Planar Geometry

CSE P576

Dr. Matthew Brown

Image Alignment

- Aim: warp our images together using a 2D transformation

Image Alignment

- Aim: warp our images together using a 2D transformation

Image Alignment

- Find corresponding (matching) points between the images

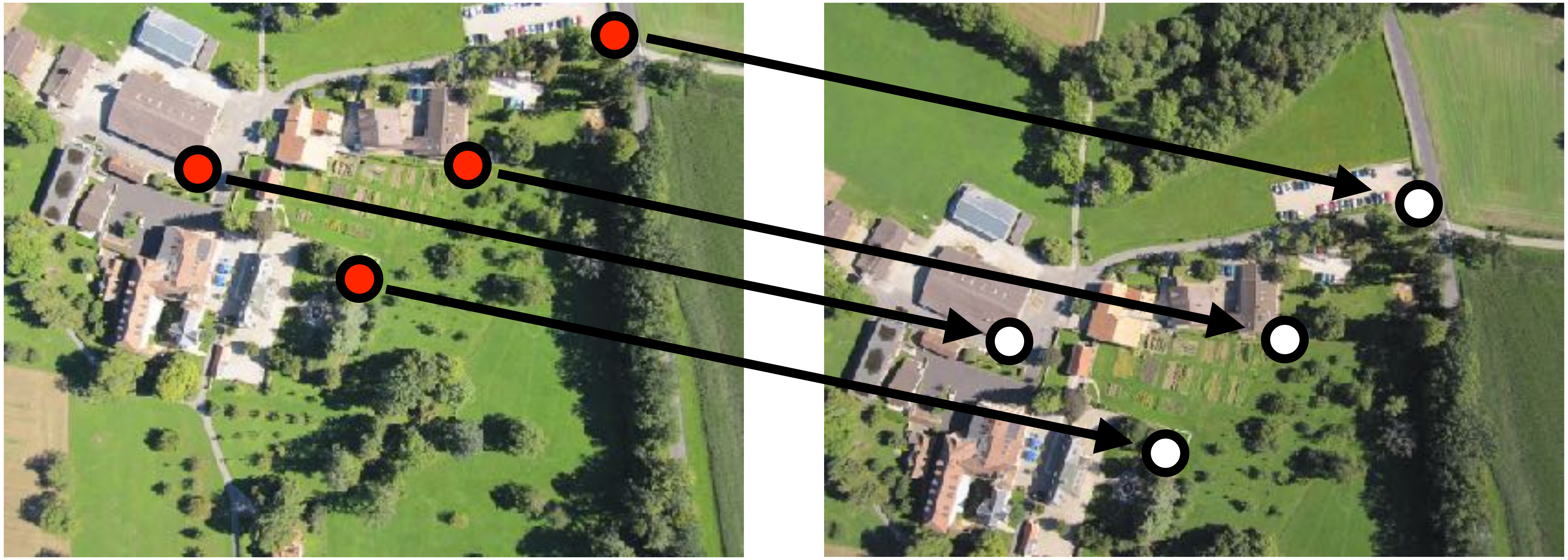


Image Alignment

- Compute the transformation to align the points

Image Alignment

- We can also use this transformation to reject outliers

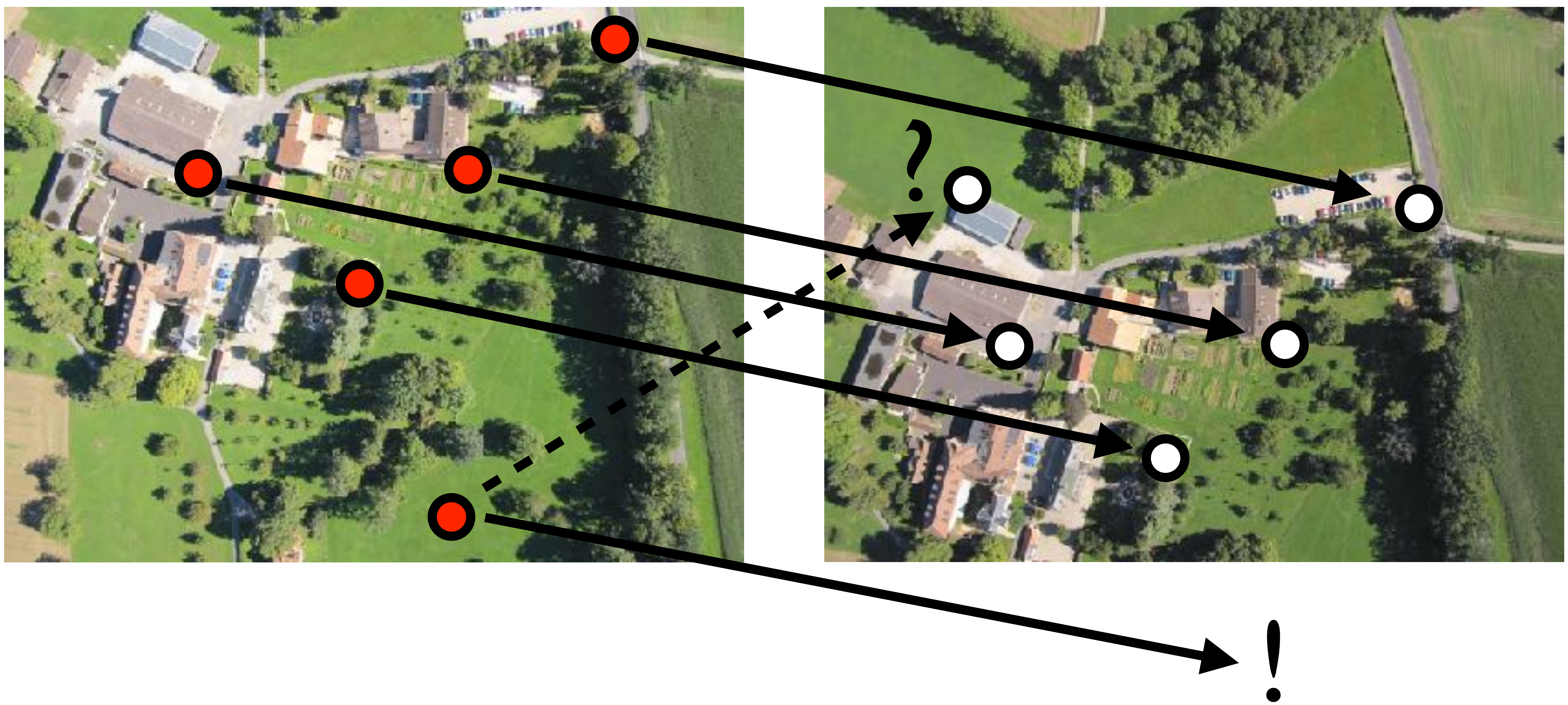


Image Alignment

- We can also use this transformation to reject outliers

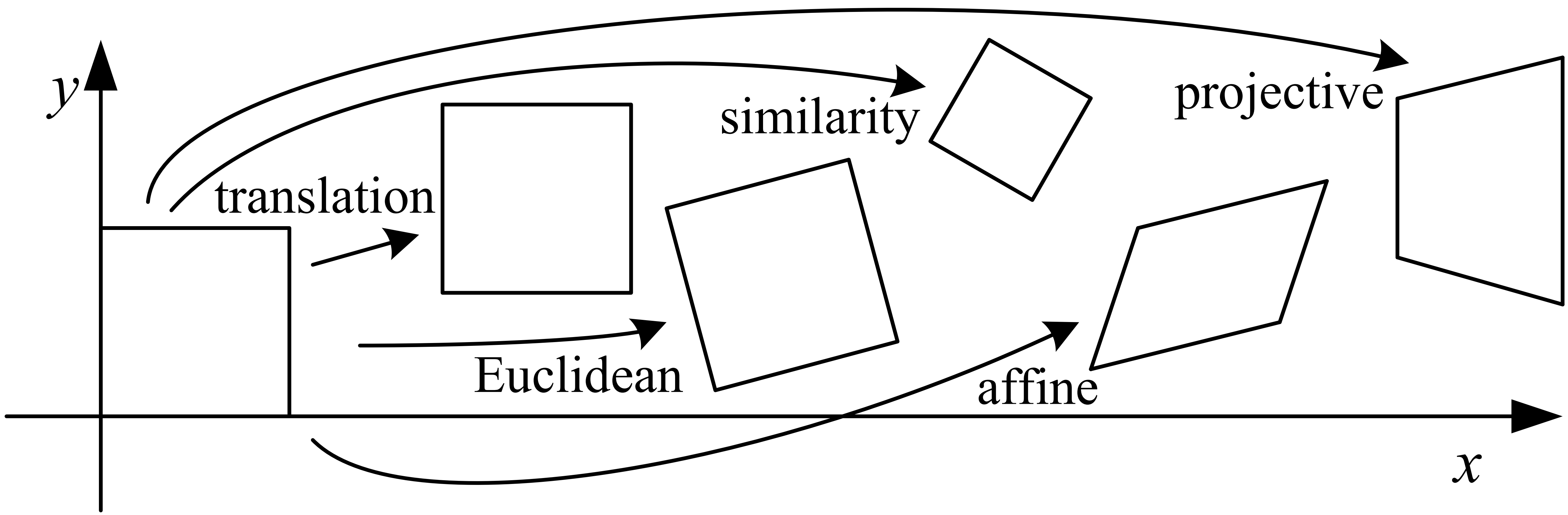
Planar Geometry

- 2D Linear + Projective transformations
  - Euclidean, Similarity, Affine, Homography
- Linear + Projective Cameras
  - Viewing a plane, rotating about a point

