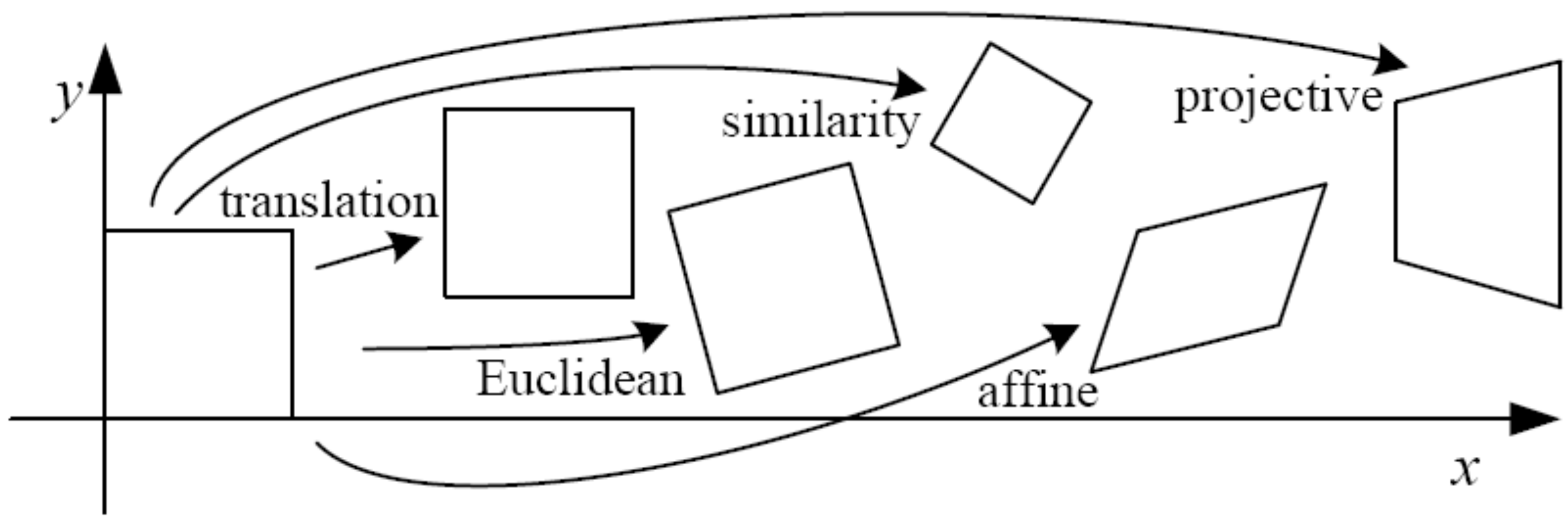
# CSC 479 Introduction to Computer Vision




Lecture 9: Image Transformations

Bei XIao

# Announcements

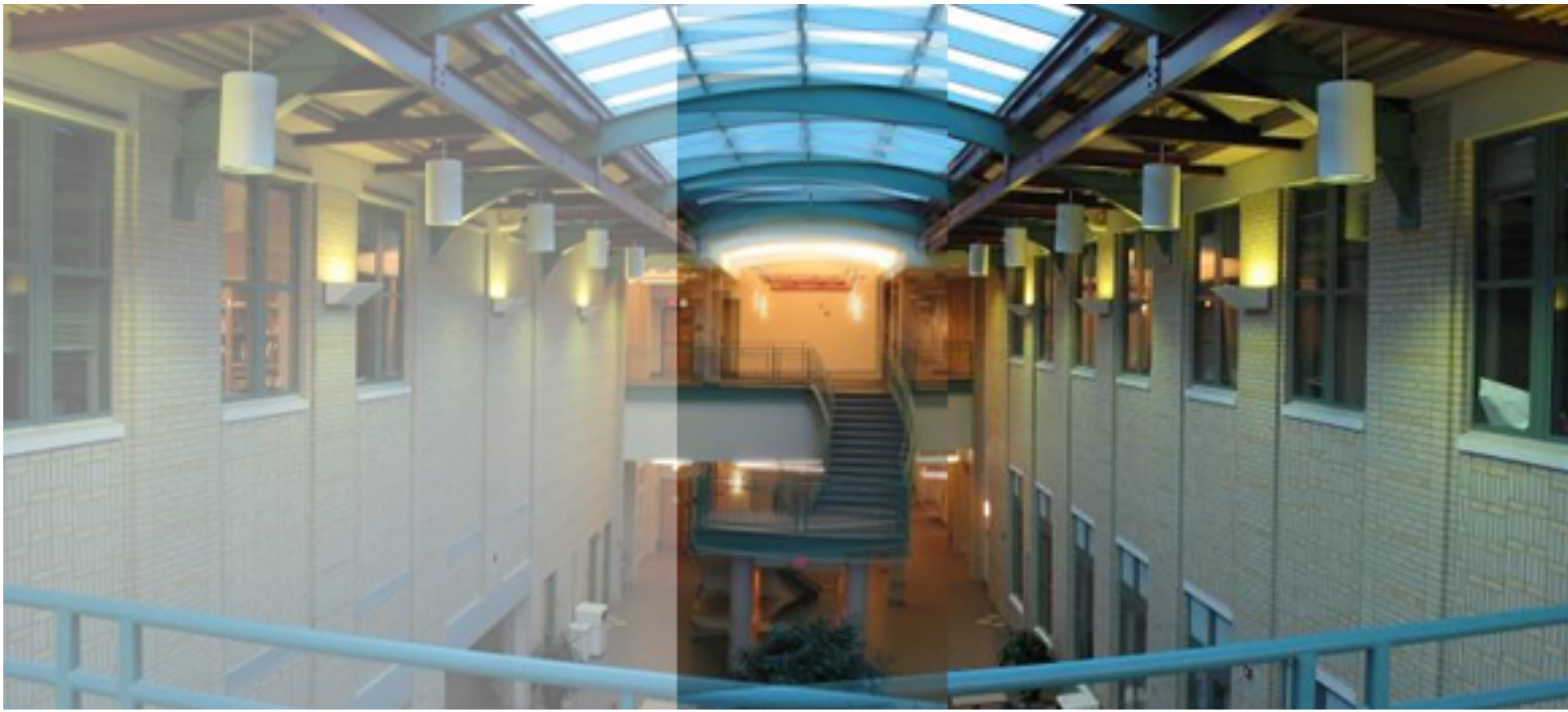
- Project 5 is out today.
- Due right after April 22nd.

