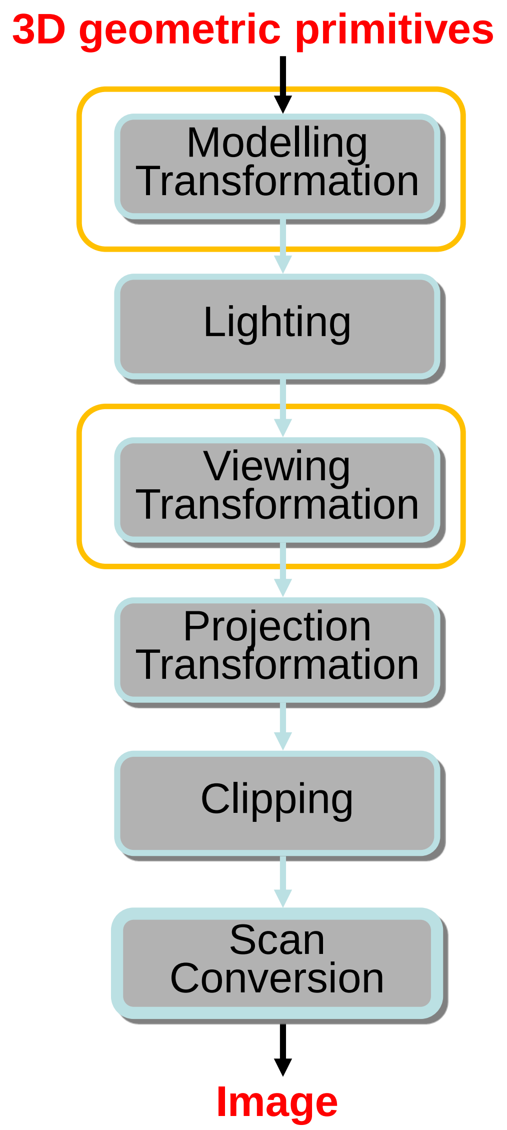
3D Graphics Programming Tools

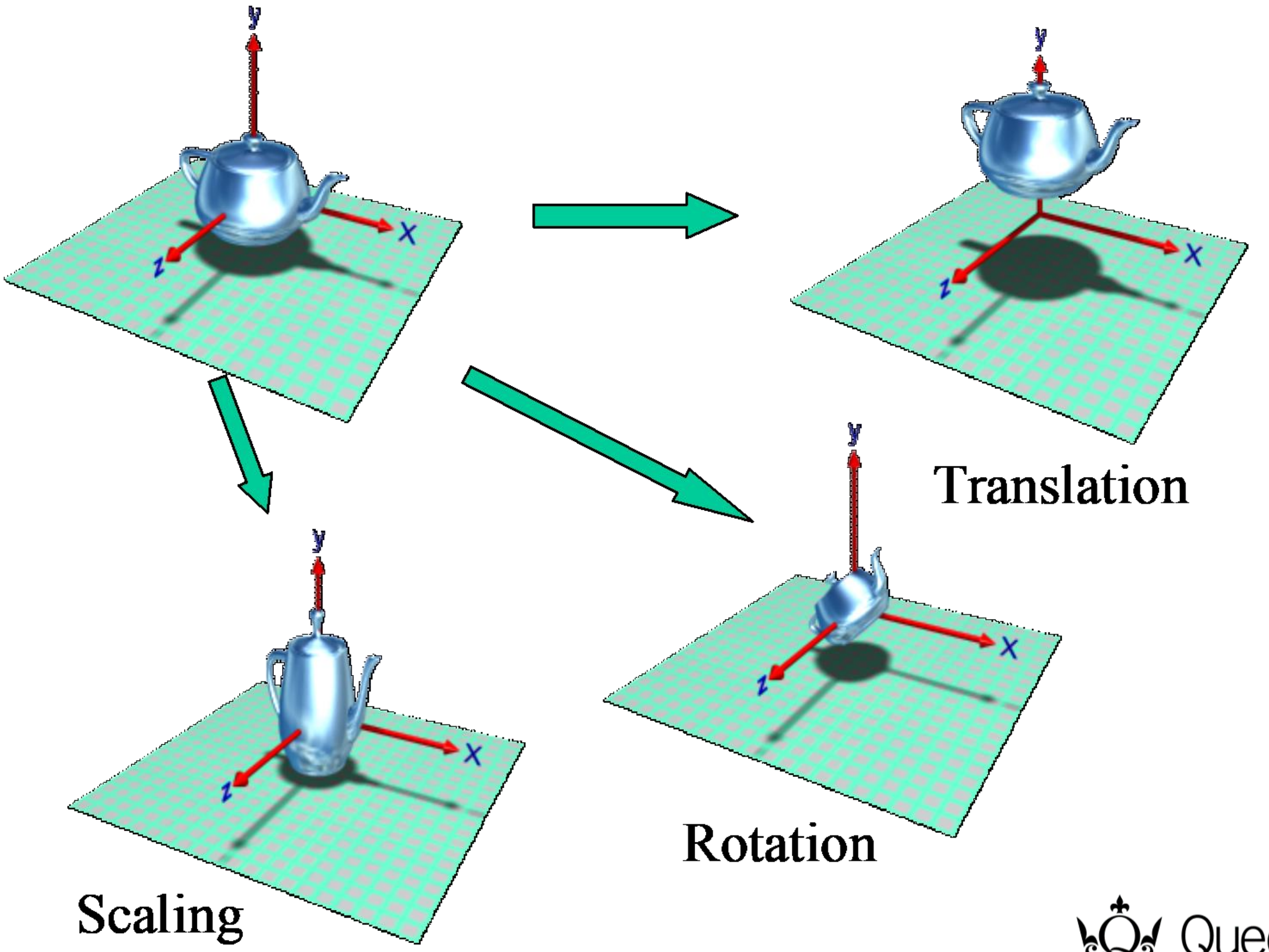
Geometric Transformations

Queen Mary
University of London

3D Computer Graphics Pipeline



Geometric transformations



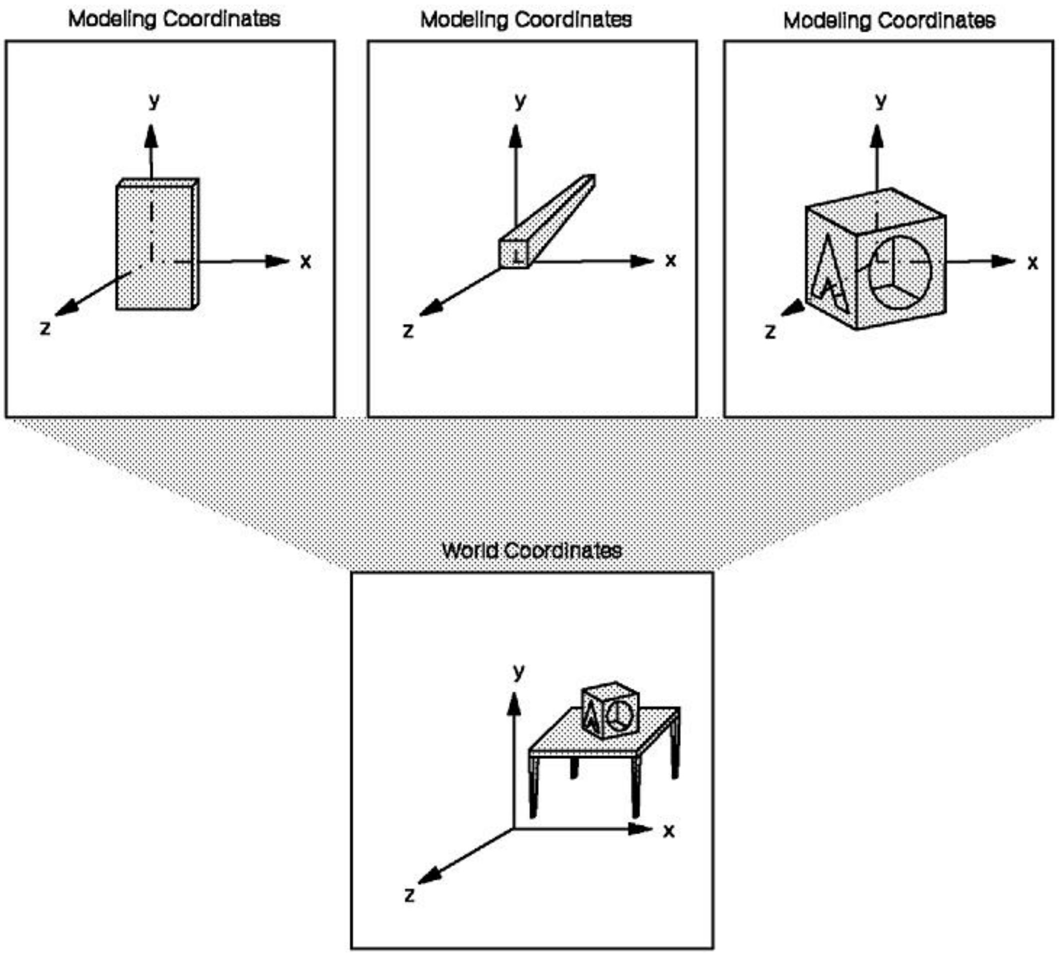
Queen Mary
University of London

Topics

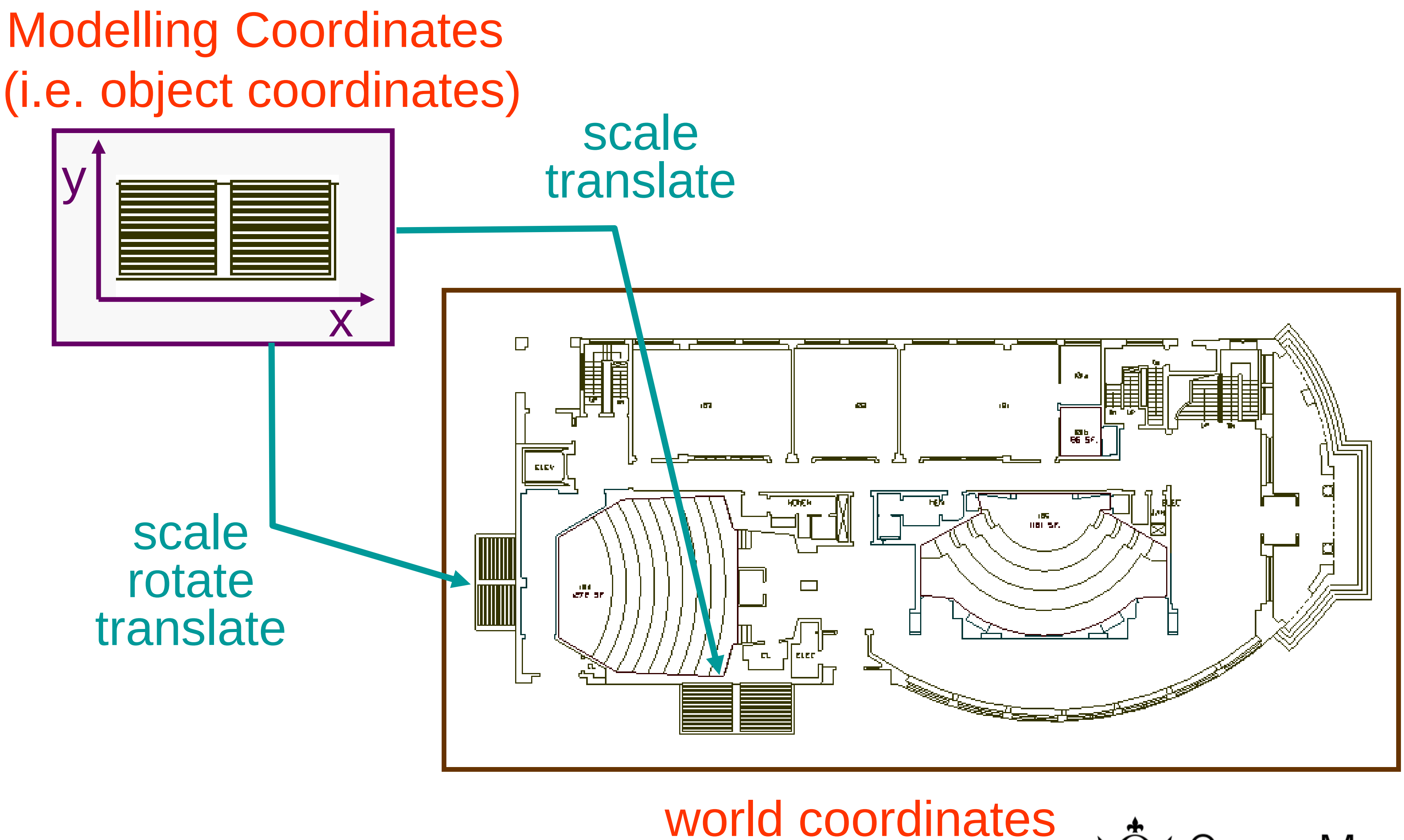
- 2D Transformations
  - Basic 2D transformations
  - Matrix representation
  - Matrix composition
- 3D Transformations
  - Basic 3D transformations

