Modeling Transformations

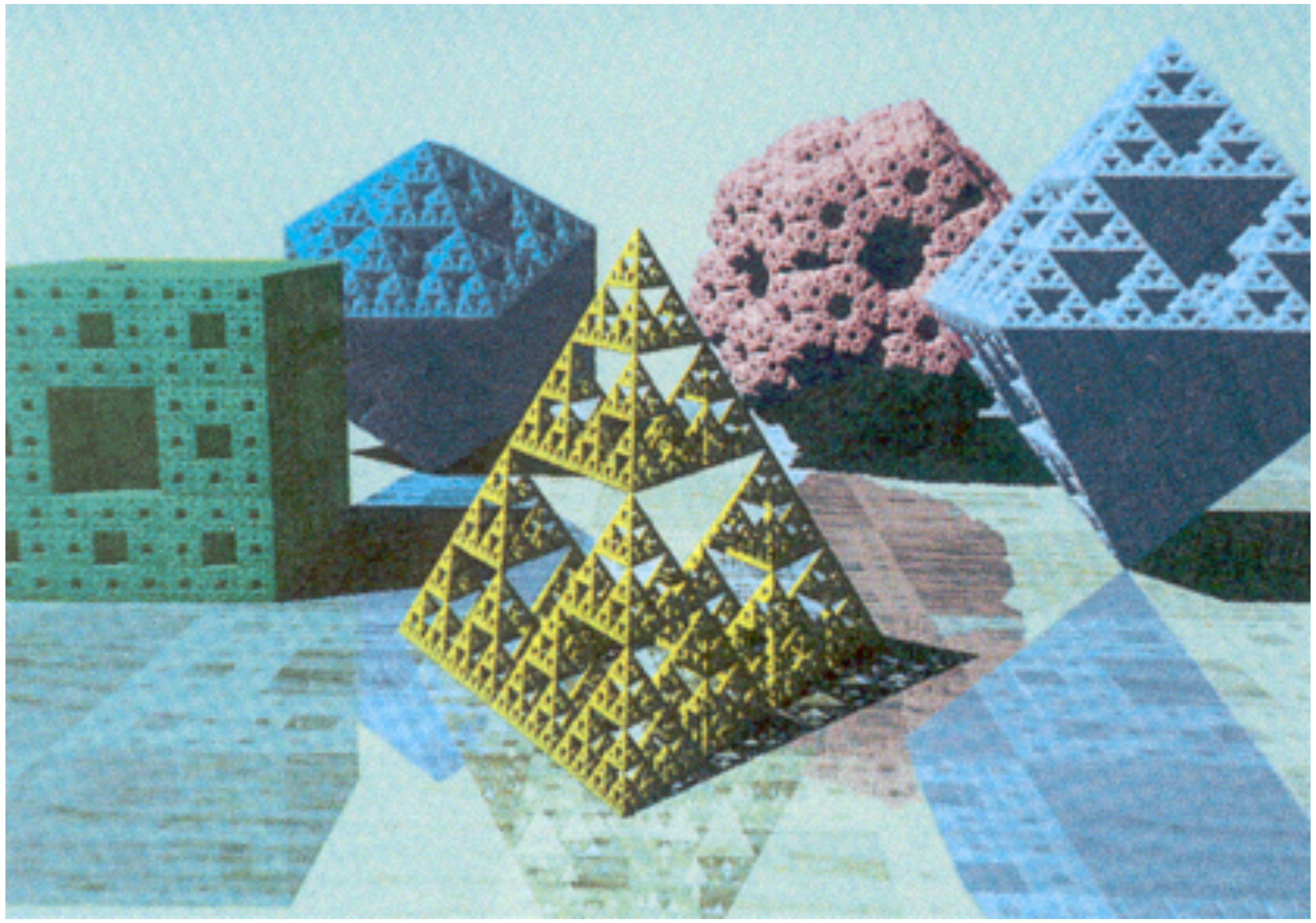
Connelly Barnes

CS 4810: Graphics

Acknowledgment: slides by Connelly Barnes, Misha Kazhdan, Allison Klein, Tom Funkhouser, Adam Finkelstein and David Dobkin

Modeling Transformations

- Specify transformations for objects
  - Allows definitions of objects in own coordinate systems
  - Allows use of object definition multiple times in a scene



H&B Figure 109

Overview

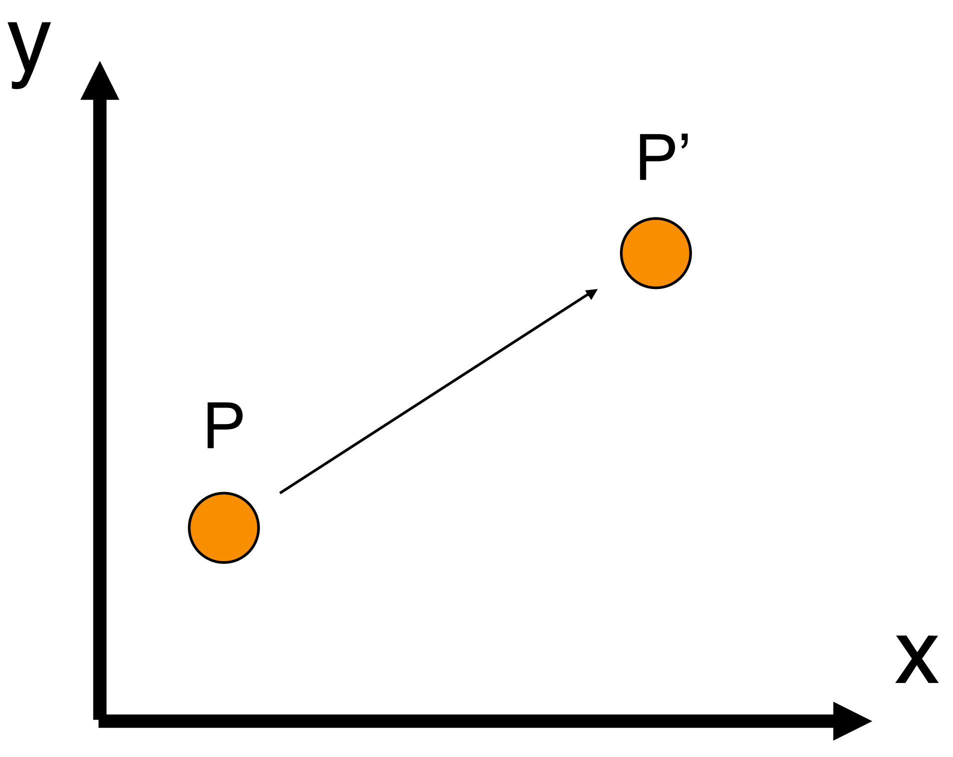
- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- 3D Transformations
  - Basic 3D transformations
  - Same as 2D

Simple 2D Transformations

Translation

$$p' = T + p$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} d_x \\ d_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

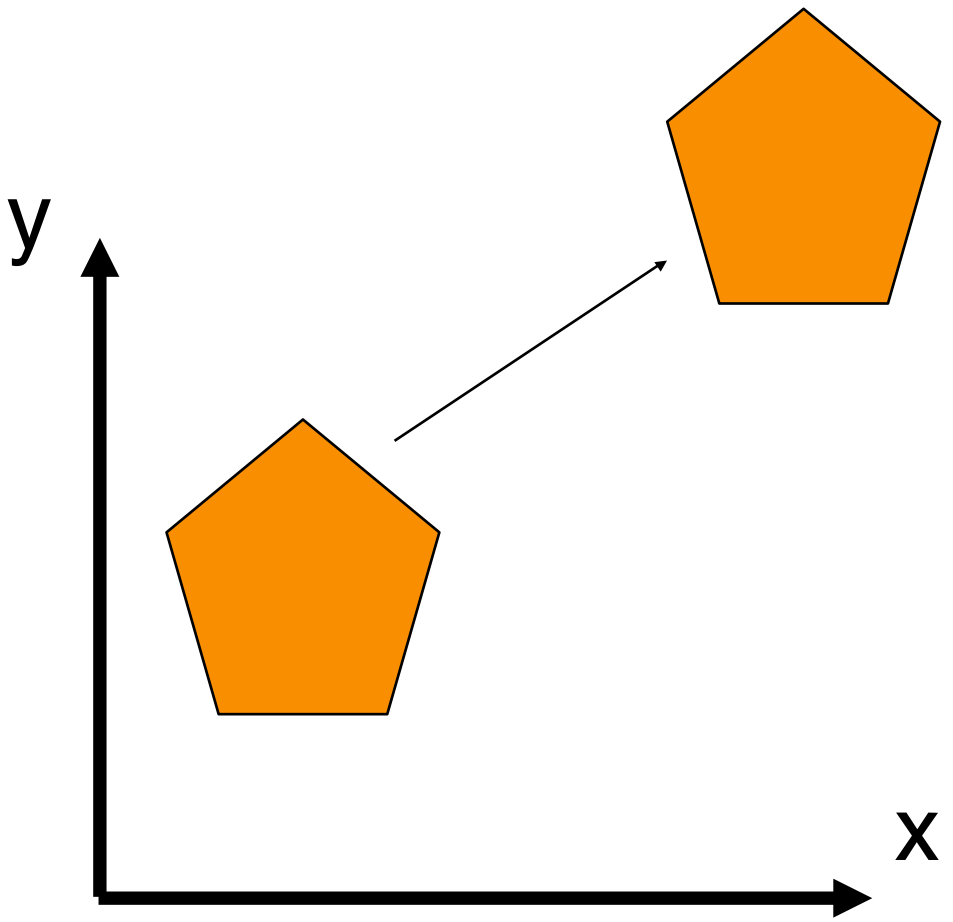


Simple 2D Transformations

Translation

$$p' = T + p$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} d_x \\ d_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