- If you are confused, come to office hour

# Reading

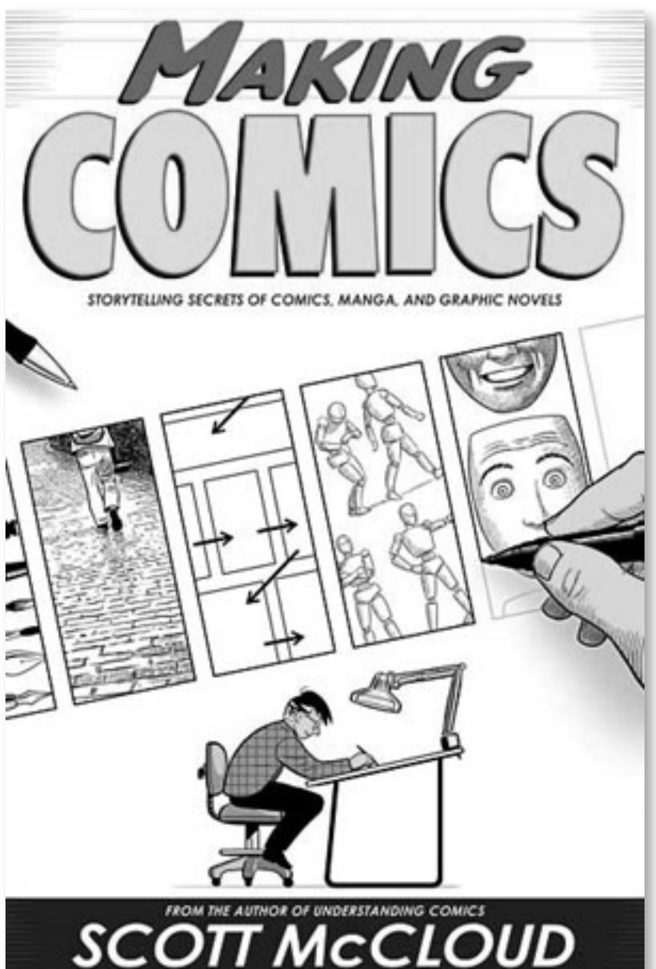
- Szeliski: Chapter 2.1, 6.1, 3.6

# Image alignment



# Augmented Reality

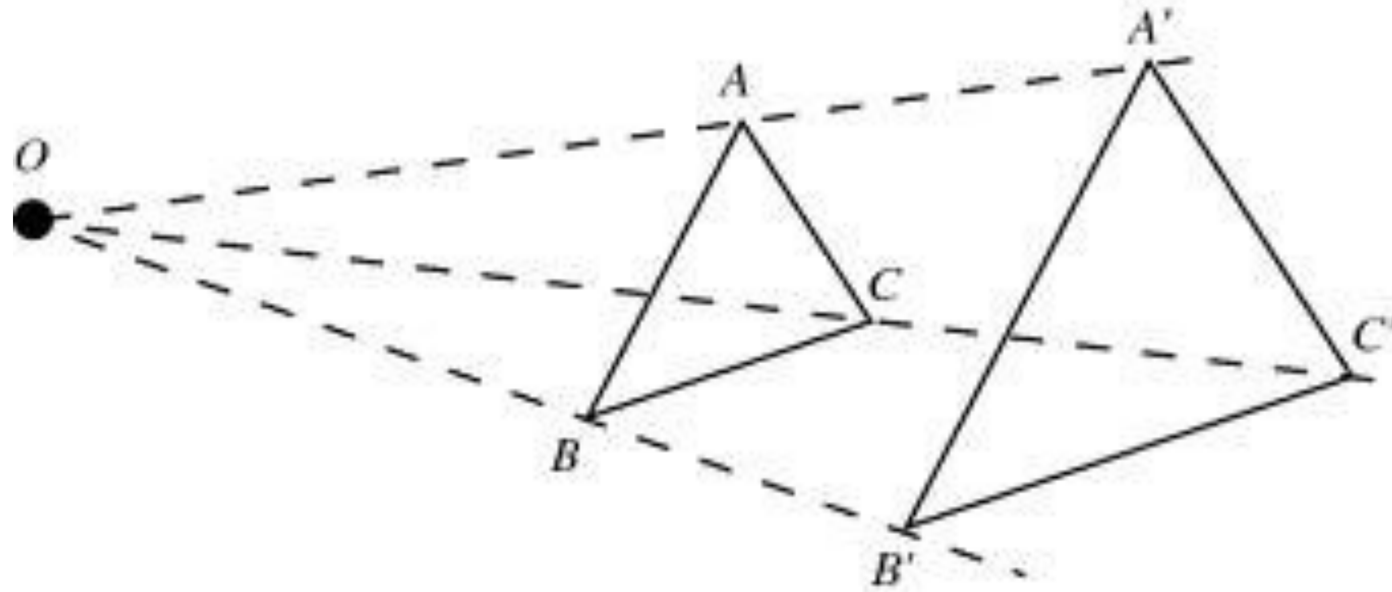
# What is the geometric relationship between these two images?




?

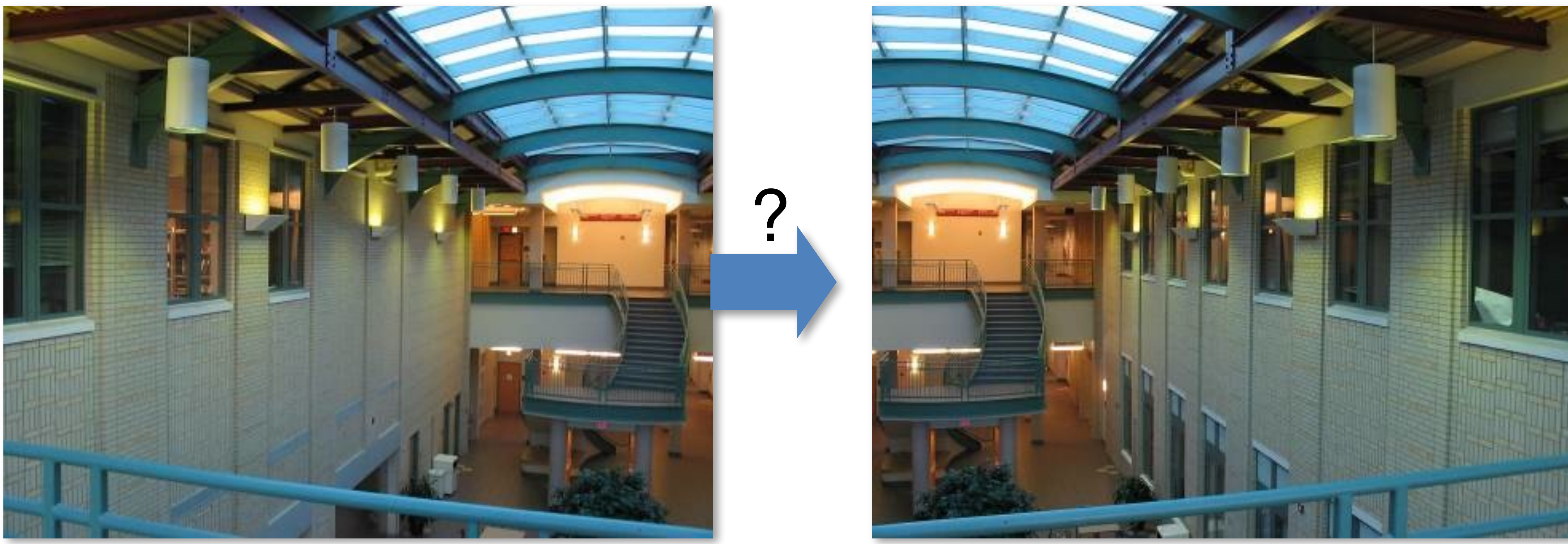

**Answer: Similarity transformation** (translation, rotation, uniform scale)

# similarity transformation




The given two triangles ABC and A'B'C' are similar, since 3 angles of both the triangles are similar and since the size of object has changed, it is a transformation and is transformed to a similar triangle. That is, transforming a given figure to a similar figure is called as similarity transformation.

# What is the geometric relationship between these two images?

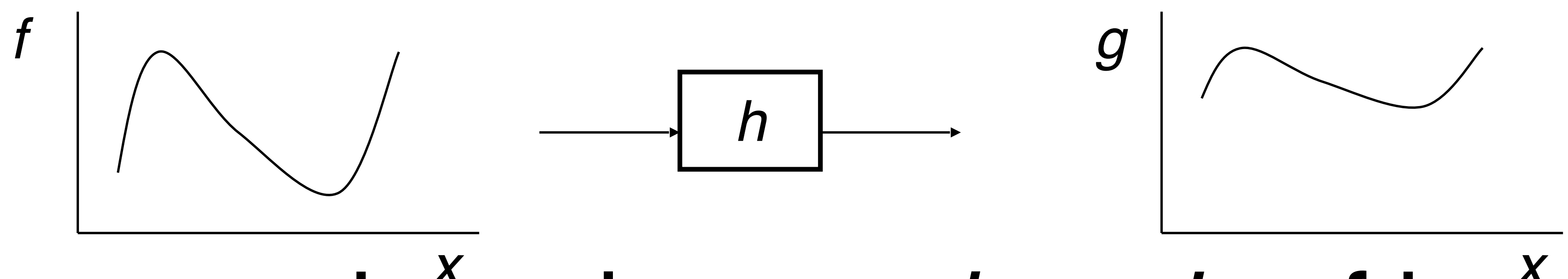


# What is the geometric relationship between these two images?

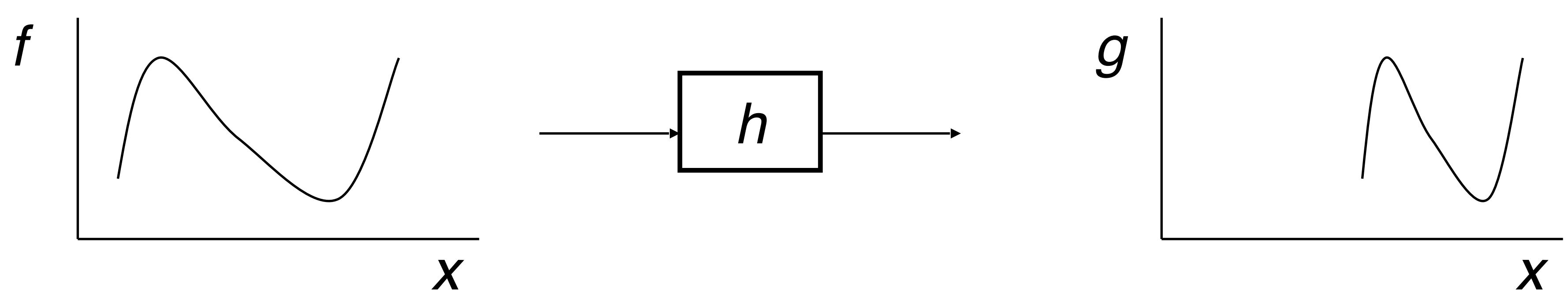
**Very important for creating mosaics**

# Image Warping

- image filtering: change *range* of image
- $$g(x) = h(f(x))$$



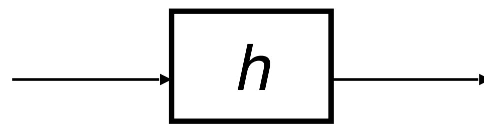

- image warping: change *domain* of image
- $$g(x) = f(h(x))$$



# Image Warping

- image filtering: change *range* of image
- $$g(x) = h(f(x))$$

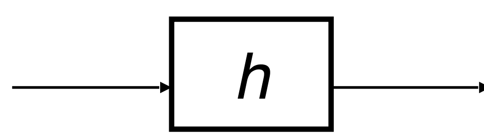
*f*




*g*

- image warping: change *domain* of image
- $$g(x) = f(h(x))$$

*f*




*g*

# Parametric (global) warping

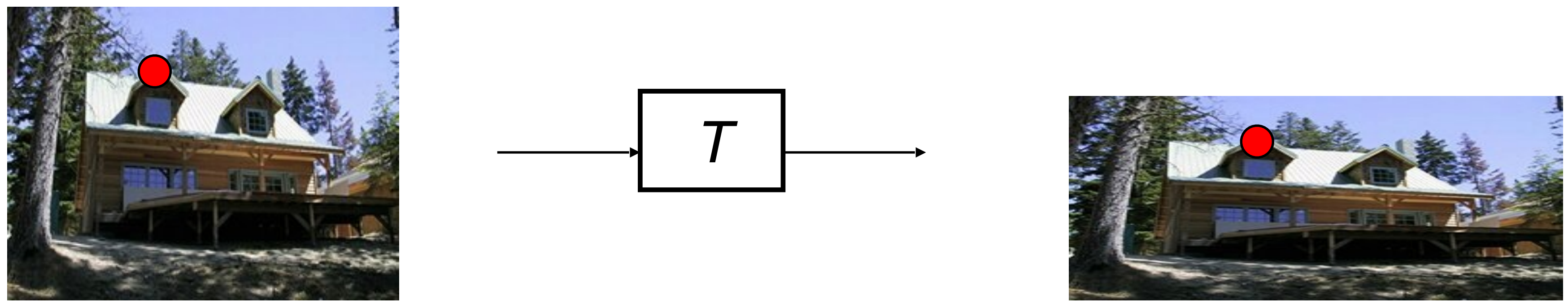
- Examples of parametric warps:

translation

rotation

aspect

# Parametric (global) warping




**p** = (x,y) **p’** = (x’,y’)

- Transformation T is a coordinate-changing machine:

  p’ = *T*(p)

