[21] X. Wang and L. Chen, "Proving the uniform boundedness of some commonly used control scheme for robot," presented at IEEE Int. Conf. Robotics Automat., Scottsdale, AZ, May 1980.

Computing Location and Orientation of Polyhedral Surfaces Using a Laser-Based Vision System

Din-Chang Tseng and Zen Chen

Abstract—**A laser-based vision system for computing the location and orientation of 3-D polyhedral surfaces is proposed. In this system, an expanded laser beam passes through a code plate marked with equally spaced vertical and horizontal lines and impinges on a polyhedral object to create a spatial-encoded image for analysis. Then, based on the vanishing points or the directly available line directions of the perceived grid lines on the polyhedral surface, the polyhedral surface orientation can be inferred. In the meantime, the given dimensions of the grid pattern on the plate are used to estimate the depth information of the polyhedral surfaces. More importantly, we shall solve the noise problem that occurs in the real image by a least squares estimation method and an iterative refinement method based on a geometric constraint criterion. Experiments are conducted to provide practical insight into the method. The experimental results indicate that the method is remarkably accurate and stable.**

I. Introduction

Computer vision endows machines with a visual capability with applications in robot navigation, surface measurement, object modeling, camera position determination, automatic target recognition, etc. In the literature, a number of methods have been presented for measuring the 3-D orientation, location, and structure of objects using structured light. In [1] and [2], a vertical slit projector and a TV camera were employed to construct a range finder based on a triangulation technique to acquire range data for measuring polyhedral surfaces. Hall *et al.* [3] used a mask of a known form to project easily detected features on the surface of the object. The known form mask and the recorded image were then used as a stereo image pair. Then by using a least squares method, the three-dimensional coordinates of points on the object were calculated from the transformation matrices of the image pair. These two methods require the point correspondences to compute the 3-D coordinates of points. However, the correspondence information is generally not easy to get. Wang *et al.* [4] used grid coding to infer the surface orientation and the structure of visible object surfaces. They used the direction of the projected stripes to infer local surface orientation and did not require any correspondence relationship either the grid lines or the grid junctions. To simplify the mathematical derivation, they assumed a parallel projection model. However, based on this model, the absolute depth can not be determined. Hu and Stockman [5] also presented a method for 3-D surface measurement using a projected grid of light. Based on triangulation computation and some geometric and topological constraints on the grid pattern, the 3-D surface points were computed. Again, the point correspondence information was used in this method.

Manuscript received September 26, 1988; revised February 20, 1991.

D.C. Tseng is with the Department of Electronic Engineering, National Central University, Chung-Li, Taiwan, Republic of China.

Z. Chen is with the Institute of Computer Engineering, National Chiao Tung University, Hsinchu, Taiwan, Republic of China.

IEEE Log Number 9102767.

In this study, a new system for determining the 3-D location and orientation of polyhedral surfaces will be proposed. Compared to previous approaches, this system can find not only the orientation but also the depth parameter of polyhedral surfaces without using the point correspondences as required in the triangulation technique. Moreover, our method's computation model is nontrigonometric and rather simple. In this system a laser beam passes through a grid plate marked with equally spaced vertical and horizontal lines and impinges on a polyhedral object to create a spatial-encoded image for analysis. Based on the vanishing points or the directly available line directions of the perceived grid lines on the polyhedral surface, the polyhedral surface orientations can be inferred. In the meantime, the given dimensions of the grid pattern on the plate are used to estimate the depth information of the polyhedral surfaces. In order to deal with the noise in the real image, the extracted perceived grid pattern is first rectified by a least squares estimation method. Then an iterative method based on a geometric constraint criterion is employed to refine the polyhedral surface estimation. Experimental results indicate that the method is remarkably accurate and stable.

II. Geometric Configurations

A. Camera Geometry

Let the x_c, y_c, and z_c axes be the three principal axes of a camera geometry, and let the origin be at the camera lens center. Assume that the image plane is parallel to the $x_c y_c$ plane and its position is at $z_c = q$. The q value is roughly equal to the focal length for an object at a great distance. Assume that (x_i, y_i, z_i) is a 3-D point and (a_i, b_i) is its corresponding 2-D image point; then they are related by $a_i = qx_i / z_i$ and $b_i = qy_i / z_i$.

B. Laser Projector Geometry

A laser projector consisting of a laser source and a grid plate is used to generate two orthogonal sets of parallel sheets of light planes. The coordinate system of the laser projector is defined as follows. The two orthogonal grid lines on the plate are defined as the x_r and y_r axes, and the normal vector to the grid plate is defined as the z_r axis. The dimensions of the grid pattern on the plate are known. The relative orientations of the x_r, y_r, and z_r axes with respect to the camera coordinate system can be determined based on the same method, to be given later, that is used to determine the orientation of an unknown polyhedral surface. For this so-called laser calibration process, a planar surface with a known orientation (say, parallel to the grid plate) is used.

