Lecture 19: Stereo

Instructor: Roni Sengupta

ULA: Andrea Dunn Beltran, William Li,
Liujie Zheng

Course Website:
Scan Me!

Recap

Camera Model

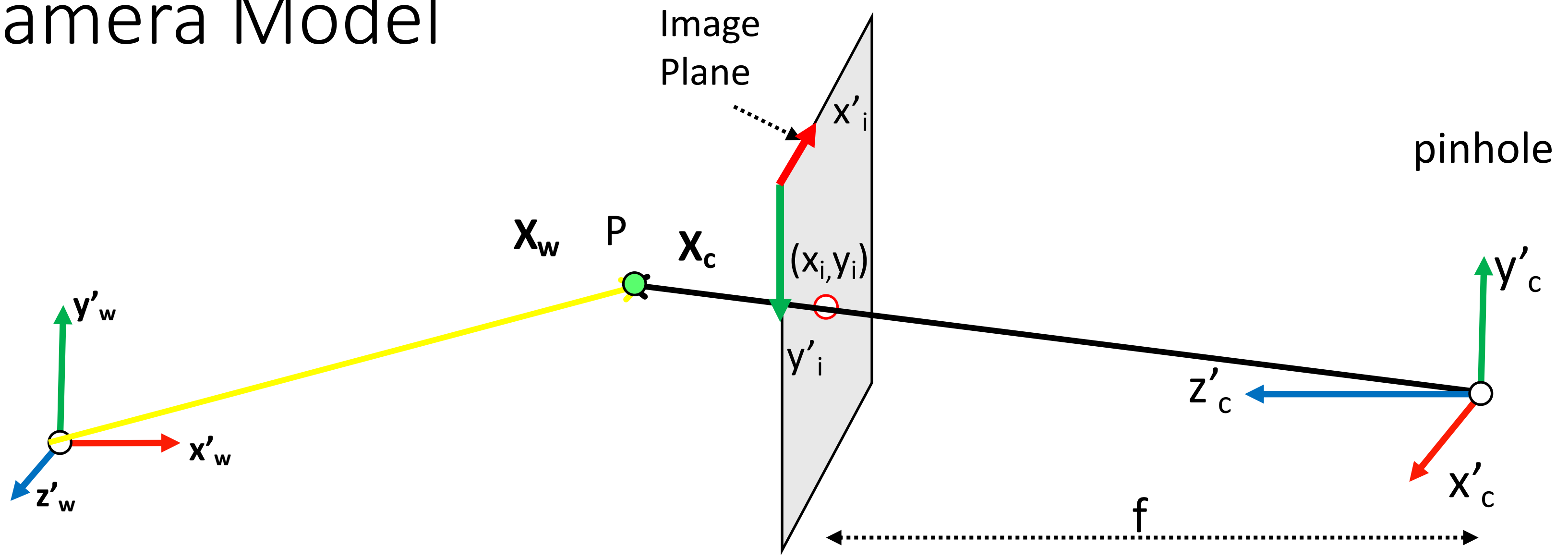


Image Coordinates

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

Perspective Projection

$$\begin{bmatrix} f_x & 0 & o_x & 0 \\ 0 & f_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Intrinsics

Camera Coordinates

$$\mathbf{X}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$$

Coordinate Transformation

$$\begin{bmatrix} R_{3\times 3} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}$$

Extrinsics

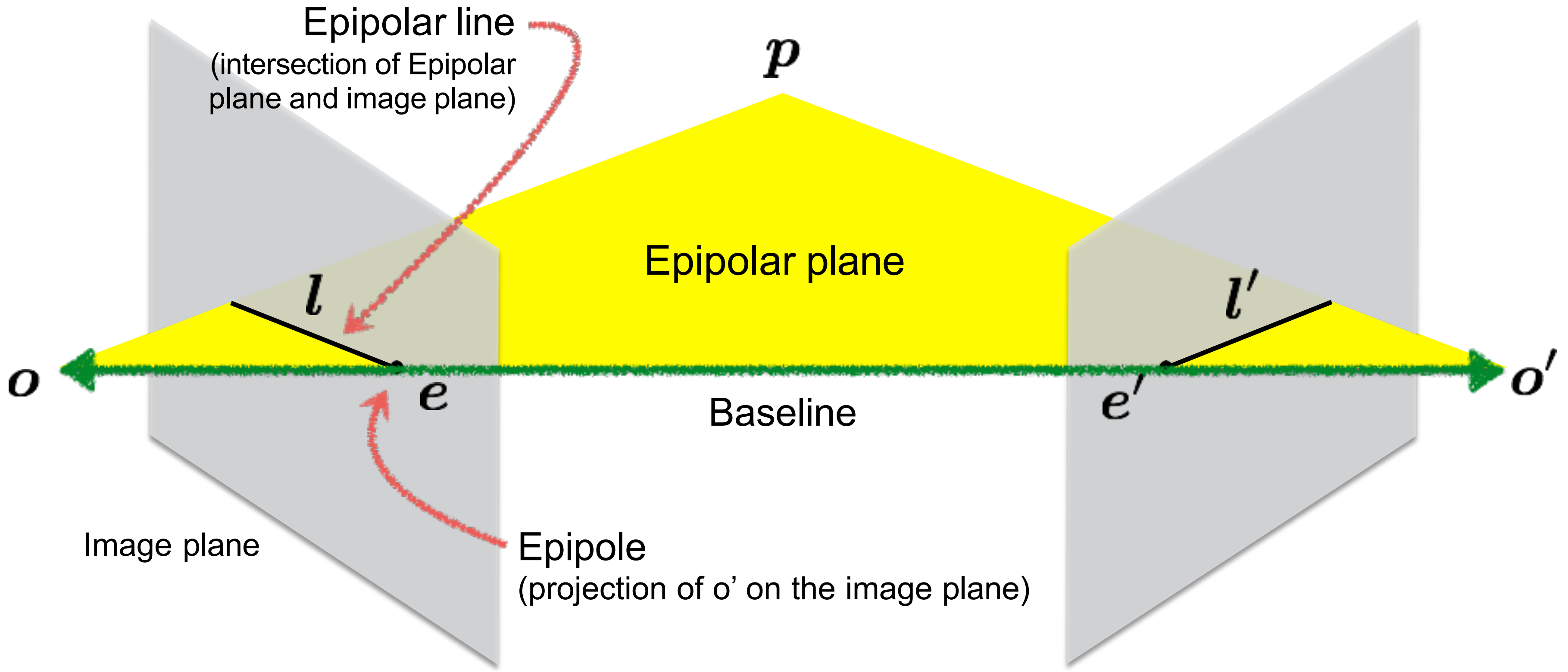
World Coordinates

$$\mathbf{X}_w = \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix}$$

Last lecture:
How to calibrate the camera?

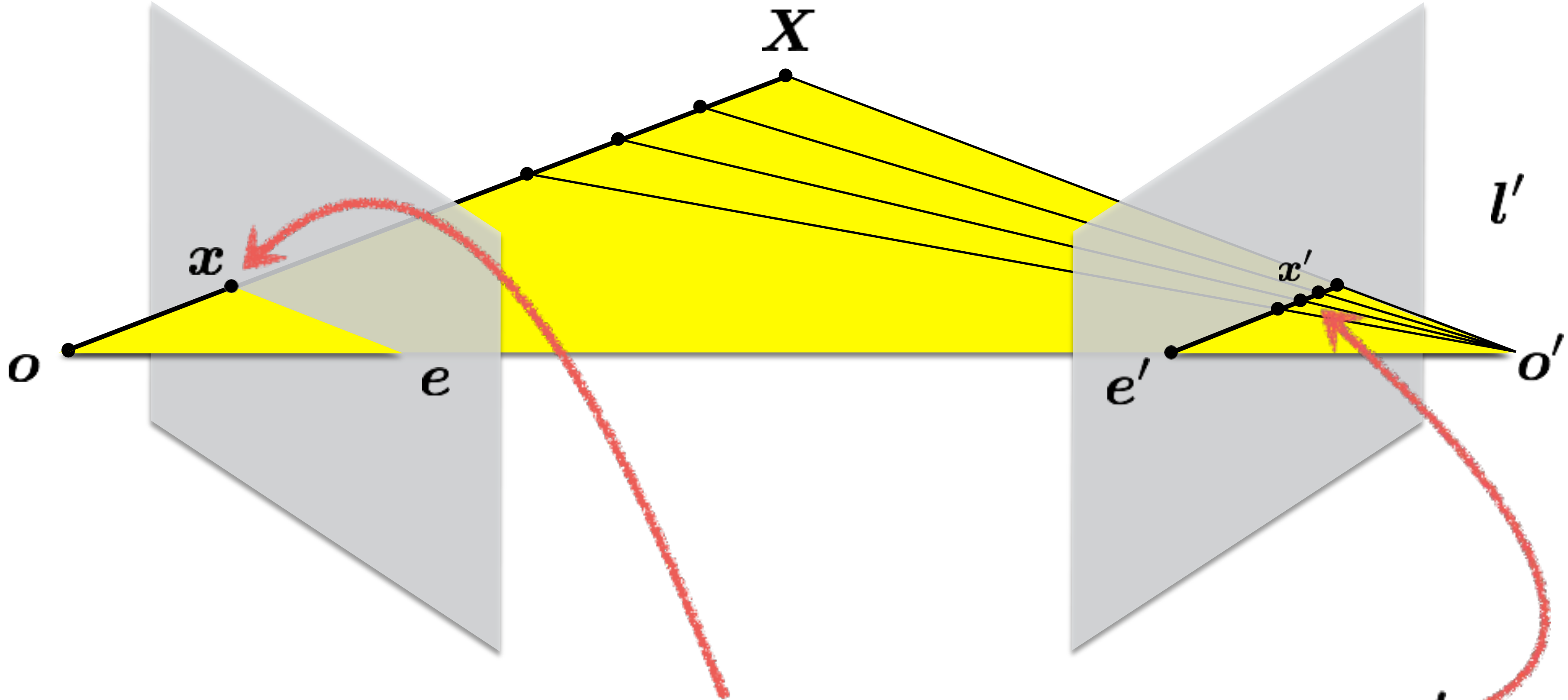
Slide inspired by Shree Nayar

Epipolar geometry



Epipolar constraint

$$\mathbf{E}\boldsymbol{x} = \boldsymbol{l}' \rightarrow \boldsymbol{x}'^{\top}\mathbf{E}\boldsymbol{x} = 0 \rightarrow \boxed{\mathbf{E} = \mathbf{R}\,[\mathbf{t}]_{\times}}$$



Potential matches for $\boldsymbol{x}$ lie on the epipolar line $\boldsymbol{l}'$

Fundamental Matrix

$$\mathbf{F} = \mathbf{K}'^{-\top}\mathbf{E}\mathbf{K}^{-1} \quad \mathbf{F} = \mathbf{K}'^{-\top}[\mathbf{t}_{\times}]\mathbf{R}\mathbf{K}^{-1}$$

- Essential Matrix operates on points in camera coordinate system (after projection from 3D to 2D)

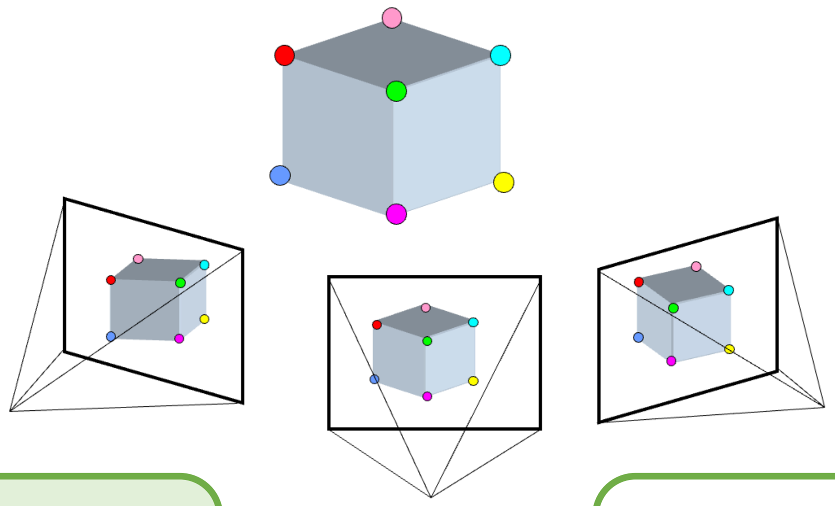
- Fundamental Matrix operates on points in pixel coordinate system

- E and F are both rank(2), but E has 2 singular values that are equal, but not F.

- E has 5 DoF and F has 7 DoF.

Big picture: 3 key components in 3D

3D Points
(Structure)



Estimate
Fundamental Matrix

Correspondences

Camera
(Motion)

How do we estimate fundamental matrix from pairs of corresponding points in two images?

(Normalized) Eight-Point Algorithm

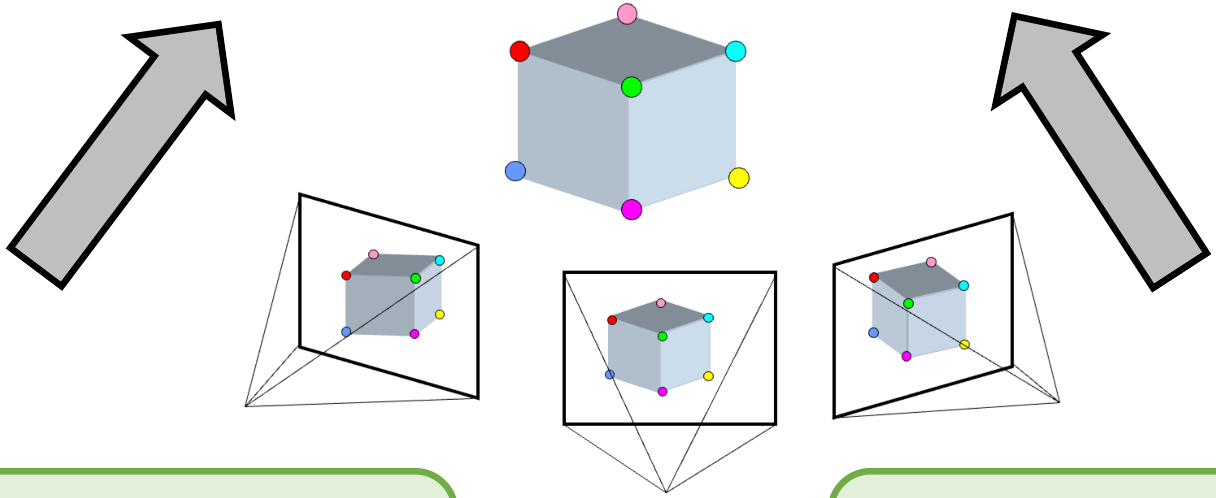
1. (Normalize points)
2. Construct the M x 9 matrix **A**
3. Find the SVD of **A**
4. Entries of **F** are the elements of column of **V** corresponding to the least singular value

4. (Enforce rank 2 constraint on F)
5. (Un-normalize F)

Big picture: 3 key components in 3D

3D Points
(Structure)

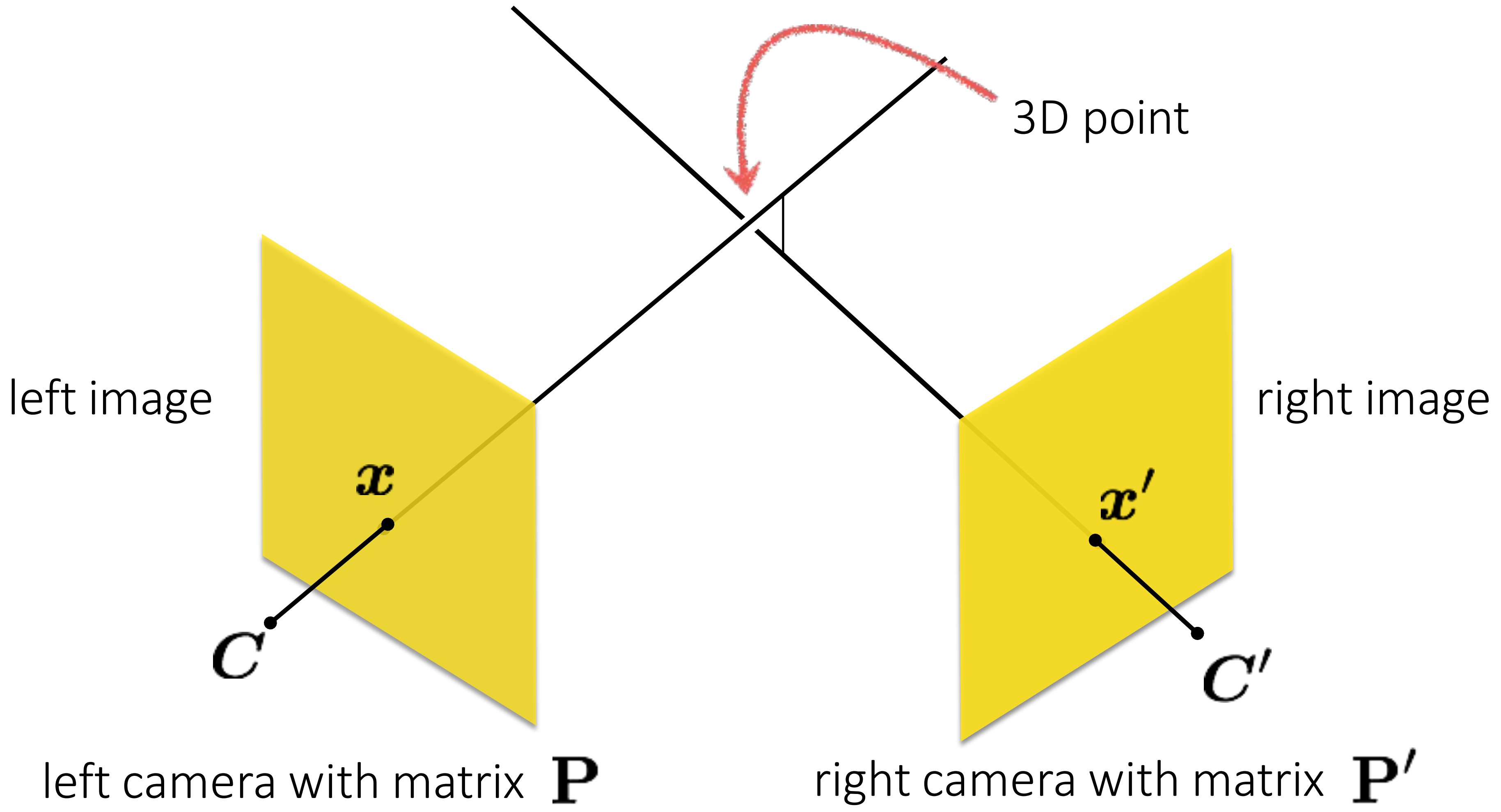
Triangulation



Correspondences

Camera
(Motion)

Triangulation



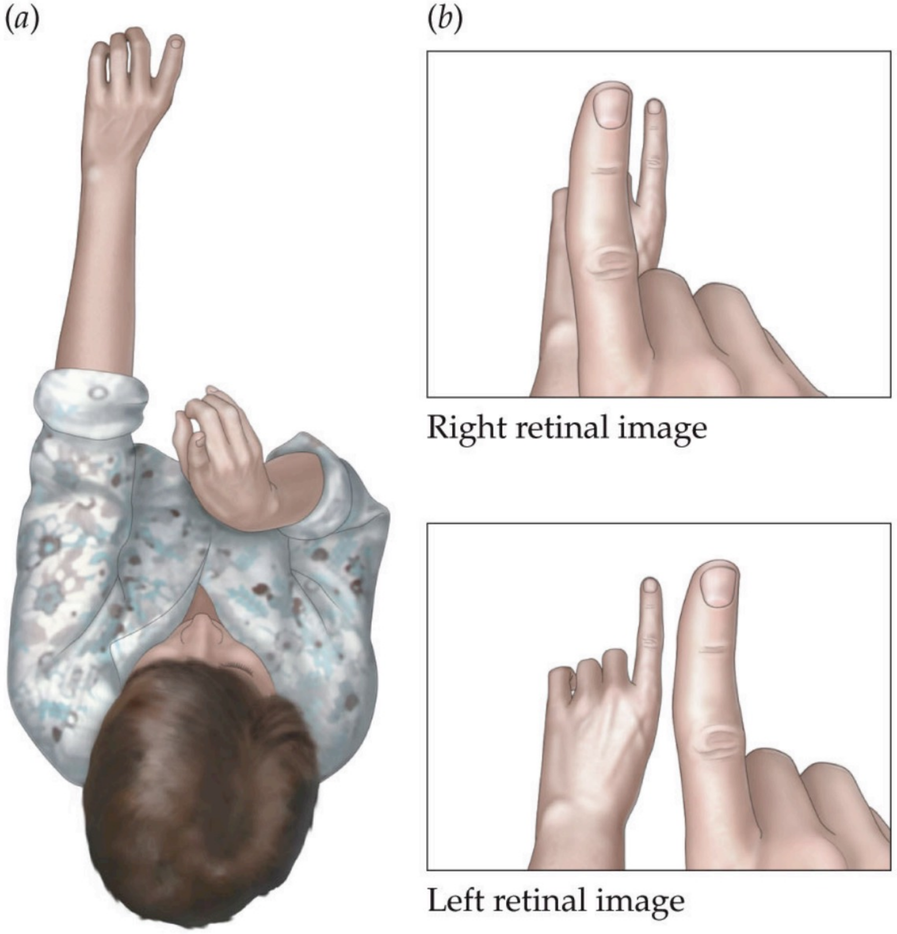
Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
- Active stereo with structured light

Today's lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
- Active stereo with structured light

We are equipped with binocular vision.
Stereo in humans!
Let’s try!

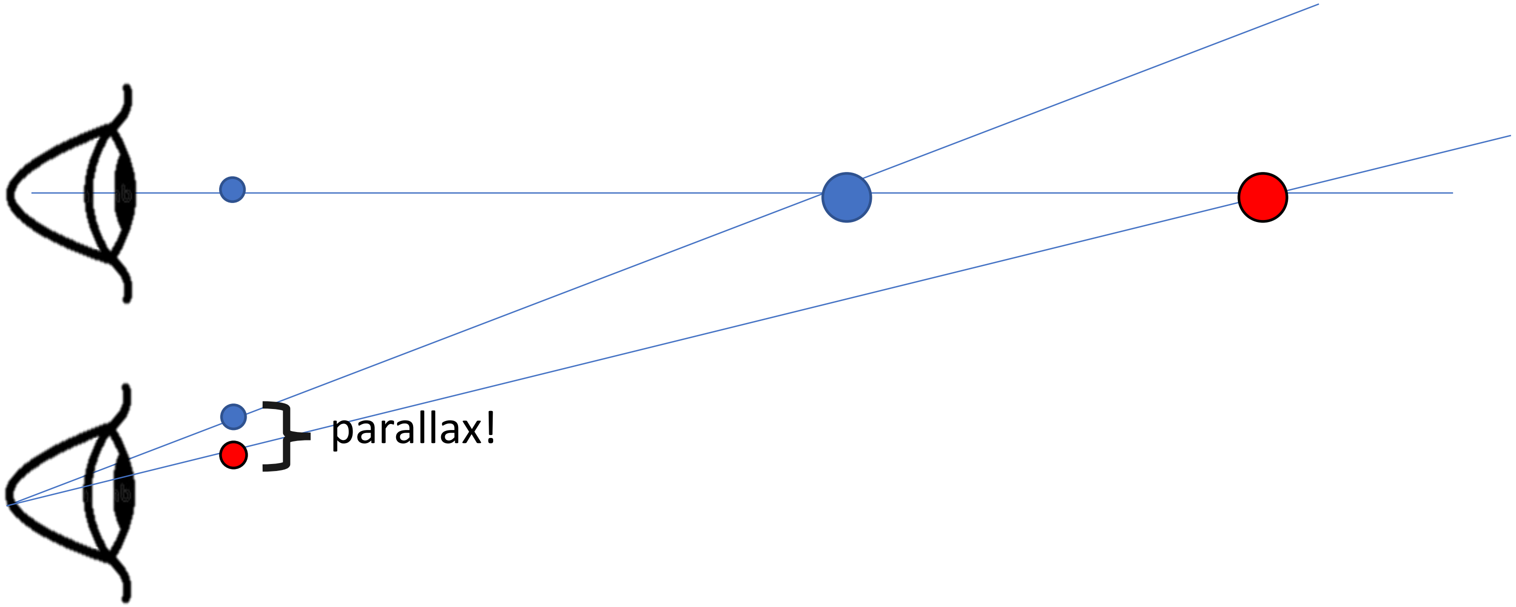


If you can’t close just one of your eyes on its own, just use a line far away in this room.

Use one hand to close your eyes and bring another hand in front of your eyes.