Modelling transformations

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

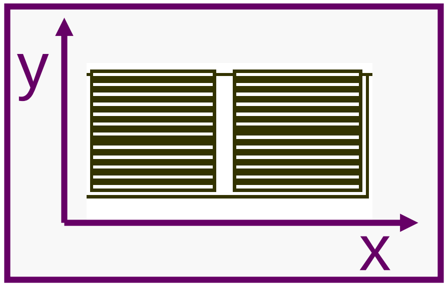


2D modelling transformations

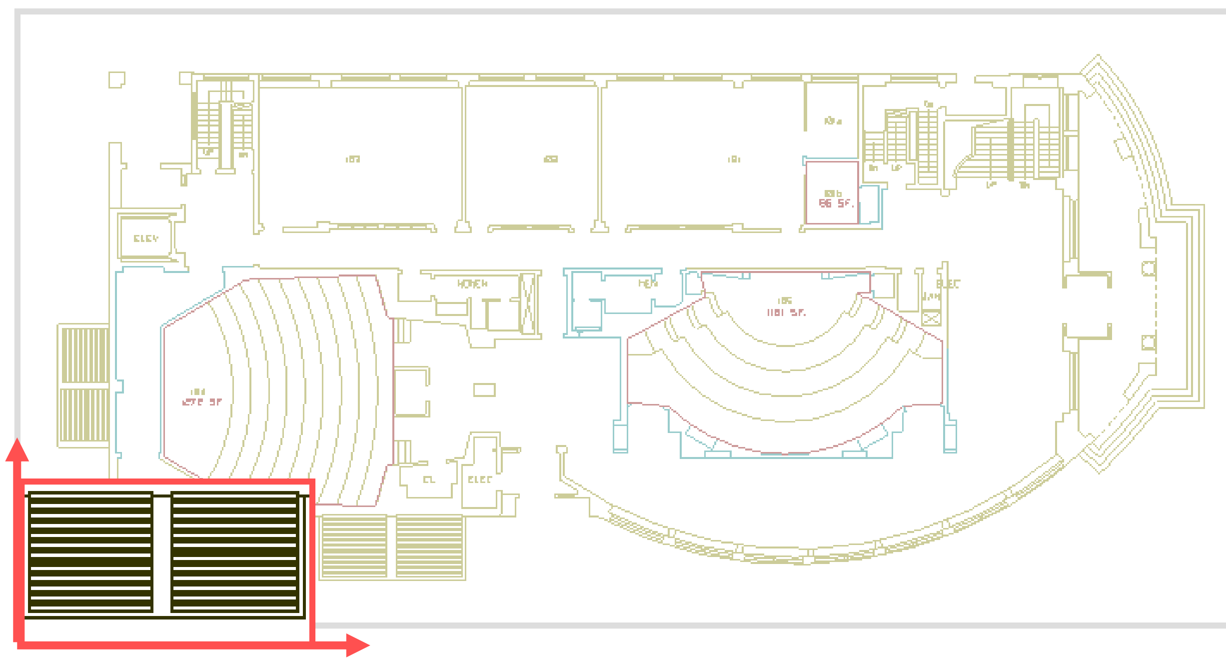


2D modelling transformations

modelling coordinates

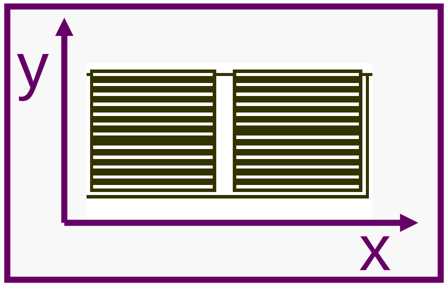


Initial location at (0, 0) with x- and y-axes aligned

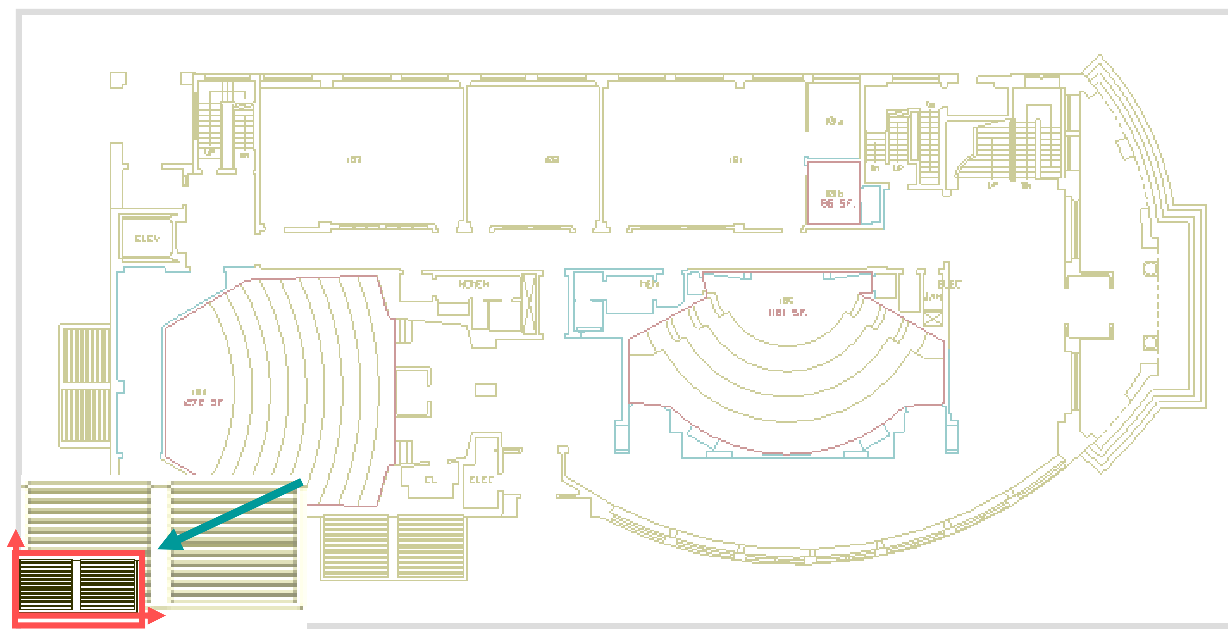


2D modelling transformations

modelling
coordinates

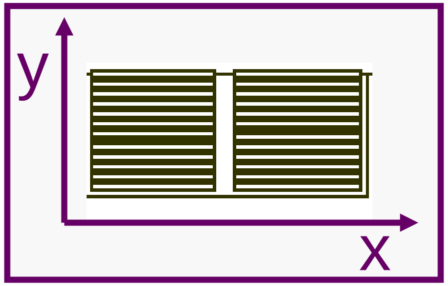


scale .3, .3

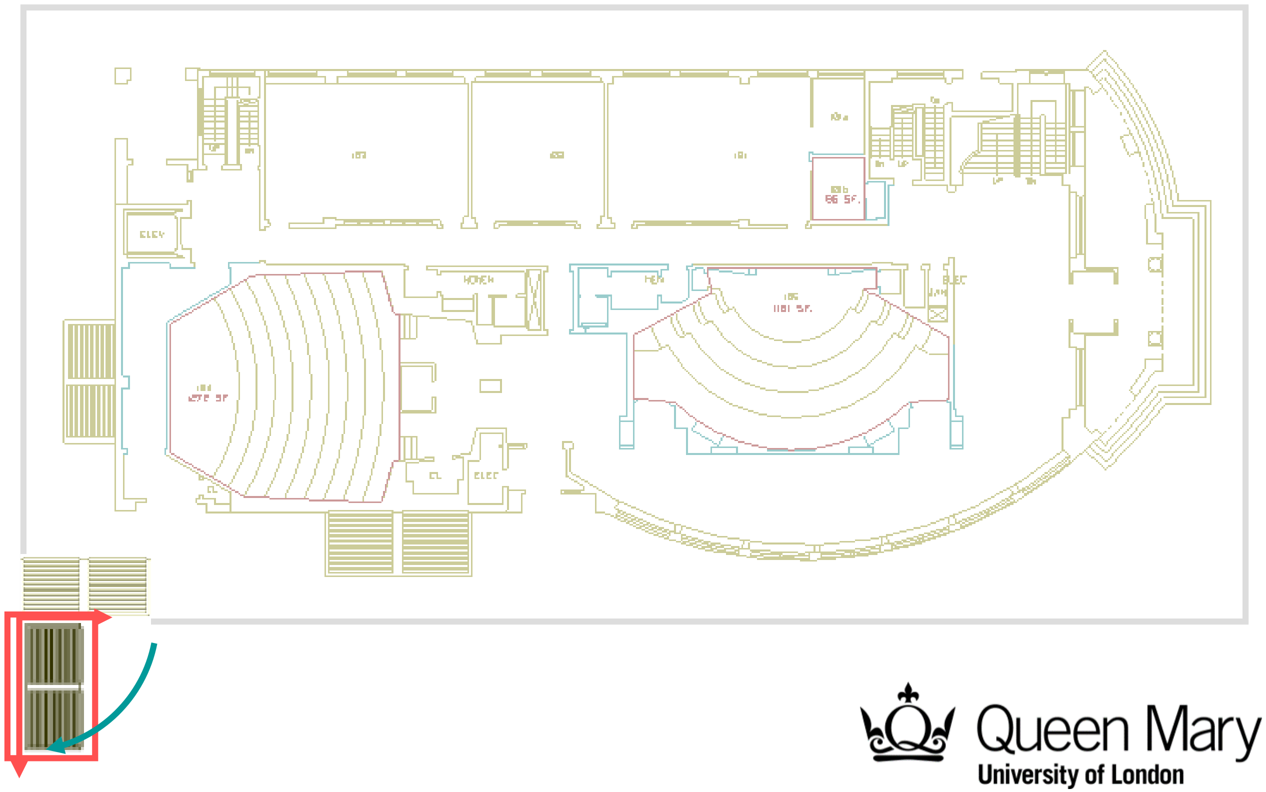


2D modelling transformations

modelling coordinates



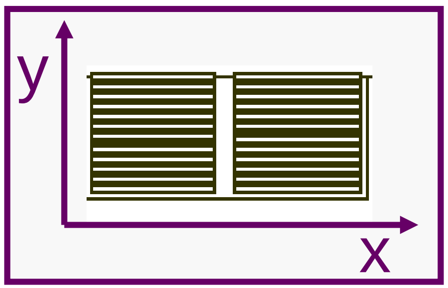
scale .3, .3
rotate -90



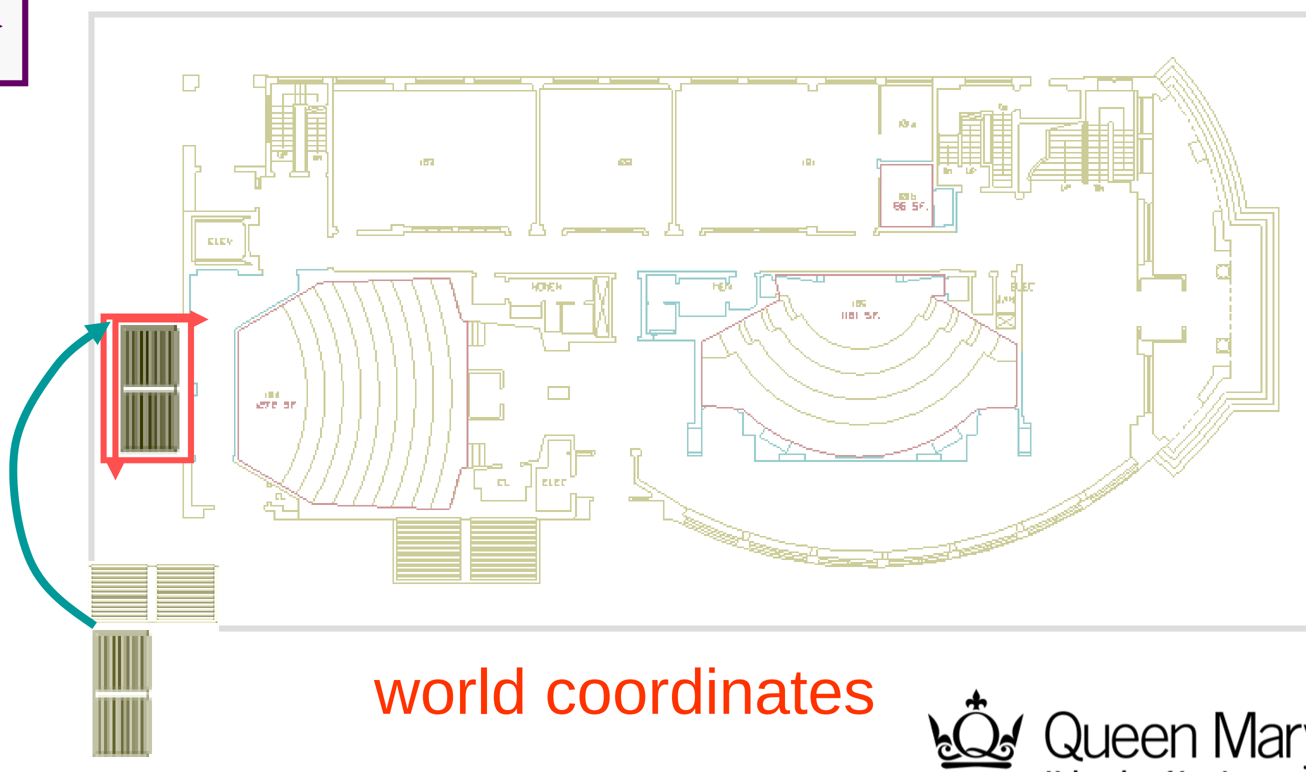
Queen Mary University of London

2D modelling transformations

modelling
coordinates



scale .3, .3
rotate -90
translate 5, 3

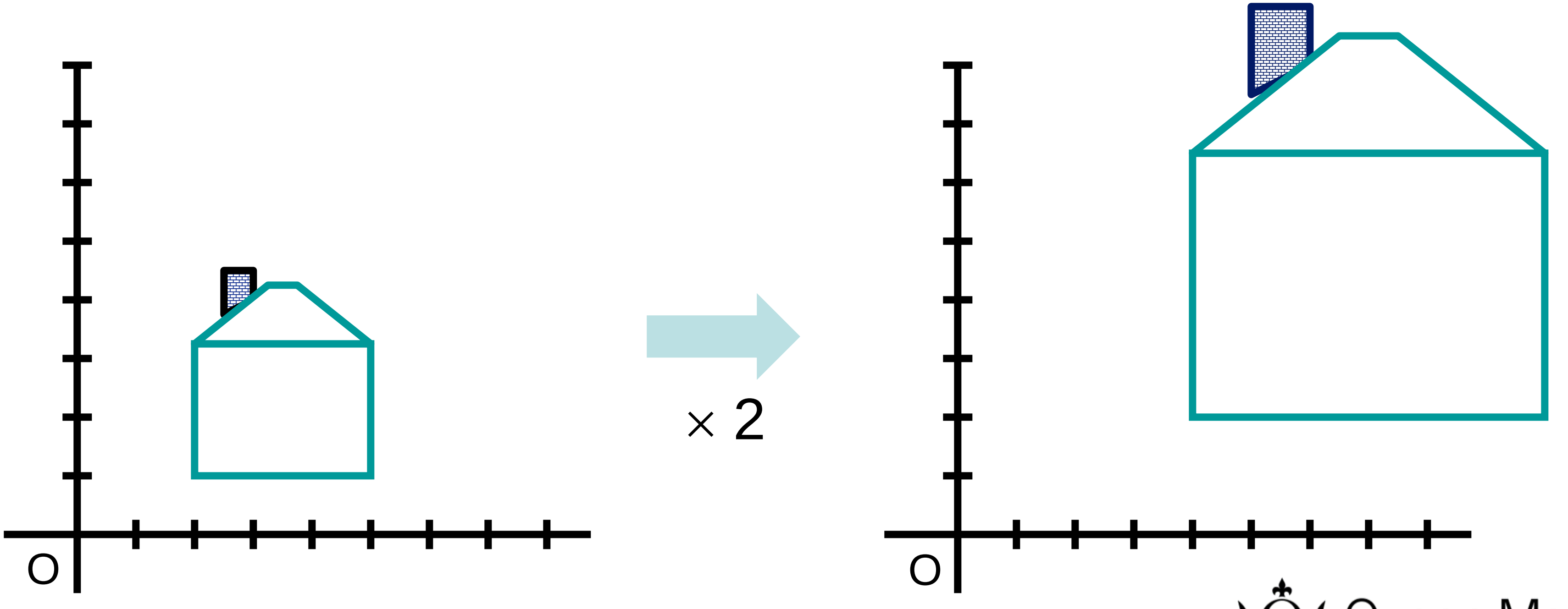


world coordinates

Queen Mary
University of London

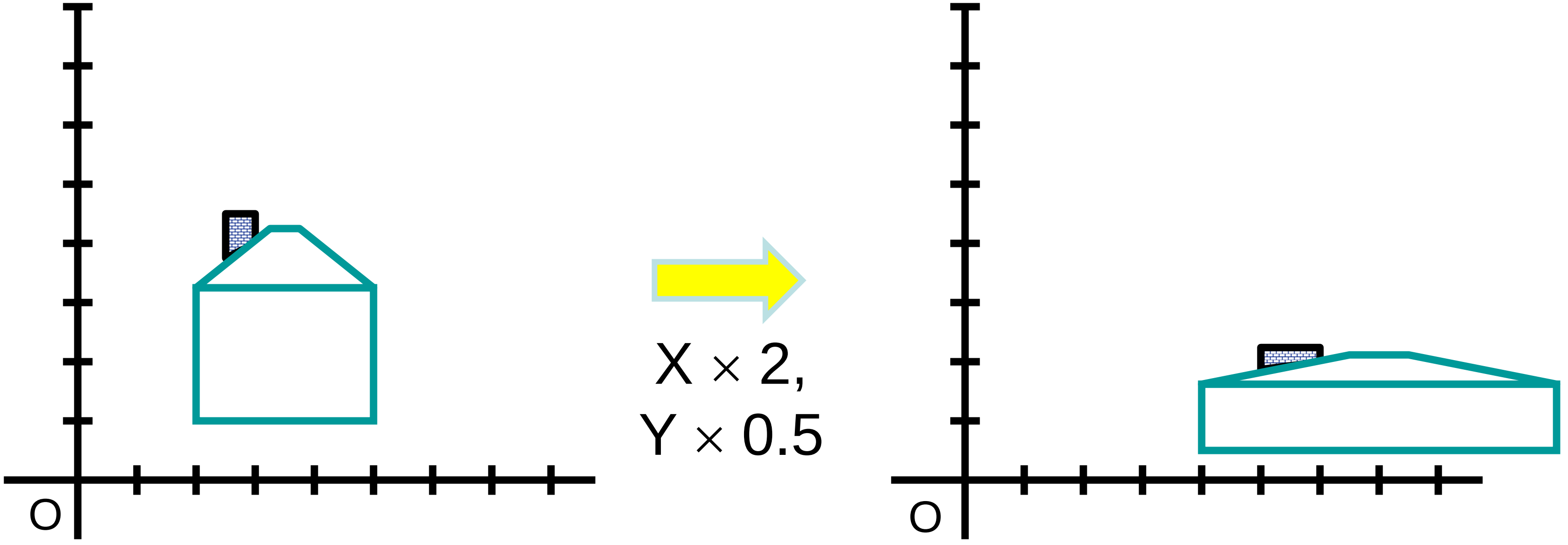
Scaling

- Scaling a coordinate
  - means multiplying each of its components by a scalar
- Uniform scaling
  - means this scalar is the same for all components



Scaling

- Non-uniform scaling
  - different scalars per component



How can we represent this in matrix form?

Scaling

- Scaling operation:

$$\boxed{\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ax \\ by \end{bmatrix}}$$

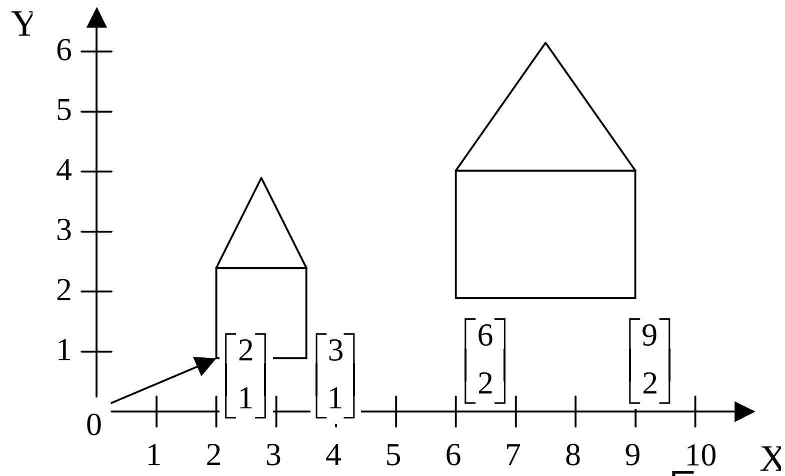
- Or, in matrix form:

$$\boxed{\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\textit{scaling matrix}} \begin{bmatrix} x \\ y \end{bmatrix}}$$

Multiplying a point (or a vector) by a matrix (a transformation) yields a new transformed point (or a new vector)

Scaling

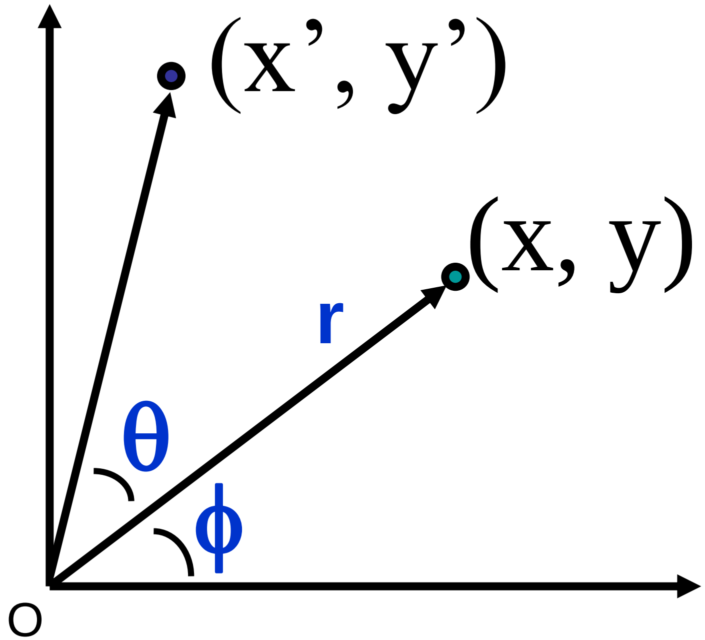
$$s_x = 3$$
$$s_y = 2$$



$$\boldsymbol{v}' = \boldsymbol{S}\boldsymbol{v} \text{ where } \boldsymbol{v} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \boldsymbol{v}' = \begin{bmatrix} x' \\ y' \end{bmatrix}, \ \boldsymbol{S} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \text{ and } \begin{cases} x' = s_x x \\ y' = s_y y \end{cases}$$

- Scaling about the origin
- Negative scaling is reflection
- Scaling needn't be uniform, differential scaling
- Does not preserve lengths
- Does not preserve angles (except uniform scaling)
- Not a rigid body transformation

