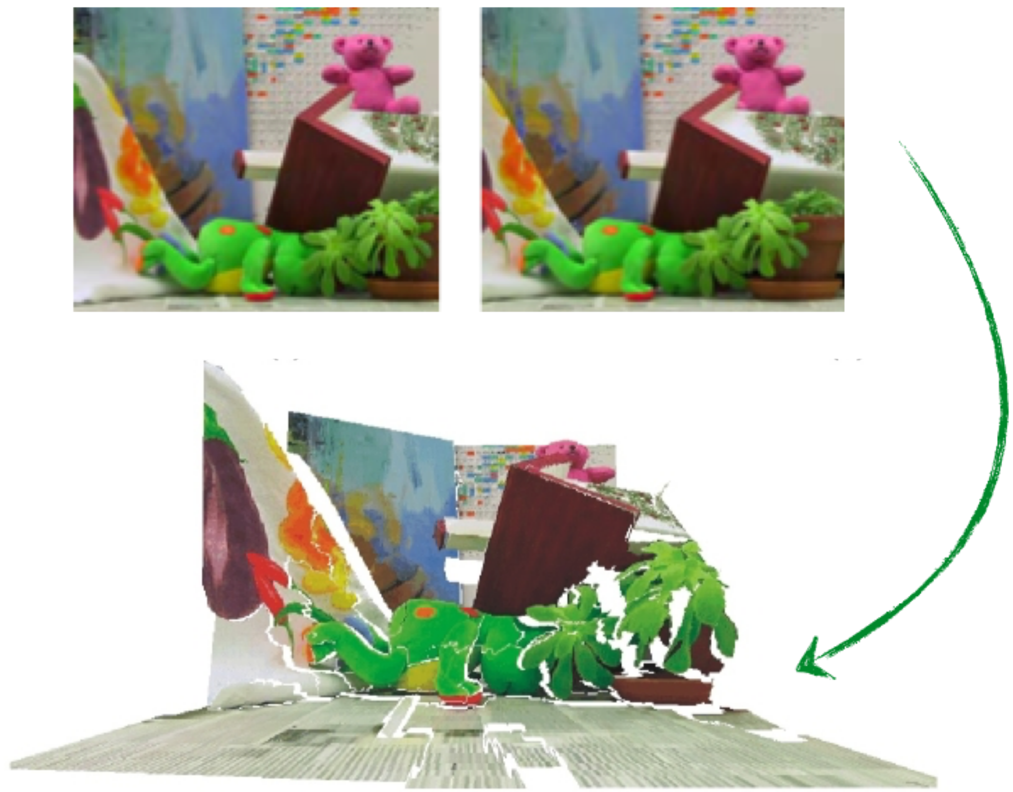
Two-View Stereo



Many slides adapted from Steve Seitz

Problem formulation

- **Given**: stereo pair (assumed calibrated)
- **Wanted**: dense depth map

Outline

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

Stereo vision and perception of depth

- What cues tell us about scene depth?

How Two Photographers Unknowingly Shot the Same Millisecond in Time

MAR 07, 2018 RON RISMAN

PetaPixel

https://petapixel.com/2018/03/07/two-photographers-unknowingly-shot-millisecond-time/

How Two Photographers Unknowingly Shot the Same Millisecond in Time

MAR 07, 2018 RON RISMAN

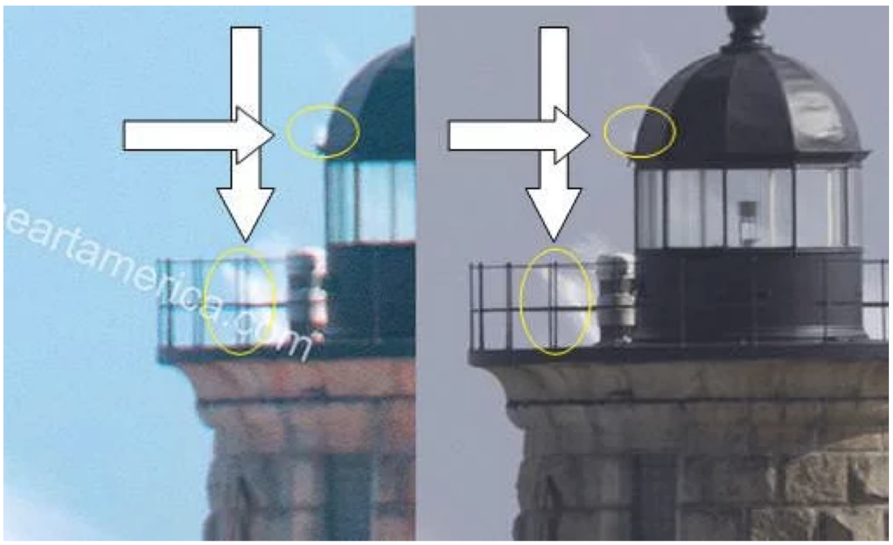
PetaPixel

https://petapixel.com/2018/03/07/two-photographers-unknowingly-shot-millisecond-time/

How Two Photographers Unknowingly Shot the Same Millisecond in Time

MAR 07, 2018 RON RISMAN

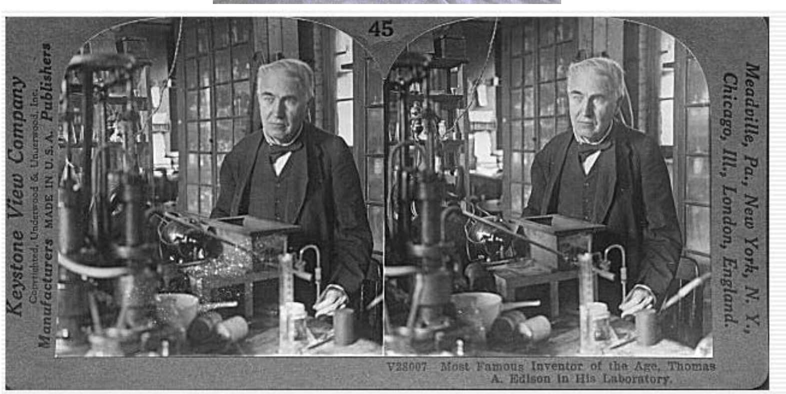
PetaPixel



https://petapixel.com/2018/03/07/two-photographers-unknowingly-shot-millisecond-time/

History: Stereograms

- Humans can fuse pairs of images to get a sensation of depth

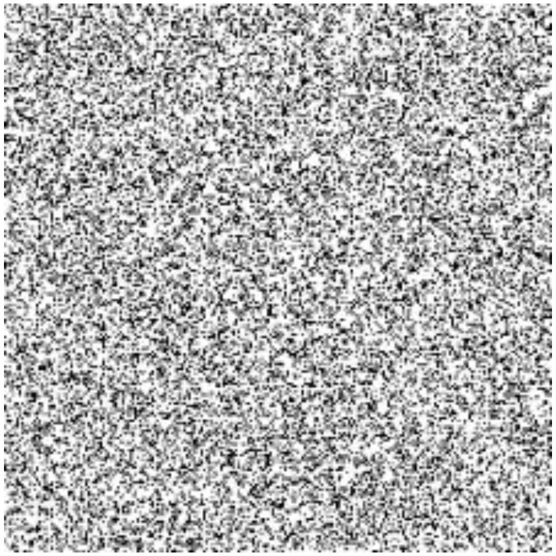


Stereograms: Invented by Sir Charles Wheatstone, 1838

https://en.wikipedia.org/wiki/Stereoscopy

History: Random dot stereograms

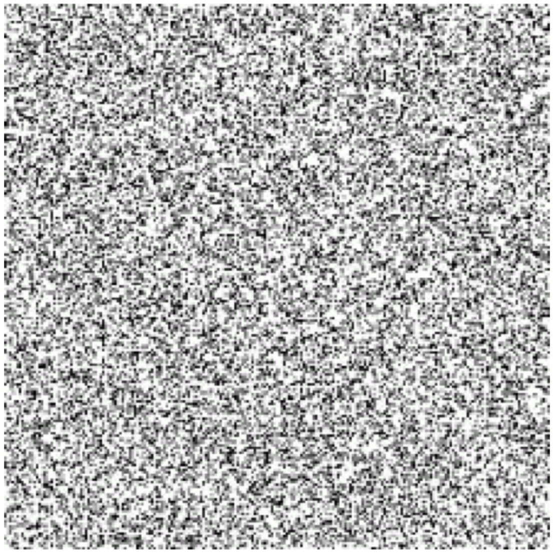
- Invented by Bela Julesz in the mid-20th century
- Demonstration that stereo perception can happen without any monocular cues



https://en.wikipedia.org/wiki/Random_dot_stereogram

History: Random dot stereograms

- Invented by Bela Julesz in the mid-20th century
- Demonstration that stereo perception can happen without any monocular cues



https://en.wikipedia.org/wiki/Random_dot_stereogram

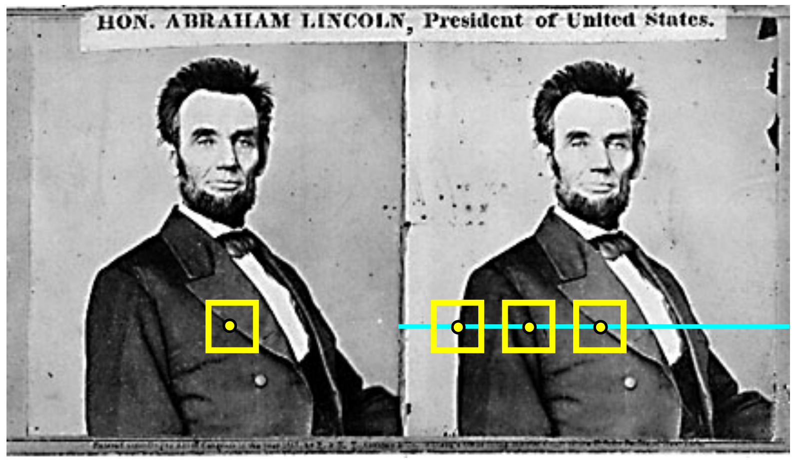
Outline

- Motivation and history
- Basic two-view stereo setup

Problem formulation

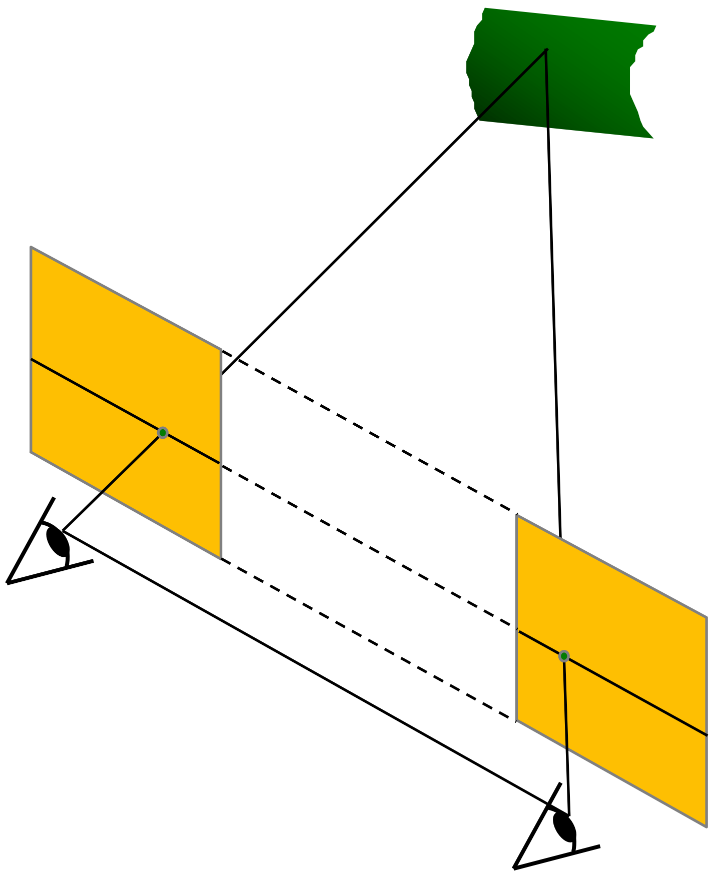
- **Given**: stereo pair (assumed calibrated)
- **Wanted**: dense depth map

Basic stereo matching algorithm



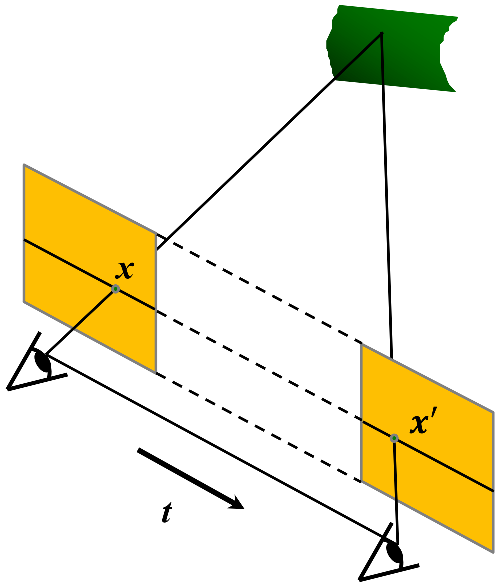
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match
  - Triangulate the matches to get depth information
- Simplest case: epipolar lines are corresponding scanlines
  - When does this happen?

