

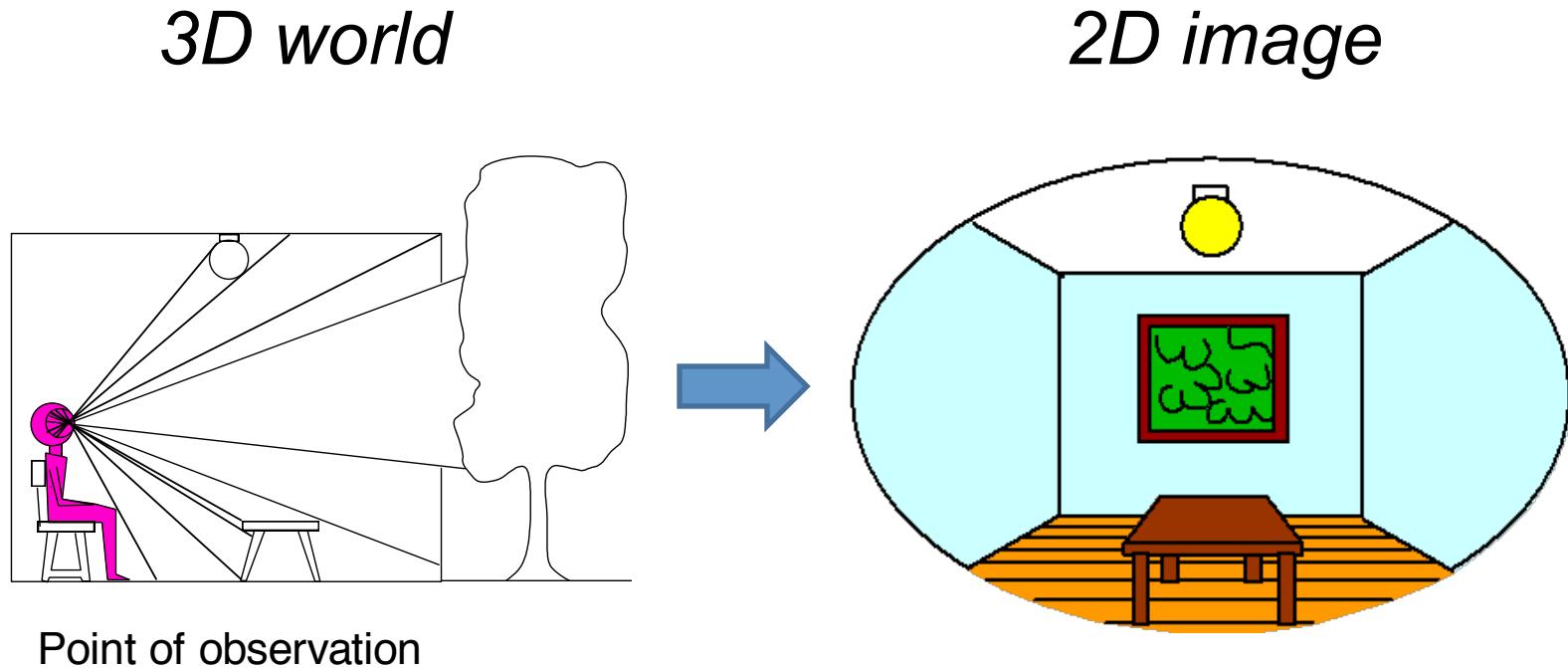


# Lecture 9 & 10: Stereo Vision

Dr. Juan Carlos Niebles  
Stanford AI Lab

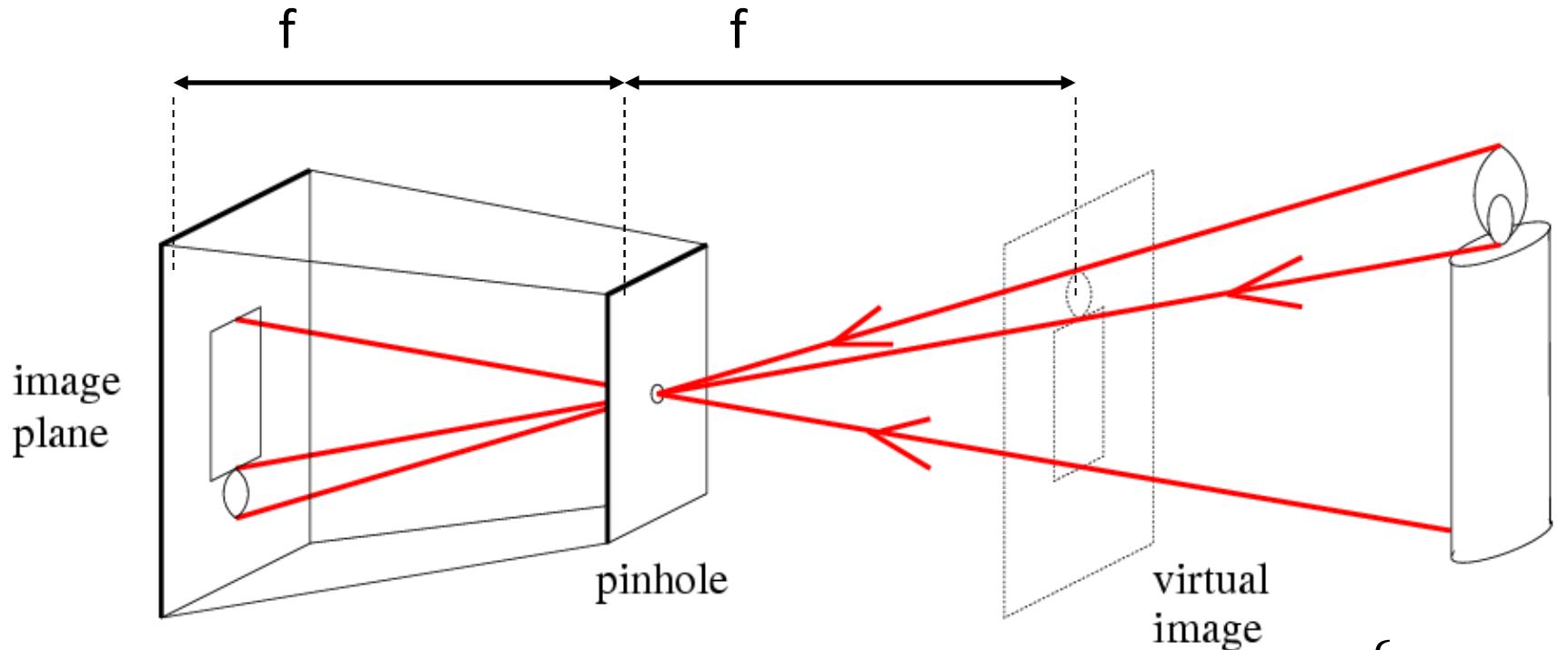
Professor Fei-Fei Li  
Stanford Vision Lab

# Dimensionality Reduction Machine (3D to 2D)



Figures © Stephen E. Palmer, 2002

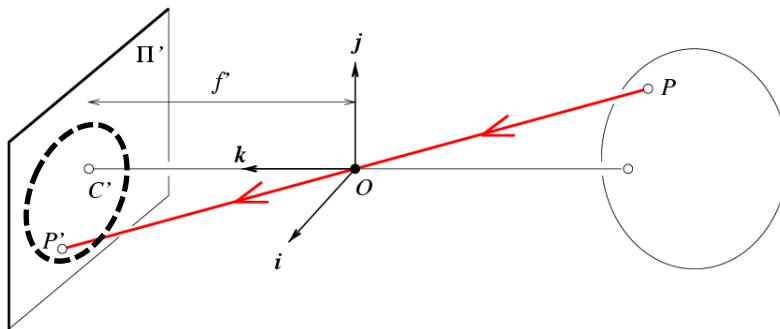
# Pinhole camera



- Common to draw image plane *in front* of the focal point
- Moving the image plane merely scales the image.

$$\begin{cases} x' = f \frac{x}{z} \\ y' = f \frac{y}{z} \end{cases}$$

# Relating a real-world point to a point on the camera



In homogeneous coordinates:

$$P' = \begin{bmatrix} f & x \\ f & y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

ideal world

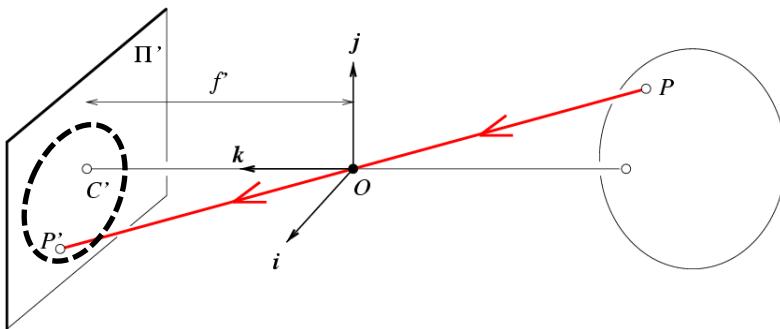
## Intrinsic Assumptions

- Unit aspect ratio
- Optical center at  $(0,0)$
- No skew

## Extrinsic Assumptions

- No rotation
- Camera at  $(0,0,0)$

# Relating a real-world point to a point on the camera



In homogeneous coordinates:

$$P' = \begin{bmatrix} f & x \\ f & y \\ z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = K \begin{bmatrix} I & 0 \end{bmatrix} P$$

$K$

## Intrinsic Assumptions

- Unit aspect ratio
- Optical center at  $(0,0)$
- No skew

## Extrinsic Assumptions

- No rotation
- Camera at  $(0,0,0)$

# Real-world camera

## Intrinsic Assumptions

- Optical center at  $(u_0, v_0)$
- Rectangular pixels
- Small skew

## Extrinsic Assumptions

- No rotation
- Camera at  $(0,0,0)$

$P' = K \begin{bmatrix} I & 0 \end{bmatrix} P \rightarrow w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$

**Intrinsic parameters**

Slide inspiration: S. Savarese

# Real-world camera + Real-world transformation

## Intrinsic Assumptions

- Optical center at  $(u_0, v_0)$
- Rectangular pixels
- Small skew

## Extrinsic Assumptions

- Allow rotation
- Camera at  $(t_x, t_y, t_z)$

$$P' = K \begin{bmatrix} R & \bar{t} \end{bmatrix} P \quad \xrightarrow{\text{blue arrow}} \quad w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Slide inspiration: S. Savarese

# What we will learn today?

- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

[HZ] Chapters: 4, 9, 11

[FP] Chapters: 10

# What we will learn today?

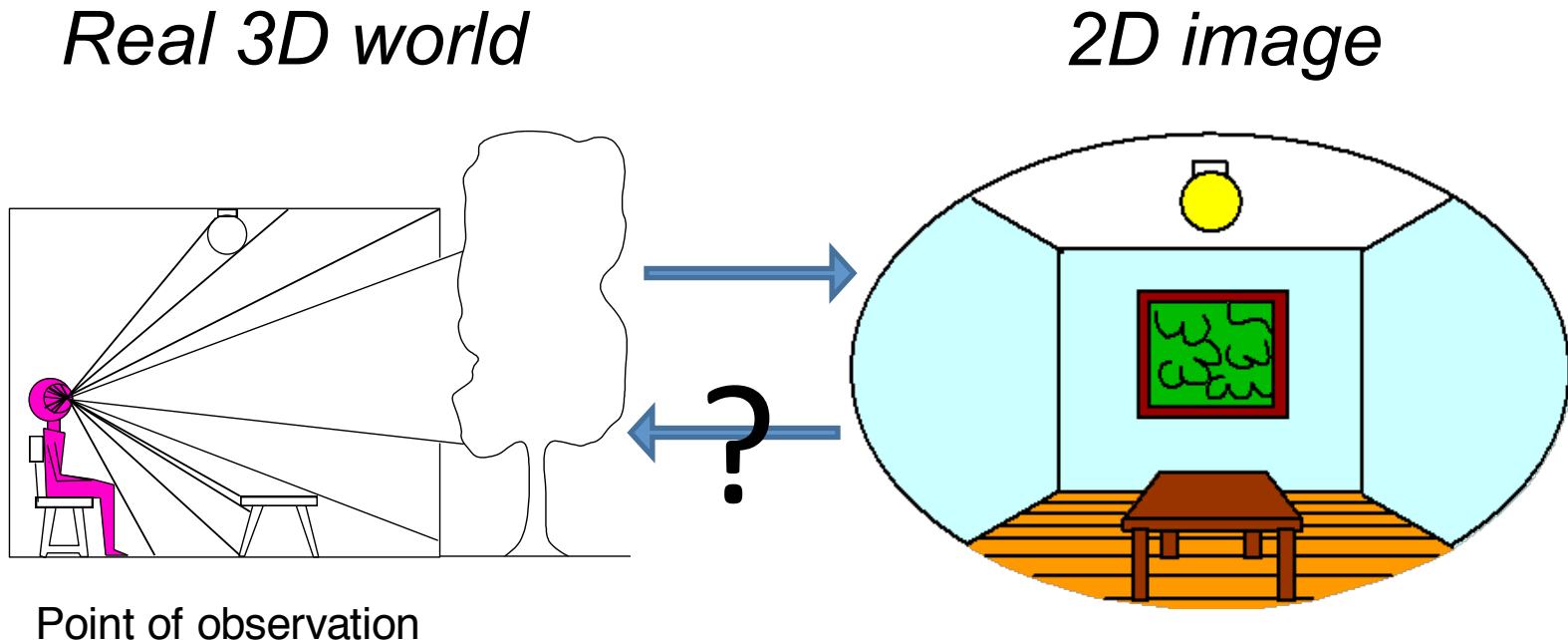
- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

[HZ] Chapters: 4, 9, 11  
[FP] Chapters: 10

# Recovering 3D from Images

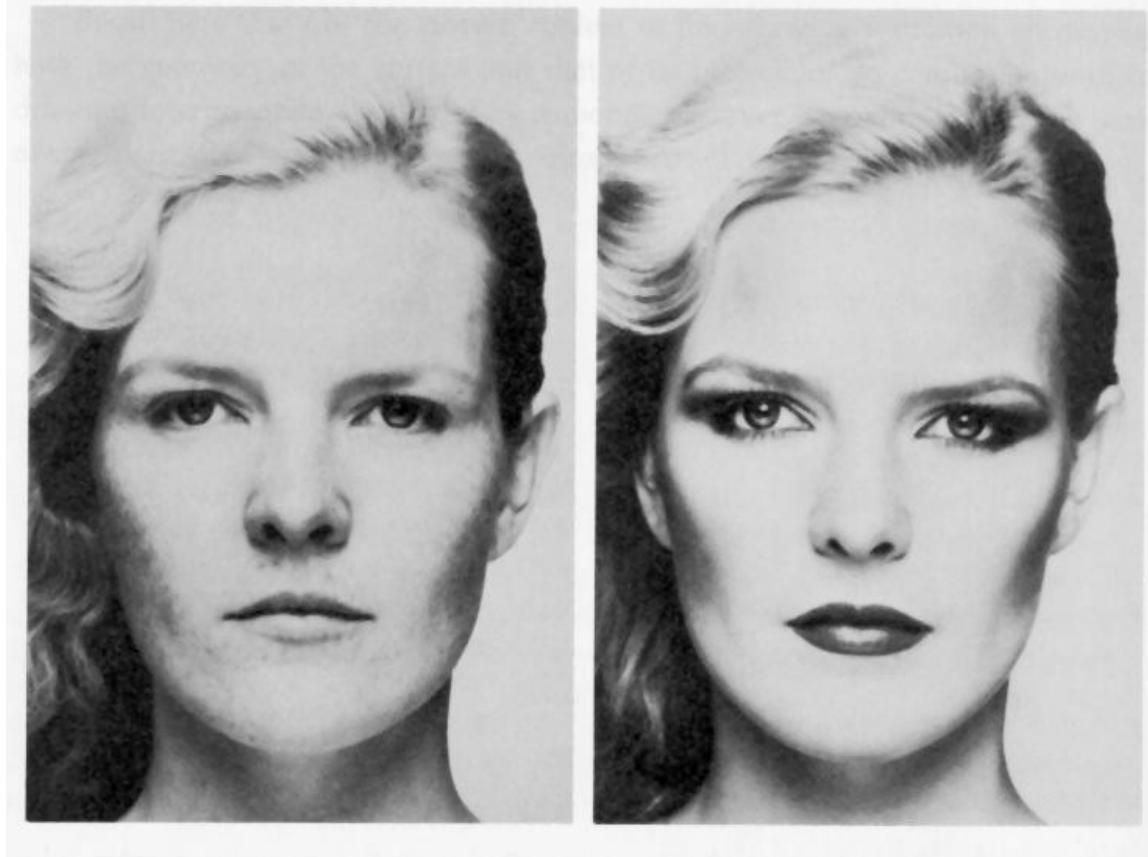
- How can we automatically compute 3D geometry from images?
  - What cues in the image provide 3D information?



# Visual Cues for 3D

---

- Shading

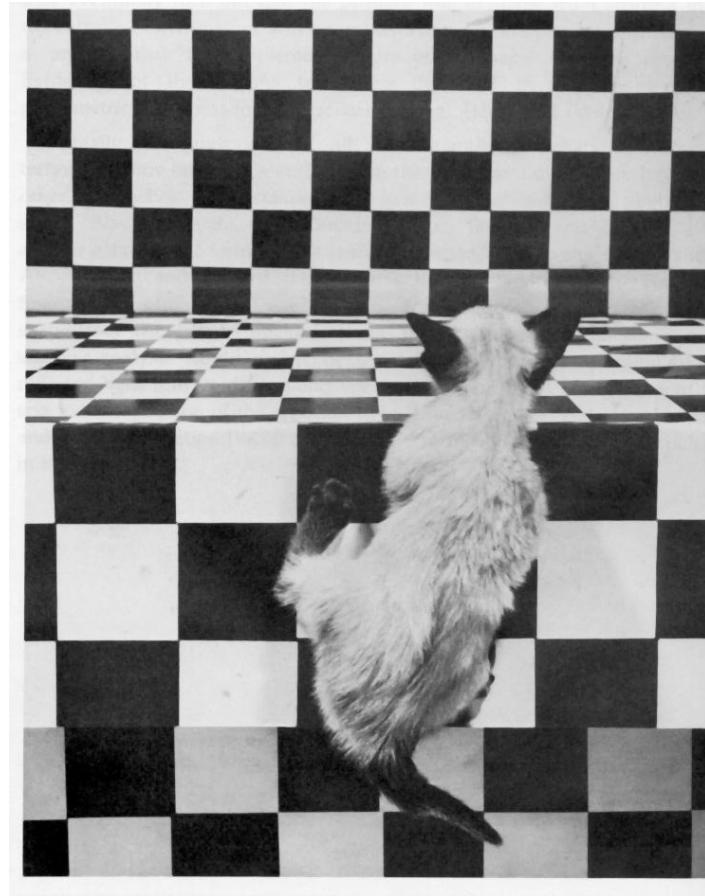


**Merle Norman Cosmetics, Los Angeles**

# Visual Cues for 3D

---

- Shading
- Texture



*The Visual Cliff, by William Vandivert, 1960*

# Visual Cues for 3D

---

- Shading
- Texture
- Focus



From *The Art of Photography, Canon*

# Visual Cues for 3D

---

- Shading
- Texture
- Focus
- Motion



# Visual Cues for 3D

---

- Shading
  - Texture
  - Focus
  - Motion
- Others:
    - Highlights
    - Shadows
    - Silhouettes
    - Inter-reflections
    - Symmetry
    - Light Polarization
    - ...

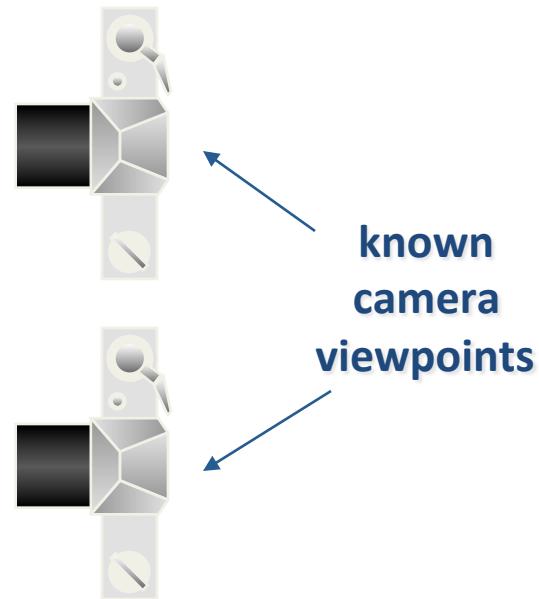
## Shape From X

- $X = \text{shading, texture, focus, motion, ...}$
- We'll focus on the motion cue

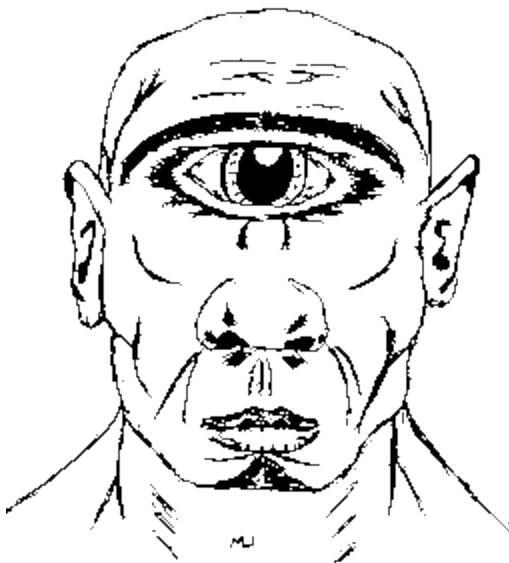
# Stereo Reconstruction

---

- The Stereo Problem
  - Shape from two (or more) images
  - Biological motivation



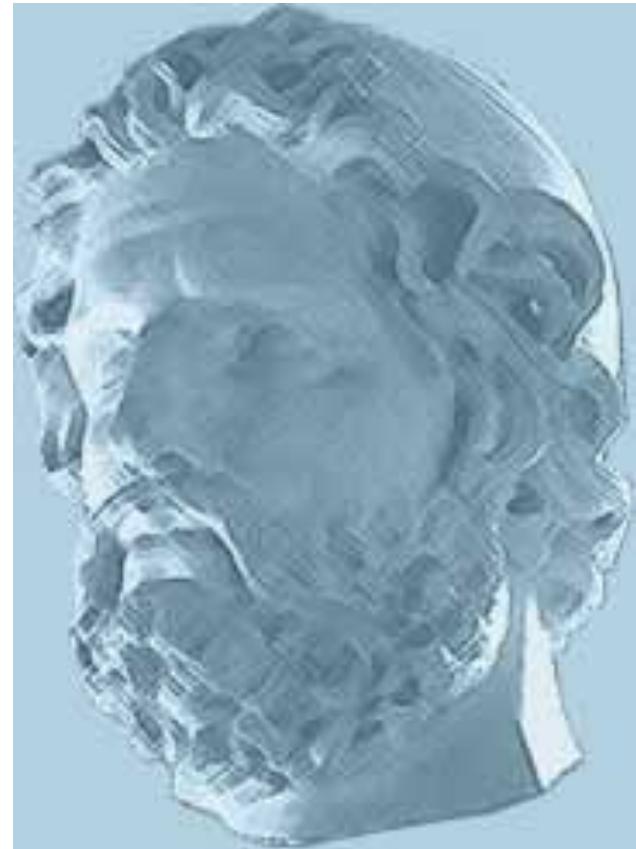
# Why do we have two eyes?



