# AN INTEGRATED LINEAR TECHNIQUE FOR POSE ESTIMATION FROM DIFFERENT GEOMETRIC FEATURES

QIANG JI
*Department of Computer Science, University of Nevada*
*Reno, NV 89557, USA*

MAURO S. COSTA, ROBERT M. HARALICK
and LINDA G. SHAPIRO
*Department of Electrical Engineering, University of Washington, USA*
*E-mail: {qiangji, mauro, haralick, shapiro}@george.ee.washington.edu*

Existing linear solutions for the pose estimation (or exterior orientation) problem suffer from a lack of robustness and accuracy partially due to the fact that the majority of the methods utilize only one type of geometric entity and their frameworks do not allow simultaneous use of different types of features. Furthermore, the orthonormality constraints are weakly enforced or not enforced at all. We have developed a new analytic linear least-squares framework for determining pose from multiple types of geometric features. The technique utilizes correspondences between points, between lines and between ellipse–circle pairs. The redundancy provided by different geometric features improves the robustness and accuracy of the least-squares solution. A novel way of approximately imposing orthonormality constraints on the sought rotation matrix within the linear framework is presented. Results from experimental evaluation of the new technique using both synthetic data and real images reveal its improved robustness and accuracy over existing direct methods.

*Keywords*: Pose estimation, exterior camera parameter estimation, camera calibration.

## 1. INTRODUCTION

Pose estimation is an essential step in many machine vision and photogrammetric applications including robotics, 3D reconstruction, and mensuration. In computer vision, the problem is also known as exterior camera calibration. It addresses the issue of determining the position and orientation of a camera with respect to an object coordinate frame. Solutions to the pose estimation problem can be classified into linear methods and nonlinear methods. Linear methods have the advantage of computational efficiency, but they suffer from lack of accuracy and robustness. Nonlinear methods, on the other hand, offer a more accurate and more robust solution. They are, however, computationally intensive and require an initial estimate. The classical nonlinear photogrammetric approach to camera calibration requires setting up a constrained nonlinear least-squares system. Given initial estimates of the camera parameters, the system is then linearized and solved iteratively. While the classical technique guarantees all orthonormality constraints and delivers the best answer, it, however, requires a good initial estimate. It is a well-known fact that the initial estimates must be close or the system may not converge quickly



Vision Interface — Real World Applications of Computer Vision (1999) 133–161

or correctly. Hence, the quality of an initial estimate is critical since it determines the convergence speed and the correctness of the iterative procedures. Linear solutions, which are often used to provide initial guesses for nonlinear procedures, are therefore important for computer vision and photogrammetric problems.

Numerous methods have been proposed to analytically obtain exterior camera parameters. Previous methods in camera parameter or pose estimation have primarily been focused on using sets of 2D–3D point correspondences including the three-point solution,[7] the four-point solutions,[15,17] and the six- or more point solutions.[5,25,26] Haralick *et al.*[13] reviewed major analytic solutions from three-point correspondences and characterized their performance under varying noisy conditions. Sutherland[25] provided a closed-form least-squares solution using six or more points. The solution, however, assumed the depth of the camera to be unity. Faugeras[5] proposed a similar constrained least-squares method to solve for the camera parameters that requires at least six-point 2D–3D correspondences. The solution includes an orthogonalization process that ensures the orthonormality of the resulting rotation matrix. Tsai[26] presented a direct solution by decoupling the camera parameters into two groups; each group is solved separately in different stages. While efficient, Tsai's method does not impose any of the orthonormal constraints on the estimated rotation matrix. Also, the errors with the camera parameters estimated in the earlier stage can significantly affect the accuracy of parameters estimated in the later stage.

These methods are effective and simple to implement. However, they are not robust and are very susceptible to noise in image coordinates,[27] especially when the number of control points approaches the minimum required. For the three-point solutions, Haralick *et al.*[13] showed that even the order of algebraic substitutions can render the output useless. Furthermore, the point configuration and noise in the point coordinates can also dramatically change the relative output errors. For least-squares based methods, a different study by Haralick *et al.*[12] showed that when the noise level exceeds a knee level or the number of points is below a knee level, these methods become extremely unstable and the errors skyrocket. The use of more points can help relieve this problem. However, fabrication of more control points often proves to be difficult, expensive, and time-consuming. Another disadvantage of point-based methods is the difficulty with point matching, i.e. finding the correspondences between the 3D scene points and 2D image pixels.

In view of these issues, other researchers have investigated the use of higher-level geometric features such as lines or curves as observed geometric entities to improve the robustness and accuracy of linear methods for estimating camera parameters. Over the years, various algorithms[1,4,21,23] have been introduced. Haralick and Chu[11] presented a method that solves the camera parameters from the conic curves. Given the shape of conic curves, the method first solves the three rotation parameters using an iterative procedure. The three translation parameters are then solved analytically. The advantage of this method is that it does not need to know the location of the curves and it is more robust than any analytical method in that rotation parameter errors are reduced to minimum before they are used

analytically to compute translation parameters. In their analytic method, Liu *et al.*[21] and Chen[1] discussed direct solutions for determining external camera parameters based on a set of 2D–3D line correspondences. The key to this algorithm lies in the linear constraint they used. This constraint uses the fact that a 3D line and its image line lie on the same plane determined by the center of perspectivity and the image line. Rothwell *et al.*[23] discussed a direct method that determines camera parameters using a pair of conic curves. The method works by extracting 4 or 8 points from conic intersections and tangencies. Camera parameters are then recovered from these points. Kumar and Hanson[20] described a robust technique for finding camera parameters using lines. Kamgar–Parsi[19] introduced a camera calibration method with small relative angles. Gao[9] introduced a method for estimating camera parameters using parallelepiped. Forsyth[8] proposed to use a pair of known conics or a single known circle for determining the pose of the object plane. Haralick[10,11] presented methods for solving camera parameters using rectangles and triangles. Ma[24] introduced a technique for pose estimation from the correspondence of 2D/3D conics. The technique, however, is iterative and requires a pair of conics in both 2D and 3D.

Analytic solutions based on high-level geometric features afford better stability and are more robust and accurate. Here the correspondence problem can be addressed more easily than in the point-based methods. However, high-level geometric features may not always be present in some applications, and points are present in many applications. Therefore, completely ignoring points while solely employing high-level geometric entities can be a waste of readily available geometric information. This is one of the problems with the existing solutions: they either use points or lines or conics but not a combination of features. In this paper, we describe an integrated least-squares method that solves the camera transformation matrix analytically by fusing available observed geometric information from different levels of abstraction. Specifically, we analytically solve for the external camera parameters from simultaneous use of 2D-3D correspondences between points, between lines and between 2D ellipses and 3D circles. The attractiveness of our approach is that the redundancy provided by features at different levels improves the robustness and accuracy of the least-squares solution, therefore improving the precision of the estimated parameters. To our knowledge, no previous research attempts have been made in this area. Work by Phong *et al.*[22] described a technique in which information from both points and lines is used to compute the pose. However, the method is iterative and involves only points and lines.

Another major factor that contributes to the lack of robustness of the existing linear methods is that orthonormality constraints on the rotation matrix are often weakly enforced or not enforced at all. In this research, we introduce a simple, yet effective, scheme for approximately imposing the orthonormal constraints on the rotation matrix. While the scheme does not guarantee that the resultant rotation matrix completely satisfies the orthonormal constraints, it does yield a matrix that is closer to orthonormality than those obtained with competing methods.

This paper is organized as follows. Section 2 briefly summarizes the perspective projection geometry and equations. Least-square frameworks for estimating camera transformation matrix from 2D–3D point, line and ellipse/circle correspondences are presented in Secs. 3–5 respectively. Section 6 discusses our technique for approximately imposing orthonormal constraints and presents the integrated linear technique for estimating the transformation matrix simultaneously using point, line and ellipse/circle correspondences. Performance characterization and comparison of the developed integrated technique is covered in Sec. 7.

## 2. PERSPECTIVE PROJECTION GEOMETRY

To set the stage for the subsequent discussion of camera parameter estimation, this section briefly summarizes the pin-hole camera model and the perspective projection geometry.

Let $P$ be a 3D point and $(xyz)^t$ be the coordinates of $P$ relative to the object coordinate frame $C_o$. Define the camera coordinate system $C_c$ to have its $Z$-axis parallel to the optical axis of the camera lens and its origin located at the perspectivity center. Let $(x_c\ y_c\ z_c)^t$ be the coordinates of $P$ in $C_c$. Define $C_i$ to be the image coordinate system, with its $U$-axis and $V$-axis parallel to the $X$ and $Y$ axes of the camera coordinate frame, respectively. The origin of $C_i$ is located at the principal point. Let $(u\ v)^t$ be the coordinates of $P_i$, the image projection of $P$ in $C_i$. Figure 1 depicts the pin-hole camera model.

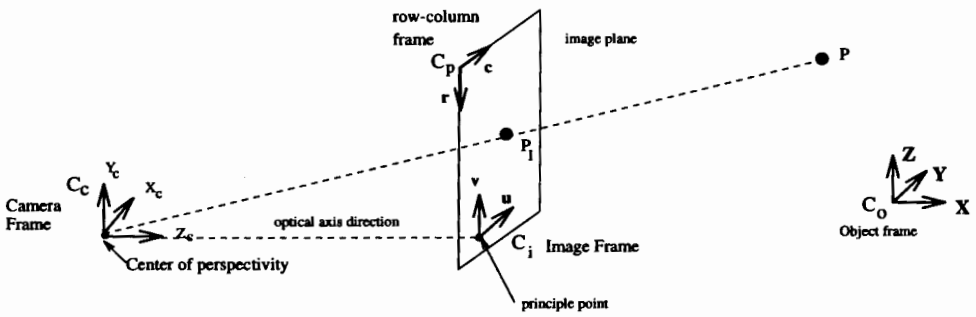


Fig. 1. Camera and perspective projection geometry.

