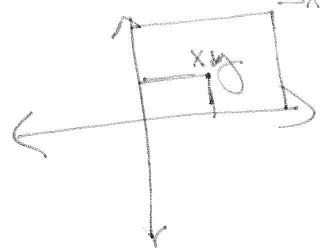
Slide #6

Scale

$$\begin{bmatrix} S_x x \\ S_y y \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$S_x x, S_y y$ where $S_x = 2$, $S_y = 2$

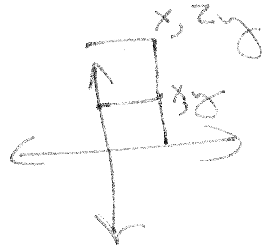


conceptually: transform each point of an object by ~~S_x, S_y~~ $\begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}$ to get new, scaled version

actually: transform each vertex

Non uniform scale

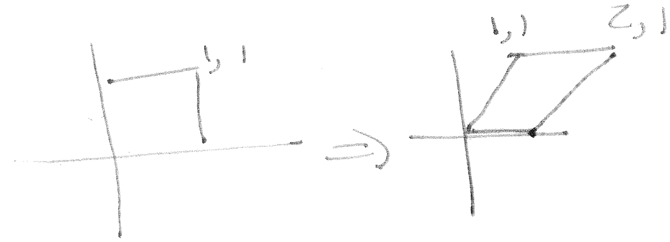
$$\begin{bmatrix} x \\ 2y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Shear

shear in x $\begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}$ shear in y $\begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix}$

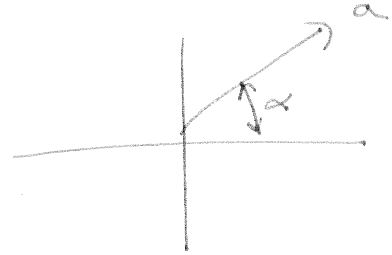
$$\begin{bmatrix} x+y \\ y \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Slide # 6

Rotation:

vector makes an angle α with the x-axis, length r

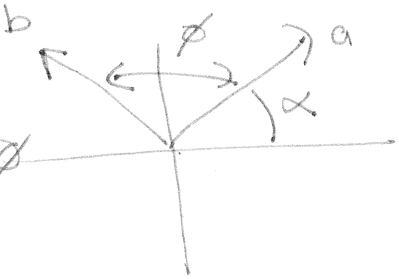


$$x_a = r\cos\alpha$$

$$y_a = r\sin\alpha$$

$$r^2 = x_a^2 + y_a^2$$

want to rotate vector a by angle ϕ counterclockwise to get vector b



$$x_b = r\cos(\alpha+\phi)$$

$$= \underbrace{r\cos\alpha}_{x_a}\cos\phi - \underbrace{r\sin\alpha}_{y_a}\sin\phi$$

(from basic trig)

$$y_b = r\sin(\alpha+\phi)$$

$$= \underbrace{r\sin\alpha}_{y_a}\cos\phi + \underbrace{r\cos\alpha}_{x_a}\sin\phi$$

in matrix form

$$\text{rotate}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}$$

Slide 7

Composition of 2D Transforms

example: first scale by S
then rotate by R

$$v_2 = S v_1 \qquad v_3 = R v_2$$

$$\Rightarrow \quad \underline{v_3} = R(S v_1)$$

$$\underline{v_3} = (RS) v_1$$

$$M = RS$$

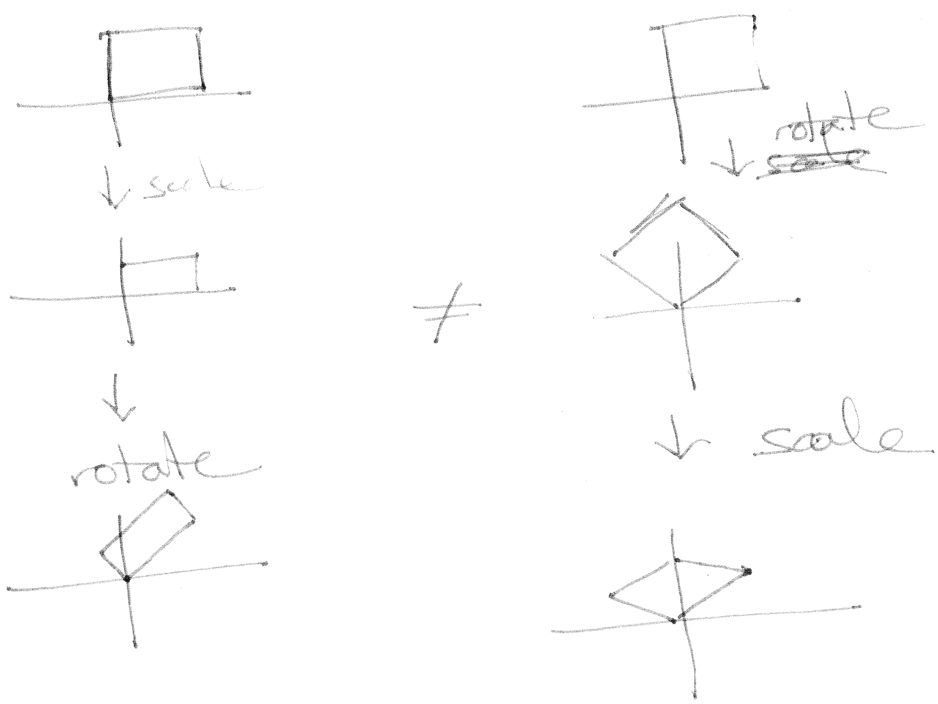
$$v_3 = M v_1$$

order matters!