2D rotation about origin



$x = r\cos(\phi)$

$y = r\sin(\phi)$

$x' = r\cos(\phi + \theta)$

$y' = r\sin(\phi + \theta)$

trigonometric identity…

$x' = r\cos(\phi)\cos(\theta) - r\sin(\phi)\sin(\theta)$

$y' = r\sin(\phi)\cos(\theta) + r\cos(\phi)\sin(\theta)$

substitute…

$x' = \mathbf{x}\cos(\theta) - \mathbf{y}\sin(\theta)$

$y' = \mathbf{x}\sin(\theta) + \mathbf{y}\cos(\theta)$

Queen Mary
University of London

2D rotation

- Or, in matrix form:

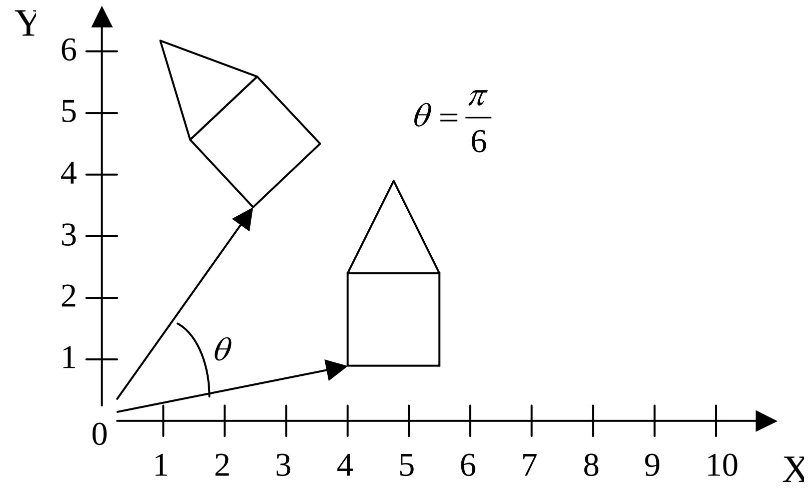
$$\boxed{\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}}$$

- Even though sin(θ) and cos(θ) are nonlinear functions of θ,
  - x' is a linear combination of x and y
  - y' is a linear combination of x and y

Queen Mary
University of London

2D rotation

- **Rotation about the origin**
- **Preserves lengths and angles**
- **Rigid body transformation**

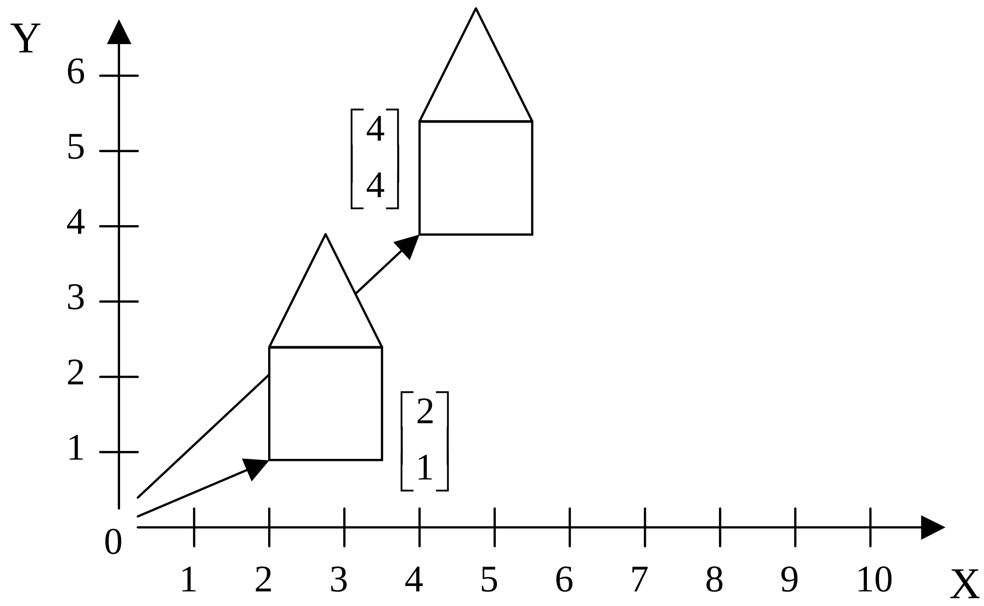


$$v' = Rv, \text{ where } v = \begin{bmatrix} x \\ y \end{bmatrix}, v' = \begin{bmatrix} x' \\ y' \end{bmatrix}, R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$\begin{cases} x' = x\cos\theta - y\sin\theta \\ y' = x\sin\theta + y\cos\theta \end{cases}$$

Queen Mary
University of London

2D Translation



dx = 2
dy = 3

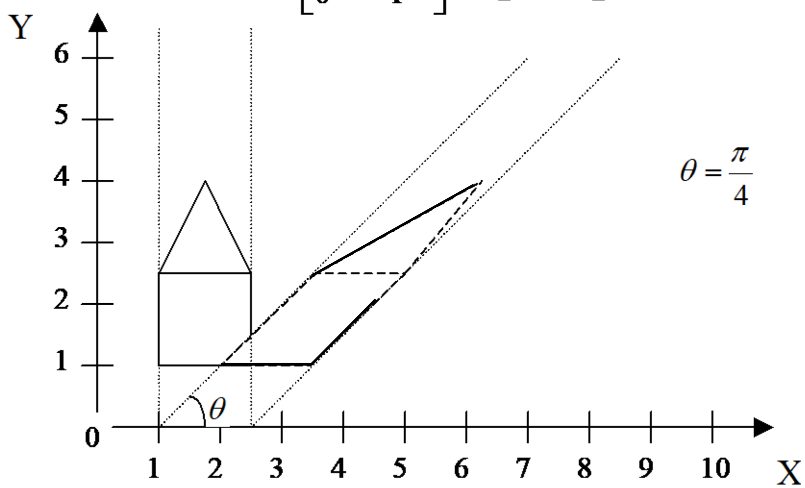
$$\boldsymbol{v}' = \boldsymbol{v} + \boldsymbol{t}, \text{ where } \boldsymbol{v} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \boldsymbol{v}' = \begin{bmatrix} x' \\ y' \end{bmatrix}, \quad \boldsymbol{t} = \begin{bmatrix} dx \\ dy \end{bmatrix}, \text{ and } \begin{cases} x' = x + dx \\ y' = y + dy \end{cases}$$

- **Preserves lengths (isometric)**
- **Preserves angles (conformal)**
- **Rigid body transformation**

Queen Mary
University of London

2D Shearing

$$Shear_\theta = \begin{bmatrix} 1 & \frac{1}{\tan\theta} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & dx \\ 0 & 1 \end{bmatrix}$$



$\theta = \frac{\pi}{4}$

Not rigid body transformation

$$\begin{cases} x' = x + sh_x * y \\ y' = y \end{cases}$$

$$\begin{cases} x' = x \\ y' = sh_y * x + y \end{cases}$$

- **Shear transformations are also affine transformations**

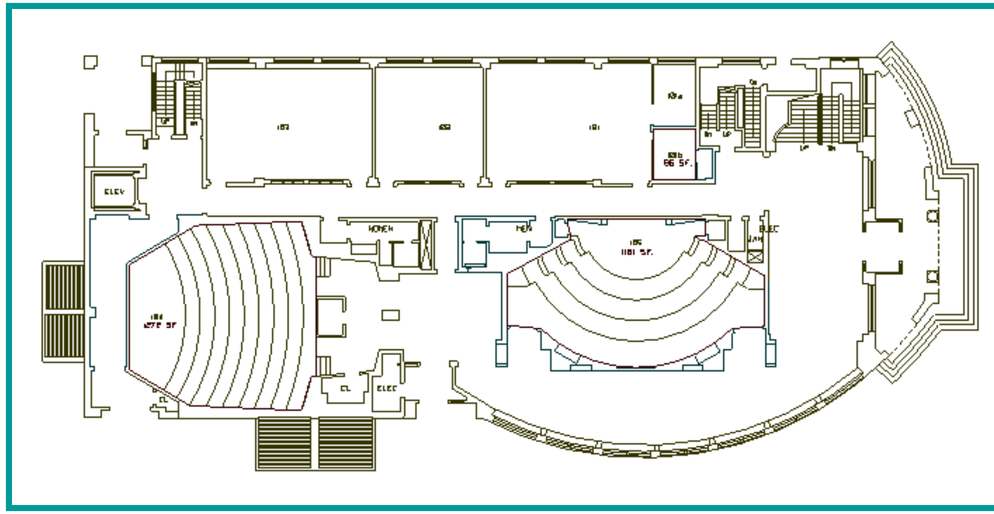
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & sh_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$ **(shear along x axis by using y)**