Based on the perspective projection theory, the projection that relates $(u, v)$ on the image plane to the corresponding 3D point $(x_c, y_c, z_c)$ in the camera frame can be described by

$$\lambda \begin{pmatrix} u \\ v \\ f \end{pmatrix} = \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \tag{1}$$

where $\lambda$ is a scalar and $f$ is the camera focal length.

Further, $(x\ y\ z)^t$ relates to $(x_c\ y_c\ z_c)^t$ by a rigid body coordinate transformation consisting of a rotation and a translation. Let a $3 \times 3$ matrix $R$ represent the rotation

and a $3 \times 1$ vector $T$ describe the translation, then

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + T \tag{2}$$

where $T$ and $R$ can be parameterized as $T = (t_x\ t_y\ t_z)$ and

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}.$$

$R$ and $T$ describe the orientation and location of the object frame relative to the camera frame respectively. Together, they are referred to as the camera transformation matrix. Substituting the parameterized $T$ and $R$ into Eq. (2) yields

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}. \tag{3}$$

Assume rotation matrix $R$ results from successive Euler rotations of the camera frame around its $X$-axis by $\omega$, its once rotated $Y$-axis by $\phi$, and its twice rotated $Z$-axis by $\kappa$, then

$$\begin{aligned} R(\omega, \phi, \kappa) &= R_X(\omega) R_Y(\phi) R_Z(\kappa) \\ &= \begin{pmatrix} \cos\phi\cos\kappa & -\cos\phi\sin\kappa & \sin\phi \\ \sin\omega\sin\phi\cos\kappa + \cos\omega\sin\kappa & -\sin\omega\sin\phi\sin\kappa + \cos\omega\cos\kappa & -\sin\omega\cos\phi \\ -\cos\omega\sin\phi\cos\kappa + \sin\omega\sin\kappa & \cos\omega\sin\phi\sin\kappa + \sin\omega\cos\kappa & \cos\omega\cos\phi \end{pmatrix}. \end{aligned} \tag{4}$$

Equation (4) describes relationships between $r_{ij}$ and the Euler angles.

Combining the projection Eq. (1) with the rigid transformation of Eq. (2) and solving for $\lambda$ yields the *collinearity* equations, which describe the ideal relationship between a point on the image plane and the corresponding point in the object frame

$$\begin{aligned} u &= f \frac{r_{11}x + r_{12}y + r_{13}z + t_x}{r_{31}x + r_{32}y + r_{33}z + t_z} \\ v &= f \frac{r_{21}x + r_{22}y + r_{23}z + t_y}{r_{31}x + r_{32}y + r_{33}z + t_z}. \end{aligned} \tag{5}$$

For a rigid body transformation, the rotation matrix $R$ must be orthonormal, that is, $R^t = R^{-1}$. The constraint $R^t = R^{-1}$ amounts to the six orthonormality constraint equations on the elements of $R$

$$\begin{aligned} r_{11}^2 + r_{12}^2 + r_{13}^2 &= 1 \qquad r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23} = 0 \\ r_{21}^2 + r_{22}^2 + r_{23}^2 &= 1 \qquad r_{11}r_{31} + r_{12}r_{32} + r_{13}r_{33} = 0 \\ r_{31}^2 + r_{32}^2 + r_{33}^2 &= 1 \qquad r_{21}r_{31} + r_{22}r_{32} + r_{23}r_{33} = 0 \end{aligned} \tag{6}$$

where the three constraints on the left are referred to as the normality constraints and the three on the right as the orthogonality constraints. The normality constraints ensure that the row vectors of $R$ are unit vectors, while the orthogonality constraints guarantee orthogonality among row vectors.

## 3. CAMERA TRANSFORMATION MATRIX FROM POINT CORRESPONDENCES

Given the 3D object coordinates of a number of points and their corresponding 2D image coordinates, the coefficients of $R$ and $T$ can be solved for by a least-squares solution of an overdetermined system of linear equations. Specifically, the least-squares method based on point correspondences can be formulated as follows.

Let $X_n = (x_n, y_n, z_n)$, $n = 1, \ldots, K$, be the 3D coordinates of $K$ points relative to the object frame and $U_n = (u_n, v_n)$ be the observed image coordinates of these points. We can then relate $X_n$ and $U_n$ via the collinearity equations as follows

$$u_n = f\frac{r_{11}x_n + r_{12}y_n + r_{13}z_n + t_x}{r_{31}x_n + r_{32}y_n + r_{33}z_n + t_z}$$

$$v_n = f\frac{r_{21}x_n + r_{22}y_n + r_{23}z_n + t_y}{r_{31}x_n + r_{32}y_n + r_{33}z_n + t_z}.$$

Rewriting the above equation yields

$$\begin{aligned} &fr_{11}x_n + fr_{12}y_n + fr_{13}z_n - u_n r_{31}x_n - u_n r_{32}y_n - u_n r_{33}z_n + ft_x - u_n t_z = 0 \\ &fr_{21}x_n + fr_{22}y_n + fr_{23}z_n - v_n r_{31}x_n - v_n r_{32}y_n - v_n r_{33}z_n + ft_y - v_n t_z = 0. \end{aligned} \quad (7)$$

We can then set up a matrix $M$ and a vector $V$ as follows

$$M^{2K\times 12} = \begin{pmatrix} fx_1 & fy_1 & fz_1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 & -u_1z_1 & f & 0 & -u_1 \\ 0 & 0 & 0 & fx_1 & fy_1 & fz_1 & -v_1x_1 & -v_1y_1 & -v_1z_1 & 0 & f & -v_1 \\ \vdots & & & & & & & & & & & \\ fx_K & fy_K & fz_K & 0 & 0 & 0 & -u_Kx_K & -u_Ky_K & -u_Kz_K & f & 0 & -u_K \\ 0 & 0 & 0 & fx_K & fy_K & fz_K & -v_Kx_K & -v_Ky_K & -v_Kz_K & 0 & f & -v_K \end{pmatrix} \quad (8)$$

$$V^{12\times 1} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{21} & r_{22} & r_{23} & r_{31} & r_{32} & r_{33} & t_x & t_y & t_z \end{pmatrix}^t \quad (9)$$

where $M$ is hereafter referred to as the collinearity matrix and $V$ is the unknown vector of transformation parameters containing all sought rotational and translational coefficients.

To determine $V$, we can set up a least-squares problem that minimizes

$$\xi^2 = ||MV||^2 \quad (10)$$

where $\xi^2$ is the sum of residual errors of all points. Given an overdetermined system, a solution to the above least-squares minimization requires $MV = 0$. Its solution

contains an arbitrary scale factor. To uniquely determine $V$, different methods have been proposed to solve for the scale factor. In the least-squares solution provided by Sutherland[25], the depth of the camera is assumed to be unity; $t_z = 1$. Not only is this assumption unrealistic for most applications, but also the solution is constructed without regard to the orthonormal constraints that $R$ must satisfy. Faugeras[5] posed the problem as a constrained least-squares problem. The third normality constraint in Eq. (6) is imposed during the minimization to solve for the scale factor and to constrain the rotation matrix.

## 4. CAMERA TRANSFORMATION MATRIX FROM LINE CORRESPONDENCES

Given correspondences between a set of 3D lines and their observed 2D images, we can set up a system of linear equations that involve $R$, $T$, and the coefficients for 3D and 2D lines as follows. Let a 3D line $L$ in the object frame be parametrically represented as

$$L: \; X = \lambda N + P$$

where $X = (x \; y \; z)^t$ is a generic point on the line, $\lambda$ is a scalar, $N = (A, B, C)^t$ is the known direction cosine and $P = (P_x \; P_y \; P_z)^t$ is a known point on the line relative to the object frame. Let the corresponding 2D line $l$ on the image plane be represented by

$$l: \; au + bv + c = 0\,.$$

Ideally the 3D line must lie on the projection plane formed by the center of perspectivity and the 2D image line as shown in Fig. 2.

Relative to the camera frame, the equation of the projection plane can be derived from the 2D line equation as

$$afx_c + bfy_c + cz_c = 0$$

where $f$ is the focal length. Since the 3D line lies on the projection plane, the plane normal must be perpendicular to the line. Denote the plane normal by $n = \frac{(af, bf, c)^t}{\sqrt{a^2f^2 + b^2f^2 + c^2}}$; then given an ideal projection, we have

$$n^t RN = 0\,. \tag{11}$$

Similarly, since point $P$ is also located on the projection plane, this leads to

$$n^t(RP + T) = 0\,. \tag{12}$$

Equations (11) and (12) are hereafter referred to as *coplanarity* equations. Equivalently, they can be rewritten as

$$\begin{aligned} &Aar_{11} + Bar_{12} + Car_{13} + Abr_{21} + Abr_{22} + Cbr_{23} + Acr_{31} + Bcr_{32} + Ccr_{33} = 0 \\ &P_x ar_{11} + P_y ar_{12} + P_z ar_{13} + P_x br_{21} + P_y br_{22} + P_z br_{23} + P_x cr_{31} + P_y cr_{32} \\ &\qquad + P_z cr_{33} + at_x + bt_y + ct_z = 0\,. \end{aligned}$$

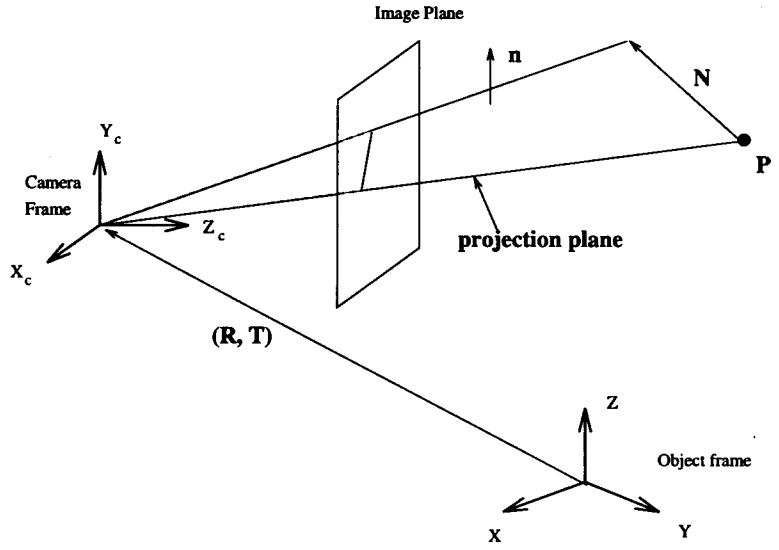


Fig. 2. Projection plane formed by a 2D image line $l$ and the corresponding 3D line $L$.

