Lecture-8

Geometric Transformations

Topics

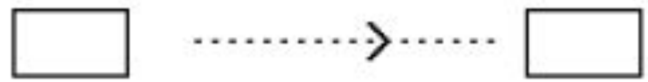
- 5.1 Basic Two-Dimensional Geometric Transformations
- 5.2 Matrix Representations and Homogeneous Coordinates
- 5.3 Inverse Transformations
- 5.4 Two-Dimensional Composite Transformations
- 5.5 Other Two-Dimensional Transformations
- 5.6 Raster Methods for Geometric Transformations
- 5.7 OpenGL Raster Transformations
- 5.8 Transformations between Two-Dimensional Coordinate Systems
- 5.9 Geometric Transformations in Three-Dimensional Space
- 5.10 Three-Dimensional Translation
- 5.11 Three-Dimensional Rotation
- 5.12 Three-Dimensional Scaling
- 5.13 Composite Three-Dimensional Transformations
- 5.14 Other Three-Dimensional Transformations
- 5.15 Transformations between Three-Dimensional Coordinate Systems
- 5.16 Affine Transformations
- 5.17 OpenGL Geometric-Transformation Functions
- 5.18 Summary

5.1 Basic Two-Dimensional Geometric Transformations

- Def : Operations that are applied to the geometric description of an object to change its position, orientation, or size are called ***geometric transformations***.
- Uses : Geometric transformations can be used to describe how objects might move around in a scene during an animation sequence or simply to view them from another angle.

- geometric transformations
  - Translation
  - Rotation
  - Scaling
  - Reflection
  - shearing

-Translation



-Scaling

-Rotation

-Reflection

Two-Dimensional Translation

- We perform a translation on a single coordinate point by adding offsets to its coordinates so as to generate a new coordinate position.
- To translate a two-dimensional position, we add translation distances, t_x and t_y to the original coordinates (x,y) to obtain the new coordinate position (x',y'),

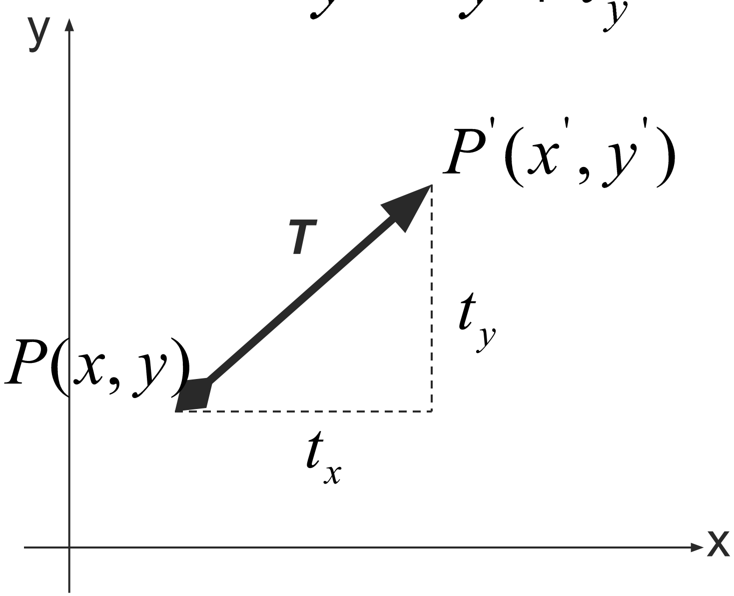
$$x' = x + t_x$$

$$y' = y + t_y$$

The two-dimensional translation equations in the matrix form

$$x' = x + t_x$$

$$y' = y + t_y$$



$$P = \begin{bmatrix} x \\ y \end{bmatrix} \qquad P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

Translation Vector $T = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$

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

$$P' = P + T$$

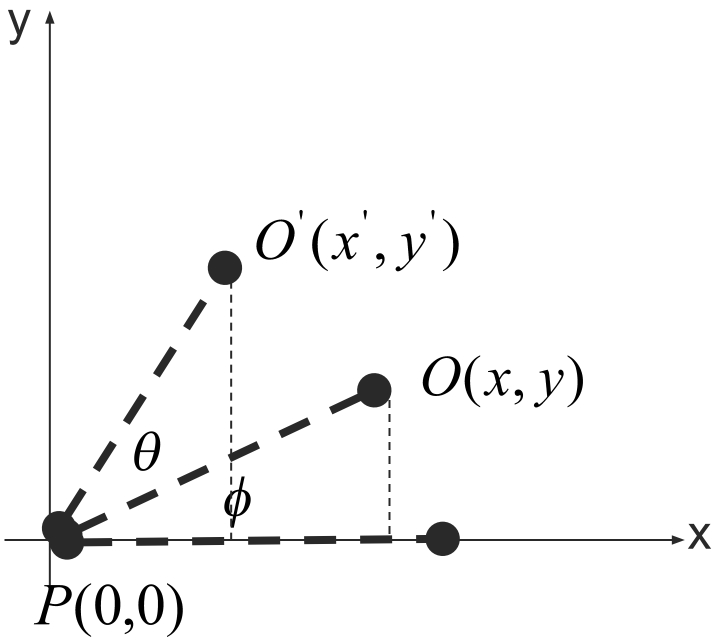
Two-Dimensional Rotation

- We generate a rotation transformation of an object by specifying a rotation axis and a rotation angle.
- A two-dimensional rotation of an object is obtained by repositioning the object along a circular path in the xy plane.
- Parameters for the two-dimensional rotation are
  - The rotation angle θ
  - A position (x,y) – rotation point (pivot point)

