What We Learned

- Matrix-based computations are crucial to the geometry of computer graphics.
- Given a linear transform T, we can derive a 2x2 matrix for T in terms of its action on the vectors $(1,0)$ and $(0,1)$

$$T(1,0) = (a,c) \quad T(0,1) = (b,d) \longrightarrow T = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

- Not all common transforms can be represented with just a change-of-basis (a 2x2 matrix). So we boost the problem into an affine space, and use homogeneous coordinates.

$$\text{point at } (x, y) \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad \text{vector } (u, v) \rightarrow \begin{bmatrix} u \\ v \\ 0 \end{bmatrix}$$

$$\text{2x2 matrix } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \longrightarrow \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Note: as above, columns of the matrix tells us what the transform does to the basis vectors and the origin (in homogeneous coordinates).

The Matrices

Identity (do nothing): $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Scale by s_x in the x direction and s_y in the y direction ($s_x < 0$ or $s_y < 0$ is reflection): $\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Rotate by angle θ (in radians): $\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Shear by amount a in the x direction: $\begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Shear by amount b in the y direction: $\begin{bmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Translate by the vector (t_x, t_y): $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$

Properties of Transforms

- Compact representation
- Fast implementation
- Easy to invert
- Easy to compose

Examples

- Rotation about a point other than the origin
- Reflection through a line other than an axis
- Viewport transform

Summary

- A small number of 2D transforms are especially useful in computer graphics: translation, rotation, scale, shear, reflection, and of course identity.
- Some of these transforms are linear and can be represented using a 2x2 matrix.
- Some are affine and can be represented using a 3x3 matrix acting on homogeneous coordinates.
- Affine transforms have certain properties which make them very practical for computer graphics.

3D Transformations

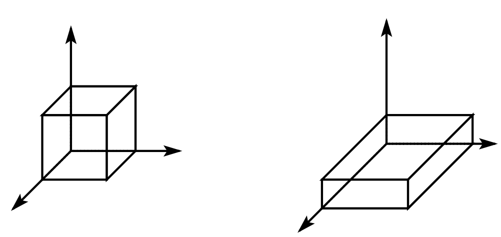
Introduction

- A large part of graphics is the manipulation of 3D geometry.
  - And the computation of a 2D representation
- We need 3D transformations analogous to the 2D transformations we just developed.
  - We also need projections for converting to 2D. We'll discuss these later.

3D Scaling

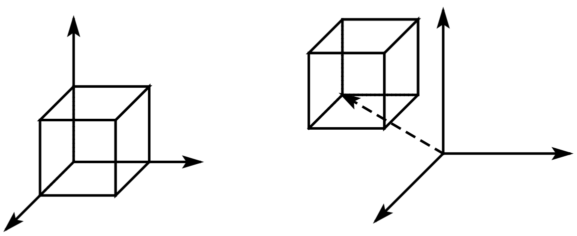
- Some of the 3D transformations look just like their 2D counterparts. Scaling is such a case:

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



3D Translation

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



3D Rotation

Rotate about the x axis: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Rotate about the y axis: $\begin{pmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Rotate about the z axis: $\begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

How can we rotate about an arbitrary line?

3D Shear

- Shear in 3D is also more complicated. Here's one example:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & a & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$