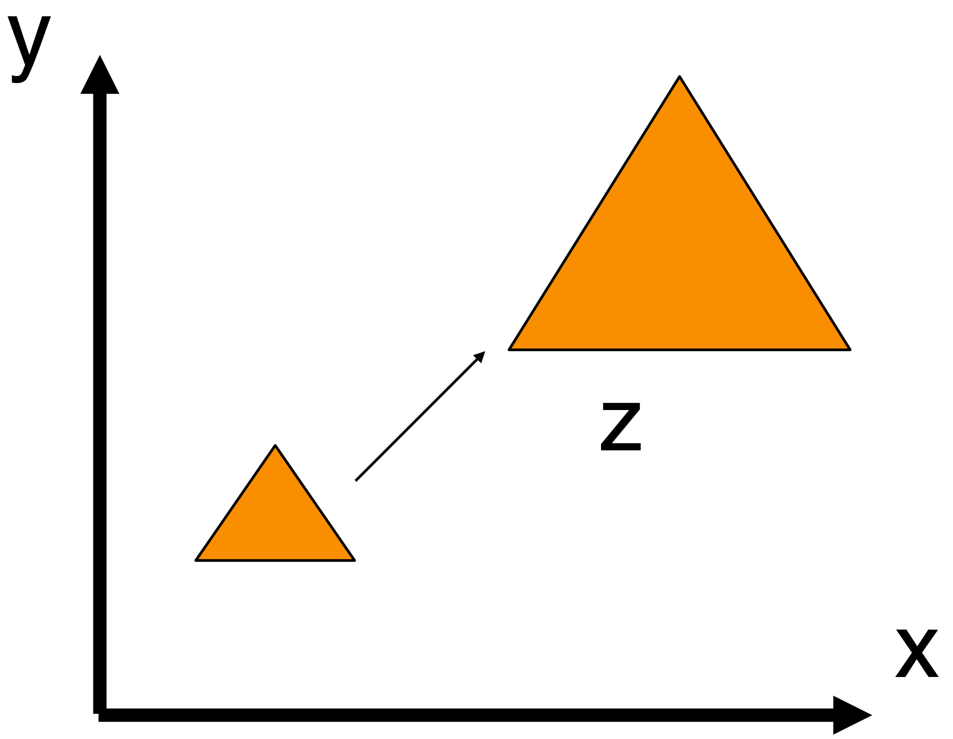


Simple 2D Transformations

Scale

$$p' = S \bullet p$$

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

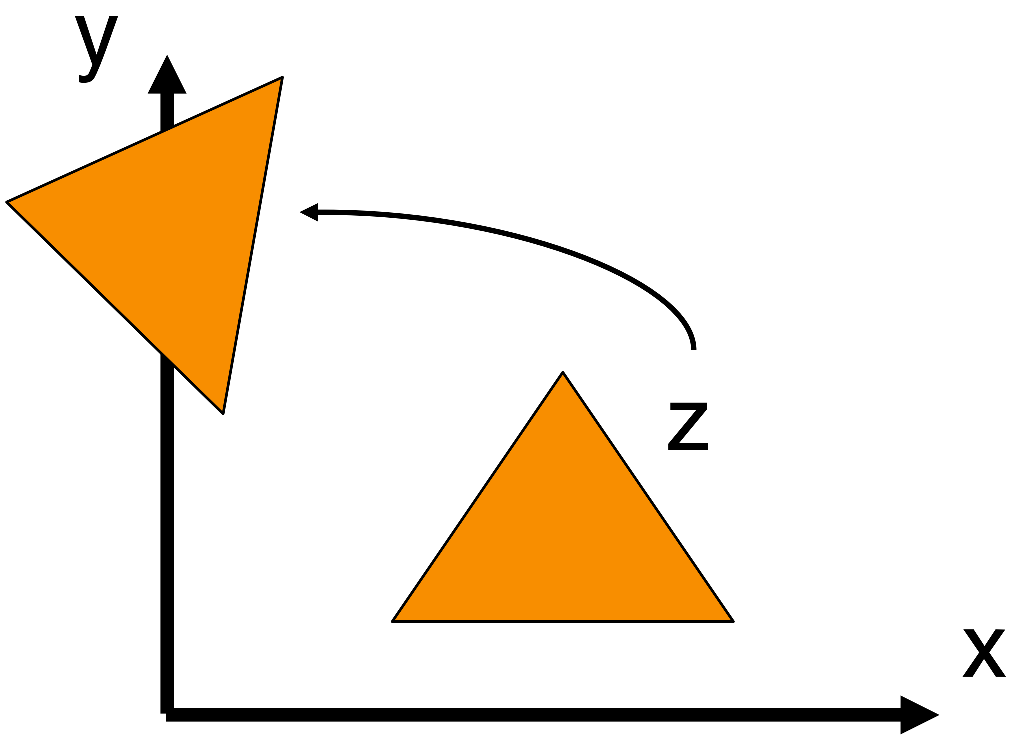


Simple 2D Transformation

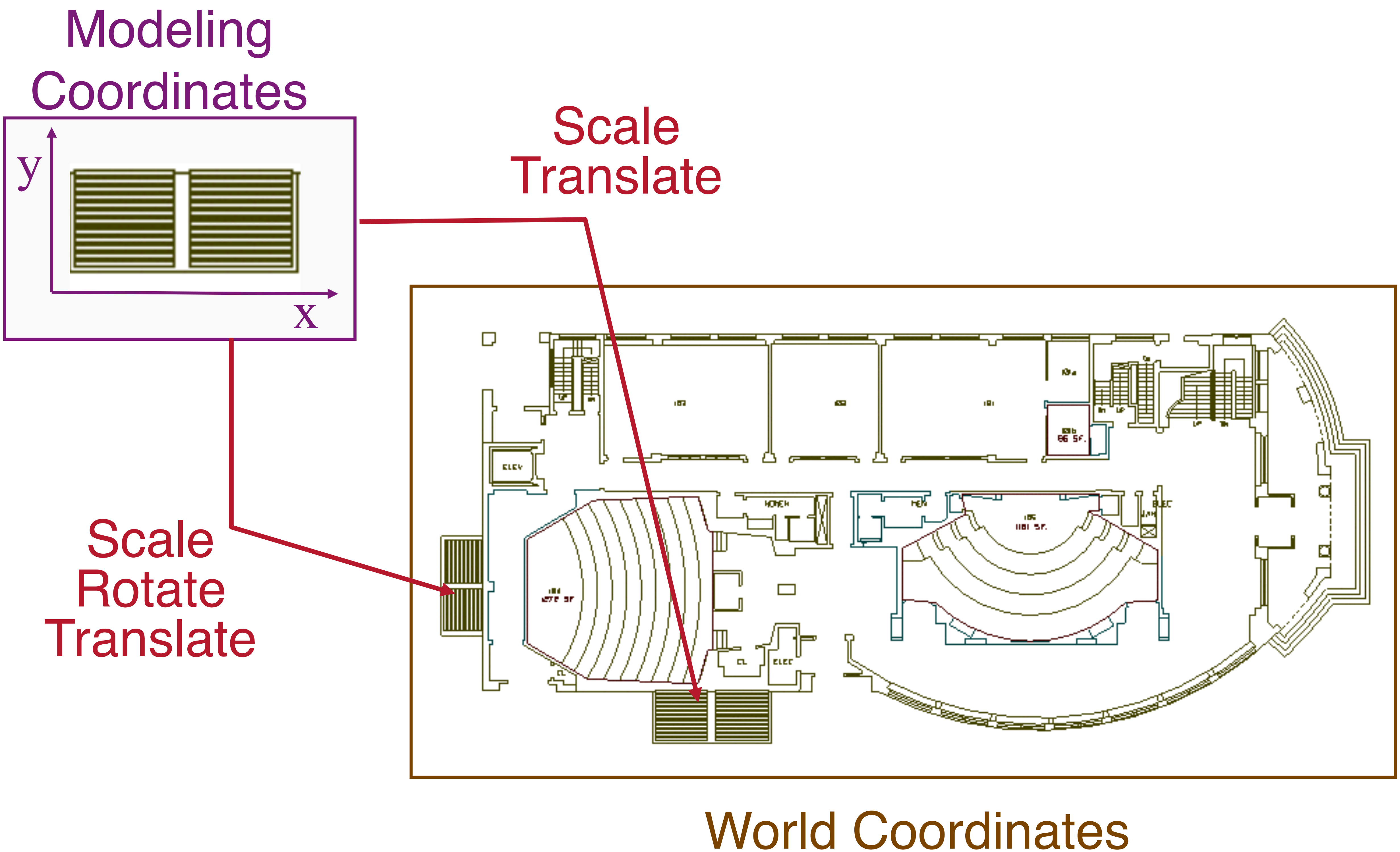
Rotation (around origin)

