An Efficient Solution to the Five-Point Relative Pose Problem

David Nistér, *Member*, *IEEE*

Abstract—An efficient algorithmic solution to the classical five-point relative pose problem is presented. The problem is to find the possible solutions for relative camera pose between two calibrated views given five corresponding points. The algorithm consists of computing the coefficients of a tenth degree polynomial in closed form and, subsequently, finding its roots. It is the first algorithm well-suited for numerical implementation that also corresponds to the inherent complexity of the problem. We investigate the numerical precision of the algorithm. We also study its performance under noise in minimal as well as overdetermined cases. The performance is compared to that of the well-known 8 and 7-point methods and a 6-point scheme. The algorithm is used in a robust hypothesize-and-test framework to estimate structure and motion in real-time with low delay. The real-time system uses solely visual input and has been demonstrated at major conferences.

Index Terms—Imaging geometry, motion, relative orientation, structure from motion, camera calibration, ego-motion estimation, scene reconstruction.

1 Introduction

Reconstruction of camera positions and scene structure based on images of scene features from multiple viewpoints has been studied for more than two centuries, first by the photogrammetry community and more recently in computer vision. In the classical setting, the intrinsic parameters of the camera, such as focal length, are assumed known a priori. This calibrated setting is where the five-point problem arises. Given the images of five unknown scene points from two distinct unknown viewpoints, what are the possible solutions for the configuration of the points and cameras? Clearly, only the relative positions of the points and cameras can be recovered. Moreover, the overall scale of the configuration can never be recovered solely from images. Apart from this ambiguity, the five-point problem was proven by Kruppa [19] to have at most eleven solutions. This was later improved upon [3], [4], [6], [22], [16], showing that there are at most 10 solutions and that there are 10 solutions in general (including complex ones). The 10 solutions correspond to the roots of a tenth degree polynomial. However, Kruppa's method requires the nontrivial operation of finding all intersections between two sextic curves and there is no previously known practical method of deriving the coefficients of the tenth degree polynomial in the general case. A few algorithms suitable for numerical implementation have also been devised. In [44], a 60×60 sparse matrix is built, which is subsequently reduced using linear algebra to a 20×20 nonsymmetric matrix whose eigenvalues and eigenvectors encode the solution to the problem. In [32], an efficient derivation is given that leads to a thirteenth degree polynomial whose roots include the solutions to the five-point problem. The solution presented in this paper is a refinement of this. A better elimination that leads directly in closed form to the tenth degree polynomial is used. Thus, an efficient algorithm that corresponds exactly to the intrinsic degree of difficulty of the problem is obtained.

For the structure and motion estimation to be robust and accurate, in practice, more than five points have to be used. The classical way of making use of many points is to minimize a least squares measure over all points, see, for example, [18]. Our intended application for the five-point algorithm is as a hypothesis generator within a random sample consensus scheme (RANSAC) [8], [26]. Many random samples containing five point correspondences are taken. Each sample yields a number of hypotheses for the relative orientation that are scored by a robust statistical measure over all points in two or more views. The best hypothesis is then refined iteratively. Such a hypothesize-and-test architecture has become the standard way of dealing with mismatched point correspondences [41], [48], [14], [24] and has made automatic reconstructions spanning hundreds of views possible [1], [34], [7], [25].

The requirement of prior intrinsic calibration was relaxed in the last decade [5], [12], [14], leading to higher flexibility and less complicated algorithms. So, why consider the calibrated setting? Apart from the theoretical interest, one answer to this question concerns stability and uniqueness of solutions. Enforcing the intrinsic calibration constraints often gives a crucial improvement of both the accuracy and robustness of the structure and motion estimates. Currently, the standard way of achieving this is through an initial uncalibrated estimate followed by iterative refinement to bring the estimate into agreement with the calibration constraints. When the intrinsic parameters *are* known a priori, the five-point algorithm is a more direct way of enforcing the calibration constraints exactly and obtaining a Euclidean reconstruction. The accuracy and robustness improvements gained by enforcing the calibration constraints are particularly

• *The author is with the Sarnoff Corporation, CN5300, Princeton, NJ 08530. E-mail: dnister@sarnoff.com.*

Manuscript received 13 Aug. 2003; revised 6 Jan. 2004; accepted 29 Jan. 2004. Recommended for acceptance by R. Klette.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0229-0803.

significant for planar or near planar scenes and scenes that *appear* planar in the imagery. The uncalibrated methods fail when faced with coplanar scene points, since there is then a continuum of possible solutions. It has been proposed to deal with this degeneracy using model selection [43], [35], switching between a homographic model and the general uncalibrated model as appropriate. In the calibrated setting, coplanar scene points only cause at most a two-fold ambiguity [21], [23]. With a third view, the ambiguity is, in general, resolved. In light of this, a RANSAC scheme that uses the five-point algorithm over three or more views is proposed. It applies to general structure but also continues to operate correctly *despite* scene planarity, without relying on or explicitly detecting the degeneracy. In essence, the calibrated model can cover both the planar and general structure cases seamlessly. This gives some hope of dealing with the approximately planar cases, where neither the planar nor the uncalibrated general structure model applies well.

The rest of the paper is organized as follows: Section 2 establishes some notation and describes the constraints used in the calibrated and uncalibrated cases. Section 3 presents the 5-point algorithm. Section 4 discusses planar degeneracy. Section 5 outlines the RANSAC schemes for two and three views. Section 6 gives results. The numerical accuracy of the algorithm is investigated in Section 6.1. The distribution of the number of solutions is studied in Section 6.2. Timing information is given in Section 6.3. The performance of the algorithm in noisy conditions is studied in Section 6.4, where the performance of the 5-point algorithm is compared to that of the well-known 8 and 7-point algorithms and a 6-point scheme. Some reconstruction results are given in Section 6.5. Section 7 concludes.

2 Preliminaries

Image points are represented by homogeneous 3-vectors q and q' in the first and second view, respectively. World points are represented by homogeneous 4-vectors Q. A perspective view is represented by a 3×4 camera matrix P indicating the image projection $q \sim PQ$, where $\sim$ denotes equality up to scale. A view with a finite projection center can be factored into $P = K[R \mid t]$, where K is a 3×3 upper triangular calibration matrix holding the intrinsic parameters and R is a rotation matrix. Let the camera matrices for the two views be $K_1[I \mid 0]$ and $P = K_2[R \mid t]$. Let $[t]_\times$ denote the skew symmetric matrix

$$[t]_\times \equiv \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix} \tag{1}$$

so that $[t]_\times x = t \times x$ for all x. Then, the fundamental matrix is

$$F \equiv K_2^{-\top}[t]_\times R K_1^{-1}. \tag{2}$$

The fundamental matrix encodes the well-known coplanarity or epipolar constraint

$$q'^\top F q = 0. \tag{3}$$

The fundamental matrix can be considered without knowledge of the calibration matrices. Moreover, it continues to exist when the projection centers are not finite. If K_1 and K_2 are known, the cameras are said to be calibrated. In this case, we can always assume that the image points q and q' have been premultiplied by K_1^{-1} and K_2^{-1}, respectively, so that the epipolar constraint simplifies to

$$q'^\top E q = 0, \tag{4}$$

where the matrix $E \equiv [t]_\times R$ is called the essential matrix. Any rank-2 matrix is a possible fundamental matrix, i.e., we have the well-known single cubic constraint, e.g., [14]:

Theorem 1. *A real nonzero 3×3 matrix, F, is a fundamental matrix if and only if it satisfies the equation*

$$det(F) = 0. \tag{5}$$

An essential matrix has the additional property that the two nonzero singular values are equal. This leads to the following cubic constraints on the essential matrix, adapted from [38], [6], [22], [32]:

Theorem 2. *A real nonzero 3×3 matrix, E, is an essential matrix if and only if it satisfies the equation*

$$EE^\top E - \frac{1}{2} trace(EE^\top)E = 0. \tag{6}$$

Both (5) and (6) will help us recover the essential matrix. Once the essential matrix is known, R, t, and the camera matrices can be recovered from it.

3 The Five-Point Algorithm

In this section, the five-point algorithm is described, first in a straightforward manner. Recommendations for an efficient implementation are then given in Section 3.2. Each of the five point correspondences gives rise to a constraint of the form (4). This constraint can also be written as

$$\tilde{q}^\top \tilde{E} = 0, \tag{7}$$

where

$$\tilde{q} \equiv \left[q_1 q'_1 \; q_2 q'_1 \; q_3 q'_1 \; q_1 q'_2 \; q_2 q'_2 \; q_3 q'_2 \; q_1 q'_3 \; q_2 q'_3 \; q_3 q'_3\right]^\top \tag{8}$$

$$\tilde{E} \equiv [E_{11} \; E_{12} \; E_{13} \; E_{21} \; E_{22} \; E_{23} \; E_{31} \; E_{32} \; E_{33}]^\top. \tag{9}$$

By stacking the vectors $\tilde{q}^\top$ for all five points, a 5×9 matrix is obtained. Four vectors $\tilde{X}, \tilde{Y}, \tilde{Z}, \tilde{W}$ that span the right nullspace of this matrix are now computed. The most common way to achieve this is by singular value decomposition [36], but QR-factorization as described in Section 3.2 is much more efficient. The four vectors correspond directly to four 3×3 matrices X, Y, Z, W and the essential matrix must be of the form

$$E = xX + yY + zZ + wW \tag{10}$$

for some scalars x, y, z, w. The four scalars are defined only up to a common scale factor and it is therefore assumed that $w = 1$. Note here that the algorithm can be extended to using more than five points in much the same way as the uncalibrated 7 and 8-point methods. In the overdetermined case, the four singular vectors X, Y, Z, W that correspond to the four smallest singular values are used. By inserting (10)

into the 10 cubic constraints (5), (6), and performing Gauss-Jordan elimination with partial pivoting, we obtain equation system A:

A	x^3	y^3	x^2y	xy^2	x^2z	x^2	y^2z	y^2	xyz	xy	x	y	1
$\langle a \rangle$	1	.	.	.	.	.	.	.	.	.	[2]	[2]	[3]
$\langle b \rangle$		1	.	.	.	.	.	.	.	.	[2]	[2]	[3]
$\langle c \rangle$			1	.	.	.	.	.	.	.	[2]	[2]	[3]
$\langle d \rangle$				1	.	.	.	.	.	.	[2]	[2]	[3]
$\langle e \rangle$					1						[2]	[2]	[3]
$\langle f \rangle$						1					[2]	[2]	[3]
$\langle g \rangle$							1				[2]	[2]	[3]
$\langle h \rangle$								1			[2]	[2]	[3]
$\langle i \rangle$									1		[2]	[2]	[3]
$\langle j \rangle$										1	[2]	[2]	[3]

where . denotes some scalar value and $[N]$ denotes a polynomial of degree N in the variable z. Note that the elimination can optionally be stopped four rows early. Further, define the additional equations

$$\langle k \rangle \equiv \langle e \rangle - z\langle f \rangle \tag{11}$$

$$\langle l \rangle \equiv \langle g \rangle - z\langle h \rangle \tag{12}$$

$$\langle m \rangle \equiv \langle i \rangle - z\langle j \rangle. \tag{13}$$

These equations are arranged into a 3×3 matrix B containing polynomials in z:

