AN ALGORITHM FOR PLOTTING COMPUTERIZED HIDDEN-LINE PERSPECTIVES *

İbrahim CANBULAT
Mustafa PULTAR

* This article is based on an unpublished Master Thesis written by İ. CANBULAT - "Theoretical Bases and *Program Description of a Computerized* Method for Perspective Drawing", unpublished Master Thesis, Middle East Technical University, Faculty of Architecture, Department of Architecture, Ankara, 1974 and supervised by Dr. M. Pultar.

Within the general classes of communication media, there are those specific ones that will match well defined communication need. The architect must be assured that the alternate costs and flow times are favorable in terms of the total communication value of the medium. Architect's disciplinary communication media are established on graphical symbols such as plans, sections, elevations,perspectives etc.

The commun ication media of computers are entirely different from those of the architects. This communication gap between architects and the computer has a detrimental effect on the use of computers by architects. If a graphical information medium is designed between the architects and the computer the architect will, then best exploit the possibilities of the computer.

Perspective drawings are the most favorable graphical instruments for the presentation of the three dimensional objects.[1] They are helpful to simulate the perceiving of space. The architect can evaluate the spatious quality of his/her design proposal by checking the drawings. He can also use the drawings to present his design to the client or the partners or any other interested group. Besides these usages, the computer can give outputs for some programs as perspective drawings if programmed so.

1. W.A FETTER, *Computer Graphics in Communication*, New York: McGraw-Hill Book Company, 1964.

There is a number of computer programs on perspective drawings already prepared.[2] Some of the programs can also be called by other technical programs to transfer output data graphically.

2. C.STEWARD, E.TEYCHOLTZ and R.LEE, *CAP: Computer Architecture Programs*,Cambridge, Massachusetts: Center for Enviromental Research, 1972.

In the following paragraphs, the process of defining the three dimensional objects into computer, the model of the perspective projection and an outstanding problem of perspective projection which is called *hidden-line* problem are discussed in detail, and some sample outputs are given. In the last section the program generating the model is described, and the physical characteristics of the program are given.

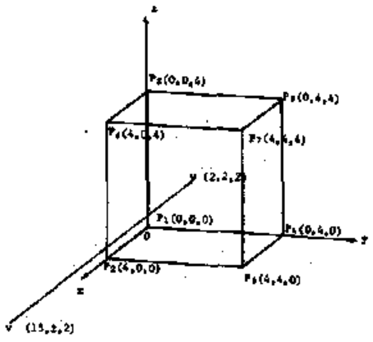


Figure 1. Topology of a cube, drafted by computer.

GEOMETRIC DEFINITION OF OBJECTS

Perspective projection deals with three dimensional objects. As a restriction in the model, the objects are assumed to be defined by convex tetragones which are portions of planes. In terms of architectural usage, the tetragonal face assumption is not problematic, for great majority of buildings are constructed to have rectangular faces. These are not limited only to the facades but include other building elements such as columns, windows, doors, etc. The tetragones are defined by four vertices; this definition is necessary for the opaque surfaces. The object is defined and plotted by two different types of lines: the edges show the boundary conditions; the line segments elaborate the details on the facades of the object. Geometrically, edges and the line segments are defined by the coordinates of the two end points or vertices.

Coordinates of the vertices, $\vec{P}_i$, are in the right-handed cartesian coordinate system. The vertices are addressed by index numbers, i, which are sufficient to define the line segments. A list of edges joining the vertices (J_1, J_2) is also part of the data. The four index numbers (J_1, J_2, J_3, J_4) of the vertices of a face must be given in clock-wise or counterclock-wise order, regarding of the sequence of numbering of the indices.[3]

PERSPECTIVE PROJECTION

In perspective projection, an object in three dimensions is projected onto a plane - *picture plane* - when such an object is viewed from a stationery point. The projection is the two dimensional image of the object. A number of mathematical methods exist for computing perspective projection.[4] In the following paragraphs another approach to the model of the perspective projection is given.

Figure 2 gives the general scheme of the perspective projection. $\vec{V}$ and $\vec{W}$ are the position vectors of the *viewing point* and the *viewed point* respectively. $\vec{V}$ and $\vec{W}$ are the basic parameters of the projection and must be given additionally to the geometrical data. The vector, $\vec{v}$, gives the viewing direction,

$$\vec{v} = \vec{W} - \vec{V} \qquad (1)$$

The viewing direction will be perpendicular to the picture plane. In other words $\vec{v}$ is the normal of the picture plane, in which the unit vector is $\vec{u}$,

$$\vec{u} = \vec{v}/|\vec{v}|. \qquad (2)$$

In order to construct a coordinate system that will generate the perspective projections, one of the axes is chosen parallel to the horizontal plane, in other words xy-plane. Actually, the horizontal plane is a very powerful referance for ordinary seeing. The first base vector, $\vec{a}$ is on the horizontal plane and orthogonal to $\vec{u}$, thus

$$\vec{u} \cdot \vec{a} = 0 . \qquad (3)$$

Defining $\vec{u} = (u_1, u_2, u_3)$ and $\vec{a} = (a_1, a_2, 0)$, from Equation 3,

$$u_1 a_1 + u_2 a_2 = 0. \qquad (4)$$

Being a unit base vector:

$$|\vec{a}| = 1, \qquad (5)$$

3. Data of a cube (Figure 1) is as follows:

Vertices:

1	0.	0.	0.
2	4.	0.	0.
3	4.	4.	0.
4	0.	4.	0.
5	0.	0.	4.
6	4.	0.	4.
7	4.	4.	4.
8	0.	4.	4.