$$p' = R \bullet p$$

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

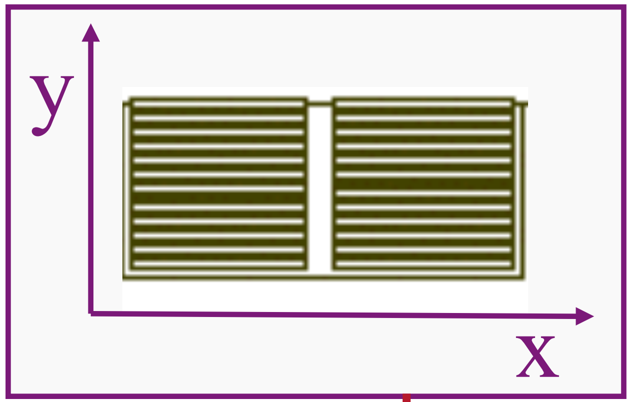


2D Modeling Transformations

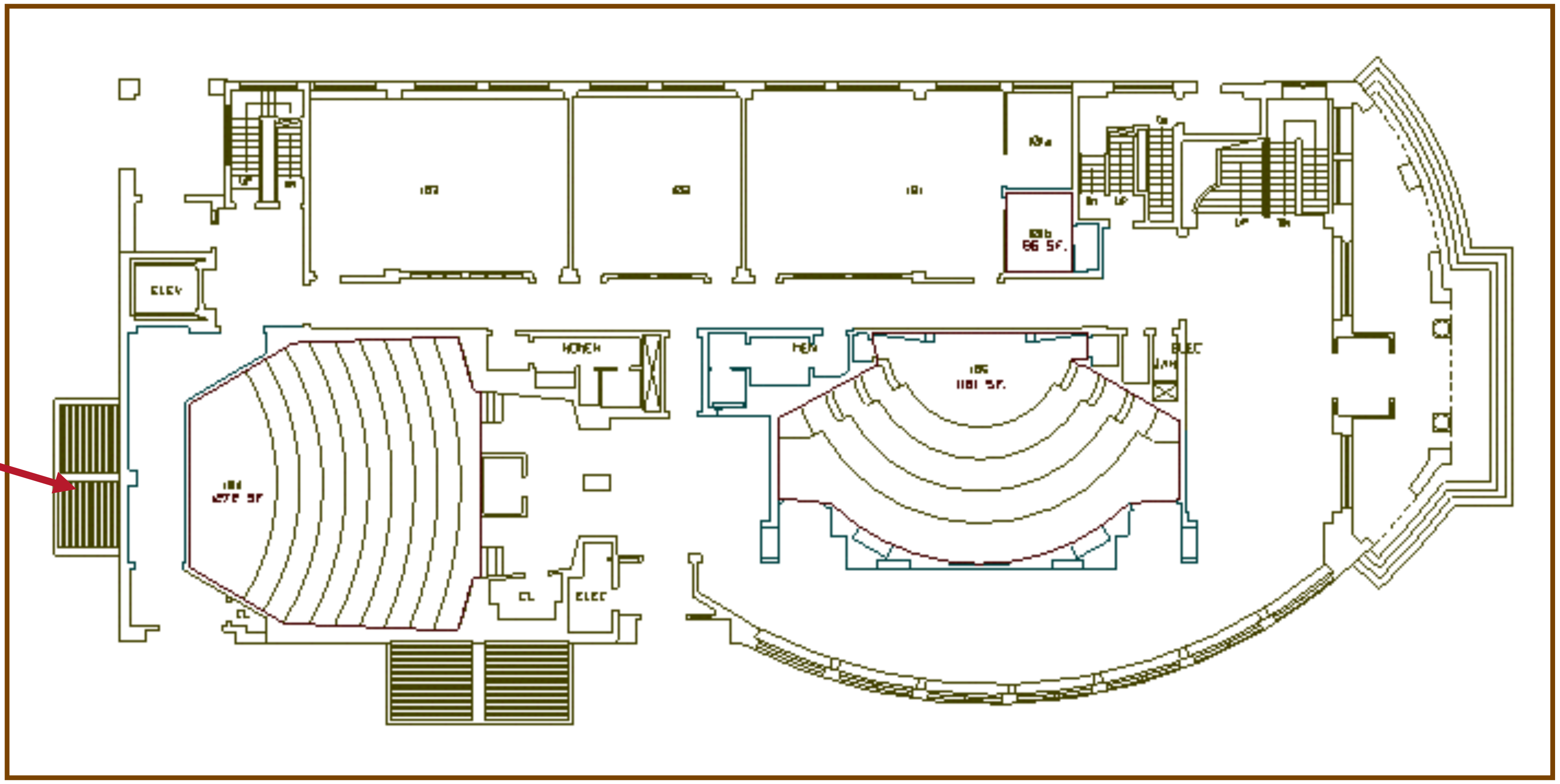


2D Modeling Transformations

Modeling Coordinates



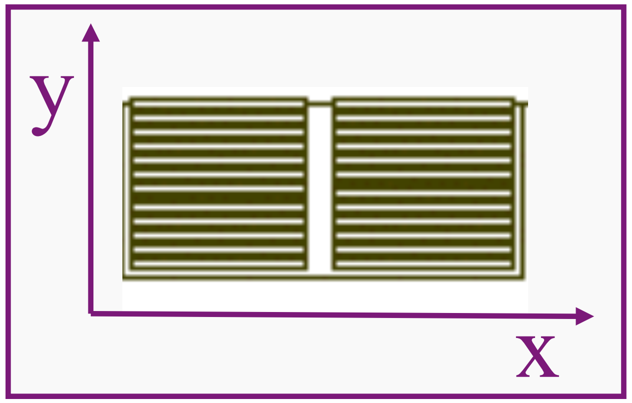
Let's look at this in detail…

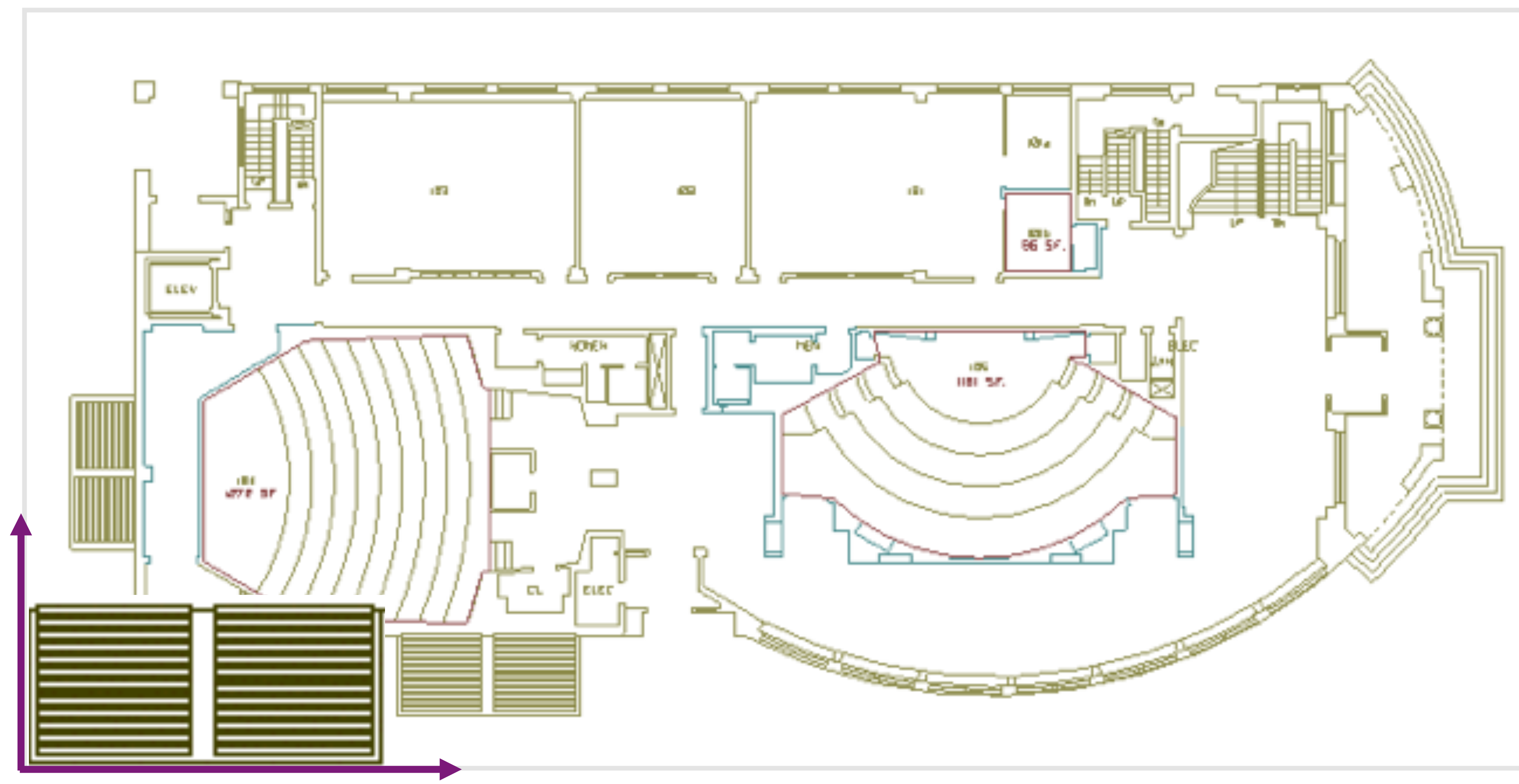


World Coordinates

2D Modeling Transformations

