ECE 1390/2390 Image Processing and Computer Vision – Fall 2021

Stereo geometry

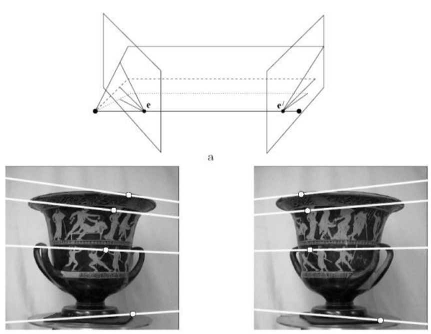
Ahmed Dallal
Assistant Professor of ECE
University of Pittsburgh

Reading

- FP chapter 7

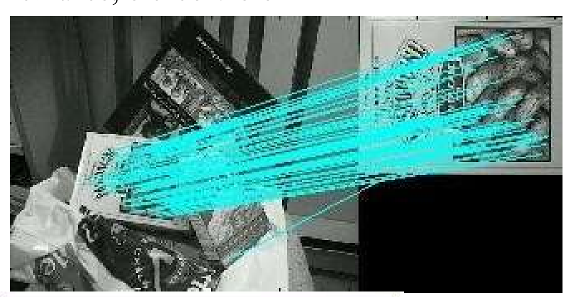
Stereo geometry

Stereo: A Special case of Multiple views



Hartley and Zisserman

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



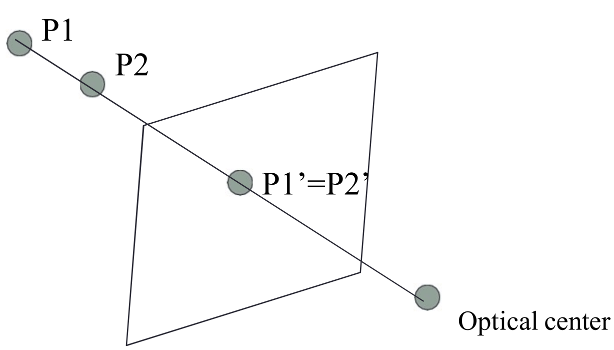
Why multiple views?

- Structure and depth are inherently ambiguous from single views.

Images from S. Lazebnik

Why multiple views?

- Structure and depth are inherently ambiguous from single views.

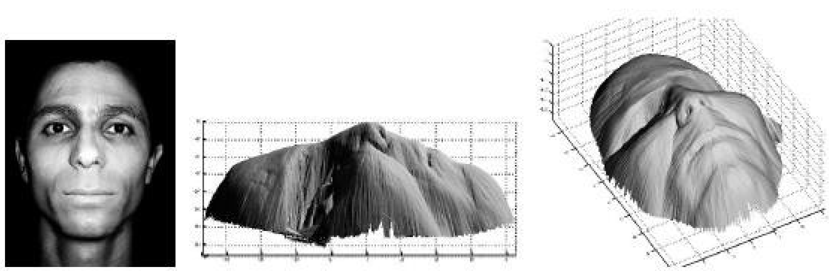


Perspective effects

- What cues help us to perceive 3d shape and depth?
- What about one eye first?

S. Seitz

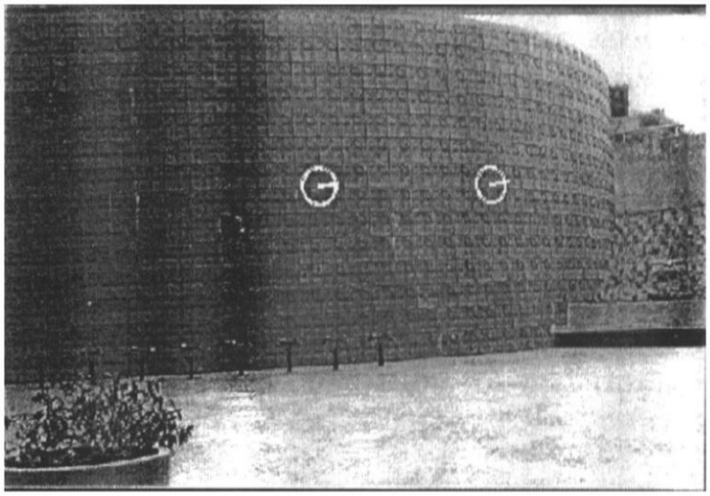
Shading



a) b) c)

K. Grauman

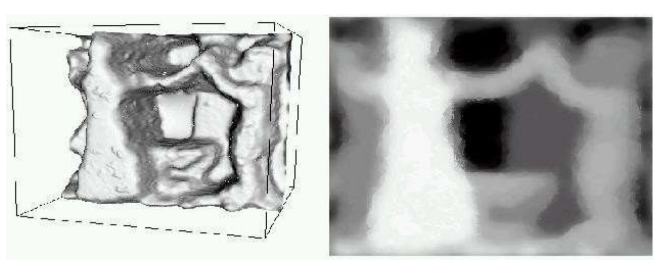
Texture



A.M. Loh. The recovery of 3-D structure using visual texture patterns.

Focus / defocus

Images from same point of view, different camera parameters



3d shape / depth estimates

Figures from H. Jin and P.Favaro, 2002

Motion

Figures from L. Zhang

But we (and lots of creatures) have two eyes!

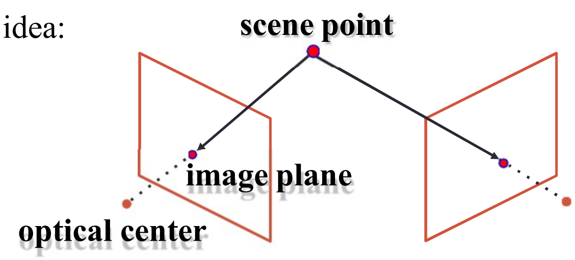
Stereo:

- The image from one eye is a little different than the image form he other eye.
- Think of shape from "motion" between two views
- Infer 3d shape of scene from two (multiple) images from different viewpoints

But we (and lots of creatures) have two eyes!

- **Stereo**:
  - shape from “motion” between two views
  - infer 3d shape of scene from two (multiple) images from different viewpoints

Main idea:

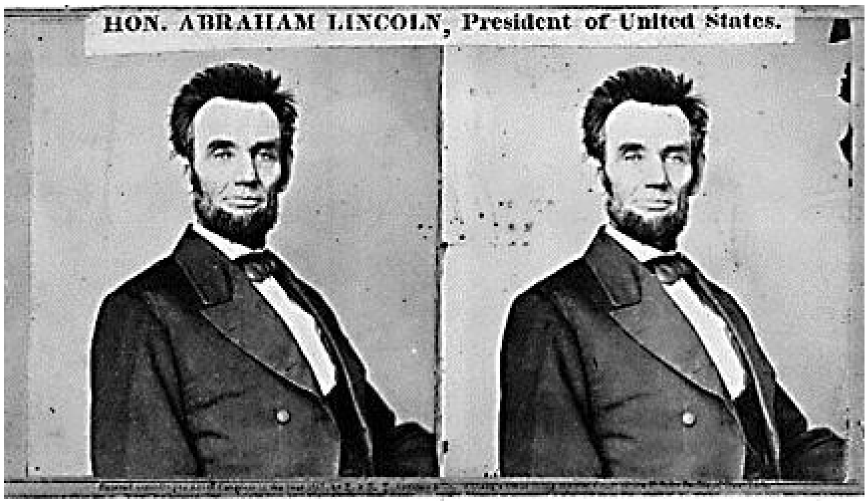


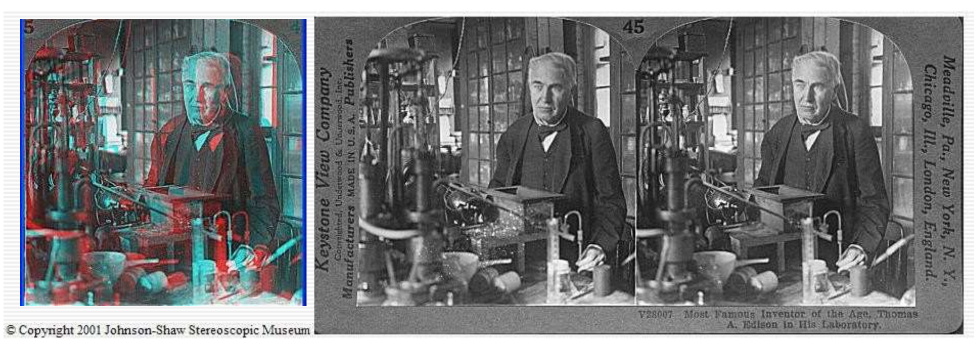
Stereo photography and stereo viewers

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

People fascinated by3D

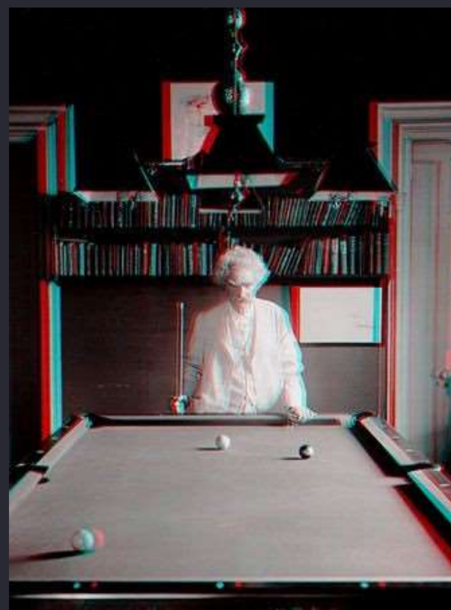




http://www.johnsonshawmuseum.org

Public Library, Stereoscopic Looking Room, Chicago, by Phillips, 1923

Teesta suspension bridge-Darjeeling, India



Mark Twain at Pool Table", no date, UCR Museum of Photography

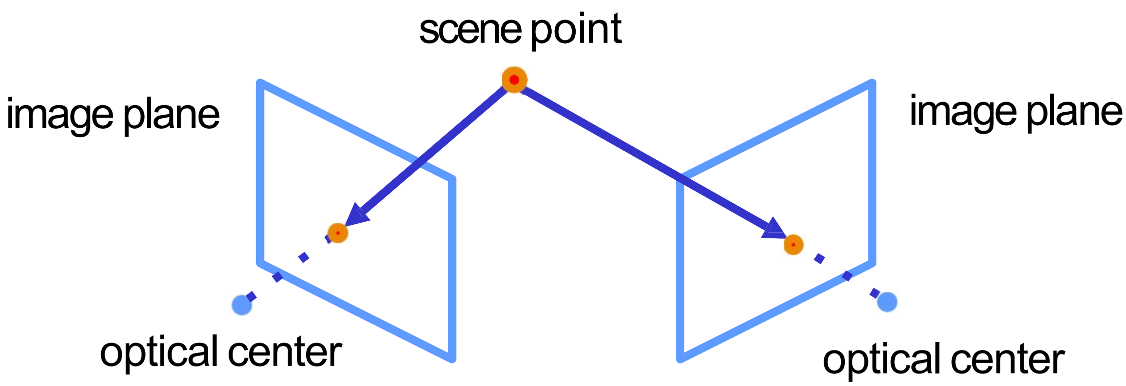
Stereo photography and stereo viewers

Stereo photography and stereo viewers

The Basic Idea: Two slightly different images

http://www.well.com/~jimg/stereo/stereo_list.html

Basic stereo geometry

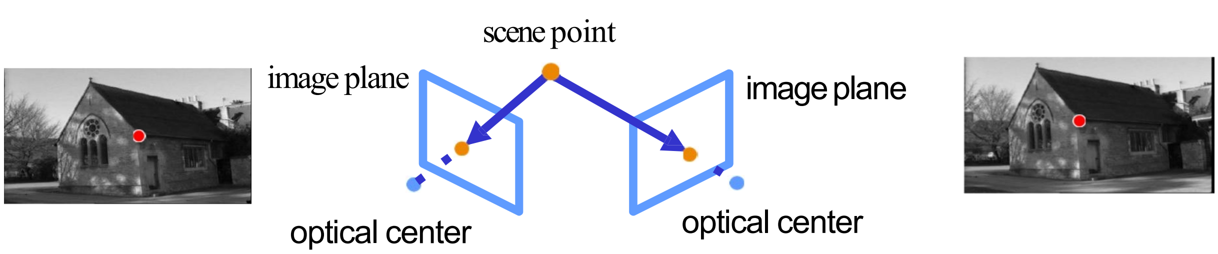


Estimating depth with stereo

Stereo: shape from "motion" between two views

We'll need to consider:

- Info on camera pose ("calibration")
- Image point correspondences



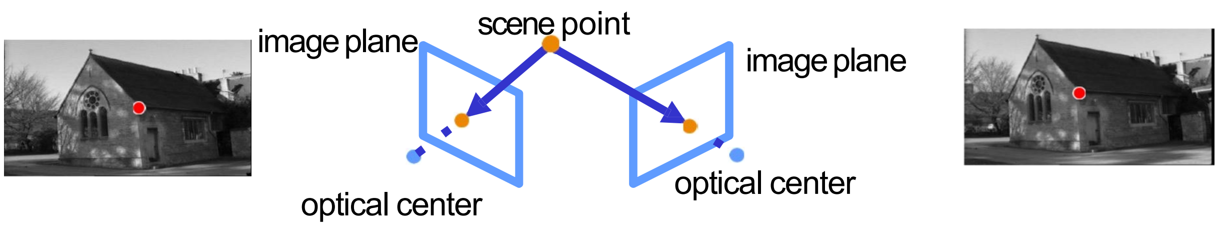
Estimating depth with stereo

Stereo: shape from "motion" between two views

We'll need to consider:

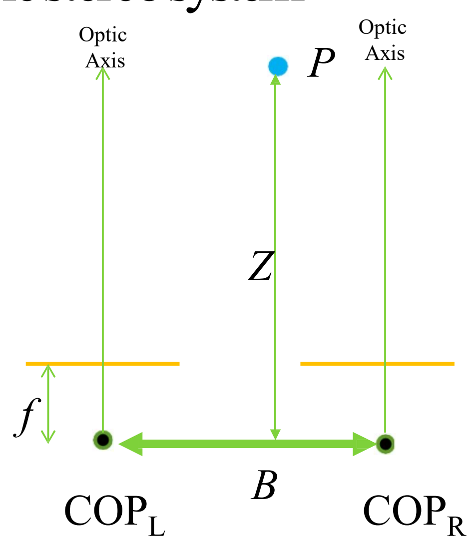
Info on camera pose ("calibration")

Image point correspondences



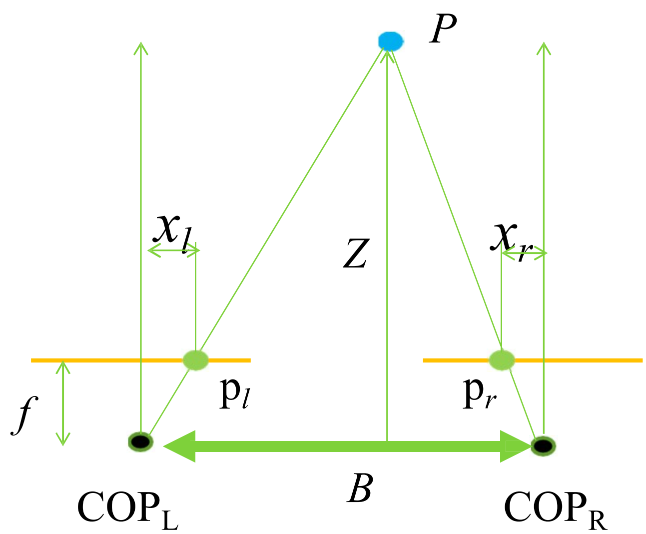
Geometry for a simple stereo system

- First, assuming parallel optical axes, known camera parameters (i.e., calibrated cameras)
- Figure is looking down on the cameras and image planes
- Baseline *B*, focal length *f*
- Point *P* is distance *Z* in camera coordinate systems