Parallel images



- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at the same height
- Focal lengths are the same
- Then epipolar lines fall along horizontal scan lines of the images

Essential matrix for parallel images



Epipolar constraint:

$$\boldsymbol{x}'^T \boldsymbol{E}\boldsymbol{x} = 0, \qquad \boldsymbol{E} = [\boldsymbol{t}_\times]\boldsymbol{R}$$

$$\boldsymbol{R} = \boldsymbol{I} \quad \boldsymbol{t} = (t, 0, 0)$$

$$\boldsymbol{E} = [\boldsymbol{t}_\times]\boldsymbol{R} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t \\ 0 & t & 0 \end{bmatrix}$$

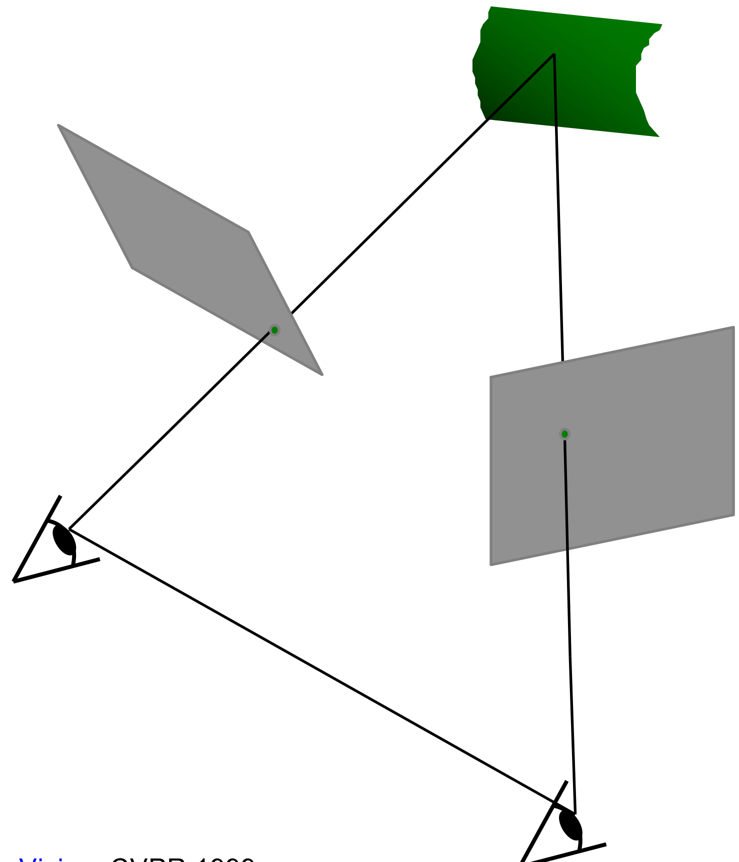
$$(u' \; v' \; 1) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t \\ 0 & t & 0 \end{bmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0 \qquad (u' \; v' \; 1) \begin{pmatrix} 0 \\ -t \\ tv \end{pmatrix} = 0 \qquad -tv + tv' = 0$$

$$v = v'$$

The y-coordinates of corresponding points are the same!

Stereo image rectification

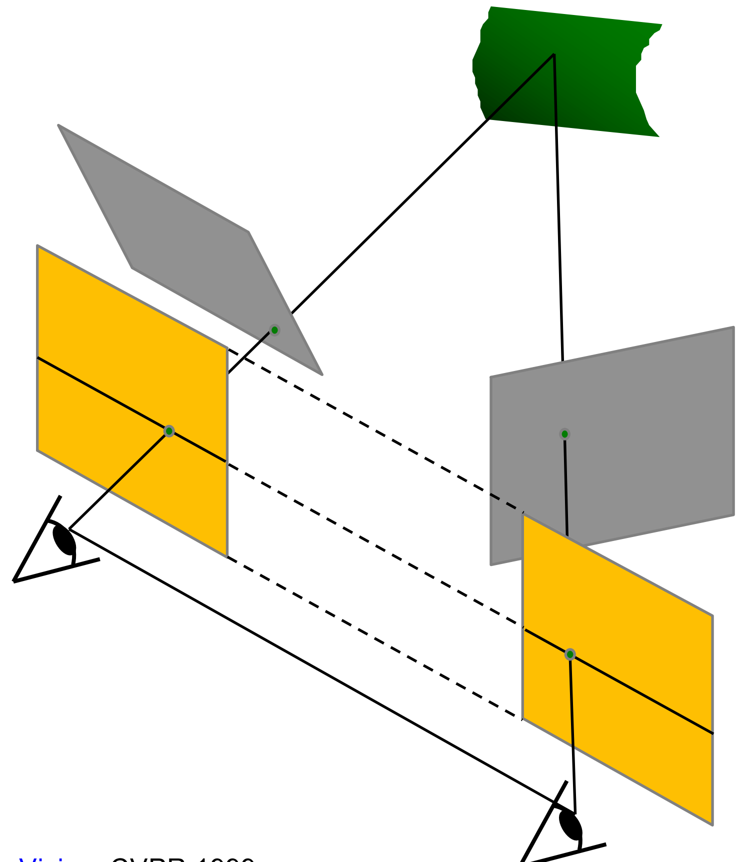
- If the image planes are not parallel, we can find homographies to project each view onto a common plane parallel to the baseline



C. Loop and Z. Zhang. Computing Rectifying Homographies for Stereo Vision. CVPR 1999

Stereo image rectification

- 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

- Before rectification:

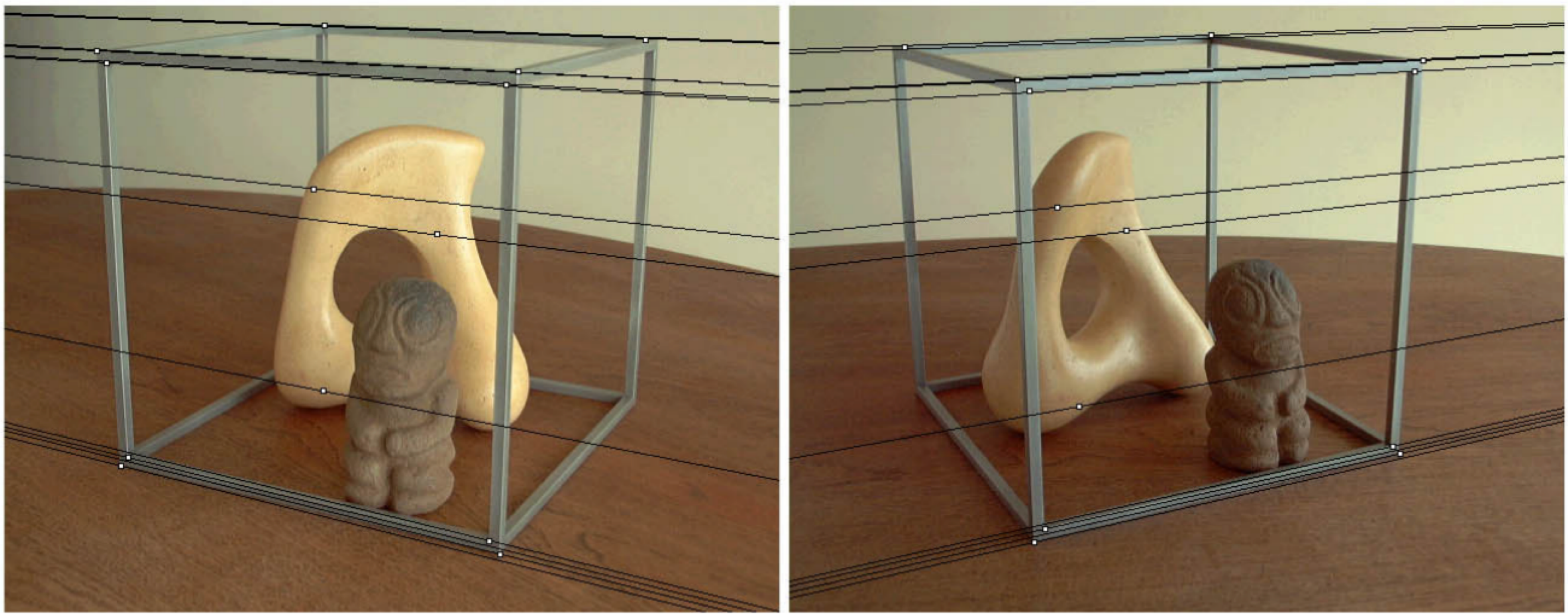


Image source

Stereo image rectification

- After rectification:

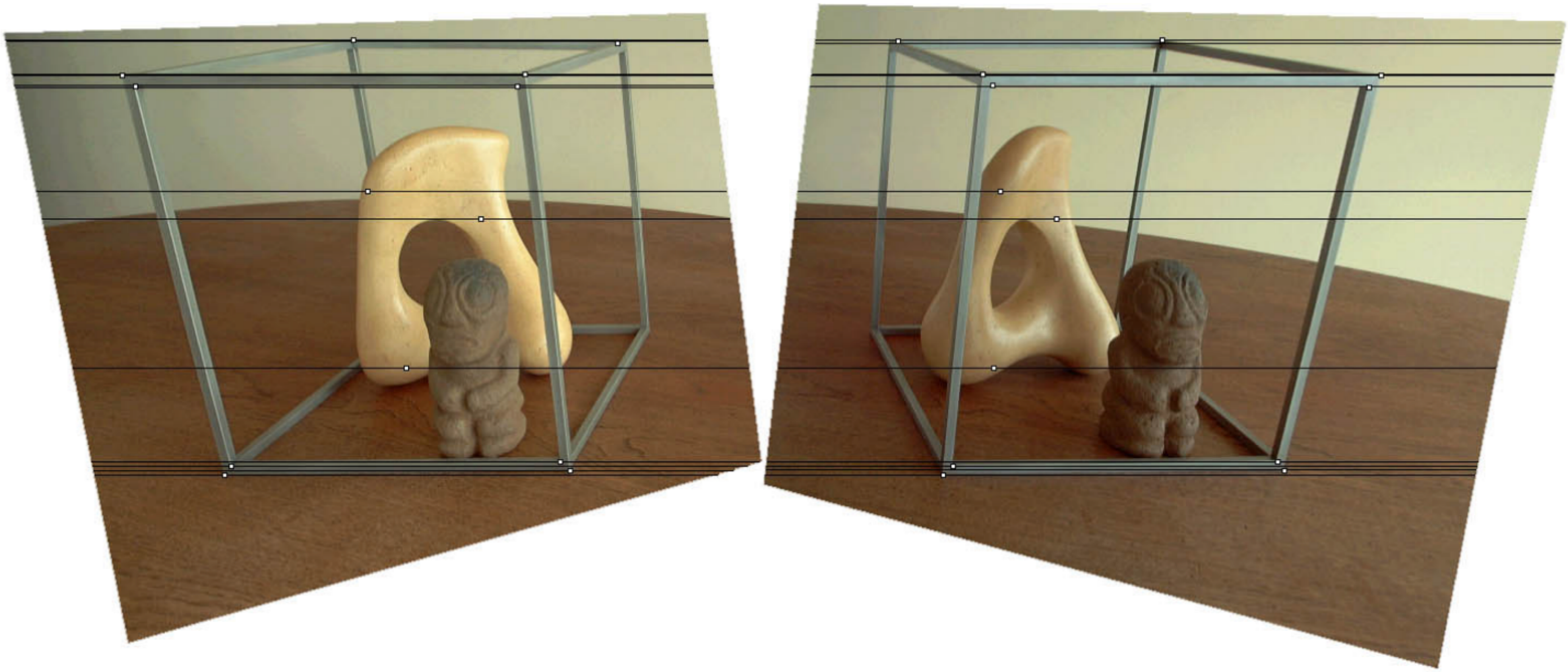
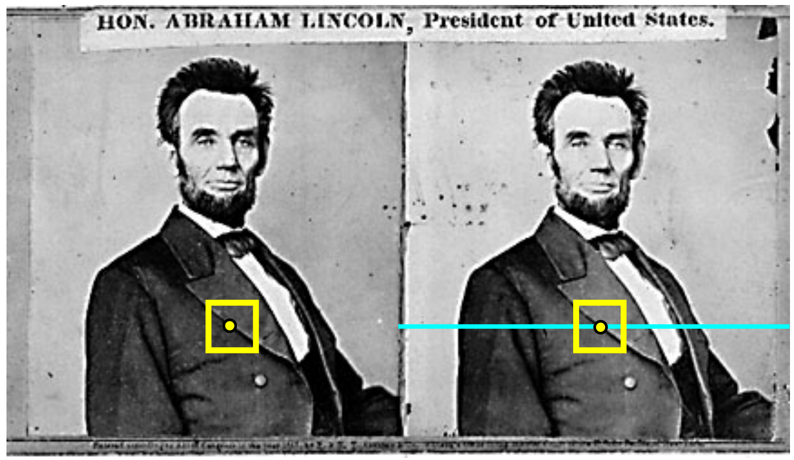


Image source

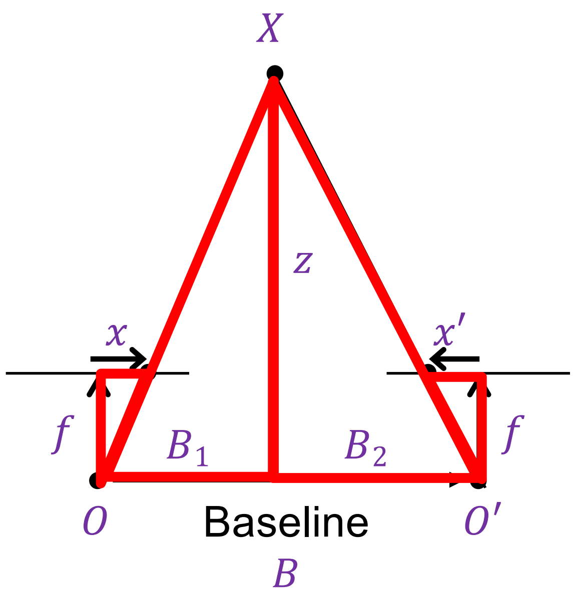
Another rectification example

Basic stereo matching algorithm



- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Triangulate the matches to get depth information

Depth from disparity



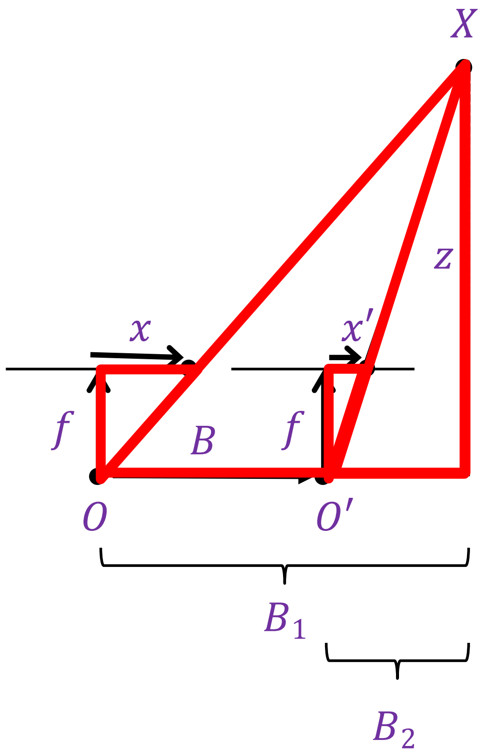
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{-x'}{f} = \frac{B_2}{z}$$

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

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

Disparity is inversely proportional to depth!

