# CAMERA CALIBRATION FOR IMAGE PROCESSING

The camera calibration problem is well-studied in photogrammetry and computer vision, and it is crucial to several applications in manufacturing, metrology, and aerospace. The topic includes the body of techniques by which we compute (1) the three-dimensional (3-D) position and orientation of the camera relative to a certain world coordinate system (exterior orientation) and (2) the internal camera geometric and optical characteristics (interior orientation). In photogrammetric terminology, the *exterior orientation* of a camera is specified by all the parameters that determine the pose of the camera in the world reference frame. The parameters consist of the position of the center of perspectivity and the direction of the optical axis. Specification of the exterior orientation therefore requires three rotation angles and three translation parameters and is accomplished by obtaining the 3-D coordinates of some control points whose corresponding positions in the image are known. The *interior orientation* of a camera is specified by all the parameters that relate the geometry of ideal perspective projection to the physics of an actual camera. The parameters include the focal length, the image center, the scale factor, and the specification of the lens distortion.

There are two types of camera calibration problems: (1) *noncoplanar* calibration, in which the world points lie on a 3-D surface, and (2) *coplanar* calibration, in which the world points are on a two-dimensional (2-D) plane. A solution to all extrinsic and intrinsic calibration parameters requires nonlinear optimization. In some applications such as passive stereo image analysis, camera calibration may be performed slowly to obtain very accurate results. However, there are several applications that require repeated computation of the camera parameters at real-time or near-real-time speeds. Examples are: (1) determining the position of a camera mounted on a moving airplane, (2) 3-D shape measurement of mechanical parts in automatic parts assembly, (3) part dimension measurement in automatic part machining, and (4) navigation of a camera-mounted land vehicle. Thus, in some applications a complete nonlinear parameter estimation procedure can be employed, whereas others require fast and near-real-time calibration algorithms. In light of these varying applications, we find many algorithms for camera calibration, each satisfying a set of applications with different levels of accuracy.

The literature for camera calibration is vast and we offer a brief review. Many earlier studies (1–7) have primarily considered the extrinsic parameters, although some intrinsic parameters are also computed only for the noncoplanar case. These methods are usually efficient mostly due to the computation of linear equations. However, the methods generally ignore nonlinear lens distortion, and the coplanar case is also dealt with inadequately. For example, Sobel (6) used the basic pinhole camera model and utilized nonlinear optimization methods to compute 18 parameters. The nonlinear approach is similar to the parametric recursive least-squares method discussed below. He did not model lens distortion, and the system depended on the user to provide initial parameters for the optimization technique. Gennery (8) found the camera parameters iteratively by minimizing the error of epipolar constraints, but the method is too error-prone as observed by Tsai (9). Yakimovsky and Cunningham (7) and Ganapathy (2) also used the pinhole model and treated some combinations of parameters as single variables in order to formulate the problem as a linear system. However, in this formulation, the variables are not completely linearly independent, yet are treated as such. Furthermore, these methods mostly ignore constraints that the extrinsic and intrinsic parameters must obey, and hence the solutions are suboptimal. For example, the orthonormality constraints on the extrinsic rotation parameters are not strictly imposed. Grosky and Tamburino (4) extended the above linear methods to also include a skew angle parameter, and then they satisfied the orthonormality constraints. In general, linear methods are computationally fast, but in the presence of noise the accuracy of the final solution is relatively poor.

Tsai (9) and Lenz and Tsai (10) offered a two-stage algorithm using a radial alignment constraint (RAC) in which most parameters are computed in closed form. A small number of parameters such as the focal length, depth component of the translation vector, and radial lens distortion parameters are computed by an iterative scheme. If image center is unknown, it is determined by a nonlinear approach (10) based on minimizing the RAC residual error. Although the solution is efficient and the closed-form solution is immune to radial lens distortion, the formulation is less effective if tangential lens distortion is also included. Furthermore, by taking the ratio of the collinearity conditions [see Eq. (1)], the method considers tangential information only and radial information is discarded. This is not an optimal solution because all information from calibration points is not fully considered. For example, Weng et al. (11) suggest that ignoring radial information can result in a less reliable estimator. Furthermore, the orthogonality of the first two rows of the rotation matrix is not guaranteed. However, Tsai (9) has offered one of the few algorithms for the coplanar case.

Next, we study the computational procedure in traditional analytical photogrammetry. The method (12–16) is based on the parametric recursive procedure of the method of least squares. Using the Euler angles for the rotation matrix [see Ref. 17], two nonlinear collinearity equations are obtained for each observation. The nonlinear equations are linearized using Newton's first-order approximation. Based on the assumption of normal distribution of errors in measurements, the condition of maximizing the sum of squares of the residuals results in the maximum-likelihood values of the unknowns. Several iterations of the solution must be made to eliminate errors due to the linearization procedure. That is, the computed corrections are applied to the approximations at the end of each iteration, which form the new approximations in the next iteration. Initial solution is a prerequisite to this recursive procedure. In addition, Malhotra and Karara (14) note that "this solution needs considerable computational effort when the number of parameters in the adjustment are large." Other researchers (4,5,9) also make the same observation. Faig (12), Wong (16), and Malhotra and Karara (14) used the above method for a general solution of all parameters. The generality of their models allows them to accommodate many types of distortions, and it leads to accurate results. However, convergence is not guaranteed. These earlier nonlinear methods obtain good results provided that the estimation model is good, and a good initial guess is available.

The direct linear transformation (DLT) method of Abdel-Aziz and Karara (18,19) is a nonlinear method when lens distortion is corrected. However, in this formulation, depth

J. Webster (ed.), *Wiley Encyclopedia of Electrical and Electronics Engineering*. Copyright © 1999 John Wiley & Sons, Inc.

components of control points in a camera-centered coordinate system are assumed to be constant. Beyer (20) uses a nonlinear formulation and a versatile method based on self-calibrating bundle adjustment. The method is general, and accurate results of up to 1/46th of the pixel spacing in image space are reported. With redundant control points, an accuracy of 1/79th of the pixel spacing is achieved. Nomura et al. (21) also uses a nonlinear method where the dimension of the parameter space for nonlinear optimization is five. This reduced dimensionality for nonlinear optimization is obtained by setting up the calibration chart precisely to eliminate two Euler angles. Scale factor and depth components of the translation vector are assumed known or simply computed.

Although starting estimates are required in all nonlinear methods, the research by Weng et al. (11) is an example where a method for obtaining starting estimates is explicitly mentioned. As shown by Weng et al., computing all extrinsic parameters with lens distortion is necessary for accurate camera calibration. The method employs a two-step algorithm that first computes a set of starting values for all parameters, and then it uses linear and nonlinear optimization schemes to obtain accurate estimates. Each iteration improves all parameters by successive minimization of an objective function. The method does not deal with the coplanar case.

We next mention the method due to Chatterjee, Roychowdhury, and Chong (22) that uses the Gauss–Seidel framework (23,24) for nonlinear minimization. It exploits the structure of the calibration problem to use smaller blocks that are solved by linear iterations or computed in closed form in each iteration. Nonlinear optimization is performed on a reduced parameter space of three in the noncoplanar case. A detailed initialization algorithm is also provided to obtain starting estimates. Furthermore, they provide an analytical proof of convergence for their algorithm.

Finally, there are several applications in aerospace and military that do not have accurate control points. An initial estimate of the calibration parameters is made with inaccurate world points and accurate image points. For such applications, a robust estimation scheme is required. One such algorithm is given by Haralick and Shapiro (25) to solve the extrinsic parameters only. This algorithm is extended to also compute the focal length.

Since the literature for calibration is vast and varied, we first discuss the camera calibration model and present off-line methods of computing some intrinsic parameters in the section entitled "Camera Calibration Model." We next consider noncoplanar camera calibration algorithms that employ (1) linear optimization methods to produce suboptimal solutions in the section entitled "Linear Methods of Noncoplanar Camera Calibration," (2) nonlinear optimization methods to produce optimal solutions in the section entitled "Nonlinear Methods of Noncoplanar Camera Calibration," and (3) robust estimation methods to solve a subset of calibration parameters in the section entitled "Robust Methods of Noncoplanar Camera Calibration." Finally, we discuss algorithms for coplanar camera calibration in the section entitled "Coplanar Camera Calibration."

## CAMERA CALIBRATION MODEL

In this discussion, we consider at least 11 calibration parameters shown in Table 1. The extrinsic parameters consist of the $3 \times 3$ rotation matrix $R$ for the camera orientation, which can be alternatively specified by the three Euler angles $\omega$, $\phi$, and $\kappa$ that designate rotations around the $X$, $Y$, and $Z$ world axes, respectively (see Fig. 1). Besides, the $3 \times 1$ translation vector $\boldsymbol{t}$ denotes the position of the camera center. The intrinsic parameters consist of the effective focal length $f$ of the camera, center of the image array $(i_0, j_0)$, horizontal scale factor $s$ of the image array, skew angle $\theta$ between the image axes, and radial, tangential, and thin prism lens distortion parameters $\{k_1, k_2, \ldots, k_{r_0}, p_1, p_2, s_1, s_2\}$.

**Table 1. Camera Calibration Parameters Discussed in This Study**

| Parameters | Type | Description |
|---|---|---|
| $R$ | Extrinsic | Rotation matrix for camera orientation |
| $\boldsymbol{t}$ | | Translation vector for position of camera center |
| $f$ | Intrinsic | Focal length |
| $i_0, j_0$ | Intrinsic | Image center displacement |
| $s$ | | Scale factor |
| $\theta$ | | Skew angle |
| $k_1, \ldots, k_{r_0}$ | Intrinsic | Radial lens distortion |
| $p_1, p_2$ | | Decentering lens distortion |
| $s_1, s_2$ | | Thin prism lens distortion |

### Extrinsic Parameters

Here we describe the geometry of the calibration system. The geometry involves three coordinate systems (see Fig. 1): (1) a 3-D world coordinate system $(X_w, Y_w, Z_w)$ centered on a point $O_w$ and including a point $\boldsymbol{p} = (x, y, z)$, (2) a camera coordinate system $(X_c, Y_c, Z_c)$ with origin at optical center $O_c$ with $Z_c$ axis the same as the optical axis, and (3) a 2-D image array system $(I, J)$ centered at a point $O_s$ in the image, with $(I, J)$ axes aligned to $(X_c, Y_c)$ respectively, and including a point $(i, j)$. Let $f$ be the effective focal length of the camera. The collinearity condition equations are (13,15,19)

$$f\left(\frac{\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) = i \quad \text{and} \quad f\left(\frac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) = j \tag{1}$$

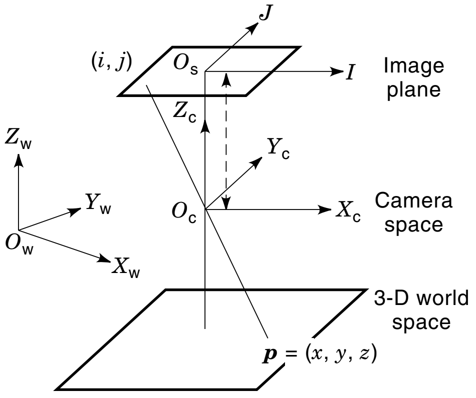

**Figure 1.** Mapping of a 3-D world point $(x, y, z)$ to image point $(i, j)$.

where $R^T = [\boldsymbol{r}_1\ \boldsymbol{r}_2\ \boldsymbol{r}_3]$ is a 3 × 3 rotation matrix defining the camera orientation and $\boldsymbol{t}^T = [t_1\ t_2\ t_3]$ is a translation vector defining the camera position. A very important constraint in calibration algorithms is the *orthonormality constraint* of the rotation matrix $R$ given by

$$R^T R = R R^T = I \tag{2}$$

that is,

$$\|\boldsymbol{r}_1\|^2 = \|\boldsymbol{r}_2\|^2 = \|\boldsymbol{r}_3\|^2 = 1 \quad \text{and} \quad \boldsymbol{r}_1^T\boldsymbol{r}_2 = \boldsymbol{r}_2^T\boldsymbol{r}_3 = \boldsymbol{r}_3^T\boldsymbol{r}_1 = 0$$

Alternatively, the rotation matrix $R$ can be represented by the Euler angles $\omega$, $\phi$, and $\kappa$ that denote rotations around the $X_w$, $Y_w$, and $Z_w$ world axes, respectively:

$$R = \begin{bmatrix} \cos\phi\cos\kappa & \sin\omega\sin\phi\cos\kappa & -\cos\omega\sin\phi\cos\kappa \\ & +\cos\omega\sin\kappa & +\sin\omega\sin\kappa \\ -\cos\phi\sin\kappa & -\sin\omega\sin\phi\sin\kappa & \cos\omega\sin\phi\sin\kappa \\ & +\cos\omega\cos\kappa & +\sin\omega\cos\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{bmatrix} \tag{3}$$

### Image Center and Scale Factor Parameters

Ideally the image center is the intersection of the optical axis of the camera–lens system with the camera's sensing plane. For real lenses, optical axis is not so easily defined, and different definitions of image centers (26) depend on whether the lens has fixed or variable parameters and on how the variable parameters are mechanically implemented. For example, for a simple lens, there can be two axes of symmetry: optical and mechanical. The optical axis is the straight line joining the centers of curvature of the two surfaces of the lens, whereas the mechanical axis is determined by the centerline of the machine used to grind the lens' edge. The angle between these axes is called *decentration* (26,27). In a compound lens, the optical axes of multiple lens elements may not be accurately aligned due to decentration of each lens element, resulting in multiple possibilities for the optical axis. In adjustable and variable focal length lenses, the misalignment between the optical and mechanical axes change as the spacing between the lens elements are changed.

