Omnidirectional Image Stabilization by Computing Camera Trajectory

Akihiko Torii, Michal Havlena, and Tomáš Pajdla

Center for Machine Perception, Department of Cybernetics,
Faculty of Electrical Engineering, Czech Technical University in Prague,
Karlovo náměstí 13, 121 35 Prague 2, Czech Republic
{torii,havlem1,pajdla}@cmp.felk.cvut.cz
http://cmp.felk.cvut.cz

Abstract. In this paper we present a pipeline for camera pose and trajectory estimation, and image stabilization and rectification for dense as well as wide baseline omnidirectional images. The input is a set of images taken by a single hand-held camera. The output is a set of stabilized and rectified images augmented by the computed camera 3D trajectory and reconstruction of feature points facilitating visual object recognition. The paper generalizes previous works on camera trajectory estimation done on perspective images to omnidirectional images and introduces a new technique for omnidirectional image rectification that is suited for recognizing people and cars in images. The performance of the pipeline is demonstrated on a real image sequence acquired in urban as well as natural environments.

Keywords: Structure from Motion, Omnidirectional Vision.

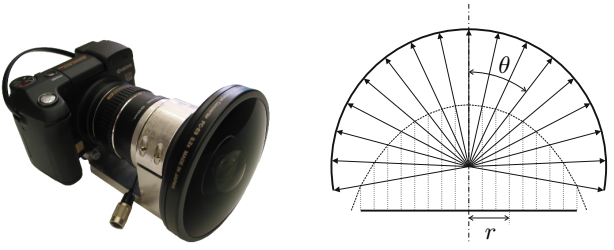
1 Introduction

Image stabilization and camera trajectory estimation plays an important role in 3D reconstruction [1,2,3], self localization [4], and reducing the number of false alarms in detection and recognition of pedestrians, cars, and other objects in video sequences [5,6,7,8].

Most of the approaches to camera pose and trajectory computation [9,1,2] work with classical perspective cameras because of the simplicity of their projection models and ease of their calibration. However, perspective cameras offer only a limited field of view. Occlusions and sharp camera turns may cause that consecutive frames look completely different when the baseline becomes longer. This makes the image feature matching very difficult (or impossible) and the camera trajectory estimation fails under such conditions. These problems can be avoided if omnidirectional cameras, e.g. a fish-eye lens convertor [10], are used. Large field of view also facilitates the analysis of activities happening in the scene since moving objects can be tracked for longer time periods [7].

In this paper we present a pipeline for camera pose and trajectory estimation, and image stabilization and rectification for dense as well as wide baseline omnidirectional images. The input is a set of images taken by a single hand-held

T. Wada, F. Huang, and S. Lin (Eds.): PSIVT 2009, LNCS 5414, pp. 71–82, 2009.



(a) (b)

Fig. 1. (a) Kyocera Finecam M410R camera and Nikon FC-E9 fish-eye lens convertor. (b) The equi-angular projection model. The angle θ between the casted ray of a 3D point and the optical axis can be computed from the radius r of a circle in the image circular view field.

camera. The output is a set of stabilized and rectified images augmented by the computed camera 3D trajectory and reconstruction of feature points facilitating visual object recognition. We describe the essential issues for a reliable camera trajectory estimation, i.e. the choice of the camera and its geometric projection model, camera calibration, image feature detection and description, robust 3D structure computation, and a suitable omnidirectional image rectification.

The setup used in this work was a combination of Nikon FC-E9, mounted via a mechanical adaptor, and a Kyocera Finecam M410R digital camera (see Figure 1(a)). Nikon FC-E9 is a megapixel omnidirectional add-on convertor with 180° view angle which provides images of photographic quality. Kyocera Finecam M410R delivers 2272×1704 images at 3 frames per second. The resulting combination yields a circular view of diameter 1600 pixels in the image.