The two-dimensional rotation

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

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



Polar coordinate system

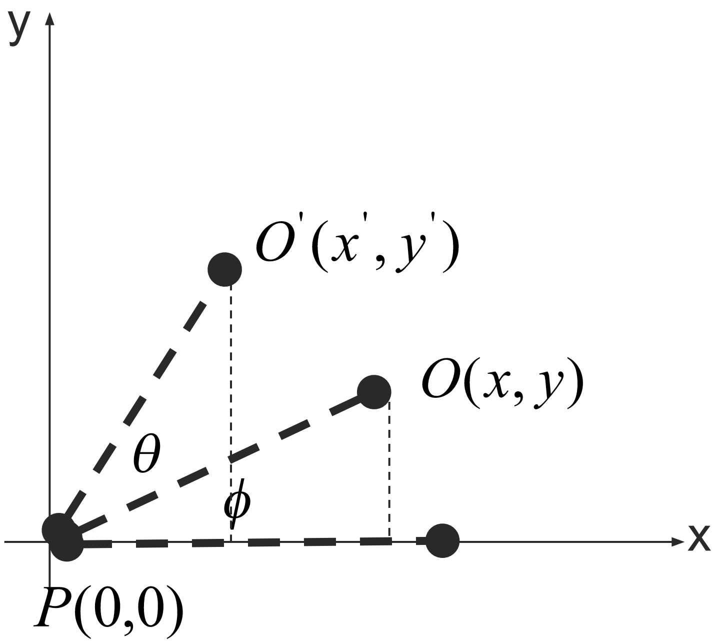
$$x = r\cos\phi$$

$$y = r\sin\phi$$

$$x' = x\cos\theta - y\sin\theta$$

$$y' = x\sin\theta + y\cos\theta$$

The two-dimensional rotation



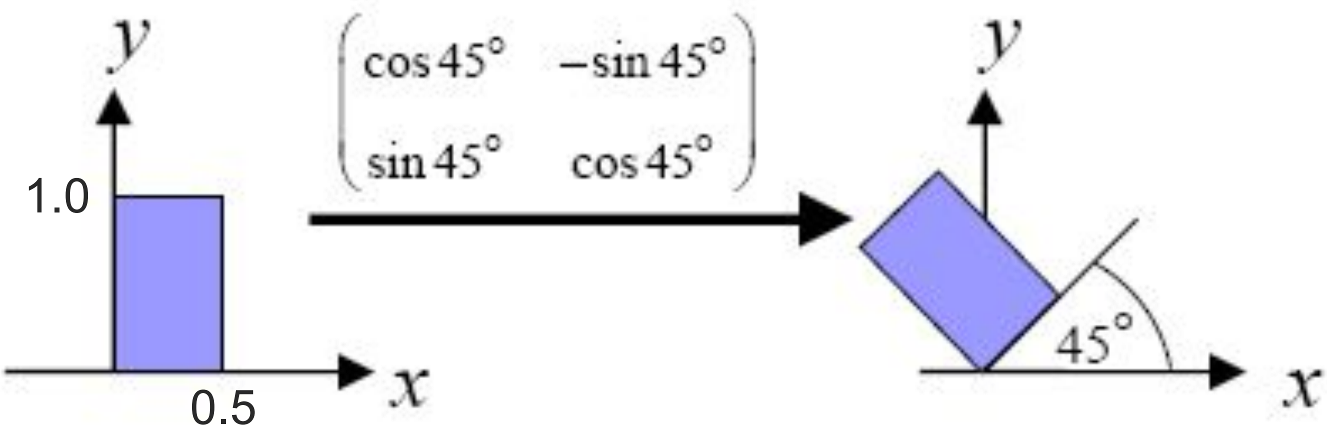
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$O' = R \bullet O$$

$$O = \begin{bmatrix} x \\ y \end{bmatrix} \qquad O' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$ ← Rotation matrix

Ex. 1

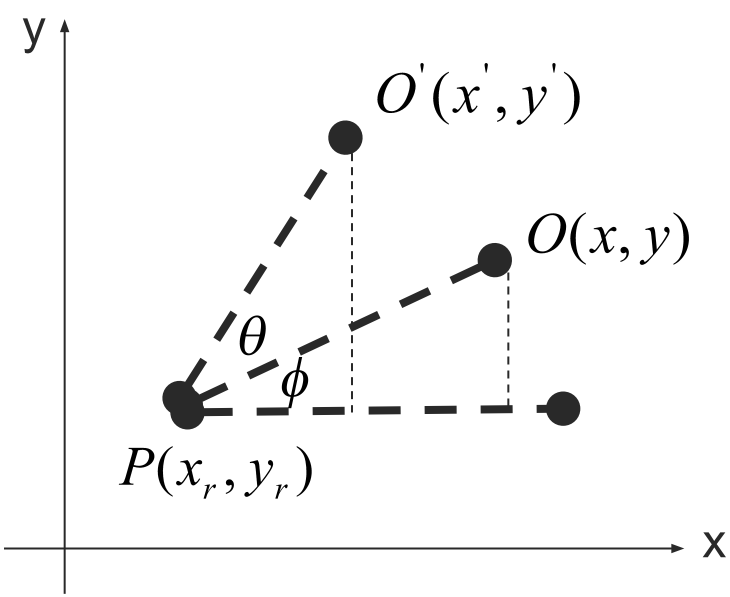


y
1.0
0.5
x
$\begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix}$
y
45°
x

Rotation of a point about an arbitrary pivot position

$$x' = (x - x_r)\cos\theta - (y - y_r)\sin\theta + x_r$$

$$y' = (x - x_r)\sin\theta + (y - y_r)\cos\theta + y_r$$



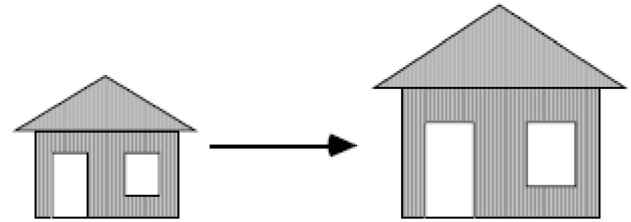
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - x_r \\ y - y_r \end{bmatrix} + \begin{bmatrix} x_r \\ y_r \end{bmatrix}$$

$$O' = R \bullet O^* + P$$

$$O^* = \begin{bmatrix} x - x_r \\ y - y_r \end{bmatrix}$$

Two-Dimensional Scaling

- To alter the size of an object, we apply a scaling transformation.
- A simple two-dimensional scaling operation is performed by multiplying object positions (x,y) by scaling factors s_x and s_y to produce the transformed coordinates (x’,y’).



$x_2 = s_x x_1$ $\quad$ $y_2 = s_y y_1$

$$x = x \cdot s_x$$

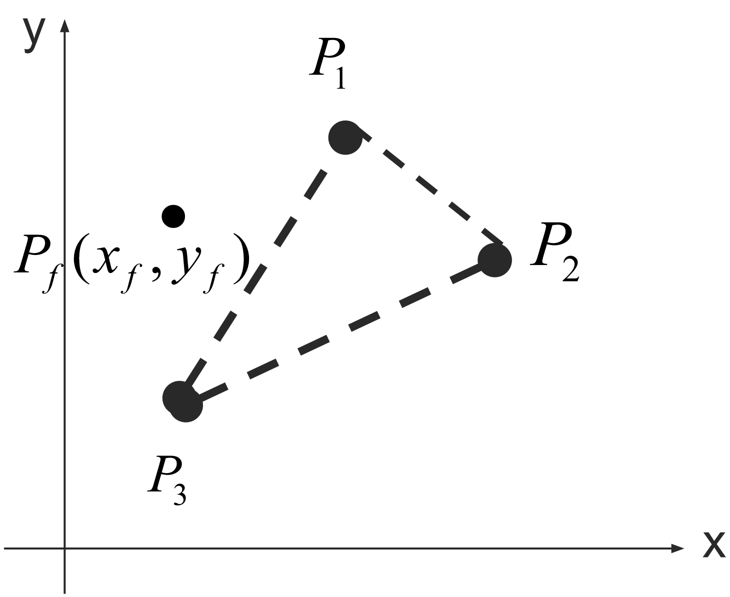
$$y' = y \cdot s_y$$

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

$$P' = S \bullet P$$

- Any positive values can be assigned to the scaling factors.
  - Values less than 1 reduce the size of object;
  - Values greater than 1 produce enlargements.
  - Uniform scaling – scaling values have the same value
  - Differential scaling – unequal of the scaling factor

Scaling relative to a chosen fixed point



$$x' = x \cdot s_x + x_f(1 - s_x)$$

$$y' = y \cdot s_y + y_f(1 - s_y)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 1 - s_x & 0 \\ 0 & 1\text{-}s_y \end{bmatrix} \begin{bmatrix} x_f \\ y_f \end{bmatrix}$$

$$P' = S \bullet P + S^* \bullet P_f$$

5.2 Matrix Representations and Homogeneous Coordinates

- Many graphics applications involve sequences of geometric transformations.
- Hence we consider how the matrix representations can be reformulated so that such transformation sequence can be efficiently processed.
- Each of three basic two-dimensional transformations (translation, rotation and scaling) can be expressed in the general matrix form

$$P' = M_1 \cdot P + M_2$$

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

Translation

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

Rotation

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

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 1 - s_x & 0 \\ 0 & 1 - s_y \end{bmatrix} \begin{bmatrix} x_f \\ y_f \end{bmatrix}$$

Scaling

- To produce a sequence of transformations such as scaling followed by rotation then translation, we could calculate the transformed coordinates one step at a time.
- A more efficient approach is to combine the transformations so that the final coordinate positions are obtained directly from the initial coordinates, without calculating intermediate coordinate values.

Homogeneous Coordinates

- Multiplicative and translational terms for a two-dimensional geometric transformations can be combined into a single matrix if we expand the representations to 3 by 3 matrices.
- Then we can use the third column of a transformation matrix for the translation terms, and all transformation equations can be expressed as matrix multiplications.

