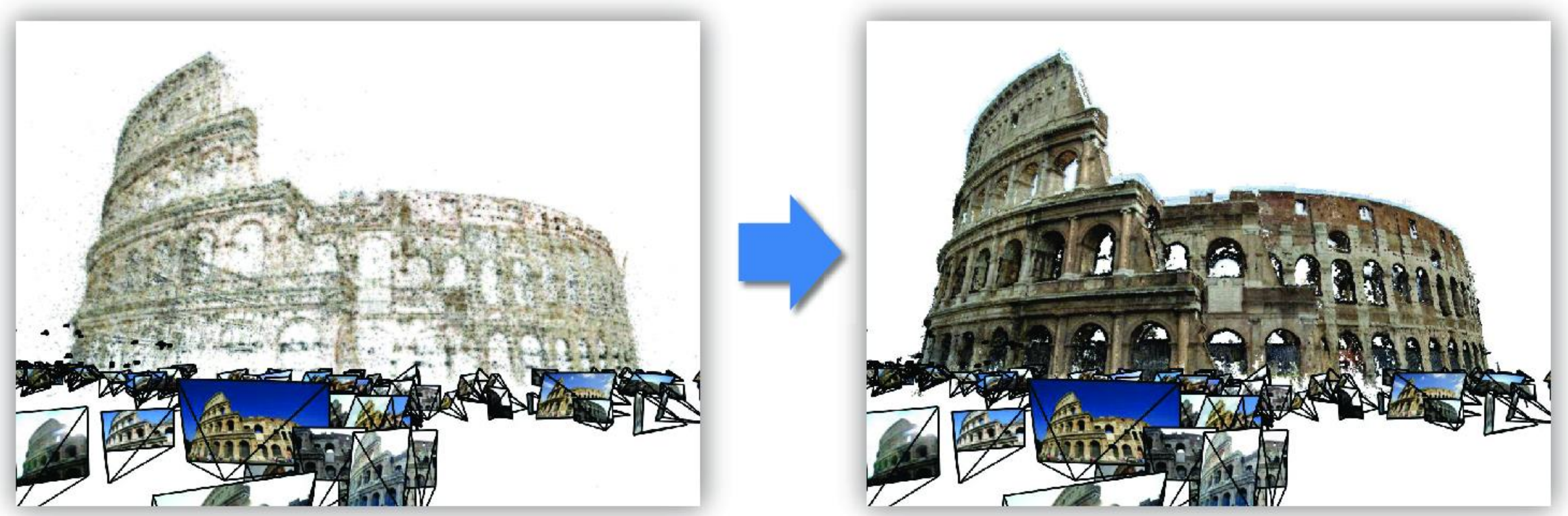
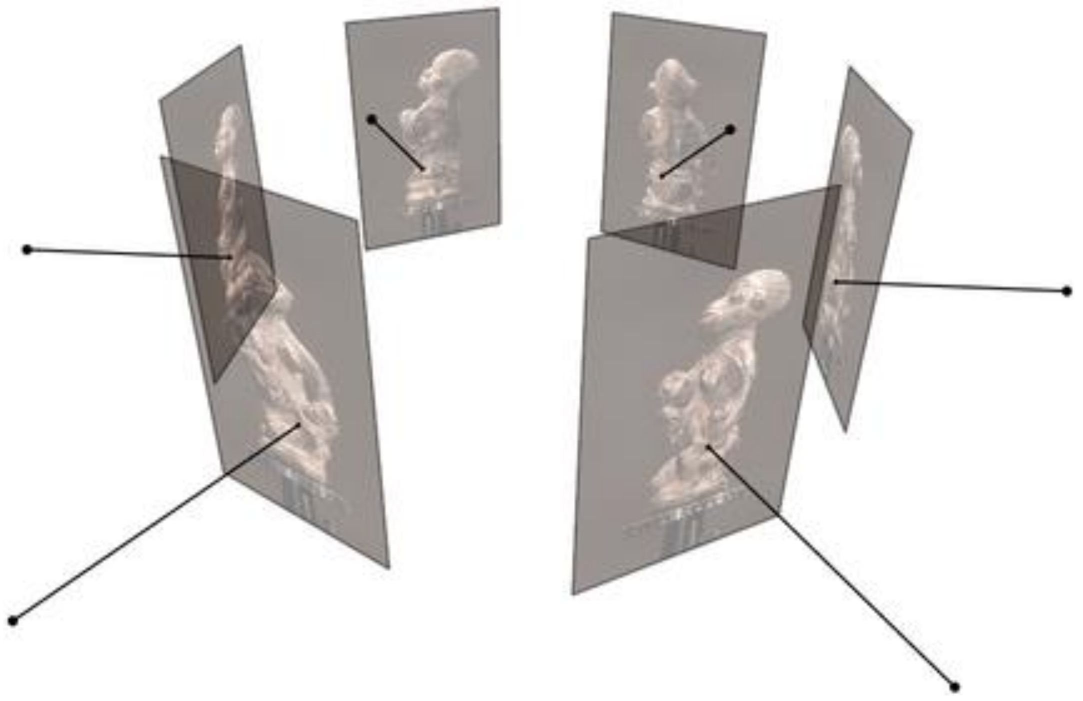
Multiview stereo



Slides largely from
- http://luthuli.cs.uiuc.edu/~daf/courses/CV23/planned.html
- https://www.cs.cornell.edu/courses/cs5670/2018sp/lectures/lec16_mvs.pdf

Multi-view stereo

- Goal: given several images of the same object or scene, compute a representation of its 3D shape



Source: C. Hernandez, N. Snavely

Multi-view stereo

- Goal: given several images of the same object or scene, compute a representation of its 3D shape
- "Images of the same object or scene"
  - Arbitrary number of images (from two to thousands)
  - Arbitrary camera positions (special rig, camera network or video)
  - Calibration may be known or unknown

Multi-view stereo

- Goal: given several images of the same object or scene, compute a representation of its 3D shape
- "Images of the same object or scene"
  - Arbitrary number of images (from two to thousands)
  - Arbitrary camera positions (special rig, camera network or video)
  - Calibration may be known or unknown
- "Representation of 3D shape"
  - Depth maps
  - Meshes
  - Point clouds
  - Volumetric models
  - ….

Outline

- Applications
- Two view stereo
- Multi-view stereo
  - Plane sweep stereo
  - Patch-based multi-view stereo
- Demonstration

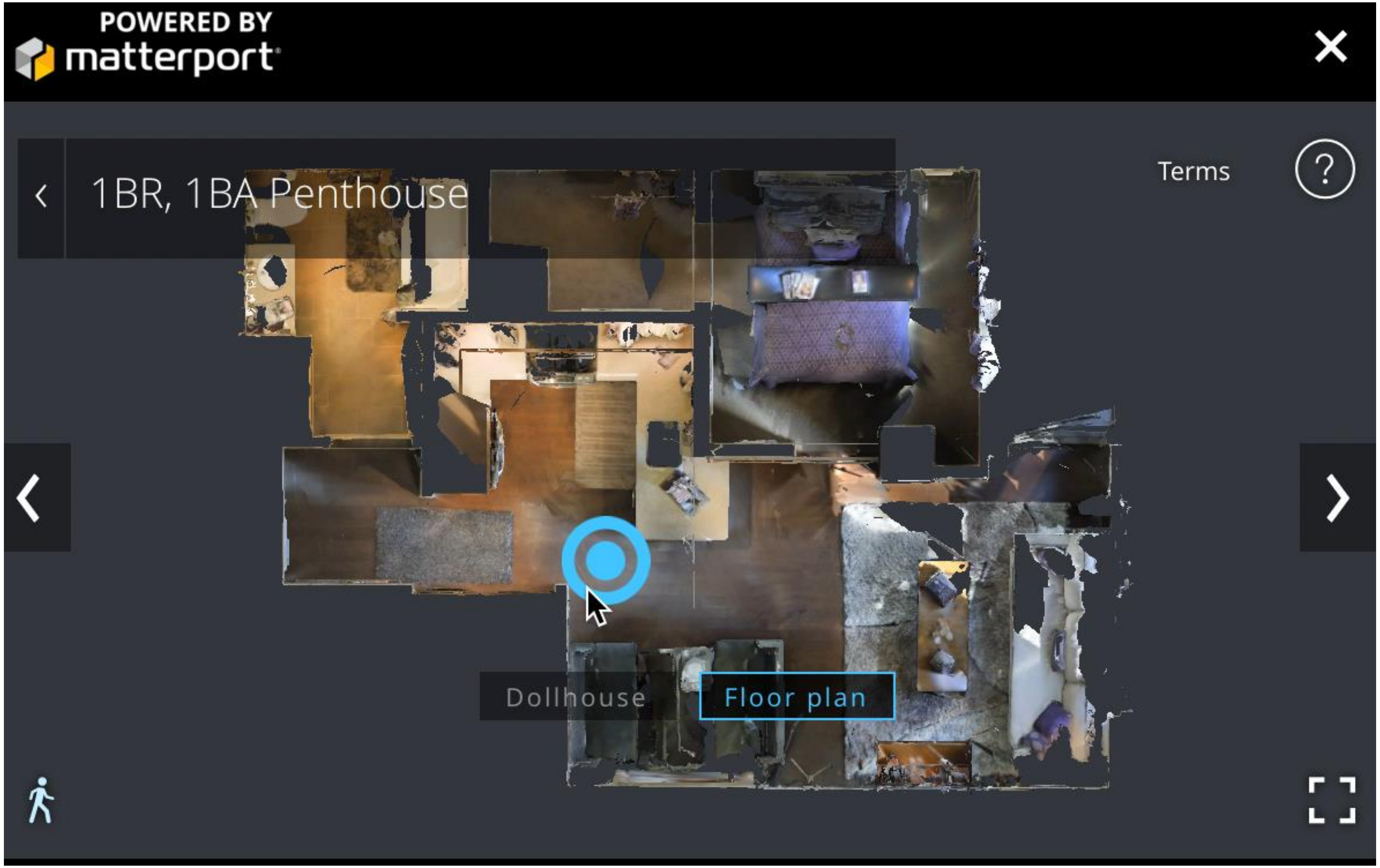
Applications

Source: N. Snavely

Applications

Source: N. Snavely

Applications



Source: N. Snavely

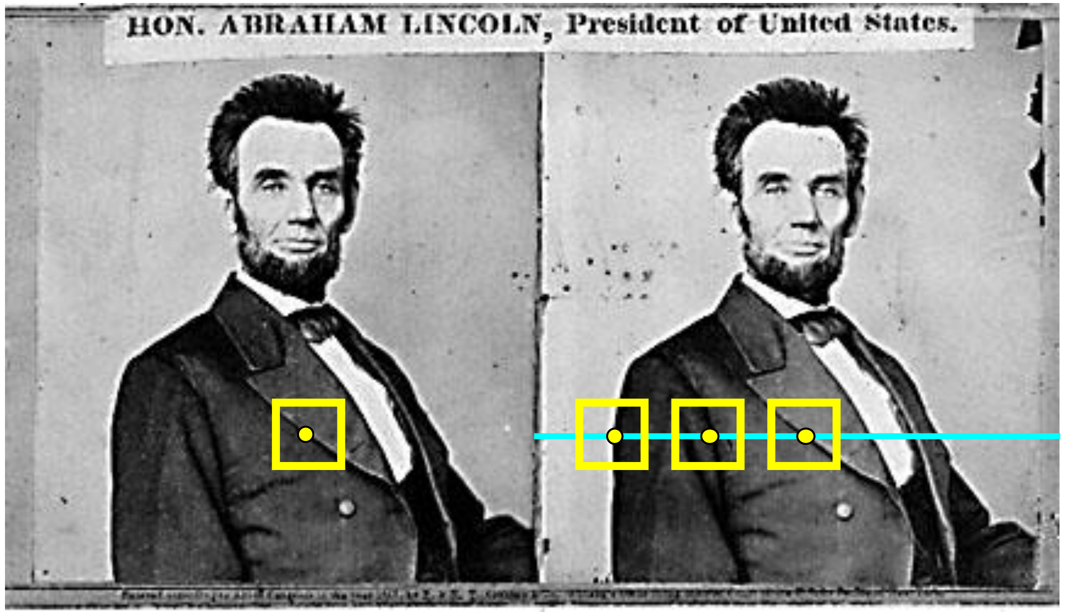
Outline

- Applications
- Two view stereo
- Multi-view stereo
  - Plane sweep stereo
  - Patch-based multi-view stereo
- Demonstration

Problem formulation

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

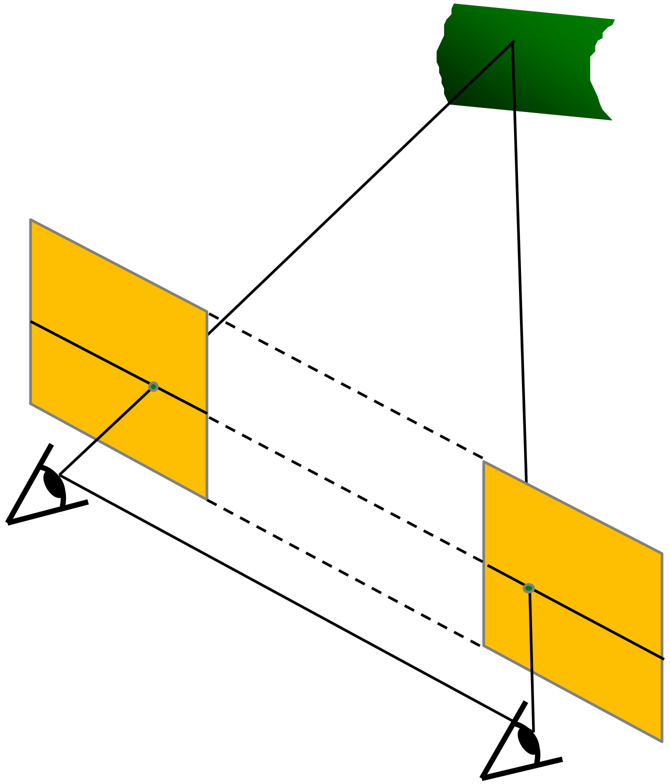
Basic stereo matching algorithm



- 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?

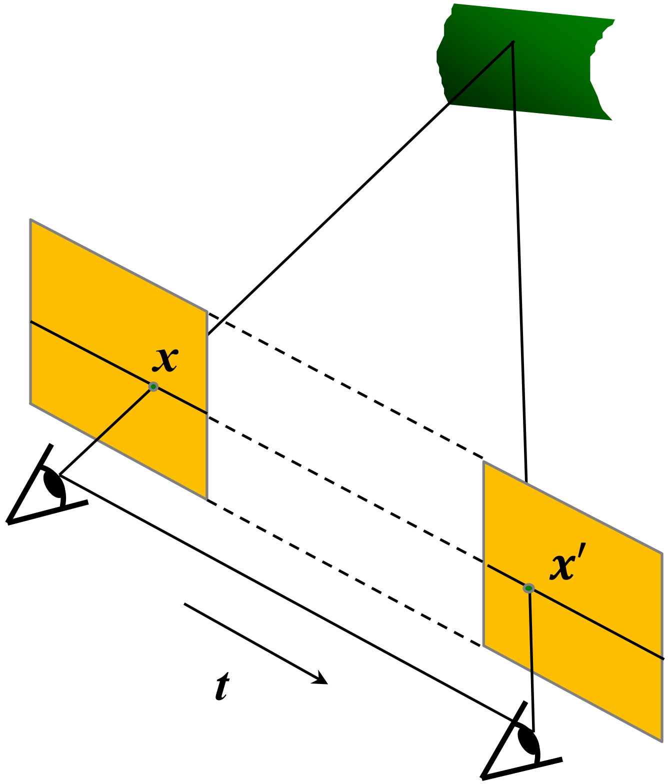
Parallel images



- 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

Can you mathematically prove this?

Essential matrix for parallel images



Epipolar constraint:

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

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

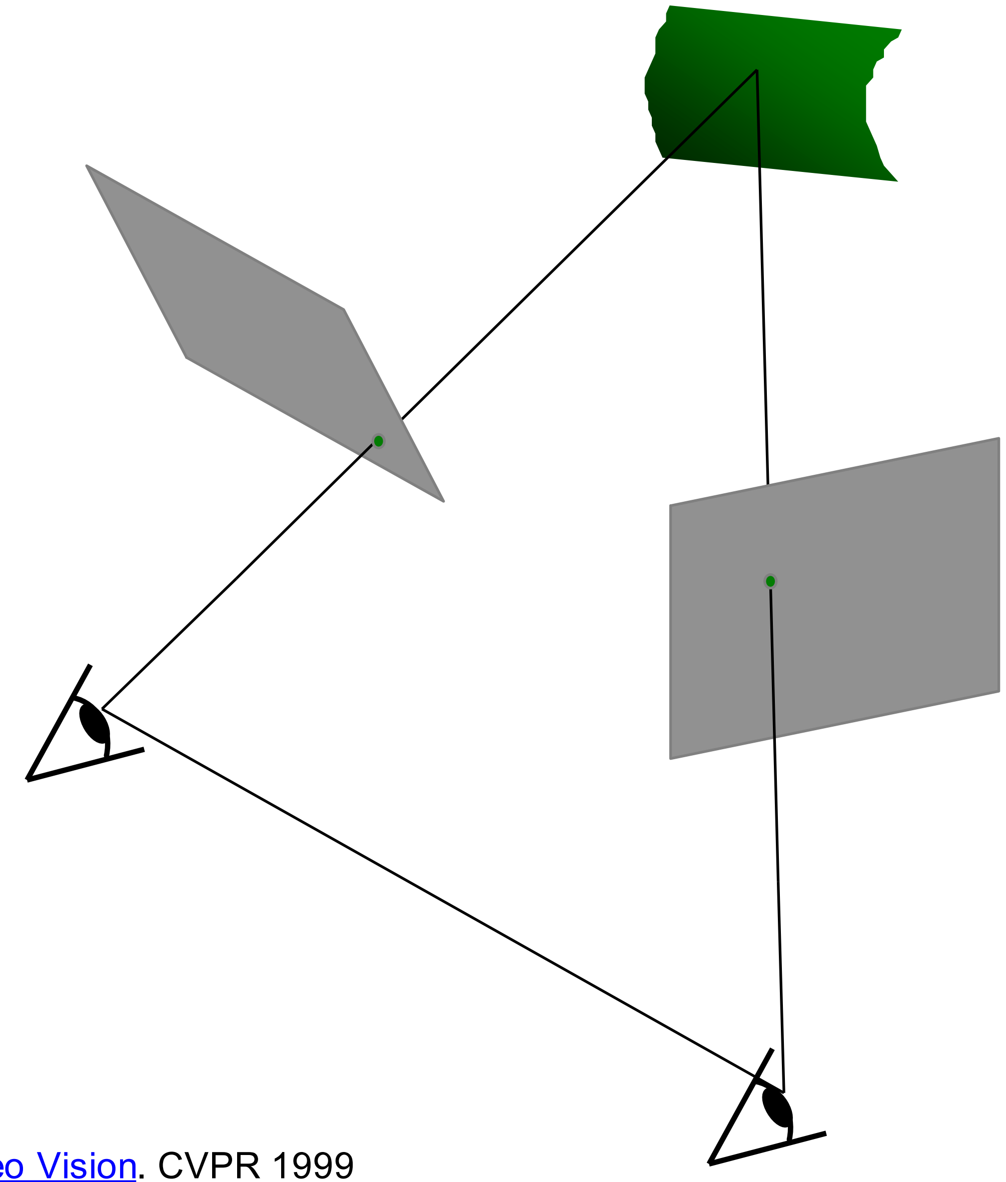
$$\boldsymbol{E} = [\boldsymbol{t}_\times]\boldsymbol{R} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -t \\ 0 & t & 0 \end{bmatrix}$$

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

The y-coordinates of corresponding points are the same!

