Solutions to Minimal Generalized Relative Pose Problems

Henrik Stewénius, Magnus Oskarsson, Kalle Åström
Centre for Mathematical Sciences
Lund University, P.O. Box 118
S-221 00 Lund, SWEDEN

David Nistér
Department of Computer Science
Center for Visualization and Virtual Environments
University of Kentucky, Lexington, USA

Abstract

We present a method to obtain the solutions to the generalized 6-point relative pose problem. The problem is to find the relative positions of two generalized cameras so that six corresponding image rays meet in space. Here, a generalized camera is a camera that captures some arbitrary set of rays and does not adhere to the central perspective projection model. The cameras are assumed to be calibrated, which means that we know the image rays in Euclidean camera coordinate systems. Mathematically, the problem is therefore, given two Euclidean configurations consisting of six lines each, to find a rigid transformation of the first six lines so that each transformed line intersects its corresponding line from the second set.

We show that the problem has 64 solutions in general and solve it by computing a 64×64 *matrix on closed form and then extracting its eigen-vectors. Hence we present a solver that corresponds to the intrinsic degree of difficulty of this minimal problem.*

Our numerical experiments show that the presented solver can be used in a RANSAC-implementation.

1. Introduction

Solving for the relative positions of two images given corresponding points is one of the fundamental tasks in computer vision in general and in structure from motion in particular. Solutions exist for most cases arising with traditional camera models, such as calibrated perspective [6], uncalibrated perspective [12] and uncalibrated affine [13].

In the case of a calibrated perspective camera, Kruppa [14] showed that there are at most eleven solutions for two images and five points. Later is has been shown that there are only ten solutions. Solutions for such minimal cases have proven useful in practice as hypothesis generators in hypothesize-and-test algorithms such as RANSAC [7], where the solutions from a large number of randomly selected minimal point samples are robustly scored based on their support over the whole data set.

Recently, there has been intense interest in generalized cameras, i.e. non-central cameras that, due to their geome-

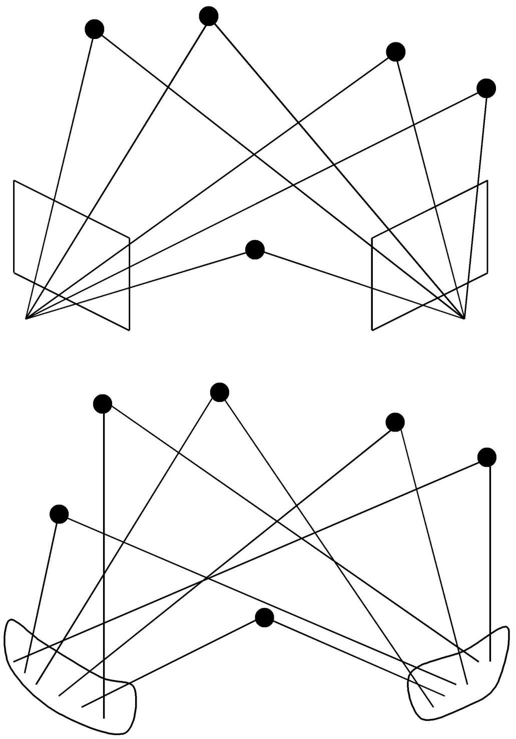

Figure 1: *Top: The minimal classical relative pose problem is to find the solutions for relative pose of two cameras so that five corresponding image rays meet in space. Bottom: We solve the minimal generalized problem, which is to find the solutions for relative pose of two generalized cameras given six corresponding points. The image rays of the generalized cameras do not go through a common projection center. It is therefore also possible to solve for the scale of the translation, which is the reason for the additional point. The generalized solution lets us deal with any of the recently very popular camera geometries that do not adhere to a central perspective model, such as for example a camera made up of several rigidly attached cameras moving together, or a camera facing an arbitrarily shaped mirror.*

try, do not have a projection center [2, 17, 15]. For example, we can consider a bug-eye camera or multi-camera rig of rigidly attached cameras moving together as a single camera. This lets us obtain larger coverage and potentially more stable motion estimation, but in order to achieve this, we have to use the generalized camera model. Another example of a generalized camera is an omni-directional camera created with a regular camera facing an arbitrarily shaped mirror. The image rays from such a camera are not in general concurrent in a single point.

