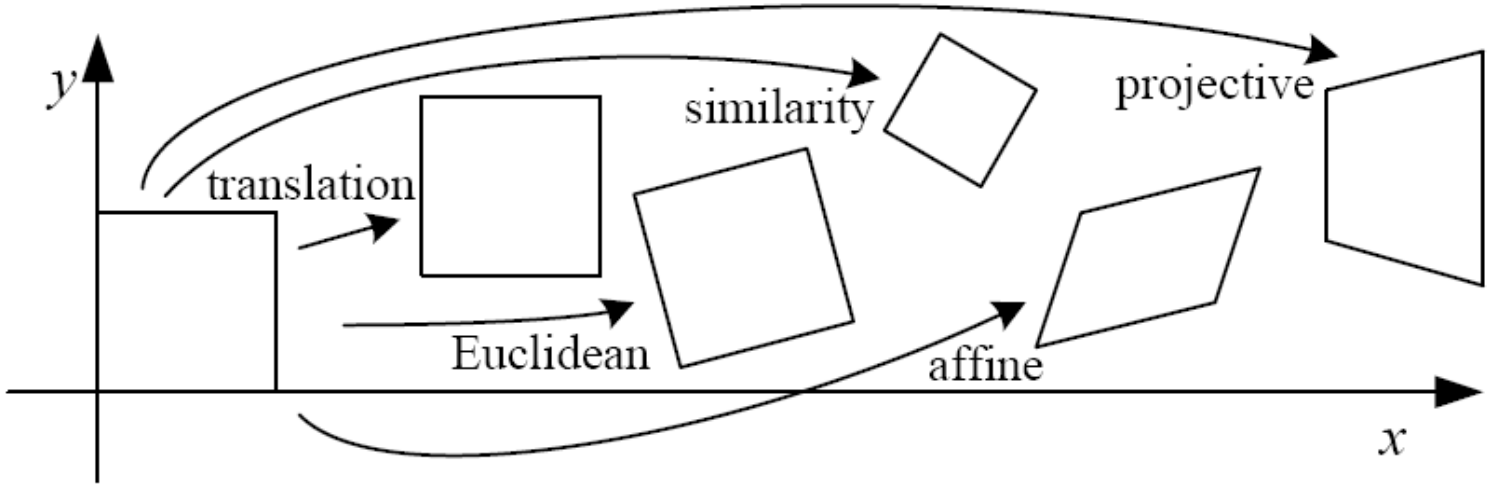


CSCI 497P/597P: Computer Vision



Lecture 12: Transformations
2D Linear and Affine Transformations

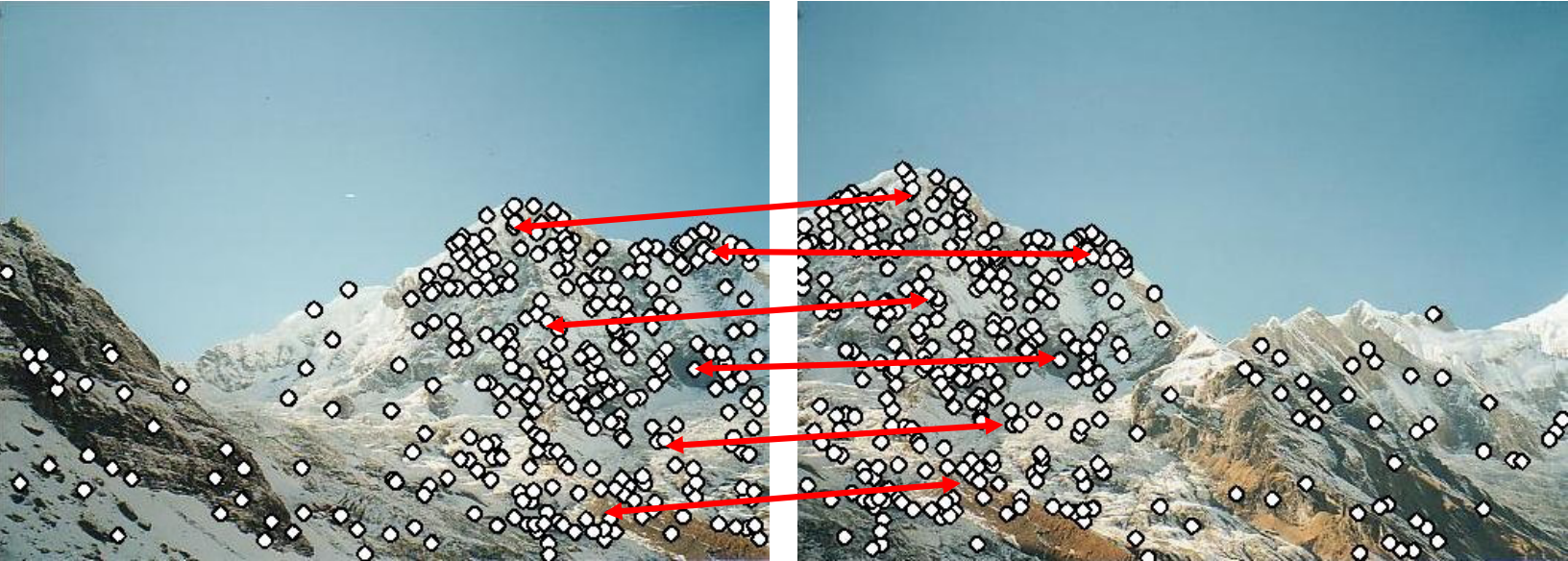
Announcements

- To use slip days, send me email **after** you've submitted late.

Goals

- Know what is possible with 2D **linear transformations**:
(scale, shear, rotation)
- Understand the motivation and math behind **homogeneous coordinates**.
- Know what is possible with 2D **affine transformations**:
(all of the above, plus translation)

Running motivational example: Panorama Stitching



Where are we?

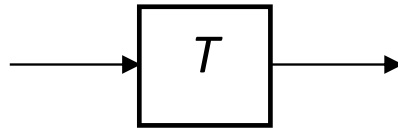


linear, affine, or projective
transformations

1. How do we describe the transformation?
2. How do we find an accurate transformation?
 - least squares