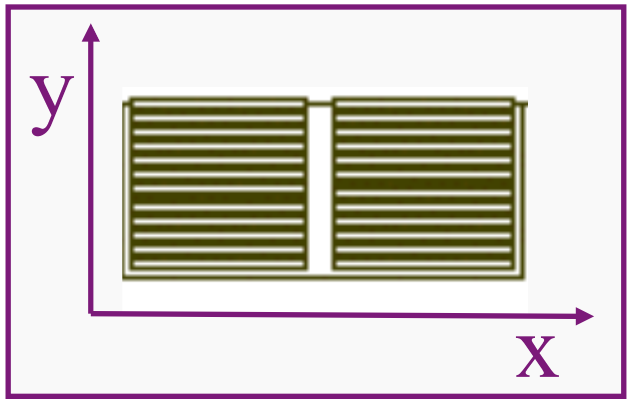
Modeling Coordinates



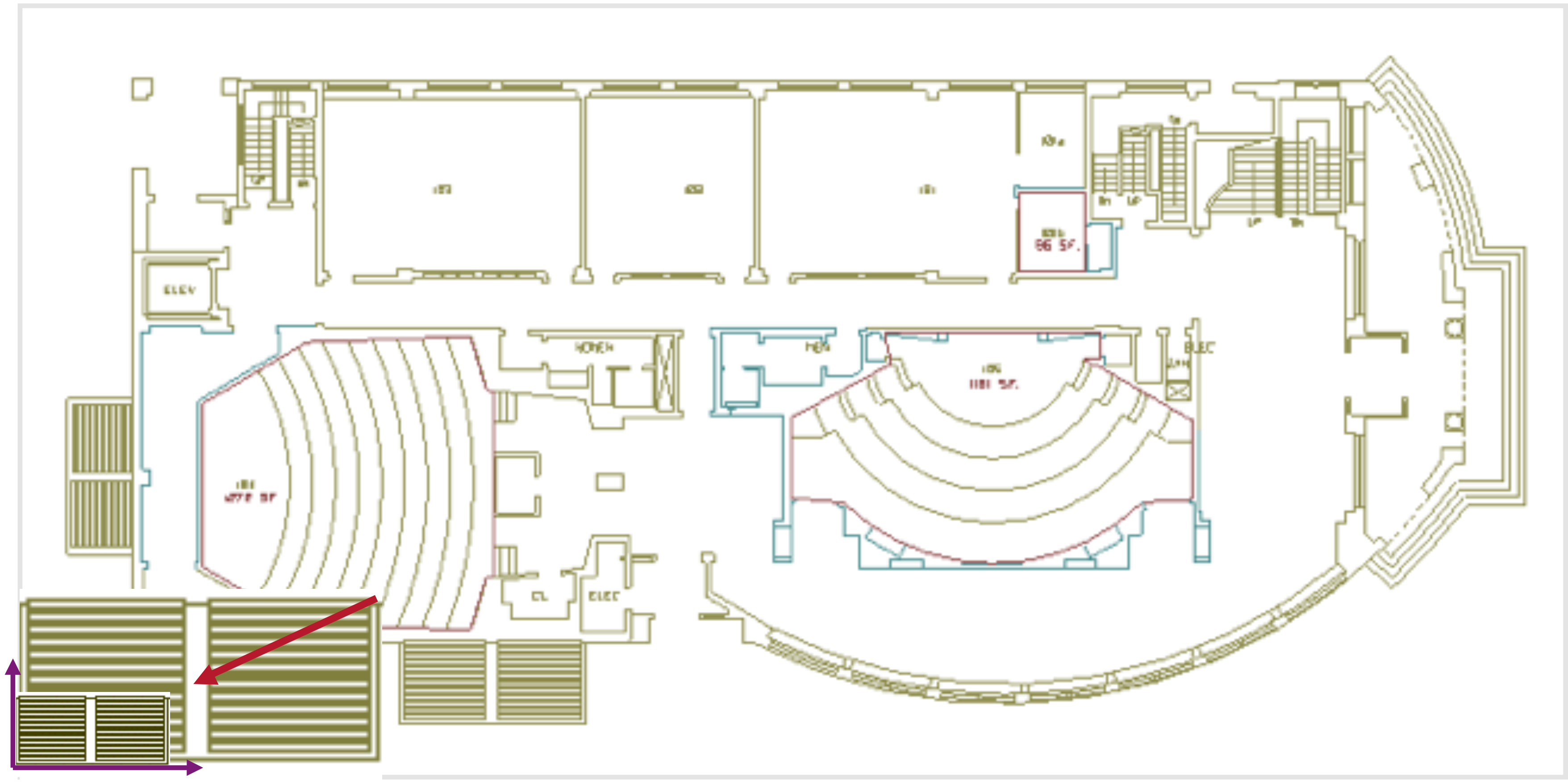


2D Modeling Transformations

Modeling Coordinates

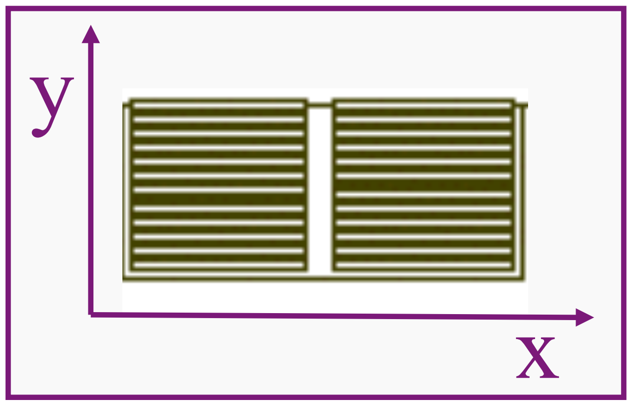


Scale .3, .3

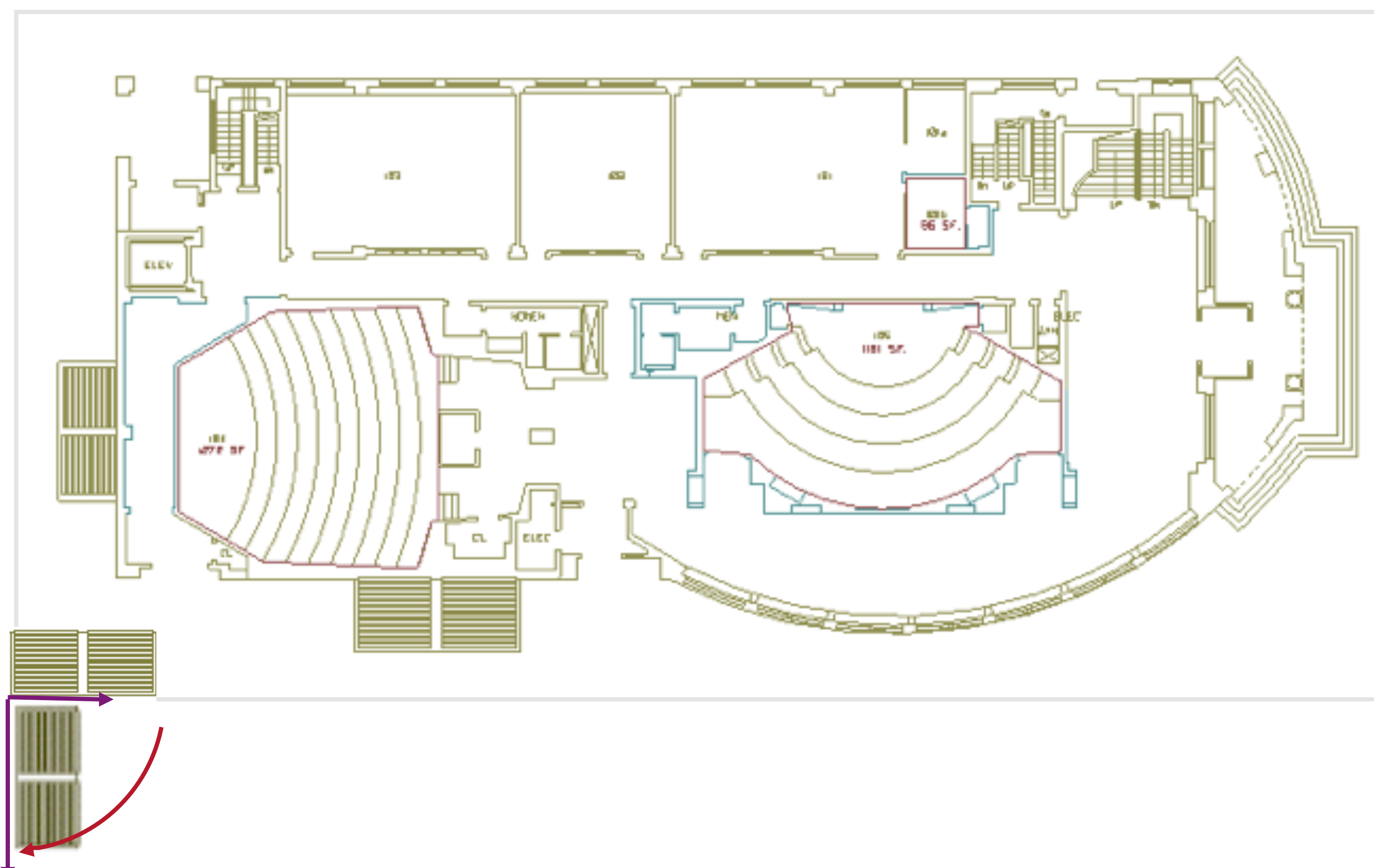


2D Modeling Transformations

Modeling Coordinates

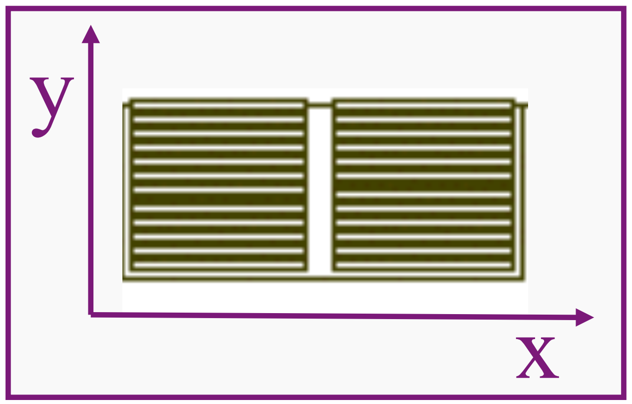


Scale .3, .3
Rotate -90

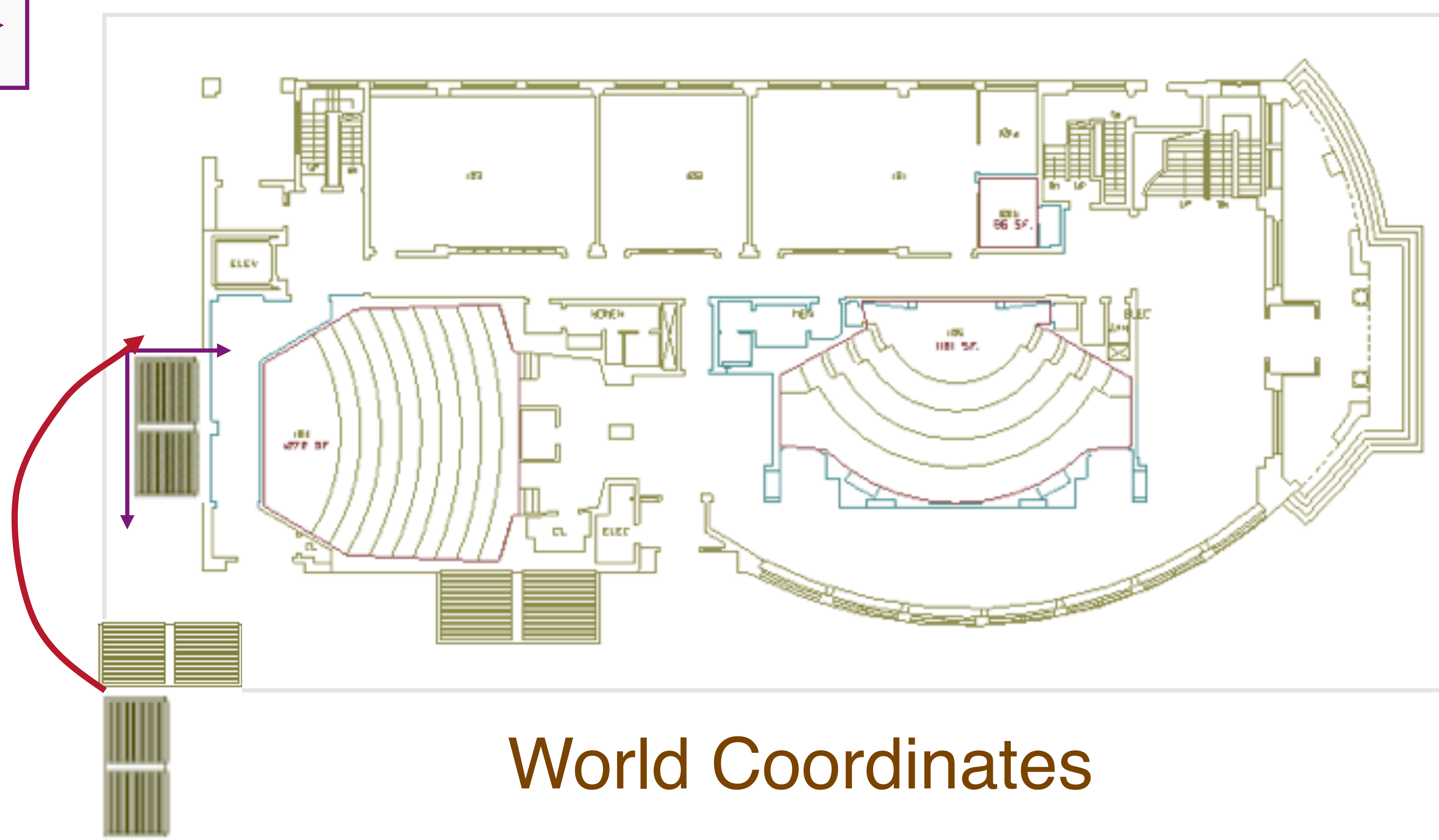


