Lecture 6

Stereo Systems
Multi-view geometry

Professor Silvio Savarese

Computational Vision and Geometry Lab

Lecture 6

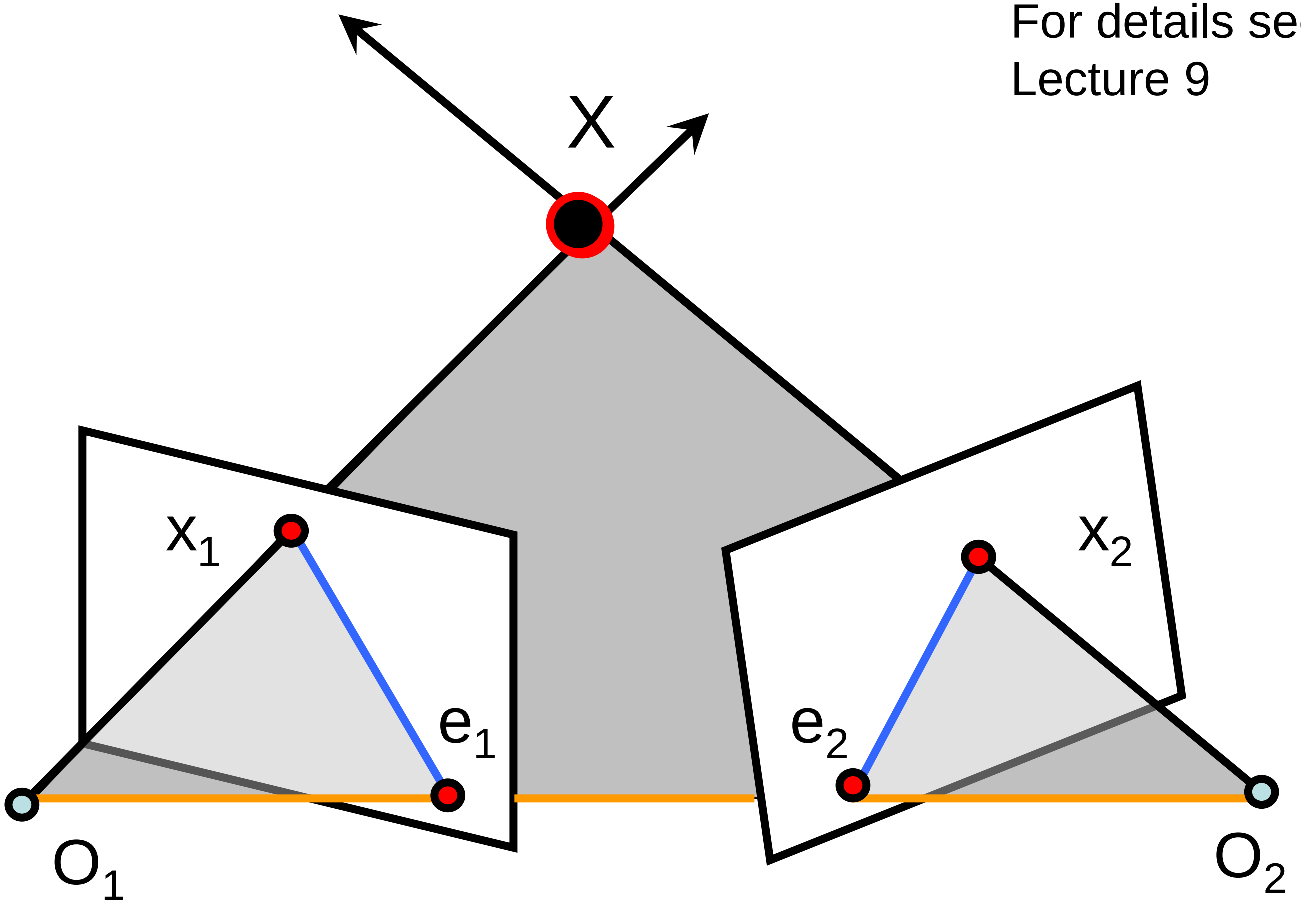
Stereo Systems
Multi-view geometry

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

Reading: [AZ] Chapter: 4 “Estimation – 2D perspective transformations
Chapter: 9 “Epipolar Geometry and the Fundamental Matrix T
Chapter: 11 “Computation of the Fundamental Matrix F”
[FP] Chapters: 10 “The geometry of multiple views”

Epipolar geometry

For details see CS131A Lecture 9

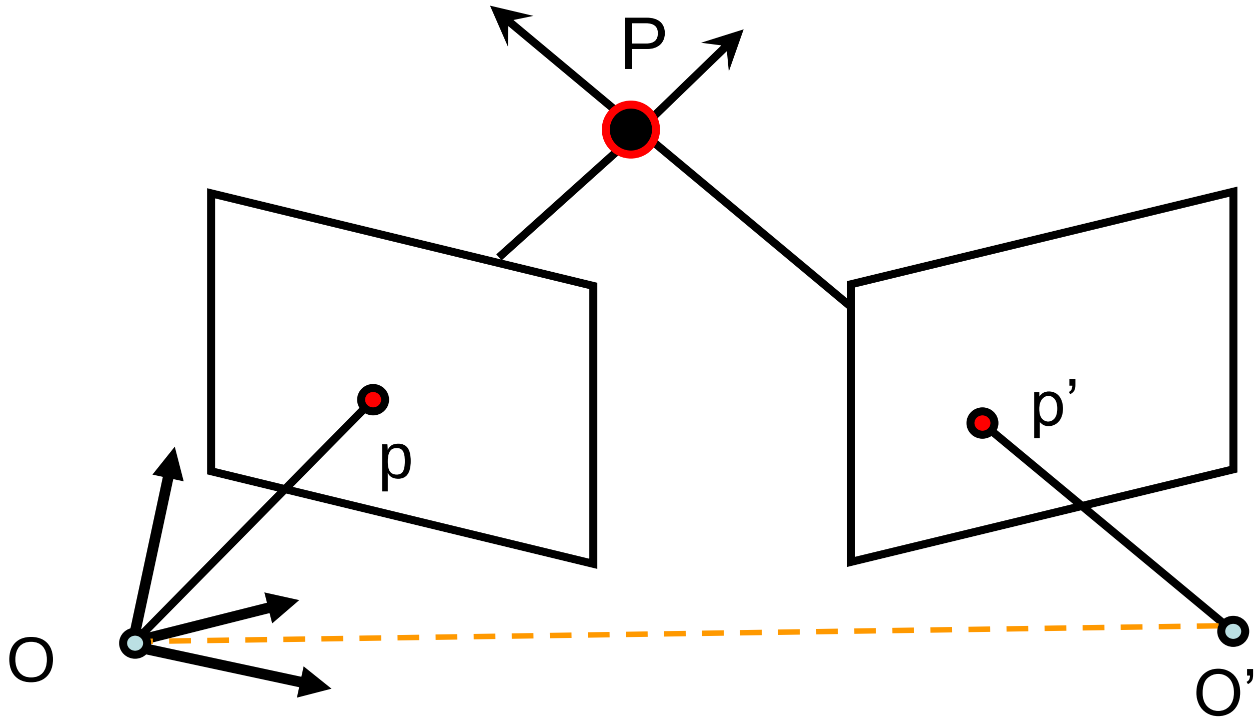


- Epipolar Plane
- Baseline
- Epipolar Lines

- Epipoles e_1, e_2

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

Epipolar Constraint



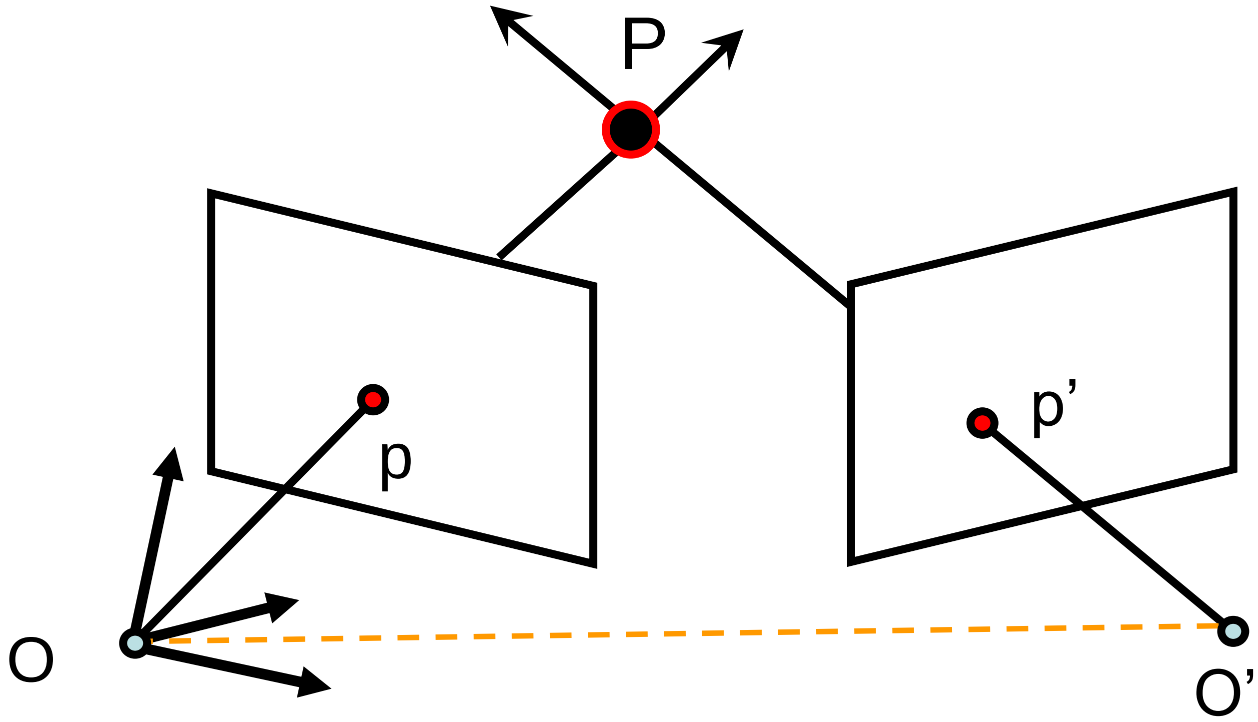
$$p^T E p' = 0$$

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

E = Essential Matrix

(Longuet-Higgins, 1981)

Epipolar Constraint



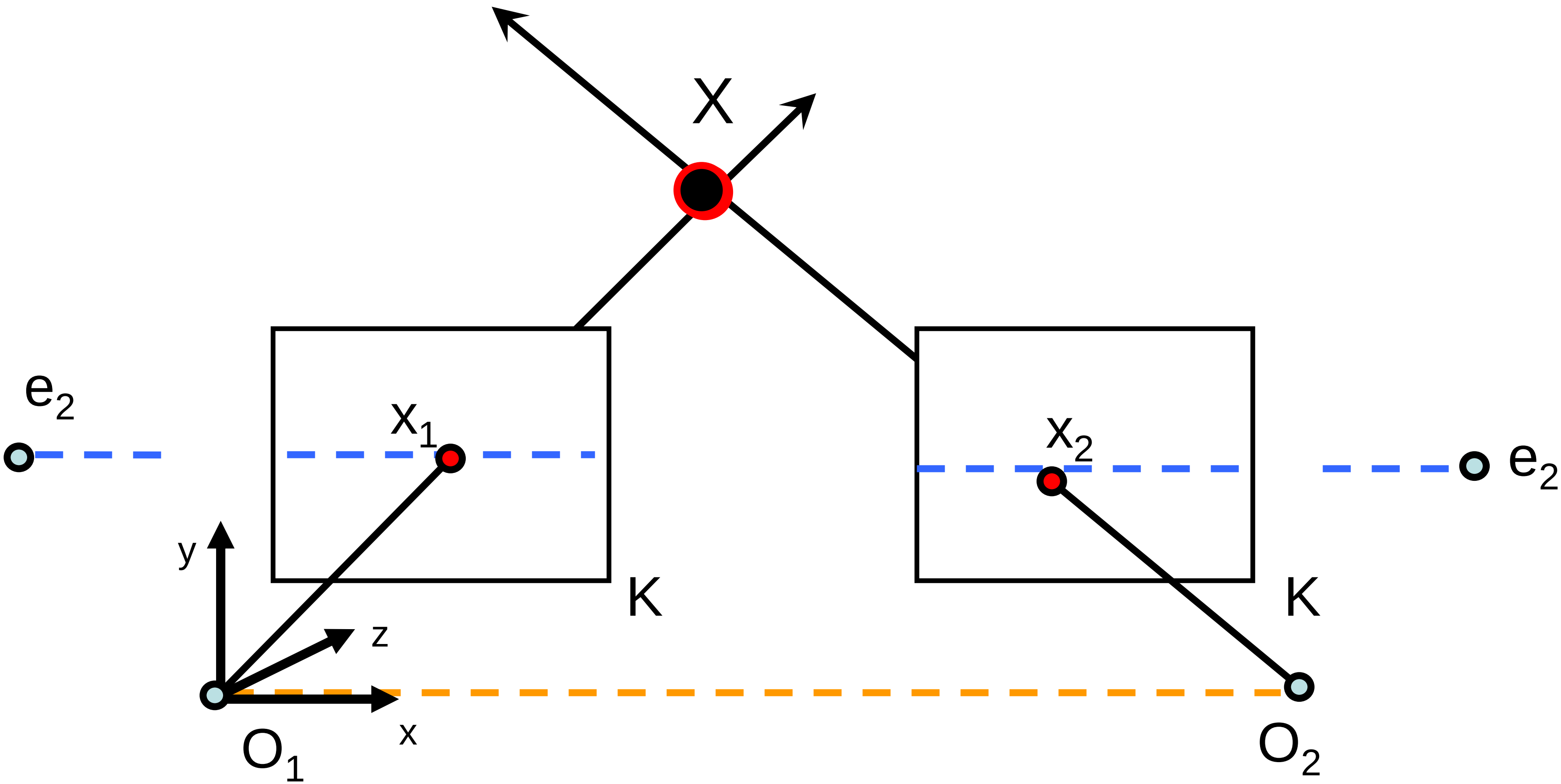
$$p^T F p' = 0$$

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

F = Fundamental Matrix

(Faugeras and Luong, 1992)

