RCS-603: COMPUTER GRAPHICS
UNIT-III

Presented By :

Dr. Vinod Jain (Associate Professor, GLBITM)

GL BAJAJ
Institute of Technology & Management
[Approved by AICTE, Govt. of India & Affiliated to Dr. APJ Abdul Kalam Technical University, Lucknow, U.P. India]
Department Of Computer Science & Engineering

Topics Left

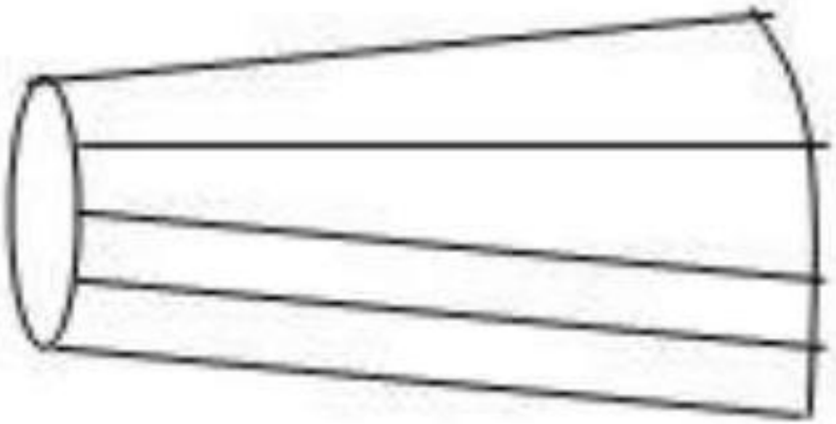
- 3d Object Representation (chapter 10, 10.1)
- 3-D Geometric Primitives, (9.1,9.2)
- 3-D viewing, projections, 3-D Clipping. (ch 12)

3d Object Representation

3D object representation is divided into two categories.

- **Boundary Representations (B-reps)** –
- It describes a 3D object as a **set of surfaces** that separates the object interior from the environment.
- **Space–partitioning representations** –
- It is used to describe interior properties, by partitioning the spatial region containing an object into a **set of small, non-overlapping, contiguous solids (usually cubes).**

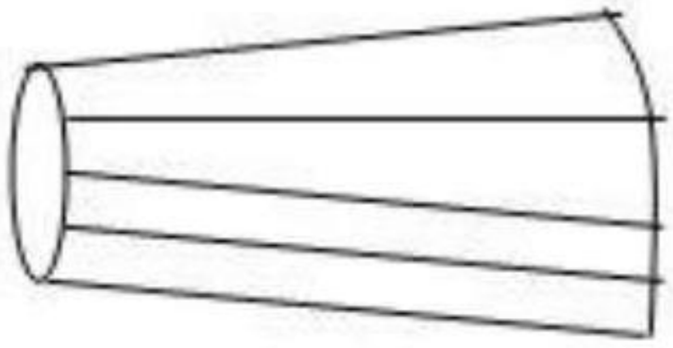
3d Object Representation : Polygon Surfaces



A 3D object represented by polygons

3d Object Representation : Polygon Surfaces

- The most commonly used representation for a 3D graphics object.
- It is a set of surface polygons that enclose the object interior.
- Set of polygons are stored for object description.
- This simplifies and speeds up the surface rendering and display of object since all surfaces can be described with linear equations.



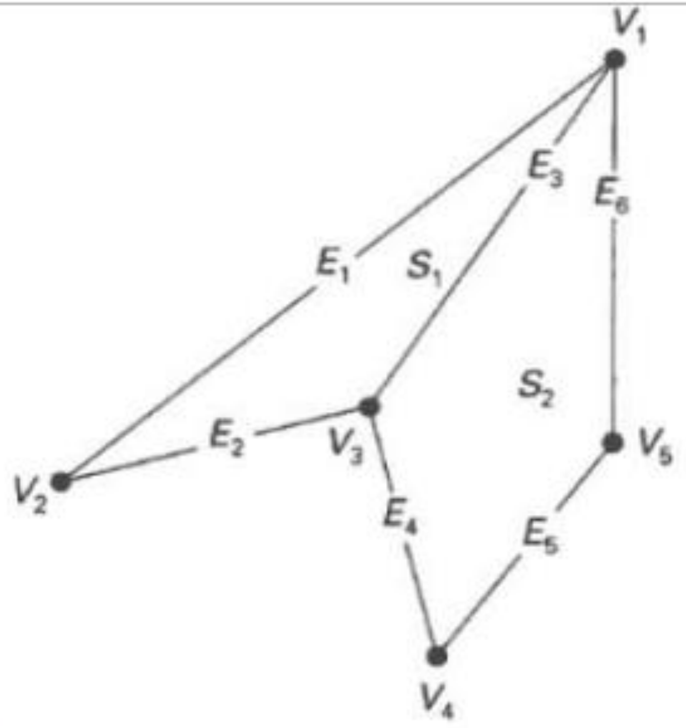
A 3D object represented by polygons

3d Object Representation : Polygon Surfaces

Three ways to represent polygon surfaces

1. Polygon Tables
2. Plane Equations
3. Polygon Meshes

Polygon Tables



VERTEX TABLE	
V_1:	x_1, y_1, z_1
V_2:	x_2, y_2, z_2
V_3:	x_3, y_3, z_3
V_4:	x_4, y_4, z_4
V_5:	x_5, y_5, z_5

EDGE TABLE	
E_1:	V_1, V_2
E_2:	V_2, V_3
E_3:	V_3, V_1
E_4:	V_3, V_4
E_5:	V_4, V_5
E_6:	V_5, V_1

POLYGON-SURFACE TABLE	
S_1:	E_1, E_2, E_3
S_2:	E_3, E_4, E_5, E_6

Polygon Tables

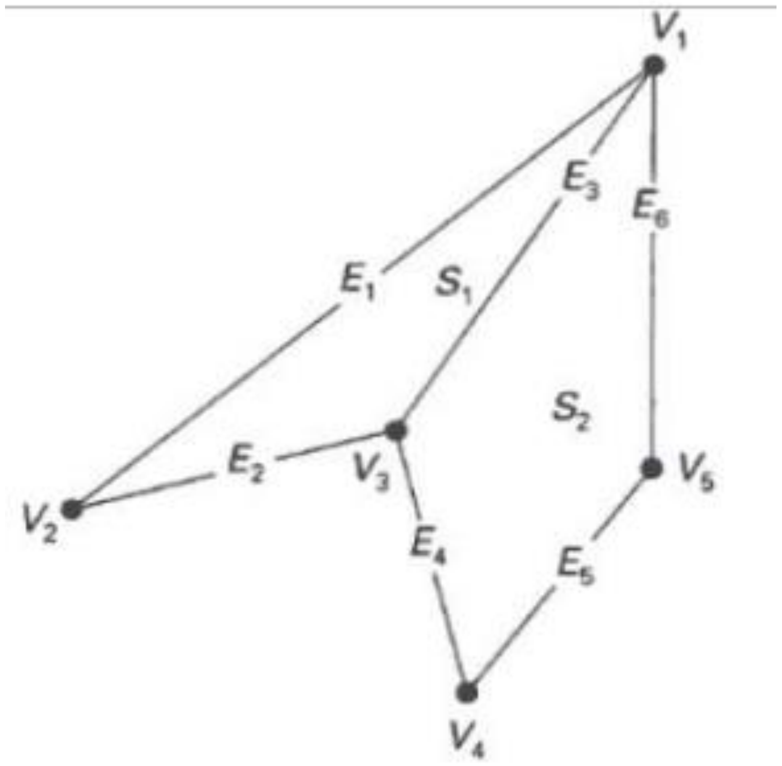
The object is store by using three tables

1. Vertex Table
2. Edge table
3. Polygon-Surface table

Polygon Tables- Vertex Table

Vertex Table

It store x, y, and z coordinate information of all the vertices as v_1: x_1, y_1, z_1.

