TR-94-006

Kinematic Parameter Indentification of a Binocular Head and Its Error Analysis

Sheng-Wen Shih, Yi-Ping Hung
Wei-Song Lin

Kinematic Parameter Identification of a Binocular Head and Its Error Analysis

Sheng–Wen Shih[2], Yi–Ping Hung[1], Wei–Song Lin[2]

[1]Institute of Information Science, Academia Sinica, Nankang, Taipei, Taiwan
[2]Institute of Electrical Engineering, National Taiwan University, Taipei, Taiwan
e-mail: hung@iis.sinica.edu.tw

ABSTRACT

This report proposes a new closed–form solution using a single calibration point for identifying the kinematic parameters of an active binocular head, which is based on the complete and parametrically continuous (CPC) kinematic model. This method can be applied to any kind of kinematic parameter identification problems with or without multiple end–effectors, providing that the links are rigid, the joints are either revolute or prismatic and no closed–loop kinematic chain is included. However, as a practical example, this report focuses on the calibration of a binocular head having four revolute joints and two prismatic joints. In general, it is more difficult to estimate the end–effectors poses (both positions and orientations) than to estimate their positions only. Therefore, this method chose to use the 3D position measurements of a point on the end–effector in the design of the algorithm for kinematic parameter identification. Theoretical analysis for the estimation error is included. This analysis gives us a method to reduce the estimation error by controlling the factors found in the derived error variances of the estimated parameters. Simulation and real experiments have shown that the proposed method of using point measurements can achieve much higher accuracy than that of using pose measurements.

Keywords: Kinematic Calibration, Binocular Head, CPC Kinematic Model, Active Vision

I. INTRODUCTION

Many computer vision problems that are ill–posed, nonlinear or unstable for a passive observer become well–posed, linear or stable for an active observer[2]. Being able to acquire information actively, the active vision system has more potential applications than a passive one has. In an active stereo vision system, the cameras are able to preform functions such as gazing, panning and tilting. To perform experiments on active vision, we have built a binocular head (referred to as the IIS head). This IIS head has four revolute joints and two prismatic joints, as shown in Fig. 1. The two joints on top of the IIS head are for camera vergence or gazing (referred to as joint $5L$ and joint $5R$). The next two joints below them are for tilting and panning the stereo cameras (referred to as joint 4 and joint 3). All of the above four joints are revolute, and are mounted on an X–Y table which is composed of two prismatic joints (referred to as joint 2 and joint 1). Since the IIS head is built with off–the–shelf components, its kinematic parameters are unknown. Our goal is to calibrate its kinematic parameters for controlling the orientations and positions of the stereo cameras.

We have been considering two approaches before using the one proposed in this report. The first approach combines the kinematic calibration problem together with the head/eye calibration problem. Methods proposed by Zhuang and Roth[23], Shih et. al. [15] and Young et. al.[21] can be used to solve the combined problem. However, there are two major problems encountered when

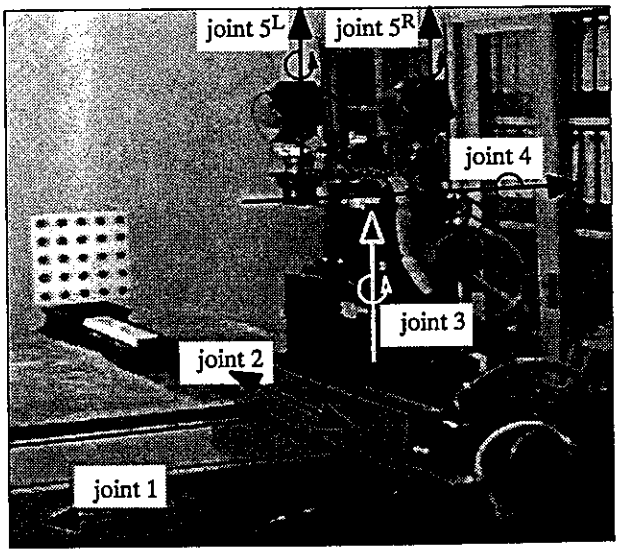

Fig. 1. A picture of the IIS head

adopting this approach. The first problem is that it is necessary to determine the poses of the cameras on the binocular head with respect to a calibration object. With our experience, the pose estimation techniques available in the literatures are not good enough for obtaining highly accurate estimates of the camera poses, especially when the distance between the camera and the calibration object is larger than 1 meter. The second problem is that it is difficult to keep the calibration object in the field of view while rotating the cameras. Even if we can keep the calibration object in the field of view while rotating the cameras, it will restrict the motion range in the calibration. Hence, after the calibration, the binocular head would have large probability of working outside the small calibration range, which may lead to larger kinematic inaccuracy in general. There is no such problem of keeping calibration object in the field of view in [10], where offset caused by the revolute motion can be compensated by the prismatic joint. However, since there is no vertical translation joint in our IIS head, there is no way to compensate the tilt axis rotation to keep the calibration object in the field of view. In addition to the two problems, we may occasionally need to change the zoom and focus setting of the lens which would result in a new camera coordinate system. With the first approach, the tedious procedure for calibrating the head kinematics is needed for obtaining the new head/eye relation. To avoid the above problems, we chose the second approach described below for calibrating our binocular head.

The second approach is a two–stage approach. At the first stage, the two cameras on the IIS head

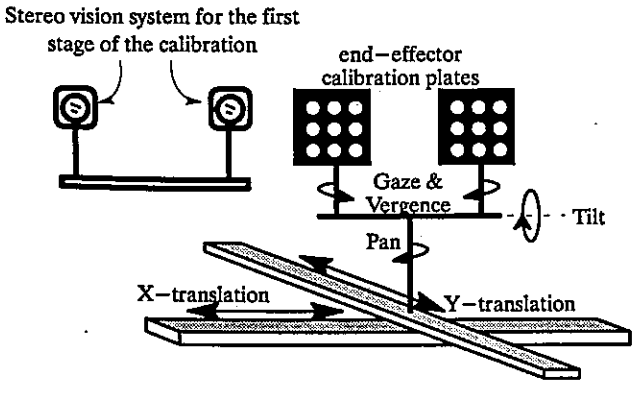

Fig. 2. The schematic diagram of the setup for the first stage of the calibration process

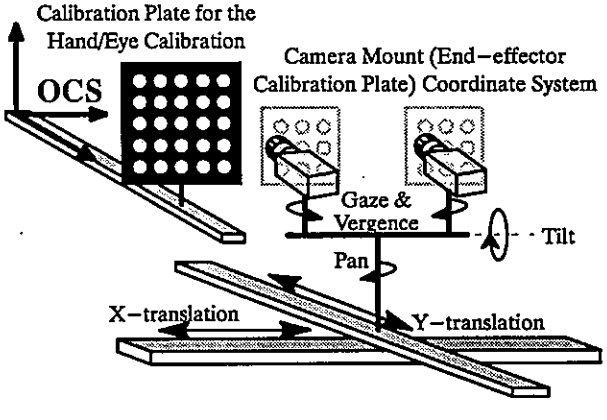

Fig. 3. The schematic diagram of the setup for the second stage of the calibration process

are replaced by two small end–effector calibration plates having nine circles (see Fig. 2). The positions of the calibration plates, or equivalently, of the end–effectors, can be estimated using the stereo vision measurement system or other more accurate 3D measurement system, such as the theodolite, CMM (Coordinate Measuring Machine), ..., etc[12]. Since the stereo vision measurement system is set apart from the binocular head and look back to the calibration plates mounted on the head (as shown in Fig. 2), it is much easier to keep the calibration plates in the field of view while moving the joints of the head. The acquired positions of the calibration plates are then used to calibrate the kinematic parameters. In the second stage, the cameras are remounted to the IIS head as show in Fig. 3. We can then use either of the methods proposed by Tsai and Lenz[18] or Shih et. al.[14] to calibrate the head–eye relation. After the two–stage calibration, the robot kinematic and the head/eye relation are combined to provide a complete kinematic model for motion control. This report will focus on the first stage of the calibration, i.e., the head calibration. However, the method can be applied to the kinematic calibration of general robots.

Existing techniques for robot kinematic calibration can be classified into two categories depending upon that the solution is in closed–form or in iterative–form. Most of the iterative techniques are based on the linearized error kinematic model [4][5][6][7][8][9][11][20][22]. Kinematic parameters are obtained from iteratively evaluating Jacobian matrices and solving linear equations by using least square techniques. This kind of methods would provide accurate results if the initial value of the kinematic parameters are close enough to the true values and the calibration configurations are carefully selected such that the singular conditions of the Jacobian matrices are avoided. Iterative kinematic calibration problem can also be solved as a large–scaled nonlinear optimization problem, as stated in [1].

Another category of calibration techniques provide closed–form solutions. Both the methods proposed by Zhuang[23] and Shih[15] give direct solution to the CPC kinematic parameters[22], by using the end–effector's poses measurements. However, most of the other closed–form solutions, e.g., [10][16][17] [21], do not estimate the kinematic parameters directly. Instead, the kinematic

parameters were extracted from the estimations of the orientation and location of the joint axes. Stone[16] and Sklar[17] estimate the joint axis from the measurements of the 3D coordinates of a calibration point attached to the robot arm, while moving each robot joint one at a time. When moving a revolute joint, the trajectory of the calibration point forms a 3D circle. First, a plane is fitted to the measured calibration points by least–square technique. The resulted plane normal defines the orientation of the revolute joint axis. The measured calibration points are then projected along the plane normal onto the plane. A 2D circle is then fitted, again by least–square technique, to find its center which defines the location of the joint axis. In this approach, the orientation and location of the joint axis are separately estimated without using the information contained in the amount of rotation angles. Lenz and Tsai[10] and Young et. al.[21] estimate the joint axis from the measurements of the end–effector's poses. Lenz and Tsai's method[10] were developed for calibrating a Cartesian robot. When collecting the calibration data, Lenz and Tsai's method[10] allows all of the prismatic joints plus one revolute joint to move at a time. Young's method[21] employed a measurement strategy by which each joint is individually moved. However, in practice, it is more difficult to accurately estimate the end–effector's pose than to estimate the 3D coordinates of a single calibration point on the end–effector.

In this report, we shall present a closed–form solution based on the CPC kinematic model using single calibration point. The advantages of using the CPC kinematic model are described in [22] and [23]. Four main features of our new method are described below:

1). With our new method, the kinematic parameters estimation problem is decomposed into many subproblems of single joint axis. Hence, the complexity is reduced and an easier implementation is derived. Also, this decomposition leads to a general solution for any robot with arbitrary combination of prismatic and revolute joints. Notice that this does not mean that only one joint can be moved at a time. Instead, all the *calibrated* joints can be moved to gather more information.

2). It is easier to obtain an accurate point measurement than an accurate pose measurement. Moreover, the calibration object can be of smaller size when using point measurements, instead of pose

measurements, which allows larger range of joint movement in calibration and leads to more accurate and robust estimation of kinematic parameters. While most methods using single calibration point cannot provide the transformations from tool to end–effector and from world to base, as described in [12], our method can obtain the transformation from world to base by treating it as a link matrix (i.e., V_0 in Fig. 7). Once the transformation from world to base is known, the transformation from the last joint to the end–effector can be easily obtained.

3). With our method, each joint is calibrated in the order of from the base to the end–effectors. Therefore, it is very suitable for kinematic calibration of robots having multiple end–effectors. On the contrary, if we calibrate a multiple end–effector robot from the end–effectors to the base as those methods described in [15] and [23], then at the link having two branching kinematic chains, we need to estimate an additional transformation matrix for unifying the coordinates systems from different end–effectors[13].

4). Theoretical error analysis are provided such that optimal calibration setup can be determined by minimizing the derived theoretical error variance.

In addition to the development of an accurate approach for the kinematic parameter identification problem, this report has two other important contributions. The first one is that we have derived an accurate and robust 3D circle fitting method which uses the rotation angle information and is applied to the kinematic calibration problem for a revolute joint. The second one is that we have derived the theoretical error variances for the estimates of the kinematic parameters obtained by using our method. Computer simulations have shown that the derived theoretical error variance is consistent with the experimental results. This report is organized as follows. The kinematic calibration problem is formulated in section II. The new calibration method is described in section III. Results of the theoretical error analysis are shown in section IV. Experimental results are shown in section V. Conclusions are given in section VI.

II. PROBLEM FORMULATION

Consider Figs. 4 and 5. The CPC kinematic model we use for a revolute or prismatic joint is as

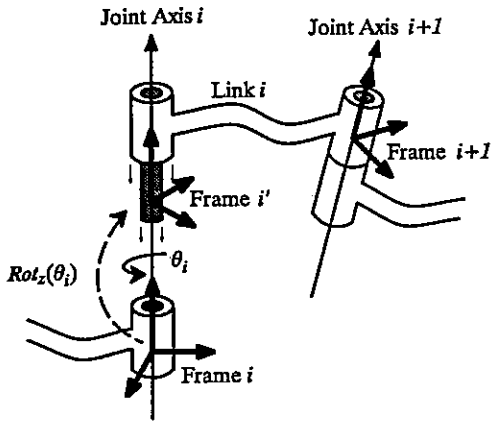

Fig. 4. CPC modeling convention for a revolute joint

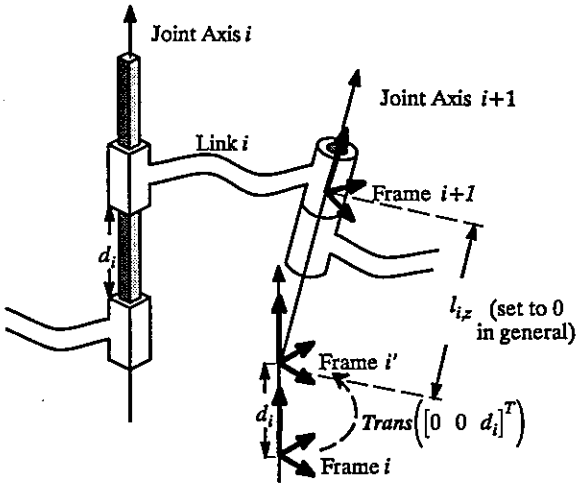

Fig. 5. CPC modeling convention for a prismatic joint

follows (see [22] for more details).

$$ {}^{i}T_{i+1} = {}^{i}T_{i'}\ {}^{i'}T_{i+1}, \tag{1} $$

where

$$ {}^{i}T_{i'} \equiv Q_i, \tag{2} $$

$$ {}^{i'}T_{i+1} \equiv V_i, \tag{3} $$

$$ Q_i = \begin{cases} Rot_z(q_i), & \text{for revolute joint,} \\ Trans\left([0\ 0\ q_i]^T\right), & \text{for prismatic joint,} \end{cases} \tag{4} $$

$$ Rot_z(\theta) \equiv \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, $$

$$ Trans\left([x\ y\ z]^T\right) \equiv \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix}, $$

$$ q_i = s_i\, q'_i,\ s_i \in \{+1,\ -1\}, \tag{5} $$

q'_i is the ith joint value,

$$ V_i = R_i\ Rot_z(\beta_i)\ Trans\left([l_{i,x}\ \ l_{i,y}\ \ l_{i,z}]^T\right), \tag{6} $$

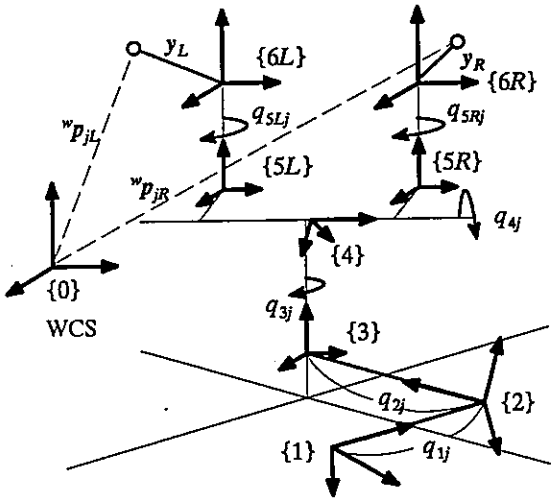

Fig. 6. The skeleton diagram of the IIS head.

and

$$R_i = \begin{bmatrix} 1 - \dfrac{b_{i,x}^2}{1+b_{i,z}} & \dfrac{-b_{i,x}b_{i,y}}{1+b_{i,z}} & b_{i,x} & 0 \\ \dfrac{-b_{i,x}b_{i,y}}{1+b_{i,z}} & 1 - \dfrac{b_{i,y}^2}{1+b_{i,z}} & b_{i,y} & 0 \\ -b_{i,x} & -b_{i,y} & b_{i,z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

Notice that we have made some modifications on the notations to accommodate our situation: calibrating a binocular head from base toward the end–effectors. Because the CPC convention requires that any two consecutive joint axes should have nonnegative inner product, i.e., $b_{i,z} \geq 0$. In general, this requirement can be achieved by changing the sign of one of the joint values of consecutive joints. This is because changing the sign of the joint value is equivalent to reversing the joint axis for both revolute and prismatic joints. Therefore, we have slightly modified the convention of the CPC model by including a sign parameter, s_i, as shown in equation (5). Furthermore, for convenience we have introduced an intermediate coordinate system between ith and $(i+1)$st frame, i.e., the i' frame in equation (1). From the frame i to the frame i', it is either a rotational or a translational motion matrix depending on the joint type, and from the frame i' to the frame i+1, it is a fixed link matrix (referred to as "shape" matrix in [22]) as defined in equation (6).

