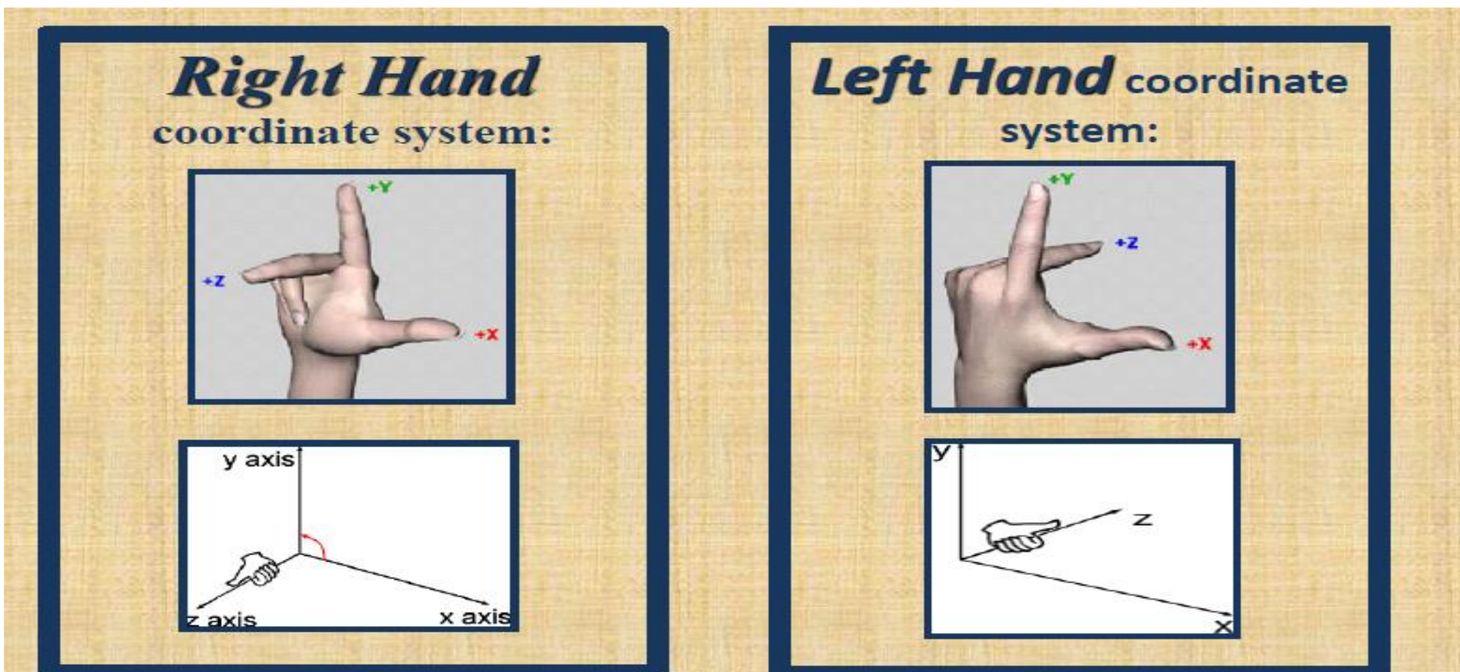
Computer Graphics

3D Transformations

3D Coordinate System

- 3D computer graphics involves the additional dimension of depth, allowing more realistic representations of 3D objects in the real world

