Hand-Eye Calibration Based on Screw Motions

Zijian Zhao, Yuncai Liu
Inst. of Image Processing & Pattern Recognition
Shanghai Jiao Tong University
200240 Shanghai, China
{zj_zhao, whomliu}@sjtu.edu.cn

Abstract

When computer vision technique is used in robotics, robotic hand-eye calibration is a very important research task. Many algorithms have been proposed for hand-eye calibration. Based on these algorithms, we introduce a new hand-eye calibration algorithm in this paper, which employs the screw motion theory to establish a hand-eye matrix equation by using quaternion and gets a simultaneous result for rotation and translation by solving linear equations. The algorithm proposed in this paper has high and stable computational efficiency without non-linear minimization and can be understood easily. Both simulations and real experiments show the superiority of our algorithm over the comparative algorithms.

1. Introduction

When computer vision technique is used in robotics, such as robot assisted measurement, the robot sensors are usually mounted on the robot hand to form a hand-eye system. It is necessary to know the relative position between the robot sensor and the robot hand. Hand-eye calibration is the process of computing the relative 3-D position and orientation of the sensor frame with respect to the robot hand frame. Because both the sensor frame and the robot hand frame are usually located inside the camera and the robot, the relative position between them cannot be measured directly.

Many approaches have been proposed in the literature [1]-[7]. The usual way to describe the hand-eye calibration is by means of 4×4 homogeneous transformation matrices. Shiu and Ahmad [1] have firstly formulated the well-known hand-eye calibration equation:

$$AX = XB \tag{1}$$

where X is the transformation from camera to robot hand, A is the camera motion, and B is the robot motion. Every homogeneous matrix has the form

$$\begin{pmatrix} R & \vec{t} \\ 0^T & 1 \end{pmatrix}$$

so equation (1) can be written as the following equations,

$$R_A R_X = R_X R_B \tag{2}$$

$$R_A \vec{t}_X + \vec{t}_A = R_X \vec{t}_B + \vec{t}_X \tag{3}$$

Tsai and Lenz [2] proposed a more efficient linear algorithm for solving the hand-eye calibration. Rotation axis for matrix R is used in their method. Zhuang and Shiu [3] solved the hand-eye calibration problem by using a one-stage iterative algorithm. Not only is this algorithm less sensitive to noise, it has not the drawback of two-stage algorithm that rotation estimation errors propagate to position errors. Chou and Kamel [4] presented an algorithm based on quaternions, and gave a closed-form solution to the problem. Horaud and Dornaika [5] firstly applied a simultaneous nonlinear minimization with respect to the rotation quaternion and the translation vector, and achieved good experimental results. Konstantinos [6] used dual quaternions in the hand-eye calibration algorithm and described the solution via the singular value decomposition (SVD).

Chen [7] employed the screw motion theory to analyze hand-eye calibration, and gave the motion constraint of hand-eye geometry. In this paper, we describe a new hand-eye calibration method, which uses Chen's motion constraint and substitutes the rotation quaternions for the rotation matrices. By using the singular value decomposition (SVD), we get the solution of our method without nonlinear minimization. Both simulations and real experiments show that the accuracy performance of our method is better than that of the comparative algorithms.

2. Screw Motion Decomposition

According to Chasles' theorem [7], a general rigid body displacement can be accomplished by means of a rotation about a unique axis and a translation along the same axis. Such a description of rigid body displacement is called a screw, and the unique axis is called the screw axis. The screw axis is a line in space, which in parametric form is described by

$$L : \vec{p} = \vec{c} + k\vec{u}, 0 < k < +\infty,$$

where $\vec{u}$ is a unit direction vector representing the axis of rotation, $\vec{c}$ is the position of the line to the origin, and $\vec{c} \cdot \vec{u} = 0$. The two quantities associated with the screw axis are the rotation angle θ and the translation d. A screw is denoted by (d, θ, L).