Different coordinate systems associated with different joints of the IIS head are shown in Fig. 6

The coordinate frames $\{6L\}$ and $\{6R\}$ are the end–effector coordinate systems. As shown in Fig. 7, we have the following relations:

$$\left.\begin{matrix} {}^{w}T_{6L}(q_j) \\ {}^{w}T_{6R}(q_j) \end{matrix}\right\} = {}^{w}T_1\,{}^{1}T_2(q_j)\,{}^{2}T_3(q_j)\,{}^{3}T_4(q_j)\left\{\begin{matrix} {}^{4}T_{5L}(q_j)\,{}^{5L}T_{6L}(q_j) \\ {}^{4}T_{5R}(q_j)\,{}^{5R}T_{6R}(q_j) \end{matrix}\right. \tag{8}$$

where $q_j = \left[q_{1j}\ \ q_{2j}\ \ q_{3j}\ \ q_{4j}\ \ q_{5Rj}\ \ q_{5Lj}\right]$ denotes the joint values for the jth robot configuration, ${}^{i}T_k$ denotes the transformation matrix from frame i to frame k. For convenience, each of the transformation matrix is written as a function of the joint value vector, q_j, instead of q_{ij}.

Suppose that the fixed coordinates of the calibration points mounted on the left and right end–effectors are y_L and y_R with respect to LECS (left end–effector coordinate system) and RECS (right end–effector coordinate system), respectively. Let ${}^{w}p_{jL}$ and ${}^{w}p_{jR}$ denote the coordinates of the calibration points, corresponding to the jth robot configuration q_j, measured in the WCS (world coordinate system). Transforming the coordinates of the calibration points, y_L and y_R, from LECS and RECS to the WCS, we have

$${}^{w}\tilde{p}_{jL} = {}^{w}T_{6L}(q_j)\,\tilde{y}_L\,, \tag{9}$$

and

$${}^{w}\tilde{p}_{jR} = {}^{w}T_{6R}(q_j)\,\tilde{y}_R\,, \tag{10}$$

where ${}^{w}\tilde{p}_{jL}$, ${}^{w}\tilde{p}_{jR}$, $\tilde{y}_L$ and $\tilde{y}_R$ are the 3D homogeneous coordinates of ${}^{w}p_{jL}$, ${}^{w}p_{jR}$, y_L and y_R, respectively.

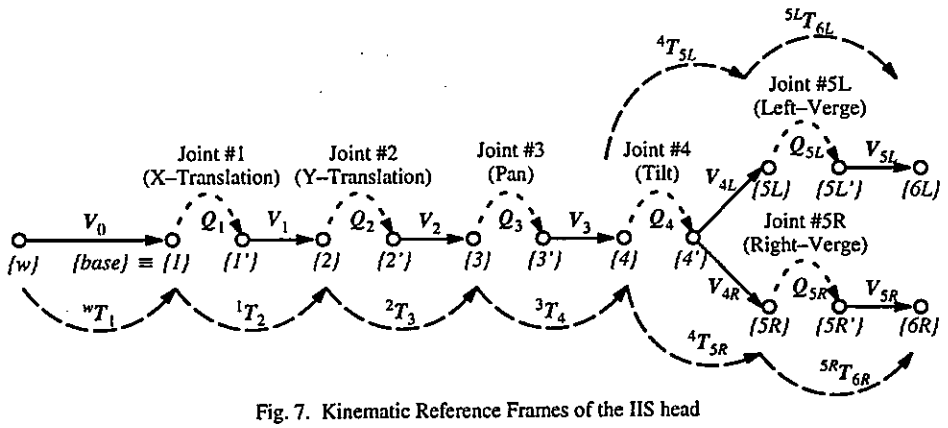

Fig. 7. Kinematic Reference Frames of the IIS head

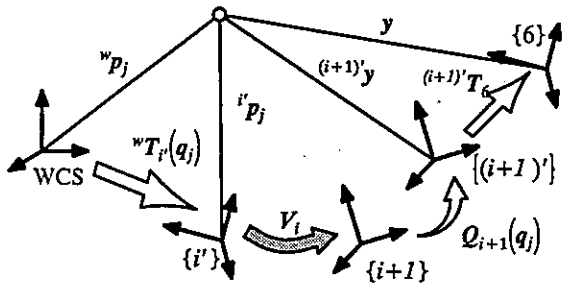

Fig. 8. Relation between 3D coordinates of a calibration point at the *j*th configuration with respect to different reference frames

With our new method, each joint is calibrated from the base toward the end–effectors. Consider Fig. 8. Without loss of generality, we assume that the kinematic parameters of the joints from the base to the *i*th joint have been known when calibrating the *i*th link matrix, i.e., V_i. Only those joints with known kinematic parameters plus the *(i+1)*st joint are permitted to be moved. For example, when calibrating V_{4L}, joints 1, 2, 3, 4 and 5*L* are allowed to be moved. From equations (8), (9) and (10), we have

$$^{i'}\tilde{p}_{jL} = {}^{i'}T_{i+1}\ {}^{i+1}T_{(i+1)'}\ {}^{(i+1)'}\tilde{y}_L, \tag{11}$$

and/or

$$^{i'}\tilde{p}_{jR} = {}^{i'}T_{i+1}\ {}^{i+1}T_{(i+1)'}\ {}^{(i+1)'}\tilde{y}_R, \tag{12}$$

where

$$^{i'}T_{i+1} = V_i,$$

$$^{i+1}T_{(i+1)'} = Q_{i+1},$$

$$^{i'}\tilde{p}_{jL} \equiv {}^{w}T_{i'}\left(q_j\right)^{-1}\ {}^{w}\tilde{p}_{jL}, \tag{13}$$

$$^{i'}\tilde{p}_{jR} \equiv {}^{w}T_{i'}\left(q_j\right)^{-1}\ {}^{w}\tilde{p}_{jR}, \tag{14}$$

$$^{(i+1)'}\tilde{y}_L \equiv {}^{(i+1)'}T_{6L}\left(q_j\right)\ \tilde{y}_L,$$

and

$$^{(i+1)'}\tilde{y}_R \equiv {}^{(i+1)'}T_{6R}\left(q_j\right)\ \tilde{y}_R.$$

Note that if the *i*th link is a common links, i.e., a link between the world reference frame and joint 4, then both of the above two equations are valid; otherwise, only one of the equations (11) and (12) is valid.

From equations (11) and (12), we have

$$V_i^{-1}\ {}^{i'}\tilde{p}_{jL} = Q_{i+1}\ {}^{(i+1)'}\tilde{y}_L\,, \tag{15}$$

and/or

$$V_i^{-1}\ {}^{i'}\tilde{p}_{jR} = Q_{i+1}\ {}^{(i+1)'}\tilde{y}_R\,. \tag{16}$$

If link i is shared by both the left and right kinematic chains, then we take the average of equations (15) and (16), which yields

$$V_i^{-1}\ {}^{i'}\tilde{p}_{jLR} = Q_{i+1}\ {}^{(i+1)'}\tilde{y}_{LR}\,, \tag{17}$$

where

$${}^{i'}\tilde{p}_{jLR} = \left({}^{i'}\tilde{p}_{jL} + {}^{i'}\tilde{p}_{jR}\right)/2$$

and

$${}^{(i+1)'}\tilde{y}_{LR} = \left({}^{(i+1)'}\tilde{y}_L + {}^{(i+1)'}\tilde{y}_R\right)/2\,.$$

Let

$$\begin{cases} {}^{i'}\tilde{p}_j = {}^{i'}\tilde{p}_{jLR},\quad {}^{(i+1)'}\tilde{y} = {}^{(i+1)'}\tilde{y}_{LR}, & \text{if link } i \text{ is shared by the left and right kinematic chains} \\ {}^{i'}\tilde{p}_j = {}^{i'}\tilde{p}_{jL},\quad {}^{(i+1)'}\tilde{y} = {}^{(i+1)'}\tilde{y}_L, & \text{if link } i \text{ belongs to the left branch only} \\ {}^{i'}\tilde{p}_j = {}^{i'}\tilde{p}_{jR},\quad {}^{(i+1)'}\tilde{y} = {}^{(i+1)'}\tilde{y}_R, & \text{if link } i \text{ belongs to the right branch only} \end{cases}$$

Then, equations (15), (16) and (17) can be unified as the following

$$V_i^{-1}\ {}^{i'}\tilde{p}_j = Q_{i+1}\ {}^{(i+1)'}\tilde{y}\,. \tag{18}$$

The unknown parameters to be estimated are the link matrix, V_i, and the vector, ${}^{(i+1)'}y$. For the case of IIS head, our goal is to estimate the kinematic parameters: V_0, V_1, V_2, V_3, V_{4L} and V_{4R}, using equation (18). Estimation of V_{5L} and V_{5R} will be handled differently as described in Section III.3.

III. NEW METHOD FOR THE KINEMATIC PARAMETER IDENTIFICATION

Using equation (18), our new method will calibrate the ith link matrix by moving the $(i+1)$st joint (joints 1, 2, ..., i can be moved too). For convenience, let R_{V_i}, R_i and $Rot_z(\theta)$ be the 3×3 rotation matrix of V_i, R_i and $Rot_z(\theta)$, respectively, and let t_{V_i} be the 3×1 translation vector of matrix V_i, and $l_i = \left[l_{i,x}\ \ l_{i,y}\ \ l_{i,z}\right]^T$. Using equation (6), we have

$$R_{V_i} = R_i\ Rot_z\left(\beta_i\right), \tag{19}$$

$$t_{V_i} = R_{V_i}\, l_i\,, \tag{20}$$

and

$$V_i = R_{V_i}\, \mathit{Trans}(l_i), \tag{21}$$

which will be used in the following derivation.

III.1 Kinematic Parameter Identification for a Prismatic Joint

The redundant parameters and the unknowns for a prismatic joint are listed below for clarity of our derivation:

i). Four given redundant parameters (typically set to zero if not pre–specified for specific reason): β_i and l_i.

ii). The unknowns: R_i, ${}^{(i+1)'}\tilde{y}$ and the sign parameter, s_i.

Consider equation (18). For a prismatic joint, we have

$$V_i^{-1}\, {}^{i'}\tilde{p}_j = \mathit{Trans}\left(\begin{bmatrix} 0 & 0 & q_{(i+1)j} \end{bmatrix}^T\right)\, {}^{(i+1)'}\tilde{y}\,, \tag{22}$$

or, using (21),

$$R_{V_i}^T\, {}^{i'}p_j - l_i = {}^{(i+1)'}y + \begin{bmatrix} 0 & 0 & q_{(i+1)j} \end{bmatrix}^T. \tag{23}$$

Notice that the value of ${}^{i'}p_j$ depend on the movements of joints 1, ..., i and can be computed using equations (13) and/or (14).

Multiplying R_{V_i} to both sides of equation (23) and noticing that

$$R_{V_i}\begin{bmatrix} 0 \\ 0 \\ q_{(i+1)j} \end{bmatrix} = R_i\, \mathit{Rot}_z(\beta_i)\begin{bmatrix} 0 \\ 0 \\ q_{(i+1)j} \end{bmatrix} = q_{(i+1)j}\, b_i\,, \tag{24}$$

where $b_i = \begin{bmatrix} b_{i,x} & b_{i,y} & b_{i,z} \end{bmatrix}^T$ is the third column vector of the rotation matrix R_i, we have

$${}^{i'}p_j = a + q_{(i+1)j}\, b_i\,, \tag{25}$$

where $\boldsymbol{a} \equiv R_{V_i}\left(\boldsymbol{l}_i + {}^{(i+1)'}\boldsymbol{y}\right)$ is independent of j. It is obvious that $\boldsymbol{l}_i$ is redundant, since $\boldsymbol{l}_i$ and ${}^{(i+1)'}\boldsymbol{y}$ can not be independently estimated. Therefore, $\boldsymbol{l}_i$ can be set to zero if it is not pre–specified for specific reason.

Suppose there are M measurements from M robot configurations, i.e., ${}^{i'}\boldsymbol{p}_j$ and $q_{(i+1)j}$, $j = 1, 2, \ldots, M$. The kinematic parameter $\boldsymbol{b}_i$ can be estimated by minimizing the following error

$$\varepsilon \equiv \sum_{j=1}^{M} \left\| {}^{i'}\boldsymbol{p}_j - \boldsymbol{a} - q_{(i+1)j}\, \boldsymbol{b}_i \right\|^2, \tag{26}$$

subject to $\boldsymbol{b}_i^T \boldsymbol{b}_i = 1$.

The solution is (refer to Appendix A for the proof)

$$\boldsymbol{b}_i = \frac{\sum_{j=1}^{M} \left(\Delta \boldsymbol{p}_j \, \Delta q_{(i+1)j} \right)}{\left\| \sum_{j=1}^{M} \left(\Delta \boldsymbol{p}_j \, \Delta q_{(i+1)j} \right) \right\|}, \tag{27}$$

where

$$\Delta \boldsymbol{p}_j \equiv {}^{i'}\boldsymbol{p}_j - {}^{i'}\overline{\boldsymbol{p}}, \tag{28}$$

$$\Delta q_{(i+1)j} \equiv q_{(i+1)j} - \overline{q}_{(i+1)}, \tag{29}$$

$$ {}^{i'}\overline{\boldsymbol{p}} = \frac{1}{M} \sum_{j=1}^{M} {}^{i'}\boldsymbol{p}_j, \tag{30}$$

and

$$\overline{q}_{(i+1)} = \frac{1}{M} \sum_{j=1}^{M} q_{(i+1)j}. \tag{31}$$

Notice that if the third component of $\boldsymbol{b}_i$ is negative, in order to be consistent with the CPC convention, we should change the sign of $\boldsymbol{b}_i$ and let $s_i = -1$; otherwise, let $s_i = +1$. Once the unit vector $\boldsymbol{b}_i$ is obtained, the rotation matrix, $\boldsymbol{R}_i$, can be computed with equation (7). Finally, ${}^{(i+1)'}\boldsymbol{y}$ can be solved by using equation (23). However, we do not have to calculate ${}^{(i+1)'}\boldsymbol{y}$ if it is not of interest.

III.2 Kinematic Parameter Identification for a Revolute Joint

The redundant parameters and the unknowns for a revolute joint are listed below for clarity of the derivation:

(1). Two given redundant parameters (typically set to zero if not pre–specified for specific reason): β_i and the z–component of l_i.

