contrast function simply results from the absolute difference of these values and the reference gray value. In Fig. 7 examples of width and contrast function of a ribbon are shown. It furthermore illustrates that both functions show mutually correlated repetitive patterns which will be used to detect single vehicles.

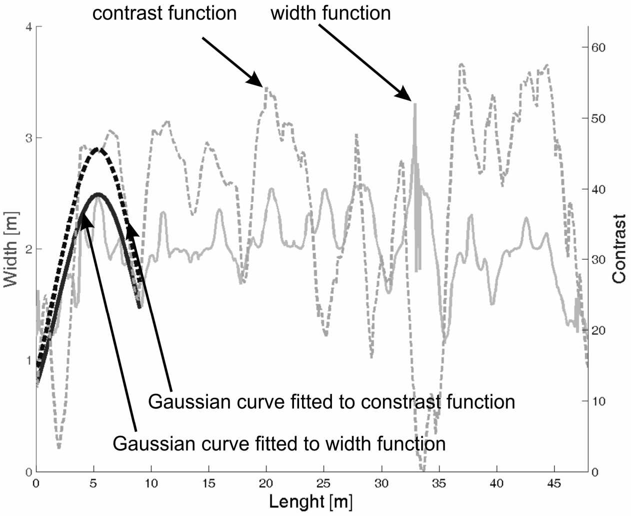


Figure 7: Width and contrast function of a ribbon

2.4 Single vehicle determination by least squares optimization

For extraction of single vehicles from a ribbon, Gaussian kernels are fitted to the width and contrast function (Fig. 7). Of course, different kernels like a second-order polynomial could be used instead. However, the estimated parameters of a fitted Gaussian kernel relate not only to the desired vehicle dimensions but also allow to establishing a link to the particular scale used for line extraction in Sect. 2.2. – especially the Gaussian kernels fitted to the contrast function. The rationale of the procedure outlined in the following should thus be understood as an attempt to embed the vehicle detection into the same scale-space framework as the line extraction approach.

The calculation of the unknown parameters of each Gaussian kernel is done by a least squares fit. The notation corresponds to the work of Mikhail (1976).

The functional model of a Gaussian function to fit to a predefined interval of the width functions has the following form:

$$w\left(a_w,\sigma_w,\mu\right)=\frac{a_w}{\sqrt{2\pi}\sigma_w}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma_w}\right)^2} \quad (1)$$

with *w(...)* width as function of a_w, σ_w and μ
a_w the amplitude of the fitted Gaussian kernel
σ_w second-order moment of the Gaussian kernel
μ first-order moment of the Gaussian kernel, i.e. the position of maximum amplitude
x position of *w* along the interval under investigation

The functional model for the contrast function is quite similar:

$$c\left(a_c,\sigma_c,\mu\right)=\frac{a_c}{\sqrt{2\pi}\sigma_c}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma_c}\right)^2} \quad (2)$$

with *c(...)* contrast as function of a_c, σ_c and μ
a_c the amplitude of the fitted Gaussian curve
σ_c second-order moment of the Gaussian curve
μ first-order moment of the Gaussian kernel, i.e. the position of maximum amplitude

x position of value c along the interval under investigation

Since a vehicle should yield the maximum of both width and contrast function at the same position, μ is a shared parameter in both functions. Fig. 7 illustrates an example of the contrast and the width signal. The fitted Gaussian curves for the first interval are also included. These intervals are defined by two consecutive minima in a smoothed version of the function. It is also apparent from Fig. 7 that additionally introducing σ as shared parameter would not lead to satisfactory results. The pronounced differences between the shapes of the two functions would cause the accuracy of the estimated unknown σ to drop down significantly.
The unknown parameters of the functional model (1) and (2) are summarized in the vector $\boldsymbol{x}$:

$$\boldsymbol{x}^{\boldsymbol{T}} = \left(a_w \quad \sigma_w \quad \mu \quad a_c \quad \sigma_c \right)$$

It is easy to see that the functions (1) and (2) are nonlinear. Therefore the determination of the unknown parameters is an iterative process, where $\boldsymbol{x}$ needs to be calculated by (see (Mikhail, 1976):

$$\boldsymbol{x} = \boldsymbol{x}^{\boldsymbol{0}} + \boldsymbol{\Delta} \qquad (3)$$

and

$$\boldsymbol{\Delta} = \left(\boldsymbol{B}^T \boldsymbol{B} \right)^{-1} \boldsymbol{B}^T \left(\boldsymbol{l} - \boldsymbol{f}\left(\boldsymbol{x}^{\boldsymbol{0}} \right) \right)$$

assuming the observations to be uncorrelated and of equal accuracy, i.e. neglecting the weighting matrix $\boldsymbol{W}$. Vector $\boldsymbol{l}$ contains the observations of the current interval and $\boldsymbol{f}(\boldsymbol{x}^{\boldsymbol{0}})$ the width and contrast function derived from the initial values $\boldsymbol{x}^{\boldsymbol{0}}$. $\boldsymbol{\Delta}$ are the corrections to the initial values and $\boldsymbol{B}$ is the Jacobian matrix containing the partial derivatives with respect to the unknowns of the Gaussian kernels.

The vectors $\boldsymbol{l}$ and $\boldsymbol{f}(\boldsymbol{x}^{\boldsymbol{0}})$ are defined by:

$$\boldsymbol{l}^{\boldsymbol{T}} = \left(w_f \cdots w_l \; c_f \cdots c_l \right)$$

$$\boldsymbol{f}(\boldsymbol{x}^{\boldsymbol{0}})^{\boldsymbol{T}} = \left(w\left(a_w^0, \sigma_w^0, \mu\right)_f \cdots w\left(a_w^0, \sigma_w^0, \mu\right)_l \; c\left(a_c^0, \sigma_c^0, \mu\right)_f \cdots c\left(a_c^0, \sigma_c^0, \mu\right)_l \right)$$

where indices f (first value) and l (last value) indicate the boundaries of the interval under investigation.
Values for $\boldsymbol{x}^{\boldsymbol{0}}$ are chosen considering that:

- $\mu^{\boldsymbol{0}}$ corresponds to the position of the maximum of the current interval
- $a^{\boldsymbol{0}}{}_c$ is the contrast value at position $\mu^{\boldsymbol{0}}$
- $\sigma^{\boldsymbol{0}}{}_w = \sigma^{\boldsymbol{0}}{}_c$ is chosen according to the supposed vehicle length
- $a^{\boldsymbol{0}}{}_w$ can be calculated by $a_w = w_\mu \cdot \sqrt{2\pi} \cdot \sigma_w$ where w_μ is the width at maximum μ

Now the unknowns $\boldsymbol{x}$ can be calculated according to Equ.3. If the L1-norm $\left\| \boldsymbol{x} - \boldsymbol{x}^{\boldsymbol{0}} \right\|$ is greater than a predefined threshold, $\boldsymbol{x}^{\boldsymbol{0}}$ is replaced by $\boldsymbol{x}$ and $\boldsymbol{\Delta}$ will be calculated again until convergence or after a maximum number of iterations is reached.
Furthermore the accuracy of the unknowns can be obtained from the diagonal of the $\boldsymbol{C}_{\boldsymbol{xx}}$ matrix, which is calculated by:

$$\boldsymbol{C}_{\boldsymbol{xx}} = \hat{\sigma}_0^2 \cdot \boldsymbol{Q}_{\boldsymbol{xx}} = \hat{\sigma}_0^2 \cdot \left(\boldsymbol{B}^T \boldsymbol{B} \right)^{-1}$$

with

$$\hat{\sigma}_0^2 = \frac{\boldsymbol{v}^T \boldsymbol{v}}{n - u}.$$

Here n is the number of observations, u is the number of unknown parameters (here 5) and $\boldsymbol{v}$ contains the observations' residuals, which are calculated by:

$$\boldsymbol{v} = \boldsymbol{B} \cdot \boldsymbol{\Delta} - \left(\boldsymbol{l} - \boldsymbol{f}\left(\boldsymbol{x}^{\boldsymbol{0}} \right) \right)$$

If the width and the contrast functions exhibit the expected repetitive pattern, only a few iterations are necessary.
As final result of the least squares adjustment, we obtain the parameters describing a fitted Gaussian kernel for a given interval including their accuracies. Thresholds are applied to these parameters to discern false and correct hypotheses. The required thresholds were acquired from test datasets.
In some cases multiple detections of the same vehicle occur due to neighboring ribbons. Therefore, an overlap analysis is carried out in which all overlapping (or nearby) hypotheses are mutually tested for consistency. In case of conflicts the worse hypothesis is rejected.

2.5 Single vehicle determination by iterative constrained search

A second method to find single vehicles also uses the appearance of significant repetitive patterns in the width function (Figure 8). Here, maximum values in this function are assumed to approximately represent the centers of single vehicles whereas minimum values are assumed to represent gaps between two vehicles of a queue.

The following parameters are used:

- $\mathrm{v_{min}}$ ... minimum length of a single vehicle (SV) and search interval (SI)
- $\mathrm{v_{max}}$ ... maximum length of SV and SI
- $\mathrm{l_{min}}$ ... position of the minimum width within SI
- $\mathrm{l_{max}}$ ... position of the maximum width within SI
- d ... distance between $\mathrm{l_{min}}$ and $\mathrm{l_{max}}$

A vehicle hypothesis is generated if the following condition is fulfilled: $\frac{\mathrm{v_{min}}}{2} \leq \mathrm{d} \leq \frac{\mathrm{v_{max}}}{2}$

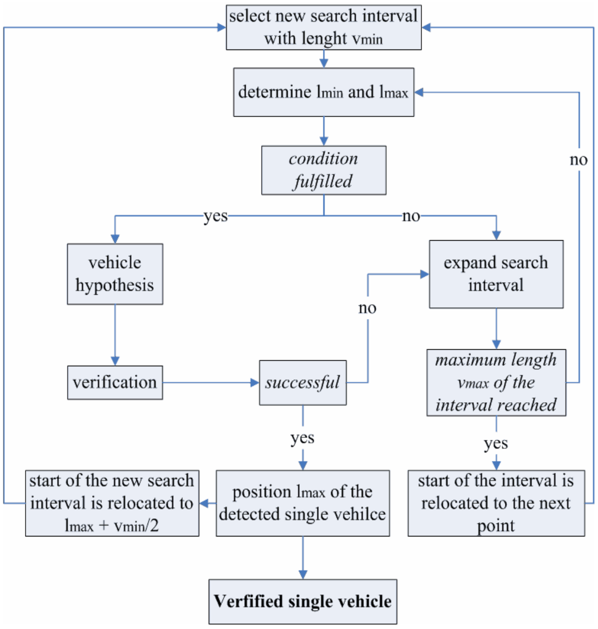

select new search interval with lenght vmin
determine lmin and lmax
condition fulfilled
yes
no
no
vehicle hypothesis
expand search interval
no
verification
successful
maximum length vmax of the interval reached
yes
yes
start of the new search interval is relocated to lmax + vmin/2
position lmax of the detected single vehilce
start of the interval is relocated to the next point
Verfified single vehicle

Figure 8: Concepts of width functions' analysis

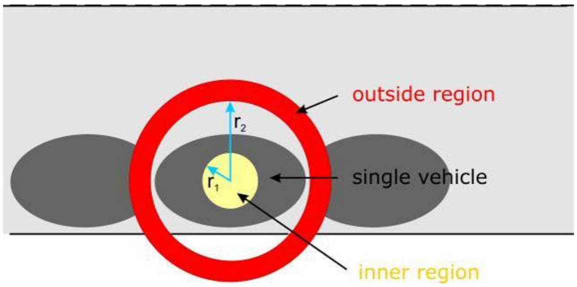


Figure 9: Verification

Figure 8 shows the flow chart of the width analysis scheme. Essentially, this algorithm tries to find local maxima and minima in the noisy width function and place the vehicle positions in such a way that vehicle hypotheses do not overlap.

It is possible that more than one hypothesis is found for a single vehicle. This is caused by two or more maxima in the width function within size of a vehicle. Therefore we control the space between two hypotheses not to fall below a certain minimum distance. If more than one hypothesis is found within this minimum distance, the hypothesis with the highest maxima in the width function will be verified.

Unlike the method described in Sect. 2.4, the contrast function is not used here. Rather, the contrast of the vehicle and the adjacent road surface is used for a simple verification after a hypothesis has been generated. Here the difference of the median gray values of the inner and the outer region is calculated (see Figure 9).

3. RESULTS

In Figures 10 and 11, results achieved with the extraction approach from Sect. 2.4 are shown. Therefore, we processed an image scene covering an area of 0.1 sq. km. Cyan ellipses correspond to correct extractions, white ellipses represent misdetections. As can be seen, the ellipses of the correct extractions coincide quite well with the actual vehicles, clearly indicating that the fitting procedure works reliably. This is a very encouraging result, especially when recalling Figure 7, which gives an impression on the “noisiness” of the contrast and width function.

However, there are also a number of misdetections, in particular at side-walks when dark objects are on either side of the side-walk (see e.g. Fig. 11). Such failures could be overcome, for instance, by a more detailed analysis of neighborhood relations of extracted vehicles. A constellation as achieved for the right queue in Fig. 10 is very unlikely to happen; four vehicles are almost perfectly aligned in a row while each of the other two vehicles is located on a different side of this row. Incorporating this kind of reasoning into the approach would allow to further reducing the misdetection rate.

Fig. 10 and 11 also show that a number of cars are not extracted, i.e. the completeness of this approach is quite fair. However, one has to keep in mind that vehicles do not always appear as queues and, furthermore, that the line extraction does not extract all existing queues. In fact, tests have shown that approximately 60% of all vehicles are contained in the ribbons that serve as initial hypotheses. Besides this, also the edge detection procedure for determining a ribbon’s width could be improved to support convergence of the least squares adjustment.

For numerical evaluation, manually created reference data sets have been utilized and the well-known criteria "correctness" and "completeness" values are calculated as evaluation measures:

$$\text{correctness} = \frac{TP}{TP + FP}$$

$$\text{completeness} = \frac{TP}{TP + FN}$$

with TP true positives
FP false positives
FN false negatives

Here true positives are correctly extracted vehicles, false positives are misdetections, and false negatives are missed vehicles with respect to the reference data. Table 1 summarizes the evaluation results depending on the type of reference data and the used method:

a) all vehicles using least squares adjustment
b) all vehicles using iterative constrained search
c) only bright and dark vehicles, i.e. without gray vehicles (using least squares adjustment)
d) only bright and dark vehicles, i.e. without gray vehicles (using iterative constrained search)

Gray vehicles have been excluded from the reference in b) and d) since they almost show no contrast to their surroundings. We would like to mention in addition that the acquisition of

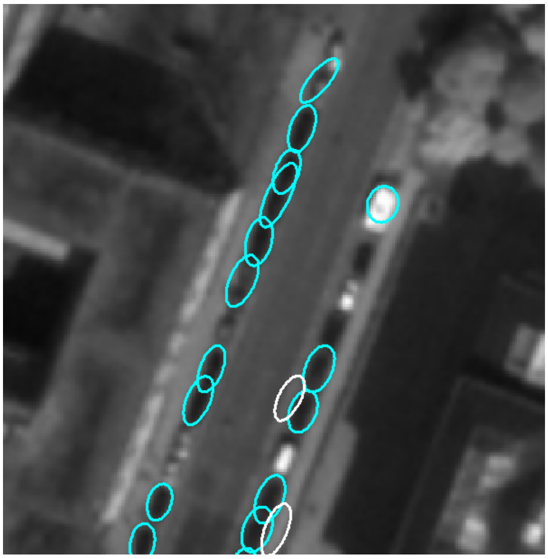


Figure 10: Extraction results I

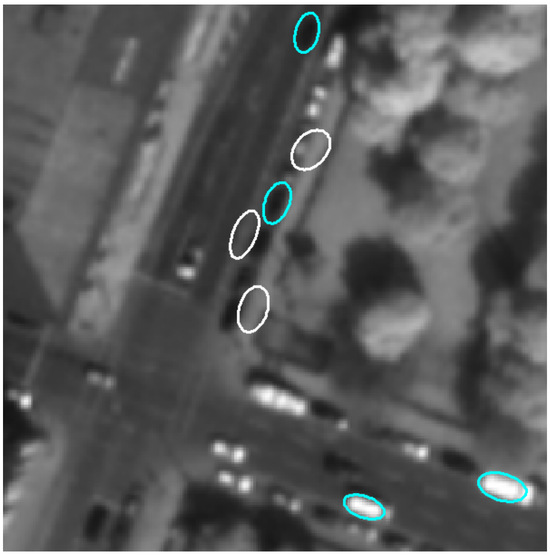


Figure 11: Extraction results II

reference data for some vehicles is certainly not free of errors. Even a human observer is sometimes not able to identify all vehicles in an image scene with high confidence. Therefore our reference data can only be considered as a very good approximation of real “ground-truth”.

It can be seen from Table 1, that on one hand both approaches deliver comparable results, although the iterative constrained search generally achieves higher completeness with better correctness at the same time. On the other hand the single vehicle determination by least square optimization gives statistical accuracy of all hypotheses. It is planned to use these values for internal evaluation, which is supposed to increase performance and reliability.

	Reference data			
	(a)	(b)	(c)	(d)
Completeness [%]	31.1	34.1	36.1	40.3
Correctness [%]	73.5	76.0	70.5	72.3

Table 1: Numerical Evaluation

Despite the weak completeness, the good correctness of the extracted vehicles allows to use them as starting point for searching additional vehicles. Therefore the next steps of implementation will include the search for isolated vehicles using the information from the previous queue detection. Preliminary investigations using a differential blob detector (Hinz, 2005) for accomplishing this task have already taken out.

Concluding the discussion, vehicles with good or even medium contrast to the road surface can be extracted very accurately. Furthermore, the results show that the analysis of width and contrast information using least square optimization allows to extracting single vehicles from queues with high correctness. Still, the completeness of the overall extraction is relatively low, since only queues can be extracted but no isolated vehicles. The results clearly show that the approach is promising but further improvements are necessary to achieve a higher completeness.

REFERENCES

Bamler R., Chiu, S., 2005. Spaceborne Traffic Monitoring from SAR and Optical Data (Jointly Organized with ISPRS WGII/5). Session at IGARSS'05 (on CD)

Burlina, P., Chellappa, R. and Lin, C. (1997): A Spectral Attentional Mechanism Tuned to Object Configurations, IEEE Transactions on Image Processing 6, 1117–1128.

Gerhardinger, A., Ehrlich, D., Pesaresi, M., 2005. Vehicle detection from very high resolution satellite imagery. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3/W24), 83-88.

Heipke, C., Jacobsen, K., Gerke, M. (eds), 2005. High-resolution earth imaging for geospatial information. International Archives of Photogrammetry and Remote Sensing. Vol 36 Part 1 W3 (on CD)

Hinz, S., 2003. Integrating local and global features for vehicle detection in high resolution aerial imagery. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 34 (Part 3/W8), 119-124.

Hinz, S., 2005. Fast and Subpixel Precise Blob Detection and Attribution . Proceedings of ICIP 05, Genua, Sept. 11-14 2005.

Hinz, S., Stilla, U., 2006. Car detection in aerial thermal images by local and global evidence accumulation. Pattern Recognition Letters 27, 308-315.

Jin X., Davis, C.H., 2004. Vector-guided vehicle detection from high-resolution satellite imagery. In: Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS '04), Vol. 2, Anchorage, USA-AK, 20-24 September 2004, 1095 – 1098.

Michaelsen, E., Stilla, U., 2001. Estimating Urban Activity on High-Resolution Thermal Image Sequences Aided by Large Scale Vector Maps. In: Proc. IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas, Rome, Italy, 8-9 November 2001, 25-29

Mikhail, E. M., 1976. Observations and least squares. IEP – A Dun-Donnelly Publisher, Copyright: Thomas Y. Crowell Company Inc., New York

Ramer, U. (1972): An Iterative Procedure for the Polygonal Approximation of Plane Curves, Computer Graphics and Image Processing 1, 244-256.

Ruskoné, R., Guiges, L., Airault, S., Jamet, O., 1996. Vehicle detection on aerial images: A structural approach. In: Kropatsch, G. (Eds.), Proc. 13th International Conference on Pattern Recognition (ICPR'96), Vol. 3, IEEE Computer Society Press, Vienna, Austria, 25 -29 August 1996, 900-903.

Sharma, G., 2002. Vehicle detection and classification in 1-m resolution imagery. MSc Thesis, Ohio State University, Columbus, USA-OH.

Steger, C., 1998. An Unbiased Detector of Curvilinear Structures. IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (2), 113-125.

Stilla, U., Michaelsen, E., Soergel, U., Hinz, S., Ender, J., 2004. Airborne monitoring of vehicle activity in urban areas. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 35 (Part B3), 973-979.

Stilla, U., Rottensteiner F., Hinz. S. (Eds.), 2005. Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring - Concepts, Algorithms, and Evaluation (CMRT05), Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, Part 3/W24

ANALYSIS OF FULL WAVEFORM LIDAR DATA FOR TREE SPECIES CLASSIFICATION

Josef Reitberger [a, *], Peter Krzystek [a], Uwe Stilla [b]

[a] Dept. of Geoinformatics, Munich University of Applied Sciences, 80333 Munich, Germany
(reitberger, krzystek)@fhm.edu
[b] Photogrammetry and Remote Sensing, Technische Universitaet München, 80290 Munich, Germany
stilla@tum.de

Commission III, WG III /3

KEY WORDS: LIDAR, analysis, segmentation, classification, forestry, vegetation

ABSTRACT:

The paper describes an approach to tree species classification based on features that are derived by a waveform decomposition of full waveform LIDAR data. Firstly, 3D points and their attributes are extracted from the waveforms, which yields a much larger number of points compared to the conventional first and last pulse techniques. This is caused by the detailed signal analysis and the possibility to detect multiple pulse reflections. Also, constraints are embedded into the mathematical model of the decomposition to avoid erroneous 3D points caused by the system electronics. Secondly, special tree saliencies are proposed, which are computed from the extracted 3D points. Subsequently, an unsupervised tree species classification is carried out using these saliencies. The classification, which groups the data into two clusters (deciduous, coniferous), leads to an overall accuracy of 80 % in a leaf-on situation. Finally, the results are shortly discussed.

1. INTRODUCTION

Remote sensing techniques have great potential to automatically derive forest structures and parameters that are captured so far in time consuming campaigns with considerable manpower. Microwave sensors, optical sensors, and laser sensors record data that contain inherent object information as a result of the interaction of the sensor specific wavelength with the forest.

Microwave sensor systems have the advantage to penetrate the forest structure, are weather independent and less expensive. Primarily due to the complex back scattering mechanism the application of both InSAR and PolInSAR in forest is restricted to the extraction of overall parameters like the biomass parameter and the generation of a crude DSM (Aulinger et al., 2005). SAR tomography does not depend on assumptions about the spatial forest structure like PolInSAR and represents a true 3D mapping technique, however is far from practicability (Reigber, 2001).

Optical sensors map at a high ground sample distance the canopy surface in particular bands depending on the spectral resolution. The measured pixel value represents only the intensity that is directly reflected from the canopy surface element. The way the sun light photons are interacting with the entire tree structure is not recorded. Thus, spatial forest parameters can only be derived indirectly from surface parameters like the crown diameter. So far, most of the applications with optical sensors aimed at medium and small scale forest inventory (Hyyppä et al., 2000). New digital aerial cameras providing multi-spectral images with high geometrical resolution are promising. DSM generation and tree height determination based on image correlation has been reported recently (Baltsavias et al., 2006). Also, tree species classification was demonstrated using the DMC digital aerial camera (Persson et al., 2004).

LIDAR comprises several advantages for forest applications. The laser beam may penetrate the forest structure, and the technique provides 3D information at a high point density and intensity values at a specific wavelength. Since over a decade conventional LIDAR - recording the first and last pulse - has been widely used to successfully retrieve forest parameters (Hyyppä et al., 2004; Heurich et al., 2004). Holmgren (2003) shows that single trees like Norway Spruce and Pine can be delineated and classified using highly dense LIDAR data in forest structures which are typical for Scandinavian forests. However, conventional LIDAR data do not provide desirable forest features like young regeneration due to the limited range resolution and penetration rate.

A conventional LIDAR system has limitations concerning the number of recordable pulse reflections. Also, the information about the reflecting object and its geometric and physical characteristics is not registered. New full waveform scanners overcome this drawback, since they record the entire laser pulse echo as a function of time. Therefore, detailed information about the geometric and physical characteristics of the tree structure can be derived and used to retrieve more sophisticated and precise forest structures.

The presented work is focused on automated extraction of forest parameters. The overall goal is to replace time consuming and expensive methods of forest inventory by new techniques exploiting LIDAR data from new full waveform scanners. In this paper we report on an approach to waveform decomposition and tree species classification using full waveform LIDAR data.

[*] Corresponding author.

The paper is structured by five sections. Section 2 describes the decomposition of the waveform based on a robust adjustment scheme. Section 3 presents the approach to tree species classification. Section 4 shows results obtained from full waveform data collected in fall 2004 by the TopEye MK II system in the Bavarian Forest National Park. Finally, the results are discussed with conclusions in section 5 and 6.

2. DECOMPOSITION OF FULL WAVEFORM DATA

2.1 General Remarks

Generally, the recorded waveform is influenced by the transmitted pulse, the atmosphere and the object. Wagner et al. (2003) present a theoretical model for the interaction of a single laser beam with topographic targets like leaves, power lines, roofs and trees. For simplification, it neglects the mitigation of the laser pulse when travelling through a tree volume. Several approaches to decompose a single waveform have been published. Hofton et al. (2000) suggest to fit several Gaussian distribution functions in a nonlinear least squares adjustment to the waveform. Likewise, Jutzi and Stilla (2005) model the waveform with Gaussians by a Gauss-Newton method. Finally, Persson et al. (2005) introduced another method based on the Expectation-Maximation algorithm.

2.2 Approach

Our approach to decompose the full waveform data is based - similar to Hofton et al. (2000) - on the assumption, that the transmitted pulse is of Gaussian type and the registered waveform is composed from several single laser returns that are also of Gaussian type. Thus, we model the waveform *w(t)* with a sum of single Gaussian distribution functions

$$w(t) = \varepsilon + \sum_{m=1}^{N_p} A_m \exp\left[-\frac{(t-t_m)^2}{2\sigma_m^2}\right] \qquad (1)$$

with

N_P : Number of peaks A_m: Amplitude of the m^th^ peak

ε : Bias (noise level) t_m : Time position of the m^th^ peak

σ_m : half width of the m^th^ peak

The nonlinear observation equation (1) is linearized with respect to the unknown model parameters $x^T = (\varepsilon, A_m, t_m, \sigma_m)$ ($m = 1, N_p$). A standard least squares adjustment estimates the unknown variables *x* by the normal equation system

$$(A^T PA)x = A^T Pl \qquad (2)$$

with A as the design matrix, *P* as the weighting matrix and *l* as the observation vector.

Since initial experiments showed that the standard least squares adjustment cannot clearly extract single returns from the registered waveform in case of overlaying return pulses, the Levenberg-Marquardt (LM) (Levenberg, 1944; Marquardt, 1963) iteration scheme was added by replacing the normal equation matrix $N = A^T PA$ in (2) with augmented normal equations N', where

$$N'_{ii} = (1+\lambda)N_{ii} \text{ and } N'_{ij} = N_{ij} \text{ for } i \neq j . \qquad (3)$$

The damping factor λ is initially set to 10^{-3} and is scaled down by the factor 10 as long as the solving of the normal equations shows a good convergence. In case of a divergence λ is multiplied by 10 and the normal equations are solved again. This process continues until the normal equation converges significantly.

The initial values $\varepsilon^0, A_m^0, t_m^0, \sigma_m^0$ for the unknown parameters are derived as follows. The median of the waveform *w(t)* is used as starting value for ε, i.e. $\varepsilon^0 = median(w(t))$. Initial values A_m^0 for the amplitudes and t_m^0 for the time positions of the peaks are found by smoothing the original signal by a 1x3 Gaussian filter and computing the first derivative of the smoothed curve. Possible time positions t_m^0 of a peak are zero crossings of the first derivative of *w(t)*. In order to distinguish between real returns and noise a threshold $C_{\text{threshold}}$ based on the median absolute deviation *MAD=median(|w(t) - median(w(t))|)* of the waveform *w(t)*, which is a measure of dispersion of a distribution about the median (Rousseeuw and Leroy, 1987), is calculated. The threshold $C_{\text{threshold}}$ is set to

$$C_{threshold} = median(w(t) + 3 \cdot 1.4826 \cdot MAD) \qquad (4)$$

in order to achieve consistency with the standard deviation for asymptotical normal distributions. We just select potential local maxima with amplitudes larger than the threshold $C_{\text{threshold}}$. The initial values σ_m^0 are set to 0.25 m, which is equivalent to the standard deviation of the transmitting pulse (pulse length 5 ns) assuming that it is of Gaussian type.

The internal accuracy of the estimated parameters are derived from the inverse normal equation matrix N^{-1} and the sigma naught σ_0. Since the scan angle of the laser beam is rather small the standard deviation σ_{t_m} of the peak position is a good estimation of the height standard deviation of the corresponding 3D point $X_m^T = (x_m, y_m, z_m)$, i.e. $\sigma_{z_m} \approx \sigma_{t_m}$. This value is used as a quality measure after the adjustment to discard possible weak points from any further analysis based on a certain threshold.

2.3 Extraction of 3D points

The estimated time positions t_m of the Gaussian functions are used along with the starting point $X_s^T = (x_s, y_s, z_s)$, the direction vector r_s and the start time t_s of the waveform to generate the 3D points of the waveform with $X_m = X_s + (t_m - t_s)r_s$ $(m = 1, N_p)$. Additionally, these points get the width W_m of the return pulse and the intensity related parameter I_m of the reflection as attributes. The width W_m is set to twice the estimated standard deviation σ_m, i.e. $W_m = 2 \cdot \sigma_m$.The parameter I_m is derived from the integral of the Gaussian function that can be approximated with $I_m = 2 \cdot \sigma_m \cdot A_m$ and is equivalent to the pulse energy of the reflection. Note that the parameters W_m and I_m are still sensor specific since they depend on the amplitude and pulse length of the emitted signal. Also, I_m depends on the run length s_m of the laser beam. Calibration is achieved by referencing W_m and I_m to

W^e and I^e of the emitted Gaussian pulse and correcting I_m with respect to a nominal distance s_0 according to the radar equation (Wagner et al. 2003). This leads to calibrated parameters $W_m^c = \frac{W_m}{W^e}$ and $I_m^c = \frac{I_m \cdot s_m^2}{I^e \cdot s_0^2}$, which represent additional information about the reflections of the laser beam on targets and can be used in a tree species classification.

3. TREE SPECIES CLASSIFICATION

3.1 Concept

Tree species classification is usually split up into three main steps. Firstly, individual tree crowns are delineated by a segmentation of the canopy height model, which describes the tree surface. Secondly, characteristic features of the individual trees are extracted. Thirdly, based on the extracted features tree species are classified using an appropriate classifier. So far, we have been concentrating on the second and third step and postponed the segmentation of the tree crowns.

3.2 Feature extraction

The finding of significant features describing the tree individually is a key issue in tree species classification. Assuming a given tree segment we have several waveforms intersecting the prismatic volume area from which in total *n* 3D points $X_j = \{x_j, y_j, z_j, I_j^c, W_j^c\}$ $(j = 1, n)$ can be derived containing the coordinates and the attributes of the waveform decomposition.

The salient features $S_t = \{S_g, S_i, S_I\}$ of a tree t are subdivided into three groups reflecting the outer tree geometry by S_g, the internal geometrical tree structure by S_i and the intensity-related tree structure by S_I.

For the group S_g we have developed two saliencies $S_g = \{S_g^1, S_g^2\}$. The first saliency S_g^1 consists of the parameters $\{a,b\}$ of a parabolic surface $z = a \cdot (x - x_0)^2 + b \cdot (y - y_0)^2 + z_0$ that is fitted to the 3D points of the crown shape. These points are found with a convex hull algorithm applied to the crown points. Crown points are selected from the tree segment points by discarding possible ground hits within a height bound of 1 m above the DTM and points below the crown base height h_{base}. The value for h_{base} is found by splitting the tree segment into height layers of 0.5 m and finding the lowest layer that contains more than 1% of the non-ground points. The parameters (x_0, y_0, z_0) are either adjusted or are set equal to the coordinates of the highest point found in the tree segment.

For the calculation of the second saliency S_g^2 we subdivide the tree in *l* tree layers (Figure 1). The saliency S_g^2 is composed of the mean radii $S_g^2 = \{r_k\}(k = 1, l)$ that are determined as the mean distances $r_k = \frac{1}{N_k}\sum_{i=1}^{N_k}\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}\,(k = 1, l)$ of all N_k layer points to the tree trunk (x_0, y_0), which is set equal to the planimetric coordinates of the highest crown point (Figure 1a).