A rigid body displacement is usually described as a rotation followed by a translation. There is a conversion between the conventional motion description, denoted by $(R, \vec{t})$, and the corresponding screw description (d, θ, L). According to the Rodrigues

The 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00

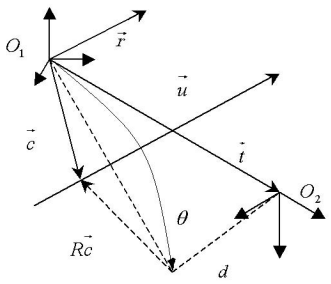

Figure 1. The geometry of a screw.

formula, we can describe R as a rotation through an angle ϕ and around a unit axis $\vec{r}$ that passes through the origin. As shown in Fig.1, we have

$$\vec{t} = d\vec{u} + (I - R)\vec{c}. \tag{4}$$

Using the Rodrigues formula,

$$R\vec{c} = \vec{c} + \sin(\phi)\vec{r} \times \vec{c} + (1 - \cos\phi)\vec{r} \times (\vec{r} \times \vec{c}) \tag{5}$$

and $\vec{r} \cdot \vec{c} = 0$, it follows that

$$\vec{c} = (\vec{t} - (\vec{t} \cdot \vec{r})\vec{r} + \cot(\phi/2)\vec{r} \times \vec{t})/2. \tag{6}$$

Then, the screw translation d, the rotation angle θ, and the direction vector $\vec{u}$ can be determined in [7].

3. Matrix Equation Based on Screw Motion Constraint

We can get the following equation from (2),

$$R_A = R_X R_B R_X^T,$$

which is a similarity transformation since R_X is an orthogonal matrix. Hence, an eigenvector $\vec{v}$ of R_B corresponds to an eigenvector $R_X\vec{v}$ of R_A. Screw axis $\vec{u_B}$ is an eigenvector of R_B, the corresponding eigenvector of R_A is $R_X\vec{u}_B$. Then we have the formulation of screw axis:

$$\vec{u}_A = R_X\vec{u}_B. \tag{7}$$

According to (4), we can rewrite $\vec{t}_A$ and $\vec{t}_B$. Substituting them into (3), and $d_A = d_B$, yields

$$\vec{c}_A = R_X\vec{c}_B + \vec{t}_X - (\vec{u}_A \cdot \vec{t}_X)\vec{u}_A. \tag{8}$$

Equation (7) and (8) are applied in our hand-eye calibration algorithm as Chen's screw motion constraint.

Substituting the quaternion q for the rotation matrix R_X, we have the screw motion constraint as the following form,

$$u_A = qu_B\bar{q}, \tag{9}$$

$$c_A = qc_B\bar{q} + Ut_X, \tag{10}$$

where

$$U = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & U_A & \\ 0 & & & \end{pmatrix}$$

$$U_A = \begin{pmatrix} 1 - u_{A1}^2 & -u_{A1}u_{A2} & -u_{A1}u_{A3} \\ -u_{A1}u_{A2} & 1 - u_{A2}^2 & -u_{A2}u_{A3} \\ -u_{A1}u_{A3} & -u_{A2}u_{A3} & 1 - u_{A3}^2 \end{pmatrix}$$

and $\vec{u}_A = [u_{A1}\ u_{A2}\ u_{A3}]$.The quaternion v is described as the form $(0, \vec{v})$, so $u_A = (0, \vec{u}_A), u_B = (0, \vec{u}_B), c_A = (0, \vec{c}_A), c_B = (0, \vec{c}_B), t_X = (0, \vec{t}_X)$.

Multiplying (9) and (10) on the right with q, we obtain

$$u_A q = qu_B,$$
$$c_A q = qc_B + Uq',$$

which may be rewritten as

$$u_A q - qu_B = 0,$$
$$c_A q - qc_B - Uq' = 0,$$

where $q' = t_X q$. Because the scale part from each of the two equations above is redundant, we have in total six equations with eight unknowns, which can be written as the following form

$$\begin{pmatrix} \vec{u}_A - \vec{u}_B & Skew(\vec{u}_A + \vec{u}_B) & 0_{3\times1} & 0_{3\times3} \\ \vec{c}_A - \vec{c}_B & Skew(\vec{c}_A + \vec{c}_B) & 0_{3\times1} & -U_A \end{pmatrix} \begin{pmatrix} q \\ q' \end{pmatrix} = 0. \tag{11}$$

The coefficient matrix of the matrix vector equation above is a 6×8 matrix, and the vector of unknowns $[q^T, q'^T]^T$ is an 8-dimensional vector. $Skew(\vec{u})$ is the antisymmetric matrix corresponding to the cross-product with $\vec{u}$. Matrix equation (11) is the key equation of our hand-eye calibration algorithm in this paper.

