Computer Vision

Lec 7 - stereo

University of Haifa

Simon Korman

Stereo

(+ recaps on 2-view geometry and homographies)

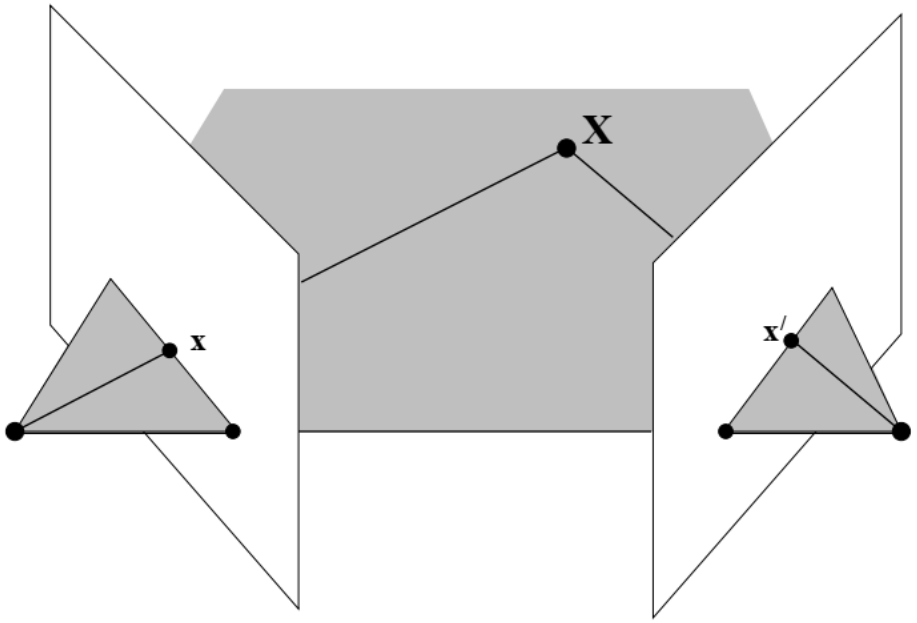
slide credit

- Svetlana Lazebnik, Sanja Fidler, Kristen Grauman, Ioannis Gkioulekas, Kris Kitani, James Hays, Fredo Durand, Rick Szeliski, Andrew Zisserman, Kyros Kutulakos, Srinivasa Narasimhan, Rob Fergus, Noah Snavely

recap 1: Two-View Geometry

or **3D reconstruction of cameras and scene structure**
(from 2 views)

- Compute point correspondences
- Compute F (or E) from point correspondences
- Compute P and P’ from F
- Triangulate – compute the point in 3D which projects to each pair of points



recap 1: Essential and Fundamental

E vs. F (general)

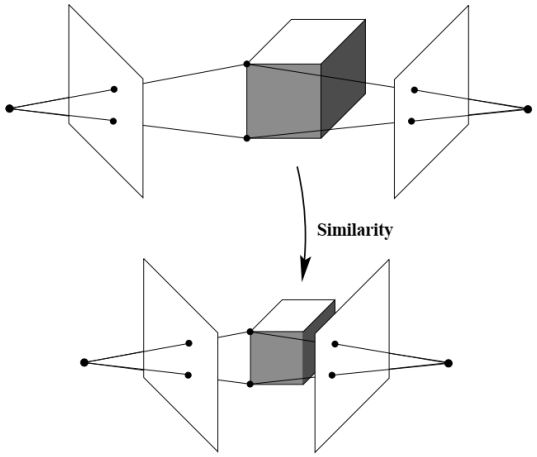
- for a match x and x':
  - $Ex = l' \quad x'^T l' = 0 \quad x'^T E x = 0$
- transpose: $x^T E^T x' = 0$
- epipole: $Ee = 0$
- $E = R[t]_\times$
  - Rank 2 ($\det E = 0$)
    - with 2 equal singular values
  - 5 dof (R, t up to scale)
  - requires 5 matches
- algorithms:
  - 8 point (linear)
  - 7 point
  - 5 point

- for a match x and x':
  - $Fx = l' \quad x'^T l' = 0 \quad x'^T F x = 0$
- transpose: $x^T F^{T\prime} x' = 0$
- epipole: $Fe = 0$
- $F = K'^{-T} R[t]_\times K^T$
  - Rank 2 ($\det F = 0$)
  - 8/7 dof (scale / + rank 2)
  - requires 8/7 matches
- algorithms:
  - 8 point (linear)
  - 7 point

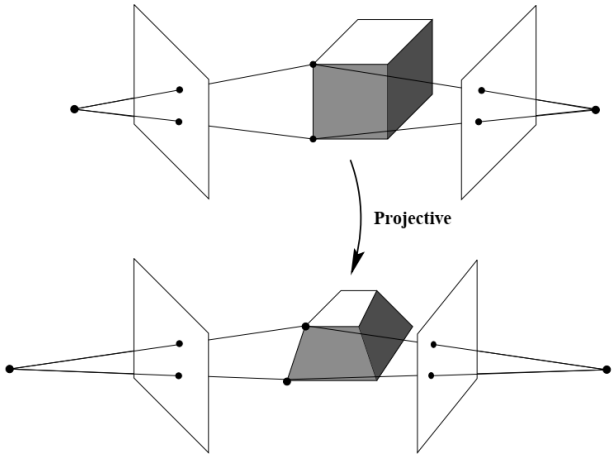
recap 1: Essential and Fundamental

E vs. F reconstruction (finding P, P')

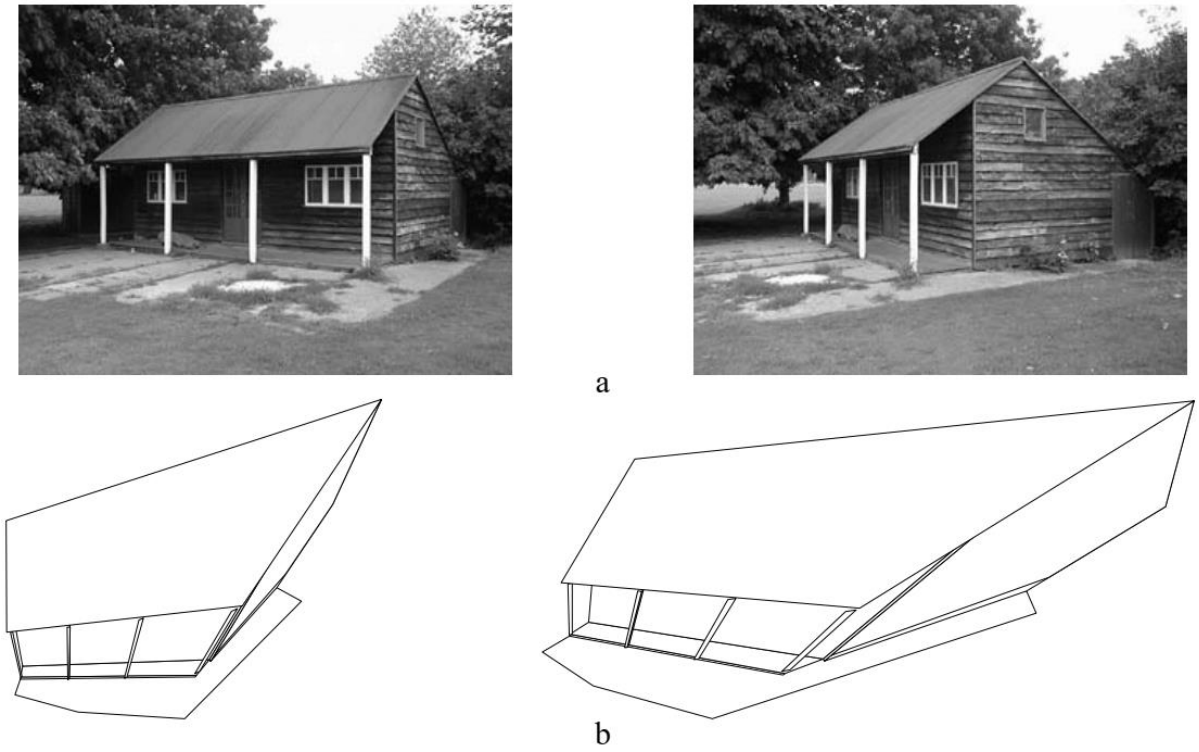
- $E = [e']_\times P' P^+$
  - where $e' = P'C$, $PC = 0$
  - **metric** (similarity) reconstruction
- camera reconstruction:
  - $P = [I|0]$
  - 4 options for P' based on *SVD*(E)
    - (only one is valid)



- $F = [e']_\times P' P^+$
  - where $e' = P'C$, $PC = 0$
  - **projective** reconstruction
- camera reconstruction:
  - $P = [I|0]$
  - $P' = [e' \times F + e' v^T | \lambda e']$
    - for any 3-vector v and scalar λ

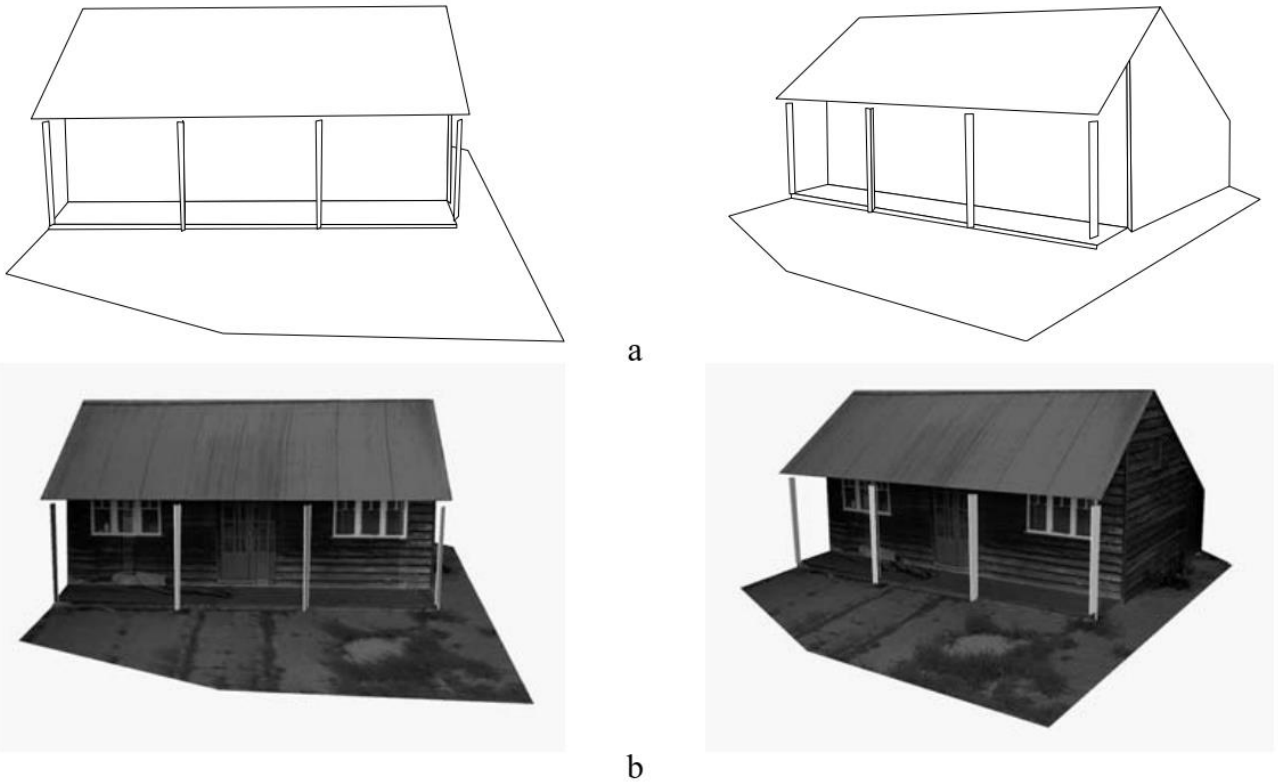


recap 1: projective reconstruction



- **projective** reconstruction $\langle P_P, P'_P \rangle$
  - preserving lines, but not angles, distances
- related to **true** reconstruction $\langle P, P' \rangle$
- by some unknonwn 3D homography H (4x4 matrix):
  - $P = P_P H^{-1}$ and $\mathrm{P}' = P'_P H^{-1}$

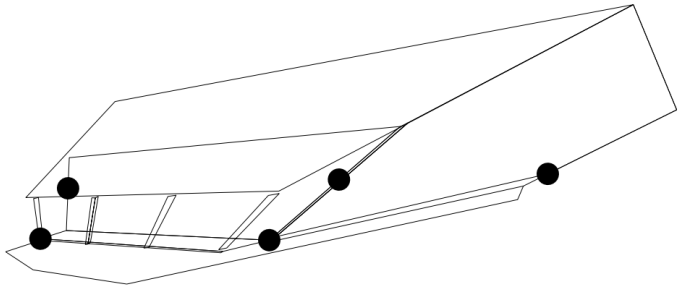
recap 1: metric reconstruction



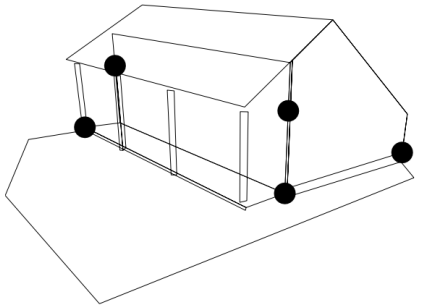
- **metric** (Euclidean, Similarity) reconstruction $\langle P_S, P'_S \rangle$
  - preserving lines, angles, distances, but not scale and absolute position
- Related to **true** reconstruction $\langle P, P' \rangle$
- by some unknonwn 3D similarity S (4x4 matrix):
  - $P_P = PS^{-1}$ and $P'_P = P'S^{-1}$

recap 1: proj. to metric reconstruction

a



b



c

- **perfect** (metric) reconstruction $\langle P, P' \rangle$
  - preserving everything
- Requires at least 5 points (each giving 3 equations)
- to reconstruct the 15 dof projective matrix H

recap 2: Homographies

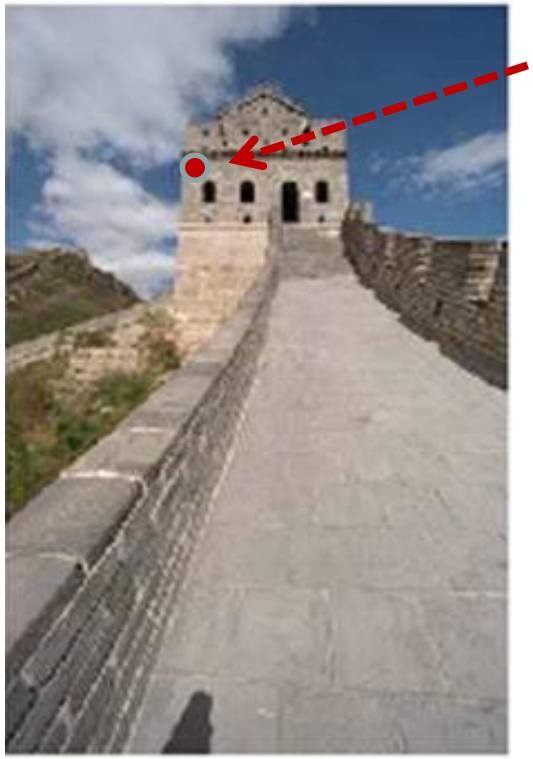
- Homography = Projectivity = Projective Transformation
- Definition(s):
  - A mapping from $\mathbb{P}^2$ (homog. 3D vectors) to itself that preserves collinearity (3 collinear points / line)
  - Any 3x3 non-singular matrix H (acting on homog. vectors)

$$\mathbf{x}' = \mathbf{H}\mathbf{x}$$

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

recap 2: Homography

(x, y)



$\left(wx'/w, wy'/w\right) = (x', y')$

To **apply** a given homography **H**

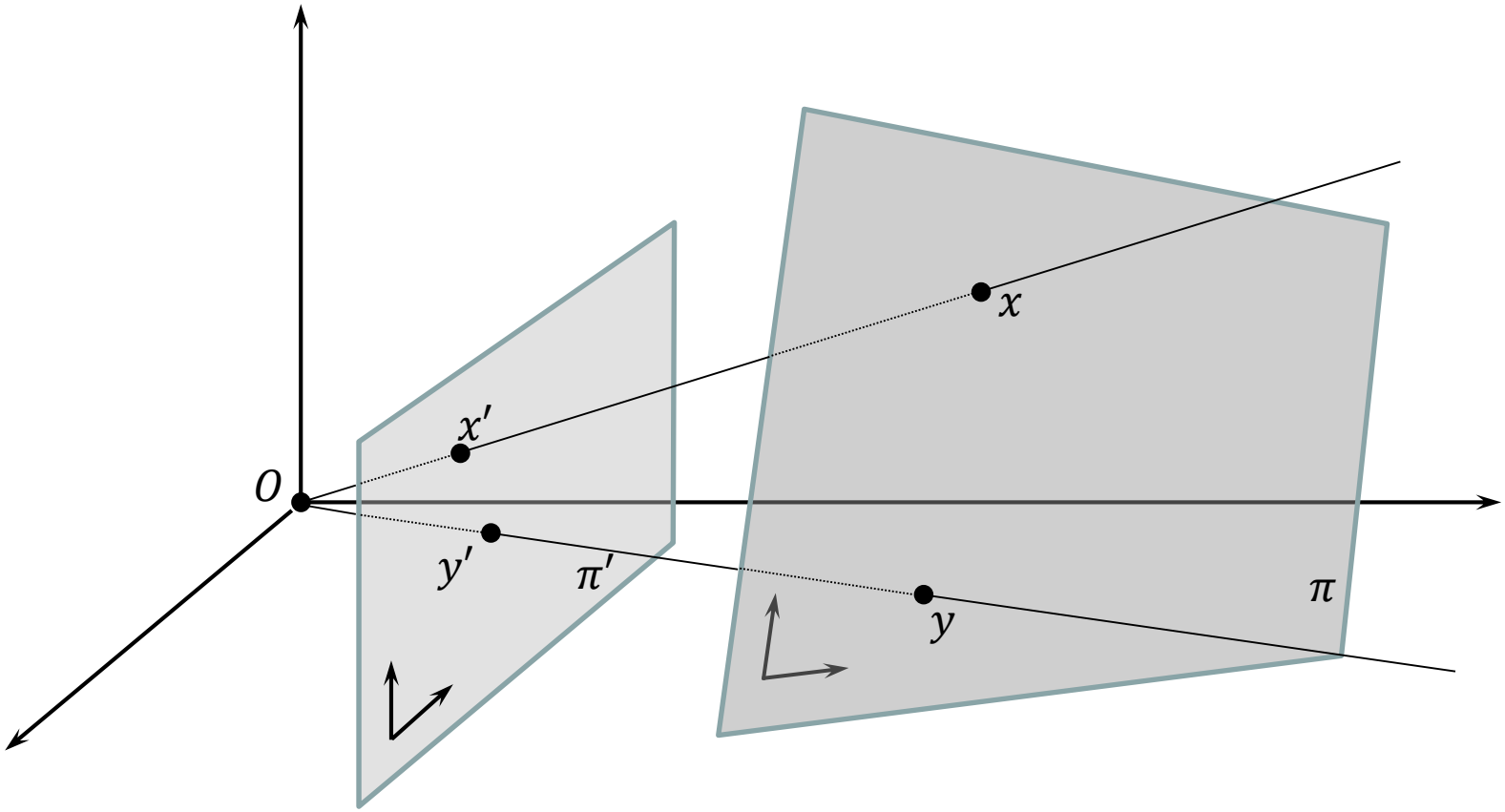
- Compute **p'** = **Hp** (regular matrix multiply)
- Convert **p'** to image coordinates