III. Grid Coding and Segmentation of Arbitrary Polyhedral Surfaces

A. Grid Coding of Polyhedral Surfaces

The perceived grid lines and grid points in the grid-coded polyhedron image need to be extracted in order to infer the 3-D location and orientation of the visible polyhedral surfaces. Since the perceived grid lines are the darkest pixels in the image, these lines can be extracted by a regular thresholding method. Also, a regular four-connected thinning algorithm can be applied to the thresholded grid lines to produce the lines. The following steps are used to obtain the 2-D coordinates of perceived grid points and a list of grid features to be used for polyhedral surface segmentation.

1) Grid Point Extraction: The thin grid lines obtained in the image are scanned row by row. The nearby branching pixels are detected at the place where two horizontal runs of grid pixels begin

1042-296X/91$01.00

to merge as a single run and at the place where a single run starts to split into two runs. The midpoint of the two branching pixels is identified as a coarse grid point and will be assigned a label.

2) Grid Line Tracking: Along each of the four grid line directions at each coarse grid point, the pixels belonging to a grid line are traced and linked based on the line slope.

3) Grid Point Refinement: Once all grid lines are traced, pixels on the same grid lines undergo a least squares line fitting. The intersection of fitted lines replace the coarse grid points as the final grid points.

4) Grid Features: Three grid features are derived at each grid point during the line tracking process. The feature list will be used in the polyhedral surface segmentation described later. It consists of

i) the neighbor identification *labels* of the four neighboring pixels at a grid point as represented by a vector (n_1, n_2, n_3, n_4) (if no such neighboring point exists, the corresponding label is set to zero),
ii) the *distance vector* (d_1, d_2, d_3, d_4) representing the distances between the current grid point and its four neighboring grid points (if no such neighboring grid points exists, the corresponding distance label is set to zero), and
iii) the *slope vector* (s_1, s_2, s_3, s_4) associated with the four line directions at the grid points.

B. Polyhedral Surface Segmentation by Clustering

Intuitively, we know that if the surface normal vectors to polyhedral surfaces are different, then the shapes of the perceived grid patterns on these polyhedral surfaces will be different. So the neighboring grid points with the same feature list will be clustered to signify a surface patch. Notice that two disconnected clusters of perceived grid patterns with the same feature list will correspond to two separate surfaces of identical surface orientation. Thus, the shape features and connectedness of grid lines serve as the bases for surface segmentation. That is, the three grid features mentioned above will be used.

Next, the clustering technique will be described. To generate a cluster, an arbitrary interior grid point (i.e., not at the border) is selected to be the seed of the cluster. Then the neighboring grid points are examined. If a neighboring point has a great similarity measure with reference to the current grid point, this point is then assigned to the same cluster. In this way, all connected interior grid points are grouped into the same cluster. If there are remaining interior grid points, then an arbitrary point is selected as the seed of a new cluster and the same process is repeated to construct another cluster. The overall image processing flow from image thresholding to surface segmentation of a polyhedron scene is given in Fig. 1.

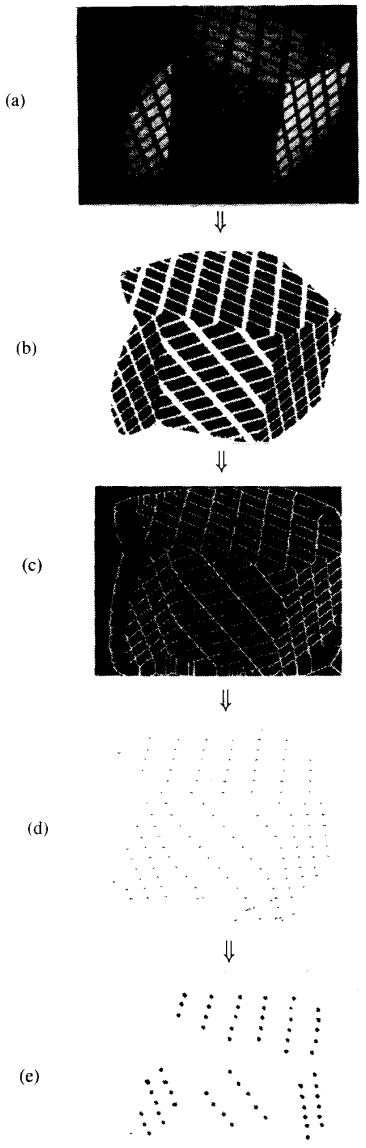

Fig. 1. The overall image processing flow of a grid-coded polyhedron scene. (a) A grid-coded image. (b) The bi-level thresholded image. (c) The thinned image. (d) The extracted grid points. (e) The clustered grid points.