Example: Parallel image planes

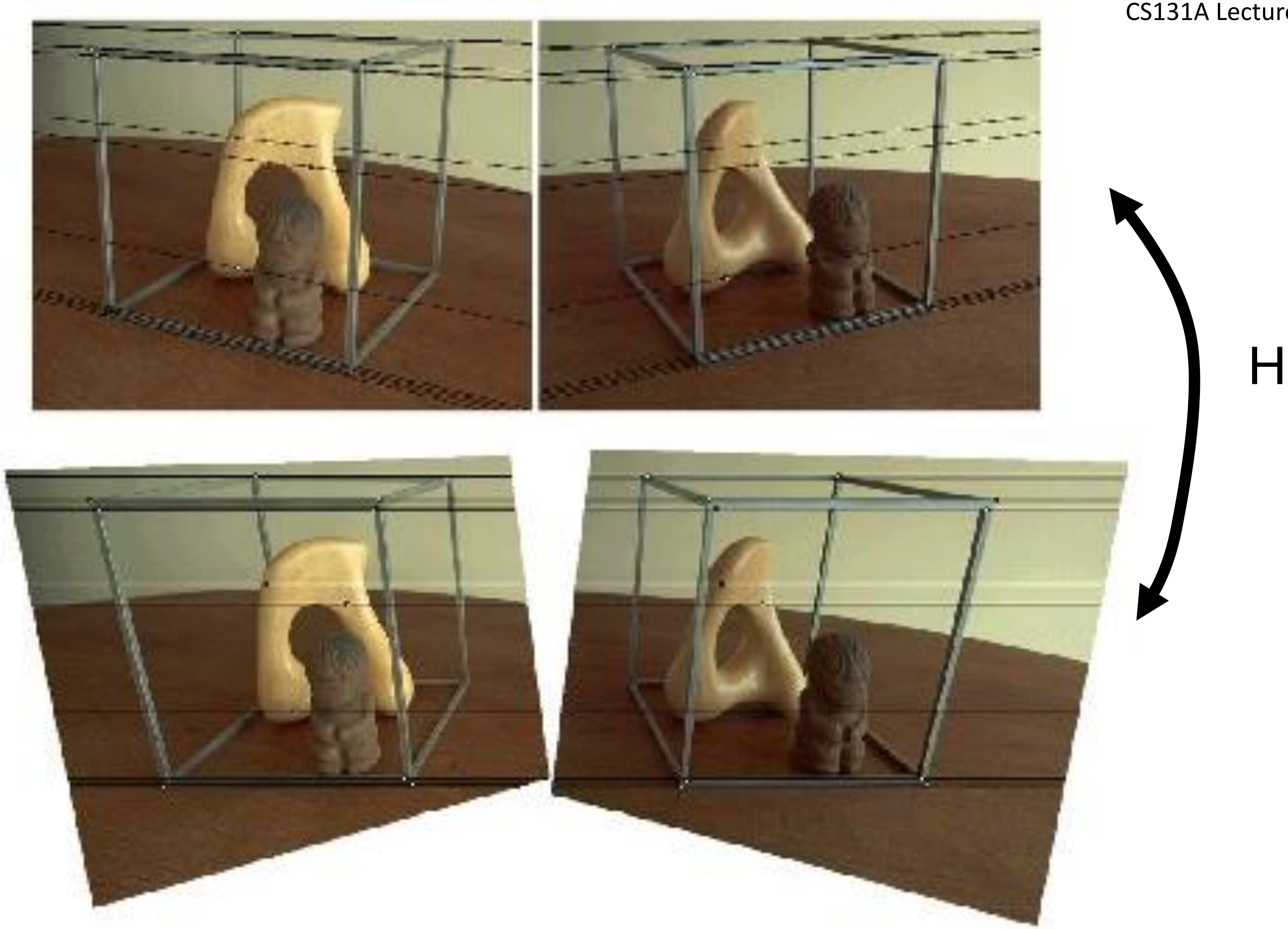


- Epipolar lines are horizontal
- Epipoles go to infinity
- y-coordinates are equal

$$\mathbf{x}_1 = \begin{bmatrix} x_1 \\ y \\ 1 \end{bmatrix} \qquad \mathbf{x}_2 = \begin{bmatrix} x_2 \\ y \\ 1 \end{bmatrix}$$

Rectification: making two images "parallel"

For details on how to rectify two views see CS131A Lecture 9



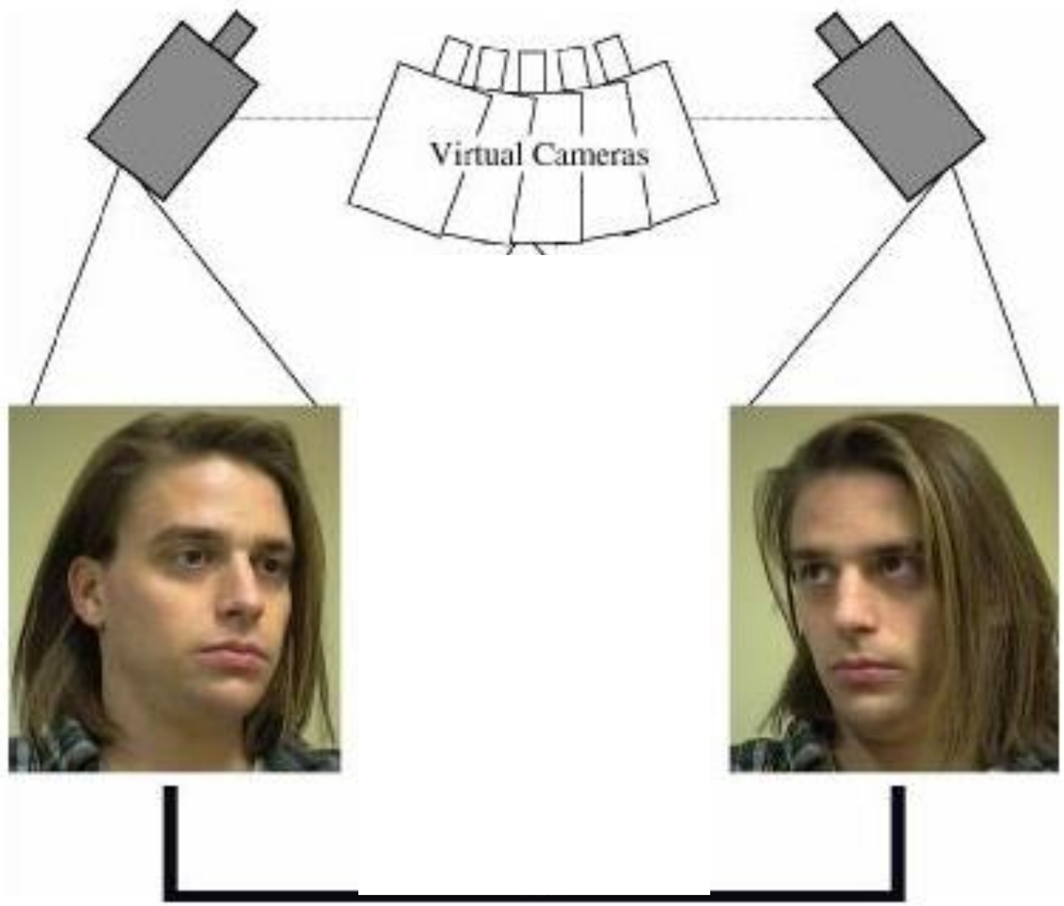
Courtesy figure S. Lazebnik

Why are parallel images useful?

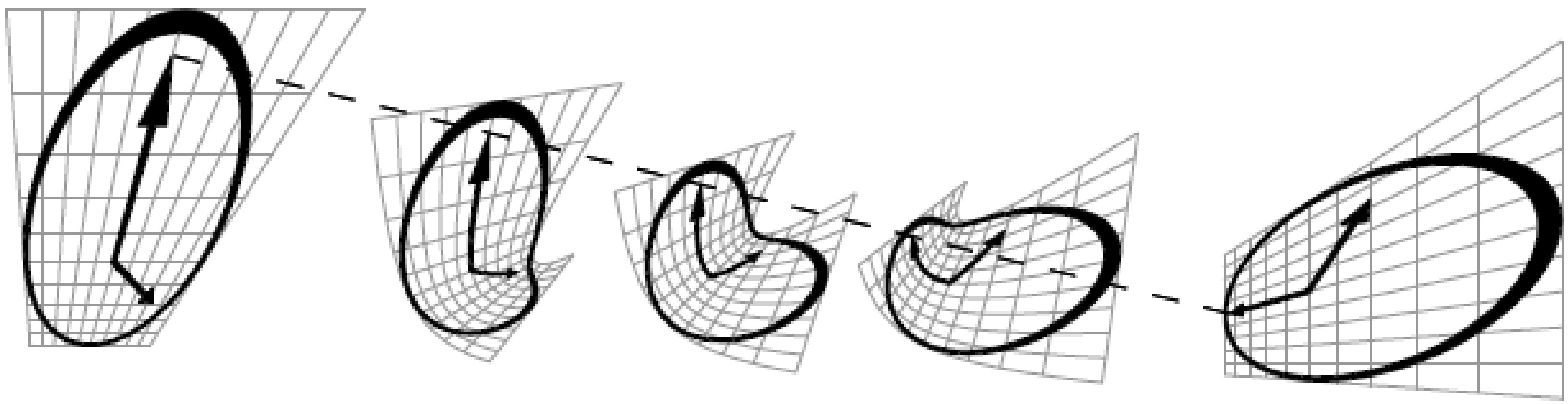
- Makes the correspondence problem easier
- Makes triangulation easy
- Enables schemes for image interpolation

Application: view morphing

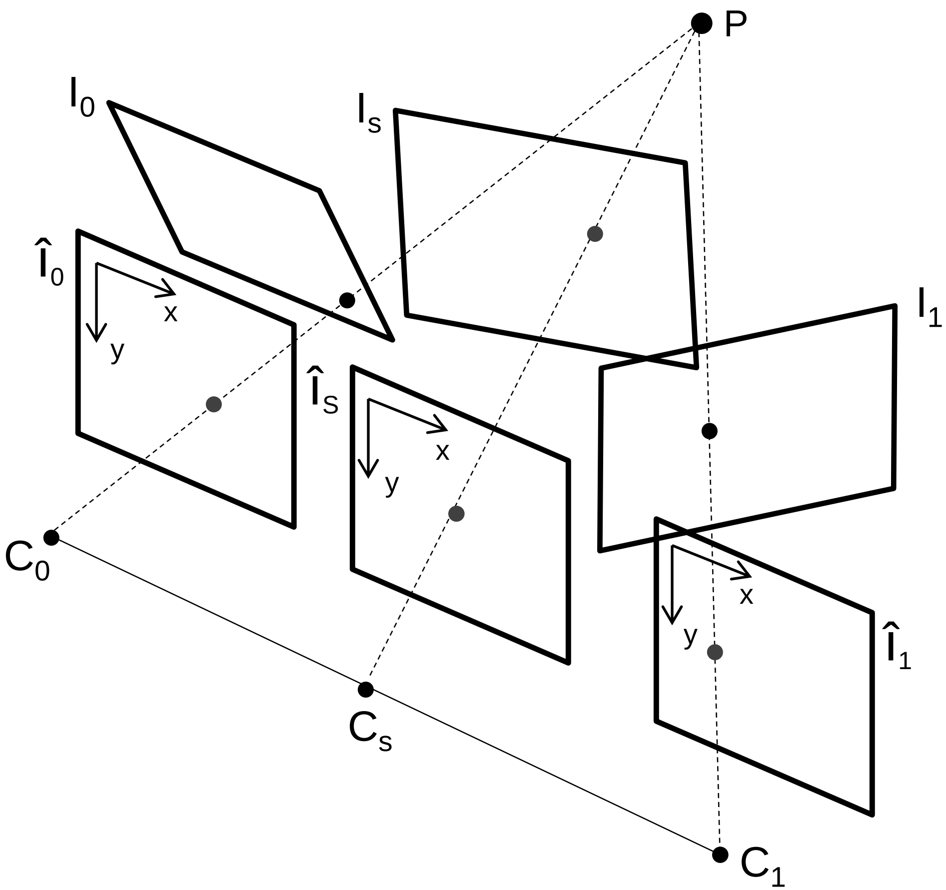
S. M. Seitz and C. R. Dyer, *Proc. SIGGRAPH 96*, 1996, 21-30