Relative displacement is higher as the relative distance grows
== Parallax

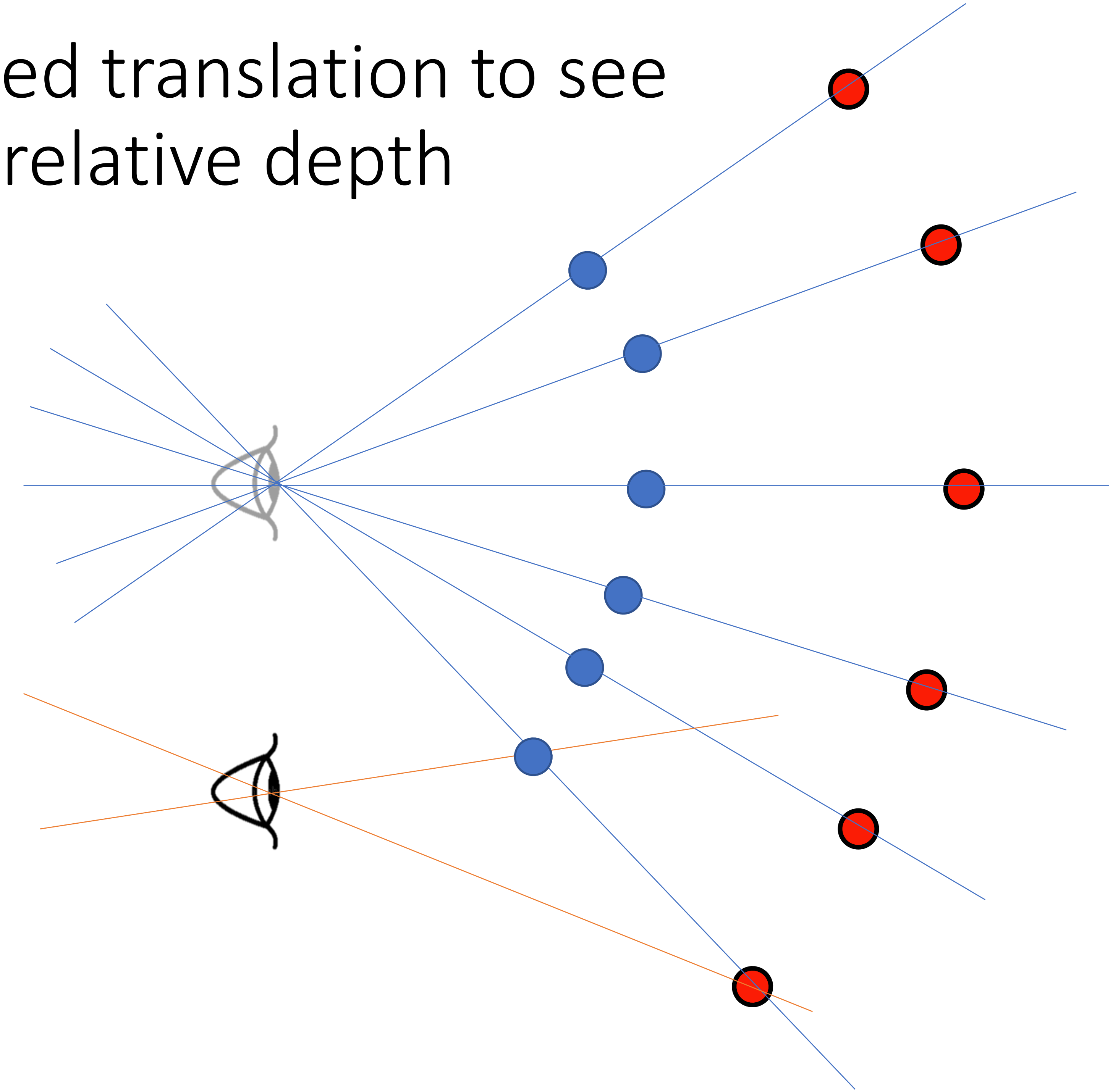
Parallax



Parallax = *from ancient Greek parállaxis*
= *Para* (side by side) + *allássō*, (to alter)
= *Change in position from different view point*

Two eyes give you parallax, you can also move to see more parallax = “Motion Parallax”

Why you need translation to see parallax i.e. relative depth



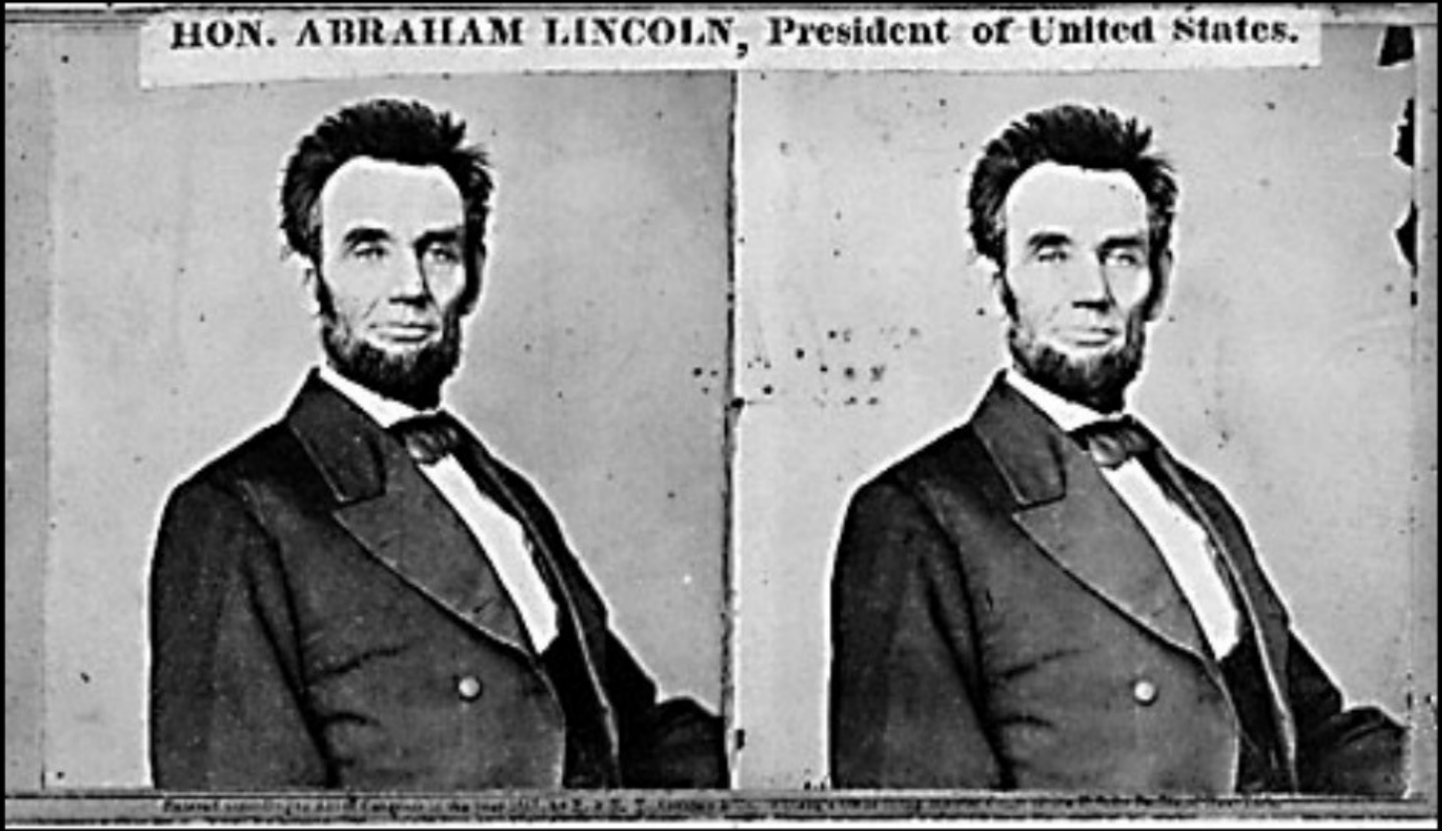
Stereo in 3D movies

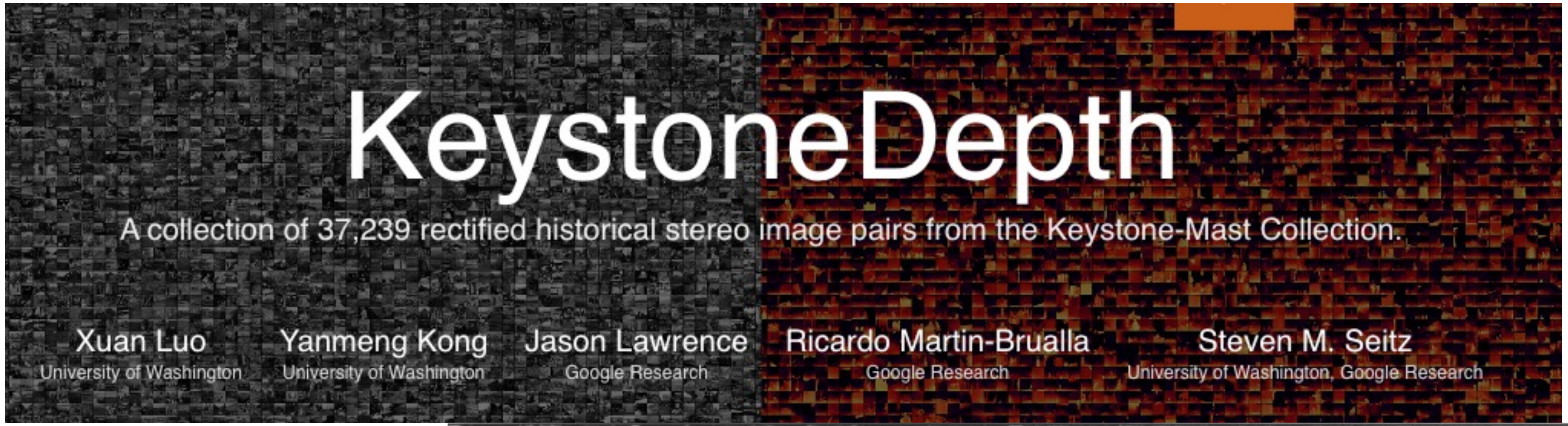
Stereo in the past

Stereograph

Credits: Xuan Luo

Stereoscopes: A 19th Century Pastime


HON. ABRAHAM LINCOLN, President of United States.



Climbing the Great Pyramid

Credits: Xuan Luo

Real-time stereo sensing

Nomad robot searches for meteorites in Antartica

http://www.cs.cmu.edu/~meteorite/

Subaru
Eyesight system

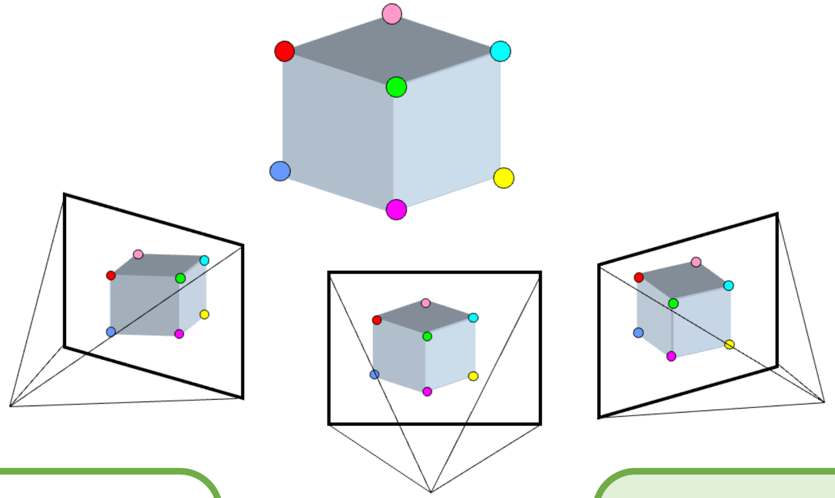
Pre-collision
braking

Today's lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
- Active stereo with structured light

Big picture: 3 key components in 3D

3D Points
(Structure)

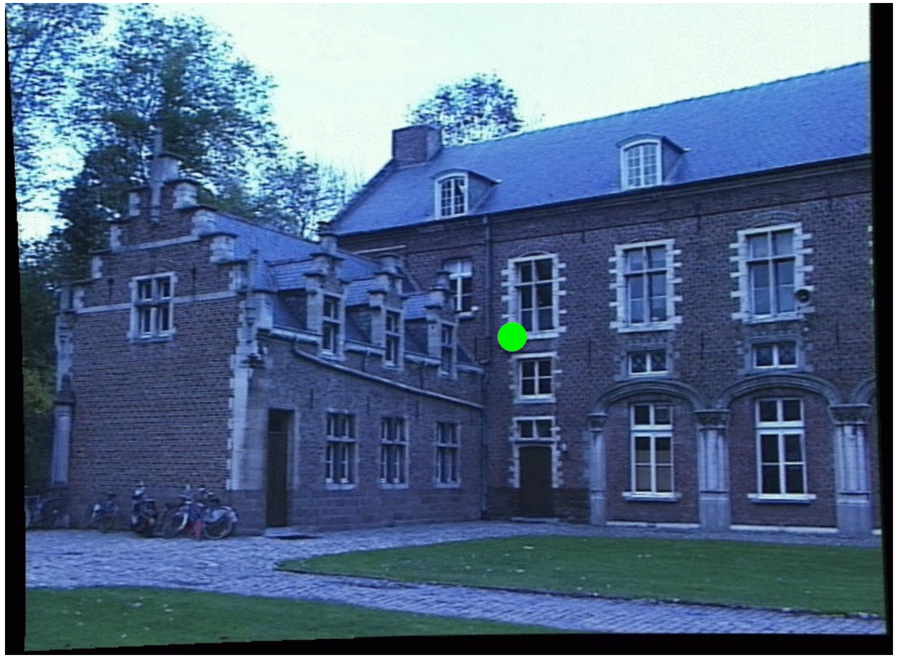


Correspondences

Camera
(Motion)

Stereo Matching

How would you reconstruct 3D points?



Left image

Right image

1. Select point in one image (how?)
2. Form epipolar line for that point in second image (how?)
3. Find matching point along line (how?)
4. Perform triangulation

Feature detection: SIFT

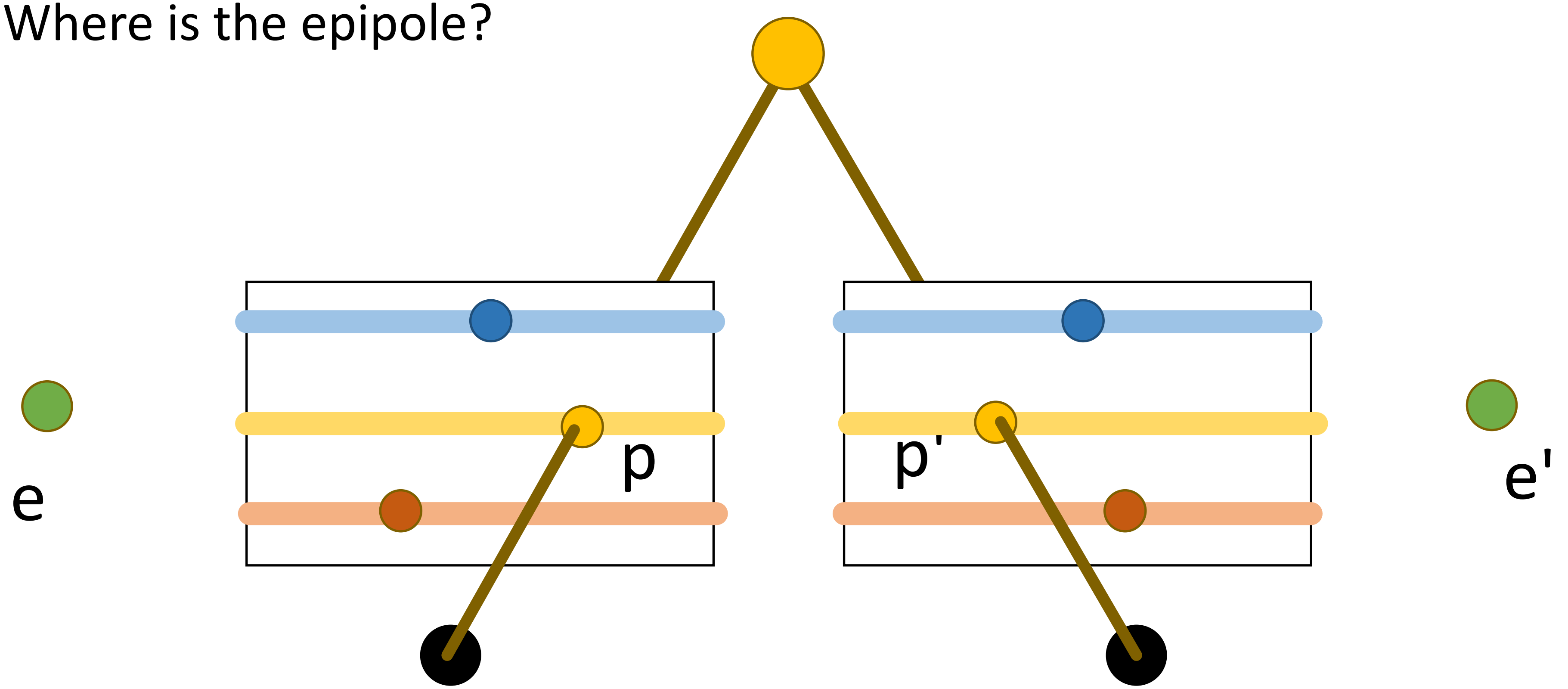
Calibrate cameras, find E or F

Stereo Matching (today!!)

What are the disadvantages of this procedure?

Example: Parallel to Image Plane

Where is the epipole?

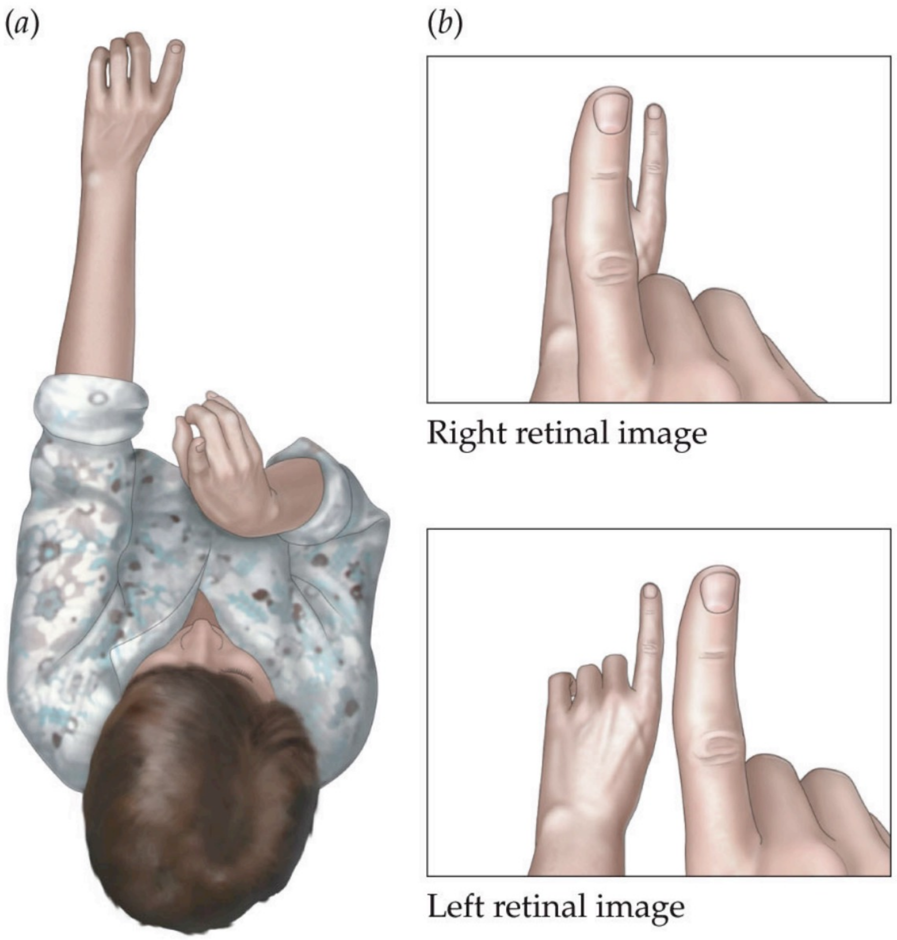


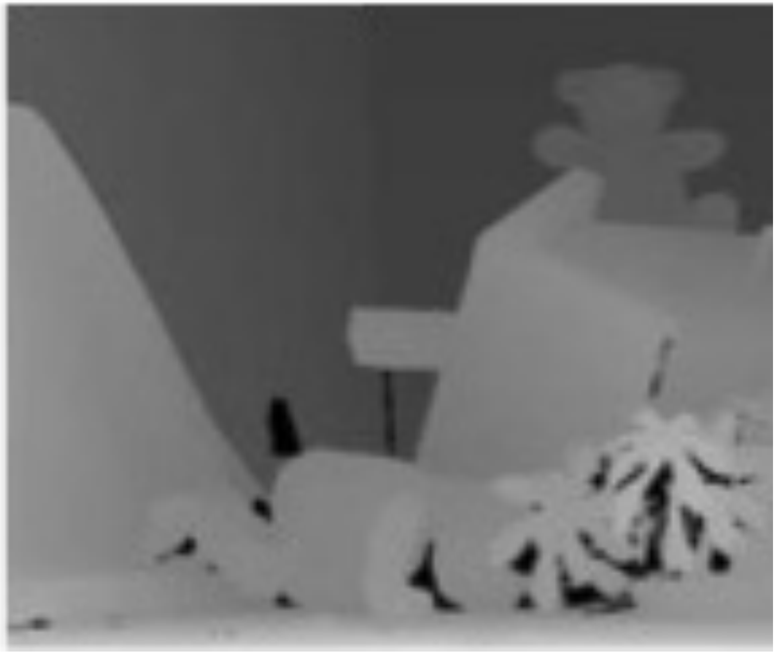
Epipoles *infinitely* far away, epipolar lines parallel

Slide credit: David Fouhey

Let’s try it again

Objects that are close move more or less?

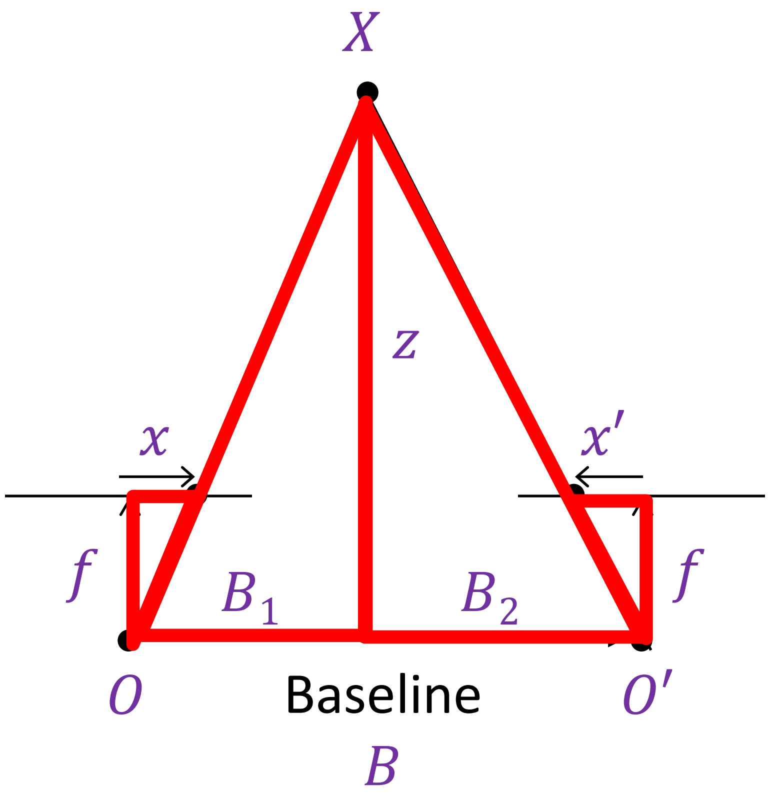




Depth map

More formally… The amount of horizontal movement is inversely proportional to the distance from the camera, i.e. depth map.

Depth from disparity



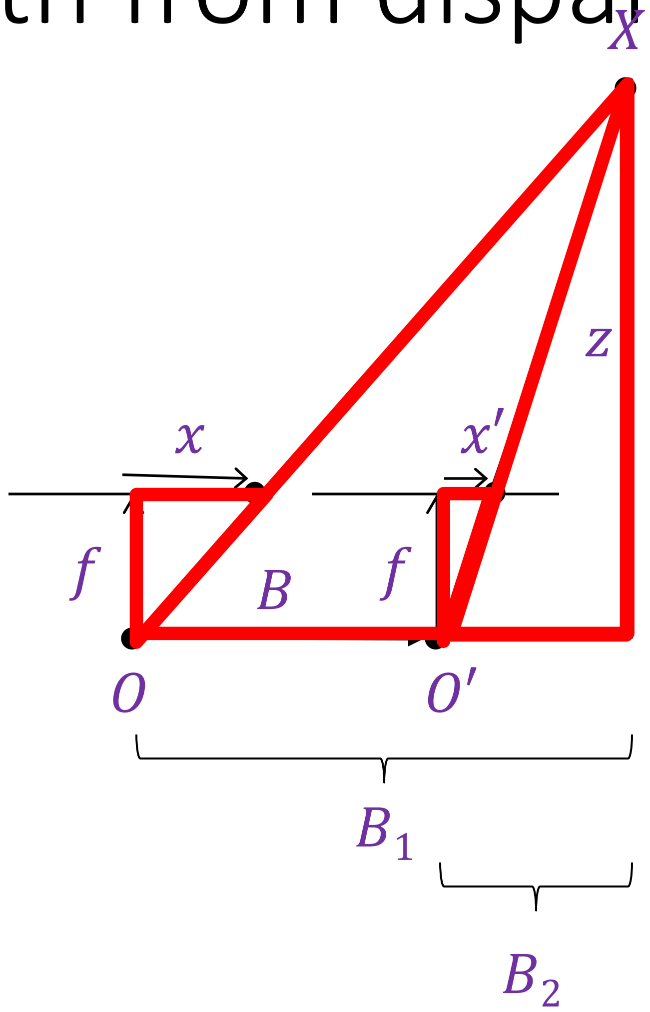
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{-x'}{f} = \frac{B_2}{z}$$

$$\frac{x - x'}{f} = \frac{B_1 + B_2}{z}$$

$$\underbrace{x - x'} = \frac{fB}{z}$$

Disparity is inversely proportional to depth!

Depth from disparity



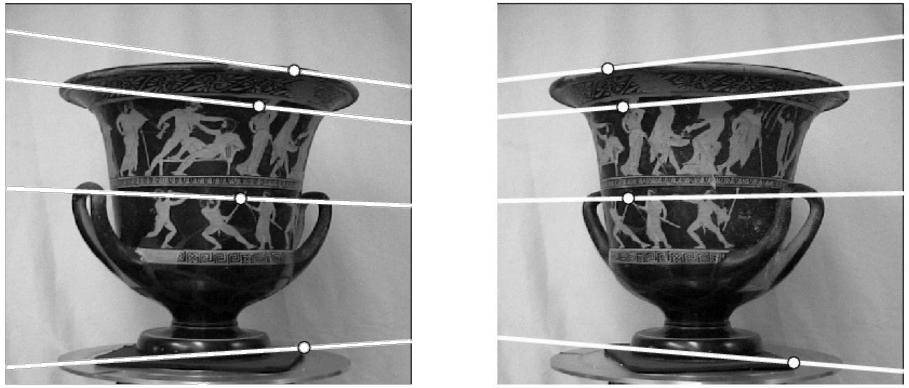
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{x'}{f} = \frac{B_2}{z}$$

$$\frac{x - x'}{f} = \frac{B_1 - B_2}{z}$$

$$x - x' = \frac{fB}{z}$$

$$z = \frac{fB}{x - x'}$$

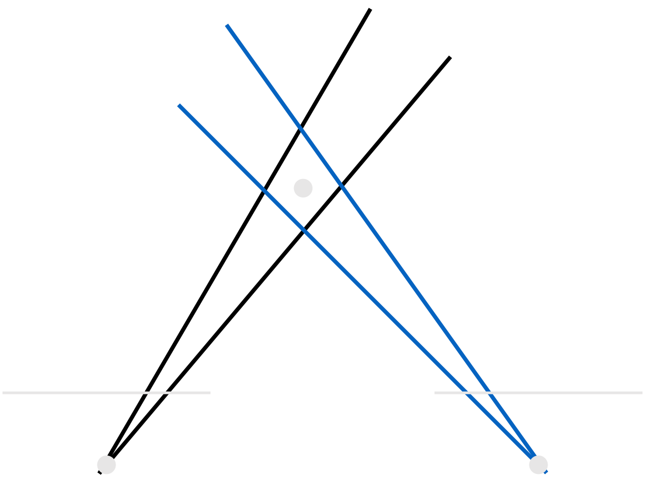
So can I compute depth from any two images of the same object?

