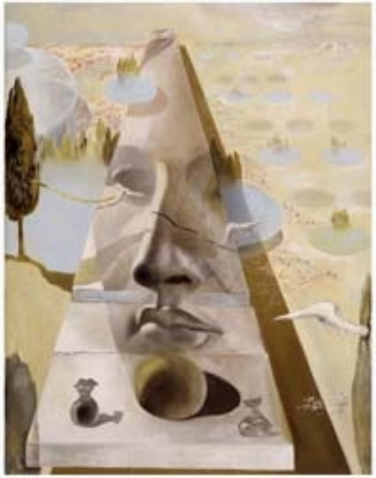
Lecture 6

Stereo Systems
Multi-view geometry

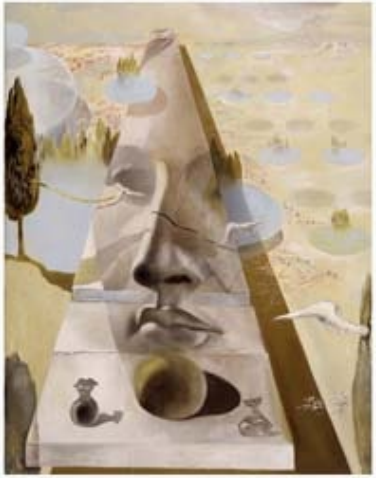


Professor Silvio Savarese

Computational Vision and Geometry Lab

Lecture 6

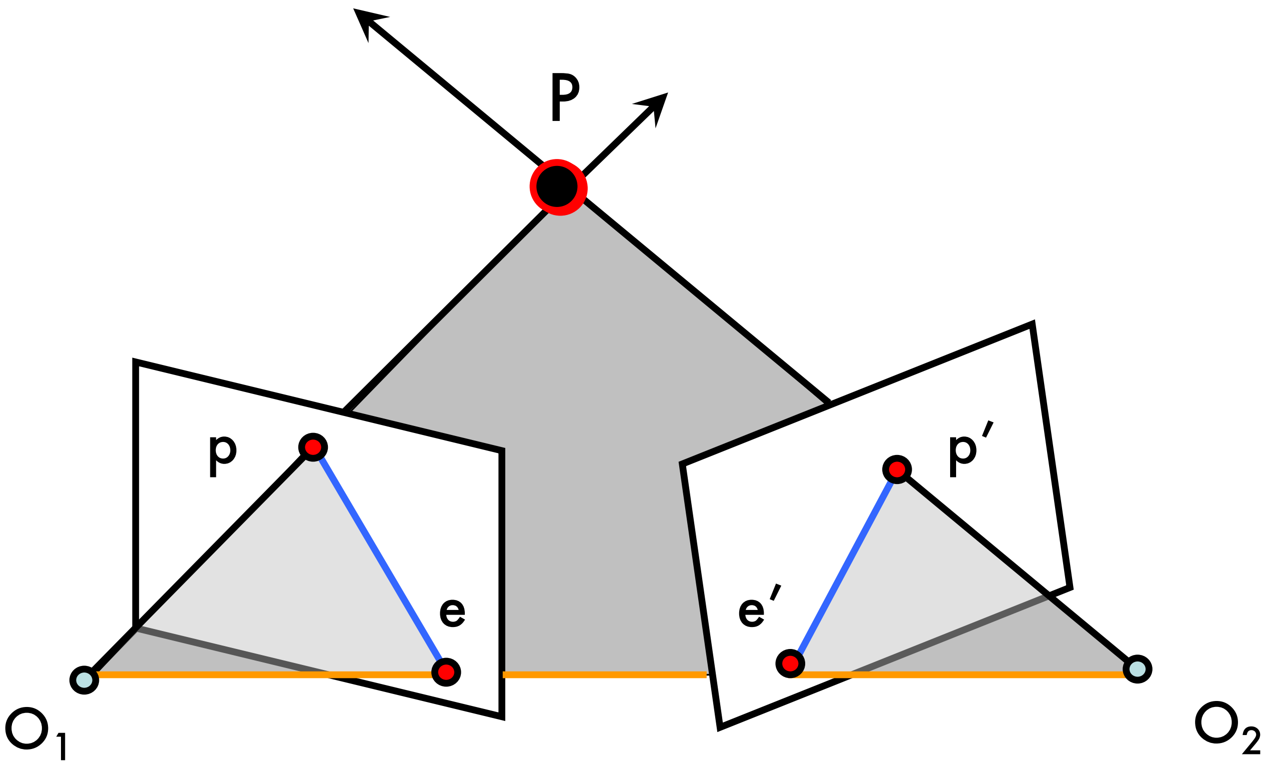
Stereo Systems
Multi-view geometry



- Stereo systems
  - Rectification
  - Correspondence problem
- Multi-view geometry
  - The SFM problem
  - Affine SFM

Reading: [AZ] Chapter: 9 "Epip. Geom. and the Fundam. Matrix Transf."
[AZ] Chapter: 18 "N view computational methods"
[FP] Chapters: 7 "Stereopsis"
[FP] Chapters: 8 "Structure from Motion"

Epipolar geometry

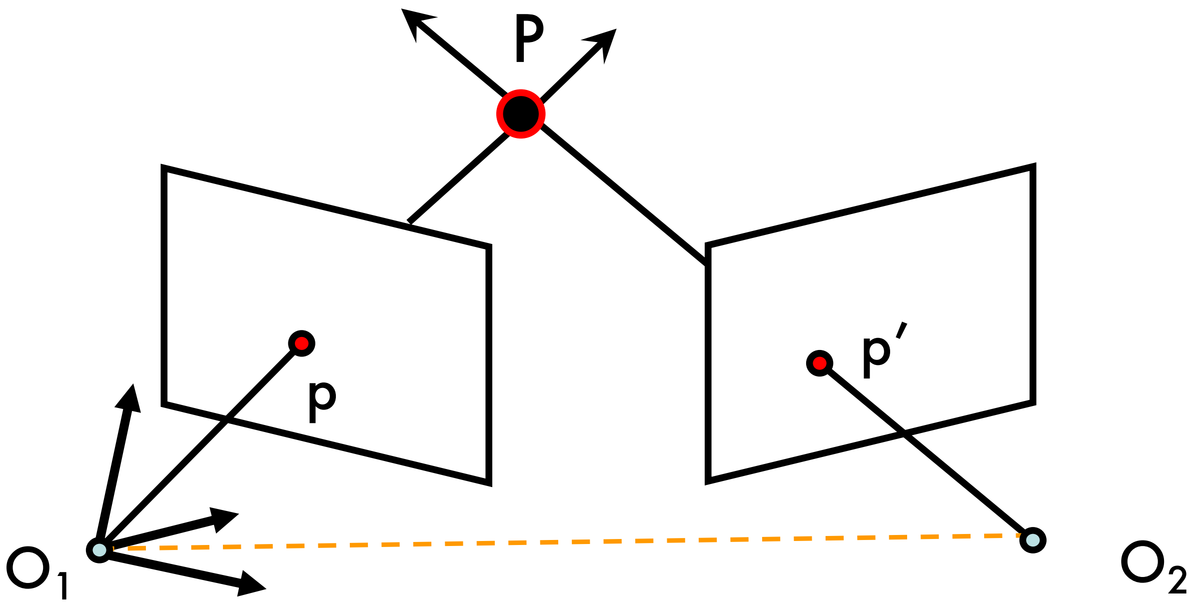


- Epipolar Plane
- Baseline
- Epipolar Lines

- Epipoles e, e′

  = intersections of baseline with image planes
  = projections of the other camera center

Epipolar Constraint



$$\boxed{p^T \; E \; p' = 0}$$

$$E = [T_\times] \cdot R$$

E = Essential Matrix

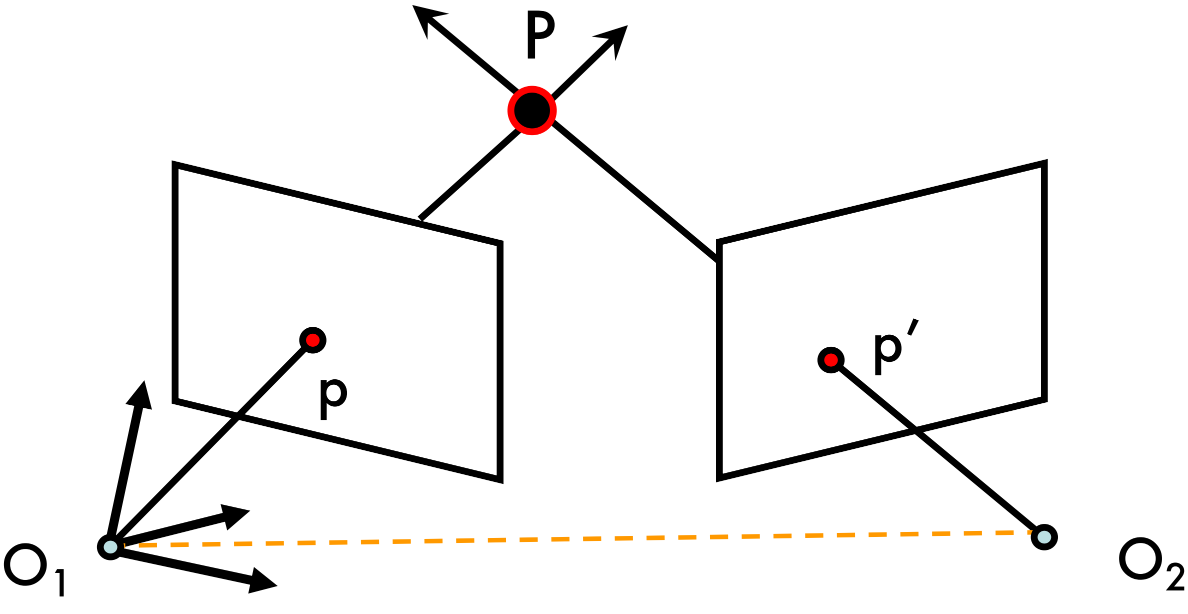
(Longuet-Higgins, 1981)

Essential matrix

$$\mathbf{E} = \left[\mathbf{T}_\times\right] \cdot \mathbf{R}$$

$$\mathbf{E} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \mathbf{R}$$

Epipolar Constraint



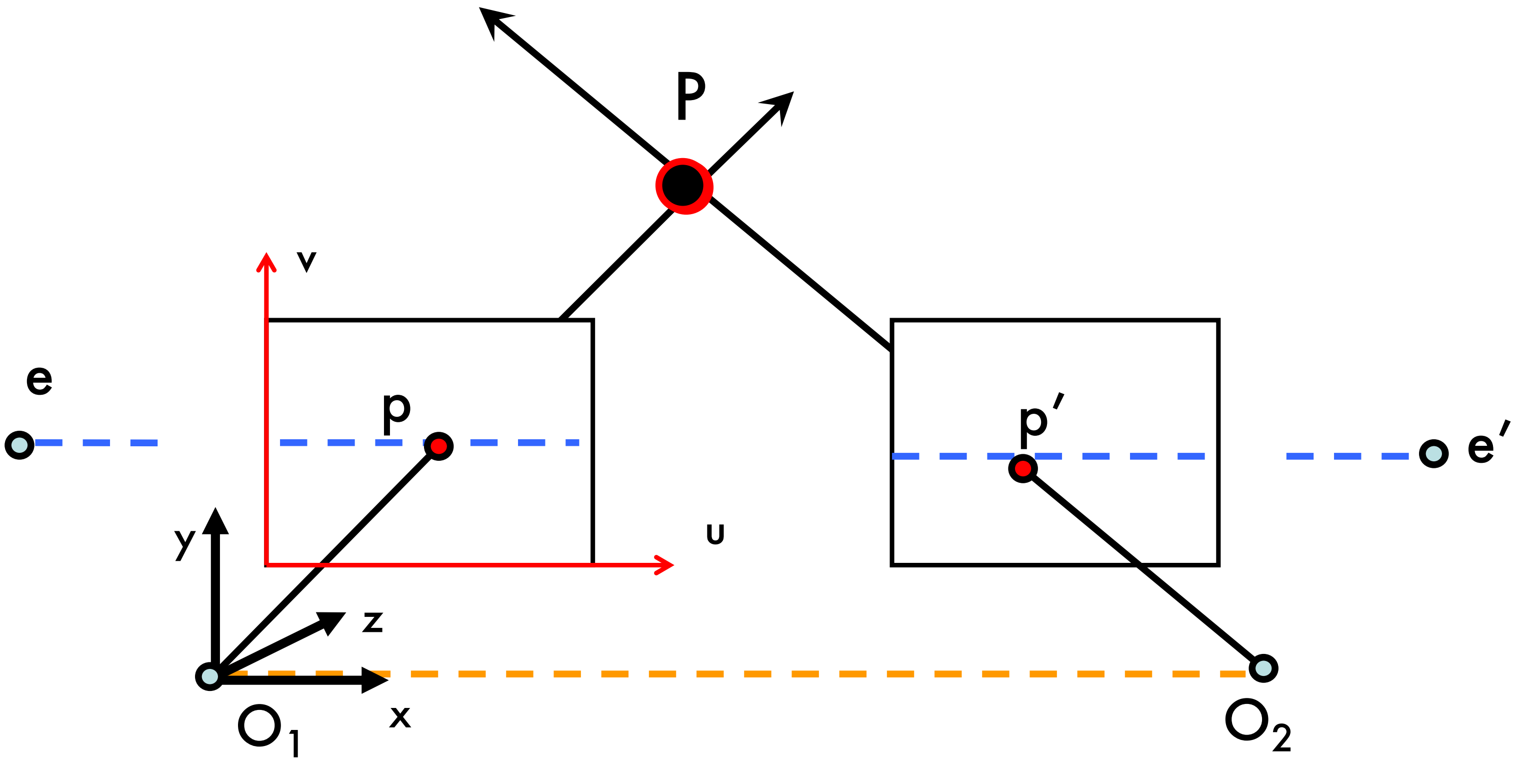
$$\boxed{p^{T}\, F\, p' = 0}$$

$$F = K^{-T} \cdot [T_{\times}] \cdot R\, K'^{-1}$$

F = Fundamental Matrix

(Faugeras and Luong, 1992)