Given a set of $J$ line correspondences, we can set up a system of linear equations similar to those for points that involve matrix $H$ and vector $V$, where $V$ is as defined before and $H$ is defined as follows

$$H^{2J\times 12} = \begin{pmatrix} A_1a_1 & B_1a_1 & C_1a_1 & A_1b_1 & B_1b_1 & C_1b_1 & A_1c_1 & B_1c_1 & C_1c_1 & 0 & 0 & 0 \\ P_{x_1}a_1 & P_{y_1}a_1 & P_{z_1}a_1 & P_{x_1}b_1 & P_{y_1}b_1 & P_{z_1}b_1 & P_{x_1}b_1 & P_{y_1}c_1 & P_{z_1}c_1 & a_1 & b_1 & c_1 \\ \vdots & & & & & & & & & & & \\ A_Ja_J & B_Ja_J & C_Ja_J & A_Jb_J & B_Jb_J & C_Jb_J & A_Jc_J & B_Jc_J & C_Jc_J & 0 & 0 & 0 \\ P_{x_J}a_J & P_{y_J}a_J & P_{z_J}a_J & P_{x_J}b_J & P_{y_J}b_J & P_{z_J}b_J & P_{x_J}c_J & P_{y_J}c_J & P_{z_J}c_J & a_J & b_J & c_J \end{pmatrix} \tag{13}$$

and is called the coplanarity matrix. Again we can solve for $V$ by minimizing the sum of residual errors $||HV||^2$. $V$ can be solved for up to a scale factor. The scale factor can be determined by imposing one of the orthonormality constraints.

## 5. CAMERA TRANSFORMATION MATRIX FROM ELLIPSE–CIRCLE CORRESPONDENCES

### 5.1. Circle Pose from Ellipse–Circle Correspondence

Given the image of a circle in 3D space and the corresponding ellipse in the image, its pose relative to the camera frame can be solved for analytically. Solutions to this problem may be found in Refs. 3, 8 and 14. Here we propose an algebraic solution as detailed below.

Assume we observe an ellipse in the image, resulting from the perspective projection of a 3D circle. Relative to the image frame $C_i$, the ellipse may be represented

by a standard conic function

$$Au^2 + 2Buv + Cv^2 + 2Du + 2Ev + F = 0 \tag{14}$$

where the coefficients A, B, C, D, E, F can be estimated from a least squares fit. A point in camera frame $(x_c, y_c, z_c)$ relates to its projection $(u, v)$ in the image via

$$u = \frac{fx_c}{z_c} \qquad v = \frac{fy_c}{z_c}.$$

Substituting $u$ and $v$ into Eq. (14) yields

$$Ax_c^2 + 2Bx_cy_c + Cy_c^2 + 2\frac{D}{f}x_cz_c + 2\frac{E}{f}y_cz_c + \frac{F}{f^2}z_c^2 = 0. \tag{15}$$

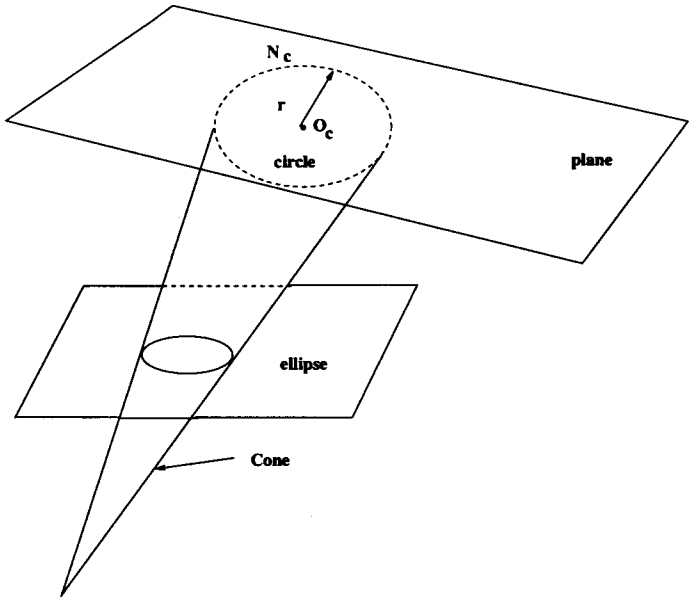


Fig. 3. The 3D cone formed by a 2D ellipse and the corresponding 3D circle.

The above equation defines the equation of a cone formed by the center of perspectivity and the ellipse in the image as shown in Fig. 3. The sought 3D circle must be the cross-section of the cone when cut with a plane (also shown in Fig. 3). Without loss of generality, assume the plane equation, relative to the camera frame, can be represented as $z_c = \alpha x_c + \beta y_c + \gamma$. Substituting the plane equation into the cone Eq. (15) and collecting like terms yield the equation for the cross-section of the cone and the plane

$$\left(A + \frac{E\alpha^2}{f} + \frac{2D\alpha}{f}\right)x_c^2 + \left(C + \frac{E\beta^2}{f} + \frac{2F\beta}{f}\right)y_c^2 + 2\left(B + \frac{E\alpha\beta}{f} + \frac{D\beta}{f} + \frac{F\alpha}{f}\right)x_cy_c$$
$$+ 2\left(\frac{E\alpha\gamma}{f} + \frac{D\gamma}{f}\right)x + 2\left(\frac{E\beta\gamma}{f} + \frac{F\gamma}{f}\right)y + \frac{E\gamma^2}{f} = 0. \tag{16}$$

Given the above equation, the pose of the circle plane is determined by exploiting the property, unique to a circle, that the space curve relative to the camera frame must have equal coefficients for the $x_c^2$ and $y_c^2$ terms and zero coefficient for the $x_c y_c$ term. This property yields the following two simultaneous equations

$$A + \frac{E\alpha^2}{f} + \frac{2D\alpha}{f} - C - \frac{E\beta^2}{f} - \frac{2F\beta}{f} = 0,$$
$$B + \frac{E\alpha\beta}{f} + \frac{D\beta}{f} + \frac{F\alpha}{f} = 0.$$

Reorganizing the above two equations gives rise to

$$\left(\alpha + \frac{D}{E}\right)^2 - \left(\beta + \frac{F}{E}\right)^2 = \frac{(C-A)f}{E}, \tag{17}$$
$$\left(\alpha + \frac{D}{E}\right)\left(\beta + \frac{F}{E}\right) = Bf - DF. \tag{18}$$

Let $s = \alpha + \frac{D}{E}$ and $w = \beta + \frac{F}{E}$, we can easily solve for $s$ and $w$ from the above equations. Given $s$ and $w$, the plane parameters $\alpha$ and $\beta$ can be derived. There will be two solutions for $\alpha$ and $\beta$, yielding a total of four possible solutions for the circle normals. With $\alpha$ and $\beta$ known, we can substitute them into Eq. (15), yielding

$$x_c^2 + y_c^2 + qx_c + ly_c = p \tag{19}$$

where

$$q = \frac{2E\alpha\gamma + 2D\gamma}{Af + E\alpha^2 + 2D\alpha}$$
$$l = \frac{2E\beta\gamma + 2F\gamma}{Af + E\alpha^2 + 2D\alpha}$$
$$p = \frac{-E\gamma^2}{Af + E\alpha^2 + 2D\alpha}.$$

Given the radius of the circle to be $r$, we can use it to solve for the third plane parameter $\gamma$ and subsequently for $q$, $l$ and $p$. Substituting the solutions for $p$, $l$ and $q$ into Eq. (19) allows us to solve for the $x$ and $y$ coordinates of the circle center in the camera frame. The $z$ coordinate of the circle center can then be solved for from the plane equation.

### 5.2. Camera Transformation Matrix from Circles

The pose of a circle, as derived in the last section, is relative to the camera frame. If we are also given the pose of the circle in the object frame, then we can use the two poses to solve for $R$ and $T$. Specifically, let $N_c = (N_{c_x}\ N_{c_y}\ N_{c_z})^t$ and $O_c = (O_{c_x}\ O_{c_y}\ O_{c_z})^t$ be the 3D circle normal and center respectively in the camera coordinate frame. Also, let $N_o = (N_{o_x}\ N_{o_y}\ N_{o_z})^t$ and $O_o = (O_{o_x}\ O_{o_y}\ O_{o_z})^t$ be the normal and center of the same circle, but in the object coordinate system. $N_c$ and

$O_c$ can be obtained from the pose-from-circle technique as described in Sec. 5.1. $N_o$ and $O_o$ are assumed to be known. The problem is to determine $R$ and $T$ from the correspondence between $N_c$ and $N_o$, and between $O_c$ and $O_o$. The two normals and the two centers are related by the transformation $R$ and $T$ as shown below

$$\begin{aligned} N_c &= RN_o \\ &= \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} N_{o_x} \\ N_{o_y} \\ N_{o_z} \end{pmatrix} \end{aligned} \tag{20}$$

and

$$\begin{aligned} O_c &= R\,O_o + T \\ &= \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} O_{o_x} \\ O_{o_y} \\ O_{o_z} \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} . \end{aligned} \tag{21}$$

Equivalently, we can rewrite Eqs. (20) and (21) as follows

$$\begin{aligned} N_{o_x} r_{11} + N_{o_y} r_{12} + N_{o_z} r_{13} &= N_{c_x} \\ N_{o_x} r_{21} + N_{o_y} r_{22} + N_{o_z} r_{23} &= N_{c_y} \\ N_{o_x} r_{31} + N_{o_y} r_{32} + N_{o_z} r_{33} &= N_{c_z} \end{aligned}$$

and

$$\begin{aligned} O_{o_x} r_{11} + O_{o_y} r_{12} + O_{o_z} r_{13} + t_x &= O_{c_x} \\ O_{o_x} r_{21} + O_{o_y} r_{22} + O_{o_z} r_{23} + t_y &= O_{c_y} \\ O_{o_x} r_{31} + O_{o_y} r_{32} + O_{o_z} r_{33} + t_z &= O_{c_z} . \end{aligned}$$

