Pattern Recognition Vol. 17, No. 6, pp. 637–645, 1984.
Printed in Great Britain.

0031–3203/84 \$3.00+ .00
Pergamon Press Ltd.

SOLVING CAMERA PARAMETERS FROM THE PERSPECTIVE PROJECTION OF A PARAMETERIZED CURVE

ROBERT M. HARALICK
Machine Vision International, Ann Arbor, MI 48104, U.S.A.

and

YU HONG CHU
Department of Computer Science, Old Dominion University, Norfolk, VA 23508, U.S.A.

(*Received* 17 *January* 1983; *in revised form* 20 *March* 1984; *received for publication* 11 *April* 1984)

Abstract—Given the two-dimensional perspective projection of a known conic or polygon of unknown size and position in three-dimensional space, we show how to determine the camera look angles relative to the plane where the curve lies. Separate cases are discussed for conic and polygons. We then show that this technique can be used to solve for any planar or nonplanar curves, as long as they can be represented in parametric forms.

Perspective projection Conics Polygons Camera parameters Optimization

1. INTRODUCTION

The problem of determining camera viewing angle parameters and camera translation parameters from 3D and 2D point correspondences is an important one in scene analysis. In this paper, we show how to decompose the one six-parameter problem to two three-parameter problems if the shape and size of the curve is known. The first problem is to solve for the camera look angle parameters. The second is to solve the translation parameters from these angles. We also show that our method does not require knowing the exact point correspondence as most other methods do.

Fischler and Bolles[1] show that, given the coordinates of three 3D points and the corresponding image points, then it is possible to compute the position of the camera, as well as its look angles. Watson and Shapiro[2] describe a method for recognizing different curves by solving camera position and look angles. They use periodic cubic splines to approximate curves and then compute the Fourier coefficients of the curvature function of the spline functions. The Fourier transform of the curvature function represents the shape. The match between a projection and curves in database is found by determining a set of camera parameters which minimizes the error between their shapes.

Haralick[3] shows that it is possible to determine the camera parameters from the observed perspective projection of a 3D rectangle of known size and unknown orientation and position in 3D space. In this paper, we generalize the techniques Haralick used to include conics, polygons and curves in parametric form. Sections 2–4 describe how to decompose the six-camera-parameter problem to one three-parameter search problem followed by an algebraic solution for the camera position parameters for projections of conics and polygons. Sections 5 and 6 discuss some possible extensions and the algorithm complexity.

2. HOW TO DECOMPOSE THE SIX-CAMERA-VIEWING-PARAMETER PROBLEM TO TWO THREE-PARAMETER PROBLEMS

Suppose we are given the perspective projection (image) of planar space curves, such as conics or line segments, and the parametric equations of these curves in three-dimensional space. We assume that the optic axis of the camera passes through the center of the image. The problem is to determine the six camera viewing parameters ($X1$, $Y1$, $Z1$, θ, ϕ, ψ), where ($X1$, $Y1$, $Z1$) is the camera lens position relative to the origin of the three-dimensional curve and (θ, ϕ, ψ) are the pan, tilt and swing angles which determine the look direction of the camera relative to the coordinate system of the three-dimensional curve. Details of this perspective geometry can be found in Haralick.[3] Without loss of generality, we can assume that the lens is at (0, 0, 0) and we will determine the look angle (θ, ϕ, ψ) of the camera and the absolute coordinate of the origin of any curve relative to the camera lens. Fig. 1 shows the geometry of the camera. Fig. 2 shows a worked example for the case of a rectangle.

We have already shown[4] that this problem can be

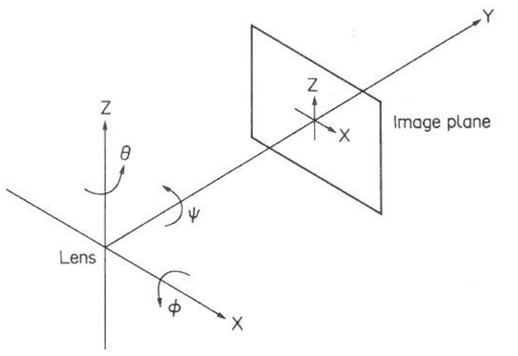

Fig. 1. The camera model. The lens is at the origin. The optical axis is the Y-axis and the image plane is parallel to the X–Z plane. Relative to the axis direction of a given three-dimensional curve, the camera coordinate system has been rotated θ around the Z-axis, ϕ around the X-axis and ψ around the Y-axis.

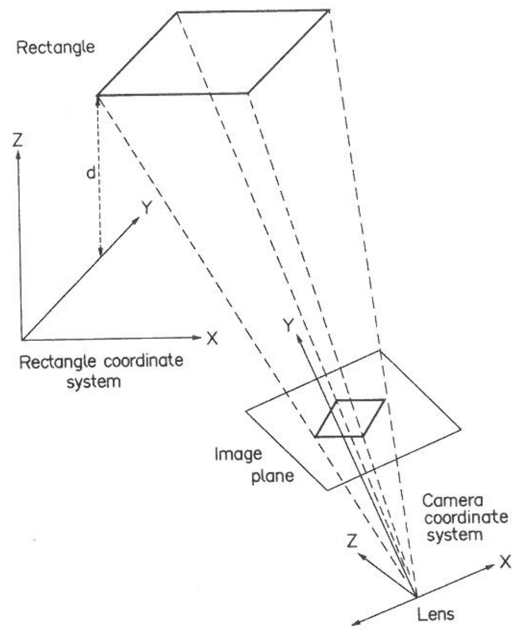

Fig. 2. A rectangle in its own coordinate system and a rotated and translated camera coordinate system viewing the rectangle.

solved as a nonlinear optimization problem with six variables. In the following, we will show that this problem can be decomposed into two parts. The first part is an optimization problem with three unknowns, namely the three look angle parameters (θ, ϕ, ψ). The second part is to solve for the translation parameters based on the camera angles computed in the first part. This decomposition is important because it reduces a six-parameter search to one three-parameter search of much smaller complexity, followed by an algebraic solution for the three-parameter specification of the absolute position of the conic.

