Int J Comput Vis (2011) 94:154–174
DOI 10.1007/s11263-010-0340-z

Coding Images with Local Features

Timo Dickscheid · Falko Schindler · Wolfgang Förstner

Received: 21 September 2009 / Accepted: 8 April 2010 / Published online: 27 April 2010

Abstract We develop a qualitative measure for the completeness and complementarity of sets of local features in terms of covering relevant image information. The idea is to interpret feature detection and description as image coding, and relate it to classical coding schemes like JPEG. Given an image, we derive a *feature density* from a set of local features, and measure its distance to an *entropy density* computed from the power spectrum of local image patches over scale. Our measure is meant to be complementary to existing ones: After task usefulness of a set of detectors has been determined regarding robustness and sparseness of the features, the scheme can be used for comparing their completeness and assessing effects of combining multiple detectors. The approach has several advantages over a simple comparison of image coverage: It favors response on structured image parts, penalizes features in purely homogeneous areas, and accounts for features appearing at the same location on different scales. Combinations of complementary features tend to converge towards the entropy, while an increased amount of random features does not. We analyse the complementarity of popular feature detectors over different image categories and investigate the completeness of combinations. The derived entropy distribution leads to a new scale and rotation invariant window detector, which uses a fractal image model to take pixel correlations into account. The results of our empirical investigations reflect the theoretical concepts of the detectors.

Keywords Local features · Complementarity · Information theory · Coding · Keypoint detectors · Local entropy

T. Dickscheid (✉) · F. Schindler · W. Förstner
Department of Photogrammetry, Institute of Geodesy and Geoinformation, University of Bonn, Bonn, Germany
e-mail: dickscheid@uni-bonn.de

F. Schindler
e-mail: falko.schindler@uni-bonn.de

W. Förstner
e-mail: wf@ipb.uni-bonn.de

1 Introduction

Local image features play a crucial role in many computer vision applications. The basic idea is to represent the image content by small, possibly overlapping, independent parts. By identifying such parts in different images of the same scene or object, reliable statements about image geometry and scene content are possible. Local feature extraction comprises two steps: (1) *Detection* of stable local image patches at salient positions in the image, and (2) *description* of these patches. The descriptions usually contain a lower amount of data compared to the original image intensities. The SIFT descriptor (Lowe 2004), for example, represents each local patch by 128 values at 8 bit resolution independent of its size.

This paper is concerned with feature detectors. Important properties of a good detector are:

1. *Robustness*. The features should be robust against typical distortions such as image noise, different lighting conditions, and camera movement.
2. *Sparseness*. The amount of data given by the features should be significantly smaller compared to the image itself, in order to increase efficiency of subsequent processing.
3. *Speed*. A feature detector should be fast.
4. *Completeness*. Given that the above requirements are met, the information contained in an image should be preserved by the features as much as possible. In other

words, the amount of information coded by a set of features should be maximized, given a desired degree of robustness, sparseness, and speed.

Popular detectors have been investigated in depth regarding Item 1, and there exists some common sense about the most robust detectors for typical application scenarios. Sparseness often depends on the parameter settings of a detector, especially on the significance level used to separate the noise from the signal. The speed of a detector should be characterized referring to a specific implementation.

Item 4 has not received much attention yet. To our knowledge, no pragmatic tool for quantifying the completeness of a specific set of features w.r.t. image content is available. This may be due to the differences in the concepts of the detectors, making such a statement difficult. As shown in Fig. 1, for example, blob-like features often cover much of the areas representing visible objects, while the characteristic contours are better coded by junction features and straight edge segments.

Complementarity of features plays a key role when using multiple detectors in an application. It is strongly related to completeness, as the information coded by sets of complementary features is higher than that coded by redundant feature sets. The detectors shown in Fig. 1 complement each other very well. Using all three, most relevant parts of the images are taken into account. Such complementarity is often taken for granted, but cannot always be expected. A tool for quantifying it would be highly desirable.

Our goal is to develop a qualitative measure for evaluating how far a specified set of detectors covers relevant image content completely and whether the detectors are complementary in this sense. We would also like to know if completeness over different image categories reflects the theoretical concepts behind well-known detectors.

This requires us to find a suitable representation of what we consider as "relevant image content". We want to follow an information theoretical approach, where image content is characterized by the number of bits using a certain coding scheme. Then it is desirable to have strong feature response at locations where most bits are needed for a faithful coding, while features on purely homogeneous areas are less important. Therefore we will derive a measure d for the incompleteness of local features sets, which takes small values if a feature set covers image content in a similar manner as a good compression algorithm would. We will model d as the difference between a feature coding density p_c derived from a set of local features, and an entropy density p_H, both related to a particular image. The entropy density is closely related to the image coding scheme used in JPEG. Although

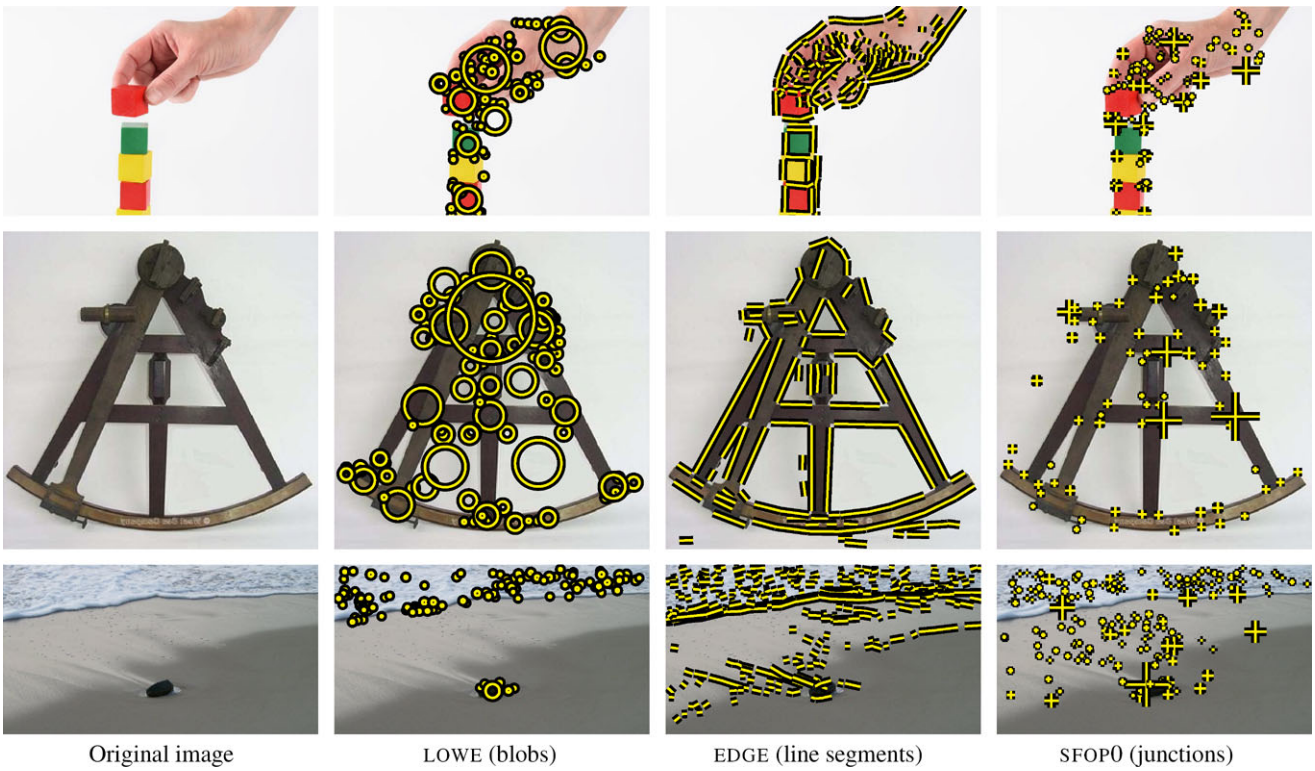


Fig. 1 Sets of extracted features on three example images. The feature detectors here are substantially different: LOWE (Lowe 2004) fires at dark and bright blobs, EDGE (Förstner 1994) yields straight edge segments, and SFOP0 (Förstner et al. 2009, using $\alpha = 0$ in (7)) extracts junctions

not being a precise model of the image, we will show that it is a reasonable reference.

We do not intend to give a general benchmark for detectors. More specifically, we do not claim sets of features with highest completeness to be most suitable for a particular application. Task usefulness can only be determined by investigating all of the four properties listed above. Therefore we assume that a set of feature detectors with comparable sparseness, robustness and speed is preselected before using our evaluation scheme.

The entropy density p_H also leads to a new scale and rotation invariant window detector, which explicitly searches for locations over scale having maximum entropy. The detector uses a fractal image model to take pixel correlations into account. It will be described and included in the investigations.

The paper is organized as follows. In Sect. 2 we will give an overview on popular feature detectors and related work in image statistics. Section 3 covers the derivation of the entropy and feature coding densities together with an appropriate distance metric d, and introduces the new entropy-based window detector. An evaluation scheme based on d is described in Sect. 4.1, followed by experimental results for various detectors over different image categories, including a broad range of mixed feature sets (Sect. 4.2). We finally conclude with a summary and outlook in Sect. 5.

2 Related Work

2.1 Local Feature Detectors

Tuytelaars and Mikolajczyk (2008) emphasized that most local feature detectors are in fact *extractors*, and that the features themselves are usually *covariant* w.r.t. distortions of the image. However we use the term "detector" for all procedures which extract some relevant features, irrespective of their invariance properties.

A broad range of local feature detectors with different properties is available today. They are often divided into corner detectors, blob detectors and region detectors, the latter two representing two-dimensional features as opposed to corners, which have dimension zero. While blobs may also be seen as regions referring to their dimension, the distinction is reasonable: With "blobs" we usually denote features attached to a particular pixel position, representing dark or bright areas around the pixel, while regions are explicitly determined by their boundaries. We also need to take detectors of one dimensional features into account, namely edge detectors, as they explicitly focus on boundaries of regions and possibly build corner and junction points.

We give a short overview on the most popular detectors, and refer to Tuytelaars and Mikolajczyk (2008) for a detailed description. Before we start however, we want to mention the interesting work of Corso and Hager (2009), who search for different scalar features arising from kernel-based projections that summarize content. This approach addresses our idea of preferring complementary feature sets, but is very different from classical feature detectors.

2.1.1 *Blob and Region Detectors*

The scale invariant blob detector proposed by (Lowe 2004), here denoted as LOWE, is by far the most prominent one. It is based on finding local extrema of the Laplacian of Gaussians (LoG) $\nabla^2_\tau g = \nabla^2_\tau * g$, where g is the image function and τ is the scale parameter, identified with the width of the smoothing kernel for building the scale space. The LoG has the well-known Mexican hat form, therefore the detector conceptually aims at extracting dark and bright blobs on characteristic scales of an image. The underlying principle of scale localization has already been described by Lindeberg (1998b). To gain speed, the LOWE detector approximates the LoG by Difference of Gaussians (DoG).

The Hessian affine detector (HESAF) introduced by Mikolajczyk and Schmid (2004) is theoretically related to LOWE, as it is also based on the theory of Lindeberg (1998b) and relies on the second derivatives of the image function over scale space. It evaluates both the determinant and the trace of the Hessian of the image function, the latter one being identical to the Laplacian introduced above. In Mikolajczyk et al. (2003), the response of an edge detector is evaluated on different scales for locating feature points. Then, in a similar fashion as for HESAF, a maximum in the Laplacian scale space is searched at each of these locations to obtain scale-invariant points. Therefore this detector, referred to as EDGELAP, computes a high amount of blobs located near edges with some non-homogeneous signal on at least one side.

Regions, as opposed to blobs, determine feature windows based on their boundary, and thus have a strong relation to image segmentation methods. A very prominent affine region detector is the Maximally Stable Extremal Region detector (MSER) proposed by Matas et al. (2004). The idea is to compute a watershed-like segmentation with varying thresholds, and to select such regions that remain stable over a range of thresholds. The MSER detector is known to have very good repeatability especially on objects with planar structures, and is widely used especially for object recognition. Another example is the intensity-based region detector IBR proposed by Tuytelaars and Van Gool (2004). Here, local maxima of the intensity function are first detected over multiple scales. Subsequently a region boundary is determined by seeking for peaks of the intensity function along rays radially emanating around these points. It has been observed that the feature sets extracted by MSER and IBR are

fairly similar (Tuytelaars and Mikolajczyk 2008). The direct output of both algorithms can be any closed boundary of a segmented region. Within this work however, we will model feature patches by elliptical shapes, following most existing evaluations.