The saliency group $S_i = \{S_i^h, S_i^d\}$ describing the internal tree structure is inspired by metrics introduced for tree characterization (Naesset, 2004). The saliencies $S_i^h = \{h_k\}(k = 1, l)$ are the percentiles of the LIDAR point height distribution in a tree segment and also referred to as height dependent variables (Figure 1b). The saliencies $S_i^d = \{d_k\}(k = 1, l)$ are defined as the number of LIDAR points in *l* tree layers from height $((k-1)/l) \cdot h_{tree}$ to height $(k/l) \cdot h_{tree}$ normalized by the total number of LIDAR points in a tree segment (Figure 1c).

The key idea to introduce the third saliency group $S_I = \{S_I^1, S_I^2\}$ is to use the intensity information the waveform decomposition provides for each point. We compute in *l* tree layers with N_k layer points the mean values $I_k^c = \frac{1}{N_k}\sum_{i=1}^{N_k} I_i^c\,(k = 1, l)$ composing the saliency $S_I^1 = \{I_k^c\}(k = 1, l)$. Additionally, we introduce the saliency $S_I^2 = \{I_{mean}^c\}$ as the overall intensity related value for the entire tree segment.

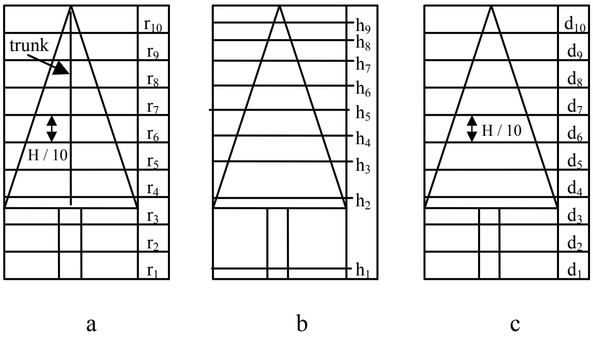


Figure 1: Tree layers a)-c) Different distributions

3.3 Classification

The tree species classification is a 2-step procedure beginning with a clustering of the tree species and a subsequent Bayes classification. Let $S_t = \{S_g, S_i, S_I\}$ be the salient features of a tree *t* to be classified and let $T_k = \{\mu_k, \Sigma_k\}$ be the density probability model (mean, covariance matrix) of the k^{th} tree class. The probability that a tree *t* is a member of the k^{th} tree class is given by

$$\mathrm{p}(S_t|T_\mathrm{k}) = \frac{1}{(2\pi)^d \sqrt{|\Sigma_k|}} \exp(-\frac{1}{2}(S_t - \mu_k)^T \Sigma_k (S_t - \mu_k)) \quad (5)$$

where *d* is the number of saliencies.

The density probability models T_k are found by the Expectation-Maximization algorithm that approximates the distribution of a saliency subset *S* ε S_t by

$$p(S) = \sum_{k=1}^{s} \pi_k N(S \mid \mu_k, \Sigma_k) \quad (6)$$

with π_k as the mixing coefficients, $N(S \mid \mu_k, \Sigma_k)$ as the multivariate Gaussian distribution and *s* as the number of

Gaussians. Note, if we just apply the clustering step (6) to the entire set of tree saliencies, we will receive the simple case of an unsupervised tree species classification. Otherwise, step (6) is the learning process of the Bayes classification (5).

4. EXPERIMENTS

Experiments were conducted in the Bavarian Forest National Park that is located in south-eastern Germany along the border to the Czech Republic. The waveform data have been collected by the new full waveform system MK II from TopEye, which is operating at a wavelength of 1550 nm and a PRF of 50 kHz. The scan angle varies within 14 and 20 degrees. The system was flown in late September 2004 at a flying height of 200 m resulting in a nominal point density of approximately 25 points/m^2. Due to the fixed sampling length of 128 samples, the waveform was limited to about 19 m. The sampling rate was 1 GHz providing a vertical resolution to 15 cm. The pulse length of 5 ns created a pulse width with a standard deviation of 25 cm. The emitted Gaussian pulse was not available. Finally, the footprint was 20 cm because of the beam divergence of 1 mrad.

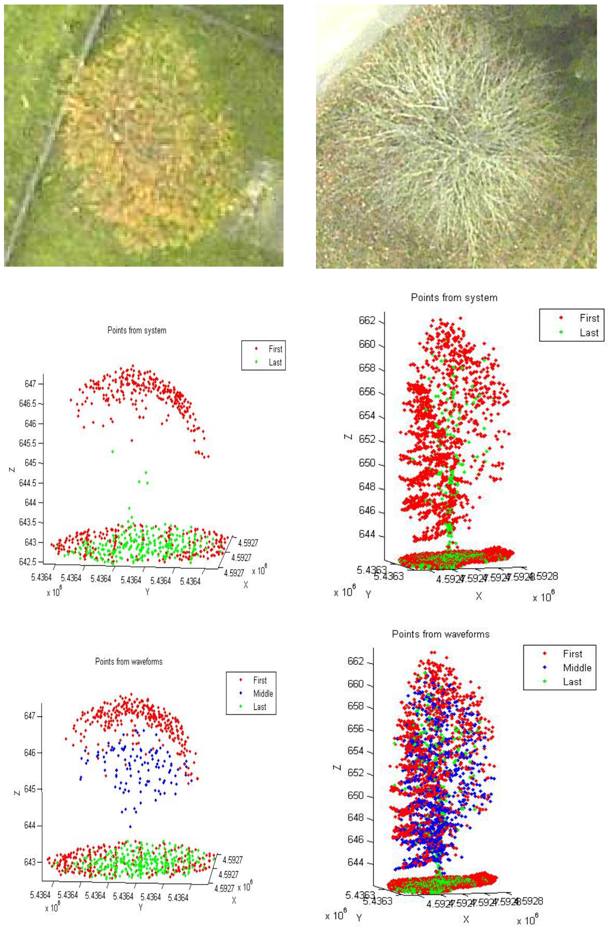


Figure 4: Aerial images of areas 1 and 2 in row 1; Points derived by the TopEye system in row 2, grouped in “First” and “Last” pulse points; Points derived from the waveforms in row 3, grouped in “First” and “Last” pulse points and points between “First” and “Last” pulse (labelled as “Middle” points)

The flown area is of size 500 m x 1700 m and is mainly characterized by Norway Spruce and European Beech. Segmented tree crowns have been derived from a canopy height model (CHM) by a watershed-based algorithm. The CHM was generated from an earlier laser scanning campaign (Heurich et al., 2004). The total number of tree segments amounted to 1000. Reference data for Norway Spruce and European Beech were available in 97 and 23 segments, respectively. In each segment we took the highest tree as the reference tree.

In a first step we applied the waveform decomposition to four sample trees and one meadow area in order to demonstrate the potential of the approach. Firstly, an area of interest was defined in digital orthophotos by manually digitizing a polygon. Secondly, 3D points were generated from all the waveforms intersecting the corresponding prismatic volume segment. The resulting 3D points were grouped into the 3 classes “First”, “Last” and “Middle”. The classes “First” and “Last” contain all the points derived from the first and last detected peak (t_1, t_{Np}). All the other points referring to t_m (m = 2, N_p–1) were classified as “Middle”. For comparison, we selected also the first and last pulse points the TopEye system created conventionally with its standard detection procedure. Figure 4 illustrates graphically two sample trees and table 1 contains numerically the number of points extracted by the TopEye system and our waveform decomposition. Note that the single trees 1, 2 and 3 are free-standing. Tree 4 refers to a group of trees in closed forest.

Area		1	2	3	4	5
Tree specie / object type		Deciduous (leaf-on)	Deciduous (leaf-off)	Coniferous	Deciduous (leaf-on) and coniferous	Meadow
Size [m^2]		21.9	72.2	22.2	86.7	28.3
Points from Top-Eye	Total	768	5594	1109	1602	362
	First	503	4168	882	1191	362
	Last	265	1426	227	411	0
Points derived from wave-forms	Total	943	7436	2555	3261	456
	First (%)	553 (59)	4648 (62)	1483 (58)	1678 (51)	456 (100)
	Last (%)	280 (30)	1548 (21)	727 (28)	969 (30)	0 (0)
	Middle (%)	110 (11)	1240 (27)	345 (14)	614 (19)	0 (0)

Table 1: Comparison of points derived by the TopEye system and by the waveform decomposition

Saliency	S_g^1	S_g^2	S_i^h	S_i^d	S_I^1	S_I^2	$S_g^1+S_I^2$	$S_g^2+S_I^2$
Conif. [%]	86	94	54	69	79	80	80	80
Decid. [%]	65	65	74	61	65	83	83	82
Total [%]	81	88	58	67	77	81	81	80

Table 2: Overall accuracy of tree species classification

In the next step tree species classification was carried out for the two classes “Coniferous” and “Deciduous” by applying the saliencies $S_t=\{S_g,S_i,S_I\}$ of 1000 tree segments to the unsupervised clustering approach (6). We postponed the supervised classification with (5) because of the few reference data for European Beech. The overall classification accuracy was derived from the reference data by determining error type I and error type II. We introduced 10 layers for the saliencies S_g^2, S_i^d and S_I^1, and calculated height percentiles for S_i^h in steps of 10%. We only used the last values r_{10} and I_{10} for S_g^2 and S_I^1. Table 2 summarizes the classification results.

5. DISCUSSION

The application of the waveform decomposition to the LIDAR data showed that on average 2 to 3 returns could be detected from a single emitted waveform. Thus, the overall point density of 25 pts/m^2, which is a function of the flying height, the PRF and the flying speed, was increased to roughly 60 pts/m^2.

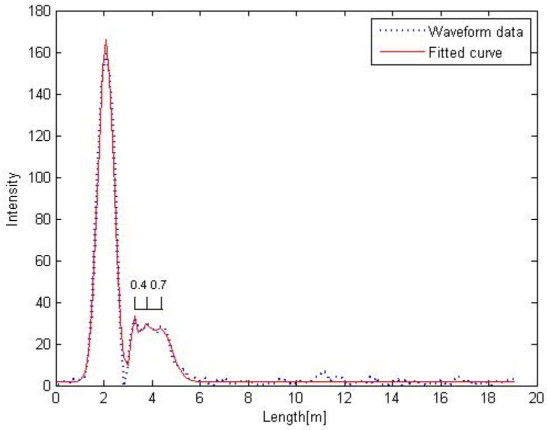


Figure 2: Separated overlaying returns

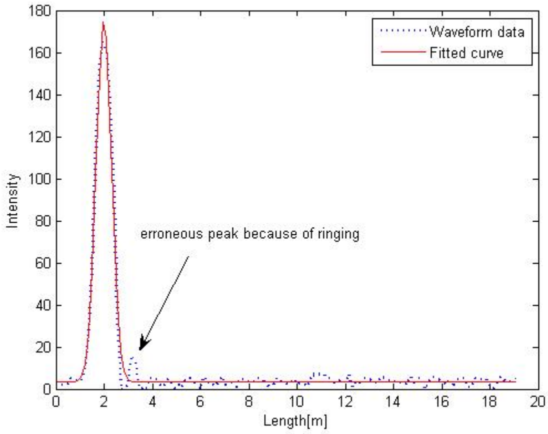


Figure 3: Waveform from a roof with an erroneous peak and the fitted curve where this peak is ignored

Interestingly, the LM iteration scheme (3) successfully separates overlaying returns. Figure 2 shows that three overlaying return pulses could be clearly split up. The corresponding peaks have a distance of 0.4 m and 0.7 m respectively, which is in the order of the nominal height separability derived from the pulse length of 5 ns (Wagner et al. 2003). Note that conventional LIDAR systems can practically discern two return pulses with a distance of about 3 m. Furthermore, the adjustment approach evidences an excellent mean height standard deviation σ_{t_m} of about 2 cm for all points decomposed from the waveforms. This is roughly by the factor 7 better than the nominal height resolution of 15 cm.

Surprisingly, some waveforms contained erroneous peaks that are a typical effect of bandwidth limited receiver electronics called “ringing" and can be observed - most prominently - when the registered light intensity is high. In worst cases there might even occur 2 additional pseudo peaks after the dominant large peak that only results from one reflection. Figure 3 shows a typical example of a waveform resulting from a roof reflection. Two rules have been established to avoid the extraction of pseudo 3D points in that case. The second peak is ignored if it is closer than 1.5 m to the first peak and, secondly, if its amplitude is smaller than 1/5 of the amplitude of the first peak.

Table 1 evidences that the waveform decomposition provides significantly more points than the standard TopEye detection mode. The smallest improvement of about 25% can be observed at tree 1, which is a small deciduous tree in a leaf-on situation. In the area of the coniferous tree 3 the waveform decomposition creates even more than 100% additional points. Two main reasons can be found for this. Firstly, the waveform decomposition decorrelates all the significant returns of the laser beam. Sometimes, up to four or even more points can be found between the first and last peak. Such points are totally ignored by a conventional system. The percentage of the “Middle” points to the total number of decorrelated points varies between 10% and 30%. Secondly, since the waveform decomposition can be flexibly controlled by tuning parameters, it also decorrelates points with a low intensity. Again, many of such points are not registered by a conventional system due to the internal threshold for signal detection. In other words, the higher sensitivity of the waveform decomposition generates much more points. This becomes especially apparent in area 5 (=meadow), where only first pulse points occur.

The classification results of table 2 show, that the saliencies S_g^i describing the crown shape work best for the coniferous trees. Classification just using the saliencies S_i^d and S_i^h representing the internal geometrical structure results in a worse accuracy. The height dependent saliency S_i^h is better for deciduous trees, whereas the density dependent saliency S_i^d works better for coniferous trees. In comparison to this, classification with intensity related saliencies S_I^i yields better results. Especially, the saliency S_I^2 describing the mean intensity value of the segmented tree improves the classification of deciduous trees. However, the saliency S_I^1 describing the intensity related value in the upper tree layer is much worse with 65%. Interestingly, if we combine the best intensity related saliency S_I^2 with the crown shape saliencies S_g the classification results for both tree species is practically the same as with the intensity related saliency S_I^2. Obviously, the classification results are mainly influenced by the crown shape geometry and the mean intensity related value of the tree. Especially, the crown shape drives the classification for coniferous trees considerably. Height and density dependent saliencies are not as good as expected. Probably, the characteristic tree structures are not clearly reflected in these saliencies. The reasons are manifold. Firstly, the waveforms have just a limited length of 19 m and do not penetrate the lower parts of the trees. Secondly, since the data collection was in September, the beeches were partly in nearly leaf-off or leaf-on situation. Possibly, this caused a different point distribution within one specie. Thirdly, the tree segments resulted from an earlier flight mission with lower point density. In some cases we could observe segments containing for instance several trees or artefacts. Also, smaller trees beneath the tree crown and branches from neighbouring trees may contribute to the tree structure and therefore falsify the saliencies. Notably, we could not clearly identify such cases by introducing a third class as an outlier class. The intensity information turned out to be as the parameter classifying both tree species practically with the same accuracy. Interestingly, just the mean intensity related value S_I^2 yielded the main contribution. Using the values of S_I^1 for all tree height layers resulted in a worse accuracy. Possibly, the number of detected return pulses was too small for the individual layers.

6. CONCLUSIONS

The presented study results show clearly the potential of full waveform data for the comprehensive analysis of tree structures. The number of extracted points is much larger if compared to conventional systems. Future research should evaluate (i) new saliencies for tree species classification based on the 3D points and the waveform signal, which clearly reflect micro structures of the trees like the stem and branches, (ii) waveform data with unlimited length, (iii) influence of point density and (iv) classification of tree sub classes.

7. REFERENCES

Aulinger, T., Mette, T., Papathanassiou, K.P., Hajnsek, I., Heurich, M., Krzystek, P. 2005. Validation of heights from interferometric SAR and LIDAR over the temperate forest site "Nationalpark Bayerischer Wald. *POLINSAR Workshop*, 17th - 20th January, Rome, Italy.

Baltsavias M., Grün, A., Küchler, M., Thee, P., Waser, L.T., Zhang, L. 2006. Tree height measurements and tree growth estimation in a mire environment using digital surface models. *International Workshop "3D Remote Sensing in Forestry", 14 - 15th February, University of Natural Resources and Applied Life Sciences*, 13 -15th, February, Vienna.

Heurich, M., Weinacker, H., 2004. Automated Tree Detection and Measurement in Temperate Forests of Central Europe Using Laserscanning Data. *Proceedings of the ISPRS working group VIII/2 Laser-Scanners for Forest and Landscape Assessment,* Volume XXXVI, PART 8/W2, 3 – 6th October, Freiburg, pp. 198 – 203.

Hofton, M., Minster, J., Blair. J.B. 2000. Decomposition of Laser Altimeter Waveforms. *IEEE Transactions on Geoscience and Remote Sensing*, 38:1989-1996.

Holmgren, J. 2003. Estimation of Forest Variables using Airborne Laser Scanning. Doctoral thesis, *Swedish University of Agricultural Sciences*, Umea.

Hyyppä, J., Hyyppä, H., Litkey, P., Yu, X., Haggren, H., Rönnholm, P., Pyysalo, U., Pikänen, J., Maltamo, M., 2000. Algorithms and Methods of Airborne Laser Scanning for Forest Measurements. *Proceedings of the ISPRS working group VIII/2 Laser-Scanners for Forest and Landscape Assessment,* Volume XXXVI, PART 8/W2, 3 – 6th October, Freiburg, pp. 82 – 89.

Hyyppä, J., Hyyppä, H., Inkinen, M., Engdahl, M., Linko, S., Zhu.Y. , 2000. Accuracy comparison of various remote sensing data sources in the retrieval of forest stand attributes. *Forest Ecology and Management*, 128:109-120.

Jutzi B., Stilla U. 2005. Waveform processing of laser pulses for reconstruction of surfaces in urban areas. *In: Moeller M, Wentz E (eds) 3th International Symposium: Remote sensing and data fusion on urban areas, URBAN 2005. International Archives of Photogrammetry and Remote Sensing.* Vol 36, Part 8 W27.

Levenberg, K. 1944. A method for the Solution of certain non-linear problems in least-squares. *Quart. Appl. Math. 2*, pp 164-168.

Marquardt, D. W. 1963. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math. 11*, pp 413-441.

Naesset, E. 2004. Practical Large-scale Forest Stand Inventory Using a Small-footprint Airborne Scanning Laser. *Scandinavian Journal of Forest Research 19*, pp. 164 – 179.

Persson, Å, Söderman, U., Töpel, J., Ahlberg, S. 2005. Visualization and analysis of full-waveform airborne laser scanner data. *In The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Enschede, Netherlands, Vol. XXXVI, Part 3/W19, pp. 103-108.

Persson, A., Holmgren, J., Södermann U., Olsson, H. 2004. Tree species Classification of individual trees in Sweden by Combining High Resoluion Laser Data with High Resolution Near-Infrared Digital Images. *Proceedings of the ISPRS working group VIII/2 Laser-Scanners for Forest and Landscape Assessment,* Volume XXXVI, PART 8/W2, pp. 204-207, 3 – 6th October, Freiburg.

Reigber, A., 2001. Airborne Polarimetric SAR Tomography, *DLR Forschungsbericht No. 2002-2*, Doctor thesis, Stuttgart University.

Rousseeuw, P.J., Leroy, A.M. 1987. Robust Regression and Outlier Detection, *Wiley-Interscience, New York (Series in Applied Probability and Statistics)*, 329 pages. ISBN 0-471-85233-3.

Wagner, W., Ullrich, A., Briese, C. 2003. Der Laserstrahl und seine Interaktion mit der Erdoberfläche. *Österreichische Zeitschrift für Vermessung und Geoinformation (VGI)*, 91(4); 223 - 235.

8. ACKNOWLEDGEMENTS

We thank the Administration of Bavarian Forest National Park to give us the opportunity to use the possibilities of their remote sensing test sites. Also, the laser measurements were financed by the Administration of Bavarian Forest National Park. We also like to thank Hakan Sterner and the staff at TopEye AB for delivering a high quality laser data set.

This research has been funded by the German Department of Education and Research (BMBF) under the contract number 17311B04.

PRECISE RANGE ESTIMATION ON KNOWN SURFACES BY ANALYSIS OF FULL-WAVEFORM LASER

B. Jutzi [a], U. Stilla [b]

[a] FGAN-FOM Research Institute for Optronics and Pattern Recognition, 76275 Ettlingen, Germany - jutzi@fom.fgan.de
[b] Photogrammetry and Remote Sensing, Technische Universitaet Muenchen, 80290 Muenchen, Germany - stilla@bv.tum.de

Commission III, WG III/3

KEY WORDS: Accuracy, Analysis, Laser scanning, LIDAR, Measurement, Full-waveform.

ABSTRACT:

Laser range data analysis is of high interest in photogrammetry. Range estimation for complex surface structures can be inaccurate. To overcome this drawback a method using a laser scanner capable of full-waveform analysis is proposed. For analysis the transmitted waveform of the emitted pulse is used to estimate the received waveform of the backscattered pulse for a known surface. We simulated a plane surface with different slopes and a sphere. Typical spatial beam distributions are considered for modeling, namely Gaussian and uniform. The surface response is determined and the corresponding received waveform is calculated. The normalized cross-correlation function in between the simulated and the measured waveform is used for precise range measurement. Additionally the position on the surface can be determined.

1. INTRODUCTION

The automatic generation of 3-d models for a description of man-made objects, like buildings, is of great interest in photogrammetric research. Laser scanner systems allow a direct and illumination-independent measurement of the range. Laser scanners capture the range of 3-d objects in a fast, contact free and accurate way. Overviews for laser scanning systems are given in (Huising & Pereira, 1998; Wehr & Lohr, 1999; Baltsavias, 1999). A general overview on how to develop and design laser systems can be found in textbooks (Jelalian, 1992; Kamermann, 1993).

Current pulsed laser scanner systems for topographic mapping are based on time-of-flight techniques to determine the range of the illuminated object. The elapsed time between the emitted and backscattered laser pulses is typically determined by a threshold detection with analog electronics. Some systems capture multiple reflections caused by objects which are smaller than the laser beam footprint located in different ranges. Such systems usually record the first and the last backscattered laser pulse.

First pulse as well as last pulse exploitation is used for different applications like urban planning or forestry surveying. While first pulse registration is the optimum choice to measure the hull of partially penetrable objects (e.g. canopy of trees), last pulse registration should be chosen to measure non-penetrable surfaces (e.g. ground surface below vegetation).

Beside the first or last pulse exploitation the complete waveform in between is of interest, because it includes the backscattering characteristic of the illuminated field. Investigations on the waveform analysis were done to explore the vegetation concerning the bio mass, foliage or density (e.g. trees, bushes, and ground). NASA has developed a prototype of the Laser Vegetation Imaging Sensor (LVIS) recording the waveform to determine the vertical density profiles in forests (Blair *et al.*, 1999). This experimental airborne system operates at altitudes up to 10 km and provides a large footprint diameter (up to 80 m) to study different land cover classes.

The spaceborne Geoscience Laser Altimeter System (GLAS) on Ice, Cloud and land Elevation Satellite (ICESat) determines changes in range through time, height profiles of clouds and aerosols, ice sheet and land elevations, and vegetation (Brenner *et al.*, 2003; Zwally *et al.*, 2002). It operates with a large footprint diameter (70 m) on Earth and measures elevation changes with decimeter accuracy (Hoften *et al.*, 2000).

Beside large footprint systems first developments of small footprint systems were done for monitoring the nearshore bathymetric environments with the Scanning Hydrographic Operational Airborne Lidar Survey system (SHOALS). SHOALS has been in full operation since 1994 (Irish & Lillycrop, 1999; Irish *et al.*, 2000). Recent developments of commercial airborne laser scanner systems led to systems that allow capturing the waveform: LITEMAPPER 5600, OPTECH ALTM 3100, TOPEYE II, and TOPOSYS HARRIER 56. The systems mentioned above are specified to operate with a transmitted pulse width of 4-10 ns and allow digitization and acquisition of the waveform with approximately 0.5-1 GSample/s.

To interpret the received waveform of the backscattered pulse, a fundamental understanding of the physical background of pulse propagation and surface interaction is important (Jutzi *et al.*, 2002; Wagner *et al.*, 2003). The influence of the surface on the transmitted waveform is discussed by Steinvall (2000) for objects with different shapes taking into account different reflection characteristics. Gardner (1982) and Bufton (1989) investigated the pulse spreading by the impact of the surface structure, e.g. surface slope and vertical roughness within the laser footprint.

The recording of the received waveform offers the possibility to use different methods for the range determination, e.g. peak detection, leading edge detection, average time value detection, constant fraction detection. This topic was investigated by different authors, e.g. Der *et al.*, 1997; Steinvall & Carlsson, 2001; Jutzi & Stilla, 2003; Thiel & Wehr, 2004; Wagner *et al.*, 2004; Vandapel *et al.*, 2004. The analysis of the pulse shape increases the reliability, accuracy, and resolution.

The range estimation is further improved by the comparison between the transmitted and the received waveform. This can be done by signal processing methods (e.g. cross-correlation, inverse filtering), if the sampling of the waveform is done with

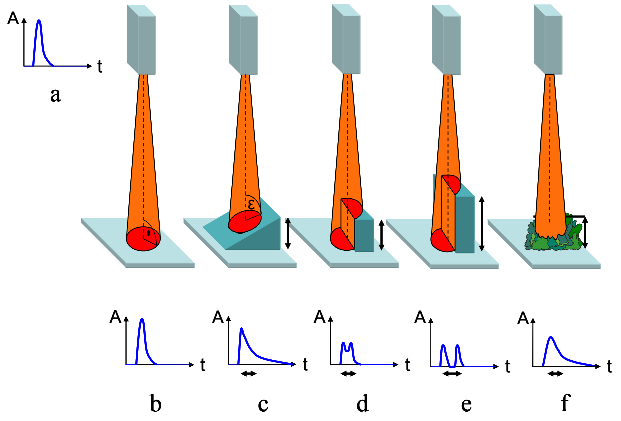


Figure 1. Effects of the surface on the received waveform.
a) transmitted waveform,
b) plane surface,
c) sloped surface,
d) two slightly different elevated areas,
e) two significantly different elevated areas,
f) randomly distributed small objects.

a high sampling rate. The maximum of the cross-correlation between the transmitted and received signal estimates the range value with a higher reliability and accuracy than considering the received waveform only (Hofton & Blair, 2001; Jutzi & Stilla, 2005; Thiel *et al.*, 2005).

Beside the range determination further surface features can be studied by waveform analysis, namely *reflectance*, *slope* and *roughness*. This specific surface features have an influence on the *amplitude* and *width* of the received waveform (Brenner *et al.*, 2003; Jutzi & Stilla 2002; Steinvall et al., 2004; Wagner *et al.*, 2006). For a parametric description of the pulse properties a Gaussian decomposition method on the waveform can be used (Hofton *et al.*, 2000; Jutzi & Stilla 2005; Persson *et al.*, 2005; Söderman *et al.*, 2005). Nowadays, waveform analysis is more and more established for remote sensing applications especially in forestry (Hug *et al.*, 2004; Reitberger *et al.*, 2006).

Depending on the application different surfaces have to be analyzed, e.g. for urban objects we have to deal with different elevated objects. In rural environment we have to deal with statistically distributed natural objects. The impact of the scene on the received waveform will be discussed using some standard examples (Figure 1). Different elevated object surfaces within the beam corridor lead to a mixture of different range values. A simple situation is given by a horizontal plane surface which will lead to a small pulse (Figure 1b). A plane which is slanted in relation to the viewing direction shows different range values within the footprint. This range interval which is given by the size of the footprint and the orientation of the plane leads to a spread of the pulse width (Figure 1c). A deformation of the pulse form can also be caused by perpendicularly oriented plane surfaces shifted by a small step in viewing direction (Figure 1d). A large step leads to two separate pulses (Figure 1e). Several surfaces with different range within the beam can result in multiple pulses. Randomly distributed small objects (e.g. by vegetation) spread over different range values within the beam leads as well to a spread of the pulse width (Figure 1f). These examples show the influence on the waveform by standard surface situations. The energy distribution within the beam was not considered. For predicting received waveforms of more complex surfaces and different energy distributions a modeling and simulation of the process is required.

The modeling of the received waveform can be done when the surface is known. A typical situation where known surfaces can be used is for registration of multiple scans received from different positions or at a different time. In these cases typically retro-reflective markers in form of spheres, cylinders or planes are used (Dold, 2005) and a precise range estimation of the known surfaces are helpful for the registration process. Beside this the surface has not to be known in advance, it can be estimated by previous measurements. Then a possible refinement of each range value can be done under consideration of the surface geometry in the close neighborhood.

In Section 2, an overview on the simulation setup is given. We simulated the surface response for different slopes and a spherical surface, which is shown in Section 3. For known surface structures corresponding received waveforms can be calculated and compared with measured waveforms, which is presented in Section 4. By proofing these waveforms for similarity the position on the surface and the precise range value can be determined.

2. SIMULATION

The simulation is necessary to estimate the received waveform of the backscattered pulse received from a known surface. For the transmitted waveform of the emitted pulse a measured or a modeled waveform can be used.

By the use of a 3-d *object representation* for the object model (Figure 2-1) and the *extrinsic orientation* parameter for sensor position and orientation (Figure 2-2), the model is *sampled* to get a high-resolution range and reflectance image (Figure 2-3). The resolution has to be higher than the scanning grid we want to simulate for further processing. Considering the transmitted waveform of the emitted pulse and the spatial energy distribution of the laser beam for temporal and spatial *laser pulse properties* is relevant for modeling the laser pulse (Figure 2-4). To simulate the *scanning* of the laser system, the values of grid spacing and the divergence of the laser beam are used for convolving the high-resolution range image with the transmitted waveform and convolving the high-resolution reflectance image with the spatial energy distribution of the beam (Figure 2-5). For a range depending *1-d surface representation*, the surface response is determined by the spatial undersampling of the high resolution range and intensity image (Figure 2-6). By convolving the surface response with the transmitted waveform the received waveform is determined at the *receiver* (Figure 2-7).

For simulating the received waveform of the backscattered pulses an *object model* (Section 2.1) and a *sensor model* (Section 2.2) is required.

2.1 Object modeling

For a 3-d object representation, our simulation setup considers geometric and radiometric features of the illuminated surface in the form of 3-d object models with homogeneous surface reflectance.

The object model with homogeneous surface reflectance is then sampled higher than the scanning grid we simulate and process, because with the higher spatial resolution we simulate the spatial distribution of the laser beam. Considering the position and orientation of the sensor system we receive a high-resolution range image and reflectance image. Depending on

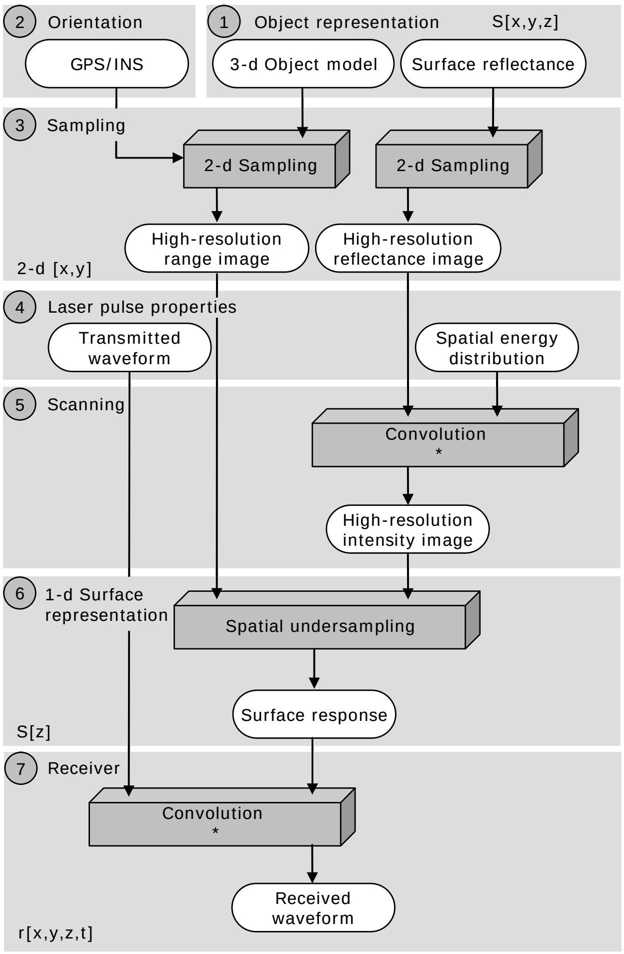


Figure 2. Simulation setup for calculating the received waveform.

the predetermined position and orientation of the sensor system, various range images can be captured.

2.2 Sensor modeling

The sensor model takes into account the specific properties of the sensing process: the position and orientation of the sensor, the laser pulse description, the scanning process and the electrical receiver properties. To simulate various aspects, a description of the extrinsic orientation of the laser scanning system with a GPS/INS system is used.

