Systems and Computers in Japan, Vol. 21, No. 5, 1990
Translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol. 72-DII, No. 4, April 1989, pp. 517-525

3D Recovery of Polyhedra by Parallelism Heuristics

Toshie Tanaka, Nonmember
Mitsubishi Electric Corporation, Tokyo, Japan 100

Takao Kawashima, Nonmember, and Ken-ichi Kanatani, Member
Department of Computer Science, Gunma University, Kiryu, Japan 376

SUMMARY

A method is presented which recovers the 3-D shape of a polyhedron under the assumption of a large number of parallel edges. The method is based on the fact that three or more parallel edges intersect at a point (vanishing point) when extended on the image plane. To determine a pair of two parallel edges, a heuristic method based on the structure of polyhedra is employed. To cope with the error in the image, a threshold processing based on the causes of error is proposed.

When parallel edges are found, the 3-D orientation of each edge can be determined from the vanishing point. However, contradictions may arise in the calculated 3-D orientations of the edges due to errors. To overcome such a problem, an optimization technique based on the constraint on polyhedra is proposed. Then our method is applied to an actual image.

1. Introduction

In viewing a 2-D image, a human can perceive a 3-D shape drawn on the image. One approach to simulate this behavior by computer is the automated interpretation of a line figure for polyhedra [1, 15]. Even though this approach can label the edges as convex or concave, or determine whether or not there exist corresponding 3-D objects, the 3-D objects cannot be reconstructed uniquely. To do this, there must be strong constraints.

Considering that humans can do this easily, many studies have been made to incorporate various heuristic hypotheses in relation to cognitive psychology. Typical hypotheses are the orthogonality hypothesis, which assumes the orthogonality of edges [2, 5, 6, 9, 10], the skew-symmetry hypothesis concerning shapes of faces [5] and the closed contour hypothesis concerning closed curves [3]. This paper attempts a 3-D recovery using the parallelism hypothesis. This assumption is reasonable since many man-made objects are composed of parallel edges.

In this paper, we assume that necessary image data have already been obtained by image processing stages, and focus on a mathematical method which can recover the 3-D shape of the object uniquely from a single image. When parallel edges in a scene are projected onto the image plane, the projected edges intersect at a single point. This intersection is called the vanishing point.

The basic principle of 3-D recovery in this paper is based on the well-known fact that the position of the vanishing point determines the 3-D orientation of the edges [13, 15]. However, the vanishing point is only a partial information. If the image contains errors, a consistent polyhedron cannot always be recovered. To circumvent this, we present an optimization technique based on the constraint on polyhedra and recover a 3-D consistent shape uniquely.

2. Vanishing Point and 3-D Direction of Edge

Consider an XYZ coordinate system with the Z axis as the optical axis of the camera. Let XY plane be the image plane. Letting point $(0, 0, -f)$ on the Z axis be the viewpoint, point (X, Y, Z) in the scene is projected perspectively onto the point (x, y) on the image plane, where (Fig. 1)

$$x=\frac{fX}{f+Z}, \quad y=\frac{fY}{f+Z} \tag{1}$$

ISSN0882-1666/90/0005-0098$7.50/0

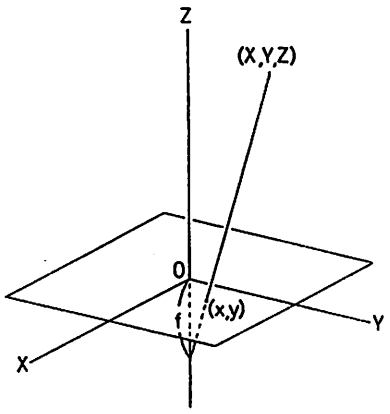


Fig. 1. Perspective projection onto the image plane.

Theorem 1. The vanishing point of a line with 3-D direction $m = (m_1, m_2, m_3)$ is given by $(fm_1/m_3, fm_2/m_3)$.

Proof. The line passing through point (X_0, Y_0, Z_0) with direction $m = (m_1, m_2, m_3)$ is given by

$$X = X_0 + tm_1, \quad Y = Y_0 + tm_2, \quad Z = Z_0 + tm_3 \quad (t: \text{ real}) \tag{2}$$

By Eq. (1), this line is projected onto the image plane as follows:

$$x = \frac{fX}{f+Z} = f\frac{X_0 + tm_1}{f + Z_0 + tm_3}, \qquad y = \frac{fY}{f+Z} = f\frac{Y_0 + tm_2}{f + Z_0 + tm_3} \tag{3}$$

Letting $t \to \infty$, the vanishing point (a, b) of this line is obtained as follows:

$$a = fm_1/m_3, \quad b = fm_2/m_3 \tag{4}$$

Corollary 1. The 3-D orientation of the line with vanishing point (a, b) on the image plane is given by the following unit vector:

$$m = \pm\left(\frac{a}{\sqrt{a^2+b^2+f^2}}, \frac{b}{\sqrt{a^2+b^2+f^2}}, \frac{f}{\sqrt{a^2+b^2+f^2}}\right) \tag{5}$$

Consequently, when parallel edges are found on the image plane, their 3-D orientation can be determined by detecting their vanishing point (Appendix 1). The algorithm for finding parallel edges is described in Sect. 5. However, there exists a serious problem: Even if 3-D edge orientations are determined, the 3-D shape of the object cannot necessarily be recovered.