(2). The unknowns: R_i, the sign parameter, s_i, ${}^{(i+1)'}y$, and the first two components of l_i.

For a revolute joint, the calibration equation can be derived from (18) that

$$R_{V_i}^T \, {}^{i'}p_j - l_i = Rot_z\left(q_{(i+1)j}\right) \, {}^{(i+1)'}y \,, \tag{32}$$

by using equation (21). For convenience, we decompose ${}^{(i+1)'}y$ into two portions as follows. Let ${}^{(i+1)'}y = \left[\, y_1 \;\; y_2 \;\; y_3 \, \right]^T$, and

$$ {}^{(i+1)'}y = y_a + y_b \,, \tag{33}$$

where $y_a = \left[\, y_1 \;\; y_2 \;\; 0 \, \right]^T$ and $y_b = \left[\, 0 \;\; 0 \;\; y_3 \, \right]^T$. Note that for any y_1 and y_2, $y_1^2 + y_2^2 \neq 0$, there exist a scalar ϱ and a rotation angle ω such that

$$y_a = \varrho \; Rot_z(\omega) \; e_1, \tag{34}$$

where $e_1 = \left[\, 1 \;\; 0 \;\; 0 \, \right]^T$, $\varrho = \sqrt{y_1^2 + y_2^2}$, ω is the angle between the vector y_a and the x–axis. Substituting equations (33) and (34) into equation (32), we have (by noting that $y_b = Rot_z(\cdot) \; y_b$)

$$R \; u_j + t = \varrho \; v_j \,, \tag{35}$$

where $R \equiv Rot_z\left(-\beta_i - \omega \right) R_i^T$, $u_j \equiv {}^{i'}p_j$, $t \equiv - \, Rot_z(-\omega) \; l_i - y_b$ and

$v_j \equiv Rot_z\left(q_{(i+1)j}\right) \; e_1$.

R, t and ϱ can be solved by minimizing the following error using the least square method.

$$e \equiv \sum_{j=1}^{M} \left\| R \; u_j + t - \varrho \; v_j \right\|^2 \tag{36}$$

The procedures for calculating a closed–form solution to equation (36) are listed in the following (refer to Appendix B for more details).

Step 1: Compute $\bar{u} \equiv \frac{1}{N}\sum_{j=1}^{M} u_j$, $\bar{v} \equiv \frac{1}{N}\sum_{j=1}^{M} v_j$, $\underline{u}_j = u_j - \bar{u}$ and $\underline{v}_j = v_j - \bar{v}$.

Step 2: Let $\boldsymbol{A} \equiv \left[\underline{v}_1 \ \cdots \ \underline{v}_j \ \cdots \ \underline{v}_M\right]$ and $\boldsymbol{B} \equiv \left[\underline{u}_1 \ \cdots \ \underline{u}_j \ \cdots \ \underline{u}_M\right]$.

Step 3: Compute the matrix $\boldsymbol{C} \equiv \boldsymbol{B}\,\boldsymbol{A}^T$.

Step 4: Compute the singular value decomposition

$$\boldsymbol{C} = \boldsymbol{U}\begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{bmatrix}\boldsymbol{V}^T, \text{ where } s_1 \geq s_2 \geq s_3 \geq 0.$$

Step 5: If $\det\left(\boldsymbol{V}\,\boldsymbol{U}^T\right) = +1$, then compute $\hat{R} = \boldsymbol{V}\,\boldsymbol{U}^T$, otherwise, let $\hat{R} = \boldsymbol{V}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}\boldsymbol{U}^T$.

Step 6: Compute $\varrho = \left[\sum_{j=1}^{M} \underline{u}_j^T\, R\, \underline{v}_j\right]\left[\sum_{j=1}^{M} \underline{v}_j^T \underline{v}_j\right]^{-1}$.

Step 7: Compute $\boldsymbol{t} = \varrho\,\bar{v} - R\,\bar{u}$.

We now show how to compute the kinematic parameters, R_i and $\boldsymbol{l}_i$, from R and $\boldsymbol{t}$. Remember that $R = Rot_z\left(-\beta_i - \omega\right) R_i^T$, as defined following equation (35). It is obvious that the third rows of R and R_i^T, i.e., $\boldsymbol{b}_i^T$, should be the same. Therefore, $\boldsymbol{b}_i^T$ can be obtained from the third row of the matrix R. Notice that if the third component of $\boldsymbol{b}_i$ is negative, in order to be consistent with the CPC convention, we should change the sign of $\boldsymbol{b}_i$ and let $s_i = -1$; otherwise, let $s_i = +1$. From equations (6)

and (7), we can compute R_{V_i} from b_i and β_i. Then we can compute $Rot_z(\omega)$ using the following relation:

$$Rot_z(\omega) = R_{V_i}^T R^T . \tag{37}$$

Remember that $t = -Rot_z(-\omega)\ l_i - y_b$ and $y_b = [\ 0\ \ 0\ \ y_3\ ]^T$, hence

$$l_i + y_b = -Rot_z(\omega)\ t , \tag{38}$$

where the z–component of l_i is a given redundant parameter, and y_b should be determined to make the above three equations of three unknowns consistent.

III.3 Estimation of the End–Effector Frame

Unlike the joint axis of a robot, the position and orientation of the end–effector has no physical meaning. The end–effector frame is simply a definition by the manufacturer. Therefore, when calibrating a binocular head, the link matrices of end–effectors, i.e, V_{5L} and V_{5R}, can be set to identity matrices. However, when calibrating a robot manipulator, it is usually necessary to calibrate the end–effector frame. Mechanically, it is possible to produce a calibration tool having at least three non–collinear measurable points and the 3D coordinates of each calibration point with respect to the end–effector frame is known. Suppose there are K non–collinear calibration points on the calibration tool and the 3D coordinates of those calibration points with respect to the end–effector frame are known to be y_k, $k = 1, 2, ..., K$. At the robot configuration j, let the measured 3D coordinates of the calibration points with respect to the WCS be p_k^j, $k = 1, 2, ..., K$. Notice that before calibrating the end–effector frame, the kinematic parameters from WCS to the last joint should have been calibrated by using our kinematic calibration method. Therefore, the transformation matrix ${}^wT_{N'}(q_j)$ from WCS to the last link frame can be computed from the kinematic model. Here, $\{N'\}$ is the last joint frame (i.e, $\{5L'\}$ or $\{5R'\}$ in our case as shown in Fig. 7). Transforming y_k to the world frame, we have

$$p_k^j = {}^wT_{N'}(q_j)\, V_N\, y_k\,, \tag{39}$$

where V_N is the last link matrix (i.e, V_{5L} or V_{5R} in our case). By moving robot to M different configurations and taking 3D measurements of the calibration points, we have totally $M \times K$ pairs of 3D coordinates, $\left(y_k,\ {}^wT_{N'}^{-1}(q_j)\, p_k^j\right), k = 1, 2, ..., K$ and $j = 1, 2, ..., M$. Then the unknown link matrix V_N can be solved by using the Arun method[3].

IV. THEORETICAL ERROR ANALYSIS

This section will present the approximate error variances of the estimates for the kinematic parameters. The error variance is a function of the amount of measurement noise, the number of calibration data and the calibration range of the corresponding joint value. In addition to the above three factors, the parameter estimation error of a revolute joint is also affected by the distance between the calibration point and the joint axis. The derived parameter error variances are based on the following assumptions.

Assumption 1. The error of the joint value is negligible. In general, there are a scale factor and an offset parameter to be calibrated for transforming the reading of the motion transducer to the joint value, where in most cases the scale factor of the transducer can be obtained from the sensor manufacturer. Hence, we may assume that only offset parameter is necessary to be calibrated. Although the motion transducer is usually designed to be vary accurate, the joint value may not be accurate enough due to the estimation error of the offset parameter. However, one advantage of using the CPC kinematic model is the elimination of the offset parameter for converting the transducer reading to the joint value. Therefore, we assume that the error of the joint values of the robot is negligible in the error analysis.

Assumption 2. The error of ${}^{i'}p_j$ due to 3D measurement noise is the 3x1 random vector δp_j, and the measurement errors δp_j , $j = 1, 2, ..., M$, are independent and identically distributed Gaussian with zero mean and covariance matrix $\sigma^2\, I_{3\times 3}$, where $I_{3\times 3}$ is the 3x3 identity matrix.

When calibrating the ith joint, let the range of joint motion during data collection be ΔQ (which will be referred to as the *calibration range*, hereafter). Within the calibration range ΔQ, we suppose totally M data points are sampled for the calibration and the data points are uniformly spread apart.

For deducing the estimation error variance, we need the following two lemmas.

Lemma 1. Let $X_1, X_2, ..., X_m$ be mutually stochastically independent random variables having, respectively, the normal distributions $n(\mu_1, \sigma_1^2)$, $n(\mu_2, \sigma_2^2)$, ... and $n(\mu_m, \sigma_m^2)$. The random variable $Y = \sum_{i=1}^{m} k_i X_i$, where $k_1, k_2, ..., k_m$ are real constants, is normally distributed with mean $\mu_Y = \sum_{i=1}^{m} k_i \mu_i$ and variance $\sigma_Y^2 = \sum_{i=1}^{m} k_i^2 \sigma_i^2$.

Lemma 2. Let δR be a 3×3 rotation matrix corresponding to the small X–Y–Z Eular angles, $\delta\phi_x$, $\delta\phi_y$ and $\delta\phi_z$, where we define an angle $\delta\phi$ to be small if $\cos(\delta\phi) \approx 1$ and $\sin(\delta\phi) \approx \delta\phi$. Then δR is approximately equal to $I + Skew\left(\left[\delta\phi_x, \quad \delta\phi_y, \quad \delta\phi_z\right]^T\right)$, where

$$Skew\left(\begin{bmatrix}\delta\phi_x\\ \delta\phi_y\\ \delta\phi_z\end{bmatrix}\right) \equiv \begin{bmatrix} 0 & -\delta\phi_z & \delta\phi_y \\ \delta\phi_z & 0 & -\delta\phi_x \\ -\delta\phi_y & \delta\phi_x & 0 \end{bmatrix}. \tag{40}$$

IV.1. Error Analysis for a Prismatic Joint.

Define the estimation error of the kinematic parameters of a prismatic joint as follows.

$$\delta b_i \equiv \hat{b}_i - b_i, \tag{41}$$

where $\hat{b}_i$ and b_i are the estimated and true kinematic parameters for a prismatic joint i, respectively.

From equation (27), we have

$$\hat{b}_i = \frac{\sum_{j=1}^{M}\left(\Delta p_j\, \Delta q_{(i+1)j} + \delta_{\Delta} p_j\, \Delta q_{(i+1)j}\right)}{\left\|\sum_{j=1}^{M}\left(\Delta p_j\, \Delta q_{(i+1)j} + \delta_{\Delta} p_j\, \Delta q_{(i+1)j}\right)\right\|}, \tag{42}$$

where Δp_j is the true value, and the measurement noise $\delta_{\Delta} p_j = \delta p_j - \delta \overline{p}$ (refer to equation (28)).

Let
$$P = \sum_{j=1}^{M}\left(\Delta p_j\, \Delta q_{(i+1)j}\right) \tag{43}$$

and
$$\delta P = \sum_{j=1}^{M}\left(\delta_{\Delta} p_j\, \Delta q_{(i+1)j}\right). \tag{44}$$

Substituting equations (43) and (44) into (42), we have

$$\hat{b}_i = \frac{P + \delta P}{\|P + \delta P\|}. \tag{45}$$

It can be shown that if the amount of effective noise, δP, is relatively small comparing to the true value, P, the denominator can be represented as follows.

$$\frac{1}{\|P + \delta P\|} = \frac{1}{\|P\|}\left[1 - \frac{\delta P^T P}{\|P\|^2}\right] + O(2) \tag{46}$$

Substituting equation (46) into (45), we have

$$\hat{b}_i = b_i + \delta\beta - \left(b_i^T\, \delta\beta\right) b_i + O(2), \tag{47}$$

where $\delta\beta = \frac{\delta P}{\|P\|}$ is the effective noise vector, and $b_i = \frac{P}{\|P\|}$ is the true kinematic parameters since P is noise free.

From equations (41) and (47), we have

$$\delta b_i = \delta\beta - \left(b_i^T\, \delta\beta\right) b_i + O(2). \tag{48}$$

Notice that in equation (48), δb_i, is approximately equal to the component of $\delta\beta$ perpendicular to b_i as shown in Fig. 9. Therefore, the size of δb_i is proportional to and bounded by the size of $\delta\beta$. Mini-

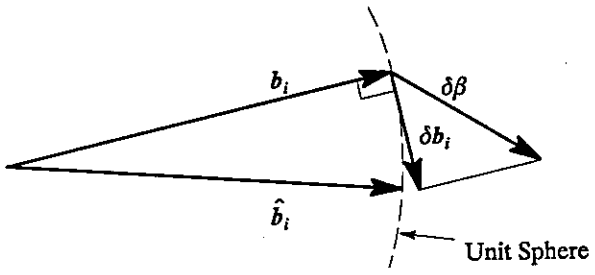

Fig. 9. Parameter Estimation Error of a Prismatic Joint

mizing the amount of $\delta\beta$ is equivalent to minimizing the amount of parameter estimation error. Hence, we would first derive the covariance matrix of the random vector $\delta\beta$, to find the factors that affect the calibration accuracy.

Suppose that in the calibration process, the joint values corresponding to the M measurements were uniformly taken from the region $\{Q_1,\ Q_1 + \Delta Q\}$, i.e., $q_{(i+1)j} = Q_1 + (j-1)\,Q$, where $Q \equiv \dfrac{\Delta Q}{M-1}$ is the constant movement of joint *(i+1)* between two consecutive measurements. Then the covariance matrix of $\delta\beta$ can be derived as follows (refer to Appendix C).

$$Cov[\delta\beta] = \sigma_{\delta\beta}^2\, I_{3\times 3}\,, \tag{49}$$

where
$$\sigma_{\delta\beta}^2 = \frac{\sigma_{\delta P}^2}{\|P\|^2} = T_\beta(M)\,\frac{\sigma^2}{\Delta Q^2} \tag{50}$$

and
$$T_\beta(M) = \frac{12(M-1)}{M\,(M+1)}. \tag{51}$$

The function $T_\beta(M)$ is plotted in Fig. 10. According to equation (50), it is clear that the estimation error can be made smaller by making ΔQ or M larger or by making the measurement noise smaller. Also from equation (50), for minimizing the estimation error, increasing size of the calibration range ΔQ is more efficient than increasing the number of measurements M.

IV.2. Error Analysis for a Revolute Joint.