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

p' **H** **p**

recap 2: Homographies

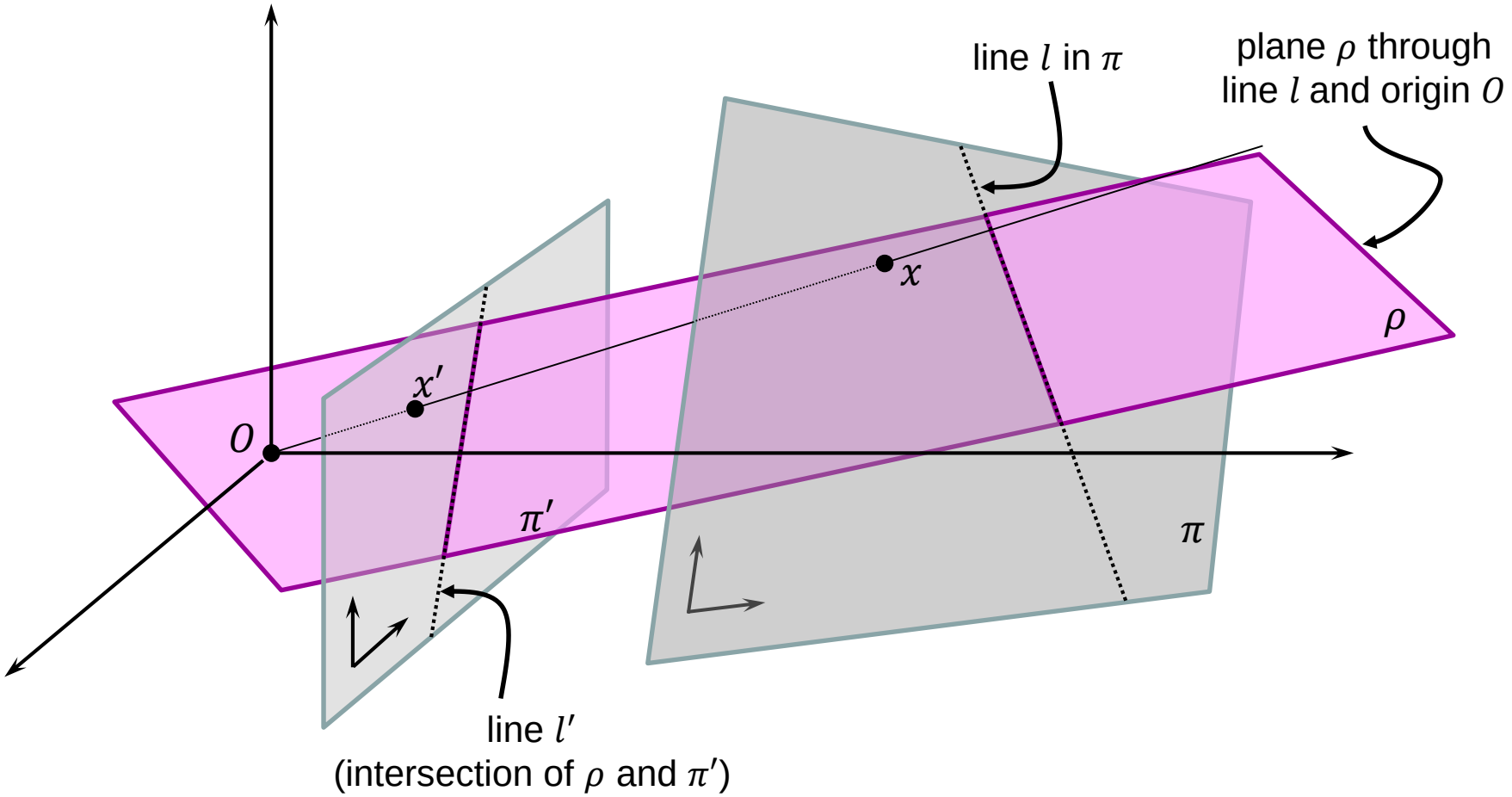
The “central projection” case



recap 2: Homographies

The “central projection” case

⇒ defines a homography between the planes



recap 2: Homographies

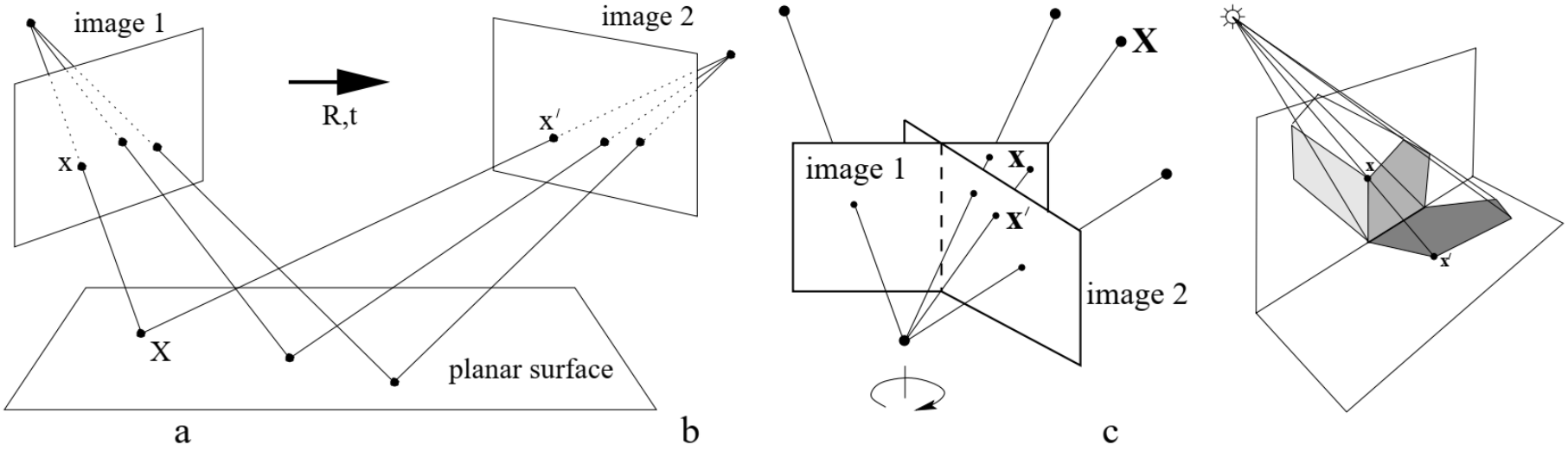
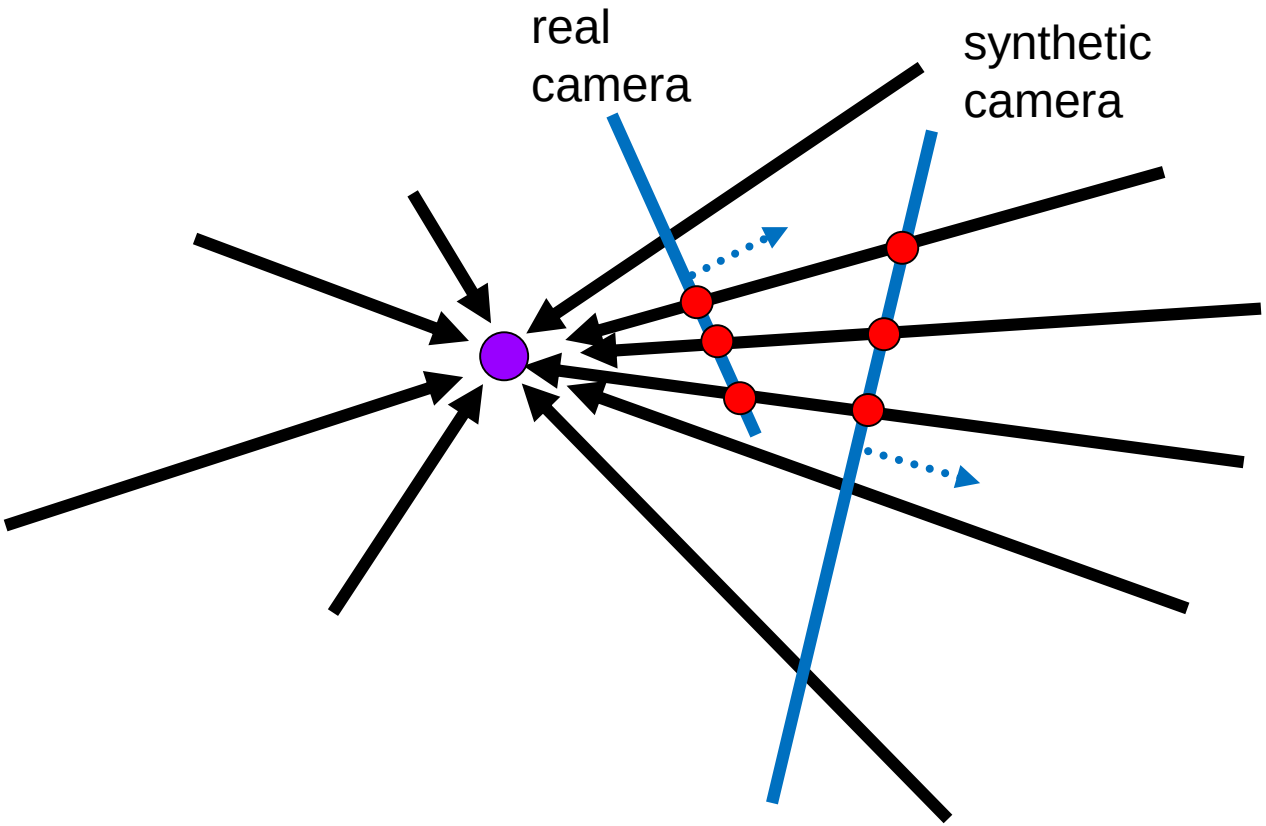


Fig. 2.5. **Examples of a projective transformation, $\mathbf{x}' = \mathtt{H}\mathbf{x}$, arising in perspective images.** *(a) The projective transformation between two images induced by a world plane (the concatenation of two projective transformations is a projective transformation); (b) The projective transformation between two images with the same camera centre (e.g. a camera rotating about its centre or a camera varying its focal length); (c) The projective transformation between the image of a plane (the end of the building) and the image of its shadow onto another plane (the ground plane). Figure (c) courtesy of Luc Van Gool.*

All cases lead to homographies through the “central projection” case

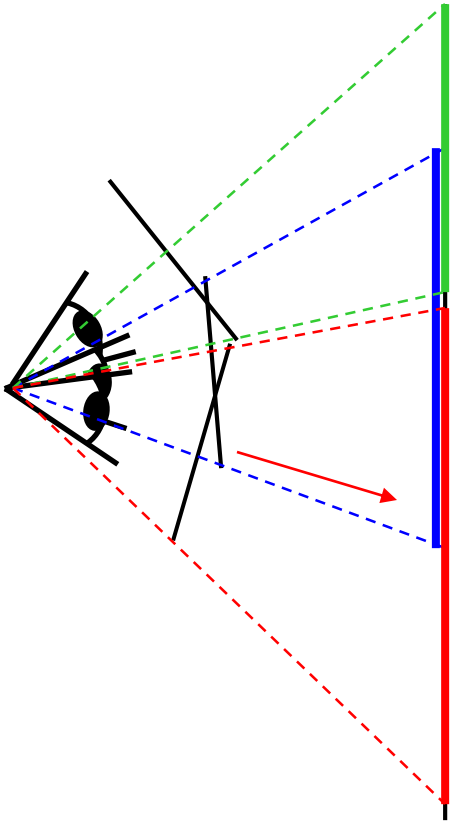
Recall: same camera center



Can generate synthetic camera view
as long as it has **the same center of projection**.

Source: Alyosha Efros

Image reprojection



mosaic common plane

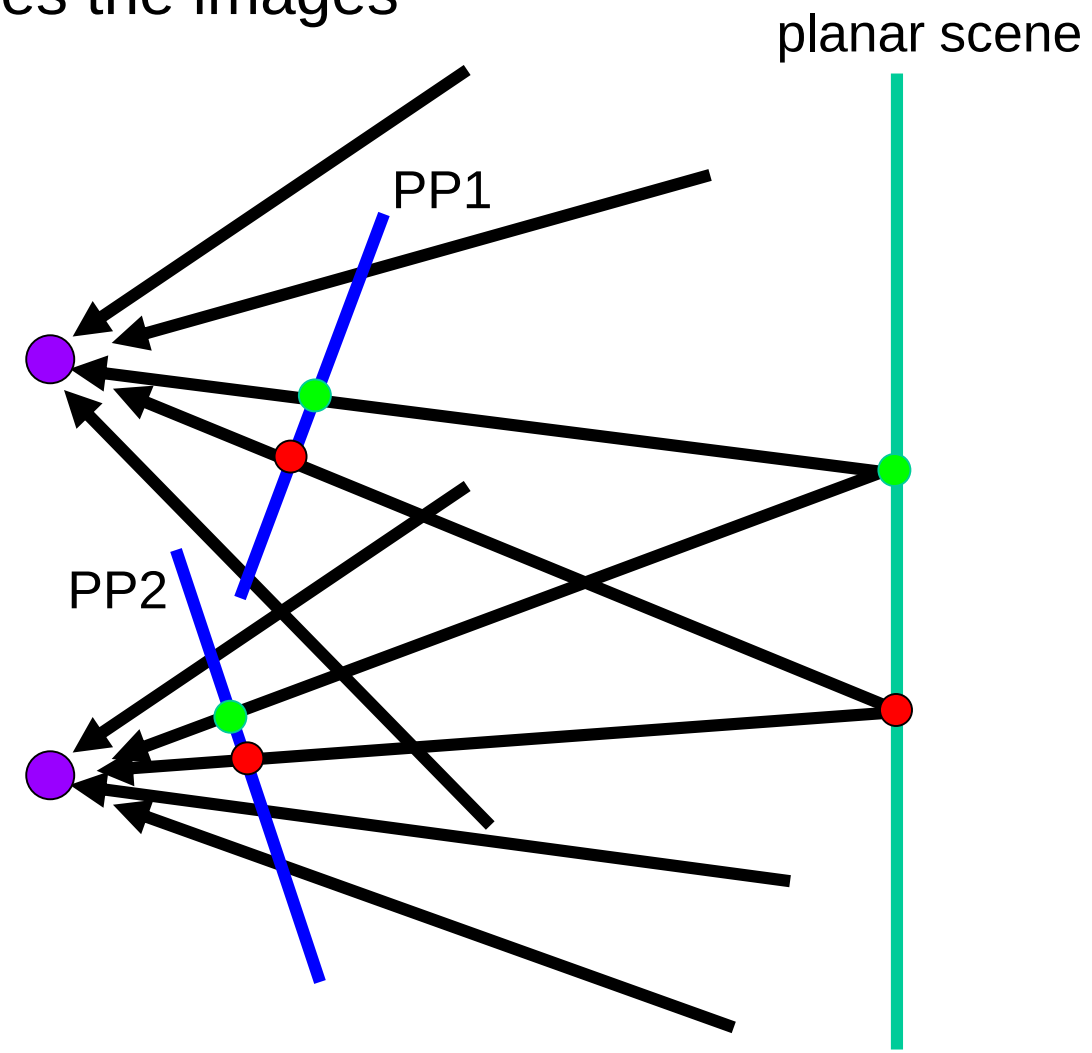
The mosaic has a natural interpretation in 3D

- The images are reprojected onto a common plane
- The mosaic is formed on this plane
- Mosaic is a *synthetic wide-angle camera*

Source: Steve Seitz

Changing camera center – planar scene

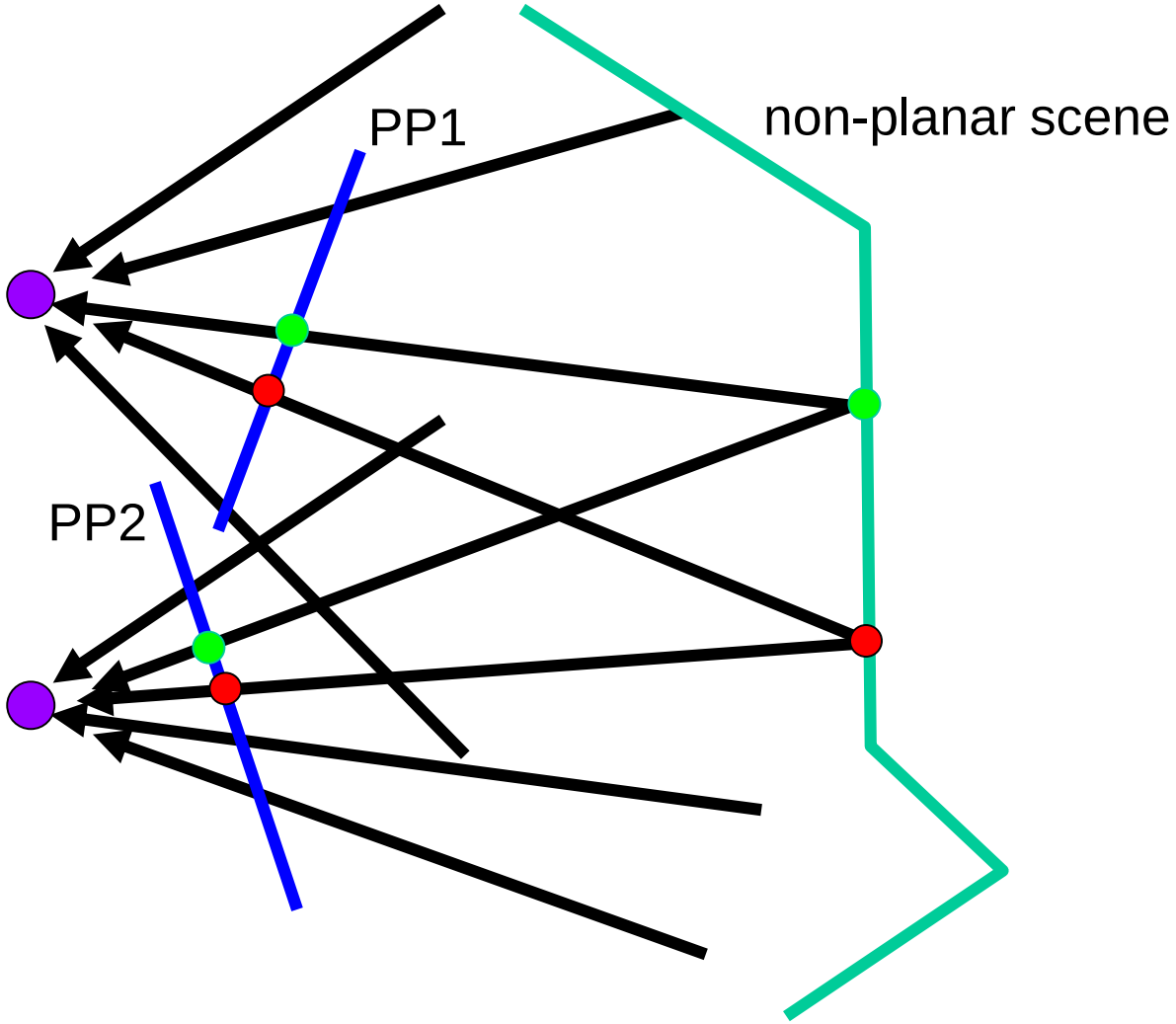
Homography relates the images



Source: Alyosha Efros

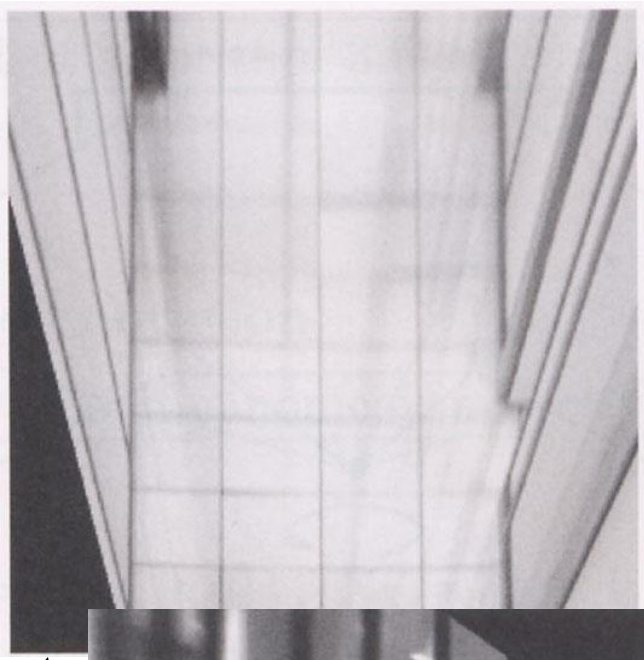
Changing camera center – nonplanar scene

Does it still work? Only for single planar region



Source: Alyosha Efros

Image warping with homographies



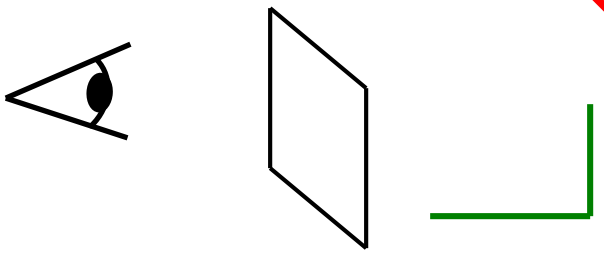
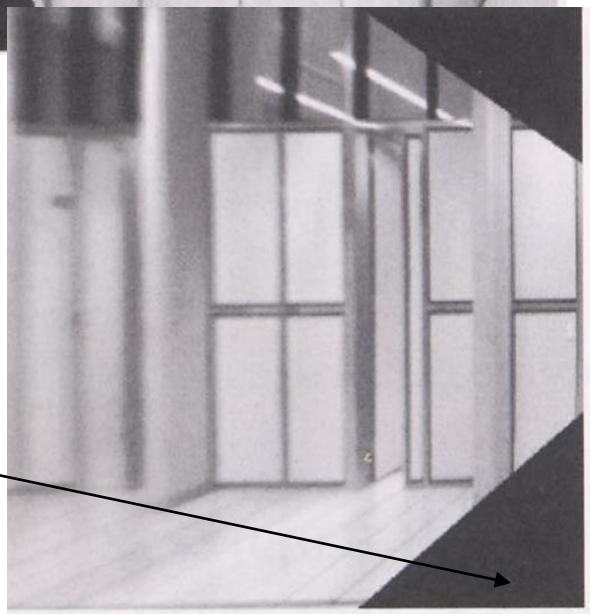