[Szeliski 2.1]

2D Transformations

- We will look at a family that can be represented by 3x3 matrices



This group represents perspective projections of **planar surfaces** in the world

Affine Transformations

- Transformed points are a linear function of the input points

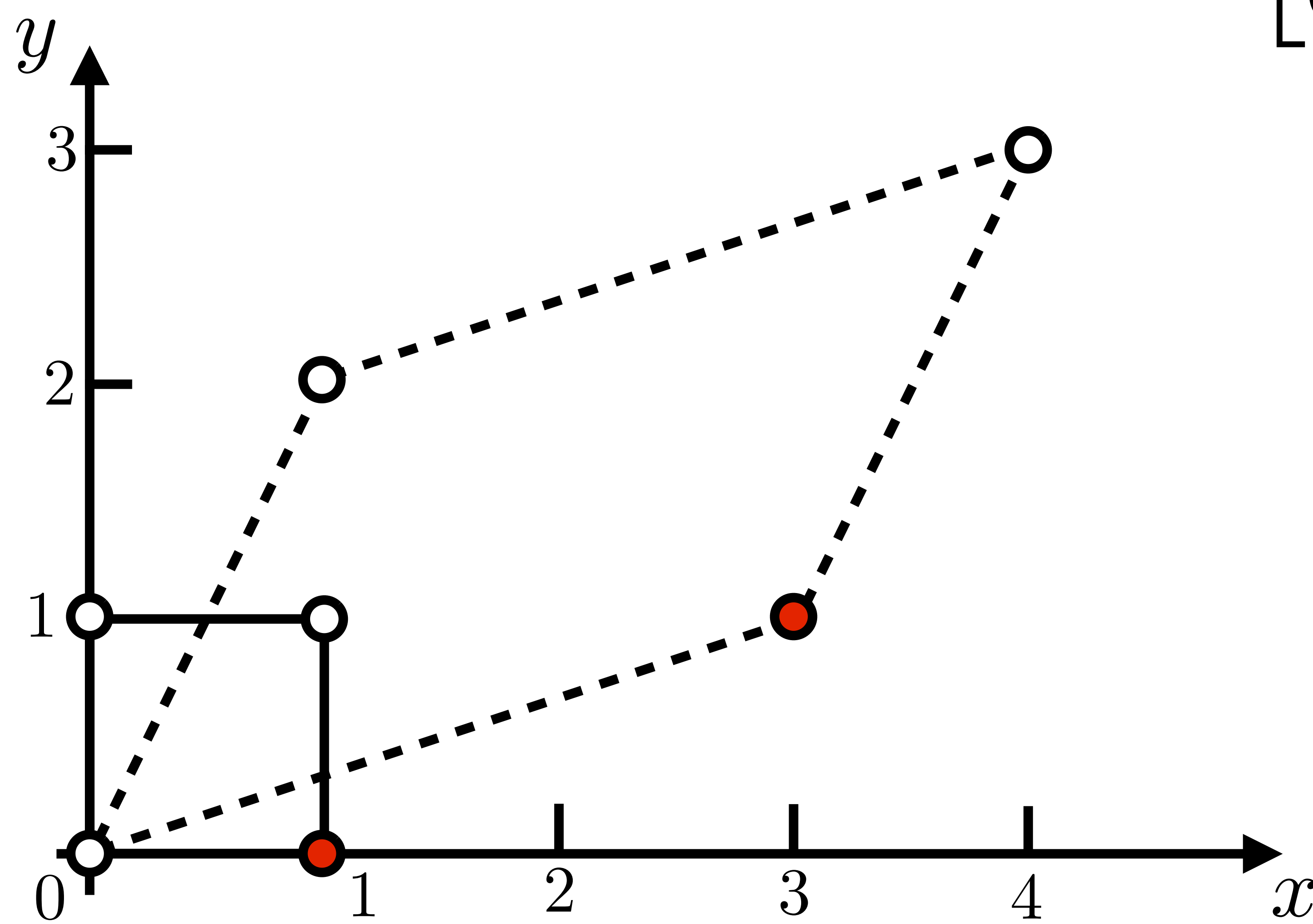
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_{13} \\ a_{23} \end{bmatrix}$$

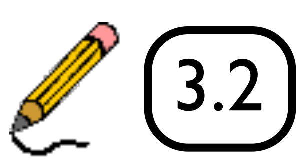
- This can be written as a single matrix multiplication

3.1

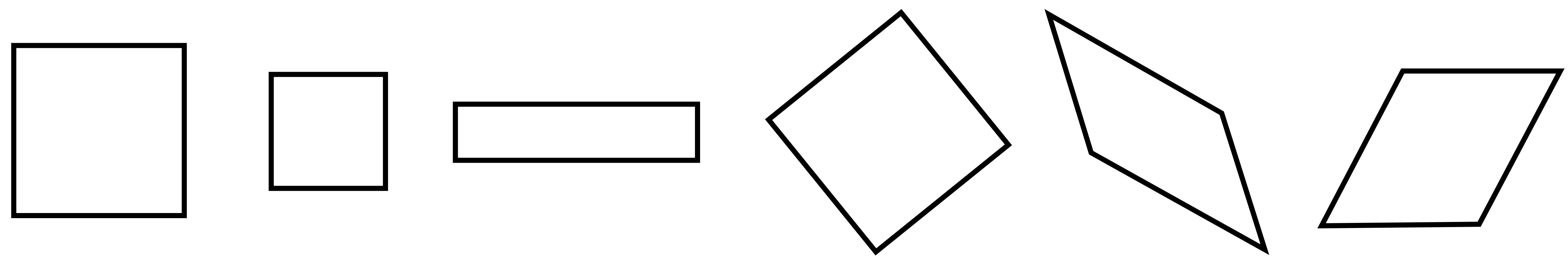
Linear Transformations

- Consider the action of the unit square under $\begin{bmatrix} 3 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

