What's a matrix?

- A system of (first order) equations
- Our mathematical tool to tweak 3D functions or point clouds with
- Remember that a matrix is shorthand:

$$\begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} \Rightarrow \begin{array}{l} x_{new} = 2x_{old} + 3y_{old} \\ y_{new} = 4x_{old} + 5y_{old} \end{array}$$

Matrix Concepts

- Guiding a shape into place in a scene
- A few common matrices perform movement or warp space.
- Recall elementary row operations:

Elementary Matrices

An elementary matrix E is an $n \times n$ matrix that can be obtained from the identity matrix I_n by one elementary row operation.

$$E = e(I)$$

where e is an elementary row operation.

- *All elementary matrices are invertible, an inverse exists.*
- *The inverse of an elementary matrix is also an elementary matrix.*

Recall identity matrices

$$[1] \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \dots$$

Row Replacement:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -4 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g-4a & h-4b & i-4c \end{bmatrix}$$

Interchange of Rows:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} d & e & f \\ a & b & c \\ g & h & i \end{bmatrix}$$

Multiplication by a constant:

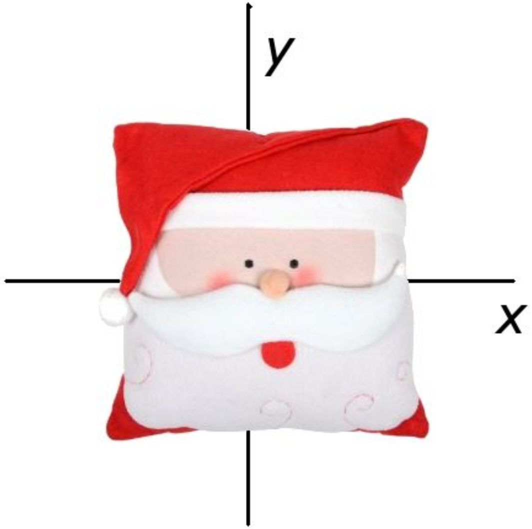
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 5 \end{bmatrix} \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} d & e & f \\ a & b & c \\ 5g & 5h & 5i \end{bmatrix}$$

OOPS Pretend these two rows are not swapped

What if we apply e_1 to all the points of an image?

Interchange of Rows:

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



What if we apply e_1 to all the points of an image?

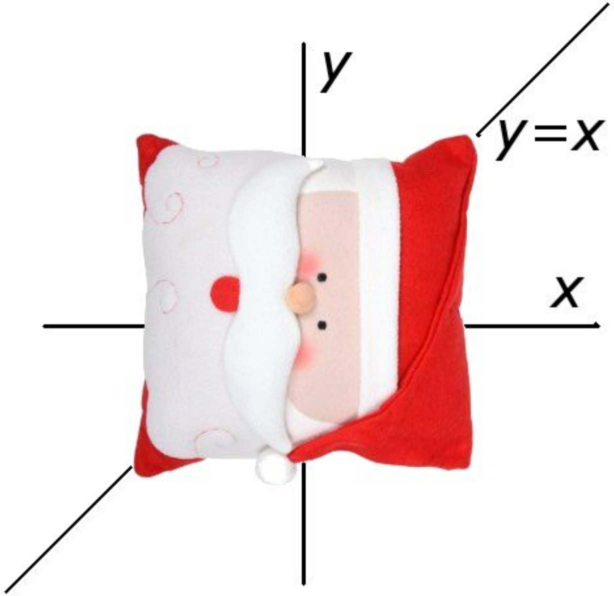
We get an axis swap.

$new_x = y, \; new_y = x$

Matrix form of that:

Interchange of Rows:

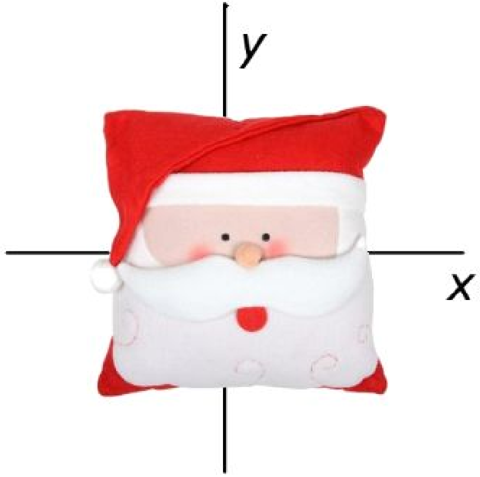
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = ?$$



What if we apply e_2 to all the points of an image?