The approach discussed here is more general than the previous one.[4] This approach does not need to know where the space curve lies (we need only know its form) and yet we can solve for all six parameters in the camera-oriented coordinate system.

Suppose the object image given is a projection of a conic, let the center of the image have coordinates (0, 0) and the upper right corner be (1, 1). The conic can be represented by a sequence of image points $\{(x_i, z_i)\}$. We will assume that this conic lies on a plane whose Z-coordinate is constant. Later we generalize it to planes of arbitrary orientation.

From Haralick,[3] we know that the ray of 3D points having (x_i, z_i) can be given by

$$\left\{\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \middle| \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \lambda_i \left| \begin{matrix} x_i' \cos\theta - f \sin\theta \cos\phi + z_i' \sin\theta \sin\phi \\ x_i' \sin\theta + f \cos\theta \cos\phi - z_i' \cos\theta \sin\phi \\ f \sin\phi + z_i' \cos\phi \end{matrix} \right| \right. \tag{1}$$

for some λ_i where

$$\left.\left| \begin{matrix} x_i' \\ z_i' \end{matrix} \right| = \left| \begin{matrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{matrix} \right| \left| \begin{matrix} x_i \\ z_i \end{matrix} \right| \right\}.$$

Since we are only dealing with planar curves, we can, without loss of generality, choose a coordinate system so that the plane has the form Z = constant in our coordinate system. For simplicity, we will assume that the parametric equations of the space curve are in standard form (i.e. no rotation of coordinates axis is needed) first. We will show in Section 3 that for curves not in standard form, similar results can be derived.

The parametric equations of a conic on a $Z = Z_0$ plane are given by

$$X(t) = X_0 + a\cos(t),$$
$$Y(t) = Y_0 + b\sin(t),$$
$$Z = Z_0 \text{ for an ellipse};$$
$$X(t) = X_0 + a\sec(t),$$
$$Y(t) = Y_0 + b\tan(t),$$

$Z = Z_0$ for a hyperbola with its axis parallel to the X-axis;

$$X(t) = X_0 + at^2,$$
$$Y(t) = Y_0 + 2at,$$

$Z = Z_0$ for a parabola with its axis parallel to the X-axis. The unknown point (X_0, Y_0, Z_0) specifies the absolute position of the conic relative to the lens position, which is at (0, 0, 0). The parameters a and b are assumed unknown. For a parabola and hyperbola with its axis parallel to the Y-axis, similar equations can be derived.

2.1. *The ellipse*

Since for every point (X, Y, Z) on the space curve there exists a parameter value t_i such that X, Y, Z can be expressed in terms of t_i, then for the i-th perspective projection point (x_i, z_i) of the space curve, there must exist parameter values t_i, λ_i satisfying

$$\begin{vmatrix} X_0 + a\cos(t_i) \\ Y_0 + b\sin(t_i) \\ Z_0 \end{vmatrix} = \lambda_i \begin{vmatrix} x_i'\cos\theta - f\sin\theta\cos\phi + z_i'\sin\theta\sin\phi \\ x_i'\sin\theta + f\cos\theta\cos\phi - z_i'\cos\theta\sin\phi \\ f\sin\phi + z_i'\cos\phi \end{vmatrix} \tag{2}$$

for some λ_i and t_i $(i = 1, \ldots, N)$ where

$$\begin{vmatrix} x_i' \\ z_i' \end{vmatrix} = \begin{vmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{vmatrix} \begin{vmatrix} x_i \\ z_i \end{vmatrix}.$$

For simplicity, let us assume

$A_i = f\sin\phi + z_i'\cos\phi$,

$B_i = x_i'\cos\theta - f\sin\theta\cos\phi + z_i'\sin\theta \sin\phi$,

$C_i = x_i'\sin\theta + f\cos\theta\cos\phi - z_i'\cos\theta \sin\phi$.

From the third component of equation (2) we have $\lambda_i = Z_0/A_i$. Substituting λ_i into the first and second components of (2) we have

$$\begin{aligned} B_jZ_0/A_j &= X_0 + a\cos(t_j), \\ C_jZ_0/A_j &= Y_0 + b\sin(t_j), \quad j = 1, \ldots, N. \end{aligned} \tag{3}$$

Subtracting equation (3) with $j = 1$ from equation (3) with $j = i$, we get

$$\begin{aligned} B_iZ_0A_1 - B_1Z_0A_i &= A_iA_1a\cos(t_i) - A_iA_1a\cos(t_1) \\ C_iZ_0A_1 - C_1Z_0A_i &= A_iA_1b\sin(t_i) - A_iA_1b\sin(t_1). \end{aligned} \tag{4}$$