2 The Pipeline

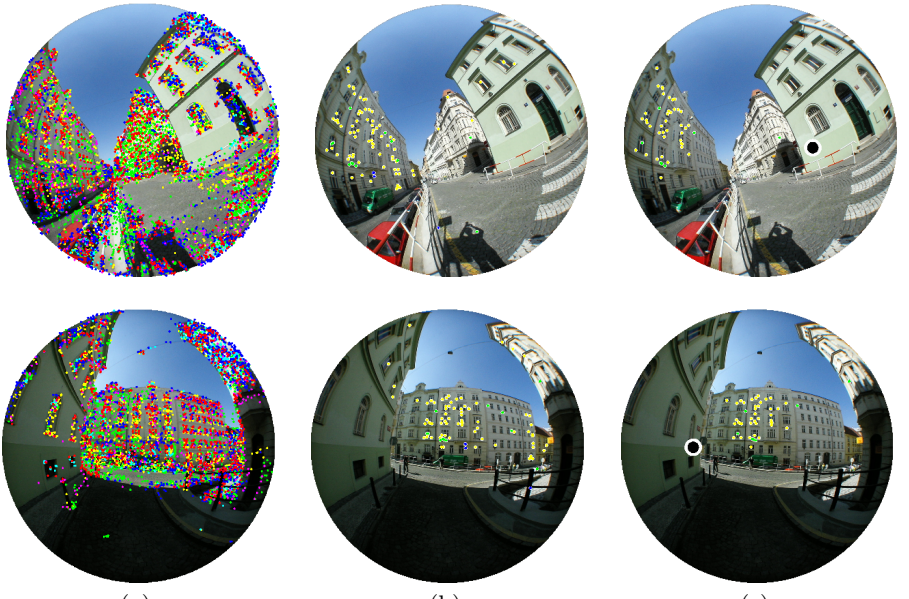
Next we shall describe our pipeline.

2.1 Camera Calibration

The calibration of omnidirectional cameras is non-trivial and is crucial for achieving good accuracy of the resulting 3D reconstruction. Our omnidirectional camera is calibrated off-line using the state-of-the-art technique [11] and Mičušík's two-parameter model [10], that links the radius of the image point r to the angle θ of its corresponding rays w.r.t. the optical axis (see Figure 1(b)) as

$$\theta = \frac{ar}{1 + br^2}\,. \tag{1}$$

After a successful calibration, we know the correspondence of the image points to the 3D optical rays in the coordinate system of the camera. The following steps aim at finding the transformation between the camera and the world coordinate systems, i.e. the pose of the camera in the 3D world, using 2D image matches.



(a) (b) (c)

Fig. 2. Example of the wide baseline image matching. The colors of the dots correspond to the detectors (yellow) MSER-Intensity+, (green) MSER-Intensity−, (cyan) MSER-Saturation+, (blue) MSER-Saturation−, (magenta) Harris Affine, and (red) Hessian Affine. (a) All detected features. (b) Tentative matches constructed by selecting pairs of features which have the mutually closest similarity distance. (c) The epipole (black circle) computed by maximizing the supports. Note that the scene dominated by a single plane does not induce the degeneracy of computing epipolar geometry due to solving the 5-point minimal relative orientation problem.

2.2 Detecting Features and Constructing Tentative Matches

For computing 3D structure, we construct a set of tentative matches detecting different affine covariant feature regions including MSER [12], Harris Affine, and Hessian Affine [13] in acquired images. These features are alternative to popular SIFT features [14] and work comparably in our situation. Parameters of the detectors are chosen to limit the number of regions to 1-2 thousands per image. The detected regions are assigned local affine frames (LAF) [15] and transformed into standard positions w.r.t. their LAFs. Discrete Cosine Descriptors [16] are computed for each region in the standard position. Finally, mutual distances of all regions in one image and all regions in the other image are computed as the Euclidean distances of their descriptors and tentative matches are constructed by selecting the mutually closest pairs. Figures 2(a) and (b) show an example of the feature detection and matching for a pair of wide baseline images.

Unlike the methods using short baseline images [2], simpler image features which are not affine covariant cannot be used because the view point can change a lot between consecutive frames. Furthermore, feature matching has to be

(a) (b) (c)

Fig. 3. Examples of pairs of images (two consecutive frames) from top to bottom in the CITY WALK sequence. Blue circles represent the epipoles and yellow dots are the matches supporting this epipolar geometry. Red dots are the matches feasibly reconstructed as 3D points. (a) contains multiple moving objects and large camera rotation. (b) contains large camera rotation and tentative matches on bushes. (c) contains tentative matches mostly constructed on a complex natural scene.

performed on the whole frame because no assumptions on the proximity of the consecutive projections can be made for wide baseline images. This is making the feature detection, description, and matching much more time-consuming than it is for short baseline images and limits the usage to low frame rate sequences when operating in real-time.

