AN ALGORITHMIC SOLUTION TO THE FIVE-POINT POSE PROBLEM BASED ON THE CAYLEY REPRESENTATION OF ROTATIONS

E.V. MARTYUSHEV

Abstract. We give a new algorithmic solution to the well-known five-point relative pose problem. Our approach does not deal with the famous cubic constraint on an essential matrix. Instead, we use the Cayley representation of rotations in order to obtain a polynomial system from epipolar constraints. Solving that system, we directly get relative rotation and translation parameters of the cameras in terms of roots of a 10th degree polynomial.

1. Introduction

In the paper presented we give an algorithmic solution to the 5-point 2-view relative pose problem. It is formulated as follows.

Problem 1. We are given two calibrated pinhole cameras with centers O_1, O_2 and five points $Q_1, \dots, Q_5$ lying in front of the cameras in 3-dimensional Euclidean space, see Figure 1. In every camera coordinate frame the directing vectors of O_jQ_i are only known. The problem is in finding the relative position and orientation of the second camera with respect to the first one.

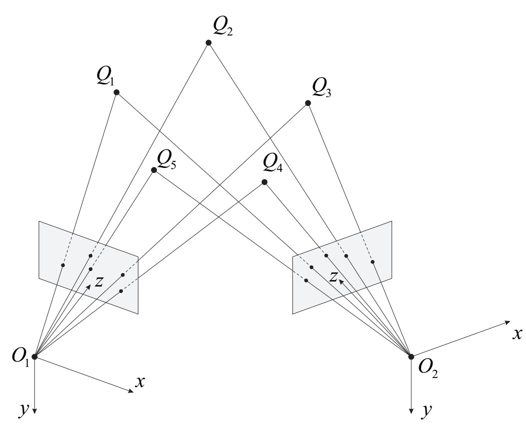


Figure 1. To formulation of the five-point relative pose problem

The 5-point relative pose problem is a key to the 3d scene reconstruction problem, which is in turn used in many computer vision applications such as augmented

Date: February 2, 2013.
Key words and phrases. Five-point pose problem, epipolar constraints, Cayley representation.

reality, self-parking systems, robot path-planning, navigation, etc. It is well known that 5-point algorithms yield significantly better results in accuracy and reliability than 6-, 7- and 8-point algorithms. Moreover, for planar and near-planar scenes only 5-point method allows to get a robust solution without any additional modification of the algorithm.

Problem 1 was first shown by Kruppa [8] in 1913 to have at most eleven solutions. Using the methods of projective geometry, he proposed an algorithm for solving the problem, although it could not lead to a numerical implementation. Demazure [2], Faugeras and Maybank [4], Heyden and Sparr [6] then sharpened Kruppa's result and proved that the exact number of solutions (including complex) is ten.

More efficient and practical solution has been presented by Philip [13] in 1996. His method requires to solve a 13th degree polynomial. In 2004 Nistér [12] improved Philip's algorithm and expressed a solution in terms of a real root of 10th degree polynomial. Afterwards, there were presented many modifications of that algorithm simplifying its implementation [10] or making it more numerically stable [9, 15].

In this paper we give yet another algorithmic solution to the problem using the well-known Cayley representation of rotation matrices [1]. Our approach does not mix rotation and translation parameters of an essential matrix and nevertheless allows one to express a solution in terms of a root of 10th degree univariate polynomial. Experiments on synthetic data show that the method is comparable in accuracy with the existing five-point solvers.

The rest of the paper is organized as follows. In Section 2 we describe in detail our algorithm. In Section 3 we make a comparison of our algorithm with the original Nistér solver [12] on synthetic data. Section 4 concludes.

