A Robust Least Squares Solution to the Relative Pose Problem on Calibrated Cameras with Two Known Orientation Angles

Gaku Nakano and Jun Takada
Information and Media Processing Laboratories, NEC, Kawasaki, Japan

Keywords: Two-view Geometry, Relative Pose Problem, Essential Matrix, Structure from Motion.

Abstract: This paper proposes a robust least squares solution to the relative pose problem on calibrated cameras with two known orientation angles based on a physically meaningful optimization. The problem is expressed as a minimization problem of the smallest eigenvalue of a coefficient matrix, and is solved by using 3-point correspondences in the minimal case and more than 4-point correspondences in the least squares case. To obtain the minimum error, a new cost function based on the determinant of a matrix is proposed instead of solving the eigenvalue problem. The new cost function is not only physically meaningful, but also common in the minimal and the least squares case. Therefore, the proposed least squares solution is a true extension of the minimal case solution. Experimental results of synthetic data show that the proposed solution is identical to the conventional solutions in the minimal case and it is approximately 3 times more robust to noisy data than the conventional solution in the least squares case.

1 INTRODUCTION

The relative pose problem on calibrated cameras is to estimate an Euclidean transformation between two cameras capturing the same scene from different positions. It is the most basic theory for an image based 3D reconstruction. ”Calibrated” means that the intrinsic camera parameters, e.g., the focal length, are assumed to be known.

The general relative pose problem is expressed by 5 parameters, i.e., 3 orientation angles and a 3D translation vector up to scale. The absolute scale factor cannot be estimated without any prior knowledge about the scene. One point correspondence gives one constraint between the correspondence and the relative pose. The general relative pose problem is solved by at least 5 point correspondences. Many solutions based on point correspondences have been proposed, which are called 5-point (Philip, 1996), (Triggs, 2000), (Nister, 2003), (Stewénius et al., 2006), (Li and Hartley, 2006), (Kukelova et al., 2008b), (Kalantari et al., 2009b), 6-point (Pizarro et al., 2003), 7-point (Hartley and Zisserman, 2004) and 8-point (Hartley and Zisserman, 2004) algorithm.

Meanwhile, a restricted relative pose problem has been raised in which two orientation angles are known. Two known orientation angles are obtained by an IMU (internal measurement unit) sensor or a vanishing point. Using the known angles brings two great benefits. The one is that an angle measured by high accurate sensors is more reliable than that obtained by the point correspondences based algorithms. The other is that the relative pose problem becomes simpler since the degree of freedom is reduced to 3. Therefore, the relative problem is solved by at least 3 point correspondences. It reduces the computational cost of the pose estimation and also reduces the number of iterations of RANSAC (Fischler and Bolles, 1981).

Actual IMU sensors in many consumer products do not have high accuracy needed in those solutions due to noise caused by camera shake and temperature change. Therefore, pragmatic solutions to the restricted pose problem must provide robustness to not only image noise but also sensor noise.

Although some solutions are proposed for the restricted problem, they are not robust and are not able to estimate the optimal relative pose. A solution to the 3-point minimal case is first proposed in (Kalantari et al., 2009a). The problem is formulated as a system of multivariate polynomial equations, and is solved by using a Gröbner basis method. Since the Gröbner basis method requires a large computational cost to decompose large matrices, Kalantari et al.’s solution is not suitable for a RANSAC scheme. Fraundorfer et al. propose 3 solutions (Fraundorfer et al., 2010). The

Nakano G. and Takada J..
A Robust Least Squares Solution to the Relative Pose Problem on Calibrated Cameras with Two Known Orientation Angles.
DOI: 10.5220/0004277801470154
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 147-154
ISBN: 978-989-8565-48-8

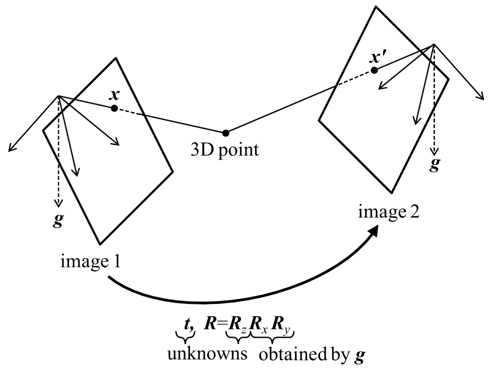


Figure 1: the relative pose problem on calibrated cameras with two known orientation angles.

one is to the minimal case, and the others are to the least squares case of 4 and more than 5 point correspondences. Fraundorfer et al. show that the minimal solution is efficient for a RANSAC scheme. However, the two least squares solutions are not optimal because they do not optimize a physically meaningful cost function with 3 degrees of freedom.