Given $I$ observed ellipses and their corresponding space circles, we can set up a system of linear equations to solve for $R$ and $T$ by minimizing the sum of residual errors $||QV - k||^2$, where $Q$ and $k$ are defined as follows

$$Q = \begin{pmatrix} N_{1_{o_x}} & N_{1_{o_y}} & N_{1_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_{1_{o_x}} & N_{1_{o_y}} & N_{1_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & N_{1_{o_x}} & N_{1_{o_y}} & N_{1_{o_z}} & 0 & 0 & 0 \\ O_{1_{o_x}} & O_{1_{o_y}} & O_{1_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & O_{1_{o_x}} & O_{1_{o_y}} & O_{1_{o_z}} & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & O_{1_{o_x}} & O_{1_{o_y}} & O_{1_{o_z}} & 0 & 0 & 1 \\ \vdots & & & & & & & & & & & \\ N_{I_{o_x}} & N_{I_{o_y}} & N_{I_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_{I_{o_x}} & N_{I_{o_y}} & N_{I_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & N_{I_{o_x}} & N_{I_{o_y}} & N_{I_{o_z}} & 0 & 0 & 0 \\ O_{I_{o_x}} & O_{I_{o_y}} & O_{I_{o_z}} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & O_{I_{o_x}} & O_{I_{o_y}} & O_{I_{o_z}} & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & O_{I_{o_x}} & O_{I_{o_y}} & O_{I_{o_z}} & 0 & 0 & 1 \end{pmatrix} \tag{22}$$

and

$$k = (N_{1_{c_x}}\ N_{1_{c_y}}\ N_{1_{c_z}}\ O_{1_{c_x}}\ O_{1_{c_y}}\ O_{1_{c_z}}\ \ldots\ N_{I_{c_x}}\ N_{I_{c_y}}\ N_{I_{c_z}}\ O_{I_{c_x}}\ O_{I_{c_y}}\ O_{I_{c_z}})^t\,. \quad (23)$$

## 6. THE INTEGRATED TECHNIQUE

In the previous sections we have outlined the least-square frameworks for computing transformation matrix from different features individually. It is desirable to be able to compute camera parameters using more than one type of feature *simultaneously*. In other words, given observed geometric entities at different levels, we want to develop a mechanism that systematically and consistently fuses this information. The reason is quite obvious: using all available geometric information will provide a more accurate and robust solution, since it increases the relative redundancy of the least-squares estimation. It also reduces the dependency on points. We can be more selective when choosing points, without worrying about the minimum number of points needed for accurate results. Furthermore, we do not need to worry about whether the selected points are coplanar or not. To our knowledge, this problem has never been addressed in the literature. This section is devoted to formulating a direct solution for computing camera parameters from 2D–3D correspondences of points, lines and ellipses/circles.

### 6.1. Fusing All Observed Information

The problem of integrating information from points, lines and circles is actually straightforward, given the frameworks we have outlined individually for points, lines and circles. The problem can be stated as follows.

Given the 2D–3D correspondences of $K$ points, $J$ lines and $I$ ellipse/circle pairs, we want to set up a system of linear equations that involves all geometric entities. The problem can be formulated as a least-squares estimation in the form of minimizing $||WV - b||$, where $V$ is the unknown vector of transformation parameters as defined before, and $b$ is a known vector defined below. $W$ is an augmented coefficient matrix, whose rows consist of linear equations derived from points, lines and circles. Specifically, given the $M$, $H$ and $Q$ matrices defined in Eqs. (8), (13), and (22), the $W$ matrix is

$$W = \begin{pmatrix} M \\ H \\ Q \end{pmatrix} \quad (24)$$

where the first $2K$ rows of $W$ represent contributions from points, the second subsequent $2J$ rows represent contributions from lines, and the last $6I$ rows represent contributions from circles. The vector $b$ is defined as

$$b = \left(0\,0\,\ldots\,0\ N_{1_{c_x}}\ N_{1_{c_y}}\ N_{1_{c_z}}\ O_{1_{c_x}}\ O_{1_{c_y}}\ O_{1_{c_z}}\ \ldots\ N_{K_{c_x}}\ N_{K_{c_y}}\ N_{K_{c_z}}\ O_{K_{c_x}}\ O_{K_{c_y}}\ O_{K_{c_z}}\right)^t. \quad (25)$$

Given $W$ and $b$, the least-squares solution for $V$ is

$$V = (W^t W)^{-1} W^t b. \tag{26}$$

It can be seen that to have an overdetermined system of linear equations, we need $2K+2J+6I \geq 12$ observed geometric entities. This may occur with any combination of points, lines and circles. For example, one point, one line and one circle or two points and one circle are sufficient to solve the transformation matrix from Eq. (26). Any additional points or lines or circles will improve the robustness and the precision of the estimated parameters.

### 6.2. Approximately Imposing Orthonormal Constraints

The least-squares solution to $V$ described in the last section cannot guarantee the orthonormality of the resultant rotation matrix. One major reason why previous linear methods are very susceptible to noise is because the orthonormality constraints are not enforced or enforced weakly. To ensure this, the six orthonormal constraints must be imposed on $V$ within the least-squares framework. The conventional way of imposing the constraints is through the use of the Lagrange multiplier. However, simultaneously imposing any two normality constraints or one orthogonality constraint using the Lagrange multiplier requires a nonlinear solution for the problem. Therefore, most linear methods choose to use a single orthonormal constraint. For example, Faugeras[6] imposed the constraint that the norm of the last row vector of $R$ be unity. This constraint, however, cannot ensure complete satisfaction of all orthonormal constraints. To impose more than one orthonormality constraint but still retain a linear solution, Liu *et al.*[21] suggested the constraint that the sum of the squares of the three row vectors be 3. This constraint, however, cannot guarantee the normality of each individual row vector. Haralick *et al.*[12] and Horn[16] proposed direct solutions, where all orthonormality constraints are imposed, but for the 3D to 3D absolute orientation problem. They are only applicable while using point correspondence and are not applicable to line and circle–ellipse correspondence. Most important, their techniques cannot be applied to the general linear framework we propose.

Given the linear framework, even imposing one constraint using the Lagrange multiplier can render the solution nonlinear. It is well-known that the solution to minimizing $||Ax - b||$ subject to $s(x) = 0$ can only be achieved by the nonlinear method. In statistics, this kind of problem is called *trust region*. Existing solutions to trust region problems are all nonlinear.

We now introduce a simple yet effective method for approximately imposing the orthonormal constraints in a way that offers a linear solution. We want to emphasize that the technique we are about to introduce cannot guarantee a perfect rotation matrix. However our experimental study proves that it yields a matrix that is closer to a rotation matrix than those obtained using the competing methods. The advantages of our technique are:

(1) all six orthonormal constraints are imposed simultaneously;

(2) constraints are globally imposed on each entry of the rotation matrix rather than locally, and
(3) asymptotically, the resulting matrix should converge to a rotation matrix.

We now address the problem of how to impose the orthonormality constraints in the general framework of finding pose from multiple geometric features described in Secs. 3–5. Given the pose of circles relative to the camera frame and the object frame, let $N_c = (N_{c_x}\ N_{c_y}\ N_{c_z})^t$ and $N_o = (N_{o_x}\ N_{o_y}\ N_{o_z})^t$ be the 3D circle normals in camera and object frames respectively. Equation (20) depicts the relation between two normals that involves $R$. The relation can also be expressed in an alternative way that involves $R^t$ (note $R^t = R^{-1}$) as follows

$$\begin{aligned} N_o &= R^t N_c \\ &= \begin{pmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{pmatrix} \begin{pmatrix} N_{c_x} \\ N_{c_y} \\ N_{c_z} \end{pmatrix}. \end{aligned} \tag{27}$$

Equivalently, we can rewrite Eq. (27) as follows

$$\begin{aligned} N_{c_x} r_{11} + N_{c_y} r_{21} + N_{c_z} r_{31} &= N_{o_x} \\ N_{c_x} r_{12} + N_{c_y} r_{22} + N_{c_z} r_{32} &= N_{o_y} \\ N_{c_x} r_{13} + N_{c_y} r_{23} + N_{c_z} r_{33} &= N_{o_z}. \end{aligned}$$

Given the same set of $I$ observed ellipses and their corresponding space circles, we can set up another system of linear equations that uses the same set of circles as in $Q$. Let $Q'$ be the coefficient matrix that contains the coefficients of the set of linear equations, then $Q'$ is

$$Q' = \begin{pmatrix} N_{1_{c_x}} & 0 & 0 & N_{1_{c_y}} & 0 & 0 & N_{1_{c_z}} & 0 & 0 & 0\ 0\ 0 \\ 0 & N_{1_{c_x}} & 0 & 0 & N_{1_{c_y}} & 0 & 0 & N_{1_{c_z}} & 0 & 0\ 0\ 0 \\ 0 & 0 & N_{1_{c_x}} & 0 & 0 & N_{1_{c_y}} & 0 & 0 & N_{1_{c_z}} & 0\ 0\ 0 \\ \vdots & & & & & & & & & \\ N_{I_{c_x}} & 0 & 0 & N_{I_{c_y}} & 0 & 0 & N_{I_{c_z}} & 0 & 0 & 0\ 0\ 0 \\ 0 & N_{I_{c_x}} & 0 & 0 & N_{I_{c_y}} & 0 & 0 & N_{I_{c_z}} & 0 & 0\ 0\ 0 \\ 0 & 0 & N_{I_{c_x}} & 0 & 0 & N_{I_{c_y}} & 0 & 0 & N_{I_{c_z}} & 0\ 0\ 0 \end{pmatrix}. \tag{28}$$

Correspondingly, we have $k'$ defined as

$$k' = (N_{1_{o_x}}\ N_{1_{o_y}}\ N_{1_{o_z}}\ 0\ 0\ 0 \ldots\ N_{I_{o_x}}\ N_{I_{o_y}}\ N_{I_{o_z}}\ 0\ 0\ 0)^t. \tag{29}$$

To implement the constraint in the least-squares framework, we can augment matrix $W$ in Eq. (26) with $Q'$, yielding $W'$, and augment vector $b$ in Eq. (25) with $k'$, yielding $b'$, where $W'$ and $b'$ are defined as follows

$$W' = \begin{pmatrix} W \\ Q' \end{pmatrix} \quad b' = \begin{pmatrix} b \\ k' \end{pmatrix}.$$

Putting it all together, the solution to $V$ can be found by minimizing $||W'V - b'||^2$, given by

$$V = (W'^t W')^{-1} W'^t b' . \tag{30}$$

The resultant transformation parameters $R$ and $T$ are more accurate and robust due to fusing information from different sources. The resultant rotation matrix $R$ is also very close to being orthonormal since the orthonormality constraints have been implicitly added to the system of linear equations used in the least-squares estimation. The obtained $R$ and $T$ can be used directly for certain applications or fed to an iterative procedure to refine the solution. Since the obtained transformation parameters are accurate, the subsequent iterative procedure can converge very quickly, usually after a couple of iterations as evidenced by our experiments. In the section to follow, we study the performance of the new linear pose estimation method against the methods that only use one type of geometric entity at a time, using synthetic and real images.