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ sh_y & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$ **(shear along y axis by using x)**

Queen Mary
University of London

Basic 2D transformations

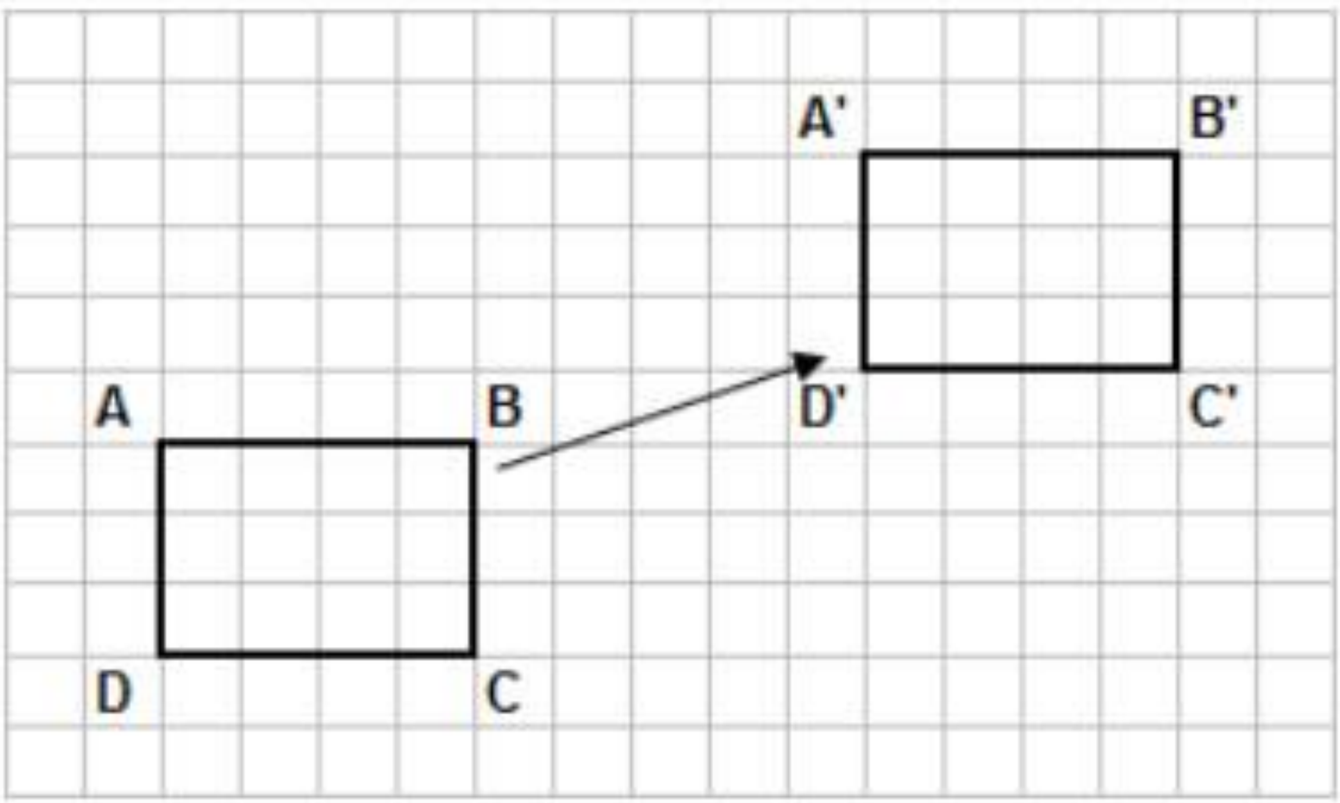
- Translation
  - $x' = x + t_x$
  - $y' = y + t_y$
- Scale
  - $x' = x * s_x$
  - $y' = y * s_y$
- Shear
  - $x' = x + h_x * y$
  - $y' = y + h_y * x$
- Rotation
  - $x' = x * \cos\Theta - y * \sin\Theta$
  - $y' = x * \sin\Theta + y * \cos\Theta$



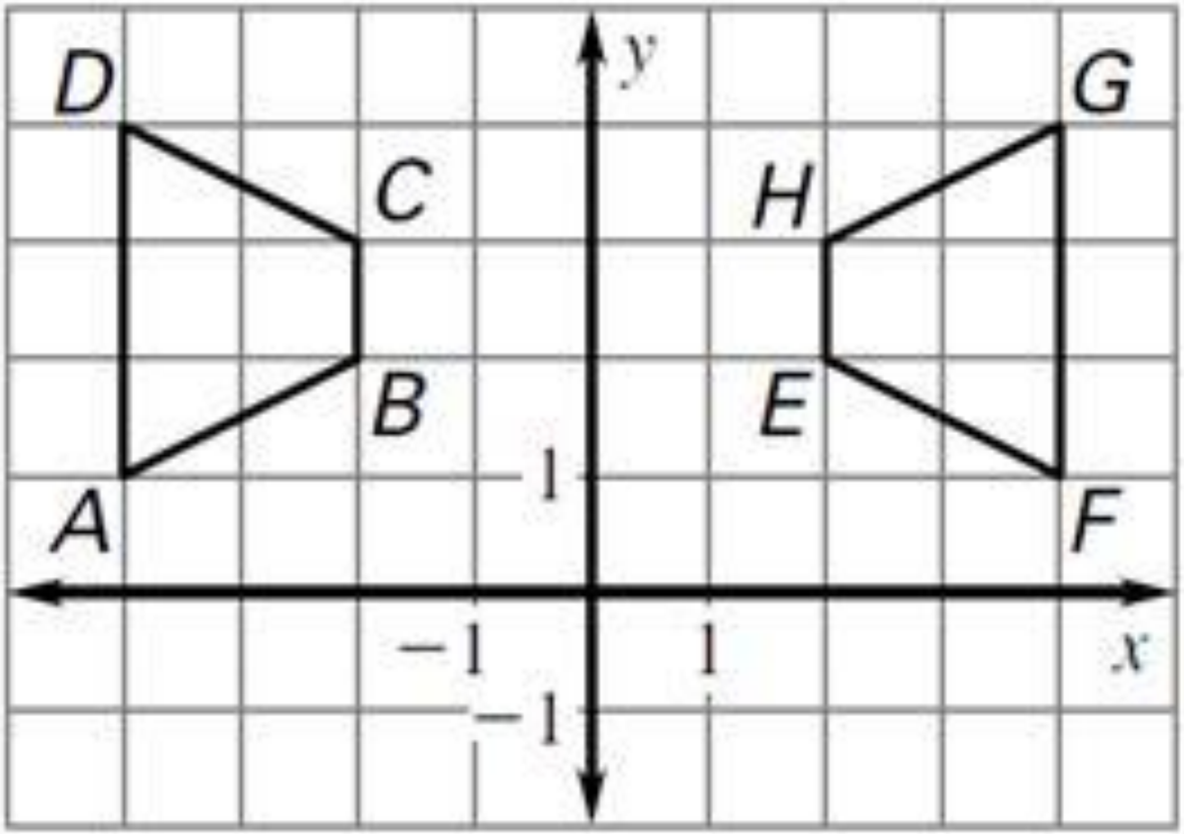
Transformations can be combined
(with simple algebra)

Queen Mary
University of London

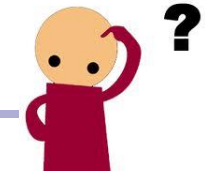
Name the transformation!

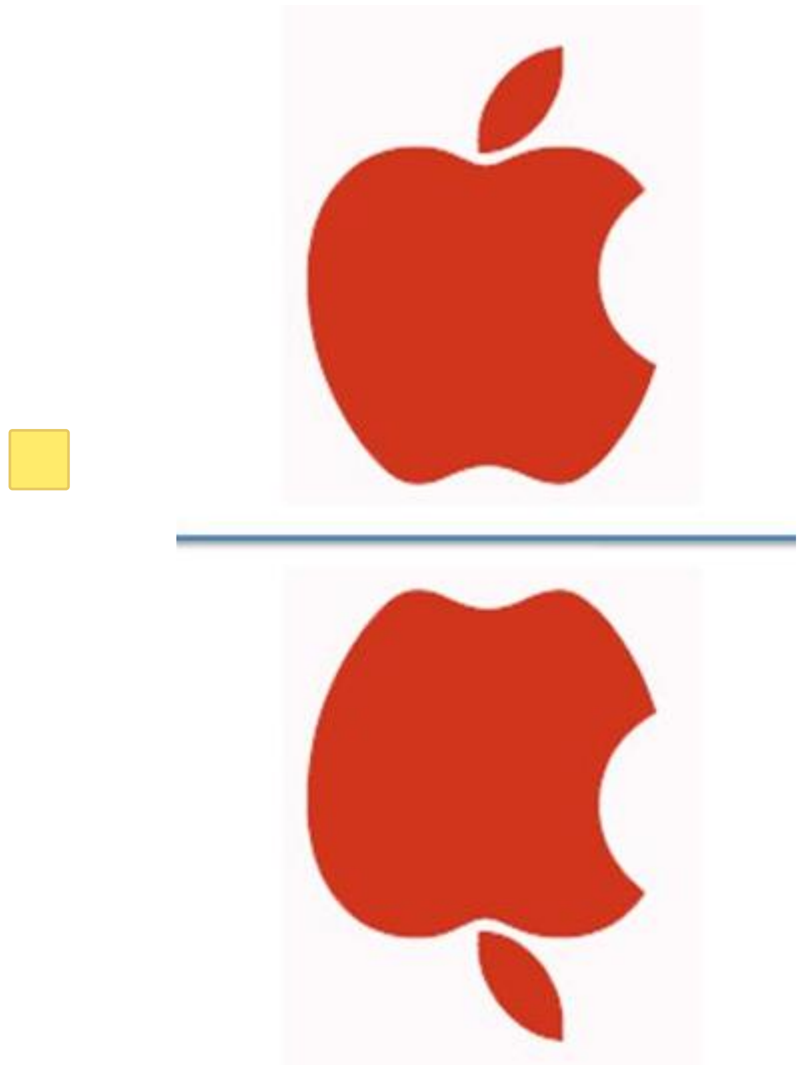


Name the transformation!



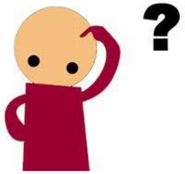
Name the transformation!

