THREE-DIMENSIONAL TRANSFORMATIONS
(Chapter 11 in *Computer Graphics*)

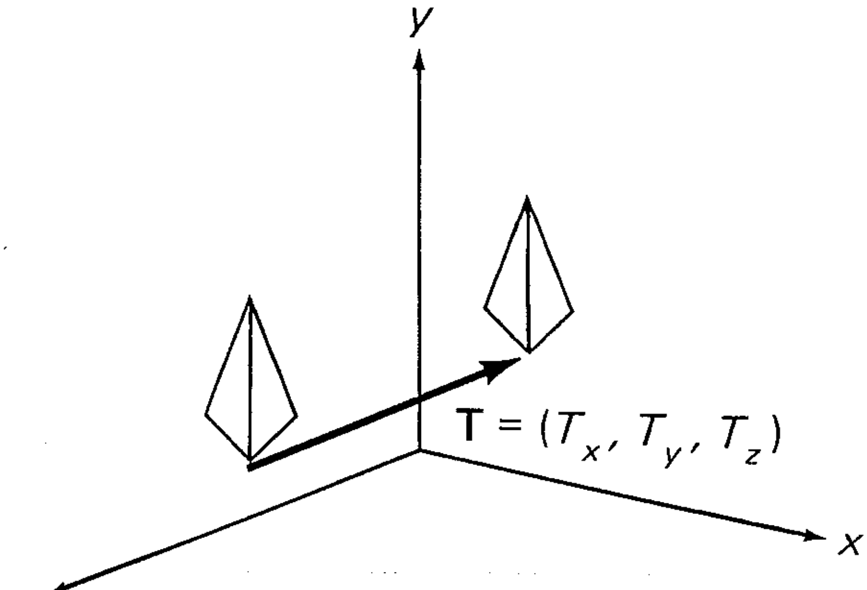
- Translation
- Scaling
- Rotation
- Rotation About an Arbitrary Axis
- Other Transformations
- Transformations of Coordinate Systems
- Transformations Commands

Translation

- in three-dimensional homogeneous coordinate space, a point is translated from (x, y, z) to (x', y', z')

$$[x'\ y'\ z'\ 1] = [x\ y\ z\ 1]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Tx & Ty & Tz & 1 \end{bmatrix}$$

- in three-dimensional homogeneous coordinate space, a object is translated by translating each vertex



- the inverse translation matrix

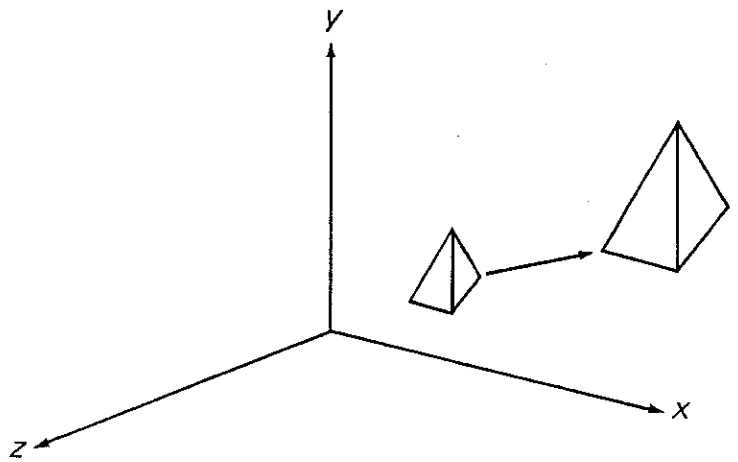
$$[x'\ y'\ z'\ 1] = [x\ y\ z\ 1]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ & & & 1 \end{bmatrix}$$

Scaling

- in three-dimensional homogeneous coordinate space, an object is scaled relative to the origin of coordinates

$$[x' \; y' \; z' \; 1] = [x \; y \; z \; 1] \begin{bmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- the scaling operation scales and translates objects relative to the coordinate origin
  - example (Sx = Sy = Sz = 2)

