Affine Transformations

CSE 457

Reading

Required:

- Angel 3.1, 3.7-3.11

Further reading:

- Angel, the rest of Chapter 3
- Foley, et al, Chapter 5.1-5.5.
- David F. Rogers and J. Alan Adams, *Mathematical Elements for Computer Graphics*, 2nd Ed., McGraw-Hill, New York, 1990, Chapter 2.

Geometric transformations

Geometric transformations will map points in one space to points in another: $(x', y', z') = \boldsymbol{f}(x, y, z)$.

These transformations can be very simple, such as scaling each coordinate, or complex, such as non-linear twists and bends.

We'll focus on transformations that can be represented easily with matrix operations.

Vector representation

We can represent a **point**, **p** = (x,y), in the plane or **p**=(x,y,z) in 3D space

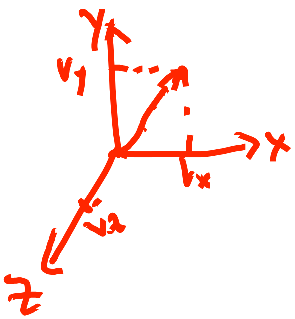
- as column vectors $p = \begin{bmatrix} x \\ y \end{bmatrix}$ $p = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$

  2D 3D

- as row vectors $p = \begin{bmatrix} x & y \end{bmatrix}$ 2D

  $p = \begin{bmatrix} x & y & z \end{bmatrix}$ 3D

Canonical axes



right hand rule

$$V = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

$$\hat{x}, \hat{y}, \hat{z} \qquad \hat{x} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \hat{y} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \hat{z} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

$$V = v_x \cdot \hat{x} + v_y \cdot \hat{y} + v_z \cdot \hat{z} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

Vector length and dot products

$v = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} \qquad u = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}$

length $\quad \|v\| = \sqrt{v_x^2 + v_y^2 + v_z^2}$

$\hat{v}$ unit vector $\quad \|\hat{v}\| = 1$

$\dfrac{v}{\|v\|} = \hat{v}$

transpose $\quad v^T = (v_x \;\; v_y \;\; v_z)$

dot product

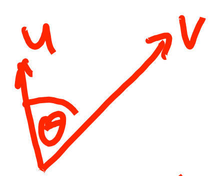
$u \cdot v = u_x v_x + u_y v_y + u_z v_z$

$= u^T \cdot v$ multiplication

$= (u_x \;\; u_y \;\; u_z) \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$

$u \cdot u = \|u\|^2$

geometric interp.



$u \cdot v = \|u\| \, \|v\| \cdot \cos\theta$

$\theta = 0^\circ \Rightarrow$ parallel $\quad \hat{u} \cdot \hat{v} = 1$

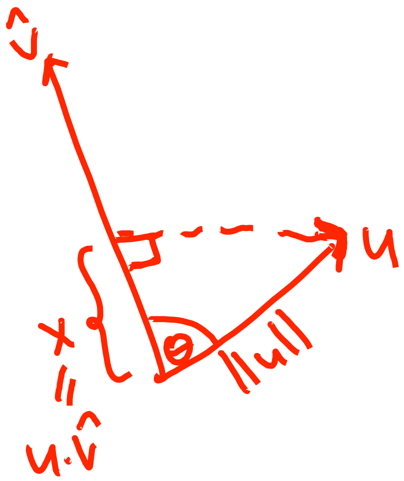
$\theta = 90^\circ \Rightarrow$ orthogonal

$u \cdot v = 0$

$u \cdot v = v \cdot u \quad$ yes true

$\theta = 180^\circ \quad \hat{u} \cdot \hat{v} = -1$

projection of u onto $\hat{v}$



$$\cos\theta = \frac{x}{\|u\|} \Rightarrow x = \|u\| \cos\theta$$

$$u \cdot v = \|u\| \, \|v\| \cos\theta = \|v\| \cdot x$$

$$x = \frac{u \cdot v}{\|v\|} = u \cdot \hat{v}$$

$$(u \cdot \hat{v}) \, \hat{v}$$

Vector cross products

$$u \times v = \begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix} = \hat{x}\begin{vmatrix} u_y & u_z \\ v_y & v_z \end{vmatrix} - \hat{y}\begin{vmatrix} u_x & u_z \\ v_x & v_z \end{vmatrix} + \hat{z}\begin{vmatrix} u_x & u_y \\ v_x & v_y \end{vmatrix}$$