4. Solution of the Hand-Eye Matrix Equation

Equation (11) is an ill equation and has not the unique solution. This is nothing new, since it is well known that the hand-eye transformation can be determined from at least two motions of the hand-eye system only if the screw axes are not parallel [7]. Suppose that $n \geq 2$ motions are given. We construct a $6n \times 8$ matrix

$$A = [S_1^T\ S_2^T\ \cdots\ S_n^T]^T, \tag{12}$$

where $S_i (i = 1, 2, \cdots, n)$ is the coefficient matrix of equation (11) at the ith motion. S_i is maximally of rank 6, so A has the maximal rank 6. The equation $A[q^T, q'^T]^T = 0$ has the unique solution, which belongs to the null space of A.

We compute the singular value decomposition $A = U\Sigma V^T$,where Σ is a diagonal matrix with the singular values, U is composed of the left-singular vectors, and V is composed of the right-singular vectors. If the rank is 6, the last two right-singular vectors $\vec{v}_7$ and $\vec{v}_8$, which correspond to the two vanishing singular values, span the null space of A. In the presence of noise, we always choose the two right-singular vectors corresponding to the two minimal singular values as $\vec{v}_7$ and $\vec{v}_8$. The solution of the equation $A[q^T, q'^T]^T = 0$ must be a linear combination of them. If $\vec{v}_7$ and $\vec{v}_8$ are written as being composed of two 4×1 vectors, $\vec{v}_7 = [\vec{u}_1^T, \vec{v}_1^T]^T$ and $\vec{v}_8 = [\vec{u}_2^T, \vec{v}_2^T]^T$, then the solution can be written as

$$\begin{pmatrix} q \\ q' \end{pmatrix} = \lambda_1 \begin{pmatrix} \vec{u}_1 \\ \vec{v}_1 \end{pmatrix} + \lambda_2 \begin{pmatrix} \vec{u}_2 \\ \vec{v}_2 \end{pmatrix}$$

Using the two constraints

$$q^T q = 1 \; and \; q^T q^{'} = 0, \tag{13}$$

we can have two quadratic equations in λ_1 and λ_2:

$$(\lambda_1 \vec{u}_1 + \lambda_2 \vec{u}_2)^T (\lambda_1 \vec{u}_1 + \lambda_2 \vec{u}_2) = 1, \tag{14}$$

$$\lambda_1^2 \vec{u}_1^T \vec{v}_1 + \lambda_1 \lambda_2 (\vec{u}_1^T \vec{v}_2 + \vec{u}_2^T \vec{v}_1) + \lambda_2^2 \vec{u}_2^T \vec{v}_1 = 0. \tag{15}$$

Since λ_1 and λ_2 never both vanish, we set $s = \lambda_1/\lambda_2$ without loss of generality that $\lambda_2 \neq 0$. Inserting $\lambda_1 = s\lambda_2$ to equation (14) and (15) yields

$$\lambda_2^2 (s\vec{u}_1 + \vec{u}_2)^T (s\vec{u}_1 + \vec{u}_2) = 1, \tag{16}$$

$$\lambda_2^2 [s^2 \vec{u}_1^T \vec{v}_1 + s(\vec{u}_1^T \vec{v}_2 + \vec{u}_2^T \vec{v}_1) + \vec{u}_2^T \vec{v}_2] = 0. \tag{17}$$

From equation (17), two solutions for s are obtained. Considering the presence of noise, we choose from the two solutions for s the one that gives the largest module value for $s\vec{u}_1 + \vec{u}_2$. λ_2 is computed from equation (16), and has the same sign with s. Then λ_1 is computed by s and λ_2. After getting $[q^T, q^{'T}]^T$, we can compute the parameter $\vec{t}_X$ from $q^{'} = t_X q$.

5. Experiments

To experimentally test our method, we performed simulations and real experiments. Our tests compared our method with two additional methods. The first one is similar to the method proposed by Horaud and Dornaika [5]. The additional method has the objective function:

$$f(q, \vec{t}_X) = \|\vec{u}_A q - q\vec{u}_B\|^2 + \|((R_A - I)\vec{t}_X - \vec{t}_A)q - q\vec{t}_B\|^2 \tag{18}$$

to be minimized with respect to q and $\vec{t}_X$ subject to $\|q\|^2 = 1$. The Levenberg-Marquardt minimization is applied in the method. Like every iterative nonlinear minimization, it needs starting values. The second comparative method we applied was the dual quaternion method described by Konstantinos [6], which is similar to our method in the matrix equation. In the following text, we denote our method based on screw motion constraint by "SM", the method based on nonlinear minimization by "LM", and the dual quaternion method by "DQ".