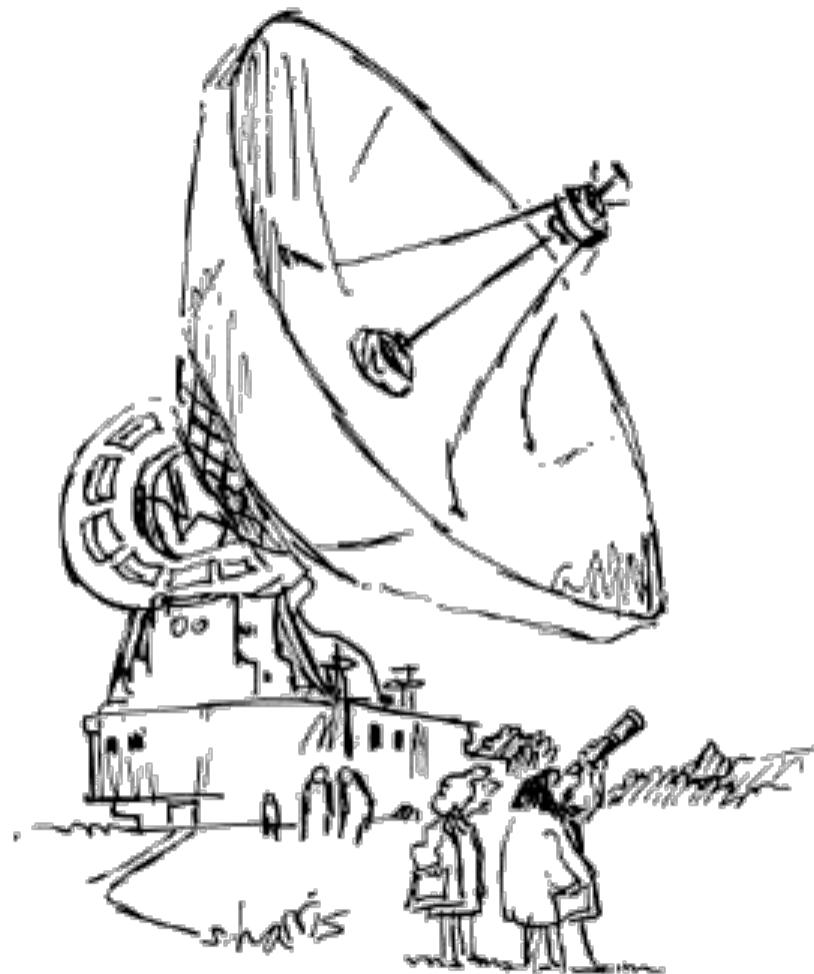
**Cyclope**

**vs.**

**Odysseus**



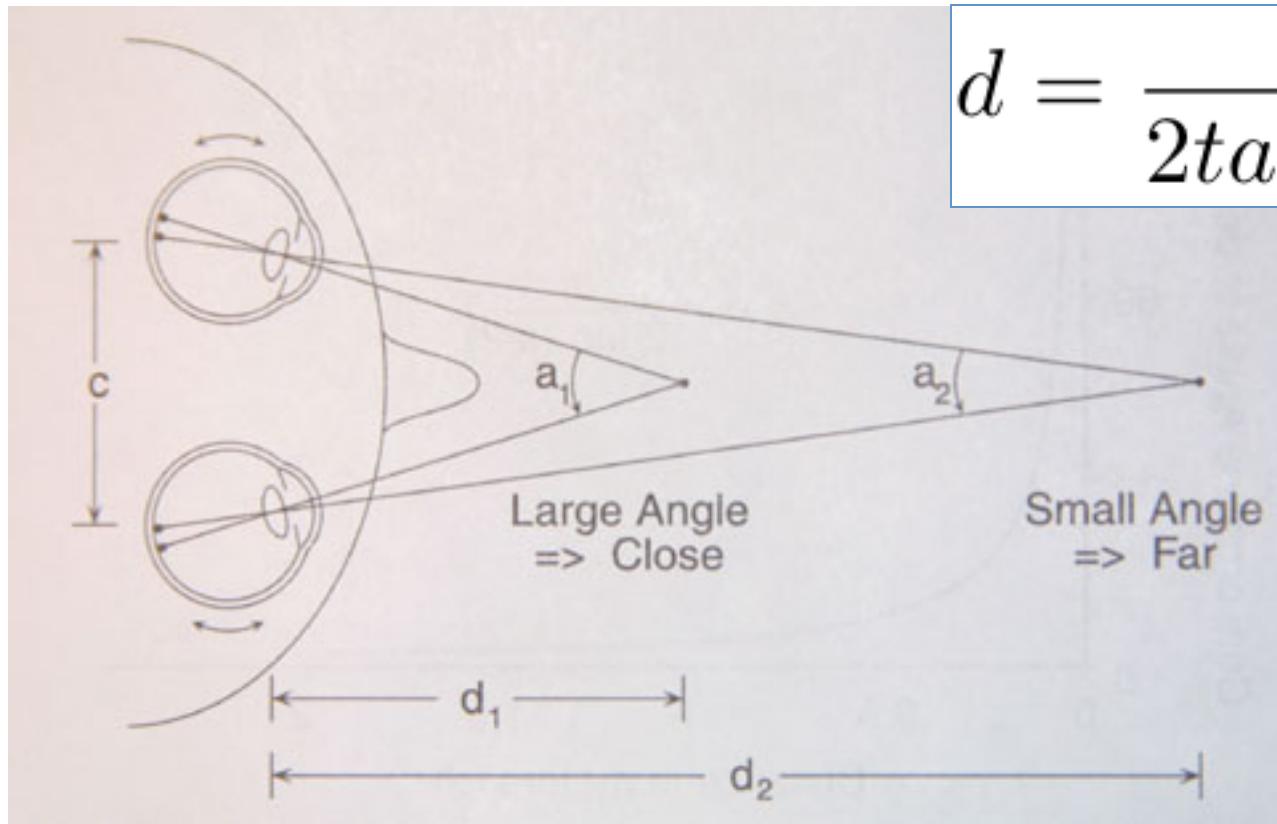
# 1. Two is better than one



"Just checking."

## 2. Depth from Convergence

$$d = \frac{c}{2\tan(a/2)}$$



*Human performance: up to 6-8 feet*

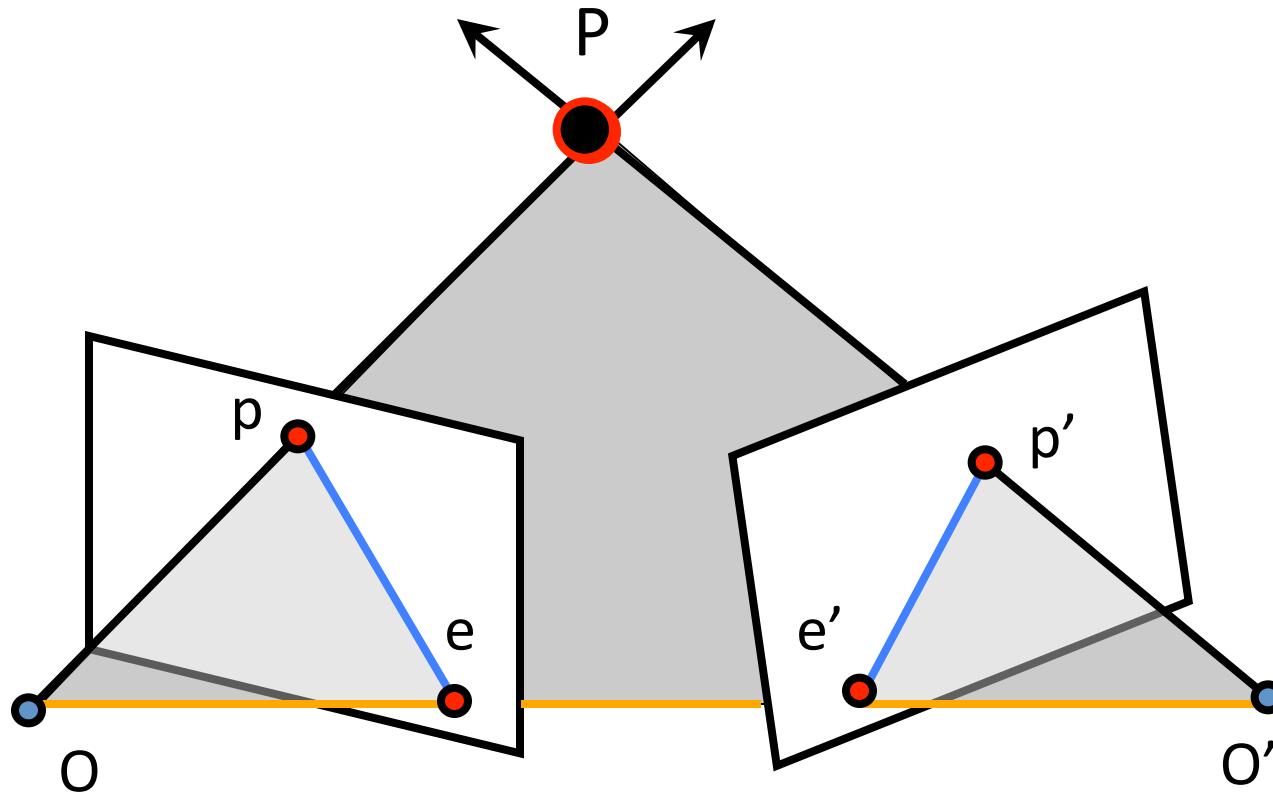
# What we will learn today?

- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

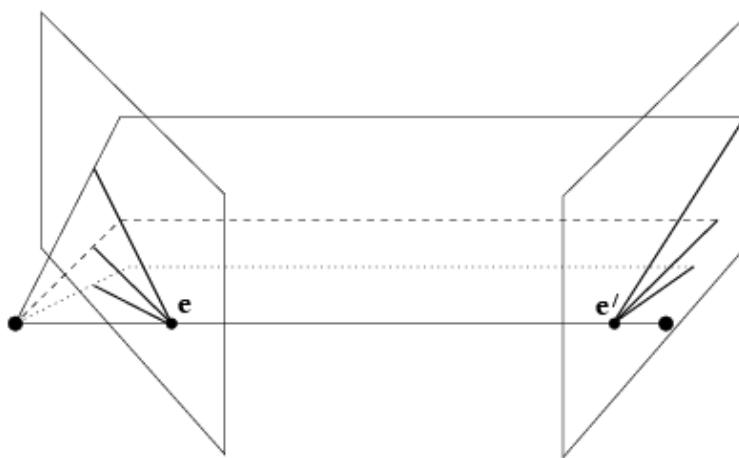
[HZ] Chapters: 4, 9, 11  
[FP] Chapters: 10

# Epipolar geometry

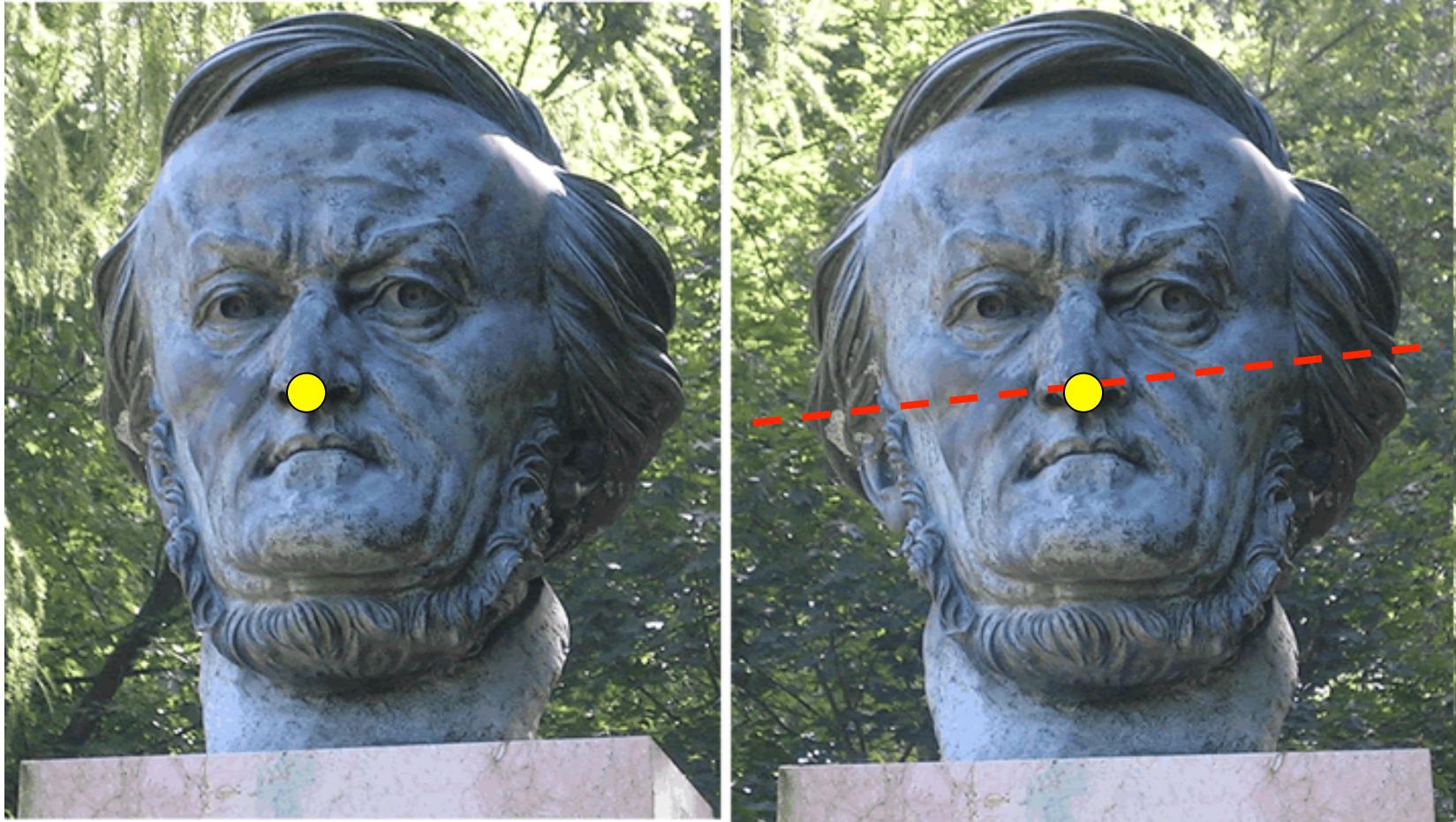


- Epipolar Plane
- Baseline
- Epipolar Lines
- Epipoles  $e, e'$ 
  - = intersections of baseline with image planes
  - = projections of the other camera center
  - = vanishing points of camera motion direction

# Example: Converging image planes

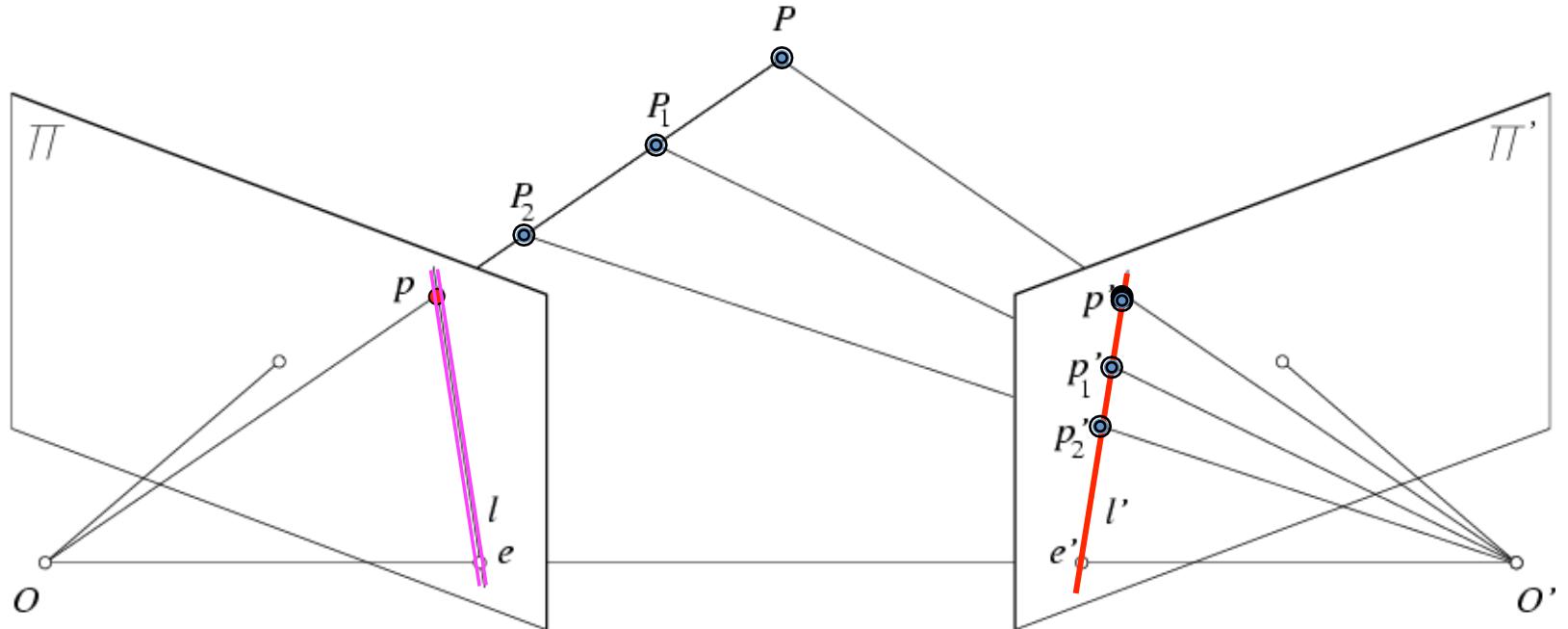


# Epipolar Constraint



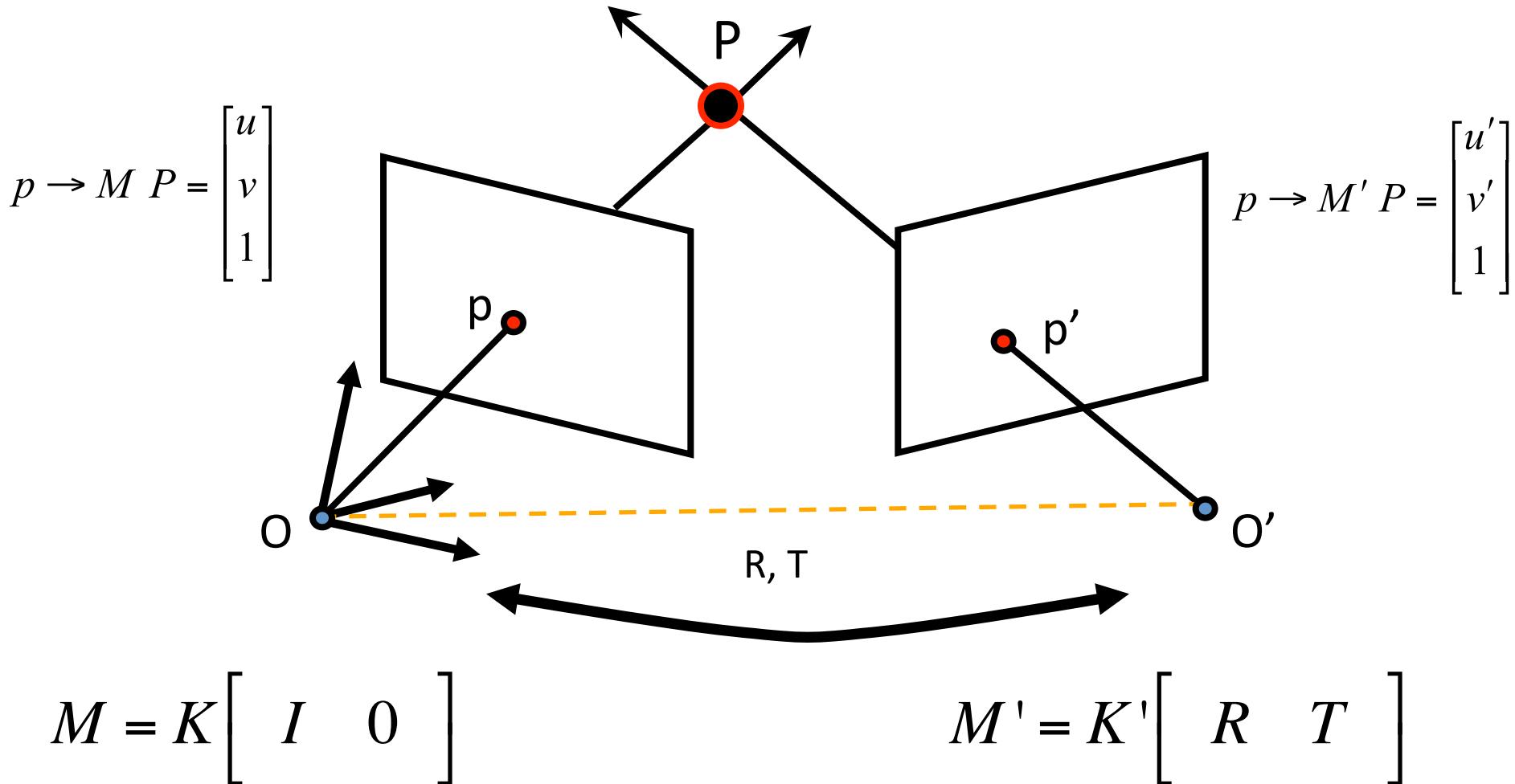
- Two views of the same object
- Suppose I know the camera positions and camera matrices
- Given a point on left image, how can I find the corresponding point on right image?

