Robotica (2013) volume 31, pp. 747–756. © Cambridge University Press 2013
doi:10.1017/S0263574713000040

A flexible method combining camera calibration and hand–eye calibration

Zijian Zhao†* and Ying Weng‡

†*School of Control Science and Engineering, Shandong University, Jinan 250061, China*
‡*School of Computer Science, Bangor University, Bangor LL57 1UT, UK*

(Accepted January 8, 2013. First published online: February 1, 2013)

SUMMARY
We consider the conventional techniques of vision robot system calibration where camera parameters and robot hand–eye parameters are computed separately, i.e., first performing camera calibration and then carrying out hand–eye calibration based on the calibrated parameters of cameras. In this paper we propose a joint algorithm that combines the camera calibration and the hand–eye calibration together. The proposed algorithm gives the solutions of the cameras' parameters and the hand–eye parameters simultaneously by using nonlinear optimization. Both simulations and real experiments show the superiority of our algorithm. We also apply our algorithm in the real application of the robot-assisted surgical system, and very good results have been obtained.

KEYWORDS: Camera calibration; Hand–eye calibration; Robot system; Homography; Factorization.

1. Introduction

When computer vision techniques are used in robotics, such as robot-assisted measurement, the robot sensors (cameras) are usually mounted on the robot hand to form a hand–eye system. Before we use the hand–eye system, we have to know the camera parameters (intrinsic and extrinsic parameters) and the relative transformation parameters between the cameras and the robot hand. Knowing both the intrinsic and extrinsic camera parameters, referred to as camera calibration, is an essential technique concerned by many people in the vision community.[1–15] Classical camera calibration is performed by capturing a reference object with a known Euclidean structure (3D,[2] 2D,[1,5,7,9,15,19] and 1D[8,10]). The camera self-calibration techniques do not use any calibration object.[11–14] By using two constraints[11,17] on the intrinsic parameters of the camera displacement, we can recover the intrinsic and extrinsic parameters from a number of image correspondences. Hand–eye calibration is an important technique to determine the relative position between the cameras and the robot hand. There are many approaches of hand–eye calibration proposed by various authors.[20–30] We can classify them into three categories: Standard hand–eye calibration,[21–24,28,29,32] Screw motion-based hand–eye calibration,[25–27] and On-line hand–eye calibration.[30]

However, all the above techniques are just focused on their special areas, and do not consider the hand–eye system as a whole. The positioning error of a hand–eye robot system depends on both camera parameters and hand–eye parameters, so it is necessary to probe the possibility of combining camera and hand–eye calibrations. Our research has focused on developing such an algorithm that can be used easily in a hand–eye vision robot system to carry out camera and hand–eye calibration simultaneously.

The algorithm proposed in this paper is performed by moving the robot and observing a 2D reference object by the cameras at a few different robot poses. The whole hand–eye vision robot system can be calibrated very efficiently by the multi-objective optimization of our algorithm. Compared with conventional calibration procedures (camera calibration then hand–eye calibration), the proposed joint calibration algorithm is more efficient in the real applications. Both computer simulations and real experiments have been used to test the proposed algorithm, and very good results have been obtained.

Note that Maim[20] proposed a method fitting the hand–eye calibration in the framework of the plane-based intrinsic camera calibration with a close form solution. His method does not consider the vision robot system as a whole system, and the results of his method are not ideal with a low precision.

The paper is organized as follows. We start in Section 2 by introducing the basic equations by observing a 2D reference object (a plane). Section 3 describes the hand–eye calibration problem. In this section we also explain the hand–eye equation and give the whole error model of the hand–eye geometry. Section 3.3 is the calibration procedure. We start with an estimated solution, followed by nonlinear optimization. Degenerated conditions of the algorithm is also considered. In Section 4, both simulation and real data are used to validate the proposed algorithm, and the paper is closed in Section 5 by some conclusions.

2. Preliminaries of Plane-Based Homography

In this section we introduce some basic equations about plane-based homography. The notation used in the paper is as follows.

* Corresponding author. E-mail: zhaozijian@sdu.edu.cn

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

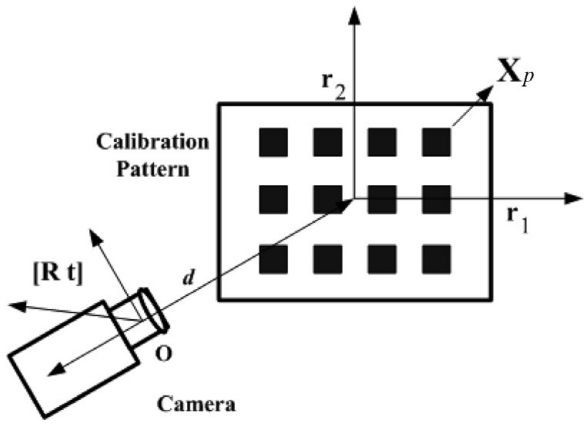


Fig. 1. Geometrical relationship between the camera and the plane.

Notation. In this paper, the camera is modeled by the usual pinpole: The relationship between the 3D point and its image projection point is given by

$$\tilde{\mathbf{u}} \simeq \mathbf{A}[\mathbf{R} \quad \mathbf{t}]\tilde{\mathbf{X}}, \tag{1}$$

where the 2D point is denoted by $\tilde{\mathbf{u}} = [u\ v\ 1]^T$, the 3D point is denoted by $\tilde{\mathbf{X}} = [x\ y\ z\ 1]^T$, and the camera intrinsic matrix[1] is given by

$$\mathbf{A} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}.$$

2.1. Plane description

A plane in the 3D space can be determined by three non-collinear 3D points a, b, c. All 3D points on the plane can be written in the following form:

$$\begin{aligned} \mathbf{X}_p &= a + x_p(b-a) + y_p(c-a) \\ &= [(b-a)\ (c-a)\ a]\tilde{\mathbf{x}}_p, \end{aligned} \tag{2}$$

where $\tilde{\mathbf{x}_p} = [x_p\ y_p\ 1]^T$ is the point's 2D position on the plane. Suppose that all 3D points $\mathbf{X}_p$ are written as homogeneous coordinates $[x\ y\ z\ 1]^T$, then from Eq. (2) we can get (Fig.1)

$$\tilde{\mathbf{X}}_p = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{d} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}. \tag{3}$$

Without loss of generality, we can think $\mathbf{r}_1$ and $\mathbf{r}_2$ are unit vectors that are parallel to the plane and orthogonal to each other, $\mathbf{r}_1^T\mathbf{r}_1 = \mathbf{r}_2^T\mathbf{r}_2 = 1, \mathbf{r}_1^T\mathbf{r}_2 = 0$.