B	x	y	1
$\langle k \rangle$	[3]	[3]	[4]
$\langle l \rangle$	[3]	[3]	[4]
$\langle m \rangle$	[3]	[3]	[4]

Since the vector $[x \;\; y \;\; 1]^\top$ is a nullvector to B, the determinant of B must vanish. The determinant is the tenth degree polynomial

$$\langle n \rangle \equiv det(B). \tag{14}$$

The real roots of $\langle n \rangle$ are now computed. There are various standard methods to accomplish this. A highly efficient way is to use Sturm-sequences [9] to bracket the roots, followed by a root-polishing scheme. This is described in Section 3.2. Another method, which is easy to implement with most linear algebra packages, is to eigen-decompose a companion matrix. After normalizing $\langle n \rangle$ so that $n_{10} = 1$, the roots are found as the eigenvalues of the 10×10 companion matrix

$$\begin{bmatrix} -n_9 & -n_8 & \cdots & -n_0 \\ 1 & & & \\ & \ddots & & \\ & & 1 & \end{bmatrix}. \tag{15}$$

For each root z, the variables x and y can be found using equation system B. The essential matrix is then obtained from (10). In Section 3.1, it is described how to recover R and t from the essential matrix.

3.1 Recovering R and t from E

Let

$$D \equiv \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{16}$$

R and t are recovered from the essential matrix on the basis of the following theorem [46], [14]:

Theorem 3. *Let the singular value decomposition of the essential matrix be $E \sim Udiag(1,1,0)V^\top$, where U and V are chosen such that $det(U) > 0$ and $det(V) > 0$. Then, $t \sim t_u \equiv [u_{13} \;\; u_{23} \;\; u_{33}]^\top$ and R is equal to $R_a \equiv UDV^\top$ or $R_b \equiv UD^\top V^\top$.*

Any combination of R and t according to the above prescription satisfies the epipolar constraint (4). To resolve the inherent ambiguities, it is assumed that the first camera matrix is $[I \mid 0]$ and that t is of unit length. There are then the following four possible solutions for the second camera matrix: $P_A \equiv [R_a \mid t_u]$, $P_B \equiv [R_a \mid -t_u]$, $P_C \equiv [R_b \mid t_u]$, $P_D \equiv [R_b \mid -t_u]$. One of the four choices corresponds to the true configuration. Another one corresponds to the twisted pair that is obtained by rotating one of the views 180 degrees around the baseline. The remaining two correspond to reflections of the true configuration and the twisted pair. For example, P_A gives one configuration. P_C corresponds to its twisted pair, which is obtained by applying the transformation

$$H_t \equiv \begin{bmatrix} I & & 0 \\ -2v_{13} & -2v_{23} & -2v_{33} & -1 \end{bmatrix}. \tag{17}$$

P_B and P_D correspond to the reflections obtained by applying $H_r \equiv diag(1,1,1,-1)$. In order to determine which choice corresponds to the true configuration, the cheirality constraint[1] is imposed. One point is sufficient to resolve the ambiguity. The point is triangulated using the view pair $([I \mid 0], P_A)$ to yield the space point Q and cheirality is tested. If $c_1 \equiv Q_3 Q_4 < 0$, the point is behind the first camera. If $c_2 \equiv (P_A Q)_3 Q_4 < 0$, the point is behind the second camera. If $c_1 > 0$ and $c_2 > 0$, P_A and Q correspond to the true configuration. If $c_1 < 0$ and $c_2 < 0$, the reflection H_r is applied and we get P_B. If, on the other hand, $c_1 c_2 < 0$, the twist H_t is applied and we get P_C and the point $H_t Q$. In this case, if $Q_3 (H_t Q)_4 > 0$, we are done. Otherwise, the reflection H_r is applied and we get P_D.

3.2 Efficiency Considerations

In summary, the main computational steps of the algorithm outlined above are as follows:

1. Extraction of the nullspace of a 5×9 matrix.
2. Expansion of the cubic constraints (5) and (6).
3. Gauss-Jordan elimination with partial pivoting on the 10×20 matrix A.
4. Expansion of the determinant polynomial of the 3×3 polynomial matrix B to obtain the tenth degree polynomial (14).
5. Extraction of roots from the tenth degree polynomial.
6. Recovery of R and t corresponding to each real root and point triangulation for disambiguation.

We will discuss efficient implementation of each step.

1. The constraint that the scene points should be in front of the cameras.

3.2.1 Step 1: Nullspace Extraction

Singular value decomposition is the gold standard for the nullspace extraction in Step 1, but a specifically tailored QR-factorization is much more efficient. The five input vectors are orthogonalized first, while pivoting, to form the orthogonal basis $\tilde{q}_1, \ldots, \tilde{q}_5$. This basis is then amended with the 9×9 identity matrix to form the matrix

$$[\, \tilde{q}_1 \quad \cdots \quad \tilde{q}_5 \quad | \quad I \,]^\top. \tag{18}$$

The orthogonalization with pivoting is now continued until nine orthogonal vectors are obtained. The last four rows constitute an orthogonal basis for the nullspace.

3.2.2 Step 2: Constraint Expansion

An efficient way to implement Step 2 is to create a function $o_1(p_i, p_j)$ that multiplies two polynomials of degree one in x, y, z and another function $o_2(p_i, p_j)$ that multiplies two polynomials p_i and p_j of degrees two and one, respectively. Equation (5) is then handled through expansion by minors, e.g., as

$$\begin{aligned} det(E) =& o_2(o_1(E_{12}, E_{23}) - o_1(E_{13}, E_{22}), E_{31}) + \\ & o_2(o_1(E_{13}, E_{21}) - o_1(E_{11}, E_{23}), E_{32}) + \\ & o_2(o_1(E_{11}, E_{22}) - o_1(E_{12}, E_{21}), E_{33}). \end{aligned} \tag{19}$$

To expand (6), we compute the upper triangular part of the symmetric polynomial matrix $EE^\top$ by

$$(EE^\top)_{ij} = \sum_{k=1}^{3} o_1(E_{ik}, E_{jk}), \tag{20}$$

followed by the upper triangular part of the symmetric polynomial matrix

$$\Lambda \equiv EE^\top - \frac{1}{2} trace(EE^\top) I \tag{21}$$

through

$$\Lambda_{ij} = \begin{cases} (EE^\top)_{ij} - \frac{1}{2}\sum_{k=1}^{3} (EE^\top)_{kk} & i = j \\ (EE^\top)_{ij} & i \neq j. \end{cases} \tag{22}$$

Equation (6) is the same as $\Lambda E = 0$ and we compute ΛE through the standard matrix multiplication formula

$$(\Lambda E)_{ij} = \sum_{k=1}^{3} o_2(\Lambda_{ik}, E_{kj}), \tag{23}$$

while using only the upper triangular part of Λ.

3.2.3 Step 3: Gauss-Jordan Elimination

Gauss-Jordan elimination with partial pivoting is presented, e.g., in [36]. For optimal efficiency, the elimination is stopped four rows before completion.

3.2.4 Step 4: Determinant Expansion

The determinant polynomial $\langle n \rangle$ in (14) is computed through expansion by minors, e.g.,

$$p_1 \equiv B_{12}B_{23} - B_{13}B_{22} \tag{24}$$

$$p_2 \equiv B_{13}B_{21} - B_{11}B_{23} \tag{25}$$

$$p_3 \equiv B_{11}B_{22} - B_{12}B_{21} \tag{26}$$

followed by

$$\langle n \rangle \equiv p_1 B_{31} + p_2 B_{32} + p_3 B_{33}. \tag{27}$$

The cofactor polynomials are retained so that, for each root z, one can recover x and y by

$$x = p_1(z)/p_3(z) \quad y = p_2(z)/p_3(z). \tag{28}$$

3.2.5 Step 5: Root Extraction

Sturm sequences are used to bracket the roots in Step 5. The definition of a Sturm sequence, also called Sturm chain, is given in Appendix A. The tenth degree polynomial has an associated Sturm sequence, which consists of 11 polynomials of degree zero to 10. The number of real roots in an interval can be determined by counting the number of sign changes in the Sturm sequence at the two endpoints of the interval. The Sturm sequence can be evaluated recursively with 38 floating point operations. Ten additional operations are required to count the number of sign changes. This is to be put in relation to the 20 floating point operations required to evaluate the polynomial itself. With this simple test for number of roots in an interval, it is fairly straightforward to hunt down a number of intervals, each containing one of the real roots of the polynomial. Any root polishing scheme [36] can then be used to determine the roots accurately. In our experiments, we simply use 30 iterations of bisection, since this provides a guaranteed precision in fixed time and requires almost no control overhead.

3.2.6 Step 6: R and t Recovery

Step 6 requires a singular value decomposition of the essential matrix and triangulation of one or more points. When all the other steps of the algorithm have been efficiently implemented, these operations can take a significant portion of the computation time since they have to be carried out for each real root. A specifically tailored singular value decomposition is given in Appendix B. Efficient triangulation is discussed in Appendix B. Note that a triangulation scheme that assumes ideal point correspondences can be used since, for true solutions, the recovered essential matrix is such that intersection is guaranteed for the five pairs of rays.

4 Planar Structure Degeneracy

The planar structure degeneracy is an interesting example of the differences between the calibrated and uncalibrated frameworks. The degrees of ambiguity that arise from a planar scene in the two frameworks are summarized in Table 1. For pose estimation with known intrinsics, there is a unique solution provided that the plane is finite and that the cheirality constraint is taken into account.[2] In theory, focal length can also be determined if the principal direction does not coincide with the plane normal. Without knowledge of the intrinsics, however, there is a three degree-of-freedom ambiguity that can be thought of as parameterized by the position of the camera center. For any camera center, appropriate choices for the calibration matrix K and rotation matrix R can together produce any homography between the plane and the image.

2. If the plane is the plane at infinity, it is impossible to determine the camera position, and without the cheirality constraint, the reflection across the plane constitutes a second solution.