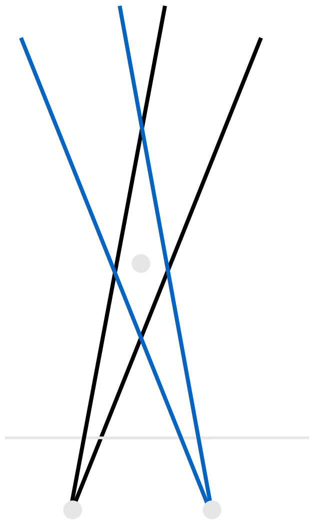


1. Need sufficient baseline

2. Images need to be ‘rectified’ first (make epipolar lines horizontal)

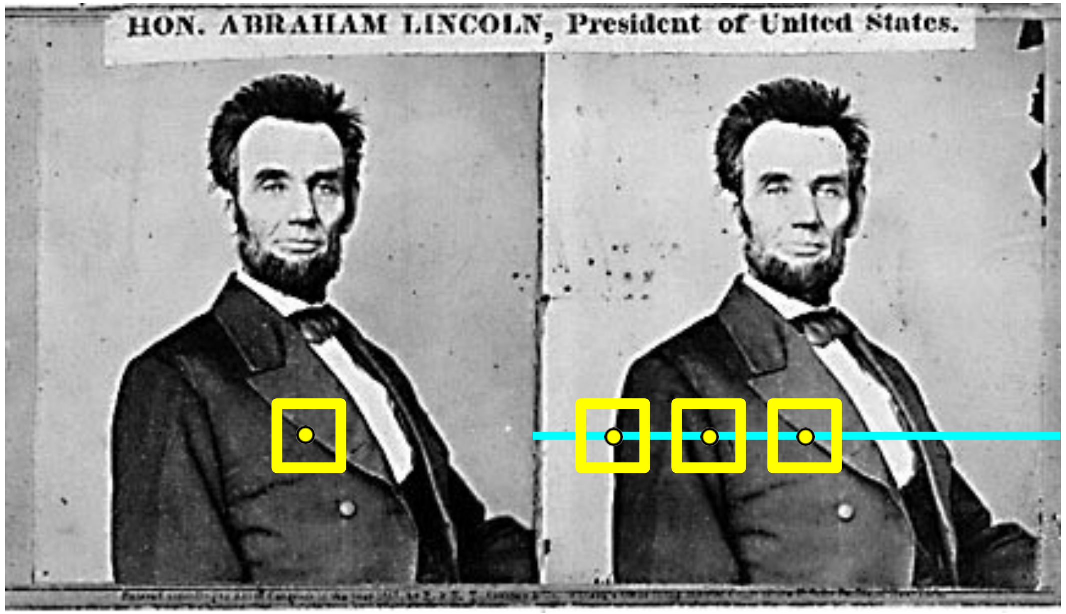
Effect of baseline on stereo results





- Larger baseline
  + Smaller triangulation error
  – Matching is more difficult

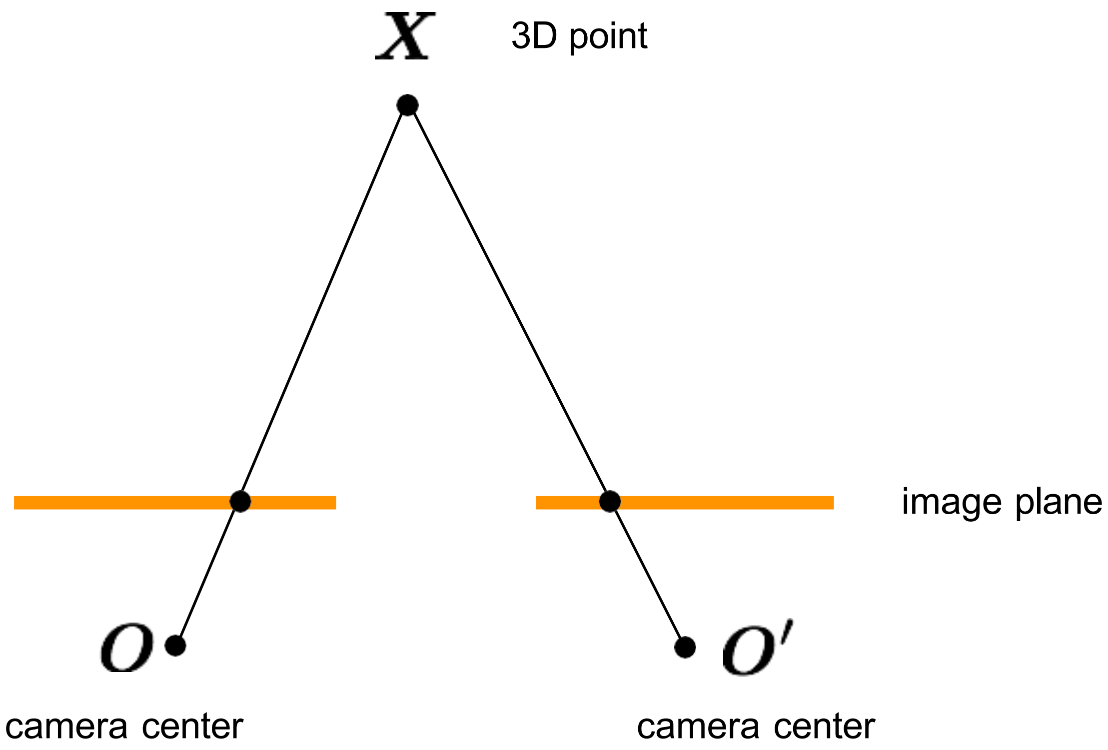
- Smaller baseline
  – Higher triangulation error
  + Matching is easier

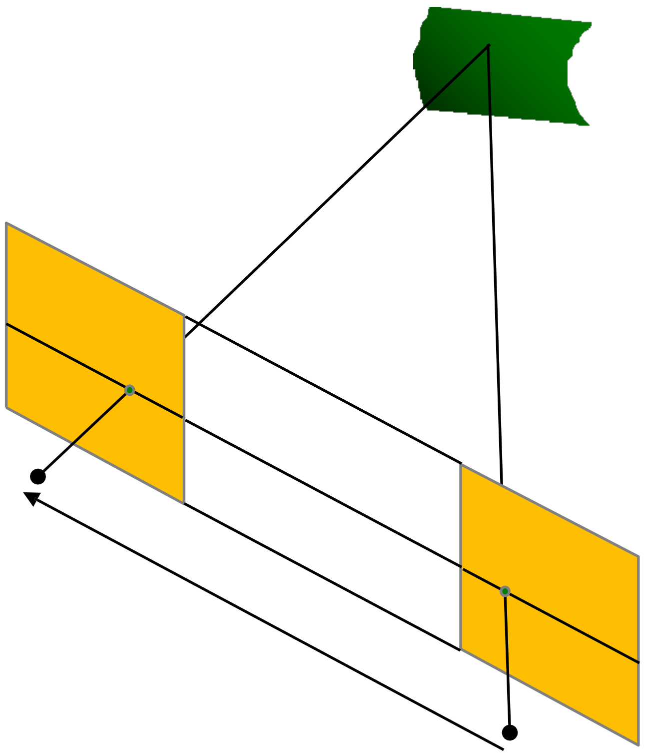


1. Rectify images
   (make epipolar lines horizontal)
2. For each pixel
   a. Find epipolar line
   b. Scan line for best match
   c. Compute depth from disparity

$$Z = \frac{bf}{d}$$

How can you make the epipolar lines horizontal?





When are epipolar lines horizontal?

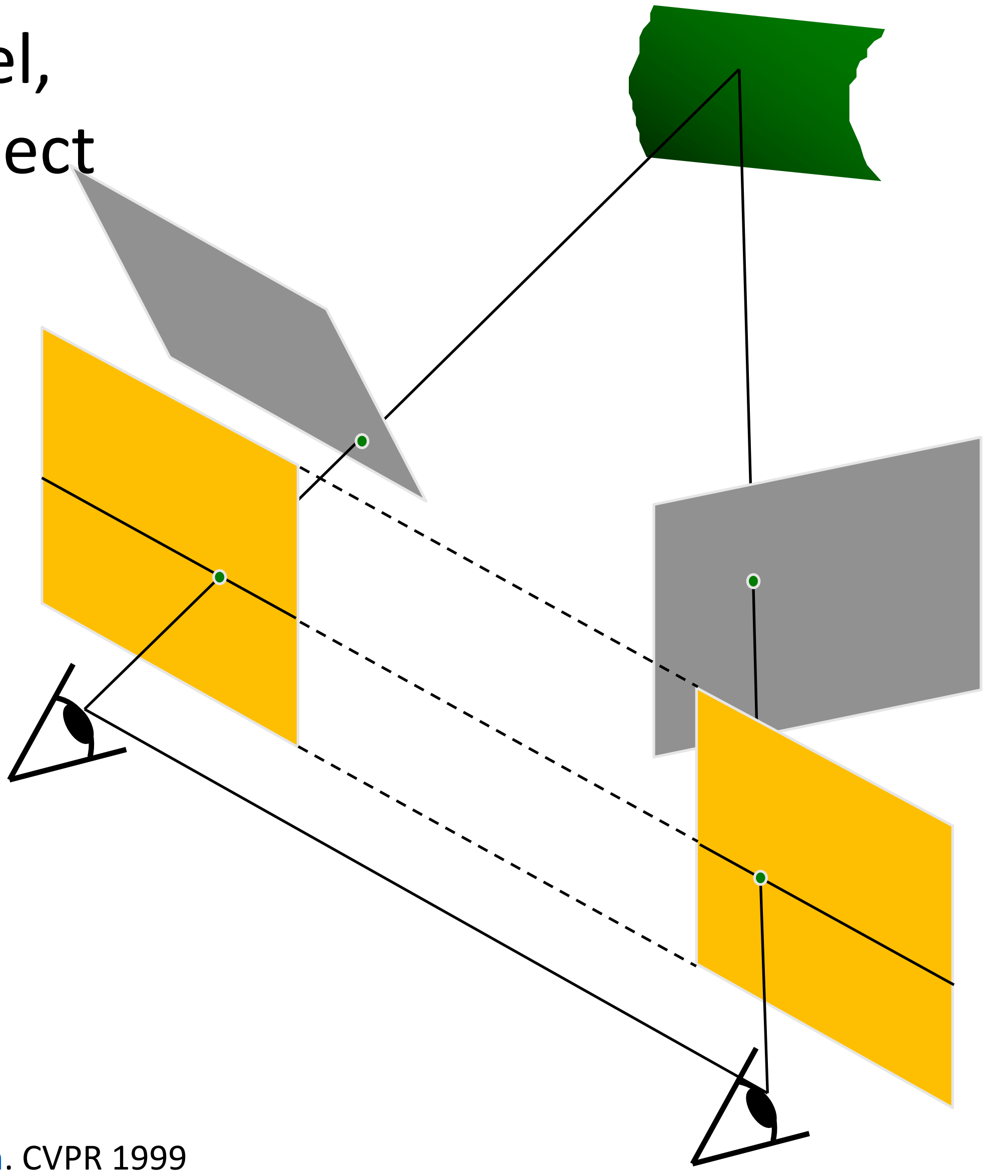
What's special about these two cameras?

When this relationship holds:

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

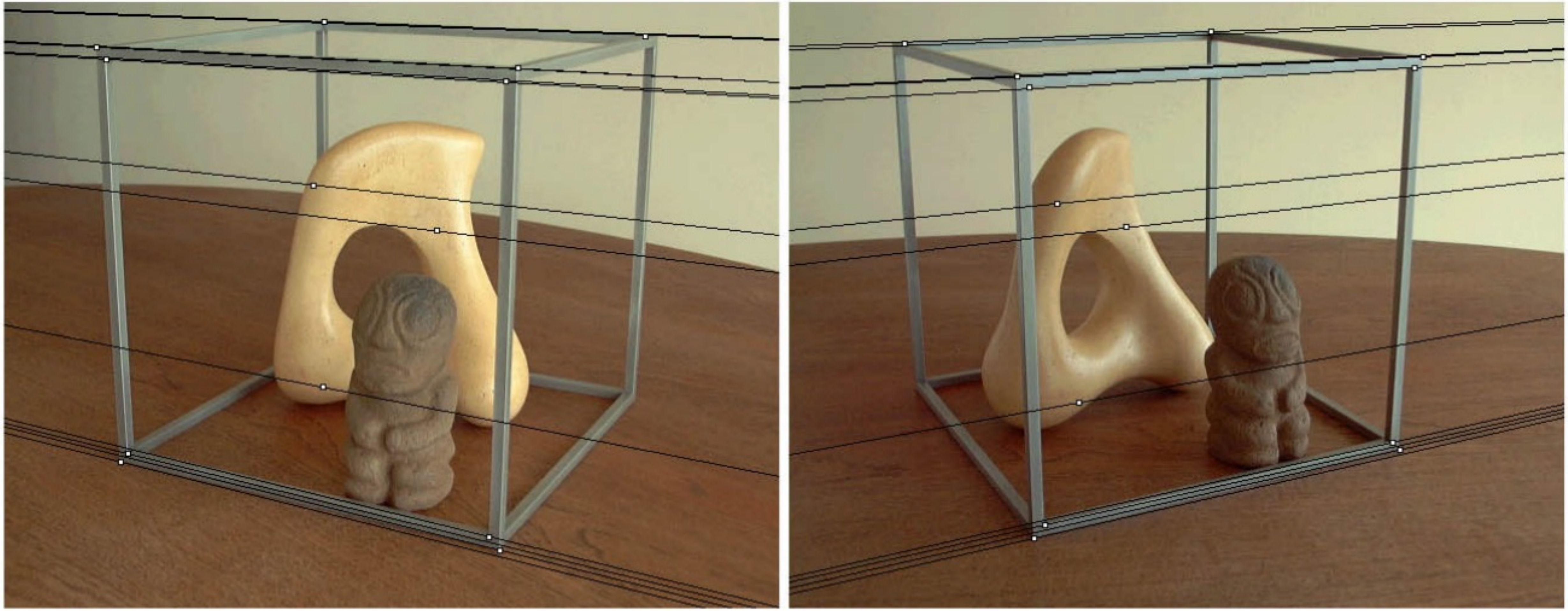
Stereo image rectification

- If the image planes are not parallel, we can find homographies to project each view onto a common plane parallel to the baseline



•C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). CVPR 1999

Stereo image rectification



[Image source](#)

Stereo image rectification

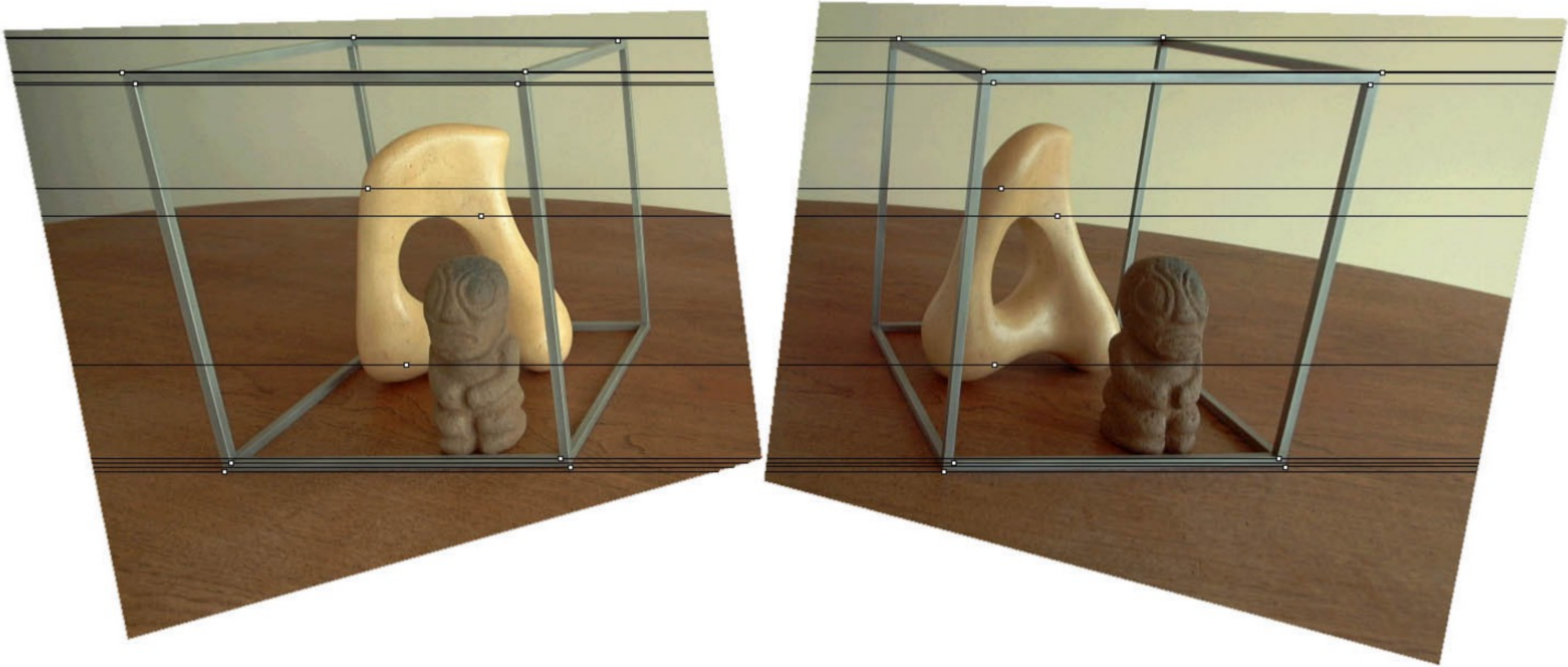
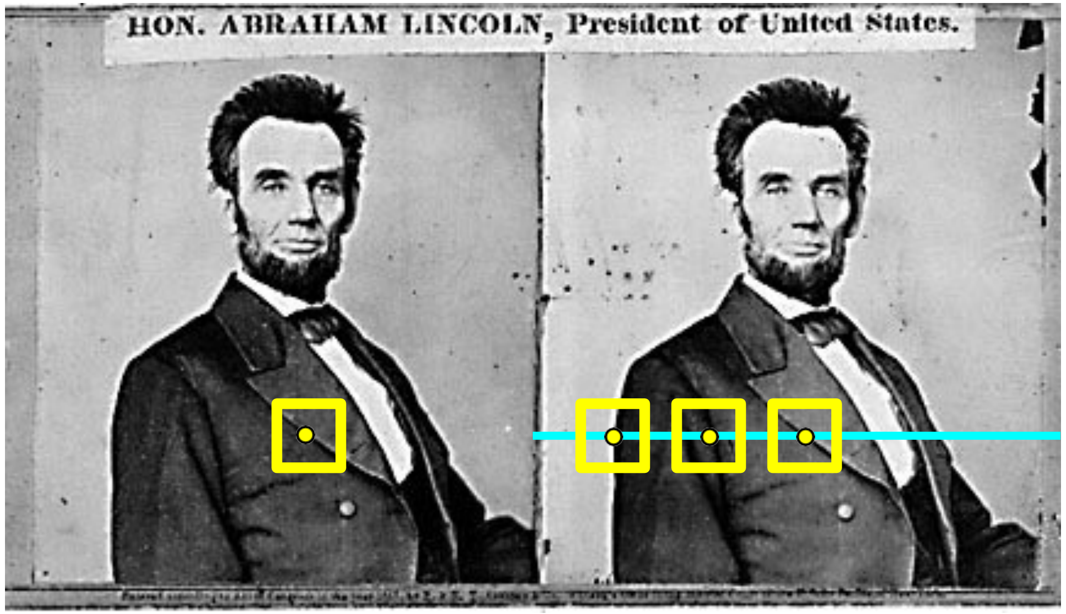


Image source


Depth Estimation via Stereo Matching



1. Rectify images
   (make epipolar lines horizontal)
2. For each pixel
   a. Find epipolar line
   b. Scan line for best match
   c. Compute depth from disparity

$$Z = \frac{bf}{d}$$

How would you do this?

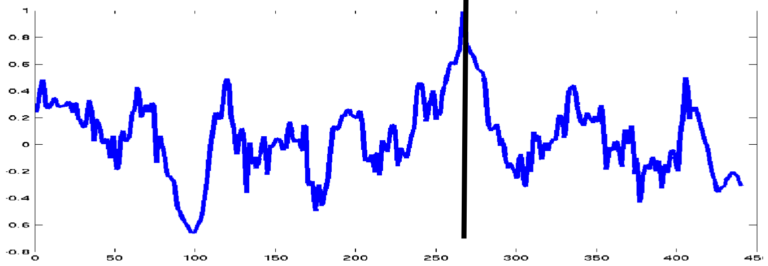
Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
- Active stereo with structured light

Matching using Epipolar Lines

Left Image

Right Image



For a patch in left image

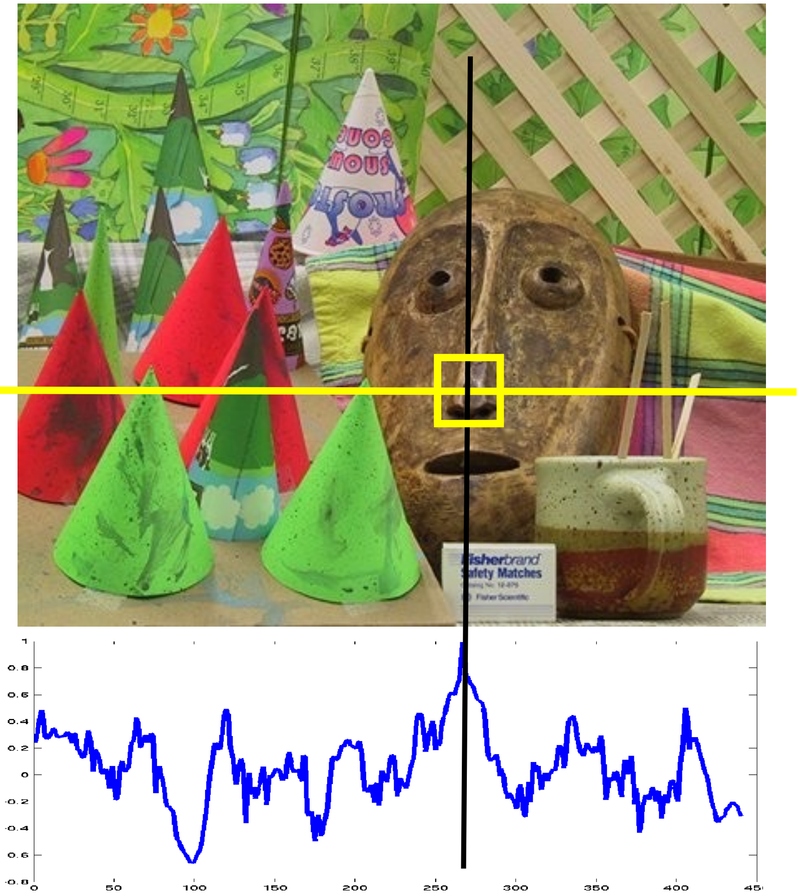
Compare with patches along same row in right image

Match Score Values (Similarity Measures)

Matching using Epipolar Lines

Left Image

Right Image

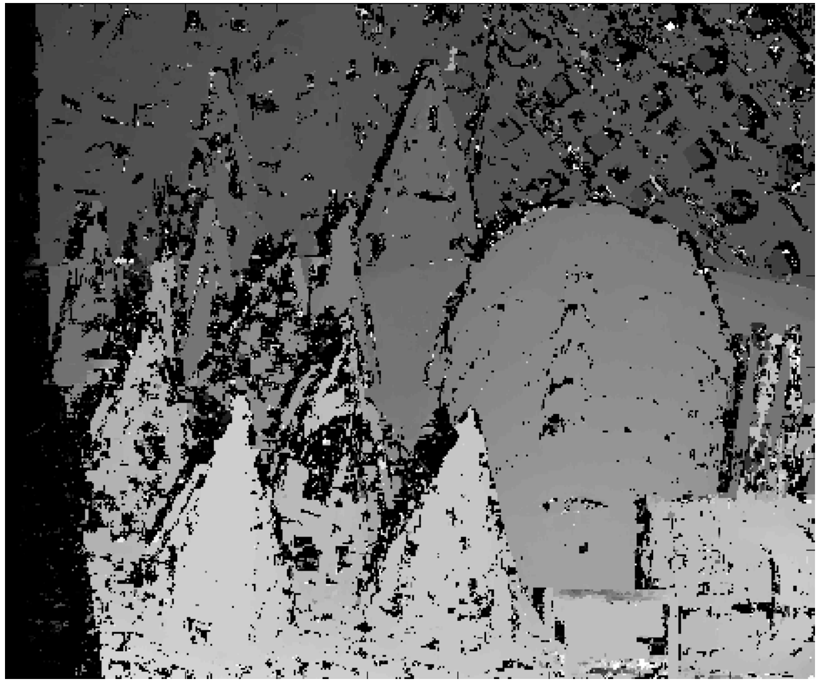


Select patch with highest match score.

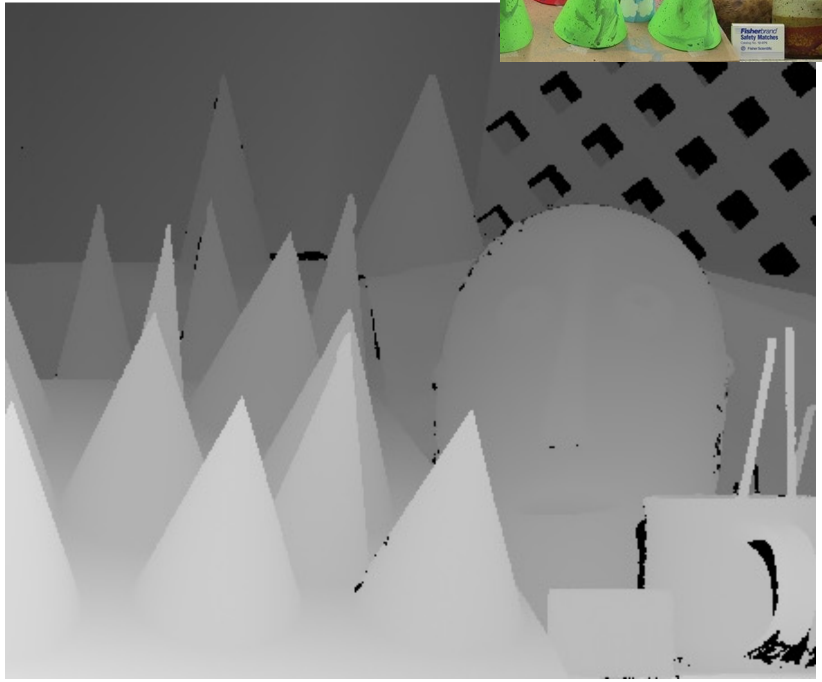
Repeat for all pixels in left image.