## 7. EXPERIMENTS

In this section, we present and discuss the results of a series of experiments aimed at characterizing the performance of the integrated linear pose estimation technique. Using both synthetic data and real images of industrial parts, the experiments conducted are to study the effectiveness of the proposed technique for approximately imposing the orthonormal constraints and to quantitatively evaluate the performance of the integrated linear technique against the existing linear techniques.

### 7.1. Performance Study with Synthetic Data

This section consists of two parts. First, we present results from a large number of controlled experiments aimed at analyzing the effectiveness of our technique for imposing orthonormal constraints. This was accomplished by comparing the errors of the estimated rotation and translation vectors obtained with and without orthonormal constraints imposed under different conditions. Second, we discuss the results from a comparative performance study of the integrated linear technique against an existing linear technique under different noisy conditions.

In the experiments with simulation data, the 3D data (3D point coordinates, 3D surface normals, 3D line direction cosines) are generated randomly within specified ranges. For example, 3D coordinates are randomly generated within the cube $[(-5,-5,-5)$ to $(5,5,5)]$. 2D data are generated by projecting the 3D data onto the image plane, followed by perturbing the projected image data with iid Gaussian distributed noise of mean 0 and standard deviation $\sigma$. From the generated 2D–3D data, we estimate the rotation and the translation vector using the linear algorithm, from which we can compute the estimation errors. The estimation error is defined as the Euclidean distance between the estimated rotation (translation) vector and the ideal rotation (translation) vector. We choose the rotation matrix (vector) rather than other specific representations like Euler angles and quaternion

parameters for error analysis. This is because all other representations depend on the estimated rotation vector. For each experiment, 100 trials are performed and the average distance errors are computed. The noise level is quantified using signal to noise ratio (SNR). SNR is defined as $20 \log \frac{d}{\sigma}$, where $\sigma$ is the standard deviation of the Gaussian noise and $d$ is the range of the quantity being perturbed.

Figures 4 and 5 plot the mean rotation and translation errors as a function of the signal to noise ratio, with and without orthonormal constraints imposed. It is clear from the two figures that imposing the orthonormal constraints improves the estimation errors for both the rotation and translation vectors. The improvement is especially significant when the SNR is low.

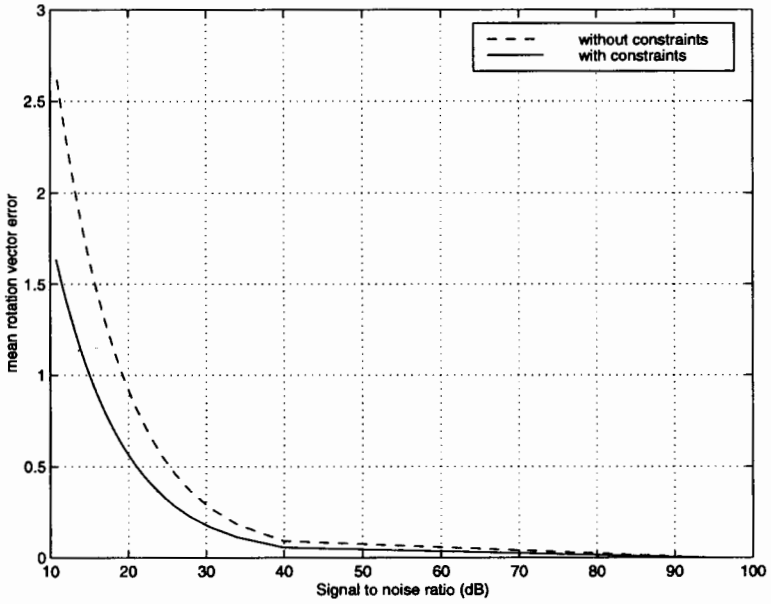


Fig. 4. Average rotation vector error versus SNR. The plot was generated using the integrated linear technique with a combination of three points, one line and one circle. Each point represents an average of 100 trials.

To further study the effectiveness of the technique for imposing constraints, we studied its performance under different numbers of pairs of ellipse/circle correspondences. This experiment is intended to study the efficacy of imposing orthonormal constraints versus the amount of geometric data used for the least-squares estimation. The results are plotted in Figs. 6 and 7, which give the average rotation and translation errors as a function of the number of ellipse/circle pairs used, with and without constraints imposed. The two figures again show that imposing orthonormal constraints leads to an improvement in estimation errors. This improvement, however, begins to taper off when the amount of data used exceeds a certain threshold. The technique is most effective when fewer ellipse/circle pairs are used. This echos the conclusion drawn from the previous two figures.

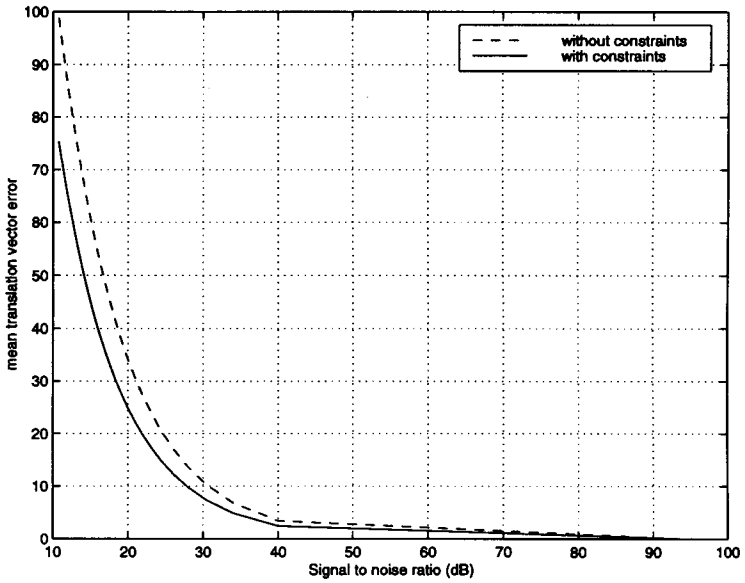


Fig. 5. Average translation vector error (mm) versus SNR. The plot was generated using the integrated linear technique with a combination of three points, one line and one circle. Each point represents an average of 100 trials.

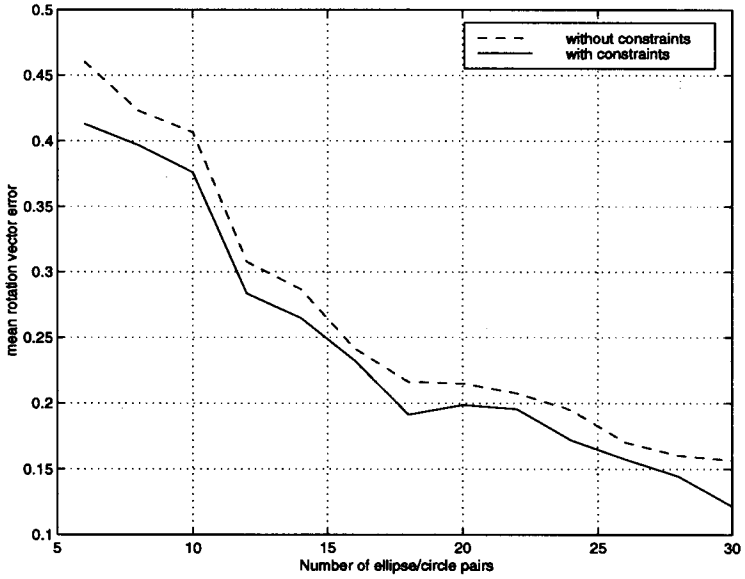


Fig. 6. Mean rotation vector error versus the number of ellipse/circle pairs (SNR = 35).

To compare the integrated linear technique with an existing linear technique, we studied its performance against that of Faugeras.[6] The results are given in Figs. 8 and 9, which plot the mean rotation and translation vector errors as a function of the SNR respectively.