The information about all light in a scene is embedded in the plenoptic function [1]. Extreme generality is obtained by just considering a generalized camera as a device that samples in an arbitrary fashion from the plenoptic function. We are able to work with this level of generality simply by assuming that we have a mapping from image points to rays in the camera coordinate system [9, 19].

Assuming that the camera coordinate system is Euclidean is equivalent to assuming a calibrated camera. In this manner we can relegate the specifics of the camera to a prior calibration procedure. Hence, the proposed method solves generalized relative pose for any generalized camera provided calibration is known.

Note that if one wanted to create a generalized analogue for uncalibrated relative pose, one almost inevitably would have to assume some parametric model for the unknowns and one then loses the ability to create a one-fits-all solution, one possible such alternative is the pushbroom camera [11].

However, there is another type of analogue to the calibrated case, which is to assume that the camera calibration is known up to an affine transformation. In Section 3 we indicate that this is what leads to the uncalibrated framework in the classical case. In the classical case this leads in general to simpler equations. We investigate the complexity of the affine formulation for the generalized case and find that for minimal problems it leads to very large numbers of solutions.

The minimal pose problem for generalized cameras has been solved [3, 16], as well as the minimal relative pose for pure planar motion [18]. Pless generalized the epipolar constraint in [17] and also investigated infinitesimal motion for generalized cameras.

However, to the best of our knowledge, none of the minimal relative orientation problems with finite motion of generalized cameras have previously been solved. We catalogue six different minimal generalized camera problems and solve the most feasible of the relative orientation problems.

The remainder of the paper is organized as follows: In Section 2 we introduce the generalized epipolar constraint, in Section 3 we catalogue the minimal cases. The solver is presented in Section 4.

2 Generalized Camera Model

Figure 1 shows the difference between classic and generalized cameras. In the classic camera, all rays pass through a common point, the camera center, while in the generalized camera this is not the case.

The conventional camera model has difficulties in treating problems involving systems of multiple cameras or cameras looking through a non-planar mirror. In such a case it is impossible to find a common camera center.

With a calibrated generalized camera we know how each image point corresponds to a unique line. The problem is posed using lines in 3D and they are here parameterized using Plücker coordinates. A Plücker vector is a six-vector

$$L = \begin{bmatrix} q \\ q' \end{bmatrix} \in \mathbb{P}^5,$$

consisting of two three-vectors q, q' where $q \cdot q' = 0$. Any point U on the line described by (q, q') satisfies

$$q' = q \times U. \tag{1}$$

The vector L is only defined up to scale, and if this scale is fixed by $||q|| = 1$, then all points U on the line described by (q, q') can be parameterized as

$$U(\lambda) = q' \times q + \lambda q\,,\ \lambda \in \mathbb{R}. \tag{2}$$

The image space of the generalized camera consists of rays (q, q') in the camera coordinate system. This model does not distinguish if the generalized camera is a multi-camera system or a camera viewing the world through a non-parabolic mirror. as long as the system is calibrated. Equation (1) is valid if the world point is given in the coordinate system of the camera; however, this is not often the case, and there is an unknown transformation between the world coordinate system and the camera coordinate system. If the transformation from the world coordinate system to the camera coordinate system is given by $T = (R, t)$ then for a point U_i in the world coordinate system equation (2) becomes

$$q'_{ij} \times q_{ij} + \lambda_{ij} q_{ij} = R_j U_i + t_j, \tag{3}$$

where i indexes over points and j over views.

In this paper the following problem will be studied,

Problem 2.1. *Given image rays*

$$\{(q, q')_{ij}\}, \quad i = 1, \ldots, n,\ j = 1, \ldots, m,$$

find cameras $\{T_j\}$, points $\{U_i\}$ and depths $\{\lambda_{ij}\}$ so that equation (3) holds.

In [17] Pless derives a generalized epipolar constraint, where the first camera position is set to $(I, 0)$. We will here derive the same constraint without this restriction.