# Epipolar Constraint

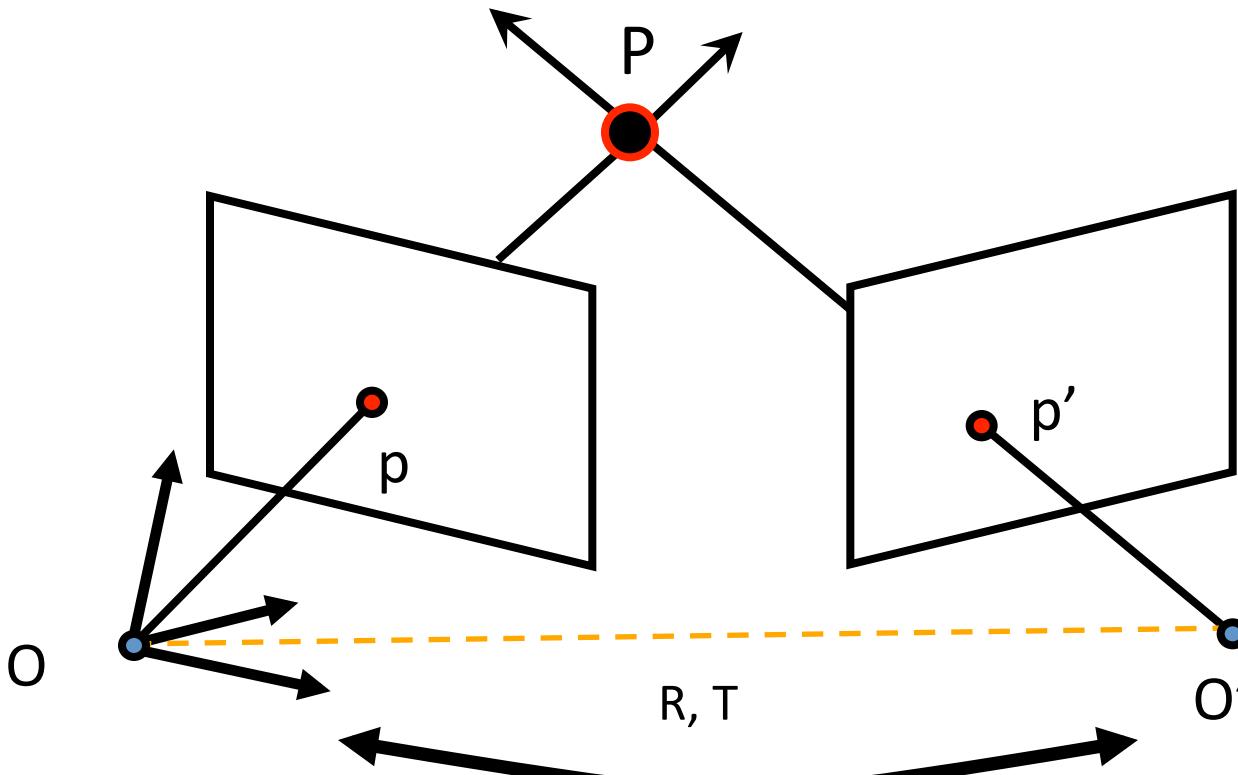


- Potential matches for  $p$  have to lie on the corresponding epipolar line  $l'$ .
- Potential matches for  $p'$  have to lie on the corresponding epipolar line  $l$ .

# Epipolar Constraint



# Epipolar Constraint



$$M = K \begin{bmatrix} I & 0 \end{bmatrix}$$

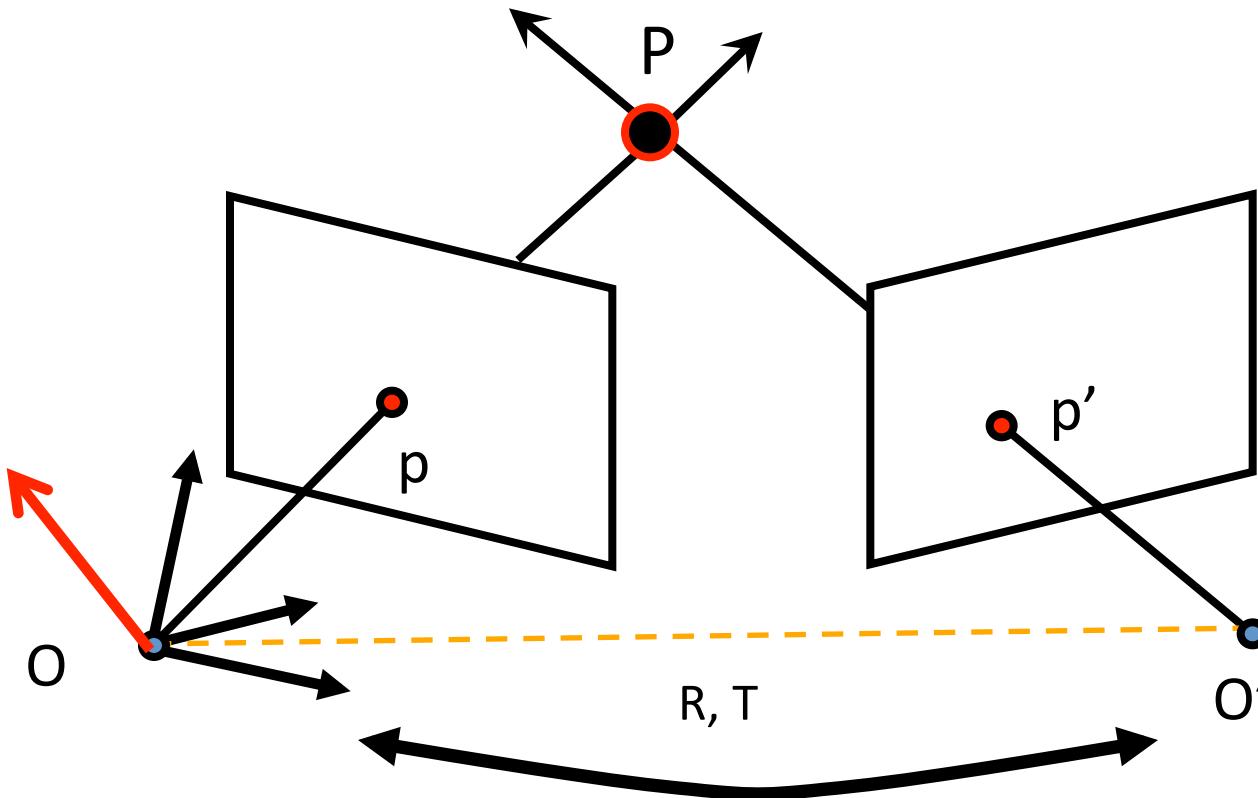
K and K' are known  
(calibrated cameras)

$$M = [I \quad 0]$$

$$M' = K' \begin{bmatrix} R & T \end{bmatrix}$$

$$M' = [R \quad T]$$

# Epipolar Constraint



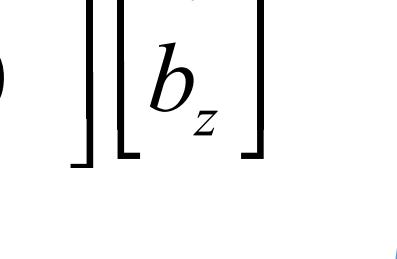
$$T \times (R p')$$

Perpendicular to epipolar plane

$$p^T \cdot [T \times (R p')] = 0$$

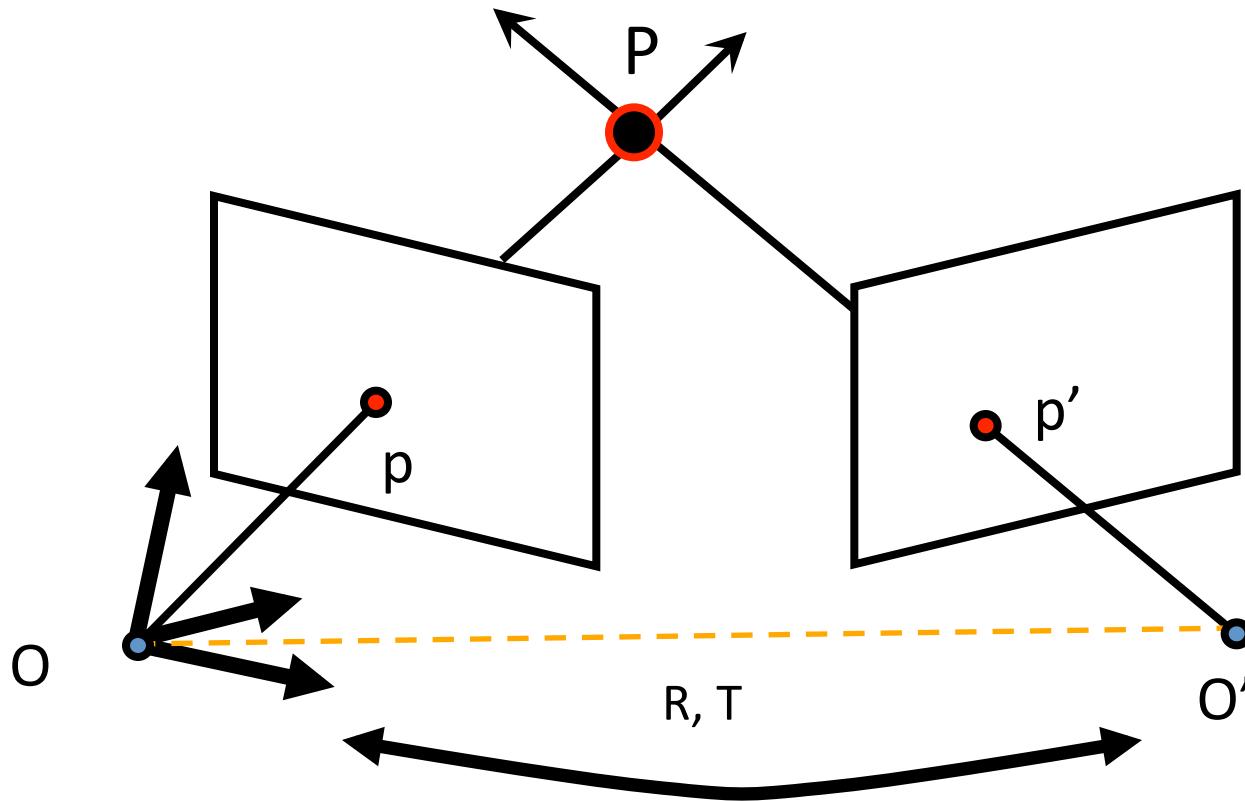
# Cross product as matrix multiplication

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = [\mathbf{a}_x] \mathbf{b}$$



“skew symmetric matrix”

# Epipolar Constraint

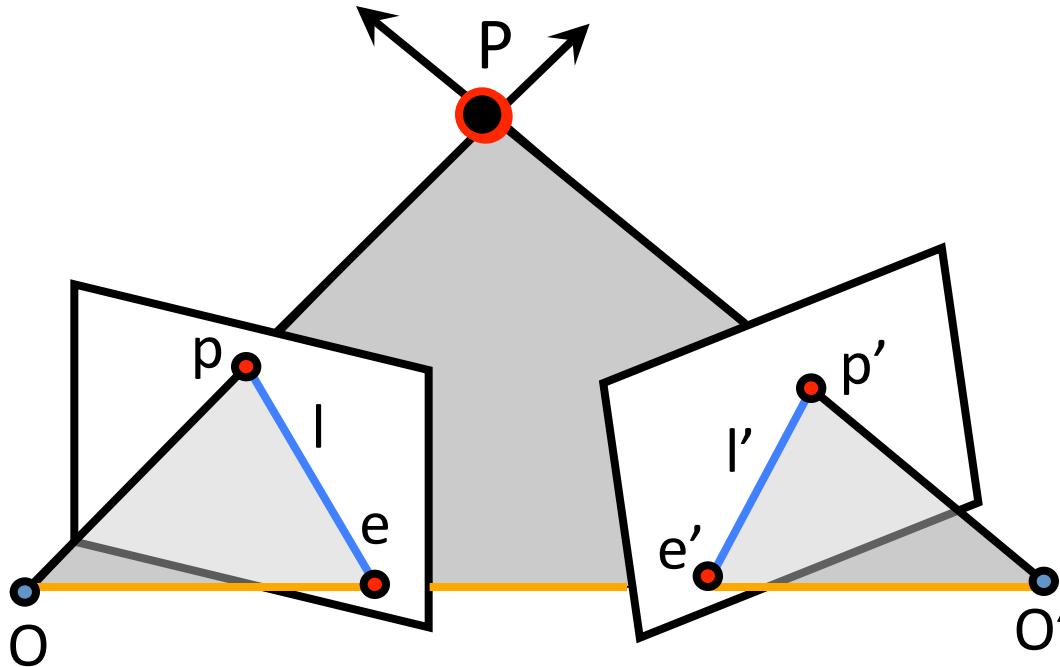


$$p^T \cdot [T \times (R p')] = 0 \rightarrow p^T \cdot [T_x] \cdot R p' = 0$$

(Longuet-Higgins, 1981)

$E$  = essential matrix

# Epipolar Constraint



- $E p'$  is the epipolar line associated with  $p'$  ( $l = E p'$ )
- $E^T p$  is the epipolar line associated with  $p$  ( $l' = E^T p$ )
- $E$  is singular (rank two)
- $E e' = 0$  and  $E^T e = 0$
- $E$  is  $3 \times 3$  matrix; 5 DOF

# What we will learn today?

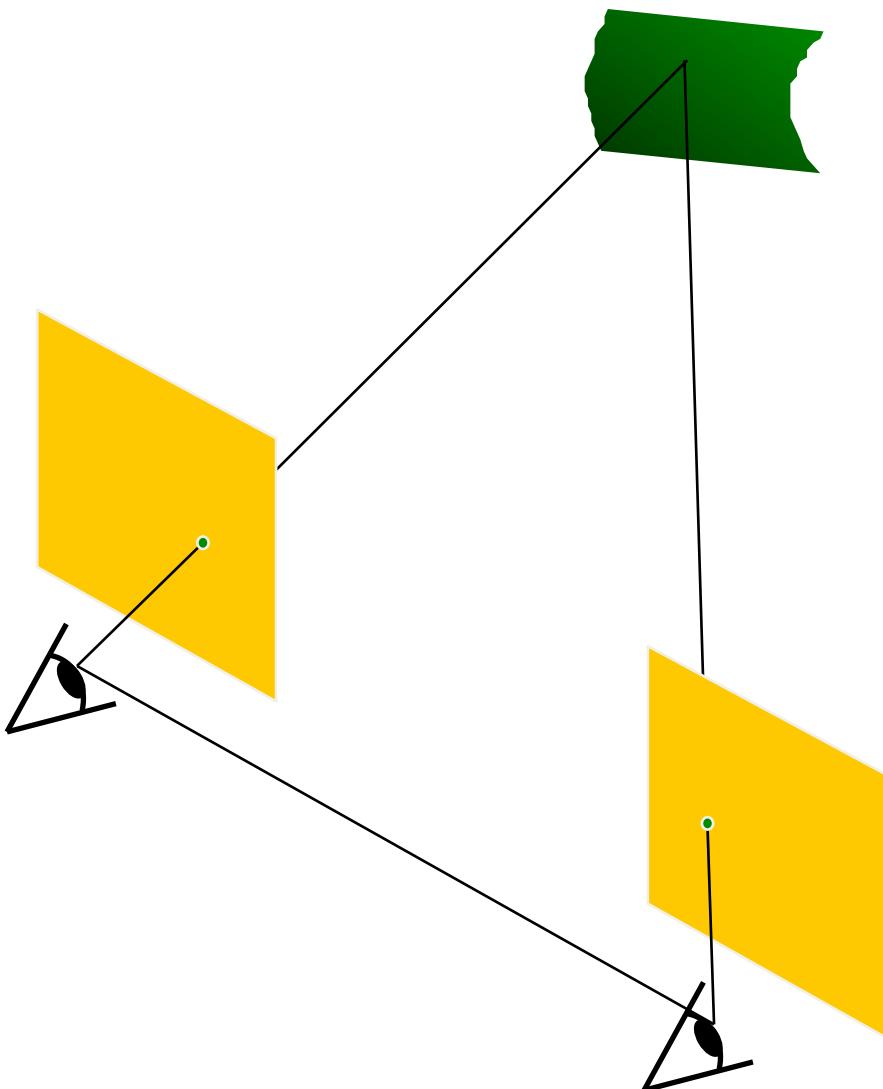
- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

[HZ] Chapters: 4, 9, 11

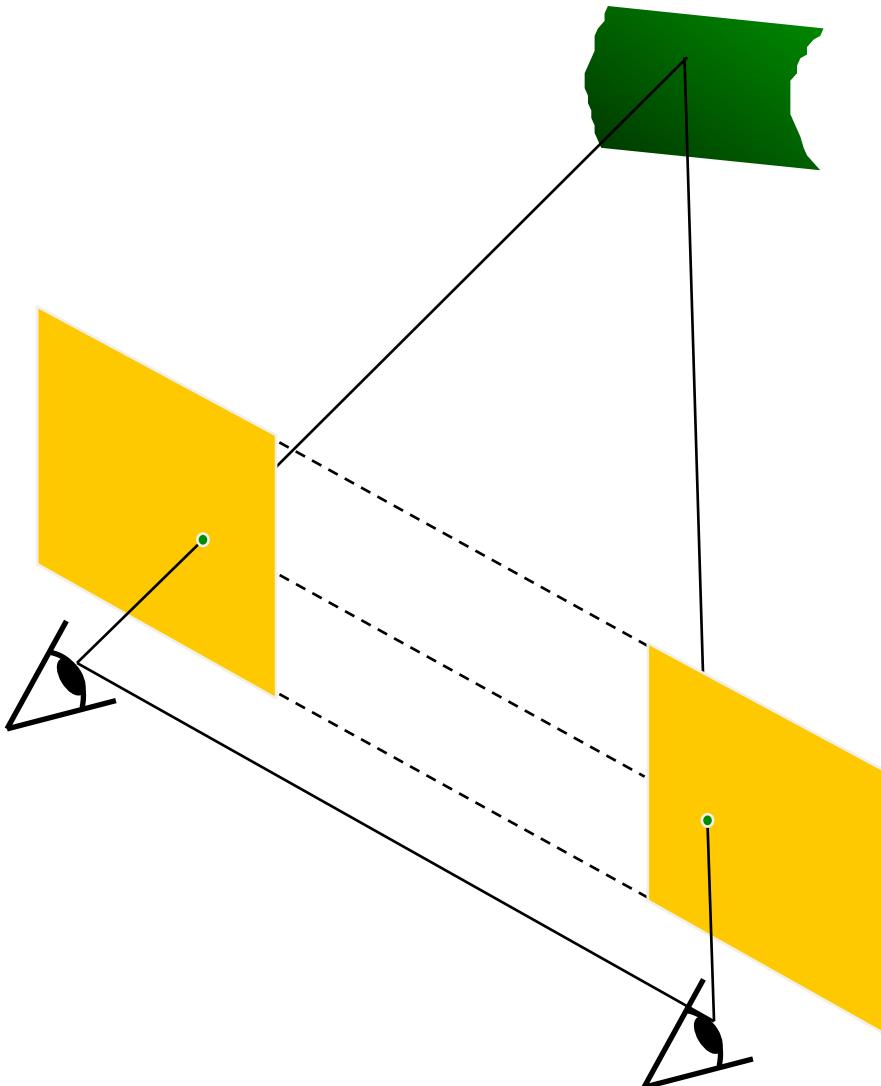
[FP] Chapters: 10

# Simplest Case: Parallel images



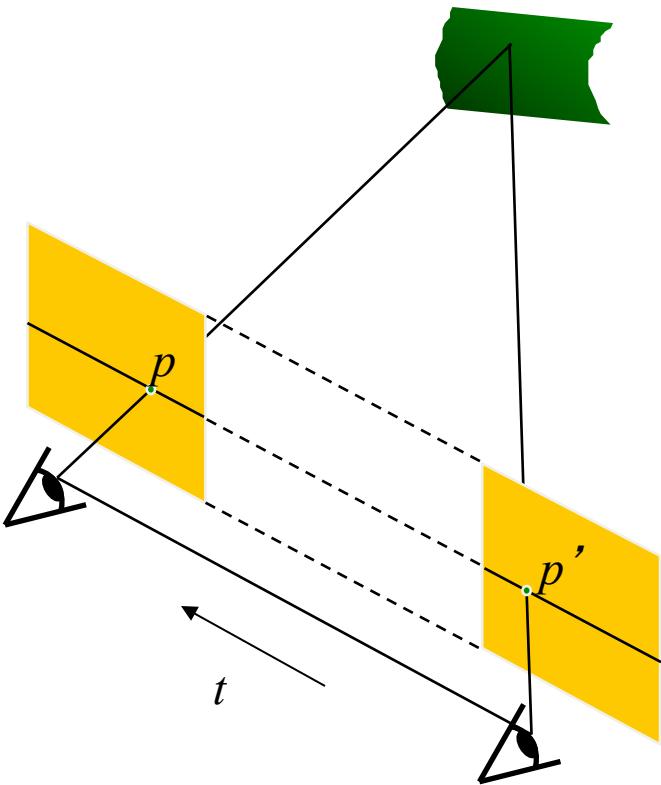
- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same