2.3 Epipolar Geometry Computation of Pairs of Consecutive Images

Robust 3D structure can be computed by RANSAC [17] which searches for the largest subset of the set of tentative matches which is, within a predefined threshold ε, consistent with an epipolar geometry [3]. We use ordered sampling as suggested in [18] to draw 5-tuples from the list of tentative matches ordered ascendingly by the distance of their descriptors which may help to reduce the number of samples in RANSAC. From each 5-tuple, relative orientation is computed by solving the 5-point minimal relative orientation problem for calibrated cameras [19,20]. Figure 2(c) shows the result of computing the epipolar geometry for a pair of wide baseline images.

Often, there are more models which are supported by a large number of matches. Thus the chance that the correct model, even if it has the largest support, will be found by running a single RANSAC is small. Work [21] suggested to generate models by randomized sampling as in RANSAC but to use soft (kernel) voting for a parameter instead of looking for the maximal support. The best model is then selected as the one with the parameter closest to the maximum in the accumulator space. In our case, we vote in a two-dimensional accumulator for the estimated camera motion direction. However, unlike in [21], we do not cast votes directly by each sampled epipolar geometry but by the best epipolar geometries recovered by ordered sampling of RANSAC [18]. With our technique, we could go up to the 98.5 % contamination of mismatches with comparable effort as simple RANSAC does for the contamination by 84 %. Finally, the relative camera orientation with the motion direction closest to the maximum in the voting space is selected. Figure 3 shows difficult examples of pairs of images to find the correct epipolar geometry.

2.4 Chaining Camera Poses for Sequence of Images

Camera poses in a canonical coordinate system are recovered by chaining the epipolar geometries of pairs of consecutive images in a sequence. For the essential matrix $\mathtt{E}_{ij}$ between frames i and $j = i + 1$, the essential matrix $\mathtt{E}_{ij}$ can be decomposed into $\mathtt{E}_{ij} = [\mathbf{e}_{ij}]_\times \mathtt{R}_{ij}$. Although there exist four possible decompositions, the right decomposition can be selected to reconstruct all points in front of both cameras [3, $p260$]. Having the normalized camera matrices [3] of the i-th frame $\mathtt{P}_i = [\mathtt{R}_i \,|\, \mathbf{T}_i]$, the normalized camera matrix $\mathtt{P}_j$ can be computed by

$$\mathtt{P}_j = [\mathtt{R}_{ij}\mathtt{R}_i \,|\, \mathtt{R}_{ij}\mathbf{T}_i + \alpha \mathbf{e}_{ij}] \tag{2}$$

where α is the scale of the translation in the canonical coordinate system. The scale α can be computed by any 3D point seen in at least three consecutive frames. The best scale is selected to maximize the number of points that pass the feasibility test of L_1- or L_∞- triangulation [22,23], i.e., the intersection of pixel-cone rays test. In the final step, we applied the sparse bundle adjustment [24] to refine the structure.

2.5 Image Stabilization Using Camera Pose and Trajectory

The recovered camera pose and trajectory can be used to rectify the original images to the stabilized images. If there exists no assumption on the camera motion in a sequence, the simplest way of stabilization is to rectify images w.r.t. the gravity vector in the coordinate system of the first camera and all other images will then be aligned with the first one. This can be achieved by taking the first image with care. When a sequence is captured by walking or driving on the roads, it is possible to stabilize the images w.r.t. the ground plane. For a gravity direction $\mathbf{g}$ and a motion direction $\mathbf{t}$, we compute the normal vector of the ground plane

$$\mathbf{d} = \frac{\mathbf{t} \times (\mathbf{g} \times \mathbf{t})}{|\mathbf{t} \times (\mathbf{g} \times \mathbf{t})|}. \tag{3}$$