TABLE 1
The Degrees of Ambiguity in the Face of Planar Degeneracy for Pose Estimation and Structure and Motion Estimation

	1 View Known Structure	2 Views Unknown Structure	$n > 2$ Views Unknown Structure
Known intrinsics	Unique	Two-fold or unique	Unique
Unknown fixed focal length	Unique in general	1 d.o.f.	Unique in general
Unknown variable intrinsics	3 d.o.f.	2 dof Projective, +8 for Metric	3n-4 dof Projective, +8 Metric

The motion is assumed to be general and the structure is assumed to be dense in the plane. See the text for further explanation.

With known intrinsics and two views of an unknown plane, there are two solutions for the essential matrix [21], [23], unless the baseline is perpendicular to the plane in which case there is a unique solution. The cheirality constraint resolves the ambiguity unless all visible points are closer to one viewpoint than the other [21]. If all visible points are closer to one viewpoint, the false solution is obtained from the true one by reflecting that view across the plane and then taking the twisted pair of the resulting configuration. The structure of the false solution also resides in a plane. However, the structure is projectively distorted. The twisted pair operation maps the plane of points that have the same distance to both viewpoints to the plane at infinity. Hence, the line of points in the plane that have the same distance to both viewpoints is mapped to infinity.

The 5-point method is essentially unaffected by the planar degeneracy and still works. Six correspondences from coplanar but otherwise general points provide linearly independent constraints on the essential matrix [33]. However, the 6-point method fails for planar scenes.

Any attempts to recover intrinsic parameters from two views of a planar surface are futile according to the following theorem, adapted from [22]:

Theorem 4. *For any choice of intrinsic parameters, any homography can be realized between two views by some positioning of the two views and a plane.*

If the calibration matrices are completely unknown, there is a two degree-of-freedom ambiguity for projective reconstruction, which can be thought of as parameterized by the epipole in one of the images, i.e., for any choice of epipole in the first image, there is a unique valid solution. Once the epipole is specified in the first image, the problem of solving for the remaining parameters of the fundamental matrix is algebraically equivalent to solving for the projective pose of a one-dimensional camera in a two-dimensional world, where the projection center of the 1D camera corresponds to the epipole in the second image, the orientation corresponds to the epipolar line homography, and the points in the second image correspond to world points in the 2D space. The problem according to Steiner's and Chasles' theorems [37] has a unique solution unless all the points and the epipole in the second image lie on a conic, which is not the case since we are assuming that the structure is dense in the plane.

For three views with known intrinsics, there is a unique solution. If the views are in general position, a common unknown focal length can also be recovered, but this requires rotation and suffers from additional critical configurations. With unknown variable intrinsics, there are three additional degrees of freedom for each view above two.

5 Applying the Algorithm Together with Preemptive RANSAC

We use the algorithm in conjunction with preemptive random sampling consensus in two or three views. A number of random samples are taken, each containing five point-tracks. The five-point algorithm is applied to each sample and, thus, a number of hypotheses are generated. We then seek the best hypothesis according to a robust measure over all the point-tracks. As described in [26], preemptive scoring is used for efficiency reasons. See, also, [2]. Finally, the best hypothesis is polished by iterative refinement [45].

When three or more views are available, we prefer to disambiguate and score the hypotheses utilizing three views. A unique solution can then be obtained from each sample of five tracks and this continues to hold true even if the scene points are all perfectly coplanar. For each sample of five point-tracks, the points in the first and last view are used in the five-point algorithm to determine a number of possible camera matrices for the first and last view. For each case, the five points are triangulated.[3] The remaining view can now be determined by any 3-point calibrated perspective pose algorithm, see [11] for a review and additional references. Up to four solutions are obtained and disambiguated by the additional two points. The reprojection errors of the five points in all of the views are now enough to single out one hypothesis per sample. Finally, the solutions from all samples are scored preemptively by a robust measure using all available point tracks. To score motion hypotheses over three views, we use a Sampson approximation [14], [42], [25] of the minimum image perturbation required to bring a triplet of points to trifocal incidence. A minimal closed form expression is essential for real-time performance. Since no such expression has been given in the literature, we present one in Appendix D.

6 Results

In the minimal case with five points, the two main requirements on the five-point method are accuracy and speed. The numerical accuracy of the algorithm is investigated in Section 6.1. The computation time is partially dependent on the number of real solutions. The distribution of the number of solutions is studied in Section 6.2. Timing information for our efficient implementation of the five-point algorithm is given in Section 6.3. The performance of the algorithm in noisy conditions is studied in Section 6.4. The performance of the 5-point algorithm is compared to that of the 6, 7, and 8-point algorithms, briefly described in Section 6.4. Since the algorithm can also be used as a least-squares method, results

3. See Appendix C.

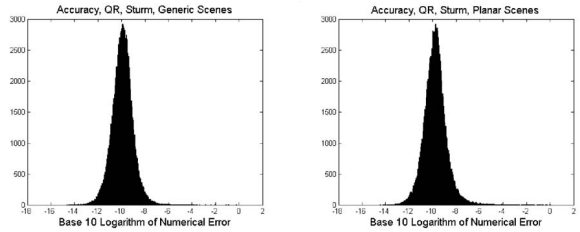


Fig. 1. Distribution of numerical error in the essential matrix $\hat{E}$ based on 10^5 random tests. QR is used in Step 1 and Sturm-bracketing plus root polishing in Step 5. The median error is $1.2 \cdot 10^{-10}$ for generic and $1.6 \cdot 10^{-10}$ for planar scenes. 0.1 percent of the trials for generic and 0.5 percent for planar scenes had error magnitudes above 10^{-6}.

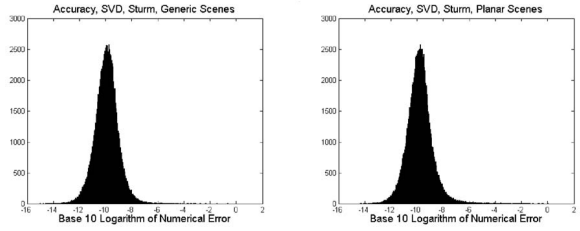


Fig. 2. Distribution of the numerical error when the QR-decomposition in Step 1 is replaced by SVD. The median error is $1.2 \cdot 10^{-10}$ for generic and $1.5 \cdot 10^{-10}$ for planar scenes. 0.1 percent of the trials for generic and 0.6 percent for planar scenes had error magnitudes above 10^{-6}.

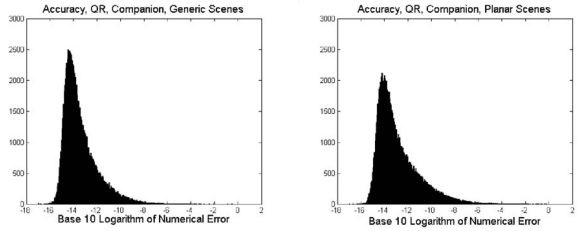


Fig. 3. Distribution of the numerical error when the Sturm-bracketing and root polishing in Step 5 is replaced by eigenvalue-decomposition of a companion matrix. The median error is $1.6 \cdot 10^{-14}$ for generic and $4.4 \cdot 10^{-14}$ for planar scenes. 0.1 percent of the trials for generic and 0.6 percent for planar scenes had error magnitudes above 10^{-6}.

are presented both for minimal and overdetermined cases. Note that in the minimal case, the effects of noise should be the same for any five-point solution method. In the overdetermined case however, this is no longer true. In fact, most of the previously suggested five-point solution methods do not generalize naturally to the overdetermined case.

The algorithm is used as a part of a system that reconstructs structure and motion from video live and in real-time. The system has been demonstrated at major conferences [26], [27], [28], [29]. Some system information and results are given in Section 6.5.

6.1 Numerical Accuracy

The numerical precision of different incarnations of the algorithm is investigated in Figs. 1, 2, 3, and 4. Since the essential matrix is defined only up to scale and there are multiple solutions $\hat{E}_i$, the minimum residual

$$\min_i \min \left(\left\| \frac{\hat{E}_i}{\| \hat{E}_i \|} - \frac{E}{\| E \|} \right\|, \left\| \frac{E}{\| E \|} + \frac{\hat{E}_i}{\| \hat{E}_i \|} \right\| \right) \tag{29}$$

from each problem instance is used. All computations were performed in double precision. The accuracy of the implementation that uses QR-decomposition in Step 1 and

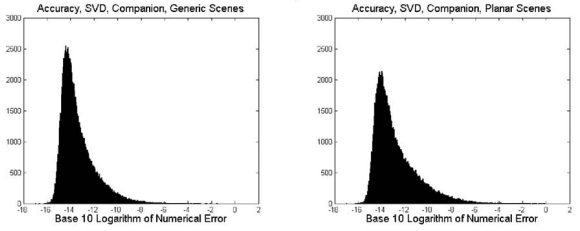


Fig. 4. Distribution of the numerical error with SVD in Step 1 and eigenvalue-decomposition of a companion matrix in Step 5. The median error is $1.7 \cdot 10^{-14}$ for generic and $4.5 \cdot 10^{-14}$ for planar scenes. 0.1 percent of the trials for generic and 0.6 percent for planar scenes had error magnitudes above 10^{-6}.

TABLE 2
The Distribution of the Number of Hypotheses that Result from Computational Steps 5 and 6 (as Numbered in Section 3.2)

Nr Hyp	0	1	2	3	4	5	6	7	8	9	10
Step 5	0	.	0.12	.	0.50	.	0.36	.	0.15	.	$4.9e-4$
Step 6	$4.2e-6$	0.17	0.28	0.29	0.17	$5.8e-2$	$2.5e-2$	$1.5e-3$	$6.6e-4$	$1.5e-6$	$2e-7$

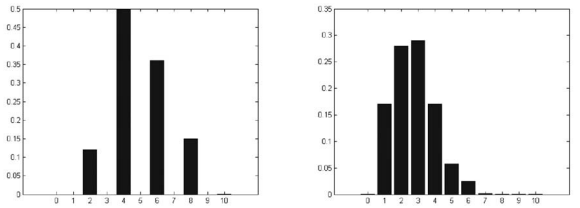
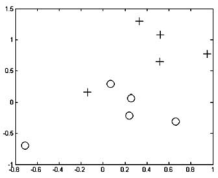


Fig. 5. Graphs of the distributions from Table 2.