Stereo image rectification

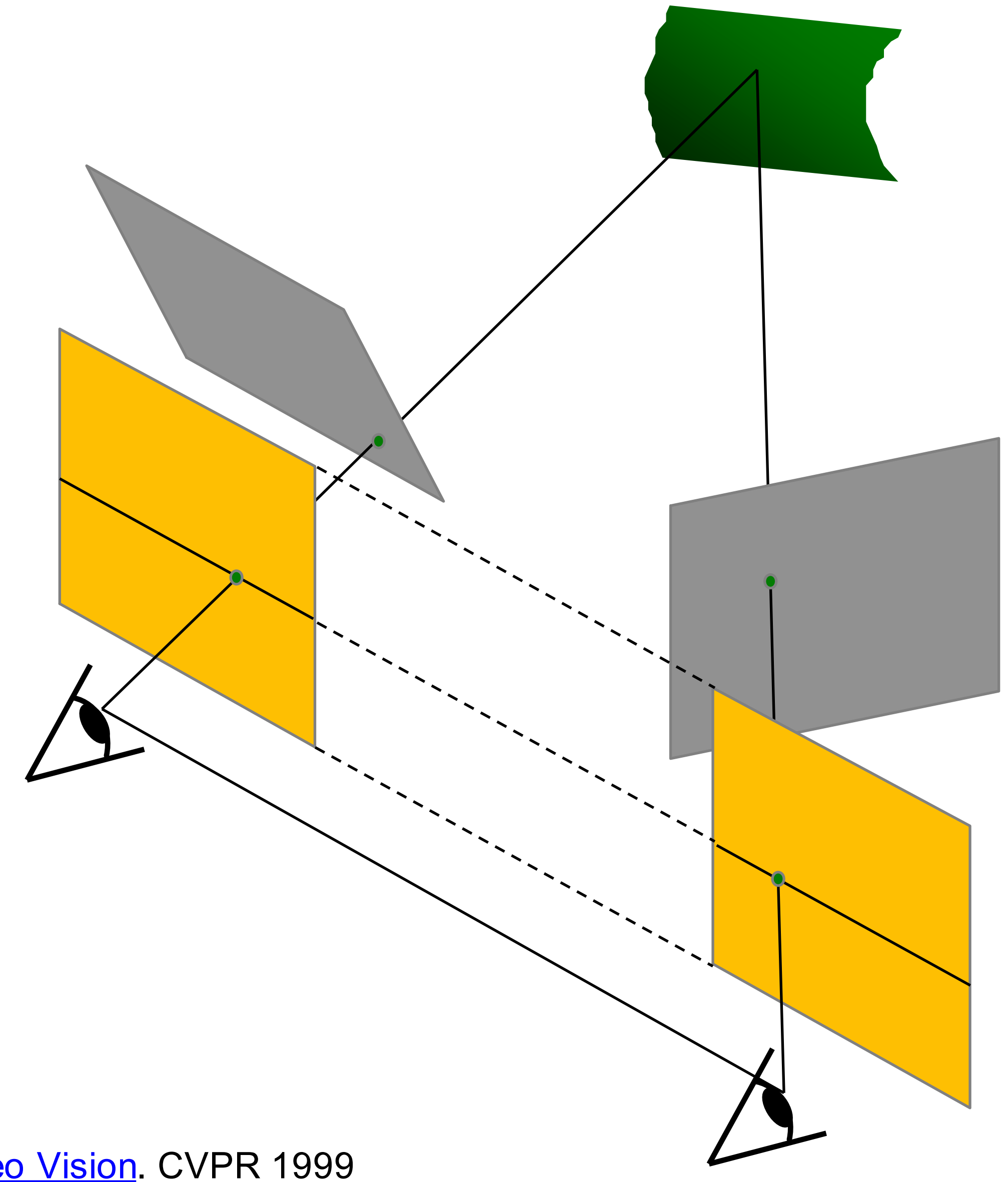
- If the image planes are not parallel, we can find homographies to project each view onto a common plane parallel to the baseline



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:

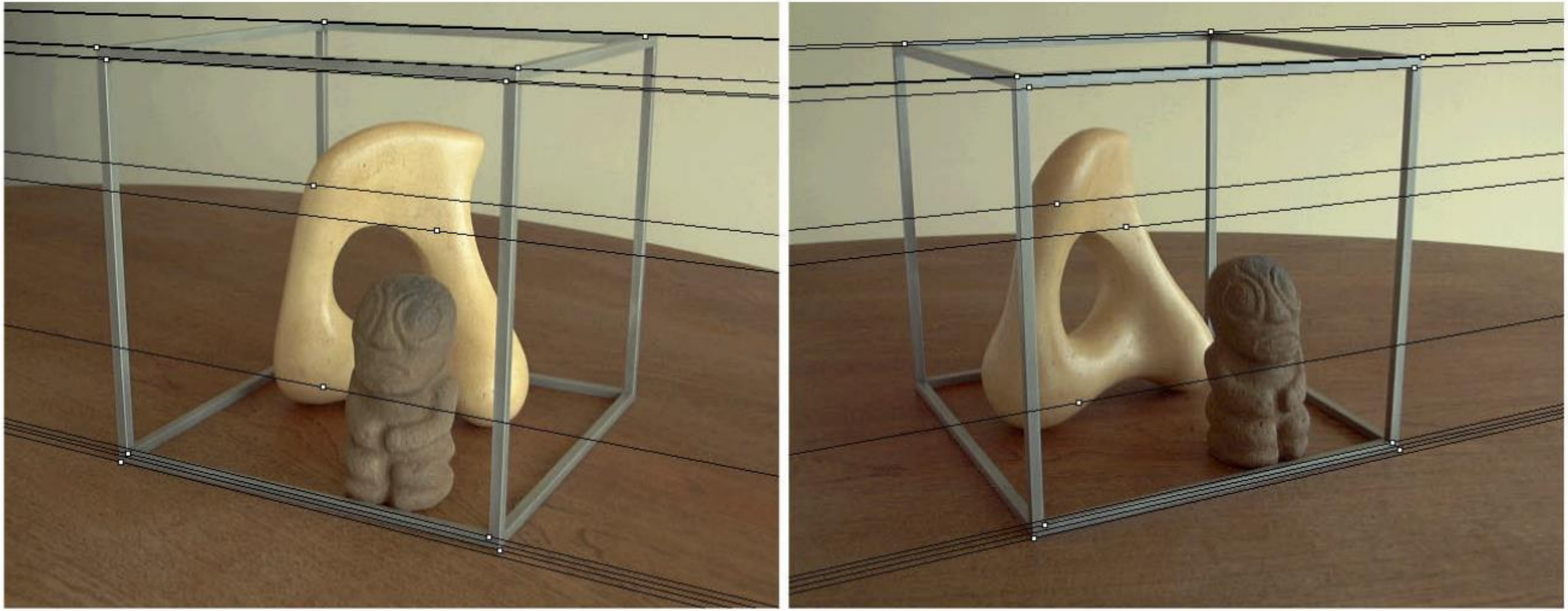


Image source

Stereo image rectification

- After rectification:

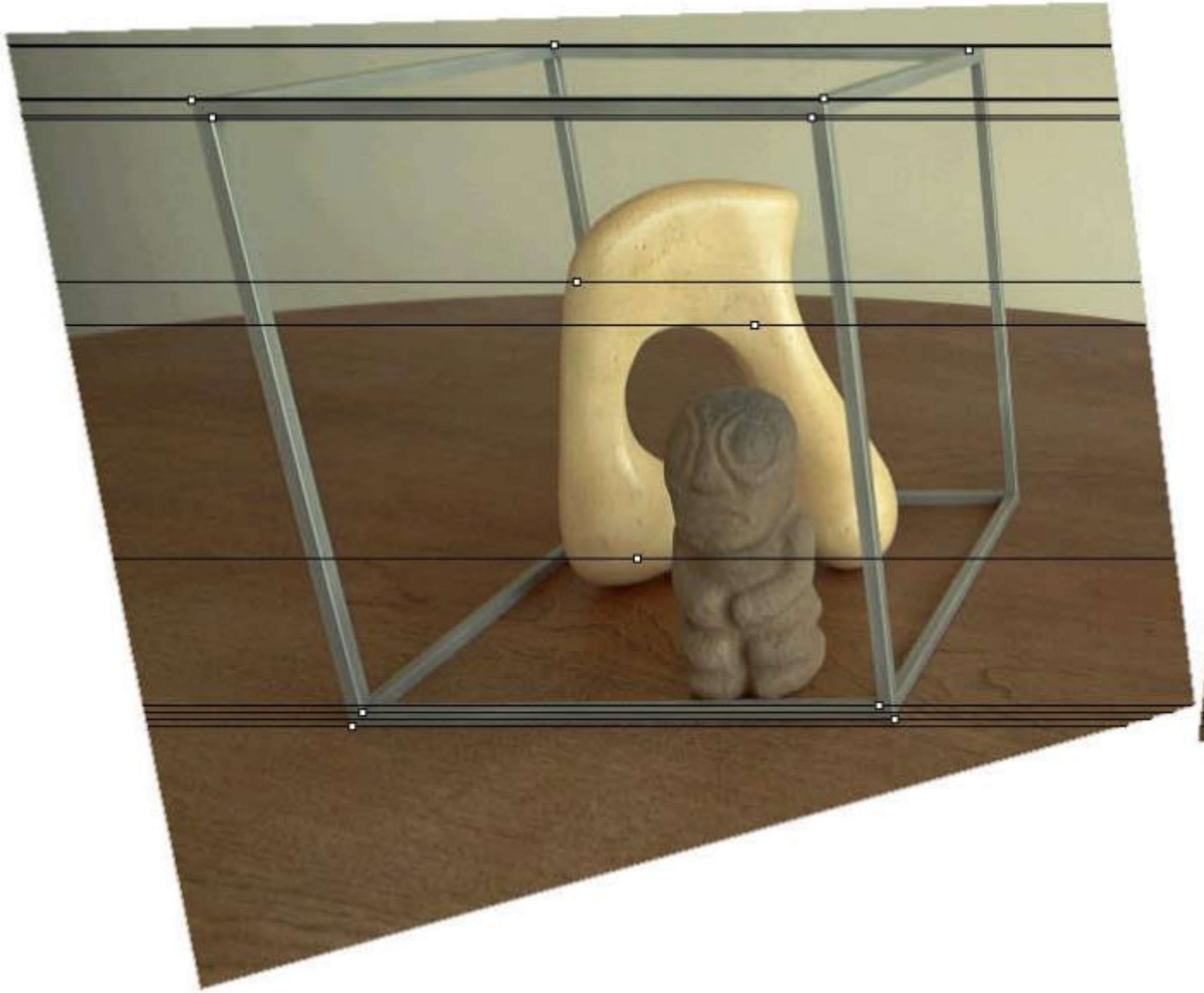
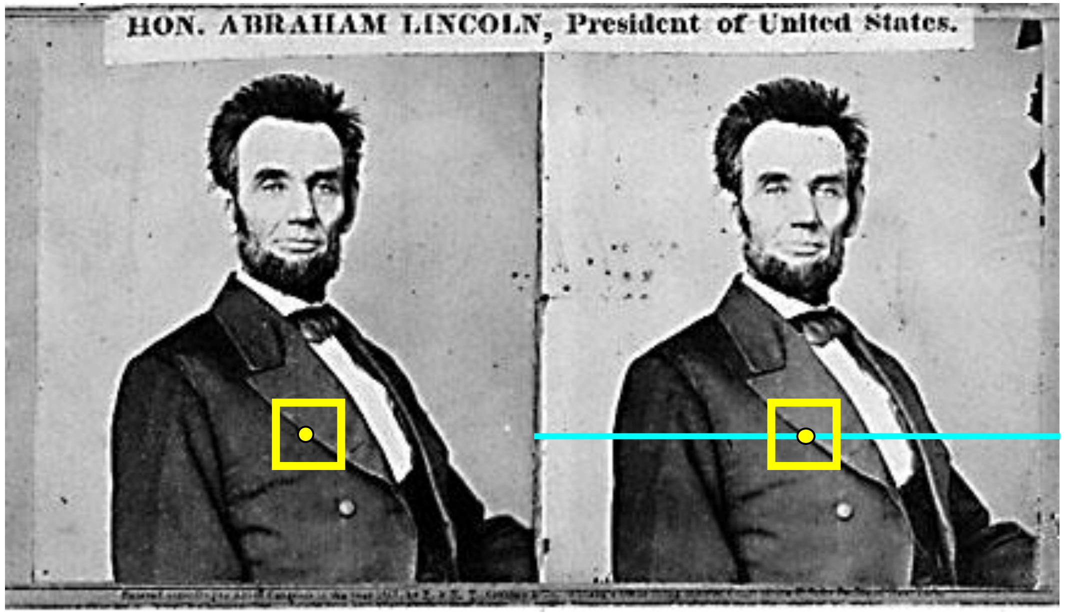


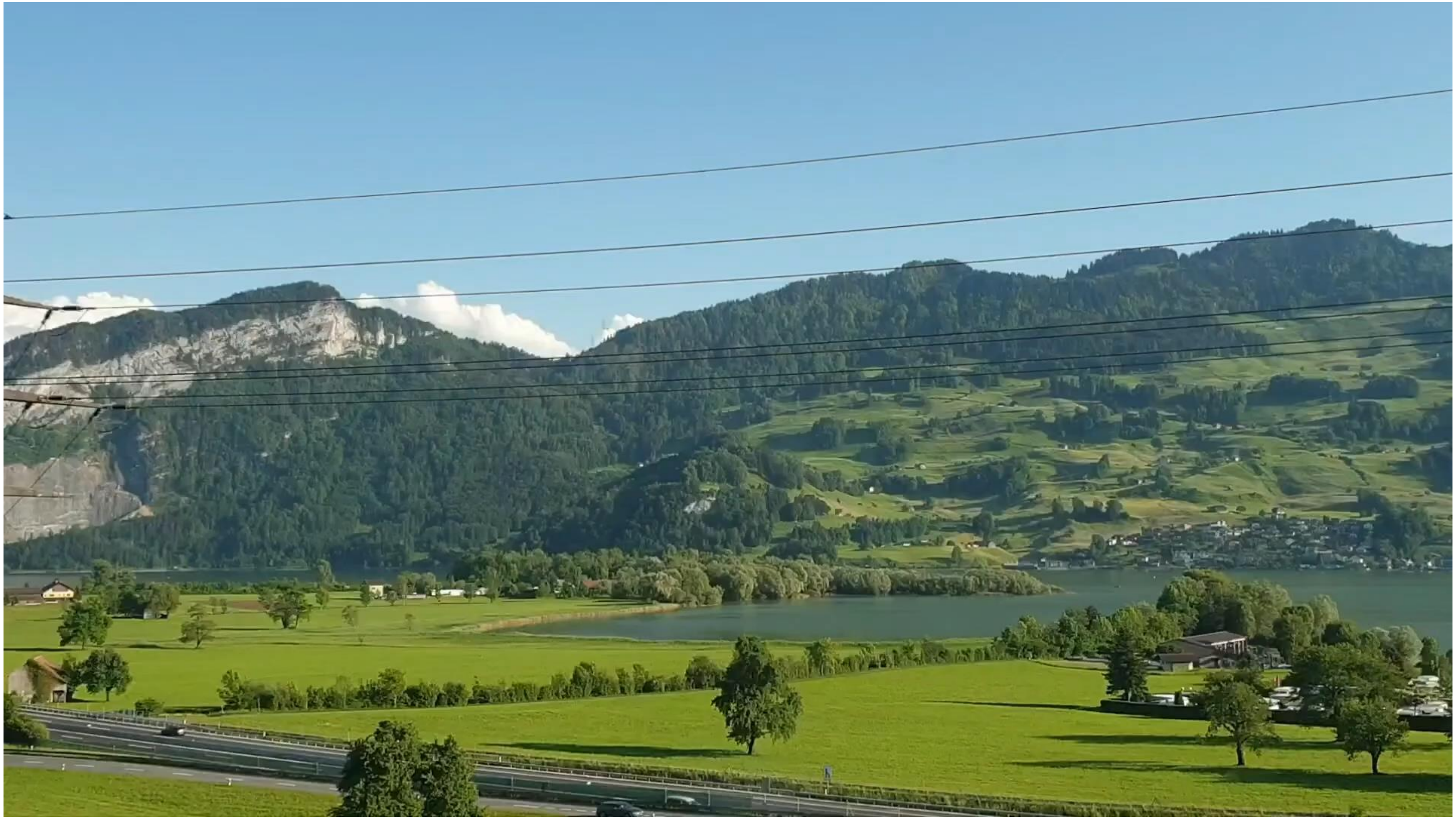
Image source

Basic stereo matching algorithm

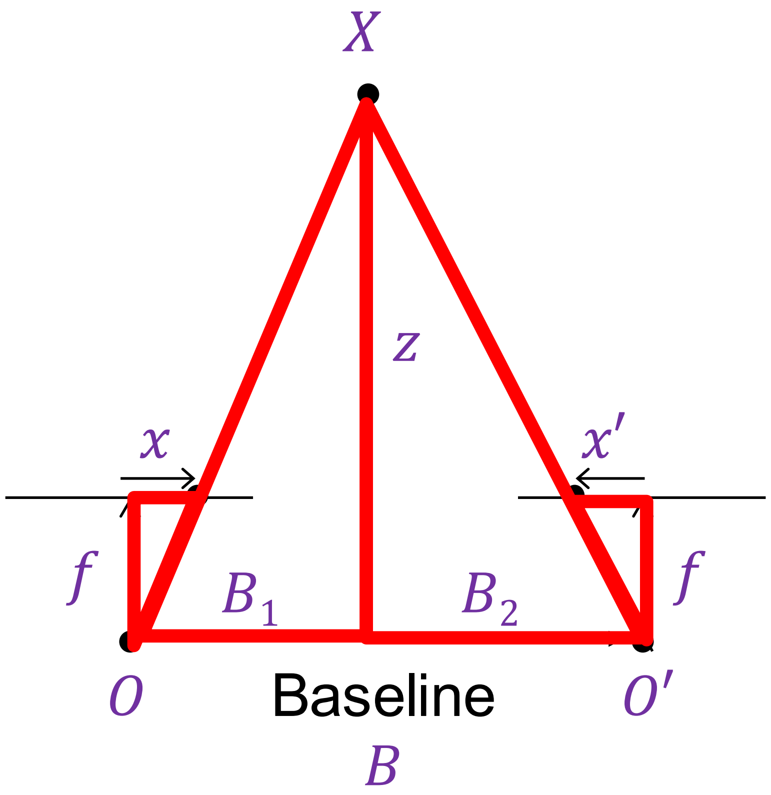


- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Triangulate the matches to get depth information

Depth from disparity



Depth from disparity



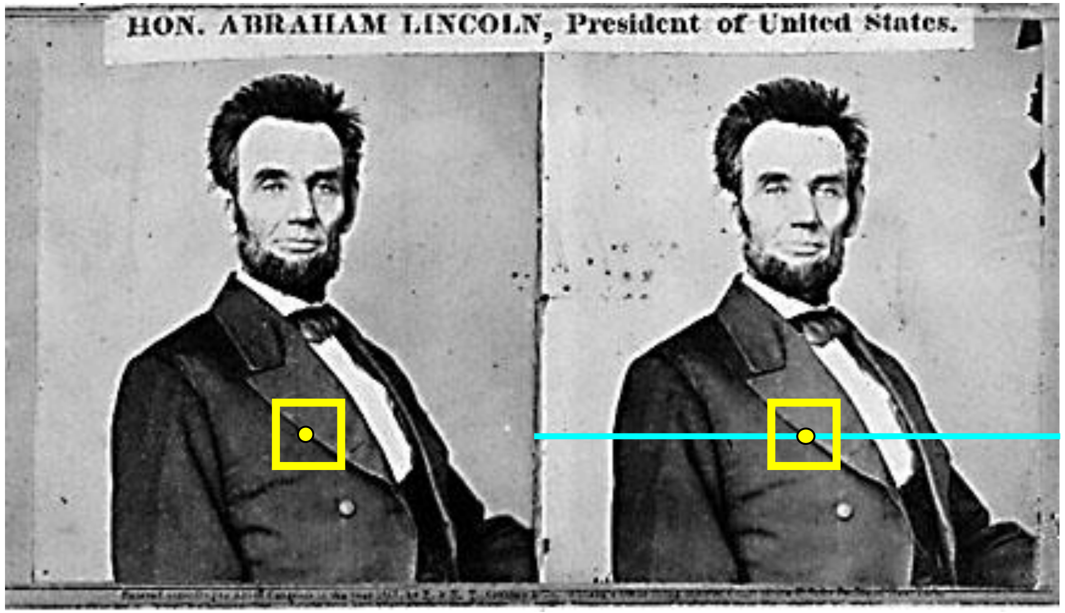
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{-x'}{f} = \frac{B_2}{z}$$

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

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

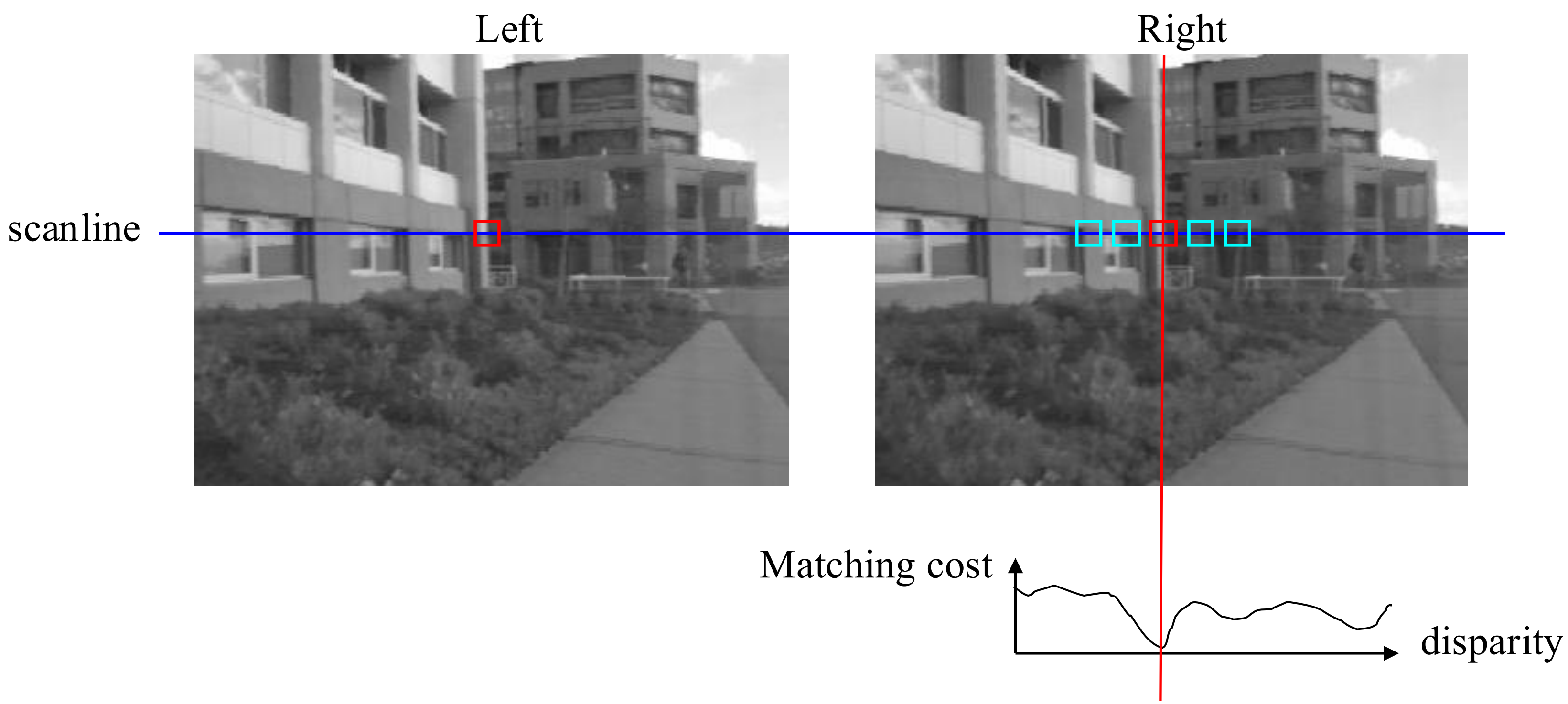
Disparity is inversely proportional to depth!

