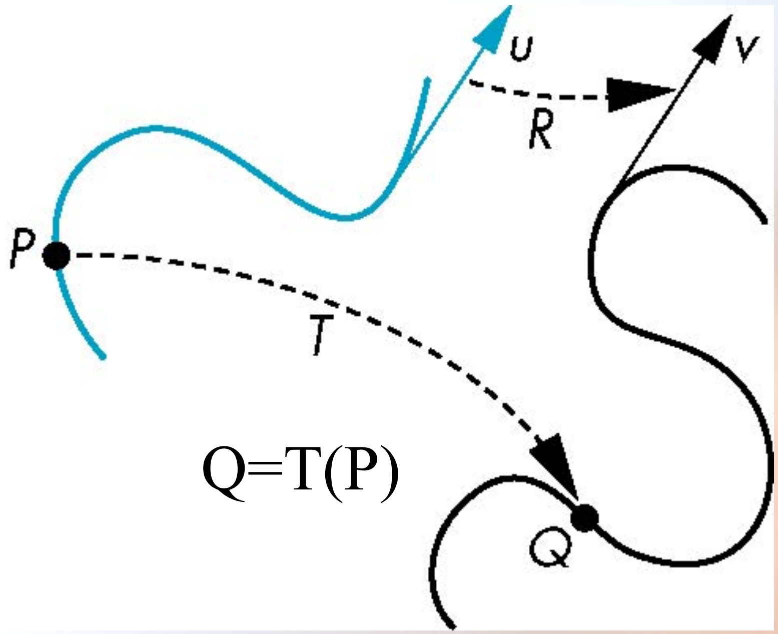
Computer Graphics

Transformations

Based on Slides by Dianna Xu, Bryn Mawr College

General Transformations

- **A transformation maps points to other points and/or vectors to other vectors**



v=T(u)

Q=T(P)

Objects and Transformations

- Objects are made out of (many) polygons
- Defined by ordered list of vertices (points).
- A transformation is a function that maps a point into another
- All transformations operate as simple changes on vertex-coordinates (2D or 3D).

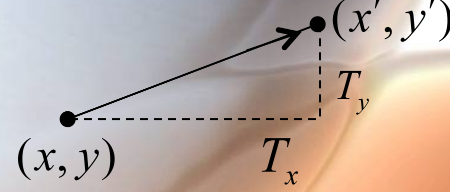
Affine Transformations

- Line preserving
- Characteristic of many physically important transformations
  - Rigid body transformations: rotation, translation
  - Scaling, shear