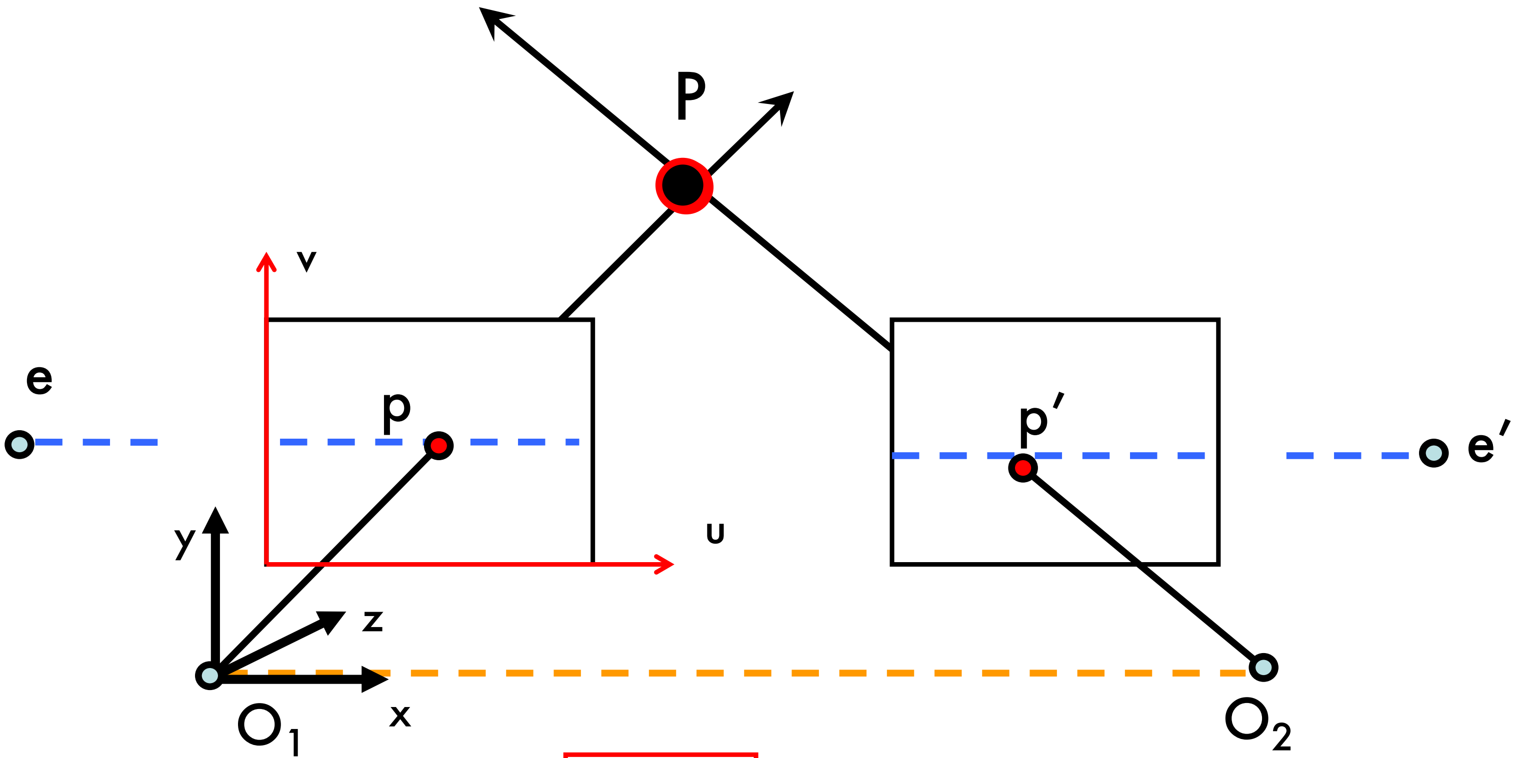
Parallel image planes



- Epipolar lines are horizontal
- Epipoles go to infinity
- v-coordinates are equal

$$p = \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix} \quad \text{Same} \quad p' = \begin{bmatrix} p'_u \\ p_v \\ 1 \end{bmatrix}$$

Parallel image planes



$$\boxed{E = ?}$$

Hint :

$\boldsymbol{R} = I \qquad \boldsymbol{T} = (T, 0, 0)$

$K_1 = K_2$ = known

x parallel to O_1O_2

Essential matrix for parallel images

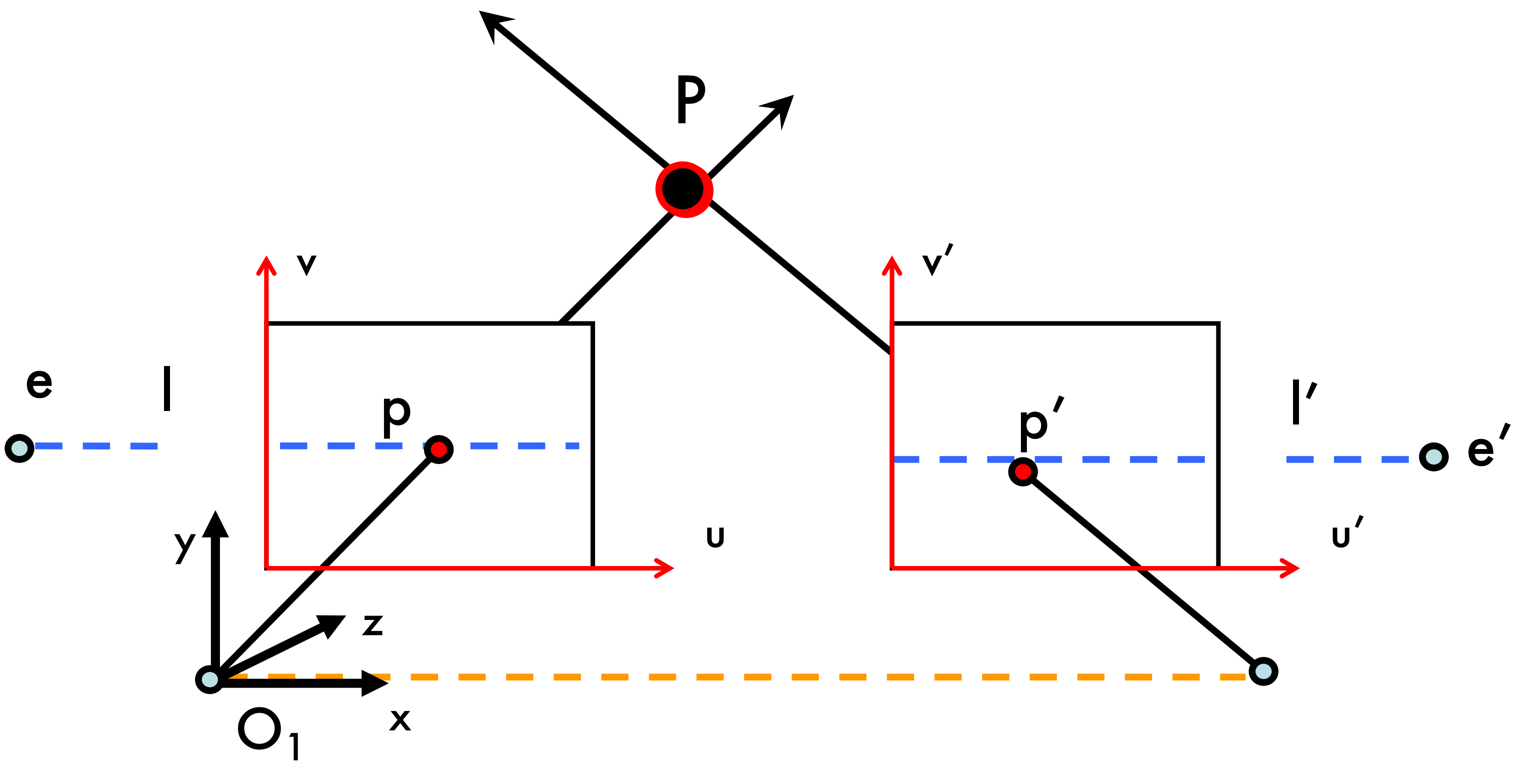
$$\mathbf{E} = \left[\mathbf{T}_{\times}\right] \cdot \mathbf{R}$$

$$\mathbf{E} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \mathbf{R} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

$\mathbf{T} = [\ \mathrm{T}\ \ 0\ \ 0\ ]$

$\mathbf{R} = \mathbf{I}$

Parallel image planes

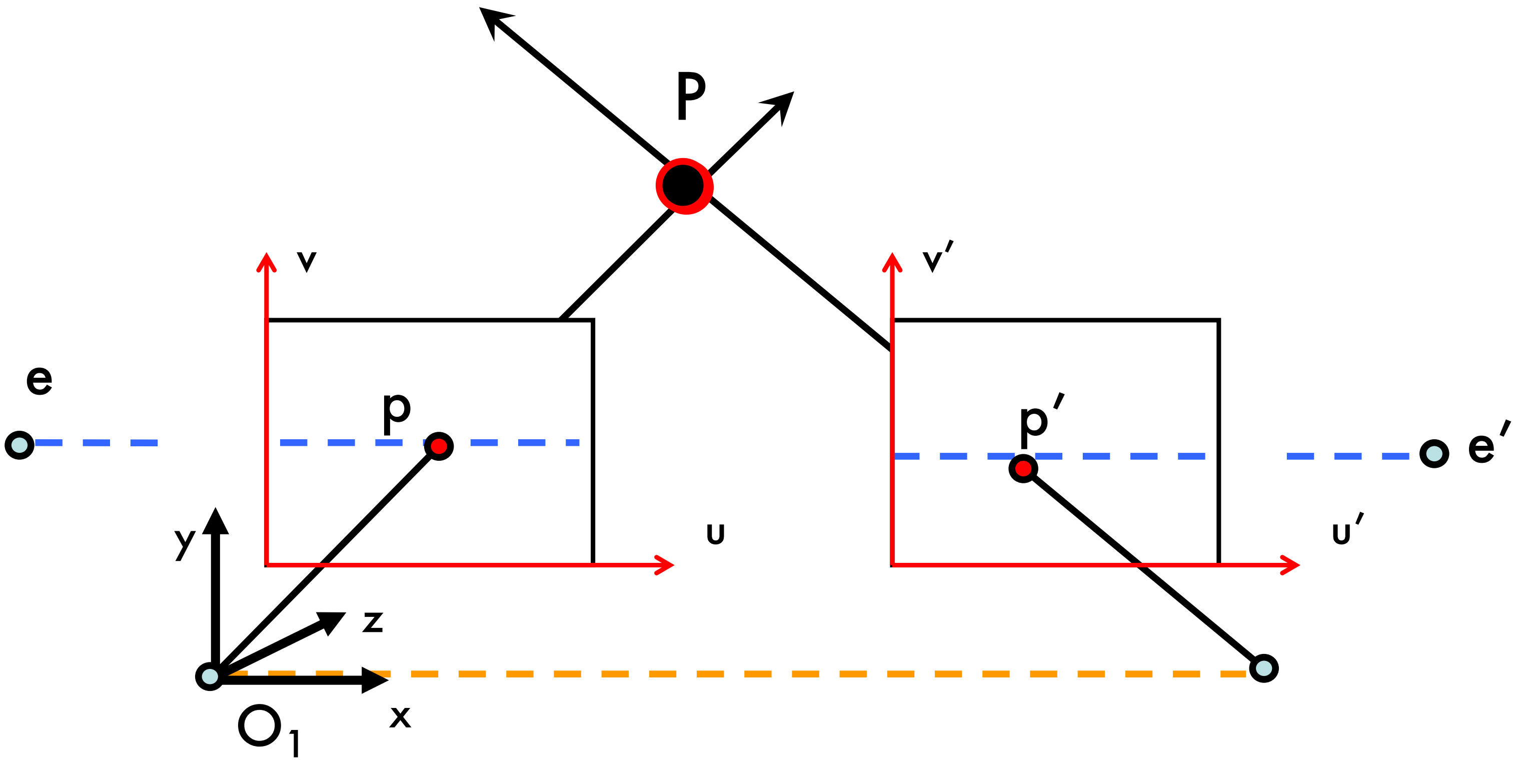


What are the directions of epipolar lines?

$$l = \mathrm{E}\, p' = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -T \\ T v' \end{bmatrix}$$

horizontal!

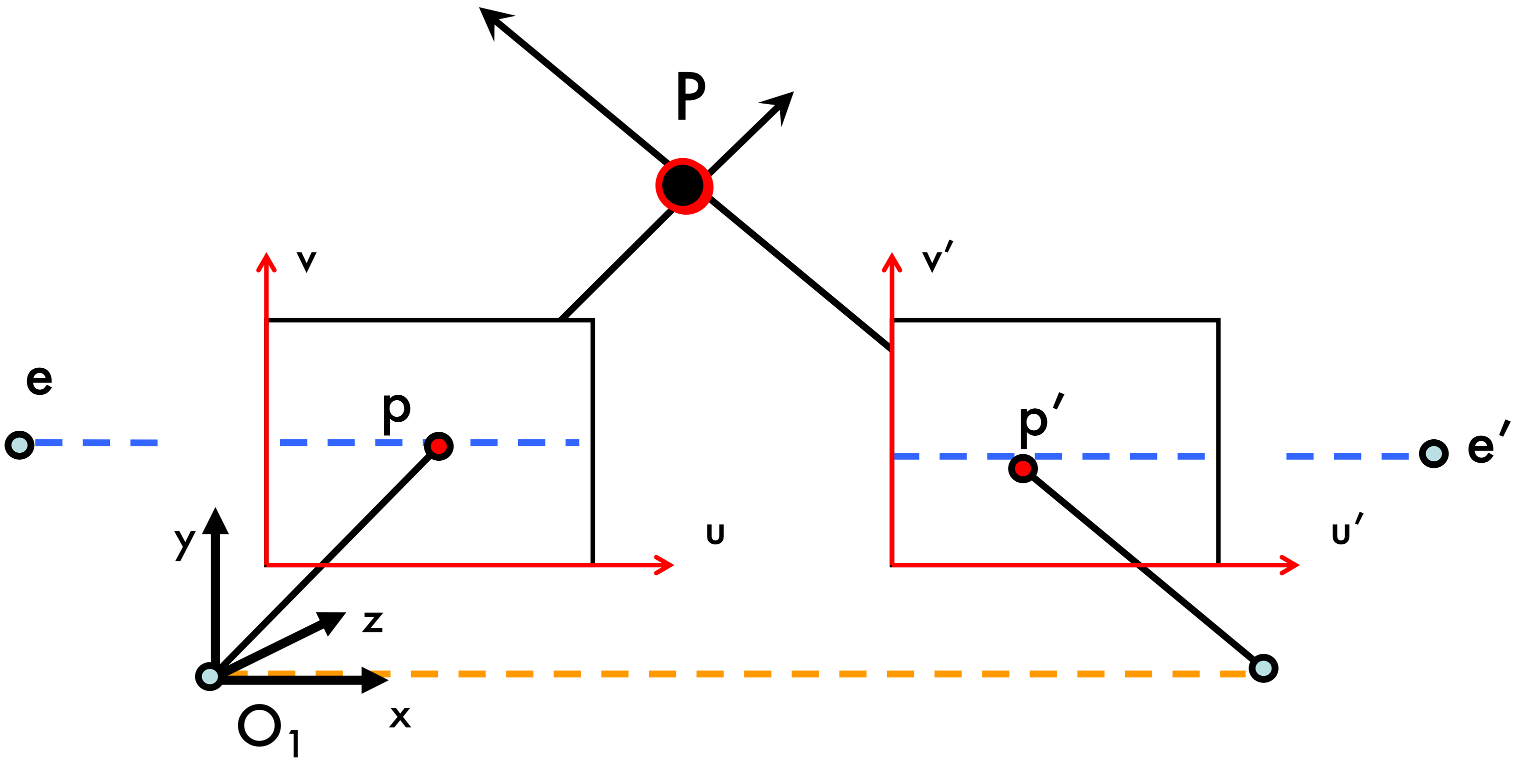
Parallel image planes



How are p and p′ related?

