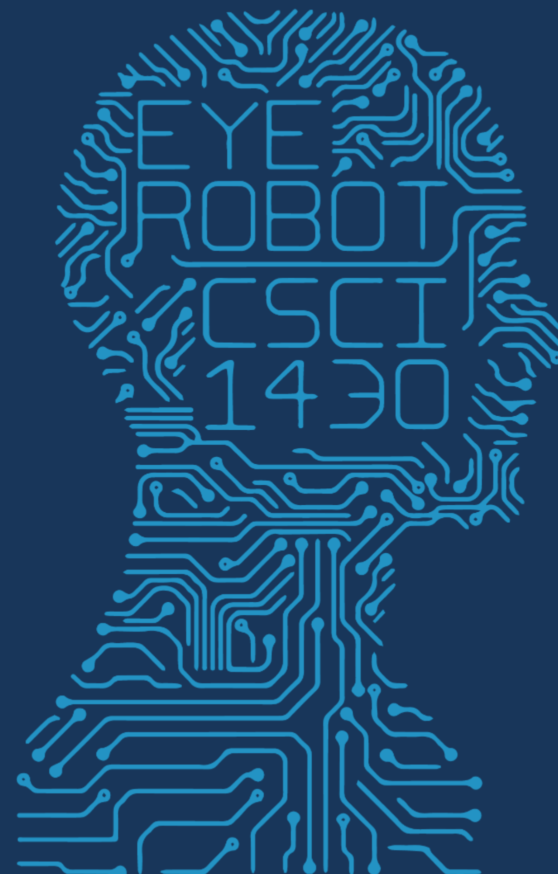
1950

Future Vision



2017 MWF 1pm 368

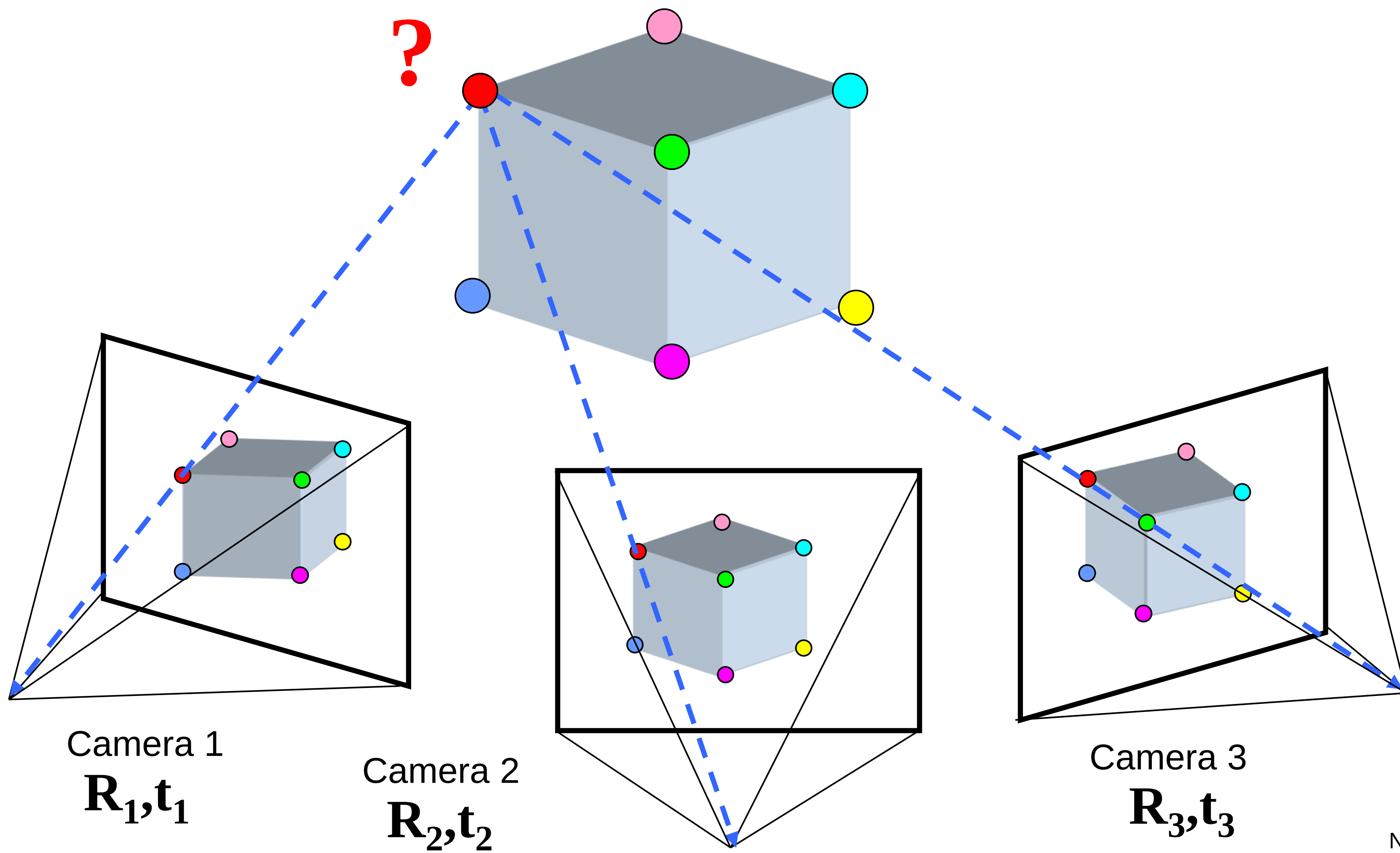
Computer Vision

Chaplin, Modern Times, 1936

[A Bucket of Water and a Glass Matte: Special Effects in *Modern Times;* bonus feature on The Criterion Collection set]

Multi-view geometry problems

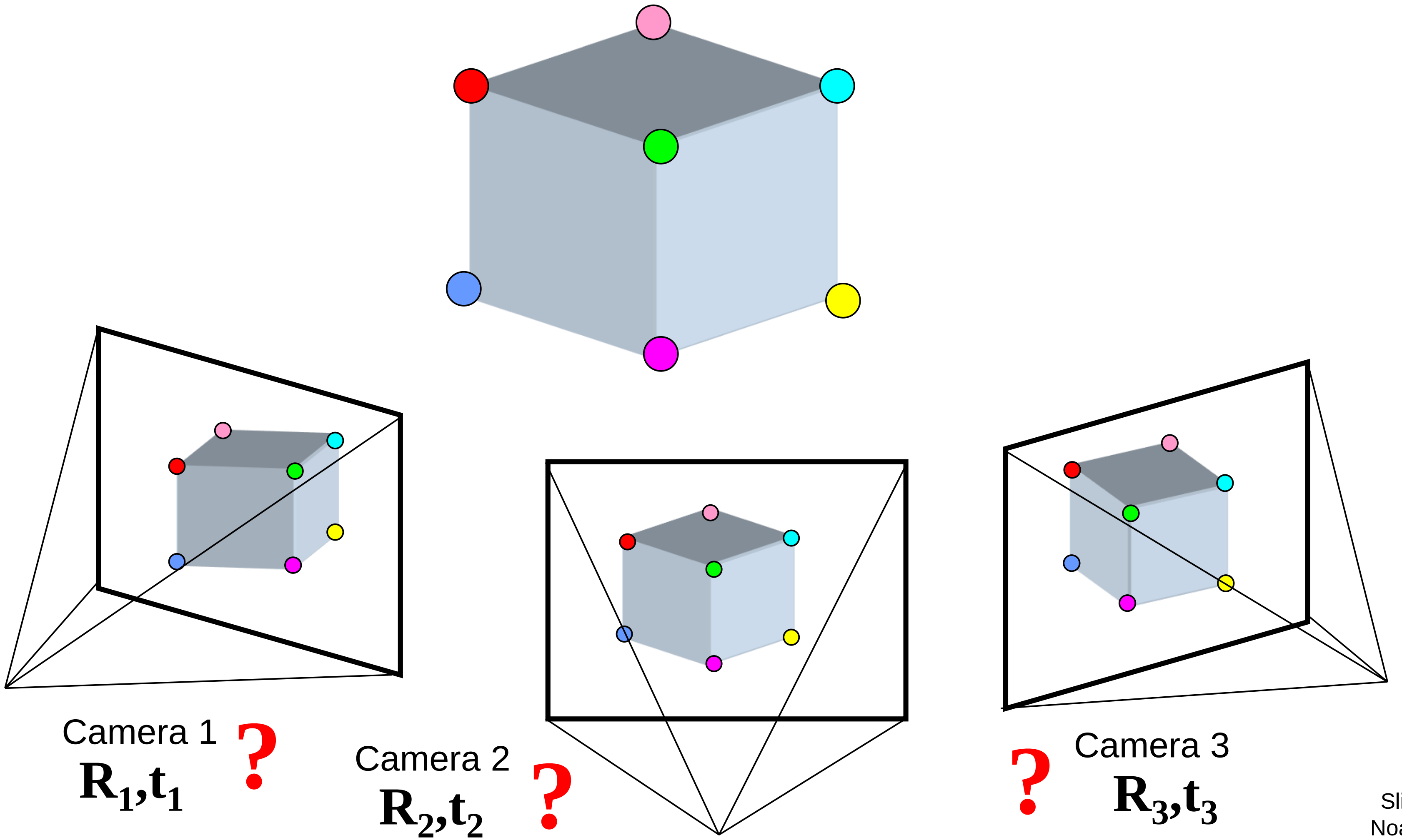
- **Structure:** Given projections of the same 3D point in two or more images, compute the 3D coordinates of that point



Slide credit: Noah Snavely

Multi-view geometry problems

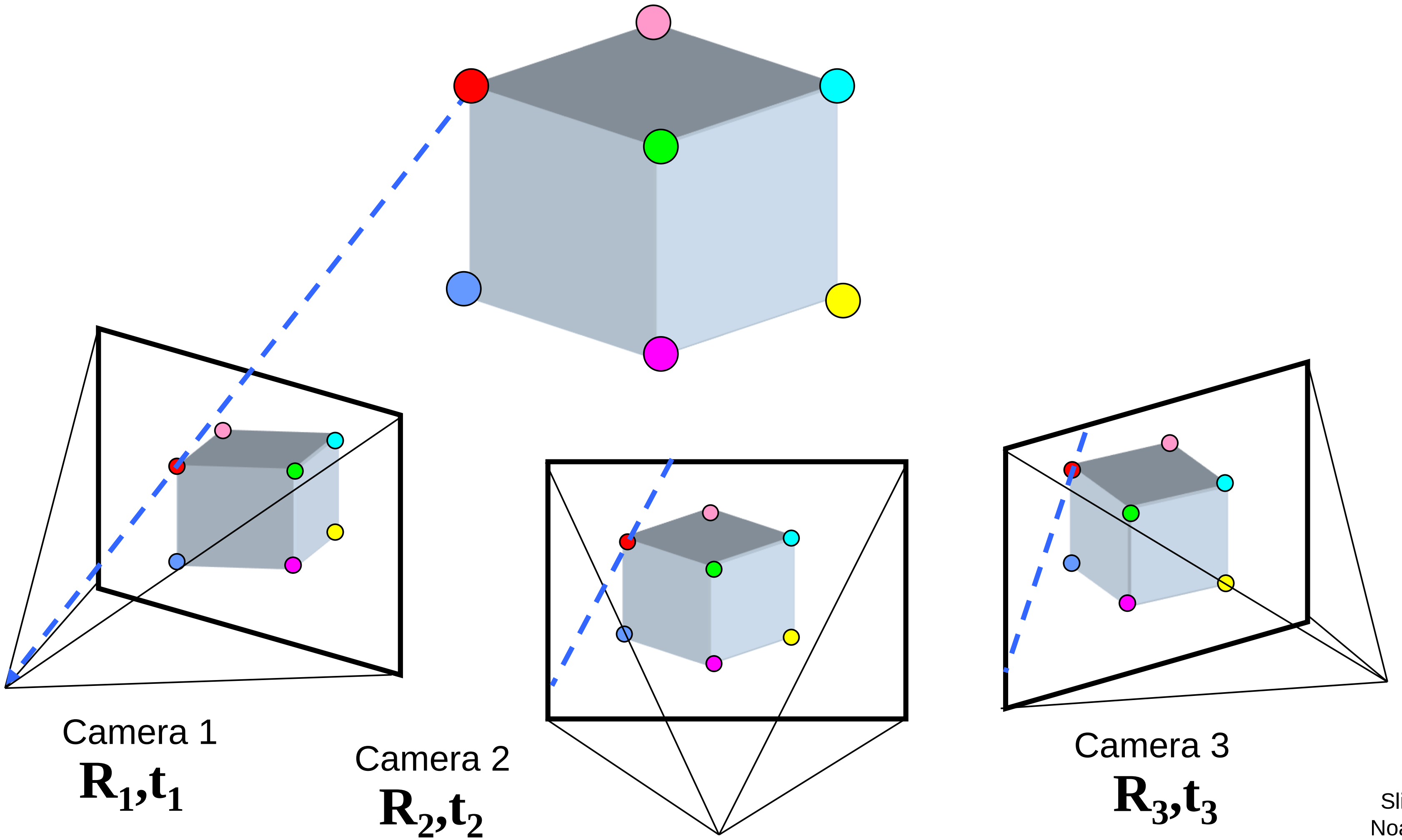
- **Motion:** Given a set of corresponding points in two or more images, compute the camera parameters



Slide credit: Noah Snavely

Multi-view geometry problems

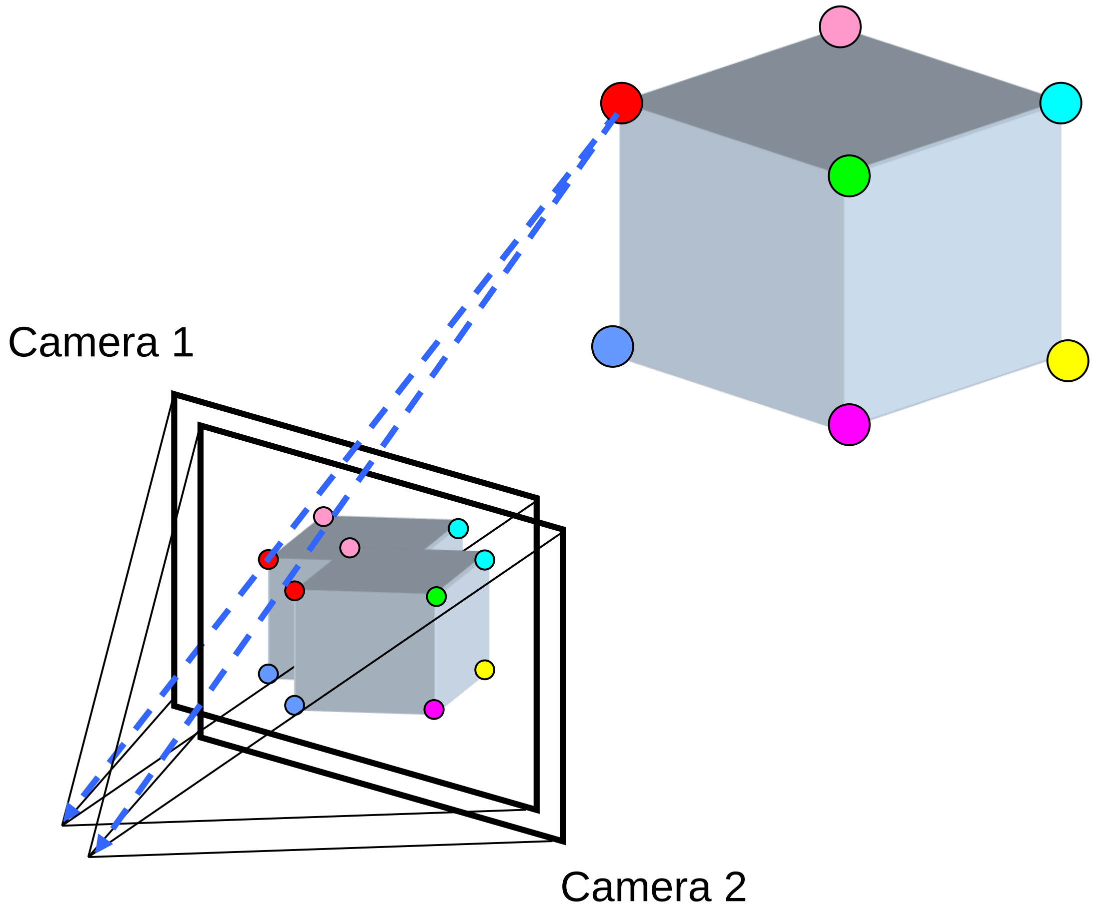
- **Stereo correspondence:** Given a point in one of the images, where could its corresponding points be in the other images?



