2D Transformation

2D Transformation

- Transformation changes an object's:
  - Position (Translation)
  - Size (Scaling)
  - Orientation (Rotation)
  - Shape (Deformation)
- Transformation is done by a sequence of matrix operations applied on vertices.

Vector Representation

- If we define a 2D vector as: $\begin{pmatrix} x \\ y \end{pmatrix}$

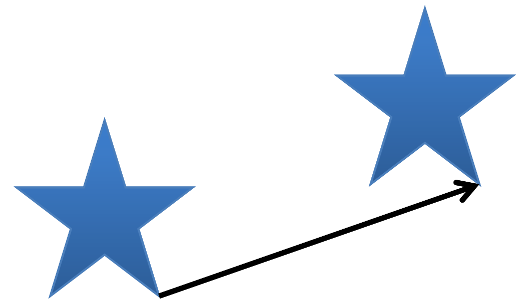
- A transformation in general can be defined as:

$$x' = ax + by + c$$
$$y' = dx + ey + f$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ d & e \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c \\ f \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Transformation: Translation



- Given a position (x, y) and an offset vector (tx, ty):

$$x' = x + tx$$

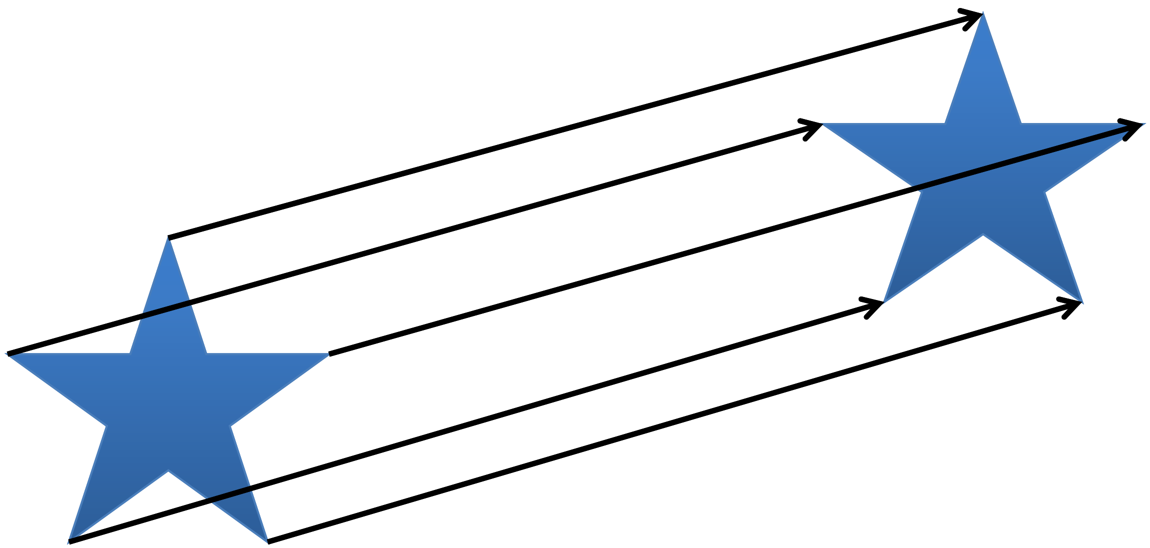
$$y' = y + ty$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} tx \\ ty \end{pmatrix}$$

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

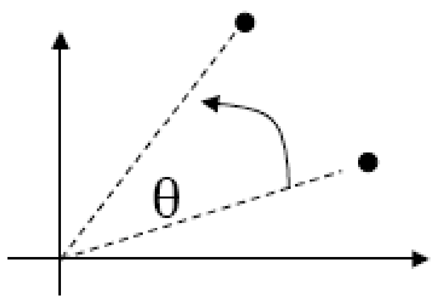
Transformation: Translation

- To translate the whole shape:

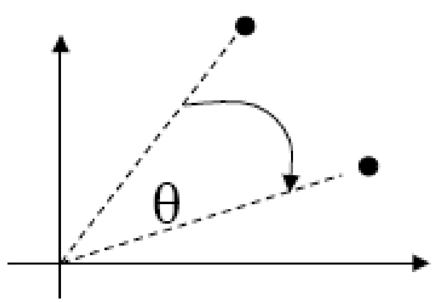


Transformation: Rotation

- Default rotation center: origin



$\theta > 0$: Rotate counter clockwise



$\theta < 0$: Rotate clockwise

Transformation: Rotation

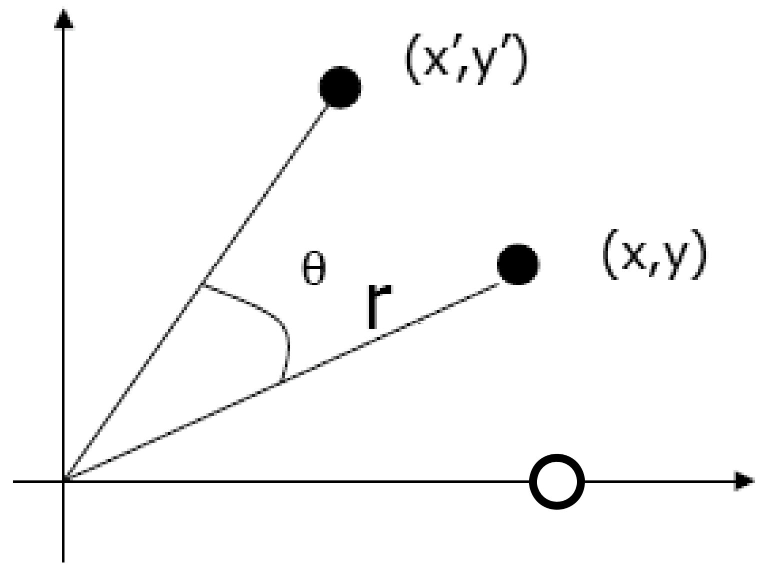
- If We rotate around the origin,
- How to rotate a vector (x, y) by an angle of θ ?
- If we assume:

