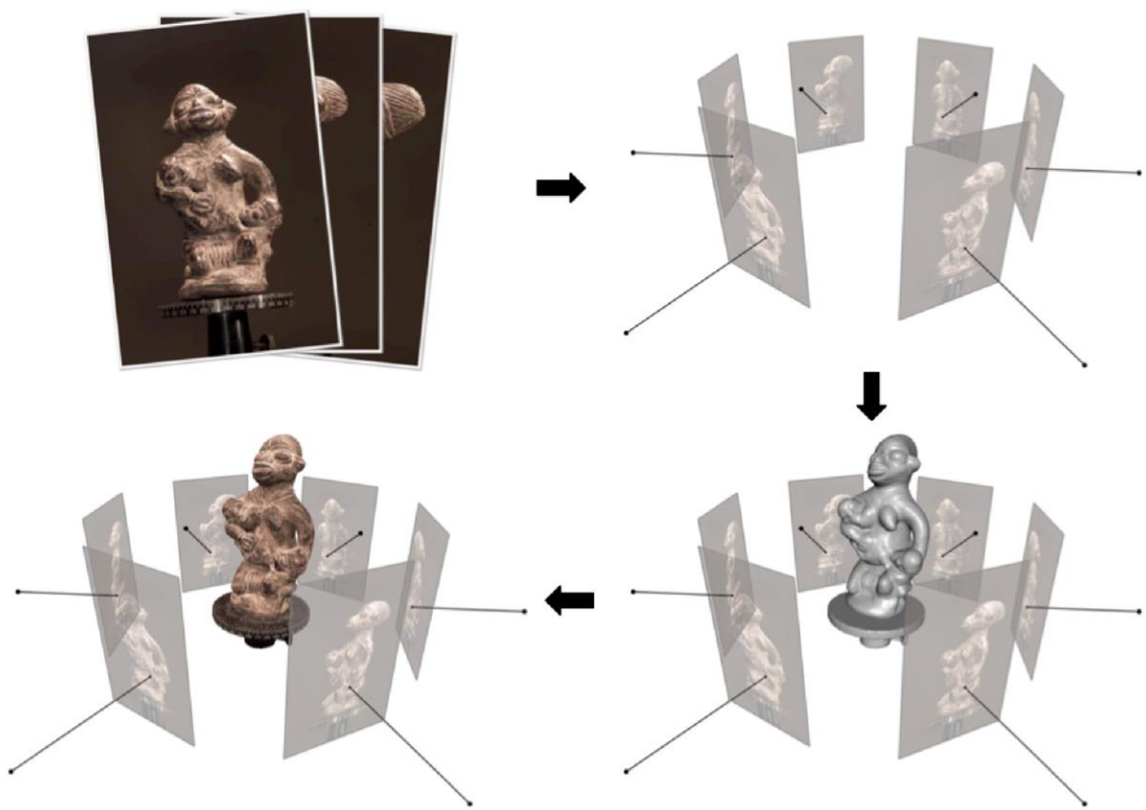
Image Based Reconstruction II

Qixing Huang

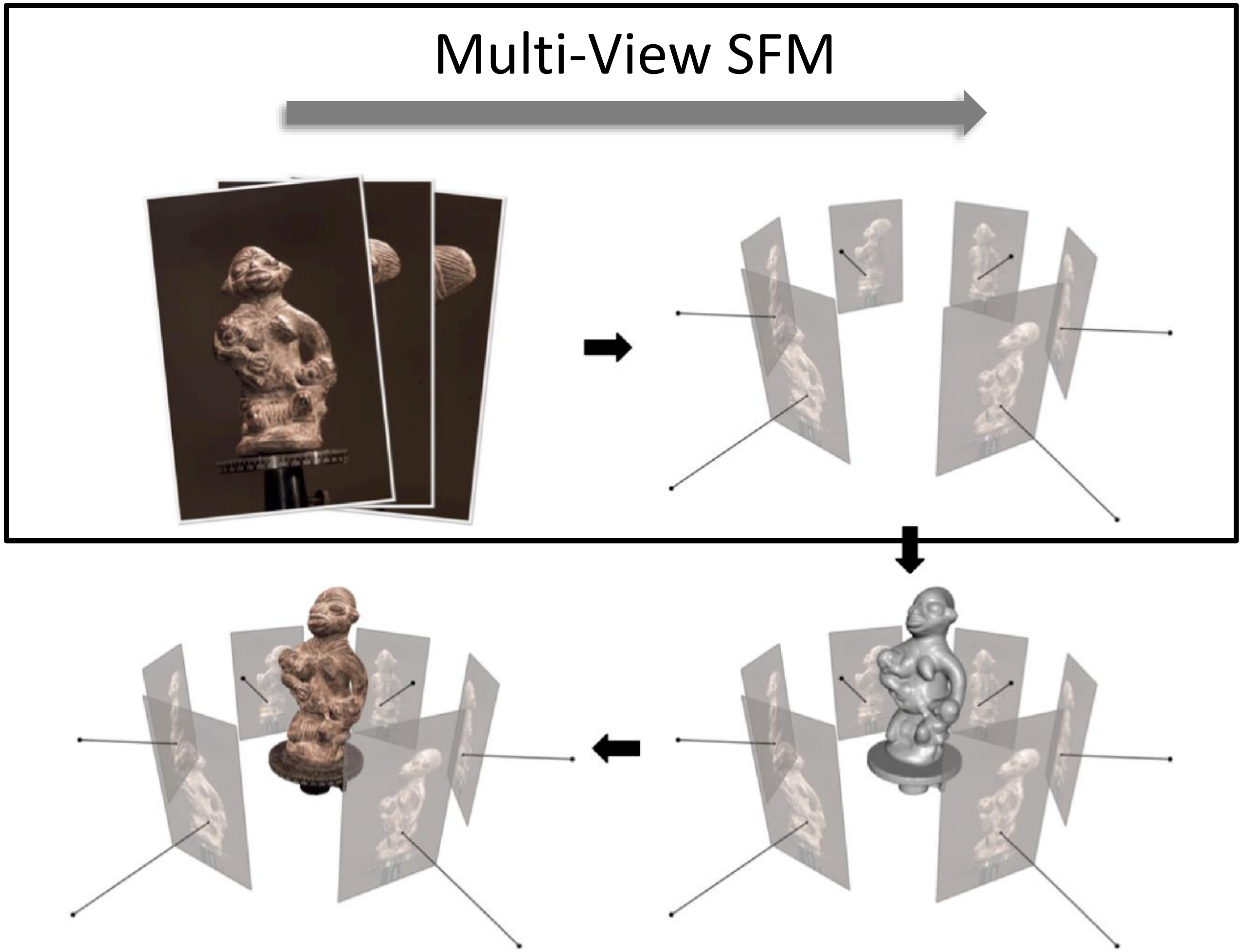
Feb. 2th 2017

Slide Credit: Yasutaka Furukawa

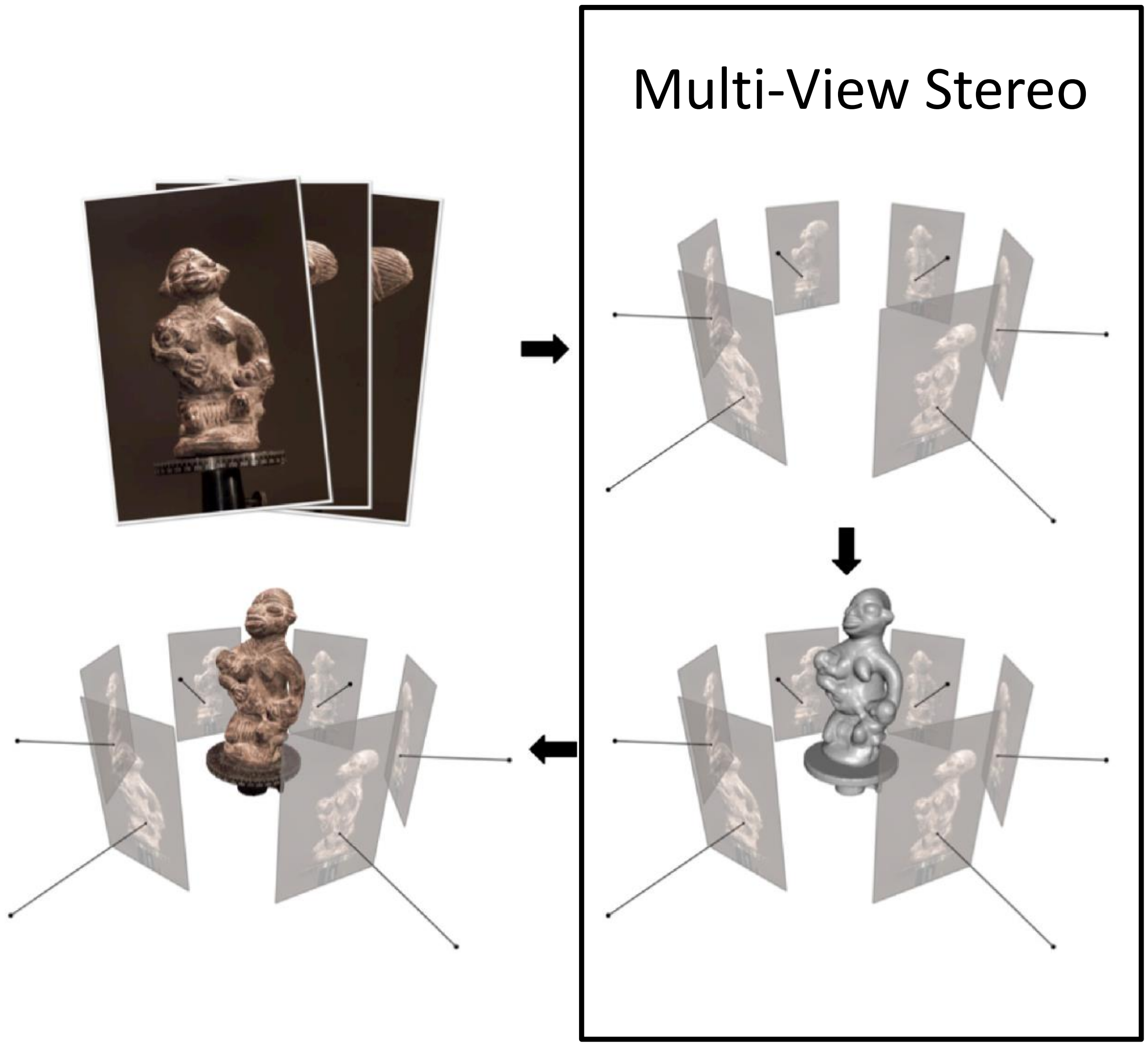
Image-Based Geometry Reconstruction Pipeline