We will now study the constraint that two corresponding image rays place on the camera geometry. For two image rays to be in correspondence, they must intersect in a point. Two rays in Plücker coordinates,

$$L_1 = \begin{bmatrix} q_1 \\ q'_1 \end{bmatrix} \quad \text{and} \quad L_2 = \begin{bmatrix} q_2 \\ q'_2 \end{bmatrix},$$

intersect if and only if

$$q_1^T q'_2 + q_2^T q'_1 = 0. \tag{4}$$

However, if these two lines are given in a local camera coordinate system they must first be transformed to the world coordinate system. In order to be consistent with Equation (3), points on the two lines will be tranformed by T_1^{-1} and T_2^{-1} respectively. Then the transformed line $\hat{L}_1$ is given by

$$\hat{L}_1 = \begin{bmatrix} \hat{q}_1 \\ \hat{q}'_1 \end{bmatrix} = \begin{bmatrix} R_1^T q_1 \\ R_1^T q'_1 + R_1^T [t_1]_\times q_1 \end{bmatrix},$$

and similarly for $\hat{L}_2$. Here $[v]_\times$ is the skew-symmetric cross-product matrix so that

$$[v]_\times x = v \times x.$$

Equation (4) then becomes

$$\begin{aligned} q_2^T R_2 R_1^T q'_1 + q_1^T R_1 R_2^T q'_2 + \\ + q_2^T (R_2 R_1^T [t_1]_\times - [t_2]_\times R_2 R_1^T) q_1 = 0. \end{aligned} \tag{5}$$

We will use quaternions to represent rotations. A quaternion is a four-vector $\begin{bmatrix} v_1 & v_2 & v_3 & s \end{bmatrix}^T$, which will be written $\begin{bmatrix} \mathbf{v}^T & s \end{bmatrix}^T$ where $\mathbf{v}$ is a three-vector. Quaternions can be used to parameterize rotations according to

$$\alpha R = 2(\mathbf{v}\mathbf{v}^T - s[\mathbf{v}]_\times) + (s^2 - \mathbf{v}^T\mathbf{v})\mathbf{I}, \tag{6}$$

where α is chosen so that $\det R = 1$ and $\mathbf{I}$ is the 3×3 identity matrix. If $\mathbf{v}^t\mathbf{v} + s^2 = 1$ then $\alpha = 1$ in Equation (6). Rescaling a quaternion will give a different α but the same R. Subsequent equations (8) are homogeneous in R, so the norm of R is unimportant. We will parameterize R by setting $s = 1$ in equation (6), i.e.

$$R \sim 2(\mathbf{v}\mathbf{v}^T - [\mathbf{v}]_\times) + (1 - \mathbf{v}^T\mathbf{v})\mathbf{I}. \tag{7}$$

3 Minimal Cases

We will here catalogue the minimal cases for pose and relative pose for generalized cameras. The key observation is that observing a ray in a generalized camera gives two equations and that a point in space has three degrees of freedom. For generalized pose and generalized relative pose we wish to solve Problem 2.1.

For the generalized relative pose problems that we have been unable to solve we have formulated analogous problems over $\mathbb{Z}_p$ where p is a large prime number and tested these in Macaulay 2 [8]. This gives an indication of what number of solutions to expect.

Relative Pose with a Euclidean Generalized Camera

Each camera has six degrees of freedom, but the world coordinate system can only be determined up to a rigid motion. With m camera positions and n rays observed in all images there are a total of $2mn - 3n - 6m + 6$ excess constraints.

Euclidean Generalized Camera

n

m	1	2	3	4	5	6	7	8
1	-1	-2	-3	-4	-5	-6	-7	-8
2	-5	-4	-3	-2	-1	**0**	1	2
3	-9	-6	-3	**0**	3	6	9	12
4	-13	-8	-3	2	7	12	17	22
5	-17	-10	-3	4	11	18	25	32

Minimal cases are when the number of equations equals the number of unknowns. In the table above there are two zeros and these are the only two minimal cases for this problem. Using the the algebraic geometry software Macaulay 2 [8], we compute the number of solutions for the analogous problem over $\mathbb{Z}_p$, this number usually corresponds to the number of solutions over $\mathbb{R}$. For $m = 2$ and $n = 6$ we get 64 solutions and for $m = 3$ and $n = 4$ we get 1320 solutions.

Relative Pose with an Affine Generalized Camera

By an affine generalized camera we mean a camera for which poses are affine transformation matrices. In this case there are $2mn - 3n - 12m + 12$ excess constraints.

