2D Geometric Transformations

Chapter 5
Intro. to Computer Graphics
Spring 2009, Y. G. Shin

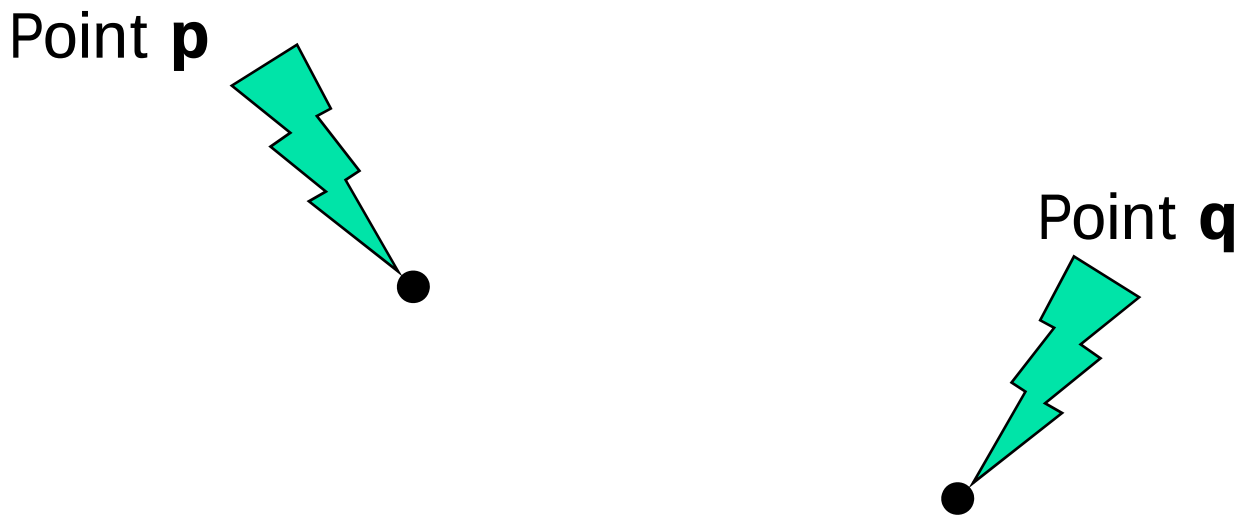
Introduction

- We deal a lots with objects defined in 2-D and 3-D worlds in computer graphics
- All objects have shape, position and orientation
- **Geometry** is the study of the relationships among objects
  - Modeling + rendering : computer programs that describe these objects and how light bounces around to illuminate them in order to calculate the final pixel values on the display.
- Two types of geometry operations
  - Vertex operations - Operate on individual vertexes
  - Primitive operations - Operate on all the vertexes of a primitive

Coordinate-Free Geometry

- **CFG** - A style of expressing geometric objects and relations that do not rely on any specific coordinate system
- Representing geometry in terms of coordinates can frequently lead to physical confusion

Example of coordinate-dependence



- What is the “sum” of these two positions ?

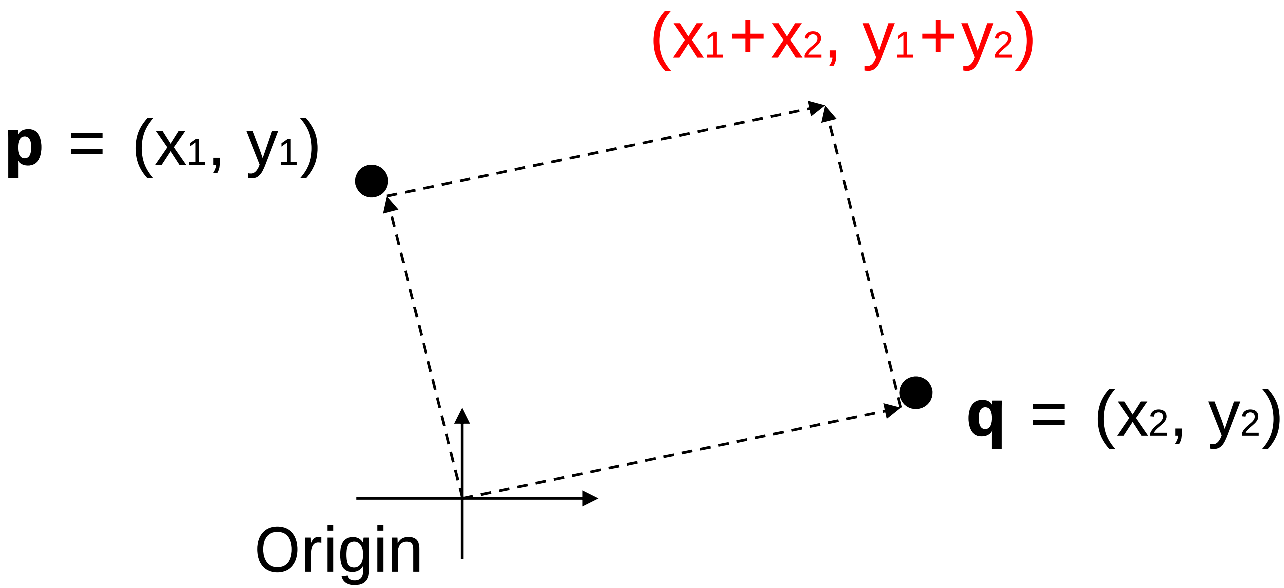
If you assume coordinates, ...

