Communications

Determination of Robot Locations by Common Object Shapes

SHU-YUAN CHEN and WEN-HSIANG TSAI

Abstract—**A new approach to the determination of robot locations by common object shapes is proposed. Any object that has a polygon-shaped top and a lateral surface perpendicular to the top can be used for robot location in the approach, as long as the top shape is known in advance. Such objects are seen frequently both in indoor and outdoor environments. In addition, the solution provided by the approach can be computed analytically. These merits make the proposed approach more practical for general applications than other approaches using specially designed marks or requiring iterative computation. From a monocular image of an object, image processing and numerical analysis techniques are applied to extract the projection characteristics of the polygon corners on the object top surface, from which the position and the orientation parameters of a camera-mounted robot can be determined. Experimental results with location errors less than 5% prove the feasibility of the proposed approach. Error analysis that is useful for choosing better viewing angles to get more accurate location results is also included.**

I. Introduction

For robots to navigate automatically in various environments, it is important to determine their positions with respect to known objects in order to catch the objects or to avoid collision. One approach to this so-called robot location problem is to simulate human stereo vision [1]. In this way, binocular images are taken, and the image correspondence problem is solved to find 3-D range data.

Several researchers have approached the problem alternatively by the use of "standard marks" [2]–[6] to simplify the problem and to save computation time. The key idea is to use special marks that include a wealth of geometric information under perspective projection such that robot location parameters can be easily computed from monocular images of the marks.

Approaches to robot location using special marks can, in general, be categorized into two classes according to the criterion of whether or not the camera optical axis goes through the center of a mark. When the camera optical axis is constrained to go through the mark center [2]–[5], the camera location is represented only by position parameters since the orientation is implicitly determined by the constraint. In the other class of approaches [6], both the position and the orientation parameters are solved. Specially designed marks include a diamond [2], [3], a sphere with horizontal and vertical great circles [4], a circle pattern with an identification code [5], a house corner composed of three perpendicular lines [6], etc.

On the other hand, in the fields of photogrammetry, object recognition, and scene interpretation, there are several model-based techniques for object location determination from single views [7]–[18], which is the equivalent of the problem of robot location by the use of standard marks. Several model-based approaches [7]–[10] concentrated on using model points to determine the object location. The solutions are, in general, sensitive to erroneous point positions. Since computed image point positions are less accurate than computed line or curve equations, several researchers approached the problem alternatively by the use of model lines or curves [11]–[18].

Manuscript received May 11, 1988; revised March 15, 1989. This work was supported by the National Science Council, Republic of China, under Grant NSC77-0404-E009-031.

S. Y. Chen is with the Institute of Computer Science and Information Engineering, National Chaio Tung University, Hsinchu, Taiwan, R.O.C.

W. H. Tsai is with the Department of Computer and Information Science, National Chaio Tung University, Hsinchu, Taiwan, R.O.C.

IEEE Log Number 9040144.

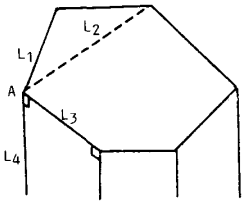

Fig. 1. Object structure and space pencil consisting of four lines L_1 through L_4.

In this communication, a new approach to robot location is proposed. The approach not only offers a location technique for general applications but also provides analytic solutions for fast computation. The shapes of common objects, such as cubes, furniture, machine parts, buildings, and so on, found in robot navigation environments can be used by the approach as the hints for robot location. Computed relative locations between a robot (or a camera) and objects can be used to plan collision-free trajectories for robot navigation in rooms, plan robot arm manipulation steps for object picking and loading, track targets for military attack, guide helicopters to land on buildings automatically, etc. Assumptions made about the objects for robot location include the following: 1) The shape of the object top is a flat polygon, and the top surface is perpendicular to a lateral surface of the object; 2) the geometric structure of the top surface of the object, including the length of each polygon side and the angle between any two neighboring sides, is known in advance.

Objects satisfying the above constraints are easy to find both in indoor and outdoor environments. This makes the proposed approach more practical for general applications than other approaches using specially designed marks. Moreover, no assumption is made about the height of the camera; the camera height can be derived as part of the solution. Although constraint 1) above makes the derivations of analytic solutions possible, it is desirable to extend the approach in further research in order to make use of more general objects.

More specifically, in the approach, a pair of opposite corners of the polygon on the top surface of the object is used as the standard mark. As shown in Fig. 1, with a virtual diagonal line drawn between a pair of opposite corners, the mark consists of a set of lines going through a common point. Such a mark is called a pencil of four lines, or more simply, a pencil in general [19]. It will herewith be called a space pencil. The common point (i.e., the object corner point A in Fig. 1) will be called the center of the pencil. A space pencil is composed of two parts: the coplanar part and the normal part. The former is a pencil of three coplanar lines on the object top surface (L_1, L_2, and L_3 in Fig. 1, for example), whereas the latter is a line going through the pencil center and perpendicular to the former (L_4 in Fig. 1, for example).

The shape of a space pencil appearing in an image is still a pencil of four lines if a virtual diagonal line is imagined to exist. It will be called an image pencil so that it can be distinguished from a space pencil. It can be easily observed that the shape of an image pencil varies relatively with respect to a viewer's position and angle. This provides, as is found in this study, enough information to obtain camera position and orientation parameters.

To locate a robot by the proposed approach, a monocular image of an object is taken first. A set of projection characteristics are then extracted from an image pencil including the pencil center position,

1042-296X/91/0200-0149$01.00

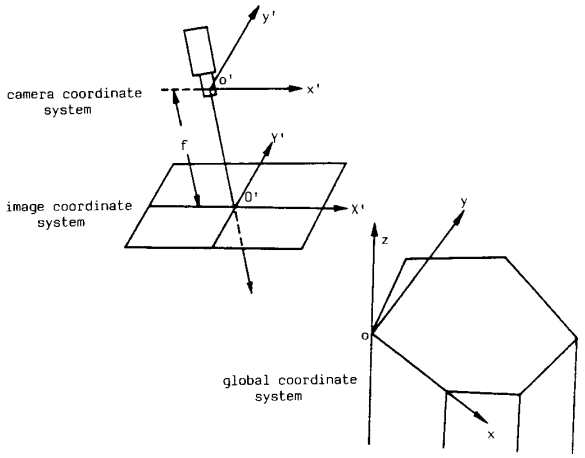

Fig. 2. Relations among camera coordinate system, image coordinate system, and global coordinate system.

the direction of each line, and the length of a polygon side. Together with the *a priori* knowledge about the corresponding space pencil, the characteristic values of the image pencil are finally substituted into a set of location formulas to compute the position as well as the orientation parameters of the camera on the robot.