We found several methods to measure image center by using special techniques. Examples of such methods are as follows: (a) measuring the center of the radial lens distortion (10,26), (b) determining the normal projection of a viewing point onto the imaging plane (26,28), (c) measuring the center of the camera's field of view (26), (d) passing a laser beam through the lens assembly and matching the reflection of the beam from the lens with the center of the light spot in the image (10,26), (e) measuring the center of cos$^{4\text{th}}$ radiometric falloff or the center of vignetting/image spot (26), and (f) changing the focal length of a camera–lens system to determine image center from a point invariant in the image (10,26). Besides, there are several algorithmic methods of computing image center as a part of the complete calibration algorithm (1,8,10,11,26,28,29).

Other intrinsic parameters commonly considered are the scale factors $s_i$ and $s_j$ in the $I$ and $J$ image directions, respectively. Array sensors such as CCD/CID sensors acquire the video information line by line, where each line of video signal is well separated by the horizontal sync of the composite video. Usually the vertical spacing between lines perfectly matches that on the sensor array, giving us no scale factor in the vertical direction; that is, $s_j = 1$. The pixels in each line of video signal are resampled by the ADC, which, in reality, samples the video lines with a rate different from the camera and causes the image to be scaled along the horizontal direction; that is, $s_i \neq 1$. Hence, the problem of determining scale factor $s = s_i$.

Some researchers (9,10,26) have suggested that the horizontal scale factor $s$ can be approximately determined from the ratio of the number of sensor elements in the $I$ image direction, to the number of pixels in a line as sampled by the processor. However, due to timing errors, inconsistency of the ADC, and possible tilt of the sensor array, this is not so accurate. A more accurate estimate of $s$ is the ratio of the frequency that sensor elements are clocked off of the CCD to the frequency at which the ADC samples.

We found a number of methods to compute the horizontal scale factor $s$ by special techniques. Examples are as follows: (1) measuring the frequency of the stripes generated by the interface of ADC-clock and camera-clock that create the scale factor problem (10), (2) measuring the distortion in an image of a perfect circle into an ellipse (30), (3) computing power spectrum of the image of two sets of parallel lines (17), and (4) counting the grid points in an image of a grid pattern (31).

Consider an image point $(i_f, j_f)$ with respect to the center $O_s$ of the image buffer. Let the actual image center be at $(i_0, j_0)$. Let $(i_d, j_d)$ be the location of the point in the image with respect to $(i_0, j_0)$. We obtain (1,2,9–11,25)

$$i_d = s^{-1}(i_f - i_0) \quad \text{and} \quad j_d = (j_f - j_0) \tag{4}$$

### Lens Distortion Parameters

As a result of imperfections in the design and assembly of lenses, the image of a plane object lies, in general, on a slightly curved field (15) (see Fig. 2), wherein objects at the edge of the field of view appear somewhat smaller or larger than they should. Types of lens distortions commonly seen are radial (15,32) and tangential (15,33). Two common radial distortions are pincushion and barrel distortions. Pincushion distortion results, for example, when a lens is used as a magnifying glass, whereas barrel distortion results when the object is viewed through a lens at some distance from the eye.

Tangential distortions are usually caused by (1) decentering of the lens (decentering distortion) (11,12,21,32,33), and (2) imperfections in lens manufacturing or tilt in camera sensor or lens (thin prism distortion) (11,12). One of the effects of tangential distortion is that a straight line passing

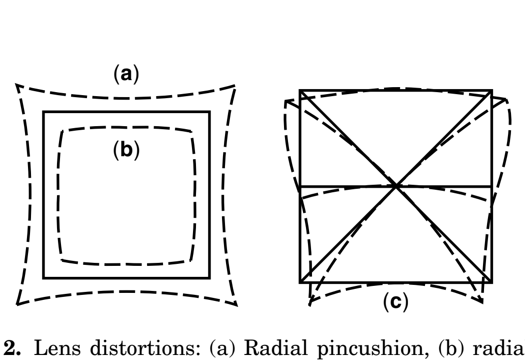

**Figure 2.** Lens distortions: (a) Radial pincushion, (b) radial barrel, and (c) tangential.

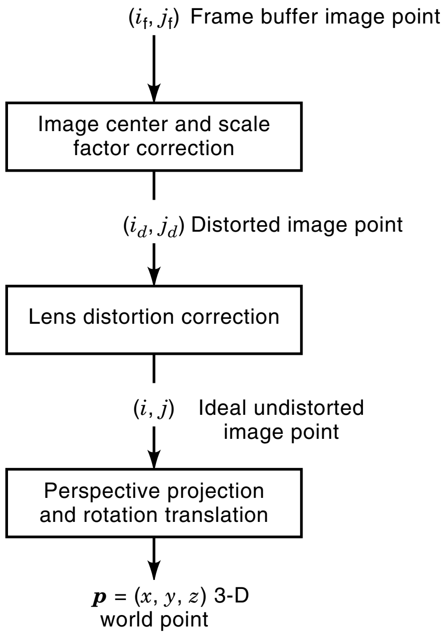
$(i_f, j_f)$ Frame buffer image point
Image center and scale factor correction
$(i_d, j_d)$ Distorted image point
Lens distortion correction
$(i, j)$ Ideal undistorted image point
Perspective projection and rotation translation
$\boldsymbol{p} = (x, y, z)$ 3-D world point

**Figure 3.** Steps for transforming frame buffer image point $(i_f, j_f)$ to ideal undistorted image point $(i, j)$ and to 3-D world point $\boldsymbol{p} = (x, y, z)$.

through the center of the field of view may appear in the image as a weakly curved line (see Fig. 2). Clearly these distortions are disturbing in applications where the ultimate task is to map a 3-D object in uniform scale from its acquired image.

Many researchers (10,11,34,35) have observed that ignoring lens distortion is unacceptable in doing 3-D measurements. Although several studies (9,10,21,30,36,37) have considered only radial distortions up to the first or second order, some studies (11,12,16,20,32,33) have considered both radial and tangential lens distortions, and have used nonlinear optimization schemes to compute them. For example Beyer (20) has demonstrated the effects of higher-order radial and tangential distortion models. By using a first-order radial model, an accuracy in image space of 1/7th of the pixel spacing is obtained. By using a third-order radial and first-order decentering distortion model, this accuracy is enhanced to 1/46th of the pixel spacing. Faugeras (1) and Weng et al. (11) used wide-angle lenses and also found that adding nonradial distortion components improved accuracy. Using a $f$ = 8.5 mm wide-angle lens with both radial and tangential models, Weng et al. (11) demonstrated a significant improvement in image error. This is also supported by our experiments.

One commonly used model for correcting lens distortion is that developed by Brown (32,33). Let $(d_i, d_j)$ be the corrections for geometric lens distortions present in image coordinates $(i_d, j_d)$ respectively, where $(i_d, j_d)$ are obtained from Eq. (4). Let $(i, j)$ be the ideal undistorted image coordinates of a 3-D point $\boldsymbol{p} = (x, y, z)$. With $r^2 = (i^2 + j^2)$, $d_i$ and $d_j$ are expressed by the following series (11,15,32,33):

$$\begin{aligned} d_i &= i(k_1 r^2 + k_2 r^4 + \cdots + k_{r_0} r^{2r_0}) \\ &\quad + (p_1(r^2 + 2i^2) + 2p_2 ij)(1 + p_3 r^2 + \cdots) + (s_1 r^2 + \cdots) \\ d_j &= j(k_1 r^2 + k_2 r^4 + \cdots + k_{r_0} r^{2r_0}) \\ &\quad + (2p_1 ij + p_2(r^2 + 2j^2))(1 + p_3 r^2 + \cdots) + (s_2 r^2 + \cdots) \end{aligned} \tag{5}$$

Here $r_0$ is the order of the radial distortion model. Terms including coefficients $(k_1, k_2, k_3, \ldots, k_{r_0})$ account for radial distortion, $(p_1, p_2, p_3, \ldots)$ represent decentering distortions, and $(s_1, s_2, \ldots)$ represent thin prism distortions. Image coordinates are corrected for lens distortion by the expression below:

$$i = i_d + d_i \quad \text{and} \quad j = j_d + d_j \tag{6}$$

### Discussions

Following the above description of calibration parameters, Fig. 3 below provides the steps for transforming the frame buffer image point $(i_f, j_f)$ to the ideal undistorted image point $(i, j)$ and to the 3-D world point $\boldsymbol{p} = (x, y, z)$.

## LINEAR METHODS OF NONCOPLANAR CAMERA CALIBRATION

In this section, we present four well-known linear methods of camera calibration. These methods are due to (a) Yakimovsky and Cunningham (7), (b) Ganapathy (2), (c) Grosky and Tamburino (4), and (d) Tsai and Lenz (9,10). In this section, we discuss all methods for the noncoplanar case only. The coplanar case is treated separately in the section entitled "Coplanar Camera Calibration."

### Yakimovsky and Cunningham's Method

One of the earliest methods for camera calibration that employs a linear estimation technique is due to Yakimovsky and Cunningham (7). The collinear condition equations in Eq. (1) can be written as the following linear equation with unknown parameter vector $\boldsymbol{b}$:

$$\begin{bmatrix} \boldsymbol{p}^T & \boldsymbol{0} & 1 & 0 & -i_f \boldsymbol{p}^T \\ \boldsymbol{0} & \boldsymbol{p}^T & 0 & 1 & -j_f \boldsymbol{p}^T \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} i_f \\ j_f \end{bmatrix} \tag{7}$$

where

$$\boldsymbol{b}^T = [f\boldsymbol{r}_1^T \quad f\boldsymbol{r}_2^T \quad ft_1 \quad ft_2 \quad \boldsymbol{r}_3^T]/t_3$$

Given at least five linearly independent control points $\boldsymbol{p}$ and corresponding image points $(i, j)$, the above equation can be solved by the method of linear least squares. The axial vector $\boldsymbol{r}_3$ and the depth component of the translation vector $t_3$ are solved by imposing the constraint $\|\boldsymbol{r}_3\| = 1$. Let $\boldsymbol{b}^T = [\boldsymbol{b}_1^T\, \boldsymbol{b}_2^T\, b_3\, b_4\, \boldsymbol{b}_5^T]$. Then $\boldsymbol{r}_3 = \boldsymbol{b}_5/\|\boldsymbol{b}_5\|$ and $|t_3| = 1/\|\boldsymbol{b}_5\|$. As drawn in Fig. 1, the sign of $t_3$ is negative (positive) if the origin of the world coordinate system $O_w$ is in front of (behind) the camera. An algorithmic method to determine the sign of $t_3$ is given in Ref. 9. Estimates of focal length $f$ can be obtained by imposing the constraints $\|\boldsymbol{r}_1\| = 1$ or $\|\boldsymbol{r}_2\| = 1$ as $f = \|\boldsymbol{b}_1\|/\|\boldsymbol{b}_5\|$ or $f = \|\boldsymbol{b}_2\|/\|\boldsymbol{b}_5\|$. However, this produces two estimates of focal length $f$, which cannot be resolved.

Although the method is fast, it ignores the image center, scale factor, and lens distortion parameters. Furthermore, the method fails to impose the orthonormality constraints given in Eq. (2). The constraints $\|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\| = \|\boldsymbol{r}_3\| = 1$ are imposed after the solution is obtained, and thus the solution is suboptimal, and hence prone to errors due to noise in calibration data.

A modification to this method that imposes the constraint $\|\boldsymbol{r}_3\| = 1$ during the least-squares solution can be obtained by reformulating Eq. (7) as follows:

$$\begin{bmatrix} \boldsymbol{p}^{\mathrm{T}} & \mathbf{0} & 1 & 0 & -i_{\mathrm{f}} & -i_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \\ \mathbf{0} & \boldsymbol{p}^{\mathrm{T}} & 0 & 1 & -j_{\mathrm{f}} & -j_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \end{bmatrix} \boldsymbol{b} = 0 \tag{8}$$

where

$$\boldsymbol{b}^{\mathrm{T}} = [f\boldsymbol{r}_1^{\mathrm{T}} \quad f\boldsymbol{r}_2^{\mathrm{T}} \quad ft_1 \quad ft_2 \quad t_3 \quad \boldsymbol{r}_3^{\mathrm{T}}]$$

The constrained least-squares problem associated with Eq. (8) is

$$\text{Minimize } \boldsymbol{b}^{\mathrm{T}}A\boldsymbol{b} \text{ subject to the constraint } \|\boldsymbol{r}_3\|^2 = 1 \tag{9}$$

Here $A$ is the $12 \times 12$ data matrix constructed from the control points and corresponding image points from Eq. (8). Let

$$A = \begin{bmatrix} A_1 & A_2 \\ A_2^{\mathrm{T}} & A_3 \end{bmatrix}$$

be a partition of matrix $A$, where $A_1 \in \Re^{9\times 9}$, $A_2 \in \Re^{9\times 3}$ and $A_3 \in \Re^{3\times 3}$. Then Eq. (9) leads to the following $3 \times 3$ symmetric eigenvalue problem for the solution of $\boldsymbol{r}_3$ (see Ref. 22):

$$(A_3 - A_2^{\mathrm{T}}A_1^{-1}A_2)\boldsymbol{r}_3 = \lambda\boldsymbol{r}_3 \tag{10}$$

Here $\boldsymbol{r}_3$ is the eigenvector corresponding to the minimum eigenvalue $\lambda$. The remaining parameters are

$$[f\boldsymbol{r}_1^{\mathrm{T}} \quad f\boldsymbol{r}_2^{\mathrm{T}} \quad ft_1 \quad ft_2 \quad t_3] = -A_1^{-1}A_2\boldsymbol{r}_3 \tag{11}$$

### Ganapathy's Method

Ganapathy (2) extended the Yakimovsky and Cunningham method to also include the image center and scale factor parameters. From Eqs. (1) and (4), we obtain the collinear condition equations below:

$$f\left(\frac{\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) = s^{-1}(i_{\mathrm{f}} - i_0) \quad \text{and} \quad f\left(\frac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) = (j_{\mathrm{f}} - j_0) \tag{12}$$