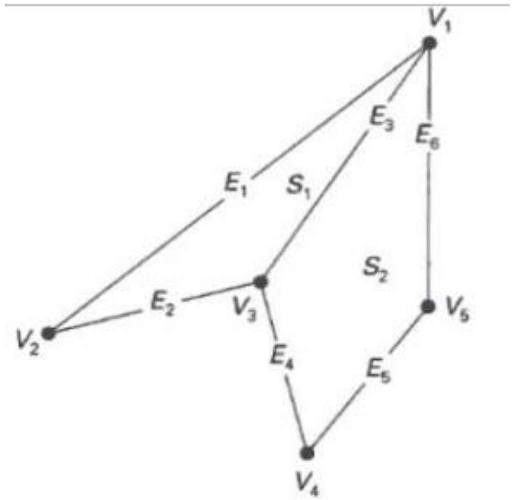


VERTEX TABLE	
V_1:	x_1, y_1, z_1
V_2:	x_2, y_2, z_2
V_3:	x_3, y_3, z_3
V_4:	x_4, y_4, z_4
V_5:	x_5, y_5, z_5

Polygon Tables - **Edge table**

Edge table

- The Edge table is used to store the edge information of polygon.
- In the following figure, edge E_1 lies between vertex v_1 and v_2 which is represented in the table as E_1: v_1, v_2.



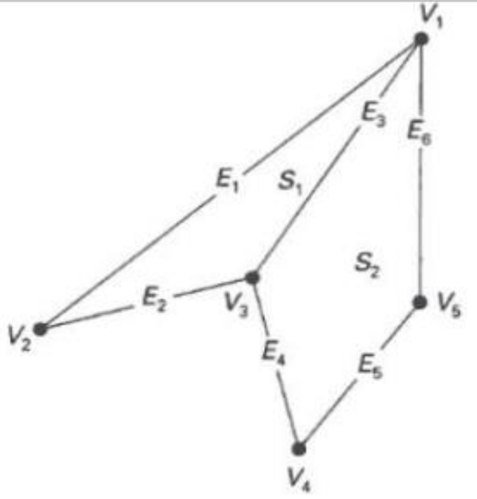
EDGE TABLE	
E_1:	V_1, V_2
E_2:	V_2, V_3
E_3:	V_3, V_1
E_4:	V_3, V_4
E_5:	V_4, V_5
E_6:	V_5, V_1

Polygon Tables - **Polygon-Surface table**

Polygon-Surface table

Polygon surface table stores the number of surfaces present in the polygon.

From the following figure, surface S_1 is covered by edges E_1, E_2 and E_3 which can be represented in the polygon surface table as S_1: E_1, E_2, and E_3



POLYGON-SURFACE TABLE	
S_1:	E_1, E_2, E_3
S_2:	E_3, E_4, E_5, E_6

Plane Equations

- The equation for plane surface can be expressed as −

$$Ax + By + Cz + D = 0$$

- Where (x, y, z) is any point on the plane, and the coefficients A, B, C, and D are constants describing the spatial properties of the plane.
- We can obtain the values of A, B, C, and D by solving a set of three plane equations using the coordinate values for three non collinear points in the plane. Let us assume that three vertices of the plane are (x_1, y_1, z_1), (x_2, y_2, z_2) and (x_3, y_3, z_3).

Plane Equations

- The equation for plane surface can be expressed as −

$$Ax + By + Cz + D = 0$$

- We can obtain the values of A, B, C, and D by solving a set of three plane equations.
- Let us assume that three vertices of the plane are (x_1, y_1, z_1), (x_2, y_2, z_2) and (x_3, y_3, z_3).

Plane Equations

- Let us solve the following simultaneous equations for ratios A/D, B/D, and C/D. You get the values of A, B, C, and D.

$$(A/D)\ x_1 + (B/D)\ y_1 + (C/D)\ z_1 = -1$$

$$(A/D)\ x_2 + (B/D)\ y_2 + (C/D)\ z_2 = -1$$

$$(A/D)\ x_3 + (B/D)\ y_3 + (C/D)\ z_3 = -1$$

Plane Equations

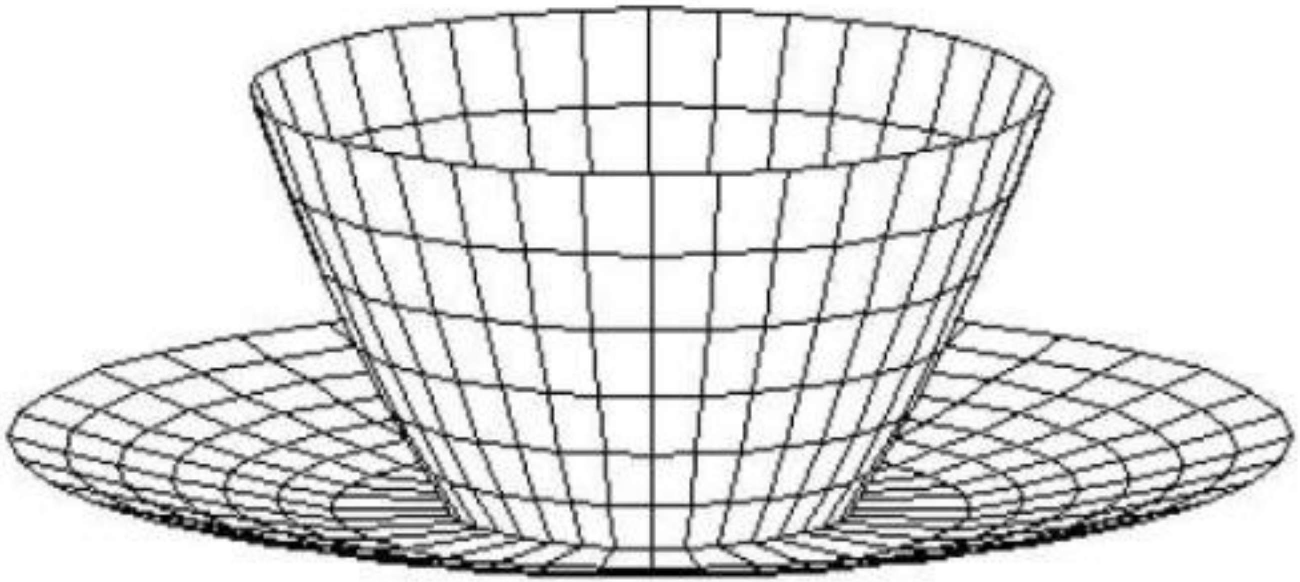
To obtain the above equations in determinant form, apply Cramer's rule to the above equations.

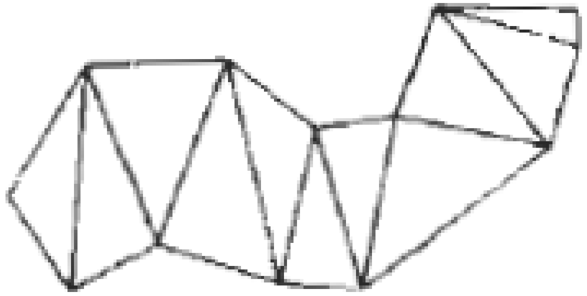
$$A = \begin{bmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{bmatrix} B = \begin{bmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{bmatrix} C = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} D = -\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

For any point (x, y, z) with parameters A, B, C, and D, we can say that −

- $Ax + By + Cz + D \neq 0$ means the point is not on the plane.
- $Ax + By + Cz + D < 0$ means the point is inside the surface.
- $Ax + By + Cz + D > 0$ means the point is outside the surface.

Plane Equations





Polygon Meshes

3D surfaces and solids can be approximated by a set of polygonal and line elements. Such surfaces are called **polygonal meshes**.

In polygon mesh, each edge is shared by at most two polygons.

The set of polygons or faces, together form the "skin" of the object.