Multiplication by a constant:

$$\begin{bmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = ?$$



What if we apply e_2 to all the points of an image?

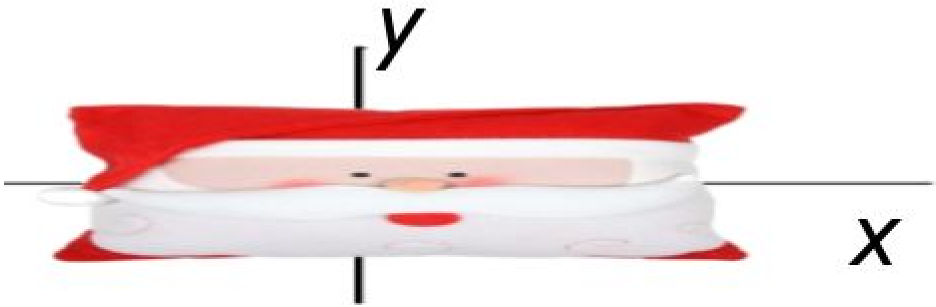
We get a scale.

$new_x = 5x$

Matrix form of that:

Multiplication by a constant:

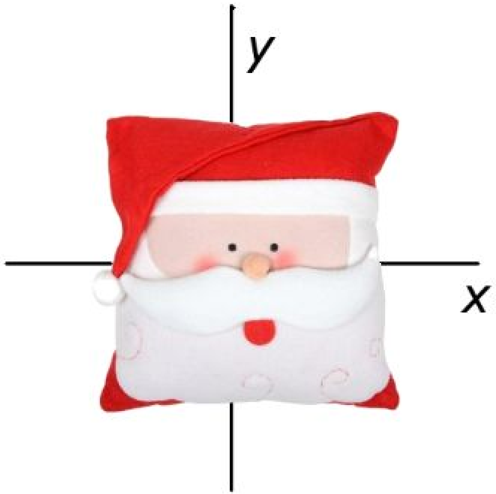
$$\begin{bmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = ?$$



What if we apply e_3 to all the points of an image?

Row Replacement:

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



What if we apply e_3 to all the points of an image?

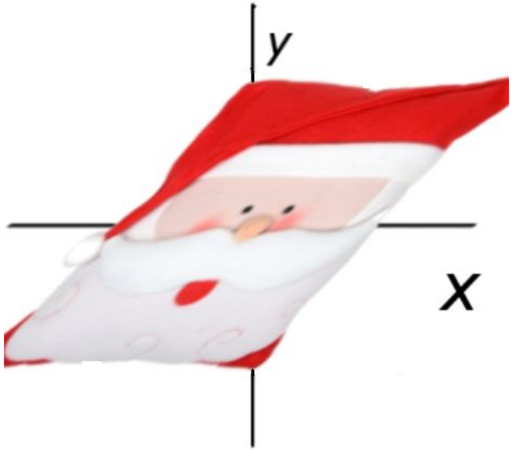
We get a shear.

$new_x = x + y$

Matrix form of that:

Row Replacement:

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



Rotation Matrices

A rotation is two shears.

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

But certain properties have to hold. This one is not a pure rotation matrix (it has some scaling effect too). Rotations have to have determinant = 1.

Rotation Matrices

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

A rotation also has to have orthogonal columns. This one passes that!

