Application of Computer Vision Methods and Algorithms in Documentation of Cultural Heritage

David Káňa[1], Vlastimil Hanzl[2]

[1]Geodis Brno, ltd
Lazaretní 11a, Brno, Czech Republic
dkana@geodis.cz

[2]Brno University of Technology, Faculty of Civil Engineering
Veveří 331/98, Brno, Czech Republic
hanzl.v@fce.vutbr.cz

Abstract

The main task of this paper is to describe methods and algorithms used in computer vision for fully automatic reconstruction of exterior orientation in ordered and unordered sets of images captured by digital calibrated cameras without prior informations about camera positions or scene structure. Attention will be paid to the SIFT interest operator for finding key points clearly describing the image areas with respect to scale and rotation, so that these areas could be compared to the regions in other images. There will also be discussed methods of matching key points, calculation of the relative orientation and strategy of linking sub-models to estimate the parameters entering complex bundle adjustment. The paper also compares the results achieved with above system with the results obtained by standard photogrammetric methods in processing of project documentation for reconstruction of the Žinkovy castle.

Keywords: computer vision, interest operator, matching

1. Introduction

Images captured by digital cameras are one of the most important form of information in documentation of cultural heritage. Effective assignment of camera pose in space is necessary for consequential usage for measuring purposes. The automatic process of finding exterior orientation can be divided to three main tasks: Key point finding and matching, relative orientation and bundle adjustment. Our paper presents practical experiment of such procedure.

2. Key Points Extraction

2.1. SIFT – scale invariant feature transform

The initial phase during comparing and relative orientation of two images is to choose characteristic or key points in images. The key point should by no mean characterize the image area so that this area could be reliably found and compared with the same area in different image. By finding corresponding points in both images a correspondent couple (correspondence) is defined. For detection and comparison of significant points in the image SIFT operator was

chosen as the most convenient detector. This detector is unlike simple correlation between two areas in the images with the size of for example 21x 21 pixels partly invariant toward view geometry change therefore rotation (circa 15 degrees), scale change and is partly invariant to noise as well. SIFT detector is based on searching for extremities in the images by finding differences among images incurred by the convolution of image function $I(x, y)$ and Gauss filter $G(x, y, \delta)$ with variable values of sigma.

$$D(x, y, \delta) = L(x, y, k\delta) - L(x, y, \delta) = (G(x, y, k\delta) - G(x, y, \delta)) * I(x, y). \quad (1)$$

Exact procedure is described for example in [2].

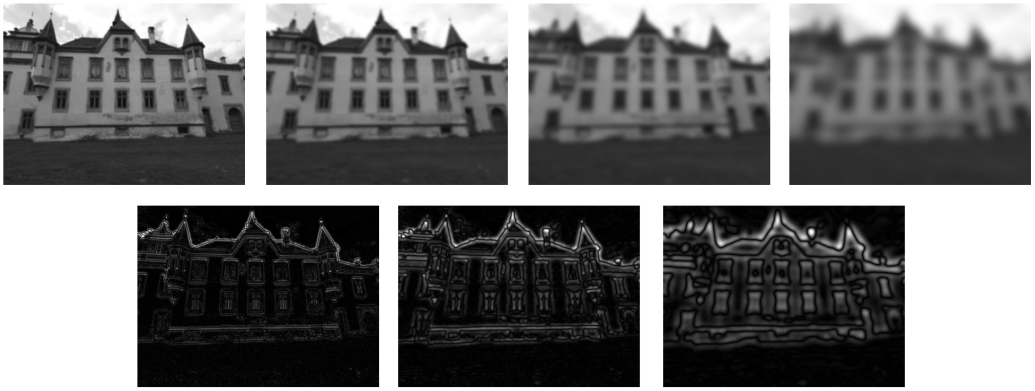

Figure 1: Blurred images and their differences

2.2. Finding extremities

Single images with a different degree of blurring are subtracted each other and difference imagess would come into being. These differences are evidently approximation of the second derivation of the image function $I(x, y)$ and serve to detect local extremities. After creation of differential images (Fig. 1) each pixel value is compared with six neighbouring pixels in the image and nine neighbouring pixels in the image with blurring partly a level higher and partly a level lower. If the value of the tested pixel is the lowest or eventually the highest out of all the neighbouring pixels, this pixel is chosen as possible key point. Once the candidate for the key point is found by comparison with its neighbours it is necessary to decide about its stability and therefore about possible denial on the bases of information about its location, scale and a rate of main curvatures. This information enables effective removal of nondesired points in low contrast areas. For each point is its surrounding in the range of 3x3 pixels approximated by the three dimensional quadratic function. Consequently is found maximum or eventually minimum of this function that defines the exact location of the key point with subpixel accuracy. The points along the edges are according to curvature diameters rate in two perpendicular directions removed as well.