Slide credit: Noah Snavely

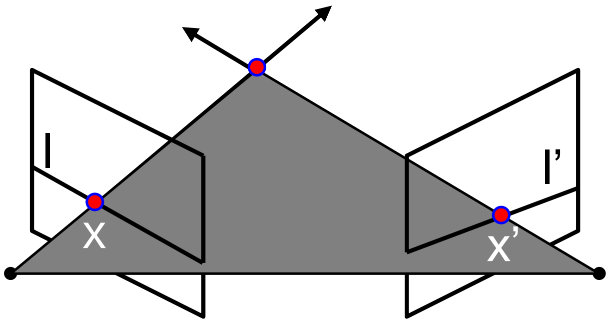
Multi-view geometry problems

- **Optical flow:** Given two images, find the location of a world point in a second close-by image with no camera info.



Fundamental matrix

Let *x* be a point in left image, *x'* in right image



Epipolar relation

- *x* maps to epipolar line *l'*
- *x'* maps to epipolar line *l*

Epipolar mapping described by a 3x3 matrix *F:*

$$l' = Fx$$
$$l = F^T x'$$

It follows that: $x' F x = 0$

Fundamental matrix

This matrix F is called

- the "Essential Matrix"
  - when image intrinsic parameters are known
- the "Fundamental Matrix"
  - more generally (uncalibrated case)

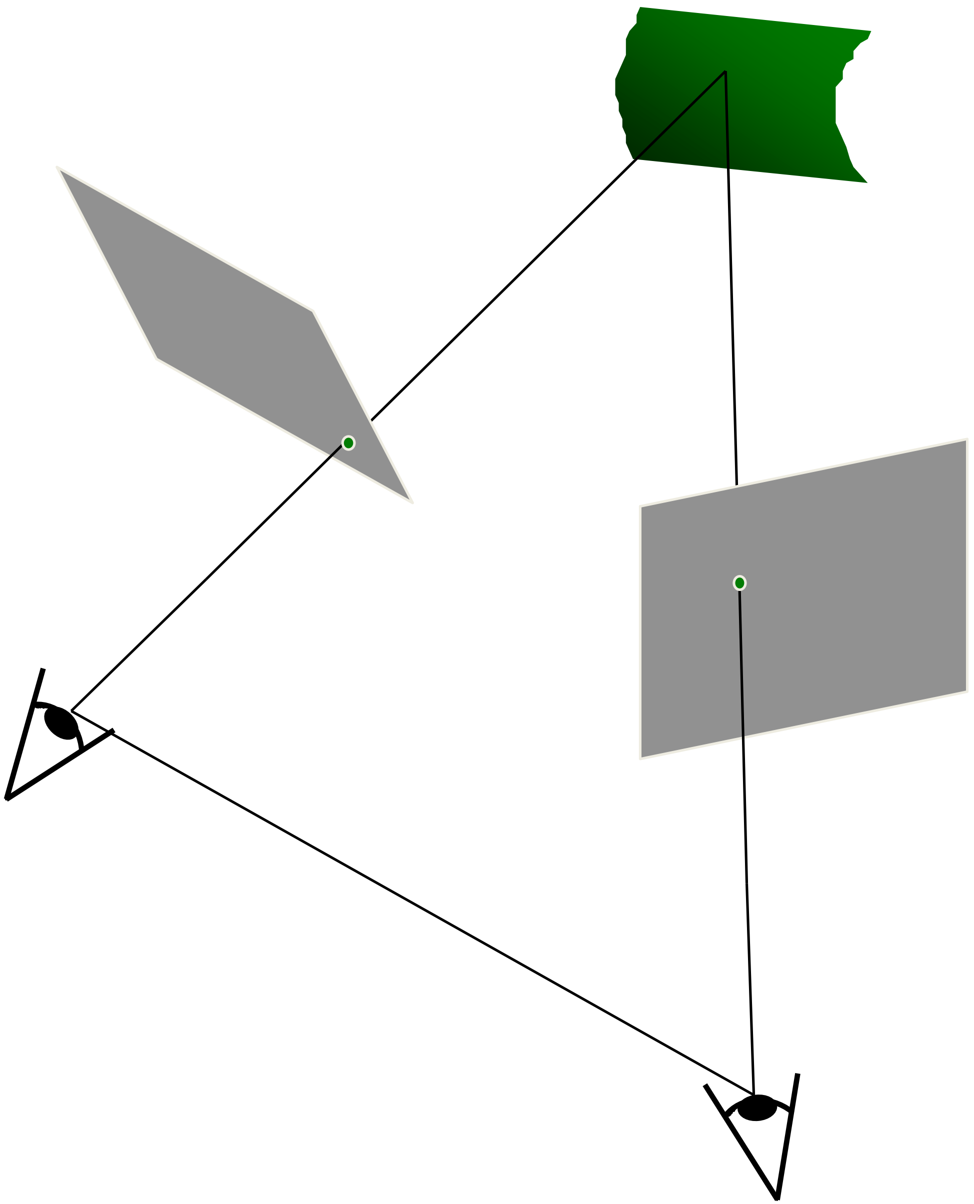
Can solve for F from point correspondences

- Each *(x, x')* pair gives one linear equation in entries of F

$$x' F x = 0$$

- F has 9 entries, but really only 7 degrees of freedom.
- With 8 points it is simple to solve for F, but it is also possible with 7. See [Marc Pollefey's notes]for a nice tutorial

Stereo image rectification



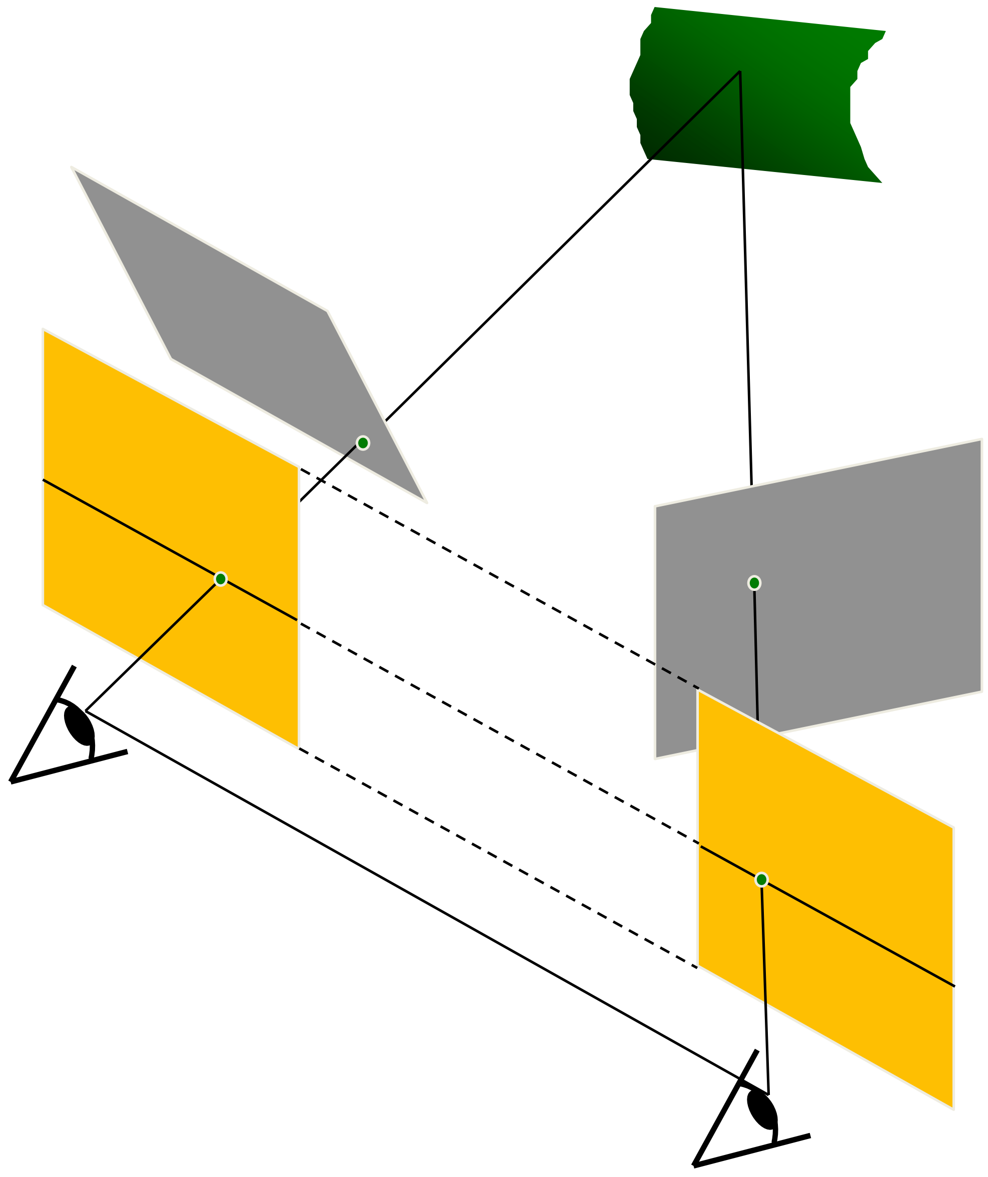
Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between camera centers

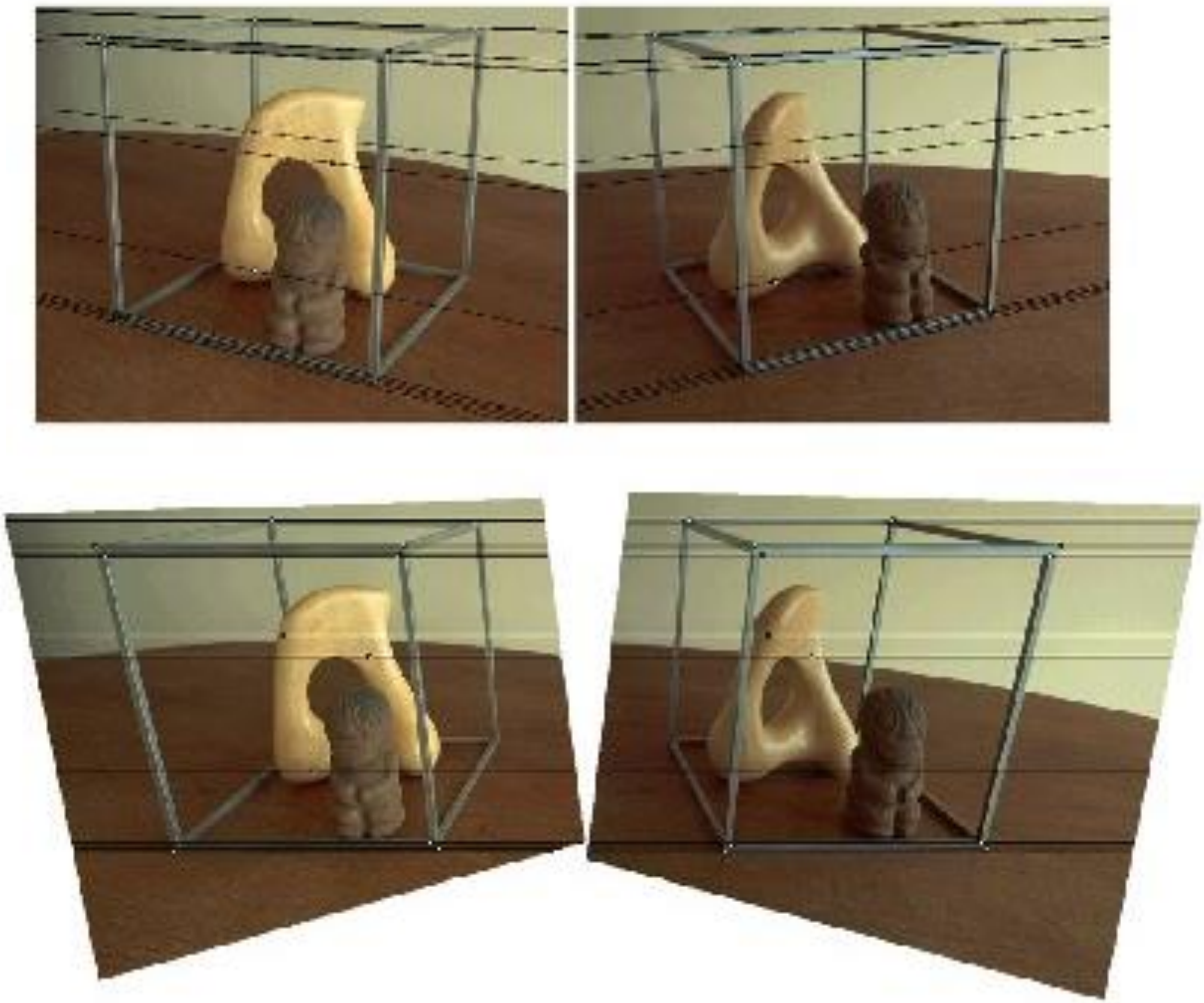
- Pixel motion is horizontal after this transformation

