論文 한국정밀공학회지 제6권 제4호 1989. 12
Journal of the Korean Society of Precision Engineering Vol. 6, No 4 December 1989

AUTOMATIC CONVERSION OF DESIGN DRAWING FOR CAD/CAM INTEGRATION

김 호룡,* 홍 지수,** 조 성봉**

Automatic Conversion of Design Drawing For CAD/CAM Integration

H.R.Kim, J.S.Hong, S.B Cho

Abstract

An algorithm and its computer program are developed for the computer aided automatic conversion from 2 dimensional (2-D) design drawings to a 3 dimensional (3-D) solid object and the 3-D object obtained from the developed program was used to generate the tool path of NC machine. The algorithm and its computer program developed were applied to several real objects for their feasibility check and showed satisfactory results. As the results of this study, it was proved that a foundation work to prepare the data base for CAD/CAM integration can be established so as to improve the productivity.

—요 약—

컴퓨터를 이용하여 2차원 설계도면을 3차원 입체도형으로 자동변환 시키기 위한 알고리즘과 그 컴퓨터 프로그램을 개발하였다. 컴퓨터 프로그램으로 얻어진 3차원 입체도형은 NC공작기계로 가공하기 위한 공구경로 발생에 사용된다. 개발된 알고리즘과 컴퓨터 프로그램은 여러개의 실물에 적용시켜 그 타당성을 조사하였고 만족한 결과를 얻었다. 연구의 결과로서 CAD/CAM을 통합시키고 생산성을 향상시키기 위한 데이터 베이스 준비용 기초를 확립하였다.

* 연세대학교 기계공학과 교수
** 연세대학원

INTRODUCTION

Design drawings generally consist of three orthographic views (top, front, and side views) and represent a 3-D solid object through the three views. Thus, the 3-D object can be recognized from the orthographic views by using the specific relationship existing between the views.

When the drawings are, however, complicated, it takes a lot of time and efforts to recognize the 3-D object form the drawings and sometimes the trial and error procedure is inevitable. Accordingly, considerable research has done to grasp the drawings through the computer aided automatic conversion of the design drawing to 3-D object.

In 1973, Idesawa [1] suggested the fundamental algorithm for automatic conversion from design drawing to 3-D object and his research was confined to a polyhedron which comprises straight lines. Chang [2] in 1986 derived the formulation for automatic conversion to a straight lined polyhedron and facilitated the recognition of solid object by replacing hidden lines with broken lines. Yoon [3] attempted in 1987 the automatic conversion of design drawings which include circles and arcs in addition to the straight lined polyhedron. In 1988, Kim [4] converted design drawings having arbitrary curves to a solid object and suggested the primary algorithm which generates a 2 dimensional tool path of NC milling machine.

「Aldefeld [5] presented a method considering a solid as a composition of several isolated elementary parts of uniform thickness. Sakurai and Gossard [6] reported an algorithm that can handle straight lines and circular arcs, solid and dashed lines. Bin [7] used a method of constructive solid geometry representations for solid modelling in which the primitives such as cuboid, pyramid, cylinder, cone and sphere are parameterized. Kargas et. al[8] described concepts and techniques for transforming data, structured to correspond to orthographic views of engineering components, into formal 3-D solid model. They employed the constructive solid geometry which is the same process as used by Bin.」

The industry in these days requires the automation and precision in design and production through CAD/CAM technique in order to improve the productivity. Therefore, recent research orients toward integrating CAD/CAM which in practice has been separated, and now is trying to not only automatically convert to solid object by using computer but also develop the part program to manufacture the object by NC machine directly form the output of the conversion.

In this study, the algorithm for the automatic conversion of design drawings, which include ruled surface as well as straight lines, circles and arcs, to 3-D object is developed. From the 3-D object obtained, a cross section cut at arbitrary direction is drawn to show the inside of the object and used to obtain the tool path of NC machine. The algorithms described above are coded into Turbo-PASCAL so that the developed computer program may provide the foundation of CAD/CAM integration which is now essential for industry to improve its productivity.

RELATIONSHIP BETWEEN 2-D DESIGN DRAWING AND 3-D SOLID

The points on a design drawing, which are called as nodes and essential data for the automatic conversion and tool path generation, are classified into 3 kinds as shown in Fig. 1 : 1) standard node 2) tangency node 3) auxiliary node. The standard nodes are the intersecting points of straight lines and curves. The auxiliary nodes are the points of a view corresponding to the tangency nodes of other view and determined from all three views.