From Eq. (13), we obtain the following linear equation in unknown parameter vector $\boldsymbol{b}$:

$$\begin{bmatrix} \boldsymbol{p}^{\mathrm{T}} & \mathbf{0} & 1 & 0 & -i_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \\ \mathbf{0} & \boldsymbol{p}^{\mathrm{T}} & 0 & 1 & -j_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \end{bmatrix} \boldsymbol{b} = 0 \begin{bmatrix} i_{\mathrm{f}} \\ j_{\mathrm{f}} \end{bmatrix} \tag{13}$$

where

$$\boldsymbol{b}^{\mathrm{T}} = [sf\boldsymbol{r}_1^{\mathrm{T}} + i_0\boldsymbol{r}_3^{\mathrm{T}} \quad f\boldsymbol{r}_2^{\mathrm{T}} + j_0\boldsymbol{r}_3^{\mathrm{T}} \quad sft_1 + i_0t_3 \quad ft_2 + j_0t_3 \quad \boldsymbol{r}_3^{\mathrm{T}}]/t_3$$

Let $\boldsymbol{b}^{\mathrm{T}} = [\boldsymbol{b}_1^{\mathrm{T}}\ \boldsymbol{b}_2^{\mathrm{T}}\ b_3\ b_4\ \boldsymbol{b}_5^{\mathrm{T}}]$. The, by imposing the orthonormality conditions of $(\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3)$, we obtain

$$i_0 = \frac{\boldsymbol{b}_1^{\mathrm{T}}\boldsymbol{b}_5}{\|\boldsymbol{b}_5\|^2}, \qquad j_0 = \frac{\boldsymbol{b}_2^{\mathrm{T}}\boldsymbol{b}_5}{\|\boldsymbol{b}_5\|^2}$$

$$f = \frac{\|\boldsymbol{b}_2 - j_0\boldsymbol{b}_5\|}{\|\boldsymbol{b}_5\|}, \qquad s = \frac{\|\boldsymbol{b}_1 - i_0\boldsymbol{b}_5\|}{\|\boldsymbol{b}_2 - j_0\boldsymbol{b}_5\|}$$

$$\boldsymbol{r}_1 = \frac{\boldsymbol{b}_1 - i_0\boldsymbol{b}_5}{sf\|\boldsymbol{b}_5\|}, \qquad \boldsymbol{r}_2 = \frac{\boldsymbol{b}_2 - j_0\boldsymbol{b}_5}{f\|\boldsymbol{b}_5\|}, \qquad \boldsymbol{r}_3 = \frac{\boldsymbol{b}_5}{\|\boldsymbol{b}_5\|}$$

$$t_1 = \frac{\boldsymbol{b}_3 - i_0}{sf\|\boldsymbol{b}_5\|}, \qquad t_2 = \frac{\boldsymbol{b}_4 - j_0}{f\|\boldsymbol{b}_5\|}, \quad \text{and} \quad |t_3| = \frac{1}{\|\boldsymbol{b}_5\|}$$

Once again, the method can be extended to include the constraint $\|\boldsymbol{r}_3\| = 1$ during the least-squares solution.

The method is fast and includes the image center and scale factor parameters. However, it does not include the lens distortion parameters. Moreover, the method imposes the orthonormality constraints after the solution is obtained, and hence the solution is suboptimal. Furthermore, since there are five components of $\boldsymbol{b}$ with six orthonormality constraints [see Eq. (2)], whereas there are 10 unknowns $\{\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3, t_1, t_2, t_3, f, i_0, j_0, s\}$, we obtain ambiguous solutions. For example, $\boldsymbol{b}_1^{\mathrm{T}}\boldsymbol{b}_2/\|\boldsymbol{b}_5\|^2 = i_0j_0$, which gives us two solutions for $i_0$:

$$i_0 = \frac{\boldsymbol{b}_1^{\mathrm{T}}\boldsymbol{b}_5}{\|b_5\|^2} \quad \text{and} \quad i_0 = \frac{\boldsymbol{b}_1^{\mathrm{T}}\boldsymbol{b}_2}{b_2^{\mathrm{T}}b_5}$$

which may not have the same values. Besides, the method treats the different variables as linearly independent when in reality they are not.

### Grosky and Tamburino's Method

Grosky and Tamburino (4) extended Ganapathy's method to obtain a unique linear solution for the noncoplanar camera calibration problem. In order to remove the ambiguity in Ganapathy's solution, Grosky and Tamburino introduced a skew angle parameter $\theta$ which is the positive angle by which the $I$ image coordinate is skewed from $J$. From Eqs. (1) and (4), we obtain the collinearity condition equations:

$$f\left(\frac{\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right)\cos(\theta) + f\left(\frac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right)\sin(\theta)$$
$$= s^{-1}(i_{\mathrm{f}} - i_0) \quad \text{and} \quad f\left(\frac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) = (j_{\mathrm{f}} - j_0) \tag{14}$$

From Eq. (14), we obtain the following linear equations in unknown parameter vector $\boldsymbol{b}$:

$$\begin{bmatrix} \boldsymbol{p}^{\mathrm{T}} & \mathbf{0} & 1 & 0 & -i_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \\ \mathbf{0} & \boldsymbol{p}^{\mathrm{T}} & 0 & 1 & -j_{\mathrm{f}}\boldsymbol{p}^{\mathrm{T}} \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} i_{\mathrm{f}} \\ j_{\mathrm{f}} \end{bmatrix} \tag{15}$$

where

$$\boldsymbol{b}^{\mathrm{T}} = [sf(\boldsymbol{r}_1^{\mathrm{T}}\cos\theta + \boldsymbol{r}_2^{\mathrm{T}}\sin\theta) + i_0\boldsymbol{r}_3^{\mathrm{T}} \quad f\boldsymbol{r}_2^{\mathrm{T}} + j_0\boldsymbol{r}_3^{\mathrm{T}}$$
$$sf(t_1\cos\theta + t_2\sin\theta) + i_0t_3 \quad ft_2 + j_0t_3 \quad \boldsymbol{r}_3^{\mathrm{T}}]/t_3$$

Let $\boldsymbol{b}^{\mathrm{T}} = [\boldsymbol{b}_1^{\mathrm{T}}\ \boldsymbol{b}_2^{\mathrm{T}}\ b_3\ b_4\ \boldsymbol{b}_5^{\mathrm{T}}]$. Then, we obtain the following solution for the calibration parameters by imposing the orthonormality conditions of ($\boldsymbol{r}_1$, $\boldsymbol{r}_2$, $\boldsymbol{r}_3$):

$$
\begin{aligned}
|t_3| &= \frac{1}{\|\boldsymbol{b}_5\|}, \qquad i_0 = \frac{\boldsymbol{b}_1^{\mathrm{T}}\boldsymbol{b}_5}{\|\boldsymbol{b}_5\|^2}, \qquad j_0 = \frac{\boldsymbol{b}_2^{\mathrm{T}}\boldsymbol{b}_5}{\|\boldsymbol{b}_5\|^2} \\
f &= \frac{\|\boldsymbol{b}_2 - j_0\boldsymbol{b}_5\|}{\|\boldsymbol{b}_5\|}, \qquad s = \frac{\|\boldsymbol{b}_1 - i_0\boldsymbol{b}_5\|}{\|\boldsymbol{b}_2 - j_0\boldsymbol{b}_5\|} \\
\sin\theta &= \frac{(\boldsymbol{b}_1 - i_0\boldsymbol{b}_5)^{\mathrm{T}}(\boldsymbol{b}_2 - j_0\boldsymbol{b}_5)}{sf^2\|\boldsymbol{b}_5\|^2} \qquad (16) \\
\boldsymbol{r}_1 &= \frac{\boldsymbol{b}_1 - i_0\boldsymbol{b}_5}{sf\|\boldsymbol{b}_5\|}\sec\theta - \frac{\boldsymbol{b}_2 - j_0\boldsymbol{b}_5}{f\|\boldsymbol{b}_5\|}\tan\theta, \qquad \boldsymbol{r}_2 = \frac{\boldsymbol{b}_2 - j_0\boldsymbol{b}_5}{f\|\boldsymbol{b}_5\|} \\
t_1 &= \frac{b_3 - i_0}{sf\|\boldsymbol{b}_5\|}\sec\theta - \frac{b_4 - j_0}{f\|\boldsymbol{b}_5\|}\tan\theta, \qquad t_2 = \frac{b_4 - j_0}{f\|\boldsymbol{b}_5\|}
\end{aligned}
$$

Once again, the method can be extended to include the constraint $\|\boldsymbol{r}_3\| = 1$ during the least-squares solution of Eq. (15).

The method is computationally efficient, solves the image center and scale factor parameters, and satisfies the orthonormality conditions given in Eq. (2). However, the method does not include the lens distortion parameters, and it produces a suboptimal solution since the constraints are imposed after the solution is obtained. Moreover, the method also treats different variables as linearly independent when they are not.

### Tsai and Lenz's Method

Tsai (9) and Lenz and Tsai (10) designed a method that includes the radial lens distortion parameters in their camera model and provides a two-stage algorithm for estimating the camera parameters. In their method, most parameters are computed in a closed form, and a small number of parameters such as focal length, the depth component of the translation vector, and the radial lens distortion parameters are computed by an iterative scheme.

Tsai (9) introduced an radial alignment constraint (RAC) which results from the observation that the vectors ($i_{\mathrm{d}}$, $j_{\mathrm{d}}$), ($i$, $j$), and ($\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1$, $\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2$) in Fig. 1 are radially aligned when the image center is chosen correctly. Algebraically, the RAC states that

$$
(i_{\mathrm{d}}, j_{\mathrm{d}}) /\!/ (i, j) /\!/ (\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1, \boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2) \qquad (17)
$$

If we determine the image center parameters ($i_0$, $j_0$) by any one of the methods described in the section entitled "Image Center and Scale Factor Parameters," then the frame buffer image coordinates ($i_{\mathrm{f}}$, $j_{\mathrm{f}}$) can be corrected for image center by using Eq. (4) to obtain the distorted image coordinates $si_{\mathrm{d}} = i_{\mathrm{f}} - i_0$ and $j_{\mathrm{d}} = j_{\mathrm{f}} - j_0$. Using first-order radial lens distortion, the collinearity condition equations are

$$
\begin{aligned}
sf\left(\frac{\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) &= si_{\mathrm{d}}(1 + k_1 r_{\mathrm{d}}^2) \quad \text{and} \\
f\left(\frac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} + t_3}\right) &= j_{\mathrm{d}}(1 + k_1 r_{\mathrm{d}}^2)
\end{aligned} \qquad (18)
$$

where $r_{\mathrm{d}}^2 = i_{\mathrm{d}}^2 + j_{\mathrm{d}}^2$. Dividing the first equation by the second, we obtain

$$
\frac{s\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + st_1}{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2} = \frac{si_{\mathrm{d}}}{j_{\mathrm{d}}} \qquad (19)
$$

From Eq. (19), we obtain the following linear equation with unknown parameter vector $\boldsymbol{b}$:

$$
[j\boldsymbol{p}^{\mathrm{T}} \;\; -i\boldsymbol{p}^{\mathrm{T}} \;\; j]\boldsymbol{b} = [i], \qquad \text{where } \boldsymbol{b}^{\mathrm{T}} = [s\boldsymbol{r}_1^{\mathrm{T}} \;\; \boldsymbol{r}_2^{\mathrm{T}} st_1]/t_2 \qquad (20)
$$

Let $\boldsymbol{b}^{\mathrm{T}} = [\boldsymbol{b}_1^{\mathrm{T}}\ \boldsymbol{b}_2^{\mathrm{T}}\ b_3]$. We next perform the following steps:

1. Solve Eq. (20) by using the linear least squares method to compute $\boldsymbol{b}$.
2. Compute $|t_2| = 1/\|\boldsymbol{b}_2\|$ by imposing the constraint $\|\boldsymbol{r}_2\| = 1$. The sign of $t_2$ is determined from Eq. (19).
3. Compute scale factor $s = \|\boldsymbol{b}_1\|/\|\boldsymbol{b}_2\|$ by imposing the constraint $\|\boldsymbol{r}_1\| = 1$.
4. Ignoring lens distortion (i.e., $k_1 = 0$), compute approximate values of focal length $f$ and depth component of translation vector $t_3$ from the following equation [derived from the second equation in Eq. (18)] by using the least-squares method:

$$
[(\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2) \;\; -j_{\mathrm{d}}]\begin{bmatrix} f \\ t_3 \end{bmatrix} = [j_{\mathrm{d}}\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}]
$$

5. Compute exact values of $f$, $t_3$, and radial lens distortion parameter $k_1$ from the following equation [derived from the second equation in Eq. (18)] by using a standard nonlinear optimization scheme such as steepest descent:

$$
f(\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2) - t_3 j_{\mathrm{d}} - k_1(j_{\mathrm{d}} r_{\mathrm{d}}^2 \boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}) - k_1 t_3 (j_{\mathrm{d}} r_{\mathrm{d}}^2) - j_{\mathrm{d}}\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} = 0
$$

where $r_{\mathrm{d}}^2 = i_{\mathrm{d}}^2 + j_{\mathrm{d}}^2$.

The method is fast and also includes the radial lens distortion parameters. This makes the method widely applicable to a variety of applications. However, the solution is suboptimal and does not impose the orthonormality constraints given in Eq. (2). Furthermore, the image center parameters are not included in the solution. Lenz and Tsai (10) proposed a nonlinear method for solving the image center parameters by minimizing the RAC residual error. However, this computation is applicable after all extrinsic parameters and focal length are computed by the above-mentioned steps. By taking the ratio of the collinearity condition equations [see Eq. (19)], the method considers only the tangential component of the collinearity equations and ignores the radial component. Furthermore, the tangential lens distortion parameters are ignored.