The emitted laser pulse of the system is characterized by specific pulse properties (Jutzi *et al.*, 2003). We assume radial symmetric uniform spatial distributions and radial symmetric Gaussian distributions for the beam profile, which are typical for the most laser systems. For this simulation we use measured transmitted waveforms to have a realistic description, where the bandwidth of the receiver to capture the waveform is 6 GHz and the data is sampled with 20 GSample/s. The transmitted waveform of the used system shows strong intensity fluctuations from pulse to pulse (Figure 3). The high sampling rate provides detailed information about the shape of the waveform with at least 100 sampling points for the typical length of the pulse (5 ns at Full-width-at-half-maximum).

Depending on the scan pattern of the laser scanner system, the grid spacing of the scanning process, and the divergence of the laser beam, a sub-area of the high-resolution range and reflectance images is processed. Therefore, the sub-area of the high-resolution reflectance image is convolved with the spatial energy distribution of the laser beam (distribution at the grid line ±2σ) to take into account the amount of backscattered laser light for each reflectance value. By focusing the beam with its specific properties on the detector of the receiver, the spatial resolution is reduced and this is simulated with a spatial undersampling of the sub-areas. Therefore the received high-resolution intensity and range image is processed by spatial undersampling to gain a weighted 1-d range distribution, which we call surface response. The determined surface response is convolved with the transmitted waveform to gain the received waveform of the backscattered pulse.

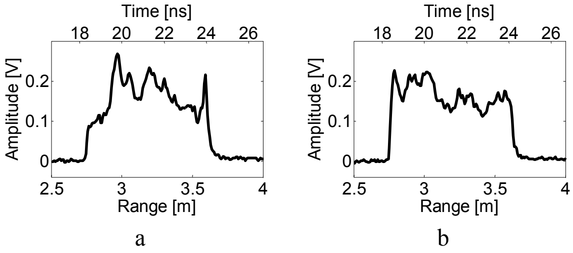


Figure 3. Two samples of the transmitted waveform.

3. CALCULATING THE SURFACE RESPONSE

The received waveform of a laser pulse depends on the transmitted waveform $s[t]$, the impulse response $h[t]$ of the receiver unit, the spatial beam distribution of the used laser $P[x,y]$, and the illuminated surface $S[x,y,z]$. The received waveform $r[x,y,z,t]$ can be expressed by a convolution of the relevant terms mentioned above and we get

$$r[x,y,z,t] = s[t] * h[t] * P[x,y] * S[x,y,z]\,, \qquad (1)$$

where (*) denotes the convolution operation. The impulse response is mainly effected by the used photodiode and amplifier, the spatial beam distribution has typically the shape of a Gaussian or uniform, and the surface characteristic can be described by its geometry and reflectance properties (mixture of diffuse and specular). We assume to have a receiver unit consisting out of an ideal photodiode and amplifier with an infinite bandwidth and a linear frequency characteristic. The 3-d surface characteristic can be reduced to a range depending 1-d signal $S[z]$, which we call in this paper surface response.

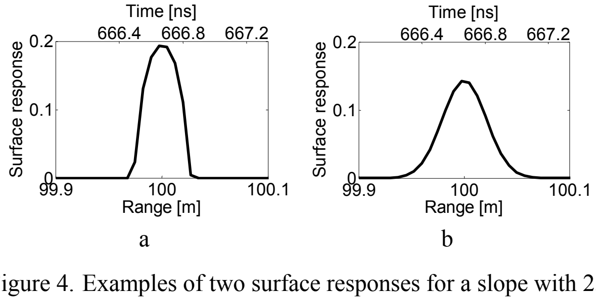


Figure 4. Examples of two surface responses for a slope with 25 degrees and different spatial beam distributions:
a) uniform,
b) Gaussian.

To study the surface response received from different surfaces, we simulated a plane surface which can be adjusted for various slopes illuminated by a beam with a spatial *uniform beam distribution* and a *Gaussian beam distribution*. Further surface responses from a small sphere with a radius of 0.3 m are determined by illuminating the surface of the sphere at different

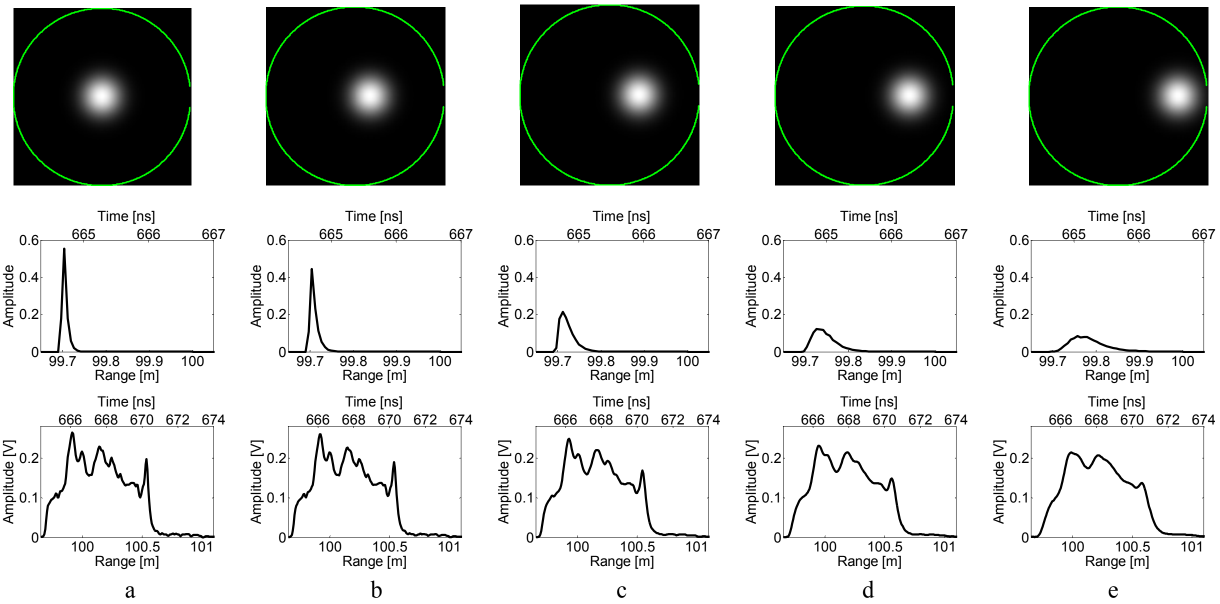


Figure 5. Position of the beam on the sphere (top row), corresponding surface responses (middle row); and the estimated received waveform (bottom row).
a) $p_0 = 0$ (center), b) $p_1 = 1/6\ r$, c) $p_2 = 2/6\ r$, d) $p_3 = 3/6\ r$, e) $p_4 = 4/6\ r$.

positions. For the simulation, the laser beam divergence is set to 1 mrad and the spatial range spacing for processing the surface response is 7.5 mm, which is equivalent to 20 GSample/s.

3.1 Plane surface with slope

We simulated a system illuminating a plane surface with different slopes. Therefore a high-resolution range image with 300x300 pixels of the sloped surfaces is calculated to determine the surface response. For surface reflectance a homogenous surface with 100% reflectance was assumed. The distance to the surface center is 100 m.

Examples of the calculated surface response $S[z]$ in dependence of the range z for a slope of 25 degrees received from an uniform beam distribution and a Gaussian beam distribution is shown in Figures 4a and 4b. The maximum of the surface response is at the range of 100m.

3.2 Spherical surface

A high-resolution range image with 300x300 pixels of a small sphere with its origin at the coordinate (0, 0, 100 m) and a radius r = 0.3 m is generated. Assuming a *Gaussian beam distribution* the surface response is calculated for five sampling positions $[p_0, p_1, p_2, p_3, p_4]$ distributed equidistantly from the center p_0 to the boundary of the sphere. For the spacing of the sampling we chose 0.5 mrad, which is approximately equivalent to 1/6 of the radius r.

The Figure 5 shows at the top row the position of the beam on the sphere, where the boundary of the sphere is visualized by a bright line. The diagrams in the middle show the corresponding surface response. With the calculated surface response the estimated received waveform for the different positions is calculated by convolving the surface response with the transmitted waveform. For exemplary waveform we selected the transmitted waveform which is depicted in Figure 3a. The received waveforms are shown in Figure 5 at the lower row. If the footprint is located at the sphere center (Figure 5a, top row) the received waveform (Figure 5a, lower row) is very similar to the transmitted waveform (Figure 3a). By shifting the footprint away from the sphere center the received waveform (Figure 5b-e, lower row) is getting more and more smeared.

4. ESTIMATING THE POSITION AND RANGE

First the transmitted and the received waveform has to be measured (Figure 6-1) with the receiver unit of the laser system. Then by the use of the transmitted waveform and the determined surface response (Figure 6-2) for each position on the surface the estimated received waveform (Figure 6-3) can be calculated by a convolution. These estimated received waveforms calculated for different positions on the surface are compared with the measured waveform by determining different normalized cross-correlation functions. With the maximum coefficient of the normalized cross-correlation functions the most likely position and the accurate range of the surface can be determined (Figure 6-4). Figure 6 depicts a schematic description of the processing chain.

4.1 Matched filter

The data analysis starts with the detection of the backscattered pulses in the temporal signal. Usually this signal is disturbed by various noise components: background radiation, amplifier

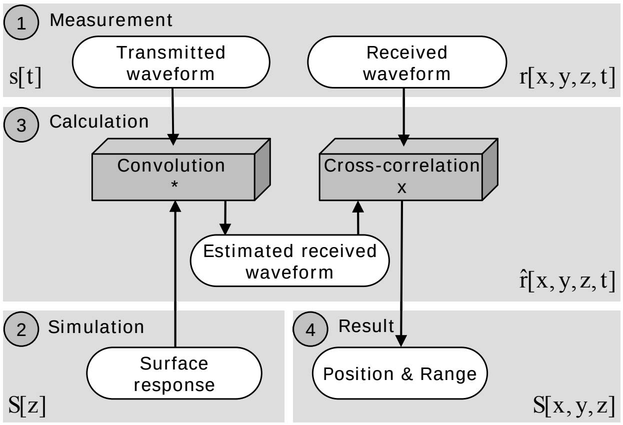


Figure 6. Processing chain to estimate position and range.

noise, photo detector noise etc. Detecting the received waveform of the backscattered pulse in noisy data and extracting the associated travel time is a well-known problem and is discussed in detail in radar techniques (Skolnik, 1980) and system theory (Papoulis, 1984). Due to this problem matched filters are used.

To improve the range accuracy and the signal-to-noise ratio (*SNR*) the matched filter for the waveform of the backscattered pulse has to be determined. In practice, it is difficult to determine the optimal matched filter. In cases where no optimal matched filter is available, sub-optimum filters may be used, but at the cost of decreasing the SNR. If the temporal deformation of the received signal can be neglected and the waveform is uniformly attenuated (isotropic attenuation by reflection or transmission of the pulse) the transmitted waveform of the emitted pulse is the best choice for the matched filter coefficients determination. In practice, the temporal deformation by the surface is common phenomenon (Figure 1). In this paper, we focus on determining this optimal filter by calculating the estimated received waveform, which can be expected from a known surface.

Let us assume that the noise components of the system mentioned above are sufficiently described by white noise with the constant factor *N*. Furthermore the signal energy of the pulse is defined as *E*. The maximum *SNR* occurs if the signal and the filter match. In this case the associated travel time *t* of the delayed pulse is τ and the *SNR* is described by

$$SNR\ [\tau] = \frac{2E}{N} \tag{2}$$

An interesting fact of this result is that the maximum of the instantaneous *SNR* depends only on the signal energy of the emitted pulse and the noise, and is independent of the shape.

Generally the matched filter is computed by the normalized cross-correlation function R_{sr} between the transmitted waveform *s*[*t*] of the emitted pulse and the estimated received waveform $\hat{r}[x, y, z, t]$ of the backscattered pulse. Assuming zero-mean waveforms, we obtain the output signal *k*[*t*] with a local maximum at the delay time τ

$$k[\tau] = R_{sr}[t+\tau] = \frac{\sum_{t=0}^{M-1} s[t] \cdot \hat{r}[x, y, z, t+\tau]}{\sqrt{\sum_{t=0}^{M-1} s[t]^2 \cdot \sum_{t=0}^{M-1} \hat{r}[x, y, z, t]^2}}, \tag{3}$$

where *M* is the length of the correlation function $k[\tau]$.

Then the output signal *k*[*t*] with improved SNR is analyzed by a detection filter searching for the local maximum to determine the travel time of the pulse. By using the correlation signal to calculate the travel time, a higher accuracy is reached than by operating on the waveform, because exploiting the shape of the pulse waveform instead of a single value increases the accuracy (Jutzi & Stilla, 2005). This is because the specific pulse properties (e.g. asymmetric shape, intensity fluctuations) are taken into account and so less temporal jitter for range estimation can be expected.

4.2 Processing the position and range

The waveform received from an unknown position on the surface is given by the measurement. To determine the position on the surface, the normalized cross-correlation functions between the measured waveform and a sample of estimated waveforms for different positions on the surface is calculated. With the maximum coefficient of the normalized cross-correlation functions, the most likely position is determined. This estimated position can be refined by calculating additional normalized cross-correlation functions and the corresponding maximum coefficients in close neighborhood. This procedure is repeated until the highest maximum coefficient is found. Then the position on the known surface and the precise range value to the surface is determined.

Because of the radial symmetry of the sphere, which is investigated in Section 3.2, the position on the surface delivers a circle of possible positions around the center of the sphere surface. If the radius of the sphere is known, then at least one additional position on the surface has to be estimated to determine the correct sphere position.

The processing time for the position and range mainly depends on calculating the surface response. The surface response is determined by the spatial undersampling of the high resolution range and intensity image. The high-resolution range image with 300x300 pixels does not have any practical relevance if it is sufficiently large to not induce errors in a higher magnitude as those incurred by our discretized beam distribution. To decrease the processing time, a smaller high-resolution range image might be sufficient on the cost of less accuracy for the range estimation.

5. CONCLUSION

In this work we have presented a scheme to estimate the precise range and position of a known surface. We simulated the surface response for different slopes and a spherical surface. Estimated waveforms received from different positions on the sphere surface are shown. The data generation and analysis we carried out are general investigations for a laser system which records the full-waveform of laser pulses. The method remains to be tested with real data, and expanded to handle more complex geometries.

REFERENCES

Baltsavias EP (1999) Airborne laser scanning: existing systems and firms and other resources. ISPRS Journal of Photogrammetry & Remote Sensing 54: 164-198.

Blair JB, Rabine DL, Hofton MA (1999) The Laser Vegetation Imaging Sensor (LVIS): A medium-altitude, digitization-only, airborne laser altimeter for mapping vegetation and topography. ISPRS Journal of Photogrammetry & Remote Sensing 56: 112-122.

Brenner AC, Zwally HJ, Bentley CR, Csatho BM, Harding DJ, Hofton MA, Minster JB, Roberts LA, Saba JL, Thomas RH, Yi D (2003) Geoscience Laser Altimeter System (GLAS) - Derivation of Range and Range Distributions From Laser Pulse Waveform Analysis for Surface Elevations, Roughness, Slope, and Vegetation Heights. Algorithm Theoretical Basis Document - Version 4.1.

Bufton JL (1989) Laser Altimetry Measurements from Aircraft and Spacecraft. Proceedings of the IEEE, Vol. 77, No. 3, 463-477.

Der S, Redman B, Chellappa R (1997) Simulation of error in optical radar measurements. Applied Optics 36 (27), 6869-6874.

Dold C (2005) Extended Gaussian images for the registration of terrestrial scan data. In: Vosselman G, Brenner C (eds) Laserscanning 2005. International Archives of Photogrammetry and Remote Sensing. Vol. 36, Part 3/W19, 180-185.

Gardner CS (1982) Target signatures for laser altimeters: an analysis. Applied Optics, Volume 21, Issue 3, 448-453.

Hofton MA, Blair JB (2002) Laser altimeter return pulse correlation: A method for detecting surface topographic change. Journal of Geodynamics special issue on laser altimetry 34, 491-502.

Hofton MA, Minster JB, Blair JB (2000) Decomposition of laser altimeter waveforms. IEEE Transactions on Geoscience and Remote Sensing 38 (4), 1989–1996.

Hug C, Ullrich A, Grimm A (2004) LITEMAPPER-5600 - a waveform digitising lidar terrain and vegetation mapping system. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), 24–29.

Huising EJ, Gomes Pereira LM (1998) Errors and accuracy estimates of laser data acquired by various laser scanning systems for topographic applications. ISPRS Journal of Photogrammetry & Remote Sensing 53: 245-261.

Irish JL, Lillycrop WJ (1999) Scanning laser mapping of the coastal zone: the SHOALS system. ISPRS Journal of Photogrammetry & Remote Sensing 54: 123-129.

Irish JL, McClung JK, and Lillycrop WJ (2000) Airborne lidar bathymetry: the SHOALS system. PIANC Bulletin. 2000 (103): 43-53.

Jelalian AW (1992) Laser Radar systems. Norwood, MA, Boston: Artech House.

Jutzi B, Eberle B, Stilla U (2002) Estimation and measurement of backscattered signals from pulsed laser radar. In: Serpico SB (ed) (2003) Image and Signal Processing for Remote Sensing VIII, SPIE Proc. Vol. 4885: 256-267.

Jutzi B, Stilla U (2003) Laser pulse analysis for reconstruction and classification of urban objects. In: Ebner H, Heipke C, Mayer H, Pakzad K (eds) Photogrammetric Image Analysis PIA'03. International Archives of Photogrammetry and Remote Sensing. Vol. 34, Part 3/W8, 151-156.

Jutzi B, Stilla U (2005) Measuring and processing the waveform of laser pulses. In: Gruen A, Kahmen H (eds) Optical 3-D Measurement Techniques VII. Vol. I, 194-203.

Kamermann GW (1993) Laser Radar. In: Fox CS (ed) Active Electro-Optical Systems, The Infrared & Electro-Optical Systems Handbook. Michigan: SPIE Optical Engineering Press.

Papoulis A (1984) Probability, Random Variables, and Stochastic Processes. Tokyo: McGraw-Hill.

Persson Å, Söderman U, Töpel J, Ahlberg S (2005) Visualization and analysis of full-waveform airborne laser scanner data. In: Vosselman G, Brenner C (Eds) Laser scanning 2005. International Archives of Photogrammetry and Remote Sensing 36 (3/W19), 109-114.

Reitberger J, Krzystek P, Heurich M (2006) Full-Waveform analysis of small footprint airborne laser scanning data in the Bavarian forest national park for tree species classification. In: Koukal T, Schneider W (Eds) 3D Remote Sensing in Forestry. 218-227.

Skolnik MI (1980) Introduction to radar systems. McGraw-Hill International Editions, Second Edition.

Söderman U, Persson Å, Töpel J, Ahlberg S (2005) On analysis and visualization of full-waveform airborne laser scanner data. Laser Radar Technology and Applications X. In: Kamerman W (ed) SPIE Proc. Vol. 5791: 184-192.

Steinvall O, Larsson H, Gustavsson F, Chevalier T, Persson Å, Klasén L (2004) Characterizing targets and backgrounds for 3D laser radars. Military Remote Sensing. In: Kamerman W, Willetts DV (eds) SPIE Proc. Vol. 5613: 51-66.

Steinvall O (2000) Effects of target shape and reflection on laser radar cross sections. Applied Optics 39 (24), 4381-4391.

Steinvall O, Carlsson T (2001) Three-dimensional laser radar modeling. In: Kamerman GW (Ed) Laser Radar Technology and Application VI, SPIE Proc. Vol. 4377, 23-34.

Thiel KH, Wehr A (2004) Performance Capabilities of Laser-Scanners - An Overview and Measurement Principle Analysis. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), 14–18.

Thiel KH, Wehr A, Hug C (2005) A New Algorithm for Processing Fullwave Laser Scanner Data. EARSeL 3D-Remote Sensing Workshop, CDROM.

Vandapel N, Amidi O, Miller JR (2004) Toward Laser Pulse Waveform Analysis for Scene Interpretation. IEEE International Conference on Robotics and Automation (ICRA 2004).

Wagner W, Ullrich A, Briese C (2003) Der Laserstrahl und seine Interaktion mit der Erdoberfläche. Österreichische Zeitschrift für Vermessung & Geoinformation, VGI 4/2003, 223-235.

Wagner W, Ullrich A, Ducic V, Melzer T, Studnicka N (2006) Gaussian Decomposition and Calibration of a Novel Small-Footprint Full-Waveform Digitising Airborne Laser Scanner. ISPRS Journal of Photogrammetry & Remote Sensing, 60 (2), 100-112.

Wagner W, Ullrich A, Melzer T, Briese C, Kraus K (2004) From single-pulse to full-waveform airborne laser scanners: Potential and practical challenges. In: Altan MO (ed) International Archives of Photogrammetry and Remote Sensing. Vol 35, Part B3, 201-206.

Wehr A, Lohr U (1999) Airborne laser scanning – an introduction and overview. ISPRS Journal of Photogrammetry & Remote Sensing 54: 68-82.

Zwally HJ, Schutz B, Abdalati W, Abshire J, Bentley C, Brenner A, Bufton J, Dezio J, Hancock D, Harding D, Herring T, Minster B, Quinn K, Palm S, Spinhirne J, Thomas R (2002) ICESat's laser measurements of polar ice, atmosphere, ocean, and land. Journal of Geodynamics 34 (3–4), 405–445.

PERFORMANCE ANALYSIS OF SPACEBORNE SAR VEHICLE DETECTION AND VELOCITY ESTIMATION

Franz Meyer[‡], Stefan Hinz[†], Andreas Laika[†], Steffen Suchandt[‡], Richard Bamler[†,‡]

[†]Remote Sensing Technology, Technische Universitaet Muenchen, Arcisstr. 21, D - 80333 Muenchen, Germany
[‡]Remote Sensing Technology Institute, German Aerospace Center (DLR), Oberpfaffenhofen, D - 82234 Wessling, Germany

Commission III/5

KEY WORDS: Traffic Monitoring, SAR, Satellite Images

ABSTRACT:

With TerraSAR-X and RADARSAT-2, two dual-channel SAR satellites will be launched in the next months. Both sensors allow for detecting moving objects, and, by this, enable traffic monitoring from space. This paper revises the theoretical background of traffic monitoring with space-based SARs and presents concepts for the TerraSAR-X traffic monitoring system. Compared to previous work an extensive analytical and empirical accuracy analysis is included for both vehicle detection and velocity estimation. The accuracy analysis includes a theoretical accuracy evaluation and a validation with real data.

1 INTRODUCTION

Since the launch of new optical satellite systems, e.g. Ikonos and QuickBird, satellite imagery with 1-meter resolution or higher is commercially available and a number of approaches have been developed to detect or track vehicles in this imagery (see e.g. references in (Leitloff et al., 2005)). Traffic monitoring based on optical satellite systems, however, is only possible at daytime and cloud-free conditions. Two high-resolution Spaceborne RADAR systems, TerraSAR-X (Germany) and RADARSAT-2 (Canada), which will be launched this year will overcome this limitations. Yet there are other difficulties inherent in the SAR imaging process that must be solved to design a reasonably good approach for traffic monitoring using spaceborne Radar.

The task of detecting moving vehicles with SAR sensors (ground moving target indication (GMTI)) has been addressed in several scientific publications. The method of choice in GMTI is to use a Radar or SAR sensor with at least 3 channels and use space-time adaptive processing (STAP) for target detection. Further reference to that topic can be found e.g. in (Klemm, 1998, Livingstone et al., 2002, Gierull, 2004). Unfortunately, civilian space borne SAR systems with 3 or more channels are currently not available. The upcoming TerraSAR-X mission as well as the Canadian RADARSAT-2 mission will be equipped with a single channel SAR that can be switched to an experimental mode with two channels to enable along-track interferometric applications like traffic monitoring. Although the use of a 2-channel system is not optimal for detecting vehicles, some methods exist that allow detection under certain conditions. The classical approach to do so is to use the displaced phase center array (DPCA) method. Along-track interferometry (ATI) is another method that can be used. The issue of detecting moving targets using ATI is for instance discussed in (Gierull, 2001, Sikaneta and Gierull, 2005). In (Gierull, 2002) special emphasis is put on the probability density functions associated with this detection. The influence of vehicle acceleration is discussed in (Sharma et al., 2006). Traffic monitoring from space is quite rare so far. But as shown in (Breit et al., 2003, Meyer and Hinz, 2004, Meyer et al., 2005) first endeavors have already been carried out.

Based on a revision of the effects of moving objects in SAR Data we present a concept of detection and velocity estimation of vehicles, thereby considering the restrictions of *civilian* SAR satellite systems. The main focus of this paper, finally, lies on the performance characterization of the main components of this concept, in order to predict and validate the expected results of the system for TerraSAR-X. The performance analysis includes both a theoretical accuracy evaluation and a validation with real airborne SAR data.

2 MOVING OBJECTS IN SPACEBORNE SAR IMAGES

Before outlining the concepts for vehicle detection and velocity estimation we briefly summarize the effects of moving objects in spaceborne SAR images. Here, only the resulting formulae are included; a derivation of the formulae can be found, e.g. in (Meyer et al., 2005), while a comprehensive overview on SAR image processing is given in (Cumming and Wong, 2005).

2.1 Object Motion Effects in SAR — A Summary

The position of a Radar transmitter on board a satellite is given by $P_{sat}(t) = [x_{sat}(t), y_{sat}(t), z_{sat}(t)]$ with x being the along-track direction, y the across-track ground range direction and z being the vertical. A point scatterer is assumed to be at position $P_{mover} = [x_{mover}(t), y_{mover}(t), z_{mover}(t)]$, and the range to this arbitrarily moving and accelerating point target from the radar platform is defined by $R(t) = P_{sat}(t) - P_{mover}(t)$.

Omitting pulse envelope, amplitude, and antenna pattern for simplicity reasons, and approximating the range history $R(t)$ by a parabola, the measured echo signal $u(t)$ of a static point scatterer can be written as

$$u_{stat}(t) = \exp\{j\pi FMt^2\} \tag{1}$$

with

$$FM = -\frac{2}{\lambda}\frac{d^2}{dt^2}R(t) = -\frac{2}{\lambda R}v_{sat}v_B \tag{2}$$

being the frequency modulation (FM) rate of the azimuth chirp. Azimuth focussing of the SAR image is performed using the matched filter concept(Bamler and Schättler, 1993, Cumming and

Wong, 2005). According to this concept, an optimally focused image is obtained by complex-valued correlation of $u_{stat}(t)$ with the filter $s(t) = \exp\{-j\pi FMt^2\}$. To construct $s(t)$ correctly, the actual range history of each target in the image, and thus, the position and motion of sensor and scatterer, must be known. Usually, the time dependence of the scatterer position is ignored yielding $P_{mover}(t) = P_{mover}$. This concept is commonly referred to as *stationary-world matched filter* (SWMF). Because of this definition, a SWMF does not correctly represent the phase history of a significantly moving object, which eventually results in image deteriorations.

We first evaluate targets moving with velocity v_{y0} in *across-track* direction. This movement causes a change of range history proportional to the projection of the motion vector into the line-of-sight direction of the sensor $v_{los} = v_{y0} \cdot sin(\theta)$, with θ being the local incidence angle. In case of constant motion during illumination the change of range history is linear and causes an additional linear phase trend in the echo signal. The resulting signal of an object moving in line-of-sight direction with velocity v_{los} is consequently:

$$u(t) = \exp\{j\pi FMt^2\} \cdot \exp\{-j\frac{4\pi}{\lambda}v_{los}t\} \qquad (3)$$

If $u(t)$ is focused with the SWMF $s(t)$ defined above, the linear phase term in Equ. (3) is not compensated for, and remains in the phase of the focused signal. This linear phase term corresponds to a shift of the signal in space domain, which is given by

$$\Delta az = -R\frac{v_{los}}{v_{sat}} \quad [m] \qquad (4)$$

According to Equ. (4), across-track motion results in an along-track displacement of the moving object. It is displaced in flying direction if the object moves towards the sensor and reverse to flying direction if the movement is directed away from the sensor. When inserting the TerraSAR-X parameters into the above formulae, one can see, that moving vehicles are displaced significantly from their real position even for small across-track velocities (about $1\,km$ for $50\,km/h$ at 45° inc. angle). This effect strongly hampers the recognition of cars in TerraSAR-X images as their position is not anymore related to semantic information, e.g. streets. A detailed analysis and illustration of these effects is given in (Meyer et al., 2005).

The target is now assumed to move with velocity v_{x0} in *along-track*. In this case the relative velocity of sensor and scatterer is different for moving objects and surrounding terrain. Thus, along-track motion changes the frequency modulation (FM) rate of the received scatterer response. The FM rate FM_{mt} of a target moving in along-track with velocity v_{x0} is defined by $FM_{mt} = FM\left(1 - \frac{v_{x0}}{v_B}\right)$. If the echo signal of this object is focused with a SWMF $s(t)$, a quadratic phase component remains in the focused signal leading to a spread of the signal energy in time or space domain. The width of the focused peak as a function of the object's along-track velocity v_{x0} can be approximated by

$$\Delta t \approx 2T_A\sqrt{\frac{v_{sat}}{v_B}}\frac{v_{x0}}{\sqrt{v_{sat}v_B}} \quad [s] \qquad (5)$$

with T_A being the aperture time. Interpretation of Equation (5) shows that a moving vehicle is smeared by twice the distance it moved along track during the illumination time T_A. Note that the approximation in Equation (5) only holds for $v_{x0} \gg 0$. As the backscattered energy of the moving object is now spread over a larger area the peak value of the signal drops down. Using the parameter set of TerraSAR-X, it is obvious that blurring and peak power decrease are quite drastic. The strong blurring distributes the backscattered energy and results in a drop of $50\,\%$ peak power or more if $v_{x0} \geq 15\,km/h$ (Meyer et al., 2005). Thus, nearly all ground moving targets suffer from energy dispersion, which decreases the signal-to-clutter ratio and renders target detection more difficult.

Similar analyses are conducted for first order accelerations. Such effects not only appear if drivers physically accelerate or brake but also along curved roads, as the object's along-track and across-track velocity components vary during illumination time. The analysis is based on a third order Taylor series expansion of the range $R(t)$ to an accelerating and isotropic point scatterer. The scatterer is assumed to be at position $(0, y_0, 0)$ at azimuth time $t = 0$ and to move with velocity $(v_{x0}, v_{y0}, 0)$ and acceleration $(a_x, a_y, 0)$. With R_0 being the range at azimuth time $t = 0$ the third order Taylor series expansion of $R(t)$ calculates to:

$$R(t) \approx R_0 + \tfrac{y_0 v_{y0}}{R_0}t - \tfrac{1}{2R_0}\left[\tfrac{y_0 v_{y0}(v_{x0}-v_{sat})^2 + y_0 v_{y0}^3}{R_0^2}\right]t^3 +$$
$$\tfrac{1}{2R_0}\left[y_0 a_{y0}\left(1 - \tfrac{y_0^2}{R_0^2}\right) + (v_{x0} - v_{sat})a_{x0}\right]t^3 +$$
$$\tfrac{1}{2R_0}\left[(v_{x0} - v_{sat})^2 + v_{y0}^2\left(1 - \tfrac{y_0^2}{R_0^2}\right) + y_0 a_{y0}\right]t^2 \qquad (6)$$

It can be seen in Equation (6) that acceleration components appear in the quadratic and the cubic term of the Taylor series expansion. The acceleration in across-track direction (a_y) causes a quadratic phase component, which results in a spread of the signal energy in time or space domain. Considering the TerraSAR-X system parameters it comes clear that image degradation due to across-track accelerations is significant for $a_y > 1\,\frac{m}{s^2}$, which is commonplace for traffic on roads or highways (Meyer et al., 2005). On the other side, along-track acceleration a_x appears only in the cubic term of Equation (6) and results in an asymmetry of the focused point spread function. For TerraSAR-X, this effect is very small even for unrealistic accelerations, and can be neglected.

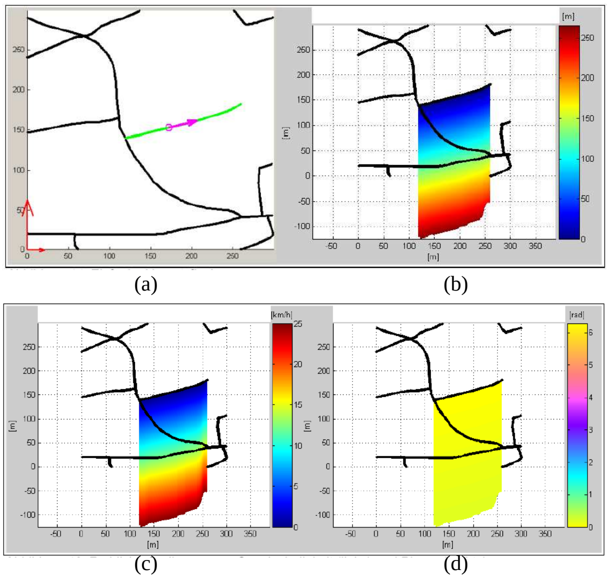
2.2 Detection Approaches

On one hand, all the above described effects of moving objects hinder the detection of cars in conventionally processed SAR images. On the other hand, these effects are mainly deterministic and can be exploited to not only detect vehicles but also measure their velocity. Our system for moving object detection consists of two major components: a detection and a velocity estimation component. Both components make use of a-priori knowledge in form of a road database and expectation values for the aspect-angle dependent Radar cross-section of vehicles. In the following sections we discuss the approaches employed in the system in more detail.