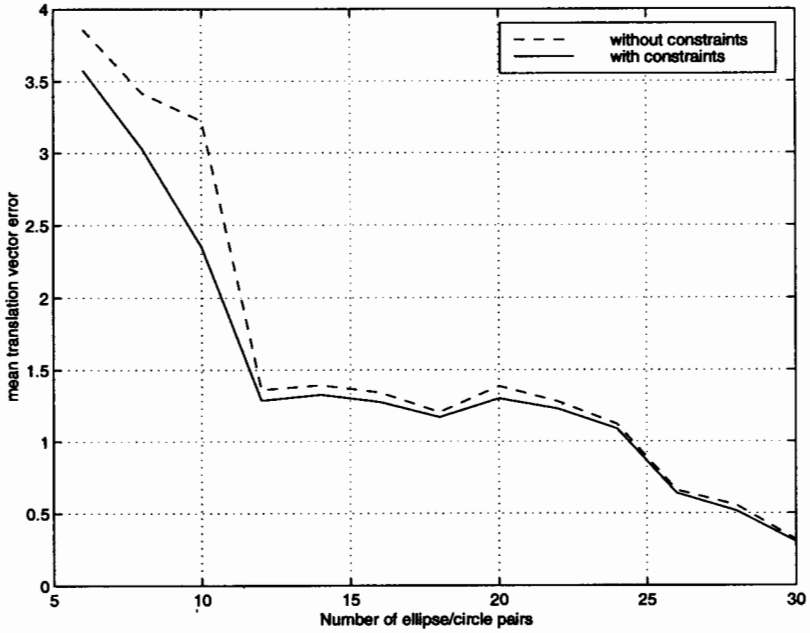


Fig. 7. Mean translation vector error (mm) versus the number of ellipse/circle pairs (SNR = 35).

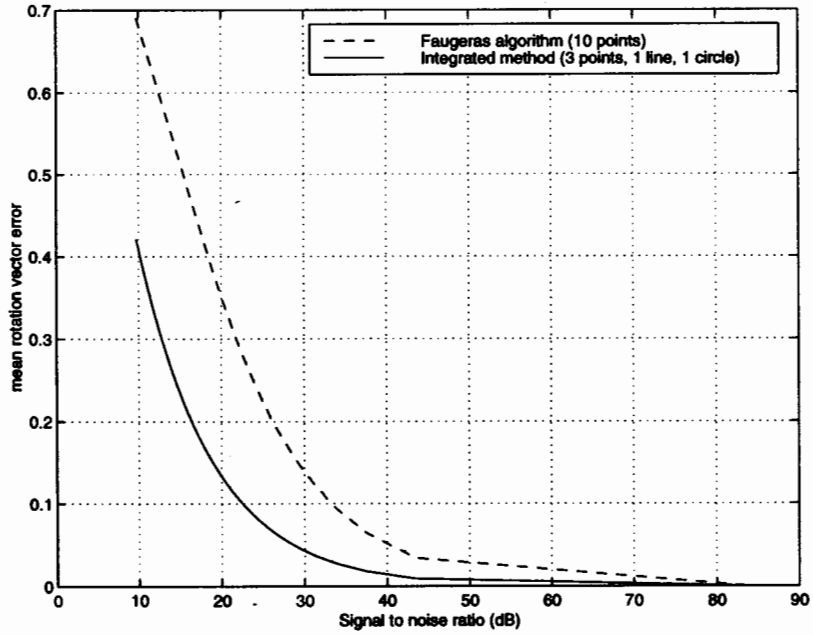


Fig. 8. Mean rotation vector error versus SNR. The curve for Faugeras' was obtained using 10 points while the curve for the integrated technique was generated using a combination of three points, one line and one circle.

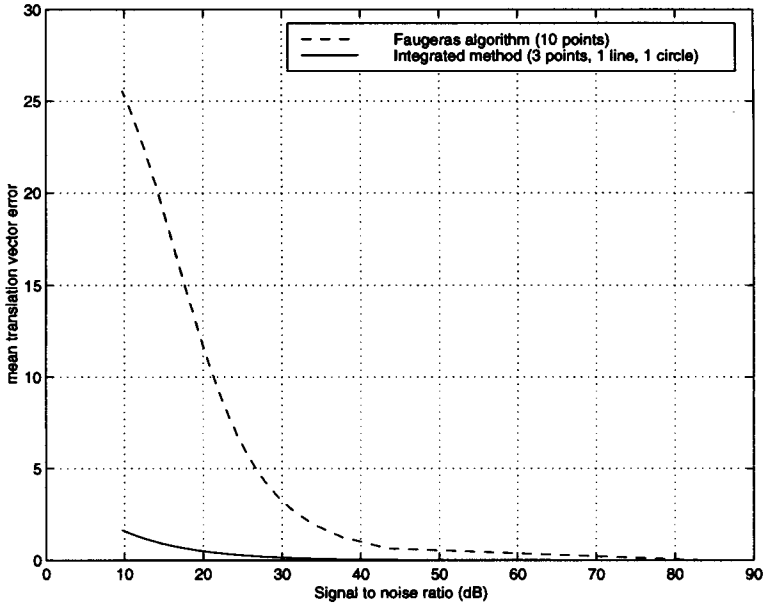


Fig. 9. Mean translation vector error (mm) versus SNR. The curve for Faugeras' was obtained using 10 points while the curve for the integrated technique was generated using a combination of three points, one line and one circle.

The two figures clearly show the superiority of the new integrated technique over Faugeras' linear technique, especially for the translation errors. To further compare the sensitivity of the two techniques to viewing parameters, we changed the position parameters of the camera by increasing $z$. Figures 10 and 11 plot the mean rotation and translation vector errors as a function of SNR respectively under the new camera position. While increasing $z$ causes an increase in the estimation errors for both techniques, its impact on Faugeras' technique is more serious. This leads to a much more noticeable performance difference between the two linear techniques. The fact that the integrated technique using only five geometric entities (3 points, 1 line and 1 circle) still outperforms Faugeras' technique, which uses 10 points, shows that the higher-level geometric features such as lines and circles can provide more robust solutions than those provided solely by points. This demonstrates the power of combining features on different levels of abstraction. Our study also shows that Faugeras' linear technique is very sensitive to noise when the number of points used is close to the required minimum. For example, when only six points are used, a small perturbation of the input data can cause significant errors on the estimated parameters, especially the translation vector. Figures 10 and 11 reveal that the technique using only points is numerically unstable to viewing parameters.

### 7.2. Performance Characterization with Real Images

This section presents results obtained using real images of industrial parts. The images contain linear features (points and lines) and nonlinear features (circles).

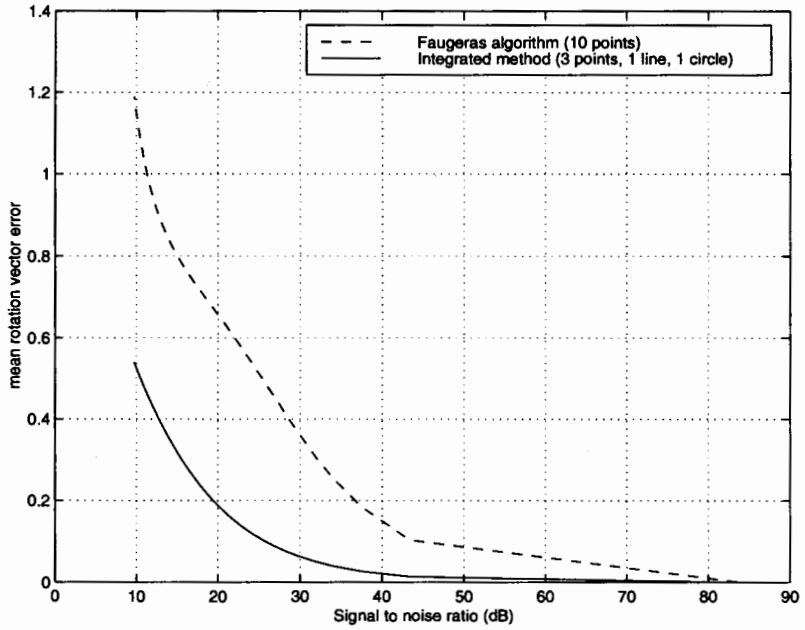


Fig. 10. Mean rotation vector error versus SNR with an increased camera position parameter $z$. The curve for Faugeras' was obtained using 10 points while the curve for the integrated technique was generated using a combination of three points, one line and one circle.

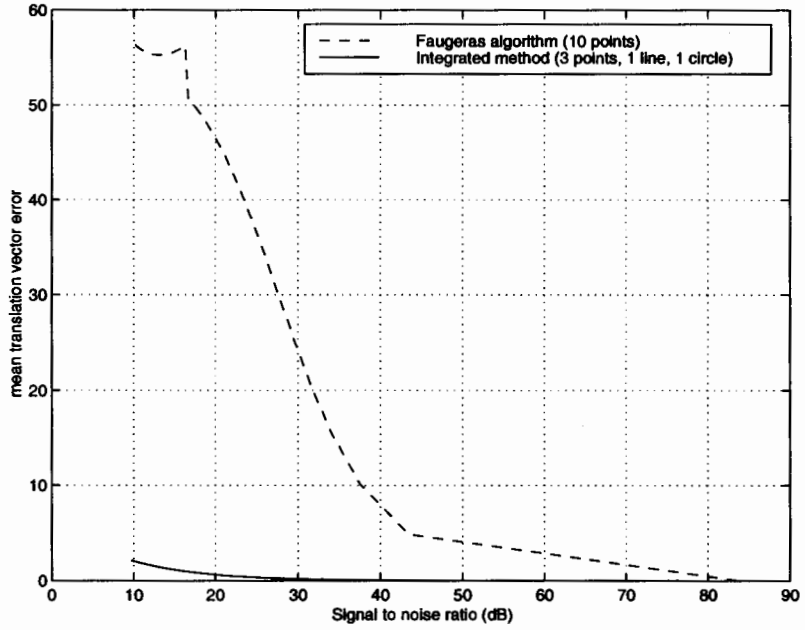


Fig. 11. Mean translation vector error (mm) versus SNR with an increased camera position parameter $z$. The curve for Faugeras' was generated using 10 points while the curve for the integrated technique was generated using a combination of three points, one line and one circle.

Fig. 12. The pose computed using two points (as indicated by the black squares) and one circle (the upper circle) with the integrated technique.