$\mathbf{p} = (x_1, y_1)$

$\mathbf{q} = (x_2, y_2)$

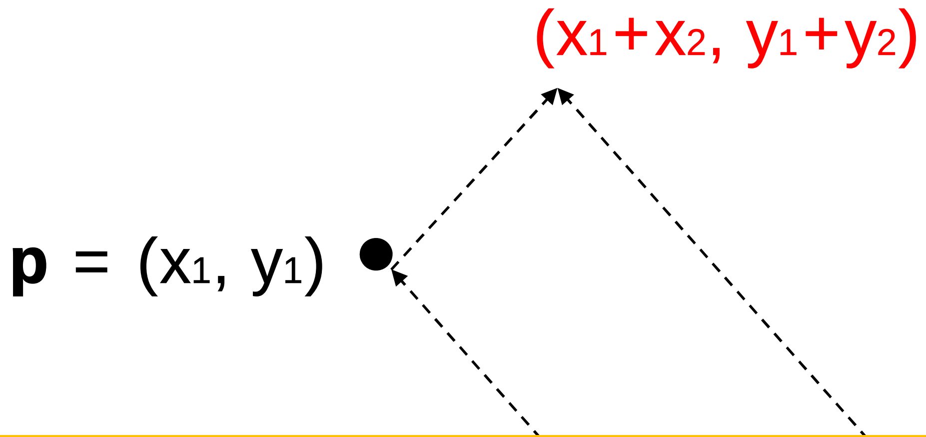
- The sum is (x_1+x_2, y_1+y_2)
  - Is it correct ?
  - Is it geometrically meaningful ?

If you assume coordinates, ...



- Vector sum
  - (x_1, y_1) and (x_2, y_2) are considered as vectors from the origin to **p** and **q**, respectively.

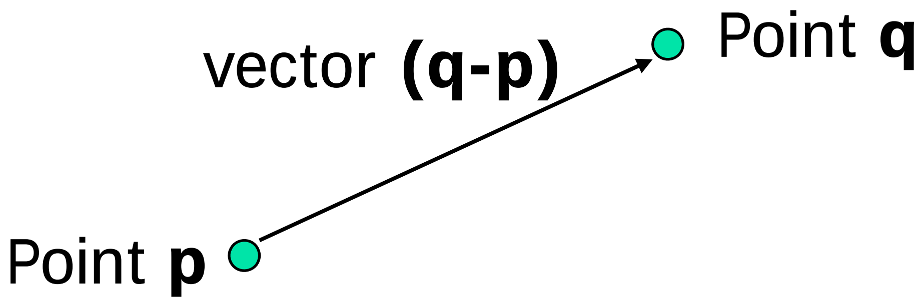
If you select a different origin, ...



✓ The geometric relationship of the result of adding two points depends on the coordinate system. There is no clear geometric interpretation for adding two points.

- If you choose a different coordinate frame, you will get a different result

Points and Vectors



- A **scalar** is just a real number.
- A ***point*** is a location in space. It does not have any intrinsic coordinates.
- A ***vector*** is a direction and a magnitude. It may be specified as the difference between two points.

CFG Operations

- magnitude of a vector : $\left|\vec{v}\right|$
- point-vector addition: $p_1 + \vec{v}_1 = p_2$

$$\vec{v}_1 = p_2 - p_1$$

- vector addition : $\vec{v}_1 + \vec{v}_2 = \vec{v}_3$
- vector scaling : $\alpha\vec{v}_1$
- dot product : $\vec{v}_1 \cdot \vec{v}_2 = \left|\vec{v}_1\right|\left|\vec{v}_2\right|\cos\theta$

CFG Operations

- cross product: $\left|\vec{v}_1 \times \vec{v}_2\right| = \left|\vec{v}_1\right|\left|\vec{v}_2\right| \sin\theta$
- Linear combination of vectors :

$$\sum_i \alpha_i \vec{v}_i = \vec{v}$$

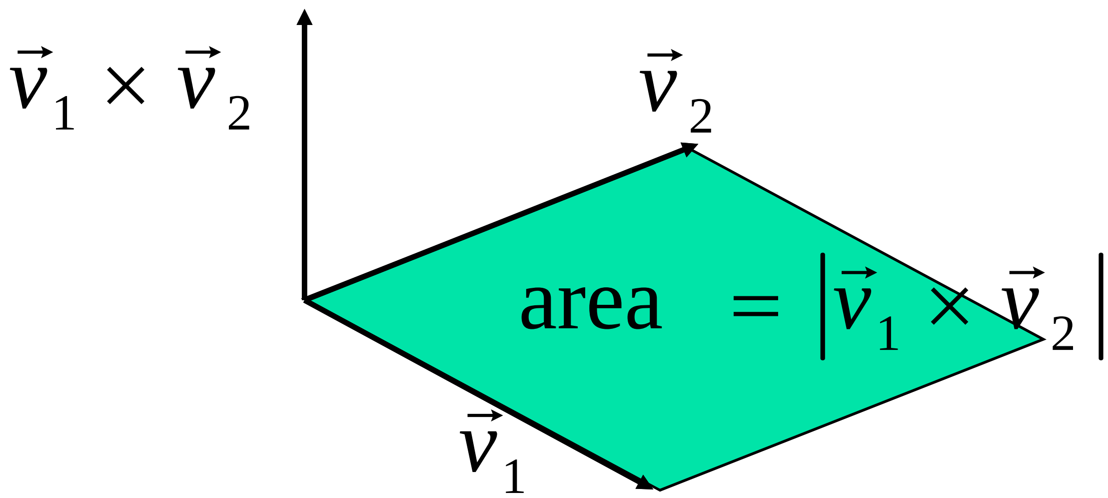
- Affine combination of points :

