Accurate linear technique for camera calibration considering lens distortion by solving an eigenvalue problem

Sheng-Wen Shih
Academia Sinica
Institute of Information Science
Nankang, Taipei, Taiwan
and
National Taiwan University
Institute of Electrical Engineering
Taipei, Taiwan

Yi-Ping Hung, MEMBER SPIE
Academia Sinica
Institute of Information Science
Nankang, Taipei, Taiwan

Wei-Song Lin
National Taiwan University
Institute of Electrical Engineering
Taipei, Taiwan

Abstract. A new technique for calibrating a camera with high accuracy and low computational cost is proposed. The geometric camera parameters considered include camera position, orientation, focal length, radial lens distortion, pixel size, and optical axis piercing point. With this method, the camera parameters to be estimated are divided into two parts: the radial lens distortion coefficient κ and a composite parameter vector **c** composed of all the above-mentioned geometric camera parameters other than κ. Instead of using nonlinear optimization techniques, the estimation of κ is transformed into an eigenvalue problem of an 8×8 matrix. The method is fast because it requires only linear computation; it is accurate because the effect of the lens distortion is considered and because all the information contained in the calibration points is used. Computer simulations and real experiments show that the calibration method can achieve an accuracy of 1 part in 10,000 in 3-D measurement, which is better than that of the well-known method proposed by R. Y. Tsai.

Subject terms: camera calibration; lens distortion; camera model; stereo vision; eigenvalue problem; three-dimensional metrology.

Optical Engineering 32(1), 138–149 (January 1993).

1 Introduction

1.1 *Importance of Camera Calibration*

Camera calibration in the context of 3-D machine vision, as defined by Tsai,[1] is the process of determining the internal camera geometric and optical characteristics (intrinsic parameters) and/or the 3-D position and orientation of the camera frame relative to a certain world coordinate system (extrinsic parameters). To infer 3-D objects using two or more images, it is essential to know the relationship between the 2-D image coordinate system and the 3-D object coordinate system. This relationship can be described by the following two transformations:

1 Perspective projection of a 3-D object point onto a 2-D image point: Given an estimate of a 3-D object point and its error covariance, we can predict its projection (mean and covariance) on the 2-D image. This is useful for reducing the searching space in matching features between two images or for hypothesis verification in scene analysis.

2. Back projection of a 2-D image point to a 3-D ray: Given a 2-D image point, there is a ray in the 3-D space within which the corresponding 3-D object point must lie on. If we have two (or more) views available, an estimate of the 3-D point location can be obtained by using triangulation. This is useful for inferring 3-D information from 2-D image features.

For applications that require these two transformations, e.g., automatic assembling, 3-D metrology, robot calibration, tracking, trajectory analysis, and vehicle guidance, it is essential to calibrate the camera either on-line or off-line. In many cases, the overall system performance strongly depends on the accuracy of camera calibration. In general, the calibration method that achieves higher accuracy requires more computations, and the trade-offs between accuracy and computational cost depends on the requirements of the application. Our objective is to develop a camera calibration method that is accurate enough to meet most application requirements while maintaining a computational cost low enough to fulfill the real-time requirement.

1.2 *Existing Techniques for Camera Calibration*

Many techniques have been developed for camera calibration because of the strong demand of applications. These techniques can be classified into two categories: one that considers lens distortion,[1–5] and one that neglects lens distortion.[6–9]

A typical linear technique that does not consider lens distortion is the one that estimates the perspective transformation matrix[8] **H**. The estimated **H** can be used directly for forward and backward 3-D to 2-D projection. If necessary, given the estimated **H**, the geometric (or physical) camera parameters β can be easily determined.[7,10,11] The precise definition of β is given in Sec. 2. For those techniques that

Paper AO-023 received Jan. 31, 1992; revised manuscript received June 7, 1992; accepted for publication June 28, 1992. This paper is a revision of a paper presented at the SPIE conference on Optics, Illumination, and Image Sensing for Machine Vision VI, November 1991, Boston, Mass. The paper presented there appears (unrefereed) in SPIE Proceedings Vol. 1614.

estimate the perspective transformation matrix, note that the homogeneous projection matrix **H** is not unique. The projection matrix is undetermined up to a scale factor, thus one constraint must be imposed to ensure a unique solution. Usually, the constraint is to set one of the entries of **H** to a constant. Faugeras and Toscani[6] showed that by imposing a physical constraint, the estimation problem remains linear.

Grosky and Tamburino[5] regard the camera calibration as having two independent phases; the first is to remove geometric camera distortion so that rectangular calibration grids are straightened in the image plane, and the second is to use a calibration method similar to the described linear one. Grosky and Tamburino[5] have found some physical constraints and try to unify the approach to the linear camera calibration problem; both coplanar and noncoplanar calibration points can be used, however, the former case requires three user-supplied constraints. Although the lens distortion is assumed compensated at phase 1, the phase-2 calibration method also can be applied directly to an uncompensated image.

Martins et al.[12] developed a two-plane method that does not use an explicit camera model. The back-projection problem is solved by computing the vector that passes through the interpolated points on each calibration plane. However, the projection problem is not solved in their paper.

Faig's method[2] is a good example of those that consider lens distortion. For methods of this type, equations are established that relate the camera parameters to the 3-D object coordinates and 2-D image coordinates of the calibration points. Nonlinear optimization techniques are then used to search for the camera parameters with an objective to minimize residual errors of these equations. Disadvantages of this kind of method include the fact that a good initial guess is required to start the nonlinear search and that it is computationally expensive.

A few years ago, Tsai proposed an efficient two-stage technique using the "radial alignment constraint." [1] His method involves a direct solution for most of the calibration parameters and some iterative solutions for the remaining parameters. Some drawbacks of Tsai's method have been mentioned by Weng et al.[3] Our experience has also shown[13] that Tsai's method can be worse than the simple linear method[8] if lens distortion is relatively small.

Recently, Weng et al.[3] presented some experimental results using a two-step method. The first step involves a closed-form solution based on a distortion-free camera model, and the second step improves the camera parameters estimated in the first step by taking the lens distortion into account. This method solves the initial guess problem in the nonlinear optimization, and is more accurate than Tsai's method, according to our experiments.

In this paper, we develop a camera calibration method which is fast and accurate. With this method, the camera parameters to be estimated are divided into two parts: the radial lens distortion coefficient κ and a composite parameter vector $\mathbf{c}$ composed of all the geometric camera parameters other than κ. Instead of using nonlinear optimization techniques, the estimation of κ is transformed into an eigenvalue problem of an 8×8 matrix. Our method is fast because it requires only linear computation. It is accurate because the effect of the lens distortion is considered and because all the information contained in the calibration points

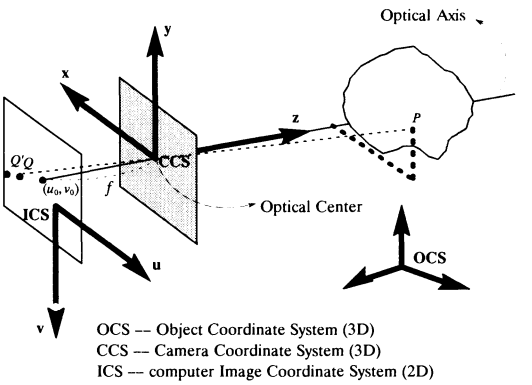


Fig. 1 Pinhole camera model with lens distortion, where P is a 3-D object point and Q and Q' are its undistorted and distorted image points, respectively.

is used. Section 2 introduces the camera model adopted in this paper. Section 3 describes the details of the proposed camera calibration method. Results of computer simulations and real experiments shown in Sec. 4 demonstrate the performance of our method compared to others.

2 Camera Model

Consider the pinhole camera model with lens distortion, as shown in Fig. 1. Let P be an object point in the 3-D space, and $\mathbf{r}_O = [x_O\ y_O\ z_O]^t$ be its coordinates, in millimeters, with respect to a fixed object coordinate system (OCS). Let the camera coordinate system (CCS), also in millimeters, have its x-y plane parallel to the image plane (such that the x axis is parallel with the horizontal direction of the image and the y axis is parallel with the vertical one), with its origin located at the optical center and its z axis aligned with the optical axis of the lens (see Fig. 1). Let $\mathbf{r}_C = [x_C\ y_C\ z_C]^t$ be the coordinates of the 3-D point P with respect to the CCS. If there is no lens distortion, the corresponding image point of P on the image plane would be Q (see Fig. 1). However, because of the effect of lens distortion, the actual image point is Q'. Let $\mathbf{s}_I = [u_I\ v_I]^t$ denote the 2-D coordinates (in pixels) with respect to the computer image coordinate system (ICS) of the actual image point Q', where the origin of ICS is located at the center of the frame memory coordinate [e.g., the origin of the ICS is right at (256,256) for a 512 by 512 image].

As shown in Fig. 2, the 3-D to 2-D transformation from $\mathbf{r}_O$ to $\mathbf{s}_I$ can be divided into the four steps described next.

