Learning Optimal Parameters for Self-diagnosis in a System for Automatic Exterior Orientation

Wolfgang Förstner, Thomas Läbe

Institute for Photogrammetry, University of Bonn
Nussallee 15, 53115 Bonn, Germany
wf|laebe@ipb.uni-bonn.de,
WWW home page: http://www.ipb.uni-bonn.de/staff/wf|laebe.html

Abstract. The paper describes the automatic learning of parameters for self-diagnosis of a system for automatic orientation of single aerial images used by the State Survey Department of Northrhine–Westfalia. The orientation is based on 3D lines as ground control features, and uses a sequence of probabilistic clustering, search and ML-estimation for robustly estimating the 6 parameters of the exterior orientation of an aerial image. The system is interpreted as a classifier, making an internal evaluation of its success. The classification is based on a number of parameters possibly relevant for self-diagnosis. A hand designed classifier reached 11 % false negatives and 2 % false positives on appr. 17 000 images. A first version of a new classifier using support vector machines is evaluated. Based on appr. 650 images the classifier reaches 2 % false negatives and 4 % false positives, indicating an increase in performance.

1 Motivation and Goal

Vision systems are nearly always embedded systems and are often the bottle neck of an automation chain. Their performance therefore is crucial for their usefulness within the embedding system. Independent of the output type of the vision system, e. g. reading bar codes, detecting passengers, tracking objects, or determining ego-motion, the vision system may be successful and produce one of these outputs, or it may fail and not deliver its output. Acceptable failure rates of systems may vary between 25 % e. g. in automatic image retrieval, and 0.0001 % in visual control of production processes. Many applications allow an efficient reaction to self-diagnosis of the vision system. Self-diagnosis is the internal evaluation of the system based on redundant information within the image analysis process. However, often such redundancy is intentionally introduced into the design of the vision system only to enable self-diagnosis.

An increase in efficiency of a system with self-diagnosis could be achieved by eliminating failures in case the system actually predicts a failure based on internal self-diagnosis. A system with self-diagnosis therefore can be interpreted as a classifier with respect to its ability to succeed or to fail, independent of the type of its output within the embedding system. The situation is sketched in fig. 1. A vision module in a first instance uses input data, preferably with measures on

their quality and - using a vision algorithm - produces output data, preferably with measures on their quality, which allows chaining of vision modules based on the exchange of evaluated data. Decisions to use certain modules need to be made subject to the success of the module, this would require active checking. Using an algorithm for self-diagnosis could produce values characterizing the internal prediction of success, again preferably with quality measures. Together with the quality of the output data, the self-diagnosis can be used by the control module. The distinction between the quality of the output data and the result of the self-diagnosis algorithm is motivated by the inability or reduced ability of the control module to interpret the details of the output characteristics of the vision module.

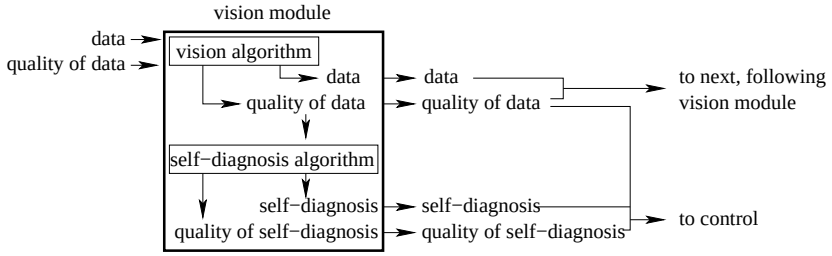

Fig. 1. Components of a vision module with self-diagnostic capabilities.

Characterizing the performance of a system therefore may also refer to characterizing the performance of its self-diagnostic abilities, on top of the performance of the output of the vision system as such.

