CSE328 Fundamentals of Computer Graphics (Theory, Algorithms, and Applications)

Hong Qin
Department of Computer Science
Stony Brook University (SUNY at Stony Brook)
Stony Brook, New York 11794-2424
Tel: (631)632-8450; Fax: (631)632-8334
qin@cs.stonybrook.edu
http://www.cs.stonybrook.edu/~qin

2D-3D Transformations

- From local, model coordinates to global, world coordinates

Modeling Transformations

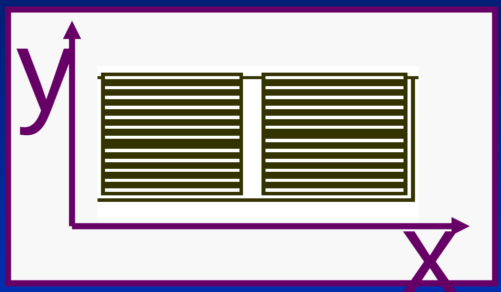
- 2D-3D transformations
- Specify transformations for objects
  - Allows definitions of objects in their own coordinate systems
  - Allows use of object definition multiple times in a scene
  - Please pay attention to how OpenGL provides a transformation stack because they are so frequently reused

Overview

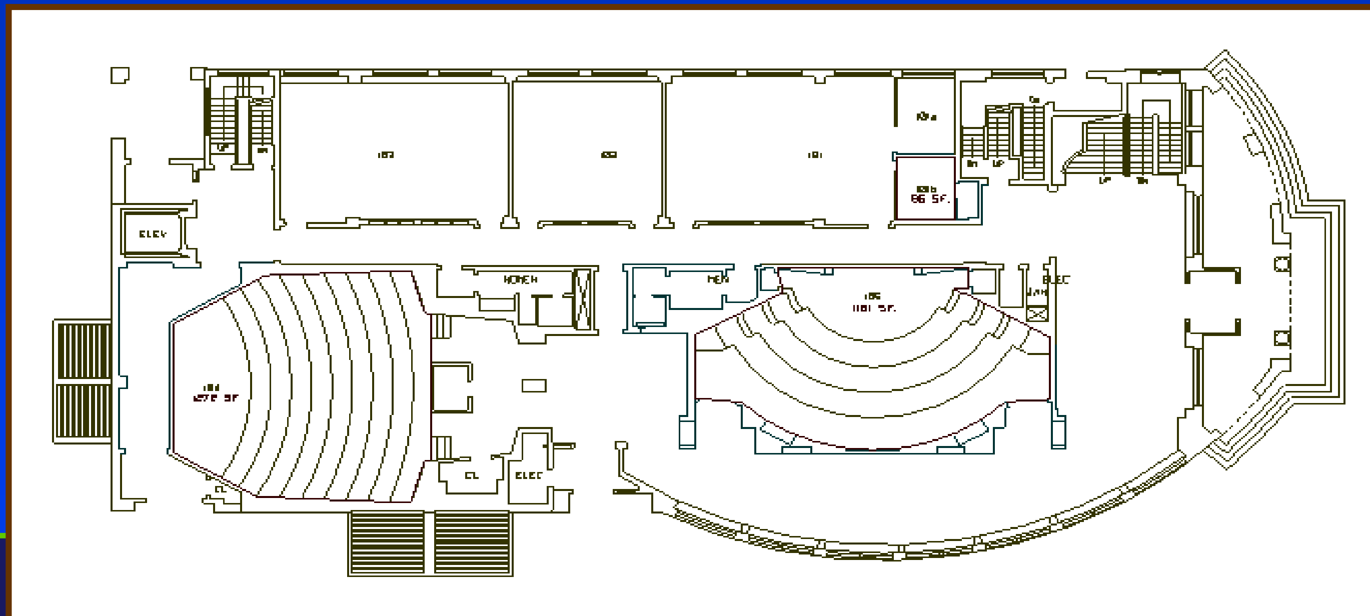
- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- Generalization to 3D Transformations
  - Basic 3D transformations
  - Same as 2D

From Model Coordinates to World Coordinates (Local to Global)

Model coordinates (local)



World coordinates (global)



Basic 2D Transformations

- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Shear:
  - $x' = x + h_x * y$
  - $y' = y + h_y * x$
- Rotation:
  - $x' = x * \cos\Theta - y * \sin\Theta$
  - $y' = x * \sin\Theta + y * \cos\Theta$

Scaling

- Scaling a coordinate means multiplying each of its components by a scalar
- Uniform scaling means this scalar is the same for all components:
- Non-uniform scaling: different scalars per component:
- How can we represent scaling in matrix form?