IV. Three-Dimensional Location and Orientation Determination of Arbitrary Polyhedral Surfaces

A. Rectification of Grid Pattern

The image formation of the perceived grid pattern can be viewed as the consequence of a sequence of geometric projection transformations applied to the grid pattern on the grid plate. Namely, it involves a parallel projection transformation from the grid pattern on the plate to the grid pattern on the polyhedral surface, a perspective transformation from the grid pattern on the polyhedral surface to the projected grid pattern on the image plane, and possible rotation and translation of the projected grid pattern to the actual perceived image. Such a sequence of transformations can be represented by a resultant transformation matrix in the homogeneous coordinate system, i.e.,

$$[\text{perceived grid pattern}] = [\text{grid plate pattern}]\,[\text{transformation matrix}].$$

In general, a 4×3 transformation matrix is used to describe the relation between a set of noncoplanar points in 3-D space and its corresponding image points [6]. However, grid points on the grid plate are coplanar, and the z coordinates of all these grid points will be set to zero, so a 3×3 submatrix of the 4×3 transformation matrix is used for analysis. This is given as

$$[wx_{pi}, wy_{pi}, w] = [x_{oi}, y_{oi}, 1]\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (1)$$

where (x_{pi}, y_{pi}) is the 2-D coordinates of a perceived grid point, (x_{oi}, y_{oi}) is the 2-D coordinates of a grid point on the grid plate, $[A_{ij}]$ is the 3×3 transformation matrix, and w is a real constant. For $i = 1, 2, \cdots, n$, the combination of equations can be rewritten in a matrix form as

$$\begin{bmatrix} x_{01} & y_{01} & 1 & 0 & 0 & 0 & -x_{01}x_{p1} & -y_{01}x_{p1} \\ 0 & 0 & 0 & x_{01} & y_{01} & 1 & -x_{01}y_{p1} & -y_{01}y_{p1} \\ & & & & & \vdots & & \\ x_{on} & y_{on} & 1 & 0 & 0 & 0 & -x_{on}x_{pn} & -y_{on}x_{pn} \\ 0 & 0 & 0 & x_{on} & y_{on} & 1 & -x_{on}y_{pn} & -y_{on}y_{pn} \end{bmatrix} \cdot \begin{bmatrix} B_{11} \\ B_{21} \\ \vdots \\ B_{13} \\ B_{23} \end{bmatrix} = \begin{bmatrix} x_{p1} \\ y_{p1} \\ \vdots \\ x_{pn} \\ y_{pn} \end{bmatrix}$$

or

$$\boldsymbol{QB} = \boldsymbol{P} \tag{2}$$

where $B_{ij} = A_{ij}/A_{33}$, $1 \le i \le 3$, $1 \le j \le 3$.

Then the least squares estimation of $\boldsymbol{B}$ using a pseudoinverse matrix can be obtained as

$$\boldsymbol{B} = (\boldsymbol{Q}^t\boldsymbol{Q})^{-1}\boldsymbol{Q}^t\boldsymbol{P}. \tag{3}$$

Thus

$$[wx'_{pi}, wy'_{pi}, w] = [x_{oi}, y_{oi}, 1] \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \tag{4}$$

where B_{33} is set to 1. Here, (x'_{pi}, y'_{pi}) is the new rectified value of (x_{pi}, y_{pi}) based on the above least squares estimation.

In the above computation, the relative coordinate values of points (x_{oi}, y_{oi}) of a grid line on the grid plate corresponding to the points (x_{pi}, y_{pi}) of a perceived grid line are specified according to the relative positions on the grid plate. Here, exact point correspondences are not required. Any translation or any rotation of a multiple of 180° should be included in transformation matrix A. One only needs to check if the set of perceived grid lines is correspondent to the right set of grid lines on the grid plate. This can be done through the determination of the polyhedral grid line direction based on the set of perceived grid lines, which will be introduced next.

B. Deriving the Initial Surface Equations

1) Deriving the Polyhedral Grid Line Direction: It is well known that the perspective projection of parallel lines that are not parallel to the image plane will converge at a vanishing point [7], [8]. Therefore, the lines projected from the parallel grid lines onto the polyhedral surface will intersect at a vanishing point assuming the surface is not parallel to the image plane. Due to camera and image-processing errors, the lines projected from the parallel grid lines may not intersect at a point. However, the perceived grid lines rectified by a least squares estimation technique will converge to the vanishing point.