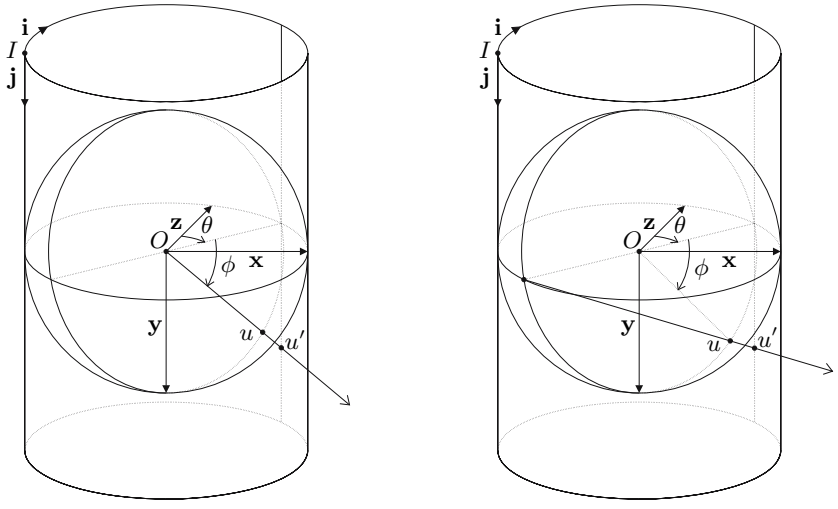


Fig. 4. Projection of a pixel u' of the resulting cylindrical image onto a pixel u on a unit sphere. Column index u'_i is transformed into angle θ and row index u'_j into angle ϕ. These angles are then transformed into the coordinates u_x, u_y, and u_z of a unit vector. Left: Central cylindrical projection. Right: Non-central cylindrical projection.

We construct the stabilization and rectification transform $\mathtt{R}_s$ for the image point represented as a 3D unit vector such that $\mathtt{R}_s = [\,\mathbf{a}, \mathbf{d}, \mathbf{b}\,]$ where $\mathbf{a} = (0,0,1)^\top \times \mathbf{d} \,/\, \left|(0,0,1)^\top \times \mathbf{d}\right|$ and $\mathbf{b} = \mathbf{a} \times \mathbf{d} \,/\, |\mathbf{a} \times \mathbf{d}|$. This formulation is sufficient because the roads usually go up and down to the view direction.

2.6 Central and Non-central Cylindrical Image Generation

Using the camera trajectories, it is possible to construct perspective cutouts rectified w.r.t. the ground plane and an arbitrary object recognition routine designed to work with images acquired by perspective cameras can be used without any further modifications. For instance, object recognition methods could benefit from image stabilization (e.g. [6]) which is usually trained on perspective images. On the other hand, as a true perspective image is able to cover only a small part of the available omnidirectional view-field, we propose to use cylindrical images which can cover a much larger part of it.

Knowing the camera and lens calibration, we represent our omnidirectional image as a part of a surface of a unit sphere, each pixel is represented by a unit vector. It is straightforward to project such surface on a surface of a unit cylinder surrounding the sphere using rays passing through the center of the sphere (see Figure 4). We transform the column index u'_i of a pixel of the resulting cylindrical image into angle θ and the row index u'_j into angle ϕ using

$$\theta = \left(u'_i - \frac{I_W}{2}\right)\frac{\theta_{max}}{I_W}, \quad \phi = \arctan\left(\left(u'_j - \frac{I_H}{2}\right)\frac{\theta_{max}}{I_W}\right), \tag{4}$$

(a) (b)

(c) (d)

Fig. 5. (a) Original omnidirectional image (equiangular). (b) Central cylindrical projection. (c) Perspective projection. (d) Non-central cylindrical projection. Note there is a large deformation at the borders of the perspective image and at the top and bottom borders of the central cylindrical image. The borders of the non-central cylindrical image are less deformed.

where I_W and I_H are the dimensions of the resulting image and θ_{max} is the horizontal field of view of the omnidirectional camera. These angles are then transformed into the coordinates u_x, u_y, and u_z of a unit vector as

$$u_x = \cos\phi \sin\theta, \quad u_y = \sin\phi, \quad u_z = \cos\phi\cos\theta\,. \tag{5}$$

Note that the top and bottom of the rectified image look rather deformed for the vertical field of view reaching π if the height of the resulting image I_H is being increased (see Figure 5). We propose to use a generalization of the stereographic projection which we call a non-central cylindrical projection. Projecting rays do not pass through the center of the sphere but are cast from points on its equator. The desired point is the intersection of the plane determined by the column of the resulting image and the center of the sphere with the equator of the sphere. The equation for angle θ remains the same but angle ϕ is now computed using

$$\phi = 2\arctan\left(\frac{\left(u'_j - \frac{I_H}{2}\right)\frac{\theta_{max}}{I_W}}{2}\right). \tag{6}$$