STONY BROOK
STATE UNIVERSITY OF NEW YORK

Scaling Operation in Matrix Form

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ax \\ by \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling

- Scaling operation: $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ax \\ by \end{bmatrix}$$

- Or, in matrix form: $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\text{scaling matrix}} \begin{bmatrix} x \\ y \end{bmatrix}$$

scaling matrix

Rotation

2-D Rotation

$$x' = x\ \mathbf{cos}(\theta) - y\ \mathbf{sin}(\theta)$$

$$y' = x\ \mathbf{sin}(\theta) + y\ \mathbf{cos}(\theta)$$

2D Rotation Derivation

- $x = r \cos(\phi)$
- $y = r \sin(\phi)$
- $x' = r \cos(\phi + \theta)$
- $y' = r \sin(\phi + \theta)$

- $x' = r \cos(\phi) \cos(\theta) - r \sin(\phi) \sin(\theta)$
- $y' = r \sin(\phi) \sin(\theta) + r \cos(\phi) \cos(\theta)$

- $x' = x \, \mathbf{cos}(\theta) - y \, \mathbf{sin}(\theta)$
- $y' = x \, \mathbf{sin}(\theta) + y \, \mathbf{cos}(\theta)$

2-D Rotation

- It is straightforward to see this procedure in matrix form:

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

Even though sin(θ) and cos(θ) are nonlinear functions of θ,

- ***x' is a linear combination of x and y***
- ***y' is a linear combination of x and y***

2D Rotation

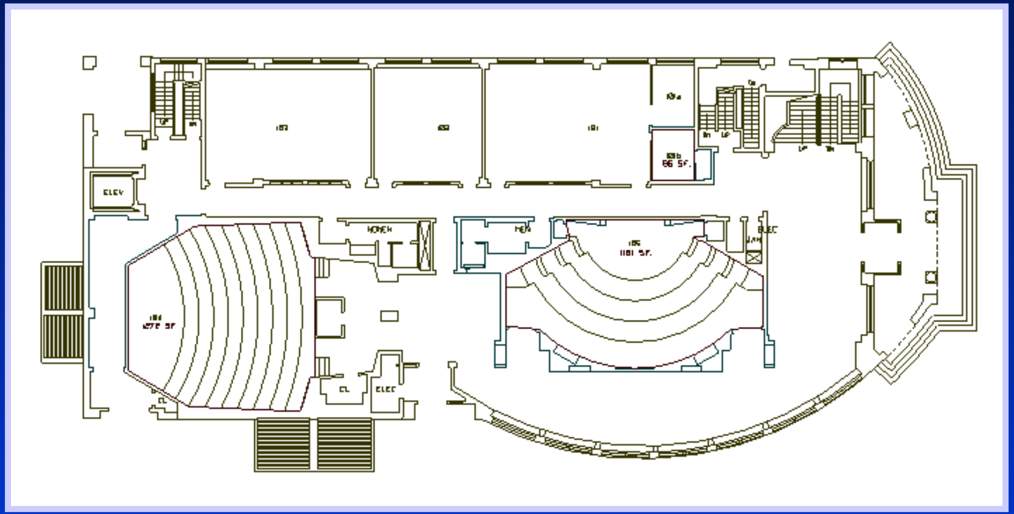
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Basic 2D Transformations

- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Shear:
  - $x' = x + h_x*y$
  - $y' = y + h_y*x$
- Rotation:
  - $x' = x*\cos\Theta - y*\sin\Theta$
  - $y' = x*\sin\Theta + y*\cos\Theta$