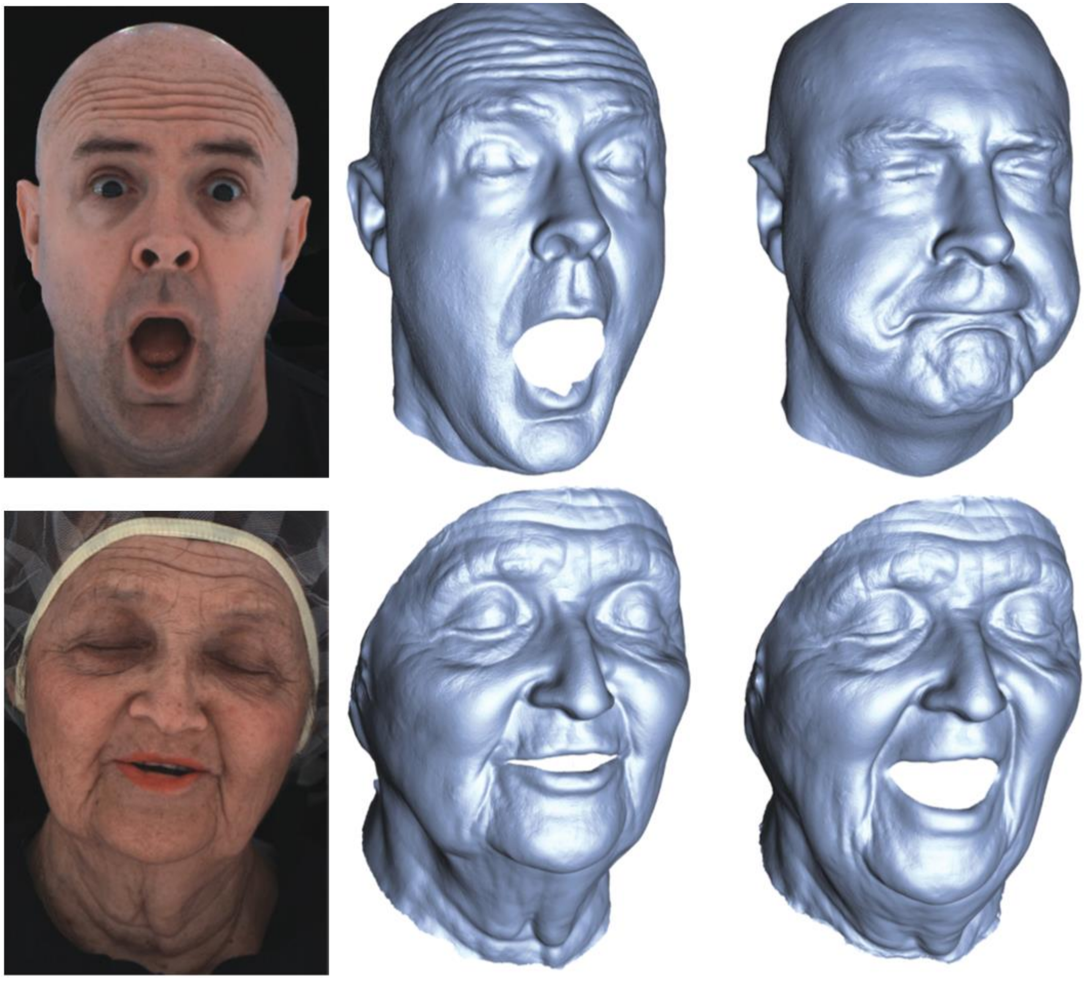
Last Lecture: Multi-View SFM



This Lecture: Multi-View Stereo



Multi-view Stereo for Visual Effects



Input Images

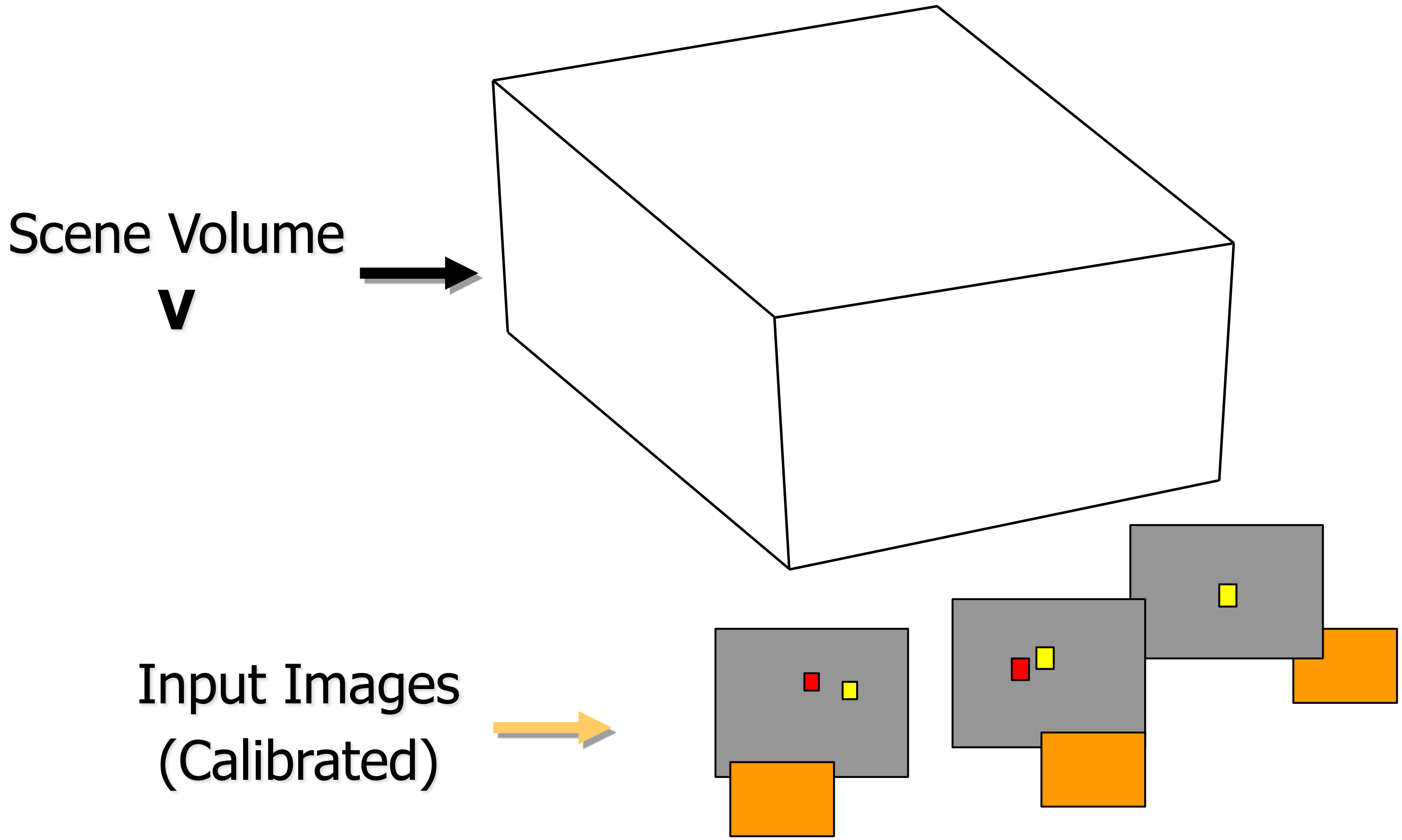
Fig. 1.3 Different MVS capture setups. From left to right: a controlled MVS capture using diffuse lights and a turn table, outdoor capture of small-scale scenes, and crowd-sourcing from online photo-sharing websites.

Volumetric Stereo

Space Carving

Multi-Baseline Stereo

Volumetric Stereo

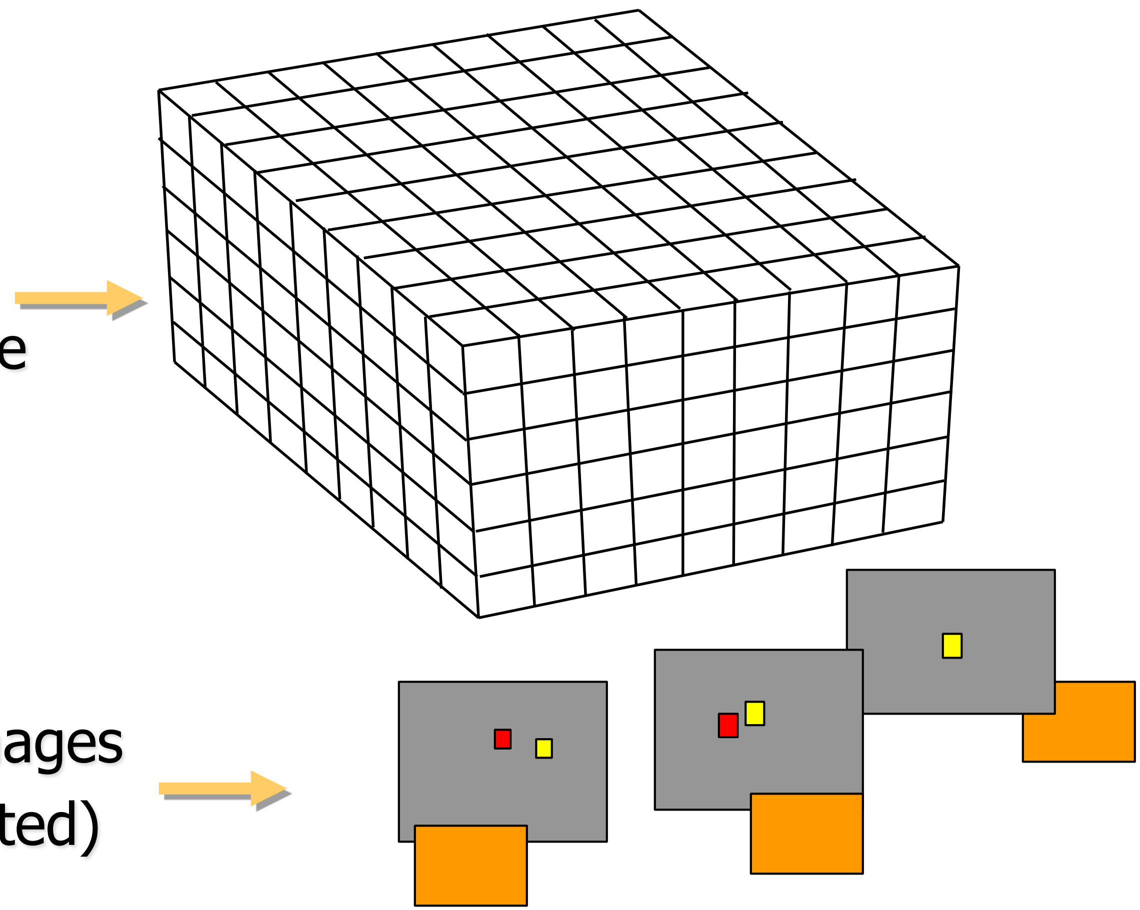


Goal: **Determine occupancy, "color" of points in V**

Discrete formulation: Voxel Coloring

Discretized
Scene Volume

Input Images
(Calibrated)



Goal: **Assign RGBA values to voxels in V**
photo-consistent **with images**

Voxel Coloring Solutions

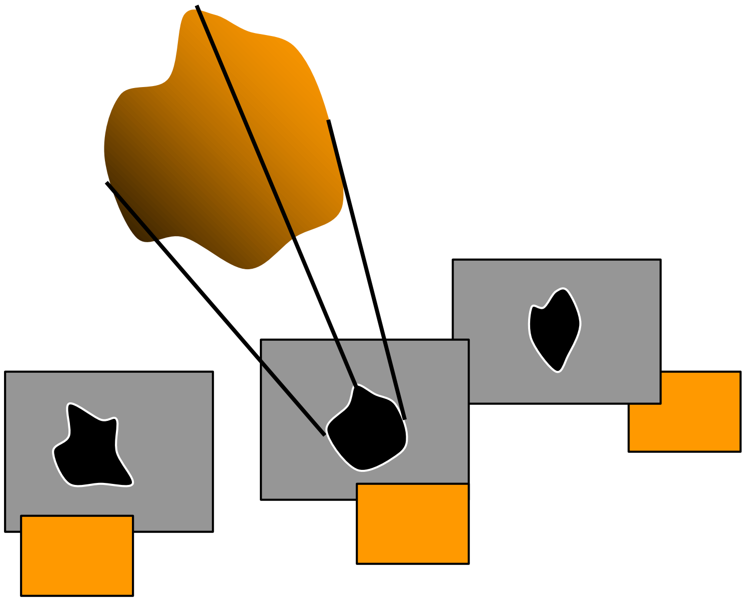
1. C=2 (shape from silhouettes)
   - Volume intersection [Baumgart 1974]

2. C unconstrained, viewpoint constraints
   - Voxel coloring algorithm [Seitz & Dyer 97]

3. General Case
   - Space carving [Kutulakos & Seitz 98]

Reconstruction from Silhouettes (C = 2)

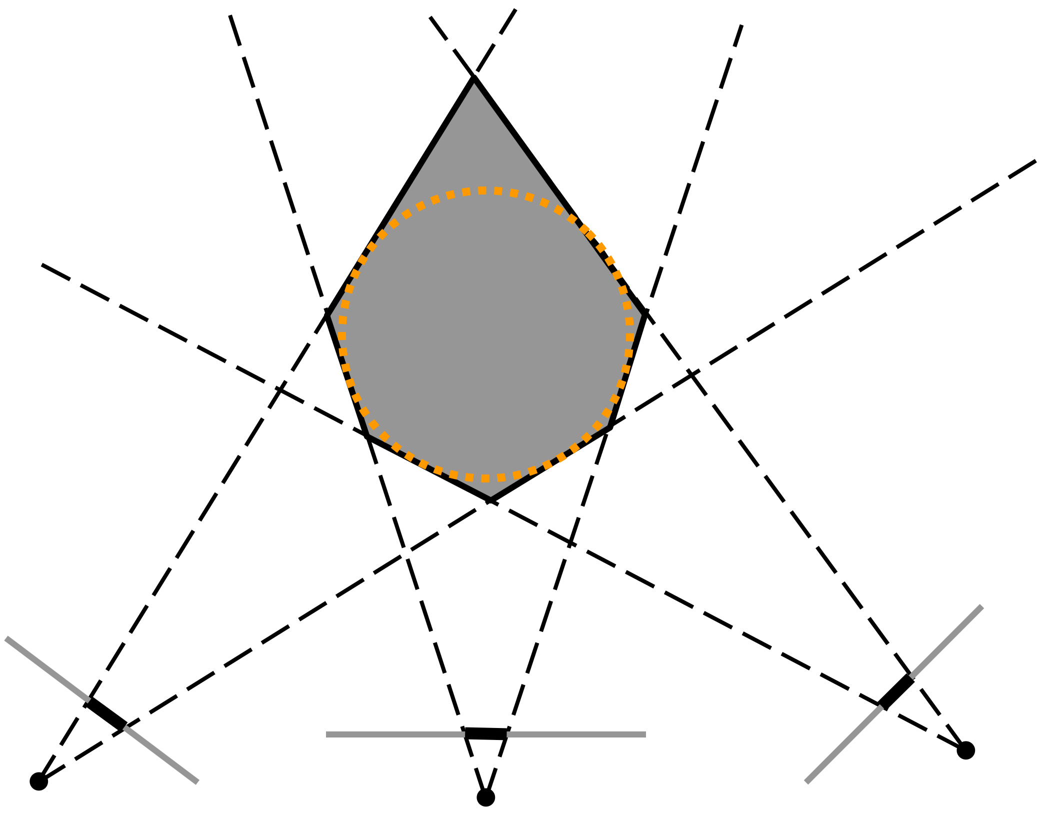
Binary Images →



Approach:

- *Backproject* each silhouette
- Intersect backprojected volumes

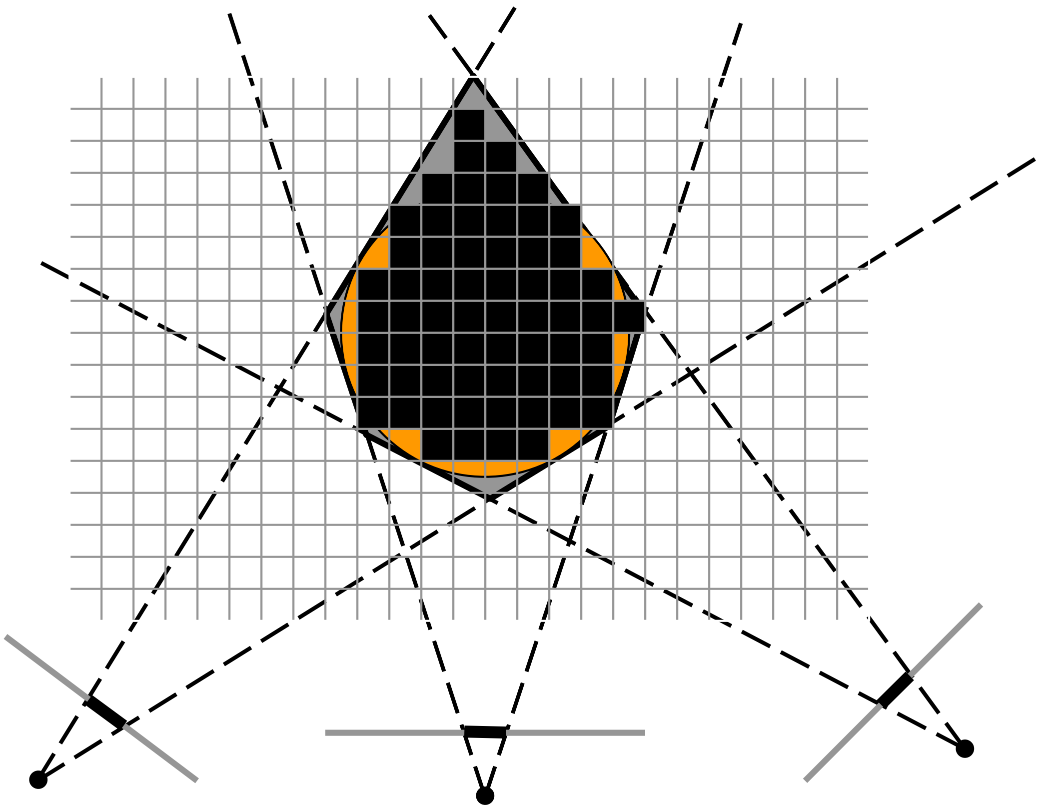
Volume Intersection



Reconstruction Contains the True Scene

- In the limit (all views) get *convex hull*

Voxel Algorithm for Volume Intersection



Color voxel black if on silhouette in every image

Properties of Volume Intersection

- Pros
  - Easy to implement, fast
  - Accelerated via octrees [Szeliski 1993] or interval techniques [Matusik 2000]
- Cons
  - No concavities
  - Reconstruction is not photo-consistent
  - Requires identification of silhouettes

