AUTOMATIC RELATIVE ORIENTATION OF IMAGES[1]

T. Läbe and W. Förstner

Institute for Photogrammetry, University of Bonn, Germany - (laebe, wf)@ipb.uni-bonn.de

KEY WORDS: automatic orientation, matching, rotation invariance, SIFT, RANSAC

ABSTRACT:

This paper presents a new approach to full automatic relative orientation of several digital images taken with a calibrated camera. This approach uses new algorithms for feature extraction and relative orientation developed in the last few years. There is no need for special markers in the scene nor for approximate values for the parameters of the exterior orientation. We use the point operator developed by D. G. Lowe (Lowe, 2004), which extracts points with scale- and rotation-invariant descriptors (SIFT-features). These descriptors allow a successful matching of image points even in situations with highly convergent images. The approach consists of the following steps: After extracting image points on all images each image pair is matched using the SIFT parameters only. No prior information about the pose of the images or the overlapping parts of the images is needed. For every image pair a relative orientation is computed using a RANSAC procedure. Here we use the new 5-point algorithm developed by D. Nister (Nister, 2004). Based on these orientations approximate values for the orientation parameters and the object coordinates are calculated. This is achieved by computing the relative scale and transforming into a common coordinate system. Several tests are carried out to ensure reliable inputs for the currently final step: a bundle block adjustment. The paper discusses the practical impacts of the algorithms involved. Examples of different indoor- and outdoor-scenes including a dataset of tilted aerial images are presented and the results of the approach are evaluated. These results show that the approach can be used for a wide range of scenes with different types of the image geometry and taken with different types of cameras including inexpensive consumer cameras. In particular we investigate in the robustness of the algorithms, e.g. in geometric tests on image triplets. In the outlook further developments like the use of image pyramids with a modified matching are discussed.

1. INTRODUCTION

Orienting images is one of the basic tasks in photogrammetry. Automating this process has a long tradition in research. For aerial images this task can be considered as solved; methods are available commercially (HATS by Helava/Leica, Phodis-AT by Zeiss, ISDM by Intergraph, MATCH-AT by INPHO). Pollefeys (Pollefeys, 1999) probably was the first to demonstrate the feasibility of full automatic image orientation, which even more, does not presume the calibration of the images to be known. The approach of Hao & Mayer (Hao & Mayer, 2003) has the same goal. Stimulated by the tasks of ISPRS Working Group III/1 „Image Orientation'' (http://www.ipb.uni-bonn.de/isprs/wg.html) our aim is to develop a system for the automated orientation of images which can be handled by a non-specialist. The requirements are high:

1. The system should be able to handle both types of images; where the calibration is known or not known.
2. The definition of control information should be flexible.
3. If necessary, the system should inform about weak configurations and give hints at how to resolve the situation.
4. In order to allow immediate response to diagnostic reports, the system should work in real time

We started to realize the system by restricting to the relative orientation of images of calibrated cameras. This paper reports our concept and shows first results on real images.

2. THE PROCEDURE

Figure 1 gives an overview. As inputs, the procedure takes several digital images together with their interior orientation. In case of image distortions the images need to be rectified in order to obtain straight line preserving images. The procedure first determines all relative orientations, transforms the orientation parameters into a common coordinate system and performs a classical free bundle adjustment to optimally estimate all unknown orientation parameters and 3D-coordinates. The result is a set of oriented images with freely chosen datum parameters. In the following we discuss the individual algorithms.

2.1 Feature extraction

First step of the orientation is the automatic extraction of image features followed by an automatic matching process in order to assign each image detail a unique number referring to the corresponding object detail. As we start with relative orientations we use point type image features. As we do not pose any restrictions on the exterior orientation their detection and description should be scale and rotation invariant. We therefore use the point operator proposed by D. Lowe (D. Lowe, 2004).