In order to detect moving objects in SAR data one has to predict their appearance in the image. Thus, the main tasks to solve are the *estimation* of the blurring, the displacement, and the interferometric phase values associated with the particular moving object. The solution to this typical inverse problem can be facilitated when incorporating a priori knowledge about the appearance, location, and velocity of vehicles. Hence, we will first turn to the integration of a priori knowledge (Sect. 2.2.1) before describing different detection approaches in Sects. 2.2.2, 2.2.3 and 2.2.4.

2.2.1 Integration of A-priori Information Assuming objects being point scatterers and given the SAR- and platform parameters, the displacement effect in the along-track direction can be predicted when real position, velocity, and motion direction of the vehicle are known. Because of the functional relation of interferometric phase and object velocity in across-track direction, also the interferometric phase of a displaced moving object can be derived (see below).

In our case, road network databases serve as basic source for acquiring a priori knowledge. Typically, these databases contain road axes in form of polygons and attributes like road class, road width, maximum velocity, etc. attached to each polygon. Using this information a number of ”maps” representing the a priori information can be derived (i.e. displacement map, velocity map, and interferometric phase map). Figure 1 shows an example for the different maps derived for a single road segment.



(a) (b)

(c) (d)

Figure 1: Example for maps derived from a single road segment associated with travelling direction (see (a)): (b) Displacement map, (c) velocity map, (d) phase map.

Besides the information about the phase, also a priori information about the vehicle’s radar cross section strongly supports detection. As it is well known, significant variations of radar cross section exist over different aspect angles of cars. An example of radar cross section variations as a function of aspect angle α for a Volkswagen Golf car derived from experimental measurements of DLRs airborne SAR system E-SAR is shown in Fig. 2. The analysis of the RCS curve shows that cars have quite high RCS values if their front, rear or side faces the sensor. RCS values for the angles around 45° and 135° are significantly lower. It also can be seen that the RCS is subject to high variation even for small changes of aspect angle. Such information is incorporated into the detection scheme with the help of a road database, since—given the sensor and platform parameters—the aspect angle under which a car must have been illuminated by the sensor can be calculated for each road segment.

2.2.2 Along-Track Interferometry In along-track interferometry (ATI) an interferogram I is formed from two original SAR images acquired with a short time lag in along-track direction. The interferogram phase can be related to object motion by:

$$\psi = \frac{4\pi}{\lambda}\Delta R = \frac{4\pi}{\lambda} v_{los} t = \frac{4\pi}{\lambda} v_{los} \frac{\Delta l}{v_{sat}} \qquad (7)$$

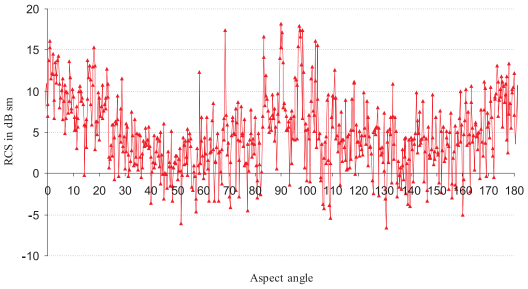


Figure 2: Radar cross section depending on aspect angle. Experimental data of a VW Golf car in X-band.

where t is the temporal separation of the aquisitions defined by the satellite motion and the effective distance Δl between the phase centers of the two antennas. Since both interferometric phase ψ and azimuth displacement Δaz are caused by across-track motion, an analytic relation between both measurements can be established:

$$\Delta az = -R\frac{v_{los}}{v_{sat}} = -R\psi\frac{\lambda}{4\pi\Delta l} \qquad (8)$$

To design a constant false alarm rate (CFAR) detection scheme, the probability density distributions of vehicles and background in interferometric data need to be known. Here, we follow the derivation presented in (Lee et al., 1994) and (Joughin et al., 1994). For all stationary targets the interferometric phase values are assumed to be statistically distributed around the expectation value $E[\psi] = 0$. Using the underlying assumption of jointly Gaussian-distributed data in the two images, the joint probability density function (pdf) $f_c(\eta,\psi)$ of amplitude and phase of an interferogram is given by:

$$f_c(\eta,\psi) = \frac{2n^{n+1}\eta^n}{\pi\Gamma(n)\left(1-|\rho|^2\right)} \exp\!\left(\frac{2n\eta|\rho|\cos(\psi)}{1-|\rho|^2}\right) K_{n-1}\!\left(\frac{2n\eta}{1-|\rho|^2}\right) \qquad (9)$$

where n is the number of looks (effectively the amount of averaging), $\Gamma(\cdot)$ is the gamma function and $K_n(\cdot)$ is the modified Bessel function of the nth kind. For medium resolution SAR the jointly Gaussian assumption has been validated for most agricultural and vegetated areas (Ulaby and Dobson, 1989). As outlined in Sect. 2.2.1 it is possible to derive expectation values for position, interferometric phase, and aspect-dependent radar cross section of vehicles using ancillary data. Hence, from these data also a pdf for ”clutter+mover” $f_{c+m}(\eta,\psi)$ should be established. An approximation valid for $n \gg 1$ has been derived in (Gierull, 2002) and is given by:

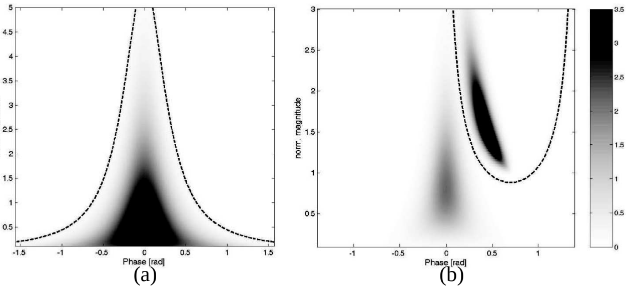
$$f_{c+m}(\eta,\psi) = \frac{2n^{n+1}\eta\left((\eta-\delta\cos(\zeta))^2+\delta^2\sin(\zeta)^2\right)^{\frac{n-1}{2}}}{\pi\Gamma(n)\left(1-|\rho|^2\right)} \cdot$$

$$\exp\!\left(\frac{2n\rho(\eta\cos(\psi)-\delta\cos(\vartheta))}{1-\rho^2}\right) K_{n-1}\!\left(\frac{2n\sqrt{(\eta-\delta\cos(\zeta))^2+\delta^2\sin(\zeta)^2}}{1-\rho^2}\right) \qquad (10)$$

while the moving target’s signal is assumed to have a peak amplitude β, and with $\delta = \frac{\beta}{\eta}$ and $\zeta = \psi - \vartheta$. Using this approximation as an alternative hypothesis, $f_{c+m}(\eta,\psi)$ allows to define a likelihood ratio to which a threshold can be applied.

Figure 3a) shows a typical example of $f_c(\eta,\psi)$ assuming a coherency of $|\rho| = 0.95$, $n = 1$ and a expected signal amplitude of $E[\eta] = 1$, while Fig. 3b) shows an example of $f_{c+m}(\eta,\psi)$ and a corresponding curve of separation.

2.2.3 Displaced Phase Center Array Method In a similar way one may derive a CFAR detector based on the displaced



(a) (b)

Figure 3: PDFs for background only (a) and background as well as moving objects (b). The dashed line is an example for curve of separation.

phase center array (DPCA) technique, where the two coregistered images are simply substracted, yielding

$$\begin{aligned} I_{DPCA} &= I_1 - I_2 = |I_{DPCA}| \cdot (e^{j\phi_1} - e^{j\phi_2}) \\ &= 2|I_{DPCA}|sin(\frac{\phi_1 - \phi_2}{2})e^{j\left(\frac{\phi_1 - \phi_2}{2} + \frac{\pi}{2}\right)} \quad (11) \end{aligned}$$

Here, only the magnitude $2|I_{DPCA}|sin(\frac{\phi_1-\phi_2}{2})$ of the signal is evaluated for classification. Hence, the above pdf's simplify to a one-dimensional case. The magnitude of I_{DPCA} is high whenever moving objects cause a phase shift between the two images and low if the observed surface elements are stable.

2.2.4 Frequency Modulation Method The approaches outlined so far can only be applied if displacement or interferometric phase occurs at all. This does not happen for objects moving purely in azimuth (along-track) direction. As explained in Sect. 2.1 such vehicles appear defocussed in the image. Focusing these objects is however possible when choosing a FM rate that corresponds to the relative velocity of platform and object. Our strategy for finding the correct FM rate relies on hypothesizing a series of FM rates and analyzing a pixel's ”sharpness function” over these FM rates (see (Weihing et al., 2006) for details). Since blurring occurs only in azimuth direction, searching the correct FM rate for a given pixel reduces to a 2D-problem. Moreover, the known location of roads as well as the expected range of vehicle velocities allow to further restrict the search space to a limited number of FM rates. For extracting the energy peak, we implemented a simple but effective blob detection scheme that analyzes the local curvatures in azimuth- and FM-direction, thereby incorporating a certain amount of smoothing depending on the expected noise level of the images. Combining local curvature maxima and peak amplitude by the geometric mean yields the final decision function, from which the maximum is selected (see (Hinz, 2005) for details). The FM-rate at the extracted peak corresponds to the correct along-track velocity – assuming that target acceleration can be neglected for a first guess.

2.3 Velocity Estimation

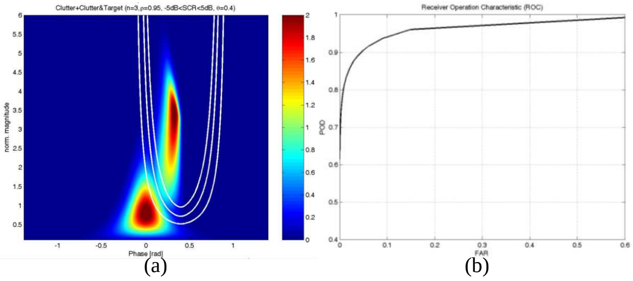
The estimation of the velocity of detected vehicles can be done based on all effects moving objects cause in SAR images and SAR interferograms. Thus, approaches may use *i)* the interferometric phase values, *ii)* the displacement of detected vehicles from their corresponding roads, and *iii)* the along-track defocus caused by along-track motion and/or across-track acceleration. All possible approaches have their advantages and disadvantages and differ in the accuracy of their results (see Sect. 4). The presence of several methods for estimating velocities leads to an overdetermination of the estimation problem. This redundancy might be used to estimate across-track acceleration in addition to the vehicle's velocity. However, this has not yet been realized in the current implementation of the system.

3 PERFORMANCE ANALYSIS OF DETECTION

In order to assess the detection performance for varying scenarios, three different approaches have been used: *i*) an analytical performance analysis based on analytical pdf's and Receiver Operator Characteristic (ROC) curves obtained therefrom (Sect. 3.1); *ii*) a numerical performance analysis derived from simulations (Sect. 3.2); and *iii*) a performance analysis based on data from airborne SAR experiments. The system parameters are tuned to produce images that correspond to the expected space-borne data. In the following, we concentrate mainly on the detection based on the across-track components of vehicle motion. Analyses of the FM-Rate method described in Sect. 2.2.4 are given in (Weihing et al., 2006).

3.1 Analytical Performance Analysis of Detection

The analytical performance analysis is based on the pdf's given in Equs. 9 and 10 and shown in Fig. 3b). These pdf's allow for the calculation of detection and false-alarm probabilities for a given line of separation, i.e. a predefined likelihood ratio, see Fig. 4a). Thereby each parameterization of the pdf's corresponds to different characteristics of background and vehicle appearance. Finally, ROC-curves are obtained when varying the likelihood ratio. Figure 4a) depicts an example for a typical parameterization of the pdf's and Fig. 4b) shows the corresponding ROC curve. However, one has to keep in mind that a number of simplifications have been necessary to obtain the analytical pdf's, most notably the restriction to more than 3 looks and the precondition of Gaussian distributed clutter. Hence, although this approach allows for maximum flexibility, a ROC curve derived this way is only valid for open and rural areas.

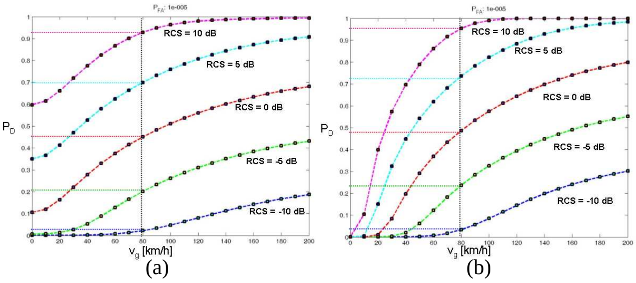


(a) (b)

Figure 4: Analytical Detection Characterization: (a) Analytic PDFs $f_c(\eta, \psi)$, $f_{c+m}(\eta, \psi)$ and varying curves of separation. (b) Corresponding ROC curve.

3.2 Numerical Performance Analysis of Detection

To extent the analysis and to overcome some of the above limitations, a simulator for ATI and DPCA has been developed, which can be parameterized in such a way that a priori information about the interferometric phase and amplitude can be integrated. To generate a random sample, the whole process of data acquisition is simulated for both vehicles and clutter, i.e., the SAR-Data-Acquisition process, multilooking if required, and the generation of interferograms. Then, for each set of random samples a histogram is computed substituting the probability density functions. As above, to evaluate the performance of the detectors, a threshold is varied and the probability of detection and probability of false alarm are determined for each step of this variation. Figure 5 illustrates the detection probability using ATI (a) and DPCA (b) over different vehicle velocities (i.e. phases) for certain vehicle brightnesses as well as fixed background and false alarm rate. As can be seen, for low velocities and bright vehicles ATI delivers generally better results while for faster vehicles it is outperformed by DPCA. The reason for this behavior is that DPCA purely relies on the interferometric phase, i.e., for low phase values the detection is strongly influenced by noise, which leads to the significant decrease of performance. In contrast, ATI makes also use of the amplitude so that, for low velocities, one additional feature is still left to detect a vehicle.



(a) (b)

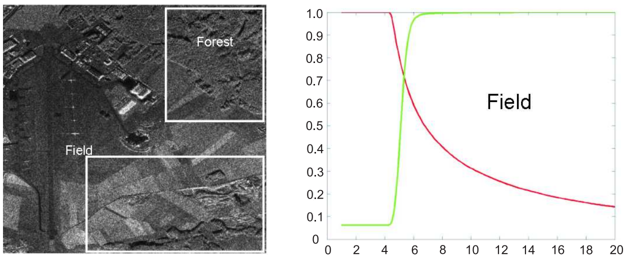
Figure 5: Numerical Detection Characterization: Detection Probabilities for given Background Clutter (bushes) and fixed False Alarm Rate (10e-5) calculated for varying vehicle brightnesses (RCS). (a) Results for ATI. (b) Results for DPCA.

3.3 Performance Analysis Based on Airborne Data

The validity of the simulation results has been assessed using real data of flight campaigns. Besides of this, tests on real data sets also allow to discover bottlenecks of the techniques employed and to reveal unforeseen problems. An additional goal is to simulate TerraSAR-X data for predicting the performance of the extraction procedures. To this end, an airborne Radar system has been used that has been modified so that the resulting raw data is comparable with future satellite data of TerraSAR-X. We followed two different ways of assessment: *i*) using real background data and, to have a ”ground-truth”, vehicles that have been artificially impainted into the background (Sect. 3.3.1) , and *ii*) detection of real vehicles in scenes for which optical data has been simultaneously acquired.

3.3.1 Background Data and Impainted Vehicles Figure 6a) shows a larger SAR scene composed of different types of background. In two test areas, vehicles in form of point targets have been impainted. The appearance of a vehicle (amplitude and phase) has been randomized using a random generator. Since in this case ground-truth is available one is able to obtain completeness and correctness curves when varying the detection threshold, which replace the detection and false alarm rates before. Figure 6b) shows these curves for a typical image background using a fixed vehicle velocity, statistically distributed vehicle brightness and DPCA as detection method. Although not being directly comparable with Fig. 5b), the typical behavior of DPCA is confirmed also by this evaluation, i.e., there is a striking lock-in of the quality of the results depending on the detection threshold.

3.3.2 Vehicle Detection in Airborne Data In the following, results of a flight campaign are shown during which images over real-life traffic scenarios on highways were acquired. To evaluate the results of SAR-based vehicle detection, time series of aerial photographs have been taken – almost synchronized with the SAR acquisition.



(a) (b)

Figure 6: Detection Characterization based on airborne background image (a) and impainted vehicles with $RCS = 3dB \pm 8dB$ and phase according to 65km/h: Curves for completeness (red) and correctness (green) for agricultural area obtained by DPCA with varying thresholds.

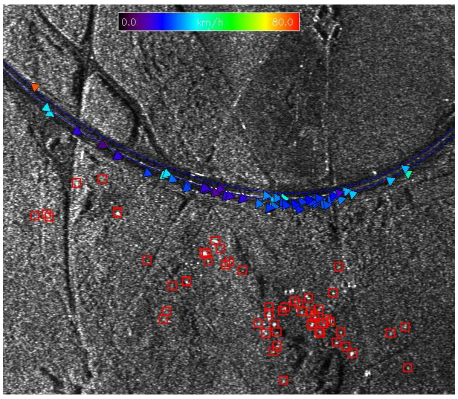
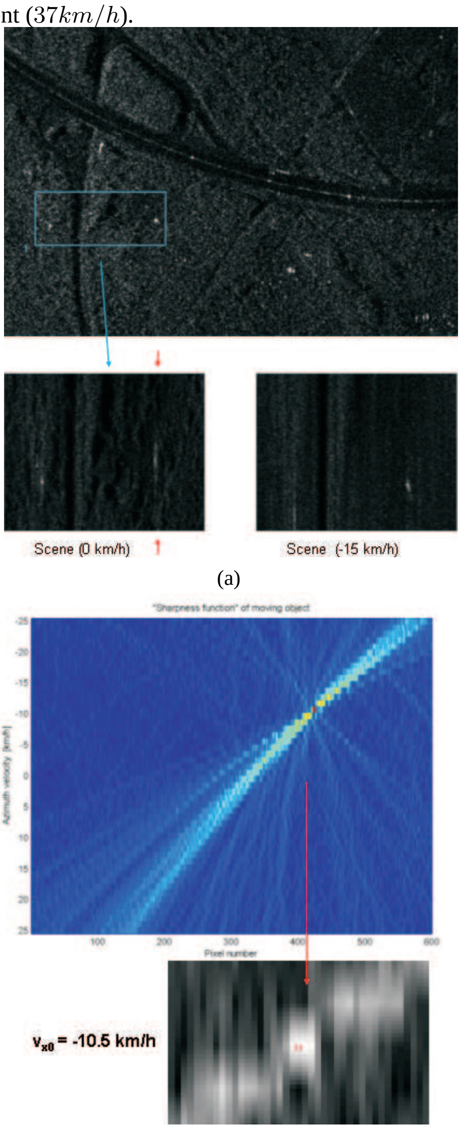


Figure 7: Experiments with airborne SAR: Detection results and velocity estimation for a dense traffic scenario

First encouraging results have been achieved with the system described above, although we have to admit that too few scenes have been processed up to now to give reliable and statistically confirmed statements about the system’s performance. The used experimental proccesing system includes a combination of an ATI and a DPCA detector, and allows for an automatic integration of a-priori knowledge (NavTeq road data). It performs velocity estimation based on ATI phase and on along-track displacement. The incorporated road data not only enables displacement measurements but also the prediction of displacement intervals and thus a limitation of the search space. Typical results are depicted in Figure 7. It shows the detector performance for rather dense traffic. Although simultaneously acquired optical images are available for this scene, it was–due to unknow time delays–unfortunately not possible to match the car reference data form optical images uniquely to the detection results. Yet the evaluation of these results based on traffic flow parameters has shown that flow parameters can be derived precisely, although the completeness of detected cars is only moderate ((Suchandt et al., 2006)).

Figure 8 illustrates the detection of vehicle by FM-rate variation. The azimuth direction points from bottom to top, thus, along-track velocity components of vehicles travelling along the main road in the center of the image are quite small and moving vehicles are both blurred and displaced. At the bottom of Figure 8 a) the marked image patch is focused with FM rates corresponding to $0km/h$ and $15km/h$ (assuming absence of acceleration). As can be seen, the background of the image blurs for the second case, while one bright point gets sharp (marked by red arrows). Figure 8 b) shows the corresponding FM-slice, the detected peak, and an estimated along-track velocity of approx. $10km/h$ assuming zero acceleration. Considering a road orientation of 15 degree the vehicle velocity computes to approximately $40km/h$, which fits reasonably well to the velocity computed from the displacement ($37km/h$).



(a)

(b)

Figure 8: (a) Image patch (blue rectangle) focused with two different FM-rates (bottom). Red arrows mark azimuth line in which the sharpened point lies. (b) FM slice computed for this azimuth line (top) and detected peak (bottom).

4 PERFORMANCE ANALYSIS OF VELOCITY ESTIMATION

For each of the three approaches for velocity estimation, i.e. *i)* via interferometric phase, *ii)* via displacement, and *iii)* via along-track blurring, the corresponding accuracy values are derived, and at the end of this section, an example for accuracy when combining approaches is given.

4.1 Velocity Estimation based on the Interferometric Phase

The interferometric phase allows for a direct access to the objects line-of-sight velocity component without the need of auxiliary information. Still, information about the relative orientation of the road axis corresponding to the particular vehicle is needed in order to derive the real heading velocity of vehicles from their line-of-sight motion. If we assume that a detected vehicle acts as point scatterer, the standard deviation σ_ψ of its interferometric phase is defined by

$$\sigma_\psi - \psi \approx \frac{1}{\sqrt{2 \cdot SCR}} \tag{12}$$

with *SCR* being the signal-to-clutter ratio of a point like target. SCR values can be determined based on RCS measurements of vehicles, which are shown in Sect. 2.2.1. Given Equation (12), the standard deviation of the derived across-track velocity estimate $\hat{v}_y^\psi$ results in

$$\sigma_{\hat{v}_y^\psi} = \frac{sin(\theta_{inc}) \cdot \lambda \cdot v_{sat}}{\sqrt{2 \cdot SCR} \cdot 4\pi \cdot \Delta l} \tag{13}$$

Given the system parameters of TerraSAR-X and assuming a SCR of 5 dB we get a standard deviation $\sigma_{\hat{v}_y^\psi}$ of approximately 30 km/h for the center of the TerraSAR-X swath. Clearly, for an analysis of traffic behavior and traffic dynamics, this accuracy level is only marginally sufficient.

4.2 Velocity Estimation from Along-track Displacement

Besides of the above mentioned approach, the heading velocity of a moving vehicle $\hat{v}_{mt}$ can be derived by measuring its along-track displacement from its corresponding road segment. The functional relation is given by

$$\hat{v}_{mt}^{\Delta az} = \frac{\hat{\Delta} az \cdot v_{mt}}{R \cdot sin(\hat{\alpha}_{road}) \cdot sin(\theta_{inc})} \tag{14}$$

where $\hat{\Delta} az = |\hat{x}_{road} - \hat{x}_{mt}|$ is the along-track displacement. The accuracy $\sigma_{\hat{v}_{mt}^{\Delta az}}$ of the velocity estimate is a function of the quality of the displacement measurement $|\hat{x}_{road} - \hat{x}_{mt}|$, and the accuracy of the road's heading angle $\hat{\alpha}_{road}$ relative to the satellite track. $\sigma_{\hat{v}_{mt}^{\Delta az}}$ is calculated by error propagation.

$$\sigma_{\hat{v}_{mt}^{\Delta az}} = \sqrt{\left(\frac{\partial \hat{v}_{mt}^{\Delta az}}{\partial x_{obj}}\right)^2 \sigma_{x_{obj}}^2 + \left(\frac{\partial \hat{v}_{mt}^{\Delta az}}{\partial x_{road}}\right)^2 \sigma_{x_{road}}^2 + \left(\frac{\partial \hat{v}_{mt}^{\Delta az}}{\partial \alpha_{road}}\right)^2 \sigma_{\alpha_{road}}^2} \tag{15}$$

From empirical evaluations of the peak detection approach we assessed the accuracy of the target's along-track position to be $\sigma_{x_{obj}} = 1$ m. The standard deviation of the road axis position $\sigma_{x_{road}}$ of the NavTeq data was estimated by comparing the vector data with precisely geocoded aerial images. The mean distance of the NavTeq axes from their corresponding reference was determined to be $\sigma_{x_{road}} = 3.5$ m (this result holds for high level roads like motorways). From this value, and by assessing the average length of the NavTeq polygon pieces, the accuracy of the road heading angle $\sigma_{\alpha_{road}}$ was deduced. For motorways its standard deviation results in $\sigma_{\alpha_{road}} = 2°$.

The accuracy of velocity estimates $\sigma_{\hat{v}_{mt}^{\Delta az}}$ is derived by inducting these empirical error measures into Equation (15). The resulting error $\sigma_{\hat{v}_{mt}^{\Delta az}}/v_{mt}$ is shown in Figure 9 as a function of heading angle α_{road} and normalized with the real target velocity v_{mt}. It can be seen from Figure 9 that the vehicles heading velocity v_{mt} can be estimated with a high accuracy of $\sigma_{\hat{v}_{mt}^{\Delta az}}/v_{mt} \leq 10\%$ if they were moving on roads with a heading angle of $\alpha_{road} \geq 4°$. For roads running nearly in along-track direction ($\alpha_{road} < 4°$) this approach fails to provide reliable velocity measures.

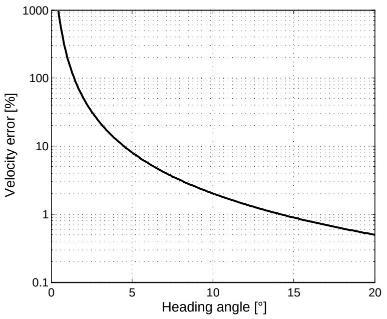


Figure 9: Relative velocity error $\sigma_{\hat{v}^{\Delta az}_{mt}}/v_{mt}$ estimated from along-track displacement as a function of heading angle α_{road}. Note the logarithmic scale.

4.3 Velocity Estimation from Along-track Blurring

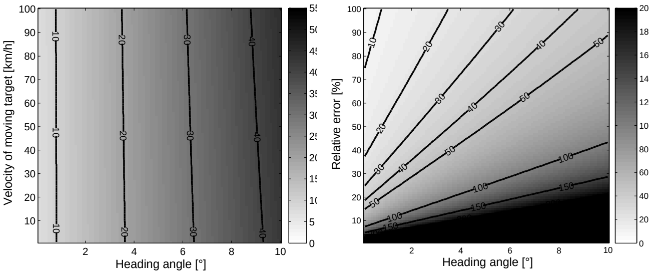
Both of the already presented estimation methods fail to give a reliable velocity estimate for vehicles moving almost in along-track direction. To fill the gap we propose to use the along-track blurring effect for estimating along-track velocities. The functional dependence of the velocity estimate on unknown or uncertain parameters is given by:

$$\hat{v}^x_{mt} = -\sqrt{(v_{sat} - v_{mt}) \cdot cos(\hat{\alpha})^2 + y_0 \cdot \hat{a}_y \cdot sin(\hat{\alpha})} + v_{sat} \tag{16}$$

As explained in Section 2.1 both along-track velocity $v_x = v_{mt} \cdot cos(\alpha)$ and across-track acceleration a_y give rise to peak broadening in along-track. Usually, it is assumed that the acceleration of vehicles is zero during the time of illumination. As a consequence, actual occurring across-track accelerations introduce errors to the velocity estimates. According to empirical studies based on inertial navigation system measurements with cars driving on city streets and highways, accelerations up to $a_y = 2\ m/s^2$ are likely to happen in common traffic scenarios. Thus, we assume $\sigma_{a_y} = 2\ m/s^2$ as a ”worst case” error source for the following calculations. Besides of possible acceleration, the standard deviation of the road heading angle $\sigma_{\alpha_{road}} = 2^\circ$ influences the accuracy of the velocity estimate $\sigma_{\hat{v}^x_{mt}}$.

$$\sigma_{\hat{v}^{\delta FM}_{mt}} = \sqrt{\left(\frac{\partial \hat{v}^{\delta FM}_{mt}}{\partial \alpha_{road}}\right)^2 \sigma^2_{\alpha_{road}} + \left(\frac{\partial \hat{v}^{\delta FM}_{mt}}{\partial a_y}\right)^2 \sigma^2_{a_y}} \tag{17}$$

Figure 10a) shows $\sigma_{\hat{v}^{\delta FM}_{mt}}$ as a function of real target velocity v_{mt} and real road heading α_{road}. The standard deviation of the velocity estimate is dominated by acceleration influences and increases with α_{road}. The dependence on v_{mt} is merely a secondary effect.



(a) (b)

Figure 10: a) Standard deviation $\sigma_{\hat{v}^x_{mt}}$ of vehicle velocities estimated from along-track blurring as a function of target velocity v_{mt} and heading angle α_{road}. $\sigma_{\hat{v}^x_{mt}}$ is given in km/h. b) relative velocity error $\sigma_{\hat{v}^x_{mt}}/v_{mt}$.

For $\alpha_{road} < 4^\circ$, i.e. for the heading angles of interest (see Sect. 4.2), the standard deviation $\sigma_{\hat{v}^{\delta FM}_{mt}}$ reaches up to 22 km/h. The relative error of the estimated velocities is indicated in Figure 10b). It indicates that the velocity of slow moving targets cannot be reliably estimated even for very small heading angles α_{road}, whereas the speed of fast moving targets can be estimated with better relative accuracy.

Sections 4.1 to 4.3 show that several possibilities exist to estimate the velocity of moving vehicles from TerraSAR-X data. According to the quality of the velocity estimates the usage of along-track displacement is the most promising approach for a wide range of heading angles α_{road}. If vehicles move nearly in along-track, the accuracy of velocity estimates is fair for all estimators. Still, the use of along-track blurring gives best results.

4.4 Examples

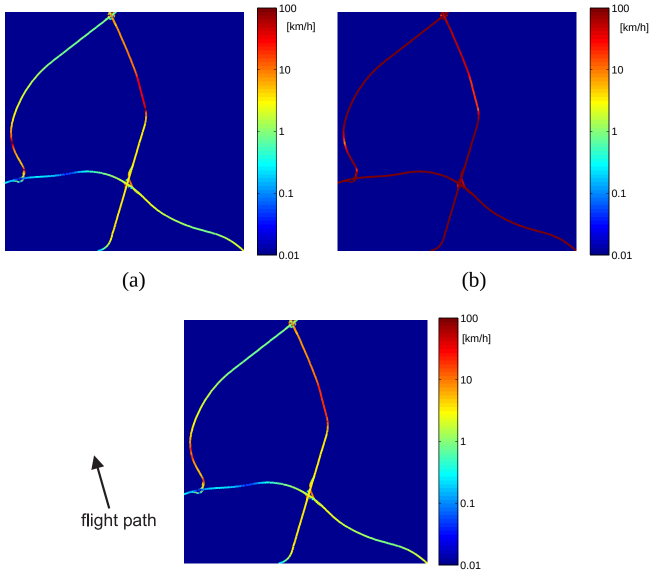
To demonstrate the quality of the velocity estimation for real live scenarios we calculated the expected standard deviation of the estimated velocity $\sigma_{\hat{v}_{mt}}$ for a road network north of Munich. In this area three large motorways are situated which are highly frequented during rush hours. We applied two different velocity estimators to this test, the displacement-based and the blur-based estimator. Real TerraSAR-X orbit and sensor parameters have been used in this simulation and an average speed of 100 km/h was assumed. The orientation of the motorways relative to the choosen TerraSAR-X orbit and the resulting $\sigma_{\hat{v}_{mt}}$ values for both estimators are show in Figures 11*a)* to 11*c)* (the corresponding flight direction of the satellite is indicated as well. The standard deviation of the displacement-based velocity estimate $\sigma_{\hat{v}^{\Delta az}_{mt}}$ is shown in Figure 11*a)* in km/h for all three motorways. It can be seen that vehicle velocities can be estimated with high accuracy for large parts of the road network. However, in areas where the road is oriented nearly in along-track, the estimation error increases dramatically. Figure 11*b)* shows that the second detector, which is based on the blurring of the impulse response, provides better results for this areas. Thus, in order to get an optimal estimation quality, we combine both methods depending on the relative orientation of road and satellite track. The performance of the combined estimator is shown in Figure 11*c)*. With the presented algorithm velocities can be estimated with an accuracy better than 10 km/h for about 80 % of the investigated road network.