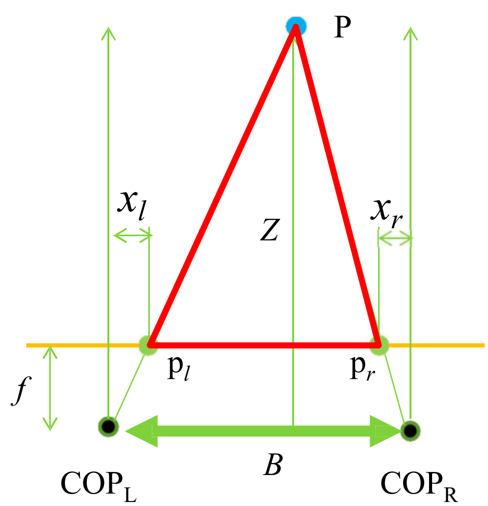
Geometry for a simple stereo system

- Point *P* projects into left and right images.
- Distance is positive in left image, and negative in right



Geometry for a simple stereo system

- What is the expression for Z?
- Similar triangles (p_l, P, p_r) and (C_L, P, C_r):

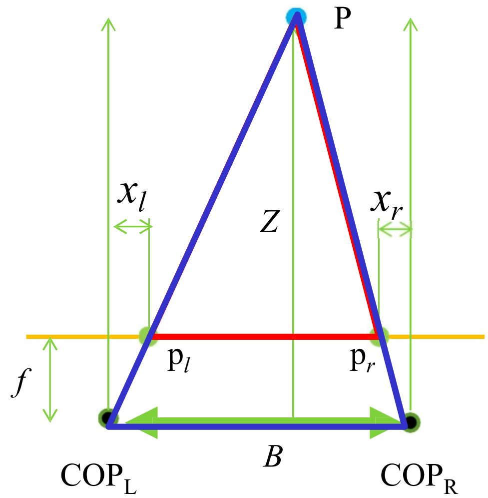


Geometry for a simple stereo system

- What is the expression for Z?
- Similar triangles (p_l, P, p_r) and (C_L, P, C_r):

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

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



Disparity *... is* ***inversely*** *proportional to depth*

Depth from disparity

image I(x,y)

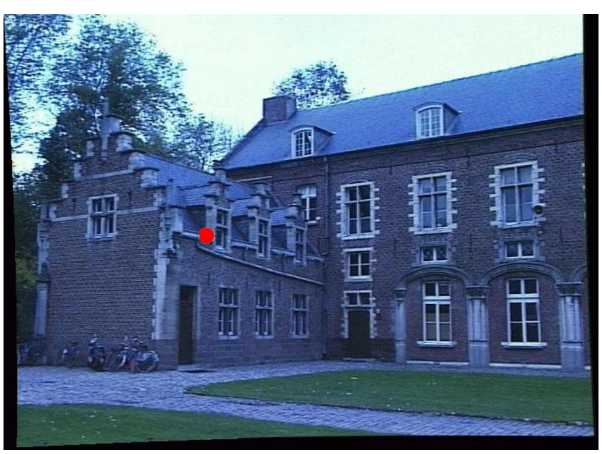


image I'(x,y)

Depth from disparity

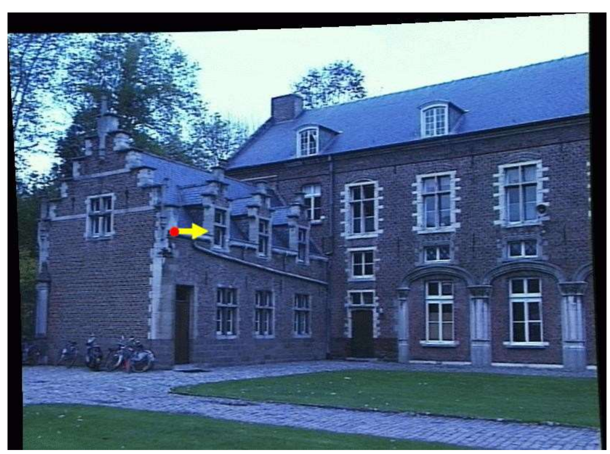
image I(x,y)

Disparity map D(x,y)

Depth from disparity

$(x',y')=(x+D(x,y), y)$

image $I'(x',y')$

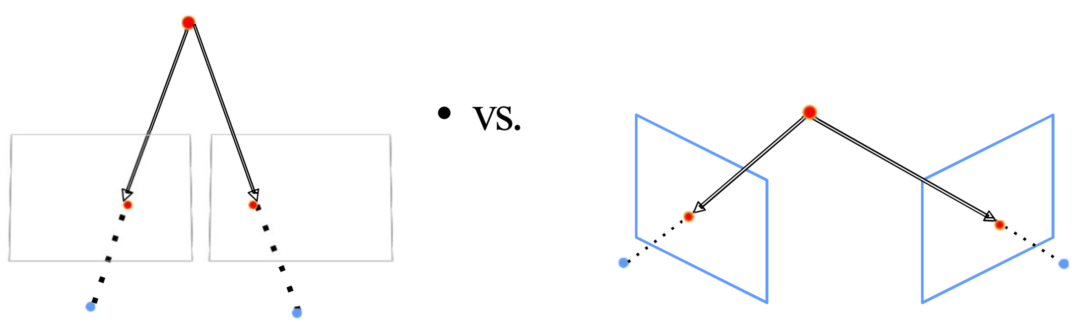


So if we could find the **corresponding points** in two images, we could **estimate relative depth**…

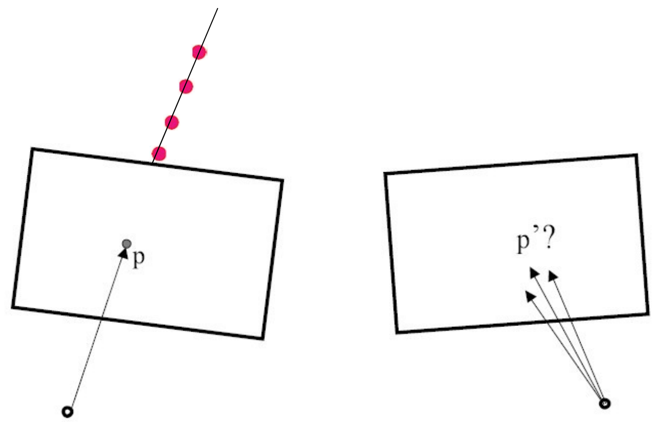
Epipolar geometry

General case, with calibrated cameras

- The two cameras need not have parallel optical axes and image planes.



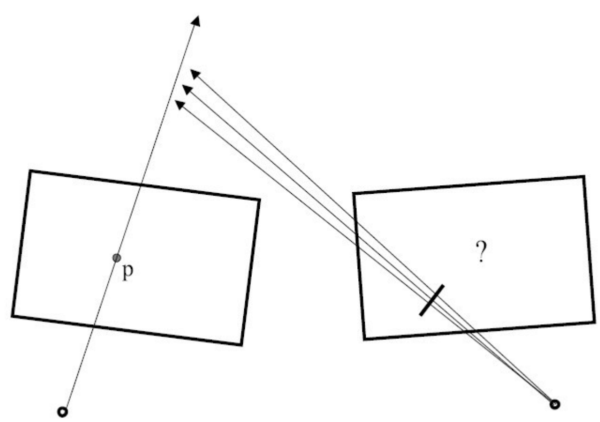
Stereo correspondenceconstraints



Given p in left image, where can corresponding point p' be?