 - ➔ RANSAC
3. How do we actually warp the image?
 - inverse warping

Parametric (global) Warping

- Apply the same function to all coordinates.



$\mathbf{p} = (x, y)$ T transforms *image coordinates* $\mathbf{p}' = (x', y')$

$$x', y' = T(x, y)$$

Self-imposed restriction: T is a matrix.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \overset{T}{\begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix}} \begin{bmatrix} x \\ y \end{bmatrix}$$

What can we do with this?

2x2 Matrices

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

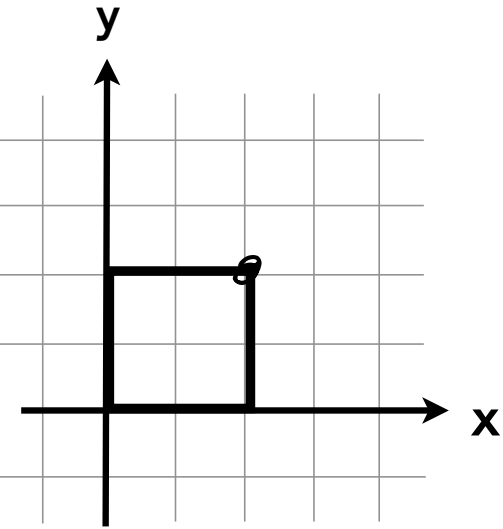
A 2x2 matrix can represent all possible **linear transformations** on input coordinates.



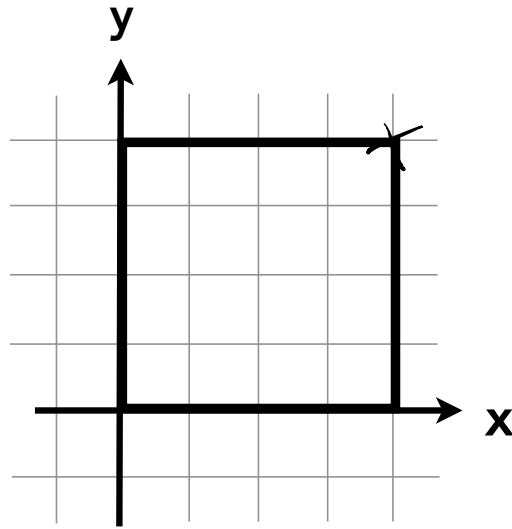
(each output coordinate = linear function of input coordinates)

