An Algorithm for Hidden Line Elimination of Polyhedra

EIHACHIRO NAKAMAE* AND TOMOYUKI NISHITA*

1. *Introduction*

Since I. E. Sutherland [1] pointed out the need of eliminating the hidden portions of the three dimensional figures on the occasion of displaying on a plane, many papers have been published [2]~[4]. The authors simplified the decision of whether each plane is front or back and of the domains of a convex polyhedron divided an object suitably. Furthermore, the decision for eliminating of hidden line about the inner edges of the independent convex polyhedra for each other is unnecessary. The merits of this algolithm are as follow ; 1) The input data are so simple that its errors are decreased exceedingly. 2) The procedure for hidden line elimination is so simple, because a contour line of each convex polyhedron is convex, too. 3) The design for the structures are very easy, as the position and size of each convex polyhedron is able to modify individually. 4) The figures on the plane of an object are able to be dealed on the same algorithm as the inner edges.

2. *Data Structure*

A vertex is given with the three dimensional coordinates and a plane is given with the vertex's number ordered clockwise from outside. The standardization for the object used very often is convenient. Our program, therefore, has the subroutines designing automatically such as the parallelepiped, the column, the convex polygonal plane and so forth.

3. *Algorithm*

We define that on the object the planes oriented to an observation point are the front faces, the opposite planes are the back faces, an edge constructed with two front faces is an inner edge, an edge constructed with a front face and a back face is a contour edge and a chain of the contour edges is a contour line. All of the inner edges and contour edges are the potentially visible line segments. The contour line on the projection plane is a convex polygon and its contour line which is traced to the direction of the front faces is clockwise. There are three correlations between the contour lines of two convex polyhedra shown in Fig. 1. 1) The contour lines do not overlap. There is no invisible

This paper first appeared in Japanese in Joho-Shori (Journal of the Information Processing Society of Japan), Vol. 13, No. 4 (1972), pp. 239~246.
* Faculty of Engineering, Hiroshima University