Geometric Transformations

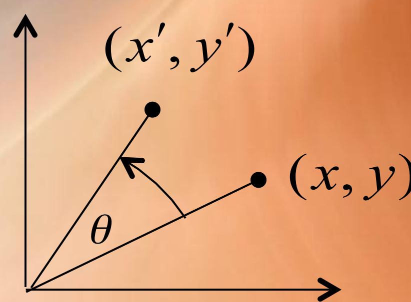
- **Translation**

$$x' = x + T_x$$
$$y' = y + T_y$$



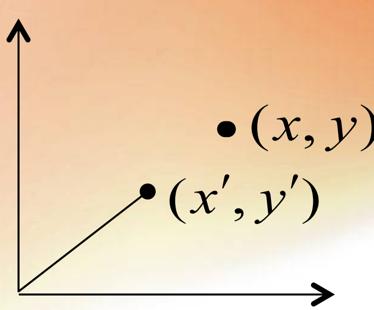
- **Rotation**

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



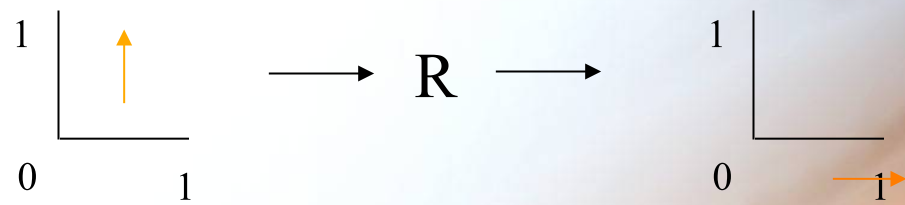
- **Dilation (scaling)**

$$x' = S_x x$$
$$y' = S_y y$$

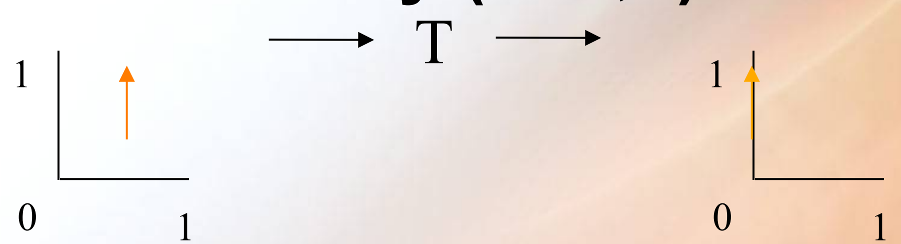


Transformations do not Commute

- Let R = rotation clockwise by 90°

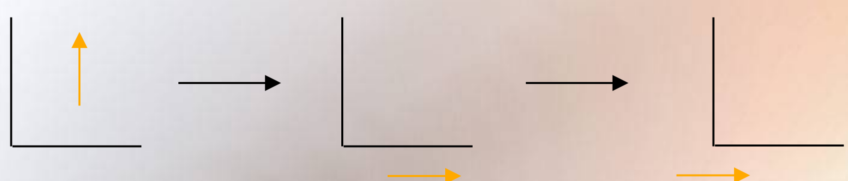


- Let T= translation by (-0.5,0)



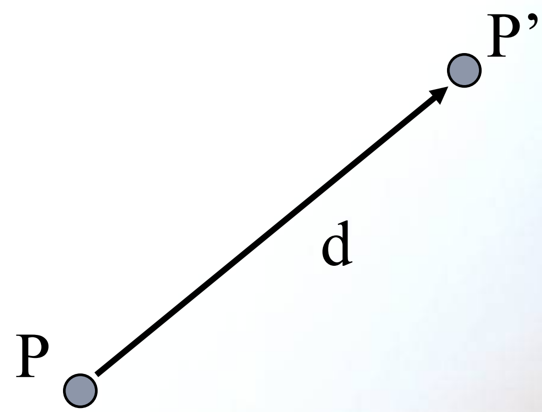
- TR=

- RT=



Translation

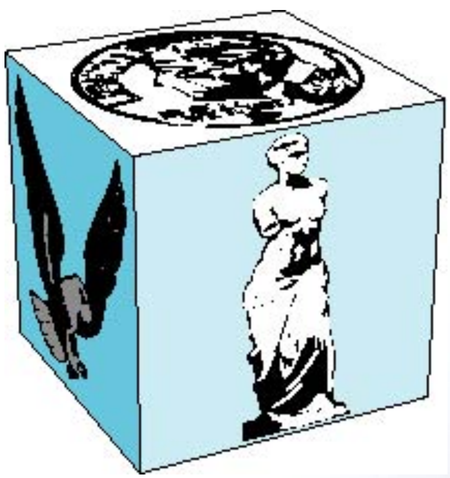
- **Move (translate, displace) a point to a new location**