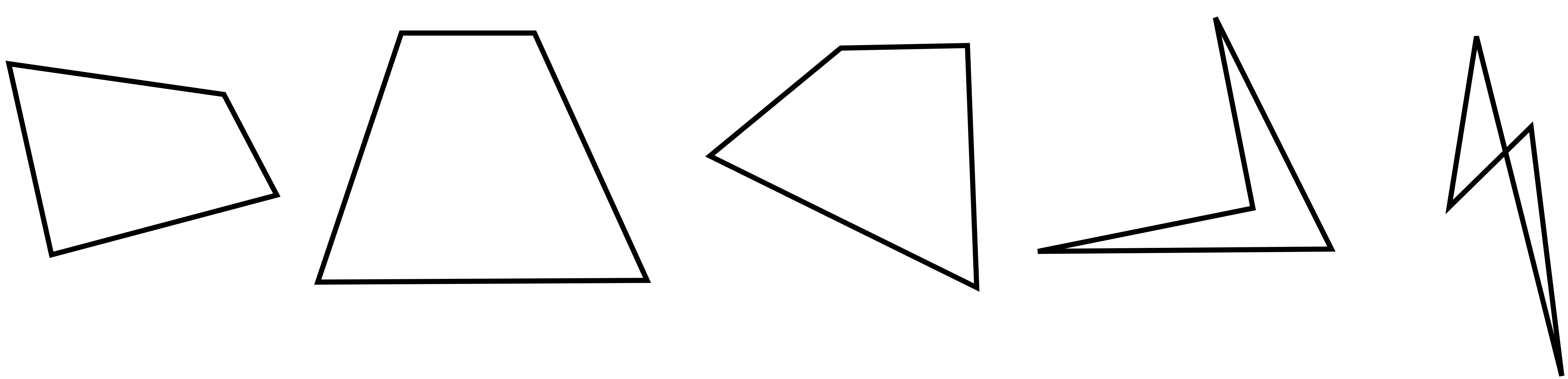




Linear Transform Examples



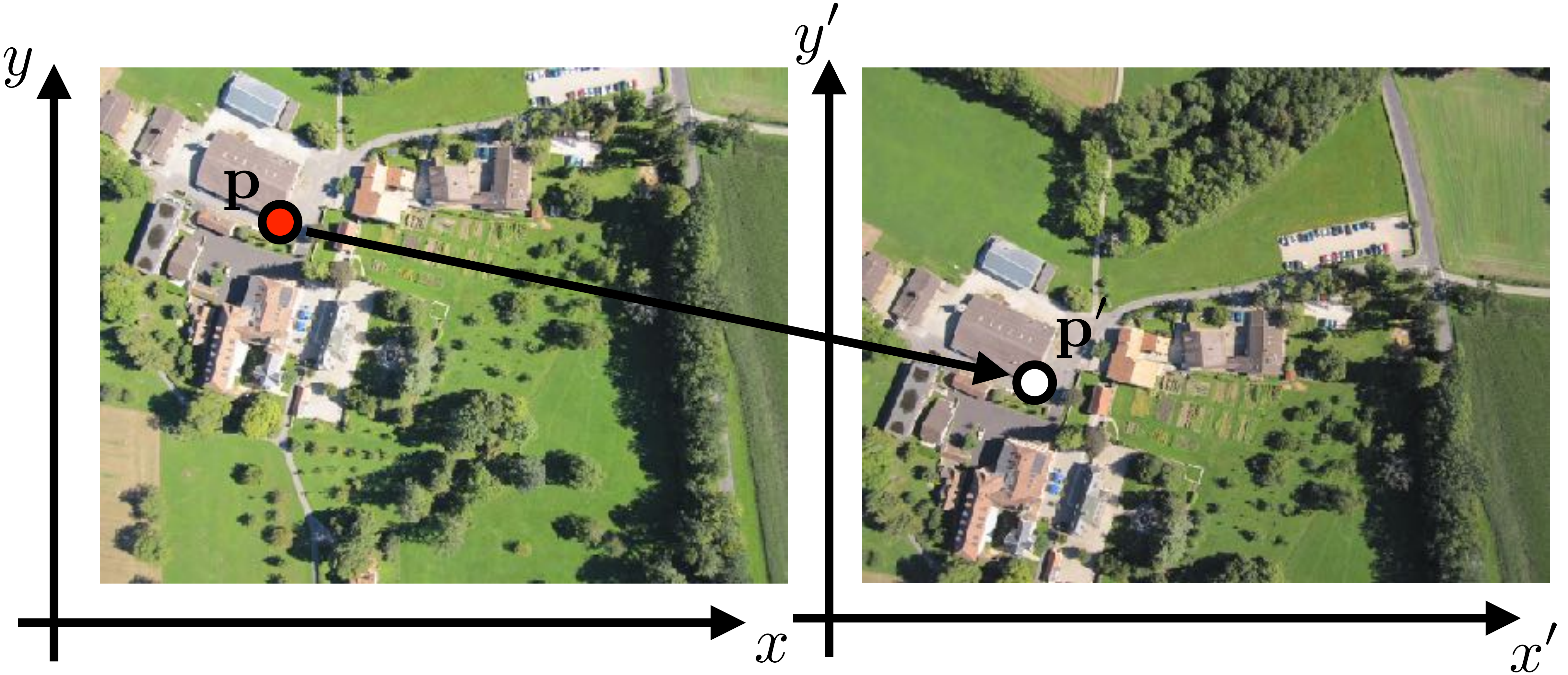
Translation, rotation, scale, shear (parallel lines preserved)



These transforms are not affine (parallel lines not preserved)

Linear Transformations

- Consider a single point correspondence



$$\begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

How many points are needed to solve for **a**?

Computing Affine Transforms

- Lets compute an affine transform from correspondences:

$$\begin{bmatrix} x_1' \\ y_1' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