$$x = r\cos\phi$$

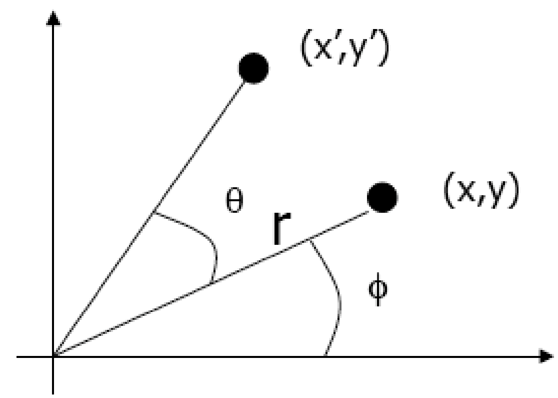
$$y = r\sin\phi$$

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

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



Transformation: Rotation



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

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

Before Rotation — After Rotation

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

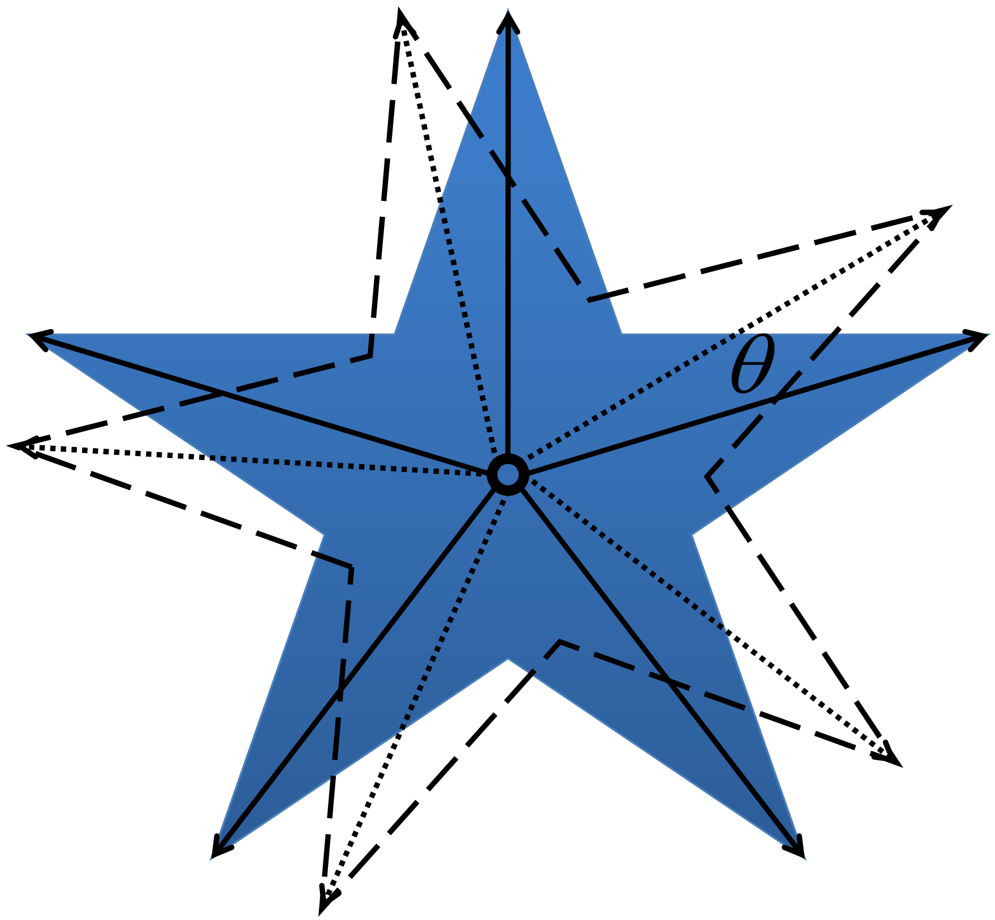
$$y' = r\cos\phi\sin\theta + r\sin\phi\cos\theta$$

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

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & \\ \sin\theta & \cos\theta & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Transformation: Rotation

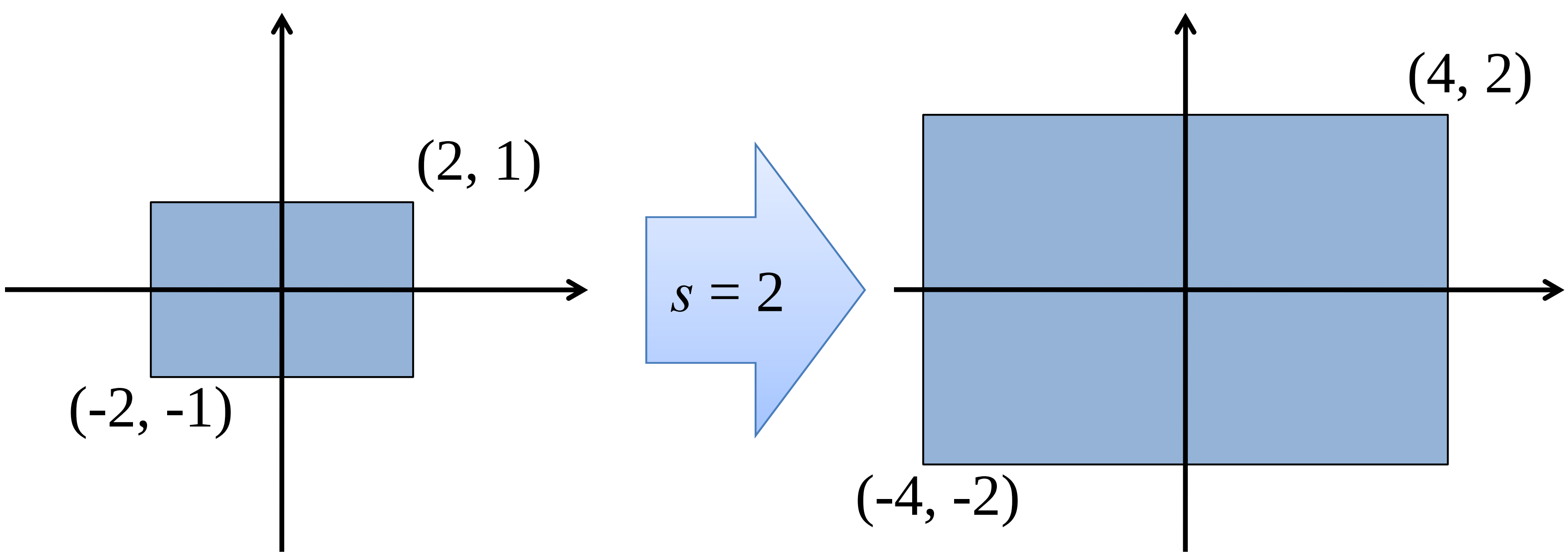
- To translate the whole shape:



Transformation: Scaling

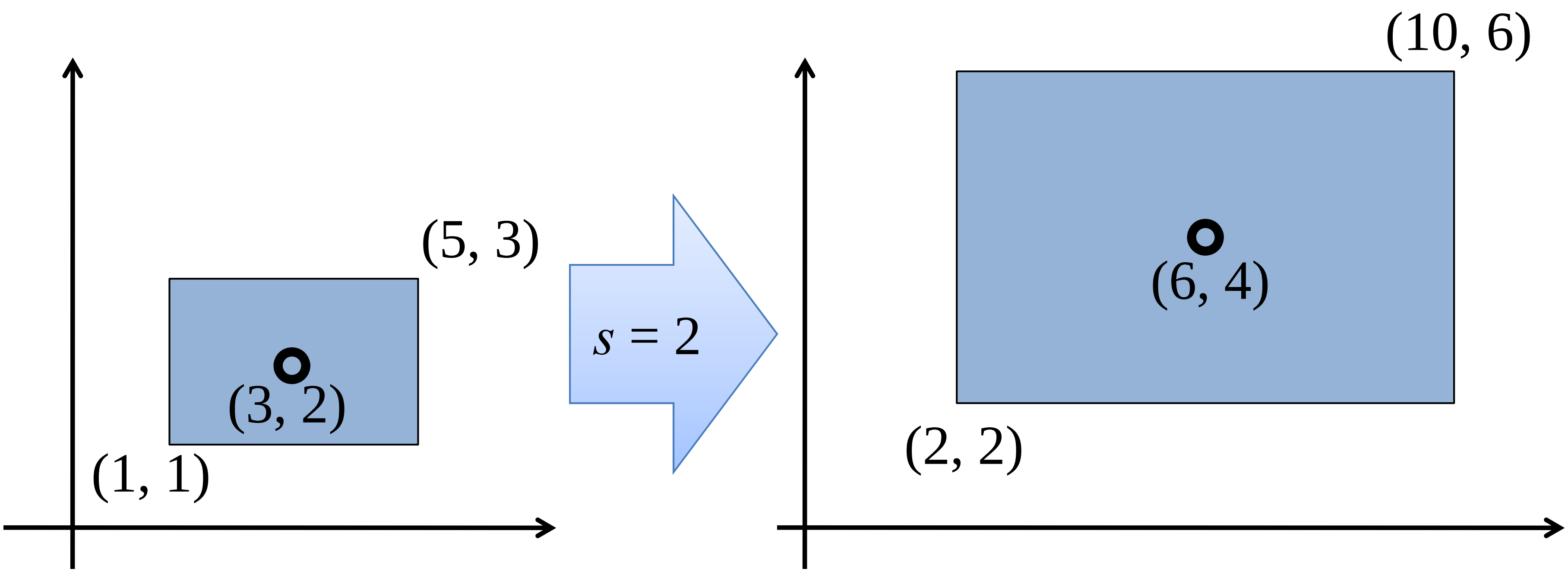
- Change the object size by a factor of s:

$$x' = s \cdot x \qquad y' = s \cdot y \quad \Rightarrow \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s & \\ & s \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s & & \\ & s & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$



Transformation: Scaling

- But when the object is not center at origin



Transformation: Scaling

- Isotropic (uniform) scaling

$$x' = s \cdot x \qquad y' = s \cdot y \quad \text{or} \quad \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s & & \\ & s & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- Anistropic (non-uniform) scaling

$$x' = sx \cdot x \qquad y' = sy \cdot y \quad \text{or} \quad \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} sx & & \\ & sy & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Summary

- Translation:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} tx \\ ty \end{pmatrix}$$

- Rotation

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

- Scaling

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

Summary

- Translation: $$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & & ox \\ & 1 & oy \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$
- Rotation $$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & \\ \sin\theta & \cos\theta & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$
- Scaling $$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s & & \\ & s & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

We use the homogenous coordinate to make sure transformations have the same form.

Why use 3-by-3 matrices?

- Transformations now become the consistent.
  - Represented by as matrix-vector multiplication.

- We can stack transformations using matrix multiplications.

Some math…

- Two vectors $\mathrm{u} = \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \end{pmatrix}, \ \mathrm{v} = \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \vdots \end{pmatrix}$ are orthogonal if:

$$\mathrm{u} \cdot \mathrm{v} = u_0 v_0 + u_1 v_1 + u_2 v_2 + \cdots = 0$$

$$\mathrm{u}^{\mathrm{T}} \mathrm{v} = \begin{pmatrix} u_0 & u_1 & u_2 & \cdots \end{pmatrix} \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \vdots \end{pmatrix} = \mathrm{u} \cdot \mathrm{v}$$

A Quiz

$$u = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \quad v = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} \qquad\qquad u = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}, \quad v = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$$

Some math…

- A matrix $\mathbf{M} = \begin{pmatrix} | & | & | & \\ \mathrm{m}_0 & \mathrm{m}_1 & \mathrm{m}_2 & \cdots \\ | & | & | & \end{pmatrix}$ is orthogonal if and only if:

$$\boxed{\mathrm{m}_i \cdot \mathrm{m}_j = 0, \qquad \forall i, j \; (i \neq j)}$$

Some math…

- A matrix $\mathbf{M}$ is orthogonal if and only if:

$$\mathbf{M}^{\mathrm{T}}\mathbf{M} = \begin{pmatrix} - & \mathrm{m}_0^{\mathrm{T}} & - \\ - & \mathrm{m}_1^{\mathrm{T}} & - \\ - & \mathrm{m}_2^{\mathrm{T}} & - \\ & \vdots & \end{pmatrix} \begin{pmatrix} | & | & | & \\ \mathrm{m}_0 & \mathrm{m}_1 & \mathrm{m}_2 & \cdots \\ | & | & | & \end{pmatrix}$$

$$= \begin{pmatrix} \mathrm{m}_0^{\mathrm{T}}\mathrm{m}_0 & \mathrm{m}_0^{\mathrm{T}}\mathrm{m}_1 & \mathrm{m}_0^{\mathrm{T}}\mathrm{m}_2 & \\ \mathrm{m}_1^{\mathrm{T}}\mathrm{m}_0 & \mathrm{m}_1^{\mathrm{T}}\mathrm{m}_1 & \mathrm{m}_1^{\mathrm{T}}\mathrm{m}_2 & \cdots \\ \mathrm{m}_2^{\mathrm{T}}\mathrm{m}_0 & \mathrm{m}_2^{\mathrm{T}}\mathrm{m}_1 & \mathrm{m}_2^{\mathrm{T}}\mathrm{m}_2 & \\ & \vdots & & \ddots \end{pmatrix} = \begin{pmatrix} ? & & & \\ & ? & & \\ & & ? & \\ & & & \ddots \end{pmatrix}$$