$$f(x, y) = \underline{ax + by} \leftarrow \text{linear fn}$$

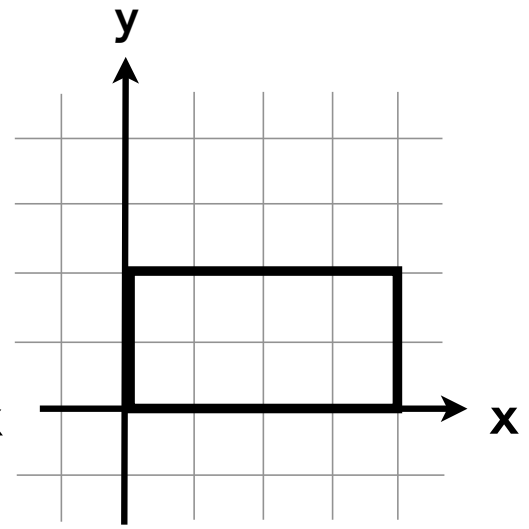
Scale



Input



Uniform Scale



Nonuniform Scale

$$x' = 2x$$

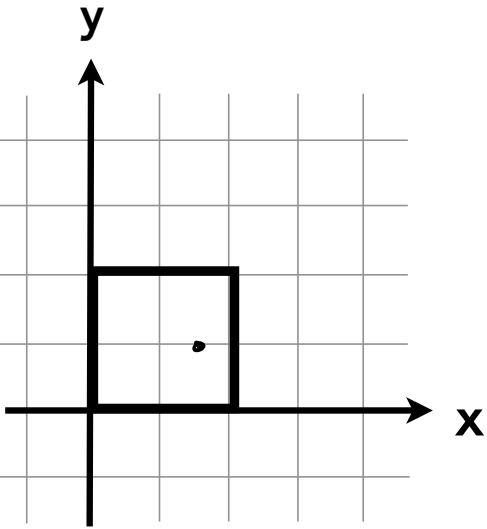
$$y' = 2y$$

$$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

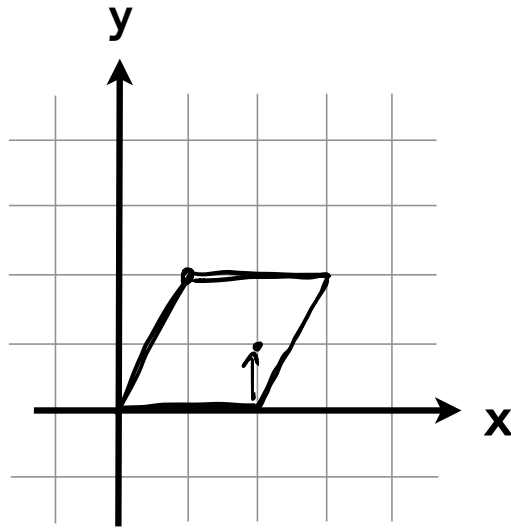
$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$$

↑
leave y unchanged

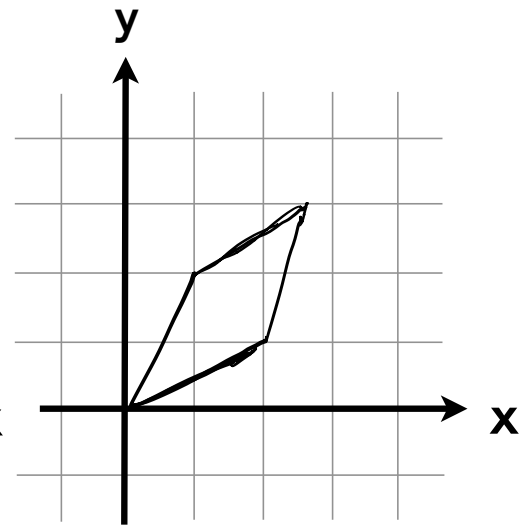
Shear



Input



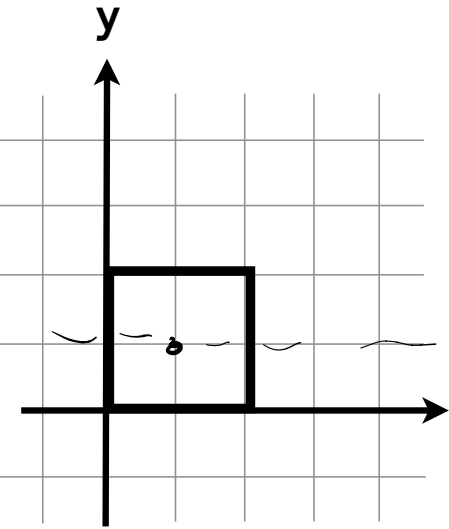
Shear (x)