Basic stereo matching algorithm



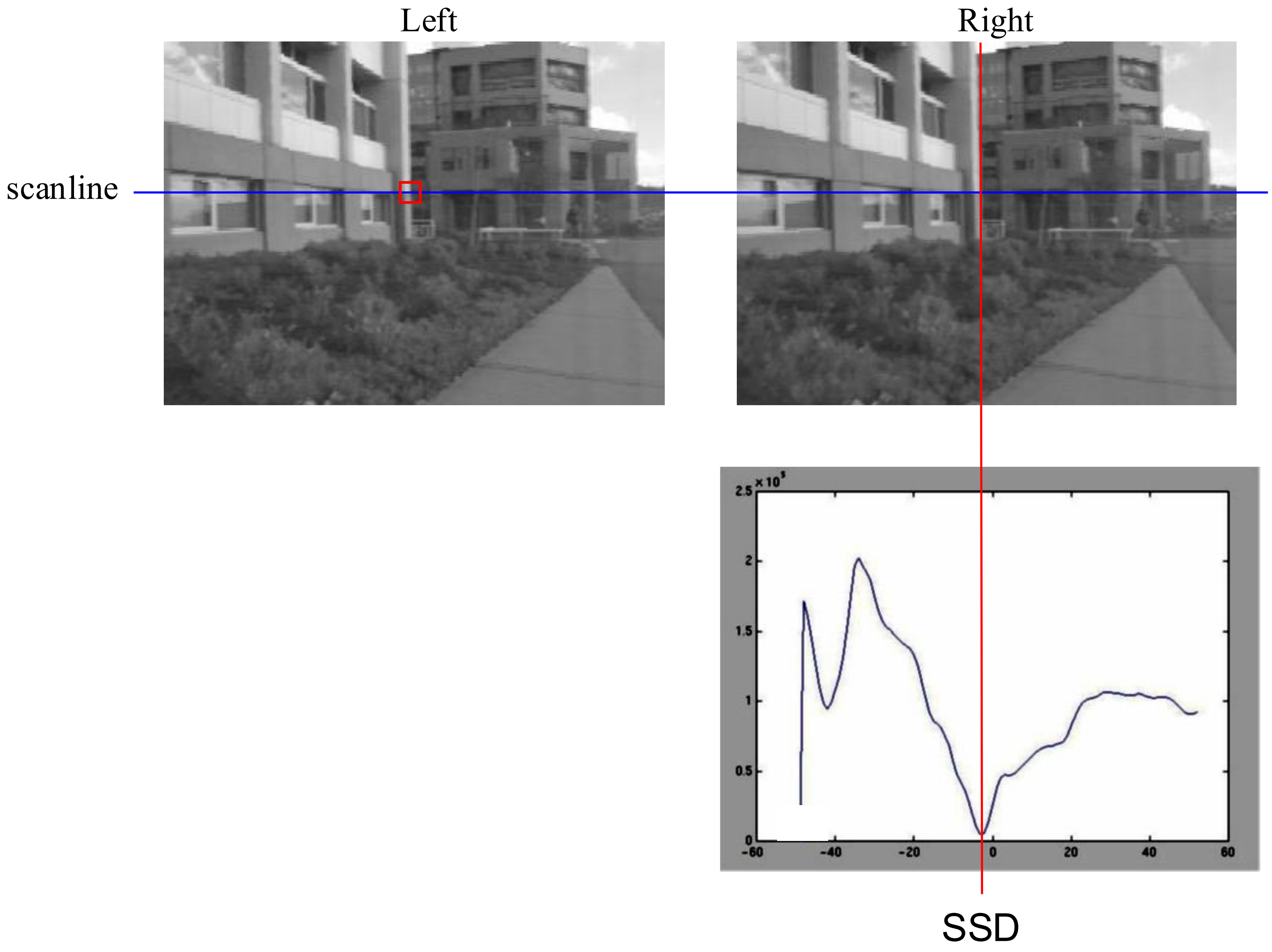
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Compute disparity $x - x'$ and set $\text{depth}(x)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

Correspondence search

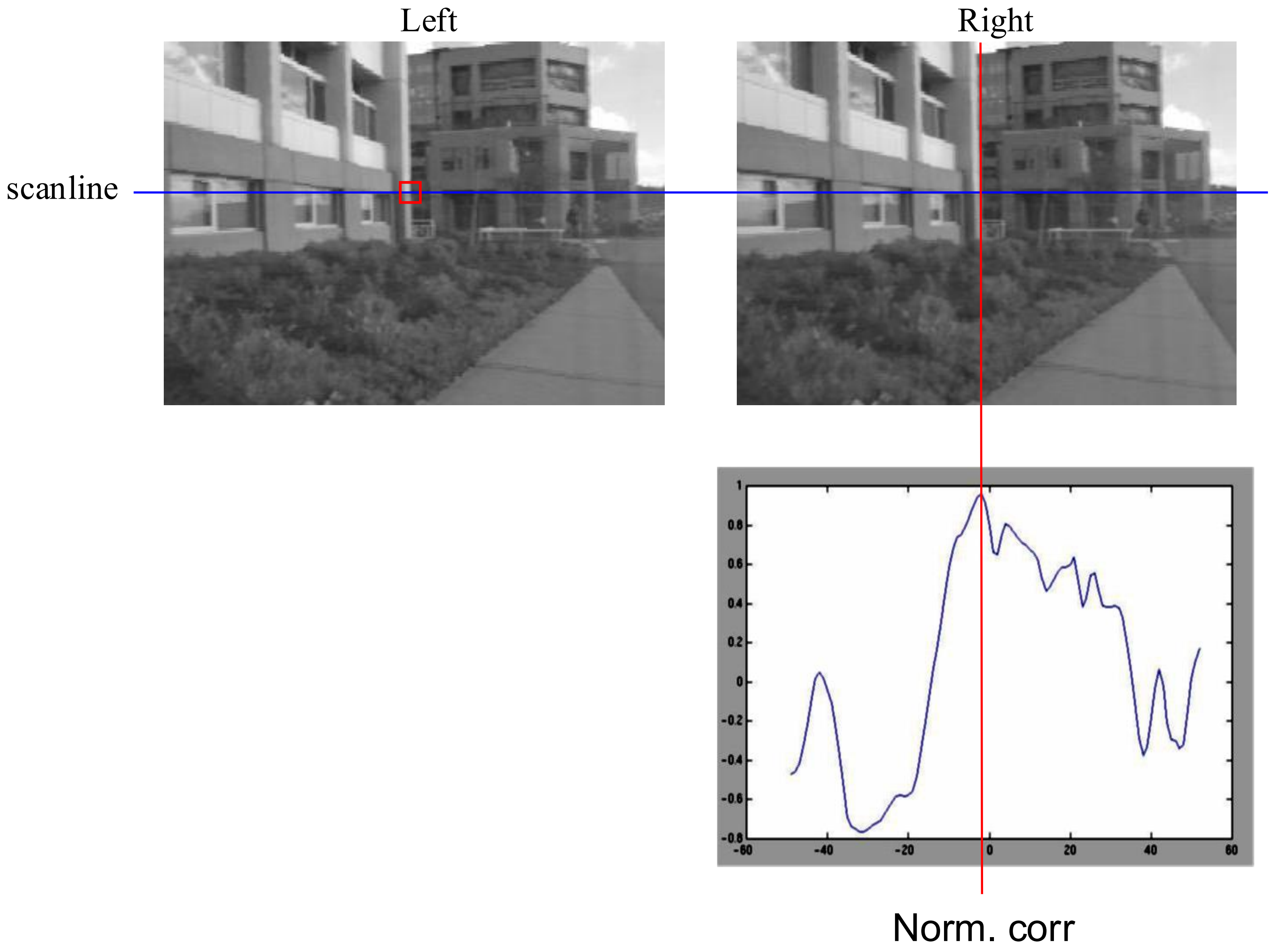


- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

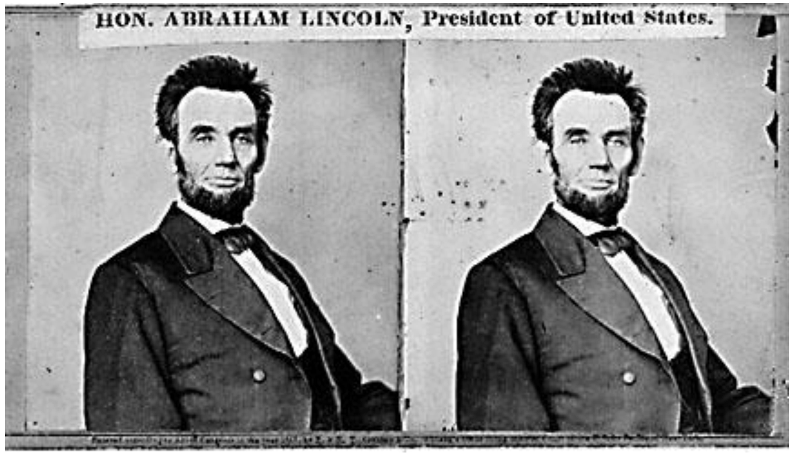
Correspondence search



Correspondence search

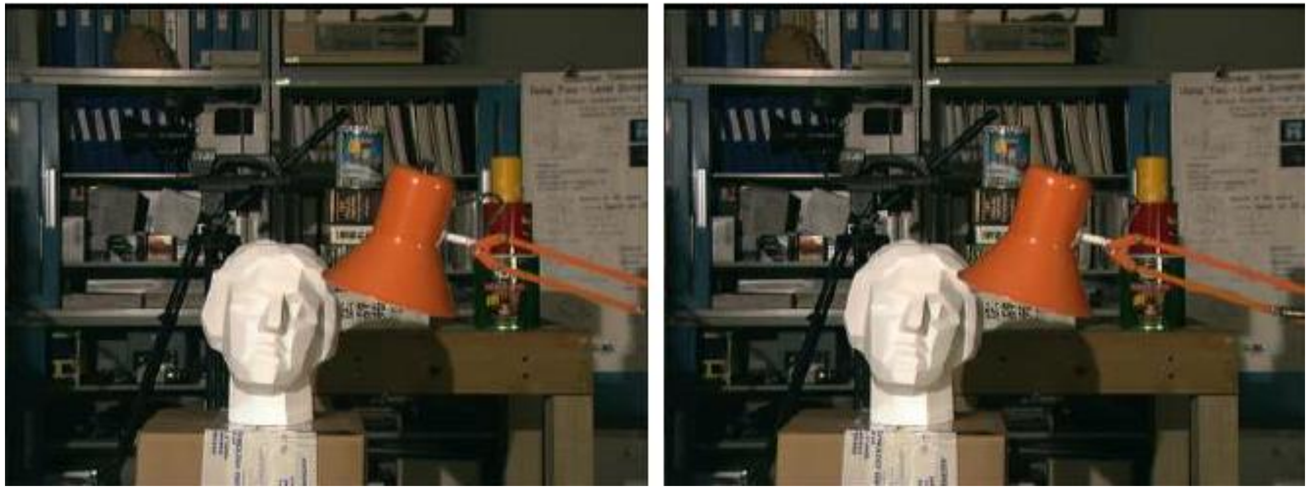


Where will basic window search fail?