- Re-arrange unknowns into a vector

3.3

Computing Affine Transforms

- Linear system in the unknown parameters $\mathbf{a}$

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \end{bmatrix}$$

- Of the form

$$\mathbf{Ma} = \mathbf{y}$$

Solve for $\mathbf{a}$ using Gaussian Elimination

Computing Affine Transforms

- We can now map any other points between the two images

$$\begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

Computing Affine Transforms

- Or resample one image in the coordinate system of the other

This allows us to “stitch” the two images

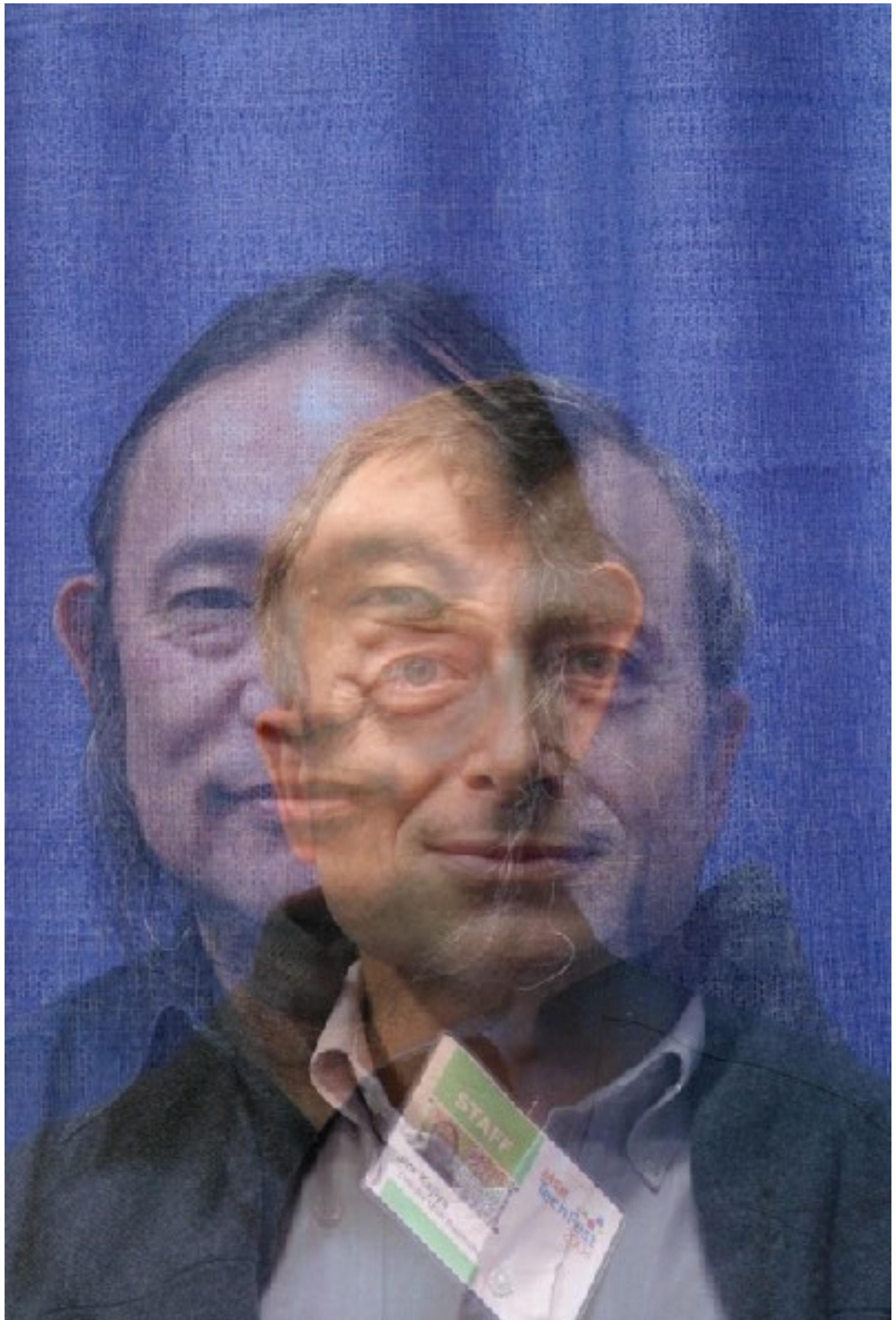
Linear Transformations

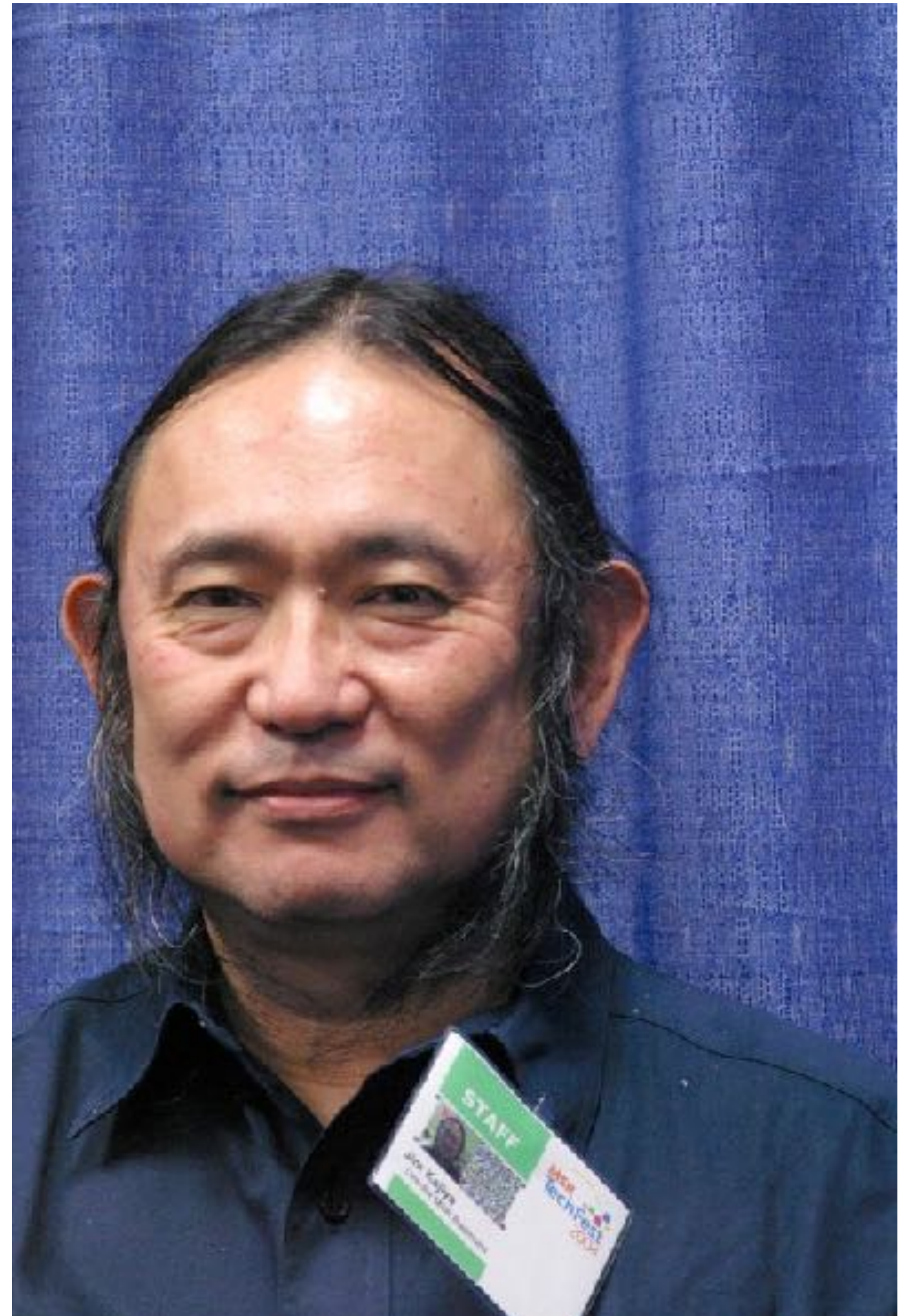
- Other linear transforms are special cases of affine

3.4

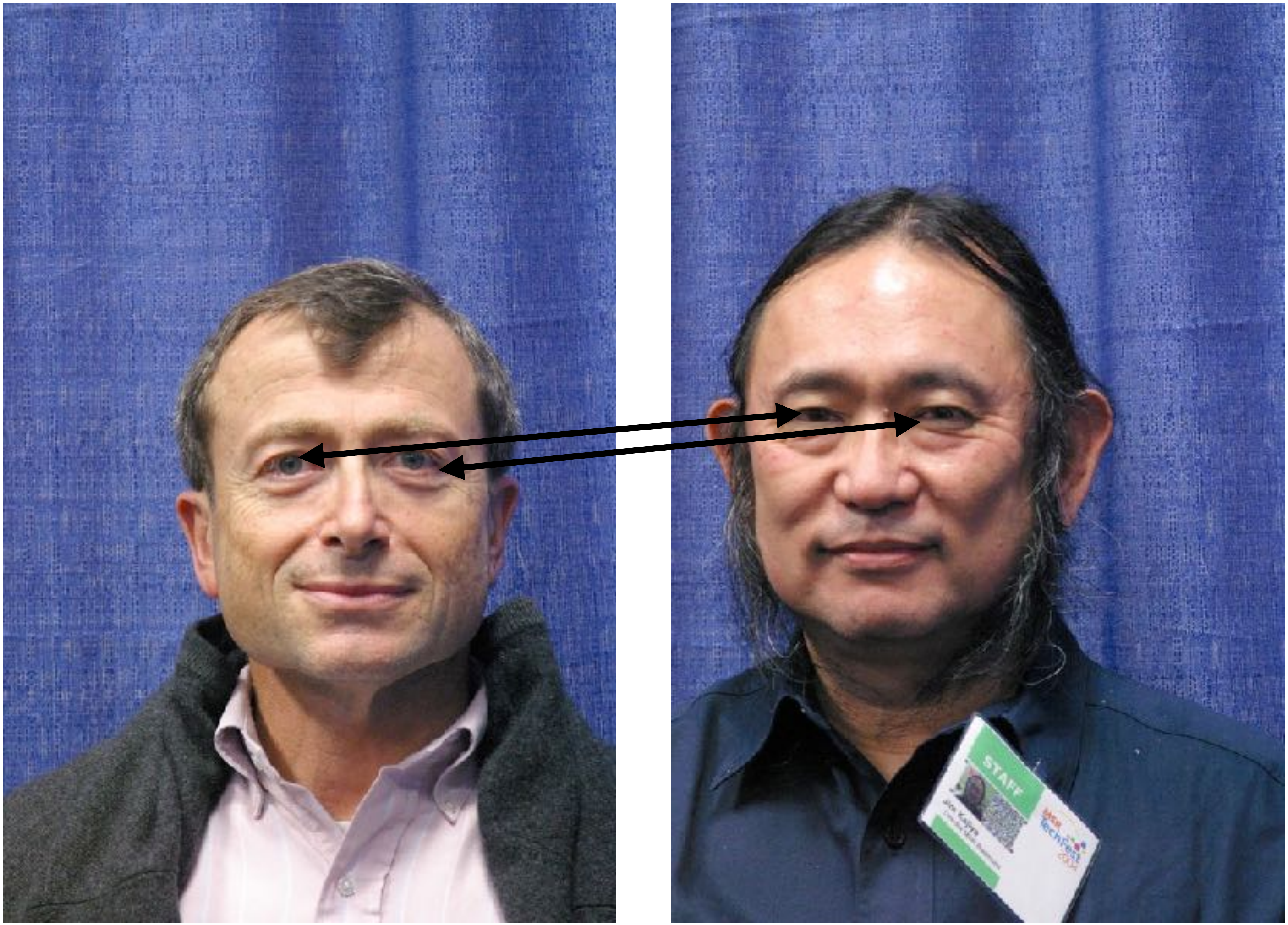
$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Face Alignment