The operator detects points in an image pyramid and describes the points by means of rotation and scale-invariant features, so called SIFT-features („Scale-Invariant Feature Transform''). The descriptor represents the scale dependent window around the point with 16 histograms of the gradient orientations leading to 128 values in the range between 0 and 255. As local

[1] Published in Proceedings of the 5th Turkish-German Joint Geodetic Days, March 29th – 31st, 2006, Berlin, ISBN 3-9809030-4-4

projective distortions can often be approximated quite well by a scale and a rotation transformation, using SIFT-features allows handling nearly arbitrary camera positions provided that enough tie points are available. This is an important advantage compared to procedures which employ a conventional point detector and subsequent correlation or least squares matching. Such procedures only work with relative rotations up to approximate 15 degrees; cf. the approach by Hao & Mayer (Hao & Mayer, 2004).

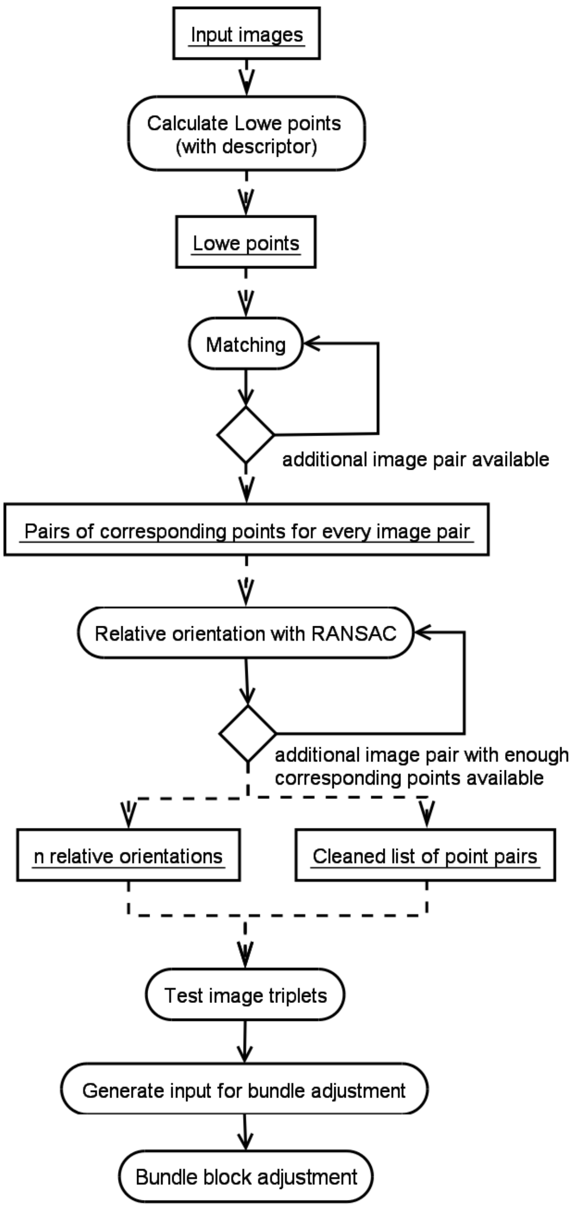

Figure 1. UML-activity diagram for the entire procedure of relative orientation of multiple images. The individual steps are discussed in sect. 2.

2.2 Matching

In our current implementation per default we assume that no prior information about the relative pose of the n images is available. Therefore we match all image pairs, leading to $O(n^2)$ image pairs to match. If n is large (e.g. n>15), it is computational expensive to match all image pairs. In these cases the user of our module has the possibility to define the images as an image sequence or an image block with certain extends (regardless of the rotations of the images). Then the number of considered image pairs can be reduced to $O(n)$, because only neighbouring images are matched.

The matching procedure for an image pair leads to a list of corresponding point pairs. As a certainty measure d for a point correspondence we simply use the Euclidean distance of the SIFT-feature vectors. The smaller this distance the more likely the two points correspond. This calculation is necessary for each possible point pair of two images, because we have no prior information about the relative pose of the images. This calculation is feasible, if the size of the images is not too large.