- What does it mean that *T* is global?
  - Is the same for any point p
  - can be described by just a few numbers (parameters)
- Let’s consider *linear* transforms (can be represented by a 2D matrix):

$$\mathbf{p}' = \mathbf{T}\mathbf{p} \qquad \begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \end{bmatrix}$$

# Common linear transformations

- Uniform scaling by *s*:

(0,0)

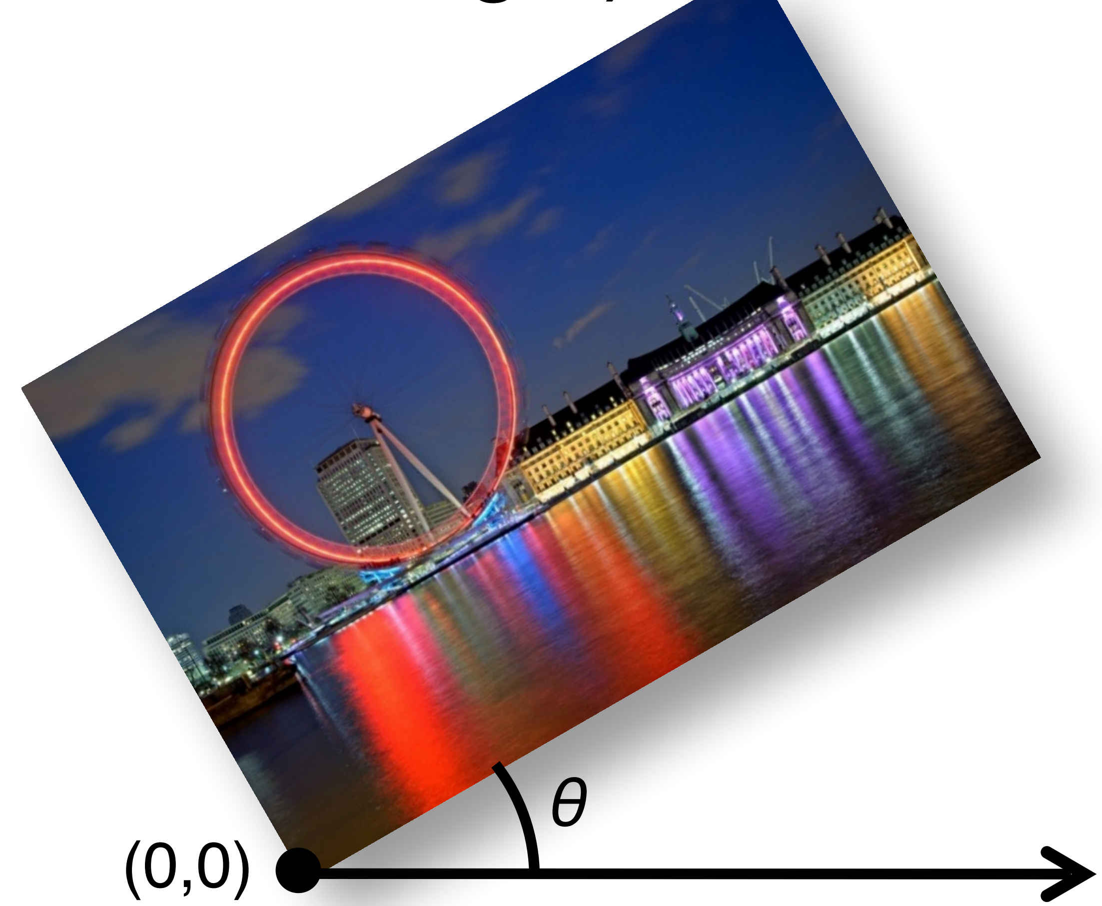
(0,0)

$$\mathbf{S} = \left[ \begin{array}{cc} s & 0 \\ 0 & s \end{array} \right]$$

What is the inverse?

# Common linear transformations

- Rotation by angle $\theta$ (about the origin)


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

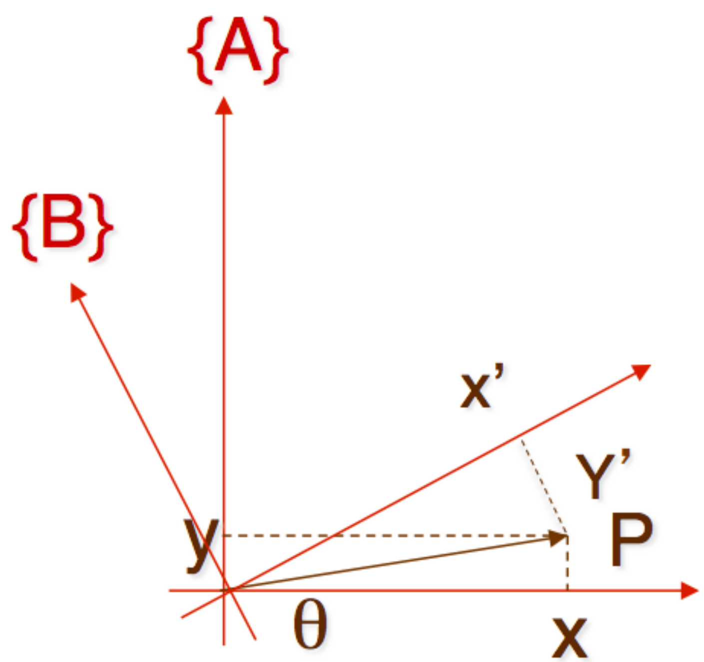
What is the inverse?

For rotations:

$$\mathbf{R}^{-1} = \mathbf{R}^T$$

# Common linear transformations

Counter-clockwise rotation of a coordinate frame by an angle θ



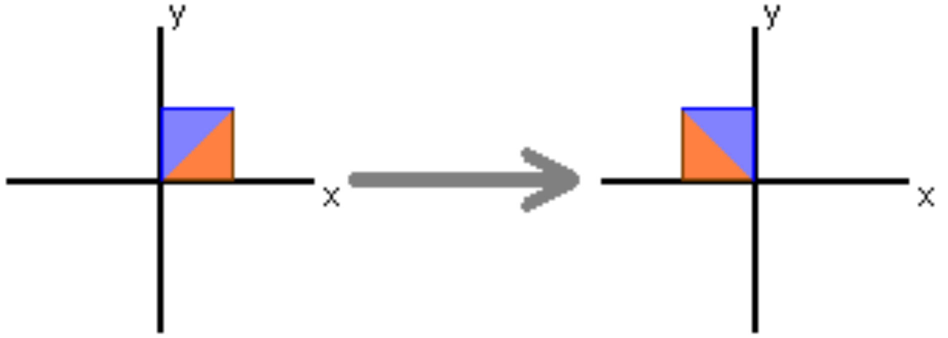

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Counter-clockwise rotation of a coordinate frame attached to a rigid body by an angle θ

# 2x2 Matrices

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

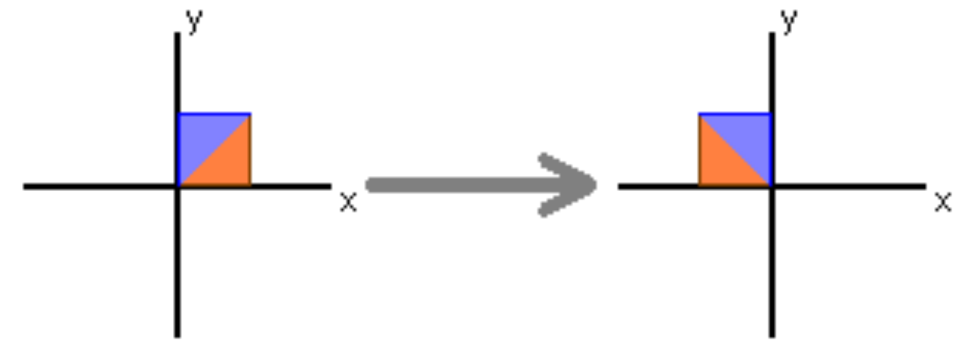
2D mirror about Y axis?



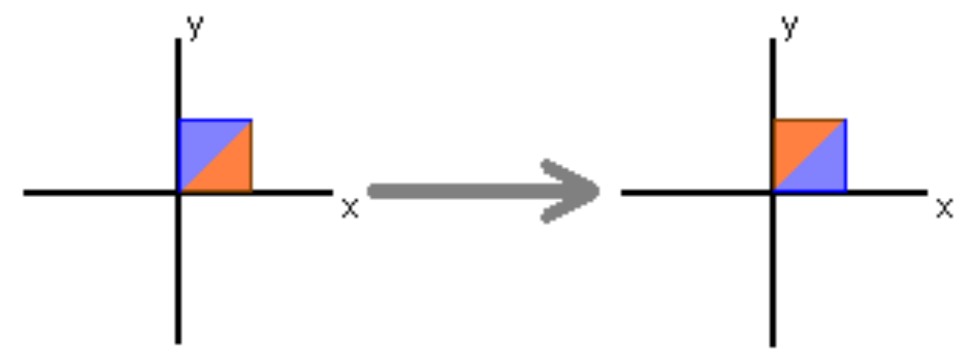
# 2x2 Matrices

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

2D mirror about Y axis?




2D mirror across line y = x?