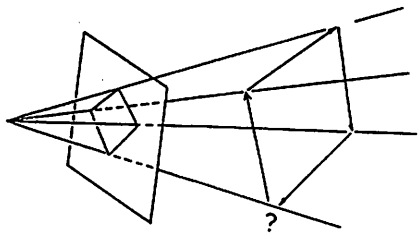


Fig. 2. Incompatibility of edge adjacency.

Theoretically, it suffices to place edges in the scene according to the computed orientations and the projection relations. However, errors of image processing are inevitable. It is not always true, for example, that the edges forming a boundary of a face close themselves (Fig. 2). In view of this, unique recovery of consistent 3-D shape must incorporate constraints on polyhedra [7, 8, 17, 18].

3. Constraint on Polyhedra

Let the coordinates of vertex V_i, $i = 1, \ldots, n$ in the scene be (X_i, Y_i, Z_i). Let the equation of face F_α, $\alpha = 1, \ldots, m$ be $Z = p_\alpha X + q_\alpha Y + r_\alpha$. Here, (p_α, q_α) is the gradient of face F_α, and r_α is the distance of face F_α from the image plane measured along the Z axis. When vertex V_i is on face F_α, we say that vertex V_i is incident to face F_α. The relation is defined by the incidence pair (F_α, V_i) [16, 18]. Let the number of incidence pairs be L. The condition (F_α, V_i) is expressed as follows:

$$Z_i = p_a X_i + q_a Y_i + r_a \tag{6}$$

We introduce variables x_i, y_i, z_i as follows:

$$x_i = \frac{fX_i}{f+Z_i}, \quad y_i = \frac{fY_i}{f+Z_i}, \quad z_i = \frac{fZ_i}{f+Z_i} \tag{7}$$

Then by Eq. (1), (x_i, y_i) are exactly the image coordinates of vertex V_i. The inverse relation is written as follows:

$$X_i=\frac{fx_i}{f-z_i},\quad Y_i=\frac{fy_i}{f-z_i},\quad Z_i=\frac{fz_i}{f-z_i} \tag{8}$$

Substituting this into Eq. (6), we obtain

$$z_i=\frac{fp_\alpha}{f+r_\alpha}x_i+\frac{fq_\alpha}{f+r_\alpha}y_i+\frac{fr_\alpha}{f+r_\alpha} \tag{9}$$

Let

$$P_\alpha\equiv\frac{fp_\alpha}{f+r_\alpha},\quad Q_\alpha\equiv\frac{fq_\alpha}{f+r_\alpha},\quad R_\alpha\equiv\frac{r_\alpha}{f+r_\alpha} \tag{10}$$

Then $(P_\alpha, Q_\alpha, R_\alpha)$ are considered to be the parameters specifying face F_α in the plane of p_α, q_α, r_α. The inverse relation is written as

$$p_\alpha=\frac{P_\alpha}{1-R_\alpha},\quad q_\alpha=\frac{Q_\alpha}{1-R_\alpha},\quad r_\alpha=\frac{fR_\alpha}{1-R_\alpha} \tag{11}$$

Equation (9) is written as follows:

$$z_i=P_\alpha x_i+Q_\alpha y_i+fR_\alpha \tag{12}$$

Since the image coordinates (x_i, y_i) are known, the scene coordinates of vertex V_i can be determined from Eq. (8) if z_i is known. Consequently, z_i can be used as the parameter specifying the space coordinate of vertex V_i in the place of Z_i. Note that Eq. (12) is linear with respect to the unknowns z_i, P_α, Q_α, R_α. So that 3-D recovery is possible, the incidence structure must be regular [16, 18]. If the structure is not regular, it can be made regular [17, 18]. In the following, the structure is always assumed to be regular.

3. 3-D Recovery by Optimization

Assume that parallel edges have been found (the algorithm is shown in Sect. 5), their vanishing points have been detected (the algorithm is shown in Appendix 1), and 3-D orientations of edges have been determined. Let the edges on face F_α for which 3-D orientations are determined be $e_{k\alpha}$, $k_\alpha = 1, \ldots, N_\alpha$. Let the unit vectors representing their 3-D orientations be $n_{k\alpha}=(n_{k\alpha(1)}, n_{k\alpha(2)}, n_{k\alpha(3)})$.

Let the gradient of face F_α be (p_α, q_α). Then the unit surface normal to face F_α is given by surface $(p_\alpha, q_\alpha, -1)/\sqrt{p_\alpha^2+q_\alpha^2+1}$. All unit vectors $n_{k\alpha}, k_\alpha=1, \cdots, N_\alpha$ should be orthogonal to this unit surface normal. In view of this, it makes sense to determine (p_α, q_α) in such a way that the (half) squared sum of scalar products

$$\frac{1}{2\sqrt{p_\alpha^2+q_\alpha^2+1}}\sum_{k\alpha=1}^{N\alpha}(n_{k\alpha(1)}p_\alpha+n_{k\alpha(2)}q_\alpha-n_{k\alpha(3)})^2 \tag{13}$$

is minimized.

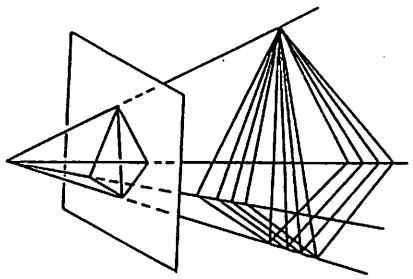


Fig. 3. One solution is chosen from among infinitely many consistent polyhedron solutions by optimization.