- Two homographies (3x3 transform), one for each input image reprojection

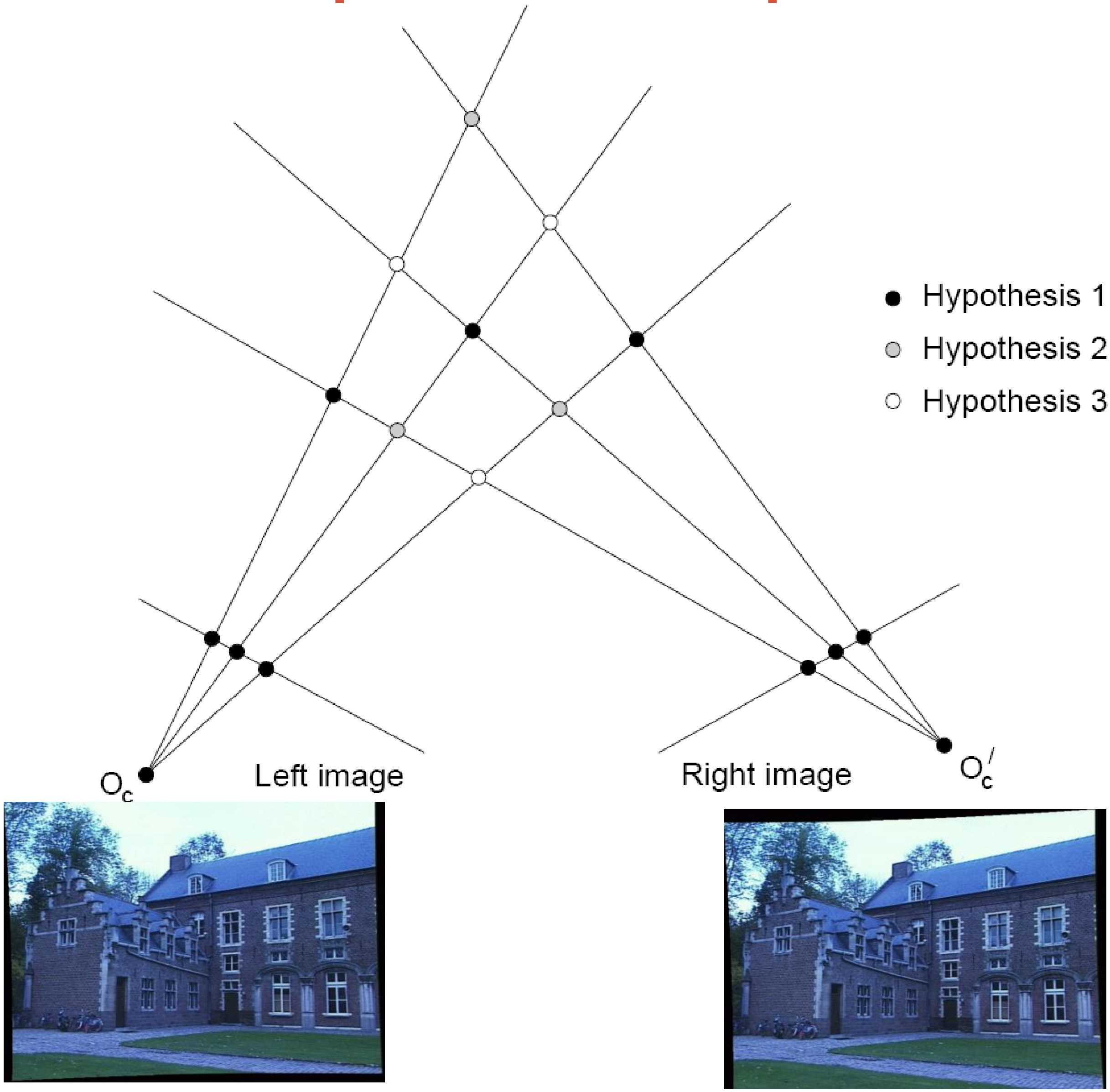
➢ C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). IEEE Conf. Computer Vision and Pattern Recognition, 1999.



Rectification example



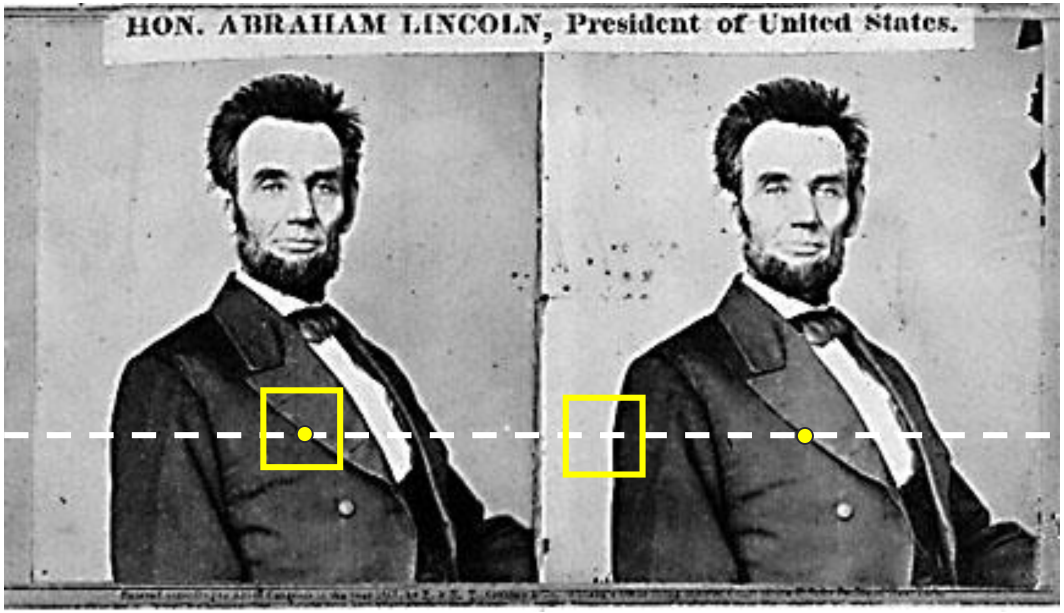
Correspondence problem



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

Figure from Gee & Cipolla 1999

Dense correspondence search



For each epipolar line:

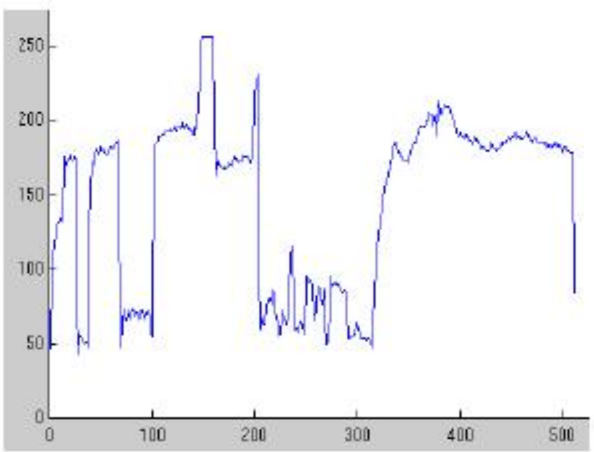
For each pixel / window in the left image:

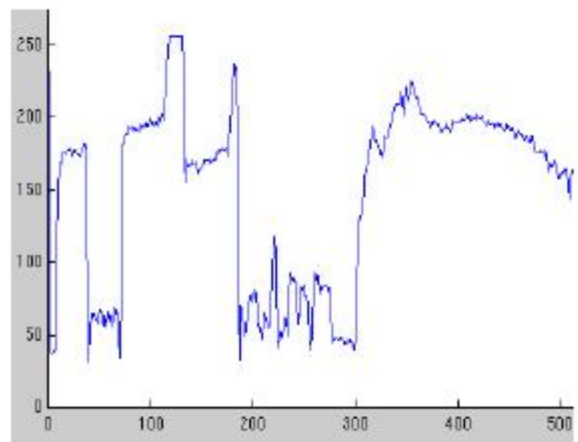
- Compare with every pixel / window on same epipolar line in right image
- Pick position with minimum match cost (e.g., SSD, normalized correlation)

Adapted from Li Zhang

Correspondence problem

Intensity profiles

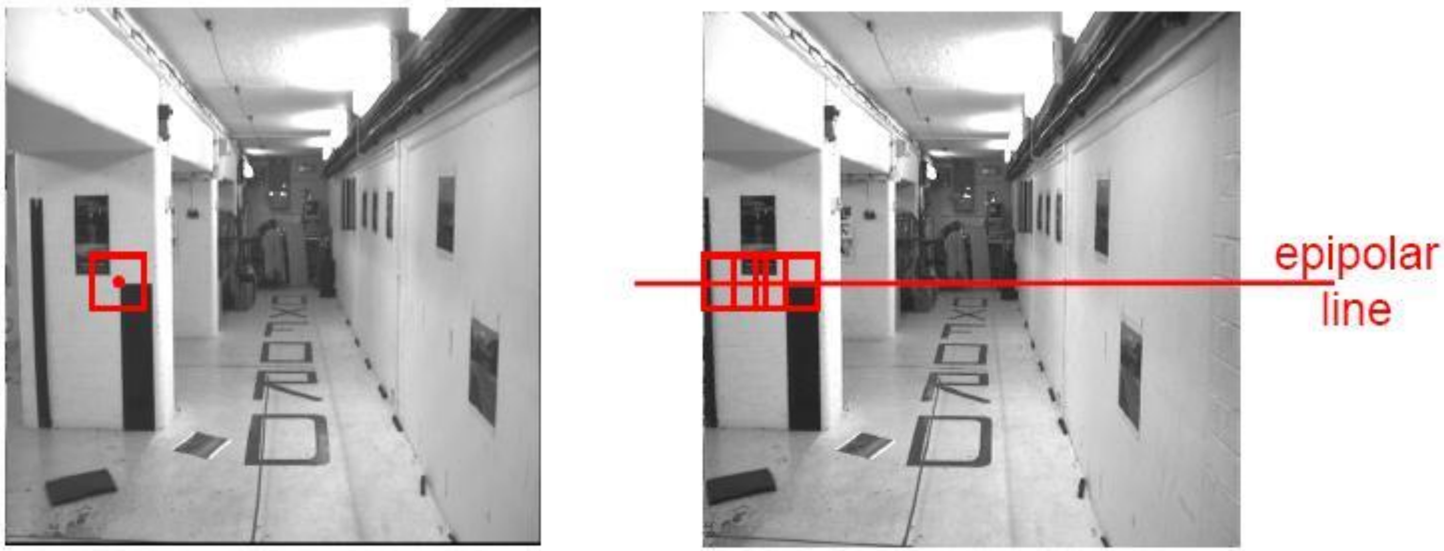




- Clear correspondence between intensities, but also noise and ambiguity

Source: Andrew Zisserman

Correspondence problem



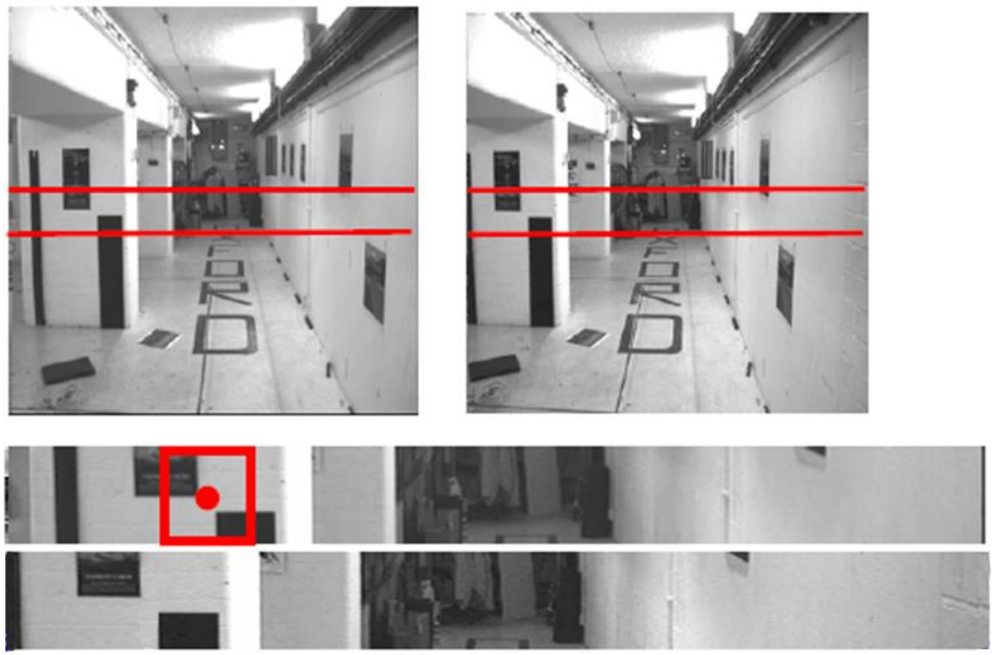
Neighborhoods of corresponding points are similar in intensity patterns.

Source: Andrew Zisserman

Correlation-based window matching

left image band (x)