2D Modeling Transformations

Modeling Coordinates



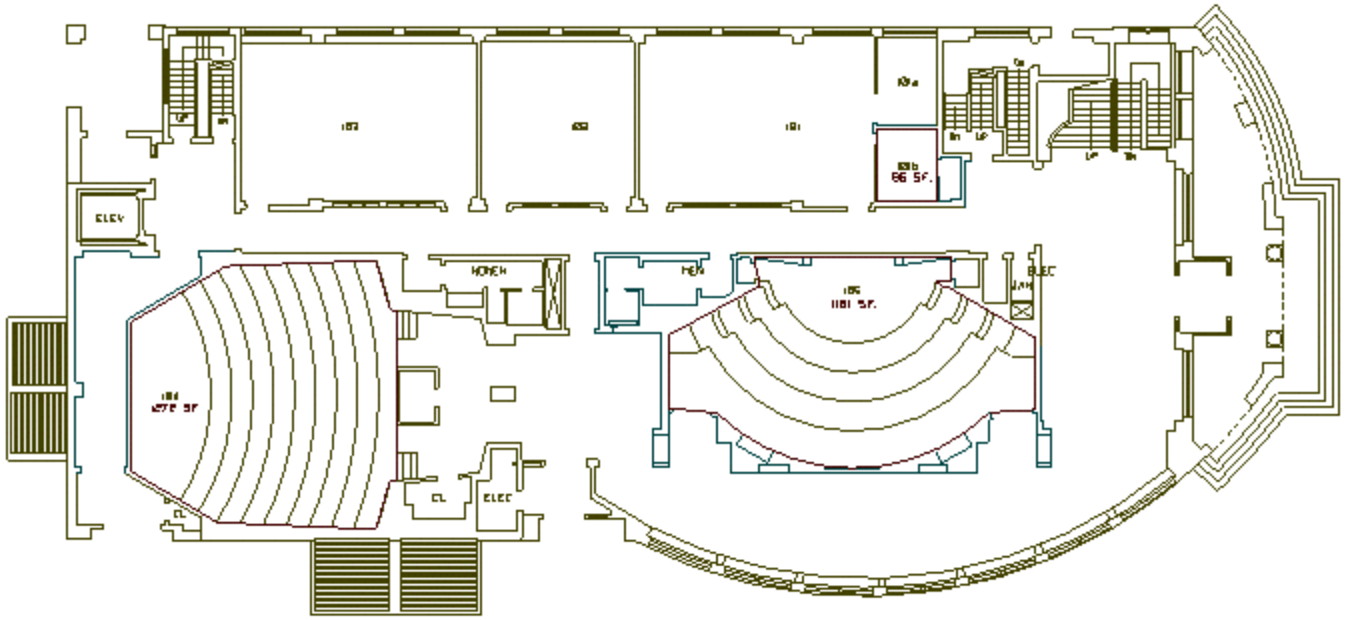
Scale .3, .3
Rotate -90
Translate 3, 5



World Coordinates

Basic 2D Transformations

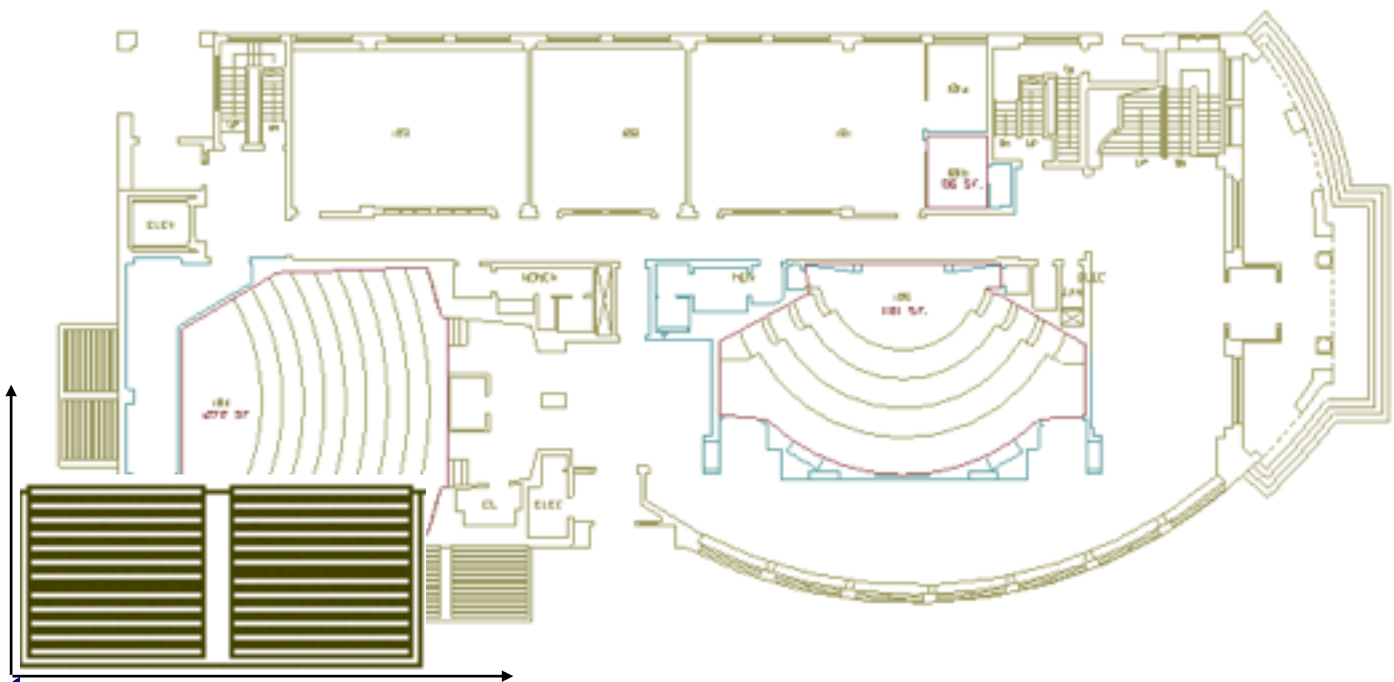
- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Rotation:
  - $x' = x*\cos\Theta - y*\sin\Theta$
  - $y' = x*\sin\Theta + y*\cos\Theta$



Transformations
can be combined
(with simple algebra)