Sturm-bracketing followed by root polishing in Step 5 is shown in Fig. 1 for generic and planar scenes. Note that the typical errors are insignificant in comparison to realistic noise levels. The accuracy of the algorithm when the QR-decomposition in Step 1 is replaced by singular value decomposition is shown in Fig. 2. The accuracy of the algorithm when the Sturm-bracketing and root polishing in Step 5 is replaced by eigenvalue-decomposition of a companion matrix is shown in Fig. 3. Fig. 4 shows the accuracy when the QR-decomposition in Step 1 is replaced by singular value decomposition and the Sturm-bracketing and root polishing in Step 5 is replaced by eigenvalue-decomposition of a companion matrix.

6.2 The Number of Solutions

The computation time is partially dependent on the number of real solutions. The distribution of the number of solutions is given in Table 2. The second row shows the distribution of the number of real roots of the tenth degree polynomial $\langle n \rangle$ in (14), based on 10^5 random point and view configurations. The average is 4.55 roots. The third row shows the distribution of the number of hypotheses once the cheirality constraint has been enforced, based on 10^7 random point and view configurations. The average number of hypotheses is 2.74. Both rows show fractions of the total number of trials. The distributions are also depicted in Fig. 5. An example of five image correspondences that give rise to 10 distinct physically valid solutions is given in Fig. 6. We have also verified experimentally that five points in three views, in general,



$$
\begin{aligned}
&[\ 0.067,\ \ 0.287\ ] \leftrightarrow [\ 0.329, 1.297\ ]\\
&[\ 0.254,\ \ 0.0646] \leftrightarrow [\ 0.523, 1.0807]\\
&[\ 0.239, -0.213\ ] \leftrightarrow [\ 0.517, 0.645\ ]\\
&[-0.710, -0.693\ ] \leftrightarrow [-0.141, 0.157\ ]\\
&[\ 0.661, -0.307\ ] \leftrightarrow [\ 0.950, 0.773\ ]
\end{aligned}
$$

Fig. 6. An example of five point correspondences that give rise to 10 distinct physically valid solutions that are well-separated in parameter space and are not caused by numerical inaccuracies. Our current randomization leads to two cases in 10^7 with 10 distinct physically valid solutions.

TABLE 3
Timings for the Algorithm Steps (as Numbered in Section 3.2) on a Modest 550MHz Machine with Highly Optimized but Platform-Independent Code

Step	1	2	3	4	5	6	Three-Point Pose	Mean Two Views	Mean Three Views
μs	8	12	23	14	$6/root$	$8/root$	$5/root$	121	134

Including overhead, the two and three view functions take $110 - 140\mu s$ and $120 - 180\mu s$, respectively. Hypothesis generation for RANSAC with 500 samples takes $60ms$ and $67ms$, respectively.

yield a unique solution, with or without planar structure and an unknown focal length common to the three views.

6.3 Timing

Timing information for our efficient implementation of the five-point algorithm is given in Table 3. The algorithm is used as a part of a system that reconstructs structure and motion from video in real-time. System timing information is given in Table 4. MMX code was used for the crucial parts of the feature detection and feature matching. In the structure and motion component (SaM), one-view and three-view estimations are combined to incrementally build the reconstruction with low latency. The whole system including all overhead currently operates at 26 frames per second on average on a 2.4GHz machine when using a 3 percent disparity range. The latency is also small since there is no self-calibration and only very local iterative refinements.

6.4 Performance under Noise

In this section, the performance of the 5-point method in noisy conditions will be studied and compared to that of the well-known 8 and 7-point methods and a 6-point scheme. These methods are the most prominent algebraic solutions for relative orientation of two perspective views with finite baseline. The names of the methods refer to the smallest number of point correspondences for which they can operate and give a finite number of possible solutions. The 5 and 6-point methods require the intrinsics to be known while the 7 and 8-point methods can operate without this knowledge.

According to [10], the linear 8-point method goes at least as far back as [47]. It was introduced to the computer vision community by [20] and defended in [13]. It yields a unique solution. The fact that seven point correspondences determine relative orientation up to at most three solutions has been known at least since [39]. An account of this is given in [22]. A description of the modern 7-point algorithm can be found, e.g., in [14]. It has been used for RANSAC in, e.g., [41], [25]. The 6-point algorithm gives a unique solution and was presented in [32].

TABLE 4
Approximate Average Timings per 720×240 Frame of Video for the System Components on a Modest 550MHz Machine

Feature Detection	Matching with Disparity Range			SaM
	3%	5%	10%	
$30ms$	$34ms$	$45ms$	$160ms$	$50ms$

Disparity range for the matching is given in percent of the image dimensions.

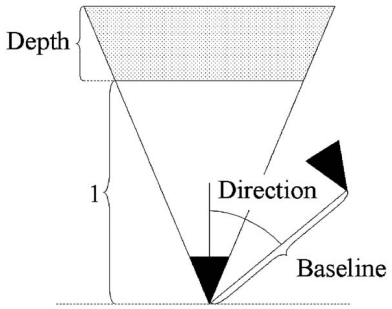


Fig. 7. The parameters of the test geometry used in the experiments. The distance to the scene volume is used as the unit of measure. The depth of the volume in which scene points are randomized is varied, as is the length of the baseline, the direction of motion, the amount of image noise, and the number of points.

Recall (10). In the 8-point method, the data is assumed to be strong enough that $x = y = z = 0$. Thus, W is extracted directly and we are done. In the 7-point method, it is assumed that $x = y = 0$. Thus, Z and W are extracted and by inserting (10) into (5), a cubic equation in z is obtained. Up to three solutions can be computed in closed form and we are done. In the 6-point method, it is assumed that $x = 0$. Thus, Y, Z, and W are extracted and insertion of (10) into the nine cubic constraints (6) yields the equation system

$$A\left[y^3\ y^2z\ yz^2\ z^3\ y^2\ yz\ z^2\ y\ z\ 1\right]^\top = 0, \tag{30}$$

where A is a 9×10 matrix. The right nullvector v of A is extracted by QR-decomposition or SVD. We are done by observing that $y = v_8/v_{10}$ and $z = v_9/v_{10}$.

As discussed in [13], coordinate system normalization is crucial for the performance of these algorithms in all but the minimal cases. We work in the calibrated coordinate system in all tests. The algorithms are mainly tested as calibrated algorithms, i.e., we compare their ability to determine translation direction and rotation when the calibration is known. The SVD of the essential matrix is used to determine the rotation and translation as proposed in [46]. This step also enforces the constraints (6).

To get quantitative results, we use experiments on synthetic data. The test geometry and its parameters are shown in Fig. 7. The distance to the scene volume is used as the unit of measure. The depth of the volume in which scene points are randomized is varied, as is the length of the baseline, the direction of motion, the amount of image noise, and the number of points. We will primarily cite results for challenging while realistic conditions. Unless otherwise noted, the parameters are as shown in Table 5. We will concentrate on the deviation of the estimated translation direction from the true value as a function of the level of noise

TABLE 5
The Challenging while Realistic Default Parameters Used in the Experiments

Depth	0.5
Baseline	0.1
Image Dimensions	352 × 288(CIF)
Noise Std-dev	1 Pixel
Field of View	45 Degrees

in the image correspondences. The reason is that the estimates of translation direction are much more sensitive than the estimates of rotation, which is widely known, see, e.g., [40]. When rotational errors are cited, they are given as the smallest angle of rotation that can bring the estimate to the true value. When referring to minimal cases, the minimal number of points necessary to get a unique solution is intended (i.e., 6, 6, 8, 8 for the 5, 6, 7, and 8-point methods, respectively). For the minimal cases, the lower quartile of the error distribution will be used in the plots. The robustness of this measure is relevant since in RANSAC it is more important to find a fraction of good hypotheses than to get consistent results. In the least-squares cases, the average of the error distribution will be used since occasional gross errors are less tolerable and should be penalized. The plots for minimal cases are based upon 1,000 trials per data point, while the least-squares plots are based upon 100 trials. In the plots for least-squares cases, the methods "5R" and "7R" are included. These methods consist of RANSAC with a robust Bayesian cost function, (compare, e.g., [41], [25], [35]) based on 5 and 7-point sampling, respectively, followed by iterative refinement. In the iterative refinement for "5R," the intrinsic constraints are enforced, while for "7R" they are not. These results are included in order to give an idea of how well it is possible to do with and without calibration knowledge. Note, however, that these methods may not always converge to the global minimum. Moreover, the robust cost function is not strictly necessary for synthetic outlier-free data. Thus, the least-squares results are occasionally better.

Fig. 8 shows results for the minimal cases under easy conditions, i.e., with large baseline and scene depth. Results are given for both sideways motion and forward motion. The trend that the 5-point method outperforms the other schemes for sideways motion is already visible, but all

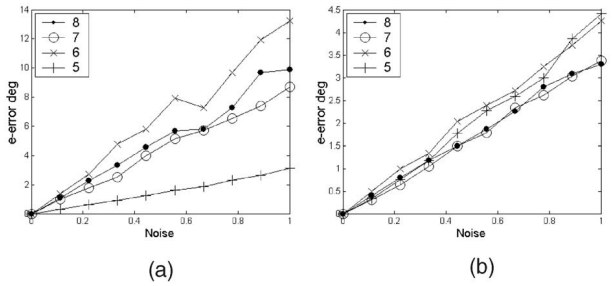


Fig. 8. Translational error in degrees against noise standard deviation in pixels of a CIF image. Minimal cases, easy conditions (Depth = 2, Baseline = 0.3). (a) Sideways motion. (b) Forward motion. The trend that the 5-point method outperforms the other schemes for sideways motion is already visible.

methods perform well under these easy conditions, which corroborates the findings of other authors. See, e.g., the related discussion in [31]. Fig. 9 shows results under more challenging conditions, both for the minimal cases and with 100 points. The 5-point method significantly outperforms the other noniterative methods for sideways motion. The 5-point results are quite good, while the results of the other noniterative methods are virtually useless. As the noise grows large, the other methods are swamped and begin placing the epipole somewhere inside the image regardless of its true position. This phenomenon has been observed previously by, e.g., [15]. It is particularly marked for the 6 and 8-point methods. For the forward motion cases, the results are quite different however. This is partly due to a slight deterioration of the results for the 5-point method, but mainly due to a vast improvement of the results for the other methods. In particular, the 8-point method gives excellent results on forward motion. Note that, in the useful regions, the 6 and 7-point methods keep up with the 8-point method for RANSAC purposes, but not for least-squares purposes.

For completeness, rotational errors are shown in Fig. 10. Perhaps somewhat surprisingly, we find that the 5-point method is weaker than the other methods for determining rotation. Observe, however, that the scale of the rotational errors is much smaller than the errors in translation direction, corresponding to hundredths of a degree and subpixel precision. Thus, the results of all methods are more than acceptable and in the sequel, we will concentrate on errors in the estimated translation direction.