This paper proposes a robust least squares solution to the relative pose problem on calibrated cameras with two known orientation angles based on a physically meaningful optimization. The problem is formulated as a minimization problem of the smallest eigenvalue of a coefficient matrix. To obtain the minimum error, a new cost function based on the determinant of a matrix is proposed instead of solving the eigenvalue problem. The new cost function is not only physically meaningful, but also common in the minimal and the least squares case. Therefore, the proposed least squares solution is a true extension of the minimal case solution.

2 PROBLEM STATEMENT

This section describes the relative pose problem on calibrated cameras with two known orientation angles. Figure 1 shows an example of the problem such that the two orientation angles are obtained by the gravity direction $\boldsymbol{g}$.

Let $\boldsymbol{x}$ and $\boldsymbol{x}'$ be point correspondences represented by 3D homogeneous coordinates in the image 1 and 2, respectively. Then, the general relative pose problem is written in the form

$$\boldsymbol{x}'^T[\boldsymbol{t}]_\times \boldsymbol{R}_z\boldsymbol{R}_y\boldsymbol{R}_x\boldsymbol{x} = 0. \tag{1}$$

where $\boldsymbol{t} = [t_x, t_y, t_z]$ denotes a 3D translation vector up to scale, $[\,]_\times$ denotes a 3×3 skew symmetric matrix representation of the vector cross product and $\boldsymbol{R}_x$, $\boldsymbol{R}_y$ and $\boldsymbol{R}_z$ are 3×3 rotation matrices around x, y and z-axis, respectively. Eq. (1) has 5 degrees of freedom (2 degrees from $\boldsymbol{t}$ and 3 degrees from $\boldsymbol{R}_x$, $\boldsymbol{R}_y$ and $\boldsymbol{R}_z$.)

Let ϕ, ψ and θ be the orientation angles around x, y and z-axis, respectively. $\boldsymbol{R}_x$, $\boldsymbol{R}_y$ and $\boldsymbol{R}_z$ are expressed as

$$\boldsymbol{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix}, \tag{2}$$

$$\boldsymbol{R}_y = \begin{bmatrix} \cos(\psi) & 0 & \sin(\psi) \\ 0 & 1 & 0 \\ -\sin(\psi) & 0 & \cos(\psi) \end{bmatrix}, \tag{3}$$

$$\boldsymbol{R}_z = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

If an IMU sensor is embedded in the cameras or if a vanishing point is detected in the images, the two orientation angles around x and y-axis, i.e., ϕ and ψ, are known. Since $\boldsymbol{R}_x$ and $\boldsymbol{R}_y$ are given by Eqs. (2) and (3), $\boldsymbol{R}_y\boldsymbol{R}_x\boldsymbol{x}$ can be simply expressed by $\boldsymbol{x}$. Then, we have

$$\boldsymbol{x}'^T[\boldsymbol{t}]_\times \boldsymbol{R}_z\boldsymbol{x} = 0. \tag{5}$$

Equation (5) represents the relative pose problem with two known orientation angles. The degree of freedom of Eq. (5) is reduced to $5-2=3$.

Replacing $[\boldsymbol{t}]_\times\boldsymbol{R}_z$ by a 3×3 matrix $\boldsymbol{E}$, Eq. (5) can be written in the linear form

$$\boldsymbol{x}'^T\boldsymbol{E}\boldsymbol{x} = 0. \tag{6}$$

Here, $E_{1,1} = E_{2,2}$, $E_{1,2} = -E_{2,1}$ and $E_{3,3} = 0$. $E_{i,j}$ is the element of $\boldsymbol{E}$ at the i–th row and the j–th column.

$\boldsymbol{E}$ is called the essential matrix if and only if the two of its singular values are nonzero and equal, and the third one is zero (Faugeras, 1993). These constraints are expressed by

$$\det(\boldsymbol{E}) = 0, \tag{7}$$

$$\boldsymbol{E}\boldsymbol{E}^T\boldsymbol{E} - \frac{1}{2}\mathrm{trace}(\boldsymbol{E}\boldsymbol{E}^T)\boldsymbol{E} = \boldsymbol{0}_{3\times3}. \tag{8}$$

$\boldsymbol{E}$ has 6 parameters. However, the degree of freedom is 3 due to the scale ambiguity and the above constraints (Fraundorfer et al., 2010).

Solving a nonlinear equation Eq. (5) and solving a linear equation Eq. (6) with the nonlinear constraints Eqs. (7) and (8) are identical.

3 PREVIOUS WORK

This section briefly describes about the conventional solutions of (Kalantari et al., 2009a) and (Fraundorfer et al., 2010), and points out the drawbacks of them. The algorithm outlines of them are shown in Figures 2(a) and 2(b), respectively.

3.1 Kalantari et al.'s Solution

Kalantari et al. propose a solution to obtain all unknowns in Eq. (5) by solving a system of multivariate polynomial equations.