Suppose that in the revolute joint calibration process, the joint values corresponding to the M measurements were uniformly distributed within the region $\{Q_1,\ Q_1 + \Delta Q\}$. Because the length

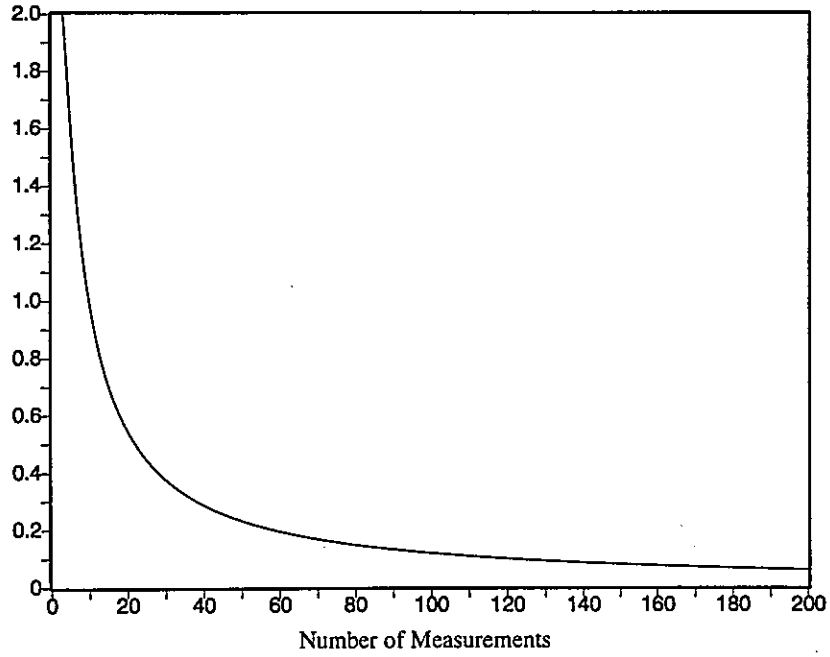

Fig. 10. Function $T_\beta(M)$

of a vector is preserved by any orthonormal transformation matrix. Therefore, minimizing equation (36) is equivalent to minimizing the following error function

$$e' \equiv \sum_{j=1}^{M} \left\| R_1 \, u_j + t_1 - \varrho \, v_{j1} \right\|^2, \tag{52}$$

where $$R_1 = Rot_z\left(- Q_1 - \frac{\Delta Q}{2} \right) R \, , \; t_1 = Rot_z\left(- Q_1 - \frac{\Delta Q}{2} \right) t \, ,$$

and $$v_{j1} = Rot_z\left(- Q_1 - \frac{\Delta Q}{2} \right) v_j .$$

Now the original revolute joint calibration problem is transformed into an equivalent problem where the joint values corresponding to the M measurements were uniformly distributed within the region $\{ - \Delta Q/2, \; \Delta Q/2 \}$. Therefore, without lost of generality, we may always assume that when calibrating a revolute joint, the joint angles are all kept within the region $\{ - \Delta Q/2, \; \Delta Q/2 \}$.

Recall that the kinematic parameters of a revolute joint is solved from R and t in Section III. Therefore, minimizing the kinematic parameter estimation error for a revolute joint is equivalent to minimizing the estimation error of R and t. We shall show the result of error analysis for the esti-

mated parameters, R, t and ϱ, which are obtained by minimizing equation (36). Let δR be a 3×3 error rotation matrix composed of the small X–Y–Z Eular angles, $\delta\phi_x$, $\delta\phi_y$ and $\delta\phi_z$. Let the estimated parameters be

$$\hat{R} = \delta R \, R^* \approx \left[I_{3\times3} + Skew(\delta\phi) \right] R^*, \tag{53}$$

$$\hat{t} = t^* + \delta t \,, \tag{54}$$

and

$$\hat{\varrho} = \varrho^* + \delta\varrho \,, \tag{55}$$

where R^*, t^* and ϱ^* are the true values, $\delta\phi = \left[\delta\phi_x \;\; \delta\phi_y \;\; \delta\phi_z\right]^T$, $\delta t = \left[\delta t_x \;\; \delta t_y \;\; \delta t_z\right]^T$ and $\delta\varrho$ are the estimation errors.

From assumptions 1 and 2, we have that v_j in equation (36) is noise free and more explicitly

$$v_j = \begin{bmatrix} \cos\left(q_{(i+1)j}\right) \\ \sin\left(q_{(i+1)j}\right) \\ 0 \end{bmatrix}, \tag{56}$$

and

$$\hat{u}_j = u_j + \delta u_j \,, \tag{57}$$

where

$$u_j = {}^{i}p_j \,,$$

and

$$\delta u_j = \delta p_j \,.$$

Rewrite equation (36) as follows

$$e \equiv \sum_{j=1}^{M} \left\| R \; u_j + \delta u'_j + t - \varrho \; v_j \right\|^2, \tag{58}$$

where $\delta u'_j \equiv R \; \delta u_j$. As stated in assumption 2, each two components of the random vector δu_j are independent and identically distributed as Gaussian, which yields (by noting that R is an orthonormal matrix)

$$Cov\left[\delta u'_j\right] = \sigma^2 \, I_{3\times3} \,, \tag{59}$$

To minimize the error defined in equation (58), it is required to have that (refer to Appendix B)

$$t = \varrho\, \bar{v} - R\, \bar{u}\,. \tag{60}$$

Substituting equation (60) into (58), we have

$$e = \sum_{j=1}^{M} \left\| R\, \underline{u}_j + \delta u'_j - \delta \bar{u}' - \varrho\, \underline{v}_j \right\|^2, \tag{61}$$

where $\bar{u} \equiv \frac{1}{N}\sum_{j=1}^{M} u_j$, $\bar{v} \equiv \frac{1}{N}\sum_{j=1}^{M} v_j$, $\underline{u}_j = u_j - \bar{u}$ and $\underline{v}_j = v_j - \bar{v}$.

For mathematical simplicity, $\delta\bar{u}'$ is assumed to be negligible. This assumption is practically true providing that the number of measurements is not too small. Therefore, the above equation can be simplified as follows.

$$e = \sum_{j=1}^{M} \left\| R\, \underline{u}_j + \delta u'_j - \varrho\, \underline{v}_j \right\|^2 \tag{62}$$

Substituting the optimal solutions of R and ϱ in equations (53) and (55) into (62) and omitting the high order error terms, we have

$$e^* = \sum_{j=1}^{M} \left\| Skew(\delta\phi)\, R^* \underline{u}_j + \delta u'_j - \delta\varrho\, \underline{v}_j \right\|^2. \tag{63}$$

Notice that $Skew(\delta\phi)\, R^* \underline{u}_j$ is actually the cross product of vectors $\delta\phi$ and $R^* \underline{u}_j$, i.e.,

$$Skew(\delta\phi)\, R^* \underline{u}_j = \delta\phi \times R^* \underline{u}_j = -\, Skew\left(R^* \underline{u}_j\right) \delta\phi\,, \tag{64}$$

Since $\varrho^* v_j = R^* u_j + t^*$, we have

$$R^* \underline{u}_j = \varrho^* \underline{v}_j. \tag{65}$$

Substituting equations (64) and (65) into equation (63), we have

$$e = \sum_{j=1}^{M} \left\| -\, Skew\left(\varrho^* \underline{v}_j\right) \delta\phi + \delta u'_j - \delta\varrho\, \underline{v}_j \right\|^2. \tag{66}$$

The estimation error $\delta\phi$ and $\delta\varrho$ can be found by minimizing equation (66), and the variance can also be derived as follows (refer to Appendix D).

$$\sigma^2_{\delta\phi_x} = \frac{\sigma^2}{\varrho^2 M}\ \Phi_x(\Delta Q), \tag{67}$$

$$\sigma^2_{\delta\phi_y} = \frac{\sigma^2}{\varrho^2 M}\ \Phi_y(\Delta Q), \tag{68}$$

$$\sigma^2_{\delta\phi_z} = \frac{\sigma^2}{\varrho^2 M}\ \Phi_z(\Delta Q), \tag{69}$$

$$\sigma^2_{\delta\varrho} = \frac{\sigma^2}{M}\ \Phi_z(\Delta Q), \tag{70}$$

$$\sigma^2_{\delta t_x} = \Phi_t^2(\Delta Q)\ \sigma^2_{\delta\varrho} + t_z^2\ \sigma^2_{\delta\phi_y} + t_y^2\ \sigma^2_{\delta\phi_z}, \tag{71}$$

$$\sigma^2_{\delta t_y} = \left(\varrho\ \Phi_t(\Delta Q) - t_x\right)^2 \sigma^2_{\delta\phi_z} + t_z^2\ \sigma^2_{\delta\phi_x}, \tag{72}$$

and

$$\sigma^2_{\delta t_z} = \left(\varrho\ \Phi_t(\Delta Q) - t_x\right)^2\ \sigma^2_{\delta\phi_y} + t_y^2\ \sigma^2_{\delta\phi_x}, \tag{73}$$

where t_x, t_y and t_z are the x, y and z components of the true translation vector $\boldsymbol{t}^*$, respectively,

$$\Phi_x(\Delta Q) \equiv \frac{2\Delta Q}{\Delta Q - \sin(\Delta Q)}, \tag{74}$$

$$\Phi_y(\Delta Q) \equiv \frac{2\Delta Q^2}{\Delta Q^2 - 4(1 - \cos(\Delta Q)) + \Delta Q \sin(\Delta Q)}, \tag{75}$$

$$\Phi_z(\Delta Q) \equiv \frac{\Delta Q^2}{\Delta Q^2 - 2(1 - \cos(\Delta Q))}, \tag{76}$$

and

$$\Phi_t(\Delta Q) \equiv \frac{\sin(\Delta Q/2)}{\Delta Q/2}. \tag{77}$$

Functions $\Phi_x(\cdot)$, $\Phi_y(\cdot)$, $\Phi_z(\cdot)$ and $\Phi_t(\cdot)$ in equations (74)–(77) are plotted in Fig. 11. Notice that when ΔQ is small, $\Phi_y(\cdot)$ is much larger than the other two, i.e., $\delta\phi_y$ will have larger variance comparing with the other two orientation estimation errors, especially when ΔQ is small. This is because when ΔQ is very small, the trajectory of the calibration point will be very close to a straight line (denoted by L). Therefore, small amount of measurement noise would cause large amount of error rotation angle corresponding to the rotation axis parallel to the line L (see Fig. 12).

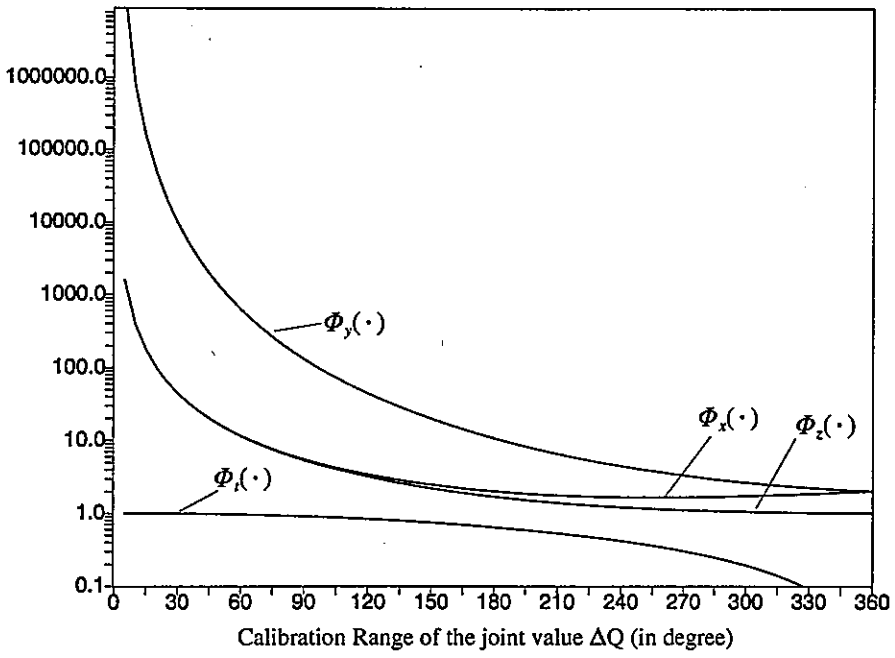

Fig. 11. Functions $\Phi_x(\Delta Q)$, $\Phi_y(\Delta Q)$, $\Phi_z(\Delta Q)$ and $\Phi_t(\Delta Q)$

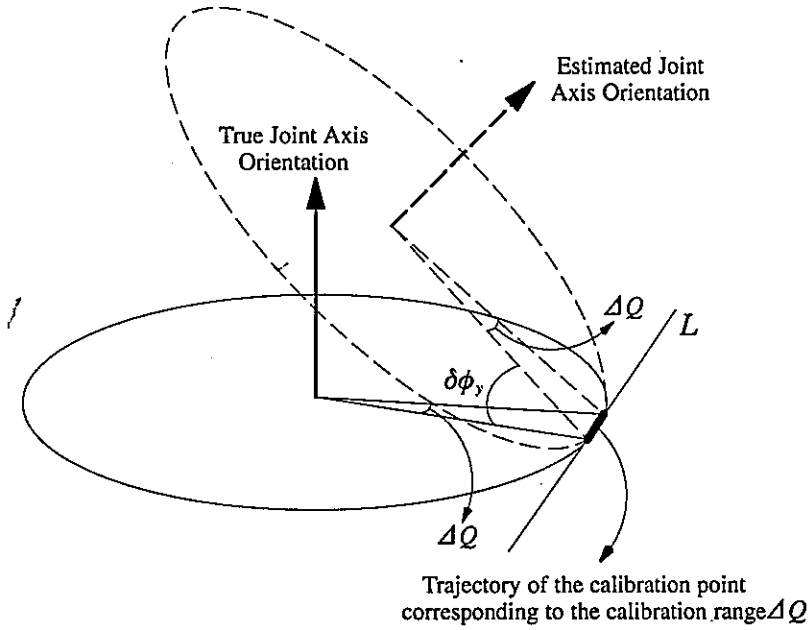

Fig. 12. Small calibration range would introduce large orientation error.

According to equations (67)–(69), to minimize the parameter estimation error for a revolute joint, we can make as large as possible the calibration range ΔQ, the number of calibration measure-

ments M, and the radius ϱ (i.e., the distance between the calibration point and the revolute joint axis). It is obvious that the more efficient way to reduce the calibration error is to make the calibration range ΔQ or the radius ϱ larger instead of increasing the number of calibration point. For instance, when the calibration range ΔQ is approximately 30° and $\Phi_y(\Delta Q)$ will be approximately 10,000 (see Fig. 11), then we may choose a radius ϱ which is much larger then 100 millimeters to reduce the error. On the other hand, if we want to reduce the error by choosing a large number of M to achieve the same effect, then we would have to take much more than 10,000 measurements.

Orientation error is inversely proportional to the radius ϱ, but according to equations (71)–(73), the translational error is not. Due to the quadratic term of the radius ϱ in equations (72) and (73), there is a range of the radius ϱ, determined by ΔQ and t_x, in which the estimation translation error is minimized. The error analysis results presented in this section can be used as a guide line for determining a proper size of the radius ϱ, the number of measurements M, the calibration range ΔQ, and the accuracy of the coordinate measurement device.

V. EXPERIMENTS

To evaluate the accuracy of robot calibration, we use the positioning error defined below as the error measure. The positioning error is defined as the Euclidian distance between the measured position of the end–effector (or more precisely, of the single calibration point) and its predicted position using the identified CPC kinematic model. The specifications of the architecture of IIS head are shown in Table 1. The 3D measurement system we use is a stereo vision system with baseline of 0.5 meter, which provides the accuracy of 0.4mm in z–direction and 0.2mm in x– or y–directions if the object distance is approximately 1.5 meters and the deviation of the 2D observation error is 0.1 pixel.

Table 1. Specifications of IIS head

Joint #	Type	Range	Resolution	Calibration Range
1	Prismatic	0 – 1000 mm	0.005mm	250 – 350mm
2	Prismatic	0 – 500 mm	0.005mm	200 – 300mm
3	Revolute	0 – 180°	0.004°	61 – 121°
4	Revolute	0 – 60°	0.005°	10 – 50°
5^L	Revolute	0 – 90°	0.0036°	10 – 70°
5^R	Revolute	0 – 90°	0.0036°	26 – 86°

The calibration objects are shown in Fig. 2, where each calibration plate has nine white disks. For each measurements, the estimated 3D coordinates of the nine disks are averaged to obtain an accurate estimate of the end–effector's 3D position, which has approximately 0.05 millimeter standard deviation in each direction.