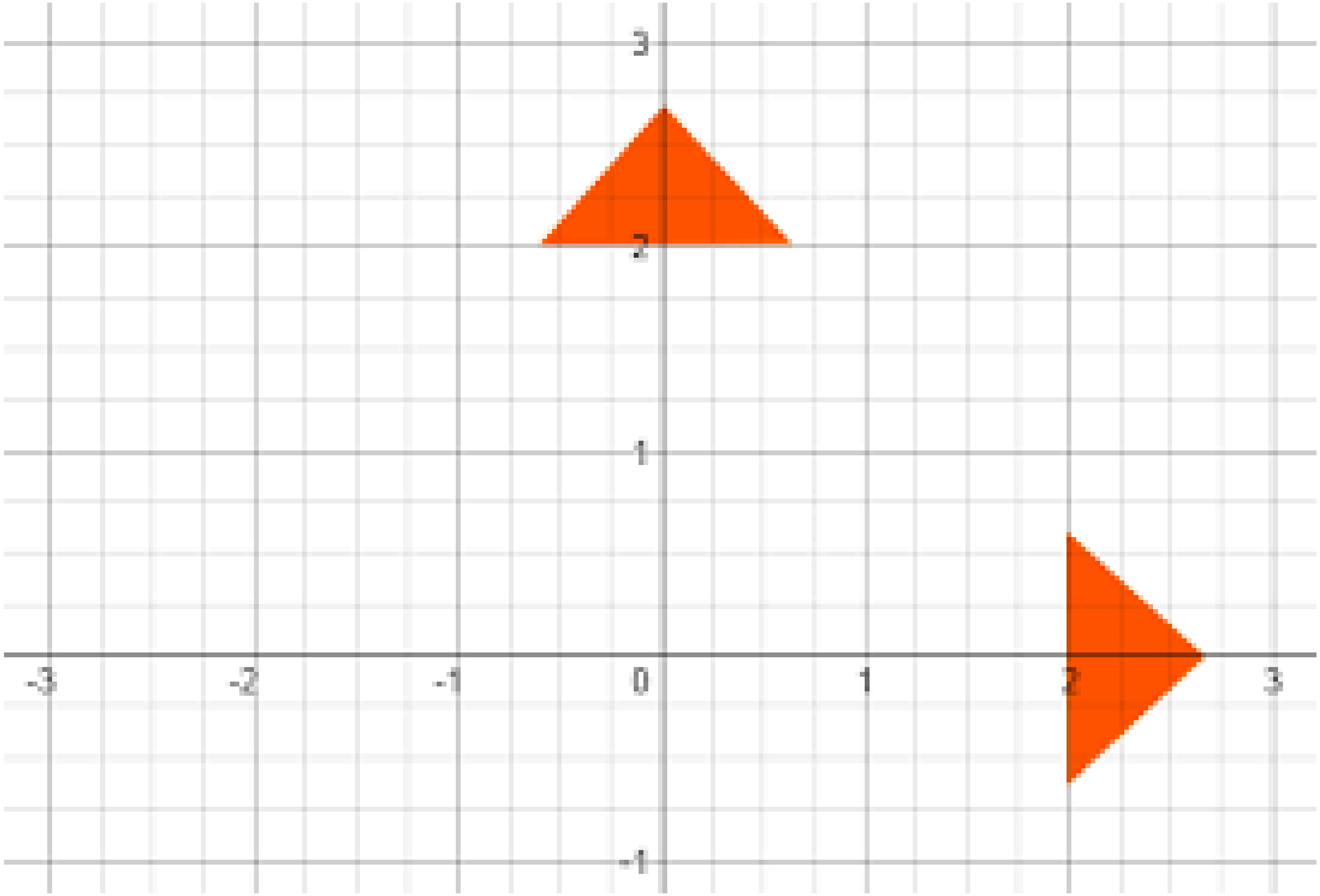




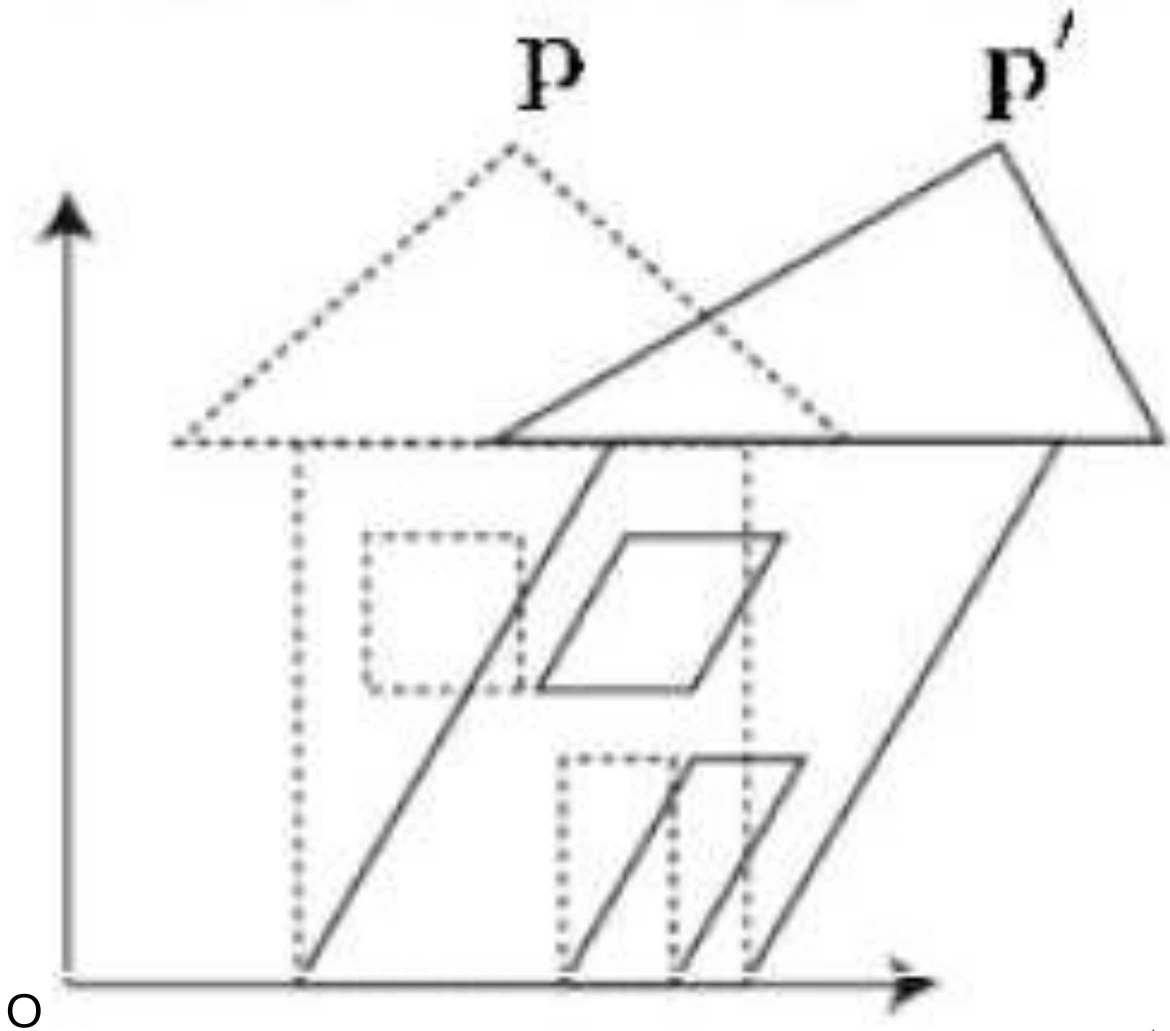
Queen Mary
University of London

Name the transformation!

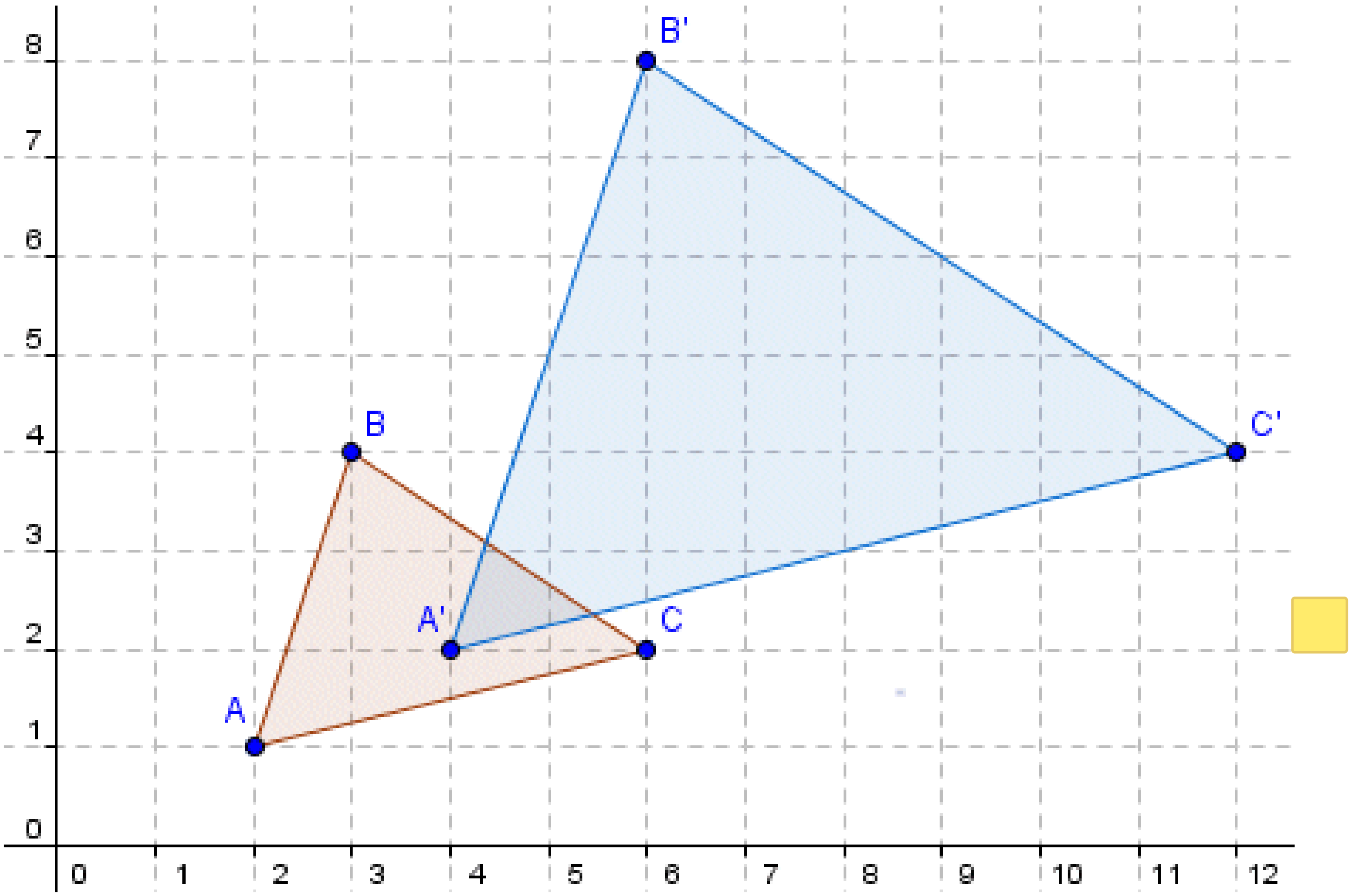




Name the transformation!

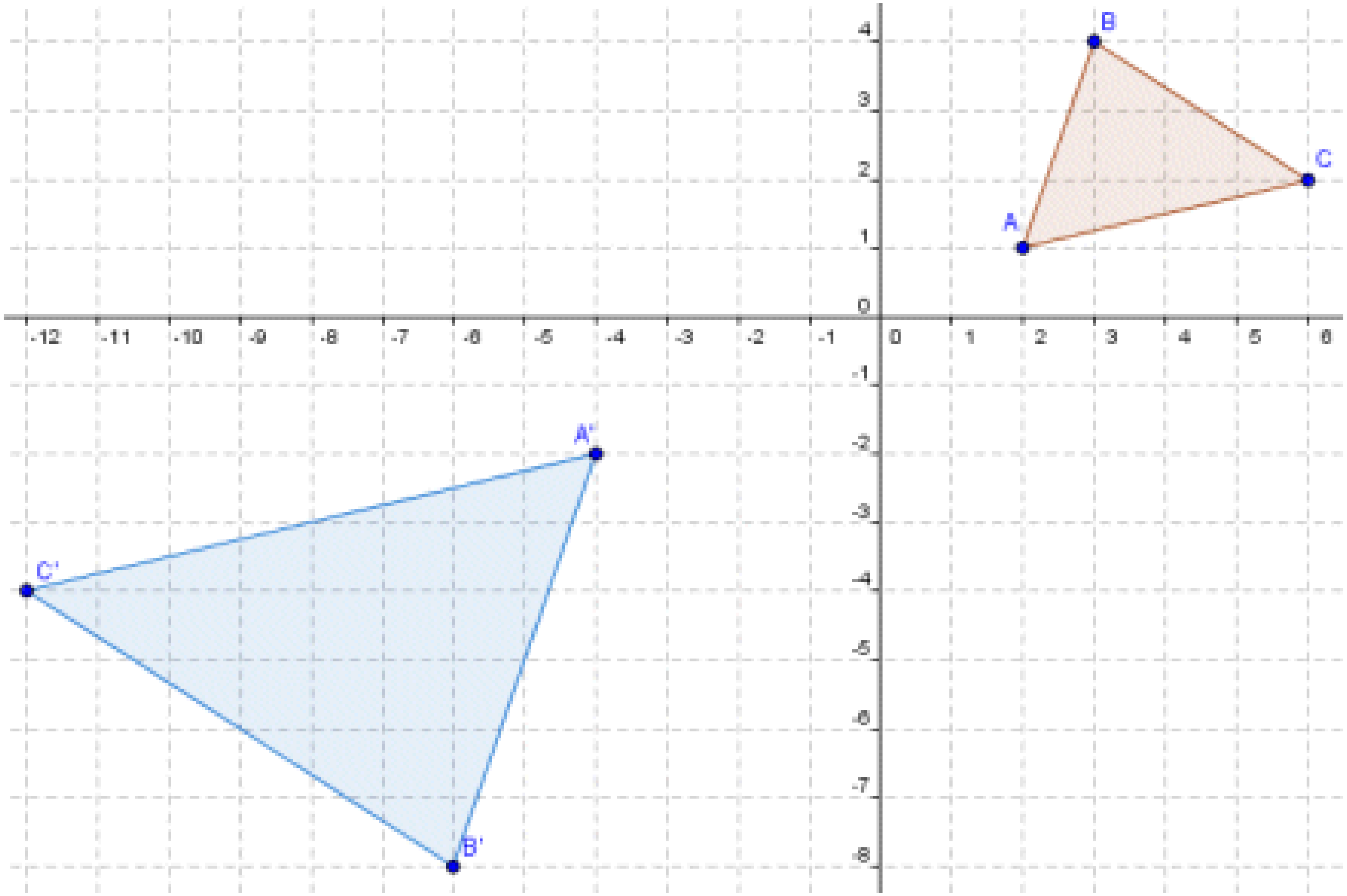


Name the transformation!

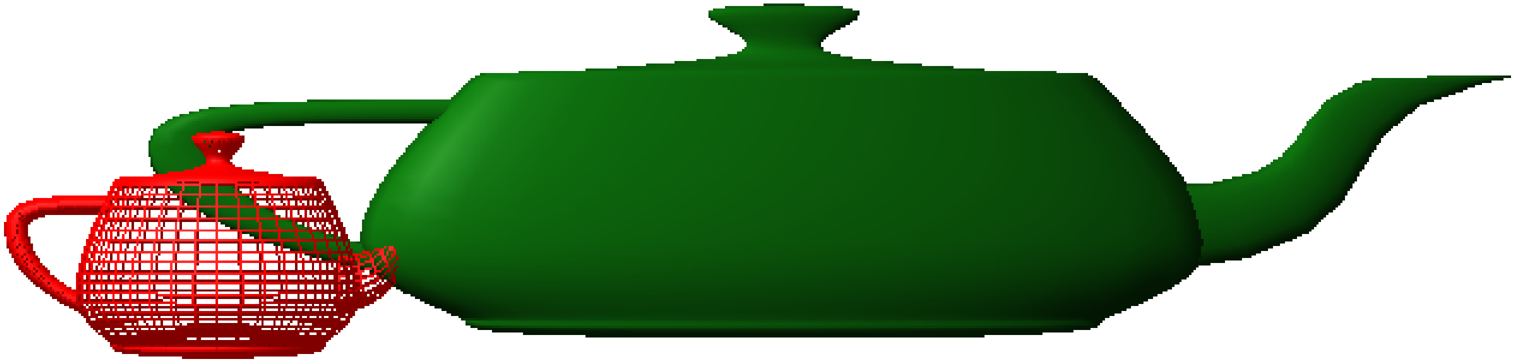


Queen Mary
University of London

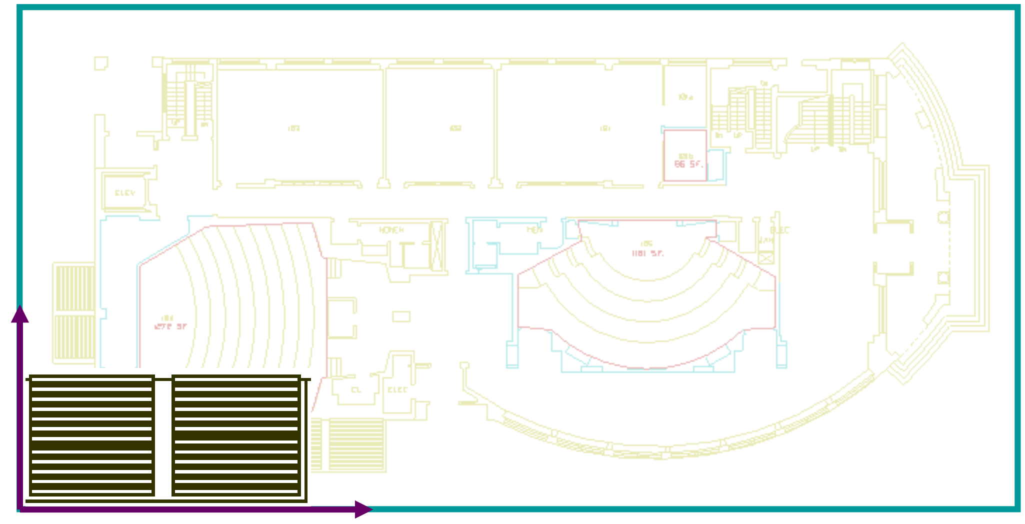
Name the transformation!



Name the transformation!