Firstly, the Weierstrass substitution is used to express $\cos(\theta)$ and $\sin(\theta)$ without the trigonometric functions: $\cos(\theta) = (1-p^2)/(1-p^2)$ and $\sin(\theta) = 2p/(1+p^2)$, where $p = \tan(\theta/2)$. By substituting 3 point correspondences into Eq. (5) and by adding a new scale constraint $\|\boldsymbol{t}\| = 1$, there are 4 polynomial equations in 4 unknowns $\{t_x, t_y, t_z, p\}$ of degree 3. Kalantari et al. adopt a Gröbner basis method to solve the system of polynomial equations. The solutions are obtained by Gauss-Jordan elimination of a 65×77 Macaulay matrix and eigenvalue decomposition of a 12×12 Action matrix. Finally, at most 12 solutions are given from the eigenvectors.

Kalantari et al.'s solution takes much more computational cost than the point correspondence based algorithms due to decomposition of large matrices. Moreover, it is difficult to extend to the least squares case in which the degree of polynomial equations becomes higher and the size of matrices becomes a few hundred dimensions.

In the experiment in this paper, the size of the decomposed matrices and the number of the solutions are not same as (Kalantari et al., 2009a). The details of the implementation are described in section 5.2.

3.2 Fraundorfer et al.'s Solution

Fraundorfer et al. estimate the essential matrix in Eq. (6) instead of the physical parameters. The most important contribution is to propose solutions to the least squares case.

Fraundorfer et al. propose 3 solutions to the case of 3 point, 4 point and more than 5 point correspondences. The basic idea is very similar to the point correspondences based algorithms, i.e., the 5-point, the 7-point and the 8-point algorithm.

From a set of n point correspondences, Eq. (6) can be equivalently written as

$$\boldsymbol{M}\mathrm{vec}(\boldsymbol{E}) = \boldsymbol{0}_{n\times 1}, \tag{9}$$

where $\boldsymbol{M} = \begin{bmatrix} \boldsymbol{x}_1 \otimes \boldsymbol{x}'_1 & \cdots & \boldsymbol{x}_n \otimes \boldsymbol{x}'_n \end{bmatrix}^T$ and vec( ) denotes the vectorization of a matrix. $\otimes$ denotes the Kronecker product.

The solution of Eq. (9) is obtained by

$$\boldsymbol{E} = \sum_{i=1}^{6-n} a_i \boldsymbol{V}_i, \tag{10}$$

where $\boldsymbol{V}_i$ is the matrix corresponding to the generators of the right nullspace of the coefficient matrix $\boldsymbol{M}$, and a_i is an unknown coefficient.

Estimating $\boldsymbol{E}$ is equivalent to calculate a_i. One of a_i can be set to 1 to reduce the number of unknowns due to the scale ambiguity of $\boldsymbol{E}$. In the 3-point case, Eqs. (7) and (8) are used to solve 2 unknowns. Similarly, Eq. (7) is used to solve 1 unknown in the 4-point case. For more than 5 point correspondences, the solution is obtained by taking the eigenvector corresponding to the smallest eigenvalue of $\boldsymbol{M}^T\boldsymbol{M}$.

An essential matrix can be decomposed to 2 $\boldsymbol{R}_z$s and $\pm\boldsymbol{t}$ (Horn, 1990), (Hartley and Zisserman, 2004). Fraundorfer et al.'s 3-point, 4-point and 5-point algorithm estimate at most 4, 3 and 1 essential matrices, respectively. Therefore, they give at most 16, 12 and 4 solutions.

Fraundorfer et al.'s 3-point algorithm satisfies all the constraints. However, the 4-point algorithm considers only one constraint, and the 5-point algorithm does not consider any constraints. For this reason, the solutions of the 4-point and the 5-point algorithm may not be an essential matrix. To correct an estimated $\boldsymbol{E}$ to an essential matrix, a constraint enforcement is carried out by replacing the singular values of $\boldsymbol{E}$ so that the two are nonzero and equal, and the third one is zero. The enforcement does not guarantee to optimize θ and $\boldsymbol{t}$ which minimize Eq. (6) but the change of the Frobenius norm. The 4-point and the 5-point algorithm do not minimize physically meaningful cost function, therefore, they are not optimal solution.