We are concerned with the automatic determination of the parameters of the exterior orientation of single aerial images being the basis for ortho-photo maps. Being able to orient single images avoids the need for digitizing all images of a flight for bundle block adjustment, but enables to restrict digitization and storage to a single image (appr. 25 Mbyte) per map sheet (cf. fig. 2). In a country like Nordrhein-Westfalen, with appr. 8 000 map sheets in $1:5\,000$ this is a significant cost saving. The orientation of aerial images is supported by the global positioning system (GPS) giving sufficiently accurate projection centers, however allows no precise determination of the rotation parameters. As ortho-photo map update is done on a regular basis, the Survey Department already decided to build up a control point data base in the 70's, mainly consisting of buildings, especially the 3D coordinates of two of their roof points. Automating the process of ortho-photo production requires automation of the up to recently manual orientation procedure. The database therefore has been newly built up in the years 1992-1996, each control point consisting of a set of 3D line segments for each building, actually being derived from a manually measured wire-frame model of the buildings. The matching of the 3D line segments with 2D image line segments allows to determine the orientation parameters automatically (Sester & Förstner 1989). The system has recently been integrated and now is in use at the Survey Department.

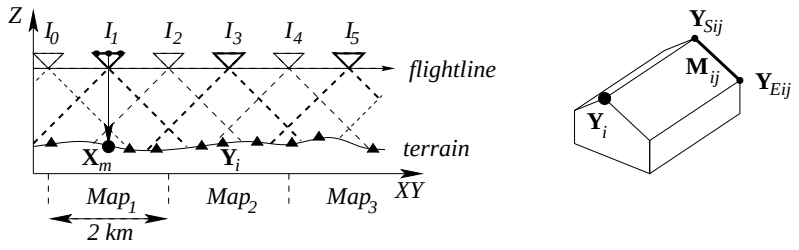

Fig. 2. Left: Setup of externally checking a procedure for automatic orientation of aerial images. Given: Images $I_0, ...,$ control points (reference coordinates $\mathbf{Y}_i$) (black filled triangle), terrain surface. *In practice* every second image is oriented and rectified. *Here* we orient all images individually by automatic mensuration of the control points. The same images and manually measured control points are used for a bundle adjustment. Evaluation is based on the difference of ground point coordinates $\mathbf{X}_m$ (black circle) determined with single image orientation and bundle orientation. Right: Structure of a control point. Set of 3D-line segments $\{\mathbf{M}_{ij}\}$ and reference point $\mathbf{Y}_i$.

This paper reports attempts to evaluate the self-diagnostic tools of the system and to improve them by automatic learning techniques, while not changing the automatic procedure for orientation. The paper is organized as follows: Section 2 describes the system, however, only to such a detail that the reader can locate and assess the self-diagnosis within the system. The manually designed self-diagnosis and its evaluation, based on 17 000 images is discussed in Section 3. Learning the relevant criteria for self-diagnosis is the topic of Section 4. Based on appr. 650 images it is shown that an improvement of the self-diagnosis can be achieved by using the result of a support vector machine classifier.

2 The System for Automatic Model Based Orientation

The core algorithm of the system for automatic model based orientation of aerial images consists of three robust techniques. A feature extraction procedure precedes these steps, the self-diagnosis finalizes the processing chain.

2.1 Preprocessing

Prediction of approximate image coordinates: Based on approximate values for the orientation of the aerial image the system selects those control points which are likely to appear in the image. The approximate projection matrix $\mathbf{P}$ is derived from the GPS-coordinates, the calibration parameters of the cameras and the knowledge, that the viewing direction is nearly towards nadir. Each control point P_i consists of a set $\{\mathbf{M}_{ij}\} = \{(\mathbf{Y}_S, \mathbf{Y}_E)_{ij}\}$ of 3D-line segments $\mathbf{M}_{ij}$ and has a reference point $\mathbf{Y}_i = (X, Y, Z, 1)_i^\mathsf{T}$ attached to it. It predicts the image coordinates $\mathbf{y}'_i = \mathbf{P}\mathbf{Y}_i$ of the reference point $\mathbf{Y}_i$ and of all starting and end points $\mathbf{y}'_{Sij}$ and $\mathbf{y}'_{Eij}$, leading to predicted 2D-line segments $\mathbf{m}'_{ik}$.

