Real Time Versatile Robotics Hand/Eye Calibration using 3D Machine Vision

Roger Y. Tsai and Reimar K. Lenz [1]

Manufacturing Research
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598

Abstract

This paper describes a new technique for computing 3D position and orientation of a camera relative to the last joint of a robot manipulator in an eye-on-hand configuration. The calibration can be done within a fraction of a millisecond after the robot finishes the movement. The setup is very simple (a planar set of calibration points arbitrarily placed on the work table, in addition to robot and camera) and is the same as that for a common camera calibration. To the best of our knowledge, this method is faster, simpler and more accurate than any existing technique for hand/eye calibration. A series of generic geometric properties or lemma are presented, leading to the derivation of the final algorithms, which are aimed at simplicity, efficiency and accuracy while giving ample geometric and algebraic insights. Besides describing the new technique, critical factors influencing the accuracy are analyzed, and procedures for improving accuracy are introduced. Test results of both simulation and real experiments on an IBM Cartesian robot are reported and analyzed. A preliminary version of this paper has been published in Ref. 9. This paper gives more critical details on the proofs and derivations of the algorithms and the lemmas leading to the algorithm, while Ref. 9 contains more details on the accuracy analysis and experiments.

1.1 Introduction

1.1.1 What is 3D Robotics Hand/Eye Calibration?

3D robotics hand/eye calibration is the task of computing the relative 3D position and orientation between the camera and the robot gripper in an eye-on-hand configuration, meaning that the camera is rigidly connected to the robot gripper. The camera is either grasped by the gripper, or just fastened to it. More specifically, this is the task of computing the relative rotation and translation (homogeneous transformation) between two coordinate frames, one centered at the camera lens center, and the other at the robot gripper. The gripper coordinate frame is centered on the last link of the robot manipulator, and as we shall see in this paper, the robot manipulator must possess enough degrees of freedom so as to be able to rotate the camera around two different axes while at the same time keeping the camera focused on a stationary calibration object in order to resolve uniquely the full 3D geometric relationship between the camera and the gripper.[2]

1.1.2 Why Isn't It Trivial?

It is obvious to see that if the robot knows the exact 3D positions of a number of points on a calibration setup in the robot world coordinate system as well as 3D location of the gripper, while at the same time, the camera can view these points in a proper way, then it is possible to determine the 3D homogeneous transformation between the camera and the calibration world coordinate frame, making it a trivial matter to compute the homogeneous transformation between the camera and the manipulator. However, it is very difficult, if possible, for the robot to acquire the accurate knowledge of the 3D positions of a number of feature points easy enough for camera to view simultaneously with the right resolution, field of view, etc., while the position information has to be known in the robot world coordinate system. Some researchers treat the difference between the calibration world coordinate system and the robot world coordinate system as a 6-degree-freedom unknown, and incorporate them into a much larger non-linear optimization process (see "Why Isn't the State of the Art Sufficient?" on page 2). We propose a much easier and faster approach.

1.1.3 Why is Hand/Eye Calibration Important?

It is important in several aspects:

Automated 3D Robotics Vision Measurement

When vision is used to measure the 3D geometric relationships between different parts of an object in a robotics work cell, it is often necessary to use the manipulator to move the vision sensor to different positions in the work space in order to see different features of the object (see Tsai and Lavin, 1984). At each position, the 3D position and orientation of the feature measured by the vision system is only relative to the vision sensor. As the manipulator moves the sensor to different positions, the measurements taken at different positions are not related to one another unless we know the 3D relative position and orientation of the sensor at different locations. If the robot system is capable of knowing where the gripper is in the robot world coordinate system to some degree of accuracy, then it should know how much 3D motion it has undertaken from one position to another. Since the camera is rigidly connected to the gripper, of course it too undergoes the same rigid body motion, BUT ONLY in the robot world coordinate system. If the hand/eye calibration is not done, one does not know the 3D homogenous transformation between the camera 3D coordinate systems at different locations simplify from the motion of the robot manipulator.

Automated Sensor Placement Planning

In order to do automated 3D measurement with robot vision, sensor planning is vital in order to automatically determine the optimum positions of the sensor in order to view all the desired features while taking care of problems of occlusion, depth of focus, resolution, field of view, etc. However, even if the robot knows where to put the sensor for optimum viewing, it does not know where the manipulator should be in order to achieve this goal, unless the 3D geometric relationships between the last link of the manipulator and the sensor is known.

Automated Part Acquisition or Assembly

When vision is used to aid the robot in grasping an object for automated assembly or part transport with eye-on-hand configuration, unless iterative visual feedback is used, the vision system may be able to determine where the part is relative to the sensor, but the robot does not know how to place the manipulator to grasp it. This problem can be resolved if the robot hand/eye calibration is done.

Stereo Vision

[1] Reimar K. Lenz is now with the Lehrstuhl fuer Nachrichtentechnik, Technical University of Munich, FRG.

[2] It takes at least two rotary joints and one linear joint, or three rotary joints. It is possible to use just two rotary joints, but the rotation axes for these two joints must coincide at the calibration block.

If only one camera is used to do stereo vision, one way to create a stereo base is to move the camera with the manipulator [3]. Although the robot system may know how much the manipulator has moved, it does not know the homogenous transformation between the 3D camera coordinate system, even if the camera undergoes the same rigid body motion as the gripper does (since this rigid body motion is defined only in the robot world coordinate system). Again, when the hand/eye calibration is done, this problem is solved.

1.1.4 Why Isn't the State of the Art Sufficient?

From our literature survey, there are two categories of approaches for doing robot hand/eye calibration:

Coupling Hand/Eye Calibration with Conventional Robot Kinematic Model Calibration

References (partial list): G. Puskorius and L. Feldkamp (1987); A. Izaquirre (1987); M. Bowman & A. Forrest (submitted for publication)

In this approach, global nonlinear optimization is done over the robot kinematic model parameters and the hand/eye parameters silmultanously, making the number of unknowns generally over 30. Such large scale nonlinear optimization is very time consuming, and needs a very good initial guess and accurate data for convergence. It also cannot easily exploit the use of redundant images and stations for reducing error since the computation would become prohibitive.