# Simplest Case: Parallel images



- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same
- Then, epipolar lines fall along the horizontal scan lines of the images

# Essential matrix for parallel images



Epipolar constraint:

$$R = I \quad t = (T, 0, 0)$$

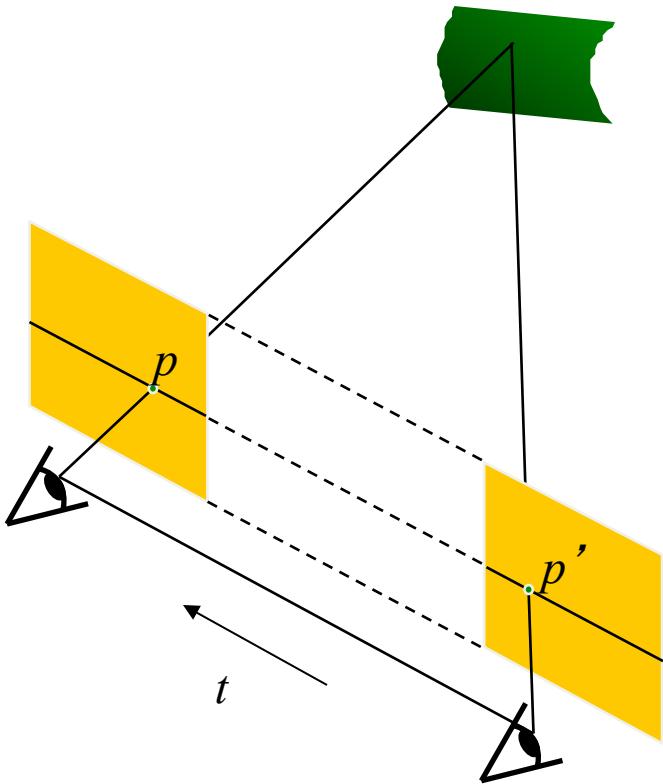
$$p^T E p' = 0, \quad E = [t_x]R$$

$$E = [t_x]R = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

Reminder: skew symmetric matrix

$$[a_x] = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}$$

# Essential matrix for parallel images



Epipolar constraint:

$$R = I \quad t = (T, 0, 0)$$

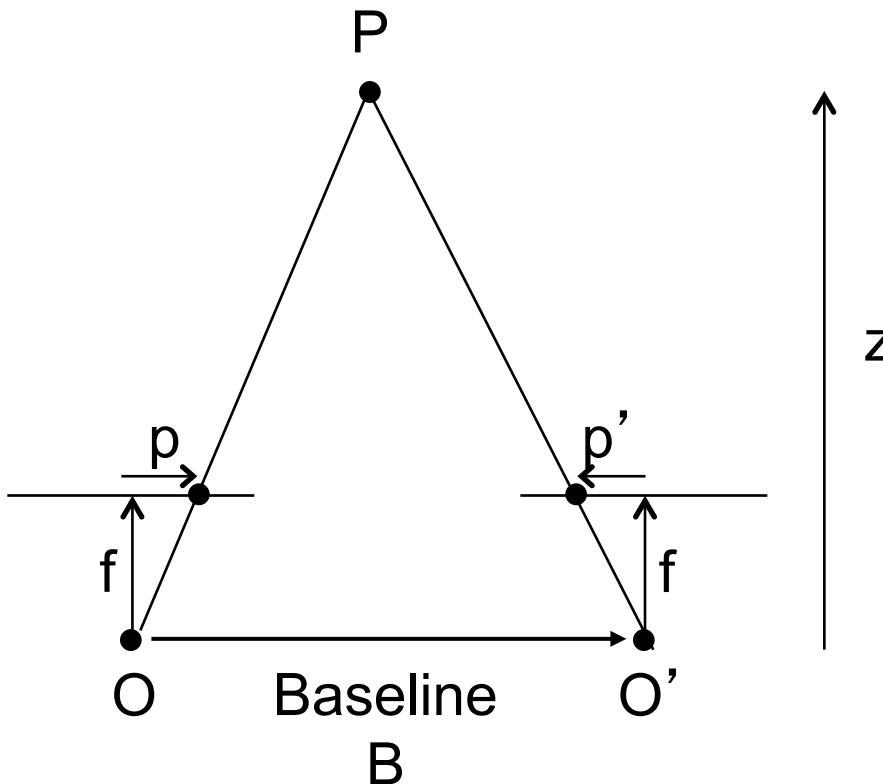
$$p^T E p' = 0, \quad E = [t_x]R$$

$$E = [t_x]R = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

$$(u \quad v \quad 1) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = 0 \quad (u \quad v \quad 1) \begin{pmatrix} 0 \\ -T \\ Tv' \end{pmatrix} = 0 \quad Tv = Tv'$$

The y-coordinates of corresponding points are the same!

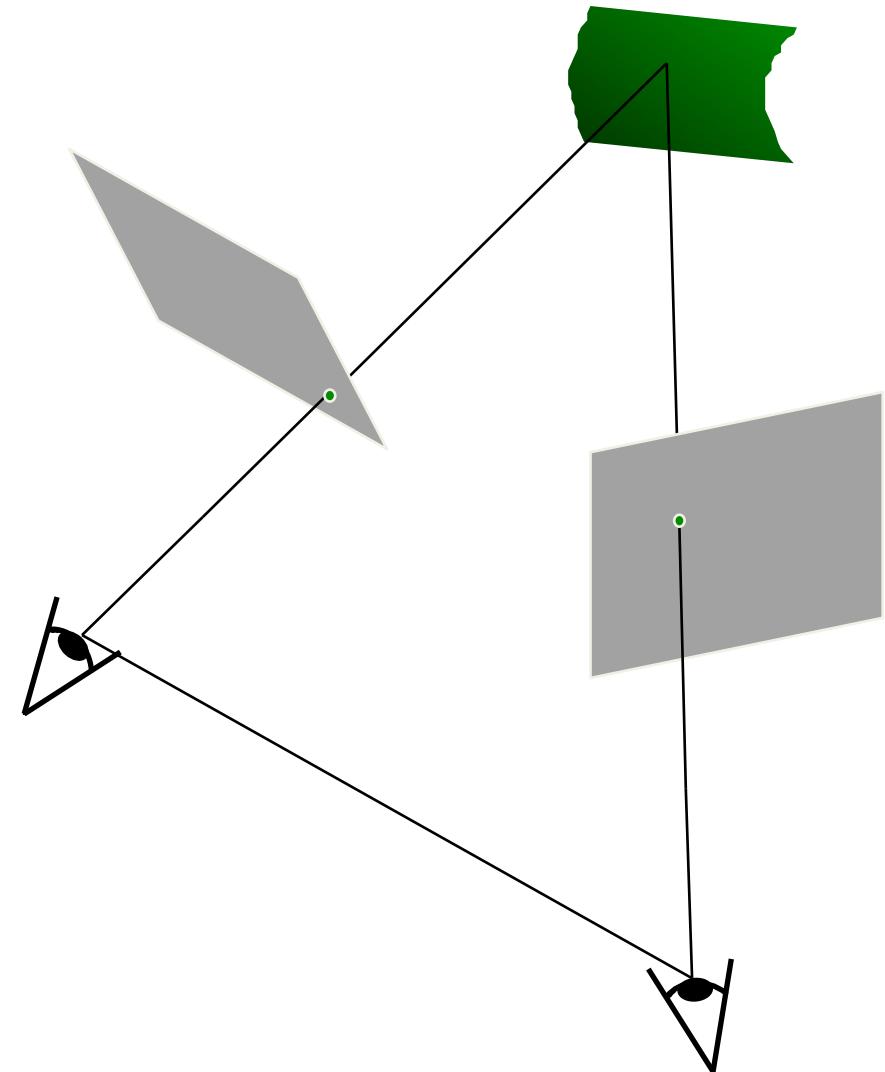
# Triangulation -- depth from disparity



$$\text{disparity} = u - u' = \frac{B \cdot f}{z}$$

Disparity is inversely proportional to depth!

# Stereo image rectification

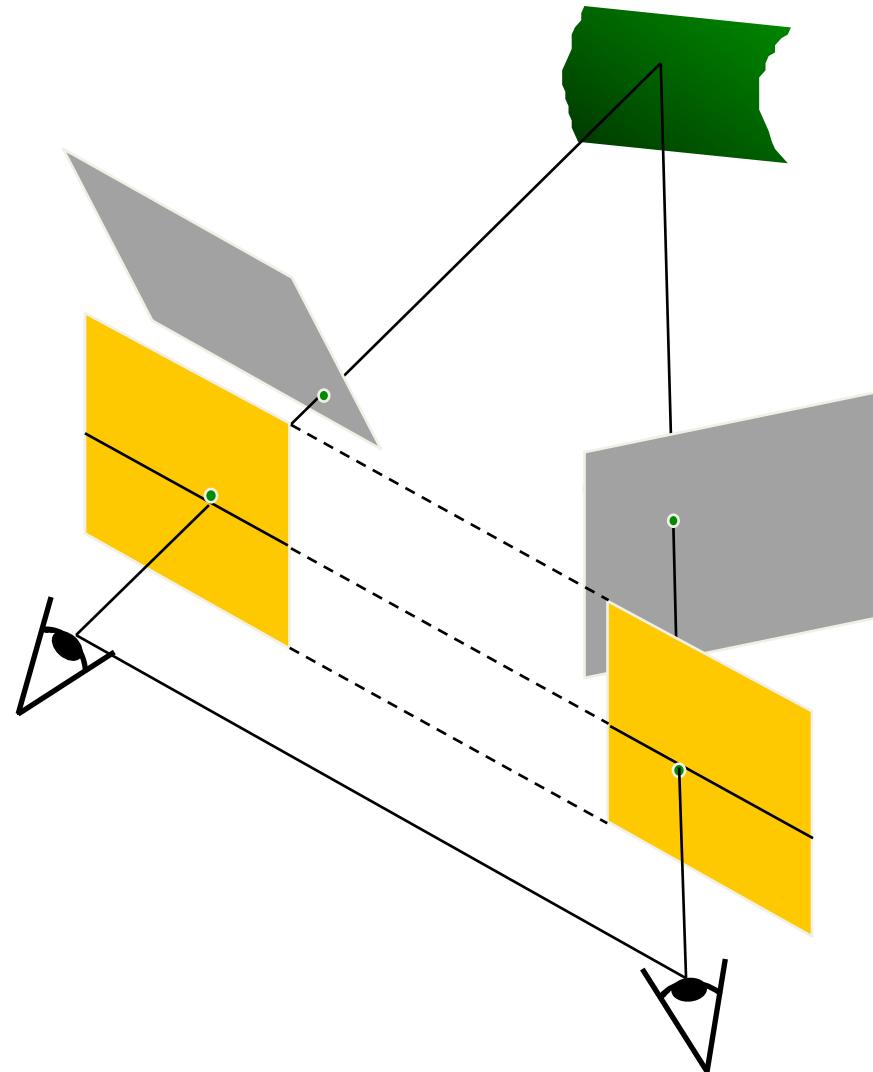


Slide credit: J. Hayes

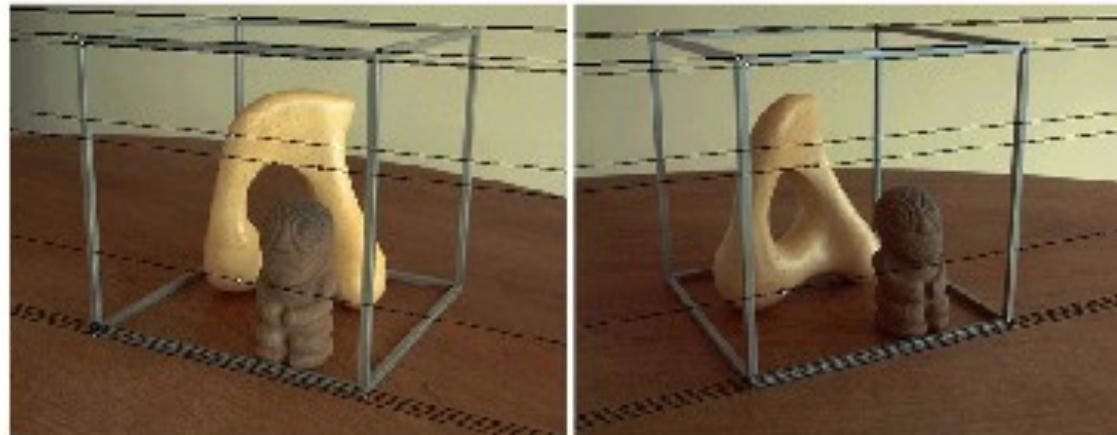
# Stereo image rectification

Algorithm:

- Re-project image planes onto a common plane parallel to the line between optical centers
  - Pixel motion is horizontal after this transformation
  - Two transformation matrices, one for each input image reprojection
- C. Loop and Z. Zhang,  
[Computing Rectifying Homographies for Stereo Vision](#). IEEE Conf. Computer Vision and Pattern Recognition, 1999.

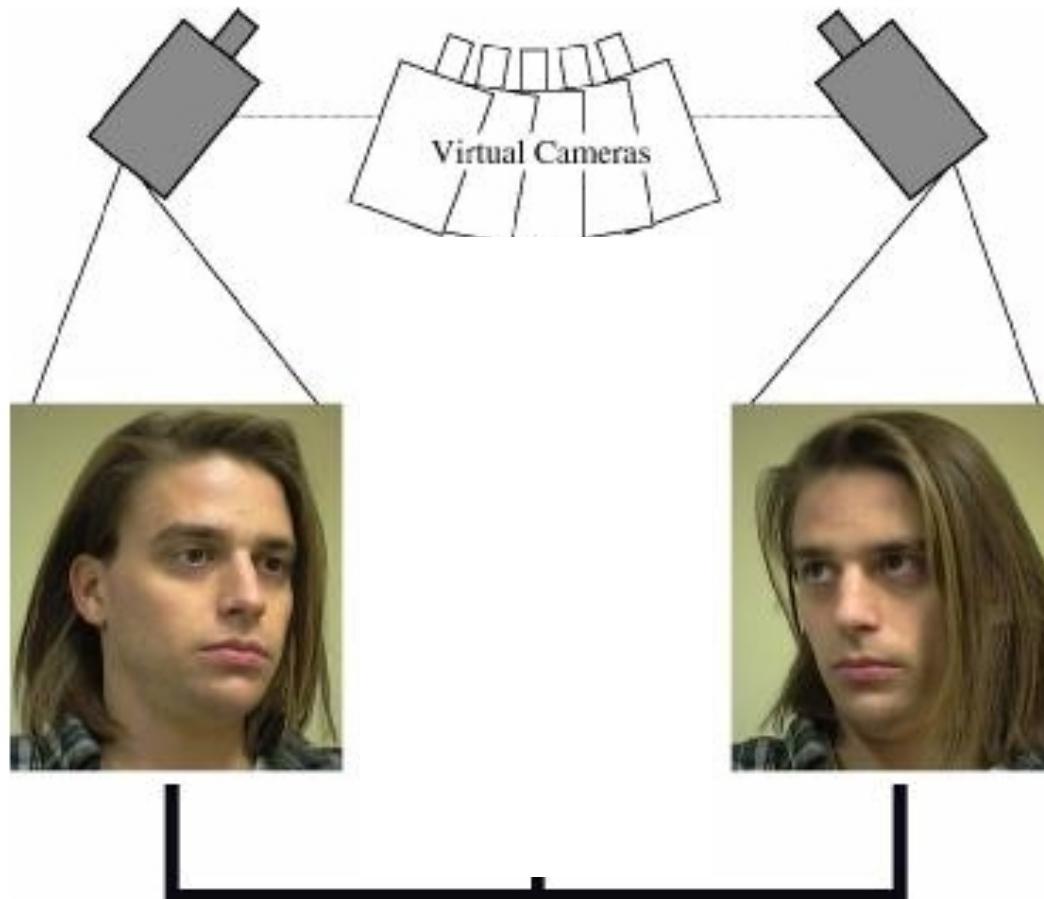


# Rectification example

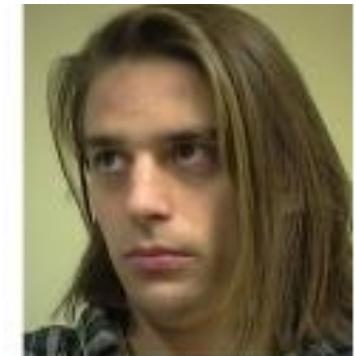


# Application: view morphing

S. M. Seitz and C. R. Dyer, *Proc. SIGGRAPH 96*, 1996, 21-30



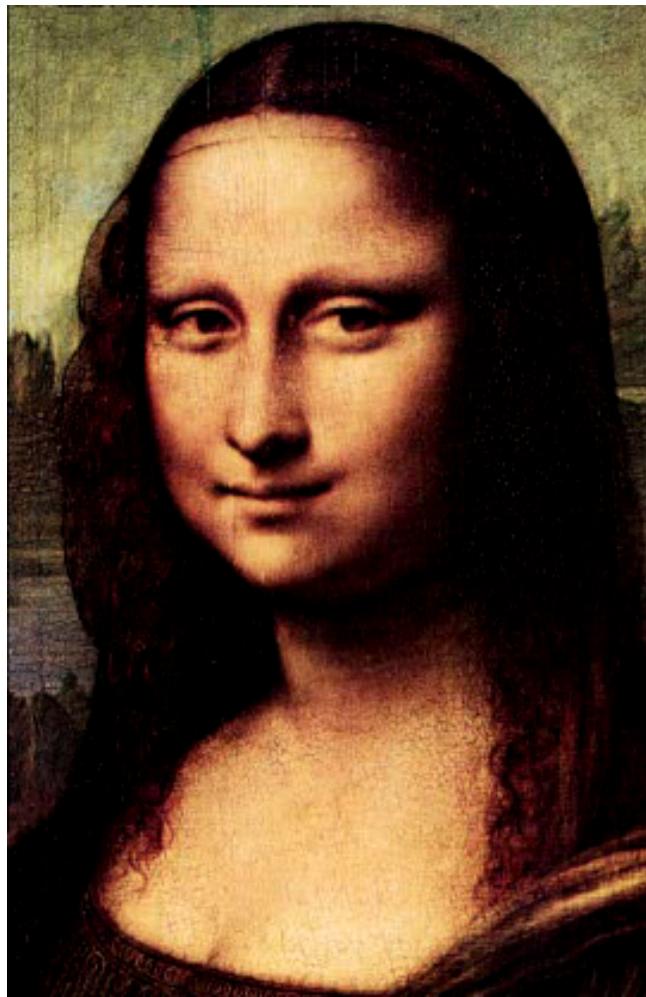
# Application: view morphing



# Application: view morphing

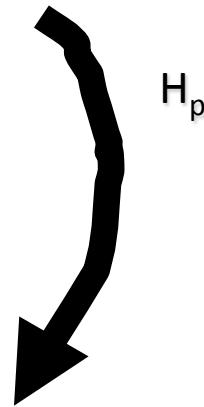


# Application: view morphing



# Removing perspective distortion

(rectification)



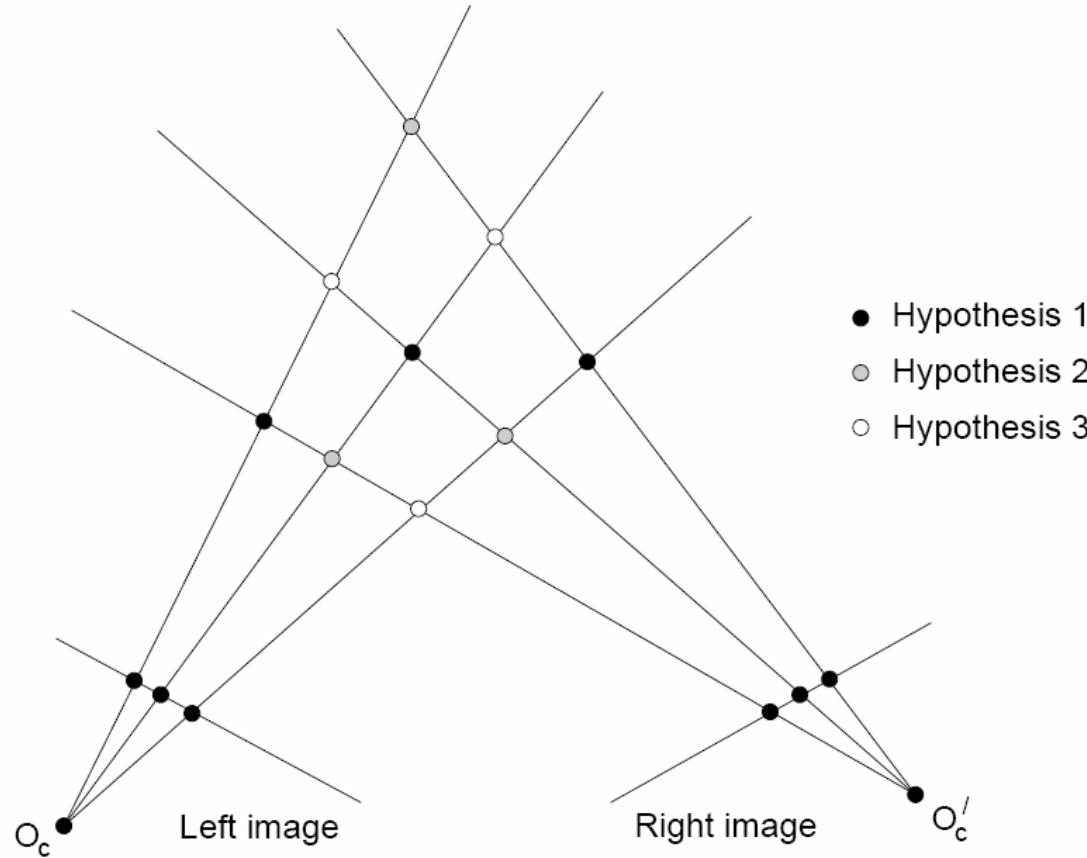
# What we will learn today?

- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

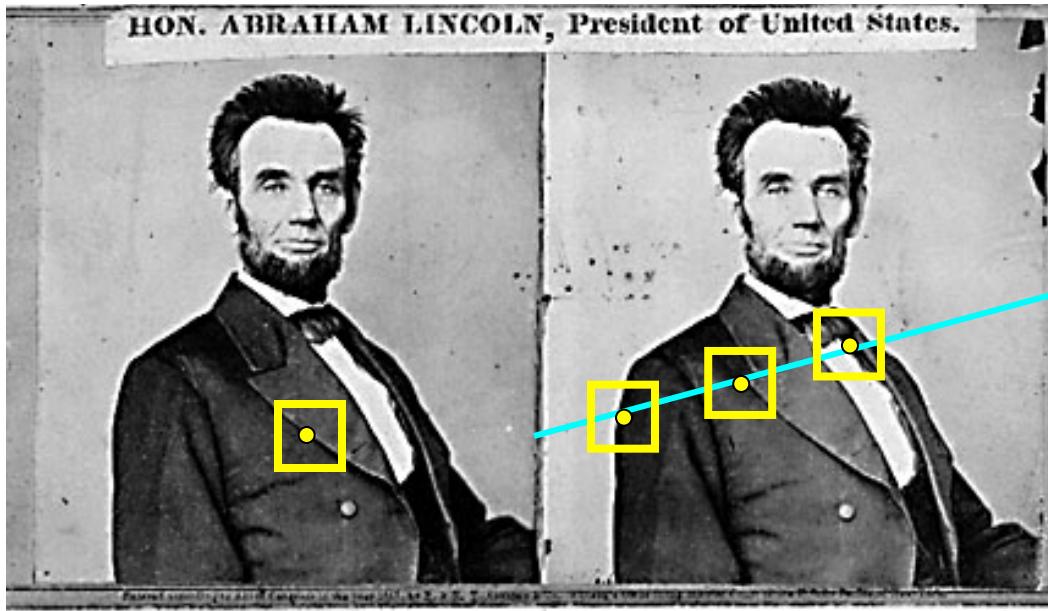
[HZ] Chapters: 4, 9, 11  
[FP] Chapters: 10

# Stereo matching: solving the correspondence problem



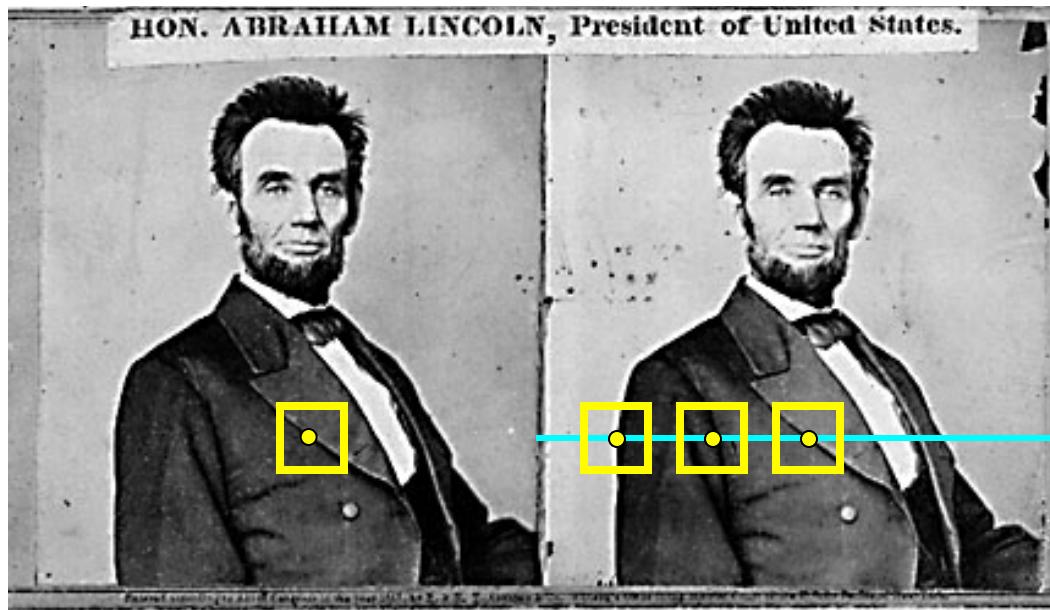
- Goal: finding matching points between two images

# Basic stereo matching algorithm



- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match
  - Triangulate the matches to get depth information
- Simplest case: epipolar lines are scanlines
  - When does this happen?

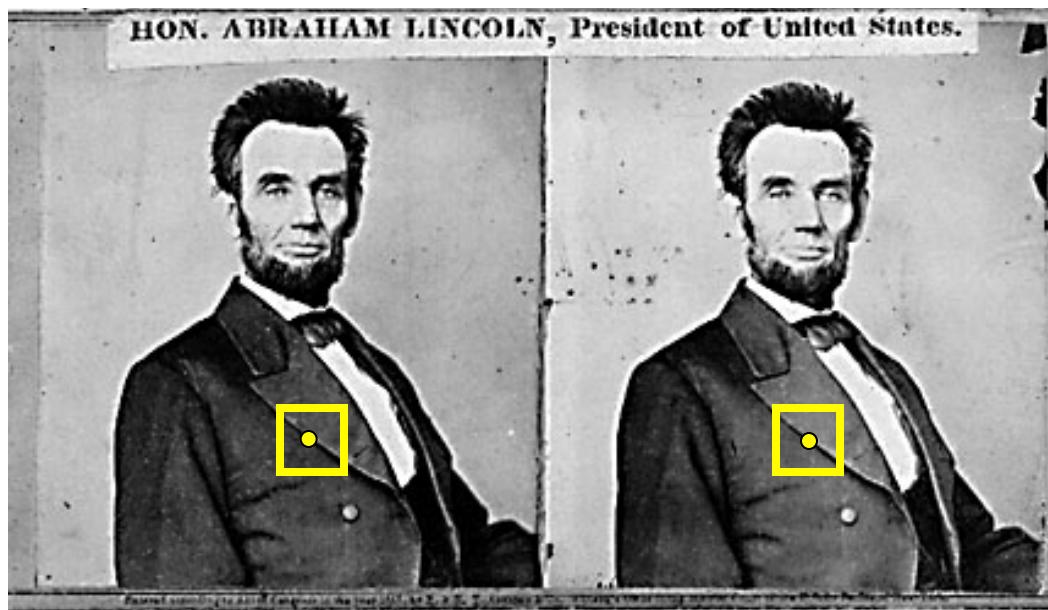
# Basic stereo matching algorithm



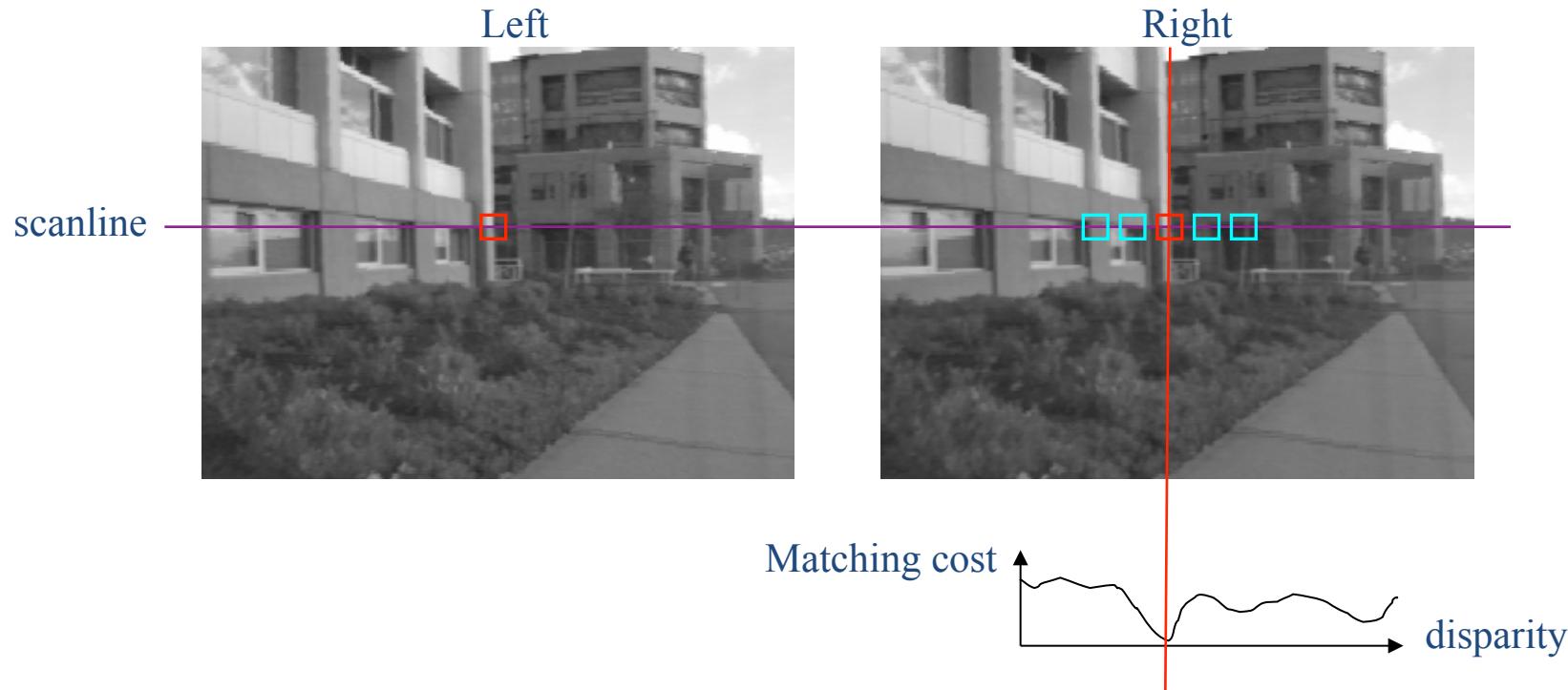
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel  $x$  in the first image
  - Find **corresponding** epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match  $x'$
  - Compute disparity  $x-x'$  and set  $\text{depth}(x) = 1/(x-x')$

# Correspondence problem

- Let's make some assumptions to simplify the matching problem
  - The baseline is relatively small (compared to the depth of scene points)
  - Then most scene points are visible in both views
  - Also, matching regions are similar in appearance

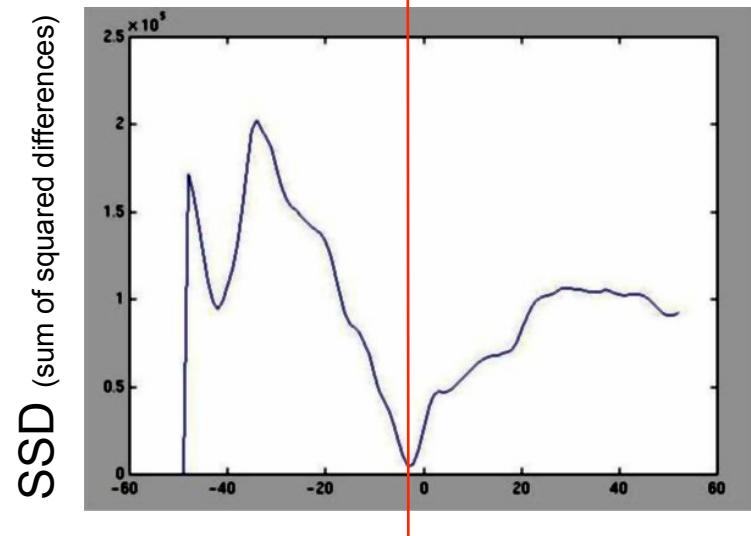
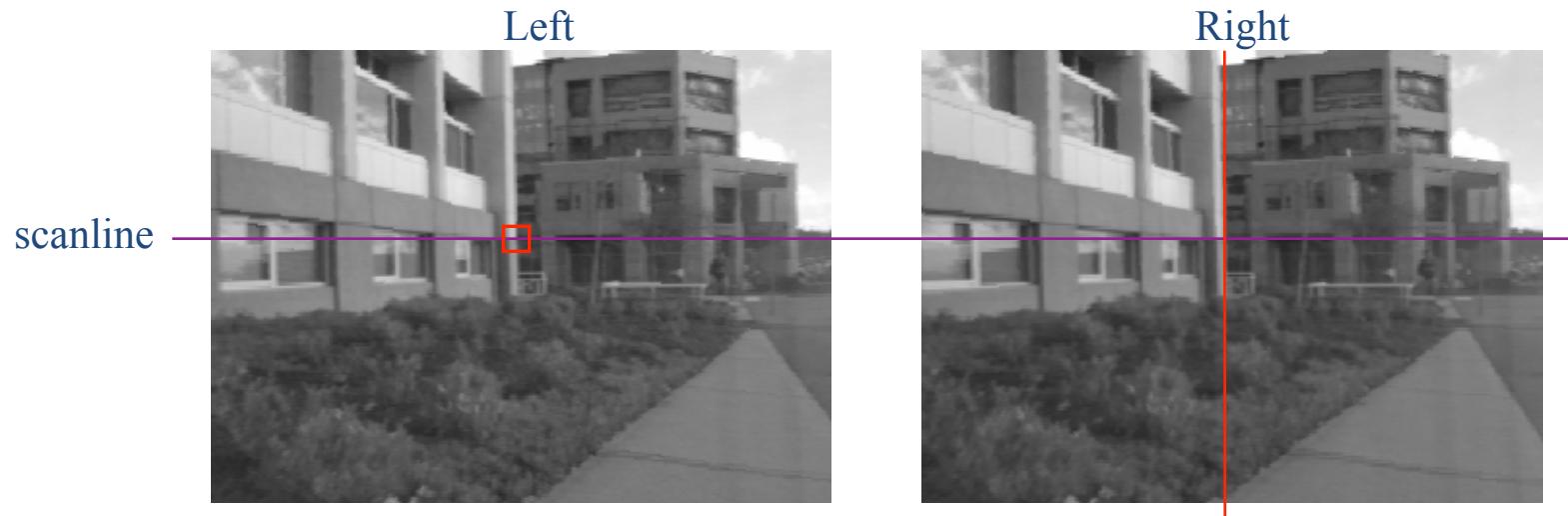


# Correspondence search with similarity constraint

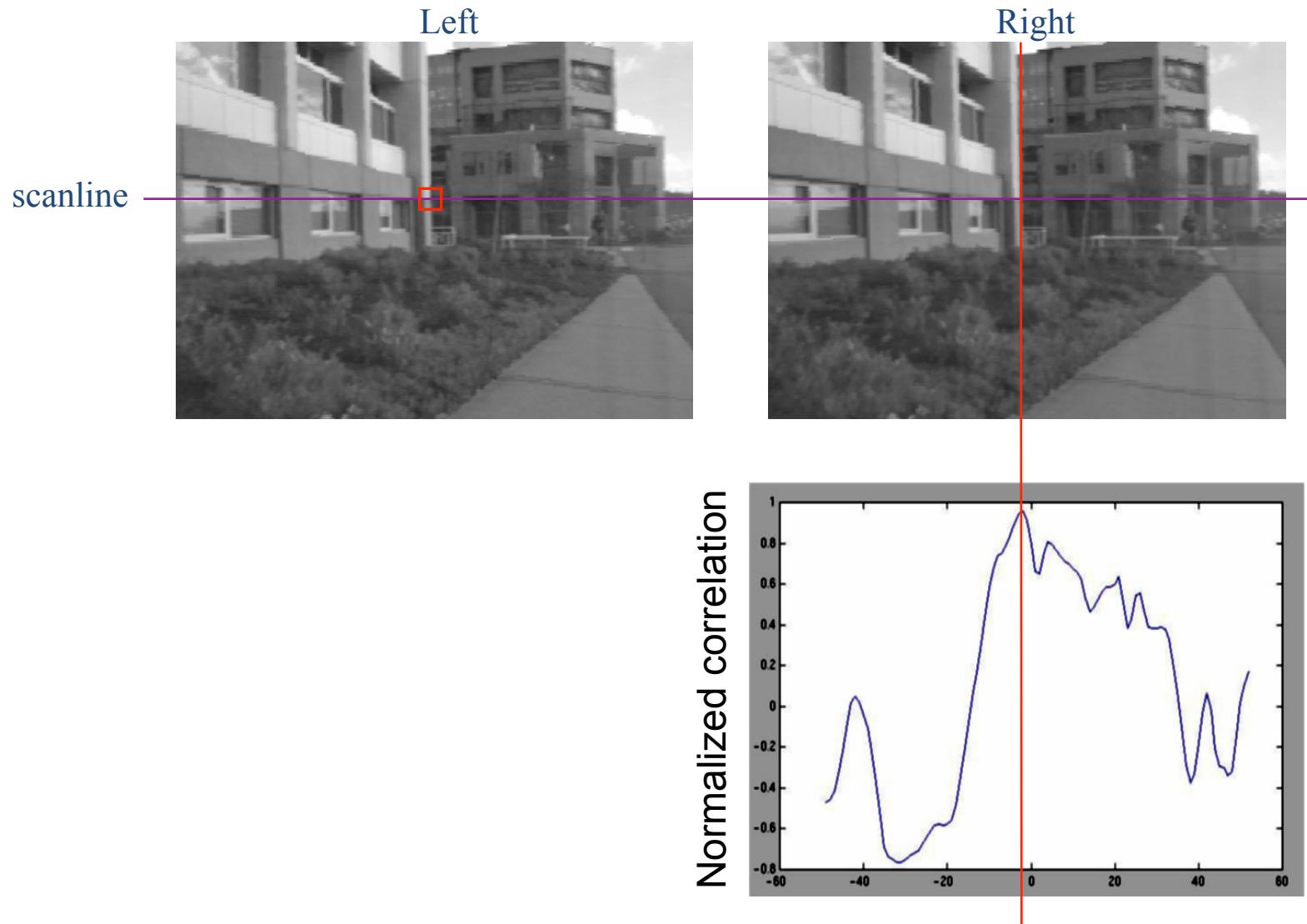


- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

# Correspondence search with similarity constraint



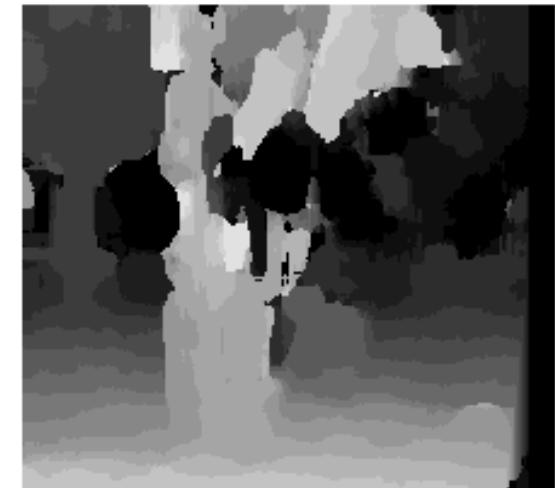
# Correspondence search with similarity constraint



# Effect of window size



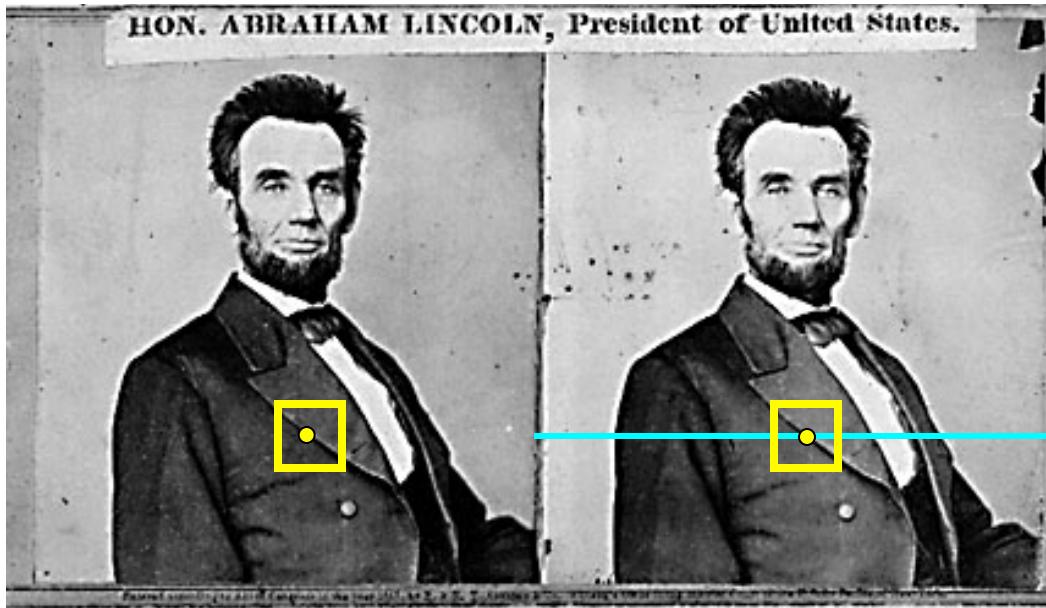
$$W = 3$$



$$W = 20$$

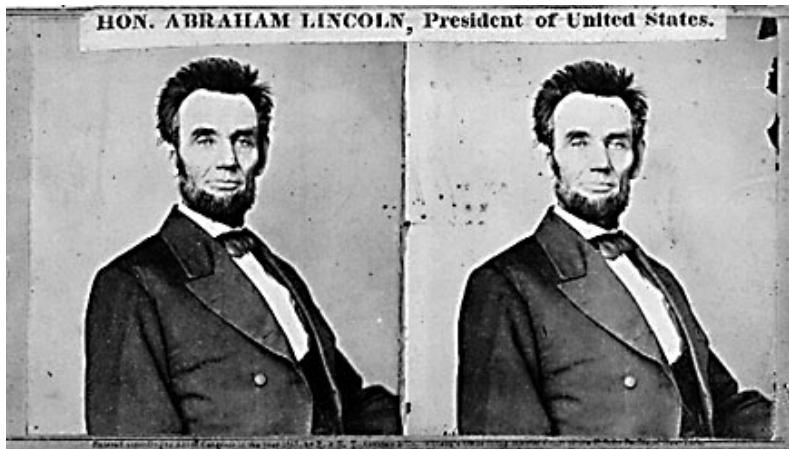
- Smaller window
  - + More detail
  - More noise
  
- Larger window
  - + Smoother disparity maps
  - Less detail

# The similarity constraint



- Corresponding regions in two images should be similar in appearance
- ...and non-corresponding regions should be different
- When will the similarity constraint fail?