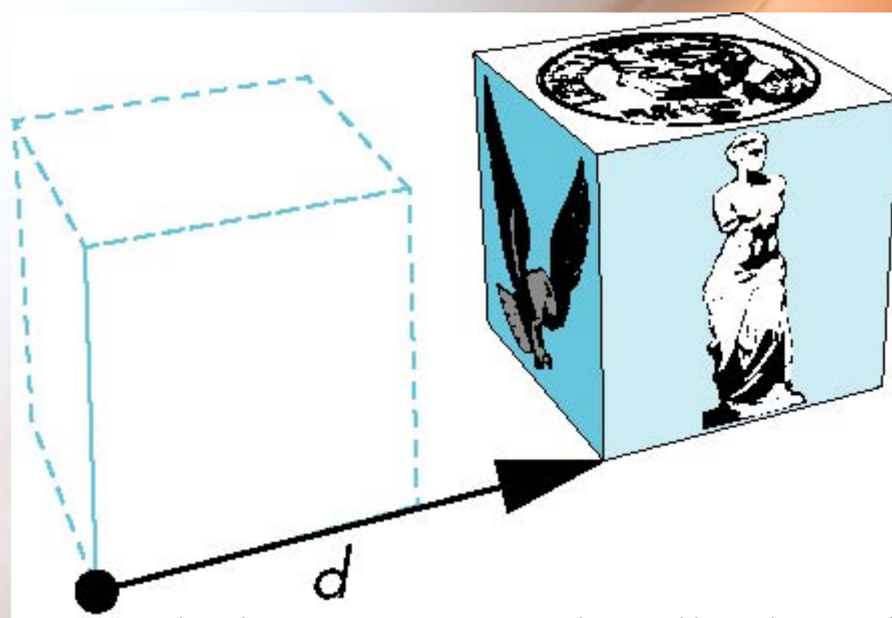
- **Displacement determined by a vector d**
  - **Three degrees of freedom**
  - **P'=P+d**

How many ways?

Although we can move a point to a new location in infinite many ways, when we move many points there is usually only one way



object



translation: every point displaced by same vector

Translation Using Representations

Using the 2D homogeneous coordinate representation in some frame

$$P = (x, y, 1)$$

$$P' = (x', y', 1)$$

$$d = (d_x, d_y, 0)$$

$$P' = P + d \Leftrightarrow \begin{array}{l} x = x' + d_x \\ y = y' + d_y \end{array}$$

Matrix Multiplications

- The (i,j) entry of *AB* is the dot product of the *i*-th row of *A* and j-th column of B

$$AB_{ij} = \sum_{k=1}^{m} A_{ik} B_{kj}$$

$$\begin{bmatrix} 2 & -2 \\ 2 & -2 \\ 2 & -2 \\ 2 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & -1 \end{bmatrix}$$

The Identity Matrix

- The identity Matrix I is a square matrix such that

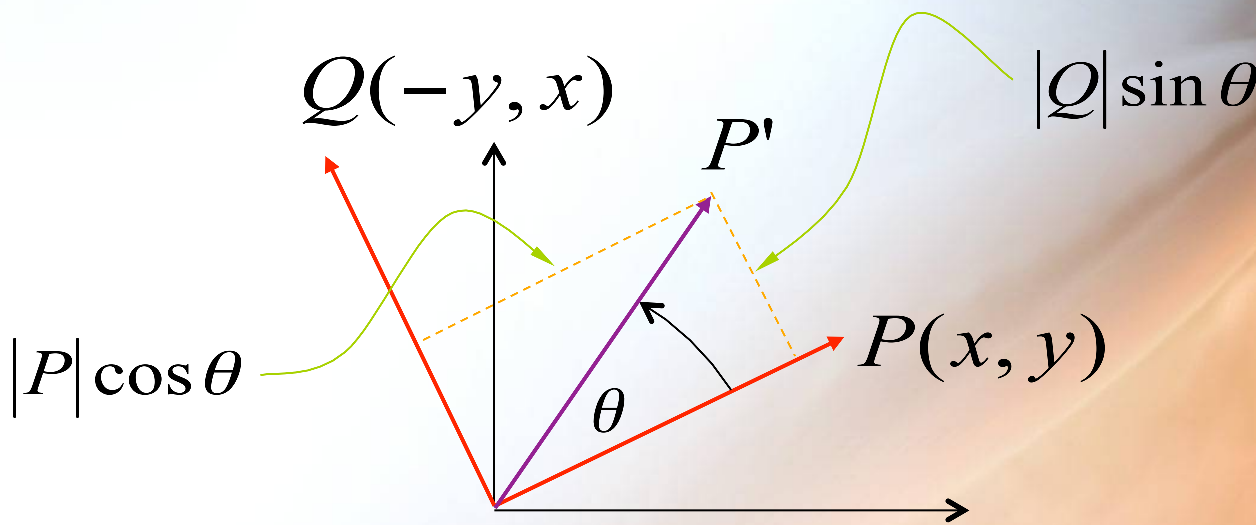
$$MI_n = I_n M = M$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Given a Matrix M, the inverse of M is defined as