2.3. Orientation assignment

A certain orientation can be assigned to each key point. By this step is ensured key point descriptor invariance toward the image rotation. Descriptor is expressed relatively in the view of key point rotation. Orientation is computed in dependence of smoothing out rate for given key point and does not depend on scale. In blurred image size values $m(x, y)$ and orientation values $\theta(x, y)$ of the function gradient $L(x, y)$ are computed.

Consequently orientations in the surrounding of the key point are computed, when an orientation histogram of neighbouring pixels that contains 36 items for 360 degrees range with 10 degrees interval is put together. In this histogram a peak is found (an item with biggest occurrence) and this dominant rotation is assigned to the key point. Farther it is tested whether other occurrences reaching 80 percent of the biggest occurrence are present. If so a new key point is made out with the same coordinate but with a new orientation given by this direction. After all in some places a rank of key points can ensue with the same coordinate but with various orientations.

2.4. Key point descriptor

The parameters defining the location, scale and orientation of the key points also define auxiliary coordinate system, to which descriptor can be expressed clearly describing the area around key point location. Descriptor is based on image gradients and their orientation in a region surrounding key point, where area 16 x 16 pixels is divided into 16 blocks of 4 x 4 pixels. For every block is created histogram of eight meaningful orientations weighted by gradient magnitude using Gaussian function. Subsequently by ordering and normalizing those 16 partial histograms containing 8 intervals descriptor, vector of dimension 128 is formed.

For key points detection an implementation [1] was used, for another acceleration of computing an implementation with usage of hardware acceleration on CUDA [4] platform would be tested.

3. Finding Correspondences

If there are detected key points in each image counting descriptors, we can approach to pairing and finding of corresponding point couples - correspondences, which came into being by projection of point in three dimensional space to both images. The rate of agreement of the two key points is determined on the basis of their SIFT descriptor vector's Euclidean distance in two ways. Because we do not have any information about image sequence it is necessary to compare them by each to each method.

3.1. Symmetric pairing

To every key point in the first image we can find a point with the nearest descriptor distance in the second image. In the second image is to each key point found the nearest key point in the first image as well. As potential pair of corresponding points is labeled the one where mutual agreement is in both comparisons.

Figure 2: Key points and their orientations

3.2. Distance ratio test

Lowe [2] recommends testifying the agreement of the key points by rate of descriptor Euclidean distance to the first and second nearest point. As limiting he offers distance ratio of 0.6. It is obvious that this criterion can fulfill more key points. For correspondence's stability and reliability reasons it is convenient not to count on these multiple correspondences in farther calculations. This test proofed as not much convenient with markedly repeating texture (for example the facade of panel house images).

4. Fundamental and Essential Matrix

The correspondences set obtained by above mentioned procedures is however loaded with errors and false correspondences, which arouse from a change of camera location, change of lighting, digital image noise and so on. These false correspondences could be eliminated by usage of geometric criterion – epipolar condition.

The X point together with projection centers C and C' forms epipolar plane in 3D space. By epipolar plane intesection with projection planes are formed epipolar lines – epipolars. And points x and x', which are projections of X point into projection plane, lies on this epipolar lines. This lines also pass thru epipoles e and e' where epipole is projection of first camera projection center to the projection plane of the second camera. Algebraic formulation of the epipolar condition is equation (2):

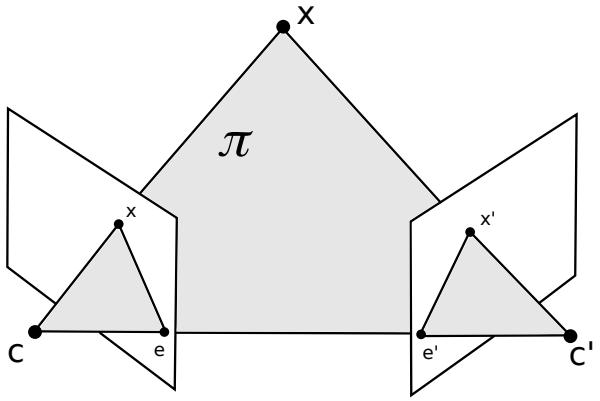

Figure 3: Illustration of epipolar geometry

$$\begin{bmatrix} x' & y' & 1 \end{bmatrix} \quad F \quad \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \tag{2}$$

Where F is fundamental matrix size is 3 x 3 and rank of this matrix is 2. This fundamental matrix defines relation between two cameras without dependency on scene structure. For calculation it is not necessary to know cameras interior orientation parameters.