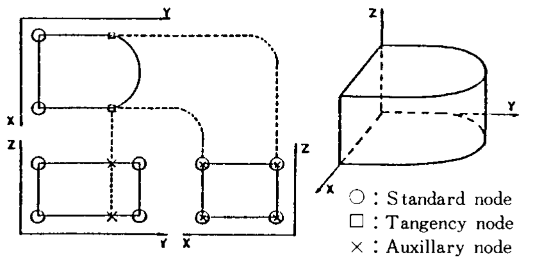


Fig. 1 Coordinates of 3-views with the kinds of nodes

On the contrary to the nodes of 2-D drawing, the points of 3-D solid is called as vertices. The 3-D solid and its 3 views of design drawing are defined as shown in Fig. 2.

The mathematical expressions are given for the 3-D solid

$$\underset{\sim}{V}=\underset{\sim}{S}_{xyz}=\{\underset{\sim}{R}_{PL},\ \underset{\sim}{P}_{xyz},\ \underset{\sim}{L}_{xyz}\} \quad \cdots\cdots(1)$$

Where

$\underset{\sim}{S}_{xyz}$ = set of surfaces

$\underset{\sim}{R}_{PL}$ = set of rows of edge lines

$\underset{\sim}{P}_{xyz}$ = set of vertices

$\underset{\sim}{L}_{xyz}$ = set of edge lines

and for its 3 views of design drawing

$$\underset{\sim}{D}=\underset{\sim}{F}(V)=\{\underset{\sim}{P}_{xy},\ \underset{\sim}{P}_{yz},\ \underset{\sim}{P}_{zx},\ \underset{\sim}{L}_{xy},\ \underset{\sim}{L}_{yz},\ \underset{\sim}{L}_{zx}\} \quad \cdots\cdots(2)$$

In accordance with the orthographic projection

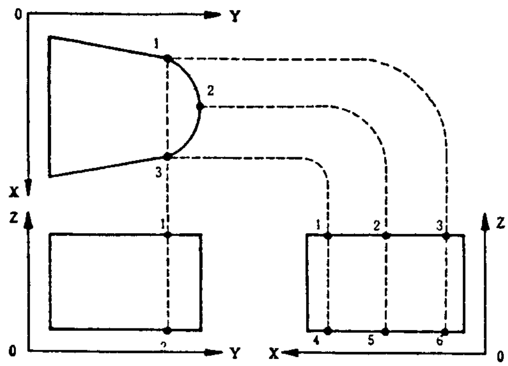


Fig. 2 Representation of 2-dimensional 3-views

rule, the 3-D solid can be transformed into 2 dimensional 3 views as shown in Fig. 3. Fig. 3(a) shows the process to transform the vertices and edges of 3-D solid into the nodes and lines of 2-D 3 views and Fig. 3(b) exhibits the process to transform the surface of 3-D solid into the lines of 2-D 3 views.

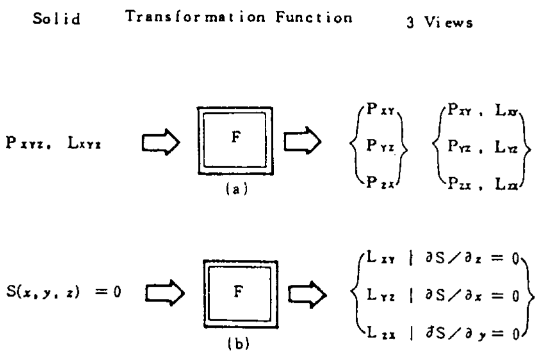


Fig. 3 Transformation from 3-dimensional solid to 2-dimensional 3-views through transformation function F.

The inverse transformation from 2-D 3 views to 3-D solid can be performed through the inverse function F of transformation function F as illustrated in Fig. 4.

The inverse function F^{-1} is expressed as

$$V = F^{-1}(D) = F_{PL}[F_{GH}\{F_{LN}(D) \cup F_{PN}(D)\}] \quad (3)$$

Where F_{PL} : Surface construction function

F_{GH} : ghost figure eliminating function

F_{LN} : transformation function to edge

F_{PN} : transformation function to vertices

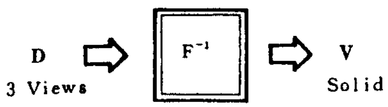
and defined as below

$$F_{PN}(D) = P'_{xyz} = P_{xy} \cap P_{yz} \cap P_{zx} \quad \cdots\cdots(4)$$

$$F_{LN}(D) = L'_{xyz} = \{L_{xy} \cap L_{yz} \cap L_{zx}\} \cup \{L_{xy} \cap L_{yz} \cap P_{zx}\} \cup \{L_{xy} \cap P_{yz} \cap L_{zx}\} \cup \{P_{xy} \cap L_{yz} \cap L_{zx}\} \quad \cdots\cdots(5)$$

$$F_{GH}[\{P'_{xyz}, L'_{xyz}\}] = \{P_{xyz}, L_{xyz}\} = \{P'_{xyz}, L'_{xyz}\} - \{P_{GH} \cup L_{GH}\} \cap \{P'_{xyz}, L'_{xyz}\} \quad \cdots\cdots(6)$$

$$F_{PL}[\{P_{xyz}, L_{xyz}\}] = S_{xyz} = K_{PL} \cap \{P_{xyz}, L_{xyz}\} \quad \cdots\cdots(7)$$