$$\sum_i \alpha_i p_i = p, \text{ if } \sum_i \alpha_i = 1$$

- $\sum_i \alpha_i p_i = \vec{v}, \text{ if } \sum_i \alpha_i = 0$

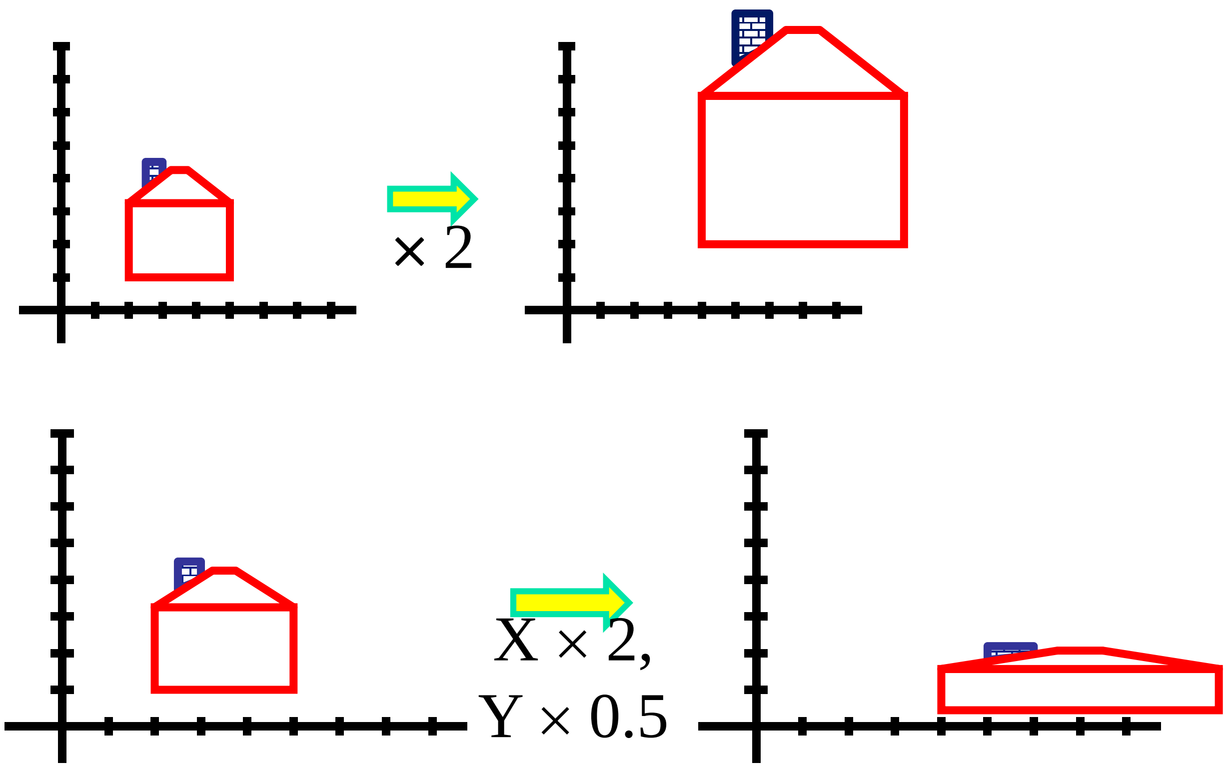
The Cross Product

The length is also equal to the area of the parallelogram whose sides are given by $\vec{v}_1$ and $\vec{v}_2$



Various Geometry Operations – Scaling

- *Scaling* a coordinate means multiplying each of its components by a scalar



Scaling

- Scaling operation: $x' = ax$

  $y' = by$

- Or, in matrix form:

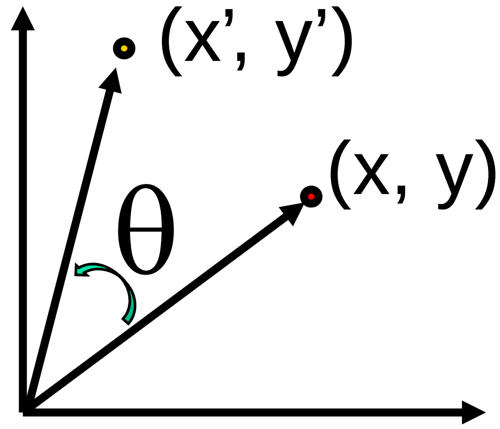
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\textit{scaling matrix } S} \begin{bmatrix} x \\ y \end{bmatrix}$$

scaling matrix S

Rotation

- This is easy to capture in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$



- 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*
- What is the inverse transformation?
  - Rotation by $-\theta$
  - For rotation matrices, det(R) = 1 so $\mathbf{R}^{-1} = \mathbf{R}^T$

Translation

- Translation: $$\begin{bmatrix} t_x \\ t_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_x + x \\ t_y + y \end{bmatrix}$$

  - Not linear transformation

$$T(P) + T(Q) = \begin{bmatrix} x_1 + t_x \\ y_1 + t_y \end{bmatrix} + \begin{bmatrix} x_2 + t_x \\ y_2 + t_y \end{bmatrix} = \begin{bmatrix} x_1 + x_2 + 2t_x \\ y_1 + y_2 + 2t_y \end{bmatrix}$$

$$T(P + Q) = \begin{bmatrix} x_1 + x_2 + t_x \\ y_1 + y_2 + t_y \end{bmatrix}$$

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

All 2D Linear Transformations

- **2D 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

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Homogeneous Coordinates

- Add a 3rd coordinate to every 2D point
- Cartesian coordinate ⇒ homogeneous coordinate

