Camera Calibration by Vanishing Lines for 3-D Computer Vision

Ling-Ling Wang and Wen-Hsiang Tsai

Abstract—A new approach to camera calibration by vanishing lines is proposed. Calibrated parameters include the orientation, the position, and the focal length of a camera. A hexagon is employed as the calibration target to generate a vanishing line of the ground plane from its projected image. It is shown that the vanishing line includes useful geometric hints about the camera orientation parameters and the focal length, from which the orientation parameters can be solved easily and analytically. And the camera position parameters can be calibrated by the use of related geometric projective relationships. The simplicity of the target eliminates the complexity of the environment setup and simplifies the feature extraction in relevant image processing. The calibration formulas are also simple to compute. Experimental results show the feasibility of the proposed approach.

Index Terms—Camera calibration, computer vision, vanishing line, vanishing point.

I. INTRODUCTION

With the advent of 3-D computer vision, it is important to calibrate cameras for various computer vision applications. Camera calibration is the problem of determining the elements that govern the relationship between the 2-D image that a camera perceives and the 3-D information of the imaged object. It is identical to the problem of robot location in 3-D space. There are many existing techniques for solving this problem. Fukui [2] used a diamond shape target placed on the wall to determine the 2-D location of the camera with respect to the diamond. The camera lens center and the target center have to be set at the same height. In addition, the optical axis of the camera must pass through the center of the target. Courtney and Aggarwal [4] used the same target as Fukui’s but relaxed the restriction that the camera lens center must be as high as the diamond center. Instead, they made the assumption that the height of the camera is known. In [5], Magee and Aggarwal used a sphere with two perpendicular great circles as the target to determine the 3-D location of the camera relative to the sphere. Before the image of the sphere is taken, the camera optical axis must be pointed through the sphere center. Chou and Tsai [6] used house corners as calibration targets. Camera position and orientation parameters are computed provided that the height of the camera lens center is known.

Fischler and Bolles [7] found camera position and orientation parameters by first computing the heights of rays from the camera lens center to the control points in the image plane. His algorithm is non-linear and usually six point correspondences are required to get a unique solution. In [8], Tsai used 60 control points to derive camera position and orientation parameters as well as the focal length, radial lens distortion, and image scanning parameters. In certain other non-linear optimization approaches [9], [10], the computation is complicated and a good initial guess to start the non-linear search is required.

II. USING VANISHING LINES FOR CAMERA CALIBRATION

The problem of camera calibration is to compute the camera intrinsic and extrinsic parameters. The extrinsic parameters of a camera indicate the position and the orientation of the camera with respect to a world coordinate system, and the intrinsic parameters characterize the inherent properties of the camera optics, including the focal length, the image center, the image scaling factor, and the lens distortion coefficients.

For general computer vision applications, the intrinsic parameters of the camera remain the same except that the focal length may vary for different requirements of imaging distances. So, we may calibrate the intrinsic parameters in advance and calibrate only the extrinsic parameters and the focal length during application tasks. This may reduce the calibration complexity and enhance the efficiency. New methods for calibrating the extrinsic parameters and the focal length of a camera using vanishing lines are described in this section.

A. The Calibration Model

The target proposed for use to calibrate the camera orientation and position parameters, and the focal length is a flat hexagon shape put on the ground with three pairs of parallel opposite sides as illustrated in Fig. 1. Let \( P_i \) through \( P_6 \) be the six vertices of the hexagon. Two right-handed coordinate systems are defined in this study. One is the world coordinate system and the other is the camera coordinate system. The origin of the world coordinate system is located at vertex \( P_1 \) of the hexagon with the positive Y axis being parallel to edges \( \overrightarrow{P_5 P_2} \) and \( \overrightarrow{P_4 P_3} \), and the positive Z axis being vertical to the ground and pointing upwards. For clarity, the Z axis is not shown in Fig. 1.
Fig. 1. The calibration target used in the proposed method.

The world coordinates of the six vertices are known in advance by manual measurement. Shown in Fig. 2 is the camera coordinate system with the lens center as the origin. The \( V \) axis is the optical axis of the camera and the \( U-W \) plane is parallel to the image plane located at \( V = f \) with \( f \) being the negation of the camera focal length. The image coordinates of any point in the image plane are specified as \((u, w)\) with respect to the camera coordinate system.