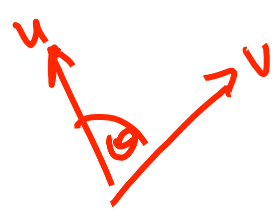
$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc \implies = \begin{pmatrix} u_y v_z - u_z v_y \\ -u_x v_z + u_z v_x \\ u_x v_y - u_y v_x \end{pmatrix}$$

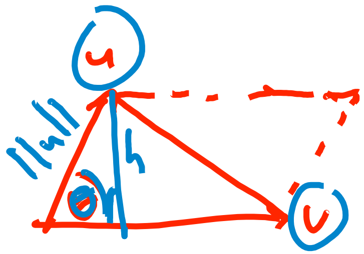
$$(u \times v) \cdot u = 0$$

$$(u \times v) \cdot v = 0$$

$$u \times v = -v \times u$$

$$\|u \times v\| = \|u\| \, \|v\| \sin\theta$$





$$A_{\text{parallelogram}} = \|u \times v\|$$

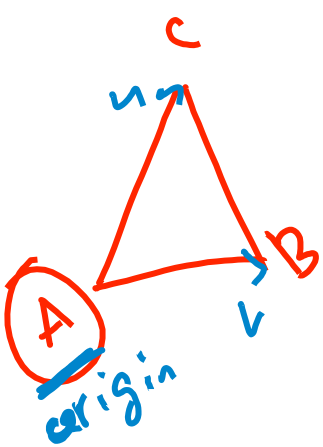
$$A_{\Delta} = \frac{\|u \times v\|}{2}$$

v - base

$$A_{\text{parallelogram}} = h \cdot \|v\|$$

$$h = \|u\| \sin\theta$$

$$A_{\text{parallelogram}} = \|u\| \, \|v\| \sin\theta$$
$$= \|u \times v\|$$



$$\frac{\|(C-A) \times (C-B)\|}{2}$$

Representation, cont.

We can represent a **2-D transformation** *M* by a matrix

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

2×2

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

3×3

If **p** is a column vector, *M* goes on the left:

$$\mathbf{p}' = M\mathbf{p}$$

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

If **p** is a row vector, M^T goes on the right:

$$\mathbf{p}' = \mathbf{p}M^T$$

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

We will use **column vectors**.

Two-dimensional transformations

Here's all you get with a 2 x 2 transformation matrix *M*:

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

So:

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

We will develop some intimacy with the elements *a, b, c, d*…

Identity

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Suppose we choose *a=d=1, b=c=0:*

- Gives the **identity** matrix:

$$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- Doesn't move the points at all

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

$$\boxed{\begin{matrix} x' = x \\ y' = y \end{matrix}}$$

Scaling

$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Suppose we set *b=c=0*, but let *a* and *d* take on any *positive* value:

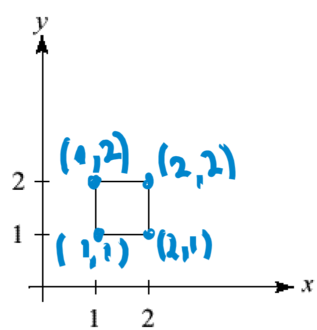
- Gives a **scaling** matrix:

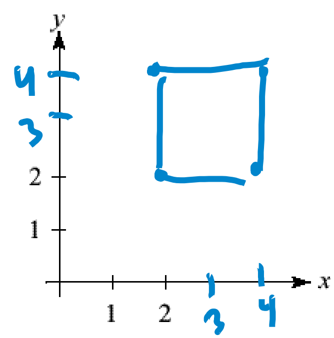
$$M = \begin{bmatrix} a & 0 \\ 0 & d \end{bmatrix}$$