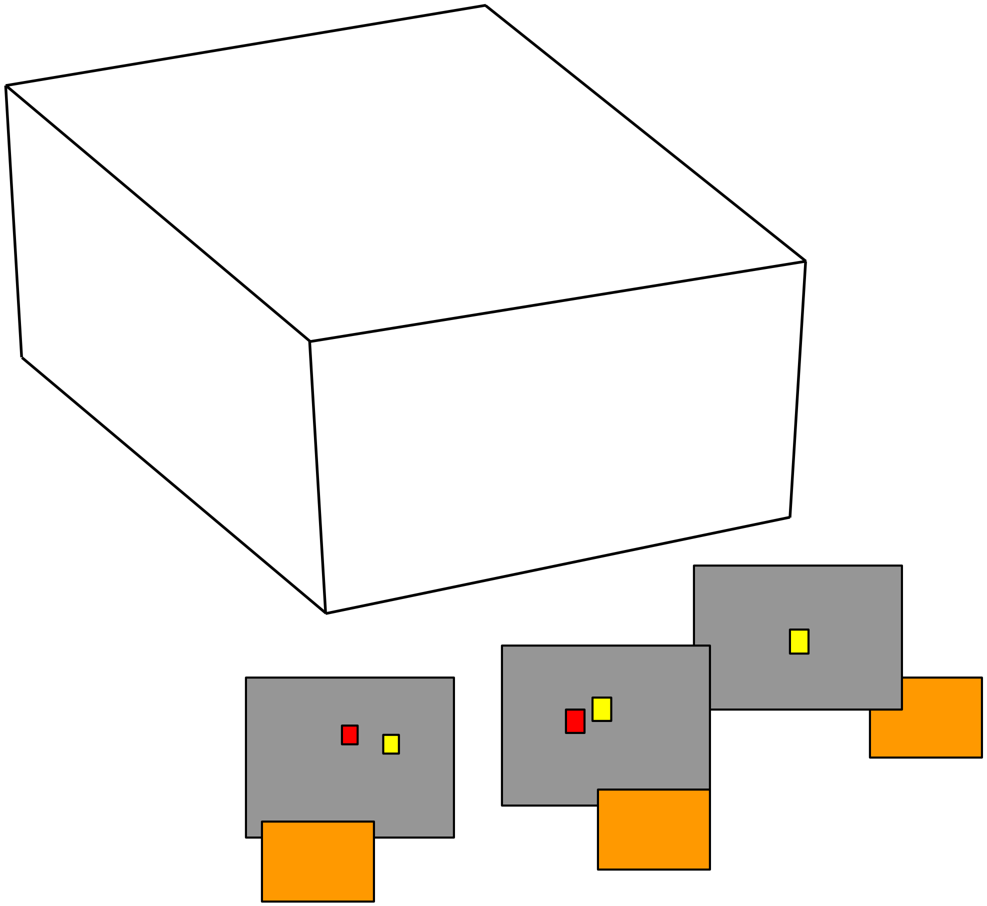
Voxel Coloring Solutions

1. C=2 (shape from silhouettes)
   - Volume intersection [Baumgart 1974]

2. C unconstrained, viewpoint constraints
   - Voxel coloring algorithm [Seitz & Dyer 97]

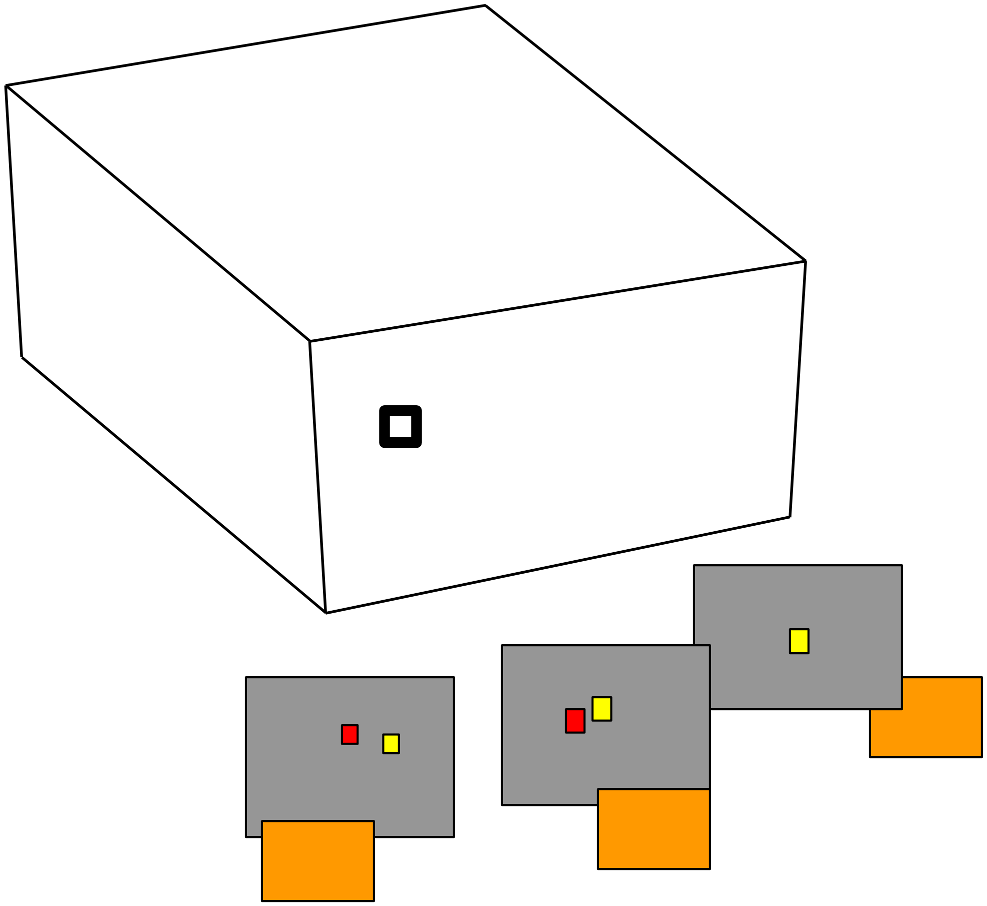
3. General Case
   - Space carving [Kutulakos & Seitz 98]