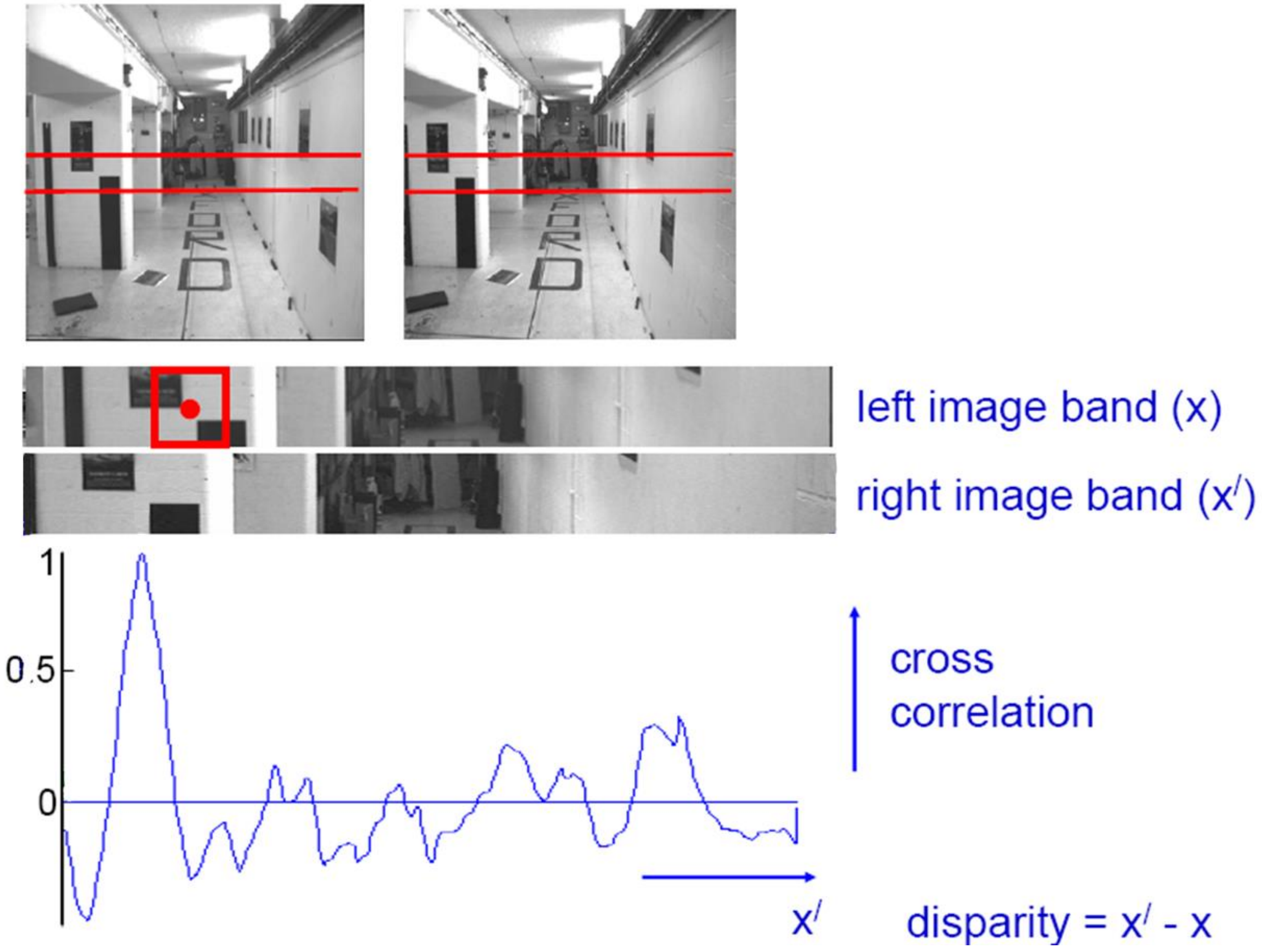
Correlation-based window matching



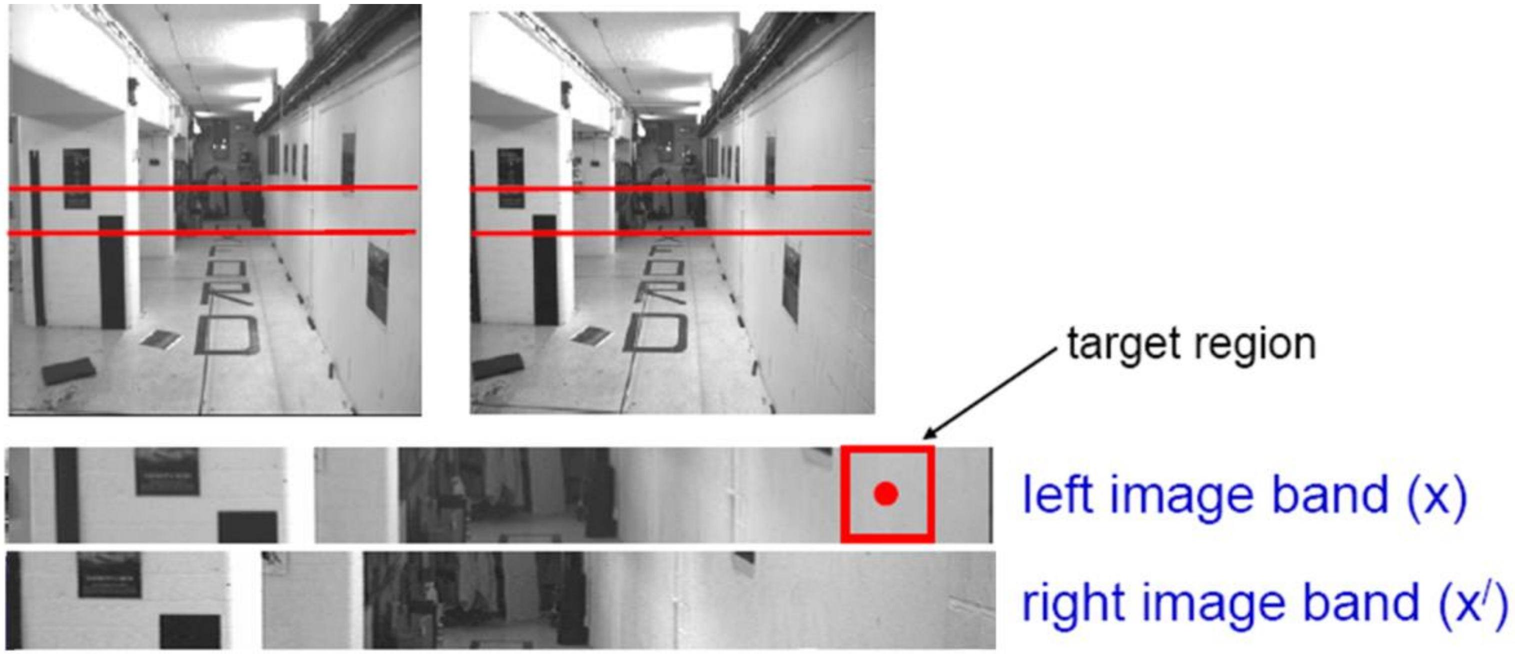
left image band (x)

right image band (x')