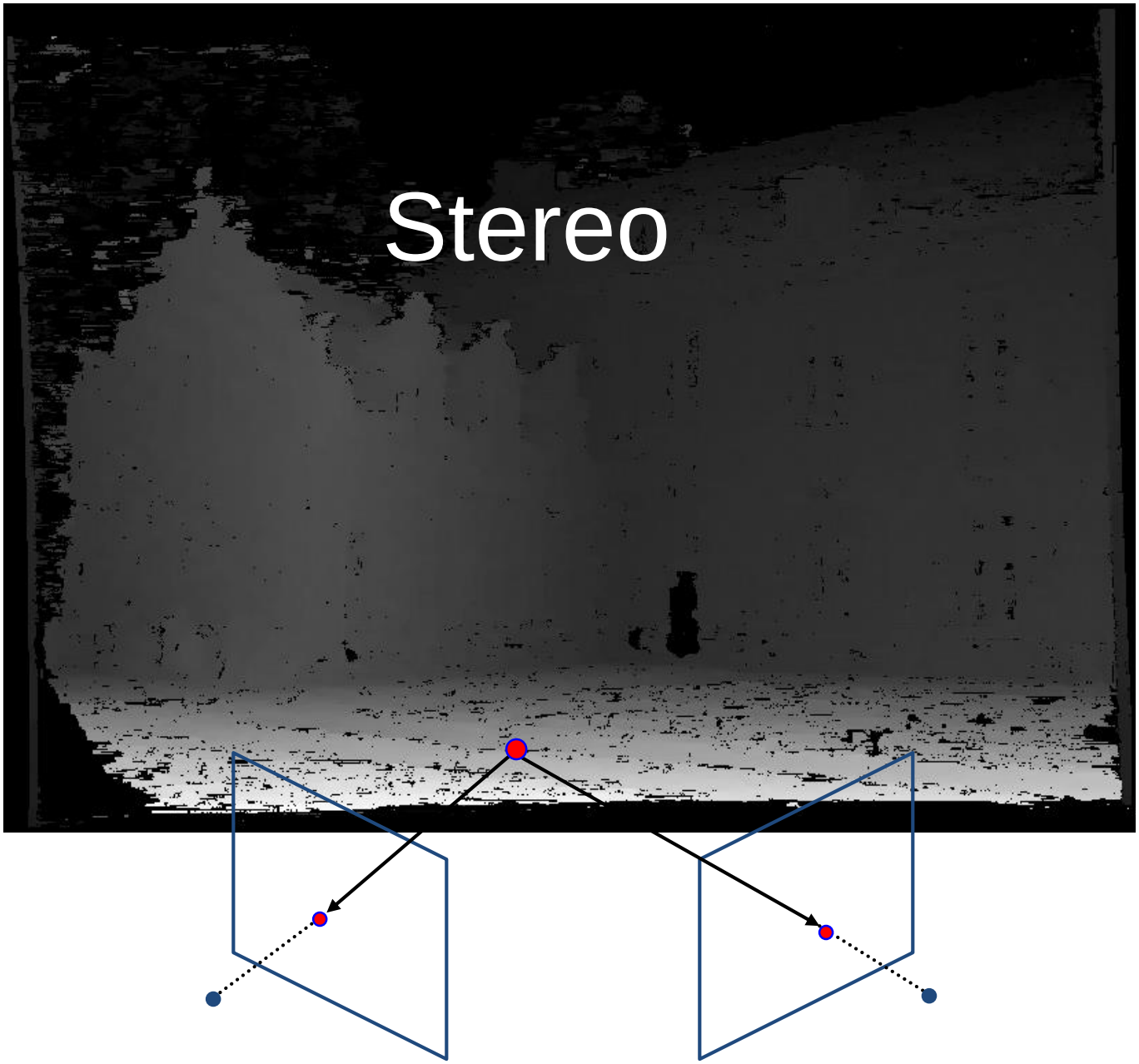
image plane in front



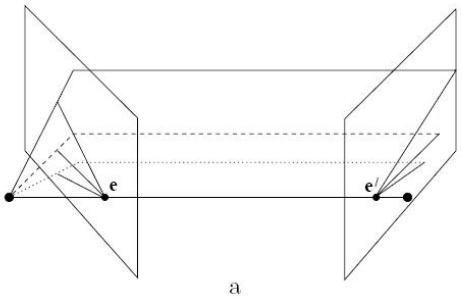
black area
where no pixel
maps to

Source: Steve Seitz

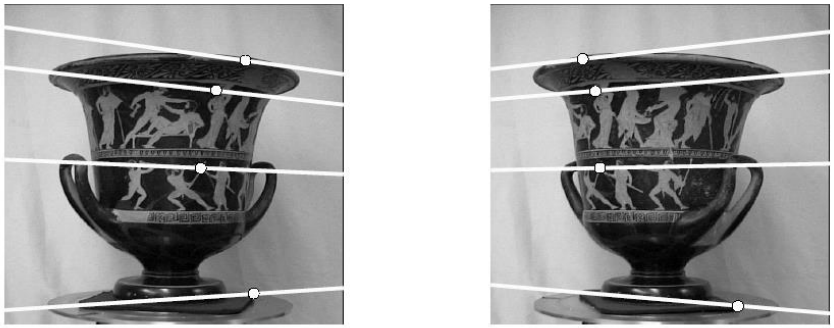
Stereo



Multiple views

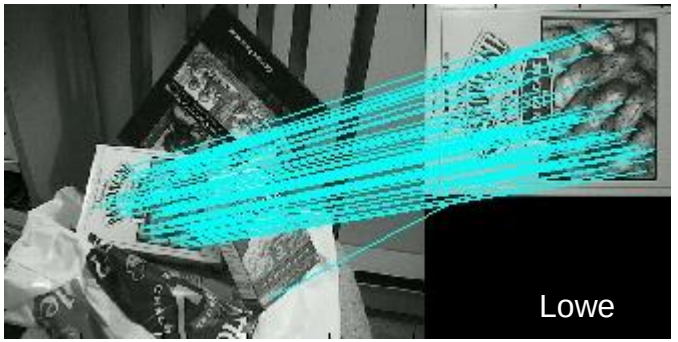


a



Hartley and Zisserman

Multi-view geometry, matching, invariant features, stereo vision



Kristen Grauman

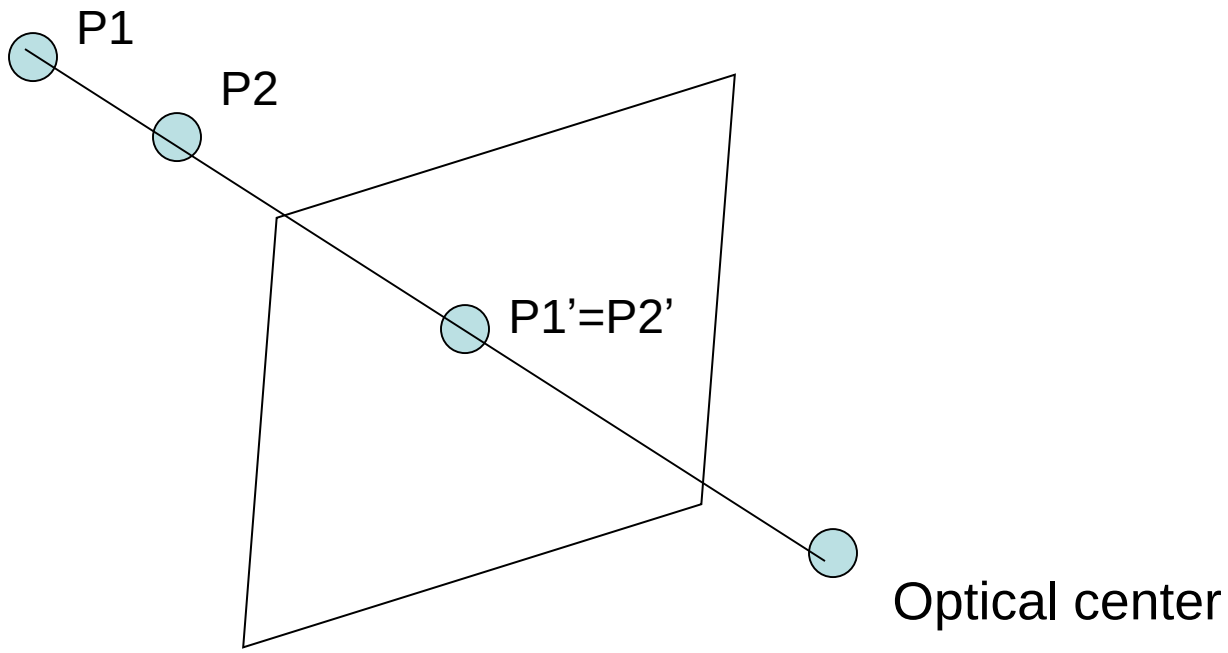
Why multiple views?

- Structure and depth are inherently ambiguous from single views.

Images from Lana Lazebnik

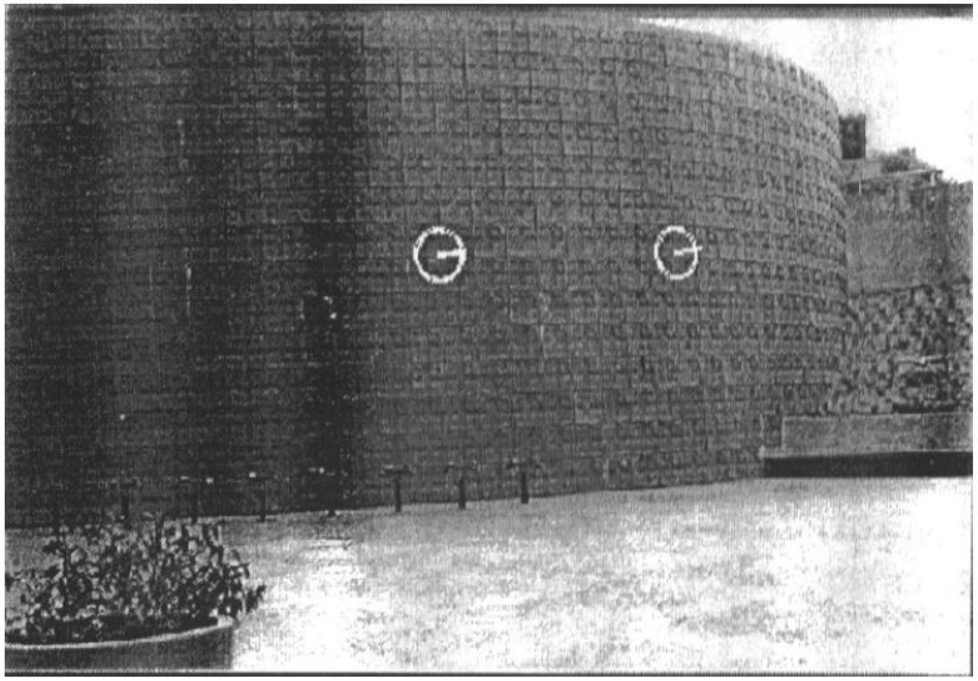
Why multiple views?

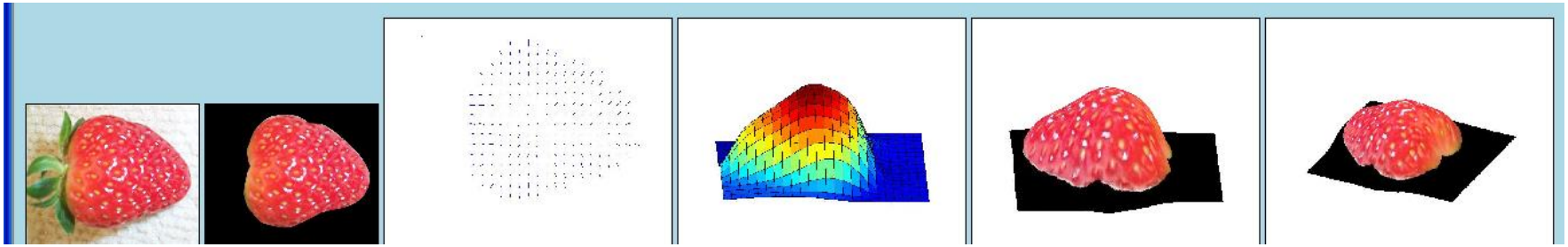
- Structure and depth are inherently ambiguous from single views.



- What cues help us to perceive 3d shape and depth?

Texture (single image)



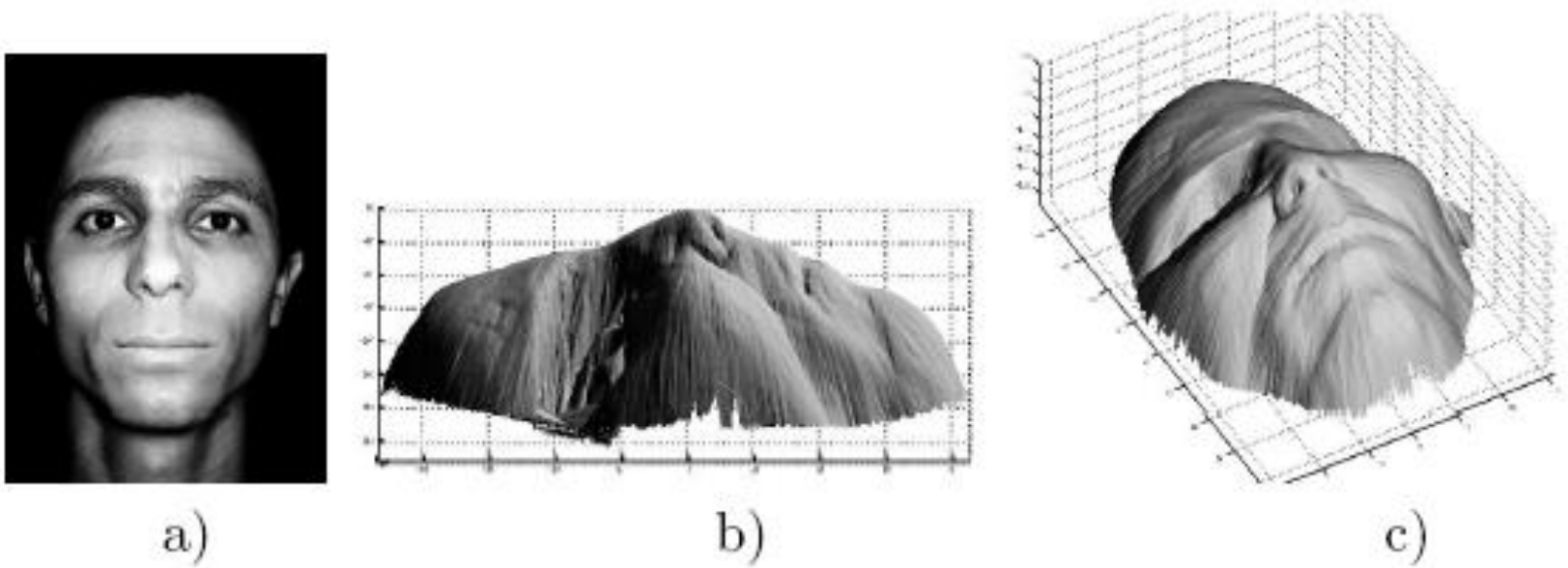


[From A.M. Loh. The recovery of 3-D structure using visual texture patterns. PhD thesis]

Perspective effects (single image)

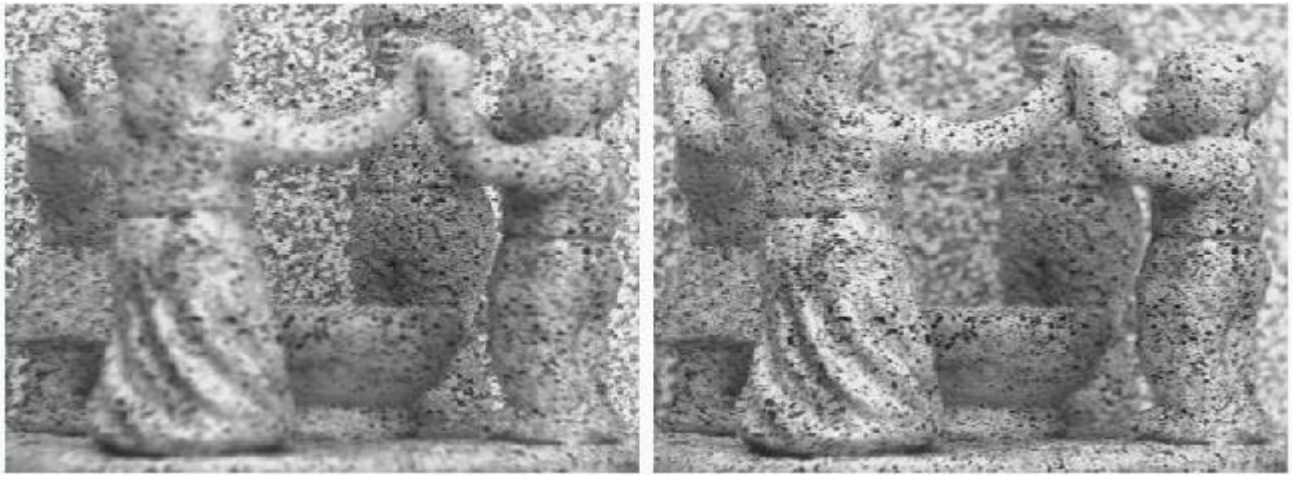
Image credit: S. Seitz

Shading (single image)

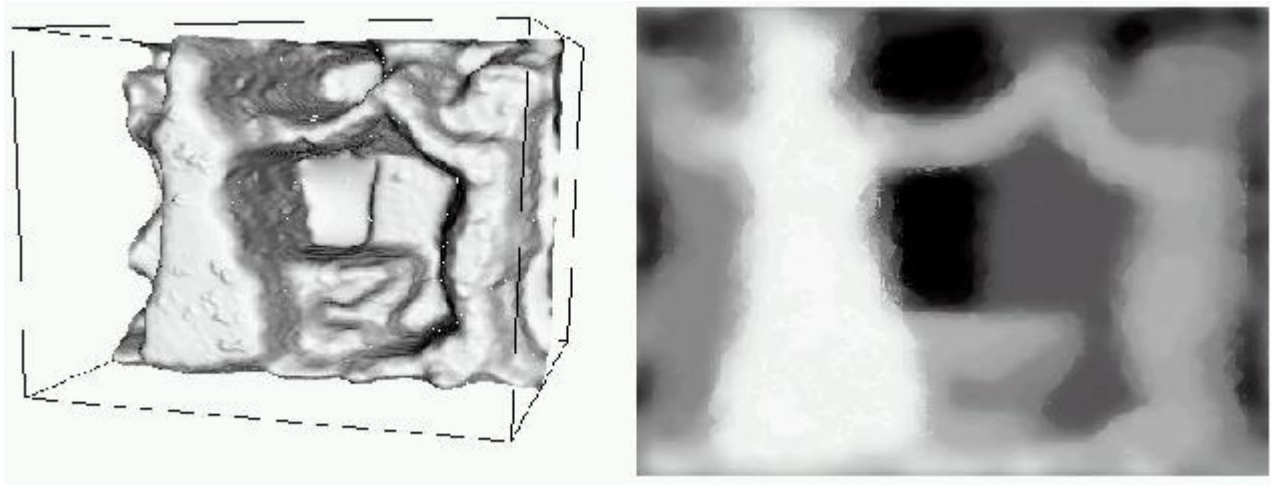


[Figure from Prados & Faugeras 2006]

Focus/defocus (single image)



Images from same point of view, different camera parameters



3d shape / depth estimates

[figs from H. Jin and P. Favaro, 2002]

Motion (**multiple** images)

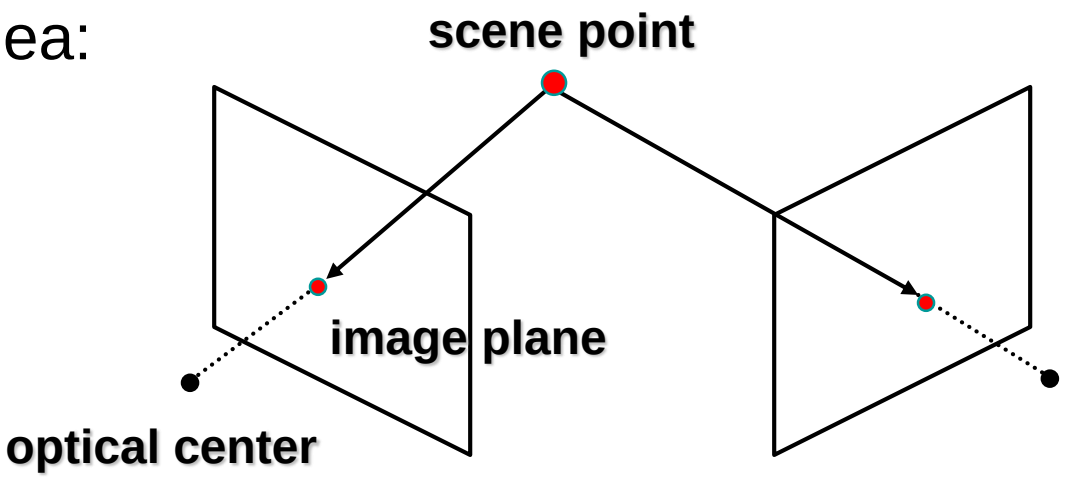
Figures from L. Zhang

http://www.brainconnection.com/teasers/?main=illusion/motion-shape

Estimating scene shape

- "Shape from X": Shading, Texture, Focus, Motion…
- **Stereo**:
  - shape from "motion" between two views
  - infer 3d shape of scene from two (multiple) images from different viewpoints

