Klaus Häming; Gabriele Peters

The structure-from-motion reconstruction pipeline – a survey with focus on short image sequences

Kybernetika, Vol. 46 (2010), No. 5, 926--937

Persistent URL: http://dml.cz/dmlcz/141400

THE STRUCTURE-FROM-MOTION RECONSTRUCTION PIPELINE – A SURVEY WITH FOCUS ON SHORT IMAGE SEQUENCES

Klaus Häming and Gabriele Peters

The problem addressed in this paper is the reconstruction of an object in the form of a realistically textured 3D model from images taken with an uncalibrated camera. We especially focus on reconstructions from *short* image sequences. By means of a description of an easy to use system, which is able to accomplish this in a fast and reliable way, we give a survey of all steps of the reconstruction pipeline. For the purpose of developing a coherent reconstruction system it is necessary to integrate a number of different techniques such as feature detection, algorithms of the RANSAC-family, and methods for auto-calibration. We describe and review recent developments of distinct strands of these techniques. While developing our system the necessity of improvements of several steps of the state-of-the-art reconstruction pipeline emerged. Two of these innovations are introduced in detail in this paper: an advanced SIFT-based feature detector and a two-stage RANSAC process facilitating a faster selection of relevant object points. In addition, we give a recommendation regarding auto-calibration for short image sequences.

Keywords: structure from motion, feature detection, RANSAC, auto-calibration

Classification: 68U10, 68U05

1. INTRODUCTION

Creating a 3D model of our world simply by taking images of it is a fascinating idea that has inspired many researchers. Though there are attempts to create such a model from one image alone [36, 37], better results are usually achieved by using more images, establishing correspondences, and then triangulating 3D points. For the triangulation the relative positioning of the cameras and their technical properties, i. e., their *extrinsic* and *intrinsic* parameters, have to be known. Generally, there are two approaches to gain this knowledge. The first is to measure the parameters of the used camera(s) as exactly as possible before attempting to reconstruct anything. This procedure is called calibration [45, 49]. On the one hand, a calibrated camera simplifies the reconstruction process substantially, but on the other hand it imposes inflexibility since the setup must not be altered.

The other approach tries to calculate the cameras along with the 3D points, only relying on established correspondences between the observed images. In many

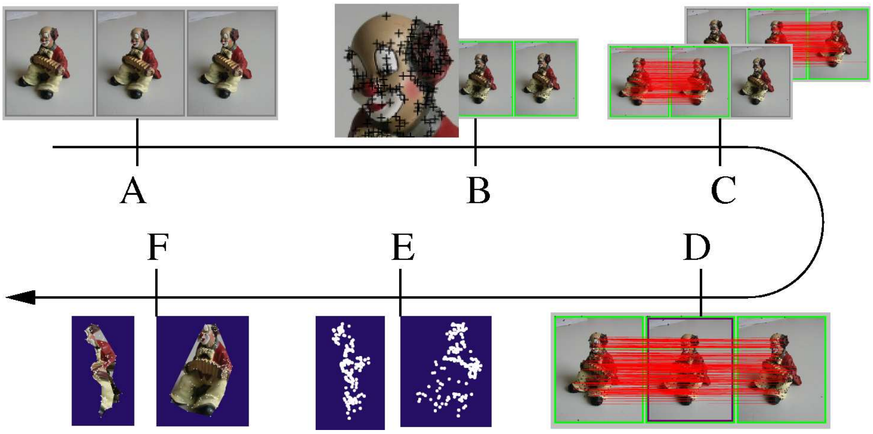

Fig. 1. The reconstruction pipeline. A: Three input images; B: Feature detection; C: Feature matching D: Filtering; E: Metric reconstruction, visualized as point cloud; F: Final object reconstruction, visualized as textured model.

publications these systems and their improvements are covered [3, 14, 15, 21, 22, 23, 30, 33, 34, 38, 42, 46]. From these, [46] gives a compact yet accessible overview covering a complete reconstruction system.