In the remainder of this communication, the location formulas with respect to a space pencil are derived in Sections II through IV. In Section II, the characteristics of an image pencil are described as a set of equations according to 3-D imaging geometry. The equations are reduced into simpler forms in Section III, and then solved in Section IV. The image processing and numerical techniques used for the extraction of pencil characteristics are described in Section V. Experimental results and error analysis are included in Section VI and are followed by conclusions and suggestions for further research in Section VII.

II. Relations Between 3-D Space Pencils and 2-D Image Pencils

A. Terminologies and Definitions

Three coordinate systems are involved in the proposed robot location method, namely, the global coordinate system $o - x - y - z$, the camera coordinate system $o' - x' - y' - z'$, and the image coordinate system $O' - X' - Y'$. The geometrical relations among these three coordinate systems are illustrated in Fig. 2. The robot or the camera location is described with respect to the global system, which is attached on the object. In the right-handed global coordinate system, an object corner point is selected as the origin, and the normal line through the corner point, the top surface of the object, and one polygon side through the corner point on the top surface are chosen as the z axis, the x-y plane, and the x axis of the system, respectively. In the left-handed camera coordinate system, the origin is located at the camera lens center, the z' axis is aligned with the camera optical axis, and the x'-y' plane is parallel to the X'-Y' plane of the image plane in the image coordinate system. The image plane is located at $z' = f$, where f is the focal length of the camera.

As is illustrated in Fig. 3(a), the space pencil with center A is located at the origin of the global coordinate system, L_1 and L_2 are in the x-y plane, L_3 is aligned with the x axis, L_4 coincides with the z axis, and the length of L_1 is l. The direction of any line on the object top is defined as the angle measured from the x axis to the line in the counterclockwise direction, which is just a polar angle (called the space polar angle in the sequel). For example, the space polar angles θ_1 and θ_2 for L_1 and L_2, respectively, are sketched in Fig. 3(a). Hereafter, (θ_1, θ_2, l) will be called the space-pencil characteristics because they can be used to specify the structure of the pencil. Space-pencil characteristics can be obtained easily because the geometrical structure of the object top is assumed to be known in advance.

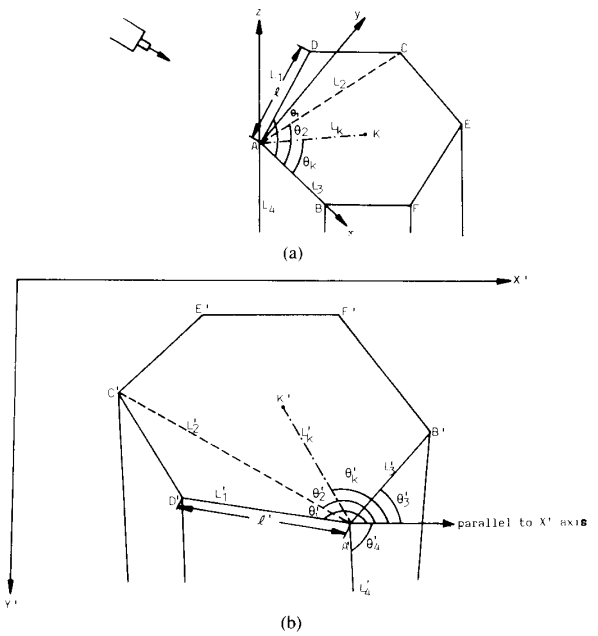

Fig. 3. Space pencil and its image: (a) Space pencil located at the origin; (b) corresponding image of (a).

A possible projection result of the object in Fig. 3(a) is shown in Fig. 3(b). Let A' with image coordinates (A'_x, A'_y) be the projection of the center of the space pencil located at A, L'_1 through L'_4 be the projection lines of L_1 through L_4, respectively, and the length of L'_1 be l'. The direction of any line in the image plane is defined as the angle measured from the X' axis to the line in the counterclockwise direction (called the image polar angle henceforth). The image polar angles θ'_1 through θ'_4 of lines L'_1 through L'_4, respectively, are shown in Fig. 3(b). $(A'_x, A'_y, \theta'_1, \theta'_2, \theta'_3, \theta'_4, l')$ will be called the image-pencil characteristics because they can be used to describe the projection of the corresponding space pencil. Such characteristic values can be obtained from the image of the object and the process will be discussed in Section V.

The desired camera location will be represented by three position parameters x_c, y_c, and z_c and three orientation parameters φ, θ, and δ, where φ, θ, and δ are called the pan, the tilt, and the swing angles, respectively, of the camera with respect to the global coordinate system. It is desired to derive the six camera location parameters in terms of the space-pencil characteristics and the image-pencil characteristics as the desired result of robot location.

B. Derivation of Image Pencil Characteristic Values

In this section, the seven values $(A'_x, A'_y, \theta'_1, \theta'_2, \theta'_3, \theta'_4, l')$ of the image-pencil characteristics will be derived in terms of the six camera parameters as well as the space-pencil characteristics (θ_1, θ_2, l). Refer to Fig. 3 for the following derivations of the image-pencil characteristics.

First, the coordinates (A'_x, A'_y) of the image A' of the origin A of the global system will be derived. Let K be any point in the 3-D space with global coordinates (x, y, z), and let K' be the projection of K in the image plane with image coordinates (K'_x, K'_y). Then, according to imaging geometry [20], [21], we have

$$K'_x = f \cdot (x \cdot M_{11} + y \cdot M_{21} + z \cdot M_{31} + x_0) / (x \cdot M_{13} + y \cdot M_{23} + z \cdot M_{33} + z_0) \quad (1)$$

$$K'_y = f \cdot (x \cdot M_{12} + y \cdot M_{22} + z \cdot M_{23} + y_0)/ (x \cdot M_{13} + y \cdot M_{23} + z \cdot M_{33} + z_0) \quad (2)$$

where

$$(M_{11}, M_{12}, M_{13}) = (\cos\varphi\cos\delta - \sin\varphi\cos\theta\sin\delta, -\cos\varphi\sin\delta - \sin\varphi\cos\theta\cos\delta, -\sin\varphi\sin\theta)$$

$$(M_{21}, M_{22}, M_{23}) = (\sin\varphi\cos\delta + \cos\varphi\cos\theta\sin\delta, -\sin\varphi\sin\delta + \cos\varphi\cos\theta\cos\delta, \cos\varphi\sin\theta)$$

$$(M_{31}, M_{32}, M_{33}) = (\sin\theta\sin\delta, \sin\theta\cos\delta, -\cos\theta)$$

$$(x_0, y_0, z_0) = (-x_c M_{11} - y_c M_{21} - z_c M_{31}, -x_c M_{12} - y_c M_{22} - z_c M_{32}, -x_c M_{13} - y_c M_{23} - z_c M_{33}). \quad (3)$$