Diagonal Matrix

$$\mathrm{m}_i \cdot \mathrm{m}_j = 0, \qquad \forall i, j$$

Some math…

- A vector $\mathrm{v} = \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \vdots \end{pmatrix}$ is normalized if:

$$v_0 v_0 + v_1 v_1 + v_2 v_2 + \cdots = 1$$

$$\mathrm{v}^{\mathrm{T}}\mathrm{v} = \begin{pmatrix} v_0 & v_1 & v_2 & \cdots \end{pmatrix} \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \vdots \end{pmatrix} = \mathrm{v} \cdot \mathrm{v} = 1$$

Some math…

- A matrix $\mathbf{M} = \begin{pmatrix} | & | & | & \\ \mathrm{m}_0 & \mathrm{m}_1 & \mathrm{m}_2 & \cdots \\ | & | & | & \end{pmatrix}$ is orthonormal

  if and only if:

Ortho: $\boxed{\mathrm{m}_i \cdot \mathrm{m}_j = 0, \qquad \forall i, j \; (i \neq j)}$

Normalized: $\boxed{\mathrm{m}_i \cdot \mathrm{m}_i = 1,}$

Some math…

- A matrix $\mathbf{M}$ is orthonormal if and only if:

$$\mathbf{M}^T\mathbf{M} = \begin{pmatrix} - & m_0^T & - \\ - & m_1^T & - \\ - & m_2^T & - \\ & \vdots & \end{pmatrix} \begin{pmatrix} | & | & | & \\ m_0 & m_1 & m_2 & \cdots \\ | & | & | & \end{pmatrix}$$

$$= \begin{pmatrix} m_0^T m_0 & m_0^T m_1 & m_0^T m_2 & \\ m_1^T m_0 & m_1^T m_1 & m_1^T m_2 & \cdots \\ m_2^T m_0 & m_2^T m_1 & m_2^T m_2 & \\ & \vdots & & \ddots \end{pmatrix} = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & \ddots \end{pmatrix}$$

Identity Matrix

Examples

- A 2D rotational matrix is orthonormal:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \mathrm{m}_0 & \mathrm{m}_1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Ortho: $\mathrm{m}_0 \cdot \mathrm{m}_1 = -\cos\theta\sin\theta + \sin\theta\cos\theta = 0$

Normalized: $\mathrm{m}_0 \cdot \mathrm{m}_0 = \cos\theta\cos\theta + \sin\theta\sin\theta = 1$

$\mathrm{m}_1 \cdot \mathrm{m}_1 = \cos\theta\cos\theta + \sin\theta\sin\theta = 1$

Examples

- Is a 2D scaling matrix orthonormal? (if sx, sy is not 1)

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \mathrm{m}_0 & \mathrm{m}_1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} sx & 0 \\ 0 & sy \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Ortho: $\mathrm{m}_0 \cdot \mathrm{m}_1 = sx \cdot 0 + 0 \cdot sy = 0$

Normalized: $\mathrm{m}_0 \cdot \mathrm{m}_0 = sx \cdot sx$

$\mathrm{m}_1 \cdot \mathrm{m}_1 = sy \cdot sy$

Summary

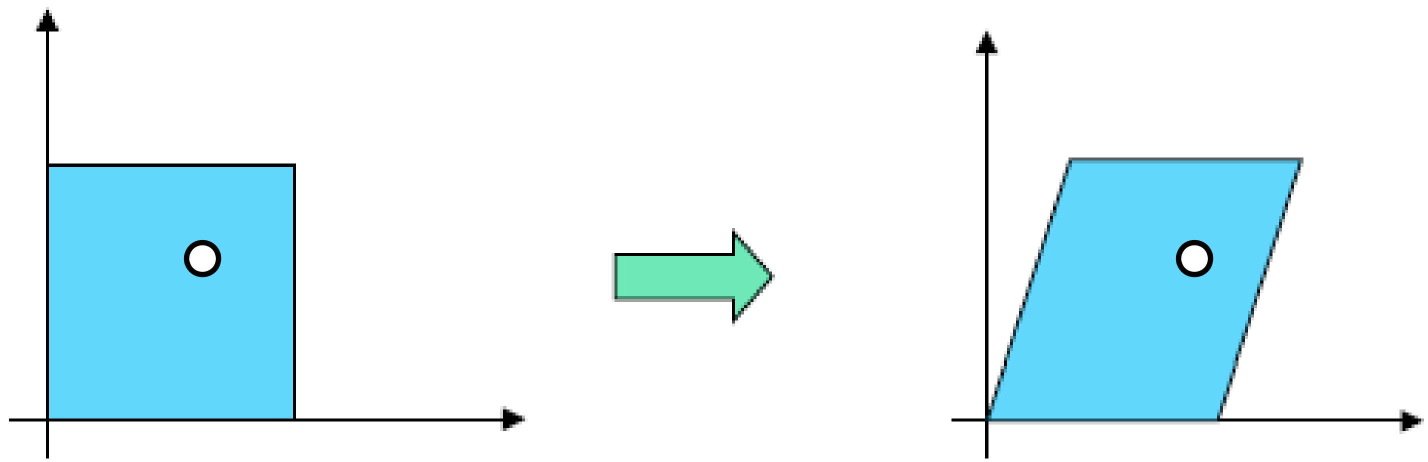
Transformation	Exists?	Ortho?	Normalized?
Translation	No	NA	NA
Rotation	Yes	Yes	Yes
Scaling	Yes	Yes	No

2D Transformation (2-by-2 matrix)

Transformation	Exists?	Ortho?	Normalized?
Translation	Yes	No	No
Rotation	Yes	Yes	Yes
Scaling	Yes	Yes	No

2D Transformation (3-by-3 matrix)