# Limitations of similarity constraint



Textureless surfaces



Occlusions, repetition



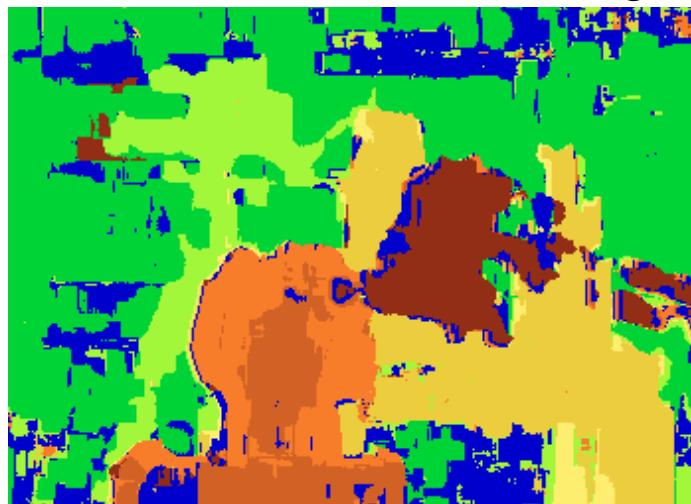
Specular surfaces



# Results with window search



Window-based matching



Ground truth



# Better methods exist... (CS231a)



Graph cuts

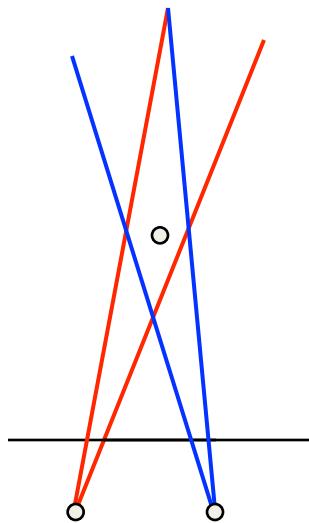


Ground truth

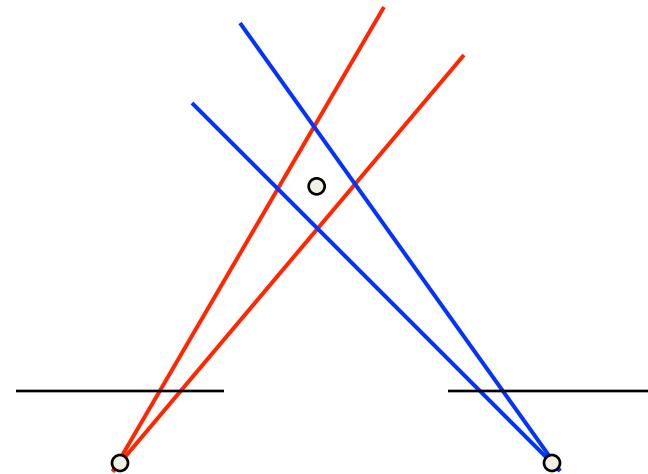
Y. Boykov, O. Veksler, and R. Zabih,

Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

# The role of the baseline



**Small Baseline**

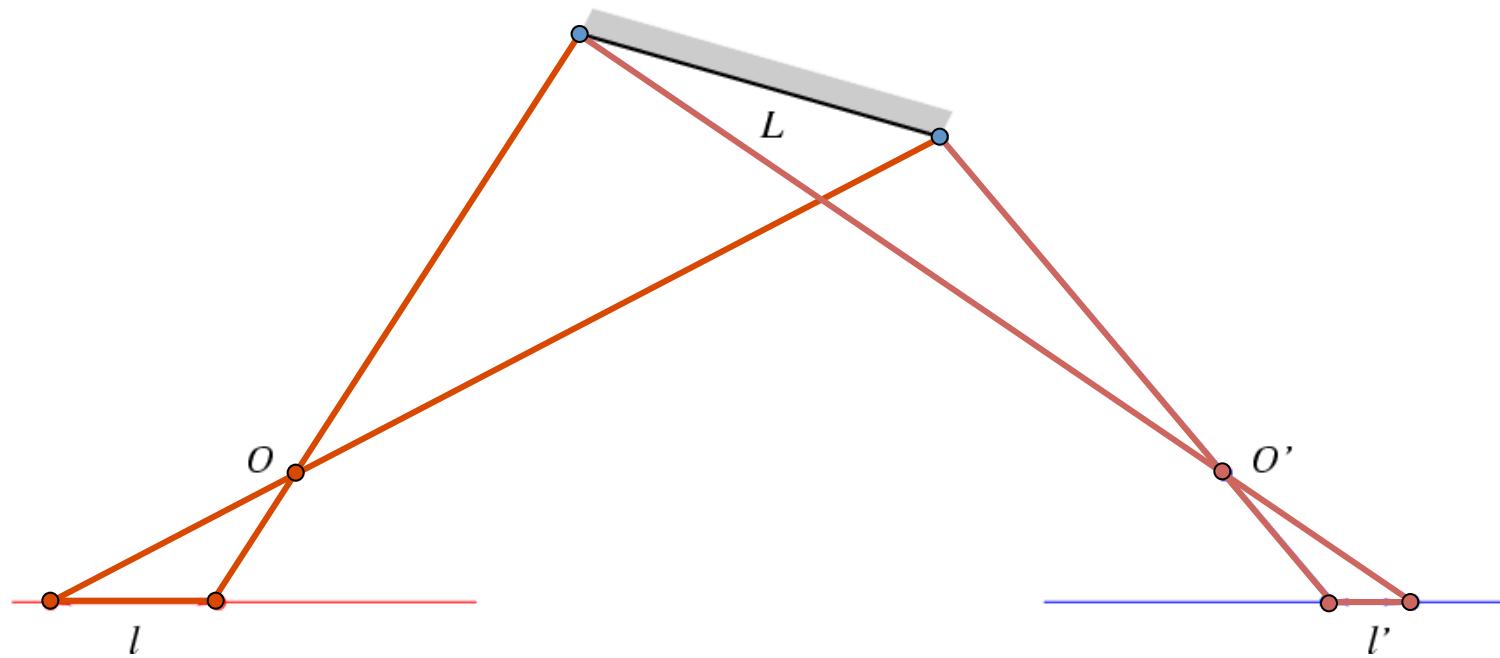


**Large Baseline**

- Small baseline: large depth error
- Large baseline: difficult search problem

Slide credit: S. Seitz

# Problem for wide baselines: Foreshortening



- Matching with fixed-size windows will fail!
- Possible solution: adaptively vary window size
- Another solution: *model-based stereo* (CS231a)

Slide credit: J. Hayes

# What we will learn today?

- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

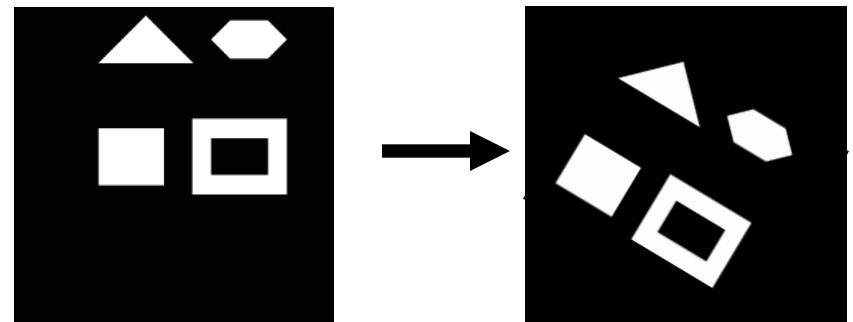
[HZ] Chapters: 4, 9, 11  
[FP] Chapters: 10

# Reminder: transformations in 2D

Special case  
from lecture 2  
(planar rotation  
& translation)

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H_e \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

- 3 DOF
- Preserve distance (areas)
- Regulate motion  
of rigid object

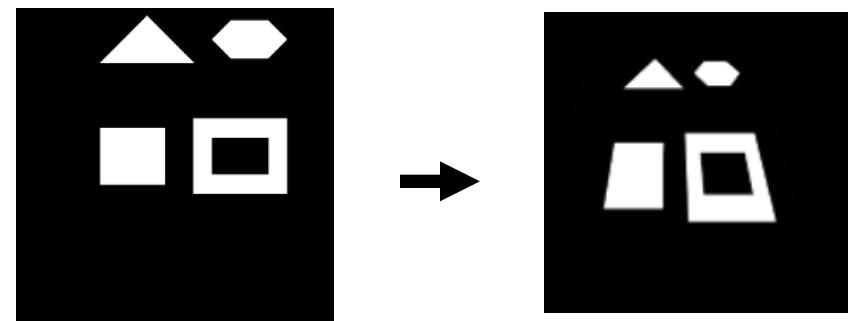


# Reminder: transformations in 2D

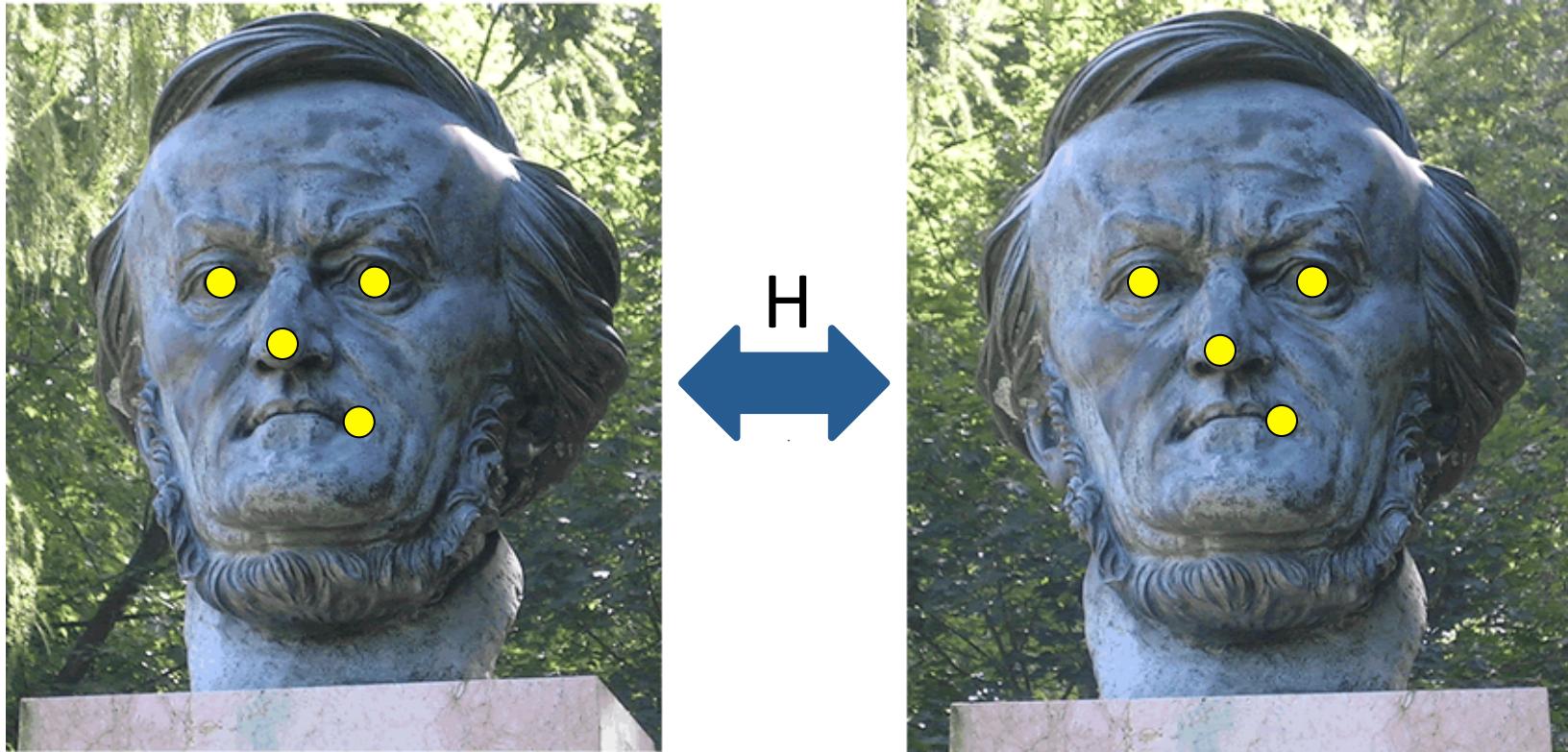
**Generic case**  
(rotation in 3D, scale  
& translation)

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

- 8 DOF
- Preserve colinearity

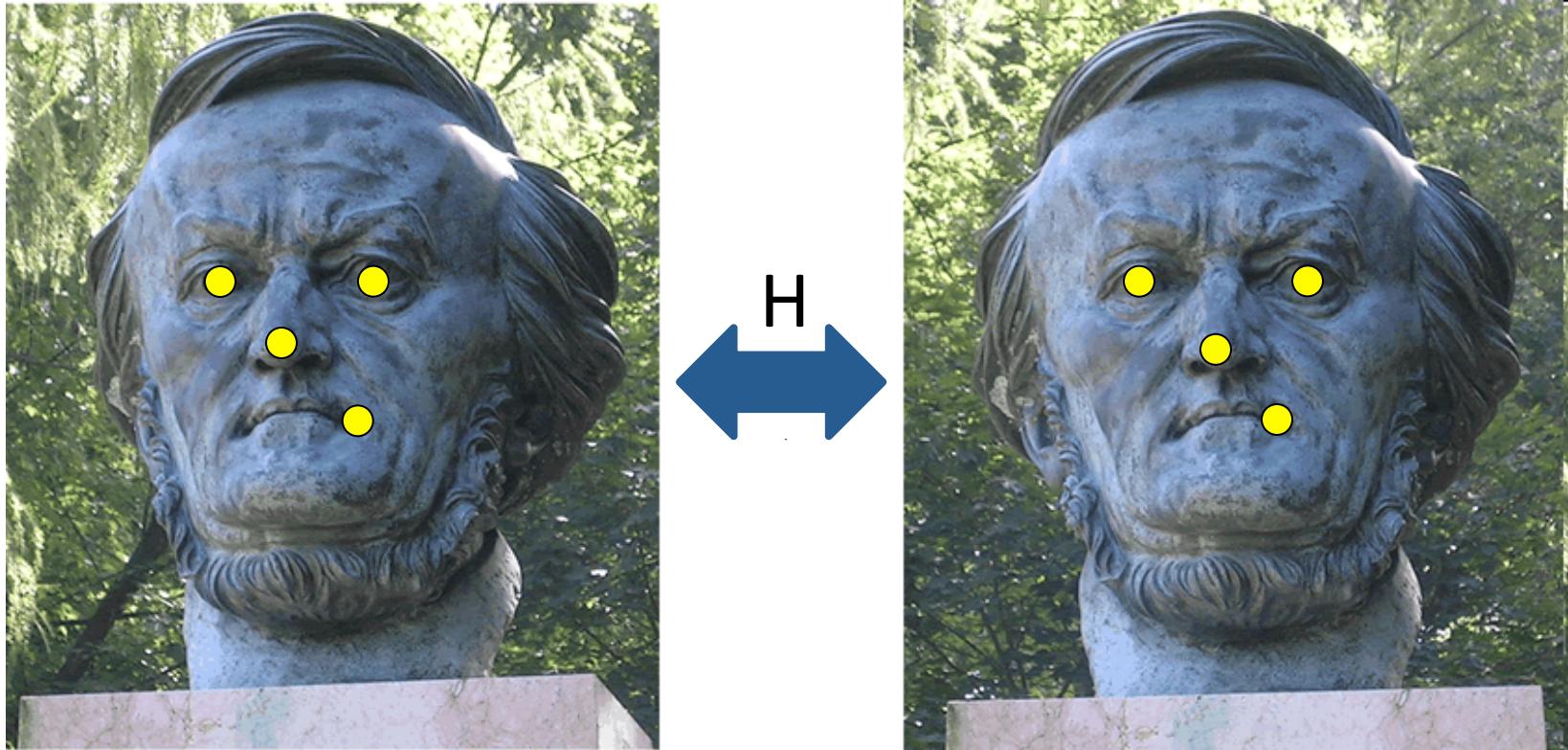


# Goal: estimate the homographic transformation between two images



Assumption: Given a set of corresponding points.

# Goal: estimate the homographic transformation between two images



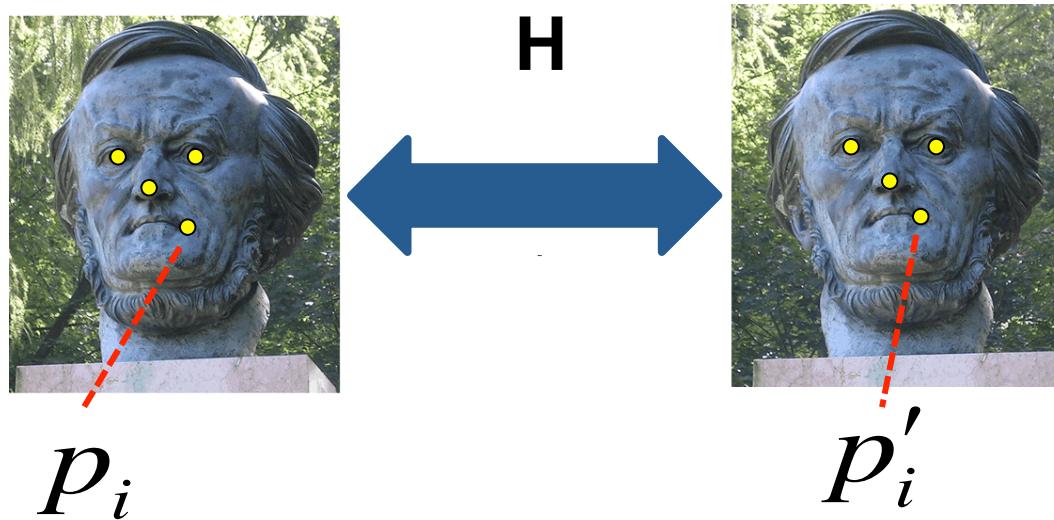
Assumption: Given a set of corresponding points.

Question: How many points are needed?