The results with respect to a varying number of points are shown in Fig. 11. Again, the 5-point method is outstanding for

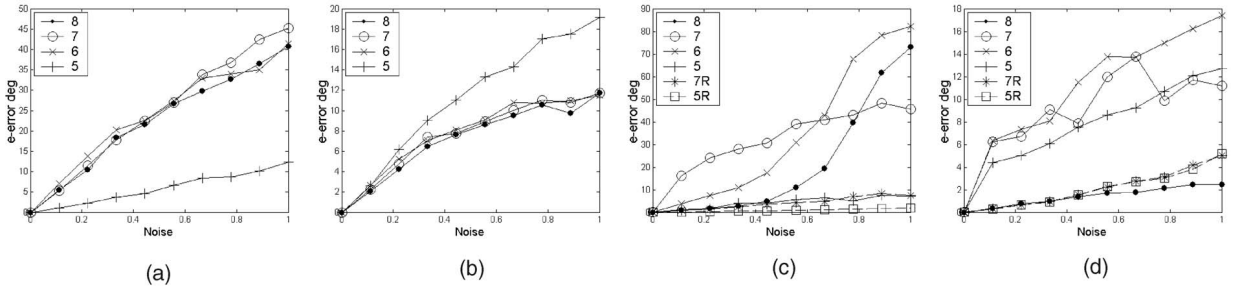


Fig. 9. Translational error against noise under the more challenging default conditions of Table 5. (a) Minimal cases, sideways motion, (b) minimal cases, forward motion, (c) 100 points, sideways motion, and (d) 100 points, forward motion. The 5-point method significantly outperforms the other noniterative methods for sideways motion.

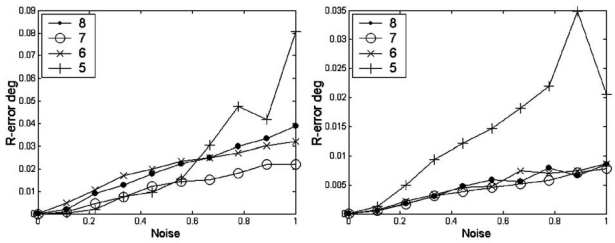


Fig. 10. Rotational error in degrees at varying levels of noise. Minimal cases. (a) Sideways motion. (b) Forward motion. Observe that the scale of the rotational errors is much smaller than the errors in translation direction, corresponding to subpixel precision. Thus, the results of all methods are more than acceptable.

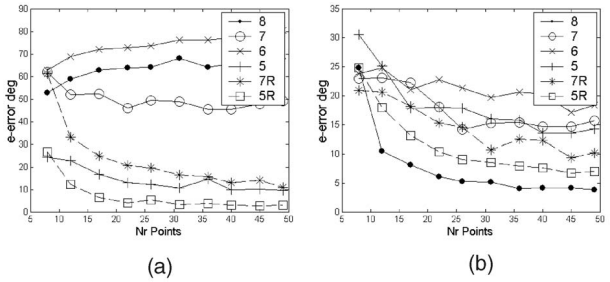


Fig. 11. Translational error in degrees with varying number of points. (a) Sideways motion. (b) Forward motion. The 5-point method is outstanding for sideways motion, rivaled only by the iterative methods. The absolute results for the 5-point method show the most consistent overall results.

sideways motion, rivaled only by the iterative methods. The sideways results for the 6, 7, and 8-point methods are useless and neither is able to make efficient use of the additional points. Again, their results for forward motion are greatly improved, headed by the 8-point method. Note that the absolute results for the 5-point method are almost the same for both cases, showing the most consistent overall results. This is seen even more clearly in Fig. 12, where the performance for varying direction of motion is investigated. The results for the 5-point method and the iterative methods exhibit a satisfying consistency over all directions, although one might consider combining the minimal 5-point method with one of the other methods for forward motion in challenging conditions. Meanwhile, the results for the other methods are acceptable for minimal cases in easy conditions, but unacceptable for sideways motion in challenging conditions, both for minimal cases and with many points. As already mentioned, when overwhelmed by noise, these methods are prone to incorrectly selecting an epipole estimate somewhere inside the image, where it is close to the actual point correspondences.

In Fig. 13, the performance for various magnitudes of motion is investigated. Again, the 5-point method performs well for sideways motion and similarly for forward motion although slightly worse for forward motion with very small baseline. The sideways results for the other noniterative methods are not useable unless the baseline is quite long. Their results are much better for forward motion. The 8-point method outperforms the 6 and 7-point methods for least-squares purposes, but all three are very similar in the minimal cases. Fig. 14 depicts results for shallow scenes. Theory says that for perfectly planar scenes there are up to two solutions for relative orientation in the calibrated setting, but a two-dimensional family of solutions in the uncalibrated framework. In the uncalibrated framework, the

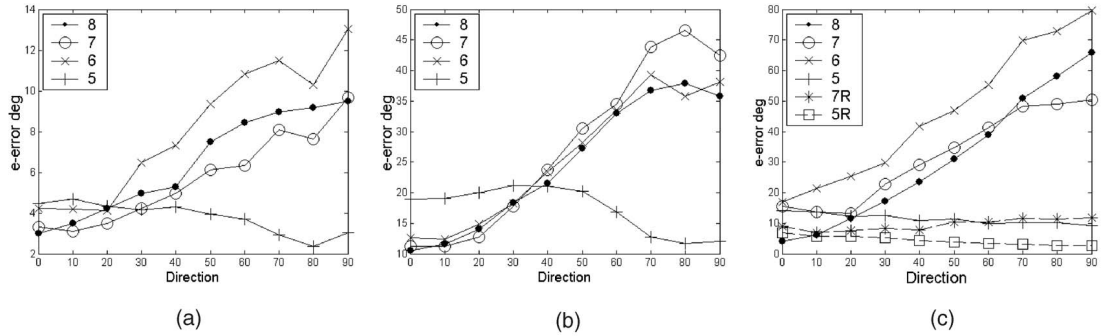


Fig. 12. Translational error at varying directions of motion given in degrees from the forward direction. (a) Minimal cases, easy conditions (Depth = 2, Baseline = 0.3), (b) minimal cases, default conditions, and (c) 50 points, default conditions.

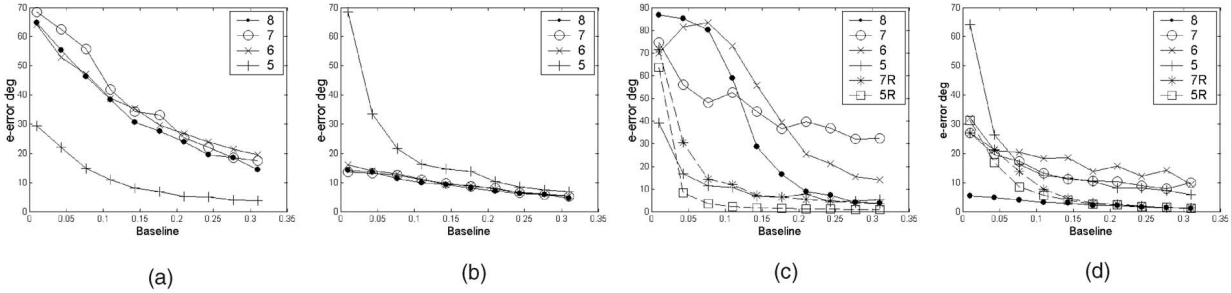


Fig. 13. Translational error at varying magnitudes of motion. (a) Minimal cases, sideways motion, (b) minimal cases, forward motion, (c) 50 points, sideways motion, and (d) 50 points, forward motion.

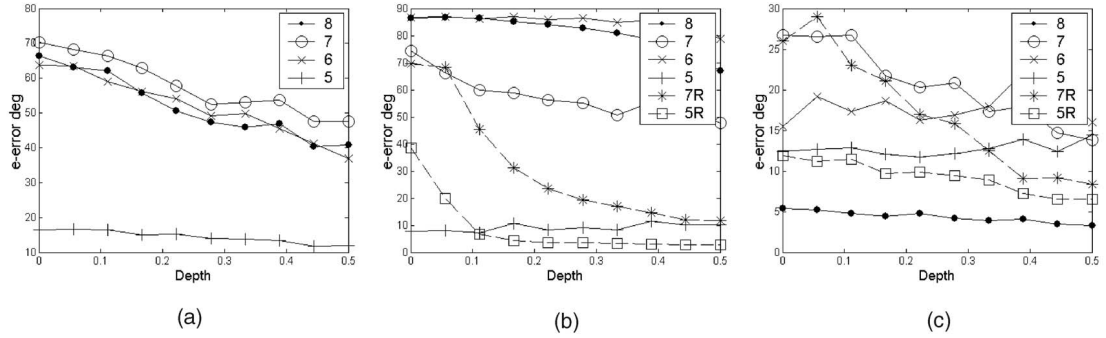


Fig. 14. Translational error for small scene depths. (a) Minimal cases, sideways motion, (b) 50 points, sideways motion, and (c) 50 points, forward motion. For sideways motion, the 5-point method is the only noniterative method of any use.

translation direction is not resolvable. See Section 4 for further details and references. This is also reflected in practice for sideways motion, where the 5-point method is the only noniterative method of any use. The reader may wonder why the 6-point method, which is essentially a calibrated method, is not performing better. The answer is that it makes suboptimal use of the intrinsic constraints and does not enforce them fully. It is therefore degenerate for planar scenes in much the same way as the uncalibrated methods [33]. Note how, in the face of the ambiguity, the 6 and 8-point methods default the position to the center of the image, yielding the worst possible results. It is almost tautological that the results are better for forward motion.

One may now wonder how accurate the calibration knowledge has to be in order to be useful. A part of the answer is given in Fig. 15, where we investigate how an imprecise focal length affects the estimate of the epipole. The rotations were randomized. Note how the uncalibrated methods are unaffected, while the calibrated methods do better for the true focal length. With accurate calibration, the 5-point method is superior. Under the default conditions, the only way to obtain acceptable results is to use the 5-point method with accurate calibration. The figure suggests that to get any benefits, the calibration should be at least within 10 percent and that significant benefits are reaped with higher accuracy. With inaccurate calibration, the results are impaired so that the uncalibrated methods do better under easy conditions.