2.2. Plane-based homography

From Eqs. (1) and (3), we have

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \simeq \mathbf{A}[\mathbf{R} \quad \mathbf{t}] \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{d} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}, \tag{4}$$

which gives the homography relationship between a reference plane and the image plane. The homography matrix $\mathbf{H}$ is defined as

$$\mathbf{H} = \mathbf{A}\mathbf{H}_c\mathbf{H}_p, \tag{5}$$

with $\mathbf{H}_c = [\mathbf{R} \quad \mathbf{t}]$, $\mathbf{H}_p = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{d} \\ 0 & 0 & 1 \end{bmatrix} = \mathbf{H}_{\text{ep}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

Matrix $\mathbf{H}_c$ is inherently determined by the camera, and matrix $\mathbf{H}_p$ describes the plane's position in the world coordinate. Matrix $\mathbf{H}_{\text{ep}}$ is the expanded matrix of $\mathbf{H}_p$ and has the following form:

$$\mathbf{H}_{\text{ep}} = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_1 \times \mathbf{r}_2 & \mathbf{d} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

3. Joint Calibration of Hand–Eye Robot System

3.1. Factorization of multi-homography matrices

In general, a hand–eye robot system always has a stereo head configuration of two cameras. Without loss of generality, we make the world frame coincides with the first camera frame, so matrix $\mathbf{H}_p(\mathbf{H}_{\text{ep}})$ can be regarded as the pose parameters from reference plane to the first camera. Moving the stereo head of the robot, a number of images are captured by observing the static plane of calibration. From these images, the homography matrices of two cameras at different robot poses are recovered up to an unknown scale factor.[7]

Suppose $\mathbf{H}_j^i (i = 1, 2; j = 1, 2, \cdots, n)$ is the homography matrix of the ith camera at the jth robot pose. In fact, we can only get $\hat{\mathbf{H}}_j^i = \lambda_j^i \mathbf{H}_j^i$, where λ_j^i is an unknown scale factor. Ueshiba and Tomita[15] proposed a method using "relative homography" to estimate the unknown scale λ_j^i, but the estimated scale factors are relative values up to a constant. Sturm[16] also gave a solution with the condition that cameras are calibrated, although the intrinsic camera matrix is usually unknown in our case. We thus use an eclectic method for scale factors.

First, we give the estimated solutions of the two cameras' intrinsic matrices ($\mathbf{A}_1$ and $\mathbf{A}_2$) as described in Appendix. Then, left-multiplying $\hat{\mathbf{H}}_j^i$ by $\mathbf{A}_i^{-1}$ on the left, we get

$$\begin{aligned} \tilde{\mathbf{M}}_j^i &= \mathbf{A}_i^{-1}\hat{\mathbf{H}}_j^i = \lambda_j^i \mathbf{H}_c^i \mathbf{H}_{pj} \\ &= \lambda_j^i [\mathbf{R}^i \quad \mathbf{t}^i] \begin{bmatrix} \mathbf{r}_{1j} & \mathbf{r}_{2j} & \mathbf{d}_j \\ 0 & 0 & 1 \end{bmatrix} \\ &= \lambda_j^i [\mathbf{R}^i\mathbf{r}_{1j} \quad \mathbf{R}^i\mathbf{r}_{2j} \quad \mathbf{R}^i\mathbf{d}_j + \mathbf{t}^i]. \end{aligned} \tag{6}$$

We take the first two columns of $\tilde{\mathbf{M}}_j^i$ as $\tilde{\mathbf{M}}_{j3\times 2}^i$. According to the theory given by Horn and Hilden,[31] we do the singular value decomposition (SVD) to $\tilde{\mathbf{M}}_{j3\times 2}^i$, getting $\tilde{\mathbf{M}}_{j3\times 2}^i =$

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

$\mathbf{U}_{3\times2}\Sigma_{2\times2}\mathbf{V}^T_{2\times2}$. The scale factor λ^i_j is obtained as

$$\lambda^i_j = \frac{\text{Trace}\left(\left(\tilde{\mathbf{M}}^i_{j3\times2}\right)^T\tilde{\mathbf{M}}^i_{j3\times2}\right)}{\text{Trace}\left(\mathbf{T}^T\tilde{\mathbf{M}}^i_{j3\times2}\right)}, \tag{7}$$

where $\mathbf{T} = \mathbf{U}_{3\times2}\mathbf{V}^T_{2\times2}$.

After obtaining the λ^i_j, we can construct the measurement matrix $\mathbf{M}$, which is written as

$$\begin{aligned}\mathbf{M} &= \begin{bmatrix}\tilde{\mathbf{M}}^1_1/\lambda^1_1 \cdots \tilde{\mathbf{M}}^1_n/\lambda^1_n \\ \tilde{\mathbf{M}}^2_1/\lambda^2_1 \cdots \tilde{\mathbf{M}}^2_n/\lambda^2_n\end{bmatrix} \\ &= \underbrace{\begin{bmatrix}\mathbf{H}^1_c \\ \mathbf{H}^2_c\end{bmatrix}}_{\mathbf{C}}\underbrace{\begin{bmatrix}\mathbf{H}_{p1}\cdots\mathbf{H}_{pn}\end{bmatrix}}_{\mathbf{P}}.\end{aligned} \tag{8}$$

$\mathbf{C}$ is a 6×4 matrix, and $\mathbf{P}$ is a $4\times3n$ matrix, so the measurement matrix $\mathbf{M}$ has a rank of 4. Matrix $\mathbf{C}$ involves the extrinsic parameters of two cameras, and matrix $\mathbf{P}$ includes the pose parameters of the plane. The factorization method[18] using SVD is chosen to solve the problem.

Suppose that the matrix $\mathbf{M}$ is factored into $\mathbf{C}'$ and $\mathbf{P}'$, $\mathbf{M} = \mathbf{C}'\mathbf{P}'$, with

$$\mathbf{C}' = \begin{bmatrix}\mathbf{H}^{1'}_c \\ \mathbf{H}^{2'}_c\end{bmatrix}, \mathbf{P}' = \begin{bmatrix}\mathbf{H}'_{p1}\cdots\mathbf{H}'_{pn}\end{bmatrix}. \tag{9}$$

However, the factored $\{\mathbf{C}', \mathbf{P}'\}$ are usually not the real $\{\mathbf{C}, \mathbf{P}\}$ that we wanted. We have to construct a 4×4 non-singular matrix $\mathbf{Q}$[17,32] to recover the real $\{\mathbf{C}, \mathbf{P}\}$, namely $\mathbf{H}^{i'}_c \simeq \mathbf{H}^i_c\mathbf{Q}$, $\mathbf{H}'_{pj} \simeq \mathbf{Q}^{-1}\mathbf{H}_{pj}$. We have already fixed the world frame to the first camera, so it is obvious that $\mathbf{H}^1_c = [\mathbf{I} \quad \mathbf{0}]$, and the matrix $\mathbf{Q}$ has the following form:

$$\mathbf{Q} = \begin{bmatrix}\mathbf{H}^{1'}_c \\ \mathbf{q}^T\end{bmatrix}, \tag{10}$$

where $\mathbf{q}$ is a 4-vector needing to be solved. Considering the orthogonal constraints of $\mathbf{H}_{pj} \simeq \mathbf{Q}\mathbf{H}'_{pj}$, we have

$$\mathbf{r}'^T_{1j}\mathbf{W}\mathbf{r}'_{1j} = \mathbf{r}'^T_{2j}\mathbf{W}\mathbf{r}'_{2j} = \alpha^2_j, \tag{11}$$

where $[\mathbf{r}'_{1j} \quad \mathbf{r}'_{2j}]$ is the 3×2 upper-left submatrix of $\mathbf{H}'_{pj}$; $\mathbf{W} = \mathbf{P}^T\mathbf{P}$, the matrix $\mathbf{P}$ is the 3×3 upper-left submatrix of $\mathbf{H}^{1'}_c$; and α_j is an unknown positive scale factor. From Eq. (11), each α_j is determined. After getting $\alpha_j (j = 1, 2, \cdots, n)$, we have the equation

$$\mathbf{q}^T[\mathbf{H}'_{p1} \quad \cdots \quad \mathbf{H}'_{pn}] = [0 \quad 0 \quad \alpha_1 \quad \cdots \quad 0 \quad 0 \quad \alpha_n] \tag{12}$$

to solve the vector $\mathbf{q}$.