- But to do so, we also need to expand the matrix representation for a two-dimensional coordinate position to a three-element column matrix.
- A standard technique for accomplishing this is to expand each two-dimensional coordinate-position representation (x,y) to a three-element representation (x_h,y_h,h), called homogeneous coordinates, where the homogeneous parameter h is a nonzero value such that

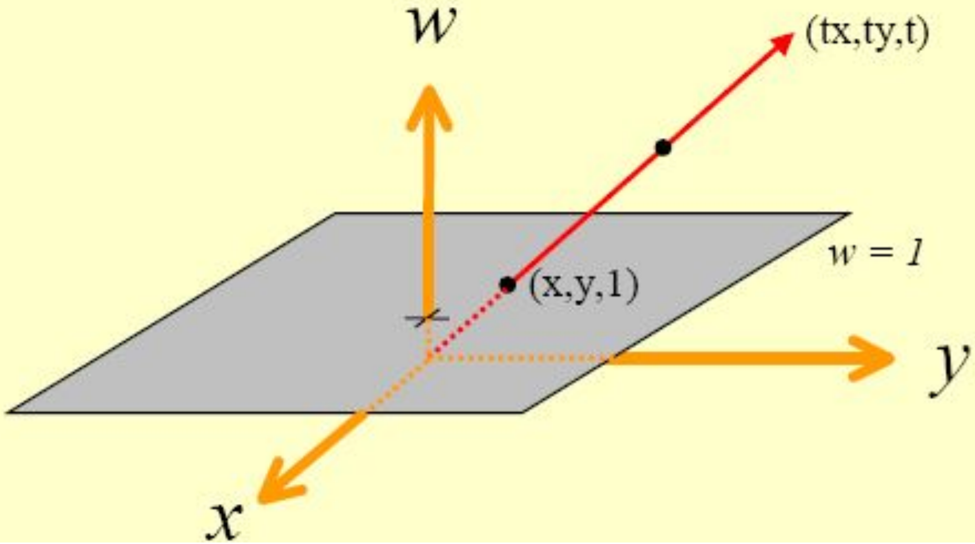
$$x = \frac{x_h}{h}, \qquad y = \frac{y_h}{h}$$

- A convenient choice is simply to set h=1.
- Each two-dimensional position is then represented with homogeneous coordinate (x,y,1).
- The term “homogeneous coordinates” is used in mathematics to refer to the effect of this representation on Cartesian equations.

Homogeneous Coordinates

An infinite number of points correspond to (x,y,1).

They constitute the whole line (tx,ty,t).



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

Translation

$$P' = T(t_x, t_y) \cdot P$$

Rotation

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

$$P' = R(\theta) \cdot P$$

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

Scaling

$$P' = S(s_x, s_y) \cdot P$$

5.3 Inverse Transformations

Inverse translation matrix

$$T^{-1} = \begin{bmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Inverse rotation matrix

$$R^{-1} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$S^{-1} = \begin{bmatrix} \frac{1}{s_x} & 0 & 0 \\ 0 & \frac{1}{s_y} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

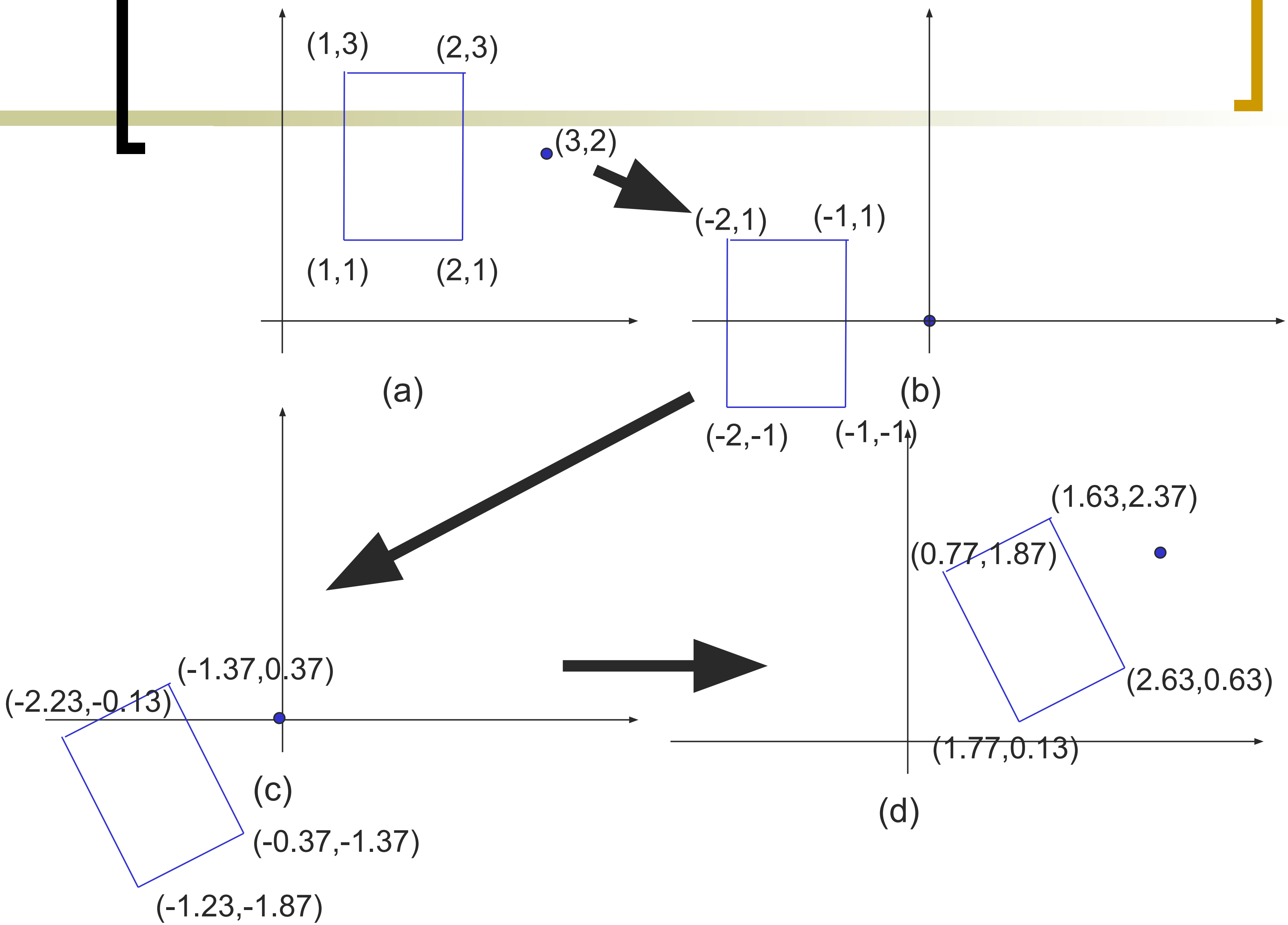
Inverse scaling matrix

5.4 Two-Dimensional Composite Transformations

- Using matrix representations, we can set up a sequence of transformations as a composite transformation matrix by calculating the product of the individual transformations.
- Thus, if we want to apply two transformations to point position P, the transformed location would be calculated as

$$P' = M_2 \cdot M_1 \cdot P$$

$$P' = M \cdot P$$


(1,3)
(2,3)
(1,1)
(2,1)
(3,2)
(a)
(-2,1)
(-1,1)
(-2,-1)
(-1,-1)
(b)
(-1.37,0.37)
(-2.23,-0.13)
(-0.37,-1.37)
(-1.23,-1.87)
(c)
(1.63,2.37)
(0.77,1.87)
(2.63,0.63)
(1.77,0.13)
(d)