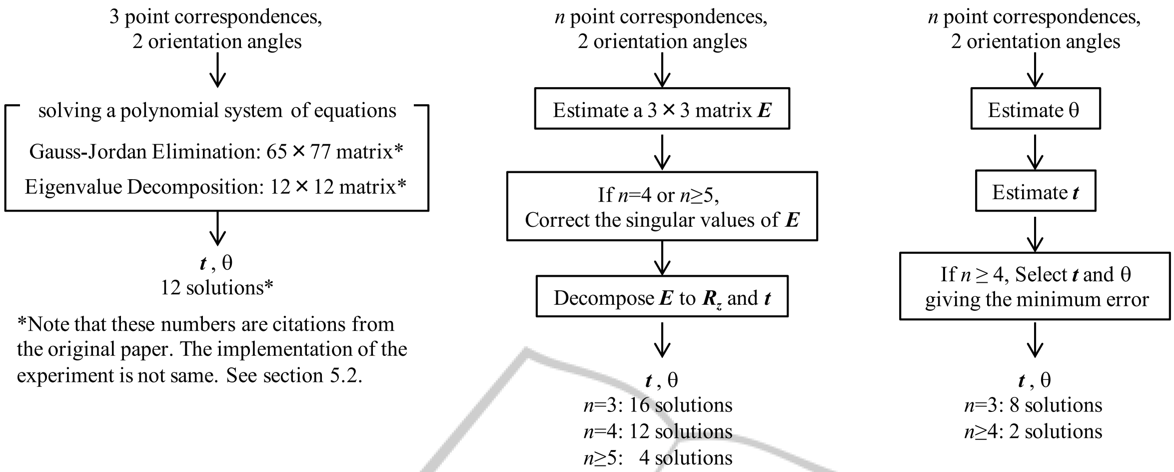
4 PROPOSED SOLUTION

This section describes about the basic idea of the proposed solution in the minimal case firstly, and how to extend the idea to the least squares case secondly. The algorithm outline is shown in Figure 2(c).

4.1 3-point Algorithm for the Minimal Case

Equation (5) can be equivalently written as

$$\boldsymbol{v}^T\boldsymbol{t} = 0, \tag{11}$$



(a) Kalantari et al.'s Solution (b) Fraundorfer et al.'s Solution (c) Proposed Solution

Figure 2: Outlines of the conventional and the proposed solution.

where $\boldsymbol{v} = [\boldsymbol{x}']_\times^T \boldsymbol{R}_z \boldsymbol{x}$.

Given 3 point correspondences, we have

$$\boldsymbol{A}\boldsymbol{t} = \boldsymbol{0}_{3\times1}, \tag{12}$$

where $\boldsymbol{A} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3]^T$ is a 3×3 coefficient matrix involving the unknown θ.

Equation (12) shows that $\boldsymbol{A}$ is singular and $\boldsymbol{t}$ is the nullspace of $\boldsymbol{A}$. Therefore, θ is the solution of $\det(\boldsymbol{A}) = 0$.

In the proposed 3-point algorithm, $\cos(\theta)$ and $\sin(\theta)$ are replaced by new unknowns c and s, respectively, instead of using the Weierstrass substitution. The reason is that it changes the range such that $-\pi \leq \theta \leq +\pi$ to $-\infty < p < +\infty$. This may cause computational instability. Furthermore, a symbolic fractional calculation makes polynomial equations complex in the least squares case.

The unknowns c and s are obtained by solving the following system of polynomial equations:

$$\begin{cases} f_1(c,s) = \det(\boldsymbol{A}) = 0, \\ g(c,s) = c^2 + s^2 - 1 = 0. \end{cases} \tag{13}$$

Equation (13) can be solved by the resultant based method which is also known as the hidden variable method (Cox et al., 2005). Let f_1 and g be polynomial equations of s, and c be regarded as a constant, the resultant $\mathrm{Res}(f_1, g, c) = 0$ is a 4th degree univariate polynomial in c. We get at most 4 solutions as the real roots of $\mathrm{Res}(f_1, g, c) = 0$.

As a result, θ is obtained by

$$\theta = \mathrm{atan2}(s, c). \tag{14}$$

Subsisting estimated θ into Eq. (12), $\boldsymbol{t}$ is obtained by the cross product of two arbitrary rows of $\boldsymbol{A}$. The largest of these three cross products should be chosen for numerical stability (Horn, 1990).

If $\boldsymbol{v}_i \times \boldsymbol{v}_j$ is the largest, we get $\boldsymbol{t}$ up to scale,

$$\boldsymbol{t} = \pm \frac{\boldsymbol{v}_i \times \boldsymbol{v}_j}{\|\boldsymbol{v}_i \times \boldsymbol{v}_j\|}. \tag{15}$$

The proposed 3-point algorithm gives at most 8 possible combinations of 4 θs and $\pm\boldsymbol{t}$.

4.2 4-point Algorithm for the Least Squares Case

This section describes how to extend the proposed 3-point algorithm to the least squares case.

Given more than 4 point correspondences, the pose estimation problem is expressed by an optimization problem:

$$\underset{\boldsymbol{t},\theta}{\text{minimize}} \quad \|\boldsymbol{B}\boldsymbol{t}\|^2 \tag{16}$$
$$\text{subject to} \quad \|\boldsymbol{t}\| = 1$$

where $\boldsymbol{B} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n]^T$ is an $n \times 3$ coefficient matrix involving the unknown θ, and $\|\boldsymbol{t}\| = 1$ is a constraint to avoid the trivial solution $\boldsymbol{t} = \boldsymbol{0}_{3\times1}$.