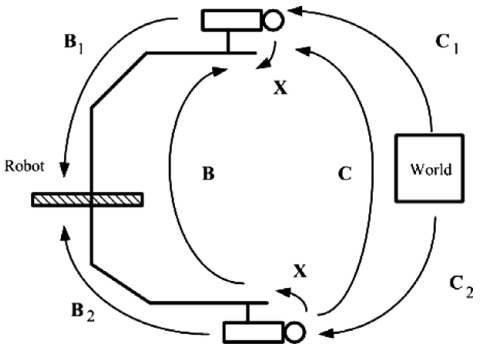


Fig. 2. Hand–eye calibration by solving $\mathbf{CX} = \mathbf{XB}$, $\mathbf{C} = \mathbf{C}_2^{-1}\mathbf{C}_1$ and $\mathbf{B} = \mathbf{B}_2\mathbf{B}_1^{-1}$.

After determining the matrix $\mathbf{Q}$, we can recover the real $\{\mathbf{C}, \mathbf{P}\}$ by

$$\mathbf{C} = \begin{bmatrix}\mathbf{H}^{1'}_c\mathbf{Q}^{-1} \\ \mathbf{H}^{2'}_c\mathbf{Q}^{-1}\end{bmatrix}, \mathbf{P} = [(1/\alpha_1)\,\mathbf{Q}\mathbf{H}'_{p1} \quad \cdots \quad (1/\alpha_n)\,\mathbf{Q}\mathbf{H}'_{pn}]. \tag{13}$$

3.2. Hand–eye error model

In order to use the stereo cameras of a robot system, the position and orientation of the camera frame with respect to the gripper frame must be known. The process of determining the relationship is referred to as hand–eye calibration. We can find this relationship by moving the robot and observing the resulting motion of the cameras. As shown in Fig. 2, the problem of hand–eye calibration yields a homogeneous matrix equation of the form $\mathbf{CX} = \mathbf{XB}$. Many hand–eye calibration algorithms[21–30] were proposed in the past to solve the hand–eye parameter matrix $\mathbf{X}$.

To model the position error of a hand–eye robot system, we also use the transformation $\mathbf{Z}$ from the calibration plane frame to the robot-base frame. In Fig. 2, we have the relationship

$$\mathbf{Z} = \mathbf{B}_j\mathbf{X}\mathbf{C}_j. \tag{14}$$

Though Zhuang *et al.*[28] and Dornaika and Horaud[29] proposed their own algorithms to solve $\mathbf{X}$ and $\mathbf{Z}$ simultaneously, it is not convenient to apply them. Therefore, first we determine $\mathbf{X}$ in Tsai and Lenz's method,[22] and then solve for $\mathbf{Z}$ as a mean solution according to Eq. (14).

As shown in Fig. 3, the position error of the hand–eye configuration is formulated as

$$f_e = \sum_{k=1}^{m}\sum_{j=1}^{n}\|\mathbf{Z}\tilde{\mathbf{X}}^k_p - \mathbf{B}_j\mathbf{X}\mathbf{C}_j\tilde{\mathbf{X}}^k_p\|, \tag{15}$$

where $\tilde{\mathbf{X}}^k_p (k = 1, 2, \ldots, m)$ is the reference object point's 3D homogeneous coordinate; $\mathbf{B}_j (j = 1, 2, \cdots, n)$ is the jth robot pose obtained from the robot controller; $\mathbf{C}_j (j = 1, 2, \cdots, n)$ is the jth camera pose, which is solved by expanding the matrix $\mathbf{P}$ (in Section 2.2).

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

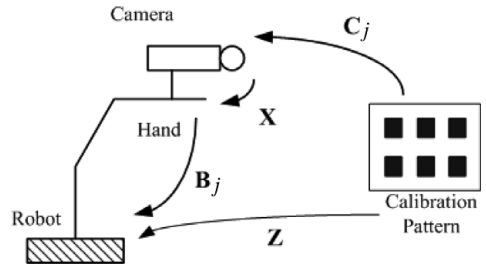


Fig. 3. Geometry of a hand–eye robot system.

Except for the position error of the hand–eye configuration, the error of the whole robot system also includes the projection error of stereo cameras, which is formulated as

$$f_c = \sum_{i=1}^{2}\sum_{k=1}^{m}\sum_{j=1}^{n} \|\tilde{\mathbf{u}}_k^i - \check{\mathbf{u}}(\mathbf{A}_i, \mathbf{H}_c^i, \mathbf{r}_{1j}, \mathbf{r}_{2j}, \mathbf{d}_j, \tilde{\mathbf{X}}_p^k)\|, \quad (16)$$

where $\tilde{\mathbf{u}}_k^i$ is the image of $\tilde{\mathbf{X}}_p^k$ in the ith camera, and $\check{\mathbf{u}}(\mathbf{A}_i, \mathbf{H}_c^i, \mathbf{r}_{1j}, \mathbf{r}_{2j}, \mathbf{d}_j, \tilde{\mathbf{X}}_p^k)$ is the back projection of $\tilde{\mathbf{X}}_p^k$ onto the ith camera's image plane at the jth robot pose. Therefore, the error formula of the whole hand–eye robot system is described as

$$\mathbf{F} = \mu f_c + (1-\mu) f_e, \quad (17)$$

where μ is the weighting factor, $0 < \mu < 1$.

3.3. Joint calibration algorithm

The recommended system calibration algorithm procedure combining camera and hand–eye calibrations is as follows:

1. Take n images of a static calibration plane at different orientations by moving the robot hand where two cameras are mounted, meanwhile read the robot pose $\mathbf{B}_j (j = 1, 2, \cdots, n)$ from the robot controller.
2. Compute the homography matrix $\mathbf{H}_j^i (i = 1, 2; j = 1, 2, \ldots, n)$ from the images, estimate the corresponding scale factor λ_j^i and the camera intrinsic matrix $\mathbf{A}_i$, then construct the measurement matrix $\mathbf{M}$.
3. Perform SVD to matrix $\mathbf{M}$, i.e., $\mathbf{M} = \mathbf{USV}^T$, $\mathbf{S} = \text{diag}(\sigma_1, \sigma_2, \sigma_3, \sigma_4)$, factorize $\mathbf{M}$ into $\mathbf{C}' = \mathbf{US}^{1/2}$ and $\mathbf{P}' = \mathbf{S}^{1/2}\mathbf{V}^T$, then recover the real $\{\mathbf{C}, \mathbf{P}\}$ from Eq. (13).
4. Expand matrix $\mathbf{P}$ (Section 2.2) with the orthogonal constraints in Section 2.1, solve the parameter matrixes $\mathbf{X}$ and $\mathbf{Z}$ by using matrix $\mathbf{P}$ and the robot pose data.
5. Refine all parameters $(\mathbf{A}_i, \mathbf{H}_c^i, \mathbf{H}_{pj}, \mathbf{X}, \mathbf{Z})$ by minimizing Eq. (17) by using the Levenberg–Marquardt algorithm.