$$p^T \cdot E\, p' = 0$$

Parallel image planes

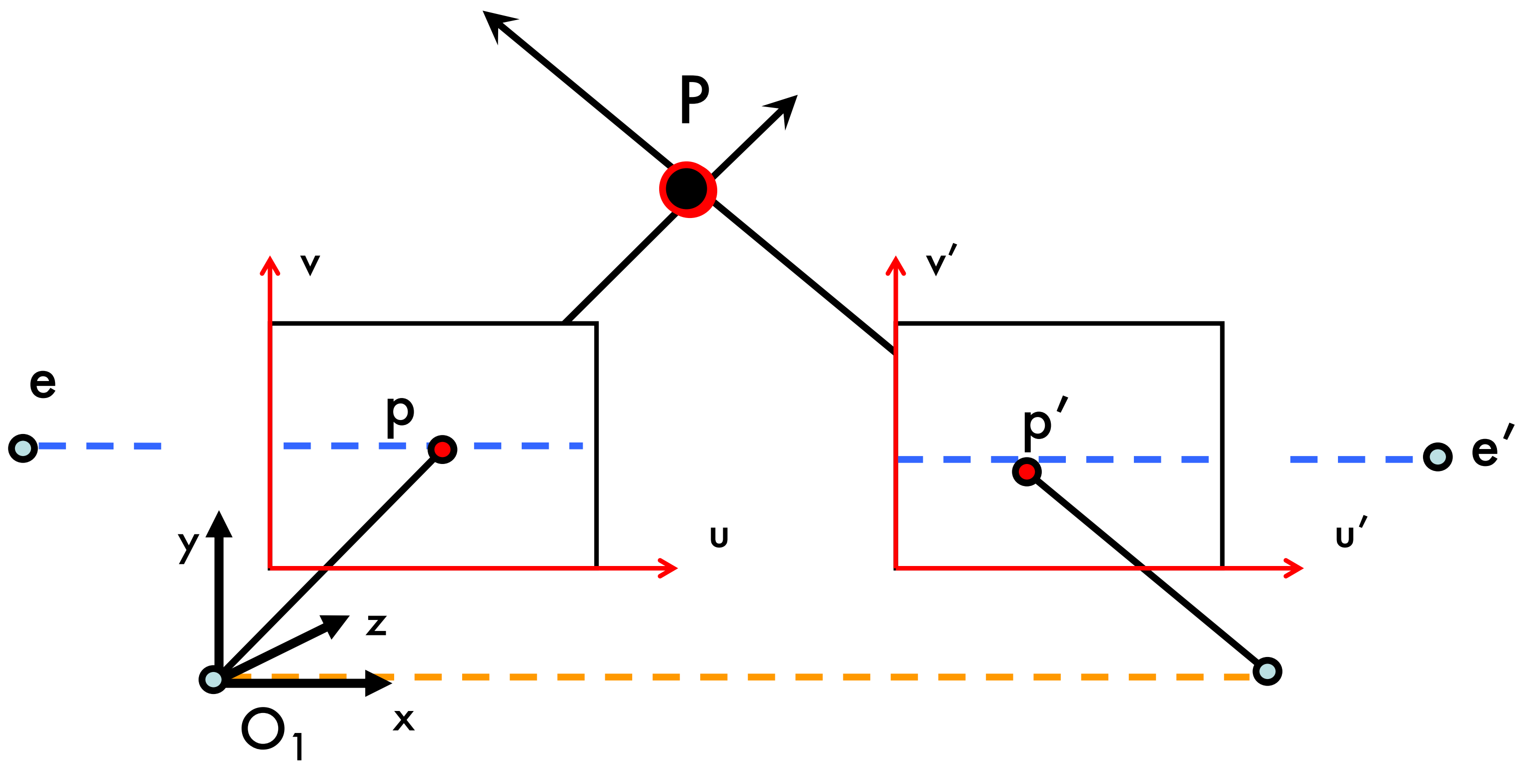


How are p and p′ related?

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

$$\Rightarrow v = v'$$

Parallel image planes

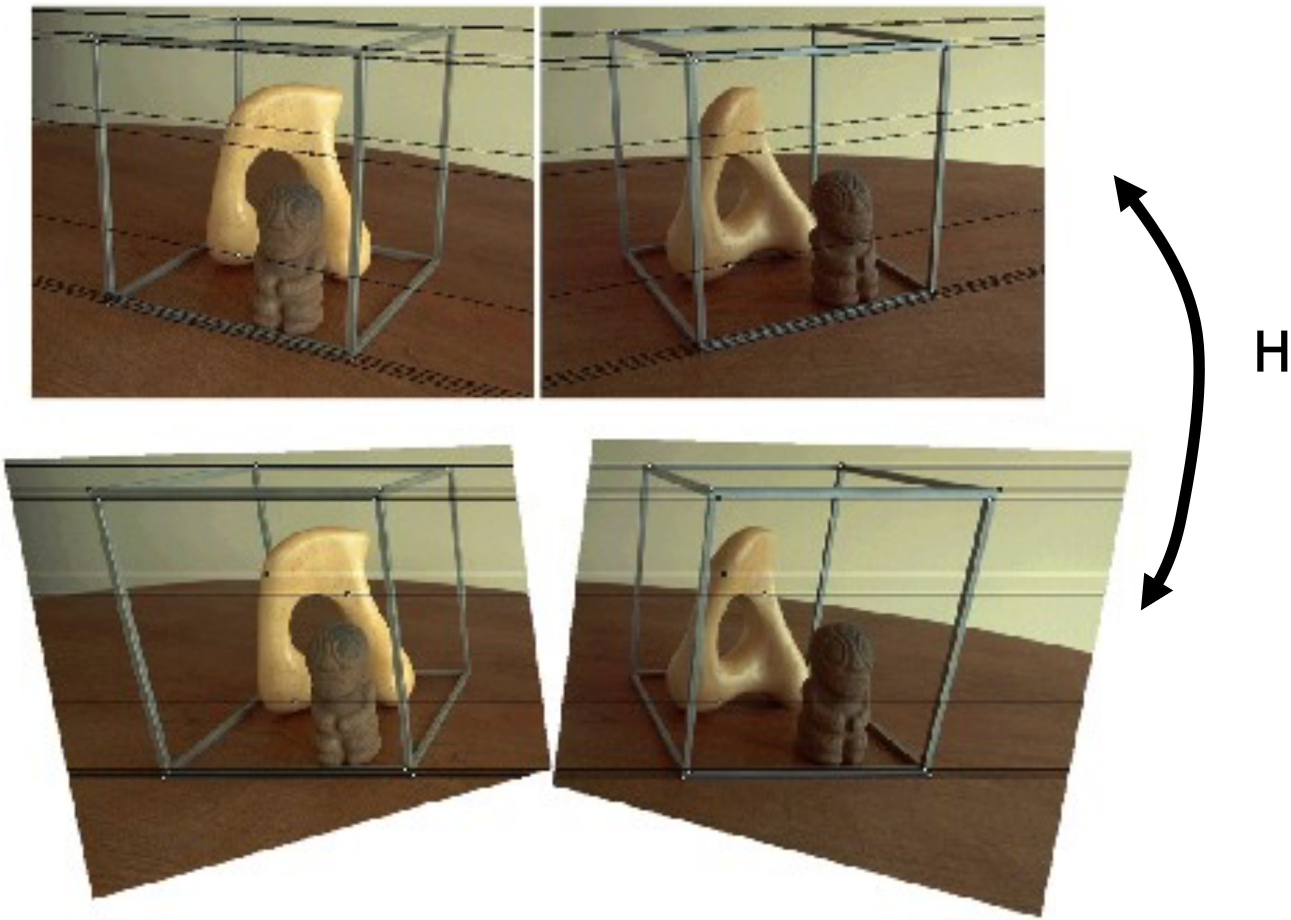


Rectification: making two images "parallel"

Why it is useful?

- Epipolar constraint → v = v'
- New views can be synthesized by linear interpolation

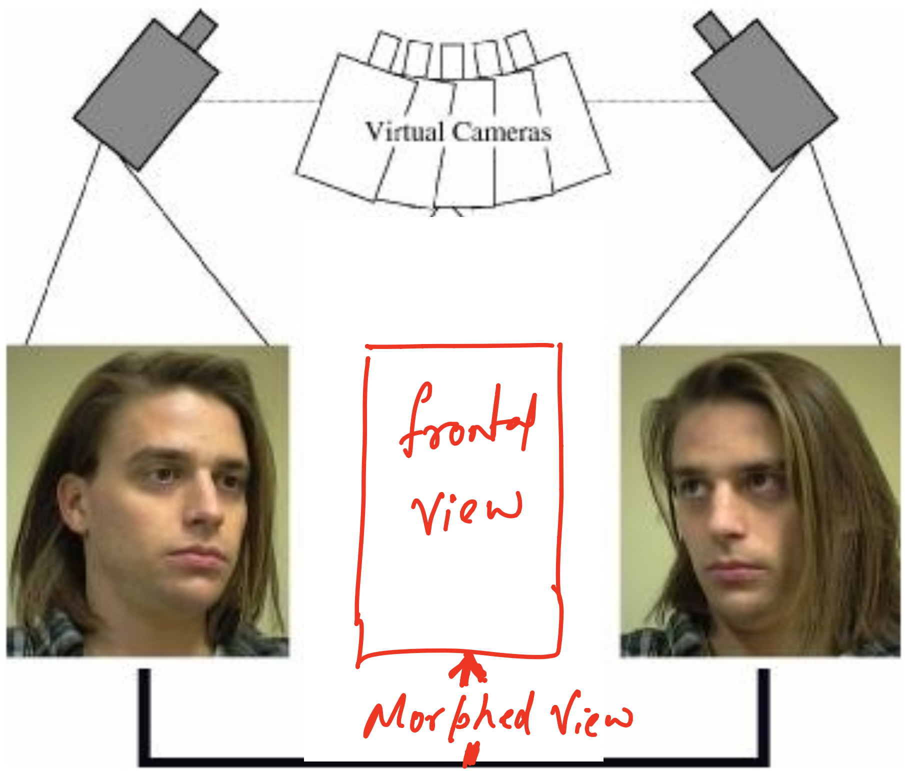
Rectification: making two images “parallel”



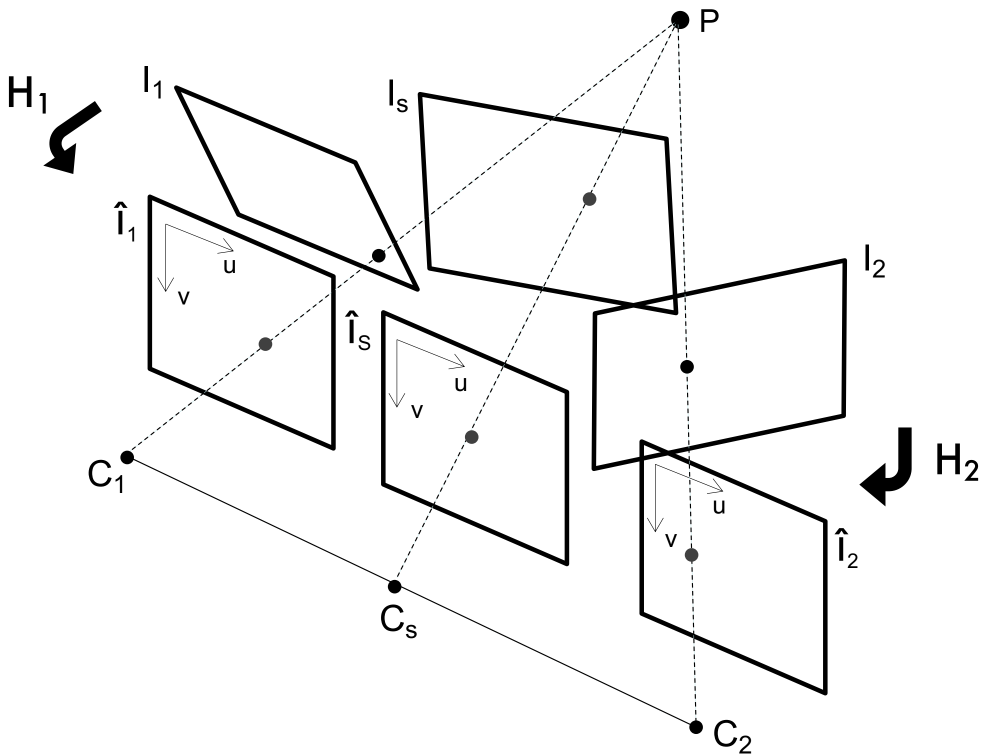
Courtesy figure S. Lazebnik

Application: view morphing

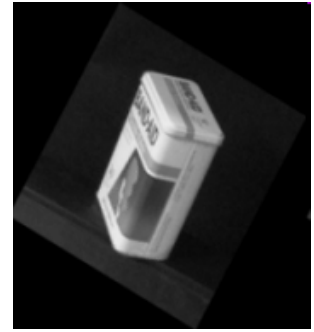
S. M. Seitz and C. R. Dyer, *Proc. SIGGRAPH 96*, 1996, 21-30