F.g. 4 Transformation from 2-dimensional 3-views to 3-dimensional solid through transformation function F.

In addition to circles and arcs, 3-D object such as machine elements or part may have irregularly curved surfaces. Those curved surfaces often can not be expressed in mathematical equation and the B-spline curve is generally employed to describe the surfaces.

When u is a parameter of B-spline curve and n+1 points of P_0, P_1, $\cdots P_n$ are given in space, the B-spline curve of K-1 degree is defined as

$$r(u) = \sum_{i=0}^{n} P_i N_{i,k}(u), \quad (k < n) \cdots\cdots(8)$$

where

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } t_i \leq u \leq t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,k}(u) = \frac{(u - t_i) N_{i,k-1}(u)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - u) N_{i+1,k-1}(u)}{t_{i+k} - t_{i+1}} \quad \cdots\cdots(9)$$

In Eq. (9), t_i is a knot of spline curve and defined as

$$t_i = \begin{cases} 0 & \text{if } i < k \\ i - k + 1 & \text{if } k \leq i \leq n \\ n - k + 2 & \text{if } i > n \end{cases} \quad \cdots\cdots(10)$$

The surface of B-spline curve can be obtained by the cartesian product of two B-spline curves which are denoted in terms of parameters u and v respectively. The surface is thus expressed as

$$r(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} P_{i,j} N_{i,k}(u) N_{j,l}(v) \quad (k \leq n, \ \ell \leq m) \cdots\cdots(11)$$

in which $P_{i,j}$ are points in space, $N_{i,k}(u)$ and $N_{j,l}(v)$ are defined as in Eq. (9), and the degrees of parameter u and v curves are (k-1) and (l-1) respectively.

ALGORITHM FOR AUTOMATIC CONVERSION

1. Read input data from 2-D design drawing (Fig. 5). In Fig. 5, P_{xy} is a set of points, C_{xy} is a set of point numbers adjacent to P_{xy}, G_{xy} is a set of group number to which P_{xy} belongs, and D_{xy} is a set of information concerning P_{xy}.

The group number is assigned by grouping more than three points on a same line and D_{xy} is determined according to the three cases below :

1) P_{xy} is on only solid line : 1

2) cases except 1) and 3) : 2

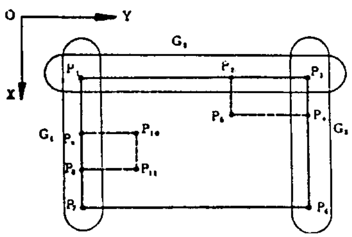
3) P_{xy} belongs to only broken line and its all neighboring points are on solid line : 3

A similar method is equally applied to yz (front view) and zx (side view) planes.

2. Transformation to vertices

The set of points in each view is expressed as

$$P_{xy} = \{(x_i,\ y_i,\ c) \mid i=1,\ N_{xy}\} \quad \cdots\cdots\cdots\cdots (12)$$
$$P_{yz} = \{(c,\ y_j,\ z_j) \mid j=1,\ N_{yz}\}$$
$$P_{zx} = \{(x_k,\ c,\ z_k) \mid k=1,\ N_{zx}\}$$



P_{XY}	C_{XY}	G_{XY}	D_{XY}
$P_1\ (x_1,\ y_1)$	2, 9, 0	1, 2	1
$P_2\ (x_2,\ y_2)$	1, 3, 5	2, 0	2
$P_3\ (x_3,\ y_3)$	2, 4, 0	2, 3	1
$P_4\ (x_4,\ y_4)$	3, 5, 6	3, 0	2
$P_5\ (x_5,\ y_5)$	2, 4, 0	0, 0	3
$P_6\ (x_6,\ y_6)$	4, 7, 0	3, 0	1
$P_7\ (x_7,\ y_7)$	6, 8, 0	1, 0	1
$P_8\ (x_8,\ y_8)$	7, 9, 11	1, 0	2
$P_9\ (x_9,\ y_9)$	1, 8, 10	1, 0	2
$P_{10}\ (x_{10},\ y_{10})$	9, 11, 0	0, 0	2
$P_{11}\ (x_{11},\ y_{11})$	8, 10, 0	0, 0	2

Fig. 5 Input format for 2-dimensional 3-views (Top view used)

where c = any values

n = number of points in each view.