Match Score Values (Similarity Measures)

Example: 5x5 windows
NCC match score



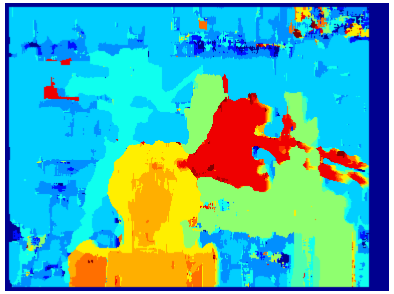
Computed disparities

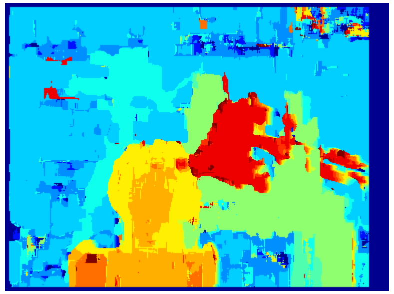


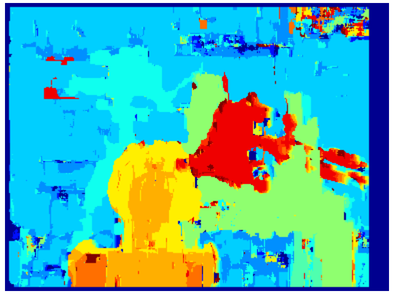
Ground truth

Black pixels: bad disparity values,
or no matching patch in right image

Similarity Measure	Formula
Sum of Absolute Differences (SAD)	$\sum_{(i,j)\in W} \lvert I_1(i,j) - I_2(x+i,y+j) \rvert$
Sum of Squared Differences (SSD)	$\sum_{(i,j)\in W} \left(I_1(i,j) - I_2(x+i,y+j)\right)^2$
Zero-mean SAD	$\sum_{(i,j)\in W} \lvert I_1(i,j) - \bar{I}_1(i,j) - I_2(x+i,y+j) + \bar{I}_2(x+i,y+j) \rvert$
Locally scaled SAD	$\sum_{(i,j)\in W} \lvert I_1(i,j) - \frac{\bar{I}_1(i,j)}{\bar{I}_2(x+i,y+j)} I_2(x+i,y+j) \rvert$
Normalized Cross Correlation (NCC)	$\frac{\sum_{(i,j)\in W} I_1(i,j).I_2(x+i,y+j)}{\sqrt[2]{\sum_{(i,j)\in W} I_1^2(i,j).\sum_{(i,j)\in W} I_2^2(x+i,y+j)}}$

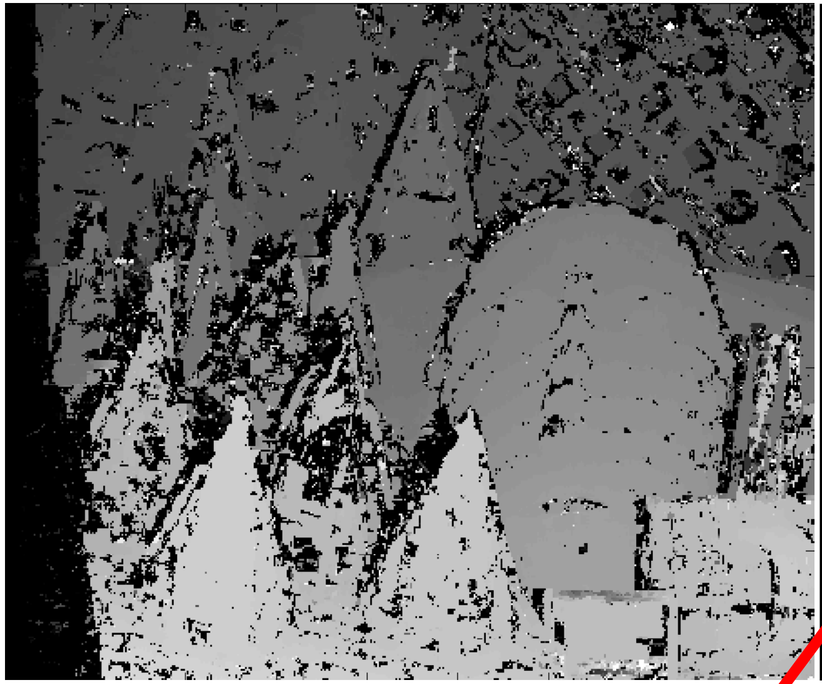

SAD


SSD

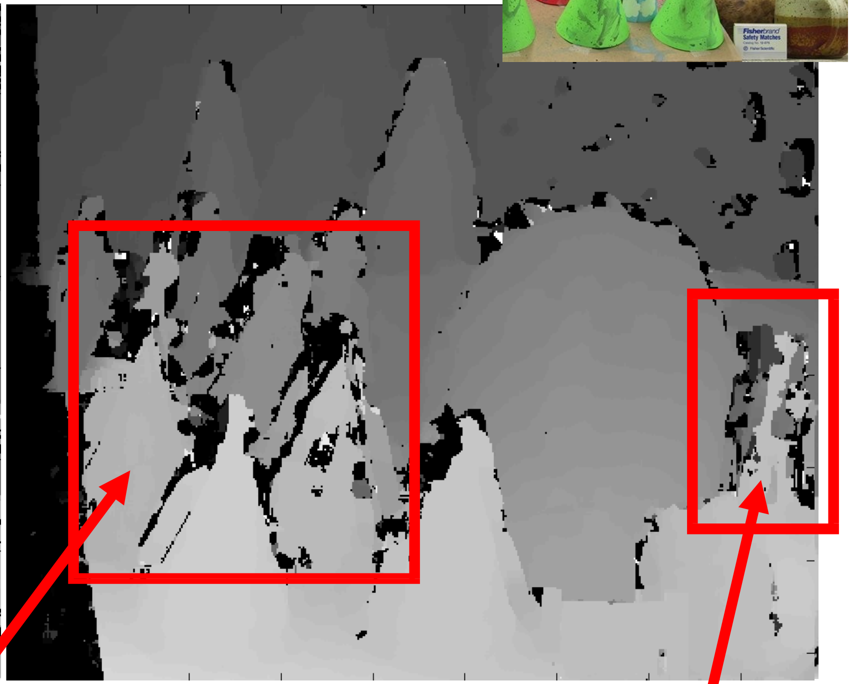

NCC


Ground truth

Effects of Patch Size



5x5 patches



11x11patches

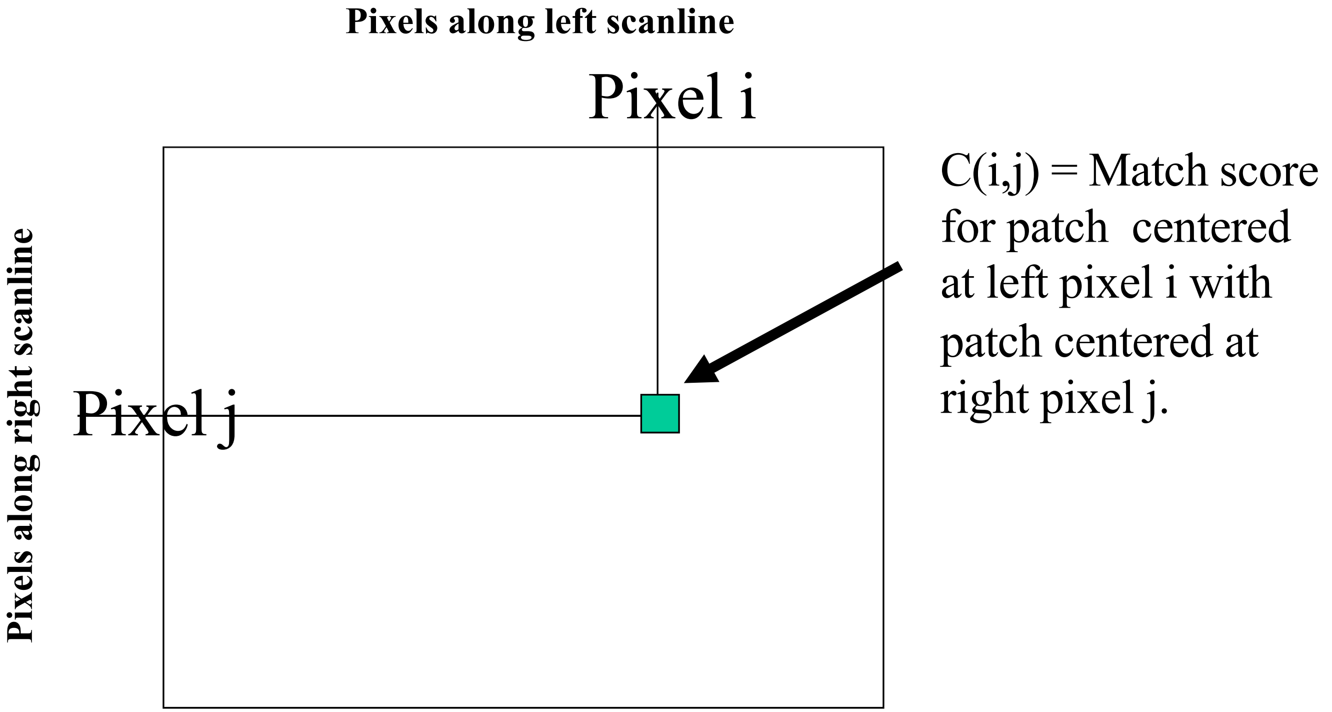
Smoother in some areas

Loss of finer details

Disparity Space Image

First we introduce the concept of DSI.

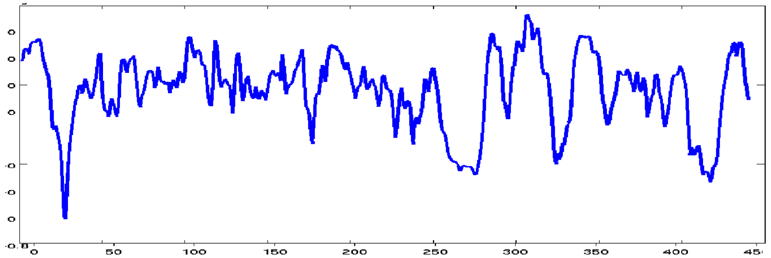
The DSI for one row represents pairwise match scores between patches along that row in the left and right image.



Disparity Space Image (DSI)

Left Image

Right Image



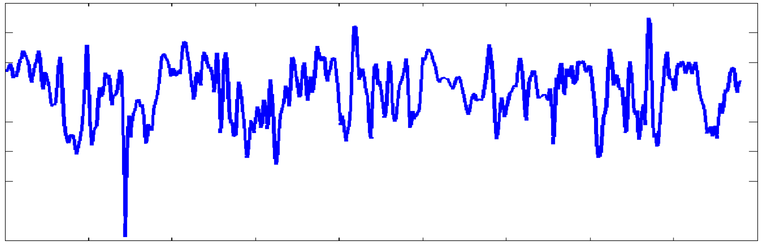
Dissimilarity Values

(1-NCC) or SSD

Disparity Space Image (DSI)

Left Image

Right Image



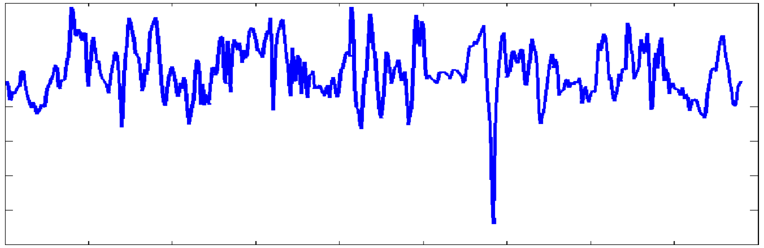
Dissimilarity Values

(1-NCC) or SSD

Disparity Space Image (DSI)

Left Image

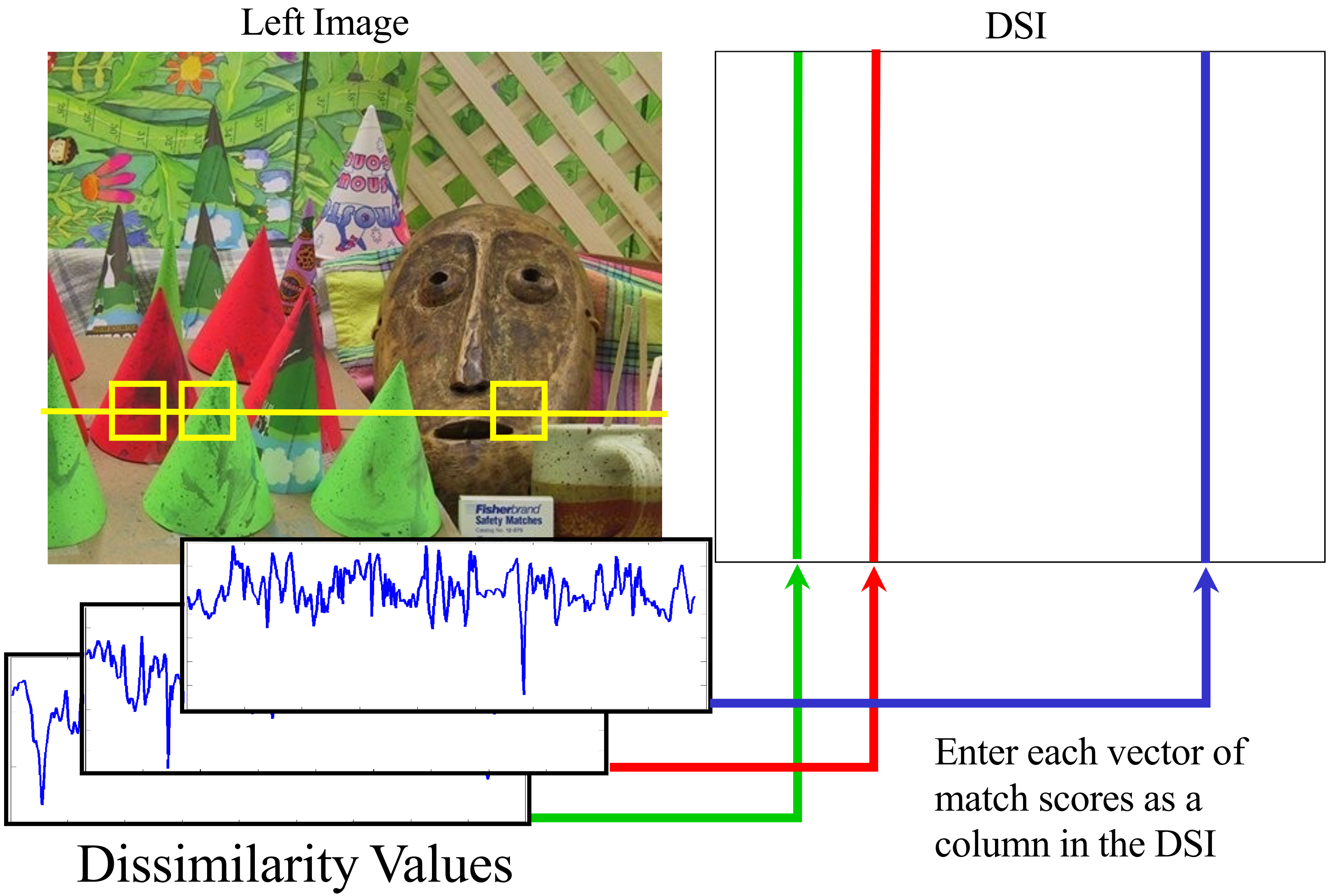
Right Image



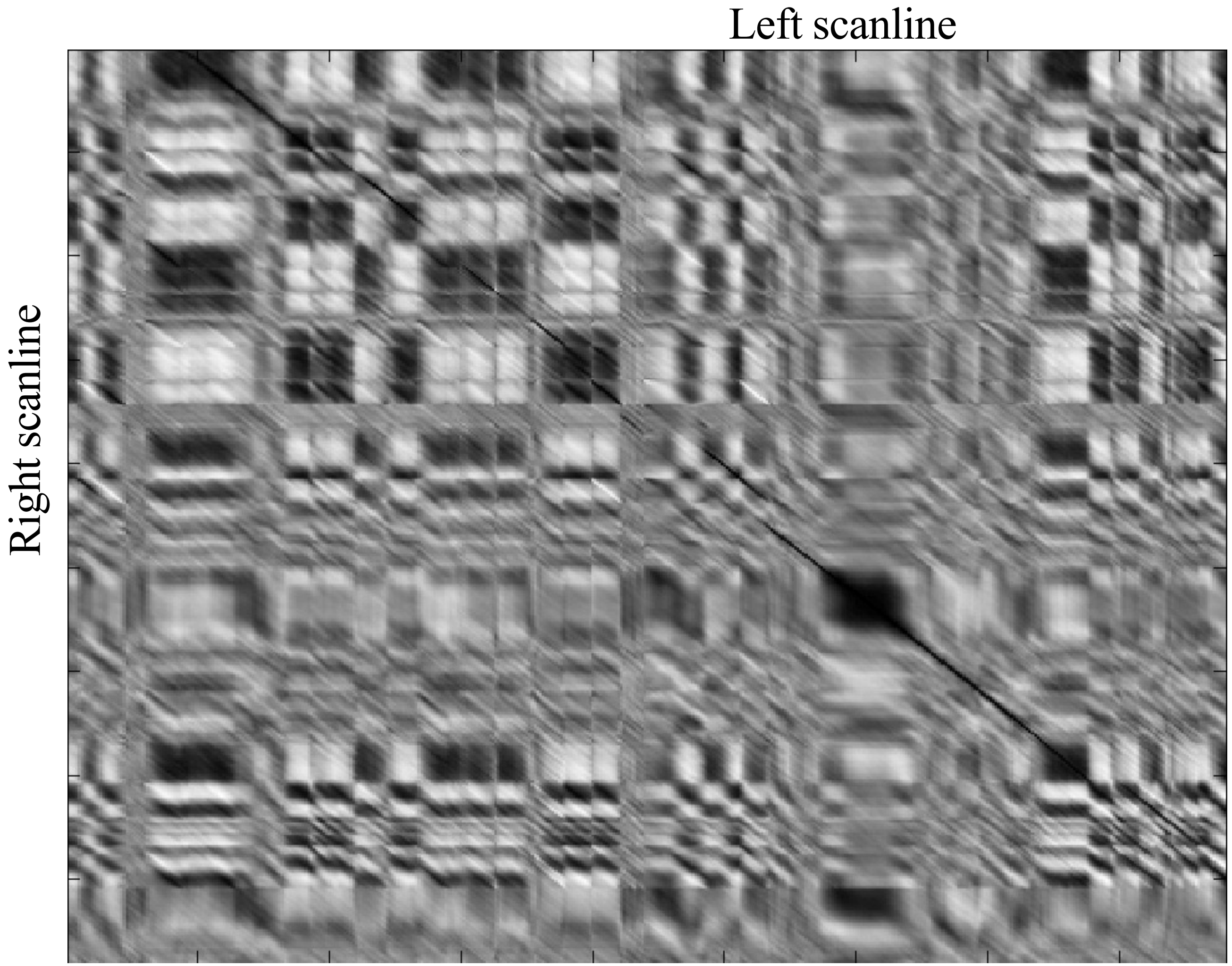
Dissimilarity Values

(1-NCC) or SSD

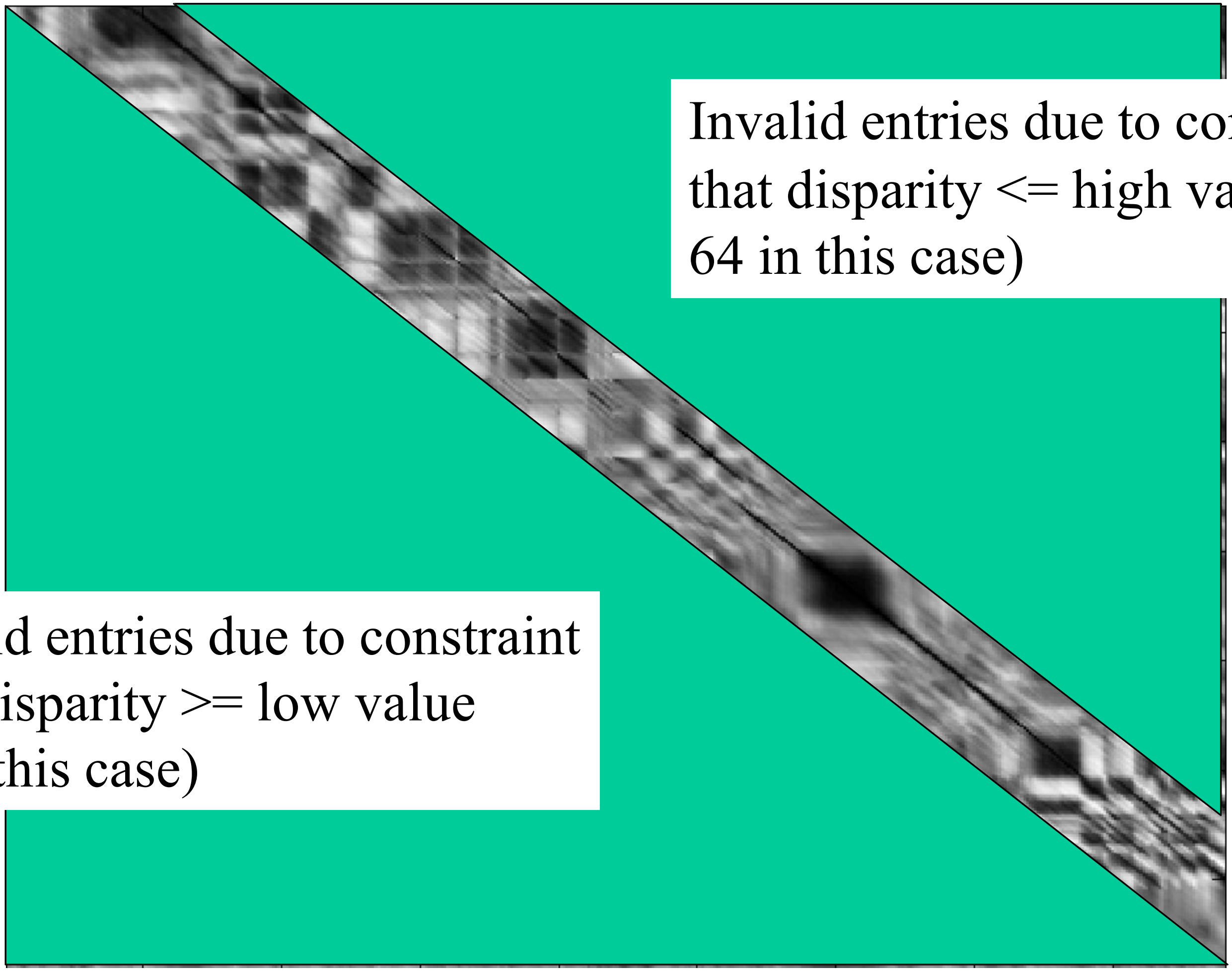
Disparity Space Image (DSI)



Disparity Space Image

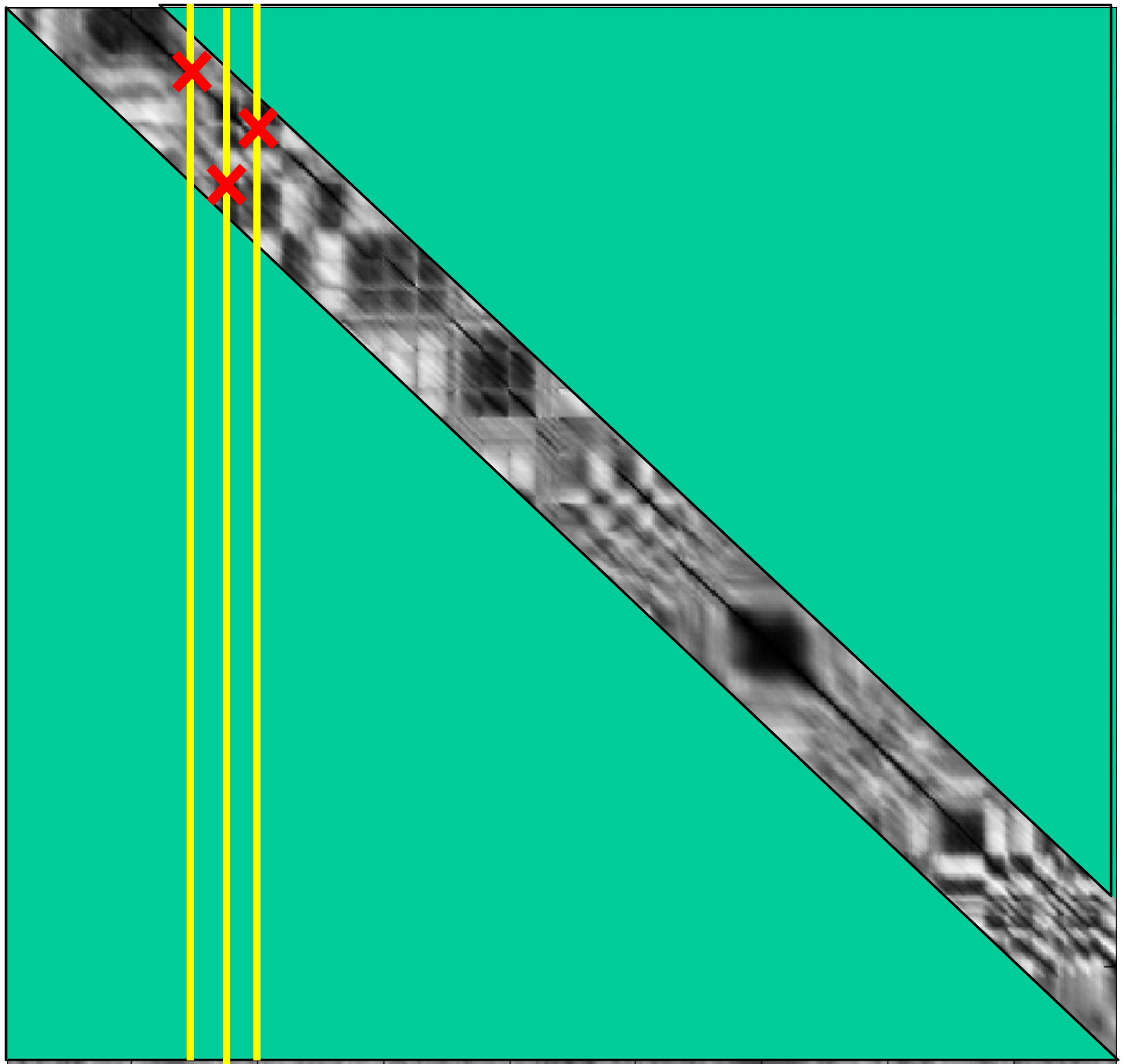


Disparity Space Image

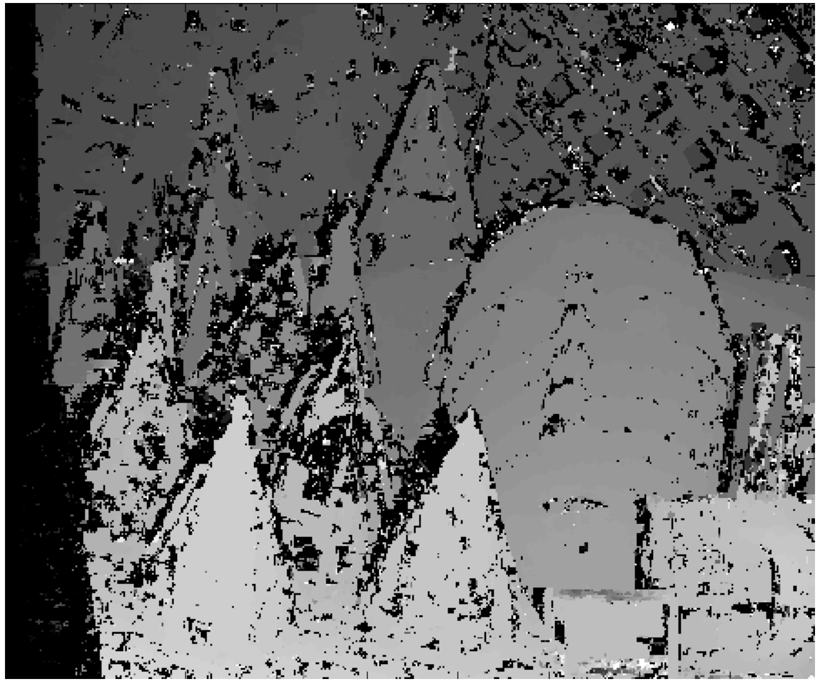


Greedy Selection: Simply choose the row with least disparity for each column

Greedy Per-pixel Path matching



Greedy selection often do not satisfy order constraints and produces non-smooth disparity map.