Divide both sides of (4) by A_iA_1 and use the relation $\cos(t_i)^2 + \sin(t_i)^2 = 1$ to get

$$Z_0 = \frac{-2\{[B_i/(aA_i) - B_1/(aA_1)]\cos(t_1) + [C_i/(bA_i) - C_1/(bA_1)]\sin(t_1)\}}{[B_i/(aA_i) - B_1/(aA_1)]^2 + [C_i/(bA_i) - C_1/(bA_1)]^2}. \tag{5}$$

Setting $5.i = 5.2$ $i = 3, 4, \ldots, N$, we can eliminate Z_0 and obtain

$$\begin{aligned} &\frac{-2\{[B_i/(aA_i) - B_1/(aA_1)]\cos(t_1) + [C_i/(bA_i) - C_1/(bA_1)]\sin(t_1)\}}{[B_i/(aA_i) - B_1/(aA_1)]^2 + [C_i/(bA_i) - C_1/(bA_1)]^2} \\ &= \frac{-2\{[B_2/(aA_2) - B_1/(aA_1)]\cos(t_1) + [C_2/(bA_2) - C_1/(bA_1)]\sin(t_1)\}}{[B_2/(aA_2) - B_1/(aA_1)]^2 + [C_2/(bA_2) - C_1/(bA_1)]^2}. \end{aligned} \tag{6}$$

From (6) we regroup the terms of $\cos(t_i)$, $\sin(t_i)$ together.

Let

$$\begin{aligned} P_i = &\frac{B_i/(aA_i) - B_1/(aA_1)}{[B_i/(aA_i) - B_1/(aA_1)]^2 + [C_i/(bA_i) - C_1/(bA_1)]^2} \\ &- \frac{B_2/(aA_2) - B_1/(aA_1)}{[B_2/(aA_2) - B_1/(aA_1)]^2 + [C_2/(bA_2) - C_1/(bA_1)]^2}. \end{aligned}$$

Let

$$\begin{aligned} Q_i = &\frac{-C_i/(aA_i) - C_1/(aA_1)}{[B_i/(aA_i) - B_1/(aA_1)]^2 + [C_i/(bA_i) - C_1/(bA_1)]^2} \\ &+ \frac{C_2/(aA_2) - C_1/(aA_1)}{[B_2/(aA_2) - B_1/(aA_1)]^2 + [C_2/(bA_2) - C_1/(bA_1)]^2}. \end{aligned}$$

Now equation (6) is equivalent to

$$\cos(t_1)P_i = \sin(t_1)Q_i.$$

Squaring both sides and simplifying it, we have

$$\cos(t_1)^2 = Q_i^2/(P_i^2 + Q_i^2), \quad i = 3, 4, 5 \ldots N.$$

Setting equation $6.i = 6.3$, we obtain

$$Q_i^2/(P_i^2 + Q_i^2) = Q_3^2/(P_3^2 + Q_3^2), \quad i = 4, 5, \ldots, N. \tag{7}$$

Equation (7) has three unknowns: θ, ϕ, ψ.

We can now apply an optimization technique to solve equation (7) for the three angle parameters θ, ϕ, ψ. After they are solved, by equation (6), we can solve for t_1.

$$\cos(t_1) = \pm[Q_i^2/(P_i^2 + Q_i^2)]^{0.5},$$

$$\sin(t_1) = \pm[P_i^2/(P_i^2 + Q_i^2)]^{0.5}.$$

Substituting this value for $\cos(t_1)$ and $\sin(t_1)$ into (5), we obtain

$$Z_0 = \frac{-2[(B_i/(aA_i) - B_1/(aA_1))\cos(t_1) + (C_i/(bA_i) - C_1/(bA_1))\sin(t_1)]}{[B_i/(aA_i) - B_1/(aA_1)]^2 + [C_i/(bA_i) - C_1/(bA_1)]^2}.$$

From (1), $\lambda_1 = Z_0/A_1$. Hence,

$$X_0 = -a\cos(t_1) + \lambda_1 B_1,$$
$$Y_0 = -b\sin(t_1) + \lambda_1 C_1.$$

For the case where only the shape is known, the parameters a and b are unknown. The above derivation is still true, except that the unknown optimization parameters are the three angle parameters and the major and minor axis length. For a parabola, the last two unknowns can be reduced to one directrix length parameter.

2.2. *The hyperbola*

For every point (X, Y, Z) on the hyperbola there exists a parameter t_i such that X, Y, Z can be expressed in terms of it. Equation (1) can then be rewritten as

$$\begin{vmatrix} X_0 + a\sec(t_i) \\ Y_0 + b\tan(t_i) \\ Z_0 \end{vmatrix} = \lambda_i \begin{vmatrix} x_i'\cos\theta - f\sin\theta\cos\phi + z_i'\sin\theta\sin\phi \\ x_i'\sin\theta + f\cos\theta\cos\phi - z_i'\cos\theta\sin\phi \\ f\sin\phi + z_i'\cos\phi \end{vmatrix} \quad (2')$$

for some λ_i and t_i $(i = 1, \ldots, N)$, where

$$\begin{vmatrix} x_i' \\ z_i' \end{vmatrix} = \begin{vmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{vmatrix} \begin{vmatrix} x_i \\ z_i \end{vmatrix}.$$

From the third component of equation (2′) we have $\lambda_i = Z_0/A_i$. Substituting λ_i into the first and second components of (2) we have

$$B_j Z_0/A_j = X_0 + a\sec(t_j), \quad (3')$$
$$C_j Z_0/A_j = Y_0 + b\tan(t_j), \quad j = 1, \ldots, N.$$