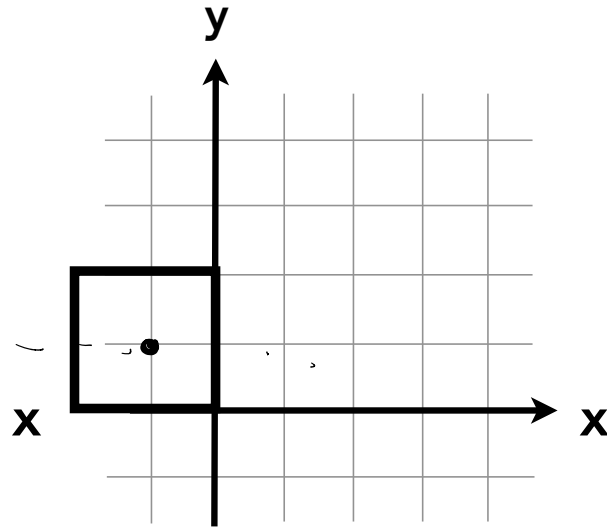
Shear (both)

$$\begin{aligned} x' &= x + 0.5y \\ y' &= y \end{aligned} \quad \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad \begin{bmatrix} 1 & s_x \\ s_y & 1 \end{bmatrix}$$

Reflection

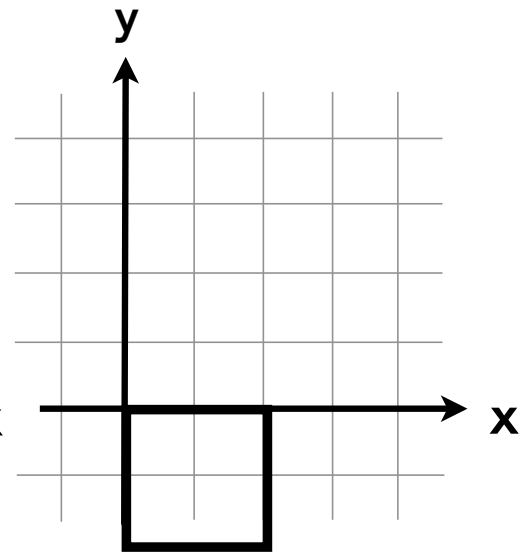


Input



Reflection
(across y)

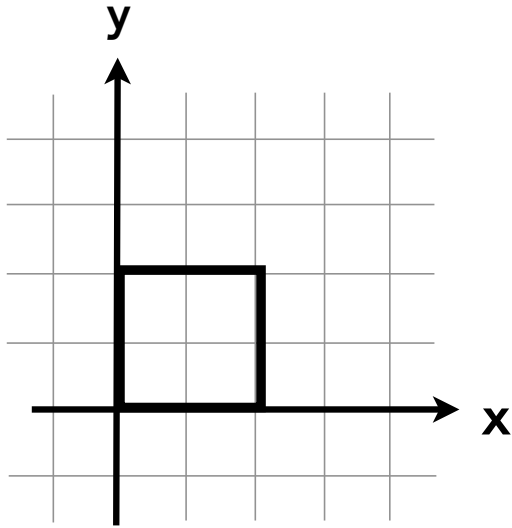
$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



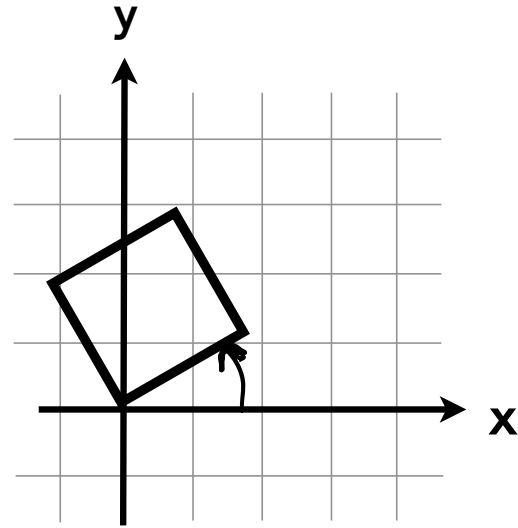
Reflection
(across x)