In general, if the expectation value of an error is nearly zero, then its variance would be close to its RMSE. For convenience, we define the predicted radius, translational and rotational RMSE for a revolute joint to be respectively $\sigma_{\delta\varrho}$, $\sqrt{\sigma^2_{\delta t_x} + \sigma^2_{\delta t_y} + \sigma^2_{\delta t_z}}$ and $\sqrt{\sigma^2_{\delta\phi_x} + \sigma^2_{\delta\phi_y} + \sigma^2_{\delta\phi_z}}$, and the predicted RMSE for a prismatic joint to be $\sigma_{\delta\beta}$. In the following, we shall first present some results obtained from the computer simulations for verifying the predicted RMSE derived in the previous section. Then we shall show the results of both the computer simulations and the real experiments on calibrating the kinematic parameters of the IIS head.

In the first experiment, the RMSE of the parameter estimates for a prismatic joint are computed from 100 random trials. There are three factors govern the estimation error for a prismatic joint, i.e., the number of measurements, M, the standard deviation of the 3D measurement noise, σ, and the calibration range, ΔQ. As stated in section IV, the RMSE of the orientation parameter estimates are bounded by $\sigma_{\delta\beta}$. The results obtained from computer simulations are shown in Figs.

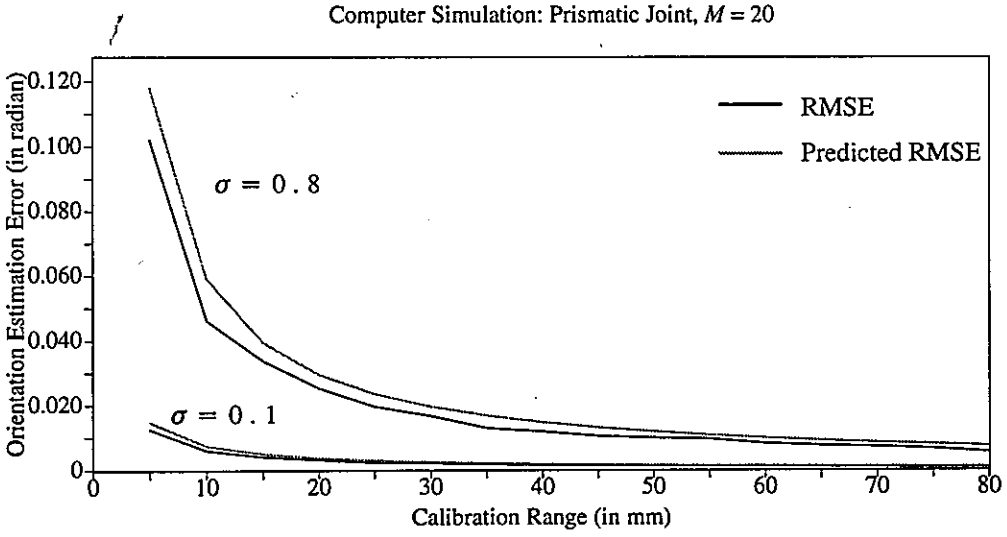

Fig. 13. The orientation estimation error versus the calibration range.

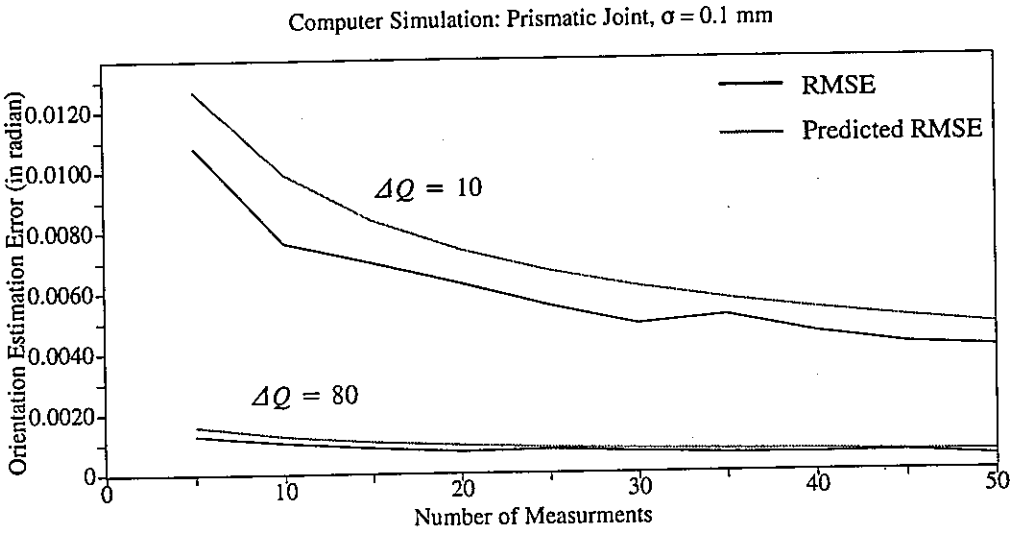

Fig. 14. The orientation estimation error versus the number of measurements.

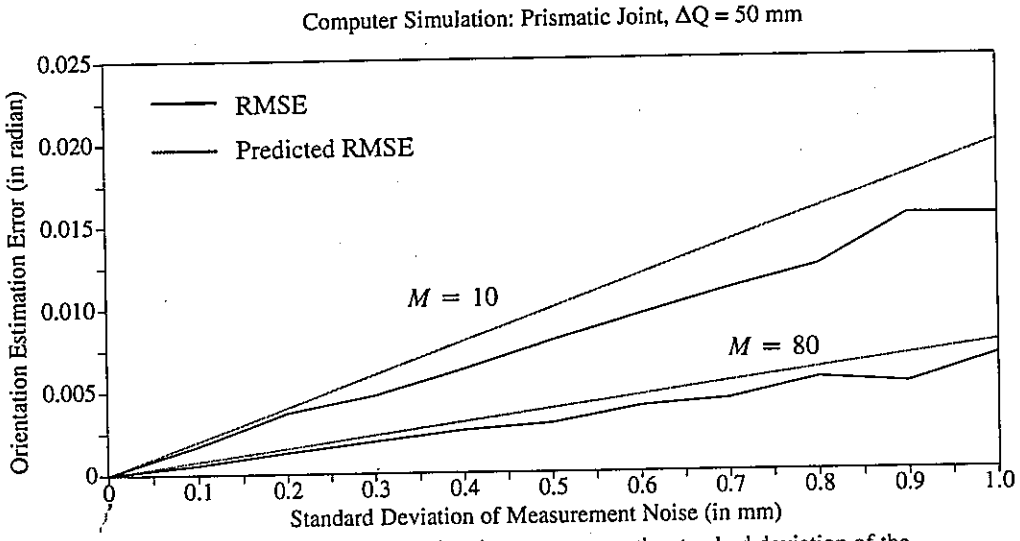

Fig. 15. The orientation estimation error versus the standard deviation of the measurement noise.

13–15, which show that the derived variance dose reflect the influences of the above three factors on the estimation error.

The second experiment assesses the validity of the derived (or predicted) RMSE for a revolute joint. Each data point shown in Figs. 16–27. is computed from 100 random trials. The four factors which govern the estimation error for a revolute joint are the number of measurements, M, the standard deviation of the 3D measurement noise, σ, the calibration range, ΔQ, and the radius, ϱ (i.e., the distance between the calibration point and the joint axis). This experiments shows that the

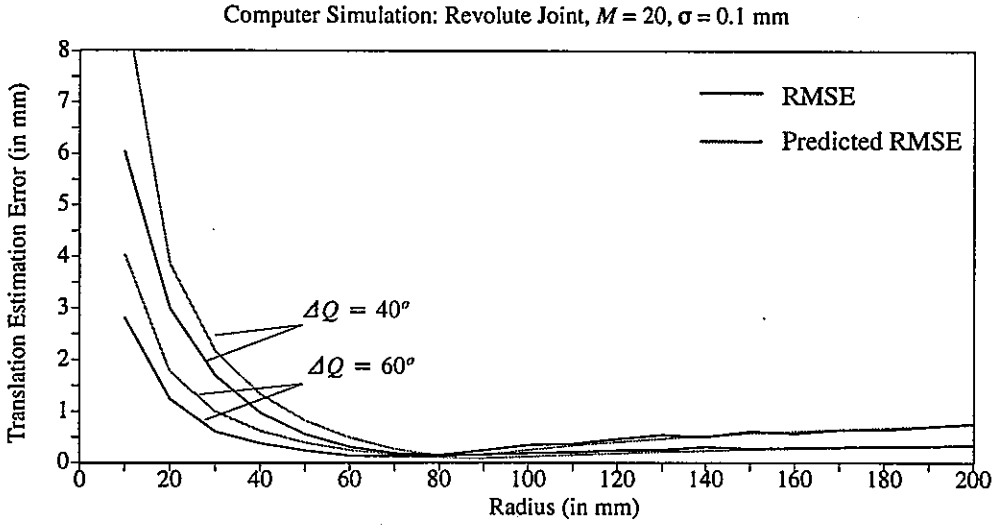

Fig. 16. The translation estimation error versus the radius length

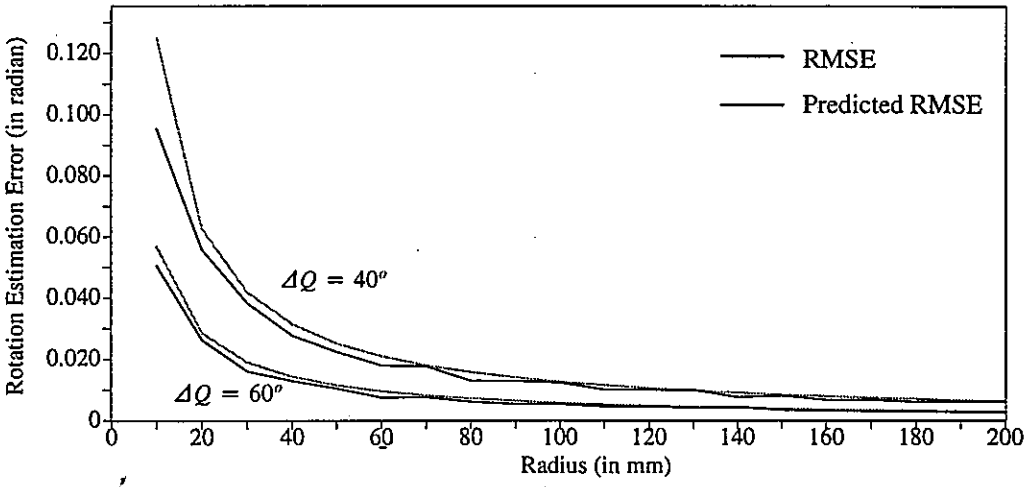

Fig. 17. The rotation estimation error versus the radius length.

derived RMSE is very close to the simulation results. However, the derived estimation RMSE will not be accurate when the rotational RMSE is larger than 0.2 radian (or approximately 10°). This is because the error variance is derived from a locally linearized model. Therefore, when error is too large, the predicted RMSE would be inaccurate.

From the first two experiments, we find that the error analysis presented in Section IV are reliable. Therefore, we can use these equations for determining a good configuration for kinematic calibrations to minimize the parameter estimation error.

In the following, we will present the results of applying the proposed method to our IIS head.

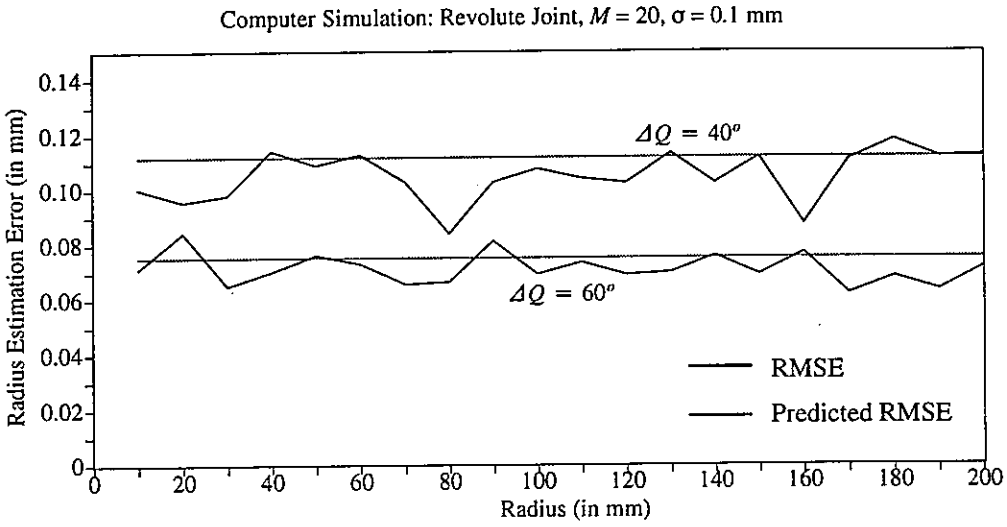

Fig. 18. The radius estimation error versus the radius length.

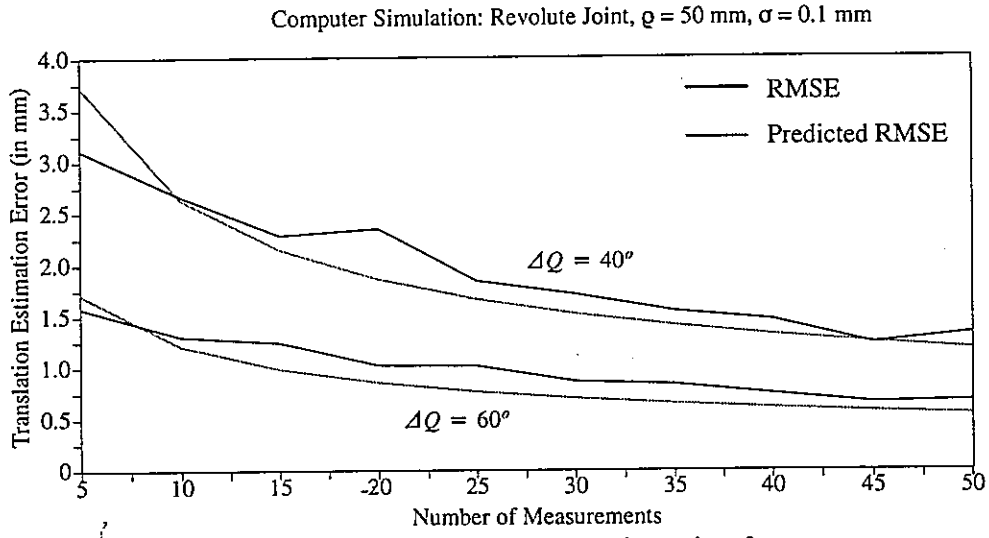

Fig. 19. The translation estimation error versus the number of measurements.

For comparison, we have also implemented a kinematic calibration method using pose measurements[15]. To obtain the pose measurements, a fixed coordinate system called the left (right) calibration plate coordinate system (LCPCS, RCPCS) is defined, on the left (right) calibration plate. The coordinates of the centroid of each disk on the calibration plate with respect to LCPCS or RCPCS are known *a priori*. The pose measurements are obtained by applying Arun's Algorithm[3] to the 3D coordinates of the nine disks measured in the LCPCS (RCPCS) and the WCS, respectively. For convenience, the method using pose measurements[15] will be referred to as the *Pose–Method*, and our new method will be referred to as the *Point–Method*.