5.1. Simulations

We establish 20 hand motions $(R_B, \vec{t}_B)$ in a realistic setup similar to the real experiments of the next section, and then use the setup to compute the corresponding camera motions $(R_A, \vec{t}_A)$. As usual, Gaussian noise is added to both the hand and camera motions. The noise is added as absolute value to the unit rotation quaternion, and as relative value to the translation vector. For each noise level (defined by standard deviation) and for a large number N of trials, we compute the errors associated with rotation and translation as follows:

$$e_q = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \|q - \tilde{q}_i\|^2}, \tag{19}$$

$$e_t = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^{N} \|\vec{t} - \tilde{t}_i\|^2}}{\|\vec{t}\|}, \tag{20}$$

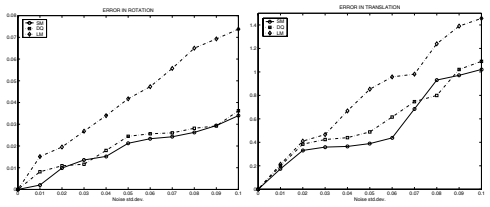

Figure 2. Errors in rotation and translation with variation in noise level.

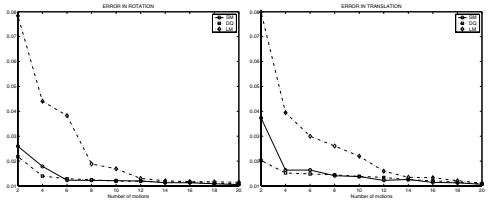

Figure 3. Errors in rotation and translation as a function of the number of motions.

where q and $\vec{t}$ are the nominal values of the hand-eye transformation, $\tilde{q}_i$ and $\tilde{t}_i$ are the estimated rotation and translation for some trial i, and $N = 1000$.

In Fig.2 we compare our SM algorithm with the DQ algorithm and the LM algorithm under different noise levels with 20 hand-eye motions. Our method (SM) and the dual quaternion method (DQ) exhibit better behavior than the nonlinear methods (LM) in the experiment. Our method (SM) and the dual quaternion method (DQ) have very close performance, though our method is a little better than the dual quaternion method in performance. The reason for this similarity is that the two methods have the similar form of the solution in the hand-eye calibration matrix equation. We also compute the standard deviation of the rotation and translation errors in the SM algorithm at noise level 0.1. The standard deviation of rotation errors is 0.015, and the standard deviation of translation errors is 0.1. The results of the standard deviation show that the error fluctuation of the SM algorithm is small and acceptable.

Fig.3 shows the rotational and translational errors as a function of the number of motions. In the experiment, we vary the number of motions from 2 to 20, and we keep the noise level at 0.1. We can notice that the SM method and the DQ method are more accurate than the LM method. For a few motions, the nonlinear method (LM) does not converge properly. With the increasing of motions, all the three methods can get the low rotational and translational errors.

5.2. Real Experiments

The real experiment is conducted with two CCD cameras mounted on the last joint of a MOTOMAN CYR-UPJ Robot (Fig.4 Left). The cameras are calibrated with the Zhang's method [8]. We

Algorithm	E_R	E_t
LM	0.00065	0.0231
DQ	0.00043	0.0187
SM	0.00032	0.0197

Table 1. The first real experiment results.

have done three independent experiments by using the robot system. The first one is proposed in [5], which uses the residual error to show the performance of hand-eye calibration algorithms. The second one, which is also applied in the classic paper by Tsai and Lenz [2], is the ability to predict the camera pose by using only robot-motion data. The third one is the dot-hit experiment: the gripper of the robot is moved to hit the dots on the calibration pattern, by using reconstruction of dots to show the error of position.

In the first experiment, a data set was obtained with 20 different positions of the hand-eye device with respect to a calibration object after 20 motions. Using the data set, we test our algorithm (SM), the dual quaternion algorithm (DQ) and the nonlinear algorithm (LM). Table.1 summarizes the results obtained with the data set. The second column of the table shows the sum of squares of the absolute error in rotation, E_R

$$E_R = \sum \|R_A R_X - R_X R_B\|^2. \tag{21}$$

The third column shows the relative error in translation, E_t

$$E_t = \frac{\sum \|(R_A - I)\vec{t}_X - R_X \vec{t}_B + \vec{t}_A\|^2}{\sum \|R_X \vec{t}_B - \vec{t}_A\|^2}. \tag{22}$$

In the real experiment, the SM algorithm provides less error than the LM algorithm and the DQ algorithm.