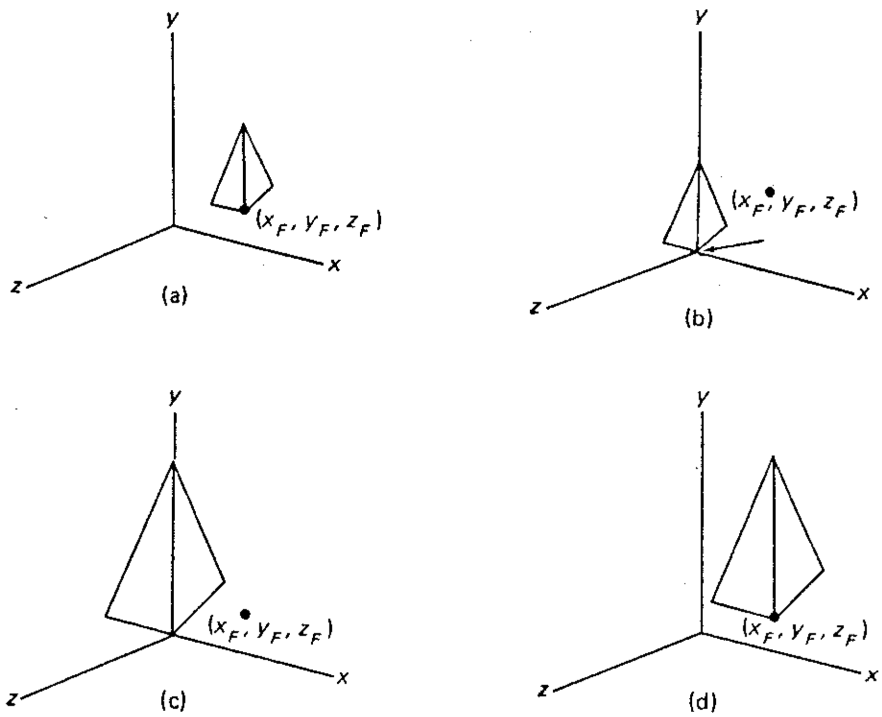


- the inverse scaling matrix

$$[x' \; y' \; z' \; 1] = [x \; y \; z \; 1] \begin{bmatrix} & 0 & 0 & 0 \\ 0 & & 0 & 0 \\ 0 & 0 & & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

scaling relative to a fixed position

- translate the fixed point to the origin
- scale
- translate the fixed point back to its original position