We now define the camera parameters with respect to the world coordinate system. Suppose that the camera lens center \( L \) is located at \((x_c, y_c, z_c)\), and the pan, tilt, and swing angles of the camera are \( \theta, \phi, \psi \), respectively. Based on these parameters, two matrices used in the world-to-camera coordinate transformation [1] are defined in the following:

\[
T = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
x_c & y_c & z_c & 1
\end{bmatrix},
\]

\[
M = \begin{bmatrix}
\cos \theta \cos \psi + \sin \theta \sin \phi \sin \psi & -\sin \theta \cos \phi & \sin \theta \sin \phi \cos \psi - \cos \theta \sin \psi & 0 \\
\sin \theta \cos \psi - \cos \theta \sin \phi \sin \psi & \cos \phi \cos \theta & \cos \theta \sin \phi \cos \psi - \sin \theta \sin \psi & 0 \\
\sin \phi \cos \psi & 0 & \cos \phi \cos \psi & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\]

For brevity of representation, matrix \( M \) is denoted as

\[
M = \begin{bmatrix}
A & D & G & 0 \\
B & E & H & 0 \\
C & F & I & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

The coordinate transformation between the two coordinate systems can be written as

\[
(u, v, w, 1) = (x, y, z, 1) \cdot T^{-1} \cdot M.
\]

B. Calibrating the Camera Orientation Parameters and the Focal Length

It is well known that parallel lines in the 3-D world which are not parallel to the image plane converge to one point in the image. This point is called a vanishing point [12]. The vanishing points of all the parallel lines on the ground will align to form a line, called the vanishing line of the ground plane [13], [14]. This vanishing line in the image is just the projection of the horizon or the skyline. The vanishing line in the image provides a clue for camera calibration. In the following, the orientation, the position, and the scaling information of the vanishing line will be used to derive analytic solutions for the camera orientation parameters and the focal length.

1) Relation Between the Vanishing Line and the Camera Swing Angle: When we see a sea or an outdoor world, the slope of the sea horizon or the skyline varies with different swing angles of our heads. This fact indicates a hint that the swing angle \( \psi \) of the camera can be obtained from the orientation of the vanishing line of the ground plane in the image. And this is indeed a truth found in this study. More specifically, we will prove next that the vanishing line of the ground plane can be described by the following equation:

\[
u \cdot \sin \psi + w \cdot \cos \psi = -f \cdot \tan \phi,
\]

and so the slope of the vanishing line is just \(-\tan \psi\) which is the tangent of the swing angle. The above relation is illustrated in Fig. 3.

The perspective projection of any set of parallel lines which are not parallel to the image plane converges to a vanishing point. In 3-D space the parallel lines can be imagined to meet at a point at infinity, and the projection of this point at infinity is just the vanishing point of any set of parallel lines. So to any line \( L \), there corresponds a vanishing point which is also the vanishing point of any set of parallel lines which are parallel to \( L \). Let \( L_1 \) and \( L_2 \) be two nonparallel lines on the ground plane:

\[
L_1 : a_1 x + b_1 y = c_1, \quad z = 0;
\]

\[
L_2 : a_2 x + b_2 y = c_2, \quad z = 0.
\]

Also let \( V_1 = (u_1, w_1) \) and \( V_2 = (u_2, w_2) \) be the image coordinates of the two vanishing points of \( L_1 \) and \( L_2 \), respectively. We can compute

Fig. 3. Relation between the vanishing line and the camera swing angle.
Angle to infinity. From the coordinate transformation described in (2) and imaging geometry, the projection of \( P \) in the image, i.e., the vanishing point \( V_1 \), is located at the following image coordinates:

\[
(u_1, w_1) = \lim_{y_p \to \infty} \left( \frac{f \cdot [A(x_p - x_c) + B(y_p - y_c) - C z_c]}{D(x_p - x_c) + E(y_p - y_c) - F z_c}, \frac{f \cdot [G(x_p - x_c) + H(y_p - y_c) - I z_c]}{D(x_p - x_c) + E(y_p - y_c) - F z_c} \right)
\]