## NONLINEAR METHODS OF NONCOPLANAR CAMERA CALIBRATION

The nonlinear methods of noncoplanar camera calibration first originated in the photogrammetry literature, and they were later refined in computer vision research. We present three methods of nonlinear camera calibration. These are (a) nonlinear least-squares method (25), (b) Weng et al.'s method (11), and (c) Chatterjee, Roychowdhury, and Chong's method (22). These methods are computationally more intensive than the linear methods discussed before, but they lead to very accurate estimates of the parameters. We shall consider only the noncoplanar case in this section, and the coplanar case is considered separately in the section entitled "Coplanar Camera Calibration."

### Nonlinear Least-Squares Method

In this formulation of the camera calibration problem, the representation of the rotation matrix $R$ in terms of the Euler angles $\omega$, $\phi$, and $\kappa$ as given in Eq. (3) is used. Substituting this $R$ and the image coordinate corrections [see Eqs. (4) and (6)] in the collinearity condition equations [Eq. (1)], we obtain two nonlinear equations below (for first-order radial lens distortion only):

$$i_{\mathrm{f}} = i_0 + sf\left(\frac{\boldsymbol{r}_1(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_1}{\boldsymbol{r}_3(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_3}\right) + (i_{\mathrm{f}}-i_0)k_1 r(s,i_0,j_0)^2$$

and

$$j_{\mathrm{f}} = j_0 + f\left(\frac{\boldsymbol{r}_2(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_2}{\boldsymbol{r}_3(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_3}\right) + (j_{\mathrm{f}}-j_0)k_1 r(s,i_0,j_0)^2$$

where $\boldsymbol{r}(s, i_0, j_0)^2 = s^{-2}(i_{\mathrm{f}} - i_0)^2 + (j_{\mathrm{f}} - j_0)^2$.

The unknowns are the parameters of exterior orientation $\{\omega, \phi, \kappa, t_1, t_2, t_3\}$ and interior orientation $\{f, s, i_0, j_0, k_1\}$. These equations are of the following general form:

$$\alpha_i = g_i(\beta_1,\ldots,\beta_{\mathrm{M}}) + \xi_i \quad \text{for } i=1,2$$

where $\beta_1, \ldots, \beta_{\mathrm{M}}$ are the unknown parameters, $\alpha_1$ and $\alpha_2$ are the observations, and $\xi_1$ and $\xi_2$ are additive zero mean Gaussian noise having covariance $\Sigma$. The maximum likelihood solution leads to the minimization of the criterion

$$J = (\alpha - g)^{\mathrm{T}}\Sigma^{-1}(\alpha - g) \tag{21}$$

where $\alpha^{\mathrm{T}} = [\alpha_1\ \alpha_2]$ and $g^{\mathrm{T}} = [g_1\ g_2]$. Starting from a given approximate solution $\beta^0 = (\beta_1^0, \ldots, \beta_{\mathrm{M}}^0)^{\mathrm{T}}$, we begin by linearizing the nonlinear transformations about $\beta^0$ and solve for adjustments $\Delta\beta = (\Delta\beta_1, \ldots, \Delta\beta_{\mathrm{M}})^{\mathrm{T}}$, which when added to $\beta$ constitute a better approximate solution. We perform this linearization and adjustment iteratively. If good starting estimates $\beta^0$ are available, in most cases 5 to 10 iterations are required to produce the solution of desired accuracy.

In the $k$th iteration, let $\beta^k$ be the current approximate solution. The linearization proceeds by representing each $g_i(\beta^k + \Delta\beta)$ by a first-order Taylor series expansion of $g_i$ taken around $\beta^k$:

$$g_i(\beta^k + \Delta\beta) = g_i(\beta^k) + \Delta g_i(\Delta\beta;\beta^k) \quad \text{for } i=1,2$$

where $\Delta g_i$, the total derivative of $g_i$, is a linear function of the vector of adjustments $\Delta\beta$ given by

$$\Delta g_i(\Delta\beta;\beta^k) = \left[\frac{\partial g_i}{\partial\beta_1}(\beta^k)\cdots\frac{\partial g_i}{\partial\beta_{\mathrm{M}}}(\beta^k)\right]\Delta\beta \quad \text{for } i=1,2$$

Substituting the linearized expressions into the least-squares criterion [Eq. (21)] and then minimizing it with respect to $\Delta\beta$, we obtain

$$\Delta\beta = (G^{k^{\mathrm{T}}}\Sigma^{-1}G^k)^{-1}G^{k^{\mathrm{T}}}\Sigma^{-1}(\alpha - g^k)$$

where

$$g^k = \begin{bmatrix} g_1(\beta_1^k,\ldots,\beta_M^k) \\ g_2(\beta_1^k,\ldots,\beta_M^k)\end{bmatrix} \quad \text{and}$$

$$G^k = \begin{bmatrix} \partial g_1/\partial\beta_1 & \partial g_1/\partial\beta_2 & \ldots & \partial g_1/\partial\beta_M \\ \partial g_2/\partial\beta_1 & \partial g_2/\partial\beta_2 & \ldots & \partial g_2/\partial\beta_M \end{bmatrix}$$

The method produces very accurate parameter estimates and is less prone to errors in the data when compared with the linear methods. The method solves all calibration parameters, and the orthonormality constraints are imposed during the optimization process. This is an optimal solution to the calibration problem. However, the method is computationally slow and requires at least 5 to 10 iterations to converge, provided that good starting estimates are available. Furthermore, accurate starting estimates are very important for the method to converge to the correct solution. Usually, a linear method such as Grosky and Tamburino's method is used to obtain starting parameter estimates.

### Weng et al.'s Method

Weng et al. (11) described a two-step nonlinear algorithm of camera calibration which first computes a set of starting estimates for all parameters in an initialization algorithm and then uses linear and nonlinear optimization schemes to obtain accurate estimates. Weng et al. also showed that computing all extrinsic parameters with lens distortion is necessary for accurate camera calibration.

Weng et al. presents two sets of parameters: (1) extrinsic and intrinsic nondistortion parameters $\boldsymbol{m} = \{f, s, i_0, j_0, \omega, \phi, \kappa, t_1, t_2, t_3\}$ and (2) set of distortion parameters $\boldsymbol{d} = \{k_1, k_2, p_1, p_2, s_1, s_2\}$. Given control points $\boldsymbol{p}$ and corresponding image points $(i_{\mathrm{f}}, j_{\mathrm{f}})$, they use the following optimization criterion:

$$J(\boldsymbol{m},\boldsymbol{d}) = \sum_{\text{all points}}[(i_{\mathrm{f}} - i(\boldsymbol{m},\boldsymbol{d}))^2 + (j_{\mathrm{f}} - j(\boldsymbol{m},\boldsymbol{d}))^2] \tag{22}$$

where

$$i(\boldsymbol{m},\boldsymbol{d}) = i_0 + sf\left(\frac{\boldsymbol{r}_1(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_1}{\boldsymbol{r}_3(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_3}\right) + (i_f - i_0)k_1 r(s,i_0,j_0)^2 + \text{higher-order lens distortion terms}$$

and

$$j(\boldsymbol{m},\boldsymbol{d}) = j_0 + f\left(\frac{\boldsymbol{r}_2(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_2}{\boldsymbol{r}_3(\omega,\phi,\kappa)^{\mathrm{T}}\boldsymbol{p}+t_3}\right) + (j_f - j_0)k_1 r(s,i_0,j_0)^2 + \text{higher-order lens distortion terms}$$

where $r(s, i_0, j_0)^2 = s^{-2}(i_{\mathrm{f}} - i_0)^2 + (j_{\mathrm{f}} - j_0)^2$.

In the initialization stage, Weng et al. uses two steps shown below:

1. Use a linear algorithm (by ignoring lens distortion) such as Grosky and Tamburino's method (see section entitled "Grosky and Tamburino's Method") to obtain the starting estimates of the extrinsic parameters $\{\omega, \phi, \kappa, t_1, t_2, t_3\}$, focal length $f$, image center parameters $(i_0, j_0)$, and scale factor $s$.

2. Ignoring lens distortions parameters $\boldsymbol{d}$, minimize the optimization criterion [Eq. (22)] with respect to the parameter set $\boldsymbol{m}$ by a nonlinear optimization technique such as Levenberg–Marquardt (23,38).

For the initialization algorithm, only the *central points* are used. Central points are those calibration points close to the center of the image, and they do not contain significant lens distortion effects.

After the initial estimates are obtained, the main estimation algorithm uses iterations between the following two steps:

1. Estimate the distortion parameters $\boldsymbol{d}$ with the nondistortion parameters $\boldsymbol{m}$ held constant by a linear estimation step. Note that in Eq. (22), if the nondistortion parameters $\boldsymbol{m}$ are held constant, then the distortion parameters $\boldsymbol{d}$ can be represented by the following linear equation:

$$\begin{bmatrix} si_{\mathrm{d}}r_{\mathrm{d}}^2 & si_{\mathrm{d}}r_{\mathrm{d}}^4 & s(r_{\mathrm{d}}^2+2i_{\mathrm{d}}^2) & 2si_{\mathrm{d}}j_{\mathrm{d}} & sr_{\mathrm{d}}^2 & 0 \\ j_{\mathrm{d}}r_{\mathrm{d}}^2 & j_{\mathrm{d}}r_{\mathrm{d}}^4 & 2i_{\mathrm{d}}j_{\mathrm{d}} & (r_{\mathrm{d}}^2+2j_{\mathrm{d}}^2) & 0 & r_{\mathrm{d}}^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ p_1 \\ p_2 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} i_{\mathrm{f}} - i_0 - sf\left(\dfrac{\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p}+t_1}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}+t_3}\right) \\ j_{\mathrm{f}} - j_0 - f\left(\dfrac{\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p}+t_2}{\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}+t_3}\right) \end{bmatrix}$$

   Here, the distorted image points $(i_{\mathrm{d}}, j_{\mathrm{d}})$ are obtained from frame buffer points $(i_{\mathrm{f}}, j_{\mathrm{f}})$ by using Eq. (4).

2. Estimate the nondistortion parameters $\boldsymbol{m}$ with the distortion parameters $\boldsymbol{d}$ held constant by nonlinear minimization of the optimization criterion $J(\boldsymbol{m}, \boldsymbol{d})$ in Eq. (22).

The method iterates between the above two steps until a desired accuracy in the estimates is obtained.

The method provides very accurate estimates, solves all calibration parameters, and gives us an optimal solution. However, the method is computationally slow and requires good starting estimates to converge to the correct solution. Since the method also provides an initialization algorithm, reliable starting estimates can be obtained.

### Chatterjee, Roychowdhury, and Chong's Method

This recent technique (22) for camera calibration also offers a two-step algorithm that consists of an initialization phase and a main estimation step. Instead of using the Euler angles ($\omega$, $\phi$, $\kappa$) to represent the rotation matrix $R$, this method instead directly imposes the orthonormality constraints given in Eq. (2). This relieves the objective function of periodic transcendental terms which frequently results in many false minima. Due to the structure of the objective function in the calibration problem, this method employs the *Gauss–Seidel* (23,24) technique of nonlinear optimization. The Gauss–Seidel method for block components iteratively minimizes the objective function for a given block with the remaining held constant, and the most recently computed values of the remaining blocks are used. This method is particularly attractive because of its easy implementation (23,24,38), and it lends itself to a comprehensive theoretical convergence analysis (22,24). These properties are utilized to obtain a simpler solution and also obtain a rigorous convergence study.

The method partitions the parameter set into three blocks: (1) extrinsic parameters and focal length $\boldsymbol{b} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3, t_1, t_2, t_3, f\}$; (2) image center and scale factor parameters $\boldsymbol{m} = \{i_0, j_0, s\}$; and (3) lens distortion parameters $\boldsymbol{d} = \{k_1, k_2, \ldots, k_{r_0}, p_1, p_2, s_1, s_2\}$. In the noncoplanar case, most parameters (blocks $\boldsymbol{b}$ and $\boldsymbol{d}$) are computed by linear iterations or in closed form. Unlike the previous two methods, nonlinear optimization is performed only for the parameter block $\boldsymbol{m}$—that is, in a reduced parameter space of dimension 3.