Assume the parametric equation of a grid line on a polyhedral surface is given by

$$\begin{aligned} Ax(t) &= x_0 + at \\ y(t) &= y_0 + bt \qquad -\infty \le t \le \infty \\ z(t) &= z_0 + ct \end{aligned} \tag{5}$$

which passes through point (x_0, y_0, z_0) and has a direction specified by the vector $[a, b, c]$. Let $(x(t), y(t), z(t))$ be a point on this polyhedral grid line and $(u(t), v(t))$ be the corresponding image point on the image plane; then

$$u(t) = \frac{qx(t)}{z(t)} = \frac{q(x_0 + at)}{(z_0 + ct)}$$

$$v(t) = \frac{qy(t)}{z(t)} = \frac{q(y_0 + bt)}{(z_0 + ct)} \tag{6}$$

where q is the distance from the camera lens center to the image plane. The vanishing point (u_0, v_0) is obtained as $(u(t), v(t))$ when $t \to \infty$. That is

$$u_0 = \lim_{t\to\infty} u(t) = \frac{qa}{c}$$

$$v_0 = \lim_{t\to\infty} v(t) = \frac{qb}{c}. \tag{7}$$

It implies that the line directions $[a, b, c]$ and $[u_0, v_0, q]$ are equivalent. In case the grid line on the polyhedral surface is parallel to the image plane (i.e., $c = 0$), the grid line direction can be directly determined as the direction of the perceived grid line on the image plane without the need to find the vanishing point.

2) Deriving the Normal Vector of a Polyhedral Surface: Assume two vanishing points (u_1, v_1) and (u_2, v_2) are found for a polyhedral surface. Then the surface normal vector of the planar surface is defined by the cross product of the two line direction vectors $[u_1, v_1, q] \times [u_2, v_2, q]$. Let this unit normal vector be denoted by $\vec{N} = [A, B, C]$; then the plane equation is given by

$$Ax + By + Cz = D \tag{8}$$

where D is the depth parameter that specifies the distance from the origin to the given plane. One can use the procedure presented here to compute the orientations of the two sets of laser planes (i.e., the laser calibration) by placing a planar object surface with a known orientation to the laser light; an object surface parallel to the grid plate is used for this purpose.

3) Deriving the Depth Parameter D of a Polyhedral Surface: Assume the directions of the x_r and y_r axes of the laser coordinate system relative to the camera coordinate system are given by $\vec{L}_x$ and $\vec{L}_y$, respectively.

In Fig. 2, let the grid line $\overline{g_1g_2}$ be the intersection between the polyhedral surface $Ax + By + Cz = D$ and one of the y_rz_r laser plane, i.e., one with the normal vector $\vec{L}_x$. Assume $\overline{g_1g_2}$ is bounded by two x_rz_r laser planes with a space of k grids, and each grid on the grid plate is length l. Then the angle θ between line $\overline{g_1g_2}$ and the normal vector $\vec{L}_y$ of the x_rz_r laser plane can be computed as

$$\theta = \cos^{-1}\left[(\vec{N} \times \vec{L}_x) \cdot \vec{L}_y / |\vec{N} \times \vec{L}_x \| \vec{L}_y|\right]. \tag{9}$$

Let D_1 be the length of $\overline{g_1g_2}$; then

$$D_1 = kl/|\cos\theta|.$$

Assume (x_1, y_1, z_1) and (x_2, y_2, z_2) are two coordinates of the grid points g_1 and g_2, and (a_1, b_1) and (a_2, b_2) are their two corresponding 2-D coordinates on the image plane. It then can be shown that

$$D = \pm D_1 \Bigg/ \sqrt{\left(\frac{a_2}{qp_2} - \frac{a_1}{qp_1}\right)^2 + \left(\frac{b_2}{qp_2} - \frac{b_1}{qp_1}\right)^2 + \left(\frac{1}{p_2} - \frac{1}{p_1}\right)^2} \tag{10}$$

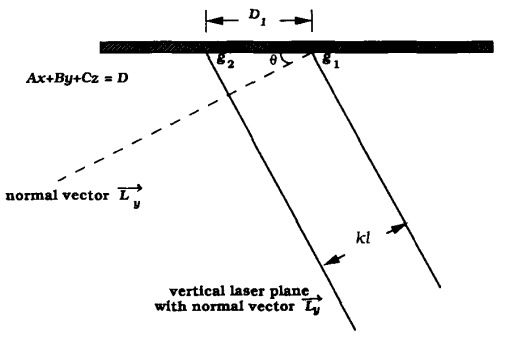

Fig. 2. The configuration of the laser planes of light impinging on the polyhedral surface. In this view, the direction of $\vec{L}_x$ is perpendicular to the page.

where $p_i = (a_i A/q) + (b_i B/q) + C$, $i = 1, 2$. Since z_1 and z_2 are positive, the sign of D depends on p_i, $i = 1$ or 2. That is, D and $p_i (i = 1$ or 2) have the same sign.