Composite Two-dimensional Translations

$$P' = T(t_{2x}, t_{2y}) \cdot \{T(t_{1x}, t_{1y}) \cdot P\}$$

$$P' = \{T(t_{2x}, t_{2y}) \cdot T(t_{1x}, t_{1y})\} \cdot P$$

composite transformation matrix

$$\begin{bmatrix} 1 & 0 & t_{2x} \\ 0 & 1 & t_{2y} \\ 0 & 0 & 1 \end{bmatrix} \cdot \left[\begin{bmatrix} 1 & 0 & t_{1x} \\ 0 & 1 & t_{1y} \\ 0 & 0 & 1 \end{bmatrix}\right] = \left[\begin{bmatrix} 1 & 0 & t_{1x} + t_{2x} \\ 0 & 1 & t_{1y} + t_{2y} \\ 0 & 0 & 1 \end{bmatrix}\right]$$

$$T(t_{2x}, t_{2y}) \cdot T(t_{1x}, t_{1y}) = T(t_{1x} + t_{2x}, t_{1y} + t_{2y})$$

Composite Two-dimensional Rotations

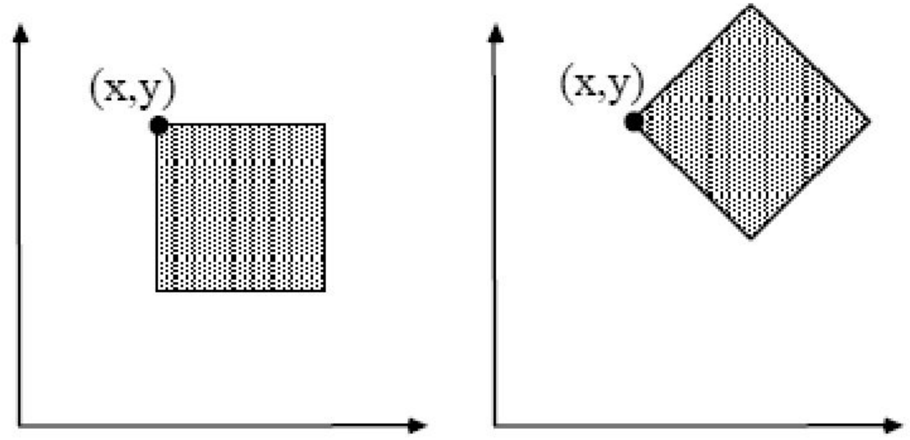
$$P' = R(\theta_2) \cdot \{R(\theta_1) \cdot P\}$$

$$P' = \{R(\theta_2) \cdot R(\theta_1)\} \cdot P$$

composite transformation matrix

$$R(\theta_2) \cdot R(\theta_1) = R(\theta_1 + \theta_2)$$

- Rotation Of ø Degrees About Point (x,y)
  - Translate (x,y) to origin
  - Rotate by ø
  - Translate origin to (x,y)



Composite Two-dimensional Scaling

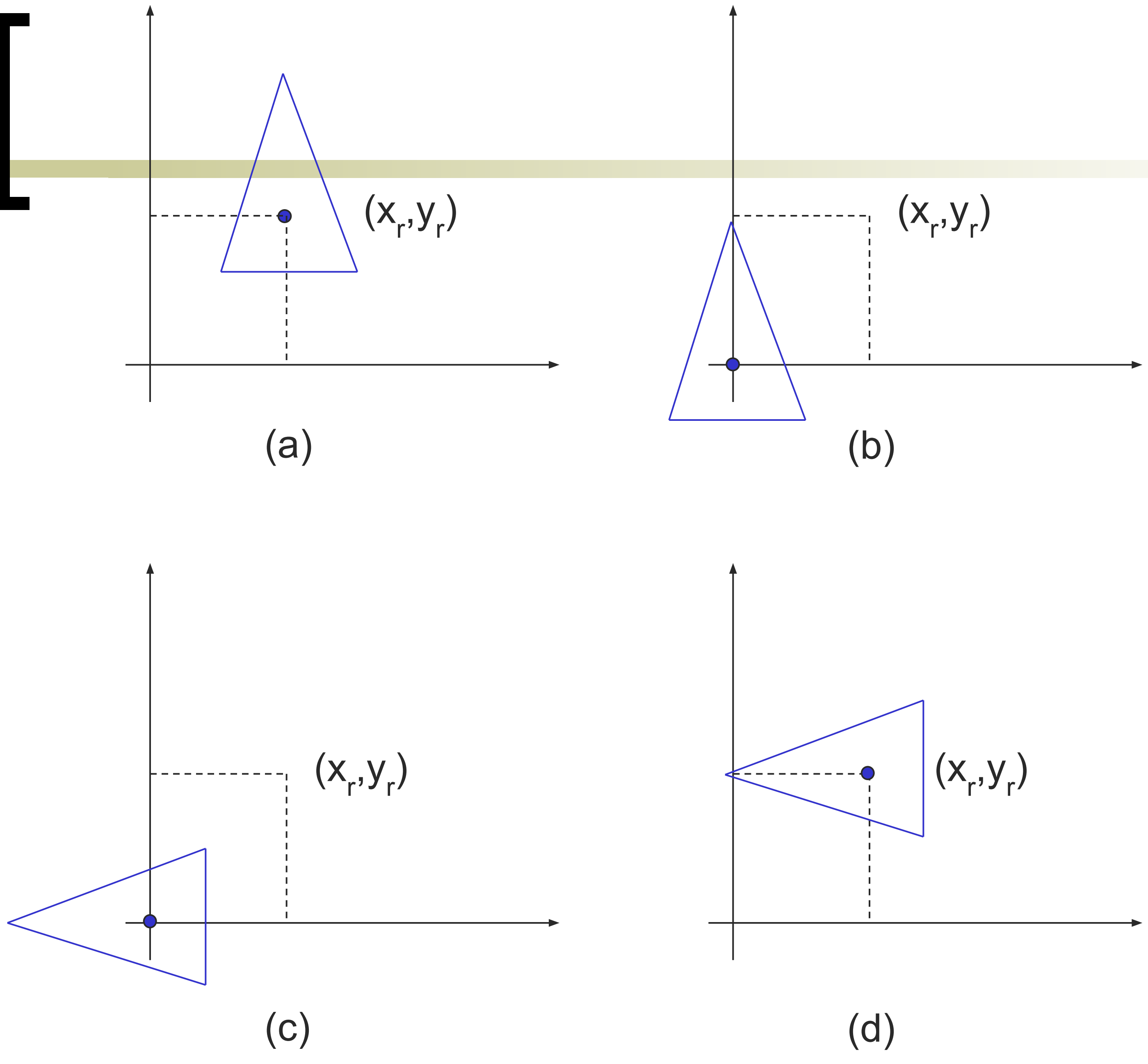
composite transformation matrix

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

$$S(s_{2x}, s_{2y}) \cdot S(s_{1x}, s_{1y}) = S(s_{1x} \cdot s_{2x}, s_{1y} \cdot s_{2y})$$

General Two-dimensional Pivot-Point Rotation

- A transformation sequence for rotating an object about a sepcified pivot point using the rotation matrix **R**(θ).
  - Translate the object so that the pivot-point position is moved to the coordinate origin.
  - Rotate the object about the coordinate origin.
  - Translate the object so that the pivot point is returned to its original position.