The global coordinates of the origin A are $(0, 0, 0)$. Therefore, from (1) and (2), we have

$$A'_x = f \cdot x_0 / z_0 \quad (4)$$

$$A'_y = f \cdot y_0 / z_0. \quad (5)$$

Next, the image polar angles θ'_1, θ'_2, and θ'_3 of the three lines L'_1, L'_2, and L'_3, respectively, will be derived. Let L_k be a space line going through the origin A of the global system and a space point K with global coordinates (x, y, z); then, the image polar angle θ'_k of the corresponding projection L'_k in the image plane can be derived easily from (1) through (5). The result is

$$\tan\theta'_k = (x \cdot Q + y \cdot S + z \cdot U)/(x \cdot P + y \cdot R + z \cdot T) \quad (6)$$

where

$$(P, Q) = (-x_0 \cdot M_{13} + z_0 \cdot M_{11}, -y_0 \cdot M_{13} + z_0 \cdot M_{12})$$

$$(R, S) = (-x_0 \cdot M_{23} + z_0 \cdot M_{21}, -y_0 \cdot M_{23} + z_0 \cdot M_{22})$$

$$(T, U) = (-x_0 \cdot M_{33} + z_0 \cdot M_{31}, -y_0 \cdot M_{33} + z_0 \cdot M_{32}). \quad (7)$$

Since L_1, L_2, and L_3 with polar angles θ_1, θ_2, and θ_3 all go through the origin A of the global system and lie in the x-y plane, $(\cos\theta_i, \sin\theta_i, 0)$ is a point on L_i, $i = 1, 2$, and 3. Therefore, from (6) we have

$$\tan\theta'_1 = (Q \cdot \cos\theta_1 + S \cdot \sin\theta_1)/(P \cdot \cos\theta_1 + R \cdot \sin\theta_1) \quad (8)$$

$$\tan\theta'_2 = (Q \cdot \cos\theta_2 + S \cdot \sin\theta_2)/(P \cdot \cos\theta_2 + R \cdot \sin\theta_2) \quad (9)$$

$$\tan\theta'_3 = (Q \cdot \cos\theta_3 + S \cdot \sin\theta_3)/(P \cdot \cos\theta_3 + R \cdot \sin\theta_3). \quad (10)$$

Now, the image polar angle θ'_4 of L'_4 will be derived. Since L_4 coincides with the z axis, L'_4 is just the projection line of the z axis in the image plane. Since $(0, 0, z)$ is a point on L_4, from (6), we can get

$$\tan\theta'_4 = U/T. \quad (11)$$

Finally, the length of the projection line segment L'_1 is to be derived. Let space point D with global coordinates $(d_x, d_y, 0)$ be the other end point of L_1, which is projected onto the point D' in the image plane. From (1) through (5), the x component V'_{dx} and the y component V'_{dy} of the vector $\overrightarrow{A'D'}$ can be derived easily to be

$$V'_{dx} = f \cdot (d_x \cdot P + d_y \cdot R)/[(d_x \cdot M_{13} + d_y \cdot M_{23} + z_0) \cdot z_0]$$

$$V'_{dy} = f \cdot (d_x \cdot Q + d_y \cdot S)/[(d_x \cdot M_{13} + d_y \cdot M_{23} + z_0) \cdot z_0]$$

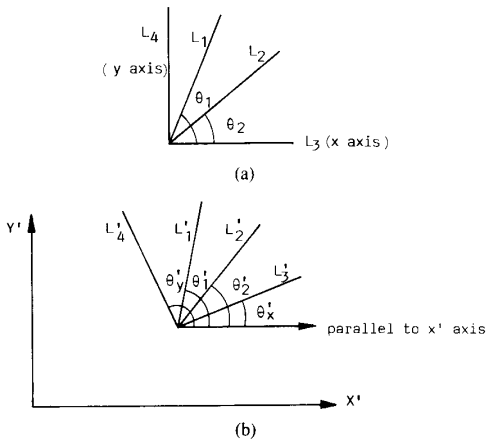

Fig. 4. Pencil of four coplanar lines and its image: (a) Space pencil; (b) corresponding image of (a).

which together with the equalities $d_x = l\cos\theta_1$ and $d_y = l\sin\theta_1$ can then be substituted into the length formula of l' (i.e., $l'^2 = V'^2_{dx} + V'^2_{dy}$) to get

$$l'^2 = (f/z_0)^2 \cdot (p^2 + q^2)/(c + z_0/l)^2 \quad (12)$$

where

$$(p, q, c) = (P \cdot \cos\theta_1 + R \cdot \sin\theta_1, Q \cdot \cos\theta_1 + S \cdot \sin\theta_1, M_{13} \cdot \cos\theta_1 + M_{23} \cdot \sin\theta_1). \quad (13)$$

In summary, we can represent the image pencil characteristic values $(A'_x, A'_y, \theta'_1, \theta'_2, \theta'_3, \theta'_4, l')$ in terms of the space pencil characteristic values (θ_1, θ_2, l) as well as the camera parameters $(x_c, y_c, z_c, \varphi, \theta, \delta)$ by (4), (5), and (8) through (12). These seven equations will be simplified in the next section, and the results will be solved to obtain the camera parameters in Section IV.

III. Properties of Image Polar Angles

In this section, the properties of image polar angles will be explored, based on which the seven equations derived in Section II can be reduced into six simpler equations. The key idea in the simplification is to use $(A'_x, A'_y, \theta'_x, \theta'_y, \theta'_z, l')$ instead of $(A'_x, A'_y, \theta'_1, \theta'_2, \theta'_3, \theta'_4, l')$ as the image-pencil characteristics, where θ'_x, θ'_y, and θ'_z represent the image polar angles of the x, the y, and the z axes of the global system, respectively. The validity of such a change of the image-pencil characteristic parameters will be proved first, followed by a discussion on the advantage of using the new parameters.

In fact, θ'_x and θ'_z are just θ'_3 and θ'_4 because L_3 and L_4 are just the x axis and the z axis, respectively (see Fig. 3). What is left is to derive the value of θ'_y in terms of other known characteristic parameters. This derivation is necessary because, in general, neither L_1 nor L_2 coincides with the y axis, and therefore, θ'_y cannot be obtained directly from the object images. The derivation can be done according to the following theorem.