Depth from disparity



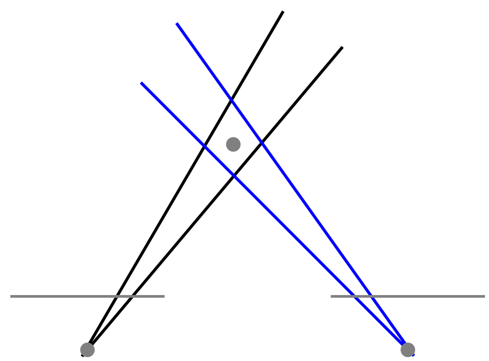
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{x'}{f} = \frac{B_2}{z}$$

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

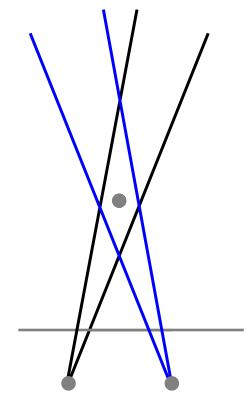
$$x - x' = \frac{fB}{z}$$

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

Effect of baseline on stereo results

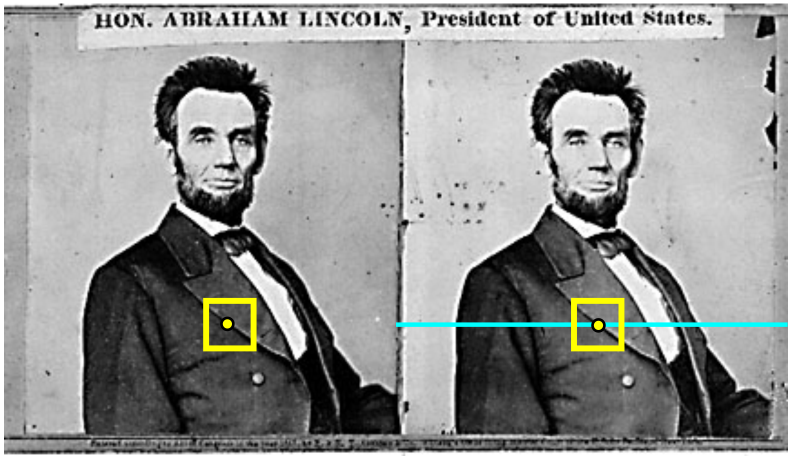


- Larger baseline
  - + Smaller triangulation error
  - – Matching is more difficult



- Smaller baseline
  - – Higher triangulation error
  - + Matching is easier

Basic stereo matching algorithm

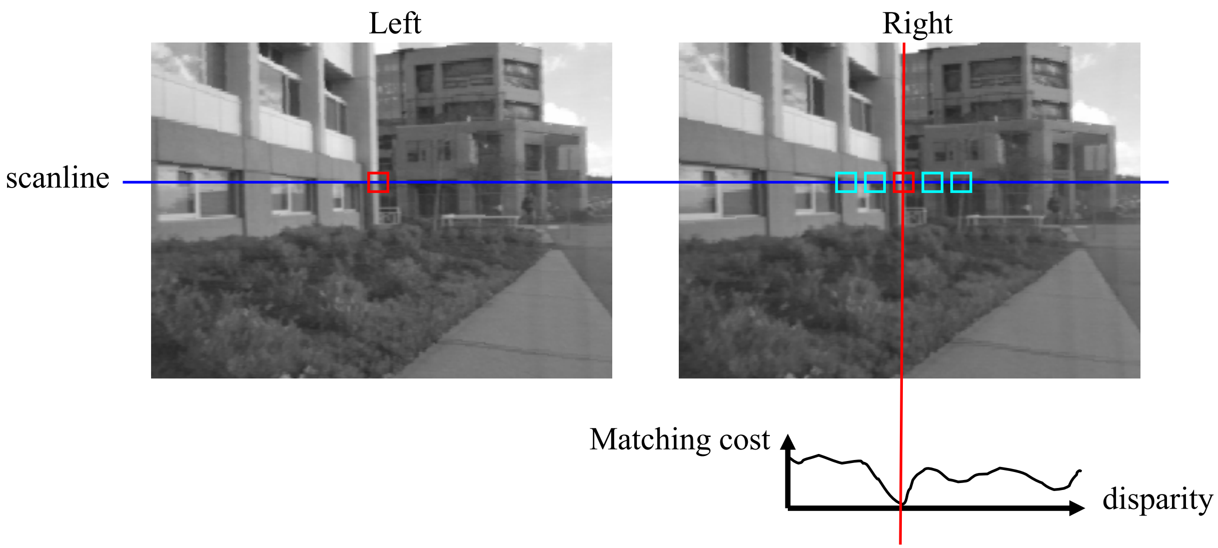


- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Compute disparity $x - x'$ and set $\mathrm{depth}(x) = Bf/(x - x')$

Outline

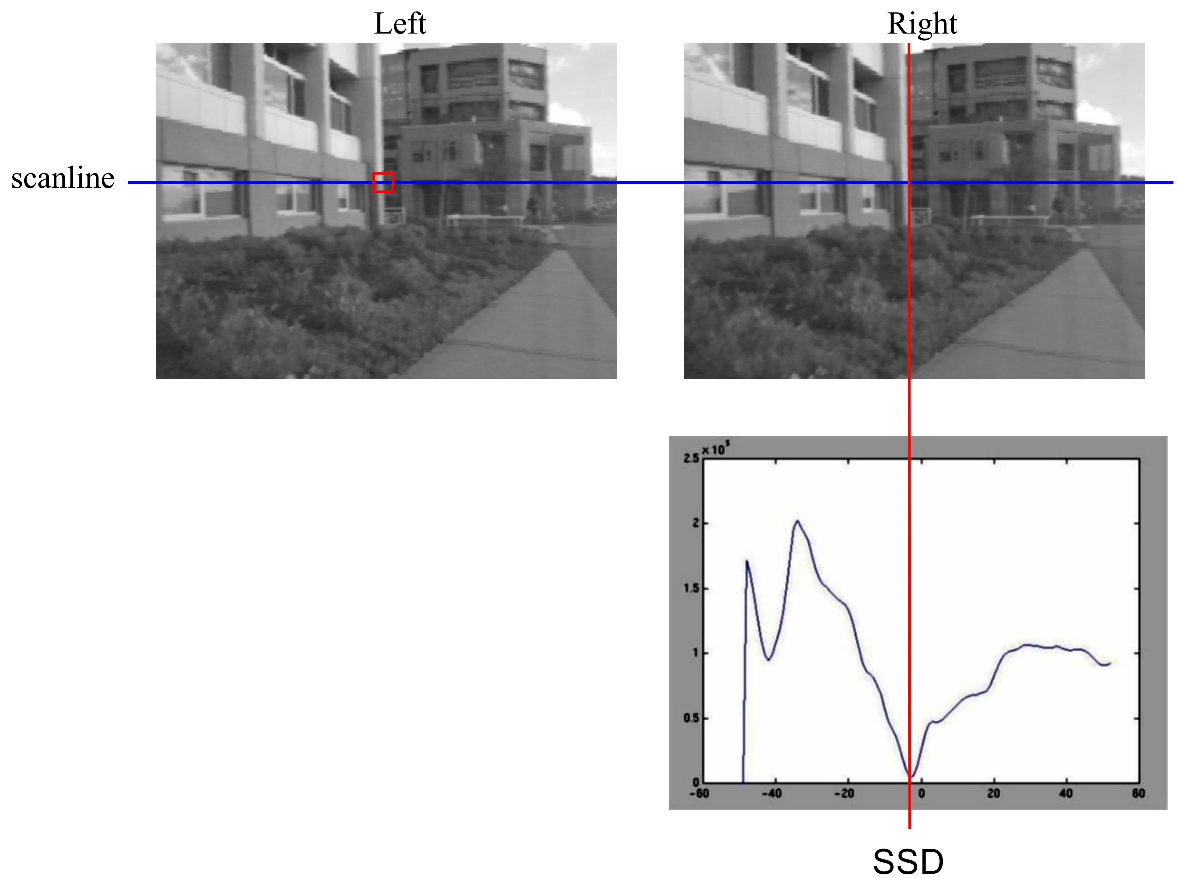
- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm

Correspondence search

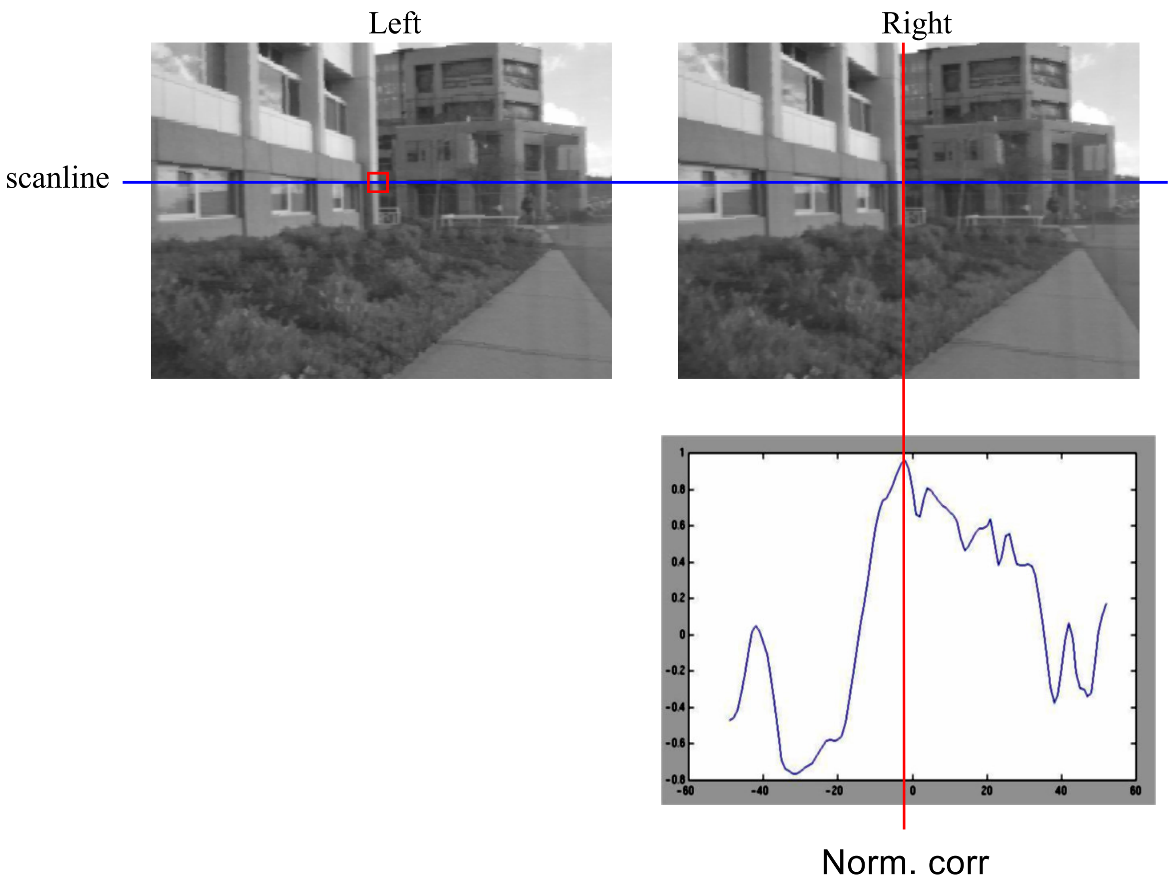


- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

Correspondence search

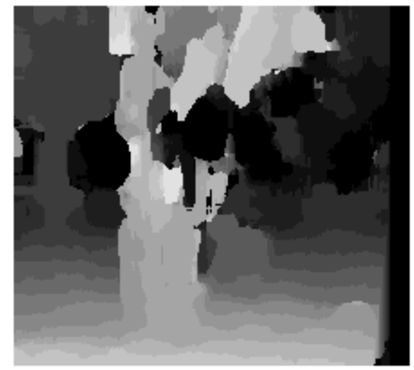


Correspondence search



Effect of window size on correspondence search

Window size 3

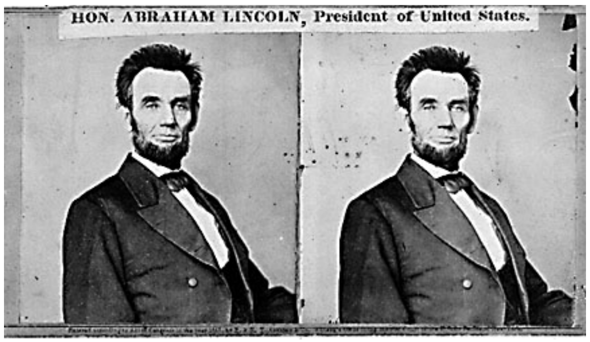


Window size 20

- Smaller window:
  + More detail
  – More noise

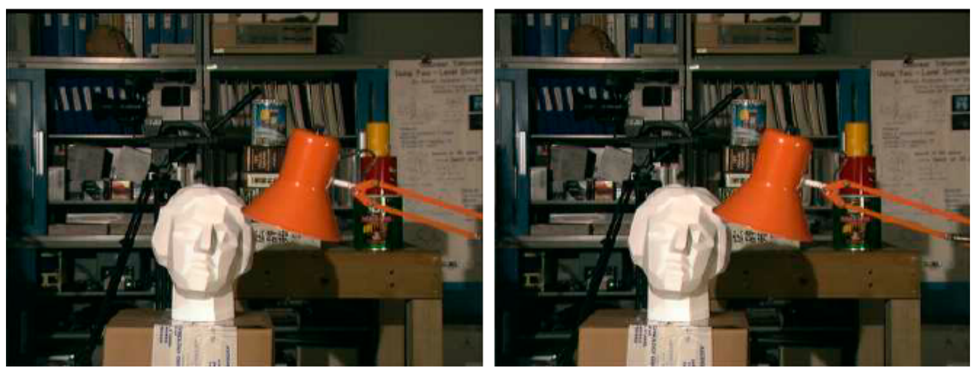
- Larger window:
  + Smoother disparity maps
  – Less detail

Where will basic window search fail?