Transformation: Shearing



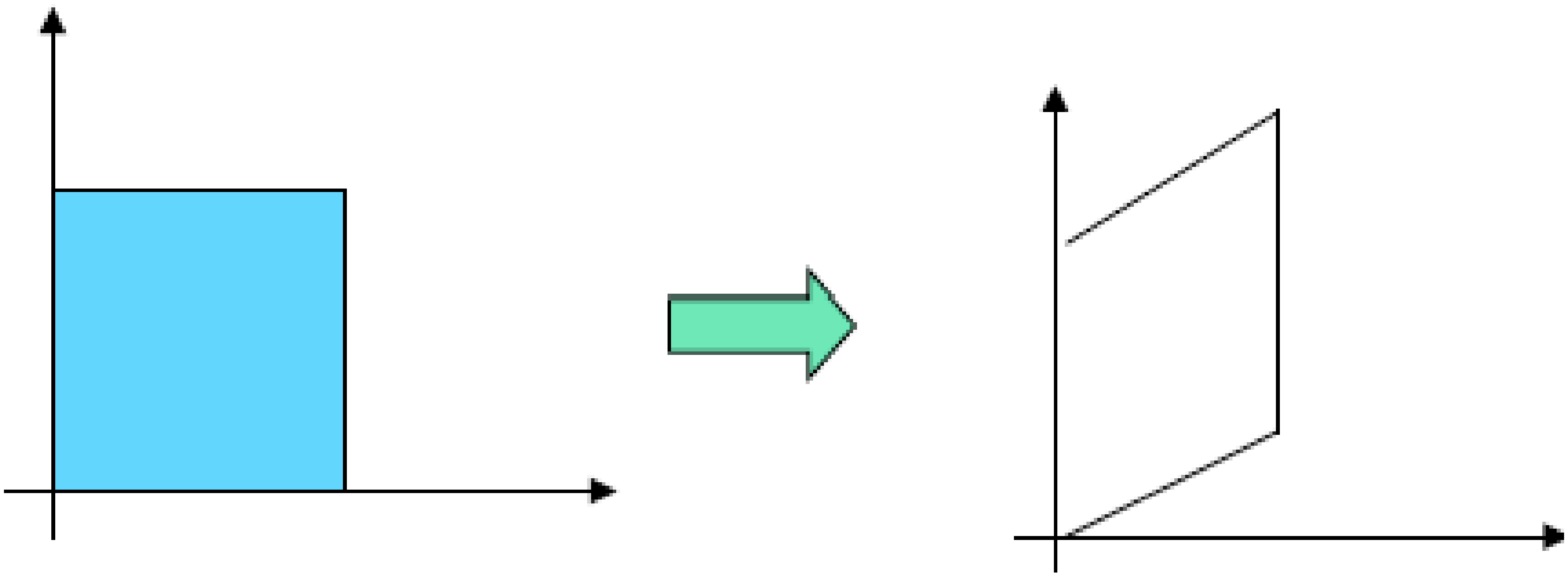
- y is unchanged.
- x is shifted according to y.

$$x' = x + h \cdot y$$
$$y' = y$$

or

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

Transformation: Shearing



$$x' = x$$
$$y' = g \cdot x + y$$

or

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

Facts about shearing

- Any 2D rotation matrix can be defined as the multiplication of three shearing matrices.
- Shearing doesn’t change the object area.
- Shearing can be defined as:

  rotation->scaling->rotation

In fact,

- Without translation, any 2D transformation matrix can be defined as:

Rotation->Scaling->rotation

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

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} sx & \\ & sy \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Orthonormal Diagonal Orthonormal

Singular Value Decomposition (SVD)

Another fact

- Same thing when using homogenous coordinates:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & \\ c & d & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \cos\phi & -\sin\phi & \\ \sin\phi & \cos\phi & \\ & & 1 \end{pmatrix}}_{\text{Orthonormal}} \underbrace{\begin{pmatrix} sx & & \\ & sy & \\ & & 1 \end{pmatrix}}_{\text{Diagonal}} \underbrace{\begin{pmatrix} \cos\theta & -\sin\theta & \\ \sin\theta & \cos\theta & \\ & & 1 \end{pmatrix}}_{\text{Orthonormal}} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

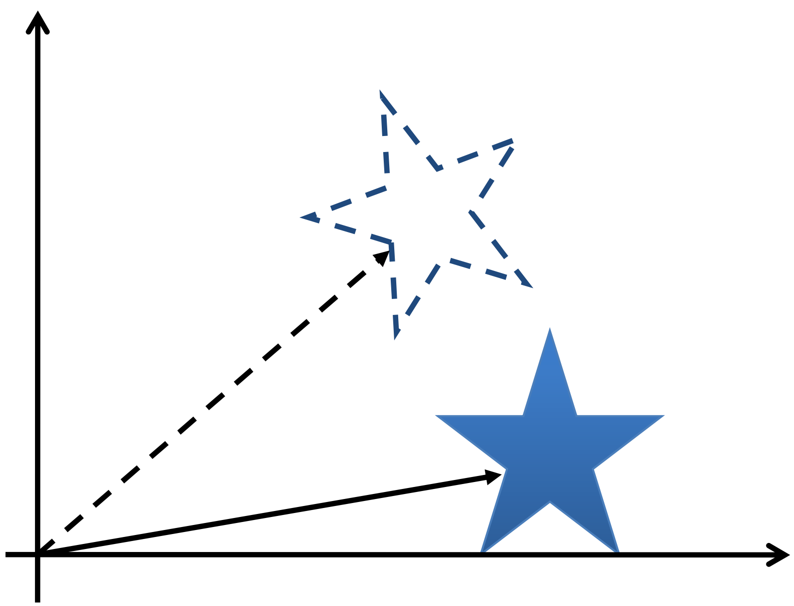
Conclusion

- Translation, rotation, and scaling are three basic transformations.

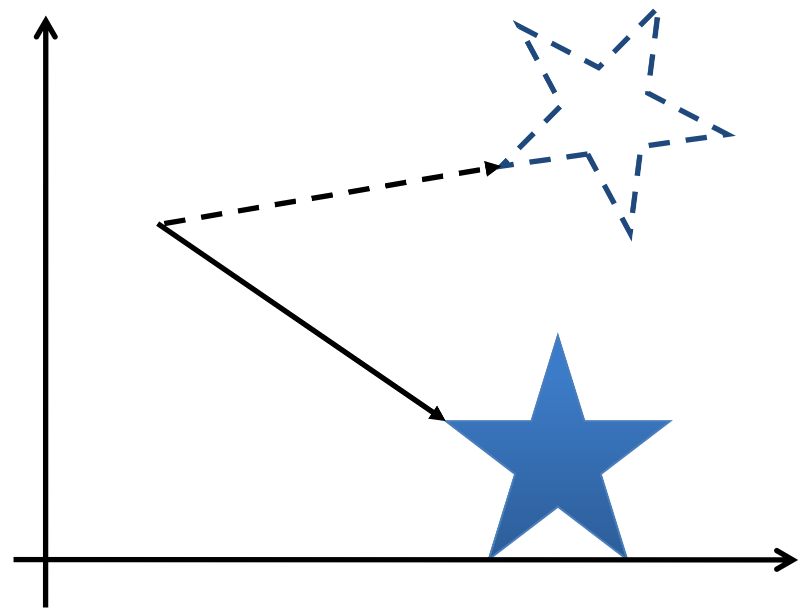
- The combination of these can represent any transformation in 2D.