As widely known in the 8-point algorithm, the optimal $\boldsymbol{t}$ is the eigenvector corresponding to the smallest eigenvalue of $\boldsymbol{B}^T\boldsymbol{B}$ and the minimum error of the cost function $\|\boldsymbol{B}\boldsymbol{t}\|^2$ is equal to the smallest eigenvalue of $\boldsymbol{B}^T\boldsymbol{B}$. The optimization problem Eq. (16) is essentially identical to the eigenvalue problem. However, it is difficult to compute directly the smallest eigenvalue of $\boldsymbol{B}^T\boldsymbol{B}$ represented by a complex number.

To avoid the eigenvalue computation, this paper proposes a new cost function, $\det(\boldsymbol{B}^T\boldsymbol{B})$. The determinant is equal to the product of all eigenvalues and $\boldsymbol{B}^T\boldsymbol{B}$

is positive semidefinite. Therefore, $\|\boldsymbol{Bt}\|^2$ is assumed to be minimized if $\det(\boldsymbol{B}^T\boldsymbol{B})$ is minimized. Thus, the proposed 4-point algorithm minimizes $\det(\boldsymbol{B}^T\boldsymbol{B})$ instead of $\|\boldsymbol{Bt}\|^2$.

Similar to the proposed 3-point algorithm, θ is obtained by solving the following polynomial system of equations:

$$\begin{cases} f_2(c,s) = \dfrac{d}{d\theta}\det(\boldsymbol{B}^T\boldsymbol{B})\Big|_{\substack{\cos(\theta)=c,\\ \sin(\theta)=s}} = 0, \\ g(c,s) = c^2 + s^2 - 1 = 0. \end{cases} \tag{17}$$

Here, $\dfrac{d}{d\theta}\det(\boldsymbol{B}^T\boldsymbol{B})\Big|_{\substack{\cos(\theta)=c,\\ \sin(\theta)=s}}$ denotes that $\cos(\theta)$ and $\sin(\theta)$ in $\dfrac{d}{d\theta}\det(\boldsymbol{B}^T\boldsymbol{B})$ are replaced by c and s, respectively.

The resultant $\mathrm{Res}(f_2, g, c) = 0$ is an 8th degree univariate polynomial in c. We get the optimal θ from the real roots so that it minimizes $\det(\boldsymbol{B}^T\boldsymbol{B})$ or the eigenvalue of $\boldsymbol{B}^T\boldsymbol{B}$.

Finally, we obtain the optimal $\boldsymbol{t}$ by taking the eigenvector corresponding to the smallest eigenvalue of $\boldsymbol{B}^T\boldsymbol{B}$. The proposed 4-point algorithm gives at most 2 possible combination of one θ and $\pm\boldsymbol{t}$.

Moreover, the proposed 4-point algorithm includes the solutions of the proposed 3-point algorithm. For this reason, the proposed 4-point algorithm is a true extension of the 3-point algorithm. The proof is described in Appendix.

5 EXPERIMENTS

5.1 Synthetic Data

The robustness of the proposed solutions are evaluated under various image and angle noise. 3D points are generated randomly similar to (Fraundorfer et al., 2010) so that the 3D points have a depth of 50% of the distance of the first camera to the scene. In (Fraundorfer et al., 2010), 2 camera configurations are performed, i.e., sideway and forward motion with random rotation. To simulate more realistic environment, random motion with random rotation is performed in this experiment. The baseline between the two cameras is 10% of the distance to the scene.

Kalantari et al. and Fraundorfer et al. assume that the error of the two orientation angles measured by a low cost sensor is from 0.5 [degree] to at most 1.0 [degree]. However, the accuracy of almost of all low cost sensors are not necessarily opened. Some of them may have more larger noise than 1.0 [degree]. Therefore, in this experiment, the error is assumed at most 3.0 [degree].

For an image noise test, the standard deviation of Gaussian noise is fixed $\sigma = 0.5$ [degree] for the two known angles, and is changed $0 \le \sigma \le 3$ [pixel] for the point correspondences. Similarly, for an angle noise test, the standard deviation is fixed $\sigma = 0.5$ [pixel] for the point correspondences, and is changed $0 \le \sigma \le 3$ [degree] for the two known angles.

The estimation errors of θ and $\boldsymbol{t}$ are evaluated as follows:

$$Error(\theta_{est}, \theta_{true}) = \mathrm{abs}(\theta_{est} - \theta_{true}), \tag{18}$$

$$Error(\boldsymbol{t}_{est}, \boldsymbol{t}_{true}) = \cos^{-1}\left(\frac{\boldsymbol{t}_{est}^T\boldsymbol{t}_{true}}{\|\boldsymbol{t}_{est}\|\,\|\boldsymbol{t}_{true}\|}\right), \tag{19}$$

where the subscript *est* and *true* denote the estimated and the ground truth value, respectively. If multiple solutions are found, the one having minimum error is selected. The RMS (root mean square) errors in degrees are plotted over 500 independent trials for each noise level in the result figures.