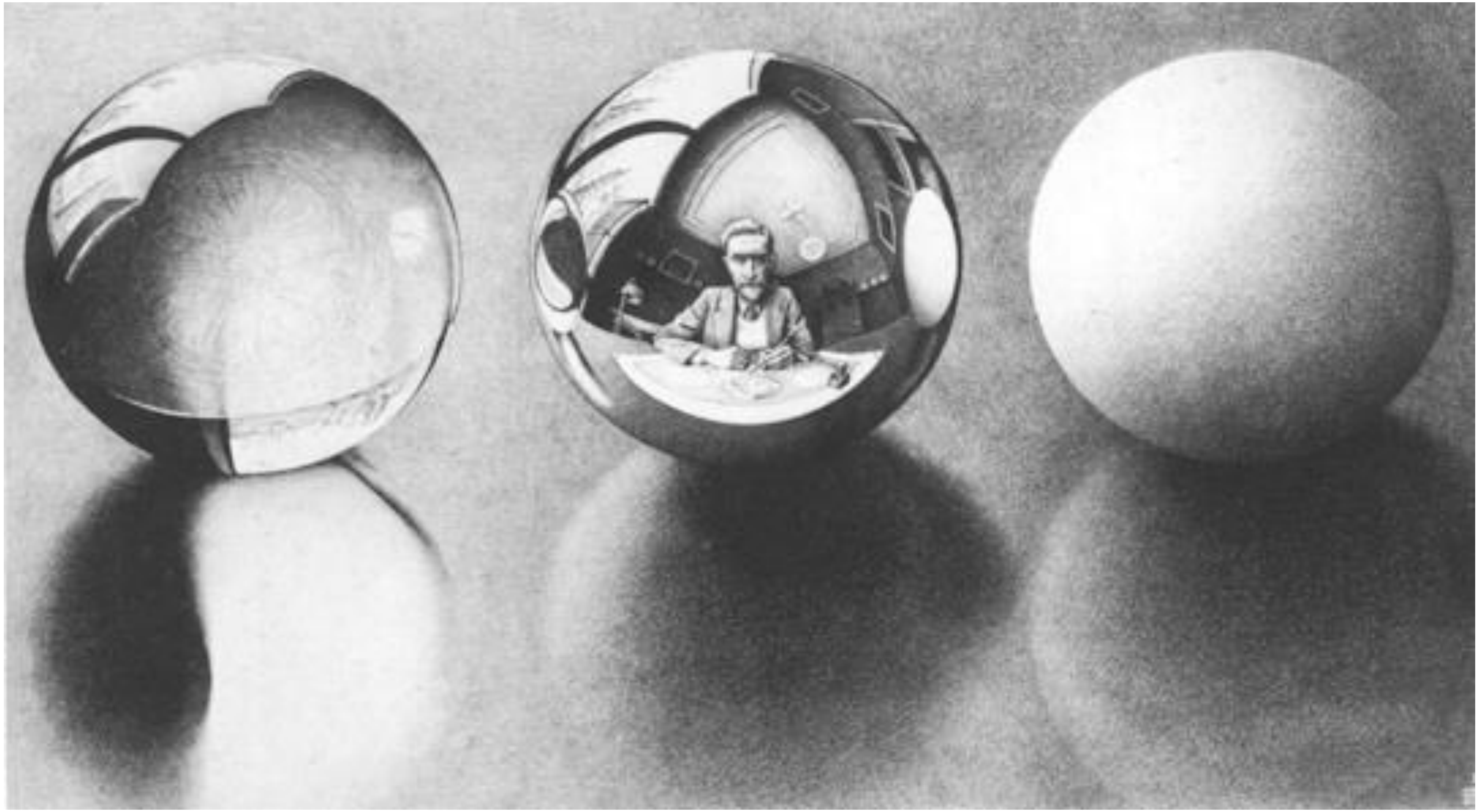
Textureless surfaces

Occlusions, repetition



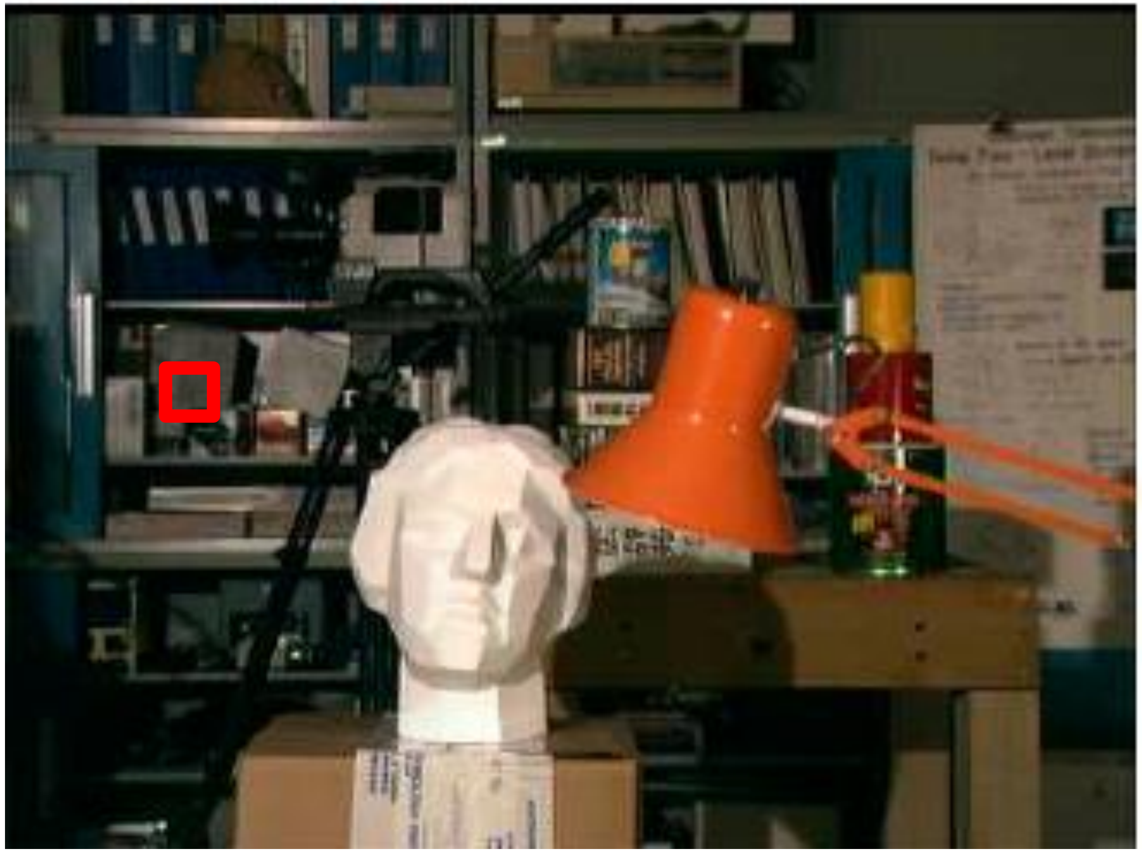
Non-Lambertian surfaces, specularities

Non-Lambertian surfaces

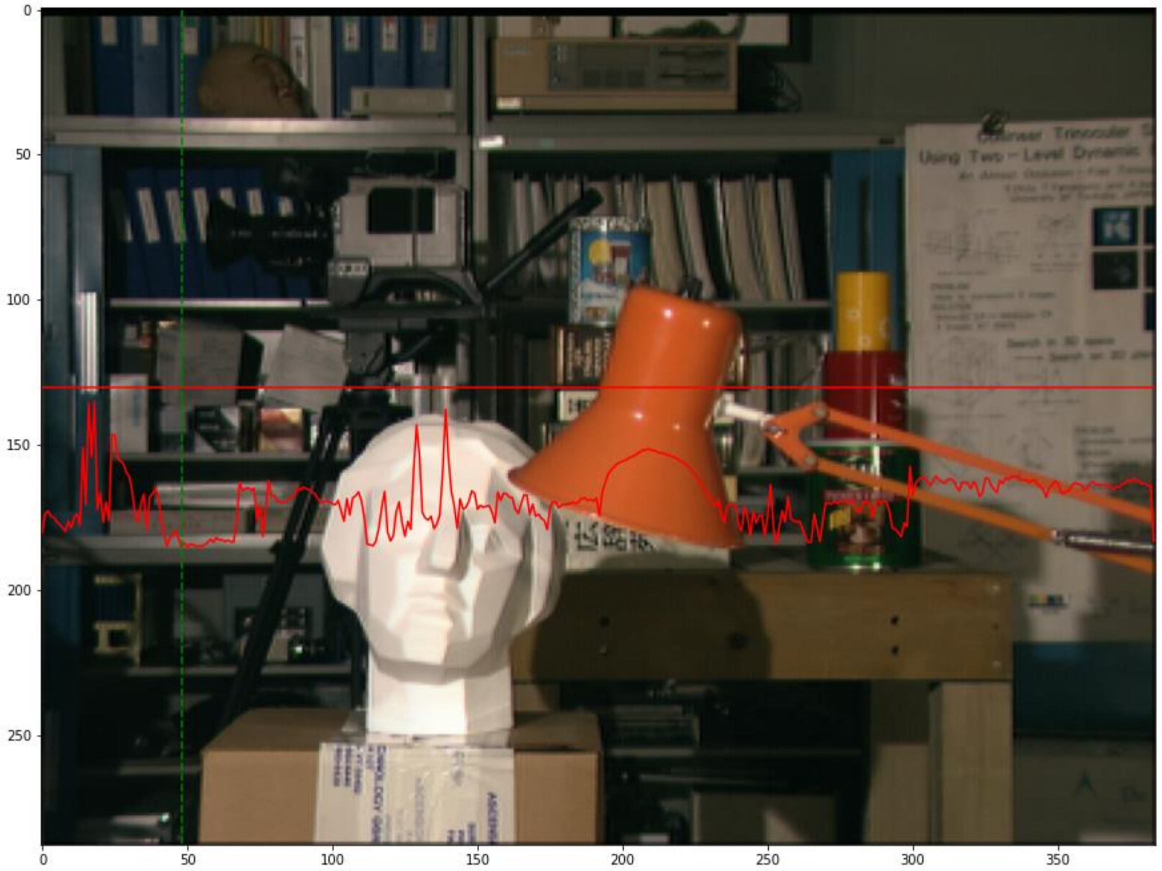


The same object look different from different angles

Example: Textured neighborhood



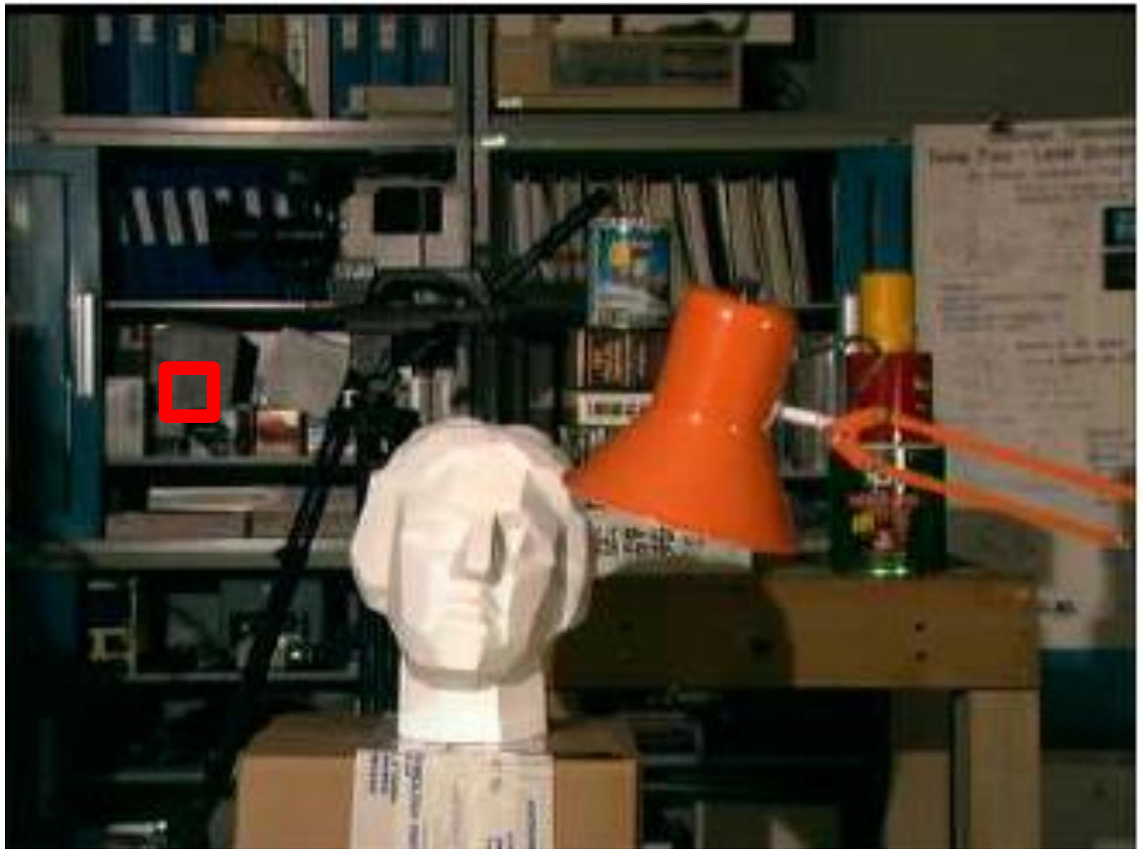
Window size: 1 pixel



Source: [D. Hoiem]

Example: Textured neighborhood

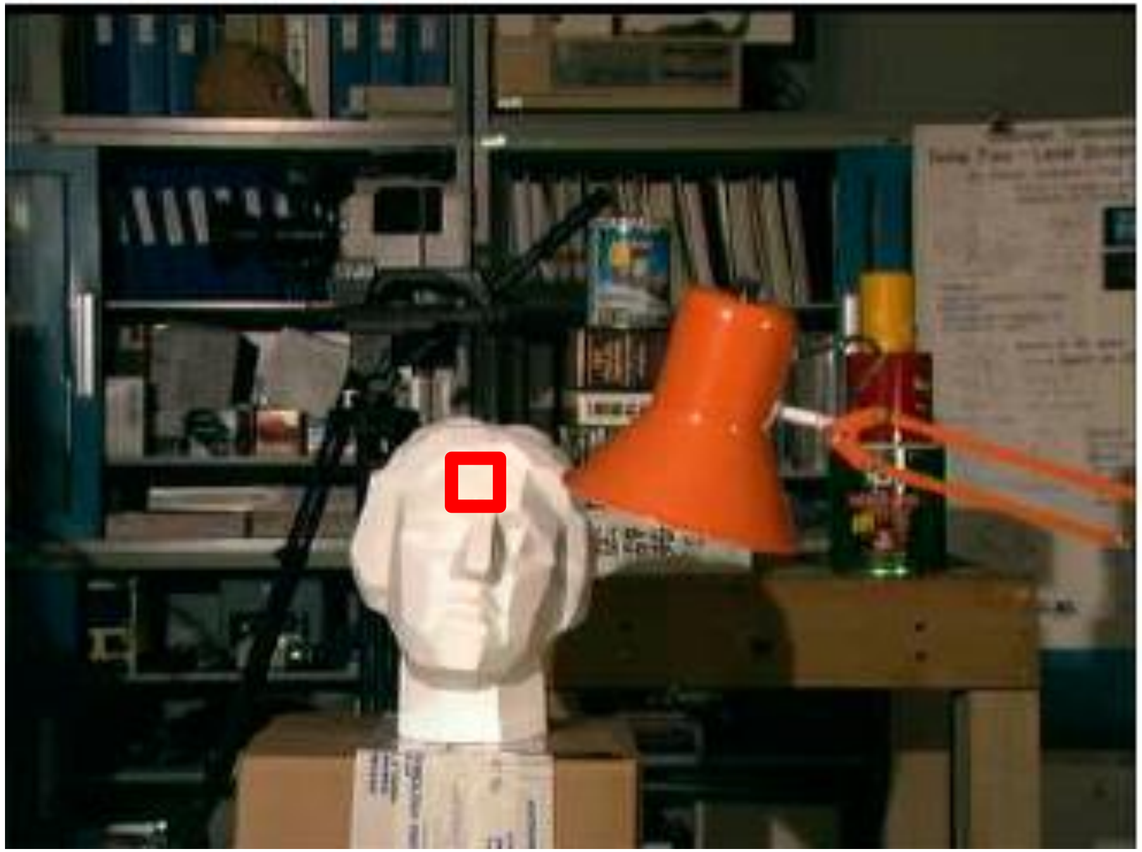
Window size: 7 pixels



Source: [D. Hoiem]()

Example: Smooth neighborhood

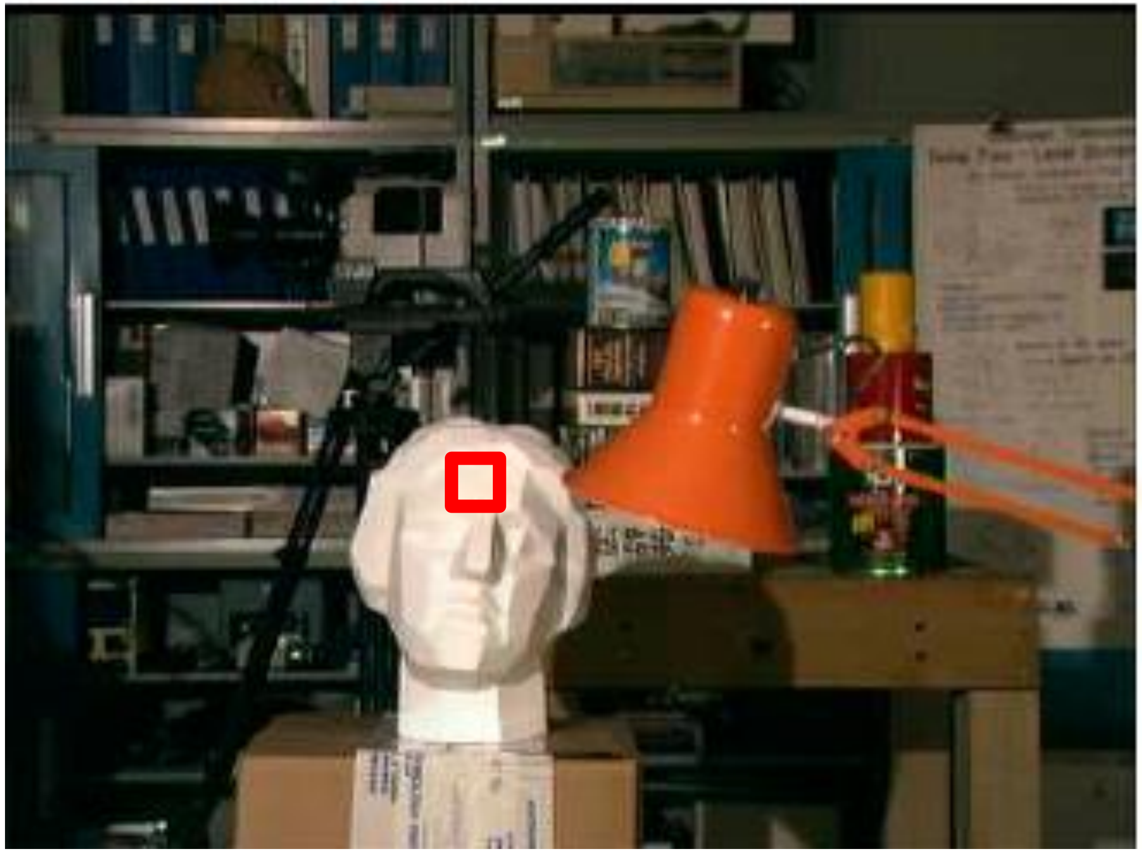
Window size: 1 pixel

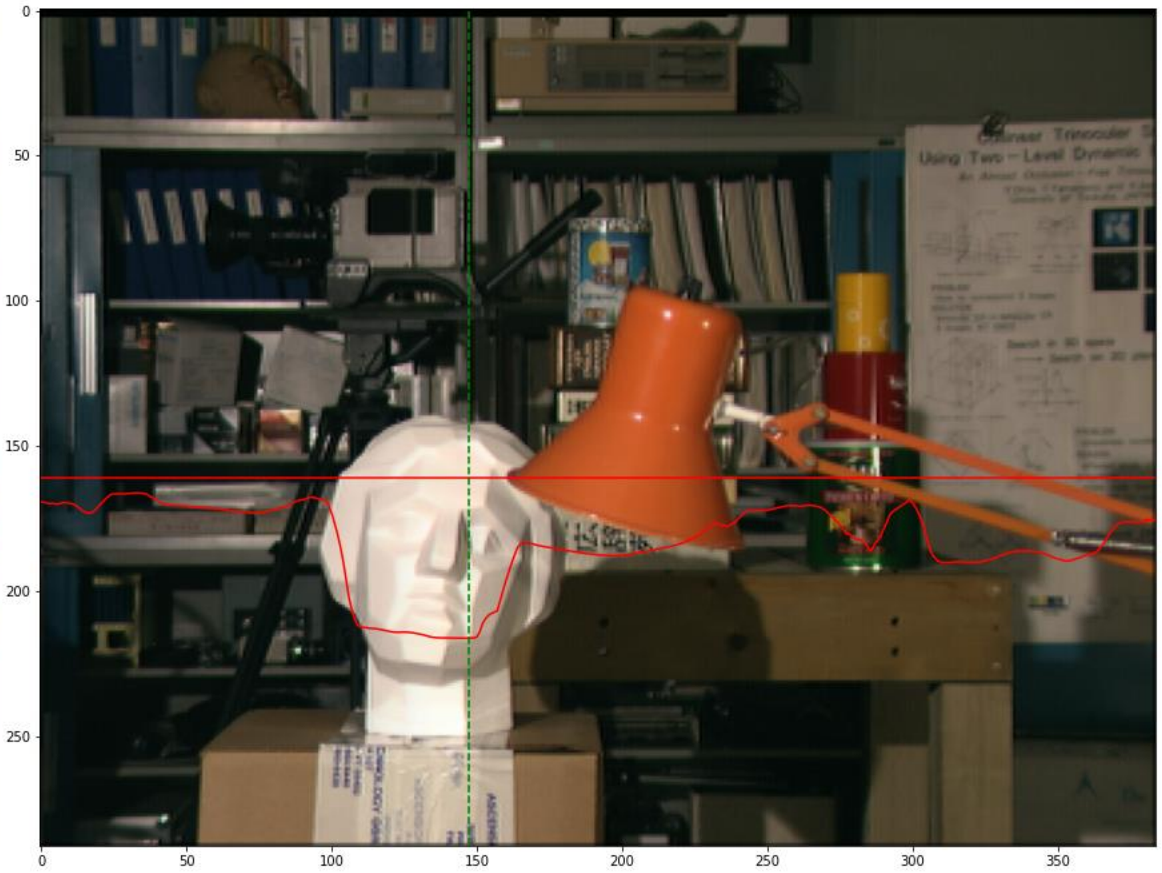


Source: [D. Hoiem]

Example: Smooth neighborhood

Window size: 7 pixels

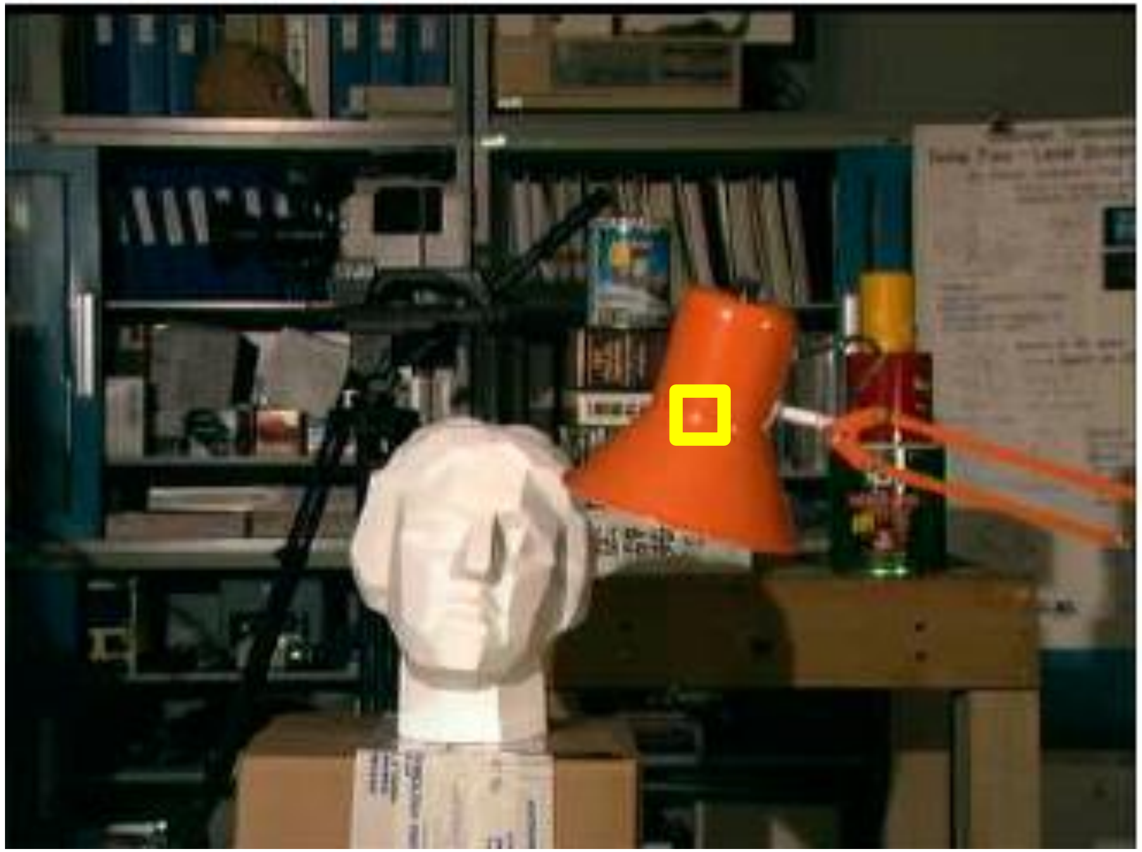




Source: [D. Hoiem]()

Example: Specular highlight

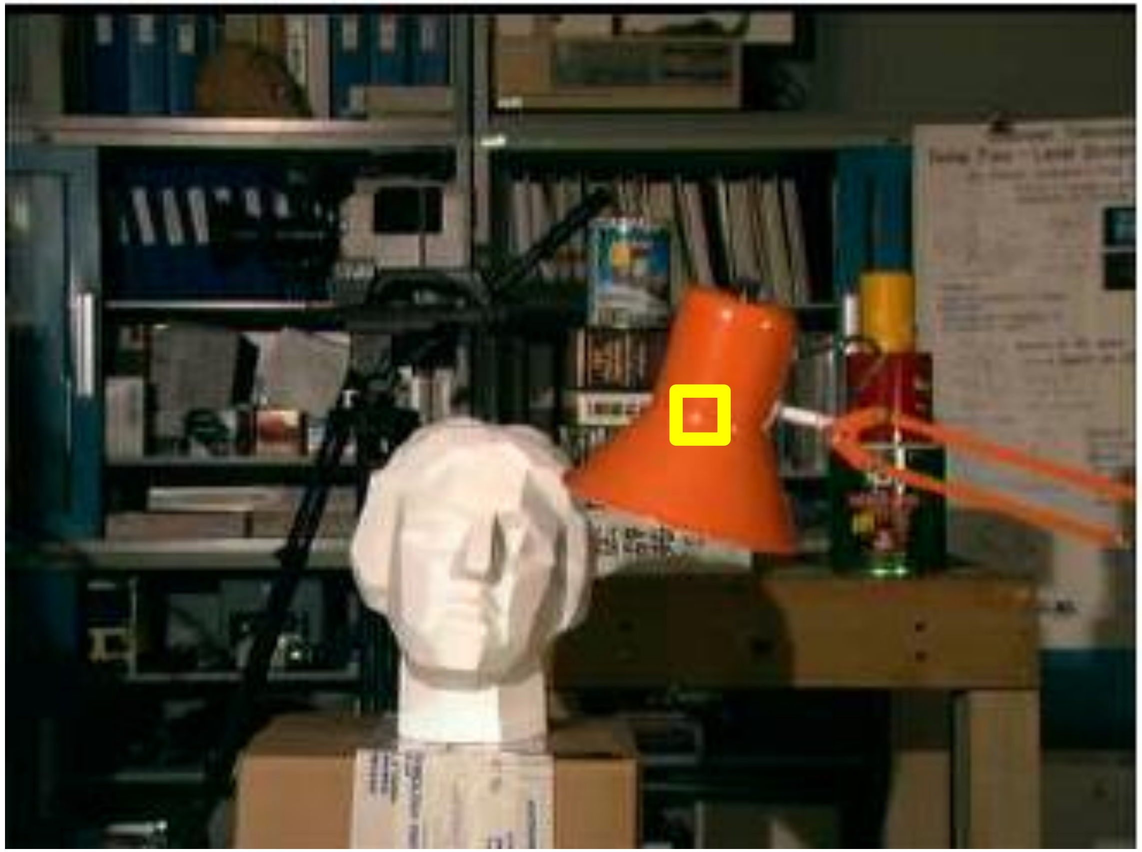
Window size: 1 pixel



Source: [D. Hoiem]()