Our simpler shear matrix from earlier did not (check the columns):

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

Rotation Matrices

$$\begin{bmatrix} cos\Theta & -sin\Theta & \\ sin\Theta & cos\Theta & \\ & & 1 \end{bmatrix}$$

This one is more like it. **Trig** identities ensure this matrix always has determinant=1. The two columns are also always perpendicular. Suppose theta is 45 degrees:

Rotation Matrices

$$\begin{bmatrix} cos\Theta & -sin\Theta & \\ sin\Theta & cos\Theta & \\ & & 1 \end{bmatrix}$$

This one is more like it. **Trig** identities ensure this matrix always has determinant=1. The two columns are also always perpendicular. Suppose theta is 45 degrees:

$$\begin{bmatrix} cos45^\circ & -sin45^\circ & \\ sin45^\circ & cos45^\circ & \\ & & 1 \end{bmatrix} = \begin{bmatrix} .5\sqrt{2} & -.5\sqrt{2} & \\ .5\sqrt{2} & .5\sqrt{2} & \\ & & 1 \end{bmatrix}$$

Rotation Matrices

Let’s try our rotation matrix, but we don’t want to multiply ugly radicals so let’s combine it with a scale matrix for testing.

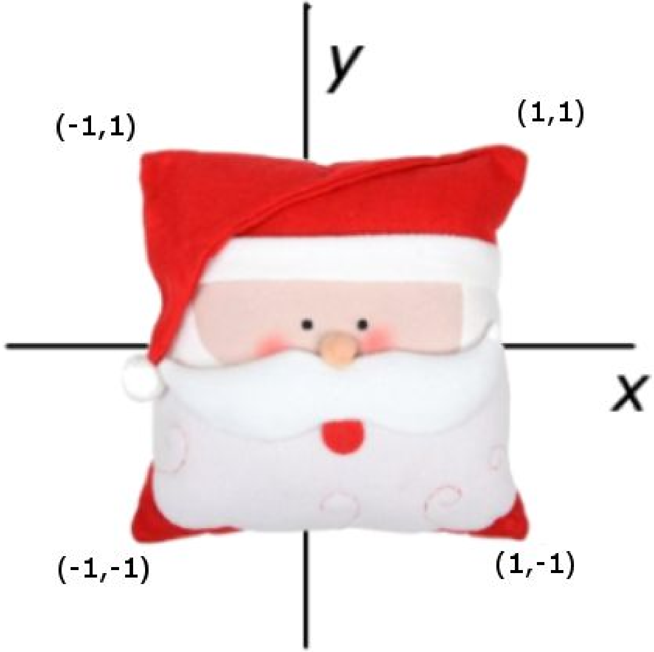
$$\begin{bmatrix} .5\sqrt{2} & -.5\sqrt{2} & \\ .5\sqrt{2} & .5\sqrt{2} & \\ & & 1 \end{bmatrix} * \begin{bmatrix} \sqrt{2} & & \\ & \sqrt{2} & \\ & & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & \\ 1 & 1 & \\ & & 1 \end{bmatrix}$$

There, much better to work with. (This was our matrix from earlier that had a scaling influence).

Rotation Matrices

What it means:

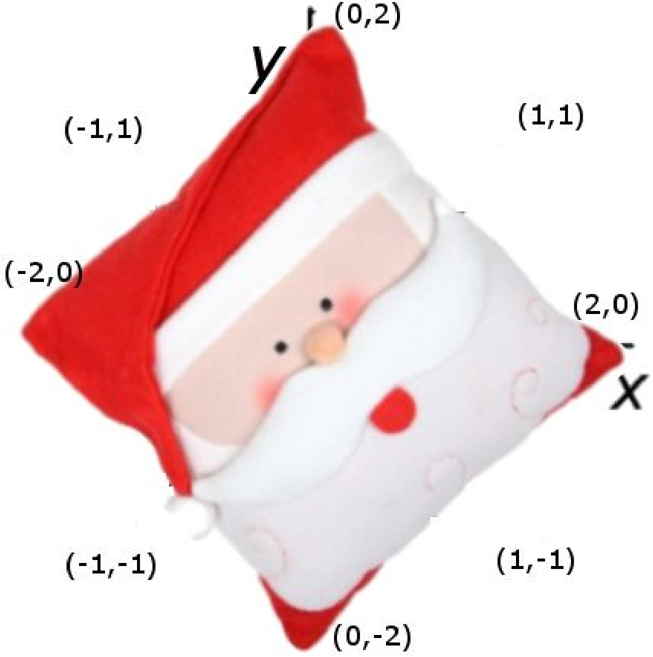
$$\begin{bmatrix} 1 & -1 & \\ 1 & 1 & \\ & & 1 \end{bmatrix} \Rightarrow \begin{matrix} x_{new} = x_{old} - y_{old} \\ y_{new} = x_{old} + y_{old} \end{matrix}$$