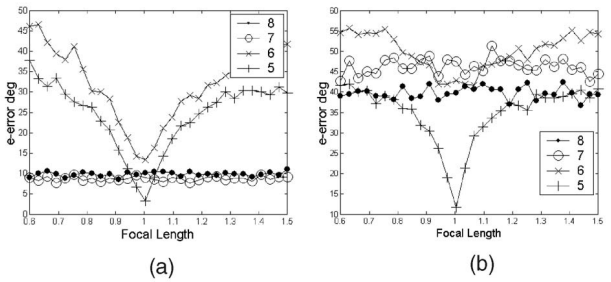


Fig. 15. Translational error with varying miscalibration of the focal length. The presumed focal length is given as a portion of its true value. Minimal cases, sideways motion. (a) Easy conditions. (b) Default conditions. Here, the only way to get acceptable results is to use the 5-point method and have good calibration.

6.5 System Results

So far, we have concentrated on synthetic data in order to get a quantitative evaluation of the 5-point method. In Fig. 16, a real result is shown. The intrinsic parameters of the camera were calibrated with the method of Zhang [49]. The reconstruction exhibits a regularity and accuracy that is typically not obtained with an uncalibrated method until the calibration constraints have been enforced through a global bundle adjustment. Observe that our default test parameters correspond well to the geometry in this example in terms of baseline and scene depth. With the sideways motion, it is right in the domain where we have just shown that the 5-point method gives crucial accuracy improvements. Fig. 17 shows how this result changes with an incorrectly assumed focal length.

The 5-point method is used in a real-time system for estimation of camera motion in video sequences. This is enabled by fast feature tracking, the efficient 5-point method and the preemptive scoring. A result is shown in Fig. 18. In the past, camera motion was successfully estimated with an uncalibrated method in sections of this sequence [25], but never in one piece or in real-time. Note that, especially, the outer circle is in a domain where our results suggest that when using neighboring frames, uncalibrated methods stab blindly in the dark for the translation direction. They would be saved only by the redundancy of RANSAC and the improvement from subsequent iterative refinement. Thus, selecting baselines carefully becomes imperative. Careful selection of baselines is always important, but is out of the scope of this paper, instead the reader is referred to [24]. Another example is shown in Fig. 19. A practical example of successful reconstruction in the face of planar degeneracy is shown in Fig. 20. Only approximate intrinsic parameters were used and no global bundle adjustment was performed.

Fig. 21 shows a result of estimation that was done in real-time from a camcorder tape playing back through the PC capture card. Thus, the system has to deal with capture, dropping frames if not able to keep up, etc. The estimation was made with a graphical feature-track window and a 3D scene window displaying. In this mode, the delay from the tape to the reconstruction displaying on the screen is less than a second. A result from live estimation is shown in Fig. 23. No knowledge of the motion was used in either case and the system parameters were identical. This shows that the system can handle forward as well as sideways motion. The system has been demonstrated live at major conferences [26], [27], [28], [29]. The system has been used to process video from a variety of platforms, including handheld video, video from

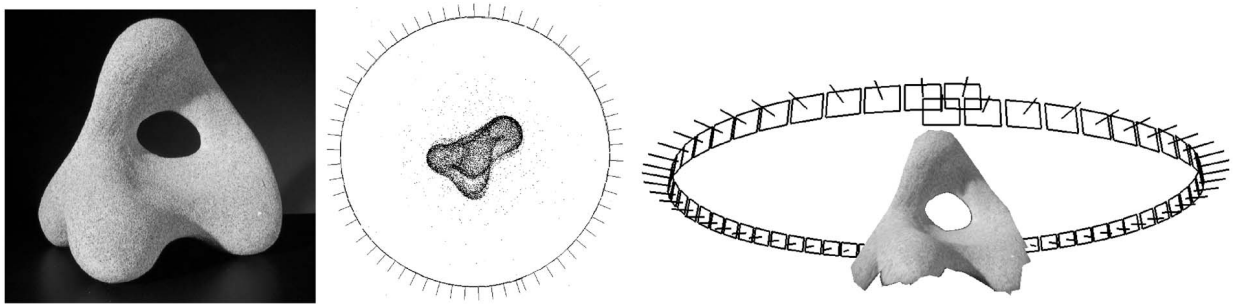


Fig. 16. Result from the turntable sequence "Stone." No prior knowledge about the motion or that it closes on itself was used in the estimation. The circular shape of the estimated trajectory is a verification of the correctness of the reconstruction that was made with low delay and bundle adjustment only for groups of three views.

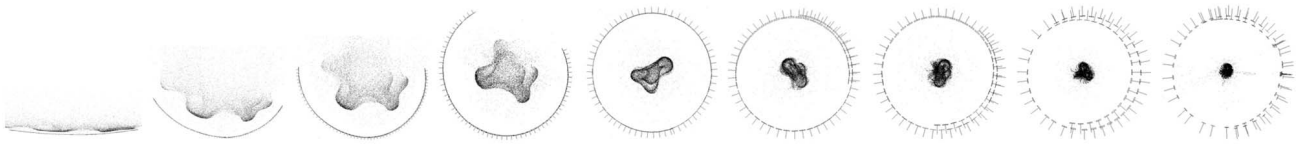


Fig. 17. Reconstructions obtained from the "Stone" sequence by setting the focal length to incorrect values. The focal lengths used were 0.05, 0.3, 0.5, 0.7, 1.0, 1.3, 1.5, 2.0, and 3.0 times the value obtained from calibration. For too small focal lengths, the reconstruction "unfolds" and vice versa.

automotive, ground vehicle and robotics applications, and aerial video. An example with aerial video is shown in Fig. 22.

7 Summary and Conclusions

An efficient algorithm for solving the five-point relative pose problem was presented. The algorithm was used in conjunction with random sampling consensus to solve for unknown structure and motion over two, three or more views. The efficiency of the algorithm is very important since it will typically be applied within this kind of hypothesize-and-test architecture, where the algorithm is executed for hundreds of different five-point samples. Practical live and in real-time reconstruction results were given and it was shown that the calibrated framework can continue to operate correctly despite scene planarity.

The performance of the 5-point algorithm was compared to the performance of the well-known 8 and 7-point methods and a 6-point scheme. It was shown quantitatively that there are realistic conditions under which the 5-point method can operate successfully while the other nonitera-tive methods fail. The results indicate that a combination of the 5-point method and the 8-point method may be a good option, since the 5-point method clearly outperforms the other methods for sideways motion and the 8-point method does best for forward motion. If one had to pick a winner overall between the noniterative methods, it would have to be the 5-point algorithm, which performs acceptably in both cases, while the 8-point method performs very well on forward but very poorly on sideways motion.

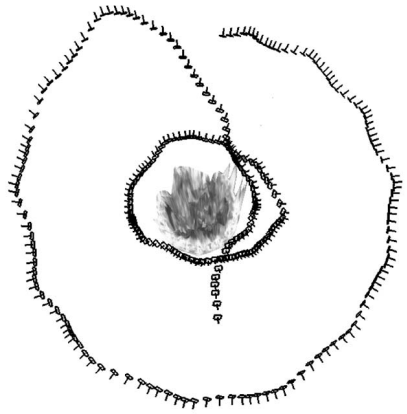


Fig. 18. Reconstruction result made at real-time rate from the sequence "Flowerpot" taken with a hand-held camera. Only approximate intrinsic parameters were used and no global bundle adjustment was performed. The handheld camera first moves in an outer circle and then an inner circle, with some forward motion in between.

Appendix A

Definition of Sturm Chain

Let $p(z)$ be a general polynomial of degree $n >= 2$. Here, the significance of general is that we ignore special cases for the sake of brevity. For example, $p(z)$ is assumed to have no multiple roots. Moreover, the polynomial divisions carried out below are assumed to have a nonzero remainder. Under these assumptions, the Sturm chain is a sequence of

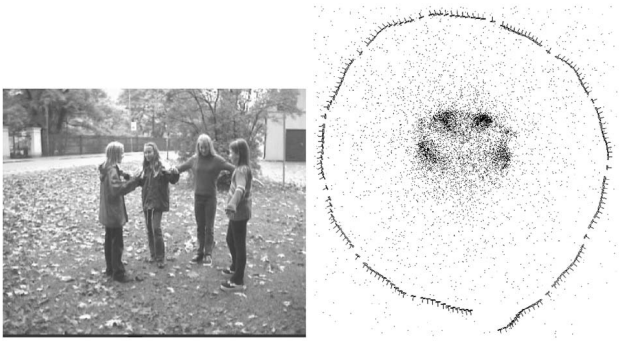


Fig. 19. Reconstruction from the sequence "Girlsstatue" that was acquired with a handheld camera. Only approximate intrinsic parameters were used and no global bundle adjustment was performed.

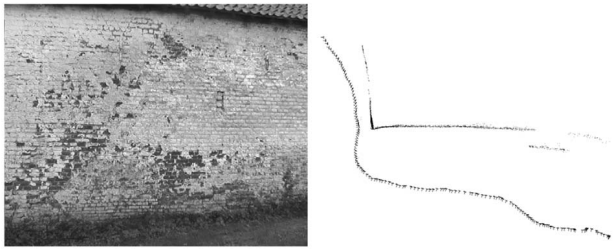


Fig. 20. Reconstruction from the sequence "Farmhouse," which contains long portions where a single plane fills the field of view. The successful reconstruction is a strong practical proof of the fact that the calibrated framework can overcome planar structure degeneracy without relying on the degeneracy or trying to detect it.

polynomials $f_0, \ldots, f_n$ of degrees $0, \ldots, n$, respectively. f_n is the polynomial itself and f_{n-1} is its derivative:

$$f_n(z) \equiv p(z) \tag{31}$$

$$f_{n-1}(z) \equiv p'(z). \tag{32}$$

For $i = n, \ldots, 2$, we carry out the polynomial division f_i / f_{i-1}. Let the quotient of this division be $q_i(z) = k_i z + m_i$ and let the remainder be $r_i(z)$, i.e., $f_i(z) = q_i(z) f_{i-1}(z) + r_i(z)$. Then, define $f_{i-2}(z) \equiv -r_i(z)$. Finally, define the coefficients m_0, m_1, and k_1 such that

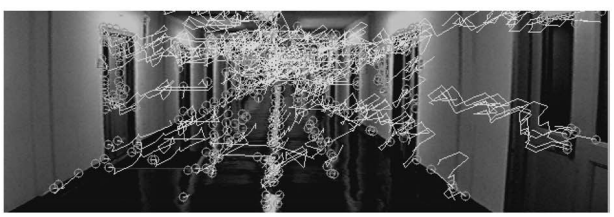
$$f_0(z) = m_0 \tag{33}$$

$$f_1(z) = k_1 z + m_1. \tag{34}$$

Once the scalar coefficients $k_1, \ldots, k_n$ and $m_0, \ldots, m_n$ have been derived, the Sturm chain can be evaluated at any point z through (33) and (34) and the recursion

$$f_i(z) = (k_i z + m_i) f_{i-1}(z) - f_{i-2}(z) \quad i = 2, \ldots, n. \tag{35}$$

Let the number of sign changes in the chain be $s(z)$. The number of real roots in an interval $[a, b]$ is then $s(a) - s(b)$. Unbounded intervals such as, for example, $[0, \infty)$, can be treated by looking at m_0 and $k_0, \ldots, k_n$ in order to calculate $\lim_{z \to \infty} s(z)$. See also [17], which, however, does not use the more efficient recursive formulation.