Therefore, the elements of P'_{xyz} in Eq. (4) becomes $(x_i,\ y_j,\ z_k)$ which is a combination of i, j, k elements in Eq. (12) and satisfies the conditions of $x_i = x_k$, $y_i = y_j$, and $z_j = z_k$

3. Transformation to edge lines.

L'_{xyz} in Eq. (5) is determined as follows. Let $(P_{xyz})_1$, and $(P_{xyz})_2$ be different points of P'_{xyz}, $(P_{xy})_1$, $(P_{yz})_1$, $(P_{zx})_1$ be points of views which are transformed to $(P_{xyz})_1$, and $(P_{xy})_2$, $(P_{yz})_2$, $(P_{zx})_2$ be points of views which are transformed to $(P_{xyz})_2$. Then, when $(P_{xy})_1$, and $(P_{xy})_2$ in xy plane satisfy at least one of following three conditions, the index A is given a value of 1.

1) $(P_{xy})_1$ and $(P_{xy})_2$ are same node

2) $(P_{xy})_1$ and $(P_{xy})_2$ are adjacent points

3) $(P_{xy})_1$ and $(P_{xy})_2$ belong to same group

Similary, when $(P_{yz})_1$ and $(P_{yz})_2$ in yz plane and $(P_{zx})_1$ and $(P_{zx})_2$ in zx plane satisfy at least one of the above three conditions, indices B and C are given a value of 1 respectively and $(P_{xyz})'_1$, and $(P_{xyz})'_2$ are connected together only if each value of A, B and C is 1.

4. Ghost figure elimination.

Let H'_{cxyz} be the number of vertices connected to a vertex. Then, the process in Eq. (6) to eliminate ghost figure is as below.

1) Verify all values of H'_{cxyz}

2) If all values of H'_{cxyz} are 3, the elimination process is completed.

3) If all values of H_{cxyz} are greater than 3, it means the existence of ghost vertices. Thus, eliminate P'_{xyz} corresponding to the maximum H'_{cxyz}, calculate all values of H'_{cxyz} and return to 1).

4) If a value of H'_{cxyz} is less than 3, a ghost edge exists.

Accordingly, eliminate all edges including ghost edge, connect real edges, compute all values of H'_{cxyz} again, and return to 1).

5. Surface Construction.

The set of $\underset{\sim}{K}_{PL}$ in Eq. (7), which is necessary condition for constructing the surfaces of 3-D solid, is found from the surface properties described below.

i) a surface consists of more than 3 edges and has a closed boundary

ii) a boundary edge exists between two surfaces and the direction of two edge row is opposite each other.

Then, the surface construction is performed as follows.

1) Find all surfaces satisfying the property i)

2) Choose a reference surface

3) Rearrange so that the direction of edge row. $\underset{\sim}{R}_{PL}$ of reference surface is always counterclockwise. In this case, the visual point is set at the outside of surface.

4) Taking the relationship with the reference surface into account, rearrange the remaining surfaces so that the $\underset{\sim}{R}_{PL}$ of those surfaces satisfy the property ii)

5) Return to step 2) to rearrange other surfaces.

6) Complete the procedure if all surfaces are rearranged.

Finally, a 3-D solid is represented as in Eq. (13) by family sets of the vertices, $\underset{\sim}{P}_{xyz}$ and the rows $\underset{\sim}{R}_{PL}$ which show the boundary of faces.

$$\underset{\sim}{V} = \underset{\sim}{S}_{xyz} = \{\underset{\sim}{R}_{PL}, \underset{\sim}{P}_{xyz}\} \cdots\cdots\cdots\cdots\cdots\cdots (13)$$

6. Hidden lines/surfaces removal and visualizing on screen.

To recognize correctly the 3-D solid object drawn on computer screen, the hidden lines and surfaces are removed by using the vector operation.

Let $\underset{\sim}{N}$ be a normal vector of surface and $\underset{\sim}{W}$ a visual vector. Then, if $\underset{\sim}{N} \cdot \underset{\sim}{W}$ is negative the surface is invisible and if positive, it is visible.

For curved surfaces, the similar vector method can be used. First of all the coordinates of curved surface are obtained from B-spline curve, which is directly found from the input data of the surfaces. If one plane of curved surfaces is visible, the other plane becomes invisible as shown in Fig. 6.