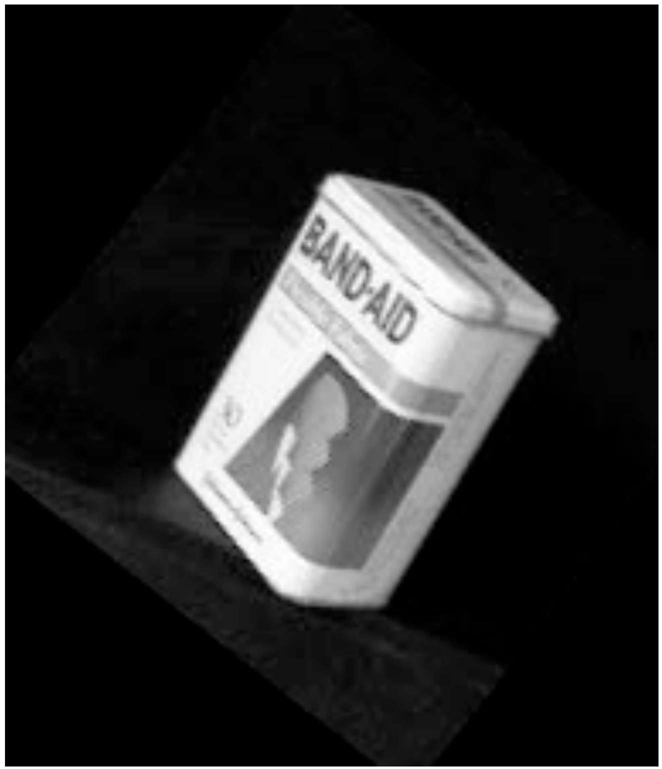


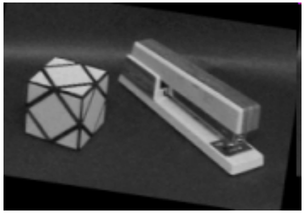
Rectification

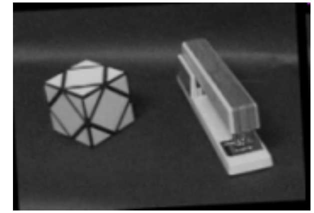


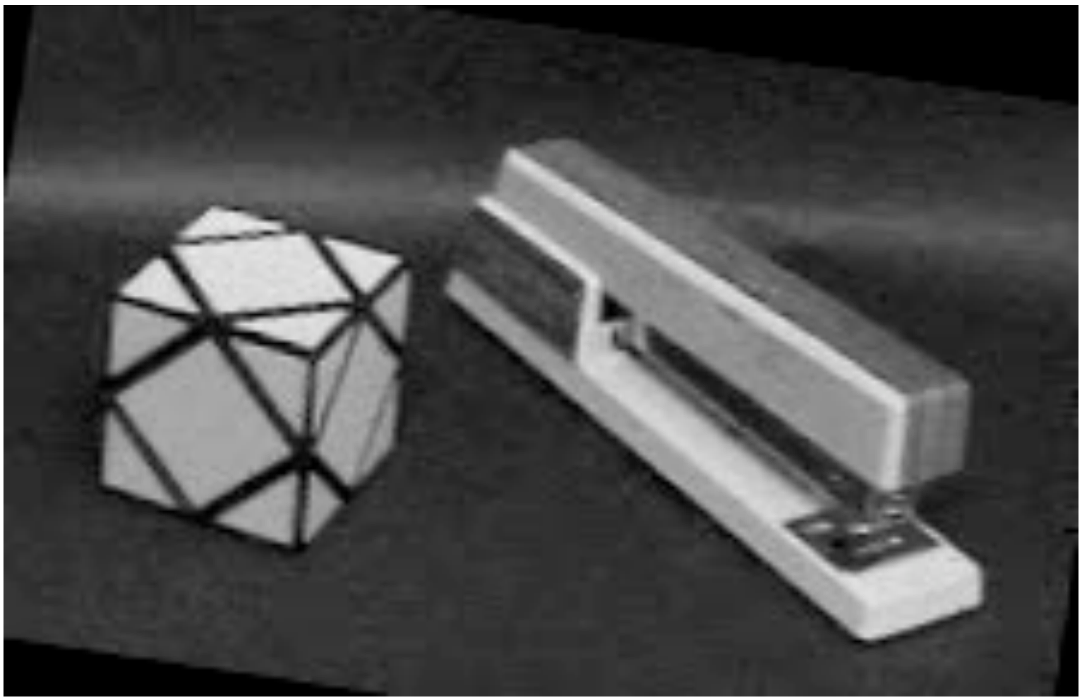

BAND-AID

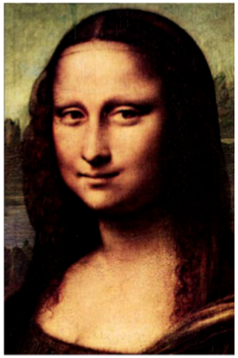

BAND-AID

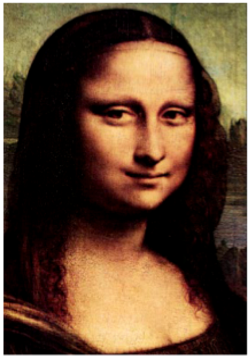

BAND-AID

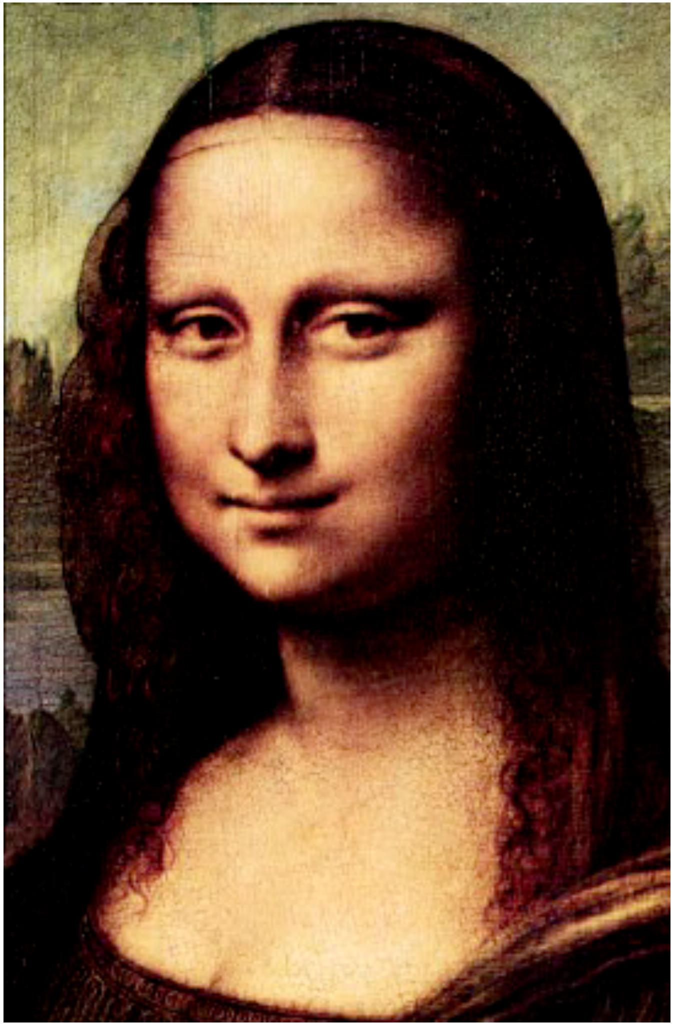










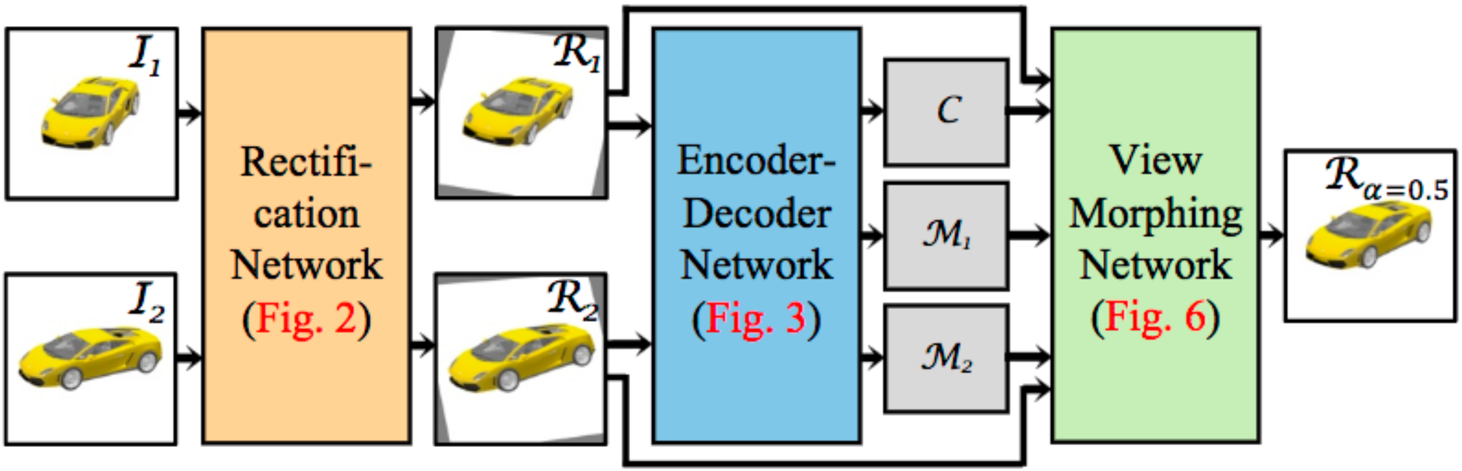


From its reflection!



Deep view morphing

D. Ji, J. Kwon, M. McFarland, S. Savarese, CVPR 2017

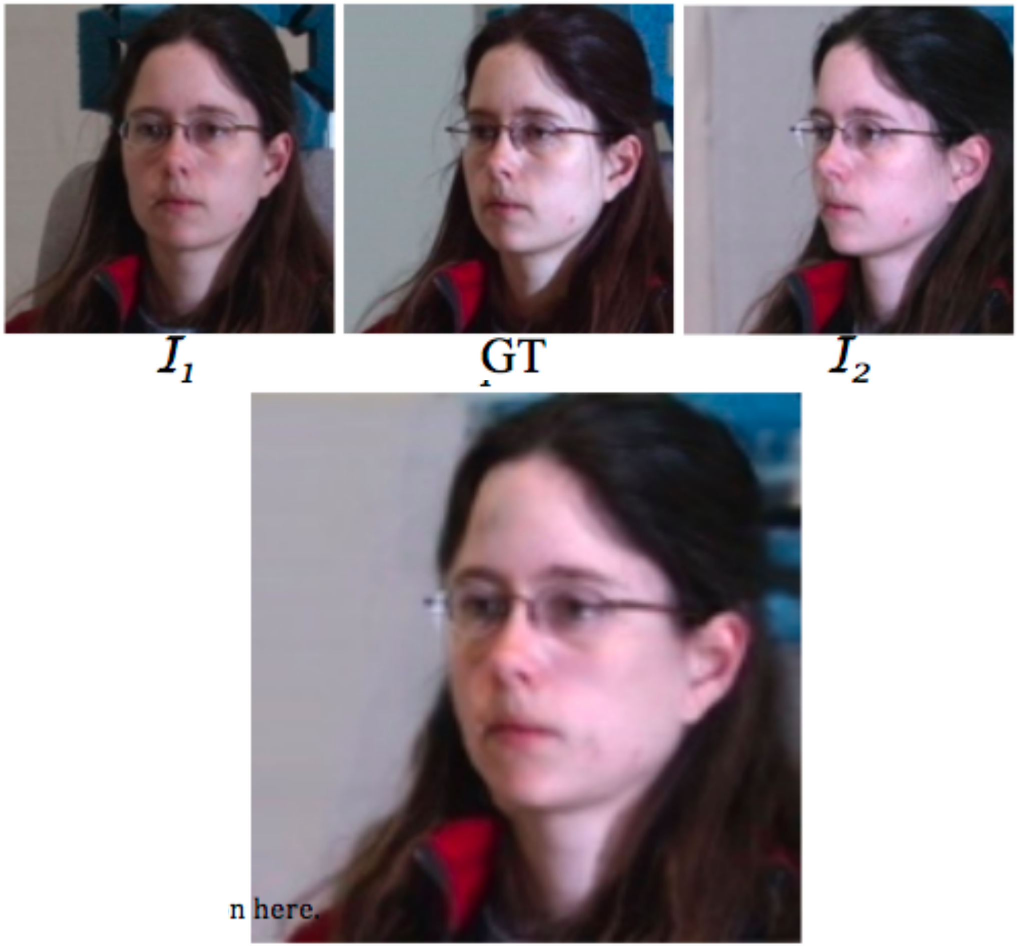


Deep view morphing

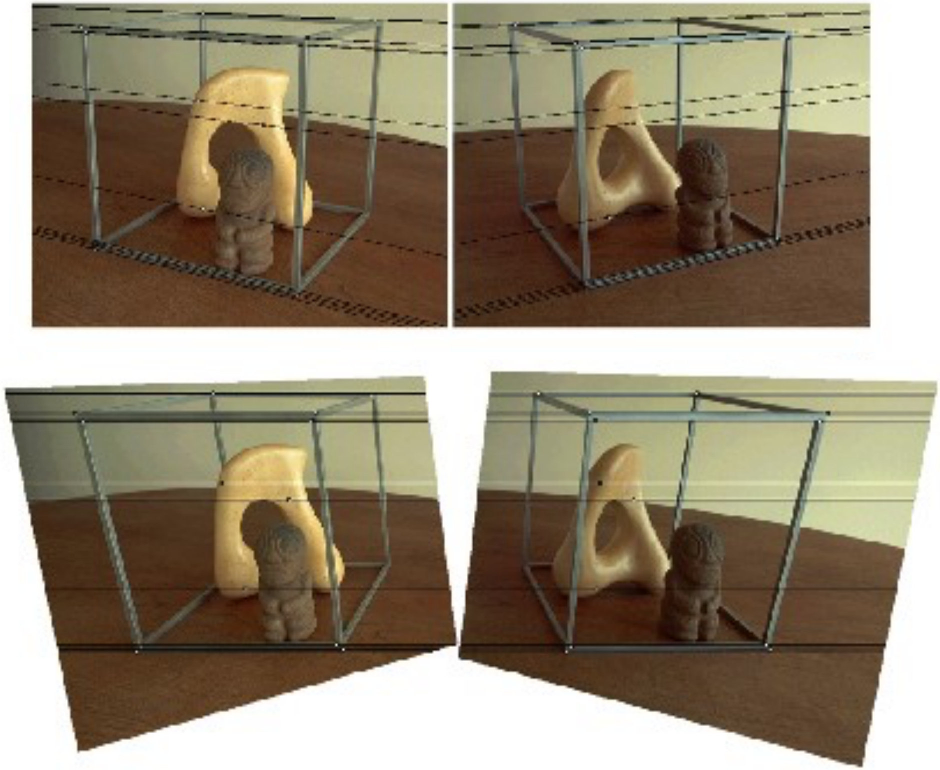
D. Ji, J. Kwon, M. McFarland, S. Savarese, CVPR 2017