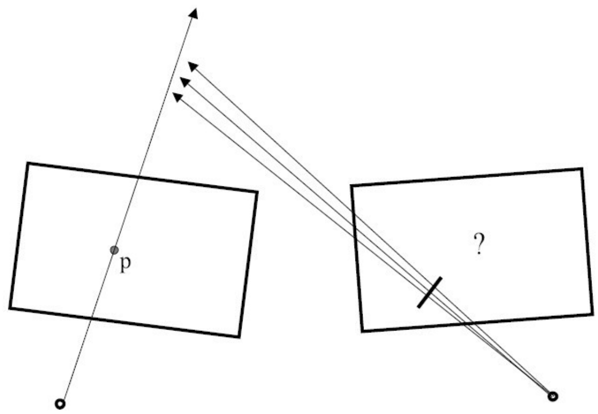
Stereo correspondenceconstraints

Remember: in perspective projection, lines project into lines.

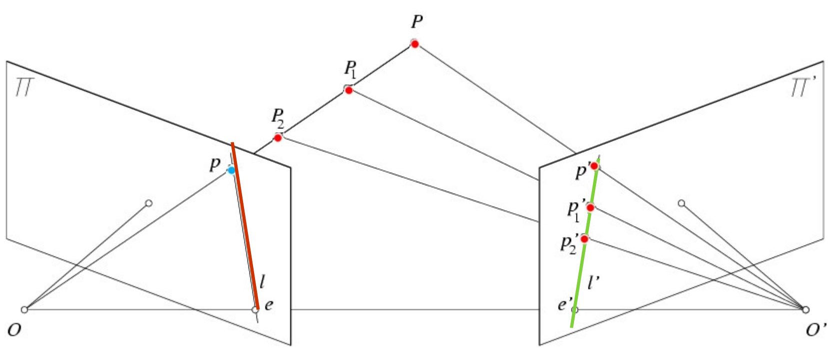


Stereo correspondence constraints

So the ***line*** containing the center of projection and the point P in the left image must project to a ***line*** in the right image.

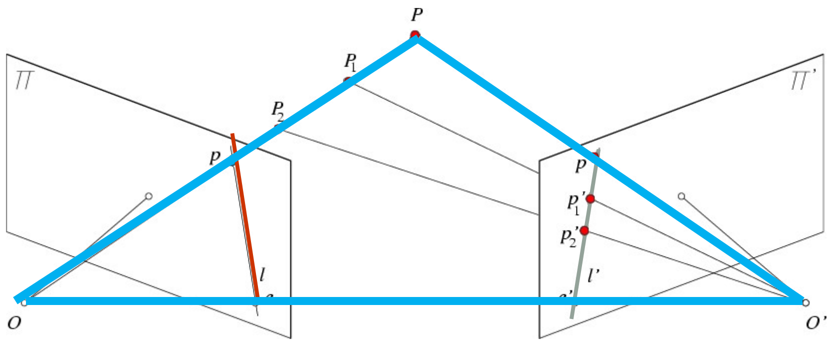


Epipolar constraint



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

Epipolar constraint

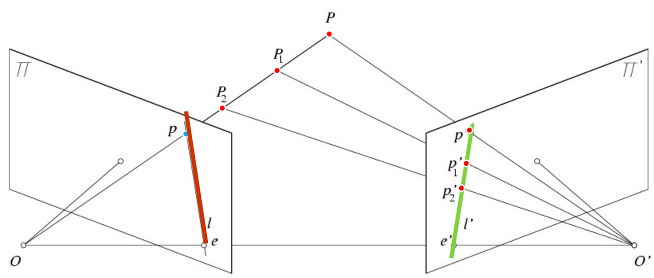


Geometry of two views constrains where the corresponding pixel for some image point in the first view must occur in the second view.

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

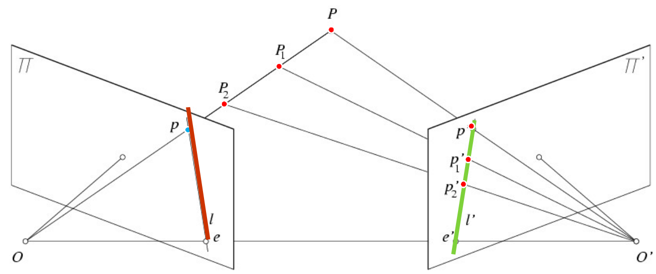
Epipolar geometry: Terms

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



Epipolar geometry: Terms

- ***Baseline***: line joining the camera centers
- *Epipolar plane*: plane containing baseline and world point
- *Epipolar line*: intersection of epipolar plane with the image plane – come in pairs
- *Epipole*: point of intersection of baseline with image plane
  - All epipolar lines intersect at the epipole
  - An epipolar plane intersects the left and right image planes in epipolar lines



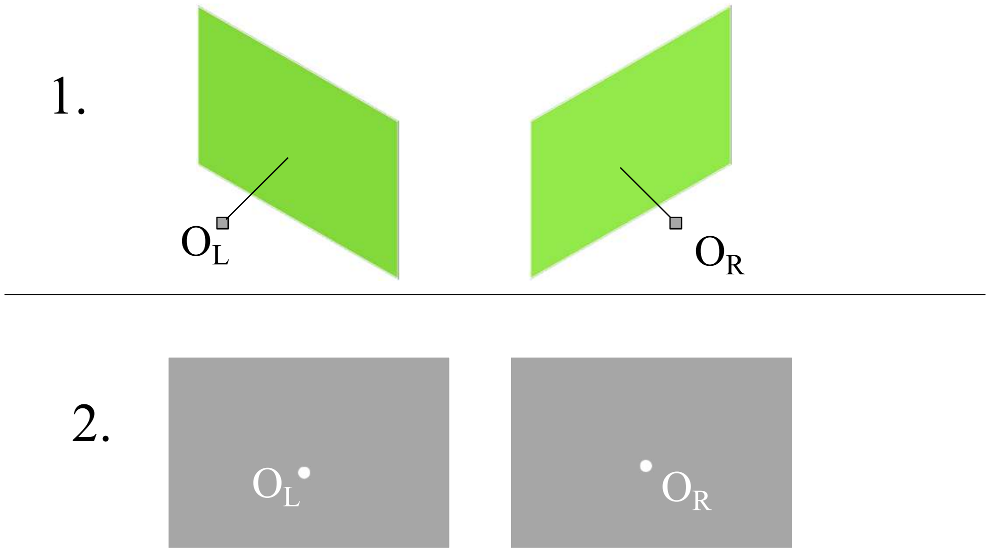
Why is the epipolar constraint useful?

Epipolar constraint

The *epipolar constraint* reduces the correspondence problem to a 1D search along an epipolar line.

Image from Andrew Zisserman

What do the epipolar lines look like?



Example: converging cameras

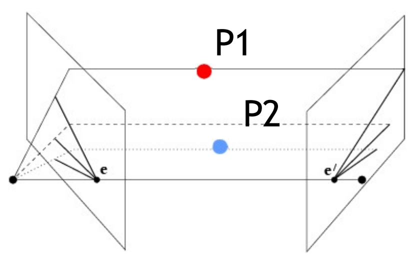
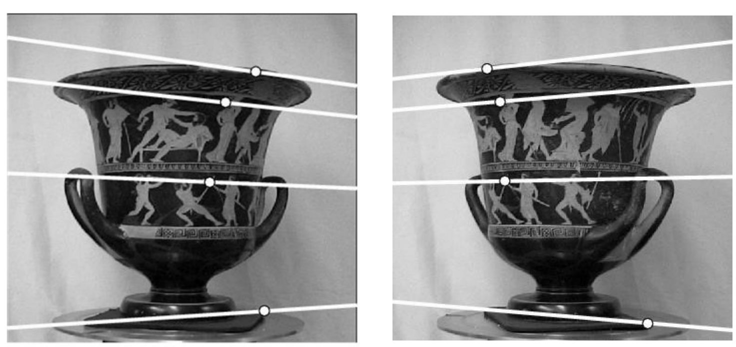
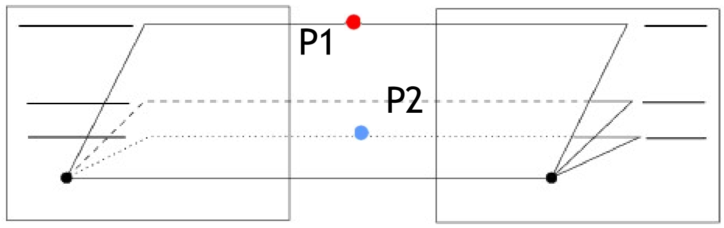


Figure from Hartley & Zisserman



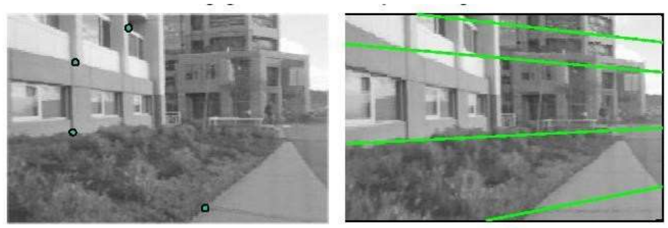
Example: parallel image planes



Where are the epipoles?