- That’s why OpenGL only defines these three.

Rotation Revisit

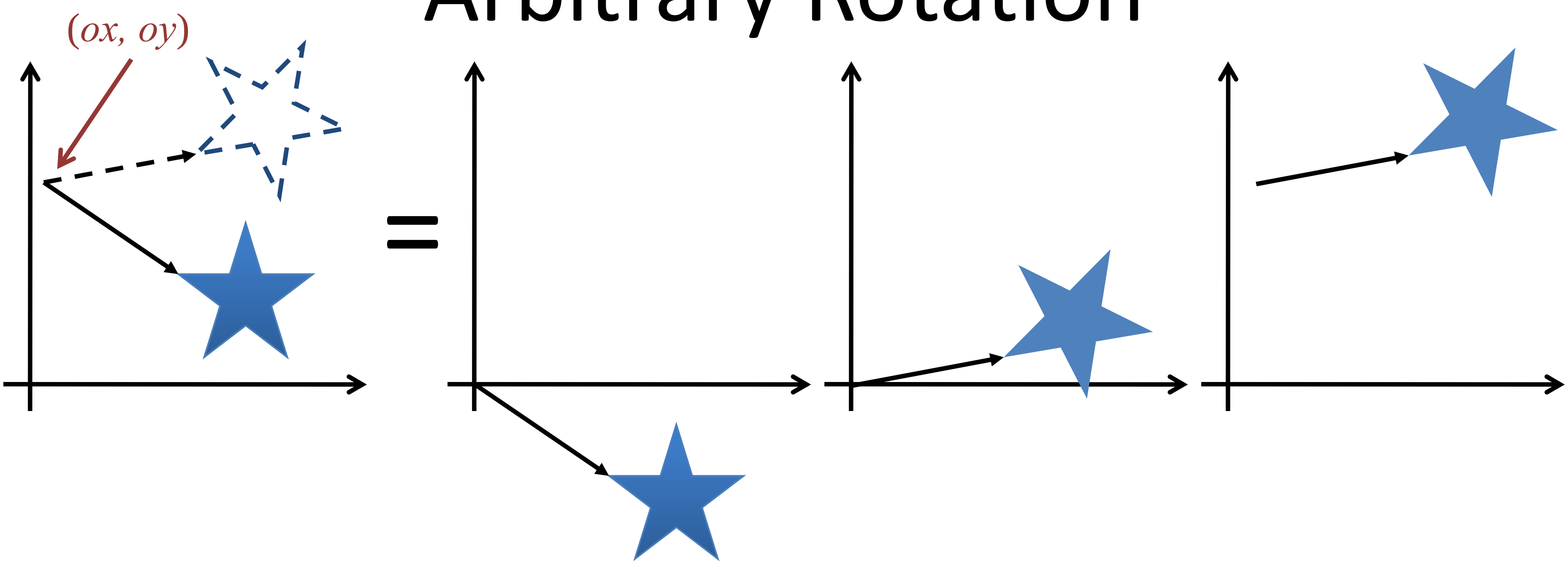


Rotate about the origin



Rotate about any center?

Arbitrary Rotation



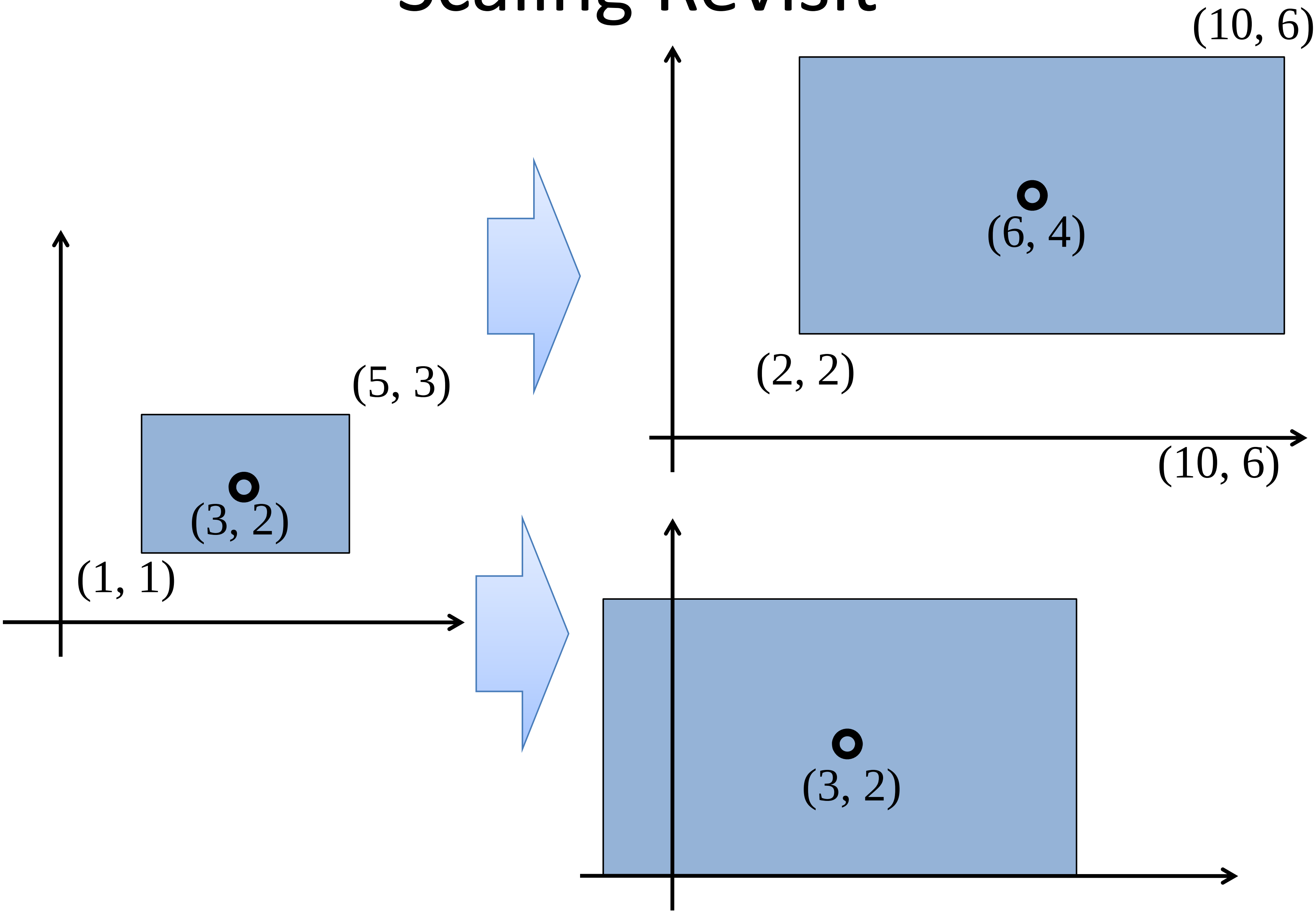
Step 1:
Translate (-ox, -oy)