**Initialization Algorithm.** From Eq. (1), we obtain the following objective function:

$$J(\boldsymbol{b},\boldsymbol{m},\boldsymbol{d}) = \sum_{\text{all points}} [(f\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + ft_1 - \boldsymbol{r}_3^{\mathrm{T}}i\boldsymbol{p} - it_3)^2 + (f\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + ft_2 - \boldsymbol{r}_3^{\mathrm{T}}j\boldsymbol{p} - jt_3)^2] \qquad (23)$$

under constraint $\|\boldsymbol{r}_3\|^2 = 1$. Here $i = i(\boldsymbol{m}, \boldsymbol{d})$ and $j = j(\boldsymbol{m}, \boldsymbol{d})$ by using Eqs. (4) and (6). The initialization algorithm consists of the following steps:

1. Ignoring lens distortion (i.e., $\boldsymbol{d} = \boldsymbol{0}$), an initial estimate of $\boldsymbol{m}$ is obtained by using Grosky and Tamburino's method with only central calibration points.
2. Iterate between the following two steps for all calibration points:
   a. Compute $\boldsymbol{b}$ with $\boldsymbol{d}$ held constant. From Eq. (23), we obtain the following constrained linear equation in terms of $\boldsymbol{b}$:

$$\begin{bmatrix} \boldsymbol{p}^{\mathrm{T}} & \boldsymbol{0} & 1 & 0 & -i & -i\boldsymbol{p} \\ \boldsymbol{0} & \boldsymbol{p}^{\mathrm{T}} & 0 & 1 & -j & -j\boldsymbol{p} \end{bmatrix} \boldsymbol{b} = 0 \text{ under constraint } \|\boldsymbol{r}_3\|^2 = 1$$

   where $\boldsymbol{b}^{\mathrm{T}} = [f\boldsymbol{r}_1^{\mathrm{T}}\ f\boldsymbol{r}_2^{\mathrm{T}}\ ft_1\ ft_2\ t_3\ \boldsymbol{r}_3^{\mathrm{T}}]$. As discussed in the section entitled "Yakimovsky and Cunningham's Method" this problem leads to the constrained least-squares problem given in Eq. (9) whose solutions are given in Eqs. (10) and (11).

   b. Compute $\boldsymbol{d}$ with $\boldsymbol{b}$ held constant. Here $\boldsymbol{d}$ can be solved from the following unconstrained least-squares equation derived from Eq. (23):

$$(\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}+t_3)\begin{bmatrix} ir^2 & ir^4 & \ldots & ir^{2r_0} & r^2+2i^2 & 2ij & r^2 & 0 \\ jr^2 & jr^4 & \ldots & jr^{2r_0} & 2ij & r^2+2j^2 & 0 & r^2 \end{bmatrix} \boldsymbol{d} = \begin{bmatrix} f(\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p}+t_1) - i(\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}+t_3) \\ f(\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p}+t_2) - j(\boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p}+t_3) \end{bmatrix}$$

   where $d = [k_1\ k_2\ .\ .\ .,\ k_{r_0}\ p_1\ p_2\ s_1\ s_2]$

$$r^2 = s^{-2}(i_{\mathrm{f}} - i_0)^2 + (j_f - j_0)^2$$
$$i = s^{-1}(i_{\mathrm{f}} - i_0),\ j = (j_f - j_0)$$

   and $r_0$ is the order of the radial lens distortion model.

**Main Algorithm.** From Eq. (1), we obtain the following objective function:

$$J(\boldsymbol{b},\boldsymbol{d},\boldsymbol{m}) = \sum_{\text{all points}} \left[ \left(\frac{qi}{f} - \boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} - t_1\right)^2 + \left(\frac{qj}{f} - \boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} - t_2\right)^2 + (q - \boldsymbol{r}_3^{\mathrm{T}}\boldsymbol{p} - t_3)^2 \right] \tag{24}$$

under orthonormality constraints [Eq. (2)]. Here $i = i(\boldsymbol{m}, \boldsymbol{d})$ and $j = j(\boldsymbol{m}, \boldsymbol{d})$ by using Eqs. (4) and (6), and $q$ is a depth variable that is also estimated. We define parameter blocks $\boldsymbol{b} = \{R, \boldsymbol{t}, f\}$, $\boldsymbol{d}^{\mathrm{T}} = [k_1\ k_2\ .\ .\ .\ k_{r_0}\ p_1\ p_2\ s_1\ s_2]$ and $\boldsymbol{m}^{\mathrm{T}} = [i_0\ j_0\ s]$. The algorithm starts with an initial estimate of the depth parameter $q$ for each calibration point. The initial values could be the same constant for each point, with the constant representing an initial guess of how far the object is from the camera center. The algorithm then iterates between the following three steps:

1. Compute $\boldsymbol{b}$ with $\boldsymbol{d}$ and $\boldsymbol{m}$ held constant. This step is similar to the exterior orientation or space resection problem (15,25,39) in analytical photogrammetry. For this step, we define $\boldsymbol{v}^{\mathrm{T}} = [i/f\ j/f\ 1]$. We then iterate between the following three steps:

   a. Compute $R$ and $\boldsymbol{t}$ with $f$ and $q$ held constant. From Eq. (24) we obtain the following constrained optimization problem:

$$\text{Minimize} \sum \|q\boldsymbol{v} - R\boldsymbol{p} - \boldsymbol{t}\|^2 \text{ under constraint } R^{\mathrm{T}}R = RR^{\mathrm{T}} = I \tag{25}$$

   This is the well-known problem (13,15,25,40) in photogrammetry, commonly known as the *absolute orientation* problem. In this formulation, $R$ can be obtained from the singular value decomposition of $B = \Sigma(\boldsymbol{p} - \overline{\boldsymbol{p}})(q\boldsymbol{v} - \overline{q\boldsymbol{v}})^{\mathrm{T}}$ which is $B = UDV^{\mathrm{T}}$. We obtain (24,39,40) $R = VU^{\mathrm{T}}$. Here $\overline{\boldsymbol{p}}$ and $\overline{q\boldsymbol{v}}$ are the averages of $\boldsymbol{p}$ and $q\boldsymbol{v}$, respectively, with all calibration points. The translation vector $\boldsymbol{t}$ is obtained from: $\boldsymbol{t} = \overline{q\boldsymbol{v}} - R\overline{\boldsymbol{p}}$.

   b. Compute $q$ with $R$, $\boldsymbol{t}$, and $f$ held constant. From Eq. (25), we obtain the following solution of $q$:

$$q = \frac{(R\boldsymbol{p} + \boldsymbol{t})^{\mathrm{T}}\boldsymbol{v}}{\|\boldsymbol{v}\|^2} \tag{26}$$

   c. Compute $f$ with $R$, $\boldsymbol{t}$, and $q$ held constant. From Eq. (25), we obtain the following solution of $f$:

$$f = \frac{\sum q^2(i^2 + j^2)}{\sum q(\boldsymbol{r}_1^{\mathrm{T}}\boldsymbol{p} + t_1)i + \sum q(\boldsymbol{r}_2^{\mathrm{T}}\boldsymbol{p} + t_2)j} \tag{27}$$

2. Compute $\boldsymbol{d}$ with $\boldsymbol{b}$ and $\boldsymbol{m}$ held constant. This is a linear optimization step similar to the initialization algorithm, Step 2b.

3. Compute $\boldsymbol{m}$ with $\boldsymbol{b}$ and $\boldsymbol{d}$ held constant. This is an unconstrained nonlinear optimization step with three parameters. The objective function $J$ in Eq. (24) is minimized by standard optimization methods such as the Conjugate Direction (31) or Quasi-Newton (31) methods. Needless to say, a proper starting value of $\boldsymbol{m}$, obtained by the initialization algorithm, is essential for accurate convergence of this step.

The method obtains accurate parameter estimates and solves all calibration parameters. The method offers an optimal solution. Although the method reduces the nonlinear optimization to only a dimension of 3, it is still computationally more intensive than the linear optimization methods discussed in the section entitled "Linear Methods of Noncoplanar Camera Calibration." The method also needs good starting estimates to converge to the correct solution. Since this method also provides an initialization algorithm, we can obtain reliable starting estimates. In addition, the method provides a Lyapunov-type convergence analysis of the initialization and main algorithms (22).

## ROBUST METHODS OF NONCOPLANAR CAMERA CALIBRATION

In some applications, we do not have accurate control points to obtain good parameter estimates. Such applications are common in aerospace and military problems, where a camera mounted on a moving airplane estimates the locations of points on the ground. Since ground points may not be available accurately, the algorithm has to adaptively improve the calibration parameter estimates as more points are obtained. We describe a robust estimation technique (41) for such applications. Since estimating all parameters with a robust technique is cumbersome and in most cases unnecessary, we present a method to robustly estimate only the extrinsic parameters and focal length. The method discussed below is only applicable to the noncoplanar camera calibration problem. This method can, however, be easily extended to the coplanar camera calibration problem discussed in the section entitled "Coplanar Camera Calibration."

The standard least-squares techniques are ideal when the random data perturbations or measurement errors are Gaussian distributed. However, when a small fraction of the data have large non-Gaussian perturbations, least-squares techniques become useless. With least-squares techniques, the perturbation of the estimates caused by the perturbation of even a single data component grows linearly in an unbounded way. In order to estimate a parameter $\theta$ from a set of observations $x_1, . . ., x_N$ by the robust method, we use a function $\rho$ whose derivative with respect to $\theta$ is $\psi$ as shown below:

$$J(\theta) = \sum_{n=1}^{N} \rho(x_n - \theta) \quad \text{and} \quad \frac{\partial J(\theta)}{\partial \theta} = \sum_{n=1}^{N} \psi(x_n - \theta)$$

The minimization of $J(\theta)$ is achieved by finding the appropriate $\theta$ that satisfies $\sum_{n=1}^{N} \psi(x_n - \theta) = 0$, where $\psi(x) = \partial\rho(x)/\partial x$. The solution to this equation that minimizes $J(\theta)$ is called the maximum-likelihood or M-estimator of $\theta$. Thus, in robust M-estimation, we determine a $\psi$ function so that the resulting estimator will protect us against a small percentage (say, around 10%) of "outliers." But, in addition, we desire these procedures to produce reasonably good (say, 95% efficient) estimators in case the data actually enjoy the Gaussian assumptions. Here, the 5% efficiency that we sacrifice in these cases is sometimes referred to as the "premium" that we pay to gain all that protection in very non-Gaussian cases.

A scale invariant version of robust M-estimator can also be obtained by finding the solution of

$$\sum_{n=1}^{N} \psi\left(\frac{x_n - \theta}{s}\right) = 0$$

where $s$ is the robust estimate of scale such as

$$s = \frac{\text{median}|x_n - \text{median}(x_n)|}{0.6745}$$

or

$$s = \frac{\text{75th percentile} - \text{25th percentile}}{2(0.6745)}$$

If the sample arises from a Gaussian distribution, then $s$ is an estimate of the standard deviation $\sigma$.

There are several choices of $\psi$ functions given below that are commonly used in literature (36):

*Huber's* $\psi$:

$$\psi(x) = \begin{cases} -a, & x < -1 \\ x, & |x| \le a \\ a, & x > a \end{cases}$$

where the "tuning constant" $a$ equals 1.5.

*Hampel's* $\psi$:

$$\psi(x) = \begin{cases} |x|, & 0 \le |x| < a \\ a, & a \le |x| < b \\ \dfrac{c - |x|}{c - b}, & b \le |x| < c \\ 0, & c \le |x| \end{cases}$$

where reasonably good values are $a = 1.7$, $b = 3.4$, and $c = 8.5$

*Andrew's Sine* $\psi$:

$$\psi(x) = \begin{cases} \sin(x/a), & |x| \le a\pi \\ 0, & |x| > a\pi \end{cases}$$

with $a = 2.1$.

*Tukey's Biweight* $\psi$:

$$\psi(x) = \begin{cases} x[1 - (x/a)^2]^2, & |x| \le a \\ 0, & |x| > a \end{cases}$$

with $a = 6.0$.

Let us mention three iteration schemes that can be used to solve $\Sigma\, \psi[(x_n - \theta)/s] = 0$, where $\psi$ is any one of the functions above. We use the sample median as the initial guess $\theta_0$ of $\theta$. Define $\psi'(x)$ as the derivative of $\psi$ with respect to $x$, and $\theta_k$ as the estimate of $\theta$ at the $k$th iteration of the algorithm. The 3 methods are:

*Newton's method:*

$$\theta_{k+1} = \theta_k + \frac{s \sum_{n=1}^{N} \psi[(x_n - \theta_k)/s]}{\sum_{n=1}^{N} \psi'[x_n - \theta_k)/s]}$$

where the denominator of the second term counts the number of items that enjoy $|x_n - \theta_k|/s \le a$ for Huber's and Tukey's $\psi$, $|x_n - \theta_k| \le a\pi$ for Andrew's Sine $\psi$, and $b \le |x_i - \theta_k| < c$ for Hampel's $\psi$.

*H-Method:*

$$\theta_{k+1} = \theta_k + \frac{s \sum_{n=1}^{N} \psi[(x_n - \theta_k)/s]}{N}$$

where the second term is the average of the pseudo (Winsorized) residuals (that is, "least squares" is applied to the residuals).

*Weighted Least-Squares:*

$$\theta_{k+1} = \frac{\sum_{n=1}^{N} w_n x_n}{\sum_{n=1}^{N} w_n} = \theta_k + \frac{s \sum_{n=1}^{N} \psi[(x_n - \theta_k)/s]}{\sum_{n=1}^{N} w_n}$$

where the weight is

$$w_n = \frac{\psi[(x_n - \theta_k)/s]}{(x_n - \theta_k)/s} \quad \text{for } n = 1, 2, \ldots, N$$

We invoke the objective function (25) to compute robust estimates of the rotation matrix $R$, translation vector $\mathbf{t}$, and focal length $f$. Given world (control) points $\mathbf{p}_n$ and corresponding image points $(i_n, j_n)$ for $i = 1, \ldots, N$, from Eq. (25), we obtain the following constrained robust estimation problem:

$$\text{Minimize} \sum_{n=1}^{N} \rho\left(\frac{\|q_n \mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\|}{s}\right) \quad \text{under constraint } R^T R = RR^T = I \qquad (28)$$

where

$$\mathbf{v}_n^T = [i_n/f \quad j_n/f \quad 1]$$

which leads to the problem

$$\text{Solve} \sum_{n=1}^{N} \psi\left(\frac{\|q_n \mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\|}{s}\right) = 0 \text{ under constraint } R^T R = RR^T = I$$

Since $\psi$ is a nonlinear function and the above equation is difficult to solve, we use the weighted least-squares method to obtain robust estimates of the parameters $R$, $\mathbf{t}$, and $f$.