5.2 Results of the Minimal Case

The robustness of the proposed are evaluated in the minimal case and compared to the two conventional 3-point algorithms and Nister's 5-point algorithm based on point correspondences. The conventional 3-point algorithms are implemented by the authors and the 5-point algorithm is implemented by H. Stewenius[1]. Kalantari et al's solution is implemented by using Kukelova's automatic generator of Gröbner basis solvers (Kukelova et al., 2008a)[2]. As mentioned in section 3.1, the matrix sizes are not same as the original. The Macaulay matrix is 58×66 and the Action matrix is 8×8 in this experiment.

As shown in Figures 3 and 4, all the 3-point algorithms are almost same performance. There is no difference between the degree of freedom of each algorithm. Thus, all the 3-point algorithms solve the mathematically identical problem.

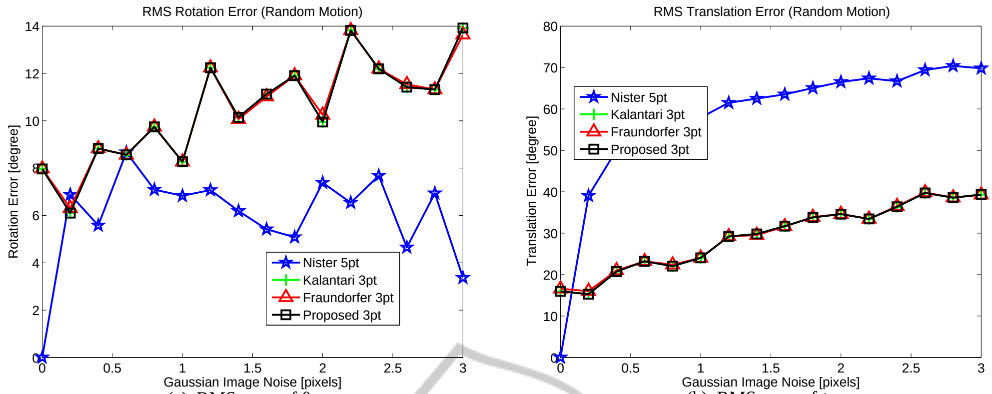
5.3 Results of the Least Squares Case

The robustness of the proposed 4-point algorithm are evaluated in the least squares case and compared to Fraundorfer et al.'s 5-point algorithm and Hartley's 8-point algorithm based on point correspondences. All algorithms are implemented by the authors.

Figure 5 shows the robustness for 100 point correspondences with the fixed angle noise and variable

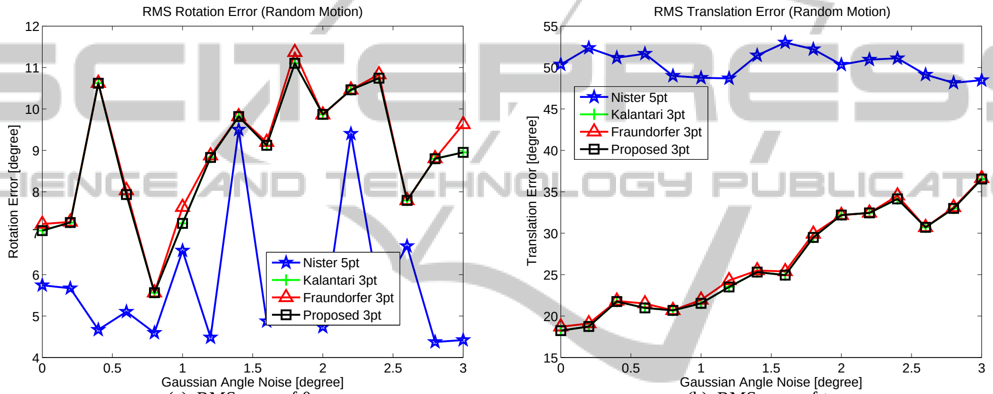
[1] http://www.vis.uky.edu/∼stewe/FIVEPOINT/

[2] http://cmp.felk.cvut.cz/minimal/automatic_generator.php



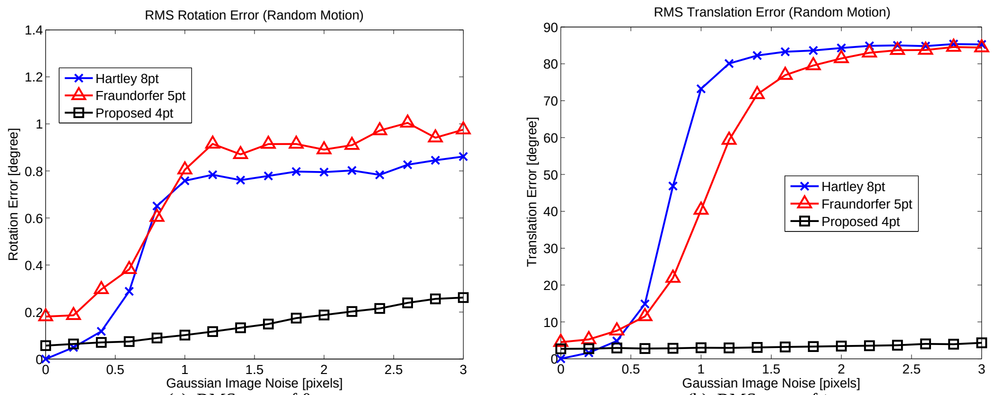
(a) RMS error of θ (b) RMS error of $\boldsymbol{t}$