- Provides **differential (non-uniform) scaling** in *x* and *y*:

$$\boxed{\begin{aligned} x' &= ax \\ y' &= dy \end{aligned}}$$

2D





M uniform scaling = 2

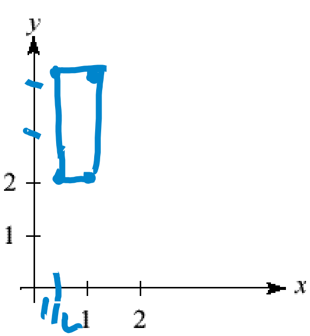
$$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$

$\begin{pmatrix} 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \end{pmatrix}$

$\begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$

$\begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$



M

$$\begin{bmatrix} 1/2 & 0 \\ 0 & 2 \end{bmatrix}$$

$\begin{pmatrix} 1 \\ 2 \end{pmatrix}$

$\begin{pmatrix} 2 \\ 1 \end{pmatrix}$

Reflection/Mirroring

$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Suppose we keep *b=c=0*, but let either *a* or *d* go negative.

$\begin{pmatrix} a & 0 \\ 0 & d \end{pmatrix}$

Examples:

M_1

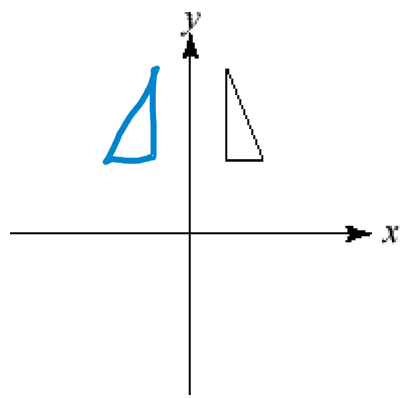
$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

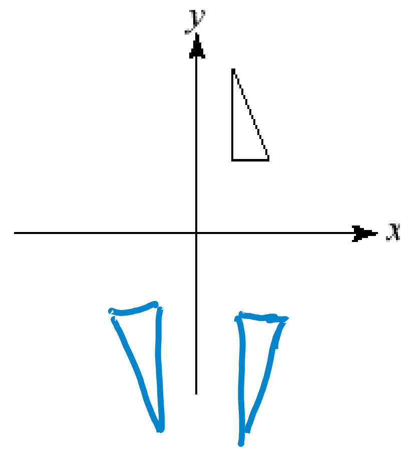
$x' = -x$
$y' = y$

M_2

$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

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





rotation = $M_1 M_2 \begin{pmatrix} x \\ y \end{pmatrix}$

Shearing

$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Now let's leave *a=d=1* and experiment with *b*. . . .

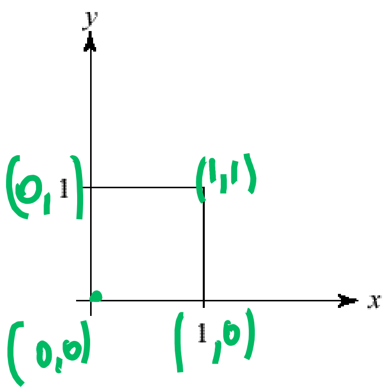
The matrix

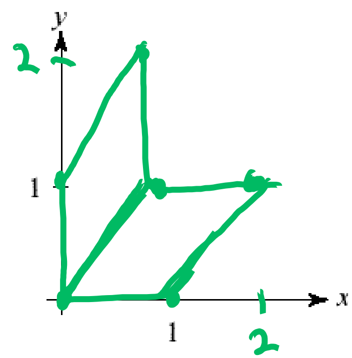
$$M = \begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix}$$

gives:

$$x' = x + by$$
$$y' = y$$

$b = 1$ $\qquad M = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$





M

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

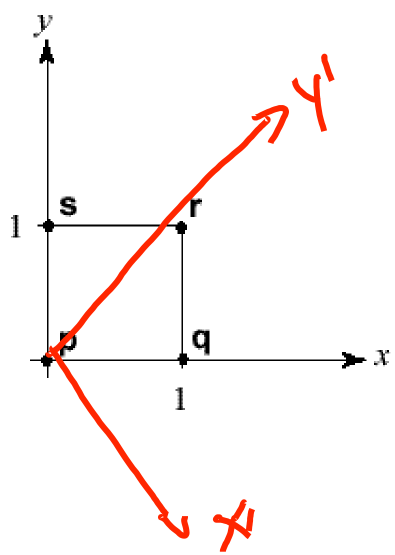
Effect on unit square

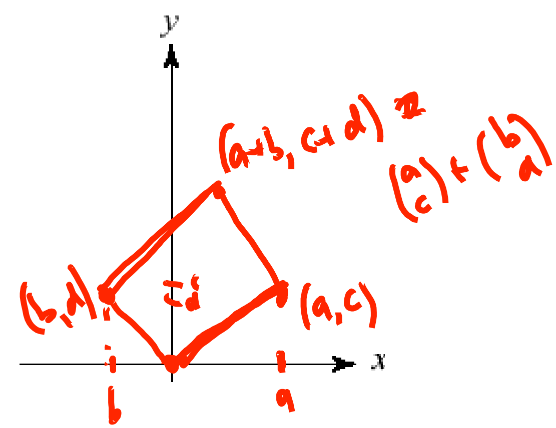
Let's see how a general 2 x 2 transformation *M* affects the unit square:

M points on sq.

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \mathbf{p} & \mathbf{q} & \mathbf{r} & \mathbf{s} \end{bmatrix} = \begin{bmatrix} \mathbf{p'} & \mathbf{q'} & \mathbf{r'} & \mathbf{s'} \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & a & a+b & b \\ 0 & c & c+d & d \end{bmatrix}$$





Effect on unit square, cont.

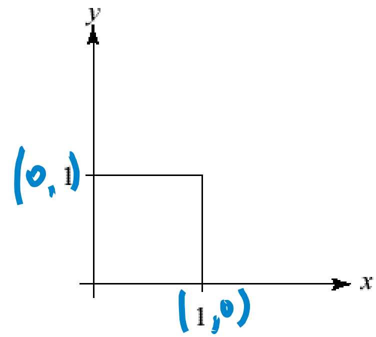
Observe:

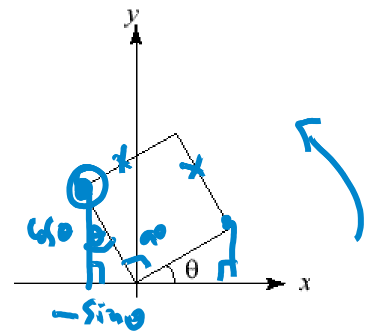
- Origin invariant under *M*
- *M* can be determined just by knowing how the corners (1,0) and (0,1) are mapped
- ~~*a* and *d* give *x*- and *y*-scaling~~
- *b* and *c* give *x*- and *y*-shearing

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Rotation

From our observations of the effect on the unit square, it should be easy to write down a matrix for "rotation about the origin":





$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} \rightarrow \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} \rightarrow \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix}$$

Thus,

$$M = R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

2D rotation by θ

Limitations of the 2 x 2 matrix

A 2 x 2 linear transformation matrix allows

- Scaling
- Rotation
- Reflection
- Shearing

Q: What important operation does that leave out?

Translation

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \underbrace{M}_{2\times 2} \begin{pmatrix} x \\ y \end{pmatrix}$$

Homogeneous coordinates

Idea is to loft the problem up into 3-space, adding a third component to every point:

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$p' = p + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

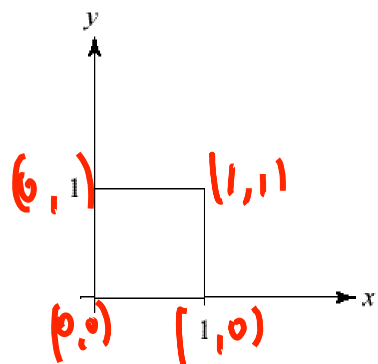
Adding the third "*w*" component puts us in **homogenous coordinates**.