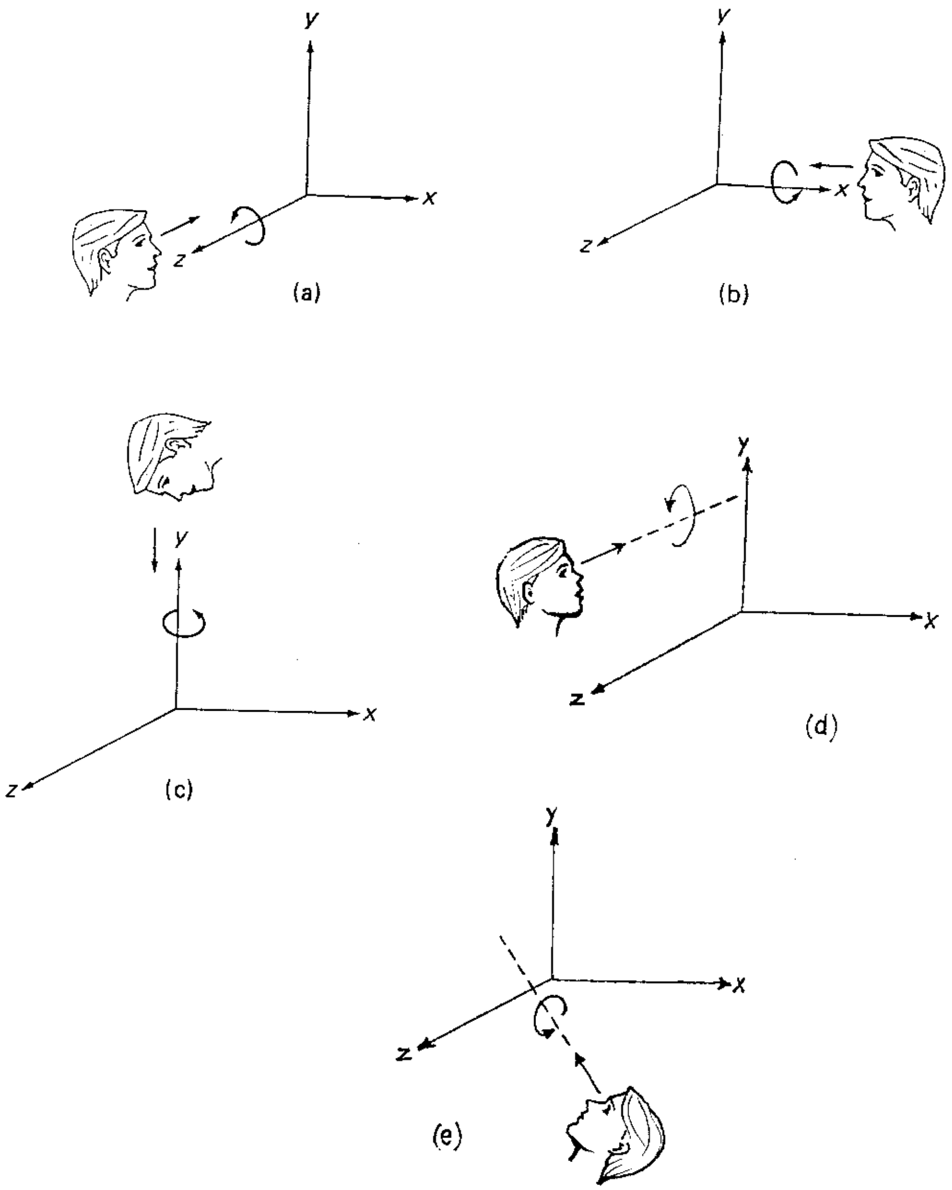


$$T(-x_F,\ -y_F,\ -z_F)\cdot S(Sx,\ Sy,\ Sz)\cdot T(x_F,\ y_F,\ z_F)$$

$$= \begin{bmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ (1 - Sx)x_F & (1 - Sy)y_F & (1 - Sz)z_F & 1 \end{bmatrix}$$

Rotation

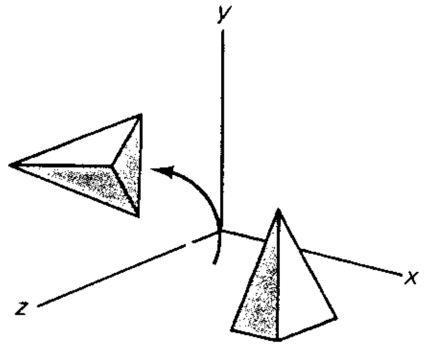
- designate an axis of rotation and an angle of rotation
- our convention: positive rotation is counterclockwise



