VECTORS, MATRICES AND COORDINATE SYSTEMS

2011 Introduction to Graphics

Lecture 5

Overview

- Vector math
- Transformations of points
  - Rotations
  - Scales
  - Translations
- Homogenous coordinates
- Coordinate systems
  - Screen, User and Mappings

Vectors

- 2D vector (x y)
- 3D vector (x y z)
- Vector addition
  - (x y) + (p q) = (x+p y+q)
- Vector scale
  - s*(x y) = (s*x s*y)

Matrices

- Matrix is an array of numbers with dimensions M (rows) by N (columns)
  - 3 by 6 matrix
  - Element 2,3 is (3)

$$\begin{pmatrix} 3 & 0 & 0 & -2 & 1 & -2 \\ 1 & 1 & 3 & 4 & 1 & -1 \\ -5 & 2 & 0 & 0 & 0 & 1 \end{pmatrix}$$

- Vector can be considered a 1 x M matrix
  - $v = \begin{pmatrix} x & y & z \end{pmatrix}$

Operation on Matrices

- Addition

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} + \begin{pmatrix} p & q \\ r & s \end{pmatrix} = \begin{pmatrix} a+p & b+q \\ c+r & d+s \end{pmatrix}$$

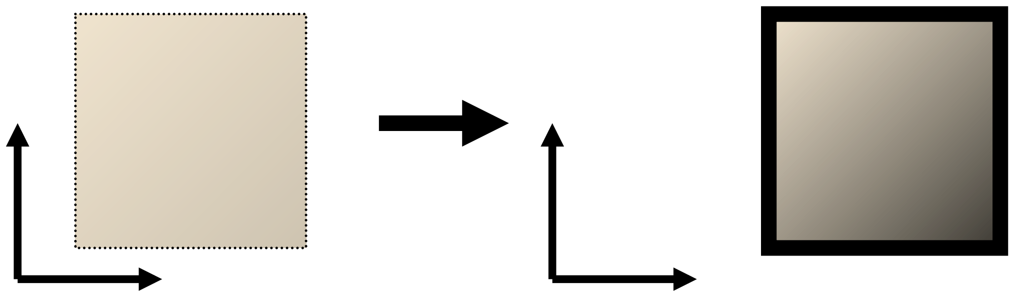
- Multiplication

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} * \begin{pmatrix} p & q \\ r & s \end{pmatrix} = \begin{pmatrix} ap+br & aq+bs \\ cp+dr & cq+ds \end{pmatrix}$$

General Transformations

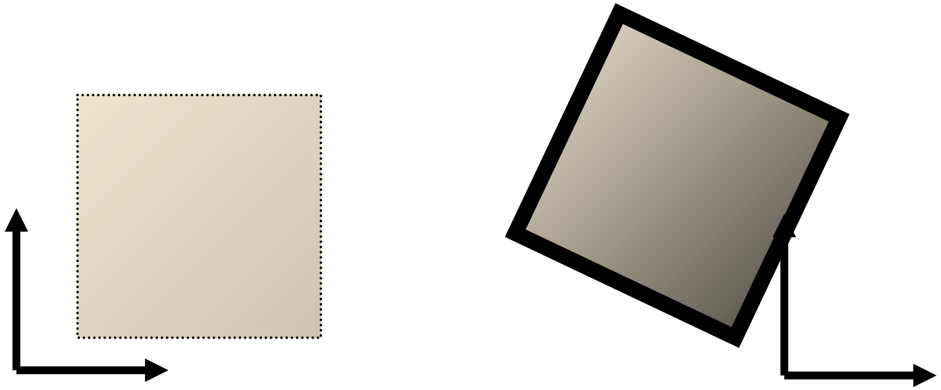
- Affine transformations
  - Translate
  - Rotate
  - Scale
  - Shear

Translate



$$\begin{pmatrix} x' & y' \end{pmatrix} = \begin{pmatrix} x & y \end{pmatrix} + \begin{pmatrix} p & q \end{pmatrix}$$