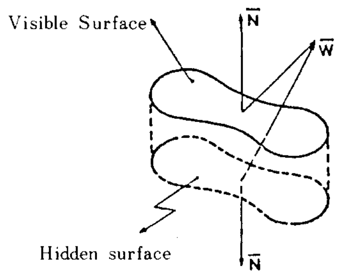


Fig. 6 Criterion for the visible and hidden surface

Accordingly, the visible and invisible planes are distinguished by the sign of inner product $\underset{\sim}{N} \cdot \underset{\sim}{W}$. The visibility of curved edge line in curved surface is determined as shown in Fig. 7 by dividing

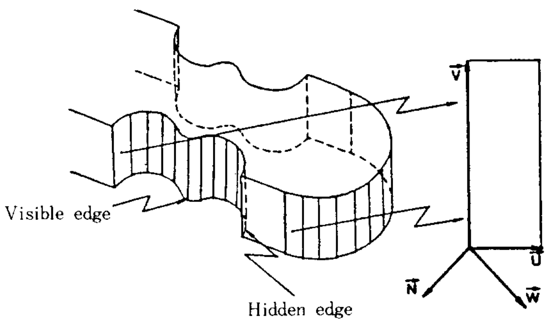


Fig. 7 Criterion for the visible and hidden edges of an arc and curved surface

the surface into several rectangles and judging the sign of $\underset{\sim}{N} \cdot \underset{\sim}{W}$.

The 3-D solid is drawn on 2 dimensional screen by perspective representation through the coordinates transformation procedure as follows.

Rectangular coordinates(x, y, z)→Spherical coordinates(x′, y′, z′) Eye coornates(x_e, y_e, z_e)→Screen coordinates(x_s, y_s).

TOOL PATH GENERATION

The tool path generation to machine the designed object by NC machine is started from drawing a cross section of 3-D solid. The cross section is also necessary to recognize the internal shape of the solid and to calculate the volume of material occupied by the solid.

To obtain the cross section, a 3-D solid is first cut at arbitrary direction by a cutting plane of Ax+By+Cz=D in which A, B, C and D are constant and determined by the cutting direction or angle. As the 3-D solid is constructed by the rectangular surfaces consisting of piecewise straight line, the edges of each surface will have in space the straight line equation defined as

$$\frac{x-x_1}{x_2-x_1} = \frac{y-y_1}{y_2-y_1} = \frac{z-z_1}{z_2-z_1} = t \quad \cdots\cdots\cdots\cdots (14)$$

where t is constant.

The cross section is therefore found by calculating the intersecting points between the cutting plane and the edge of Eq. (14), and connecting the points together.

When the designed object is machined by a tool having a larger radius than the smallest radius of the object curvature, the overcut or interference phenomenon happens and eventually ruins the object. The radius of cutting tool should be thus found by calculating the radius of curvature of curve to be cut. The three points P, Q, R on the curve are initially chosen as shown in Fig. 8 and vectors a and b are determined as shown. Then, the radius of curvature at Q is obtained by.

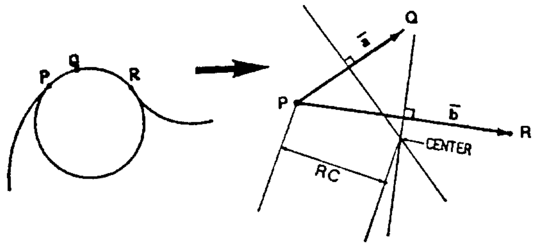


Fig. 8 Radius of curvature of curve

$$RC = \frac{|\underset{\sim}{a}|\ |\underset{\sim}{b}|\ |\underset{\sim}{a}-\underset{\sim}{b}|}{2\,|\underset{\sim}{a}\times\underset{\sim}{b}|} \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (15)$$

The smallest radius of curvature is chosen from the results of Eq. (15) and the radius of cutting tool is decided so as to be less than the smallest radius of curvature.

The tool path can now be generated according to the surface characteristics of the object. If the surface is ruled one and comprises B-spline curve, the tool path will be on the distance of tool radius apart from a point on the spline. Thus, if the B-spline function is S(x) and the radius of tool is r, the position of cutting tool at (x_i, S(x_i)) of spline curve can be found as shown in Fig. 9 and expressed as

$$\begin{aligned} x_P &= x_1 + r\cos\theta \\ y_P &= s(x_1) + r\sin\theta \end{aligned} \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (16)$$

where θ is the angle of normal direction at (x_i, S(x_i)).