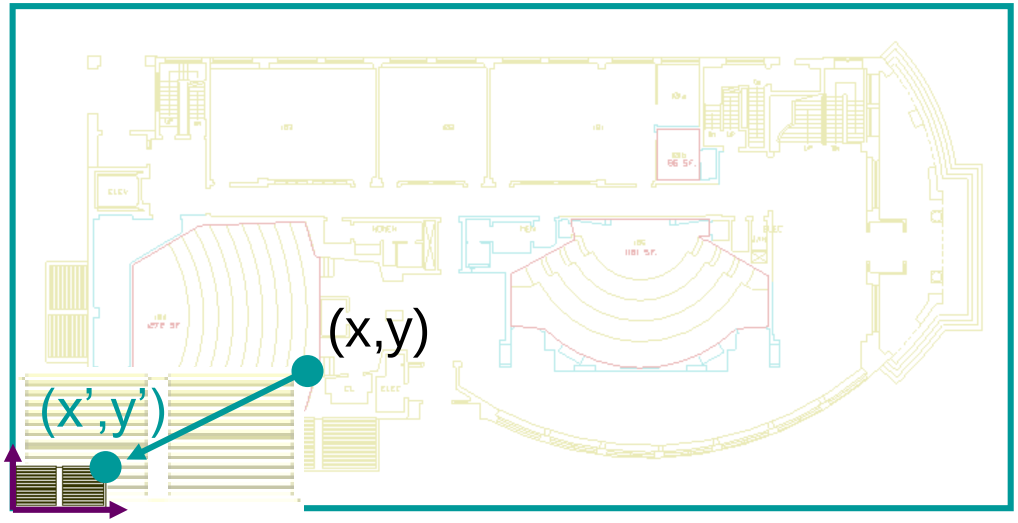


Basic 2D transformations (combination)



Basic 2D transformations (combination)

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



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

Queen Mary
University of London

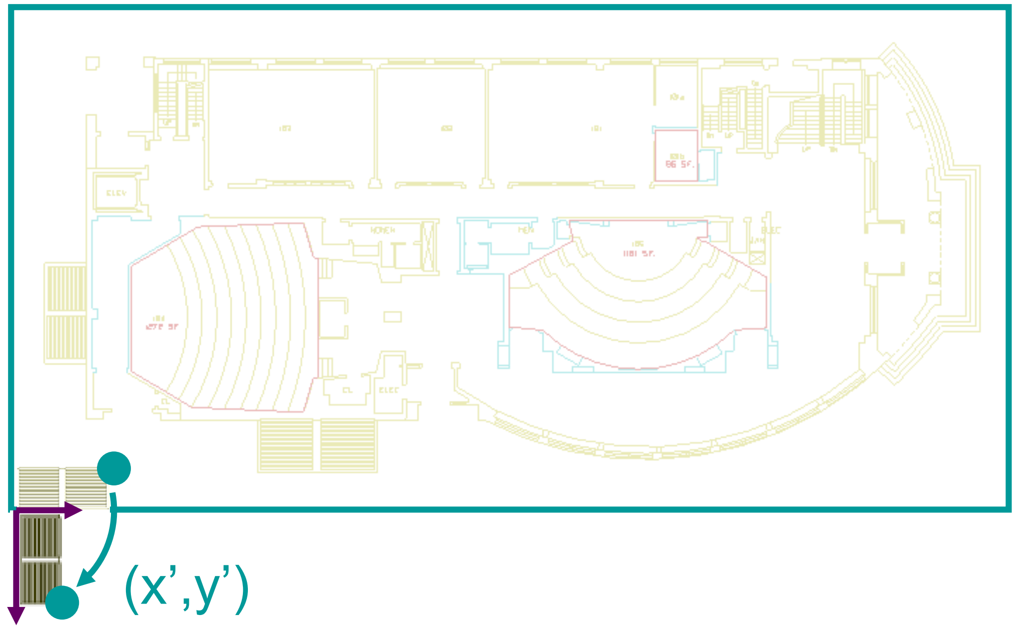
Basic 2D transformations (combination)

- 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 (combination)

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

- Rotation

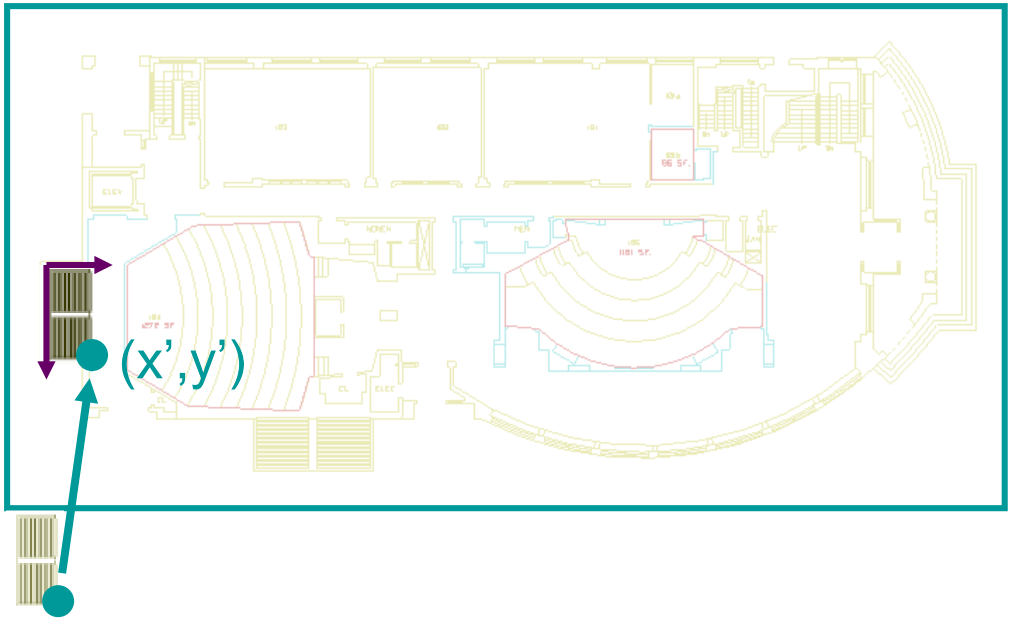
  $x' = x * \cos\Theta - y * \sin\Theta$

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

- Translation

  $x' = x + t_x$

  $y' = y + t_y$



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

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

Queen Mary
University of London

Today's agenda

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

Queen Mary University of London

Matrix representation

- Represent 2D transformation by a matrix

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

- Multiply matrix by column vector
  $\Leftrightarrow$ 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

Matrix multiplication is not generally commutative !

Queen Mary University of London

2x2 matrices

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

2D identity

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

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

2D scale

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

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

Queen Mary
University of London

2x2 matrices

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

2D rotate around (0,0)

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

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

2D shears

$$\begin{cases} x' = x + sh_x * y \\ y' = y \end{cases}$$

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

$$\begin{cases} x' = x \\ y' = sh_y * x + y \end{cases}$$

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

Queen Mary
University of London

2x2 matrices

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

2D mirror about Y axis

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

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

2D mirror over (0,0)

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

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

2x2 matrices

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

2D translation

$$x' = x + t_x$$

$$y' = y + t_y$$

NO!

Only linear 2D transformations
can be represented with a 2x2 matrix

Queen Mary
University of London

Linear transformations

- Linear transformations are combinations of
  - scale
  - rotation
  - shear and
  - mirror

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

- Properties of linear transformations
  - origin maps to origin
  - lines map to lines
  - parallel lines remain parallel
  - ratios are preserved
  - closed under composition

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

Queen Mary University of London

Homogeneous coordinates

- Homogeneous coordinates
  - represent coordinates in 2 dimensions with a 3D vector
  - seem unintuitive, but they make graphics operations much easier

$$\begin{bmatrix} x \\ y \end{bmatrix} \xrightarrow{\text{homogeneous coords}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Queen Mary University of London

Homogeneous coordinates

- How can we represent translation as a 3x3 matrix?
  - Using the rightmost column

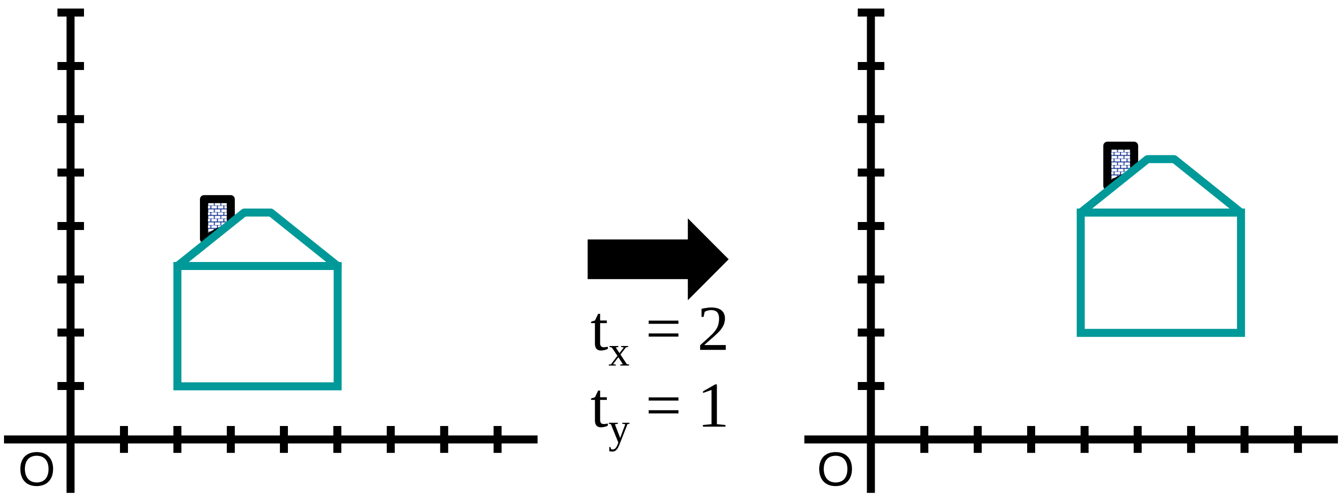
$$x' = x + t_x$$
$$y' = y + t_y$$

$$\mathbf{Translation} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Queen Mary
University of London

Translation

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

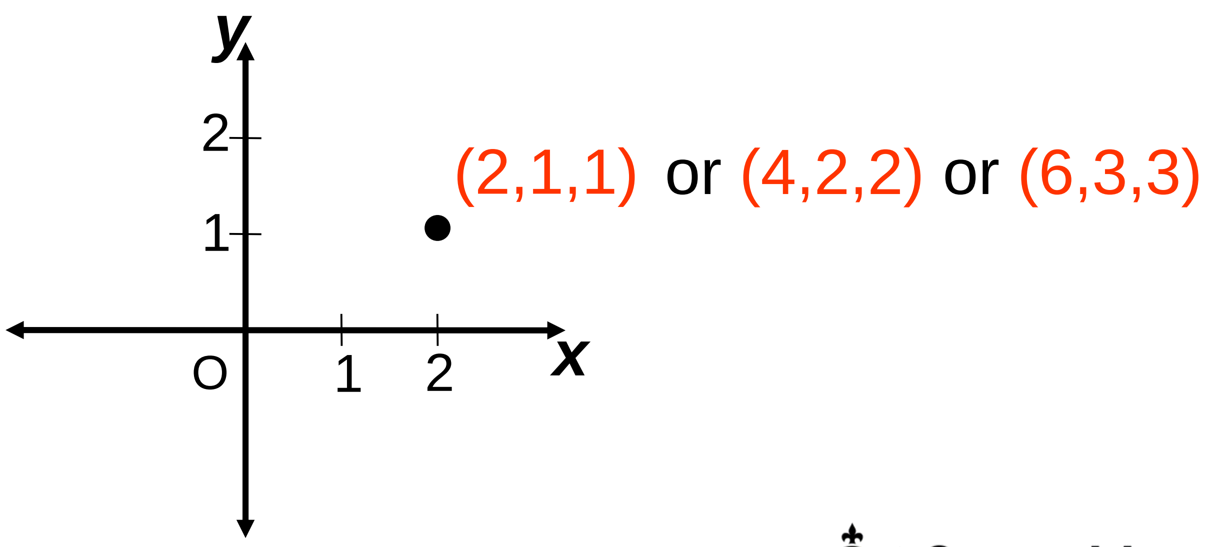


Queen Mary University of London

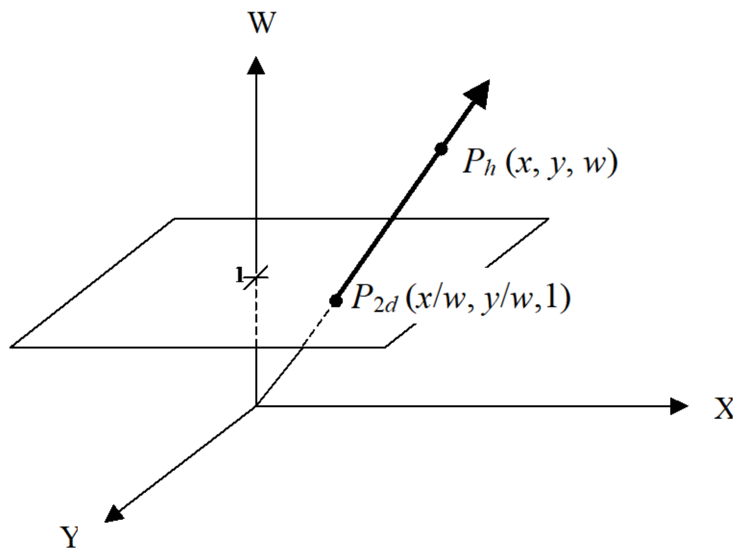
Homogeneous coordinates

- 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



Homogeneous Coordinates



$$P_{2d}(x, y) \quad \rightarrow \quad P_h(wx, wy, w), \quad w \neq 0$$

$$P_h(x', y', w), \quad w \neq 0 \quad \rightarrow \quad P_{2d}(x, y) = P_{2d}\left(\frac{x'}{w}, \frac{y'}{w}\right)$$

Homogeneous coordinates allow translation, scaling and rotation to be expressed homogeneously, allowing composition via multiplication

Basic 2D transformations

- Basic 2D transformations as 3x3 matrices

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & \boldsymbol{t}_x \\ 0 & 1 & \boldsymbol{t}_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

translate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_x & 0 & 0 \\ 0 & \boldsymbol{s}_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

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

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{sh}_x & 0 \\ \boldsymbol{sh}_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

shear

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
  - <u>origin does not necessarily map to origin</u>
  - lines map to lines
  - parallel lines remain parallel
  - ratios are preserved
  - closed under composition

Today's agenda

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

Matrix composition

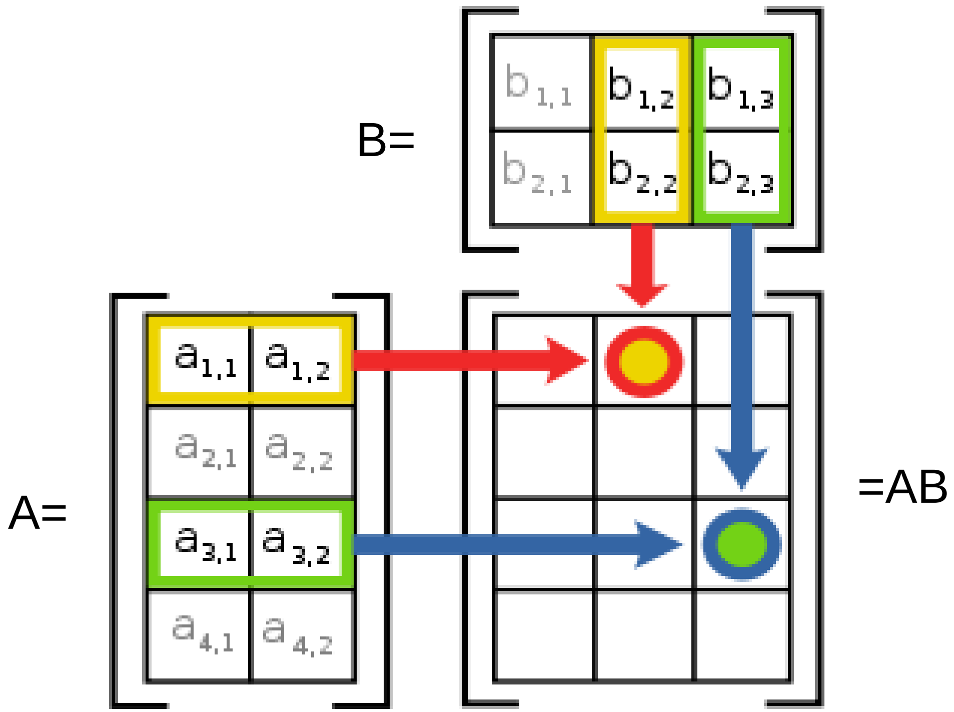
- Transformations can be combined by matrix multiplication