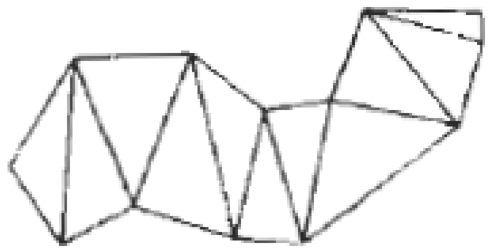


Figure 10-6
A triangle strip formed with 11 triangles connecting 13 vertices.

Polygon Meshes

- **Advantages**
  - It can be used to model almost any object.
  - They are easy to represent as a collection of vertices.
  - They are easy to transform.
  - They are easy to draw on computer screen.
- **Disadvantages**
  - Curved surfaces can only be approximately described.
  - It is difficult to simulate some type of objects like hair or liquid.

Three Dimensional Concepts

Three Dimensional Display Methods:

- Parallel Projection
- Perspective Projection
- Depth Queing
- Visible Line and Surface Identification
- Surface Rendering
- Exploded and Cutaway Views
- Three-Dimensional and Stereoscopic Views

Three Dimensional Concepts

Three Dimensional Display Methods:

- To obtain a display of a three dimensional scene that has been modeled in world coordinates, we must setup a co-ordinate reference for the 'camera'.

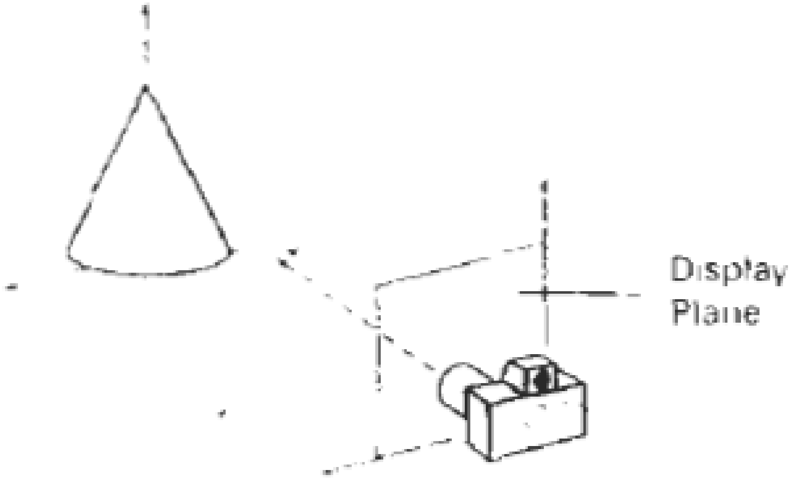
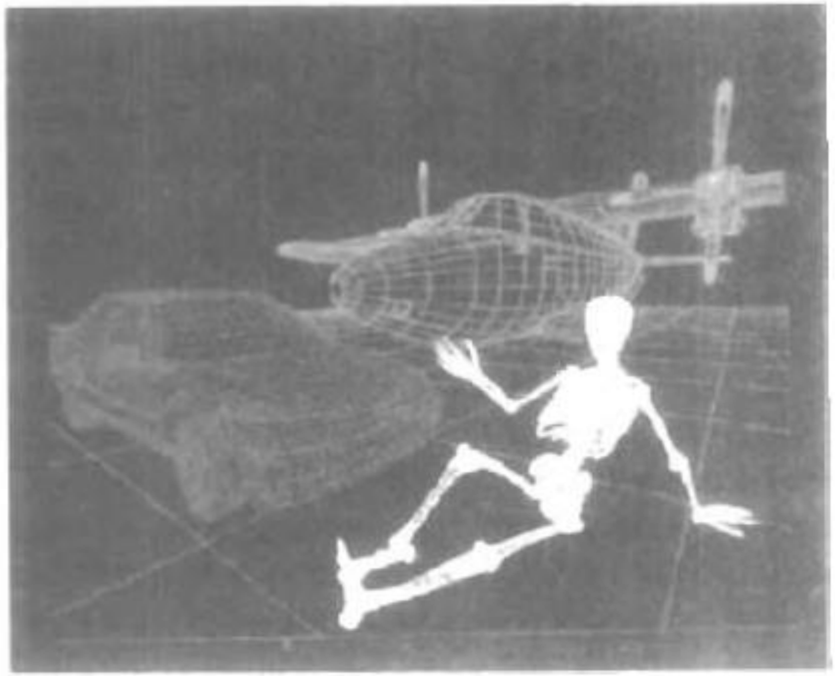


Figure 9-1
Coordinate reference for obtaining a particular view of a three-dimensional scene.

Three Dimensional Concepts

Three Dimensional Display Methods:

- The objects can be displayed in wire frame form, or we can apply lighting and surface rendering techniques to shade the visible surfaces.



Parallel Projection

- Parallel projection is a method for generating a view of a solid object is to **project points on the object surface along parallel lines** onto the display plane.
- This technique is used in engineering and architectural drawings to **represent an object with a set of views** that maintain relative proportions of the object.

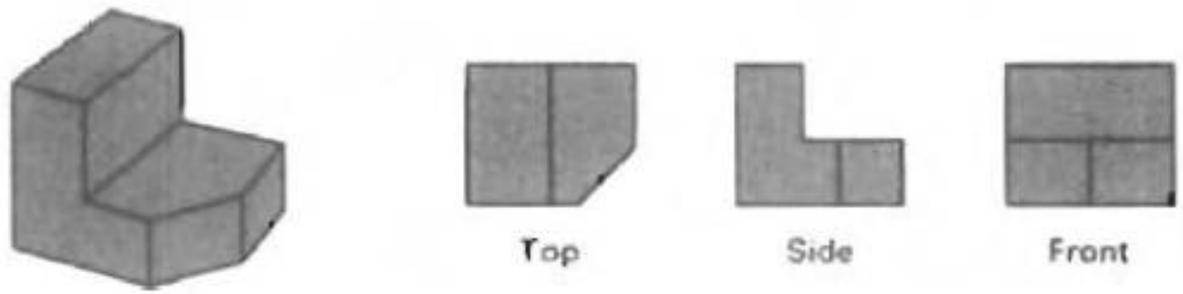
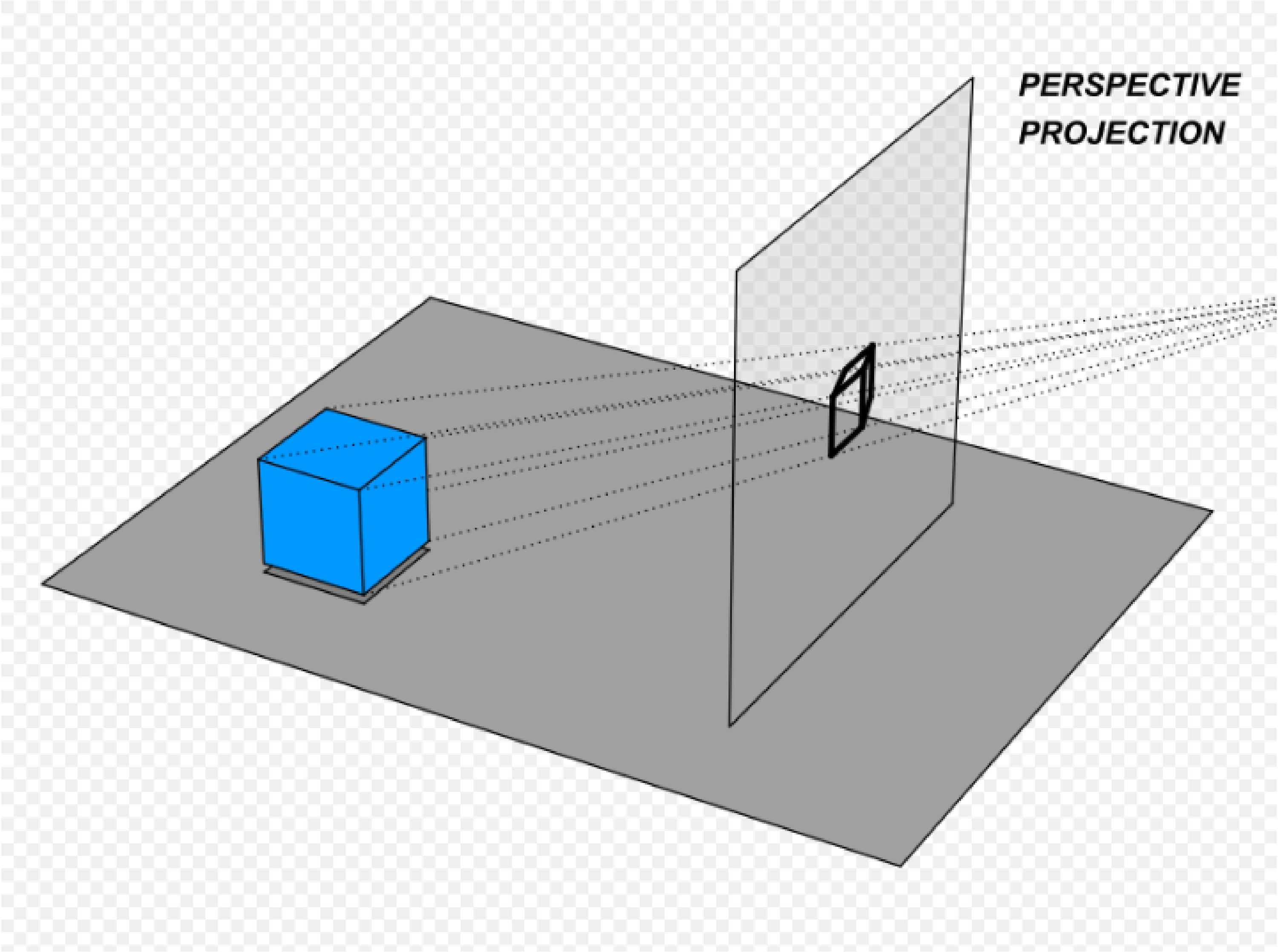