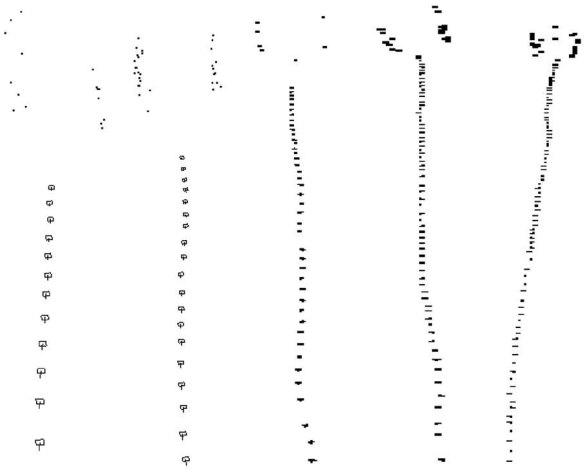


Fig. 21. Estimation of the motion of a handheld camera, induced by walking down a 160 meters long corridor. The whole motion is successfully integrated into the same coordinate frame. Note how the straight trajectory builds up, with different stages shown from left to right.

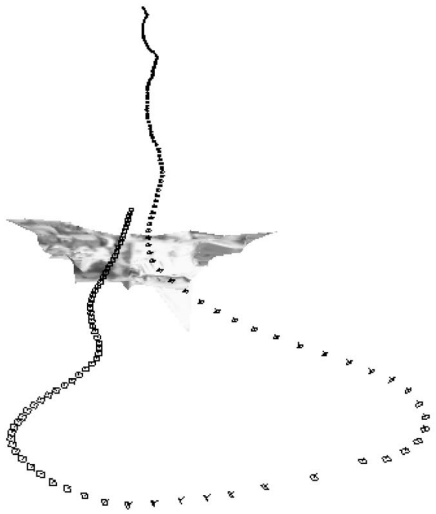


Fig. 22. Reconstruction from an aerial video that was processed in real-time without any use of external positioning data or assumptions on the motion. The plane first flies in a straight line and then makes a cul-de-sac shaped turn and returns.

APPENDIX B

EFFICIENT SINGULAR VALUE DECOMPOSITION OF THE ESSENTIAL MATRIX

An efficient, singular value decomposition according to the conditions of Theorem 3 is given. Let the essential matrix be $E = [\, e_a \quad e_b \quad e_c \,]^\top$, where e_a, e_b, e_c are column-vectors. It is assumed that it is a true essential matrix, i.e., that it has rank two and two equal nonzero singular values. First, all the vector products $e_a \times e_b$, $e_a \times e_c$, and $e_b \times e_c$ are computed and the one with the largest magnitude chosen. Assume without loss of generality that $e_a \times e_b$ has the largest magnitude. Define $v_c \equiv (e_a \times e_b)/|e_a \times e_b|$, $v_a \equiv e_a/|e_a|$, $v_b \equiv v_c \times v_a$, $u_a \equiv Ev_a/|Ev_a|$, $u_b \equiv Ev_b/|Ev_b|$, and $u_c \equiv u_a \times u_b$. Then, the singular value decomposition is given by $V = [\, v_a \quad v_b \quad v_c \,]$ and $U = [\, u_a \quad u_b \quad u_c \,]$.

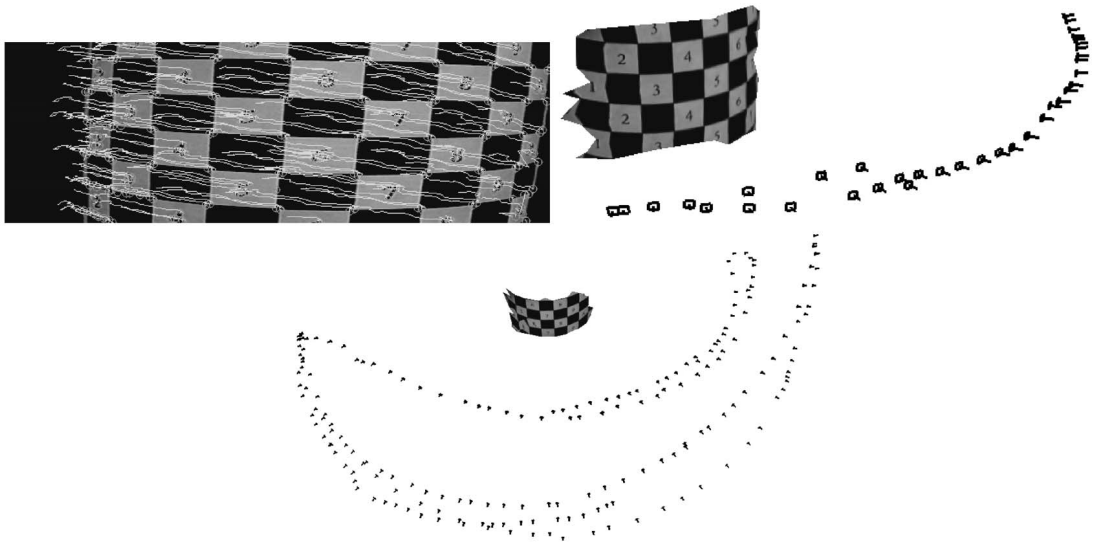


Fig. 23. Estimation of a freehand motion created with a calibration cylinder in one hand and a camcorder in the other. The estimation was done live. No knowledge of the structure or motion was used and the system is identical to the one used in Fig. 21.

APPENDIX C

EFFICIENT TRIANGULATION OF AN IDEAL POINT CORRESPONDENCE

In the situation encountered in the five-point algorithm where triangulation is needed, a hypothesis for the essential matrix E has been recovered and along with it the two camera matrices $[I \mid 0]$ and P. No error metric has to be minimized since, for the true solution, the rays backprojected from the image correspondence $q \leftrightarrow q'$ are guaranteed to meet. For nonideal point correspondences, prior correction to guarantee ray-intersection while minimizing a good error metric is recommended. Global minimization of $\| \, . \, \|_2$ -norm in two views requires solving a sixth degree polynomial, see [14]. Minimization of $\| \, . \, \|_\infty$ -norm [25], or directional error [30], also yields good results in practice and can be achieved in closed form an order of magnitude faster. In the ideal situation, triangulation can be accomplished very efficiently by intersecting three planes that are back-projected from image lines. The image lines chosen to generate the three planes are the epipolar line a corresponding to q', the line b through q that is perpendicular to a and the line c through q' that is perpendicular to Eq. For nonideal point correspondences, this scheme finds the world point on the ray backprojected from q' that minimizes the reprojection error in the first image. It triangulates world points at infinity correctly and is invariant to projective transformations of the world space. Observe that $a = E^\top q'$, $b = q \times (diag(1,1,0)a)$ and $c = q' \times (diag(1,1,0)Eq)$. Moreover, $A \equiv [\, a^\top \quad 0 \,]^\top$ is the plane backprojected from a, $B \equiv [\, b^\top \quad 0 \,]^\top$ is the plane backprojected from b and $C \equiv P^\top c$ is the plane backprojected from c. The intersection between the three planes A, B, and C is now sought. Formally, the intersection is the contraction $Q_l \equiv \epsilon_{ijkl} A^i B^j C^k$ between the epsilon tensor ϵ_{ijkl}[4] and the three planes. More concretely, $d \equiv a \times b$ is the direction of the ray backprojected from the intersection between a and b. The space point is the intersection between this ray and the plane C:

$$Q \sim \begin{bmatrix} d^\top C_4 & -(d_1 C_1 + d_2 C_2 + d_3 C_3) \end{bmatrix}^\top. \tag{36}$$

Finally, it is observed that, in the particular case of an ideal point correspondence, we have $d = q$, so that computing a, b and A, B can be avoided altogether.

APPENDIX D

TRIFOCAL SAMPSON APPROXIMATION IN CLOSED FORM

To score motion hypotheses over three views, we use a Sampson approximation [14], [42], [25] of the minimum image perturbation required to bring a triplet of points to trifocal incidence. A minimal closed form expression is essential for real-time performance. Since no such expression has been given in the literature we present one here. The Sampson approximation is defined by a vector-valued function g that is zero for trifocal incidence. The key to obtaining an efficient expression is selecting a vector function with the minimal number of dimensions, which is three. We use a carefully chosen combination of one bilinearity and two trilinearities. Let a, b, and c be homogeneous coordinate representations of the observed points in view 1, 2, and 3, respectively. Let F be the fundamental matrix between view 1 and 3 and let T_i^{jk} be the trifocal tensor that takes a point a^i in view 1, a line w_j in view 2, and a line z_k in view 3. For trifocal incidence, we then have

$$c^\top F a = 0 \tag{37}$$

$$a^i w_j z_k T_i^{jk} = 0. \tag{38}$$

The line Fa is the epipolar line of a in the third view. The point $d = diag(1,1,0)Fa$ is the point at infinity in the direction perpendicular to Fa. Let $\tilde{a}$, $\tilde{b}$, and $\tilde{c}$ be the perturbed points, represented in homogeneous coordinates with unit final coordinate. The line $z_\perp(\tilde{c}) = \tilde{c} \times d$ is the line through $\tilde{c}$ perpendicular to Fa. Moreover, define the horizontal line $w_{\_}(\tilde{b}) = \begin{bmatrix} 0 & 1 & -\tilde{b}_2 \end{bmatrix}^\top$ and the vertical line

4. The epsilon tensor ϵ_{ijkl} is the tensor such that $\epsilon_{ijkl} A^i B^j C^k D^l = det([\, A \quad B \quad C \quad D \,])$.