Subtracting equation (3′) with $j = 1$ from equation (3′) with $j = i$, we get

$$B_i Z_0 A_1 - B_1 Z_0 A_i = A_i A_1 a\sec(t_i) - A_i A_1 a\sec(t_1)$$
$$C_i Z_0 A_1 - C_1 Z_0 A_i = A_i A_1 b\tan(t_i) - A_i A_1 b\tan(t_1). \quad (4')$$

Dividing both sides of (4′) by $A_i A_1$ and using $\sec(t_i)^2 - \tan(t_i)^2 = 1$ we get

$$Z_0 = \frac{-2\{[B_i/(aA_i) - B_1/(aA_1)]\sec(t_1) - [C_i/(bA_i) - C_1/(bA_1)]\tan(t_1)\}}{[B_i/(aA_i) - B_1/(aA_1)]^2 - [C_i/(bA_i) - C_1/(bA_1)]^2}. \quad (5')$$

Setting $5'.i = 5'.2$ $i = 3, 4, \ldots, N$, we can eliminate Z_0 and get

$$\frac{-2\{[B_i/(aA_i) - B_1/(aA_1)]\sec(t_1) - [C_i/(bA_i) - C_1/(bA_1)]\tan(t_1)\}}{[B_i/(aA_i) - B_1/(aA_1)]^2 - [C_i/(bA_i) - C_1/(bA_1)]^2}$$
$$= \frac{-2\{[B_2/(aA_2) - B_1/(aA_1)]\sec(t_1) - [C_2/(bA_2) - C_1/(bA_1)]\sin(t_1)\}}{[B_2/(aA_2) - B_1/(aA_1)]^2 - [C_2/(bA_2) - C_1/(bA_1)]^2}. \quad (6')$$

From (6′) we regroup the terms of $\sec(t_i)$, $\tan(t_i)$ together.
Let

$$P_i = \frac{B_i/(aA_i) - B_1/(aA_1)}{[B_i/(aA_i) - B_1/(aA_1)]^2 - [C_i/(bA_i) - C_1/(bA_1)]^2} - \frac{B_2/(aA_2) - B_1/(aA_1)}{[B_2/(aA_2) - B_1/(aA_1)]^2 - [C_2/(bA_2) - C_1/(bA_1)]^2}.$$

Let

$$Q_i = \frac{C_i/(aA_i) - C_1/(aA_1)}{[B_i/(aA_i) - B_1/(aA_1)]^2 - [C_i/(bA_i) - C_1/(bA_1)]^2} - \frac{C_2/(aA_2) - C_1/(aA_1)}{[B_2/(aA_2) - B_1/(aA_1)]^2 - [C_2/(bA_2) - C_1/(bA_1)]^2}.$$

Equation (6′) is equivalent to $\sec(t_1)P_i = \tan(t_1)Q_i$. We have $\sin(t_1) = P_i/Q_i$ and $\cos(t_1)^2 = 1 - P_i^2/Q_i^2$, $i = 3, 4, 5 \ldots$ Setting $6'.i = 6'.3$, we obtain

$$P_i/Q_i = P_3/Q_3. \quad (7')$$

Equation (7′) has only three unknowns, θ, ϕ, ψ.

We can now apply an optimization technique to solve the angle parameters. After they are solved, by equation (6′), we can solve for t_1.

$$\cos(t_1) = \pm(1 - P_i^2/Q_i^2)^{0.5}.$$

$$\sin(t_1) = P_i/Q_i.$$

Substituting these values for sec (t_1) and tan (t_1) into (5), we obtain

$$Z_0 = \frac{-2[(B_i/(aA_i) - B_1/(aA_1))\sec(t_1) - (C_i/(bA_i) - C_1/(bA_1))\tan(t_1)]}{[B_i/(aA_i) - B_1/(aA_1)]^2 - [C_i/(bA_i) - C_1/(bA_1)]^2}.$$

Since $\lambda_1 = Z_0/A_1$, we have

$$X_0 = -a\sec(t_1) + \lambda_1 B_1,$$

$$Y_0 = -b\tan(t_1) + \lambda_1 C_1.$$

2.3. *The parabola*

For every point (X, Y, Z) on the parabola there exists a parameter t_i such that X, Y, Z can be expressed in terms of it. Equation (1) can then be rewritten as

$$\begin{vmatrix} X_0 + t_i \\ Y_0 + bt_i^2 \\ Z_0 \end{vmatrix} = \lambda_i \begin{vmatrix} x_i' \cos\theta - f\sin\theta\cos\phi + z_i'\sin\theta\sin\phi \\ x_i'\sin\theta + f\cos\theta\cos\phi - z_i'\cos\theta\sin\phi \\ f\sin\phi + z_i'\cos\phi \end{vmatrix} \quad (2'')$$

for some λ_i and t_i $(i = 1, \ldots, N)$, where

$$\begin{vmatrix} x_i' \\ z_i' \end{vmatrix} = \begin{vmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{vmatrix} \begin{vmatrix} x_i \\ z_i \end{vmatrix}.$$