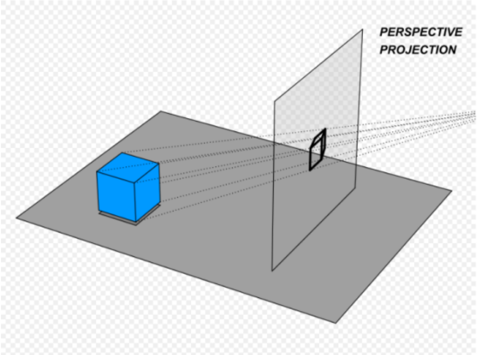
Figure 9-3
Three parallel-projection views of an object, showing relative proportions from different viewing positions.

Perspective Projection



Perspective Projection

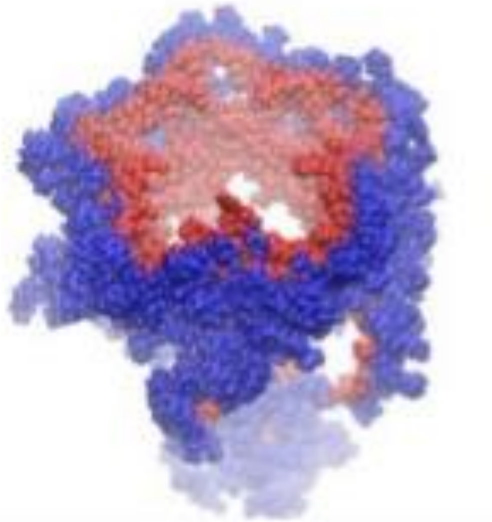
- It is a method for generating a view of a three dimensional scene is to **project points to the display plane alone converging paths.**
- In a perspective projection, parallel lines in a scene that are not **parallel to the display plane are projected into converging lines.**
- Scenes displayed using perspective **projections appear more realistic**, since this is the way that our eyes and a camera lens form images.



Depth Cueing:

Depth Cueing

- Hidden surfaces are not removed but displayed with different effects such as intensity, color, or shadow for giving hint for third dimension of the object.
- Simplest solution: use different colors-intensities based on the dimensions of the shapes.



Depth Cueing:

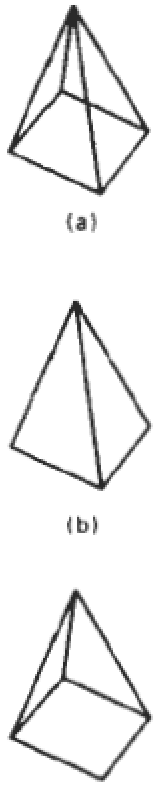
- Depth cueing is a method for indicating depth with wire frame displays is to vary the intensity of objects according to their distance from the viewing position.
- Depth cueing is applied by choosing maximum and minimum intensity (or color) values and a range of distance over which the intensities are to vary.

Visible Line and Surface Identification

- A simplest way to identify the visible line is to **highlight the visible lines or to display them in a different color.**
- Another method is to display the non visible lines as dashed lines.

Visible Line and Surface Identification

- The wireframe representation of the pyramid in
- (a) contains no depth information to indicate whether the viewing direction is
- (b) downward from a position above the apex or
- (c) upward from a position below the base.



(a) (b) (c)

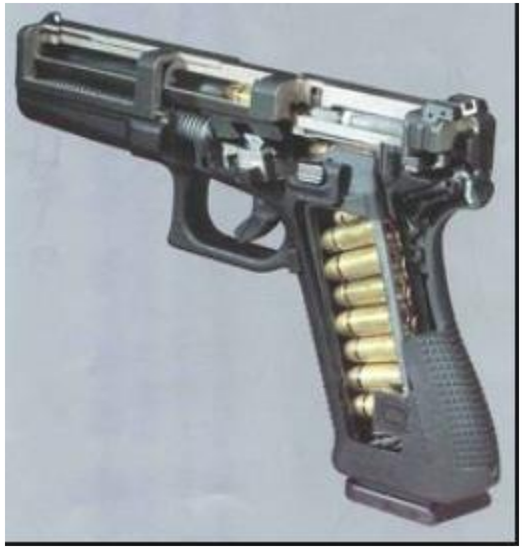
Surface Rendering

Surface Rendering

- Surface rendering method is used to generate a degree of realism in a displayed scene.
- Realism is attained in displays by setting the surface intensity of objects according to the lighting conditions in the scene and surface characteristics.
- Lighting conditions include the intensity and positions of light sources and the background illumination.
- Surface characteristics include degree of transparency and how rough or smooth the surfaces are to be.

Exploded and Cutaway Views

- Exploded and cutaway views of objects can be to show the internal structure and relationship of the objects parts.
- An alternative to exploding an object into its component parts is the cut away view which removes part of the visible surfaces to show internal structure.