Compared to the calibrated approach, this projective approach takes more computational effort and is less accurate. In addition, it needs a subsequent autocalibration step to remove the projective distortion. But besides of these disadvantages it features greater flexibility and ease of use. Following this approach we have implemented a system [29] that has proven its accessibility for people unversed in handling IT devices.

One of the challenges that arise from *short* image sequences and that does not arise from *long* image sequences, is, for example, given by the fact that the 3d shape of an object has to be recovered from few information on the object only, i. e., from few images only.

The reconstruction pipeline is presented by means of an example, i. e., by means of our own development which includes two minor contributions - one concerning feature detection, the other concerning filtering of matches. We have included these contributions because they are tailor-made just for the special application area of short image sequences. These contributions induce a higher robustness (the contribution to feature detection) and speed (the contribution to filtering of matches). This leads to the fact that our system accomplishes its task reliably and instantly.

2. THE RECONSTRUCTION PIPELINE

The acquisition system we describe comprises a few subsystems which follow the common reconstruction pipeline (Fig. 1). In this section we give a short overview of this reconstruction process.

Generally, the reconstruction starts by taking a number of images of the same object. Therefore, single features of that object are expected to be present in more than one image. For example, a feature at position x_1 in the first image I_1 may be detected at positions x_2 and x_3 in the second and third images I_2 and I_3, respectively. Such a tuple of corresponding features is called correspondence. If the cameras and their positions in space are known it is possible to reconstruct 3D points of the observed object directly. This can be achieved by intersecting the rays from the camera centers through the feature points of one particular correspondence. This technique is called triangulation. However, since we consider the case with unknown camera parameters, we use the images only. Though there are also approaches, which work without given correspondences [11], the information on corresponding points is sufficient to reconstruct points of the object's surface, because correct correspondences have to meet certain geometrical constraints. These constraints can be represented algebraically as multi-view tensors, namely the fundamental matrix F in the 2-view case and the trifocal tensor T in the 3-view case. In tensor notation, the constraints are:

$$x_1^i x_2^j F_{ij} = 0 \tag{1}$$

and

$$x_1^i x_2^j x_3^k \epsilon_{jqs} \epsilon_{krt} T_i^{qr} = 0_{st}. \tag{2}$$

Once a sufficient number of correspondences has been established F and T can be recovered. The sufficient number is seven in the case of F and six in the case of T.

One advantage of the tensor computation is the ability to compute generic cameras that create the same multi-view relation as the original cameras, which have been used to take the images. The difference between both of these sets of cameras is a projective transformation in projective 3-space P^3. This projective ambiguity of the reconstruction can be upgraded to differ from the observed scene by a similarity transformation only. This is done by a subsequent auto-calibration step. A reconstruction is said to differ from the original setup by a similarity, if that transformation comprises translation, rotation, and scaling only, which is obviously as close as one can get, because of the lack of an absolute coordinate system. Auto-calibration is described in more detail in Section 5.

The described approach depends on reliable correspondences. To establish them, first image points have to be identified, whose surrounding image patches allow for a robust recognition and localization throughout the image sequence. These image points are called features, and an algorithm to calculate them is called a feature detector (Fig. 1B). Each feature has a numerical description of the properties of its surrounding image patch. These descriptors are used in a subsequent matching step (Fig. 1C). The result of this step are candidates for correspondences only, because they contain a number of mismatches. Suitable alternatives for feature and descriptor computation are discussed in Section 3. To accelerate the set-up of feature correspondences by comparing feature descriptors, one can use a kd-tree [17] with best-bin-first optimization [4].