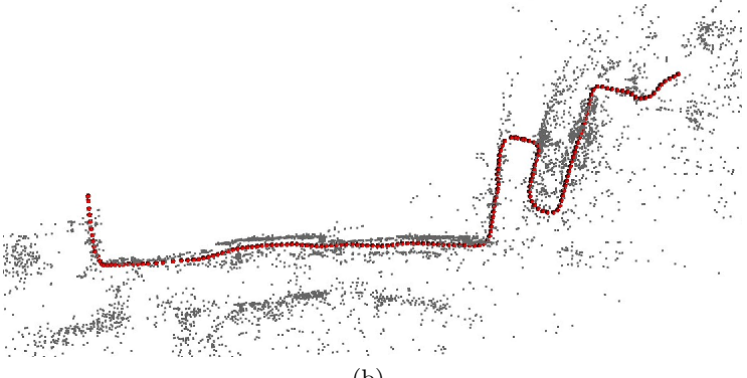
When generating the images, bilinear interpolation is used to suppress the artifacts caused by image rescaling.

3 Experimental Results

The experiment with real data demonstrates the use of proposed image stabilization method. Two image sequences of a city scene captured by a single hand-held fish-eye lens camera are used as our input sequences.

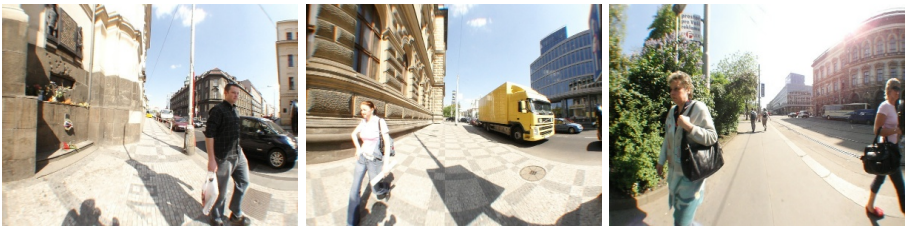
The CITY WALK sequence is 190 frames long and the distance between consecutive frames is 1-3 meters. This sequence is challenging for recovering the camera trajectory due to sharp turns, objects moving in the scene, and natural complex environment. The benefit of wide field of view can be seen in Figure 3. The camera motions are reasonably recovered by using the features detected from stational rigid objects. Figure 6(b) shows the camera positions and the world

(a)



(b)

Fig. 6. Camera trajectory of the CITY WALK sequence. (a) A bird's eye view of the city area used for the acquisition of our test sequence. The trajectory is drawn with a white line. (b) The bird's eye view of the resulting 3D model view. Red dots represent the camera positions recovered by our proposed method. Small gray dots represent the reconstructed world 3D points.



(a) Central projection

(b) Non-central projection

Fig. 7. Results of image transformations of frame 67 in the CITY WALK sequence. The images are stabilized w.r.t. the ground plane and panoramic images transformed by (a) central cylindrical projection and (b) non-central cylindrical projection. Note the pedestrians are less deformed on the non-central cylindrical projection while convening larger field of view than the central one.

3D points reconstructed by our structure from motion. The reconstruction is comparable to the walking trajectory shown in Figure 6(a). Since the sequence is captured walking along the planar street, all the images are stabilized using the recovered camera pose and trajectory w.r.t. the ground plane. Figure 7 shows the images generated by using central and non-central cylindrical projections. It can be seen that the non-central cylindrical projection in Figure 7(b) successfully suppresses the deformation at the top and bottom and makes people standing close to the camera looking much more natural.

The FREE MOTION sequence is 187 frames long and the distance between consecutive frames is 0.3-2 meters. This sequence is also challenging for recovering the camera pose and trajectory due to the large view changes by camera rotation and translation. Figure 8(a) shows several frames of the original images in the FREE MOTION sequence. Figure 8(b) shows the panoramic images generated by the non-central cylindrical projection. Since the motion is completely irrelevant w.r.t. the ground plane, all images are stabilized w.r.t. the gravity vector in the coordinate system of the first camera. Figure 8(c) shows the panoramic images stabilized using the recovered camera pose and trajectory. It can be seen clearly from this result that the large image rotation is successfully canceled using the recovered camera pose and trajectory.