Line segment extraction: In a pre-defined window around the predicted reference point $\mathbf{y}'_i$, with 250^2 pixel covering appr. 0.3 % of the image area, a set $\{\mathbf{l}'_{ik}\} = \{(\mathbf{x}'_S, \mathbf{x}'_E)_{ik}\}$ of line segments are extracted. No correspondence to the

control point line segments $\mathbf{M}_{ij}$ is available at this stage. Their quality is characterized by the standard deviations of their position component orthogonal to the line segment and of their direction. In a first approximation the standard deviation of their position component is $\sigma_{x'}/\sqrt{l}$ and of their direction $\sigma_{x'}\sqrt{12/l^3}$, where $\sigma_{x'}$ is the positional accuracy of an edge pixel, say 0.3 [pel], and l is the length of the line segment measured in pixels.

2.2 Automatic Orientation

1. step: Probabilistic clustering. For finding good approximate values for the projected control point features a probabilistic clustering, described in (SESTER & FÖRSTNER 1989) determines the position $\widehat{\mathbf{y}}_i$ of each individual *reference point separately.* This is achieved by integrating, realized by an accumulator, the likelihoods of each observed image segment $\mathbf{l}'_{ik}$ matching to each projected model line segment $\mathbf{m}'_{ij}$. In this context a parallel projection for the small image windows is sufficient, and only a translation $\mathbf{t}$, between the predicted model set and the observed segments is determined. Only line segments $(\mathbf{l}'_{ik}, \mathbf{m}'_{ij})$ with similar direction give rise to a range of translation values $\mathbf{t}_i$ their likelihood depending on the length difference, on the uncertainty of the observed line segments and on the assumed uncertainty of the translational model.

2. step: Orientation with points. The resulting set $\{\widehat{\mathbf{y}}_i\}$ of estimated reference points may contain blunders. This mostly occurs due to similar line segment patterns in the vicinity of the building caused by shadows or due to bad image contrast. Using these initial matches $\{\mathbf{Y}_i, \widehat{\mathbf{y}}_i\}$, namely the reference coordinates $\mathbf{Y}_i$ of each control point and its measured image coordinates $\mathbf{y}'_i$, we determine the six orientation parameters. Here we implemented two versions:

A For all *quadruples of control points* we determined the orientation parameters by minimizing the reprojection error using a bundle adjustment. In case the geometric configuration is acceptable (see sect. 2.3) the smallest robust sum $\sum_l \min(\widehat{e}_l^2, t_e^2)$ of the square of all remaining residuals $\widehat{e}_i$ indicated the most likely set of outliers, which then do not take part in the last step. The tolerance t_e was set to 10 pixels. In this version the initial approximate values or the orientation parameters are used for the final step 3.

B Here we attempt to perform a *complete search for the erroneous control points.* In case of N control points this requires $M = \sum_{i=4}^{N} \binom{N}{i} = 2^N - (1 + N + \binom{N}{2} + \binom{N}{3})$ trials, as a minimum of 4 points is necessary to obtain a check on errors. In order to reduce the number of trials, we first compute the exterior orientation with *all* N control points. If this rectification yields an acceptable precision, i. e. the estimated reprojection standard deviation $\widehat{\sigma}_{x'} < 1.5$ [pel], the search is stopped and no error has been found. In all other cases we continue to evaluate all N cases with $N-1$ control points (and one error), then all $\binom{N}{2}$ cases with $N-2$ points, etc. until the threshold 1.5 [pel] has been reached.
All erroneous control points are re-projected by using the determined orientation to get a new set of matching image line segments for these control

point models. The resulting orientation parameters are used as initial values for the following final step.

3. step: Estimation for orientation with line segments. In the last step we use all the line segments $\mathbf{l}'_{ik}$ of *all* control point models and perform a robust maximum-likelihood-type bundle adjustment for the image, minimizing the weighted reprojection error. The re-weighting scheme of the ML-estimation is required in order to eliminate wrong matches of line segments. The weights of the line segments are taken from the feature extraction. All available information is used, not only the reference points as in the last step.