Hence, we multiply the forementioned expression by weight W_α of face F_α and sum it for all the faces, and minimize the resulting expression. Then the minimization for all the faces can be executed in one stage. The expression to be minimized is obtained after substituting Eq. (11) and rewriting p_α, q_α by P_α, Q_α, R_α:

$$J=\frac{1}{2}\sum_{\alpha=1}^{m}\frac{W_\alpha}{\sqrt{p_\alpha^2+q_\alpha^2+1}}\left(\frac{f+r_\alpha}{f}\right)^2 \times\sum_{k\alpha=1}^{N\alpha}(n_{k\alpha(1)}P_\alpha+n_{k\alpha(2)}Q_\alpha+n_{k\alpha(3)}R_\alpha-n_{k\alpha(3)})^2 \tag{14}$$

Estimation of vanishing points becomes inaccurate as the distance r_α along the Z axis becomes large or when the gradient of the face is small. Consequently, the weight W_α is defined as follows:

$$W_\alpha=\sqrt{p_\alpha^2+q_\alpha^2+1}\left(\frac{f}{f+r_\alpha}\right)^2 \tag{15}$$

Then the 3-D shape is recovered by minimizing Eq. (14) under the constraint (12) for all incidence pairs (F_α, V_i) (Fig. 3). However, since the recovery is made from a single image with gradient cues of the faces, the absolute distance cannot be determined. Consequently, the Z coordinate is assumed for one of the vertices. Let that vertex be V_n.

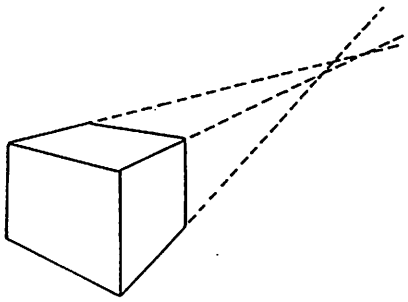


Fig. 4. In the presence of noise, parallel edges, when extended, may not necessarily intersect at a single vanishing point.

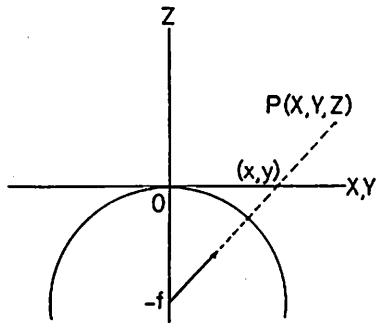


Fig. 5. The image sphere of radius f centered at the viewpoint.

If Lagrange multipliers $\Lambda_{\alpha i}$ for incidence pairs (F_α, V_i), the problem is to minimize

$$\tilde{J}=\frac{1}{2}\sum_{a=1}^{m}\sum_{a=1}^{Na}(n_{ka(1)}P_a+n_{ka(2)}Q_a+n_{ka(3)}R_a-n_{ka(3)})^2 + \sum_{a,i}^{k}\Lambda_{ai}(P_a x_i+Q_a y_i+fR_a-z_i) \tag{16}$$

without a constraint. The equations to determine the solution are Eq. (12) as well as those obtained by partial differentiations of Eq. (16) by P_α, Q_α, R_α. In summary,

Theorem 2. $n+3m+L-1$ unknowns z_i, $i = 1, \ldots, n-1$, P_α, Q_α, R_α, $\alpha = 1, \ldots, m$ and $\Lambda_{\alpha i}$ are determined from the following linear equations:

$$P_a x_i+Q_a y_i+fR_a-z_i=0 \qquad (F_a, V_i) \tag{17}$$

$$N_{a11}P_a+N_{a12}Q_a+N_{a13}R_a+\sum_i x_i\Lambda_{ai}=N_{a13} \qquad a=1,\cdots,m \tag{18}$$

$$N_{a21}P_a+N_{a22}Q_a+N_{a23}R_a+\sum_i y_i\Lambda_{ai}=N_{a23} \qquad a=1,\cdots,m \tag{19}$$

$$N_{a31}P_a+N_{a32}Q_a+N_{a33}R_a+f\sum_i\Lambda_{ai}=N_{a33} \qquad a=1,\cdots,m \tag{20}$$

$$\sum_a\Lambda_{ai}=0 \qquad i=1,\cdots,n-1 \tag{21}$$

where $N_{\alpha ij}$ is given by

$$N_{aij}=\sum_{k=1}^{Na}n_{ka(i)}n_{ka(j)} \qquad i,j=1,2,3 \tag{22}$$

Summations in Eqs. (18) - (20) are performed over all i for which (F_α, V_i).

In Theorem 2, it is assumed that each face has at least an edge for which 3-D orientation is computed. However, this assumption is not necessary. If the terms corresponding to the face F_α, whose edges are not given 3-D orientations, are removed from the right-hand side of Eq. (16), Eqs. (18) to (20) are replaced by the following equations:

$$\sum_i x_i\Lambda_{ai}=0, \quad \sum_i y_i\Lambda_{ai}=0, \quad \sum_i\Lambda_{ai}=0 \tag{23}$$

where summations are performed over all i for which (F_α, V_i).