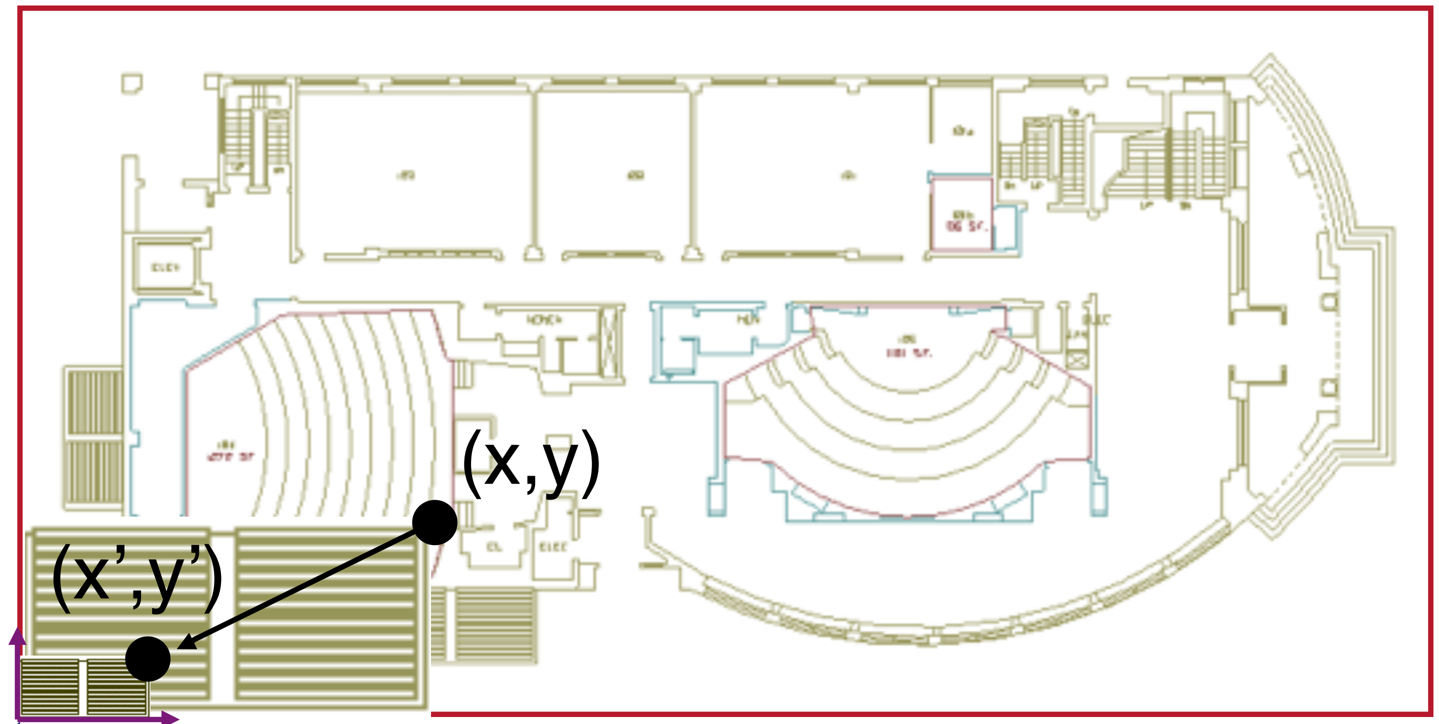
Basic 2D Transformations

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



Basic 2D Transformations

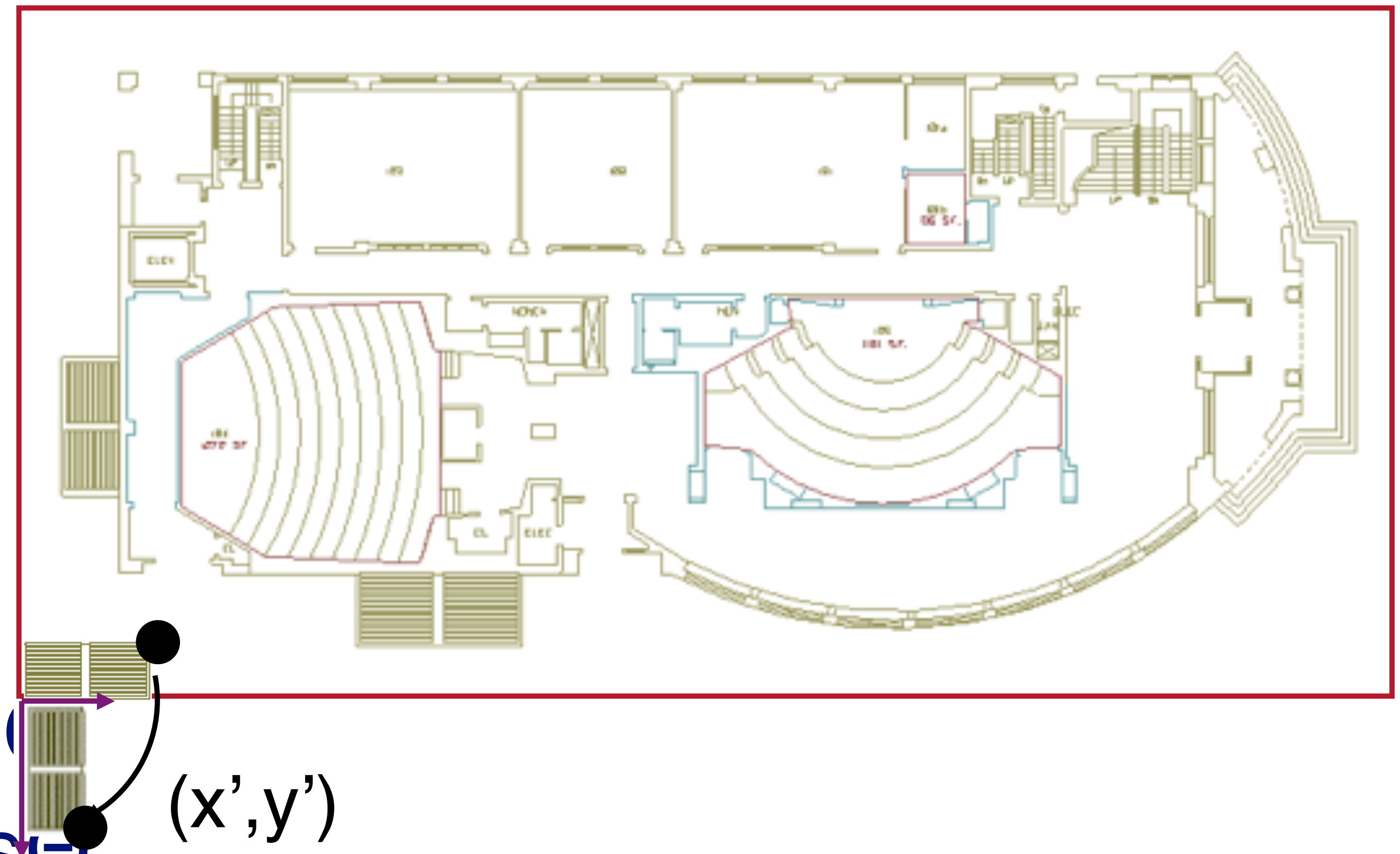
- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Rotation:
  - $x' = x*\cos\Theta - y*\sin\Theta$
  - $y' = x*\sin\Theta + y*\cos\Theta$



$x' = x*s_x$
$y' = y*s_y$

Basic 2D Transformations

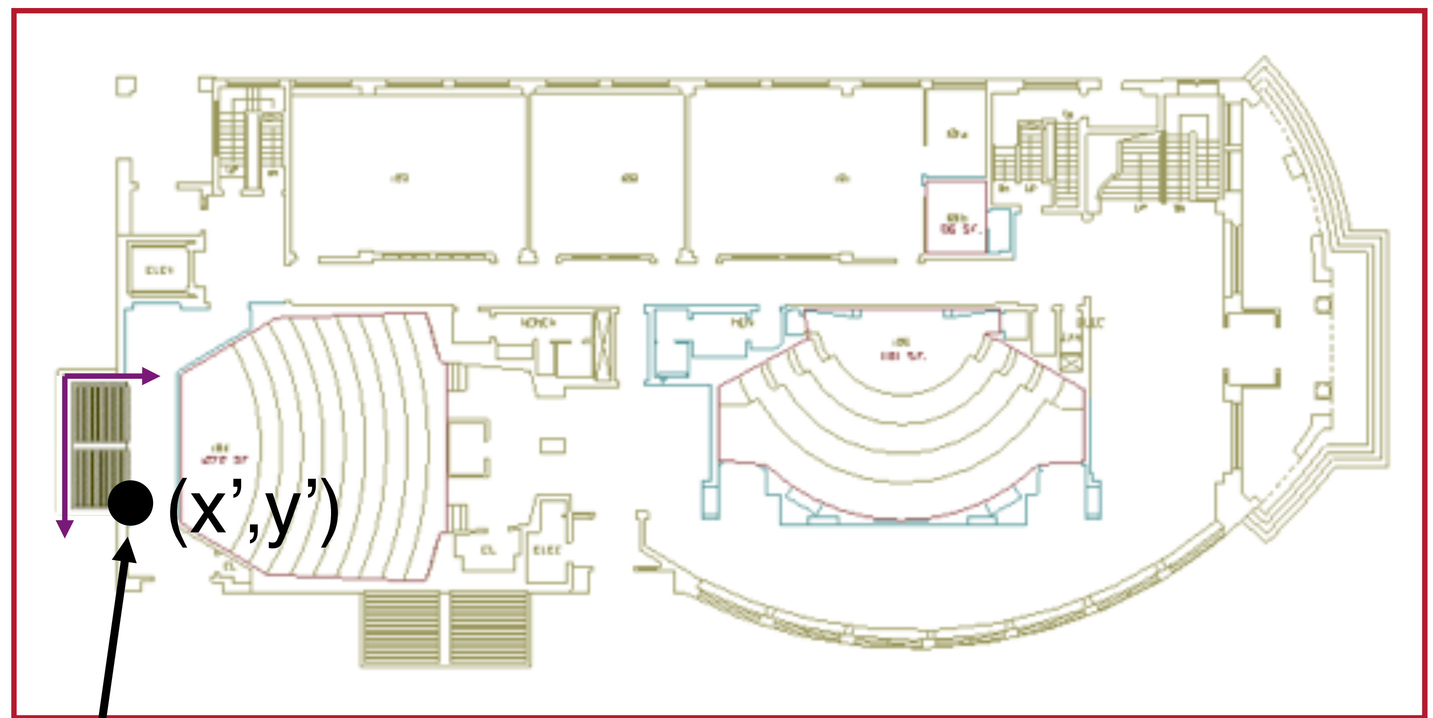
- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Rotation:
  - $x' = x*\cos\Theta - y*\sin\Theta$
  - $y' = x*\sin\Theta + y*\cos\Theta$



$$x' = (x*s_x)*\cos\Theta - (y*s_y)*\sin\Theta$$
$$y' = (x*s_x)*\sin\Theta + (y*s_y)*\cos\Theta$$

Basic 2D Transformations

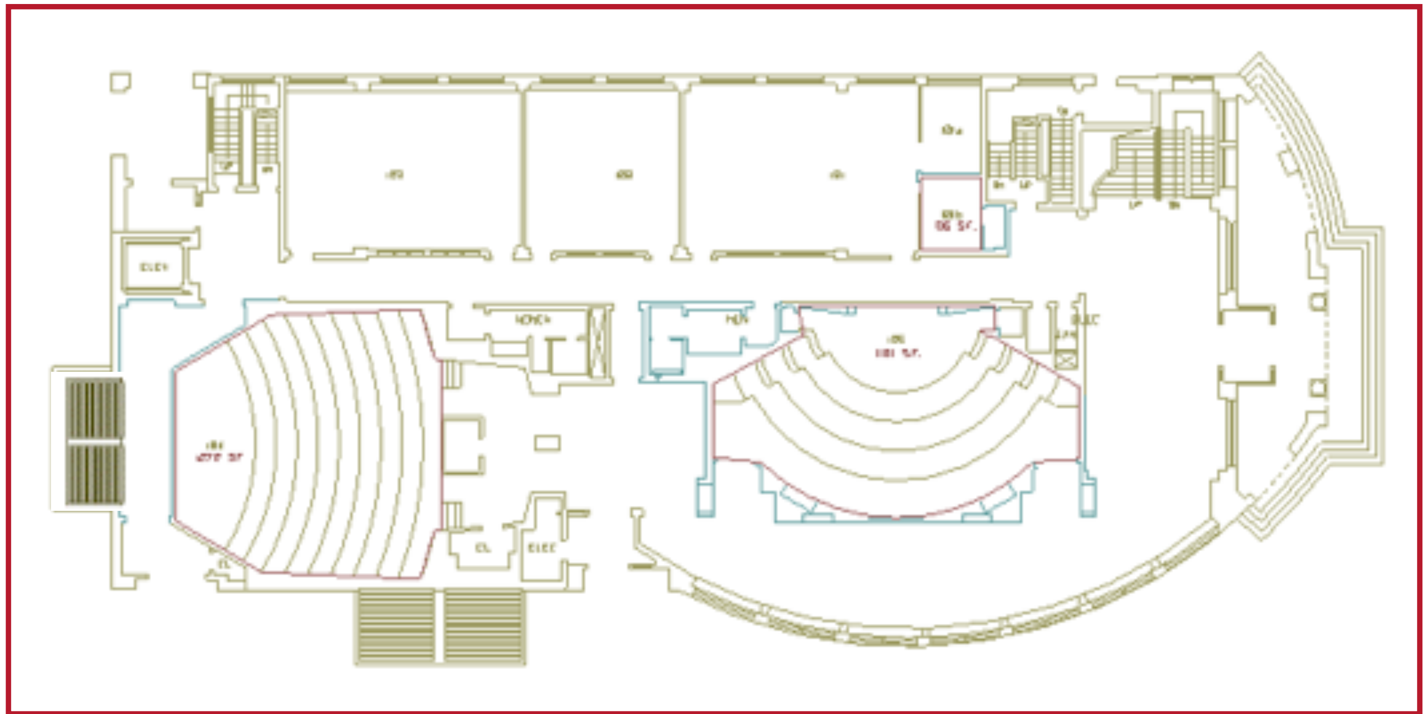
- Translation:
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale:
  - $x' = x * s_x$
  - $y' = y * s_y$
- Rotation:
  - $x' = x*\cos\Theta - y*\sin\Theta$
  - $y' = x*\sin\Theta + y*\cos\Theta$



$$x' = ((x*s_x)*\cos\Theta - (y*s_y)*\sin\Theta) + t_x$$
$$y' = ((x*s_x)*\sin\Theta + (y*s_y)*\cos\Theta) + t_y$$

Basic 2D Transformations

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



$$x' = ((x * s_x) * \cos\Theta - (y * s_y) * \sin\Theta) + t_x$$
$$y' = ((x * s_x) * \sin\Theta + (y * s_y) * \cos\Theta) + t_y$$

Overview

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

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

- 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 Scale around (0,0)?

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

2D Rotate around (0,0)?

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

2D Rotate around (0,0)?

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

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

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

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 Mirror over Y axis?

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

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 Mirror over Y axis?