Theorem 1 [property of unique determination]: Given a pencil of four coplanar space lines, including the x axis, the y axis, and two other noncollinear lines L_1 and L_2 with known space polar angles θ_1 and θ_2, respectively (see Fig. 4), if the image polar angles θ'_x, θ'_1 and θ'_2 of the x axis, L_1, and L_2, respectively, are known, then the unknown image polar angle θ'_y of the y axis can be computed by

$$\tan\theta'_y = (a \cdot \tan\theta'_1 + b \cdot \tan\theta'_2)/(a + b) \quad (14)$$

where

$$(a, b) = ((\tan\theta'_2 - \tan\theta'_x)\cot\theta_2, (\tan\theta'_x - \tan\theta'_1)\cot\theta_1).$$

Proof: Since space points (1, 0, 0) and (0, 1, 0) lie on the x axis and the y axis, respectively, from (6), θ'_x and θ'_y can be described by

$$\tan\theta'_x = Q/P \tag{15}$$

$$\tan\theta'_y = S/R \tag{16}$$

where P, Q, R, and S are as specified by (7). Since the values of θ'_1, θ'_2, and θ'_x are known, (8), (9), and (15) can be transformed into a system of three linear equations in terms of four unknown variables (i.e., P, Q, R, and S). The results are

$$\begin{aligned} P\cdot\tan\theta'_1\cdot\cot\theta_1 + R\cdot\tan\theta'_1 - Q\cdot\cot\theta_1 + S = 0 \\ P\cdot\tan\theta'_2\cdot\cot\theta_2 + R\cdot\tan\theta'_2 - Q\cdot\cot\theta_2 + S = 0 \\ P\cdot\tan\theta'_x - Q = 0. \end{aligned} \tag{17}$$

Since the desired $\tan\theta'_y$ is a ratio of the variable S to the variable R as described by (16), the key idea of the remaining proof is to solve the above underdetermined homogeneous system to obtain the ratio S/R in terms of the known values of θ_1, θ_2, θ'_1, θ'_2, and θ'_x. For this, the Cramer rule can be used. The result is

$$\begin{aligned} S/R = &\left[(\tan\theta'_2 - \tan\theta'_x)\tan\theta'_1\cot\theta_2\right. \\ &\left.+ (\tan\theta'_x - \tan\theta'_1)\tan\theta'_2\cot\theta_1\right] / \\ &\left[(\tan\theta'_2 - \tan\theta'_x)\cot\theta_2 + (\tan\theta'_x - \tan\theta'_1)\cot\theta_1\right], \\ = &\tan\theta'_y, \end{aligned}$$

which is just (14). This completes the proof of Theorem 1. ■

The foregoing theorem is actually a degenerate case of a more general theorem proved in Chen and Tsai [22], which says that a pencil of multiple coplanar lines including at least three noncollinear lines with known corresponding space polar angles, if the image polar angles of any three noncollinear lines are provided, the image polar angle of any fourth line in the pencil can then be uniquely determined. It is a property similar to that of the cross ratio discussed in Duda and Hart [19] and in Rosenfeld and Kak [23]. The proof in [22] is too long to be included in this communication.

Now, based on Theorem 1, the value of θ'_y can be easily derived in terms of θ_1, θ_2, θ'_1, θ'_2, and θ'_x. Therefore, (16) can be used to replace (8) and (9). Furthermore, (15) and $\tan\theta'_x = U/T$ can be used to replace (10) and (11), respectively, since θ'_x and θ'_z are just θ'_3 and θ'_4, respectively. As a summary, we have the following six simpler equations for the image-pencil characteristics $(A'_x, A'_y, \theta'_x, \theta'_y, \theta'_z, l')$ in terms of the space-pencil characteristics and the camera parameters:

$$A'_x = f\cdot x_0/z_0 \tag{18}$$

$$A'_y = f\cdot y_0/z_0 \tag{19}$$

$$\tan\theta'_x = Q/P \tag{20}$$

$$\tan\theta'_y = S/R \tag{21}$$

$$\tan\theta'_z = U/T \tag{22}$$

$$l'^2 = (f/z_0)^2\cdot(p^2+q^2)/(c+z_0/l)^2 \tag{23}$$

where x_0, y_0, and z_0 are as specified in (3); $\tan\theta'_y$ is as specified in (14); P, Q, R, S, T, and U are as specified in (7); and p, q, and c are as specified in (13).

The advantage of using the image polar angles of the three orthogonal axes (the x, y, and z axes) can be demonstrated by the following theorem.

Theorem 2 [property of independence]: The image polar angles of the x, y, and z axes are independent of the camera position parameters x_c, y_c, and z_c, respectively (i.e., the image polar angle of the x axis can be computed in terms of the y_c and z_c without involving x_c, and so on).

Proof: The elements M_{ij} of the matrix M described in (3) have the following properties of orthonormality [24], which will be used for equation reduction in the sequel:

$$\begin{aligned} (M_{11}, M_{12}, M_{13}) &= (M_{23}\cdot M_{32} - M_{22}\cdot M_{33}, M_{21}\cdot M_{33} \\ &\quad - M_{23}\cdot M_{31}, M_{22}\cdot M_{31} - M_{21}\cdot M_{32}) \\ (M_{21}, M_{22}, M_{23}) &= (M_{33}\cdot M_{12} - M_{32}\cdot M_{13}, M_{31}\cdot M_{13} \\ &\quad - M_{33}\cdot M_{11}, M_{32}\cdot M_{11} - M_{31}\cdot M_{12}) \\ (M_{31}, M_{32}, M_{33}) &= (M_{13}\cdot M_{22} - M_{12}\cdot M_{23}, M_{11}\cdot M_{23} \\ &\quad - M_{13}\cdot M_{21}, M_{12}\cdot M_{21} - M_{11}\cdot M_{22}). \end{aligned} \tag{24}$$

Substituting the values of x_0, y_0, and z_0 in (3) into (7) and using (24) to reduce the results, we can get

$$\begin{aligned} (P, Q) &= (-y_c\cdot M_{32} + z_c\cdot M_{22}, y_c\cdot M_{31} - z_c\cdot M_{21}) \\ (R, S) &= (x_c\cdot M_{32} - z_c\cdot M_{12}, -x_c\cdot M_{31} + z_c\cdot M_{11}) \\ (T, U) &= (-x_c\cdot M_{22} + y_c\cdot M_{12}, x_c\cdot M_{21} - y_c\cdot M_{11}). \end{aligned} \tag{25}$$

Then, the theorem can be seen to be true by observing the results of substituting the above values of P, Q, R, S, T, and U into (20), (21), and (22). This completes the proof of Theorem 2. ■

The above property of the image polar angles θ'_x, θ'_y, and θ'_z facilitates the derivations of the analytic solutions for the camera parameters, as will be seen in the next section.

IV. Computation of Camera Location Parameters

In this section, we will solve the six equations derived in Section III [i.e., (18) through (23)] to obtain the desired camera location including the position parameters (x_c, y_c, z_c) and the orientation parameters $(\varphi, \theta, \delta)$. The procedure can be decomposed into two stages. The first stage is to solve the orientation parameters $(\varphi, \theta, \delta)$, and the second is to compute the position parameters (x_c, y_c, z_c) in terms of the orientation parameters. Four steps are involved in the first stage: 1) combining (18) and (19) to derive three ratios among x_c, y_c, and z_c; 2) employing the property of independence of Theorem 2 to derive another set of ratios among x_c, y_c, and z_c using (20), (21) and (22); 3) eliminating x_c, y_c, and z_c from these two sets of ratios to get three equations involving only orientation parameters φ, θ, and δ; and 4) manipulating the three equations to get the solutions for the orientation parameters. In the second stage, the results of the first stage as well as (23) for determining the scaling factor are used to compute the position parameters. The detailed procedure is described in the following.

