Composing Geometric Transformations

- composition of transformations
- scene graphs

Transformations

Affine transformations
- linear transformation + translations
- can be expressed as a 3x3 matrix + 3 vector

\[ P' = M \cdot P + T \]

4x4 matrices

\[
\begin{bmatrix}
  a & b & c & d \\
  e & f & g & h \\
  i & j & k & l \\
  0 & 0 & 0 & 1
\end{bmatrix}
\]

Simple Compositions

translate(a,b,c) translate(d,e,f)

scale(a,b,c) scale(d,e,f)

Composing Transformations

suppose we want

\[ P_w = \text{Trans}(2,3,0) \cdot \text{Rot}(z,-90) \cdot P_a \]

- R-to-L: interpret operations with fixed coords
- L-to-R: interpret operations with local coords
- OpenGL (L-to-R, local coords)

\[
\begin{align*}
\text{glTranslatef}(2,3,0); \\
\text{glRotatef}(-90,0,0,1); \\
\end{align*}
\]

updates current transformation matrix by postmultiplying
Composing Transformations

Equivalence

\[ P_p = \text{Trans}(2,3,0) \text{Rot}(z,-90) P_p \]
\[ P_p = \text{Rot}(z,-90) \text{Trans}(-3,2,0) P_p \]

Undoing Transformations

\[ \text{Trans}(x,y,z)^{-1} = \text{Trans}(-x,-y,-z) \]
\[ \text{Trans}(x,y,z) \text{Trans}(-x,-y,-z) = I \]
\[ \text{Rot}(z,\theta)^{-1} = \text{Rot}(z,-\theta) \]
\[ \text{Rot}(z,\theta) \text{Rot}(z,-\theta) = I \]

Test yourself ...

Rotation about a point

\[ \text{Rot}(x,y,z,\theta) \]

\[ \text{Trans}(x,y,z) \text{Rot}(z,\theta) \text{Trans}(-x,-y,-z) \]

Rotation about an arbitrary axis

- axis defined by two points
- translate point to the origin
- rotate to align axis with z-axis (or x or y)
- perform rotation
- undo aligning rotations
- undo translation

\[ \text{glRotatef}(\text{angle}, x, y, z); \]

Transformation Hierarchies

scene graph
Transformation Hierarchies

Matrix Stack

Hierarchical Modeling with Matrix Stacks

Example

Projective Rendering Pipeline
Coming Up…

- animation
- projection transformations
- viewing transformations