5 SUMMARY

A system to detect moving vehicles from TerraSAR-X data and to estimate their respective velocities has been presented. Besides a detailed description of the methods used, performance analyses are shown in addition. The detection of fast moving traffic seams to be very promising, whereas slow moving cars are hard to distinguish from non moving background. The estimation of the velocity of detected vehicles can be done with high accuracy for nearly all possible observation geometries. All approaches are subject to further improvement and a more detailed performance analysis will be presented as soon as the satellite is in its orbit.

ACKNOWLEDGEMENT

The authors would like to thank all members of the DLR/TUM TerraSAR-X traffic processor team for their valuable contribution to this work.



(a)

(b)

(c)

Figure 11: Simulation of $\sigma_{\hat{v}_{mt}}$ for a road network north of Munich ($v_{mt} = 100$ km/h assumed). a) shows the estimation accuracy for a displacement-based detector, b) for a blur-based detector, and c) indicates the estimation quality if both detectors are combined.

REFERENCES

Bamler, R. and Schättler, B., 1993. SAR Data Acquisition and Image Formation. In: G. Schreier (ed.), Geocoding: ERS-1 SAR Data and Systems, Wichmann-Verlag.

Breit, H., Eineder, M., Holzner, J., Runge, H. and Bamler, R., 2003. Traffic Monitoring using SRTM Along-Track Interferometry. In: Proc. of IGARSS03, Vol. 2, pp. 1187–1189.

Cumming, I. and Wong, F., 2005. Digital Processing of Synthetic Aperture Radar Data. Artech House, Boston.

Gierull, C., 2001. Statistics of SAR interferograms with application to moving target detection. Technical Report DREO-TR-2001-045, Defence R&D Canada.

Gierull, C., 2002. Moving target detection with along-track sar interferometry. a theoretical analysis. Technical Report DRDC-OTTAWA-TR-2002-084, Defence R&D Canada.

Gierull, C., 2004. Statistical analysis of multilook SAR interferograms for CFAR detection of ground moving targets. IEEE Transactions on Geoscience and Remote Sensing 42, pp. 691–701.

Hinz, S., 2005. Fast and Subpixel Precise Blob Detection and Attribution. In: Proceedings of ICIP'05, Genua, on CD.

Joughin, I., Winebrenner, D. and Percival, D., 1994. Probability density functions for multilook polarimetric signatures. IEEE Transactions on Geoscience and Remote Sensing 32(2), pp. 562–574.

Klemm, R., 1998. Space-time adaptive processing. The Institute of Electrical Engineers, London, UK.

Lee, J.-S., Hoppel, K. and Mango, S., 1994. Intensity and Phase Statistics of Multilook Polarimetric and Interferometric SAR Imagery. IEEE Transactions on Geoscience and Remote Sensing 32(5), pp. 1017–1028.

Leitloff, J., Hinz, S. and Stilla, U., 2005. Automatic Vehicle Detection in Space Images Supported by Digital Map Data. In: ISPRS/DAGM joint Workshop on City Models, Road Databases, and Traffic Monitoring, CMRT05, Vienna.

Livingstone, C.-E., Sikaneta, I., Gierull, C.-H., Chiu, S., Beaudoin, A., Campbell, J., Beaudoin, J., Gong, S. and Knight, T.-A., 2002. An Airborne Sythentic Apertur Radar (SAR) Experiment to Support RADARSAT-2 Ground Moving Target Indication (GMTI). Canadian Journal of Remote Sensing 28(6), pp. 794–813.

Meyer, F. and Hinz, S., 2004. The Feasibility of Traffic Monitoring with TerraSAR-X - Analyses and Consequences . In: International Geoscience and Remote Sensing Symposium, pp. 1502–1505.

Meyer, F., Hinz, S., Laika, A. and Bamler, R., 2005. A-Priori Information Driven Detection of Moving Objects for Traffic Monitoring by Spaceborne SAR. In: Proc. of CMRT05, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, 3/W24, pp. 89–94.

Sharma, J., Gierull, C. and Collins, M., 2006. Compensating the effects of target acceleration in dual-channel SAR-GMTI. IEE Radar, Sonar, and Navigation.

Sikaneta, I. and Gierull, C., 2005. Two-Channel SAR Ground Moving Target Indication for Traffic Monitoring in Urban Terrain. In: Proc. of CMRT05, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI, 3/W24, pp. 95–101.

Suchandt, S., Eineder, M., Mueller, R., Laika, A., Hinz, S., Meyer, F., Breit, H. and Palubinskas, G., 2006. A GMTI processing system for the extraction of traffic information from TerraSAR-X data. European Conference on Synthetic Aperture Radar EUSAR'06 p. in print.

Ulaby, F. and Dobson, M., 1989. Handbook on Radar Scattering Statistics for Terrain. Artech House, Boston.

Weihing, D., Hinz, S., Meyer, F., Laika, A. and Bamler, R., 2006. Detection of along-track ground moving targets in high resolution spaceborne SAR images. International Archives of Photogrammetry and Remote Sensing, Congress Enschede, Commission VII.

OVERVIEW AND EXPERIENCES IN AUTOMATED MARKERLESS IMAGE ORIENTATION

Fabio Remondino[1], Camillo Ressl[2]

[1] Institute for Geodesy and Photogrammetry, ETH Zurich, Switzerland, fabio@geod.baug.ethz.ch
[2] Institute of Photogrammetry and Remote Sensing, TU Vienna, Austria, car@ipf.tuwien.ac.at

KEY WORDS: Features, Orientation, Adjustment, Matching, Automation

ABSTRACT

Automated image orientation is still a key problem in close-range photogrammetry, in particular if wide baseline images are employed. Nowadays, within the image-based modeling pipeline, the orientation step is the one which could be fully and reliably automated, exploiting the potentiality of computer and image processing algorithms. In this paper, we summarize recent developments in this field and apply them in three different workflows to automatically extract markerless tie points from close-range images of different types (video sequence, large and wide baseline images). Furthermore we compare the results obtained from bundle block adjustment using the automatic tie points with the results obtained by manual measurements and show how the accuracies of the automatic tie point extraction can further be improved by including least squares matching techniques.

1. INTRODUCTION

Image orientation is the first and thus very important step within the 3D modeling pipeline. To achieve the best results together with accuracy estimates the image orientation is usually performed by means of a bundle block adjustment. In order to speed up the entire modeling pipeline an automation of the orientation step is necessary. This has to operate on two issues: (i) the automatic measurement of tie points (without requiring to stick markers (i.e. signalized targets) on the object) and (ii) the automatic provision of initial orientation parameters for the bundle block adjustment. Whereas (ii) is nowadays easily solved – once the image correspondences are given – using perspective [Cronk et al., 2006] or projective [Hartley and Zisserman, 2001] geometry based formulations of the relative orientation of calibrated and uncalibrated images, the automatic measurement of *markerless* tie points is still a challenging topic especially in close-range images.

Commercial photogrammetric digital stations have some tools for the automated and markerless relative orientation of stereo pairs (HATS from Helava/Leica, ISDM from Z-I, MATCH-AT from Inpho). These systems, however, are mainly designed for (aerial) images acquired in the photogrammetric normal case and thus they generally fail with tilted close-range images. On the other hand, systems able to automatically calibrate and orient a set of close-range images using signalized *coded target* are already available (e.g. iWitness^{TM}). Commercial systems for automatic measurement of *markerless* tie points in close-range images, however, are still missing.

In the literature a lot of work on automated markerless tie point extraction from images can be found [Beardsley et al., 1996; Fitzgibbon and Zisserman, 1998; Pollefeys et al., 1999; Roth and Whitehead, 2000; Nister, 2001; Mayer, 2005]. Most of these point-based systems rely on very short baseline between consecutive frames and work only based on cross-correlation matching procedures. On the other hand, wide baseline images are also receiving great attention [Matas et al., 2002; Lowe, 2004; Georgescu and Meer, 2004; Tuytelaars and Van Gool, 2004]. Although the reported methods seem to be successfully applied on images with very large baselines and with wide intersection angles, still further research in this area is needed. Therefore automated markerless sensor orientation is one of the most attractive and difficult research themes in close-range photogrammetry and computer vision, in particular if wide baseline images are used.

In this paper we show how the current methods can be applied on three different scenarios and how the accuracy can be improved using Least Squares Matching (LSM) [Gruen, 1985]. In this way we check the feasibility and accuracy of automated extraction of markerless correspondences from different data sets. The found correspondences are then imported in a bundle adjustment software to retrieve the orientation parameters. We also compare the automated results with those coming from manual tie point measurements.

We consider the following three scenarios: 'short-range motion', 'long-range motion' and 'wide baseline' images. 'Short-range motion' sequences have a very short baseline between the images and are typically acquired with a video-camera. 'Long-range motion' sequences contain a significant baseline compared to the distance between camera and scene. Wide baseline images present a very large baseline and the intersection angle of homologues rays can be wider than 20-25 degrees. Different tests have been performed, using self-acquired images with a still-video or a video-camera as well as digitized movies without imposing any hard restriction on the camera motion.

In these images natural tie points are automatically extracted by different strategies with increasing complexity (depending on the base-to-distance ratio). This is to show that, depending on the application, simpler or more involved extraction and matching routines should be applied.

The workflows in these three different scenarios require no or only little manual intervention. The main sources for manual measurements are control point measurements, if required, to define the global scale and orientation of the image block and an initial guess about the disparity between the analyzed frames (only for the long-range motion images).

In all these scenarios we work with calibrated cameras or with fixed interior parameters. Although the interior orientation can be determined by self-calibration using the extracted tie points, the geometry of the images acquired for 3D modeling often does not allow for an adequate (i.e. accurate and reliable) determination of the interior parameters. In fact typical sequences include images acquired in only one direction or the imaged object shows no depth variation. A weak determination of the interior orientation also deteriorates the accuracy of the object reconstruction. Therefore, in practical cases, rather than carrying out the calibration and reconstruction simultaneously, it may often be better first to determine the camera calibration (including the non-linear distortion parameters) using the most appropriate network (with or without control points) and afterwards recover the object geometry using the calibration parameters.

2. ORIENTATION OF SHORT-RANGE MOTION IMAGE SEQUENCES

Sequences with a 'short range motion' between consecutive frames present a very small parallax (often in one unique direction) which can be exploited during the search of the correspondences. Usually these images are acquired with a video-camera and all the frames are analyzed. Due to the small camera displacement, given the location of a feature in the reference image, the position of the same feature in the consecutive frame is found with a tracking process, as long as the feature is visible and matchable. When the frame-to-frame displacement is larger than a few pixels, the tracking process must be replaced with a more robust stereo matching. Optical flow techniques and feature tracker methods are widely used in the vision community if sufficiently high time frequency sequences are used. One of the most known feature tracker is the one developed by [Shi and Tomasi, 1994], based on the results of [Lucas and Kanade, 1981] and [Tomasi and Kanade, 1991]. More recent works were presented in [Nister, 2001].

We developed a feature tracker based on interest points and Least Squares Matching (LSM). The procedure tracks interest points through the images according to the following steps:

1. *Extraction of interest points from the first image*: different operators like [Förstner and Gülch, 1987], [Harris and Stephens, 1988], [Heitger et al., 1992] or [Smith and Brady, 1997] can be employed.
2. *Prediction of the position in the next frame*: due to the very short baseline, the images are strongly related to each other and the image position of two corresponding features is very similar. Therefore, for the frame at time t+1, the predicted position of a point is the same as at time t.
3. *Research of the position with cross-correlation*: around the predicted position a search box is defined and scanned to identify the position which has the highest cross-correlation value. This position is considered only as an approximation for the corresponding point in frame at time t+1.
4. *Establishment of the precise correspondence's position*: the approximation found with cross-correlation is refined using LSM, which provides a more precise sub-pixel location of the feature.
5. *Replacement of the lost features with new interest points*: new interest points are extracted in the areas where the matching process has failed or if a feature is no more visible in the image.

At the end of the tracking process, the correspondences which are visible in at least 2 frames are used for the successive bundle adjustment to recover the camera parameters.

Some commercial software is available to automatically solve the feature tracking problem [e.g. 3D Equilizer^{TM}, MatchMover^{TM}, Boujou^{TM}]. They work properly with sequences acquired with a video-camera (high frame-rate and very short baseline) and they can reliably extract the image correspondences if there are no rapid changes of the camera position. They are mainly used in the film industry (movies, advertisements) and industrial design. Once the features are extracted, the camera poses are recovered and a virtual object can be seamlessly inserted into the sequence and rendered for special effects.

2.1 Experiments

A video digitized from the television was used to test the tracking system and recover the camera poses. The images are acquired from a helicopter which is flying above a hotel (Figure 1). The hotel is approx. 30 m wide and 10 m high while the mountain in background is ca. 60 m far away from the hotel.

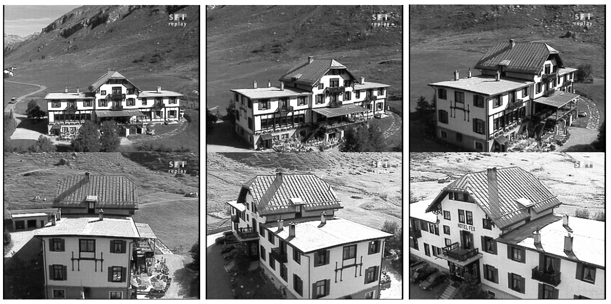


Figure 1: Six images of a sequence digitized from the Swiss TV (SF1) consisting of 240 frames (720x576 pixel).

The tracking process extracted approximately 1900 correspondences which were then used to recover the camera exterior parameters within a bundle adjustment (Figure 2). The project was scaled using the width of an hotel's window.

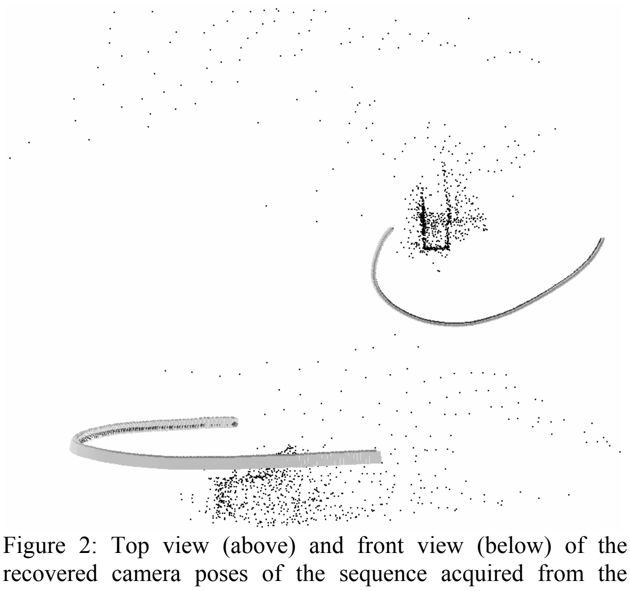


Figure 2: Top view (above) and front view (below) of the recovered camera poses of the sequence acquired from the helicopter.

The final average theoretical precision of the computed object coordinates is $\sigma_x = 0.27$ m, $\sigma_y = 0.15$ m, $\sigma_z = 0.21$ m. As depicted in Figure 2, the smooth trajectory of the helicopter could be successfully recovered. The recovered average distance between two consecutive projection centers is ca 0.5 m. The average distance between the camera and the hotel resulted as ca 45 m while the average distance with the background mountain is ca 120 m.

3. ORIENTATION OF LONG-RANGE MOTION IMAGE SEQUENCES

Long-range motion image sequences contain a significant baseline compared to the distance between camera and scene. They can be acquired with a video-camera (but not all the frames are used) or a still-video camera. The approaches for automatically orienting such image sequences (typically called 'shape-from-video' or 'video-to-3D') [Van Gool and Zisserman, 1996; Fitzgibbon and Zisserman, 1998; Pollefeys et al., 1999; Läbe and Förstner, 2006] require large overlap and good features. In practical situations, such conditions are not always satisfied, due to occlusions, illumination changes and lack of texture.

Our approach, after an initial guess of the average disparity between the images, extracts automatically corresponding points based on the following 5 steps:

1. *Interest points identification*. A set of interest points or corners in each image of the sequence is extracted using detectors like [Förstner and Gülch, 1987], [Harris and Stephens, 1988] or [Heitger et al., 1992]. According to the image size, a threshold on the number of corners extracted is defined and a good point distribution is assured by sub-dividing the images in small patches (e.g. 9x9 pixel on an image of 1600x1200) and keeping only the points with the highest interest value in those patches.

2. *Features matching*. All extracted feature points between adjacent images are matched at first with cross-correlation and then the results are refined using least squares matching (LSM). The point with biggest correlation coefficient is used as approximation for the LSM matching process. Cross-correlation alone cannot always guarantee the correct match while LSM, with patch rotation and reshaping, provides more accurate results. The process returns the best match in the second image for each interest point in the first image.

3. *Epipolar geometry between image pairs*. A pairwise relative orientation and an outlier rejection are performed afterwards. Based on the coplanarity condition, the fundamental matrix is computed with the Least Median of Squares (LMedS) method. Because LMedS estimators solve non-linear minimization problems by minimizing the median of the squared residuals, they are very robust in case of false matches or outliers due to false localisation. The computed epipolar geometry is then used to refine the matching process of step 2, which is now performed as guided matching along the epipolar line.

4. *Epipolar geometry between image triplets*. Not all the correspondences that support the pairwise relative orientation are necessarily correct. In fact a pair of correspondences can support the epipolar geometry by chance (e.g. a repeated pattern aligned with the epipolar line). These kinds of ambiguities and blunders are reduced considering the epipolar geometry between three consecutive images. A linear representation for the relative orientation of three frames is represented by the trifocal tensor T [Shashua, 1997]. T is represented by a set of three 3x3 matrices and is computed using image correspondences without knowledge of the motion or calibration of the cameras. In our process, the tensor is computed with a RANSAC algorithm [Fischler and Bolles, 1981] using 7 correspondences that support two adjacent pairs of images and their epipolar geometry. RANSAC is a robust estimator, which fits a model (tensor T) to a data set (triplet of correspondences) starting from a minimal subset of the data. The found tensor T is used (1) to verify whether the image points are correct corresponding features between three views and (2) to compute the image coordinates of a point in a view, given the corresponding image positions in the other two images. This transfer is very useful in case not many correspondences were found in one view. As result of this step, for each triplet of images, a set of corresponding points, supporting the related epipolar geometry, is recovered.

5. *Tracking image correspondences through the sequence*. After the computation of the trifocal tensor for each consecutive triplet of images, we consider all the overlapping tensors (e.g. T_{123}, T_{234}, T_{345}, ...) and we look for those correspondences which support consecutive tensors. That is, given two adjacent tensors T_{abc} and T_{bcd} with supporting points (p_a, p_b, p_c) and (q_b, q_c, q_d), if (p_b, p_c) in the first tensor T_{abc} is equal to (q_b, q_c) in the successive tensor T_{bcd}, this means that the point in the images a, b, c and d is the same and therefore this point must have the same identifier. Each point is tracked as long as possible in the sequence and the obtained correspondences are used as tie points for a subsequent bundle adjustment.

3.1 Experiments

For the 3D modeling of the empty niche of the Great Buddha of Bamiyan, Afghanistan, five images were acquired with a Sony Cybershot F707. The camera was pre-calibrated in the laboratory. For the image orientation, the tie points were firstly measured semi-automatically by means of LSM and then imported in a bundle adjustment to recover the orientation parameters. Then, the results were used as reference and compared with the results achieved by extracting the tie points automatically.

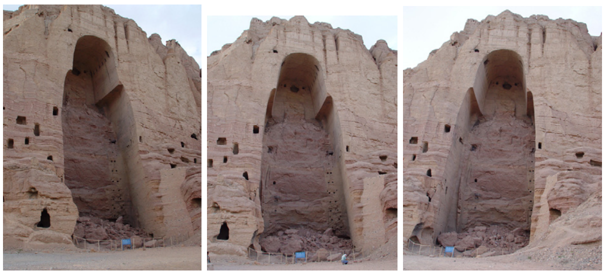


Figure 3: Three (out of five) images (1920x2560 pixel) of the empty niche of the Great Buddha of Bamiyan, Afghanistan, approximately 60 m high and 20 m wide.

The automated procedure, run with the [Förstner and Gülch, 1987] operator, could extract a high number of correspondences (388 points) which were then used for the image orientation.

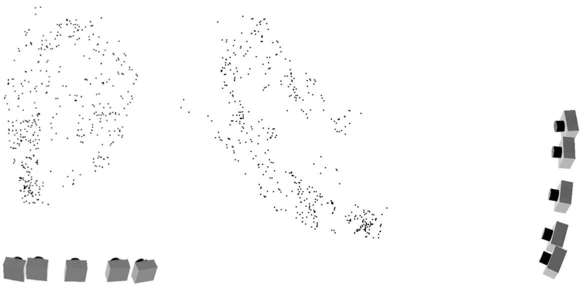


Figure 4: The camera poses recovered using the automatically extracted tie points.

After the adjustment, the estimated theoretical precision of the computed 3D object coordinates turned out to be the same for the manual and for the automatic measurements (Table 1). This suggests that for normal network configurations and good image contents, automated markerless orientation procedures can be as good and reliable as manual measurements.

	Manual	Automated
Numb. of images	5	5
Numb. of tie points	24	388
Points in 2 images	-	253
Points in 3 images	24	135
STD X [m]	0.014	0.012
STD Y [m]	0.017	0.019
STD Z [m]	0.021	0.021

Table 1: Comparison between manual and automated tie point measurements. Number of extracted points and estimated theoretical precisions (STD) are reported.

4. ORIENTATION OF WIDE BASELINE IMAGE SEQUENCES

In some applications, due to acquisition constraints or occlusions, images are acquired from substantially different viewpoints. In this cases, the baseline between the images is very large (e.g. Figure 6) and the intersection angle between homologues rays may be larger than 25 degrees. A standard automated tie point extraction procedure, based on corner detectors, would fail because of the big perspective effects generated by the large camera displacement. Due to these effects interest points (e.g. points or corners simply described with their image location) cannot be correctly matched across the images, as:

– The patches in the search image ought to have large enough size in order to contain enough signal information. Due to the big perspective effects, however, the transformation of large patches between template and search image can no longer be described by a simple affine transformation. A small patch would probably allow the matching process, but it might not contain enough signal information to perform correctly the matching.
– The initialization of an LSM refinement will not work, as LSM requires rather precise approximate values for the parameters.

For these reasons, different researchers tried to solve the challenging problem of automatically orienting widely separated views and interest point detectors have been replaced with region detectors and descriptors [Pritchett and Zisserman, 1998; Baumberg, 2000; Matas et al., 2002; Schaffalisky and Zisserman, 2002; Xiao and Shah, 2003; Georgescu and Meer, 2004; Lowe, 2004; Mikolajczyk and Schmid, 2004; Tuytelaars and Van Gool, 2004; Mikolajczyk et al., 2005]. Indeed while corners might be occluded, regions could still be visible and matchable. Generally local features are extracted independently from the images, then characterized with invariant descriptors and finally matched (by means of the Euclidean or Mahalanobis distance between the descriptor elements). These descriptors (usually a vector of attributes) are invariant under affine transformation and illumination changes and can help in matching homologues points in widely separated views.

[Mikolajczyk and Schmid, 2003] have shown experimentally that the Lowe operator [Lowe, 2004] is the most robust algorithm for wide baseline matching and different applications [Roth, 2004; Roncella et al., 2005; Läbe and Förstner, 2006] have also shown its great potentiality.

For the automated orientation of images acquired with a very wide baseline, a strategy has been developed according to the following steps:

1. *Interest regions identification* by means of Lowe detector and SIFT descriptor [Lowe, 2004];
2. Using the vector of attributes extracted by the descriptor, *matching of corresponding points* (centroid of the extracted regions) is done by searching for points with minimal Euclidean distance between their attribute vectors;
3. *Wrong matches removal* by robust computation of the epipolar geometry (described by the fundamental matrix) between image pairs;
4. *Guided matching* by exploiting the epipolar geometry constraint and increasing the localisation accuracy by applying LSM on previously matched regions.
5. Retrieve the *epipolar geometry between image triplets* by means of the trifocal tensor and further refined checking of extracted correspondences.

As shown in [Remondino, 2006] the orientation derived from correspondences based on region detectors/descriptors has worse accuracy than from correspondences based on point detectors. The reason is that, although the correct regions are found to be corresponding, the centroids of the regions (i.e. the points used for computing the image orientation) might be shifted due to perspective effects. However, refining the centroid using LSM, the location accuracy can be improved. LSM requires approximations for the affine transformation parameters. These can be derived from the regions descriptors, which usually include an ellipse representation, whose parameters (major and minor axis and inclination) are derived from the eigenvalues of the second moment matrix of the intensity gradient [Lindeberg, T., 1998; Mikolajczyk, K. and Schmid, C., 2002]. LSM can cope with different image scales (up to 30%) and significant camera rotations (up to 20 degrees), if good and weighted approximations are used to constraint the estimation in the least squares adjustment. An example is shown in Figure 5: given a detected affine region in the template image and its ellipse parameters, LSM is computed in the search image without and with initial approximations, showing the improved matching results depending on the approximations.

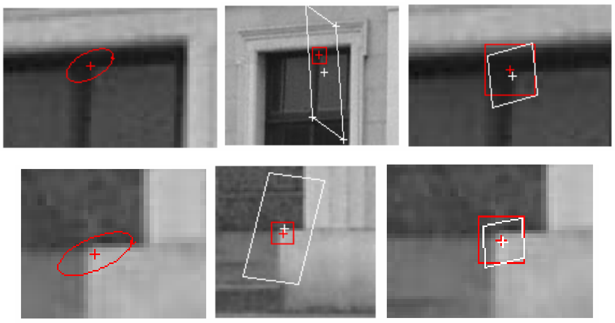


Figure 5: Detected affine regions in the template image (left). Wrong LSM results in the search image (center) with strongly deformed image patches (white patch), initialized by the centroid of the region detector (red patch) and without approximations for the affine transformation parameters. Correct LSM results (right) obtained using the approximations derived from the region descriptors.

4.1 Experiments

A building (Figure 6) has been imaged with two widely separated views. The base-to-distance ratio is approximately 1:0.77 and the camera interior parameters are known. The tie points are automatically extracted as previously described by means of region detectors and descriptors and the extracted correspondences (Figure 7) are used for the relative orientation of the image pair (Figure 8). The final mean RMSE of the image residuals of the computed object coordinates is 0.17 pixels.

A second example consists of three images (Figure 9), representing a Bayon Buddha statue [Gruen et al., 2001], acquired with an analogue Minolta camera and digitized afterwards. Due to the large baseline, there is a very small overlap between the first (A) and the third image (C). The images were firstly processed with the Wallis filter [Wallis, 1976] for radiometric equalization and especially contrast enhancement. The filter enables a strong enhancement of the local contrast by retaining edge details and removing low-frequency information in an image.

For the automated image orientation, the extracted regions are matched between the two adjacent pairs and then the epipolar geometry between the triplet is computed (Table 2).

All the extracted tie points are afterwards imported in a bundle adjustment to retrieve the exterior parameters (Figure 10). A total of 1047 object points are computed and the final RMSE of the image residuals is 0.71 pixels.

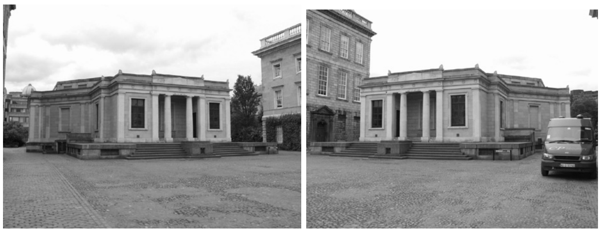


Figure 6: Two widely separated images (courtesy of S. El-Hakim, NRC Canada). The base-to-distance ratio is approximately 1:0.77.

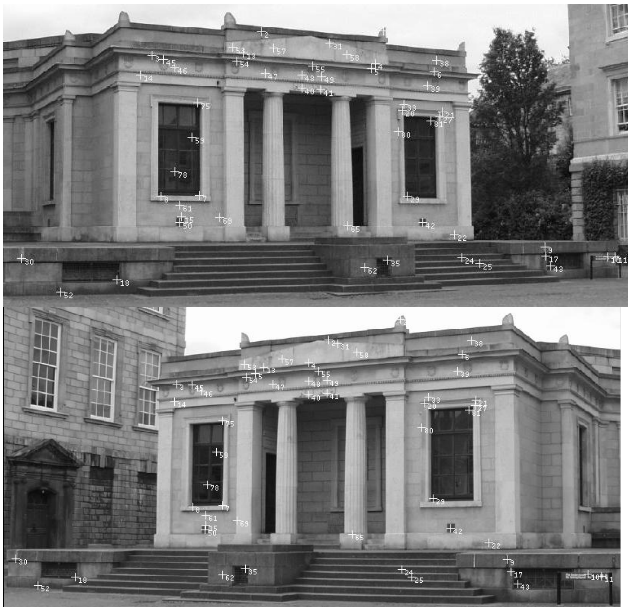


Figure 7: The 55 correspondences, which where automatically extracted using the Lowe region detector and used for the orientation procedure.

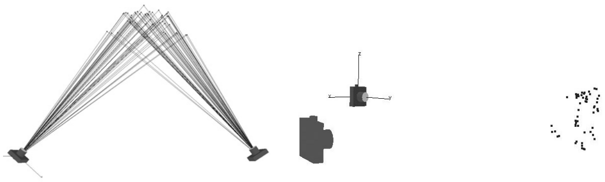


Figure 8: Top and side view of the recovered camera poses of the two widely separated views.

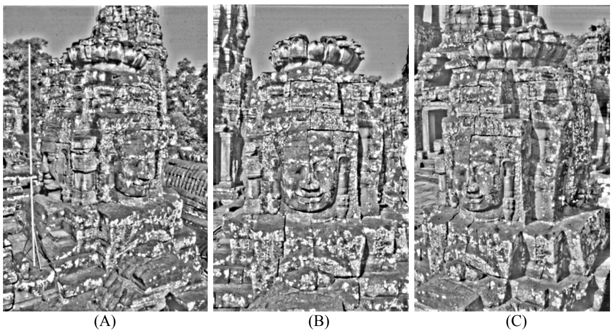


Figure 9: The three analyzed images of the smiling Buddha in Angkor Wat, Cambodia, after the Wallis filter enhancement.

	A	*B*	*C*
Extracted regions	18122	16778	17715
Matched A-B	197		
Matched B-C		902	
New matched A-B after guided matching	16		
New matched B-C after guided matching		7	
Points in 3 images	10		
Total number of 3D points	1047		

Table 2: Results of the tie point extraction between the three widely separated images of Figure 9.

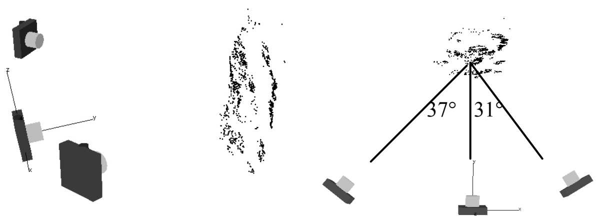


Figure 10: Recovered camera poses of the three widely separated views. The two relative rotation angles between the images are approximately 37 and 31 degrees.

5. CONCLUSIONS AND OUTLOOK

In this article we gave an overview on recent developments in automated orientation of close-range images and demonstrated their applicability on different data sets of real images.

The methods for automated image orientation have to deal with two issues: (i) the automatic measurement of markerless tie points and (ii) the provision of initial orientation parameters for the bundle block adjustment. The second issue is nowadays simply solved (once the correspondences are known) e.g. using orientation methods based on projective geometry (fundamental matrix and trifocal tensor) or with robust relative orientation procedures based on Monte Carlo type strategy [Cronk et al., 2006] – even for uncalibrated images. Although image orientation and calibration are often combined in the literature, we argue that both should be separated if possible. Image sequences acquired for object reconstruction usually do not allow for a proper calibration and consequently, the accuracies of the results will deteriorate. Therefore we prefer to work with calibrated images.

The automatic measurement of markerless tie points, however, is still a difficult and active research topic in close-range photogrammetry and computer vision. A clear fact is that no commercial solutions are still available. Depending on the baseline length between consecutive images, the approaches for automatic tie point extraction can be divided in point and region-based procedures. Whereas the point-based procedures apply simple and well-known extraction and matching techniques, the region-based procedures require more processing time and involved techniques, however, with the benefit that they are able to deal also with wide baseline images. Although the region-based techniques are the most general and are thus also applicable for short baselines, the original results do not exploit the full accuracy potential. However, by using LSM to refine the extracted features, it was shown that the accuracies can be significantly improved [Remondino, 2006].