Deep view morphing

D. Ji, J. Kwon, M. McFarland, S. Savarese, CVPR 2017

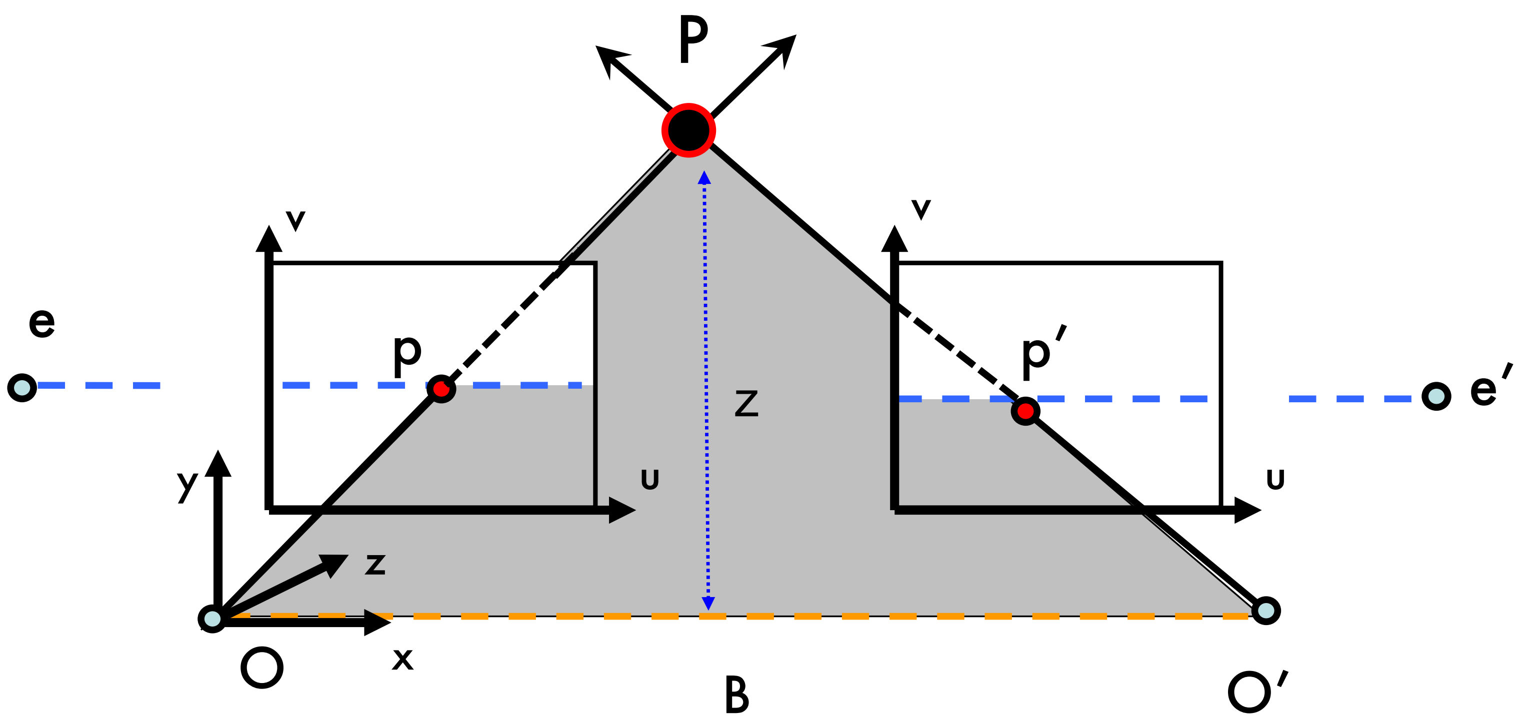


Why are parallel images useful?



- Makes triangulation easy
- Makes the correspondence problem easier

Point triangulation



$$p = \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix} \qquad p' = \begin{bmatrix} p'_u \\ p_v \\ 1 \end{bmatrix}$$

$$\text{disparity} = p_u - p'_u \propto \frac{B \cdot f}{z} \qquad \text{[Eq. 1]}$$

Disparity is inversely proportional to depth z!

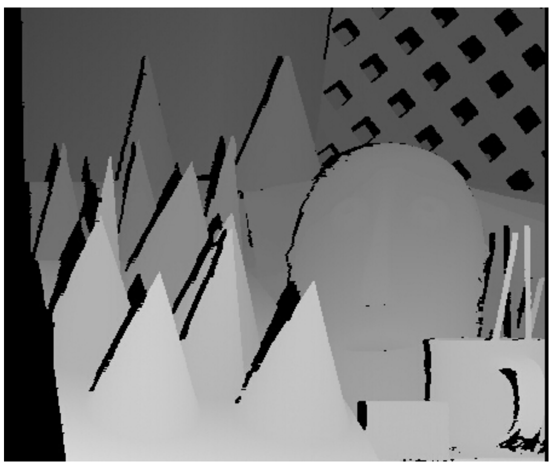
Disparity maps

http://vision.middlebury.edu/stereo/

$$p_u - p'_u \propto \frac{B \cdot f}{z}$$

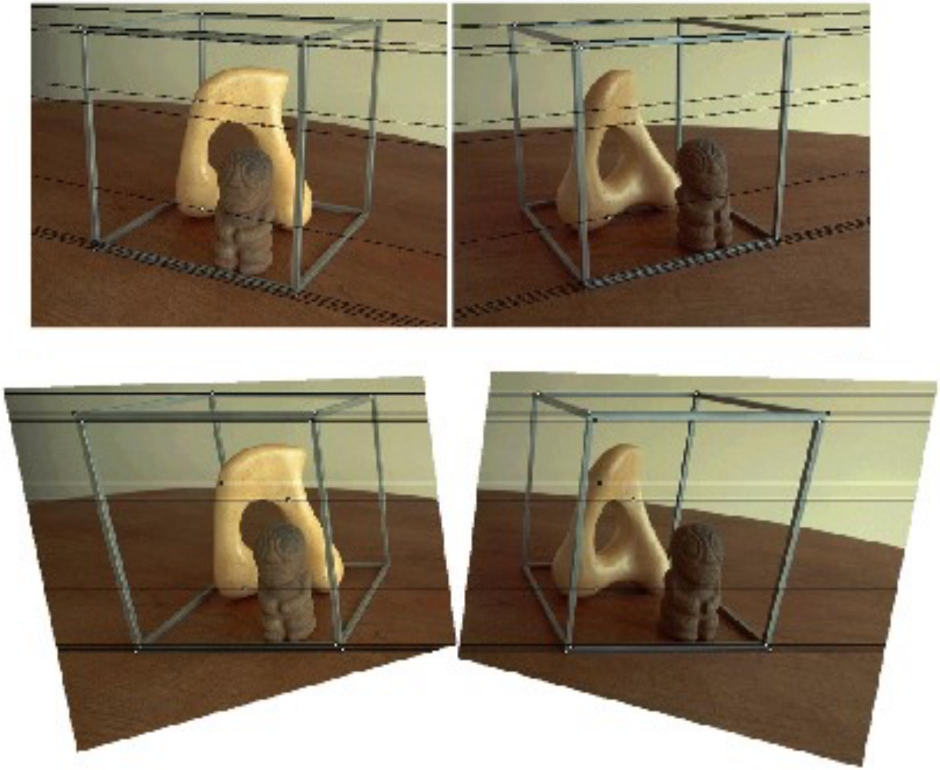
[Eq. 1]

Stereo pair



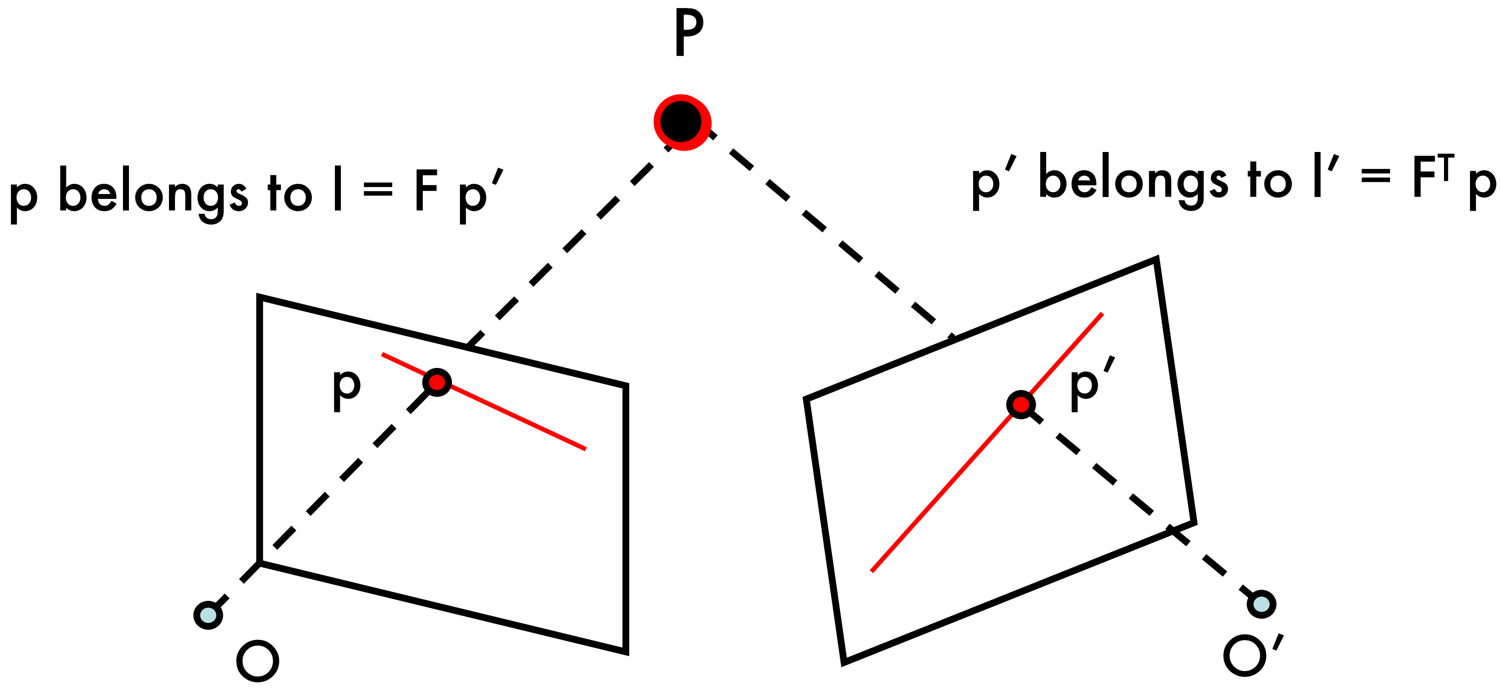
Disparity map / depth map

Why are parallel images useful?



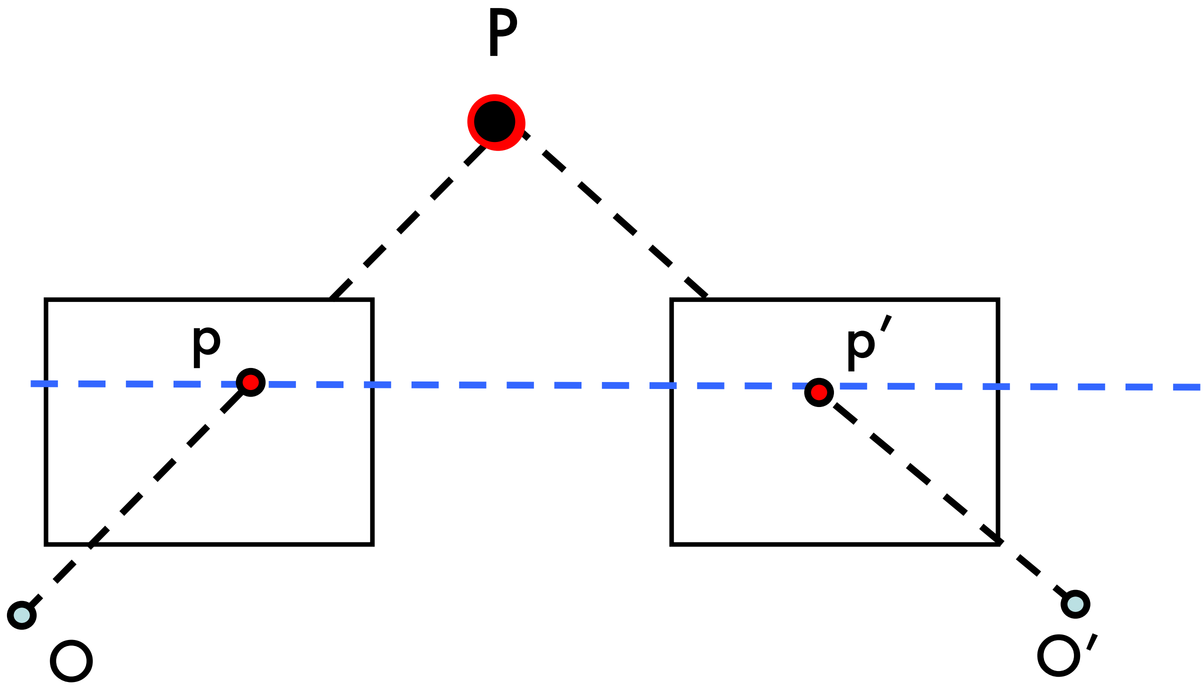
- Makes triangulation easy
- Makes the correspondence problem easier

Correspondence problem



Given a point in 3D, discover corresponding observations in left and right images [also called binocular fusion problem]

Correspondence problem



When images are rectified, this problem is much easier!