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

Rotation



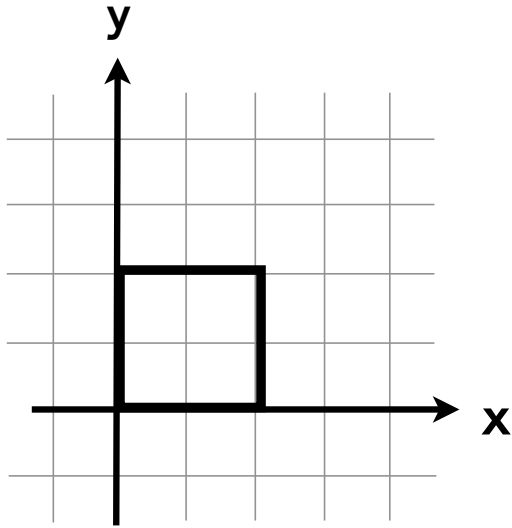
Input



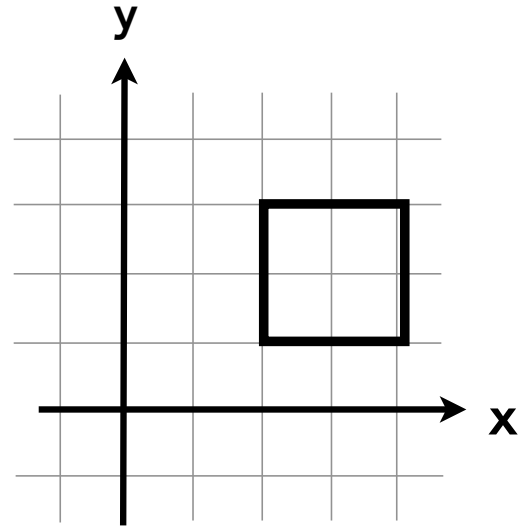
Rotation about the origin

$$\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

Translation



Input



Translation by (u, v)

Today's Problems

1. What's the fewest correspondences you could use to (unambiguously) find a 2x2 linear transformation T ?
2. Come up with a matrix that represents a *translation*.

Diagram illustrating a linear transformation T between two coordinate systems:

Left system: $\begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$

Right system: $\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$

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

The matrix is circled in orange. Arrows indicate the flow of information: x_1 and y_1 are labeled "known", the matrix elements a, b, c, d are labeled "unknown?", and x_2 and y_2 are labeled "known".

Matrices can't translate.

We'll use a clever math hack to make them do it anyway:

Homogeneous Coordinates

Use a **3D** vector to represent a **2D** point.

Always put a **1** in the third dimension.

Matrices can't translate.

We'll use a clever math hack to make them do it anyway:

Homogeneous Coordinates

Use a **3D** vector to represent a **2D** point.

Always put a **1** in the third dimension.

A hand-drawn diagram illustrating the concept of homogeneous coordinates. On the left, a vertical vector is drawn with a bracket on the right, containing the labels x , y , and $-$ (representing a missing third component). To its right is a large, empty square bracket. Further right, a 3×3 transformation matrix is represented by a square bracket with a T inside. Above this matrix, the text "3x3 ???" is written. To the right of the matrix is a vertical vector with a bracket on the right, containing the labels x , y , and 1 . A thick arrow points from the right side of this vector towards the right edge of the slide.

Matrices can't translate.

We'll use a clever math hack to make them do it anyway:

Homogeneous Coordinates

Use a **3D** vector to represent a **2D** point.

Always put a **1** in the third dimension.

How do we transform these?

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} t_{11}x + t_{12}y + t_{13} \\ t_{21}x + t_{22}y + t_{23} \\ 1 \end{bmatrix}$$

linear *translation*

Interactive Demo

- <https://iis.uibk.ac.at/public/piater/courses/demos/homography/homography.xhtml>

Affine Transformations

The transformations possible with a 3x3 matrix like this

$$\begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

are called **affine transformations**.