(xr,yr)
(a)
(xr,yr)
(b)
(xr,yr)
(c)
(xr,yr)
(d)

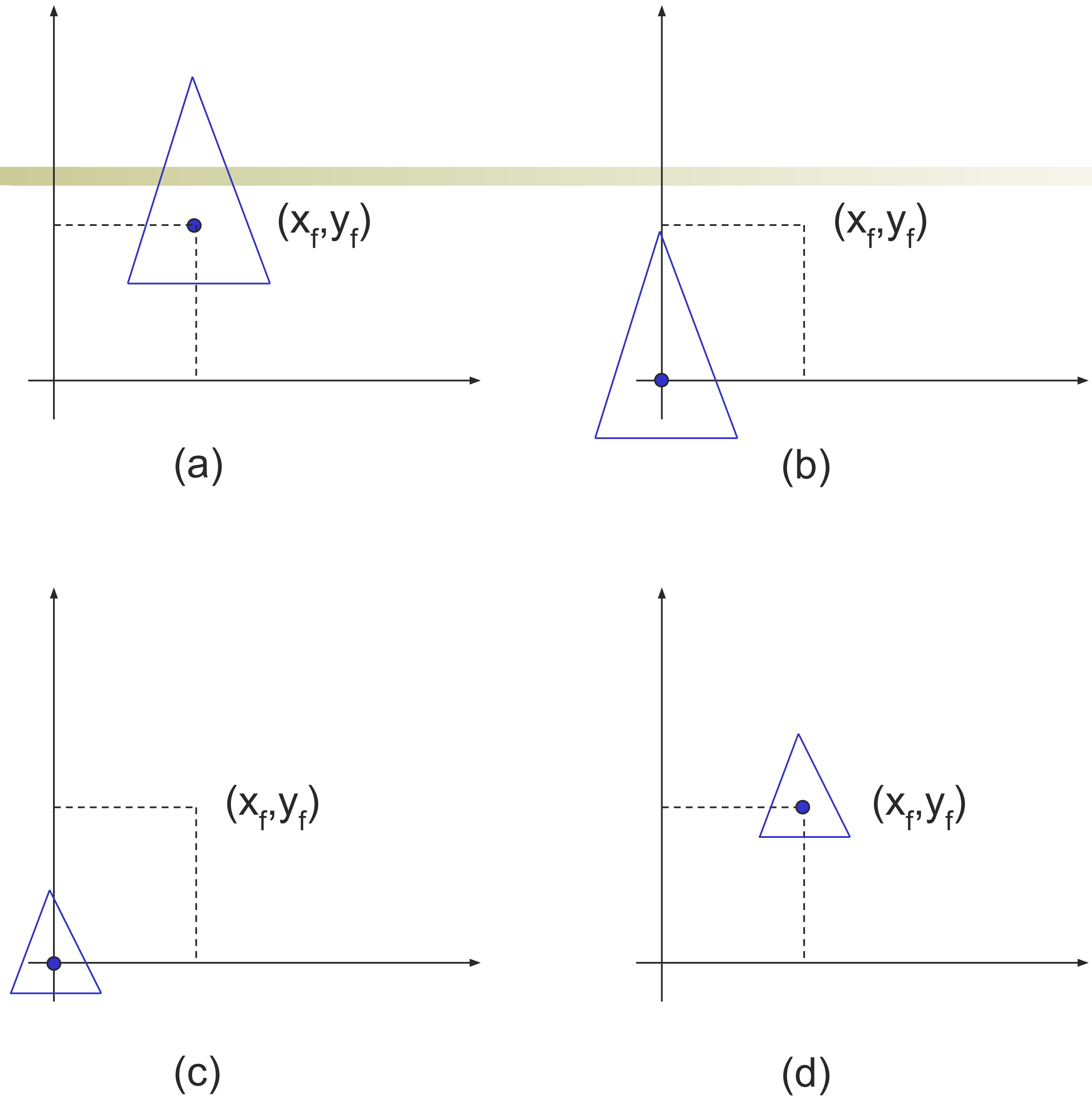
$$\begin{bmatrix} 1 & 0 & x_r \\ 0 & 1 & y_r \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_r \\ 0 & 1 & -y_r \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta & -\sin\theta & x_r(1-\cos\theta)+y_r\sin\theta \\ \sin\theta & \cos\theta & y_r(1-\cos\theta)-x_r\sin\theta \\ 0 & 0 & 1 \end{bmatrix}$$

$$T(x_r, y_r) \cdot R(\theta) \cdot T(-x_r, -y_r) = R(x_r, y_r, \theta)$$

General Two-dimensional Fixed-Point Scaling

- A transformation sequence to produce a two-dimensional scaling with respect to a selected fixed position (x_f, y_f).
  - Translate the object so that the fixed point coincides with the coordinate origin.
  - Scale the object with respect to the coordinate origin.
  - Use the inverse of the translation in step (1) to return the object to its original position.


(x_f,y_f)
(a)
(x_f,y_f)
(b)
(x_f,y_f)
(c)
(x_f,y_f)
(d)

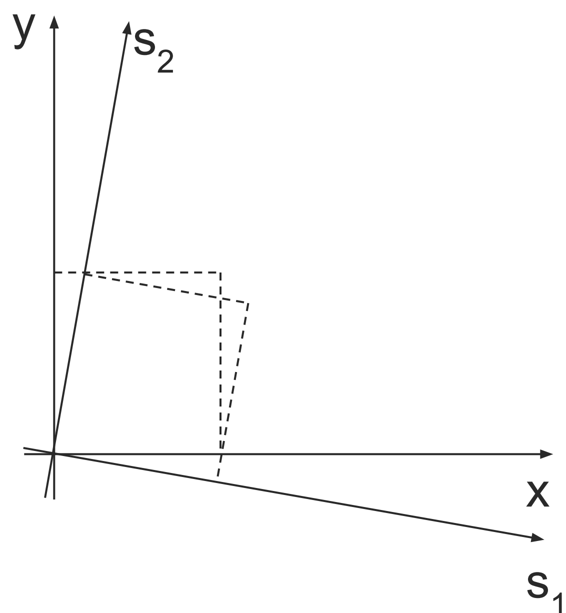
$$\begin{bmatrix} 1 & 0 & x_f \\ 0 & 1 & y_f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_f \\ 0 & 1 & -y_f \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} s_x & 0 & x_f(1-s_x) \\ 1 & s_y & y_f(1-s_y) \\ 0 & 0 & 1 \end{bmatrix}$$

$$T(x_f, y_f) \cdot S(s_x, s_y) \cdot T(-x_f, -y_f) = S(x_f, y_f, (s_x, s_y))$$

General Two-dimensional Scaling Directions

- We can scale an object in other directions by rotating the object to align the desired scaling directions with the coordinate axes before applying the scaling transformation.
- Suppose we want to apply scaling factors with values specified by parameters s_1 and s_2 in the directions shown in fig.



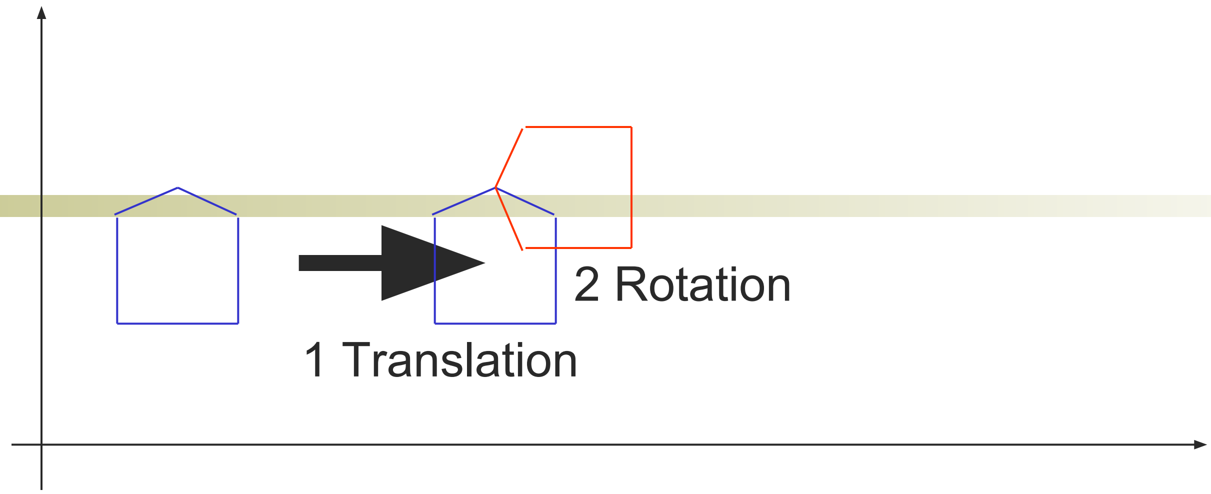
The composite matrix resulting from the product of