$w_|(\tilde{b}) = \begin{bmatrix} 1 & 0 & -\tilde{b}_1 \end{bmatrix}^\top$ through $\tilde{b}$. Then, our Sampson function is

$$g(\tilde{a}, \tilde{b}, \tilde{c}, d, F, T) = \begin{bmatrix} \tilde{c}^\top F \tilde{a} \\ \tilde{a}^i w_-(\tilde{b})_j z_\perp(\tilde{c})_k T_i^{jk} \\ \tilde{a}^i w_|(\tilde{b})_j z_\perp(\tilde{c})_k T_i^{jk} \end{bmatrix}. \quad (39)$$

The 3×6 derivative matrix J of g with respect to $(\tilde{a}_1, \tilde{a}_2, \tilde{b}_1, \tilde{b}_2, \tilde{c}_1, \tilde{c}_2)$ is computed and the squared Sampson error is

$$g^\top (JJ^\top)^{-1} g, \quad (40)$$

which is computed at $(\tilde{a}, \tilde{b}, \tilde{c}) = (a, b, c)$ by tensor contractions and LU-decomposition, Cholesky factorization, or even Cramers rule for speed. Assuming a Cauchy distribution for the reprojection errors, the robust likelihood contribution is

$$\rho = ln\left(1 + \frac{g^\top (JJ^\top)^{-1} g}{\sigma^2}\right), \quad (41)$$

where σ is a scale parameter.

ACKNOWLEDGMENTS

The author would like to thank Frederik Schaffalitzky for bringing to his attention that (5) contributes a constraint that is, in general, linearly independent of the constraints obtained from (6), despite the algebraic dependency manifested in Theorem 2.

This paper was prepared through a collaborative participation in the Robotics Consortium sponsored by the US Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0012. The US Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory or the US government.

REFERENCES

[1] P. Beardsley, A. Zisserman, and D. Murray, "Sequential Updating of Projective and Affine Structure from Motion," *Int'l J. Computer Vision,* vol. 23, no. 3, pp. 235-259, 1997.

[2] O. Chum and J. Matas, "Randomized RANSAC with $T_{d,d}$ Test," *Proc. British Machine Vision Conf.,* pp. 448-457, 2002.

[3] M. Demazure, "Sur Deux Problemes de Reconstruction," Technical Report No. 882, INRIA, France, 1988.

[4] O. Faugeras and S. Maybank, "Motion from Point Matches: Multiplicity of Solutions," *Int'l J. Computer Vision,* vol. 4, no. 3, pp. 225-246, 1990.

[5] O. Faugeras, "What Can Be Seen in Three Dimensions with an Uncalibrated Stereo Rig?" *Proc. European Conf. Computer Vision,* pp. 563-578, 1992.

[6] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint.* MIT Press, 1993.

[7] A. Fitzgibbon and A. Zisserman, "Automatic Camera Recovery for Closed or Open Image Sequences," *Proc. European Conf. Computer Vision,* pp. 311-326, 1998.

[8] M. Fischler and R. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography," *Comm. ACM,* vol. 24, pp. 381-395, 1981.

[9] W. Gellert, K. Küstner, M. Hellwich, and H. Kastner, *The VNR Concise Encyclopedia of Mathematics.* Van Nostrand Reinhold Company, 1975.

[10] A. Gruen and T. Huang, *Calibration and Orientation of Cameras in Computer Vision.* Springer Verlag, 2001.

[11] R. Haralick, C. Lee, K. Ottenberg, and M. Nölle, "Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem," *Int'l J. Computer Vision,* vol. 13, no. 3, pp. 331-356, 1994.

[12] R. Hartley, "Estimation of Relative Camera Positions for Uncalibrated Cameras," *Proc. European Conf. Computer Vision,* pp. 579-587, 1992.

[13] R. Hartley, "In Defense of the Eight-Point Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 6, pp. 580-593, June 1997.

[14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge Univ. Press, 2000.

[15] D. Heeger and A. Jepson, "Subspace Methods for Recovering Rigid Motion," *Int'l J. Computer Vision,* vol. 7, no. 2, pp. 95-117, 1992.

[16] A. Heyden and G. Sparr, "Reconstruction from Calibrated Cameras–A New Proof of the Kruppa-Demazure Theorem," *J. Math. Imaging & Vision,* vol. 10, pp. 1-20, 1999.

[17] D. Hook and P. McAree, "Using Sturm Sequences to Bracket Real Roots of Polynomial Equations," *Graphic Gems I,* Academic Press, pp. 416-423, 1990.

[18] B. Horn, "Relative Orientation," *Int'l J. Computer Vision,* vol. 4, pp. 59-78, 1990.

[19] E. Kruppa, "Zur Ermittlung eines Objektes aus Zwei Perspektiven mit Innerer Orientierung," *Sitz.-Ber. Akad. Wiss., Wien, Math. Naturw. Kl., Abt. IIa.,* vol. 122, pp. 1939-1948, 1913.

[20] H. Longuet-Higgins, "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature,* vol. 293, no. 10, pp. 133-135, 1981.

[21] H. Longuet-Higgins, "The Reconstruction of a Plane Surface from Two Perspective Projections," *Proc. Royal Soc. London B,* vol. 277, pp. 399-410, 1986.

[22] S. Maybank, *Theory of Reconstruction from Image Motion.* Springer Verlag, 1993.

[23] S. Negahdaripour, "Closed-Form Relationship Between the Two Interpretations of a Moving Plane," *J. Optical Soc. of Am.,* vol. 7, no. 2, pp. 279-285, 1990.

[24] D. Nistér, "Reconstruction from Uncalibrated Sequences with a Hierarchy of Trifocal Tensors," *Proc. European Conf. Computer Vision,* pp. 649-663, 2000.

[25] D. Nistér, "Automatic Dense Reconstruction from Uncalibrated Video Sequences," PhD thesis, Royal Inst. of Technology KTH, Mar. 2001.

[26] D. Nistér, "Preemptive RANSAC for Live Structure and Motion Estimation," *Proc. Int'l Conf. Computer Vision,* pp. 199-206, 2003.

[27] D. Nistér, "An Efficient Solution to the Five-Point Relative Pose Problem," *Proc. Computer Vision and Pattern Recognition,* pp. 195-202, 2003.

[28] D. Nistér, "Live Structure and Motion Estimation," *Proc. Computer Vision and Pattern Recognition,* 2003.

[29] D. Nistér, "Live Ego-Motion Estimation," *Proc. Int'l Conf. Computer Vision,* 2003.

[30] J. Oliensis and Y. Genc, "New Algorithms for Two-Frame Structure from Motion," *Proc. Int'l Conf. Computer Vision,* pp. 737-744, 1999.

[31] J. Oliensis, "A Critique of Structure from Motion Algorithms," *Computer Vision and Image Understanding,* vol. 80, pp. 172-214, 2000.

[32] J. Philip, "A Non-Iterative Algorithm for Determining All Essential Matrices Corresponding to Five Point Pairs," *Photogrammetric Record,* vol. 15, no. 88, pp. 589-599, 1996.

[33] J. Philip, "Critical Point Configurations of the 5-, 6-, 7-, and 8-point Algorithms for Relative Orientation," TRITA-MAT-1998-MA-13, Feb. 1998.

[34] M. Pollefeys, R. Koch, and L. Van Gool, "Self-Calibration and Metric Reconstruction in Spite of Varying and Unknown Internal Camera Parameters, *Int'l J. Computer Vision,* vol. 32, no. 1, pp. 7-25, 1999.

[35] M. Pollefeys, F. Verbiest, and L. Van Gool, "Surviving Dominant Planes in Uncalibrated Structure and Motion Recovery," *Proc. European Conf. Computer Vision,* pp. 837-851, 2002.

[36] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C,* Cambridge Univ. Press, 1988.
[37] J. Semple and G. Kneebone, *Algebraic Projective Geometry.* Oxford Univ. Press, 1952.
[38] P. Stefanovic, "Relative Orientation-A New Approach," *I.T.C.J.,* vol. 3, pp. 417-448, 1973.
[39] R. Sturm, "Das Problem der Projektivität und seine Anwendung auf die Flächen Zweiten Grades," *Math. Annalen 1,* pp. 533-573, 1869.
[40] T. Tian, C. Tomasi, and D. Heeger, "Comparison of Approaches to Egomotion Computation," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 315-320, 1996.
[41] P. Torr and D. Murray, "The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix," *Int'l J. Computer Vision,* vol. 24, no. 3, pp. 271-300, 1997.
[42] P. Torr and A. Zisserman, "Robust Parameterization and Computation of the Trifocal Tensor," *Image and Vision Computing,* vol. 15, pp. 591-605, 1997.
[43] P. Torr, A. Fitzgibbon, and A. Zisserman, "The Problem of Degeneracy in Structure and Motion Recovery from Uncalibrated Image Sequences," *Int'l J. Computer Vision,* vol. 32, no. 1, pp. 27-44, 1999.
[44] B. Triggs, "Routines for Relative Pose of Two Calibrated Cameras from 5 Points," technical report, http://www.inrialpes.fr/movi/people/TriggsINRIA, France, 2000.
[45] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle Adjustment-A Modern Synthesis," *Lecture Notes in Computer Science,* vol. 1883, pp. 298-375, 2000.
[46] R. Tsai and T. Huang, "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 6, no. 1, pp. 13-27, 1984.
[47] H. von Sanden, "Die Bestimmung der Kernpunkte in der Photogrammetrie," PhD thesis, Univ. of Gottingen, 1908.
[48] Z. Zhang, "Determining the Epipolar Geometry and Its Uncertainty: A Review," *Int'l J. Computer Vision,* vol. 27, no. 2, pp. 161-195, 1998.
[49] Z. Zhang, "Flexible Camera Calibration by Viewing a Plane from Unknown Orientations," *Proc. Int'l Conf. Computer Vision,* pp. 666-673, 1999.

David Nistér received the MSc degree in computer science and engineering, specializing in applied mathematics, from Chalmers University of Technology, Gothenburg, Sweden, in 1997, and the Licentiate of Engineering degree in the area of image compression from Chalmers University of Technology in 1998. In 2001, he received the PhD degree in computer vision, numerical analysis, and computing science from the Royal Institute of Technology (KTH), Stockholm, Sweden, with the thesis "Automatic Dense Reconstruction from Uncalibrated Video Sequences." He is a researcher in the Vision Technologies Laboratory, Sarnoff Corporation, Princeton, New Jersey. Before joining Sarnoff, he worked at Visual Technology, Ericsson Research, Stockholm, Sweden, at Prosolvia Clarus, Gothenburg, Sweden, specializing in virtual reality and at Compression Lab, Ericsson Telecom. He has also served as a mathematical consultant at the Department of Physical Chemistry, Uppsala University, Sweden. His research interests include computer vision, computer graphics, structure from motion, multiple view geometry, Bayesian formulations, tracking, recognition, image, and video compression. He is a member of the IEEE, the IEEE Computer Society, and American Mensa.