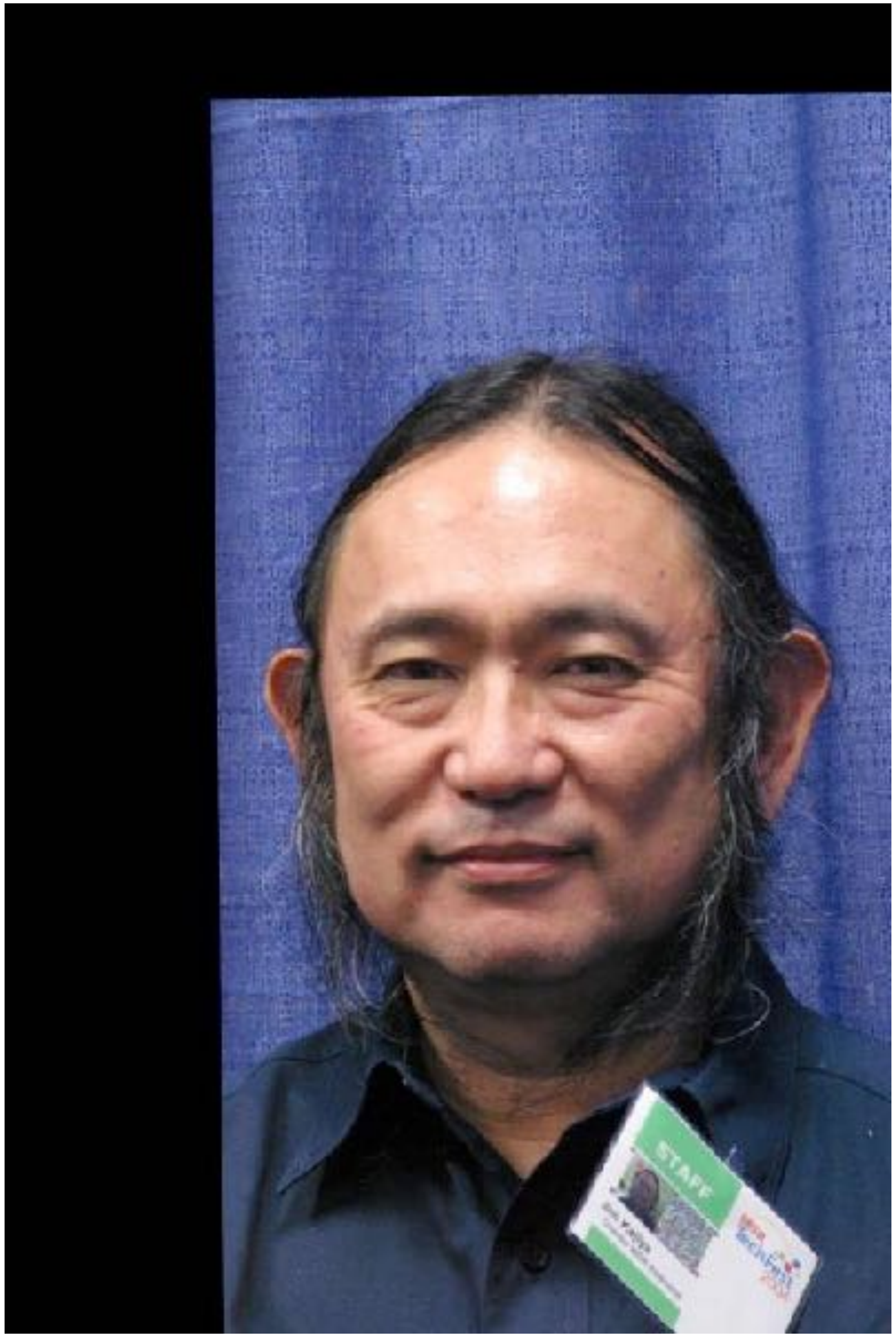


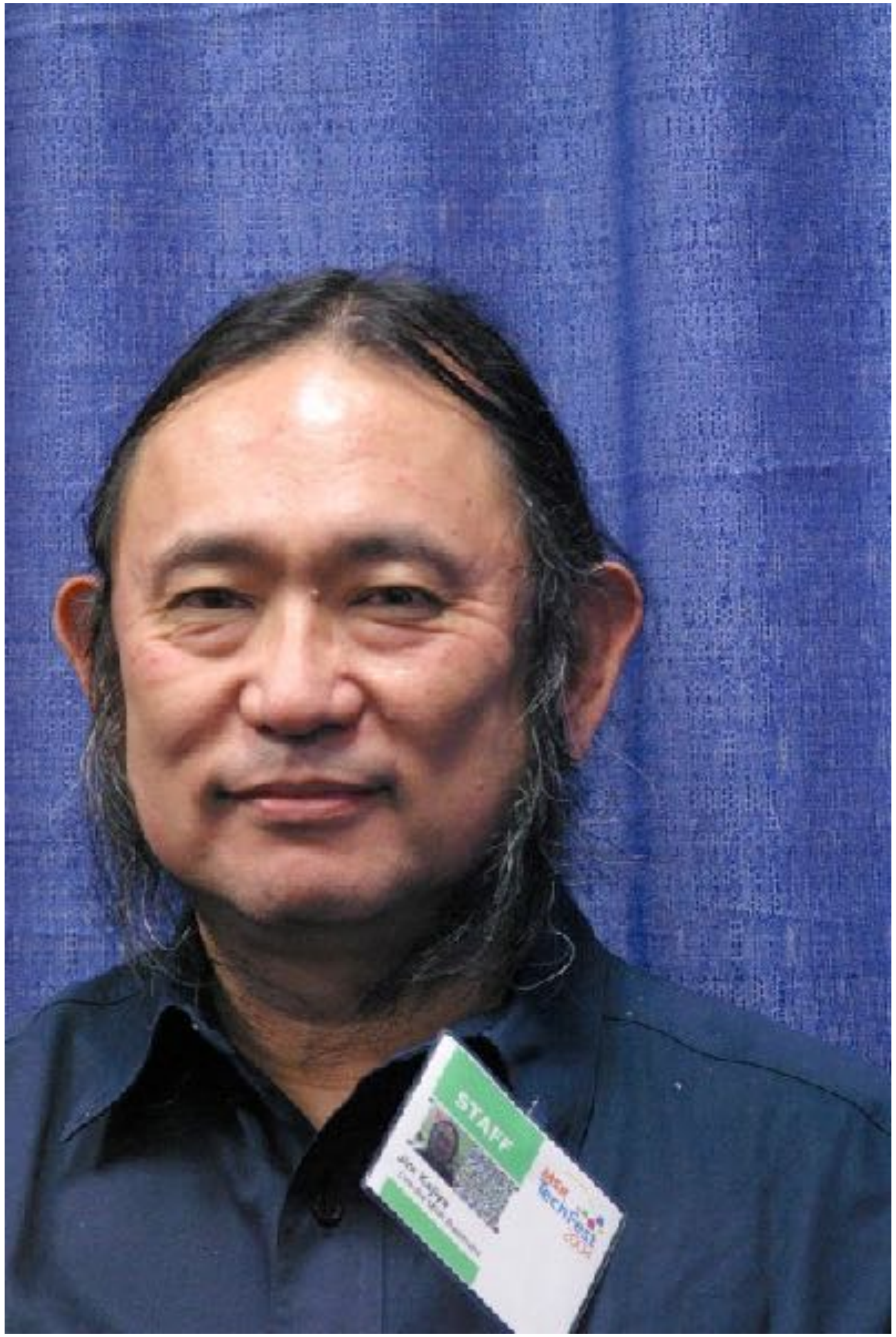


Face Alignment

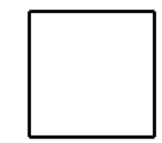
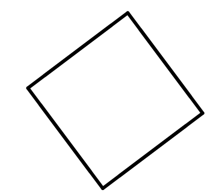
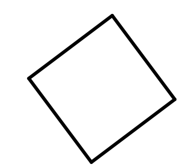
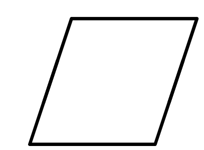
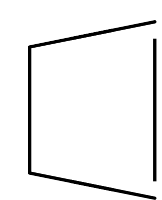


Face Alignment





2D Transformations

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	

Projective Transformation

- General 3x3 matrix transformation (note need scale factor)

$$s \begin{bmatrix} x'_1 \\ y'_1 \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

3.5

Project 2

P2

- Try out the **Image Warping Test** section in Project 2, particularly similarity, affine and projective transforms. You can also try warping with the inverse transform, e.g., using `P=np.linalg.inv(P)`

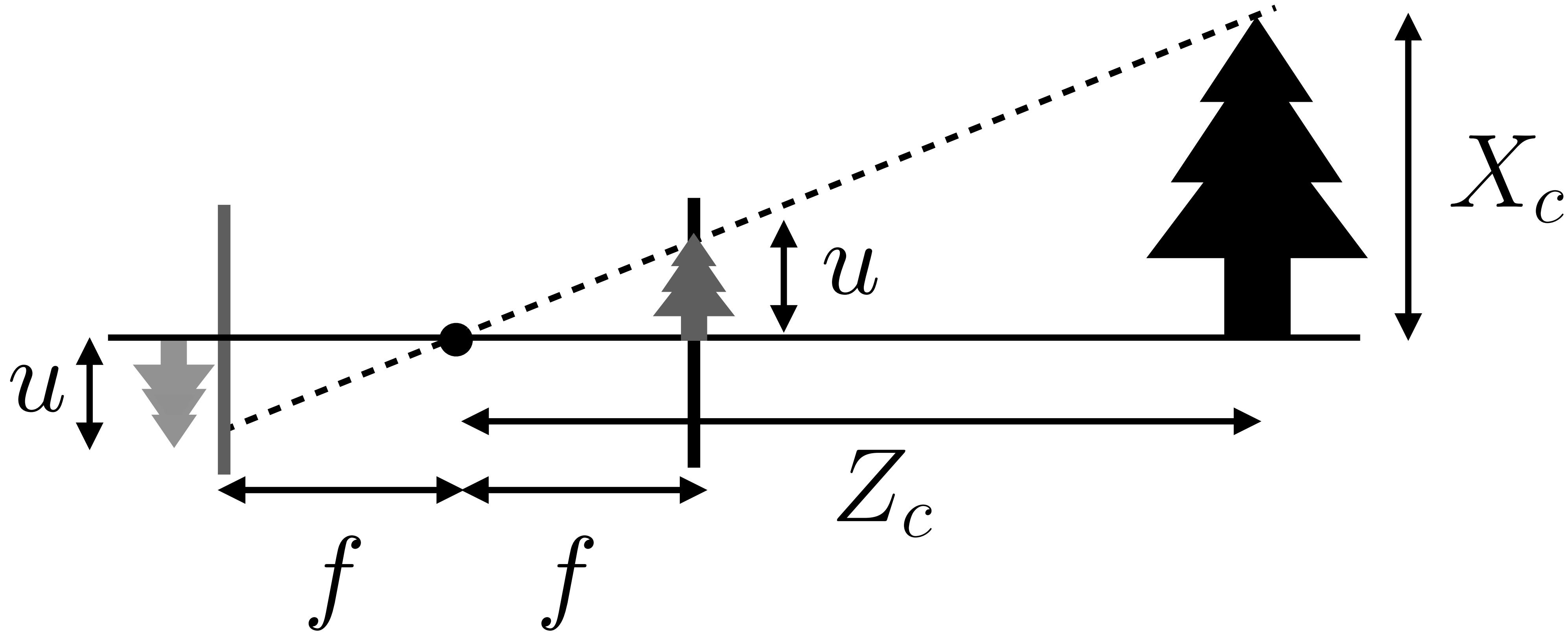
Camera Models + Geometry

- Pinhole camera, rigid body coordinate transforms
- Perspective, projective, linear/affine models
- Properties of cameras: viewing parallel lines, viewing a scene plane, rotating about a point