To eliminate the mismatches a filtration of good matches has to be performed in a subsequent filtering step (Fig. 1D). A hypothesize-and-verify framework based on the RANdom SAmple Consensus (RANSAC) of Fischler and Bolles [13] reliably provides for this. The details are presented in Section 4. The general idea of RANSAC is to take a minimal number of samples from the set of all correspondences and compute a compatible multi-view tensor. After this, the support for that geometry is determined by examining the whole population of correspondences and summing up an error measure. For our purposes the minimum number of correspondences needed to compute F and T is important.

The tensor computed from the correspondences can easily be used to add more correspondences by applying a guided matching step. During this step feature pairs that match have to comply with the epipolar geometry imposed by the tensor. This is achieved by restricting the area of candidate matches to a tight envelope around epipolar lines.

Finally, we get a reconstruction in form of a 3D point cloud (Fig. 1E), not yet in form of a surface. One approach to create a dense surface is to apply a disparity algorithm [5, 10, 18, 48] on a rectified image pair [19, 32]. Another approach is to densify the correspondences locally [23]. A simpler approach that works well especially for short image sequences computes the Delaunay Triangulation of the feature point locations. This triangulation can then be used to create triangles in 3D space by connecting those 3D points which share an edge in the triangulation (Fig. 1F). In comparison to the disparity map approach one advantage of this method consists in the fact that feature points are processed at their sub-pixel positions instead of discarding the sub-pixel information and proceeding with an integer disparity value only.

3. FEATURE DETECTION

Many feature detectors have been proposed in the past. For reconstruction a feature detector is necessary that is capable of computing feature positions with sub-pixel accuracy. This holds especially for small baselines between camera positions. In such a case, small displacements in the image often result in large errors in 3D-space as depicted in Figure 2.

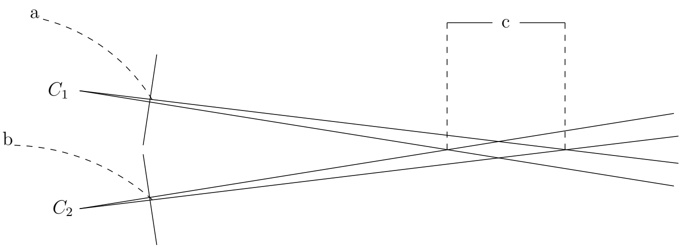

Fig. 2. Small errors in the image plane (a and b) can lead to large errors in 3D-space (c).

The feature point positions are also the main input against which the reconstruction is assessed. A measure for the accuracy of the reconstruction is the re-projection error

$$E_{\text{repro}} = \sum_{x} d^2(x, x'), \tag{3}$$

where x is the feature position as returned by the feature detector and x' the position of the image point to which the associated 3D point projects after reconstruction. The distance measure d is the Euclidean Distance of the inhomogeneous point locations. The sum is taken over all images.

We will now introduce those feature detectors that proved well in our reconstruction system in terms of performance and accuracy. The most prominent and widely successful one is the SIFT feature detector [26]. It uses the maxima detected in a Difference-of-Gaussians (DOG) pyramid as the returned features. The Differences-of-Gaussians are a close approximation of the Laplacian-of-Gaussian, which have been found to be very stable to viewpoint changes [28]. The feature descriptor of SIFT is calculated in several steps. The first step is to find a dominant gradient direction. This is used to make the descriptor invariant to rotations. In [26], the descriptor is rotated after its computation to fit that main orientation. Contrary to this, we found an affine mapping of the image patch that surrounds the feature location preceding to the descriptor computation beneficial. This affine transformation combines the scale and orientation information the feature detector provides. The result is a 16 × 16 image patch. The values in the grid are used to compute the 16 $(= 4 \times 4)$ gradient histograms, that comprise the descriptor.

Another widely accepted detector/descriptor combination is SURF [2], which tries to improve SIFT by replacing the DOG with a Hessian matrix based blob detector. Using integral images [47], this detector can be evaluated quite fast. Instead of the gradient histograms of SIFT, SURF uses sums of gradient components and sums of their absolute values. This leads to descriptors with a smaller dimensionality which in turn eases the feature matching. The improved speed, however, comes with the drawback of less accurate feature positions.