2.3 Self-diagnosis

Self-diagnosis requires performance measures and criteria for their evaluation. The following two types of measures can be used for both steps, the estimation with points and the estimation with line segments, cf. (FÖRSTNER 2001):

Measuring the precision for self-diagnosis: The first performance measure reflects the achieved precision, namely the influence of random errors between observations and assumed model. The estimated variance factor $\widehat{\sigma}_0^2$ is derived from the reprojection errors by (MIKHAIL & ACKERMANN 1976)

$$\widehat{\sigma}_0^2 = \frac{\sum_{ik} \widehat{\boldsymbol{e}}_{ik}{}^{\mathsf{T}} \boldsymbol{\Sigma}^{-1}_{l_{ik}l_{ik}} \widehat{\boldsymbol{e}}_{ik}}{R} \qquad \widehat{\sigma}_0 \le T_P \tag{1}$$

where $\widehat{\boldsymbol{e}}_{ik}$ is the reprojection error of the k-th image feature, point or line segment, at control point i, $\boldsymbol{\Sigma}_{l_{ik}l_{ik}}$ the a priori covariance matrix of the image feature, again point or line segment, and $R = \mathrm{rank}(\boldsymbol{\Sigma}_{ll}) - 6$ the redundancy of the single image bundle adjustment.

In case the coordinates of the observed features are normally distributed, the a priori covariance matrix is chosen realistically, and the model $\mathbf{x}' = \mathbf{PX}$ holds, the estimated variance factor will have an expected value 1, following a $F_{R,\infty}$ distribution. However, as all types of errors in the underlying model may have an influence on the estimated variance factor and the model only holds approximately, the Fisher-test, if applied rigorously, almost always leads to a rejection of the model. It has been found empirically, that assuming the standard deviation $\sigma_{x'}$ to be larger by a factor 2 or 3 is reasonable.

Measuring the quality of the configuration for self-diagnosis: The second performance measure evaluates the difference between the geometric configuration of the control point models and a reference configuration. It actually measures the closeness of the geometric configuration to a critical configuration.

The standard reference configuration for this application consists of 4 control points located in the corners of the image. This configuration would guarantees a reliable determination of the exterior orientation. The measure compares the expected precision reachable with both configurations. It uses the largest possible variance $\sigma_g^{(est)2}$ of a function $g = \boldsymbol{f}^{\mathsf{T}}\widehat{\boldsymbol{p}}$ (BAARDA 1973)

$$\lambda_{\max} = \max_g \left(\frac{\sigma_g^{(est)}}{\sigma_g^{(ref)}} \right)^2 = \max_f \frac{\boldsymbol{f}^{\mathsf{T}} \boldsymbol{\Sigma}^{(est)}_{\widehat{p}\widehat{p}} \boldsymbol{f}}{\boldsymbol{f}^{\mathsf{T}} \boldsymbol{\Sigma}^{(ref)}_{\widehat{p}\widehat{p}} \boldsymbol{f}} \qquad \lambda_{\max} \le T_C \tag{2}$$

of the orientation parameters $\widehat{\boldsymbol{p}}$ in relation to that variance $\sigma_g^{(ref)2}$ reachable with the reference configuration i. e. with covariance matrix $\boldsymbol{\Sigma}_{\widehat{p}\widehat{p}}^{(ref)}$. The maximum at the same time is the maximum eigenvalue of the matrix $\boldsymbol{\Sigma}_{pp}^{(est)}[\boldsymbol{\Sigma}_{pp}^{(ref)}]^{-1}$.

In case the configuration is close to a critical one, then $\lambda_{\max}$ would be very large. A threshold $T_C = 10$ for the configuration measure $\lambda_{\max}$ has been found to be reasonable, as then the achievable standard deviations stay below approx. 3 times the standard deviations of the desired reference situation, being consistent with a three-fold larger measuring standard deviation $\sigma_{x'}$ above.

3 Evaluation of the Manually Designed Self-diagnosis

3.1 External Evaluation

In order to decide, whether the self-diagnosis was successful or not one needs an external reference, or ground truth. To establish a reference data set playing the role of ground truth is usually an enormous effort. Here we evaluated the automatic procedure with the result of a simultaneous bundle block adjustment of *all* available images, and manually measured ground control points. The complete set of images shows an overlap of at least 60 %, not only of about 20 %, as this is the case for the images used for ortho-photo production, which, when used alone, would not allow a bundle adjustment. This reference bundle adjustment results in highly reliable orientation parameters.