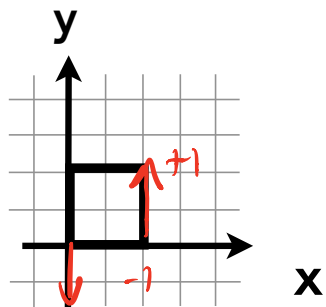
each input coordinate is an affine function of the input coordinates

Affine basically means linear plus shift:

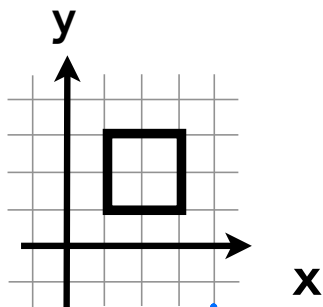
$f(x, y) = \underline{ax + by}$ is linear

$f(x, y) = \underline{ax + by + c}$ is affine

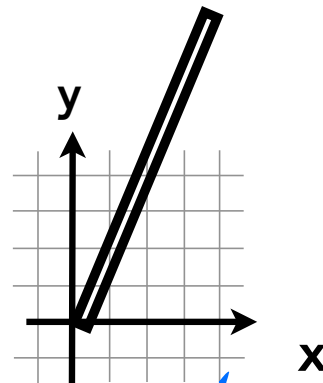
Which of these **can** be done by a **2D linear** transformation?



Input



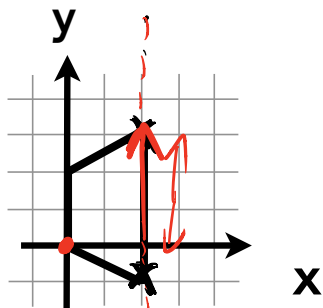
A *N*



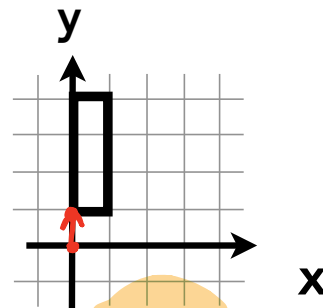
B *Y*

$$y' = ax + by$$

W *W*

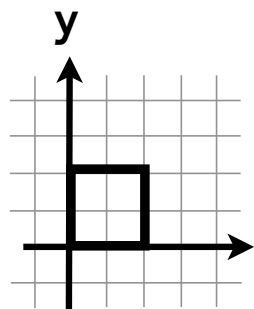


C *N*

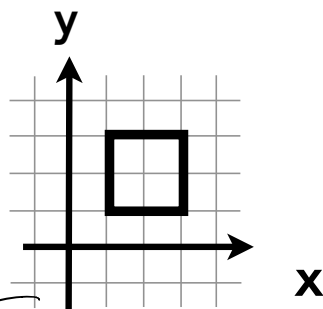


D *N*

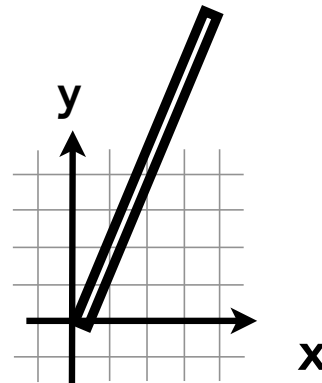
Which of these **can't** be done by a **2D affine** transformation?



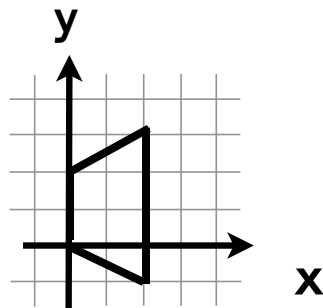
Input



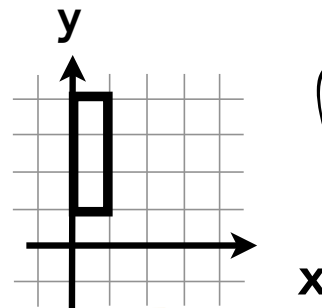
A



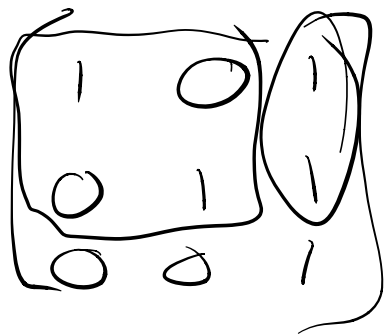
B



C



D



$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Transformations: Properties

- Anything you get from matrix multiplication comes for free!
 - Associative! Composable!
 - **Not** commutative.