The user can currently choose between two matching strategies:

Matching strategy I: For each point in the first image of an image pair we search for the best and second best candidates in the other image. In case the ratio $r=d_{second}/d_{first}$ of the certainty measures for the second best to the best candidate is lower than a certain percentage, e.g. r<60 %, we accept the correspondence and use it in the following steps.

Matching strategy II: All correspondences with certainty measure better than a certain threshold are accepted. Only in case the number of correspondences for a point is larger than a threshold n_{cmax}, the matching seems uncertain and all correspondences are deleted.

The threshold for strategy I can be defined quite easily, as it is well interpretable. However, also the threshold for the certainty measure in strategy II can be found easily by a few representative experiments, as the threshold refers to the SIFT-feature values which are in an image independent range 0 to 255. Figure 2 shows a representative histogram of the distance between feature vectors of correctly matched points.

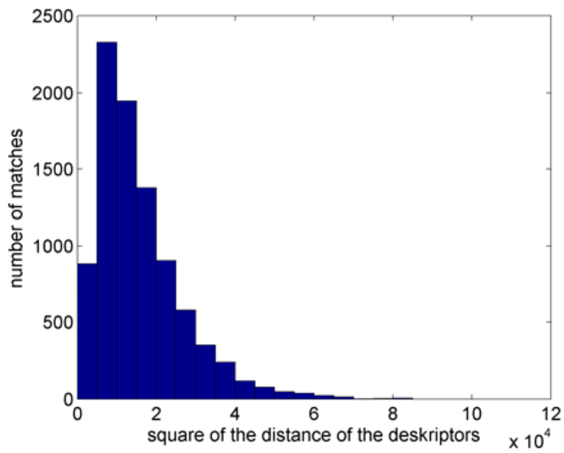

Figure 2. Distribution of the squared distance between the descriptor vectors for correct correspondences. They were determined with strategy I and verified by the simultaneous orientation of 37 images from 3 different data sets. A reasonable threshold therefore lies between approximate $6x10^4$ and $8x10^4$.

Strategy I has the advantage, that the resulting correspondences are quite certain. However, it yields only one correspondence per point and therefore might lead to a set of correspondences which is too small for further processing. Strategy II is meant to have an advantage in images with repetitive patterns, e. g. in images of facades, as the decision on the correct correspondence is left to the subsequent relative orientation. Both strategies were experimentally evaluated with 6 data sets (data sets 1-6 in Table 1). The comparison showed that strategy I is able to deliver reliable results even with less than 10% outliers especially when using a low threshold (60%). But it can not be used with every data set because of the too small number of correspondences. Strategy II should be used with a not too small threshold for the maximal number n_{cmax} of correspondences per point (e.g. $n_{cmax} >= 4$). We use strategy I as default.

2.3 Relative Orientation of image pairs

We determine a relative orientation for each image pair containing enough matching candidates. As there still may be quite many wrong correspondences we need to apply a robust estimation which furthermore does not require approximate values. We use a RANSAC-procedure ("Random Sample Consensus") following Fischler & Bolles (Fischler & Bolles, 1981). It randomly selects five candidate matches, determines the parameters of the relative orientation and selects the best sample based on the total fulfilment of the coplanarity constraints. The kernel of this procedure is the recently published direct solution for the relative orientation from five point pairs (D. Nister, 2004). The solution is not unique; one may obtain up to 10 solutions. In practice, however, only 4 or 6 solutions are found in most cases, some of which may be eliminated by requiring the 3D-points of the photogrammetric model to lie in front of the two cameras. For all accepted solutions we determine their quality in the RANSAC loop by means of all coplanarity constraints.

The number of required trials (e.g. 292) depends on the expected error rate (e.g. 60 %) and the required probability to find a correct solution (e.g. 95 %). The accepted parameters of the relative orientations allow us to eliminate bad correspondences.