Affine Generalized Camera

n

m	5	6	7	8	9	10	11	12
1	-5	-6	-7	-8	-9	-10	-11	-12
2	-7	-6	-5	-4	-3	-2	-1	**0**
3	-9	-6	-3	**0**	3	6	9	12
4	-11	-6	-1	4	9	14	19	24
5	-13	-6	1	8	15	22	29	36

There are two minimal cases

- Two positions and twelve rays for which the analogous problem over $\mathbb{Z}_p$ problem has 348 solutions.
- Three positions and eight rays. We expect the number of solutions to be very high as we have not been able to compute the number of solutions to the analogous problem.

Generalized Pose

In [16] Nist´er solves the pose problem for Euclidean generalized cameras.

We observe here that generalized pose for affine cameras is a linear problem with a unique solution in general. It is a direct generalization of the DLT for the classical case [12]. In the generalized case the problem is exactly minimal instead of overdetermined by one.

4 Solving Two Cameras and Six Rays

Theorem 1. *For two (calibrated) generalized cameras taking images of six labeled points in space there are in general 64 solutions, (not up to scale), to Problem 2.1.*

Proof. We will in the following give a solver which has this number of solutions with zero residual. □

The coordinate system can be chosen up to a rigid transformation, i.e. a rotation and a translation. The rotation is set by choosing $R_1 = \mathbf{I}$, the 3×3 identity matrix.

With $R_1 = I$, the generalized epipolar constraint (5) becomes

$$q_2^T R_2 q_1' + q_2^T (R_2 [t_1]_\times - [t_2]_\times R_2) q_1 + q_1^T R_2^T q_2' = 0. \quad (8)$$

Equation (8) is linear in the rotation R_2. This means that we can scale R_2 with any non-zero factor without violating (8) and use Equation (7).

4.1 Removing the Translation Parameters

In order to be able to eliminate the translation parameters t_1 and t_2 in (8) a special parameterization is used. This parameterization is asymmetric and by repeatedly choosing different asymmetries an increasing number of linearly independent equations can be obtained.

Until now only the rotation of the coordinate system has been set. By choosing one of the world points as the origin, i.e. $U_k = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ for some $k = 1, \ldots, 6$, the translational part of the coordinate system is set. Now, consider equation (3) for U_k

$$q'_{kj} \times q_{kj} + \lambda_{kj} q_{kj} = R_j \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + t_j. \quad (9)$$

This can be simplified to

$$t_j = q'_{kj} \times q_{kj} + \lambda_{kj} q_{kj} \qquad j = 1, 2. \quad (10)$$

We have used point k to express the six translation parameters, t_j, as linear expressions in the two unknown depth parameters $\{\lambda_{kj}\}_{j=1,2}$ and the measured (q_{kj}, q'_{kj}). Inserting the expressions for t_1 and t_2 in Equation (8) for the remaining points gives five equations in the unknown rotation R_2 (represented by the quaternion $\mathbf{v}$) and the two depths λ_{k1} and λ_{k2}. These equations are linear in λ_{k1} and λ_{k2} and can be rewritten

$$\underbrace{F^k(\mathbf{v})}_{5\times 3} \begin{bmatrix} \lambda_{k1} \\ \lambda_{k2} \\ 1 \end{bmatrix} = 0. \quad (11)$$

The entries in $F^k(\mathbf{v})$ are quadratic in $\mathbf{v}$, and depend otherwise only on measured image data.

Equation (11) has non-trivial solutions, and therefore $\mathrm{rank}(F^k(\mathbf{v})) \leq 2$, that is, all 3×3 submatrices of $F^k(\mathbf{v})$ must have determinant zero. There are ten such submatrices. This gives ten equations of degree six in the unknowns (v_1, v_2, v_3). This set of ten equations has two one-dimensional families of false roots.

Computing F^k and the resulting equations for several different values of $k = 1, \ldots, 6$, gives ten new equations for each value of k. The number of linearly independent equations will grow with the number n of different values of k used. For $n > 1$ the false roots are eliminated.

n	1	2	3	4	5	6
equations	10	20	30	40	50	60
lin. indep. eqs.	10	14	15	15	15	15
false roots	y	n	n	n	n	n

We chose to use $n = 3$ but all $n > 1$ work well. The order of elimination is a little more elegant for $n = 2$ but the numerical stability is a little better for $n = 3$ as is shown in Figure 6.

4.2 Creating a Gröbner-Basis

In the following steps we will create and use a Gröbner-basis. More information and theory on Gröbner-bases can be found in textbooks on Algebraic Geometry, for example [4, 5].