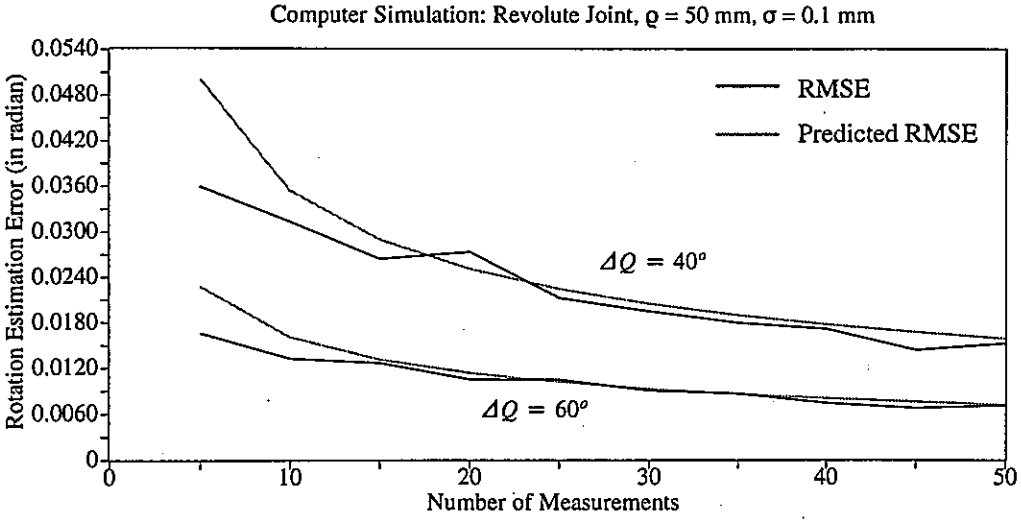

Fig. 20. The rotation estimation error versus the number of measurements.

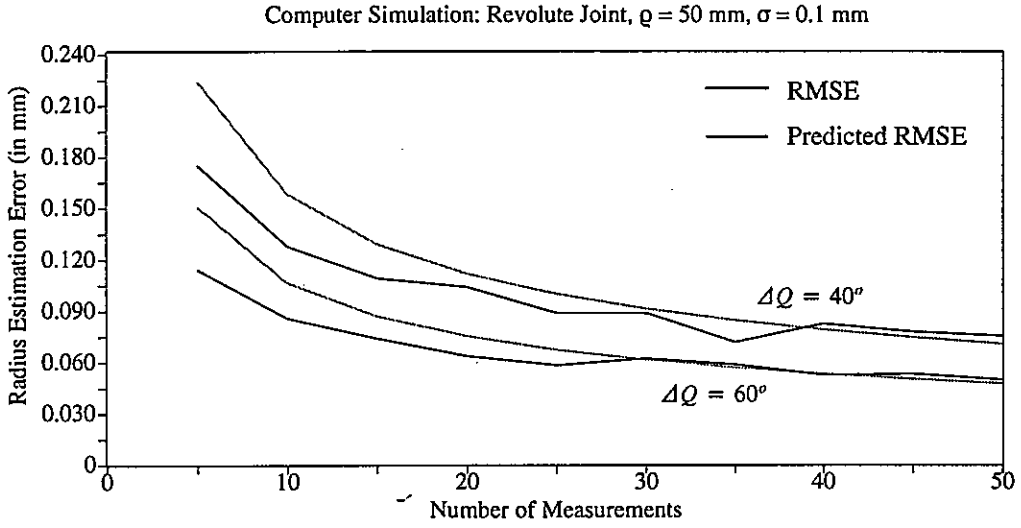

Fig. 21. The radius estimation error versus the number of measurements.

In the third experiment (real experiment), twenty measurements are taken for each joint. Also, for testing the calibration accuracy we have taken extra nineteen measurements with arbitrary configurations . We then choose M from the twenty measurements to calibrate the robot and use all of the nineteen testing measurements to test the estimated parameters. The results are shown in Fig. 28 for $M = 4, 6, ..., 20$. One purpose of this experiment is to determine a proper M such that the position error is less than 1mm. Fig. 28 shows that the *Point–Method* can achieve higher accuracy than the *Pose–Method*. Notice that Fig. 28 does not show the potential that the position error decreases when M increases. We believe it is because the non–geometric error dominates and the random noise in-

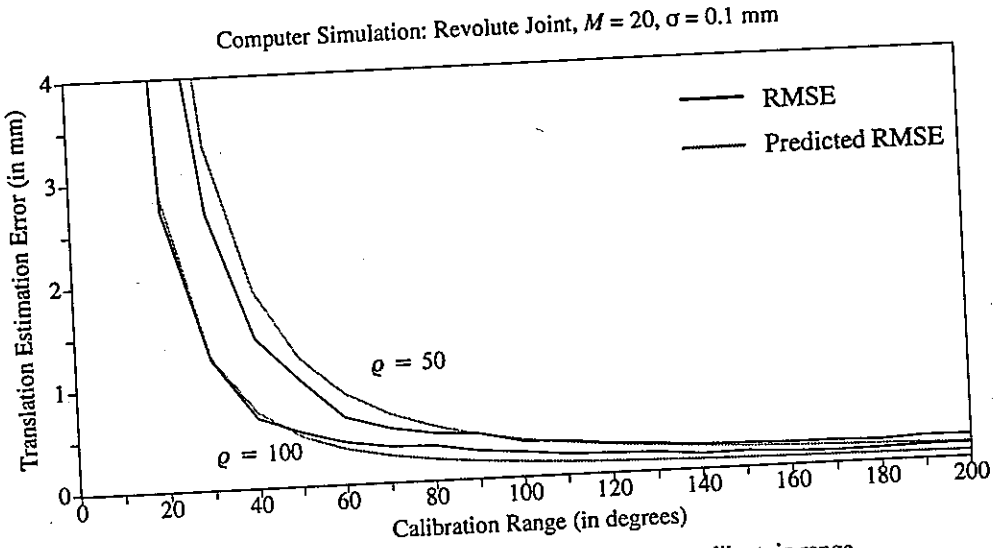

Fig. 22. The translation estimation error versus the calibratoin range.

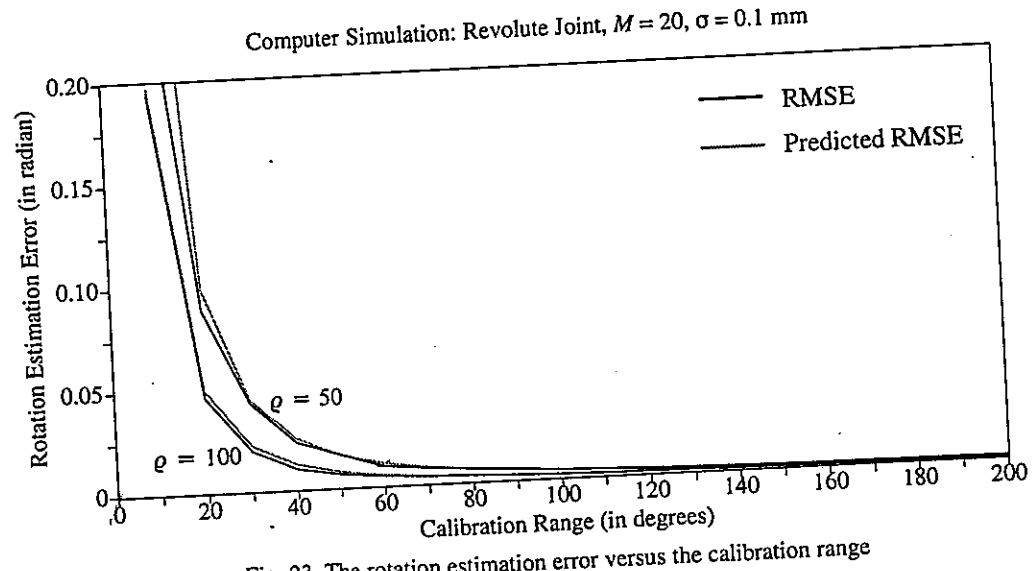

Fig. 23. The rotation estimation error versus the calibration range

duced error is relatively negligible. However, whatever M is, the error is less than 1 millimeter. We use $M = 20$ in the following experiments.

In the forth experiment, we run simulations using the kinematic parameters obtained from the previous real experiment. We assume that the 3D stereo measurement noise is due to the 2D image observation noise of 0.1 pixel standard deviation. Each data points in Fig. 29 is the average of fifty random trials. Notice that in the real experiments, the test data also contain measurement errors. Also, the joints of the IIS head are not ideal at all. Therefore, the real experimental results (in Fig. 28) have larger position error than the simulation ones (Fig. 29).

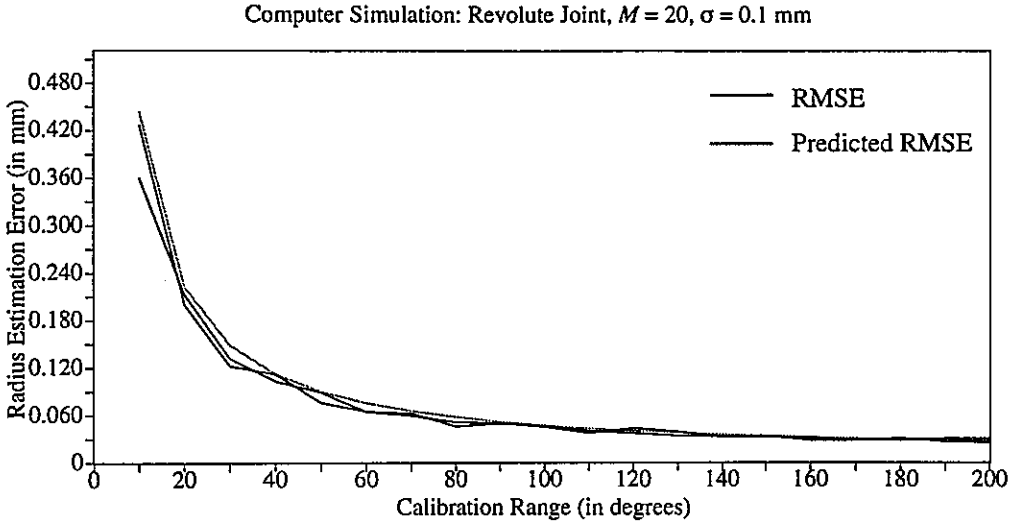

Fig. 24. The radius estimation error versus the calibration range.

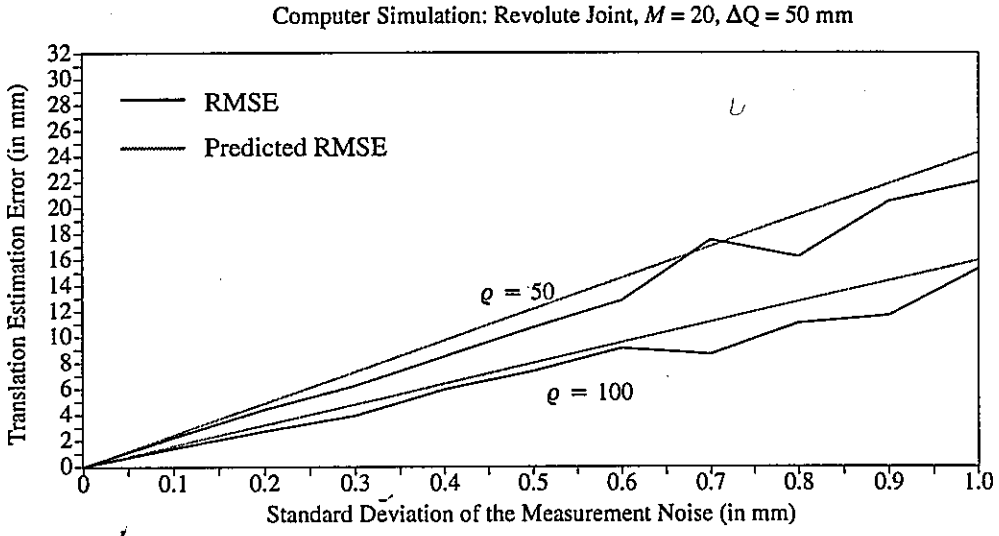

Fig. 25. The translation estimation error versus the standard deviation of the measurement noise.

The last experiment is to test the robustness of the calibration algorithm. The 2D observation error is assumed to be a normal random noise with zero mean and deviation σ_{2D}. Fig. 30 shows the results from computer simulations with each point is obtained from the average of fifty random trials, for $\sigma_{2D} = 0.0$–1.0 pixel. The number of measurements for each joint is set to 20 as in the real experiment. This experiment shows that the proposed method is robust against the observation noise.

VI. CONCLUSIONS

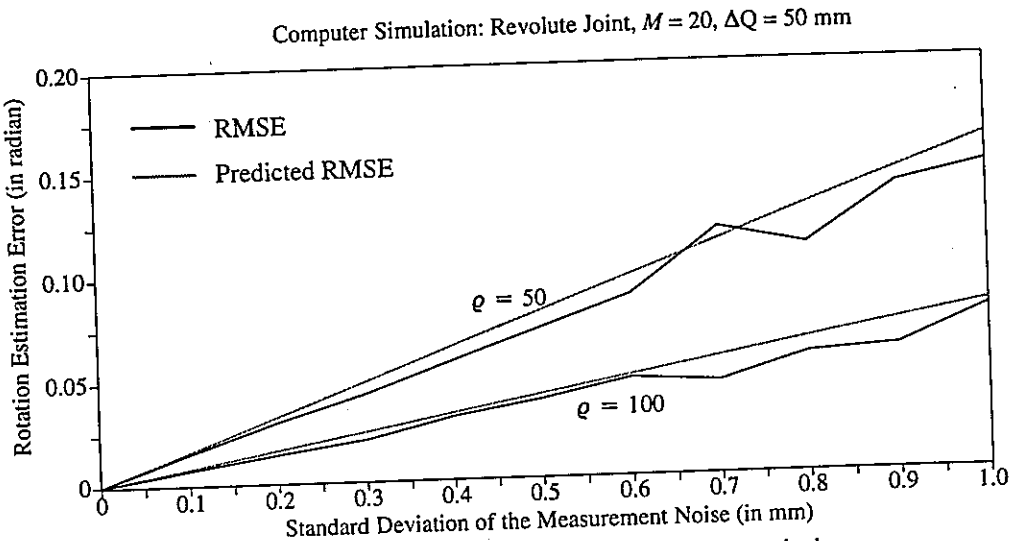

Fig. 26. The rotation estimation error versus the standard deviation of the measurement noise.

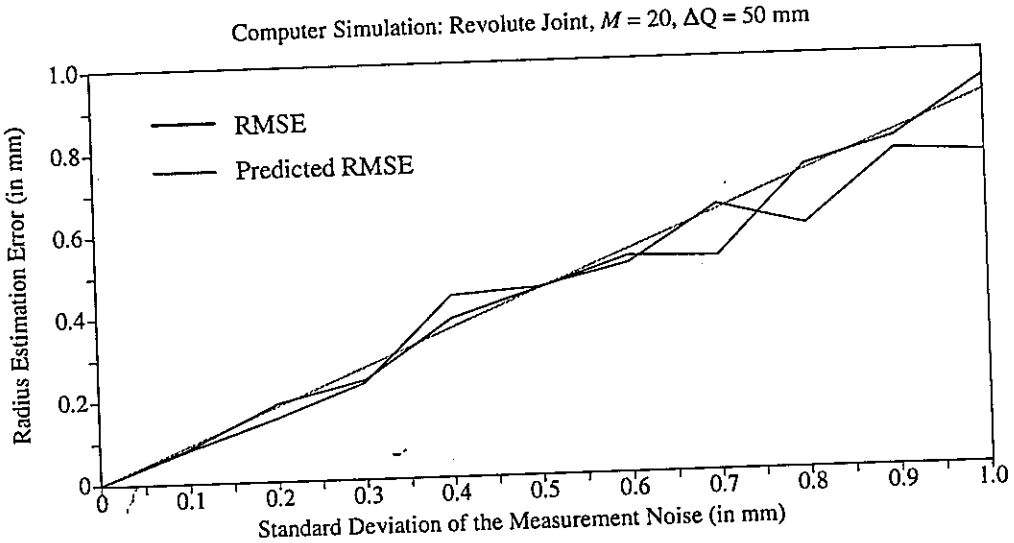

Fig. 27. The radius estimation error versus the standard deviation of the Mesaurement Noise.