Edges:

1	2
2	3
3	4
4	1
5	6
6	7
7	8
8	;
1	5
2	6
3	7
4	8

Faces:

1	2	3	4
5	6	7	8
1	2	6	5
2	3	7	6
3	4	8	7
4	1	5	8

Viewing Point

15. 2. 2.

Viewed Point

2. 2. 2.

4. M.KRAMPEN and P.SEITZ (Eds.), *Design and Planning 2*, New York: Hasting House, Publishers, 1967, p.167. P.P.LOUTREL, A Solution to the Hidden-Line problem for Computer-Draw Polyhedra, *IEEE Transactions on Computers*, v. C-19, n.3, 1970, pp.205-207.

and

$$(a_1^2 + a_2^2)^{\frac{1}{2}} = 1. \qquad (6)$$

Simultaneous solution to Equation 4 and 6 gives

$$a_1 = u_2 / (u_1^2 + u_2^2)^{\frac{1}{2}} \qquad (7)$$

and

$$a_2 = -u_1 a_1 / u_2. \qquad (8)$$

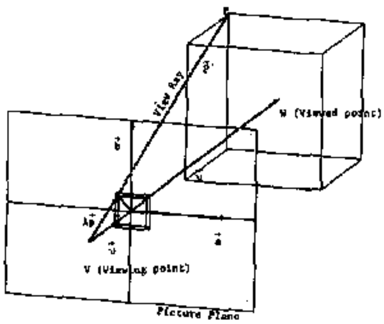


Figure 2. Scheme of the perspective projection, drafted by computer.

Since the second base vector, $\vec{b}$ of the new coordinate system is orthogonal to u and a;

$$\vec{b} = \vec{u} \text{ and } \vec{a}; \qquad (9)$$

hence,

$$\vec{b} = \begin{Bmatrix} -u_3 a_2 \\ u_3 a_1 \\ u_1 a_2 - u_2 a_1 \end{Bmatrix} \qquad (10)$$

We can define any point on the projection plane by $\vec{a}$ and $\vec{b}$.[5] $\vec{P}$ is the position vector of any vertex. $\vec{p}$ represents the *view ray* coming from $\vec{P}$ to $\vec{V}$,

$$\vec{p} = \vec{P} - \vec{V}. \qquad (11)$$

A vector which is collinear with $\vec{p}$ and whose projection on $\vec{v}$ is $\vec{u}$ satisfies the relation,

$$\lambda\vec{p} \cdot \vec{u} = 1. \qquad (12)$$

Deriving Equation 12 for scaler λ gives

$$\lambda = 1/ \vec{p} \cdot \vec{u}. \qquad (13)$$

The position vector $\vec{q}$ of the $\vec{p}$, due to the bases $\vec{a}$ and $\vec{b}$ that define the picture plane is

$$\vec{q} = \begin{Bmatrix} \lambda\vec{p} \cdot \vec{a} \\ \lambda\vec{p} \cdot \vec{b} \end{Bmatrix} \qquad (14)$$

As can be seen from Equation 14, $\vec{q}$ has two components which are the coordinates of the new coordinate system.[6]

5. If $u_2 = 0$ Equation 8 is undefined. This is the case of the viewing direction being in the direction of the y-axis. The model is assumed to be conditioned to accept $\vec{a} = \vec{j}$ where $\vec{j}$ is the second base vector of the original coordinate system.

6. When the view ray $\vec{p}$ is orthogonal to $\vec{u}$ the denominator vanishes. These points are important if this point is connected by a line segment to another point which is visible. With an approximation such points are assumed to be on the picture plane and their perspective projection coordinates are $q_1 = \vec{p} \cdot \vec{a}$ and $q_2 = \vec{p} \cdot \vec{b}$.

HIDDEN LINE ALGORITHM

Perspectives that are perceived in three dimensions in drawings are actually, two dimensional representations. The elimination of the unseen lines of an opaque object intensifies the perception of three dimensions.[7] The *hidden-line* algorithm comprises the obscuring of certain portions of line segments which are masked by the opaque faces of the object.

To incorporate the opaqueness and masking of the invisible line segments in the model is not as easy as the actual physical phenomenon. There are some algorithms to this challenging problem. In the following paragraphs the authers' attemp to the hidden-line problem is given.

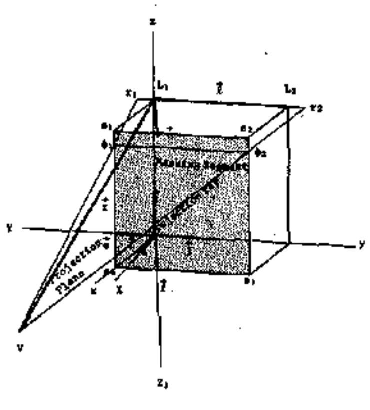


Figure 3. Scheme of the hidden-line model, drafted by computer

The model is constructed to check the alternative positions under the perspective projection of one face and one edge of the object each time.