Voxel Coloring Approach



Voxel Coloring Approach

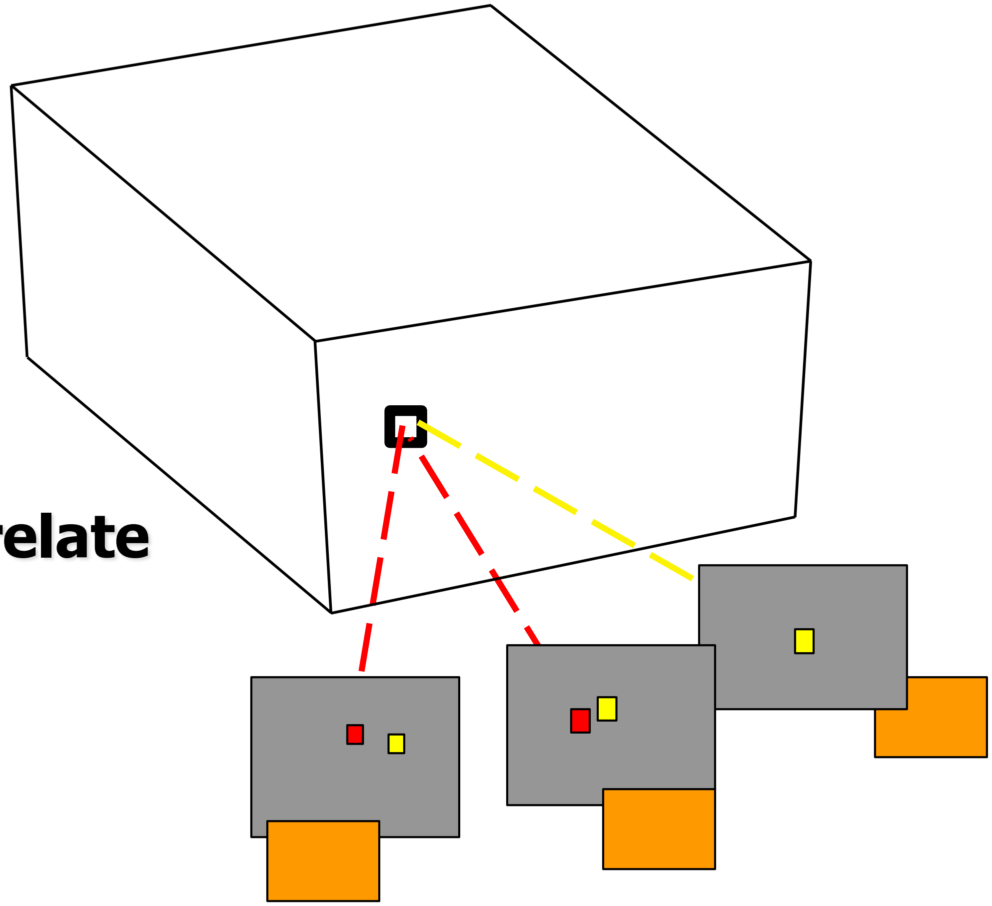
1. Choose voxel



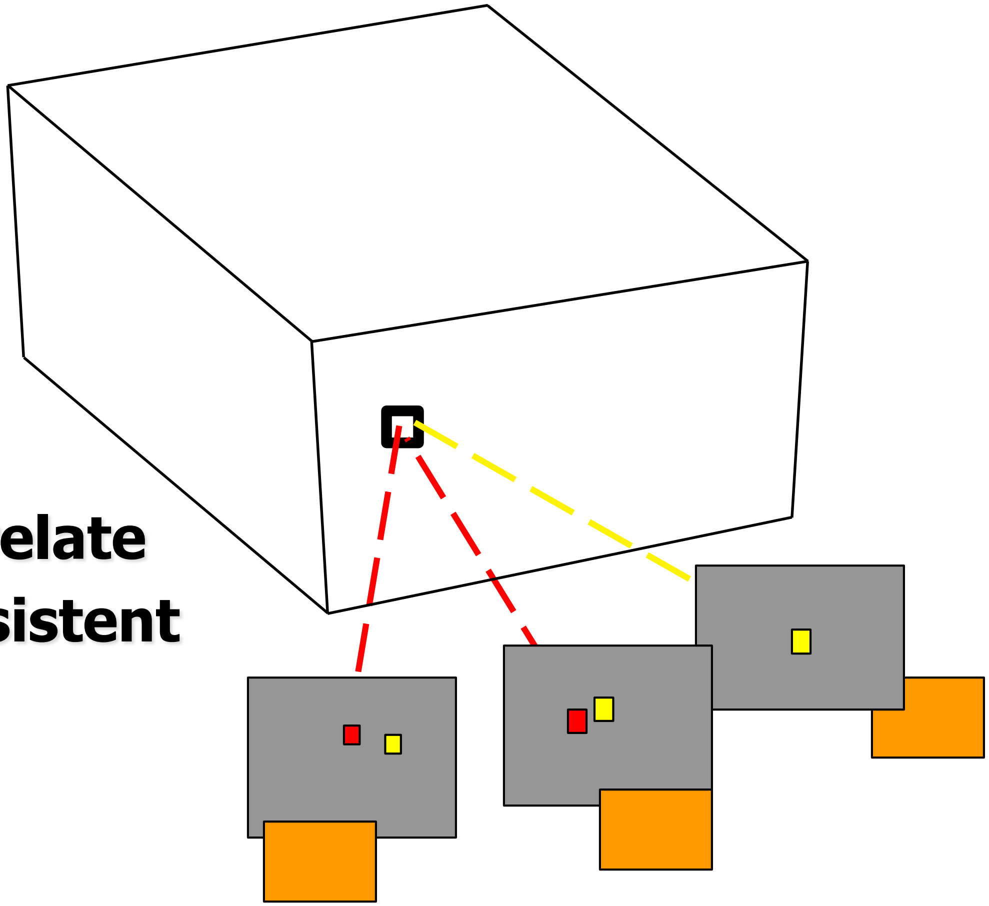
Voxel Coloring Approach

1. Choose voxel

2. Project and correlate

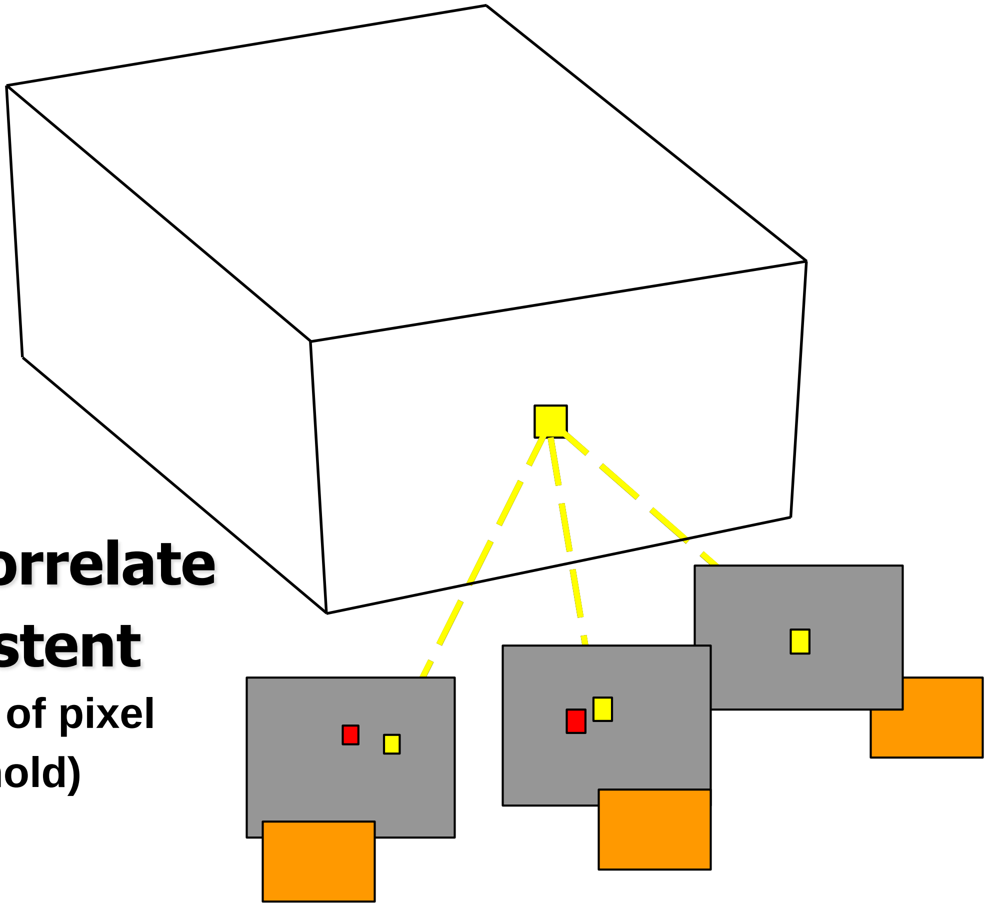


Voxel Coloring Approach



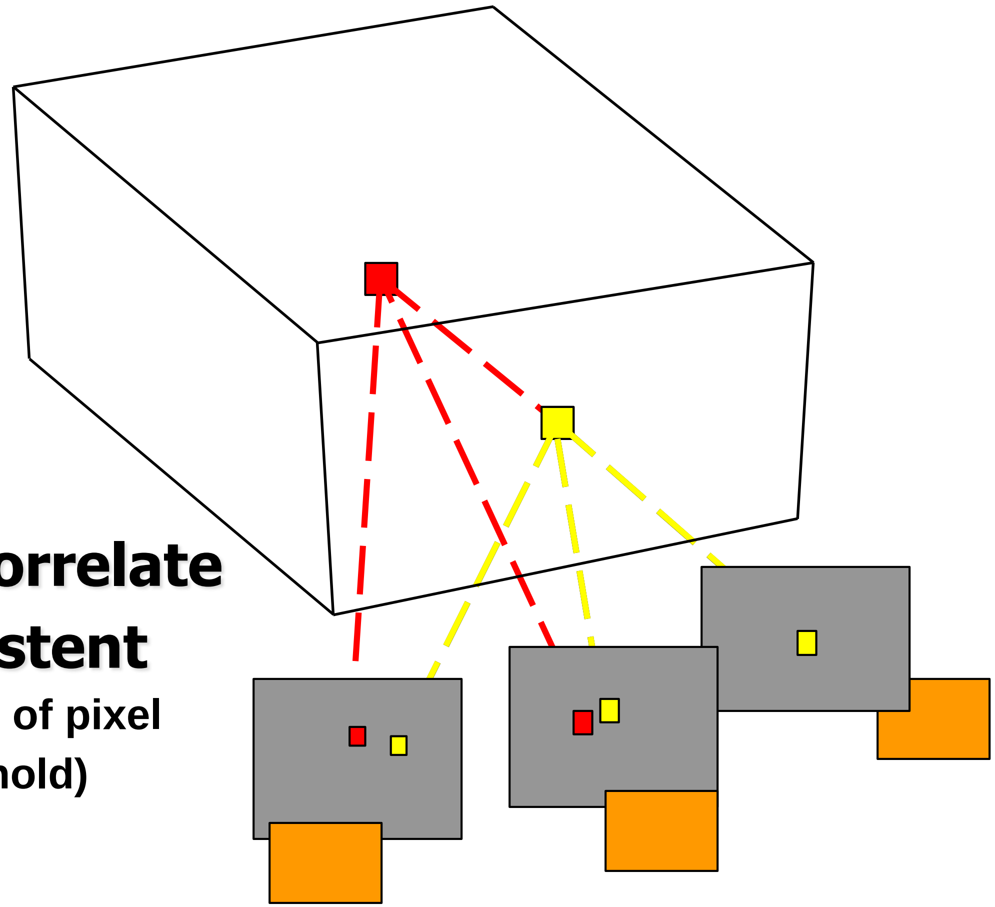
1. **Choose voxel**
2. **Project and correlate**
3. **Discard if inconsistent**

Voxel Coloring Approach



1. **Choose voxel**
2. **Project and correlate**
3. **Color if consistent**
   (standard deviation of pixel colors below threshold)

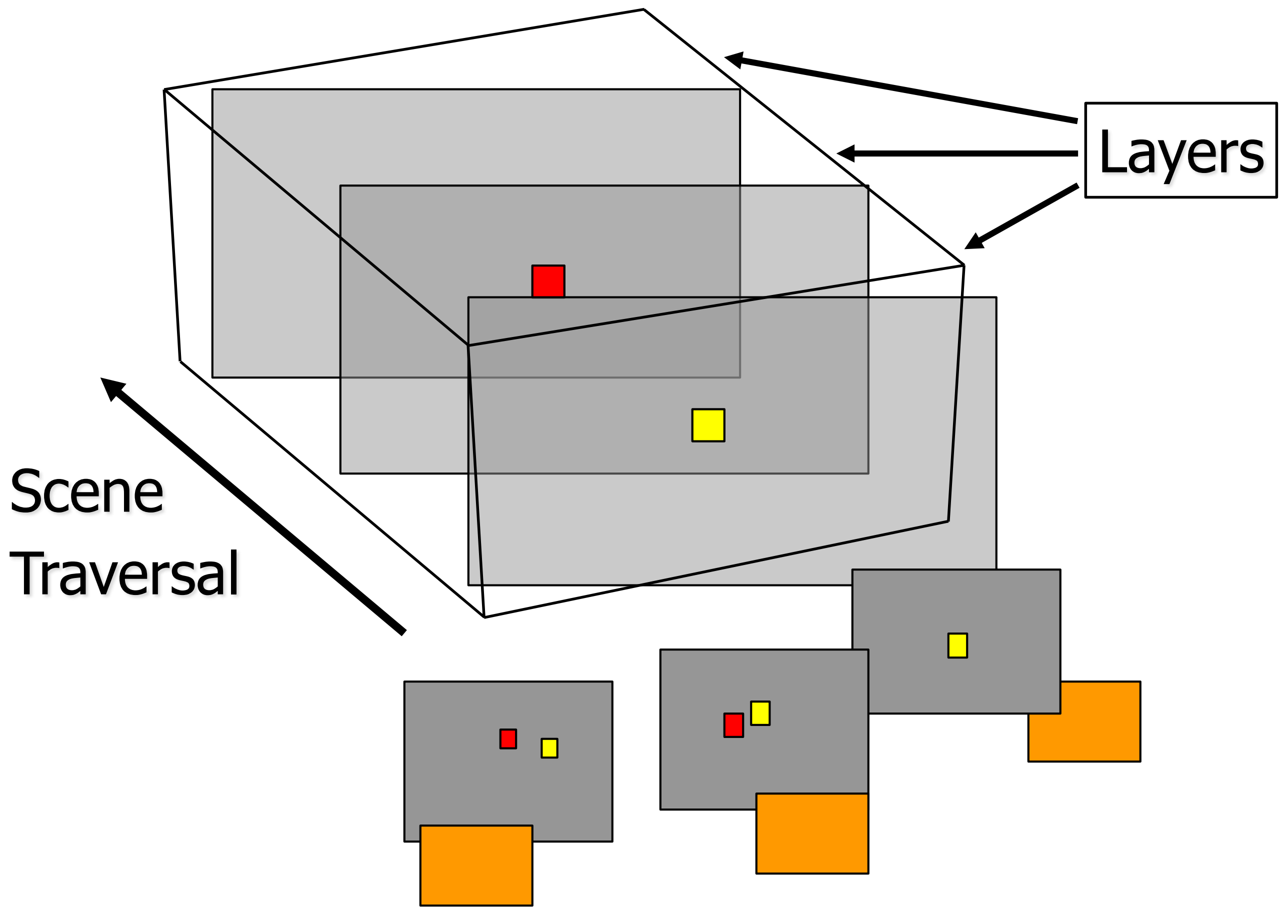
Voxel Coloring Approach



1. **Choose voxel**
2. **Project and correlate**
3. **Color if consistent**
   (standard deviation of pixel colors below threshold)

Visibility Problem: **in which images is each voxel visible?**

Depth Ordering: Visit Occluders First!



Calibrated Image Acquisition

Calibrated Turntable

Selected Dinosaur Images

Selected Flower Images

Voxel Coloring Results

Dinosaur Reconstruction
72 K voxels colored
7.6 M voxels tested
7 min. to compute
on a 250MHz SGI

Flower Reconstruction
70 K voxels colored
7.6 M voxels tested
7 min. to compute
on a 250MHz SGI

Space Carving Results: African Violet

Input Image (1 of 45)

Reconstruction

Reconstruction

Reconstruction

Source: S. Seitz

Improvements

Unconstrained camera viewpoints

- Space carving [Kutulakos & Seitz 98]

Evolving a surface

- Level sets [Faugeras & Keriven 98]
- More recent work by Pons et al.

Global optimization

- Graph cut approaches
  - [Kolmogoriv & Zabih, ECCV 2002]
  - [Vogiatzis et al., PAMI 2007]

Modeling shiny (and other reflective) surfaces

- e.g., Zickler et al., *Helmholtz Stereopsis*

Binocular Stereo

Binocular Stereo

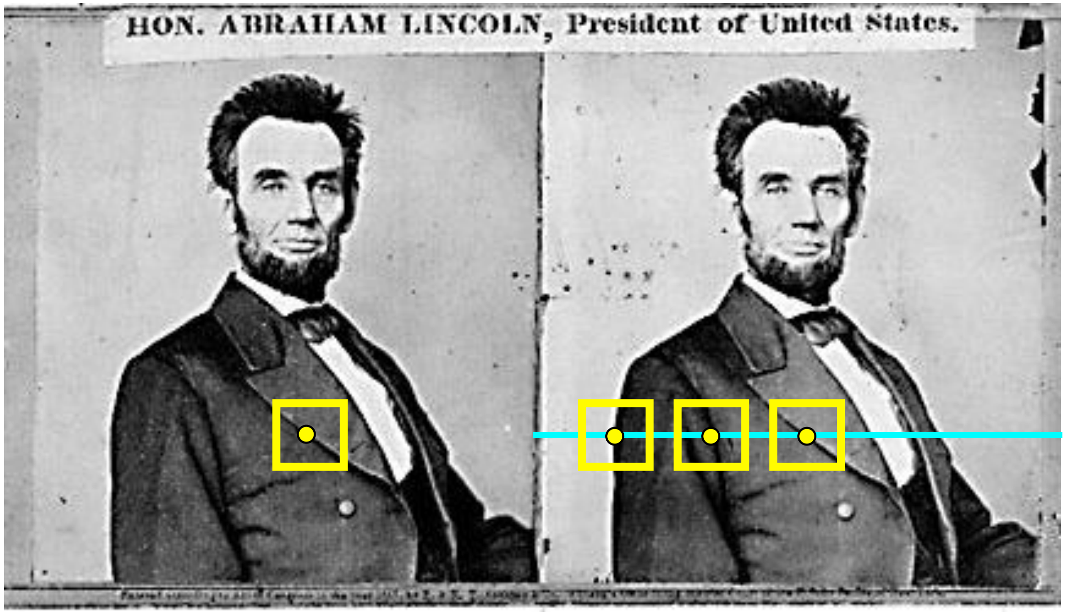
- Given a calibrated binocular stereo pair, fuse it to produce a depth image

image 1

image 2

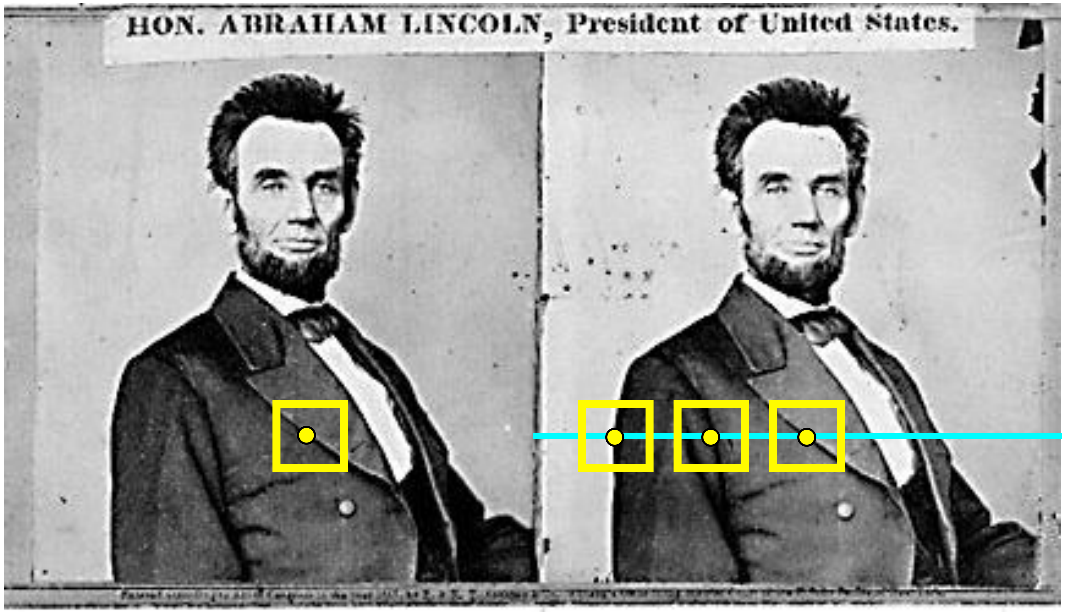
Dense depth map

Basic Stereo Matching Algorithm



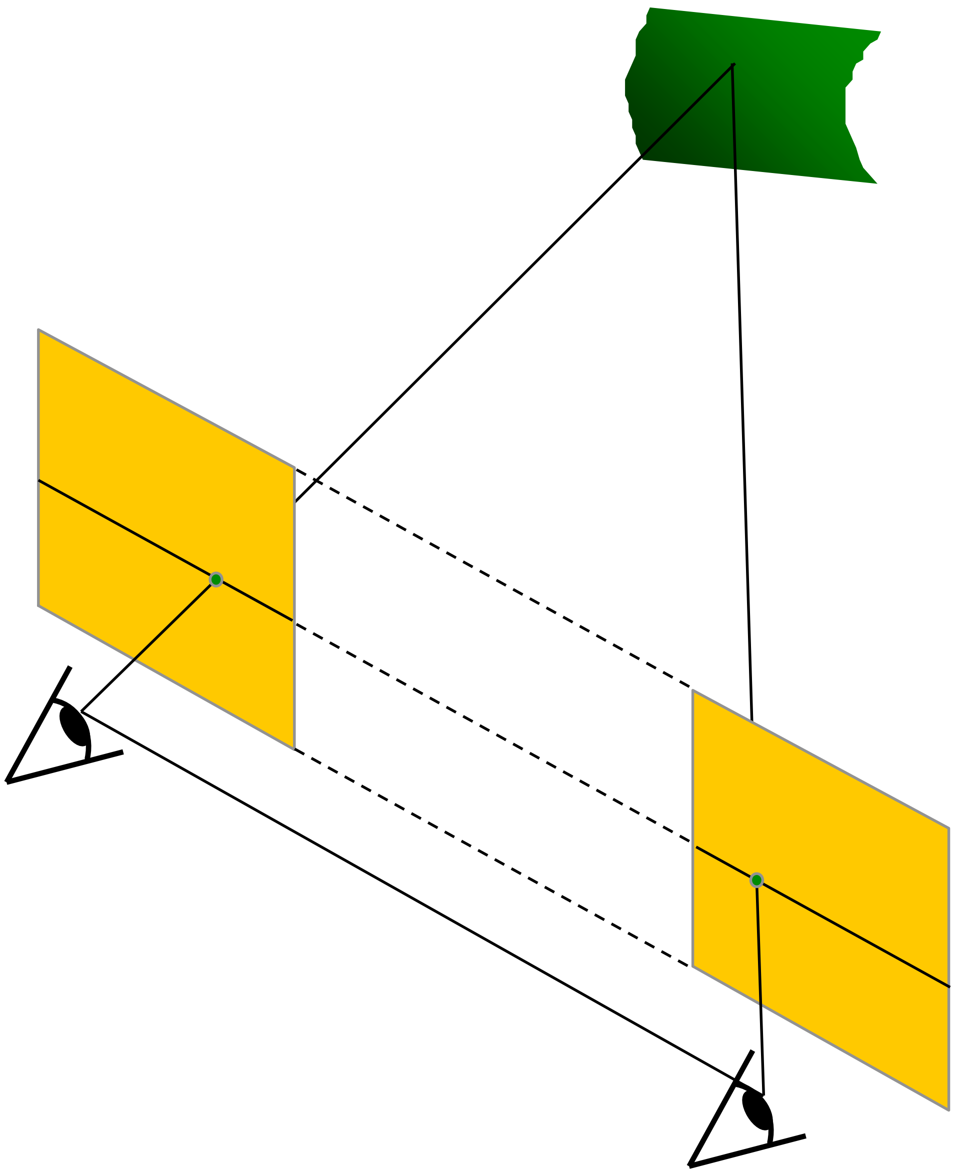
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match
  - Triangulate the matches to get depth information
- Simplest case: epipolar lines are corresponding scanlines
  - When does this happen?