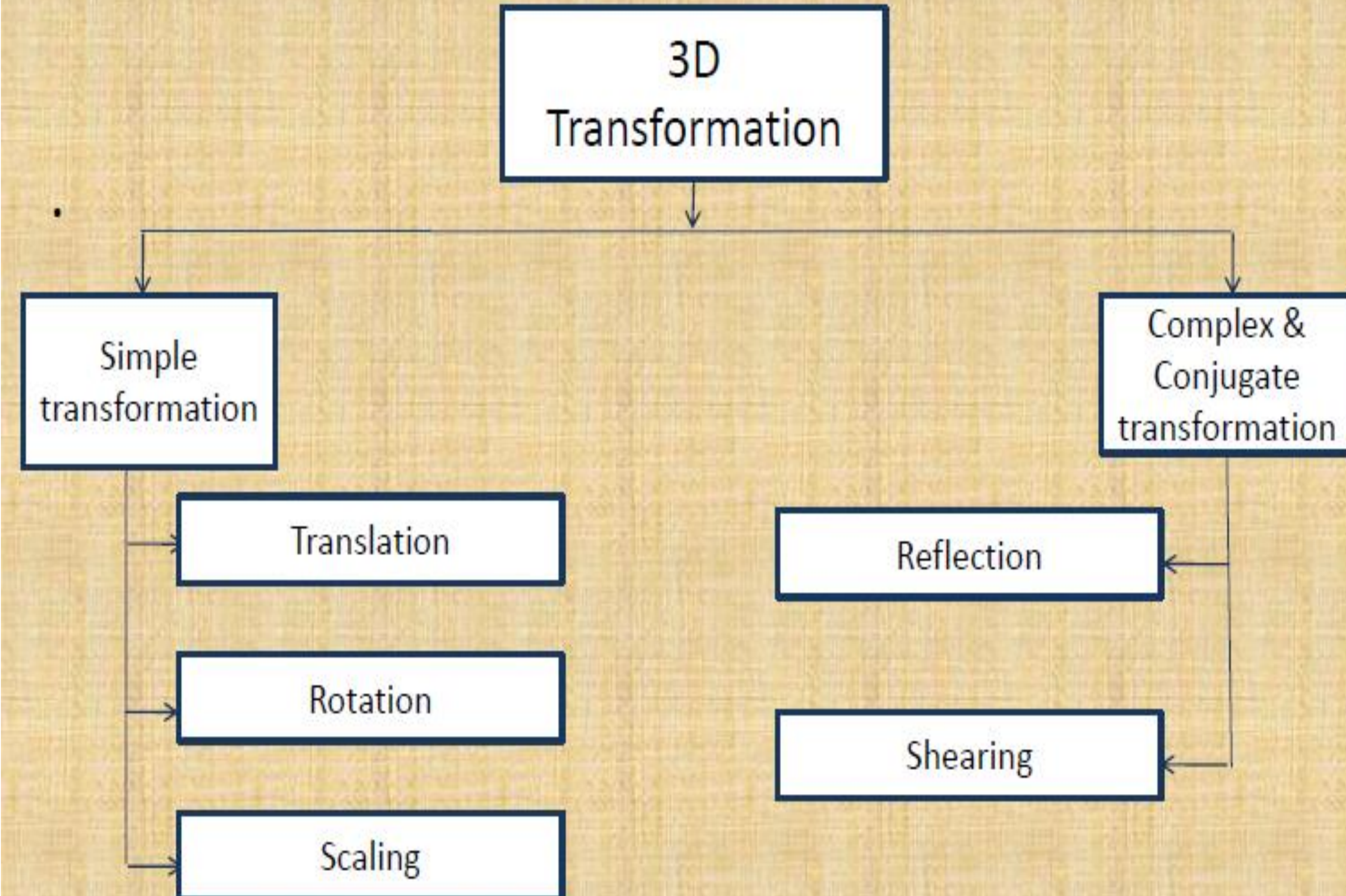
- There are two possible ways of “attaching” the Z-axis, which gives rise to a left-handed or a right-handed system



3D Transformation

- The translation, scaling and rotation transformations used for 2D can be extended to three dimensions
- In 3D, each transformation is represented by a 4x4 matrix
- Using homogeneous coordinates it is possible to represent each type of transformation in a matrix form and integrate transformations into one matrix
- To apply transformations, simply multiply matrices, also easier in hardware and software implementation
- Homogeneous coordinates can represent directions
- Homogeneous coordinates also allow for non-affine transformations, e.g., perspective projection