[T. Bacha]

Pinhole Camera

- Put the projection plane in front to avoid the 180° rotation



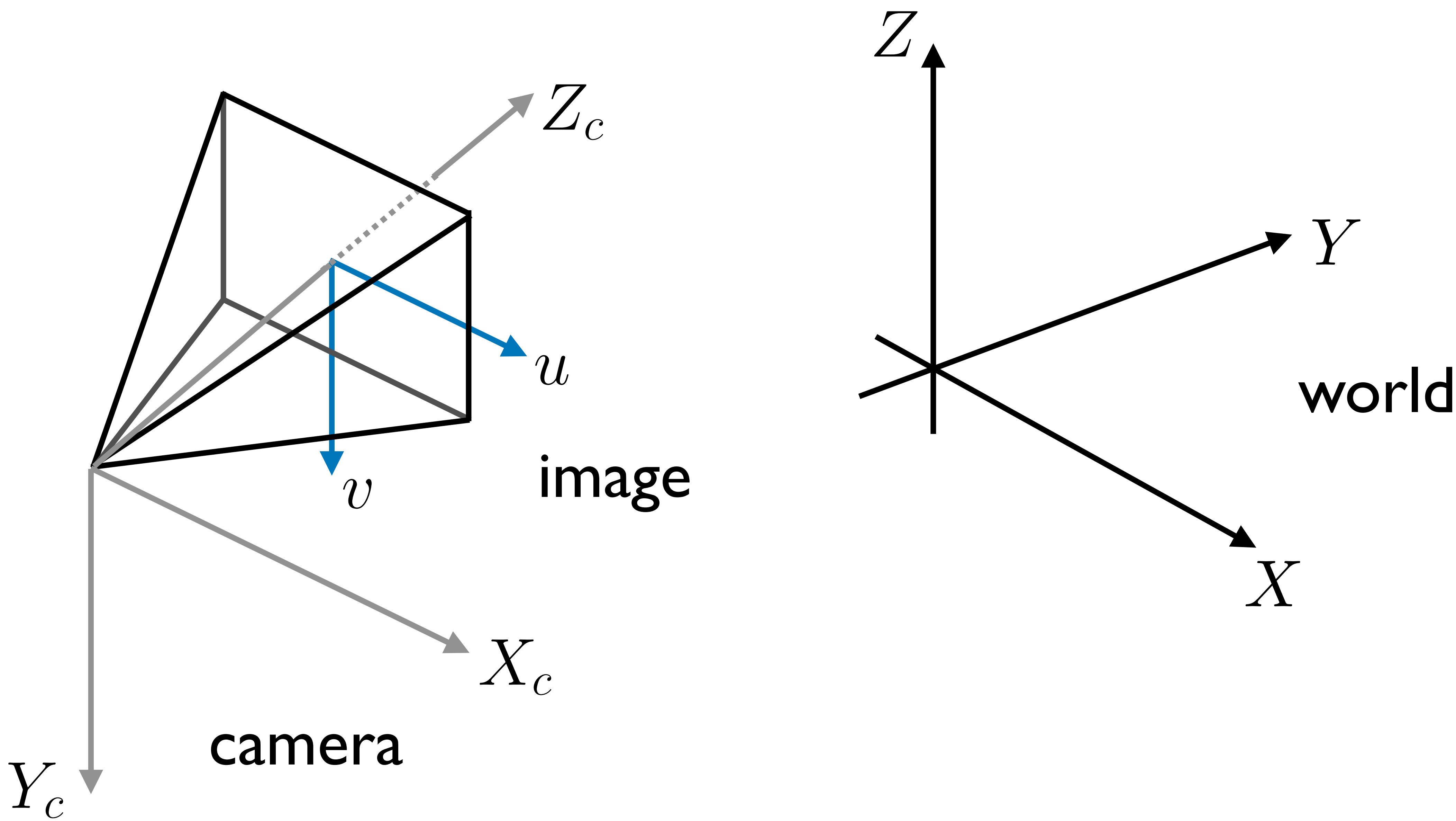
$$u = f X_c / Z_c$$
$$v = f Y_c / Z_c$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

- Note that X_c Y_c Z_c are **camera coordinates**

Perspective Camera

- Transform world to camera, to image coordinates



3.6

Projective Camera

- Perspective camera equation

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- Multiply and drop constraints to get a general 3x4 matrix

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

This is called a **projective camera**

How many degrees of freedom do these 2 models have?

Linear Camera

- Zero out bottom row to eliminate perspective division

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\downarrow$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