If the motions of the cameras mounted on the robot hand have the rotation axes that are parallel to the normal of the calibration board, our joint calibration algorithm may not give the right solutions. To avoid the degenerated configurations, we can control the robot to perform the right motions that are suitable for both camera calibration and hand–eye calibration.

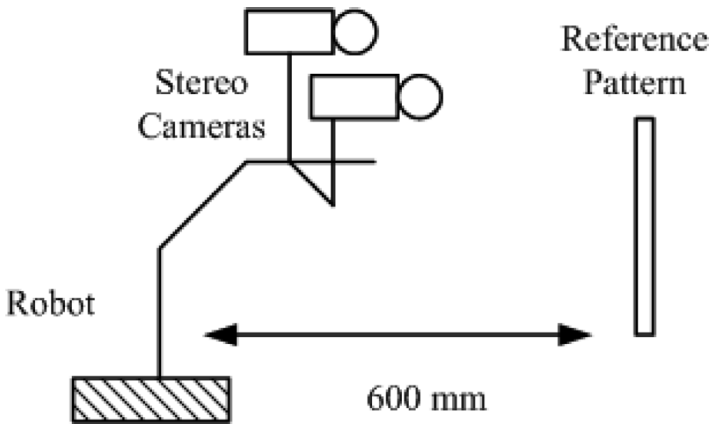


Fig. 4. Spatial configuration of the simulations.

4. Experiments

In this section, our proposed algorithm has been tested through both simulations and real data experiments. To compare its performance, we implemented two additional methods in the literature. The first method integrates Zhang's single camera calibration algorithm[7] with Sturm's pose estimation method[16] to obtain the camera's parameters and the plane's pose parameters; it then uses the plane's parameters to compute the hand–eye matrix in nonlinear optimization with the result of Tsai and Lenz's method method[22] as initial values. The second method that we applied is based on Ueshiba and Tomita's method[15] to compute the cameras' parameters and the plane's pose parameters, and uses Tsai and Lenz's method method[22] to estimate the initial hand–eye parameters. Then, as before, we use nonlinear optimization to give final results. In the following experiments, our joint calibration algorithm is denoted by "*JC*", the Zhang–Sturm–Tsai-style method is denoted by "*ZST*," and the Ueshiba–Tsai-style method is denoted by "*UT*."

4.1. Computer simulations

The spatial configuration in the simulations is shown in Fig. 4. The two cameras mounted on the robot hand are placed along a straight line at 150-mm interval, and the reference plane has a 600-mm distance to the robot base. The intrinsic parameters of the cameras are set to common values: $f_x = f_y = 900, u_0 = 385, v_0 = 288, s = 0.00$. The reference plane is painted with 6×8 reference points at 30-mm intervals. We establish nine motions of the robot and capture 9×2 images of the reference plane. Independent Gaussian noise with 0 mean is added to the captured image points while varying its standard deviation at various levels. At each noise level, 30 trials ($\mu = 0.5$) are made and average results are displayed.

Figure 5 shows average errors in the estimated intrinsic parameters of the two cameras at different noise levels. It can be seen that the *JC* method yields better results than the *ZST* method and the *UT* method in the focal length and the principle point. As the objective function $\mathbf{F}$ of the *JC* method has more parameter constraints than the other two methods for comparison, the influence of noise perturbation in the *JC* method is smaller.

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

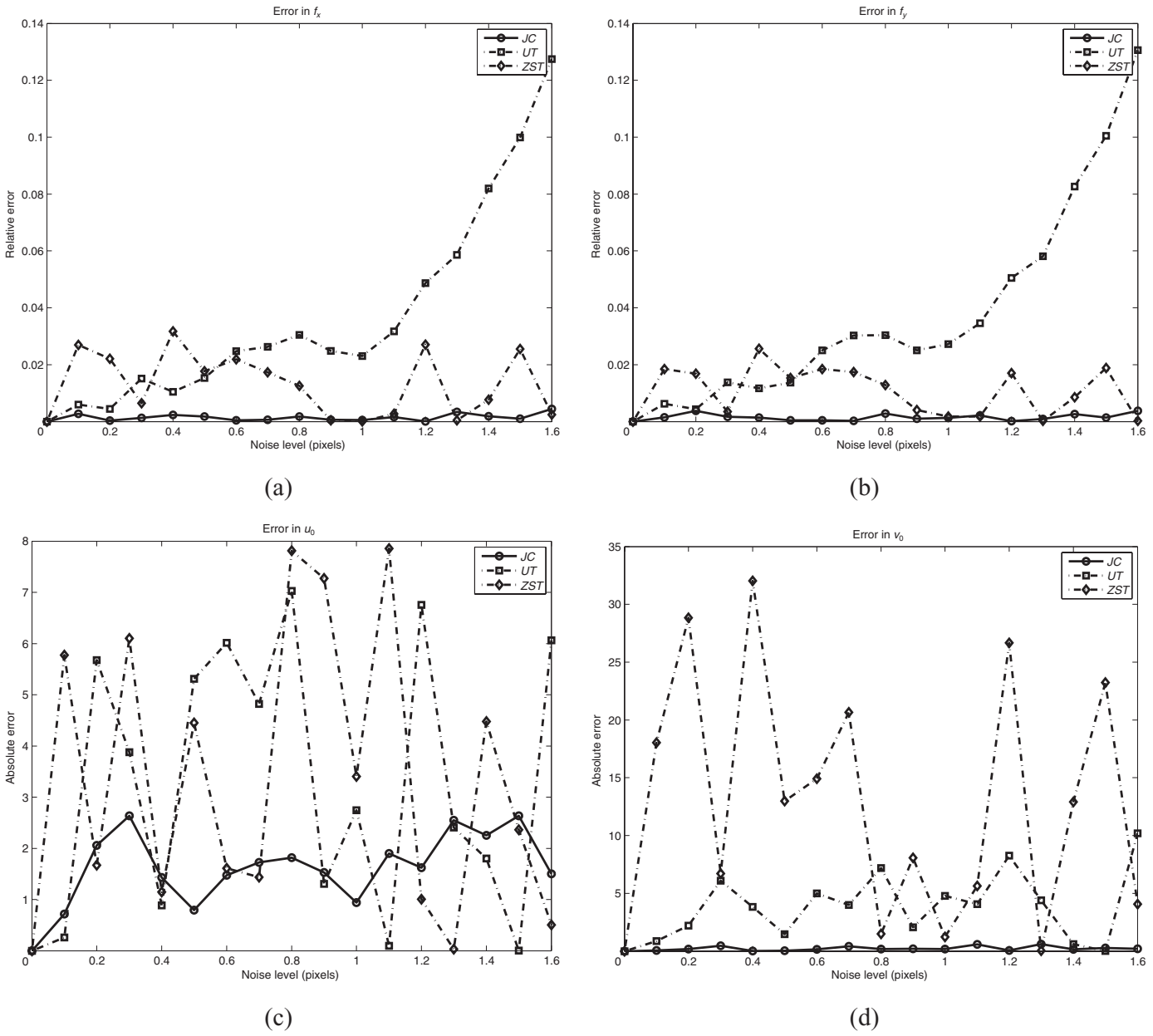


Fig. 5. Errors in focal length ((a) f_x and (b) f_y) and principle points ((c) u_0 and (d) v_0) vs. the noise level of vthe image points.