Today’s lecture

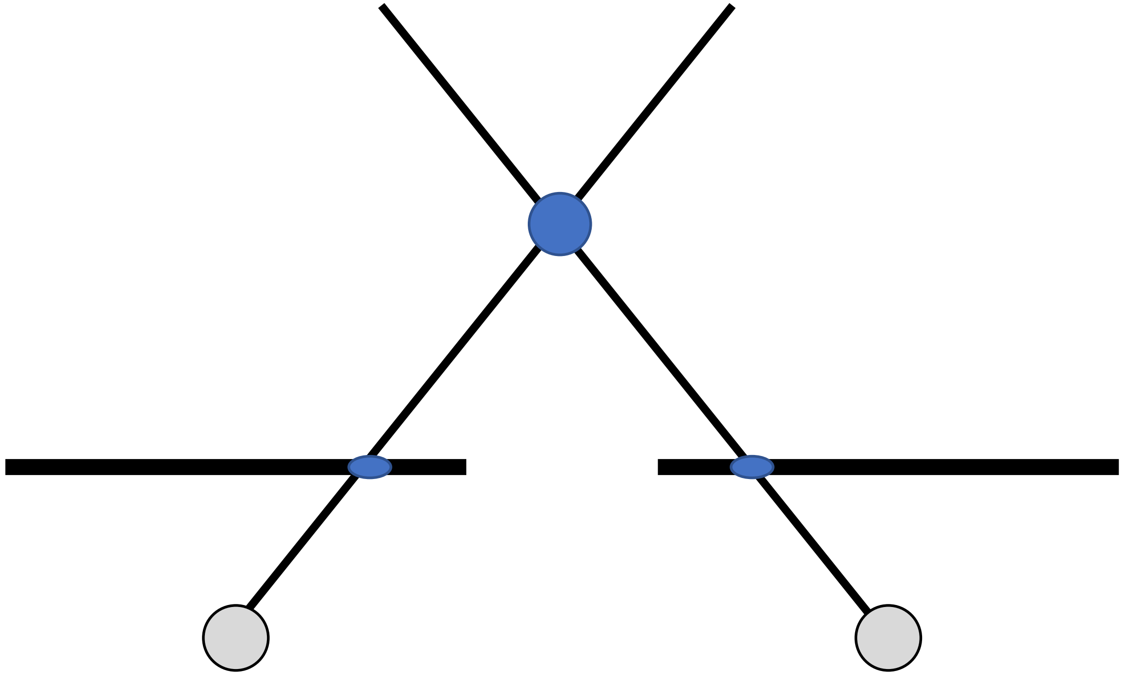
- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
  - Challenges in Stereo Matching
  - Stereo Matching with Dynamic Programming
  - Stereo Matching with Graph Cut algorithm
  - Stereo in Deep Learning era
- Active stereo with structured light

Why is matching challenging?

- Uniqueness:
  - Each point in one image should match at most one point in the other image
- Smoothness
  - We expect disparity to change slowly
- Occlusion
  - What if a pixel in the left image is not seen in the right image?
  - What if a pixel in the right image not seen in the right image?
- Ordering Constraint
  - If pixels (a,b,c) are ordered in left image, it should have same order in right image.
  - Not always true, depends on the depth of the object.

Non-local constraint: Uniqueness

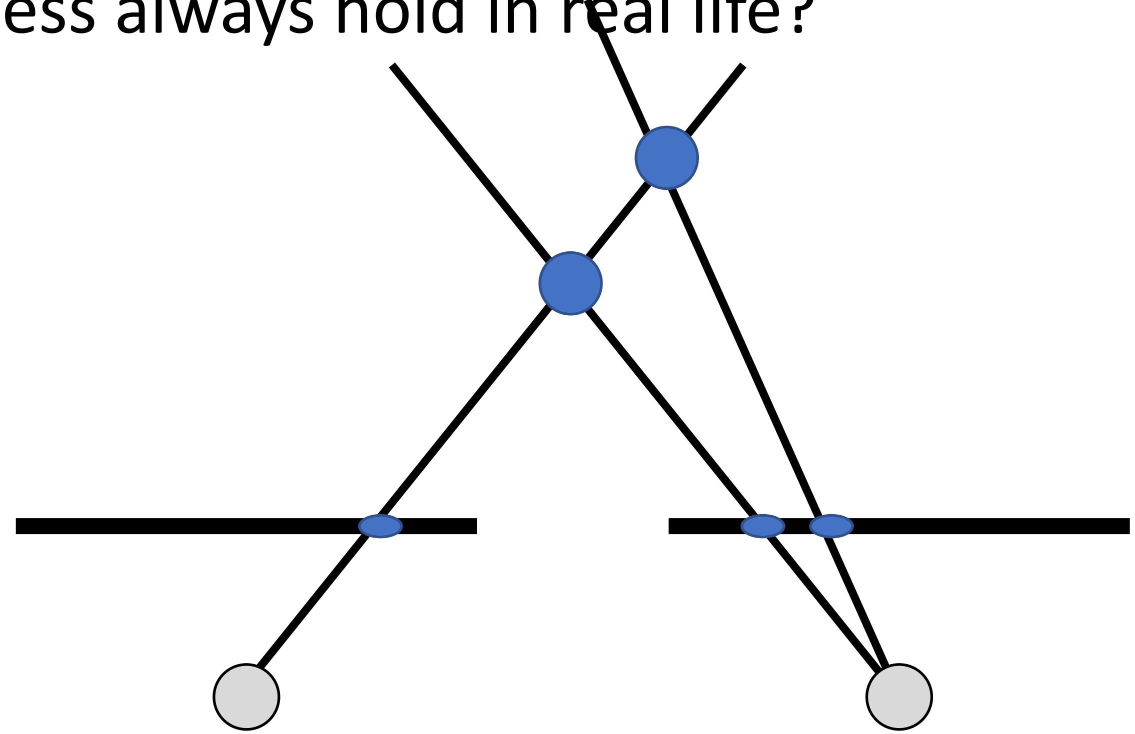
- Each point in one image should match at most one point in the other image
- Does uniqueness always hold in real life?



Source: J. Johnson and D. Fouhey

Non-local constraint: Uniqueness

- Each point in one image should match at most one point in the other image
- Does uniqueness always hold in real life?

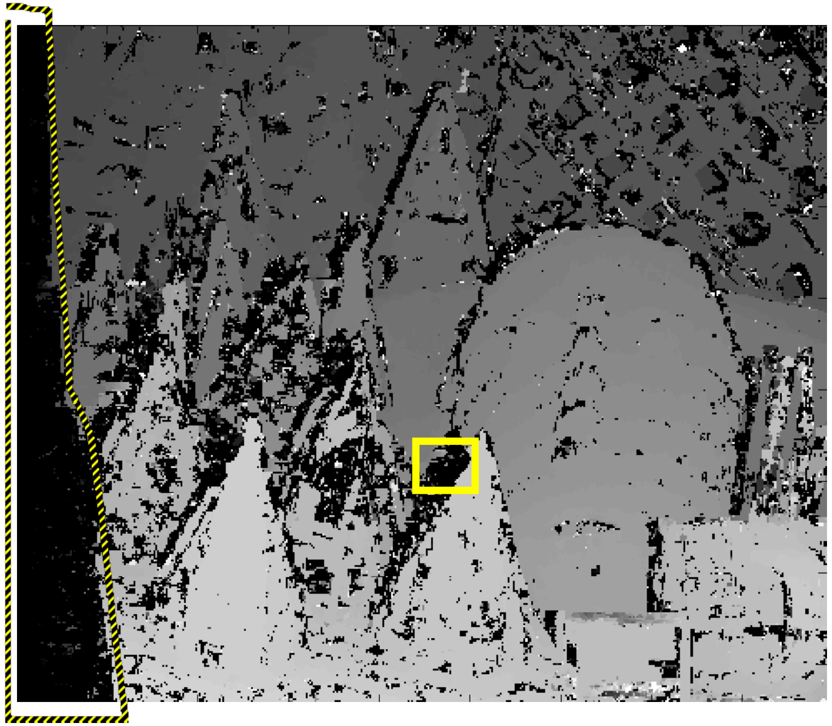


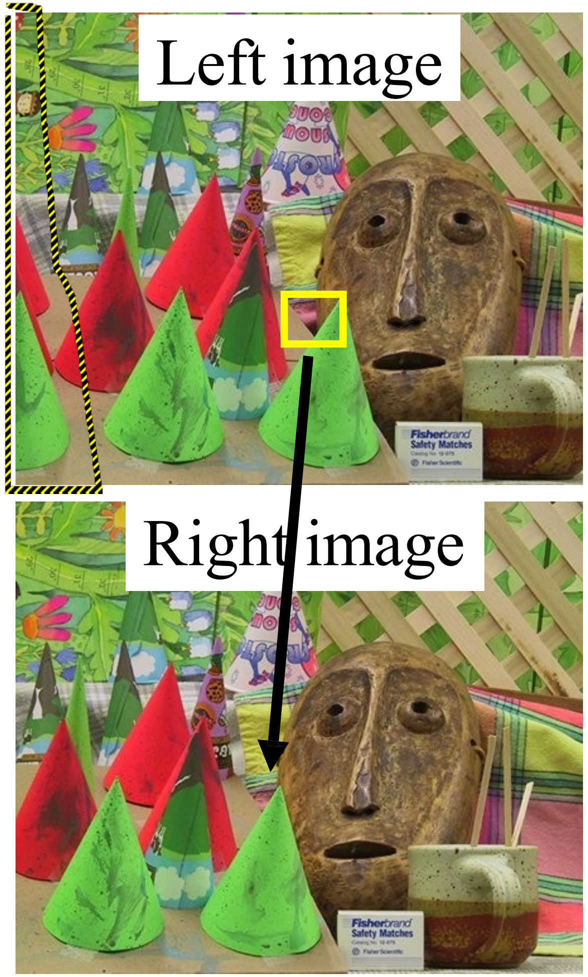
Source: [J. Johnson and D. Fouhey]

Non-local constraint: Smoothness

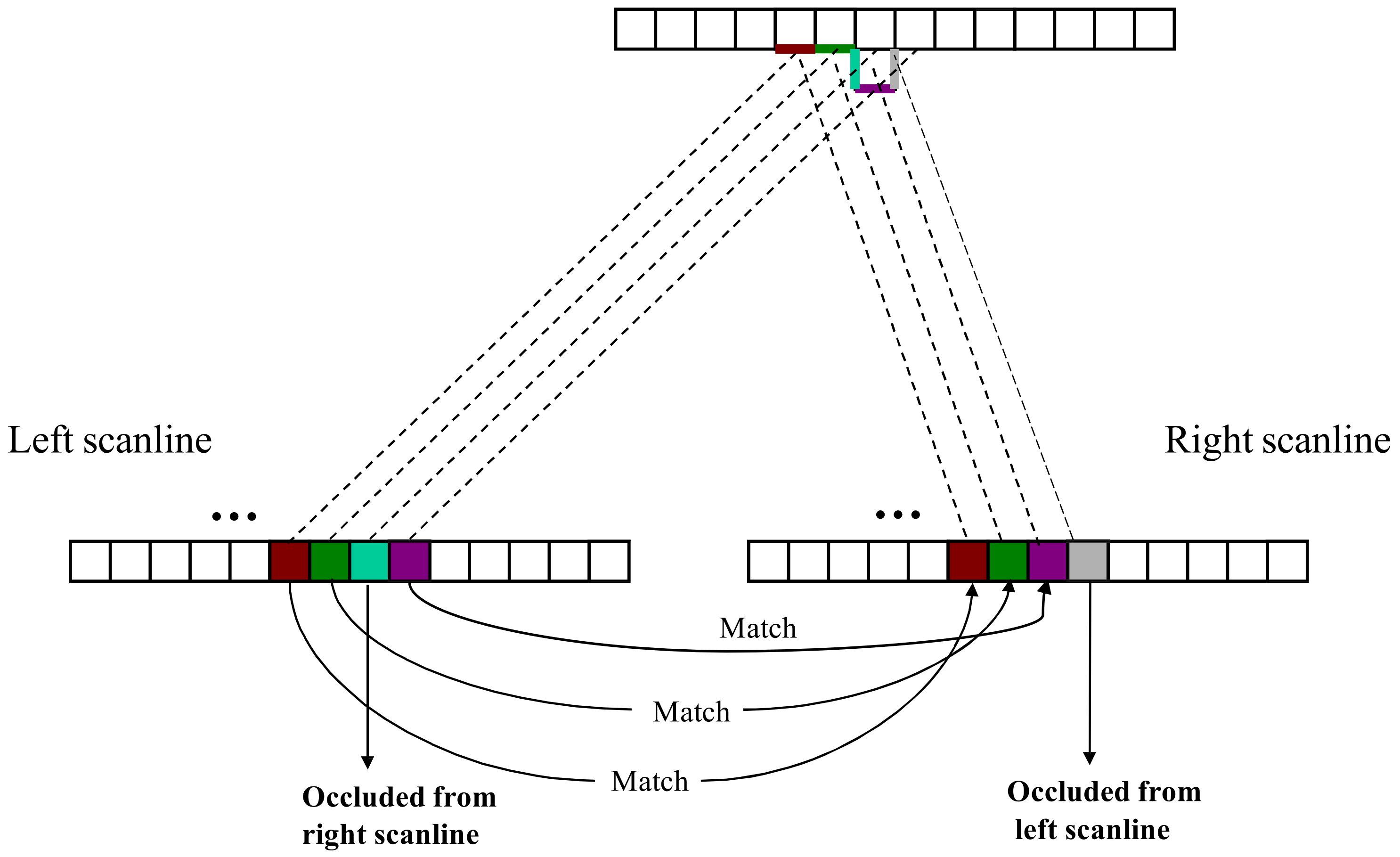
- We expect disparity values to change slowly (for the most part)

Occlusions: No matches

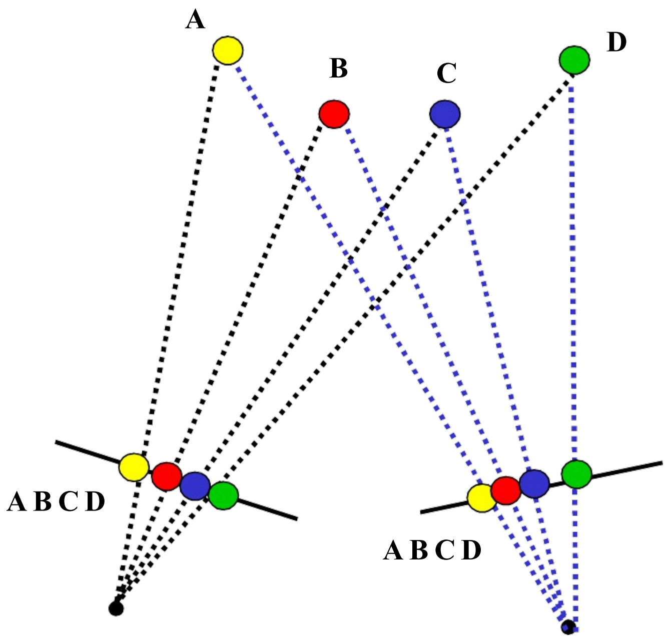




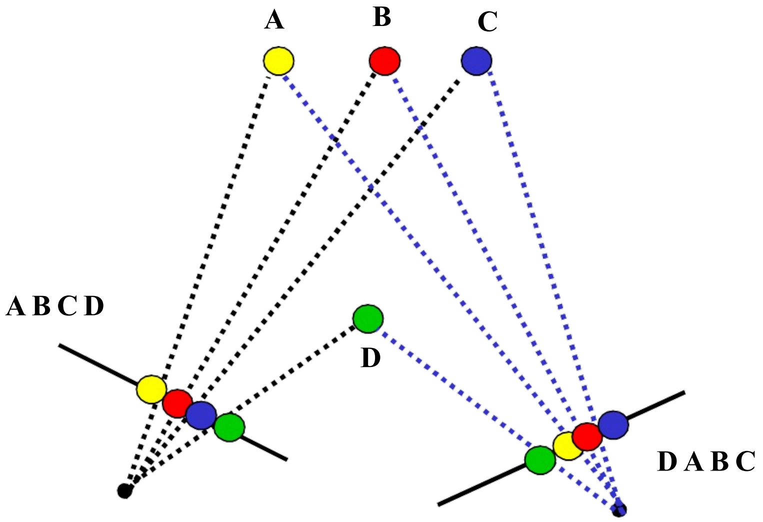
Dealing with Occlusions



Ordering Constraint



Ordering constraint…



…and its failure

Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
    - Challenges in Stereo Matching
    - **Stereo Matching with Dynamic Programming**
    - Stereo Matching with Graph Cut algorithm
    - Stereo in Deep Learning era
- Active stereo with structured light

Adding Inter-Scanline Consistency

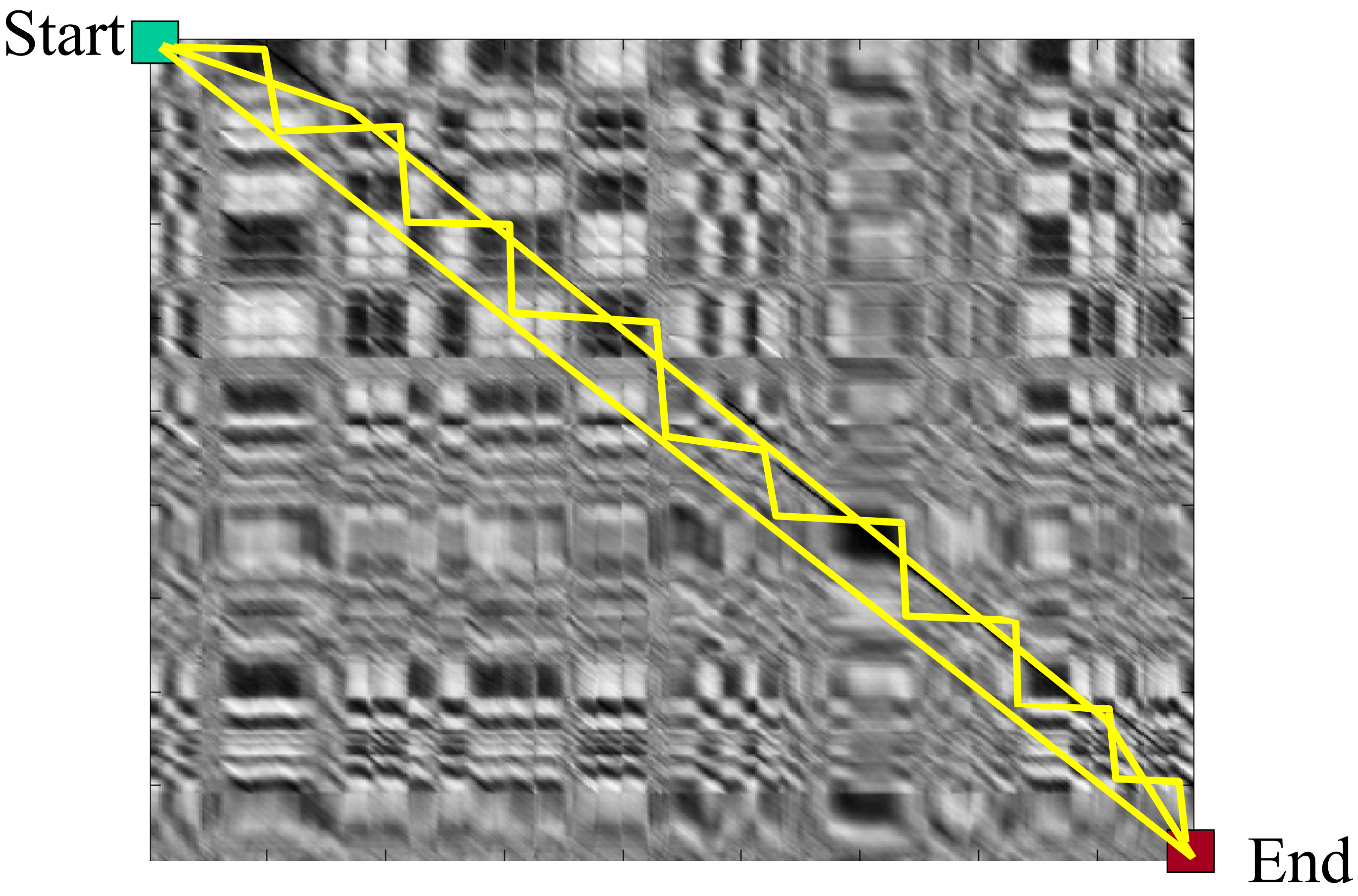
So far, each left image patch has been matched independently along the right epipolar line.

This can lead to errors.

We would like to enforce some consistency among matches in the same row (scanline).

DSI and Scanline Consistency

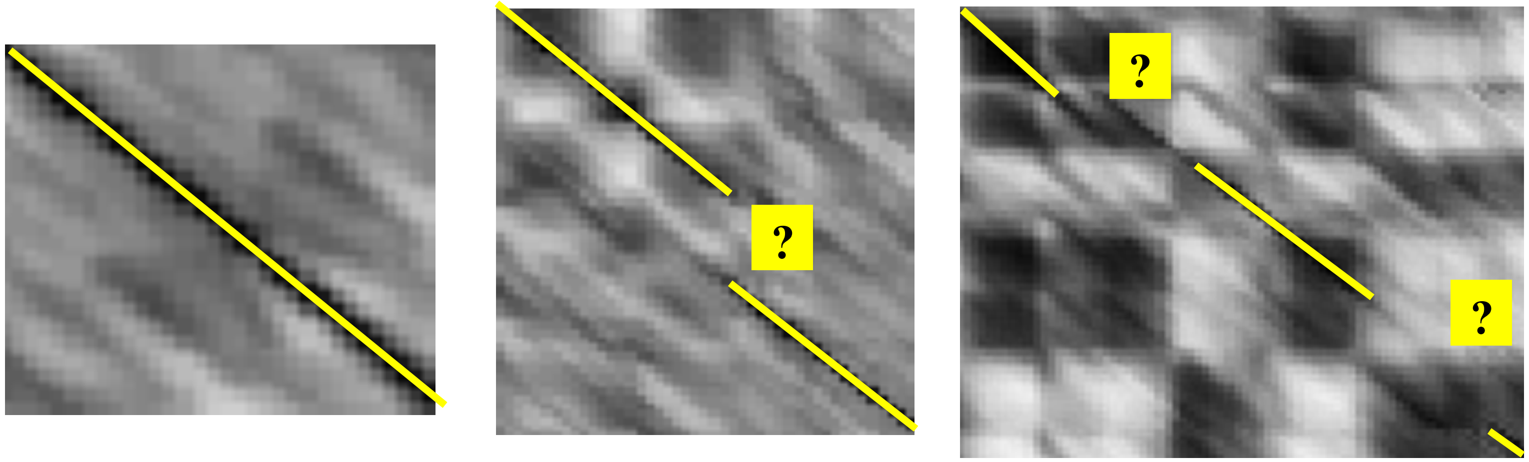
Assigning disparities to all pixels in left scanline now amounts to finding a connected path through the DSI



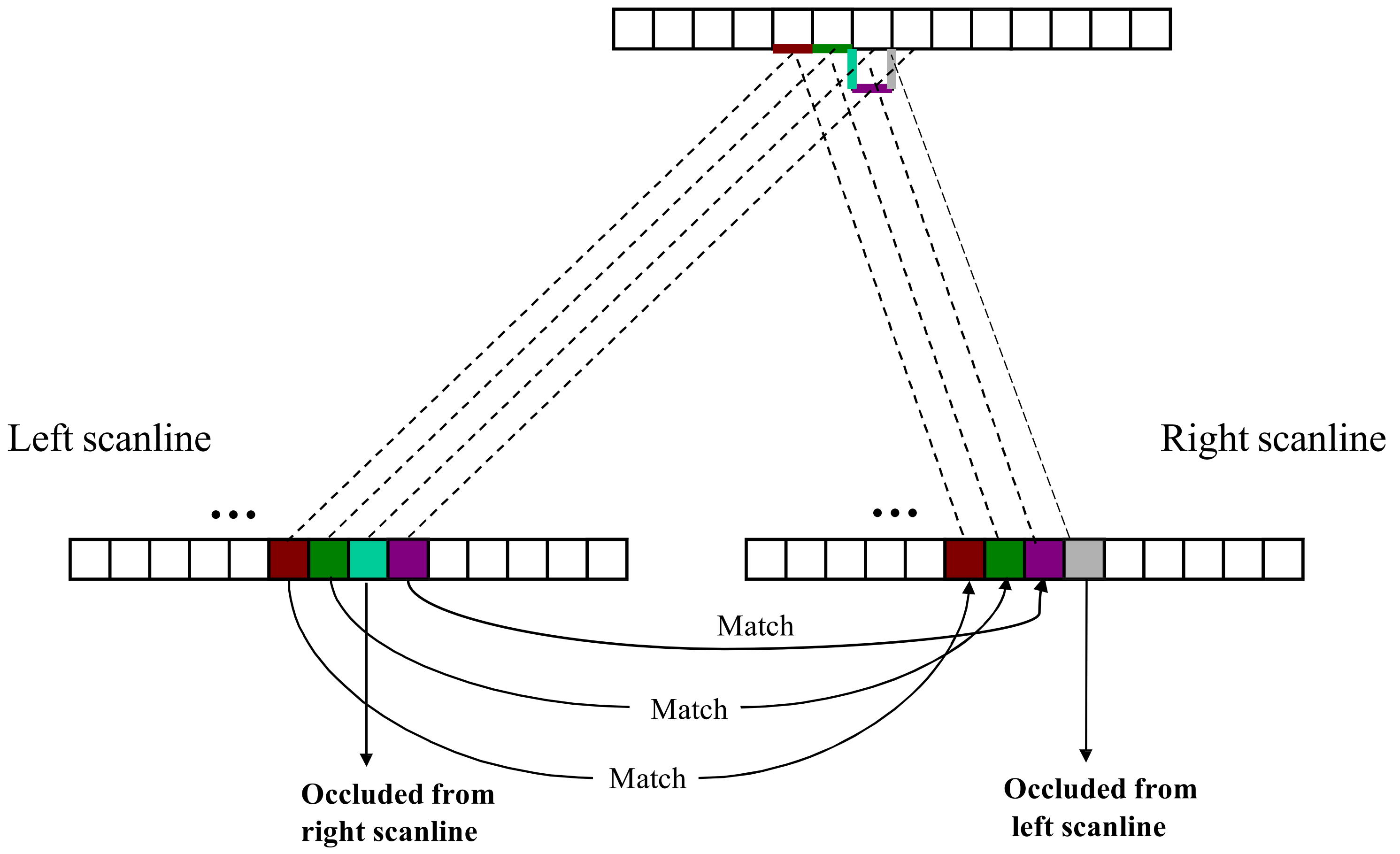
Lowest Cost Path

We would like to choose the "best" path.