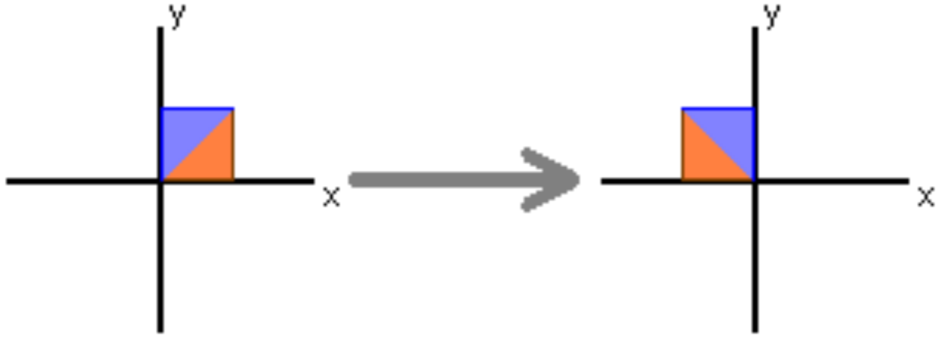


# 2x2 Matrices

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

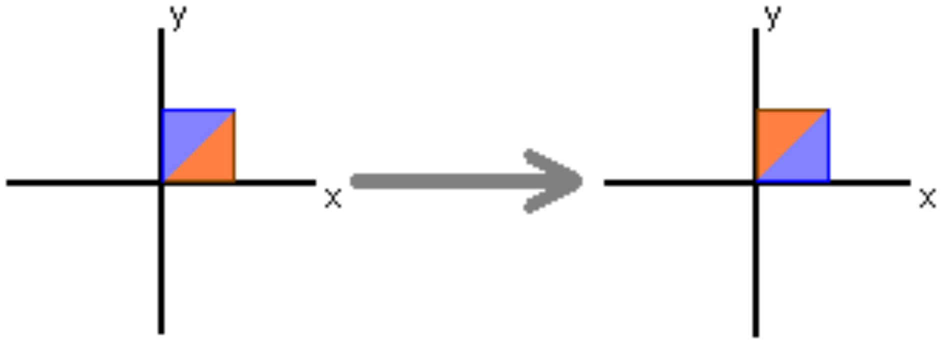
2D mirror about Y axis?

$$\begin{aligned} x' &= -x \\ y' &= y \end{aligned} \qquad T = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$



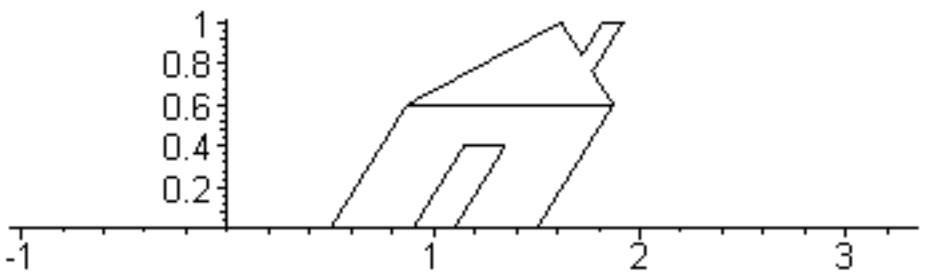

2D mirror across line y = x?

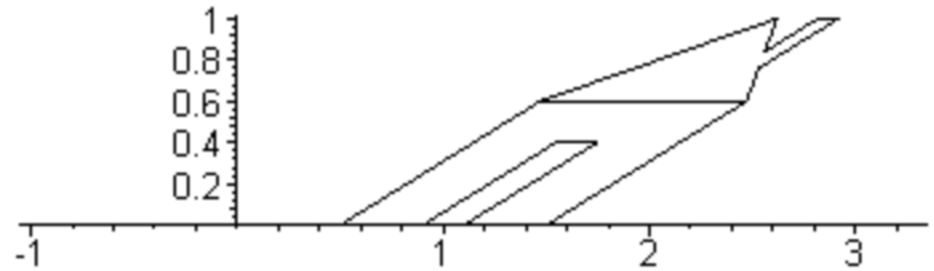
$$\begin{aligned} x' &= y \\ y' &= x \end{aligned} \qquad \mathbf{T} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$



# 2x2 Matrices

- 2D Shearing




shear along x axis = $\begin{bmatrix} 1 & shearx \\ 0 & 1 \end{bmatrix}$

shear along y axis = $\begin{bmatrix} 1 & 0 \\ sheary & 1 \end{bmatrix}$

http://www.willamette.edu/~gorr/classes/GeneralGraphics/Transforms/images/shearAnimate.gif

# 2x2 Matrices

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

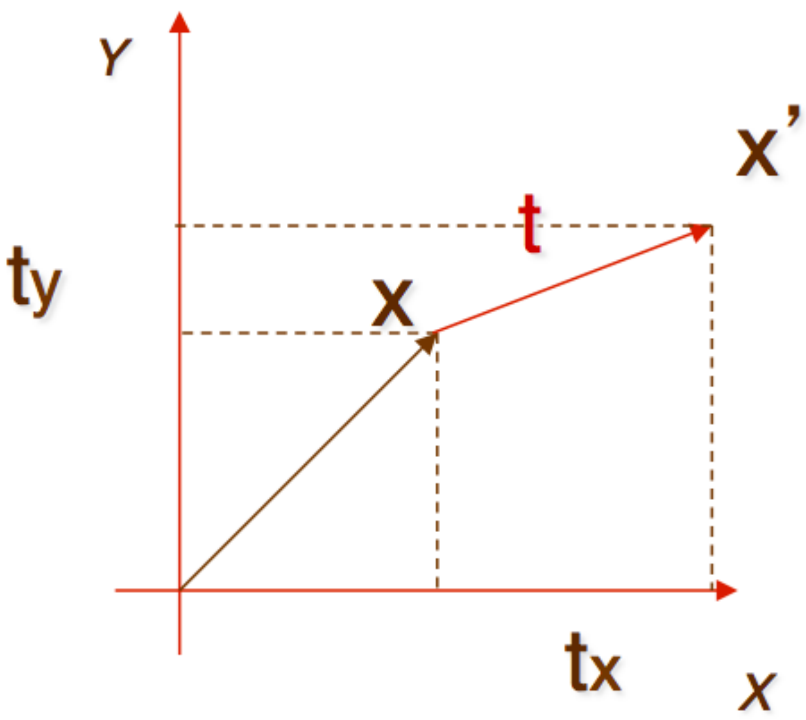
2D Translation?

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

NO!

Translation is not a linear operation on 2D coordinates

# Translation




$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\mathbf{t} = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

$$\mathbf{x}' = \mathbf{x} + t = \begin{bmatrix} \mathbf{x} + t_x \\ \mathbf{y} + t_y \end{bmatrix}$$

Translation is not a linear operation on 2D coordinates

# Rotation + Translation?

- To scale or rotate about a particular point (the fixed point) we must first translate the object so that the fixed point is at the origin. We then perform the scaling or rotation and then the inverse of the original translation to move the fixed point back to its original position.

- **Quiz:** What is the transformation matrix q if we want to scale an image by 2 in each direction about the point fp = (1.5,1)?

# All 2D Linear Transformations

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

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

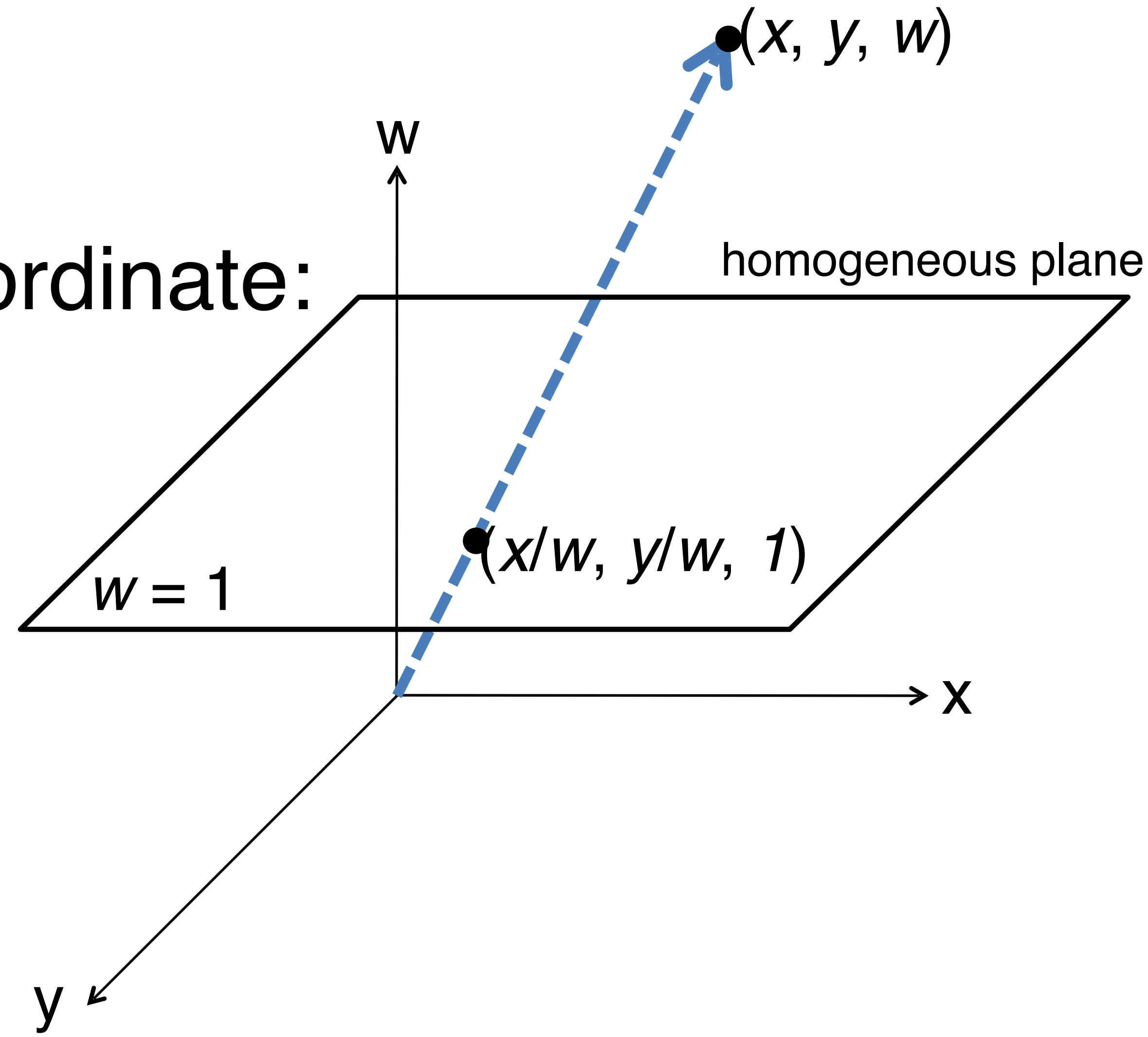
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