Quiz: two stereo pairs

a)



b)

Quiz:

How do we know that (B) has parallel image planes

a) The epipolar lines are horizontal
b) The epipolar lines are parallel
c) Because I just said (B) had parallel image planes

Stereo correspondence

For now assume parallel image planes…

- Assume parallel (co-planar) image planes…
- Assume same focal lengths…
- Assume epipolar lines are horizontal…
- Assume epipolar lines are at the same y location in the image…

- That's a lot of assuming, but it allows us to move to the correspondence problem – which you will be solving!

Correspondence problem

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

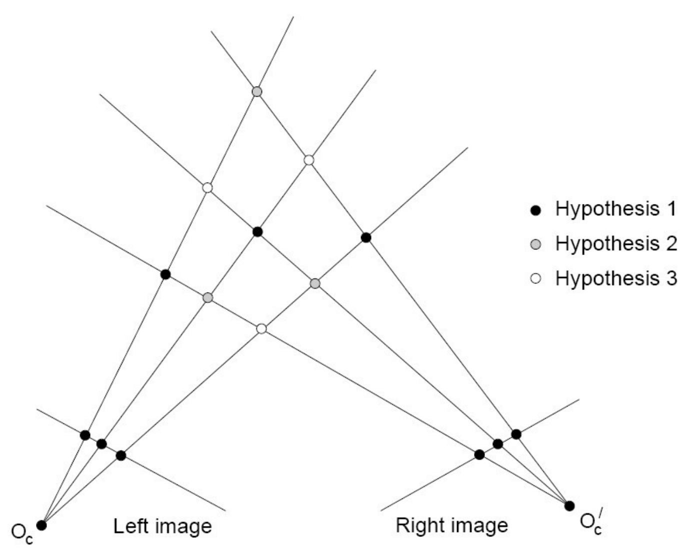


Figure from Gee & Cipolla 1999

Correspondence problem

Beyond the hard constraint of epipolar geometry, there are "soft" constraints to help identify corresponding points

- Similarity
- Uniqueness
- Ordering
- Disparity gradient is limited
  - Depth doesn't change too quick

Correspondence problem

Beyond the hard constraint of epipolar geometry, there are "soft" constraints to help identify corresponding points

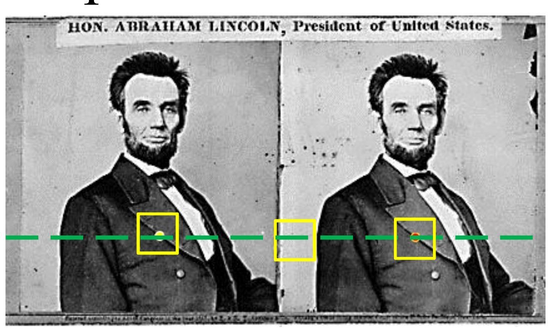
- **Similarity**
- Uniqueness
- Ordering
- Disparity gradient is limited

Correspondence problem

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 pixel / window in the left image

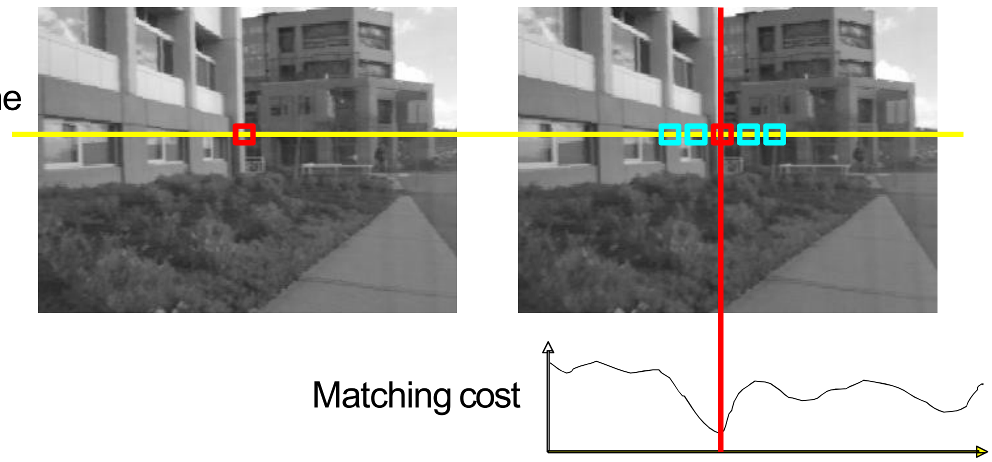
- Compare with every pixel / window on same epipolar line in right image
- Pick position with minimum match cost (e.g., SSD, normalized correlation)