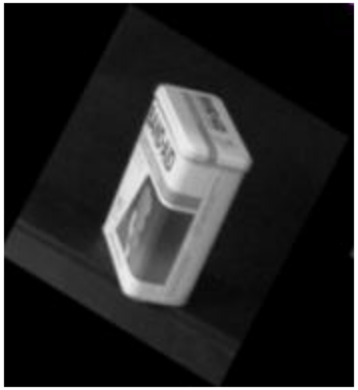
Morphing without using geometry

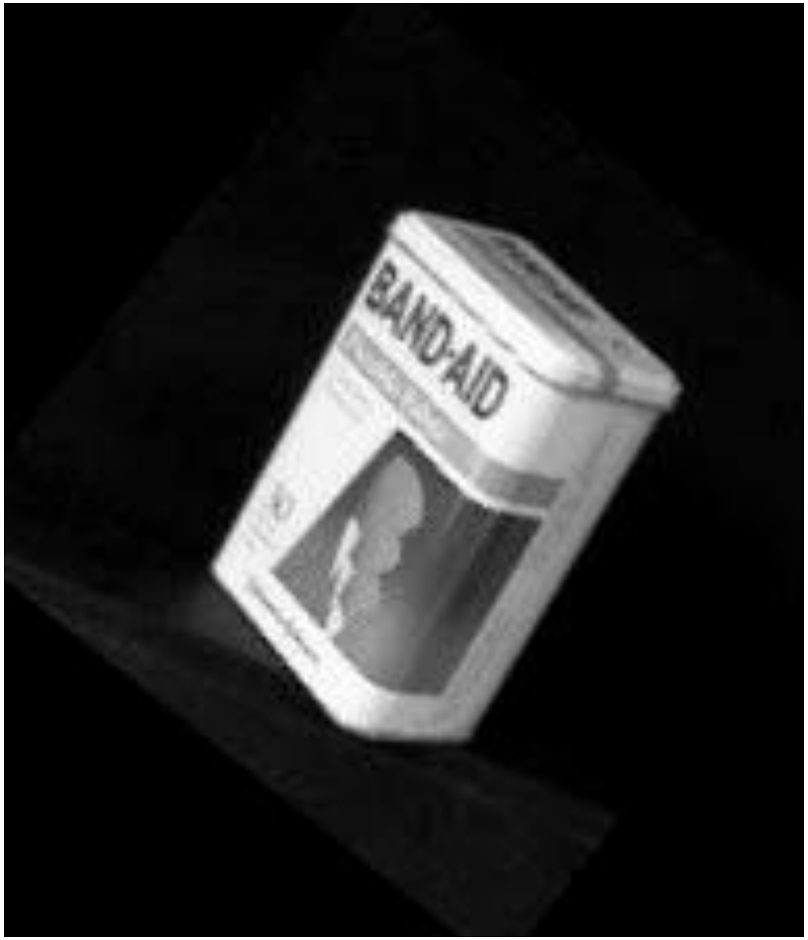


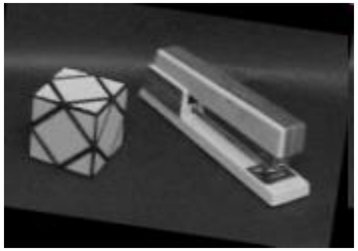
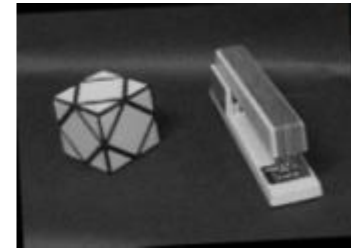
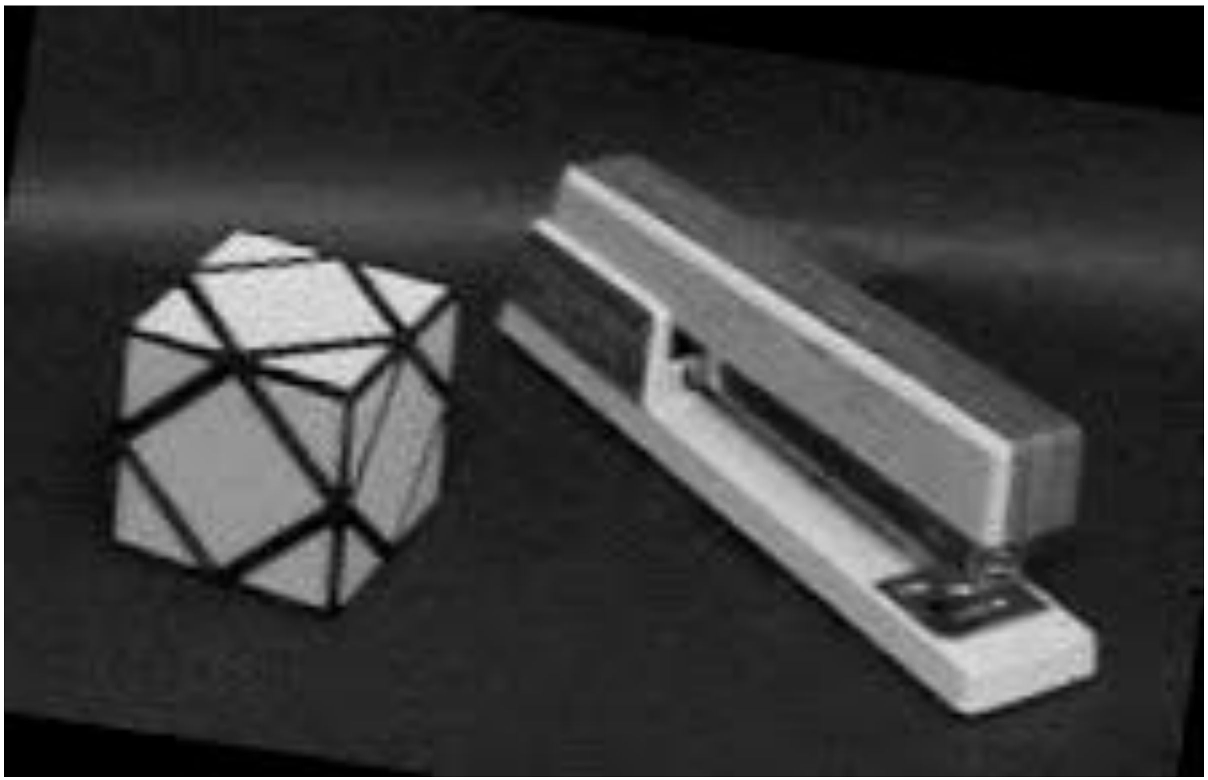
Rectification