Because \( a_1 x_p + b_1 y_p = c_1 \), we can substitute \( x_p = (c_1 - b_1 y_p)/a_1 \) into the above equation, simplify the resulting equation, and compute the limit to obtain

\[
(u_1, w_1) = \left( \frac{f \cdot (-b_1 A + a_1 B)}{-b_1 D + a_1 E}, \frac{f \cdot (-b_1 G + a_1 H)}{-b_1 D + a_1 E} \right)
\]

Similarly, \( V_2 = (u_2, w_2) \) can be derived to be

\[
(u_2, w_2) = \left( \frac{f \cdot (-b_2 A + a_2 B)}{-b_2 D + a_2 E}, \frac{f \cdot (-b_2 G + a_2 H)}{-b_2 D + a_2 E} \right)
\]

In addition, the line passing through \( V_1 \) and \( V_2 \) is just the vanishing line of the ground plane. Using (4) and (5) and resuming the terms of \( A \) through \( H \) described in (1), we can derive the vanishing line equation as (3). Note that it is independent of \( V_1 \) and \( V_2 \), and also of \( L_1 \) and \( L_2 \). It depends only on the tilt angle \( \phi \), the swing angle \( \psi \), and the focal length \( f \) of the camera.

2) Relation Between the Vanishing Line and the Camera Pan Angle: Let \((u_1, w_1)\) be the image coordinates of the vanishing point \( V_1 \) of two parallel lines with slope \( m_1 \) on the ground. From (4), we have

\[
(u_1, w_1) = \left( \frac{A + m_1 B}{D + m_1 E}, \frac{G + m_1 H}{D + m_1 E} \right)
\]

Similarly, if \((u_2, w_2)\) are the coordinates of the vanishing point \( V_2 \) of two other parallel lines with slope \( m_2 \) on the ground, then

\[
(u_2, w_2) = \left( \frac{A + m_2 B}{D + m_2 E}, \frac{G + m_2 H}{D + m_2 E} \right)
\]

Let \((u_0, w_0)\) be the image coordinates of the depth vanishing point \( V_0 \) [1] (i.e., the vanishing point of two parallel lines parallel to the \( Y \) axis of the world coordinate system), then by considering the slope value in (6) or (7) to be infinity, it is easy to figure out that

\[
(u_0, w_0) = \left( \frac{B \cdot f}{H \cdot f}, \frac{H \cdot f}{E \cdot f} \right)
\]

From (6), (7), and (8) and the definition of the terms of \( A \) through \( H \) of matrix \( M \) in (1) we can derive the following equation:

\[
\frac{|V_1 V_2|^2}{|V_0 V_2|^2} = \left( \frac{(u_1 - u_2)^2 + (w_1 - w_2)^2}{(u_0 - u_2)^2 + (w_0 - w_2)^2} \right)^2 = \left( \frac{(\tan \theta - m_2)^2}{(\tan \theta - m_1)^2} \right)^2
\]

where \(|V_1 V_2|^2\) means the length of line segment \( V_1 V_2 \) with \( i = 1 \) and 2. As illustrated in Fig. 4, the known ratio value \( r \) of \(|V_1 V_2|^2\) to

\[
\begin{align*}
\text{Fig. 4. Relation between the vanishing line and the camera pan angle.} \\
\end{align*}
\]

\[
\text{Fig. 5. Relation between the vanishing line and the camera tilt angle.} \\
\]

\[|V_2 V_0|\] (available from the image) determines the tilt angle \( \phi \) of the camera.

Which of \( \theta_1 \) and \( \theta_2 \) is correct is determined after the tilt angle \( \phi \) is computed, as can be seen later.

It is intuitively apparent that if the camera is right, unbiased (i.e., \( \theta = 0, \phi = 0 \), and \( \psi = 0 \)), and \( m_1 = \frac{y}{x} \), \( |V_1 V_2| \) will be equal to \( |V_2 V_0| \). But when the camera has a nonzero pan angle, \( |V_1 V_2| \) and \( |V_2 V_0| \) will not be identical. So we can say that the pan angle \( \theta \) determines the scaling characteristic of the vanishing line.