The cameras are moved to 20 different locations in the second experiment. We compute the hand-eye calibration by using the three methods (LM, SM and DQ) from stations 1 through $N(4 < N < 15)$. Then we predict the camera pose $\hat{A}_i(i = 15 \cdots 20)$ for the stations 15 to 20 from the robot motion $B_i(i = 15 \cdots 20)$ and the first camera pose A_1:

$$\hat{A}_i = X B_i^{-1} X^{-1} A_1. \tag{23}$$

We compare $\hat{A}_i$ and A_i $(i = 15 \cdots 20)$, and average the error with 6 stations. Fig.4(Right) shows the rotational and translational errors as a function of the number N. For other reasons such as camera calibration error, the experiment results do not show the obvious difference in the performance of the three methods. However, the mean error of our method is the lowest.

The third experiment is the dot-hit assessment. As shown in Fig.5(Left), we move the gripper of the robot to hit ten dots on the calibration pattern, and reconstruct the positions of dots through using the position probe. Using the position probe, we can achieve the coordinates of the dots in the camera frame $\vec{D}_c^i$ $(i = 1, 2, \cdots, 10)$. If we compute the hand-eye transformation X by hand-eye calibration algorithms, we can get the coordinates of the dots in the robot base frame $\vec{D}_r^i$ $(i = 1, 2, \cdots, 10)$, and

$$\tilde{D}_r^i = C^{-1} X \tilde{D}_c^i, \tag{24}$$

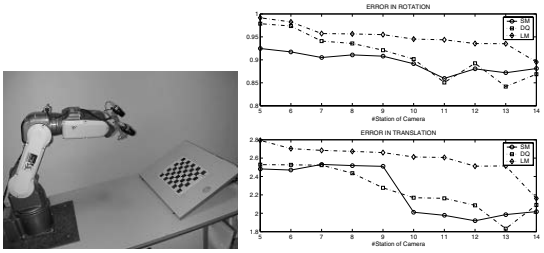

Figure 4. The physical hand-eye calibration setup.(Left) Errors in comparing the predicted camera pose $\hat{A}_i$ and the camera pose A_i averaged over 6 stations.(Right)

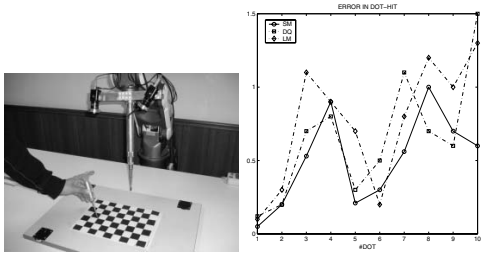

Figure 5. The dot-hit experiment with the position probe.(Left) The absolute errors of the dot positions in the dot-hit experiment.(Right)

where C is the transformation from the robot base frame to the robot gripper frame, $\tilde{D}_r^i = (\vec{D}_r^{iT}, 1)^T$ and $\tilde{D}_c^i = (\vec{D}_c^{iT}, 1)^T$. Comparing the positions $\vec{D}_R^i$ from the robot controller and the reconstruction positions $\vec{D}_r^i$ of the position probe, we can get Fig.5(Right), which show the absolute position errors $\|\vec{D}_R^i - \vec{D}_r^i\|$ by using three hand-eye calibration methods (LM, SM and DQ). The absolute position errors of the three methods are very close at every dot, but the mean (0.505) and standard deviation (0.313) of our method's errors are smaller than those of other two methods' errors.

To clarify the merits of our algorithm (SM), we apply it in a robot assisted surgical system (Fig.6 Left), which has a hand-eye robot for total knee arthroplasty. We do the hand-eye calibration for the robot by using the SM algorithm, and then the robot performs the operation on 100 sawbone models (phantom bones). The distribution of cutting errors is shown in Fig.6(Right). 92% of sawbone models' cutting errors are less than 1.5mm, which contain camera calibration errors, hand-eye calibration errors and robot operation errors. The hand-eye calibration error accounts for a very small proportion of the cutting error.

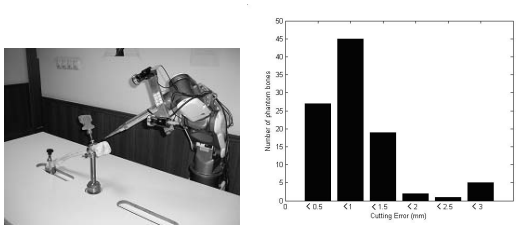

Figure 6. The robot assisted surgical system.(Left) The distribution of cutting errors.(Right)

5.3. Qualitative Perturbation Analysis

It is interesting to notice that the SM algorithm and the DQ algorithm have the superior performance in the simulations and the real experiments. We will try to explain it by using qualitative perturbation analysis with respect to the rotation quaternion q.