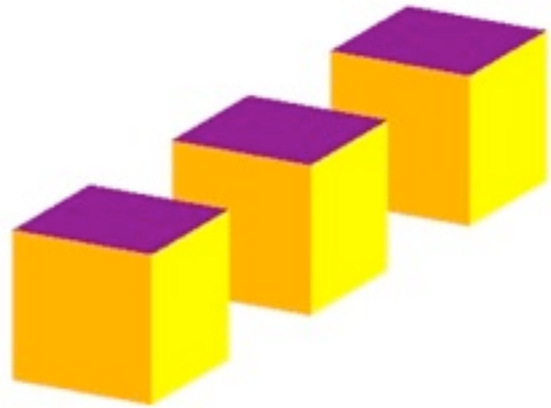
Linear a.k.a. affine camera

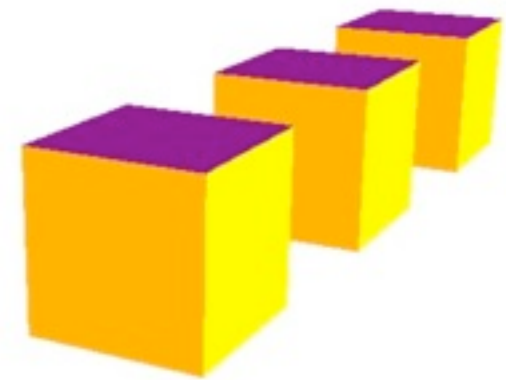
Linear vs Projective Cameras

- Consider a linear / affine camera viewing parallel world lines

 3.7

Linear vs Projective Cameras

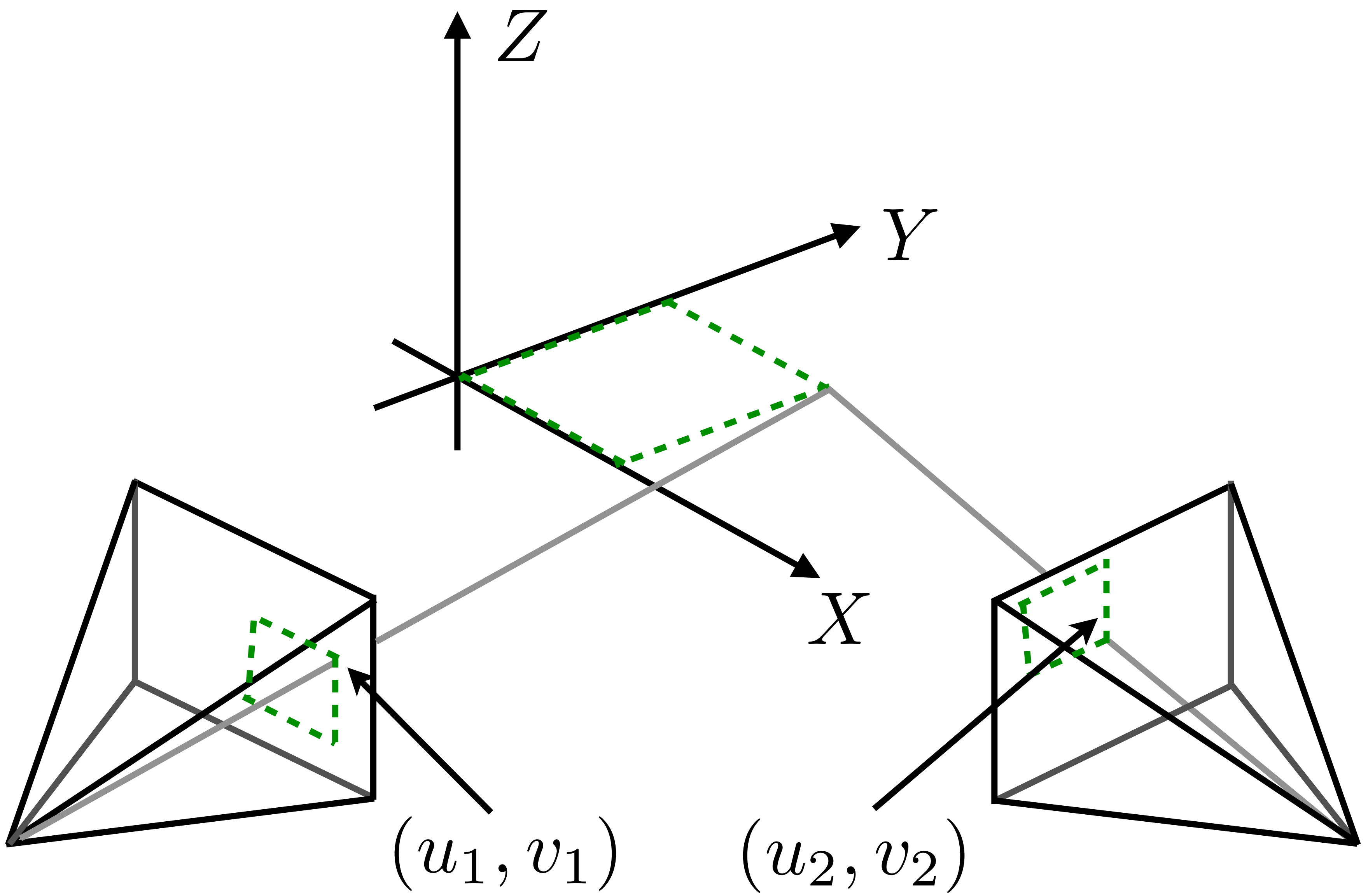




Parallelism preserved if depth variation in scene << depth of scene

Viewing a Plane

- Consider a pair of cameras viewing a plane



Without loss of generality, we can make it the world plane Z=0

Viewing a Plane

- Viewing the plane Z=0 with projective + linear cameras

$$s\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}p_{11} & p_{12} & \cancel{p_{13}} & p_{14}\\p_{21} & p_{22} & \cancel{p_{23}} & p_{24}\\p_{31} & p_{32} & \cancel{p_{33}} & p_{34}\end{bmatrix}\begin{bmatrix}X\\Y\\0\\1\end{bmatrix}=\begin{bmatrix}p_{11} & p_{12} & p_{14}\\p_{21} & p_{22} & p_{24}\\p_{31} & p_{32} & p_{34}\end{bmatrix}\begin{bmatrix}X\\Y\\1\end{bmatrix}$$

Projective — Homography

$$s\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}p_{11} & p_{12} & \cancel{p_{13}} & p_{14}\\p_{21} & p_{22} & \cancel{p_{23}} & p_{24}\\0 & 0 & \cancel{0} & 1\end{bmatrix}\begin{bmatrix}X\\Y\\0\\1\end{bmatrix}=\begin{bmatrix}p_{11} & p_{12} & p_{14}\\p_{21} & p_{22} & p_{24}\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}X\\Y\\1\end{bmatrix}$$

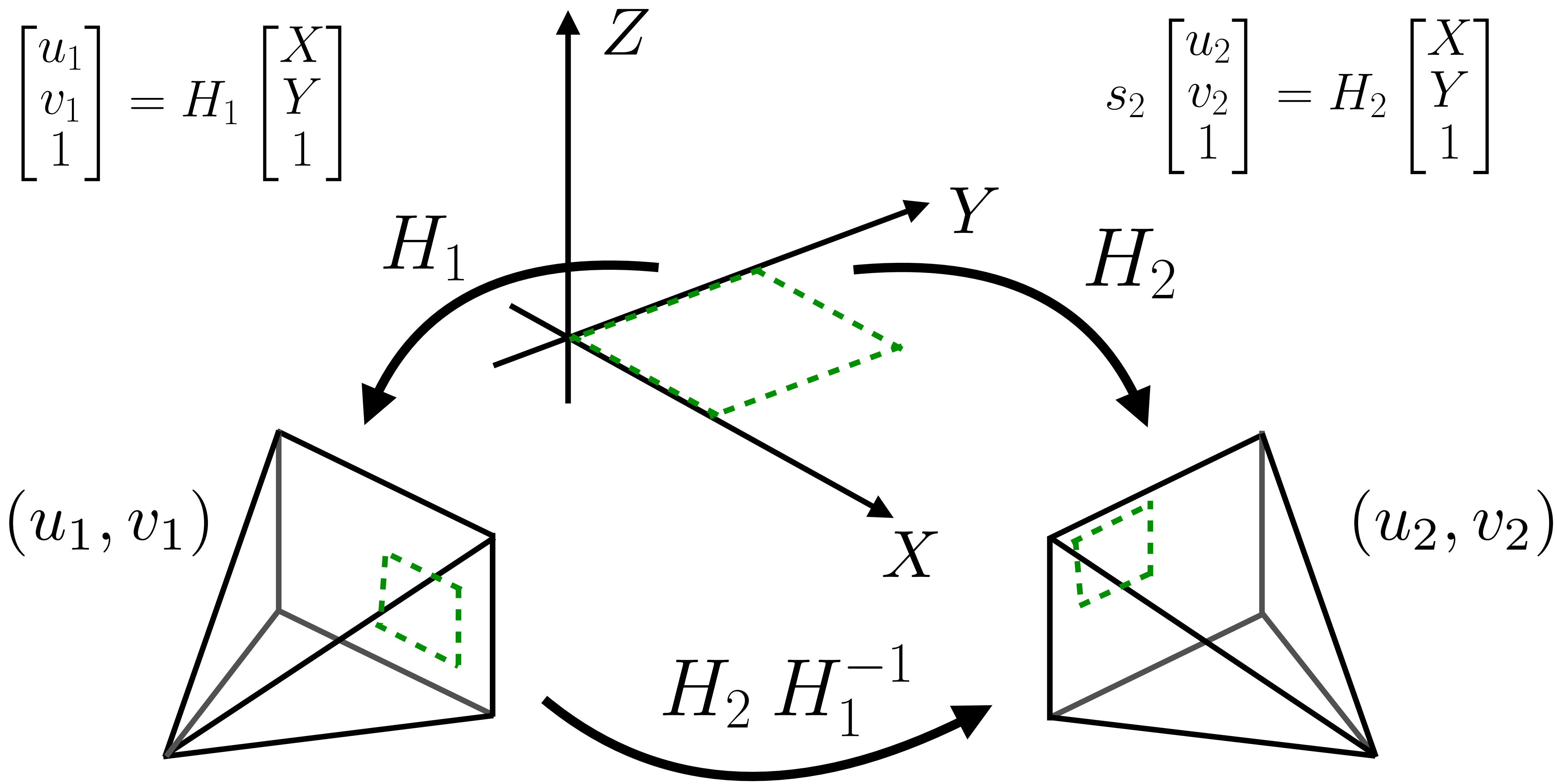
Linear — (2d) Affine

Viewing a Plane

- Consider a pair of cameras viewing a plane

$$s_1 \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = H_1 \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

$$s_2 \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = H_2 \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