BAND-AID

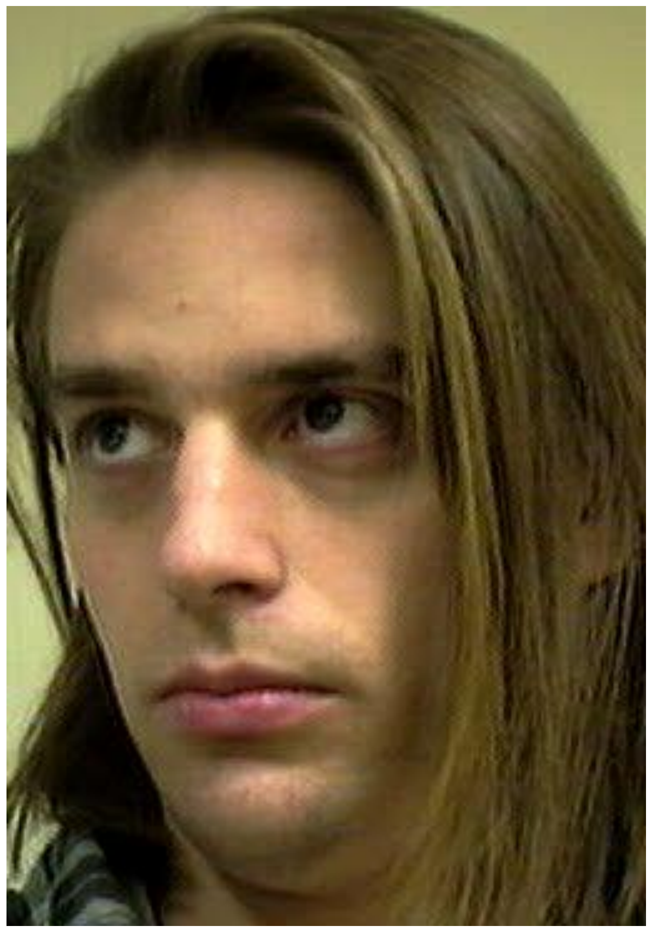


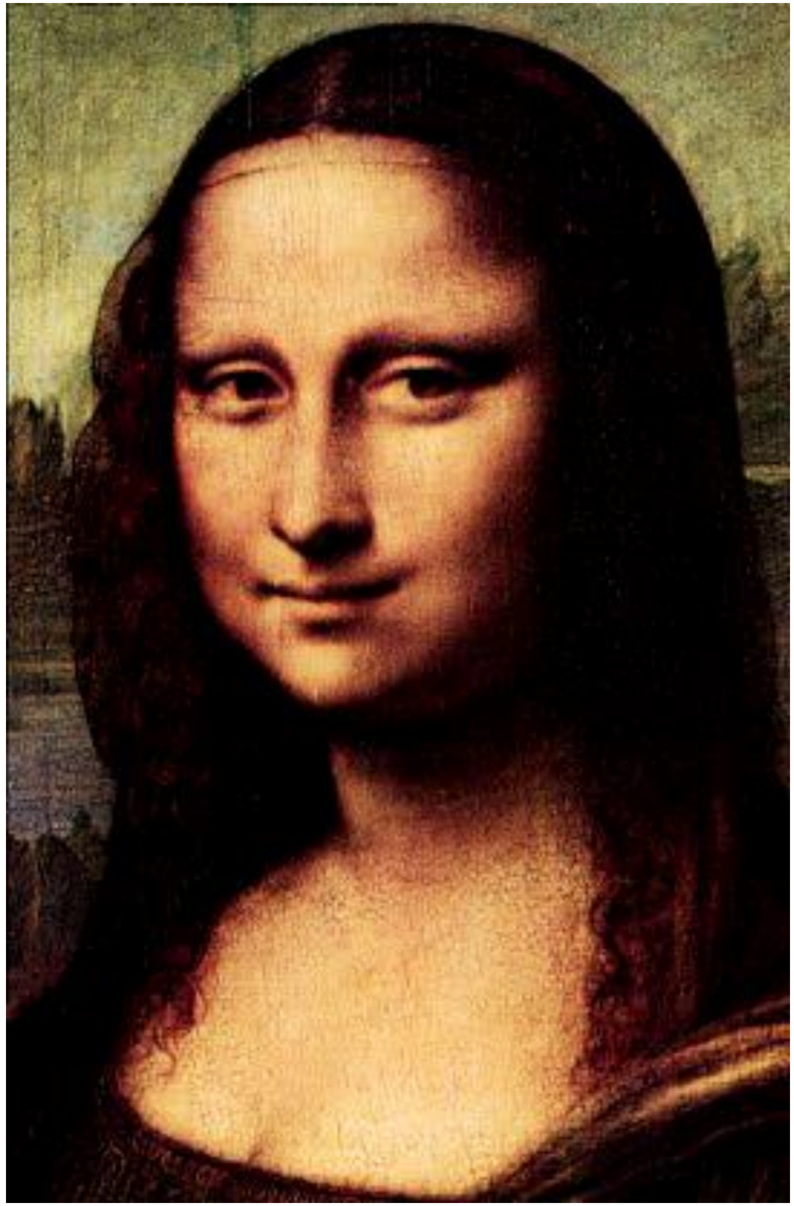

BAND-AID





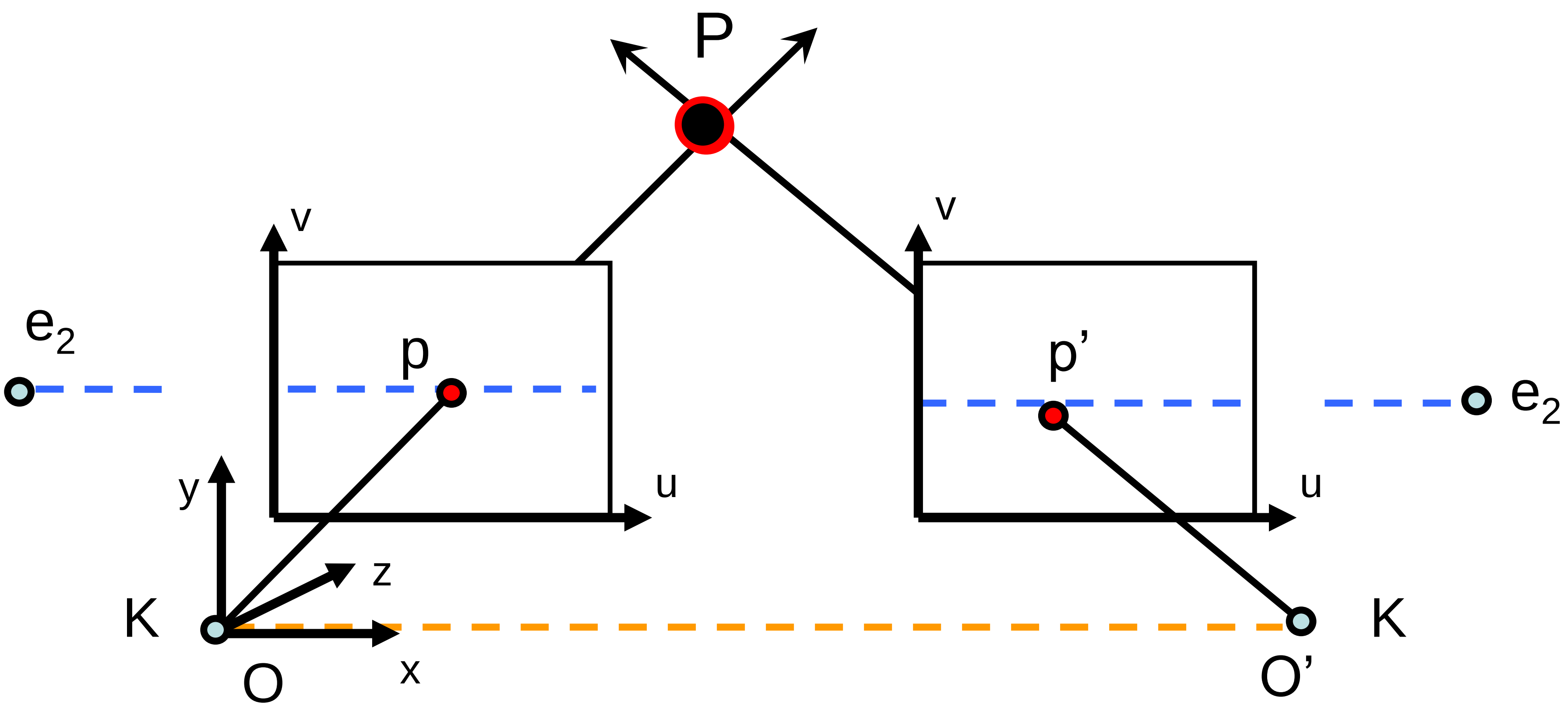


From its reflection!

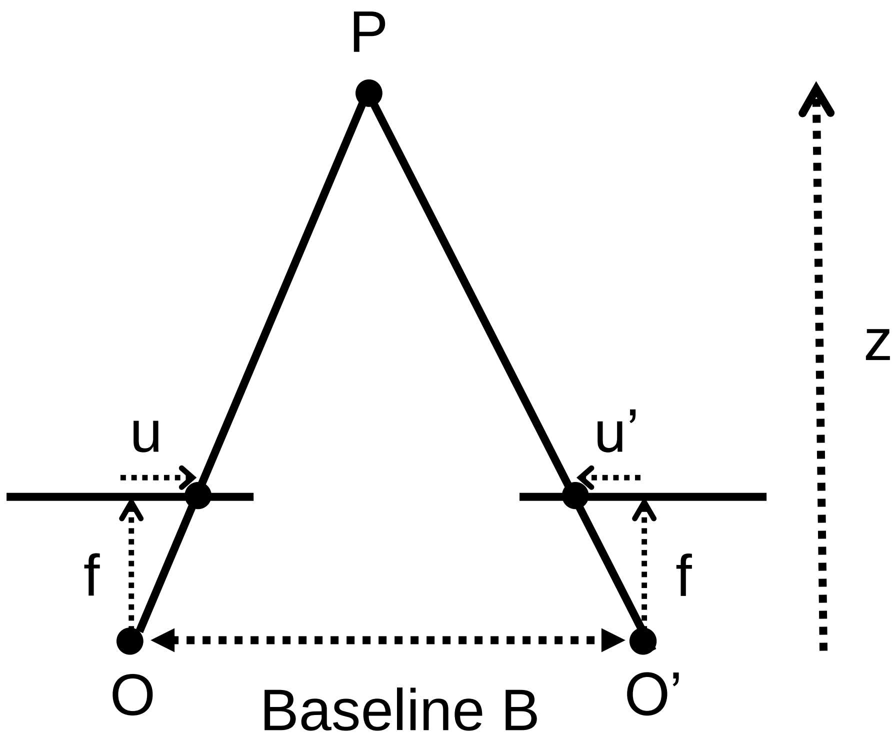
Why are parallel images useful?

- Makes the correspondence problem easier
- Makes triangulation easy
- Enables schemes for image interpolation

Point triangulation



Computing depth



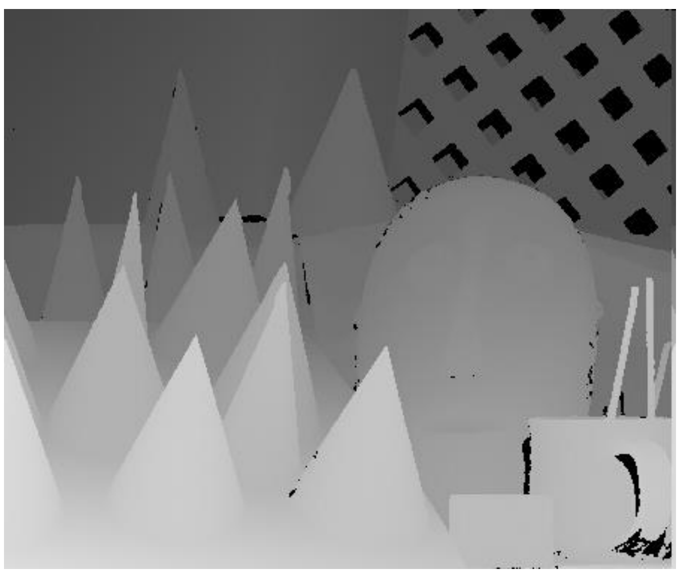
$$u - u' = \frac{B \cdot f}{z}$$ = disparity

Note: Disparity is inversely proportional to depth

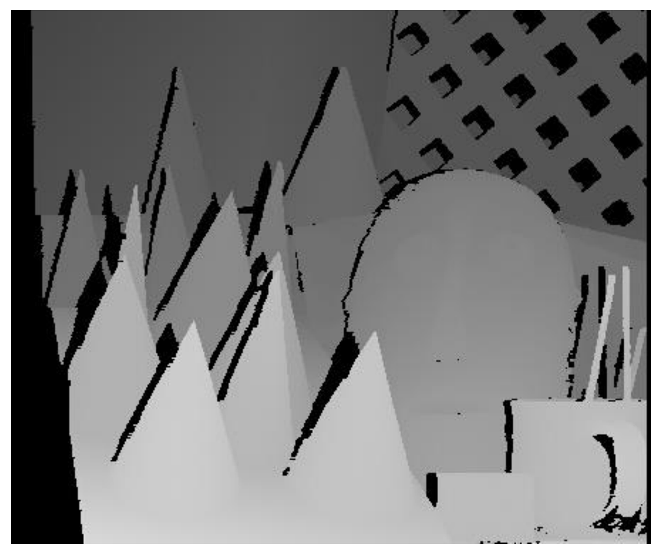
Disparity maps

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

Stereo pair



Disparity map / depth map



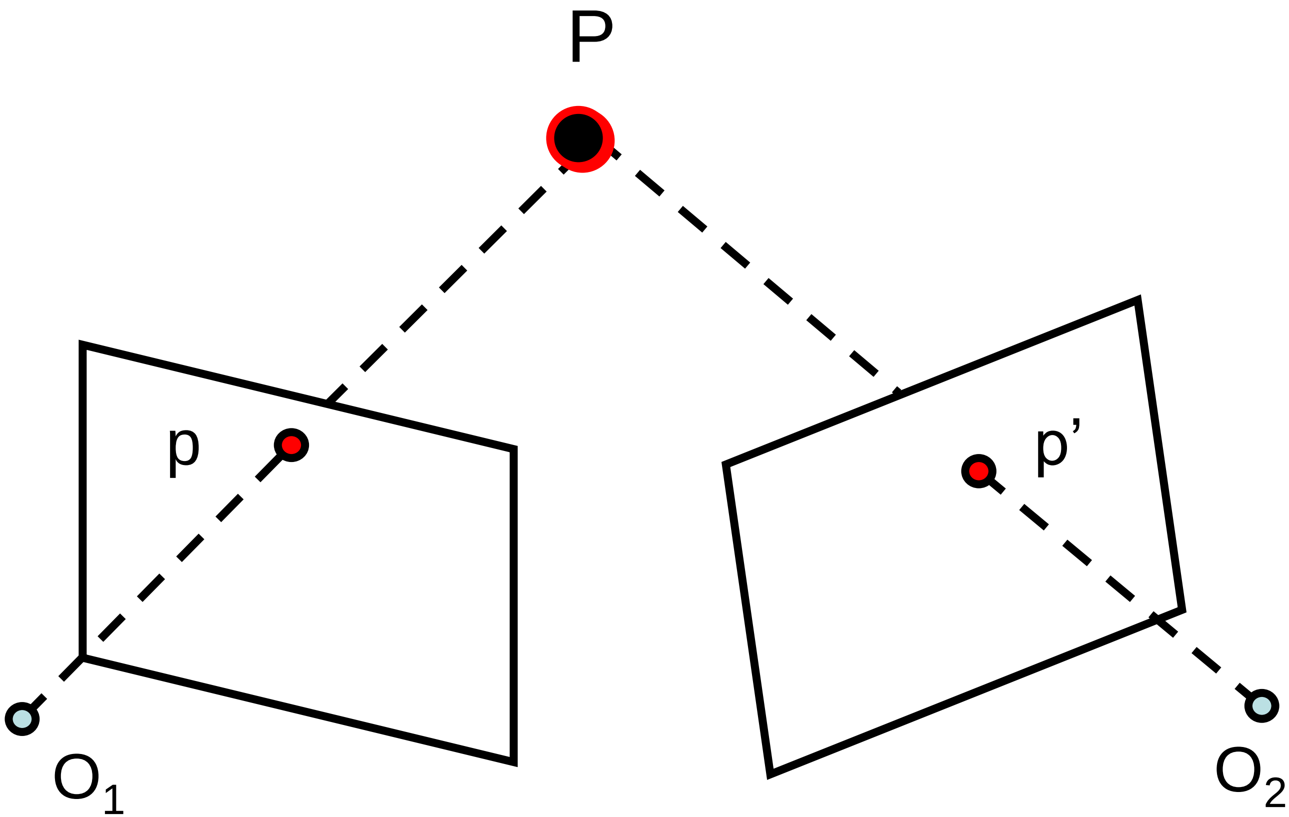
Disparity map with occlusions

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

Why are parallel images useful?