# Homogeneous coordinates

Trick: add one more "virtual" coordinate:

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

homogeneous image coordinates




Converting *from* homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

# Translation

- Solution: homogeneous coordinates to the rescue

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

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

# Affine transformations

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

any transformation with last row [ 0 0 1 ] we call an

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

# Basic affine transformations

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

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

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

2D *in-plane* rotation

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

Shear

# Affine Transformations

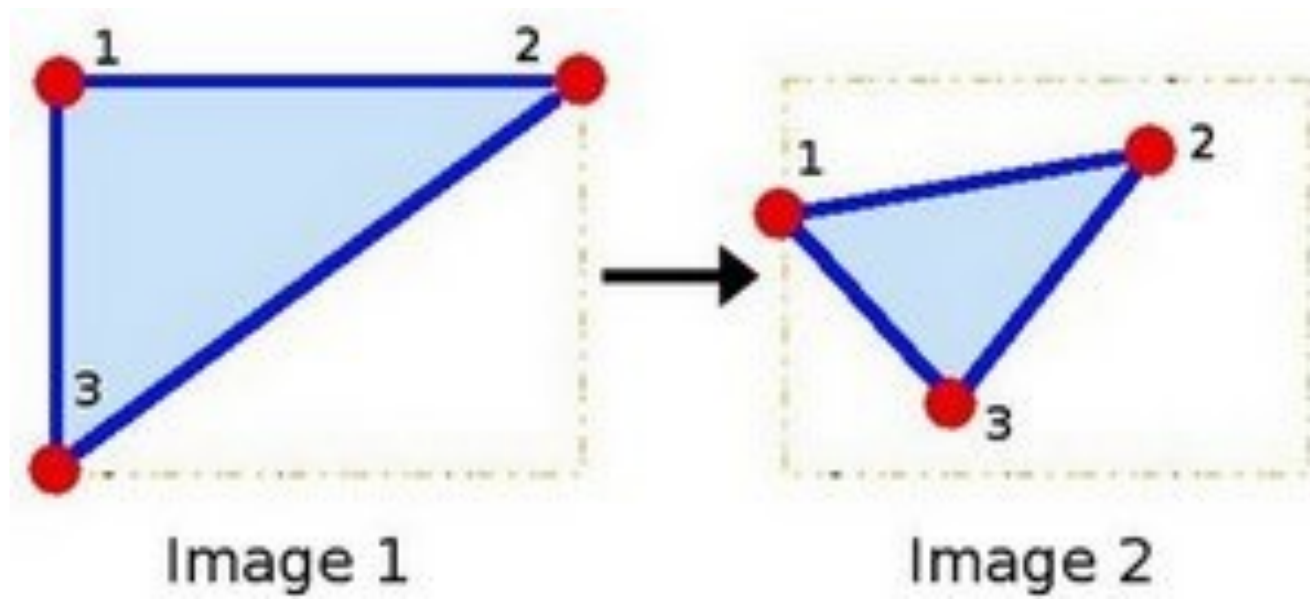
- Affine transformations are combinations of …
  – Linear transformations, and
  – Translations

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

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

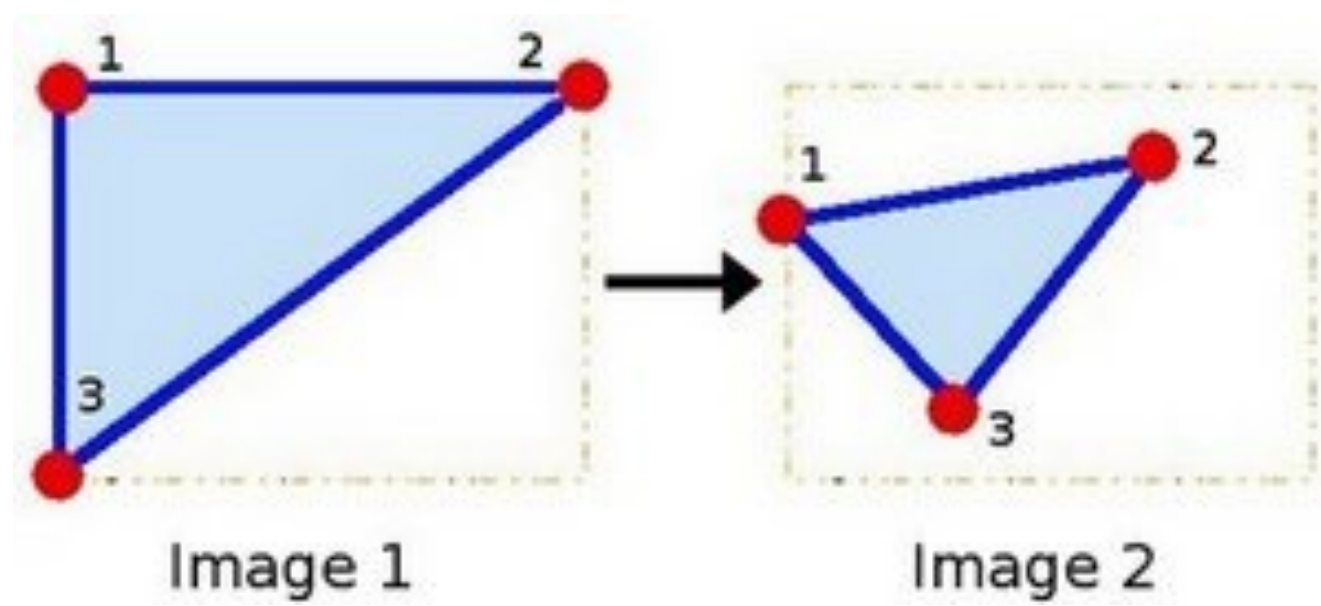
  – Python: cv2.warpAffine.

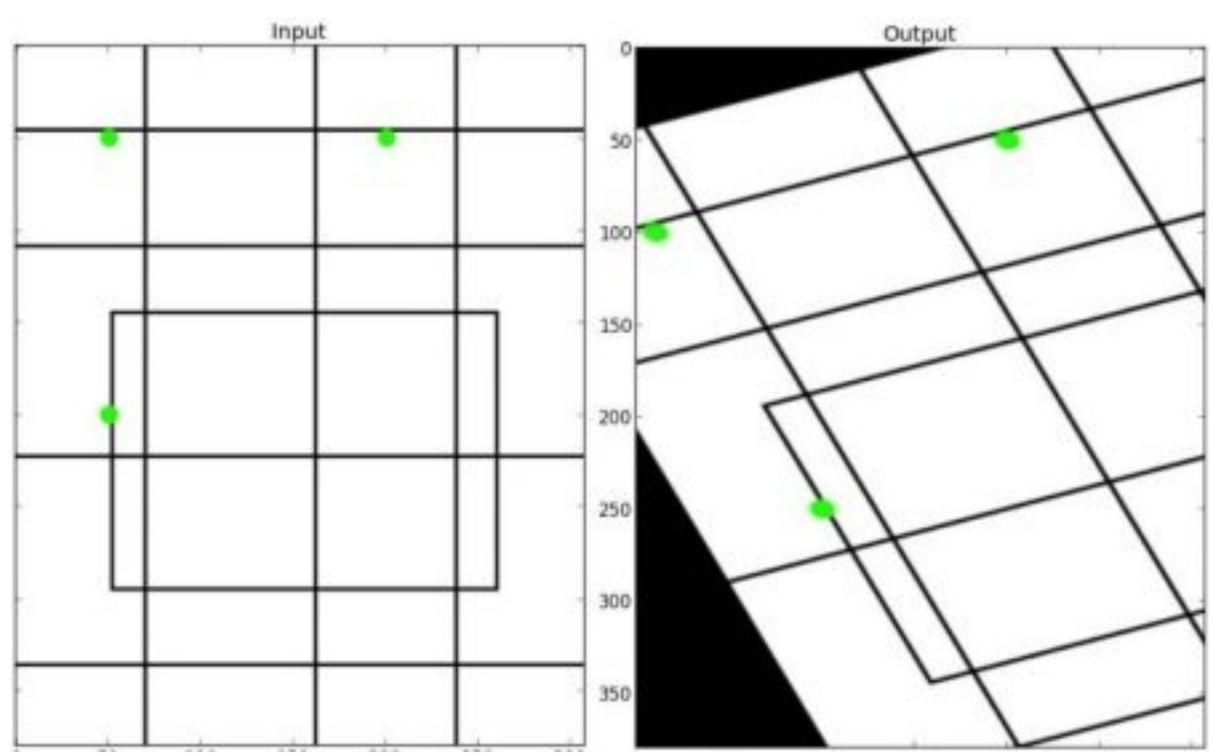
# Affine Transformation Matrix




There are 6 unknowns in the matrix!
Once we specify 3 points, we can obtain transformation matrix M