A different approach is been taken by Kadir and Brady (2001) who estimate the entropy of local image attributes over a range of scales and extract points exhibiting entropy peaks in this space with significant magnitude change of the local probability density. Their "salient regions" detector (SALIENT) has been extended to affine invariance later (Kadir et al. 2004). It typically extracts many features, but is known to have significantly higher computational complexity and lower repeatability than most other detectors (Mikolajczyk et al. 2005).

2.1.2 Corner and Junction Detectors

Corners have been used extensively since the early works of Förstner and Gülch (1987) and Harris and Stephens (1988). Both of these detectors in principle provide rotation invariance and good localization accuracy. The detectors are based on the second moment matrix, or structure tensor,

$$M_{\tau,\sigma} = \overline{\nabla_\tau g \nabla_\tau g^\mathsf{T}} = G_\sigma * \begin{bmatrix} g_{x,\tau}^2 & g_{x,\tau} g_{y,\tau} \\ g_{x,\tau} g_{y,\tau} & g_{y,\tau}^2 \end{bmatrix}, \quad (1)$$

computed from the dyadic products of the image gradients. Here, τ is the smoothing parameter used for computing the derivatives, and σ used for specifying the size of the integration window. Usually they are tied, i.e. by $\tau(\sigma) = \sigma/3$. Strictly speaking, these detectors are window detectors: They compute optimal local patches for describing a feature instead of optimal feature locations, and usually fire close to junctions. Window detectors also cover circular symmetric (Förstner and Gülch 1987) and spiral type image features (Bigün 1990). More specifically, spirals are a generalization of junctions and circular symmetric features.

In recent applications rotation invariance is usually not sufficient. Therefore a number of attempts for exploiting the structure tensor over scale space have been proposed. The Harris affine (HARAF) detector (Mikolajczyk and Schmid 2004) computes the structure tensor on multiple scales to detect 2D extrema within each scale. The final points are determined by locating characteristic scales at these positions using the Laplacian, which effectively fires at blobs. This way the corners seen in the image usually get lost, as they do not exhibit extrema in the Laplacian. We therefore believe that HARAF should be considered a blob detector, in contrast to Tuytelaars and Mikolajczyk (2008) who refer to it as a corner detector.

Lindeberg (1998b) uses the junction model of Förstner and Gülch (1987), determines the differentiation scale by analyzing the local sharpness of the image and chooses the integration scale by optimizing the precision of junction localization. The recent work of Förstner et al. (2009) proposes a scale space formulation that directly exploits the structure tensor and the general spiral feature model to detect scale-invariant features (SFOP). It includes junctions as a special case and generalizes the point detector in Förstner (1994). The authors have shown that the scale and rotation invariant features have repeatability comparable to that of LOWE.

Extensions from scale to affine invariant detectors usually also rely on the structure tensor, by undistorting a local image patch such that the structure tensor becomes isotropic. This implicitly yields elliptical image windows.

2.1.3 Edge and Line Features

Edge or line features are usually obtained by first computing the response of a pixel-wise detector, e.g. using the structure tensor, followed by a grouping stage to obtain connected straight or curved segments. In object recognition, such one-dimensional features were historically considered less often due to the lack of a robust wide-baseline matching technique. However, some promising approaches have been proposed. Bay et al. (2005) choose weak intensity-based matching using two quantized color-histograms on each side of an oriented straight line segment, followed by a sophisticated topological filter and boosting stage for both eliminating outliers and iteratively collecting previously discarded inliers. Meltzer and Soatto (2008) proposed a technique for computing robust descriptors for scale-invariant edge segments (Lindeberg 1998a) with possibly curved shape. They achieve impressive results for shape and object recognition.

We refer to other literature for an overview of edge and line detection methods, e.g. Heath et al. (1996). The focus of this initial study on the completeness of detectors is on zero- and two-dimensional features. However, we will include a straight edge segment detector (EDGE) based on the framework by Förstner (1994) in our investigations.

2.1.4 Performance Evaluation for Feature Detectors

As a generally accepted criterion, the repeatability of extracted patches under changes of viewpoint and illumination has been investigated in mostly planar scenes (Mikolajczyk et al. 2005) and on 3D structured objects (Moreels and Perona 2007). Recent work has also focussed on localization accuracy (Haja et al. 2008; Zeisl et al. 2009) and the general impact on automatic bundle adjustment (Dickscheid and Förstner 2009). Benchmarks usually evaluate each method separately, not addressing the positive effect when using a combination of multiple detectors, which may be very useful in many applications (Dickscheid and Förstner 2009). For

example, Bay et al. (2005) propose to use edge segments together with regions for calibrating images of man-made environments with poor texture. Their approach benefits from stable topological relations between features of complementary detectors.

Completeness of feature detection received less attention up to now. Perďoch et al. (2007) use the notion of "coverage" for stating that a good distribution of features over the image domain is favorable over a cluster of features in a narrow region. This concept is strongly related to completeness. In the interesting work of Lillholm et al. (2003), the ability of edge- and blob-like features to carry image information is investigated based on a model of retinal cells and considering the task of image reconstruction. The authors show that features based on the second derivatives of the image, namely blobs, carry most image information, while fine details are coded in the first order derivatives, namely edges. They also report that increasing the number of edge features effectively improves the image details, while increasing the number of blobs is less rewarding, and emphasize the complementarity of these two feature types.

Neither Perďoch et al. (2007) nor Lillholm et al. (2003) formalize their notions of coverage or complementarity. To our best knowledge a tool for measuring complementarity of different detectors is still lacking.

2.2 Information Contained in an Image

Describing the information content of an image can be approached from at least two sides: by analyzing the visual system, following a biologically inspired approach, or by investigating the image statistics, following an information theoretic approach (Shannon 1948). An excellent review on these paradigms, together with a mathematical formalization, is given by Mumford (2005).

The biologically inspired approach of (Marr 1982) identified relevant image information with the so-called *primal sketch*, which mainly refers to the blobs, edges, ends and zero-crossings that can be found in an image. Such a representation in principle is achieved using a combined set of different feature detectors. Marr's approach has been supported by the recent work of Olshausen and Field (1996). Extracting image features for subsequent processing in image analysis actually can be seen as mimicking the visual system. Higher level structures, resulting from grouping processes, are essential for interpretation and give rise to models like unsupervised clustering for segmentation (Puzicha et al. 1999) or grammars (Liang et al. 2010).

Following Mumford (2005), at least two properties characterize the local image statistics: (1) The histograms of intensities or gradients are heavy tailed, (2) the power spectrum $P_g(u)$ of an image $g(x)$ falls off with a power of the frequency u. For large frequencies one observes

$$P_g(u) \propto \frac{1}{u^b} \tag{2}$$

with exponents b in the range 1 to 3, smaller exponents representing rougher images.

The entropy of a signal depends on the probability distribution of the events, which is usually not known and hence has to be estimated from data. In spite of the empirical findings, approximations are frequently used to advantage. Restricting to second order moments, thus variances and covariances, one implicitly assumes the signal to be a Gaussian process, neglecting the long tails of real distributions. As an example, Bercher and Vignat (2000) proposed a general method for estimating the entropy of a signal by modeling the unknown distribution as an autoregression process, focusing on finding a good approximation with tractable numerical integration properties. The widely accepted JPEG image coding scheme is also based on entropy encoding. It uses the Discrete Cosine Transformation (DCT) and implicitly assumes that second order statistics are sufficient. Other coding schemes, e.g. based on wavelets, follow a similar argument (Davis and Nosratinia 1998). The "salient region" detector by Kadir and Brady (2001) derives the entropy from the histogram of an image patch, neglecting the interrelations between intensities within an image patch.

In the context of this paper we are only concerned with local image statistics, not taking higher level context into account.

3 Completeness of Coding with Image Features

3.1 Completeness vs. Coverage

A simple approach for evaluating the completeness of a feature set would measure the amount of image area covered by the local patches, as applied in Perďoch et al. (2007) or Dickscheid and Förstner (2009). Such approaches have a number of shortcomings, as illustrated in Fig. 2.

1. Full completeness can be achieved using a few features with large scales as in Fig. 2(a), or a regular grid of adjacent features. However, such feature sets do not code the image structures well. We would like a good measure not to overrate such feature sets.
2. Superimposed features on different scales, as in Fig. 2(b), would not contribute to the completeness, although often carrying important additional information. We would like a proper measure to take such effects into account.
3. Fine capturing of local structures, as in Fig. 2(c), is not rewarded. As a consequence, adding complementary features may not converge towards full completeness, while

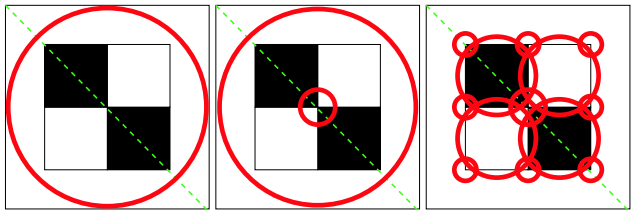


Fig. 2 Illustration of three simple feature sets (*circles*) covering an image of a checkerboard. (**a**) Single feature on a large scale, (**b**) two superimposed features with different scales, (**c**) mixed set of junction and blob features. Obviously (**c**) captures the image structure most completely. *The dashed line* indicates the stripe used in Fig. 3

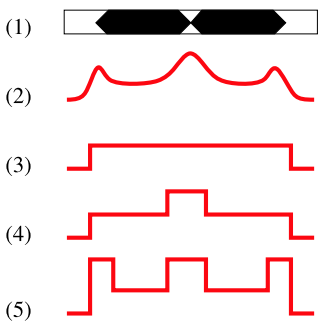


Fig. 3 From top: (**1**) Image stripe across the main diagonal line of the checkerboard in Fig. 2, (**2**) profile of the intuitive amount of relevant image information along the stripe, (**3–5**) coverage profiles along the stripe for the feature sets in Fig. 2 (a–c). The profiles take overlap into account

adding random features often would. A proper measure should show the opposite behavior.

Obviously we need a better measure. Consider the image signal in a diagonal stripe across the checkerboard image depicted in Fig. 3(1). The intuitive amount of relevant image information contained in small local patches along the stripe is depicted by Fig. 3(2): It is zero on the image border, has peaks at the checkerboard corners, especially in the center of the checkerboard, and is also high within the checkerboard patches. Fig. 3(3–5) show coverage profiles along the image diagonal, obtained by taking the overlap of the features in Fig. 2(a–c) into account. The similarity between these coverage profiles and the profiles in the second row would obviously give a better measure: The additional feature in Fig. 3(4) is rewarded w.r.t. Fig. 3(3), and the reasonable coverage of complementary features in Fig. 3(5) is most similar to the reference profile.

The measure that we derive in the following is modelled after this principle.

3.2 Basic Principle

As motivated in the introduction, we want to interpret feature detection and description as coding of image content, so we propose to measure completeness of local feature extraction as a comparison with classical coding schemes. We will apply three quantities for deriving the measure (Fig. 5):

1. *Representation of local image content*. We need a reasonable representation for the parts considered to reflect "relevant information" within an image, corresponding to Fig. 3(2).
2. *Representation of local feature content*. We need a meaningful representation of the content that is coded by a particular set of local features, roughly related to Fig. 3(3–5).
3. *Distance measure*. Having two such representations, we need a proper distance measure. If the distance vanishes, the features are supposed to capture the image content completely, so it should reflect the incompleteness of the local image information preserved by the features.

For representing local image content, we use the entropy density $p_H(\mathbf{x})$. It is based on the local image statistics and gives us the distribution of the number of bits needed to represent the complete image over the image domain. Thus, if we need $H^{(I)}$ [bits] for coding the complete image, we can use p_H for computing the fraction of $H^{(I)}$ needed for coding an image region $\mathcal{R}$ by $H_{\mathcal{R}} = H^{(I)} \int_{x \in \mathcal{R}} p_H(\mathbf{x}) \mathrm{d}\mathbf{x}$. The number of bits will be high in busy areas, and low in homogeneous areas of the image, as depicted in the right column of Fig. 6. For taking into account features of different scales at the same image position, we compute the local entropies over scales, as will be explained in Sect. 3.5.

For the local feature content, a feature coding density $p_c(\mathbf{x})$ is directly computed from each particular set of local features. Each region covered by a feature is represented with an anisotropic Gaussian distribution spreading over the image domain. We assume that a certain number of bits is needed to represent each feature, i.e. the area covered by each Gaussian. Within this paper, the number of bits remains constant for all features, but one may also derive it explicitly for each feature. The coding density $p_c(\mathbf{x})$ is finally obtained by the normalized sum of these Gaussians. We have illustrated such coding densities for different feature sets in the center columns of Fig. 6.