Want one with lowest "cost" (Lowest sum of dissimilarity scores along the path)



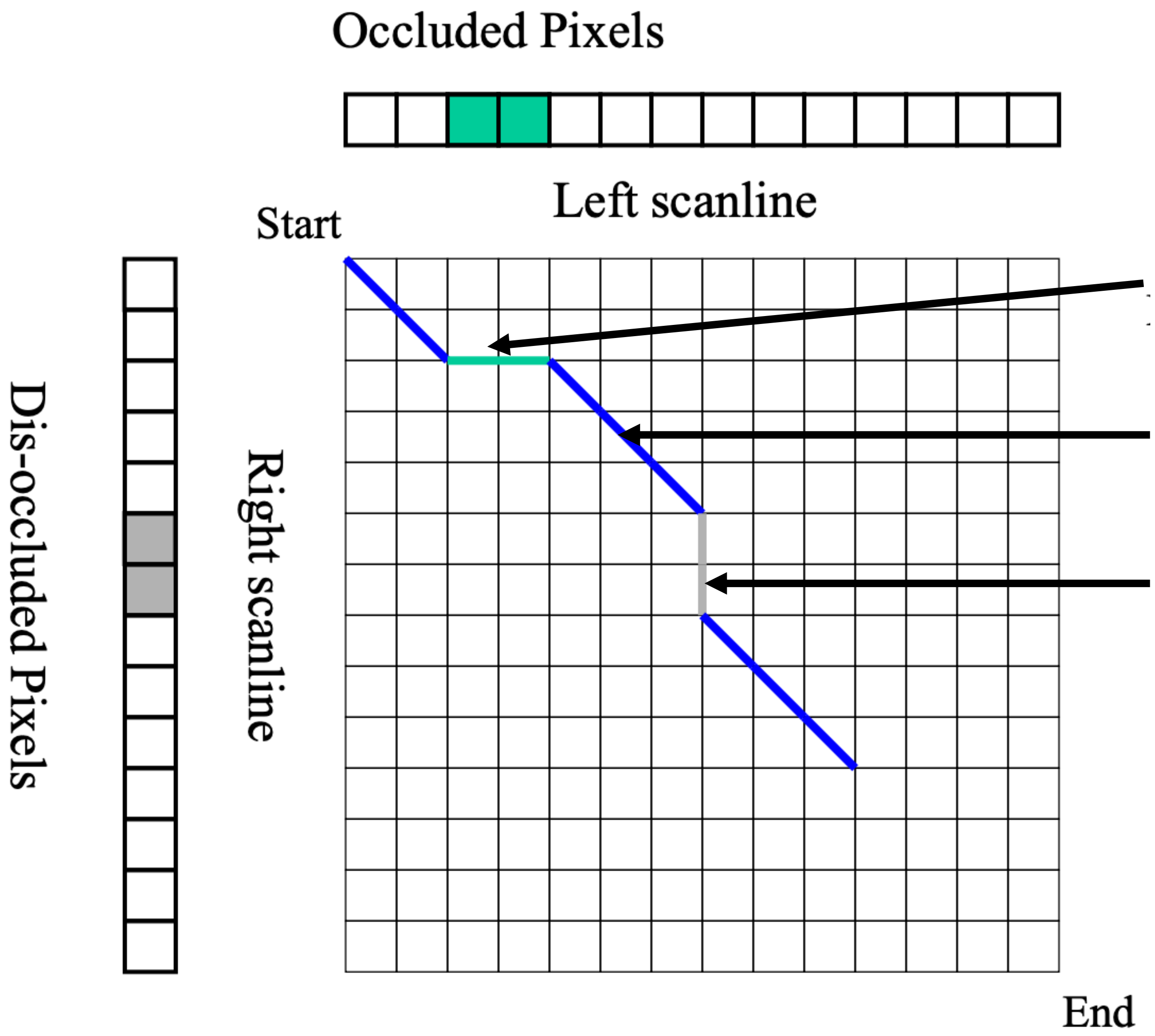
Dealing with Occlusions



Pixel is not present in the right image

Pixel is not present in the left image

Stereo Matching with Dynamic Programming



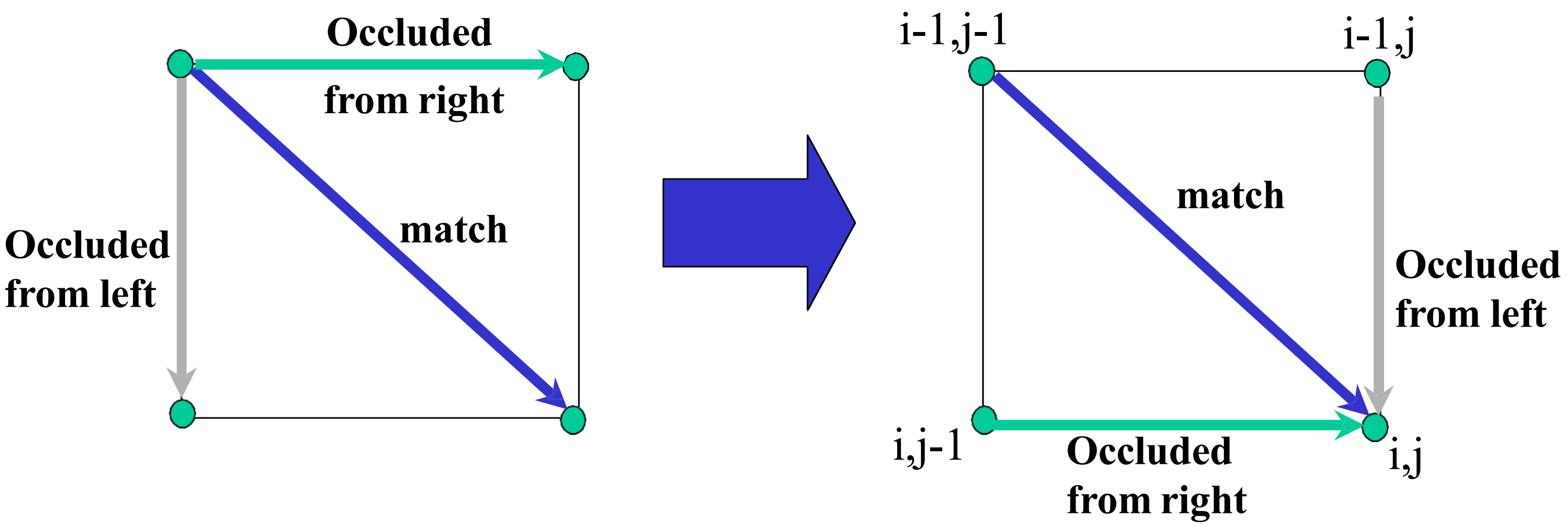
Dynamic programming yields the optimal path through grid. This is the best set of matches that satisfy the ordering constraint

Occluded from right image: Pixel is present in left image, but no match found in right image.

Correspondence found

Occluded from left image: Pixel is present in right image, but no match found in left image. Disparity mapping has a jump as it ignores these pixels.

Cox et.al. Stereo Matching



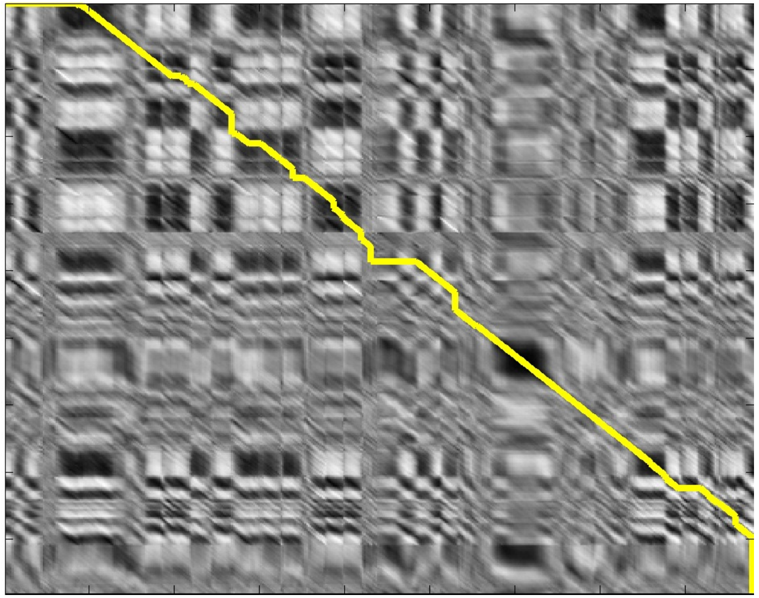
Three cases:

- Matching patches. Cost = dissimilarity score
- Occluded from right. Cost is some constant value.
- Occluded from left. Cost is some constant value.

```
C(i,j)= min([ C(i-1,j-1) + dissimilarity(i,j)
              C(i-1,j) + occlusionConstant,
              C(i,j-1) + occlusionConstant]);
```

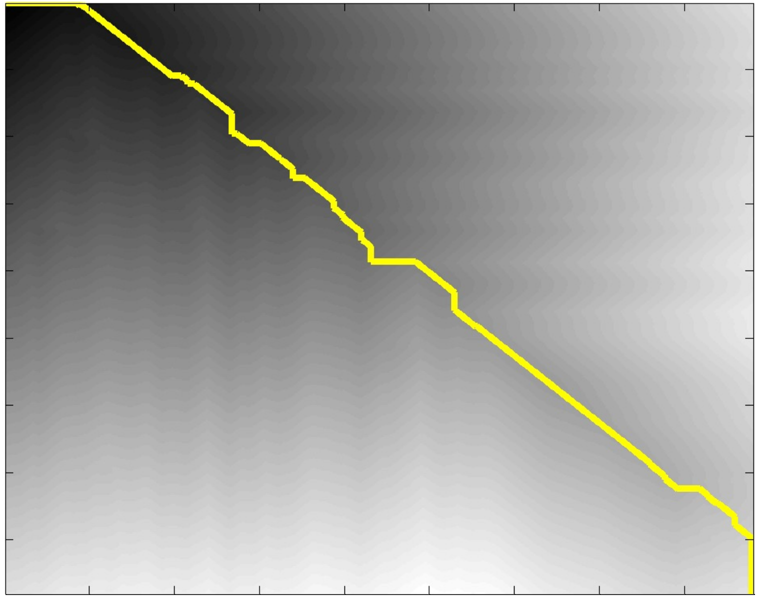
Real Scanline Example

DSI



DP cost matrix

(cost of optimal path from each point to END)

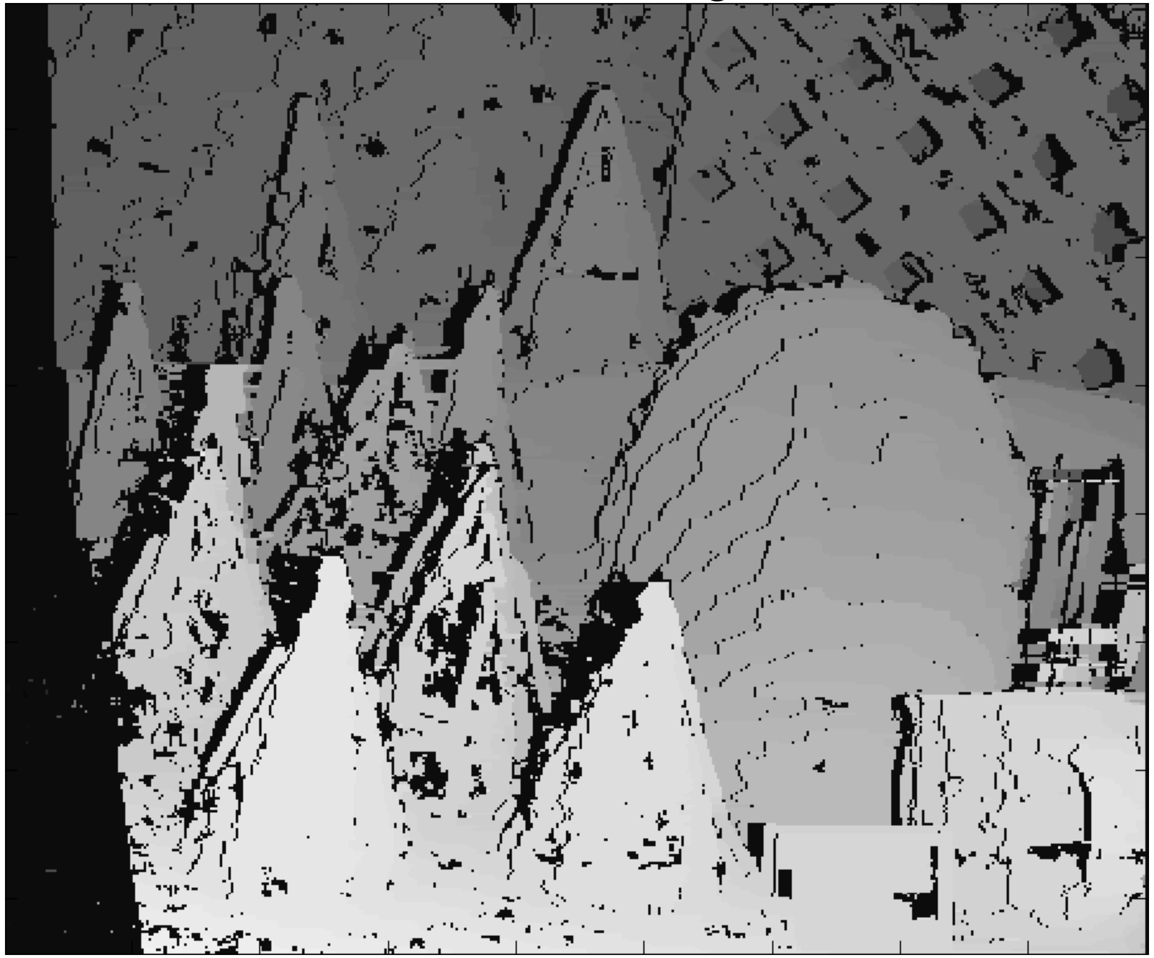


Every pixel in left column now is marked with either a disparity value, or an occlusion label.

Proceed for every scanline in left image.

Example

Result of DP alg

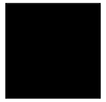


Result without DP (greedy)

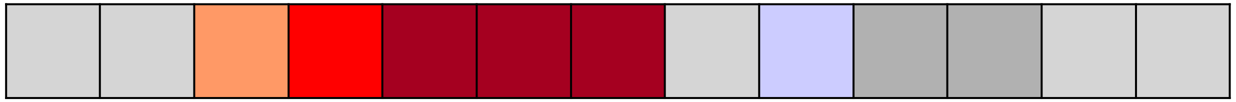
Result of DP alg. Black pixels = occluded.

Occlusion Filling

Simple trick for filling in gaps caused by occlusion.

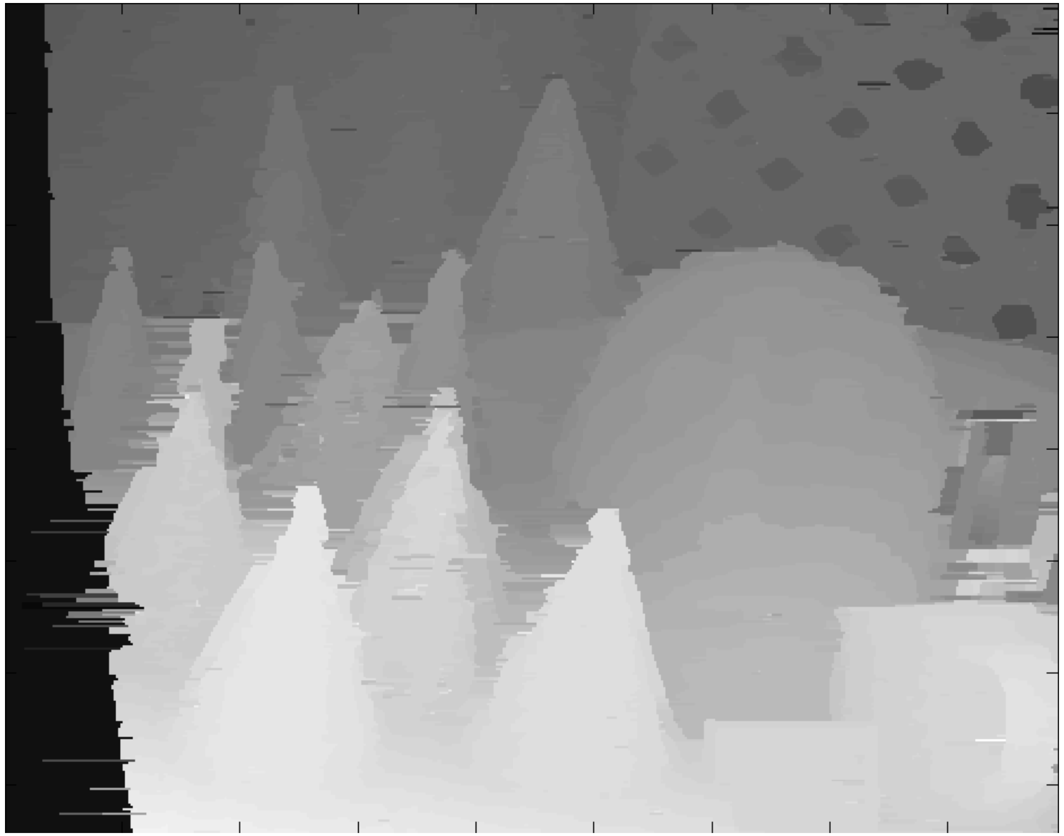
 = left occluded

Fill in left occluded pixels with value from the nearest valid pixel preceding it in the scanline.



Similarly, for right occluded, look for valid pixel to the right.

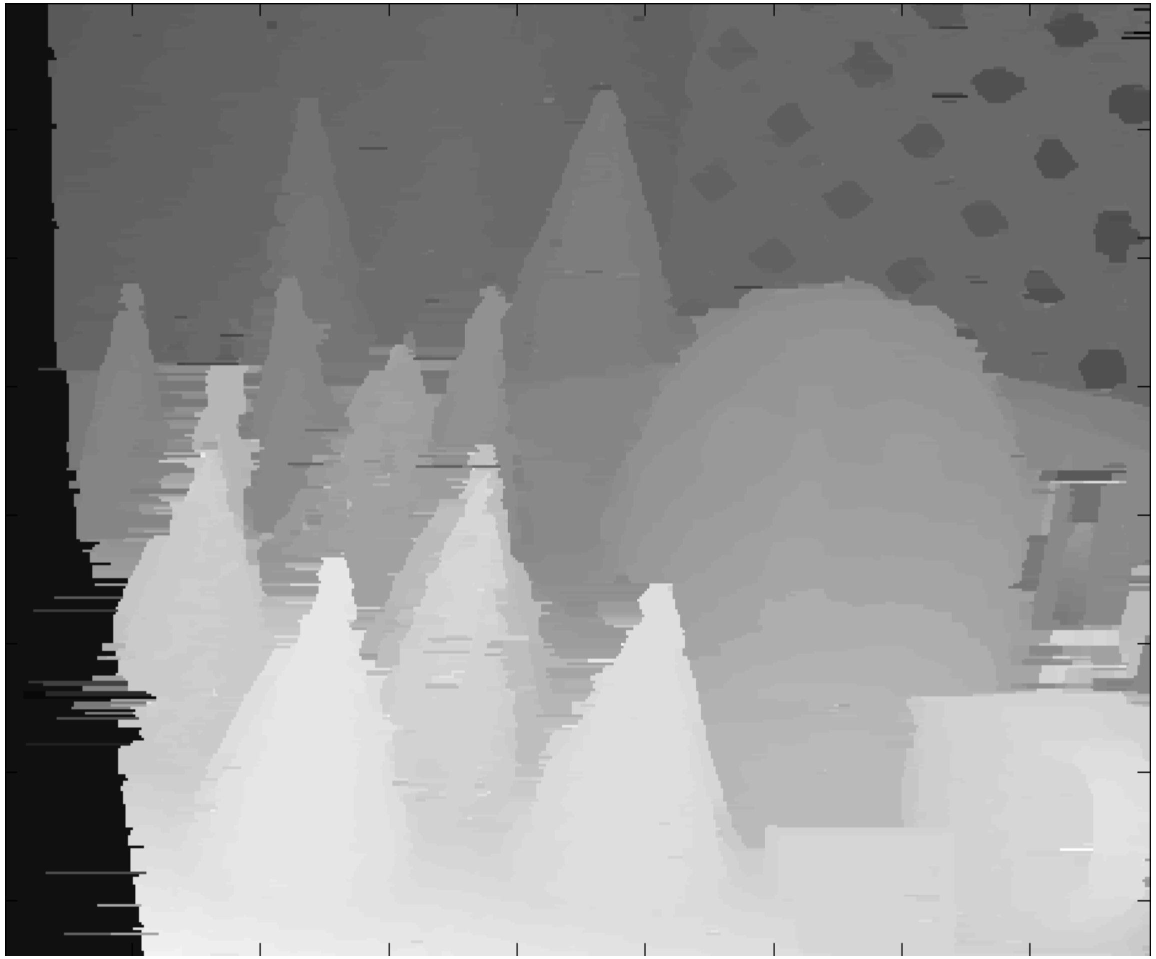
Example



Result of DP alg with occlusion filling.

Example

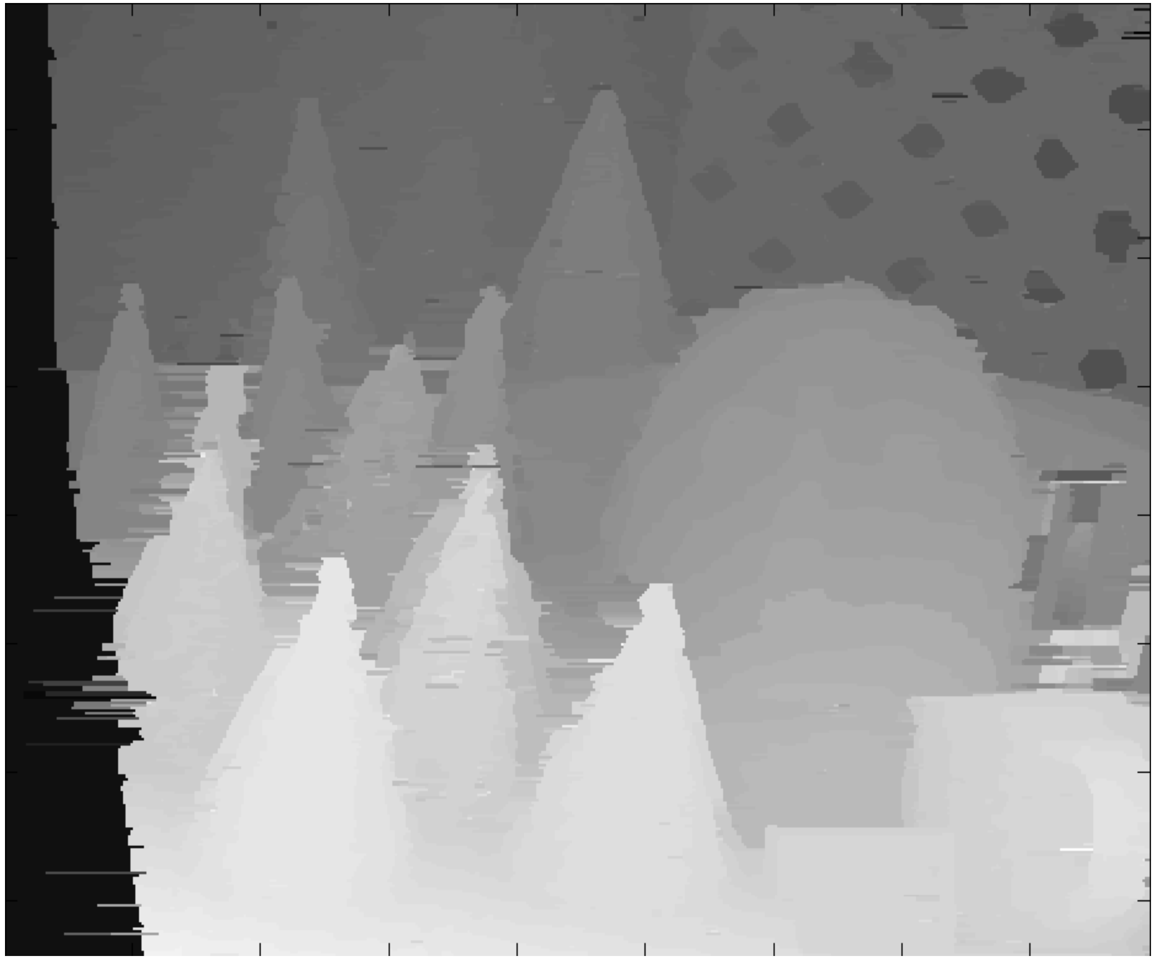
Result of DP alg with occlusion filling.



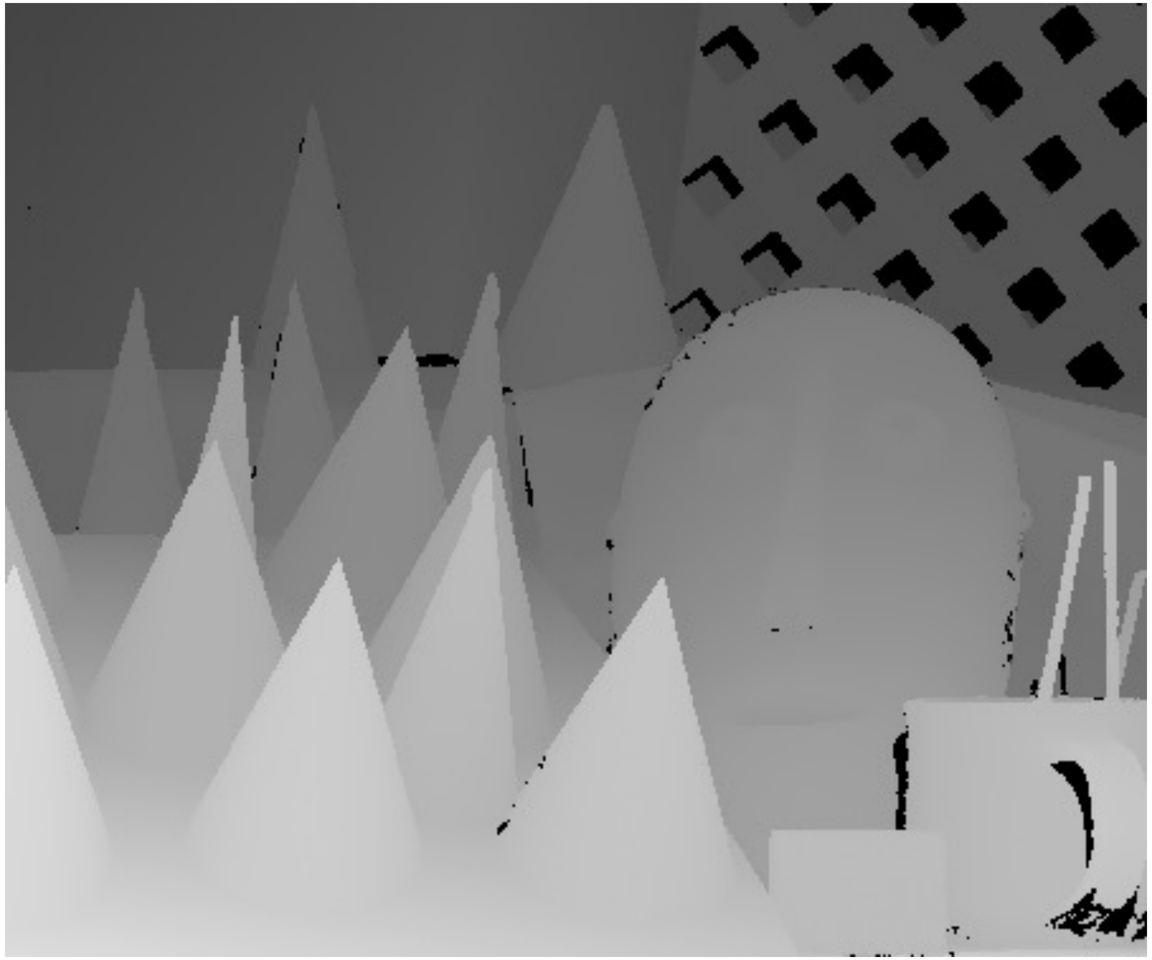
Result without DP (independent pixels)

Example

Result of DP alg with occlusion filling.