5. Algorithm to Find Parallel Edges

To find parallel edges from a projected image, it may seem reasonable to search for almost parallel edges on the image plane. Such a naive approach is insufficient, since the angle on the image plane formed by parallel edges in the scene depends on the direction of projection. The proposed method makes use of the fact that parallel edges in the scene have a single vanishing point on the image plane. In other words, if three or more edges, when extended on the image plane, intersect at a point, they are judged as parallel in space.

In general, images resulting from image processing stages, however, contain errors. Consequently, it may happen that parallel

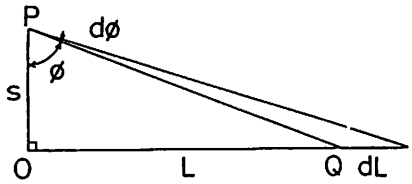


Fig. 6. A right spherical triangle $\triangle OPQ$.

edges, when extended, do not intersect at a single point (Fig. 4). Hence, a threshold ε must be set, and if the maximum separation between the intersections is less than ε, they are judged as parallel. However, we cannot use a fixed value for ε, since the error becomes larger as the intersections are further apart from the center of the image.

Another point to consider is the overflow of computation, since the intersection of two edges may be far out of the image. To deal with this case, the image sphere (Gauss sphere) [4, 11, 19] with radius f centered at the viewpoint (0, 0, $-f$) is used instead of the image plane (Fig. 5). Consider a spherical rectangular triangle $\triangle OPQ$ drawn on the image sphere (Fig. 6).

Let the arc lengths of OP and OQ be s and L, respectively, and let the angle formed by PO and PQ be ϕ. Then by differentiating the formula $\tan\phi = \tan(L/f)/\sin(s/f)$ for spherical triangles, we obtain

$$\frac{dL}{d\phi}=f\left(\frac{1}{\sin(s/f)}-\left(\frac{1}{\sin(s/f)}-\sin\frac{s}{f}\right)\cos^2\frac{L}{f}\right) \tag{24}$$

Letting $s/f \ll 1$, we obtain

$$\frac{dL}{d\phi}\approx\frac{f^2}{s}(1-\cos^2(L/f)) \tag{25}$$

The foregoing expression is regarded as the uncertainty of the intersection of two lines separated by distance s. We identity L with the arc length from the origin to the intersection of two edges on the image sphere.

Let the unit surface normal to the plane passing through edge e and the viewpoint (0, 0, $-f$) be n. Similarly, let the unit surface normal in regard to edge e' be n'. Let k = (0, 0, 1). Then the arc length L to the intersection of extensions of edges e and e' from the center O of the image is given by

$$L=f\cos^{-1}\eta_{ee'} \tag{26}$$

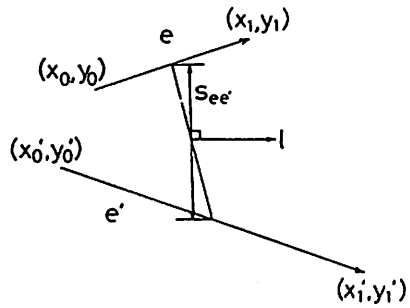


Fig. 7. The average separation $s_{ee'}$ of edges e, e'.

where $|nn'k|$ is the scalar triple product and

$$\eta_{ee'}=\left|\frac{|nn'k|}{\|n\times n'\|}\right| \tag{27}$$

Using the forementioned expression, the threshold for two edges e and e' is defined as follows:

$$\varepsilon_{ee'}=\frac{f^2\Delta\phi_{ee'}}{s_{ee'}}(1-\eta_{ee'}{}^2) \tag{28}$$

Here, $s_{ee'}$ is the average separation between the two edges, and $\Delta\phi_{ee'}$ is the admissible error of the orientations of the two edges. They are determined as follows. Let the coordinates of the endpoints of edge e be (x_0, y_0) and (x_1, y_1), and those of edge e' be (x_0', y_0') and (x_1', y_1'). Then let

$$\begin{aligned} e&=(x_1-x_0, y_1-y_0)\\ e'&=(x_1'-x_0', y_1'-y_0') \end{aligned} \tag{29}$$

The average orientation l weighted by the lengths of edges e and e' is defined by

$$l=\begin{cases}\dfrac{e+e'}{\|e+e'\|} & (e, e')\geq 0\\[2ex] \dfrac{e-e'}{\|e-e'\|} & (e, e')<0\end{cases} \tag{30}$$

The average separation $s_{ee'}$ is defined as the distance of the projection of the segment connecting midpoints of edges e and e' onto the axis perpendicular to the average orientation $l = (l_1, l_2)$ (Fig. 7)

$$s_{ee'}\equiv\left(\frac{x_0+x_1}{2}-\frac{x_0'+x_1'}{2}\right)l_2$$

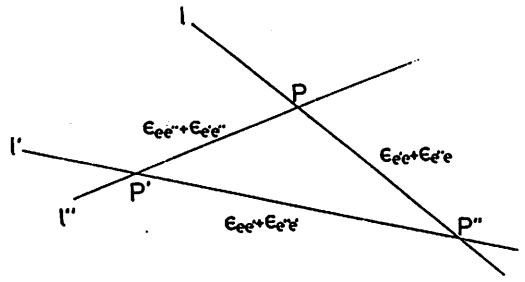


Fig. 8. Concurrency test: The concurrency of three lines is tested on the image sphere.