Hint: DoF for  $H$ ? **8!**

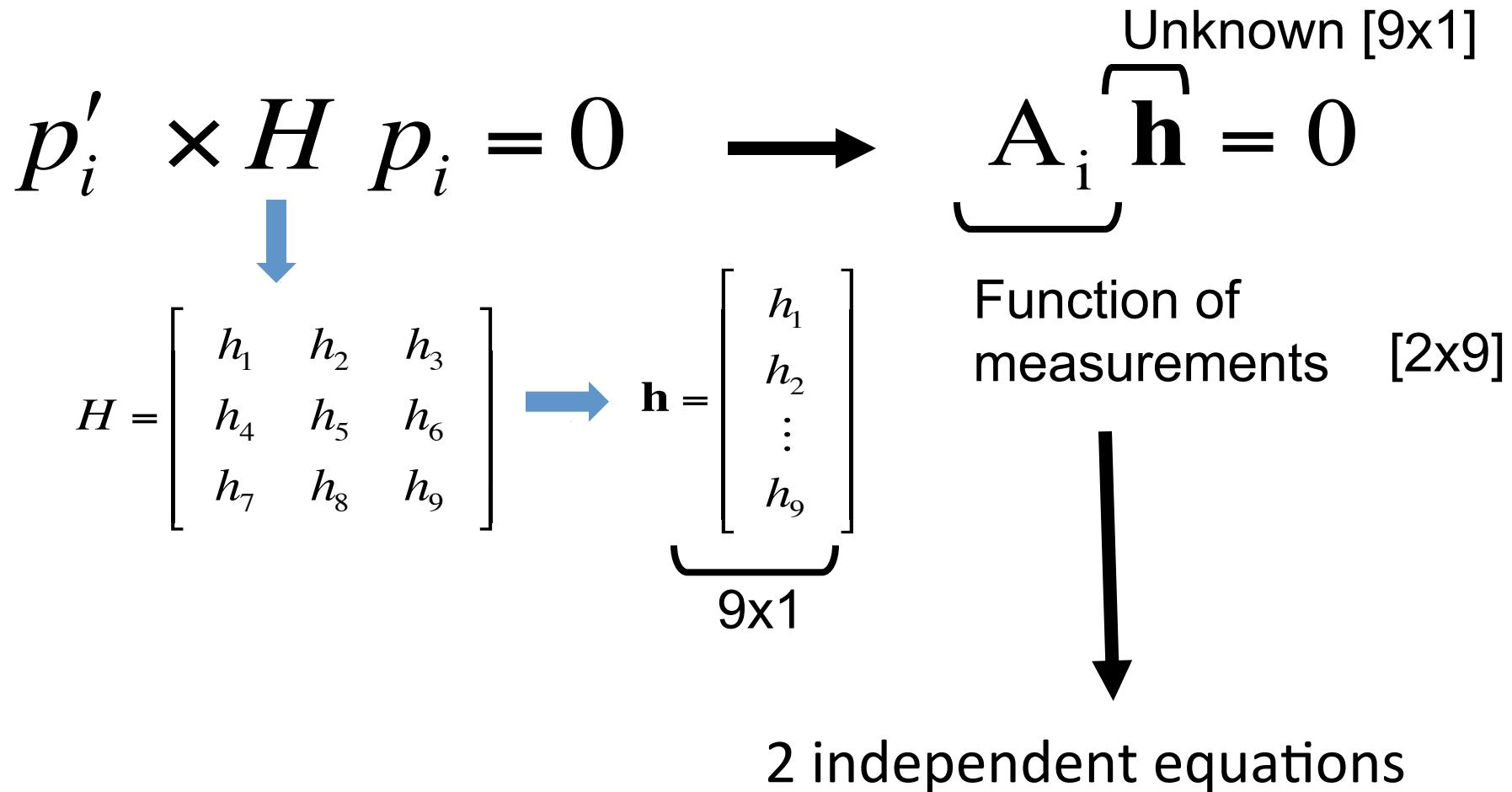
At least 4 points  
(8 equations)

# DLT algorithm (Direct Linear Transformation)



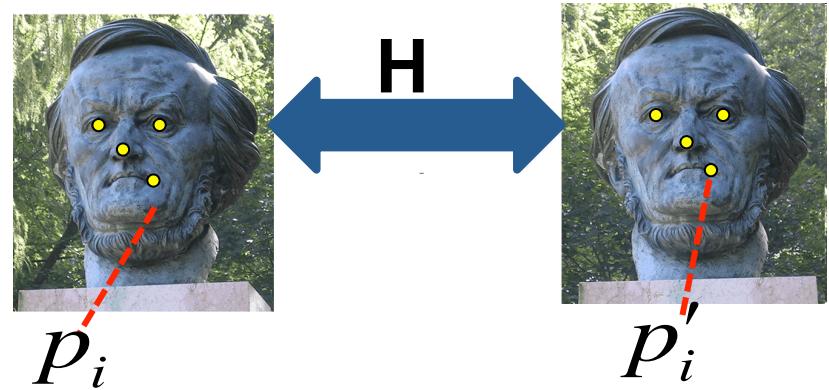
$$p'_i = H p_i$$

# DLT algorithm (direct Linear Transformation)



# DLT algorithm (direct Linear Transformation)

$$A_{2 \times 9} \quad h_{9 \times 1}$$
$$\boxed{A_i} \boxed{h} = 0$$



$$\left\{ \begin{array}{l} A_1 h = 0 \\ A_2 h = 0 \\ \vdots \\ A_N h = 0 \end{array} \right. \rightarrow A_{2N \times 9} h_{9 \times 1} = 0$$

Over determined  
Homogenous system

# DLT algorithm (direct Linear Transformation)

How to solve  $A_{2N \times 9} h_{9 \times 1} = 0$  ?

Singular Value Decomposition (SVD)!

# DLT algorithm (direct Linear Transformation)

How to solve  $A_{2N \times 9} h_{9 \times 1} = 0$  ?

Singular Value Decomposition (SVD)!



$$U_{2n \times 9} \Sigma_{9 \times 9} V^T_{9 \times 9}$$

Last column of V gives h!  $\rightarrow H!$

Why? See pag 593 of AZ

# DLT algorithm (direct Linear Transformation)

How to solve  $A_{2N \times 9} h_{9 \times 1} = 0$  ?

```
[U,D,V] = svd(A,0);  
x = V(:,end);
```

# Clarification about SVD

$$P_{m \times n} = \boxed{U}_{m \times n} D_{n \times n} \boxed{V}^T_{n \times n}$$

Has n orthogonal columns

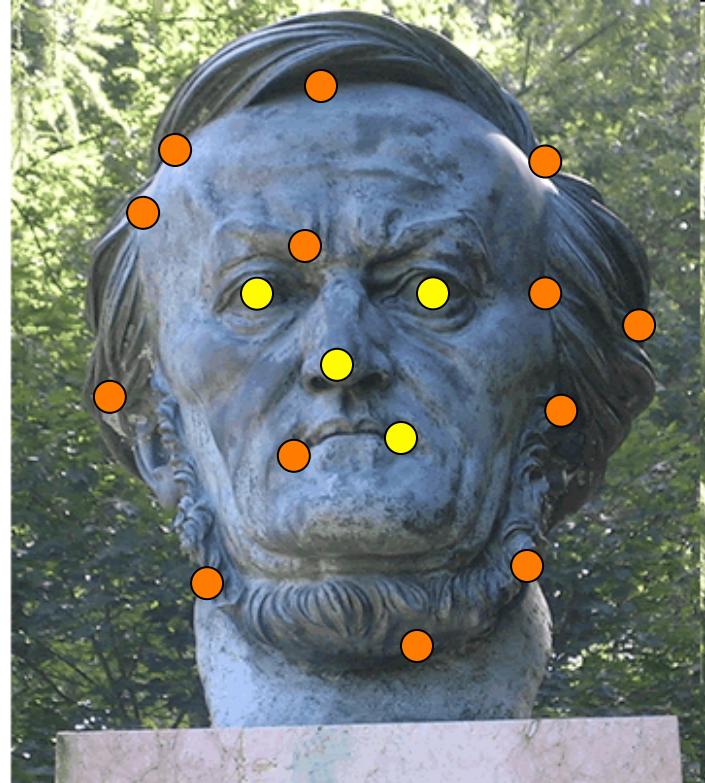
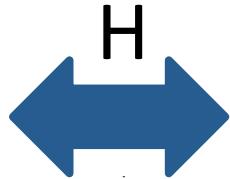
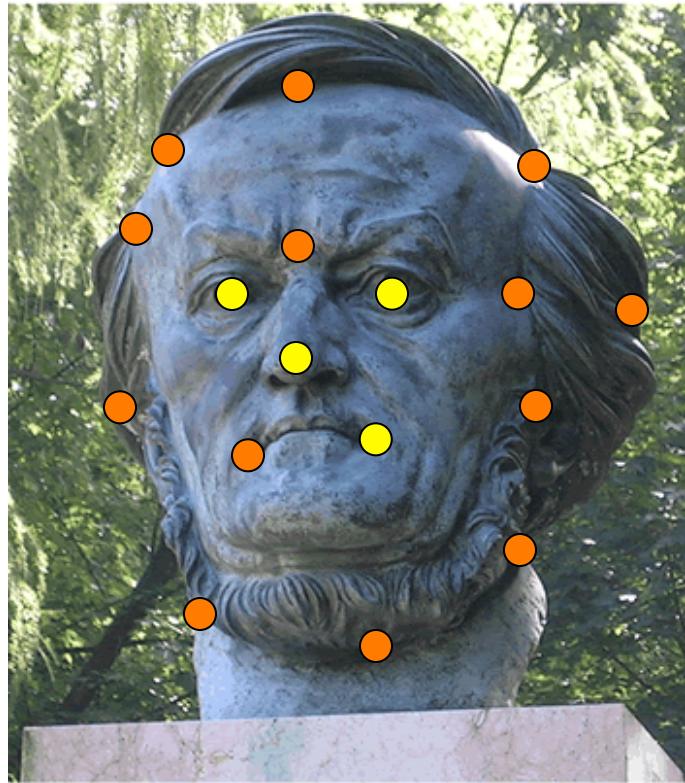
Orthogonal matrix

- This is one of the possible SVD decompositions
- This is typically used for efficiency
- The classic SVD is actually:

$$P_{m \times n} = \boxed{U}_{m \times m} D_{m \times n} \boxed{V}^T_{n \times n}$$

orthogonal

Orthogonal



# What we will learn today?

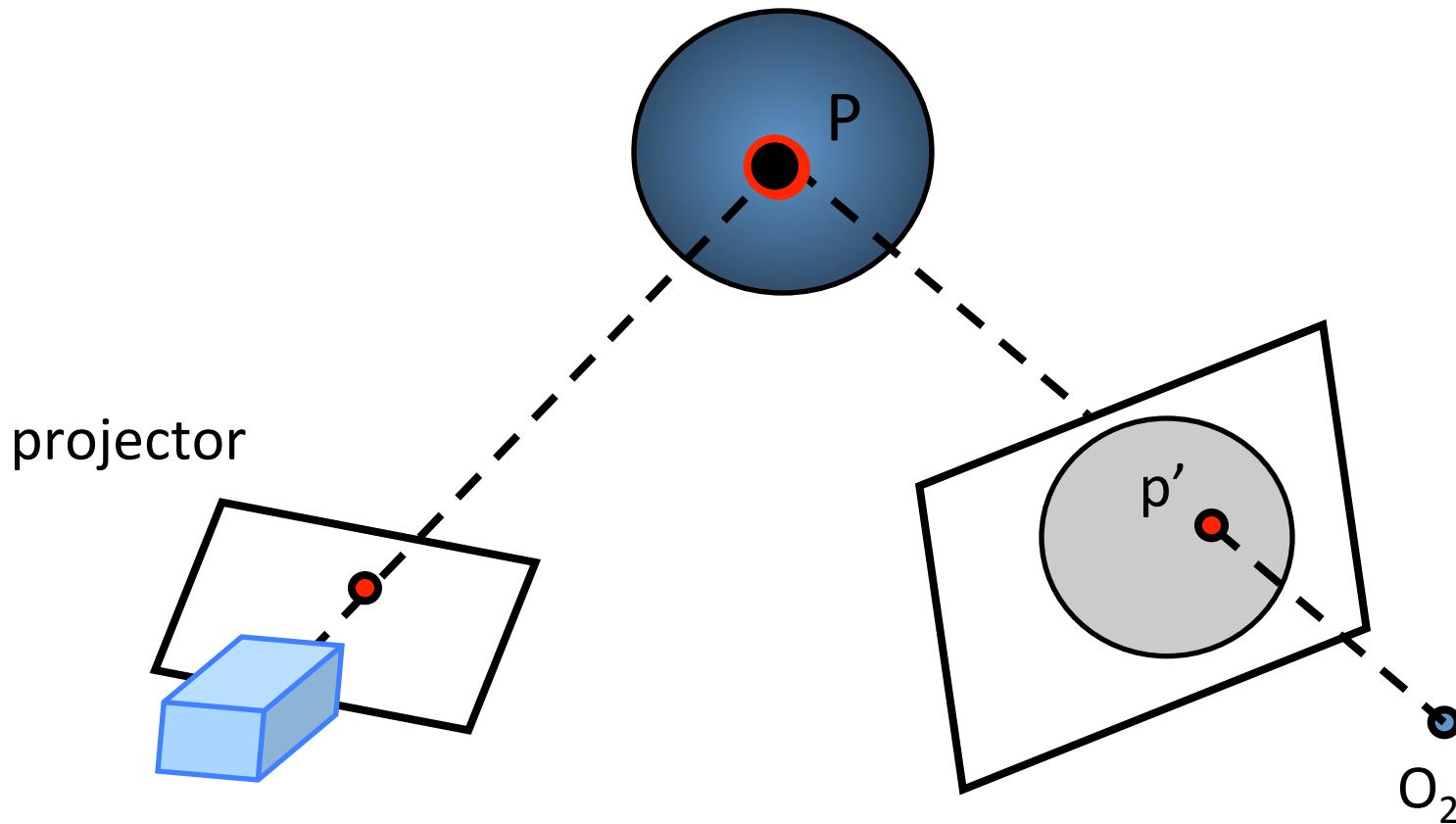
- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

[HZ] Chapters: 4, 9, 11

[FP] Chapters: 10

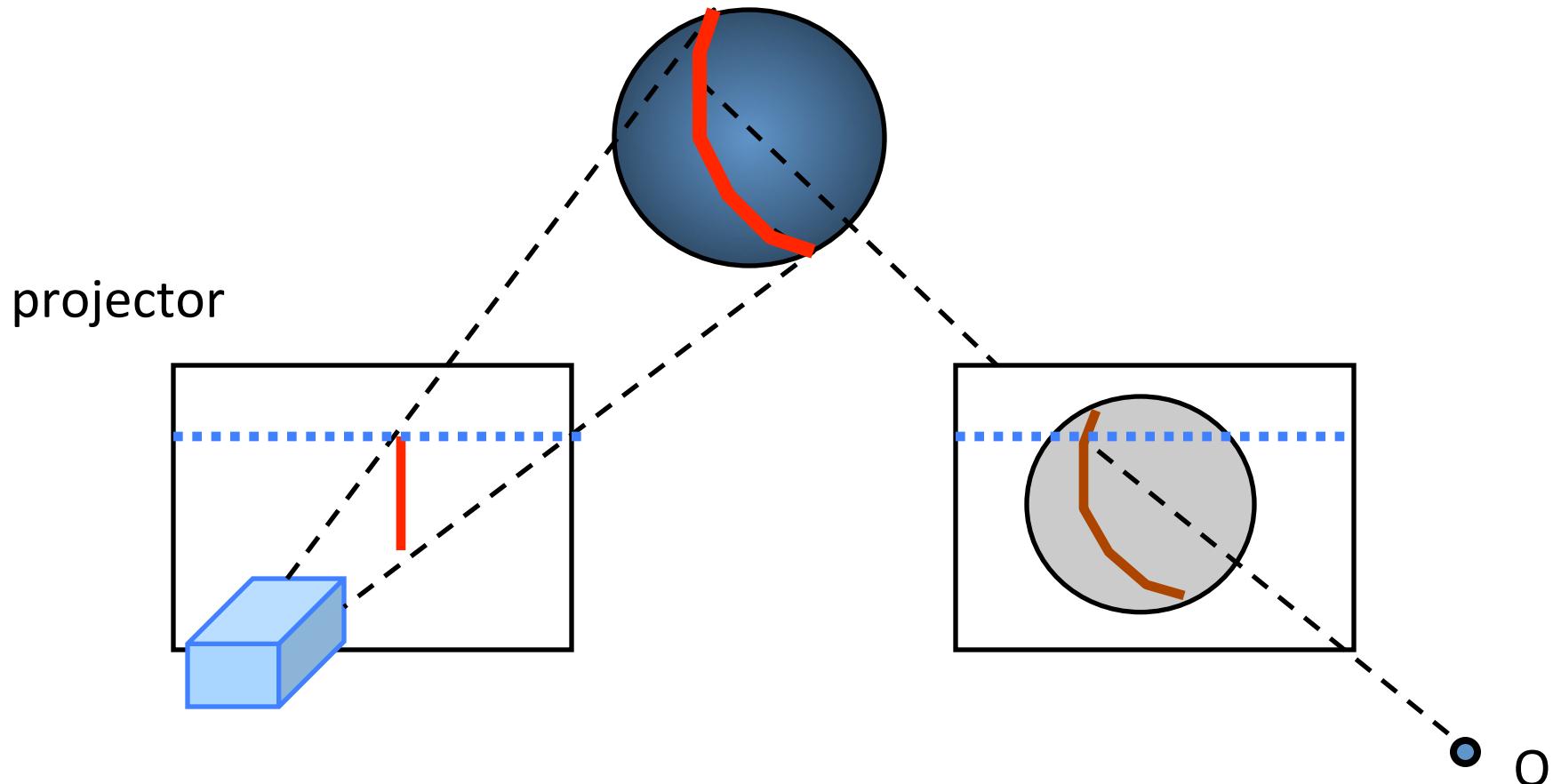
# Active stereo (point)



Replace one of the two cameras by a projector

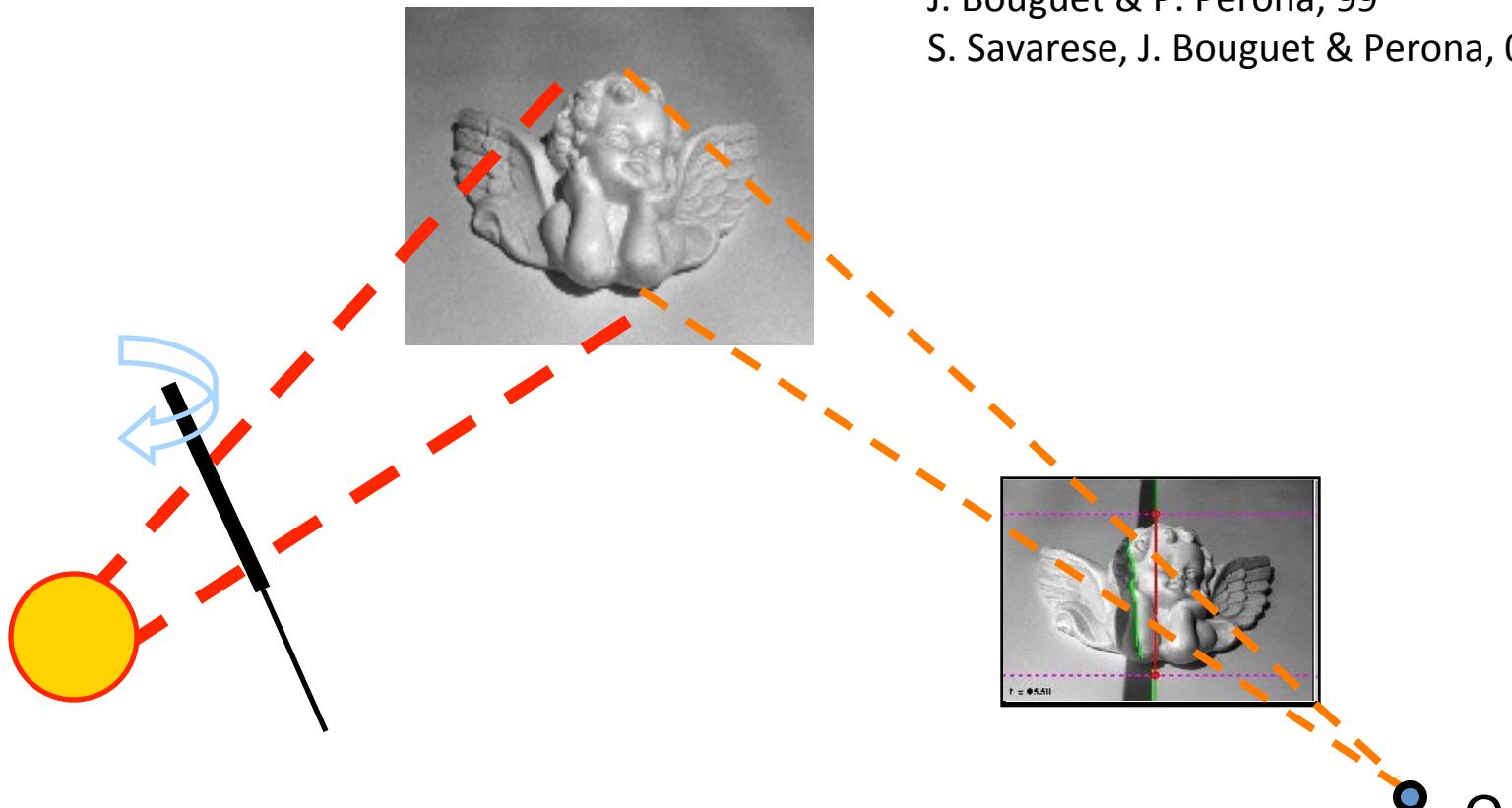
- Single camera
- Projector geometry calibrated
- What's the advantage of having the projector? Correspondence problem solved!

# Active stereo (stripe)



- Projector and camera are parallel
- Correspondence problem solved!