$$MM^{-1} = M^{-1}M = I_n$$

Understanding Rotation



$$P' = P\cos\theta + Q\sin\theta$$

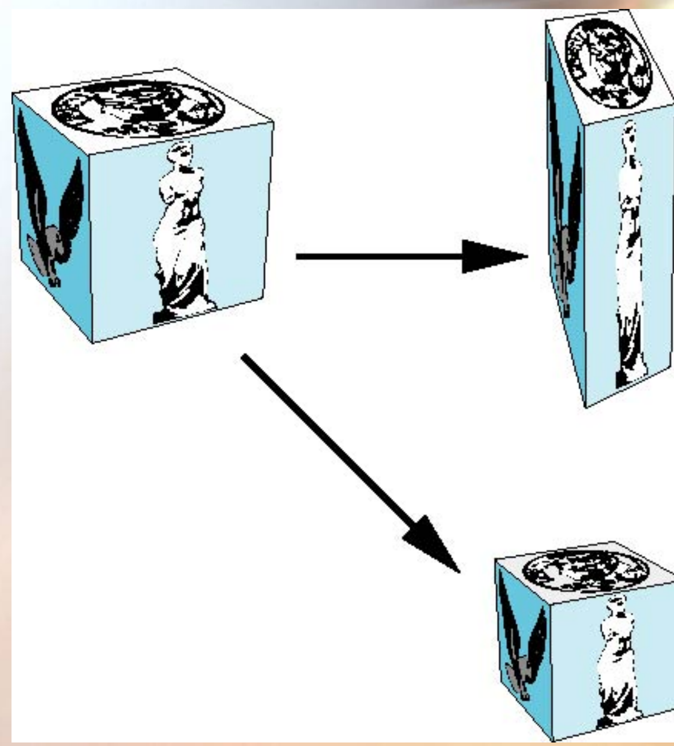
$$P'_x = P_x\cos\theta - P_y\sin\theta$$

$$P'_y = P_y\cos\theta + P_x\sin\theta$$

Scaling

- Expand or contract along each axis (fixed point of origin)

$$
\begin{aligned}
x &= s_x x \\
y &= s_y y
\end{aligned}
\Leftrightarrow P' = SP
$$