rotation about the principal axes

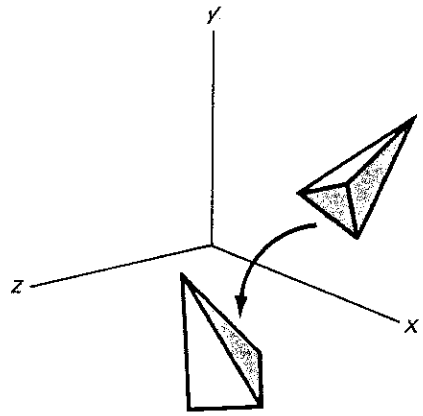
- z-axis rotation

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



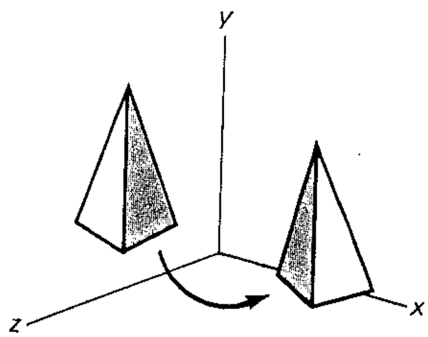
- x-axis rotation

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



- y-axis rotation

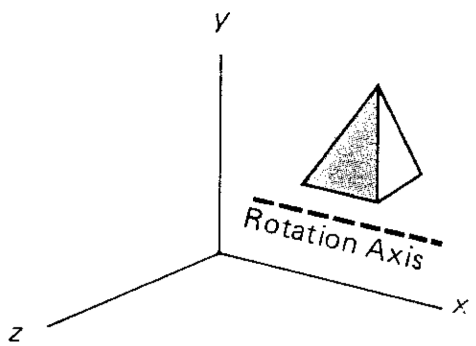
$$[x'\ y'\ z'\ 1] = [x\ y\ z\ 1]\begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



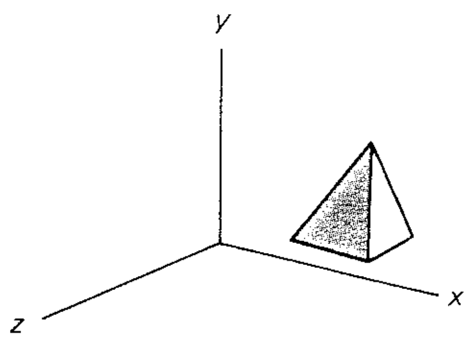
- inverse rotation matrices

rotation about an axis parallel to a principal axis

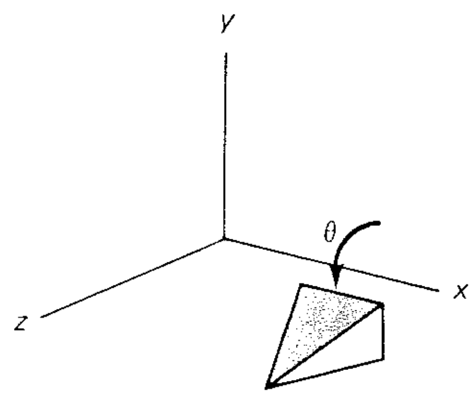
- translate the object so the rotation axis coincides with a principal axis
- rotate through the specified angle
- translate the object so the rotation axis returns to its original position