# Affine Transformation Matrx



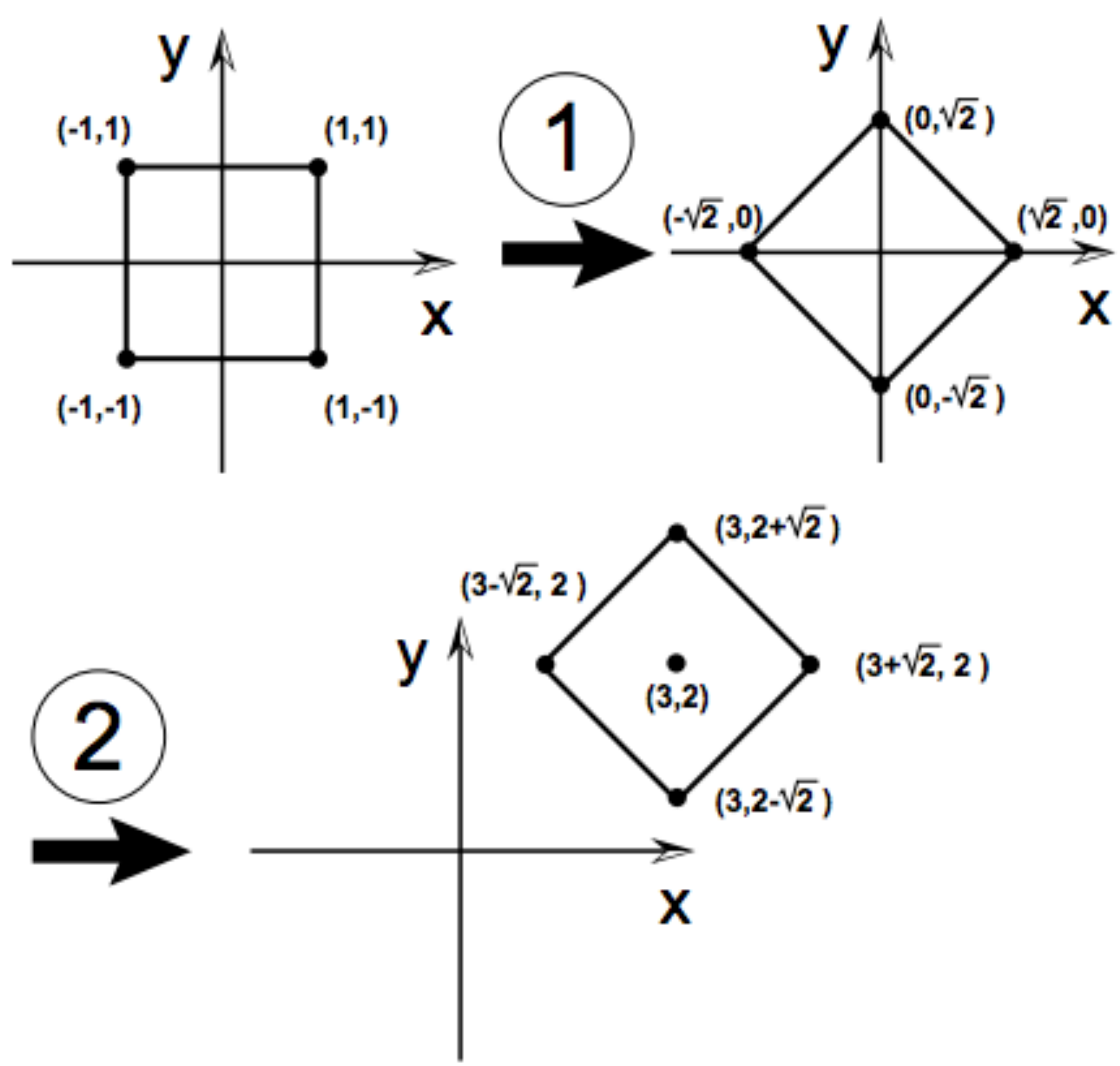
# Quiz (take out a piece of paper and pen)

1. Affine transformation is linear (True or False)
2. Similarity invariant of length ratios and angles between line are not preserved under affine transformation. (True or False)
3. Are the two translations the same (Yes or No)

   A. rotation->translation
   B. translation->rotation

3 Write down an affine transformation matrix M of the following operations: suppose we have a 2×2 square centered at the origin and we want to first rotate the square by 45 about its center and then move the square so its center is at (3, 2).

# Ordering of operations matter!



# Now we can rewrite

- What is the transformation matrix q if we want to scale an image by 2 in each direction about the point fp = (1.5,1)?

$$q = (-T)\ S\ T\ p = A\ p,$$

$$\text{where } A = (-T)\ S\ T$$

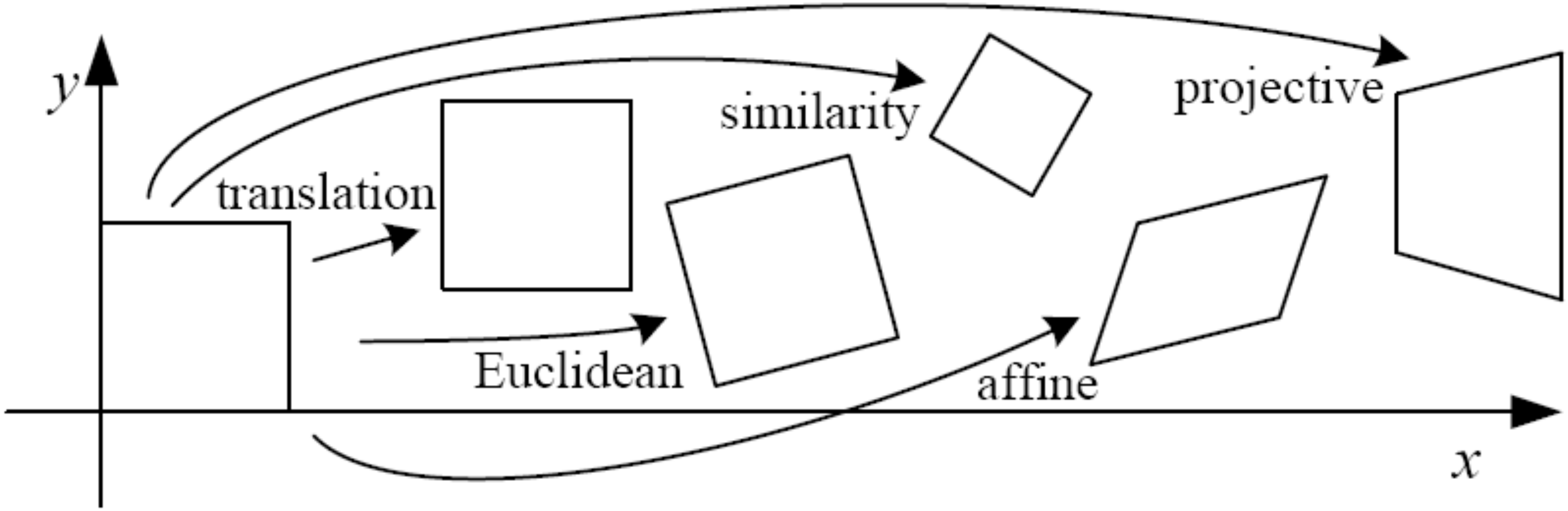
Everything now is in the form of matrix multiplication

# Affine Transformations

- Affine transformations are combinations of …
  - Linear transformations, and
  - Translations

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

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




- Euclidean: translation, rotation, reflection
- Similarity: translation, rotation, uniform scale, reflection
- Affine: linear transformations + translation

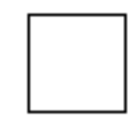
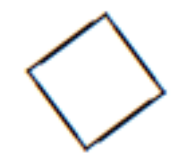
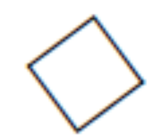
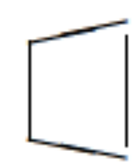
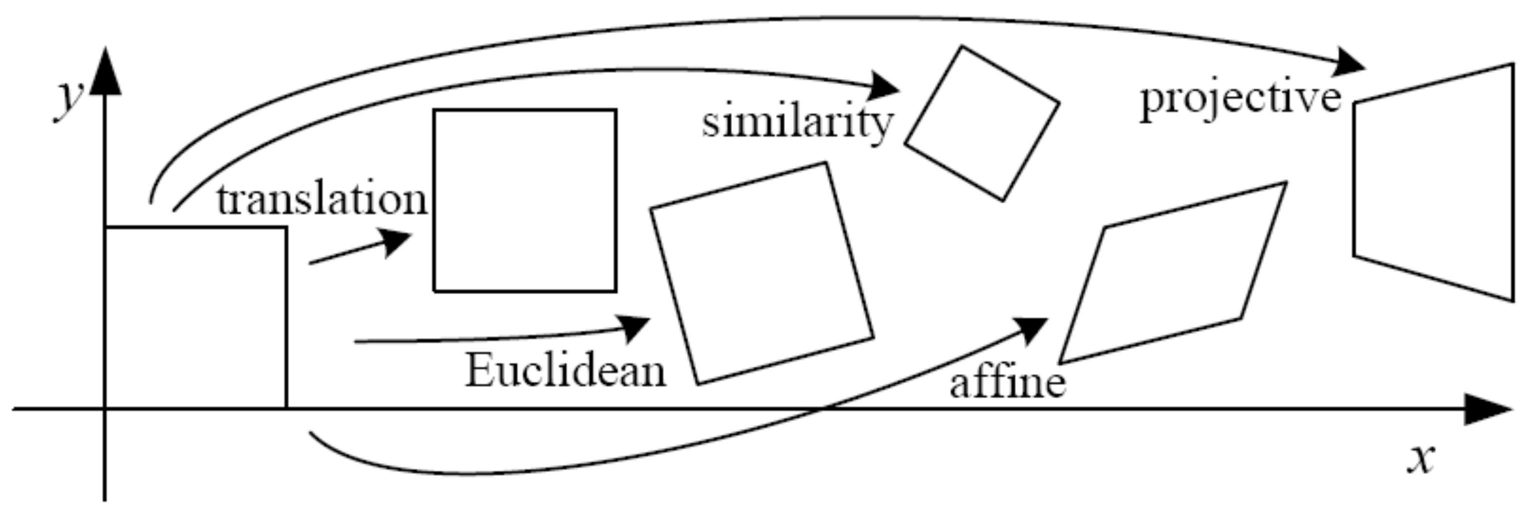
| Transformation | Matrix | # DoF | Preserves | Icon |
|---|---|---|---|---|
| translation | $\begin{bmatrix} \boldsymbol{I} & \vert & \boldsymbol{t} \end{bmatrix}_{2\times 3}$ | 2 | orientation |  |
| rigid (Euclidean) | $\begin{bmatrix} \boldsymbol{R} & \vert & \boldsymbol{t} \end{bmatrix}_{2\times 3}$ | 3 | lengths |  |
| similarity | $\begin{bmatrix} s\boldsymbol{R} & \vert & \boldsymbol{t} \end{bmatrix}_{2\times 3}$ | 4 | angles |  |
| affine | $\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$ | 6 | parallelism |  |
| projective | $\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$ | 8 | straight lines |  |