1.1. **Notation.** We use $\mathbf{a}, \mathbf{b}, \ldots$ for column vectors, and $\mathbf{A}, \mathbf{B}, \ldots$ for matrices. For a matrix $\mathbf{A}$, the entries are A_{ij}, the transpose is $\mathbf{A}^{\mathrm{T}}$, the trace is $\mathrm{Tr}(\mathbf{A})$, and the determinant is $\det(\mathbf{A})$. For two vectors $\mathbf{a}$ and $\mathbf{b}$, the vector product is $\mathbf{a}\times\mathbf{b}$, and the scalar product is $\mathbf{a}^{\mathrm{T}}\mathbf{b}$. For a vector $\mathbf{a}$, the notation $[\mathbf{a}]_\times$ stands for a skew-symmetric matrix such that $[\mathbf{a}]_\times \mathbf{b} = \mathbf{a} \times \mathbf{b}$ for any vector $\mathbf{b}$.

We use $\mathbf{I}$ for identical matrix and $\mathbf{0}$ for zero matrix or vector, $\|\cdot\|$ for the Frobenius norm.

2. Description of the algorithm

2.1. **Initial data transformation.** Initial data for our algorithm are the homogeneous coordinates x_{ji}, y_{ji}, z_{ji} of points Q_i in the coordinate frame of jth camera, $j = 1, 2$, $i = 1, \ldots, 5$ (see Figure 1).

Without loss of generality we can set $x_{j1} = y_{j1} = x_{j2} = 0$ for $j = 1, 2$. The numerically stable way of doing this is as follows. We combine the initial data into two 3×5 matrices

$$\mathbf{A}_j = \begin{bmatrix} x_{j1} & \ldots & x_{j5} \\ y_{j1} & \ldots & y_{j5} \\ z_{j1} & \ldots & z_{j5} \end{bmatrix}, \tag{1}$$

and compute the matrices

$$\mathbf{A}_j'' = \mathbf{H}_{j2}\mathbf{A}_j' = \mathbf{H}_{j2}\mathbf{H}_{j1}\mathbf{A}_j, \tag{2}$$

where $\mathbf{H}_{j1}$ and $\mathbf{H}_{j2}$ are the Householder matrices zeroing x_{j1}, y_{j1} and x_{j2} respectively. The corresponding Householder vectors are

$$\mathbf{h}_{j1} = \begin{bmatrix} x_{j1} \\ y_{j1} \\ z_{j1} + \operatorname{sign}(z_{j1})\sqrt{x_{j1}^2 + y_{j1}^2 + z_{j1}^2} \end{bmatrix}, \quad \mathbf{h}_{j2} = \begin{bmatrix} x'_{j2} \\ y'_{j2} + \operatorname{sign}(y'_{j2})\sqrt{x'^2_{j2} + y'^2_{j2}} \\ 0 \end{bmatrix}.$$

We will see that transformation (2), being quite simple, noticeably simplifies our further computations. In particular, this will allow us to easily convert the resulting 20th degree univariate polynomial (12) to the 10th degree polynomial (13).

2.2. **Epipolar constraints and essential matrix.** We first recall some definitions from multiview geometry, see [3, 5, 11] for details. A *pinhole camera* is a triple $(O, \pi, \mathbf{P})$, where π is an image plane, $\mathbf{P}$ is a central projection of points in 3-dimensional Euclidean space onto π, and O is a camera center (center of projection $\mathbf{P}$). The *focal length* is the distance between O and π, the orthogonal projection of O onto π is called the *principal point*. A pinhole camera is called *calibrated* if all its intrinsic parameters (such as focal length and principal point's coordinates) are known.

Let there be given two calibrated pinhole cameras $(O_j, \pi_j, \mathbf{P}_j)$, $j = 1, 2$. Without loss of generality we can set $\mathbf{P}_1 = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}$, $\mathbf{P}_2 = \begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$, where $\mathbf{R} \in \mathrm{SO}(3)$ is the *rotation matrix* and $\mathbf{t} = \begin{bmatrix}t_1 & t_2 & t_3\end{bmatrix}^{\mathrm{T}}$ is the *translation vector* normalized so that $\|\mathbf{t}\| = 1$.

