A Camera Calibration Using 4 Point–Targets

Sei-ichiro Kamata[†], Richard O. Eason[††], Masafumi Tsuji[†] and Eiji Kawaguchi[†]
[†]Dept. of Computer Engr., Kyushu Institute of Technology,
Tobata, Kitakyushu, Japan 804
[††] Dept. of Electrical Engr., University of Maine, Orono, Maine, USA 04469

Abstract

A method for determining the position of a camera using 4 point-targets is studied. We use 3 rotation angles and translation vector to describe the position of the camera for a pinhole model. For solving the 6 unknown parameters, a minimum of six point-targets is required to uniquely define the matrix (rotation and translation). However, we show that by using the properties of the matrix we can reduce this number to four. In the experiment we discuss the error properties of this method using real image data.

1 Introduction

A camera calibration technique plays an important role in the field of computer vision. There are a lot of techniques to this problem for several decades [5, 6]. These are classified into two categories: an itelative least squares fitting and a closed form solution. In early studies [1, 4], the problem of camera modeling and calibration was raised by photogrammetrists. The former method was mainly developing in the field of photogrammetric engineering. Recently several techniques using the later method have developed to be accurate, fast and efficient calibration [2, 3, 5]. Most of them use more than 10 points to compute the accurate calibration. In general, the closed form solution is simpler and faster than the least squares fitting. We concentrate on the closed form solution for the camera calibration to be fast, accurate and less point-targets.

3 rotation angles and translation vector are used to describe the position of the camera for a pinhole model. For solving the 6 unknown parameters, a minimum of six point-targets is required to uniquely define the matrix (rotation and translation). However, we show that by using the properties of the matrix we can reduce this number to *four*. A method for determining the position of a camera using 4 point–targets is studied. The solution is obtained by direct computation. This method has been investigated, implemented, and tested using a real set-up. In the experiment we discuss the error properties of this method using the real image data.

2 A camera calibration using 4 point–targets

In this section we present a method that determines the 3-D position of a camera using 4 point–targets.

2.1 The camera model

Let us recover the 3-dimensional position of each of the point-targets, P_0, P_1, P_2, and P_3 knowing their respective images p_0, p_1, p_2, and p_3 and their relative position with respect to each other ($P_i = (X_i, Y_i, Z_i)$ and $p_i = (x_i, y_i)$). In this discussion we use the pinhole camera model for our vision sensor. In our implementation, however, the sensor-to-image transformation, which is affine under the pinhole assumption, is approximated by a cubic polynomial to account for lens aberrations. The image coordinates (x, y) and the world coordinates (X, Y, Z) are related as follows:

$$x = X^c \frac{\lambda}{-Z^c}, \quad y = Y^c \frac{\lambda}{-Z^c}, \quad W^c = AW,$$

where $W = (X, Y, Z, 1)^T$, $W^c = (X^c, Y^c, Z^c, 1)^T$,

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & a_{10} \\ a_4 & a_5 & a_6 & a_{11} \\ a_7 & a_8 & a_9 & a_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The intermediate parameters X^c, Y^c, and Z^c represent the position in the camera coordinate system; the elements a_1 through a_{12} represent the matrix transformation A from world coordinates to camera coordinates. The parameter λ represents the focal length of camera. By the very nature of the problem, the parameters a_1 through a_9 characterize the effect of rotation only, whereas a_{10}, a_{11}, and a_{12} characterize the effect of translation. The objective is to compute the elements of the matrix transformation, a_1 through a_{12}, using the image of the target points as well as their 3-D relative position in order to determine the exact 3-dimensional position of those targets with respect to a coordinate system relative to the camera.

Each point provides a pair of linear equations involving the unknown parameters a_1 through a_{12}; therefore, using these equations alone, a minimum of six points is required to uniquely define the matrix A. However, we show that by using the properties of the matrix A we can reduce this number to *four*.