Our Gröbner-basis is built with respect to GrevLex order in order to avoid the heavy calculations necessary to build a lex-order base. When working with only up to 6th degree monomials, the 84 monomials are ordered in pure GrevLex. When working with the 120 coefficients of monomials up to degree 7, the order is altered so that v_3^7 is placed as the 64th smallest element. The reason for altering the order is that we know which monomials will be indivisible by the leading terms of the Gröbner-basis.

All polynomials are replaced by rows in matrices with the columns sorted according to the above modified monomial order.

1. Choosing $n = 3$ when performing the elimination of depths gives 30 equations. From these equations it is possible to choose 15 linearly independent equations either by prior knowledge or Gauss-Jordan elimination. There are now 15 equations of degree 6.

2. The results of multiplication by 1, v_1, v_2 and v_3 are stacked into a 60×120 matrix. This matrix (generally) has rank 56. Gauss-Jordan elimination is applied. After elimination there are 56 polynomials of degree 7 and lower.

3. This matrix contains 27 of the 28 elements needed for a Gröbner-basis.

4. To get the last element in the basis, the polynomial represented by row 6 is multiplied by v_2 and is subtracted from the polynomial in row 4 multiplied by v_3. This is done by using the multiplication matrices for the parts of the polynomials that are of degree ≤ 6 and separately handling the 4 elements of degree 7. The remainders of this polynomial are computed with respect to all old polynomials.

Now a GrevLex Gröbner-basis has been calculated. Let I be the ideal formed by our polynomials. A basis for the ring $R = \mathbb{R}[v_1, v_2, v_3]/I$ containing 64 monomials is then chosen. Using the Gröbner-basis it is straightforward to compute the action matrix m_{v_1} for the linear operator

$$T_{v_1} : R \ni f \mapsto v_1 f \in R.$$

For some basis monomials p the product pv_1 is still in the basis so the corresponding column contains a single non-zero entry. For other monomials the results will be monomials outside the matrix. Here the Gröbner-basis is used to find a representative of the equivalence class pv_1 with monomials in the basis of R. The transpose of the action matrix is shown in Figure 5.

The eigen-values of $m_{v_1}^T$ are the solutions for v_1 and as long as these solutions are distinct (generic case) the eigen-vectors give solutions to (v_1, v_2, v_3), cf. [5]. The argument is briefly as follows. Denote μ_v the row vector of monomials and c_f the coordinates (or coefficients) of the polynomial f, i.e. we can think of the polynomial f as $\mu_v c_f$. Now $v_1 \mu_v c_f$ is in the same equivalence class as $\mu_v m_{v_1} c_f$ for each choice of c_f. This means that $m_{v_1}^T \mu_v^T = v_1 \mu_v^T$ for each point $v \in V$. This is, however, an eigen-value problem for the transposed action matrix $m_{v_1}^T$.

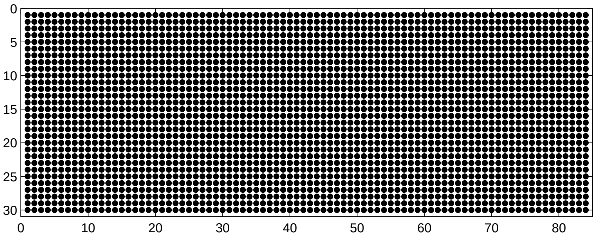

Figure 2: The 30 equations we begin with. Here there are 84 monomials with non-zero coefficients and only these are represented.

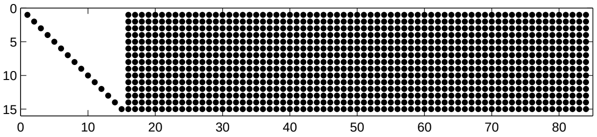

Figure 3: After the first elimination.

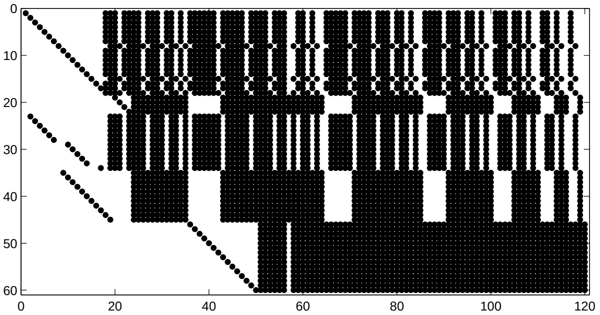

Figure 4: After multiplying by 1,v_1,v_2 and v_3. There are now 120 monomials.