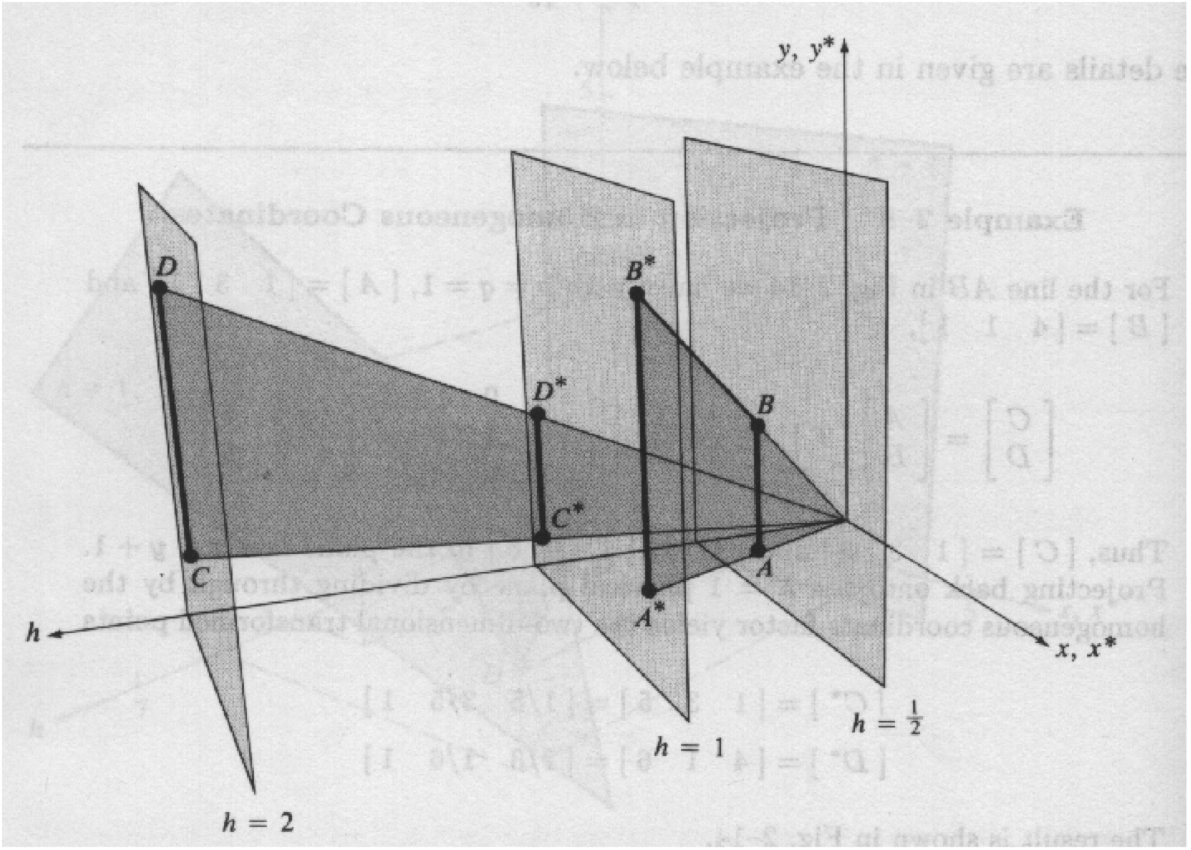
  [x y] ⇒ [x' y' h]

  h : real number

  x = x'/h, y = y'/h

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

Homogeneous coordinates



- (x, y, 0) represents a point at infinity
- (0, 0, 0) is not allowed

Homogeneous Coordinate

- In 2-dimensional spaces:
  - Point : (x, y, 1), Vector : (x, y, 0)
- For example

$(x_1, y_1, 1) + (x_2, y_2, 1) = (x_1+x_2, y_1+y_2, 2)$

point *point* *undefined*

$(x_1, y_1, 1) - (x_2, y_2, 1) = (x_1-x_2, y_1-y_2, 0)$

point *point* *vector*

$(x_1, y_1, 1) + (x_2, y_2, 0) = (x_1+x_2, y_1+y_2, 1)$

point *vector* *point*

- But, it is algebraically inconsistent !!

$$(1,0,0,1) + (1,1,0,1) = (2,1,0,2) = (1,\frac{1}{2},0,1)$$

$$\parallel \qquad \parallel \qquad \nparallel$$

$$(1,0,0,1) + (2,2,0,2) = (3,2,0,3) = (1,\frac{2}{3},0,1)$$

Homogeneous Coordinates

- Convenient coordinate system to represent many useful transformations
- Possible to represent scaling, rotation, and translation in a matrix form
- *Any* sequence of translation, rotation, scale operations can be collapsed into a single homogeneous matrix.