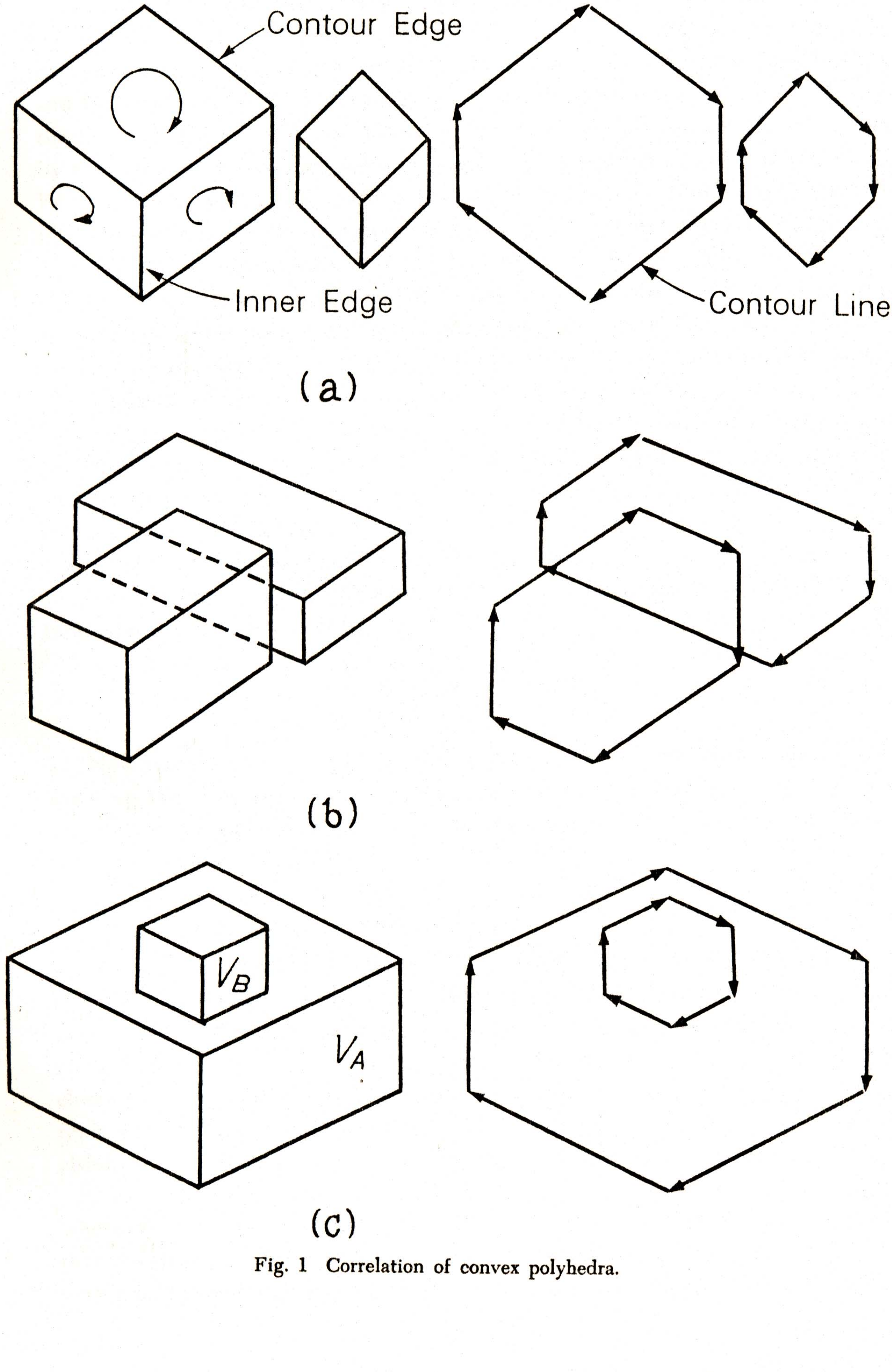


Fig. 1 Correlation of convex polyhedra.

segment because the both contour lines are independent each other. 2) The contour lines overlap partially. The invisible line segments can be found by searching the intersection between the contour line of the near convex polyhedron and the inner edges of another. 3) A contour line encloses another. Whether the convex polyhedron V_A enclosing V_B is nearer than V_B or not can be found by the relation between an arbitrary vertex of V_B and the plane enclosing it. If V_B is nearer than V_A, the invisible line segments can be found by searching the intersection between the contour line of V_B and the inner edges of V_A.

3.1 *Decision of Orientation of Plane and Classification of Edges.*

The all planes are convex polygon and if the direction of two edges of a plane is clockwise on the projection plane, the plane is a front face. If a plane shown by an ordered set of vertices $[P_1, P_2, P_3, \cdots]$, angle $\angle P_1, P_2, P_3$ is convex. The decision of the orientation of the plane, therefore, can be obtained by a value of $f_{1,2}(P_3)$ in the formula (1). The edges are classified the inner edges and the contour edges by the relation of adjacent vertices and the planes including them and the contour line is also searched by using the characteristics of the directions of contour edges.

3.2 *Decision of the Intersection of Contour Line.*

A point $P(x, y)$ exists on right half of the line segment $[P_i, P_{i+1}]$, if the following formula is positive,

$$f_{i,i+1}(P) = -\begin{vmatrix} x & y & 1 \\ x_i & y_i & 1 \\ x_{i+1} & y_{i+1} & 1 \end{vmatrix} = a_i \cdot x + b_i \cdot y + c \qquad (1)$$

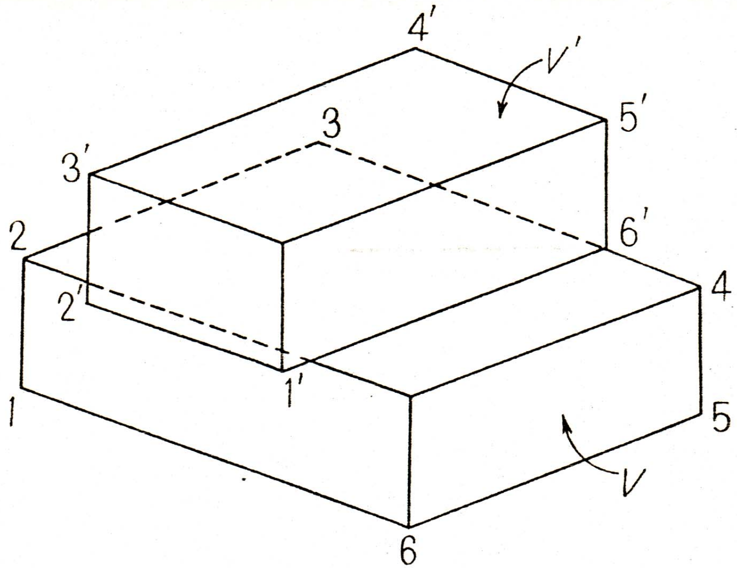
Now we consider the contour lines of two convex polyhedra V and V', and draw up the PL and LP tables finding out whether the vertices of contour line have place on right half of the each line segment of another contour line or not. The elements of them are