In Fig. 6 the rotation and translation errors of stereo and hand–eye parameters are shown as the function of noise levels. As shown in Fig. 6(a), the *JC* method's rotation errors in hand–eye parameters are not the smallest. This is because our method does not give the good estimation of hand–eye matrix as the initial values for nonlinear refinement through the expanded P matrix. Therefore, it is very important to choose a right method to estimate the hand–eye matrix through the expanded P matrix. However, the C matrix from factorization gives the good estimation of stereo parameters for nonlinear refinement. Figure 6(c) shows that the rotation errors of our method in stereo parameters are the smallest.

The *JC* method solves all the parameters of a hand–eye robot system simultaneously by using nonlinear optimization, and achieves much better performance, but at the cost of suboptimizing hand–eye calibration results.

4.2. Real data experiments

The real data experiments are conducted with two CCD cameras mounted on the last joint of a MOTOMAN CYR-UPJ Robot (Fig. 7). The two cameras with lenses of 8-mm focal length were calibrated. The image solution is 768×576. The reference plane has 6×8 square grids. With robot moving to seven different positions, the robot motions and the two cameras' images of reference plane were obtained. We applied *JC*, *ZST*, and *UT* algorithms to all the data, and got the intrinsic parameters of the two cameras: the stereo parameters and the hand–eye parameters. To show the performance of all three algorithms (*JC*, *ZST*, and *UT*) completely, we considered the hand–eye system as a whole and did the dot-hit experiments by using the three algorithms' results.

As shown in Fig. 8, we moved the gripper of the robot to hit the crossing points (dots) on the calibration pattern. When

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

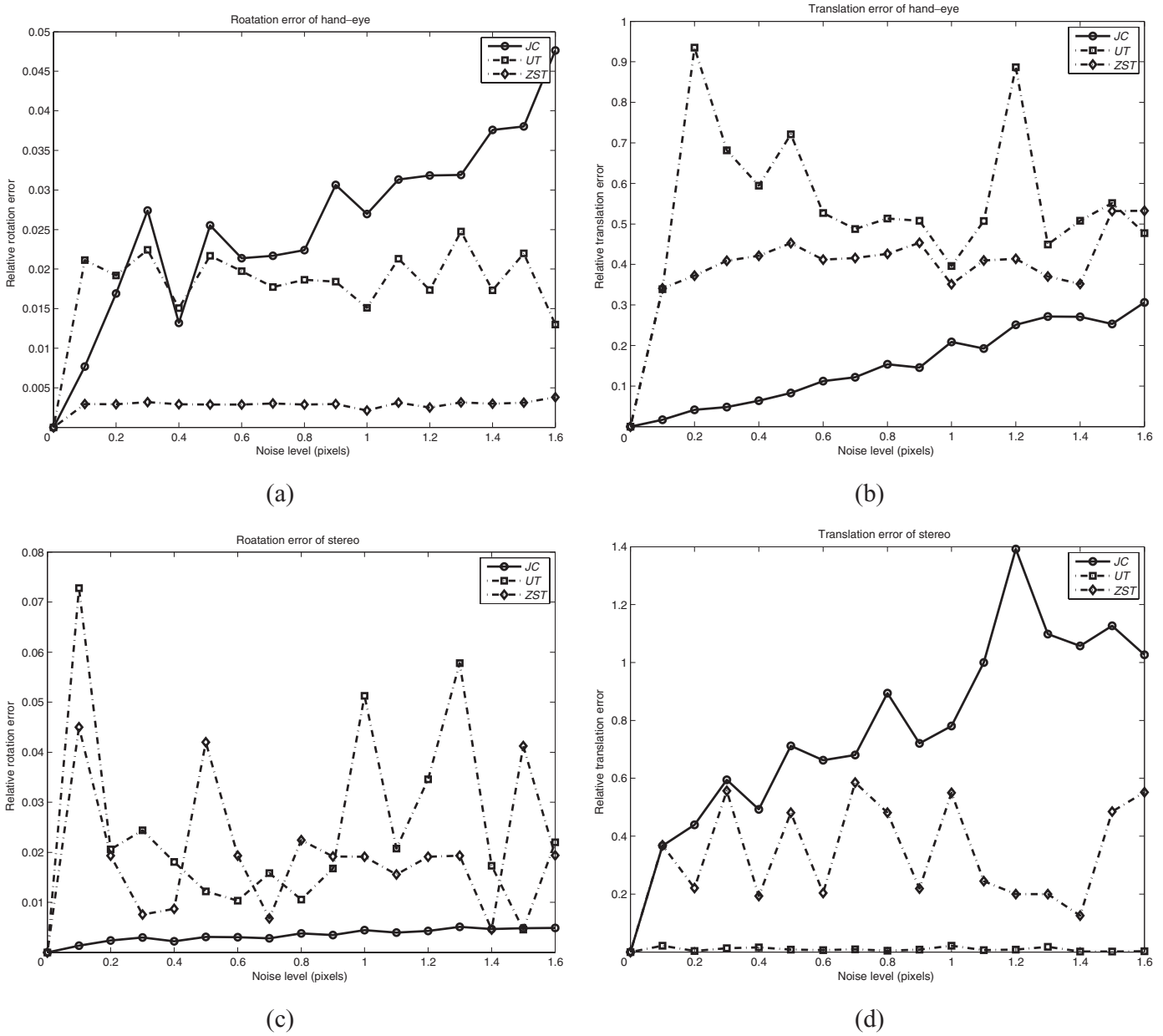


Fig. 6. Rotation and translation errors in the hand–eye parameters ((a) and (b)) and the stereo parameters ((c) and (d)) vs. the noise level of image points.

the gripper touched the crossing points on the calibration pattern, the coordinates of the points in the robot base $\vec{\mathbf{D}}_R^i (i = 1, 2, \ldots, n)$ were read from the robot controller. We then reconstructed the positions of the points through the position probe in the stereo algorithm. The position probe constructed the coordinates of the points in the camera coordinate frame $\vec{\mathbf{D}}_c^i (i = 1, 2, \ldots, n)$. According to Eq. (14), we could also get the coordinates of the points in the robot base frame

$$\tilde{\mathbf{D}}_r^i = \mathbf{B}_i \mathbf{X} \tilde{\mathbf{D}}_c^i, \tag{18}$$

where $\mathbf{B}_i$ is the transformation from the gripper frame to the robot base frame at the ith position, $\tilde{\mathbf{D}}_r^i = (\vec{\mathbf{D}}_r^{iT}, 1)^T$, and $\tilde{\mathbf{D}}_c^i = (\vec{\mathbf{D}}_c^{iT}, 1)^T$. We could consider $\vec{\mathbf{D}}_R^i (i = 1, 2, \ldots, n)$ as the ground truth, and compare them with $\vec{\mathbf{D}}_r^i (i = 1, 2, \ldots, n)$ to assess the performance of calibration methods with two task-dependent methods.

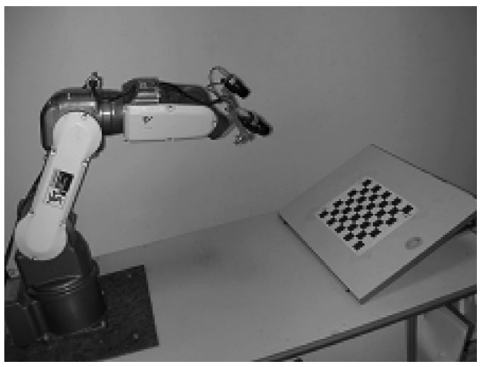


Fig. 7. The physical setup of a hand–eye robot system.