Basic stereo matching algorithm



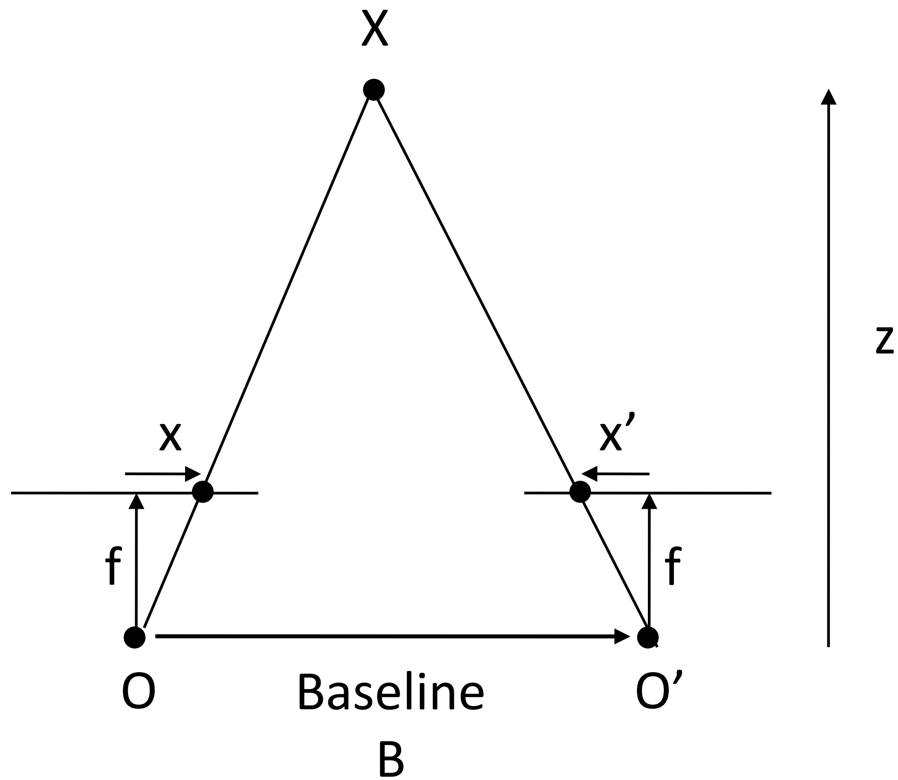
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match
  - Triangulate the matches to get depth information
- Simplest case: epipolar lines are corresponding scanlines
  - When does this happen?

Simplest Case: Parallel Images



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

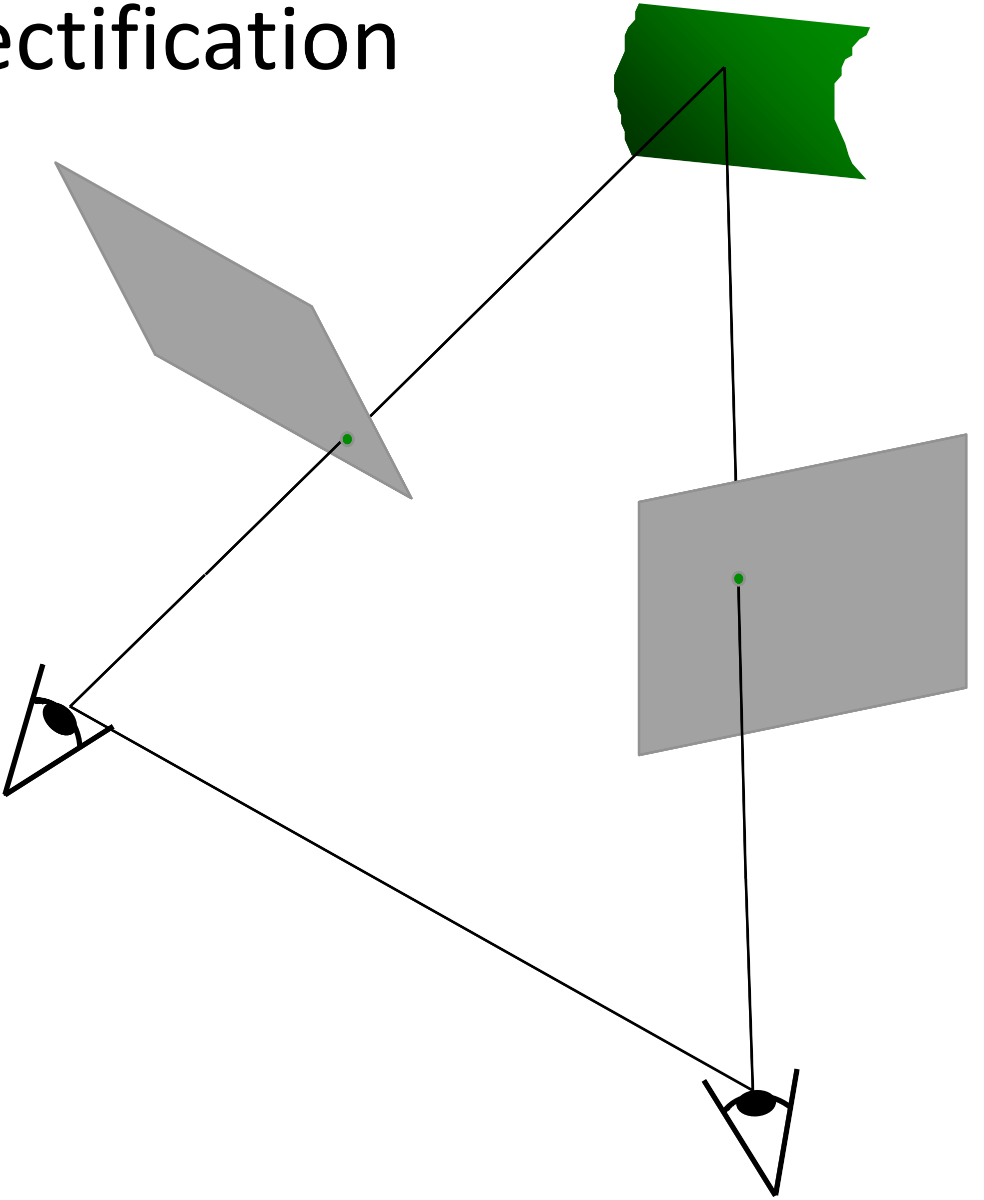
Depth from Disparity



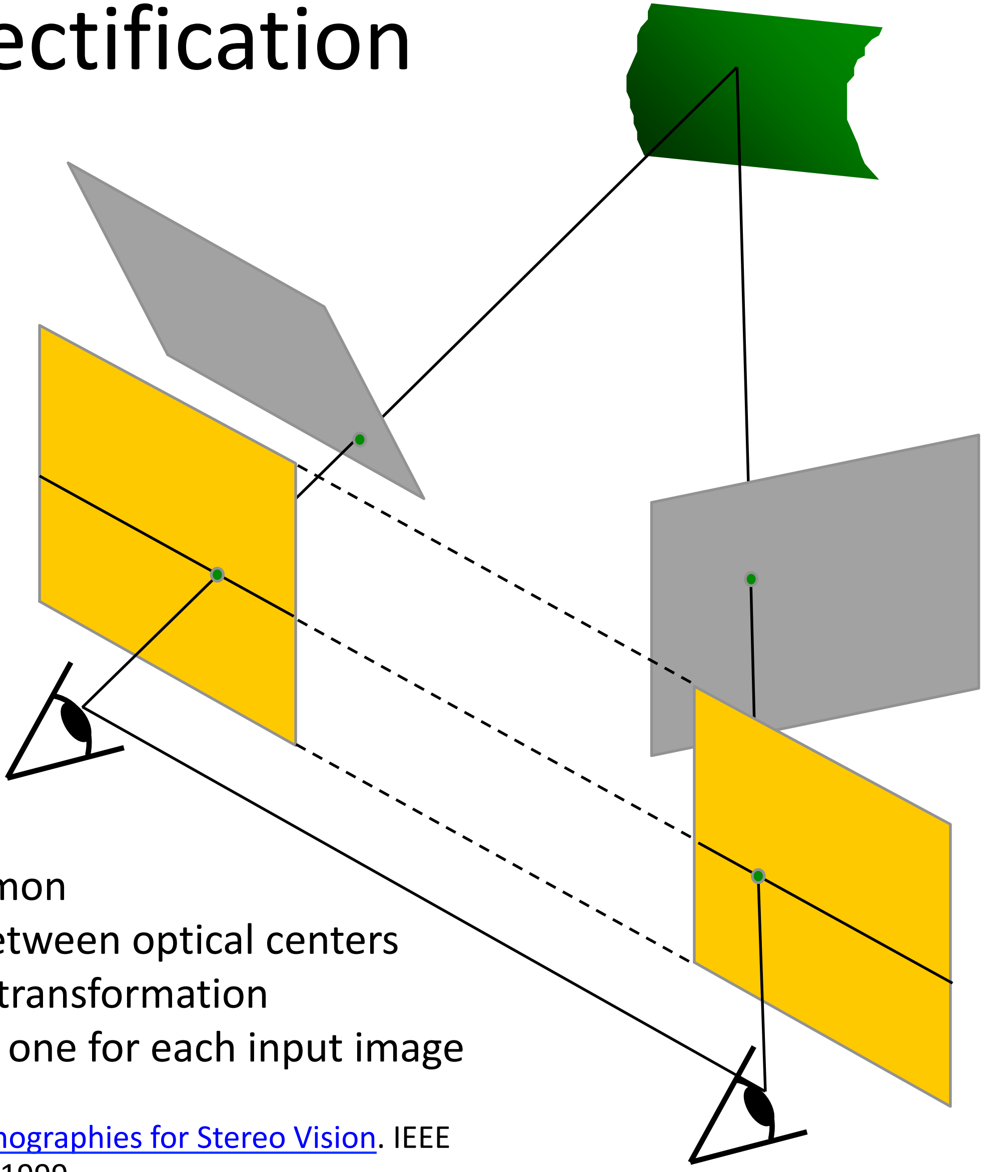
$$disparity = x - x' = \frac{B \times f}{z}$$

Disparity is inversely proportional to depth!

Stereo Image Rectification



Stereo Image Rectification

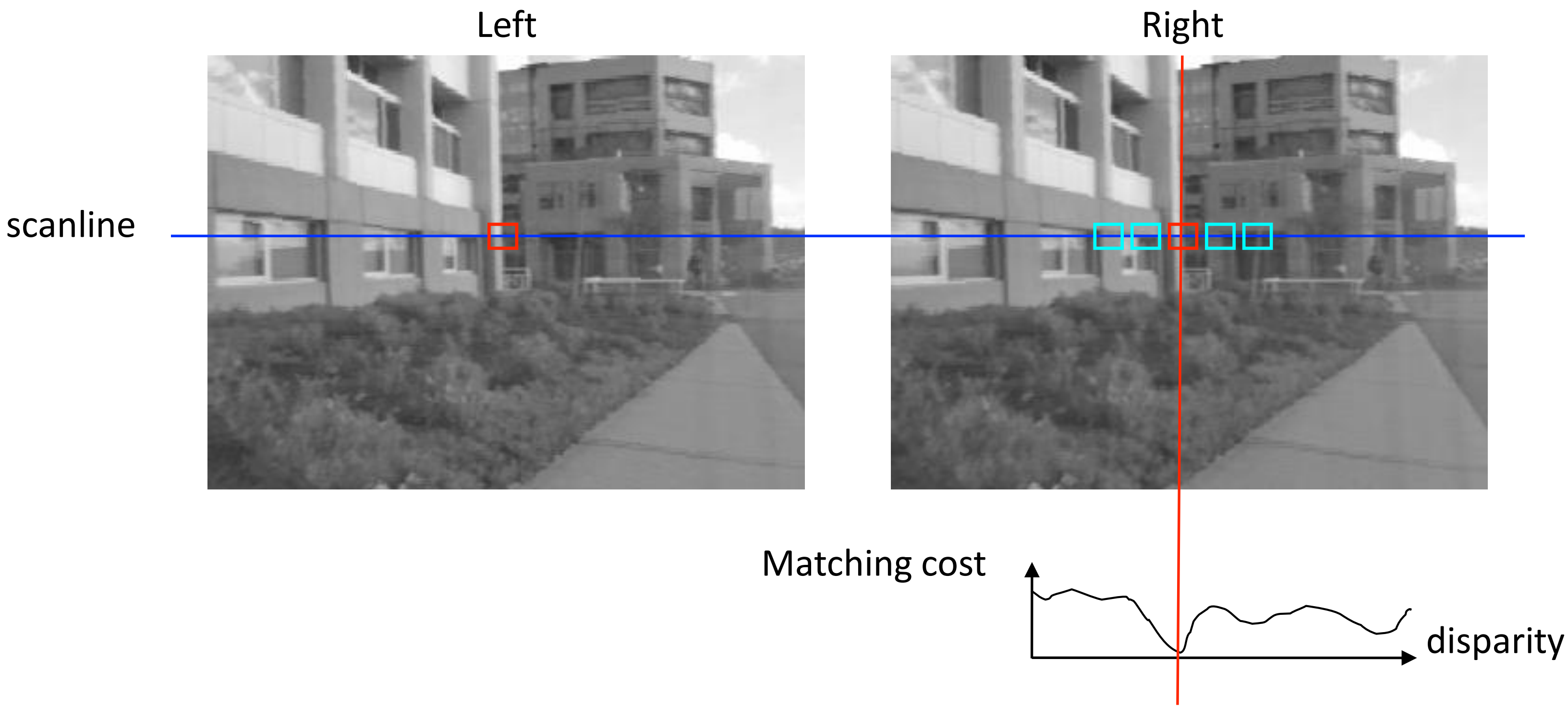


- reproject image planes onto a common
- plane parallel to the line between optical centers
- pixel motion is horizontal after this transformation
- two homographies (3x3 transform), one for each input image reprojection