$$PL(j, i) = sgn f_{i,i+1}(P_j)$$
$$LP(j, i) = sgn f_{j,j+1}(P_i) \qquad (2)$$

The PL table has a following characteristics (refer to Fig. 2). 1) If there is at least one column including no element of 1, V is independent of V'. 2) If no element of the table has any -1, V encloses V'. 3) A line segment not including any -1 in its column has not any intersection. 4) The vertex not including any -1 in its row is enclosed by another contour line. If the row is replaced with the column, the LP table has the same characteristics. The decision of line segments having a possibility to intersect is obtained by

$$d_{i,j} = (LP(j, i+1) - LP(j, i)) \cdot (PL(j+1, i) - PL(j, i)) \qquad (3)$$

where $d_{i,j}$ equals -4 means that $[P_i, P_{i+1}]$ intersects to $[P_j, _{j+1}]$, -2 means that a vertex lies on another line segment, -1 means that a vertex overlaps a vertex of another line segment and 0 means not to intersect with each other.



V' \ V	* 1−2	2−3	3−4	* 4−5	* 5−6	* 6−1
1'	1	1	1	1	1	1
2'	1	1	1	1	1	1
3'	1	−1	1	1	1	1
4'	1	−1	−1	1	1	1
5'	1	1	−1	1	1	1
6'	1	1	0	1	1	1

PL table

V' \ V	** 1	2	3	4	** 5	** 6
* 1'−2'		1	1	1		
2'−3'		−1	1	1		
* 3'−4'		1	1	1		
* 4'−5'		1	1	1		
5'−6'		1	1	−1		
6'−1'		1	1	−1		

LP table

	−4	0			
	0	−2			
	0	−2			

d_{ij}

Fig. 2 Decision of intersection of two contour lines.

3.3 *Decision of Relation of Location between Two Convex Polyhedra*

Whether the convex polyhedron V is nearer than V' or not is found out by the following method. In the case that V and V' intersect with each other, assuming that the points $P1$ and $P2$ are the points on V and V' corresponding to the intersection on the projection plane and a point P_0 is the observation point, P_1, P_2, and P_0 exist on a line. Which of P_1 and P_2, that is V and V', is near is found out by the sign of

$$dr=(P_1-P_0)\cdot(P_1-P_2) \tag{4}$$

In the case that V encloses V', which of them is near can be found out by the relation between any vertex P on V' and a plane S which belongs to V and encloses it. Assuming that a front face S is constructed $[P_1, P_2, P_3, \cdots]$, which of P and S is near is obtained by the sign of

$$\Delta r(P)=-\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} \tag{5}$$

3.4 *Decision of Intersection of Contour Line and Inner Edge and Elimination of Ghost Line*

In the case that V hides V', the decision of the intersection of contour line and inner edges is found out by the relation of the contour line segments of V enclosed by the contour line of V', and the inner edges of V'. The decision can be done by the method as like as the algorithm 3.3 using LP and PL table