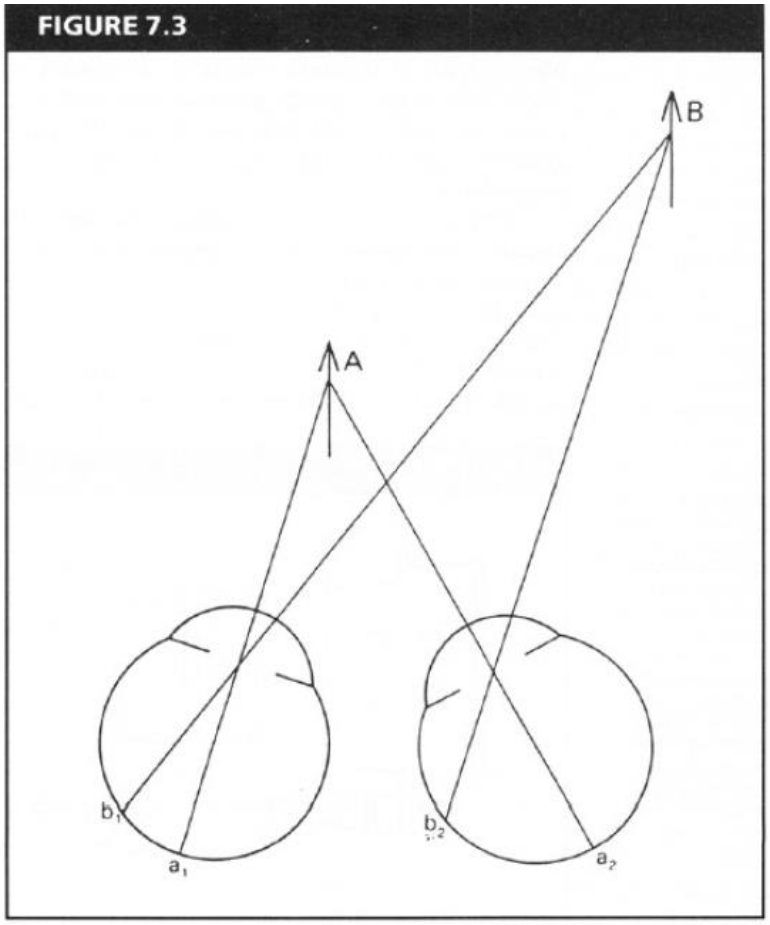
Main idea:



Outline

- Human stereopsis
- Stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras

Human stereopsis: disparity



From Bruce and Green, Visual Perception, Physiology, Psychology and Ecology

Disparity occurs when eyes fixate on one object; others appear at different visual angles

Stereo photography and stereo viewers

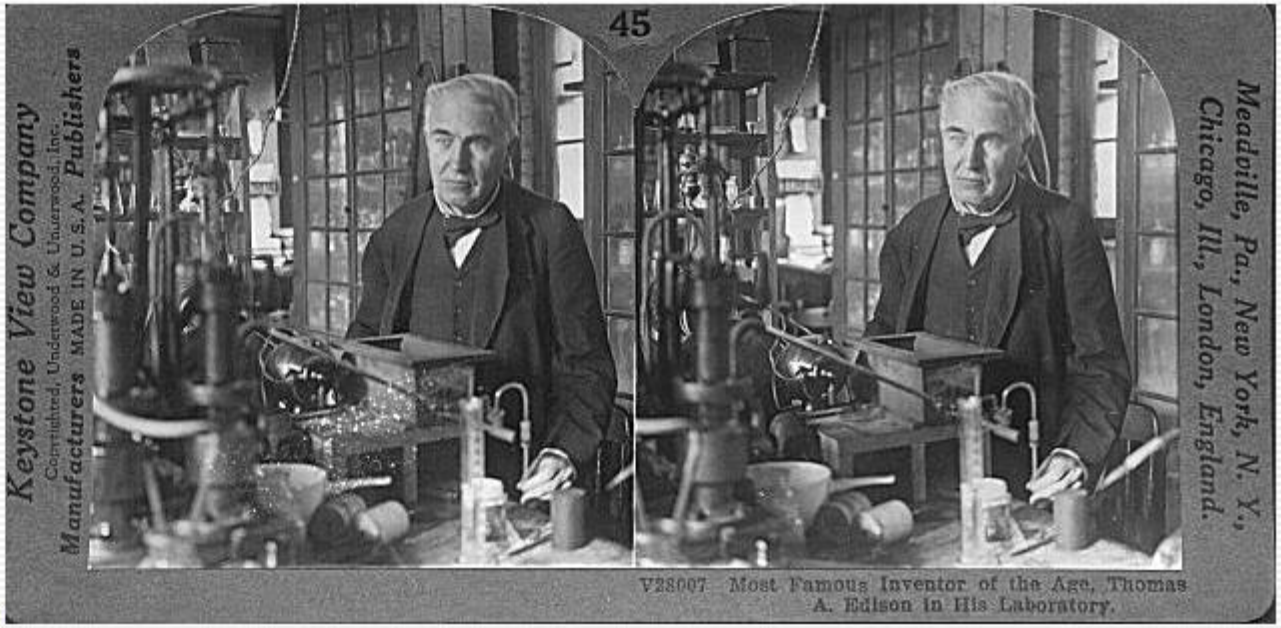
Take two pictures of the same subject from two slightly different viewpoints and display so that each eye sees only one of the images.

Invented by Sir Charles Wheatstone, 1838

Image from fisher-price.com

3D anaglyphs

© Copyright 2001 Johnson-Shaw Stereoscopic Museum



http://www.johnsonshawmuseum.org

Autostereograms

Exploit disparity as depth cue using single image.

(Single image random dot stereogram, Single image stereogram)

Images from magiceye.com

Autostereograms

Images from magiceye.com

Outline

- Human stereopsis
- Stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras
- Stereo solutions
  - Correspondences
  - Additional constraints

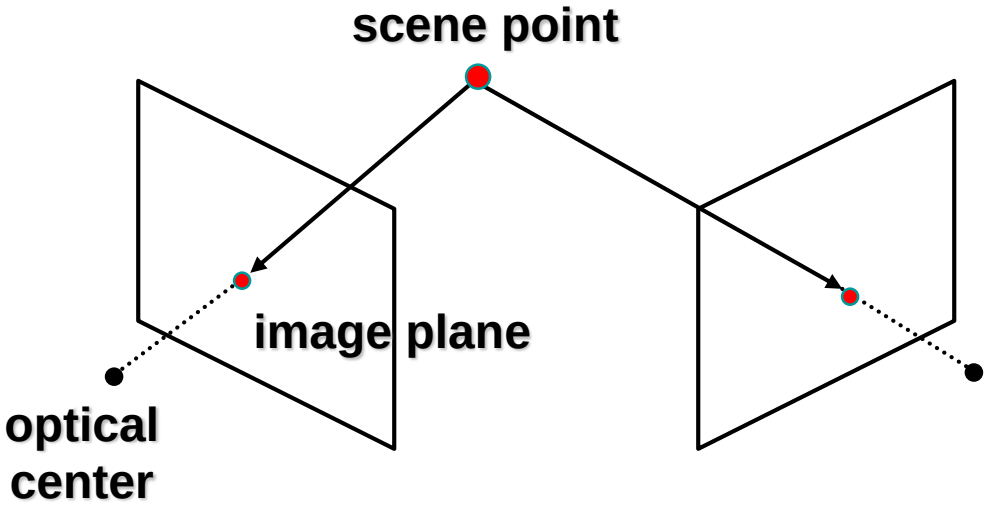
Stereo vision

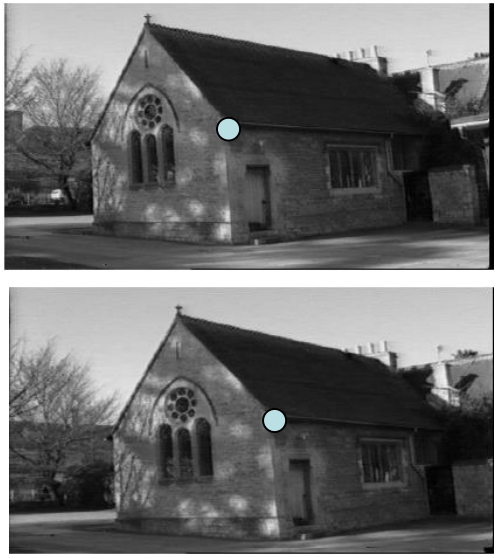
Two cameras, simultaneous views

Single moving camera and static scene

Estimating depth with stereo

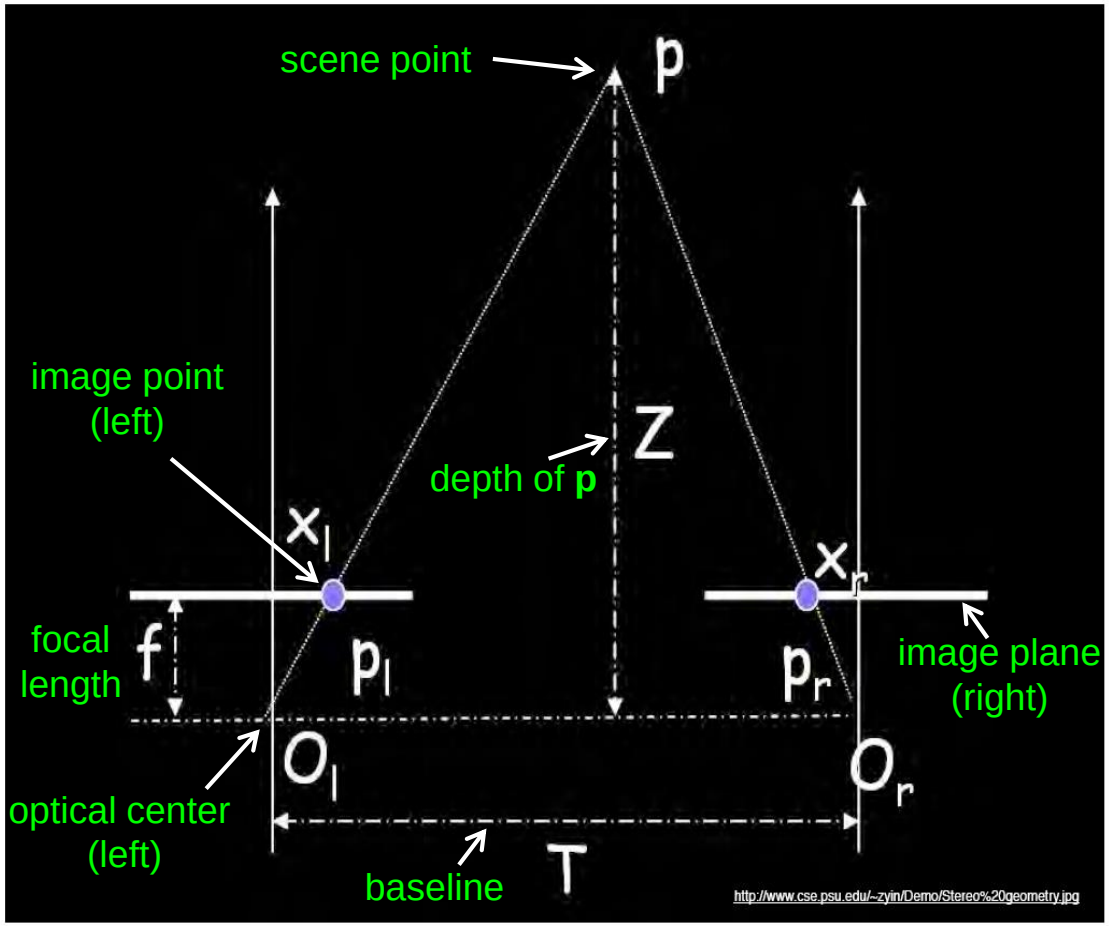
- **Stereo**: shape from “motion” between two views
- We’ll need to consider:
  - Info on camera pose (“calibration”)
  - Image point correspondences





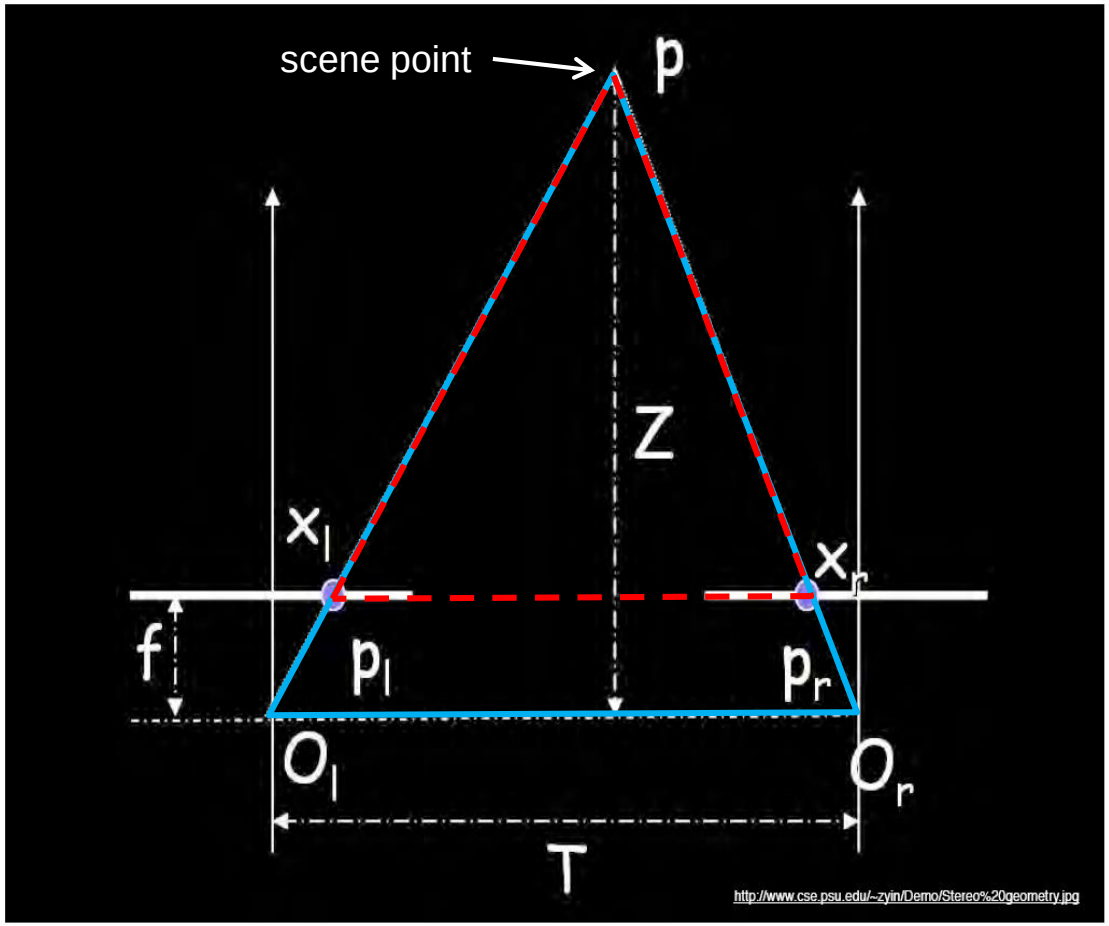
Geometry for a simple stereo system

- Assume parallel optical axes, known camera parameters (calibrated). **What is the expression for Z?**



Geometry for a simple stereo system

- Assume parallel optical axes, known camera parameters (calibrated). **What is the expression for Z?** simple triangulation



similar triangles:

- (p_l, P, p_r) and (O_l, P, O_r):

$$\frac{T + x_r - x_l}{Z - f} = \frac{T}{Z}$$

$$Z = f\frac{T}{x_l - x_r}$$

disparity

Depth from disparity

image I(x,y)

Disparity map D(x,y)

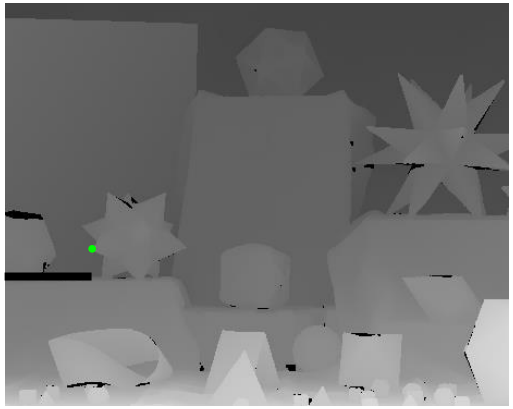


image I´(x´,y´)

(x´,y´) = (x-D(x,y), y)

So if we could find the **corresponding points** in two images, we could **estimate relative depth**…

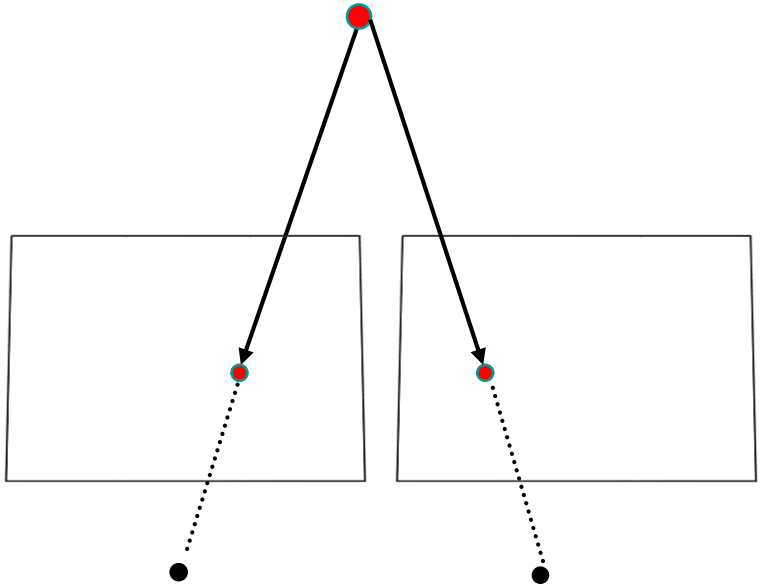
$$Z = f \frac{T}{x_l - x_r}$$

Outline

- Human stereopsis
- Stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras
- Stereo solutions
  - Correspondences
  - Additional constraints

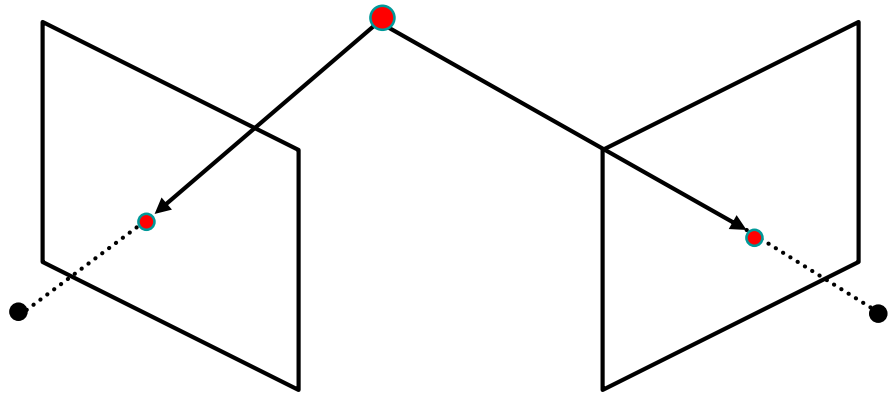
General case, with calibrated cameras

- The two cameras need not have parallel optical axes.