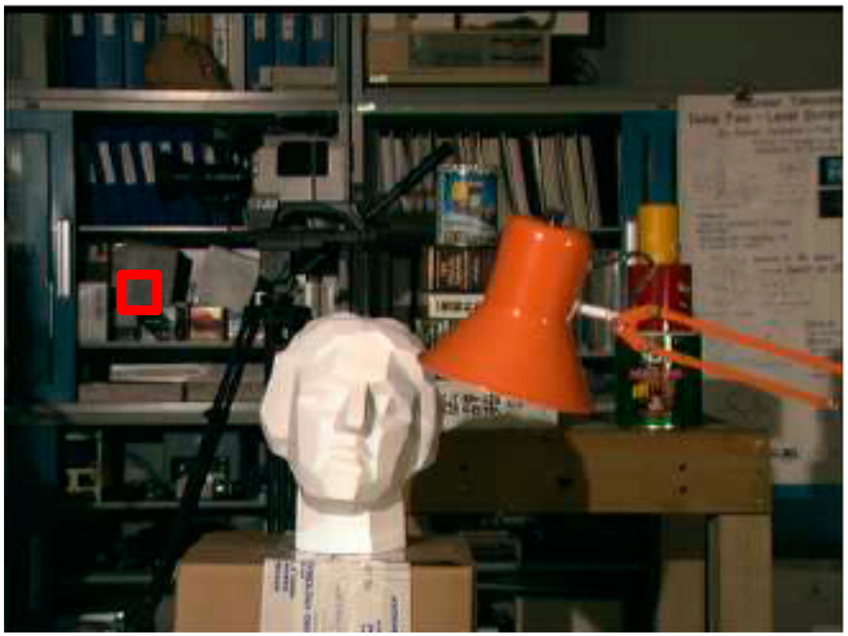
Textureless surfaces

Occlusions, repetition

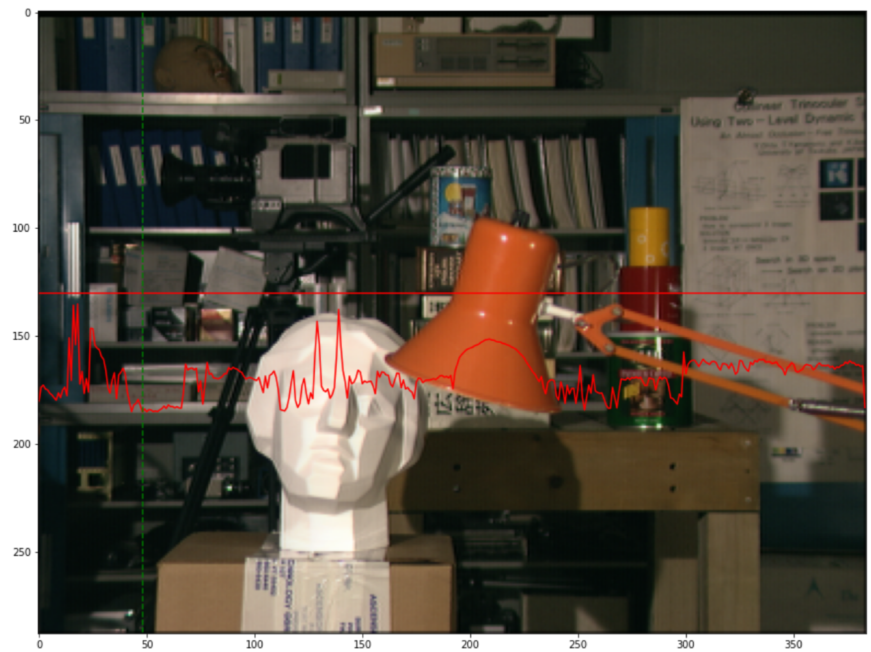


Non-Lambertian surfaces, specularities

Example: Textured neighborhood



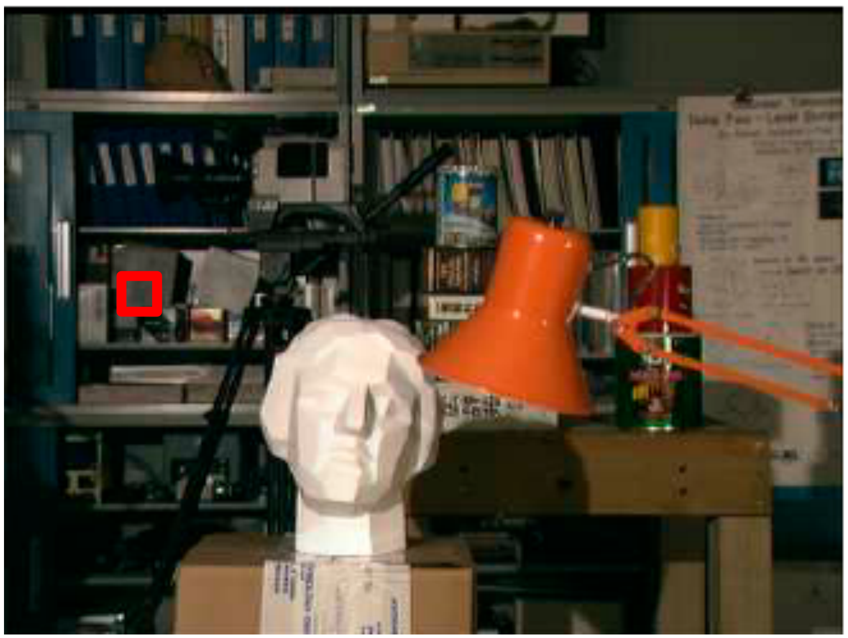
Window size: 1 pixel

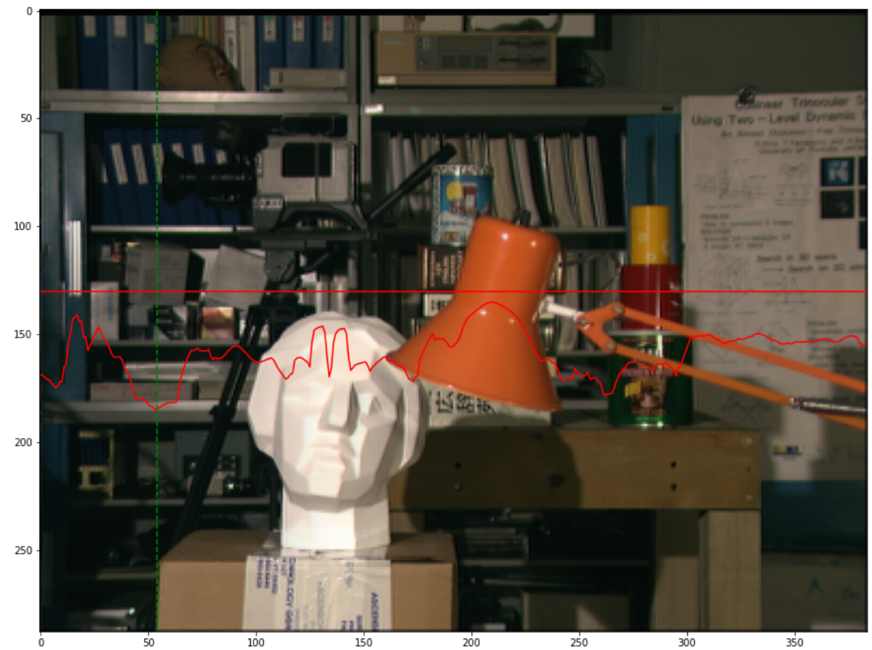


Source: [D. Hoiem]()

Example: Textured neighborhood

Window size: 7 pixels

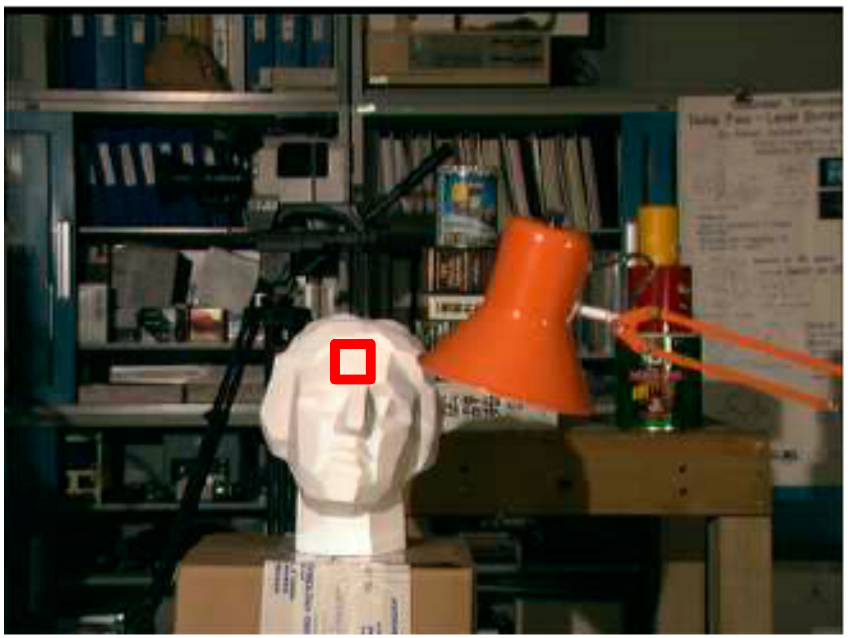


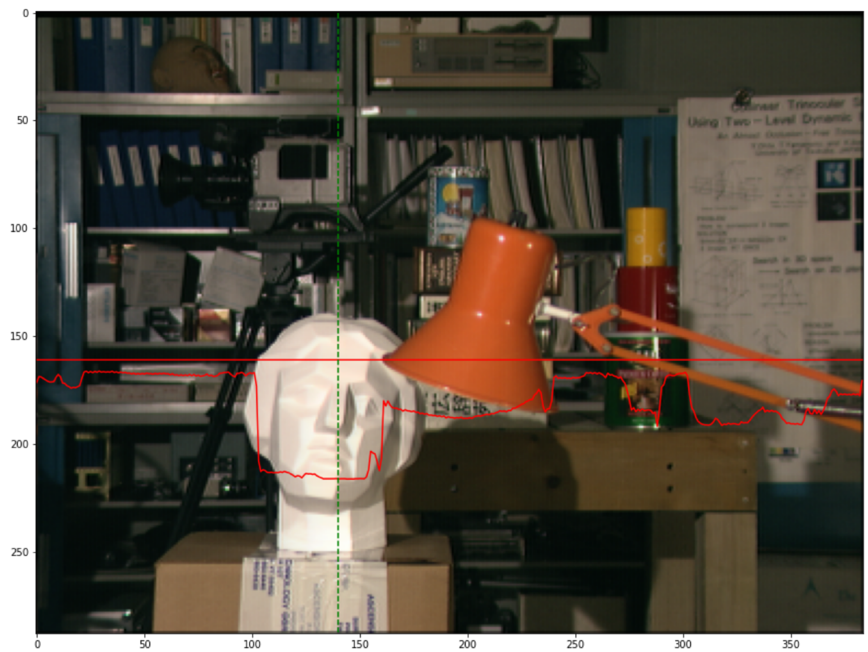


Source: [D. Hoiem]

Example: Smooth neighborhood

Window size: 1 pixel

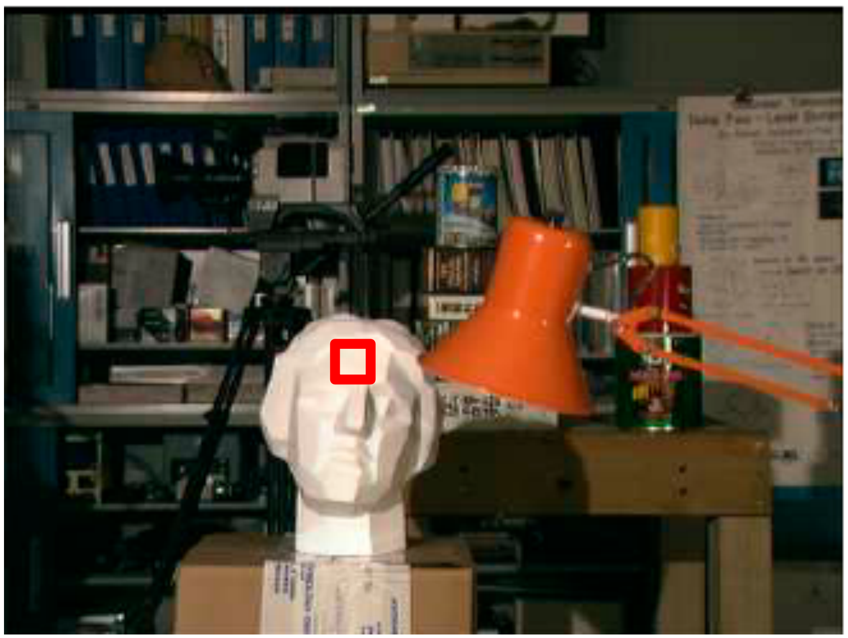


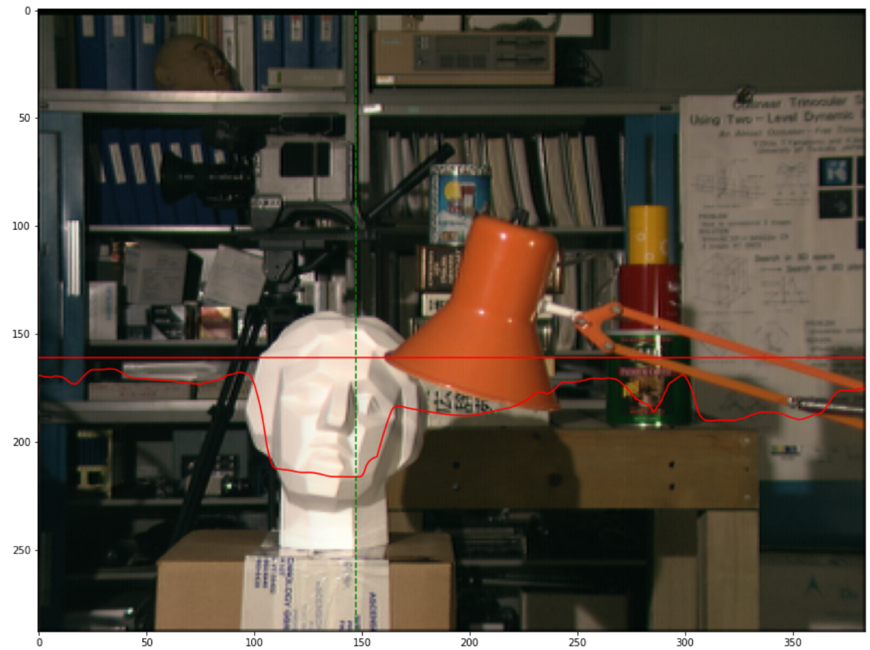


Source: [D. Hoiem]