Step2:
Rotate (θ)

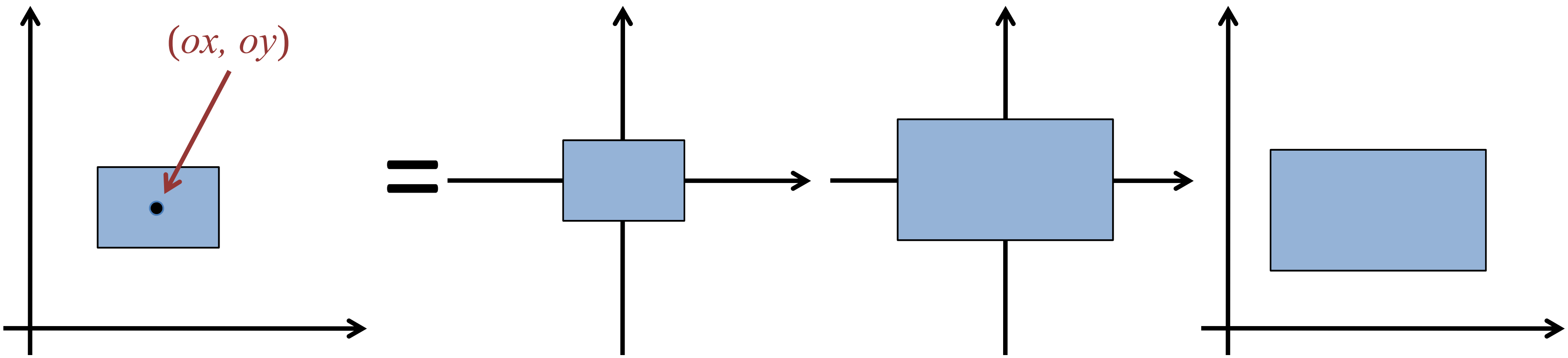
Step 1:
Translate (ox, oy)

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & & ox \\ & 1 & oy \\ & & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta & \\ \sin\theta & \cos\theta & \\ & & 1 \end{pmatrix} \begin{pmatrix} 1 & & -ox \\ & 1 & -oy \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Scaling Revisit



Arbitrary Scaling



Step 1: Translate ($-ox$, $-oy$)

Step2: Scale(sx, sy)

Step 1: Translate (ox, oy)

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & & ox \\ & 1 & oy \\ & & 1 \end{pmatrix} \begin{pmatrix} sx & & \\ & sy & \\ & & 1 \end{pmatrix} \begin{pmatrix} 1 & & -ox \\ & 1 & -oy \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

(ox, oy) can be arbitrary too!

Rigid Transformation

- A rigid transformation is:

Orthonormal Matrix

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- Rotation and translation are rigid transformation.

- Any combination of them is also rigid.

Affine Transformation

- An affine transformation is:

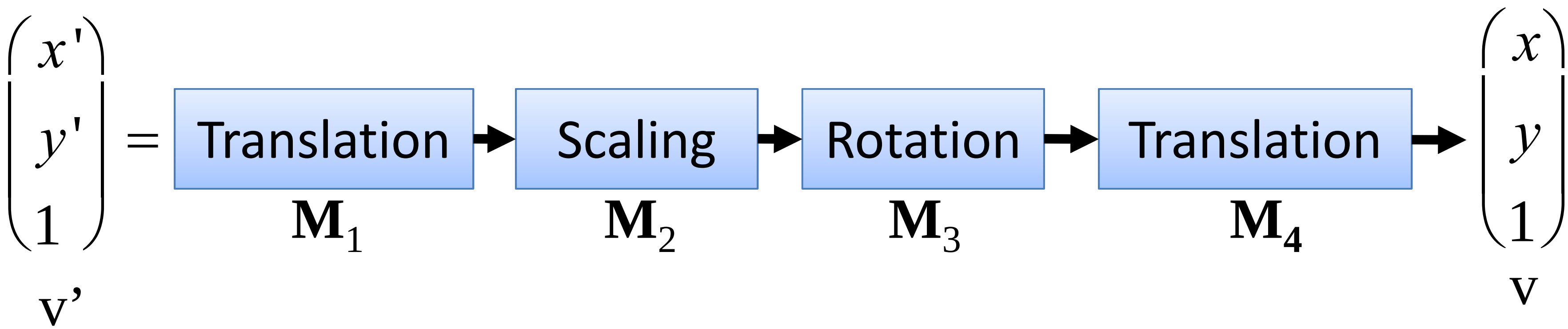
$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & & c \\ & 1 & f \\ & & 1 \end{pmatrix} \begin{pmatrix} a & b & \\ d & e & \\ & & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- Rotation, scaling, translation are affine transformation.
- Any combination of them (e.g., shearing) is also affine.
- Any affine transformation can be decomposed into:

Translation->Rotation->Scaling->Rotation

(Last) (First)

We can also compose matrix



$$v' = (\mathbf{M_1}\,(\mathbf{M_2}(\,\mathbf{M_3}\,(\mathbf{M_4}\,v))))$$

$$v' = \mathbf{M}\,v$$

Some Math

- Matrix multiplications are associative:

$$\mathbf{A}\times\mathbf{B}\times\mathbf{C}=\left(\mathbf{A}\times\mathbf{B}\right)\times\mathbf{C}=\mathbf{A}\times\left(\mathbf{B}\times\mathbf{C}\right)$$

- Matrix multiplications are not commutative:

$$\mathbf{A}\times\mathbf{B}=\mathbf{B}\times\mathbf{A}$$

- Some exceptions are:
  - Translation X Translation
  - Scaling X Scaling
  - Rotation X Rotation
  - Uniform scaling X Rotation

For example

- Rotation and translation are not commutative:

$$\begin{pmatrix} 1 & & 1 \\ & 1 & \\ & & 1 \end{pmatrix}\begin{pmatrix} 0.6 & -0.8 & \\ 0.8 & 0.6 & \\ & & 1 \end{pmatrix} = \begin{pmatrix} 0.6 & -0.8 & 1 \\ 0.8 & 0.6 & \\ & & 1 \end{pmatrix}$$

Orders are important!

$$\begin{pmatrix} 0.6 & -0.8 & \\ 0.8 & 0.6 & \\ & & 1 \end{pmatrix}\begin{pmatrix} 1 & & 1 \\ & 1 & \\ & & 1 \end{pmatrix} = \begin{pmatrix} 0.6 & -0.8 & 0.6 \\ 0.8 & 0.6 & 0.8 \\ & & 1 \end{pmatrix}$$

How OpenGL handles transformation

- All OpenGL transformations are in 3D.
- We can ignore z-axis to do 2D transformation.