In stage 1, the orientation parameters are computed by the following four steps.

Step 1—Derivation of a set of ratios among x_c, y_c, and z_c: Eliminating x_c from (18) and (19), regrouping the terms of y_c and z_c, and reducing the results by using (24), we get

$$\begin{aligned} y_c/z_c = &(A'_x\cdot M_{21} + A'_y\cdot M_{22} + f\cdot M_{23}) / \\ &(A'_x\cdot M_{31} + A'_y\cdot M_{32} + f\cdot M_{33}). \end{aligned} \tag{26}$$

In similar ways, from (18) and (19) and using (24), the ratio of x_c to z_c and that of x_c to y_c can be obtained, respectively, as

$$\begin{aligned} x_c/z_c = &(A'_x\cdot M_{11} + A'_y\cdot M_{12} + f\cdot M_{13}) / \\ &(A'_x\cdot M_{31} + A'_y\cdot M_{32} + f\cdot M_{33}) \end{aligned} \tag{27}$$

$$\begin{aligned} x_c/y_c = &(A'_x\cdot M_{11} + A'_y\cdot M_{12} + f\cdot M_{13}) / \\ &(A'_x\cdot M_{21} + A'_y\cdot M_{22} + f\cdot M_{23}). \end{aligned} \tag{28}$$

Step 2—Derivation of another set of ratios among x_c, y_c, and z_c: On the other hand, substituting the values of P and Q in (25) into (20) will result in an equation with no term of x_c (by Theorem 2). We can then regroup the terms of y_c and z_c in the resulting

equation and get another ratio of y_c to z_c as

$$y_c/z_c = (M_{21} + \tan\theta'_x \cdot M_{22})/(M_{31} + \tan\theta'_x \cdot M_{32}). \quad (29)$$

Applying similar steps to (21) and (22), respectively, we can get two other ratios as

$$x_c/z_c = (M_{11} + \tan\theta'_y \cdot M_{12})/(M_{31} + \tan\theta'_y \cdot M_{32}) \quad (30)$$

$$x_c/y_c = (M_{11} + \tan\theta'_z \cdot M_{12})/(M_{21} + \tan\theta'_z \cdot M_{22}). \quad (31)$$

Step 3—Derivation of three equations involving only orientation parameters: Eliminating y_c and z_c from (26) and (29) and using (24) to simplify the result, we get an equation involving only orientation parameters. Then, substituting the values of M_{11}, M_{12}, and M_{13} in (3) into the equation, dividing the equation by $\cos\varphi$ and simplifying the result, we get

$$\tan\varphi = (f \cdot \tan\theta'_x \cos\delta + f \cdot \sin\delta)/ (f \cdot \tan\theta'_x \cos\theta \sin\delta - f \cdot \cos\theta \cos\delta + B'_x \cdot \sin\theta) \quad (32)$$

where

$$B'_x = A'_y - A'_x \cdot \tan\theta'_x. \quad (33)$$

Similarly, from (27), (30), and (24) and from (28), (31), and (24), we can get, respectively

$$\tan\varphi = (f \cdot \tan\theta'_y \cos\theta \sin\delta - f \cdot \cos\theta \cos\delta + B'_y \cdot \sin\theta)/ (-f \cdot \tan\theta'_y \cos\delta - f \cdot \sin\delta) \quad (34)$$

$$\tan\theta = B'_z/(f \cdot \tan\theta'_z \sin\delta - f \cdot \cos\delta) \quad (35)$$

where

$$B'_y = A'_y - A'_x \cdot \tan\theta'_y \quad (36)$$

$$B'_z = A'_y - A'_x \cdot \tan\theta'_z. \quad (37)$$

Step 4—Derivation of the solutions for orientation parameters: Equating (32) and (34) and considering $\tan\theta$ as a variable, we can transform the result into the following equation

$$A_\theta \tan^2\theta + B_\theta \tan\theta + C_\theta = 0 \quad (38)$$

where

$$A_\theta = f^2(\tan\theta'_x \cos\delta + \sin\delta)(\tan\theta'_y \cos\delta + \sin\delta) + B'_x \cdot B'_y$$

$$B_\theta = f \cdot \sin\delta \cdot m_\delta - f \cdot \cos\delta \cdot o_\delta$$

$$C_\theta = f^2 \cdot n_\delta$$

with B'_x and B'_y as specified in (33) and (36), respectively, and

$$(m_\delta, n_\delta, o_\delta) = (B'_x \tan\theta'_y + B'_y \tan\theta'_x, \tan\theta'_x \tan\theta'_y + 1, B'_x + B'_y). \quad (39)$$

On the other hand, it is noted that no term of $\sin\varphi$ or $\cos\varphi$ is involved in (35); such terms were eliminated in the reduction. Substituting (35) (which expresses the value of $\tan\theta$ in terms of $\sin\delta$ and $\cos\delta$) into (38) and simplifying the result, we get a quadratic equation with $\tan\delta$ as a variable. Then, the quadratic equation can be solved to obtain

$$\tan\delta = \frac{-B_\delta \pm \sqrt{B_\delta^2 - 4A_\delta C_\delta}}{2A_\delta}$$

where

$$A_\delta = B'_x B'_y B'^2_z + f^2 B'^2_z + f^2 B'_z \tan\theta'_z \cdot m_\delta + f^4 \tan^2\theta'_z \cdot n_\delta$$

$$B_\delta = f^2 B'^2_z \cdot p_\delta - f^2 B'_z \tan\theta'_z \cdot o_\delta - f^2 B'_z \cdot m_\delta - 2f^4 \tan\theta'_z \cdot n_\delta$$

$$C_\delta = B'_x B'_y B'^2_z + f^2 B'^2_z \cdot q_\delta + f^2 \cdot B'_z \cdot o_\delta + f^4 \cdot n_\delta$$

with B'_x, B'_y, B'_z, m_δ, n_δ, and o_δ as specified in (33), (36), (37), and (39), and

$$(p_\delta, q_\delta) = (\tan\theta'_x + \tan\theta'_y, \tan\theta'_x \tan\theta'_y).$$

Note that the above equation has two solutions as indicated by the $\pm$ signs in front of the square root; the one leading to a positive tilt angle in (35) is the desired one because the robot must be high enough to see the object top.

After the value δ is computed, it can be substituted into (35) to obtain the value of $\tan\theta$ or θ. Thereafter, the values of δ and θ can be substituted into (32) to obtain the value of $\tan\varphi$ or φ. This completes the derivation of the camera orientation parameters.