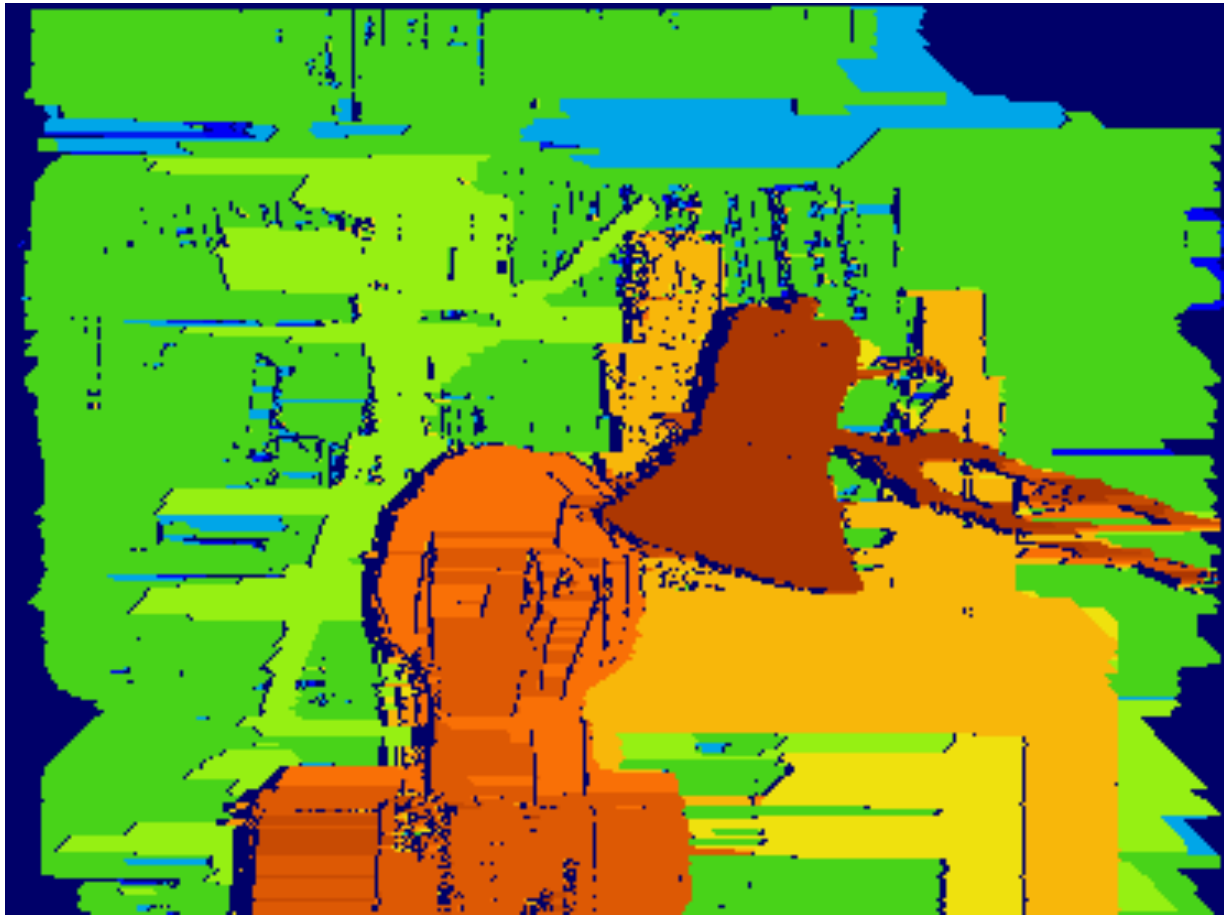


Ground Truth



Scanline stereo by dynamic programming

- Generates streaking artifacts!



Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
  - Challenges in Stereo Matching
  - Stereo Matching with Dynamic Programming
  - **Stereo Matching with Graph Cut algorithm**
  - Stereo in Deep Learning era
- Active stereo with structured light

Stereo as energy minimization

energy function
(for one pixel)

$$E(d) = \underbrace{E_d(d)}_{\text{data term}} + \lambda \underbrace{E_s(d)}_{\text{smoothness term}}$$

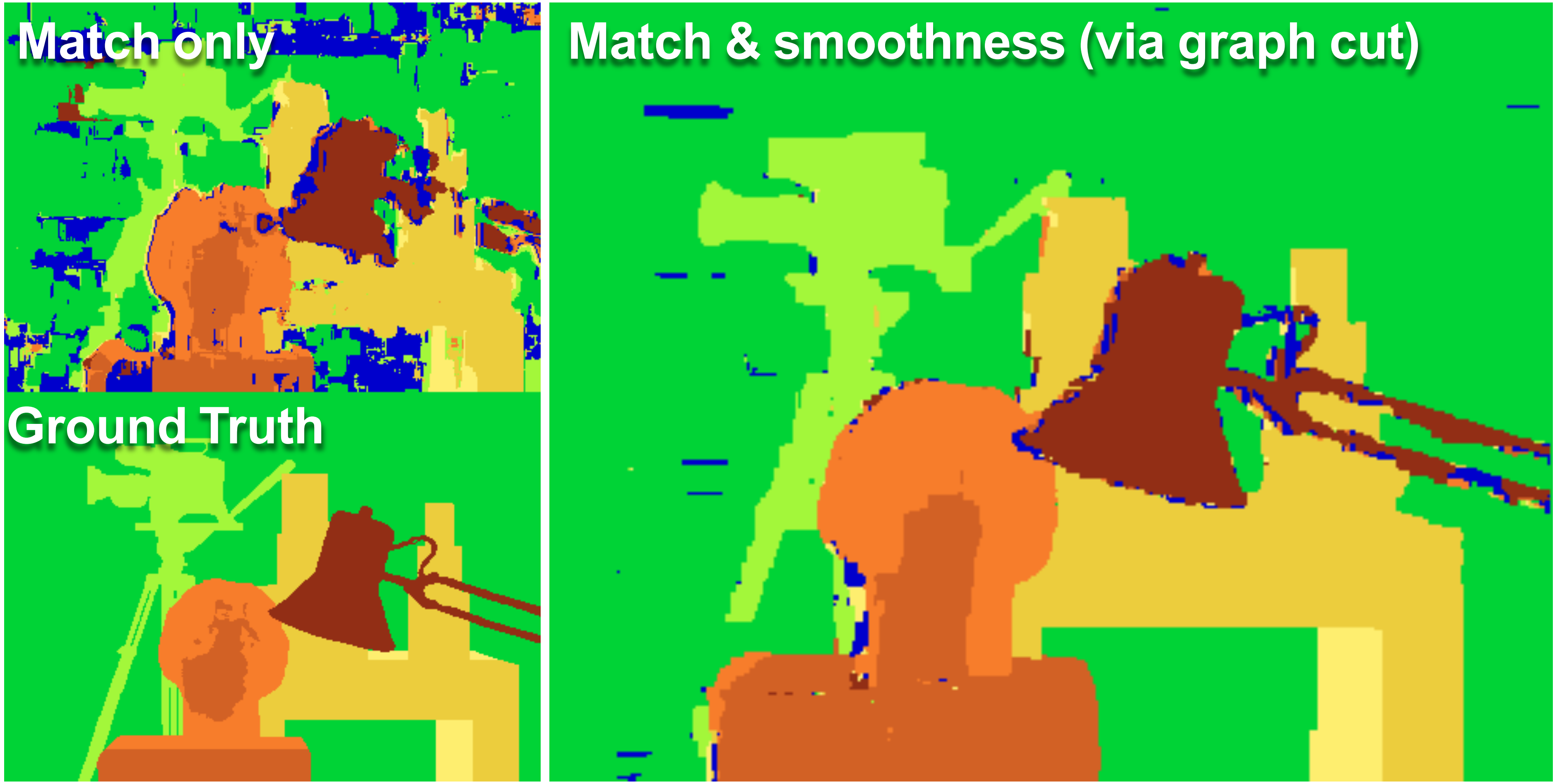
Want each pixel to find a good match in the other image

(block matching result)

Adjacent pixels should (usually) move about the same amount

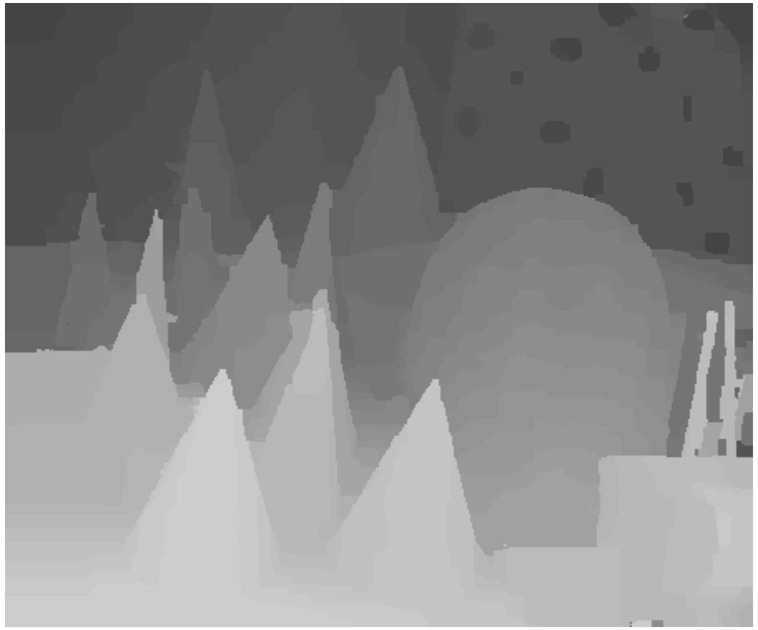
(smoothness function)

Energy Minimization via Graph Cut Algorithm

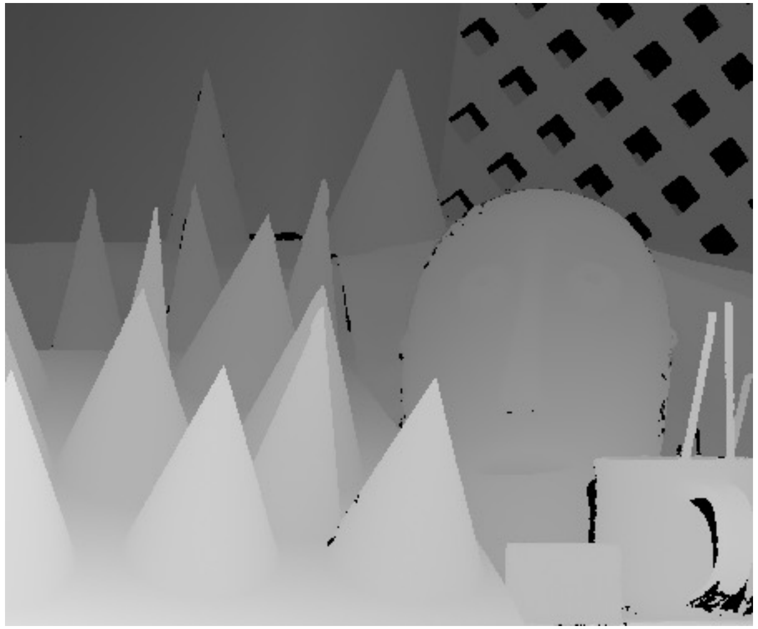


Y. Boykov, O. Veksler, and R. Zabih, [Fast Approximate Energy Minimization via Graph Cuts](), PAMI 2001

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



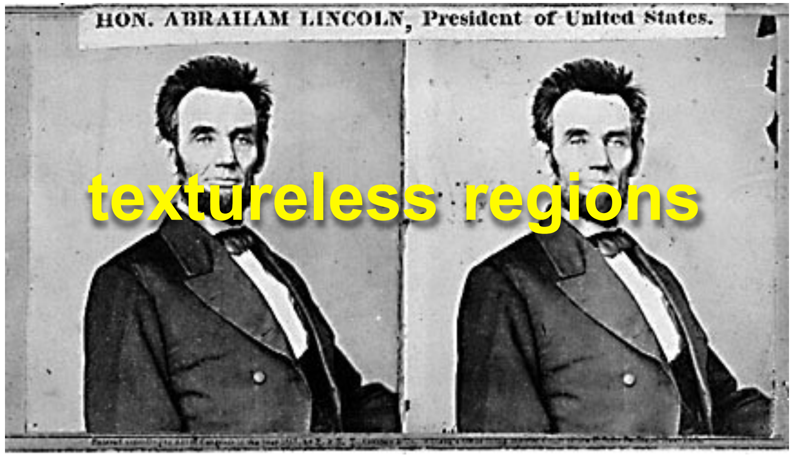
Algorithm Results



Ground truth

J. Sun, Y. Li, S.B. Kang, and H.-Y. Shum. "Symmetric stereo matching for occlusion handling". IEEE Conference on Computer Vision and Pattern Recognition, June 2005.

When will stereo block matching fail?



Stereo reconstruction pipeline

- Steps
  - Calibrate cameras
  - Rectify images
  - Compute disparity
  - Estimate depth

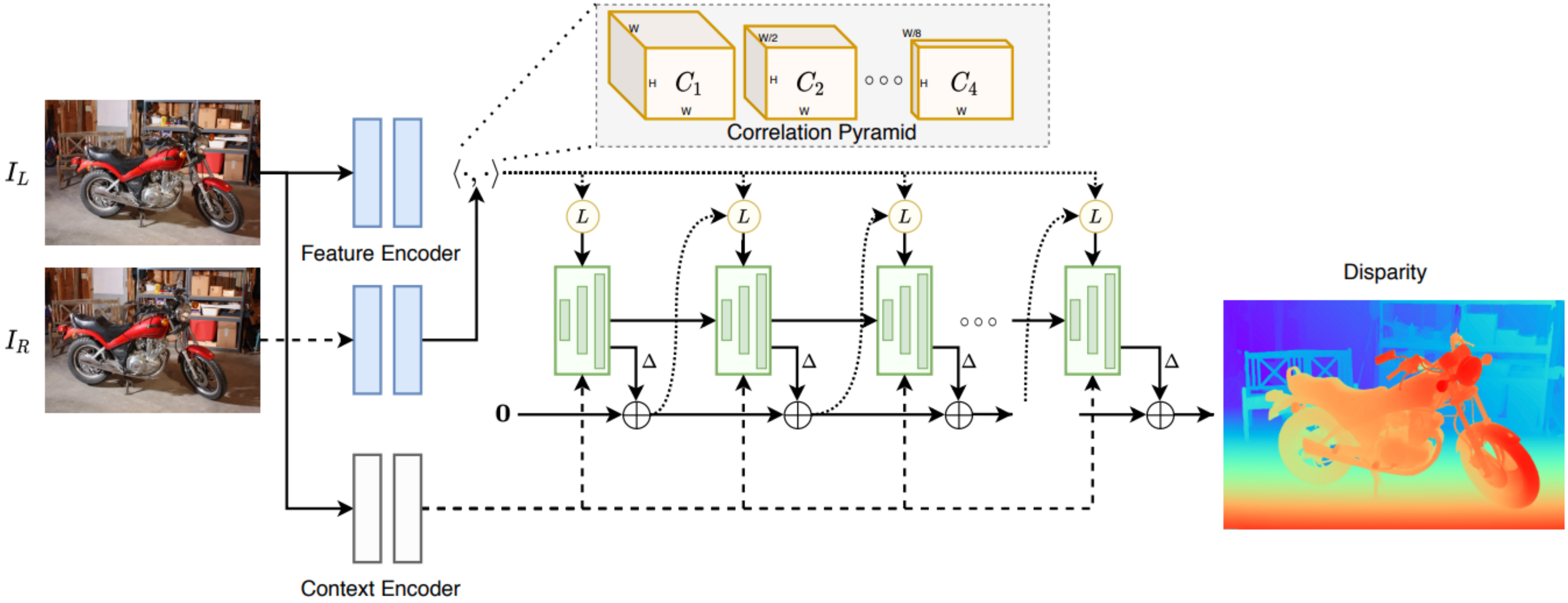
What will cause errors?

- Camera calibration errors
- Poor image resolution
- Occlusions
- Violations of brightness constancy (specular reflections)
- Large motions
- Low-contrast image regions

Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
  - Challenges in Stereo Matching
  - Stereo Matching with Dynamic Programming
  - Stereo Matching with Graph Cut algorithm
  - **Stereo in Deep Learning era**
- Active stereo with structured light

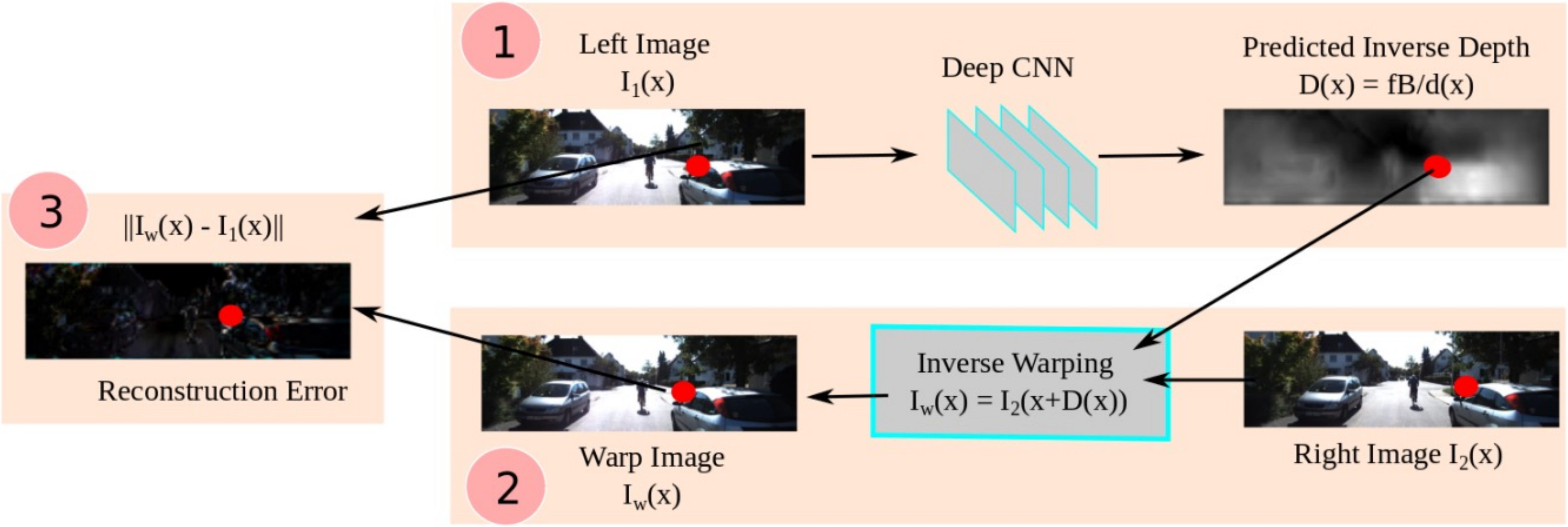
Stereo matching with deep networks



L. Lipson et al. [RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching](). arXiv 2021

Self-supervised depth estimation

Predict disparity



R. Garg et al. [Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue](). ECCV 2016

Stereo datasets

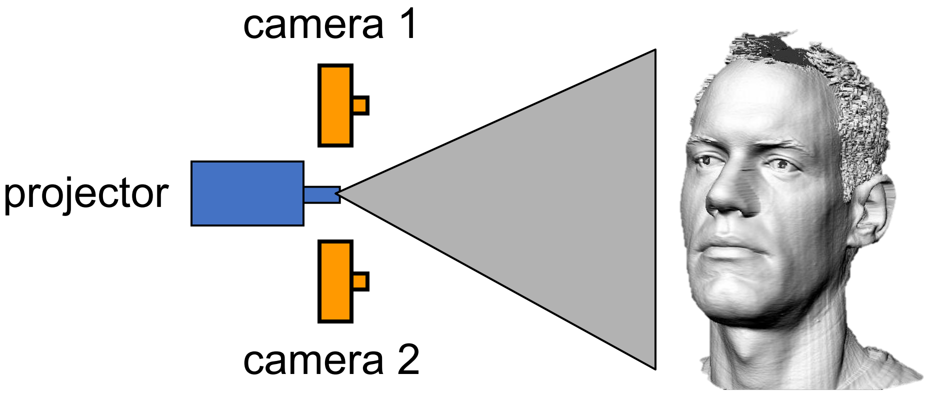
- [Middlebury stereo datasets]()
- [KITTI]()
- [Synthetic data]()

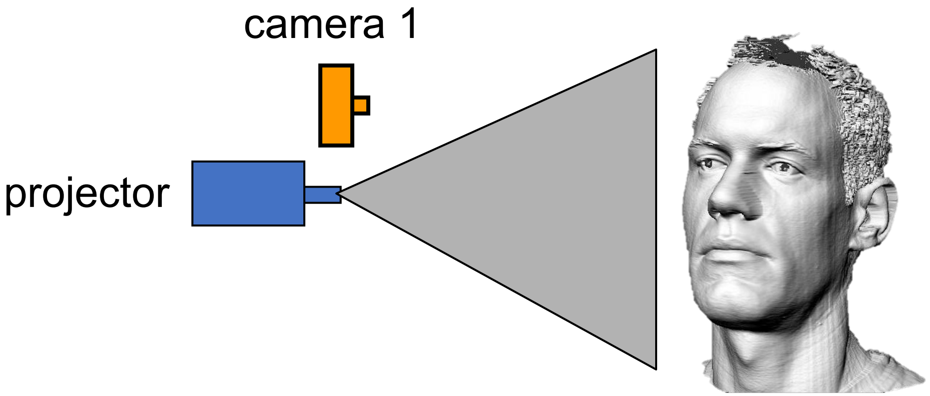
Today’s lecture

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching
- Active stereo with structured light

Active stereo with structured light

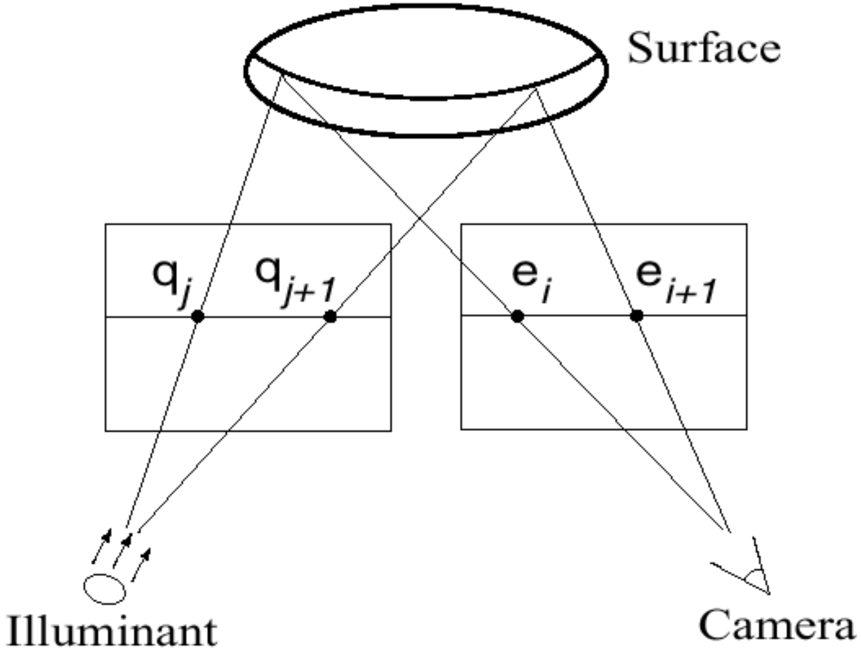
Li Zhang’s one-shot stereo

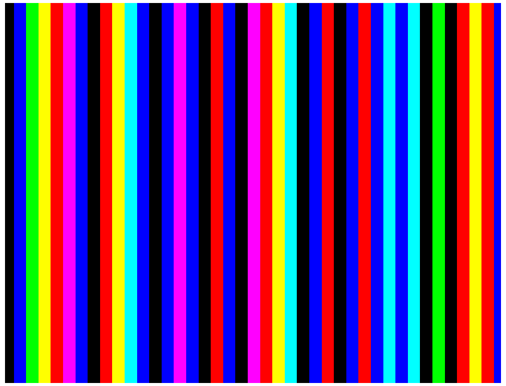


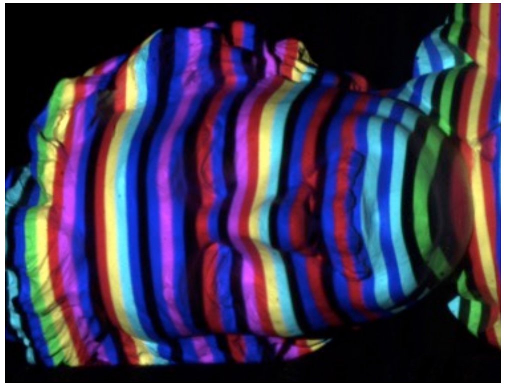


- Project “structured” light patterns onto the object
  - simplifies the correspondence problem
  - basis for active depth sensors, such as Kinect and iPhone X (using IR)