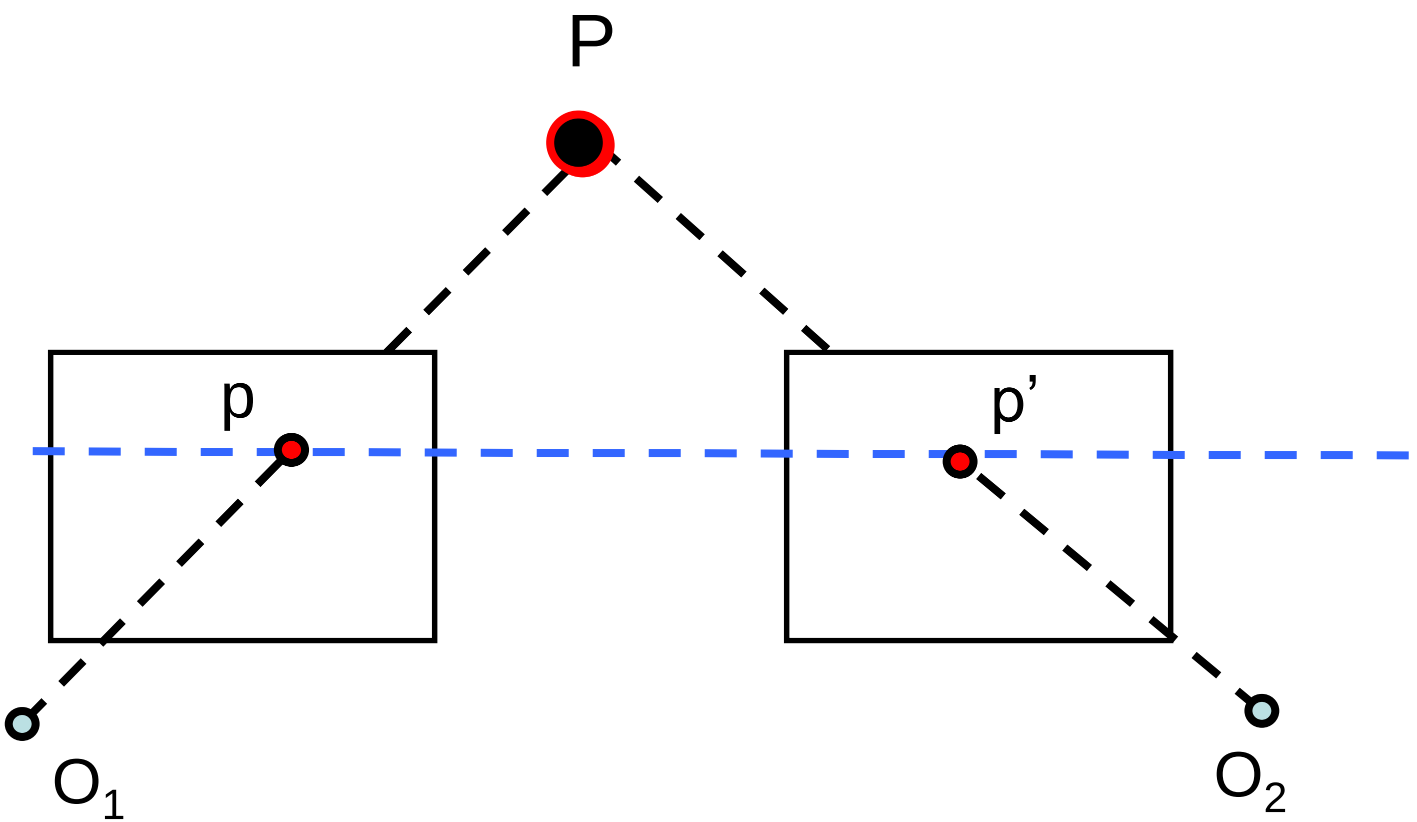
- Makes the correspondence problem easier
- Makes triangulation easy
- Enables schemes for image interpolation

Correspondence problem



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

Correspondence problem



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

Correspondence problem

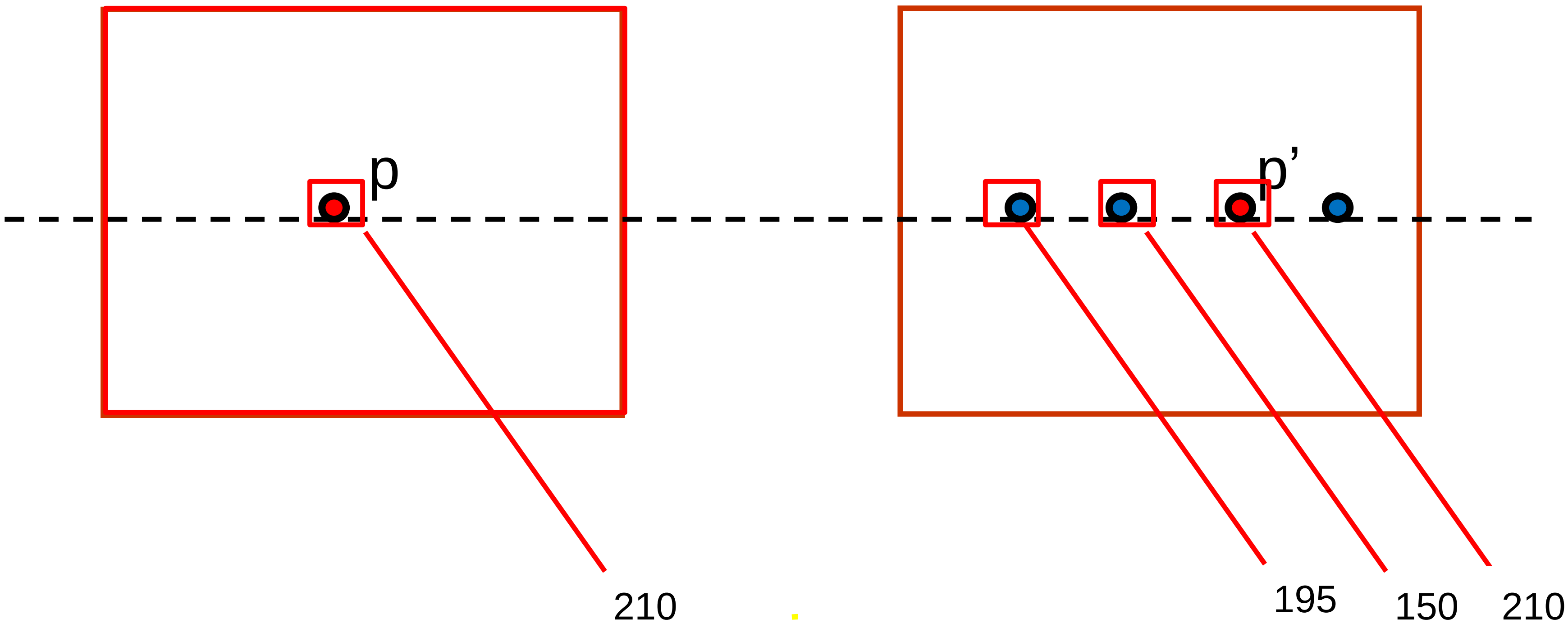
- A Cooperative Model (Marr and Poggio, 1976)

- Correlation Methods (1970--)

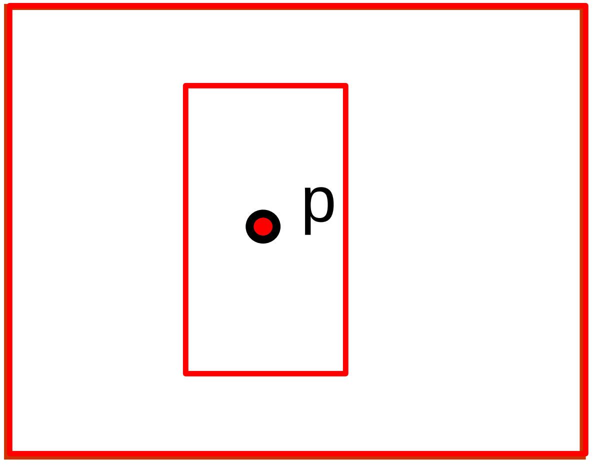
- Multi-Scale Edge Matching (Marr, Poggio and Grimson, 1979-81)

[FP] Chapters: 11

Correlation Methods (1970--)

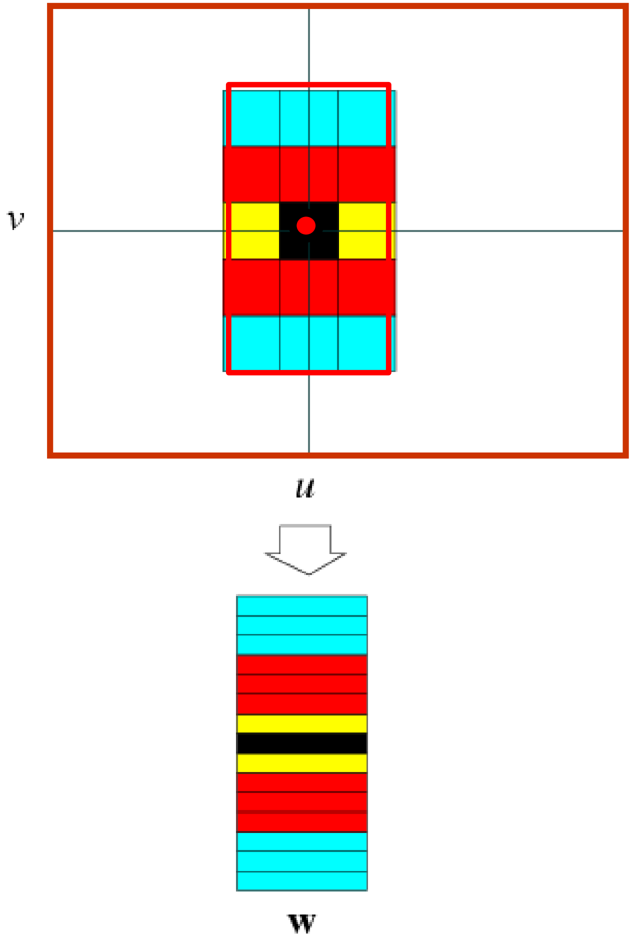


Correlation Methods (1970--)



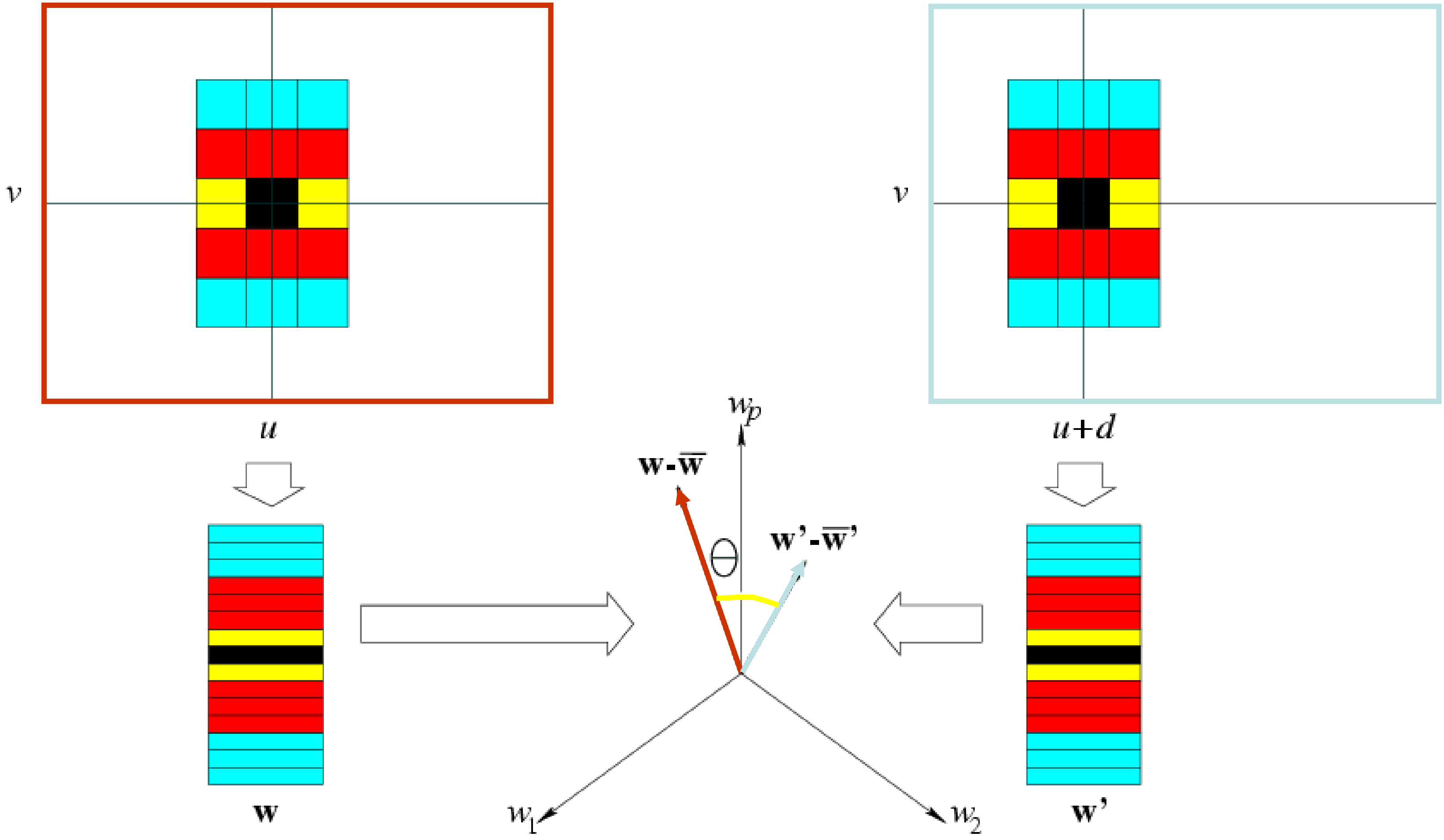
- Pick up a window around p(u,v)

Correlation Methods (1970--)



- Pick up a window around p(u,v)
- Build vector **w**
- Slide the window along v line in image 2 and compute **w**’
- Keep sliding until **w**·**w**’ is maximized.