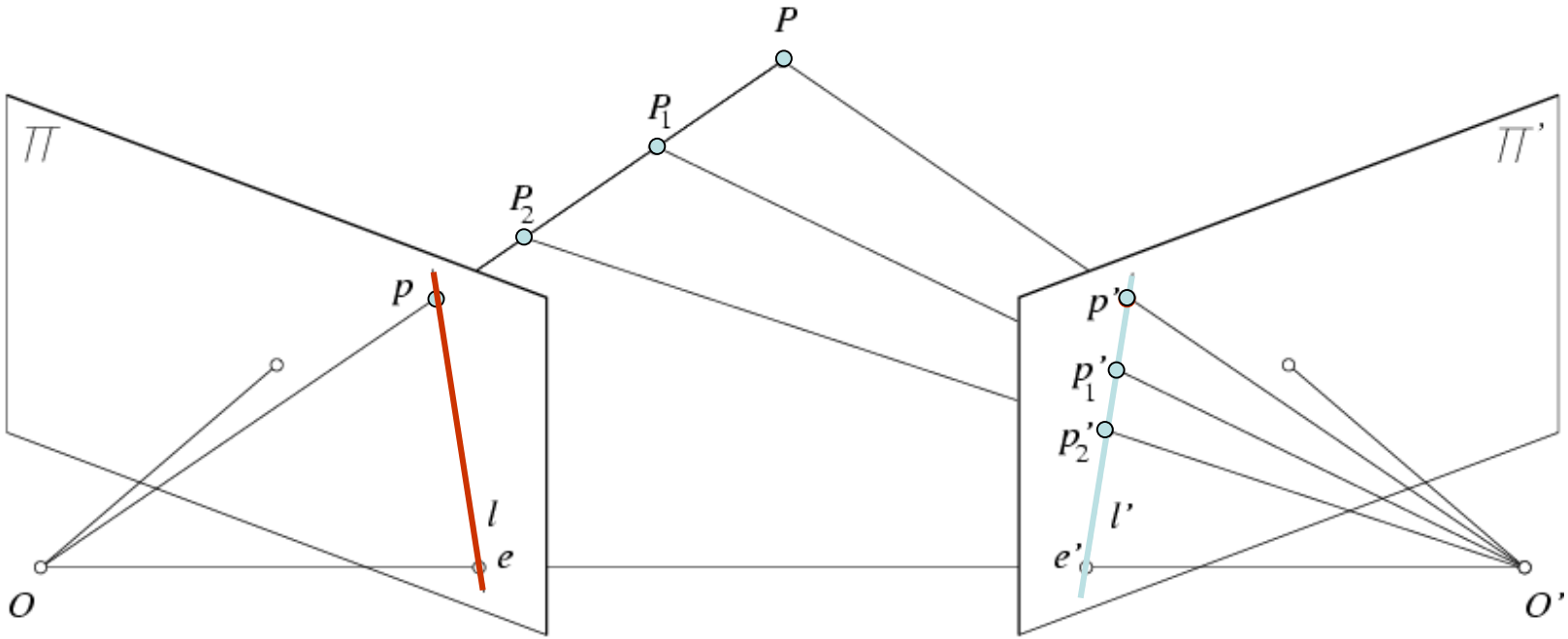
rectified

vs.



general

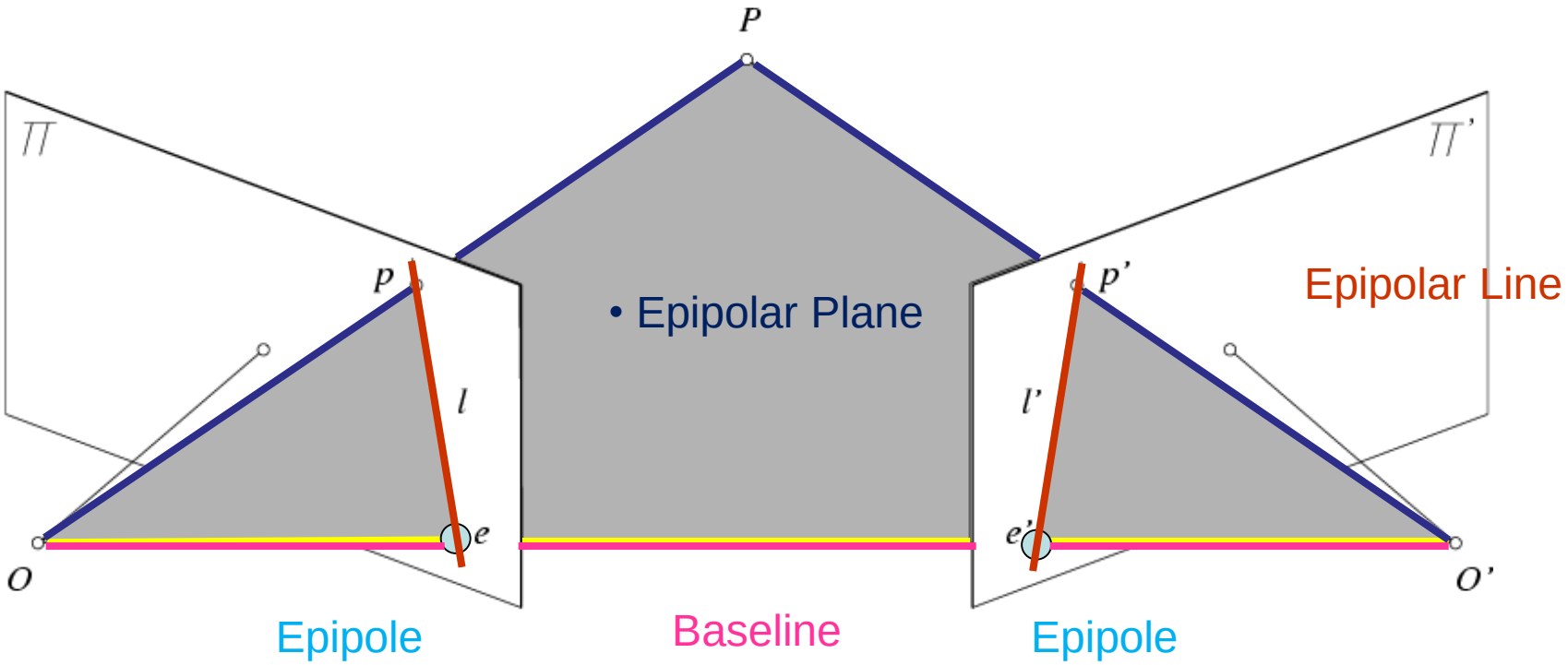
Epipolar constraint



Geometry of two views constrains where the corresponding pixel for some image point in the first view must occur in the second view.

- It must be on the line carved out by a plane connecting the world point and optical centers.

Epipolar geometry



Epipolar geometry: terms

- **Baseline**: line joining the camera centers
- **Epipole**: point of intersection of baseline with image plane
- **Epipolar plane**: plane containing baseline and world point
- **Epipolar line**: intersection of epipolar plane with the image plane

- All epipolar lines intersect at the epipole
- An epipolar plane intersects the left and right image planes in epipolar lines

Why is the epipolar constraint useful?

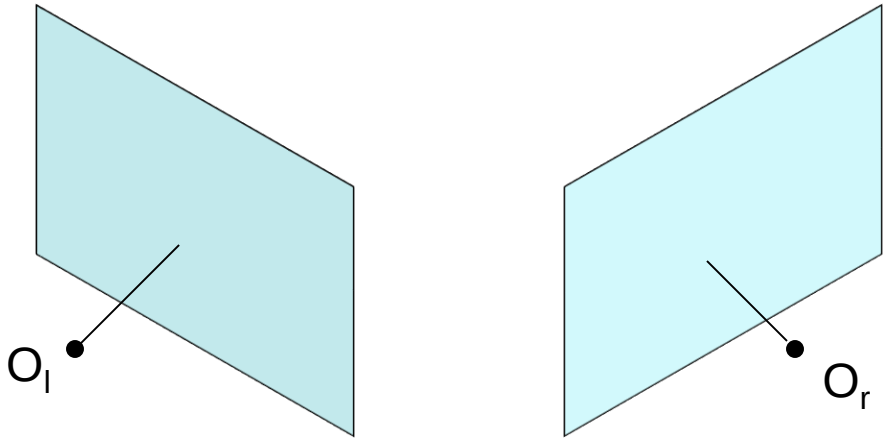
Epipolar constraint

This is useful because it reduces the correspondence problem to a 1D search along an epipolar line.

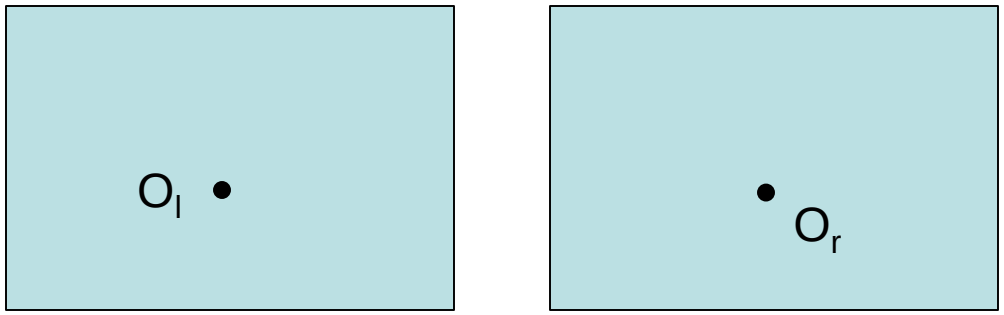
Image from Andrew Zisserman

What do the epipolar lines look like?

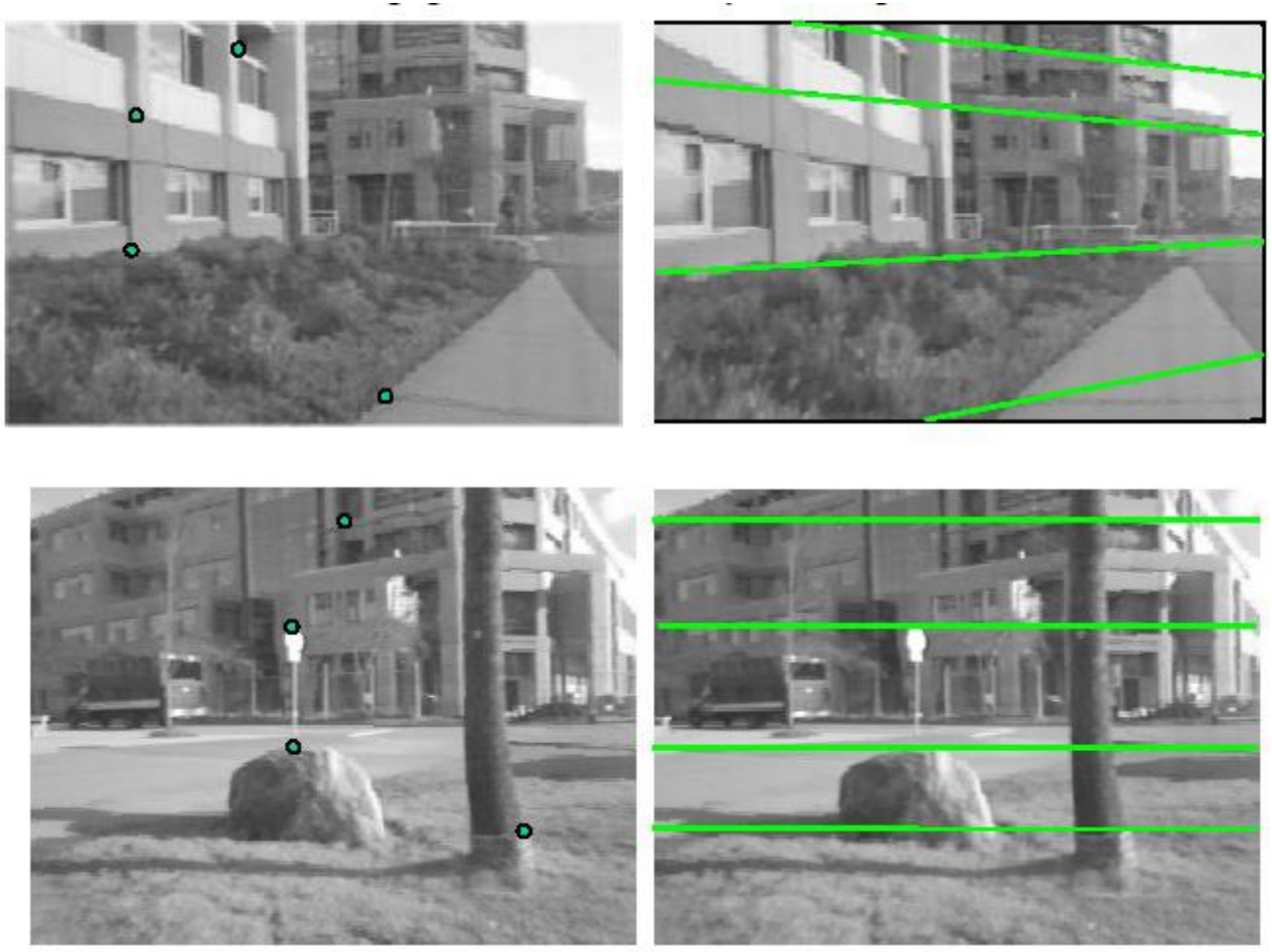
1.



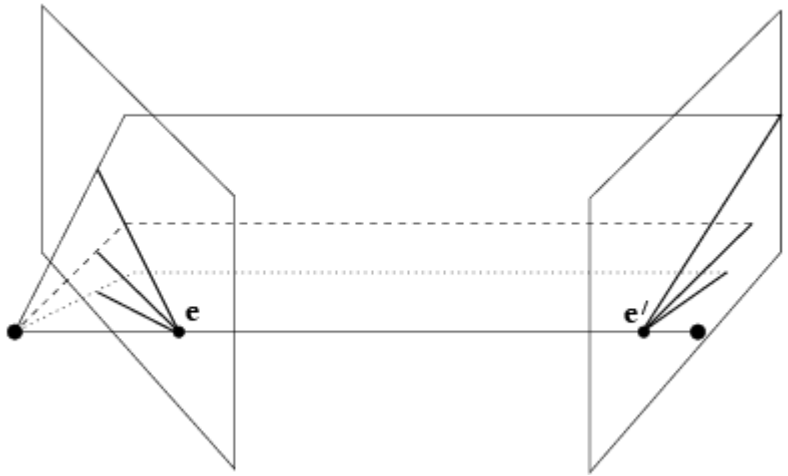
2.



Example



Example: converging cameras



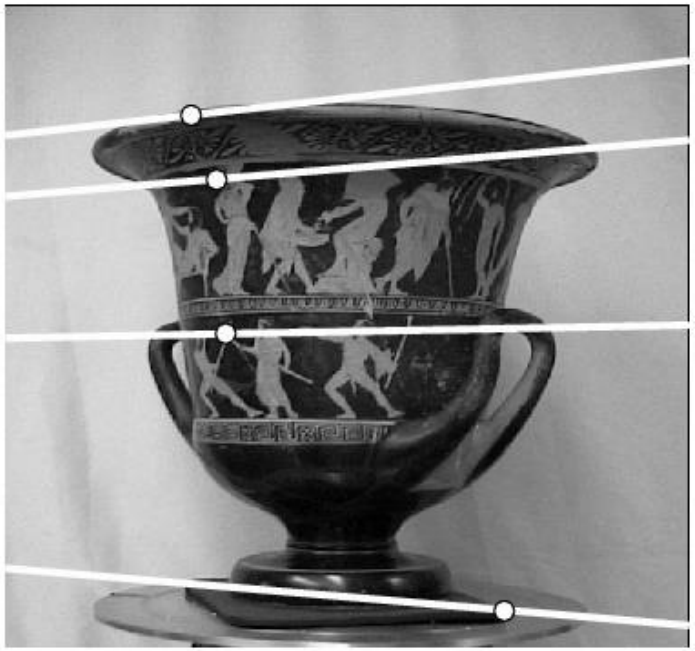
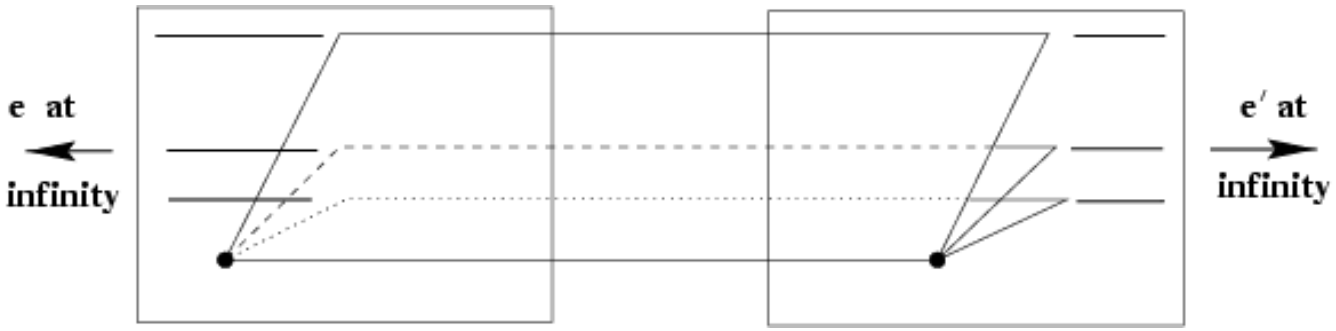


Figure from Hartley & Zisserman

Example: parallel cameras



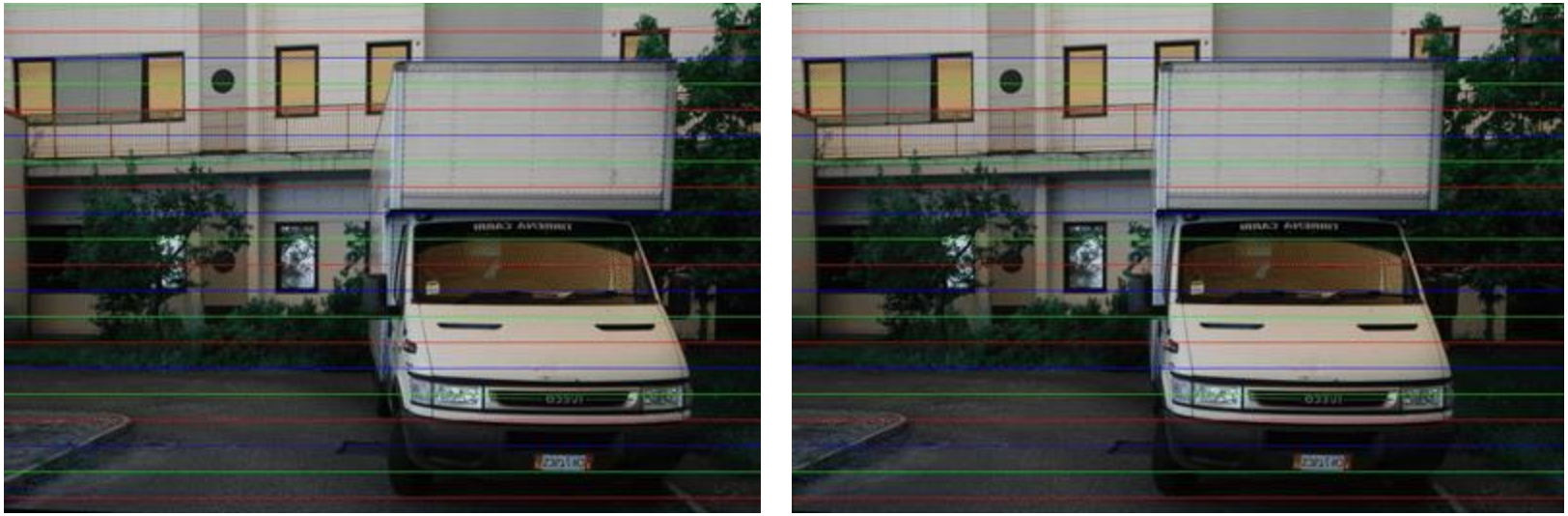
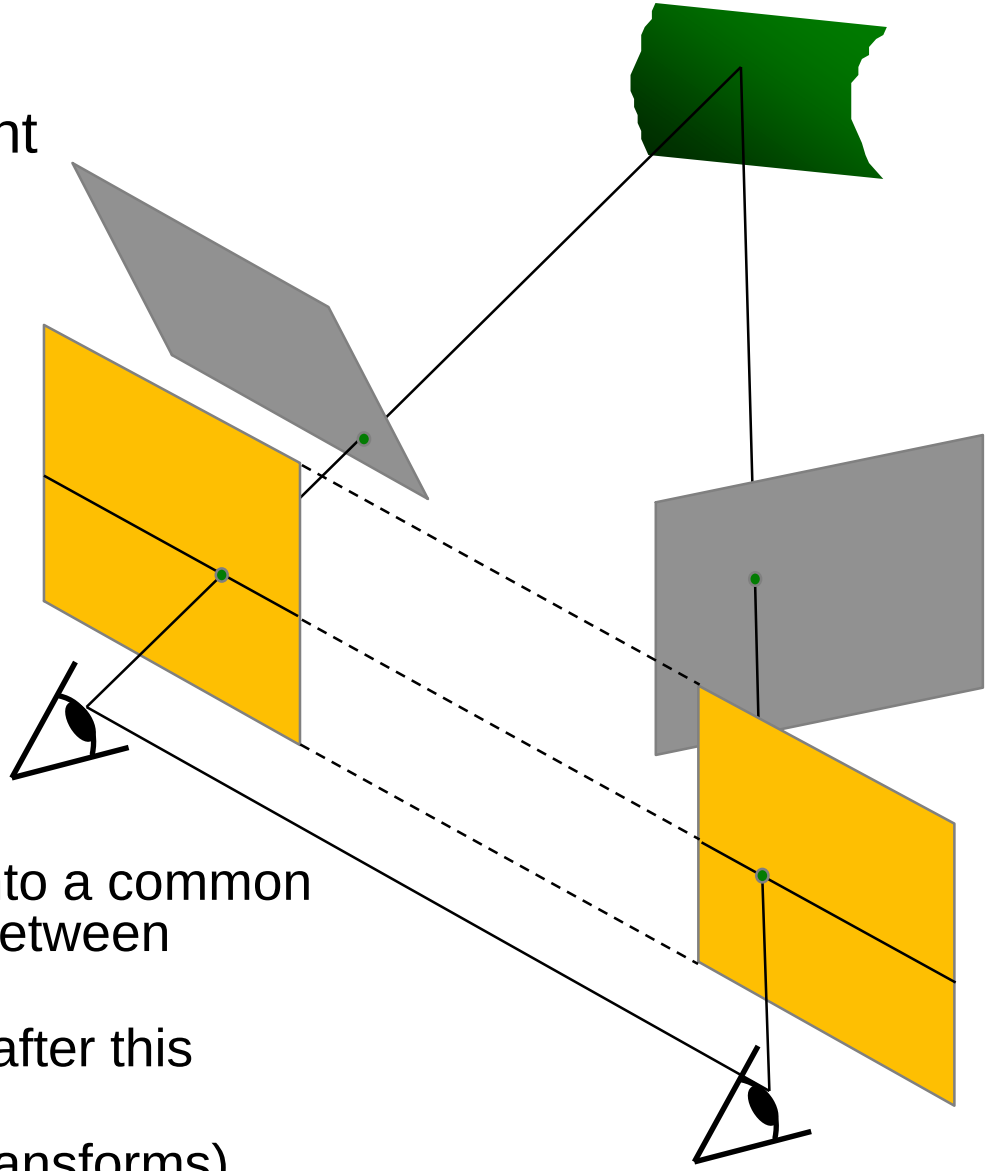


Figure from Hartley & Zisserman

Stereo image rectification

In practice, it is convenient if image scanlines (rows) are the epipolar lines.