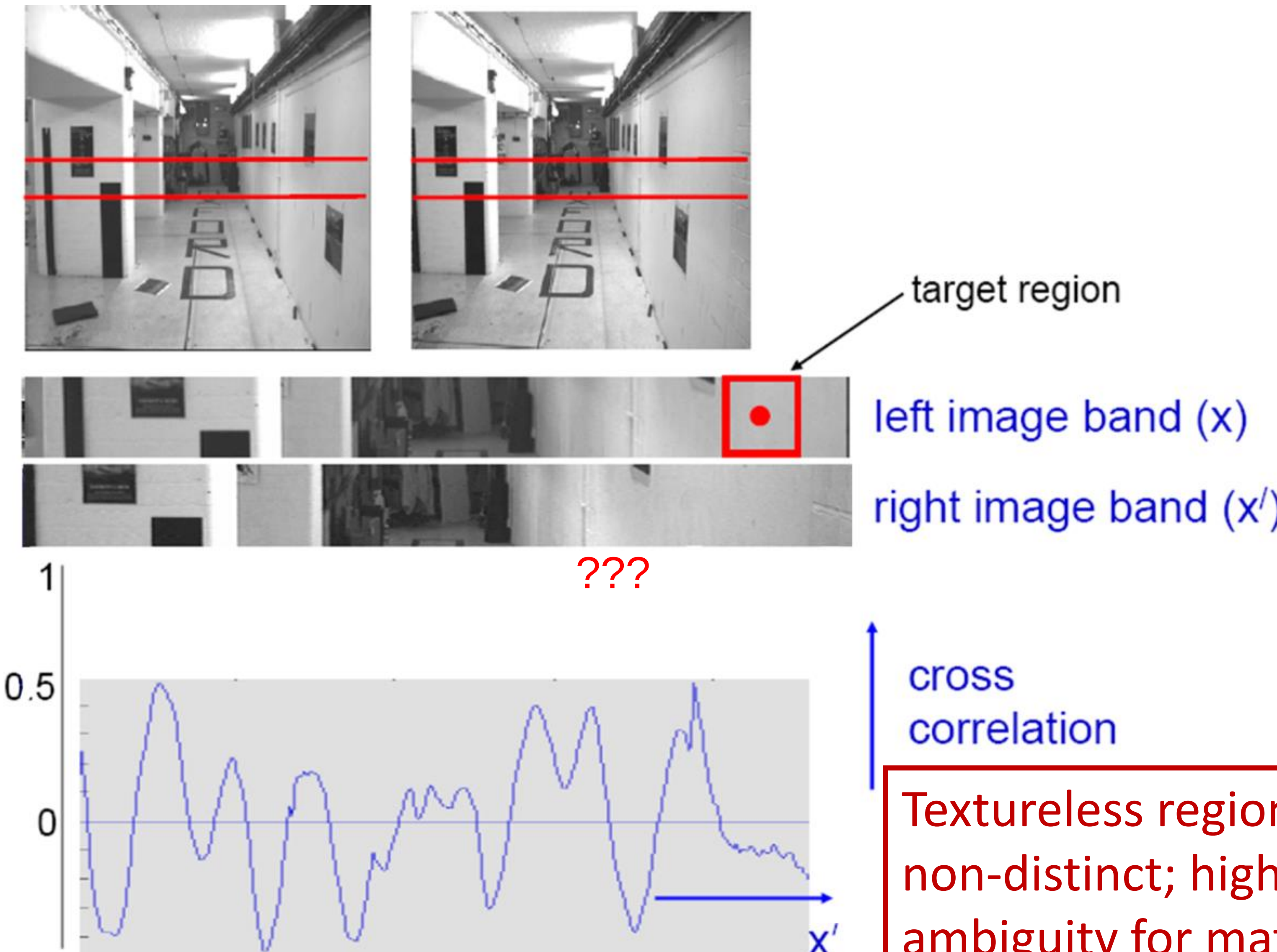
Correlation-based window matching



Correlation-based window matching



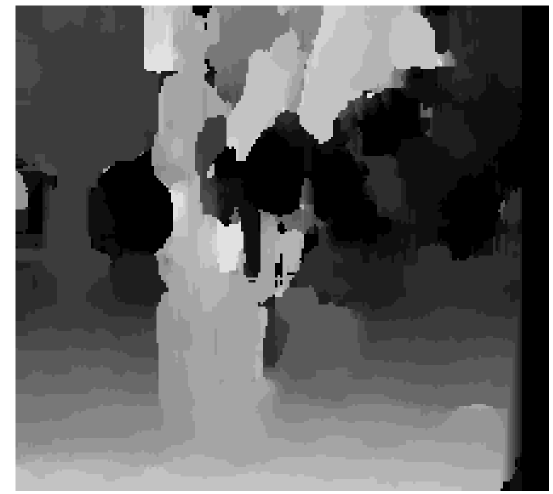
Correlation-based window matching



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

Effect of window size

W = 3



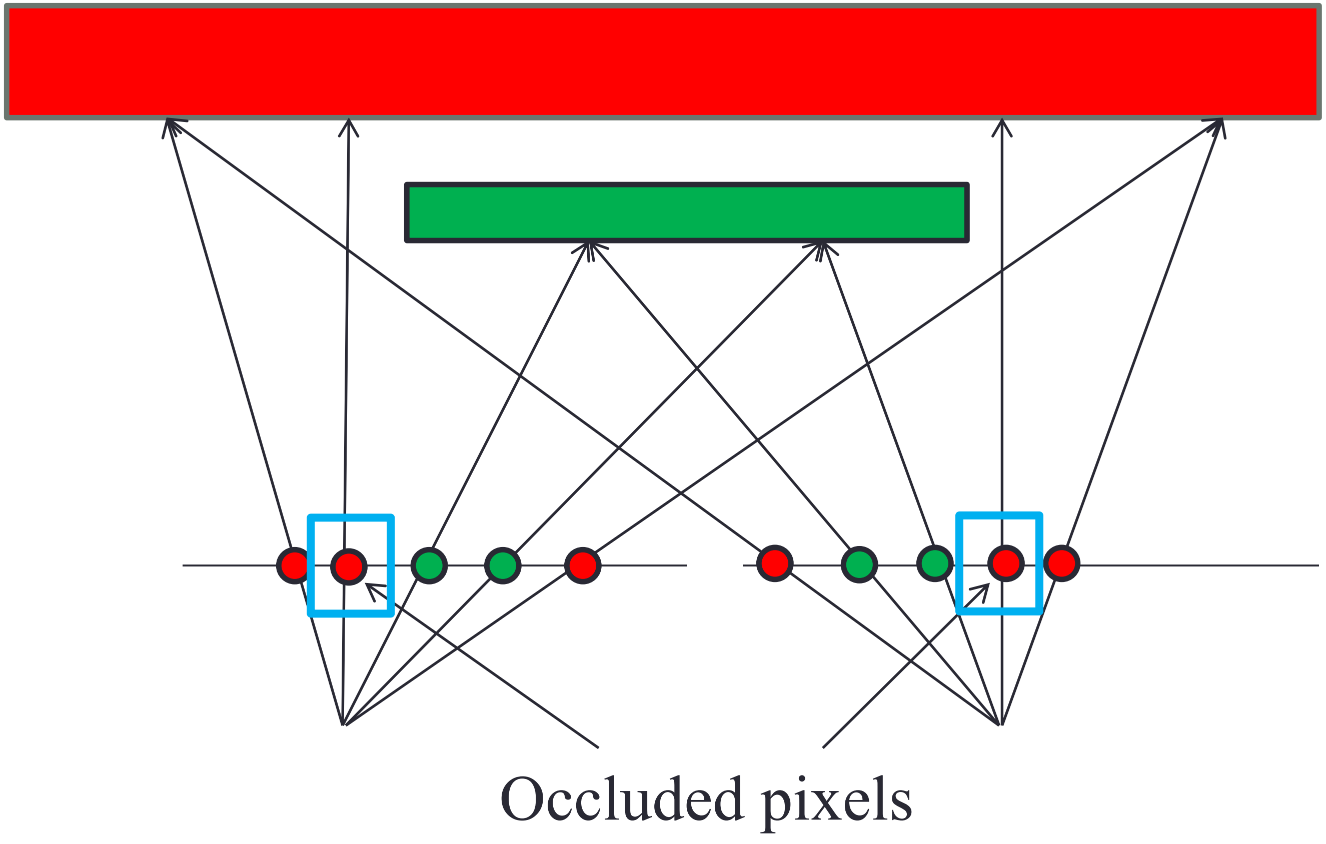
W = 20

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

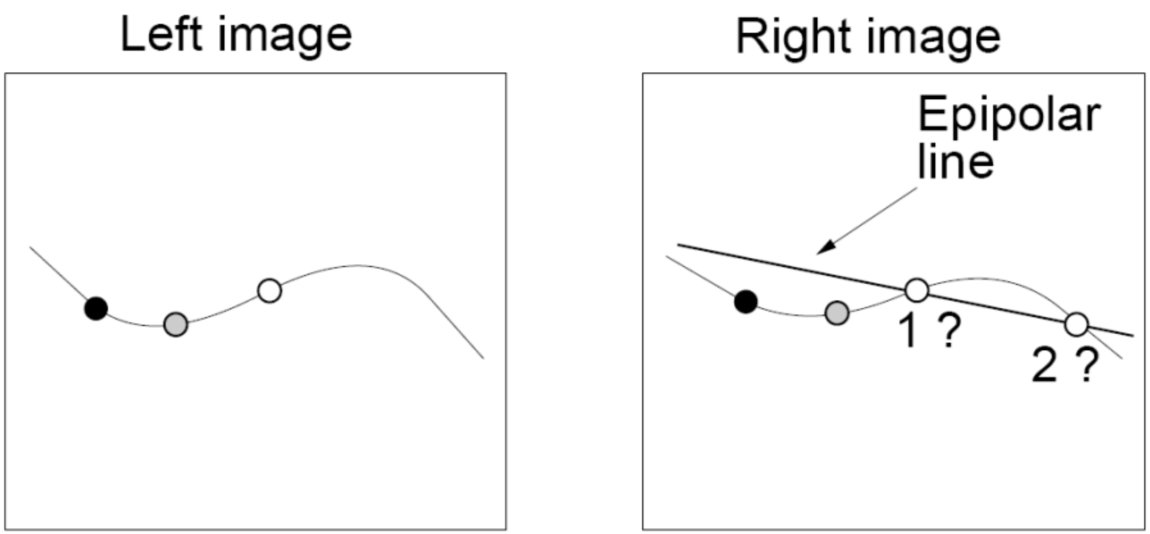
Problem: Occlusion

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



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

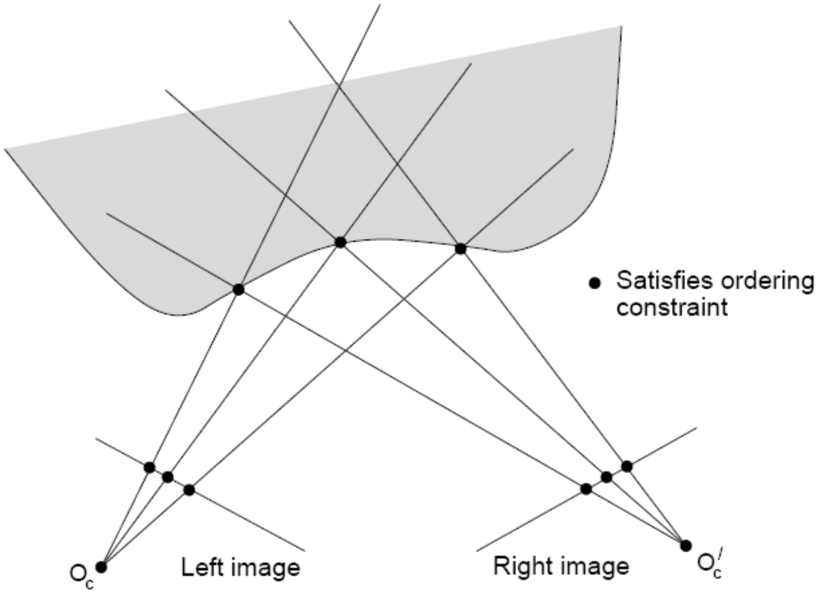
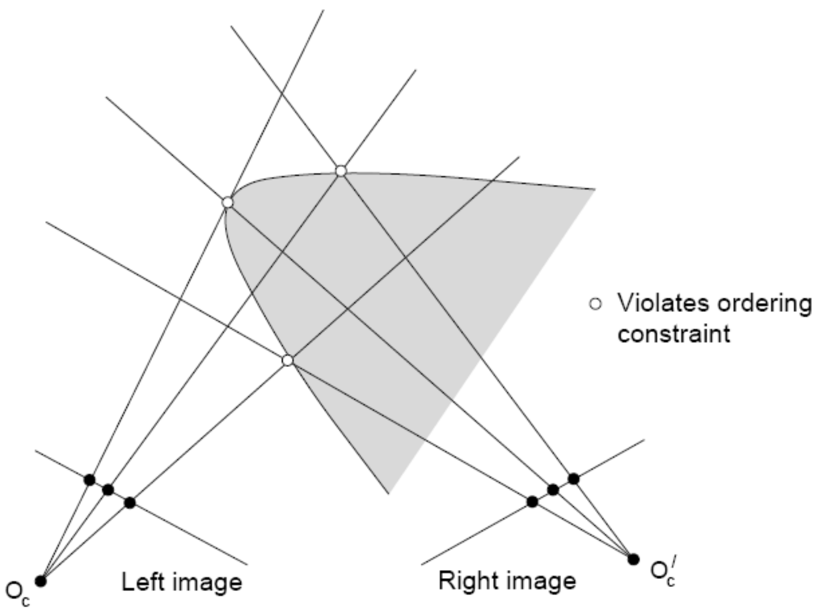
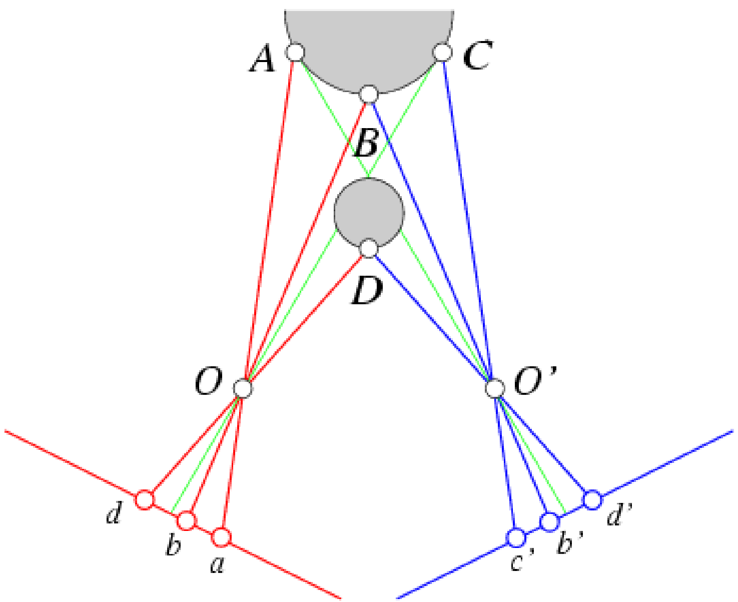


Figure from Gee & Cipolla 1999

Ordering constraint

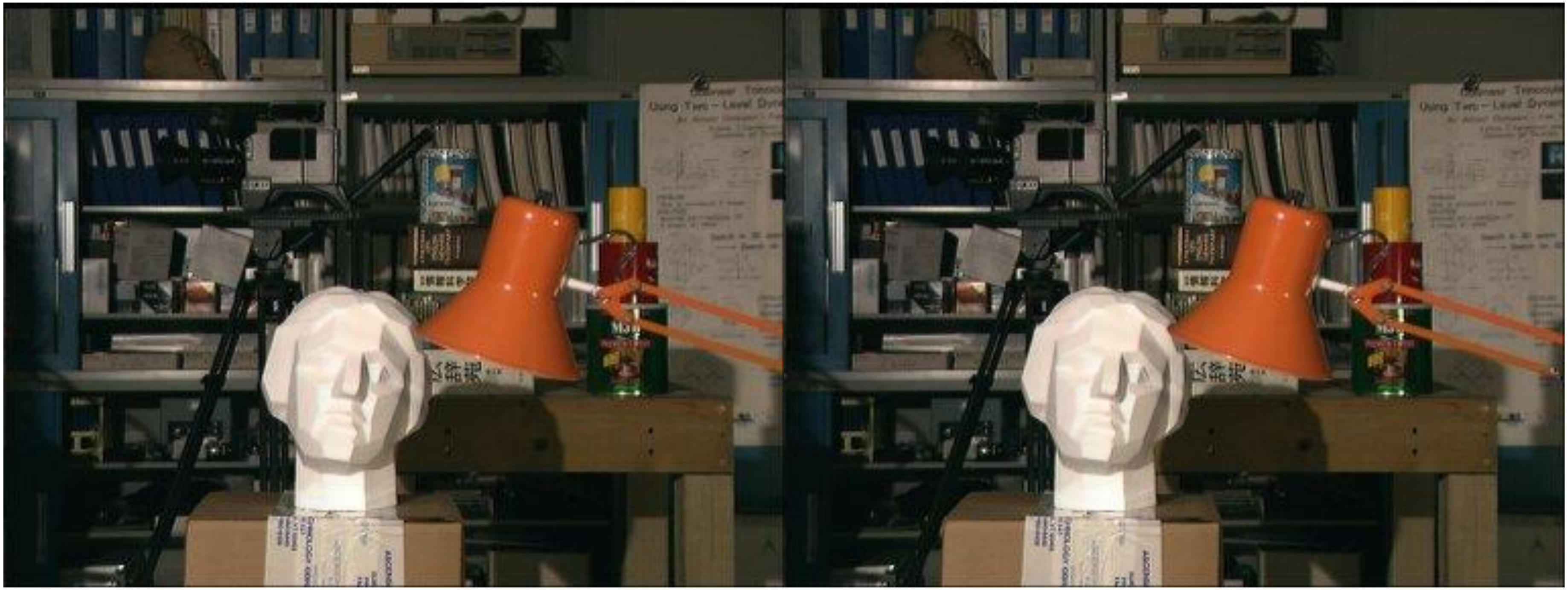
- Won’t always hold, e.g. consider transparent object, or an occluding surface



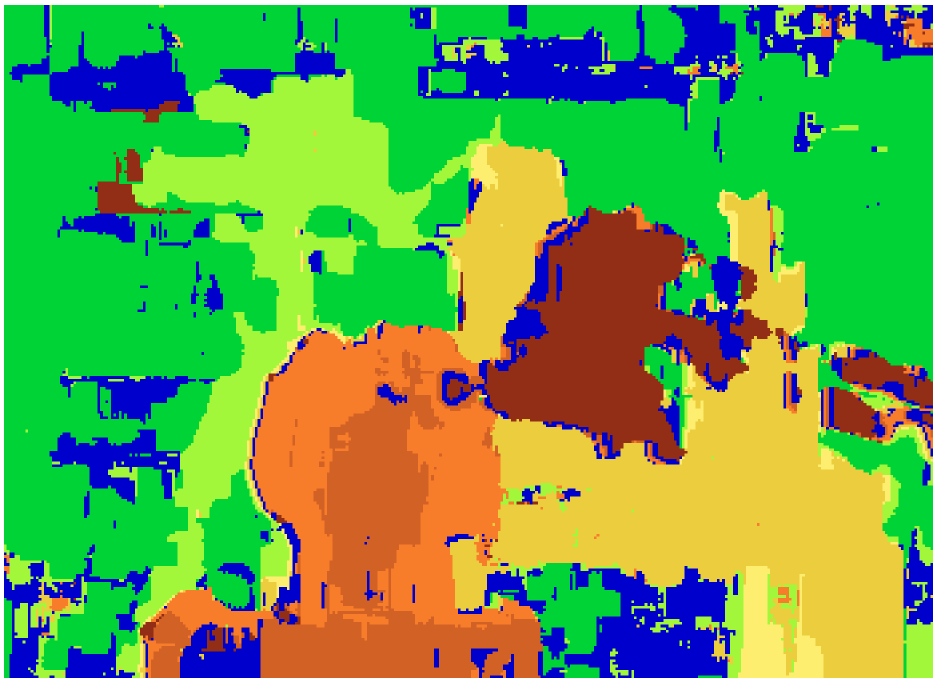


Figures from Forsyth & Ponce

Stereo – Tsukuba test scene (now old)



Results with window search



Window-based matching
(best window size)

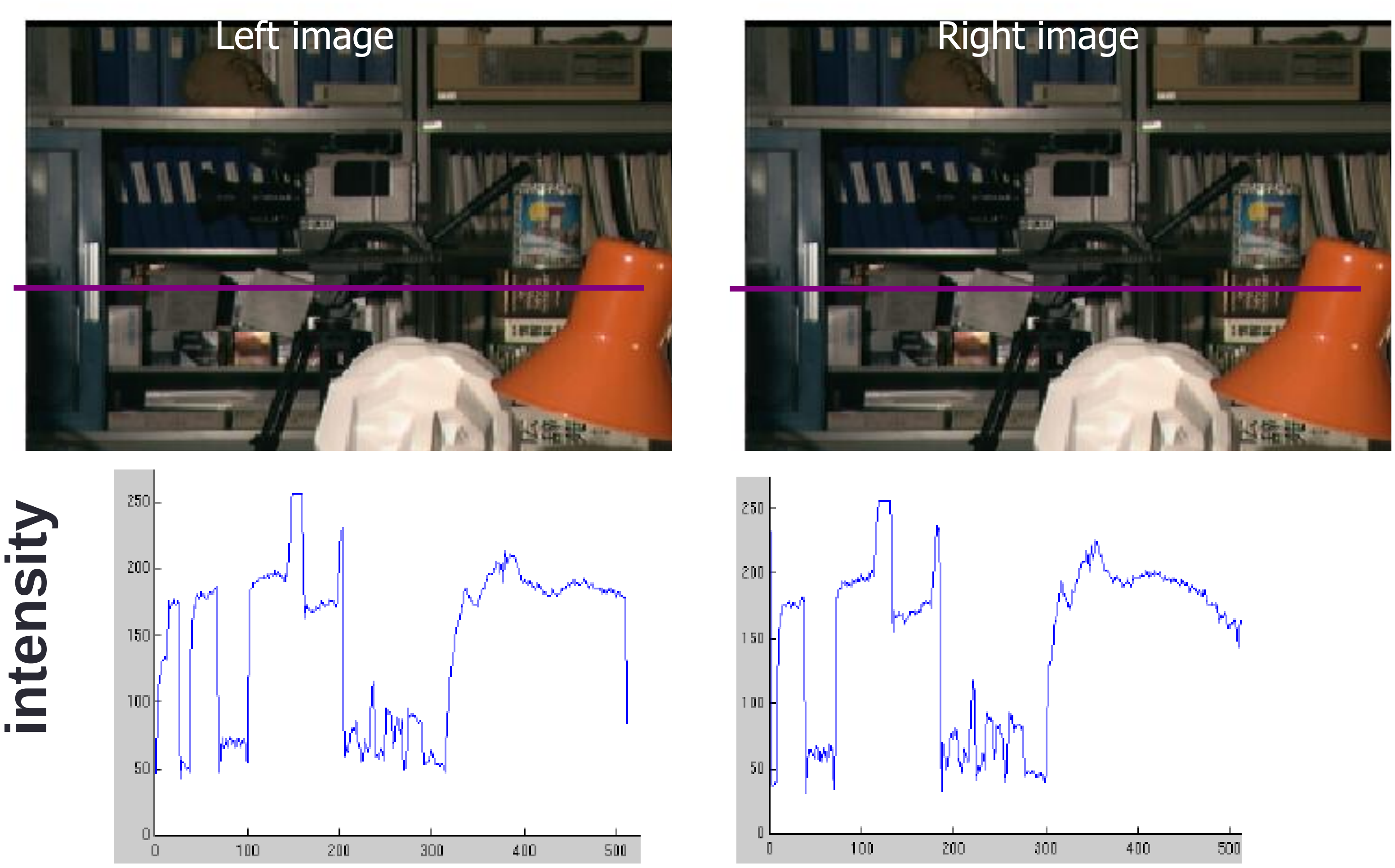
‘Ground truth’

Better solutions

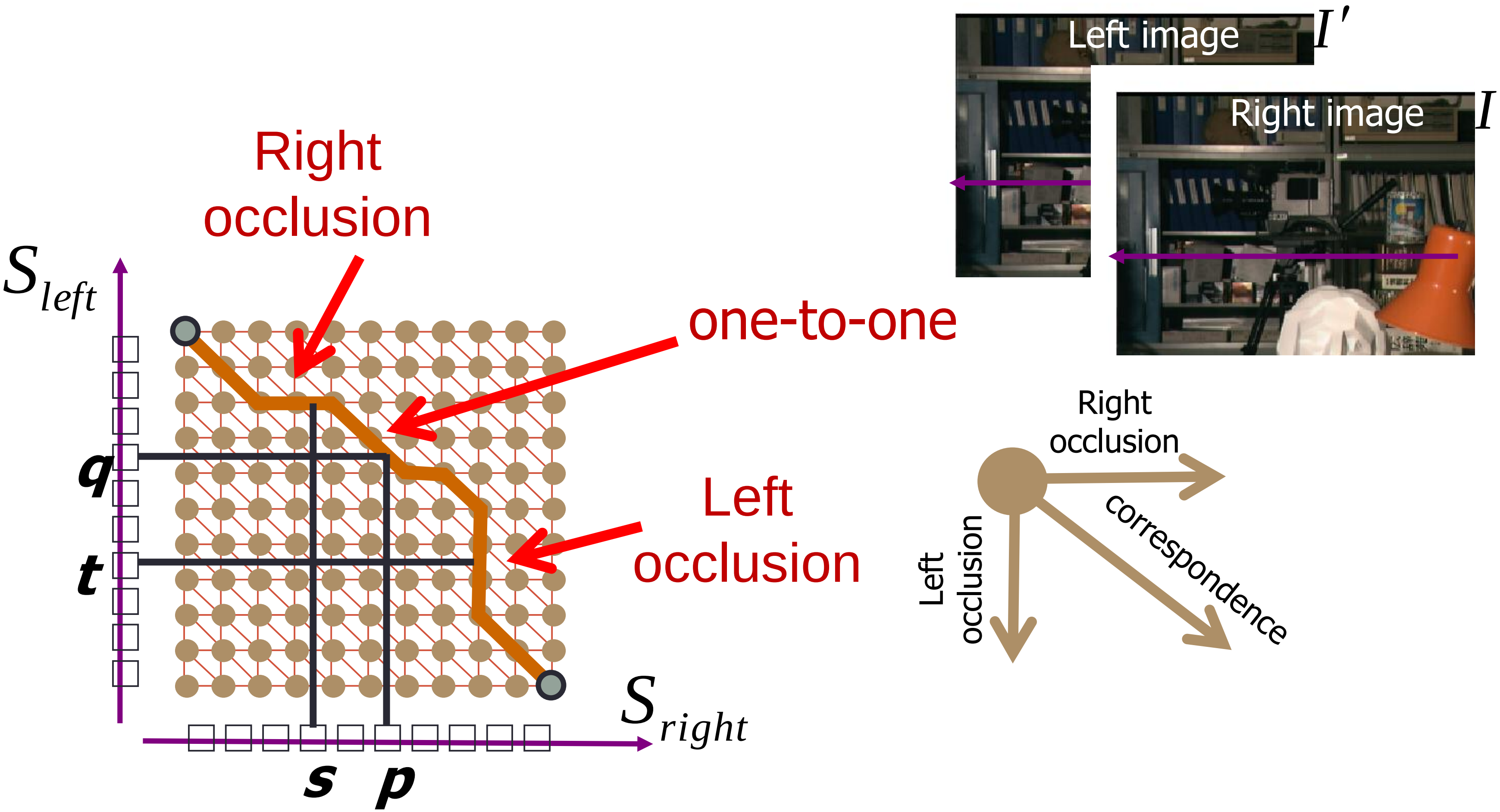
- Beyond individual correspondences to estimate disparities:
- Optimize correspondence assignments jointly
  - Scanline at a time (DP)
  - Full 2D grid (graph cuts)