From the third component of equation (2") we have $\lambda_i = Z_0/A_i$. Substituting λ_i into the first and second components of (2) we have

$$B_j Z_0/A_j = X_0 + t_j, \quad (3'')$$

$$C_j Z_0/A_j = Y_0 + bt_j^2, \quad j = 1, \ldots, N.$$

Subtracting equation (3") with $j = 1$ from equation (3") with $j = i$ we get

$$B_i Z_0 A_1 - B_1 Z_0 A_i = A_i A_1 t_i - A_i A_1 t_1$$

$$C_i Z_0 A_1 - C_1 Z_0 A_i = A_i A_1 bt_i^2 - A_i A_1 bt_1. \quad (4'')$$

Dividing both sides of (4") by $A_i A_1$, leaving terms with t_i on the right hand side, squaring both sides of the first component of (4") and multiplying by b and then subtracting it from the second component, we obtain

$$A_i A_1(C_i A_1 - C_1 A_i) - Z_0(B_i A_1 - B_1 A_i)^2 b$$
$$- 2(B_i A_1 - B_1 A_i) A_i A_1 t_1 b = 0. \quad (5'')$$

$$Z_0 = \frac{-2b\{[B_i A_1 - B_1 A_1]A_i A_1\}t_1 + A_i A_1(C_i A_1 - C_1 A_i)}{[B_i A_1 - B_1 A_i]^2 b}$$

$$= \frac{-2[B_i/A_i - B_1/A_1]t_1 + (C_i/A_i - C_1/A_1)}{[B_i/A_i - B_1/A_1]^2 b}$$

For simplicity, we introduce new variables such that

$$Z_0 = (E_i t_1 + F_i)/G_i.$$

Setting $5''.i = 5''.2$ $i = 3, 4, \ldots, N$, we can eliminate Z_0 and get

$$[E_i t_1 + F_i]/G_i = [E_2 t_1 + F_2]/G_2. \quad (6'')$$

Rearranging we get

$$t_1 = (F_2/G_2 - F_i/G_i)/(E_i/G_i - E_2/G_2)$$

or

$$(F_2/G_2 - F_i G_i)/[E_i/G_i - E_2/G_2]$$
$$= (F_2/G_2 - F_3/G_3)/[E_3/G_3 - E_2/G_2]. \quad (7'')$$

Equation (7") has only three unknowns, θ, ϕ, ψ. We can now apply an optimization technique to solve the angle parameters. After they are solved, by equation (6"), we can solve t_1. Now we can solve Z_0 from (5") and X_0, Y_0 from (3") with $j = 1$.

3. OPTIMIZATION AND QUANTIZATION ERRORS

From Section 2 we are minimizing the sum

$$\sum \|P_i/Q_i - P_3/Q_3\|^2 \quad i = 4, \ldots, N.$$

N is the number of points of the input image coordinates. The optimization technique we use is the same as the one used in Haralick *et al.*[4] The routine we use is 'LMDIF' in MINPACK.[5]

The algorithm employed by this program is a version of the Levenberg–Marquardt algorithm. The Levenberg–Marquardt algorithm is a combination of the method of steepest descent (gradient search) and the classical Gauss–Newton method for nonlinear least squares problems. It is essentially steepest descent when the initial guess is far from the minimum point. Thus its global behavior is good and its ultimate convergence rate to the minimum point is also good. The method uses only first derivative information, yet typically, has second derivative convergence rate.

The disadvantage of this type of iteration method is that it usually requires the user to supply an initial guess. In our experiments, this problem is avoided by generating multiple random initial guesses. If it does not converge with one random guess we try and repeat the process until it converges.

The input image coordinates always contain the quantization errors and noise introduced in digitization. The point correspondences approach will degrade more rapidly than the optimization approach, because it depends heavily on the exact location of these image points. By using the optimization approach, this error is reduced to a minimum, because it always finds the best fit if no exact solution exists.

4. CURVES NOT IN STANDARD FORM

We can obtain the standard form of a conic after proper rotation. Let us assume the rotation angle is R. Suppose the parametric equations after rotation are

$$X' = X_0 + a\cos(t),\ Y' = Y_0 + b\sin(t).$$

We know $X' = X\cos(R) + Y\sin(R)$ and $Y' = -X\sin(R) + Y\cos(R)$. Substituting X', Y' in terms of X, Y and simplifying, the equations for X and Y become

$$X = (X_0 + a\cos(t))\cos(R) - (Y_0 + b\sin(t))\sin(R),$$

$$Y = (X_0 + a\cos(t))\sin(R) + (Y_0 + b\sin(t))\cos(R).$$

We also have $X\cos(R) + Y\sin(R) = X_0 + a\cos(t)$,

$-X\sin(R) + Y\cos(R) = Y_0 + b\sin(t)$.

From equation (2) we know that

$$\begin{vmatrix} X \\ Y \\ Z_0 \end{vmatrix} = \lambda_i \begin{vmatrix} x'_i\cos\theta - f\sin\theta\cos\phi + z'_i\sin\theta\sin\phi \\ x'_i\sin\theta + f\cos\theta\cos\phi - z'_i\cos\theta\sin\phi \\ f\sin\phi + z'_i\cos\phi \end{vmatrix} \quad (2)$$

for some λ_i and t_i ($i = 1, \ldots, N$), where

$$\begin{vmatrix} x'_i \\ z'_i \end{vmatrix} = \begin{vmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{vmatrix} \begin{vmatrix} x_i \\ z_i \end{vmatrix}$$

4.1. *The ellipse case*

Equation (3) of section (2) is then changed to

$B_iZ_0/A_i = X$,

$C_iZ_0/A_i = Y$.