A little-known feature detector uses a Canny edge detector [6] combined with a local Hough transform [40]. The motivation for this combination is the fact, that image homographies map lines to lines and therefore intersections of lines should constitute stable features. These intersections are the returned feature points. This detector finds less features than SIFT, but we found their localization comparably accurate.

Also very successful and widely used is the Harris detector [20]. This detector uses the auto-correlation matrix (also known as second moment matrix) to detect corners:

$$\mu(p) = \left(\begin{array}{cc} L_x^2(p) & L_x L_y(p) \\ L_x L_y(p) & L_y^2(p) \end{array} \right) , \tag{4}$$

where L_a is the derivative in a direction and p an image point. The cornerness is computed as

$$\det(\mu(p) - 0.04\,\mathrm{trace}^2(\mu(p)))\,. \tag{5}$$

Because the observable features depend on the distance between the object and the camera the notion of scale space has been introduced [24], which embeds the feature detection into a framework of repeated image smoothing. SIFT and SURF regard this by using repeated downsampling and increasing of the filter size, respectively. In either case, a detector has to be adapted in order to make the feature strengths comparable across scales. The adaptation necessary for the Harris detector can be found in [28]. A fast way to compute a scale space representation has been introduced in [25].

There is another reason why a Harris-based detector may be beneficial. It provides the option to compute affinely invariant image patches as in [1]. This technique also requires the computation of the auto-correlation matrix, which the Harris detector provides as a side effect.

For the purpose of scene or object reconstruction the images of a recorded sequence are usually quite similar to each other. Therefore it seems to be obvious to use a tracking algorithm such as the Lukas-Kanade tracker [27, 41]. Tracking algorithms provide a large number of features with good accuracy for small baselines, but they become increasingly unthrifty with increasing camera baselines.

To sum it up, we recommend the feature detector/descriptor combination that creates correspondences from which the largest number survives the filtering described in the next section. In our set-up, this was our implementation of the scale-space Harris feature detector in combination with the SIFT descriptor.

4. FILTERING OF GOOD MATCHES

The random sample consensus [13] has received many improvements such as MSAC and MLESAC [43], Locally Optimized RANSAC (LO-RANSAC) [8, 9], PROgressive ranSAC (PROSAC) [7], and more recently QDEGSAC [16] to detect quasi degenerate cases.

MSAC improves RANSAC by refining the function that assigns a cost to each match from

$$c(e^2) = \begin{cases} 0 & e^2 < T^2 \\ 1 & e^2 >= T^2 \end{cases} \tag{6}$$

to

$$c(e^2) = \begin{cases} e^2 & e^2 < T^2 \\ T^2 & e^2 >= T^2 \end{cases} \quad . \tag{7}$$

This allows a decision based on an error value rather then a number of inliers and is strongly recommended. MLESAC models the error as a mixture of Gaussian and uniform distribution. It uses the EM algorithm [12] to achieve a maximum likelihood estimate.

LO-RANSAC improves the estimated model by applying a model refinement using a larger set of samples. This larger set consists of inliers only, therefore a least squares approach can be used for refinement. The set of inliers that belongs to the refined estimation is then computed from the whole population.

PROSAC reduces the running time of RANSAC by preferring more promising samples. In the case of correspondences, this is done by sorting them according to their matching cost, which has been computed in the feature matching step. Assuming, that a greater matching cost implies a smaller probability of being a good match, a growth function is applied to progressively take more samples into account.

QDEGSAC employs a hierarchical RANSAC to test for the number of constraints the samples obey. QDEGSAC, however, needs a RANSAC run to fail to determine the number of model parameters which is time consuming.