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

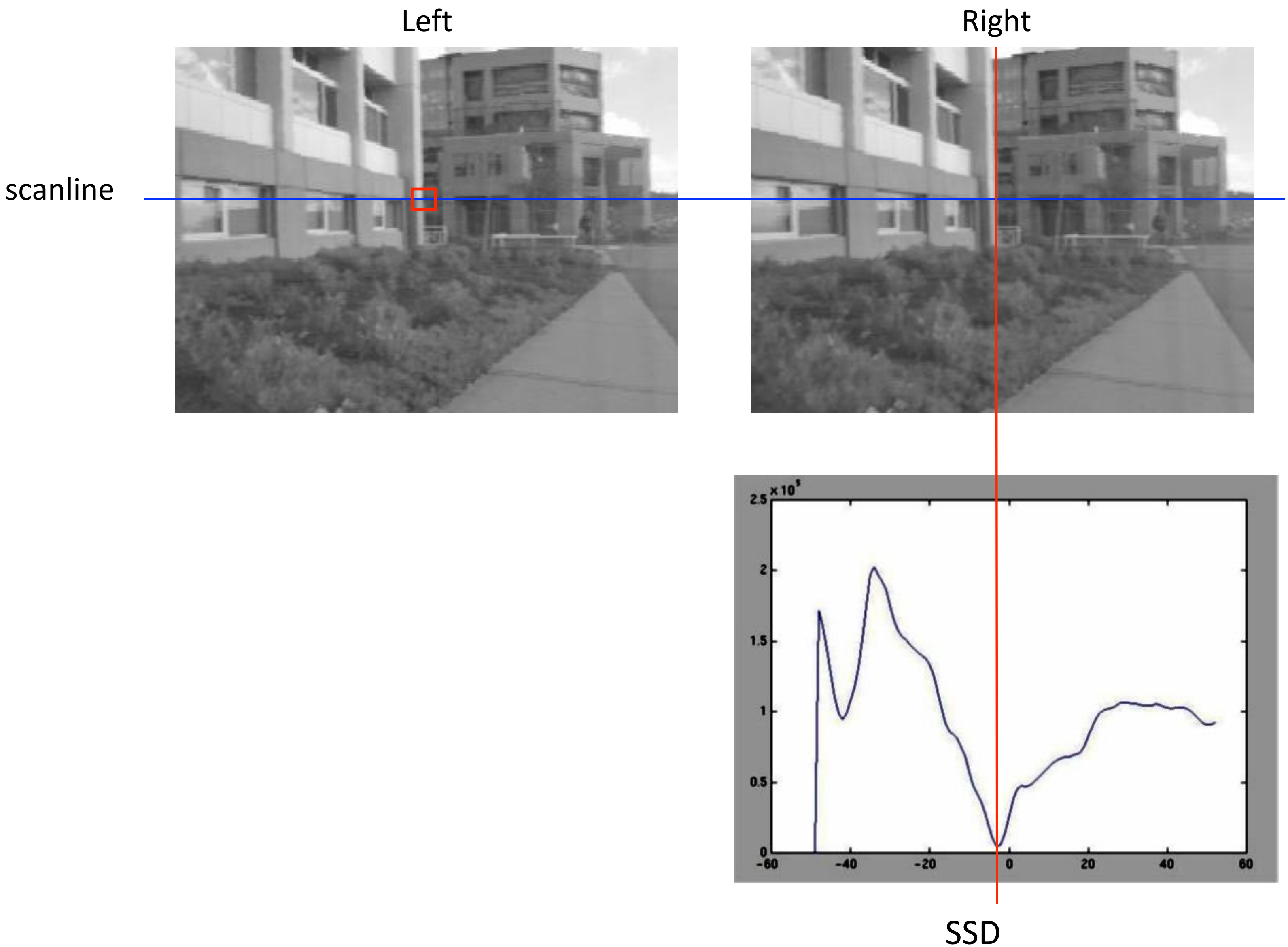
Rectification Example

Correspondence search

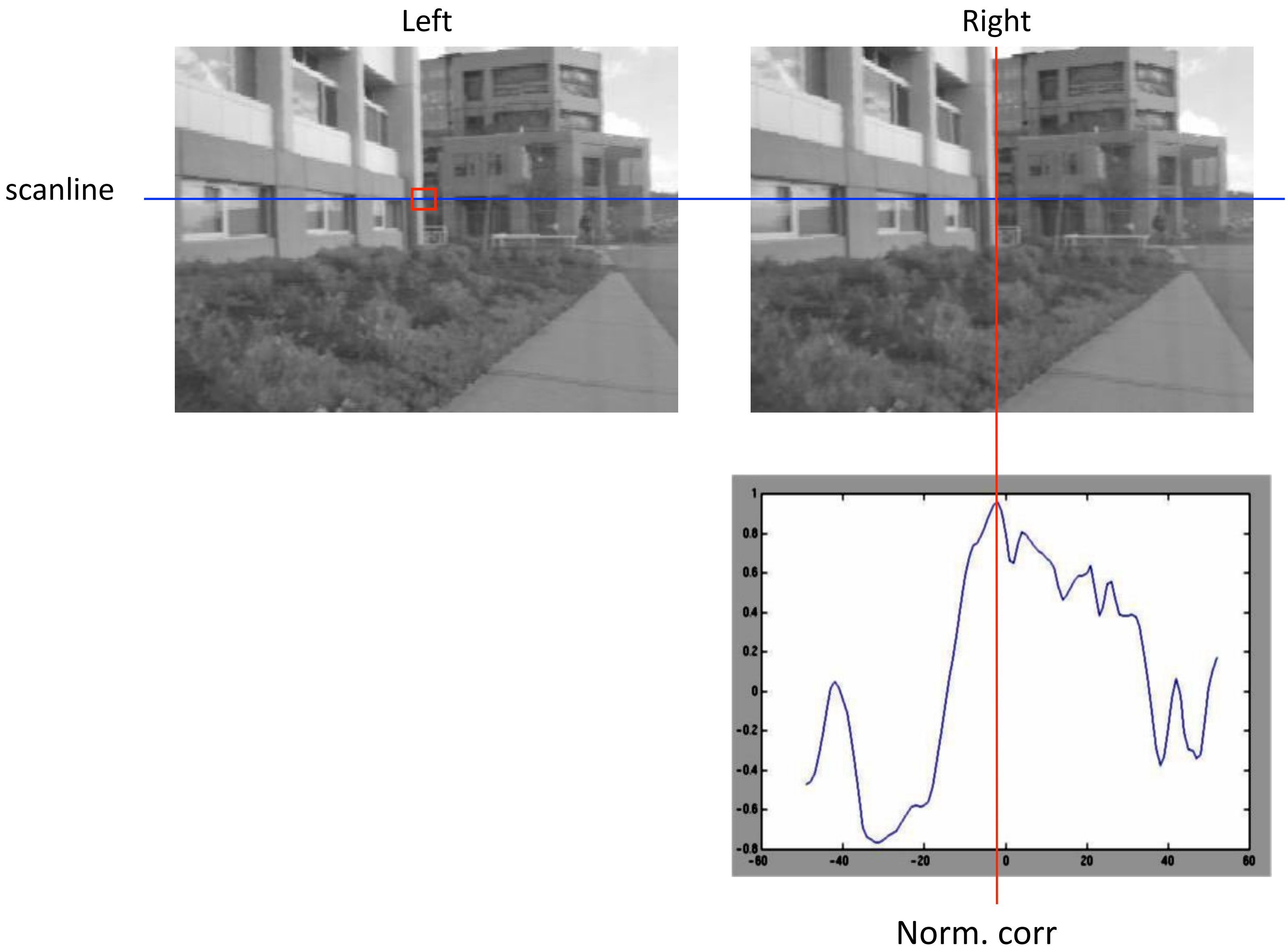


- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

Correspondence search

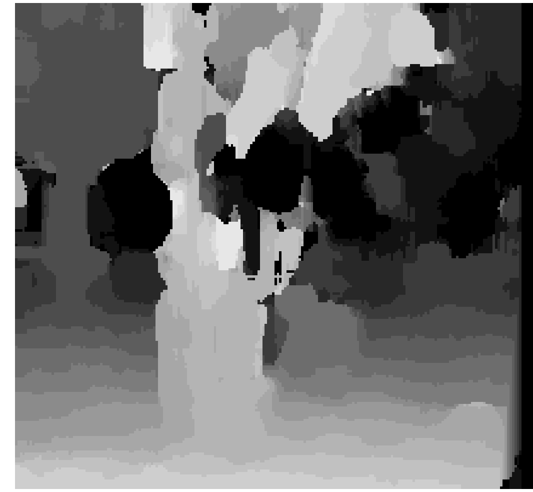


Correspondence search



Effect of window size

W = 3



W = 20

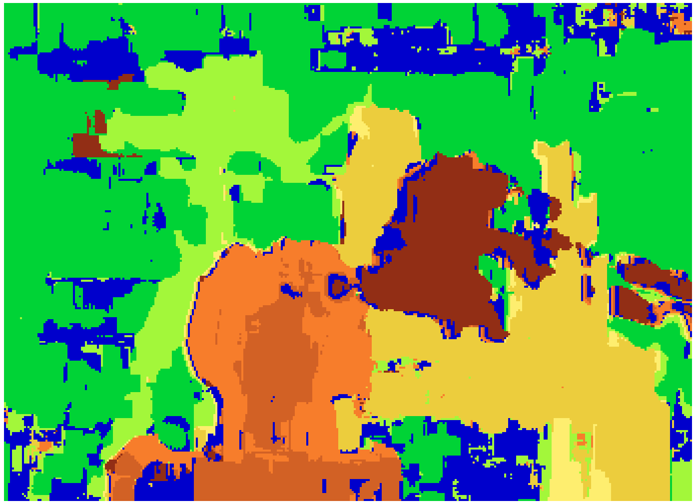
- Smaller window
  - \+ More detail
  - – More noise

- Larger window
  - \+ Smoother disparity maps
  - – Less detail

Results with window search

Data

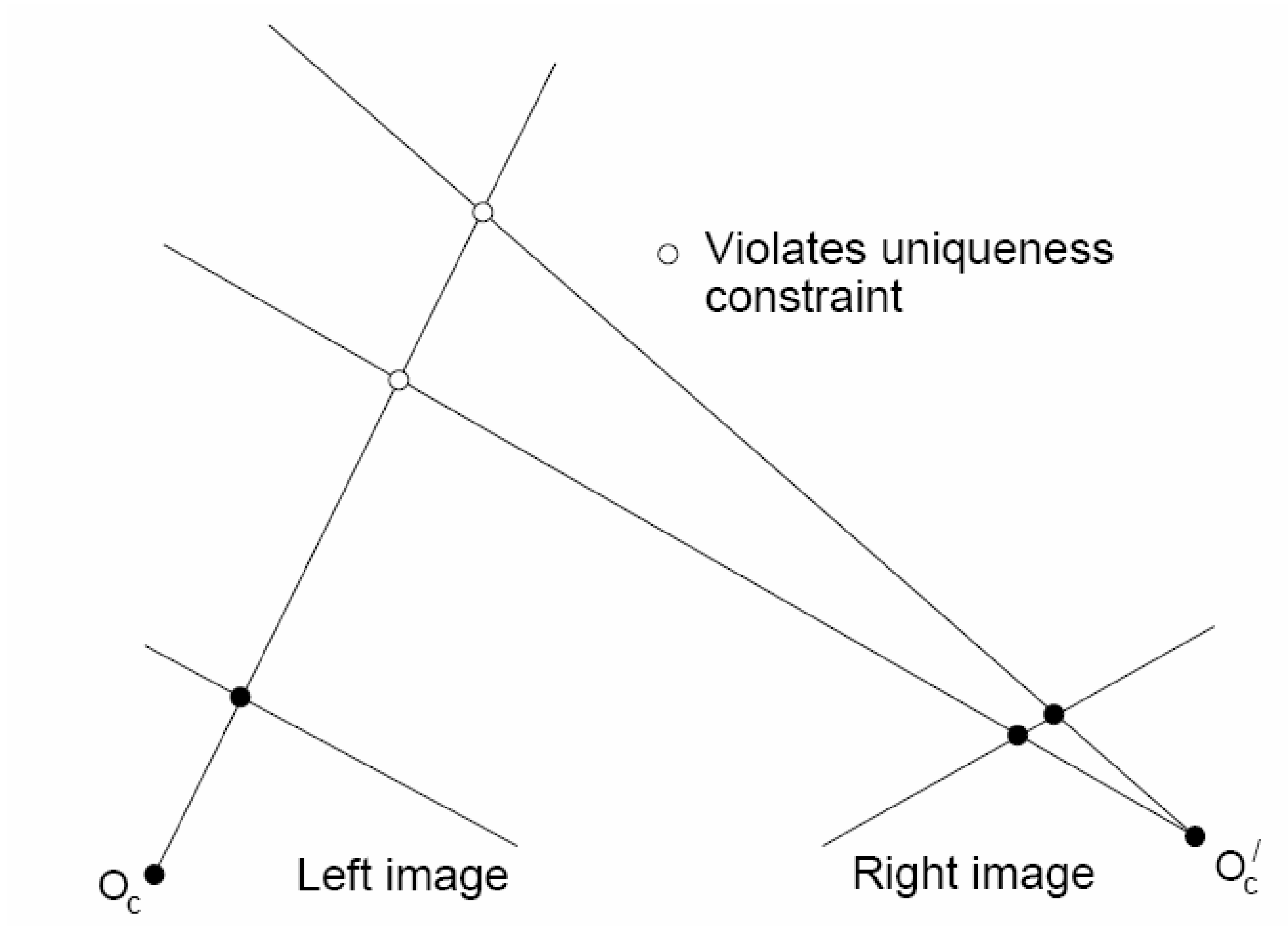
Window-based matching



Ground truth

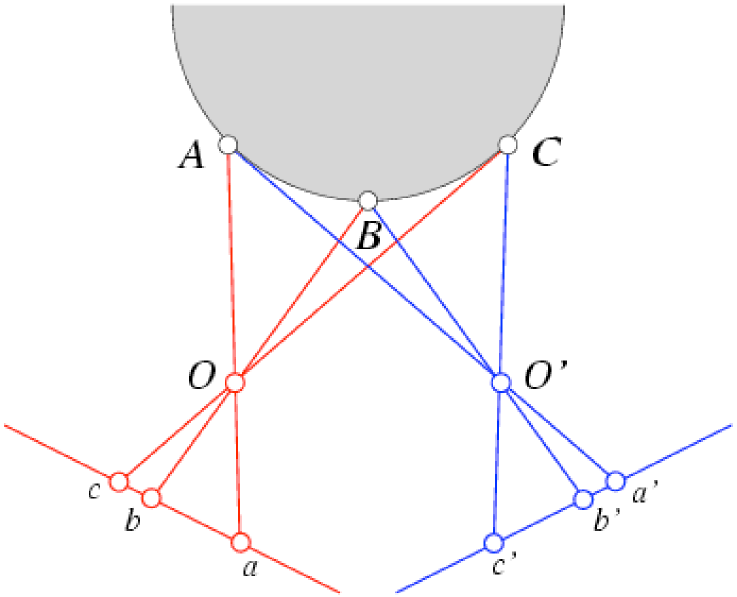
Non-local constraints

- Uniqueness
  – For any point in one image, there should be at most one matching point in the other image