$$(B_iZ_0/A_i)\cos(R) + (C_iZ_0/A_i)\sin(R) = X_0 + a\cos(t_i)$$
$$-(B_iZ_0/A_i)\sin(R) + (C_iZ_0/A_i)\cos(R) = Y_0 + b\sin(t_i). \quad (3)$$

We can also derive similar equations for (4), (5) and (6). We carry out the computation here.
Let

$$Z_0E_i = (B_iZ_0/A_i)\cos(R) - (B_1Z_0/A_1)\cos(R) + (C_iZ_0/A_i)\sin(R) - (C_1Z_0/A_1)\sin(R),$$

$$Z_0F_i = -(B_iZ_0/A_i)\sin(R) + (B_1Z_0/A_1)\sin(R) + (C_iZ_0/A_i)\cos(R) - (C_1Z_0/A_1)\cos(R).$$

We have

$$Z_0E_i = a\cos(t_i) - a\cos(t_1),$$
$$Z_0F_i = b\sin(t_i) - b\sin(t_1).$$

Eliminating t_i first we have

$$(Z_0E_i/a + \cos(t_1))^2 + (Z_0F_i/b + \sin(t_1))^2 = 1.$$

If $Z_0 \neq 0$,

$$Z_0 = -2(\cos(t_1)E_i/a + \sin(t_1)F_i/b)/(E_i^2/a^2 + F_i^2/b^2). \quad (5)$$

Setting $5.i = 5.2\ i = 2, \ldots, N$, where N is the number of image points, then

$$\left(\frac{E_i/a}{E_i^2/a^2 + F_i^2/b^2} - \frac{E_2/a}{E_2^2/a^2 + F_2^2/b^2}\right)\cos(t_1) = \left(\frac{-F_i/b}{E_i^2/a^2 + F_i^2/b^2} + \frac{F_2/b}{E_2^2/a^2 + F_2^2/b^2}\right)\sin(t_1).$$

Introducing new variables U_i, V_i, such that

$$U_i\cos(t_1) = V_i\sin(t_1),$$

and squaring both sides, we get

$$\cos(t_1)^2 = V_i^2/(U_i^2 + V_i^2). \quad (6)$$

Setting $6.i = 6.3$ we get

$$V_i^2/(U_i^2 + V_i^2) = V_3^2/(U_3^2 + V_3^2).$$

Again, this is an equation which only involves three unknown angle parameters if a, b and R are known.

4.2. *The hyperbola case*

For the hyperbola, the computation is similar. Equation (3′) in Section 2 is changed to

$B_iZ_0/A_i = X$,

$C_iZ_0/A_i = Y$.

$(B_iZ_0/A_i)\cos(R)$

$(C_iZ_0/A_i)\sin(R) = X_0 + a\sec(t_i)$,

$-(B_iZ_0/A_i)\sin(R)$

$(C_iZ_0/A_i)\cos(R) = Y_0 + b\tan(t_i)$. (3′)

Then, we can derive similar equations for (4), (5) and (6). We carry out the computation here.
Let

$$Z_0E_i = (B_iZ_0/A_i)\cos(R) - (B_1Z_0/A_1)\cos(R) + (C_iZ_0/A_i)\sin(R) - (C_1Z_0/A_1)\sin(R),$$

$$Z_0F_i = -(B_iZ_0/A_i)\sin(R) + (B_1Z_0/A_1)\sin(R) + (C_iZ_0/A_i)\cos(R) - (C_1Z_0/A_1)\cos(R).$$

We have

$$Z_0E_i = a\sec(t_i) - a\sec(t_1),$$
$$Z_0F_i = b\tan(t_i) - b\tan(t_1).$$

Eliminating t_i first we have

$$(Z_0E_i/a + \sec(t_1))^2 - (Z_0F_i/b + \tan(t_1))^2 = 1.$$

If $Z_0 \neq 0$,

$$Z_0 = -2(\sec(t_1)E_i/a - \tan(t_1)F_i/b)/(E_i^2/a^2 + F_i^2/b^2). \quad (5')$$

Setting $(5'.i) = (5'.2)\ i = 2, \ldots, N$, we have

$$\left(\frac{E_i/a}{E_i^2/a^2 + F_i^2/b^2} - \frac{E_2/a}{E_2^2/a^2 + F_2^2/b^2}\right)\sec(t_1)$$

$$= \left(\frac{F_i/b}{E_i^2/a^2 + F_i^2/b^2} - \frac{F_2/b}{E_2^2/a^2 + F_2^2/b^2}\right)\tan(t_1)$$

Introducing new variables U_i, V_i such that $U_i \sec(t_1) = V_i \tan(t_1)$, we have

$$\sin(t_1) = U_i/V_i. \tag{6'}$$

Setting $6'.i = 6'.3$ we get

$$U_i/V_i = U_3/V_3.$$

Again, this is an equation which only involves three unknown angle parameters if a, b and R are known.