3) Relation Between the Vanishing Line and the Camera Tilt Angle: From (3), the vanishing line equation, we can compute the \( W \) axis intercept \( w_0 \) of the vanishing line to be \( w_0 = -f \cdot \tan \phi \cdot \cos \psi \cdot \sin \psi \), as shown in Fig. 5. It means that the tilt angle \( \phi \), the swing angle \( \psi \), and the focal length \( f \) determine the position of the vanishing line. Therefore, if the focal length \( f \) is known, then we can get the tilt angle \( \phi \) from the position of the vanishing line in the image, or equivalently, from \( \tan \phi = -w_0 / \cos \psi \). But if \( f \) is unknown, we can use the additional information of two parallel line pairs to compute both the tilt angle and the focal length simultaneously, as described below.

From (6) we can get

\[
\frac{u_1}{u_0} = \frac{A + m_1 B}{G + m_1 H}.
\]

From matrix \( M \) defined in (1), the preceding equation can be reduced to

\[
\sin \phi = \frac{\cos \theta \sin \psi + m_1 \sin \theta \sin \psi}{\sin \theta \cos \psi - m_1 \sin \theta \cos \psi} w_1 + \frac{\cos \theta \cos \psi + m_1 \sin \theta \cos \psi}{\sin \theta \cos \psi - m_1 \sin \theta \cos \psi} w_2.
\]

Similarly, from (7) we have

\[
\sin \phi = \frac{\cos \theta \sin \psi + m_2 \sin \theta \sin \psi}{\sin \theta \cos \psi - m_2 \sin \theta \cos \psi} w_1 + \frac{\cos \theta \cos \psi + m_2 \sin \theta \cos \psi}{\sin \theta \cos \psi - m_2 \sin \theta \cos \psi} w_2.
\]
The value of the vanishing line, we can get the focal length in the calibration process. Furthermore, using the $W$ axis intercept value $w_0$ of the vanishing line, we can get the focal length $f$ as $f = -w_0 \cdot \cos \psi / \tan \phi$.

\[
\cos \alpha = \frac{\overline{LP}_1 \cdot \overline{LP}_2}{|\overline{LP}_1| \cdot |\overline{LP}_2|} = \frac{(x_1 - x_0)(x_2 - x_0) + (y_2 - y_0)(y_2 - y_0) + (z_2 - z_0)(z_2 - z_0)}{\sqrt{[(x_1 - x_0)^2 + (y_2 - y_0)^2 + (z_2 - z_0)^2]^2}}.
\]

From (9) or (10), the tilt angle $\phi$ can be obtained, but averaging can be used to improve the accuracy when real images are used in the calibration process. Furthermore, using the $W$ axis intercept value $w_0$ of the vanishing line, we can get the focal length $f$ as $f = -w_0 \cdot \cos \psi / \tan \phi$.

Now we have obtained the camera swing angle $\psi$, tilt angle $\phi$, and two pan angles $\theta_1$ and $\theta_2$. To determine which of $\theta_1$ and $\theta_2$ is correct, we substitute, respectively $(\theta_1, \phi, \psi)$ and $(\theta_2, \phi, \psi)$ into (6) and choose the one which satisfies (6) to be the desired pan angle $\theta$.

### C. Calibrating the Camera Position Parameters

Suppose that a known ground point $P_1$ is located at $(x_1, y_1, z_1)$ in the world coordinate system and that its projected point in the image is $P_1' = (u_1, v_1)$. By (2), the world coordinates $(x_1', y_1', z_1')$ of image point $P_1'$ can be computed as

\[
\begin{align*}
(x_1', y_1', z_1') &= (u_1, f, w_1, 1) \cdot M^{-1} \cdot T \\
&= (u_1 A + f D + w_1 G + x_1, u_1 B + f E + w_1 H + y_1, u_1 C + f F + w_1 I + z_1).
\end{align*}
\]

The line $P_1P_2$ passing through the lens center $L$, in the world coordinate system can be represented as

\[
x - x_1 = \frac{y - y_1}{y_1' - y_1} = \frac{z - z_1}{z_1' - z_1}
\]
or

\[
x - x_1 = \frac{y - y_1}{u_1 A + f D + w_1 G} = \frac{y - y_1}{u_1 B + f E + w_1 H} = \frac{z - z_1}{u_1 C + f F + w_1 I}.
\]