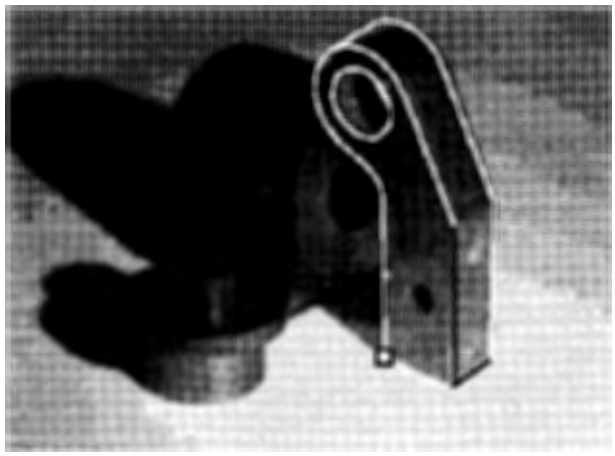

Fig. 13. The pose computed using one point, two lines and one circle (the upper circle) with the integrated technique. The points and lines used are marked with black squares and lines.

This phase of the experiments consist of two parts. First, we visually assess the performance of the linear least-squares framework using different combinations of geometric entities, such as one circle and six points; one circle and two points; and one circle, one point and two lines. The performance of the proposed technique is judged by visual inspection of the alignment between the image of a part and the reprojected outline of the part using the estimated transformation matrix. Second, the technique is compared against existing methods that use only one type of geometric entity as well as against the Gauss–Newton iterative method. The closeness between the solutions from the linear method and from the iterative method, as represented by the residual errors, as well as the number of iterations required, are

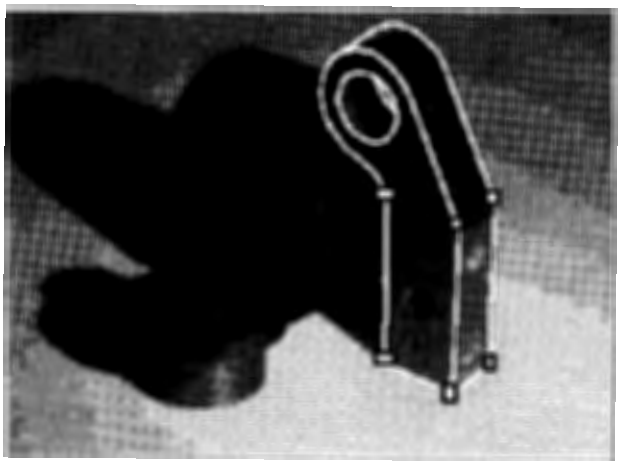

Fig. 14. The pose computed using six points (as indicated by the black squares) and one circle (the upper circle) with the integrated technique.

used as measures to indicate the goodness of the solution obtained using the new method.

To test the integrated technique, we performed the following experiments. The pose transformation ($RT$) was analytically computed using our integrated technique with different combinations of points, lines and circles. Figures 12–14 illustrate the results obtained using the following combinations of features: two points plus one circle; one point, two lines and one circle; and six points and one circle respectively. The visual inspection of the figures reveals that results obtained from the three configurations are all good enough to serve as an initial guess to an iterative procedure. It is also evident from Figs. 12–14 that the result using six points and one circle is superior to the ones obtained using the other two configurations.

The significance of these sample results is as follows. First, they demonstrate the feasibility of the proposed framework applied to real image data. Second, they show that using multiple geometric primitives simultaneously to compute the pose reduces the dependency on points. One can be more selective when choosing which point correspondence to use in pose estimation. This can potentially improve the robustness of the estimation procedure since points are more susceptible to noise than lines and circles. Third, the use of more than the minimum required number of geometric features provides redundancy to the least-squares estimation, therefore improving the accuracy of the solution, as evidenced by the progressively improved results as the number of linear equations increase.

In order to compare the results with those of other existing techniques, we computed the pose of the same object using the same six points and the same circle, separately. The result for the pose computation using a linear technique[18] (similar to that of Faugeras[5]) with six points is given in Fig. 15. The algorithms of Dhome[3] and Forsyth[8] for the pose-from-circle computation were augmented in Ref. 2 to handle nonrotationally symmetric objects. The results of this augmented

algorithm using the single ellipse/circle correspondence is shown in Fig. 16. Notice that due to the localized concentration of detectable feature points and the physical distance between the circle and these points, the poses computed align well only in the areas where the features used are located. Specifically, the result in Fig. 16 shows a good alignment in the upper portion of the object where the circle is located and a poor alignment in the lower part (as indicated by the arrow). On the other hand, the result in Fig. 15 shows good alignment only at the lower part of the object where the concentration of detectable feature points is located and a poor alignment on the upper part of the object (as indicated by the arrow).

Fig. 15. The pose computed using six points alone. It shows good alignment only at the lower part of the object where the concentration of detectable feature points is located and a poor alignment on the upper part of the object (as indicated by the arrow).

Fig. 16. The pose computed using a single circle. It shows a good alignment in the upper portion of the object where the circle is located and a poor alignment in the lower part (as indicated by the arrow).

Visual inspection of the results in Figs. 14–16 shows the superiority of the new technique over the existing methods. The model reprojection using the transformation matrix obtained using the new technique yields a better alignment than those using only points or only ellipse/circle correspondences. To compare the performance quantitatively, we compare the transformation matrices obtained using the three methods against the one obtained from the iterative procedure. Table 1 shows the numerical results for the transformations obtained from using only points, only the circle, and a combination of points and circle. The results from each method were then used as the initial guess to the iterative Gauss–Newton method. The final transformation obtained after the convergence of the iterative method is shown in the last row of Table 1. These final results are the same regardless of which initial guess was used.

Table 1. Pose transformations from different methods.

| Methods | R | T |
|---|---|---|
| Point only | $\begin{bmatrix} 0.410 & -0.129 & -0.902 \\ 0.606 & -0.700 & 0.376 \\ -0.681 & -0.701 & -0.208 \end{bmatrix}$ | [−43.125 − 25.511 1232.036] |
| Circle only | $\begin{bmatrix} 0.302 & 0.302 & -0.932 \\ 0.692 & -0.628 & 0.355 \\ -0.655 & -0.753 & -0.054 \end{bmatrix}$ | [−35.161 − 15.358 1195.293] |
| Points and circle | $\begin{bmatrix} 0.398 & -0.142 & -0.902 \\ 0.554 & -0.667 & 0.336 \\ -0.700 & -0.684 & -0.201 \end{bmatrix}$ | [−43.077 − 26.400 1217.855] |
| Nonlinear | $\begin{bmatrix} 0.341 & -0.156 & -0.927 \\ 0.631 & -0.693 & 0.349 \\ -0.697 & -0.704 & -0.137 \end{bmatrix}$ | [−43.23 − 28.254 1273.07] |

Table 2. Performance of different methods.

| Method | Number of iterations |
|---|---|
| Points only | 4 |
| Circle only | 6 |
| Points and circle | 1 |

Table 2 summarizes the number of iterations required for the iterative procedure to converge using as initial guesses the results from the three linear methods mentioned in Table 1. Figure 17 shows the results from the iterative procedure.

It is evident from Tables 1 and 2 and Fig. 17 that the new technique yields a transformation matrix that is closer to the one obtained from the iterative procedure and therefore requires fewer iterations (one here) for the iterative method to converge. By contrast, the final results for initial guesses obtained using only points and only one circle require 4 and 6 iterations, respectively, for the iterative procedure to converge. The result from the quantitative study echoes the conclusion from visual inspection: the new technique offers better estimation accuracy.

Fig. 17. The final pose obtained from the Gauss–Newton iterative method after only one iteration using the solution in Fig. 14 as the initial solution.

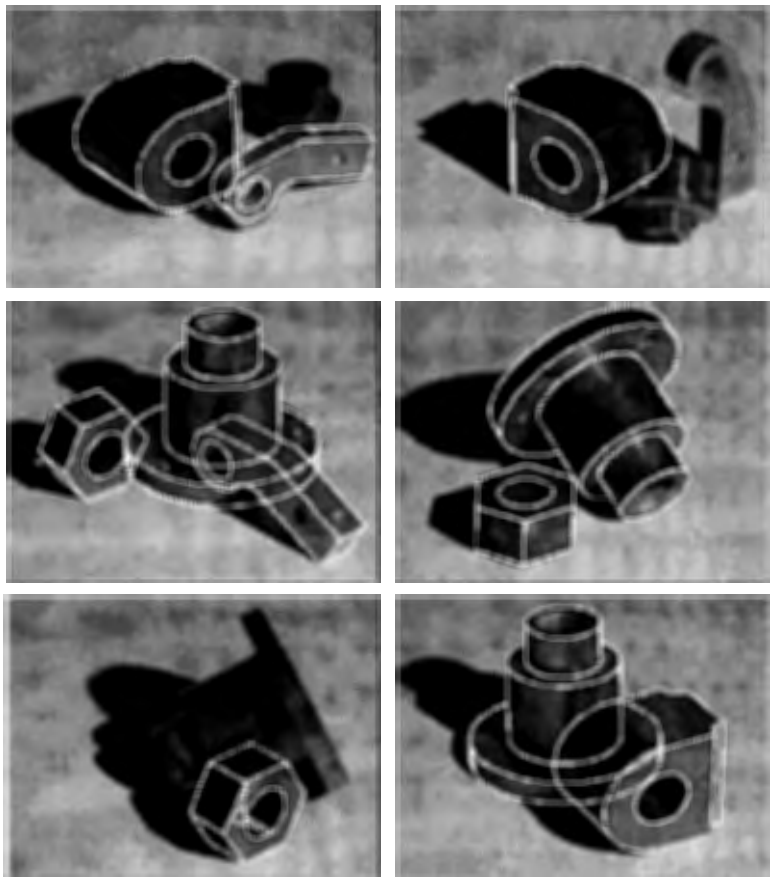

Fig. 18. Results of the integrated method applied to different industrial parts.

To further validate our technique, we tested it on over fifty real images with similar results. Figure 18 gives results of applying the integrated technique to different industrial parts with different combinations of geometric features. Our experiments also reveal, as was theoretically expected for a system of linear equations of the form $AX = b$, a decay in robustness when the number of equations in the linear system reaches the minimum required for a solution to be found. The premise is that one should make use of as many available features as possible in order to improve accuracy and robustness. Our technique allows for that.