Linear Transformations: Properties

- Linear transformations
$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} a & b & 0 \\ d & e & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

- Properties:

- Lines map to lines
- Parallel lines remain parallel
- Ratios of lengths along lines are preserved
- Closed under composition

linear

- **Origin maps to origin**

Affine Transformations: Properties

- Affine transformations

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

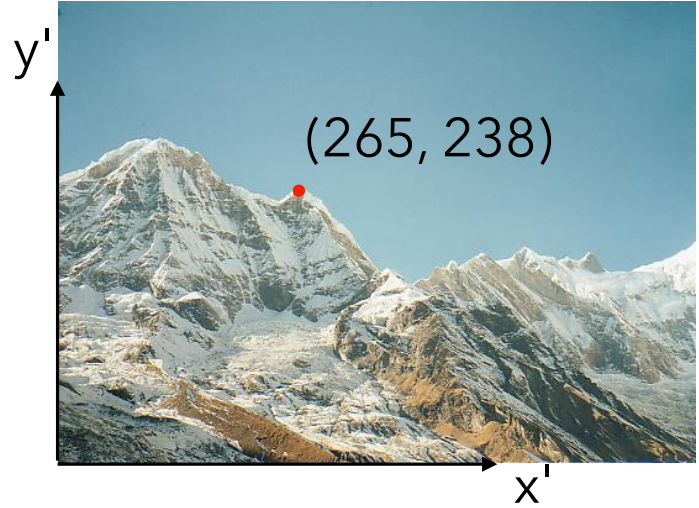
- Properties:

- Lines map to lines
- Parallel lines remain parallel
- Ratios of lengths along lines are preserved
- Closed under composition
- Origin **does not** necessarily map to origin

linear

affine

Warping



We've found correspondence. What's the transformation?

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

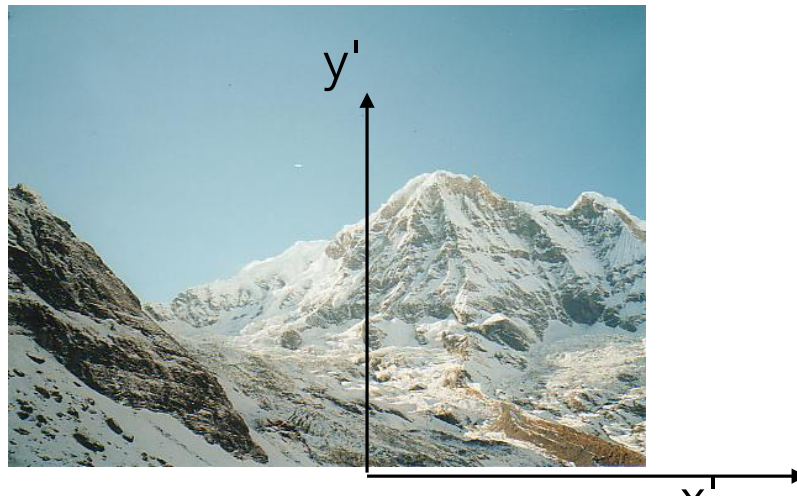
Warping



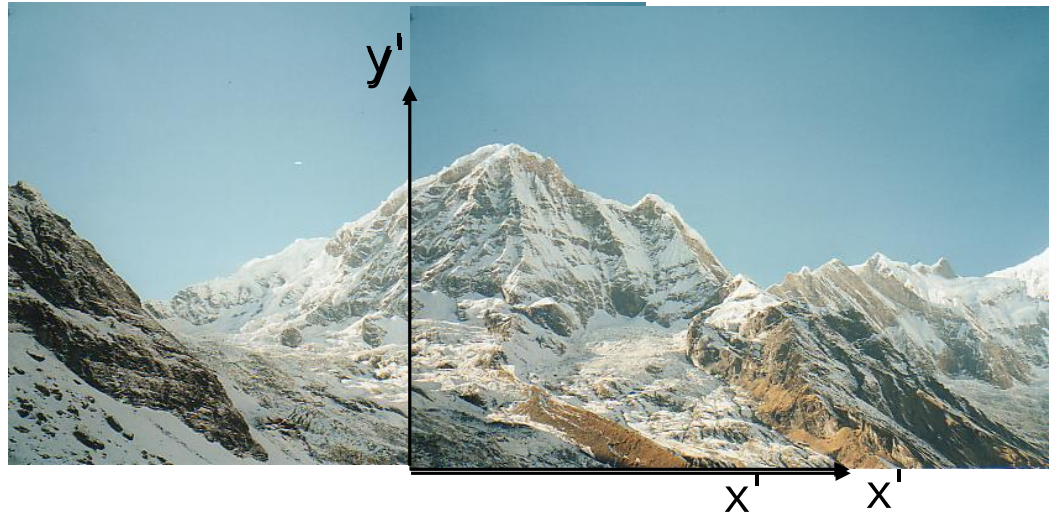
We've found the transformation. How do we warp the image?