From Fig. 6, the intersection of line $P_1P_2$ and the plane $x = z_1 + h$, where $h$ is the camera height, is just the lens center $L$. So the world coordinates $(x_2, y_2, z_2)$ of $L$ can be derived by solving the simultaneous equations of (12) and $z = z_1 + h$. And the solutions are

\[
\begin{align*}
x_2 &= x_1 + h(u_1 A + f D + w_1 G)/(u_1 C + f F + w_1 I), \\
y_2 &= y_1 + h(u_1 B + f E + w_1 H)/(u_1 C + f F + w_1 I), \\
z_2 &= z_1 + h.
\end{align*}
\]

Once $h$ is determined, so is the position $(x_2, y_2, z_2)$ of the lens center. The following is one way to determine the value of $h$.

Let $P_2$ be another known ground point with world coordinates $(x_2, y_2, z_2)$ and its projected point be $P_2'$ with image coordinates $(u_2, v_2)$. Suppose that the angle between $LP_1$ and $LP_2$ is $\alpha$. By vector inner product, we have

\[
\overline{LP}_1 \cdot \overline{LP}_2 = |\overline{LP}_1| \cdot |\overline{LP}_2| \cdot \cos \alpha.
\]

The camera coordinates for $L$ are $(0, 0, 0)$, for $P_1'$ are $(u_1, f, w_1)$, and for $P_2'$ are $(u_2, f, w_2)$, so we can compute $\cos \alpha$ by

\[
\cos \alpha = \frac{\overline{LP}_1 \cdot \overline{LP}_2}{|\overline{LP}_1| \cdot |\overline{LP}_2|} = \frac{u_1 u_2 + f^2 + w_1 w_2}{\sqrt{(u_1^2 + f^2 + w_1^2)(u_2^2 + f^2 + w_2^2)}}.
\]

In addition, $\alpha$ is also the angle between $LP_1$ and $LP_2$, so we have

\[
\overline{LP}_1 \cdot \overline{LP}_2 = |\overline{LP}_1| \cdot |\overline{LP}_2| \cdot \cos \alpha.
\]

The world coordinates for $L$ are $(x_1, y_1, z_1)$, for $P_1$ are $(x_1, y_1, z_1)$, and for $P_2$ are $(x_2, y_2, z_2)$. The value of $\cos \alpha$ can be computed similarly by

\[
\text{Equating (14) and (15), and using the equations in (13) for } x_c, y_c, \text{ and } z_c, \text{ we can get a quadratic equation of } h \text{ as follows:}
\]

\[ph^2 + qh + r = 0\]

where

\[p = (1 - \cos^2 \alpha)(a^2 + b^2 + c^2),
\]

\[q = -2(1 - \cos^2 \alpha)(a^2 + b^2 + c^2) \times |a(x_2 - x_1) + b(y_2 - y_1) + (z_2 - z_1)|,
\]

\[r = [(a(x_2 - x_1) + b(y_2 - y_1) + (z_2 - z_1))^2 - \cos^2 \alpha(a^2 + b^2 + c^2)^2][(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2],
\]

\[a = A_{11} + D_{11} + G_{11} \cdot C_{11} \cdot F_{11} + I_{11},
\]

\[b = B_{11} + E_{11} + H_{11} \cdot C_{11} \cdot F_{11} + I_{11}.
\]

The possible root for $h$ can then be used to compute $x_c, y_c, \text{ and } z_c$ using (13).

On the calibration target, there are six vertices which can be used as the calibration points. From every two of them, we can compute a set of $(x_c, y_c, z_c)$ values as the camera position parameters. So more than one set of $(x_c, y_c, z_c)$ can be computed from the six vertices and they can be averaged to get a more accurate solution when real images are used in the calibration process.

In the above derivation, we know that the calibration target may be of any shape on the ground, provided that it has three groups of parallel lines with distinct directions. To get the depth vanishing point in the image, one group of parallel lines should be set to be parallel to the $Y$ axis of the world coordinate system, and the directions of the other two may be arbitrary. For example, shown in Fig. 7 is a ground texture with square tiles connected together, which can also be used as a calibration target. This type of texture can be seen frequently on the grounds of ordinary factories and buildings. The parallel lines in direction $d$ as shown in the figure may be used as the $Y$ axis of the world coordinate system, from which we can compute the depth vanishing point in the image. The points, for example, denoted as $P_{11}$,
Fig. 7. A ground texture with tiles connected together which could be used as a calibration target.