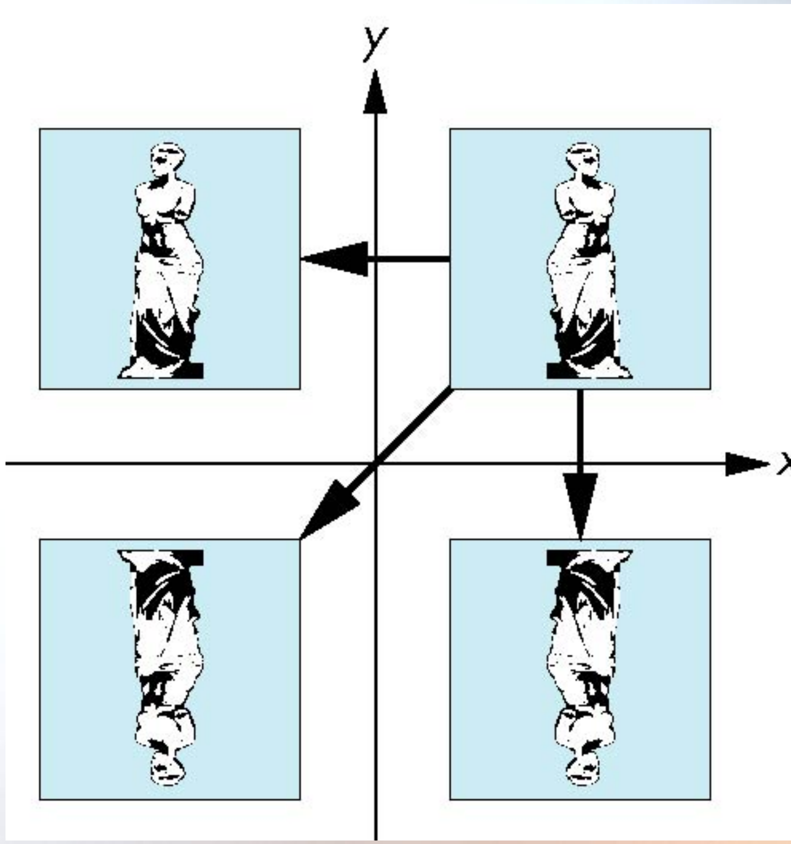


Reflection

- **corresponds to negative scale factors**

$s_x = -1 \; s_y = 1$

original



$s_x = -1 \; s_y = -1$

$s_x = 1 \; s_y = -1$

Matrix Representations in Homogenous Coordinates

- **Translations**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- **Rotation about origin**

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

- **Scale about origin**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Multiple Transformations: Concatenation

- [new]= [transform *n*] … [transform 2] [transform 1] [old]

$$P' = T_n \ldots T_2 T_1 P$$

  - Inefficient

$$P' = T_n \ldots T_2 \left(T_1 P \right) \quad P' = T_n \left(\ldots \left(T_2 \left(T_1 P \right) \right) \right)$$

  - Efficient

$$P' = (T_n \ldots T_2 T_1) P \qquad P' = \mathrm{T} P$$

Combine into Single Matrix

Since we usually have many vertices to transform, compute once:

$$T = (T_n \ldots T_2 T_1)$$

and each new point is a simple matrix-vector product:

$$p_i' = \mathrm{T} p_i \text{ for } i = 0,1,2,\ldots n-1$$

Rotation about the z axis

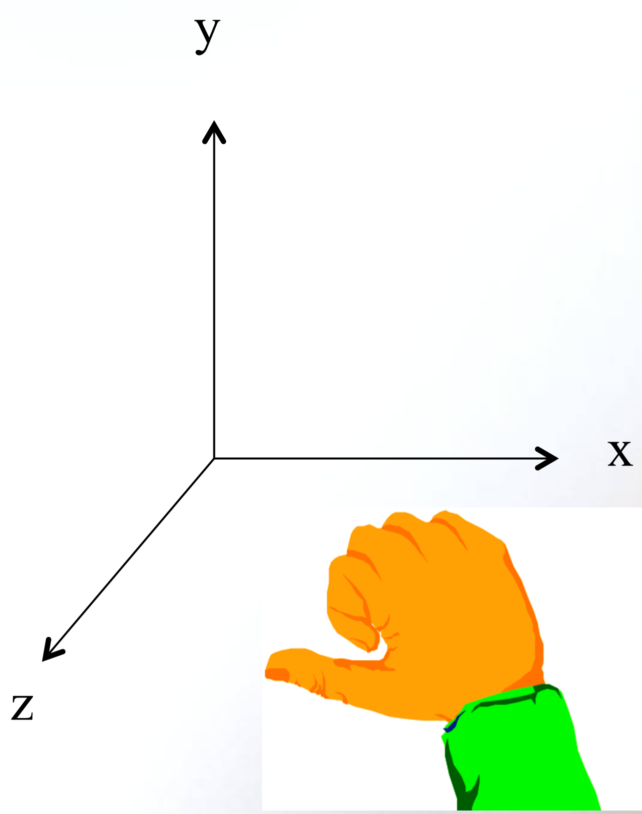
- Rotation about z axis in three dimensions leaves all points with the same z
  - Equivalent to rotation in two dimensions in planes of constant z

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

  - or in homogeneous coordinates

$$p' = R_z(\theta)p$$

3D Geometry



Right handed coordinate system

3D Transformations with Homogeneous Coordinates

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ W' \end{bmatrix} = \begin{bmatrix} A_{00} & A_{01} & A_{02} & A_{03} \\ A_{10} & A_{11} & A_{12} & A_{13} \\ A_{20} & A_{21} & A_{22} & A_{23} \\ A_{30} & A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix}$$

$$x = X' / W'$$
$$y = Y' / W'$$
$$z = Z' / W'$$

- **Used because:**
  - **Uniform representation for all common transformations**
  - **Easy to manipulate with matrix algebra**