V. Iterative Refinement Based on a Geometric Constraint Criterion

A. Principle

The above surface determination is based on the perceived grid points rectified by the least squares estimation; thus, the noise in the original image points is reduced. However, the residual error may still cause some errors in the transformation matrix estimation. The obtained matrix may not fully fulfill the geometric projection constraints. Therefore, we use the geometric constraints to iteratively modify the solution obtained above.

In the iterative modification, the 3-D grid points g_1 and g_2 mentioned previously are selected to be two points on a polyhedral grid line coplanar with the y_r axis. Similarly, the grid points on a polyhedral grid line coplanar with the x_r axis are also selected. From these four points, the grid pattern on the polyhedral surface can be constructed by interpolation as well as extrapolation. Then the constructed grid pattern is reprojected via the precise perspective transformation matrix onto the image plane to compare the projected grid points with the original perceived grid points to compute a sum of distance errors to evaluate the accuracy of the polyhedral surface estimation.

Assume (x_{pj}, y_{pj}), $j = 1, 2, \cdots, n$, are n original grid points on the perceived grid pattern, and (x_{oj}, y_{oj}), $j = 1, 2, \cdots, n$ are the n corresponding grid points obtained from the reprojection of the 3-D grid points on the estimated polyhedral surface. Let

$$S = \sum_{j=1}^{n} \sqrt{(x_{pj} - x_{oj})^2 + (y_{pj} - y_{oj})^2} \qquad (11)$$

be the sum of distance errors between these two sets of grid points. Each estimation of the polyhedral surface will be associated with such an error. The optimal polyhedral surface is defined as the one associated with a minimal error.

B. Iterative Modification of Plane Equation Based on the Geometric Constraint Satisfaction

The normal vector of the estimated polyhedral surface can be reduced to a vector containing only two variables. That is, without loss of generality, let $|A| =$ maximum $(|A|, |B|, |C|)$; then $[A, B, C] = |A| \cdot [\text{sign}(A) \cdot 1, B', C']$. Thus, $[A, B, C]$ is reduced to $[\text{sign}(A) \cdot 1, B', C']$, which contains two variables B' and C'.

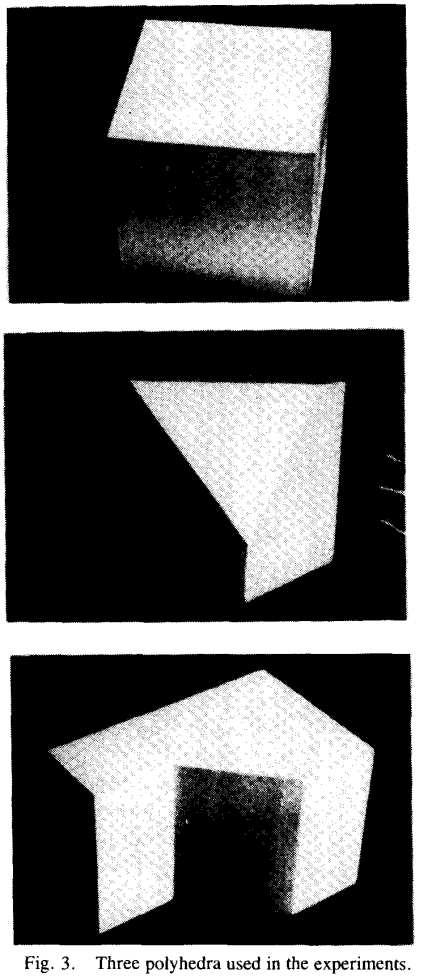

Fig. 3. Three polyhedra used in the experiments.

Now we can modify the normal vector by changing the values of B' and C' in a 2-D space based on the sum of distance errors defined in (11). The initial solution of the normal vector is the one obtained based on the grid points rectified by the least squares estimation. Assume the starting location of the 2-D point (B', C') is associated with an error SS_0. We modify (B', C') by adding to or subtracting from B' or C' a value Δd (i.e., change (B', C') to $(B' + \Delta d, C')$, $(B' - \Delta d, C')$, $(B', C' + \Delta d)$ and $(B', C' - \Delta d)$) where Δd is the step size of the modification. Find the errors for these new locations, called SS_1, SS_2, SS_3, and SS_4. In the set of $\{SS_i, i = 0, 1, 2, 3, 4\}$, if SS_0 is the smallest, then reduce Δd to $\Delta d/2$; otherwise, move (B', C') to the location with the smallest error SS_i, and set SS_0 to this smallest error; repeat the modification process until the error is smaller than a threshold or no significant improvement is made. To ensure a global minimum solution, a coarse search is applied first to a larger region around the initial position of (B', C').