Firstly, consider the following matrix

$$\hat{A} = A + \Delta A,$$

where $\hat{A}$ is a perturbed version of the coefficient matrix A in the SM algorithm and the DQ algorithm with perturbation ΔA. Define the correlation matrix

$$\hat{R} = E\{\hat{A}^T\hat{A}\} = E\{(A+\Delta A)^T(A+\Delta A)\}.$$

If the perturbation ΔA is irrelevant to A statistically, we have

$$\hat{R} = E\{A^TA\} + E\{\Delta A^T\Delta A\} = R + \sigma^2 I,$$

where σ is the standard deviation of the noise. If $rank(A) = 6$ and R has the eigenvalue decomposition $V\Sigma V^T$, then $\hat{R}$ can be written as

$$\hat{R} = V\Sigma V^T + \sigma^2 I = V(\Sigma + \sigma^2 I)V^T = V\Pi V^T,$$

where $\Pi = \Sigma + \sigma^2 I = diag(\sigma_1^2 + \sigma^2, \sigma_2^2 + \sigma^2, \cdots, \sigma_6^2 + \sigma^2, \sigma^2, \sigma^2)$, and $\Sigma = diag(\sigma_1^2, \sigma_2^2, \cdots, \sigma_6^2, 0, 0)$, $\sigma_i^2 (i = 1, 2, \cdots, 6)$ are the nonzero eigenvalues of R. According to the formula above, we know that the last two right-singular vectors of $\hat{A}$ corresponding to the two minimal singular values are equal in statistical concept to the two right-singular vectors of A corresponding to the two vanishing singular values. Hence, the lower error limit of the rotation quaternion q in the SM algorithm and the DQ algorithm is vanishing.

Secondly, consider the following objective function in the LM algorithm:

$$\|u_A q - q u_B\|^2.$$

When it converges to the global minimum δ, the objective function is written as

$$\|u_A q - q u_B\|^2 = q^T(B + \Delta B)q = q^T Bq + q^T \Delta Bq = \delta,$$

where ΔB is the perturbation of B (see [5]). $\|\delta - q^T\Delta Bq\|$ reflects indirectly the error of the rotation quaternion q in the LM algorithm. The lower error limit of q is not always vanishing.

From the perturbation analysis above, we know that the influence of perturbation in the SM algorithm and the DQ algorithm is smaller than that in the LM algorithm. The conclusion that the SM algorithm and the DQ algorithm have better performance than the LM algorithm is proved.

6. Conclusion

In this paper, we proposed a new hand-eye calibration algorithm based on screw motion constraints, which enabled us to establish a linear homogeneous system for the solution of rotation and translation parameters. The computation of the null space with SVD in the algorithm yields an accurate solution of hand-eye transformation. We implemented two comparative algorithms (LM, DQ), the LM algorithm involving a nonlinear minimization and solving simultaneously for rotation and translation, the DQ algorithm applying the dual quaternion in the hand-eye calibration. We compared three algorithms in simulations and real experiments, where we observe the superior performance of our algorithm (SM). Furthermore, it is explained by using qualitative perturbation analysis. In the future, we will do some researches on the perturbation of noise to achieve a better performance of our algorithm.

References

[1] Y. C. Shiu, S. Ahmad. Calibration of wrist-mounted robotics sensors by solving homogeneous transform equations of the form AX=XB. *IEEE Transactions on Robotics and Automation*, 5(1):16-27, 1989.

[2] R. Tsai, R. Lenz. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 5(3):345-358, 1989.

[3] Hanqi Zhuang, Y. C. Shiu. A noise-tolerant algorithm for robotic hand-eye calibration with or without sensor orientation measurement. *IEEE Transactions on Systems, Man and Cybernetics*, 23(4):1168-1175, 1993.

[4] J. Chou, M. Kamel. Finding the position and orientation of a sensor on a robot manipulator using quaternions. *The International Journal of Robotics Research*, 10(3):240-254, 1991.

[5] R. Horaud, F. Dornaika. Hand-eye calibration. *The International Journal of Robotics Research*, 14(3):195-210, 1995.

[6] Konstantinos Daniilidis. Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286-298, 1999.

[7] H. H. Chen. A screw-motion approach to uniqueness analysis of head-eye geometry. *In:Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 145-151, 1991.

[8] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330-1334, 2000.