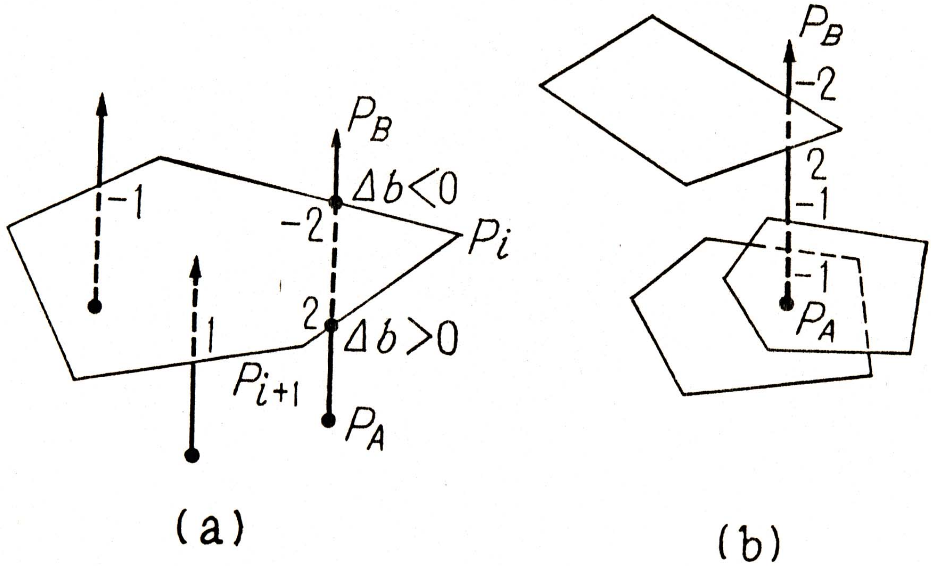


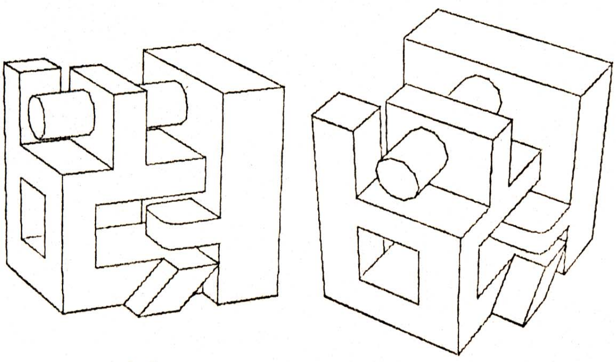
Fig. 3 Invisibility of line segment.

which is constructed by the sign of formula (4).

As a object is divided to the convex polyhedra, it needs that the ghost line which is caused by dividing a plane is eliminated. The overlapping of two line segments can be found out by the elements in *LP* or *PL* table. In the case that the two front faces which include the overlapping line segments exist on the same plane, this line is, therefore, eliminated as a ghost line.

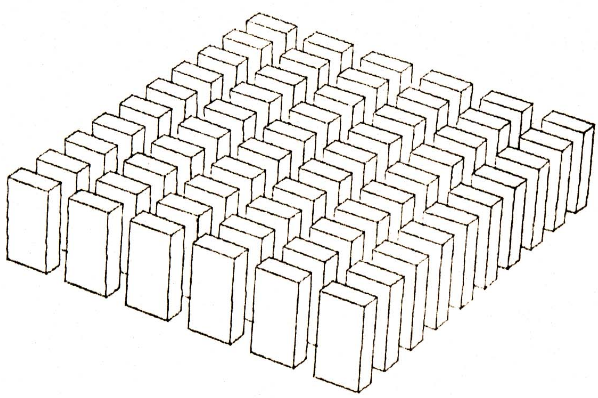
3.5 *Decision of Invisible Part*

The number of the intersection between a contour line and a segment is no more than two, and the conditions to intersect are only three as shown in Fig. 3 (a). As the contour line is always clockwise direction, the invisible part of a segment $[P_A, P_B]$ is decided by $(\boldsymbol{P}_i-\boldsymbol{P}_{i+1})\times(\boldsymbol{P}_A-\boldsymbol{P}_B)$. An index I_0 of the line segment going in or out and the intersection are necessary to memorize, because the segment may have more intersections with other contour lines as shown in Fig. 3 (b),