VI. Experiments and Discussions

In the experiments a CCD camera with a fixed focal length lens of 25 mm was used and the pictures taken were all in a 256×240 array with 256 intensity levels. The polyhedra used are shown in Fig. 3.

TABLE I
THE SUMMARIZED ESTIMATION ERRORS OF 3-D POLYHEDRAL SURFACE DETERMINATION IN 26 EXPERIMENTS

Distance error (percentage)			Orientation error (degree)		
mean	minimum	maximum	mean	minimum	maximum
1.2%	0%	2.9%	0.67°	0.06°	1.7°

TABLE II
SOME RESULTS OF THE POLYHEDRAL SURFACE ESTIMATION

Item	Polyhedron ID	g11	g14	g17	g1902
Surface Rotation angle	measured	0°	30°	60°	80°
	computed	1.7°	30.3°	58.9°	80.1°
Distance from a point on polyhedral surface to camera lens center[1]	measured	590	590	588	566
	computed	600	587	577	555
Coefficients of polyhedral plane equation (A, B, C, D)	computed	−0.702	−0.276	0.218	0.582
		0.086	0.101	0.093	0.017
		0.707	0.956	0.972	0.813
		424	557	558	450
Angle between two intersecting grid lines on polyhedral surface	measured	90°	81.5°	53°	22.5°
	computed	91.4°	81.6°	53.2°	20.7°
Lengths of two intersecting grid lines on polyhedral surface[1]	measured	10	11	16	40
		10	11	16	40
	computed	10	10.77	15.39	39.9
		10.18	10.79	15.2	39
Sum of Distance errors/ (number of grid points)[2]	computed	84/(72)	21/(48)	31/(26)	3.3/(7)

[1] All distances and lengths are in millimeters.
[2] The sum of distance errors is in pixels.

TABLE III
SOME RESULTS FOR DIFFERENT ORIENTATION CONFIGURATIONS OF LASER, CAMERA, AND POLYHEDRAL SURFACE

Item	Polyhedron ID	g57	g51	g53	g55
Measured camera rotation angle[1]		10.4°	28.9°	51.7°	68.8°
Distance from a point on polyhedral surface to camera lens center	measured	679	624	604	538
	computed	683	624	603	537
Coefficients of polyhedral plane equation (A, B, C, D)		−0.208	−0.504	−0.795	−0.937
		0.023	0.020	0.038	0.032
		0.978	0.864	0.606	0.347
		665.9	546.7	381.6	184.2
Angle between two intersecting grid lines on polyhedral surface		89.6°	90.4°	90.6°	92.7°
Lengths of two intersecting grid lines on polyhedral surface		10.01	10.00	10.00	10.02
		10.01	9.84	10.08	10.26
Sum of distance errors/(number of grid points)		17.7/(68)	50.9/(63)	22/(65)	63/(39)

[1] The camera rotation angle means the angle between two consecutive setups of the camera when the camera is moving around the polyhedon.

TABLE IV
EFFECTS OF SELECTING DIFFERENT SET OF GRID POINTS ON THE ACCURACY OF POLYHEDRAL SURFACE DETERMINATION

Item	Polyhedron ID	g57	g51	g53	g55
Measured camera rotation angle[1]		10.2°	30.5°	50.9°	69.4°
Distance from a point on polyhedral surface to camera lens center	measured	679	624	604	538
	computed	680	616	601	534
Coefficients of polyhedral plane equation (A, B, C, D)		−0.203	−0.530	−0.789	−0.943
		0.031	0.016	0.039	0.030
		0.979	0.848	0.613	0.333
		665.0	534.3	387.6	172.0
Angle between two intersecting grid lines on polyhedral surface		89.9°	89.3°	90.8°	89.1°
Lengths of two intersecting grid lines on polyhedral surface		10.00	10.00	10.00	10.03
		10.00	9.8	10.00	10.35
Sum of distance errors/(number of grid points)		7.4/(35)	12.5/(20)	5.5/(22)	11.8/(15)

[1] The camera rotation angle means the angle between two consecutive setups of the camera when the camera is moving around the polyhedon.