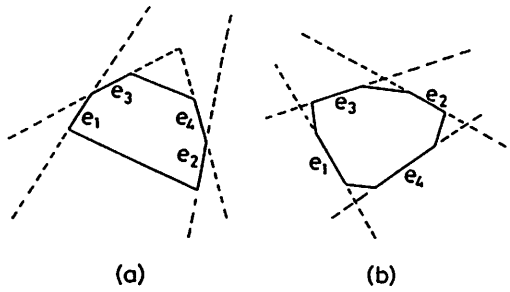


Fig. 9. Parallelogram test: If edges e_1 and e_2 are parallel, edges e_3 and e_4 cannot be parallel in (a) but can be parallel in (b).

$$-\left(\frac{y_0+y_1}{2}-\frac{y_0'+y_1'}{2}\right)l_1 \tag{31}$$

Since the error is likely to be larger with the decrease of the edge length, the admissible error of edge orientation is defined by

$$\Delta\phi_{ee'}=\frac{\text{const.}}{\min(|e|,|e'|)} \tag{32}$$

where $|e|$ and $|e'|$ are the lengths of edges e and e', respectively. The algorithm to find the parallel edges is given in the following.

(1) Concurrency test

First, a pair of edges e and e', which do not share end points, are selected as a candidate for parallel edges. Let the intersection of their extensions be P''. Consider a third edge e''. Let the intersection of the the extensions of edges e'' and e be P and that of edges e'' and e' be P'. If

$$PP'<\varepsilon_{ee''}+\varepsilon_{e'e''},\quad PP''<\varepsilon_{ee'}+\varepsilon_{e''e'}$$
$$P'P''<\varepsilon_{e'e}+\varepsilon_{e''e} \tag{33}$$

measured in arc length on the image sphere, these three edges are judged as parallel, and they are registered into the list of parallel edges (Fig. 8). Since the arc length on the image sphere is used, there is no danger of computational overflow.

If edges that share end points are judged to be parallel, the one which passes this test with a larger threshold is deleted from the list. If no other edges are found to be parallel to the first selected edges e and e', the first selected edges are judged to be not parallel.

(2) Vanishing point heuristic

It makes sense to select the candidates for parallel edges in the order likelihood of being parallel. It is assumed in this paper that a pair of edges is more likely to be parallel since their intersection is further apart from the center of the image. We called this the vanishing point heuristic. In other words, one should start from two edges e and e' with the smallest value of $\eta_{ee'}$, defined by Eq. (27). If the intersection is near the center of the image, the two edges are less likely to be parallel. Consequently, those edges for which $\eta_{ee'} > \cos(L_0/f)$ are not selected, where L_0 is the distance to the vertex position farthest from the center of the image.

(3) Coplanarity test

The foregoing procedure detects a set of three or more parallel edges. The remaining parallel edges are found by the following procedure. It is quite exceptional that two edges which are not on the same face happen to be parallel, and yet no other edges are parallel to those. Consequently, only those pairs of edges that share common faces but no common end points are tested. The following two tests are applied to such pairs of edges, and those edges that pass both of those tests are judged to be parallel edges.

(4) Parallelogram test

Consider two pairs of parallel lines. When projected, two half lines starting from their respective vanishing points must yield four intersections (Fig. 9) (for the algorithm, see Appendix 2).

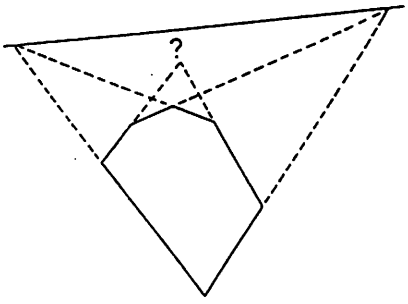


Fig. 10. Collinearity test: The vanishing points of parallel edges belonging to the same face must be collinear.

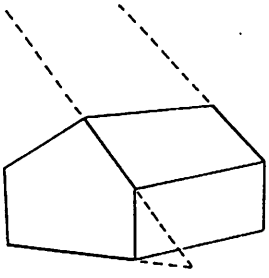


Fig. 11. Vanishing point heuristic: Two edges are more likely to be parallel if their intersection is farther away from the image origin.

(5) Collinearity test

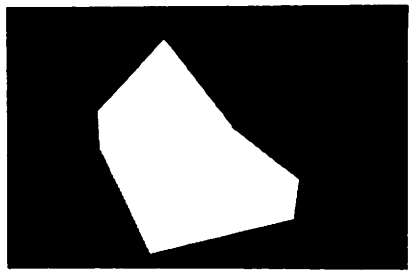
If there exists a set of three or more parallel edges sharing the same face, their vanishing points must be located on a common line (Fig. 10, the algorithm is given in Appendix 3).

If an edge is judged as parallel to more than one edge, the one whose vanishing point is farthest from the center of the image is selected by invoking the vanishing point heuristic (Fig. 11).