Scaling and Translation Transformation Matrices

Scaling

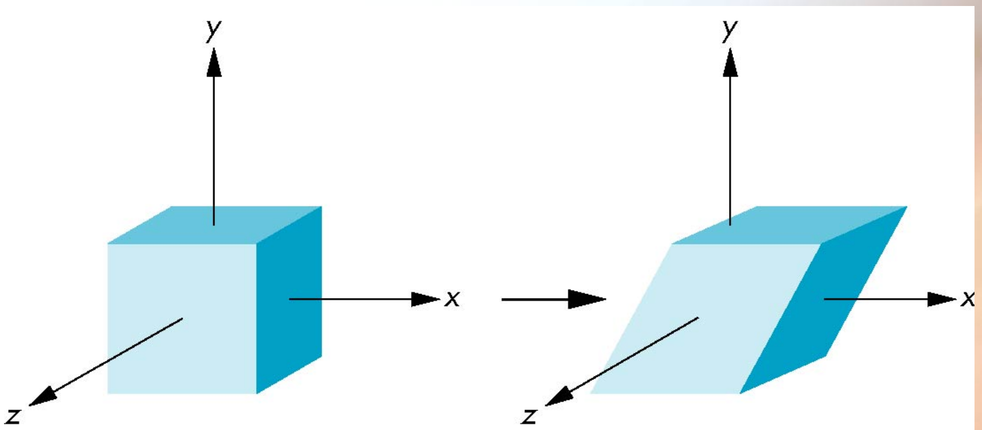
$$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Translation

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Shear

- Helpful to add one more basic transformation
- Equivalent to pulling faces in opposite directions



Shearing Transformation Matrix

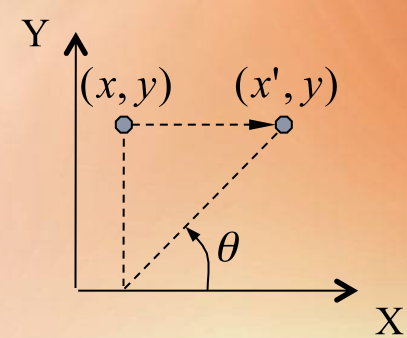
- **It is scaling restricted to one axis**
- **x-axis example**

$$x' = x + y\cot(\theta)$$

$$y' = y$$

$$z' = z$$

$$\begin{bmatrix} 1 & \cot\theta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



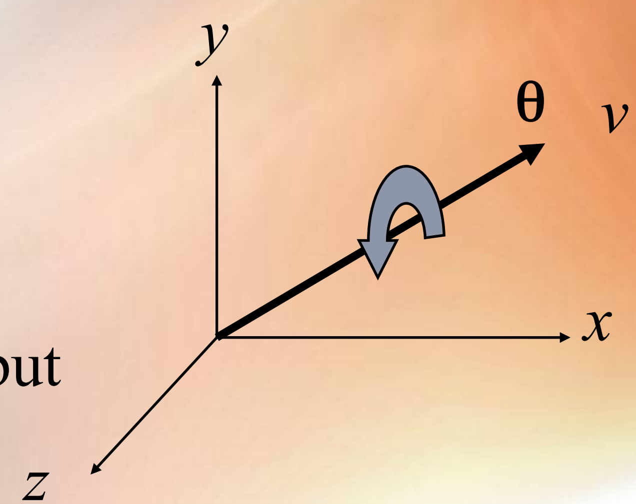
General Rotation About the Origin

- A rotation by θ about an arbitrary axis can be decomposed into the concatenation of rotations about the *x*, *y* and *z* axes

$$\mathbf{R}(\theta) = \mathbf{R}_z(\theta_z)\,\mathbf{R}_y(\theta_y)\,\mathbf{R}_x(\theta_x)$$