A. Accuracy of the Polyhedral Surface Estimation

Twenty-six experiments were conducted to demonstrate the accuracy of the polyhedral surface estimation. The results are summarized in Table I; part of the detailed results are given in Tables II and III. For a more detailed report, refer to [9]. In Table I, the distance is defined from a preselected grid point on the polyhedral surface to the camera lens center, and the surface orientation means either the relative angle of the polyhedral surface with respect to a reference direction or the angle between two adjacent polyhedral surfaces, depending on which is appropriate. Twenty-six experiments were analyzed for distance accuracy, and 19 experiments were analyzed for orientation accuracy. The estimation errors are generally small; the errors range from 0 to 2.9% with a mean 1.2% for the distance estimation and range from 0.06° to 1.7° with a mean 0.67% for the orientation estimation.

B. Influence of Different Orientation Configurations of Laser, Camera, and Polyhedral Surface

Here, the influence of different orientation configurations of laser, camera, and polyhedral surface over the accuracy of surface determination is examined. The results are shown in Tables II and III. The image data used in Table II were taken by rotating the polyhedral surface around the laser optical axis from 0° to 80° with the laser and camera being fixed. The image data used in Table III were taken by moving the camera around the polyhedral surface from 11° to 70° with the laser and the polyhedral surface fixed and the grid light perpendicular to the polyhedral surface.

From the estimation errors listed in these two tables, we can see that the different orientations of laser, camera, and polyhedral surface do not cause any serious problem. The estimation errors for the cases with a larger camera-to-polyhedron angle in Table II (such as case $g1902$) are due to the fact that the projected grid lines become larger in size and thicker in width. In these cases, the coordinates of the grid points become slightly inaccurate. However, under such situations a small grid plate can be used instead to prevent the grid lines from becoming sparse and thick.

C. Effect of Selecting Different Sets of Grid Points

On the other hand, we repeated the experiments reported in Table III by selecting different numbers of grid points. The grid points near the boundary of the projected grid pattern were selected on purpose. The results are shown in Table IV. The experiments indicate that rather consistent results are obtained regardless of the set of grid points selected.

VII. CONCLUSION

A structured light vision system for computing 3-D location and orientation of arbitrary polyhedral surfaces is proposed. In this system a laser projector was constructed to encode the scene containing a polyhedral object to create a grid-coded image for analysis. A clustering technique was developed to segment grid-coded polyhedral surfaces. In order to deal with the errors caused by the laser projector, camera lens distortion, and image processing, the perceived grid pattern was first rectified by a least squares model. Based on the vanishing points of one planar surface of the polyhedron and the physical dimensions of the grid plate, the plane equation then was estimated. At last, the iterative method based on a geometric constraint criterion was employed to refine the plane equation. The experiments indicate that the results are quite accurate and stable for various orientation configurations of laser, camera, and polyhedral surface. We are currently applying the same equipment to measuring a cylindrical object surface. A 3-D object recognition system based on the proposed vision system is also under investigation.

REFERENCES

[1] Y. Shirai, "Recognition of polyhedrons with a range finder," *Pattern Recognition*, vol. 4, pp. 243–250, 1972.
[2] M. Oshima and Y. Shirai, "Object recognition using three-dimensional information," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, no. 4, pp. 353–361, July 1983.
[3] E. L. Hall, J. B. K. Tio, C. A. McPherson, and F. A. Sadjadi, "Measuring curved surfaces for robot vision," *IEEE Comput.*, vol. C-15, no. 12, pp. 42–54, Dec. 1982.

[4] Y. F. Wang, A. Mitiche, and J. K. Aggarwal, "Computation of surface orientation and structure of objects using grid coding," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, no. 1, pp. 129-137, Jan. 1987.
[5] G. Hu and G. C. Stockman, "3-D surface solution using structured light and constraint propagation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 4, pp. 390-402, 1989.
[6] D. H. Ballard and C. M. Brown, *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
[7] J. D. Foley and A. Van Dam, *Fundamentals of Interactive Computer Graphics*. Reading, MA: Addison-Wesley, 1982.
[8] Z. Chen, D. C. Tseng, and J. Y. Lin, "A simple vision algorithm for 3-D position determination using a single calibration object," *Pattern Recognition*, vol. 22, no. 2, pp. 173-187, 1989.
[9] D. C. Tseng, "Monocular computer vision for 3-D position determination in a polyhedral scene," Ph.D. dissertation, National Chiao Tung Univ., Hsinchu, Taiwan, R.O.C., 1988.