Correspondence problem

- A Cooperative Model (Marr and Poggio, 1976)
- Correlation Methods (1970–)
- Multi-Scale Edge Matching (Marr, Poggio and Grimson, 1979-81)

[FP] Chapters: 7

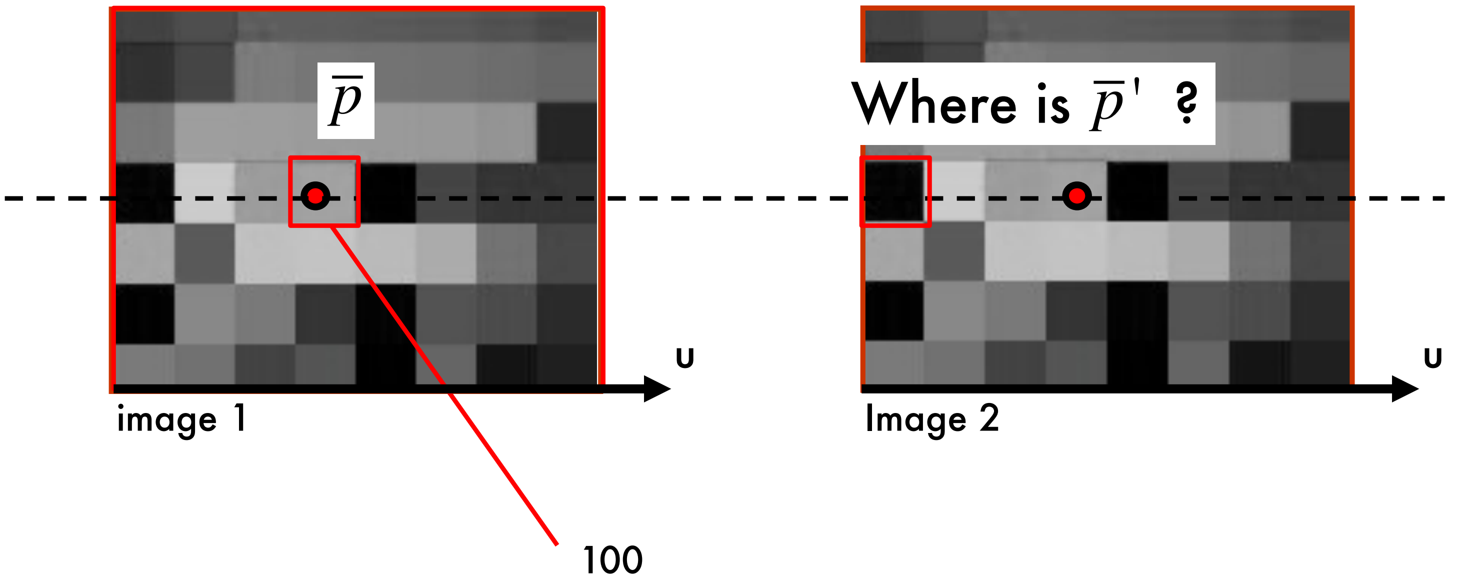
Correlation Methods (1970–)

image 1

Image 2

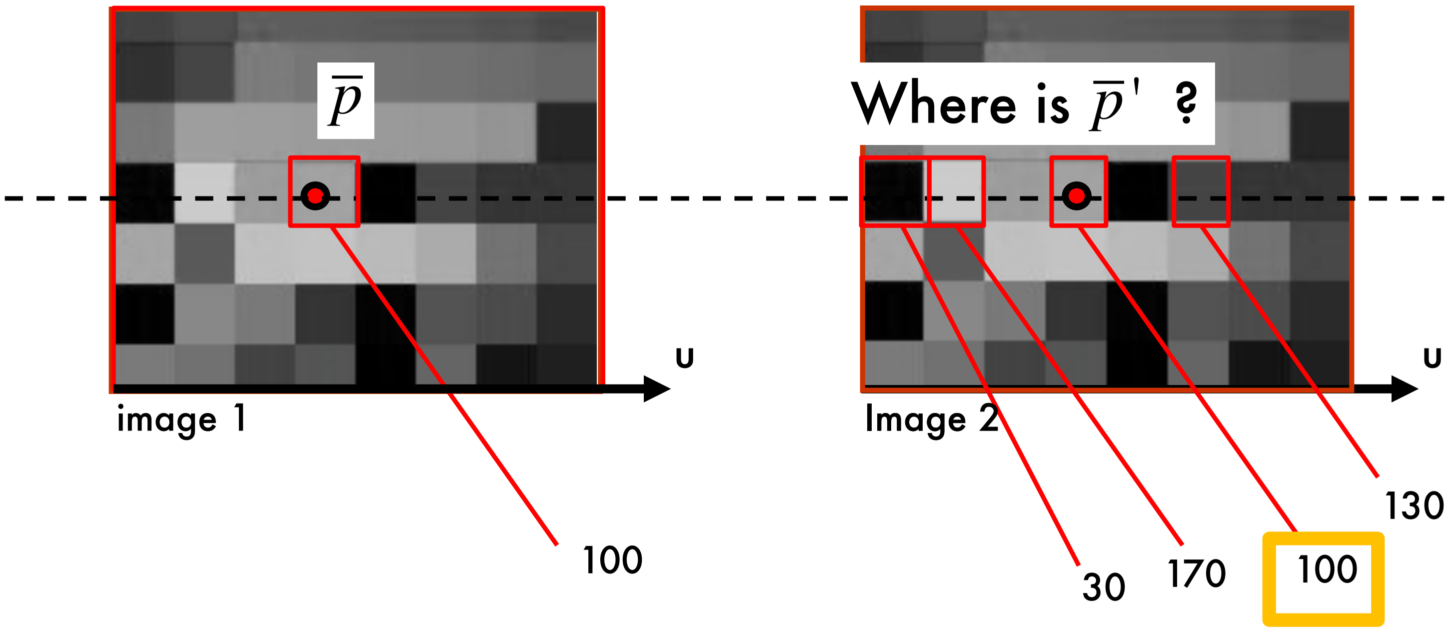
$$\bar{p} = \begin{bmatrix} \bar{u} \\ \bar{v} \\ 1 \end{bmatrix} \qquad \bar{p}' = \begin{bmatrix} \bar{u}' \\ \bar{v} \\ 1 \end{bmatrix}$$

Correlation Methods (1970–)



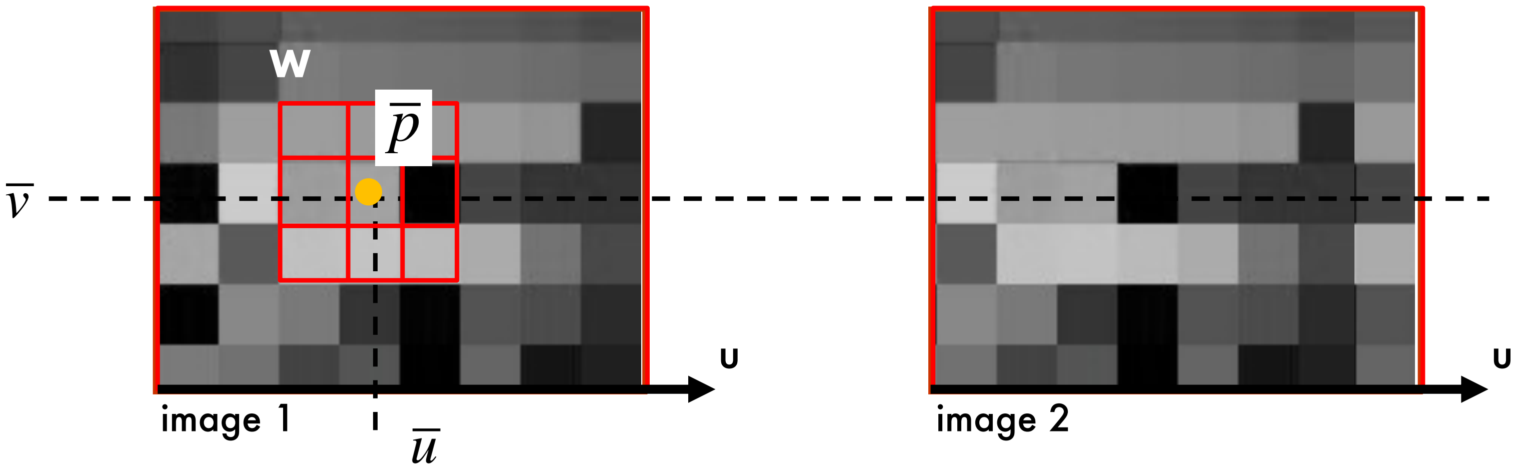
$$\overline{p} = \begin{bmatrix} \overline{u} \\ \overline{v} \\ 1 \end{bmatrix} \qquad \overline{p}' = \begin{bmatrix} \overline{u}' \\ \overline{v} \\ 1 \end{bmatrix}$$

Correlation Methods (1970–)



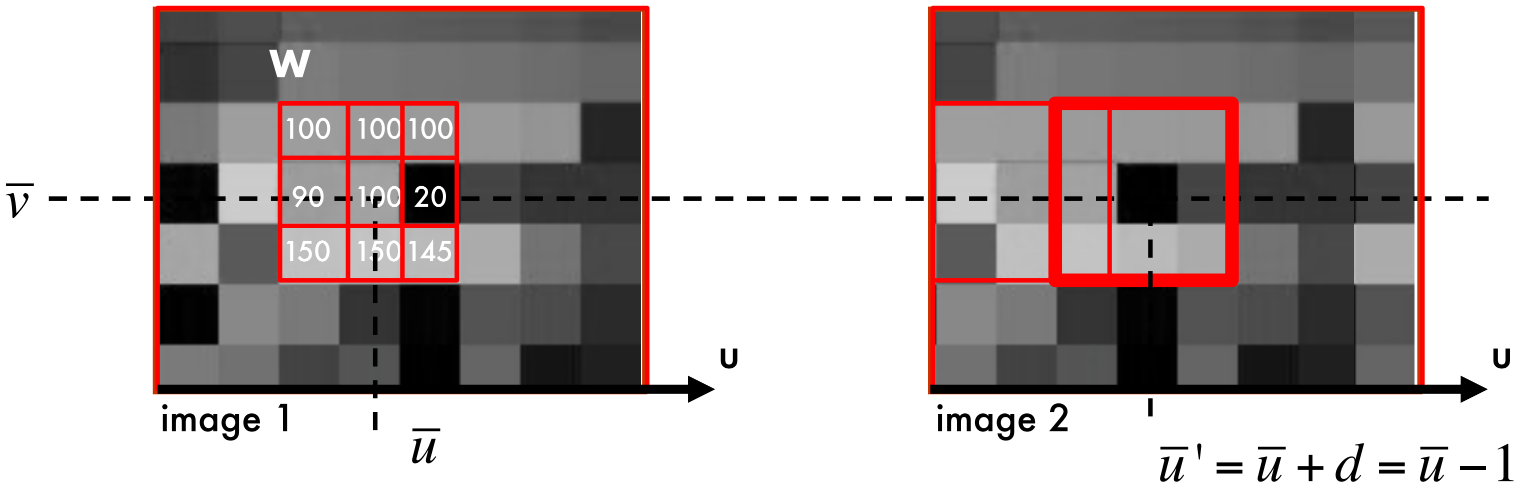
What's the problem with this? noise and photometric inconsistency

Window-based correlation



- Pick up a window **W** around $\overline{p} = (\overline{u}, \overline{v})$
- Build vector **w**

Window-based correlation



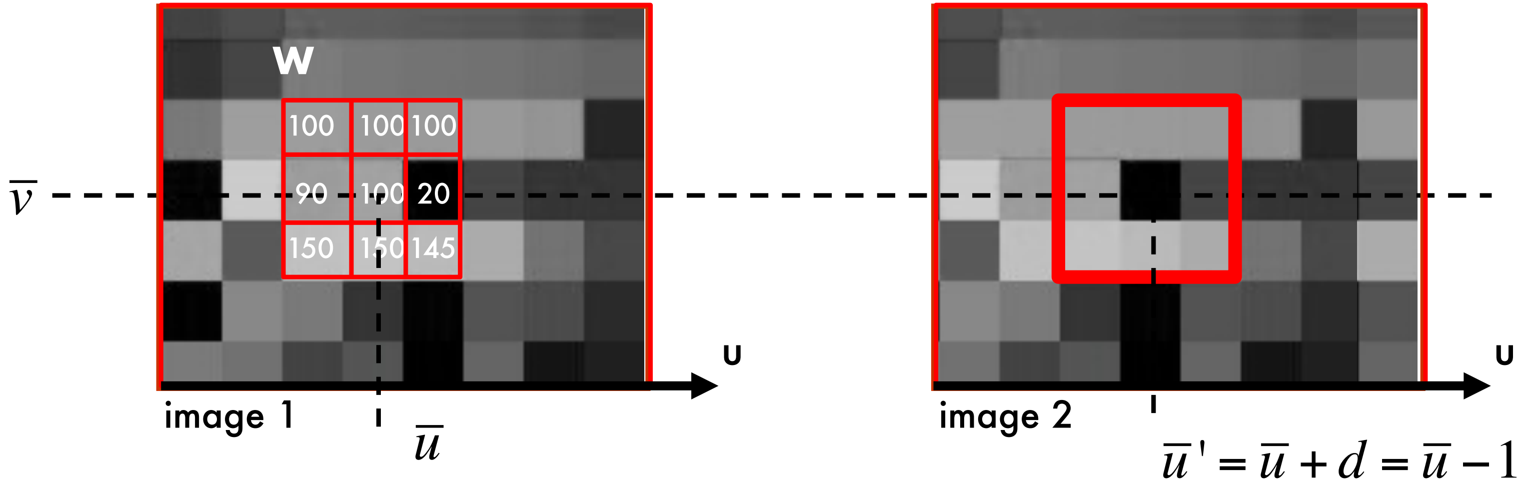
Example: **W** is a 3x3 window in red

w is a 9x1 vector

w = $[100, 100, 100, 90, 100, 20, 150, 150, 145]^T$

- Pick up a window **W** around $\bar{p} = (\bar{u}, \bar{v})$
- Build vector **w**
- Slide the window **W** along v = $\bar{v}$ in image 2 and compute **w'** (u) for each u
- Compute the dot product $\mathbf{w}^T$ **w'**(u) for each u and retain the max value

Window-based correlation



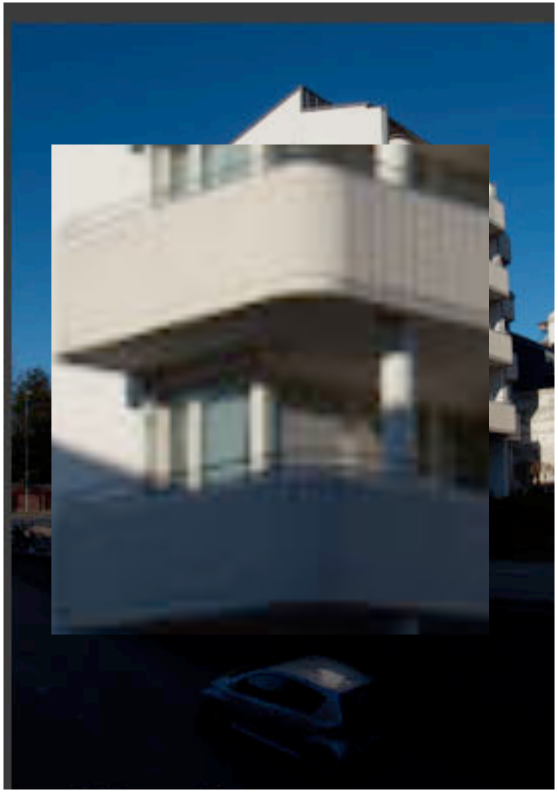
Example: **W** is a 3x3 window in red

w is a 9x1 vector

w = $[100, 100, 100, 90, 100, 20, 150, 150, 145]^T$

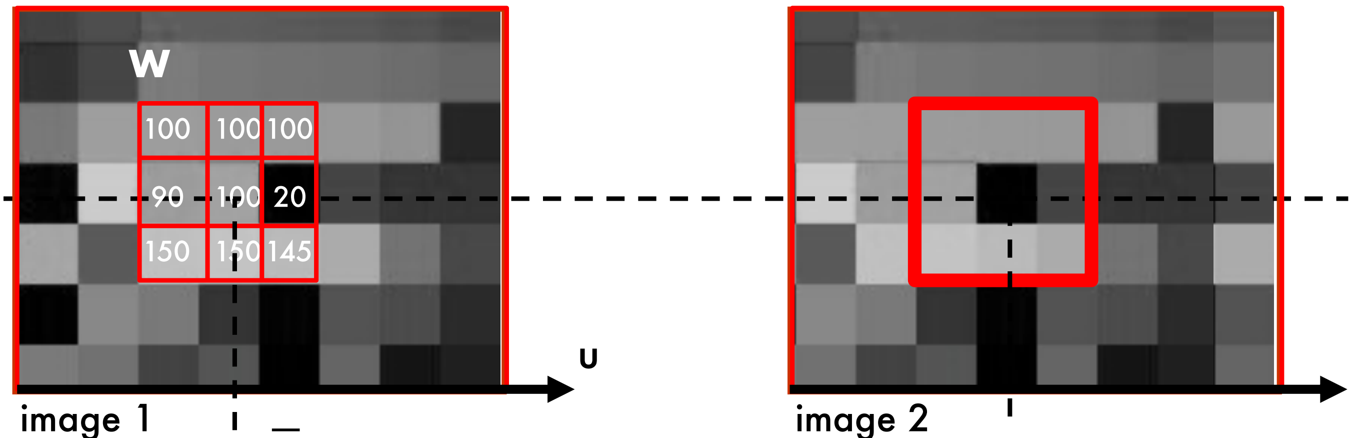
What's the problem with this?