Decoupling Hand/Eye Calibration from Conventional Robot Kinematic Model Calibration

References (partial list): Y. Shiu and S. Ahmad (1987); R. Tsai and R. Lenz (1987, this paper) (partial list): G. Puskorius and L. Feldkamp (1987); A. Issaquirre (1986); M. Bowman & A. Forrest (submitted for publication)

As far as we know, Shiu and Ahmad's work and the work reported in this paper are the first attempts to decouple the hand/eye calibration from robot model calibration and not use global high dimensional nonlinear optimization. The starting point between these two works are similar (although independently developed), the solutions are very different. In Shiu and Ahmad's method, the number of unknowns to solve for is twice the number of degrees of freedom, since they treat sin and cos as independent. We found it advantageous to use redundant frames to improve accuracy, but in our algorithm, the number of unknowns stays the same no matter how many frames are used simultaneously, and for each additional frame, only 60 additional arithmetic scalar operations are needed (each operation takes less than half a microsecond on a typical minicomputer). In Shiu and Ahmond's method, the number of unknowns increases by two for each extra frame. Our procedure is simpler and faster, and the derivation procedure is also simpler. We have also done extensive error analysis, simulation or real experiments for testing the accuracy potential or problems of hand/eye calibration, and propose means for improving accuracy.

1.2 The New Approach

1.2.1 Basic Setup

Figure 1 is a schematic depiction of the basic setup. Figure 2 shows two photos of the actual setup. The robot carrying a camera makes a series of motions with the camera acquiring a picture of a calibration object at the pause of each motion. The calibration object is a block with an array of target points on the top surface. The position of each calibration point is known very accurately relative to an arbitrarily selected coordinate system setup on the block (see Tsai, 1986, 1987; Lenz and Tsai, 1987; Tsai and Lenz, 1987B). Detailed description of the setup can found in "Simulation and Real Experiment Results" on page 7. The following is a list of definitions for the various coordinate frames (Note: All coordinate frames mentioned here are Cartesian coordinate frames in 3D):

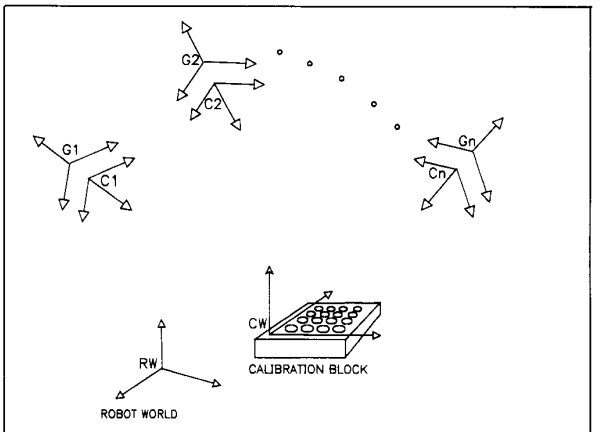

Figure 1. Basic setup for robot hand/eye calibration. C_i and G_i are coordinate frames for the camera and gripper respectively.

G_i : The gripper coordinate system. That is, the coordinate frame fixed on the robot gripper and as the robot moves, it moves with the gripper.

C_i : The camera coordinate system. That is, the coordinate frame fixed on the camera, with the z axis coinciding with the optical axis, and the x, y axes parallel to the image X, Y axes

CW : The calibration block world coordinate frame. This is an arbitrarily selected coordinate frame set on the calibration block so that the coordinate of each target point on the calibration block is known *a priori* relative to CW.

RW : The robot world coordinate frame. It is fixed in the robot work station, and as the robot arm moves around, the encoder output of all the

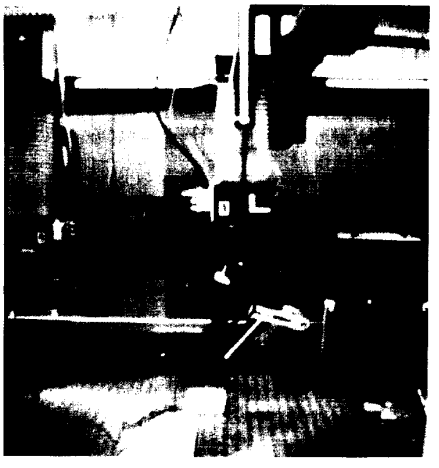

Figure 2. The physical setup. A CCD camera is rigidly mounted on the last joint of an IBM Clean Room Robot for performing hand/eye calibration.

[3] This is not highly recommendable except in low accuracy applications. It is better for the robot to carry a stereo pair of cameras or laser-camera pair, or to use one camera with model based location determination.

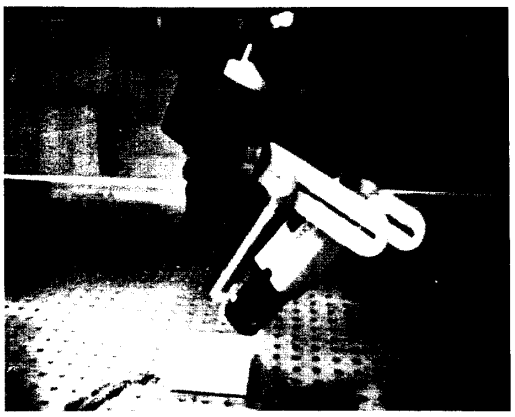

Figure 3. The physical setup. A CCD camera is rigidly mounted on the last joint of an IBM Clean Room Robot for performing hand/eye calibration.

robot joints enable the system to tell where the gripper is relative to RW.

Definition of a list of Homogeneous Transformation Matrix

H_{gi} defines coordinate transformation from G_i to RW (1)

$$H_{gi} \equiv \begin{bmatrix} R_{gi} & T_{gi} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

H_{ci} defines coordinate transformation from CW to C_i (2)

$$H_{ci} \equiv \begin{bmatrix} R_{ci} & T_{ci} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

H_{gij} defines coordinate transformation from G_i to G_j (3)

$$H_{gij} \equiv \begin{bmatrix} R_{gij} & T_{gij} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

H_{cij} defines coordinate transformation from C_i to C_j (4)

$$H_{cij} \equiv \begin{bmatrix} R_{cij} & T_{cij} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

H_{cg} defines coordinate transformation from C_i to G_i (5)

$$H_{cg} \equiv \begin{bmatrix} R_{cg} & T_{cg} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

In the above, i,j range from 1 to N, where N is the number of stations in Figure 1 on page 2 where the camera grabs pictures of the calibration block. Figure 4 illustrates the relationship between the homogeneous matrices and the various coordinate frames in Figure 1 on page 2. Note that H_{cg} does not have any station index (i or j). This is because the camera is rigidly mounted on the gripper of the robot arm and therefore, H_{cg} is the same for all stations.