Fig. 8. An image of the calibration target of Fig. 1 after thresholding.

Fig. 9. An intermediate image processing result of Fig. 8.

Fig. 10. Imaging processing errors result in the deviation of the points from the vanishing line.

III. RELATED IMAGE PROCESSING TECHNIQUES

The images of the calibration target can be acquired with a TV camera. The edges of the target can then be extracted and the six vertices computed. From the six edges in the image, we can get three vanishing points, and by fitting these vanishing points, we can construct the vanishing line. From the information on the vanishing line, the orientation parameters of the camera and the focal length can be computed. The computed parameters, together with the known 3-D world coordinates and the 2-D image coordinates of the six hexagon vertices, can then be used to compute the camera position parameters.

In the following, we describe the related image processing techniques used in the proposed camera calibration method.

A. Finding the Hexagon Edges and Vertices

The line equations with respect to the image coordinate system for the six edges of the projected hexagon are computed as follows.

1) Acquire a gray image of the hexagon.
2) Threshold the image to produce a binary image, as shown in Fig. 8, the threshold value is computed automatically by moment-preserving thresholding [11].
3) Extract a set of approximate boundary points of the hexagon in the binary image.
4) Use the Hough transform to locate the six edges of the hexagon.
5) Use least-square-error fitting to compute more accurate line equations for the six edges from their edge points. Fig. 9 shows the result of superimposing the six computed lines in the hexagon image.

After the six edges are found, we can compute their intersections to get the six vertices of the projected hexagon.

B. Finding the Vanishing Points

First compute the line equations of the six hexagon edges in the image. Then compute the depth vanishing point \( V_0 \) by intersecting line \( \overline{P_1P_2} \) and line \( \overline{P_3P_4} \) in the image, as shown in Fig. 9, and the other two vanishing points \( V_1 \) and \( V_2 \) by intersecting line \( \overline{P_3P_4} \) and \( \overline{P_5P_6} \), and \( \overline{P_5P_6} \) and \( \overline{P_1P_2} \), respectively.

C. Finding the Vanishing Line

\( V_0, V_1, \) and \( V_2 \) are the three vanishing points produced by the projected parallel lines on the ground. Theoretically, they must be on the vanishing line. In practice, we fit these vanishing points in the least-square-error sense to get the line equation of the vanishing line.

D. Modifying the Vanishing Points

Accuracy of the vanishing points plays an important role on the performance of the proposed calibration method. It should be attempted to correct the coordinates of these points such that they can be located more accurately. Two methods are proposed here.

1) By the Vanishing Line:

In the ideal case, the vanishing points must be on the vanishing line. Due to camera distortion or image processing errors, these points may not stay on the line, as shown in Fig. 10. Therefore, we can use this line to adjust the vanishing points in order that they can all align on the line.

Let \( V_1 \) be a vanishing point obtained from intersecting lines \( L_1 \) and \( L_2 \). Let \( I_1, I_2 \) be the two intersections of the vanishing line and \( L_1 \) and \( L_2 \), respectively. Theoretically, \( V_1 \) must be on the vanishing line. As a compromise, we substitute the middle point between \( I_1 \) and \( I_2 \) for \( V_1 \). This modification is also applied to the other vanishing points.

2) By the Target Shape:

The correctness of the positions of the vanishing points dominates the accuracy of the calibration result. It is not difficult to figure out
that if the distance of a vanishing point is large, a little error in the line equations of the target edges will result in a large variation of the position of the vanishing point. In certain computer vision applications, the pan, the tilt, and the swing angles of the camera may all be small. For example, in the application to autonomous land vehicle guidance, the camera usually is mounted on the vehicle with little panning, tilting, and swinging. We can make use of this situation to select a proper target shape such that projected vanishing points will not be far away (i.e., the image coordinates of the vanishing points will not be large in magnitude).