Transformations in Homogeneous Coordinates

- Translation

$$P' = M \cdot P$$

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

  - When we use a row vector P

$$P' = P \cdot M^{-1}$$

Transformations in Homogeneous Coordinates

- Scaling

$$M = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

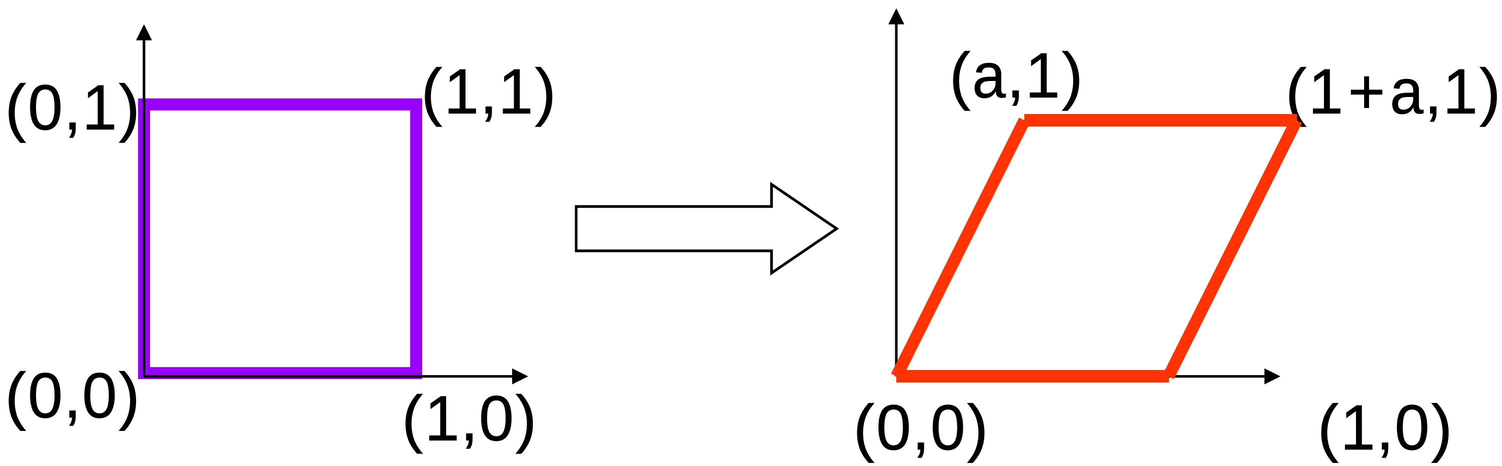
- Rotation

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

Transformations in Homogeneous Coordinates

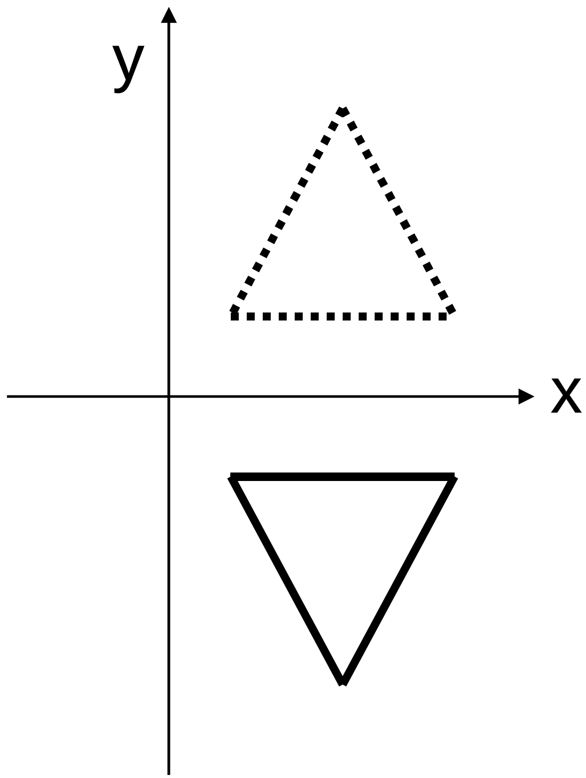
- Shearing
  - shear along x-axis
  - x' = x + ay, y' = y

$$M = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Transformations in Homogeneous Coordinates

- Reflection



$$\text{Ref}(x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Vector and Affine spaces

- A ***vector space*** consists of
  - Set of vectors, together with
  - Two operations: addition of vectors and multiplication of vectors by scalar numbers

- An ***affine space*** consists of
  - Set of points, an associated vector space, and
  - Two operations: the difference between two points and the addition of a vector to a point

Affine Transformation

Affine transformation $T : A_1 \to A_2$

Where A_1, A_2 are affine spaces.

- *T* maps vectors to vectors and points to points

$$T(P+\vec{u}) = T(P) + T(\vec{u})$$

- *T* is linear transformation + translation

$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

Affine Transformations

- **Properties of affine transformations:**
  - Origin does not necessarily map to origin
  - Lines map to lines
  - Parallel lines remain parallel
  - Ratios are preserved
  - Closed under composition
  - Models change of basis

Affine Transformations

- With homogeneous coordinates,
  - Affine transformations are associative.
    - For affine transformations F1, F2, and F3,

      $(F3 \circ F2) \circ F1$ is the same as $F3 \circ (F2 \circ F1)$.

  - Affine transformations are *not commutative.*
    - For affine transformations F1 and F2,

      $F2 \circ F1$ is NOT the same as $F1 \circ F2$.

Projective Transformations

- A transformation that maps lines to lines (but does not necessarily preserve parallelism)
- The affine transformations are a subset of the projective transformations.

$$\begin{bmatrix} x' \\ y' \\ h \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ p & q & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [x \quad y \quad (px+qy+1)^t]$$

We get $x^* = x'/h$

$$y^* = y'/h$$

Projective Transformations

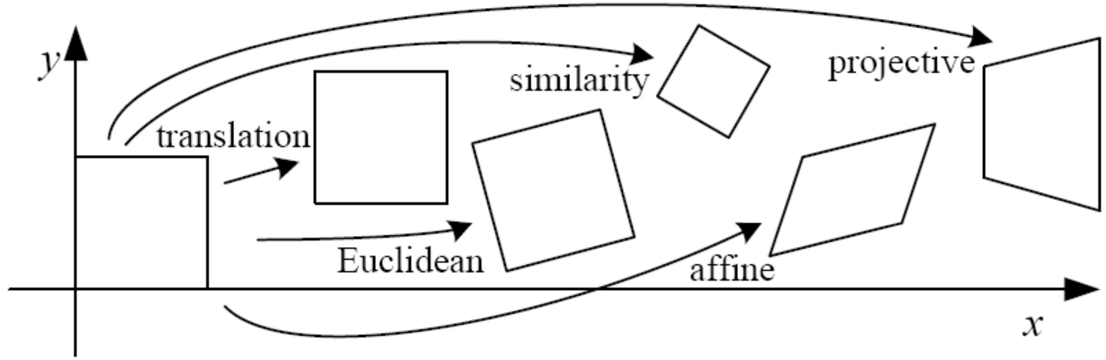
- **Properties of projective transformations:**
  - Origin does not necessarily map to origin
  - Lines map to lines
  - Parallel lines do not necessarily remain parallel
  - Ratios are not preserved
  - Closed under composition
  - Models change of basis

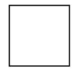
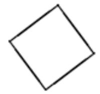
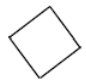
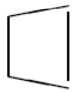
Rigid Transformations

- A ***rigid transformation*** *T* is a mapping between affine spaces
  - *T* maps vectors to vectors, and points to points
  - *T* preserves distances between all points
  - *T* preserves cross product for all vectors (to avoid reflection)
- Preserves angles and lengths
- In 3-spaces, T can be represented as

$$T(\mathbf{p}) = \mathbf{R}_{3\times3}\mathbf{p}_{3\times1} + \mathbf{T}_{3\times1}$$