Rotation Matrices

What it means:

$$\begin{bmatrix} 1 & -1 & \\ 1 & 1 & \\ & & 1 \end{bmatrix} \Rightarrow \begin{array}{l} x_{new} = x_{old} - y_{old} \\ y_{new} = x_{old} + y_{old} \end{array}$$



All this time...

SANTA’S NOSE HAS NEVER LEFT THE ORIGIN ***(!!!)***

None of these operations have resulted in **translation** (sliding all of the points at once somewhere else)

Translation Matrices

- Every matrix we've discovered using the elementary ones only has a linear effect, stretching and warping the image around the origin in various ways
- Why can't we accomplish translation of Santa to other places with our current machinery?
- Every point always takes a contribution of x_{old}, y_{old}, and z_{old} to get its value, and so the origin point (0,0,0) will always contribute zero to each axis for the new point, mapping onto itself.
- We need to be able to take a component of some other quantity besides x_{old}, y_{old}, and z_{old}
- Our vector we're multiplying needs to grow by 1 term.

Translation Matrices

Solution:

$$\begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \\ z_{old} \\ 1 \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \\ 1 \end{bmatrix}$$

One of the components that contributes to the answer is now the constant 1, which we can pull from even when all others are zero. This allows us to do a non-linear, affine operation to the picture - moving it.

Translation Matrices

Solution:

$$\begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \\ z_{old} \\ 1 \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \\ 1 \end{bmatrix}$$

We'd like our new vector to have "1" in the 4^{th} coordinate as well, so we can use matrices on it right away. What does our matrix have to be like to ensure that will happen?

Translation Matrices

Solution:

$$\begin{bmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \\ z_{old} \\ 1 \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \\ 1 \end{bmatrix}$$

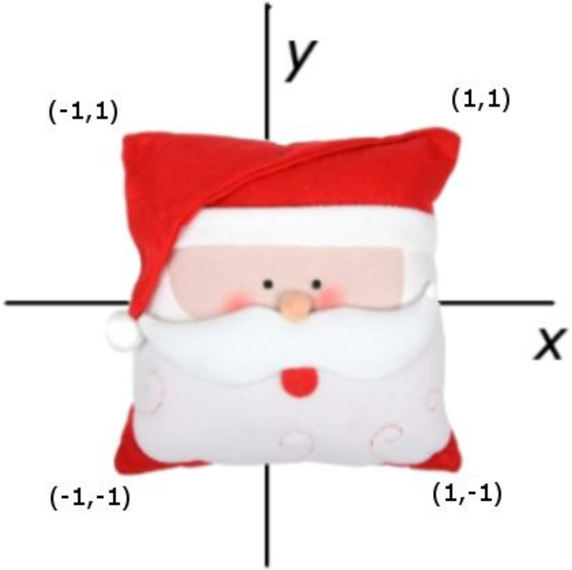
Just make sure the bottom row is like above. This 4th row is an equation that literally says "$1_{new}=1_{old}$".

Translation Matrices

3D translations are the <u>reason</u> we use 4x4 matrices instead of 3x3.

Let's do one:

$$\begin{bmatrix} 1 & & & 1 \\ & 1 & & 1 \\ & & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \\ z_{old} \\ 1 \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \\ 1 \end{bmatrix}$$