These RANSAC derivatives can be combined. We recommend to combine LO-RANSAC with PROSAC. PROSAC is straight forward to implement, but LO-RANSAC rises the question of how to implement the local optimization. In [8] the authors suggest the incorporation of an inner RANSAC loop while iteratively tightening the inlier threshold. In our tests however, this leads to less inliers without improving the results. For this reason we rather propose the following algorithm:

- Take a minimum number of samples in an outer loop and estimate the multi-view tensor. The samples are taken according to their matching quality as in PROSAC.

- After each improvement in the error value, take the inliers of the outer loop and perform a traditional RANSAC on them inside an inner loop. This time however, take twice the number of features as in the minimum case to robustify the least squares solution. The inliers are again computed on the whole population of correspondences.

This scheme is performed in a two-stage process, in which we first estimate fundamental matrices and subsequently trifocal tensors. The fundamental matrix is estimated using the standard 7-point algorithm for the minimum case and the linear algorithm for the inner loop [21]. Then, the trifocal tensor is estimated using the algorithm from [39] for the minimum case and the algebraic minimization algorithm in the inner loop [21].

The novelty proposed in this section is the simplified inner loop of the LO-RANSAC process. The method performed best in our experiments and is simpler than the one proposed in [8], and thus more time-saving, because it does not involve iterative tightening of the threshold.

5. AUTO-CALIBRATION

To transform the projective reconstruction into a metric reconstruction we follow the approach described in [31, 44] and use the absolute quadric and its dual. An alternative approach is described in [35], which uses the absolute quadratic complex. The absolute dual quadric Ω^* is a geometric entity with the useful property that it is *invariant* under similarity transformations. Let P be a camera matrix and K the matrix that represents the camera's intrinsic parameters. Then the following must hold for each camera:

$$KK^T \cong P\Omega^* P^T. \tag{8}$$

Because this "equation" only holds up to an arbitrary factor, we use "$\cong$" instead of "$=$". Generally, Ω^* is a 4-by-4 matrix. In the metric case it is a diagonal matrix with diagonal entries $(1, 1, 1, 0)$. We solve K and Ω^* simultaneously while imposing constraints on K. Decomposing Ω^* into HDH^T yields a transformation H, which can be applied to the cameras ($P \rightarrow PH$) to create the metric frame.

The widely used camera model is a 3×4 matrix $P = K[R|t]$, with K capturing the intrinsic parameters and R a rotation. The vector t encodes the camera center C as $t = -RC$. Let $(p_x, p_y)^T$ be the principal point, s the skew parameter, f the focal length, and α the aspect ratio. Then K has the form

$$K = \begin{bmatrix} f & s & p_x \\ 0 & \alpha f & p_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

Therefore the left hand side of (8) is a symmetric matrix of the form

$$KK^T = \begin{bmatrix} k_1 & k_2 & k_3 \\ k_2 & k_4 & k_5 \\ k_3 & k_5 & 1 \end{bmatrix} = \begin{bmatrix} f^2 + s^2 + p_x^2 & s\alpha f + p_x p_y & p_x \\ s\alpha f + p_x p_y & \alpha^2 + f^2 & p_y \\ p_x & p_y & 1 \end{bmatrix}. \tag{10}$$

To get a linear auto-calibration approach we impose the following constraints: $\alpha = 1$, $s = 0$, $p_x = 0$, and $p_y = 0$. This leads to a simplified KK^T:

$$KK^T = \begin{bmatrix} f^2 & 0 & 0 \\ 0 & f^2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{11}$$

which gives us 4 constraints on formula (8): $k_1 = k_4$, $k_2 = 0$, $k_3 = 0$, and $k_5 = 0$. These constraints lead to a linear system of equations in the parameters of Ω.

Unfortunately, this linear approach practically never succeeds for the examined short image sequences. Therefore we pursued a non-linear approach. Again, the skew parameter is fixed to $s = 0$ for all cameras. The other parameters however are assumed to be constant throughout the scene and constrained softly by punishing values further away from supposedly sane values. Given a range r for one parameter p and given $\hat{p}$ the supposed value of p, then its error is calculated as

$$E_p = \left| \frac{p - \hat{p}}{r} \right|^2. \tag{12}$$

The supposed values and ranges we used for the parameters are $f = 1.0 \pm 3.0$, $\alpha = 1.0 \pm 0.1$, $p_x = 0.0 \pm 0.1$, and $p_y = 0.0 \pm 0.1$.