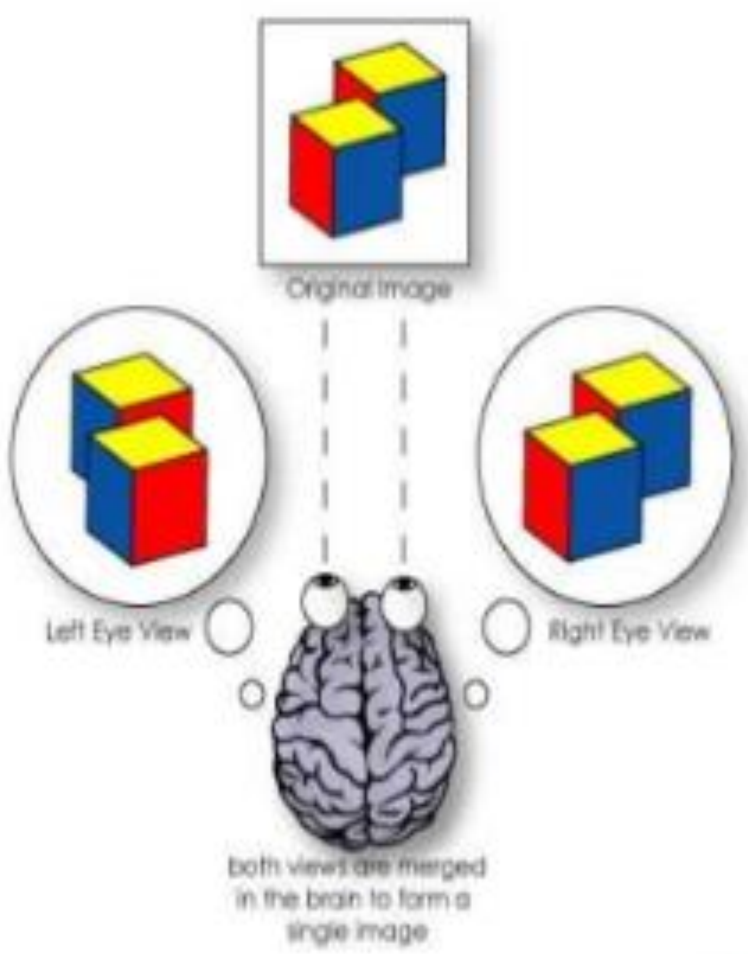


Three-Dimensional and Stereoscopic Views

- In Stereoscopic views, three dimensional views can be obtained by reflecting a raster image from a vibrating flexible mirror.
- The vibrations of the mirror are synchronized with the display of the scene on the CRT.
- Stereoscopic devices present two views of a scene; one for the left eye and the other for the right eye.

Three-Dimensional and Stereoscopic Views

- Stereoscopic devices present two views of a scene;
- one for the left eye
- and the other for the right eye.



3-D Transformation

- Methods for geometric transformations are extended from two-dimensional methods by including considerations for the ***z*** coordinate.
- We will discuss following transformations in 3D

1. Translation
2. Rotation
3. Scaling
4. Reflection
5. Shear

Unit- III

1. 3-D Geometric Primitives
2. 3-D Object representation
3. **3-D Transformation**
4. 3-D viewing
5. projections
6. 3-D Clipping.

3-D Translation

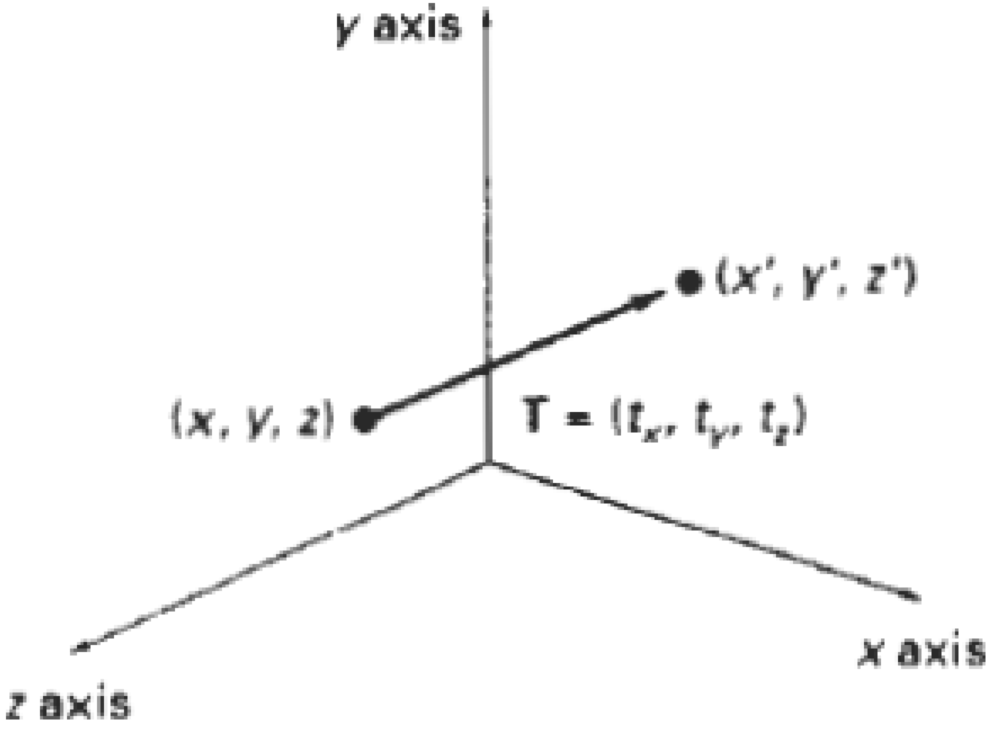


Figure 11-1
Translating a point with translation vector $\mathbf{T} = (t_x, t_y, t_z)$.

3-D Translation by translation vector $T=(t_x,t_y,t_z)$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{P}' = \mathbf{T} \cdot \mathbf{P}$$

$$x' = x + t_x, \qquad y' = y + t_y, \qquad z' = z + t_z$$

3-D Rotation

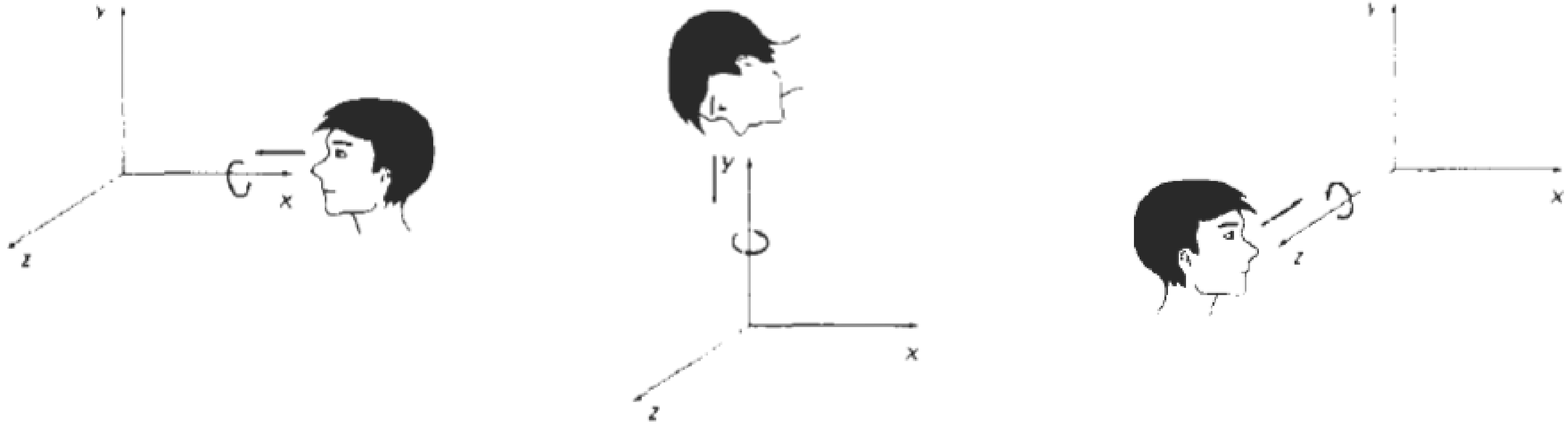
To generate a rotation transformation for an object, we must designate

1. An axis of rotation (about which the object is to be rotated)
2. The amount of angular rotation.

3-D Rotation

- In two-dimensional applications, where all transformations are carried out in the xy plane
- A three-dimensional rotation can be specified around any line in space.
- The easiest rotation axes to handle are those that are parallel to the coordinate axes.