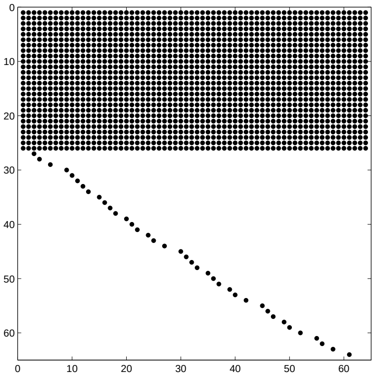

Figure 5: The transpose of the action matrix.

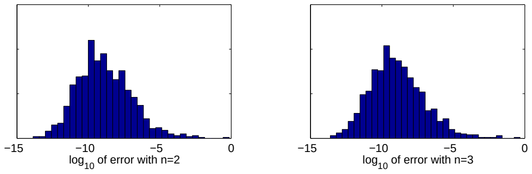

Figure 6: Numerical error when using $n = 2$ and $n = 3$. The mean error is 0.4210^{-4} for $n = 2$ and 0.2710^{-4} for $n = 3$. The median error is 0.8510^{-9} for $n = 2$ and 0.7410^{-9} for $n = 3$. Using $n = 3$ is a little better.

When the eigen-values are not distinct it is possible to consider the action matrix m_f for a random linear combination f of (v_1, v_2, v_3) instead.

The solutions for the translations have to be computed using back substitution, i.e. from Equation (8). This problem is linear in the translation parameters (t_1, t_2).

By studying specific instances of the problem it can be shown that (i) there are examples of image data with 64 solutions. The functional determinant of the problem is non-degenerate and the equations are continuous in the parameters. Using the inverse function theorem there is an open neighborhood of instances with 64 solutions. This completes the proof of Theorem 1.

4.3 Degenerate Situations

With a setup where each point is observed for both times from the same point in the camera coordinate system and the motion is a pure translation, then it is impossible to solve for the scale of the translation. This is typically the case for a multicamera setup.

If all cameras in a multicamera setup all lie on a line and all points are observed by the same camera both in image-set one and image set two, then there is a one-dimensional family of solutions consisting of the line rotating around itself with all points at depth zero. If these false roots are removed then the corresponding analogous problem over $\mathbb{Z}_p$ has 56 solutions.

4.4 Numerical Experiments

As this is a minimal case solver, the errors must be divided into two parts. Part one is the numerical stability of the solver itself and part two is the stability of the solution under noise. The floating point calculations are performed as `double`. When no noise is applied the errors are usually small, 97.8% of the estimated rotation matrices have errors of less than 1 degree. This is shown in Figure 7.

In order to test if this solver can be used for RANSAC implementation we have set up some test with synthetic data. The setup is as follows: The room has side-length two and has 100 points per wall. The camera rig is in the center of the room at the time of image one and rotated by R and translated by t at the time of image two. There is one camera on the rig for each observed point and each camera make an observation for image one and one observation for image two. The cameras lie in a plane in the camera coordinate system and half of them point forward and half of them points backwards. The cameras have a mean distance of 0.125 to the camera coordinate system center.

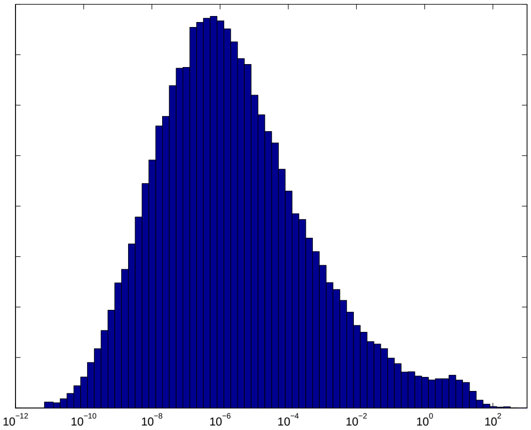

Figure 7: Error (in degrees) in rotation matrix for random points and movements.

When observing points each camera can only see points within $30°$ from it principal direction.