Using these constraints, the equation to minimize is

$$\sum_i \| KK^T - P^i \Omega P^{iT} \|_F^2, \tag{13}$$

where i enumerates the cameras.

The non-linear approach leads to good results even for short sequences. One can argue, that it may be beneficial to start off with a linear algorithm and refine the result non-linearly afterwards. Though this has been suggested in [31], we have observed no improvement for short sequences. Neither the quality of calibration has been improved nor the number of iterations has been decreased. Therefore, using a linear auto-calibration technique for short sequences is considered useless.

6. CONCLUSION

We have presented state-of-the-art methods and techniques for a structure from motion system that is able to metrically reconstruct objects. The methods are well suited especially for short image sequences acquired with uncalibrated cameras. They work in a fast and reliable way. Special attention has been paid to two major parts of the reconstruction pipeline. Options to handle the feature detection step have been examined and a few of the most successful algorithms were introduced in a concise manner. We also addressed feature filtering by presenting the recent improvements of the RANSAC methods and proposed a particular combination of LO-RANSAC and PROSAC with an improved inner RANSAC loop. Finally, we gave a recommendation on how to approach the auto-calibration step in the case of short image sequences.

ACKNOWLEDGEMENT

This research was funded by the German Research Association (DFG), Grant PE 887/3-3.

(Received April 6, 2010)

REFERENCES

[1] A. Baumberg: Reliable feature matching across widely separated views. In: IEEE Conf. on Computer Vision and Pattern Recognition 2000, Vol. 01, pp. 1774–1781.

[2] H. Bay, T. Tuytelaars, and L. Van Gool: Surf: Speeded up robust features. In: 9th European Conference on Computer Vision, Graz 2006.

[3] P. A. Beardsley, P. H. S. Torr, and A. Zisserman: 3d model acquisition from extended image sequences. In: ECCV '96: Proc. 4th European Conference on Computer Vision-Volume II, Springer, London 1996, pp. 683–695.

[4] J. S. Beis and D. G. Lowe: Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In: Proc. IEEE Conf. Comp. Vision Patt. Recog 1997, pp. 1000–1006.

[5] S. Birchfield and C. Tomasi: Depth discontinuities by pixel-to-pixel stereo. Internat. J. Comput. Vision *3* (1999), 269–293.

[6] J. Canny: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. *8* (1986), 6, 679–698.

[7] O. Chum and J. Matas: Matching with PROSAC – progressive sample consensus. In: Proc. Conference on Computer Vision and Pattern Recognition (C. Schmid, S. Soatto, and C. Tomasi, eds.), Vol. 1, Los Alamitos 2005, IEEE Computer Society, pp. 220–226.

[8] O. Chum, J. Matas, and J. Kittler: Locally optimized ransac. In: DAGM-Symposium 2003, pp. 236–243.

[9] O. Chum, J. Matas, and Š. Obdržálek: Enhancing RANSAC by generalized model optimization. In: Proc. Asian Conference on Computer Vision (ACCV) (K.-S. Hong and Z. Zhang, eds.), Vol. 2, Seoul 2004, Asian Federation of Computer Vision Societies, pp. 812–817.

[10] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs: A maximum likelihood stereo algorithm. Comput. Vis. Image Underst. *63* (1996), 3, 542–567.

[11] F. Dellaert, S. M. Seitz, Ch. E. Thorpe, and S. Thrun: Structure from motion without correspondence. In: IEEE Conf. on Computer Vision and Pattern Recognition 2000, pp. 557–564.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin: Maximum likelihood from incomplete data via the em algorithm. J. Roy. Statist. Soc. Ser. B *39* (1977), 1, 1–38.