2.4 Image triplet test

Before generating the input for the bundle adjustment, the relative orientations of the image pairs can be validated using the geometry of image triplets. If all three pair wise orientations of an image triplet are available, they must be consistent (Trautwein et al., 1999): the product of the rotation matrices must yield the unit matrix and the three base vectors must be coplanar. These tests are computationally efficient and therefore can be computed for every triplet. As a relative orientation of an image pair may be part of several image triplets, it can be tested several times. The generation of the input of the bundle block adjustment is started with those pair wise orientations which have been tested successfully with many triplets.

2.5 Bundle block adjustment

The final step is a bundle adjustment. It requires approximate values for all orientation parameters and all coordinates of object points.

Therefore, we first transform all photogrammetric models into a common coordinate system. The sequence is determined by the quality of the image pairs (successful triplet tests and number of correspondences). The best image pair defines the coordinate system. The other images are integrated sequentially, increasing the bundle block step by step. An image pair is selected with an image which is already integrated and shows the best quality. Common points, i.e. triple points, are used for the scale transfer. If there are no triple points, another image pair is selected.

Data set	Camera	Resolution in Mpixel	Pyramid level	# images	scene
1	HP 435	3	2.	6	Table
2	HP 435	3	2.	3	Regular metal grid on the wall and chair
3	HP 435	3	1.	4	Big photo and box
4	Canon EOS1 Ds	11	2.	4	fassade with large distance between photos
5	HP 435	3	1.	4	Map on the wall
6	HP 435	3	1.	5	Map on the wall
7	HP 435	3	1.	32	Boxes and books on a floor
8	Rollei AIC-45	22	2.	70	Oblique aerial images of harbour
9	RMK-TOP	7678x7678 pixel	4.	43	Aerial images

Table 1. Some data sets used for the test of the orientation procedure

Before actually integrating the image the triple points are geometrically checked for consistency. This is, because the correspondence of three points might be wrong even though the pair wise correspondences are correct. We follow the proposal in (McGlone, 2004, S. 268). It checks the intersection of certain four planes through the three projection centres: only when the three points really correspond, these planes intersect in one 3D-point. This can be checked by only using image coordinates and the orientation parameters without determining the 3D-point.

Thus, we integrate images into the block until no images are left or the remaining image pairs cannot be integrated. In this case only a part of the complete set of images can be orientated. Finally, we determine approximate values for the 3D-points by triangulation.

For efficiency we may reduce the number of points as a next step. In order to achieve a homogeneous reduction, we partition all images into patches and require that the reduction does not lead to a number of points per patch which is smaller than a threshold. For stability reasons we start with 2-fold points then eliminate 3-fold points, etc. The final adjustment yields variances for all orientation parameters allowing an evaluation of the final quality.

The program is written in MATLAB, including the bundle adjustment. The point operator and the matching algorithms are written in C. The triangulation uses a Java library.

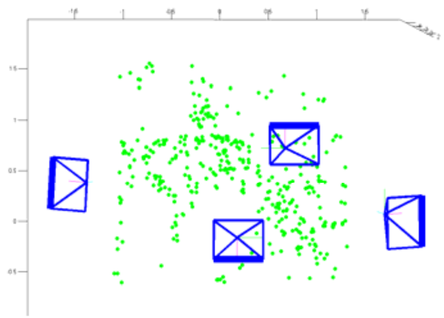

Figure 3. 3D-visualisation of the camera poses and of the object points of data set 5 (images with 3 mega pixels, first pyramid level; camera: HP 435). One side of the image is indicated with a bold line in order to visualize the rotation.

3. RESULTS

The procedure has been tested with a large number of various data sets which cannot all be shown here. For these tests we mainly used digital consumer cameras which are not produced for photogrammetric purposes, and which – under certain conditions - show a stable interior orientation (Läbe & Förstner, 2004). Table 1 summarises some information about 9 data sets.

Figure 3, Figure 4 and Figure 5 show the results of the orientation of the data sets 5, 6, and 7 as 3D-views. Figure 3 and Figure 4 are results of taking views of a map and clearly show that the algorithm can cope with large rotations (rotation by 90° in Figure 3) and with large scale differences (Figure 4).