Changes of brightness/exposure



Changes in the mean and the variance of intensity values in corresponding windows!

Normalized cross-correlation



$$\bar{v}$$

$$\bar{u}$$

$$\bar{u}' = \bar{u} + d = \bar{u} + 1$$

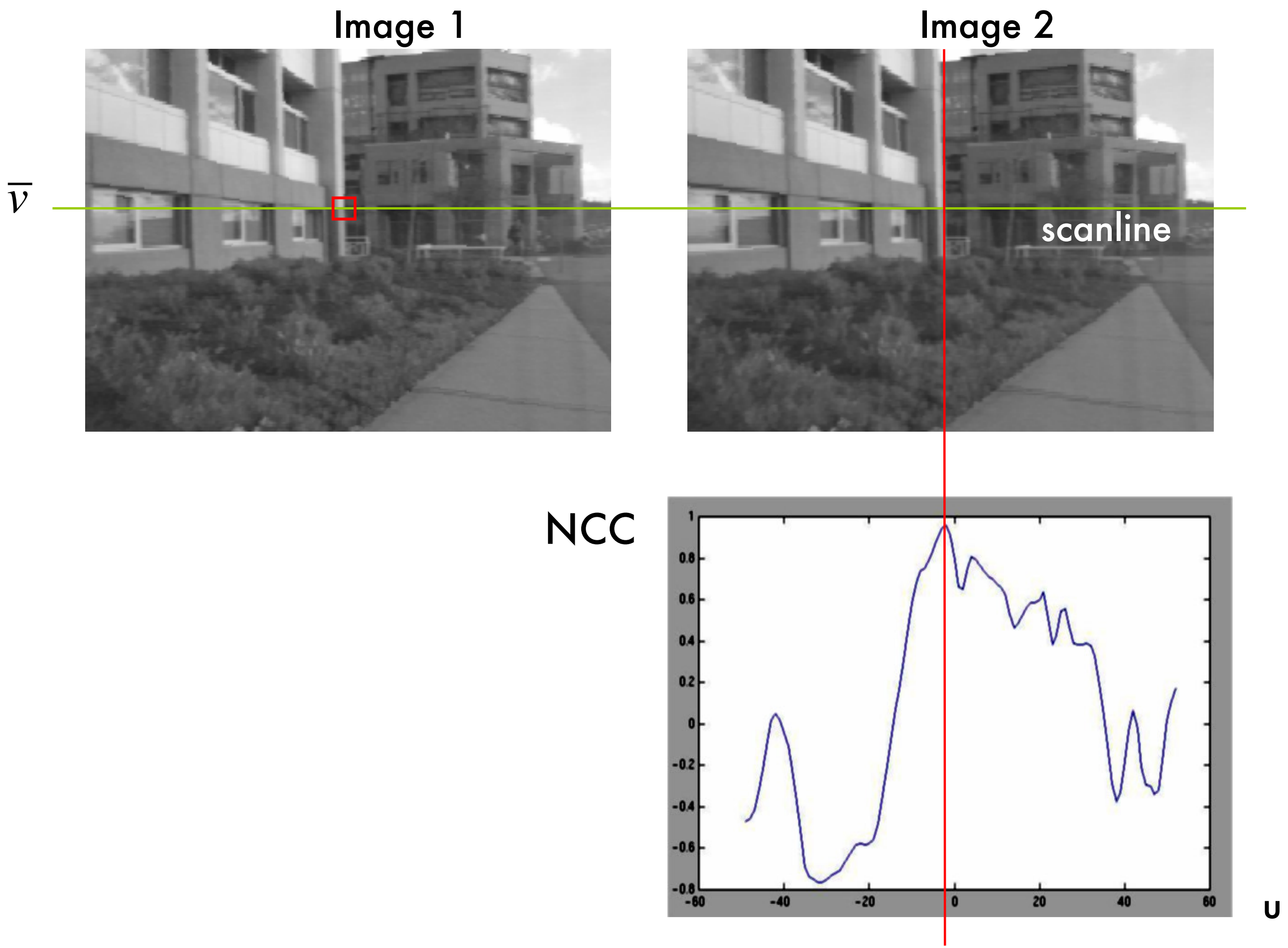
Find u that maximizes:

$$\frac{(w - \bar{w})^T (w'(u) - \bar{w}')}{\left\|(w - \bar{w})\right\| \left\|(w'(u) - \bar{w}')\right\|}$$ [Eq. 2]

$\bar{w} =$ mean value within **W** located at u^{bar} in image 1

$\bar{w}'(u) =$ mean value within **W** located at u in image 2

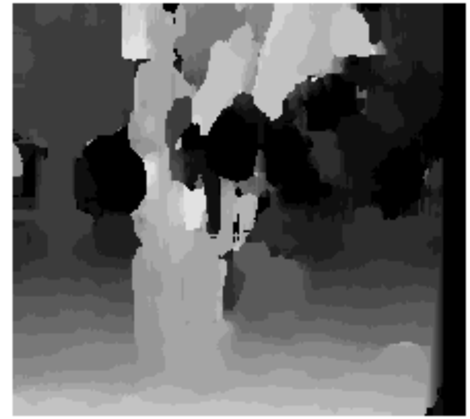
Example



Credit slide S. Lazebnik

Effect of the window's size

Window size = 3



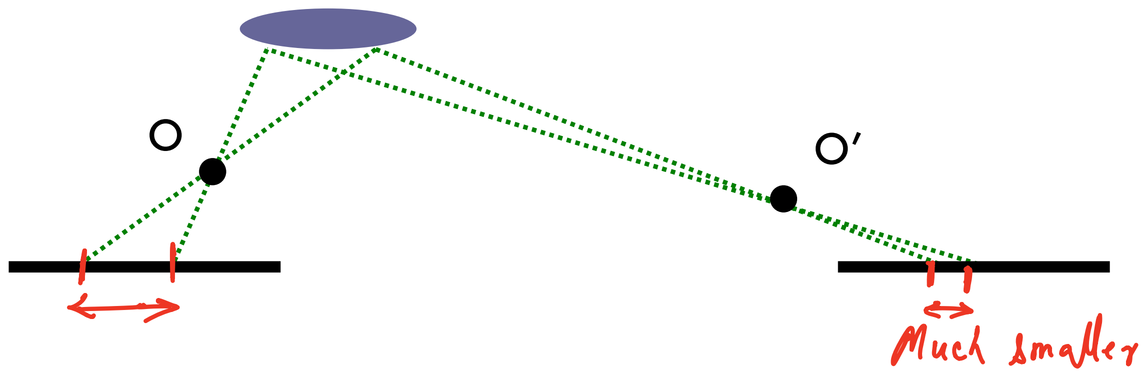
Window size = 20

- Smaller window
  - More detail
  - More noise
- Larger window
  - Smoother disparity maps
  - Less prone to noise

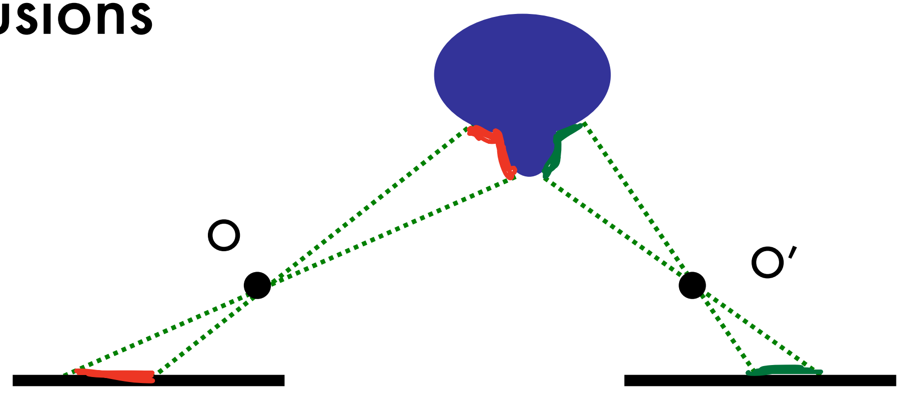
Credit slide S. Lazebnik

Issues

- Fore shortening effect



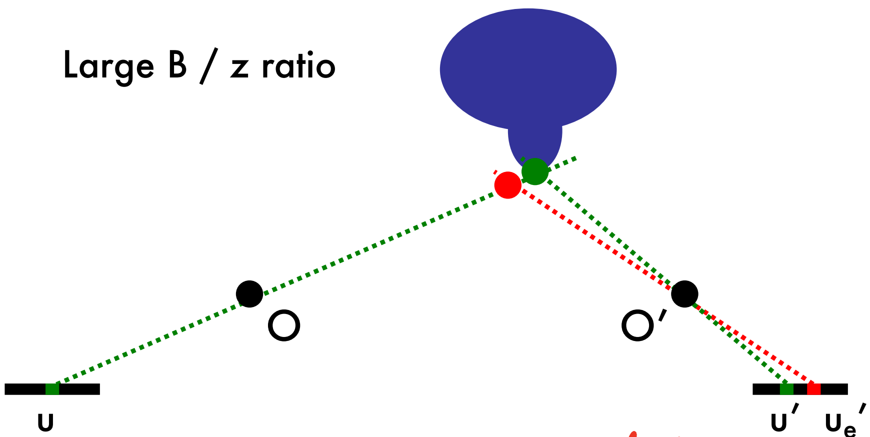
- Occlusions

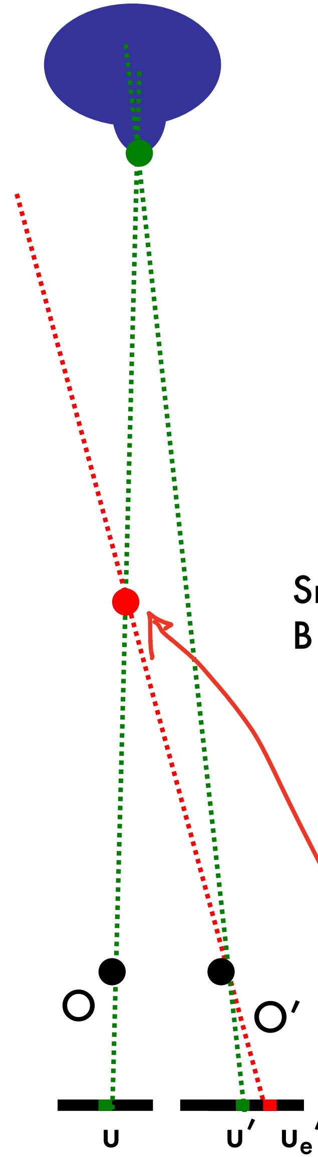


Issues

- To reduce the effect of foreshortening and occlusions, it is desirable to have small B / z ratio!
- However, when B/z is small, small errors in measurements imply large error in estimating depth

Large B / z ratio



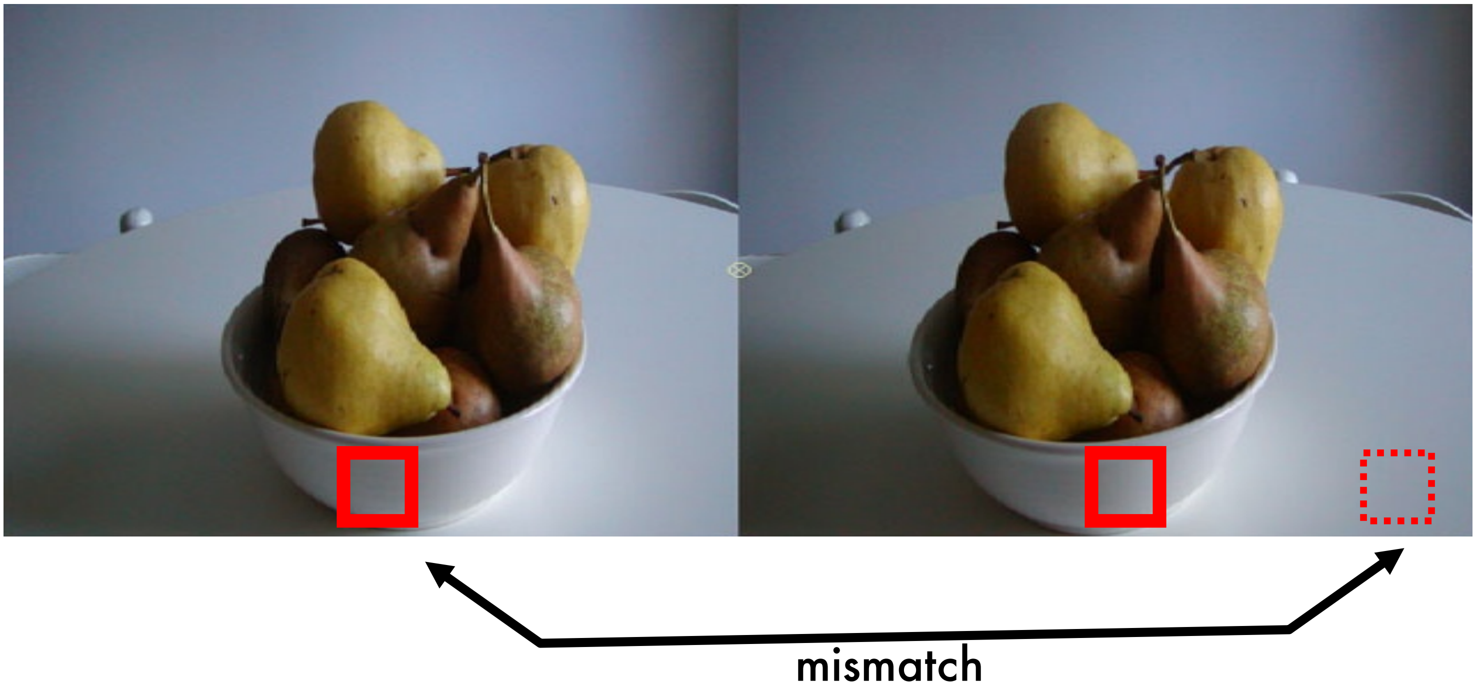


Small B / z ratio

($u' - u_e'$) → Same pixel error leads to a larger 'z' error

Issues

- Homogeneous regions



Issues

- Repetitive patterns

Correspondence problem is difficult!