In this approach we introduce two intermediate coordinate systems, the "α-system" and the "β-system," to reduce the number of point-targets and write A as $A = SCT$ where T is a matrix representing the transformation between the world coordinate system and the α-system, C is a matrix representing the transformation between the α-system and the β-system, and S is a matrix representing the transformation between the β-system and the camera coordinate system. We choose the α-system so that the target points in this system are in what we call *standard position* whereby P_0 is at the origin, P_1 is on the positive X-axis, and P_2 is in the first or second quadrant of X-Y plane. We assume the four points are coplanar, so P_3 will also be in the X-Y plane. The β-system is chosen to be aligned with a fictitious camera which has its lens center (i.e., origin) at the same location as the real camera, but is oriented such that the image of P_0 is at the image origin and the image of P_1 is on the positive x-axis. We say this fictitious camera is in *ideal position*.

Both T and S are easily determined from the data for the problem, T from the world coordinates of the target points, and S from the image coordinates of their projections. The matrix C represents the solution to a somewhat simpler camera calibration problem in which target points in standard position are viewed by a camera in ideal position. This sets many of the coefficients to zero, and leads to a closed form solution to the problem. Once S, C, and T are found, A is determined by a simple matrix product. The steps for determining S, C, and T are outlined in the following.

2.2 Calculation of the matrix T

We find the matrix T, which carries the world coordinate system to the α coordinate system, by decomposing it into the following sequence of motions: translate so that P_0 is at the origin, rotate about the new Z-axis by an angle θ so that the Y-coordinate of P_1 is zero and the X-coordinate is positive, rotate about the new Y-axis by an angle β so that the Z-coordinate of P_1 is zero, and finally rotate about the new X-axis by an angle α so that the Z-coordinate of P_2 is zero and the Y-coordinate is positive. If X_0, Y_0, and Z_0 are the coordinates of P_0 in the world system, then we have the following expressions for the elements of T:

$$
\begin{aligned}
T_{11} &= cos\beta cos\theta; T_{12} = cos\beta sin\theta; T_{13} = -sin\beta \\
T_{14} &= -X_0 cos\beta cos\theta - Y_0 cos\beta sin\theta + Z_0 sin\beta \\
T_{21} &= -cos\alpha sin\theta + sin\alpha sin\beta cos\theta \\
T_{22} &= cos\alpha cos\theta + sin\alpha sin\beta sin\theta \\
T_{23} &= sin\alpha cos\beta \\
T_{24} &= -X_0(-cos\alpha sin\theta + sin\alpha sin\beta cos\theta) \\
&\quad -Y_0(cos\alpha cos\theta + sin\alpha sin\beta sin\theta) \\
&\quad -Z_0(sin\alpha cos\beta) \\
T_{31} &= sin\alpha sin\theta + cos\alpha sin\beta cos\theta \\
T_{32} &= -sin\alpha cos\theta + cos\alpha sin\beta sin\theta \\
T_{33} &= cos\alpha cos\beta \\
T_{34} &= -X_0(sin\alpha sin\theta + cos\alpha sin\beta cos\theta) \\
&\quad -Y_0(-sin\alpha cos\theta + cos\alpha sin\beta sin\theta) \\
&\quad -Z_0(cos\alpha cos\beta) \\
T_{41} &= T_{42} = T_{43} = 0; T_{44} = 1;
\end{aligned}
$$

According to the above requirements, we can determine θ by transforming P_1 using the transformation $R_\theta T_0$ and then setting the resulting Y component to zero. Solving the resulting equation for θ yields $\theta = tan^{-1}\left\{\frac{Y_1-Y_0}{X_1-X_0}\right\}$. If $X_1 < X_0$, then we add π to θ in order that P_1 now have a positive X component. Also, if $X_1 = X_0$ and $Y_1 = Y_0$, then we arbitrarily set $\theta = 0$. Similarly, we can determine β by transforming P_1 using the transformation $R_\beta R_\theta T_0$ and setting the resulting Z component to zero. Solving the resulting equation for β yields $\beta = tan^{-1}\left\{\frac{-(Z_1-Z_0)}{(X_1-X_0)cos\theta+(Y_1-Y_0)sin\theta}\right\}$. α is now determined by transforming P_2 using the transformation $R_\alpha R_\beta R_\theta T_0$ and setting the resulting Z component to zero. Solving this equation for α yields $\alpha = tan^{-1}\left\{\frac{[(X_2-X_0)cos\theta+(Y_2-Y_0)sin\theta]sin\beta+(Z_2-Z_0)cos\beta}{-(X_2-X_0)sin\theta+(Y_2-Y_0)cos\theta}\right\}$. We add π to α if the denominator is less than zero, so P_2 will be in the first or second quadrant of the $X-Y$ plane.