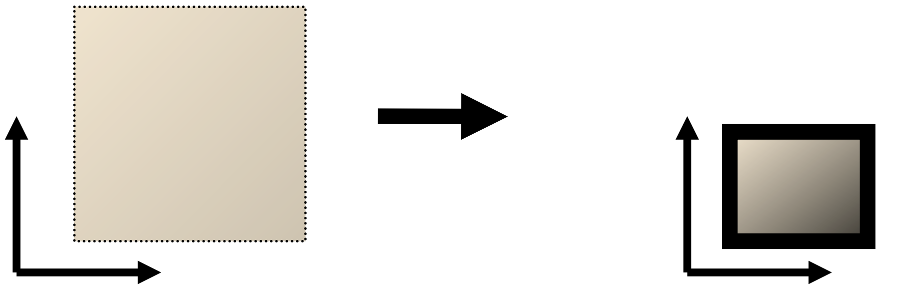
Rotation



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

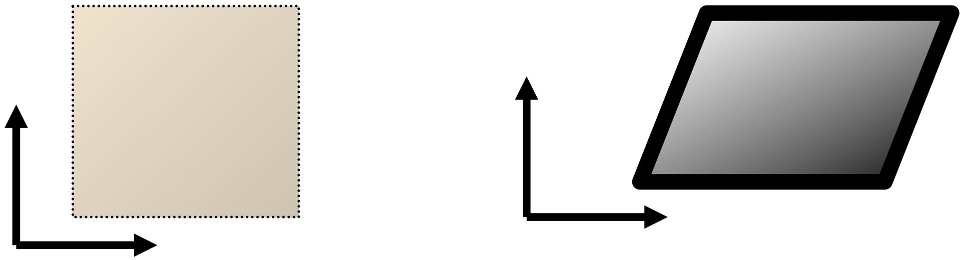
Positive rotation is anti-clockwise

Scale



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

Shear



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

Combining Transformations

- Transform M followed by transform N can be represented by M*N
- E.g., rotate PI/2 followed by scale 0.5

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} * \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix} = \begin{pmatrix} 0 & 0.5 \\ -0.5 & 0 \end{pmatrix}$$

Homogenous Coordinates (1)

- A mechanism to allow us to combine translations into the matrix multiplication, so as to simplify long calculations involving several rotations, translations, etc…
- 2D points are now represented by three numbers, where the 3rd is always 1
  - (x,y) becomes (x,y,1)

Homogenous Coordinates (2)

- Our rotation, scale and shear matrices now have to become 3x3 rather than 2x2

$$(x \quad y)\begin{pmatrix} R1 & R2 \\ R3 & R4 \end{pmatrix} \qquad (x \quad y \quad 1)\begin{pmatrix} R1 & R2 & 0 \\ R3 & R4 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Before **After**

Homogenous Coordinates (3)

- The advantage is now we can include translation within the 3x3 matrix

$$\begin{pmatrix} x & y & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ r & s & 1 \end{pmatrix} = \begin{pmatrix} x+r & y+s & 1 \end{pmatrix}$$

Homogenous Coordinates (4)

- Rotation by 45 degrees followed by a translation of -1,1 can be represented as:

$$\begin{pmatrix} 0.707 & 0.707 & 0 \\ -0.707 & 0.707 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0.707 & 0.707 & 0 \\ -0.707 & 0.707 & 0 \\ -1 & 1 & 1 \end{pmatrix}$$

Order Matters!

- Rotate 90 degrees, then translation 2,3

$$\begin{pmatrix} x & y & 1 \end{pmatrix}\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 3 & 1 \end{pmatrix} = \begin{pmatrix} x & y & 1 \end{pmatrix}\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 2 & 3 & 1 \end{pmatrix} = \begin{pmatrix} 2-y & x+3 & 1 \end{pmatrix}$$

- Translation 2,3 then rotate 90 degrees

$$\begin{pmatrix} x & y & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 3 & 1 \end{pmatrix}\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} x & y & 1 \end{pmatrix}\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ -3 & 2 & 1 \end{pmatrix} = \begin{pmatrix} -3-y & x+2 & 1 \end{pmatrix}$$

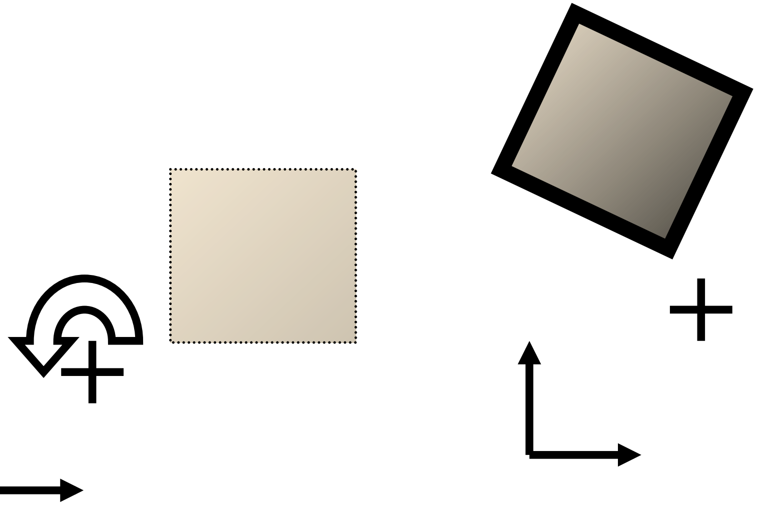
Inverting Transformations

- Inverse of “rotate by x” is “rotate by -x”

$$\begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}^{-1} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