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

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

2x2 Matrices

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

2D Scale around (0,0)?

$$x' = sx * x$$
$$y' = sy * y$$

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

Like scale with negative scale values

2D Mirror over Y axis?

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

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

2x2 Matrices

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

2D Translation?

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, and
  - Rotation
- Properties of linear transformations:
  - Satisfies:
  - Origin maps to origin
  - Lines map to lines
  - Parallel lines remain parallel
  - Closed under composition

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

$$T(s_1\mathbf{p}_1 + s_2\mathbf{p}_2) = s_1 T(\mathbf{p}_1) + s_2 T(\mathbf{p}_2)$$

Linear Transformations

- Linear transformations are combinations of …
  - Scale, and
  - Rotation
- Properties of linear transformations:
  - Satisfies:
  - Origin maps to origin
  - Lines map to lines
  - Parallel lines remain parallel
  - Closed under composition

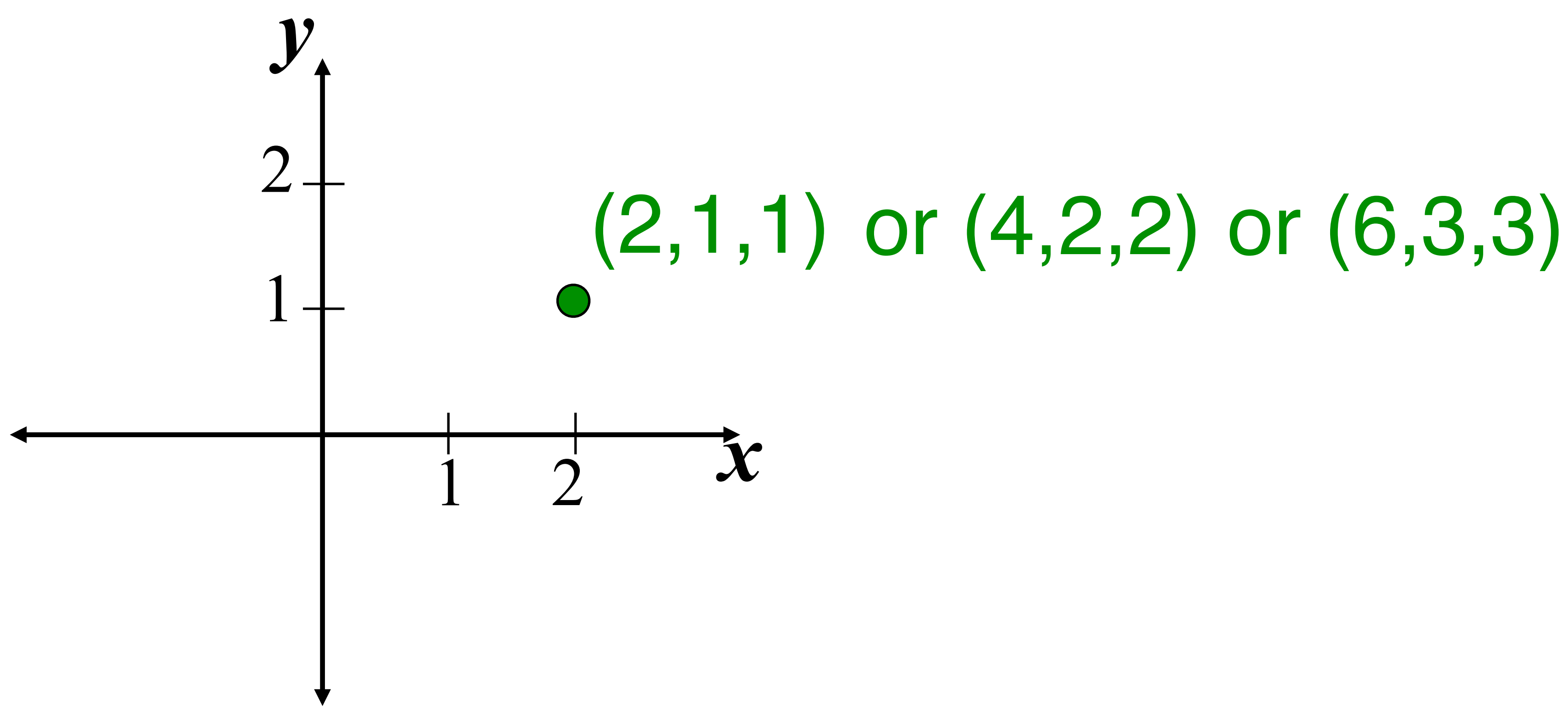
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Translations do not map the origin to the origin

$$T(s_1\mathbf{p}_1 + s_2\mathbf{p}_2) = s_1T(\mathbf{p}_1) + s_2T(\mathbf{p}_2)$$

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



Convenient coordinate system to represent many useful transformations

2D Translation

- 2D translation represented by a 3x3 matrix
  - Point represented with homogeneous coordinates

$$x' = x + tx * w$$
$$y' = y + ty * w$$
$$w' = w$$

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

2D Translation

- 2D translation represented by a 3x3 matrix
  - Point represented with homogeneous coordinates

$$x' = x + tx$$
$$y' = y + ty$$

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

Basic 2D Transformations

- Basic 2D transformations as 3x3 matrices

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

Translate

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

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

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
  - 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
  - Closed under composition

Overview

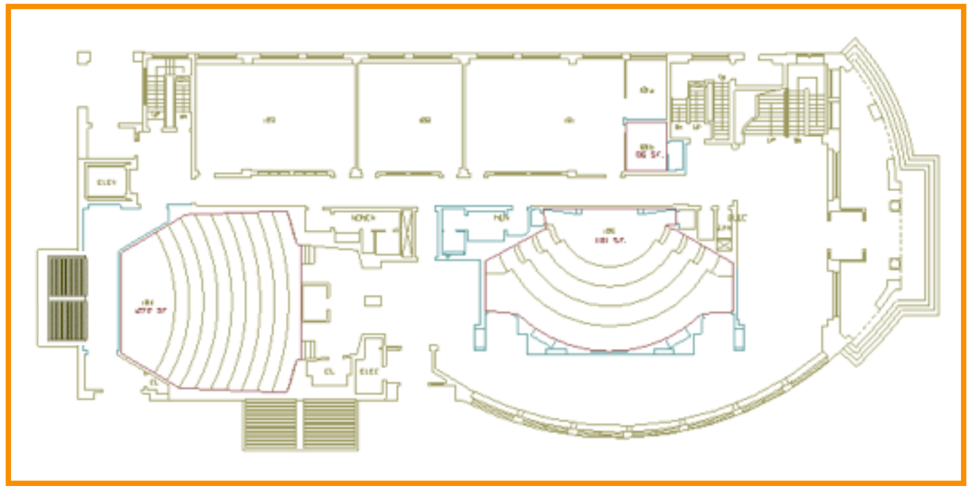
- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- 3D Transformations
  - Basic 3D transformations
  - Same as 2D

Matrix Composition

- Transformations can be combined by matrix multiplication

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

$$\mathbf{p}' \quad = \quad T(tx,ty) \qquad R(\Theta) \qquad S(sx,sy) \qquad \mathbf{p}$$

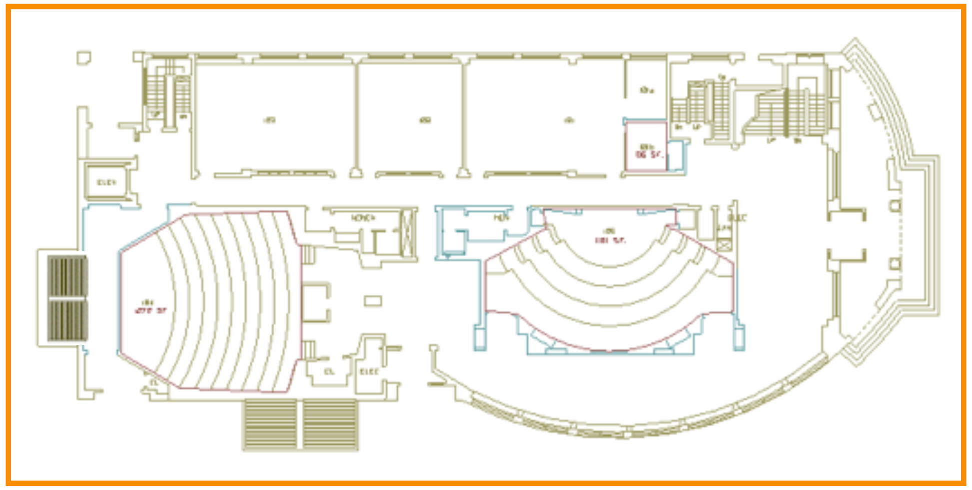


Matrix Composition

- Matrices are a convenient and efficient way to represent a sequence of transformations
  - General purpose representation
  - Hardware matrix multiply
  - Efficiency with pre-multiplication
    - Matrix multiplication is associative

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

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

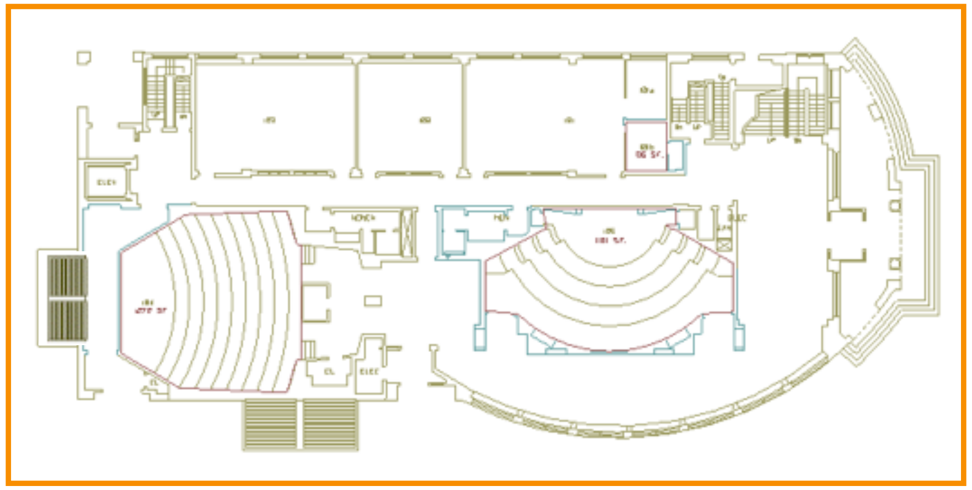


Matrix Composition

- Be aware: order of transformations matters
  - » Matrix multiplication is not commutative