3D Transformation



Matrix Representation: 2-D Transformations

Basic Transformations

Translation
P'=TP

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

Rotation [O]
P'=RP

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

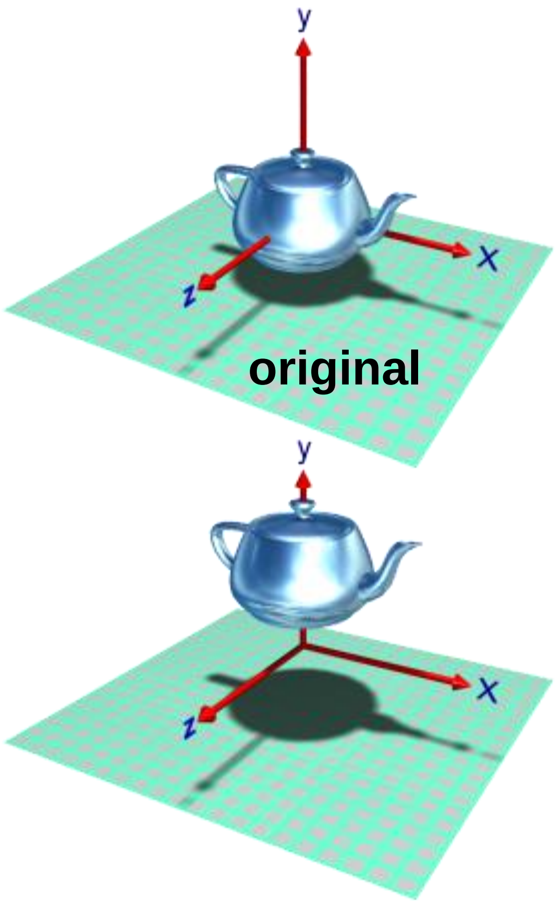
Scaling [O]
P'=SP

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

Translation

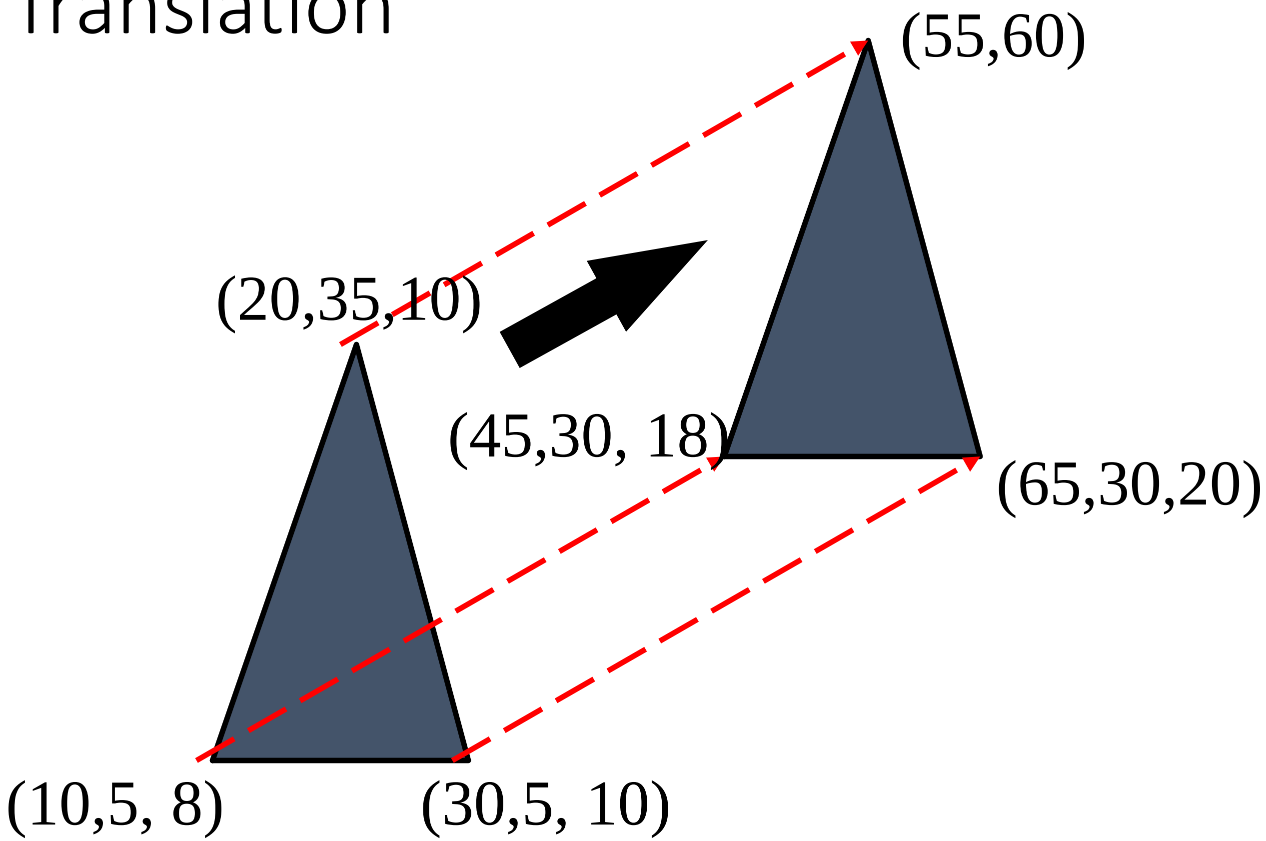
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$V=(t_x\,\mathbf{i} + t_y\,\mathbf{j} + t_z\,\mathbf{k}) == (t_x, t_y, t_z)$$



translation along y,
or V = (0, k, 0)