Figure 4 shows the profiles of some feature coding and entropy densities along the diagonal of a noisy image of a 4×4 checkerboard with grey background. These plots can be compared to the introductory illustration in Fig. 3, visually indicating that the proposed representations have the desired properties.

The incompleteness d is determined as the distance between $p_H(\mathbf{x})$ and $p_c(\mathbf{x})$, using the Hellinger metric. When p_c is close to p_H, thus d being small, the image is efficiently covered with features, and the completeness is high. We hereby require busy parts of images to be densely covered with features, and smooth parts not to be covered with features, as motivated in the Sect. 3.1.

We now describe the individual quantities used in our approach in more detail.

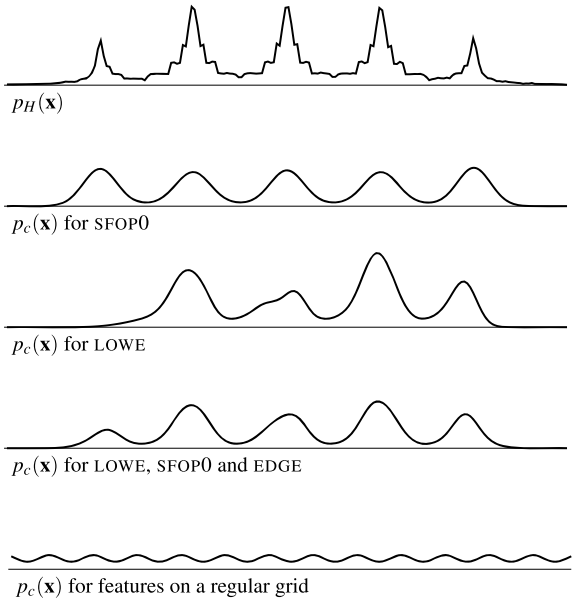


Fig. 4 Profiles of the entropy density p_H and different feature coding densities p_c along the diagonal stripe of a noisy image of a checkerboard, now with 4×4 patches and a grey border (top row, cf. Fig. 10). The profiles correspond to the diagonal slices of the distributions. The combination of LOWE, SFOP0 and EDGE is most similar to the entropy $p_H(\mathbf{x})$. The regular grid of features is not very similar, but may have higher completeness than a very sparse detector. That is why we recommend to use the evaluation scheme on feature sets of comparable sparseness

3.3 Representing Image Content by Local Entropy

The total number of bits required for coding an image with a certain fidelity can be derived using rate distortion theory, which has already been discussed in the classical paper of Shannon (1948) and worked out by Berger (1971). The idea is to determine a lower bound for the number of bits that are necessary for coding a signal, given a certain error tolerance. We will use this theory for determining how the required bits are distributed over the image, and derive a pixel related measure from the local image statistics of a square patch.

3.3.1 Model for the Signal of an Image Patch

We assume the signal $g(\mathbf{x})$ in an $J \times J = K$ image patch to be the noisy version of a true signal $f(\mathbf{x})$. The true signal and the noise are assumed to be mutually independent stochastic variables. While $f(\mathbf{x})$, vectorized to $\mathbf{f}$, has arbitrary mean $\boldsymbol{\mu}_f$ and covariance matrix Σ_{ff}, we assume the noise $\underline{\boldsymbol{n}}$ to have zero mean and diagonal covariance $\Sigma_{nn} = N_0 I_K$. The model hence reads

$$\underline{\boldsymbol{g}} = \underline{\boldsymbol{f}} + \underline{\boldsymbol{n}} \tag{3}$$

with

$$\underline{\boldsymbol{f}} \sim \mathcal{N}(\mathbf{0}, \Sigma_{ff}), \qquad \underline{\boldsymbol{n}} \sim \mathcal{N}(\mathbf{0}, N_0 I_K), \qquad \Sigma_{nf} = 0. \tag{4}$$

The vanishing covariance Σ_{nf} between signal and noise reflects the assumed mutual independence.

We want to derive the minimum number of bits needed to code the local area at each pixel, based on the statistics of the surrounding image patch, with the requirement that the loss of information is below a certain error tolerance. The tolerance is expressed by the mean square discrepancy $D_0 = E[(\underline{\boldsymbol{f}} - \underline{\widehat{\boldsymbol{f}}})^2]$, where the given signal $\mathbf{f}$ is approximated by a reconstruction $\widehat{\mathbf{f}}$.

We first derive the number of bits required for lossy coding a single Gaussian variable and then generalize to Gaussian vectors with correlations. In order to obtain a reasonable estimate without knowledge of the correlations, we finally assume the image patch to be the representative part of a doubly periodic infinite signal, and will derive the number of bits from the power spectrum.

Using a Gaussian model for the image patch is an approximation that has been used by other authors before. For example, Lillholm et al. (2003) use it to motivate different norms for regularization during image reconstruction, assuming either uncorrelated pixels or white noise for the gradients. At first the choice is pragmatic, as the model allows us to exploit the power spectrum, which considers only first and second moments. Due to the lack of an alternative model, we cannot verify the influence of this approximation on our results theoretically. However, in our experiments we will investigate in how far the model leads to meaningful results, and claim that it is reasonable for the given problem.

3.3.2 Entropy of a Lossy Coded Single Gaussian

Assume that we want to code a signal y which is a sample of a Gaussian with a given mean and variance V_y. Its differential entropy is (Bishop 2006, eq. 1.110)

$$H(y) = \frac{1}{2}\log_2(2\pi \mathrm{e} V_y) = \frac{1}{2}\log_2(2\pi \mathrm{e}) + \frac{1}{2}\log_2 V_y. \tag{5}$$

We consider y to be the signal that has originally been emitted, and want the reconstructed signal $\widehat{y}$ to lie within an error tolerance $E[(\underline{y} - \underline{\widehat{y}})^2] = D_0$ after transmission. Under the further assumption that the transmission process is ergodic, i.e. that a very long sample of the transmission will be typical for the process with probability near 1, rate distortion theory gives the following result (Davisson 1972):

- We only need to transmit the signal if $V_y \geq D_0$. If V_y is smaller, D_0 dominates, hence we have

$$D = \min(D_0, V_y). \tag{6}$$

- The minimum number of bits R_y—the rate—required for the transmission is given by the mutual entropy of y and

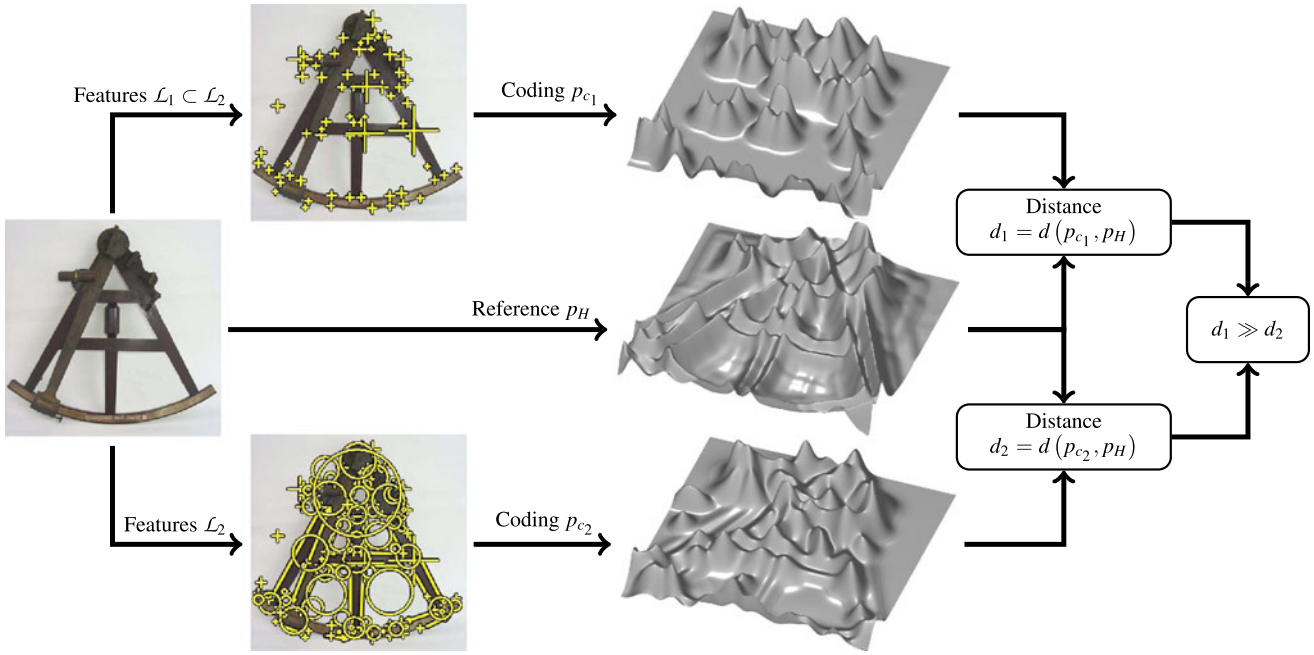


Fig. 5 Workflow for a single image of the evaluation scheme described in Sect. 4.1.1, using only two different sets of features. $\mathcal{L}_2$ is a combined set of junctions, edges and blobs, while $\mathcal{L}_1$ is its subset of junctions. Having $|\mathcal{L}_1| \ll |\mathcal{L}_2|$, we expect $d_1 \gg d_2$. Indeed the density p_{c_2} looks much more similar to the entropy distribution p_H than p_{c_1}. If $d_1 \gg d_2$ would not hold, we would conclude that the blobs and edges $\{\mathcal{L}_2 \setminus \mathcal{L}_1\}$ do not complement the junctions $\mathcal{L}_1$ well

$\widehat{y}$, sometimes also denoted as mutual information:

$$R_y \doteq H(y) - H(y|\widehat{y}) \tag{7}$$

$$= \frac{1}{2}\log_2(2\pi e V_y) - \frac{1}{2}\log_2(2\pi e D_0) \tag{8}$$

$$= \frac{1}{2}\log_2 \frac{V_y}{D_0}. \tag{9}$$

Here we use the assumption that the noise and the true signal are mutually independent, hence that their entropies add. Therefore the conditional entropy—the second part of (8)—is simply the entropy of the noise, i.e. the number of bits needed for coding the distortion.

Combining (9) and (6), we have the required number of bits for a lossy coded single Gaussian (Davisson 1972, eq. (11))

$$R_y = \frac{1}{2}\max\left(0, \log_2 \frac{V_y}{D_0}\right). \tag{10}$$

We use this approximation for coding, assuming that the power spectrum of the local patch, from which we determine R_y, is representative for its statistical properties.

3.3.3 Entropy of a Lossy Coded Gaussian Vector

We now want to generalize (10) to the case of a stochastic Gaussian K-vector $\underline{\boldsymbol{y}} \sim N(\boldsymbol{\mu}_y, \Sigma_{yy})$ with correlations between its components. Using the factorization of the covariance matrix Σ_{yy} into eigenvectors and eigenvalues

$$\Sigma_{yy} = \sum_{k=1}^{K} \lambda_k^2 \mathbf{r}_k \mathbf{r}_k^{\mathsf{T}}, \tag{11}$$

we may use the alternative representation

$$\underline{\boldsymbol{y}} = \boldsymbol{\mu}_y + \sum_{k=1}^{K} \mathbf{r}_k \underline{z}_k \quad \text{with } \underline{z}_k \sim N(0, \lambda_k^2). \tag{12}$$

This representation allows us to effectively replace the original components of $\underline{\boldsymbol{y}}$ by K stochastically independent Gaussian variables $\underline{z}_k$. We therefore can determine the total entropy for lossy coding of the stochastic vector $\underline{\boldsymbol{y}}$ as the sum of the entropies for the individual mutually independent $\underline{z}_k$ (Davisson 1972, eq. (23))

$$R_y = \frac{1}{2}\sum_{k=1}^{K} \max\left(0, \log_2 \frac{\lambda_k^2}{D_0}\right). \tag{13}$$

This expression depends essentially on the eigenvalues λ_k^2 of the covariance matrix Σ_{yy} of the signal to be coded.