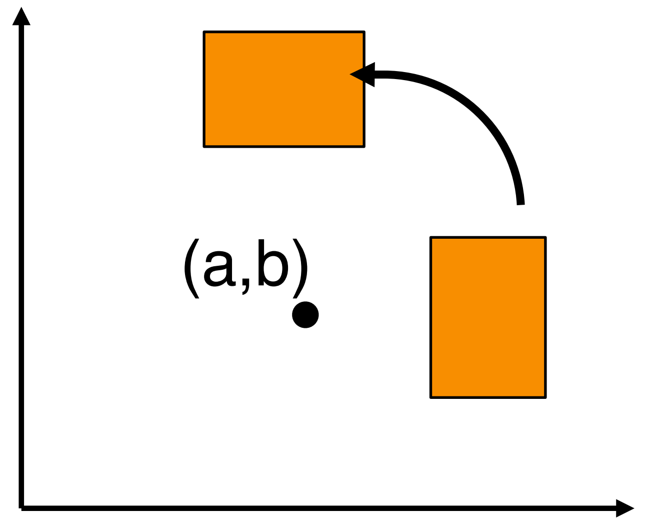
$$\mathbf{p}' = T * R * S * \mathbf{p}$$

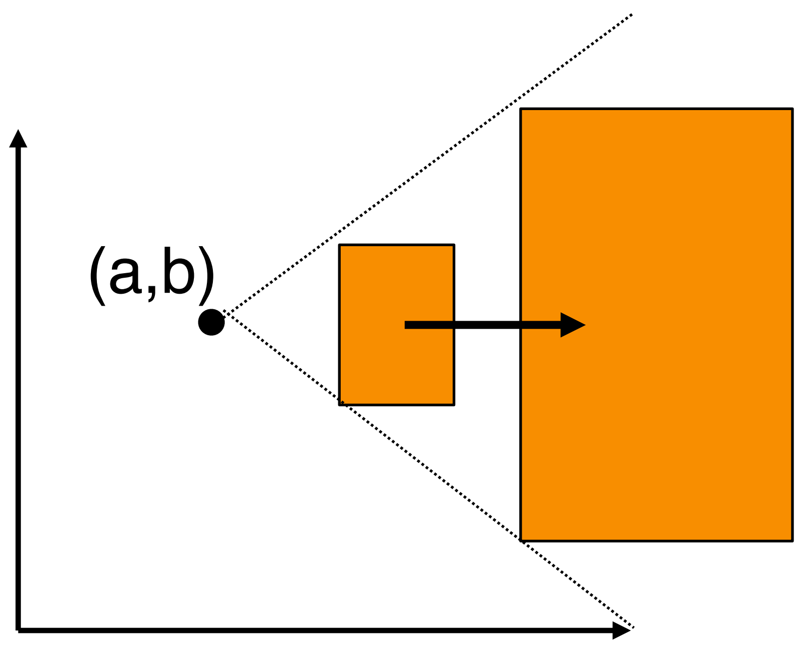
"Global" ⟷ "Local"



Matrix Composition

- Rotate by Θ around arbitrary point (a,b)

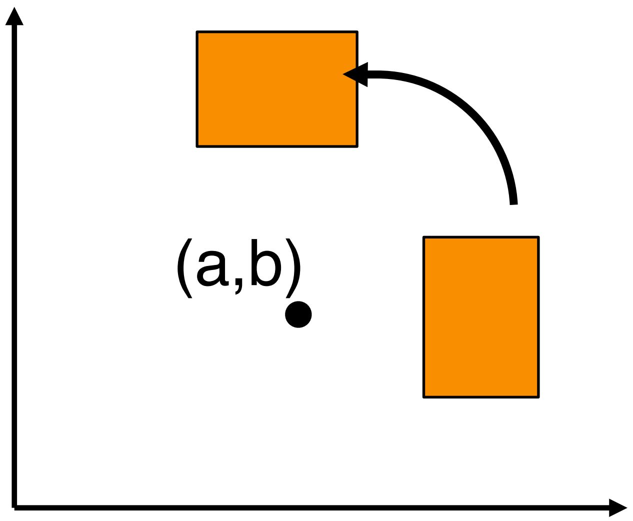




Matrix Composition

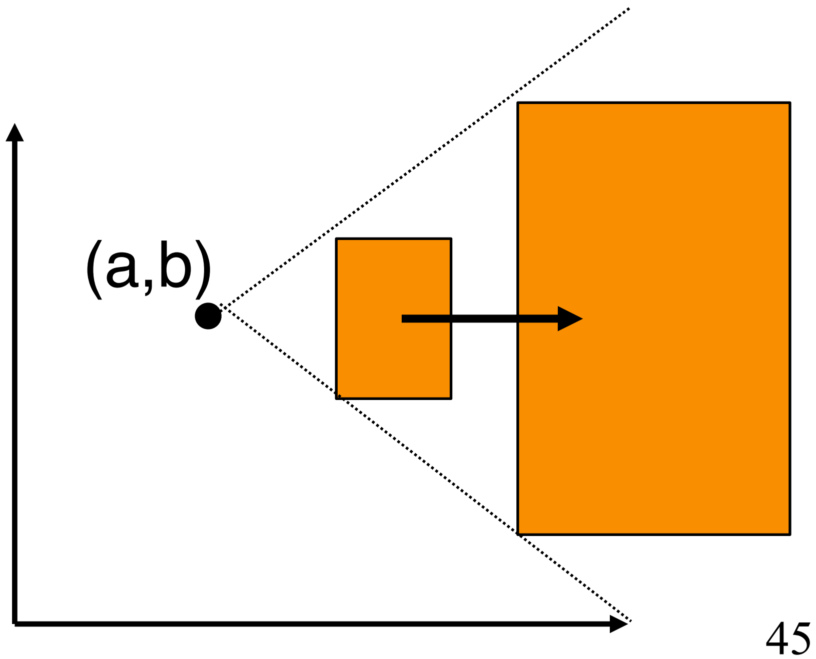
- Rotate by Θ around arbitrary point (a,b)
  - M=T(a,b) * R(Θ) * T(-a,-b)

The trick:
First, translate (a,b) to the origin.
Next, do the rotation about origin.
Finally, translate back.



- Scale by sx,sy around arbitrary point (a,b)
  - M=T(a,b) * S(sx,sy) * T(-a,-b)

(Use the same trick.)



Overview

- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- 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

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

Scale

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

Translation

Basic 3D Transformations

Pitch-Roll-Yaw Convention:

- Any rotation can be expressed as the combination of a rotation about the *x*-, the *y*-, and the *z*-axis.

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

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

Basic 3D Transformations

Pitch-Roll-Yaw Convention:

- Any rotation can be expressed as the combination of a rotation about the *x*-, the *y*-, and the *z*-axis.

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 ax

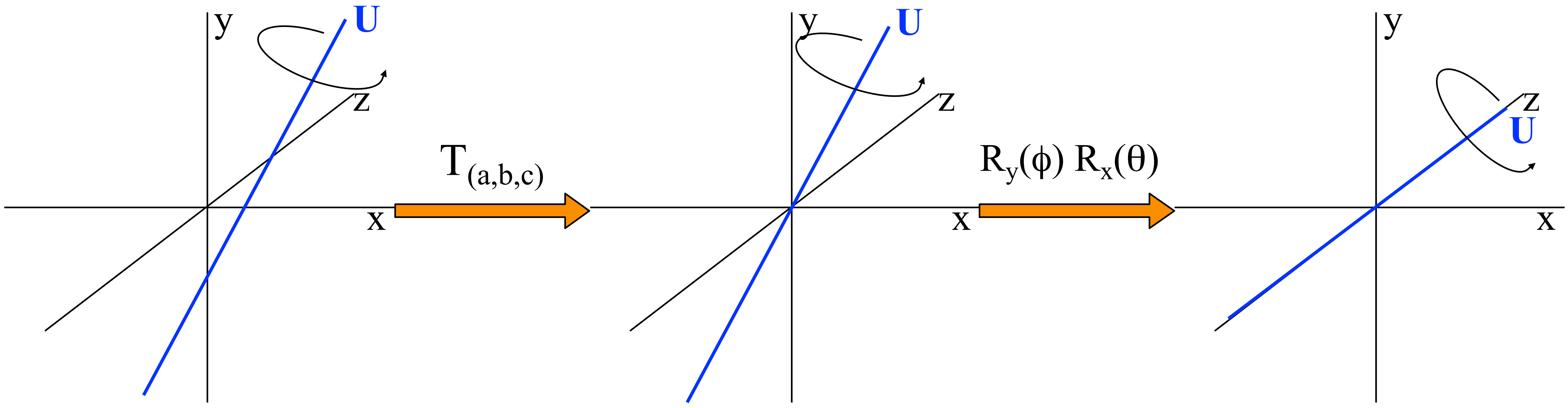
$\begin{bmatrix} x' \\ \vdots \\ w \end{bmatrix} \quad \begin{bmatrix} \cos\Theta & 0 & \sin\Theta & 0 \\ \vdots & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ \vdots \\ w \end{bmatrix}$

How would you rotate around an arbitrary axis U?

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

Rotation By ψ Around Arbitrary Axis U

- Align U with major axis
  - T(a,b,c) = Translate U by (a,b,c) to pass through origin
  - Rx(θ), Ry(ϕ)= Do two separate rotations around two other axes (e.g. x, and y) by θ and ϕ degrees to get it aligned with the third (e.g. z)
- Perform rotation by ψ around the major axis = Rz(ψ)
- Do inverse of original transformation for alignment



Rotation By ψ Around Arbitrary Axis U

- Align U with major axis
  - T(a,b,c) = Translate U by (a,b,c) to pass through origin
  - Rx(θ), Ry(ϕ)= Do two separate rotations around two other axes (e.g. x, and y) by θ and ϕ degrees to get it aligned with the third (e.g. z)
- Perform rotation by ψ around the major axis = Rz(ψ)
- Do inverse of original transformation for alignment

$$p' = \underbrace{\left(R_y(\phi) \times R_x(\theta) \times T_{(a,b,c)}\right)^{-1}}_{} R_z(\psi) \underbrace{\left(R_y(\phi) \times R_x(\theta) \times T_{(a,b,c)}\right)}_{} p$$

Aligning Transformation

Rotation By ψ Around Arbitrary Axis U

- Homogeneous coordinates matrix to rotate an angle ψ around axis U passing through origin:

$$\begin{pmatrix} xx(1-c)+c & xy(1-c)-zs & xz(1-c)+ys & 0 \\ yx(1-c)+zs & yy(1-c)+c & yz(1-c)-xs & 0 \\ xz(1-c)-ys & yz(1-c)+xs & zz(1-c)+c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Here (x, y, z) are components of U (a unit vector)
- $c = \cos(\psi)$
- $s = \sin(\psi)$

- Derivation: this is Rodrigues' rotation formula in matrix form