In Figure 3, $\vec{L}_1$ and $\vec{L}_2$ are the position vectors of the end

7. D.P.GREENBERG, Computer Graphics in Architecture, *Scientific American*, v.230, n.5, 1974, pp.93-106.

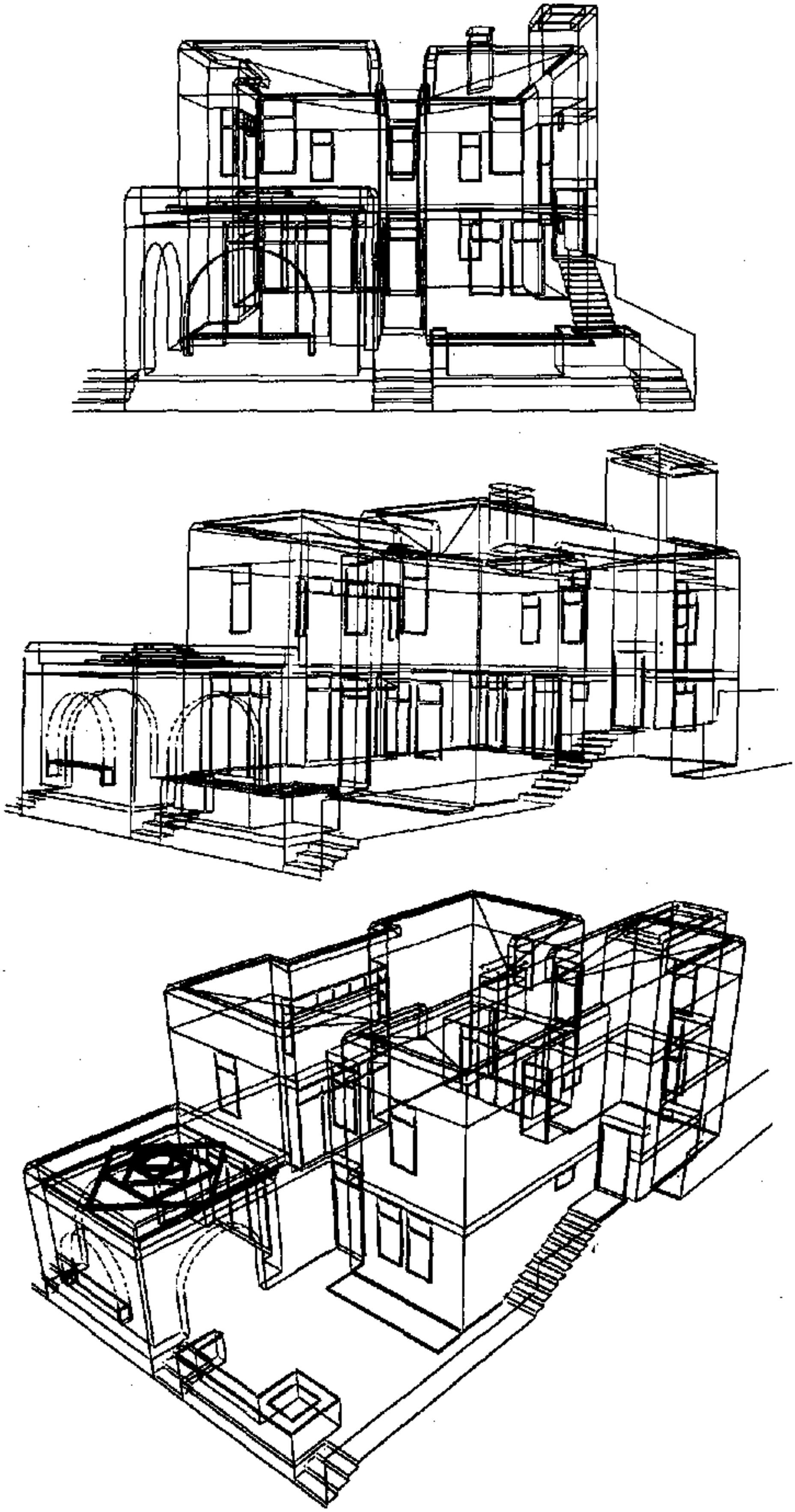


Figure 4. Perspective Drawings of Güleryüz Residence. (Architect Y.Yavuz)

points of an edge, $\vec{S}_1$, $\vec{S}_2$, $\vec{S}_3$ and $\vec{S}_4$ are the four position vectors of the vertices of any face.

To overcome the computational difficulties generated by the three dimensional geometry, the projection is transformed into planer geometry in the model. In reality the *masking segment* of the face is the intersection between the face and the plane defined by the two view rays joining $\vec{L}_1$, and $\vec{L}_2$ to $\vec{V}$. It is obvious that $\vec{V}$, $\vec{L}_1$,$\vec{L}_2$ and the masking segment are on the same plane. This is referred to as the *projection plane*.