As the final goal is the computation of ortho-photos, i. e. the rectification of the aerial images to the geometry of the map, we measure the *maximum* of the *planimetric distortion* $\Delta\boldsymbol{X} = \boldsymbol{X}^{(est)} - \boldsymbol{X}^{(bundle)}$ at the ground by re-projecting a 3×3-grid from the images to the ground (cf. fig. 2) using both the orientation parameters $\widehat{\boldsymbol{p}}^{(est)}$ of the automatic procedure and the orientation parameters $\widehat{\boldsymbol{p}}^{(bundle)}$ of the complete bundle adjustment:

$$D_{\max} = \max \sqrt{(X_m^{(est)} - X_m^{(bundle)})^2 + (Y_m^{(est)} - Y_m^{(bundle)})^2}\,, \quad D_{\max} \leq T_D \tag{3}$$

Following the requirements of the application, the average planimetric distortion should not exceed T_D =1.5 m.

3.2 Empirical Evaluation of the Self-diagnosis

A first evaluation of the quality of the self-diagnosis used the two criteria for precision and configuration in version A, thus just the thresholded mean reprojection errors and the thresholded configuration measure. The external evaluation was based on the distortion.

The result of appr. 17 000 samples, i. e. image orientations is shown in the following table: 69 % orientations were correct and this was reported by the self-diagnosis. In 18 % of the cases the orientation was incorrect, which was detected by the analysis. The percentage of false negatives with 11 % is high. Though the

Table 1. Result of an empirical test using appr. 17000 aerial images.

		reality	
		correct	wrong
self-diagnosis	correct	69 % (correct decision)	2 % (false positives)
	wrong	11 % (false negatives)	18 % (correct decision)

percentage of false positives with 2 % is quite low, this still represents appr. 340 images!

In nearly all cases, the wrong orientations resulted from errors in the second step (cf. sect. 2.2), namely the error detection using the reference coordinates of the control points.

This result, especially the low success rate of 69 %, was the motivation to increase the quality of the self-diagnosis.

4 Learning Criteria for Self-diagnosis Using SVM

In order to improve the performance of the orientation procedure we tried to use the result of the second step, i. e. the orientation with points, to predict the final result and at the same time to automatically learn the criteria for self-diagnosis using support vector machines (SVM, (SCHÖLKOPF et al. 1998)). We also used version B, i. e. all meaningful orientations per image.

For finding a good predictor we used the following four features $\boldsymbol{x} = (x_1, x_2, x_3, x_4)^\mathsf{T}$, which are available after the second step, and which can be expected to be useful for predicting the final result:

1. x_1: the number N of control points in the image, because we expect larger numbers N to increase the performance
2. x_2: the number E of control points eliminated for the final estimation process, as we expect smaller numbers E to increase the performance.
3. x_3: the precision measure $\widehat{\sigma}_0$, as we expect smaller $\widehat{\sigma}_0$ to increase the performance. Actually we used logarithm, which is identical to the negative self-information $-I(\widehat{\sigma}_0) = \log \widehat{\sigma}_0$.
4. x_4: the configuration measure $\lambda_{\max}$, as we assume smaller $\lambda_{\max}$ to increase the performance. In this case we actually used the logarithm $\log(\lambda_{\max})$ too.

For training we used *all M trials* of the orientation with points in step 2. They are used for determining the best decision function $d(\boldsymbol{x})$ for a given sample $(\boldsymbol{x}, y)_i$. For labeling the images as $y =$correct or $y =$wrong we performed the ML-estimation of step 3 and evaluated the orientation using the distortion $D_{\max}$ at ground level.

We used the support vector machine implementation LIBSVM of Chang and Lin (CHANG & LIN 2002) for determining a decision function $d(\boldsymbol{x})$. As it is a binary decision problem, the decision function determines an optimal evaluation

using the score function $s(\boldsymbol{x}) = \text{abs}(d(\boldsymbol{x}))$, thus thresholding at 0: if $d(\boldsymbol{x}) < 0$ the case is wrong, otherwise it is correct. The resultant support vector classifier determines whether a sample is correct or wrong.

Actually we are not interested in the classification of each of the M trials per image, but only in the best achievable result per image. Looking for the best score per image leads to an unacceptable result, as some of the M cases get a very high score for being wrong, which might be higher than the maximum score of the correct cases.