Figure 3: Results of the minimal case with fixed angle noise and variable image noise.



(a) RMS error of θ (b) RMS error of $\boldsymbol{t}$

Figure 4: Results of the minimal case with fixed image noise and variable angle noise.

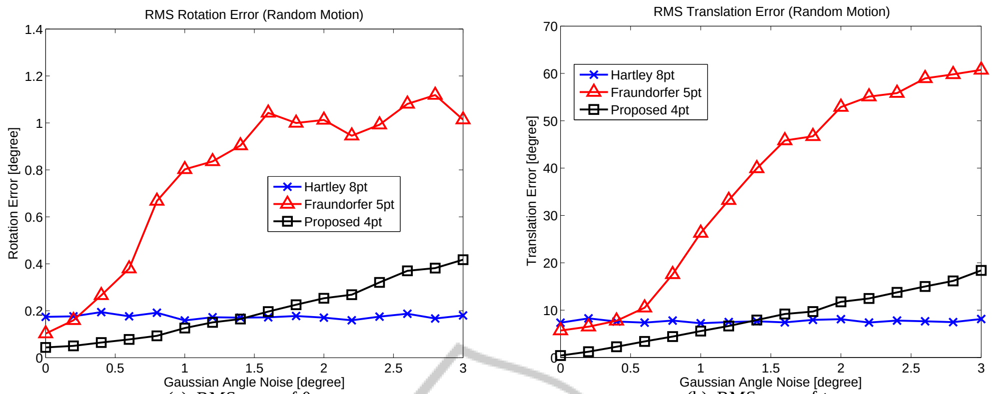


(a) RMS error of θ (b) RMS error of $\boldsymbol{t}$

Figure 5: Results of the least squares case with fixed angle noise and variable image noise.

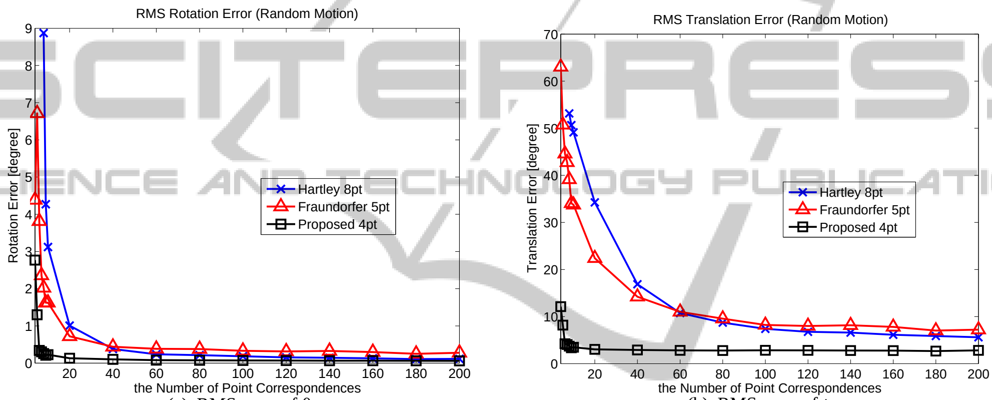
image noise. As the image noise becomes larger, the error of the conventional algorithms become larger sharply. By contrast, that of the proposed 4-point algorithm is much less than them.

Figure 6 shows the result of the fixed image noise and variable angle noise. Hartley's 8-point algorithm is not influenced by the angle noise because it does not use the known angles. Fraundorfer et al.'s 5-point algorithm is more accurate than Hartley's 8-point algorithm if the angle noise is less than 0.4 [degree]. The tolerance of the proposed 4-point algorithm is approximately 1.4 [degree]. This is 3 times more robust than Fraundorfer et al.'s 5-point algorithm.



(a) RMS error of θ (b) RMS error of $\boldsymbol{t}$

Figure 6: Results of the least squares case with fixed image noise and variable angle noise.



(a) RMS error of θ (b) RMS error of $\boldsymbol{t}$

Figure 7: Results of changing the number of the point correspondences.

5.4 Results of Changing the Number of the Point Correspondences

The influence of changing the number of the point correspondences are evaluated in the least squares case. The image and angle noise are fixed 0.5 [pixel] and 0.5 [degree], respectively. From 4 to 200 point correspondences are evaluated.