The well-known *epipolar constraints* [5] on $\mathbf{R}$ and $\mathbf{t}$ read:

$$\begin{bmatrix}x_{2i} & y_{2i} & z_{2i}\end{bmatrix} \mathbf{E} \begin{bmatrix} x_{1i} \\ y_{1i} \\ z_{1i} \end{bmatrix} = 0, \qquad i = 1, \ldots, 5, \tag{3}$$

where $\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$ is called the *essential matrix*.

2.3. **Ten fourth degree polynomials.** Our approach is based on the following well-known result.

Theorem 1 ([1]). *If a matrix $\mathbf{R} \in \mathrm{SO}(3)$ is not a rotation through the angle $\pi + 2\pi k$, $k \in \mathbb{Z}$, about certain axis, then $\mathbf{R}$ can be represented as*

$$\mathbf{R} = \left(\mathbf{I} - \begin{bmatrix} u \\ v \\ w \end{bmatrix}_\times\right) \left(\mathbf{I} + \begin{bmatrix} u \\ v \\ w \end{bmatrix}_\times\right)^{-1}, \tag{4}$$

where $u, v, w \in \mathbb{R}$.

Let $\mathbf{R}$ be represented by (4) and $\mathbf{E}(u, v, w, \mathbf{t}) = [\mathbf{t}]_\times \mathbf{R}$ be an essential matrix.

Proposition 1. *If*

$$\begin{aligned} u' &= \frac{-t_1 - vt_3 + wt_2}{\delta}, \\ v' &= \frac{-t_2 - wt_1 + ut_3}{\delta}, \\ w' &= \frac{-t_3 - ut_2 + vt_1}{\delta}, \end{aligned} \tag{5}$$

where $\delta = ut_1 + vt_2 + wt_3$, then $\mathbf{E}(u', v', w', \mathbf{t}) = -\mathbf{E}(u, v, w, \mathbf{t})$.

Proof. Consider a matrix $\mathbf{R}' = -\mathbf{H_t}\mathbf{R} \in \mathrm{SO}(3)$, where the Householder matrix $\mathbf{H_t} = \mathbf{I} - 2\mathbf{t}\mathbf{t}^{\mathrm{T}}$. Then, $\mathbf{E}' = [\mathbf{t}]_\times \mathbf{R}' = -\mathbf{E}$. By a straightforward computation, the equation $\mathbf{R}'(u', v', w') = -\mathbf{H_t}\mathbf{R}(u, v, w)$ has a unique solution (5). □

Since epipolar constraints (3) are linear and homogeneous in $\mathbf{t}$, we can rewrite them as

$$\mathbf{S}\,\mathbf{t} = \mathbf{0}, \tag{6}$$

where the ith row of 5×3 matrix $\mathbf{S}$ is

$$\begin{bmatrix} x_{1i} & y_{1i} & z_{1i} \end{bmatrix} \mathbf{R}^{\mathrm{T}} \begin{bmatrix} x_{2i} \\ y_{2i} \\ z_{2i} \end{bmatrix}_\times .$$

Now we represent rotation $\mathbf{R}$ in form (4) and take the determinants of all 3×3 submatrices of matrix $\mathbf{S}$. This yields ten polynomial equations:

$$\begin{aligned} f_i = [0]u^4 + [0]u^3v + [0]u^2v^2 + [0]uv^3 + [0]v^4 + [1]u^3 + [1]u^2v \\ + [1]uv^2 + [1]v^3 + [2]u^2 + [2]uv + [2]v^2 + [3]u + [3]v + [4] = 0, \end{aligned} \tag{7}$$

where $i = 1, \ldots, 10$, $[n]$ means a polynomial of degree n in the variable w, $[0]$ is a constant.

Remark 1. Actually, the determinants of 3×3 submatrices of $\mathbf{S}$ give the following expressions:

$$\frac{F_i}{\Delta^3},$$

where $\Delta = 1 + u^2 + v^2 + w^2$ and F_i is a polynomial of 6th total degree. However, one can verify that F_i is factorized as $F_i = f_i\Delta$ and the coefficients of f_i are easily deduced from the coefficients of F_i.