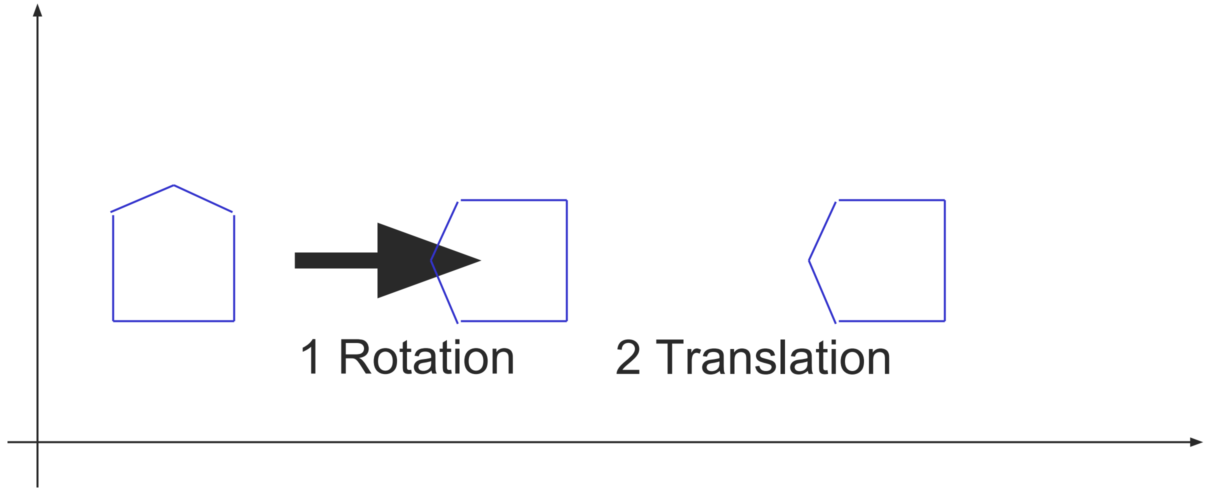
- rotation so that the directions for s_1 and s_2 coincide with the x and y axes
- scaling transformation $S(s_1,s_2)$
- opposite rotation to return points to their original orientations

$$\mathrm{R}^{-1}(\theta)\cdot S(s_1,s_2)\cdot R(\theta)=\begin{bmatrix} s_1 Cos^2\theta+s_2 Sin^2\theta & (s_2-s_1)Cos\theta Sin\theta & 0 \\ (s_2-s_1)Cos\theta Sin\theta & s_1 Sin^2\theta+s_2 Cos^2\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matrix Concatenation Properties

- Transformation products may not be commutative.
- The matrix product M_2M_1 is not equal to M_1M_2.
- This means that if we want to translate and rotate an object, we must be careful about the order in which the composite matrix is evaluated.
- Reversing the order in which a sequence of transformations is performed may affect the transformed position of an object.


2 Rotation
1 Translation


1 Rotation
2 Translation

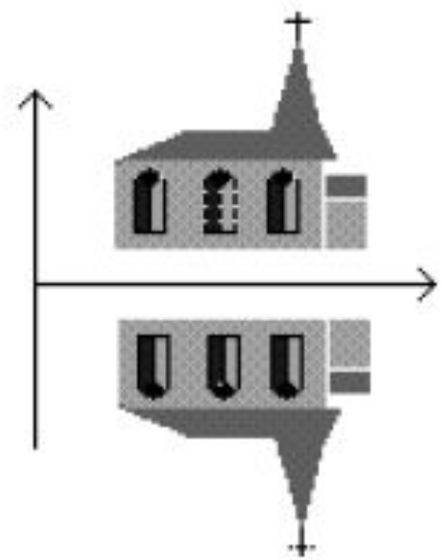
5.5 Other Two-Dimensional Transformations

Reflection

- For a two-dimensional reflection, the image is generated relative to an axis of reflection by rotating the object 180º about the reflection axis.

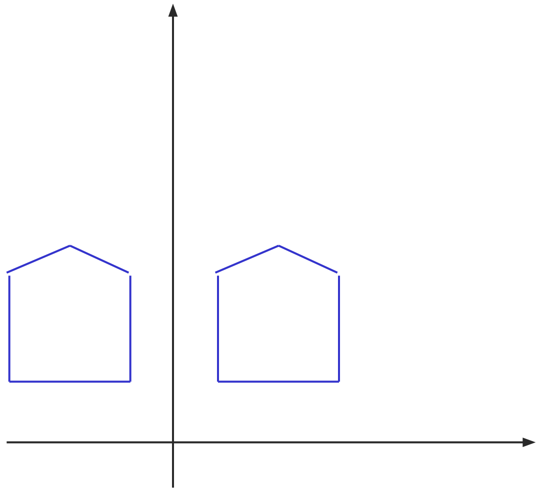
Reflection about the line y=0 (the x axis)

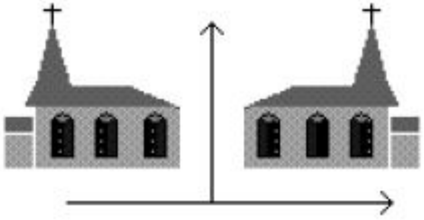
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



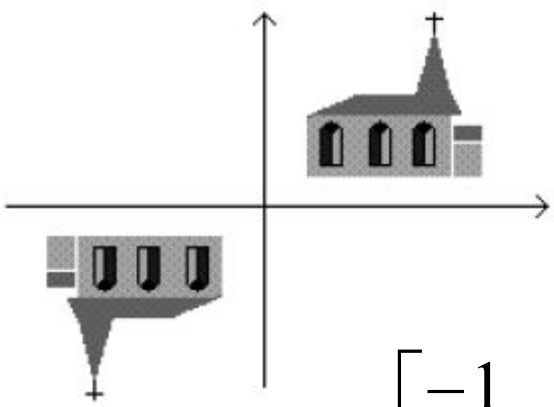
Reflection about the line x=0 (the y axis)

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

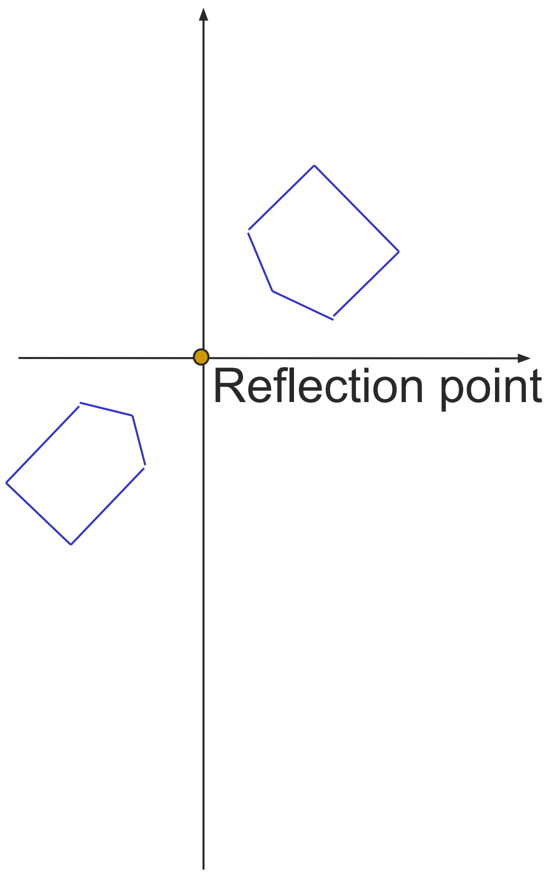




Reflection about any reflection point in the *xy* plane



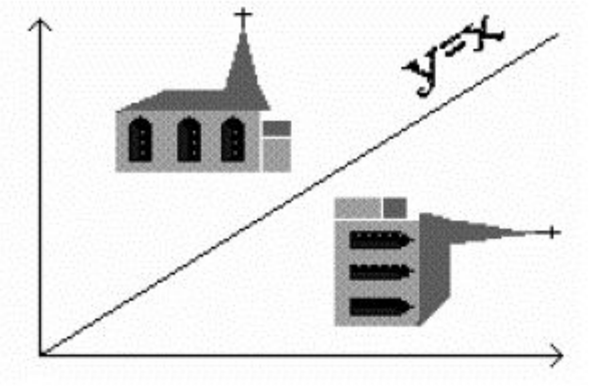
$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Reflection about the reflection axis y=x