1.2.2 What Are the Observables and What is to be computed?

1.2.2.1 The Observables or the Measurables

The Observables are H_{ci} and H_{gi} for i=1,...,N. H_{ci} is obtained from computing the extrinsic calibration parameters(see Tsai 1986, 1987; Tsai and Lenz 1987) using the image grabbed at the ith pause of robot movement. It defines the relative 3D rotation and translation from CW to C_i. For 36 calibration points, it takes about 20 msec to compute, and is accurate to one part in 4000. The other set of observables are H_{gi}s . Any robot that can supply the information of where the gripper is within the robot workstation is capable of delivering H_{gi} . This requires good robot calibration. Actually, even if H_{gi} may be bad, so long as if H_{gij} is good, there is no problem. This due to the fact the computational procedure entails only H_{gij}, but not H_{gi}.

1.2.2.2 Elements to be Computed

Intermediate: H_{gij}, H_{cij}

Since H_{gi} defines transformation from G_i to RW , and H_{gj} from G_j to RW, obviously,

$$H_{gij} = H_{gj}^{-1} H_{gi} \tag{6}$$

Similarly,

$$H_{cij} = H_{cj} H_{ci}^{-1} \tag{7}$$

Notice that (6) and (7) are incompatible in terms of where the inverse signs are placed. This is due to the fact that H_{gi} is from G_i to RW while H_{ci} is from CW to C_i .

Final: H_{cg}

Notice that if RW coincided with CW, it would be trivial to compute H_{cg}, which in this case would be equal to $H_{gi}^{-1}H_{ci}^{-1}$. However, it is very difficult for the coordinate system on the calibration block to be set in a fixed and precisely known 3D relationship with respect to the robot coordinate system such that the positions of all the points on the calibration block are known relative to the robot.

1.2.3 Some Basic Background for a General Rotation Matrix and Its Real Eigenvectors

Before describing the new technique , we introduce the representation for transformation (in particular, rotations) used below. It is well known (Rogers and Adams, 1976) that any rigid body motion or Cartesian coordinate transformation can be modelled as a rotation by an angle θ around an axis through the origin with direction cosines n_1, n_2, n_3, followed by a translation T such that

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R \begin{bmatrix} x \\ y \\ z \end{bmatrix} + T \quad or \quad \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

where (x, y, z) and (x', y', z') are the coordinates of any point before and after the transformation.

R is a 3 × 3 orthonormal matrix of the first kind (i.e., $\det(R) = 1$)

$$R = \begin{bmatrix} n_1^2 + (1 - n_1^2)\cos\theta & n_1 n_2(1 - \cos\theta) - n_3 \sin\theta & n_1 n_3(1 - \cos\theta) + n_2 \sin\theta \\ n_1 n_2(1 - \cos\theta) + n_3 \sin\theta & n_2^2 + (1 - n_2^2)\cos\theta & n_2 n_3(1 - \cos\theta) - n_1 \sin\theta \\ n_1 n_3(1 - \cos\theta) - n_2 \sin\theta & n_2 n_3(1 - \cos\theta) + n_1 \sin\theta & n_3^2 + (1 - n_3^2)\cos\theta \end{bmatrix} \tag{8}$$

H is the homogeneous transformation matrix (used in (1)-(5)), and is defined as

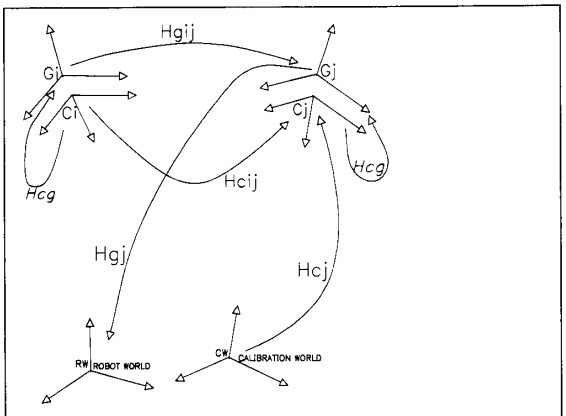

Figure 4. Relationship between the homogeneous matrices and the coordinate frames.

$$H = \begin{bmatrix} R & T \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

One of the eigenvector and eigenvalue of R must be the rotation axis and 1, since, by definition, R is a rotation around axis $P_R \equiv [n_1\ n_2\ n_3]^T$, and obviously,

$$R\,P_r = P_r$$

thus P_r is an eigenvector (or principal vector) and its corresponding eigenvalue is 1. From (8) and by definition, it is obvious that specifying P_r and θ completely specifies R. Therefore, it is quite convenient to represent R by P_r scaled some function of θ. We use a modefied version of Rodrigues formula (see Junkins and Turner, 1986), and define P_r as

$$P_r = 2 \sin \frac{\theta}{2} [n_1 \quad n_2 \quad n_3]^T \qquad 0 \le \theta \le \pi \tag{9}$$

Besides the advantages associated with Quaternions or other vector representation of rotation matrix, one advantage for us is that some error formulas hold true for even non-infinitesimal perturbation. For example, Lemma V in Tsai and Lenz, 1987, regarding error analysis of Hand/Eye calibration, is exact. Also, the error formula in (29) is simpler. Another obvious advantage is that R is a simple function of P_r without any trigonometric functions

$$R = \left(1 - \frac{|P_r|^2}{2}\right) I + \frac{1}{2}\left(P_r P_r^T + \alpha \cdot Skew(P_r)\right) \tag{10}$$

where $\alpha = \sqrt{4 - |P_r|^2}$ and $Skew(P_r)$ is defined in (11.5). For the rest of the paper, the principal axis is defined as such, and all the computational procedure is given for P_r explicitly, and not for R.

1.2.4 Computational Procedures and Conditions for Uniqueness

We first give the computational procedures and conditions of uniqueness before we derive them. The derivations and proof follow from the eleven properties or lemmas in "Derivations of Computational Procedures and Conditions of Uniqueness using Eleven Lemmas". The minimum number of stations is three, where station means the location where robot pauses for doing camera extrinsic calibration. Using more than three stations improves the accuracy, as to be seen later.