Base vectors need to be defined accordingly, in order to transfer these points into a new coordinate system. In the new coordinate system the projection plane is chosen to be parallel to the new XY-plane. Also, choosing the new Y-axis parallel to the line segment will eliminate operational difficulties. This way, one of the base vectors is taken in the direction of the line segment.

$$\vec{e} = \vec{l}/|\vec{l}| \, , \tag{15}$$

where

$$\vec{l} = \vec{L}_2 - \vec{L}_1 . \tag{16}$$

To find the normal of the projection plane,

$$\vec{n} = \vec{r} \times \vec{l} \, ,$$

where

$$\vec{r} = \vec{V} - \vec{L}_1 , \tag{17}$$

and the second base vector

$$\vec{f} = \vec{n}/ |\vec{n}| \, . \tag{18}$$

The third base vector $\vec{d}$ is orthogonal to $\vec{e}$ and $\vec{f}$ so that

$$\vec{d} = \vec{e} \times \vec{f} \, . \tag{19}$$

$[T]$ is the linear transformation matrix to transform the points into the new coordinate system;

$$[T] = \begin{bmatrix} \vec{i}\vec{d} & \vec{j}\vec{d} & \vec{k}\vec{d} \\ \vec{i}\vec{e} & \vec{j}\vec{e} & \vec{k}\vec{e} \\ \vec{i}\vec{f} & \vec{j}\vec{f} & \vec{k}\vec{f} \end{bmatrix} \tag{20}$$

in which $\vec{i}$, $\vec{j}$ and $\vec{k}$ are the bases of the original coordinate system, where

$$\vec{i} = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix} , \quad \vec{j} = \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix} , \quad \vec{k} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{21}$$

and

$$\vec{d} = \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} , \quad \vec{e} = \begin{Bmatrix} e_1 \\ e_2 \\ e_3 \end{Bmatrix} , \quad \vec{f} = \begin{Bmatrix} f_1 \\ f_2 \\ f_3 \end{Bmatrix} . \tag{22}$$

Computation of the transformation matrix $[T]$ for the values of $\vec{i}$, $\vec{j}$, $\vec{k}$ and $\vec{d}$, $\vec{e}$, $\vec{f}$ is achieved by

$$[T] = \begin{bmatrix} d_1 & d_2 & d_3 \\ e_1 & e_2 & e_3 \\ f_1 & f_2 & f_3 \end{bmatrix} . \tag{23}$$

To transform any point into a new coordinate system we have to take

$$\vec{P}' = \left[T\right] \vec{P} . \qquad (24)$$

In Equation 24, $\vec{P}$ represents any point in the original coordinate system and $\vec{P}'$ the transformed point. Note that the transformed points will hereafter, be represented by the same symbols, unless otherwise stated.

The general form of the equation of a line passing through a given point $\vec{P}$ (x_0, y_0, z_0) and parallel to a given nonzero vector is

$$\frac{x - x_0}{A} = \frac{y - y_0}{B} = \frac{z - z_0}{C} = t, \qquad (25)$$

where A, B, and C are the components of the vector.

In order to find the end points of the masking segment of the face, the intersections of the edges of the face and the projection plane must be computed. We know that these intersection points have the same Z value which is equal to the distance from the origin to the projection plane. From Equation 25, we have

$$t = \frac{z - z_0}{C} \qquad (26)$$

and we can easily compute X and Y values as

$$X = t\, s_{12} + (1 - t)\, s_{11} \qquad (27)$$

$$Y = t\, s_{22} + (1 - t)\, s_{21} \qquad (28)$$

where s_{ij} $(i = 1,3)$ are the components of $\vec{S}_j$ the vertices of the face.

In the masking situation, there should always be two intersection points between the edges of the tatragonal face and the projection plane. If parameter t is less than 0 or greater than 1 in Equation 26 for two edges of the face, the face and the projection place have no intersection, in other words, the face does not mask the line segment.[8] Another control for the masking condition is checking the triple positions of the viewing point, the face and the line segment.[9] If the face and the line segment are on the either sides of the viewing point or the face is behind the line segment referring to the viewing point, the line segment could not be masked. If so, comparison starts from the very beginning of the algorithm between the line segment and a new face.

8. If $0 \leq t \leq 1$, $\vec{\Phi}$ is the convex combination of S_1 and S_2

9. Note that these comparisons are made on the X coordinates of the viewing point, the end points of line segment and the masking segment.

When the face and the projection plane intersects and the masking segment is between the viewing point and the line segment the two end points are defined as $\vec{\Phi}$ and $\vec{\Phi}$. We can define the equation of the lines passing through $\vec{V}$ and the points $\vec{\Phi}_1$ and $\vec{\Phi}_2$ respectively. These two lines are called *projection rays*.

Solving the equations of projection rays, by substituting X coordinate of the line segment, gives the coordinates of the intersections, $\vec{\psi}_1$, $\vec{\psi}_2$, of the line segment and the projection rays. The new line segment defined by $\vec{\psi}_1$ and $\vec{\psi}_2$ is the perspective projection of the face onto the line defined by $\vec{L}_1$ and $\vec{L}_2$.