- Occlusions
- Fore shortening
- Baseline trade-off
- Homogeneous regions
- Repetitive patterns

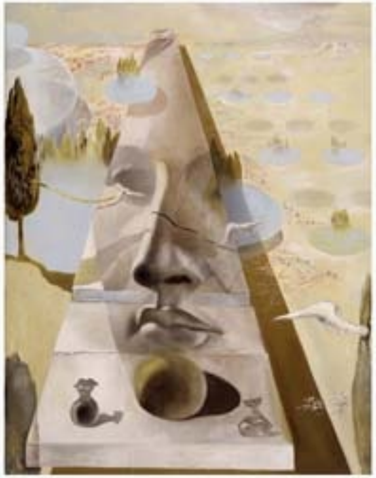
Apply non-local constraints to help enforce the correspondences

Non-local constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views
- Smoothness
  - Disparity is typically a smooth function of x (except in occluding boundaries)

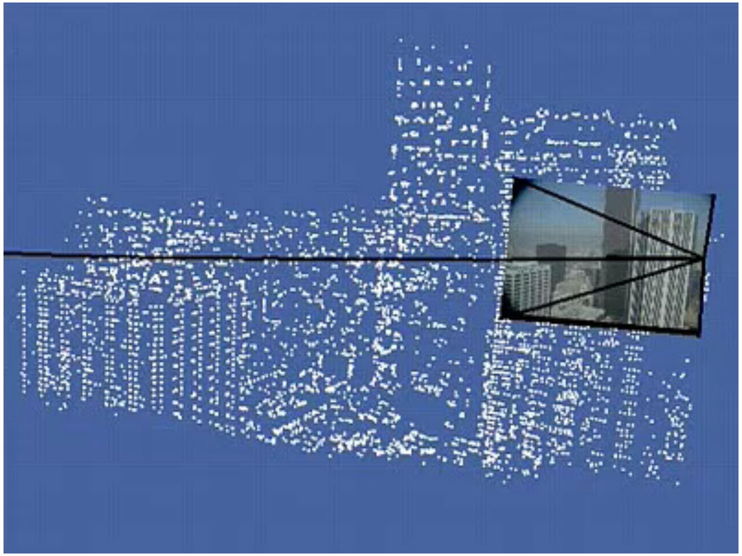
Lecture 6

Stereo Systems
Multi-view geometry



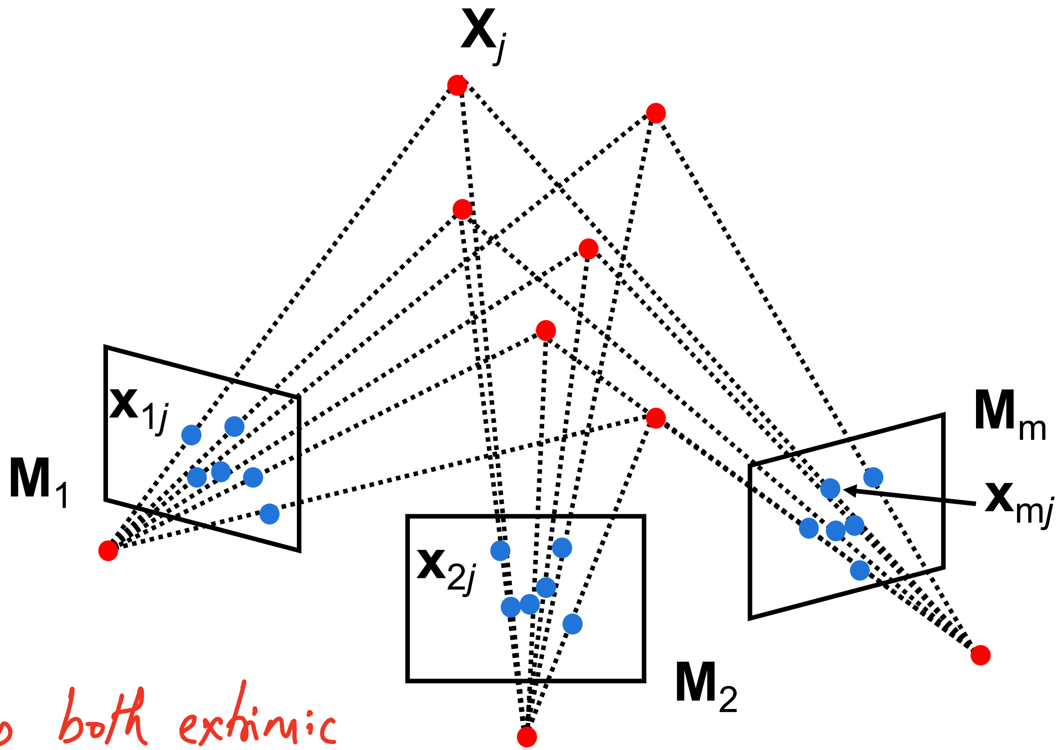
- Stereo systems
  - Rectification
  - Correspondence problem
- Multi-view geometry
  - The SFM problem
  - Affine SFM

Structure from motion problem



Courtesy of Oxford **Visual Geometry Group**

Structure from motion problem

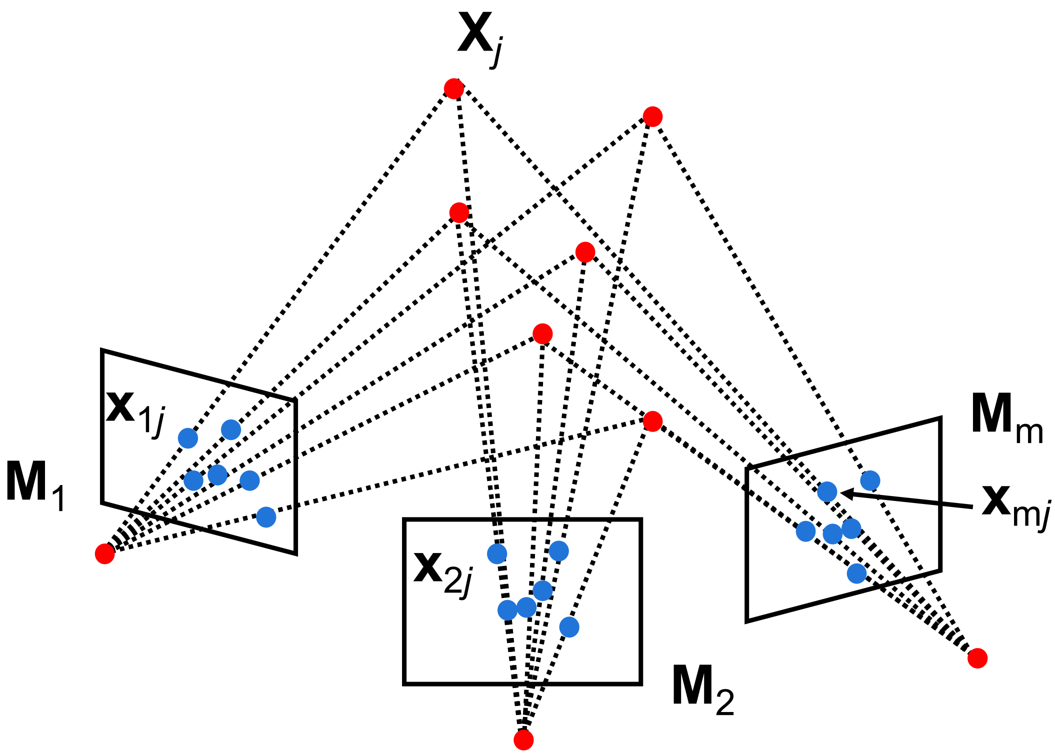


M_i encodes both extrinsic and intrinsic parameters.

Given *m* images of *n* fixed 3D points

• $\mathbf{x}_{ij} = \mathbf{M}_i \mathbf{X}_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n$

Structure from motion problem

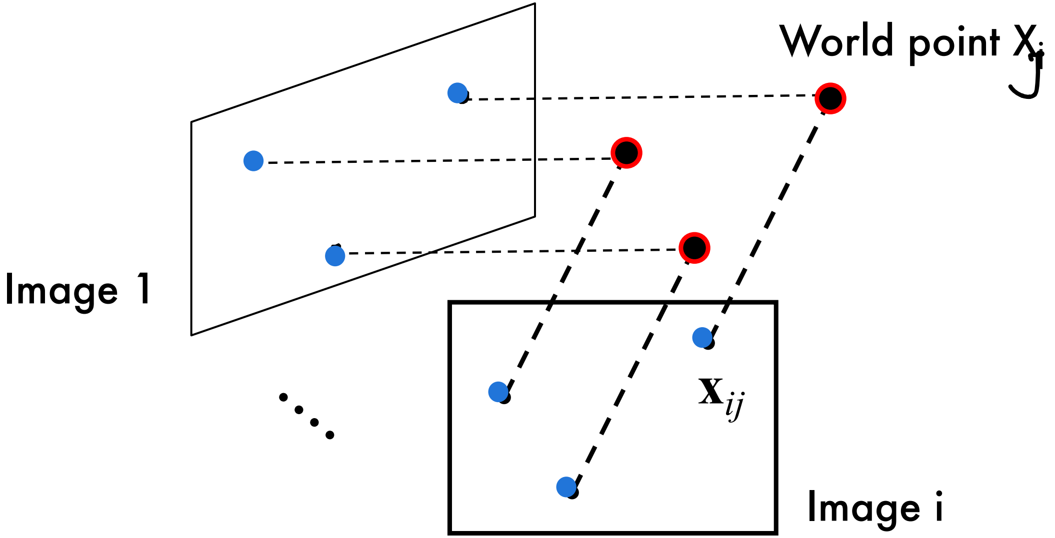


From the m×n observations $\mathbf{x}_{ij}$, estimate:

- *m* projection matrices $\mathbf{M}_i$ — motion
- *n* 3D points $\mathbf{X}_j$ — structure

Affine structure from motion

(simpler problem)



From the m×n observations $\mathbf{x}_{ij}$, estimate:

- *m* projection matrices $\mathbf{M}_i$ (affine cameras)
- *n* 3D points $\mathbf{X}_j$

Perspective

$$\mathbf{x} = M\,\mathbf{X} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} \mathbf{X} = \begin{bmatrix} \mathbf{m}_1\mathbf{X} \\ \mathbf{m}_2\mathbf{X} \\ \mathbf{m}_3\mathbf{X} \end{bmatrix}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{v} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

$$\mathbf{x}^E = \left(\frac{\mathbf{m}_1\mathbf{X}}{\mathbf{m}_3\mathbf{X}}, \frac{\mathbf{m}_2\mathbf{X}}{\mathbf{m}_3\mathbf{X}}\right)^T$$

Affine

$$\mathbf{x} = M\,\mathbf{X} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} \mathbf{X} = \begin{bmatrix} \mathbf{m}_1\mathbf{X} \\ \mathbf{m}_2\mathbf{X} \\ 1 \end{bmatrix}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{2x3} & \mathbf{b}_{2x1} \\ \mathbf{0}_{1x3} & 1 \end{bmatrix} = \begin{bmatrix} & \mathbf{m}_1 & & \\ & \mathbf{m}_2 & & \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

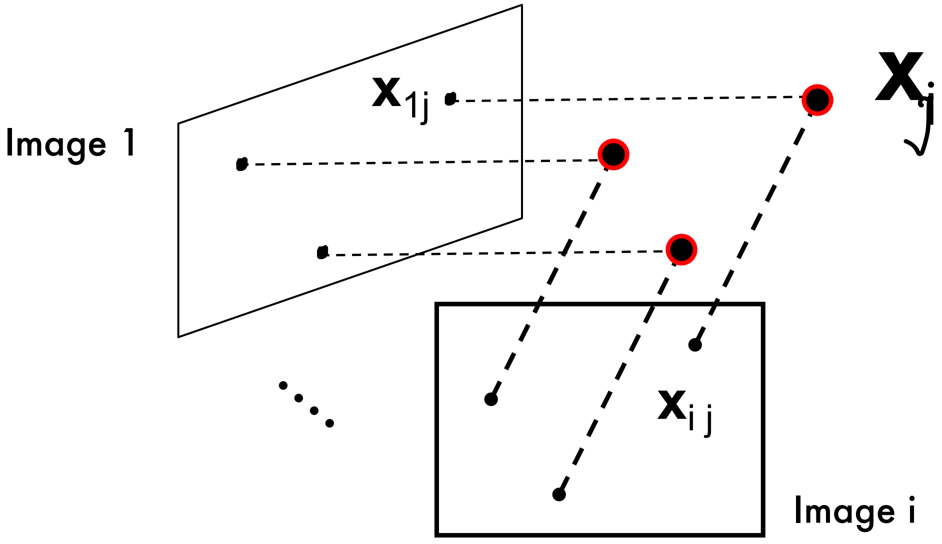
$$\mathbf{x}^E = (\mathbf{m}_1\mathbf{X}, \mathbf{m}_2\mathbf{X})^T = \begin{bmatrix} \mathbf{A} & \mathbf{b} \end{bmatrix}\mathbf{X} = \begin{bmatrix} \underset{3x3}{\mathbf{A}} & \underset{3x1}{\mathbf{b}} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \underline{\underline{\mathbf{A}\mathbf{X}^E + \mathbf{b}}}$$

$$\mathbf{X}^E = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

magnification

[Eq. 3]

Affine cameras



For the affine case (in Euclidean space)

$$\mathbf{x}_{ij} = \mathbf{A}_i \mathbf{X}_j + \mathbf{b}_i \qquad \text{[Eq. 4]}$$

2x1 2x3 3x1 2x1

The Affine Structure-from-Motion Problem

Given *m* images of *n* fixed points $\mathbf{X}_j$ we can write

$$\mathbf{x}_{ij} = \mathbf{A}_i \mathbf{X}_j + \mathbf{b}_i$$

for i = 1, ..., $\boxed{m}$ and j = 1, ... , $\boxed{n}$

N. of cameras N. of points

Problem: estimate m matrices A_i, m matrices b_i and the n positions $\mathbf{X}_j$ from the m×n observations $\mathbf{x}_{ij}$.

How many equations and how many unknown?

2m × n equations in 8m + 3n - 8 unknowns

The Affine Structure-from-Motion Problem

Two approaches:

- Algebraic approach (affine epipolar geometry; estimate F; cameras; points)

- Factorization method

Next lecture

Multiple view geometry:
Affine and Perspective structure from Motion