## 8. SUMMARY

In this paper, we present a linear solution to the exterior camera parameters or pose estimation. The main contributions of this research are the linear framework for fusing information available from different geometric entities and for introducing a novel technique that approximately imposes the orthonormality constraints on the rotation matrix sought. Experimental evaluation using both synthetic data and real images show the effectiveness of our technique for imposing orthonormal constraints in improving estimation errors. The technique is especially effective when the SNR is low or fewer geometric entities are used. The performance study also revealed superiority of the integrated technique to a competing linear technique in terms of robustness and accuracy. The new technique proposed in this paper is ideal for applications such as industrial automation where robustness, accuracy, computational efficiency and speed are needed. Its results can also be used as initial estimates in certain applications where more accurate camera parameters are needed.

## REFERENCES

1. S.-Y. Chen and W.-H. Tsai, "Systematic approach to analytic determination of camera parameters by line features," *Patt. Recogn.* **23**, 8 (1990) 859–897.
2. M. S. Costa, "Object recognition and pose estimation using appearance-based features and relational indexing," Ph.D. Dissertation, University of Washington, Seattle, 1997.
3. M. D. Dhome, J. T. Lapreste, G. Rives and M. Richetin, "Spatial localization of modeled objects of revolution in monocular perspective vision," *1st European Conf. Computer Vision*, 1989, pp. 415–485.
4. T. Echigo, "Camera calibration technique using three sets of parallel lines," *Mach. Vis. Appl.* **3**, 3 (1990) 159–167.
5. O. Faugeras, *Three-Dimensional Computer Vision*, The MIT Press, Cambridge, Massachusetts, 1993.
6. O. D. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, 1993.
7. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM* **24**, 6 (1981) 381–395.
8. D. Forsyth, J. L. Mundy, A. Zisserman *et al.*, "Invariant descriptors for 3D object recognition and pose," *IEEE Trans. Patt. Anal. Mach. Intell.* **13**, 10 (1991) 971–991.
9. M. Gao, "Estimating camera parameters from projection of a rectangular parallelepiped," *J. Northwestern Polytechnic Univ.* **10**, 4 (1992) 427–433.

10. R. M. Haralick, "Determining camera parameters from the perspective projection of rectangle," *Patt. Recogn.* **22** (1988) 225–230.
11. R. M. Haralick and Y. H. Chu, "Solving camera parameters from perspective projection of a parameterized curve," *Patt. Recogn.* **17**, 6 (1989) 637–645.
12. R. M. Haralick, H. Joo, C. Lee, X. Zhang, V. Vaidya and M. Kim, "Pose estimation from corresponding point data," *IEEE Trans. Syst. Man Cynern.* **19**, 6 (1989) 1426–1446.
13. R. M. Haralick, C. Lee, K. Ottenberg and M. Nolle, "Review and analysis of solutions of the three-point perspective pose estimation," *Int. J. Comput. Vis.* **13**, 3 (1994) 331–356.
14. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Vol. 2, Addison-Wesley Publ. Co., 1993.
15. R. J. Holt and A. N. Netravali, "Camera calibration problem: some new results," *Comput. Vis. Graph. Imag. Process.* **54**, 3 (1991) 368–383.
16. B. K. P. Horn, "Closed-form solution of absolute orientation using quaternions," *J. Opt. Soc. Am.* **A4** (1987) 629–642.
17. Y. Hung, P. S. Yeh and D. Harwood, "Passive ranging to known planar point sets," *Proc. IEEE Int. Conf. Robotics and Automation*, 1985, pp. 80–85.
18. Q. Ji and M. S. Costa, "New linear techniques for pose estimation using point correspondence," Intelligent Systems Lab Technical Report #ISL-04-97.
19. B. Kamgar-Parsi and R. D. Eas, "Calibration of stereo system with small relative angles," *Comput. Vis. Graph. Imag. Process.* **51**, 1 (1990) 1–19.
20. R. Kumar and A. R. Hanson, "Robust estimation of camera location and orientation from noisy data having outliers," *IEEE Workshop on Interpretation of 3D Scenes*, 1989.
21. Y. Liu, T. Huang and O. D. Faugeras, "Determination of camera locations from 2D to 3D line and point correspondence," *IEEE Trans. Patt. Anal. Mach. Intell.* **12**, 1 (1990) 28–37.
22. T. Q. Phong, R. Horward, A. Yassine and P. Tao, "Object pose from 2D to 3D point and line correspondence," *Int. J. Comput. Vis.* **15** (1995) 225–243.
23. C. A. Rothwell, A. Zisserman, C. I. Marinos, D. Forsyth and J. L. Mundy, "Relative motion and pose from arbitrary plane curves," *Imag. Vis. Comput.* **10**, 4 (1992) 251–262.
24. D. M. Song, "Conics-based stereo, motion estimation and pose determination," *Int. J. Comput. Vis.* **10**, 1 (1993) 7–25.
25. I. E. Sutherland, "Three-dimensional data input by tablet," *Proc. IEEE* **62**, 4 (1994) 453–461.
26. R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lens," *IEEE J. Robot. Autom.* **3** (1987) 223–244.
27. X. Wang and G. Xu, "Camera parameters estimation and evaluation in active vision system," *Patt. Recogn.* **29**, 3 (1996) 439–447.

**Qiang Ji** received a M.S. degree from the University of Arizona and his Ph.D. degree in electrical engineering from the University of Washington in 1998. As a research engineer with Western Research Company, Tucson, Arizona, Dr. Ji has served as a principle investigator on several NIH funded research projects to develop computer vision and pattern recognition algorithms for biomedical applications. In summer 1995, as a visiting technical staff with the Robotics Institute, Carnegie Mellon University, he developed computer vision algorithms for industrial inspection. From 1995 to 1998, he worked at the Intelligent Systems Laboratory (ISL) at the University of Washington, and was involved in a Boeing-funded research project consisting of developing computer vision techniques for 3D geometric tolerancing of manufactured parts from their images. Dr. Ji is currently an assistant Professor at the Department of Computer Science at University of Nevada at Reno.

He has published numerous papers in refereed journals and conferences and his research has been funded by government agencies like NIH and NSF and companies like Boeing and HONDA. He currently serves as PI for a project funded by HONDA developing a computer vision system for monitoring driver's vigilance level.

His areas of research include computer vision, image processing, pattern recognition and robotics.

**Robert M. Haralick** received from the University of Kansas a B.A. degree in mathematics in 1964, a B.S. degree in electrical engineering in 1966, and a M.S. degree in electrical engineering in 1967. In 1969, after completing his Ph.D. at the University of Kansas, he joined the faculty of the Electrical Engineering Department where he served as Professor from 1975 to 1978.

In 1979, Dr. Haralick joined the Electrical Engineering Department at Virginia Polytechnic Institute and State University where he was a Professor and Director of the Spatial Data Analysis Laboratory. From 1984 to 1986, Dr. Haralick served as Vice President of Research at Machine Vision International, Ann Arbor, Michigan.

Dr. Haralick now occupies the Boeing Clairmont Egtvedt Professorship in the Department of Electrical Engineering at the University of Washington. Professor Haralick was elected Fellow of IEEE for his contributions in computer vision and image processing. He serves on the Editorial Boards of *Machine Vision and Applications and Real Time Imaging*, and he is an Associate Editor for *IEEE Transactions on Image Processing* and *Journal of Electronic Imaging*.

Prof. Haralick's recent work is in shape analysis and extraction using the techniques of mathematical morphology, and in robust pose estimation, techniques for making geometric inferences from perspective projection information, propagation of random perturbations through image analysis algorithms, and document analysis. He has developed the morphological sampling theorem that establishes a sound shape/size basis for the focus of attention mechanisms that can process image data in multiresolution mode, thereby resulting in the more efficient execution of some image feature extraction processes.

**Mauro S. Costa** earned his Bachelor's degree in electrical engineering from the Federal University of Para', Brazil, where he received the Kawaguchi Award of Excellence offered by the Mitsui Aluminum Company to the best graduating Senior in the College of Engineering. He received his Master's Degree from the University of Missouri-Rolla and his Ph.D. from the University of Washington, Seattle, both in electrical engineering. His doctoral research was in the area of feature-based object recognition and he worked in several areas such as affine matching, appearance-based matching, relational indexing, pose estimation, performance characterization, and mathematical morphology. Since 1997 he has been working with Boeing Commercial Airplanes in Seattle, in the area of STEP (Standard for the Exchange of Product Data) applied to manufacturing, and in the areas of imaging and scanning. His current work includes the definition of multi-axis manufacturing features to aid in the automation of the machining process, and the use of standards throughout the entire CAD-CAM-CNC manufacturing chain.

His research interests include feature-based manufacturing, real-time application of industrial machine vision techniques, image understanding and pattern recognition.

**Linda G. Shapiro** received the B.S. degree in mathematics from the University of Illinois, Urbana, in 1970, and the M.S. and Ph.D. degrees in computer science from the University of Iowa, Iowa City, in 1972 and 1974, respectively.

She was an Assistant Professor of Computer Science at Kansas State University, Manhattan, from 1974 to 1978 and was an Assistant Professor of Computer Science from 1979 to 1981 and Associate Professor of Computer Science from 1981 to 1984 at Virginia Polytechnic Institute and State University, Blacksburg. She was Director of Intelligent Systems at Machine Vision International in Ann Arbor from 1984 to 1986. She is currently Professor of Computer Science and Engineering and of Electrical Engineering at the University of Washington.

Dr. Shapiro is a Fellow of the IEEE and a member of the Pattern Recognition Society and the American Association for Artificial Intelligence. She is currently an editorial board member of *Pattern Recognition* and of *Computer Vision and Image Understanding*. She has served as Editor of *CVGIP: Image Understanding*, co-Program Chairman of the IEEE Conference on Computer Vision and Pattern Recognition in 1994, and General Chairman of the IEEE Workshop on Directions in Automated CAD-Based Vision in 1991, and the IEEE Conference on Computer Vision and Pattern Recognition in 1986. She has also served on the program committees of numerous vision and AI workshops and conferences.

Her research interests include computer vision, image database systems, artificial intelligence, pattern recognition and robotics. She has co-authored two textbooks, one on data structures and one on computer and robot vision.