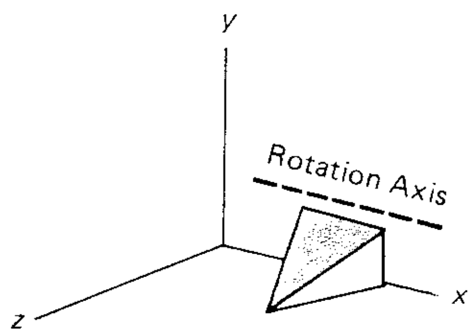
(a)
Original Position of Object



(b)
Translate Rotation Axis onto x axis

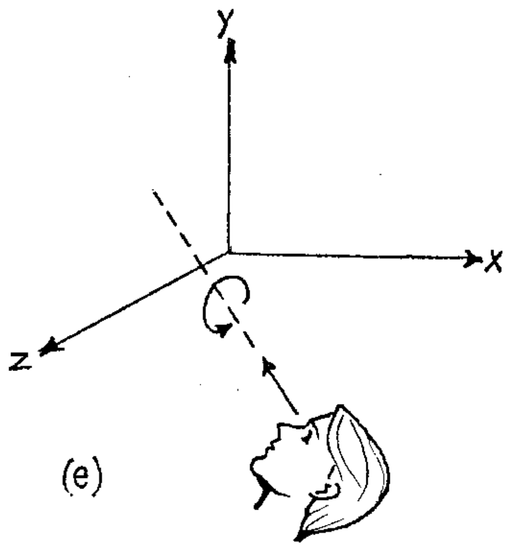


(c)
Rotate Object Through Angle θ



(d)
Translate Rotation
Axis to Original Position

rotation about an arbitrary axis

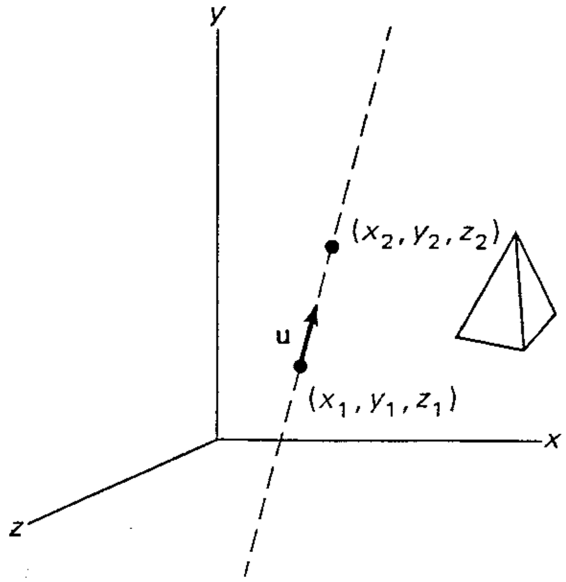


(e)

- translate the object so the rotation axis passes through the origin of coordinates
- rotate the object so the rotation axis coincides with a principal axis
- rotate through the specified angle
- return the rotation axis to its original orientation
- return the rotation axis to its original position

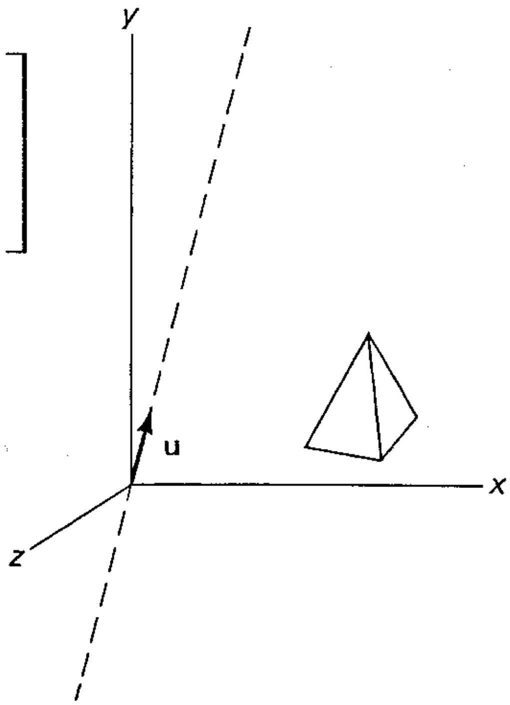
rotation about an arbitrary axis, continued

- assume the axis of rotation is defined by two points



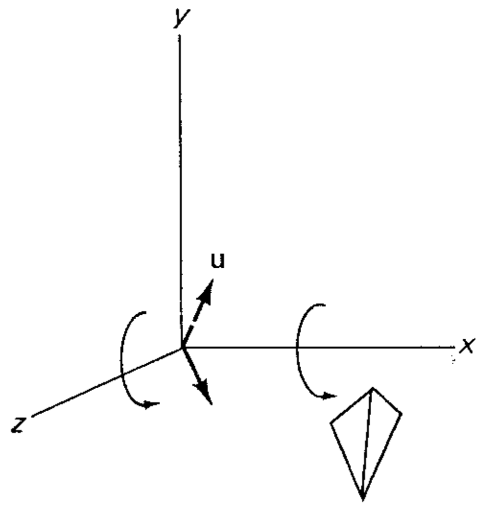
- translate the object so the rotation axis passes through the origin of coordinates

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_1 & -y_1 & -z_1 & 1 \end{bmatrix}$$