Adapted from Li Zhang

Correspondence search: (dis)similarity constraint

Left Right

scanline



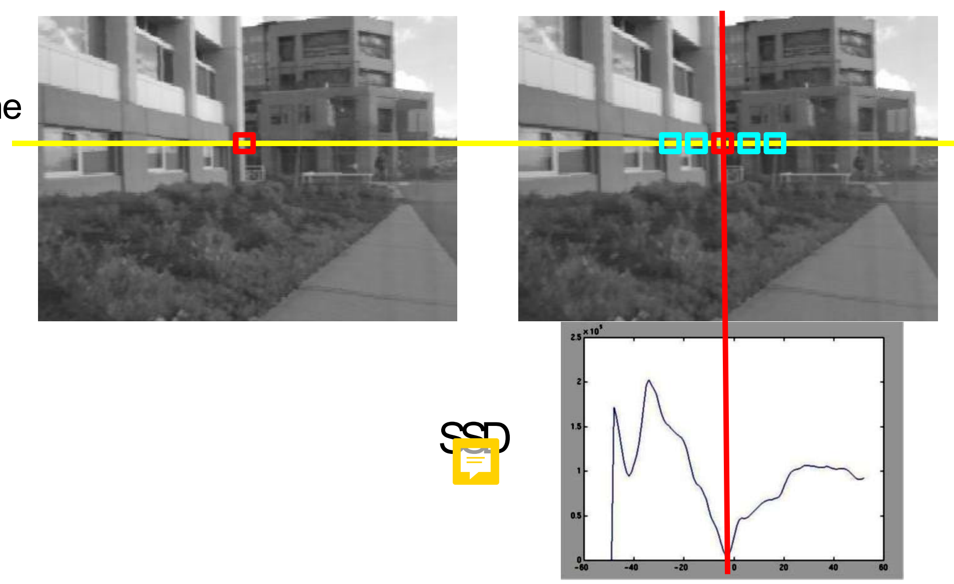
Matching cost

disparity

Correspondence search: (dis)similarity constraint

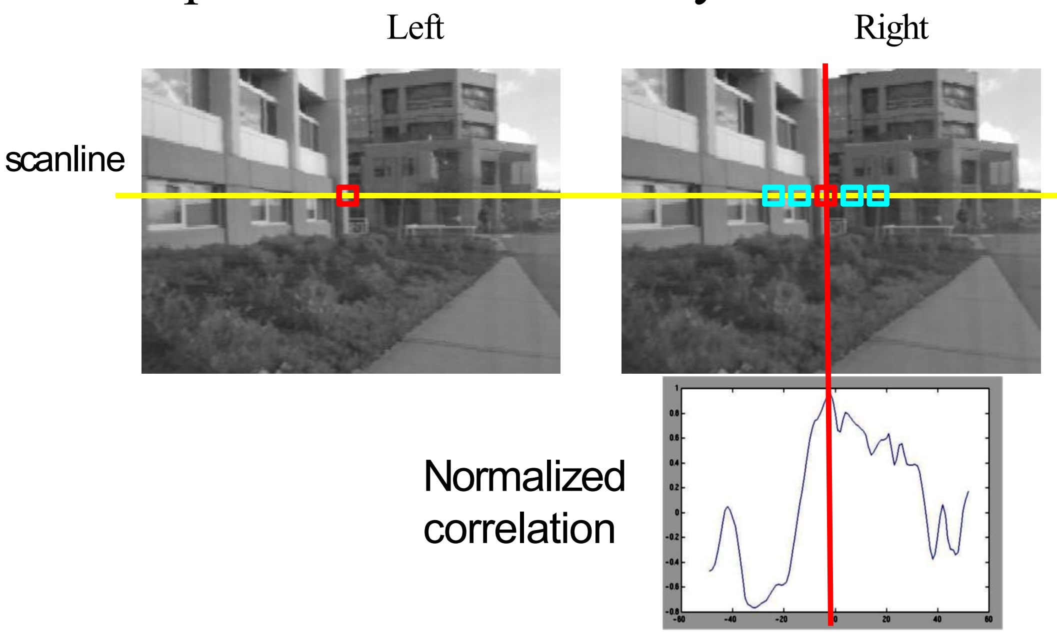
Left Right

scanline



SSD

Correspondence search: similarity constraint



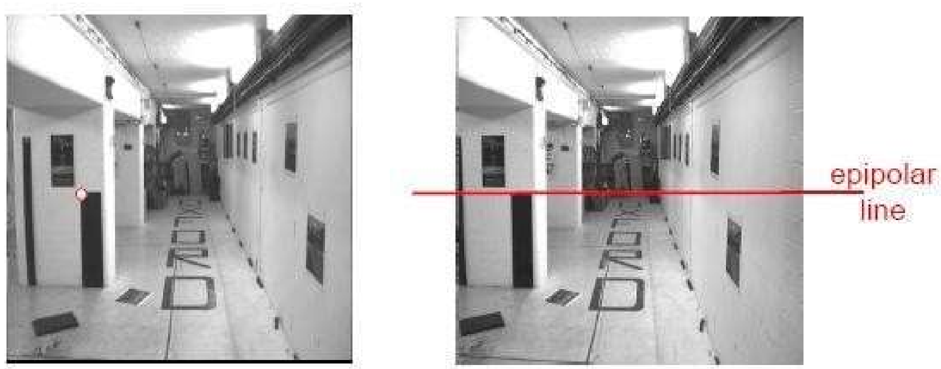
Matlab Implementation

```
function best_x = find_best_match(patch, strip)
    min_diff = Inf;
    best_x = 0; % haven't found it yet
    for x = 1:(size(strip)(2) - size(patch)(2))
        other_patch = strip(:, x:(x + size(patch)(2) - 1));
        diff = sumsq((patch - other_patch)(:));
        if diff < min_diff
            min_diff = diff;
            best_x = x;
        endif
    endfor
endfunction
```

Matlab Implementation

- Test code is posted to courseweb
- Feel free to try normalized correlation

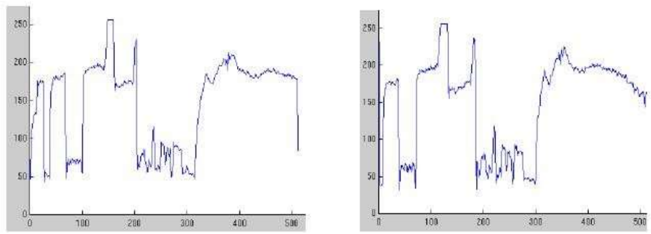
Correspondence problem



Source: Andrew Zisserman

Correspondence problem

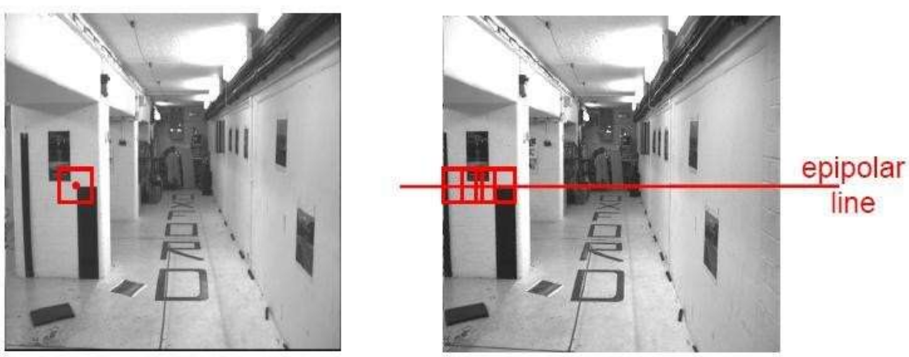
Intensity profiles



Clear correspondence between intensities, but also noise and ambiguity

Source: Andrew Zisserman

Correspondence problem



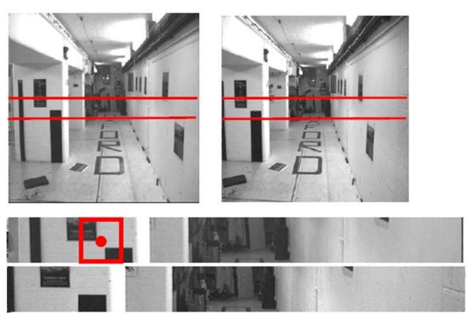
Neighborhoods of corresponding points are similar in intensity patterns.

Source: Andrew Zisserman

Correlation-based window matching

left image band (x)

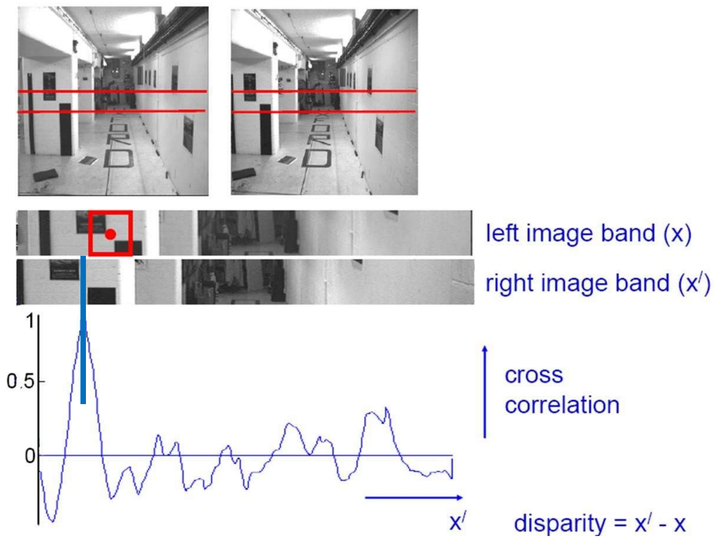
Correlation-based window matching



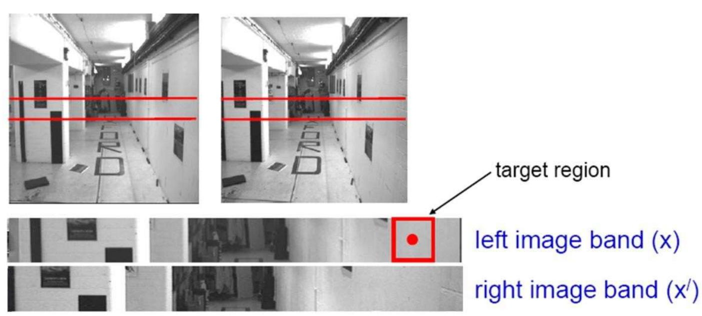
left image band (x)

right image band ($x^{/}$)