Basic 2D Transformations

- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Shear:
  - $x' = x + h_x * y$
  - $y' = y + h_y * x$
- Rotation:
  - $x' = x * \cos\Theta - y * \sin\Theta$
  - $y' = x * \sin\Theta + y * \cos\Theta$



Transformations can be combined (with simple algebra)

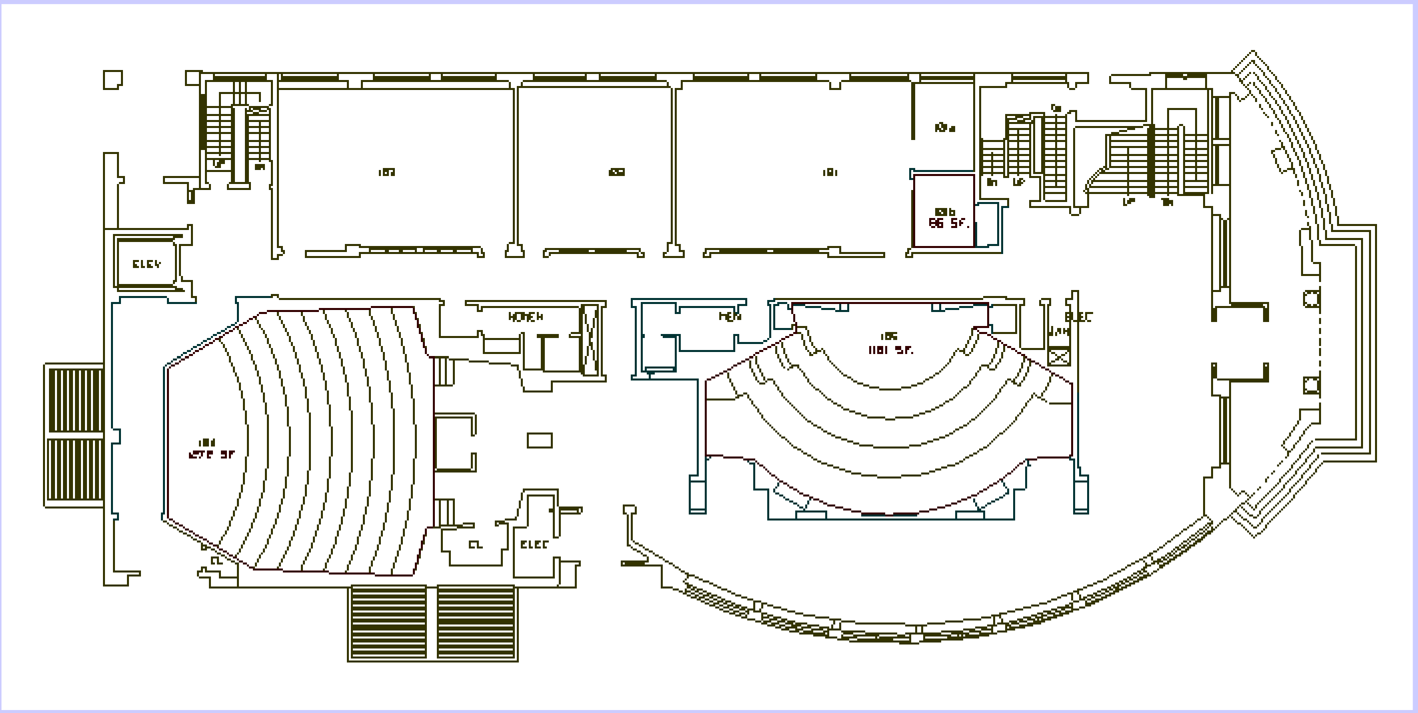
Combining Transformations

- Transformations can be combined (with simple algebra)

Composite Transformations

Transformations can be combined

(with simple algebra)



Matrix Representation

- Represent 2D transformation by a matrix

- Multiply matrix by column vector

  $\Leftrightarrow$ apply transformation to point

Overview

- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- Generalization to 3D Transformations
  - Basic 3D transformations
  - Same as 2D

Matrix Representation

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$x' = ax + by$$
$$y' = cx + dy$$

Matrix Representation

- Represent 2D transformation by a matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

- Multiply matrix by column vector
  ⇔ apply transformation to point

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$x' = ax + by$$
$$y' = cx + dy$$

Matrix Representation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrix Representation

- Transformations can be combined by multiplication
  - Matrices are a convenient and efficient way to represent a sequence of transformations!