$\theta_x\ \theta_y\ \theta_z$ are called the Euler angles

Note that rotations do not commute
We can use rotations in another order but with different angles



Rotation Transformation Matrix

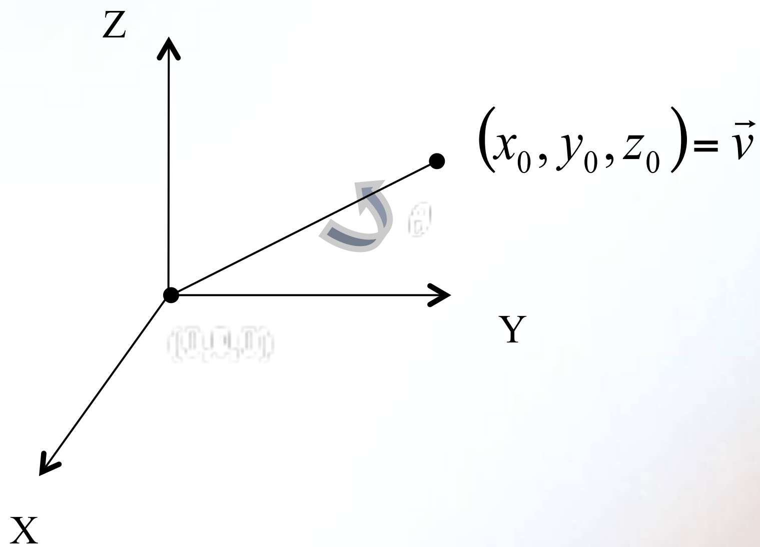
- **Counter-clockwise rotation around individual axes:**

Rx(α) Ry(β) Rz(γ)

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos a & -\sin a \\ 0 & \sin a & \cos a \end{bmatrix} \quad \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \quad \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- **Any rotation can be given as a composition of rotations about the three axes**

Rotation about an Arbitrary Axis



Length of $\vec{v}$

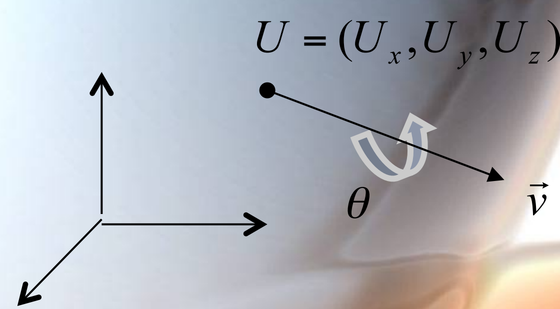
$$|\vec{v}| = \sqrt{x_0^{\,2} + y_0^{\,2} + z_0^{\,2}}$$

a, b, c are direction cosines: $a = \dfrac{x_o}{|\vec{v}|}$ $\quad b = \dfrac{y_o}{|\vec{v}|}$ $\quad c = \dfrac{z_o}{|\vec{v}|}$

General Rotation Transformation Matrix

$$\begin{bmatrix} a^2+(1-a^2)\cos\theta & ab(1-\cos\theta)-c\sin\theta & ac(1-\cos\theta)+b\sin\theta & 0 \\ ab(1-\cos\theta)+c\sin\theta & b^2+(1-b^2)\cos\theta & bc(1-\cos\theta)-a\sin\theta & 0 \\ ac(1-\cos\theta)-b\sin\theta & bc(1-\cos\theta)+a\sin\theta & c^2+(1-c^2)\cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

