CPSC 453:
Composing Transformations

(Chapter 4.8)

Mark Matthews
matthews@cpsc.ucalgary.ca
Office Hours: 3:15-4:00PM TR (same)
Office: 680J (same)
www.cpsc.ucalgary.ca/~sheelagh/courses/453

Composing Affine Transformations

- Composing or concatenating the transformation
- The concatenation of 2 affine transforms is also affine.

$$P'' = T_2(P') = T_2(T_1(P)) = (T_2T_1)(P) = T(P)$$
$$P'' = M_2P' = M_2M_1P = MP$$
$$M = M_2M_1$$

- Compose transforms with Matrix multiplication
- Reverse order
- Example

2D Rotation About an Arbitrary Point

- Pivot Point: $V = (V_x, V_y)$
- Is this a new matrix or a combined transform?
  - Translate through $(-V_x, -V_y)$
  - Rotate about the origin by q
  - Translate back through (V_x, V_y)
- Matrix form:

$$\begin{bmatrix} 1 & 0 & V_x \\ 0 & 1 & V_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -V_x \\ 0 & 1 & -V_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & dx \\ \sin\theta & \cos\theta & dy \\ 0 & 0 & 1 \end{bmatrix}$$

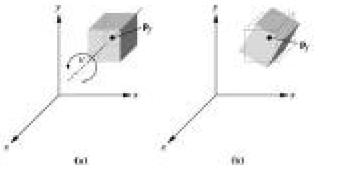
- Where: $\begin{array}{l} dx = -\cos\theta V_x + \sin\theta V Y + V_x \\ dy = -\sin\theta V_x - \cos\theta V_y + V_y \end{array}$

3D Example

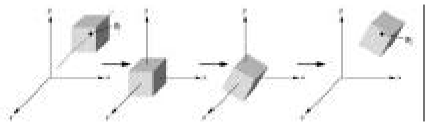
- Similar process for scaling about a pivot point
  - Scaling factors S_x, S_y, S_z
  - Pivot Point (V_x, V_y, V_z)
  - Steps:
    - 1. Translate through $(-V_x, -V_y, -V_z)$: M_1
    - 2. Scale about the origin: M_2
    - 3. Translate back through (V_x, V_y, V_z): M_3

$$M = M_3 M_2 M_1$$

3D Rotation About an Arbitrary Point

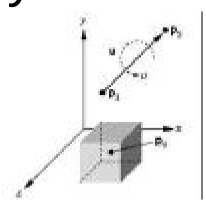


- Pivot Point P_f
- Steps:
  - Translate through $-P_f$ (M_1)
  - Rotate about the z-axis q (M_2)
  - Translate back through P_f (M_3)
  - $M = M_3 M_2 M_1$



- Matrix form: $\begin{bmatrix} \cos\theta & -\sin\theta & 0 & dx \\ \sin\theta & \cos\theta & 0 & dy \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Where: $dx = x_f - x_f \cos\theta + y_f \sin\theta$
  $dy = y_f - x_f \sin\theta - y_f \cos\theta$

Rotation About an Arbitrary Axis

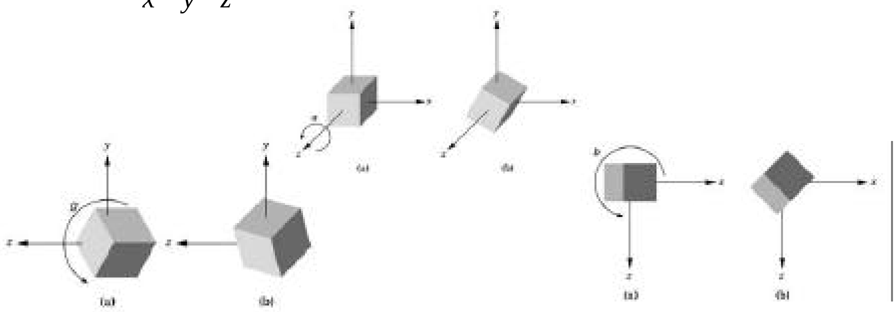


- A fixed point P_0
- An arbitrary axis $u = P_0 - P_1$
- A rotation angle θ
- A normalized version of u: $V=(ax, ay, az)$
  - Translate through $-P_0$
  - Carry out two rotations q_x and q_y to align V with the z-axis
  - Rotate q about z
  - Undo the two rotations by q_x and q_y
  - Translate through P_0
- What are q_x and q_y ?

$$M = T(P_0)R_x(-\theta_x)R_y(-\theta_y)R_z(\theta)R_y(\theta_y)R_x(\theta_x)T(-P_0)$$

General Rotation

- Three successive rotations about 3 axes
- The order is not unique
- The resulting rotation matrix is unique
- $R = R_x R_y R_z$



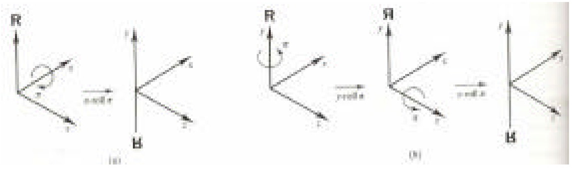
Euler Angles

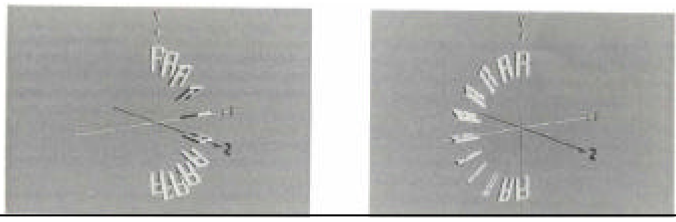
- A result of the Euler theorem
  - Any $3D$ rotation can be obtained by rolls about the x, y and z axes.
- Euler Angles
- Problem of steady rotations in $3D$

Example

- 1. A single rotation of *180º* about the *x*-axis
- 2. First a *y* roll of *180º*, then a *z* roll of 180º
- 3. Many other possibilities

case 1 **case 2**





OpenGL Transformation Matrices

- Current Transform Matrix (CTM)
- Resetting
- Premultiplication and Postmultiplication
- OpenGL's rule:
  - "The last transform specified is the first one applied"