There are three possible alternatives due to the positions of

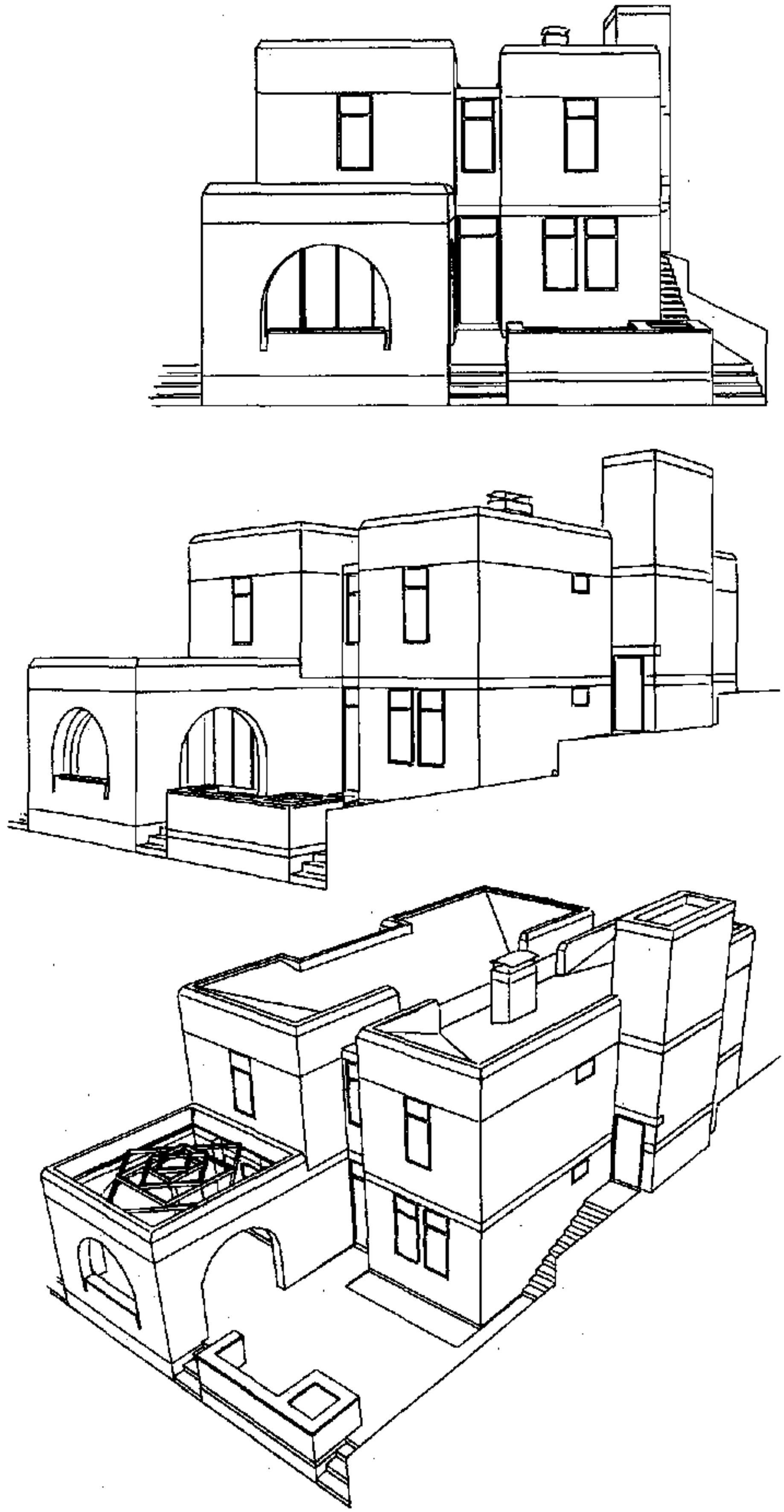


Figure 5. Perspective drawings, GÜleryüzResidence (Architect Y.Yavuz) after hidden-lines removed.

the line segment and the projection of the face:
i. The line segment is totally masked;
ii. A portion of the line segment is masked;
iii. The line segment is not masked.[10]
If the line segment is not totally obscured, the end points of the unmasked portion is transformed back to the original coordinate system by the relation,

$$\vec{P} = T\ \vec{P}'.^{11} \qquad (29)$$

Furthermore, the unmasked or partially masked line segments are compared with a new opaque face and the algorithm starts from the beginning. On the other hand, if the line segment is masked, a new line segment and face combination is searched algorithmically from the beginning. The computation is repeated until all of line segments are searched.

10. The positions of the line segments and the projection are understood by comparing the Y components of the end points.

11. Being an orthogonal transformation $[T] = [T]^t = [T]^{-1}$.

12. I.CANBULAT, "Theoretical Bases and Program Description of a Computerized Method for Perspective Drawing", unpublished Masters Thesis, Middle East Technical University, Faculty of Architecture, Department of Architecture, Ankara, 1974.

THE PROGRAM

A program[12] is written to compute the perspective model for a set of input parameters as viewing direction, viewing cone angle and drawing scale. Inputs of the program are defined in the section labeled "Geometric Definition of Objects". Program controls the data and searches for the undefined and duplicate points. The user can produce optional stereoscopic drawings couple (Figure 6) and control hidden line elimination procedure.

Program can eliminate the line segments out of the viewing cone defined by an angle which is part of the input. The algorithm of the viewing cone problem is not discussed here. The user can control the size and effect of the drawings by double control on the viewing direction and the viewing cone angle very similar to the optical phenomenon. One can get some tele or wide angle effect by using these parameters. The program is also capable of solving the *clipping problem*. The clipping problem is caused by the line segments joining the points on either sides of the observer, one being in the front and one being on the back. It is very obvious that the perspective projection model also defined for the points behind of the observer is contradictory to the physical phenomenon. The program computes the portion of the line segments which are in the vision of the observer. The model of clipping problem is out of the interest of this article.