3.3.4 Entropy of a Local Image Patch

The covariance matrix of a local image patch is not known, hence we do not have the eigenvalues for applying (13) di-

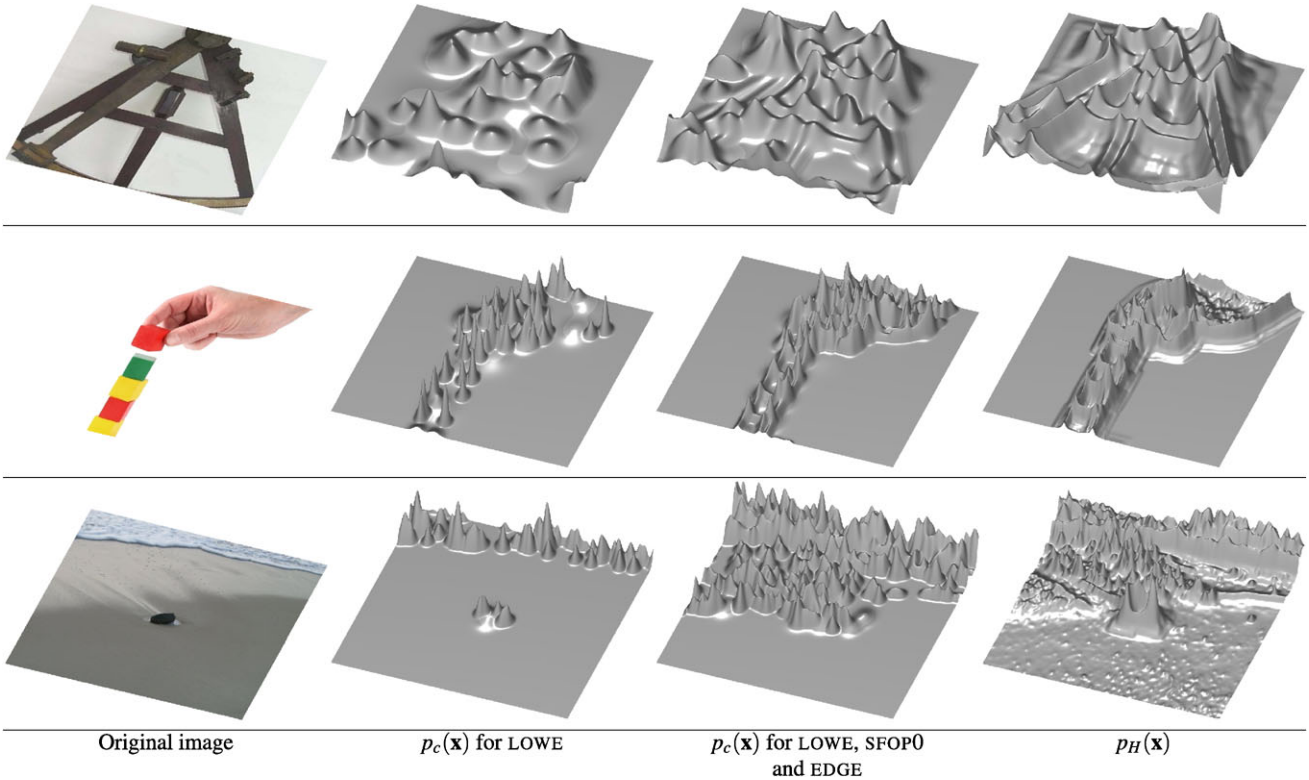


Fig. 6 Estimated entropy densities $p_H(\mathbf{x})$ and feature coding densities $p_c(\mathbf{x})$ on three different images. We illustrate feature coding densities for LOWE separately, and for a combination of LOWE, EDGE and SFOP0. Combining complementary features (*third column*) seems to converge towards the entropy (*right column*). Observe that the *white area* in the image of *the middle row* is below the noise level, thus not considered by the entropy, while the sand in the bottom row carries a small amount of information

rectly onto an image patch. Therefore we assume the image patch to be the central part of an infinite periodic signal, and use the power spectrum for deriving the eigenvalues of the covariance matrix of the intensities in the image patch.

Let $F = [F_{jk}] = [\exp(-2\pi ijk/K)]$ be the $K \times K$-Fourier matrix. Then the discrete Fourier transform of the cyclical signal $y(k)$ with K-vector $\mathbf{y}$ is $\mathbf{Y} = F\mathbf{y}$ and the power spectrum of $y(k)$ is

$$P_y(u) = \frac{1}{K}|Y(u)|^2. \tag{14}$$

One can show that the eigenvalues of the covariance matrix Σ_{yy} of $\mathbf{y}$ are identical to the elements of the power spectrum $P(u)$ of the periodic signal $y(k)$: We use the unitary matrix

$$\overline{F} = \frac{1}{\sqrt{K}}F, \qquad \overline{F}\overline{F}^* = \overline{F}^*\overline{F} = I_K, \tag{15}$$

where $\overline{F}^*$ is the transposed complex conjugate of $\overline{F}$. It is equivalent to a rotation matrix. The empirical covariance matrix of $y(k)$ then is $\Sigma_{yy} = \frac{1}{K}F\,\mathrm{Diag}(P_y(u))F^*$. Finally this can be written as

$$\Sigma_{yy} = \overline{F}\,\mathrm{Diag}(P_y(u))\overline{F}^*, \tag{16}$$

showing the entries of the power spectrum to be the eigenvalues of the covariance matrix.

If the patch is a doubly periodic two-dimensional signal $f(\mathbf{x})$ of size $K = J \times J$, the power spectrum $P_f(\mathbf{u}) = P_f(u_1, u_2)$ depends on the row and column frequencies u_1 and u_2. Thus the squared eigenvalues λ_k^2 of Σ_{ff} are $\lambda_k^2 = P_f(\mathbf{u})$, the index k capturing the indices $\mathbf{u} = [u_k]$. Given the power spectrum, we obtain the entropy of an $K = J \times J$ patch $f(\mathbf{x})$ by (Davisson 1972, discussion after eq. (36))

$$R_f = \frac{1}{2}\sum_{\mathbf{u}\setminus(0,0)} \max\left(0, \log_2 \frac{P_f(\mathbf{u})}{D_0}\right). \tag{17}$$

Note that we omit the DC term $P_f(\mathbf{0})$, as it represents the mean of the signal and we are only concerned with the variance.

Finally, as the true signal is not available, but only the observed signal as in (3), we need to estimate the power spectrum P_f. From the structure of the covariance matrix $\Sigma_{gg} = \Sigma_{ff} + \Sigma_{nn}$ we know that $P_g(\mathbf{u}) = P_f(\mathbf{u}) + P_n(\mathbf{u})$. Therefore we choose to estimate the power spectrum of the true signal by

$$\widehat{P}_f(\mathbf{u}) = \max(0, P_g(\mathbf{u}) - N_0) \tag{18}$$

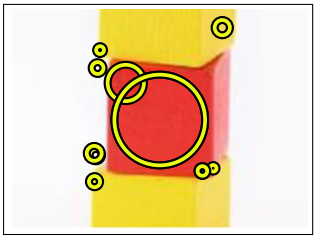


Fig. 7 Detail of the example image depicted in Fig. 1 left. Note that the regions are plotted with radius equal to the scale of the feature. The support region used for computing descriptors is usually larger but proportional

and obtain the final entropy per patch as the rate

$$\widehat{R}_f = \frac{1}{2} \sum_{\mathbf{u} \setminus (0,0)} \max\left(0, \log_2 \frac{\max(0,\, P_g(\mathbf{u}) - N_0)}{D_0}\right). \tag{19}$$

We estimate the standard deviation of the noise N_0 for each input image separately using the approach of Förstner (1998). As a minimum, we use the rounding error, identified by the standard round-off error variance $D_0 = N_0 = \epsilon^2/12$ (Smith 2007). The quantization unit ϵ depends on the actual image representation, e.g. $\epsilon = 1/256$ for 8-bit intensities.

As the estimated number of bits spreads over the local patch however, we assign to the pixel only the corresponding fraction according to the patch size, i.e.

$$\widehat{R}_f^{(p)}(\mathbf{x}, M) = \frac{1}{2K} \sum_{\mathbf{u} \setminus (0,0)} \max\left(0, \log_2 \frac{\widehat{P}_f(\mathbf{u})}{D_0}\right). \tag{20}$$

In the end we use the Discrete Cosine Transform Type-II instead of the Discrete Fourier Transform. It suppresses effects of signal jumps at the borders of the window onto the spectrum, while preserving the frequencies except for a factor of 2 (Rosenfeld and Kak 1982, p. 159). The equivalence of the power spectrum entries with the eigenvalues of the covariance matrix therefore still holds.

We cannot proceed with a fixed patch size J, as can be seen from the image detail shown in Fig. 7: The large feature in the middle covers content with rich information, although a smaller patch placed in its center for computing the entropy might have minimal entropy.

Like in wavelet transform coding, we assume that the coding is hierarchical for the entropy density, and choose to integrate information from different scales in order to take such scale effects into account. For doing so, we compute the sum

$$H(\mathbf{x}) = \sum_{s=1}^{S} \widehat{R}_f^{(p)}(\mathbf{x}, 1 + 2^s). \tag{21}$$

In our experiments we use $S = 7$, so the patch size is limited to $3 \leq M \leq 129$. Observe that the omission of the DC term in a lower scale is compensated by the coding in higher scales. Taking the sum in (21) is a pragmatic choice. We did not yet investigate to which extent this is an approximation regarding possible correlations between scale levels. As will be seen in the next section however, we handle superimposed features at different scales in a very similar manner when computing the feature coding densities.

Finally we obtain the entropy density by normalizing (21):

$$p_H(\mathbf{x}) = \frac{H(\mathbf{x})}{\sum_z H(\mathbf{z})}. \tag{22}$$

The expected number of bits in a certain region $\mathcal{R}$ therefore is $H^{(I)} \sum_{x \in \mathcal{R}} p_H(\mathbf{x})$, where $H^{(I)}$ is the total number of bits for the complete image. However, for comparing the entropy distribution over the image with a particular set of local features, we cannot use absolute values (i.e. bits per pixel), but only relative values. This is why we take $p_H(\mathbf{x})$ as reference. In the right column of Fig. 1, this entropy density is depicted for two example images.

3.4 Representing Local Feature Content

The local feature content is a density comparable to $p_H(\mathbf{x})$, representing a set of L local features $\mathcal{L} = \{l_1, \ldots, l_L\}$. Each local feature represents a certain image region with known shape and is represented as a tuple $l_l = \{\mathbf{x}_l, \Sigma_l, \mathbf{d}_l, c_l\}$, where

- $\mathbf{x}_l$ is the spatial location of the feature in the image,
- Σ_l is a scale matrix representing the shape of the feature window, using the model of a 2D ellipse,
- $\mathbf{d}_l$ is a vector used as a description of the feature window, and
- c_l is the number of bits required for $\mathbf{d}_l$, coding the whole region $\mathcal{R}_l$. The number of bits may be different between feature types, but we usually assume a constant number of bits for all features.

Positioning Σ_l at $\mathbf{x}_l$ gives us the actual region $\mathcal{R}_l$ covered by the feature. For straight edge segments, identified by their start- and endpoints $(\mathbf{x}_S, \mathbf{x}_E)_l$, we use $\mathbf{x}_l = (\mathbf{x}_{S,l} + \mathbf{x}_{E,l})/2$ and use the direction of the edge as the major semi-axis of Σ_l with length $|\mathbf{x}_{E,l} - \mathbf{x}_{S,l}|/2$ while setting the length of the minor semi-axis to 1 [pel].

If we spread the bits c_l uniformly over each region $\mathcal{R}_l$, we obtain a feature coding map by

$$c(\mathbf{x}) = \sum_{l=1}^{L} \frac{\mathbf{1}_{\mathcal{R}_l}(\mathbf{x})}{|\Sigma_l|} c_l \tag{23}$$

where $\mathbf{1}_{\mathcal{R}_l}(\mathbf{x})$ is an indicator function, being 1 within the region $\mathcal{R}_l$, and 0 outside. In order to emphasize the information next to a feature's center, it is common practice to apply Gaussian weighting within the local patches before computing feature descriptors (Lowe 2004, 6.1). So it is reasonable

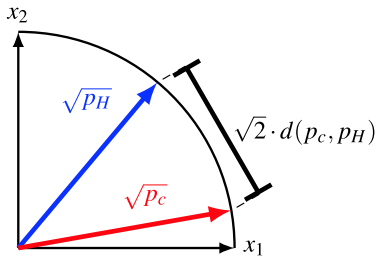


Fig. 8 Hellinger's metric d illustrated for two sample distributions p_H and p_c of an image with two pixels x_1, x_2

to replace the uniform distribution by a Gaussian and derive the feature coding map by

$$c(\mathbf{x}) = \sum_{l=1}^{L} c_l G(\mathbf{x}; \mathbf{x}_l, \Sigma_l). \tag{24}$$

Then the actual density to be compared with the entropy density $p_H(\mathbf{x})$ is

$$p_c(\mathbf{x}) = \frac{c(\mathbf{x})}{\sum_{\mathbf{z}} c(\mathbf{z})}. \tag{25}$$

We illustrate feature coding densities of some example images in Fig. 6.