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

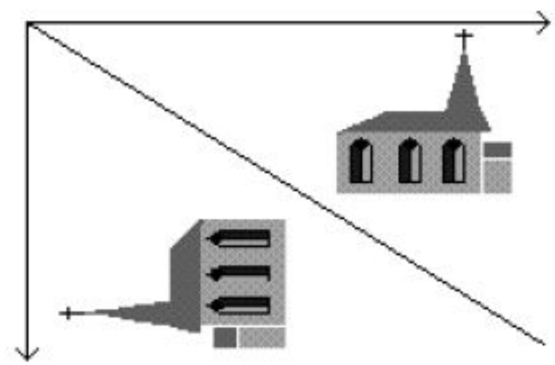
1. Rotate the line y=x with respect to the original through a 45 angle
2. Reflection with respect to the x axis
3. Rotate the line y=x black to its original position with a counterclockwise rotation through 45



Reflection about the reflection axis y=-x

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

1. Clockwise rotate the line y=-x with respect to the original through a 45 angle
2. Reflection with respect to the y axis
3. Rotate the line y=-x black to its original position with a counterclockwise rotation through 45



y = -x

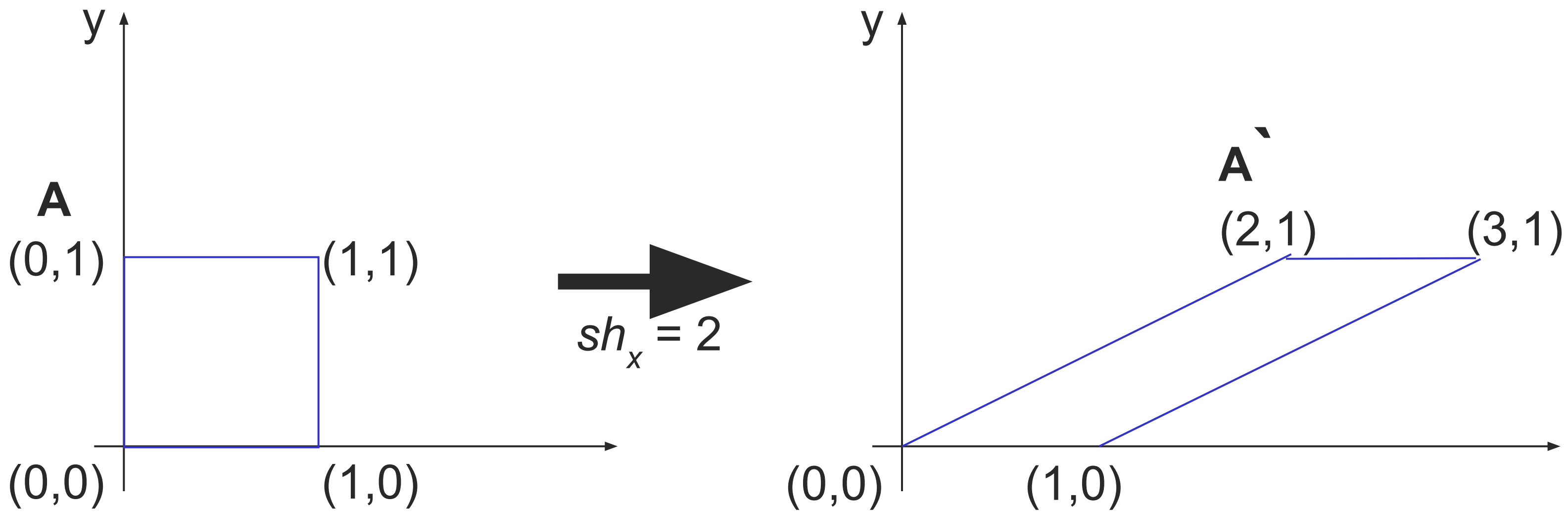
Shear

- A transformation that distorts the shape of an object such that the transformed shape appears as if the object were composed of internal layers that had been caused to slide over each other is called a shear.

An x-direction shear

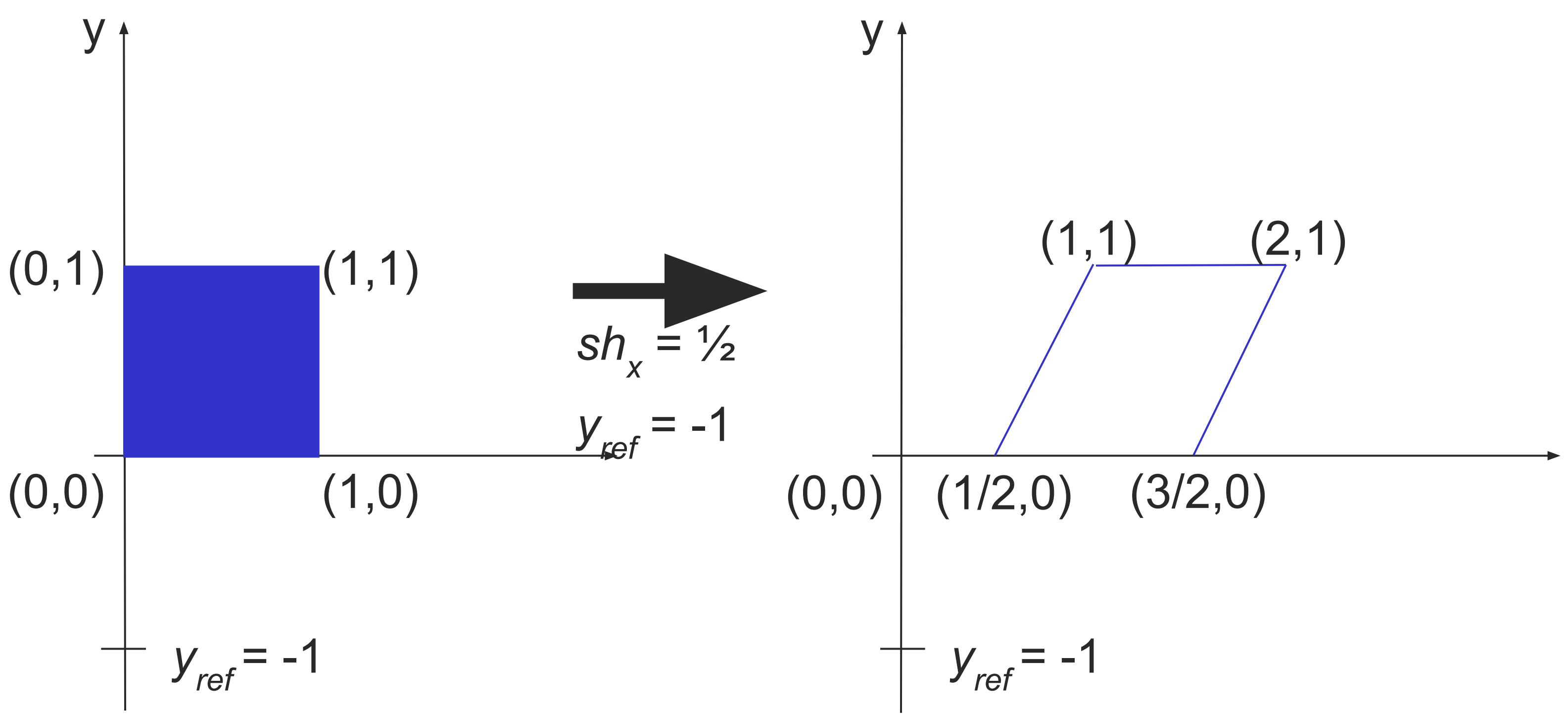
$$\begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textit{transformation matrix}$$

$$\mathbf{A}` = \begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \mathbf{A}$$