2.4. **Tenth degree univariate polynomial.** Let us rewrite system (7) in form

$$\mathbf{B}\mathbf{m} = \mathbf{0}, \tag{8}$$

where $\mathbf{B}$ is a 10×35 coefficient matrix and

$$\mathbf{m} = \begin{bmatrix} u^4 & u^3v & u^3w & \ldots & v & w & 1 \end{bmatrix}^{\mathrm{T}}$$

is a monomial vector.

We expand system (8) with 20 more polynomials uf_i, vf_i for $i = 1, \ldots, 5$, and wf_i for $i = 1, \ldots, 10$. Thus we get

$$\mathbf{B}' \begin{bmatrix} \mathbf{m}' \\ \mathbf{m} \end{bmatrix} = \mathbf{0}, \tag{9}$$

where $\mathbf{B}'$ is a new 30×50 coefficient matrix and

$$\begin{aligned} \mathbf{m}' = [u^4w, u^3vw, u^3w^2, u^2v^2w, u^2vw^2, u^2w^3, \\ uv^3w, uv^2w^2, uvw^3, uw^4, v^4w, v^3w^2, v^2w^3, vw^4, w^5]^{\mathrm{T}} \end{aligned}$$

is the five-degree monomial vector. It is clear that system (9) is equivalent to (8).

We rearrange columns of matrix $\mathbf{B}'$ and perform Gauss-Jordan elimination with partial pivoting on it. Then the last six rows of the resulting matrix can be represented in form

	u^3w^2	u^3w	u^3	v^3w^2	v^3w	v^3	uv	u	v	1
g_1	1						[3]	[4]	[4]	[5]
g_2		1					[3]	[4]	[4]	[5]
g_3			1				[3]	[4]	[4]	[5]
g_4				1			[3]	[4]	[4]	[5]
g_5					1		[3]	[4]	[4]	[5]
g_6						1	[3]	[4]	[4]	[5]

,

where empty spaces are occupied by zeroes. Also, we have omitted first 28 zero columns. From the corresponding six polynomials $g_1, \ldots, g_6$ we obtain the following four polynomials

$$\begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} \equiv \begin{bmatrix} g_1 \\ g_2 \\ g_4 \\ g_5 \end{bmatrix} - w \begin{bmatrix} g_2 \\ g_3 \\ g_5 \\ g_6 \end{bmatrix} = \mathbf{C}(w) \begin{bmatrix} uv \\ u \\ v \\ 1 \end{bmatrix} = \mathbf{0}, \tag{10}$$

where matrix $\mathbf{C}(w)$ can be represented as

$$\mathbf{C}(w) = \begin{bmatrix} [4] & [5] & [5] & [6] \\ [4] & [5] & [5] & [6] \\ [4] & [5] & [5] & [6] \\ [4] & [5] & [5] & [6] \end{bmatrix}. \tag{11}$$

Remark 2. Since we use only six last rows of matrix $\mathbf{B}'$, there is no need to perform a "complete" Gauss-Jordan elimination on matrix $\mathbf{B}'$. For the first 24 rows of $\mathbf{B}'$ only lower triangular entries should be zeroed.

Denote by $\mathcal{W} = \det \mathbf{C}(w)$. In general, it is a 20th degree polynomial in w.

Proposition 2. *Polynomial $\mathcal{W}$ has a special symmetric form:*

$$\mathcal{W} = \sum_{k=0}^{10} p_k \left[w^{10+k} + (-w)^{10-k} \right], \tag{12}$$

where $p_k \in \mathbb{R}$.