3.5 Evaluating Completeness of Feature Detection

In case the empirical feature coding density $p_c(\mathbf{x})$ would be identical to the entropy density $p_H(\mathbf{x})$, coding the image with features would be equivalent to using image compression. Of course image features may use less or more bits for coding the complete image, depending on the coding of the individual feature, so we do not compare the absolute number of bits per pixel, but their densities. We use Hellinger's metric for measuring the difference between $p_H(\mathbf{x})$ and $p_c(\mathbf{x})$:

$$d(p_H(\mathbf{x}), p_c(\mathbf{x})) = \sqrt{\frac{1}{2}\sum_{\mathbf{x}} (\sqrt{p_H(\mathbf{x})} - \sqrt{p_c(\mathbf{x})})^2}. \tag{26}$$

Hellinger's metric is computed from the pixel-wise differences of the square roots of the densities, and may be explained as illustrated in Fig. 8: Each of the square-rooted densities can be thought of as a unit vector in the space of all possible densities over the image, the dimension of this space being equal to the number of pixels. Then Hellinger's distance between the two densities is proportional to the Euclidean distance between the piercing points on the unit sphere, so it basically relates to the angle between the vectors representing each density.

Note that although one often uses the Kullback-Leibler divergence for comparing density functions, it is not useful here: It is no metric, and a larger divergence does not necessarily indicate a larger deviation from the reference density.

3.6 Euclidean Embedding of Detectors

In Sect. 3.5 we derived as an incompleteness measure the distance d between the entropy p_H distributed over an image and the information p_c coded with a set of local features. In addition to comparing image codings with entropy, we can also measure distances between pairwise feature codings p_{c_i} and p_{c_j}. This allows us to build up a distance network in high dimensional space representing the similarity or complementarity of feature detectors.

From such a network of N detectors we can determine their relative positions in a $(N-1)$-dimensional space using a technique called "Euclidean embedding" (Cayton and Dasgupta 2006). First the matrix of squared distances $D: [D_{ij}] = d(p_{c_i}, p_{c_j})^2$ is centered using the centering matrix $H = I - \frac{1}{D}\mathbf{1}\mathbf{1}^\mathsf{T}$, with $\mathbf{1}$ being a vector filled with ones:

$$B = -\frac{1}{2} HDH. \tag{27}$$

Computing the spectral decomposition of $B = U\Lambda U^\mathsf{T}$, and ensuring positive eigenvalues $[\Lambda_+]_{ij} = \max(\Lambda_{ij}, 0)$, we obtain relative positions $X = [x_n]$ for each feature detector:

$$X = \Lambda_+^{1/2} U^\mathsf{T}. \tag{28}$$

This way we end up with a mapping for feature detectors, taking one image into account only. By evaluating the mean of the distances d between each two codings p_{c_i} and p_{c_j}, we can average the positions X over multiple images, since they are supposed to be rather invariant w.r.t. image content. In order to propagate the variance of the distances d into the detector space, we perform the embedding for each image individually. Afterwards we transform the detector positions X into one common reference frame, taking changes in orientation and directions of the eigenvectors into account.

The $(N-1)$-dimensional detector space can be visualized by neglecting small principal components. Figure 9 illustrates the mapping of some detectors into 3D space. The ellipsoids represent an approximation of their distribution. Some well-known relationships between the detectors become visually apparent in the plot, as for example between IBR and MSER.

3.7 A Feature Detector Based on Local Entropy

So far we discussed two strategies for selecting the characteristic scale of a feature: The Laplacian, or Difference of Gaussian space; and the structure tensor. Both methods have been summarized in Sect. 2. The entropy density p_H that we developed in the previous section gives rise to a third paradigm, namely using the local entropy for building the scale space.

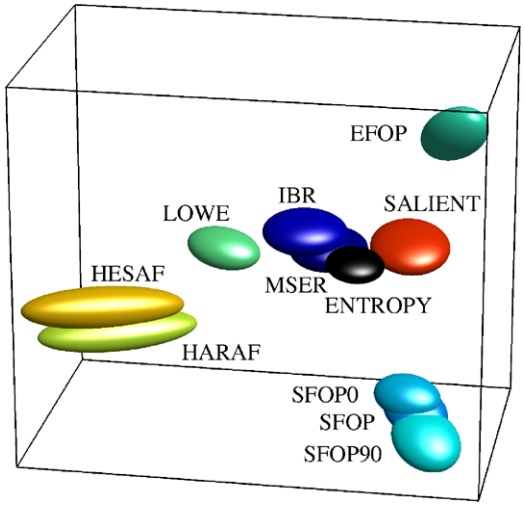


Fig. 9 Visualization of the Euclidean embedding (Cayton and Dasgupta 2006) of detectors: Several feature detectors are mapped into a 3D subspace spanned by the three largest principal components of the centred distance matrix B. The ellipsoids approximate the distribution of the detectors over all considered images. Notice how the close theoretical relationship between the SFOP variants, MSER/IBR and HESAF/HARAF becomes visible in terms of proximity

To our knowledge, Kadir and Brady (2001) were the first to present a local feature detector based on information theory (SALIENT). They estimate the entropy of local image attributes over a range of scales and extract points at peaks with significant magnitude change of the local probability density. A set of such features is shown on the right side of Fig. 10. The number of features extracted by SALIENT is high compared to other detectors (see Table 1), resulting in good image coverage. However, by estimating entropy from local histograms, the approach ignores the unknown pixel correlations, which usually affect local image attributes. It has rather low repeatability compared to other detectors (Mikolajczyk et al. 2005).

In this section, we want to derive a scale-invariant feature detector which models the local entropy in the same way as when deriving the entropy density p_H proposed in Sect. 3.3. It extracts sparse sets of features near peaks of the entropy distribution p_H. One possible way to model the detector would be to build a stack of entropy distributions for varying patch sizes using (20), and then search for local extrema within $3 \times 3 \times 3$ dices of the resulting cuboid. However, computing the power spectrum over all scales would result in high computational complexity. Therefore we will take another approach: We express the entropy density using an approximate functional model for the power spectrum, depending only on the variances of the intensities, the image gradients and the image noise. These can easily be determined. The detector is included in our evaluation (Sect. 4.2), denoted by EFOP.

3.7.1 The Local Entropy for Rough Signals

We represent the power spectrum, which is a decaying function, by the model

$$P(u) = P_0 \left(1 + \frac{u}{u_0}\right)^{-1} \tag{29}$$

taking the extreme exponent $b = 1$ in (2) and regularizing for small frequencies. This can be used to replace the power spectrum in (17), and allows us to approximate the local entropy as a function of the signal to noise ratio SNRand the effective bandwidth b_g of the signal in the local image content:

$$R_f = k \cdot b_g \sqrt{\log_2(1 + \mathrm{SNR}^2)} \tag{30}$$

with $k \simeq 5.43$. Here we use the relationships for the signal-to-noise ratio

$$\mathrm{SNR}^2 = \frac{P_0}{N_0} = \frac{V_g}{V_n} \tag{31}$$

and the effective bandwidth (McGillem and Svedlow 1976, Eq. (16))

$$b_g^2 = \frac{V_{g'}}{V_g} = 4\pi^2 \frac{\int_{u=0}^{u_n} u^2 P(u)\mathrm{d}u}{\int_{u=0}^{u_n} P(u)\mathrm{d}u}, \tag{32}$$

to rewrite (30) as

$$R_f \propto \sqrt{\frac{V_{g'}}{V_g} \log_2\left(1 + \frac{V_g}{V_n}\right)}, \tag{33}$$

which only depends on the variance of the image gradients $V_{g'}$, the variance of the gray-scale values V_g and the noise variance V_n.

The derivation is outlined in Appendix. Note that this model is only proportional to the approximate entropy (40), as we intend to perform a maximum search over the function.

3.7.2 Finding Maxima of Local Entropy over Scale

Following Shi and Tomasi (1994), we identify the variance of the gradients $V_{g'}$ with the smaller eigenvalue of the structure tensor $\lambda_2(M_{\tau,\sigma})$ instead of using its trace (see Sect. 2.1.2). This avoids detecting points located on lines with poor localization accuracy. The integration scale σ then spans the space for selecting features at different scales, yielding the following scale space of the entropy rate:

$$R_f^2(\mathbf{x}, \tau, \sigma) = \frac{\lambda_2(M(\mathbf{x}, \tau, \sigma))}{V_g(\mathbf{x}, \tau, \sigma)} \log_2\left(1 + \frac{V_g(\mathbf{x}, \tau, \sigma)}{V_n(\mathbf{x}, \tau)}\right). \tag{34}$$

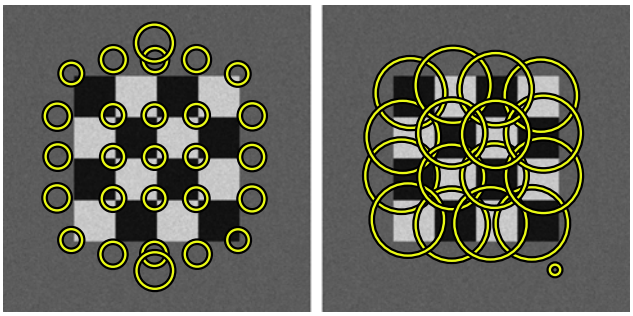


Fig. 10 A noisy image of a checkerboard, overlaid with features detected by the maximum-entropy detector EFOP described in Sect. 3.7 (*left*) and the one proposed by Kadir et al. (2004) (*right*). Note that we use σ as a radius for the features, while one would usually compute descriptors on a 2σ patch

The close relationship of the structure tensor with the local information in an image has already been observed by McClure (1980, eq. (7)).

The variance of the gray-scale values can be computed via

$$V_g(\sigma,\tau) = G_\sigma * (G_\tau * g - G_{\sqrt{\sigma^2+\tau^2}} * g)^2, \tag{35}$$

and the noise variance by $V_n(\tau) = V_n/(8\pi\tau^4)$. Therefore, (34) meets our goal of approximating the local entropy by a formulation based on convolutions of the gray-scale values and gradients only.

The final detection algorithm proceeds as follows:

1. Compute the entropy of each pixel for a range of scales σ using (34), fixing $\tau = \sigma/3$.
2. Search for local maxima over $3 \times 3 \times 3$ dices of the resulting 3D cuboid, yielding a set of keypoint candidates.
3. Keep only those keypoints where $\lambda_2(\mathbf{M})$ is significant.
4. Perform a non-maximum-suppression and subpixel localization.

The procedure is almost identical to the algorithm described in Förstner et al. (2009, Chap. 3) which contains more implementation details, except that the scale-space representation (34) is used.

The character of the detected features is illustrated by the result on a checkerboard image in Fig. 10. The detector finds the junctions with gradients in all four directions quite exactly. Junctions on the border, with homogeneous areas towards one side, yield extrema on slightly larger scales with a strong but stable offset from the border. They are practically located over homogeneous areas similar to blobs. We do not claim the detector to have high repeatability or precision, its use for object detection is outside the scope of this paper. However, we used it successfully for camera calibration, as reported in Dickscheid and Förstner (2009).

4 Experiments

In the following we will apply the above-mentioned evaluation to a data set involving various image categories, and analyze the complementarity and completeness of both separate detectors and detector combinations. After describing the experimental setup in Sect. 4.1, we will present comprehensive results, and show in Sect. 4.2 how they coincide with the concepts of the detectors.

4.1 Experimental Setup

4.1.1 Evaluation Scheme

We use the distance measure d derived in Sect. 3 (26) for evaluating sets of local features. The procedure is illustrated for two sets $\mathcal{L}_1, \mathcal{L}_2$ in Fig. 5. We start by computing feature coding densities p_{c_i} for each set $\mathcal{L}_i$ as well as the entropy distribution p_H of the image as a reference. Next we compute the distances $d_i = d[p_H(\mathbf{x}), p_{c_i}(\mathbf{x})]$, and compare them. We typically draw conclusions of these types:

1. If $d_i < d_j$, feature set $\mathcal{L}_i$ is in general more complete w.r.t. p_H referring to an image. It is then often interesting to compare the amount of features: If $|\mathcal{L}_i| \simeq |\mathcal{L}_j|$, the improved completeness is an obvious advantage. As mentioned before, we recommend comparing feature sets of comparable sparseness.
2. If $\mathcal{L}_i \subset \mathcal{L}_j$ and $|\mathcal{L}_i| \ll |\mathcal{L}_j|$, we conversely expect $d_i \gg d_j$, as in Fig. 5. In other words: We expect a combined set of features $\mathcal{L}_j$ to be more complete than any significantly smaller subset $\mathcal{L}_i$ of it, otherwise we conclude that $\{\mathcal{L}_j \setminus \mathcal{L}_i\}$ does not complement $\mathcal{L}_i$ well.