Some Definition of Notation:

θ_R : angle of rotation for R

P_{gij} : principal axis or rotation axis for R_{gij} defined in (3), which is the 3D rotation from gripper coordinate frame G_i to G_j, as defined in (9). (11.1)

P_{cij} : rotation axis for R_{cij} in (4). (11.2)

P_{cg} : rotation axis for R_{cg} (11.3)

$$P_{cg}' : \ |P_{cg}'| = \frac{1}{2}\cos^{-1}\left(\frac{\theta_{R_{cg}}}{2}\right) P_{cg} = \frac{1}{\sqrt{4 - |P_{cg}|^2}} P_{cg} \tag{11.4}$$

$Skew(V)$: a skew-symmetric matrix generated by a 3D vector V such that

$$Skew(V) = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix} \tag{11.5}$$

N: number of stations described in "Basic Setup" on page 2.

Notice that the vectors defined above will also be used as a 3 × 1 column matrix. Also note that since P_{gij}, P_{cij} and P_{cg} are rotation axis with angle as its length, they completely specify R_{gij}, R_{cij} and R_{cg}. That is why for the procedures in the following, the formula for P_{cg} is given, not R_{cg}.

1.2.4.1 Procedure for Computing R_{cg}

Step 1: Compute P_{cg}'

For each pair of stations i,j, set up a system of linear equations with P_{ij}' as unknowns:

$$Skew(P_{gij} + P_{cij}) \cdot P_{cg}' = P_{cij} - P_{gij} \tag{12}$$

Since $Skew(P_{gij} + P_{cij})$ is always singular, it takes at least two pairs of stations to solve for a unique solution for P_{cg}' using linear least square technique.

Exception Handling: If $P_{gi1j1} + P_{ci1j1}$ is colinear with $P_{gi2j2} + P_{ci2j2}$ while P_{gi1j1} is not colinear with P_{gi2j2}, then the rotation angle of R_{cg} must be 180° and rotation axis the same as $P_{gi1j1} + P_{ci1j1}$.

Step 2: Compute $\theta_{R_{cg}}$

$$\theta_{R_{cg}} = 2 \tan^{-1} |P_{cg}'| \tag{13}$$

Note: Step 2 is not quite necessary since P_{cg} in Step 3 is sufficient to represent rotation. However, (13) may be handy.

Step 3: Compute P_{cg}

$$P_{cg} = \frac{2P_{gc}'}{\sqrt{1 + |P_{cg}|^2}} \tag{14}$$

1.2.4.2 Procedure for Computing T_{cg}

Given at least two pairs of stations i,j, set up a linear systems of three linear equations with T_{cg} as unknowns:

$$(R_{gij} - I)T_{cg} = R_{cg}T_{cij} - T_{gij} \tag{15}$$

For at least two pairs of stations, two sets of (15) are established and can be solved for the common unknowns T_{cg} using linear least square solutions.

1.2.5 Speed Performance

After the robot finishes the movement and grabbing the images, it takes only about $100 + 60N$ arithmetic operations to complete the computation. For a typical minicomputer, this only takes about 1/2 millisecond for ten stations. This complexity figure $(100+60N)$ can be derived as follows: The majority of computation is for solving the overdetermined linear least square solutions of (12) and (15). It takes about $3 \times N \times 3^2$ to form the normal equation of either one of (11.5) and (16), and $3^3 \times 2$ to solve the 3×3 normal equation. With a minimum of three stations and two interstation pairs, it takes about 1/10 of a millisecond. This is negligible comparing with the robot movement and image acquisition and analysis; at the pause of each movement, it takes about 90 millisecond to grab an image, extract all the 36 feature point coordinates with high accuracy, and compute the extrinsic camera parameters defined in (4).

1.2.6 Derivations of Computational Procedures and Conditions of Uniqueness using Eleven Lemmas

In order to outline the derivations of the computational procedures without going into actual details, eleven lemmas will first be stated and the significance of each explained. Selected sets of the key lemmas will be proved. Then the proof for the computational procedure for R_{cg} and T_{cg} will be given, followed by the conditions for uniqueness will be stated and proved.

Lemma I R_{gij} and R_{cij} differ by a unitary similarity transformation

$$R_{gij} = R_{cg} R_{cij} R_{cg}^T \tag{16}$$

Proof: This follows easily from the fact that $H_{cg}H_{gij}^{-1}$, H_{cg}^{-1}, H_{cij} in Figure 4 on page 3 form a closed loop and thus their product equals identity.

Significance: As a result, the eigenvector matrix of R_{gij} can be transformed from that of R_{cij} using R_{cg}.

Lemma II R_{cg} rotates the rotation axis of R_{cij} into that of R_{gij}, or

$$P_{gij} = R_{cg} P_{cij} \tag{17}$$

Proof: This follows from expanding R_{cij} and R_{gij} in (16) by their associated eigenvector and eigenvalue matrixes, and making use of the fact that P_{gij} and P_{cij} are the only and real eigenvectors of R_{gij} and R_{cij} respectively, and that the resultant rotation matrices on the left and right hand sides of (16) have common real eigenvector.

Significance: Since, from "The Observables or the Measurables" on page 3, P_{cij} and P_{gij} can be readily available from the observables H_{cc} and H_{gg}, (17) establishes constraints on R_{cg} in order to solve for it. Lemma II also says that if we regard all P_{cij} and P_{gij} as two clusters of vectors or points, then R_{cg} transforms one cluster into another.

Lemma III The rotation axis of R_{cg} is perpendicular to the vector joining the ends of the rotation axes for R_{cc} and R_{gg}, or

$$P_{cg} \perp (P_{gij} - P_{cij}) \tag{18}$$

Proof: This can be seen by observing Figure 5, but algebraically, here is the proof: (We will omit the subscript ij for clarity) The purpose is to show that $(P_g - P_c)^T P_c = 0$. By making use of (17) and the fact that $R_{cg}P_g = P_c$, we have

$$(P_g - P_c)^T P_{cg} = (P_g - P_c)^T R_{cg}^T R_{cg} P_{cg} = (R_{cg}R_g - P_g)^T P_{cg}$$

$$= [(R_{cg} - I)P_g]^T P_{cg} = P_g^T (R_{cg}^T - I)P_{cg} = 0$$

Significance: This implies that for a given pair of distinct P_{gi} and P_{gj}, P_{cg} is confined to be in the bisecting plane of P_{ci} and P_{cj}. With two such pairs, the direction of P_{cg} can be determined. In fact, (18) implies that P_{cg} can be determined up to a scale factor s from

$$P_{cg} = s(P_{gi_1j_1} - P_{ci_1j_1}) \times (P_{gi_2j_2} - P_{ci_2j_2}) \tag{19}$$

However, we will not use Lemma III in this manner. Instead, Lemma III is used to build up procedure for computing R_{cg} via Lemma IV,V and VI. The reason is that (19) is more error sensitive and has unnecessary degeneracies due to the fact that angle is not considered jointly.