3-D Rotation – three rotation axis are



3-D Rotation : In homogeneous coordinates

z-axis rotation equations are expressed as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$x' = x\cos\theta - y\sin\theta$$

$$y' = x\sin\theta + y\cos\theta$$

$$z' = z$$

3-D Rotation : In homogeneous coordinates

z-axis rotation equations are expressed as

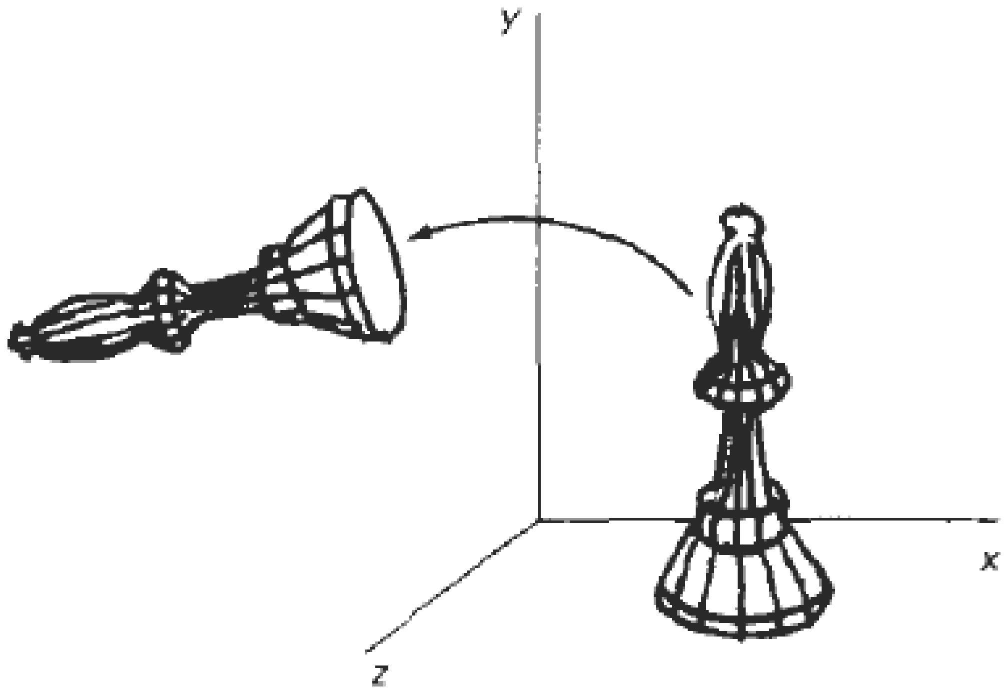


Figure 11-4
Rotation of an object about the z axis.

which we can write more compactly as

$$\mathbf{P}' = \mathbf{R}_z(\theta) \cdot \mathbf{P}$$

3-D Rotation : In homogeneous coordinates

x-axis rotation equations are expressed as

Substituting permutations 11-7 in Eqs. 11-4, we get the equations for an **x-axis rotation:**

$$
\begin{aligned}
y' &= y\cos\theta - z\sin\theta \\
z' &= y\sin\theta + z\cos\theta \\
x' &= x
\end{aligned}
\qquad (11\text{-}8)
$$

which can be written in the homogeneous coordinate form

$$
\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (11\text{-}9)
$$

$$
\mathbf{P}' = \mathbf{R}_x(\theta) \cdot \mathbf{P}
$$

3-D Rotation : In homogeneous coordinates

y-axis rotation equations are expressed as

$$z' = z\cos\theta - x\sin\theta$$

$$x' = z\sin\theta + x\cos\theta$$

$$y' = y$$

The matrix representation for y-axis rotation is

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

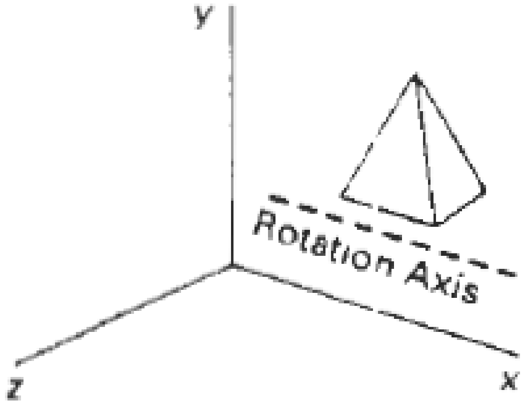
or

$$\mathbf{P}' = \mathbf{R}_y(\theta) \cdot \mathbf{P}$$

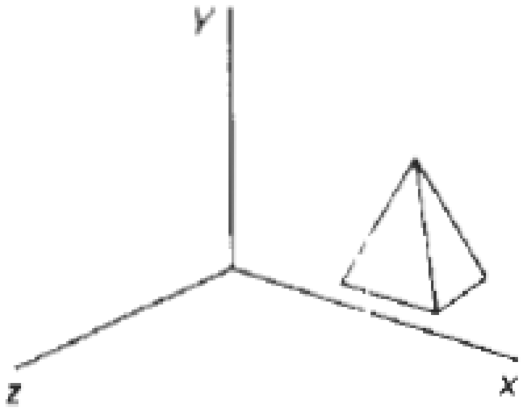
General Three-Dimensional Rotations (About any given axis)

Three Steps

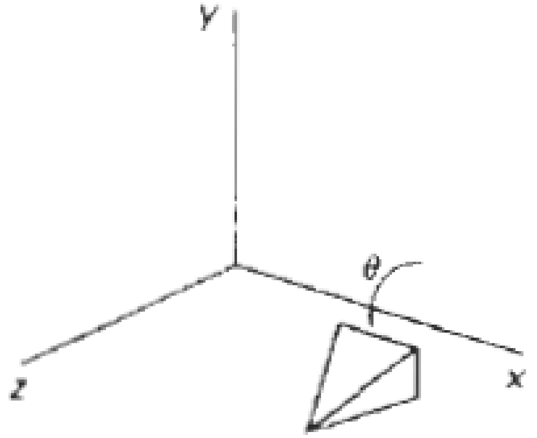
1. Translate the object so that the rotation axis coincides with the parallel coordinate axis.
2. Perform the specified rotation about that axis.
3. Translate the object so that the rotation axis is moved back to its original position.



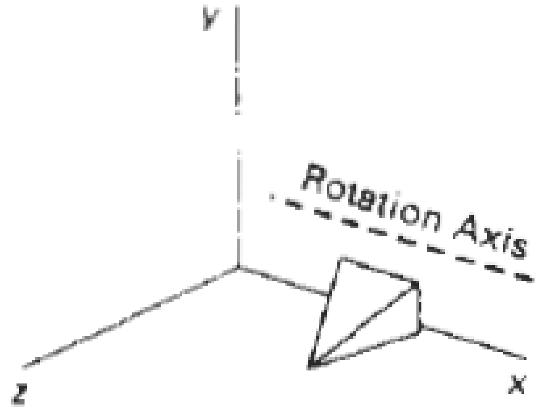
(a)
Original Position o' Object



(b)
Translate Rotation Axis onto *x* Axis



(c)
Rotate Object Through Angle θ



(d)
Translate Rotation
Axis to Original Position

General Three-Dimensional Rotations (About any given axis)

The steps in this sequence are illustrated in Fig. 11-8. Any coordinate position **P** on the object in this figure is transformed with the sequence shown as

$$\mathbf{P}' = \mathbf{T}^{-1} \cdot \mathbf{R}_x(\theta) \cdot \mathbf{T} \cdot \mathbf{P}$$

where the composite matrix for the transformation is

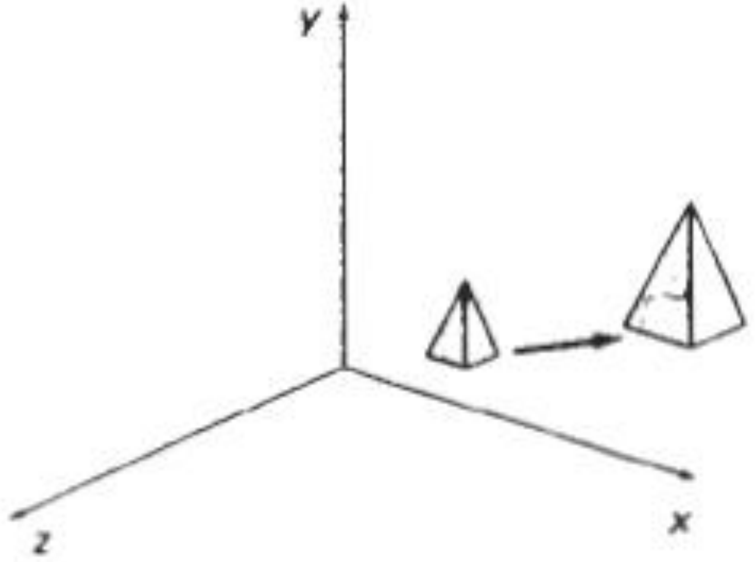
$$\mathbf{R}(\theta) = \mathbf{T}^{-1} \cdot \mathbf{R}_x(\theta) \cdot \mathbf{T}$$