If the surface is sculptured or free surface, the machining should be done by 3 dimensional ball type and milling cutter. In this case, the

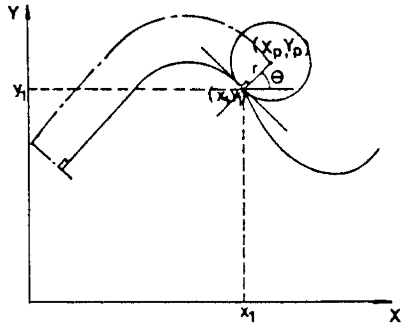


Fig. 9 Offset spline curve

path of cutter tip $\underline{r}_c$ can be found from Fig. 10 and represented as

$$\underline{r}_c = \underline{r}(u_o,\ v) + R(\underline{n} - \underline{u}) \quad \cdots\cdots (17)$$

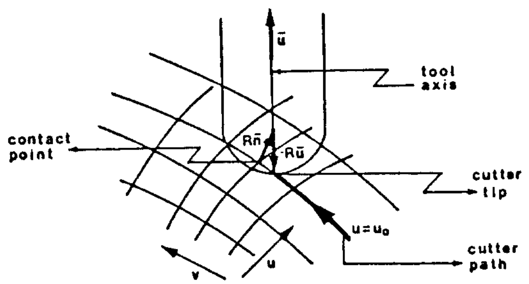


Fig. 10 Cutter path of 3-dimensional ball endmill

in which u_o = path parameter at cutter contact point

$\underline{n}$ = unit normal vector at u

$\underline{u}$ = unit vector of cutter axis

R = radius of cutter

$\underline{r}(u_o,\ v)$ = position vector of contact point

$R(\underline{n} - \underline{u})$ = position vector from contact point to cutter tip

The process taken for automatic conversion and tool path generation described so far can be illustrated by the flow chart indicated in Fig. 11.

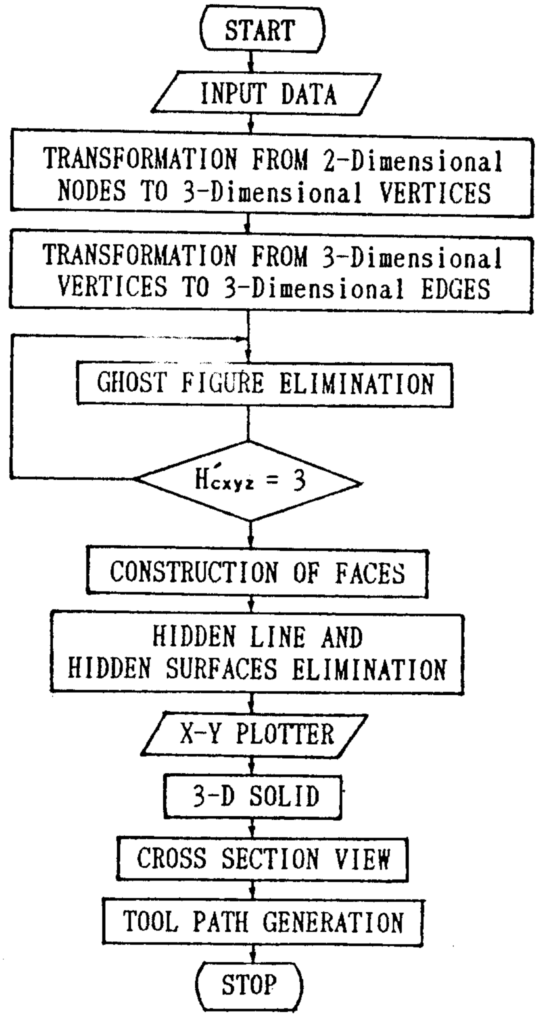


Fig. 11 Flow chart of program for generating 3-dimensional solid and tool path from 2-dimensional 3-views

APPLICATION

The algorithms described above were coded into turbo-PASCAL and the resulting computer program was applied to two typical examples.

Example 1

The first example to which the computer program was applied is an object having an arc and arbitrary curve in front view. View as shown in Fig. 12 (a) and the tool path to cut the object is given in Fig. 12 (b). Table 1 shows iuput data for drawing the 3-D solid object form the 2-D three views of design drawing and describes the relationships between nodes and information necessary for the automatic conversion.