2.1 *Rotation and Translation from the OCS to the CCS*

The transformation from $\mathbf{r}_0$ to $\mathbf{r}_C$ can be expressed as

$$\begin{cases} x_C = r_1 x_O + r_2 y_O + r_3 z_O + t_1 \\ y_C = r_4 x_O + r_5 y_O + r_6 z_O + t_2 \\ z_C = r_7 x_O + r_8 y_O + r_9 z_O + t_3 \end{cases}, \tag{1}$$

or $\tilde{\mathbf{r}}_C = \mathbf{T}_C^O \tilde{\mathbf{r}}_O$ with

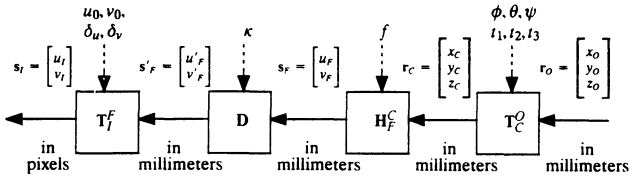


Fig. 2 Relation between different transformation matrices.

$$\mathbf{T}_C^O = \begin{bmatrix} \mathbf{R}_C^O & \mathbf{t}_C^O \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 & t_1 \\ r_4 & r_5 & r_6 & t_2 \\ r_7 & r_8 & r_9 & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} , \tag{2}$$

where tilde (~) denotes homogeneous[14] coordinates, $\mathbf{t}_C^O = (t_1\ t_2\ t_3)^t$ is a translation vector, and $\mathbf{R}_C^O$ is a 3×3 rotation matrix determined by the three Euler angles, ϕ, θ, ψ, rotating about the z, y, and z axes sequentially.

2.2 *Perspective Projection from a 3-D Object Point in the CCS to a 2-D Image Point on the Image Plane*

Let f be the effective focal length, and let $\mathbf{s}_F = [u_F\ v_F]^t$ be the 2-D coordinates (in millimeters) of the undistorted image point Q lying on the image plane. Then, we have

$$u_F = f\frac{x_C}{z_C} , \quad v_F = f\frac{y_C}{z_C} . \tag{3}$$

Also, we can express this perspective projection in the homogeneous coordinates as

$$\tilde{\mathbf{s}}_F = \mathbf{H}_F^C \tilde{\mathbf{r}}_C \quad \text{with} \quad \mathbf{H}_F^C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} . \tag{4}$$

2.3 *Lens Distortion from Q to Q'*

For practical reasons, we consider only the first term of the radial lens distortion, i.e.,

$$\begin{cases} u_F = (1 - \kappa\rho^2)u'_F \\ v_F = (1 - \kappa\rho^2)v'_F \end{cases} \quad \text{where} \quad \rho^2 = u'^2_F + v'^2_F , \tag{5}$$

$$\text{or} \quad \mathbf{s}_F = (1 - \kappa\|\mathbf{s}'_F\|^2)\mathbf{s}'_F , \tag{6}$$

where $\mathbf{s}'_F = (u'_F\ v'_F)^t$ are the coordinates of the distorted 2-D image (in millimeters). In this paper, κ is given in inverse square millimeters.

2.4 *Scaling and Translation of 2-D Image Coordinates*

The transformation from $\mathbf{s}'_F$ (in millimeters) to $\mathbf{s}_I$ (in pixels) involves (1) scaling from millimeters to pixels and (2) translation due to the misalignment of the sensor array with the optical axis of the lens. Hence,

$$\begin{cases} u'_F = (u_I - u_0)\delta_u \\ v'_F = (v_I - v_0)\delta_v \end{cases} , \tag{7}$$

$$\text{or} \quad \tilde{\mathbf{s}}_I = \mathbf{T}_I^F \tilde{\mathbf{s}}'_F \quad \text{with} \quad \mathbf{T}_I^F = \begin{bmatrix} 1/\delta_u & 0 & u_0 \\ 0 & 1/\delta_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} , \tag{8}$$

where δ_u and δ_v are the horizontal and vertical pixel spacing (millimeters/pixel) and u_0 and v_0 are the coordinates (in pixels) of the origin of the CCS in the computer image coordinate system.

Using this notation for camera parameters, the geometric camera parameters $\beta = [t_1\ t_2\ t_3\ \phi\ \theta\ \psi\ f\ \kappa\ \delta_u\ u_0\ v_0]^t$. The vertical scaling factor δ_v is not included here because it is a known parameter when we use a solid state camera—otherwise, only the ratios f/δ_u and f/δ_v can be determined. Combining Eqs. (1), (3), (5), and (7), we have

$$(1 - \kappa\rho^2)(u_I - u_0)\delta_u = f\frac{x_O r_1 + y_O r_2 + z_O r_3 + t_1}{x_O r_7 + y_O r_8 + z_O r_9 + t_3} , \tag{9}$$

$$(1 - \kappa\rho^2)(v_I - v_0)\delta_v = f\frac{x_O r_4 + y_O r_5 + z_O r_6 + t_2}{x_O r_7 + y_O r_8 + z_O r_9 + t_3} , \tag{10}$$

where $\rho = [\delta_u^2(u_I - u_0)^2 + \delta_v^2(v_I - v_0)^2]^{1/2}$.

3 New Camera Calibration Technique

Given a set of 3-D calibration points and their corresponding 2-D image coordinates, the problem is to estimate β, the parameters of the camera model. Instead of estimating β directly, we first estimate the coefficient κ and the composite parameters $\mathbf{c}$ [as described following Eq. (16)], then decompose them into β. Two camera calibration algorithms are described in this paper; one requires a set of noncoplanar calibration points, and the other only needs coplanar calibration points. Although the former requires noncoplanar calibration points, it is more accurate than the latter.

Our method requires initial guess for u_0, v_0, and δ_u, which can be easily set as follows. Let f_{camera} denote the pixel scanning rate of the camera (e.g., $f_{\text{camera}} = 14.31818$ MHz for PULNiX TM-745E), and $f_{\text{digitizer}}$ denote the pixel scanning rate of the digitizer or frame grabber (e.g., $f_{\text{digitizer}} = 10$ MHz for ITI Series 151). Let δ'_u denote the horizontal pixel spacing of the solid state imager (e.g., $\delta'_u = 11$ μm for PULNiX TM-745E). Then a good estimate of δ_u can be obtained by the following equation:

$$\delta_u = \delta'_u \frac{f_{\text{camera}}}{f_{\text{digitizer}}} . \tag{11}$$

The other two parameters $(\hat{u}_0, \hat{v}_0)$ can be temporally set[1] to (0,0) if no other reliable information about them is available. More accurate estimates of the three parameters can be obtained with our calibration procedure, which can then be used as a better initial guess iteratively. If the amount of radial lens distortion is not unreasonably large, say $|\kappa| \leq 0.0008\ \text{mm}^{-2}$, and the computational speed is not the main concern, then another two or three iterations can give even higher accuracy according to our experience (see Fig. 9 in Sec. 4.1).

Hereafter, for simplicity, we will use u, v, x, y, and z to denote u_I, v_I, x_O, y_O, and z_O, respectively.

3.1 *New Camera Calibration Technique Using Noncoplanar Calibration Points*

Rewrite Eqs. (9) and (10) as

$$(1-\kappa\rho^2)(u-u_0)[x\ y\ z\ 1]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} = [x\ y\ z\ 1]\begin{bmatrix} r_1 f/\delta_u \\ r_2 f/\delta_u \\ r_3 f/\delta_u \\ t_1 f/\delta_u \end{bmatrix}, \tag{12}$$

$$(1-\kappa\rho^2)(v-v_0)[x\ y\ z\ 1]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} = [x\ y\ z\ 1]\begin{bmatrix} r_4 f/\delta_v \\ r_5 f/\delta_v \\ r_6 f/\delta_v \\ t_2 f/\delta_v \end{bmatrix}. \tag{13}$$

From Eq. (12) we have

$$(u-u_0)[x\ y\ z\ 1]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} - \kappa\rho^2(u-u_0)[x\ y\ z\ 1] \times \begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} = [x\ y\ z\ 1]\begin{bmatrix} r_1 f/\delta_u \\ r_2 f/\delta_u \\ r_3 f/\delta_u \\ t_1 f/\delta_u \end{bmatrix}, \tag{14}$$

which leads to

$$[x\ y\ z\ 1]\begin{bmatrix} r_1 f/\delta_u + r_7 u_0 \\ r_2 f/\delta_u + r_8 u_0 \\ r_3 f/\delta_u + r_9 u_0 \\ t_1 f/\delta_u + t_3 u_0 \end{bmatrix} + [-ux-uy-uz-u]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} + \kappa\rho^2(u-u_0)[x\ y\ z\ 1]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} = 0 . \tag{15}$$

Similarly, from Eq. (13), we have

$$[x\ y\ z\ 1]\begin{bmatrix} r_4 f/\delta_v + r_7 v_0 \\ r_5 f/\delta_v + r_8 v_0 \\ r_6 f/\delta_v + r_9 v_0 \\ t_2 f/\delta_v + t_3 v_0 \end{bmatrix} + [-vx-vy-vz-v]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} + \kappa\rho^2(v-v_0)[x\ y\ z\ 1]\begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix} = 0 . \tag{16}$$