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press



Fig. 8. The dot-hit experiments with the position probe.

In the first experiment, the robot gripper was moved to hit 20 crossing points of the calibration pattern, and the distance D_g between these crossing points was the multiple of a standard length ($M \times 30$ mm, $M = 1, 3, 5, 7$) in the robot base frame. Through the position probe, we could also compute distance D_m between the square points by using $\vec{\mathbf{D}}_r^i (i = 1, 2, \ldots, 20)$. The relative position errors $\|D_g - D_m\|/M$ reflected the performance of the three algorithms. As shown in Fig. 9, the relative errors of the three algorithms are very close, but the mean error of our *JC* algorithm is smaller than those of the other two algorithms.

The second experiment assesses the performances of the three algorithms using the dot-hit target picture. We chose five crossing points on the reference plane and moved the robot's gripper to hit them. For every crossing point, we read $\vec{\mathbf{D}}_R^i$ from the robot controller as the ground truth and computed $\vec{\mathbf{D}}_r^i$ for seven times as the measurement values. Plotting them on the same figure, we got the dot-hit target picture (see Fig. 10), which shows the absolute position errors of the calibration algorithms. As shown in Fig. 10, if the hand–eye robot system was calibrated by the *ZST* algorithm, the dot-hit $\vec{\mathbf{D}}_r^i$ is the closest value to the ground truth $\vec{\mathbf{D}}_R^i$. The dot-hit $\vec{\mathbf{D}}_r^i$ of the *JC* algorithm is not the most accurate one because the hand–eye rotation performance of the *JC* algorithm is worse than that of the *ZST* algorithm. The *UT* algorithm has the worst results because of its bad performance in both stereo and hand–eye parameters.

For a hand–eye robot system, the hand–eye parameters are very important to its positioning accuracy. Therefore, we applied three hand–eye calibration methods[22,25,30] in our proposed algorithm to estimate the initial hand–eye matrix, and did the dot-hit experiment to show which hand–eye calibration method was suitable for our algorithm. The dot-hit target picture (Fig. 11) shows that Tsai and Lenz's method[22] has smaller positioning errors than the other two methods. That is why we chose Tsai and Lenz's method[22] to estimate the initial hand–eye matrix in our *JC* algorithm.

In the real experiments, the *JC* algorithm was not the algorithm that made the robot system's positioning errors the smallest, but it needed only one operation of nonlinear optimization to compute all the camera and hand–eye parameters simultaneously with a smaller cost of computation than *ZST* and *UT*. Therefore, it is very convenient and practical to use the *JC* algorithm to calibrate a hand–eye robot system in many applications.

4.3. Real application

To clarify the merits of the *JC* method, we applied it in a robot-assisted surgical system (Fig. 12(a)), which has a hand–eye robot for total knee arthroplasty. After calibrating the whole hand–eye robot system using the *JC* algorithm, we used the robot to do the sawbone (phantom bone) operations. After cutting a number of sawbone models by the robot, we computed the distribution histogram (Fig. 12(b)) of cutting errors in the operations. As shown in Fig. 12, most (90%) of

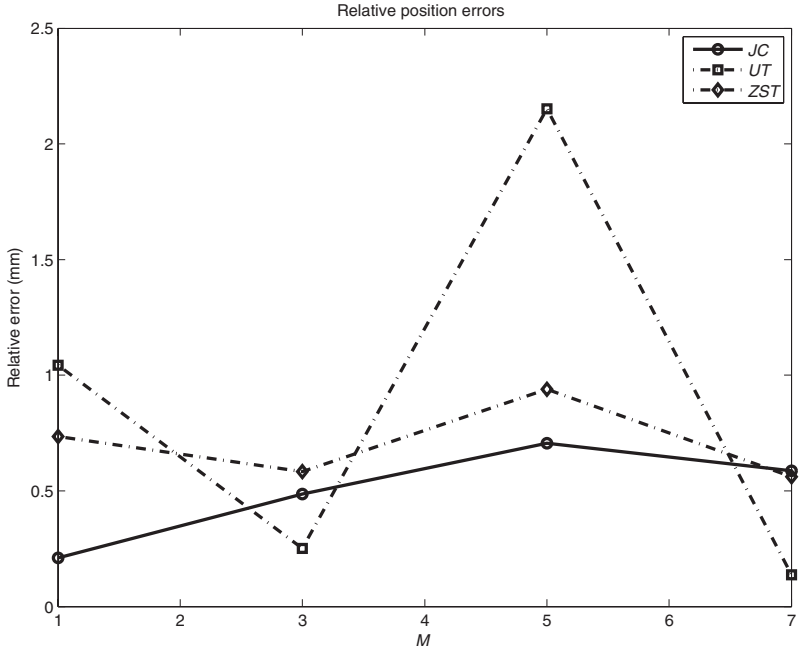
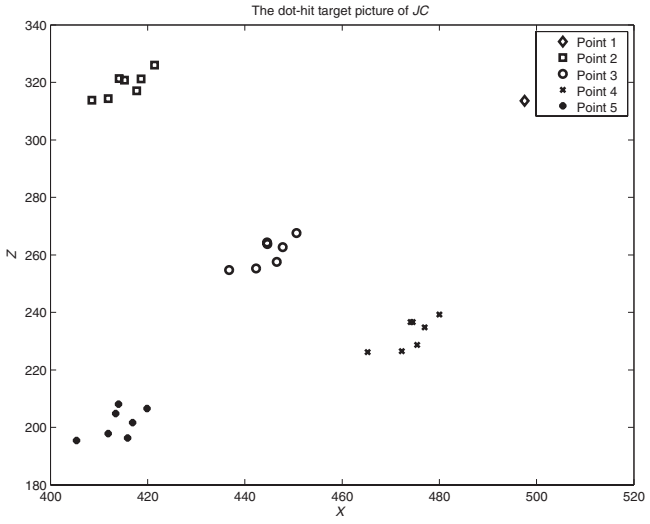
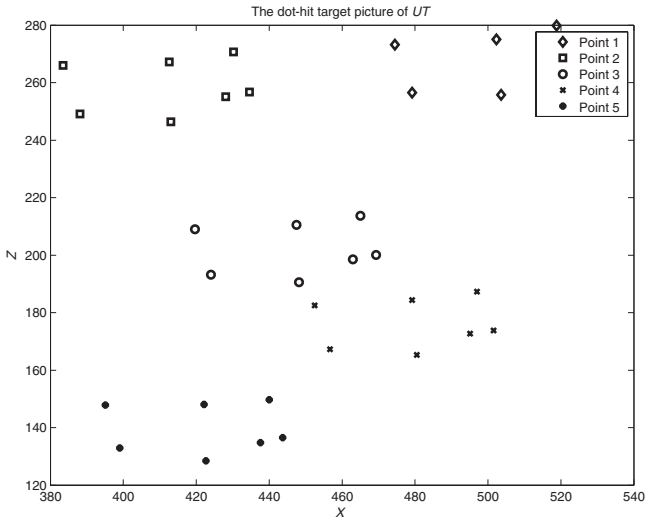


Fig. 9. The relative position errors in the dot-hit experiment ($\mu = 0.5$).