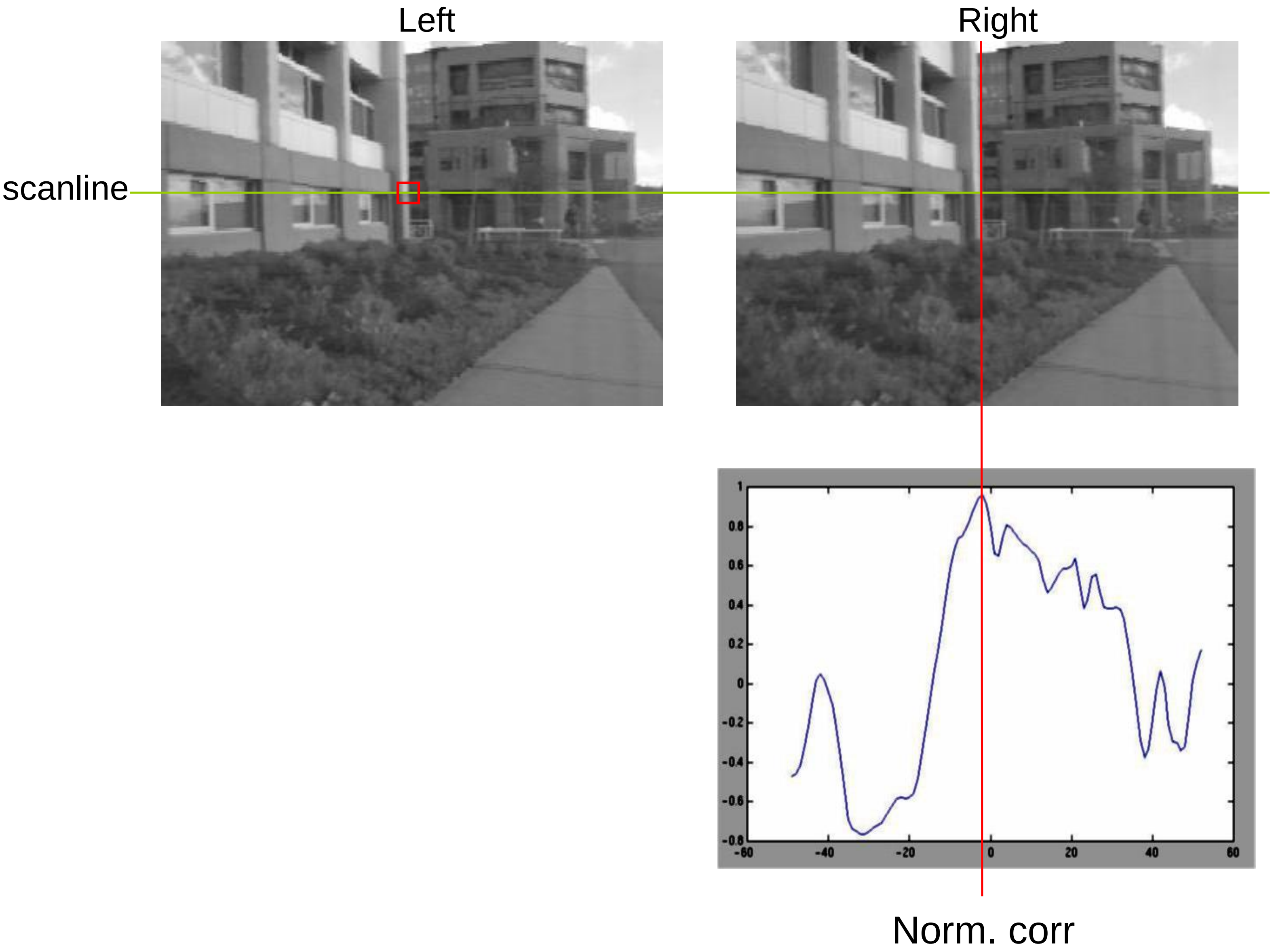
Correlation Methods (1970--)



Normalized Correlation; minimize:

$$\frac{(\mathrm{w}-\overline{\mathrm{w}})(\mathrm{w}'-\overline{\mathrm{w}}')}{\left\|(\mathrm{w}-\overline{\mathrm{w}})(\mathrm{w}'-\overline{\mathrm{w}}')\right\|}$$

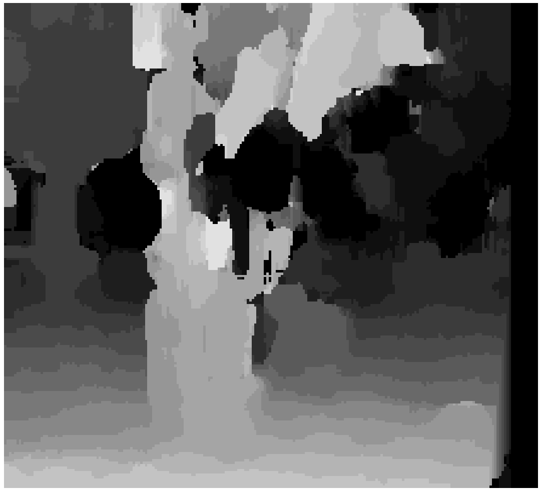
Correlation methods



Credit slide S. Lazebnik

Correlation methods

Window size = 3



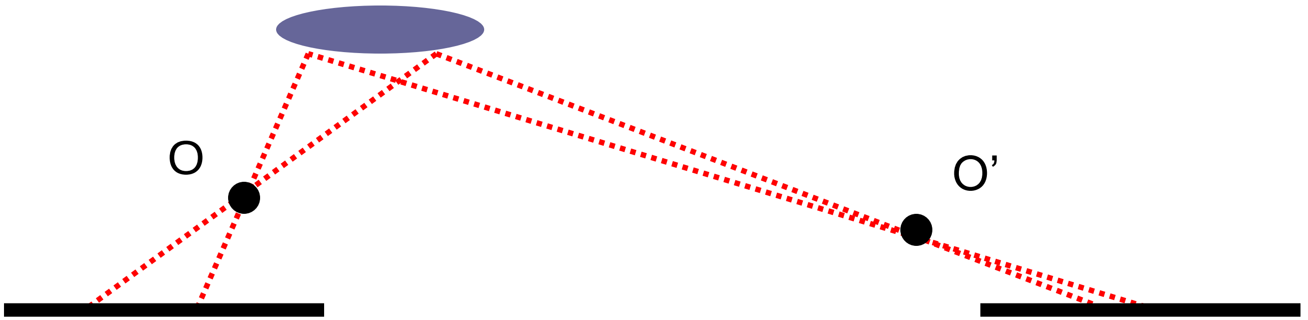
Window size = 20

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

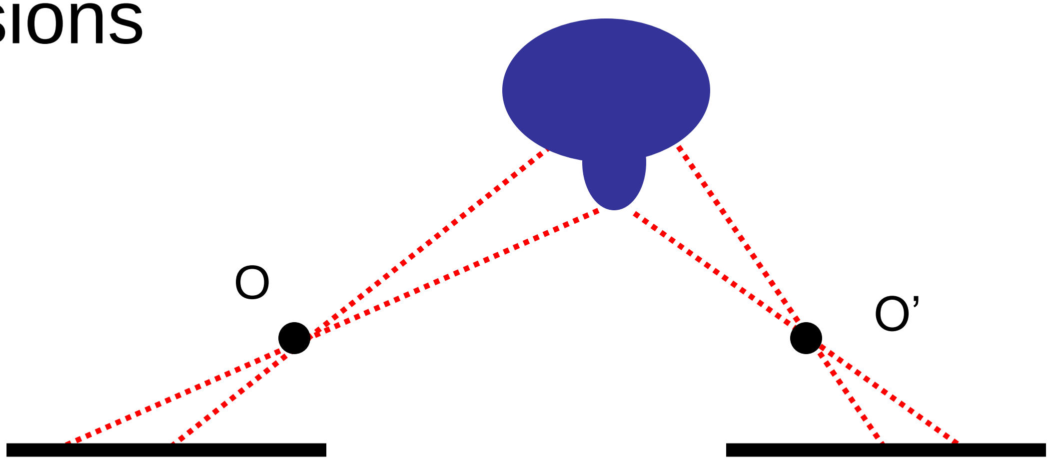
Credit slide S. Lazebnik

Issues

- Fore shortening effect

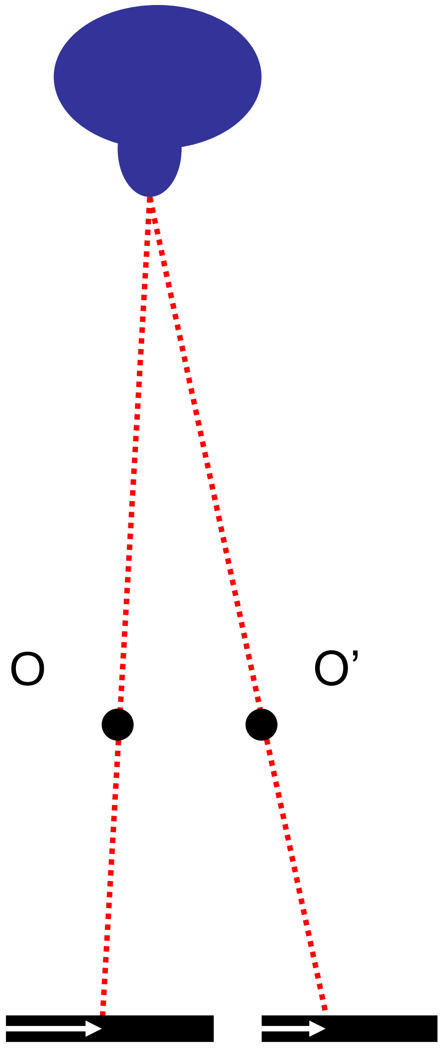


- Occlusions



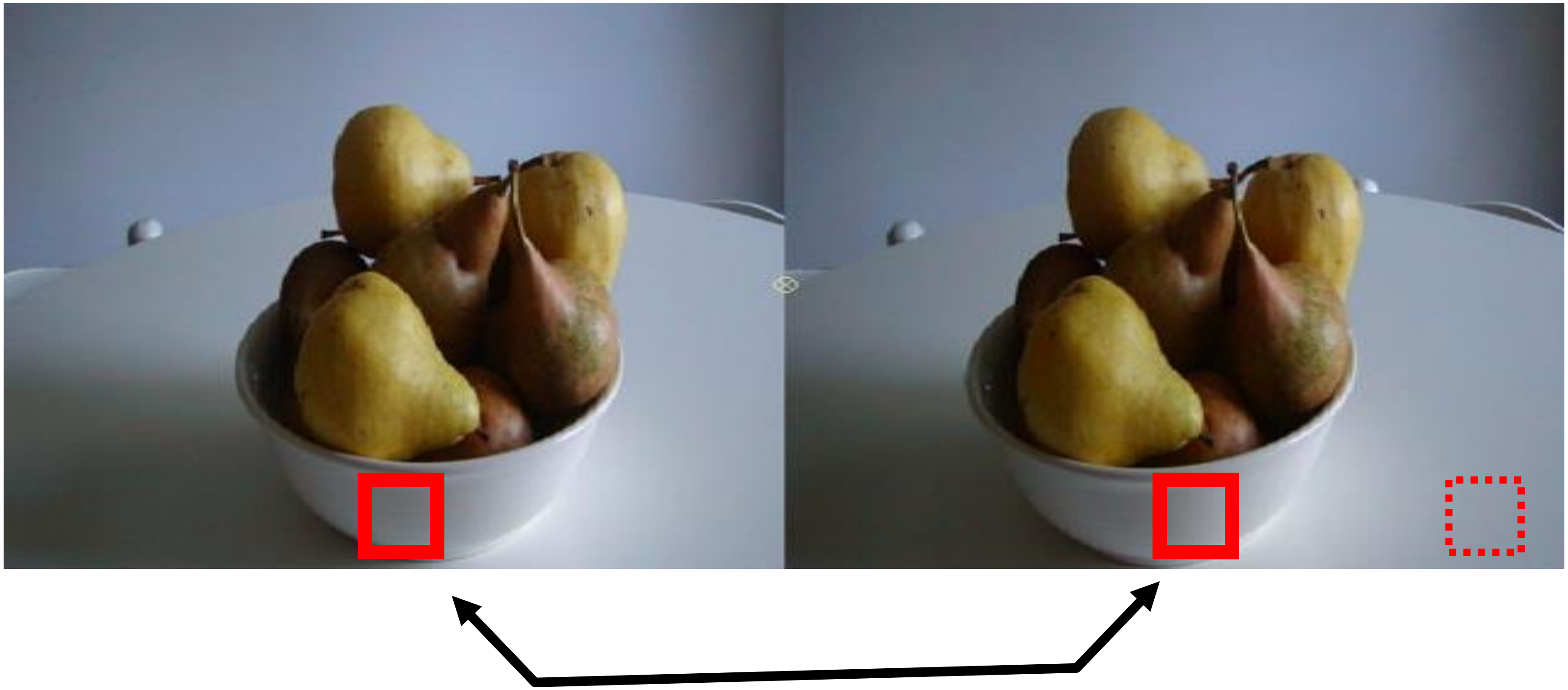
Issues

- It is desirable to have small B/z ratio!
- Small error in measurements implies large error in estimating depth



Issues

- Homogeneous regions



Hard to match pixels in these regions

Issues

- Repetitive patterns

Correspondence problem is difficult!