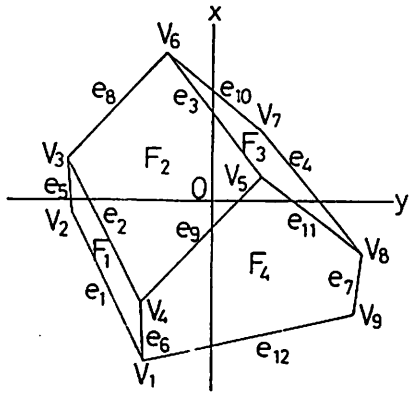
6. Example

Figure 12(a) is an actual image of a polyhedron. Assume that the line drawing of Fig. 12(b) is obtained from the image. The concurrency test detects the following sets of parallel edges:

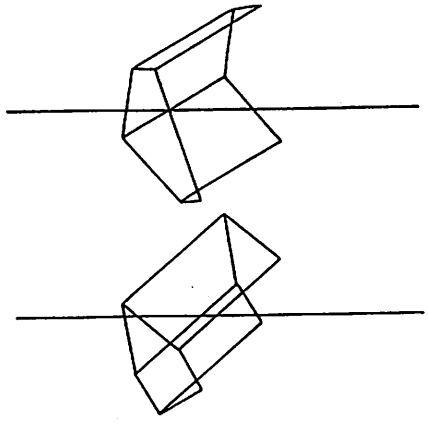
$$\{e_1, e_2, e_3, e_4\},$$
$$\{e_5, e_6, e_7\}$$



(a)



(b)



(c)

Fig. 12. (a) An image of a polyhedron; (b) labelling of its drawing; and (c) the top view and the side view of the reconstructed shape.

Among the remaining edges, the following pairs of edges share common faces but no common end points (coplanarity test).

$$\{e_8, e_9\},\ \{e_{10}, e_{11}\},\ \{e_9, e_{12}\},\ \{e_{11}, e_{12}\}$$

All those pairs pass the parallelogram test and the collinearity test. Since e_9 is shared by the pairs $\{e_8, e_9\}$ and $\{e_9, e_{12}\}$.

Hence, the former, whose intersection is farther away, is selected (the vanishing point heuristic). Similarly, edge e_{11} is selected. As a result, the following pairs of parallel edges are obtained,

$$\{e_1, e_2, e_3, e_4\},\ \{e_5, e_6, e_7\},\ \{e_8, e_9\},\ \{e_{10}, e_{11}\}$$

Then their vanishing points are computed and their 3-D orientations are estimated. Finally, the optimization is applied, and the 3-D shape shown in Fig. 12(c) is recovered.

7. Conclusions

This paper has presented a mathematical method that recovers the 3-D shape of a polyhedron from its image under perspective projection. If parallel edges are detected, then 3-D orientations are determined from their vanishing points. The error of image processing is overcome by our optimization technique based on the structure of the polyhedron, and a consistent 3-D shape is recovered.

The method in this paper is based on the nonlinear optimization constrained by the incidence structure of polyhedra originally presented by Sugihara [17, 18]. Our method employs a special variable transformation to reduce the problem into a system of linear equations. Consequently, we need not guess a starting point and do iterative searches. We have also presented a method to find parallel edges based on the concurrency test, the vanishing point heuristic, the coplanarity test, the parallelogram test, and the collinearity test. The proposed method was applied to an example derived from an actual image.

Acknowledgement. The authors are indebted for the discussions by Mr. T. Shakunaga, NTT Human Interface Lab., and Prof. K. Sugihara of the University of Tokyo. This study was supported in part by grants from Casio Sci. Found., Yazaki Mem. Found., Inamori Found., and Sci. Grant Min. Educ. (Gen. (C) 63550268).

REFERENCES

1. D. H. Ballard and C. M. Brown. Computer Vision. Englewood Cliffs, NJ (1982) [Tr. A. Fukumura et al. Jap. Computer Soc., 1987].
2. S. T. Barnard. Choosing a basis for perceptual space. Comput. Vision Graphics Image Process., 29, pp. 87-99 (1985).
3. M. Brady and A. Yuille. An extremum principle for shape from contour. IEEE Trans. Pattern Anal. Machine Intell., PAMI-6, pp. 288-301 (1984).
4. Y. Inamoto, S. Kawakami, T. Uchiyama, Y. Yasukawa, and T. Morita. Measurement of 3-D direction and distance of line segments by spherical mapping. IPSJ SIG Reports, Japan, 86-CV-45 (Nov. 1986).
5. T. Kanade. Recovery of the three-dimensional shape of an object from a single view. Artif. Intell., 17, pp. 409-460 (1981).
6. K. Kanatani. The constraints on images of rectangular polyhedra. IEEE Trans. Pattern Anal. Machine Intell., PAMI-8, pp. 456-463 (1986).
7. K. Kanatani, T. Tanaka, K. Maehara, and T. Kawashima. Three-dimensional shape recovery of polyhedra by optimization. Proc. 36th Annual Convention IPS, Japan, 5V-4, pp. 1643-1644 (1988).
8. K. Kanatani and J. Yoshida. Noise robust 3-D recovery from optical flow. IPSJ SIG Reports, 87-CV-48 (May 1987).
9. A. K. Mackworth. Model-driver interpretation in intelligent vision systems. Perception, 5, pp. 349-370 (1976).
10. K. Maehara, T. Kawashima, and K. Kanatani. Three-dimensional recovery of polyhedra by rectangularity heuristics. Trans. (D), I.E.C.I.E., Japan, J72-D-II, 6, pp. 887-895 (June 1989).
11. M. J. Magee and J. K. Aggarwal. Determining vanishing points from perspective images. Comput. Vision Graphics Image Process., 26, pp. 256-267 (1984).
12. P. G. Mulgaonkar, L. G. Shapiro, and R. M. Haralick. Shape from perspective: A rule-based approach. Comput. Vision Graphics Image Process., 36, pp. 298-320 (1986).
13. H. Nakatani and T. Kitahashi. Reconstruction of object in outdoor scene based on vanishing point. Trans. (D), I.E.C.E., Japan, J68-D, 8, pp. 1481-1488 (Aug. 1985).
14. H. Nakatani and T. Kitahashi. An iterative method for locating a vanishing point. Trans. (D), I.E.C.E., Japan, J68-D, 8, pp. 1541-1542 (Aug. 1985).
15. Y. Shirai. Computer Vision. Shokodo Publ. Co. (1980).
16. K. Sugihara. Mathematical structures of line drawings of polyhedra toward man-machine communication by means of line drawings. J. IPS, Japan, 22, 3, pp. 209-217 (March 1981).
17. K. Sugihara. Algebraic approach to the recovery of three-dimensional shape from from single images. Trans. (D), I.E.C.E., Japan, J66-D, 5, pp. 541-548 (May 1983).
18. K. Sugihara. Machine Interpretation of Line Drawings. MIT Press, Cambridge, MA (1986).
19. Y. Yagi, M. Asada, M. Yachida, and S. Tsuji. Dynamic scene analysis for a mobile robot in a mammade environment. Trans. (D), I.E.C.E., Japan, J69-D, 6, pp. 967-974 (June 1986).