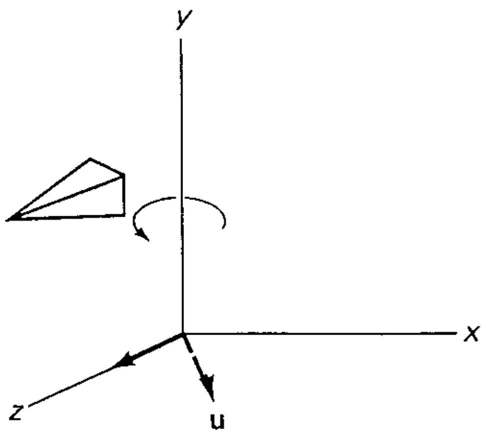
rotation about an arbitrary axis, continued

- rotate the object so the rotation axis coincides with the z axis
  - rotate about the x axis to place the rotation axis in the xz plane



(a)

  - rotate about the y axis to align the rotation axis with the z axis



(b)

rotation about an arbitrary axis, continued

- rotate through the specified angle about the z axis

$$Rz(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- return the rotation axis to its original orientation
- return the rotation axis to its original position
- concatenate transformations

$$R(\theta) = T \cdot Rx(\alpha) \cdot Ry(\beta) \cdot Rz(\theta) \cdot Ry^{-1}(\beta) \cdot Rx^{-1}(\alpha) \cdot T^{-1}$$

Other Transformations

- reflections
- shears
- transformations of coordinate systems

reflections

- about a specified plane
- example: conversion from a right-handed coordinate system to a left-handed coordinate system (or vice versa)

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

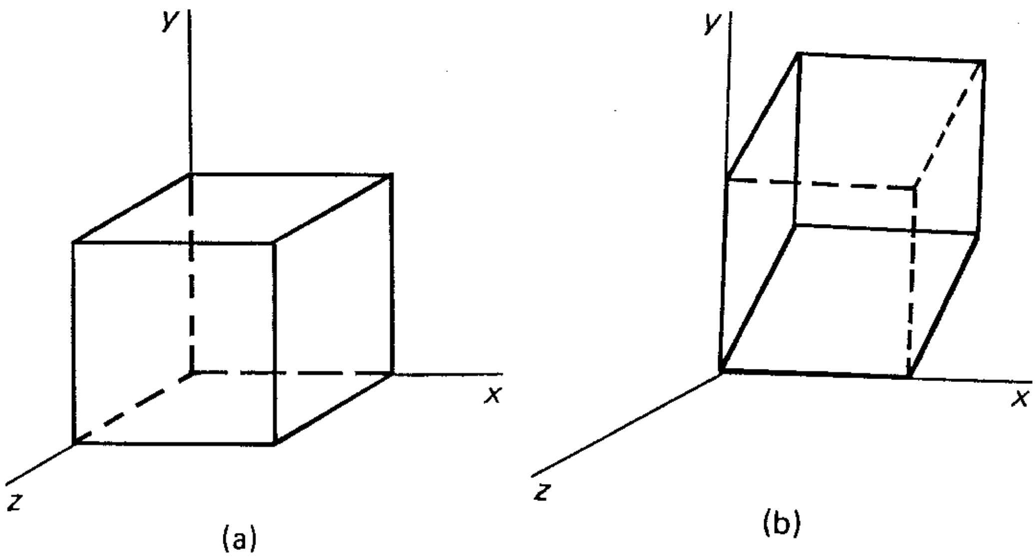
- reflections about arbitrary planes are extensions of two-dimensional reflections about arbitrary lines

shears

- transform two of the three coordinate values of defined points
- example: z-axis shear (a = b = 1)

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

  - effect: alter x- and y- coordinate values by an amount proportional to z, leaving z unchanged



Shears, continued

alter any coordinate value by an amount proportional to either or both of the other two coordinates