Motion Estimation under Orthographic Projection

Xiaoping Hu and Narendra Ahuja

Abstract—**In this short paper, we present some new results for the problem of motion estimation under orthographic projection. We refine some basic results obtained by previous researchers and provide more detailed and precise results. We show that, in the two-view problem, when the rotation is around the optical axis, the motion (but not the structure) is uniquely determined. We show that, in the three-view problem, only under certain conditions are the motion and structure uniquely determined. We show that, for any motion problem, if two-view matching cannot determine the motion, only under certain conditions can three- or multiview matching help.**

Index Terms—**Depth, motion, motion estimation, orthography, rigidity, structure.**

I. Introduction

In the literature, two projection models of image formation have been widely used: perspective projection and orthographic projection. The motion estimation problem has been investigated mainly for perspective projection [9]-[18] with some work on orthographic projection [1]-[7]. Ullman [2] started the research on the motion problem with orthographic projection. But in later work, the primary interest of motion researchers has been in perspective projection. This is probably due to the fact that perspective projection models the imaging process of ordinary cameras more accurately and is better conditioned in the sense of determinedness from correspondence data. But, when a long-focus telephoto lens is used, the imaging process can be approximated by orthographic projection if the motion and the object size in the direction of the optical axis are negligible compared with the object distance, although a scale constant may be involved [4]. In medical imaging such as X-ray imaging, the imaging process can be considered as an orthographic projection. Therefore, it is necessary to investigate the motion problem under orthographic projection. Another projection model that lies between the perspective and orthographic projections also has been investigated and is called *paraperspective projection* [8]. In this work, we will discuss the motion estimation problem under orthographic projection only.

This short paper concerns conditions under which motion is uniquely determined from only monocular image point correspondence data. For the two-view problem, determinedness means that the rotation matrix $\mathbf{R}$ is uniquely determined and the translation vector $\mathbf{T} = [t_1\ t_2\ t_3]^T$ is determined up to a scale and a constant, i.e., $[t_1\ t_2]^T$ is determined up to a scale, and t_3 is not determined. For the three-view problem, determinedness means that all rotation matrices are uniquely determined, the translation vector between the second and the third views is determined to a scale, and the translation vector between the first and the second views is determined to a scale and a constant (similar to the two-view case).

Ullman [2] showed that the two-view motion problem is generally not determined, but the three-view motion problem can generally be determined with four correspondences of projections of noncoplanar space points, and he proposed a nonlinear algorithm for motion estimation from three-view matching. Later, Aloimonos and Brown [3] showed that the two orthographic views of four noncoplanar points admit only four interpretations of the structure of the four points and that it is possible to uniquely recover structure from three orthographic views of three points in space, contradicting Ullman's results. Huang and Lee [1] proposed a linear algorithm for three-view motion estimation. They gave formal proof that the two-view motion problem is generally not determined and the three-view motion problem is generally determined.

In this short paper, we will reexamine some of the problematic results obtained in the above referenced papers. We concern motion estimation only and do not discuss structure estimation. We show that, for monocular vision, the two-view motion is determined if and only if the rotation is around the optical axis, and three-view motion is determined if and only if certain necessary and sufficient conditions are satisfied. We show that, given a sequence of images under orthographic projection, only under certain conditions can the motion between each pair of views be determined by multiview matching. These results contrast those obtained by Ullman [2] and Huang and Lee [1].

In Section II, we present some preliminary results for motion estimation. In Section III, we investigate the two-view motion problem. We show that rotation is uniquely determined if and only if it is around the optical axis. In Section IV, we reexamine Huang and Lee's three-view algorithm and show that the three-view motion problem is determined only under certain conditions. Section V summarizes the paper.

II. Representation of Two-View Motion

We use $x - y$ to denote image coordinates and $X - Y - Z$ to denote real-world coordinates. An image point (x, y) represents the projection of a scene point $\mathbf{X} = (X, Y, Z)$. For orthographic projection, we have

$$x = X$$
$$y = Y. \tag{1}$$

Throughout this work, we will use the following notation. Bold capital letters represent vectors or matrices, italic capital letters coordinates in the space, italic lowercase letters coordinates in the image plane or elements of vectors or matrices. (x, y), (x_i, y_i), and (x_i', y_i') correspond to the projections of $\mathbf{X} = (X, Y, Z)$, $\mathbf{X}_i = (X_i, Y_i, Z_i)$, and $\mathbf{X}_i' = (X_i', Y_i', Z_i')$, respectively. $\mathbf{\Theta}$, $\mathbf{\Theta}_i$, and $\mathbf{\Theta}_i'$ denote $[x, y]^T$, $[x_i, y_i]^T$, and $[x_i', y_i']^T$, respectively.

Manuscript received March 13, 1991; revised July 30, 1991. This work was supported by the U.S. Army Advanced Construction Technology Center under Grant DAAL03-87-K-0006.

The authors are with the Beckman Institute and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801.

IEEE Log Number 9103509.