In stage 2, the camera position parameters are computed in the following way. First, the values of δ, θ, and φ can be substituted into (30) and (29) to obtain

$$r_x = x_c/z_c = (M_{11} + \tan\theta'_y \cdot M_{12})/(M_{31} + \tan\theta'_y \cdot M_{32})$$

$$r_y = y_c/z_c = (M_{21} + \tan\theta'_x \cdot M_{22})/(M_{31} + \tan\theta'_x \cdot M_{32}) \quad (40)$$

respectively. Next, substituting the values of P, Q, R, and S in (25) into (23) and replacing all x_c and y_c terms by $x_c = r_x z_c$ and $y_c = r_y z_c$, respectively, we get a quadratic equation in terms of z_c. Then, the quadratic equation can be solved to get

$$z_c = \frac{-B_z \pm \sqrt{B_z^2 - 4A_z C_z}}{2A_z}$$

where

$$A_z = l'^2 \cdot r_z^4,$$

$$B_z = 2l \cdot l'^2 \cdot r_z^3 \cdot c$$

$$C_z = l^2 \cdot (l'^2 \cdot r_z^2 \cdot c^2 - f^2 \cdot r_p^2 - f^2 \cdot r_q^2)$$

with c, r_x, and r_y as specified in (13) and (42), and

$$r_z = -r_x \cdot M_{13} - r_y \cdot M_{23} - M_{33}$$

$$r_p = (-r_y \cdot M_{32} + M_{22}) \cdot \cos\theta_1 + (r_x \cdot M_{32} - M_{12}) \cdot \sin\theta_1.$$

$$r_q = (r_y \cdot M_{31} - M_{21}) \cdot \cos\theta_1 + (-r_x \cdot M_{31} + M_{11}) \cdot \sin\theta_1.$$

The positive value of z_c is the desired one satisfying the constraint that the robot is high enough to see the object top. After the value of z_c is computed, we can compute x_c and y_c from (40). This completes the derivation of the camera position parameters.

V. Pencil-Line Detection by Image Analysis Techniques

Image and numerical analysis techniques are used in this study to find pencil lines in object images. First of all, line segments in object images are found by edge detection, smoothing, and thinning. Three object images and the corresponding processing results are shown in Fig. 5. Sobel edge operators [23] are used to compute edge values. Eight-neighbor averaging is applied to the edge value map in order to smooth out noise and to avoid hole creation during thinning. Pixels with edge values larger than a threshold are detected as edge points. For thinning, the fast algorithm proposed by Chen and Hsu [25] is adopted.

To detect object lines, the Hough transform [26] is applied. After that, any line in the image with its length large enough is detected. The parameters of the equations computed from the detected lines are not accurate enough for the purpose of robot location. Improvement is made by fitting the point set of each detected line to a line equation in the least-square-error sense [27].

It is necessary to separate desirable object lines from noisy ones among the detected lines. This is accomplished by checking the condition that each desirable line must possess at least one end point that is very close to an end point of another line. In each set of

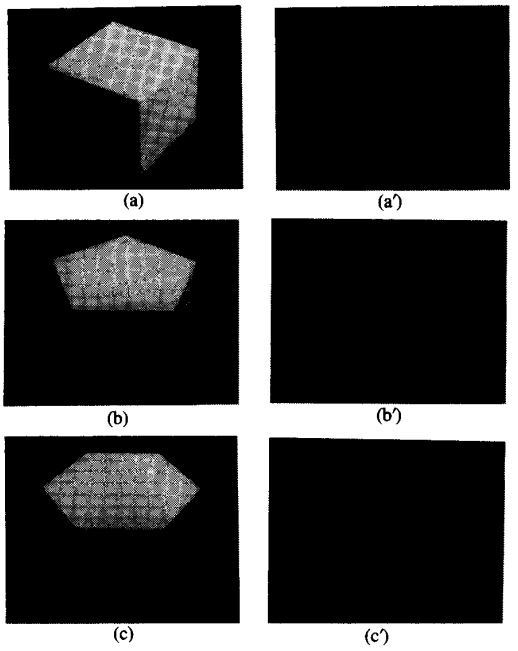

Fig. 5. Object images of three wood objects and corresponding preprocessing results: (a) Cube image; (b) pentagonal prism image; (c) hexagonal prism image; (a'), (b'), and (c') are the results of (a), (b), and (c), respectively.

detected close object lines, the intersection point of every two lines is computed as an object corner point, and each set of close triple lines is regarded as a pencil.

After the coordinates of all corner points and the direction parameters of all pencil lines are computed, it is easy to get the image-pencil characteristics, including the image polar angle of a virtual diagonal line, which can be computed from the coordinates of a pair of opposite corner points. The image-pencil characteristics are finally used for computing the camera location parameters.

VI. Experimental Results and Error Analysis

Experiments have been performed on a system consisting of a PC-EYE imaging interface and a CCD camera for taking pictures with a resolution of $640 \times 400 \times 4$ b and a Sun workstation for preprocessing and location computation.

In order to get precise location, calibration works for computing the image center and the camera focal length must be done first. The calibration procedures were based on Chen and Tsai [28]. Imaging distortion correction was not necessary since such distortion existing in the CCD camera pictures was found to be negligible [28].

Three wood objects have been made for testing, including a cube, a pentagonal prism, and a hexagonal prism. Each object was placed at eight different locations, and a picture was taken from each object at each location. Three images of the wood objects and the corresponding preprocessing results are shown in Fig. 5. The camera position data x_c, y_c, and z_c were manually measured as the references for location result checking. After each object image was processed, location solutions were computed by the formulas derived in Section IV.

For each object image, the camera position parameters were compared with the real reference positions to compute the location accuracy. The deviation of each computed position value from the reference one was computed, and the ratio of the deviation to the reference value was defined as the error percentage. The experimental results are shown in Table I. From the table, all the error percentages can be seen to be less than 5%, which shows that the approach is feasible for practical applications.

A computer simulation has additionally been performed to analyze the influence of the orientations of viewing angles on the errors of computed camera position parameters. In addition, since the work by Hung *et al.* [9] also relaxed constraints on camera locations, their results will also be discussed and compared to those obtained by our method. The analysis results include too many graphic diagrams to be included here. We only mention the main conclusions [22].

First, it was found that our method is sensitive in some orientations, for example, when the tilt angle approaches 90° or 0° or when the pan angle approaches −90° or 0°. The reason is that in these critical orientations, the lengths of some image pencil line segments become very small, resulting in erroneous coefficient values in the computed pencil line equations since the least-square-error fitting is sensitive to the lengths of line segments. The erroneous pencil line equation coefficients in turn affect the accuracy of the computation results in the image pencil characteristics, leading to erroneous computation results of the position parameters. The results are as intuitively expected because the angles $\varphi = -90$, $\varphi = 0$, $\theta = 90$, and $\theta = 0°$ represent the orientations for the camera optical axes to be parallel to the x-z, y-z, and x-y planes and normal to the x-y plane of the global system, respectively.