Translation



$$x' = x + t_x$$

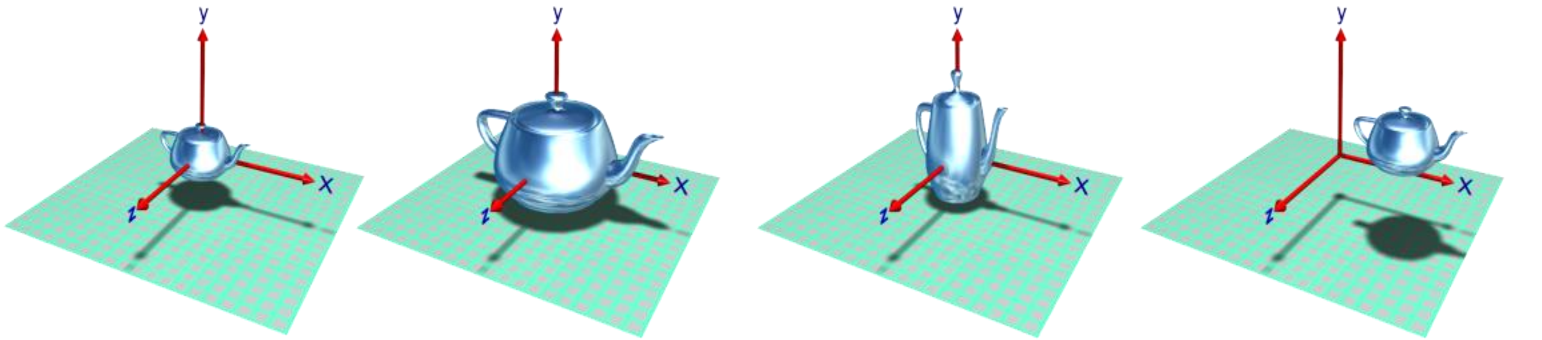
$$y' = y + t_y$$

$$z' = z + t_z$$

Just assume (not actual 3-D image)

The vector (t_x, t_y, t_z) is called the *offset vector*.

Scaling



Original | scale all axes | scale Y axis | offset from origin

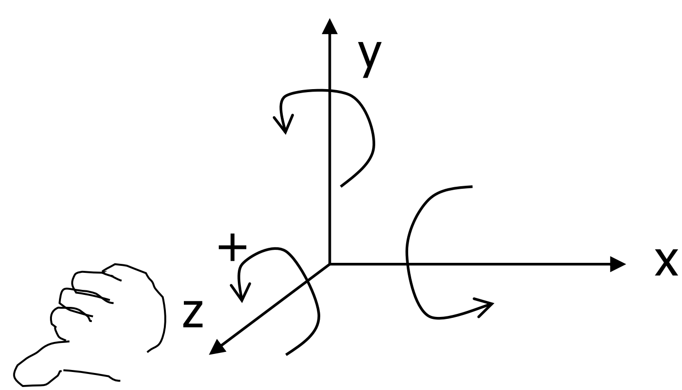
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotation: Rotating in 3D

- Cannot do mindless conversion like before
- Why?
  - Rotate about what axis?
  - 3D rotation: about a defined axis
  - Different Xform matrix for:
    - Rotation about x-axis
    - Rotation about y-axis
    - Rotation about z-axis
- In general, rotations are specified by a rotation axis and an angle. In 2-D, there is only one choice of rotation axis that leaves points in the plane.

Rotating in 3D

- New terminology
  - X-roll: rotation about x-axis
  - Y-roll: rotation about y-axis
  - Z-roll: rotation about z-axis
- Which way is +ve rotation
  - Look in –ve direction (into +ve arrow)
  - CCW is +ve rotation



3D Rotation

- The easiest rotation axes are those that parallel to the coordinate axis.
- Positive rotation angle produce counter clockwise rotations about a coordinate axis, if we are looking along the positive half of the axis towards the coordinate origin.