Let

$$\mathbf{P}_1 \equiv \begin{bmatrix} r_1 f/\delta_u + r_7 u_0 \\ r_2 f/\delta_u + r_8 u_0 \\ r_3 f/\delta_u + r_9 u_0 \\ t_1 f/\delta_u + t_3 u_0 \end{bmatrix}, \quad \mathbf{P}_2 \equiv \begin{bmatrix} r_4 f/\delta_v + r_7 v_0 \\ r_5 f/\delta_v + r_8 v_0 \\ r_6 f/\delta_v + r_9 v_0 \\ t_2 f/\delta_v + t_3 v_0 \end{bmatrix},$$

$$\mathbf{P}_3 \equiv \begin{bmatrix} r_7 \\ r_8 \\ r_9 \\ t_3 \end{bmatrix}, \quad \text{and} \quad \mathbf{c} \equiv \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix}.$$

Then using Eqs. (15) and (16), for all 2-D to 3-D pairs, we have

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_j & y_j & z_j & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_j & y_j & z_j & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}\begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \end{bmatrix} + \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ -u_j x_j & -u_j y_j & -u_j z_j & -u_j \\ -v_j x_j & -v_j y_j & -v_j z_j & -v_j \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}\mathbf{P}_3 + \kappa\begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ (u_j-u_0)\rho_j^2 x_j & (u_j-u_0)\rho_j^2 y_j & (u_j-u_0)\rho_j^2 z_j & (u_j-u_0)\rho_j^2 \\ (v_j-v_0)\rho_j^2 x_j & (v_j-v_0)\rho_j^2 y_j & (v_j-v_0)\rho_j^2 z_j & (v_j-v_0)\rho_j^2 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \times \mathbf{P}_3 = \begin{bmatrix} \vdots \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \tag{17}$$

with $\rho_j^2 = \delta_u^2(u_j-u_0)^2 + \delta_v^2(v_j-v_0)^2$.

Define

$$\mathbf{A} \equiv \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_j & y_j & z_j & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_j & y_j & z_j & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix},$$

$$\mathbf{B} \equiv \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ -u_j x_j & -u_j y_j & -u_j z_j & -u_j \\ -v_j x_j & -v_j y_j & -v_j z_j & -v_j \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix},$$

$$\mathbf{C} \equiv \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ (u_j - u_0)\rho_j^2 x_j & (u_j - u_0)\rho_j^2 y_j & (u_j - u_0)\rho_j^2 z_j & (u_j - u_0)\rho_j^2 \\ (v_j - v_0)\rho_j^2 x_j & (v_j - v_0)\rho_j^2 y_j & (v_j - v_0)\rho_j^2 z_j & (v_j - v_0)\rho_j^2 \\ \vdots & \vdots & \vdots & \end{bmatrix},$$

$$\mathbf{p} \equiv \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \end{bmatrix} / \|\mathbf{P}_3\| \quad \text{and} \quad \mathbf{q} \equiv \mathbf{P}_3 / \|\mathbf{P}_3\| \ . \tag{18}$$

Because the 2-D observation noise always exists, Eq. (17) will not be exactly zero, i.e.,

$$\mathbf{Ap} + \mathbf{Bq} + \kappa\mathbf{Cq} = \varepsilon \neq 0 \ . \tag{19}$$

In practice, the estimate of δ_u obtained by Eq. (11) is usually quite accurate, and the image center (u_0, v_0) is relatively small in comparison to almost all the image feature points (u_j, v_j)—when estimating κ, we can use only those calibration points that are relatively far away from the image center. Therefore, we can use the initial estimates of u_0, v_0, and δ_u as if they were the true values of u_0, v_0, and δ_u in the matrix $\mathbf{C}$. The deviation of the matrix $\mathbf{C}$ caused by the estimation error in $\hat{u}_0$, $\hat{v}_0$, and $\hat{\delta}_u$ is negligible in the following estimation of $\mathbf{p}$ and $\mathbf{q}$ because κ is usually very small and the deviation of the matrix $\mathbf{C}$ times κ in Eq. (17) is even more trivial. In other words, larger lens distortion should result in more calibration inaccuracy when using the proposed algorithm, because of this approximation of u_0, v_0, and δ_u. It is indeed the case, as shown in Fig. 8 in Sec. 4.1. However, a reasonable amount of lens distortion, as in most practical cases, should not cause any difficulties in using this approximation. Using this approximation, the matrix $\mathbf{C}$ is treated as a known matrix in the following computation.

Hence, the parameters to be estimated, κ and $\mathbf{c}$ (or equivalently, κ, $\mathbf{p}$, and $\mathbf{q}$) can be computed by minimizing the following criterion E with respect to κ, $\mathbf{p}$, and $\mathbf{q}$:

$$E \equiv \|\mathbf{Ap} + \mathbf{Bq} + \kappa\mathbf{Cq}\|^2 \ ,$$

subject to the constraint $\|\mathbf{q}\|^2 = 1$.

To minimize E, we form the Lagrangian

$$L \equiv \|\mathbf{Ap} + \mathbf{Bq} + \kappa\mathbf{Cq}\|^2 + \lambda(1 - \|\mathbf{q}\|^2) \ , \tag{20}$$

and set $\partial L/\partial \mathbf{p} = 0$, $\partial L/\partial \mathbf{q} = 0$, $\partial L/\partial \lambda = 0$, and $\partial L/\partial \kappa = 0$, which yields

$$\frac{1}{2}\frac{\partial L}{\partial \mathbf{p}} = \mathbf{A}^t\mathbf{Ap} + \mathbf{A}^t\mathbf{Bq} + \kappa\mathbf{A}^t\mathbf{Cq} = 0 \ , \tag{21}$$

$$\frac{1}{2}\frac{\partial L}{\partial \mathbf{q}} = \mathbf{B}^t\mathbf{Bq} + \kappa^2\mathbf{C}^t\mathbf{Cq} + \mathbf{B}^t\mathbf{Ap} + \kappa\mathbf{C}^t\mathbf{Ap} + \kappa\mathbf{C}^t\mathbf{Bq} + \kappa\mathbf{B}^t\mathbf{Cq} - \lambda\mathbf{q} = 0 \ , \tag{22}$$

$$\frac{1}{2}\frac{\partial L}{\partial \kappa} = \mathbf{q}^t\mathbf{C}^t\mathbf{Ap} + \mathbf{q}^t\mathbf{C}^t\mathbf{Bq} + \kappa\mathbf{q}^t\mathbf{C}^t\mathbf{Cq} = 0 \ . \tag{23}$$

From Eq. (21), we have

$$\mathbf{p} = -[(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{B} + \kappa(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{C}]\mathbf{q} \ . \tag{24}$$

Substituting Eq. (24) into Eqs. (20), (22), and (23), we have

$$L = \mathbf{q}^t[\mathbf{D} - \lambda\mathbf{I}]\mathbf{q} + \lambda \ , \tag{25}$$

$$\frac{1}{2}\frac{\partial L}{\partial \mathbf{q}} = [\mathbf{D} - \lambda\mathbf{I}]\mathbf{q} = 0 \ , \tag{26}$$

$$\frac{1}{2}\frac{\partial L}{\partial \kappa} = \mathbf{q}^t[\kappa\mathbf{T} + \mathbf{S}/2]\mathbf{q} = 0 \ , \tag{27}$$

where

$$\mathbf{D} \equiv [\kappa^2\mathbf{T} + \kappa\mathbf{S} + \mathbf{R}] \ , \tag{28}$$

$$\mathbf{R} \equiv \mathbf{B}^t\mathbf{B} - \mathbf{B}^t\mathbf{A}(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{B}, \quad \mathbf{T} \equiv \mathbf{C}^t\mathbf{C} - \mathbf{C}^t\mathbf{A}(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{C} \ , \tag{29}$$

and

$$\mathbf{S} \equiv \mathbf{C}^t\mathbf{B} - \mathbf{C}^t\mathbf{A}(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{B} + \mathbf{B}^t\mathbf{C} - \mathbf{B}^t\mathbf{A}(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{C} \ . \tag{30}$$

Notice that λ is the eigenvalue of $\mathbf{D}$, and that

$$E\Big|_{\frac{\partial E}{\partial \mathbf{p}} = 0, \frac{\partial E}{\partial \mathbf{q}} = 0, \frac{\partial E}{\partial \kappa} = 0} = \lambda \text{ subject to } \|\mathbf{q}\|^2 = 1,$$

i.e., the minimal residual error is $\lambda_{\min} = \min\{\text{eigenvalues of } \mathbf{D}\}$. Given a κ, $\mathbf{D}$ is uniquely determined and so is the minimal residual error $\lambda_{\min}(\kappa)$. When there is no 2-D observation error and the initial estimates of u_0, v_0, and δ_u are accurate, the minimal residual error $\lambda_{\min}$ is equal to zero. Then from Eqs. (26), we have

$$\mathbf{T}^{-1}[\mathbf{D} - 0\mathbf{I}]\mathbf{q} = [\kappa^2\mathbf{I} + \kappa\mathbf{T}^{-1}\mathbf{S} + \mathbf{T}^{-1}\mathbf{R}]\mathbf{q} = 0 \ . \tag{31}$$

It follows that

$$\kappa(\kappa\mathbf{q}) = [-\mathbf{T}^{-1}\mathbf{R} - \mathbf{T}^{-1}\mathbf{S}]\begin{Bmatrix} \mathbf{q} \\ \kappa\mathbf{q} \end{Bmatrix} \ , \tag{32}$$

and

$$\kappa(\mathbf{q}) = [0 \ \mathbf{I}]\begin{Bmatrix} \mathbf{q} \\ \kappa\mathbf{q} \end{Bmatrix} \ . \tag{33}$$

It is now obvious that κ is the real eigenvalue of an 8×8 matrix $\mathbf{K}$, where

$$\mathbf{K} \equiv \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{T}^{-1}\mathbf{R} & -\mathbf{T}^{-1}\mathbf{S} \end{bmatrix},$$

since

$$\kappa\begin{bmatrix} \mathbf{q} \\ \kappa\mathbf{q} \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{q} \\ \kappa\mathbf{q} \end{bmatrix} \ . \tag{34}$$

But in practice, no real κ can be found by solving the eigenvalues of $\mathbf{K}$, i.e., we will usually obtain a complex (impractical) κ that makes the residual error equal to zero. In our experience, we can choose to use the real part of the eigenvalue with the smallest absolute imaginary part as an

estimate of κ, denoted as $\hat{\kappa}$. Of course, we can use $\hat{\kappa}$ as the initial guess, and perform the one-dimensional nonlinear search to find the optimal κ. However, because the $\hat{\kappa}$ is accurate enough, further nonlinear search gains little according to our experience. Therefore, we omit the nonlinear search in the calibration procedure. Once $\hat{\kappa}$ is determined, the vector $\mathbf{q}$ can be obtained by selecting the eigenvector corresponding to the smallest eigenvalue of the matrix $[\hat{\kappa}^2\mathbf{T}+\hat{\kappa}\mathbf{S}+\mathbf{R}]$.