Example: Smooth neighborhood

Window size: 7 pixels

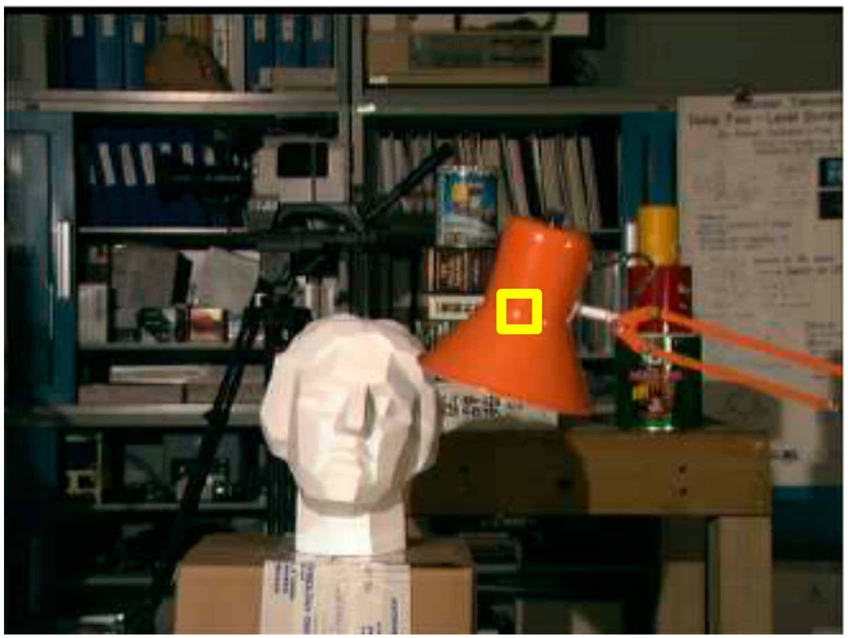




Source: [D. Hoiem]

Example: Specular highlight

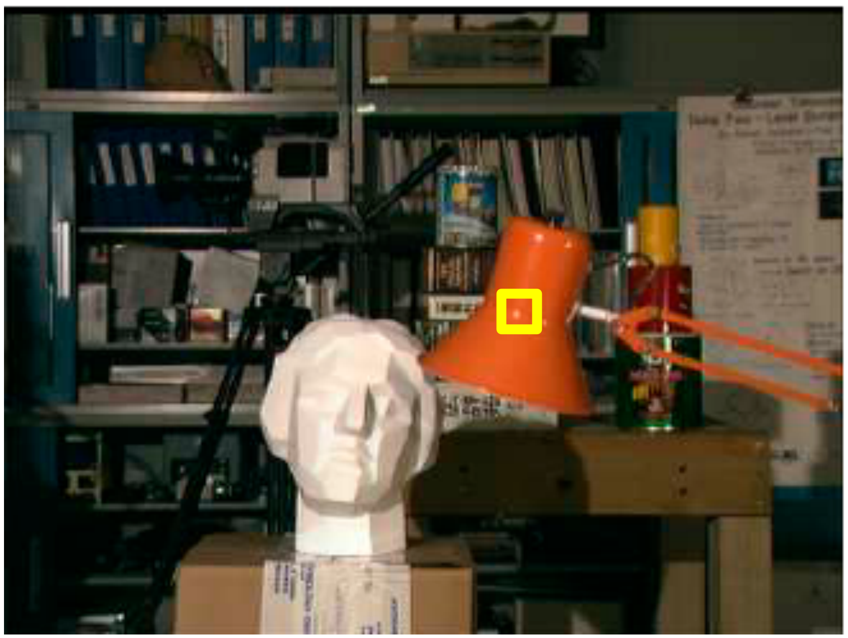
Window size: 1 pixel

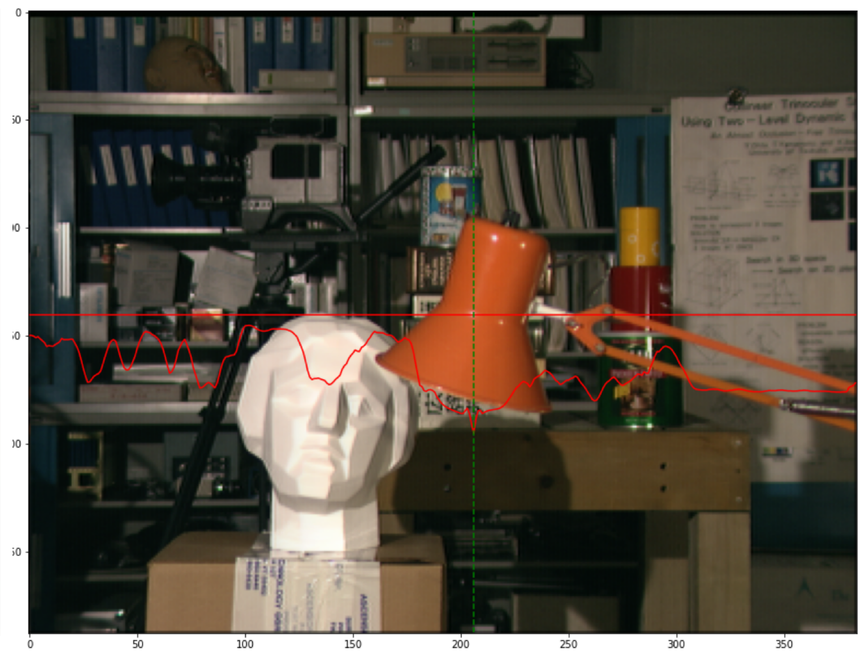


Source: [D. Hoiem]

Example: Specular highlight

Window size: 7 pixels





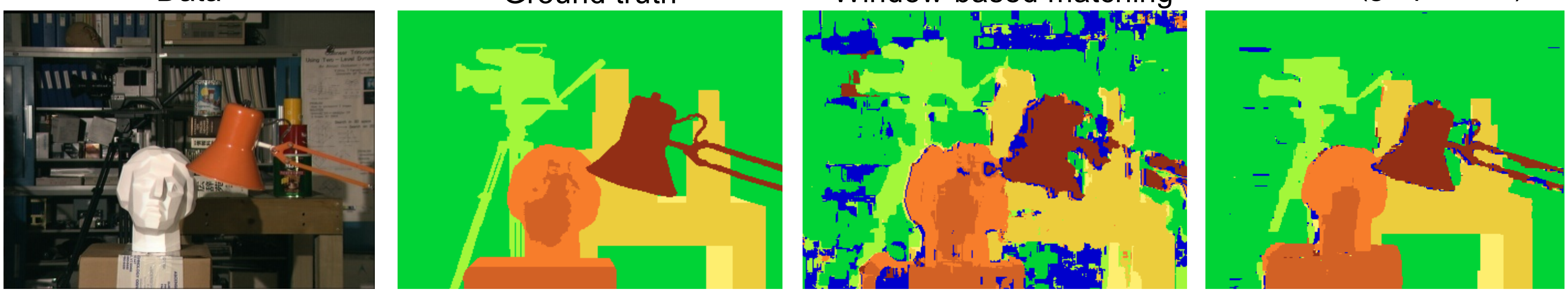
Source: [D. Hoiem]

Outline

- Motivation and history
- Basic two-view stereo setup
- Local stereo matching algorithm
- Beyond local stereo matching

Stereo as optimization with non-local constraints

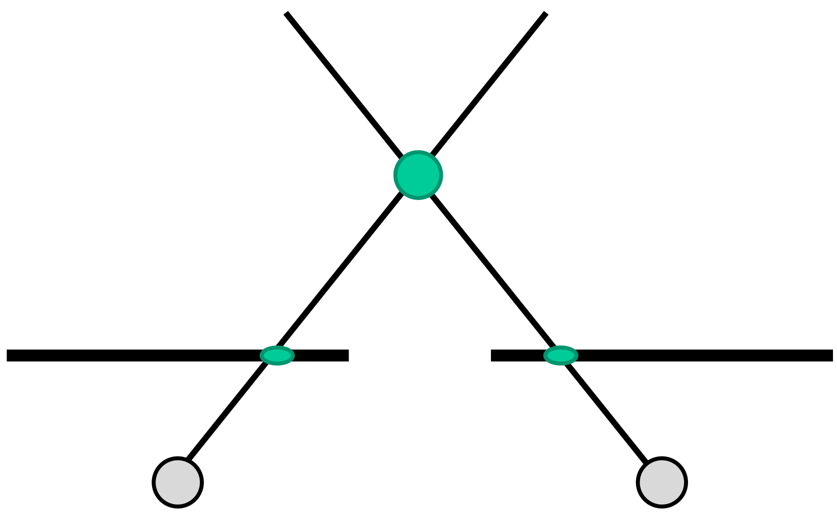
Data | Ground truth | Window-based matching | Global optimization method (graph cuts)



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

Non-local constraint: Uniqueness

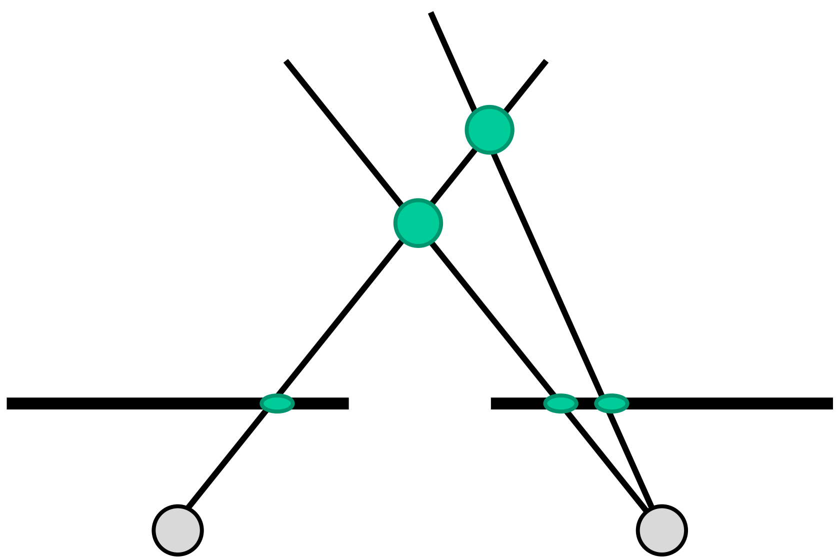
- Each point in one image should match at most one point in the other image
- Does uniqueness always hold in real life?



Source: J. Johnson and D. Fouhey

Non-local constraint: Uniqueness

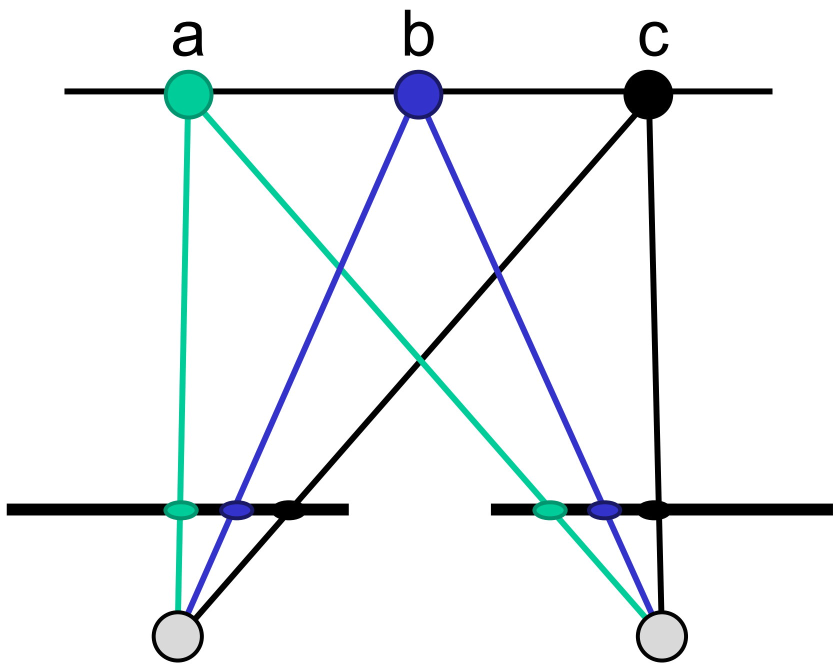
- Each point in one image should match at most one point in the other image
- Does uniqueness always hold in real life?



Source: J. Johnson and D. Fouhey

Non-local constraint: Ordering

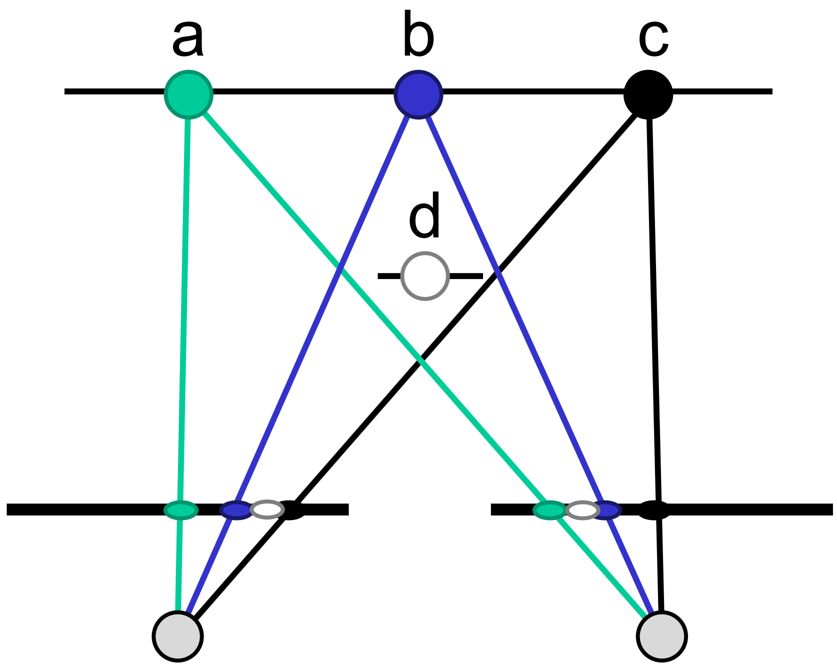
- Corresponding points should appear in the same order
- Is ordering always preserved in real life?