4.1.2 Applied Feature Detectors

Blob detectors We investigate most of the prominent scale and affine invariant blob detectors presented in Mikolajczyk et al. (2005), using the implementations from the corresponding website. These include

- the Harris and Hessian Laplace detectors (HARLAP, HESLAP) of Mikolajczyk and Schmid (2004) as well as their affine covariant extensions (HARAF, HESAF),
- the Maximally Stable Extremal Regions detector (MSER) by Matas et al. (2004),
- the intensity-based region detector (IBR) by Tuytelaars and Van Gool (2004),
- the Edge-Laplace detector (EDGELAP) by Mikolajczyk et al. (2003), and
- the salient regions detector (SALIENT) of Kadir and Brady (2001).

Furthermore we include the popular Laplacian blob detector by Lowe (2004) using the original source code kindly

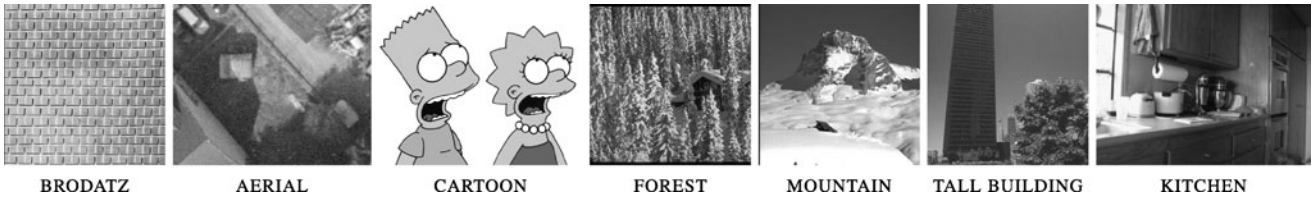


Fig. 11 Example images from the datasets used for the experiments

provided by the author. Here we choose to build the scale space starting with the original instead of the double image resolution, for comparability with the other detectors.

Detectors based on the structure tensor For point-like features based on the structure tensor, we use the recently proposed scale invariant detector by Förstner et al. (2009), but distinguish the results into three classes:

- The subset of pure junction points (SFOP0) obtained when restricting to $\alpha = 0$ (Förstner et al. 2009, eq. (7)),
- the set of pure circular features (SFOP90) with $\alpha = 90$, and
- the complementary set of features with optimal α, i.e. the full power of the detector (SFOP).

Additionally we compare that to the non-scale-invariant detectors described in Förstner (1994) for junctions and circular symmetric features (FOP0, FOP90), as well as the classical detector HARRIS by Harris and Stephens (1988) with an implementation taken from Kovesi (2009). The maximum-entropy-detector (EFOP) described in Sect. 3.7 is not specifically a point detector. It represents the class of window or texture detectors exploiting the structure tensor.

Line and edge detectors Here we restrict to a straight edge detector (EDGE) based on the theory in Förstner (1994), using a minimum edge length of ten pixels. We intentionally do not include a whole range of line segment detectors in order to keep the number of detector combinations tractable within this paper. We also do not expect significantly different results using other edge detectors.

Parameter settings Completeness is in general lower for sparse feature sets. Decreasing a detector's significance level will yield more features but at the same time reduce repeatability and performance. We have chosen to use default parameter settings provided by the authors whenever available in order to make our results a direct complement to existing evaluations. Unfortunately this involves different amounts of features being extracted, which we report in Table 1 for clarity.

4.1.3 Image Data

We compare the completeness and mutual complementarity of the above mentioned detectors on a variety of images. Our experiments are built on the fifteen natural scene category dataset (Lazebnik et al. 2006; Li and Perona 2005), complemented by the well-known Brodatz texture dataset, a collection of cartoon images from the web and a set of image sections of an aerial image. We present results for a subset of seven categories, as depicted in Fig. 11. For each of the categories, we report average results over all images.

4.1.4 Investigated Sets of Features

We start by computing the incompleteness $d(p_c, p_H)$ for each of the detectors separately over all images of a category. This will give us a first impression on the completeness of the different methods. The results are discussed in Sect. 4.2.1. Our special interest is the effect of combining sets of features, i.e. for finding evidence about their complementarity. However, we obviously cannot discuss results for arbitrary tuples of all of these detectors. We therefore choose to constrain the set of all possible combinations in two ways:

1. In Sect. 4.2.2 we start by identifying detectors with very similar properties and close theoretical relationships, and use such sets as groups. Detectors within a group are not combined mutually to decrease the overall number of combinations.
2. We concentrate on combinations of three detectors. We hereby assume that using more than three detectors is not feasible in most applications. Results for triplets are given in Sect. 4.2.3.

4.2 Experimental Results

4.2.1 Results for Separate Detectors

Figure 12 shows the average of the distances $d(p_c, p_H)$ on seven image categories for each detector separately. The corresponding average amounts of features are listed in Table 1.

The most noticeable result is that the EDGELAP detector shows significantly highest completeness on all image categories with rather good confidence. This is certainly due to the very high number of features compared to other detectors (Table 1).

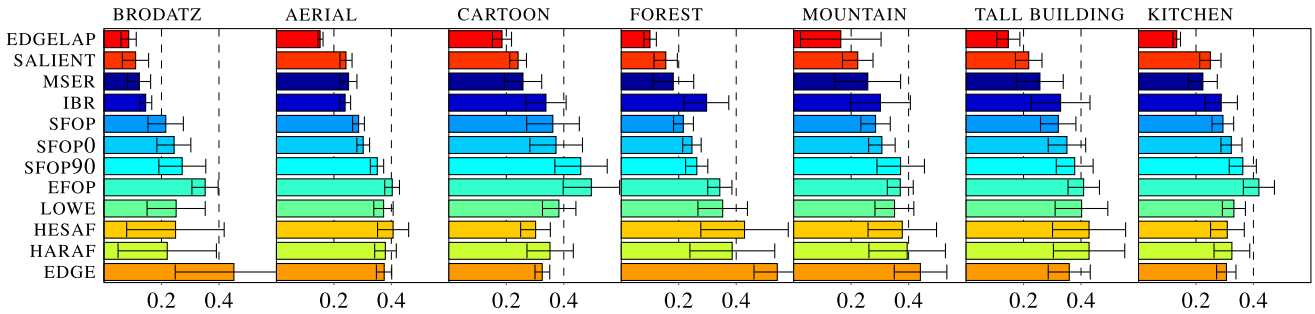


Fig. 12 Average dissimilarity $d(p_c, p_H)$ for separate detectors over all images of seven image categories. The categories for BRODATZ, AERIAL and CARTOON contain between 15 and 30 images, all others contain around one hundred images. Small distances denote high completeness of a detector w.r.t. the entropy distribution p_H. The additional black bars denote the 1-σ-confidence region over all images in a category

Table 1 Average number of features per image category extracted by the different detectors. We see that SALIENT and especially EDGELAP extract more features than most other detectors, while IBR and EFOP are very sparse

	BRODATZ	AERIAL	CARTOON	FOREST	MOUNTAIN	TALL BUILDING	KITCHEN
EDGELAP	28011	4658	5518	4678	2548	3252	3210
SALIENT	5832	613	846	607	296	385	200
MSER	2202	212	187	186	82	122	96
IBR	1078	182	122	40	29	31	31
SFOP	2052	494	238	290	157	152	145
SFOP0	1355	344	175	197	110	102	95
SFOP90	1177	295	120	183	91	95	87
EFOP	473	153	72	66	57	54	42
LOWE	2093	324	307	152	105	111	115
HESAF	3499	338	837	156	134	139	195
HARAF	4847	359	596	239	138	156	166

The MSER and SALIENT detectors show overall best results aside from EDGELAP. While in case of SALIENT we have to put the higher number of features into perspective again, the result for MSER is really remarkable: It has significantly higher completeness compared to most other detectors, but at the same time similar sparseness. SFOP and IBR both have slightly lower completeness than the above mentioned. The good score of IBR is especially noticeable, as the feature sets are very sparse. For EFOP, having similar sparseness, the completeness is rather poor instead. The LOWE, HARAF and HESAF detectors, all in principle exploiting the Laplacian scale space, achieved similar results among each other. They rank slightly behind IBR and SFOP on average.

As expected, SFOP performs better than its special cases SFOP0 and SFOP90. Among these two, the junctions are more complete than the circular points, which is also intuitive. Especially on AERIAL and FOREST, the completeness of detectors based on the structure tensor seems to be better than that of Laplacian-based methods. This observation is interesting, as the structure tensor has been shown to be related to high local information (McClure 1980). The behavior of the straight edge features (EDGE) depends on the amount of man-made structures. While it is acceptable on KITCHEN, TALL BUILDING, CARTOON and AERIAL, it shows overall worst results for the natural structures in FOREST, MOUNTAIN and BRODATZ. Conversely, the completeness for the blob and corner detectors rather benefits from natural structures, the behavior hence being almost opposite over categories compared to EDGE.

4.2.2 *Relationships Between Detectors*

The goal of this section is to reduce the set of detectors used for the final evaluation of triple combinations, both by selectively excluding some detectors and by grouping mostly redundant methods. To do so, we will use the Euclidean embedding of different sets of detectors as explained in Sect. 3.6 and identify clusters. We will finally end up with the six groups of detectors depicted in Table 2.

Role of the EDGELAP *detector* The EDGELAP detector produces significantly more points than all other detectors, as it does not explicitly suppress responses on edges. This paradigm is superior regarding the proposed completeness measure, but targets at special applications. When combined

Table 2 Groups of detectors used for evaluating possible triplets, referring to the conclusions made in Sect. 4.2.2

Group	Members
Edges	EDGE
Segmentation-based	IBR, MSER
Laplacian	LOWE
Mixed Laplacian	HESAF, HARAF
Structure tensor	EFOP
Spiral type	SFOP, SFOP0, SFOP90

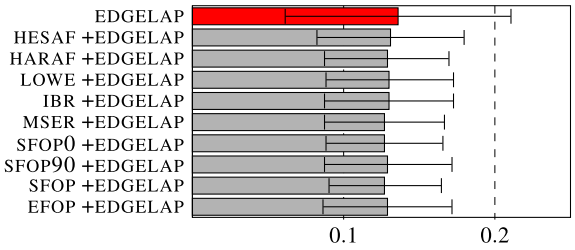


Fig. 13 Average incompleteness $d(p_c, p_H)$ for pairwise combinations of other detectors complementing EDGELAP over all images in our evaluation. Black bars denote the 1-σ-confidence region. We see that EDGELAP is not significantly complemented by any of the detectors, as it has strong completeness on its own already

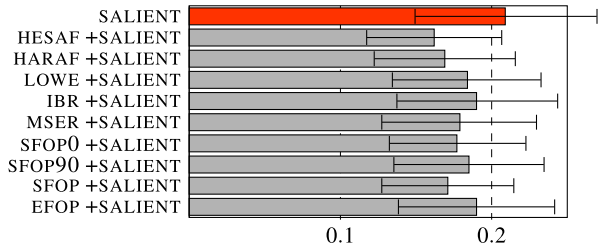


Fig. 14 Average incompleteness $d(p_c, p_H)$ for pairwise combinations of other detectors complementing SALIENT over all images in our evaluation. Black bars denote the 1-σ-confidence region. We see that SALIENT is most efficiently complemented by HESAF, followed by HARAF and SFOP

with other detectors, the increase in complementarity becomes hardly visible, as shown in Fig. 13. Therefore we choose not to include EDGELAP in the final evaluation of detector combinations.

Role of the SALIENT *detector* SALIENT detects less features than EDGELAP but still significantly more than the other detectors. Despite its high completeness scores, it is still complemented by some other detectors (Fig. 14), especially by HESAF. In our experiments we found that triple combinations including SALIENT achieve best scores among all combinations, apart from those including EDGELAP. As mentioned before however, the computational complexity of the detector is orders of magnitude higher than that of other

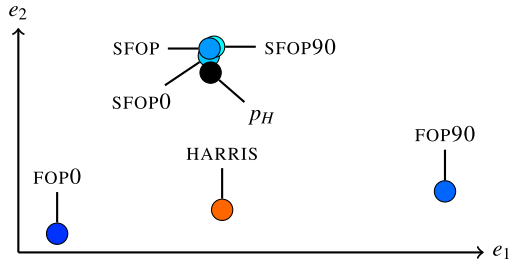


Fig. 15 Projection of the Euclidean embedding (Sect. 3.6) of SFOP, SFOP0, SFOP90 together with their classical non-scale-invariant counterparts FOP0, FOP90 and HARRIS onto its first two principal components. The nodes represent average values over all images of all categories. The scale-invariant feature sets are by far closer to the reference p_H, thus significantly more complete

detectors, and its repeatability ranges significantly below average (Mikolajczyk et al. 2005). It is therefore rather unlikely that SALIENT will be used in conjunction with other detectors, so we choose not to include it in our evaluation of combinations.