The method consists of the following steps:

1. Obtain initial estimates of the depth parameter $q_n$ for each calibration point. The initial values could be the same constant for each point, the constant representing an initial guess of how far the object is from the camera center.
2. Iterate between the following three steps:
   A. (Optional) Estimate scale $s$ from

$$s = \underset{\|q_n \mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\| \ne 0}{\mathit{median}} \frac{\|q_n \mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\|}{0.6745}$$

   B. Iterate between the following steps:
      a. Compute $R$ and $\mathbf{t}$ with $f$, $q_n$, and $w_n$ held constant. As described in Section 4.3.1, this is the *abso-*

*lute orientation* problem in photogrammetry (25,39,40). We first compute the matrix $B = \sum_{n=1}^{N} w_n(\mathbf{p}_n - \overline{\mathbf{p}})(q_n\mathbf{v}_n - \overline{q\mathbf{v}})^T$ and then compute its singular value decomposition as $B = UDV^T$. We then obtain (24,39,40): $R = VU^T$. Here, $\overline{\mathbf{p}} = \sum_{n=1}^{N} w_n\mathbf{p}_n / \sum_{n=1}^{N} w_n$ and $\overline{q\mathbf{v}} = \sum_{n=1}^{N} w_n q_n \mathbf{v}_n / \sum_{n=1}^{N} w_n$ are the weighted averages of $\mathbf{p}_n$ and $q_n\mathbf{v}_n$, respectively, for all calibration points. The translation vector $\mathbf{t}$ is obtained from: $\mathbf{t} = \overline{q\mathbf{v}} - R\overline{\mathbf{p}}$.

b. Compute $q_n$ with $R$, $\mathbf{t}$, $f$ and $w_n$ held constant. From Eq. (28), we obtain the following solution of $q_n$ for each calibration point:

$$q_n = \frac{(R\mathbf{p}_n + \mathbf{t})^T\mathbf{v}_n}{\|\mathbf{v}_n\|^2} \quad \text{for } n = 1, \ldots, N \tag{29}$$

c. Compute $f$ with $R$, $\mathbf{t}$, $q_n$, and $w_n$ held constant. From Eq. (28), we obtain the following estimate of $f$:

$$f = \frac{\sum_{n=1}^{N} w_n q_n^2 (i_n^2 + j_n^2)}{\sum_{n=1}^{N} w_n q_n (\mathbf{r}_1^T\mathbf{p}_n + t_1) i_n + \sum_{n=1}^{N} w_n q_n (\mathbf{r}_2^T\mathbf{p}_n + t_2) j_n} \tag{30}$$

where $\mathbf{t}^T = [t_1\ t_2\ t_3]$.

C. Compute weights $w_n$ from:

$$w_n = \frac{\psi(\|q_n\mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\|/s)}{\|q_n\mathbf{v}_n - R\mathbf{p}_n - \mathbf{t}\|/s} \quad \text{for } n = 1, \ldots, N$$

Although most people concerned with robustness do iterate on scale s also (Step A), this step is optional since some problems of convergence may be created.

## COPLANAR CAMERA CALIBRATION

In coplanar camera calibration, the calibration points lie on a plane. Without loss of generality, we assume that it is the $z$ coordinate that is unimportant. Due to this assumption, the last column of the rotation matrix $R$ and the depth component of the translation vector $\boldsymbol{t}$ are not available in the collinearity condition equations [Eq. (1)]. Instead of the six orthonormality constraints described in Eq. (2), we have three constraints:

$$r_{11}^2 + r_{21}^2 + r_{31}^2 = 1, \qquad r_{12}^2 + r_{22}^2 + r_{32}^2 = 1, \quad \text{and} \\ r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} = 0 \tag{31}$$

where $r_{ij}$ is the $j$th component of $\mathbf{r}_i$. We discuss two methods of estimating the parameters in the coplanar case: (1) iterative linear methods that produce a suboptimal solution and (2) nonlinear methods that produce an optimal solution.

### Iterative Linear Methods for Coplanar Camera Calibration

Here we discuss two methods of parameter computation. The first method is due to Tsai (9) and computes all extrinsic and the focal length parameters

$$b = \{\boldsymbol{r}_{11}\boldsymbol{r}_{12}\boldsymbol{r}_{21}, \boldsymbol{r}_{22}, \boldsymbol{r}_{31}, \boldsymbol{r}_{32}, f, t_1, t_2\}$$

and first-order lens distortion parameter $d = \{k_1\}$. This method uses the ratio of the two collinearity condition equations given in Eq. (1). The second method utilizes both collinearity condition equations and is called the "Conventional Method". In both methods we drop the image center and scale factor parameters $m = \{i_0, j_0, s\}$ which may be estimated by several offline methods discussed in the section entitled "Image Center and Scale Factor Parameters".

**Tsai's Coplanar Method.** Let $\mathbf{p} = (x, y, z)$ be a world (control) point with corresponding image point $(i_f, j_f)$. Given image center and scale factor parameters $\{i_0, j_0, s\}$, computed before, we can correct the image point $(i_f, j_f)$ to obtain distorted image point $(i_d, j_d)$ by applying Eq. (4). If we use radial lens distortion only, then the undistorted image point $(i, j)$ is:

$$i = i_d(1 + k_1 r_d^2 + \cdots + k_{r_0} r_d^{2r_0}) \text{ and}$$
$$j = j_d(1 + k_1 r_d^2 + \cdots + k_{r_0} r_d^{2r_0})$$

where $r_d^2 = i_d^2 + j_d^2$ and $k_1, \ldots, k_{r_0}$ are the radial lens distortion parameters. Then, by taking the ratio of the two collinearity condition equations (see Eq. (1)), we obtain:

$$\frac{r_{11}x + r_{12}y + t_1}{r_{21}x + r_{22}y + t_2} = \frac{i}{j} = \frac{i_d}{j_d}$$

where $\{r_{11}, r_{12}, r_{21}, r_{22}, t_1, t_2\}$ are the extrinsic parameters. Tsai's algorithm uses the above equations and consists of the following steps:

1. Compute the extrinsic parameters $\mathbf{b}$ from the following linear equation:

   $$[j_d x \quad j_d y \quad j_d \quad -i_d x \quad -i_d y]\mathbf{b} = [i_d]$$

   where

   $$\mathbf{b}^T = \begin{bmatrix} \dfrac{r_{11}}{t_2} & \dfrac{r_{12}}{t_2} & \dfrac{t_1}{t_2} & \dfrac{r_{21}}{t_2} & \dfrac{r_{22}}{t_2} \end{bmatrix}$$

2. Compute $t_2$ from the following equations:

   $$t_2^2 = \frac{S - \sqrt{S^2 - 4(b_1b_5 - b_4b_2)^2}}{2(b_1b_5 - b_4b_2)^2} \text{ or}$$
   $$|t_2| = \frac{2}{\sqrt{(b_1 + b_5)^2 + (b_2 - b_4)^2} + \sqrt{(b_1 - b_5)^2 + (b_2 + b_4)^2}}$$

   where

   $$\mathbf{b}^T = [b_1 \quad b_2 \quad b_3 \quad b_4 \quad b_5], \text{ and } S = b_1^2 + b_2^2 + b_4^2 + b_5^2$$

3. Determine the sign of $t_2$. If $Sign(b_1x + b_2y + b_3)$ and $Sign(i_d)$ are same then $Sign(t_2) = +1$, otherwise $Sign(t_2) = -1$.
4. Determine the extrinsic parameters:

   $$r_{11} = t_1b_1, r_{12} = t_2b_2, r_{21} = t_2b_4, r_{22} = t_2b_5, t_1 = t_2b_3$$
   $$r_{13} = -\sqrt{1 - r_{11}^2 - r_{12}^2}$$
   $$r_{23} = Sign(r_{11}r_{21} + r_{12}r_{22})\sqrt{1 - r_{21}^2 - r_{22}^2}$$
   $$r_{31} = r_{12}r_{23} - r_{22}r_{13}$$
   $$r_{32} = r_{21}r_{13} - r_{11}r_{23}, r_{33} = r_{11}r_{22} - r_{21}r_{12}$$

5. Obtain an approximate solution for focal length $f$ and translation component $t_3$ from the following equation by ignoring lens distortion:

$$[(r_{21}x + r_{22}y + t_2) - j_d]\begin{bmatrix} f \\ t_3 \end{bmatrix} = [j_d(r_{31}x + r_{32}y)]$$

6. Obtain an accurate estimate of focal length $f$, translation component $t_3$, and first-order radial lens distortion parameter $k_1$ from the following equation by using a standard nonlinear minimization scheme such as steepest descent:

$$\begin{aligned} f(r_{21}x + r_{22}y + t_2) - t_3 j_d - k_1(j_d r_d^2(r_{31}x + r_{32}y)) \\ - k_1 t_3(j_d r_d^2) - j_d(r_{31}x + r_{32}y) = 0 \end{aligned}$$

where $r_d^2 = i_d^2 + j_d^2$. The method is computationally efficient but it produces a suboptimal solution. Moreover, the image center and scale factor parameters are not considered.

**The Conventional Method.** The conventional method is based on the following unconstrained objective function:

$$\begin{aligned} J(\boldsymbol{b}, \boldsymbol{d}) = \sum_{\text{all points}} [(fr_{11}x + fr_{12}y + ft_1 - r_{31}ix - r_{32}iy - it_3)^2 \\ + (fr_{21}x + fr_{22}y + ft_2 - r_{31}jx - r_{32}jy - jt_3)^2] \end{aligned} \tag{32}$$

Here, $i = i(\boldsymbol{d})$ and $j = j(\boldsymbol{d})$ by using Eq. (6). The method consists of the following iterative algorithm consisting of two linear least-squares step:

1. Compute $\boldsymbol{b}$ with $\boldsymbol{d}$ held constant. From Eq. (32), we obtain the following linear equation in unknown parameter vector $\boldsymbol{b}$:

$$\begin{bmatrix} x & y & 0 & 0 & -ix & -iy & 1 & 0 \\ 0 & 0 & x & y & -jx & -jy & 0 & 1 \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} i \\ j \end{bmatrix}$$

where

$$\boldsymbol{b}^{\mathrm{T}} = [fr_{11} \quad fr_{12} \quad fr_{21} \quad fr_{22} \quad r_{31} \quad r_{32} \quad ft_1 \quad ft_2]/t_3$$

We use the linear least-squares method to solve for $\boldsymbol{b}$.

2. Compute $\boldsymbol{d}$ with $\boldsymbol{b}$ held constant. From Eq. (32), we obtain the following linear equation in unknown parameter vector $\boldsymbol{d}$:

$$c_3 \begin{bmatrix} ir^2 & ir^4 & \dots & ir^{2r_0} & r^2 + 2i^2 & 2ij & r^2 & 0 \\ jr^2 & jr^4 & \dots & jr^{2r_0} & 2ij & r^2 + 2j^2 & 0 & r^2 \end{bmatrix} \boldsymbol{d} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

where

$$\begin{aligned} \boldsymbol{d}^{\mathrm{T}} &= [k_1 k_2 \dots k_{r_0} p_1 p_2 s_1 s_2] \\ r^2 &= i^2 + j^2 \\ c_1 &= (b_1 x + b_2 y + b_7) - i(b_5 x + b_6 y + 1) \\ c_2 &= f(b_3 x + b_4 y + b_8) - j(b_5 x + b_6 y + 1) \\ c_3 &= (b_5 x + b_6 y + 1) \end{aligned}$$

$r_0$ is the order of the radial lens distortion model, and $b^T = [b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ b_7\ b_8]$.

Then, by imposing the orthonormality constraints [Eq. (31)], we obtain the parameters below:

$$f^2 = -\frac{b_1 b_2 + b_3 b_4}{b_5 b_6}, t_3^2 = \frac{f^2}{b_1^2 + b_3^2 + f^2 b_5^2} \text{ or } t_3^2 = \frac{f^2}{b_2^2 + b_4^2 + f^2 b_6^2}$$

$$r_{11} = \frac{t_3 b_1}{f}, r_{12} = \frac{t_3 b_2}{f}, r_{21} = \frac{t_3 b_3}{f} = r_{22} = \frac{t_3 b_4}{f}$$

$$r_{31} = t_3 b_5, r_{32} = t_3 b_6, t_1 = \frac{t_3 b_7}{f}, \text{ and } t_2 = \frac{t_3 b_8}{f}$$

The sign of $t_3$ can be determined from the camera position with respect to the world coordinate system as described in the section entitled "Yakimovsky and Cunningham's Method."

The method is reasonably efficient, but it produces a suboptimal solution including an ambiguous solution for $t_3$. Furthermore, the image center and scale factor parameters are not considered.

### Nonlinear Method for Coplanar Camera Calibration

The nonlinear method is based on the objective function, Eq. (32), except that the image coordinates $i$ and $j$ are corrected for image center, scale factor, and lens distortion; that is, they are functions of both $\boldsymbol{m}$ and $\boldsymbol{d}$: $i = i(\boldsymbol{m}, \boldsymbol{d})$ and $j = j(\boldsymbol{m}, \boldsymbol{d})$. Furthermore, the orthonormality constraints, Eq. (31), are imposed during the optimization process. We define the parameter block containing extrinsic parameters and focal length as $\boldsymbol{b}^{\mathrm{T}} = [fr_{11}\ fr_{12}\ fr_{21}\ fr_{22}\ r_{31}\ r_{32}\ ft_1\ ft_2]/t_3$. We can now write the three constraints in Eq. (31) in terms of the elements of $\boldsymbol{b}$. Let $\boldsymbol{b}^{\mathrm{T}} = [b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ b_7\ b_8]$. The constraint for coplanar camera calibration is

$$\begin{aligned} h(\boldsymbol{b}) = (b_5 b_1 + b_6 b_2)(b_6 b_1 - b_5 b_2) \\ + (b_5 b_3 + b_6 b_4)(b_1 b_3 - b_5 b_4) = 0 \end{aligned} \tag{33}$$

The method consists of the following three iterative steps:

1. Compute $\boldsymbol{b}$ with $\boldsymbol{d}$ and $\boldsymbol{m}$ held constant. From Eq. (32), we derive the problem: Solve for $\boldsymbol{b}$ from the following constrained equations:

$$\begin{bmatrix} x & y & 0 & 0 & -ix & -iy & 1 & 0 \\ 0 & 0 & x & y & -jx & -jy & 0 & 1 \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} i \\ j \end{bmatrix}$$

under constraint $h(\boldsymbol{b}) = 0$ [see Eq. (33)]. We use nonlinear optimization methods to solve this problem.