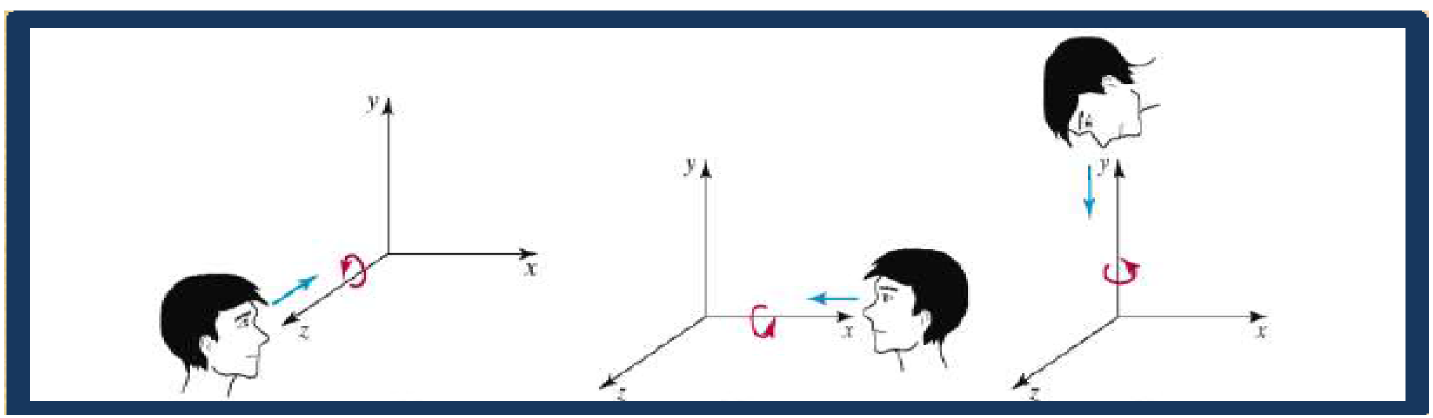


Fig: 3D Rotation

Coordinate Axis Rotations

- Obtain rotations around other axes through cyclic permutation of coordinate parameters:

$$x \rightarrow y \rightarrow z \rightarrow x$$

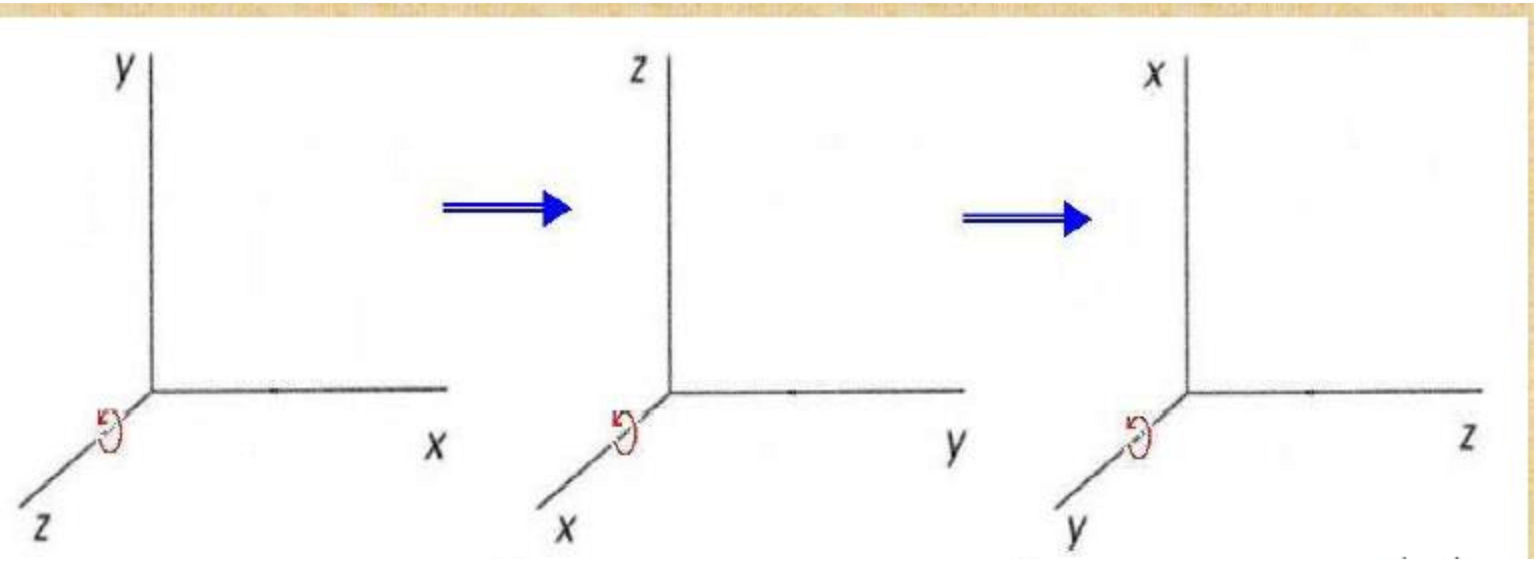


Fig: Coordinate axis rotation

Coordinate Axis Rotations

Z-axis rotation: For z axis same as 2D rotation:

x'=x*cos θ-y*sin θ

Y'=x*sin θ +y*cos θ

Z'=z

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

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

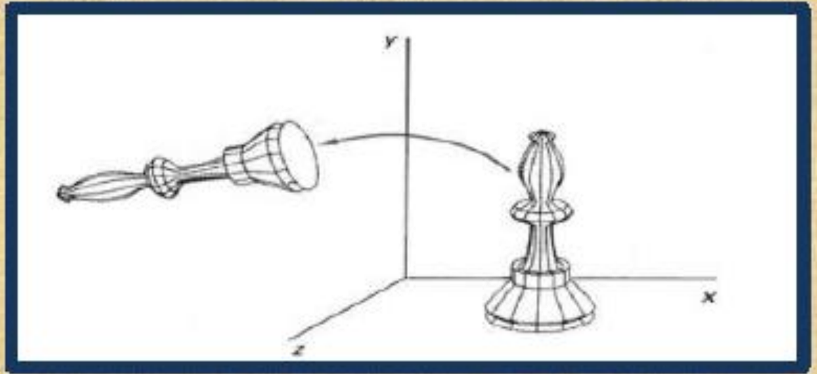


Fig: Z-axis rotation

Coordinate Axis Rotations

- X- axis rotation:

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

$$x' = x$$
$$y' = y\cos\boldsymbol{\theta} - z\sin\boldsymbol{\theta}$$
$$z' = y\sin\boldsymbol{\theta} + z\cos\boldsymbol{\theta}$$

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

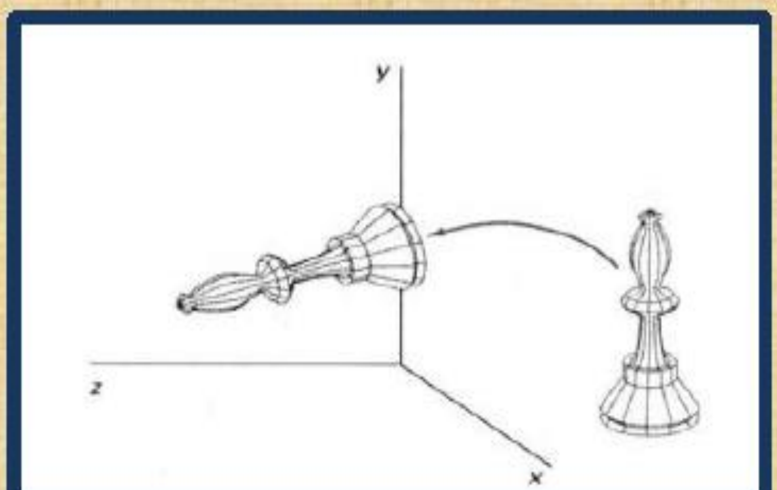


Fig: X-axis rotation

Coordinate Axis Rotations

Y-axis rotation:

Z’=z*cos θ -x*sin θ

X’=z*sin θ +x*cos θ

Y’=y

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad \mathbf{P}' = \mathbf{R}_y(\theta) \cdot \mathbf{P}$$