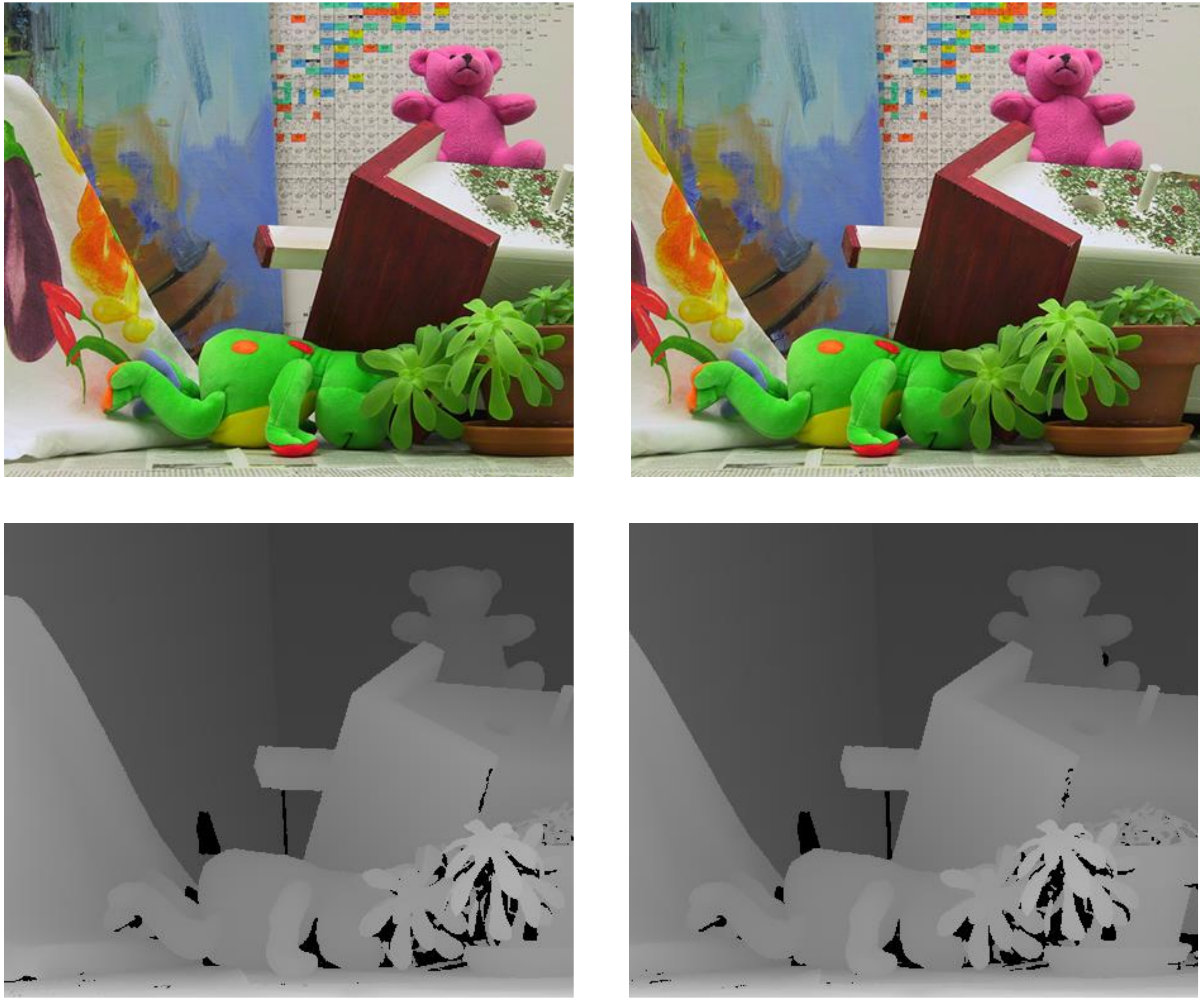
Example: Specular highlight

Window size: 7 pixels



Source: [D. Hoiem]()

Resulting depth maps



Kinect: Structured infrared light

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

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

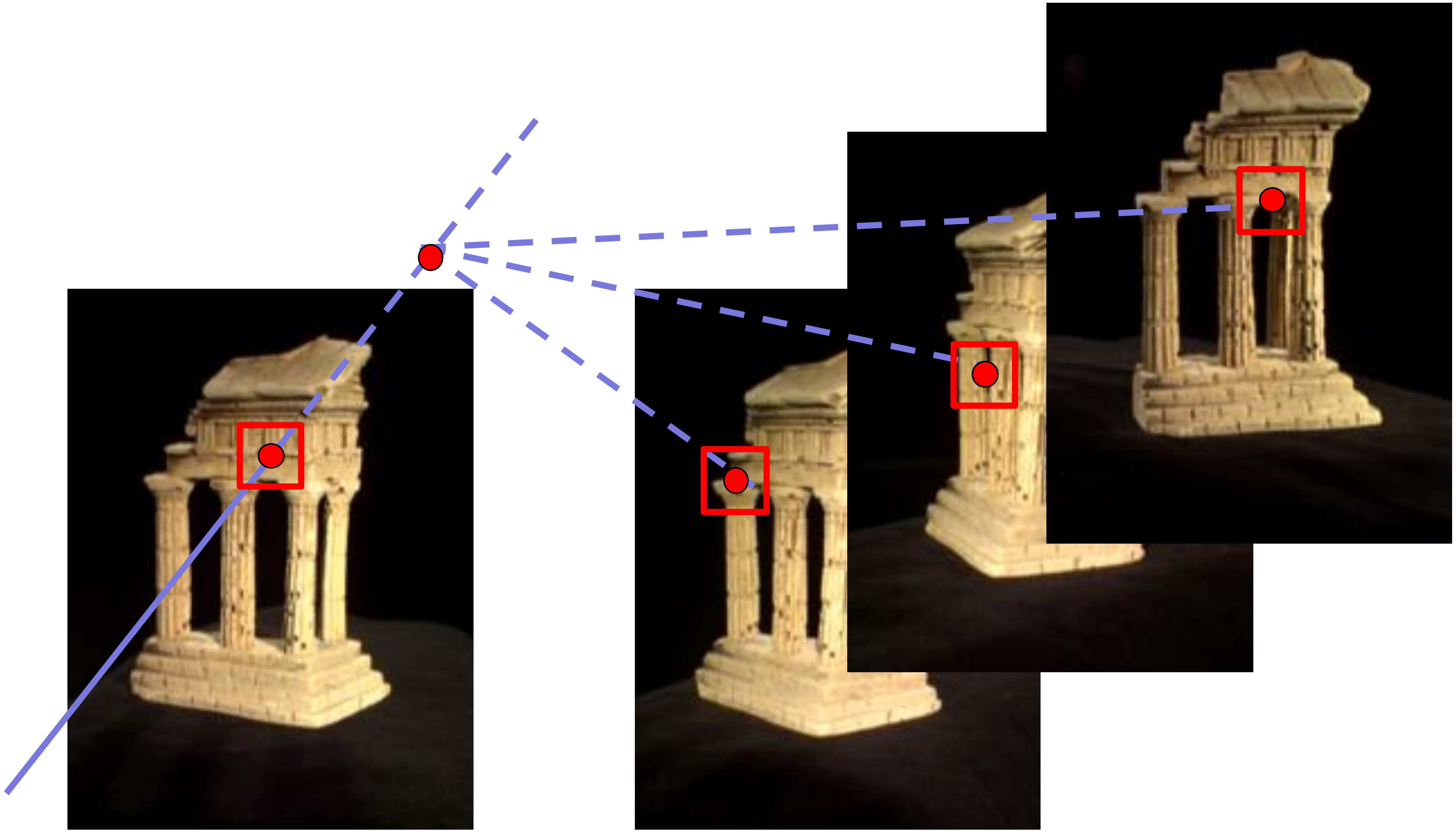
- A single range scan is not sufficient to capture a complex surface
- Need techniques to register multiple range images

… which brings us to *multi-view stereo*

Outline

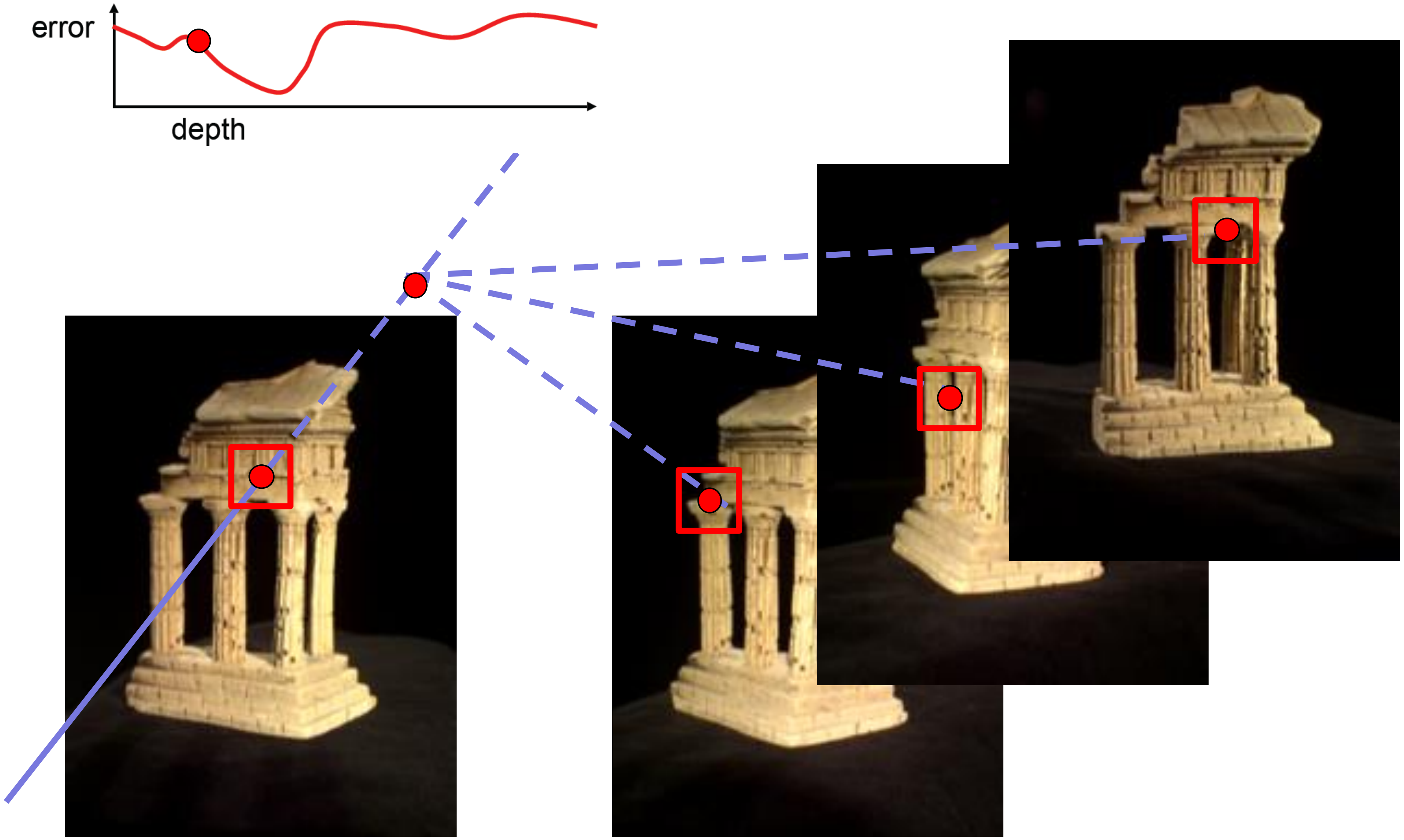
- Applications
- Two view stereo
- Multi-view stereo
  - Plane sweep stereo
  - Patch-based multi-view stereo
  - Depth map fusion
- Demonstration