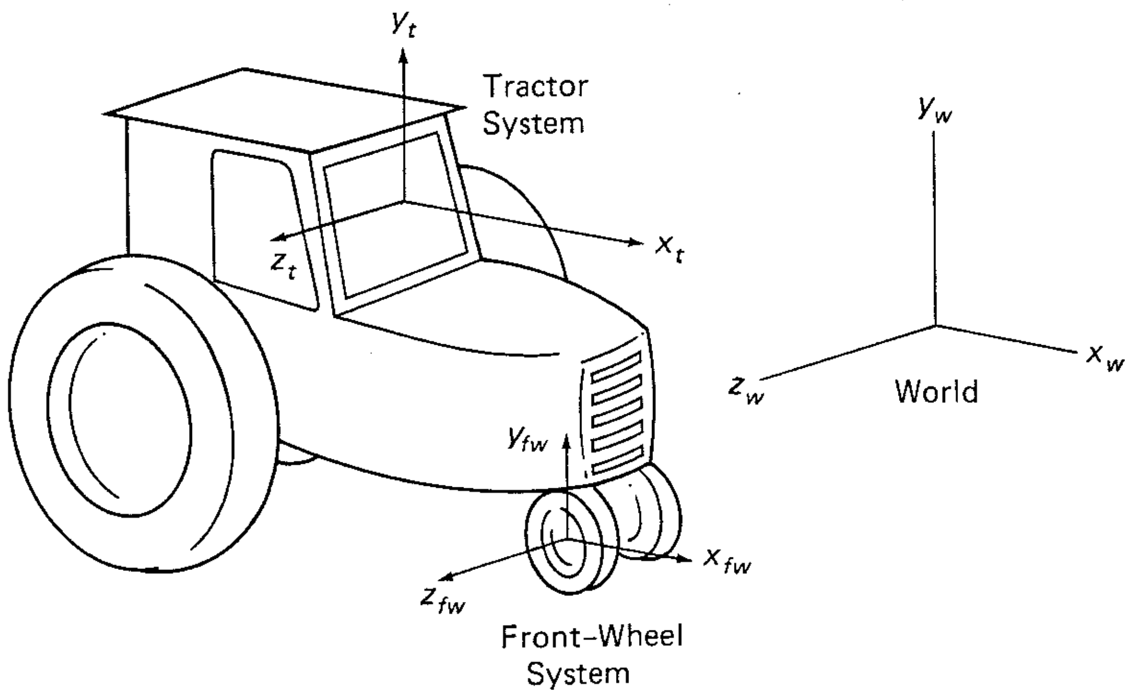
$$\begin{bmatrix} 1 & Syxx & Szxx & 0 \\ Sxxy & 1 & Szxy & 0 \\ Sxxz & Syxz & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

transformations of coordinate systems

- to build a scene, objects can be defined in one coordinate system and placed in a second coordinate system
- objects can be transformed relative to one coordinate system which is transformed relative to another coordinate system
- transform objects by mapping one set of coordinate axes onto the other
  - translate so the origins coincide
  - rotate to superimpose axes
  - scale as needed

example of multiple coordinate systems

- front wheel rotations are described in the front-wheel coordinate system
- the front-wheel coordinate system is described in the tractor coordinate system
- the tractor coordinate system is described in the world coordinate system



Transformation Commands

- simple extensions of two-dimensional commands
  - create_transformation_matrix
  - accumulate_transformation_matrix

  or
- provide separate functions
  - create_translation_matrix_3 (tx, ty, tz, t)
  - create_scaling_matrix_3 (sx, sy, sz, s)
  - create_x_rotation_matrix_3 (a, rx)
  - create_y_rotation_matrix_3 (a, ry)
  - create_z_rotation_matrix_3 (a, rz)
  - accumulate_matrices_3 (m1, m2, m)
  - set_segment_transformation_3 (id, m)

THREE-DIMENSIONAL TRANSFORMATIONS

- Translation
- Scaling
- Rotation
- Rotation About an Arbitrary Axis
- Other Transformations
- Transformations of Coordinate Systems
- Transformations Commands