Therefore we use the original decision function $d(\boldsymbol{x})$ for classification. Instead of thresholding at 0, we choose a threshold T_d leading to the classifier:

$$y = \text{correct}, \text{ if } d(\boldsymbol{x}) > T_d, else \text{wrong} \tag{4}$$

The choice of T_d can be used to optimize the classifier, e. g. minimizing expected costs for the classification result.

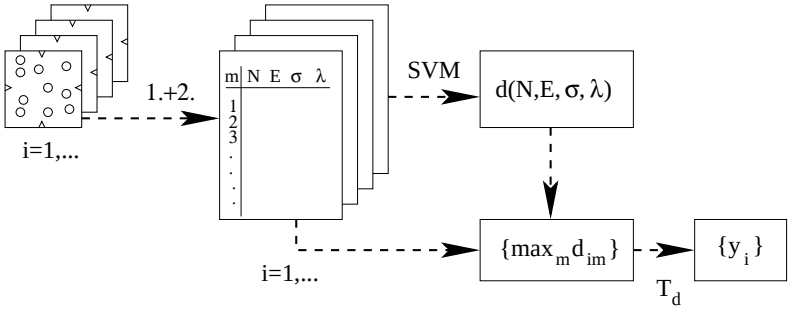

Fig. 3. The procedure for optimizing self-diagnosis: for each image i a set of alternative orientations is calculated in steps 1 and 2 of the procedure, using the N reference points of the control points. The result depends on the number E of eliminated control points. The numbers N and E and the two performance measures $\widehat{\sigma}_0$ and $\lambda_{\max}$ for precision and configuration are used to automatically determine a decision function $d(\boldsymbol{x})$ using binary support vector machines. The best orientation per image is used for classification, leading to a binary y_i, thus $y_i \in \{\text{correct}, \text{wrong}\}$.

4.1 Empirical Accuracy of the Self-diagnosis Based on SVM Decision Function

We tested this procedure on two sets of images of blocks with 60 % overlap covering two areas in Nordrhein-Westfalen. The 436 images from Area I show good quality, in the sense that all images are of good quality and contain enough control points, whereas the 243 images from Area II do not have good image quality and partially contain badly identifiable control points.

As we now have one free parameter for classification, namely the threshold T_d for deciding between correct and wrong orientations we give the confusion tables for four values of T_d.

We give the result as confusion tables with the number of images for the four cases ($selfdiagnosis$: correct|$reality$: correct), ($selfdiagnosis$: correct|$reality$:

wrong) etc. The confusion tables are obtained by using approximately half the data set for training of the SVM and the other half for testing.

To evaluate the quality of the resulting confusion tables we might use the expected cost. They depend on the costs for each case {cc, cw, wc, ww} (abbreviating the labels). The expected cost is

$$E(C) = P_{cc}C_{cc} + P_{cw}C_{cw} + P_{wc}C_{wc} + P_{ww}C_{ww}$$

which requires a specification by the user. For a first evaluation we use the additional costs $C_{cc} = 0$, $C_{cw} = 40$, $C_{wc} = 5$, $C_{ww} = 10$. The expected cost is given below the confusion tables.

	$T_d = 0$			$T_d = 1$			$T_d = 2$			$T_d = 3$		
		c	w		c	w		c	w		c	w
1 — 2	c	173	21	c	172	17	c	166	5	c	140	5
	w	0	2	w	1	6	w	7	18	w	33	18
	$E(C)$=4.39			$E(C)$=3.80			$E(C)$=2.11			$E(C)$=2.78		
		c	w		c	w		c	w		c	w
2 — 1	c	233	7	c	229	6	c	217	5	c	172	0
	w	0	0	w	4	1	w	16	2	w	61	7
	$E(C)$=1.17			$E(C)$=1.13			$E(C)$=1.25			$E(C)$=1.56		

Table 2. Results from Area I. In each table, column index (c, w): result of self-diagnosis, row index (c, w): reference. Changing the threshold T_d may be used to increase performance, e. g. by reducing the number of false positives (upper right). Expected cost below tables. First row of tables: training data set: 240 images, test set: 196 images. Second row of tables: exchange of test and training data.