2D image transformations

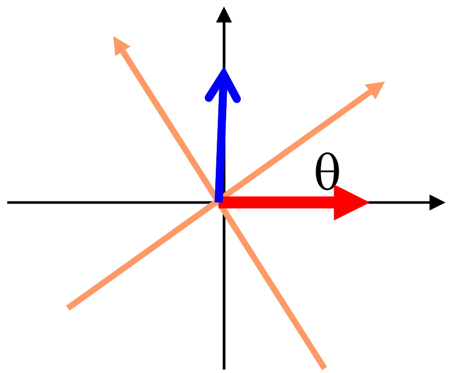


Name	Matrix	# D.O.F.	Preserves:	Icon
translation	$\begin{bmatrix} \boldsymbol{I} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	2	orientation $+ \cdots$	
rigid (Euclidean)	$\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	3	lengths $+ \cdots$	
similarity	$\begin{bmatrix} s\boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	4	angles $+ \cdots$	
affine	$\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$	6	parallelism $+ \cdots$	
projective	$\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

Orthogonal Matrix

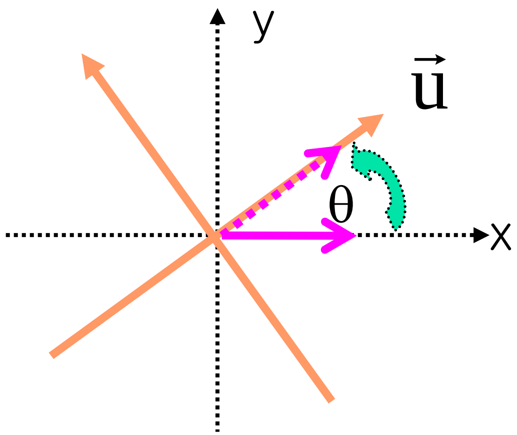
- Rotation matrix is a special orthogonal matrix

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



- R is normalized: the squares of the elements in any row or column sum to 1.
- R is orthogonal: the dot product of any pair of rows or any pair of columns is 0.
- The rows of R represent the coordinates in the original space of unit vectors along the coordinate axes of the rotated space.
- The columns of R represent the coordinates in the rotated space of unit vectors along the axes of the original space.

Orthogonal Matrix



$$\vec{u}_x = \frac{\vec{u}}{|\vec{u}|} = [\cos\theta \quad \sin\theta]$$

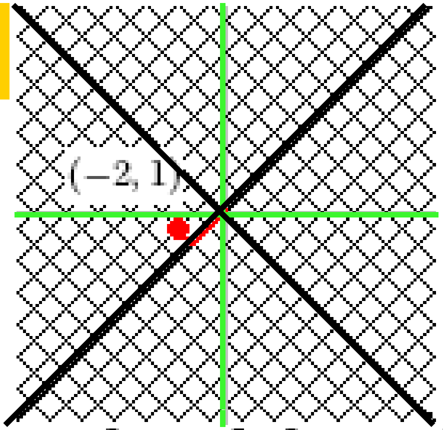
$$\vec{u}_y = [-\sin\theta \quad \cos\theta]$$

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

Coordinate Systems

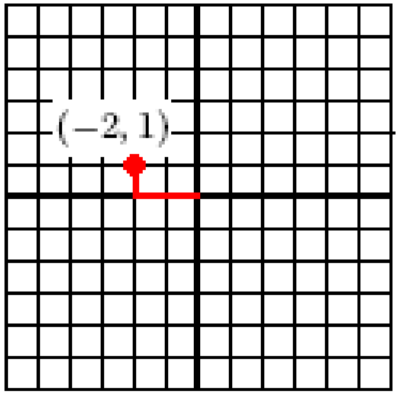
- **Every coordinate system is specified by a basis**
- **Linear transformations as a change of bases**

C_1



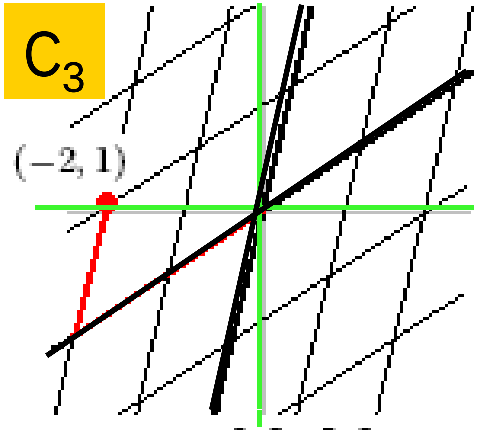
Basis: $\begin{bmatrix} \cos 45^\circ \\ \sin 45^\circ \end{bmatrix}, \begin{bmatrix} -\sin 45^\circ \\ \cos 45^\circ \end{bmatrix}$

C_2



Basis: $\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

C_3



Basis: $\begin{bmatrix} 3 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 \\ 4 \end{bmatrix}$

The point (-2,1) in C_3 is (-5,1) in C_2

$$\begin{bmatrix} 3 & 1 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} -2 \\ 1 \end{bmatrix} = \begin{bmatrix} -5 \\ 0 \end{bmatrix}$$

The point (-2,1) in C_2 is (?,?) in C_3 ⇐

$$\begin{bmatrix} 3 & 1 \\ 2 & 4 \end{bmatrix}^{-1} \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

Composite Transformation

- Translations

$$\begin{bmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y1} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{x1} + t_{x2} \\ 0 & 1 & t_{y1} + t_{y2} \\ 0 & 0 & 1 \end{bmatrix}$$

$$M = T(t_{x2}, t_{y2}) \cdot T(t_{x1}, t_{y1})$$

$$= T(t_{x1}+t_{x2}, t_{y1}+t_{y2})$$

Composite Transformation

- Scaling

$$\begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{x1} \cdot s_{x2} & 0 & 0 \\ 0 & s_{y1} \cdot s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M = S(s_{x2}, s_{y2}) \cdot S(s_{x1}, s_{y1})$$
$$= S(s_{x1} \cdot s_{x2}, s_{y1} \cdot s_{y2})$$