Substituting $\mathbf{q}$ into Eq. (24), we then have $\mathbf{p}$. Using the fact that the nine r_i's, in the definition of $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$, are components of a rotation matrix, we can decompose the vectors $\mathbf{p}$ and $\mathbf{q}$, and obtain[7,11] the parameters $\mathbf{R}_C^0$, t_1, t_2, t_3, f, u_0, v_0, and δ_u given δ_v.

3.2 *New Camera Calibration Technique Using Coplanar Calibration Points*

When only coplanar calibration points are available, the derivation of the algorithm is quite similar to the one that uses noncoplanar calibration points. But the decomposition of the parameters should be done carefully because the estimated results are more sensitive to noise. Without loss of generality, we can choose the x-y plane of the OCS to be the plane where the calibration plane lies, and all z's in Eqs. (12) and (13) vanish, thus

$$(1-\kappa\rho^2)(u-u_0)[x\ y\ 1]\begin{bmatrix} r_7 \\ r_8 \\ t_3 \end{bmatrix} = [x\ y\ 1]\begin{bmatrix} r_1 f/\delta_u \\ r_2 f/\delta_u \\ t_1 f/\delta_u \end{bmatrix}, \tag{35}$$

$$(1-\kappa\rho^2)(v-v_0)[x\ y\ 1]\begin{bmatrix} r_7 \\ r_8 \\ t_3 \end{bmatrix} = [x\ y\ 1]\begin{bmatrix} r_4 f/\delta_v \\ r_5 f/\delta_v \\ t_2 f/\delta_v \end{bmatrix}. \tag{36}$$

Similar to Sec. 3.1, we define

$$\mathbf{P}_1^C \equiv \begin{bmatrix} r_1 f/\delta_u + r_7 u_0 \\ r_2 f/\delta_u + r_8 u_0 \\ t_1 f/\delta_u + t_3 u_0 \end{bmatrix},$$

$$\mathbf{P}_2^C \equiv \begin{bmatrix} r_4 f/\delta_v + r_7 v_0 \\ r_5 f/\delta_v + r_8 v_0 \\ t_2 f/\delta_v + t_3 v_0 \end{bmatrix}, \quad \mathbf{P}_3^C \equiv \begin{bmatrix} r_7 \\ r_8 \\ t_3 \end{bmatrix}, \tag{37}$$

and then have

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_j & y_j & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_j & y_j & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}\begin{bmatrix} \mathbf{P}_1^C \\ \mathbf{P}_2^C \end{bmatrix} + \begin{bmatrix} \vdots & \vdots & \vdots \\ -u_j x_j & -u_j y_j & -u_j \\ -v_j x_j & -v_j y_j & v_j \\ \vdots & \vdots & \vdots \end{bmatrix}\mathbf{P}_3^C$$

$$+\kappa\begin{bmatrix} \vdots & \vdots & \vdots \\ (u_j-u_0)\rho_j^2 x_j & (u_j-u_0)\rho_j^2 y_j & (u_j-u_0)\rho_j^2 \\ (v_j-v_0)\rho_j^2 x_j & (v_j-v_0)\rho_j^2 y_j & (v_j-v_0)\rho_j^2 \\ \vdots & \vdots & \vdots \end{bmatrix}\mathbf{P}_3^C = \begin{bmatrix} \vdots \\ 0 \\ 0 \\ \vdots \end{bmatrix}. \tag{38}$$

To estimate the composite camera parameters, we repeat the derivation similar to that in Sec. 3.1, except that the definitions of matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are replaced by

$$\mathbf{A}^C \equiv \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_j & y_j & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_j & y_j & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix},$$

$$\mathbf{B}^C \equiv \begin{bmatrix} \vdots & \vdots & \vdots \\ -u_j x_j & -u_j y_j & -u_j \\ -v_j x_j & -v_j y_j & -v_j \\ \vdots & \vdots & \vdots \end{bmatrix},$$

$$\mathbf{C}^C \equiv \begin{bmatrix} \vdots & \vdots & \vdots \\ (u_j-\hat{u}_0)\hat{\rho}_j^2 x_j & (u_j-\hat{u}_0)\hat{\rho}_j^2 y_j & (u_j-\hat{u}_0)\hat{\rho}_j^2 \\ (v_j-\hat{v}_0)\hat{\rho}_j^2 x_j & (v_j-\hat{v}_0)\hat{\rho}_j^2 y_j & (v_j-\hat{v}_0)\hat{\rho}_j^2 \\ \vdots & \vdots & \vdots \end{bmatrix},$$

and the decomposition procedure are modified as follows.

Suppose that the estimated composite parameters are $\mathbf{P}_1^C = t_3 \cdot [a_1\ a_2\ a_3]^t$, $\mathbf{P}_2^C = t_3 \cdot [a_4\ a_5\ a_6]^t$ and $\mathbf{P}_3^C = t_3 \cdot [a_7\ a_8\ 1]^t$, where a_i, $i=1, \ldots, 8$, are real numbers. Substituting these values into the definitions of $\mathbf{P}_1^C$, $\mathbf{P}_2^C$, and $\mathbf{P}_3^C$, i.e., Eq. (37), we have the following eleven equations in eleven unknowns: r_1, r_2, r_4, r_5, r_7, r_8, t_1, t_2, t_3, δ_u, and f:

$$\begin{aligned} r_1 &= (a_1 - a_7 u_0)\delta_u t_3/f \\ r_2 &= (a_2 - a_8 u_0)\delta_u t_3/f \\ t_1 &= (a_3 - u_0)\delta_u t_3/f \\ r_4 &= (a_4 - a_7 v_0)\delta_v t_3/f \\ r_5 &= (a_5 - a_8 v_0)\delta_v t_3/f \\ t_2 &= (a_6 - v_0)\delta_v t_3/f \\ r_7 &= a_7 t_3 \\ r_8 &= a_8 t_3 \\ r_1^2 + r_4^2 + r_7^2 &= 1 \\ r_2^2 + r_5^2 + r_8^2 &= 1 \\ r_1 r_2 + r_4 r_5 + r_7 r_8 &= 0, \end{aligned} \tag{39}$$

where the last three equations are the constraints of a rotation matrix. Substituting the first eight equations into the last three, we have

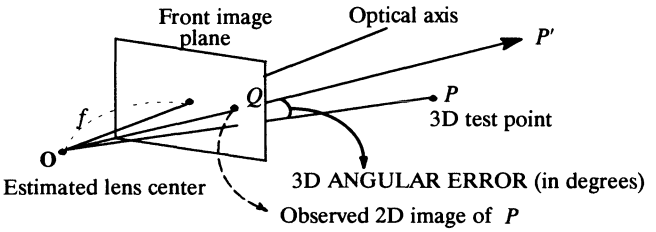


Fig. 3 Definition of 3-D angular error.

$$\begin{bmatrix} (a_1-a_7u_0)^2 & (a_4-a_7v_0)^2 & a_7^2 \\ (a_2-a_8u_0)^2 & (a_5-a_8v_0)^2 & a_8^2 \\ (a_1-a_7u_0)(a_2-a_8u_0) & (a_4-a_7v_0)(a_5-a_8v_0) & a_7a_8 \end{bmatrix} \times \begin{bmatrix} (\delta_u t_3/f)^2 \\ (t_3/f)^2 \\ t_3^2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} . \quad (40)$$

By solving Eq. (40), t_3, δ_u, and f can be found, because they are known to be positive numbers. Knowing the values of t_3, δ_u, and f, we can easily obtain all the extrinsic parameters in Eq. (39). Finally, the last column of the rotation matrix can be obtained by calculating the cross product of the first two columns, i.e.,

$$\begin{bmatrix} r_3 \\ r_6 \\ r_9 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_4 \\ r_7 \end{bmatrix} \times \begin{bmatrix} r_2 \\ r_5 \\ r_8 \end{bmatrix} . \quad (41)$$

4 Experimental Results

To evaluate the accuracy of the camera calibration for 3-D vision applications, it is necessary to define certain types of error measures. The measure used in this paper is the 3-D angular error, i.e., the angle $\sphericalangle POP'$ shown in Fig. 3, where P is the 3-D test point, O is the estimated lens center, and OP' is the 3-D ray back projected from the observed 2-D image of P using the estimated camera parameters. A 3-D angular error of 0.005 deg is roughly equivalent to 1 part in 10,000, because $\tan(0.005 \text{ deg}) \approx 1/10{,}000$.