Active stereo with structured light

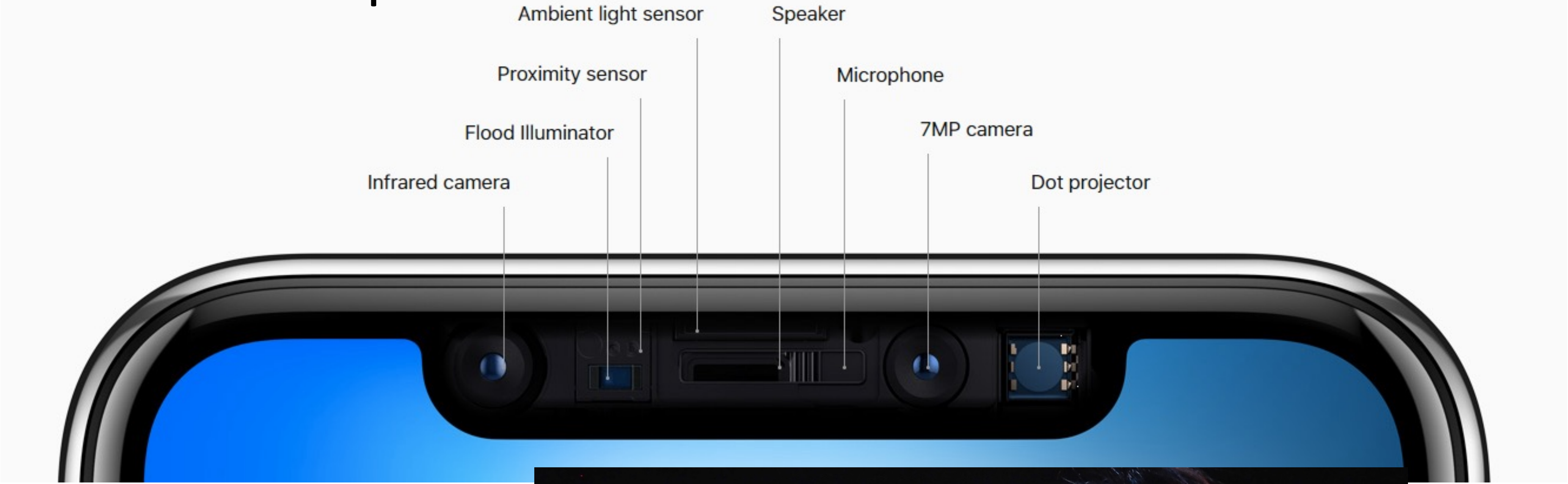






L. Zhang, B. Curless, and S. M. Seitz. [Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming.]() *3DPVT* 2002

Apple TrueDepth



https://www.cnet.com/news/apple-face-id-truedepth-how-it-works/

Active stereo with structured light

https://ios.gadgethacks.com/news/watch-iphone-xs-30k-ir-dots-scan-your-face-0180944/

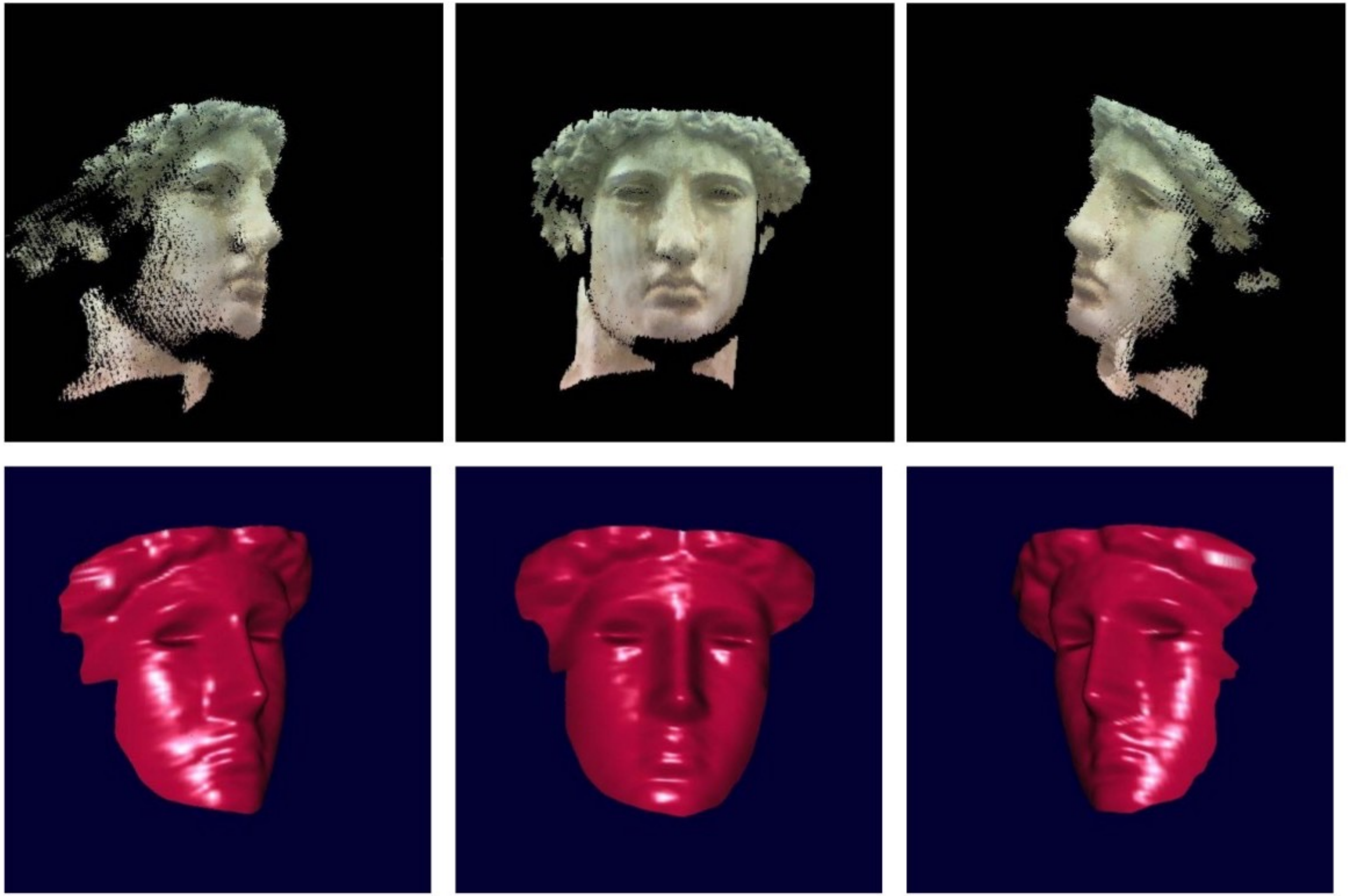
Kinect: Structured infrared light

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

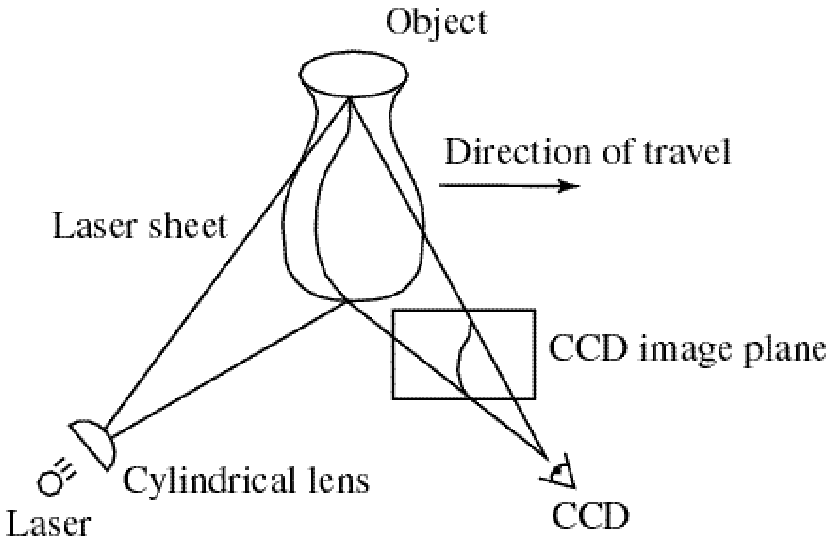
Use controlled ("structured") light to make correspondences easier

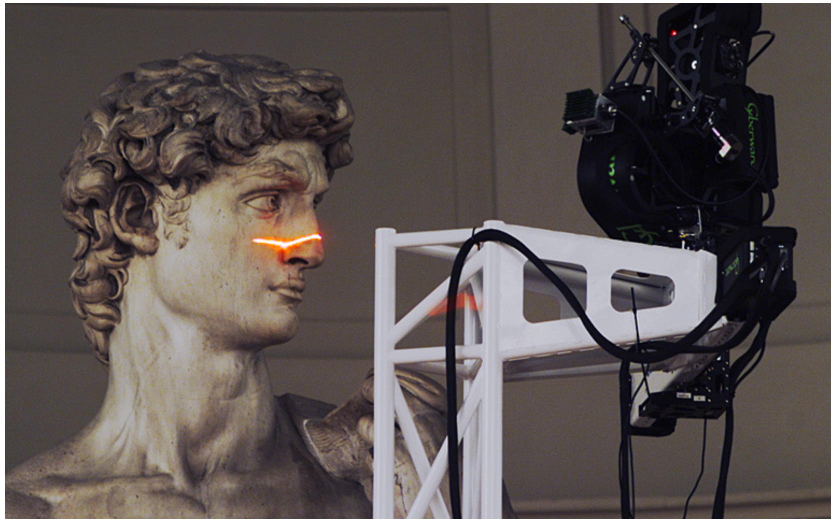
Disparity between laser points on the same scanline in the images determines the 3-D coordinates of the laser point on object

Use controlled ("structured") light to make correspondences easier



Laser scanning





Digital Michelangelo Project
http://graphics.stanford.edu/projects/mich/

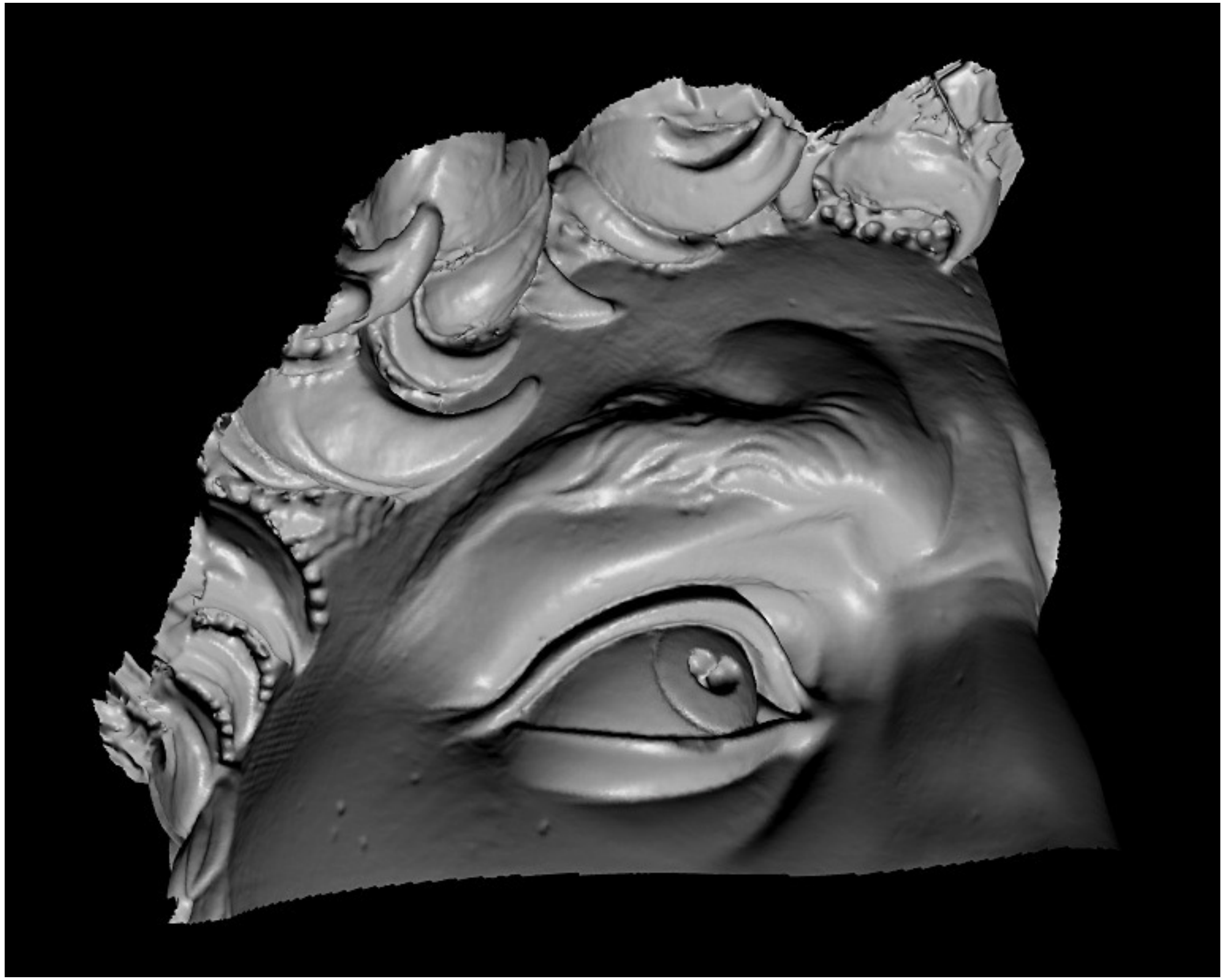
- Optical triangulation
  - Project a single stripe of laser light
  - Scan it across the surface of the object
  - This is a very precise version of structured light scanning

The Digital Michelangelo Project, Levoy et al.

Laser scanned models

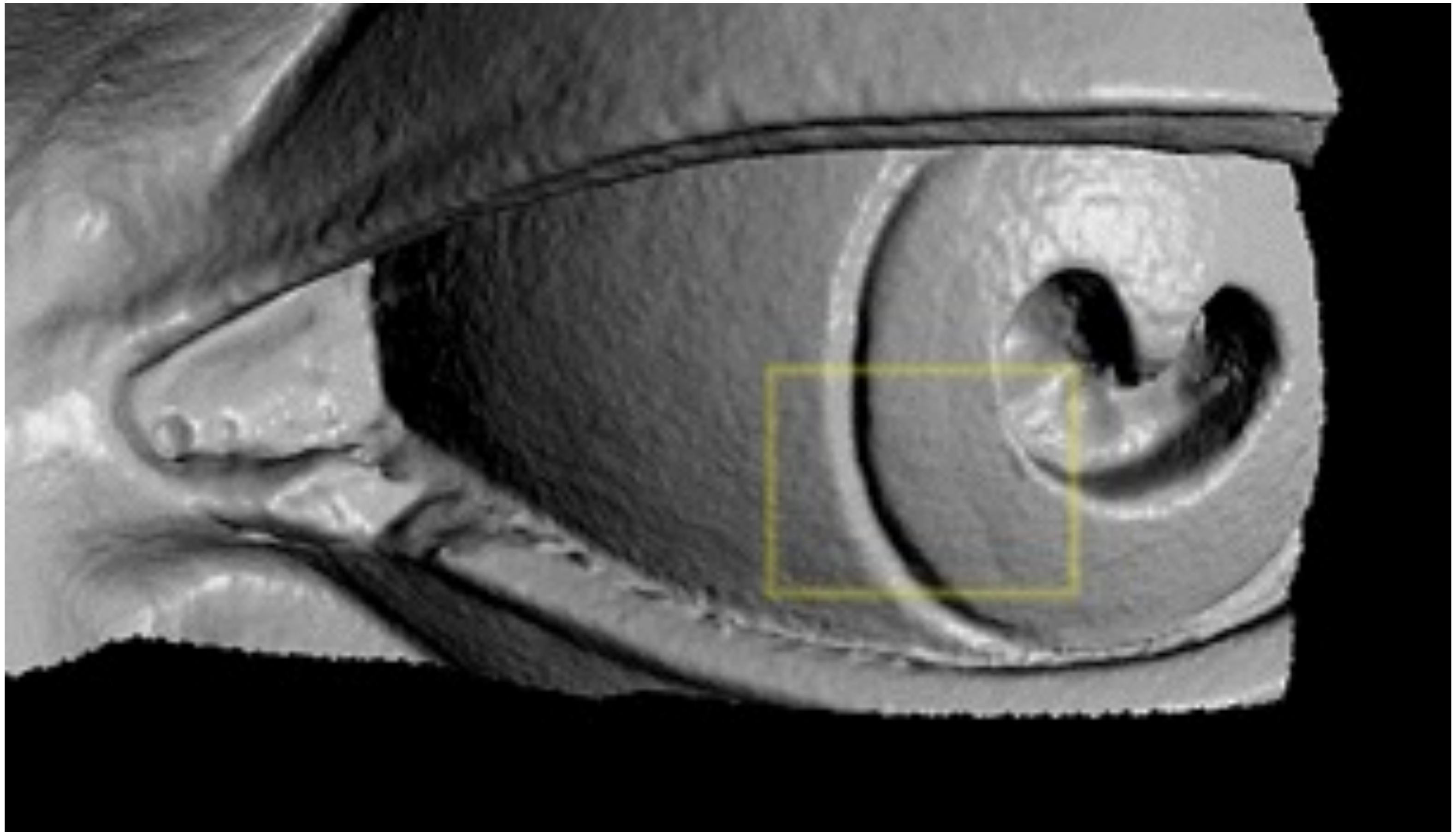
The Digital Michelangelo Project, Levoy et al.

Laser scanned models



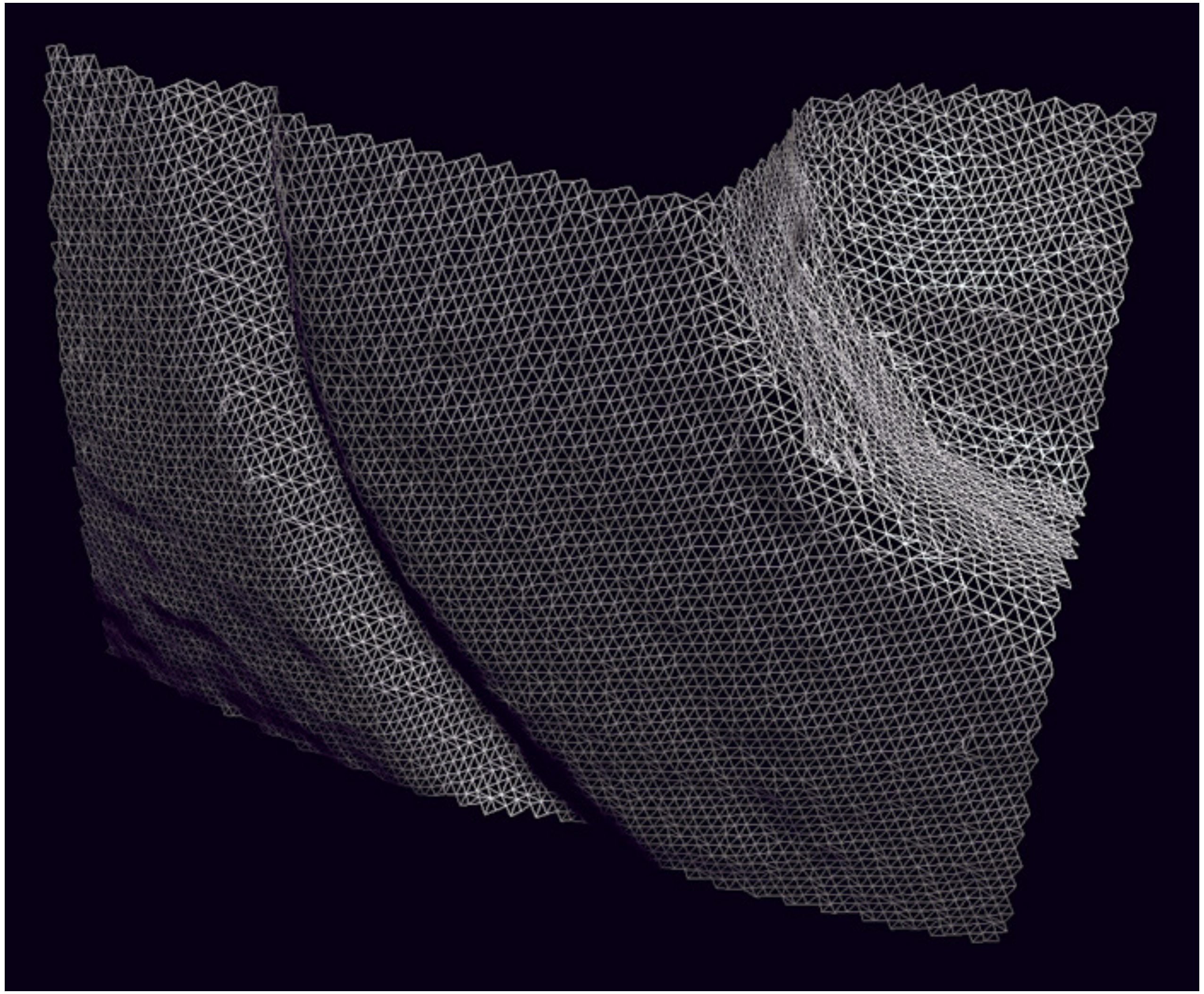
The Digital Michelangelo Project, Levoy et al.

Laser scanned models



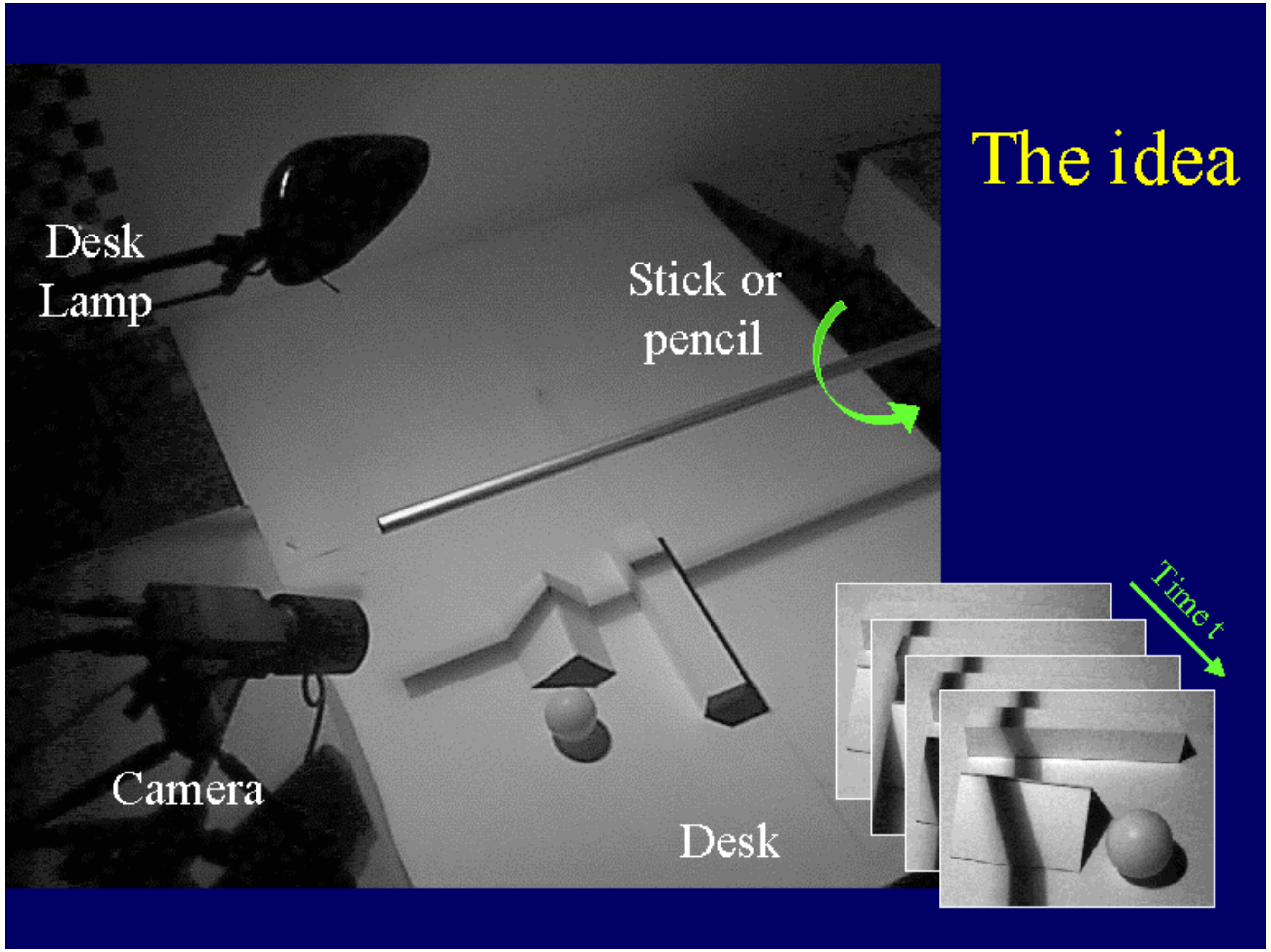
The Digital Michelangelo Project, Levoy et al.

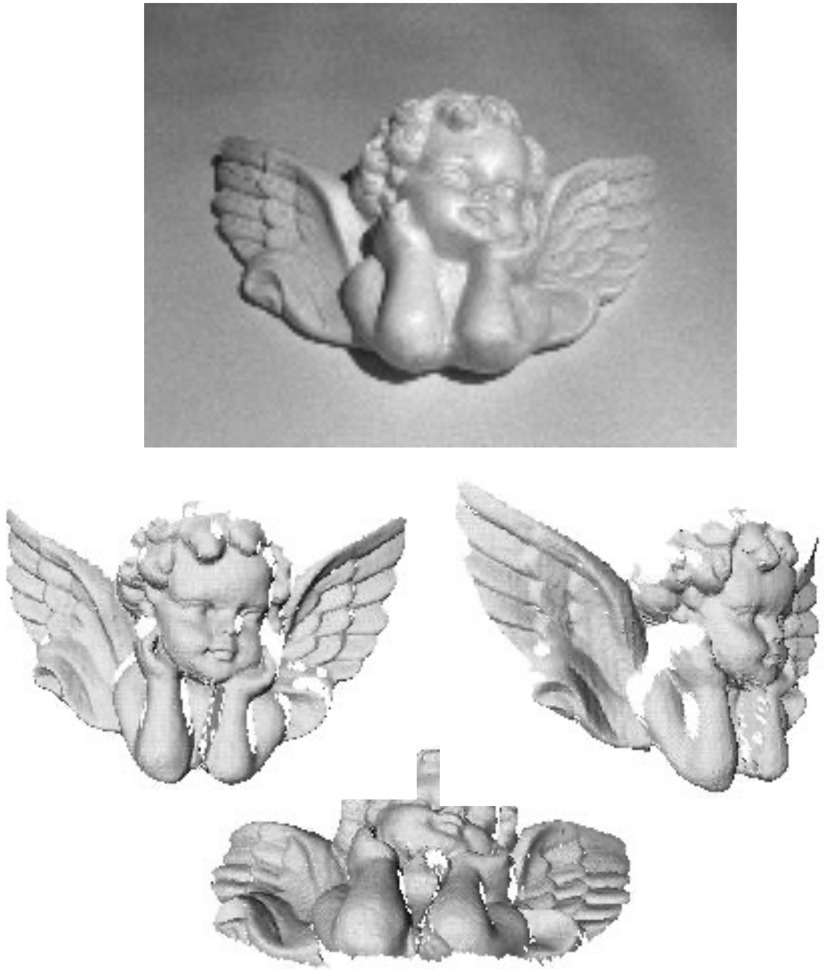
Laser scanned models



The Digital Michelangelo Project, Levoy et al.

3D Photography on your Desk





http://www.vision.caltech.edu/bouguetj/ICCV98/

Aligning range images

- A single range scan is not sufficient to capture a complex surface
- Need techniques to register multiple range images
- … which brings us to *multi-view stereo (next class!)*

B. Curless and M. Levoy, A Volumetric Method for Building Complex Models from Range Images,
SIGGRAPH 1996

Slide Credits

- CS5670, Introduction to Computer Vision, **Cornell Tech, by Noah Snavely.**
- CS 194-26/294-26: Intro to Computer Vision and Computational Photography, **UC Berkeley, by Angjoo Kanazawa.**
- CS 16-385: Computer Vision, **CMU, by Matthew O'Toole.**
- **CSE 486: Computer Vision, by Robert Collins, Penn State.**
- **CS 543 Computer Vision, by Stevlana Lazebnik, UIUC.**