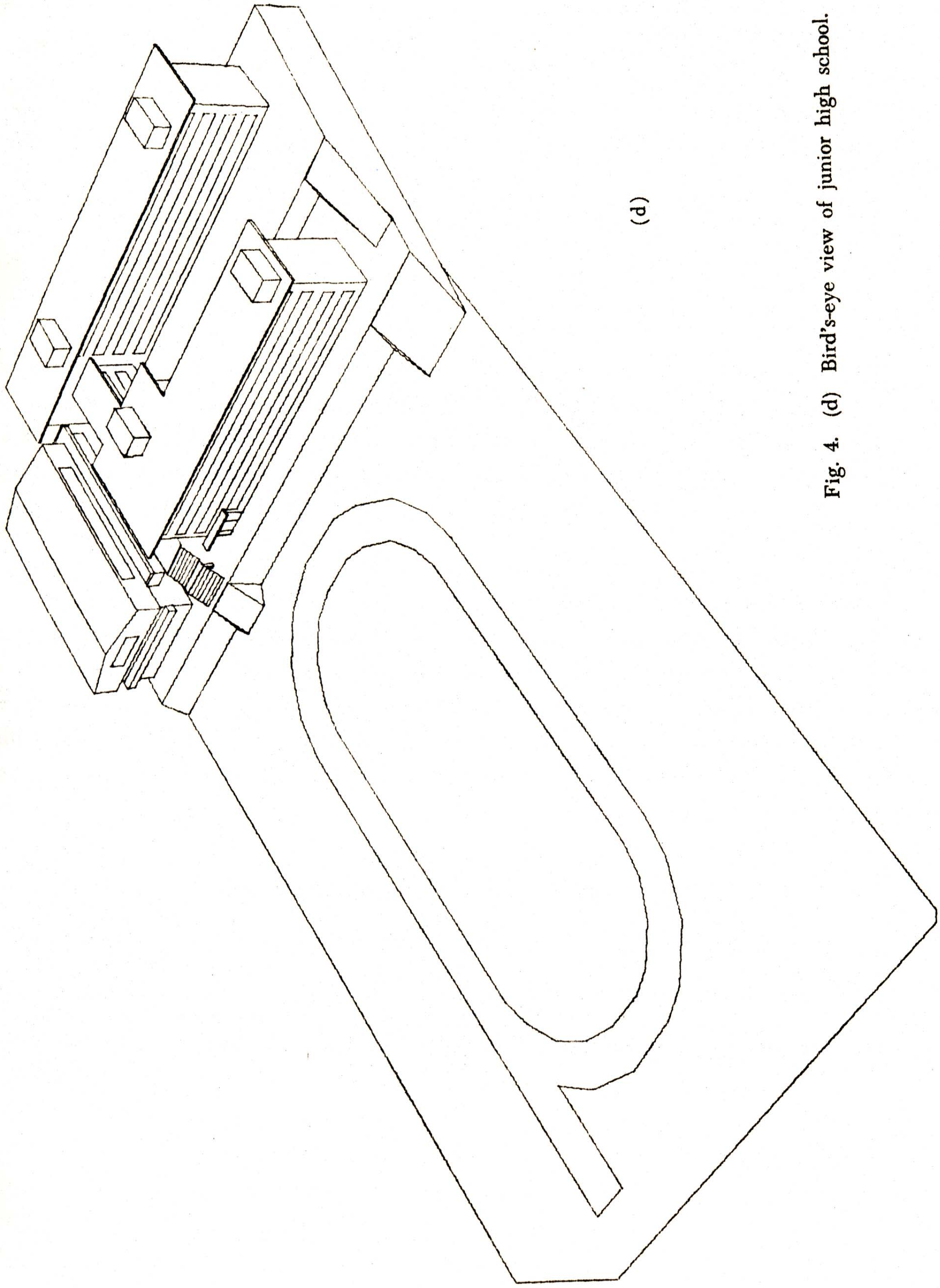
(a) (b)

Fig. 4 (a), (b) Set of standardized convex polyhedra.



(c)

Fig. 4 (c) Array of 60 cuboids.



(d)

Fig. 4. (d) Bird's-eye view of junior high school.

Table 1. Computing time.

Fig.	Convex polyhedra	Vertices	Faces	Figures (points)	Computing time (second)		
					I	II	III
(a)	11	116	80	0	0.67	0.13	0.33
(b)	11	116	80	0	0	0.13	0.44
(c)	60	480	360	0	2.49	0.47	5.09
(d)	30	216	158	184	2.14	0.24	1.90

$$I_o = k \cdot sgn \begin{vmatrix} x_{i+1}-x_i & y_{i+1}-y_i \\ x_A - x_B & y_A - y_B \end{vmatrix} \qquad (6)$$

where k is the number of the intersection. On the occasion of drawing a picture, the all intersections of the line segment is replaced in order locating near P_A, and the order of invisibility of P_A is counted by the number of -1. The visible portion begins from the location which the order becomes 0, because the index is counted each time of passing the point of intersections.

4. *Example and Conclusion*

Some examples are shown in Fig. 4. They are calculated by FACOM230/60. The computing time is shown by Table 1 in which (I) is the time for making up the objects, (II) is the time for transformation of the coordinate and searching the contour lines and the inner edges, and (III) is the time for finding out the hidden lines. We believe that this algorithm is practical from some points like that its computing time is short and it has ability of application to many polyhedra.

References

[1] I. E. Sutherland : Computer Graphics Ten Unsolved Problems, *Datamation*, Vol. 12, No. 5, p. 22, 27 (May. 1966).

[2] R. Galimberti & U. Nontanari : An algorithm for hidden line elimination, *Comm. ACM*, *12*, p. 206, 211 (Apr. 1969).

[3] P. P. Loutrel : A solution to the hidden-line problem for computer-drawn polyhedra, *IEEE Trans. Computers*, Vol. c-19, p. 205, 213 (March. 1970).

[4] F. J. Mcgrath : A method for eliminating hiddenlines with polyhedra, Simulation, p. 37, 41 (Jan. 1971).