Compared with the Hung *et al.* algorithm [9]. Our method is worse than theirs in some critical orientations, for example, when the tilt angle approaches 0°. However, since more information is used in our method and since computed image line equations are more accurate than computed image point positions, our method has more accurate results than theirs in general orientations.

VII. Conclusions and Suggestions

A new approach to robot location is presented in this communication. The approach utilizes space pencils, which exist on common objects, as the standard marks. If the space-pencil characteristics of an object can be provided in advance, analytic formulas for camera position and orientation determination can be derived using the corresponding image-pencil characteristics in an object image. These merits make the proposed approach more practical for general applications than other approaches using specially designed marks

TABLE I
EXPERIMENTAL RESULT OF ROBOT LOCATION
(Each table entry of computed values consists of three parts: (1) computed position data, (2) deviation (in parentheses), and (3) error percentage. The unit of position data is millimeters.)

Object Location			x_c	y_c	z_c
1	Reference Value		-435.00	-125.00	275.00
	Computed Value from	Cube	-434.18(0.82),0.19%	-119.58(5.42),4.34%	278.54(3.54),1.29%
		Pentagonal Prism	-435.92(0.92),0.21%	-131.16(6.16),4.93%	276.05(1.05),0.38%
		Hexagonal Prism	-432.53(2.47),0.57%	-122.97(2.03),1.62%	273.66(1.34),0.49%
2	Reference Value		-460.00	-155.00	275.00
	Computed Value from	Cube	-460.75(0.75),0.16%	-152.33(2.67),1.72%	274.92(0.08),0.03%
		Pentagonal Prism	-455.95(4.05),0.88%	-162.35(7.35),4.74%	278.29(3.29),1.20%
		Hexagonal Prism	-462.47(2.47),0.54%	-155.30(0.30),0.19%	272.46(2.54),0.92%
3	Reference Value		-475.00	-120.00	275.00
	Computed Value from	Cube	-476.65(1.65),0.35%	-124.24(4.24),3.53%	285.08(10.08),3.67%
		Pentagonal Prism	-477.81(2.81),0.59%	-119.76(0.24),0.20%	273.99(1.01),0.37%
		Hexagonal Prism	-470.84(4.16),0.88%	-115.22(4.78),3.98%	270.29(4.71),1.71%
4	Reference Value		-505.00	-145.00	275.00
	Computed Value from	Cube	-505.85(0.85),0.17%	-147.86(2.86),1.97%	270.94(4.06),1.48%
		Pentagonal Prism	-514.83(9.83),1.95%	-142.31(2.69),1.85%	285.88(10.88),3.96%
		Hexagonal Prism	-506.30(1.30),0.26%	-145.51(0.51),0.35%	277.63(2.63),0.96%
5	Reference Value		-785.00	-195.00	290.00
	Computed Value from	Cube	-792.24(7.24),0.92%	-202.30(7.30),3.74%	290.19(0.19),0.07%
		Pentagonal Prism	-775.39(9.61),1.22%	-200.07(5.07),2.60%	285.84(4.16),1.43%
		Hexagonal Prism	-776.53(8.47),1.08%	-192.56(2.44),1.25%	283.36(6.64),2.29%
6	Reference Value		-845.00	-255.00	290.00
	Computed Value from	Cube	-846.81(1.81),0.21%	-245.11(9.89),3.88%	280.44(9.56),3.29%
		Pentagonal Prism	-833.51(11.49),1.36%	-246.43(8.57),3.36%	301.04(11.04),3.81%
		Hexagonal Prism	-835.19(9.81),1.16%	-259.82(4.82),1.89%	287.36(2.64),0.91%
7	Reference Value		-1030.00	-300.00	380.00
	Computed Value from	Cube	-1028.13(1.87),0.18%	-301.70(1.70),0.57%	397.47(17.47),4.60%
		Pentagonal Prism	-1034.71(4.71),0.46%	-294.44(5.56),1.85%	377.29(2.71),0.71%
		Hexagonal Prism	-1024.36(5.64),0.55%	-300.56(0.56),0.19%	376.59(3.41),0.90%
8	Reference Value		-1090.00	-350.00	380.00
	Computed Value from	Cube	-1086.19(3.81),0.35%	-353.00(3.00),0.86%	365.76(14.24),3.75%
		Pentagonal Prism	-1085.92(4.08),0.37%	-351.48(1.48),0.42%	396.48(16.48),4.34%
		Hexagonal Prism	-1105.10(15.10),1.39%	-339.58(10.42),2.98%	385.52(5.52),1.45%
Average Error Percentage			0.67%	2.21%	1.83%

or requiring iterative computation. Furthermore, in the derivations of location formulas, a property of coplanar lines similar to that of the cross ratios of collinear points is proved. This property makes the derivations of analytic solutions possible.

Good location results were obtained experimentally. The error percentages were all less than 5%. Using better imaging devices, performing more accurate focal length calibration, and employing subpixel preprocessing techniques [29], [30], etc., are possible ways to upgrade robot location accuracy. Simulation results were also included, and these results provide a guidance for choosing better orientations of viewing angles to get more accurate location results. Further research may be directed towards implementing the proposed approach for outdoor object location, applying the robot location approach to mobile robot guidance, generalizing the shapes of the object tops from polygons to curved figures, extending the approach by using a more general standard mark composed of several lines with any known relative positions and orientations, etc.

REFERENCES

[1] H. P. Moravec, "The Standford cart and the CMU rover," *Proc. IEEE*, vol. 71, pp. 872-884, 1983.

[2] I. Fukui, "TV image processing to determine the position of a robot vehicle," *Patt. Recognition*, vol. 14, pp. 101-109, 1981.

[3] J. W. Courtney and J. K. Aggarwal, "Robot guidance using computer vision," *Patt. Recognition*, vol. 17, pp. 585-592, 1984.

[4] M. J. Magee and J. K. Aggarwal, "Determining the position of a robot using a single calibration object," in *Proc. IEEE Conf. Robotics* (Atlanta, GA), Mar. 1984, pp. 140-149.

[5] M. R. Kabuka and A. E. Arenas, "Position verification of a mobile robot using standard pattern," *IEEE J. Robotics Automat.*, vol. RA-3, pp. 505-516, 1987.

[6] H. L. Chou and W. H. Tsai, "A new approach to robot location by house corners," *Patt. Recognition*, vol. 19, pp. 439-451, 1986.

[7] P. R. Wolf, *Elements of Photogrammetry*. New York: McGraw-Hill, 1983.

[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. Ass. Comput. Mach.*, vol. 24, pp. 381-395, 1981.

[9] Y. Hung, P. S. Yeh, and D. Harwood, "Passive ranging to known planar point sets," *Proc. IEEE Conf. Robotics Automat.* (St. Louis, MO), Mar. 1985, pp. 80-85.

[10] S. Linnainmaa, D. Harwood, and L. S. Davis, "Pose determination of a three-dimensional object using triangle pairs," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, pp. 634-647, 1988.