2.3 Calculation of the matrix S

The method used in solving for S is similar to the one used in solving for T. Here, we can decompose the *inverse* of S (the transform from the camera system to the β-system) by using the following intermediate transformations: pan the camera system about its Y-axis by angle ϕ until the image of P_0 is on the image y-axis, tilt about the new X-axis by angle ω until the image of P_0 is at the image origin, and finally, rotate about new Z-axis by angle ρ until the image of P_1 is on the positive image x-axis. Because the origin (i.e., lens center) of the fictitious camera remains at the origin of the real camera, each of these rotations will have a computable affect on the image.

These transformations are described by the matrices $R_\phi R_\omega$ and R_ρ and we have $S^1 = R_\rho R_\omega R_\phi$. Because the inverse of a rotation matrix is equal to its transpose, we can write $S = R_\phi^T R_\omega^T R_\rho^T$ which when multiplied out gives

$$
S = \begin{bmatrix}
\begin{matrix} cos\phi cos\rho + \\ sin\phi sin\omega sin\rho \end{matrix} & \begin{matrix} -cos\phi sin\rho + \\ sin\phi sin\omega cos\rho \end{matrix} & sin\phi cos\omega & 0 \\
cos\omega sin\rho & cos\omega cos\rho & -sin\omega & 0 \\
\begin{matrix} -sin\phi cos\rho + \\ cos\phi sin\omega sin\rho \end{matrix} & \begin{matrix} sin\phi sin\rho + \\ cos\phi sin\omega cos\rho \end{matrix} & cos\phi cos\omega & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}.
$$

We can determine the required angles as we did in solving for T, but we must know how the image of a point will change following camera rotation about the lens center. For image point (x, y), camera focal length λ, and camera rotation described by the matrix R, we first transform the 3-D coordinates of the image point, $(x, y, -\lambda)^T$, using the transformation R, calling the result (X^r, Y^r, Z^r). Then by using the projective equation we find this 3-D point (as well as the originating target point) has its image at $(-\lambda X^r/Z^r, -\lambda Y^r/Z^r)$ as seen by this rotated camera.

By transforming the image coordinates of P_0 in this way using the transformation R_ϕ and setting the x-coordinate of the result to zero, we can solve the resulting equation for ϕ getting $\phi = tan^{-1}(-x_0/\lambda)$. We solve for ω by transforming the image coordinates of P_0 using the transformation $R_\omega R_\phi$ and setting the y-coordinate of the result to zero. Solving the resulting equation for ω we get $\omega = tan^{-1}(y_0 cos\phi/\lambda)$. Finally we transform the image coordinates of P_1 using the transformation $R_\rho R_\omega R_\phi$, set the y-coordinate of the result to zero, and solve the resulting equation for ρ getting $\rho = tan^{-1}\left\{\frac{y_1 cos\omega + x_1 sin\phi sin\omega - \lambda cos\phi sin\omega}{x_1 cos\phi + \lambda sin\phi}\right\}$.

2.4 Calculation of the matrix C

The matrix C represents the solution to a problem in which target points in standard position are viewed by a camera in ideal position. The coordinates of the target points in standard position, $(X_i^\alpha, Y_i^\alpha, Z_i^\alpha)^T$, are found by applying the transform T to the original target points. The image coordinates of these points as seen by a camera in ideal position, $(x_i^\beta, y_i^\beta)^T$, are found by applying the transform S^{-1} $(= S^T)$ to the 3-D points $(x_i, y_i, -\lambda)$, producing the points (X_i^r, Y_i^r, Z_i^r), and then setting $x_i^\beta = -\lambda X_i^r/Z_i^r$ and $y_i^\beta = -\lambda Y_i^r/Z_i^r$.