[13] M. A. Fischler and R. C. Bolles: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM *24* (1981), 6, 381–395.

[14] A. W. Fitzgibbon and A. Zisserman: Automatic 3D model acquisition and generation of new images from video sequences. In: Proc.European Signal Processing Conference (EUSIPCO '98), Rhodes 1998, pp. 1261–1269.

[15] A. W. Fitzgibbon and A. Zisserman: Automatic camera recovery for closed or open image sequences. In: Proc. European Conference on Computer Vision 1998, pp. 311–326.

[16] J.-M. Frahm and M. Pollefeys: Ransac for (quasi-)degenerate data (qdegsac). In: CVPR '06: Proc. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington 2006, IEEE Computer Society, pp. 453–460.

[17] J. H. Friedman, J. L. Bentley, and R. A. Finkel: An algorithm for finding best matches in logarithmic expected time. ACM Trans. Math. Software *3* (1997), 3, 209–226.

[18] M. Pollefeys, L J. Van Gool, G. Van Meerbergen, and M. Vergauwen: A hierarchical symmetric stereo algorithm using dynamic programming. Internat. J. Comput. Vision *47* (2002), 275–285.

[19] K. Häming and G. Peters: Extension of the generalized image rectification – Catching the infinity cases. In: Proc. 4th International Conference on Informatics in Control, Automation, and Robotics (ICINCO 2007) (J. Zaytoon, J.-L. Ferrier, J. A. Cetto, and J. Filipe, eds.), Vol. RA-2, Angers 2007, Institute for Systems and Technologies of Information, Control and Communication, pp. 275–279.

[20] Ch. Harris and M. Stephens: A combined Corner and Edge detector. In: 4th ALVEY Vision Conference 1988, pp. 147–151.

[21] R. I. Hartley and A. Zisserman: Multiple View Geometry in Computer Vision. Second edition. Cambridge University Press 2004.

[22] R. Koch, M. Pollefeys, and L. J. Van Gool: Realistic surface reconstruction of 3d scenes from uncalibrated image sequences. J. Visualization and Computer Animation *11* (2000), 3, 115–127.

[23] M. Lhuillier and L. Quan: A quasi-dense approach to surface reconstruction from uncalibrated images. IEEE Trans. Pattern Analysis and Machine Intelligence *27* (2005), 3, 418–433.

[24] T. Lindeberg: Feature detection with automatic scale selection. Internat. J. Comput. Vision *30* (1998), 2, 77–116.

[25] T. Lindeberg and L. Bretzner: Real-time scale selection in hybrid multi-scale representations. In: Proc. Scale-Space, Lect. Notes in Comput. Sci. *2695*, Springer 2003, pp. 148–163.

[26] D. G. Lowe: Distinctive image features from scale-invariant keypoints. Internat. J. Comput. Vision *60* (2004), 2, 91–110.

[27] B. D. Lucas and T. Kanade: An iterative image registration technique with an application to stereo vision. In: IJCAI81, pp. 674–679.

[28] K. Mikolajczyk and C. Schmid: Scale and affine invariant interest point detectors. Internat. J. Comput. Vision *60* (2004), 1, 63–86.

[29] G. Peters and K. Häming: Fast freehand acquisition of 3d objects and their visualization. J. Commun. Comput. *7* (2010), 2–3.

[30] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch: Visual modeling with a hand-held camera. Internat. J. Comput. Vision *59* (2004), 3, 207–232.

[31] M. Pollefeys, R. Koch, and L. J. van Gool: Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In: ICCV 1998, pp. 90–95.

[32] M. Pollefeys, R. Koch, and L. J. van Gool: A simple and efficient rectification method for general motion. In: Proc. Internat. Conference on Computer Vision (ICCV 1999), pp. 496–501.