For convenience, let $\beta_{L\kappa}$, β_T, and β_N denote the estimate of β obtained respectively by our algorithm, Tsai's[1] two-stage algorithm, and the Weng et al.[3] two-step nonlinear algorithm (only radial lens distortion is considered). To compute $\beta_{L\kappa}$ and β_T, we require that u_0 and v_0 are known *a priori*, or at least, we must have some initial guesses. In the real experiments, the true parameter values are unknown, and we set the initial values for u_0 and v_0 to zero and the initial value for δ_u to be δ_u' ($f_{\text{camera}}/f_{\text{digitizer}}$) for the reasons explained in Sec. 3. In the simulations, the parameters of the synthetic camera is set (and known) to be $f = 25.85$ mm, $\delta_u = 15.66$ μm, and $\delta_v = 13$ μm, whereas the initial value for δ_u is chosen to be $\delta_u'(f_{\text{camera}}/f_{\text{digitizer}}) = 11$ μm(14.31818 MHz/10 MHz) = 15.75 μm, which introduces about 0.58% error for δ_u. The images used in the experiments are 480×512 pixels in size. The data shown later in Figs. 5 to 16 and Figs. 18 to 22 are an average of at least 10 random trials.

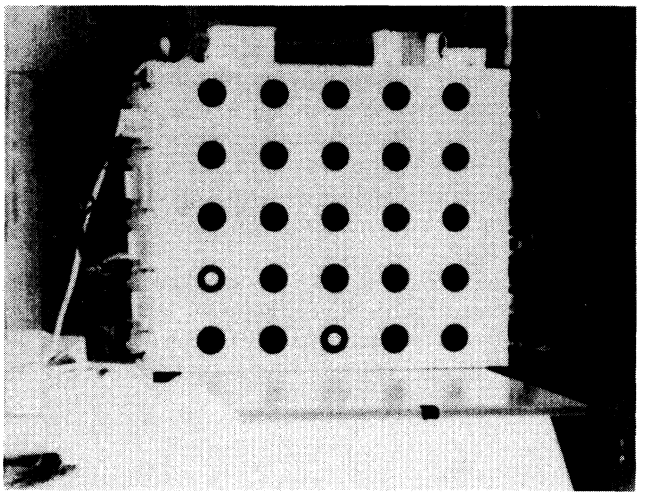


Fig. 4 A typical image of the calibration plate containing 25 calibration points used in the real experiment.

Finally, in Sec. 4.3, one of the most interesting applications, stereo vision, is used to test the performance of different camera calibration techniques. The accuracy of the estimate of a 3-D point position is strongly related to the 3-D angular error defined above. This experiment demonstrates that our calibration method can be used to achieve highly accurate 3-D measurements as expected.

4.1 *Performance of Different Techniques Using Noncoplanar Calibration Points*

The first experiment is to determine a proper number of calibration points to be used for camera calibration. The image acquisition system we use includes a PULNiX TM-745E CCD camera and an ITI Series 151 frame grabber. The calibration object is a 300 by 300-mm white plate having 25 black calibration circles on it. This calibration plate was mounted rigidly on a translation stage, and was initially about 1800 mm away from the camera. For this experiment one image was taken each time the calibration plate was moved toward the camera by 25 mm, and altogether 21 images of the calibration plate were taken when the calibration plate was moved from the range of 1800 mm to the range of 1300 mm. A typical image is shown in Fig. 4. The centers of the black circles (and the black doughnuts) are used as the calibration points. The 2-D image coordinates of those calibration points can be estimated with subpixel accuracy. Because the calibration plate and the translation stage are manufactured with very high accuracy, the 3-D positions of those calibration points can be treated as known values with respect to a fixed object coordinate system. Thus, we have $25 \times 21 = 525$ pairs of 2-D, 3-D coordinates of control points. Here, we randomly choose N_{calib} points ($N_{\text{calib}} = 10, 20, \ldots, 200$) from the 525 2-D, 3-D pairs to calibrate the camera and use all remaining points to test the calibrated parameters. The average of the results of 10 random trials are plotted in Fig. 5, which shows the 3-D angular error decreases as the number of calibration points increases, no matter which of the three methods is used. When the number of calibration points is greater than 60, increasing N_{calib} results in little gain in accuracy, for all the three

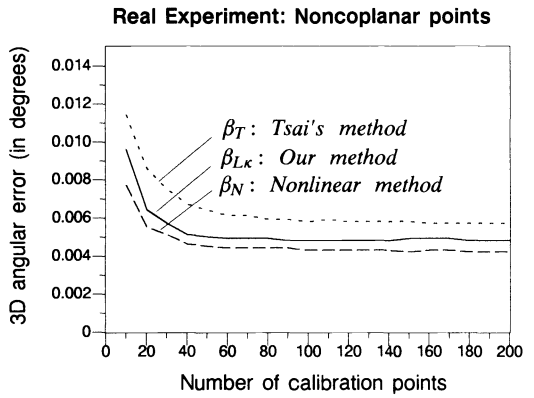


Fig. 5 3-D angular error versus the number of calibration points.

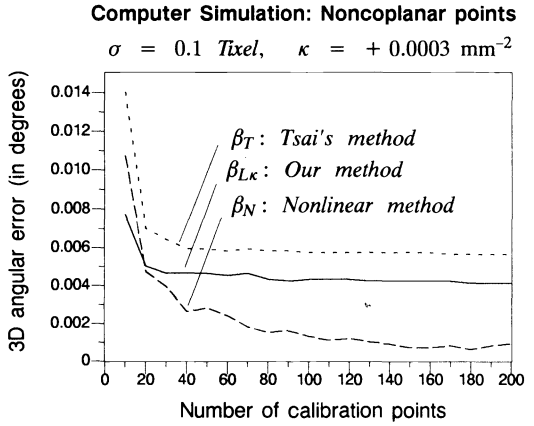


Fig. 6 3-D angular error versus the number of calibration points.

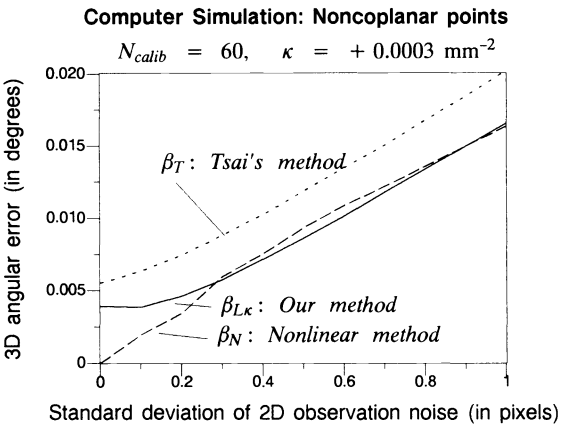


Fig. 7 3-D angular error versus the 2-D observation noise.

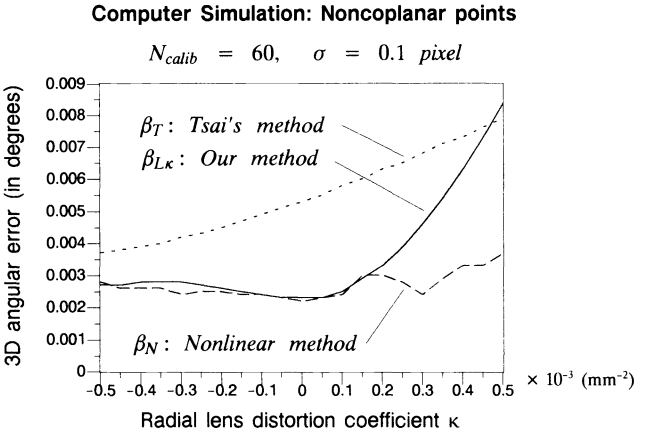


Fig. 8 3-D angular error versus the radial lens distortion coefficient.

calibration methods. Therefore, we simply choose $N_{\text{calib}} = 60$ in the simulations shown in Figs. 6 through 16. Because the positions of the 3-D calibration points can be controlled with high accuracy, we assume that the 3-D coordinates of the calibration points are known exactly, and the only source of the measurement error is the error in estimating the 2-D image coordinates of the 3-D calibration points, i.e., the 2-D observation error. The standard deviation of the 2-D observation error is denoted by σ.

The purpose of the first simulation is to check the results of these real experiments. In this simulation, the radial lens distortion is chosen to be $\kappa = 0.0003$ mm^{-2} (which is the estimate of κ in these real experiments, and corresponds to roughly 2 to 3 pixels distortion near the four corners of a 480×512 image) and σ is assumed to be 0.1 pixel (which is roughly the subpixel accuracy we can achieve using the circular calibration patterns shown in Fig. 4). Figure 6 shows similar decreasing of the 3-D angular error as N_{calib} increases, except that the 3-D angular error can decrease further as N_{calib} increases when using the nonlinear method. This is mainly because the 2-D and 3-D coordinates of the test points used in the simulations are noiseless, whereas in the real experiment, ground truth is never known and the measurement error dominates.