The plotters are very slow units comparing by the other units of a computer system. When plotters are used on-line, the plotting time is very important. The program is also comprises the plotting time. Heuristically, the plotting time is minimized by minimizing the number of pen controls and pen-up movements of the plotter.

The program has 650 FORTRAN statements. The main program calls a dummy subroutine is coded to produce card outputs which will be used as an input for the plotter.

The program is run on IBM 370/45 system and as it is mentioned earlier, a card output is received. Cards which code the x, y coordinates and pen control parameter, are used as input for Interdata 7/32 which controls Calcomp 565 Plotter.

The building which is shown in Figure 3 and Figure 5 has 980 points, 1099 line segments and 344 opaque surfaces. The

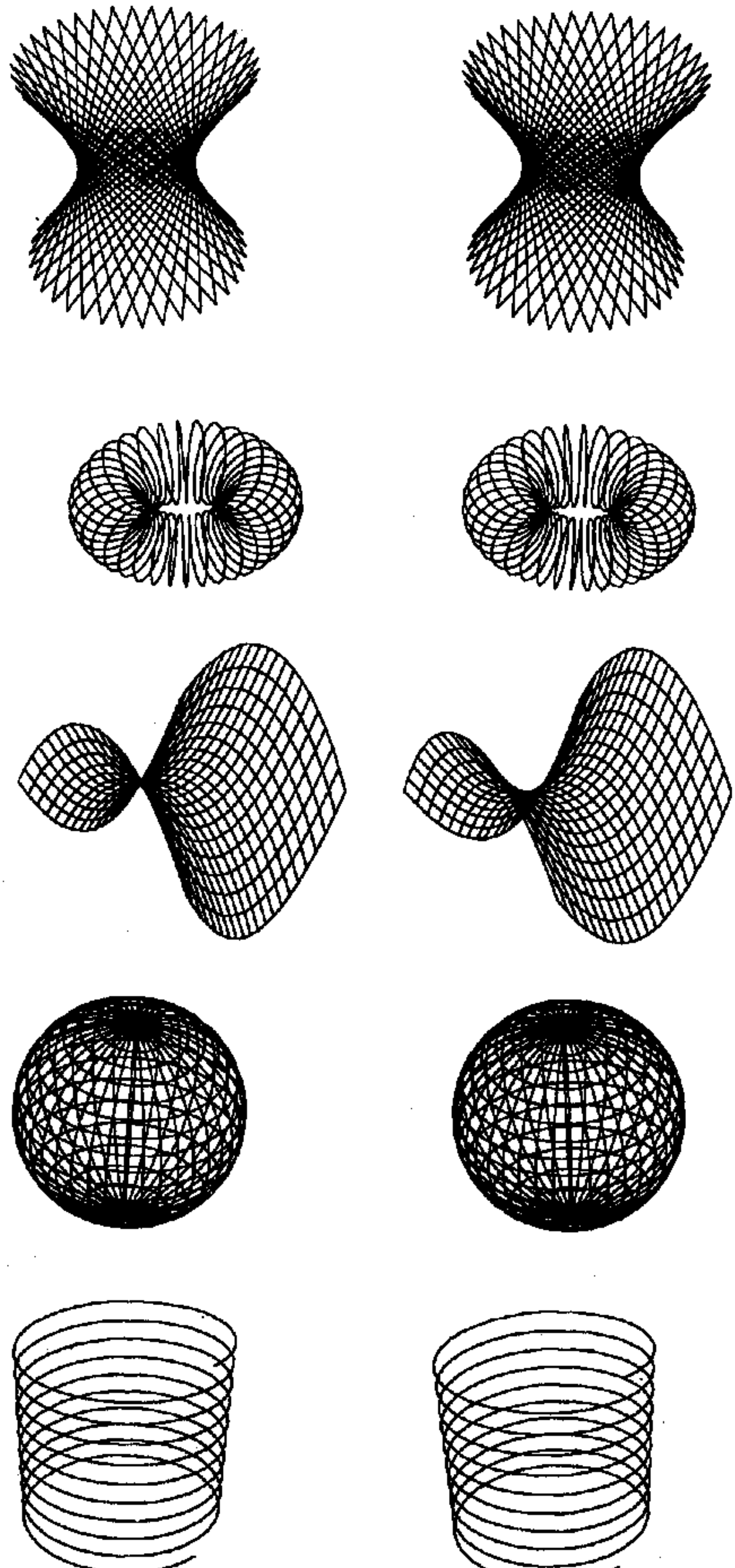


Figure 6. Stereoscopic drawings.

A. Hyperboloid

B. Torus

C. Hyperbolic paraboloid

D. Sphere

E. Helix

To perceive the three dimensions the left eye must permit to see only the left view and the right eye only the right view.

execution time is changed up to the viewing direction from 5 to 8 CPU minutes, including input control control, hidden-line, clipping and viewing cone eliminations and shortest joining path solutions.

BİLGİSAYARLA PERSPEKTİF ÇİZİM

ÖZET

Mimarların imgeler üzerine kurulu dili ile bilgisayarların sayısal dili arasındaki karşıtlık mimarın bilgisayarları kullanmasında sorunlar yaratmaktadır. Bilgisayar ürünü çizgeler bu açıdan yararlı olmaktadırlar. Mimar, bilgisayar çizgelerinden, tasarımını değerlendirdiği gibi başka teknik programların da çıktılarını çizge olarak kolaylıkla çözümleyebilecektir.