4.1. Fundamental matrix computation using correspondences

Matrix F has seven degrees of freedom so minimal number of correspondences is seven. Number of solutions can vary from one to three. Due to the numeric stability is more optimal to use eight-point algorithm and find the best solution using SVD decomposition. Formula (2) defines relation between two corresponding points. Providing F:

$$F = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \tag{3}$$

We can obtain one linear equation for each correspondence:

$$x'x\ f_{11} + \ x'y\ f_{12} + x'\ f_{13} + y'x\ f_{21} + y'y\ f_{22} + y'\ f_{23} + x\ f_{31} + y\ f_{32} + f_{33} = 0 \tag{4}$$

For n correspondences we obtain homogeneous system of linear equations in following form:

$$\mathrm{Af} = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} \quad F = 0 \tag{5}$$

Solution which minimizes distances of projected points to epipolar lines is such a vector f where $\|\mathrm{Ax}\|$ is minimal and $\|f\| = 1$.

If UDV^T is SVD decomposition of matrix A, the solution is vector corresponding to the smallest A matrix singular value which is the last column of V matrix. To meet real fundamentality, the F matrix should have rank of 2. Due to noise and small inaccuracies, computed matrix has usually rank of 3. For conversion to rank of 2 SVD decomposition is used again, where last eigenvalue is set to zero and matrixes formed by decomposition are multiplied.

$$F = \mathrm{UDV}^T \qquad D = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix} \qquad \bar{D} = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \bar{F} = U\bar{D}V^T \tag{6}$$

Out of numeric stability reason it is good to reduce input pixel coordinates towards the center of gravity and normalize. For successful result it is necessary that 3D points which projections are used for computation of the fundamental matrix must not lie in one plane. Described eight point algorithm is linear, nonlinear solution can be found for example in [5].

4.2. RANSAC – RAndom SAample Consensus

For key points selection meeting epipolar condition the RANSAC algorithm was used. This algorithm enables to find the best solution suiting given model iteratively. In our case x' F x = 0. Comparing to results obtained by using least mean squares (LMS) method which doesn't take into account blunders and mistakes, results obtained using RANSAC are more consistent. Model (fundamental matrix) is computed in every iteration from randomly selected sample formed by 8 pairs of corresponding key points. Consequently the rest of this points not contained in selected sample is tested against computed model. If pre-specified percentage of key points meets the model parameters and total model error is at the same time smaller than error obtained in previous iteration, given selection is marked as the best obtained selection. In other case is computed model rejected. The computation can be terminated after a specified number of iterations or after reaching the minimal threshold value of error from the statistically significant part of the correspondences. The last step is to determine the fundamental matrix using the SVD method from the best selection (5, 6).

5. Essential Matrix and Relative Orientation

Provided the fundamental matrix F is computed and calibration matrixes of both cameras are known it is possible to formulate equation for essential matrix computation which defines relative position of cameras regardless of scale.

$$E = K'^T\, F\, K \qquad K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

If only relative orientation is solved it is possible to set up first camera projection center into coordinate system origin and its rotation matrix as identity matrix. Projection matrix of the first camera can be formulated: $P = [I \mid 0]$ And for the second camera: $P' = [R \mid t]$ where R is rotation matrix and t is translation vector regardless the scale.

$$E = [t] \times R$$

Rotation and translation we can determine by SVD decomposition of the essential matrix E in following way:

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad Z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{8}$$

$U = \mathrm{SR} = U \ \mathrm{diag}(1, 1, 0) \ V^T$ where $S = \mathrm{UZU}^T \ \ R = \mathrm{UWV}^T$ or $R = \mathrm{UW}^T V^T$.

Mentioned decomposition has four possible solutions. Valid solution is selected by projection depth testing, where projection depth is computed for any point and checked whether it is positive for both cameras. Relative orientation is computed for any image pair combination containing specified minimal number of correspondences. As sufficient number regarding noise appears to be 16 correspondences. In the case of unordered collection of images where no prior parameters are known, the images are compared each to each and the total number of comparisons is $O(n^2)$, where n means number of images. In our practical experiment we selected 25 images and 300 possible combinations were tested. For larger images sets is convenient to use parallel computations. In case of image sequence is suitable due to computation complexity limit the number of compared images to 3 – 5.