Lemma IV $P_{gij} - P_{cij}$ is colinear with $(P_{gij} + P_{cij}) \times P_{cg}$

Proof: This follows from the fact that $P_{gij} - P_{cij}$ is simultaneously orthogonal to P_{cg} (according to Lemma III) and to $P_{gij} + P_{cij}$ (this latter property can be easily proved).

Significance This says that $P_{gij} - P_{cij} = s(P_{gij} + P_{cij}) \times P_{cg}$ for some scale factor s. Lemma V forces s to be 1. This lemma makes use of Lemma III, but the formula it generates is more accurate and robust than (18) (coming out of Lemma III) so far as computing P_{cg} is concerned, as to be seen.

Lemma V $P_{gij} - P_{cij}$ and $(P_{gij} + P_{cij}) \times P_{cg}'$ has the same length, where P_{cg}' was defined in (11.4).

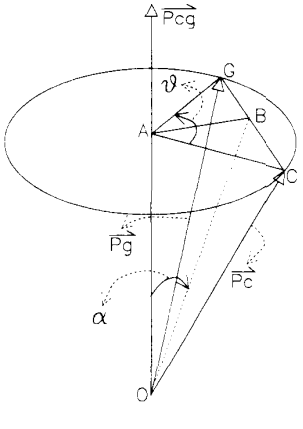

Figure 5. Geometrical relationships between P_{cg}, P_c and P_g. P_{cg} rotates P_c into P_g. The plane containing the circle is perpendicular to P_{cg}, and point B is the midpoint of point C and G.

Proof: Again, in the following proof and in Figure 5., we will drop the subscript ij for P_{gij} and P_{cij}. Let the angle between the vectors represented by P_{cg}' and $P_c + P_g$ be α. Then by definition,

$$|(P_g + P_c) \times P_{cg}'| = |P_g + P_c| \; |P_{cg}'| \sin\alpha$$

Substituting (9) and (11.4) into the above gives

$$|(P_g + P_c) \times P_{cg}'| = |P_g + P_c| 2 \sin\frac{\theta}{2} \left(4 - 4\sin^2\frac{\theta}{2}\right)^{-1/2} \sin\alpha$$

$$= |P_g + P_c| \tan\frac{\theta}{2} \sin\alpha = 2|\overline{OB}| \sin\alpha \tan\frac{\theta}{2}$$

$$= 2|AB| \tan\frac{\theta}{2} = 2|CB| = |\overline{CG}| = |P_g - P_c|$$

In the above derivations, A, B, C, G, O are points in Figure 5, and $|OB|$ means length of the vector extending from point O to point B, etc.. Also, in the above derivations, several geometric and trigonometric relationships in Figure 5 are used. One is $|P_c + P_g| = 2|OB|$. Another is $|AB| = |OB| \sin\alpha$. Still another is $|CB| = |AB| \tan\theta/2$. These properties follow easily from Figure 5. Thus we have shown that

$$|(P_g + P_c) \times P_{cg}'| = |P_g - P_c|$$

Significance Given Lemma IV and V, Lemma VI is readily derived, which easily leads to the computational procedure for P_{cg} in (12).

Lemma VI $(P_{gij} + P_{cij}) \times P_{cg}' = P_{cij} - P_{gij}$ (20)

Proof: This is a direct consequence of Lemma IV and Lemma V.

Significance Although (18) provides a constraint on the direction of P_{cg} for any pair of stations ij, (20) provides a stronger constraint since it constrains $\theta_{R_{cg}}$ as well as P_{cg}.

Lemma VII $Skew(P_{gij} + P_{cij})$ is singular and has rank 2.

Significance $Skew(P_{gij} + P_{cij})$ is the coefficient matrix for the systems of linear equation in (12) used to solve for P_{cg}'. Therefore Lemma VII implies that it is impossible to compute R_{cg} with only two stations.

Lemma VIII $(R_{gg} - I)T_{cg} = R_{cg}T_{cc} - T_{gg}$ (21)

Proof: This follows from the same derivation as Lemma I.

Significance Lemma VIII establishes the equation in (15) used to solve for T_{cg}.

Lemma IX $R_{cc} - I$ is singular and has rank 2.

Significance $R_{cc} - I$ is the coefficient matrix for the systems of linear equation in (15) used to solve for T_{cg}. Therefore Lemma IX implies that it is impossible to compute T_{cg} with only two stations.

Lemma X If $\theta_{R_{cg}} \neq \pi$ 2, or equivalently, $|P_{cg}| \neq \pm 2$, then

$$\begin{bmatrix} Skew(P_{gi_1j_1} + P_{ci_1j_1}) \\ Skew(P_{gi_2j_2} + P_{ci_2j_2}) \end{bmatrix} \tag{22}$$

has full column rank if and only if $P_{gi_1j_1}$ and $P_{gi_2j_2}$ have different directions (or equivalently, $P_{ci_1j_1}$ and $P_{ci_2j_2}$ have different directions).

Significance (22) is just the compound matrix of two $Skew(P_{gij} + P_{cij})$ in Lemma VII, and therefore is the coefficient matrix for solving R_{cg} given two pairs of P_{gij} and P_{cij}. Thus Lemma X ensures that given a minimum of three stations, the solution for R_{cg} is unique.

Lemma XI $\begin{bmatrix} R_{cc_1} - I \\ R_{cc_2} - I \end{bmatrix}$ (23)

has full column rank if and only if P_{cc_1} and P_{cc_2} have different directions (or equivalently, P_{gc_1} and P_{gc_2} have different directions).