Scanline stereo

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



"Shortest paths" for scan-line stereo



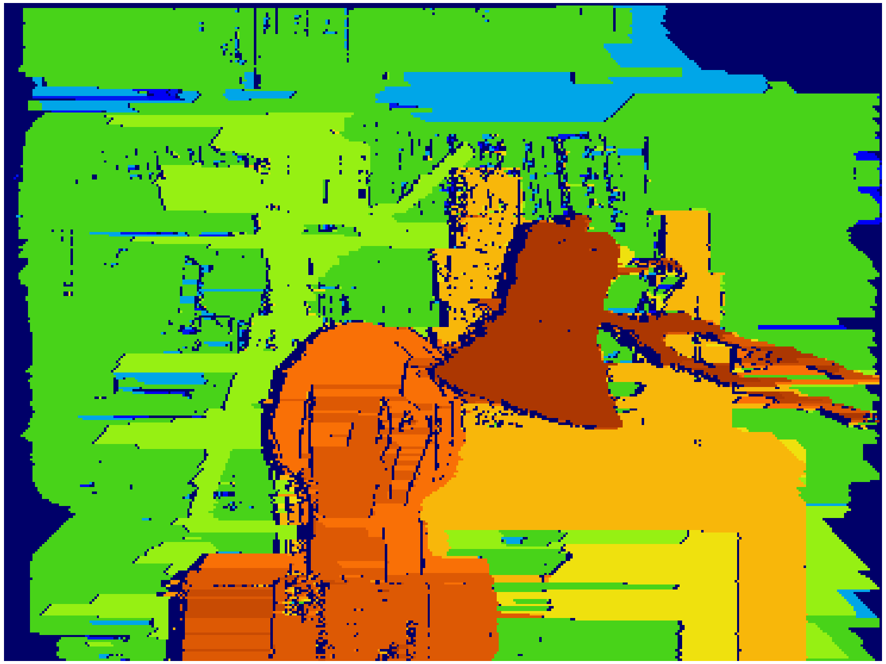
Can be implemented with dynamic programming

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

Slide credit: Y. Boykov

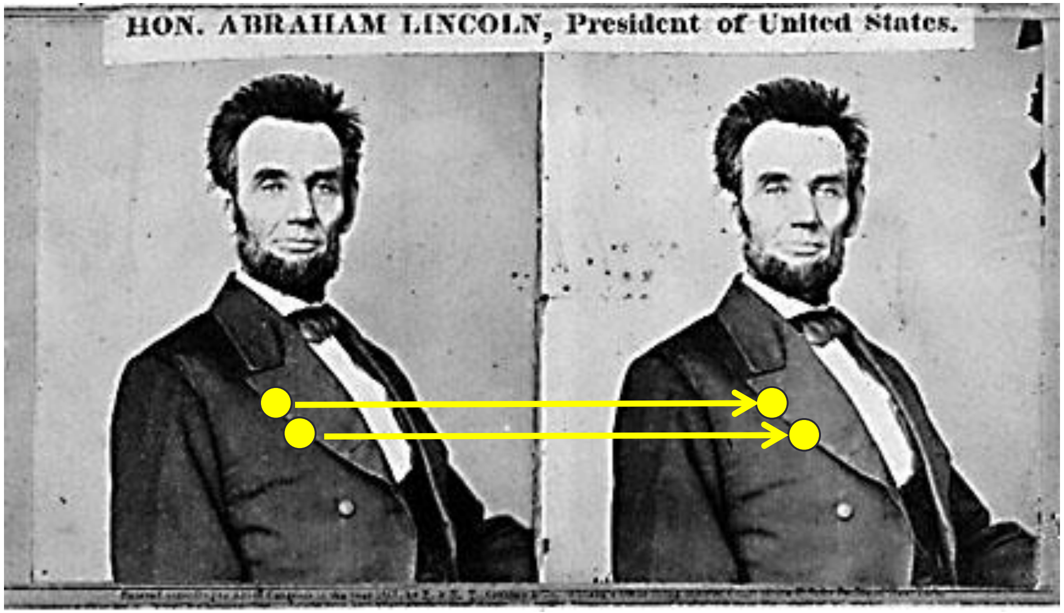
Coherent stereo on 2D grid

- Scanline stereo generates streaking artifacts



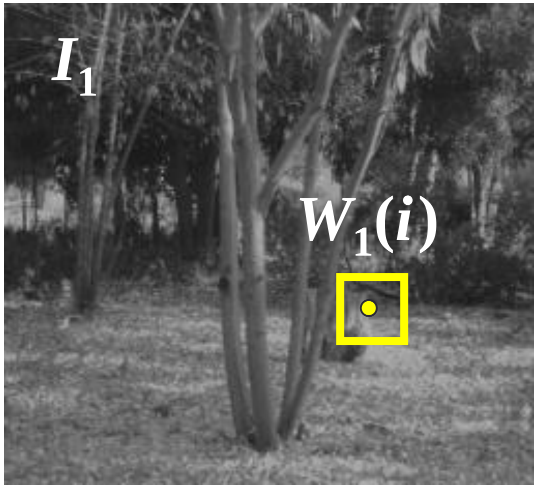
- Can’t use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid

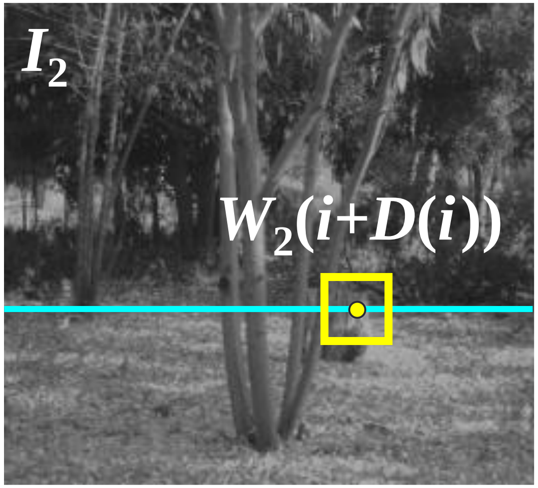
Stereo as energy minimization

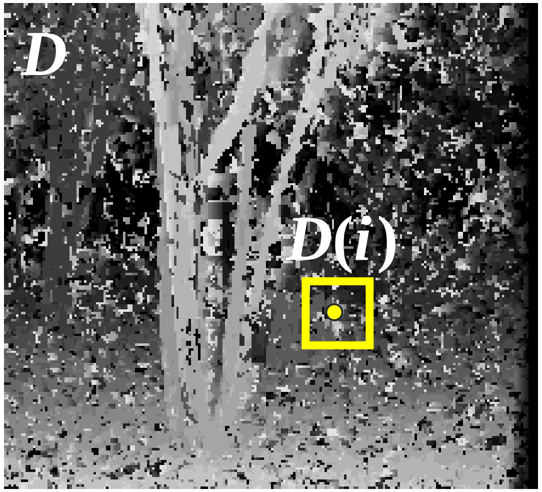


- What defines a good stereo correspondence?
  1. Match quality
     - Want each pixel to find a good match in the other image
  2. Smoothness
     - If two pixels are adjacent, they should (usually) move about the same amount

Stereo matching as energy minimization







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

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

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

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…

Graph cut method

Ground truth

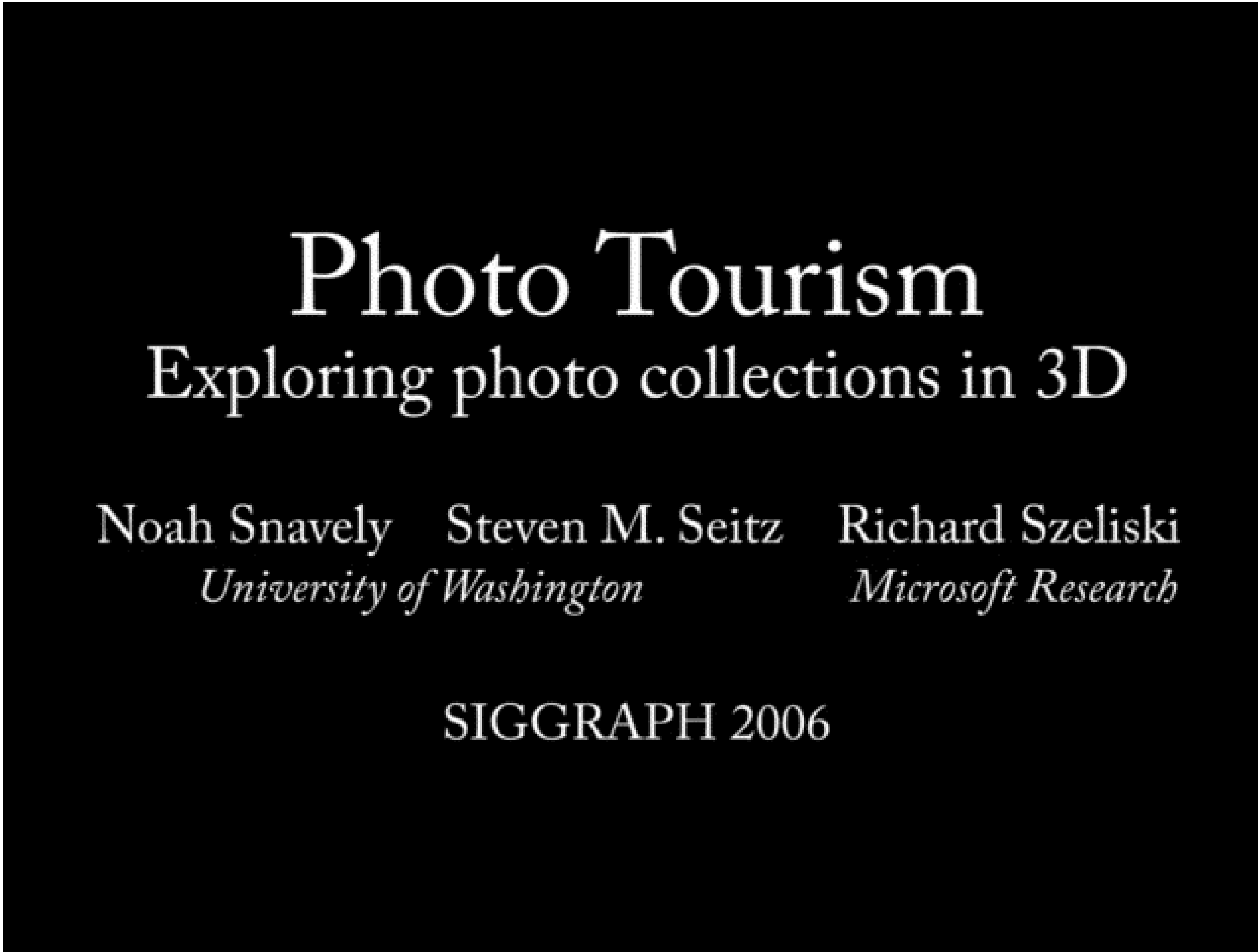
Boykov et al., [Fast Approximate Energy Minimization via Graph Cuts](), International Conference on Computer Vision, September 1999.

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

Challenges

- Low-contrast ‘textureless’ image regions
- Occlusions
- Violations of brightness constancy
  - Specular reflections
- Really large baselines
  - Foreshortening and appearance change
- Camera calibration errors

SIFT + Fundamental Matrix + RANSAC + Sparse correspondence



SIFT + Fundamental Matrix + RANSAC + dense correspondence

Despite their scale invariance and robustness to appearance changes, SIFT features are *local* and do not contain any global information about the image or about the location of other features in the image. Thus feature matching based on SIFT features is still prone to errors. However, since we assume that we are dealing with rigid scenes, there are strong geometric constraints on the locations of the matching features and these constraints can be used to clean up the matches. In particular, when a rigid scene is imaged by two pinhole cameras, there exists a 3×3 matrix F, the *Fundamental matrix*, such that corresponding points x_{ij} and x_{ik} (represented in homogeneous coordinates) in two images j and k satisfy[10]:

$$x_{ij}^{\top} F x_{ij} = 0. \qquad (3)$$

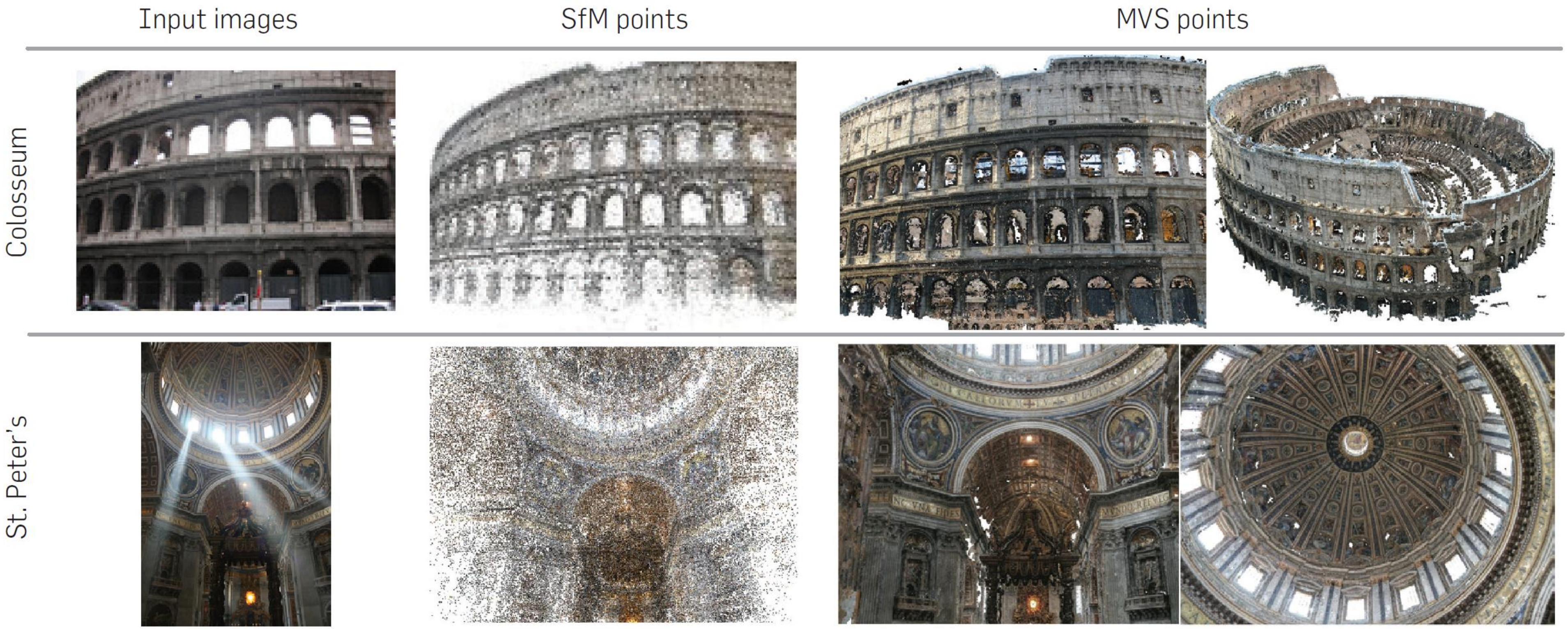
A common way to impose this constraint is to use a greedy randomized algorithm to generate suitably chosen random estimates of F and choose the one that has the largest support among the matches, i.e., the one for which the most matches satisfy (3). This algorithm is called Random Sample Consensus (RANSAC)[6] and is used in many computer vision problems.