Composite Transformation

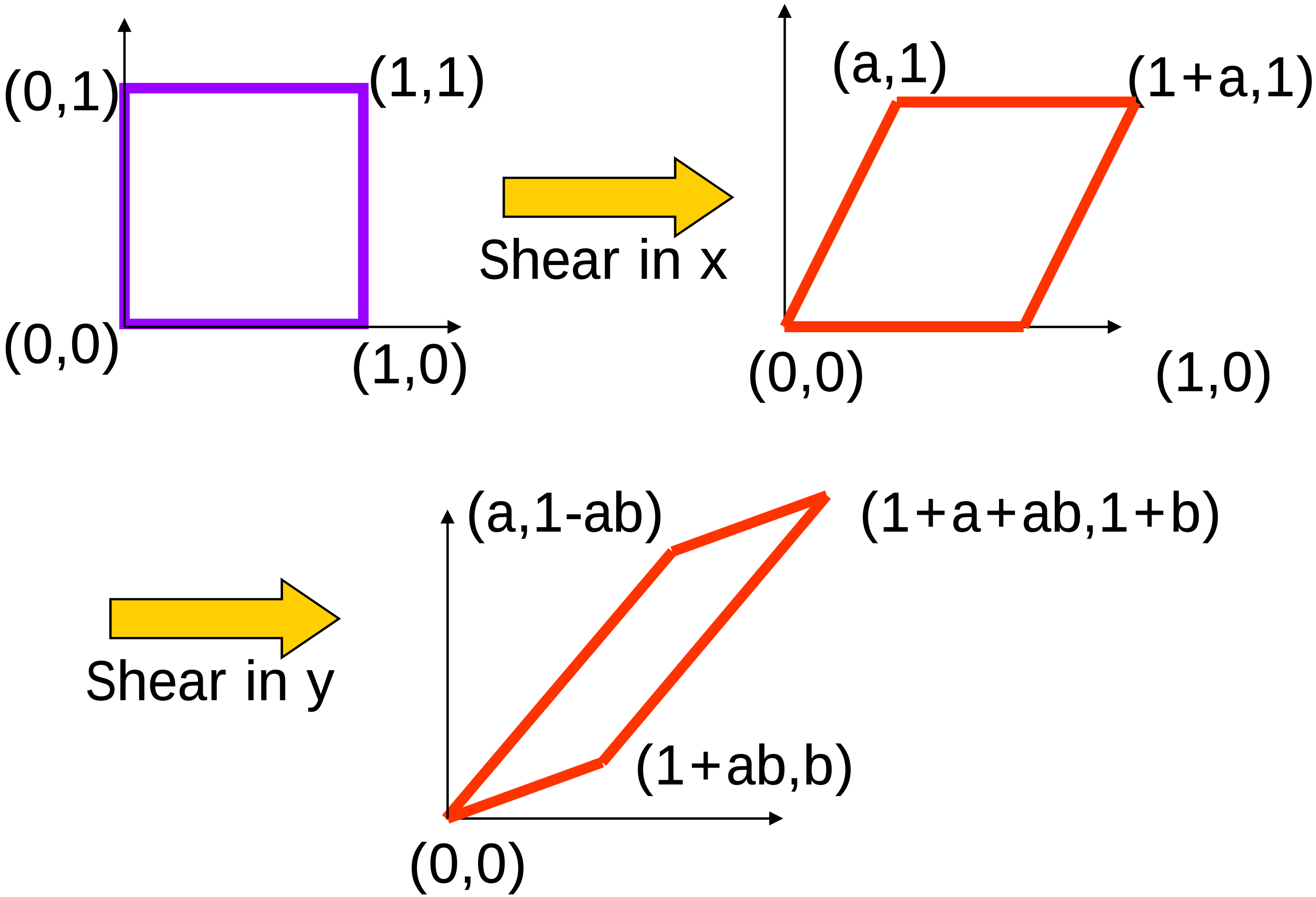
- Rotations

$$\begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

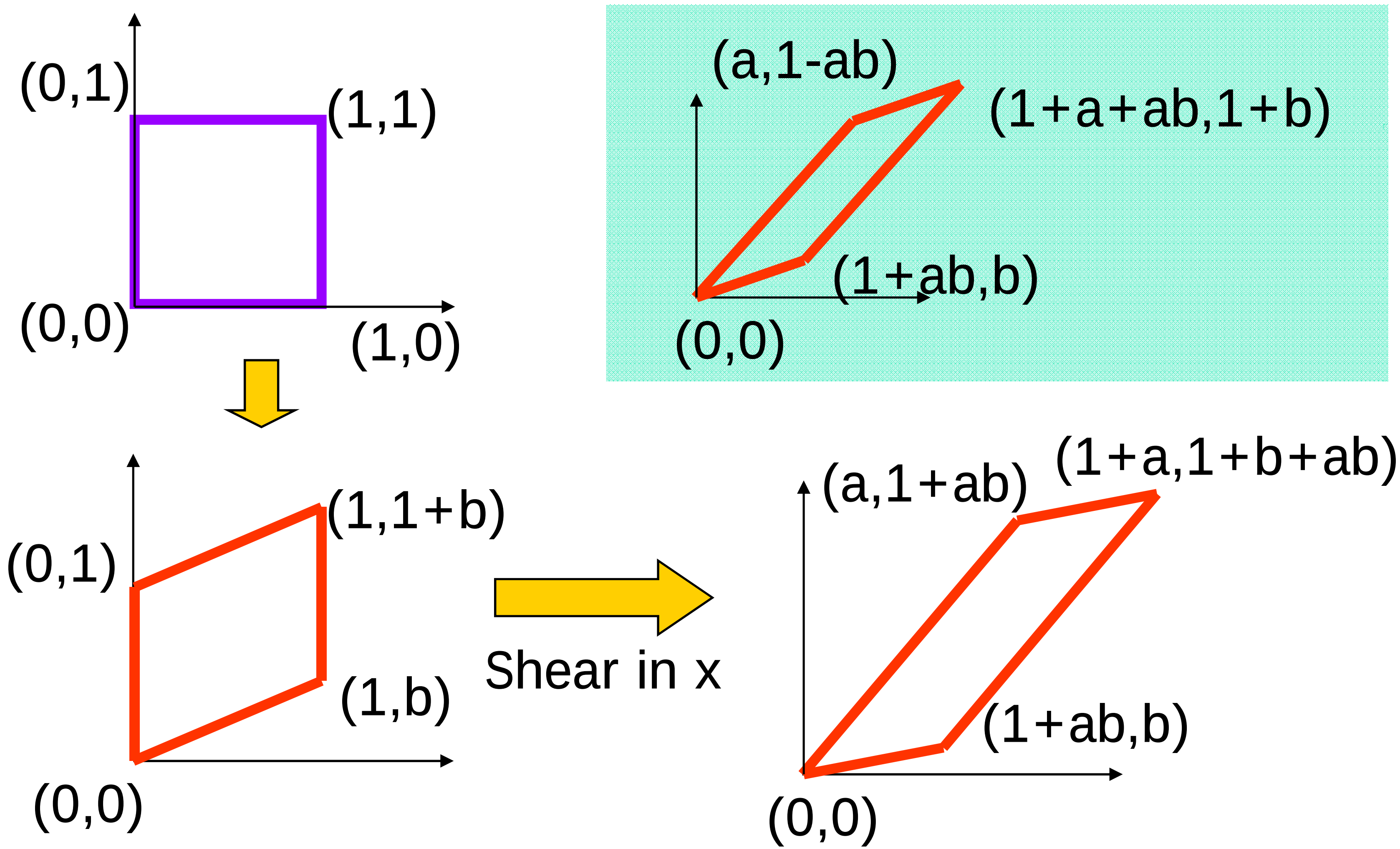
$$= \begin{bmatrix} \cos(\theta_2+\theta_1) & -\sin(\theta_2+\theta_1) & 0 \\ \sin(\theta_2+\theta_1) & \cos(\theta_2+\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

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

Shear in x then in y



Shear in y then in x

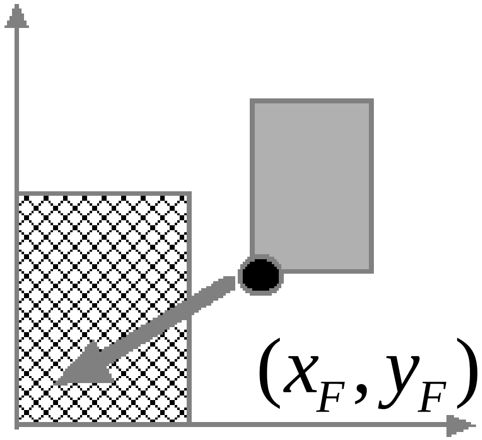


Scaling Relative to a Fixed Point

$$T(x_F, y_F) \cdot S(s_x, s_y) \cdot T(-x_F, -y_F)$$

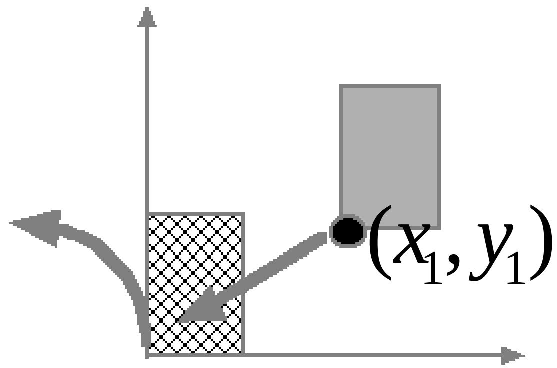
$$= \begin{bmatrix} 1 & 0 & x_F \\ 0 & 1 & y_F \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_F \\ 0 & 1 & -y_F \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} s_x & 0 & (1-s_x) \cdot x_F \\ 0 & s_y & (1-S_y) \cdot y_F \\ 0 & 0 & 1 \end{bmatrix}$$