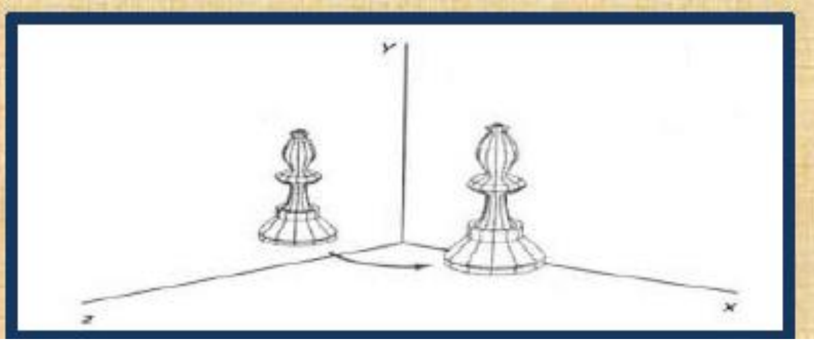


Fig: Y-axis rotation

Reflection

- Reflection in computer graphics is used to emulate reflective objects like mirrors and shiny surfaces
- Reflection may be an x-axis, y-axis, z-axis, and also in the planes xy-plane, yz-plane, and zx-plane.
- Reflection relative to a given axis are equivalent to 180 degree rotations.

Fig: Reflection

3D Reflection

Reflection about x-axis:-

$x' = x \qquad y' = -y \qquad z' = -z$

$$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

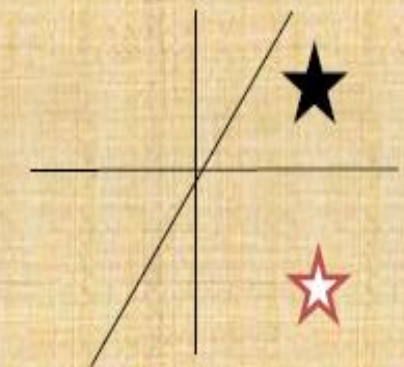


Fig: X axis reflection

Reflection about y-axis:-

$y' = y \qquad x' = -x \qquad z' = -z$

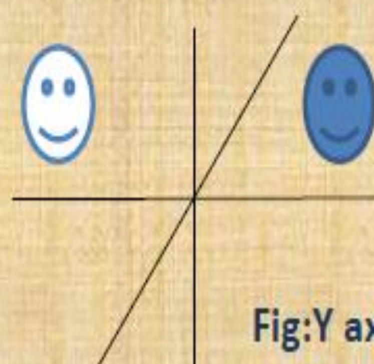


Fig:Y axis reflection

3D Reflection

- The matrix for reflection about y-axis:-

$$\begin{matrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

- **Reflection about z-axis:-**

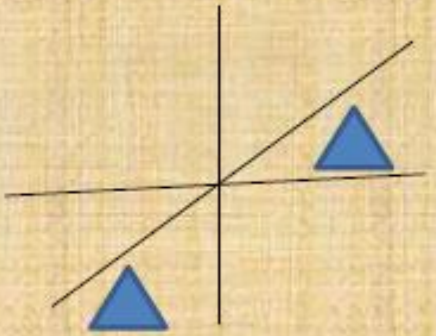


Fig: Z axis reflection

$x'=-x$ $y'=-y$ $z'=z$

$$\begin{matrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Shearing

- A Transformation that distorts the shape of an object such that the transformed shape appears as if the object were decomposed of internal layers that had been caused to slide over each other is called a shearing
- In 2-D, transformations relative to X or Y axis to produce distortion in the shapes of objects. In 3-D, we can also generate shears relative to Z-axis.