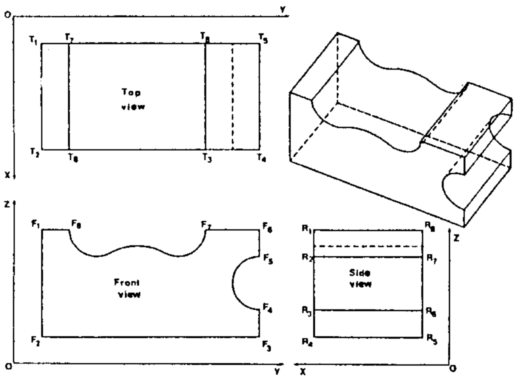


Fig. 12 (a) Example of an object having an arc and arbitrary curve in front view

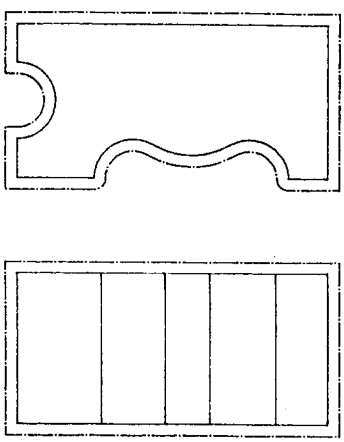


Fig. 12 (b) Tool path of the Fig. 12-(a)

Table1 Input data of Fig. 12-(a)

View point coordinate:	Rho(ρ) = 500				
	Distance = 450				
	Theta(θ) = 0.5 (rad)				
	Phi(ϕ) = 0.8 (rad)				
Straight line data input					
Node NO.	Top view : 8 , Front view : 8 , Side view : 8				
	node number	coordinate value	adjacent node NO.	group number	node nature
Top view	T_1	1 , 1	2, 7, 0	1, 2	standard
	T_2	5 , 1	1, 8, 0	2, 3	standard
	T_3	5 , 7	8, 6, 4	3, 5	tangential
	T_4	5 , 9	3, 5, 0	3, 4	standard
	T_5	1 , 9	4, 6, 0	4, 1	standard
	T_6	1 , 7	3, 5, 7	1, 5	tangential
	T_7	1 , 2	1, 8, 6	1, 6	tangential
	T_8	5 , 2	2, 7, 3	3, 6	tangential
Front view	F_1	1 , 5	2, 8, 0	1, 2	standard
	F_2	1 , 1	1, 3, 0	2, 3	standard
	F_3	9 , 1	2, 4, 0	3, 4	standard
	F_4	9 , 2	3, 5, 0	4, 5	tangential
	F_5	9 , 4	4, 6, 0	5, 6	tangential
	F_6	9 , 5	7, 5, 0	7, 6	standard
	F_7	7 , 5	6, 8, 0	7, 0	tangential
	F_8	2 , 5	1, 7, 0	1, 0	tangential
Right side view	R_1	5 , 5	2, 8, 0	1, 2	standard
	R_2	5 , 4	1, 7, 3	2, 6	standard
	R_3	5 , 2	2, 6, 4	2, 5	standard
	R_4	5 , 1	3, 5, 0	2, 3	standard
	R_5	1 , 1	4, 6, 0	3, 4	standard
	R_6	1 , 2	3, 5, 7	4, 5	standard
	R_7	1 , 4	6, 2, 8	6, 4	standard
	R_8	1 , 5	1, 7, 0	4, 1	standard

Curve data input	
Circle No.	Top view : 0 , Front view : 0 , Side view : 0
	radius of circle:
	coordinate of center:
	coordinate of top and bottom surface of cylinder
	nature of cylinder: 1 for internal and
	2 for external cylinder

Arc No.	Top view : 0 , Front view : 1 , Side view : 0
	radius of arc : 1 coordinate of center : (9 , 3) starting angle (θ_1) and construction angle (θ_2) : 90 , 270 (degree) coordinate of top and bottom surface of cylinder : 5 , 1 nature of cylinder : 1
Arbitrary curve	Top view : 0 , Front view : 1 , Side view : 0
	knot number : 24 nature of arbitrary curve : arbitrary number coordinate of top and bottom surface : 5 , 1 coordinate value of data point : (2.00, 5.00) (2.00, 4.91) (2.02, 4.83) (2.06, 4.66) (2.13, 4.50) (2.36, 4.23) (2.66, 4.06) (3.00, 4.00) (3.34, 4.06) (3.50, 4.13) (3.82, 4.28) (4.50, 4.40) (5.18, 4.28) (5.50, 4.13) (5.66, 4.06) (6.00, 4.00) (6.34, 4.06) (6.50, 4.13) (6.64, 4,23) (6.82, 4.43) (6.91, 4.58) (6.94, 4,66) (6.99, 4.83) (7.00, 5.00)

Example 2

The same program was also applied to an object having arc and arbitrary curve in top and front views as shown in Fig. 13 (a) and its results on the cross section view and tool path are given in Fig. 13 (b) and (c) respectively.