Perspektif çizim üçüncü boyutun çizimle tanımlanmasında en etkin araç olmaktadır. Bir yapının geometrik özellikleri sayısal olarak bir kez tanımlandıktan sonra bilgisayarla istenilen sayıda ve istenilen bakış açısından perspektif çizimleri elde etmek olanığı vardır. Metinde perspektif izdüşümün matamatiksel modeli anlatılmaktadır.

Perspektif çizim programlarının en önemli sorunu *görülmeyen çizgilerin* ayıklanmasıdır. Fiziksel olarak çok açık olan bu özelliğin bilgisayara aktırılması ve ayıklanması oldukça karmaşık model ve programlar gerektirmektedir. Yine metinde, *görülmeyen çizgilerin* ayıklanması için bir model ve çözümü verilmektedir.

Geliştirilen program *görülmeyen çizgileri* ayıklanmış perspektifler çizdiği gibi, Stereoskopik görüntü ikizleri de verebilmektedir. Programın bir özelliği de bir kısım noktaları bakış noktasının arkasında kalan çizgilerin görülmeyen kısımlarının ayıklanmasının işlemini yapabilmesi diğer bir özelliği ise tepe açısı tanımlanan bir görüş konisi dışındaki noktaları da ayıklayabilmesidir.

LIST OF SYMBOLS

A	First component of a vector
$\vec{a}$	First base vector of the projection plane
a_i	i th component of $\vec{a}$
B	Second component of a vector
$\vec{b}$	Second base vector of the projection plane
C	Third component of a vector
$\vec{d}$	First base vector of the transformed coordinate system
d_i	i th component of $\vec{d}$
$\vec{e}$	Second base vector of the transformed coordinate system
e_i	i th component of $\vec{e}$
$\vec{f}$	Third base vector of the transformed coordinate system
f_i	i th component of $\vec{f}$
$\vec{i}$	First base vector of the original coordinate system
i	Index for the components of a vector
$\vec{j}$	Second base vector of the orijinal coordinate system
j	Index for vectors in a set
$\vec{k}$	Third base vector of the orijinal coordinate system
$\vec{L}_j$	End points of a line segment
$\vec{l}$	Vector parallel to a line segment
$\vec{n}$	Normal of the projection plane
$\vec{P}_j$	Points of an object
$\vec{p}^1$	Transformed point
$\vec{p}$	View ray vector
$\vec{q}$	Perspective projection vector
$\vec{r}$	Vector passing through viewing point and any end point of the line segment
$\vec{S}_j$	Vertices of an opaque face
s_{ij}	i th component of $\vec{S}_j$
$[T]$	Transformation matrix
t	A parameter
$\vec{u}$	Normal of the picture plane
u_i	i th component of $\vec{u}$
$\vec{V}$	Viewing point
$\vec{v}$	Viewing vector
$\vec{W}$	Viewed point
X	First coordinate of a point
x	A variable
x_0	First coordinate of a point
Y	Second coordinate of a point
y	A variable