APPENDIX

1. Algorithm for Computing Vanishing Points without Overflow

Three or more parallel edges should have the same vanishing point, which, however, is not true in the presence of error (Fig. 3). Consequently, the least-mean-square method is employed as follows.

Assume that lines $L_i : A_i x + B_i y + C_i = 0$, $i = 1, \cdots, N$ are projections of parallel edges. Let the plane passing through line L_i on the image plane and the viewpoint $(0, 0, -f)$ be S_i. Let its unit surface normal be $n_i = (n_{i(1)}, n_{i(2)}, n_{i(3)})$. Let the shared vanishing point to be determined be (a, b). The distance from point (a, b) to plane S_i is $| n_{i(1)}a + n_{i(2)}b + n_{i(3)}f |$ (Fig. A1).

Consequently, (a, b) is determined by minimizing the following expression:

$$\sum_{i=1}^{N} (n_{i(1)}a + n_{i(2)}b + n_{i(3)}f)^2 \tag{A1}$$

Differentiating the foregoing expression with respect to a and b, and equating the results to 0, we obtain

$$a = fm_1'/m_3', \quad b = fm_2'/m_3' \tag{A2}$$

where

$$m_1' = \sum_{i=1}^{N} n_{i(1)}n_{i(3)} \sum_{i=1}^{N} n_{i(2)}^2 - \sum_{i=1}^{N} n_{i(2)}n_{i(3)} \sum_{i=1}^{N} n_{i(1)}n_{i(2)},$$

$$m_2' = \sum_{i=1}^{N} n_{i(1)}^2 \sum_{i=1}^{N} n_{i(2)}n_{i(3)} - \sum_{i=1}^{N} n_{i(1)}n_{i(2)} \sum_{i=1}^{N} n_{i(1)}n_{i(3)},$$

$$m_3' = -\left(\sum_{i=1}^{N} n_{i(1)}^2 \sum_{i=1}^{N} n_{i(2)}^2 - \left(\sum_{i=1}^{N} n_{i(1)}n_{i(2)} \right)^2 \right) \tag{A3}$$

Then $m' = (m_1', m_2', m_3')$ is the vector indicating the 3-D orientation of those parallel edges. Since no division is required, there is no danger of overflow.

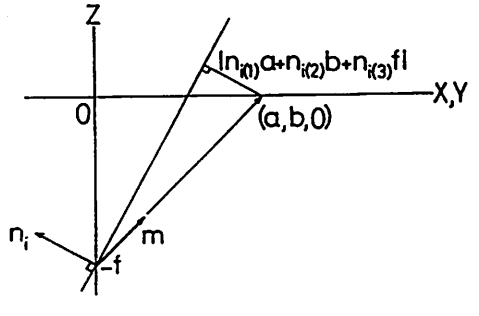


Fig. A1. The distance from point (a, b) on the image plane $Z = 0$ to the plane S_i passing through the viewpoint and line L_i on the image plane is minimized.

2. Algorithm of Intersection Test for Half-Lines without Overflow

The parallelogram test splits into tests for the existence of intersections of two half lines. Let the half line starting from point (a, b) and passing through point (x, y) be L, and let the one starting from point (a', b') and passing through point (x', y') be L' (Fig. A2).

The intersection of L and L' exists if and only if there exist $t, t' > 0$ such that

$$\begin{aligned} a + t(x - a) &= a' + t'(x' - a'), \\ b + t(y - b) &= b' + t'(y' - b') \end{aligned} \tag{A4}$$

It follows from the foregoing expressions that

$$\begin{aligned} t &= \frac{1}{\Delta}[(x' - a)(b' - b) - (y' - b)(a' - a)], \\ t' &= \frac{1}{\Delta}[(x - a)(b' - b) - (y - b)(a' - a)], \\ \Delta &= (x - a)(y' - b) - (y - b)(x' - a) \end{aligned} \tag{A5}$$

Thus, the decision can be made by examining whether or not all the following expressions have the same sign:

$$(x' - a)(b' - b) - (y' - b)(a' - a),$$

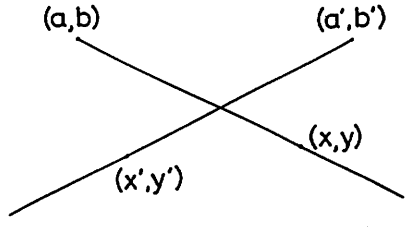


Fig. A2. Test for intersection of two half-lines.