Multi-view stereo: Basic idea



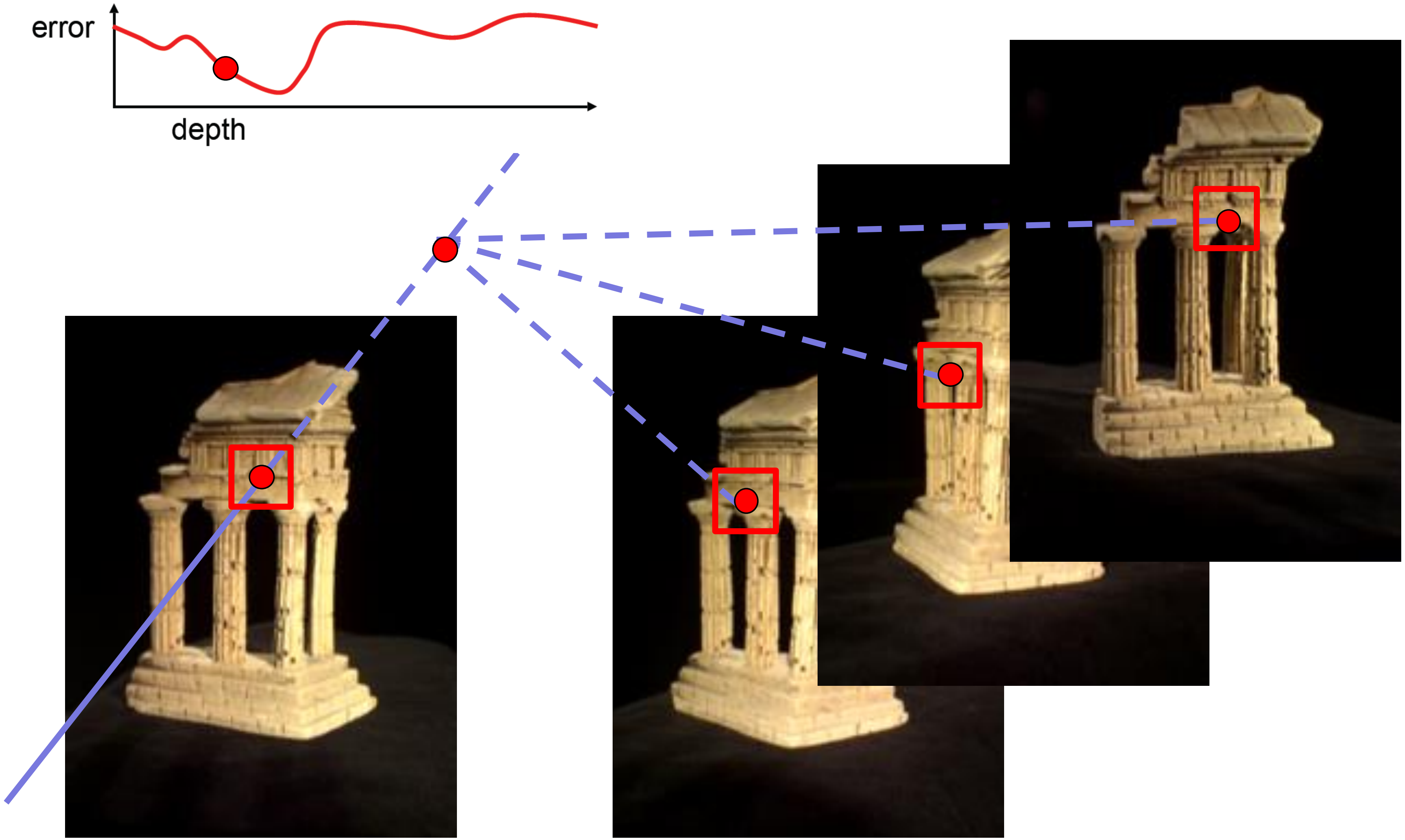
Source: Y. Furukawa

Multi-view stereo: Basic idea



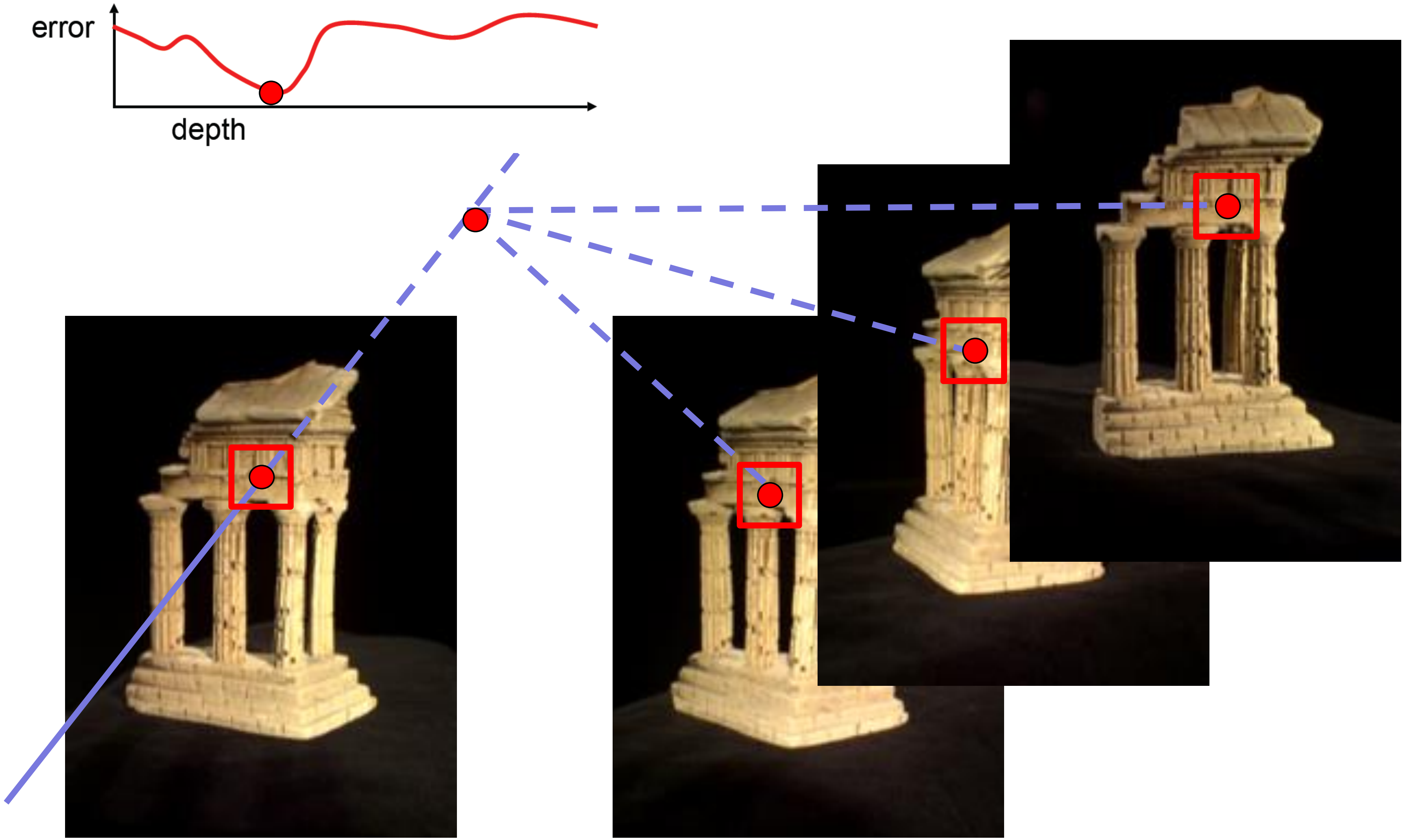
Source: Y. Furukawa

Multi-view stereo: Basic idea



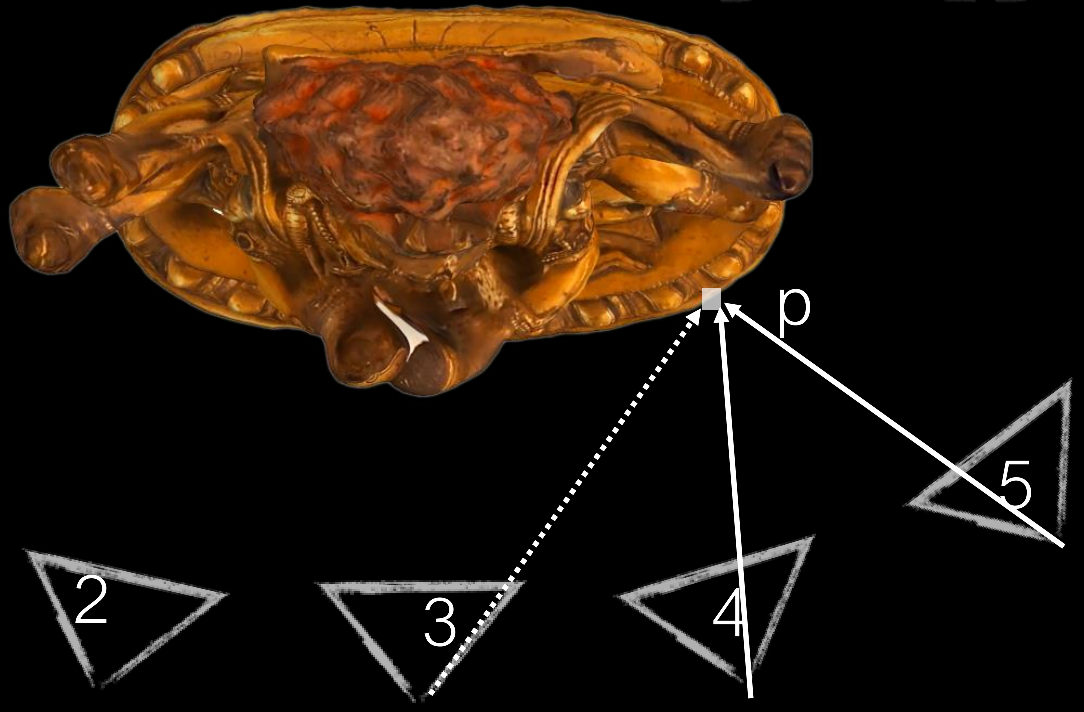
Source: Y. Furukawa

Multi-view stereo: Basic idea



Source: Y. Furukawa

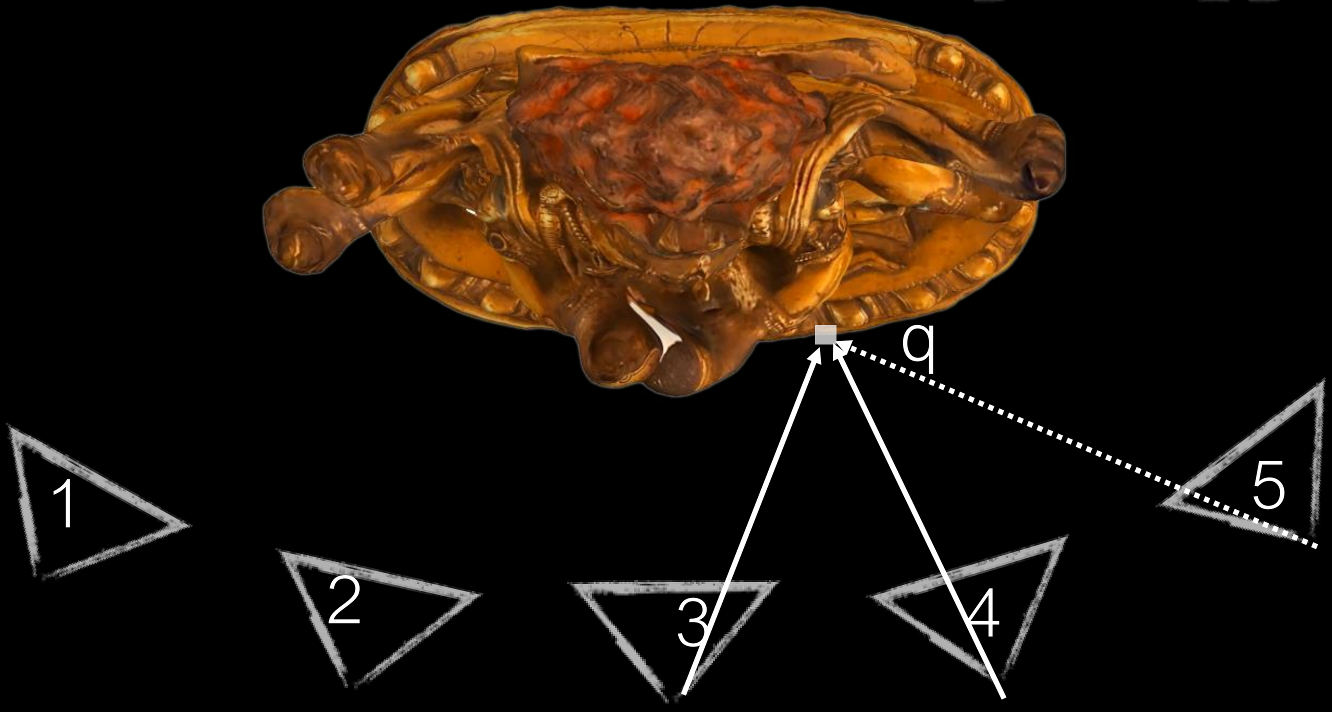
Some points occluded entirely in certain views



Cameras 4 and 5 can more clearly see point p

Source: N. Snavely

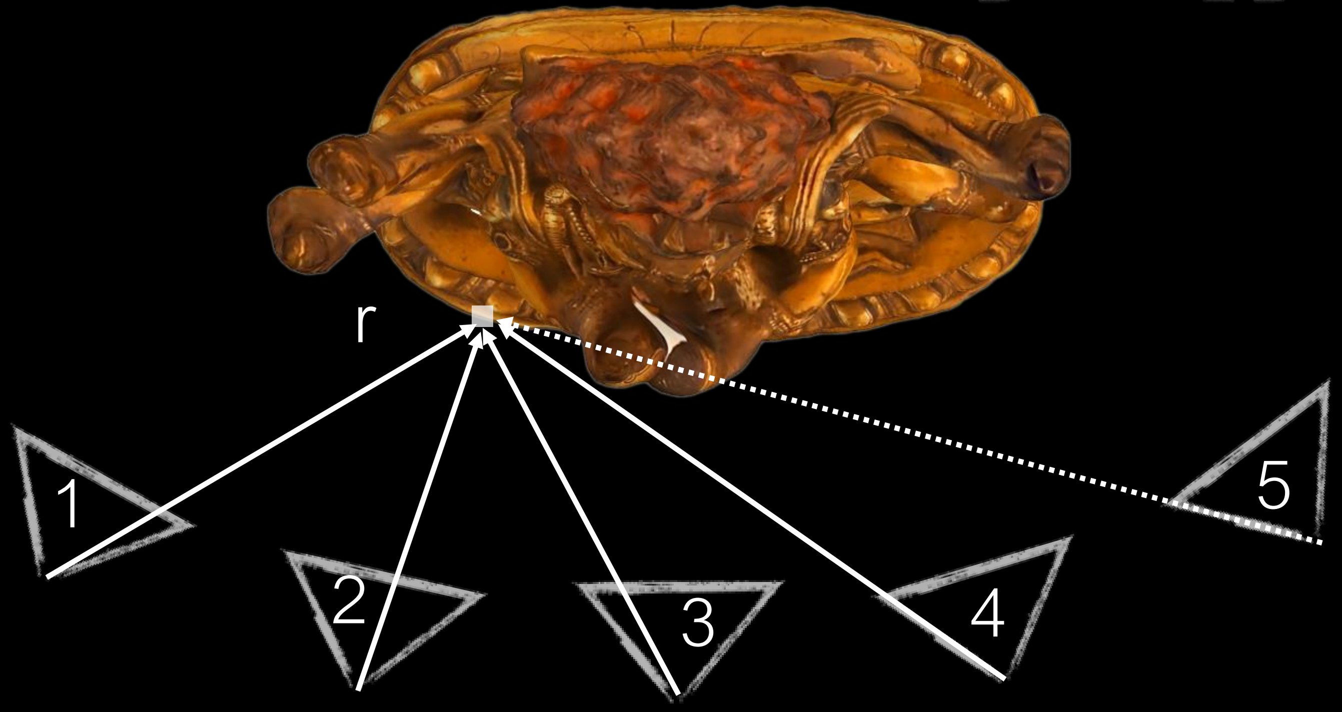
Some points occluded entirely in certain views



Cameras 3 and 4 can more clearly see point q

Source: N. Snavely

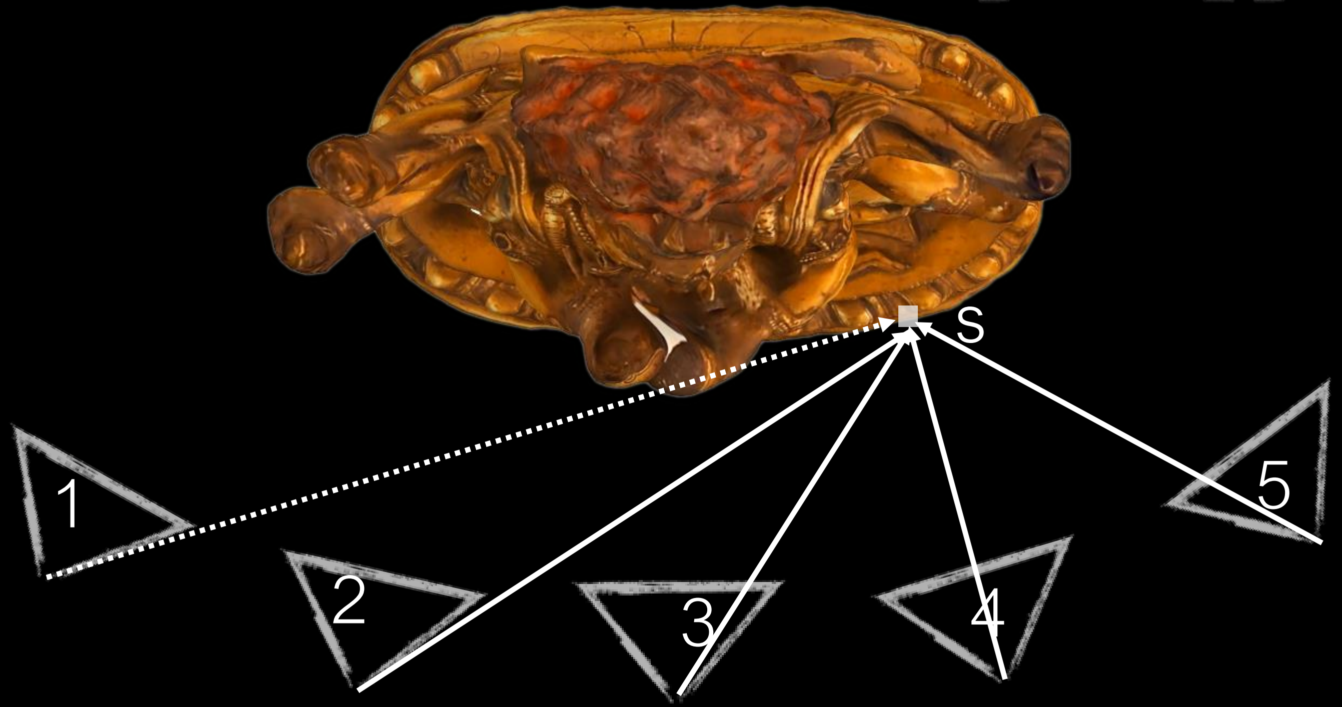
Some points occluded entirely in certain views



Camera 5 can’t see point r

Source: N. Snavely

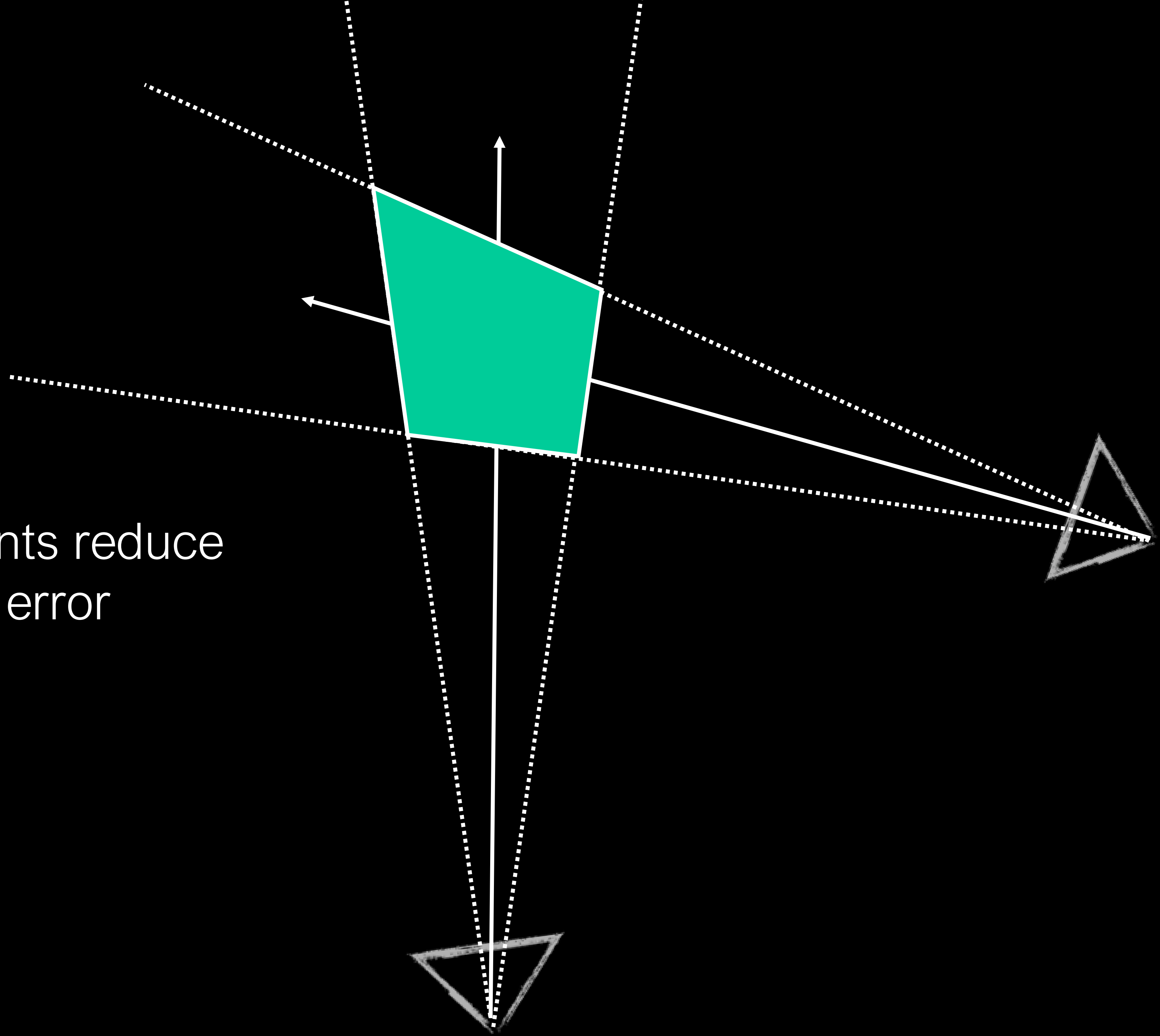
Some points occluded entirely in certain views



Camera 1 can't see point s

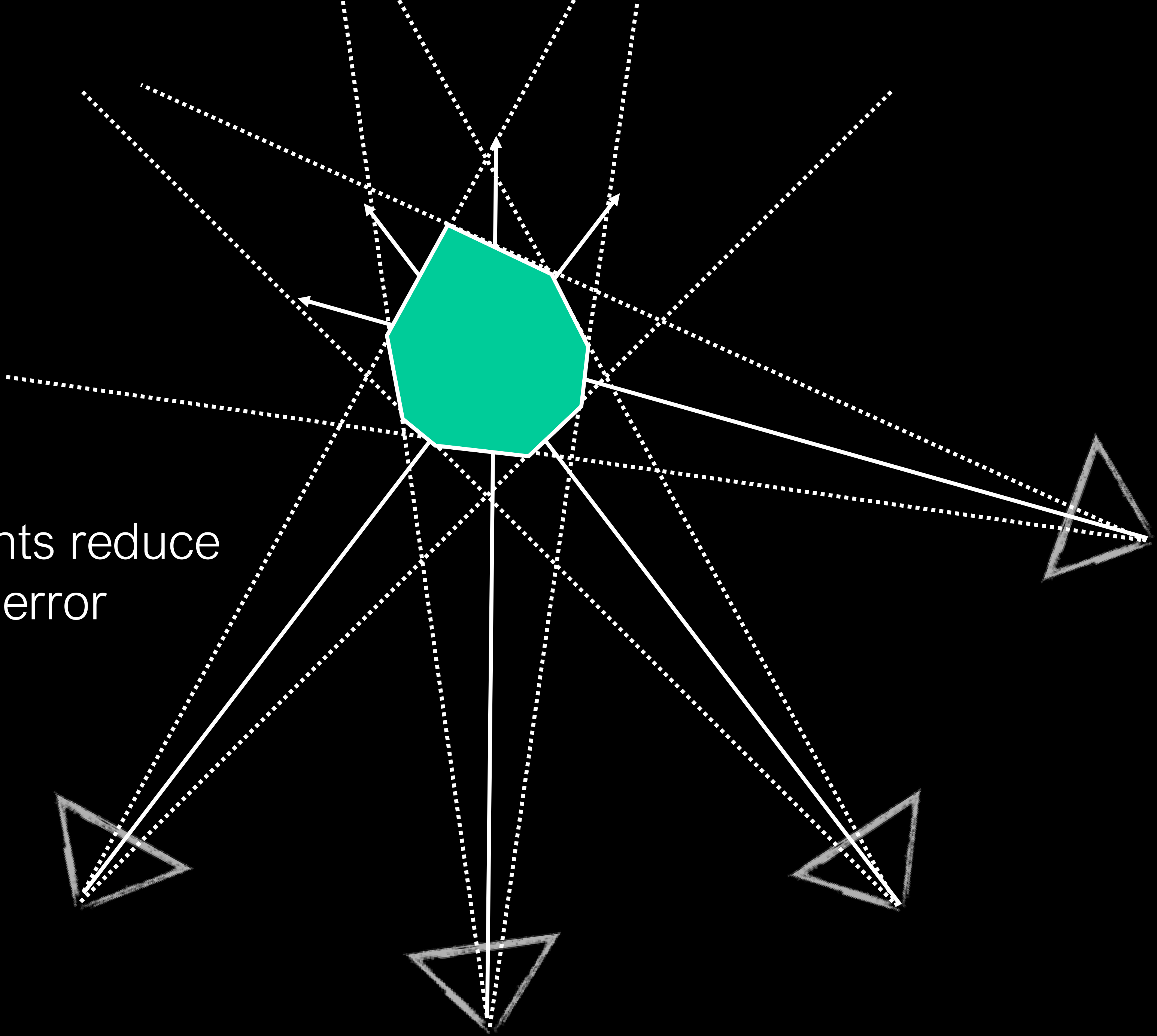
Source: N. Snavely

More measurements reduce triangulation error



Source: N. Snavely

More measurements reduce triangulation error



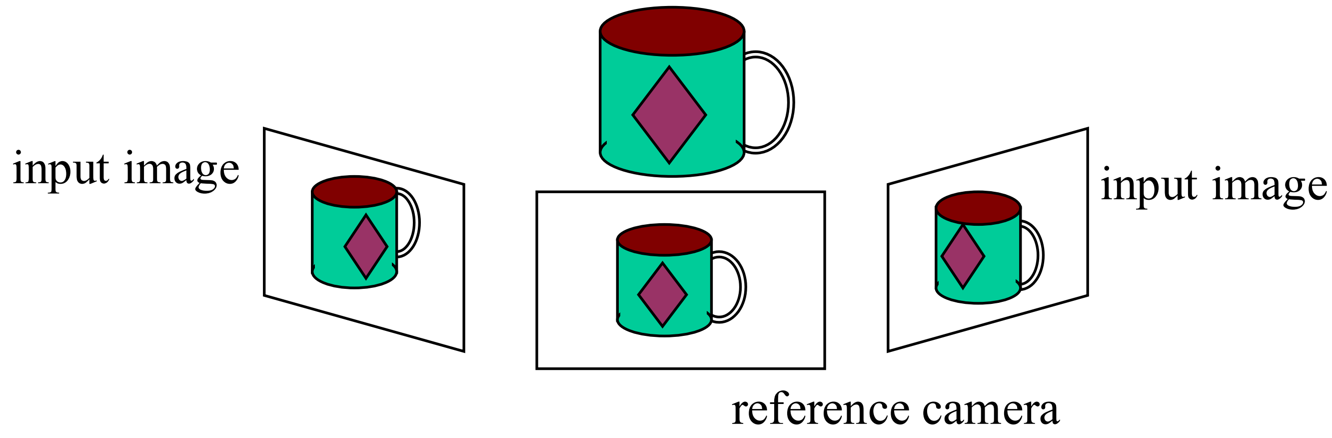
Source: N. Snavely

Outline

- Applications
- Two view stereo
- Multi-view stereo
  - Plane sweep stereo
  - Patch-based multi-view stereo
  - Depth map fusion
- Demonstration