Significance (23) is just the compound matrix of two $R_{cc} - I$ in Lemma IX, and therefore is the coefficient matrix for solving T_{cg} given two pairs of P_{gc} and P_{cc}. Thus Lemma XI ensures that given a minimum of three stations, the solution for T_{cg} is unique.

Proof of the Computational Procedure for R_{cg} in (12),(13) and (14)

(12) follows from Lemma VI by considering the fact that for any two 3×1 vectors a and b,

$$a \times b = Skew(a) \cdot b \tag{24}$$

where a and b on the left denotes vectors while a and b on the right are 3×1 column matrices. (13) and (14) simply follow from the definitions of P_{cg}' in (11.4).

Proof of the Computational Procedure for T_{cg} in (15)

This follows simply from Lemma VIII.

Minimum number of Stations: Three

This follows from Lemma VII, IX, X and XI. Equivalently, the minimum number of pairs of stations needed is two.

Conditions of Uniqueness

For a minimum of three stations (or two pairs of stations), the necessary and sufficient conditions for a unique solution for R_{cg} and T_{cg} is that the inter-station rotation axes are not colinear for different pairs of stations.

Proof: This follows from Lemma VIII and X. Note that when the sum of rotation axes ($P_{gij} + P_{cij}$) are colinear while P_{gij} is different for different inter-station rotations, then the solution is still unique except that (12) cannot be used. In this case, $angle(R_{cg})$ is simply 180° and rotation axis is the same as $P_{cij} + P_{gij}$.

1.3 Accuracy Issues

In Tsai and Lenz, 1987A, some accuracy analysis is given. In the following, we give a list of critical factors dominating the error, and describe steps for improving the accuracy. These results are supported by the experimental findings to be described later.

1.3.1 Critical Factors Affecting the Accuracy and Steps in Improving Accuracy

Let $\sigma(R)$ denote the root mean square error (RMSE) of P_R and $\sigma(T)$ denote the RMSE or the translation vector T. By observing the accuracy formulas for R_{cg} and T_{cg} in Tsai and lenz, 1987, five observations can be made. The proof follows from the accuracy formula. In the following, we merely list these five observations and give intuitive reasoning whenever appropriate. Thse observations give the background for the list of steps to improve accuracy, and are tested during the experimentations.

Observation I: *The RMS error of rotation from gripper to camera $\sigma_{R_{cg}}$ is inversely proportional to the sine of the angle between the inter-station rotation axes.*

This is reasonable since, from Lemma II, R_{cg} rotates P_{cij} into P_{gij}. With a minimum of two pairs of ij's, (17) is used to solve for R_{cg}. When $\angle(P_{g12}, P_{g23})$ becomes smaller, P_{g12} becomes closer to P_{g23}, making (17) for each ij more similar to each other, thus causing the equation to be closer to singularity. Alternatively, one can see that the coefficient matrix for solving P_{cg}' (see Lemma VII) becomes singular as $P_{g_{1}j_{1}}$ approaches $P_{g_{2}j_{2}}$ and $P_{c_{1}j_{1}}$ approaches $P_{c_{2}j_{2}}$. In fact, it can be shown that the row vectors of the coefficient matrix in Lemma X lies in two planes, with $P_{g_{1}j_{1}} + P_{c_{1}j_{1}}$ and $P_{g_{2}j_{2}} + P_{c_{2}j_{2}}$ being the normal vectors of the two planes. Thus the greater the difference is between $P_{c_{1}j_{1}}$ and $P_{c_{2}j_{2}}$, the closer the two planes are to being orthogonal, making the coefficient matrix more linearly independent.

Observation 2: *The rotation and translation error are both inversely proportional to the inter-station rotation angle. That is, $\sigma_{R_{cg}} \propto^{-1} \theta_{R_{gij}}$ and $\sigma_{T_{cg}} \propto^{-1} \theta_{R_{gij}}$*

This is reasonable since R_{cg} is determined solely from P_{cij} and P_{gij}, and the greater $\theta_{R_{cij}}$ and $\theta_{R_{gij}}$ are, the smaller the effect of a small perturbation (with given size σ_{R12}, σ_{Rg12}) is on the result.

Observation 3: *The distance between the camera lens center and the calibration block has a dominant effect on the translation error.*

This is reasonable since the greater this distance is, the greater the effect of angular error for camera extrinsic calibration has on the position of the camera relative to the calibration block, which in turns influence the accuracy of the translation from camera to hand.

Observation 4: *The distance between the robot gripper coordinate centers at different stations is also is a critical factor in forming the error of translation. But the distances between different camera stations are not important.*

The situation is similar to that described in Observation 3. Notice this distance is not the distance between gripper tips at different stations. It is the amount of movement of the *robot gripper coordinate center.*

Observation 5: *The error of rotation is linearly proportional to the error of orientation of each station relative to the base. The error of translation is approximately linearly proportional to this error of orientation unless the error of robot translational positioning accuracy is big.*

It is convenient to define two types of critical factors. One is first degree, and the other second degree. The first degree factor is more dominant in most cases, but sometimes, some second degree factor (σ_{Tgi}) can be so bad such that it becomes dominant.

First Degree Critical Factors:

a. Angle between different inter-station rotation axes (e.g., $\angle(P_{g12}, P_{g23})$)

   Note: $\angle(P_{g12}, P_{g23}) = \angle(P_{c12}, P_{c23})$

b. Rotation angle of inter-station rotation $\theta_{R_{gij}} (= \theta_{R_{cij}})$

c. Distance between camera lens center and calibration block $|T_{ci}|$, and distance between robot arm coordinate centers at different stations $|T_{g1} - T_{g2}|$

d. Error of rotation of each station relative to base σ_{Rgi}, σ_{Rci}, or error of inter-station rotation $\sigma_{R_{gij}}$, $\sigma_{R_{cij}}$

Second Degree Critical Factors:

1. Error of translation of each station relative to base σ_{Tci}, σ_{Tgi}

1.3.2 Steps to Improve Accuracy

1. Adopt the setup that will achieve maximum angles between different inter-station rotation axes, no matter how many stations are used. One example is given in Tsai and Lenz, 1987A.
2. Maximize the rotation angle for inter-station rotations. This again can be done using the setup mentioned earlier.
3. Minimize the distance between camera lens center and calibration block. This requires a small calibration block and suitable optics for short range viewing.
4. Minimize the distance between robot arm coordinate centers at different stations. This requires some planning and is robot dependent.
5. Use redundant stations. Since the extrinsic calibration plus feature extraction can be done within 90 msec when 36 points are used, using more frames poses no problem. The error due to non-systematic sources will be reduced by a factor of $\sqrt{N}$ where N is the number of stations. (See results in "Simulation and Real Experiment Results" on page 7). However, the other criteria described above should not be violated when increasing station numbers.
6. Use camera calibration algorithm setup that yields high accuracy to improve error on translation and rotation of each individual station.
7. Try to precalibrate the robot itself so that the position and orientation of each station is known more accurately. If this is difficult, then at least try to make inter-station translation and rotation more accurate, if possible. That is, the robot system may not be able to tell the user the absolute location and orientation of its gripper coordinate frame, it may be able to better tell the amount of relative movement from station to station.