Scale invariance of detectors for corners and circular features We analyzed the classical detectors for corners and circular features FOP0, FOP90 and HARRIS together with the scale-invariant extensions SFOP, SFOP0, SFOP90 over different image categories. Throughout the experiments we found that the scale-invariant methods are significantly more complete than the classical ones, which is expected due to the variable patch sizes. The projection of a mapping over all image categories is shown in Fig. 15. We therefore choose to only include the scale invariant features in the final evaluation.

Affine invariance of blob detectors We compared HARLAP and HESLAP with their affine covariant versions HARAF and HESAF, which differ by a final affine refinement of the window shapes. The differences between the affine and non affine versions are negligible referring to our completeness measure, as illustrated in Fig. 16. Therefore we only include the affine-extended versions in the subsequent evaluation.

Groups of similar detectors We want to find out in how far different detectors are mutually complementary referring to the proposed measure. It is therefore not necessary to consider combinations of highly redundant detectors, where complementarity cannot be expected. We illustrate such redundant groups by a projection of a high-dimensional mapping over all image categories in Fig. 17. The results of the IBR and MSER detectors are strongly related throughout the image categories. Therefore we put them into a group denoted "Segmentation-based". This is in agreement with the observation already made by Tuytelaars and Mikolajczyk (2008, p. 211) that IBR and MSER yield very similar regions.

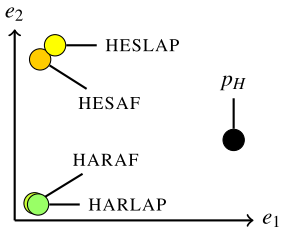


Fig. 16 Projection of Euclidean embedding (Sect. 3.6) of HARAF, HESAF, HARLAP and HESLAP onto its first two principal components. The affine invariant feature sets are not significantly different from the non-affine invariant ones w.r.t. the completeness measure

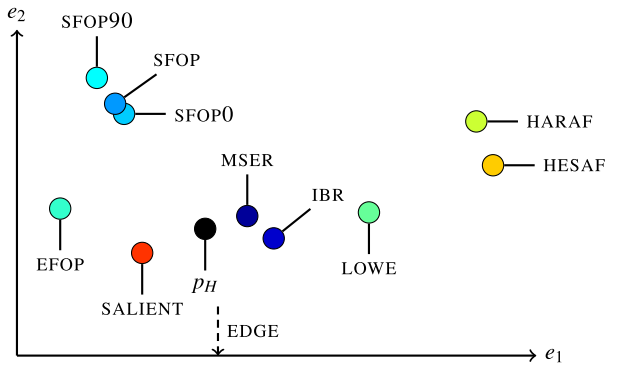


Fig. 17 Projection of Euclidean embedding (Sect. 3.6) of the most important detectors onto its first two principal components. Regarding the proposed completeness measure, the detectors roughly build three groups: MSER and IBR form a cluster, as well as the spiral model based feature sets SFOP, SFOP0 and SFOP90, and the HESAF and HARAF detectors. The other detectors are rather distinguished and not put into groups. Note that the EDGE detector has significant larger distance and is not included in this figure

The SFOP, SFOP0 and SFOP90 detectors are also closely related, which is intuitive as they all result from the same theoretical framework (Bigün 1990). Therefore we put these three into a group called "Spiral type". The LOWE detector, though theoretically similar to HESAF and also HARAF, shows significant distance towards these two in all our experiments. Therefore it builds a separate group "Laplacian". The EFOP detector developed in Sect. 3.7 is also distinguished from the others in most image categories. As it basically fires at texturedness, classified by the structure tensor, it represents the group "Structure tensor". The HESAF and HARAF detectors show mostly similar results and have small mutual distance over all image categories. This observation contradicts the statement of Tuytelaars and Mikolajczyk (2008, 4.3) that HESAF and HESLAP are "complementary to their Harris-based counterparts, in the sense that they respond to a different type of feature in the image." Although making use of the Laplacian scale space, the detectors differ from LOWE in that the scale search is induced by local extrema of multi-scale representations differing from the pure Laplacian. Therefore they are grouped as "Mixed Laplacian", above the Laplacian exploiting the structure tensor and the Hessian respectively.

4.2.3 *Results for Selected Triplets of Detectors*

We computed the incompleteness $d(p_H, p_c)$ of all possible triplets over the groups given in Table 2, without combining detectors within the same group. This yields 82 overall combinations. The average incompleteness scores for these combinations over all image categories are listed in Table 3.

The most complete triplets contain a combination of the groups "segmentation based" and "spiral type". These groups seem to be complementary over a wide range of image categories. As a third component, EDGE, LOWE or EFOP are most promising.

Besides this, two "holes" in the table attract the attention: The Laplacian-based blobs do not appear among the first ten to fifteen entries, and the last rows of the table are empty for the spiral type features. This indicates that spiral-type features play an important role for promising complementary sets. Regarding the Laplacian-based detectors, it suggests some redundancy with one of the other groups, which can be verified when observing the last rows of the table: Here we find mainly combinations of the segmentation based with the Laplacian detectors. It therefore seems that combining MSER or IBR with one of the Laplacian based detectors is rather redundant, but that IBR/MSER have higher complementarity to the remaining groups.

There are many other combinations that also work well: Combining HARAF or HESAF as a blob detector with junctions, edges or segmentation based regions usually achieves good results.

We motivated our approach by the fact that highly redundant feature sets are punished. Considering combinations including LOWE and HESAF, which are theoretically most closely related, strengthens this statement. They appear prevalently in the last rows of the table, but not even once within the first half.

It is important to note that the differences between combinations are substantially smoothed due to the large amount of images used. For separate categories, one obtains more significant differences.

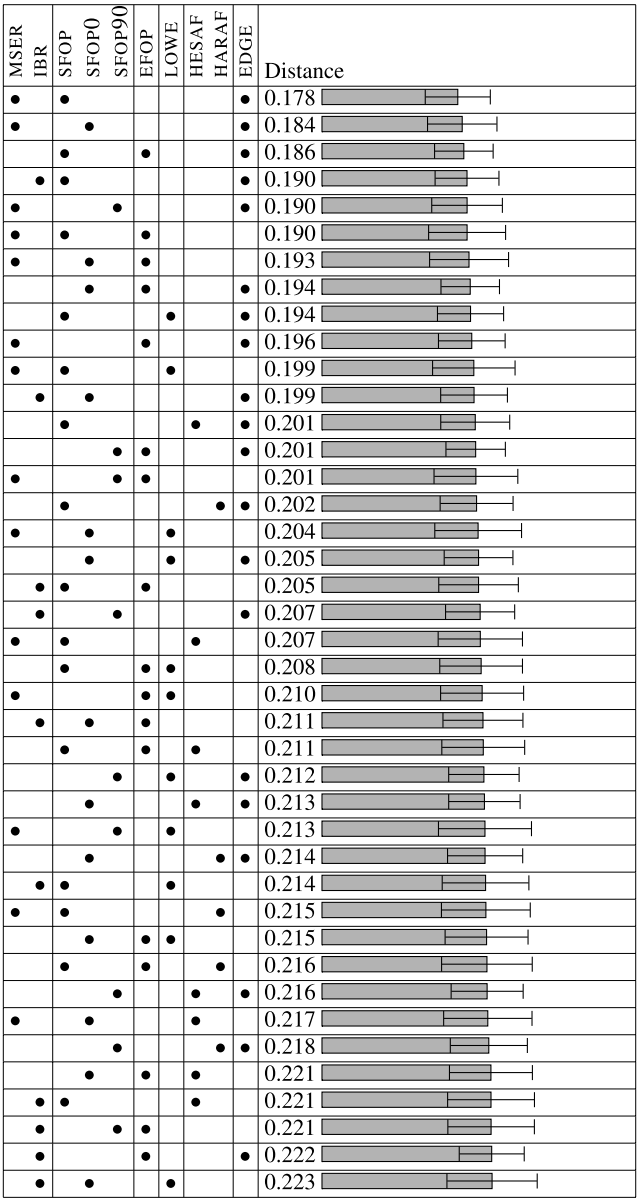
4.2.4 *Other Combinations of Detectors*

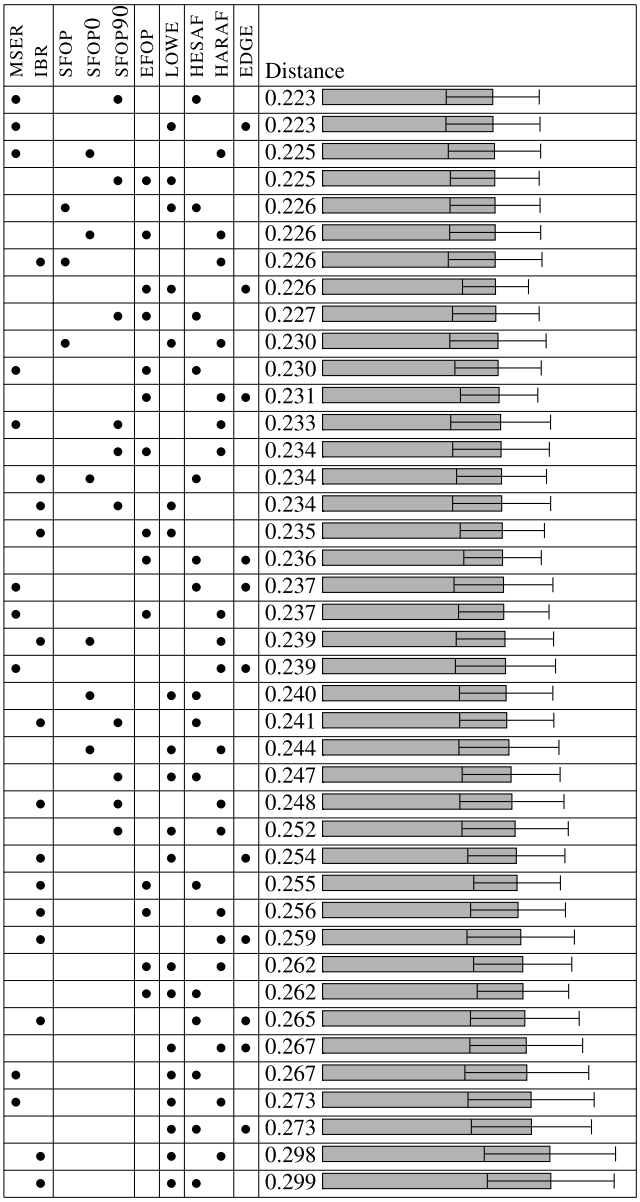
Combinations containing detectors within the same group (Sect. 4.2.2) show worse results, confirming our selection principle. As an example, a combination of three blob detectors, i.e. LOWE, HESAF and HARAF, usually achieves significantly lower completeness scores. A similar effect occurs when using three spiral-type detectors.

In addition to triple combinations, we also computed the results for combinations of four and more detectors, respecting the grouping shown in Table 2. We found that the best group of four or five detectors is usually only slightly better than the best group of three.

A complete list of our results is available at http://www.ipb.uni-bonn.de/completeness.

Table 3 Average incompleteness $d(p_H, p_{c_i})$ for feature sets $\mathcal{L}_i$, arising from all considered triplets of detectors, over all image categories. The additional black bars denote the 1-σ-confidence region over all images in a category. Black column borders for detectors indicate groups (Table 2). The triplets are sorted in ascending order w.r.t. their completeness regarding the entropy density p_H





4.3 Applicability of the Results in Real Applications

In the end we are interested in the impact of these results onto some real application scenarios. The two most important applications are certainly camera calibration and object recognition.

The impact of a variety of detector combinations onto camera calibration has been recently studied by Dickscheid and Förstner (2009). There, a fixed strategy for automatic image orientation has been used with different detector combinations as an input. Then results of a final bundle adjustment were compared. The most meaningful measure in such a setting is the accuracy of the estimated camera projection centers compared to ground truth (Dickscheid and Förstner 2009, Fig. 7), which we want to relate to our findings about completeness.

For separate detectors, SFOP and MSER achieved best results in the bundle adjustment. This is in full agreement with the incompleteness in Fig. 12, considering that EDGELAP, SALIENT and IBR have not been included in the bundle adjustment test. The LOWE detector performed only slightly worse, which can also be verified from our completeness scores. Combinations of detectors usually had positive effects onto the bundle adjustment. The best combination of

three was LOWE with SFOP and MSER. This combination also achieves one of the best completeness scores, when ignoring combinations including EDGE. Especially interesting is the fact that combining LOWE and HESAF, which are closely related and highly redundant, had negative effects onto the bundle adjustment. In Table 3, we also see that combinations including these two appear only towards the end, in the second part of the table.