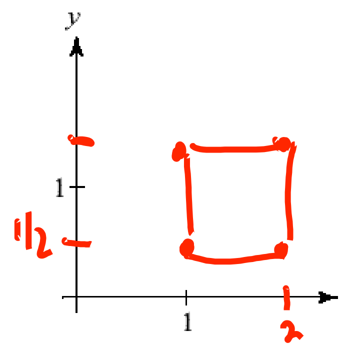
And then transform with a 3 x 3 matrix:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = T(\mathbf{t}) \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{array}{c} \begin{array}{cc} M & \vec{t} \end{array} \\ \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \end{array} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \\ 1 \end{pmatrix}$$

$$M \cdot \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 3/2 \\ 1 \end{pmatrix}$$

$$M \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 3/2 \\ 1 \end{pmatrix}$$





$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1/2 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1/2 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1/2 \\ 1 \end{pmatrix}$$

. . . gives **translation**!

Anatomy of an affine matrix

The addition of translation to linear transformations gives us **affine transformations**.

In matrix form, 2D affine transformations always look like this:

$$M = \begin{bmatrix} a & b & t_x \\ c & d & t_y \\ 0 & 0 & 1 \end{bmatrix} = \left[\begin{array}{cc|c} \multicolumn{2}{c|}{A} & \mathbf{t} \\ \hline 0 & 0 & 1 \end{array}\right]$$

2D affine transformations always have a bottom row of [0 0 1].

An “affine point” is a “linear point” with an added *w*-coordinate which is always 1:

$$\mathbf{p}_{\text{aff}} = \begin{bmatrix} \mathbf{p}_{\text{lin}} \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Applying an affine transformation gives another affine point:

$$M\mathbf{p}_{\text{aff}} = \begin{bmatrix} A\mathbf{p}_{\text{lin}} + \mathbf{t} \\ 1 \end{bmatrix}$$

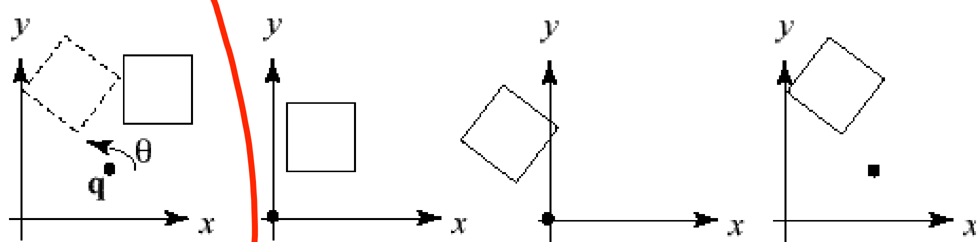
Rotation about arbitrary points

T – translation

R – rotation operation

Until now, we have only considered rotation about the origin.

With homogeneous coordinates, you can specify a rotation, q, about any point $\mathbf{q} = [q_x\ q_y]^T$ with a matrix:



$T(-q)$ $R(\theta)$ $T(q)$

$p' = R(p-t)+t$

$p' = Rp - Rt + t$

$p' = \begin{pmatrix} R & -Rt+t \\ 0\ 0 & 1 \end{pmatrix} p$

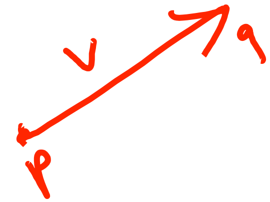
1. Translate **q** to origin
2. Rotate
3. Translate back

$\cancel{\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = T(-q)\,R(\theta)\,T(q)\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}}$

Note: Transformation order is important!!