**Table 2.1** Hierarchy of 2D coordinate transformations. Each transformation also preserves the properties listed in the rows below it, i.e., similarity preserves not only angles but also parallelism and straight lines. The $2 \times 3$ matrices are extended with a third $[\mathbf{0}^T\ 1]$ row to form a full $3 \times 3$ matrix for homogeneous coordinate transformations.

# Homographies



# Reading

- Szeliski: Chapter 2.1 & 6.1

- Geometric transformation basics:
- http://www.willamette.edu/~gorr/classes/GeneralGraphics/Transforms/transforms2d.htm

# Exercises in Python

- Download Demo.py
- Finish the exercises in the code
- with image Messi15.png

# Is this an affine transformation?

# Where do we go from here?

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

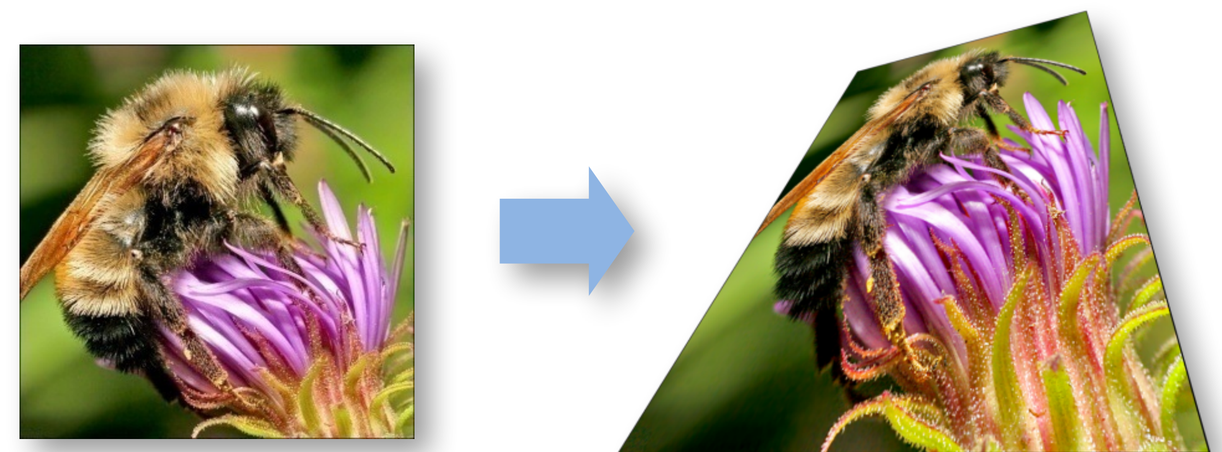
what happens when we mess with this row?

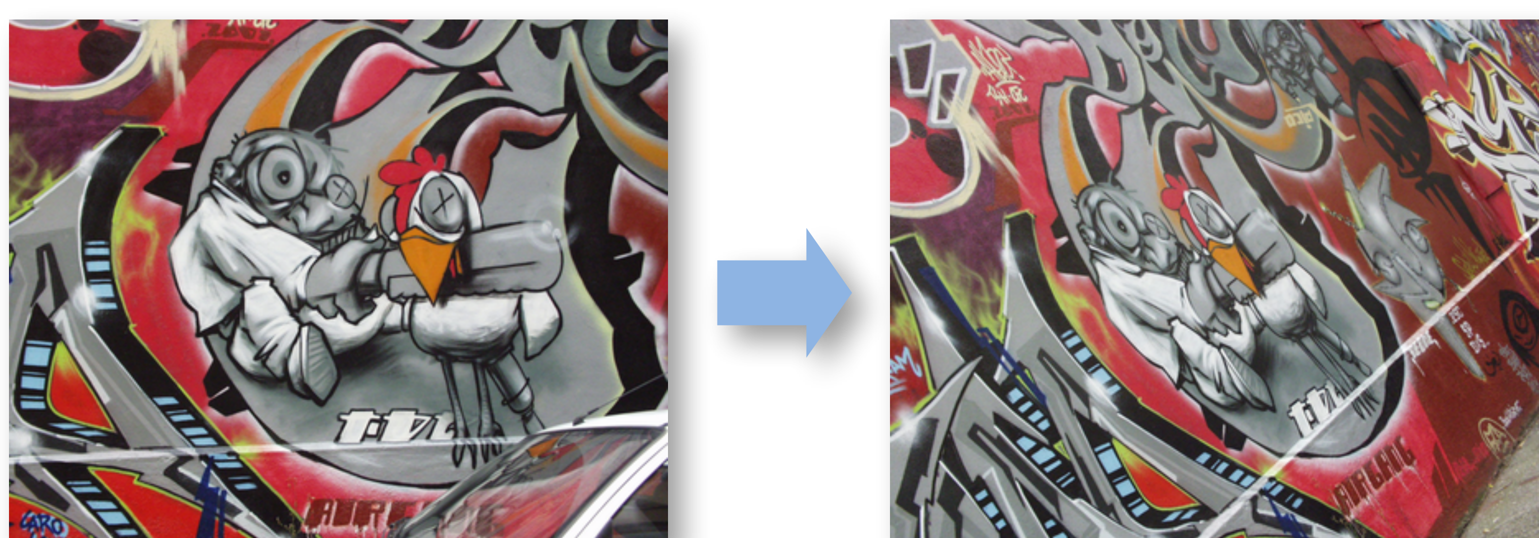
affine transformation

# Projective Transformations aka Homographies aka Planar Perspective Maps

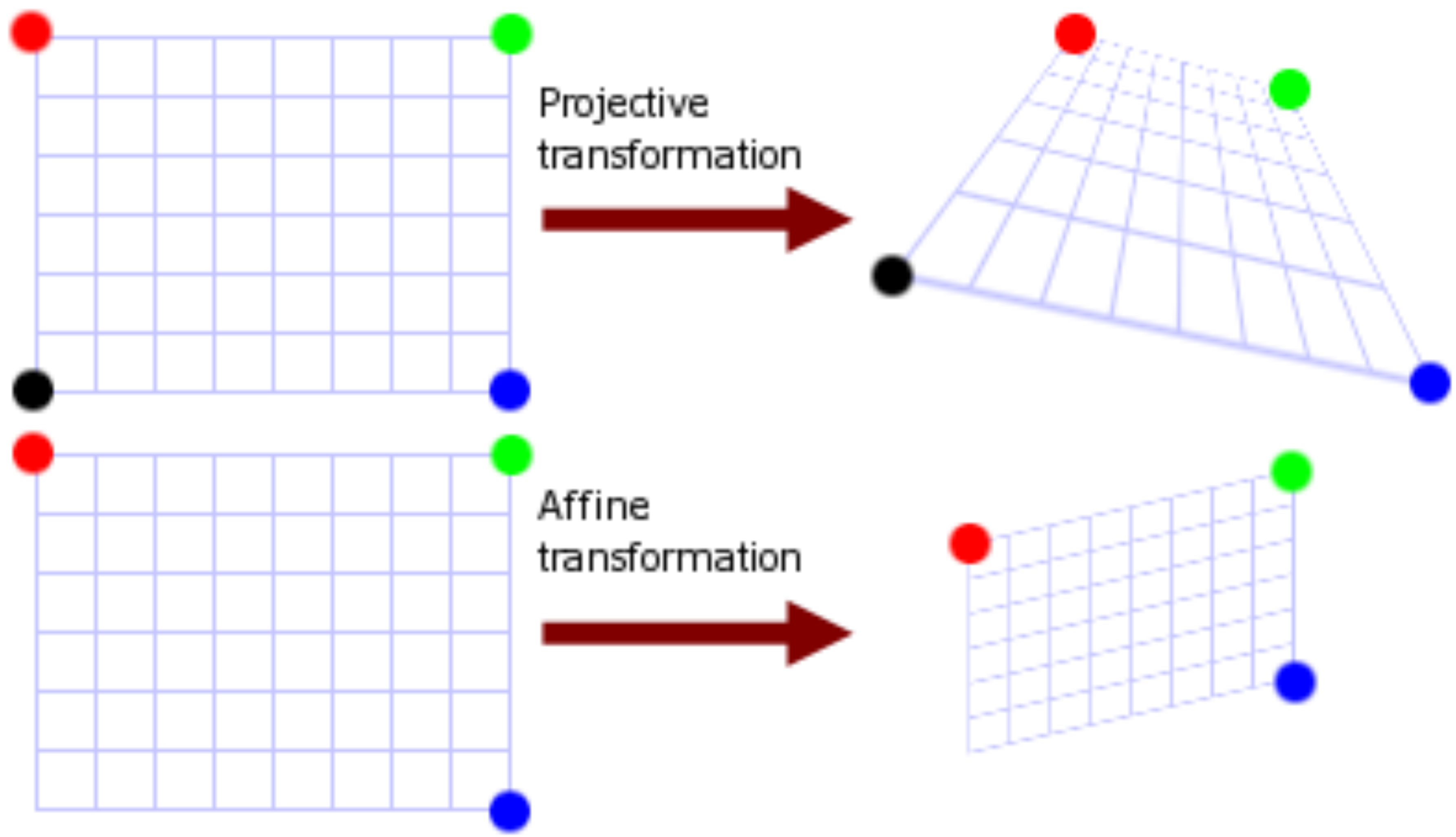
$$\mathbf{H} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix}$$