C is then a rotation matrix which is a solution to a problem described by

$$(X_i^\beta, Y_i^\beta, Z_i^\beta, 1)^T = C(X_i^\alpha, Y_i^\alpha, Z_i^\alpha, 1)^T,$$

$$(x_i^\beta, y_i^\beta)^T = (-\lambda\frac{X_i^\beta}{Z_i^\beta}, -\lambda\frac{Y_i^\beta}{Z_i^\beta})^T,$$

where $X_i^\alpha, Y_i^\alpha, Z_i^\alpha, x_i^\beta$, and y_i^β are known for $i = 0, 1, 2,$ and 3. Because of the way in which we defined the α and β systems, the parameters $X_0^\alpha, Y_0^\alpha, Z_0^\alpha, Y_1^\alpha, Z_1^\alpha, Z_2^\alpha, Z_3^\alpha, x_0^\beta, y_0^\beta$, and y_1^β are all equal to zero. Substituting $P_0 = (0,0,0,1)^T$ into the equations represented above immediately gives $C_{14} = C_{24} = 0$. Substituting in the other points gives the following

$$\lambda X_1^\alpha C_{11} + x_1^\beta X_1^\alpha C_{31} + x_1^\beta C_{34} = 0 \tag{1}$$

$$\lambda X_1^\alpha C_{21} = 0 \tag{2}$$

$$\lambda X_2^\alpha C_{11} + \lambda Y_2^\alpha C_{12} + x_2^\beta X_2^\alpha C_{31} + x_2^\beta Y_2^\alpha C_{32} + x_2^\beta C_{34} = 0 \tag{3}$$

$$\lambda X_2^\alpha C_{21} + \lambda Y_2^\alpha C_{22} + y_2^\beta X_2^\alpha C_{31} + y_2^\beta Y_2^\alpha C_{32} + y_2^\beta C_{34} = 0 \tag{4}$$

$$\lambda X_3^\alpha C_{11} + \lambda Y_3^\alpha C_{12} + x_3^\beta X_3^\alpha C_{31} + x_3^\beta Y_3^\alpha C_{32} + x_3^\beta C_{34} = 0 \tag{5}$$

$$\lambda X_3^\alpha C_{21} + \lambda Y_3^\alpha C_{22} + y_3^\beta X_3^\alpha C_{31} + y_3^\beta Y_3^\alpha C_{32} + y_3^\beta C_{34} = 0 \tag{6}$$

We immediately see from the second of these equations that C_{21} must be zero.

We can solve for the remaining elements using Gaussian elimination in the following way. We first take Y_3^α times Eq. 3 minus Y_2^α times Eq. 5 and Y_3^α times Eq. 4 minus Y_2^α times Eq. 6 getting

$$B_1 C_{11} + B_2 C_{31} + B_3 C_{32} + B_4 C_{34} = 0 \tag{7}$$

$$B_5 C_{31} + B_6 C_{32} + B_7 C_{34} = 0 \tag{8}$$

where $B_1 = \lambda(X_2^\alpha Y_3^\alpha - X_3^\alpha Y_2^\alpha)$, $B_2 = x_2^\beta X_2^\alpha Y_3^\alpha - x_3^\beta X_3^\alpha Y_2^\alpha$, $B_3 = (x_2^\beta - x_3^\beta) Y_2^\alpha Y_3^\alpha$, $B_4 = x_2^\beta Y_3^\alpha - x_3^\beta Y_2^\alpha$, $B_5 = y_2^\beta X_2^\alpha Y_3^\alpha - y_3^\beta X_3^\alpha Y_2^\alpha$, $B_6 = (y_2^\beta - y_3^\beta) Y_2^\alpha Y_3^\alpha$, $B_7 = y_2^\beta Y_3^\alpha - y_3^\beta Y_2^\alpha$. We next take B_6 times Eq. 7 minus B_3 times Eq. 8 getting

$$B_1 B_6 C_{11} + (B_2 B_6 - B_5 B_3) C_{31} + (B_4 B_6 - B_7 B_3) C_{34} = 0. \tag{9}$$

Finally we take $(B_4 B_6 - B_7 B_3)$ times Eq. 1 minus x_1^β times Eq. 9 and get