(a) (b) (c)

Fig. 8. Results of our image stabilization and transformation in the FREE MOTION sequence. (a) Original images. (b) Non-stabilized images. (c) Stabilized images w.r.t. the gravity vector in the first camera coordinates. The rotation is successfully canceled and all images are stabilized using the recovered camera pose and trajectory.

4 Conclusions

The pipeline for camera pose and trajectory estimation, and image stabilization and rectification for an image sequence acquired by a single omnidirectional camera is presented. The experiments demonstrated that the robust camera pose and trajectory estimation based on epipolar geometry is useful to stabilize the image sequence. Furthermore, the non-central cylindrical projection can generate perspective-projection-like images while preserving a large field of view. The stabilized images can be instantly used as the preprocess for the recognition techniques [6,7] that assume ground plane positions and codebooks trained on perspective images.

Acknowledgments

The authors were supported by EC project FP6-IST-027787 DIRAC, FP7-218814 ProVisG, and by Czech Government under the research program MSM-684 0770038. Any opinions expressed in this paper do not necessarily reflect the views of the European Community. The Community is not liable for any use that may be made of the information contained herein. Finally, we would like to thank Přemysl Volf for fruitful discussions.

References

1. Akbarzadeh, A., Frahm, J.M., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Merrell, P., Phelps, M., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H., Nistér, D., Pollefeys, M.: Towards urban 3d reconstruction from video. In: 3DPVT (May 2006) (invited paper)
2. Cornelis, N., Cornelis, K., Van Gool, L.: Fast compact city modeling for navigation pre-visualization. In: CVPR 2006, pp. II:1339–II:1344 (2006)
3. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2003)
4. Goedemé, T., Nuttin, M., Tuytelaars, T., Van Gool, L.: Omnidirectional vision based topological navigation. IJCV 74(3), 219–236 (2007)
5. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: CVPR, vol. 2, pp. 2137–2144 (June 2006)
6. Leibe, B., Cornelis, N., Cornelis, K., Van Gool, L.: Dynamic 3d scene analysis from a moving vehicle. In: CVPR 2007, Minneapolis, MN, USA (2007)
7. Leibe, B., Schindler, K., Van Gool, L.: Coupled detection and trajectory estimation for multi-object tracking. In: ICCV 2007 (2007)
8. Torii, A., Havlena, M., Pajdla, T., Leibe, B.: Measuring camera translation by the dominant apical angle. In: CVPR 2008, Anchorage, AK, USA (2008)
9. 2d3 Boujou (2001), `http://www.boujou.com`
10. Mičušík, B., Pajdla, T.: Structure from motion with wide circular field of view cameras. IEEE Trans. PAMI 28(7), 1135–1149 (2006)
11. Bakstein, H., Pajdla, T.: Panoramic mosaicing with a 180° field of view lens. In: Proc. IEEE Workshop on Omnidirectional Vision, pp. 60–67 (2002)

12. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing 22(10), 761–767 (2004)
13. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. IJCV 65(1-2), 43–72 (2005)
14. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
15. Obdržálek, Š., Matas, J.: Object recognition using local affine frames on distinguished regions. In: BMVC 2002, London, UK, vol. 1, pp. 113–122 (2002)
16. Obdržálek, Š., Matas, J.: Image retrieval using local compact DCT-based representation. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 490–497. Springer, Heidelberg (2003)
17. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comm. ACM 24(6), 381–395 (1981)
18. Chum, O., Matas, J.: Matching with PROSAC - progressive sample consensus. In: CVPR 2005, Los Alamitos, USA, vol. 1, pp. 220–226 (2005)
19. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Trans. PAMI 26(6), 756–770 (2004)
20. Stewénius, H.: Gröbner Basis Methods for Minimal Problems in Computer Vision. PhD thesis, Centre for Mathematical Sciences LTH, Lund University, Sweden (2005)
21. Li, H., Hartley, R.: A non-iterative method for correcting lens distortion from nine point correspondences. In: OMNIVIS 2005 (2005)
22. Kahl, F.: Multiple view geometry and the L-infinity norm. In: ICCV (2005)
23. Ke, Q., Kanade, T.: Quasiconvex optimization for robust geometric reconstruction. IEEE Trans. PAMI 29(10), 1834–1847 (2007)
24. Lourakis, M., Argyros, A.: The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical Report 340, Institute of Computer Science - FORTH, Heraklion, Crete, Greece (August 2004), `http://www.ics.forth.gr/~lourakis/sba`