- Inverse of “scale s” is “scale 1/s”

Rotation About Arbitrary Point



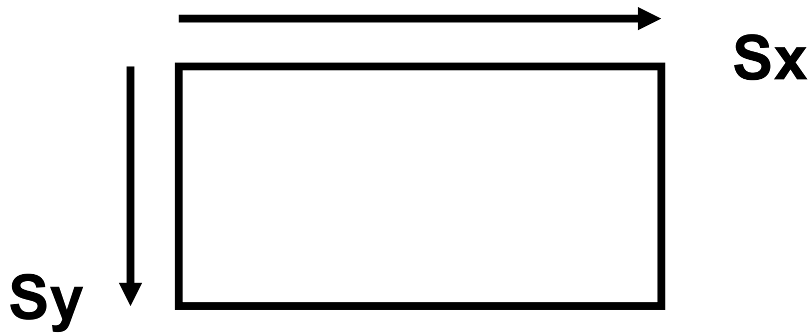
- Outline
  - translate so that rotation point is at origin
  - rotate
  - invert translation

Transformations in Java

- *AffineTransform()*
  - Class to store, concatenate and apply transformations
    - *trans.translate(x,y)*: translate by x,y (auto-concat.)
    - *trans.rotate(angle)*: rotate by angle (auto-concat.)
    - *trans.concatenate(oth_trans)*: multiply two transformations together
    - *transshape = trans.createTransformedShape(shape)*: create transformed shape
    - *trans.setToTranslation(x,y)*: set transform to translate by x,y
    - *trans.setToRotation(angle)*: set transform to rotate by angle

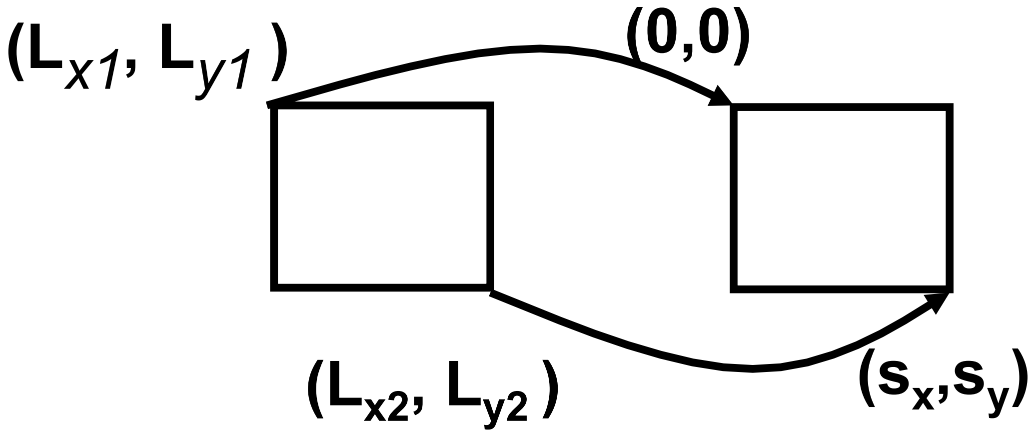
Logical and Screen Coordinates

- Screen co-ordinates specified by width, height of screen starting at top left



- We often want to specify logical co-ordinates that do not depend on window scale

Mapping from Logical to Screen



- Set up a mapping as a scale and translation scale
  - translate by ($-L_{x1}$, $-L_{y1}$)
  - scale $\begin{pmatrix} \frac{s_x}{L_{dx}} & 0 & 0 \\ 0 & \frac{s_y}{L_{dy}} & 0 \\ 0 & 0 & 1 \end{pmatrix}$ where $L_{dx}=L_{x2}- L_{x1}$

Summary

- Transformations of points
  - Rotations
  - Scales
  - Translations
- Homogenous coordinates
- Relationship between screen and logical coordinates
- Plenty of example exercises in the handout!