3 D SCALING

- The matrix expression tor the scaling transformation of a position P = **(x,** y, *z)* relative to the coordinate origin can be written as

P’ = S . P

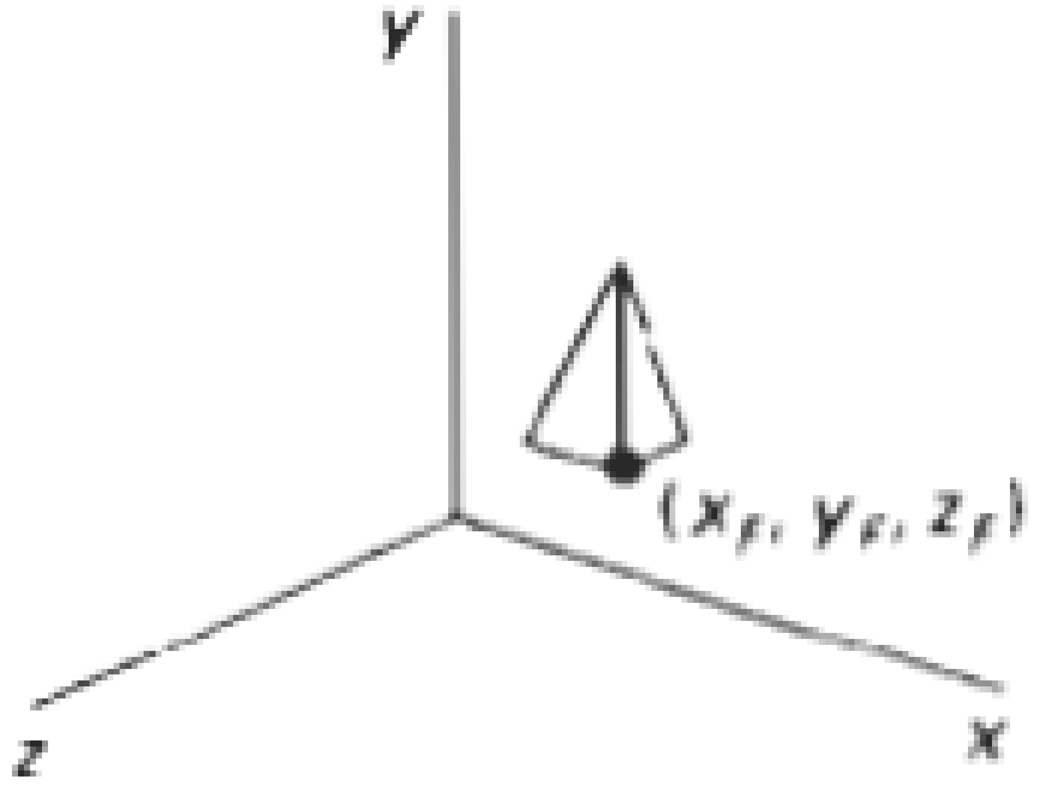
Where Sx, Sy and Sz are scaling factors along three axis



$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{P}' = \mathbf{S} \cdot \mathbf{P}$$

Scaling with respect to a selected fixed position (x_f, y_f, z_f)

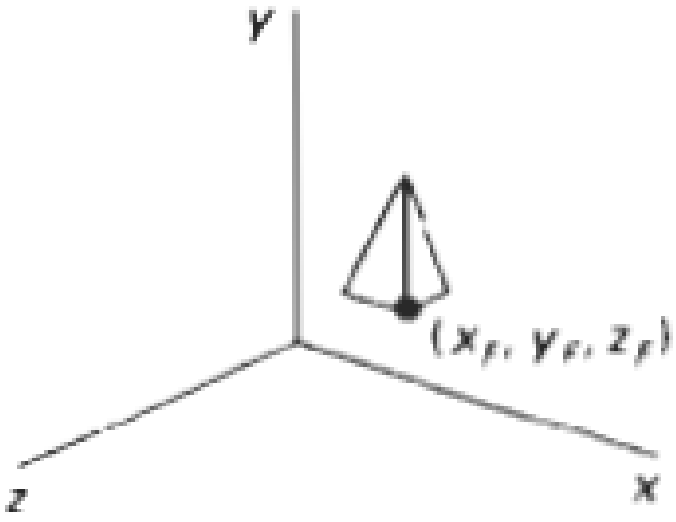


(a)

Scaling with respect to a selected fixed position (x_f, y_f, z_f)

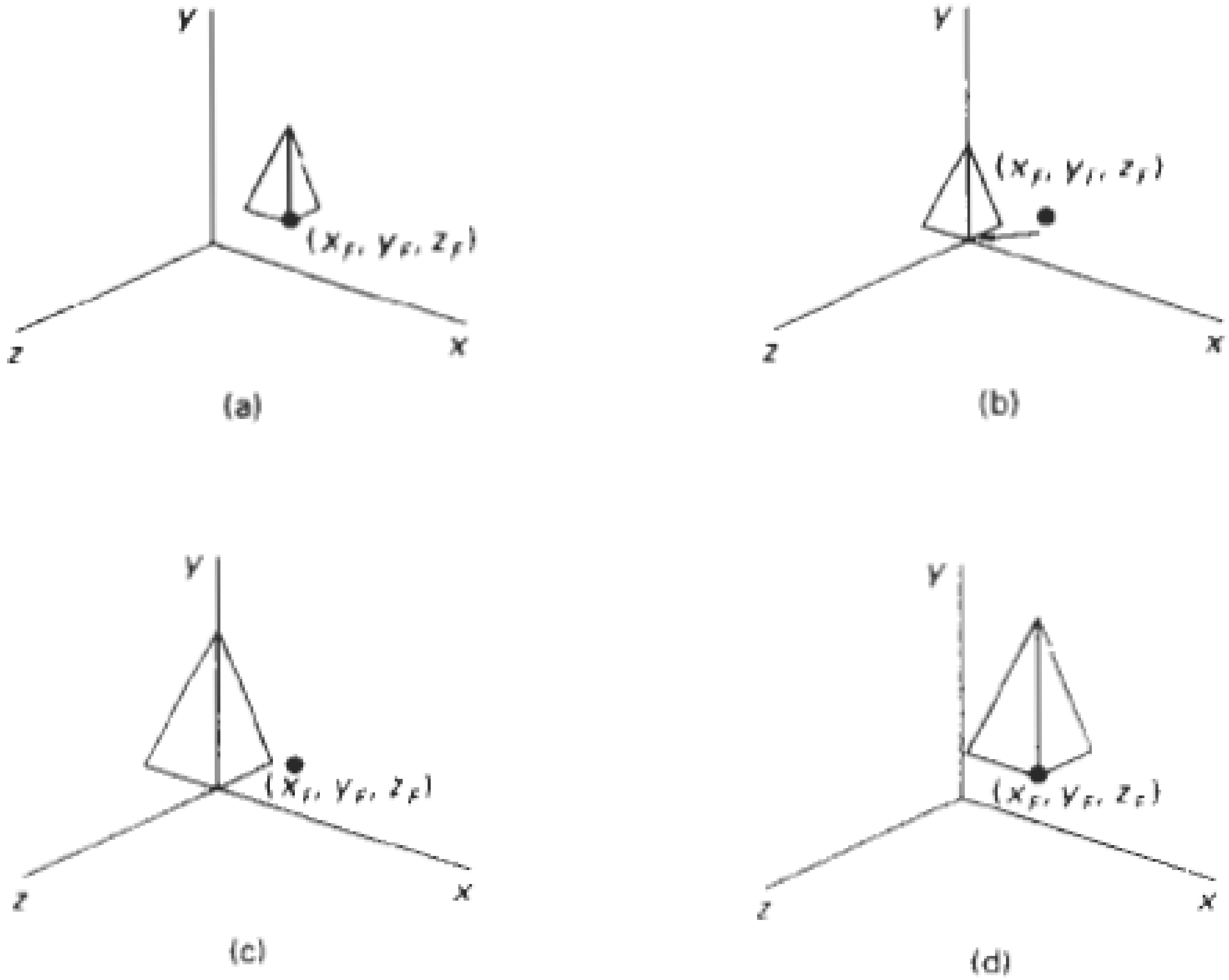
- Scaling with respect to a selected fixed position can be represented with the following transformation sequence:

1. Translate the fixed point to the origin.
2. Scale the object relative to the coordinate origin.
3. Translate the fixed point back to its original position.



(a)

Scaling with respect to a selected fixed position (x_f, y_f, z_f)



Scaling with respect to a selected fixed position (x_f, y_f, z_f)

- The matrix representation for an arbitrary fixed-point scaling can then be expressed as the concatenation of these translate-scale-translate transformations as

$$\mathbf{T}(x_f, y_f, z_f) \cdot \mathbf{S}(s_x, s_y, s_z) \cdot \mathbf{T}(-x_f, -y_f, -z_f) = \begin{bmatrix} s_x & 0 & 0 & (1 - s_x)x_f \\ 0 & s_y & 0 & (1 - s_y)y_f \\ 0 & 0 & s_z & (1 - s_z)z_f \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Other Transformations in 3D

- Reflection
- Shear

Reflection in 3D

- A three-dimensional reflection can be performed relative to a selected *reflection axis* or with respect to a selected reflection *plane.*
- When the reflection plane is a coordinate plane (either xy, *xz, or* yz), we can think of the transformation as a conversion between Left-handed and right-handed systems.

Reflection related to xy plane

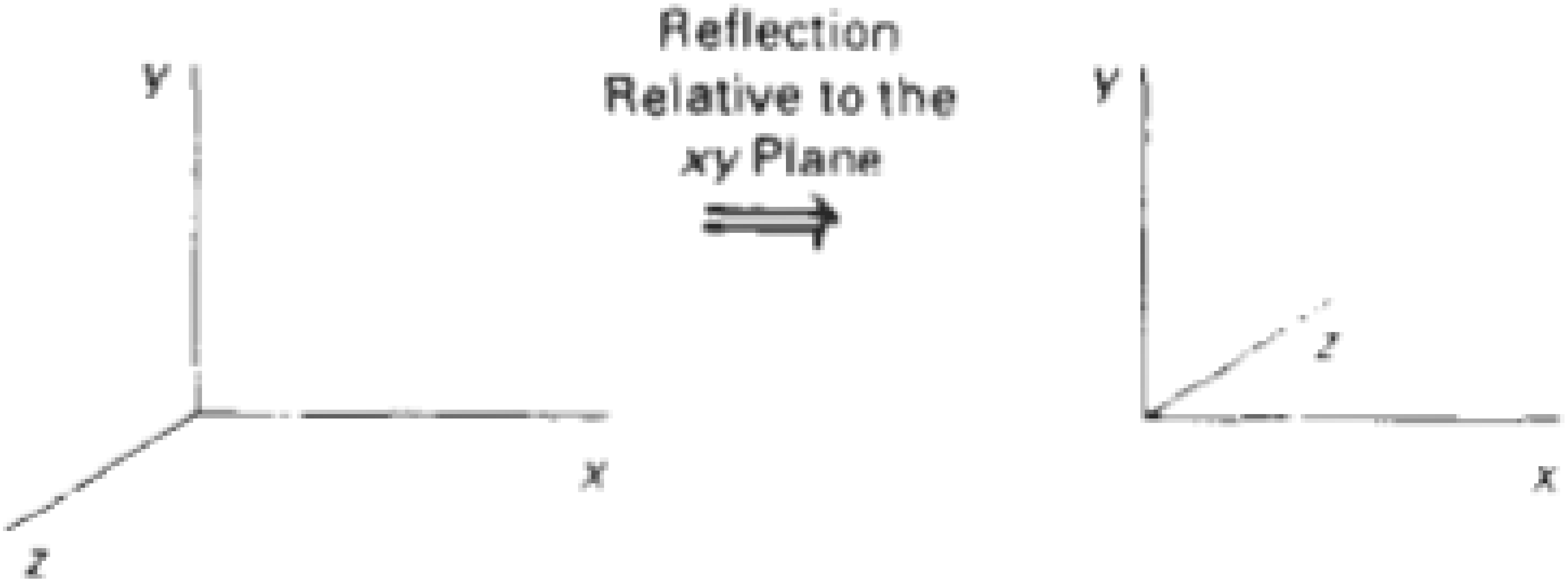


Figure 11-19
Conversion of coordinate specifications from a right-handed to a left-handed system can be carried out with the reflection transformation 11-46.

Reflection related to xy plane

- The matrix representation for this reflection of points relative to the xy plane is

$$RF_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Shear in 3D

- In two dimensions, we discussed transformations relative to the **x** or y axes to produce distortions in the shapes of objects.
- In three dimensions, we can also generate shears relative to the *z* axis.

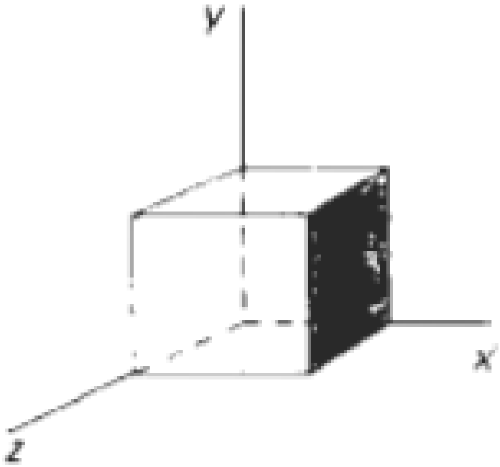
Shear in 3D

- As an example of three-dimensional shearing. the following transformation produces a z-axis shear:

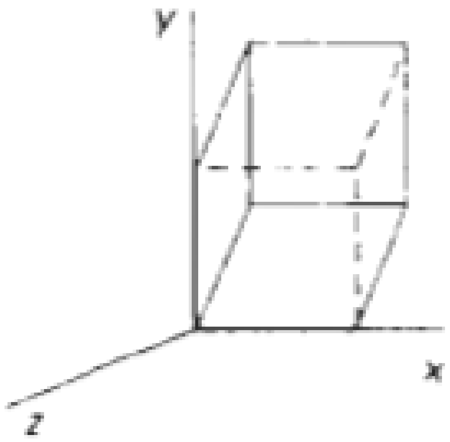
$$SH_z = \begin{bmatrix} 1 & 0 & a & 0 \\ 0 & 1 & b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Parameters a and *b* can be assigned any real values.
- The effect of this transformation matrix is to alter **x-** and y-coordinate values by an amount that is proportional to the z value, while leaving the ***z*** coordinate unchanged.

Shear in 3D



(a)



(b)

$$SH_z = \begin{bmatrix} 1 & 0 & a & 0 \\ 0 & 1 & b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- A unit cube (a) is sheared (b) by transformation matrix with a=b=1

References

3d Object Representation :

https://www.tutorialspoint.com/computer_graphics/computer_graphics_surfaces.htm