4.3. *The parabola case*

For the parabola, equation (3) in Section 2 is changed to

$$B_iZ_0/A_i = X,$$

$$C_iZ_0/A_i = Y.$$

$$(B_iZ_0/A_i)\cos(R) + (C_iZ_0/A_i)\sin(R) = X_0 + t_i$$

$$-(B_iZ_0/A_i)\sin(R) + (C_iZ_0/A_i)\cos(R) = Y_0 + bt_i^2.$$

Then, we can derive similar equation for (4), (5) and (6). We carry out the computation here.
Let

$$Z_0E_i = (B_iZ_0/A_i)\cos(R) - (B_1Z_0/A_1)\cos(R) + (C_iZ_0/A_i)\sin(R) - (C_1Z_0/A_1)\sin(R),$$

$$Z_0F_i = -(B_iZ_0/A_i)\sin(R) + (B_1Z_0/A_1)\sin(R) + (C_iZ_0/A_i)\cos(R) - (C_1Z_0/A_1)\cos(R).$$

We have

$$Z_0E_i = t_i - t_1,$$

$$Z_0F_i = bt_i^2 - bt_1^2.$$

Eliminating t_i first we have

$$b(Z_0E_i + t_1)^2 - (Z_0F_i + bt_1^2) = 0.$$

or

$$Z_0 = (F_i - 2bE_it_1)/(bE_i^2). \tag{4''}$$

Setting $4''.i = 4''.2$ and rearranging it we have

$$t_1 = \frac{(F_i/(bE_i^2) - F_2/(bE_2^2))}{2bE_i/(bE_i^2) - 2bE_2/(bE_2^2)}. \tag{5''}$$

Now setting $5''.i = 5''.3$ we get

$$\frac{F_i/(bE_i^2) - F_2/(bE_2^2)}{2bE_i/(bE_i^2) - 2bE_2/(bE_2^2)} = \frac{F_3/(bE_3^2) - F_2/(bE_2^2)}{2bE_3/(bE_3^2) - 2bE_2/(bE_2^2)}.$$

Again, this is an equation involving only three unknowns.

5. STRAIGHT LINE PROJECTION

Finally, we solve the same problem for the projection of two or more lines. If there is only one line on the image, the constraints we derived are not sufficient to determine a unique solution. We will use the same techniques as in Section (2) to derive the constraints for the three angle parameters first. Based on these constraints, the parameters can be solved via minimization, then the position parameters are solved algebraicly.

First consider one line lying in the unknown plane. Let $M_i = (B_i, C_i, A_i)^T$ be the same kind of vector we defined in (1) of Section 2. For each image point (x_i, z_i) on the same line we have $M_i\lambda_i = (X_0 + at_i, Y_0 + bt_i, Z_0 + ct_i)^T$. Now, compute λ_i from the third component of each equation and substitute it into the first and second components. We have

$$\begin{aligned} B_i(Z_0 + ct_i)/A_i &= X_0 + at_i, \\ C_i(Z_0 + ct_i)/A_i &= Y_0 + bt_i. \end{aligned} \tag{1}$$

Subtracting 1.1 from 1.i we get

$$\begin{aligned} B_i/A_i(Z_0+ct_i) - B_1/A_1(Z_0+ct_1) &= at_i - at_1, \\ C_i/A_i(Z_0+ct_i) - C_1/A_1(Z_0+ct_1) &= bt_i - bt_1. \end{aligned} \tag{2}$$

Multiplying the second component by a/b and then subtracting it from the first component to eliminate t_i we have

$$(B_i/A_i - a/b\, C_i/A_i)(Z_0 + ct_i) - (B_1/A_1 - a/b\, C_1/A_1)(Z_0 + ct_1) = 0. \tag{3}$$

If $c = 0$ and $Z_0 \neq 0$ then we have

$$B_i/A_i - a/b\, C_i/A_i = B_1/A_1 - a/b\, C_1/A_1.$$

If $a = 0$ or $b = 0$ then $B_i/A_i = B_1/A_1$ or $C_i/A_i = C_1/A_1$. These three equations have only three unknowns, the angle parameter (θ, ϕ, ψ).

Applying the same techniques to the remaining lines lying in the plane, each line correspondence will introduce a new set of constraints on the unknowns. Putting all these constraints together we can then optimize on the three angle parameters (and slope b/a if it is unknown). After they are solved, solve for A_i, B_i, C_i from the matrix equation. Then solve for λ_i in terms of Z_0 and substitute it into the first and second components of the matrix equation (1). We will have $2N$ equations in $N + 3$ unknowns: N t_i's and X_0, Y_0, Z_0. (N is the number of points on these lines.)

The slope of the line with parametric form, as we described above, in the Z = constant plane is b/a. For the remaining lines, their parametric equations will have a different X_0, Y_0 from the first line. For example, the line which intersects line 1 at the other end points of line 1 has values $X_0 + L\cos(p)$ and $Y_0 + L\sin(p)$, where L is the length of the first segment and p is the angle between the first segment and the X-axis. Nevertheless, equation (2) is still true for these lines, as long as we use the right image point coordinates to

Table 1.

V	P	N
6	8	3528
5	8	1950
5	5	1500
4	8	980
4	7	880
4	6	800
4	5	720
4	4	640
3	8	406
3	3	216

compute A_i, B_i and C_i. It is easy to see $a = L\cos(p)$ and $b = L\sin(p)$.