6. Sparse Bundle Adjustment

Based on the computed relative orientations between single images we can build an approximate scene model in relative coordinates that serve as an estimate of input parameters entering into complex bundle adjustment. As initializing pair is selected a pair of images with the largest number of correspondences (inliers). The objective of the bundle adjustment is the reprojection error minimalization, when the corrections are assigned both to three dimensional points and parameters of outer or possibly inner orientation of single cameras.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} d(P_j(X_i), \ x_{\mathrm{ij}})^2 \tag{9}$$

Where $d(P_j(X_i), \ x_{\mathrm{ij}})^2$ marks the square of Euclidean distance between the predicted projection P_j of the point X_i in three dimensional global coordinate system and real projection x_{ij} to image j.

The base of SBA algorithm (Sparse Bundle Adjustment) is implementation of nonlinear Levenberg-Marquart algorithm where local function minimum is sought for point X projection in global coordinates into the image coordinates using a combination of Gauss-Newton method and the method of the steepest descent. If the input estimate of parameters is too far from the f function local minimum, the algorithm behavior is similar to the steepest descent method which guarantees at least slow convergence. In the case of moving towards local minimum Gauss-Newton method is used to guarantee fast convergence.

The main advantage of the SBA software package [7] is the usage of optimized memory structures for sparse matrices storing arising from the normal form equations. Without the use of sparse matrices the solution would be (when having hundreds of thousands unknown images) out of extreme memory and computation cost almost impossible.

7. Experimental Results

The input selection contained 25 images of Žinkovy castle facade (Fig. 5) shot by digital camera Olympus, the computations were performed on a standard PC with Intel Core 2 Duo processor. Algorithms were implemented in C/C++ language. Based on the results of the lens calibration the distorsions were neglected.

The original images of size 3648 x 2736 pixels were modified due memory cost of key points computation to 1204 x 903 size.

Figure 4: Input dataset: 25images, size 1204x903 pixels

Computation SIFT descriptors took 424 seconds, average 17 seconds per image and 485536 were detected. Key points pairing for 300 image combinations took 900 seconds. Finally 43052 three-dimensional points entered bundle adjustment.

The results of the bundle adjustment were the absolute orientations of images in a relative coordinate system, which was defined by the first pair of images. The coordinates of projection centers were also independently determined in Intergraph ISAT software using ground control points in geodetic system. To compare the accuracy of automatic orientation, transformation