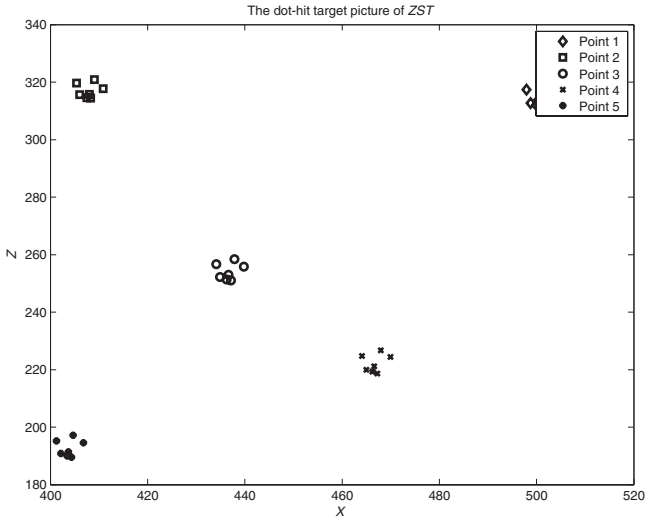
https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press



(a)

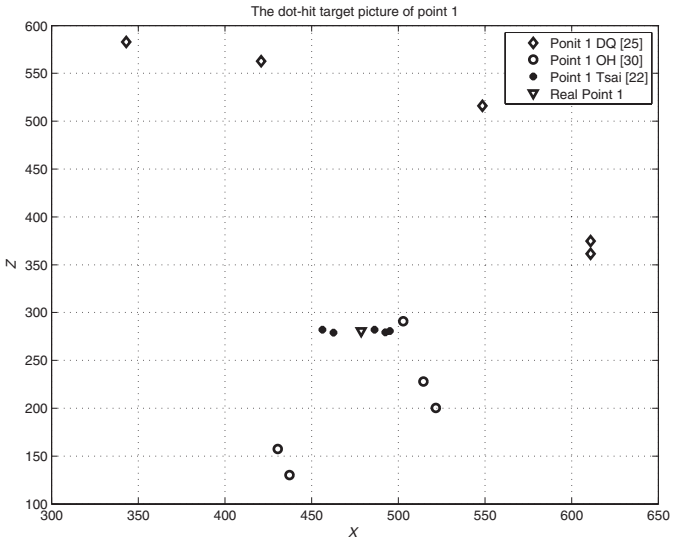


(b)

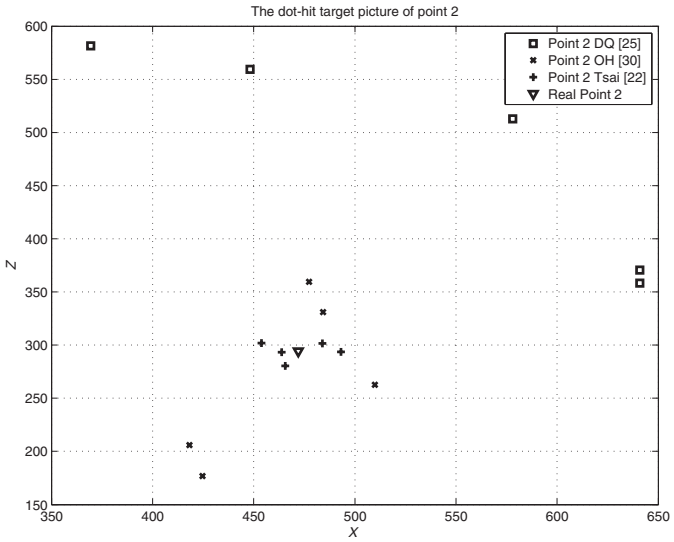


(c)

Fig. 10. The dot-hit target pictures of (a) *JC*, (b) *UT*, and (c) *ZST*. The five points have the real position coordinates (410, 320), (500, 320), (440, 260), (470, 230), and (410, 200) in the *XZ* plane($\mu = 0.5$).



(a)



(b)

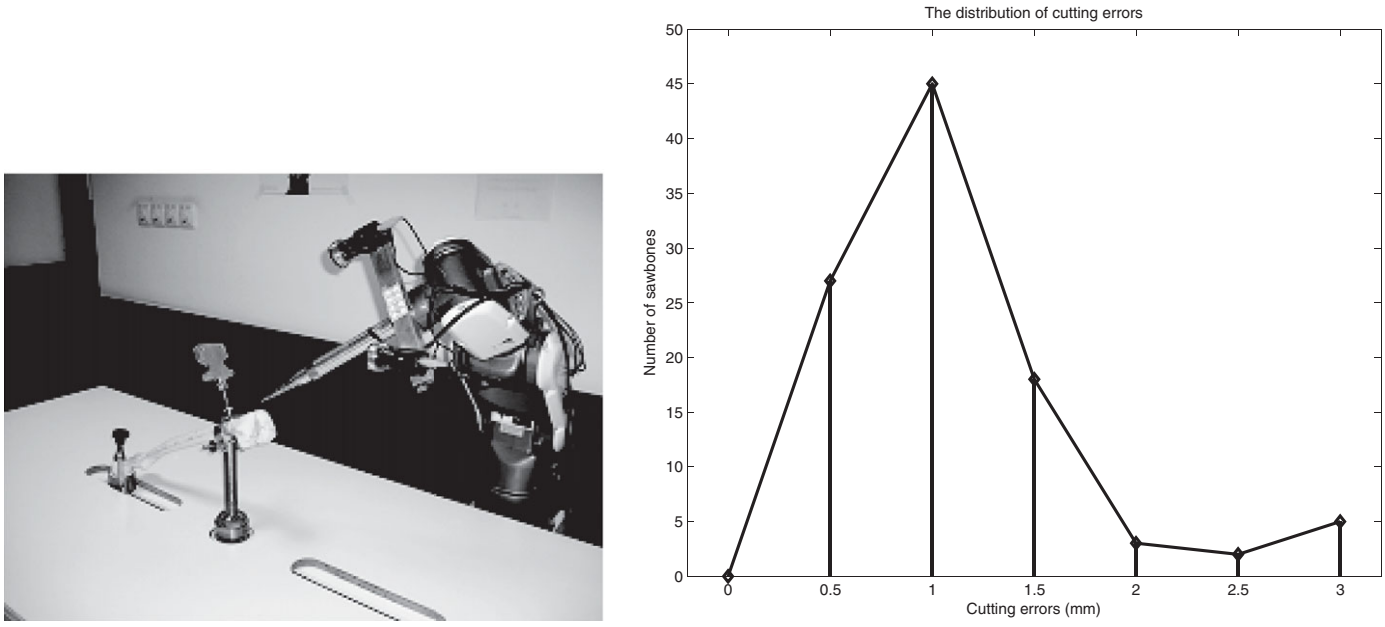
Fig. 11. The dot-hit target pictures of testing the hand–eye estimation algorithms in *JC* ($\mu = 0.5$).

the sawbone models have the cutting errors of less than 2 mm, which are less than the errors allowed in clinical surgeries.