1.4 Simulation and Real Experiment Results

1.4.1 Simulation Experiments

In order to simulate the actual physical setup, all the setup parameters are selected to be almost the same as those used in the real experiments to be described, except that due to the x axis problem with our robot (to be described later), the station generation process used in the real experiment is modified. The station generation process used in the simulation is described in Tsai and Lenz, 1987A. In the following, the setup parameters are set as follows: $|T_c| = 6.65\ inch$, $|T_{cg}| = 9.5\ inch$, $N = 3$, $\sigma_{Tci} = 3\ mil$, $\sigma_{Tgi} = 5\ mil$, $\sigma_{Rci} = \sigma_{Rgi} = 1.5\ milliradian$, $\angle(P_{g12}, P_{g23}) = \pi/3$. In the following, we show the simulation results of the effect of number of stations on the error of rotation R_{cg} and translation T_{cg}. The simulations on the effect of four other critical parameters are described in Tsai and Lenz, 1987A, and agree surprisingly well with the five observations made earlier. 1000 tests are done, and statistics are gathered. The results are shown in figures.

Effect of Number of Inter-Station Pairs on Accuracy

Figure 6 and Figure 7 shows the error of translation and rotation as a function of the inverse of square root of the number of inter-station pairs. The solid line shows the RMS error, while the dashed line is the maximum error out of one thousand tests. As expected, the RMS error increases linearly as the inverse of $\sqrt{N}$. Since the proposed technique is quite efficient, and the station pose planning and robot motion are automatic, increasing the number of station is quite feasible, and pays off well.

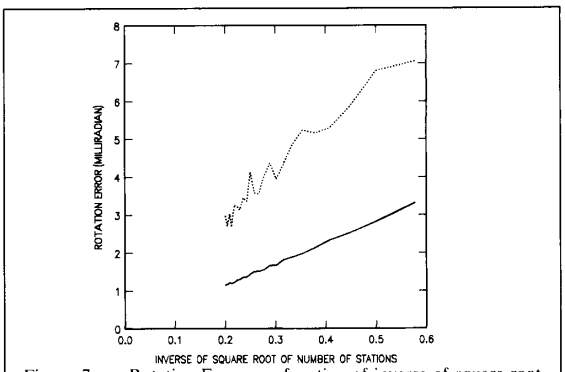

Figure 7. Rotation Error as a function of inverse of square root of the number of inter-station pairs. The solid line is RMSE and dashed line is maximum error.

1.4.2 Real Experiments

1.4.2.1 Setup Description

Figure 2 on page 2 shows the physical setup we used. A Javelin CCD 480 x 388 camera is fastened to the last joint of an IBM Clean Room Robot (CRR). The CRR has two manipulators, each with seven degrees of freedom (including gripper opening). We only use one of the manipulators. The CRR is an electric box frame Cartesian robot. There are three linear joints (x, y, z) and three rotary joints (roll, pitch and yaw) for each manipulator. The work volume is about 6 feet by 4 feet by 2 feet and the repeatability for linear joint is about 4 mil, and that for the rotary joints 1 milliradian. The accuracy is calibrated to a limited extent. The scale and offset for each rotary joint is calibrated to 3 milliradian accuracy. The rotation axes for the three rotary joints are supposed to be intersecting at the same point (origin of RW coordinate frame), but we did not calibrate that. The x axis has some problems: For our robot, the z beam sags, causing the movement in x axis to be like a pendulum. This effect is not fully calibrated yet, but we suspect that it generates about 20 mils translation and 15 millirad rotation within a work range of 15 inches. Due to this problem, we are forced to modify the station generation procedure used in the simulation in order to avoid using x axis. Either with or without moving x axis, the station placement and manipulator motion planning is automatic, and the number of stations can be arbitrary without manual intervention.

The calibration block is a clear glass plate with the center 1 inch by 1 inch area filled with 36 black disks printed on it using step-and-repeat photographic emulsion (See Figure 8 on page 8). The disks are 5000 microns apart with 2000 micron radius (accurate to 1 micron). The calibration is back lighted and sits in the middle of the work space.

1.4.2.2 Accuracy Assessment

The accuracy of our hand/eye calibration results is assessed by how accurately we can predict the placement of camera in 3D world with any arbitrary manipulator movement. As was indicated in "Why is Hand/Eye Calibration Important?" on page 1, one of the main reasons why robot hand/eye calibration is important is that robot needs to know not only where the gripper is, but also where the camera is in the work space, so that the measurement taken by vision can be related to the robot. Being able to determine where the camera is in the work space for an arbitrary manipulator movement is thus the primary goal. This is tested in the following steps:

Step 1: Move the manipulator to 2N different positions where N is greater than 2. For each station i, compute camera to calibration block homogenous transformation H_{ci} using extrinsic calibration. This takes about 90 msec per station. The robot gripper position and orientation relative to robot world, which is H_{gi}, is also recorded.

Step 2: Compute H_{cg} using procedures in "Computational Procedures and Conditions for Uniqueness" on page 4, using data from station 1 through N.

Step 3: For each station k (k from 1 to N), compute homogenous matrix H_{RC} (homogenous transformation from robot world frame RW to calibration block world frame CW) by:

$$H_{RC} = H_{gi}^{-1} H_{cg}^{-1} H_{ci}^{-1}$$

Make an average of H_{RC} computed from these N stations.