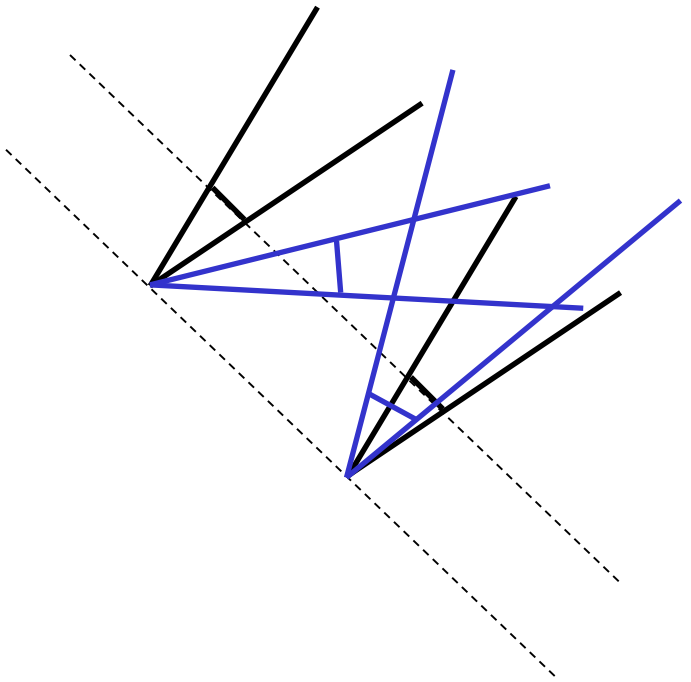


- reproject image planes onto a common plane parallel to the line between optical centers
- pixel motion is horizontal after this transformation
- two homographies (3x3 transforms), one for each input image reprojection

Slide credit: Li Zhang

Stereo image rectification

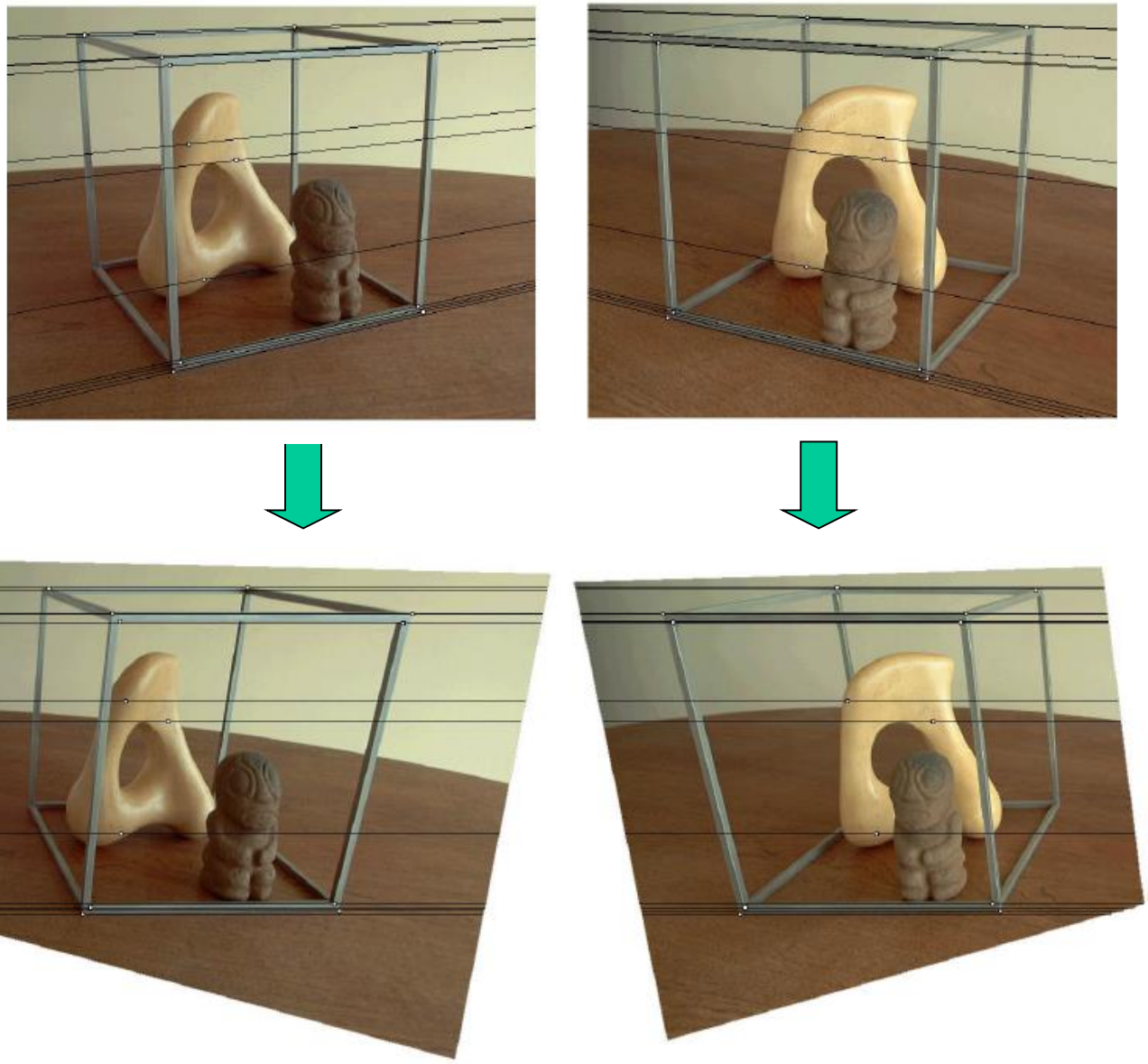
In practice, it is convenient
if image scanlines (rows)
are the epipolar lines.



- reproject image planes onto a common plane parallel to the line between optical centers
- pixel motion is horizontal after this transformation
- two homographies (3x3 transforms), one for each input image reprojection

Slide credit: Li Zhang

Stereo image rectification: example



Source: Alyosha Efros

Summary so far

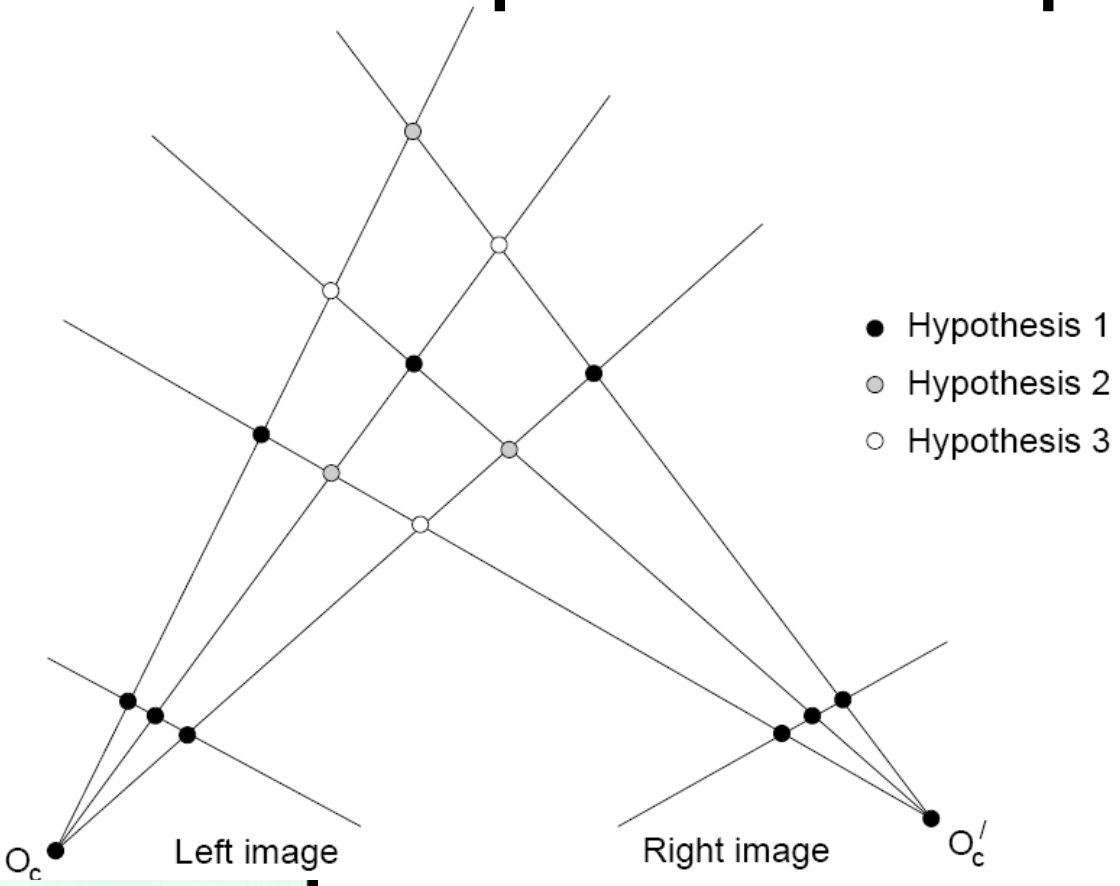
- Depth from stereo: main idea is to triangulate from corresponding image points.
- Epipolar geometry defined by two cameras
  - We've assumed known extrinsic parameters relating their poses
- Epipolar constraint limits where points from one view will be imaged in the other
  - Makes search for correspondences quicker

- **Terms**: epipole, epipolar plane / lines, disparity, rectification, baseline

Outline

- Human stereopsis
- Stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras
- Stereo solutions
  - Correspondences
  - Additional constraints

Correspondence problem



Multiple match hypotheses satisfy epipolar constraint, but which is correct?

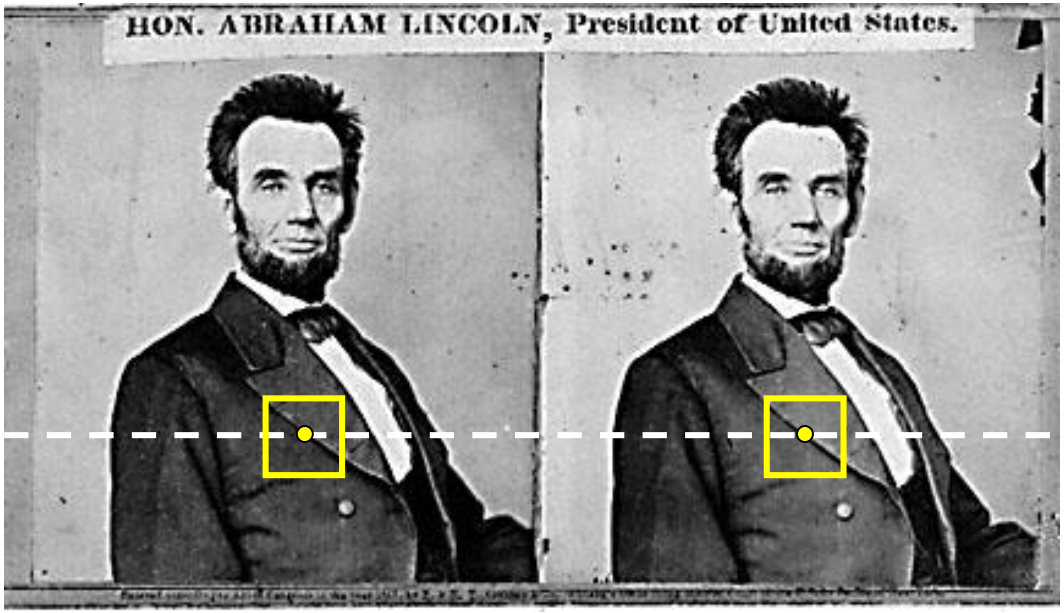
Figure from Gee & Cipolla 1999

Correspondence problem

- Beyond the hard constraint of epipolar geometry, there are “soft” constraints to help identify corresponding points
  - Similarity
  - Uniqueness
  - Ordering
  - Disparity gradient

- To find matches in the image pair, we will assume
  - Most scene points visible from both views
  - Image regions for the matches are similar in appearance

Dense correspondence search



For each epipolar line

For each pixel / window in the left image

- compare with every pixel / window on same line in right image
- pick position with best match (e.g., SSD, NCC, Census)

Adapted from Li Zhang

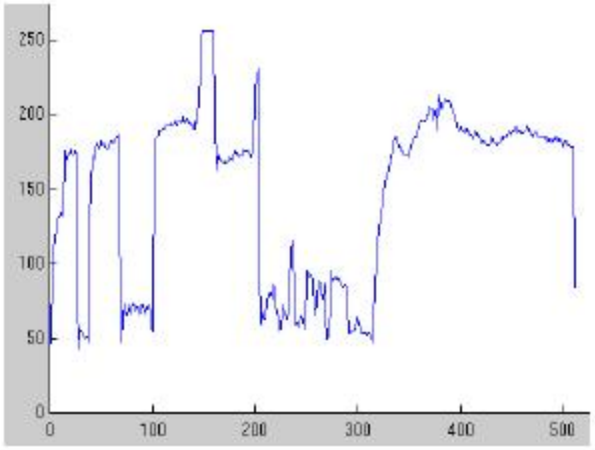
Correspondence problem

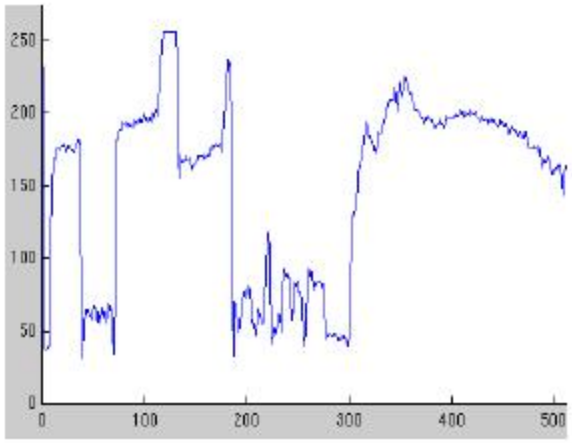
Parallel camera example: epipolar lines are corresponding image scanlines

Source: Andrew Zisserman

Correspondence problem

Intensity profiles





- Clear correspondence between intensities, but also noise and ambiguity

Source: Andrew Zisserman

Correspondence problem

Neighborhoods of corresponding points are similar in intensity patterns.

Source: Andrew Zisserman

Normalized cross correlation

subtract mean: $A \leftarrow A - < A >, B \leftarrow B - < B >$

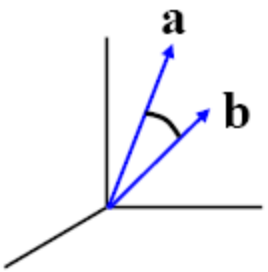
$$\text{NCC} = \frac{\sum_i \sum_j A(i,j)B(i,j)}{\sqrt{\sum_i \sum_j A(i,j)^2}\sqrt{\sum_i \sum_j B(i,j)^2}}$$

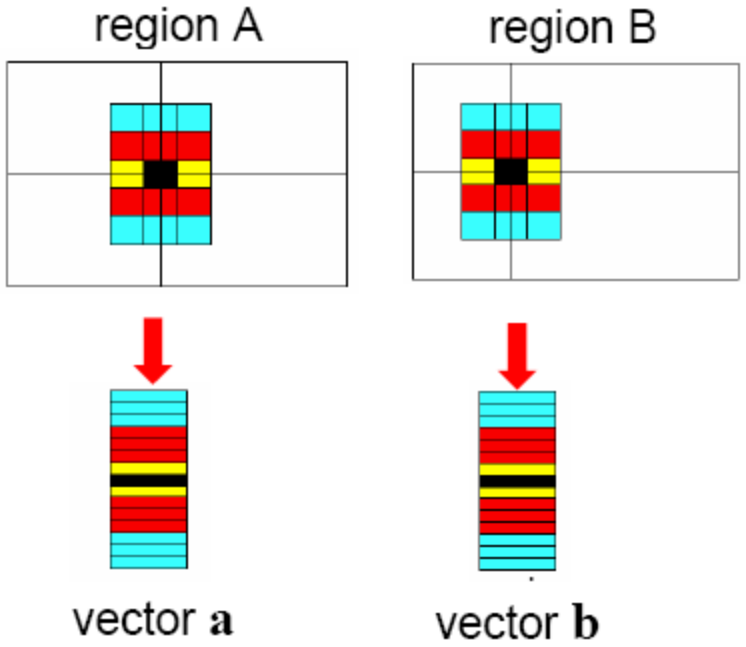
Write regions as vectors

$$\mathbf{A} \rightarrow \mathbf{a}, \quad \mathbf{B} \rightarrow \mathbf{b}$$

$$\text{NCC} = \frac{\mathbf{a}.\mathbf{b}}{|\mathbf{a}||\mathbf{b}|}$$

$$-1 \leq \text{NCC} \leq 1$$





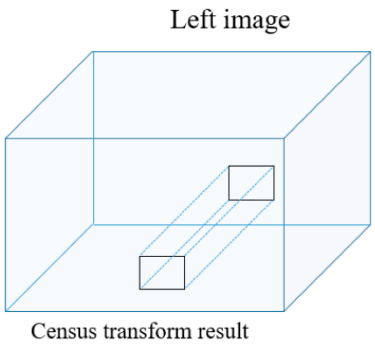
Source: Andrew Zisserman

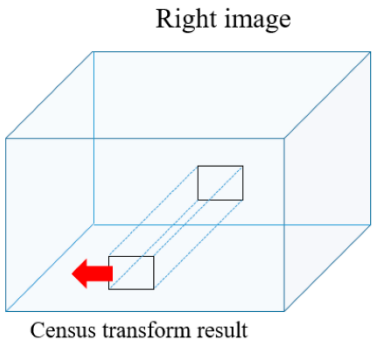
Census Hamming Distance

121	130	26	31	39
109	115	35	60	65
98	102	78	67	45
47	97	36	165	199
30	87	91	134	207