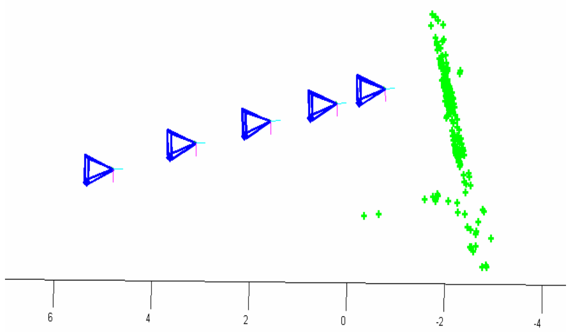

Figure 4. 3D- visualisation of the camera poses and of the object points of data set 6 (images with 3 mega pixels, first pyramid level; camera: HP 435)

Experience shows that the texture of the objects is much more decisive than the geometry of the setup, provided there is enough overlap. In ideal cases the texture shows rich details as for example on posters or paintings. Data set 5 contains 429 object points (out of 1579 before the reduction for the bundle adjustment), data set 6 contains 254 points (out of 1153). Fig. 5 demonstrates that also larger blocks can be handled. This data set consists of 32 images with 3000 object points.

The next example shows the orientation of a set of 70 aerial images taken with a Rollei AIC-45 digital camera (thanks to Prof. R. Bill, University of Rostock, for making the images available). The oblique images (cf. Figure 6) are taken from a helicopter flying around the harbour of Rostock. The orientation was done on the second image pyramid level resulting in an average of 5450 image points per image. Due to the matching, 35700 object coordinates could be calculated but only 1166 were used for the bundle adjustment. Points with only 2 image measurements were not used in the adjustment, because there were matches on the wing of the plane which can't be detected as unusable because the epipolar constraint holds. The average number of image points per object point was 4.8. The 3D-visualisation (cf. Figure 7) demonstrates that the automatic method is able to cope with non-standard imaging conditions.

The last example (data set 9) is a set of 43 classical aerial images with cross strips (thanks to Inpho GmbH, Stuttgart, for providing the images). Without any prior information of the

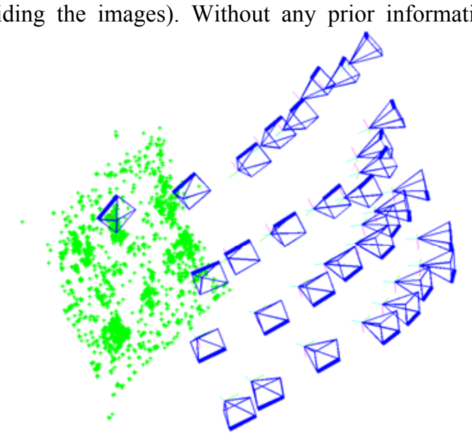

Figure 5. Data set 7 with 32 images and 3000 object points used for automatic orientation.

block we were able to do a relative orientation of all images on a high pyramid level (image size was 480x480 pixel on that pyramid level). The matching of the SIFT-features worked very well with a small number of outliers. Therefore, an investigation into using the Lowe operator on aerial images seems to be promising. The resulting orientation can be used as approximate values for a classical aerial triangulation if there are no projection centre coordinates or orientation angles (especially κ) available.

Figure 6. Example image of data set 8: harbour of Rostock, Germany.

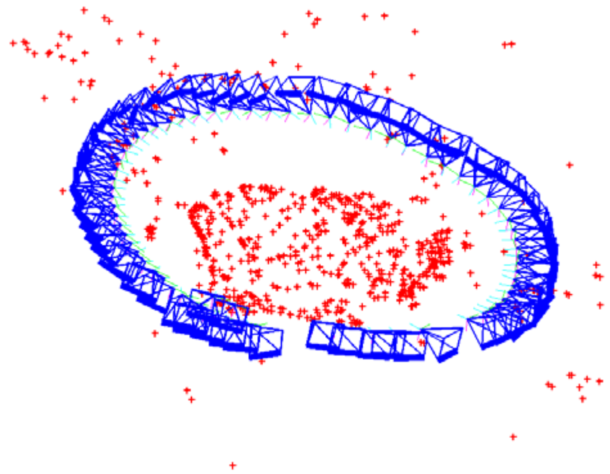

Figure 7. 3D View of data set 8. The dataset consists of 70 images and 1166 objects points.