Building Rome in a Day

By Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, Richard Szeliski
Communications of the ACM, Vol. 54 No. 10, Pages 105-112

2009

SIFT + Fundamental Matrix + RANSAC + dense correspondence



Building Rome in a Day

By Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, Richard Szeliski
Communications of the ACM, Vol. 54 No. 10, Pages 105-112

SIFT + Fundamental Matrix + RANSAC + dense correspondence

The Visual Turing Test for Scene Reconstruction
Supplementary Video

Qi Shan[+] Riley Adams[+] Brian Curless[+]
Yasutaka Furukawa[*] Steve Seitz[+*]

[+]University of Washington [*]Google

3DV 2013

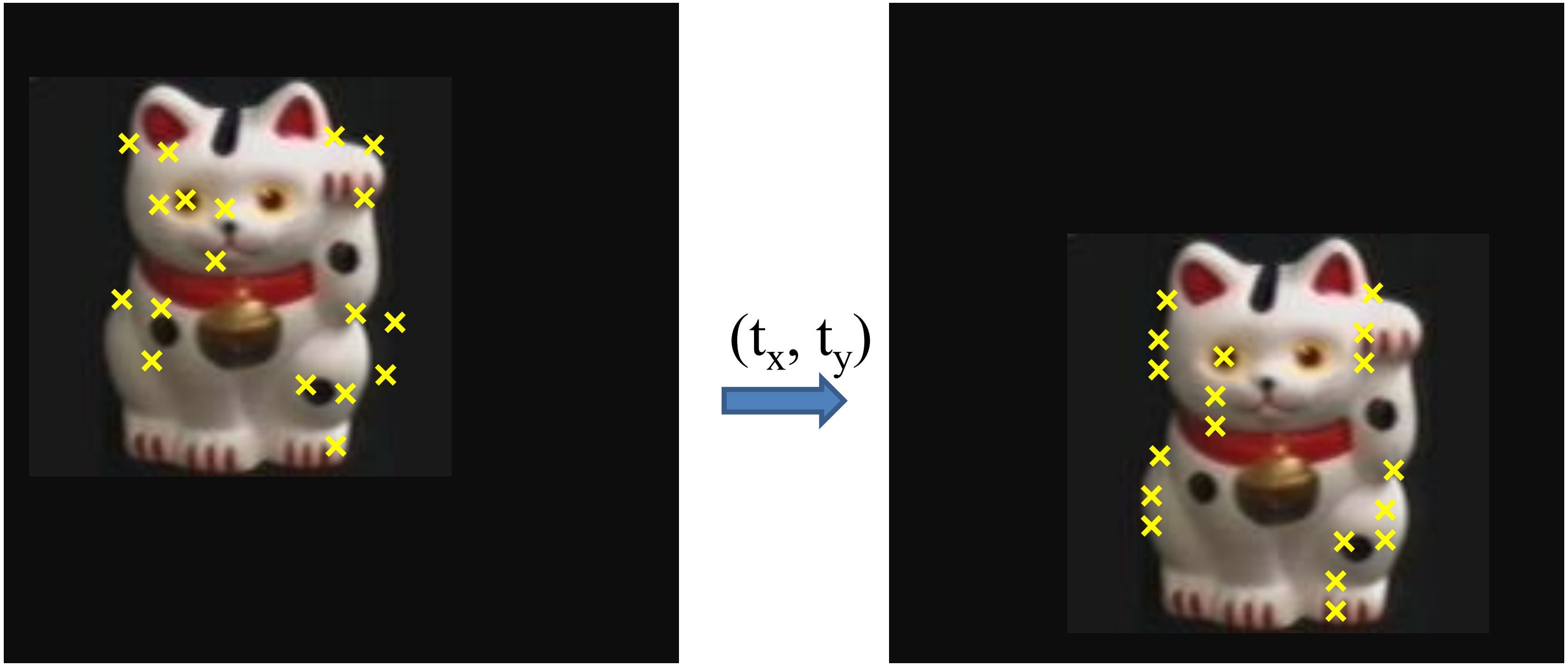
Once I have my depth map,
what can I do with it?

Measure.

Combine! (Reorganize?)

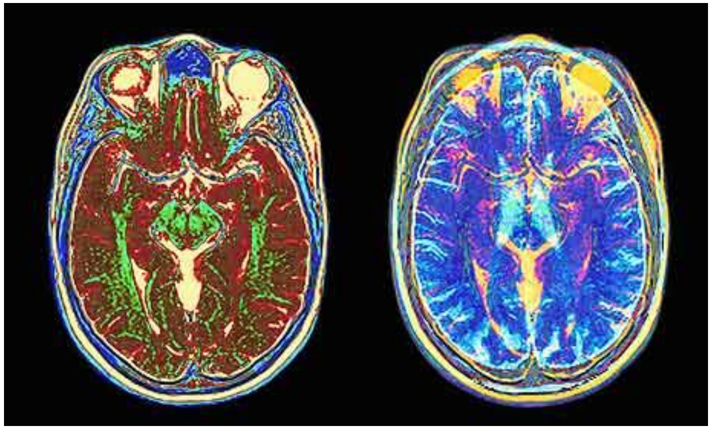
What if we want to align...
but we have no matched pairs?

- Hough transform and RANSAC not applicable

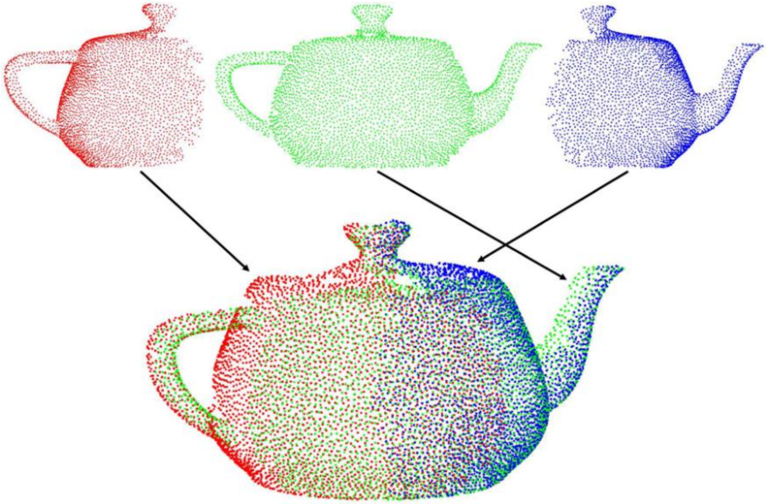


Problem: no initial guesses for correspondence

Important applications



Medical imaging: match brain scans or contours



Robotics: match point clouds

Kwok and Tang

ICP demonstration

Bouaziz et al.

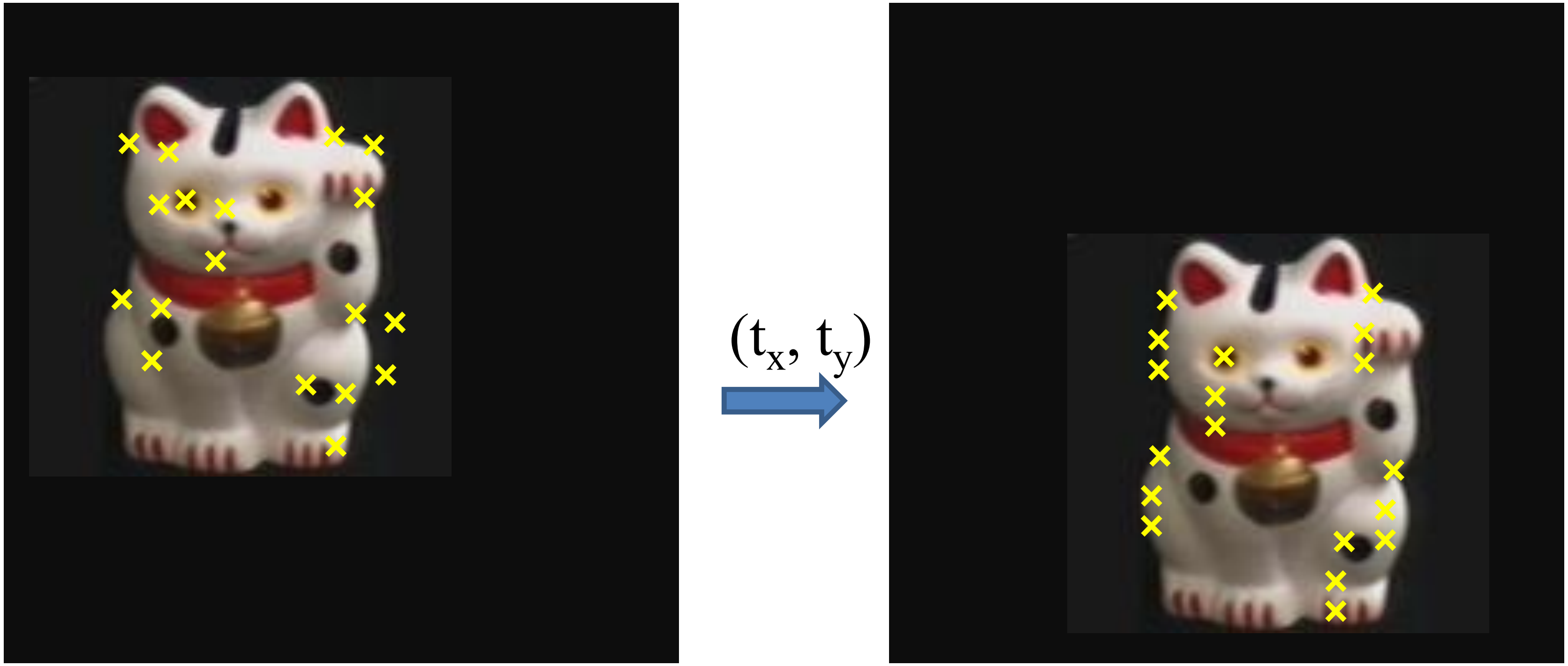
Iterative Closest Points (ICP) Algorithm

Goal:

Estimate transform between two dense point sets S_1 and S_2

1. **Initialize** transformation
   - Compute difference in mean positions, subtract
   - Compute difference in scales, normalize
2. **Assign** each point in S_1 to its nearest neighbor in S_2
3. **Estimate** transformation parameters T
   - Least squares or robust least squares, e.g., rigid transform
4. **Transform** the points in S_1 using estimated parameters T
5. **Repeat** steps 2-4 until change is very small (convergence)

Example: solving for translation



Problem: no initial guesses for correspondence

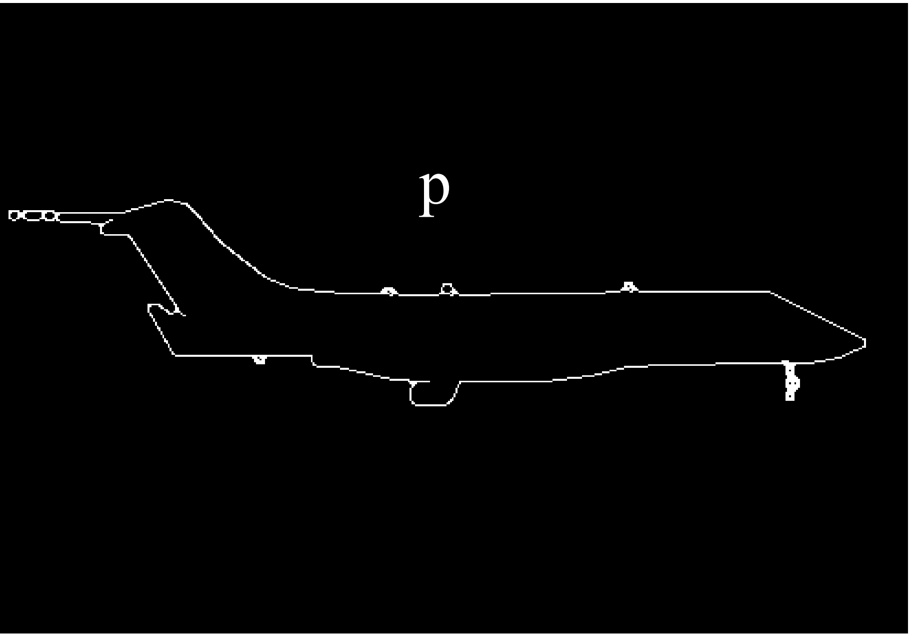
ICP solution

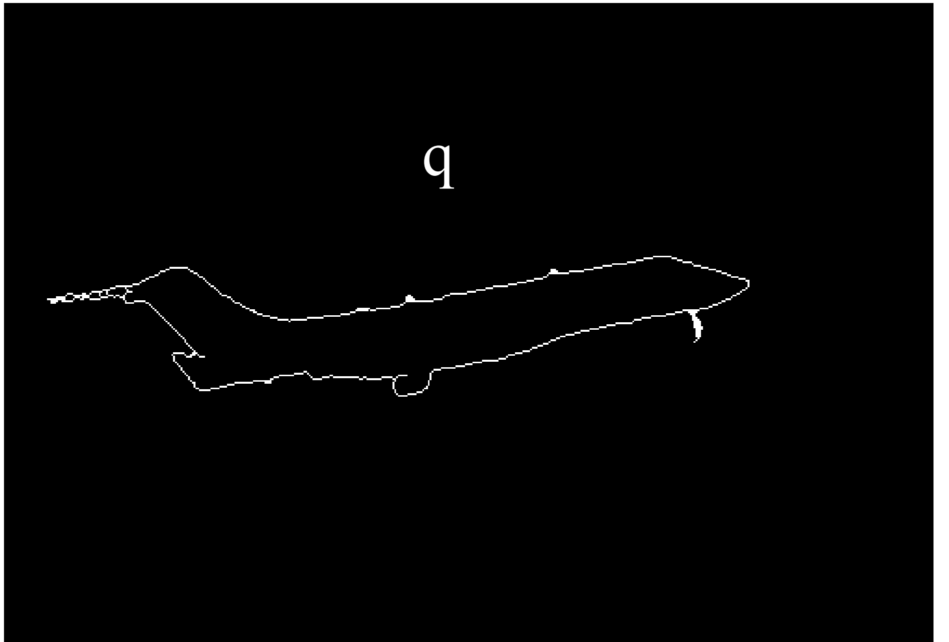
1. Initialize t by mean point translation
2. Find nearest neighbors for each point
3. Compute transform using matches
4. Move points using transform
5. Repeat steps 2-4 until convergence

$$\begin{bmatrix} x_i^B \\ y_i^B \end{bmatrix} = \begin{bmatrix} x_i^A \\ y_i^A \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Example: aligning boundaries