Source: J. Johnson and D. Fouhey

Non-local constraint: Ordering

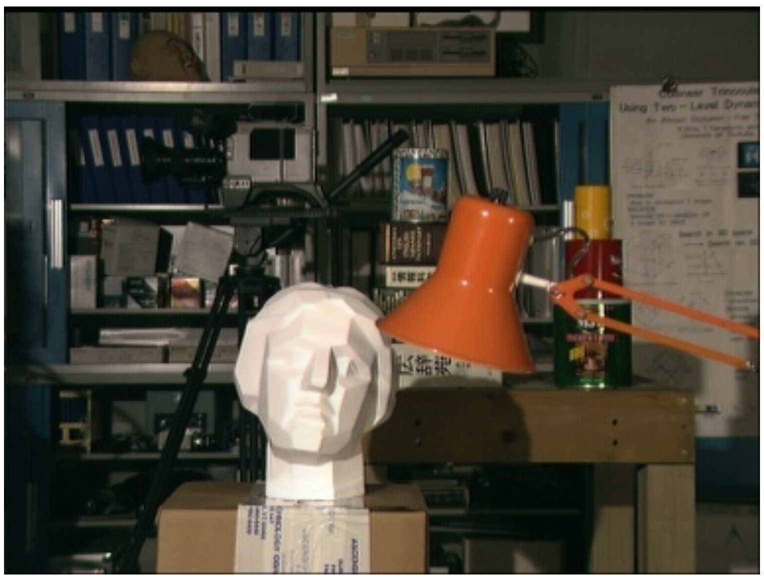
- Corresponding points should appear in the same order
- Is ordering always preserved in real life?



Source: [J. Johnson and D. Fouhey]

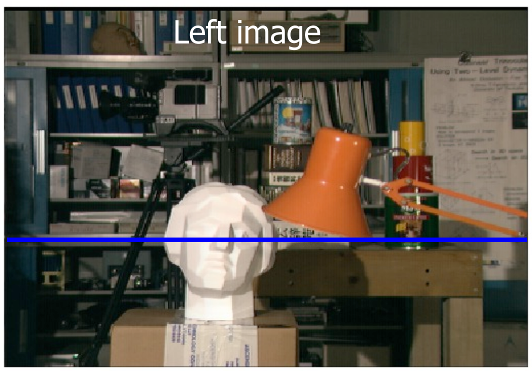
Non-local constraint: Smoothness

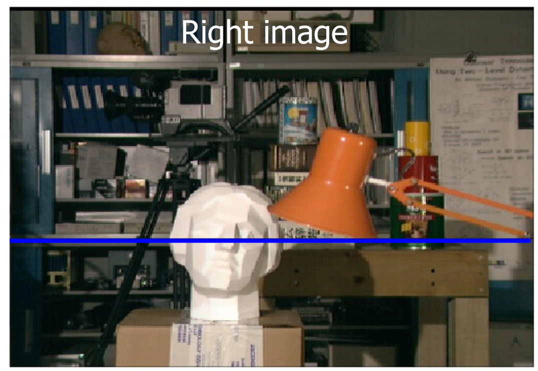
- We expect disparity values to change slowly (for the most part)



Scanline stereo by dynamic programming

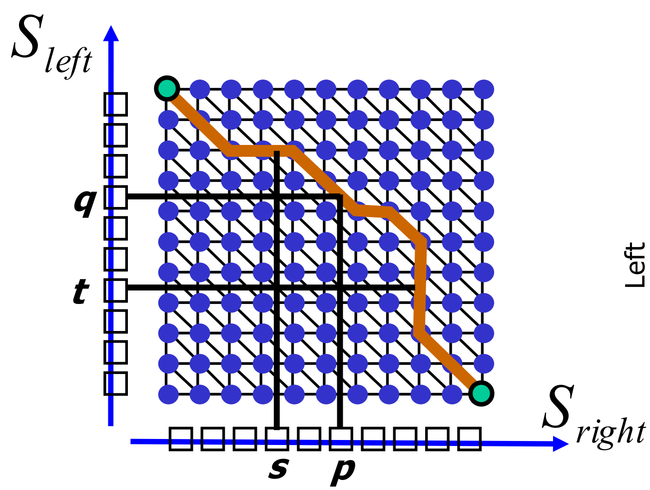
- Match pixels along the entire scanline while preserving uniqueness and ordering
- Different scanlines are still optimized independently

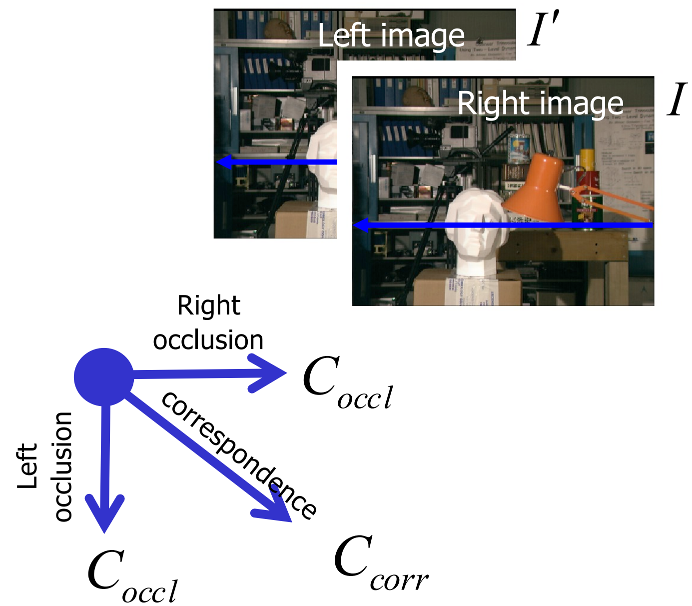




Y. Ohta and T. Kanade. Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming. IEEE Trans. PAMI, 1985

Scanline stereo by dynamic programming



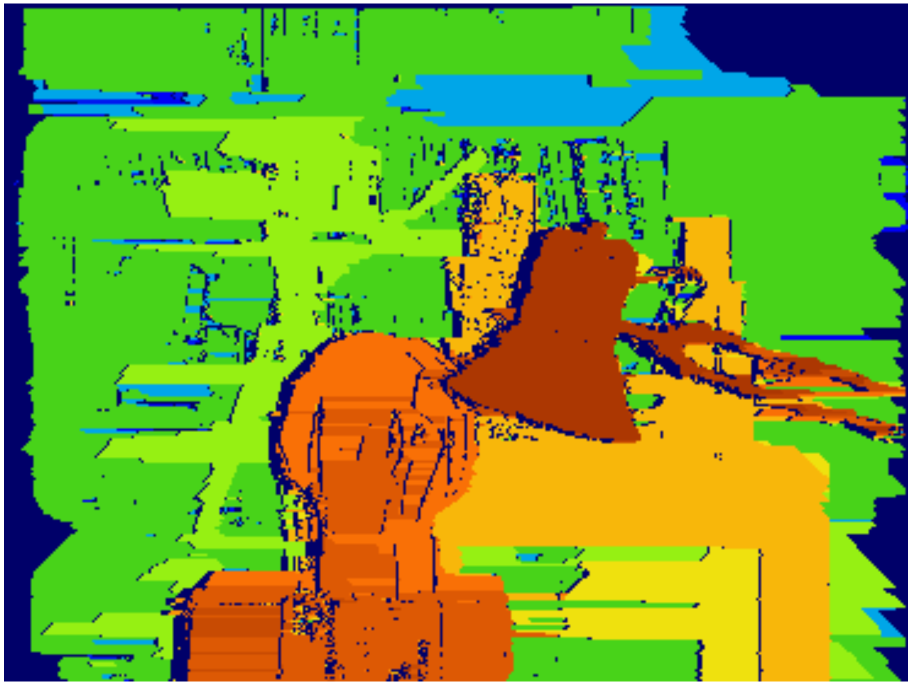


Source: Y. Boykov

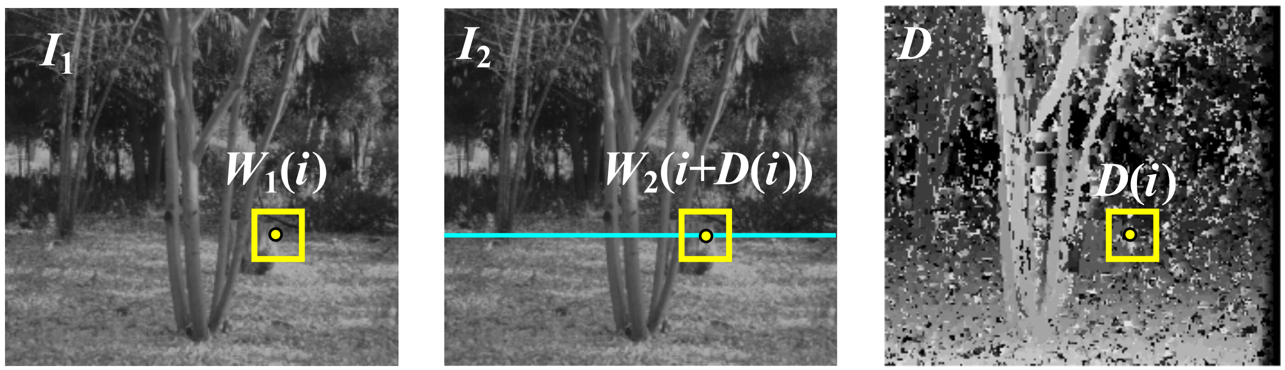
Y. Ohta and T. Kanade. [Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming](). IEEE Trans. PAMI, 1985

Scanline stereo by dynamic programming

- Generates streaking artifacts!



Stereo matching as global optimization



$$E(D) = \sum_i \left(W_1(i) - W_2(i + D(i))\right)^2 + \lambda \sum_{(i,j)\in\mathcal{N}} \rho(D(i) - D(j))$$

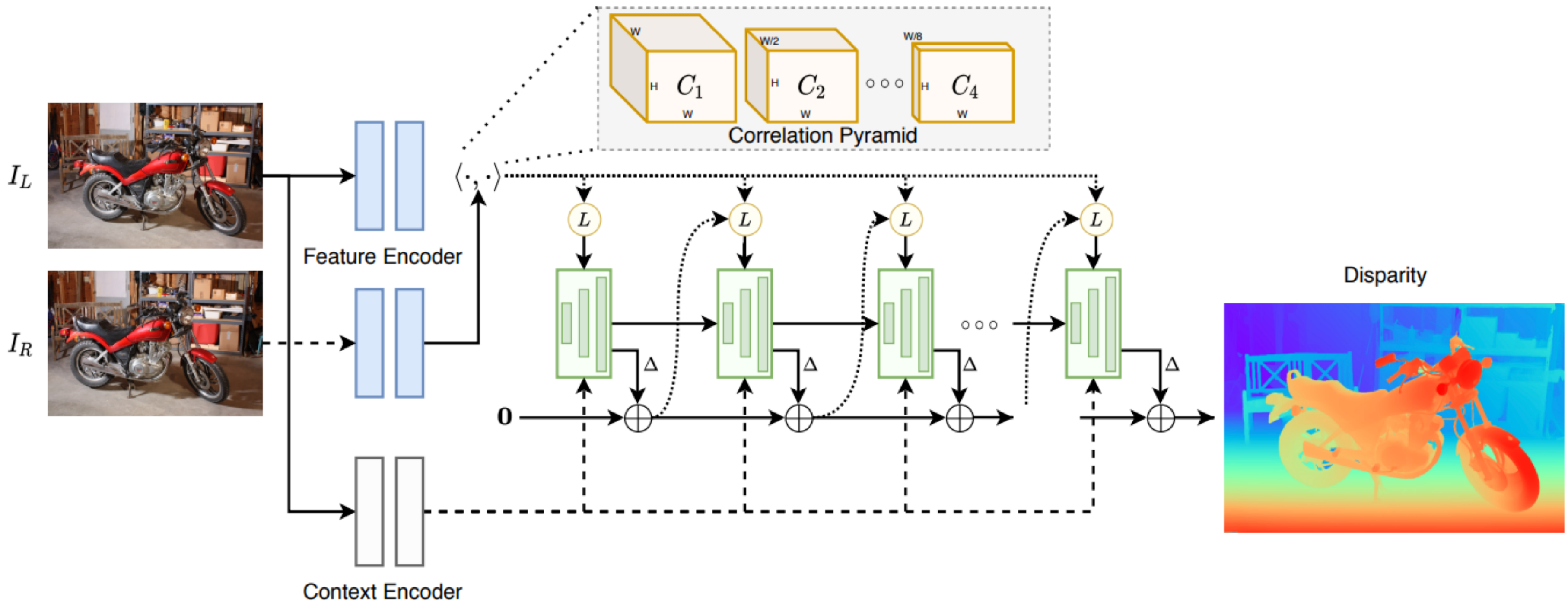
Data term

Neighborhood smoothness term

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

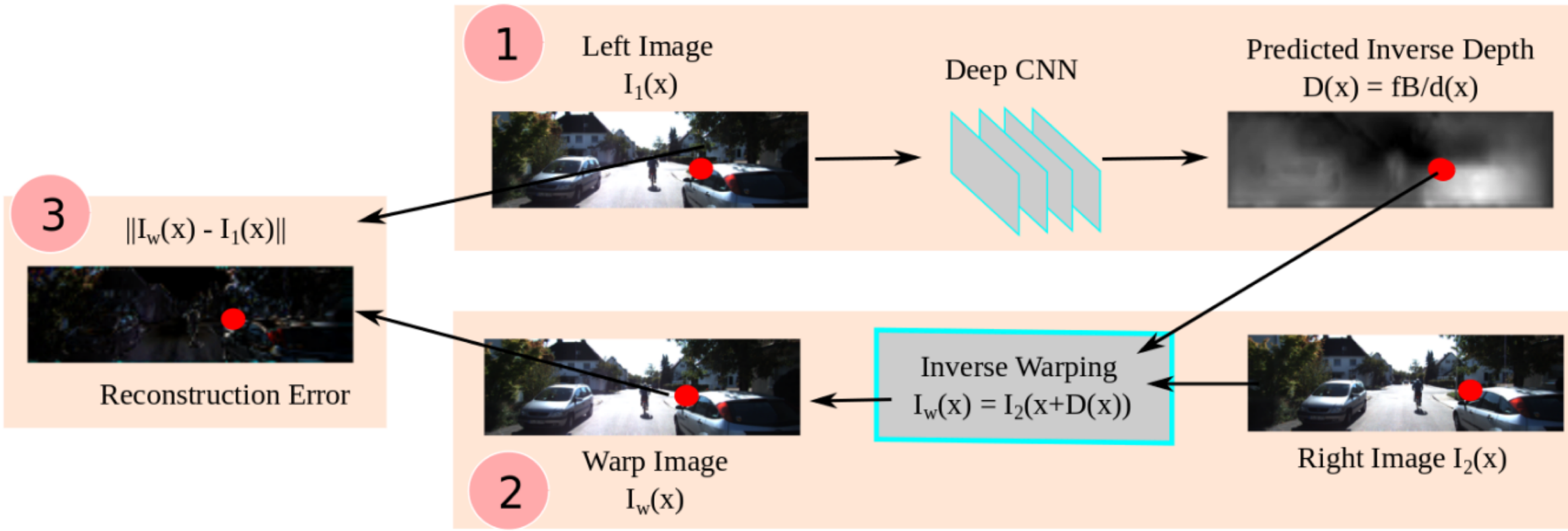
Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