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

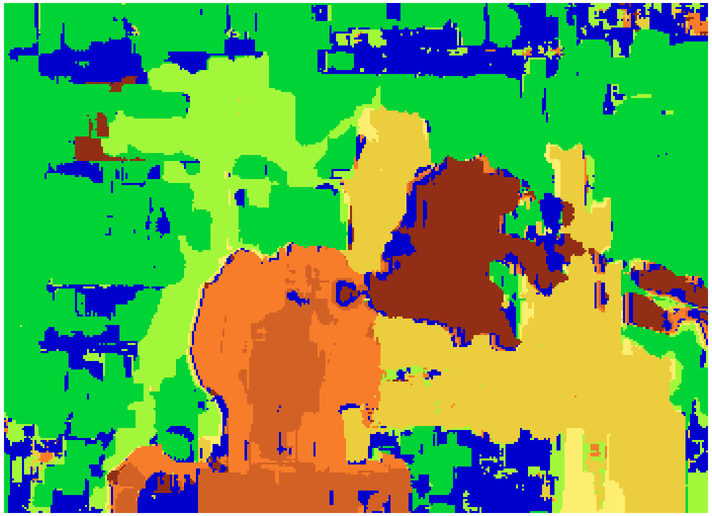
Apply non-local constraints to help enforce the correspondences

Results with window search

Data

Ground truth

Window-based matching



Credit slide S. Lazebnik

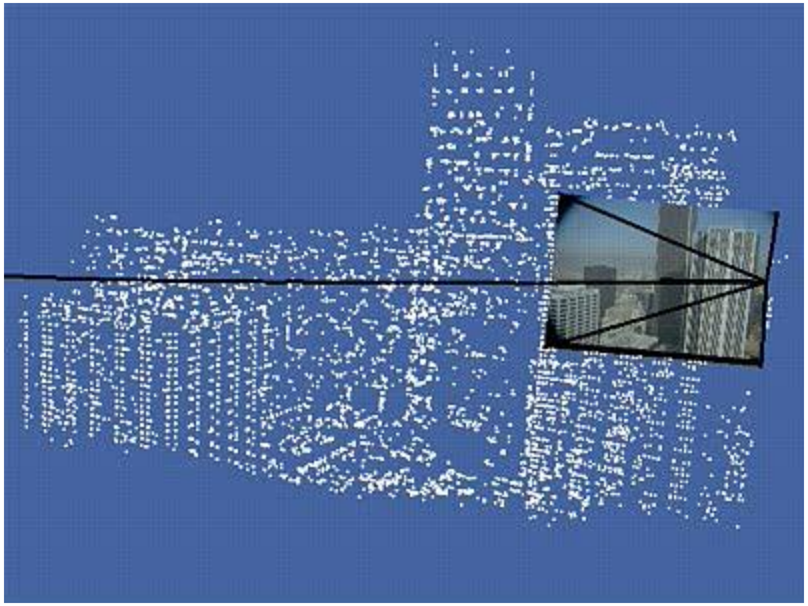
Lecture 6

Stereo Systems
Multi-view geometry

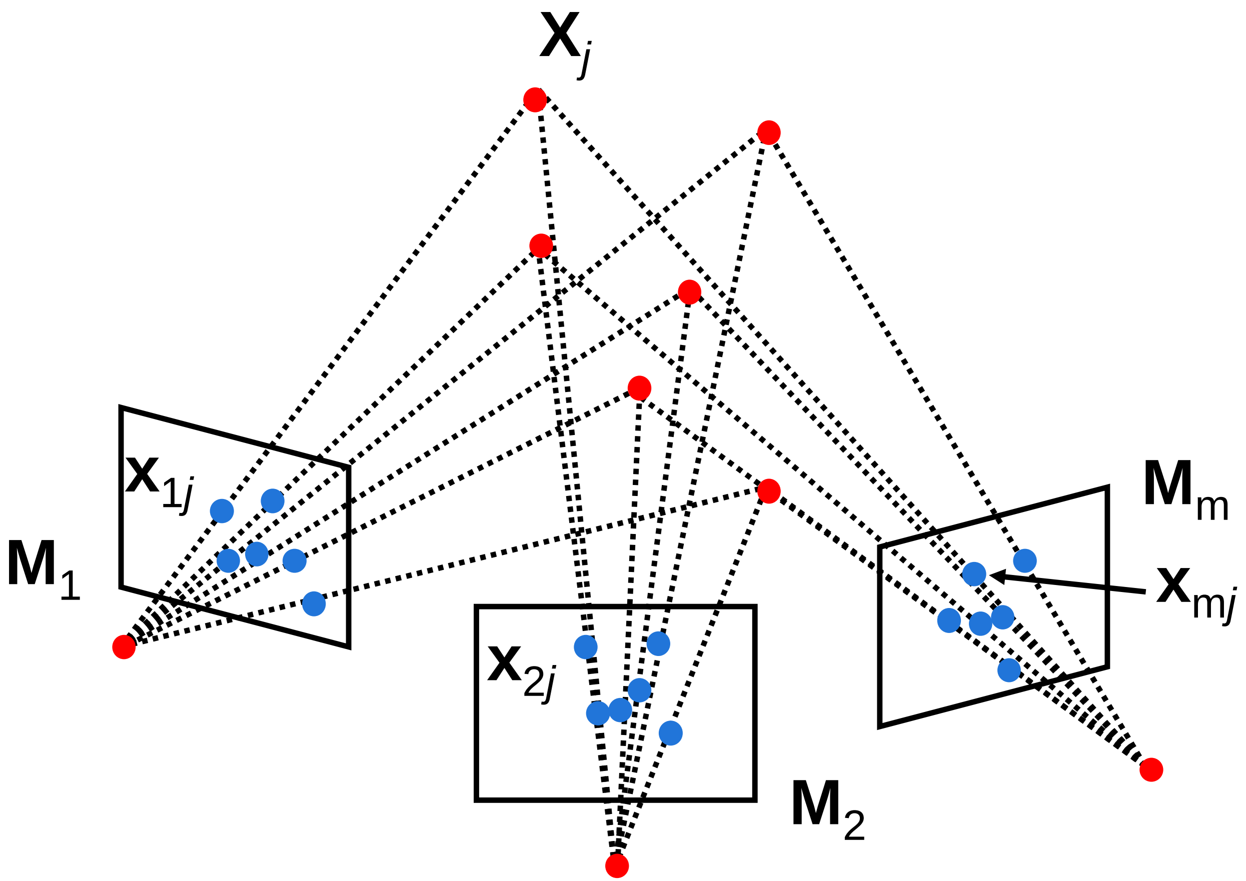
- Stereo systems
    - Rectification
    - Correspondence problem
- Multi-view geometry
    - The SFM problem
    - Affine SFM

Structure from motion problem

Courtesy of Oxford **Visual Geometry Group**



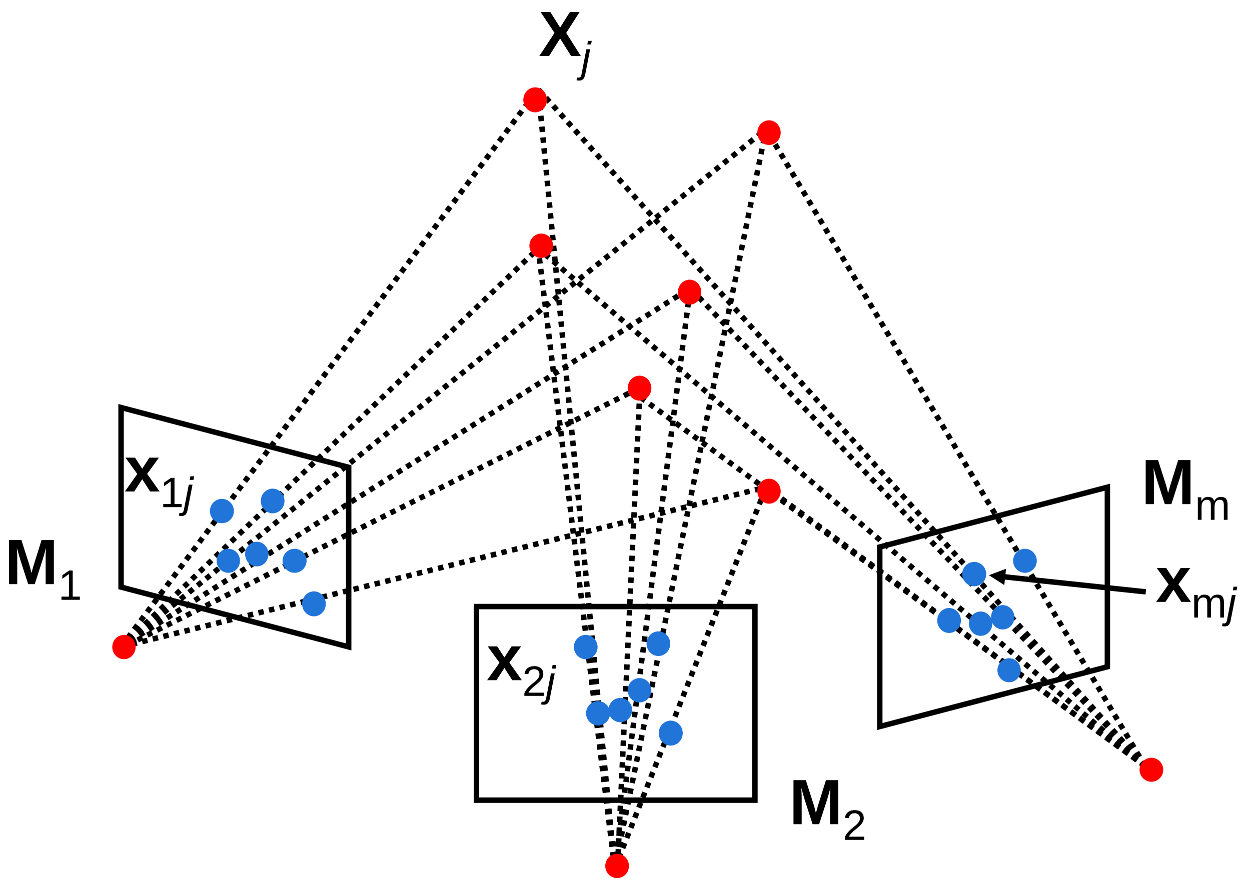
Structure from motion problem



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

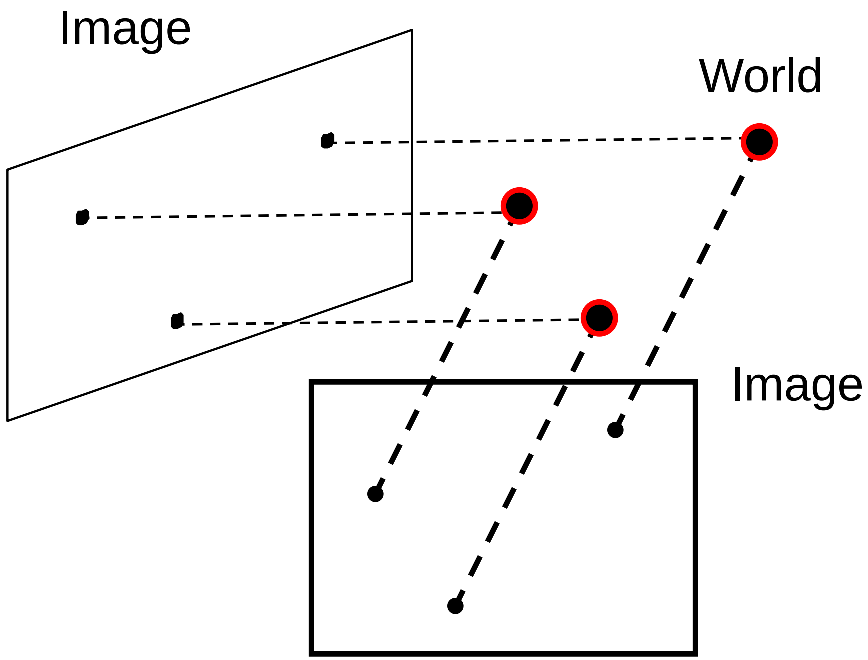


From the m×n correspondences $\mathbf{x}_{ij}$, estimate:

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

Affine structure from motion

(simpler problem)