STONY BROOK
STATE UNIVERSITY OF NEW YORK

Matrix Representation

- Transformations combined by multiplication

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrices are a convenient and efficient way

to represent a sequence of transformations!

2x2 Matrices

- What types of transformations can be represented with a 2x2 matrix?

2D Identity?

$$x' = x$$
$$y' = y$$

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

2D Scale around (0,0)?

$$\mathbf{x}' = \mathbf{s}_x * \mathbf{x}$$
$$\mathbf{y}' = \mathbf{s}_y * \mathbf{y}$$

$$\begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \end{bmatrix} = \begin{bmatrix} \mathbf{s}_x & 0 \\ 0 & \mathbf{s}_y \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

2x2 Matrices

- What types of transformations can be represented with a 2x2 matrix?

2D Rotate around (0,0)?

$$x' = \cos\Theta * x - \sin\Theta * y$$
$$y' = \sin\Theta * x + \cos\Theta * y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D Shear?

$$x' = x + sh_x * y$$
$$y' = sh_y * x + y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & sh_x \\ sh_y & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2x2 Matrices

- What types of transformations can be represented with a 2x2 matrix?

2D Mirror about Y axis?

$$x' = -x \\ y' = y$$

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

2D Mirror over (0,0)?

$$x' = -x \\ y' = -y$$

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

2x2 Matrices

- What types of transformations can be represented with a 2x2 matrix?

2D Translation?

$$x' = x + t_x$$

$$y' = y + t_y$$

NO!

Only linear 2D transformations

can be represented with a 2x2 matrix

Linear Transformations

- Linear transformations are combinations of …
  - Scale,
  - Rotation,
  - Shear, and
  - Mirror

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- Properties of linear transformations:
  - Satisfies: $T(s_1\mathbf{p}_1 + s_2\mathbf{p}_2) = s_1T(\mathbf{p}_1) + s_2T(\mathbf{p}_2)$
  - Origin maps to origin
  - Lines map to lines
  - Parallel lines remain parallel
  - Ratios are preserved
  - Closed under composition

Homogeneous Coordinates

- Q: How can we represent translation as a 3x3 matrix?

$$x' = x + t_x$$

$$y' = y + t_y$$

STONY BROOK
STATE UNIVERSITY OF NEW YORK

Homogeneous Coordinates

- Homogeneous coordinates
  - represent coordinates in 2 dimensions with a 3-vector

$$\begin{bmatrix} x \\ y \end{bmatrix} \xrightarrow{\text{homogeneous coords}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Homogeneous coordinates appear to be far less intuitive, but they indeed make graphics operations much easier

Homogeneous Coordinates

- Q: How can we represent translation as a 3x3 matrix?

$$x' = x + t_x$$
$$y' = y + t_y$$

- A: Using the rightmost column:

$$Translation = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Translation

- Example of translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix}$$

Homogeneous Coordinates

- Add a 3rd coordinate to every 2D point
  - (x, y, w) represents a point at location (x/w, y/w)
  - (x, y, 0) represents a point at infinity
  - (0, 0, 0) is not allowed
- Note that, (6,3,1); (12,6,2); and (18,9,3) represent the SAME POINT in 2D

Convenient coordinate system to represent many useful transformations

Basic 2D Transformations

- Basic 2D transformations as 3x3 matrices

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scale

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

Rotate

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & sh_x & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Shear

Affine Transformations

- Affine transformations are combinations of …
  - Linear transformations, and
  - Translations

$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

- Properties of affine transformations:
  - Origin does not necessarily map to origin
  - Lines map to lines
  - Parallel lines remain parallel
  - Ratios are preserved
  - Closed under composition

Projective Transformations

- Projective transformations …
  - Affine transformations, and
  - Projective warps

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

- Properties of projective transformations:
  - Origin does not necessarily map to origin
  - Lines map to lines
  - Parallel lines do not necessarily remain parallel
  - Ratios are not preserved
  - Closed under composition