Stereo matching with deep networks



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

Self-supervised depth estimation



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

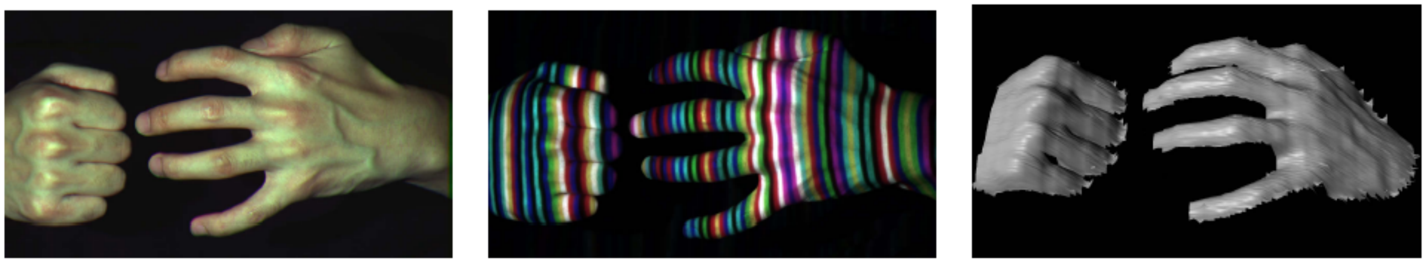
Stereo datasets

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

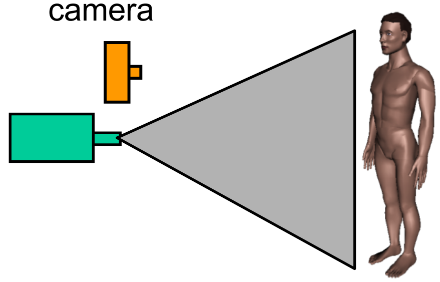
Outline

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

Active stereo with structured light

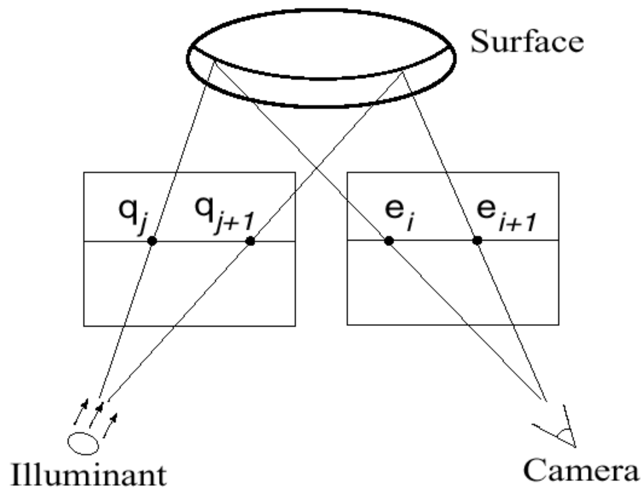


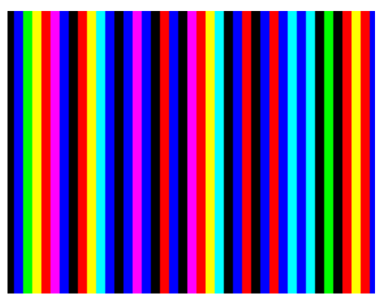
- Project "structured" light patterns onto the object
  - Simplifies the correspondence problem
  - Allows us to use only one camera

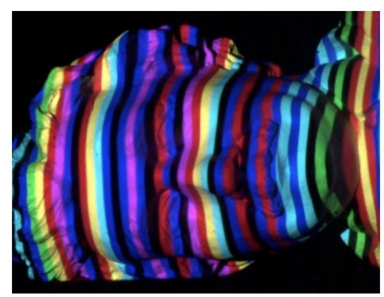


L. Zhang, B. Curless, and S. M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002

Active stereo with structured light

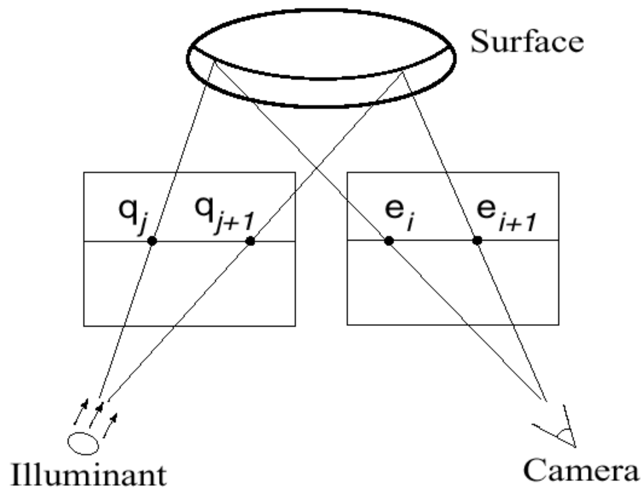


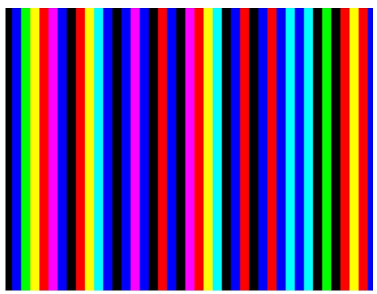


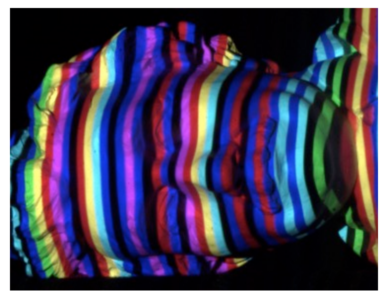


L. Zhang, B. Curless, and S. M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002

Active stereo with structured light





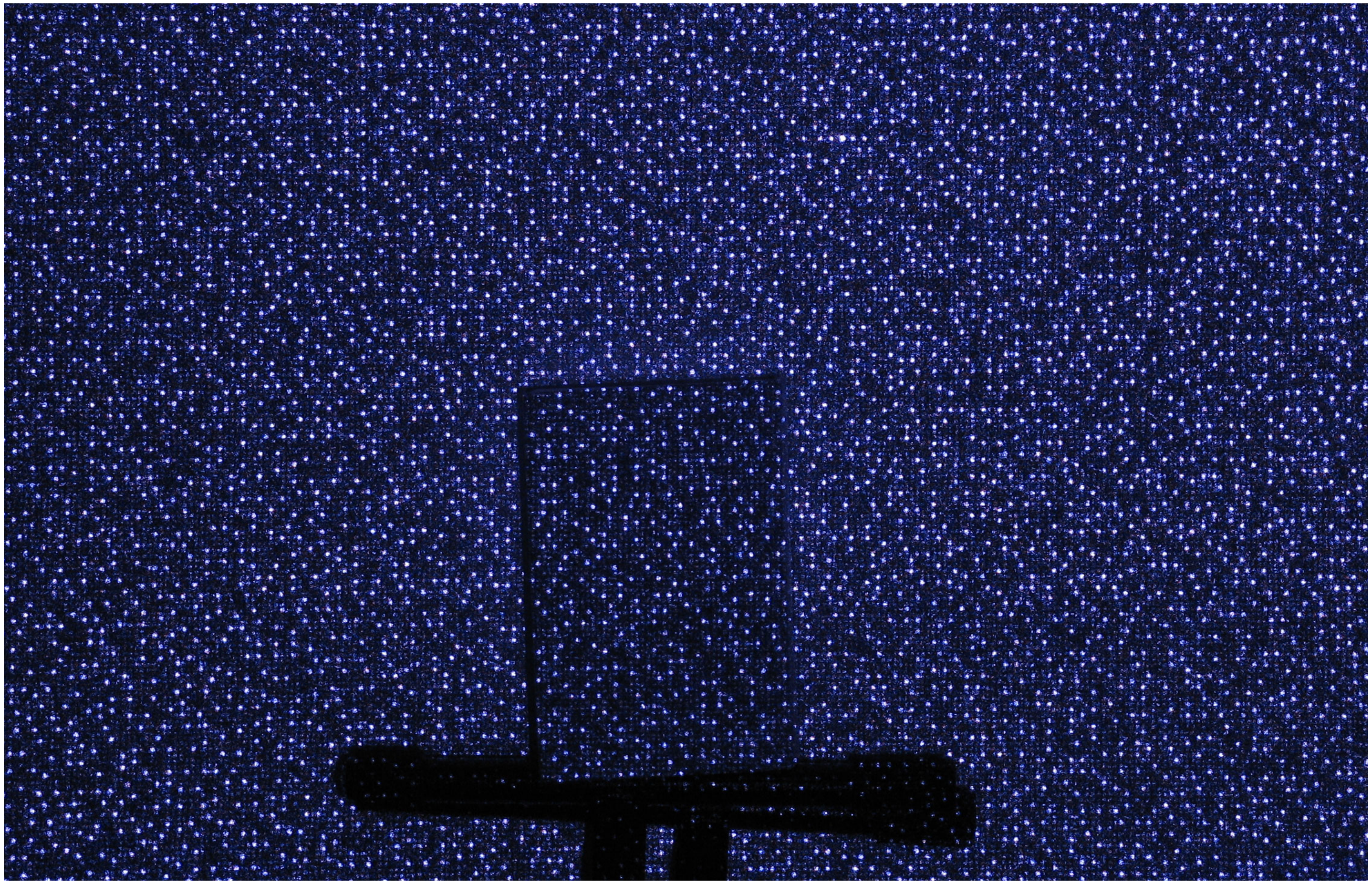


http://en.wikipedia.org/wiki/Structured-light_3D_scanner

Kinect: Structured infrared light

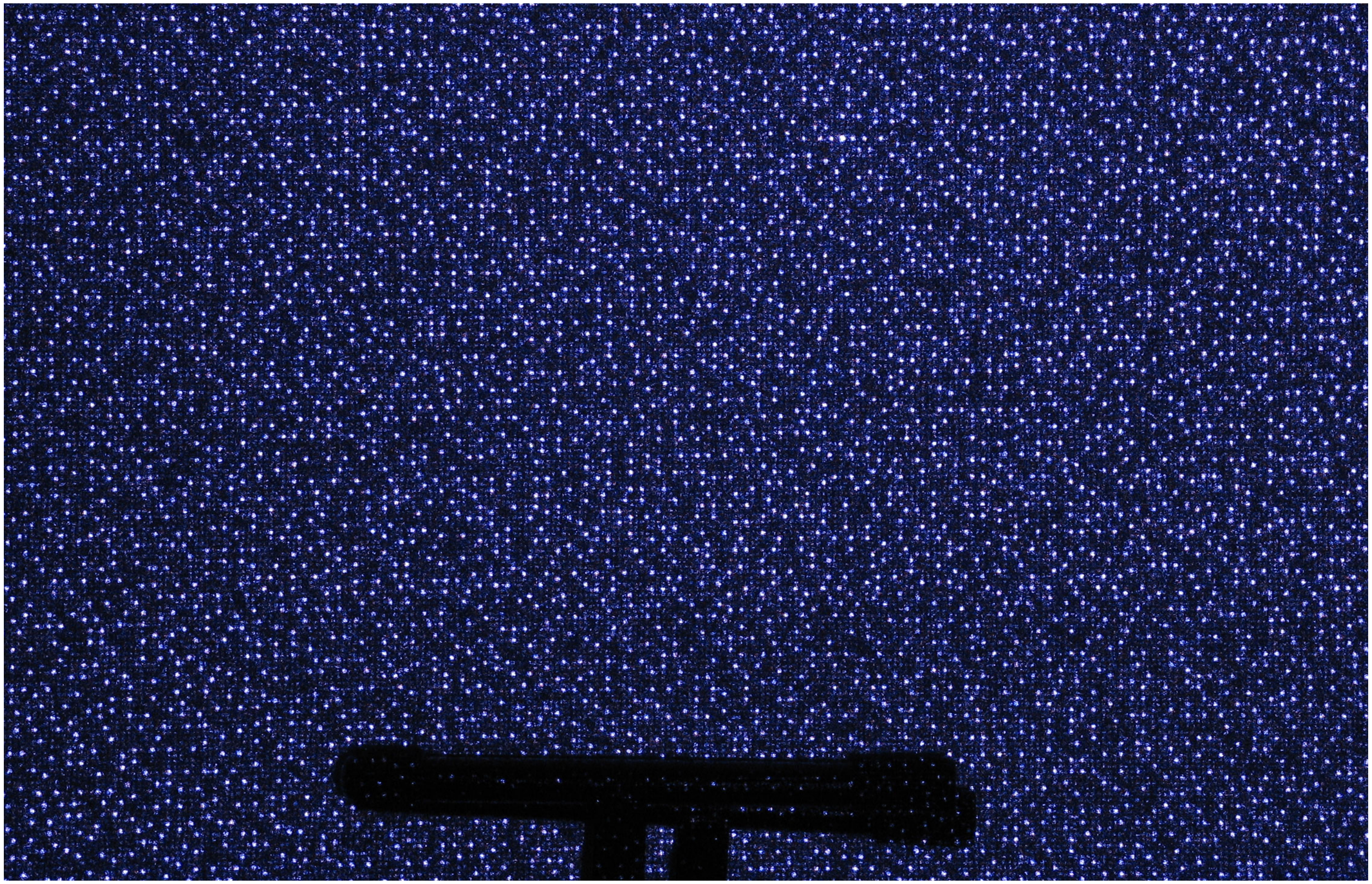
http://bbzippo.wordpress.com/2010/11/28/kinect-in-infrared/