The measured directions to the points are disturbed by Gaussian noise centered at the correct point with standard deviation in degrees given in the plots. In order to get an estimate from degrees to pixel, consider a camera with a field of view of $60°$ with a resolution of 600 pixels. Then one pixel is $0.1°$.

We perform 100 iterations in the RANSAC loop and the solutions are scored based on the re-projected errors of the intersected points, the target function is computed as $F = \sum f_i(f_i + 0.1)^{-1}$ where f_i is the norm of the image error of observation i computed in normalized image coordinates. No iterative adjustment is performed.

In Figure 8 the error in estimating the rotation and the translation are shown. The estimates for the size of the translations is not of comparable quality and getting good estimates for the translation size requires good precision, this is presented in Figure 9.

In Figures 10 and 11. we present the same experiment as in Figures 8 and 9 but with 10% outliers added. No attempt to remove the outliers.

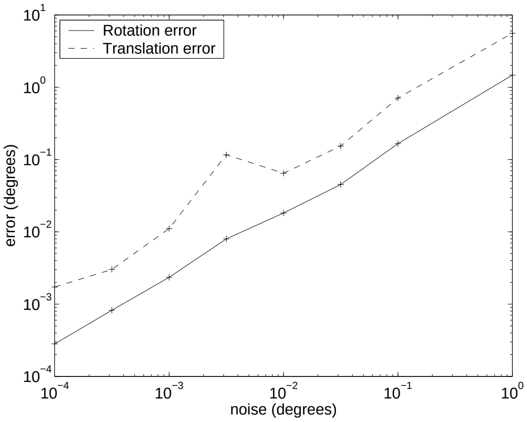

Figure 8: Error in determining the rotation matrix. No outliers, 10° rotation. No outlier removal attempted. Note that one pixel corresponds to 0.1°.

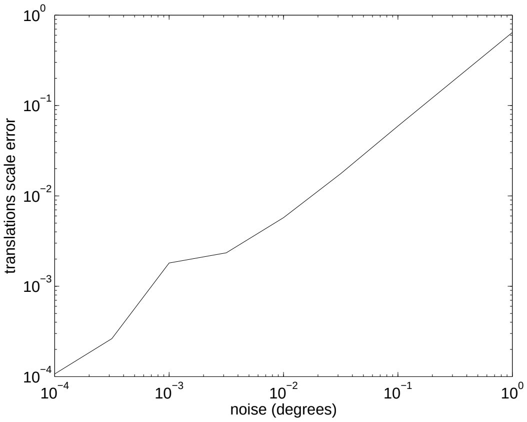

Figure 9: Error in determining the translation scale. No outliers, 10° rotation. No outlier removal attempted. Note that one pixel corresponds to 0.1°. The presented error is calculated as the mean of $abs(log(abs(estimate/true))$.

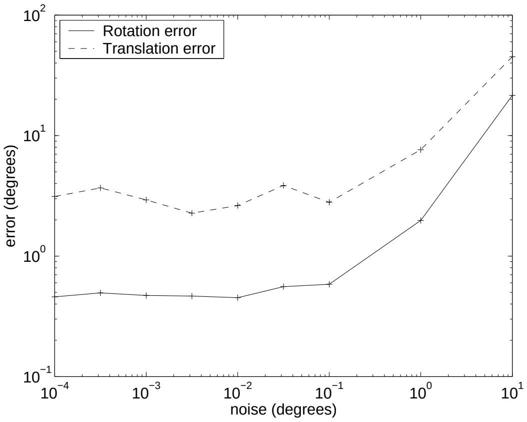

Figure 10: Error in determining the translation scale. 10% outliers, 10° rotation. No outlier removal attempted. Note that one pixel corresponds to 0.1°. The presented error is calculated as the mean of $abs(log(abs(estimate/true))$.

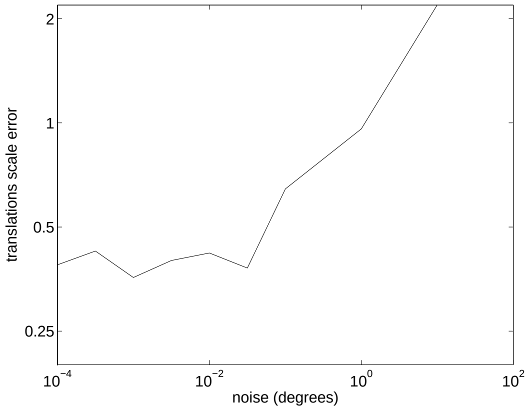

Figure 11: Error in determining the translation scale. 10 % outliers, 10° rotation. No outlier removal attempted. Note that one pixel corresponds to 0.1°. The presented error is calculated as the mean of $abs(log(abs(estimate/true))$.