We can safely conclude that the success of automatically orienting close-range images depends on the following main issues: (i) image arrangement (baselines and viewing directions) and (ii) the imaged scene properties (geometry and texture, even if it was shown that the image content can be enhanced for tie point extraction by image preprocessing).

Even if the image orientation step can be fully automated, within the 3D image-based modeling pipeline some user interaction is still required – especially in the subsequent modeling-phase – as the extracted features, even if well distributed for the orientation, are not sufficient for the object reconstruction, as not located in the salient object areas.

Although the presented examples show the applicability of automated image orientation to a variety of image configurations, in the future we plan to conduct more tests to further investigate the feasibility and limitations of present methods – especially with respect to wide baseline images.

REFERENCES

Baumberg, A., 2000: Reliable feature matching across widely separated views. Proc. of CVPR, pp. 774-781

Beardsley, P, Torr, P. and Zisserman, A., 1996: 3D model acquisition from extended image sequences. Proceedings of ECCV'96, Lecture Notes in Computer Sciences, Vol, 1065, pp. 683-695

Cronk, S., Fraser, C.S. and Hanley, H., 2006: Automatic calibration of colour digital cameras. The Photogrammetric Record. In press

Fischler, M. A. and Bolles, R. C., 1981: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol. 24, pp 381-395

Fitzgibbon, A and Zisserman, A., 1998: Automatic 3D model acquisition and generation of new images from video sequence. Proceedings of European Signal Processing Conference, pp. 1261-1269

Förstner, W. and Guelch, E., 1987: A fast operator for detection and precise location of distinct points, corners and center of circular features. ISPRS Conference on Fast Processing of Photogrammetric Data, Interlaken, Switzerland, pp. 281-305

Georgescu, B. and Meer, P., 2004: Point Matching under Large Image Deformations and Illumination Changes. PAMI, Vol. 26(6), pp. 674-688

Gruen, A., 1985: Adaptive least square correlation: a powerful image matching technique. South African Journal of PRS and Cartography, Vol. 14(3), pp. 175-187

Gruen, A., Zhang, L., Visnovcova, J., 2001: Automatic Reconstruction and Visualization of a Complex Buddha Tower of Bayon, Angkor, Cambodia. Proceedings of 5th International Conference on Optical 3-D Measurement Techniques, Vienna, Austria

Hartley, R. and Zisserman, A., 2001: Multiple View Geometry in Computer Vision, Reprinted Edition. Cambridge University Press, UK

Harris, C. and Stephens, M., 1988: A combined edge and corner detector. Proc. of Alvey Vision Conference, pp. 147-151

Heitger, F., Rosenthalter, L., von der Heydt, R., Peterhans, E. and Kuebler, O., 1992: Simulation of neural contour mechanism: from simple to end-stopped cells. Vision Research, Vol. 32(5), pp. 963-981

Läbe, T. and Förstner, W., 2006: Automated relative orientation of images. Proceedings of the 5th Turkish-German Joint Geodetic Days, 29-31 March, Berlin

Lindeberg, T., 1998: Feature detection with automatic scale selection. International Journal of Computer Vision, Vol. 30(2), pp. 79-116

Lowe, D., 2004: Distinctive image features from scale-invariant keypoints. IJCV, Vol. 60(2), pp. 91-110

Lucas, B.D. and Kanade, T., 1981: An iterative image registration technique with an applicatiuon to stereo vision. Proc. 7th Intern. Joint Conference on Artificial Intelligence

Matas, J., Chum, O., Urban, M. and Pajdla, T., 2002: Robust wide baseline stereo from maximally stable extremal regions. Proc. of British Machine Vision Conference, pp. 384-393

Mayer, H., 2005: Robust Least-Squares Adjustment Based Orientation and Auto-Calibration of Wide-Baseline Image Sequences. IAPRS, Vol. 36(3/W36), Beijing, China

Mikolajczyk, K. and Schmid, C., 2003: A performance evaluation of local descriptors. Proc. of CVPR

Mikolajczyk, K. and Schmid, C., 2004: Scale and Affine Invariant Interest Point Detectors. Int. Journal Computer Vision, Vol. 60(1), pp. 63-86

Mikolajczyk, K. Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Van Gool, L. 2005: A comparison of affine region detectors. Int. Journal of Computer Vision. In press

Nister, D., 2001: Automatic dense reconstruction from uncalibrated video sequences. PhD Thesis, Computational Vision and Active Perception Lab, NADA-KHT, Stockholm, 226 pages

Pollefeys, M., Koch, R. and Van Gool, L., 1999: Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. IJCV, 32(1), pp. 7-25

Pritchett, P. and Zisserman, A. 1998: Matching and reconstruction from widely separated views. 3D Structure from Multiple Images of Large-Scale Environments, LNCS 1506

Remondino, 2006: Detectors and descriptors for photogrammetric applications. IAPRS, Commission III Symposium, Bonn, Germany. In press

Roncella, R., Remondino, F. and Forlani, G., 2005: Photogrammetric bridging of GPS outages in mobile mapping. Videometrics VIII, Beraldin/El-Hakim/Grün/Walton (Eds), SPIE Electronic Imaging, Vol.5665, pp. 308-319

Roth, G. and Whitehead, A., 2000: Using projective vision to find camera positions in an image sequence. Proceedings of 13th Vision Interface Conference

Roth, G., 2004: Automatic correspondences for photogrammetric model building. IAPRS, Vol. 35(B5), pp. 713-718

Schaffalitzky, F. and Zisserman, A., 2002: Multi-view matching for unordered image sets. Proc. of ECCV

Shashua, A., 1997: Trilinear Tensor: The Fundamental Construct of Multiple-view Geometry and its Applications. Int. Workshop on Algebraic Frames for the Perception Action Cycle (AFPAC), Kiel, Germany

Shi, J. and Tomasi, C., 1994: Good features to track. IEEE Proceedings of CVPR, pp. 593-600

Smith, S.M. and Brady, J.M., 1997: SUSAN – a new approach to low level image processing. IJCV, Vol. 23(1), pp. 45-78

Tomasi, C. and Kanade, T., 1991: Shape and motion from image streams: a factorizaton method - part 3 on ‘Detection and Tracking of Point Features’. Technical Report CMU-CS-91-132, Carnegie Mellon University, Pittsburgh, PA, USA

Tuytelaars, T. and Van Gool, L., 2004: Matching widely separated views based on affine invariant regions. IJCV, Vol. 59(1), pp. 61-85

Van Gool, L. and Zisserman, A., 1996: Automatic 3D model building from video sequences. Proceedings of European Conference on Multimedia Applications, Services and Techniques, pp. 563-582

Xiao, J. and Shah, M. 2003. Two-frame wide baseline matching. IEEE Proceeding of 9th ICCV, Vol. 1, pp. 603-610

Wallis, R., 1976: An approach to the space variant restoration and enhancement of images. Proc. of Symposium on Current Mathematical Problems in Image Science, Naval Postgraduate School, Monterey, CA

3D ASPECTS OF 2D EPIPOLAR GEOMETRY

Ilias Kalisperakis, George Karras, Lazaros Grammatikopoulos

Laboratory of Photogrammetry, Department of Surveying,
National Technical University of Athens (NTUA), GR-15780 Athens, Greece
E-mail: ilias_k@central.ntua.gr, gkarras@central.ntua.gr, lazaros@central.ntua.gr

KEY WORDS: calibration, relative orientation, epipolar geometry

ABSTRACT

Relative orientation in a stereo pair (establishing 3D epipolar geometry) is generally described as a rigid body transformation, with one arbitrary translation component, between two formed bundles of rays. In the uncalibrated case, however, only the 2D projective pencils of epipolar lines can be established from simple image point homologies. These may be related to each other in infinite variations of perspective positions in space, each defining different camera geometries and relative orientation of image bundles. It is of interest in photogrammetry to also approach the 3D image configurations embedded in 2D epipolar geometry in a Euclidean (rather than a projective-algebraic) framework. This contribution attempts such an approach initially in 2D to propose a parameterization of epipolar geometry; when fixing some of the parameters, the remaining ones correspond to a 'circular locus' for the second epipole. Every point on this circle is related to a specific direction on the plane representing the intersection line of image planes. Each of these points defines, in turn, a circle as locus of the epipole in space (to accommodate all possible angles of intersection of the image planes). It is further seen that knowledge of the lines joining the epipoles with the respective principal points suffices for establishing the relative position of image planes and the direction of the base line in model space; knowledge of the actual position of the principal points allows full relative orientation and camera calibration of central perspective cameras. Issues of critical configuration are also addressed. Possible future tasks include study of different a priori knowledge as well as the case of the image triplet.

1. INTRODUCTION

In photogrammetric textbooks a typical definition of the task of relative orientation (RO) is that of establishing the relative position of two – already formed – homologue bundles of rays (involving 5 independent parameters). The object may then be reconstructed by bundle intersection in an arbitrarily oriented and scaled model space. In this sense, certain explicit or implicit assumptions are made:

- In order to establish RO, the camera interior orientation (IO) must be fully known beforehand. Knowledge of IO is also a prerequisite for linear algorithms for estimating RO – in fact equivalent to the computation of the 'essential matrix', as it came to be known in computer vision literature – which have been presented in photogrammetry (Thompson, 1959; Stefanovic, 1973; Khlebnikova, 1983).
- Determination of epipolar lines presupposes knowledge of both IO and RO. For instance: "If relative orientation is known for a given stereo pair, the coplanarity condition can be used to define epipolar lines" (Wolf & DeWitt, 2000). Or: "The epipolar lines can be determined after the photographs have been relatively oriented" (Mikhail et al., 2001).

Thus, most photogrammetric textbooks (rather understandably, in the context of routine mapping tasks using metric cameras) restrict the definition of RO to that for calibrated images. Currently, however, a more general view on RO is also adopted. For instance, according to the new edition of the *Manual of Photogrammetry* (McGlone, 2004) "the relative orientation of two uncalibrated straight-line-preserving cameras is characterized by 7 independent parameters. An object can be reconstructed only up to a spatial homography". It is also noted there that, 150 years ago, M. Chasles had detected the 1D homography between corresponding pencils of epipolar lines, whose 3 parameters combine with the 4 parameters defining the epipoles to yield the total of 7 independent parameters required for establishing RO in the uncalibrated case. It is further stated that 7 pairs of homologue points allow finding RO in uncalibrated stereopairs. It needs to be noted that 'relative orientation' stands here for something more general than the conventional photogrammetric concept (since no unique spatial relationship between the two images is fixed); it actually means 'recovery of 2D epipolar geometry'.

Clearly, it is thanks to extensive research in the field of computer vision that this point of view is being (re)introduced into the photogrammetric literature. In particular, Faugeras (1992) and Hartley (1992) have demonstrated that the 2D epipolar geometry of an image pair may still be established even with unknown IO. The 'fundamental matrix' **F** – having 7 independent parameters, found from simple point homologies – establishes the epipolar constraint on the uncalibrated pair and allows projectively distorted, i.e. non Euclidean, 3D reconstructions (Hartley & Zisserman, 2000, Faugeras & Luong, 2001).

Undoubtedly, the notion of the fundamental matrix has allowed a deeper insight into the structure of the stereopair. In fact – although somehow obscured in the many decades of technological advance and massive photogrammetric production – this generalization of the term 'relative orientation' to include the uncalibrated case (and thus signify the establishment of 2D epipolar geometry) is not unfamiliar to photogrammetry. Thus, in the framework of projective geometry Bender (1971) had formulated the equivalent of the fundamental matrix, which represents "the most general relative orientation of two photos". He also explained that use of one arbitrary camera matrix leads to construction of a model space related to the real object via a 3D projectivity. He concluded that its 15 parameters, added to the 7 of relative orientation, yield the 22 parameters (two DLT matrices), which fully relate two individual images to object space. Yet, it was Sebastian Finsterwalder who – in one of his remarkable publications – had already shown that, given the two epipoles, one can reconstruct an 'auxiliary' 3D object which is collinear to that depicted on the images, since all straight lines of the real object also correspond to straight lines on the two perspective projections from which this 'auxiliary' object is reconstructed. He pointed out that, assuming central perspective cameras, ∞^5 such projective reconstructions are possible from an uncalibrated image pair (Finsterwalder, 1899).

Notwithstanding the elegance and ‘compactness’ of essentially algebraic approaches, it is believed that the more geometric reasoning of Finsterwalder (also adopted in Rinner & Burkhardt, 1972) is indeed also useful to photogrammetry. It might help further illuminate the actual 3D geometry of the stereo pair, by indicating the countless combinations of relative – in its conventional meaning – and interior orientations embedded in one and the same 2D epipolar geometry. Besides, it could also further clarify why partial knowledge of interior orientation can allow recovering camera geometry from simple image point correspondences. It is from this point of view that the authors wish to address here certain aspects of two-image geometry.

2. THE GENERAL CASE

The base line o, defined by the projection centres O_1 and O_2 of a stereopair, intersects the two image planes ε_1, ε_2 in the respective epipoles e_1 and e_2 (Fig. 1). Epipolar planes, defined by the base and each imaged object point (such as P), intersect the image planes in homologue epipolar lines passing through the epipoles. Corresponding epipolar lines intersect on the intersection line (g) of the two image planes; the pencils of epipolar rays in both images are, therefore, projective (Hallert, 1960).

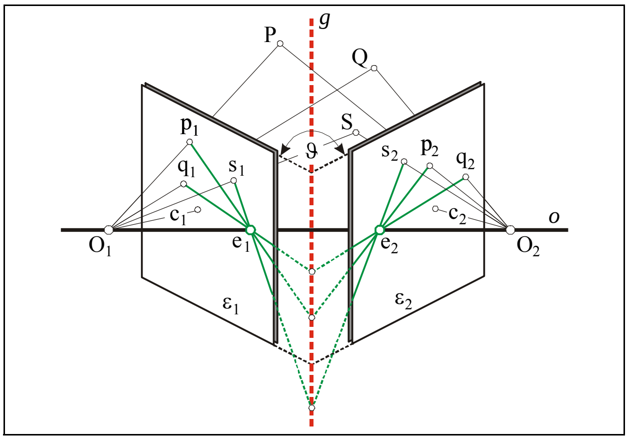


Figure 1. Epipolar geometry.

When only a sufficient number of image point correspondences are at hand, determination of the fundamental matrix allows establishing the epipoles (and consequently the projective pencils of epipolar lines). Once these two pencils are brought to some position in which they are perspective to each other, it is seen in Fig. 1 that changes of angle ϑ between the two image planes do not affect the perspective position of the pencils (they still intersect on g) and the coincidence of the epipolar planes. Thus, one may for the moment address the problem in 2D (i.e. set $\vartheta = \pi$).

First, it is assumed that the two image planes are not parallel (g is not at infinity) and that no image is parallel to the base line o (no epipole is at infinity). Thus, referring to Fig. 2, the pencil of e_1 may be intersected with any line g on image plane ε_1 not passing through e_1. No generality is lost if only lines through some fixed point K are considered, since the position of K only affects scale. For convenience, K is fixed on some epipolar ray through e_1 and all lines g are characterised by the angle δ they form with this ray. A line $g(\delta)$ intersects two other epipolar rays of e_1 in points A and B. Epipole e_2 can be constructed as the intersection of the two circular arcs c_a and c_b which see segments KA and KB under the respective angles α_2 and $\alpha_2+\beta_2$ of the epipolar pencil of the second image. All other couples of homologue epipolar rays also intersect on g (cross ratio constraint).

It is to note that a valid second location also exists for e_2, on the other side of line g, as the intersection of the circular arcs c_a and c_b if these are mirrored with respect to g. However, this could be disregarded since it corresponds to $\vartheta = 0$. A further remark is that, for some direction δ of line g, any translation of K (point of rotation of g) simply slides the position of e_2 along e_1e_2 (i.e. direction δ defines a line through e_1 as the locus of e_2).

Following Fig. 2, it can be also shown that the geometric locus for epipole e_2 is a circle c_k (which also contains e_1). Consider, for instance, the intersections M and N of epipolar lines e_1A and e_1B with c_a and c_b, respectively. In c_a point A views chord KM under the angle $\alpha_1+\delta$, while e_2 sees it under the supplementary angle. In c_b point B views chord KN under the angle $\alpha_1+\beta_1+\delta$, while e_2 sees it under the supplementary angle. Thus, the angular difference $Ke_2M - Ke_2N = KBN - KAM$, namely segment MN is seen from e_2 under the fixed angle β_1 (or, if e_2 lies between M and N, under angle $\pi-\beta_1$).

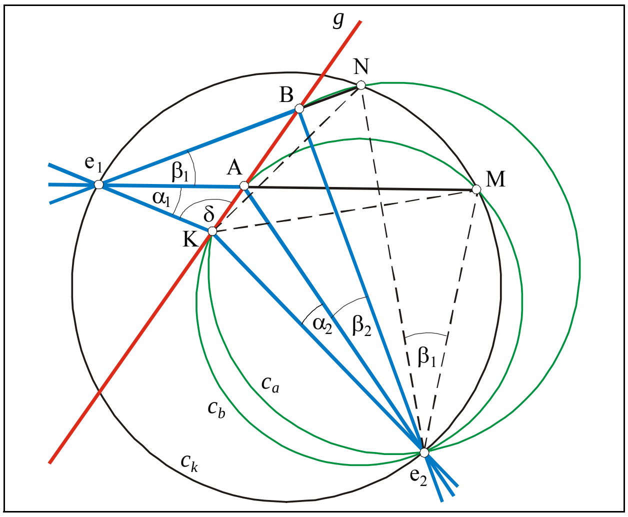


Figure 2. Epipolar geometry on the plane.

Thus, quite independently from the direction δ of line g, every point K fixes a circle c_k as the geometric locus of the epipole e_2. This circle can be constructed from e_1 and the two points M, N which are fixed by angles α_2 and β_2 of the pencil through e_2. Indeed, as seen from Fig. 2, M views segment KA under angle α_2 regardless of δ, i.e. of the actual position of A on epipolar line e_1A; the same is true for point N, which sees segment KB under angle $\alpha_2+\beta_2$ regardless of the position of B on epipolar line e_1B. Fixing K also fixes these two points (and all other similarly defined points corresponding to other epipolar lines of the pencil through e_1) and, as a consequence, c_k can be constructed.

Thus, the 7 degrees of freedom in 2D epipolar geometry (fundamental matrix) may be parameterized in the following way. The epipoles on ε_1 and ε_2 are fixed with 4 parameters. The remaining 3 degrees of freedom on the plane, which bring the two pencils of epipolar rays in perspective position (they intersect on a line g of direction δ rather than on a conic section), may be geometrically described as follows. For a fixed point K – its location only affects scale and is irrelevant to relative orientation – 2 parameters define with epipole e_1 the circle c_k, whose points are valid locations of epipole e_2. The third parameter is the rotation which, for any valid e_2, brings the corresponding epipolar ray to pass through K. This constrains the intersection of the two projective pencils on a line $g(\delta)$. In this sense, and disregarding scale, all possible positions of epipole e_2 can be grasped as a circular movement of all points of c_k around their corresponding line $g(\delta)$ and normal to it. This includes all possible angles of intersection of the image planes ($0 < \vartheta < 2\pi$, $\vartheta \neq \pi$) for each individual direction δ (cf. Fig. 3).

Thus, two further parameters δ, ϑ (9 in total) are required to determine in 3D the relative orientation of the two pencils of epipolar lines. These represent the direction of the second image plane relative to the first, i.e. a given 2D epipolar geometry includes, in principle, all possible directions. As discussed in the following section, knowledge of the line through the epipoles and the corresponding principal points allows establishing full relative orientation of the image planes. Then the base line *o* is also fixed in model space. Since a full relative orientation of an uncalibrated pair from central perspective cameras involves 11 independent parameters, the additional knowledge of one image coordinate (c, x_o or y_o) of the projective centre of each image provides full relative orientation of the pair.

• Note: Two special cases are pointed out. First, the two pencils of rays are identical, while the epipoles are not at infinity. This occurs if image planes are parallel but not coplanar (and *o* is not parallel to them); if image planes are coplanar but the camera constants differ; if epipoles are equidistant from the intersection of image planes. In such a case, circle c_k degenerates to a point coinciding with e_1. For every line *g*, a circle through e_1 about *g* and on a plane normal to it is the locus of e_2. Here, nonetheless, $\vartheta = 0$ is a valid angle referring to parallel image planes (*g* at infinity). Second, the epipolar lines of one image run parallel to each other, which occurs if this image is parallel to the base line *o*. It can be shown that, if epipole e_1 is at infinity, circle c_k degenerates to line MN (which can also be constructed).

3. PARTLY CALIBRATED IMAGES

Contrary to a typical photogrammetric approach, even in order to perform RO in its conventional sense (namely, allowing metric reconstruction) one does not need to assume already formed bundles; indeed, recovery of RO is possible together with partial camera calibration. Chang (1986) had given an early illustration of the possibility to find the IO parameters with a simultaneous adjustment of independent stereo pairs from the same camera. Faugeras et al. (1992) showed that assumption of common IO in an image pair produces two independent conditions among the elements of **F** and the IO parameters. Hence, if certain camera elements are considered as known, partial self-calibration is feasible from a single stereo pair. By fixing the principal point, for instance, one may recover the constant of a central perspective camera even if it varies between the two views (Hartley, 1992). Non-iterative algorithms have been reported for estimating one or two camera constants from the fundamental matrix, while critical configurations have also been demonstrated (Newsam et al., 1996; Sturm, 2001; Sturm et al., 2005).

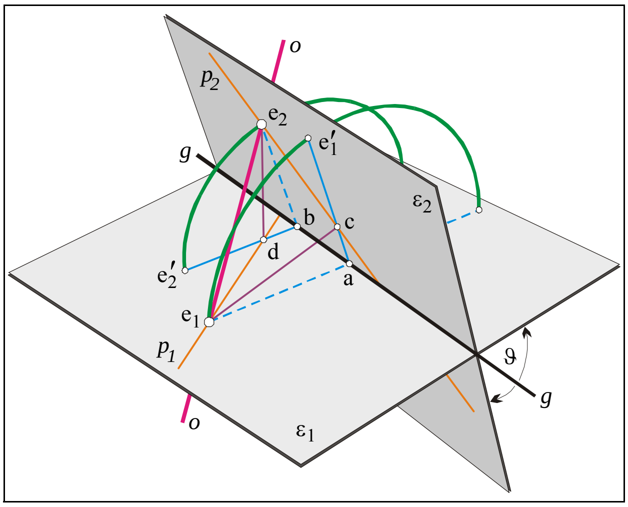


Figure 3. A possible relative position of images in 3D.

In order to address this issue here, it is referred to Fig. 3, which presents one out of the possible relative orientations of two images ε_1, ε_2 given their line of intersection *g*. The rotation of an image plane about *g* by a change in angle ϑ does not affect the epipolar lines or the coplanarity constraint. As mentioned, such rotations move epipoles e_1, e_2 on two parallel circles, which are normal to both image planes and their intersection *g*. In Fig. 3 line segments e_1a, $e_2'b$ represent the projections of the two circles on image plane ε_1, and e_2b, $e_1'a$ are their projections on ε_2.

Any two points on line *o* (e_1e_2) which joins the two epipoles can be chosen as projection centres. Camera constant and principal point corresponding to each projection centre can be determined through its normal to the respective image plane. As a consequence, for a particular angle ϑ the locus of the principal point on each image is a line through its epipole. One of them (e_1d) is constructed through the projection d of e_2 on plane ε_1; in a similar way, the line e_2c of the principal point may be found on ε_2. The two right triangles e_1ac and e_2bd are then similar since their angles e_1ac and e_2bd are equal to angle ϑ of the image planes. Hence, $ac/e_1a = bd/e_2b = \cos\vartheta$. Since $e_1a = e_1'a$ and $e_2b = e_2'b$ (radii of circles) it also holds that $ac/e_1'a = bd/e_2'b = \cos\vartheta$. Consequently, the change of angle ϑ affects on both images the direction of the line through the epipole and the principal point.

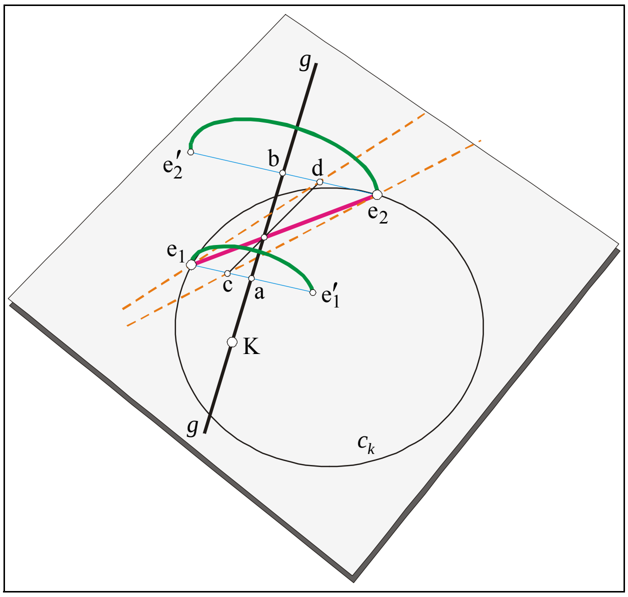


Figure 4. Inclination of plane ε_2 on ε_1 (cf. Fig. 3).

Referring now to Fig. 4, which shows the inclination on image plane ε_1, the line e_1d of the principal point of the first image has to intersect segment e_2e_2'. Thus, for a given δ only a part of the image plane represents a valid location for the principal point. The same holds for the second image. But if the principal point of the first image, or simply the direction of line e_1d, is known, then line e_2c of the principal point of the other image is further constrained to intersect segment e_1e_1' at a specific point c such that $ac/e_1a = bd/e_2b$. From the isosceles trapezium $e_1e_1'e_2e_2'$ it can be shown that equality of these two ratios exists only when the intersection of lines cd and e_1e_2 lies on line ***g*.** To summarize, if line e_1d is known, then for every angle δ the line of the principal point of the second image as well as the angle ϑ of the two images can both be found. It is further observed that the direction of e_1d also constrains angle δ to those values which give segments e_2e_2' that can be intersected by e_1d, which means that only part of circle c_k represents now valid positions for e_2.

If in addition to e_1d also line e_2c of the other principal point is known, the constraint that the two lines must intersect segments

e_1e_1' and e_2e_2' to equal ratios allows finding the compatible angle δ, since random selections of δ will not produce ratios ac/e_1a and bd/e_2b which are equal[1]. The fact that both ratios also equal cosϑ finally provides the 2 missing parameters, thus allowing a full estimation of relative orientation of the two image planes and, consequently, fixing through the two epipoles the base line *o* in space. It is noted, however, that ϑ only establishes the angle between the image planes. Thus, cosϑ provides the four angles ϑ, π−ϑ, π+ϑ, 2π−ϑ, which in fact represent the four possible solutions for relative orientation[2].

From the above it is seen that if not only the directions of these two lines (loci of the principal points) but the principal points themselves are known, the camera constant of the two images may be found through the projections of the respective principal points onto the base line *o* in space. This is in agreement to the knowledge that, in general, fixed principal points allow computation of the camera constants from the fundamental matrix.

• Note: In case of coplanarity of the optical axes[3] their common plane will be perpendicular to both images and their line of intersection *g*. In such a case the two circles of Figs. 3 and 4 will be coplanar and the two principal point lines (e_1d, e_2c) will coincide with segments e_1e_1' and e_2e_2'. In this situation none of the two ratios mentioned above can be determined and a rotation ϑ will not affect the coplanarity of the two axes. If the principal points themselves are known, for every ϑ there emerge different camera constants. Thus, coplanarity of image axes renders partial camera calibration impossible (Newsam et al., 1996). Yet, if the images are assumed as having identical camera constants, it is indeed possible to find that angle ϑ which will result to equal camera constants (Sturm, 2001). This will not hold if the two principal points are equidistant from the intersection *g* of image planes since then all angles ϑ produce equal values for the two camera constants. For identical camera constants this, of course, is equivalent with the equidistance of projection centres from g, also including parallelism of the optical axes, which is the critical geometry pointed out by Sturm (2001).

4. CONCLUDING REMARKS

In photogrammetric literature two distinct definitions of relative orientation of a stereopair coexist. The one presents it strictly as a separate 5-parameter orientation step following camera calibration (six parameters for two central perspective cameras) and founded on the intersection in 3D space of corresponding rays, which thus permits the metric reconstruction of object shape. A much wider view (embodied in the fundamental matrix, as elaborated in the computer vision literature) bypasses the camera calibration step and conceives relative orientation also as the 7-parameter 2D task of establishing homologue pencils of epipolar lines, which then allows object reconstruction up to a 3D projective transformation. Along with the relative position of (not known) bundles of rays which created the original image pair, the second group of parameters apparently incorporates ∞^4 combinations of interior and relative image orientations. The authors feel that photogrammetric literature needs to further scrutinize this ground between 2D and 3D epipolar geometry in a Euclidean framework. This is the motivation behind the attempt made here to handle existing degrees of freedom in a more directly geometric manner and illustrate how these are constrained once partial knowledge of interior orientation is available. Besides further elaboration of the approach presented here, future tasks include similar studies of the case when only the camera constants are regarded as known and, also, of the possible image configurations if an identical interior orientation of the image pair is assumed. Further, it is intended to investigate in this framework the possibility of other factorizations of epipolar geometry with some practical use. Finally, study of overlapping image triplets in a similar manner is also within the authors' intentions.

REFERENCES

Bender, L.U., 1971. Analytical Photogrammetry: a Collinear Theory. Final Technical Report RADC-TR-71-147, Rome Air Development Center, New York.

Chang, B., 1986. The formulae of the relative orientation for non-metric camera. International Archives of Photogrammetry & Remote Sensing, 26(5):14-22.

Faugeras, O.D., 1992. What can be seen in three dimensions with an uncalibrated stereo-rig? European Conference on Computer Vision, Springer, pp. 563-578.

Faugeras, O.D., Luong Q.T., Maybank S. J., 1992. Camera self-calibration: theory and experiments. European Conference on Computer Vision, Springer, pp. 321–334.

Faugeras, O.D., Luong Q.-T., 2001. The Geometry of Multiple Images. The MIT Press, Cambridge, Mass.

Finsterwalder, S., 1899. Die geometrischen Grundlagen der Photogrammetrie. Jahresbericht der Deutschen Mathem.-Vereinigung (Deutsche Gesellschaft für Photogrammetrie, Sebastian Finsterwalder zum 75. Geburtstage, Verlag Herbert Wichmann, Berlin, 1937, pp. 17-45).

Hallert B., 1960. Photogrammetry: Basic Theory and General Survey. McGraw-Hill, New York.

Hartley, R., 1992. Estimation of relative camera positions for uncalibrated cameras. European Conference on Computer Vision, Springer, pp. 579-587.

Hartley, R., Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press.

Khlebnikova, T., 1983. Determining relative orientation angles of oblique aerial photographs. Mapping Science & Remote Sensing, 21(1):95-100.

McGlone, J.C. (ed.), 2004. Manual of Photogrammetry. American Society for Photogrammetry and Remote Sensing, 5th edition, Bethesda, Maryland.

Mikhail E.M., Bethel J.S., McGlone J.C., 2001. Introduction to Modern Photogrammetry. John Wiley & Sons, Inc., New York.

Newsam, G.N., Huynh, D.Q., Brooks, M.J., Pan H.P., 1996. Recovering unknown focal lengths in self-calibration: an essentially linear algorithm and degenerate configurations. International Archives of Photogrammetry & Remote Sensing, 31(3):575-580.

Rinner, K., Burkhardt, R., 1972. Photogrammetrie. Handbuch der Vermessungskunde, Band III a/3, J.B. Metzlersche Verlagsbuchhandlung, Stuttgart.

[1] To this point, however, it is not known to the authors whether more than one solution for δ is possible.

[2] At this stage the criterion that reconstructed points should be in front of both cameras (Stefanovic, 1973) cannot be applied, since the principle point could be on either side of the epipole in both images.

[3] If the principal points or the lines connecting them with the epipoles are known, a way to distinguish whether the optical axes are coplanar or not is to check whether these lines are also homologue epipolar lines.

Stefanovic, P., 1973. Relative orientation – a new approach. ITC Journal, no. 3, pp. 417-447.

Sturm, P., 2001. On focal length calibration from two views. IEEE Int. Conference on Computer Vision & Pattern Recognition, pp. 145–150.

Sturm, P., Cheng, Z.L., Chen, P.C.Y., Poo A.N., 2005. Focal length calibration from two views: method and analysis of singular cases. Computer Vision and Image Understanding, 99(1): 58-95.

Thompson, E. H., 1959. A rational algebraic formulation of the problem of relative orientation. The Photogrammetric Record, 3(14):152-159.

Wolf P.R., DeWitt B.A., 2000. Elements of Photogrammetry with Applications in GIS. McGraw-Hill, New York.