# Active stereo (shadows)



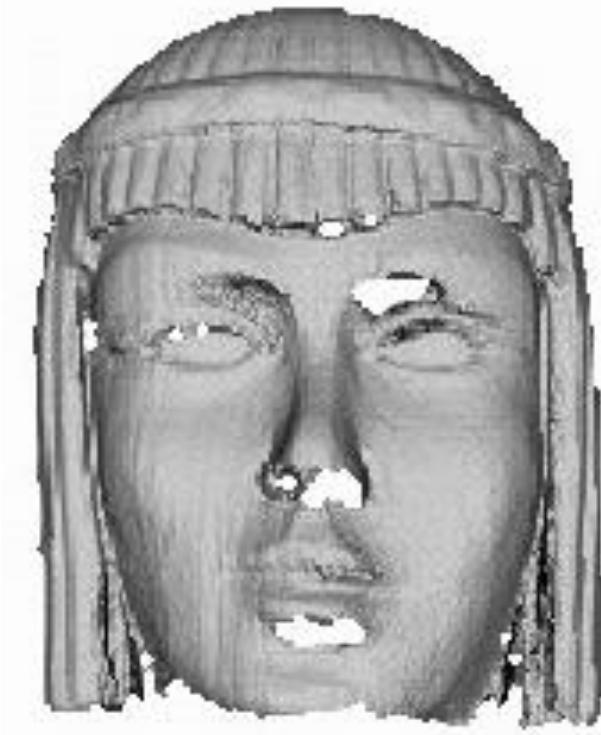
Light source

- 1 camera, 1 light source
- very cheap setup
- calibrated light source

J. Bouguet & P. Perona, 99  
S. Savarese, J. Bouguet & Perona, 00

# Active stereo (shadows)

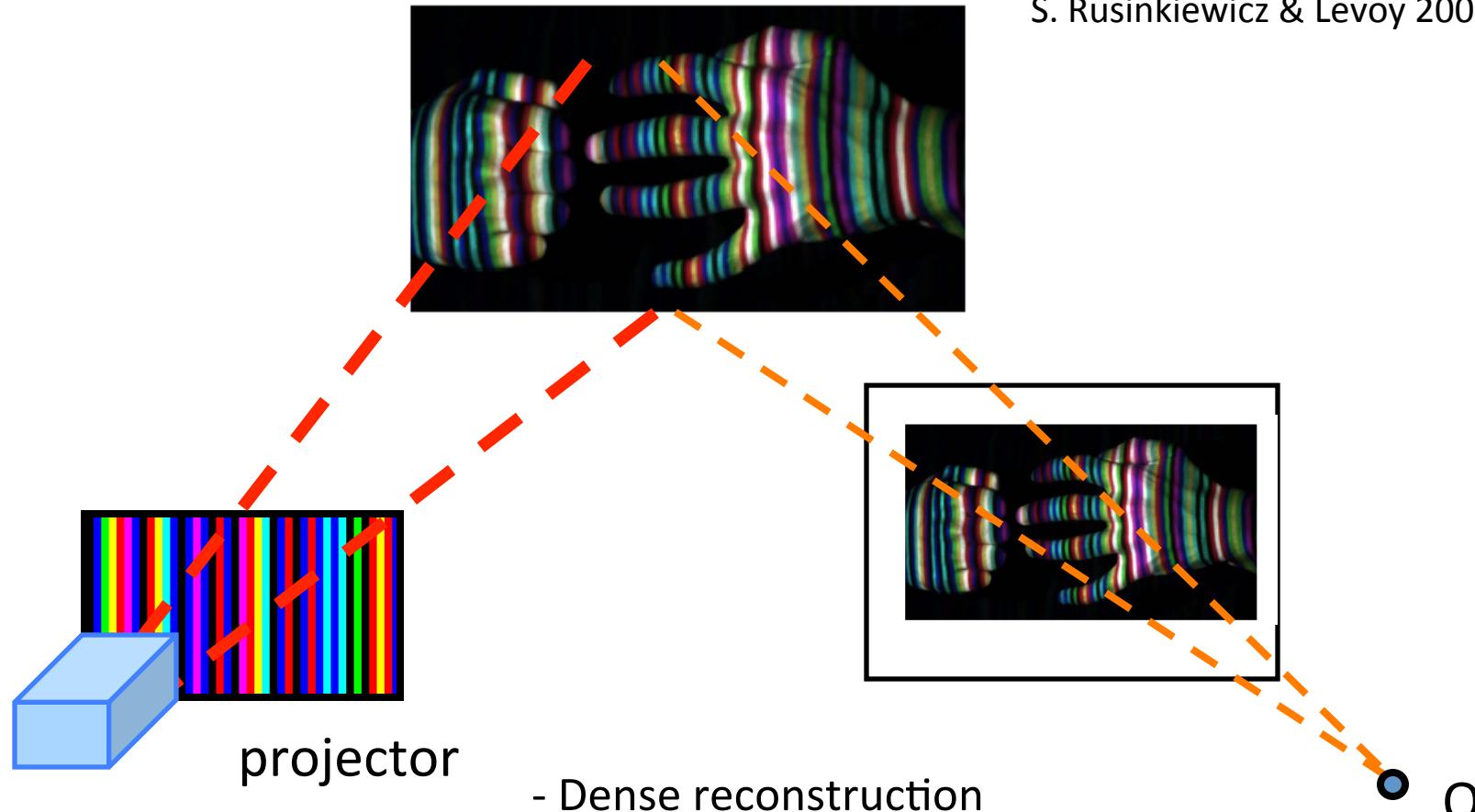
J. Bouguet & P. Perona, 99  
S. Savarese, J. Bouguet & Perona, 00



# Active stereo (color-coded stripes)

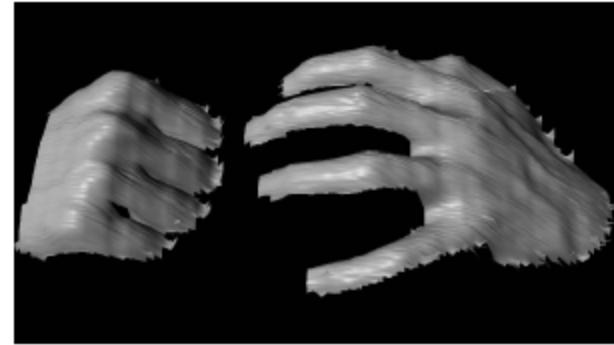
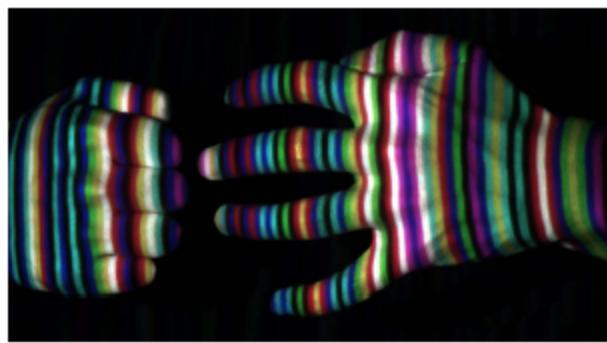
L. Zhang, B. Curless, and S. M. Seitz 2002

S. Rusinkiewicz & Levoy 2002



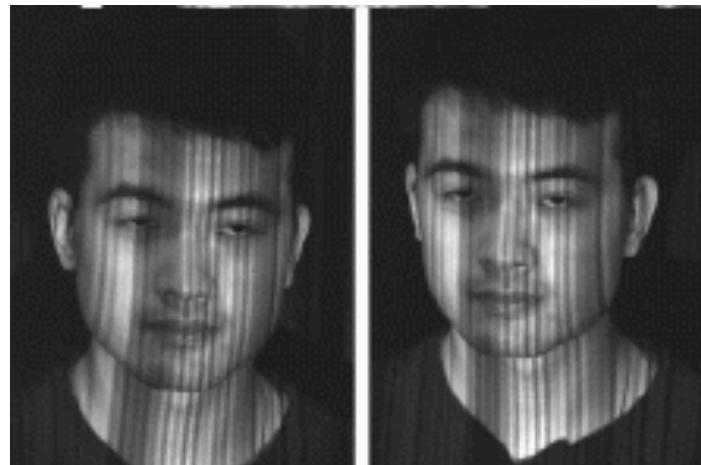
- Dense reconstruction
- Correspondence problem again
- Get around it by using color codes

# Active stereo (color-coded stripes)

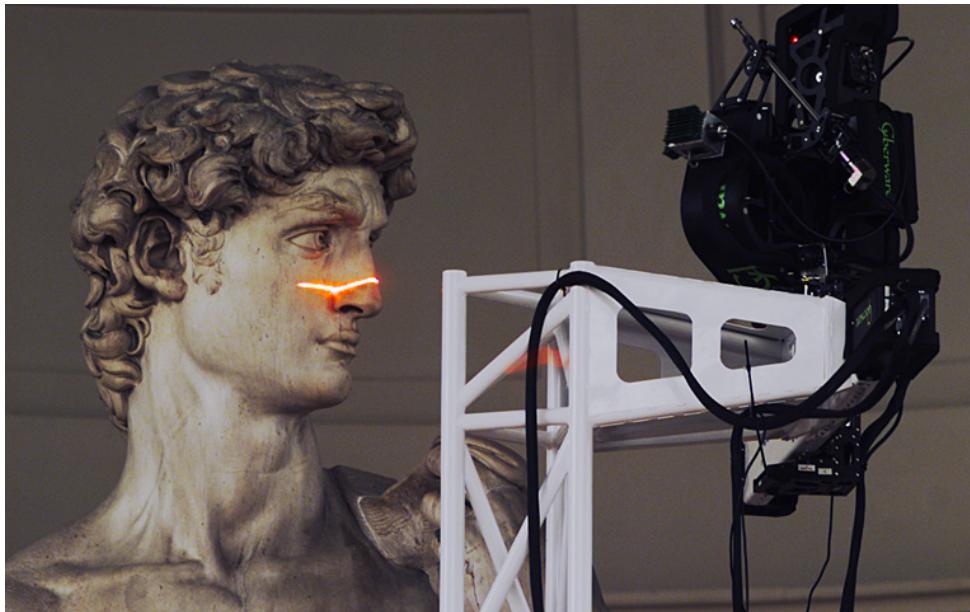


Rapid shape acquisition: Projector + stereo cameras

L. Zhang, B. Curless, and S. M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002



# Active stereo (stripe)



Digital Michelangelo Project  
<http://graphics.stanford.edu/projects/mich/>

- Optical triangulation
  - Project a single stripe of laser light
  - Scan it across the surface of the object
  - This is a very precise version of structured light scanning

# Laser scanned models



*The Digital Michelangelo Project*, Levoy et al.

Slide credit: S. Seitz

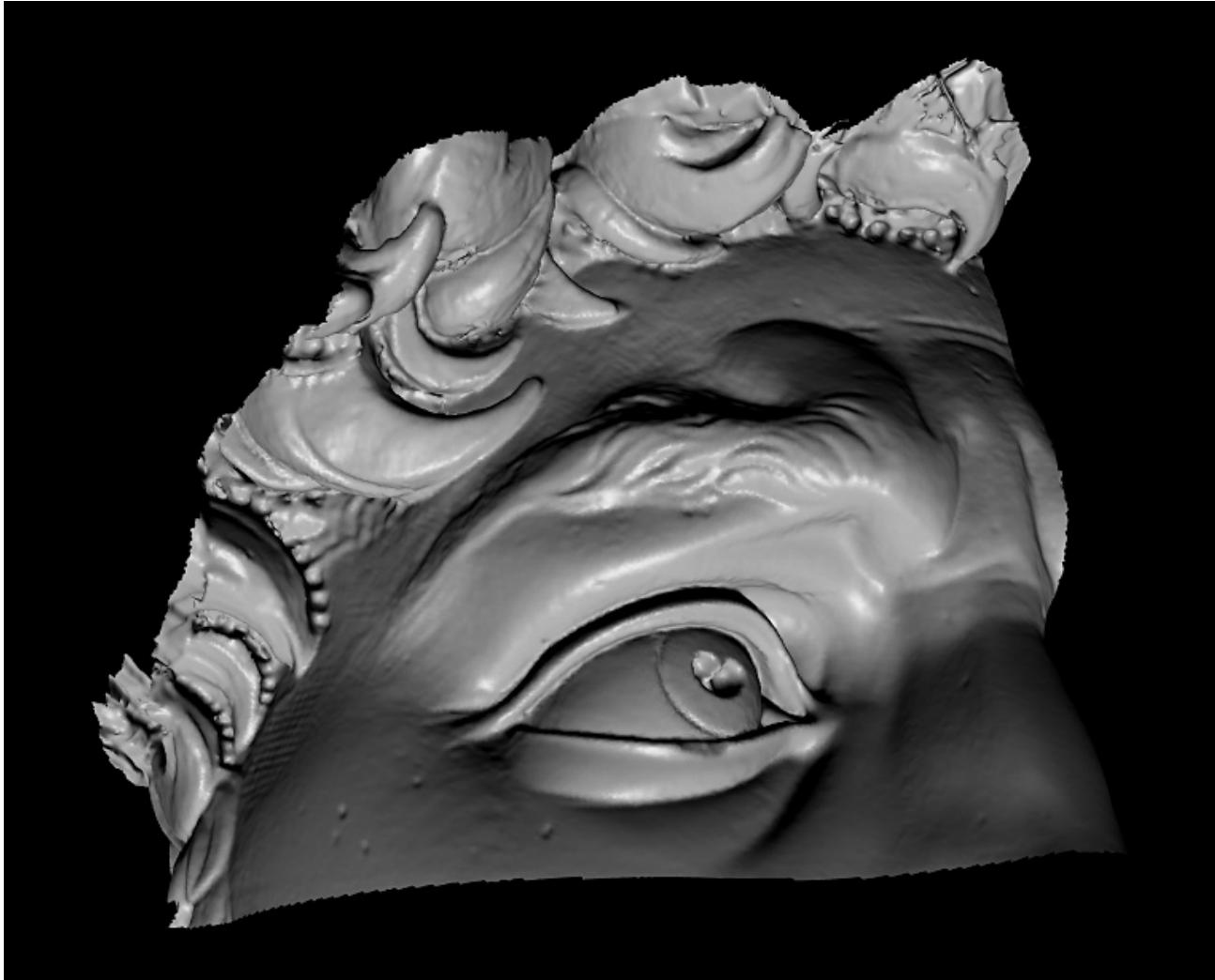
# Laser scanned models



*The Digital Michelangelo Project*, Levoy et al.

Slide credit: S. Seitz

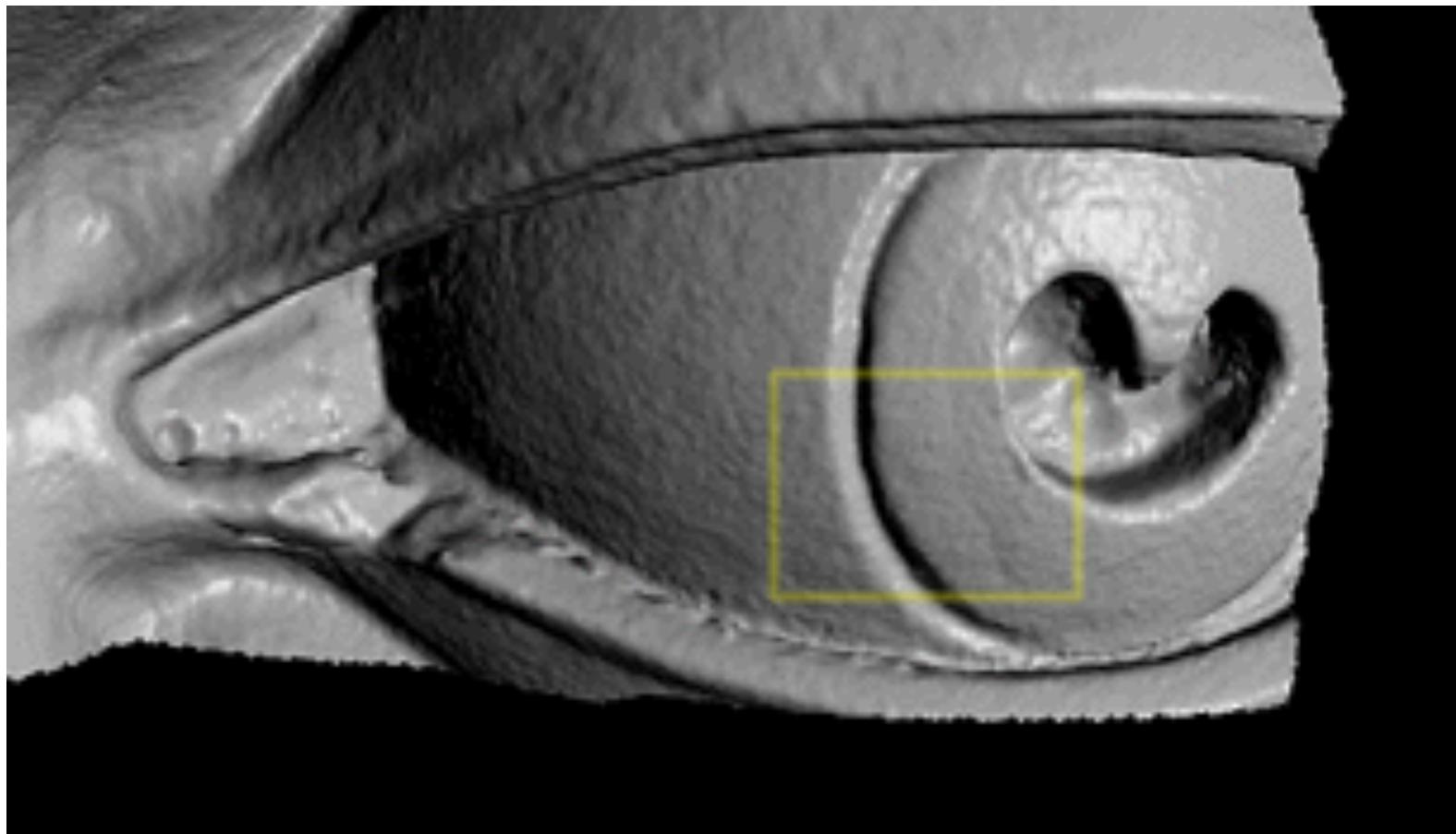
# Laser scanned models



*The Digital Michelangelo Project*, Levoy et al.

Slide credit: S. Seitz

# Laser scanned models

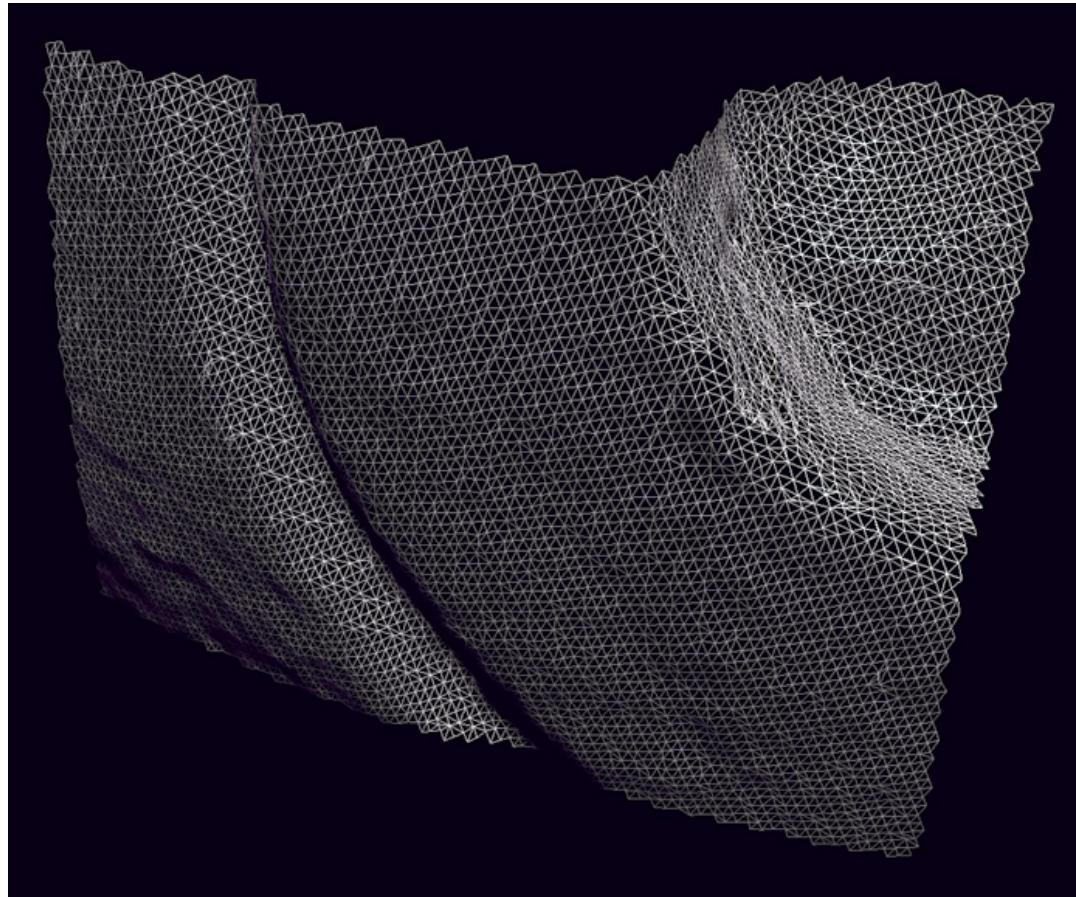


*The Digital Michelangelo Project*, Levoy et al.

Slide credit: S. Seitz

# Laser scanned models

1.0 mm resolution (56 million triangles)



*The Digital Michelangelo Project*, Levoy et al.

Slide credit: S. Seitz

# What we have learned today

- Introduction to stereo vision
- Epipolar geometry: a gentle intro
- Parallel images & image rectification
- Solving the correspondence problem
- Homographic transformation
- Active stereo vision system

**Reading:**

[HZ] Chapters: 4, 9, 11

[FP] Chapters: 10