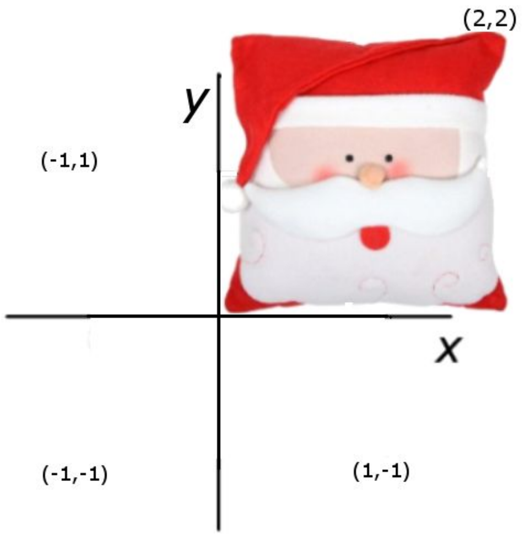


Translation Matrices

3D translations are the reason we use 4x4 matrices instead of 3x3.

Let's do one:

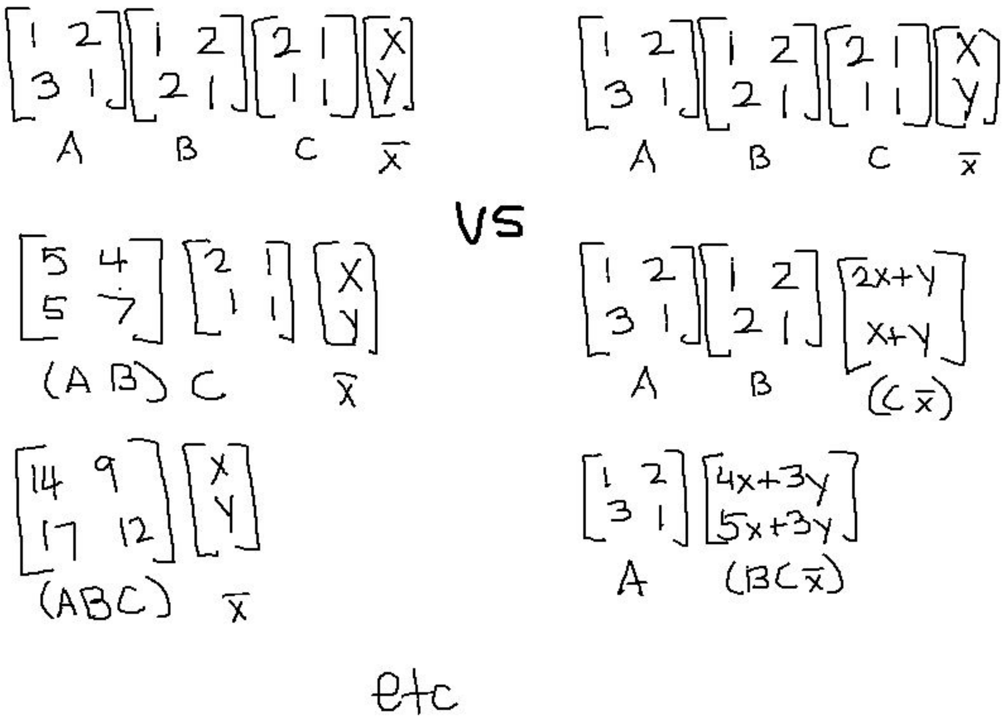
$$\begin{bmatrix} 1 & & & 1 \\ & 1 & & 1 \\ & & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{old} \\ y_{old} \\ z_{old} \\ 1 \end{bmatrix} = \begin{bmatrix} x_{new} \\ y_{new} \\ z_{new} \\ 1 \end{bmatrix}$$



Matrix Order

- The trickiest concept in the class - we'll look at it as many times as possible until it's clear. Might as well start seeing it now.
- Matrix products can only be written in one left-right order. Changing the order changes the answer.
- Matrix products can be evaluated in any left-right order you want though.
- Two common approaches. Multiply starting from:
  - Right to left (pre-multiply all new terms onto the product) or,
  - Left to right (post-multiply)
- The choice determines what your intermediate products are (points vs matrices?), and what each intermediate step intuitively means (an updated image vs. an updated basis to draw it in?).

Matrix order (example)



Option 1: Starting from left, post-multiply each matrix in turn before finally applying to point (moves the universe’s bases around for drawing a stationary point set)

Option 2: Starting on right, multiply each matrix onto the point in turn (moves points around a stationary universe)