1	1	0	0	0
1	1	0	0	0
1	1	×	0	0
0	1	0	1	1
0	1	1	1	1

1	1	0	0	0	1	1	0	0	0	1	1	0	0	0	1	0	1	1	0	1	1	1	1





Left image

1	1	0	0	0	1	1	0	0	0	1	1	0	0	0	1	0	1	1	0	1	1	1	1

Right image

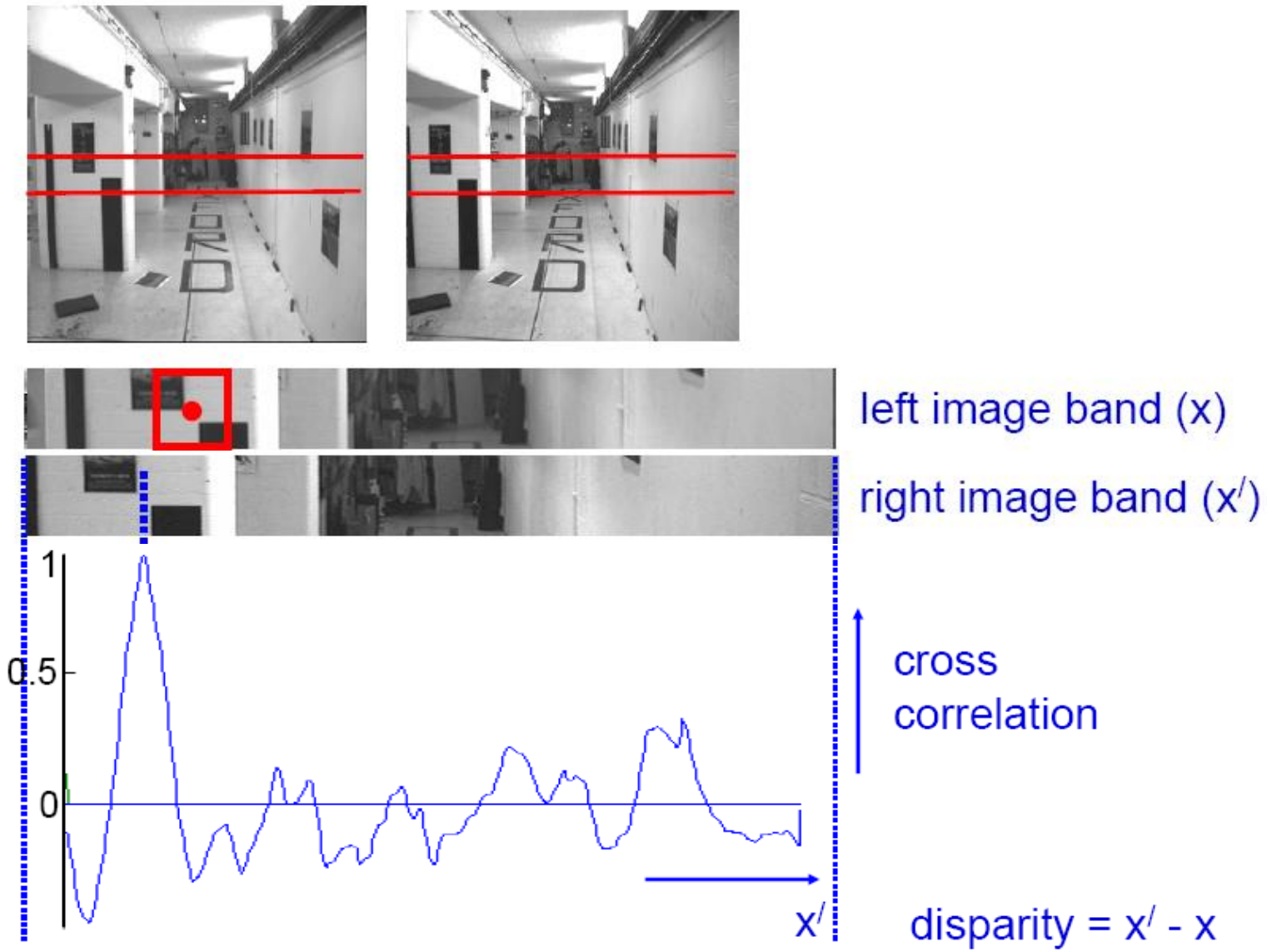
0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	1	0	1	0	0	1	1	0	0

XOR
logical operation

Hamming distance = 7

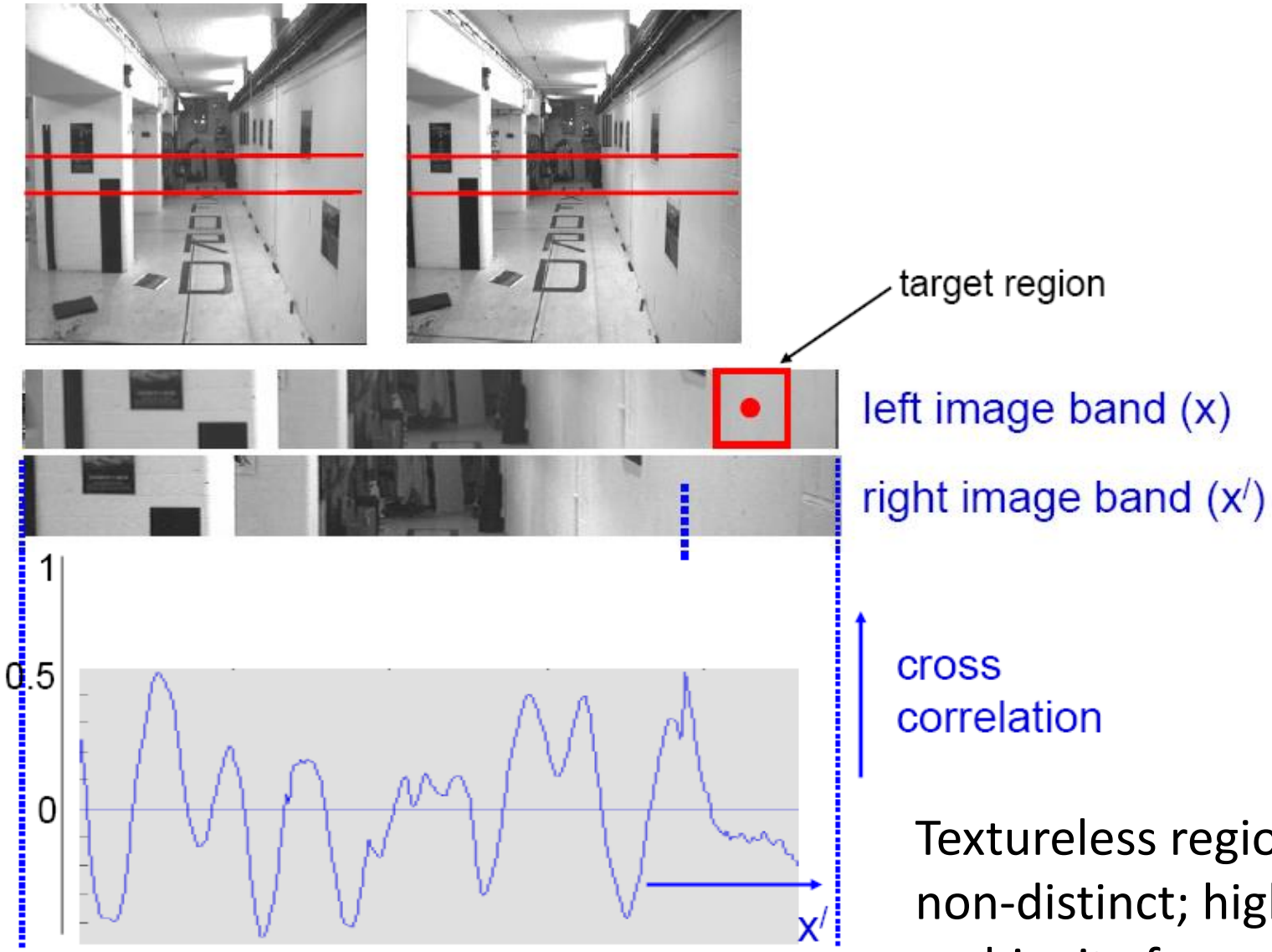
1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1

Correlation-based window matching



Source: Andrew Zisserman

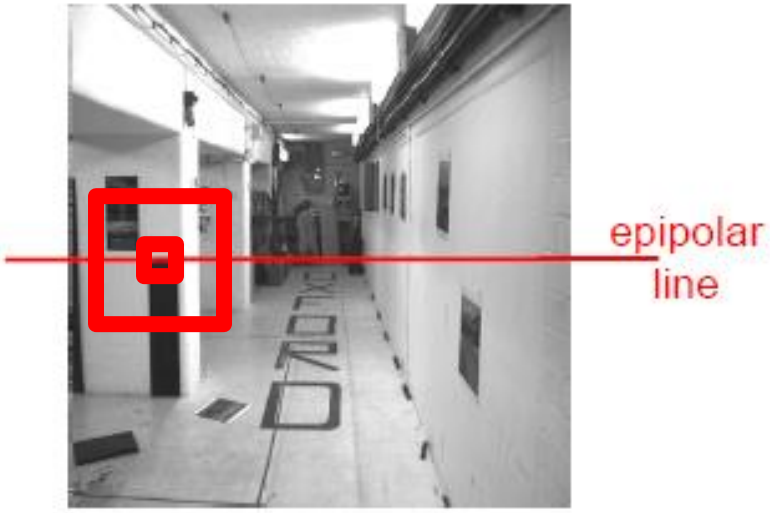
Textureless regions



Textureless regions are non-distinct; high ambiguity for matches.

Source: Andrew Zisserman

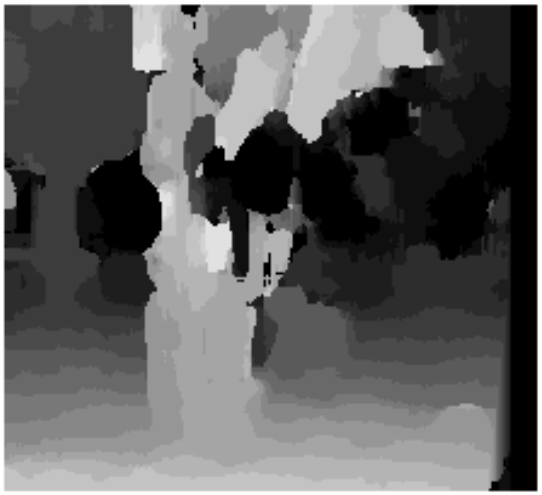
Effect of window size?



Source: Andrew Zisserman

Effect of window size

W = 3

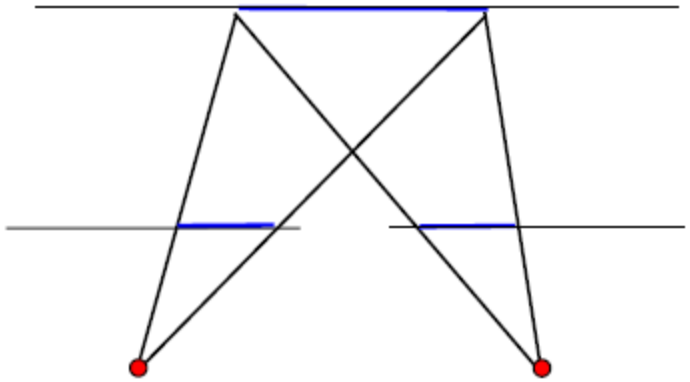


W = 20

Want window large enough to have sufficient intensity variation, yet small enough to contain only pixels with about the same disparity.

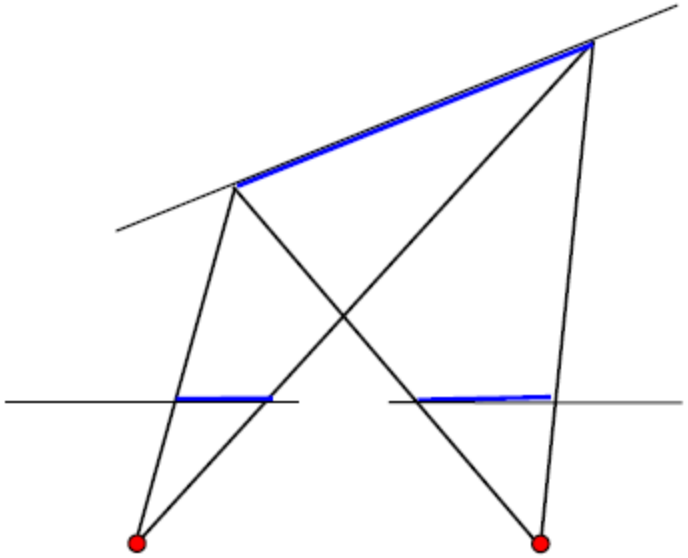
Figures from Li Zhang

Foreshortening effects



fronto-parallel surface

imaged length the same

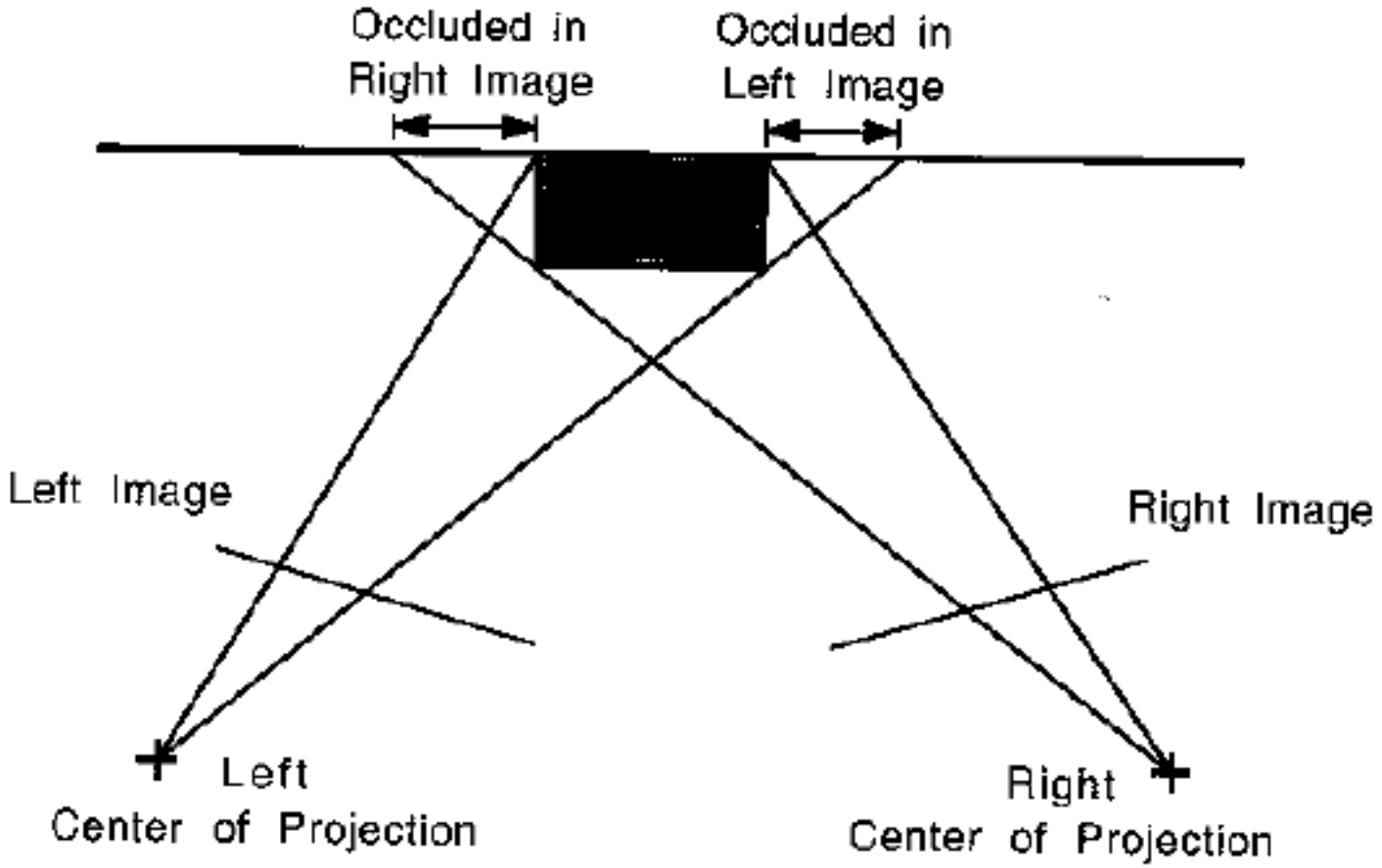


slanting surface

imaged lengths differ

Source: Andrew Zisserman

Occlusion



Slide credit: David Kriegman

Correspondence problem

- Beyond the hard constraint of epipolar geometry, there are “soft” constraints to help identify corresponding points
  - Similarity
  - Uniqueness
  - Disparity gradient
  - Ordering

Uniqueness constraint

- Up to one match in right image for every point in left image

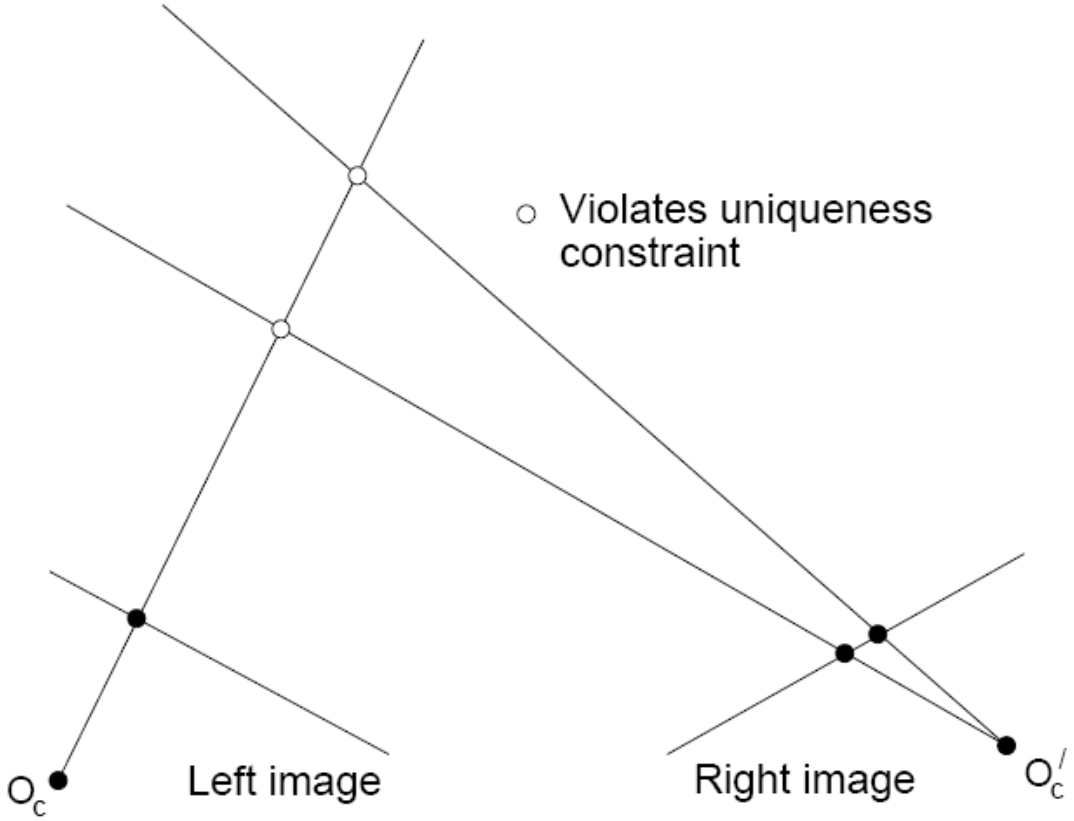
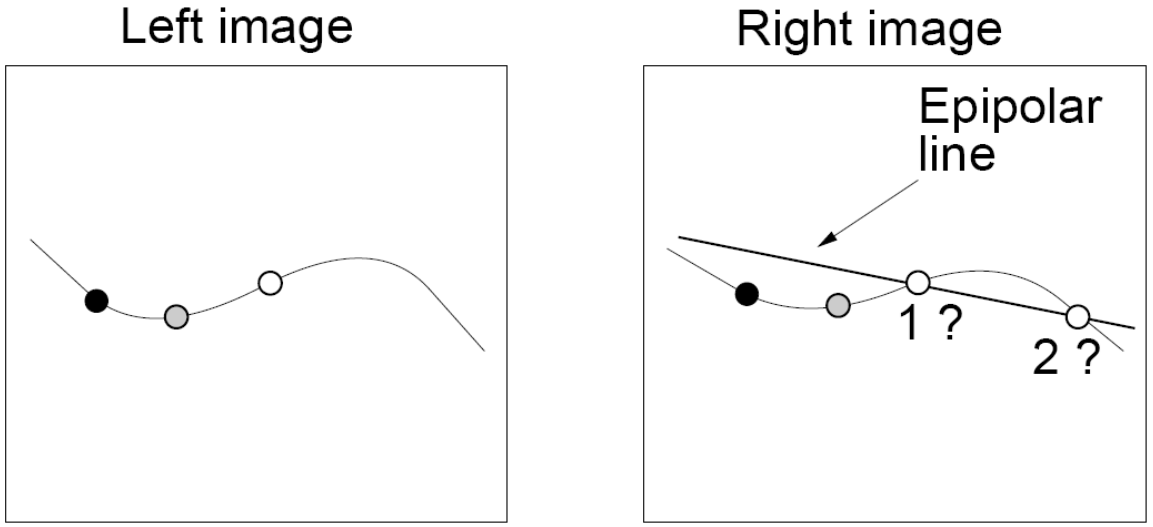


Figure from Gee & Cipolla 1999

Disparity gradient constraint

- Assume piecewise continuous surface, so want disparity estimates to be locally smooth



Given matches ● and ◌, point ○ in the left image must match point 1 in the right image. Point 2 would exceed the disparity gradient limit.