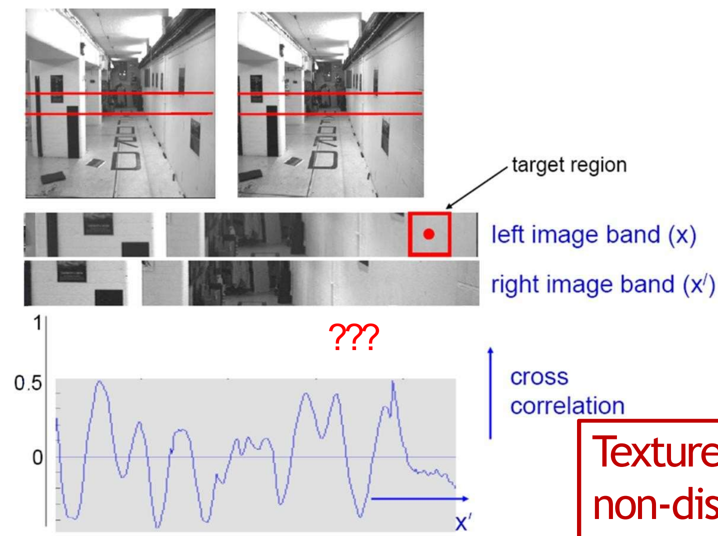
Correlation-based window matching



Correlation-based window matching

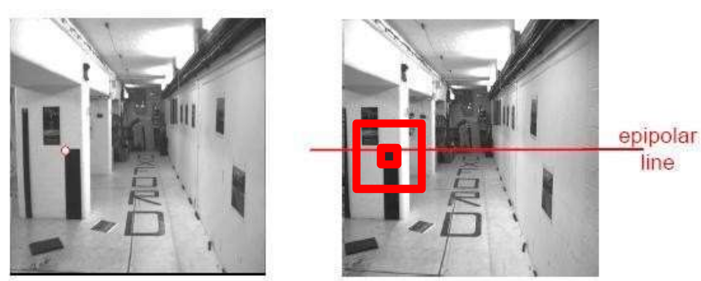


Correlation-based window matching



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

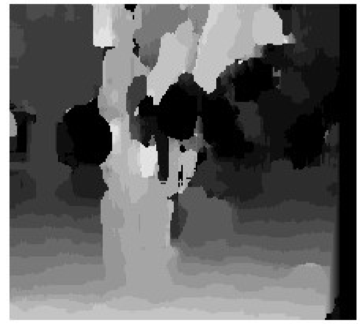
Effect of window size



Source: Andrew Zisserman

Effect of window size

W = 3



W = 20

Figures from Li Zhang

Correspondence problem

Beyond the hard constraint of epipolar geometry, there are "soft" constraints to help identify corresponding points

- Similarity
- Uniqueness
- Ordering
- Disparity gradient is limited

Uniquenessconstraint

No more than one match in right image for every point in left image

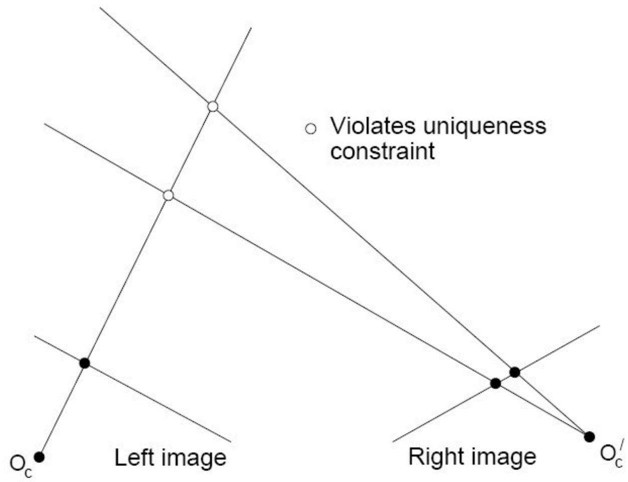
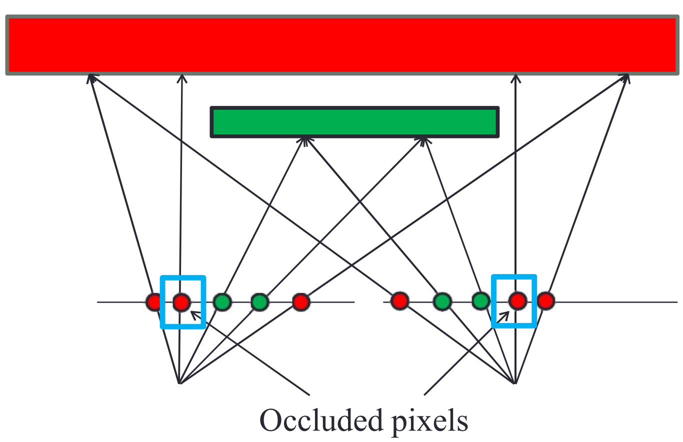


Figure from Gee & Cipolla 1999

Problem: Occlusion

- Uniqueness says "up to match" per pixel
- When is there no match?



Ordering constraint

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

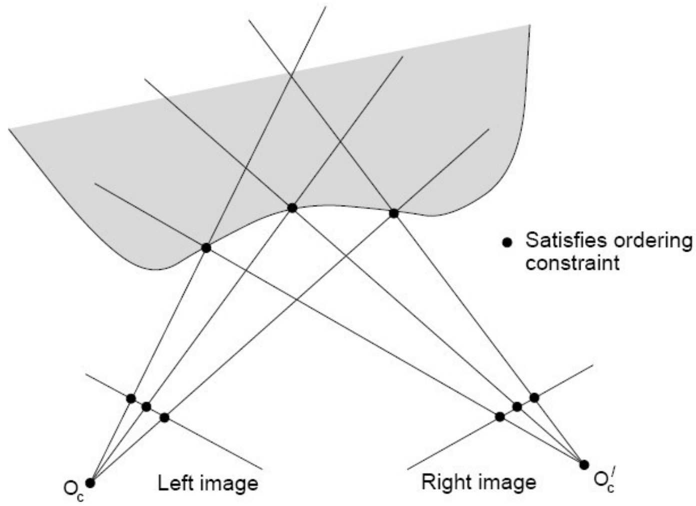
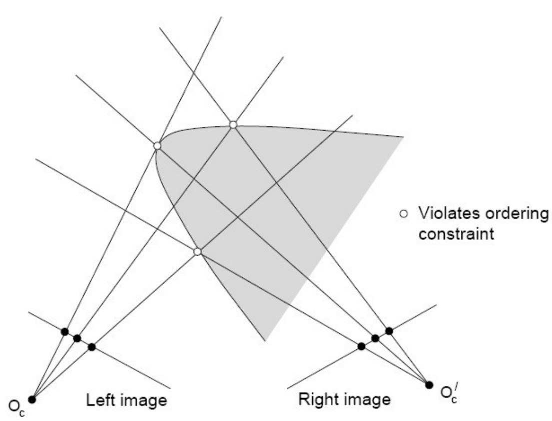


Figure from Gee & Cipolla 1999

Ordering constraint

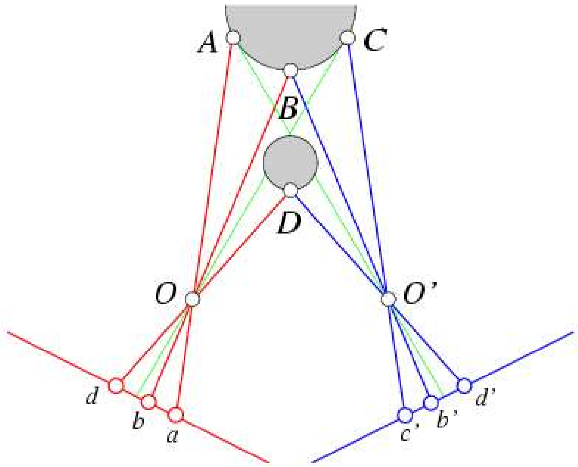
Won't always hold, e.g. consider transparent object…