The next simulation shows the deterioration of the three calibration techniques as the 2-D observation errors increase (see Fig. 7). The 3-D angular error of Tsai's[1] technique is always greater than the other two. This is partially because each 2-D, 3-D pair of the calibration points contributes only one equation when using the radial alignment constraint,[1] whereas the other two techniques use both the horizontal and vertical components of the perspective projection relation and each 2-D, 3-D pair contributes two equations in the minimization. Another reason is that Tsai's[1] algorithm does not estimate the image center. Although the image center could be estimated via a separate procedure,[3] we did not implement that image center estimation algorithm. Instead, we showed that, given the same guess for the image center, our method performs better than Tsai's method. When given a more accurate image center, both Tsai's and our techniques obtain better results, but our method still performs better than Tsai's does. Sometimes, our method has even better performance than the Weng et al.[3] method, because their method can be trapped in a local minimum.

In the third simulation, we set the 2-D observation error $\sigma = 0.1$ and the number of calibration points $N_{\text{calib}} = 60$ and observe that the 3-D angular error varies versus the radial lens distortion coefficient (see Fig. 8). Notice that both Tsai's[2] method (β_T) and our method ($\beta_{L\kappa}$) degrade when κ becomes larger. This is partially caused by being given a wrong image center. Figure 8 also displays a phenomenon that we predicted in Sec. 3, i.e., our technique should have

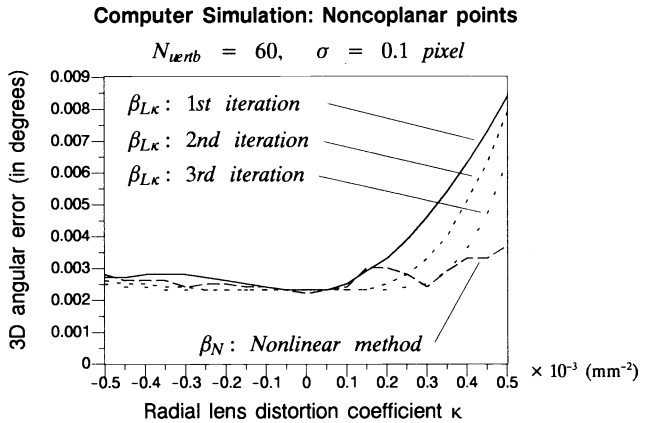


Fig. 9 3-D angular error versus the radial lens distortion coefficient.

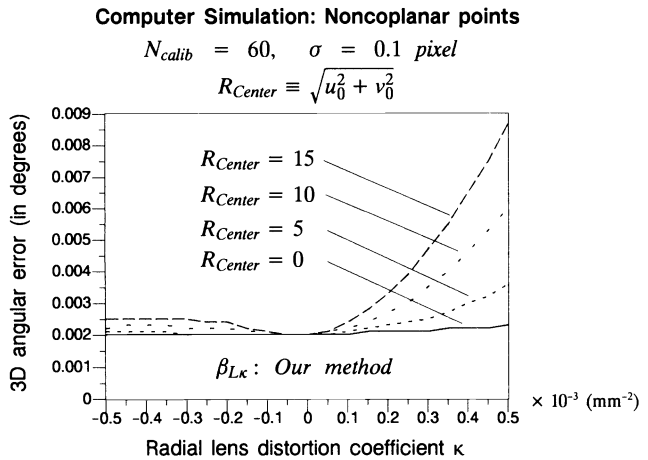


Fig. 10 3-D angular error versus the radial lens distortion coefficient.

better performance when the radial lens distortion is smaller. When the given image center is not accurate and $\kappa > 0$, our technique tends to overestimate κ. Thus, for a large positive κ, our technique may give a worse estimate than those given by others if the initial estimate for the image center is not good enough. However, in applications, we often choose to use a standard lens ($|\kappa| \approx 0$) or a wide-angle lens ($\kappa < 0$), rather than a lens with $\kappa \gg 0$. Therefore, the proposed technique is suitable for most 3-D computer vision applications requiring high accuracy. If for some reason, we need to choose a lens with relatively large positive κ, our calibration procedure can be executed iteratively by substituting the originally given image center (u_0, v_0) and δ_u with the newly estimated ones. Results of the first three iterations are shown in Fig. 9. Notice that the performance of the nonlinear method, which typically requires 40 to 100 iterations, is sometimes even worse than that obtained after the second (or the third) iteration of our method.

The fourth simulation shows how Tsai's[1] method and our method degrade as the error of the given image center becomes large. In this simulation, the true image center is $[R_{\text{center}} \cos(\alpha), R_{\text{center}} \sin(\alpha)]$, $R_{\text{center}} = 0$, 5, 10, 15, and the *a priori* given image center is set to (0,0). Each data point shown in Figs. 10 and 11 is the average of the results of 40 random trials, 10 for each angle of $\alpha = \pm 45, \pm 135$

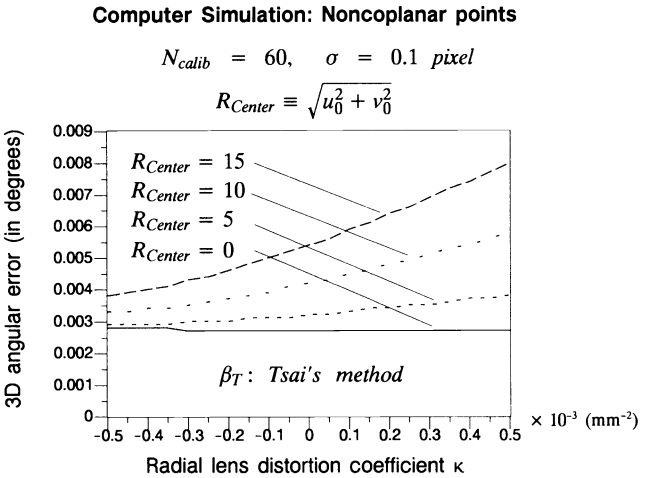


Fig. 11 3-D angular error versus the radial lens distortion coefficient.

deg. Notice that when the given image center is less accurate, both Tsai's method (β_T) and our method ($\beta_{L\kappa}$) degrade more when κ becomes larger. If we use the true image center in the simulation, then the curves corresponding to β_T and $\beta_{L\kappa}$ are both quite flat.

4.2 *Performance of Different Techniques Using Coplanar Calibration Points*

In Sec. 3.2 we proposed a method for calibrating a camera using coplanar calibration points. Because the calibration technique using coplanar calibration points (coplanar technique, for short) behaves somewhat differently compared to using a noncoplanar one (noncoplanar technique, for short), we discuss the experiments using coplanar techniques in this section. We set the 2-D observation error $\sigma = 0.1$ and the number of calibration points $N_{\text{calib}} = 60$ for the following experiments. Note that neither Tsai's[1] coplanar techniques nor our coplanar techniques estimate the image center, but ours gives an estimate of δ_u, whereas Tsai's does not.

We first show that Tsai's[1] coplanar technique is less sensitive to the error of the *a priori* given image center. Similar to the fourth simulation described in Sec. 4.1, the true image center is $[R_{\text{center}} \cos(\alpha), R_{\text{center}} \sin(\alpha)]$, whereas the image center given to the algorithm is (0,0). Figures 12 and 13 are the results obtained by using the true δ_u, whereas Figs. 14 and 15 are obtained by using Eq. (11) as an estimate of δ_u, which has a relative error of about 0.58%. Notice that our coplanar technique is more robust when the given estimate of the horizontal pixel spacing δ_u is not exactly the true value, whereas Tsai's method is more robust to the deviation of the given estimate of the image center (u_0, v_0). The former is because the parameter δ_u is estimated in our coplanar technique, and the latter is because what we obtain from our linear calibration procedure are composite parameters. The composite parameters are the redundant combination of the real parameters, which means an erroneous combination of these parameters can still make a good fit between experimental observations and model prediction.[1]

Also, it is interesting to note that Tsai's[1] coplanar technique performs almost equally to Tsai's noncoplanar technique, if the given δ_u is accurate (see Fig. 13). However, the performance of Tsai's coplanar technique is sensitive to

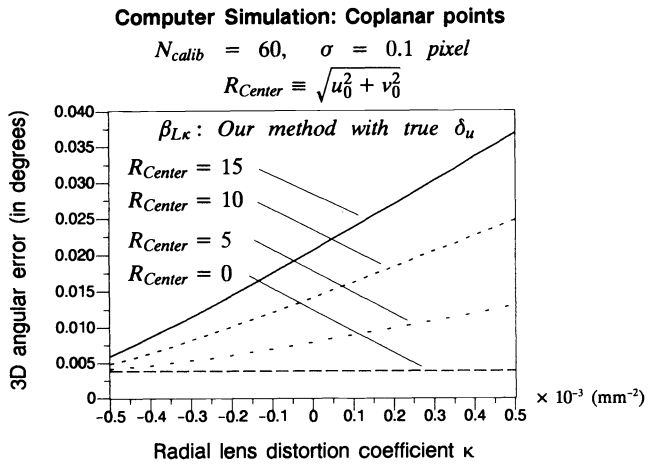


Fig. 12 3-D angular error versus the radial lens distortion coefficient using our method with true δ_u.