In (1), when \( \theta, \phi, \) and \( \psi \) are small, the values of \( B \) and \( D \) approach to zero. So from (6) we can approximate the \( u \) coordinate of a vanishing point as follows:

\[
u = \frac{A \cdot f}{m_1 \cdot \varepsilon}.
\]

It is apparent that if the absolute slope value \( |m_1| \) is large, \( |\nu| \) will be small. Hence, as shown in Fig. 11, the target in Fig. 11(a) is better from the target. The lens center of the camera is assumed to be at \((0,-70,80)\) cm and the pan, tilt, and swing angles of the camera are assumed to be \(3^\circ, -30^\circ, \) and \(-6^\circ\), respectively. The perspective projection of the calibration target in the image was computed and perturbed by adding normally distributed noise to each pixel on the target boundary edges. Least-square-error fitting was used to find the boundary-line equation. The simulation results are listed in Table III, where the simulated noise has zero mean and varying standard deviations in the unit of pixel. For each noise deviation, 100 simulation results are generated and averaged. In the table, the distance error is defined as the difference between the computed distance from the camera lens center to the origin of the world coordinate system and the real one, and the distance error rate is the ratio of this difference to the real distance. And each of the other types of errors is defined as the difference between the computed parameter and the real one. It can be found that the results are tolerable when noise deviations are small.

Computer simulations have also been performed to analyze the relative errors in the presence of noise. A simulated camera with a focal length of 800 pixels was assumed to be at a reasonable distance from the target. The lens center of the camera is assumed to be at \((0,-70,80)\) cm and the pan, tilt, and swing angles of the camera are assumed to be \(3^\circ, -30^\circ, \) and \(-6^\circ\), respectively. The perspective projection of the calibration target in the image was computed and perturbed by adding normally distributed noise to each pixel on the target boundary edges. Least-square-error fitting was used to find the boundary-line equation. The simulation results are listed in Table III, where the simulated noise has zero mean and varying standard deviations in the unit of pixel. For each noise deviation, 100 simulation results are generated and averaged. In the table, the distance error is defined as the difference between the computed distance from the camera lens center to the origin of the world coordinate system and the real one, and the distance error rate is the ratio of this difference to the real distance. And each of the other types of errors is defined as the difference between the computed parameter and the real one. It can be found that the results are tolerable when noise deviations are small.

V. CONCLUSIONS

A new approach to camera calibration based on the use of the vanishing line is proposed in this correspondence. A monocular image of a hexagon shape is adequate for the calibration purpose. In addition to being able to compute the camera position, the viewing angles as well as the focal length can also be obtained. The computation is analytic; no iteration is necessary. This speeds up the calibration work.

The calibrated position error is found to be within 5% on the average. Only feasibility is emphasized in this study, though the error can be reduced further if improvements can be directed to the use of better imaging devices and more sophisticated image processing techniques. The proposed method is appropriate both for outdoor and for indoor computer vision applications like robot location and autonomous land vehicle guidance, because of its simplicity of environment setup.

REFERENCES

TABLE II
EXPERIMENTAL RESULTS OF CALIBRATING THE CAMERA POSITION PARAMETERS

<table>
<thead>
<tr>
<th>Measured Position Parameters</th>
<th>Computed Position Parameters</th>
<th>Average Error Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(94.5, -56.0, 16.5)</td>
<td>(89.83, -53.49, 17.02)</td>
</tr>
<tr>
<td>2</td>
<td>(198.0, -160.0, 15.7)</td>
<td>(198.23, -158.74, 17.16)</td>
</tr>
<tr>
<td>3</td>
<td>(297.5, -258.0, 15.5)</td>
<td>(295.29, -259.35, 15.72)</td>
</tr>
<tr>
<td>4</td>
<td>(92.0, -51.0, 30.0)</td>
<td>(91.11, -52.37, 30.54)</td>
</tr>
<tr>
<td>5</td>
<td>(196.5, -150.0, 30.0)</td>
<td>(195.26, -152.42, 30.63)</td>
</tr>
<tr>
<td>6</td>
<td>(298.5, -250.0, 30.0)</td>
<td>(298.90, -250.99, 30.96)</td>
</tr>
<tr>
<td>7</td>
<td>(99.0, -43.0, 41.0)</td>
<td>(100.72, -42.34, 42.24)</td>
</tr>
<tr>
<td>8</td>
<td>(204.0, -43.0, 41.5)</td>
<td>(208.72, -144.72, 43.39)</td>
</tr>
<tr>
<td>9</td>
<td>(303.0, -245.0, 41.5)</td>
<td>(307.60, -246.73, 42.37)</td>
</tr>
</tbody>
</table>