Proof. Due to the conditions $x_{j1} = y_{j1} = 0$, we have $E_{33} = 0$. As a consequence,

$$t_2 = t_1 \frac{R_{23}}{R_{13}} = t_1 \frac{vw + u}{uw - v}.$$

Substituting this into the last identity in (5), we get $w' = -w^{-1}$. Thus, if w_i is a root of $\mathcal{W}$, then so is $-w_i^{-1}$. It follows that

$$\mathcal{W} = p_{10} \prod_{i=1}^{10} (w - w_i)(w + w_i^{-1}) = \sum_{k=0}^{10} p_k \left[w^{10+k} + (-w)^{10-k} \right].$$

$\square$

Substituting $\tilde{w} = w - w^{-1}$, we transform $\mathcal{W}$ to a 10th degree polynomial

$$\tilde{\mathcal{W}} = \sum_{k=0}^{10} \tilde{p}_k \tilde{w}^k, \tag{13}$$

where $\tilde{p}_k$ can be deduced using the formula $\tilde{w}^k = \sum\limits_{i=0}^{k} (-1)^i \binom{k}{i} w^{2i-k}$. The result reads

$$\tilde{p}_k = {\sum_{i=k}^{10}}' \frac{i}{k} \binom{\frac{i+k}{2} - 1}{\frac{i-k}{2}} p_i, \tag{14}$$

where the primed sum is taken over all i from k to 10 such that $i - k \bmod 2 = 0$. Note that in case $k = 0$ the r.h.s. of (14) becomes ${\sum\limits_{i=0}^{10}}' 2p_i$.

2.5. **Structure recovery.** A complex root of $\tilde{\mathcal{W}}$ leads to a complex root of $\mathcal{W}$ and by (4) to complex rotation matrix having no geometric interpretation. Hence only real roots of $\tilde{\mathcal{W}}$ must be treated.

Real roots of $\tilde{\mathcal{W}}$ can be efficiently found first using Sturm sequences [7] for isolating and then Ridders' method [14] for polishing. Then we can recover the second camera matrix applying the following algorithm.

Let $\tilde{w}_0$ be a real root of $\tilde{\mathcal{W}}$. First we find the value

$$w_0 = \tilde{w}_0/2 + \operatorname{sign}(\tilde{w}_0)\sqrt{(\tilde{w}_0/2)^2 + 1},$$

which is a root of $\mathcal{W}$ subject to $|w_0| \geq 1$. After that, we obtain the u- and v-components of the solution by applying Gaussian elimination with partial pivoting on matrix $\mathbf{C}(w_0)$ in (11).

Then we find the entries of $\mathbf{R}$ by (4). Given $\mathbf{R}$, the translation vector $\mathbf{t}$ can be found by performing Gaussian elimination with partial pivoting on matrix $\mathbf{S}(u_0, v_0, w_0)$ in (6). Here we have also taken into account the normalization constraint $\|\mathbf{t}\| = 1$.

Let $\mathbf{H}_\mathbf{t} = \mathbf{I} - 2\mathbf{t}\mathbf{t}^\mathrm{T}$ and $\mathbf{R}' = -\mathbf{H}_\mathbf{t}\mathbf{R}$. It is well-known [5, 12] that there are four possibilities for the second camera matrix: $\mathbf{P}_A = \begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$, $\mathbf{P}_B = \begin{bmatrix}\mathbf{R} & -\mathbf{t}\end{bmatrix}$, $\mathbf{P}_C = \begin{bmatrix}\mathbf{R}' & \mathbf{t}\end{bmatrix}$ and $\mathbf{P}_D = \begin{bmatrix}\mathbf{R}' & -\mathbf{t}\end{bmatrix}$. The only of these matrices is correct, all others correspond to unfeasible configurations.

The true second camera matrix $\mathbf{P}_2$ can be derived from the so-called *cheirality constraint* saying that all the scene points must be in front of the cameras. In particular, this is valid for the first scene point Q_1. Denote by

$$c_1 = -\frac{t_1}{R_{13}} = -\frac{t_2}{R_{23}}, \qquad c_2 = c_1 R_{33} + t_3. \tag{15}$$