[33] M. Pollefeys, F. Verbiest, and L. Van Gool: Surviving dominant planes in uncalibrated structure and motion recovery. In: Computer Vision – ECCV 2002, 7th European Conference on Computer Vision (Johansen, ed.). Lect. Notes Comput. Sci. *2351*, Springer-Verlag 2002, pp. 837–851.

[34] M. Pollefeys, M. Vergauwen, K. Cornelis, J. Tops, F. Verbiest, and L. Van Gool. Structure and motion from image sequences. In: Proc. Conference on Optical 3-D Measurement Techniques V (K. Gruen, ed.), Vienna 2001. pp. 251–258.

[35] J. Ponce, T. Papadopoulo, M. Teillaud, and B. Triggs: On the absolute quadratic complex and its application to autocalibration. In: IEEE Conference on Computer Vision & Pattern Recognition 2005, Vol. I., pp. 780–787.

[36] M. Prasad and A. W. Fitzgibbon: Single view reconstruction of curved surfaces. In: IEEE Conf. on Computer Vision and Pattern Recognition 2006, Vol. 02, pp. ,1345–1354.

[37] A. Saxena, M. Sun, and A. Y. Ng: Make3d: Depth perception from a single still image. In: AAAI (D. Fox and C. P. Gomes, eds.), AAAI Press 2008, pp. 1571–1576.

[38] F. Schaffalitzky and A. Zisserman: Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". In: Proc. 7th European Conference on Computer Vision, Copenhagen 2002, Springer, Vol. 1, pp. 414–431.

[39] F. Schaffalitzky, A. Zisserman, R. I. Hartley, and P. H. S. Torr: A six point solution for structure and motion. In: ECCV '00: Proc. 6th European Conference on Computer Vision, Vol. I, London 2000, Springer, pp. 632–648.

[40] F. Shen and H. Wang: A local edge detector used for finding corners. Proc. ICICS, 2001.

[41] J. Shi and C. Tomasi. Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), Seattle 1994.

[42] N. Snavely, S. M. Seitz, and R. Szeliski: Photo tourism: Exploring photo collections in 3d. ACM Trans. on Graphics (SIGGRAPH Proc.), *25* (2006), 3, 835–846.

[43] P. H. S. Torr and A. Zisserman: Mlesac: a new robust estimator with application to estimating image geometry. Comput. Vis. Image Underst. *78* (2000), 1, 138–156.

[44] B. Triggs: Autocalibration and the absolute quadric. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico 1977, IEEE Computer Society Press, pp. 609–614.

[45] R. Y. Tsai: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. In: Radiometry (L. B. Wolff, S. A. Shafer, and G. Healey, eds.), Jones and Bartlett Publishers, Inc., pp. 221–244, 1992.

[46] M. Vergauwen and L. Van Gool: Web-based 3d reconstruction service. Mach. Vision Appl. *17* (2006), 6, 411–426.

[47] P. Viola and M. Jones: Rapid object detection using a boosted cascade of simple features. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2001, Vol. 1, 511.

[48] O. J. Woodford, P. H. S. Torr, I. D. Reid, and A. W. Fitzgibbon: Global stereo reconstruction under second order smoothness priors. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage 2008.

[49] Z. Zhang: A flexible new technique for camera calibration. IEEE Trans. Pattern Analysis and Machine Intelligence *22* (1998), 1330–1334.

Klaus Häming, FernUniversität in Hagen, Fakultät für Mathematik und Informatik, Lehrgebiet Mensch-Computer-Interaktion, Universitätsstr. 1, D-58097 Hagen. Germany.
e-mail: klaus.haeming@FernUni-Hagen.de

Gabriele Peters, FernUniversität in Hagen, Fakultät für Mathematik und Informatik, Lehrgebiet Mensch-Computer-Interaktion, Universitätsstr. 1, D-58097 Hagen. Germany.
e-mail: gabriele.peters@FernUni-Hagen.de