$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \underbrace{T(q)\,R(\theta)\,T(-q)}_{M}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

$$p = \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix} \qquad q = \begin{pmatrix} x_q \\ y_q \\ 1 \end{pmatrix}$$



$$p - q = \begin{pmatrix} x_p - x_q \\ y_p - y_q \\ 0 \end{pmatrix}$$

vector
3rd coord. is always 0

$v_1 + v_2 \rightarrow$ vector

$\text{scale} \cdot v \rightarrow$ vector

$p_1 - p_2 \rightarrow$ vector

$p_1 + p_2 \rightarrow$?

$$p + q = \begin{pmatrix} x_p + x_q \\ y_p + y_q \\ 2 \end{pmatrix}$$

$$\alpha p + \beta q = \begin{pmatrix} \vdots \\ \alpha + \beta \end{pmatrix}$$

$\alpha + \beta = 1 \Rightarrow$ point

$\alpha + \beta = 0 \Rightarrow$ vector

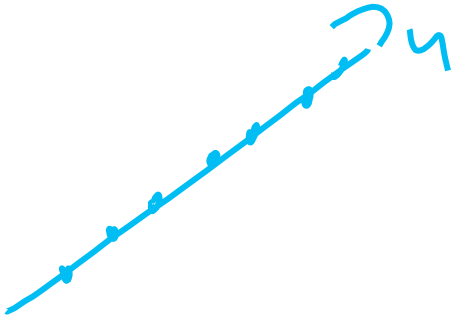
$\alpha p + \beta q + \gamma r = p'$

$\alpha + \beta + \gamma = 1 \Rightarrow$ point

$\sum \alpha_i = 1 \Rightarrow$ point

$p' = p + \alpha \vec{u} \qquad \alpha \in [-\infty, \infty]$

family of point which is a line



$p' = p + \alpha \vec{u} \qquad \alpha \in [0, \infty)$

Ray

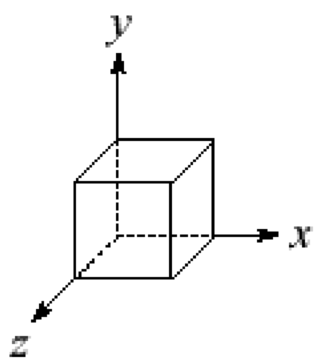
Basic 3-D transformations: scaling

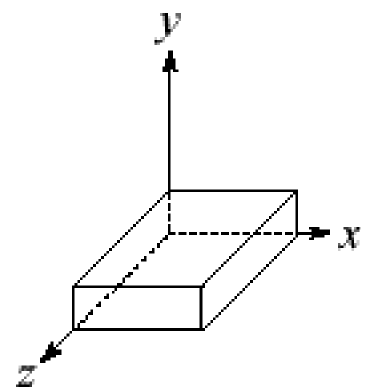
Some of the 3-D transformations are just like the 2-D ones.

For example, <u>scaling</u>:

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

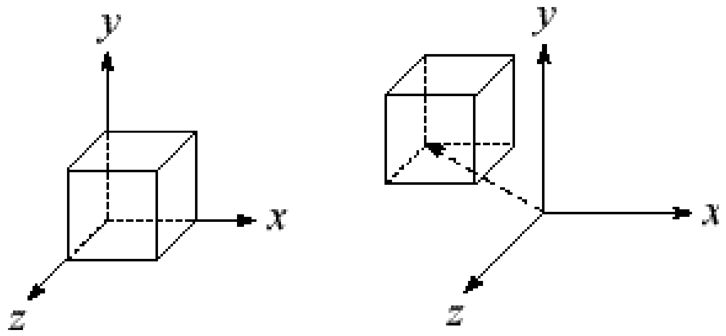
4x4





Translation in 3D

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



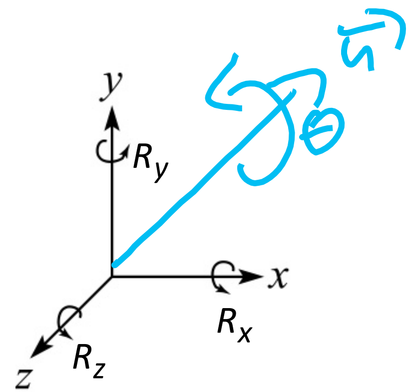
Rotation in 3D (cont'd)