The completeness therefore seems to be a good indicator for promising feature sets w.r.t. camera calibration, besides robustness and localization accuracy. A comparison with results in an object recognition framework still has to be done.

5 Conclusion and Outlook

We have proposed a scheme for measuring completeness of local features in the sense of image coding. To achieve this, we derived suitable estimates for the distribution of relevant information and the coverage by a set of local features over the image domain, which we compared by the Hellinger distance. This enables us to quantify the completeness of different sets of local features over images, and especially in how far detectors are complementary in this sense. The approach has important advantages over a simple comparison of image coverage. For example, it favors response on structured image parts while penalizing features in purely homogeneous areas, and it accounts for features appearing at the same location on different scales.

The proposed scheme does not give a general benchmark for detectors. To make a good choice for a particular application, detectors have to be preselected according to task usefulness, considering sparseness, robustness, and speed. Then the proposed evaluation helps in finding the most complete ones, and especially the most promising complementary combinations.

We made a number of interesting observations. Scale invariance is clearly beneficial for covering image content, while at the same time we could not observe improvements when using affine covariant windows. The EDGELAP detector (Mikolajczyk et al. 2003), basically using edge information, showed clearly superior completeness as a separate detector, but targets at specific applications. Also the SALIENT detector (Kadir and Brady 2001) achieved very good results, extracting a significantly higher number of features than other detectors. The MSER detector (Matas et al. 2004) seems to be most complete among the detectors with average sparseness. For combining features, it is most profitable to use a detector implementing a junction model (like SFOP, Förstner et al. (2009)) together with a good blob or region detector, ideally MSER (Matas et al. 2004). Best triplet combinations are achieved when additionally using a texture-related detector, such as an edge or entropy-based detector.

The proposed entropy density $p_H(\mathbf{x})$ gives rise to a new scale invariant keypoint detector which locally maximizes the entropy over position and scale. We have proposed such a detector (EFOP), which, compared to the detector of Kadir and Brady (2001), yields significantly sparser feature sets and takes pixel correlations into account. The completeness of the new detector however is below average.

Evaluating feature detectors in general suffers from the somewhat arbitrary and implementation dependent parameter settings. We believe that in principle the parameters should be clearly determined from the image, i.e. via the estimated noise variance (Förstner 1998; Liu et al. 2008). It would be highly desirable for future benchmarks to have a common, reduced set of input parameters, which in our opinion should be no more than the possibly signal dependent noise variance and maximum number of features. Each detector could then select the best ones according to its own formulation of significance. This way the performance of detectors could be characterized as a function of the number of features. By evaluating the minimum number of necessary features to reach a prespecified performance in a given application, a characterization in terms of efficiency would also be possible.

Of course the results have to be transferred to practical applications. By relating them to a recent evaluation in the context of camera calibration, we found that the accuracy of bundle adjustment results was strongly correlated with completeness. A comparison with results in an object recognition framework still has to be done.

We did not consider edge detectors in detail due to their rather special role. It would be interesting to include a number of edge detectors, especially scale invariant ones, into the setup. An investigation into individual coding schemes for different feature types—points, lines, and blobs—would be desirable. Finally a more realistic image model than the Gaussian could lead to a more detailed insight into the relationship between different detectors.

Acknowledgements We wish to thank the anonymous reviewers for valuable comments leading to an improvement of the manuscript.

Appendix: Derivation of the Effective Bandwidth in (30)

This section outlines the derivation of the approximation (30) based on the model (29). Based on (17), a continuous formulation of the entropy of a local patch using the fractal model is

$$R_f = \frac{1}{2}\int_{u_0}^{u_n} \log_2 \frac{P_0(1+\frac{u}{u_0})^{-1}}{D_0} \mathrm{d}u. \tag{36}$$

For asserting that we do not use any bits where the signal is below distortion, we have to restrict the integration limits and require $u \leq u_n$ satisfying

$$\frac{P_0}{1+\frac{u_n}{u_0}} = N_0 \quad \Longleftrightarrow \quad u_n = u_0\left(\frac{P_0}{N_0} - 1\right). \tag{37}$$

In order to find a suitable value for the model parameter u_0, we need further knowledge about the signal. We can solve the integrals in the right hand side of (32) to relate the effective bandwidth b_g^2 to u_0 and the signal-to-noise ratio by

$$b_g^2 = \frac{2\log_2 \mathrm{SNR}^2 + 3 - 4\mathrm{SNR}^2 + \mathrm{SNR}^4}{4\log_2 \mathrm{SNR}} u_0^2. \tag{38}$$

Here we again assume additive white noise with known variance, as in the model introduced in (3) and (4). Assuming the bandwidth and signal-to-noise ratio of the signal to be known, we can use (38) for replacing u_0 in (36), with the integration limit as specified above, and obtain the entropy

$$R_f = 2(\mathrm{SNR}^2 - 1)(\log_2 2\pi\mathrm{e} + 1 - 2\log_2 \mathrm{SNR}) \times \sqrt{\frac{\log_2 \mathrm{SNR}}{2\log_2 \mathrm{SNR}^2 + 3 - 4\mathrm{SNR}^2 + \mathrm{SNR}^4}}\, b_g. \tag{39}$$

This function can be approximated by

$$H_a(g) = k\, b_g \sqrt{\log_2(1+\mathrm{SNR}^2)} \tag{40}$$

with

$$k = \sqrt{2}(1 + \ln(2\pi\mathrm{e})) \approx 5.43. \tag{41}$$

Using the correct scale factor, one can show by analytic comparison that the approximation error between (30) and (17) is below 1%.

References

Bay, H., Ferrari, V., & Gool, L. V. (2005). Wide-baseline stereo matching with line segments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Washington, DC, USA (Vol. 1, pp. 329–336).

Bercher, J. F., & Vignat, C. (2000). Estimating the entropy of a signal with applications. *IEEE Transactions on Signal Processing*, *48*(6), 1687–1694.

Berger, T. (1971). *Rate distortion theory: a mathematical basis for data compression*. Englewood Cliffs: Prentice-Hall.

Bigün, J. (1990). A structure feature for some image processing applications based on spiral functions. *Computer Vision, Graphics and Image Processing*, *51*(2), 166–194.

Bishop, C. M. (2006). *Information Science and Statistics*. *Pattern recognition and machine learning*. Berlin: Springer.

Cayton, L., & Dasgupta, S. (2006). Robust Euclidean embedding. In *Proceedings of the international conference on machine learning* (pp. 169–176). New York: ACM.

Corso, J. J., & Hager, G. D. (2009). Image description with features that summarize. *Computer Vision and Image Understanding*, *113*(4), 446–458.

Davis, G. M., & Nosratinia, A. (1998). Wavelet-based image coding: an overview. *Applied and Computational Control, Signals, and Circuits*, *1*, 205–269.

Davisson, L. D. (1972). Rate-distortion theory and application. *Proceedings of the IEEE*, *60*, 800–808.

Dickscheid, T., & Förstner, W. (2009). Evaluating the suitability of feature detectors for automatic image orientation systems. In *Proceedings of the international conference on computer vision systems*, Liege, Belgium (pp. 305–314).

Förstner, W. (1994). A framework for low level feature extraction. In *Proceedings of the European conference on computer vision*, Stockholm, Sweden (Vol. III, pp. 383–394).

Förstner, W. (1998). Image preprocessing for feature extraction in digital intensity, color and range images. In *Lecture notes in earth sciences: Vol. 95/2000. Geomatic methods for the analysis of data in earth sciences* (pp. 165–189). Berlin: Springer.

Förstner, W., & Gülch, E. (1987). A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *ISPRS conference on fast processing of photogrammetric data*, Interlaken (pp. 281–305).

Förstner, W., Dickscheid, T., & Schindler, F. (2009). Detecting interpretable and accurate scale-invariant keypoints. In *Proceedings of the IEEE international conference on computer vision*, Kyoto, Japan.

Haja, A., Jähne, B., & Abraham, S. (2008). Localization accuracy of region detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Harris, C., & Stephens, M. J. (1988). A combined corner and edge detector. In *Proceedings of the Alvey vision conference* (pp. 147–152).

Heath, M., Sarkar, S., Sanocki, T., & Bowyer, K. (1996). Comparison of edge detectors: a methodology and initial study. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (p. 143).

Kadir, T., & Brady, M. (2001). Saliency, scale and image description. *International Journal of Computer Vision*, *45*(2), 83–105.

Kadir, T., Zisserman, A., & Brady, M. (2004). An affine invariant salient region detector. In *Proceedings of the European conference on computer vision* (pp. 228–241). Berlin: Springer.

Kovesi, P. D. (2009). MATLAB and Octave functions for computer vision and image processing. School of Computer Science & Software Engineering, The University of Western Australia, http://www.csse.uwa.edu.au/~pk/research/matlabfns/.

Lazebnik, S., Schmid, C., & Ponce, J. (2006). Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Washington, DC, USA (pp. 2169–2178).

Li, F. F., & Perona, P. (2005). A Bayesian hierarchical model for learning natural scene categories. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (Vol. *2*, pp. 524–531). Los Alamitos: IEEE Comput. Soc.

Liang, P., Jordan, M. I., & Klein, D. (2010). Probabilistic grammars and hierarchical Dirichlet processes. In *The Oxford handbook of applied Bayesian analysis*. London: Oxford University Press.

Lillholm, M., Nielsen, M., & Griffin, L. D. (2003). Feature-based image analysis. *International Journal of Computer Vision*, *52*(2–3), 73–95.

Lindeberg, T. (1998a). Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, *30*(2), 117–156.

Lindeberg, T. (1998b). Feature detection with automatic scale selection. *International Journal of Computer Vision*, *30*(2), 79–116.

Liu, C., Szeliski, R., BingKang, S., Zitnick, C. L., & Freeman, W. T. (2008). Automatic estimation and removal of noise from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(2), 299–314.

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

Marr, D. (1982). *Vision: a computational investigation into the human representation and processing of visual information*. New York: Freeman.

Matas, J., Chum, O., Urban, M., & Pajdla, T. (2004). Robust wide baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, *22*, 761–767.

McClure, D. E. (1980). Image models in pattern theory. *Computer Graphics and Image Processing*, *12*(4), 309–325.

McGillem, C., & Svedlow, M. (1976). Image registration error variance as a measure of overlay quality. *IEEE Transactions on Geoscience Electronics*, *14*(1), 44–49.

Meltzer, J., & Soatto, S. (2008). Edge descriptors for robust wide-baseline correspondence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Mikolajczyk, K., & Schmid, C. (2004). Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, *60*(1), 63–86.

Mikolajczyk, K., Zisserman, A., & Schmid, C. (2003). Shape recognition with edge-based features. In *Proceedings of the British machine vision conference*, Norwich, UK (pp. 779–788).

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., & Gool, L. V. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, *65*(1/2), 43–72.

Moreels, P., & Perona, P. (2007). Evaluation of features detectors and descriptors based on 3D objects. *International Journal of Computer Vision*, *73*(3), 263–284.

Mumford, D. (2005). Empirical statistics and stochastic models for visual signals. In S. Haykin, J. Principe, T. Sejnowski, & J. McWhirter (Eds.), *New directions in statistical signal processing: from systems to brain*. Cambridge: MIT Press.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*(6583), 607–609.

Perďoch, M., Matas, J., & Obdržálek, Š. (2007). Stable affine frames on isophotes. In D. Metaxas, B. Vemuri, A. Shashua, & H. Shum (Eds.), *Proceedings of IEEE international conference on computer vision* (p. 8). Los Alamitos: IEEE Comput. Soc.

Puzicha, J., Hofmann, T., & Buhmann, J. M. (1999). Histogram clustering for unsupervised image segmentation. In *Proceedings of IEEE conference on computer vision and pattern recognition*, Ft. Collins, USA (pp. 602–608).

Rosenfeld, A., & Kak, A. C. (1982). *Digital picture processing*. San Diego: Academic Press.

Shannon, C. E. (1948). *A mathematical theory of communication* (Tech. Rep.). Bell System Technical Journal.

Shi, J., & Tomasi, C. (1994). Good features to track. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 593–600).

Smith, J. O. (2007). Mathematics of the discrete Fourier transform (DFT), W3K Publishing, Chap. 16.3.

Tuytelaars, T., & Mikolajczyk, K. (2008). *Local invariant feature detectors: a survey*. Hanover: Now Publishers.

Tuytelaars, T., & Van Gool, L. (2004). Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, *59*(1), 61–85.

Zeisl, B., Georgel, P., Schweiger, F., Steinbach, E., & Navab, N. (2009). Estimation of location uncertainty for scale invariant feature points. In *Proceedings of the British machine vision conference*, London, UK.