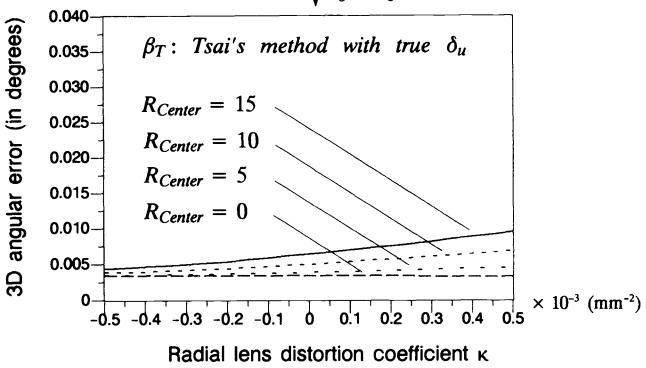


Fig. 13 3-D angular error versus the radial lens distortion coefficient using Tsai's[1] method with true δ_u.

the accuracy of the given δ_u (e.g., referring to Fig. 15, a 0.58% error of δ_u induced a large error on the calibration result). For comparison, each curve in Fig. 16 is an average of the four curves in each of the four graphs in Figs. 12 to 15. Therefore, unless a very precise estimate of δ_u is available,[4] our calibration technique is a better choice whether the calibration points are coplanar or noncoplanar.

4.3 *Performance of Different Techniques Tested Using Stereo Vision*

The following experiment demonstrates that our calibration method can be used to achieve highly accurate 3-D measurements, as expected, and that the accuracy of the estimate of a 3-D point position is strongly related to the 3-D angular error used in Secs. 4.1 and 4.2. In stereo vision, the 3-D position of an object point seen in two images is estimated by back projecting two straight lines corresponding to the two image points and finding the midpoint of their common normal line segment. The stereo vision setup is shown in Fig. 17. The distance between two optical centers of the cameras, i.e., the length of the base line, is approximately 300 mm.

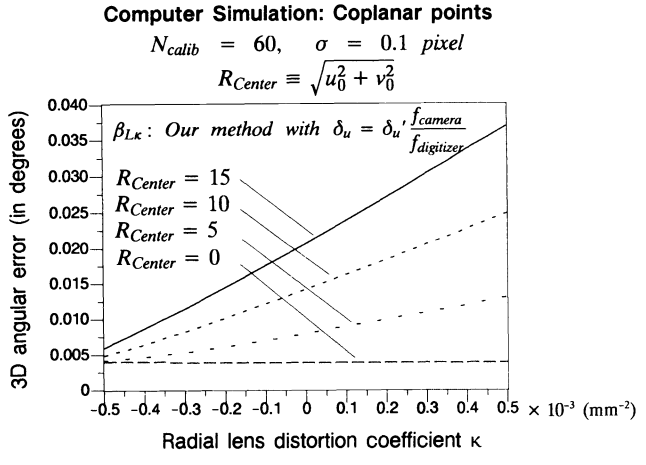


Fig. 14 3-D angular error versus the radial lens distortion coefficient using our method with an estimate of δ_u having 0.58% of error.

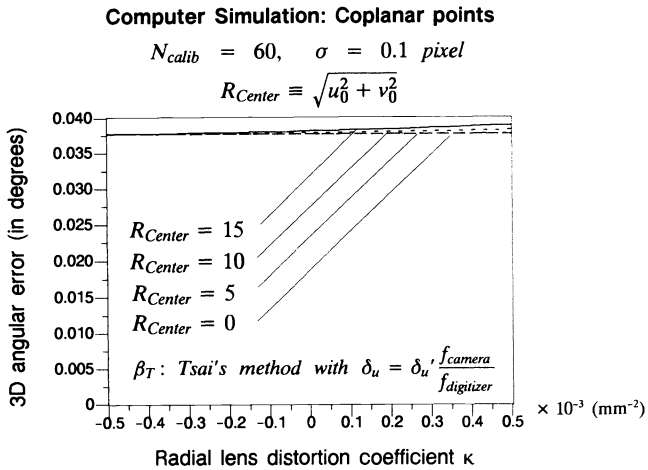


Fig. 15 3-D angular error versus the radial lens distortion coefficient using Tsai's[1] method with an estimate of δ_u having 0.58% of error.

First, 525 points distributed in the depth range of 1300 to 1800 mm are obtained for each camera by means of the procedures described in Sec. 4.1. We randomly choose N_{calib} points ($N_{\text{calib}} = 10, 20, \ldots, 200$) from the 525 2-D, 3-D pairs to calibrate the camera and use all remaining points to test the calibrated parameters. Correspondences between the two images of the test points can be determined easily because of the specific arrangement of the 23 black circles and two doughnuts. Figures 18 through 20 show the estimation error of the 3-D position in the x, y, and z directions, respectively, using a stereo vision system with $\kappa_{\text{left}} = -2.1 \times 10^{-4}$ mm^{-2} and $\kappa_{\text{right}} = -1.3 \times 10^{-4}$ mm^{-2}. Combining the results in Figs. 18 to 20, the 3-D distance errors for three different techniques are shown in Fig. 21. Here, the 3-D distance error is defined as the Euclidean norm of the estimation error of a 3-D point position. As shown earlier in the simulation, the accuracy of a single camera will be 1 part in 10,000 for our setup when more than 60 calibration points are used. Having the test points at the distance of approximately 1500 mm, the estimation error in the x and y directions should be approximately 0.15 mm in each direction. However, the stereo accuracy may be slightly worse than that we have predicted, because the

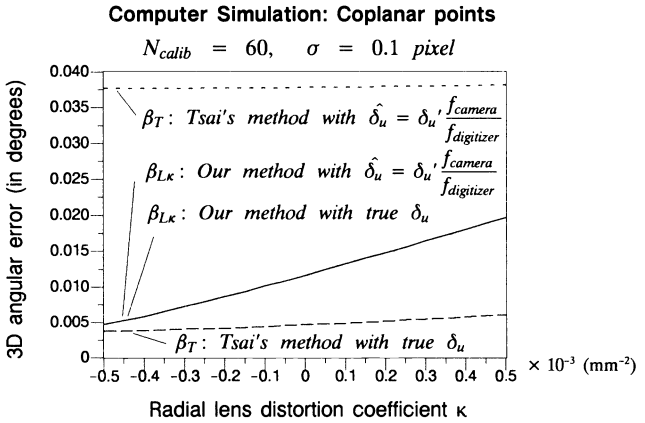


Fig. 16 3-D angular error versus the radial lens distortion coefficient.

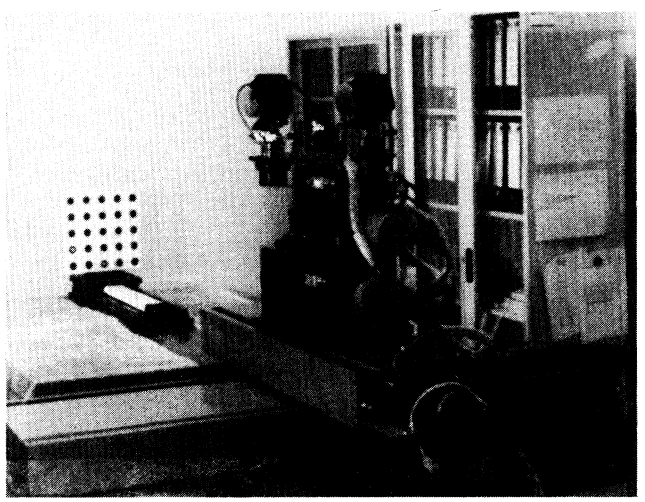


Fig. 17 Picture of the stereo vision setup.

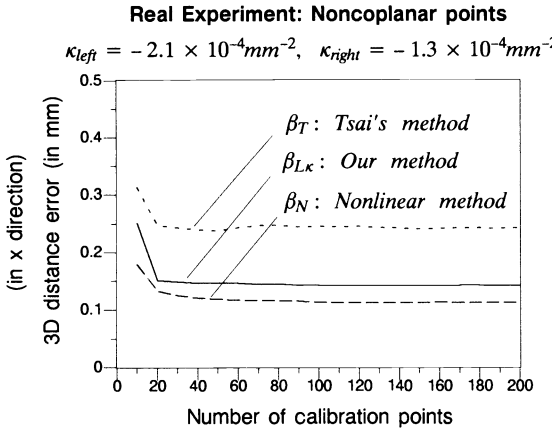


Fig. 18 3-D error in the x direction versus the number of calibration points with the distortion coefficient on the order of 10^{-4} (barrel distortion).