2. Compute $\boldsymbol{d}$ with $\boldsymbol{b}$ and $\boldsymbol{m}$ held constant. This step is similar to the corresponding step in the conventional linear method described in the section entitled "The Conventional Method." From Eq. (32), we obtain the following linear equation in unknown parameter vector $\boldsymbol{d}$:

$$c_3 \begin{bmatrix} ir^2 & ir^4 & \dots & ir^{2r_0} & r^2 + 2i^2 & 2ij & r^2 & 0 \\ jr^2 & jr^4 & \dots & jr^{2r_0} & 2ij & r^2 + 2j^2 & 0 & r^2 \end{bmatrix} \boldsymbol{d} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

where

$$\begin{aligned} \boldsymbol{d}^{\mathrm{T}} &= [k_1 k_2 \ldots k_{r_0} p_1 p_2 s_1 s_2] \\ r^2 &= s^{-2}(i_f - i_0)^2 + (j_f - j_0)^2 \\ i &= s^{-1}(i_f - i_0), j = (j_f - j_0) \\ c_1 &= (b_1 x + b_2 y + b_7) - i(b_5 x + b_6 y + 1) \\ c_2 &= f(b_3 x + b_4 y + b_8) - j(b_5 x + b_6 y + 1) \\ c_3 &= (b_5 x + b_6 y + 1), \text{ and} \end{aligned}$$

$r_0$ is the order of the radial lens distortion model.

3. Compute **m** with **b** and **d** held constant. This is an unconstrained minimization problem similar to the one described in the section entitled "Main Algorithm." The objective function $J(.)$ in Eq. (32) is minimized.

All extrinsic parameters and focal length are obtained from the equations at the end of the section entitled "The Conventional Method." The ambiguity in the solution of $t_3$ is resolved due to the constraint $h(\mathbf{b}) = 0$.

The method is computationally complex, but it produces an optimal solution. The method also computes the image center and scale factor parameters.

## EXPERIMENTAL RESULTS

Though we have presented at least 10 different methods of camera calibration, we select four representative methods for our experiments. Two of these methods are for noncoplanar calibration, and two for coplanar calibration. Of these two methods in each calibration case, one is a linear computation method and one a nonlinear method. For noncoplanar calibration, we choose the Grosky and Tamburino's method (4) as the linear method, and the Chatterjee, Roychowdhury, and Chong's method (22) as the nonlinear method. For coplanar calibration, we choose the Conventional method for linear computation, and then the nonlinear method.

We test these algorithms on two sets of data: (1) synthetic data corrupted with known noise, and (2) real data obtained from a calibration setup. The accuracy of each algorithm is evaluated according to square root of the mean square errors in both image components:

Image Error

$$= \sqrt{\frac{1}{N}\left(\sum_{n=1}^{N} s^{-2}(i_n - i_n(\mathbf{b},\mathbf{d},\mathbf{m}))^2 + \sum_{n=1}^{N}(j_n - j_n(\mathbf{b},\mathbf{d},\mathbf{m}))^2\right)} \tag{34}$$

where $(i_n, j_n)$ are points measured from the image and $(i_n(\mathbf{b}, \mathbf{d}, \mathbf{m}), j_n(\mathbf{b}, \mathbf{d}, \mathbf{m}))$ are computed from calibration parameters $(\mathbf{b}, \mathbf{d}, \mathbf{m})$.

### Experiments with Synthetic Data

The synthetic data was generated for both noncoplanar and coplanar cases with a known set of extrinsic and intrinsic camera parameters. First a 10 × 10 (i.e., $N = 100$) grid of points are generated to simulate the calibration target. $(X_w, Y_w)$ components of the calibration points are uniformly spaced in a rectangular grid within the range $(-5, +5)$. For the noncoplanar case, the $Z_w$ component is randomly generated (uniform distribution) within the range $(-5, +5)$. The rotation matrix $R$ in Eq. (3) is generated from Euler angles $\omega = \phi = \kappa = 15°$. Translation vector $\mathbf{t}$ is (0.5, 0.5, −14.0), and focal length $f = 300$. Image coordinates are obtained from the collinearity conditions, stated in Eq. (1). Lens distortion is then added to this data according to first- and second-order radial distortion models with coefficients $k_1 = 10^{-7}$ and $k_2 = 10^{-14}$. Image center and scale factor parameters are next added to the image with $i_0 = 5, j_0 = 8$ and $s = 0.8$. Finally, an independent Gaussian quantization noise is added to the image components. In accordance with (40), the variance of the added noise in $i$ and $j$ image coordinates are $(sf)^{-2}/12$ and $f^{-2}/12$, respectively. The reasoning behind these noise variances are briefly given below. Different values of $\eta$ in Tables 2, 3, and 4 denote different multiples of this noise.

In accordance with (11), calibration accuracy can be defined by projecting each pixel onto a plane that is orthogonal to the optical axis, and go through the projected pixel center on the calibration surface. The projection of the pixel on this plane is a rectangle of size $a \times b$ with $a$ in the $I$ image direction and $b$ in the $J$ image direction (see Fig. 1). Uniform digitization noise in a rectangle $a \times b$ has standard deviation $\mu_0 = \sqrt{(a^2 + b^2)/12} = z^2 \sqrt{((sf)^{-2} + f^{-2})/12}$ at depth $z$. As per (11), this positional inaccuracy is comparable to the normalized error in image plane denoted by $\mu$, and defined as below:

Normalized Image Error

$$= \mu = \sqrt{\frac{1}{N}\left(\sum_{n=1}^{N}\left(\frac{i_n - i_n(\mathbf{b},\mathbf{d},\mathbf{m})}{sf}\right)^2 + \sum_{n=1}^{N}\left(\frac{j_n - j_n(\mathbf{b},\mathbf{d},\mathbf{m})}{f}\right)^2\right)} \tag{35}$$

We shall use the normalized image error $\mu$ to evaluate the accuracy of all algorithms.

**Noncoplanar Case.** First the Grosky and Tamburino's Linear method is applied to the synthetic image data and corresponding world coordinates. Next, the Chatterjee, Roychowdhury, and Chong's nonlinear method is applied to this data. The initialization algorithm, followed by five iterations of the main algorithm is used. These results for $\eta$ = 1, 5 and 10 are shown in Table 2.

The results clearly show a significant improvement in all parameters as well as image error due to the nonlinear algorithm for all noise levels. As expected, a higher image error is seen for larger quantization noise denoted by $\eta$. Table 3 shows the image error against iterations of the nonlinear algorithm. Different amounts of quantization noise show that the image error, although better compared with the linear method, increases with increasing noise $\eta$.

**Coplanar Case.** Similar to the noncoplanar case, we estimate the calibration parameters first with the linear (Conventional) method and then with the nonlinear method. In order to use the (Conventional) linear method, we need to compute the image center $(i_0, j_0)$ and scale factor $(s)$ parameters by an off-line method. We used an algorithmic scheme given in (22) to obtain $s = 0.801860$, $i_0 = 4.768148$, and $j_0 = 8.37548$, when the true values are $s = 0.8$, $i_0 = 5$, and $j_0 = 8$. We used the 20 iterations of each algorithm. The results are given in Table 4.

**Table 2. Errors in Parameters and Image Due to the Linear and Nonlinear Methods[a] (Noncoplanar)**

| Parameters | Linear ($\eta = 1$) | Nonlinear ($\eta = 1$) | Linear ($\eta = 5$) | Nonlinear ($\eta = 5$) | Linear ($\eta = 10$) | Nonlinear ($\eta = 10$) |
|---|---|---|---|---|---|---|
| $\|\mathbf{r}_1 - \mathbf{r}_1^*\|/\|\mathbf{r}_1^*\|$ | 0.00130163 | 0.00001288 | 0.00137304 | 0.00002542 | 0.00153758 | 0.00007441 |
| $\|\mathbf{r}_2 - \mathbf{r}_2^*\|/\|\mathbf{r}_2^*\|$ | 0.00019631 | 0.00000522 | 0.00017492 | 0.00003419 | 0.00022877 | 0.00007260 |
| $\|\mathbf{r}_3 - \mathbf{r}_3^*\|/\|\mathbf{r}_3^*\|$ | 0.00131631 | 0.00001350 | 0.00148493 | 0.00004105 | 0.00174635 | 0.00010160 |
| $\|\mathbf{t} - \mathbf{t}^*\|/\|\mathbf{t}^*\|$ | 0.00117155 | 0.00002384 | 0.00125086 | 0.00014397 | 0.00153610 | 0.00031061 |
| $\|f - f^*\|/\|f^*\|$ | 0.00081911 | 0.00002200 | 0.00082520 | 0.00014350 | 0.00100966 | 0.00030045 |
| $\|i_0 - i_0^*\|/\|i_0^*\|$ | 0.05657449 | 0.00056567 | 0.05788677 | 0.00202255 | 0.06279373 | 0.00523871 |
| $\|j_0 - j_0^*\|/\|j_0^*\|$ | 0.00353542 | 0.00016246 | 0.00377974 | 0.00099542 | 0.00552706 | 0.00201209 |
| $\|s - s^*\|/\|s^*\|$ | 0.00005129 | 0.00000139 | 0.00006083 | 0.00001226 | 0.00007275 | 0.00002601 |
| $\|\mathbf{d} - \mathbf{d}^*\|/\|\mathbf{d}^*\|$ |  | 0.00697193 |  | 0.02037457 |  | 0.02843441 |
| Image Error | 0.03385128 | 0.00178726 | 0.03490645 | 0.00881193 | 0.03808799 | 0.01461288 |
| $\mu$ | 0.00011293 | 0.00000596 | 0.00011644 | 0.00002936 | 0.00012704 | 0.00004869 |

[a] $\eta$ = Multiple of quantization noise units added to the image points.

**Table 3. Image Errors for Iterations of the Nonlinear Method[a]**

|  | Linear Method | Init. Alg. | Iter. = 1 | Iter. = 2 | Iter. = 3 | Iter. = 4 | Iter. = 5 |
|---|---|---|---|---|---|---|---|
| $\eta = 1$[b] | 0.03463596 | 0.00206831 | 0.00179046 | 0.00178563 | 0.00178435 | 0.00178354 | 0.00178290 |
| $\eta = 2$ | 0.03491059 | 0.00369134 | 0.00351634 | 0.00351403 | 0.00351345 | 0.00351306 | 0.00351273 |
| $\eta = 3$ | 0.03527227 | 0.00540639 | 0.00525823 | 0.00525680 | 0.00525646 | 0.00525621 | 0.00525600 |
| $\eta = 4$ | 0.03571836 | 0.00714753 | 0.00700423 | 0.00700324 | 0.00700302 | 0.00700285 | 0.00700270 |
| $\eta = 5$ | 0.03624572 | 0.00889948 | 0.00875188 | 0.00875116 | 0.00875102 | 0.00875090 | 0.00875079 |

[a] $\eta$ = Multiple of quantization noise units added to the image points.
[b] Iterations 1–5 are for the Main Alg.

**Table 4. Errors in Parameters and Image Due to the Linear and Nonlinear Methods (Coplanar)**

| Parameters | Linear ($\eta = 1$) | Nonlinear ($\eta = 1$) | Linear ($\eta = 5$) | Nonlinear ($\eta = 5$) | Linear ($\eta = 10$) | Nonlinear ($\eta = 10$) |
|---|---|---|---|---|---|---|
| $\|\mathbf{r}_1 - \mathbf{r}_1^*\|/\|\mathbf{r}_1^*\|$ | 0.23353182 | 0.00035990 | 0.23354564 | 0.00036719 | 0.23367292 | 0.00038140 |
| $\|\mathbf{r}_2 - \mathbf{r}_2^*\|/\|\mathbf{r}_2^*\|$ | 0.00545535 | 0.00055831 | 0.00546050 | 0.00058498 | 0.00546703 | 0.00061839 |
| $\|\mathbf{r}_3 - \mathbf{r}_3^*\|/\|\mathbf{r}_3^*\|$ | 0.01506598 | 0.00456795 | 0.01516991 | 0.00468768 | 0.01530110 | 0.00485197 |
| $\|\mathbf{t} - \mathbf{t}^*\|/\|\mathbf{t}^*\|$ | 0.03428311 | 0.00475897 | 0.03432293 | 0.00491331 | 0.03437303 | 0.00510839 |
| $\|f - f^*\|/\|f^*\|$ | 0.02422284 | 0.00458828 | 0.02432405 | 0.00471164 | 0.02445051 | 0.00486651 |
| $\|i_0 - i_0^*\|/\|i_0^*\|$ |  | 0.04601510 |  | 0.04858865 |  | 0.05155711 |
| $\|j_0 - j_0^*\|/\|j_0^*\|$ |  | 0.04551644 |  | 0.04695227 |  | 0.04875117 |
| $\|s - s^*\|/\|s^*\|$ |  | 0.00000293 |  | 0.00000865 |  | 0.00001538 |
| $\|\mathbf{d} - \mathbf{d}^*\|/\|\mathbf{d}^*\|$ | 0.18899921 | 0.01137128 | 0.18898312 | 0.05210049 | 0.18896301 | 0.10238679 |
| Image Error | 0.03688418 | 0.00165670 | 0.03898028 | 0.00817535 | 0.04290334 | 0.01533643 |
| $\mu$ | 0.00012600 | 0.00000555 | 0.00013317 | 0.00002738 | 0.00014660 | 0.00005112 |