Choosing a threshold of $T_d = 2$ appears to be optimal. The expected cost, averaged over both versions 1—2 and 2—1, with $E(C) = 1.64$ is lower than the expected cost for the earlier version of the classification, where the expected cost is $E(C) = 3.15$. The result for the image set II shows significant worse results (cf. table 3). There appears to be no clear minimum of the expected cost in that range of thresholds. We also investigated whether the classifiers could be generalized to other data sets. We trained the classifier with one data set, I or II, and tested it on the other one. The two data sets turned out to be too different.

	$T_d = 0$			$T_d = 1$			$T_d = 2$			$T_d = 3$		
		c	w		c	w		c	w		c	w
1 — 2	c	72	35	c	64	21	c	45	10	c	25	4
	w	1	16	w	9	30	w	28	41	w	48	47
	$E(C)$=12.62			$E(C)$=9.55			$E(C)$=7.66			$E(C)$=7.01		
		c	w		c	w		c	w		c	w
2 — 1	c	66	18	c	55	9	c	21	4	c	1	0
	w	1	14	w	12	23	w	46	28	w	66	32
	$E(C)$=8.74			$E(C)$=6.56			$E(C)$=6.76			$E(C)$=6.56		

Table 3. Results from Area II. For each confusion table, column: result of self-diagnosis, row: reference. Expected cost below tables. First row of tables: training data set: 99 images, test set: 124 images. Second row of tables: exchange of test and training data.

4.2 Simplifying the Classifier

The scatterplots of pairs of features suggest that the two performance measures for precision and configuration might be sufficient. The scatterplot for these two features shows that a linear decision boundary with a slope of -0.5 might do. The normal of this line approximately is $\boldsymbol{w} = (2, 1)$. Therefore we investigated the following single feature $x = 2x_3 + x_4 = 2 \log \widehat{\sigma}_0 + \log \lambda_{\max} = \log(\widehat{\sigma}_0^2 \lambda_{\max})$. This appears reasonable, as $\widehat{\sigma}_0^2 \lambda_{\max}$ is the maximum increase in variance compared to the reference configuration due to both the estimated precision and the configuration.

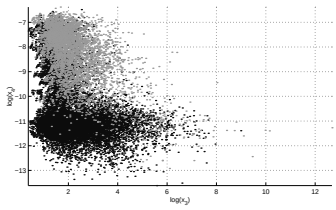

Fig. 4. Cluster of $\log \widehat{\sigma}_0$ (right) and $\log \lambda_{\max}$ (up)

However, using only this feature for classification leads to slightly worse results: The expected cost in the good data set decreased by nearly a factor 2, whereas the expected cost in the bad data set II did not change very much.

5 Conclusions

Integrating self-diagnosis into a system for automatic orientation enabled to increase its performance by training the self-diagnosis using support vector machines. The setup appears general enough and may be transferred to any type of vision system, even if its primary output is not the result of a classifier.

References

Chang, C.-C. & Lin, C.-J. (2002): LIBSVM: a Library for Support Vector Machines (Version 2.33), *Technical report*, Dep. of Computer Science and Information Engineering, National Taiwan Univ., Taipei 106, Taiwan: last update: January, 2002.

Baarda, W. (1973): S-Transformations and Criterion Matrices, *Netherlands Geodetic Commission*, Ser. 1, Vol **5**.

Förstner, W. (2001): Calibration and orientation of cameras in computer vision, *in* A. Grün & T. Huang (Eds.), *Generic Estimation Procedures for Orientation with Minimum and Redundant Information*, Springer.

Mikhail, E. M. & Ackermann, F. (1976): Observations and Least Squares, University Press of America, 1976

Schölkopf, B., Burges, C. J. C. & Smola, A. J. (1998): Introduction to Support Vector Learning, *in* B. Schölkopf et al. (Eds.), *Advances in Kernel Methods: Support Vector Learning*, MIT Press, Cambridge, chapter 1, pp. 1–15.

Sester, M. & Förstner, W. (1989): Object Location Based on Uncertain Models, *Mustererkennung 1989*, Springer Informatik Fachberichte, **219**, pp. 457–464