Then,

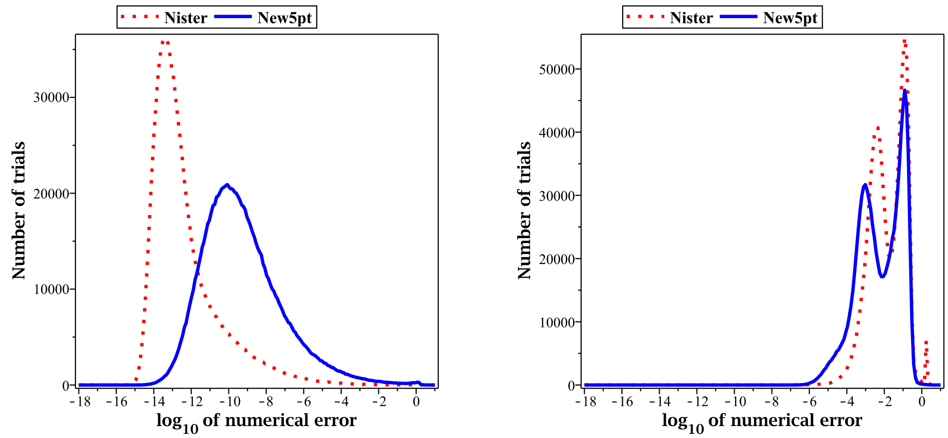
- if $c_1 > 0$ and $c_2 > 0$, then $\mathbf{P}_2 = \mathbf{P}_A$;
- else if $c_1 < 0$ and $c_2 < 0$, then $\mathbf{P}_2 = \mathbf{P}_B$;
- else if $c_1' > 0$ and $c_2' > 0$, then $\mathbf{P}_2 = \mathbf{P}_C$;
- else $\mathbf{P}_2 = \mathbf{P}_D$.

Here the value c_1' and c_2' are computed in the same manner as c_1 and c_2 in (15) with $\mathbf{R}$ being replaced by $\mathbf{R}'$.

Finally, the initial second camera matrix is given by

$$\mathbf{P}_2^{ini} = (\mathbf{H}_{22}\mathbf{H}_{21})^\mathrm{T}\mathbf{P}_2 \begin{bmatrix}\mathbf{H}_{12}\mathbf{H}_{11} & \mathbf{0} \\ \mathbf{0} & 1\end{bmatrix},$$

where the Householder matrices $\mathbf{H}_{j1}$ and $\mathbf{H}_{j2}$ are defined in Subsection 2.1.



(a) Default conditions. The median error is 1.56×10^{-13} for Nister and 2.94×10^{-10} for New5pt

(b) Planar scene and forward motion. The median error is 1.52×10^{-2} for Nister and 7.17×10^{-3} for New5pt

FIGURE 2. Numerical error distribution

3. Experiments on synthetic data

In this section we compare our algorithm with the original 5-point solver by Nistér [12] on synthetic data. The C/C++ implementations of both algorithms have been written. All computations are performed in double precision. Synthetic data setup is the same as in [12]:

Distance to the scene	1
Scene depth	0.5
Baseline length	0.1
Image dimensions	352×288
Field of view	45 degrees

The *numerical error* is defined by

$$\varepsilon = \|\bar{\mathbf{P}}_2 - \mathbf{P}_2\|, \tag{16}$$

where $\bar{\mathbf{P}}_2$ is the ground truth second camera matrix.

The numerical error distributions are reported in Figure 2. The total number of trials is 10^6 in each experiment. We have compared the algorithms first in case of default conditions (Figure 2(a)) and second in the most problematic case in sense of numerical stability — planar scene and forward motion (Figure 2(b)).

In Figure 3 we demonstrate the behaviour of the algorithms under increasing image noise. We add the Gaussian noise with a standard deviation varying from 0 to 1 pixel in a 352×288 image. One sees that in presence of noise the results of both algorithms are almost coincident.