Plane sweep stereo

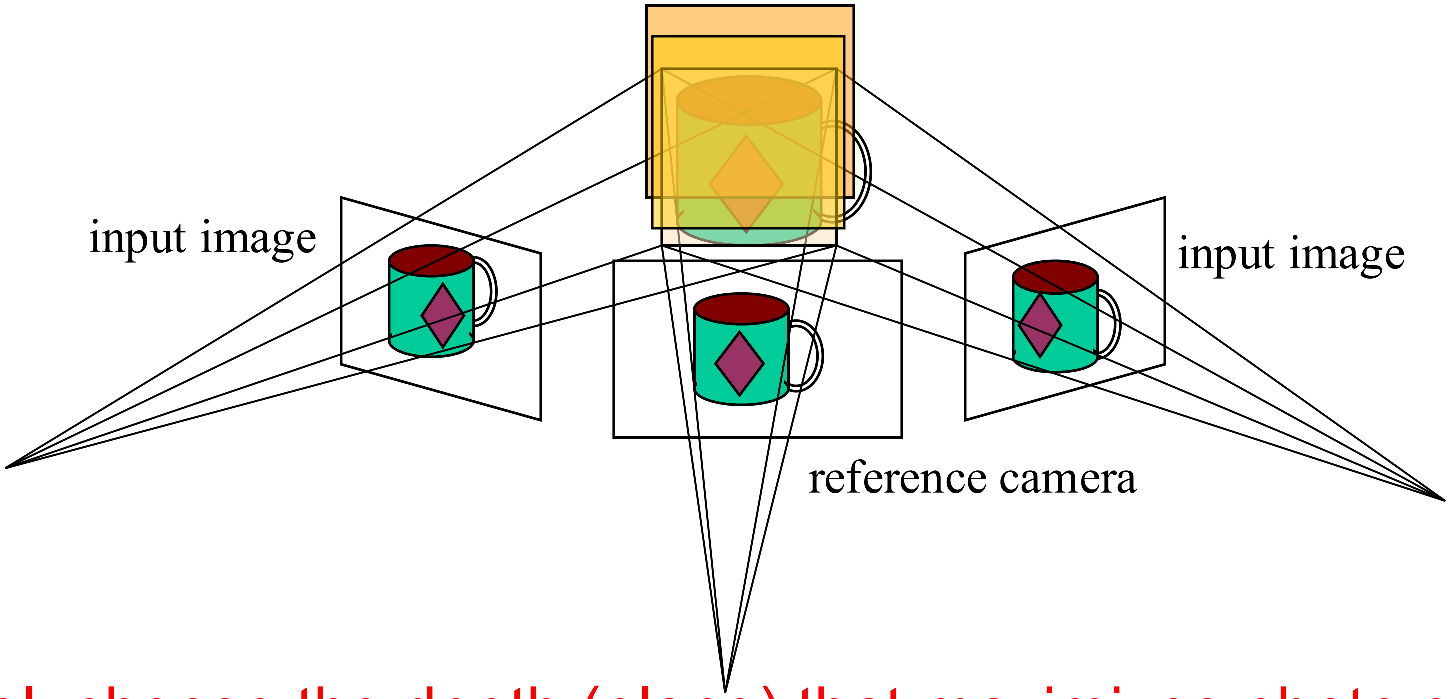
Goal: recover a **dense depth map** (per-pixel depth) rather than sparse feature points



R. Collins, A space-sweep approach to true multi-image matching, CVPR 1996

Plane sweep stereo

Idea: convert correspondence finding into structured search over depth hypotheses

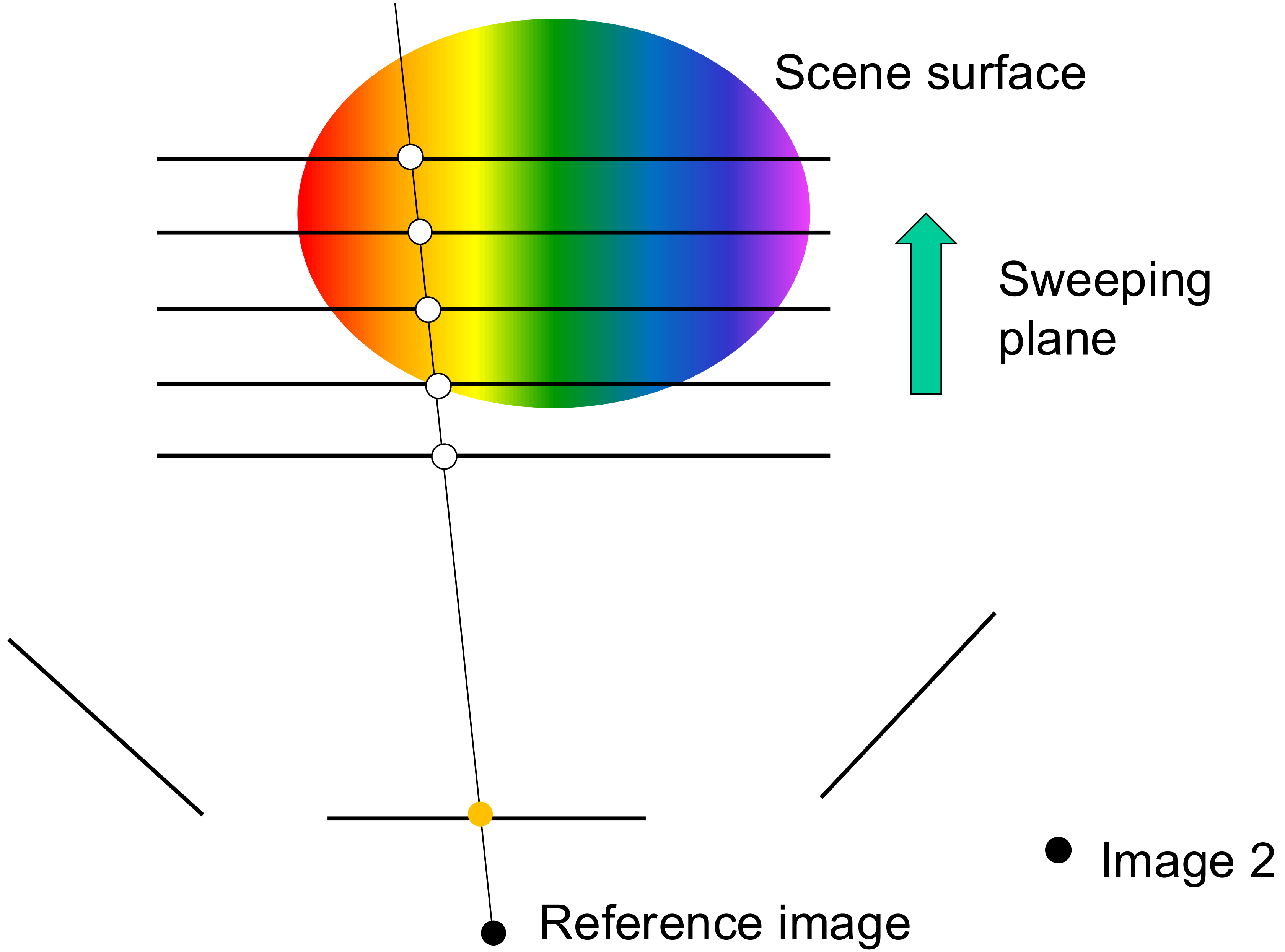


For each pixel, choose the depth (plane) that maximizes photo consistency

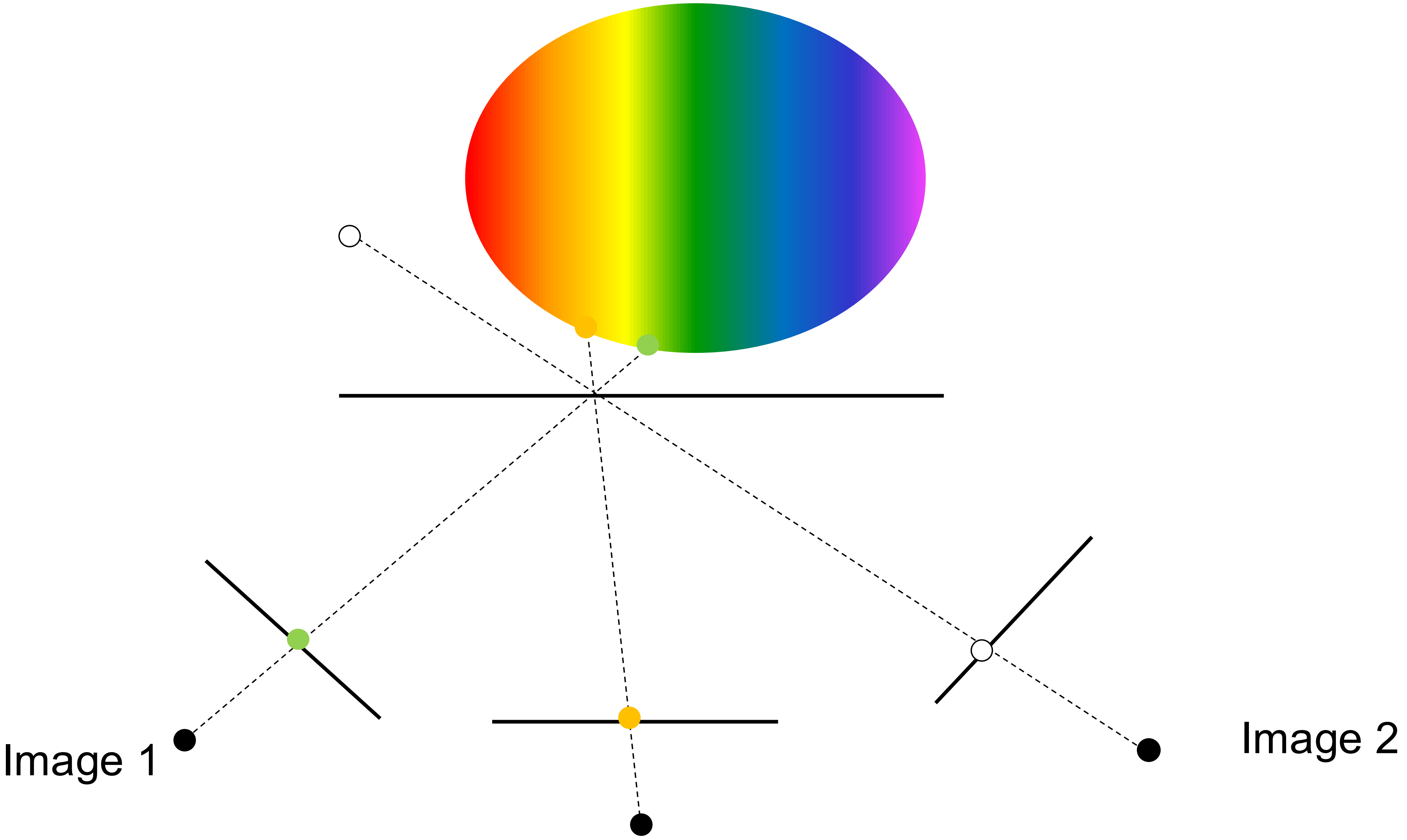
- Sweep plane across a range of depths w.r.t. a reference camera
- For each depth, project each input image onto that plane (homography) and compare the reprojections