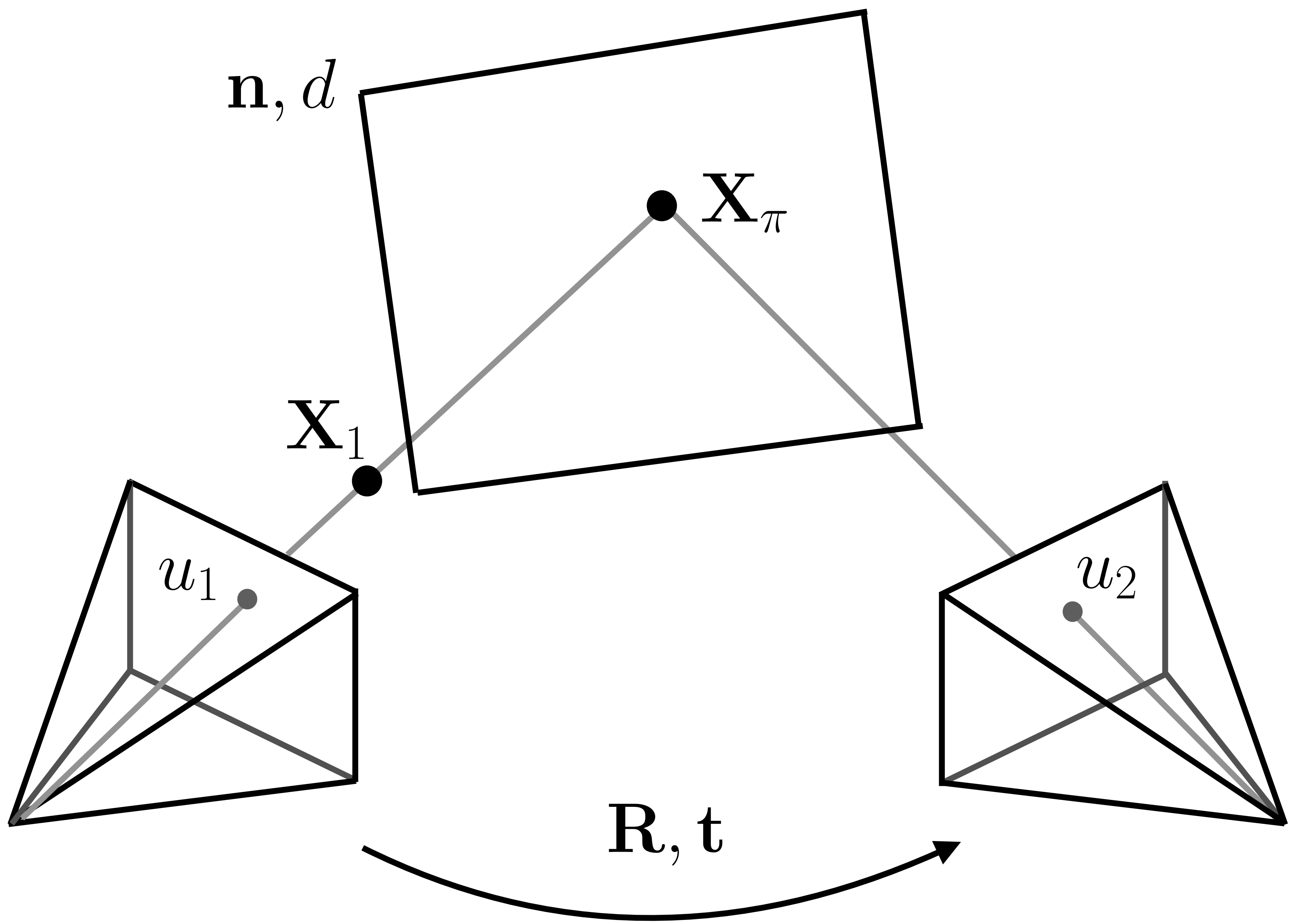


$$s_3 \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = H_2 \, H_1^{-1} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix}$$

Scene Plane

- What is the form of H in terms of scene parameters?

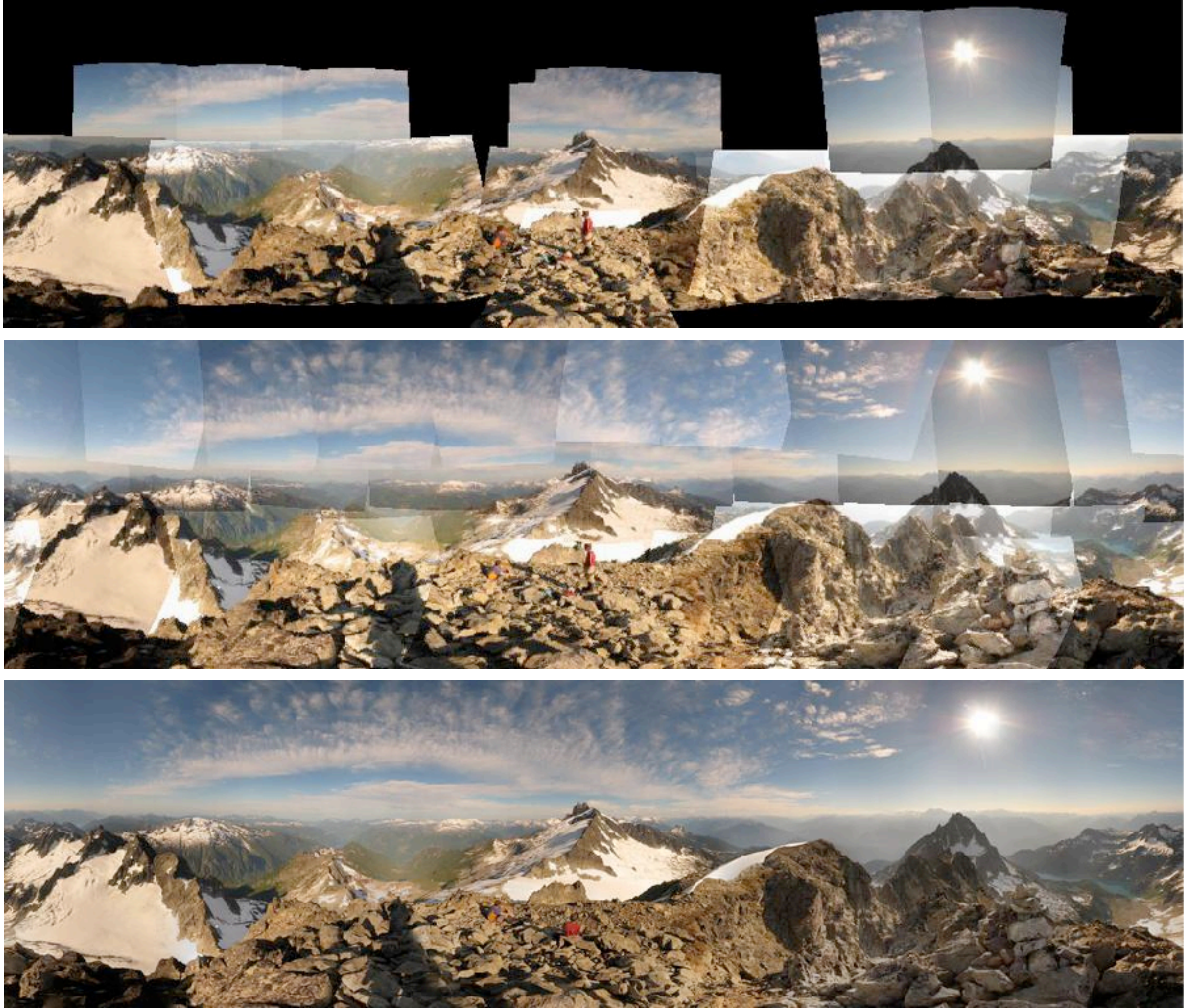
3.8



Camera Rotation

- What is the form of H in terms of scene parameters?

Hint: try setting t=0 in either the perspective camera or scene plane homography equations



Radial Distortion

- In perspective (rectilinear) projection, straight lines in the world map to straight lines in the image, but many real imagers exhibit distortion towards the image edges

"barrel" "pin cushion"

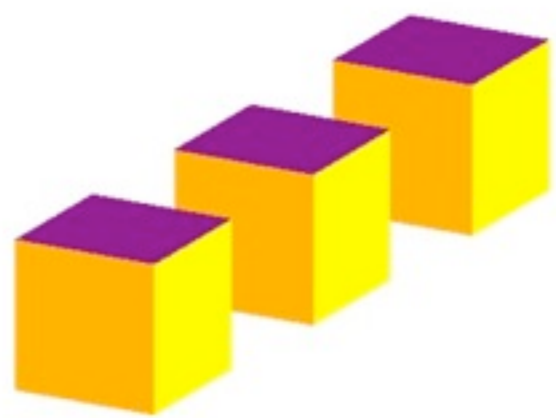
- A common first order model is $\mathbf{x}' = (1 + \kappa|\mathbf{x}|^2)\mathbf{x}$
- Wide-angle imagers may have very different projection models, e.g., for equidistant fisheye $r \propto \theta$

Linear/Affine

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

↓ viewing plane

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



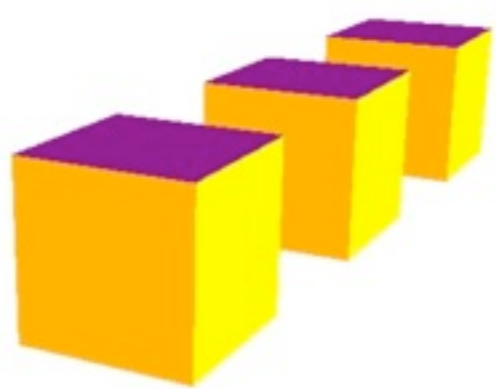
Projective

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

↓ viewing plane

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

(Homography)



Next Lecture

- RANSAC