Rotation about a Pivot Point

$$T(x_1, y_1) \cdot R(\theta) \cdot T(-x_1, -y_1)$$



Note that

$$T(x_1, y_1)R(\theta)T(-x_1, -y_1) \neq T(x_1, y_1)T(-x_1, -y_1)R(\theta)$$

General Transformation Eq.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & c & e \\ b & d & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Efficiency Considerations

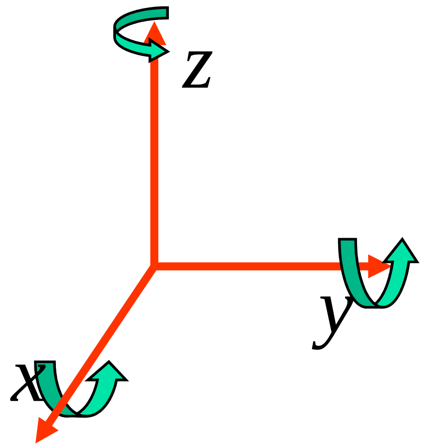
- x' = ax + cy + e, y' = bx + dy + f

  Need x : 4   + : 4
- Without concatenation: increased number of calculations

  ⇒ Apply composite transformation after concatenation of each transformation matrix.

3D Transformation

- *We can extend to 2D to 3D by considering one more dimension with very similar equations.*



- *Remember*
  - *Rigid body transformation*
  - *Affine transformation*
  - *Projective (Perspective) transformation*