Called a *homography* (or *planar perspective map*)



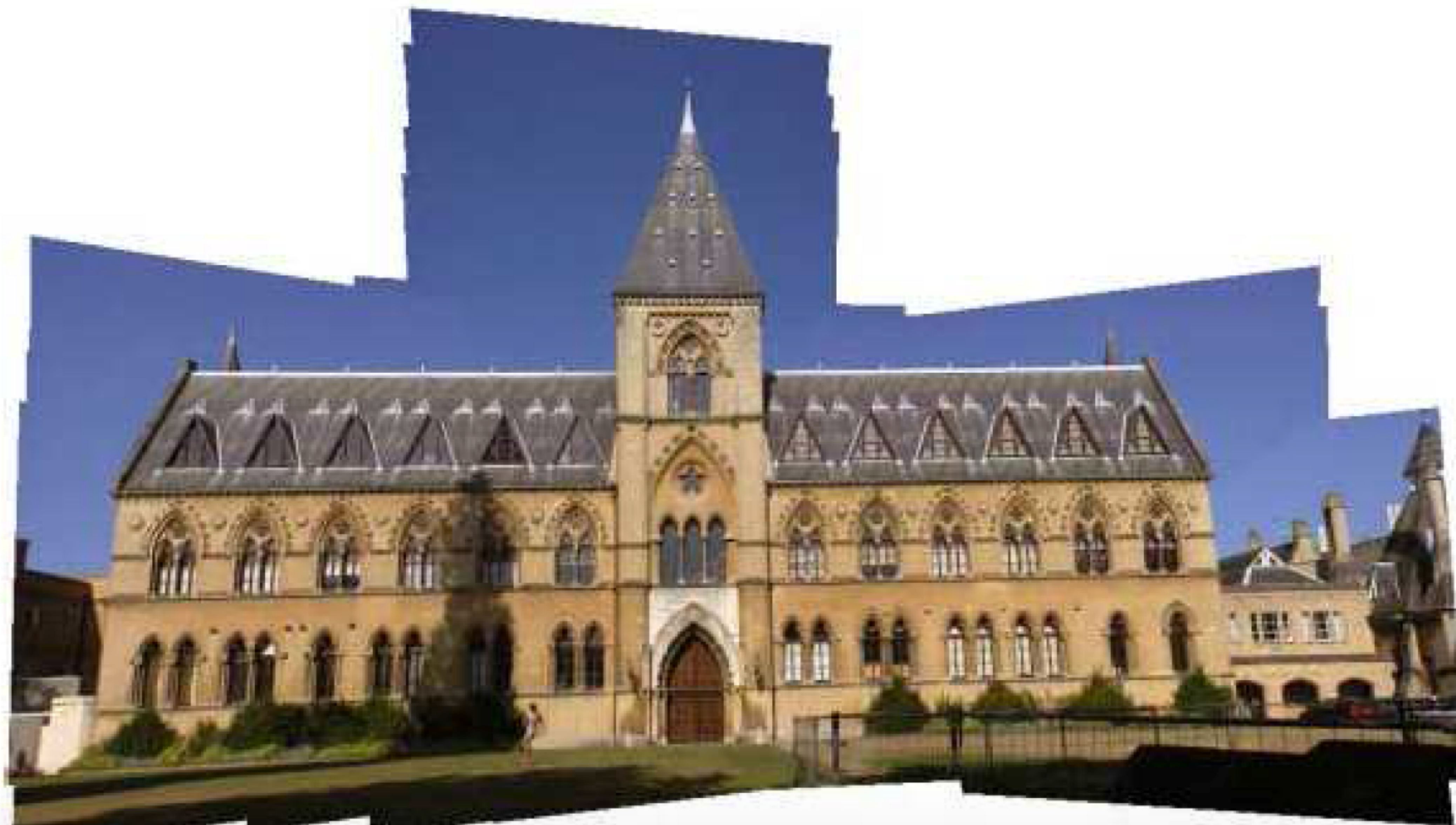


# Affine vs. Projective Transformations



# Why do we care?

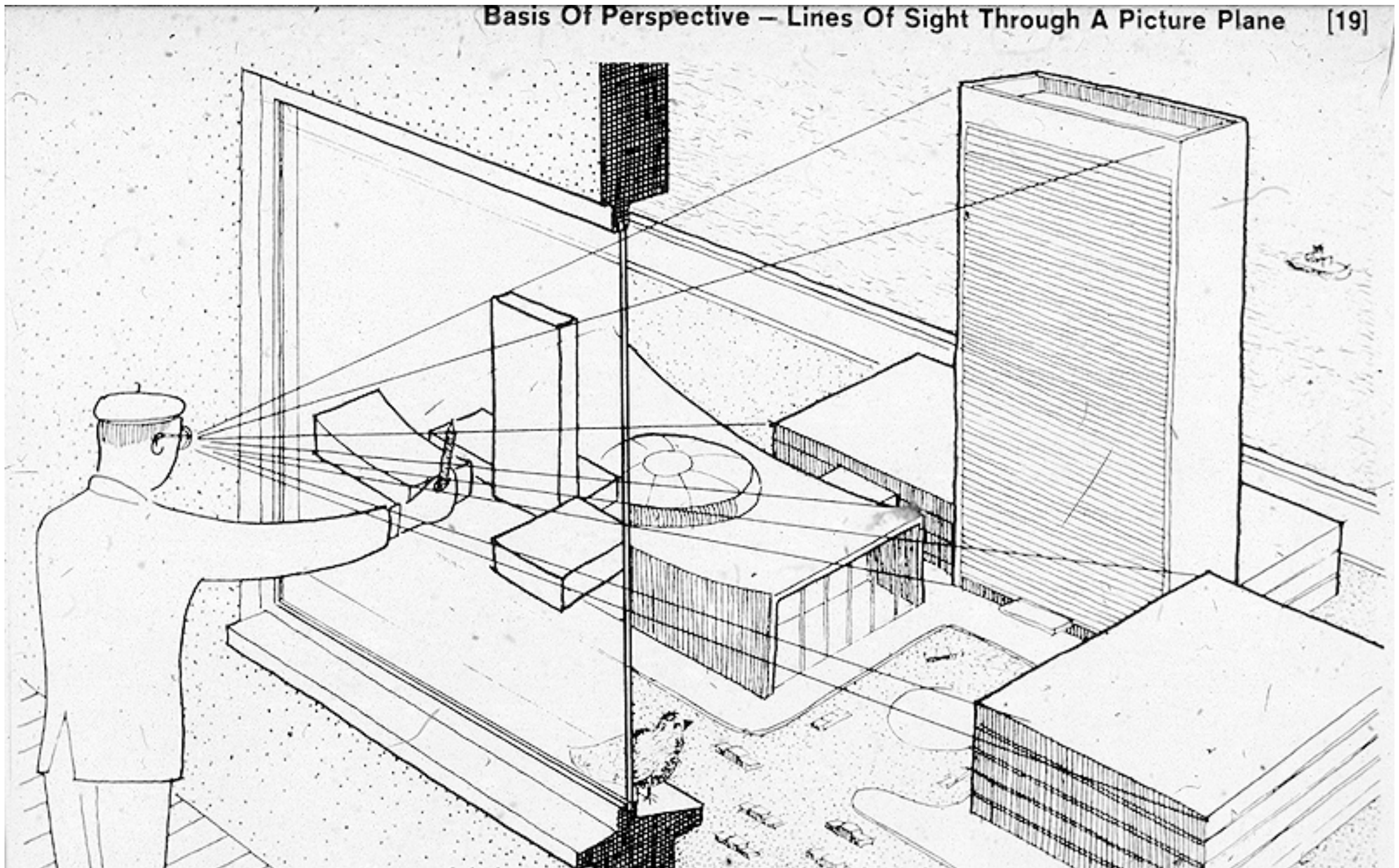
- What is the relation between a plane in the world and a perspective image of it?
- Can we reconstruct another view from one image?
- Relation between pairs of images
  - Need to make a mosaic



# Homographies

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} ax + by + c \\ dx + ey + f \\ gx + hy + 1 \end{bmatrix}$$

# Plane projection in drawing



The concept of the picture plane may be better understood by looking through a window or other transparent plane from a fixed viewpoint. Your lines of sight, the multitude of straight lines leading from your eye to the subject, will all intersect this plane. Therefore, if you were to reach out with a grease pencil and draw the image of the subject on this plane you would be "tracing out" the infinite number of points of intersection of sight rays and plane. The result would be that you would have "transferred" a real three-dimensional object to a two-dimensional plane.

[CS 417 Spring 2002]