1. Extract edge pixels $p_1..p_n$ and $q_1..q_m$
2. Compute initial transformation (e.g., compute translation and scaling by center of mass, variance within each image)
3. Get nearest neighbors: for each point p_i find corresponding $\text{match(i)} = \underset{j}{\text{argmin}}\, dist(pi, qj)$
4. Compute transformation ***T*** based on matches
5. Transform points ***p*** according to ***T***
6. Repeat 3-5 until convergence



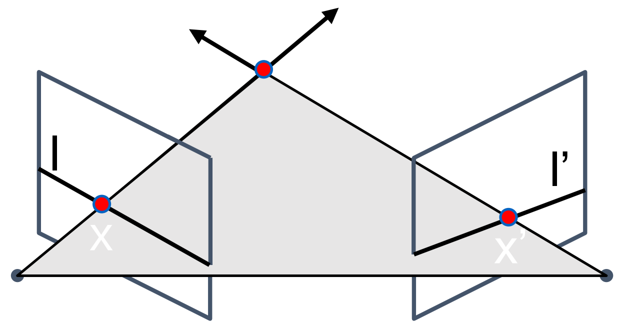


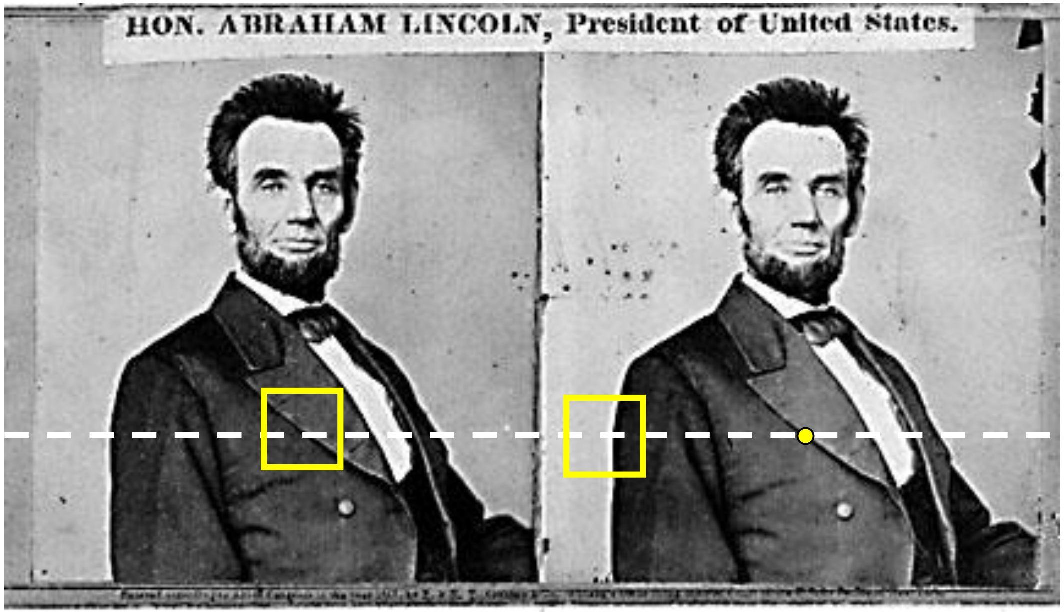
ICP demonstration

Bouaziz et al.

Stereo correspondence

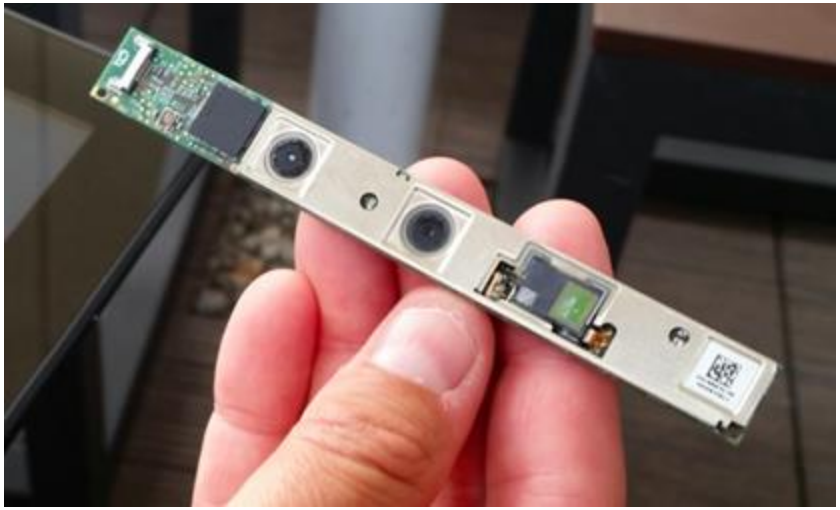
- Let x be a point in left image, x' in right image
- Epipolar relation
  - x maps to epipolar line l'
  - x' maps to epipolar line l





How does a depth camera work?

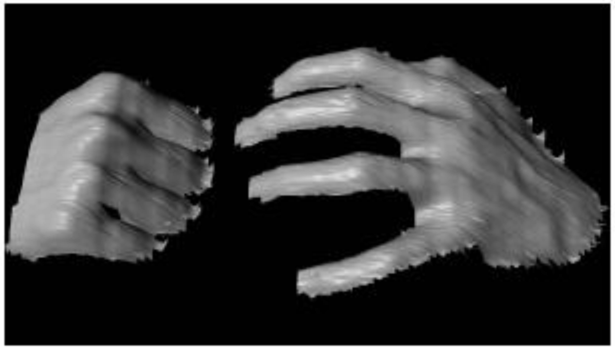
Microsoft Kinect v1



Intel laptop depth camera

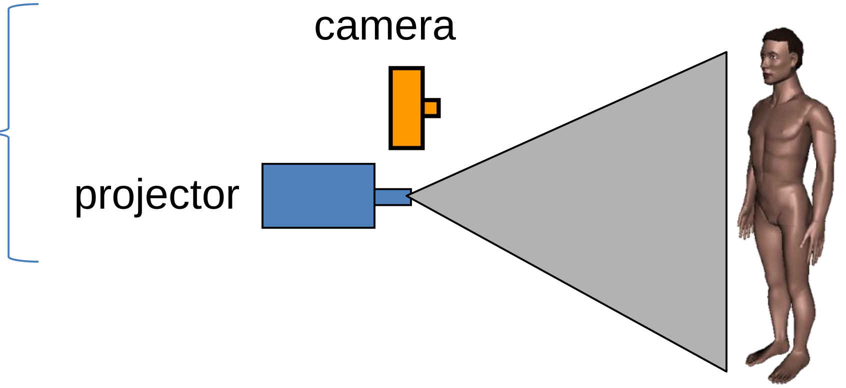
Active stereo with structured light





- Project "structured" light patterns onto the object
  - Simplifies the correspondence problem
  - Allows us to use only one camera

Stereo system!



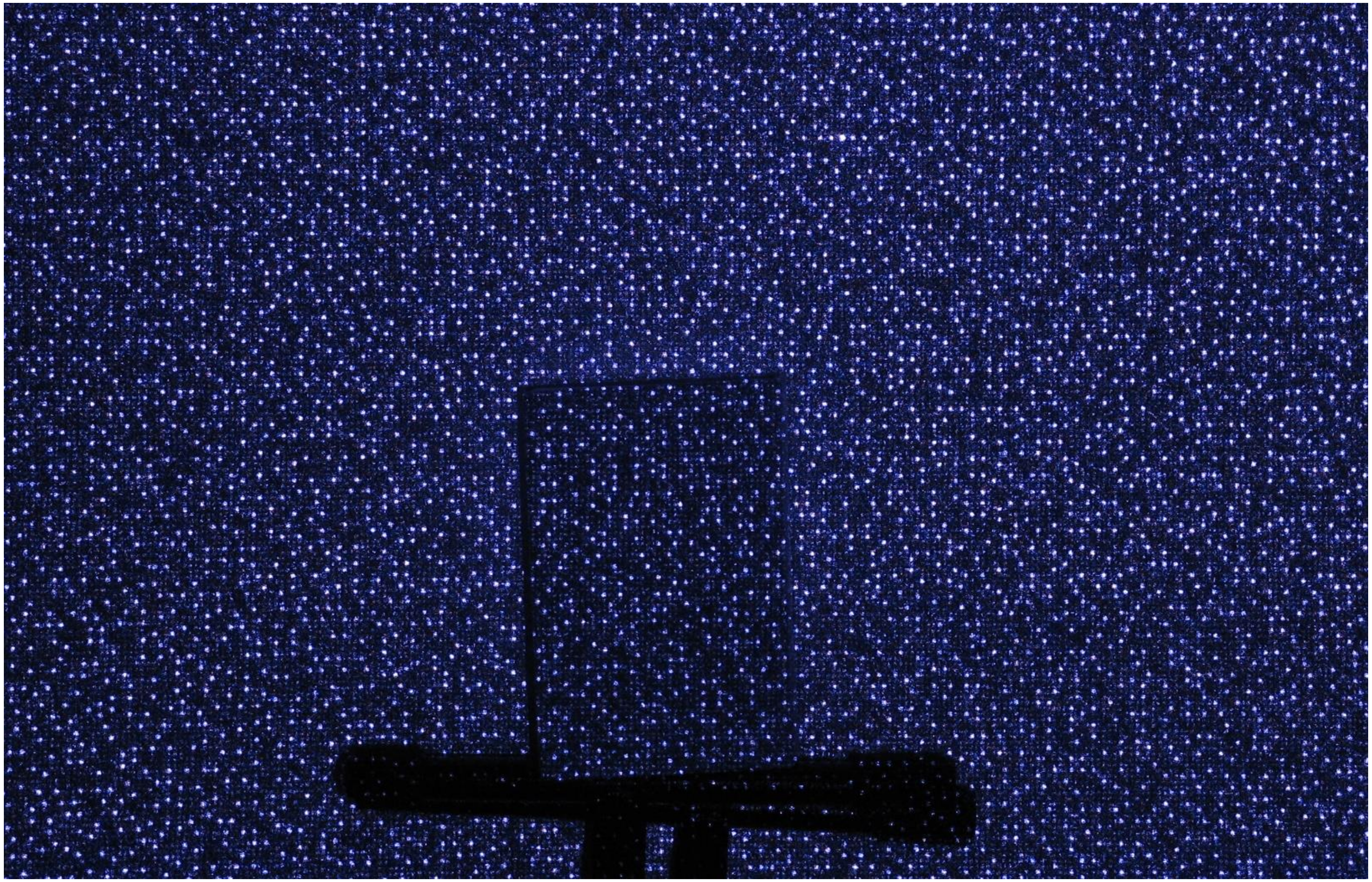
L. Zhang, B. Curless, and S. M. Seitz. [Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming.]() *3DPVT* 2002

Kinect: Structured infrared light

http://bbzippo.wordpress.com/2010/11/28/kinect-in-infrared/

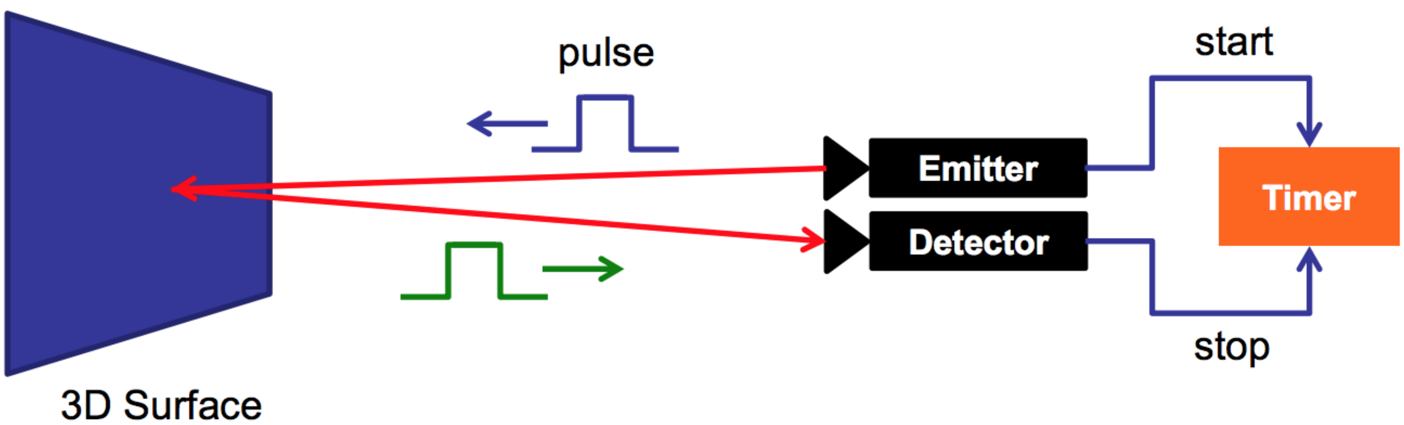
How does a depth camera work?

Stereo in infrared.



Time of Flight (Kinect V2)

- Depth cameras in HoloLens use *time of flight*
  - “SONAR for light”
  - Emit light of a known wavelength, and time how long it takes for it to come back



With either technique…

…I gain depth maps over time.

Optex Depth Camera Based on Canesta Solution

Sparse ICP

Sofien Bouaziz   *Andrea Tagliasacchi*   *Mark Pauly*

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

BundleFusion: Real-time Globally Consistent 3D Reconstruction using Online Surface Re-integration

Angela Dai[1] *Matthias Nießner*[1]
Michael Zollhöfer[2] *Shahram Izadi*[3]
Christian Theobalt[2]

[1]Stanford University
[2]Max Planck Institute for Informatics
[3]Microsoft Research

(*contains audio*)

ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes

Angela Dai   Angel X. Chang   Manolis Savva   Maciej Halber
Thomas Funkhouser   Matthias Nießner

Stanford University
Princeton University
Technical University of Munich

CVPR 2017 (Spotlight)