Matrix multiplication is <u>not</u> communative

$$RS \neq SR$$

$M = RS$ will first scale + then rotate



Slide #8

Basic 3D Transforms

$$\text{scale}(s_x, s_y, s_z) = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix}$$

$$\text{shear}(d_x, d_y) = \begin{bmatrix} 1 & d_y & d_x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

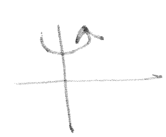
rotation ⇒ more complicated
what axis do you want to rotate about?

$$\text{rotate-z}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{rotate-x}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

$$\text{rotate-y}(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

No change in value for the ~~coordinate~~ axis being rotated about



Rotation Matrices are orthonormal

3 rows of matrix are mutually orthogonal unit vectors

~~# for~~ 3 columns are also

The inverse of a orthonormal matrix is its transpose

$$M^{-1} = M^{T} \Longleftarrow \text{also geometric inverse under action of } M$$

$$x = M^{-1} M x$$

very handy $\Rightarrow$ can compute inverse very cheaply!

See 2.4.5 for definition of orthonormal

$u \cdot v = 0$ $\qquad$ $u \cdot v = \underbrace{\|u\|}_{1}\,\underbrace{\|v\|}_{1}\,\underbrace{\cos\phi}_{0 \Rightarrow \phi = 90^\circ}$

$\|u\| = \|v\| = 1$

Translation:

What we have seen so far has the form

$$x' = m_{11}x + m_{12}y$$
$$y' = m_{21}x + m_{22}y$$

No way to represent translation?

$$x' = x + x_t$$
$$y' = y + y_t$$

$\Longrightarrow$ add a dimension to the transformation matrix

2D becomes

$$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_t \\ y + y_t \\ 1 \end{bmatrix}$$

Note: this is a 3D shear with z set = 1

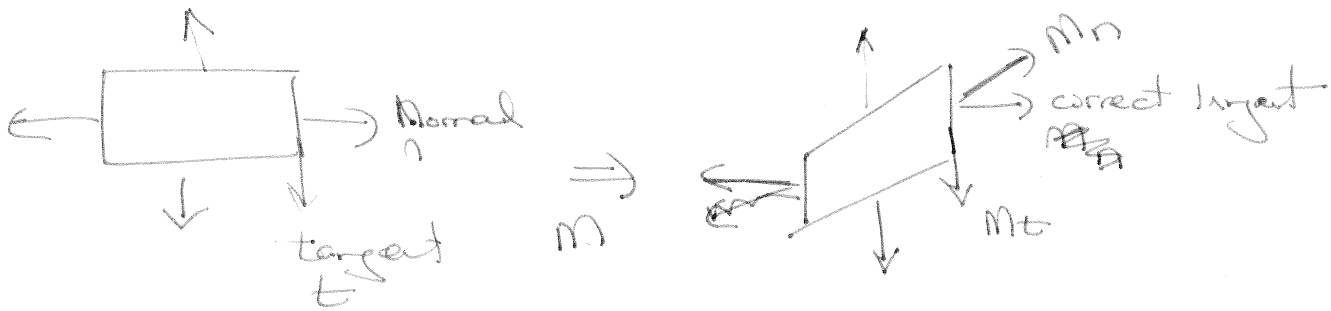
In 3D

$$\begin{bmatrix} 1 & 0 & 0 & x_t \\ 0 & 1 & 0 & y_t \\ 0 & 0 & 1 & z_t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_t \\ y + y_t \\ z + z_t \\ 1 \end{bmatrix}$$

Does break the $M^T = M^{-1}$ property that we had for rotation matrices!

if $M = M_1 M_2 M_3$ then $M^{-1} = M_3^{-1} M_2^{-1} M_1^{-1}$

Transforming Normal Vectors 6.2.2



Normals don't work
Tangents do
Normals + Tangents should be $\perp \Rightarrow$ dot product $= 0$

$$n^T t = 0$$

We want

$$t_M = Mt \qquad n_N = Nn$$

need to find N such that

$$n_N^T t_M = 0 \quad (\text{normal + tangent } \perp)$$

Algebraic trick:

$$n^T t = n^T \underline{I} t = n^T M^{-1} M t = 0$$

sneak in Identity matrix $\qquad M^{-1}M = I$

$$(n^T M^{-1})(Mt) = (n^T M^{-1}) t_M = 0$$

$$t_M = Mt$$

$$n_N^T = n^T M^{-1}$$

take the transpose to get

$$n_N = (n^T M^{-1})^T = (M^{-1})^T n$$

$N = (M^{-1})^T \Leftarrow$ what we wanted see p. 150 + Section 5.2.3 for a nice way to ~~pre~~compute N from the elements of N without expensive operations