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

p' = $T(t_x,t_y)$ $R(\Theta)$ $S(s_x,s_y)$ **p**

Matrix multiplication (reminder)

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}$$



Matrix composition

- Matrices are a convenient and efficient way to represent a sequence of transformations
  - general purpose representation
  - hardware matrix multiply

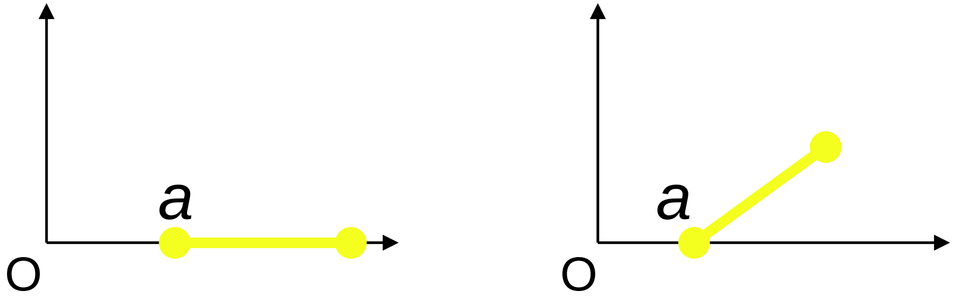
$$\mathbf{p}' = (T * (R * (S*\mathbf{p})\ )\ )$$

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

  - Note: order of transformations matters
    - matrix multiplication is generally not commutative

Example

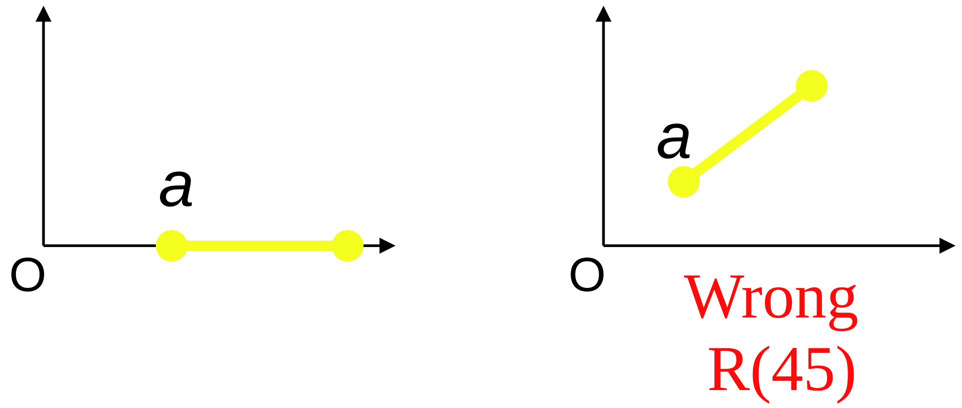
- What if we want to rotate and translate?
  - Ex: Rotate line segment by 45 degrees about endpoint *a*



Queen Mary
University of London

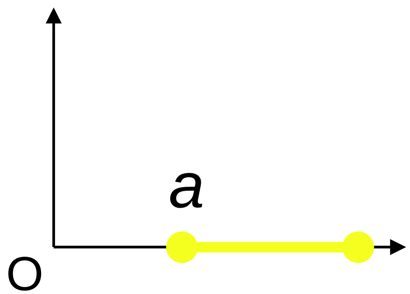
Multiplication order – wrong way

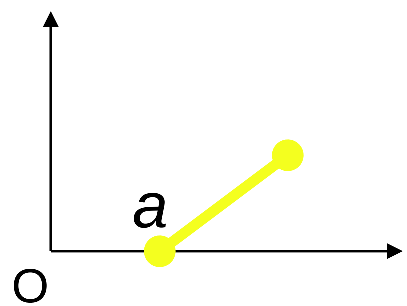
- The line segment is defined by two endpoints
  - Applying a rotation of 45 degrees, R(45), affects both points
  - We could try to translate both endpoints to return endpoint *a* to its original position, but by how much?

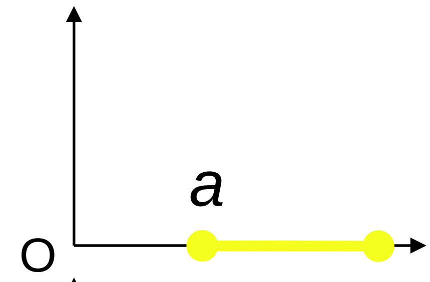


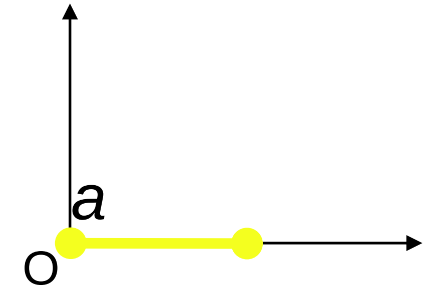
Multiplication order - correct

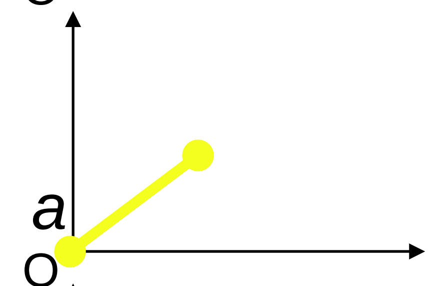
- Isolate endpoint a from rotation effects
  - First translate line so *a* is at origin: T (-3,0)
  - Then rotate line 45 degrees: R(45°)
  - Then translate back so *a* is where it was: T(3,0)

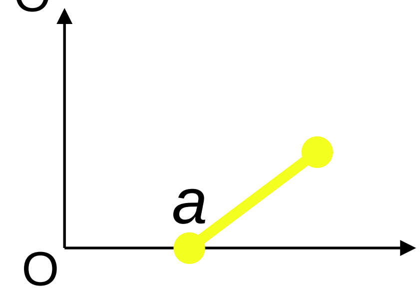












$$T(-3,0)=\begin{bmatrix}1 & 0 & -3\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}, R(45)=\begin{bmatrix}\cos(45) & -\sin(45) & 0\\ \sin(45) & \cos(45) & 0\\ 0 & 0 & 1\end{bmatrix}, T(3,0)=\begin{bmatrix}1 & 0 & 3\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$$

Example

Will this sequence of operations work?

$$\begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ 1 \end{bmatrix} = \begin{bmatrix} a'_x \\ a'_y \\ 1 \end{bmatrix}$$

$$T(-3,0) = \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R(45) = \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix}, T(3,0) = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Example

Will this sequence of operations work?

$$\begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ 1 \end{bmatrix} = \begin{bmatrix} a'_x \\ a'_y \\ 1 \end{bmatrix}$$

$$T(-3,0) = \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R(45) = \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix}, T(3,0) = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

No! Matrix multiplication is not commutative!

Example

Will this sequence of operations work?

$$\begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ 1 \end{bmatrix} = \begin{bmatrix} a'_x \\ a'_y \\ 1 \end{bmatrix}$$

$$T(-3,0) = \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R(45) = \begin{bmatrix} \cos(45) & -\sin(45) & 0 \\ \sin(45) & \cos(45) & 0 \\ 0 & 0 & 1 \end{bmatrix}, T(3,0) = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matrix composition

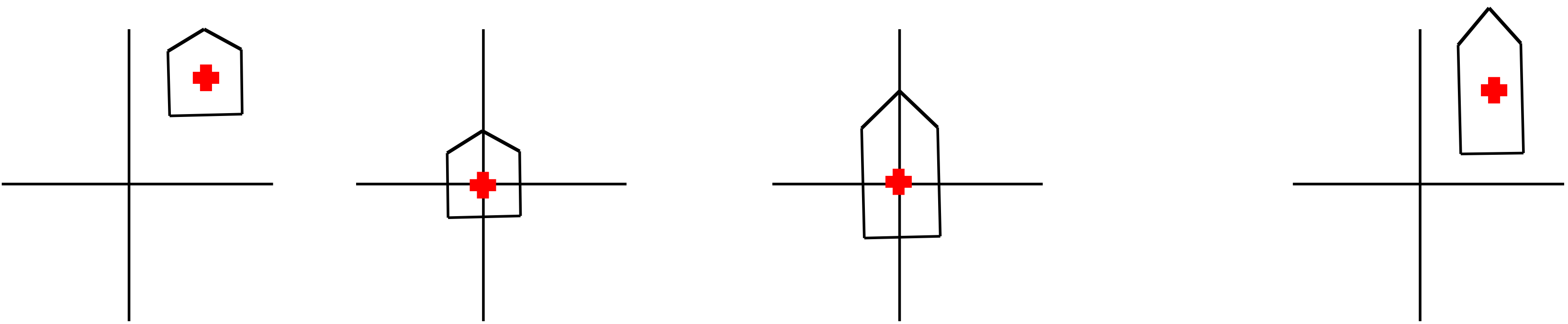
- After correctly ordering the matrices
- Multiply matrices together
- The results is just one matrix for the whole transformation
- Multiply this matrix by the vector of each vertex

→ All vertices easily transformed with one matrix to multiply

Composition of 2D Transformations

- **Scaling about a fixed point, not origin**

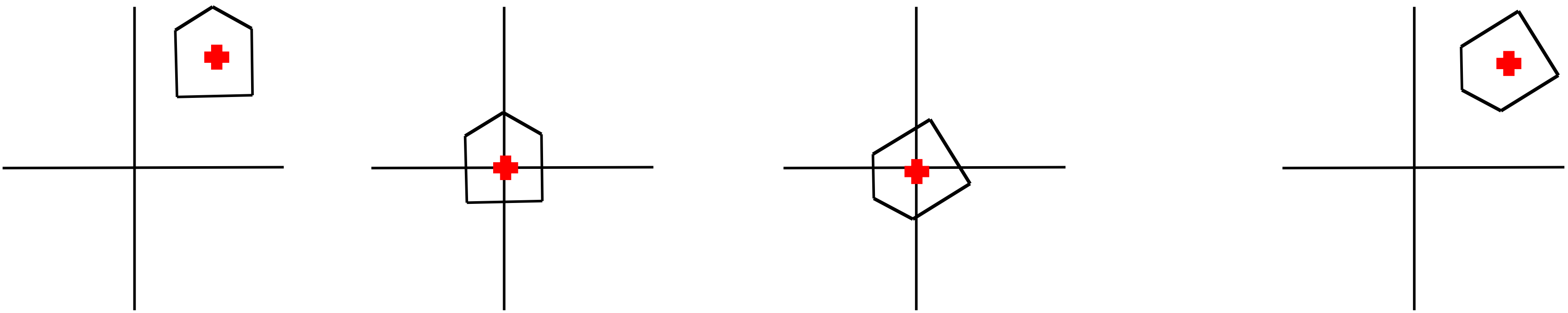
$House\ (H)$ $\qquad \boldsymbol{T}(dx, dy)H$ $\qquad \boldsymbol{S}(S_x, S_y)\boldsymbol{T}(dx, dy)H$ $\qquad \boldsymbol{T}(-dx, -dy)\boldsymbol{S}(S_x, S_y)\boldsymbol{T}(dx, dy)H$



Composition of 2D Transformations

- **Rotation about a fixed point, not origin**

$House\ (H)$ $\quad$ $\boldsymbol{T}(dx, dy)H$ $\quad$ $\boldsymbol{R}(\theta)\boldsymbol{T}(dx, dy)H$ $\quad$ $\boldsymbol{T}(-dx, -dy)\boldsymbol{R}(\theta)\boldsymbol{T}(dx, dy)H$

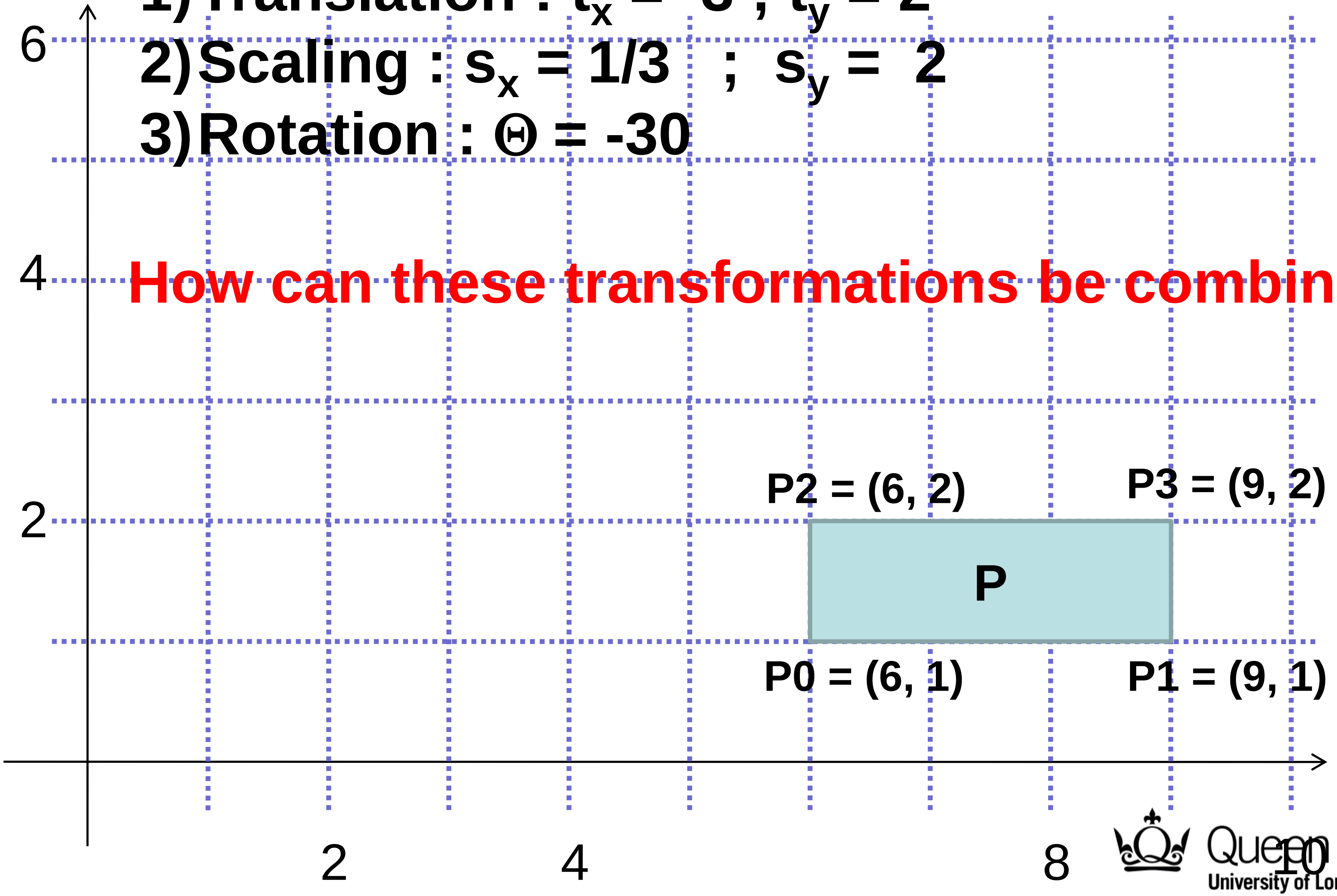


Queen Mary
University of London

Exercise

1) **Translation : $t_x = -3$; $t_y = 2$**
2) **Scaling : $s_x = 1/3$; $s_y = 2$**
3) **Rotation : $\Theta = -30$**

How can these transformations be combined?