$$B_8 C_{11} + B_9 C_{31} = 0 \tag{10}$$

where B_8 and B_9 are given by

$$B_8 = \lambda X_1^\alpha (B_4 B_6 - B_7 B_3) - x_1^\beta B_1 B_6,$$

$$B_9 = x_1^\beta [X_1^\alpha (B_4 B_6 - B_7 B_3) - (B_2 B_6 - B_5 B_3)].$$

Because the rotation submatrix of C must be orthonormal and the term C_{21} was found to be zero, we also know that

$$C_{11}^2 + C_{31}^2 = 1. \tag{11}$$

Now, using Eq. 10 and Eq. 11 we find $C_{11} = \pm\sqrt{\frac{1}{1+(B_8/B_9)^2}}$ The sign of C_{11} can be determined as follows: we note that the point P_0 must be in front of the camera, so the transformation C must give it a positive Z_0^β component. As P_0 is at the origin of the α-system, this implies that C_{34} must be positive. Using this, along with Eq. 1 and the fact that x_1^β, X_1^α, and λ are all positive, we find that C_{11} is positive if and only if λ/x_1^β is less than $-C_{31}/C_{11}$, where Eq. 10 tells us $-C_{31}/C_{11} = B_8/B_9$.

Now that C_{11} is known, the other parameters of C can be found by substitution into the previous equations with the following results:

$$\begin{array}{lcl}
C_{31} & = & -B_8 C_{11}/B_9 \\
C_{34} & = & -X_1^\alpha(\lambda C_{11}/x_1^\beta + C_{31}) \\
C_{32} & = & \begin{cases} -(B_5 C_{31} + B_7 C_{34})/B_6 & \text{if } B_6 \neq 0 \\ -(B_1 C_{11} + B_2 C_{31} + B_4 C_{34})/B_3 & \text{otherwise} \end{cases} \\
C_{12} & = & -(\lambda X_2^\alpha C_{11} + x_2^\beta X_2^\alpha C_{31} + x_2^\beta Y_2^\alpha C_{32} + x_2^\beta C_{34})/(\lambda Y_2^\alpha) \\
C_{22} & = & -y_2^\beta(X_2^\alpha C_{31} + Y_2^\alpha C_{32} + C_{34})/(\lambda Y_2^\alpha)
\end{array}$$

Because the rotation submatrix of C is orthonormal, the elements of the third column can be found using cross products of the elements in the first two rows. i.e., $C_{13} = -C_{22}C_{31}$, $C_{23} = C_{12}C_{31} - C_{11}C_{32}$, $C_{33} = C_{11}C_{22}$, where product terms involving C_{21} are omitted because C_{21} is zero. This completes the determination of the unknown elements of C.

3 Experiments

We have implemented our method on Apollo workstation connected to a PIC–2350 super scanner(2000 × 3000 CCD camera). Here the optical axis passes through a point (983,1325) on the image. Using several sets of points as shown in Fig. 1 the camera position was computed by the matrix A. We have taken a 512 × 512 image with 8bits/pixel from the camera. The position of point-targets on the world coordinate and the image coordinate is shown in the following.

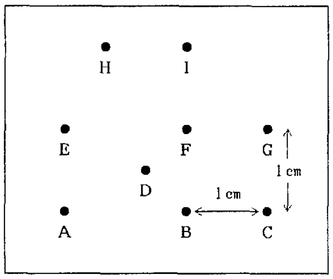

Figure 1: Calibration target

Table 1: The relation between the calculated image coordinates and the image coordinates

	Calculated points	Real points	Error
A	(0.00, 0.00)	(0.00, 0.00)	0.00000
B	(148,10, 0.10)	(149.00, 1.00)	0.00002
C	(250.91, 1.68)	(248.00, 0.50)	3.14404
D	(98.49,-48.75)	(98.50, -48.00)	0.74918
E	(0.00,-97.50)	(0.00, -97.50)	0.00000
F	(148.00,-98.00)	(148.00, -98.00)	0.00002
G	(249.21,-98.32)	(248.00, -99.00)	1.37847
H	(48.51,-194.37)	(51.00,-197.50)	4.00091
I	(147.01,-195.68)	(148.00,-197.50)	2.07023