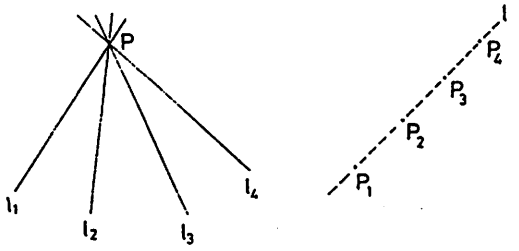


Fig. A3. Duality principle: The lines dual to collinear points are concurrent and the common intersection is dual to the common line passing through the collinear points.

$$(x-a)(b'-b)-(y-b)(a'-a),$$
$$(x-a)(y'-b)-(y-b)(x'-a) \qquad \text{(A6)}$$

However, if points (a, b) and (a', b') correspond to the vanishing points of parallel edges, they may be located far from the center of the image, causing computational overflow. In terms of the vectors n and n' starting from the viewpoint and pointing toward (a, b) and (a', b') on the image plane, a, b and a', b' are represented as follows:

$$a=fn_1/n_3,\quad b=fn_2/n_3,$$
$$a'=fn_1'/n_3',\quad b'=fn_2'/n_3' \qquad \text{(A7)}$$

Substituting those expressions into Eq. (A6) and multiplying the denominators by n_3 (>0) and n_3' (>0), we obtain

$$(n_3x-fn_1)(n_2'n_3-n_2n_3')$$
$$-(n_3y-fn_2)(n_1'n_3-n_1n_3'),$$
$$(n_3'x'+fn_1')(n_2'n_3-n_2n_3')$$
$$-(n_3'y'-fn_2')(n_1'n_3-n_1n_3'),$$
$$(n_3x-fn_1)(n_3'y'-fn_2')$$
$$-(n_3y-fn_2)(n_3'x'-fn_1') \qquad \text{(A8)}$$

Hence, no overflow occurs. If all of the foregoing expressions are of the same sign, we conclude that the half-lines L and L' intersect.

3. Algorithm for Line Fitting without Overflow

If the unit vector starting from the viewpoint $(0, 0, -f)$ and pointing toward P on the image plane is at the same time the unit surface normal to the plane passing through the viewpoint $(0, 0, -f)$ and the line L on the image plane, the point P and the line L are said to be dual. The point dual to line $Ax + By + C = 0$ is given by $(f^2A/C, f^2B/C)$. The line dual to a point (a, b) is $ax + by + f^2 = 0$.

If multiple lines intersect at a single point P, the points dual to these lines are located on a common line L, and L is the line dual to the intersection P. This is proved as follows. Let the lines intersecting at point $P(a, b)$ be $Ax_i+By_i+C_i=0$, $i=1, \cdots, N$. Then $A_ia+B_ib+C_i=0$, $i=1, \cdots, N$ The points $(f^2A_i/C_i, f^2B_i/C_i)$, $i=1, \cdots, N$ dual to those are always located on the line L: $ax + by + f^2 = 0$.

If multiple points are located on the same line L, the lines dual to these points intersect at a common point P, and P is the point dual to the line L. This is proved as follows. Let points (a_i, b_i), $i = 1, \ldots, N$ be on line L given by $Ax + By + C = 0$. Then $Aa_i+Bb_i+C=0$, $i=1, \cdots, N$. Consequently, the dual lines $a_ix+b_iy+f^2=0$, $i=1, \cdots, N$ always pass through point $P(f^2A/C, f^2B/C)$.

In view of the forementioned duality principle, the decision as to whether multiple points are on a common line is reduced to the decision as to whether their dual lines have a common intersection (Fig. A3). Here, fitting of a line to points which are not necessarily collinear is reduced to determining a common intersection of their dual lines by the algorithm of Appendix 1. This is essentially the Hough transform.

AUTHORS (from left to right)



Toshie Tanaka graduated in 1988 from the Dept. of Computer Science, Gumma University, and affiliated with Mitsubishi Electric Corp.

Takao Kawashima graduated in 1988 from the Dept. Computer Science, Gumma University, and entered the Master's program there.

Ken-ichi Kanatani graduated in 1972 from the University of Tokyo. He was an Exchange Student at Case Western Reserve University, USA, from 1969 to 1970, and obtained a Dr. of Eng. degree in 1979 from the University of Tokyo. He was an Assistant in 1979, Assoc. Professor in 1983, and Professor in 1988 at the Dept. Computer Science, Gumma University. He was a Visiting Researcher from 1985 - 1986 at the University of Maryland, USA, and Visiting Professor in 1988 at University of Copenhagen, Denmark. He received a Paper Award in 1986 from Inf. Proc. Soc., Japan. He is a member of the Inf. Proc. Soc., Japan; Robotics Society of Japan; and IEEE.