Example: Book vs. No Book



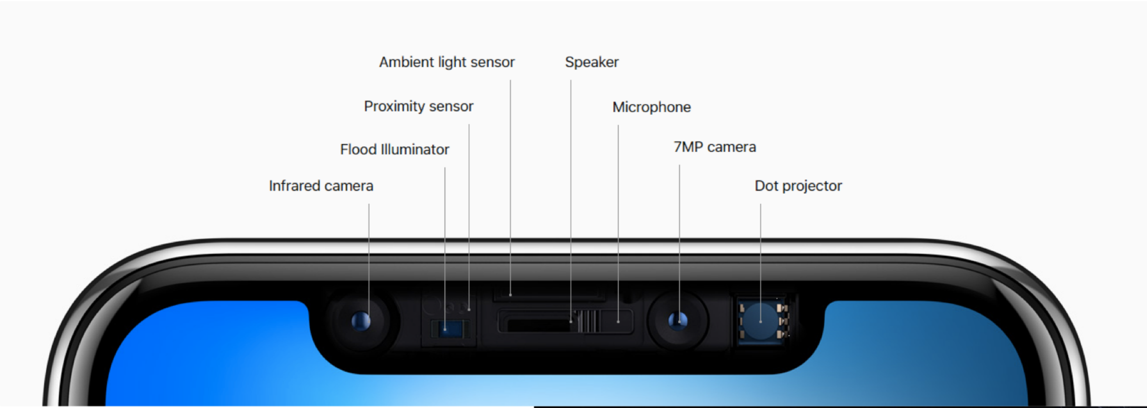
Source (via D. Hoiem)

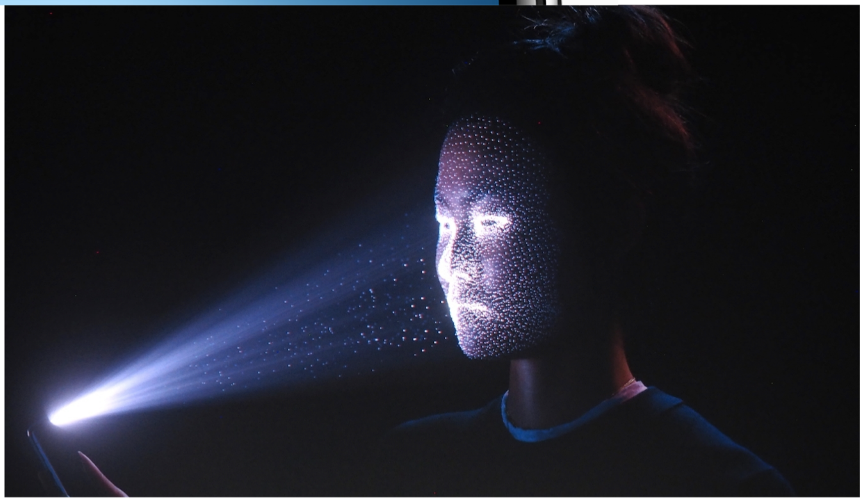
Example: Book vs. No Book



Source (via D. Hoiem)

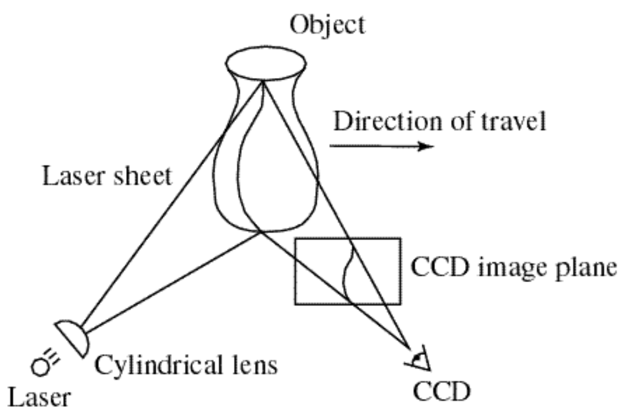
Apple TrueDepth

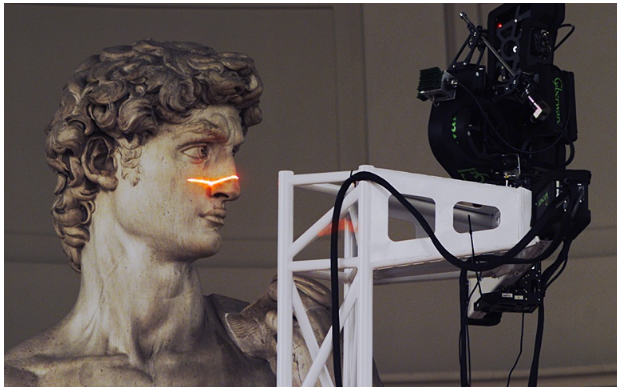




https://www.cnet.com/news/apple-face-id-truedepth-how-it-works/

Laser scanning





Digital Michelangelo Project
Levoy et al.
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

Source: S. Seitz

Laser scanned models

The Digital Michelangelo Project, Levoy et al.

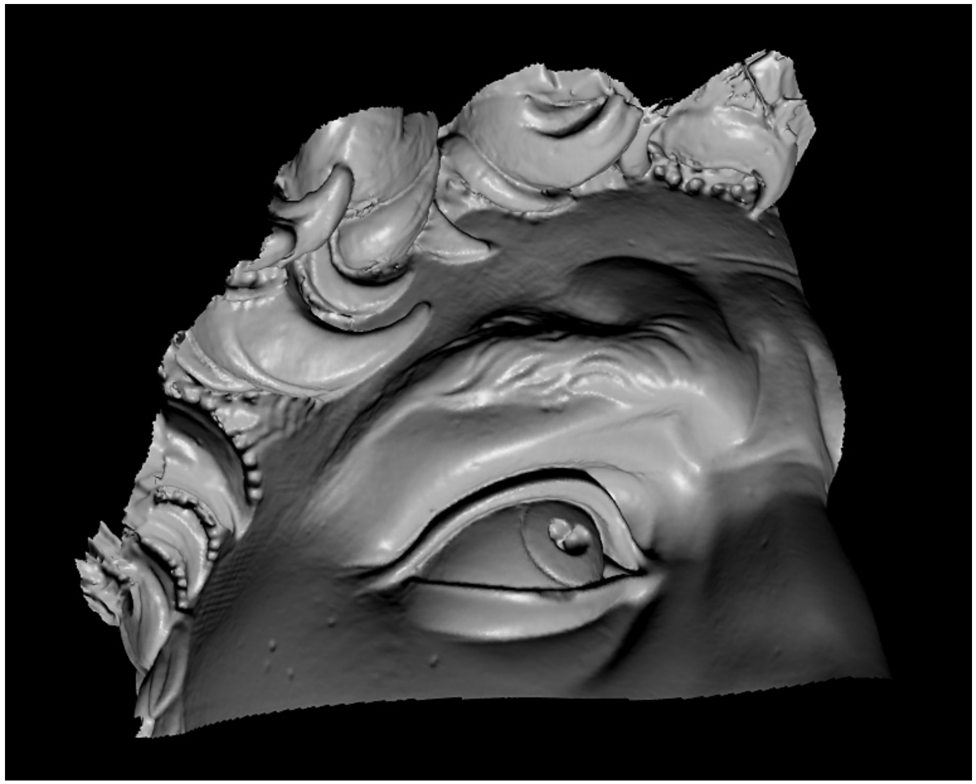
Source: S. Seitz

Laser scanned models

The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

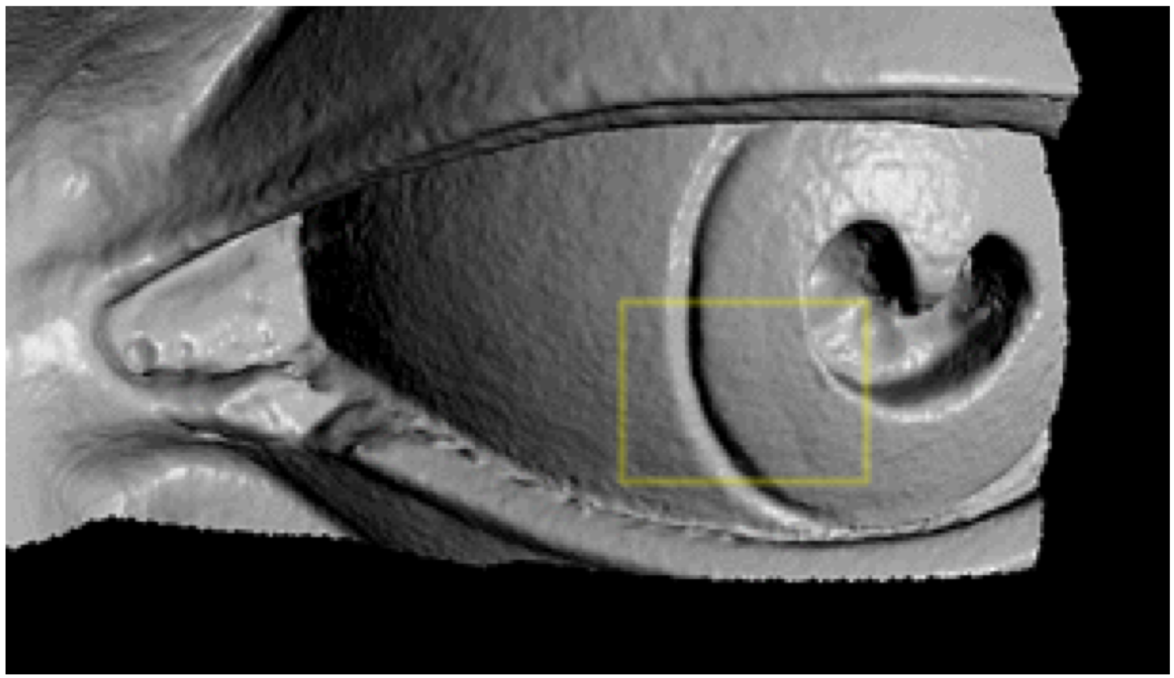
Laser scanned models



The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Laser scanned models

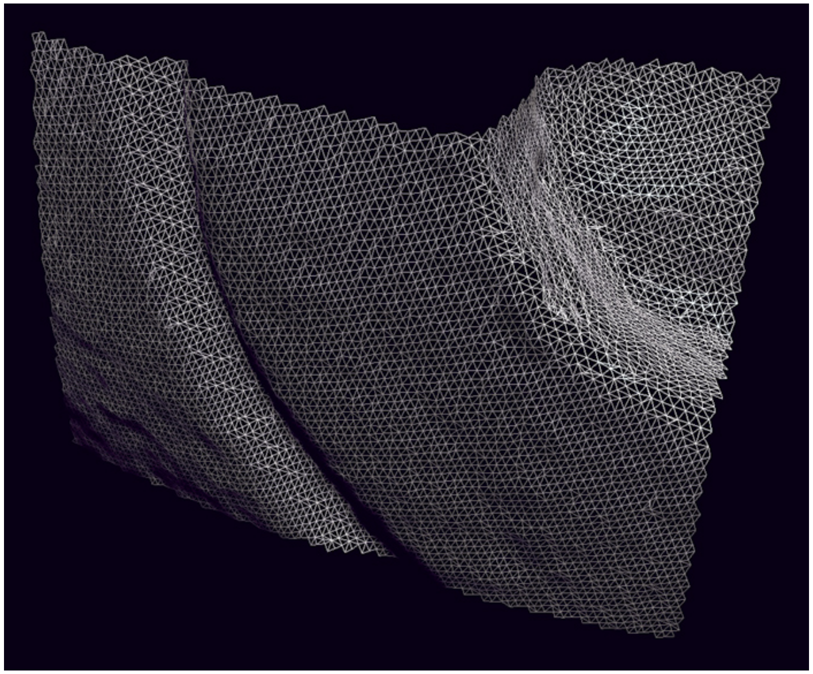


The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Laser scanned models

1.0 mm resolution (56 million triangles)

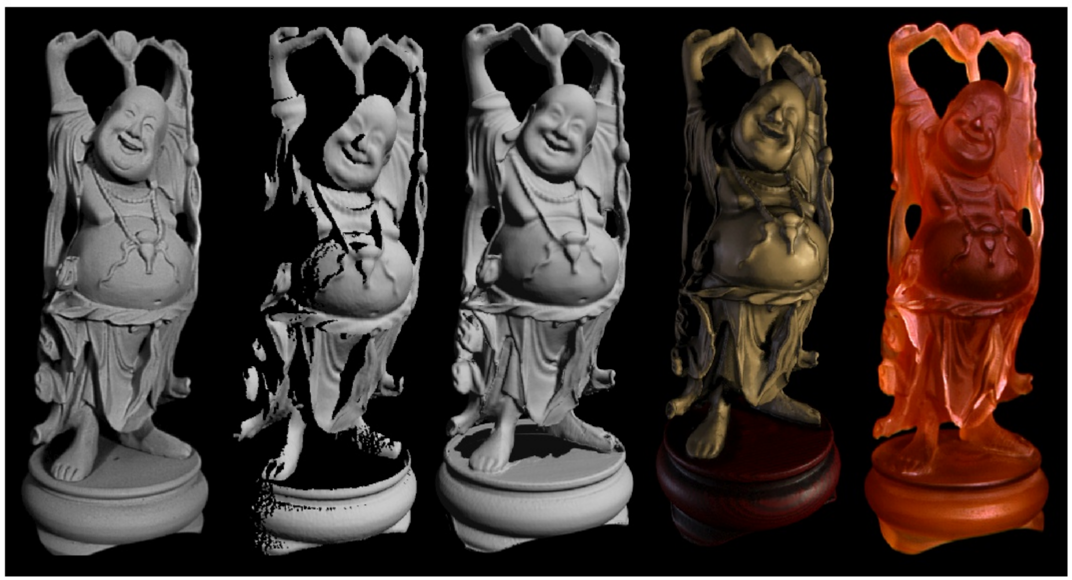


The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Aligning range images

- A single range scan is not sufficient to capture a complex surface
- Need techniques to register multiple range images



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

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*