A PROBABILISTIC NOTION OF CAMERA GEOMETRY: CALIBRATED VS. UNCALIBRATED

Justin Domke and Yiannis Aloimonos

Computational Vision Laboratory, Center for Automation Research
University of Maryland
College Park, MD, 20740, USA
domke@cs.umd.edu, yiannis@cfar.umd.edu
http://www.cs.umd.edu/∼domke/, http://www.cfar.umd.edu/∼yiannis/

KEY WORDS: Correspondence, Matching, Fundamental Matrix, Essential Matrix, Egomotion, Multiple View Geometry, Probabilistic, Stochastic

ABSTRACT:

We suggest altering the fundamental strategy in Fundamental or Essential Matrix estimation. The traditional approach first estimates correspondences, and then estimates the camera geometry on the basis of those correspondences. Though the second half of this approach is very well developed, such algorithms often fail in practice at the correspondence step. Here, we suggest altering the strategy. First, estimate probability distributions of correspondence, and then estimate camera geometry directly from these distributions. This strategy has the effect of making the correspondence step far easier, and the camera geometry step somewhat harder. The success of our approach hinges on if this trade-off is wise. We will present an algorithm based on this strategy. Fairly extensive experiments suggest that this trade-off might be profitable.

1 INTRODUCTION

The problem of estimating camera geometry from images lies at the heart of both Photogrammetry and Computer Vision. In our view, the enduring difficulty of creating fully automatic methods for this problem is due to the necessity to integrate image processing with multiple view geometry. One is given images as input, but geometry is based on the language of points, lines, etc. Bridging this gap- using image processing techniques to create objects useful to multiple view geometry- remains difficult. In both the Photogrammetric and Computer Vision literature, the object at interface between image processing and geometry is generally correspondences, or matched points. This is natural in Photogrammetry, because correspondences are readily established by hand. However, algorithmically estimating correspondences directly from images remains a stubbornly difficult problem.

One may think of most of the previous work on Essential or Fundamental matrix estimation as falling into one of two categories. First, there is a rather mature literature on Multiple View Geometry. This is well summarized in Hartley and Zisserman's recent book (Hartley and Zisserman, 2004), emphasizing the uncalibrated techniques leading to Fundamental Matrix estimation. Specifically, there are techniques for estimating the Fundamental Matrix from the minimum ofseven correspondences (Bartoli and Sturm, 2004). In the calibrated case, the Essential Matrix can be efficiently estimated from five correspondences (Nistér, 2004). Given perfect matches, it is fair to say that the problem is nearly solved.

The second category of work concerns the estimation of the correspondences themselves. Here commonly a feature detector (e.g. the Harris corner detector (Harris and Stephens, 1988)) is first used to try to find points whose correspondence is most easily established. Next, matching techniques are used to find probable matches between the feature points in both images (e.g. normalized cross correlation, or SIFT features (Lowe, 2004)). These are active research areas, and progress continues up to the present. Nevertheless, no fully satisfactory algorithm exists. Current algorithms often suffer from problems such as change in scale or surface orientation (Schmid et al., 2000). Furthermore, there are many situations in which it is essentially *impossible* to estimate correspondences with out using a higher-level understanding of the scene. These include repeated structures in the image, the aperture effect, lack of texture, etc. When humans estimate correspondences, they use this high-level information. Nevertheless, it is unavailable to algorithms.

Research in multiple view geometry, of course, has considered the difficulties in the underlying algorithms for correspondence estimation. As such, robust techniques such as RANSAC (Fischler and Bolles, 1981) are traditionally used to estimate a camera geometry from a set of correspondences known to include many incorrect matches. These techniques are fairly successful, but because even 'inlying' correct matches include noise there is a difficulty in discriminating between inlying matches with noise, and outlying, 'wrong', matches. When simultaneously adjusting the camera geometry, and 3-D points in a final optimization, bundle adjustment methods frequently use more sophisticated noise models which smoothly account for error due to both noise, and 'outlying' matches (Triggs et al., 1999).

In this paper, we suggest that it is worth stepping back and reconsidering if correspondences are the correct structure to use at the interface between image processing and multiple view geometry. Point correspondences are natural in Photogrammetry because they are easily estimated by humans. Nevertheless they are very difficult to estimate algorithmically. Here, we suggest instead using *correspondence probability distributions*. We can see immediately that this makes the image processing side of the problem much easier. If repetitive structure or the aperture effect presents itself, it is simply incorporated into the probability distribution. We will present a simple, contrast invariant, technique for estimating these correspondences from the phase of tuned Gabor filters.

The more difficult side of this strategy concerns multiple view geometry. One must estimate the camera geometry from only distributions of correspondence. As we will see, one can quite easily define a *probability* for any given camera motion, from only these distributions of correspondence. We then present a heuristic non-linear optimization scheme to find the most probable geometry. In practice, this space has a similar structure to the least-squares epipolar error space, (Oliensis, 2005) in that it contains relatively few local minima.

1.1 Previous Work

Other work has asked similar questions. First, there are techniques which generate from images feature points, and local image profiles, with out estimating an explicit correspondences (Makadia et al., 2005). These techniques then find motions which are compatible with these features, in the sense that each feature tends to have a compatible feature along the epipolar line in the second image.

Other work has created weaker notions of correspondences, such as the normal flow. If a point is along a textureless edge in one image, local measurements can only constrain it to lie along the same edge in the second image. This constraint is essentially the normal flow, and algorithms exist to estimate 3-D motion directly from it (Brodsky et al., 2000). Though these techniques will not suffer from the aperture effect, they cannot cope with situations such as repeated structures in the images. It is also important to notice that the normal flow will give up information unnecessarily at points which do not happen to suffer from the aperture effect.

2 CORRESPONDENCE PROBABILITY DISTRIBUTIONS

Given a point q in the first image, we would like the probability that this correspondences most closely to each pixel s in the second image. It is important to note that there is no obvious way to use traditional matching techniques here. Whereas traditional techniques try to find the most probably point corresponding to q, we require the relative probabilities of *all* points.

Our approach is based on the phase of tuned Gabor filters. Let $\phi_{l,\gamma}(s)$ denote the phase of the filter with scale l and orientation γ at point s. Now, given a single filter, (l,γ), we take the probability that s corresponds to a given point $\hat{q}$ to be proportional to

$$\exp((\phi_{l,\gamma}(s)-\phi_{l,\gamma}(\hat{q}))^2)+1. \qquad (1)$$

Combining the probability distributions given by all filters then yields the probability that s corresponds to $\hat{q}$, which we denote by $\rho_s(\hat{q})$.

$$\rho_s(\hat{q}) \propto \prod_{l,\omega} \exp((\phi_{l,\gamma}(s)-\phi_{l,\gamma}(\hat{q}))^2)+1. \qquad (2)$$

Note here that $\hat{q}$ corresponds to a particular *pixel* in the second image. Since we are computing probabilities over a discrete grid, we approximate the probability that s corresponds to an arbitrary point, having non-integer coordinates, though the use of a Gaussian function.

$$\rho_s(q) \propto \max_{\hat{q}} \rho_s(\hat{q}) \exp(-|q-\hat{q}|^2)+\alpha \qquad (3)$$

Here, α represents the probability that the information given by the Gabor filters is misleading. This would be the case, for example, were the point s to become occluded in the second image. Notice that adding the constant of α is equivalent to combining the distribution with the 'flat' distribution in which all points q are equally likely. In all experiments described in this paper, we have used $\alpha = 1$.

Correspondence distributions for several images are shown in Figure 2.

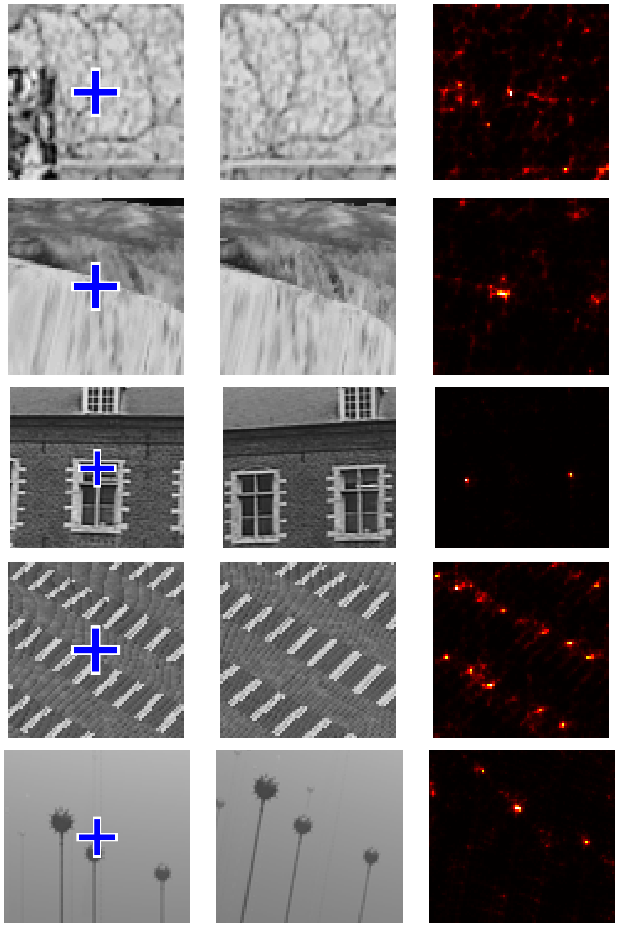


Figure 1: Correspondence Probability Distributions. Left: First image, with point in consideration marked. Center: Second image: Right: Probability distribution over the points in the second image, with probability encoded as color.

3 ESSENTIAL AND FUNDAMENTAL MATRIX ESTIMATION

Given the correspondence distributions, we will define natural distributions over the space of the Fundamental and Essential Matrices. Because the space of these matrices are of high dimension (7 and 5 respectively), it is impractical to attempt to calculate a full distribution, by sampling. It is possible that future work will directly use these distributions. Nevertheless, we use a simple heuristic optimization to maximize the probability in the Essential or Fundamental Matrix space. This makes it possible to examine the behavior of these distributions more easily.

3.1 Fundamental and Essential Matrix Probability

Given the correspondence distribution for a single point s, $\rho_s(\cdot)$, we define a distribution over the space of fundamental matrices.

$$\rho(F) \propto \max_{q:q^T Fs=0} \rho_s(q) \tag{4}$$

Thus, the probability of a given Fundamental Matrix F is proportional to *the maximum probability correspondence compatible with the epipolar constraint.* Now, to use all correspondence distributions, simply take the product of the distributions given by each point s.

$$\rho(F) \propto \prod_s \max_{q:q^T Fs=0} \rho_s(q) \tag{5}$$

Substituting our expression for $\rho_s(q)$ from Equation (3), we obtain

$$\rho(F) \propto \prod_s [\max_{q:q^T Fs=0} \max_{\hat{q}} \rho_s(\hat{q}) \exp(-|q-\hat{q}|^2) + \alpha]. \tag{6}$$

Rearranging terms, this is

$$\rho(F) \propto \prod_s [\max_{\hat{q}} \rho_s(\hat{q}) \max_{q:q^T Fs=0} \exp(-|q-\hat{q}|^2) + \alpha]. \tag{7}$$

Notice here, that we do not need to explicitly find the point q. Only required is $\max_{q:q^T Fs} |q - \hat{q}|$. Notice that this is exactly the minimum distance of the point $\hat{q}$ from the line Fs. Therefore, we can write the probability of F in it's final form.

$$\rho(F) \propto \prod_s [\max_{\hat{q}} \rho_s(\hat{q}) \exp(-(\hat{q}^T l_{(F,s)})^2) + \alpha] \tag{8}$$

Here, $l_{(F,s)}$ is the line Fs normalized such that $r^T l_{(F,s)}$ gives the minimum distance between r and the line Fs on the plane $z = 1$. If F_i is the ith row of F, then

$$l_{(F,s)} = \frac{Fs}{\sqrt{(F_1 s)^2 + (F_2 s)^2}}. \tag{9}$$

When searching for the most probable F, a parameterization of the fundamental matrices is required. We found it convenient to use three parameters f, p_x, and p_y representing the focal length, and x and y coordinates of the principal point. Next, keeping the magnitude of the translation vector t fixed to one, we took two parameters to parameterize its axis and angle. Finally, we used 3 parameters to represent the rotation vector ω. This corresponds to a rotation of an angle $|\omega|$ about the axis $\omega/|\omega|$.

$$K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \tag{10}$$

$$E = [t]_\times R(\omega) \tag{11}$$

$$F = K^{-T} E K^{-1} \tag{12}$$

Notice there are a total of 8 free parameters, despite the fact that the Fundamental Matrix has only 7 degrees of freedom. Though this presents no problem to the estimation of F, it does mean that an ambiguity is present in the underlying parameters.

To extend this to the calibrated case, we take K to be known. Thus, there are now 5 free parameters: 2 for the translation t, and 3 for the rotation ω. It would be trivial to extend this to the case that only certain calibration parameters were known, or to include a constant for camera skew.

3.2 Optimization

To explore the behavior of the probability distributions over the Fundamental and Essential Matrices, we will use a heuristic optimization to try to find $\arg\max_F \rho(F)$ and $\arg\max_E \rho(E)$, respectively. The optimization proceeds as follows: First, select N random points in the Fundamental or Essential matrix space. Evaluate $\rho(E)$ or $\rho(F)$ at each of these points. Next, take the M highest scoring points, and run a nonlinear optimization, initialized to each of these points. We have used both Simplex and Newton's type optimizations, with little change in performance. The final, highest scoring point is taken as the max.

For the calibrated case, we have found that using $N = 2500$ and $M = 25$ was sufficient to obtain a value very near the global maximum in almost all cases. As in the case for the standard least-squares error surface (Oliensis, 2005) (Tian et al., 1996), there are generally several, but only several local minima. Usually, a significant number of the nonlinear searches lead to the same (global) point.

In the uncalibrated case, we used $N = M = 100$. (Thus searches are taken from 100 random points.) We found that it was necessary to increase M to 100 to obtain reasonable certainty of obtaining the global maximum. At the same time, we found that increasing N did not improve results, and may even be counterproductive. Still, the space of $\rho(F)$ appears to have more local minima, and even this increased method does not always appear to achieve the global maximum.

4 EXPERIMENTS

To analyze the performance of the framework, we prepared three different 3-D Models with the POV-Ray software. Each model was chosen for its difficulty, including repetitive structure, lack of texture, or little image motion. The use of synthetic models makes the exact motion and calibration parameters available. For each model, we generated two different image sequences- one with a forward motion, and one with a motion parallel to the image plane.

For each image pair, 10,000 correspondence probability distributions were created. Next, the calibrated and uncalibrated algorithm were both run across a range of input sizes. For each input size, 100 random subsets of the correspondences were generated, and the algorithm was run on each input.

In the calibrated case, the measurement of error is simple. Let the true translation vector be t_0, normalized so that $|t_0| = 1$. Let the vector parameterizing the true rotation matrix be ω_0. The error metrics we use are simply the Euclidean distance between the estimated and true motion vectors, $|t - t_0|$, and $|\omega - \omega_0|$ respectively. For each input size, means are taken over the errors for all resulting motion estimates.

In the uncalibrated case, we must measure the error of a given fundamental matrix F. Commonly used metrics such as the Frobenius norm are difficult to interpret, and allow no comparison to the calibrated case. Instead, we use the known ground truth calibration matrix K to obtain E. (Hartley and Zisserman, 2004)

$$E = K^T F K \tag{13}$$

Next, Singular Value Decomposition is used to decompose E into the translation and rotational components, $E = [t]_\times R(\omega)$. From this, it is simple to recover the underlying motion parameters, t and ω. The error is then measured in the same way as the calibrated case.

Results for the 'Cloud', 'Abyss', and 'Biscuit' models are shown in figures 2, 3, and 4 respectively. Several observations are clear from the data. First, motion estimation is always more accurate when the epipole is in the middle of the image than when it is parallel to it. Surprisingly, perhaps, neither the calibrated nor uncalibrated approach clearly outperforms the other. The performance of the uncalibrated approach relative to the calibrated approach is better when the epipole is further from the image.

Two frames from the 'Castle' sequence, along with the epipolar lines are shown in Figure 5. Two frames from the popular 'Oxford Corridor' sequence are shown in Figure 6. In both cases, approximately 2000 correspondence distributions were used. Though no ground truth calibration or motion is available, the reader can observe the close correspondence among epipolar lines.

The running time of the algorithm is dominated by the time to generate correspondence distributions. In practice, the motion estimation step runs on the order of a minute on a modern laptop.

5 CONCLUSIONS AND FUTURE WORK

With real cameras, neither the fully calibrated, nor fully uncalibrated approach is fully realistic. In practice, one has some idea of the calibration parameters, even if only from knowledge of typical cameras. At the same time, even when a camera is calibrated, the true calibration is not found *exactly*. It would be quite natural to extend this paper's work to create a unifying approach between the two cases.

Write the prior distribution over the focal lengths by $\rho(f)$. Similarly, we can write the prior distributions of the principal point by $\rho(p_x, p_y)$. Now, we can make the Bayesian nature of this approach more explicit by writing 14 as

$$\rho(E|f, p_x, p_y) \propto \max_{q:q^T K^{-T} E K^{-1} s = 0} \rho_s(q) \tag{14}$$

Now, in the optimization step, instead of seeking

$$\arg\max_F \rho(F), \tag{15}$$

the optimization would be over

$$\arg\max_{E,f,p_x,p_y} \rho(E|f, p_x, p_y)\rho(f)\rho(p_x, p_y). \tag{16}$$

In this way, in one step, the most likely calibration parameters would be found as well as the most likely motion. This could be particularly useful in the common case that the camera calibration is approximately known, but the focal length changes, perhaps due to change of focus.

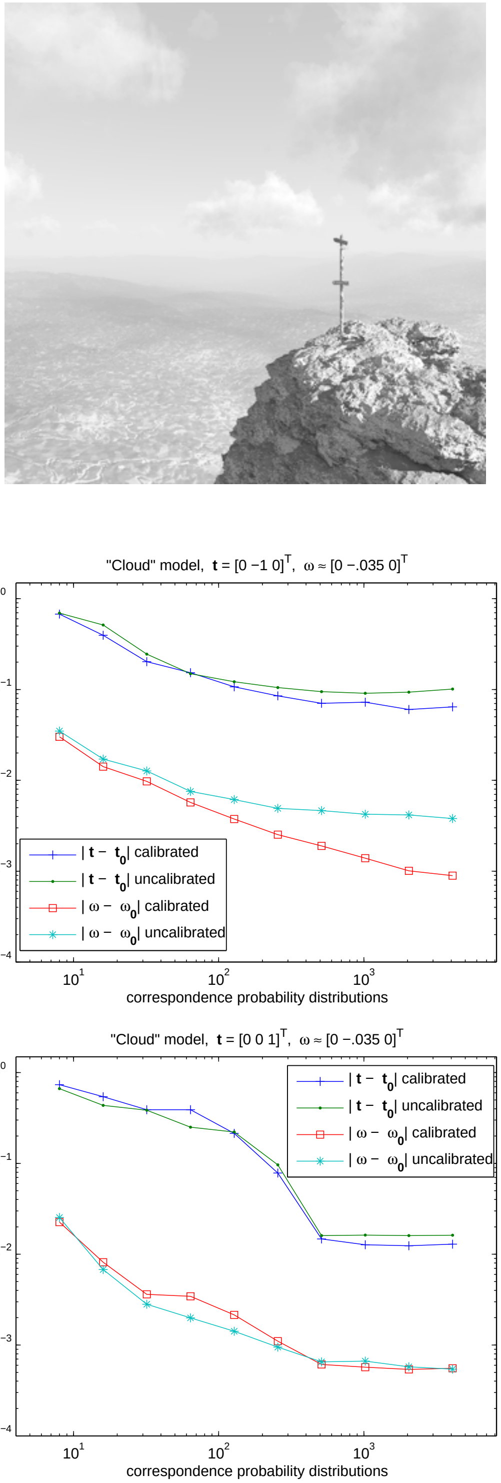


Figure 2: 'Cloud' model, and mean errors for two different motions.

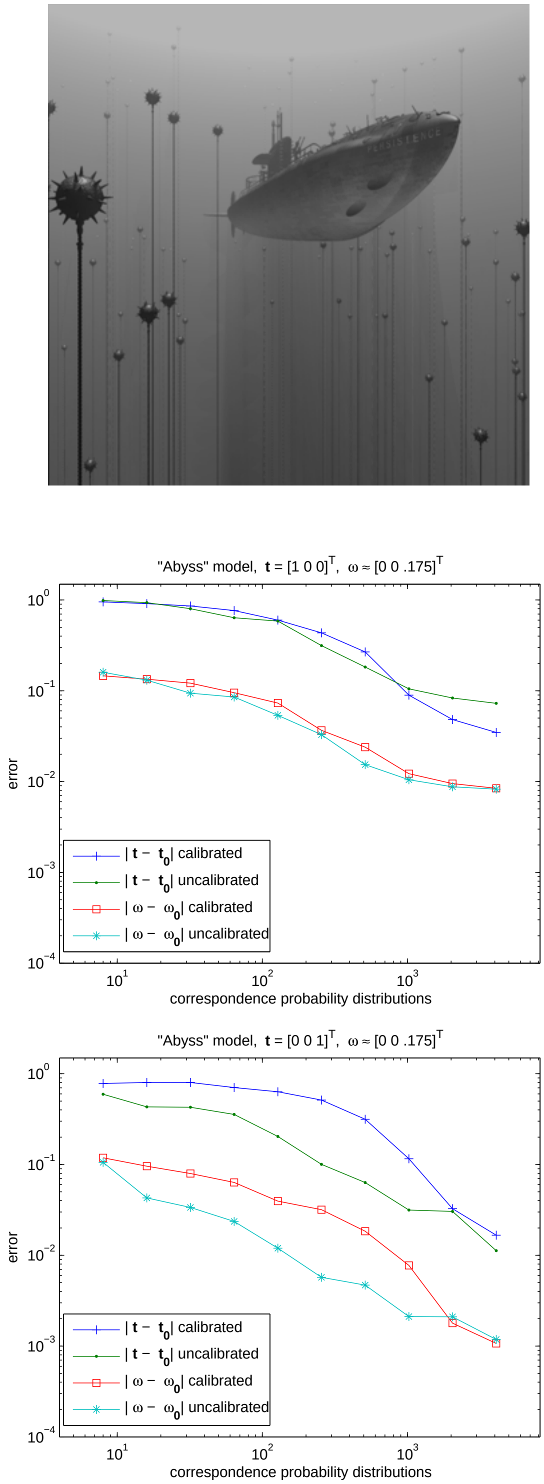


Figure 3: ’Abyss’ model, and mean errors for two different motions.

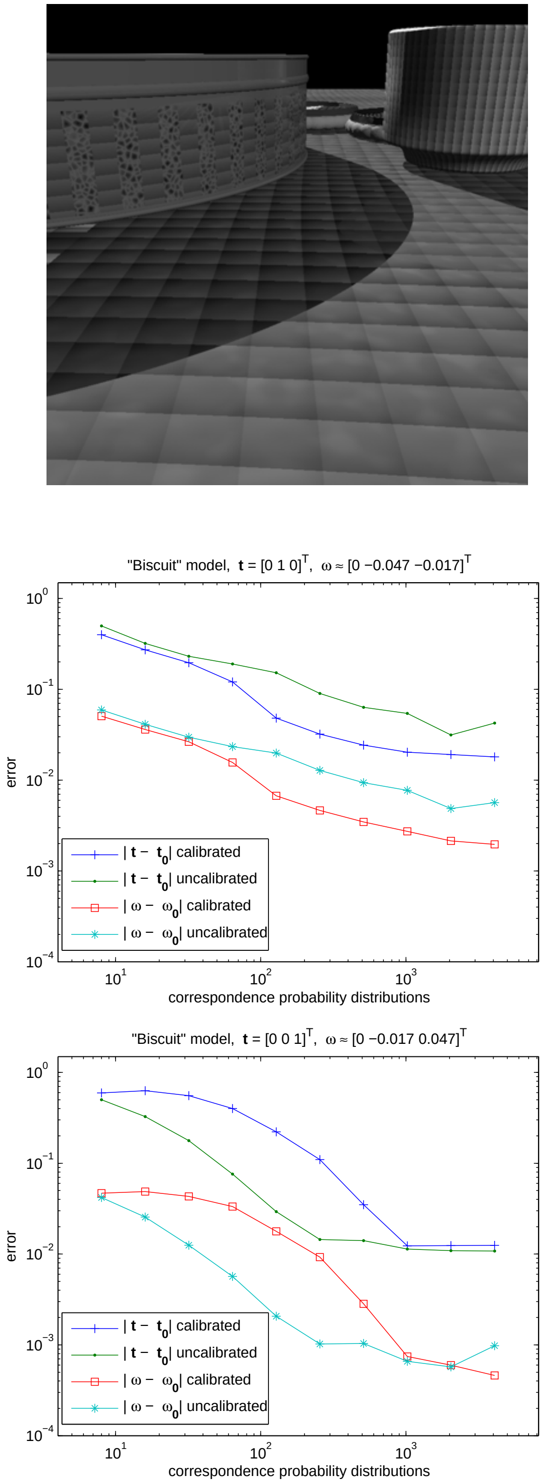


Figure 4: ’Biscuit’ model, and mean errors for two different motions.

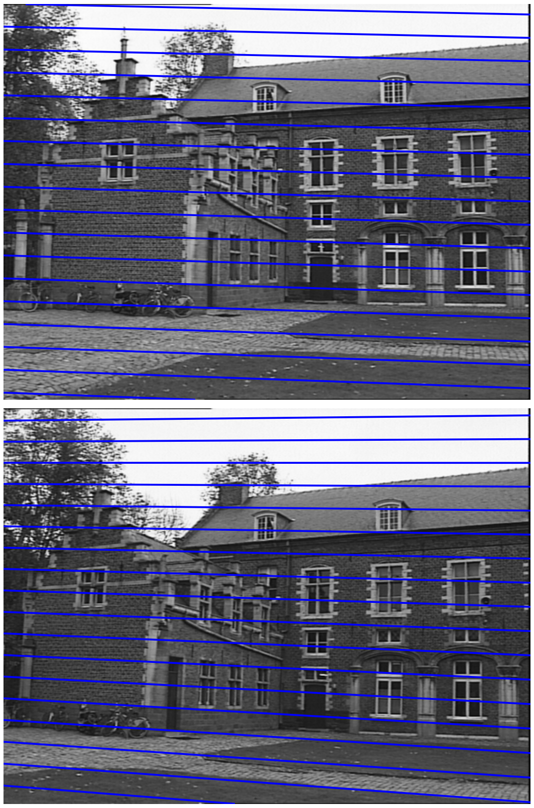


Figure 5: Two frames from the 'Castle' sequence, with epipolar lines overlaid

ACKNOWLEDGEMENTS

The authors would like to thank Gilles Tran for providing the 3-D models, The Oxford Visual Geometry Group for providing the 'Oxford Corridor' sequence, and Marc Pollefeys for providing the 'Castle' sequence.

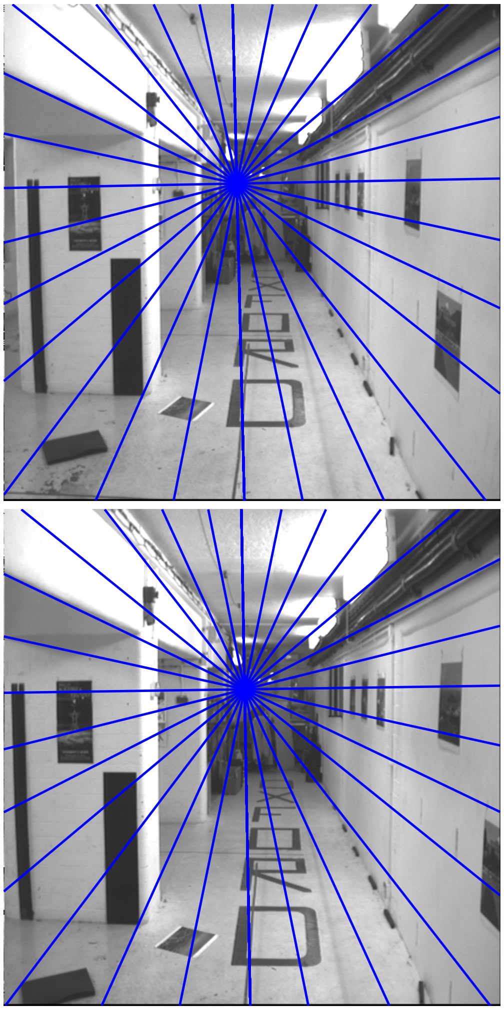


Figure 6: Two frames from the 'Oxford Corridor' sequence, with epipolar lines overlaid

REFERENCES

Bartoli, A. and Sturm, P., 2004. Non-linear estimation of the fundamental matrix with minimal parameters. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(4), pp. 426–432.

Brodsky, T., Fermuller, C. and Aloimonos, Y., 2000. Structure from motion: Beyond the epipolar constraint. International Journal of Computer Vision 37(3), pp. 231–258.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24(6), pp. 381–395.

Harris, C. G. and Stephens, M., 1988. A combined corner and edge detector. In: AVC88, pp. 147–151.

Hartley, R. I. and Zisserman, A., 2004. Multiple View Geometry in Computer Vision. Second edn, Cambridge University Press, ISBN: 0521540518.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60(2), pp. 91–110.

Makadia, A., Geyer, C. and Daniilidis, K., 2005. Radon-based structure from motion without correspondences. In: CVPR.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. IEEE Trans. Pattern Anal. Mach. Intell. 26(6), pp. 756–777.

Oliensis, J., 2005. The least-squares error for structure from infinitesimal motion. Int. J. Comput. Vision 61(3), pp. 259–299.

Pollefeys, M., Gool, L. V., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J. and Koch, R., 2004. Visual modeling with a hand-held camera. Int. J. Comput. Vision 59(3), pp. 207–232.

Schmid, C., Mohr, R. and Bauckhage, C., 2000. Evaluation of interest point detectors. International Journal of Computer Vision 37(2), pp. 151–172.

Tian, T., Tomasi, C. and Heeger, D., 1996. Comparison of approaches to egomotion computation.

Triggs, B., McLauchlan, P. F., Hartley, R. I. and Fitzgibbon, A. W., 1999. Bundle adjustment - a modern synthesis. In: Workshop on Vision Algorithms, pp. 298–372.

BUNDLE ADJUSTMENT RULES

Chris Engels, Henrik Stewénius, David Nistér

Center for Visualization and Virtual Environments, Department of Computer Science,
University of Kentucky
{engels@vis, stewe@vis, dnister@cs}.uky.edu
http://www.vis.uky.edu

KEY WORDS: Bundle Adjustment, Structure from Motion, Camera Tracking

ABSTRACT:

In this paper we investigate the status of bundle adjustment as a component of a real-time camera tracking system and show that with current computing hardware a significant amount of bundle adjustment can be performed every time a new frame is added, even under stringent real-time constraints. We also show, by quantifying the failure rate over long video sequences, that the bundle adjustment is able to significantly decrease the rate of gross failures in the camera tracking. Thus, bundle adjustment does not only bring accuracy improvements. The accuracy improvements also suppress error buildup in a way that is crucial for the performance of the camera tracker. Our experimental study is performed in the setting of tracking the trajectory a calibrated camera moving in 3D for various types of motion, showing that bundle adjustment should be considered an important component for a state-of-the-art real-time camera tracking system.

1 INTRODUCTION

Bundle adjustment is the method of choice for many photogrammetry applications. It has also come to take a prominent role in computer vision applications geared towards 3D reconstruction and structure from motion. In this paper we present an experimental study of bundle adjustment for the purpose of tracking the trajectory of a calibrated camera moving in 3D. The main purposes of this paper are

- To investigate experimentally the fact that bundle adjustment does not only increase the accuracy of the camera trajectory, but also prevents error-buildup in a way that decreases the frequency of total failure of the camera tracking.
- To show that with the current computing power in standard computing platforms, efficient implementations of bundle adjustment now provide a very viable option even for real-time applications, meaning that bundle adjustment should be considered the gold standard for even the most demanding real-time computer vision applications.