y_0 Second coordinate of a point

Z Third coordinate of a point

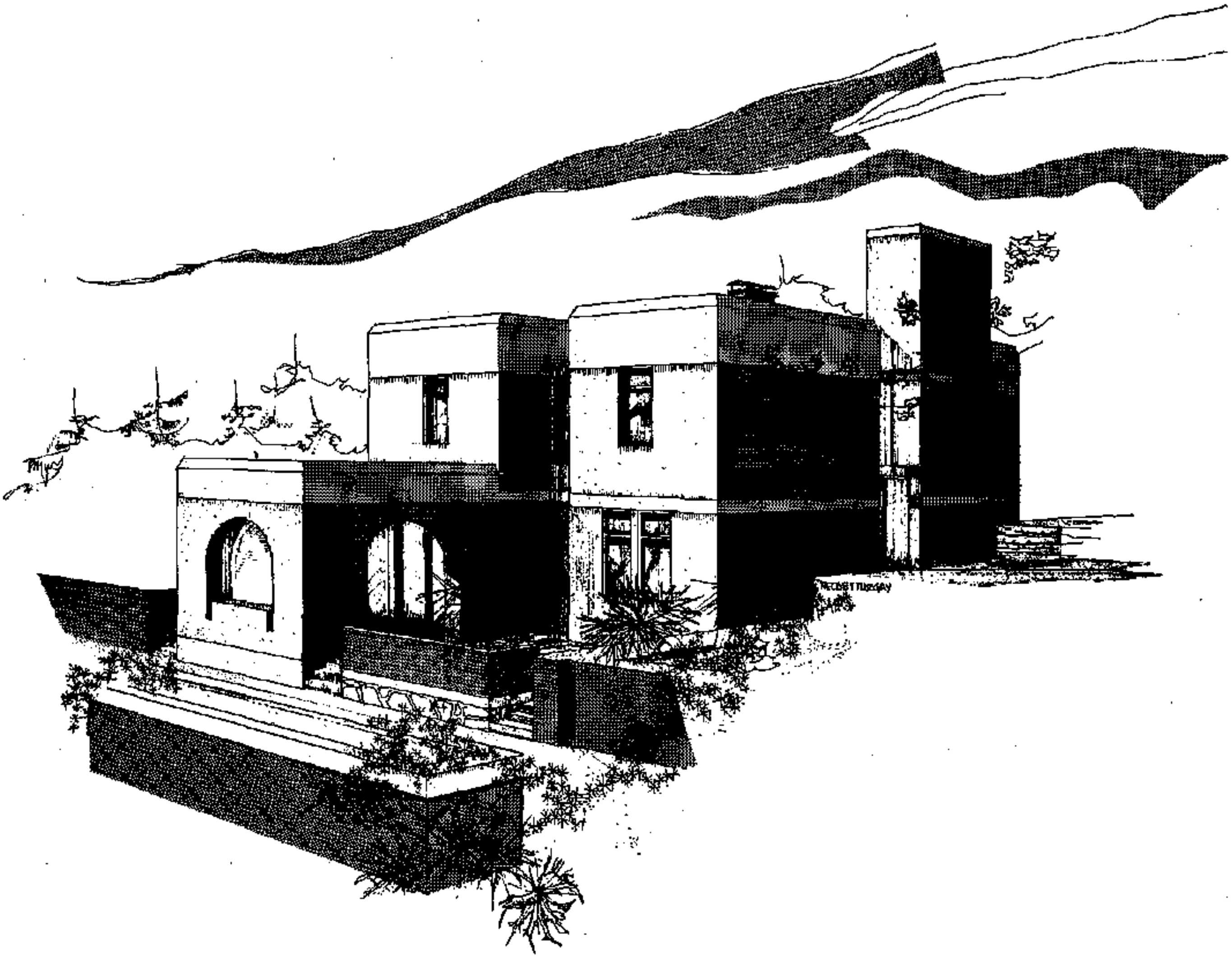
z A variable

z_0 Third coordinate of a point

λ A scalar

Φ_j *End points of the masking segment*

Ψ_j End points of the projection of the masking segment on the line segment



BIBLIOGRAPHY

APPEL, A. Interactive Graphics in Data Processing: Modeling in Three Dimensions. *IBM Systems Journal,* v.7, n.3and4 1963, pp.310-321.

CAMPION, D. *Computers in Architectural Design.* Amsterdam: Elsevier Publishing Company, 1968.

CAMPION, D. and ROBEY, K.G. Perspective Drawing by Computer. *The Architectural Review,* v.138, n.826, 1965, pp.380, 382, 384, 386.

CANBULAT, İ. "Theoretical Bases and Program Description of a Computerized Method for Perspective Drawing". Unpublished Master Thesis, Middle East Technical University, Faculty of Architecture, Department of Architecture, Ankara, 1974.

COMBA, P.G. Interactive Graphics in Data Processing: A Language for Three-Dimensional Geometry. *IBM Systems Journal,*v.7, n.3 and 4, 1968. pp.292-307

FETTER, W.A. *Computer Graphics in Communication.* New York: McGraw-Hill Book Company. 1964.

GALIMBERTI, R. and MONTANARI, V. An Algorithm for Hidden Line Elimination. *Communications of the ACM,* v.12, n.4, pp.206-211.

GREENBERG, D.P. Computer Graphics in Architecture. *Scientific American,* v.230, n.5, 1974, pp.93-106.

HADLEY, G. *Linear Algebra.* Reading, Massachusetts: Addison-Wesley Publishing Company, 1961.

HAUSER, W. *Introduction to the Principles of Mechanics.*Reading, Massachusetts: Addison-Wesley Publishing Company.1961.

KRAMPEN, M. and SEITZ, P. (Eds.) *Design and Planning 2: Computers in Design and Communication.* New York: Hasting House Publishers.1967.

LOUTREL, P.P. A Solution to the Hidden-Line Problem for Computer-Draw Polyhedra. *IEEE Transactions on Computers,* v.C-19, n.3, 1970, pp.205-213.

McCARTNEY, J. Information. *Scientific American,* v.215, n.3, 1966 (whole issue).

McCARTNEY, T.O. *Precision Perspective Drawin.* New York: McGraw-Hill Book Company, 1963.

NEGROPONTE, N. *The Architecture Machine.* Cambridge Massachusetts: The MIT Press, 1970.

ÖZDURAL, A. "Architectural Photogrammetry". Unpublished Lecture Notes, Middle East Technical University, Faculty of Architecture, Department of Preservation and Restoration of Historic Monuments. Ankara. 1972.

PRINCE, M.D. *Interactive Graphics for Computer Aided Design.* Reading, Massachusetts: Addison-Wesley Publishing Company. 1971.

SMITH, T.C. and PAO. Y.C. *Introduction to Digital Computer Plotting.* New York: Gordon and Preach Science Publishers, 1973.

STEWARD, C., TEICHOLTZ, E. and LEE, R. *CAP: Computer Architecture Programs.* Cambridge, Massachusetts: Center for Environmental Research, 1972.

SUTHERLAND, I.E. Computer Display. *Scientific American*, v.222, n.6, 1970, pp.57-60, 65-74, 79, 81.

ZOLFAGHART, M. Détermination des Parties Vues et Cachées en veu du Trâce Automatique de Perspectives Urbaines. *Sociéte Francise de Photogramméetrie,* n.49, 1973, pp.3-20.