Queen Mary
University of London

Today's agenda

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

Queen Mary
University of London

3D Transformations

- **Similar to 2D => 3D**
- **Homogenization**

3D coordinates:

$$(x, y, z) \rightarrow (x, y, z, 1) \rightarrow (wx, wy, wz, w)$$

Homogeneous:

$$(x, y, z, w) \rightarrow (x/w, y/w, z/w, 1) \rightarrow (x/w, y/w, z/w)$$

3D transformation matrices: 4x4 matrices

Queen Mary University of London

3D Transformations

- **Translation**

$$T(dx, dy, dz) = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T^{-1}(dx, dy, dz) = T(-dx, -dy, -dz)$$

- **Preserves lengths, angles, areas, and volumes**

Queen Mary
University of London

3D Transformations

- **Scaling**

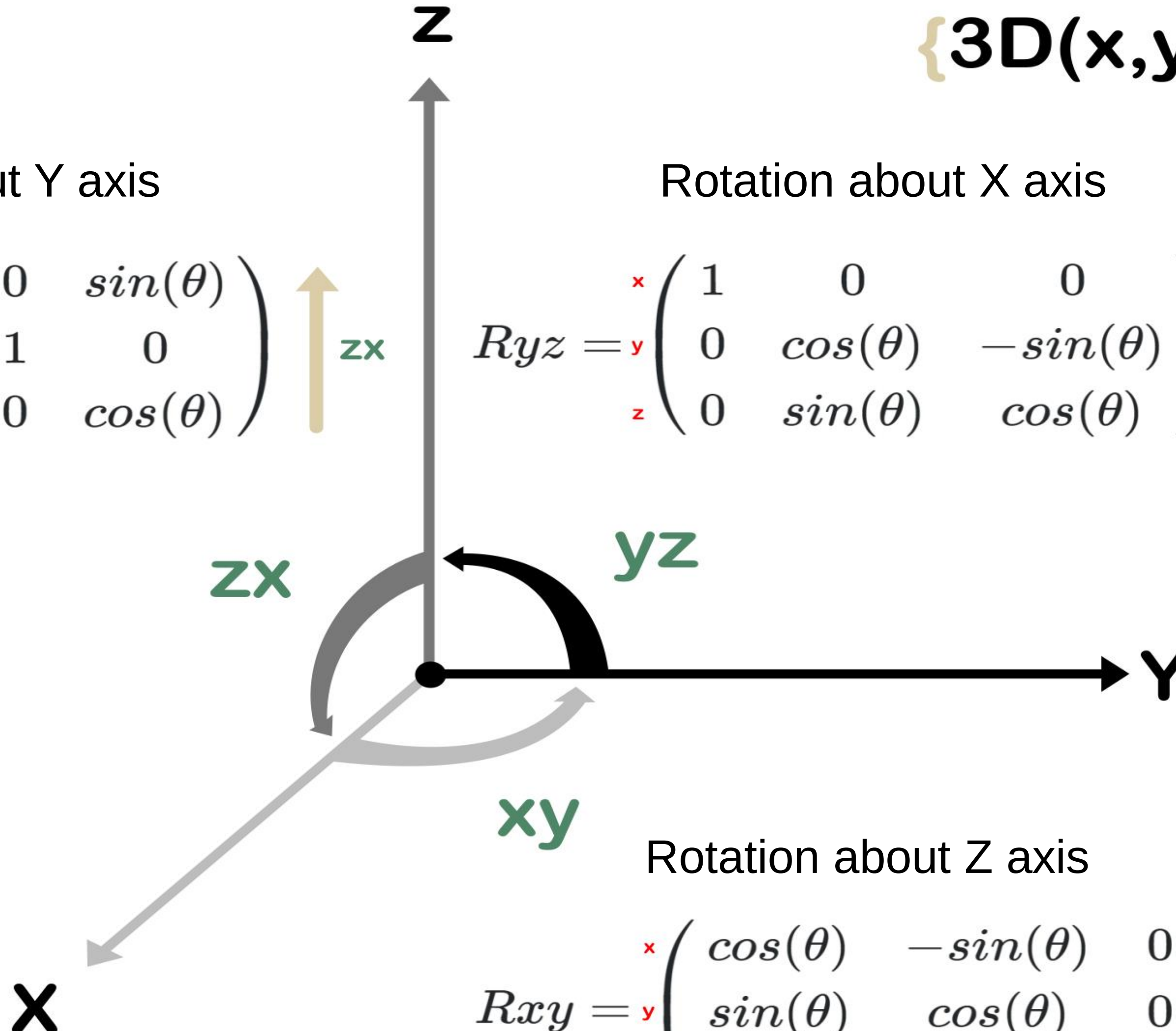
$$S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S^{-1}(s_x, s_y, s_z) = S(s_x^{-1}, s_y^{-1}, s_z^{-1})$$

- **A negative value on one or three of the components of scales results in a reflection.**
- **Does not preserve lengths, angles, areas, or volumes, except when scaling is uniform**

3D canonical rotations

{3D(x,y,z)}



Rotation about Y axis

$$Rzx = \begin{array}{c} x \\ y \\ z \end{array} \begin{pmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{pmatrix} \uparrow zx$$

Rotation about X axis

$$Ryz = \begin{array}{c} x \\ y \\ z \end{array} \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & -sin(\theta) \\ 0 & sin(\theta) & cos(\theta) \end{pmatrix} \downarrow yz$$

Rotation about Z axis

$$Rxy = \begin{array}{c} x \\ y \\ z \end{array} \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \downarrow xy$$

3D Transformations

- **Rotation**

about x-axis: **y-axis:** **z-axis:**

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

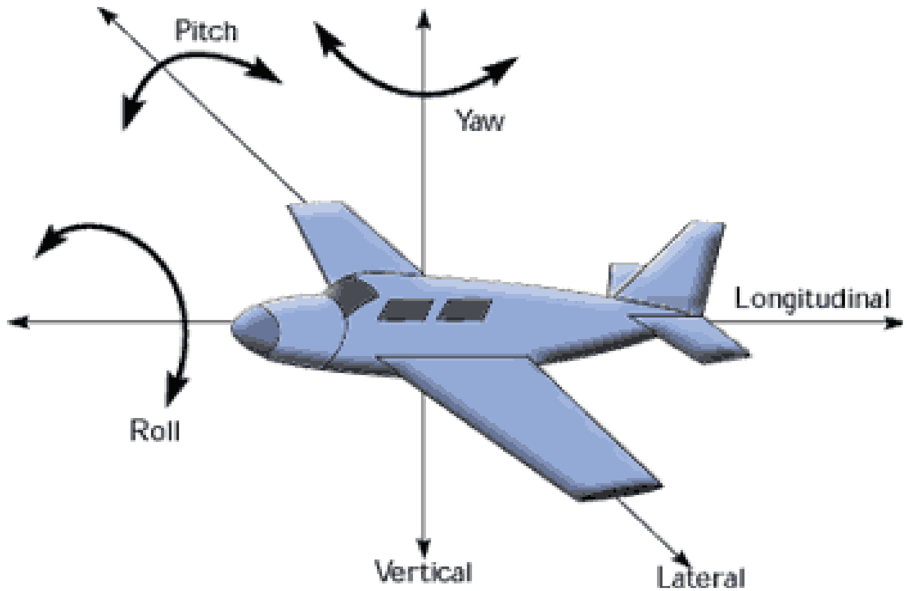
$$R_u^{-1}(\theta) = R_u(-\theta)$$

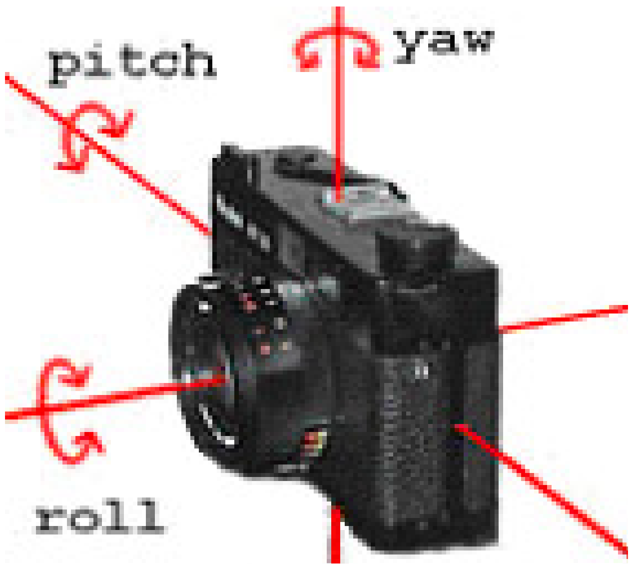
- **Preserves lengths, angles, areas, and volumes.**
- **Rotations about arbitrary axis: Any 3D rotation is a composition of 3 rotations, one about each coordinate axis (Euler angles: roll, pitch, yaw).**

Queen Mary
University of London

3D Transformations

- **Euler angles: roll, pitch, yaw**





- **Rotations about arbitrary axis: Any 3D rotation is a composition of 3 rotations, one about each coordinate axis (Euler angles: roll, pitch, yaw).**

Queen Mary
University of London

3D Transformations

- **Shearing**

$$SH = \begin{bmatrix} 1 & h_{xy} & h_{xz} & 0 \\ h_{yx} & 1 & h_{yz} & 0 \\ h_{zx} & h_{zy} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- **Basic shears use one coordinate to shear another, use one coordinate to shear other two, or use two coordinates to shear the other one.**
- **Does not preserve lengths, angles, areas in all directions, but preserve volumes.**

3D transformations

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & dx \\ m_{21} & m_{22} & m_{23} & dy \\ m_{31} & m_{32} & m_{33} & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = M * \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

- **3D geometric transformations can be composed by matrix multiplication in reverse order .**
- **Generally, the transformations are order dependent, and composition of transforms is not commutative, even of rotations.**

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

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ \boldsymbol{z}' \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_x & 0 & 0 & 0 \\ 0 & \boldsymbol{s}_y & 0 & 0 \\ 0 & 0 & \boldsymbol{s}_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \boldsymbol{z} \\ \boldsymbol{w} \end{bmatrix}$$

scale

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ \boldsymbol{z}' \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \boldsymbol{t}_x \\ 0 & 1 & 0 & \boldsymbol{t}_y \\ 0 & 0 & 1 & \boldsymbol{t}_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \boldsymbol{z} \\ \boldsymbol{w} \end{bmatrix}$$

translation

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

mirror about Y/Z plane

Reverse rotations

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

- To undo a rotation of θ, $R(\theta)$
  - apply the inverse of the rotation $R^{-1}(\theta) = R(-\theta)$

- To construct $R^{-1}(\theta) = R(-\theta)$
  - Inside the rotation matrix: $\cos(-\theta) = \cos(\theta)$
- The cosine elements of the inverse rotation matrix are unchanged
  - The sign of the sine elements will flip $\sin(-\theta) = -\sin(\theta)$

$\rightarrow R^{-1}(\theta) = R(-\theta) = R^{T}(\theta)$

Rotation matrices are orthogonal.

3D rotation

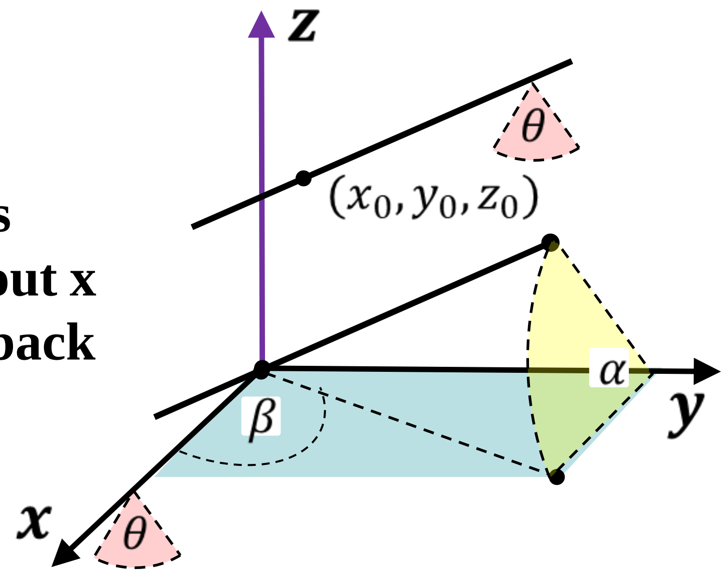
General rotations in 3D

- require rotating about an arbitrary *axis of rotation*
- deriving the rotation matrix for such a rotation directly is a good exercise in linear algebra …
- standard approach
  - express general rotation as composition of canonical rotations
    - rotations about **X, Y, Z**

Queen Mary
University of London

Composition of 3D Transformations

- **Rotation about an arbitrary axis**
  - **Step1. Translate the object to the origin**
  - **Step2. Rotate to align the axis with x-axis**
  - **Step3. Perform the specified rotation about x**
  - **Step4. Inverse rotations to turn the axis back**
  - **Step5. Inverse translation to move back**



$$R_v(\theta) = \underbrace{T(x_0, y_0, z_0)}_{\textbf{Step5}} \cdot \underbrace{R_y(-\alpha) \cdot R_z(-\beta)}_{\textbf{Step4}} \cdot \underbrace{R_x(\theta)}_{\textbf{Step3}} \cdot \underbrace{R_z(\beta) \cdot R_y(\alpha)}_{\textbf{Step2}} \cdot \underbrace{T(-x_0, -y_0, -z_0)}_{\textbf{Step1}}$$

Queen Mary
University of London