As shown in Figure 7, the proposed 4-point algorithm is always best. It is notable that the proposed 4-point algorithm reaches the performance boundary at 40–60 point correspondences, whereas the conventional algorithms needs more than 100 point correspondences. This is very important for practical use since few dozens of point correspondences are obtained generally. Moreover, for more than 40 point correspondences, Fraundorfer et al.'s 5-point algorithm is worse than Hartley's 8-point algorithm which uses only point correspondences. The proposed 4-point algorithm outperforms both algorithm regardless of the number of the point correspondences.

According to the results in sections 5.2 and 5.3, sufficiently robust and accurate solutions can be obtained by the proposed cost function without solving the eigenvalue problem.

6 CONCLUSIONS

A robust least squares solution to the relative pose problem on calibrated cameras with two known orientation angles are proposed in this paper. The problem is expressed as a minimization problem of the smallest eigenvalue of a coefficient matrix. To obtain the minimum error, a new cost function based on the determinant of a matrix is proposed instead of solving the eigenvalue problem. The new cost function is not only physically meaningful, but also common in the minimal and the least squares case. The conventional solutions employ different algorithms for the minimal case and the least squares case. By contrast, the proposed least squares solution is a true extension of the minimal case solution. Experimental results of synthetic data show that the proposed solution is identical to the conventional solutions in the minimal case and it is approximately 3 times more robust to noisy data than the conventional solution in the least squares case. A real data experiment using consumer IMU sensors is in the future research.

REFERENCES

Cox, D., Little, J., and O'Shea, D. (2005). *Using Algebraic Geometry*. Springer, 2nd edition.

Faugeras, O. (1993). *Three-dimensional computer vision: a geometric viewpoint*. the MIT Press.

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

Fraundorfer, F., Tanskanen, P., and Pollefeys, M. (2010). A minimal case solution to the calibrated relative pose problem for the case of two known orientation angles. *Computer Vision–ECCV 2010*, pages 269–282.

Hartley, R. I. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition.

Horn, B. (1990). Recovering baseline and orientation from essential matrix. *J. Optical Society of America*.

Kalantari, M., Hashemi, A., Jung, F., and Guédon, J.-P. (2009a). A new solution to the relative orientation problem using only 3 points and the vertical direction. *CoRR*, abs/0905.3964.

Kalantari, M., Jung, F., Guedon, J., and Paparoditis, N. (2009b). The five points pose problem: A new and accurate solution adapted to any geometric configuration. *Advances in Image and Video Technology*, pages 215–226.

Kukelova, Z., Bujnak, M., and Pajdla, T. (2008a). Automatic generator of minimal problem solvers. *Computer Vision–ECCV 2008*, pages 302–315.

Kukelova, Z., Bujnak, M., and Pajdla, T. (2008b). Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. *BMVC 2008*, 2(5).

Li, H. and Hartley, R. (2006). Five-point motion estimation made easy. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 630–633. IEEE.

Nister, D. (2003). An efficient solution to the five-point relative pose problem. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–195. IEEE.

Philip, J. (1996). A non-iterative algorithm for determining all essential matrices corresponding to five point pairs. *The Photogrammetric Record*, 15(88):589–599.

Pizarro, O., Eustice, R., and Singh, H. (2003). Relative pose estimation for instrumented, calibrated imaging platforms. *Proceedings of Digital Image Computing Techniques and Applications, Sydney, Australia*, pages 601–612.

Stewénius, H., Engels, C., and Nistér, D. (2006). Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(4):284–294.

Triggs, B. (2000). Routines for relative pose of two calibrated cameras from 5 points. *Technical Report, INRIA*.

APPENDIX

The proof of that the proposed 4-point algorithm includes the 3-point algorithm is as follows.

Substituting 3 point correspondences into Eq. (17), we have

$$\begin{aligned}\frac{d}{d\theta}\det(\boldsymbol{B}^T\boldsymbol{B}) &= \frac{d}{d\theta}\det(\boldsymbol{A}^T\boldsymbol{A})\\ &= \frac{d}{d\theta}\det(\boldsymbol{A})^2 \\ &= 2\det(\boldsymbol{A})\frac{d}{d\theta}\det(\boldsymbol{A}).\end{aligned} \tag{20}$$

We can construct a system of polynomial equations as follows:

$$\begin{cases} f_3(c,s) = \det(\boldsymbol{A})\dfrac{d}{d\theta}\det(\boldsymbol{A})\Big|_{\substack{\cos(\theta)=c,\\ \sin(\theta)=s}} = 0, \\ g(c,s) = c^2+s^2-1 = 0. \end{cases} \tag{21}$$

The solutions of the resultant $\mathrm{Res}(f_3, g, c) = 0$ includes that of $\mathrm{Res}(f_1, g, c) = 0$.