[11] M. F. Augusteijn and C. R. Dyer, "Recognition and recovery of the three-dimensional orientation of planar point patterns," *Comput. Vision Graphics Image Processing*, vol. 36, pp. 76-99, 1986.
[12] D. G. Lowe, *Perceptual Organization and Visual Recognition*. Boston: Kluwer Academic, 1985.
[13] ——, "Three-dimensional object recognition from single two-dimensional images," *Artificial Intell.*, vol. 31, pp. 355-395, 1987.
[14] R. Horaud, "New methods for matching 3-D objects with single perspective views," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 401-412, 1987.
[15] T. Kanade, "Recovery of the three-dimensional shape of an object from a single view," *Artificial Intell.*, vol. 17, pp. 409-460, 1981.
[16] S. T. Barnard, "Interpreting perspective images," *Artificial Intell.*, vol. 21, pp. 435-462, 1983.
[17] ——, "Choosing a basis for perceptual space," *Comput. Vision Graphics Image Processing*, vol. 29, pp. 87-99, 1985.
[18] R. M. Haralick and Y. H. Chu, "Solving camera parameters from the perspective projection of a parameterized curve," *Patt. Recognition*, vol. 17, pp. 637-645, 1984.
[19] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.
[20] J. D. Foley and A. Van Dam, *Fundamentals of Interactive Computer Graphics*. Reading, MA: Addison-Wesley, 1982.
[21] D. H. Ballard and C. M. Brown, *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
[22] S. Y. Chen and W. H. Tsai, "Analytic approaches to robot location by common object shapes," MIST Tech. Rep., TR-MIST-C78001, Microelectron. Inform. Sci. Technol. Res. Cen., National Chiao Tung Univ., Taiwan, 1989.
[23] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, 2nd ed. New York: Academic, 1982, vol. 2.
[24] S. Ganapathy, "Decomposition of transformation matrices for robot vision," in *Proc. IEEE Conf. Robotics* (Atlanta, GA), Mar. 1984, pp. 130-139.
[25] Y. S. Chen and W. H. Hsu, "A modified fast parallel algorithm for thinning digital patterns," *Patt. Recognition Lett.*, vol. 7, pp. 99-106, 1988.
[26] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Commun. Ass. Comput. Mach.*, vol. 15, pp. 11-15, 1972.
[27] S. A. Dudani and A. L. Luk, "Locating straight-line edge segments on outdoor scenes," *Patt. Recognition*, vol. 10, pp. 145-157, 1978.
[28] R. K. Chen and W. H. Tsai, "3-D object surface data acquisition using laser striping," in *Proc. Workshop Comput. Vision Graphics Image Processing* (Chitou, Taichung, Taiwan, Republic of China), Aug. 1987, pp. 31-51.
[29] L. H. Chen and W. H. Tsai, "Moment-preserving line detection," *Patt. Recognition*, vol. 21, pp. 45-53, 1988.
[30] L. H. Chen and W. H. Tsai, "Moment-preserving curve detection," *IEEE Trans. Syst. Man Cybern.*, vol. 18, pp. 148-158, 1988.

Robot Force Sensor Interacting with Environments

Yuan F. Zheng and Yuka Fan

Abstract—**In this communication, we first study the impact effect on a robotic system with a force-sensing device installed between the end effector and the end of the robot. The results show that physical contact of a robot with environments causes abrupt changes of velocity and force to the end effector and the force sensor, respectively.**

We then study the dynamic behavior of the force sensor following the impact. Theoretical analysis and experiments show that the sensor generates a low-to-high transition signal corresponding to an impact, which can be used to identify the contact. To utilize this signal, an electronic switch is designed. The signal triggers the switch, which in turn interrupts the control computer. The control computer can then stop the robot motion immediately after an impact occurs. As a result, damage to, or malfunction of, the robotic system following the impact can be avoided. Experimental results show that the proposed switching mechanism works effectively.

Manuscript received July 1988; revised January 30, 1990. This work was supported by the National Science Foundation under Grant DMC-8620409 and by a Presidential Young Investigator Award to Y. F. Zheng.

The authors are with the Department of Electrical Engineering, Ohio State University, Columbus, OH 43210.

IEEE Log Number 9040143.

I. Introduction

Industrial robots are used for moving objects, handling materials, assembling parts, and manipulating tools. In all these operations, the robot must come into physical contact with the object before the desired force and movement can be made on it [1]. In fact, robot operations can be divided into three phases, i.e., motion in free space, physical contact, and exertion of a force [1], [2]. The same situation is true for a legged walking robot whose robot legs experience the three phases when the robot walks.

Although a great deal of work has been conducted on the first and third phases, much less attention has been given to the phase of the physical contact. The contact is a short transition from the position control in the free space to the force control in a constrained subspace. A variety of physical phenomena occur in the transition. Paul pointed out in his recent work [2] that the robot was stopped from moving at some velocity when a contact took place. The energy was absorbed by the compliances in the system. This could destroy many mechanical components such as precision gears, shafts, actuators, etc. He further pointed out that the use of any form of force sensing aggravates this problem because the force sensor was the least stiff member of the system, the most fragile, and absorbed all the energy [2]. There were further complaints [2] that the contact problem was unsolved for the rigid manipulator, sensor, and environment.

In our early work [3], [4], we proposed mathematical models for industrial and legged walking robots in contact with environments. It was pointed out that robot joints suffered abrupt velocity changes at the moment of contact (or collision). In addition, impulsive forces were exerted at the contact point as well as at the robot joints. It is the large magnitude of the impulsive force that could damage the mechanical structure of the robotic system.

Recently, Wang and Mason studied the planar impact of two objects and developed simple graphical methods for predicting the mode of contact, which includes sliding, reverse sliding and sticking, the total effects of impulses, and the resultant motion of the objects [5]. In a more recent work, Mason and Wang further extended the result of [5] to include the impact modes of rigid rods [6]. Youcef-Toumi and Gutz studied the force control following an impact [7]. They suggested an integral force compensation with velocity feedback control, which improved force cracking as well as rejected impacts [7]. It was also revealed that impact response can be tuned by selecting a favorable dimensionless ratio of force to approach velocity [7].

In all the above mentioned works, two problems have never been studied. The first problem is the understanding of the impact effects on a robotic system with a force-sensing device being installed. The second one is how the force-sensing device responds to an impact, i.e., what kind of signal we can obtain from the device after an impact occurs. Those two problems are practically important and will be studied in this paper.

Force-sensing devices such as a wrist force sensor on industrial robots are commonly used to control the exerting force by the robot end effector. The contact of the end effector with an environment will bring the impact effects to the robot as well as to the force sensor. By understanding the relation between the effects and the robot motions, proper operation of the robot can be planned such that any destructive effects by the impact can be avoided.

1042-296X/91/0200-0156$01.00