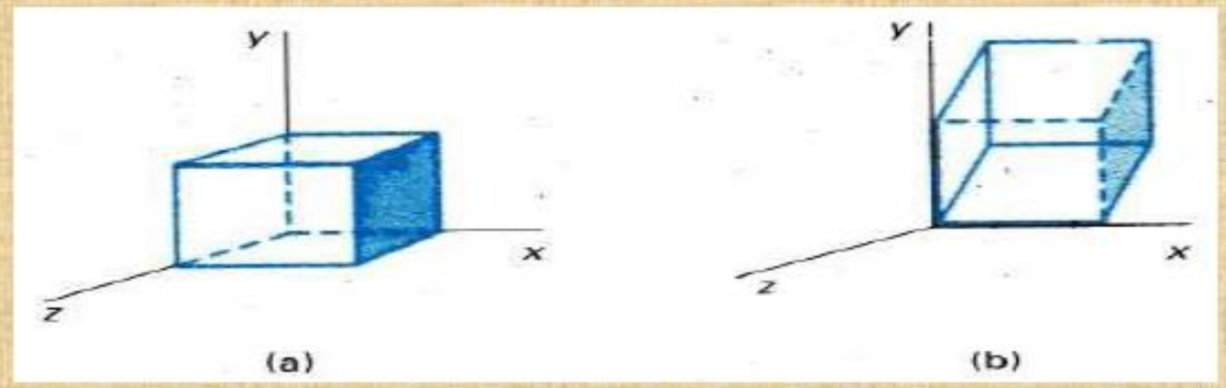
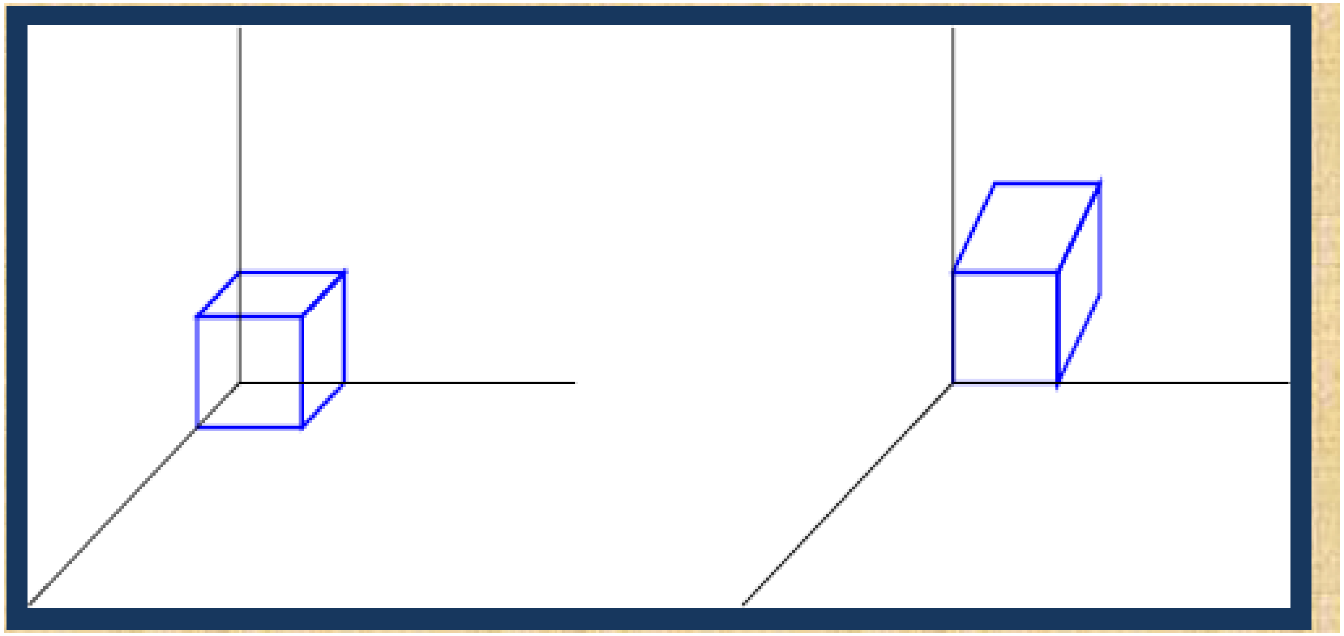


fig: before shearing fig: after shearing

Shearing

- Modify object shapes
- Useful for perspective projection:
  - Draw a cube (3D) on a screen (2D)
  - Alter the values of x and y by an amount proportional to the distance from Z ref.
  - Shearing factors: Shx, Shy, Shz



Shearing

- Shearing along x-axis: only change with y and z coordinate

$$x' = x$$

$$y' = y + Sh_y x$$

$$z' = z + Sh_z x$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ Sh_y & 1 & 0 & 0 \\ Sh_z & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Shearing

- Shearing along y-axis: only change with x and z coordinate

$$x' = x + Sh_x y$$

$$y' = y$$

$$z' = z + Sh_z y$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & Sh_x & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & Sh_z & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Shearing

- Shearing along z-axis: only change with x and y coordinate

$$x' = x + Sh_x z$$

$$y' = y + Sh_y z$$

$$z' = z$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & Sh_x & 0 \\ 0 & 1 & Sh_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} . \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Composite 3D Transformations

- **Same way as in two dimensions:**
  - Multiply matrices
  - Rightmost term in matrix product is the first transformation to be applied

Thank You