4. Discussion of results

A new algorithm for the 5-point relative pose problem is presented. A computation on synthetic data confirms that it is robust enough. In whole, it is a good

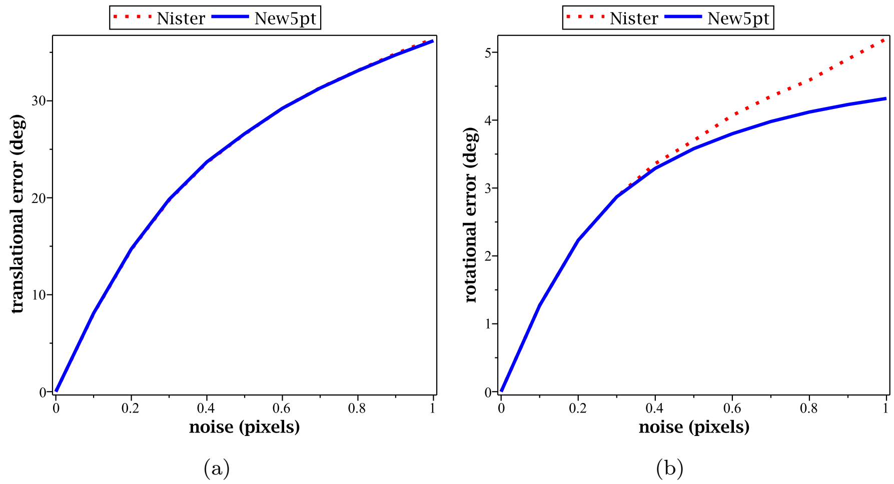


FIGURE 3. Translational (left) and rotational (right) errors relative to Gaussian noise. Default conditions. Each point is a median of 10^6 trials

alternative to the existing five-point solvers. Its major advantage is that it yields a direct structure recovery, i.e. a reconstruction without computing an essential matrix. Such approach is more flexible when we are given some additional information on the camera rotations and/or translations. For instance, if the Euler angles (φ, θ, ψ), representing matrix $\mathbf{R}$, are known to lie in some limits, then so is the variable

$$\tilde{w} = -2\cot(\varphi + \psi). \tag{17}$$

This allows one to discard some roots of the 10th degree polynomial $\tilde{\mathcal{W}}$ at once without structure recovery step.

REFERENCES

1. Cayley, A.: Sur Quelques Propriétés des Déterminants Gauches. J. Reine Angew. Math. **32**, 119-123 (1846).
2. Demazure, M.: Sur Deux Problemes de Reconstruction. Technical Report No 882, INRIA (1988).
3. Faugeras, O.: Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press (1993).
4. Faugeras, O., Maybank, S.: Motion from Point Matches: Multiplicity of Solutions. International Journal of Computer Vision **4**, 225-246 (1990).
5. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Second Edition. Cambridge University Press (2004).
6. Heyden, A., Sparr, G.: Reconstruction from Calibrated Camera – A New Proof of the Kruppa-Demazure Theorem. Journal of Mathematical Imaging and Vision **10**, 1-20 (1999).
7. Hook, D.G., McAree, P.R.: Using Sturm Sequences To Bracket Real Roots of Polynomial Equations. Graphic Gems I. Academic Press, 416-422 (1990).
8. Kruppa, E.: Zur Ermittlung eines Objektes aus zwei Perspektiven mit Innerer Orientierung. Sitz.-Ber. Akad. Wiss., Wien, Math. Naturw. Kl. Abt. **122**, 1939-1948 (1913).
9. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. British Machine Vision Conference (2008).
10. Li, H., Hartley, R.: Five-Point Motion Estimation Made Easy. IEEE-ICPR, 630-633 (2006).

11. Maybank, S.: Theory of Reconstruction from Image Motion. Springer-Verlag (1993).
12. Nistér, D.: An Efficient Solution to the Five-Point Relative Pose Problem. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**, 756-777 (2004).
13. Philip, J.: A Non-Iterative Algorithm for Determining all Essential Matrices Corresponding to Five Point Pairs. Photogrammetric Record **15**, 589-599 (1996).
14. Press, W., Teukolsky, S., Vetterling, W., Flannery, B.: Numerical recipes in C, Cambridge University Press (1988).
15. Stewénius, H., Engels, C., Nistér, D.: Recent Developments on Direct Relative Orientation. ISPRS Journal of Photogrammetry and Remote Sensing **60**, 284-294 (2006).

South Ural State University, 76 Lenin Avenue, Chelyabinsk 454080, Russia
E-mail address: mev@susu.ac.ru