R. Collins, A space-sweep approach to true multi-image matching, CVPR 1996

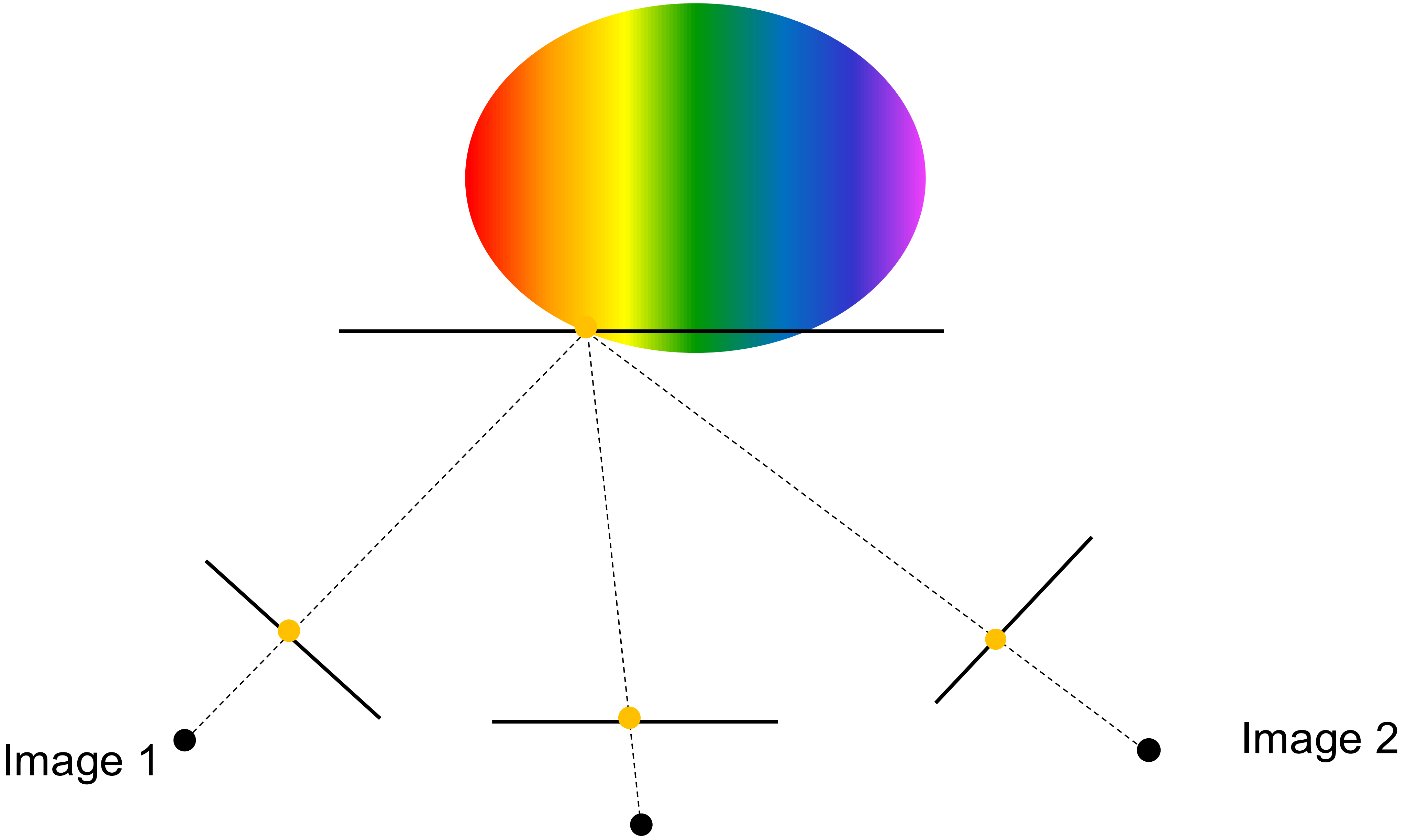
Plane sweep stereo: Key idea



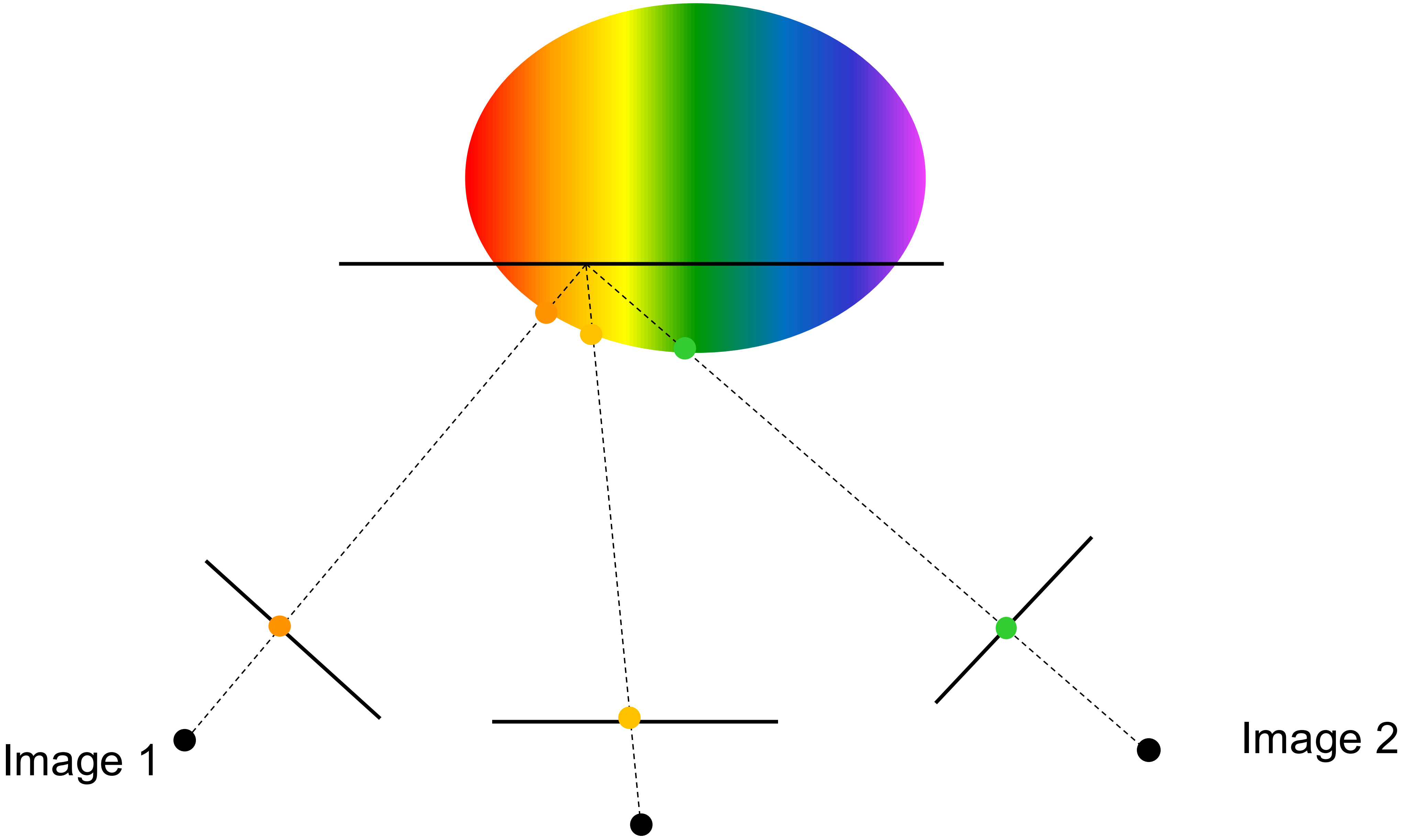
Plane sweep stereo: Key idea



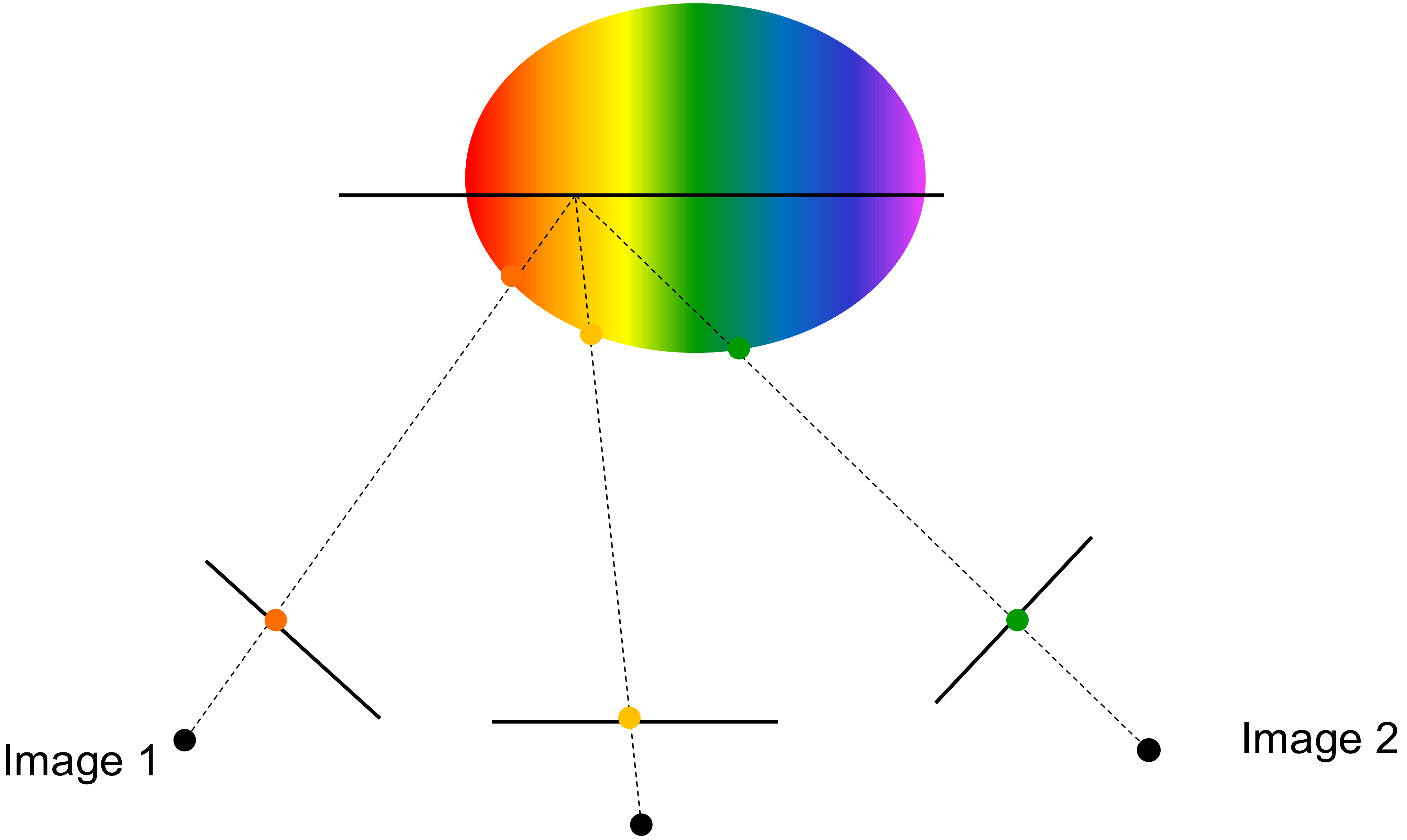
Plane sweep stereo: Key idea



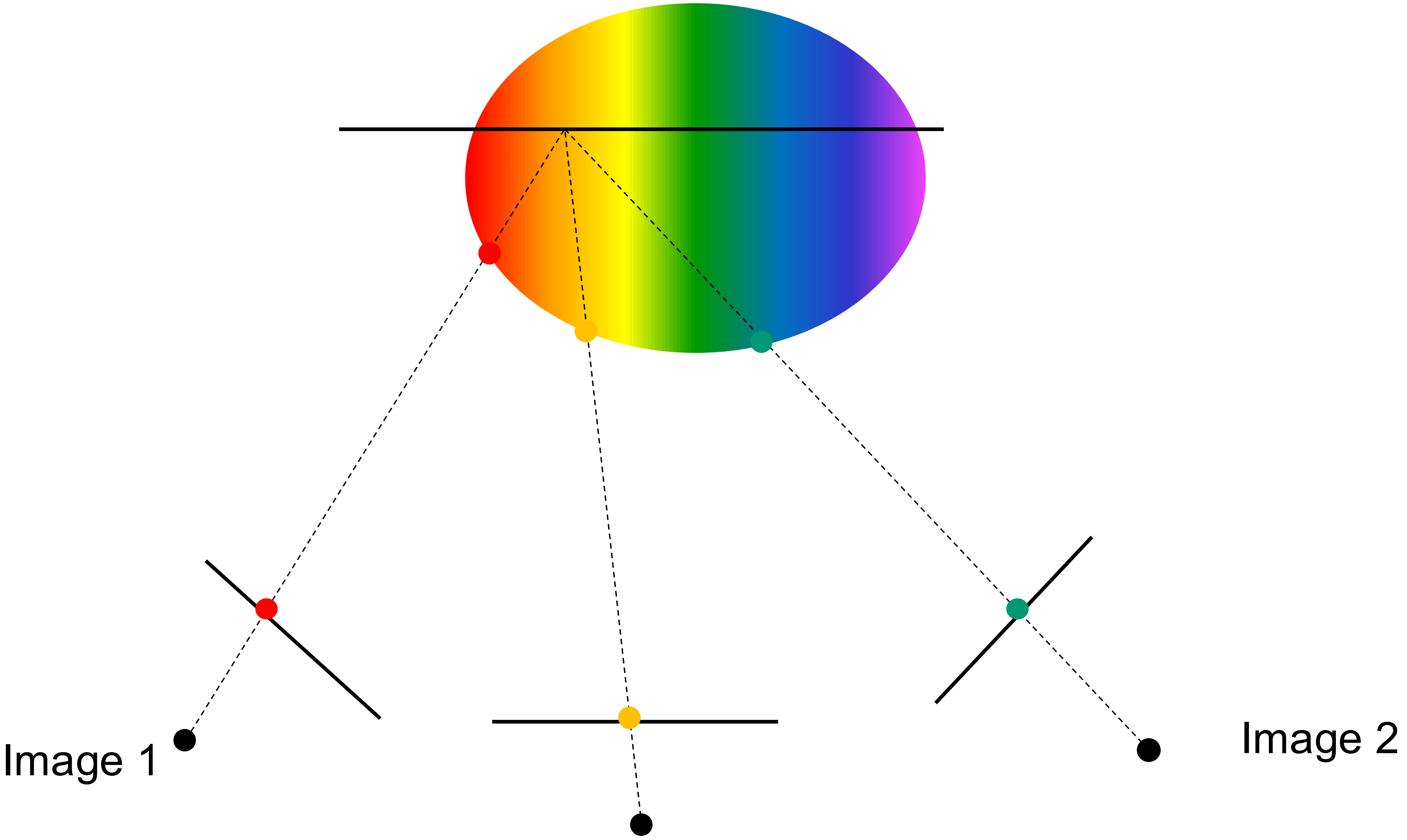
Plane sweep stereo: Key idea



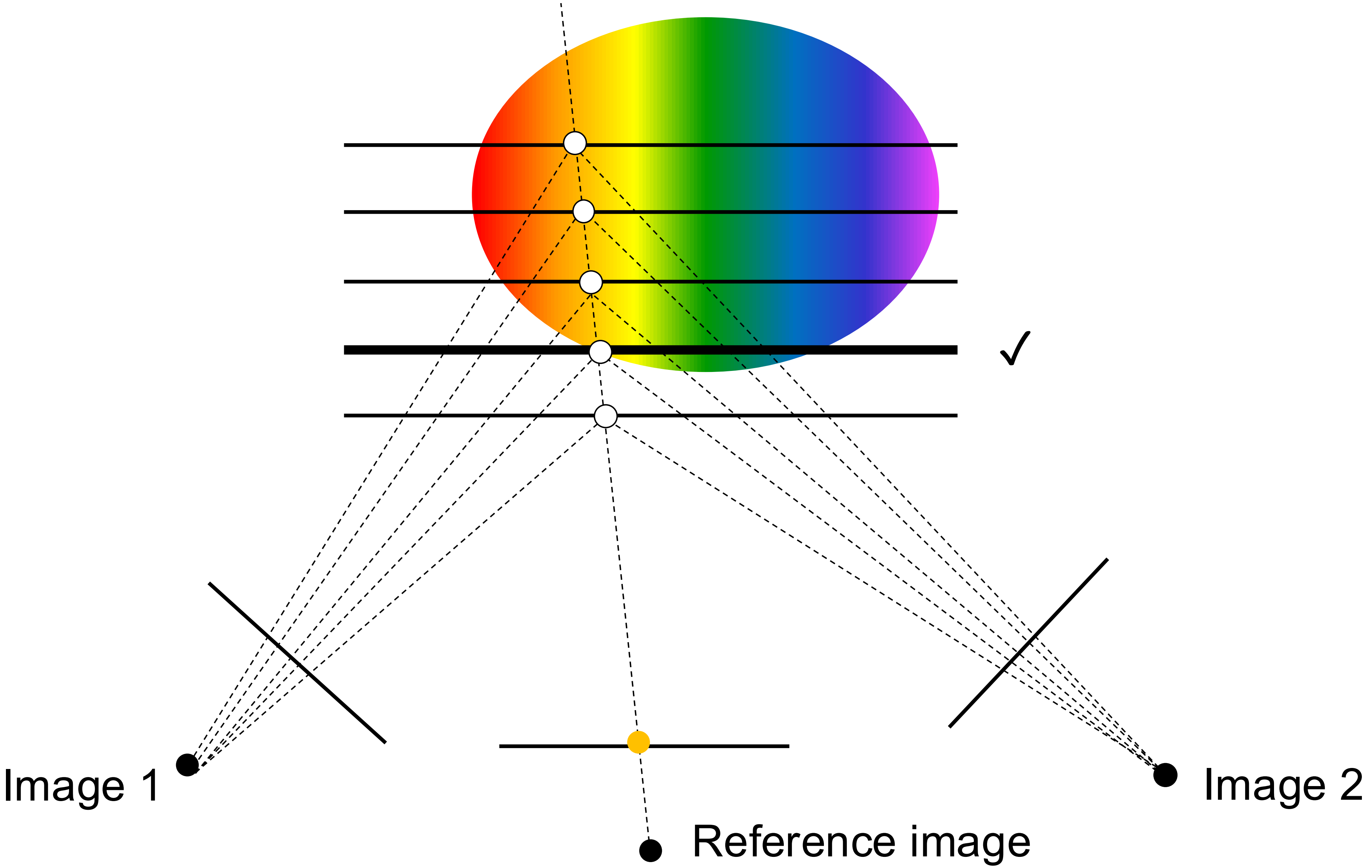
Plane sweep stereo: Key idea



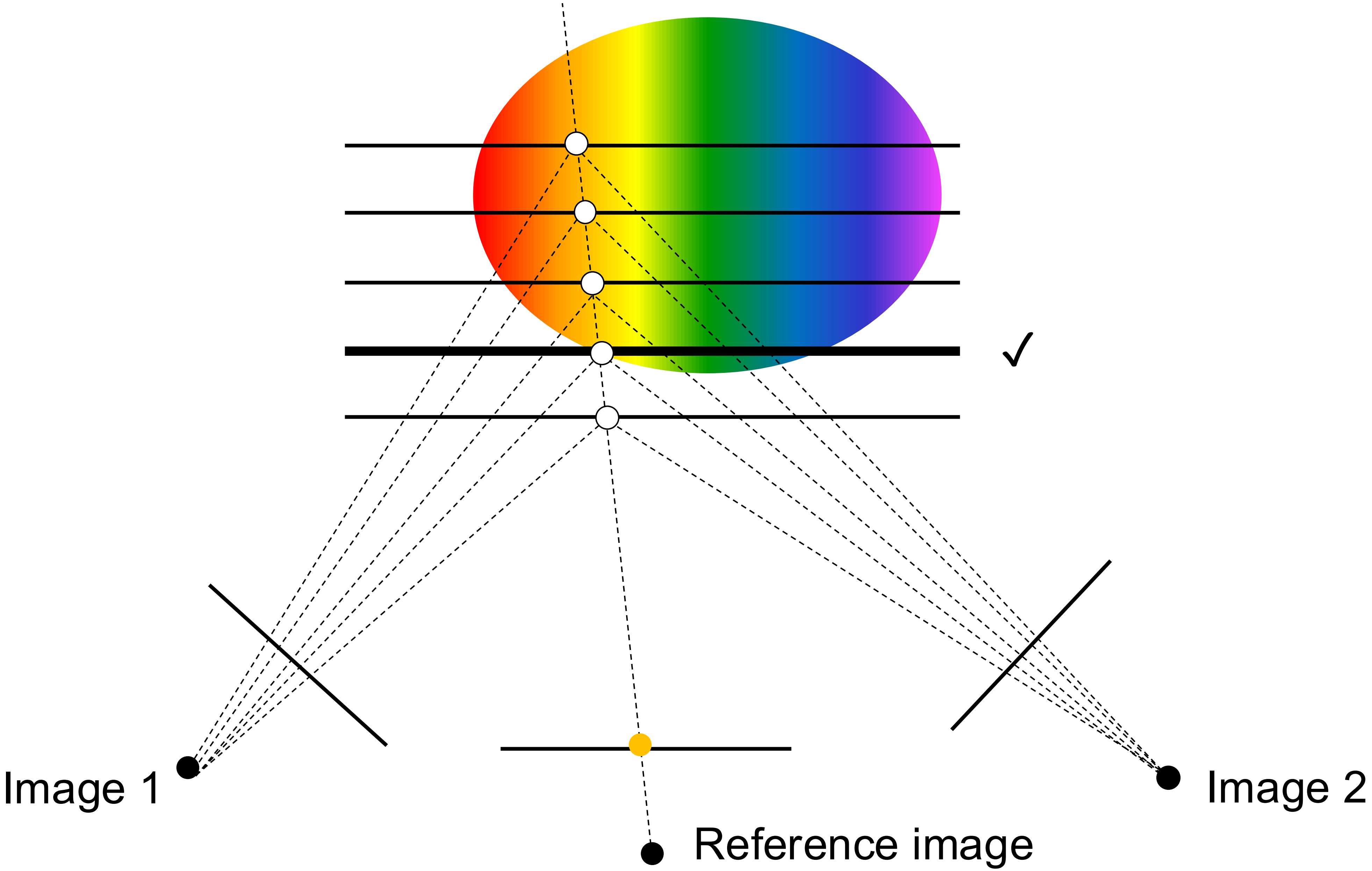
Plane sweep stereo: Key idea



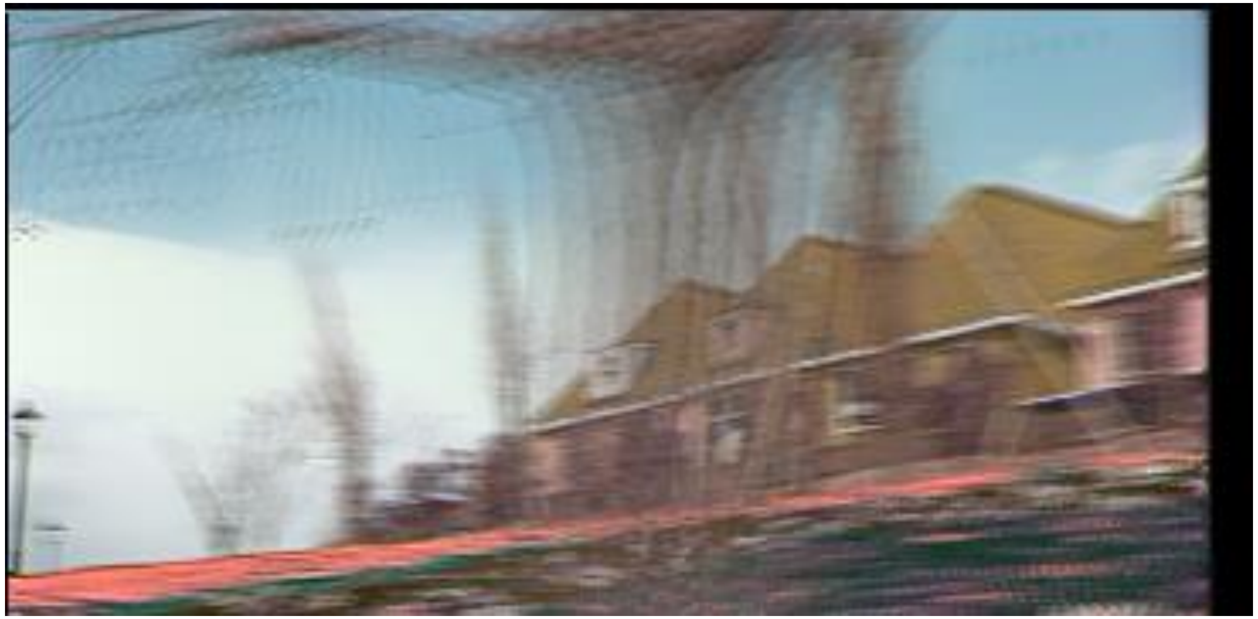
Plane sweep stereo: Key idea



Does this always work?



Plane sweep stereo: Fast implementation



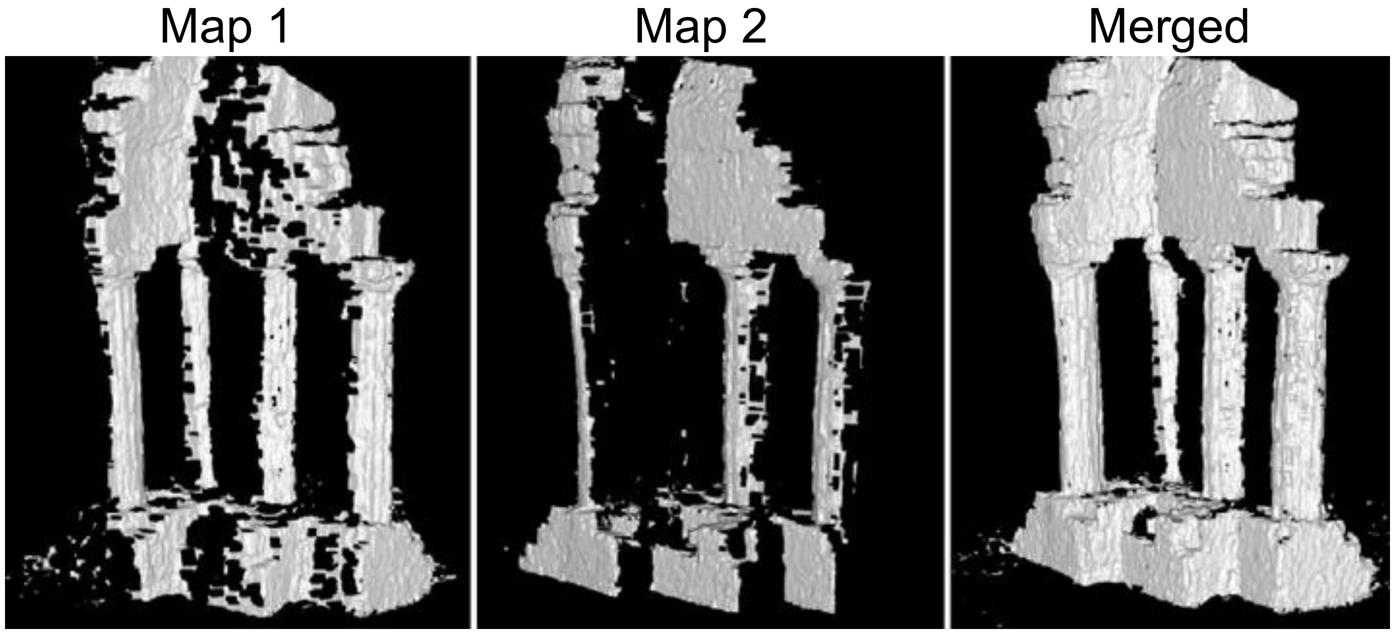
depth 1
depth 2
depth 3
depth 4
depth 5
depth 6

- For each depth plane
  - Compute homographies projecting each image onto that depth plane
  - For each pixel in the composite image stack, compute the variance
- For each pixel, select the depth that gives the lowest variance

R. Yang and M. Pollefeys, Multi-Resolution Real-Time Stereo on Commodity Graphics Hardware, CVPR 2003

Merging depth maps

- Given a group of images, compute a depth map using each view as a reference
- Merge multiple depth maps into a volume or a mesh (see, e.g., Curless and Levoy, 1996)



Outline

- Applications
- Two view stereo
- Multi-view stereo
  - Plane sweep stereo
  - Patch-based multi-view stereo
  - Depth map fusion
- Demonstration

Patch-based multi-view stereo (PMVS)

1. Detect keypoints
2. Triangulate a sparse set of initial matches
3. Iteratively expand matches to nearby locations
4. Use visibility constraints to filter out false matches
5. Perform surface reconstruction

Y. Furukawa and J. Ponce, Accurate, Dense, and Robust Multi-View Stereopsis, CVPR 2007.
PMVS software

Demonstration

Y. Furukawa and J. Ponce, Accurate, Dense, and Robust Multi-View Stereopsis, CVPR 2007.
PMVS software

Towards Internet-scale multi-view stereo

YouTube video, CMVS software

Y. Furukawa, B. Curless, S. Seitz and R. Szeliski, Towards Internet-scale Multi-view Stereo, CVPR 2010

Mesh model reconstructed from point cloud

YouTube video

More technical details: https://www.youtube.com/watch?v=OpZs7kfjFPA

Next lecture: Surface Reconstruction