TABLE III
SIMULATION RESULTS

<table>
<thead>
<tr>
<th>Noise Deviation (pixel)</th>
<th>Pan Error (degree)</th>
<th>Tilt Error (degree)</th>
<th>Swing Error (degree)</th>
<th>Focal Length Error (pixel)</th>
<th>Distance Error (cm)</th>
<th>Distance Error Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.29</td>
<td>0.07</td>
<td>0.27</td>
<td>2.48</td>
<td>0.98</td>
<td>0.92</td>
</tr>
<tr>
<td>0.50</td>
<td>0.77</td>
<td>0.14</td>
<td>0.55</td>
<td>4.99</td>
<td>1.97</td>
<td>1.85</td>
</tr>
<tr>
<td>0.75</td>
<td>1.16</td>
<td>0.21</td>
<td>0.82</td>
<td>7.52</td>
<td>2.95</td>
<td>2.78</td>
</tr>
<tr>
<td>1.00</td>
<td>1.54</td>
<td>0.28</td>
<td>1.09</td>
<td>10.07</td>
<td>3.94</td>
<td>3.71</td>
</tr>
<tr>
<td>1.50</td>
<td>2.09</td>
<td>0.42</td>
<td>1.54</td>
<td>15.19</td>
<td>5.53</td>
<td>5.29</td>
</tr>
<tr>
<td>2.00</td>
<td>2.52</td>
<td>0.55</td>
<td>1.95</td>
<td>19.97</td>
<td>6.89</td>
<td>6.48</td>
</tr>
<tr>
<td>2.50</td>
<td>2.97</td>
<td>0.72</td>
<td>2.35</td>
<td>26.25</td>
<td>8.41</td>
<td>7.91</td>
</tr>
<tr>
<td>3.00</td>
<td>3.37</td>
<td>0.89</td>
<td>2.71</td>
<td>32.86</td>
<td>9.89</td>
<td>9.30</td>
</tr>
<tr>
<td>3.50</td>
<td>3.71</td>
<td>1.01</td>
<td>3.03</td>
<td>38.29</td>
<td>11.09</td>
<td>10.43</td>
</tr>
<tr>
<td>4.00</td>
<td>3.54</td>
<td>1.14</td>
<td>3.21</td>
<td>44.30</td>
<td>11.58</td>
<td>10.89</td>
</tr>
</tbody>
</table>

Least-Squares Estimation of Transformation Parameters Between Two Point Patterns

Shinji Umeyama

Abstract—In many applications of computer vision, the following problem is encountered: Two point patterns (sets of points) \( \{x_i\} \) and \( \{y_i\}; i = 1,2, \ldots, n \) are given in \( m \)-dimensional space, and we want to find the similarity transformation parameters (rotation, translation, and scaling) that give the least mean squared error between these point patterns. Recently Arun et al. and Horn et al. have presented a solution of this problem. Their solution, however, sometimes fails to give a correct rotation matrix and gives a reflection instead when the data is severely corrupted. The theorem given in this correspondence is a strict solution of the problem, and it always gives the correct transformation parameters even when the data is corrupted.

Index Terms—Absolute orientation problem, computer vision, least-squares, motion estimation, singular value decomposition.

I. INTRODUCTION

In computer vision applications, we sometimes encounter the following mathematical problem. We are given two point patterns (sets of points) \( \{x_i\} \) and \( \{y_i\}; i = 1,2, \ldots, n \) in \( m \)-dimensional space, and we want to find the similarity transformation parameters (\( R \): rotation, \( t \): translation, and \( c \): scaling) giving the minimum value

Manuscript received October 29, 1989; revised August 30, 1990. Recommended for acceptance by C. Brown.

The author is with the Mathematical Informatics Section, Information Science Division, Electrotechnical Laboratory, 1-1-4 Umezono, Tsukuba-shi, Ibaraki 305, Japan.

IEEE Log Number 9041531.