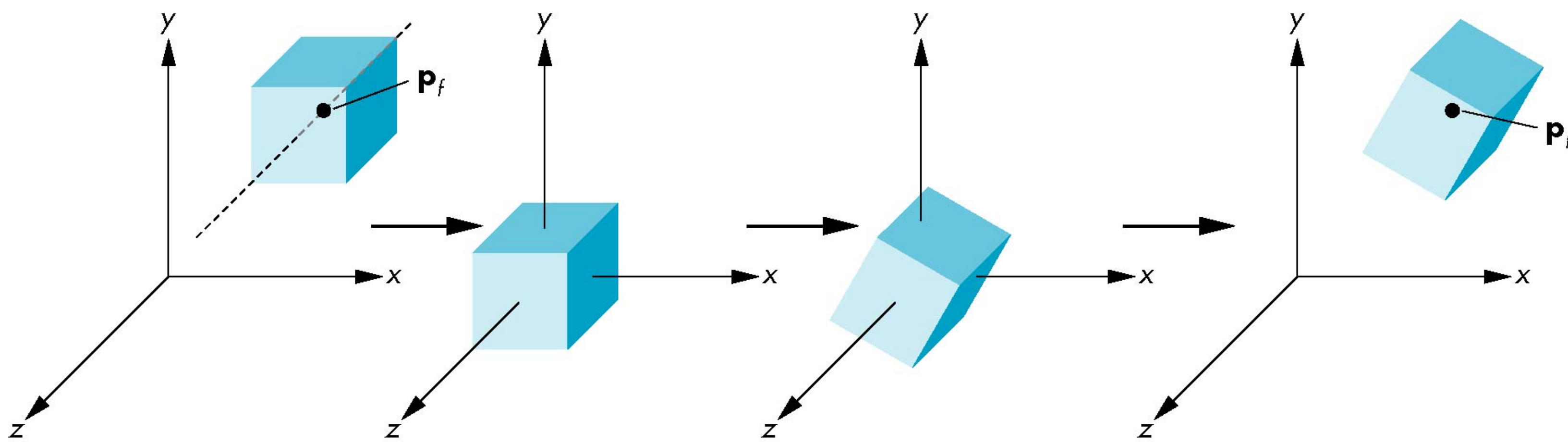
What if Axis does not go through Origin?



1. Translate to origin: $T(-U_x, -U_y, -U_z) = T(-U)$
2. Do rotation: $R(\theta)$
3. Translate back: $T(U_x, U_y, U_z) = T(U)$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = T(U)R(\theta)T(-U)\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotation About a Fixed Point other than the Origin



General Transformation Matrix

- **Combining rotation and translation**

$$Tr = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

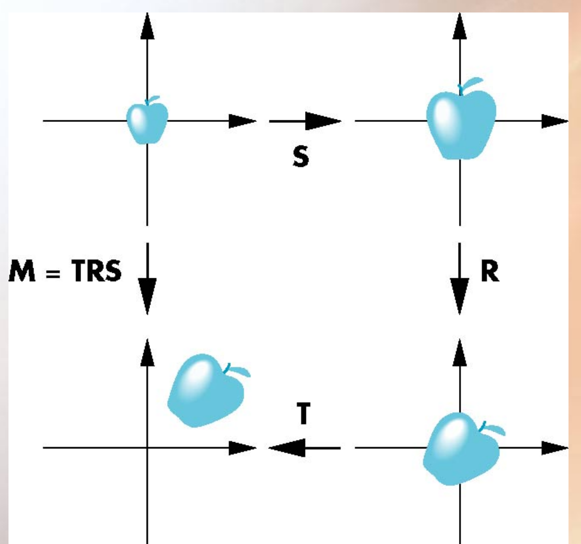
Instancing

- In modeling, we often start with a simple object centered at the origin, oriented with the axis, and at a standard size
- We apply an *instance transformation* to its vertices to

    Scale

    Orient

    Locate



Inverses

- **Although we could compute inverse matrices by general formulas, we can use simple geometric observations**
  - **Translation:** $T^{-1}(d_x, d_y, d_z) = T(-d_x, -d_y, -d_z)$
  - **Rotation:** $R^{-1}(\theta) = R(-\theta)$
    - **Holds for any rotation matrix**
    - $\begin{array}{l} \cos(-\theta) = \cos(\theta) \\ \sin(-\theta) = -\sin(\theta) \end{array} \Leftrightarrow R^{-1}(\theta) = R^T(\theta)$
  - **Scaling:** $S^{-1}(s_x, s_y, s_z) = S(\frac{1}{s_x}, \frac{1}{s_y}, \frac{1}{s_z})$