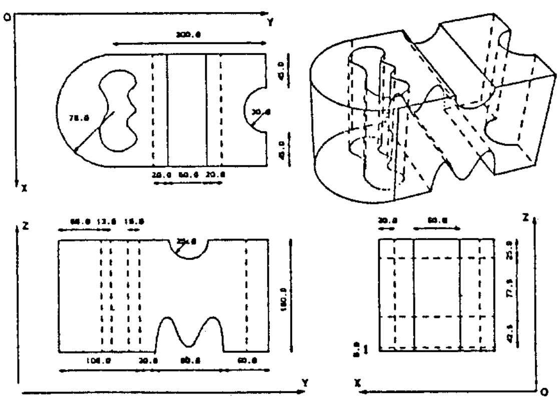


Fig. 13-(a) An object having arc and arbitrary curves in top and front views

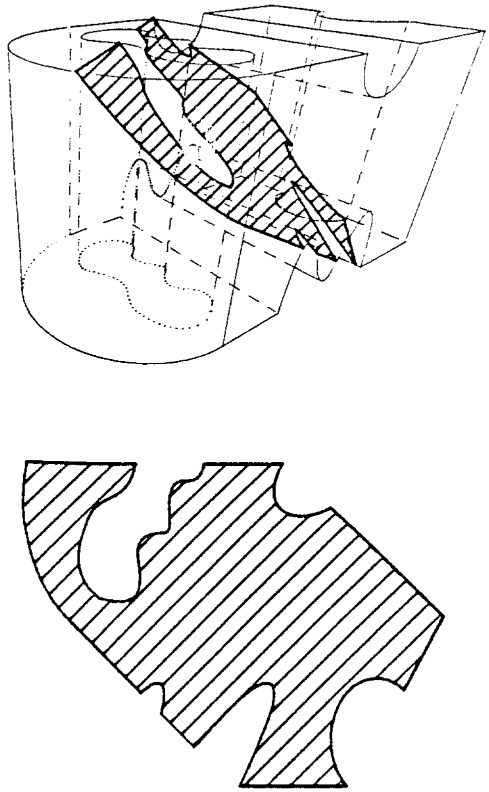


Fig. 13-(b) Section view of Fig. 13-(a) and its normal view

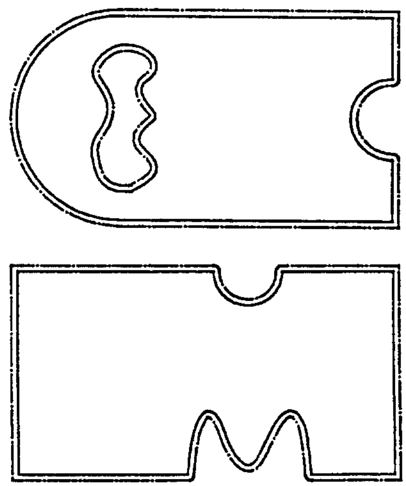


Fig. 13-(c) Tool path of Fig. 13-(a)

CONCLUSION

1. By using a micro based personal computer, it was possible to automatically visualize the 3-D solid object represented in the 2-D three views of design drawing.
2. As the cutter path of NC machine can be generated from the 3-D solid representation through drawing cross sections, the foundation for CAD/CAM integration in which the design drawing from CAD can be connected directly to CAM for its manufacturing was established making the productivity improvement possible.

參 考 文 獻

1. Idesawa, M., "A system to generate a solid figure from a three view", Bulletin of the JSME, Vol. 16, No. 92, Feb. 1973, pp. 216-225
2. Chang, K. S. and Kim, H. R., "Micro-Computer Aided Automatic Conversion of 2-D Three Views to a 3-D Solid", J. of Eng. Research Institute, Yonsei University, Vol. 18, No. 2(25), 1986, pp. 241-247
3. Yoon, T. S. and Kim, H. R., "Automatic Conversion from Design Drawing Including Circles to 3-D Objects", Proc. of KSME, spring 1987.
4. Kim, Y. K. and Kim, H. R., "PC Aided Conversion of Design Drawing for CAD/CAM Integration", Computer Aided Design, Vol. 15, No. 2, 1983, pp. 59-64
5. Aldefeld, B., "On Automatic Recognition of 3D Structures from 2D Representations", Computer Graphics, Vol. 17, No. 3, 1983, pp. 243-252
6. Sakurai, H., and Gossard, D. C., "Solid Model Input through Orthographic Views", Computer Graphics, Vol. 17, No. 3, 1983, pp. 243-252
7. Wang, C. Y., "Shape Curve And Classification of the Parametric B-spline Cubic Curve", IPC Bussiness Press, Vol. 13, No. 4, 1981, pp. 199-206