5. Conclusions

In this paper, we proposed a system calibration algorithm which calibrated the hand–eye vision robot system as a whole system to compute the cameras' parameters and the hand–eye parameters simultaneously using nonlinear optimization. We also implemented two other methods in the literature, the first denoted by *ZST*, and the second denoted by *UT*. We compared three algorithms in the simulations where we varied the noise level in image points. Our joint calibration algorithm showed better performance in the simulations. Applying the methods in the dot-hit experiments, the results of the real experiments showed the validity of our proposed algorithm in practice. With Tsai and Lenz's[22] method for initial guess of hand–eye

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press



(a) (b)

Fig. 12. (a) The robot-assisted surgical system; (b) the distribution of cutting errors.

parameters, the proposed *JC* method gave promising results of vision robot system calibration.

Acknowledgments

This work is supported by the Independent Innovation Foundation of Shandong University (2012GN042), and the Natural Science Foundation of China (60975025, 61273277).

References

1. R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the shelf TV cameras and lens," *IEEE J. Robot. Autom.* **3**(4), 322–344 (Aug. 1987).
2. O. Faugeras, *Three-Dimension Computer Vision: A Geometric Viewpoint* (MIT Press, Cambridge, MA, 1993).
3. D. Gennery, "Stereo-camera calibration," **In**: *Proceedings of the 10th Image Understanding Workshop*, Los Angeles, USA (1979) pp. 101–108.
4. S. Ganapathy, "Decomposition of transformation matrices for robot vision," *Pattern Recognit. Lett.* **2**, 401–412 (Dec. 1984).
5. R. K. Lenz and R. Y. Tsai, "Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology," *IEEE Trans. Pattern Anal. Mach. Intell.* **10**(5), 713–720 (1988).
6. G. Q. Wei and S. D. Ma, "A Complete Two-Plane Camera Calibration Method and Experimental Comparisons," **In**: *Proceedings of the 4th International Conference on Computer Vision*, Berlin, Germany (1993) pp. 439–446.
7. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334 (2000).
8. Z. Zhang, "Camera calibration with one-dimensional objects," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(7), 892–899 (2004).
9. P. Sturm and S. Maybank, "On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications," **In**: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Ft. Collins, USA (Jun. 1999) pp. 432–437.
10. F. C. Wu, Z. Y. Hu and H. J. Zhu, "Camera calibration with moving one-dimensional objects," *Pattern Recognit.* **38**(5), 355–365 (2005).
11. S. J. Maybank and O. D. Faugeras, "A theory of self-calibration of a moving camera," *Int. J. Comput. Vis.* **8**(2), 123–152 (Aug. 1992).
12. R. I. Hartley, "An Algorithm for Self-Calibration from Several Views," **In**: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Champaign, USA (Jun. 1992) pp. 908–912.
13. Q. T. Luong and O. Faugeras, "Self-calibration of a moving camera from point correspondences and fundamental matrices," *Int. J. Comput. Vis.* **22**(3), 261–289 (1997).
14. B. Triggs, "Autocalibration from Planar Scenes," **In**: *Proceedings of the 5th European Conference on Computer Vision*, Freiburg, Germany (Jun 1998) pp. 89–105.
15. T. Ueshiba and F. Tomita, "Plane-Based Calibration Algorithm for Multi-Camera Systems via Factorization of Homography Matrices," **In**: *Proceedings of the 9th International Conference on Computer Vision*, Beijing, China (2003) pp. 966–973.
16. P. Sturm, "Algorithms for Plane-Based Pose Estimation," **In**: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Dublin, Ireland (2000) pp. 706–711.
17. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision* (Cambridge, UK: Cambridge University Press, 2000).
18. C. Tomasi and T. Kanade, Shape and motion from image streams under orthography: A factorization method," *Int. J. Comput. Vis.* **9**(2), 137–154 (1992).
19. L. Zelnik-Manor and M. Irani, "Multiview constraints on homographies," *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(2), 214–223 (2002).
20. H. Malm and A. Heyden, "Simplified Intrinsic Camera Calibration and Hand-Eye Calibration for Robot Vision," **In**: *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligence Robots and Systems*, Sendai, Japan (2003) pp. 1037–1043.
21. Y. C. Shiu and S. Ahmad, "Calibration of wrist-mounted robotics sensors by solving homogeneous transform equations of the form AX=XB," *IEEE Trans. Robot. Autom.* **5**(1), 16–27 (1989).
22. R. K. Tsai and R. Y. Lenz, "A new technique for fully autonomous and efficient 3D robotics hand/eye calibration," *IEEE Trans. Robot. Autom.* **5**(3), 345–358 (1989).

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press

23. H. Zhuang and Y. C. Shiu, "A noise-tolerant algorithm for robotic hand–eye calibration with or without sensor orientation measurement," *IEEE Trans. Syst. Man Cybern.* **23**(4), 1168–1175 (1993).
24. R. Horaud and F. Dornaika, "Hand-eye calibration," *Int. J. Robot. Res.* **14**(3), 195–210 (1995).
25. K. Daniilidis, "Hand-eye calibration using dual quaternions," *Int. J. Robot. Res.* **18**(3), 286–298 (1999).
26. Z. Zhao and Y. Liu, "Hand-Eye Calibration Based on Screw Motions," **In**: *Proceedings of the 18th International Conference on Pattern Recognition*, Hong Kong, China (2006) pp. 1022–1026.
27. 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*, Maui, USA (1991) pp. 145–151.
28. H. Zhang, Z. Roth and R. Sudhakar, "Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation of the form AX=YB," *IEEE Trans. Robot. Autom.* **10**(4), 549–554 (1994).
29. F. Dornaika and R. Horaud, "Simultaneous robot-world and hand-eye calibration," *IEEE Trans. Robot. Autom.* **14**(4), 617–622 (1998).
30. N. Andreff, R. Horaud and B. Espian, "On-Line Hand-Eye Calibration," **In**: *Proceedings of the 2nd International Conference on 3-D Digital Imaging and Modeling*, Ottawa, Canada (1999) pp. 430–436.
31. B. K. P. Horn, H. M. Hilden and S. Negahdaripour, "Close-form solution of absolute orientation using orthonormal matrices," *J. Opt. Soc. Am.* **5**(7), 1127–1135 (1988).
32. Z. Zhao, "Hand-Eye Calibration Using Convex Optimization," **In**: *Proceedings of the International Conference on Robotics and Automation*, Shanghai, China (2011) pp. 2947–2952.

Appendix: Estimation of the camera intrinsic parameters

There are many ways to estimate the camera intrinsic parameters. Here we present a simple method based on the geometric relationship between the camera and the calibration board. If the image solution is $m \times n$, the principle point (u_0, v_0) can be estimated as

$$u_0 = m/2\ ,\ v_0 = n/2.$$

Suppose the metric distance between the optical center of the camera and the pattern points is D_F, the metric length of a line segment (paralleled to the image plane) on the calibration pattern board is D_m and the length of it in pixel is D_p. From the optical geometry of the camera, we get

$$f/D_F = D_p d/D_m,$$

where f is the nominal focal length of the camera, and d is the metric size of a pixel in the image. Therefore, the focal length parameters (f_x, f_y) can be obtained as

$$f_x = s_x D_F D_p/D_m, \quad f_y = s_y D_F D_p/D_m,$$

where $s_x = 2m/(m+n)$ and $s_y = 2n/(m+n)$. Without loss of generality, we can set the skewness parameter s as zero.

https://doi.org/10.1017/S0263574713000040 Published online by Cambridge University Press