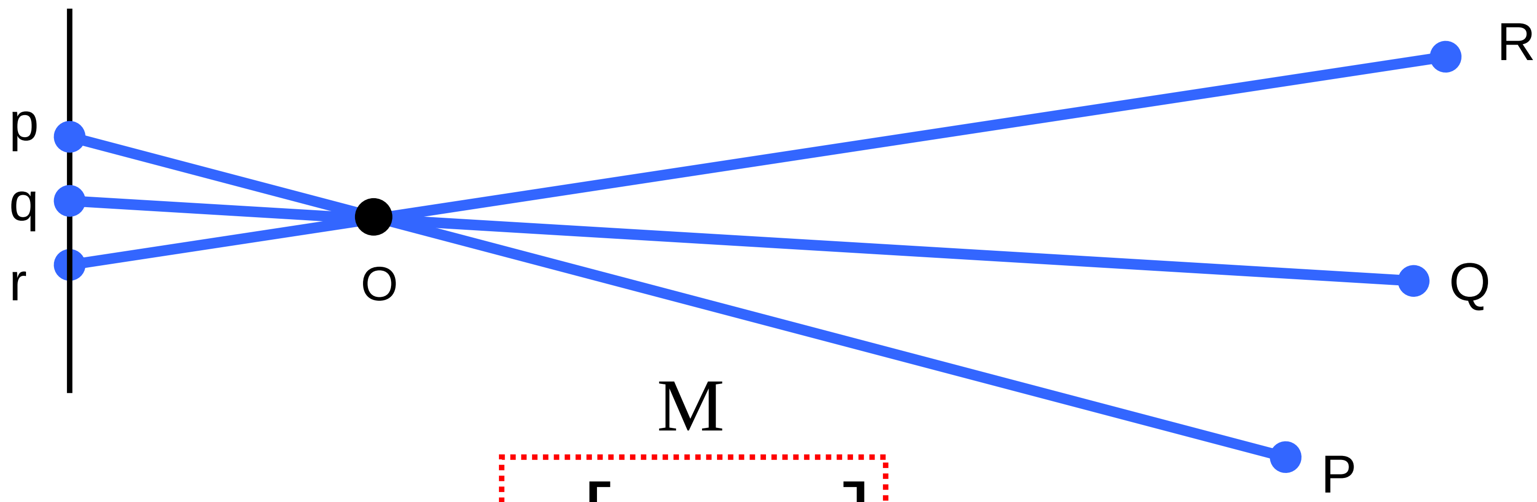


From the m×n correspondences $\mathbf{x}_{ij}$, estimate:

•*m* projection matrices $\mathbf{M}_i$ (affine cameras)

•*n* 3D points $\mathbf{X}_j$

Finite cameras

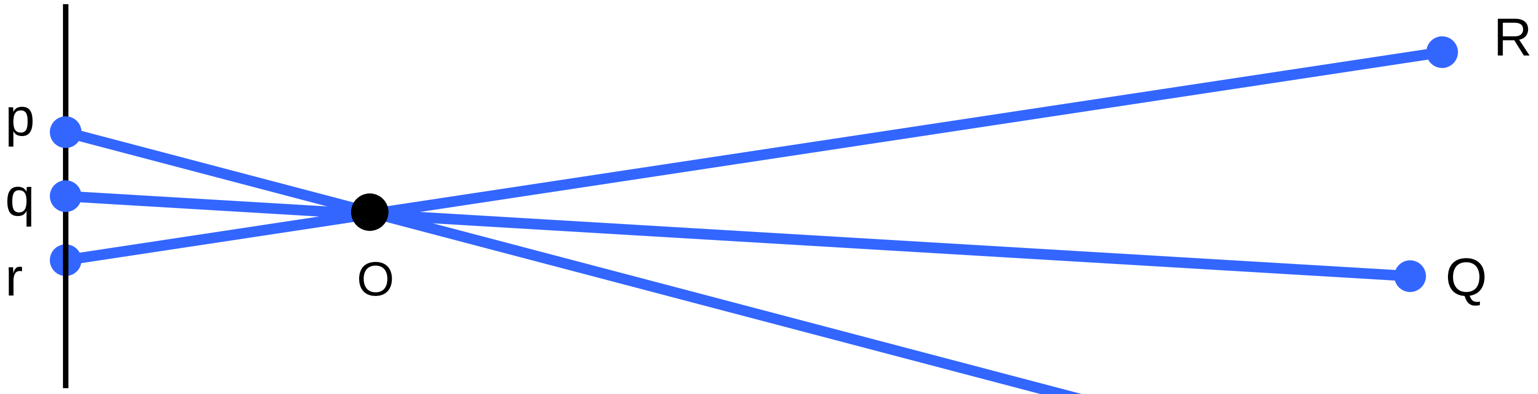


$$x = \overbrace{K\begin{bmatrix}R & T\end{bmatrix}}^{M}X$$

Question:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} = ??$$

Finite cameras



$$x = K\begin{bmatrix} R & T \end{bmatrix} X$$

Canonical perspective projection matrix

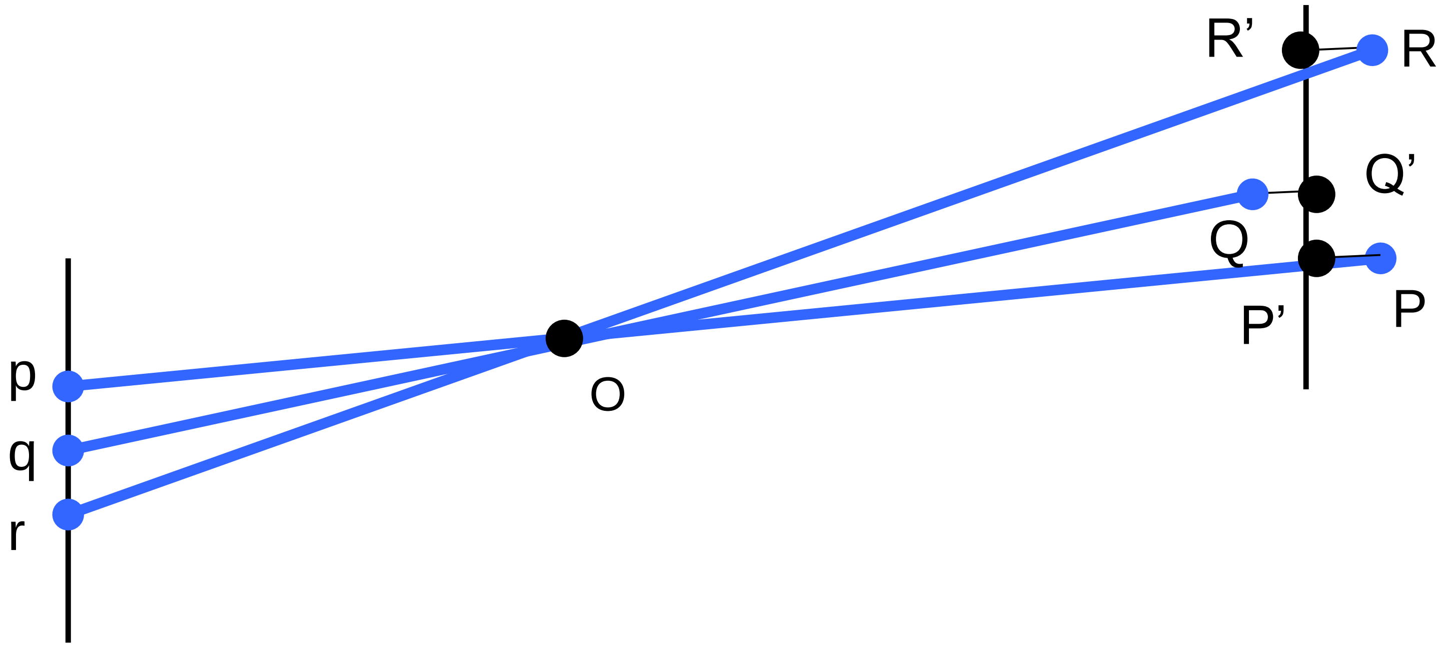
$$M = K_{3x3}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$$

Affine Homography (in 2D)

Affine homography (in 3D)

$$K = \begin{bmatrix} \alpha_x & s & x_o \\ 0 & \alpha_y & y_o \\ 0 & 0 & 1 \end{bmatrix}$$

Affine cameras



Canonical affine projection matrix
(points at infinity are mapped as points at infinity)

What's the main difference???

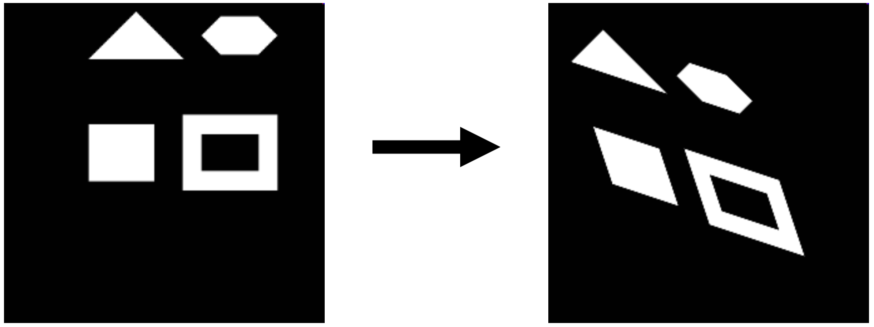
$$M = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \boxed{0} & 1 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$$

$$K = \begin{bmatrix} \alpha_x & s & x_o \\ 0 & \alpha_y & y_o \\ 0 & 0 & 1 \end{bmatrix}$$

Transformation in 2D

Affinities:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_a \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



Affine cameras

$$\mathrm{x} = \mathrm{K}\begin{bmatrix}\mathrm{R} & \mathrm{T}\end{bmatrix}\mathrm{X}$$ [Homogeneous]

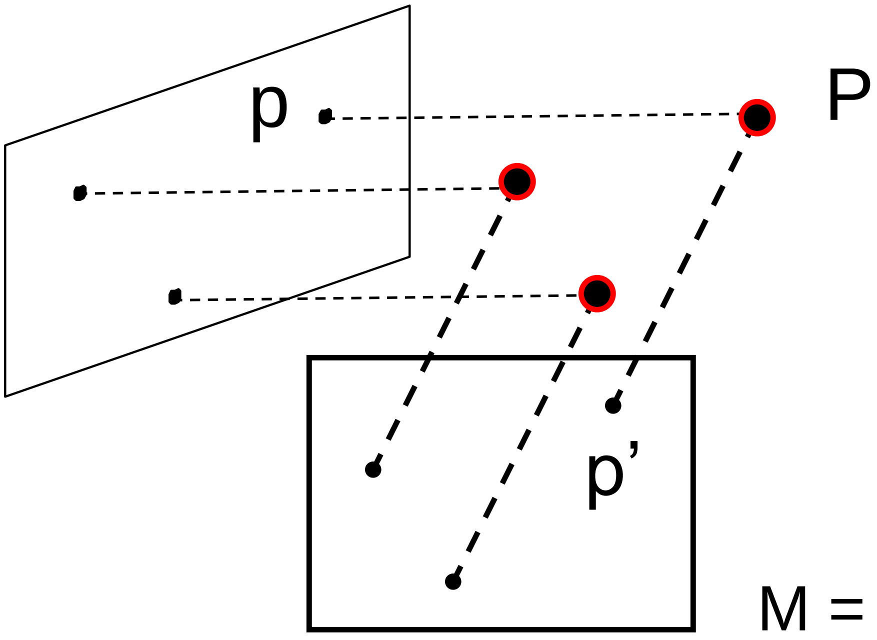
$$K = \begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathrm{M} = \mathrm{K}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \mathrm{R} & \mathrm{T} \\ 0 & 1 \end{bmatrix}$$

$$M = [3\times 3\,\text{affine}]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}[4\times 4\,\text{affine}] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \mathbf{AX} + \mathbf{b} = M_{Euc}\begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = M_{Euc}\begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix};$$

$$\mathrm{M}_{\mathrm{Euc}} = \mathrm{M} = \begin{bmatrix}\mathrm{A} & \mathrm{b}\end{bmatrix}$$ [non-homogeneous image coordinates]

Affine cameras



M = camera matrix

To recap:

from now on we define M as the camera matrix for the affine case

$$\mathbf{p} = \begin{pmatrix} u \\ v \end{pmatrix} = \mathbf{AP} + \mathbf{b} = M \begin{bmatrix} \mathbf{P} \\ 1 \end{bmatrix}; \qquad \mathrm{M} = \begin{bmatrix} \mathrm{A} & \mathrm{b} \end{bmatrix}$$

The Affine Structure-from-Motion Problem

Given *m* images of *n* fixed points P_j (=X_i) we can write

$$\boldsymbol{p}_{ij} = \mathcal{M}_i \begin{pmatrix} \boldsymbol{P}_j \\ 1 \end{pmatrix} = \mathcal{A}_i \boldsymbol{P}_j + \boldsymbol{b}_i \quad \text{for} \quad i = 1, \ldots, \boxed{m} \quad \text{and} \quad j = 1, \ldots, \boxed{n}.$$

N of cameras   N of points

Problem: estimate the m 2×4 matrices M_i and the n positions P_j from the m×n correspondences $\mathbf{p}_{ij}$.

How many equations and how many unknown?

2m × n equations in 8m+3n unknowns

Two approaches:

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

A factorization method – Tomasi & Kanade algorithm

C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137-154, November 1992.

- Centering the data
- Factorization

Next lecture

Multiple view geometry