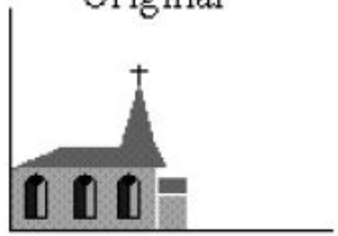


- We can generate *x*-direction shears relative to other reference lines

$$\begin{bmatrix} 1 & sh_x & -sh_x \cdot y_{ref} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textit{transformation matrix}$$



Original



X-Shear

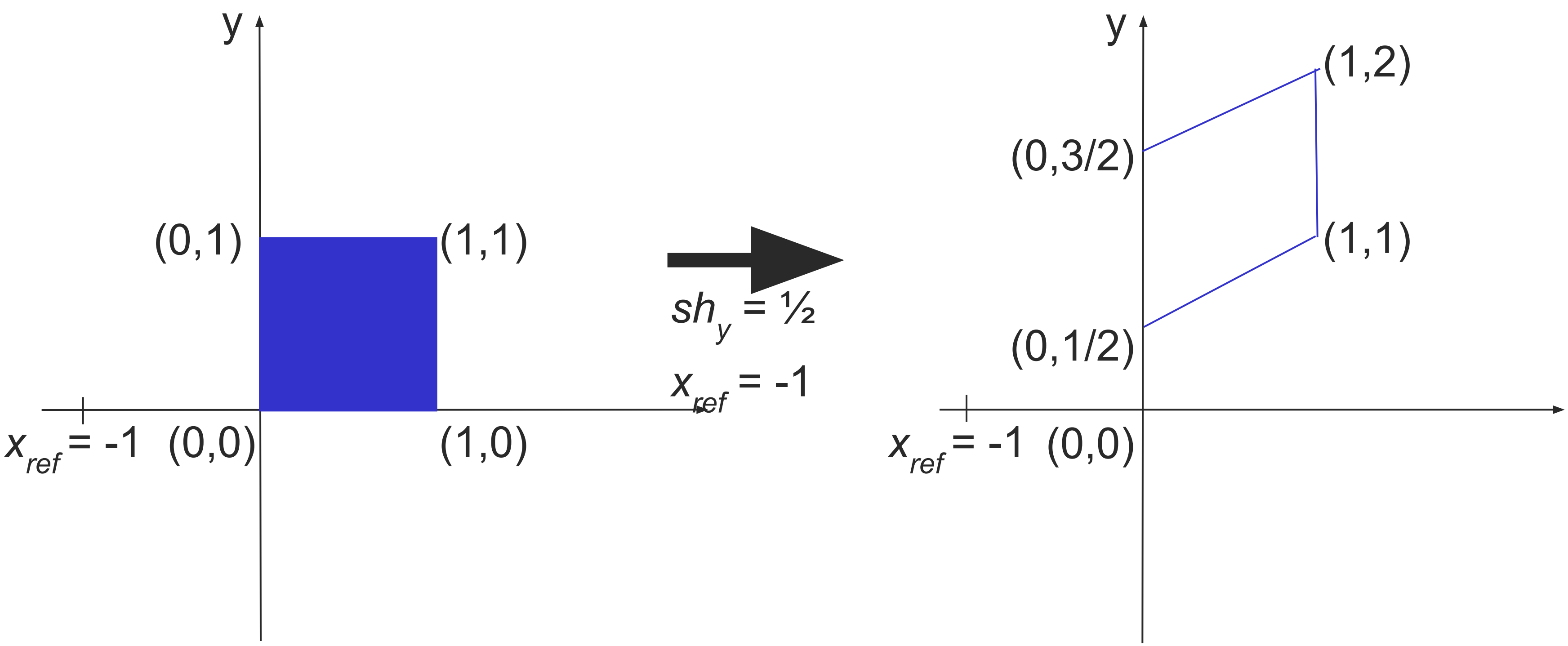
- Shear operations can be expressed as sequences of basic transformations.
- The x-direction shear matrix

$$\begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

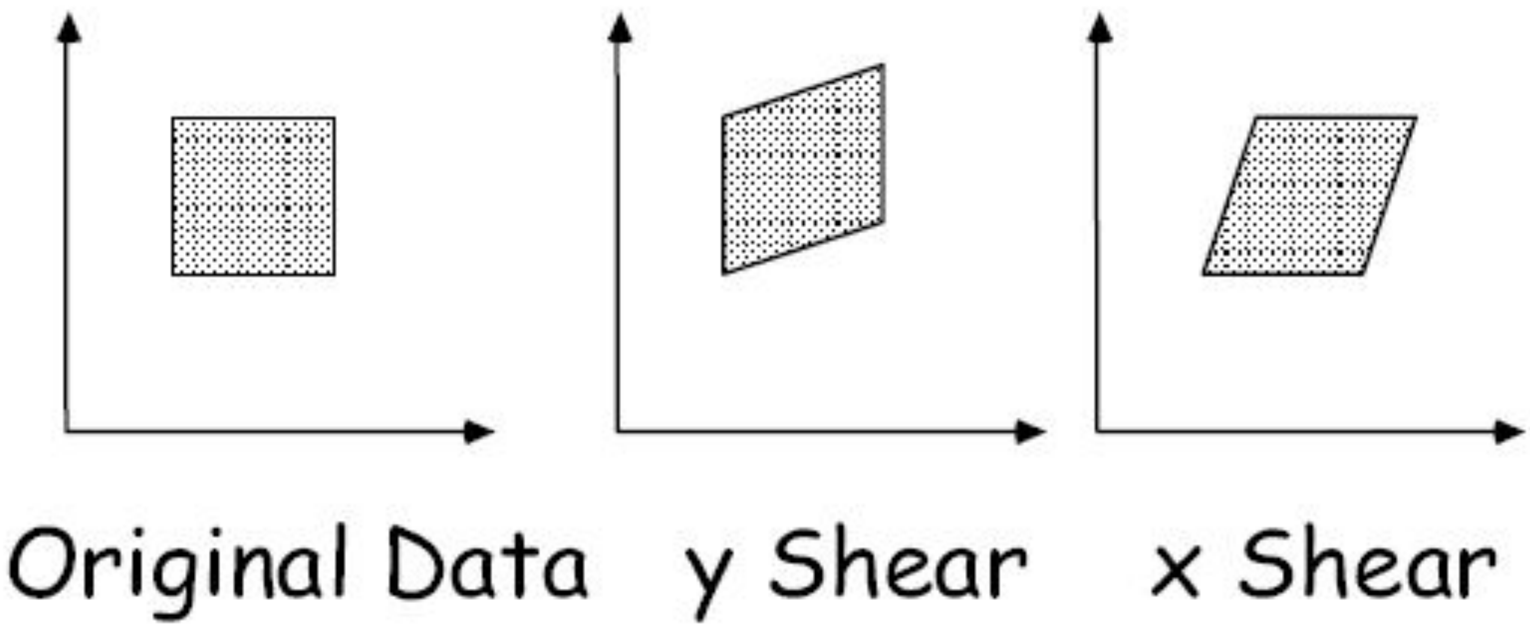
can be represented as a composite transformation involving a series of rotation and scaling metrices.

- A *y*-direction shears relative to the line $x=x_{ref}$ is generated with

$$\begin{bmatrix} 1 & 0 & 0 \\ sh_y & 1 & -sh_y \cdot x_{ref} \\ 0 & 0 & 1 \end{bmatrix} \quad \textit{transformation matrix}$$




Original
Y-Shear


Original Data
y Shear
x Shear