4.5 Time Requirements

The eigen-value calculation takes 85% of the time in the current implementation and this fraction can only be expected to increase if the other parts are further optimized. Execution time to get $\mathbf{v}$ is $20ms$ on a Celeron $2.4GHz$.

5. Conclusion

We have shown how to solve the generalized minimal case of 2 positions and 6 points in a practical way.

Performing the eliminations in a matrix formulation has allowed us to use pivoting in the elimination, whereas straight Buchberger would not permit this. To the best of our knowledge this is the first minimal solution to any of the generalized relative orientation problems.

In the experiments we have shown that this solver can be used for a RANSAC implementation for generalized cameras. This gives the solver a practical value.

References

[1] E. Adelson and J. Bergen, The Plenoptic Function and the Elements of Early Vision, In *Computational Models of Visual Processing*, ISBN 0-262-12155-7, MIT Press, 1991.

[2] P. Baker, R. Pless, C. Fermüller and Y. Aloimonos, Eyes from Eyes, *SMILE 2000, LNCS 2018*, pp. 204-217, 2001.

[3] C. Chen and W. Chang, On Pose Recovery for Generalized Visual Sensors, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):848-861, July 2004.

[4] D. Cox and J. Little and D. O'Shea, *Ideals, Varieties, and Algorithms*, ISBN 0-387-94680-2,Springer-Verlag, 1997.

[5] D. Cox and J. Little and D. O'Shea, *Using Algebraic Geometry*, ISBN 0-387-98492-5,Springer-Verlag, 1998.

[6] O. Faugeras, *Three-Dimensional Computer Vision: a Geometric Viewpoint*, ISBN 0-262-06158-9, MIT Press, 1993.

[7] M. Fischler and R. Bolles, Random Sample Consensus: a Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography, *Commun. Assoc. Comp. Mach.*, 24:381-395, 1981.

[8] D. Grayson and M. Stillman, *Macaulay 2*, http://www.math.uiuc.edu/Macaulay2/, 1993-2002.

[9] M. Grossberg and S. Nayar, A General Imaging Model and a Method for Finding its Parameters, *IEEE International Conference on Computer Vision*, Volume 2, pp. 108-115, 2001.

[10] A. Gruen and T. S. Huang, *Calibration and Orientation of Cameras in Computer Vision*, ISBN 3-540-65283-3, Springer Verlag, 2001.

[11] Linear pushbroom cameras R. Gupta and R.I. Hartley, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):963-975, September 1997.

[12] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, ISBN 0-521-62304-9, Cambridge University Press, 2000.

[13] J. Koenderink, and A. van Doorn, Affine Structure from Motion, *Journal of the Optical Society of America* 8(2):377-385, 1991.

[14] E. Kruppa, Zur Ermittlung eines Objektes Zwei Perspektiven mit innerer Orientierung, *Sitz-Ber. Akad. Wiss., Wien, math. naturw. Kl. Abt* IIa(122): 1939–194, 1913.

[15] B.Micusik and T.Pajdla. Autocalibration & 3D Reconstruction with Non-central Catadioptric Cameras, *IEEE Conference on Computer Vision and Pattern Recognition* Volume 1, pp 58-65, 2004.

[16] D. Nist´er, A Minimal Solution to the Generalised 3-Point Pose Problem, *IEEE Conference on Computer Vision and Pattern Recognition*, Volume 1, pp. 560-567, 2004.

[17] R. Pless, Using Many Cameras as One, *IEEE Conference on Computer Vision and Pattern Recognition*, Volume 2, pp. 587-593, 2003.

[18] H. Stewenius and K. Åström, Structure and Motion Problems for Multiple Rigidly Moving Cameras, *8th European Conference on Computer Vision*, Volume 3, pp. 252-263, 2004.

[19] P. Sturm and S. Ramalingam, A Generic Concept for Camera Calibration, *8th European Conference on Computer Vision*, Volume 2, pp. 1-13, 2004.