Overview

- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- Generalization to 3D Transformations
  - Basic 3D transformations
  - Same as 2D

Matrix Composition

- Transformations can be combined by matrix multiplication

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \left(\begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} sx & 0 & 0 \\ 0 & sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

$$\mathbf{p}' = T(t_x,t_y) * R(\Theta) * S(s_x,s_y) * \mathbf{p}$$

Matrix Composition

- Matrices are a convenient and efficient way to represent a sequence of transformations
  - General purpose representation
  - Hardware matrix multiply

$$\mathbf{p}' = (T * (R * (S*\mathbf{p})\ )\ )$$

$$\mathbf{p}' = (T*R*S) * \mathbf{p}$$

Matrix Composition

- From local coordinates to global coordinates
- Be aware: order of transformations matters
    - Matrix multiplication is not commutative

$$\mathbf{p}' = T * R * S * \mathbf{p}$$

Matrix Composition

- A more complicated example: rotating 90 degrees around the mid-point of a line segment (whose coordinates are (3,2))
- Can we change the order between rotation and translation?

Matrix Composition

Will this sequence of operations work?

$$\begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(90) & -\sin(90) & 0 \\ \sin(90) & \cos(90) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ 1 \end{bmatrix} = \begin{bmatrix} a'_x \\ a'_y \\ 1 \end{bmatrix}$$

Matrix Composition

- After correctly ordering the matrices
- Multiply matrices together
- What results is one matrix – store it (on stack)!
- Multiply this matrix by the vector of each vertex
- All vertices easily transformed with one matrix multiply

Overview

- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- Generalization to 3D Transformations
  - Basic 3D transformations
  - Same as 2D

3D Transformations

- Same idea as 2D transformations
  - Homogeneous coordinates: (x,y,z,w)
  - 4x4 transformation matrices

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Basic 3D Transformations

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

Identity

$$\begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \\ \mathbf{z}' \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{s}_x & 0 & 0 & 0 \\ 0 & \mathbf{s}_y & 0 & 0 \\ 0 & 0 & \mathbf{s}_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \\ \mathbf{w} \end{bmatrix}$$

Scale

$$\begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \\ \mathbf{z}' \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \mathbf{t}_x \\ 0 & 1 & 0 & \mathbf{t}_y \\ 0 & 0 & 1 & \mathbf{t}_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \\ \mathbf{w} \end{bmatrix}$$

Translation

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Mirror about Y/Z plane

Basic 3D Transformations

Rotate around Z axis:

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

Rotate around Y axis:

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ \boldsymbol{z}' \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \cos\Theta & 0 & \sin\Theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\Theta & 0 & \cos\Theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \boldsymbol{z} \\ \boldsymbol{w} \end{bmatrix}$$

Rotate around X axis:

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

Matrix Composition

- Matrices are a convenient and efficient way to represent a sequence of transformations
  - General purpose representation
  - Hardware matrix multiply

$$\mathbf{p}' = (T * (R * (S*\mathbf{p})\ )\ )$$

$$\mathbf{p}' = (T*R*S) * \mathbf{p}$$

Matrix Composition

- From local coordinates to global coordinates
- Be aware: order of transformations matters
  - Matrix multiplication is not commutative

$$\mathbf{p}' = T * R * S * \mathbf{p}$$

Reverse Rotations

- Q: How do you undo a rotation of θ, R(θ)?
- A: Apply the inverse of the rotation… $R^{-1}(\theta) = R(-\theta)$
- How to construct R-1(θ) = R(-θ)
  - Inside the rotation matrix: cos(θ) = cos(-θ)
    - The cosine elements of the inverse rotation matrix are unchanged
  - The sign of the sine elements will flip
  - This is because the rotation matrix is orthogonal matrix
- Therefore… $R^{-1}(\theta) = R(-\theta) = R^{T}(\theta)$

Summary

- Coordinate systems are the basis for computer graphics
  - World vs. model coordinates (Global vs. Local)
- 2D and 3D transformations
  - Trigonometry and geometry
  - Matrix representations
  - Linear, affine, and projective transformations
- Matrix operations
  - Matrix composition