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -115 \\ 0 & 1 & -3 \\ 0 & 9 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

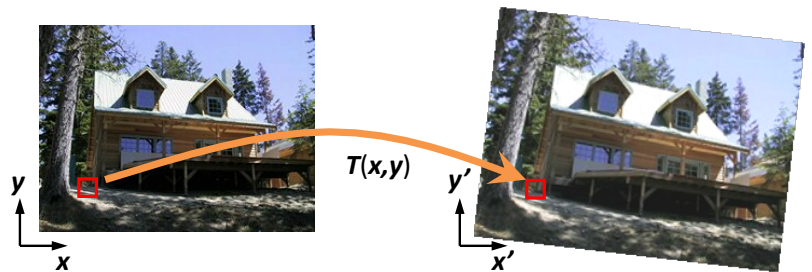
Warping



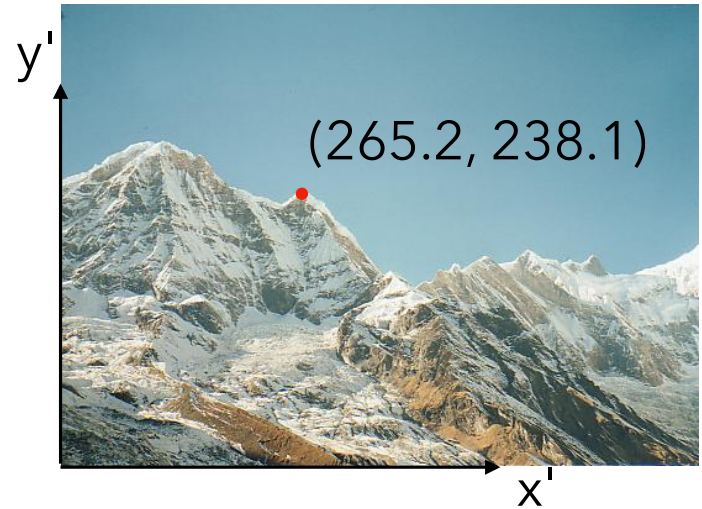
Warping



Forward Warping

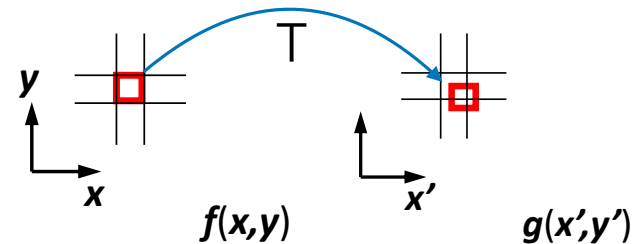


Warping

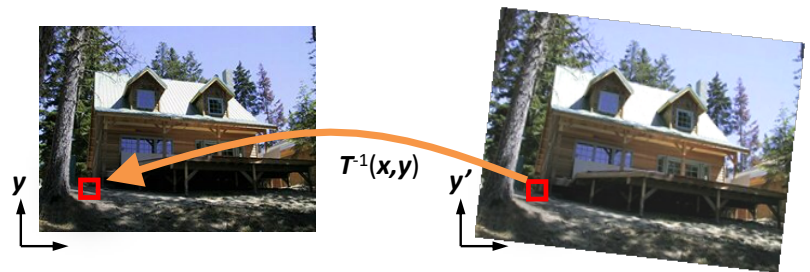


We've found the transformation. How do we warp the image?

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -114.8 \\ 0 & 1 & -2.9 \\ 0 & 9 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



Inverse Warping



Bilinear Interpolation