A better way to solve the same problem without minimization is to compute t_i directly. Setting $c = 0$ in (1) we get

$$\begin{aligned} B_i Z_0/A_i &= X_0 + at_i, \\ C_i Z_0/A_i &= Y_0 + bt_i. \end{aligned} \tag{4}$$

Subtracting 4.1 from 4.i, $i = 2, \ldots, N$, where N is the number of points in the first line, we have

$$\begin{aligned} Z_0(B_i/A_i - B_1/A_1) &= a(t_i - t_1) \\ Z_0(C_i/A_i - C_1/A_1) &= b(t_i - t_1). \end{aligned}$$

Simplifying it we have

$$\begin{aligned} Z_0 &= a(t_i - t_1)/(B_i/A_i - B_1/A_1) \\ Z_0 &= b(t_i - t_1)/(C_i/A_i - C_1/A_1). \end{aligned} \tag{5}$$

Setting 5.2 = 5.i and solving for t_i we get

$$\begin{aligned} t_i - t_1 &= (t_2 - t_1)(B_i/A_i - B_1/A_1)/(B_2/A_2 - B_1/A_1) \\ t_i - t_1 &= (t_2 - t_1)(C_i/A_i - C_1/A_1)/(C_2/A_2 - C_1/A_1). \end{aligned} \tag{6}$$

If we treat $(t_2 - t_1)$ as a constant, then we can find the proper ratio between the remaining t_i's. Setting $t_{npts} = L$, we can then find the value of the other t_i's. Next, substituting t_i's into equation (5) we get Z_0. Substituting Z_0, t_i into equation (4) we get X_0, Y_0. This procedure works if we assume that the slope is known. We will only apply it to the first line, due to remarks stated in above paragraph.

6. DISCUSSION

If the plane where the curves lie is not the Z = constant plane, we have to do a rotation first. The rotation matrix rotates the given plane normal $\langle l, m, n\rangle$ into $\langle 0, 0, 1\rangle$. Then we can carry out the procedure we just described.

The techniques discussed in the above sections depend heavily on the parametric form of the curve. For example, for the ellipse we use the identity $\cos^2 + \sin^2 = 1$ to eliminate all the t_i's. For the hyperbola we have to use $\sec^2 - \tan^2 = 1$. So, there is no general formula for conics. Nevertheless, the ideas discussed here can be used for any curve in parametric form. The difference will be the identity to use to simplify the equation and the number of parameters to optimize.

The same techniques can be used to solve for non-planar curves, as long as the curve can be described in parametric form. The disadvantage is if we cannot eliminate parameter t_i from the first and second components, then we will have to perform optimization on them as well.

The computation time needed for the optimization MINPACK[5] routine 'LMDIF' is proportional to the fourth power of the number of unknowns. Assuming that V is the number of unknowns and P is the number of constraints, according to MINPACK the cpu timing is approximately $N = (V + 1)(PV^2 + V^3)$ multiplied by the cpu time which is needed to update the optimization function in each iteration. Table 1 is a comparison on the number of operations to evaluate the optimization function with different parameters.

7. CONCLUSION

In this paper we have described a new method for solving camera parameters when observing planar conic or polygon arcs. Instead of solving all six parameters at one time, we decomposed the problem into two parts: finding the look angles and then the position. By working in the camera-oriented system first, this problem is easily decomposed into two parts, as we have described. The first part is a three-dimensional optimization problem and the second part is an algebraic computation of point coordinates.

REFERENCES

1. M. A. Fischler and R. C. Bolles, Random sample consensus, *Proc. DAPRA Image Understanding Workshop*, Arlington, VA (1980).
2. L. T. Watson and L. G. Shapiro, Identification of space curves from two dimensional perspective views *IEEE Pattern Anal. Mach. Intell.* **PAMI-4**, 469–475 (1982).
3. R. M. Haralick, Using perspective transformations in scene analysis, *Comput. Graphics Image Process.* **13**, 191–221 (1980).
4. R. M. Haralick, Y. H. Chu, L. T. Watson and L. G. Shapiro, Identification of wire frame objects from their two dimensional perspective projection, *Proc. IEEE Conf. on Pattern Recognition and Image Processing, 1982*, pp. 572–579.
5. J. J. More, MINPACK Documentation, Argone National Laboratory, Argone, IL (1979).

About the Author—ROBERT M. HARALICK was born in Brooklyn, New York, on 30 September 1943. He received a B.S. degree from the University of Kansas in 1966. He has worked with Autonetics and IBM. In

1965 he worked for the Center for Research, University of Kansas, as a Research Engineer and in 1969, when he completed his Ph.D. at the University of Kansas, he joined the faculty of the Electrical Engineering Department, where he served as a Professor from 1975 to 1978. In 1979 Dr. Haralick joined the faculty of the Electrical Engineering Department at Virginia Polytechnic Institute and State University, where he was a Professor and Director of the Spatial Data Analysis Laboratory. In 1984 Dr. Haralick joined Machine Vision International, where he is Vice President of Research. Dr. Haralick has done research in pattern recognition, multi-image processing, remote sensing, texture analysis, data compression, clustering, artificial intelligence and general systems theory. He is responsible for the development of GIPSY (General Image Processing System), a multi-image processing package which runs on a minicomputer system.

He is a member of the Institute of Electrical and Electronic Engineers, the Association for Computer Machinery, Sigma Xi, the Pattern Recognition Society and the Society for General Systems Research.

About the Author—Y. H. CHU received a B.S. degree in Mathematics from the National Central University in Chungli, Taiwan, R.O.C., an M.S. in Mathematics from Virginia Tech., Blacksburg, VA, and a Ph.D. program in Computer Science, also from Virginia Tech., in 1983.

He is currently with the Department of Computer Science of Old Dominion University, Norfolk, VA. His research interests include computer vision, expert system construction, numerical analysis, natural and programming languages and image processing.