For example

- 3D Translation

```
glTranslatef(tx, ty, tz)
```

- 2D Translation

```
glTranslatef(tx, ty, 0)
```

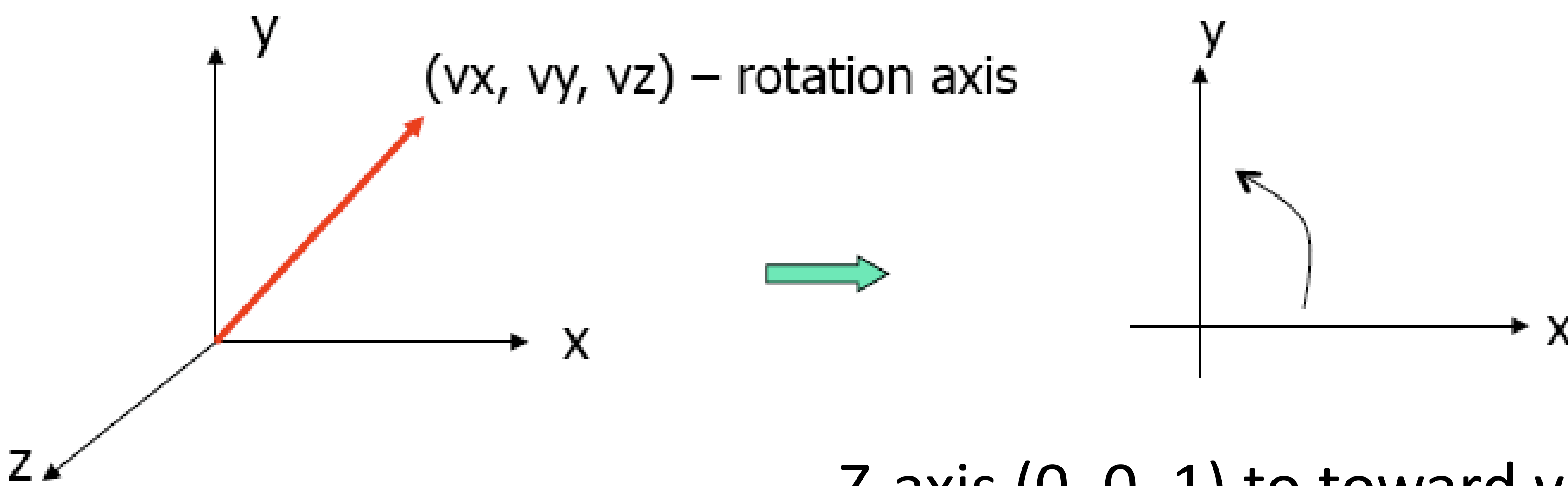
For example

- 3D Rotation

```
glTranslatef(angle, axis_x, axis_y, axis_z)
```

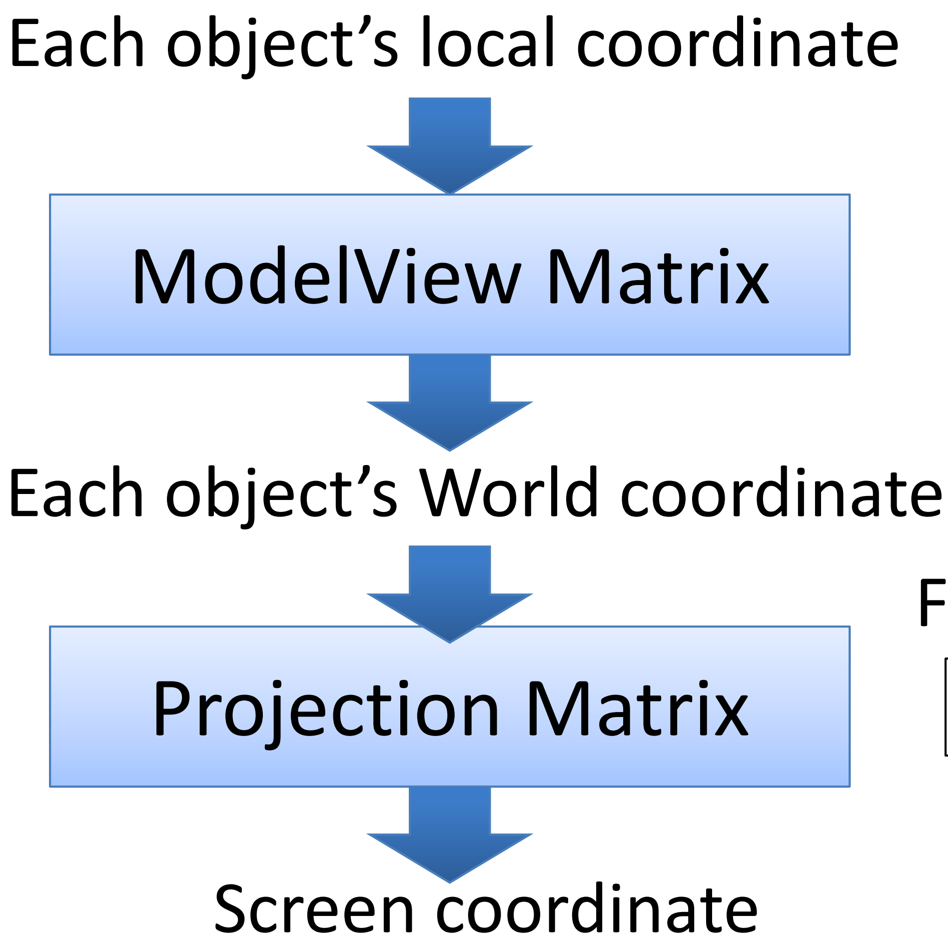
- 2D Rotation

```
glTranslatef(angle, 0, 0, 1)
```



Z-axis (0, 0, 1) to toward you

OpenGL uses two matrices:



For example:

gluOrtho2D(…)

How OpenGL handles transformation

- So you need to let OpenGL know which matrix to transform:

```
glMatrixMode(GL_MODELVIEW);
```

- You can reset the matrix as:

```
glLoadIdentity();
```

For Example

Xcode time!