These are the rotations about the canonical axes:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Use right hand rule

$\vec{u}, \theta$

$\sqrt{u_x^2 + u_y^2 + u_z^2} = 1$

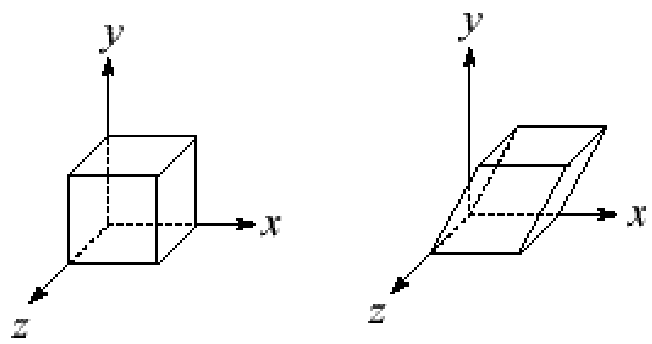
A general rotation can be specified in terms of a product of these three matrices. How else might you specify a rotation?

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = R_z R_y R_x \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Shearing in 3D

Shearing is also more complicated. Here is one example:

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



We call this a shear with respect to the x-z plane.

Properties of affine transformations

Here are some useful properties of affine transformations:

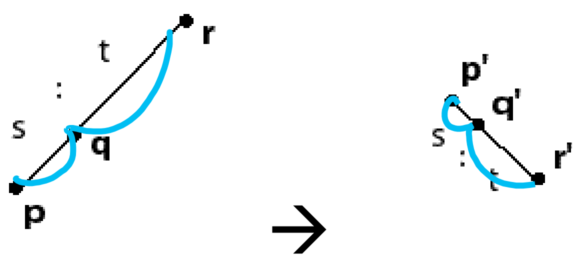
- Lines map to lines
- ✓ Parallel lines remain parallel
- Midpoints map to midpoints (in fact, ratios are always preserved)

$p_1 = p + \alpha \vec{u}$

$p_2 = q + \beta \vec{u}$

$M(p + \alpha \vec{u}) = Mp + \alpha M\vec{u}$

$M(q + \beta \vec{u}) = Mq + \beta M\vec{u}$



$$\text{ratio} = \frac{\|\mathbf{pq}\|}{\|\mathbf{qr}\|} = \frac{s}{t} = \frac{\|\mathbf{p'q'}\|}{\|\mathbf{q'r'}\|}$$

will not get

Affine transformations in OpenGL

OpenGL maintains a "modelview" matrix that holds the current transformation **M.**

The modelview matrix is applied to points (usually vertices of polygons) before drawing.

It is modified by commands including:

- `glLoadIdentity()` **$\mathbf{M} \leftarrow \mathbf{I}$**
  – set **M** to identity
- `glTranslatef(`t_x`, `t_y`, `t_z`)` **$\mathbf{M} \leftarrow \mathbf{MT}$**
  – translate by (t_x, t_y, t_z)
- `glRotatef(`θ`, x, y, z)` **$\mathbf{M} \leftarrow \mathbf{MR}$**
  – rotate by angle θ about axis (x, y, z)
- `glScalef(`s_x`, `s_y`, `s_z`)` **$\mathbf{M} \leftarrow \mathbf{MS}$**
  – scale by (s_x, s_y, s_z)

$MTRS \ldots \begin{pmatrix} x \\ y \end{pmatrix}$

Note that OpenGL adds transformations by *postmultiplication* of the modelview matrix.

Summary

What to take away from this lecture:

- All the names in boldface.
- How points and transformations are represented.
- How to compute lengths, dot products, and cross products of vectors, and what their geometrical meanings are.
- What all the elements of a 2 x 2 transformation matrix do and how these generalize to 3 x 3 transformations.
- What homogeneous coordinates are and how they work for affine transformations.
- How to concatenate transformations.
- The mathematical properties of affine transformations.