4. SUMMARY AND OUTLOOK

Following the goal of developing a general-purpose tool for the full automatic orientation, our experiences with the relative orientation of calibrated cameras are encouraging and reinforce that full automatic approaches for orientation are possible even in close-range photogrammetry applications.

The approach and the implementation could be improved by means of the following ideas:

- *Consider small base vectors.* If the ratio between the base vector length and the mean distance of the cameras to the object points of a relative orientation of an image pair becomes small, the orientation parameters cannot be determined accurately. Therefore such relative orientations should not be used for the generation of the bundle adjustment input if there are more reliable orientations available. This fact could be considered when generating the bundle adjustment input. It may be useful (but time consuming) to do an adjustment of the pair wise orientations to get accuracies of the orientation parameters.
- *Bundle adjustment with damping technique.* To increase the convergence radius, one could use the line search damping technique (Börlin et al., 2004). This should allow using also less accurate relative orientations of image pairs.
- *Bundle adjustment with robust estimation.* The consistency tests carried out before the bundle adjustment might not be sensitive enough to detect small gross errors. A robust bundle adjustment should be able to cope with such errors.
- *Use of image pyramids.* It would be useful to carry out a multiple scales approach and integrating higher pyramid levels. On a high pyramid level an initial orientation could be computed. The epipolar geometry could then be used for finding correspondences in the original images without again calculating the relative orientations.
- *Reduction of computation time.* The program is not optimised for speed. Data sets 5 and 6 need about 4 minutes on a PC with Pentium IV processor and 2.2 GHz clock rate, data set 8 needs about 5.5 hours. There should be a large potential for program optimisation. We think that an implementation in a lower level programming language than Matlab would significantly increase the computation speed.

Additional problems to cope with are: the use of uncalibrated cameras (estimating the interior orientation parameters within the orientation procedure or writing a calibration module that uses natural points), flexible integration of control point information and a self-diagnosis to help non-photogrammetrists to use the system. It would be useful that the system makes suggestions, e.g. where to take additional photos.

In summary we could state that we have taken a first step towards the full automation of the orientation for photogrammetrists, for the neighbouring disciplines and – later on – for everyone.

REFERENCES

Börlin, N., Grussenmeyer, P. Eriksson, J. Lindström, P., 2004: Pros and Cons of Constrained and Unconstrained Formulations of the Bundle Adjustment Problem. In: *International Archives of ISPRS, Volume XXXV, ISPRS Congress Istanbul 2004*, pp. 589-594.

Fischler, R., Bolles, R.C., 1981: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Comm. of the ACM*, Vol. 24, pp. 381-395.

Hao, X., Mayer, H., 2003. Orientation and Auto-Calibration of Image Triplets and Sequences. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. (34) 3/W8, pp. 73-78.

Läbe, T., Förstner, W. 2004: Geometric Stability of Low-Cost Digital Consumer Cameras. In: *Proc. International Archives of ISPRS, Volume XXXV, ISPRS Congress Istanbul 2004*, pp. 528-535.

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

McGlone, J. C., editor, 2004: *Manual of Photogrammetry*. Fifth edition, American Society for Photogrammetry and Remote Sensing, Bethesda, Maryland, USA.

Nistér, D., 2004: An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(6): pp. 756-770.

Pollefeys, M., Koch, R., and Van Gool, L. 1999: Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters, *International Journal of Computer Vision*, 32(1), pp. 7-25.

Trautwein, S., Mühlich, M., Feiden, D. und Mester, R. 1999: Estimating Consistent Motion From Three Views: An Alternative To Trifocal Analysis. *LNCS, vol. 1689*, Springer Verlag.