Active vision is attracting more and more research interest in the field of computer vision. Being able to acquire information actively, the active vision system has more potential applications than a passive one has. Specifically, in an active stereo vision system, the cameras are able to perform functions such as gazing, panning and tilting. To control the positions and orientations of the cameras, both the kinematic model and its parameters of the active vision mechanism have to be known. Unfortunately, the exact kinematic parameters are usually unknown. In this report we have solved the kinematic parameters identification problem for the IIS head based on the CPC kinematic

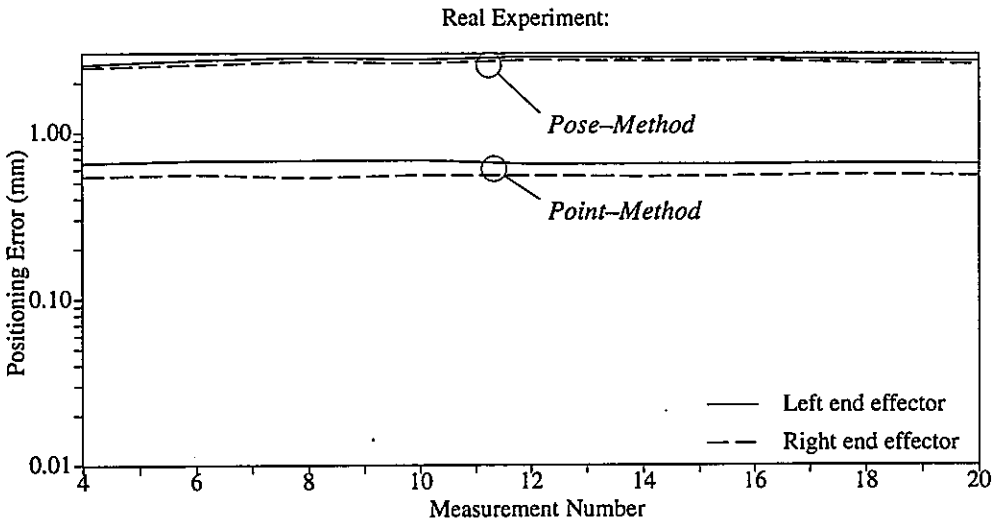

Fig. 28. The positioning error of two end effectors versus the measurement number (19 test points).

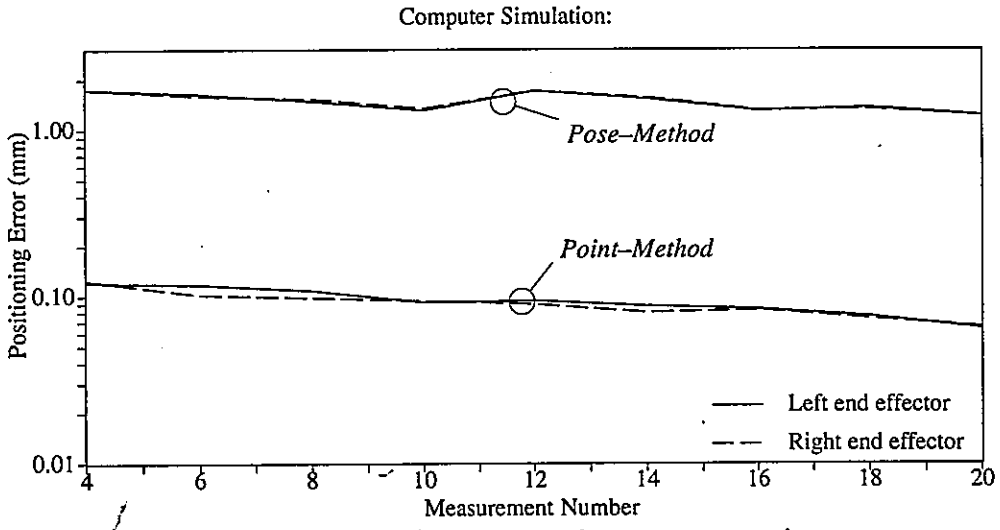

Fig. 29 The positioning error versus the measurement number

model[22]. With our method, the kinematic problem is decomposed into many subproblems of calibrating a single joint. This method can be used for calibrating the kinematic parameters of any robot with or without multiple end–effectors, providing that the links are rigid, the joints are either revolute or prismatic and no closed–loop kinematic chain is included. The proposed closed–form solution method is applied to solve the kinematic calibration problem with a branched kinematic chain having two end–effectors. The calibration results has been tested and the results show that the proposed method of using point measurements can achieve much higher accuracy than that of using pose measurements. Error analysis is also given in this report, which provides the information on

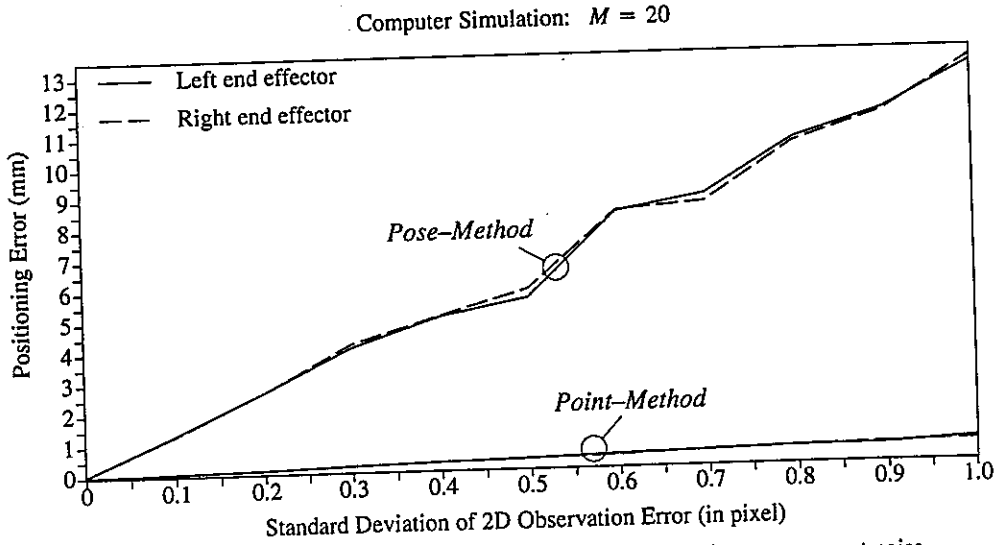

Fig. 30 The positioning error of two end effectors versus the measurement noise.

how to minimize the calibration error. Computer simulations have shown that the derived error variances are reliable.

ACKNOWLEDGEMENT

This research is supported in part by the National Science Council, Taiwan, ROC, under grant NSC–83–0408–E–001–004.

APPENDIX A.

The method presented here is for solving $\boldsymbol{b}_i$ by minimizing the following error

$$\varepsilon \equiv \sum_{j=1}^{M} \left\| {}^{i'}\boldsymbol{p}_j - \boldsymbol{a} - q_{(i+1)j}\, \boldsymbol{b}_i \right\|^2 , \tag{78}$$

where $\boldsymbol{b}_i^T \boldsymbol{b}_i = 1$. Let the gradient of the error function with respect to ξ be zero, we have

$$\boldsymbol{a} = {}^{i'}\overline{\boldsymbol{p}} - \overline{q}_{(i+1)}\, \boldsymbol{b}_i , \tag{79}$$

where ${}^{i'}\overline{\boldsymbol{p}}$ and $\overline{q}_{(i+1)}$ are defined in equations (30) and (31), respectively.

By substituting equation (79) into (78), we have

$$e \equiv \sum_{j=1}^{M} \left\| \boldsymbol{b}_i\, \Delta q_{(i+1)j} - \Delta \boldsymbol{p}_j \right\|^2 , \tag{80}$$

where $\boldsymbol{b}_i^T\ \boldsymbol{b}_i = 1$, $\Delta \boldsymbol{p}_j$ and $\Delta q_{(i+1)j}$ are defined in equations (28) and (29), respectively. To solve the above equation, we first form the Lagrangian

$$l \equiv \sum_{j=1}^{M} \left\{ \boldsymbol{b}_i^T\ \boldsymbol{b}_i\ \Delta q_{(i+1)j}^2 - 2\boldsymbol{b}_i^T\ \Delta \boldsymbol{p}_j\ \Delta q_{(i+1)j} + \Delta \boldsymbol{p}_j^{\ T}\ \Delta \boldsymbol{p}_j \right\} + \lambda\ (1 - \boldsymbol{b}_i^T\ \boldsymbol{b}_i). \tag{81}$$

The gradient of equation (81), with respect to $\boldsymbol{b}_i$, is

$$\nabla l \equiv 2 \sum_{j=1}^{M} \left\{ \boldsymbol{b}_i\ \Delta q_{(i+1)j}^2 - \Delta \boldsymbol{p}_j\ \Delta q_{(i+1)j} \right\} - 2\ \lambda\ \boldsymbol{b}_i. \tag{82}$$

By letting $\nabla l = 0$, we have

$$\boldsymbol{b}_i = \frac{\sum_{j=1}^{M} \left(\Delta \boldsymbol{p}_j\ \Delta q_{(i+1)j} \right)}{\left[\sum_{j=1}^{M} \left(\Delta q_{(i+1)j}^2 \right) - \lambda \right]}, \tag{83}$$

where λ can be determined such that $\boldsymbol{b}_i$ is a unit vector. Consequently, the solution is

$$\boldsymbol{b}_i = \frac{\sum_{j=1}^{M} \left(\Delta \boldsymbol{p}_j\ \Delta q_{(i+1)j} \right)}{\left\| \sum_{j=1}^{M} \left(\Delta \boldsymbol{p}_j\ \Delta q_{(i+1)j} \right) \right\|}. \tag{84}$$

APPENDIX B.

Define the error function as follows,

$$e \equiv \sum_{j=1}^{M} \left\| R\ \boldsymbol{u}_j + \boldsymbol{t} - \varrho\ \boldsymbol{v}_j \right\|^2, \tag{85}$$

where R, $\boldsymbol{t}$ and ϱ are a 3 by 3 rotation matrix, a 3 by 1 translation vector and a positive scalar, respectively. Umeyama has solved a similar problem [19]. However, their error function is of the following form

$$e \equiv \sum_{j=1}^{M} \left\| \varrho \, R \, u_j + t - v_j \right\|^2 . \tag{86}$$

We shall present our derivation of a closed–form optimal solution of equation (85) in this appendix.

Let the gradient of the error function (85) with respect to t be zero, we have

$$t = \varrho \, \bar{v} - R \, \bar{u} , \tag{87}$$

where $\bar{u} \equiv \frac{1}{N} \sum_{j=1}^{M} u_j$ and $\bar{v} \equiv \frac{1}{N} \sum_{j=1}^{M} v_j$. Substituting equation (87) into (85), we have

$$e \equiv \sum_{j=1}^{M} \left\| R \, \underline{u}_j - \varrho \, \underline{v}_j \right\|^2 , \tag{88}$$

where $\underline{u}_j = u_j - \bar{u}$ and $\underline{v}_j = v_j - \bar{v}$. Again, by letting the gradient of the equation (88) with respect to ϱ be zero, we have

$$\varrho = \frac{\sum_{j=1}^{M} \underline{v}_j^T \, R \, \underline{u}_j}{\sum_{j=1}^{M} \underline{v}_j^T \, \underline{v}_j} . \tag{89}$$

By substituting equation (89) into (88), we have

$$e \equiv \sum_{j=1}^{M} \underline{u}_j^T \, \underline{u}_j - \frac{\left[\sum_{j=1}^{M} \underline{v}_j^T \, R \, \underline{u}_j \right]^2}{\sum_{j=1}^{M} \underline{v}_j^T \, \underline{v}_j} , \tag{90}$$

Notice that the first term in equation (90) is independent to the unknown rotation matrix R, therefore, to minimize the error function (85) is equivalent to maximize the following function

$$\eta \equiv \sum_{j=1}^{M} \underline{v}_j^T \, R \, \underline{u}_j , \tag{91}$$

subject to $R^T \, R = I$. The solution to the above equation can be computed by following the

procedures for calculating a closed–form solution to equation (36) in Section III, Steps 2–5 (refer to Arun et. al.[3], Umeyama[19]).

APPENDIX C.

Assuming that M is an even number, then from equation (29), we have

$$\Delta q_{(i+1)j} = (j - \frac{M+1}{2})\ Q,\ j = 1, 2, ..., M \tag{92}$$

and

$$\delta P = \sum_{j=1}^{M} \delta p_j\ \Delta q_{(i+1)j} - \delta \overline{p}\ \sum_{j=1}^{M} \Delta q_{(i+1)j}\ . \tag{93}$$

Notice that the second term of equation (93) on the right hand side is zero because

$$\delta \overline{p}\ \sum_{j=1}^{M} \Delta q_{(i+1)j} = \delta \overline{p}\ \sum_{j=1}^{M} \left(q_{(i+1)j} - \overline{q}_{(i+1)} \right) = \delta \overline{p} \cdot 0. \tag{94}$$

From Lemma 1 and assumption (A.2), we have the covariance matrix

$$Cov[\delta P] = Q^2\ \sigma^2 \sum_{j=1}^{M} \left(j - \frac{M+1}{2} \right)^2 I_{3\times 3}, \tag{95}$$

which yields

$$Cov[\delta P] = \sigma_{\delta P}^2\ I_{3\times 3}\ , \tag{96}$$

where $\sigma_{\delta P}^2 \equiv Q^2\ \sigma^2\ \frac{M^3 - M}{12}$.

Notice that $\Delta \boldsymbol{p}_j$ in equation (43) is noise free, therefore, the length of $\Delta \boldsymbol{p}_j$ is exactly $\Delta q_{(i+1)j}$ and each vector $\Delta \boldsymbol{p}_j$, $j = 1, 2, ..., M$, has exactly the same direction as is the true kinematic parameter $\boldsymbol{b}_i$, i.e, $\Delta \boldsymbol{p}_j = \Delta q_{(i+1)j}\ \boldsymbol{b}_i$. Hence, from equations (43) and (92), we have

$$\|P\| = \sum_{j=1}^{M} \Delta q_{(i+1)j}^2 = Q^2 \frac{M^3 - M}{12}. \tag{97}$$

The covariance matrix of $\delta\beta$ can be derived from equations (96) and (97) as follows (remember that $Q = \Delta Q/(M-1)$).

$$Cov[\delta\beta] = \sigma_{\delta\beta}^2 \, I_{3\times 3} \,, \tag{98}$$

where
$$\sigma_{\delta\beta}^2 = \frac{\sigma_{\delta P}^2}{\|P\|^2} = T_\beta(M) \, \frac{\sigma^2}{\Delta Q^2} \tag{99}$$

and
$$T_\beta(M) = \frac{12(M-1)}{M\,(M+1)}. \tag{100}$$

In general, if the number of measurement is odd, the derived covariance matrix for a even measurement number can be used as a reasonable approximation.

APPENDIX D.