Figures from Forsyth & Ponce

Ordering constraint

…or a narrow occluding surface

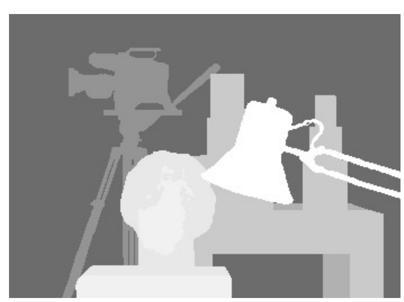


Figures from Forsyth & Ponce

Stereo results

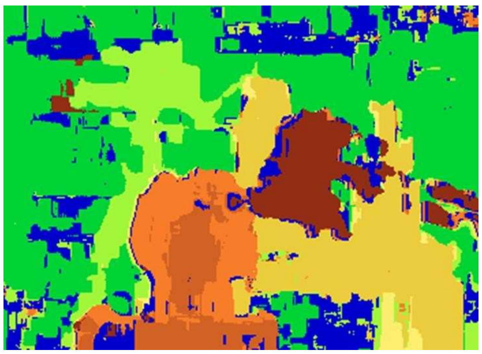
Image data from University of Tsukuba

Scene



Ground truth

Results with window search



Window-based matching
(best window size)

Ground truth

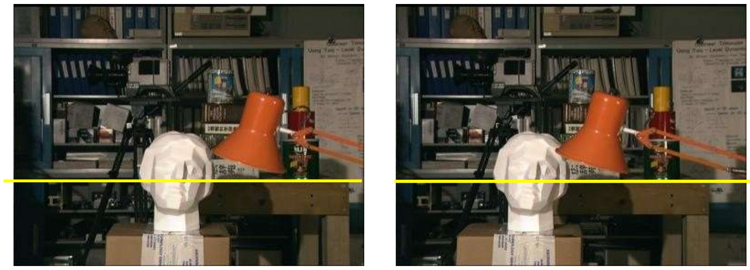
Better solutions

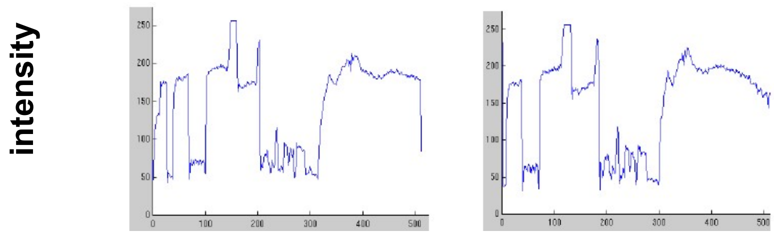
Beyond individual correspondences to estimate disparities:

- Optimize correspondence assignments jointly
  - Scanline at a time (DP)
  - Full 2D grid (graph cuts)

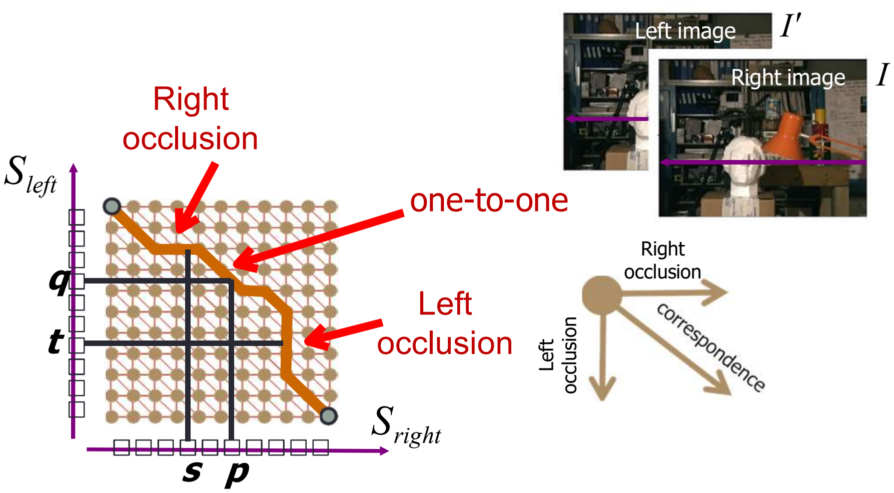
Scanline stereo

Coherently match pixels on the entire scanline





"Shortest paths" for scan-line stereo



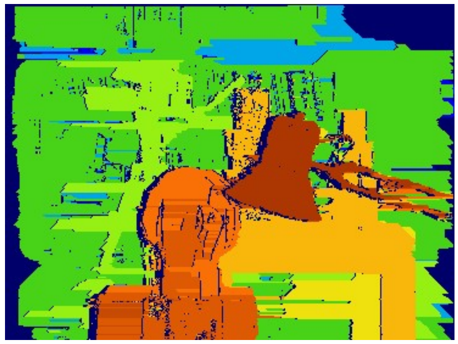
Can be implemented with dynamic programming

Ohta & Kanade '85, Cox et al. '96, Intille & Bobick, '01

Slide credit: Y. Boykov

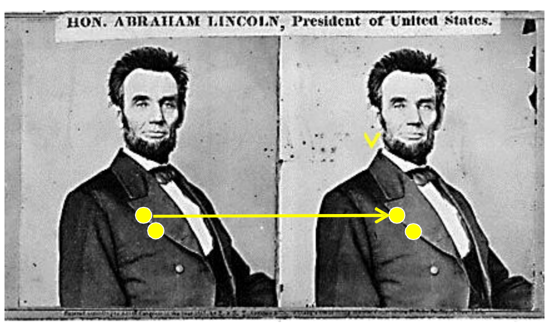
Coherent stereo on 2D grid

Scanline stereo generates streaking artifacts



Can’t use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid

Stereo as energy minimization



What defines a good stereo correspondence?

1. ***Match quality*** - Want each pixel to find a good appearance match in the other image
2. ***Smoothness*** - of two pixels are adjacent, they should (usually) move about the same amount

Stereo matching as energy minimization

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

Source: Steve Seitz

Stereo matching as energy minimization

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

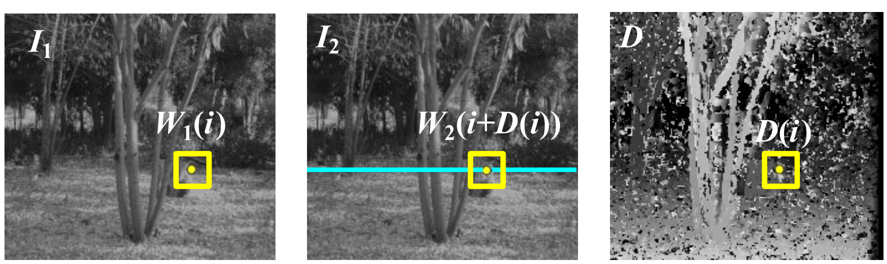
Source: Steve Seitz

Stereo matching as energy minimization

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

Source: Steve Seitz

Stereo matching as energy minimization



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

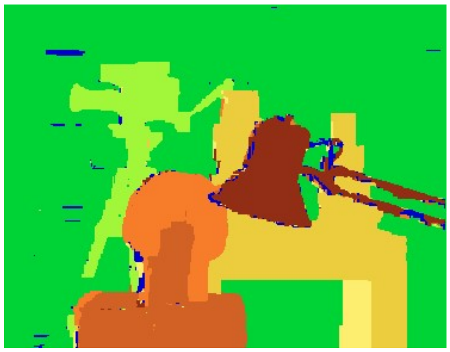
$$\boxed{E_{\text{data}} = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2} \qquad \boxed{E_{\text{smooth}} = \sum_{\text{neighbors } i,j} \rho\left(D(i) - D(j)\right)}$$

- Energy functions of this form can be minimized using *graph cuts*

Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Source: Steve Seitz

Better results…



State of the art method

Ground truth

For the latest and greatest: http://www.middlebury.edu/stereo

Challenges

- Low-contrast ; textureless image regions
- Occlusions
- Violations of brightness constancy (e.g., specular reflections)
- Really large baselines (foreshortening and appearance change)
- Camera calibration errors