$$
\begin{array}{llcll}
P_0 : & (X_0, Y_0, Z_0) = (25, 20, 0) & \rightarrow & p_0 : & (x_0, y_0) = (\ 0,\ 0.0) \\
P_1 : & (X_1, Y_1, Z_1) = (40, 20, 0) & \rightarrow & p_1 : & (x_1, y_1) = (149,\ 1.0) \\
P_2 : & (X_2, Y_2, Z_2) = (25, 30, 0) & \rightarrow & p_2 : & (x_2, y_2) = (\ 0, -97.5) \\
P_3 : & (X_3, X_3, Z_3) = (40, 30, 0) & \rightarrow & p_3 : & (x_3, y_3) = (148, -98.0)
\end{array}
$$

Using the 4 point-targets A,B,E,F as shown in Fig.1, we can obtain the following matrix A,

$$
A = \begin{bmatrix} -0.99993 & 0.00000 & 0.00982 & 24.99823 \\ -0.00671 & 1.00324 & 0.00646 & -19.89701 \\ -0.00983 & 0.00646 & -1.00317 & 9.82707 \\ -0.00000 & 0.00000 & 0.00000 & 1.00000 \end{bmatrix}.
$$

The results of the errors between the calculated image coordinates and the real image coordinates are shown in Table 1. Here we define the error by ERROR $= \sqrt{(x-\bar{x})^2 + (y-\bar{y})^2}$. From the results, we notice that the error between the calculated image coordinates and the real image coordinates is increasing in proportion to the distance from the 4 point-targets. We studied the relation between the error and the position of points on a X-axis or a Y-axis. The results are shown in Fig.2 and Fig.3. These figures show the relation between the error and the gap of optical axis established by the center of the lens. The larger the gap of optical axis is, the more the error increases. This means that the origin on the image has to correspond to the cross point of optical axis on the X-Y plane.

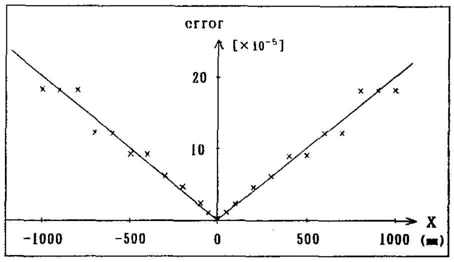

Figure 2: The relation between the error and the position of points along the X-axis

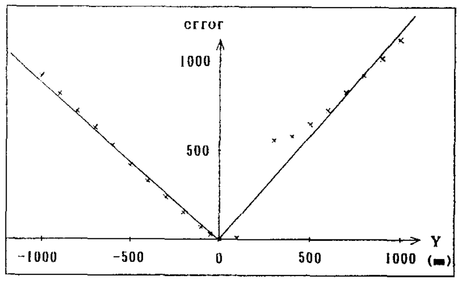

Figure 3: The relation between the error and the position of points along the Y-axis

In this experiment, the error is caused by (1) the gap of optical axis of the camera, (2) the gap of the focal point, (3) the gap on computing the center of point-targets on an image.

4 Conclusion

A method for determining the position of a camera using 4 point-targets has been studied, implemented, and tested using a real set-up. In the experiment we investiged the error properties of this method using the real image data.

References

[1] I.W.Faig, Calibration of close-range photogrammetric system : Mathmatical formulation, *Photogram. Engr. and Remote Sens.*, 41-12, 1479-1486, 1975.

[2] S.Ganapathy, Decomposition of transformation Matrices for Robot Vision, *Proc. of IEEE Int. Conf. on Robotics*, 130-139, 1984.

[3] M.Ito and A.Ishii, Range and shape measurement using three-view stereo analysis, *Proc. of CVPR*, 9-14, 1984.

[4] I.E.Sobel, On calibrating computer controlled cameras for perceiving 3-D scenes, *Artificial Intelligence*, 5, 185-198, 1974.

[5] R.Y.Tsai, A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE J. of RA*, RA-3, 4, 323-344, 1987.

[6] J.S.Yuan, A general photogrammic method for for determining object position and orientation, *IEEE J. of RA*, RA-5, 2, 129-142, 1989.