Step 4: Let stations N+1 through 2N be called verification stations. For each of the verification station, predict the position and orientation of the camera relative to robot world base coordinate RW by $H_{cg}^{-1} H_{gk}^{-1}$ where k is the station index, and H_{gk} is computed from robot joint coordinates (see "What Are the Observables and What is to be computed?" on page 3). Compare this predicted position and orientation with $H_{ck}^{-1} H_{RC}$, where H_{ck} is computed in step 1 while H_{RC} is computed in step 3.

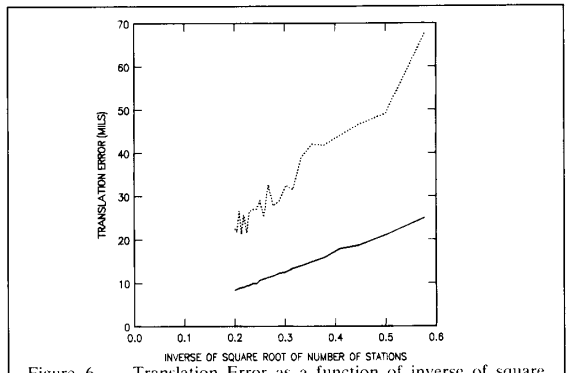

Figure 6. Translation Error as a function of inverse of square root of the number of inter-station pairs. The solid line is RMSE and dashed line is maximum error.

Figure 8. Calibration block is a clear glass plate with 36 disks printed on it using photographic emulsion. The accuracy is 1 micron.

The results of a series of experiments yield the following table:

	New Camera Pose Prediction Error	
N	Rotation error	Translation error
4	4.568 mrad	23.238 mil
6	3.304 mrad	19.078mil
8	3.264 mrad	26.712 mil
10	2.888 mrad	14.642 mil
12	2.782 mrad	12.516 mil

Since there is no absolute H_{cg} ground truth to compare with, the accuracy has to be assessed as the error of new camera pose prediction, as described earlier. The effect of N is indeed very significant. We have a program that automatically plans the movement of manipulator for an arbitrary number of stations, and since the algorithm proposed in this paper is quite efficient, increasing the number of frames is quite easy. Also observe that the error of predicted camera pose includes both the error of the calibrated hand/eye relationship and the robot's positioning error. Notice from the table that for 10 stations, the translation error is about 14 mil. But the robot's positioning accuracy is worse than 10 mil. This means that the eye-to-hand relationship is calibrated to better than 10 mil. Using the error formula in (32) of Tsai and Lenz, 1987A, scaled by $\sqrt{10/3}$, the error of T_{cg} is predicted to be 10.66 mil, agreeing well with the real experiment data. The rotation error is about 2.88 millirad. Notice that the error of rotary joint is about 2.5 millirad. Therefore, the actual error of R_{cg} should also be of this order of magnitude. This agrees very well with the prediction by (28) of Tsai and Lenz, 1987A, which gives 2.557 millirad. Notice that the error in the table is not strictly monotonic with respect to the inverse of the square root of number of stations. This is due to the fact that the simulation curves presented earlier were averaged over 1000 tests, while here, for each N, there is only one test. Also, since the robot error itself gets into it, it is more unpredictable, while the simulation curves only shows the H_{cg} error. Nevertheless, the error generally decreases nicely as the number of stations increases.

1.5 Conclusion

This paper introduced a high speed, high accuracy, versatile, simple and fully autonomous technique for 3D robotics hand/eye calibration. It is ideally suited for machine vision people who do not have special equipment or skill in robot and would like to calibrate the hand/eye relationship using the same kind of setup, processing equipment and similar algorithms as those for his familiar camera calibration. It is high speed since it takes only about 100+64N arithmetic operations to compute the hand/eye relationship after the robot finishes the movement, and incurs only an additional 64 arithmetic scalar operations for each additional station. This makes the current algorithm the fastest one compared with the state of the art. The speed performance is especially attractive to those applications where the hand/eye configuration needs be changed frequently. For example, the robot may pick up the camera to perform some task, and then put it right back to a holder. Since the grasping cannot be precise, hand/eye calibration must be performed frequently. It is also important to those tasks where and hand/eye relationships need be changed frequently due to different task requirements. As for the accuracy, no other reported hand/eye calibration technique does any better. Our results would be improved when more stations (bigger N) are used, that is, the robot makes more moves. The results in our real experiments could be further improved if we change the optics and the size of calibration block, as well as the mounting position, so that all of the critical factors described in the accuracy analysis section are taken into considerations.

1.6 References

1. Bowman M. & Forrest A.,1987, Robot Model Optimization, submitted for publication.
2. Izaguirre A., Summers J., Pu P., 1987, A New Development in Camera Calibration, Calibrating a Pair of Mobil TV Cameras, to appear in *International Journal of Robotics Research*.
3. Jenkins J. L., and Jurner J. D., 1986, *Optimal Spacecraft Rotational Manervers*, Elsevier.
4. Lenz, R. and Tsai, R., 1987, Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3D Machine Vision Metrology, *Proceedings of IEEE International Conference on Robotic and Automation*, Raleigh, NC.
5. Puskorius, G. and Feldkamp, L., 1987, Calibration of Robot Vision, *Proceedings of IEEE International Conference on Robotic and Automation*, Raleigh, NC.
6. Rogers, D. F. and Adams J. A., 1976, *Mathematical Elements for Computer Graphics*, New York, McGraw-Hill.
7. Shiu Y. and Ahmad S, 1987, *Proceedings of IEEE International Conference on Robotic and Automation*, Raleigh, NC.
8. Tsai, R., 1987, A Versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology using Off-the-Shelf TV Cameras and Lenses, to appear in IEEE Journal of Robotics and Automation, also best paper award for 1986 IEEE International Conference on Computer Visionand Pattern Recognition, Miami, Florida.
9. Tsai, R., and Lenz, R., 1987A, A New Technique for Autonomous and Efficient 3D Robotics Hand/Eye Calibration, *Proceedings of the Fourth International Symposium of Robotics Research*, Santa Cruz, CA, August 9-14.
10. Tsai, R., and Lenz, R., 1987B, Review of the Two-Stage Camera Calibration Technique plus some New Implementation Tips and New Techniques for Center and Scale Calibration, *Second Topical Meeting on Machine Vision*, Optical Society of America, Lake Tahoe.
11. Tsai, R.Y. and Lavin, M., 1984, Three-Dimensional Mechanical Part Measurement using a Vision/Robot System, *IBM Research Report RC 10506*, May 8.