**Table 5. Calibration Results for Real Data (Noncoplanar)**

| Parameters | Linear Method | Nonlinear Method |
|---|---|---|
| $\mathbf{r}_1$ | 0.9999482 0.0000020 −0.0101782 | 0.9999501 0.0000290 −0.0099916 |
| $\mathbf{r}_2$ | 0.0001275 0.9999191 0.0127197 | 0.0000262 0.9999847 0.0055243 |
| $\mathbf{r}_3$ | 0.0101774 −0.0127203 0.9998673 | 0.0099916 −0.0055243 0.9999348 |
| $\mathbf{t}$ | 5.0072589 −6.3714880 −509.2793659 | 4.8876372 −2.6923089 −507.3856448 |
| $f$ | 4731.5257593 | 4724.5813533 |
| $i_0, j_0$ | 21.9558272 −12.4401193 | 19.9538046 −17.3227311 |
| $s$ |  | 1.0005753 |
| $k_1, k_2$ |  | $1.0650 \times 10^{-07}$ $-4.6403 \times 10^{-13}$ |
| $p_1, p_2$ |  | $3.0800 \times 10^{-06}$ $-2.1293 \times 10^{-06}$ |
| $s_1, s_2$ |  | $1.1242 \times 10^{-06}$ $-5.6196 \times 10^{-07}$ |
| Image Error | 0.24930409 | 0.14625769 |
| $\mu$ | 0.00005269 | 0.00003096 |

**Table 6. Calibration Results for Real Data (Coplanar)**

| Parameters | Linear Method | Nonlinear Method |
|---|---|---|
| $\mathbf{r}_1$ | 0.9992172 $-0.0000144$ | 0.9996423 0.0000762 |
| $\mathbf{r}_2$ | 0.0000175 0.9999982 | 0.0000371 0.9999910 |
| $\mathbf{r}_3$ | 0.0016014 $-0.0019229$ | 0.0267438 $-0.0042384$ |
| $\mathbf{t}$ | $-0.5774381$ $-0.8354928$ $-76.8508100$ | 0.1626380 $-1.7551208$ $-371.4850635$ |
| $f$ | 713.9534937 | 3464.3130475 |
| $i_0, j_0$ | | 6.9151391 $-8.6055953$ |
| $s$ | | 0.9998891 |
| $k_1, k_2$ | $1.3643 \times 10^{-07}$ $-4.3674 \times 10^{-12}$ | $2.1772 \times 10^{-07}$ $-2.4362 \times 10^{-12}$ |
| $p_1, p_2$ | | $-3.0989 \times 10^{-06}$ $-1.1376 \times 10^{-06}$ |
| $s_1, s_2$ | | $5.2088 \times 10^{-06}$ $-2.0617 \times 10^{-06}$ |
| Image Error | 0.25422583 | 0.21395695 |
| $\mu$ | 0.00035608 | 0.00006176 |

These results clearly show that all parameters are estimated with greater accuracy by the nonlinear method when compared with the linear method. However, the overall estimation accuracy of parameters is lower in comparison with the noncoplanar case. This is expected, since a smaller number of parameters are estimated and fewer constraints can be imposed. However, the total image error is less with the nonlinear method. The results also show an increase in the image error due to higher quantization noise. As seen in Table 3, a similar pattern of image error against iterations is obtained for the coplanar case.

### Experiments with Real Data

Real data is generated from test calibration points created by accurately placing a set of 25 dots in a square grid of 5 × 5 dots on a flat surface. The center to center distance between the dots is 7.875 mm. The diameter of each dot is 3.875 mm. The calibration pattern is mounted on a custom-made calibration stand. The centroid pixel of each dot is obtained by image processing to subpixel accuracy. Although we observed high lens distortion for wide angle lenses, we used a 35 to 70 mm zoom lens because of its frequent use in many applications, and a depth of field that can focus within a range of 0 to 60 mm. We used an off-the-shelf camera in a monoview setup. The camera resolution is 512 × 480 pixels and the digitizer gives digital images with 16 bits/pixel.

For noncoplanar calibration, we placed the calibration target on a high precision stand that can move along the $Z_w$ axis of world coordinates with high accuracy by means of a micrometer screw. Each 360° turn of the screw moves the target by 1 mm, and the positional accuracy is within 0.003 mm. The calibration target is positioned at three positions, $z = 0$, $z = -20$ mm, and $z = -40$ mm. Image points are extracted for all 25 dots in each location to obtain a total of 75 data points for calibration. A second image is acquired at all three of the above $z$ locations for testing. Second-order radial and first-order decentering and thin prism lens distortion models are used. The linear (Grosky and Tamburino's) and nonlinear (Chatterjee, Roychowdhury, and Chong's) methods are used with five iterations of the nonlinear algorithm. The results for the noncoplanar case are shown in Table 5. As seen with the synthetic data, image error has improved due the nonlinear algorithm.

In the coplanar case, we placed the calibration grid at a fixed $z$ location and acquired the calibration and test images. We next estimated the image center ($i_0$, $j_0$) and scale factor ($s$) parameters by an off-line method (22). We obtained $s = 1.001084$, $i_0 = 6.912747, j_0 = -8.601255$. Note that the camera settings for the coplanar case are different from the noncoplanar case, and the results are expectedly different. Twenty iterations of both linear (Conventional) and nonlinear algorithms are used. The results for the coplanar case are summarized in Table 6.

As seen before, the image error improved with the nonlinear method. Further note that the parameters computed by the nonlinear method are far more consistent with the actual setup. For example, the linear method estimates focal length $f = 713.95$, whereas the nonlinear method estimates $f = 3464.31$ (see Table 6). In a similar setup for the noncoplanar case, we obtained $f = 4724.58$. Thus, the nonlinear estimate of focal length is more consistent with the actual setup.

In order to check the effects of the tangential lens distortion model, we used just the radial model on the above data and obtained a normalized image error of 0.00010177. This is compared with 0.00006176 obtained above with radial, tangential, and thin prism lens distortion models. This experiment shows that the decentering and thin prism distortion models are effective in reducing the net normalized image error. Similar results are obtained by others (1,11,20).

## CONCLUDING REMARKS

In this article, we discuss several methods of noncoplanar and coplanar camera calibration. The methods range in computational complexity, execution speed, robustness, and accuracy of the estimates. The linear methods discussed in the section entitled "Linear Methods of Camera Calibration" are computationally efficient and relatively uncomplicated, but they lack the accuracy and robustness of the parameter estimates. The nonlinear methods discussed in the section entitled "Nonlinear Methods of Camera Calibration" are computationally complex, but the parameter estimates are very accurate provided that the calibration data are also accurate. The robust methods discussed in the section entitled "Robust Methods of Noncoplanar Camera Calibration" can produce good parameter estimates even if the control points are not so accurate. However, the method cannot compute all calibration parameters and is computationally complex. In the end, we have given three methods of coplanar camera calibration: Two are computationally simple but produce suboptimal solutions, and the other is computationally complex and produces an optimal solution.

## BIBLIOGRAPHY

1. O. D. Faugeras and G. Toscani, The calibration problem for stereo, *Proc. Comput. Vision Pattern Recognition Conf.,* Miami Beach, FL, 1986, pp. 15–20.
2. S. Ganapathy, Decomposition of transformation matrices for robot vision, *Proc. Int. Conf. Robotics Autom.,* pp. 130–139, 1984.
3. K. D. Gremban et al., Geometric camera calibration using systems of linear equations, *Int. Conf. Robotics,* 1988, pp. 562–567.
4. W. I. Grosky and L. A. Tamburino, A unified approach to the linear camera calibration problem, *IEEE Trans. Pattern Anal. Mach. Intell.,* **12**: 663–671, 1990.
5. M. Ito and A. Ishii, Range and shape measurement using three-view stereo analysis, *Proc. Comput. Vision Pattern Recognition Conf.,* Miami Beach, FL: 1986, pp. 9–14.
6. I. Sobel, On calibrating computer controlled cameras for perceiving 3-D scenes, *Artif. Intell.,* **5**: 185–198, 1974.
7. Y. Yakimovsky and R. Cunningham, A system for extracting three-dimensional measurements from a stereo pair of TV cameras, *Comput. Graph. Image Process.,* **7**: 195–210, 1978.
8. D. B. Gennery, Stereo-camera calibration, *Proc. Image Understanding Workshop,* 1979, pp. 101–108.
9. R. Y. Tsai, A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE J. Robotics Automat.,* **RA-3**: 323–343, 1987.
10. R. K. Lenz and R. Y. Tsai, Techniques for calibrating the scale factor and image center for high accuracy 3-D machine vision metrology, *IEEE Trans. Pattern Anal. Mach. Intell.,* **10**: 713–720, 1988.
11. J. Weng, P. Cohen, and M. Herniou, Camera calibration with distortion models and accuracy evaluation, *IEEE Trans. Pattern Anal. Mach. Intell.,* **14**: 965–980, 1992.
12. W. Faig, Calibration of close-range photogrammetric systems: Mathematical formulation, *Photogramm. Eng. Remote Sens.,* **41** (12): 1479–1486, 1975.
13. S. K. Ghosh, *Analytical Photogrammetry,* New York: Pergamon, 1979.
14. R. C. Malhotra and H. M. Karara, A computational procedure and software for establishing a stable three-dimensional test area for close-range applications, *Proc. Symp. Close-Range Programmetric Syst.,* Champaign, IL, 1975.
15. *Manual of Photogrammetry,* 4th ed., American Society of Photogrammetry, 1980.
16. K. W. Wong, Mathematical formulation and digital analysis in close-range photogrammetry, *Photogramm. Eng. Remote Sens.,* **41** (11): 1355–1375, 1975.
17. A. Bani-Hasemi, Finding the aspect-ratio of an imaging system, *Proc. Comput. Vision Pattern Recognition Conf.,* Maui, Hawaii, 1991, pp. 122–126.
18. Y. I. Abdel-Aziz and H. M. Karara, Direct linear transformation into object space coordinates in close-range photogrammetry, *Proc. Symp. Close-Range Photogrammetry,* Univ. of Illinois at Urbana-Champaign, 1971, pp. 1–18.
19. H. M. Karara (ed.), *Non-Topographic Photogrammetry,* 2nd ed., American Society for Photogrammetry and Remote Sensing, 1989.
20. H. A. Beyer, Accurate calibration of CCD-cameras, *Proc. Comput. Vision Pattern Recognition Conf.,* Champaign, IL, 1992, pp. 96–101.
21. Y. Nomura et al., Simple calibration algorithm for high-distortion-lens camera, *IEEE Trans. Pattern Anal. Mach. Intell.,* **14**: 1095–1099, 1992.
22. C. Chatterjee, V. P. Roychowdhury, and E. K. P. Chong, A nonlinear Gauss–Seidel algorithm for noncoplanar and coplanar camera calibration with convergence analysis, *Comput. Vision Image Understanding,* **67** (1): 58–80, 1997.
23. D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation,* Englewood Cliffs, NJ: Prentice-Hall, 1989.
24. G. H. Golub and C. F. VanLoan, *Matrix Computations,* Baltimore: Johns Hopkins Univ. Press, 1983.
25. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision,* Vol. 2, Reading, MA: Addison-Wesley, 1993.
26. R. G. Wilson and S. A. Shafer, What is the center of the image?, Technical Report CMU-CS-93-122, Carnegie Mellon University, Pittsburgh, 1993.
27. W. J. Smith, *Modern Optical Engineering, The Design of Optical Systems, Optical and Electro-Optical Engineering Series,* New York: McGraw-Hill, 1966.
28. L. L. Wang and W. H. Tsai, Camera calibration by vanishing lines for 3D computer vision, *IEEE Trans. Pattern Anal. Mach. Intell.,* **13**: 370–376, 1991.
29. J. Z. C. Lai, On the Sensitivity of Camera Calibration, *Image and Vision Computing,* **11** (10): 656–664, 1993.
30. M. A. Penna, Camera calibration: A quick and easy way to determine the scale factor, *IEEE Trans. Pattern Anal. Mach. Intell.,* **13**: 1240–1245, 1991.
31. B. Caprile and V. Torre, Using vanishing points for camera calibration, *Int. J. Comput. Vision,* **4**: 127–140, 1990.
32. D. C. Brown, Decentering distortion of lenses, *Photogramm. Eng.,* **32**: 444–462, 1966.
33. D. C. Brown, Close-range camera calibration, *Photogramm. Eng.,* **37**: 855–866, 1971.
34. M. Ito and A. Ishii, A non-iterative procedure for rapid and precise camera calibration, *Pattern Recognition,* **27** (2): 301–310, 1994.
35. J. Y. S. Luh and J. A. Klaasen, A Three-Dimensional Vision by Off-Shelf System with Multi-Cameras, *IEEE Trans. Pattern Anal. Machine Intell.* **7**: 35–45, 1985.
36. D. A. Butler and P. K. Pierson, A Distortion-Correction Scheme by Industrial Machine-Vision Application, *IEEE Trans. Robotics Autom.,* **7**: 546–551, 1991.
37. S-W. Shih et al., Accurate linear technique for camera calibration considering lens distortion by solving an eigenvalue problem, *Opt. Eng.,* **32** (1): 138–149, 1993.
38. D. Luenberger, *Linear and Nonlinear Programming,* 2nd ed., Reading, MA: Addison-Wesley, 1984.
39. R. M. Haralick et al., Pose estimation from corresponding point data, *IEEE Trans. Syst. Man Cybern.,* **19**: 1426–1446, 1989.
40. K. S. Arun, T. S. Huang, and S. D. Bolstein, Least-squares fitting of two 3-D point sets, *IEEE Trans. Pattern Anal. Mach. Intell.,* **9**: 698–700, 1987.
41. R. L. Launer and G. N. Wilkinson, *Robustness in Statistics,* New York: Academic Press, 1979.

Chanchal Chatterjee
GDE Systems Inc.

Vwani P. Roychowdhury
UCLA

**CAPACITANCE.** See Dielectric polarization.