Expanding equation (66) as the follows.

$$e = \delta\phi^T \sum_{j=1}^{M}\left[Skew\left(\varrho^* \underline{v}_j\right)^T \, Skew\left(\varrho^* \underline{v}_j\right)\right] \delta\phi + \sum_{j=1}^{M}\left[\delta u'^T_j \, \delta u'_j\right] + \sum_{j=1}^{M}\left[\delta\varrho^2 \, \underline{v}_j^T \, \underline{v}_j\right]$$

$$-2\,\delta\phi^T \sum_{j=1}^{M}\left[Skew\left(\varrho^* \underline{v}_j\right)^T \delta u'_j\right] + 2\sum_{j=1}^{M}\left[\delta\varrho \; \delta u'^T_j \, \underline{v}_j\right] + 2\delta\phi^T \sum_{j=1}^{M}\left[Skew\left(\varrho^* \underline{v}_j\right)^T \underline{v}_j \, \delta\varrho \right]. \tag{101}$$

Notice that the last term in equation (101) is equal to zero, because

$$Skew\left(\varrho^* \underline{v}_j\right) \underline{v}_j = \varrho^* \underline{v}_j \times \underline{v}_j = 0 \,. \tag{102}$$

By minimizing the error in equation (101), we have

$$\varrho^* \sum_{j=1}^{M}\left[Skew\left(\underline{v}_j\right)^T \, Skew\left(\underline{v}_j\right)\right] \delta\phi = \sum_{j=1}^{M}\left[Skew\left(\underline{v}_j\right)^T \, \delta u'_j\right], \tag{103}$$

and
$$\sum_{j=1}^{M}\left[\underline{v}_j^T \, \underline{v}_j\right] \delta\varrho = \sum_{j=1}^{M}\left[\delta u'^T_j \, \underline{v}_j\right]. \tag{104}$$

Note that the left hand side of equation (103) can be approximately represented as follows.

$$\varrho^* \sum_{j=1}^{M}\left[Skew\left(\underline{v}_j\right)^T Skew\left(\underline{v}_j\right)\right]\delta\phi \approx \frac{\varrho^* M}{\Delta Q} \int_{-\Delta Q/2}^{\Delta Q/2}\left[Skew\left(\underline{v}_j(\theta)\right)^T Skew\left(\underline{v}_j(\theta)\right) d\theta \right]\delta\phi, \tag{105}$$

which leads to

$$\varrho^* M \begin{bmatrix} \left(\frac{1}{2} - \frac{\sin(\Delta Q)}{2\Delta Q}\right)\delta\phi_x \\ \left(\frac{1}{2} + \frac{2 - 2\cos(\Delta Q) - 8\sin(\Delta Q/2)^2}{\Delta Q^2} + \frac{\sin(\Delta Q)}{2\Delta Q}\right)\delta\phi_y \\ \left(\frac{\Delta Q^2 - 2(1 - \cos(\Delta Q))}{\Delta Q^2}\right)\delta\phi_z \end{bmatrix} = \sum_{j=1}^{M}\left[Skew\left(\underline{v}_j\right)^T \delta u'_j \right], \quad (106)$$

Let $\underline{v}_j \equiv \left[\underline{v}_{xj} \;\; \underline{v}_{yj} \;\; \underline{v}_{zj}\right]^T$ and $\delta u'_j \equiv \left[\delta u_{xj} \;\; \delta u_{yj} \;\; \delta u_{zj}\right]^T$, then the right hand side of equation (106) can be rewritten as follows.

$$\sum_{j=1}^{M}\left[Skew\left(\underline{v}_j\right)^T \delta u'_j \right] = \begin{bmatrix} \sum_{j=1}^{M} \underline{v}_{yj}\, \delta u_{zj} \\ -\sum_{j=1}^{M} \underline{v}_{xj}\, \delta u_{zj} \\ \sum_{j=1}^{M}\left(-\underline{v}_{yj}\, \delta u_{xj} + \underline{v}_{xj}\, \delta u_{yj}\right) \end{bmatrix}. \quad (107)$$

Computing the variance of $\delta\phi_x$ from equations (106) and (107), we have

$$\varrho^{*2}\, M^2 \left(\frac{1}{2} - \frac{\sin(\Delta Q)}{2\Delta Q}\right)^2 \sigma^2_{\delta\phi_x} = \sum_{j=1}^{M} \underline{v}^2_{yj}\, \sigma^2, \quad (108)$$

where the right hand side of the above equation can be approximated as follows.

$$\sum_{j=1}^{M} \underline{v}^2_{yj}\, \sigma^2 \approx \frac{M\, \sigma^2}{\Delta Q} \int_{-\Delta Q/2}^{\Delta Q/2} \underline{v}^2_{yj}(\theta) d\theta = \sigma^2\, M\left(\frac{1}{2} - \frac{\sin(\Delta Q)}{2\Delta Q}\right), \quad (109)$$

Therefore, the variance of the first orientation error is

$$\sigma^2_{\delta\phi_x} = \frac{\sigma^2}{\varrho^{*2}\, M}\, \Phi_x(\Delta Q), \quad (110)$$

where

$$\Phi_x(\Delta Q) \equiv \frac{2\Delta Q}{\Delta Q - \sin(\Delta Q)} \quad (111)$$

Similarly, the second and the third orientation error variance can be computed as follows.

$$\sigma^2_{\delta\phi_y} = \frac{\sigma^2}{\varrho^{*2}\, M}\ \Phi_y(\Delta Q) \tag{112}$$

and

$$\sigma^2_{\delta\phi_z} = \frac{\sigma^2}{\varrho^{*2}\, M}\ \Phi_z(\Delta Q) \tag{113}$$

where

$$\Phi_y(\Delta Q) \equiv \frac{2\Delta Q^2}{\Delta Q^2 - 4(1 - \cos(\Delta Q)) + \Delta Q \sin(\Delta Q)}, \tag{114}$$

$$\Phi_z(\Delta Q) \equiv \frac{\Delta Q^2}{\Delta Q^2 - 2(1 - \cos(\Delta Q))}. \tag{115}$$

Also, from equation (104), we have

$$M\left(\frac{\Delta Q^2 - 2(1 - \cos(\Delta Q))}{\Delta Q^2}\right)\delta\varrho = \sum_{j=1}^{M}\left[\underline{v}_{xj}\ \delta u_{xj} + \underline{v}_{yj}\ \delta u_{yj}\right]. \tag{116}$$

The error variance of the estimated radius can be computed from equation (116) as follows.

$$\sigma^2_{\delta\varrho} = \frac{\sigma^2}{M}\ \Phi_z(\Delta Q)\,. \tag{117}$$

APPENDIX E.

From equations (53), (54), (55) and (60), the estimation error of the translation vector can be derived as follows.

$$\delta t = \delta\varrho\ \bar{v} - \delta\bar{u} + \varrho^*\ Skew(\bar{v})\ \delta\phi - Skew\left(t^*\right)\ \delta\phi. \tag{118}$$

The average of vectors v_j, $j = 1, 2, ..., M$, is

$$\bar{v} = \frac{1}{M}\sum_{j=1}^{M} v_j \approx \frac{1}{\Delta Q}\int_{-\Delta Q/2}^{\Delta Q/2} v(\theta)\ d\theta = \begin{bmatrix} \bar{v}_x \\ 0 \\ 0 \end{bmatrix}, \tag{119}$$

where

$$\bar{v}_x \equiv \frac{2\sin\left(\frac{\Delta Q}{2}\right)}{\Delta Q}. \tag{120}$$

Therefore, from equation (118), we have

$$\delta t = \begin{bmatrix} \delta t_x \\ \delta t_y \\ \delta t_z \end{bmatrix} = \begin{bmatrix} \bar{v}_x\ \delta r + t_z\ \delta\phi_y - t_y\ \delta\phi_z \\ -\varrho^*\bar{v}_x\ \delta\phi_z - t_z\ \delta\phi_x + t_x\ \delta\phi_z \\ \varrho^*\bar{v}_x\ \delta\phi_y + t_y\ \delta\phi_x - t_x\ \delta\phi_y \end{bmatrix}. \tag{121}$$

Observe equations (106) (107) and (116). Random variable $\delta\phi_x$ and $\delta\phi_y$ are linear combinations of $\delta u_{z1}, \delta u_{z2}, ..., \delta u_{zM}$, and $\delta\varrho$ and $\delta\phi_z$ are linear combinations of $\delta u_{x1}, \delta u_{x2}, ..., \delta u_{xM}$ and $\delta u_{y1}, \delta u_{y2}, ..., \delta u_{yM}$. Since δu_{xj}, δu_{yj} and δu_{zj} are mutually independent random variables, therefore, $\delta\phi_x$ or $\delta\phi_y$ and $\delta\phi_z$ or $\delta\varrho$ are independent random variables. Hence, the variance of δt_y can be easily derived as follows

$$\sigma^2_{\delta t_y} = \left[-\varrho^* \frac{2\sin\left(\frac{\Delta Q}{2}\right)}{\Delta Q} + t_x \right]^2 \sigma^2_{\delta\phi_z} + t_z^2\ \sigma^2_{\delta\phi_x}. \tag{122}$$

For deriving the variance of δt_x, we have to consider the effect of dependency of $\delta\phi_z$ and $\delta\varrho$. From equations (106) and (107), we have

$$\delta\phi_z = \frac{\Phi_z(\Delta Q)}{\varrho^* M} \sum_{j=1}^{M} \left[-\underline{v}_{yj}\ \delta u_{xj} + \underline{v}_{xj}\ \delta u_{yj} \right]. \tag{123}$$

From equation (116), we have

$$\delta\varrho = \Phi_z(\Delta Q) \sum_{j=1}^{M} \left[\underline{v}_{xj}\ \delta u_{xj} + \underline{v}_{yj}\ \delta u_{yj} \right]. \tag{124}$$

Substituting equations (123) and (124) into (121), and extracting the equations related to δt_x, we have

$$\delta t_x = \Phi_z(\Delta Q) \left[\sum_{j=1}^{M} \left(\bar{v}_x\ \underline{v}_{xj} + \frac{t_y}{\varrho^* M} \underline{v}_{yj} \right) \delta u_{xj} + \sum_{j=1}^{M} \left(\bar{v}_x\ \underline{v}_{yj} - \frac{t_y}{\varrho^* M} \underline{v}_{xj} \right) \delta u_{yj} \right] + t_z\ \delta\phi_y. \tag{125}$$

By applying Lemma 1 and using the integral result to approximate the one obtained by summation, and notice that $\underline{v}_{xj} = \cos\left(q_{(i+1)j}\right) - \bar{v}_x$ and $\underline{v}_{yj} = \sin\left(q_{(i+1)j}\right)$ are respectively even and odd functions with respect to the joint angle, which yields

$$\sum_{j=1}^{M} \underline{v}_{xj}\underline{v}_{yj} \approx \frac{M}{\Delta Q} \int_{-\Delta Q/2}^{\Delta Q/2} \underline{v}_x(q)\ \underline{v}_y(q)\ dq = 0, \tag{126}$$

therefore,

$$\sigma^2_{\delta t_x} = \frac{4\sin^2\left(\frac{\Delta Q}{2}\right)}{\Delta Q^2}\ \sigma^2_{\delta \varrho} + t_z^2\ \sigma^2_{\delta \phi_y} + t_y^2\ \sigma^2_{\delta \phi_z}. \tag{127}$$

Substituting equations (123) and (124) into (121), and extracting the equations related to δt_z, we have.

$$\delta t_z = \sum_{j=1}^{M}\Big[- \Phi_y(\Delta Q)\ \left(\varrho^*\ \bar{v}_x - t_x\right)\ \underline{v}_{xj} + \Phi_x(\Delta Q)\ t_y\ \underline{v}_{yj}\Big]\ \delta u_{zj}. \tag{128}$$

Similarly, by using equation (126), the variance of the estimation error, δt_z, can be derived as follows

$$\sigma^2_{\delta t_z} = \left[\varrho^*\ \frac{2\sin\left(\frac{\Delta Q}{2}\right)}{\Delta Q} - t_x\right]^2\ \sigma^2_{\delta \phi_y} + t_y^2\ \sigma^2_{\delta \phi_x}. \tag{129}$$

REFERENCES

[1] A.L. Abbott, "Dynamic Integration of Depth Cues for Surface Reconstruction from Stereo Images," Ph.D dissertation, University of Illinois as Urbana–Champaign, 1990.

[2] J. Aloimonos and A. Badyopadhyay, "Active Vision," IEEE Proceedings of the first Int. Conf. on Computer Vision, pp. 35–54, June 1987.

[3] K. S. Arun, T.S. Huang, and S.D. Blostein, "Least–Square Fitting of Two 3–D Point Sets," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI–9, NO. 5, pp. 698–700, 1987.

[4] D.J. Bennett, J.M. Hollerbach, "Identifying the Kinematics of Robots and their Tasks," Proceedings of the Int. Conf. on Robotics and Automation, pp. 580–586, 1989.

[5] J.L. Caenen, J.C. Angue, "Identification of Geometric and Non Geometric Parameters of Robots," Proceedings of the Int. Conf. on Robotics and Automation, pp. 1032–1037, 1990.

[6] G. Duelen, U. Kirchhoff, J. Held, "Methods of Identification of Geometrical Data in Robot Kinematics," Robotics & Computer–Integrated Manufacturing, vol. 4, no. 1/2, pp. 181–185, 1988.

[7] S. Hayati, K. Tso and G. Roston, "Robot Geometry Calibration," Proceedings of the Int. Conf. on Robotics and Automation, pp. 947–951, 1988.

[8] W. Khalil, M. Gautier, C. Enguehar, "Identifiable Parameters and Optimum Configurations for Robots Calibration," Robotica, vol. 9, pp. 63–70, 1991.

[9] D.H. Kim, K.H. Cook and J.H. Oh, "Identification and compensation of robot kinematic parameter for positioning accuracy improvement," Robotica, vol. 9, pp. 99–105, 1991.

[10] R.K. Lenz, R.Y. Tsai, "Calibrating a Cartesian Robot with Eye–on–Hand Configuration Independent of Eye–to–Hand Relationship," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 11, no. 9, pp. 916–928, Sep. 1989.

[11] B.W. Mooring, T. J. Pack, "Calibration Procedure for an Industrial Robot," Proceedings of the Int. Conf. on Robotics and Automation, pp. 786–791, 1988.

[12] B.W. Mooring, Z.S. Roth and M.R. Driels, *Fundamentals of Manipulator Calibration*, A Wiley–Interscience Publication, John Wiley & Sons, Inc., 1991.

[13] S.W. Shih, J.S. Jin, K.H. Wei, Y.P. Hung, "Kinematic Calibration of a Binocular Head Using Stereo Vision with the Complete and Parametrically Continuous Model," SPIE Proceedings, Intelligent Robots and Computer Vision XI, vol. 1825, pp. 643–657, 1992.

[14] S.W. Shih, Y.P. Hung, W.S. Lin, "Head/Eye Calibration of a Binocular Head by use of Single Calibration Point," Proceedings of the IEEE Southwest Symposium on Image Analysis and

Interpretation, Dalas, Texas, pp. 154–159, April, 1994.

[15] S.W. Shih, Y.P. Hung and W.S. Lin, "Comments on 'A Linear Solution to Kinematic Parameter identification of Robot Manipulator' and Some Modifications," to appear in IEEE Trans. on Robotics and Automation.

[16] H.W. Stone, *Kinematic Modeling, Identification, and Control of Robotic Manipulators.* Norwell, MA: Kluwer Academic, 1987.

[17] M.E. Sklar, "Metrology and Calibration Techniques for the Performance Enhancement of Industrial Robots," Ph.D. Dissertation, the University of Texas at Austin, 1988.

[18] R.Y. Tsai, R.K. Lenz, "A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration," IEEE Trans. on Robotics and Automation, vol. 5, NO. 3, pp. 345–358, June 1989.

[19] S. Umeyama, "Lease–Squares Estimation of Transformation Parameters Between Two Point Patterns," IEEE Trans. Pattern Anal. Machine Intell., vol. 13, NO. 4, pp. 376–380, April 1991.

[20] W.K. Veitschegger, C. Wu, "A Method for Calibrating and Compensating Robot Kinematic Errors," Proceedings of the Int. Conf. on Robotics and Automation, pp. 39–44, 1987.

[21] G.S. Young, T.H. Hong, M. Herman and J.C.S. Yang, "Kinematic Calibration of an Active Camera System," IEEE Proceedings of the Int. Conf. on Computer Vision Pattern Recognition, pp. 748–751, 1992.

[22] H. Zhuang, Z.S. Roth, F. Hamano, "A Complete and Parametrically Continuous Kinematic Model for Robot Manipulators," IEEE Trans. on Robotics and Automation, vol. 8, no. 4, pp. 451–463, Aug. 1992.

[23] H. Zhuang, Z.S. Roth, "A Linear Solution to the Kinematic Parameter Identification of Robot Manipulators," IEEE Trans. on Robotics and Automation, Vol. 9, No. 2, pp. 174–185, April 1993.