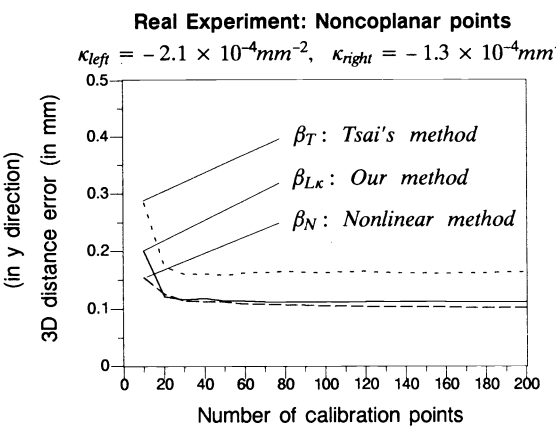


Fig. 19 3-D error in the y direction versus the number of calibration points with the distortion coefficient of the order of 10^{-4} (barrel distortion).

estimation error may come from both of the stereo cameras. Furthermore, because the length of the base line, 300 mm, is much smaller than the depth, 1500 mm, the estimation error in the z direction (0.6 mm) is much larger than that in the x (0.15 mm) or y (0.11 mm) directions. Notice that, if the lens distortion is relatively small, our method is much better than Tsai's,[1] and its accuracy is close to that of the nonlinear method (as in the above experiment). However, when the lens distortion coefficients are positive and relatively large, our method is only slightly better than Tsai's method, as shown in Fig. 22.

5 Conclusions

In this paper, we describe a new camera calibration technique that is computationally fast and can achieve very high calibration accuracy for 3-D computer vision applications. Our method is fast because it requires only linear computation and does not require any iterations. With a SPARCstation, the computation for the calibration can be done within a fraction of a second. Our method can achieve very high accuracy for 3-D estimation (i.e., small 3-D angular error) because the effect of the lens distortion is considered and all the information contained in the calibration points are used. We have shown that our new noncoplanar calibration method can achieve the 3-D angular error of 0.005 deg, or an accuracy of 1 part in 10,000 in 3-D measurement. This means that we will make only 0.1 mm of error in 3-D position estimation along the plane parallel to the image plane when the object is 1 m away from the camera. However, when the length of the base line is much smaller than the depth, the estimation error in the z direction (the depth) is much larger than that in the x or y directions. When using good quality off-the-shelf lenses, our linear method is good enough for almost all 3-D applications, and further nonlinear iteration is not necessary. Note also that it may not be worth the effort to increase the calibration accuracy using nonlinear minimization techniques unless we can increase the accuracy of 2-D feature extraction accordingly.

Acknowledgments

The authors would like to thank Dr. Roger Tsai for his help in the initial stages of this work and the reviewers for their helpful suggestions. This research is supported in part by the National Science Council, Taiwan, under grant NSC 81–0408-E-001-04.

References

1. R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3-D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Automat.* **RA-3**(4), 323–344 (1987).

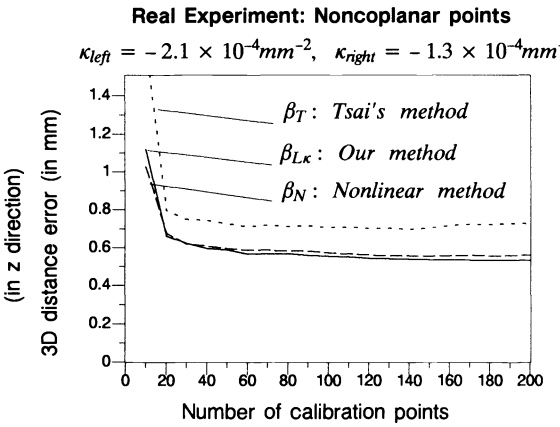


Fig. 20 3-D error in the z direction versus the number of calibration points with the distortion coefficient of the order of 10^{-4} (barrel distortion).

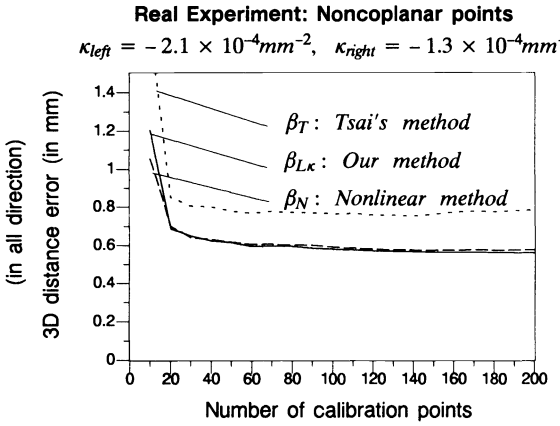


Fig. 21 3-D distance error versus the number of calibration points with the distortion coefficient of the order of 10^{-4} (barrel distortion).

2. W. Faig, "Calibration of close-range photogrammetry systems: mathematical formulation," *Photogram. Eng. Remote Sensing* **41**(12), 1479–1486 (1975).
3. J. Weng, P. Cohen, and M. Herniou, "Calibration of stereo cameras using a non-linear distortion model," in *Proc. IAPR 10th Inter. Conf. on Pattern Recognition*, pp. 246–253 (1990).
4. R. K. Lenz and R. Y. Tsai, "Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology," *IEEE Trans. Pattern Anal. Mach. Intell.* **10**(5), 713–720 (1988).
5. W. I. Grosky and L. A. Tamburino, "A unified approach to the linear camera calibration problem," *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(7), 663–671 (1990).
6. O. D. Faugeras and G. Toscani, "The calibration problem for stereo," in *Proc. Conf. on Computer Vision and Pattern Recognition*, pp. 15–20 (1986).
7. T. M. Strat, *Recovering the Camera Parameters from a Transformation Matrix*, DARPA Image Understanding Workshop, pp. 264–271 (1984).
8. I. Sutherland, "Three-dimensional data input by tablet," *Proc. IEEE* **62**(4), 453–461 (1974).
9. K. Kanatani and Y. Onodera, "Noise robust camera calibration using vanishing points," *IEICE Trans. Inform. Syst.* **E74**(10) (1991).
10. S. Ganapaphy, "Decomposition of transformation matrices for robot vision," in *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 130–139 (1984).
11. Y. P. Hung, "Three dimensional surface reconstruction using a moving camera: a model-based probabilistic approach," PhD Dissertation, Division of Engineering, Brown University, Providence, R.I. (1990); also, Technical Report LEMS-63 (1989).
12. H. A. Martins, J. R. Birk, and R. B. Kelley, "Camera models based on data from two calibration planes," *Comput. Graph. Image Process.* **17**, 173–180 (1981).

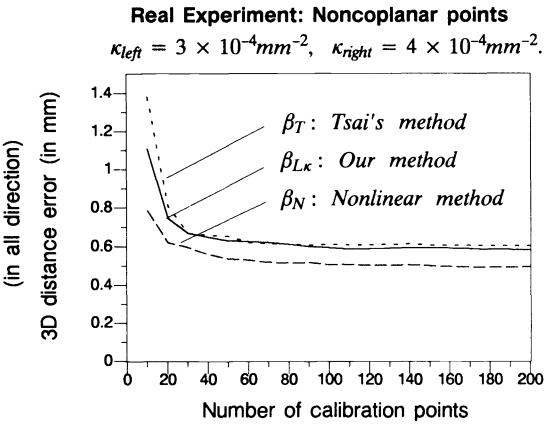
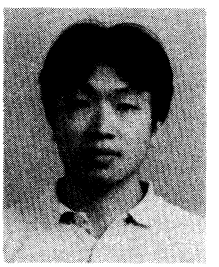
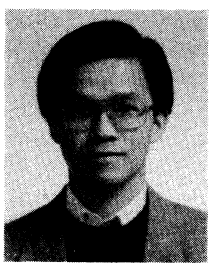


Fig. 22 3-D distance error versus the number of calibration points with the distortion coefficient of the order of 10^{-4} (pincushion distortion).

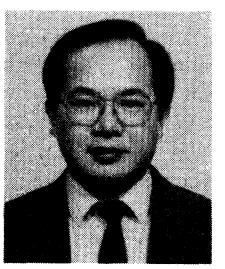
13. Y. P. Hung and S. W. Shih, "When should we consider lens distortion in camera calibration," in *IAPR Workshop on Machine Vision Applications*, Tokyo, pp. 367–370 (1990).
14. R. O. Duda and P. E. Hart, *Pattern Recognition and Scene Analysis*, Wiley, New York (1973).



Sheng-Wen Shih received the MS degree from the National Taiwan University in electrical engineering in 1990. Since then he has joined the Institute of Information Science, Academia Sinica. He is now also a PhD student in the Institute of Electrical Engineering at the National Taiwan University. His current research interests are in active vision, image sequence analysis, and robotics.



Yi-Ping Hung received the BSEE degree from the National Taiwan University in 1982, the ScM degree in electrical engineering, the ScM degree in applied mathematics, and the PhD degree in electrical engineering, all from Brown University, in 1987, 1988, and 1990, respectively. He is currently an associate research fellow in the Institute of Information Science, Academia Sinica, and an adjunct associate professor at the National Taiwan University. His research interests include computer vision and robotics.



Wei-Song Lin received the BS degree in engineering science and the MS degree in electrical engineering from National Cheng Kung University, Taiwan, in 1973 and 1975, respectively, and attained a distinguished paper award from the Association of Chinese Electrical Engineers. He received the PhD degree from the Institute of Electrical Engineering of National Taiwan University in 1982 and was an associate professor there from 1983 to 1987. From 1982 to 1984 he was also the head of Electronic Instrument Division of Ching Ling Industrial Research Center, where he worked mainly on design and implementation of electronic instruments. Since 1987 he has been a professor at National Taiwan University. He is now a member of the International Association of Science and Technology for Development and the National Committee of the Internal Union of Radio Science. He received an Excellent Research Award from the National Science Council in 1991. His current research interests are in the field of computer control, computer-based sensing and instrumentation, analysis and design of dynamic control systems, system engineering, and automation.