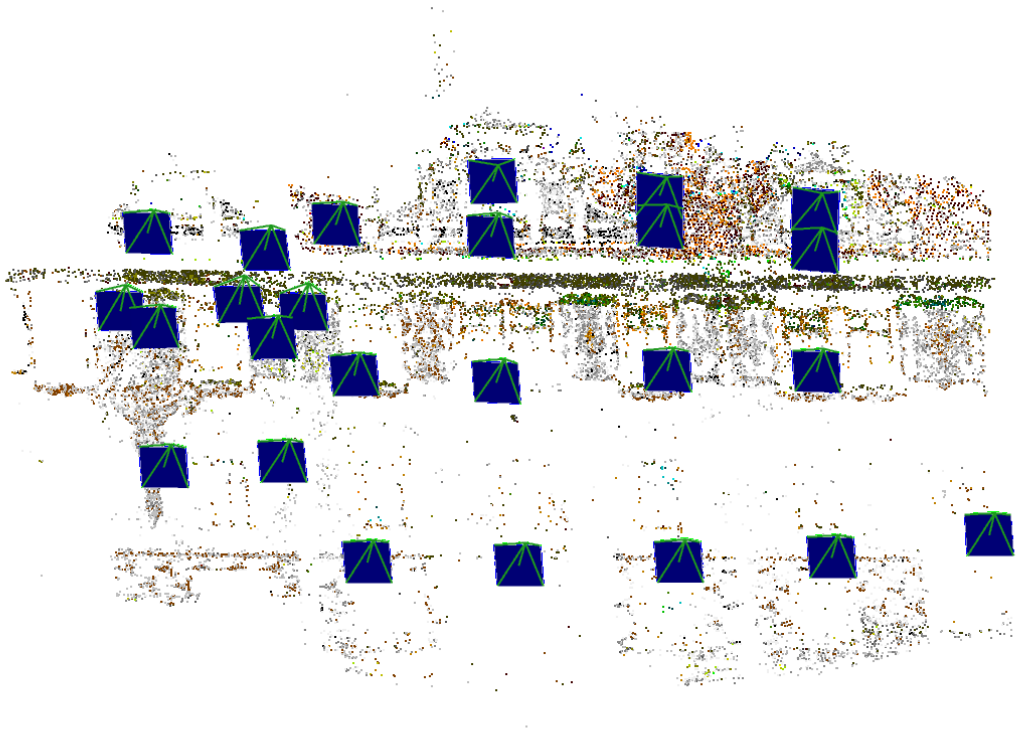

Figure 5: Reconstructed projection centers

key and coordinate differences between projection centers in both systems were also computed.

Image	dX	dY	dZ
P6027864	0.28	0.13	-0.09
P6027865	0.19	-0.01	0.03
P6027866	0.17	-0.01	0.08
P6027867	-0.24	-0.05	0.09
P6027869	0.59	0.09	-0.09
P6027870	0.56	-0.07	0.02
P6027871	1.23	0.00	0.44
P6027872	0.92	0.03	-0.01
P6027873	-0.62	-0.09	-0.26
P6027874	-0.57	-0.25	0.01
P6027875	-0.55	0.08	0.14
P6027876	-0.95	0.01	-0.15
P6027877	-0.41	0.02	-0.15
P6027878	-0.49	0.25	0.22
P6027879	-0.36	0.18	0.48

continued on next page

continued from previous page			
Image	dX	dY	dZ
P6027880	-0.35	-0.30	-1.02
P6027881	-0.14	-0.03	-0.03
P6027882	-0.25	-0.08	0.17
P6027883	-0.41	0.04	0.30
P6027884	-0.45	-0.02	-0.53
P6027885	0.19	0.05	0.04
P6027886	0.24	0.05	0.30
P6027887	0.42	0.11	0.55
P6027888	0.41	0.01	-0.51
P6027889	0.57	-0.12	-0.04

Table 1: Differences between coordinates of projection centers computed in Intergraph ISAT and coordinates obtained by automatic process

8. Summary

By above mentioned methods the absolute orientations of projection centers in a relative coordinate system using image descriptors were fully automatically determined. Parameters obtained in this way can be easily transformed into geodetic system using ground control points and ideally these parameters can directly serve as input to other tasks such as generating of orthogonalized mosaic of facade images. In less ideal case we can obtain the input estimates of parameters for another advanced calculations and decrease time and work difficulty of the absolute orientation procedure.

Inaccuracies can be attributed to both numerical instability of algorithms and the insufficient calibration of the lens.

Other options for achieving better results:

1. Using of GPU for reducing time complexity
2. Selection of key points only in a particular part of images (Gruber areas).

References

[1] Lowe, D.: Sift demo implementation. http://www.cs.ubc.ca/~lowe/keypoints/

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

[3] Harris, C. G.; Stephens, M. J.: A combined corner and edge detector. Proceeding Fourth Alvey Vision Conference, 1988, p. 147 - 151.

[4] Changchang Wu; SIFT on GPU. University of North Carolina at Chapel Hill, http://www.cs.unc.edu/~ccwu/siftgpu/

[5] Hartley, R.I.; Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, June 2000.

[6] Hartley, R.I.: An investigation of the Essential Matrix. 1993. http://users.rsise.anu.edu.au/~hartley/Papers/Q/Q.pdf

[7] Manolis I. A. Lourakis; Antonis A. Argyros, SBA: A software package for generic sparse bundle adjustment, ACM Transactions on Mathematical Software (TOMS), v.36 n.1, March 2009, p.1-30,

[8] Brown, M.; Szeliski, R.; Winder, S.: Multi-image Matching Using Multi-scale Oriented Patches. Proc. Int. Conf. on Computer Vision and Pattern Recognition, San Diego, 2005, p.510-517.

[9] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment -a modern synthesis. Vision Algorithms: Theory and Practice, pages 298–372, 1999.

[10] Manolis I. A. Lourakis: A brief description of the Levenberg-Marquardt algorithm by levmar, http://www.ics.forth.gr/lourakis/levmar/levmar.pdf, July 2004

[11] Hartley R.I.: In defence of the 8-point algorithm, ICCV, pp.1064, Fifth International Conference on Computer Vision (ICCV'95), 1995