Figure from Gee & Cipolla 1999

Ordering constraint

- Points on **same surface** (opaque object) will be in same order in both views

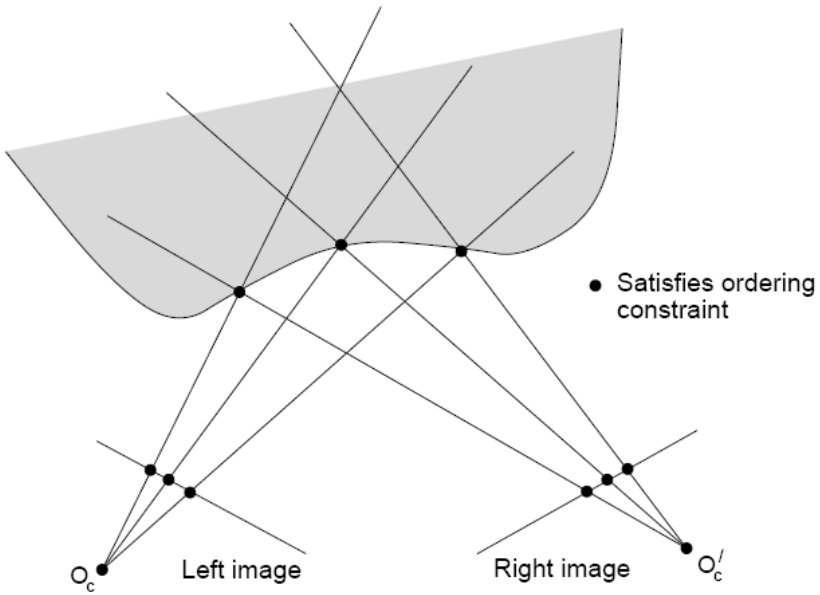


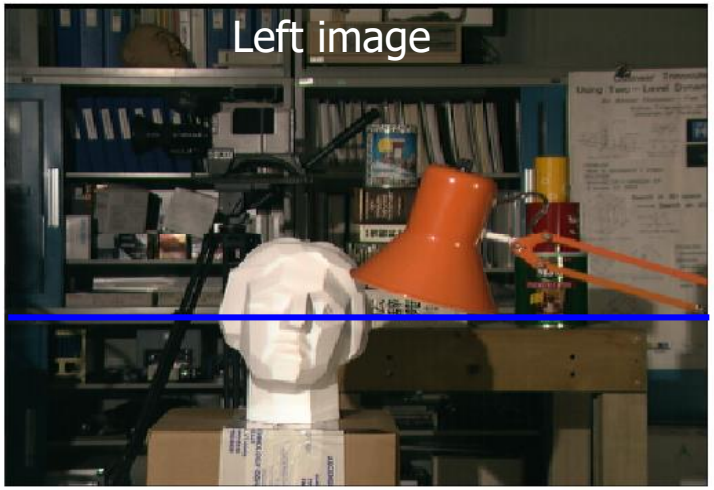
Figure from Gee & Cipolla 1999

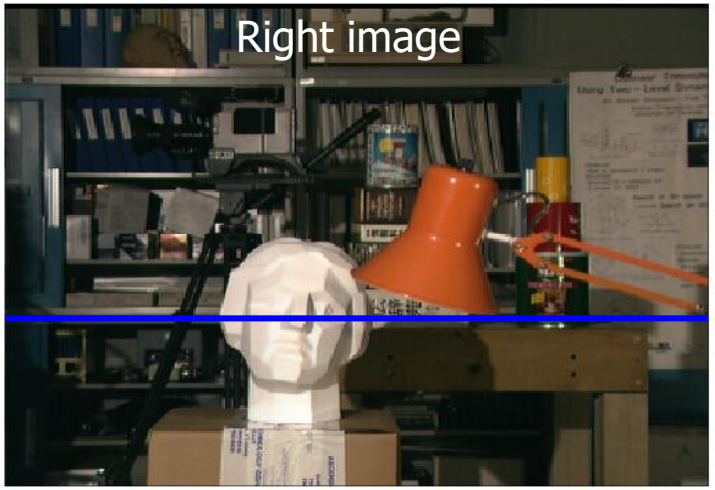
Beyond individual correspondences

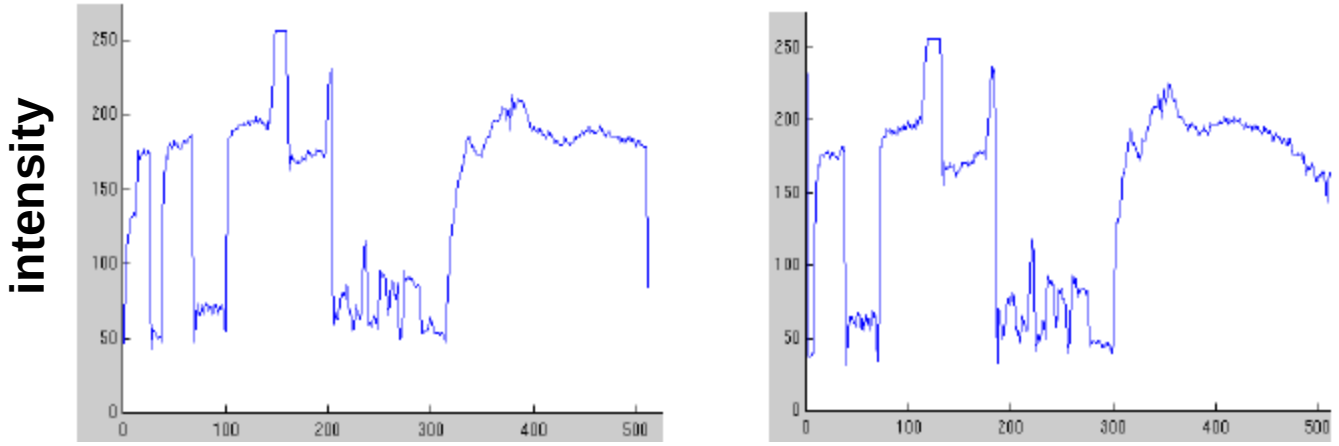
- Optimize correspondence assignments jointly
  - Scanline at a time (dynamic program)
  - Full 2D grid (graph cuts)
  - Many scan lines jointly (semi-global-matching)

Scanline stereo

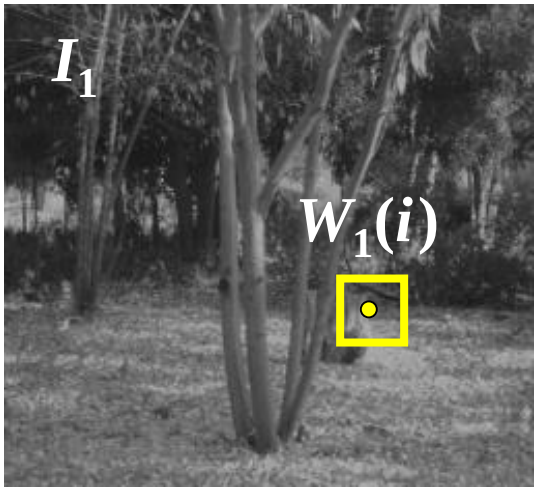
- Try to coherently match pixels on the entire scanline
- Different scanlines are still optimized independently

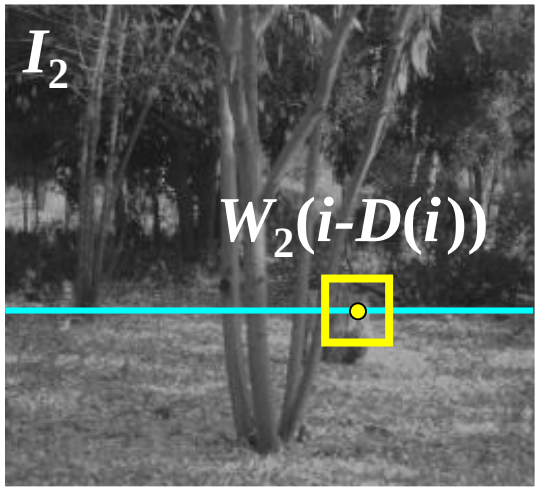


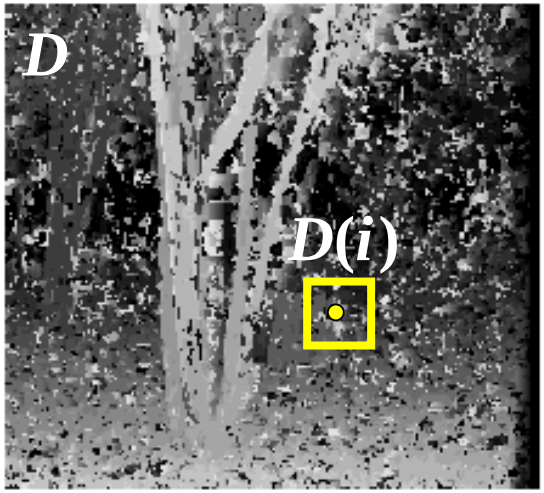




Stereo matching as energy minimization







$$E = \alpha E_{\text{data}}(I_1, I_2, D) + \beta E_{\text{smooth}}(D)$$

$$E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i - D(i))\right)^2$$

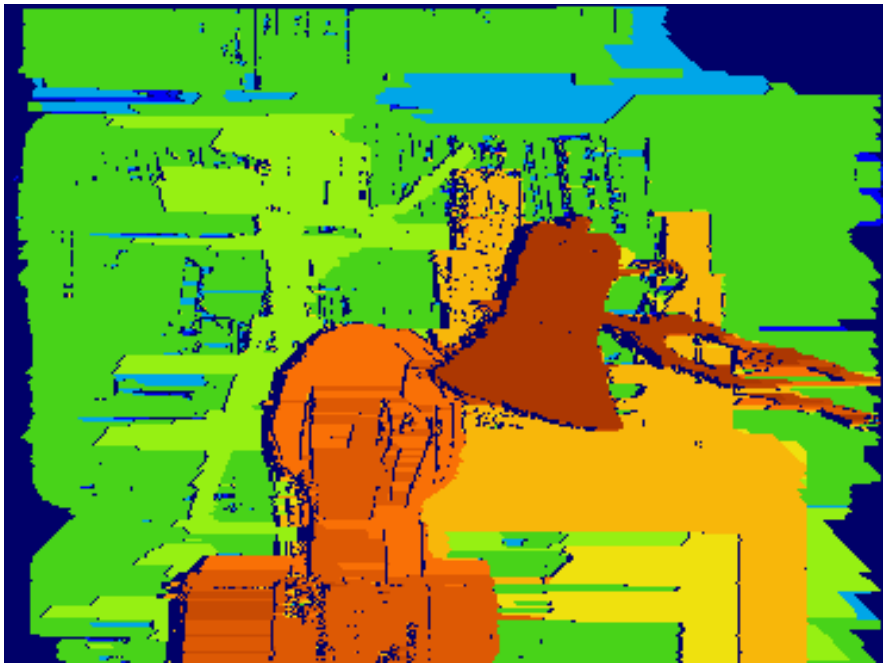
$$E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \rho\left(D(i) - D(j)\right)$$

- can be minimized in 1D (line by line) using ***dynamic programming***
  - Ohta & Kanade ’85, Cox et al. ‘96
- can be minimized in 2D (entire image) using ***graph cuts***
  - Veksler & Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Source: Steve Seitz

Coherent stereo on 2D grid

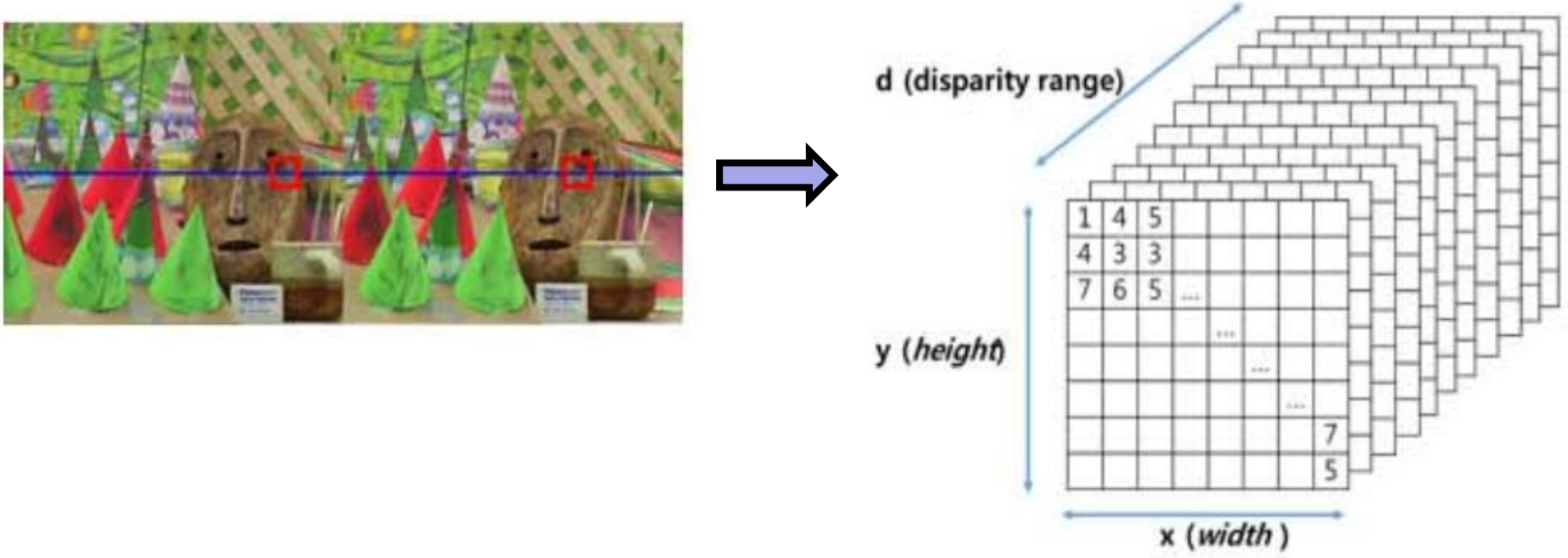
- Scanline stereo generates streaking artifacts



- Can’t use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid
- Graph-Cuts are the main tool to optimize in 2D

A typical pipeline

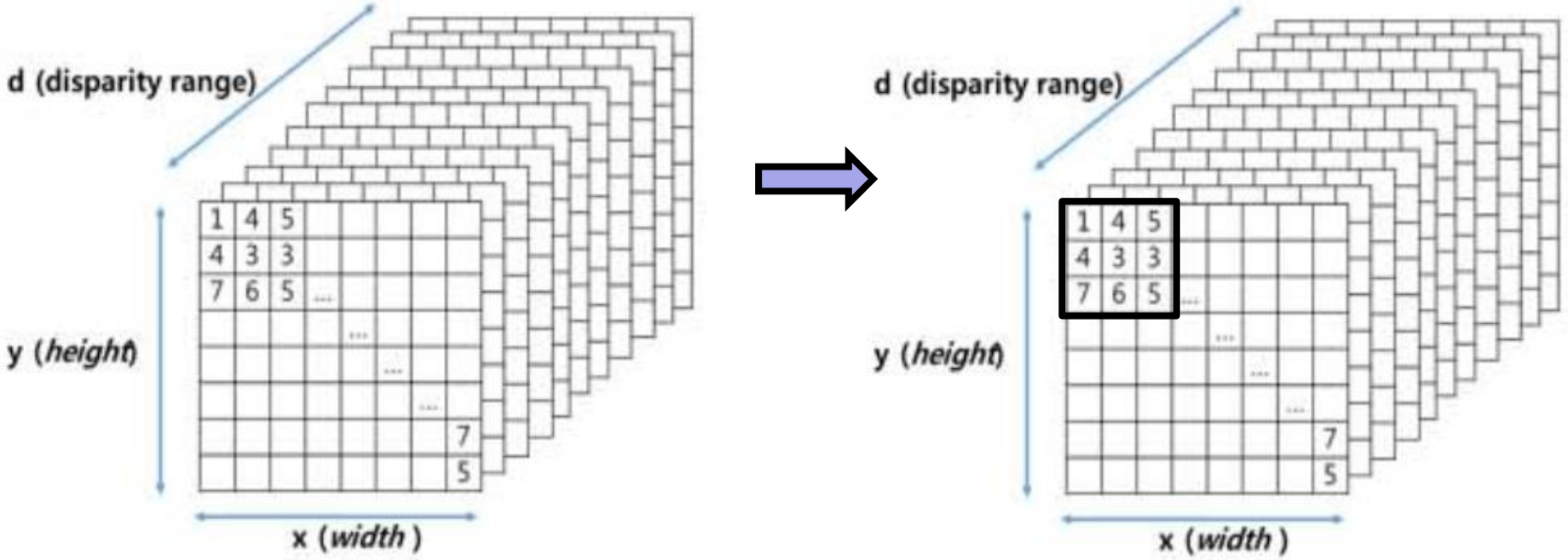
1. Build a '**cost volume**' C of size: h x w x max-disparity
   - $C_0(x, y, d)$ = cost of assigning disparity d to pixel (x, y)



A typical pipeline

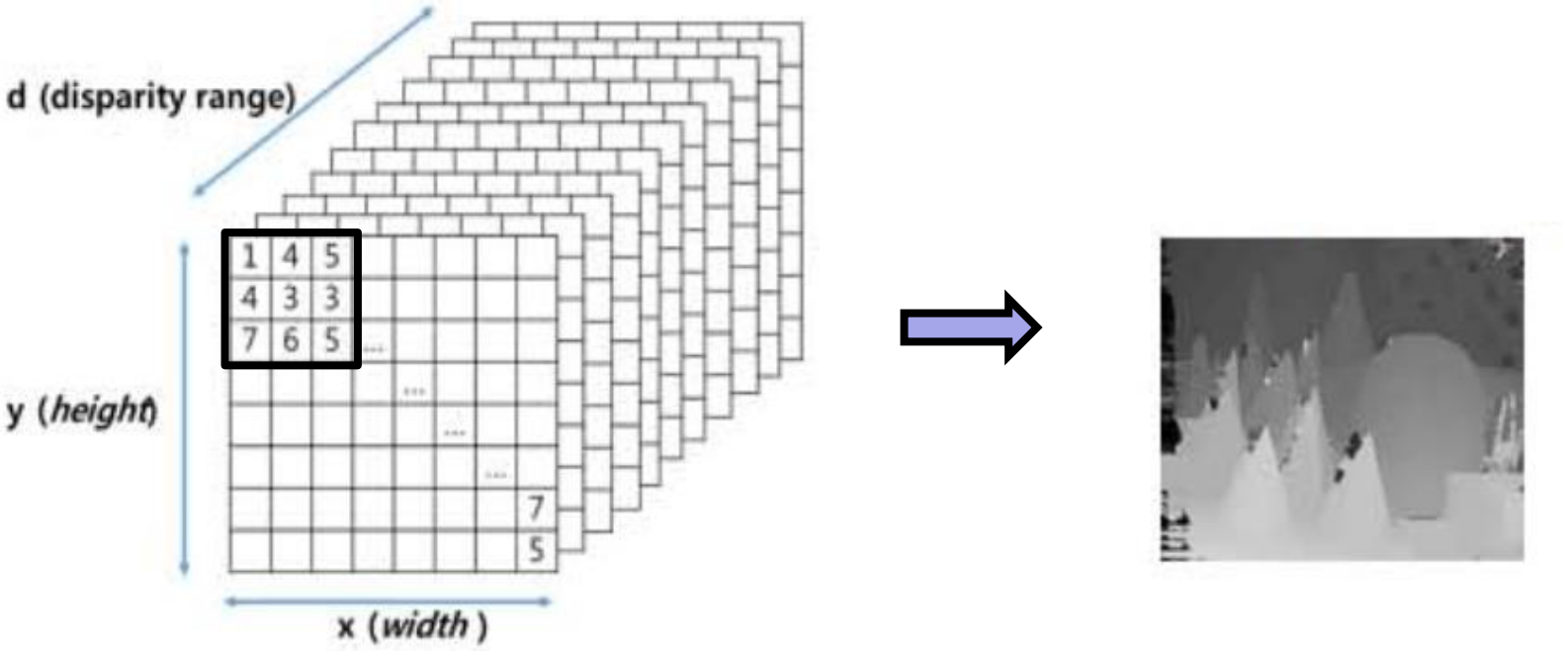
2. **cost aggregation**: smoothing operation (e.g. uniform averaging), usually 2 dimensional (per d layer)

$$C(x, y, d) = w(x, y, d) * C_0(x, y, d)$$



A typical pipeline

3. Winner Take All: choose minimal cost per pixel



Error sources

- Low-contrast ; textureless image regions
- Occlusions
- Forshortening
- Camera calibration errors
- Violations of *brightness constancy* (e.g., specular reflections)
- Large motions

Summary

- Depth from stereo: main idea is to triangulate from corresponding image points.
- Epipolar geometry defined by two cameras
  - We've assumed known extrinsic parameters relating their poses
- Epipolar constraint limits where points from one view will be imaged in the other
  - Makes search for correspondences quicker
- To estimate depth
  - Limit search by epipolar constraint
  - Compute correspondences, incorporate matching preferences