The first item, to show that bundle adjustment can in fact make the difference between total failure and success of a camera tracker, is interesting because the merits of bundle adjustment are more often considered based on the accuracy improvements it provides to an estimate that is already approximately correct. This is naturally the case since bundle adjustment requires an approximate (as good as possible) initialization, and will typically not save a really poor initialization. However, several researchers have noted (Fitzgibbon and Zisserman, 1998, Nistér, 2001, Pollefeys, 1999) that in the application of camera tracking, performing bundle adjustment each time a new frame has been added to the estimation can prevent the tracking process from failing over time. Thus, bundle adjustment can over time in a sequential estimation process have a much more dramatic impact than mere accuracy improvement, since it improves the initialization for future estimates, which can ultimately enable success in cases that would otherwise miserably fail. To our knowledge, previous authors have mainly provided anecdotal evidence of this fact, and one of

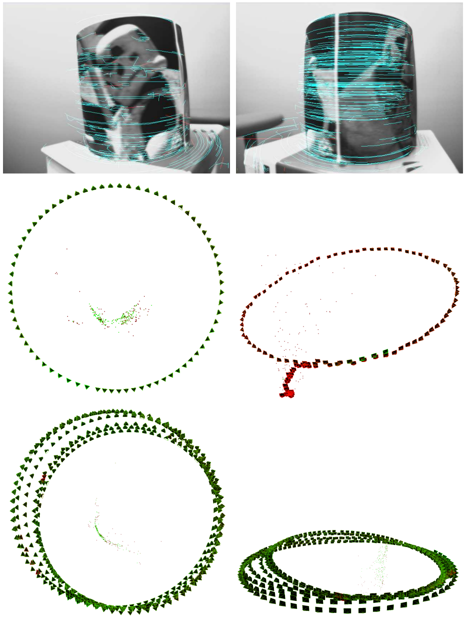


Figure 1: Top: Feature tracking on a 'turntable' sequence created by rotating a cylinder sitting on a rotating chair. Middle Left: When bundle adjusting the 20 most recent views with 20 iterations every time a view is added, the whole estimation still runs at several frames a second, and produces a nice circular camera trajectory, Middle Right: Without bundle adjustment, the estimation is more irregular, but perhaps more importantly, somewhat prone to gross failure. Here we show an example of the type of failure prevented by bundle adjustment. Bottom Left and Right: Although there is some drift, the bundle adjusted estimation is much more reliable and relatively long term stable. Here two views of multiple laps are shown. All the laps were estimated as full 6 degree of freedom unsmoothed motion and without attempting to establish correspondences to previous laps.

This work was supported in part by the National Science Foundation under award number IIS-0545920, Faculty Early Career Development (CAREER) Program.

our main contributions is to quantify the impact of bundle adjustment on the failure rate of camera tracking. In particular, we investigate the impact on the failure rate of n iterations of bundle adjustment over the last m video frames each time a frame is added, for various values of n and m.

The second item, to show that bundle adjustment can now be considered in real-time applications, is partially motivated by the fact that bundle adjustment is often dismissed as a batch-only method, often when introducing another ad-hoc method for structure from motion. Some of the 're-invention' and 'home-brewing' of ad-hoc methods for structure from motion have been avoided by rigorous and systematic exposition of bundle adjustment to the computer vision community, such as for example by (Triggs et al., 2000), but it is still an occurring phenomenon. Several researchers have previously developed systems that can perform real-time structure from motion without bundle adjustment, see e.g. (Davison and Murray, 2002, Nistér et al., 2006).

Admittedly, it is typically not possible in real-time applications to incorporate information from video frames further along in the sequence, as this would cause unacceptable latency. However, bundle adjustment does not necessarily require information from future video frames. In fact, bundle adjustment of as many frames backwards as possible each time a frame is added, will provide the best accuracy possible using only information up to the current time. If such bundle adjustment can be performed within the time-constraints of the application at hand, there is really no good excuse for not using it. We investigate the computation time required by an efficient implementation of bundle adjustment geared specifically at real-time camera tracking when various amounts of frames are included in the bundle adjustment. We then combine the failure rate experiments with our timing experiments to provide information on how much the failure rate can be decreased given various amounts of computation time, showing that bundle adjustment is an important component of a state-of-the-art real-time camera tracking system.

2 THEORETICAL BACKGROUND AND IMPLEMENTATION

In this section we describe the bundle adjustment process and the details of our implementation. For readers familiar with the details of numerical optimization and bundle adjustment, the main purpose of this section is simply to avoid any confusion regarding the exact implementation of the bundle adjuster used in the experiments. For readers who are less familiar with this material, this section also gives an introduction to bundle adjustment, which seems appropriate given that this paper argues for bundle adjustment.

A very large class of minimization schemes try to minimize a cost function $c(x)$ iteratively by approximating the cost function locally around the current (M-dimensional) position x with a quadratic Taylor expansion

$$c(x+dx) \approx c(x) + \nabla c(x)^\top dx + \frac{1}{2} dx^\top H_c(x) dx \quad (1)$$

where $\nabla c(x)$ is the gradient

$$\nabla c(x) = \begin{bmatrix} \frac{\partial c}{\partial x_1}(x) & \dots & \frac{\partial c}{\partial x_M}(x) \end{bmatrix}^\top \quad (2)$$

of c at x and $H_c(x)$ is the Hessian

$$H_c(x) = \begin{bmatrix} \frac{\partial^2 c}{\partial x_1 \partial x_1}(x) & \dots & \frac{\partial^2 c}{\partial x_1 \partial x_M}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 c}{\partial x_N \partial x_1}(x) & \dots & \frac{\partial^2 c}{\partial x_N \partial x_M}(x) \end{bmatrix} \quad (3)$$

of c at x. By taking the derivative of (1) and equating to zero, one obtains

$$H_c(x)dx = -\nabla c(x), \quad (4)$$

which is a linear equation for the update vector dx. Since there is no guarantee that the quadratic approximation will lead to an update dx that improves the cost function, it is very common to augment the update so that it goes towards small steps down the gradient when improvement fails. There are many ways to do this since any method that varies between the update defined by (4) and smaller and smaller steps down the gradient will suffice in principle. For example, one can add some scalar λ to all the diagonal elements of $H_c(x)$. When improvement succeeds, we decrease λ towards zero, since at $\lambda = 0$ we get the step defined by the quadratic approximation, which will ultimately lead to fast convergence near the minimum. When improvement fails, we increase λ, which makes the update tend towards

$$dx = -\frac{1}{\lambda} \nabla c(x), \quad (5)$$

which guarantees that improvement will be found for sufficiently large λ (barring numerical problems).

Typically, the cost function is the square sum of all the dimensions of an (N-dimensional) error vector function $f(x)$:

$$c(x) = f(x)^\top f(x). \quad (6)$$

Note that the error vector f can be defined in such a way that the square sum represents a robust cost function, rather than just an outlier-sensitive plain least squares cost function.

We use, as is very common, the so-called Gauss-Newton approximation of the Hessian, which comes from approximating the vector function $f(x)$ around x with the first order Taylor expansion

$$f(x+dx) \approx f(x) + J_f(x)dx, \quad (7)$$

where $J_f(x)$ is the Jacobian

$$J_f(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x) & \dots & \frac{\partial f_1}{\partial x_M}(x) \\ \vdots & \vdots & \vdots \\ \frac{\partial f_N}{\partial x_1}(x) & \dots & \frac{\partial f_N}{\partial x_M}(x) \end{bmatrix} \quad (8)$$

of f at x. Inserting (7) into (6), we get

$$c(x+dx) \approx f^\top f(x) + 2f^\top J_f(x)dx + dx^\top J_f^\top J_f(x)dx, \quad (9)$$

which by equating the derivative to zero results in the update equation

$$J_f(x)^\top J_f(x)dx = -J_f(x)^\top f(x). \quad (10)$$

By noting that $2J_f(x)^\top f(x)$ is the exact gradient of (6) and comparing with (4) one can see that the Hessian has been approximated by

$$H_c(x) \approx 2J_f(x)^\top J_f(x). \quad (11)$$

The great advantage of this is that computation of second derivatives is not necessary. Another benefit is that this Hessian approximation (and its inverse) is normally positive definite (unless the Jacobian has a nullvector), that is

$$dx^\top J_f(x)^\top J_f(x)dx > 0 \quad \forall dx \neq 0, \quad (12)$$

which opens up more ways of accomplishing the transition towards small steps that guarantee improvement in the cost function. For example, instead of adding λ to the diagonal, we can multiply the diagonal of $J_f(x)^\top J_f(x)$ by the scalar $(1+\lambda)$, which leads to the Levenberg-Marquardt algorithm. This is guaranteed to eventually find an improvement, because an update dx with a sufficiently small magnitude and a negative scalar product with the gradient is guaranteed to do so, and when λ increases, the update tends towards

$$dx = -\frac{1}{\lambda} diag(J_f(x)^\top J_f(x))^{-1} J_f(x)^\top f(x), \tag{13}$$

(where $diag(.)$ stands for the diagonal of a matrix), which is minus the gradient times a small positive diagonal matrix. Yet another update strategy that guarantees improvement, but without solving the linear system for each new value λ, is to upon failure divide the update step by λ, resulting in a step that tends towards

$$dx = -\frac{1}{\lambda}(J_f(x)^\top J_f(x))^{-1} J_f(x)^\top f(x), \tag{14}$$

which is minus the gradient times a small positive definite matrix. With this strategy, only the cost function needs to be reevaluated when λ is increased upon failure to improve, which can be an advantage if the cost function is cheap to evaluate, but the linear system expensive to solve. In our implementation, we use the Levenberg-Marquardt variant.

The core feature of a bundle adjuster (compared to standard numerical optimization) is to take advantage of the so-called primary structure (sparsity), which arises because the parameters for scene features (in our case 3D points) and sensors combine to predict the measurements, while the scene feature parameters do not combine directly and the sensor parameters do not combine directly. More precisely, the error vector f consists of some reprojection error (some difference measure between the predicted and the measured reprojections), which can be made robust by applying a nonlinear mapping that decreases large errors, and the Jacobian J_f has the structure

$$J_f = \begin{bmatrix} J_P & J_C \end{bmatrix}, \tag{15}$$

where J_P is the Jacobian of the error vector f with respect to the 3D point parameters and J_C is the Jacobian of the error vector f with respect to the camera parameters. This results in the Hessian approximation

$$H = \begin{bmatrix} J_P^\top J_P & J_P^\top J_C \\ J_C^\top J_P & J_C^\top J_C \end{bmatrix}, \tag{16}$$

which in the linear system may possibly have an augmented diagonal. The whole linear equation system becomes

$$\begin{bmatrix} H_{PP} & H_{PC} \\ H_{PC}^\top & H_{CC} \end{bmatrix} \begin{bmatrix} dP \\ dC \end{bmatrix} = \begin{bmatrix} b_P \\ b_C \end{bmatrix}, \tag{17}$$

where we have defined $H_{PP} = J_P^\top J_P$, $H_{PC} = J_P^\top J_C$, $H_{CC} = J_C^\top J_C$, $b_P = -J_P^\top f$,$b_C = -J_C^\top f$ to simpify the notation, and dP and dC represent the update of the point parameters and the camera parameters, respectively. Note that the matrices H_{PP} and H_{CC} are block-diagonal, where the blocks correspond to points and cameras, respectively. In order to take advantage of this block-structure, a block-wise Gaussian elimination is now applied to (17). First we multiply by

$$\begin{bmatrix} H_{PP}^{-1} & 0 \\ 0 & I \end{bmatrix} \tag{18}$$

from the left on both sides in order to get the upper left block to identity, resulting in

$$\begin{bmatrix} I & H_{PP}^{-1} H_{PC} \\ H_{PC}^\top & H_{CC} \end{bmatrix} \begin{bmatrix} dP \\ dC \end{bmatrix} = \begin{bmatrix} H_{PP}^{-1} b_P \\ b_C \end{bmatrix}, \tag{19}$$

Then we subtract $H_{PC}^\top$ times the first row from the second row in order to eliminate the lower left block. This can also be thought of as multiplying by

$$\begin{bmatrix} I & 0 \\ -H_{PC}^\top & I \end{bmatrix} \tag{20}$$

from the left on both sides, resulting in the smaller equation system (from the lower part)

$$\underbrace{(H_{CC} - H_{PC}^\top H_{PP}^{-1} H_{PC})}_{A} dC = \underbrace{b_C - H_{PC}^\top H_{PP}^{-1} b_P}_{B} \tag{21}$$

for the camera parameter update dC. For very large systems, the left hand side is still a sparse system due to the fact that not all scene features appear in all sensors. In contrast to the primary structure, this secondary structure depends on the observed tracks, and is hence hard to predict. This makes the sparsity less straightforward to take advantage of. Typical options are to use profile Cholesky factorization with some appropriate on-the-fly variable ordering, or preconditioned conjugate gradient to solve the system. We use straightforward Cholesky factorization. As we shall see, for the size of system resulting from a few tens of cameras, the time required to form the left hand side matrix largely dominates the time necessary to solve the system with straightforward Cholesky factorization. This occurs because the time taken to form the matrix is on the order of $O(N_P l^2)$, where N_P is the number of tracks and l is a representative track length. Since N_P is typically rather large, and l on the order of the number of cameras N_C for smaller systems, this dominates the order of $O(N_C^3)$ time taken to solve the linear system, until N_C starts approaching N_P or largely dominating l.

Once dC has been found, the point parameter updates can be found from the upper part of (19) as

$$dP = H_{PP}^{-1} b_P - H_{PP}^{-1} H_{PC} dC. \tag{22}$$

Since an efficient implementation of the actual computation process corresponding to this description is somewhat involved, we find it helpful to summarize the bundle adjustment process in pseudo-code in Table 1.

The main computation steps that may present bottlenecks are

- The computation of the cost function (which grows linearly in the number of reprojections).
- The computation of derivatives and accumulation over tracks (linear in the number of reprojections).
- The outer product over tracks (which grows with the square of the track lengths times the number of tracks, or thought of another way, approximately the number of reprojections times a representative track length).
- Solving the linear system (which grows with the cube of the number of cameras, unless secondary structure is exploited).
- The back-substitution (linear in the number of reprojections).

Accordingly, these are the computation steps for which we measure timing in the experiments, as well as a total computation time.

We use a calibrated camera model in the experiments, so that the only camera parameters solved for are related to the rotation and translation of the camera. The parameterization used in bundle adjustment is straightforward, with three parameters for translation of each camera, and the sines of three Euler angles parameterizing an incremental rotation from the current position (notice

```
1  Initialize $\lambda$.
2  Compute cost function at initial camera and point configuration.
3  Clear the left hand side matrix $A$ and right hand side vector $B$.
4  For each track $p$
   {
     Clear a variable $H_{pp}$ to represent block $p$ of $H_{PP}$ (in our case a symmetric $3 \times 3$ matrix) and a variable $b_p$ to represent part $p$ of $b_P$ (in our case a 3-vector).
     (Compute derivatives) For each camera $c$ on track $p$
     {
       Compute error vector $f$ of reprojection in camera $c$ of point $p$ and its Jacobians $J_p$ and $J_c$ with respect to the point parameters (in our case a $2 \times 3$ matrix) and the camera parameters (in our case a $2 \times 6$ matrix), respectively.
       Add $J_p^\top J_p$ to the upper triangular part of $H_{pp}$.
       Subtract $J_p^\top f$ from $b_p$.
       If camera $c$ is free
       {
         Add $J_c^\top J_c$ (optionally with an augmented diagonal) to upper triangular part of block $(c, c)$ of left hand side matrix $A$ (in our case a $6 \times 6$ matrix).
         Compute block $(p, c)$ of $H_{PC}$ as $H_{pc} = J_p^\top J_c$ (in our case a $3 \times 6$ matrix) and store it until track is done.
         Subtract $J_c^\top f$ from part $c$ of right hand side vector $B$ (related to $b_C$).
       }
     }
     Augment diagonal of $H_{pp}$, which is now accumulated and ready. Invert $H_{pp}$, taking advantage of the fact that it is a symmetric matrix.
     Compute $H_{pp}^{-1} b_p$ and store it in a variable $t_p$.
     (Outer product of track) For each free camera $c$ on track $p$
     {
       Subtract $H_{pc}^\top t_p = H_{pc}^\top H_{pp}^{-1} b_p$ from part $c$ of right hand side vector $B$.
       Compute the matrix $H_{pc}^\top H_{pp}^{-1}$ and store it in a variable $T_{pc}$
       For each free camera $c2 \geq c$ on track $p$
       {
         Subtract $T_{pc} H_{pc2} = H_{pc}^\top H_{pp}^{-1} H_{pc2}$ from block $(c, c2)$ of left hand side matrix $A$.
       }
     }
   }
5  (Optional) Fix gauge by freezing appropriate coordinates and thereby reducing the linear system with a few dimensions.
6  (Linear Solving) Cholesky factor the left hand side matrix $B$ and solve for $dC$. Add frozen coordinates back in.
7  (Back-substitution) For each track $p$
   {
     Start with point update for this track $dp = t_p$.
     For each camera $c$ on track $p$
     {
       Subtract $T_{pc}^\top dc$ from $dp$ (where $dc$ is the update for camera $c$).
     }
     Compute updated point.
   }
8  Compute the cost function for the updated camera and point configuration.
9  If cost function has improved, accept the update step, decrease $\lambda$ and go to Step 3 (unless converged, in which case quit).
10 Otherwise, increase $\lambda$ and go to Step 3 (unless exceeded the maximum number of iterations, in which case quit).
```

Table 1: Pseudo-code showing our implementation.

that the latter has no problems with singularities since the parameterization is updated after each parameter update step, and the rotation updates should never be anywhere close to 90 degrees. For the 3D points, we use the four parameters of a homogeneous coordinate representation, but we always freeze the coordinate with the largest magnitude, the choice of which coordinate to freeze being updated after each parameter update step.

To robustify the reprojection error, we assume that the reprojection errors have a Cauchy-distribution (which is a heavy-tailed distribution), meaning that an image distance of e between the measured and reprojected distance should contribute

$$\ln(1 + \frac{e^2}{\sigma^2}), \tag{23}$$

where σ is a standard deviation, to the cost function (negative log-likelihood). To accomplish this, while still exposing both the horizontal and vertical component of error in the error vector f, the robustifier takes the input error (x, y) in horizontal and vertical direction and outputs the robustified error vector (x_r, y_r) where

$$x_r = \sqrt{\ln(1 + \frac{x^2 + y^2}{\sigma^2})} \frac{x}{\sqrt{x^2 + y^2}} \tag{24}$$

$$y_r = \sqrt{\ln(1 + \frac{x^2 + y^2}{\sigma^2})} \frac{y}{\sqrt{x^2 + y^2}}. \tag{25}$$

The key property of this vector is that the square sum of its component is (23), while balancing the components exactly as the original reprojection error.

3 EXPERIMENTS

We investigate the failure rate of camera tracking with n iterations of bundle adjustment over the last m video frames each time a frame is added, for various values of n and m. The frames beyond the m most recent frames are locked down and not moved in the bundle adjustment. However, the information regarding the uncertainty in reconstructed feature points provided by views that are locked down is still used. That is, reprojection errors are accumulated for the entire feature track lengths backwards in time, regardless of whether the views where the reprojections reside are locked down.

In the beginning of tracking, when the number of frames yet included is less than $m + 2$ so that at most one pose is locked, the gauge is fixed by fixing the first camera pose and the distance between the first and the most current camera position. Otherwise, the gauge fixing is accomplished by the locked views.

It is interesting to note that when we set $m = n = 1$, we get an algorithm that rather closely resembles a simple Kalman filter. We then essentially gather the covariance information induced on the 3D points by all previous views and then update the current pose based on that (using a single iteration), which should be at least as good as a Kalman filter that has a state consisting of independent 3D points, with the potential improvement that the most recent estimates of the 3D point positions are used when computing reprojection errors and their derivatives in previous views.

Note that bundle adjustment as well as Kalman filtering for the application of camera tracking can be used both with or without a camera motion model. We have chosen to concentrate on experiments for the particular case of no camera motion model, i.e. no smoothness on the camera trajectory is imposed, and the only

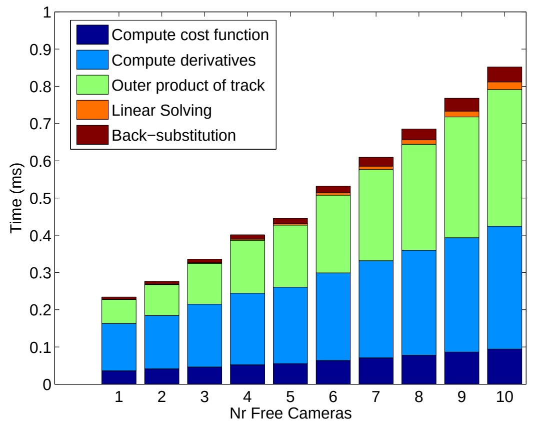


Figure 2: Time per iteration versus the number of free views. Time is dominated by derivative computations and outer products, while linear solving takes negligible time.

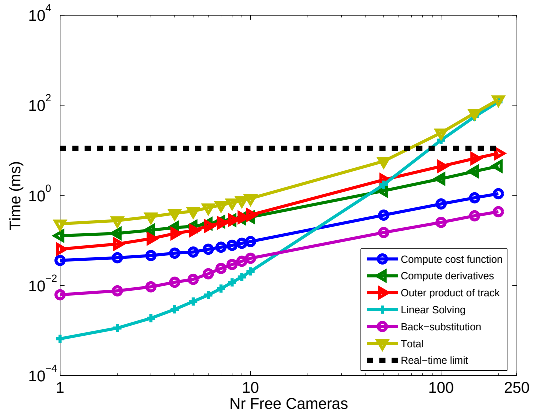


Figure 3: Time per iteration versus the number of free views for larger numbers of free views. Note that the cubic dependence of the computation time in the linear solving on the number of free views eventually makes the linear solving dominate the computation time. The real-time limit is computed as $1s/(30*3)$

constraints on the camera trajectory are the reprojection errors of the reconstruction of tracked feature points. The no motion model case is the most flexible, but also the hardest setting in which to perform estimation. It therefore most clearly elucidates the issue. While a motion model certainly simplifies the estimation task when the motion model holds, it unfortunately makes the 'hard cases even harder', in that when the camera performs unexpected turns or jerky motion, the motion model induces a bias towards status quo motion.

We measure the failure rate by defining a frame-to-frame failure criterion and running long sequences, restarting the estimation from scratch at the current position every time the failure criterion declares that a failure has occurred. Note that this failure criterion is not part of an algorithm, but only a means of measuring the failure rate. For our real-data experiments, we perform simple types of known motion, such as forward, diagonal, sideways or turntable motion, and require that the translation direction and rotation do not deviate more than some upper limit from the known values.

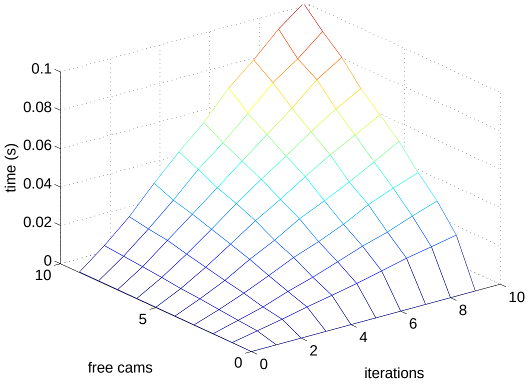


Figure 4: Computation time versus number of free views and number of iterations.

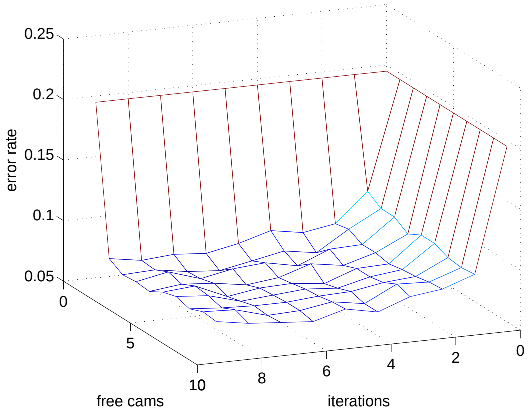


Figure 5: Failure rate as a function of the number of free cameras and the number of iterations.

The initialization that occurs in the beginning and after each failure is accomplished using the first three frames and a RANSAC (Fischler and Bolles, 1981) process using the five-point relative orientation method in the same manner as described in (Nistér, 2004). In each RANSAC hypothesis, the five-point method provides hypotheses for the first and third view. The five points are triangulated and the second view is computed by a three-point resection (R. Haralick, 1994). The whole three-view initialization is then thoroughly bundle adjusted to provide the best possible initialization.

4 METHOD

For each new single view that is added, the camera position is initialized with a RANSAC process where hypotheses are generated with three-point resections. The points visible in the new view are then re-triangulated using the reprojection in the new view and the first frame where the track was visible. The bundle adjustment with n iterations of the m most recent views is then performed. In both the RANSAC processes and the bundle adjustment the cost function is a robustified reprojection error. Thus, we do not attempt to throw out outlying tracks before the bundle adjustment, but use all tracks and instead employ a robust cost function.

5 RESULTS AND DISCUSSION

Representative computation time measurements are shown in Figures 2, 3 and 4. In Figure 2, the distribution of time over cost function computation, derivatives, outer product, linear solving and back-substitution is shown as a bar-plot for a small number of free views.

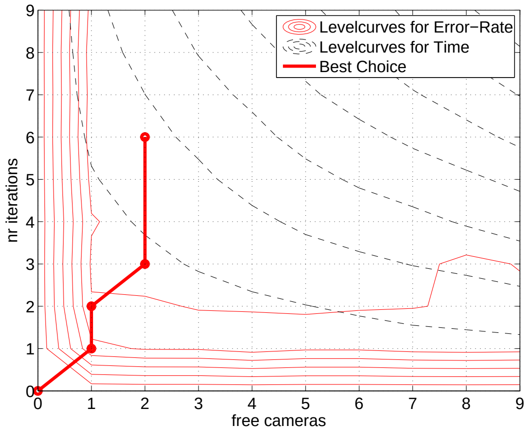


Figure 6: Level curves for time and error rate

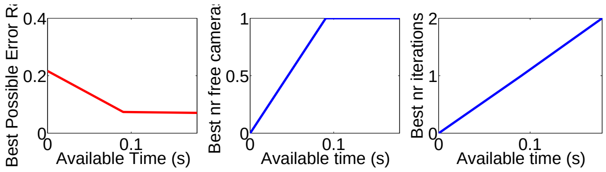


Figure 7: The first plot shows best possible quality for a given computation time. The second and third plots show the m and n values to achieve these results.

The time is per new view and iteration and was measured on the sequence shown in Figure 1, which has 640×480 resolution and 1000 frames in total. The computations were performed on an Alienware with Intel Pentium Xeon processor 3.4GHz, 2.37GB. The average number of tracks present in a frame is 260, and the average track length is 20.5. Note that the computation time is dominated by derivative computation and outer products. Note also that real-time rate can easily be maintained for ten free views. Figure 3 shows how the computation times grow with an increasing number of views. Note that the linear solving, due to the cubic dependence on the number of views eventually becomes the dominant computational burden and that real-time performance is lost somewhere between 50 and 100 free views. Note however that the other computation tasks scale approximately linearly (since the track lengths are limited). When the track length for each feature is constant, increasing the number of feature points results in a linear increase in computation time for all steps in the bundle adjustment except for the linear solver, which is independent of this number.

Some investigations of the failure rate as well as computation time for various amounts of bundle adjustment are shown in Figures 5, 6 and 7. Note that already one iteration on just the most recent view results in a significant decrease of the failure rate, but to get the full benefit of bundle adjustment and 'reach the valley floor', suppressing failures as much as possible, three iterations on three views or perhaps even four iterations on four views is desirable. Note that this does not necessarily mean that additional accuracy can not be gained with more iterations over more views, only that the gross failures have largely been stemmed after that. Also, with bundle adjustment very low failure rates can often be achieved. For example the sequence in Figure 1 can be tracked completely without failures for 1000 frames and over seven laps. Decreases in failure rate at such low failure rates require large amounts of data and computation to measure, but are clearly still very valuable.

In Figure 6 the level curves over the (n, m) space are shown for computation time and failure rate. The 'best path' in the (n, m) space is also shown there and in Figure 7, meaning that for a given amount of computation allowed, the n and m resulting in the lowest failure rate is chosen.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we have investigated experimentally the fact that bundle adjustment does not only increase the accuracy of the camera trajectory, but also prevents error-buildup in a way that decreases the frequency of total failure of the camera tracking. We have also shown that with the current computing power in standard computing platforms, efficient implementations of bundle adjustment now provide a very viable option even for real-time applications.

We have found that bundle adjustment can be performed for a few tens of the most recent views every time a new frame is added, while still maintaining video rate. At such numbers of free views, the most significant components of the computation time are related to the computation of derivatives of the cost function with respect to camera and 3D point parameters, plus the outer products that the feature tracks contribute to the Schur complement arising when eliminating to obtain a linear system for the camera parameter updates. The actual linear solving of the linear system takes negligible time in comparison for a small number of views, but the linear solver, which has a cubic cost in the number of views, eventually becomes the dominating computation when a number of views approaching a hundred are bundle adjusted every time a new frame arrives. This can probably be improved upon by exploiting the secondary structure that still remains in the linear system, which is something we hope to do in future work.

Our results are, as expected, a strong proof that proper bundle adjustment is more efficient that any ad-hoc structure from motion refinement, and the results are well worth the trouble of proper implementation.

REFERENCES

Davison, A. and Murray, D. W., 2002. Simultaneous Localization and Map-Building Using Active Vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), pp. 865–880.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications-of-the-ACM 24(6), pp. 381–95.

Fitzgibbon, A. W. and Zisserman, A., 1998. Automatic camera recovery for closed or open image sequences. In: ECCV (1), pp. 311–326.

Nistér, D., 2001. Automatic Dense Reconstruction from Uncalibrated Video Sequences. PhD thesis, Royal Institute of Technology, Stockholm, Sweden.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. PAMI 26(6), pp. 756–777.

Nistér, D., Naroditsky, O. and Bergen, J., 2006. Visual odometry for ground vehicle applications. Journal of Field Robotics 23(1), pp. 3–20.

Pollefeys, M., 1999. Self-Calibration and Metric 3D Reconstruction From Uncalibrated Image Sequences. PhD thesis, K.U.Leuven, Belgium.

R. Haralick, C. L. K. Ottenberg, M. N., 1994. Review and analysis of solutions of the three point perspective pose estimation problem. IJCV 13, pp. 331–356.

Triggs, W., McLauchlan, P., Hartley, R. and Fitzgibbon, A., 2000. Bundle adjustment: A modern synthesis. In: W. Triggs, A. Zisserman and R. Szeliski (eds), Vision Algorithms: Theory and Practice, LNCS, Springer Verlag.

Notes

Author Index

Aloimonos, Y. 260
Bacher, U. 209
Bähr, H.-P. 67
Baltsavias, E. 209
Bamler, R. 240
Batlle, E. 61
Becker, S. 19
Boldo, D. 111
Boudet, L. 136
Brenner, C. 155
Briese, C. 98
Brzank, A. 197
Butenuth, M. 1
Cervelle, B. 73
Champion, N. 111
Choi, Y. 148
d'Angelo, P. 124
Denzler, J. 37
Domke, J. 260
Drauschke, M. 7
Durupt, M. 142
Engels, C. 266
Fofi, D. 61
Förstner, W. 7
Gallis, R. de A. 215
Göpfert, J. 79
Grammatikopoulos, L. 255
Gross, H. 86, 185
Gruber, B. 31
Grussenmeyer, P. 25
Haala, N. 19
Haring, A. 98
Havasi, L. 167
Hedman, K. 55
Heinrichs, M. 43
Heipke, C. 79, 179, 197
Heuwold, J. 173
Hinz, S. 55, 203, 209, 221, 240
Jung, F. 73, 136
Jutzi, B. 234
Kada, M. 19
Kähler, O. 37
Kalisperakis, I. 255
Karner, K. 31
Karras, G. 255
Kirchhof, M. 161
Klaus, A. 31
Korpela, I. 191
Krzystek, P. 228
Laika, A. 240
Landes, T. 25
Lee, I. 148
Lee, Y. J. 148
Leitloff, J. 221
Lenhart, D. 203
Lohmann, P. 179
Martinoty, G. 136
Matabosch, C. 61
Mayer, H. 130, 209
Meidow, J. 161
Meyer, F. 240
Michaelsen, E. 161
Nistér, D. 266
Oude Elberink, S. 92
Paparoditis, N. 73, 136
Park, J. 148
Pierrot-Deseilligny, M. 136
Poz, A. P. Dal 215
Rehor, M. 67
Reitberger, J. 228
Remondino, F. 49, 117, 248
Ressl, C. 98, 248
Reznik, S. 130
Ripperda, N. 155
Rodehorst, V. 43
Rottensteiner, F. 13, 98
Salvi, J. 61
Schuster, H.-F. 7
Silva, J. F. C. da 215
Smigiel, E. 25
Soergel, U. 185
Stewénius, H. 266
Stilla, U. 55, 161, 221, 228, 234
Suchandt, S. 240
Szirányi, T. 167
Taillandier, F. 142
Tarsha-Kurdi, F. 25
Thiele, A. 185
Thoennessen, U. 86, 185
Tournaire, O. 73
von Hansen, W. 105, 161
Vosselman, G. 92
Wöhler, C. 124
Yang, Y. F. 179
Zebedin, L. 31
Zhang, L. 117

Notes