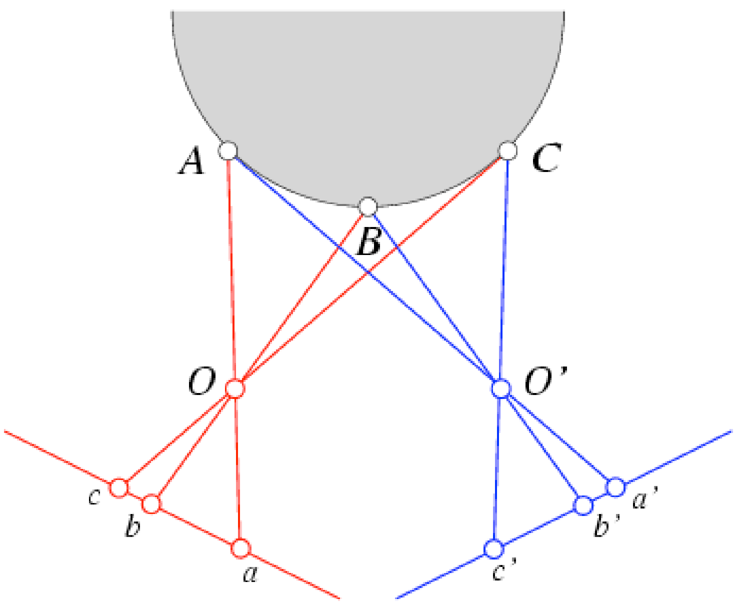
Non-local constraints

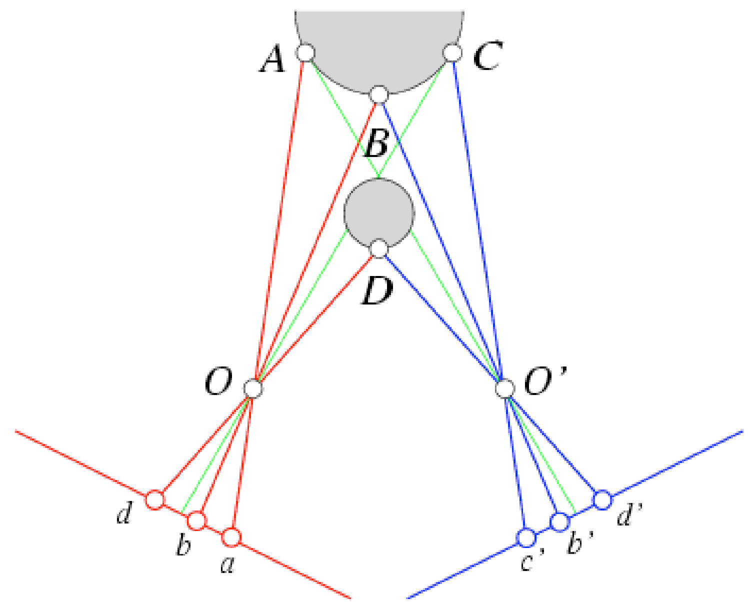
- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views



Non-local constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views





Ordering constraint doesn't hold

Consistency Constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views
- Smoothness
  - We expect disparity values to change slowly (for the most part)

MRF Formulation:

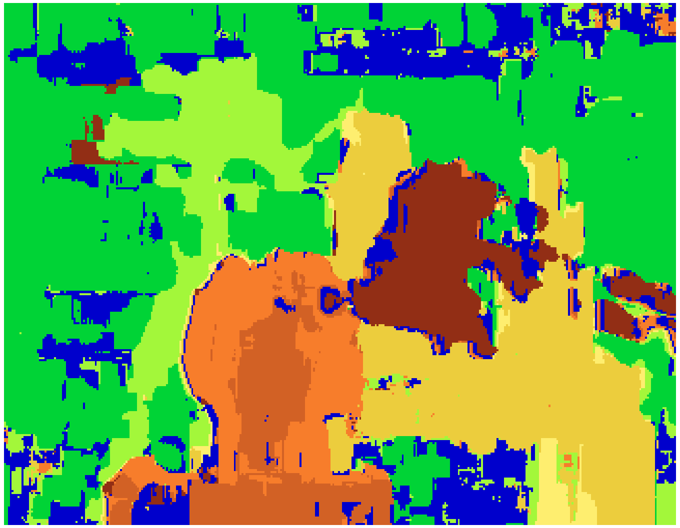
$$E(d) = E_d(d) + \lambda E_s(d)$$

Pixel matching score

Consistency Scores

Comparsion

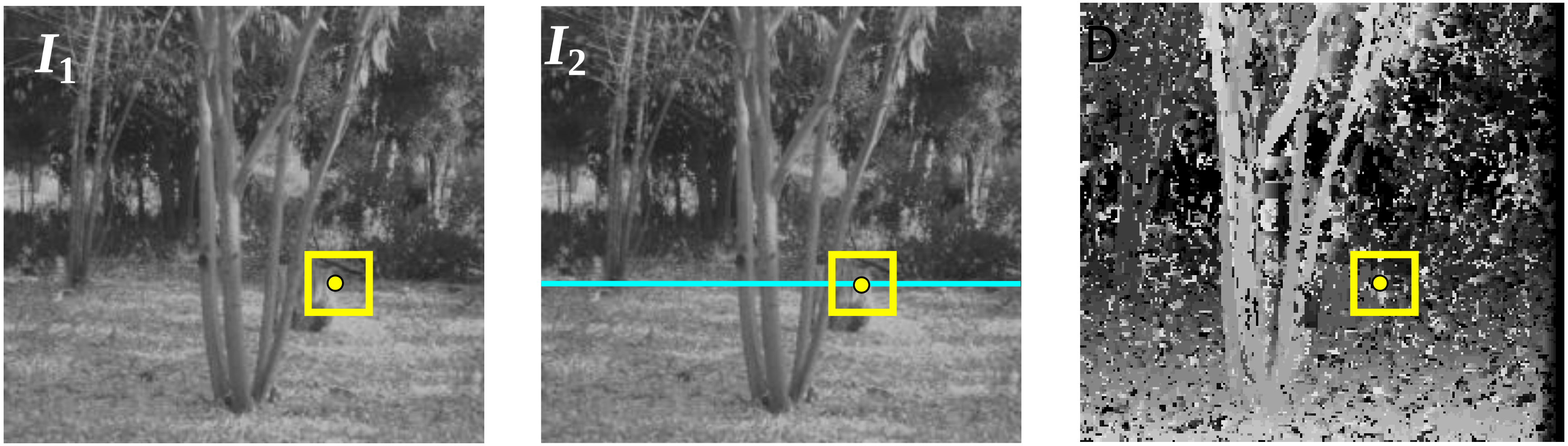
Window-Based Search:



Graph Cut:

Ground Truth

Stereo matching as energy minimization

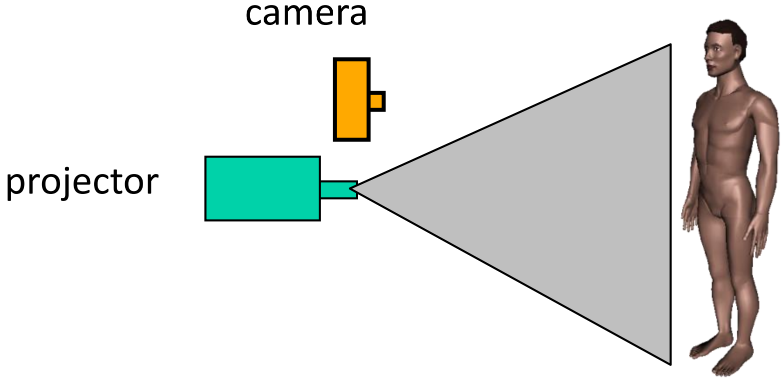


- Graph-cuts can be used to minimize such energy

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

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

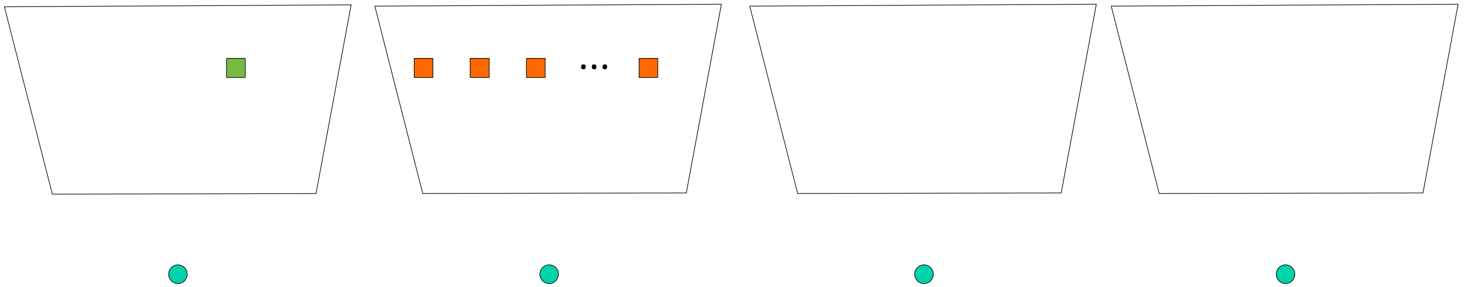
Kinect: Structured infrared light

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

Multi-Baseline Stereo

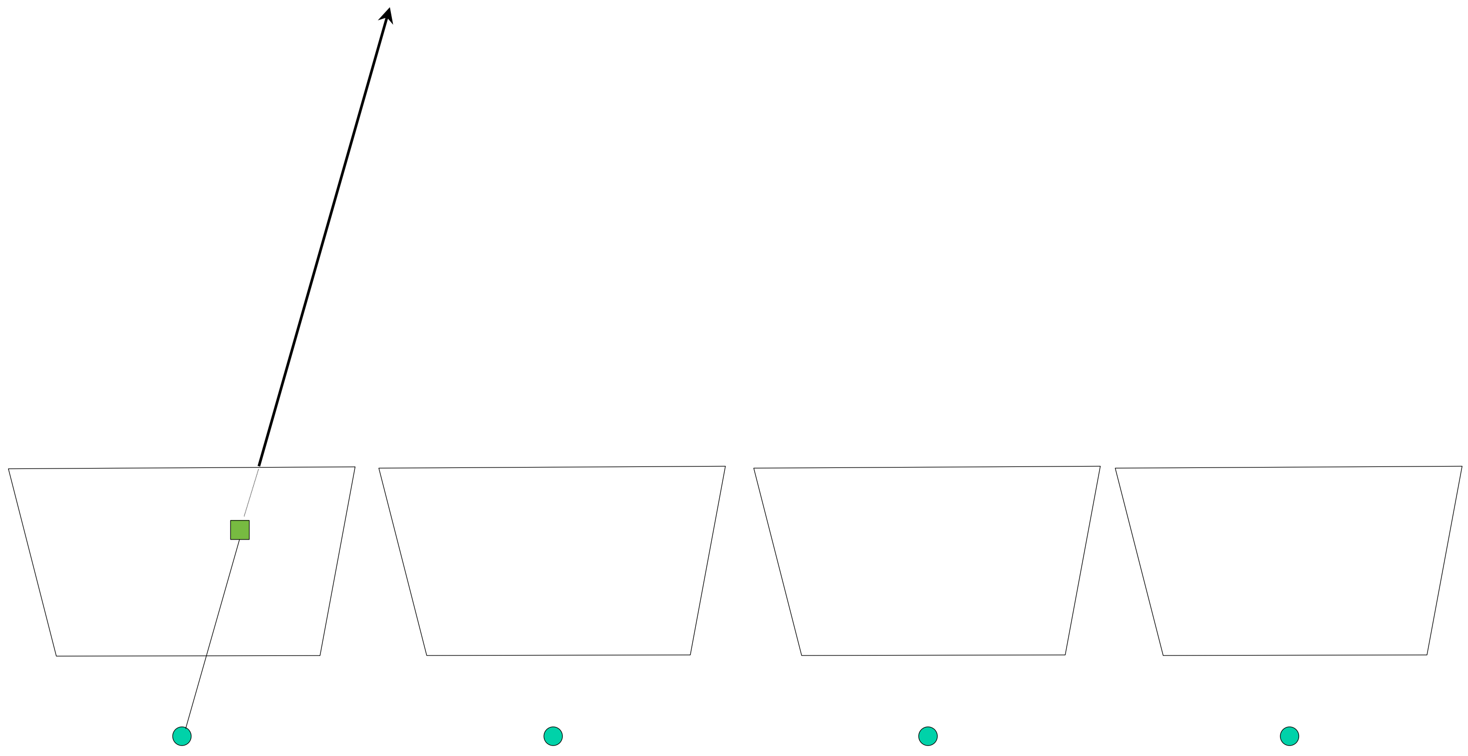
Same formulation with more images

- Change label from disparity to depth
- Change $E_d(d)$ by using more images



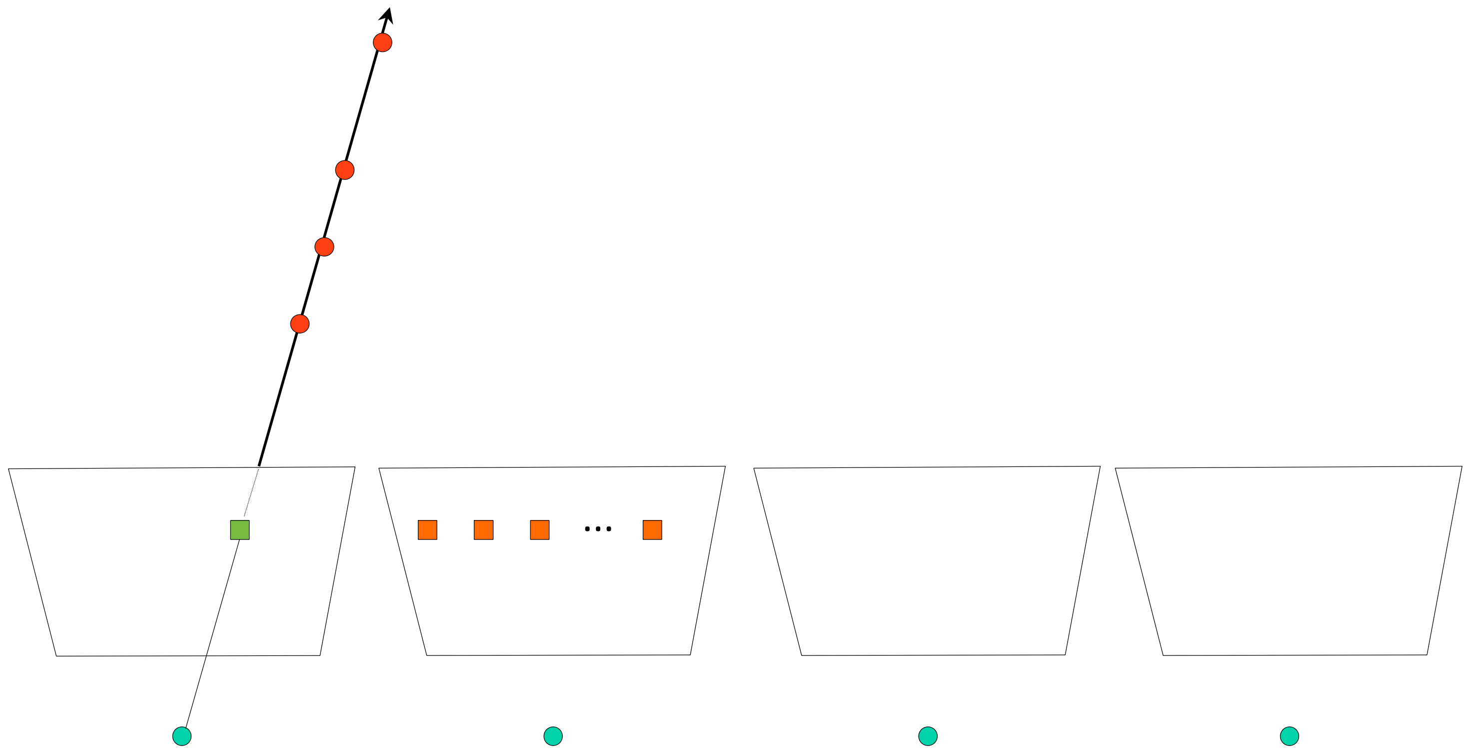
Same formulation with more images

- Change label from disparity to depth
- Change $E_d(d)$ by using more images



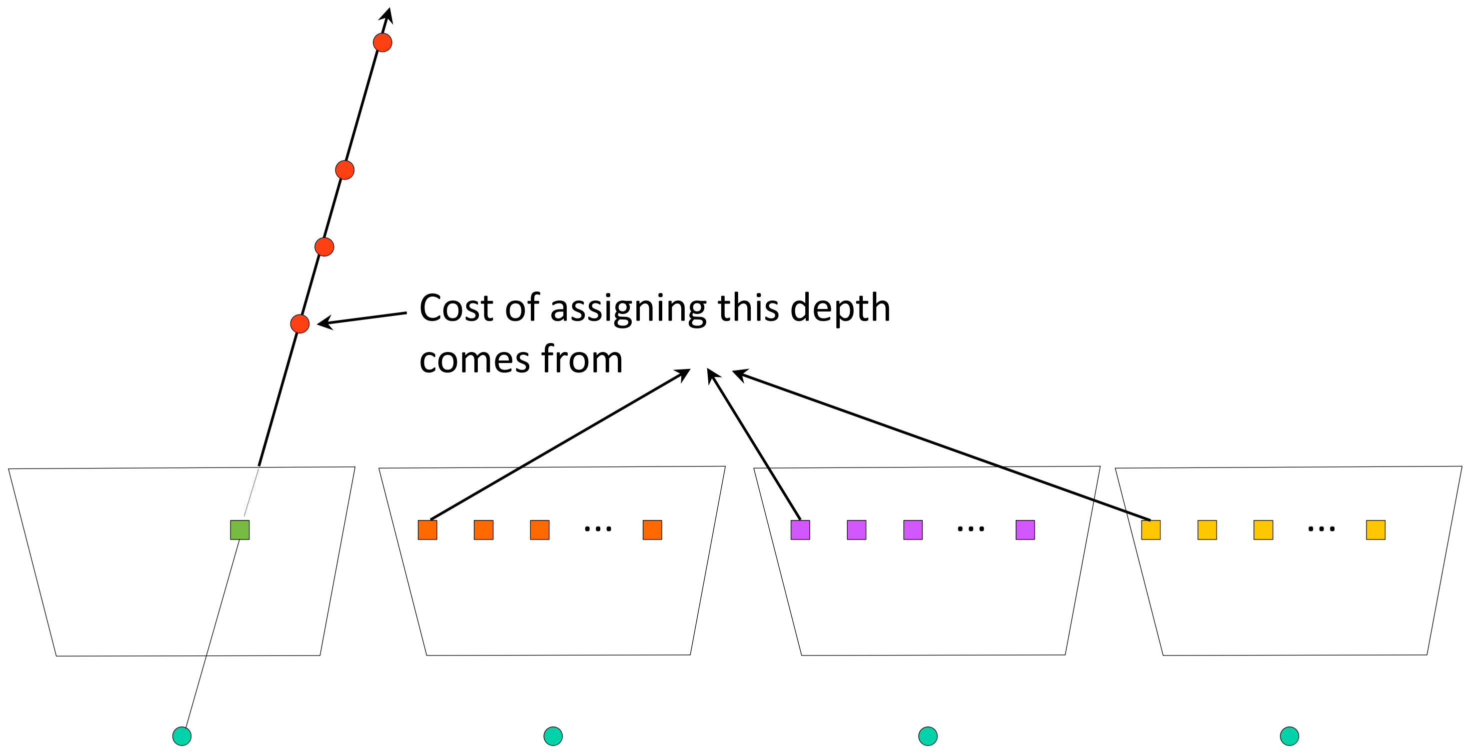
Same formulation with more images

- Change label from disparity to depth
- Change $E_d(d)$ by using more images

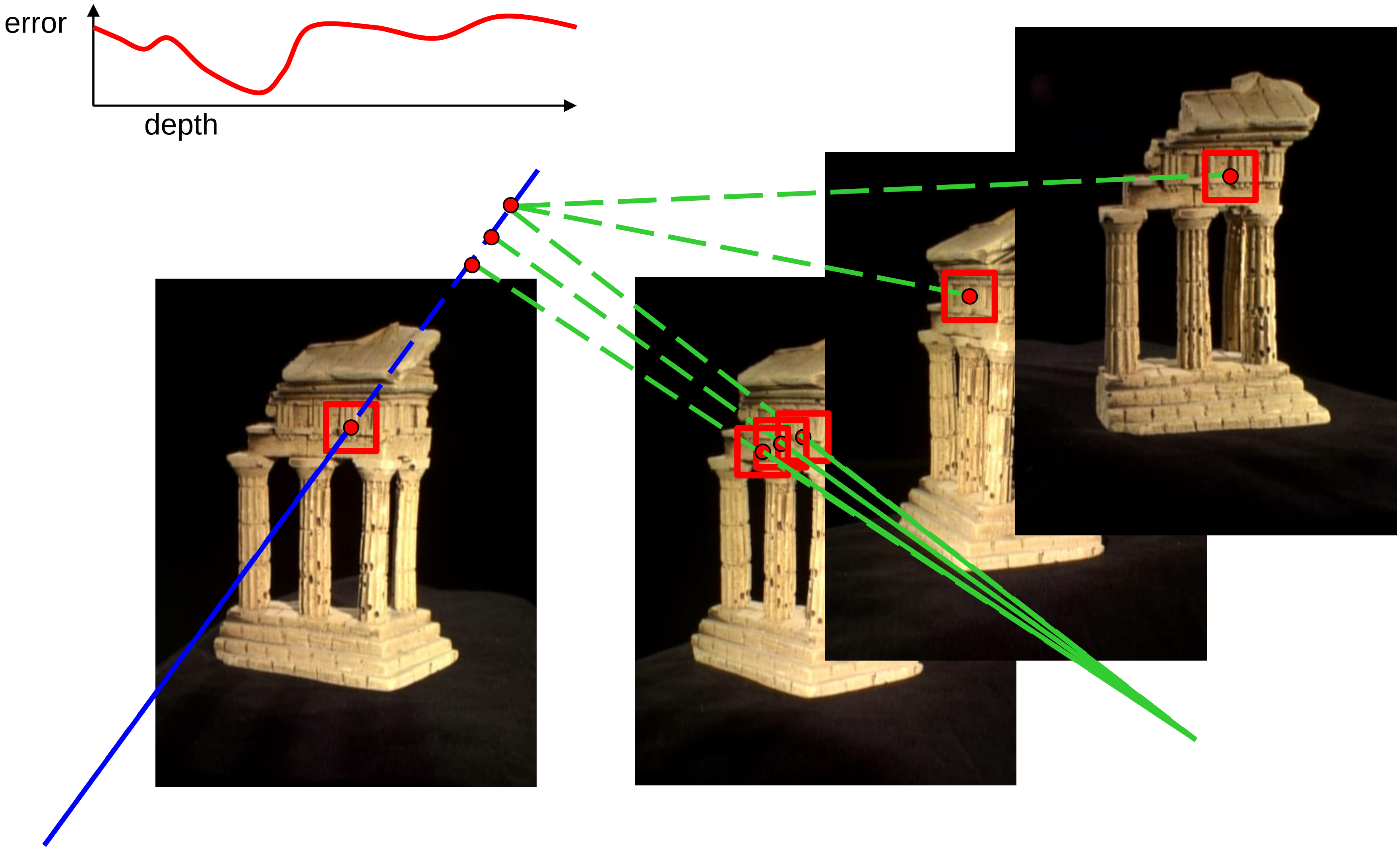


Same formulation with more images

- Change label from disparity to depth
- Change $E_d(d)$ by using more images

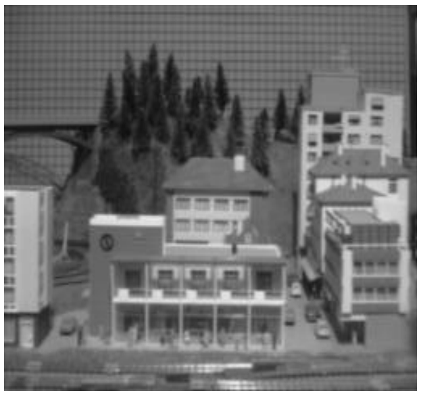


Stereo: Basic Idea



Multiple-Baseline Stereo Results

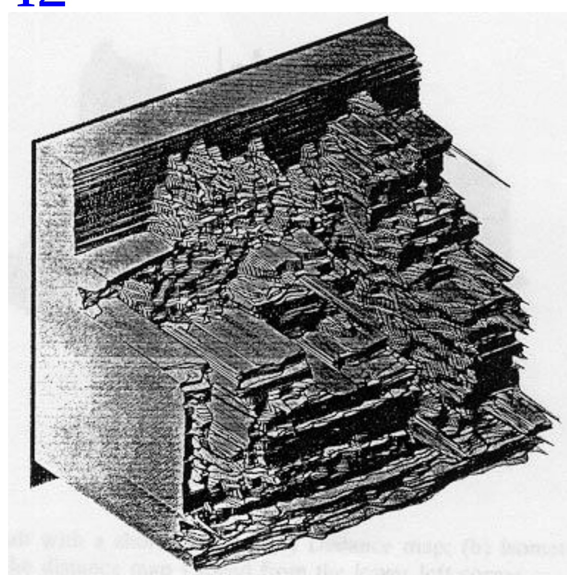
[Okutomi and Kanade' 93]



I1

I2

I10



Mesh Reconstruction

Merging Depth Maps

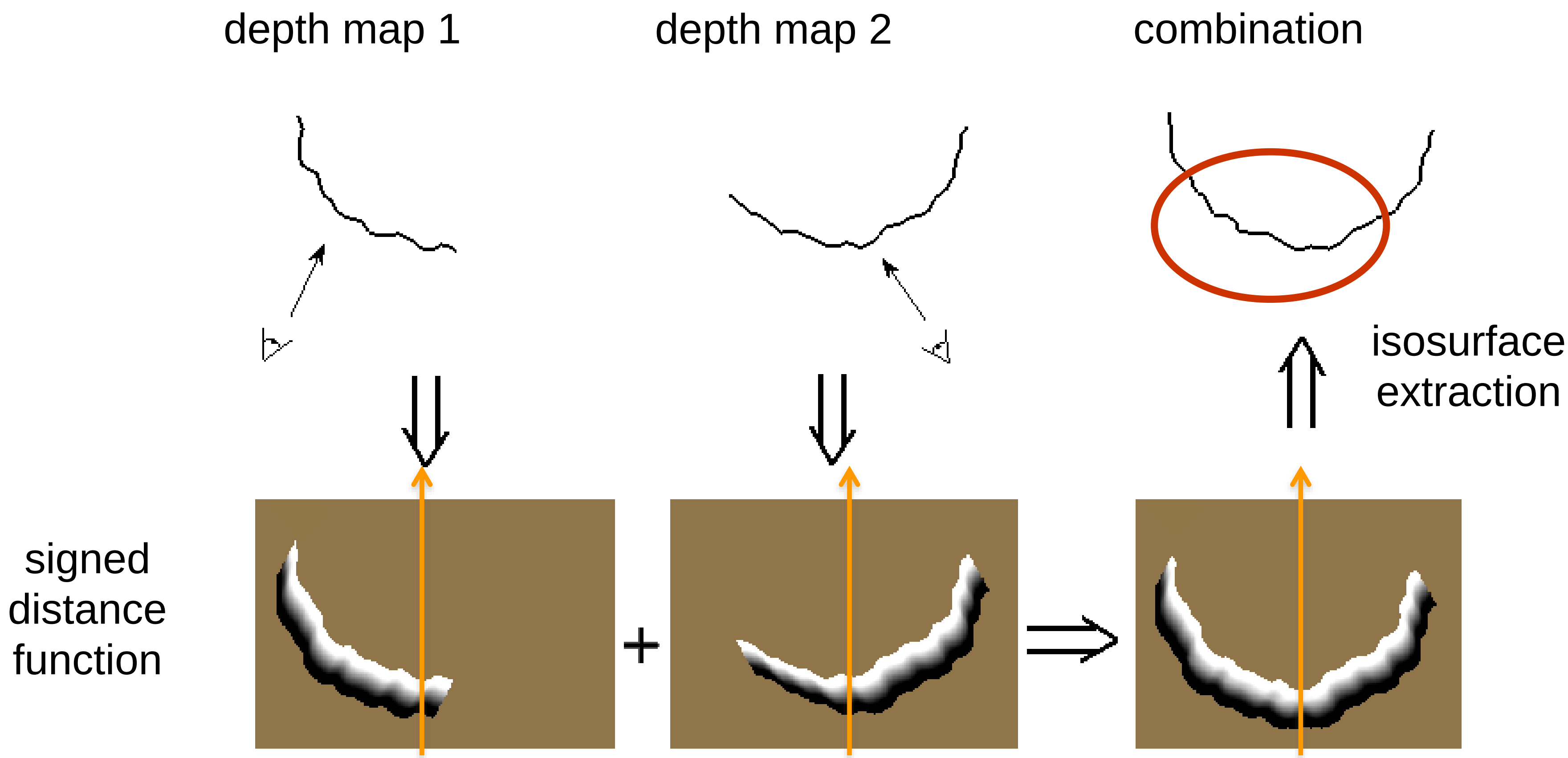
vrip [Curless and Levoy 1996]

- compute weighted average of depth maps

set of depth maps
(one per view)

merged surface
mesh

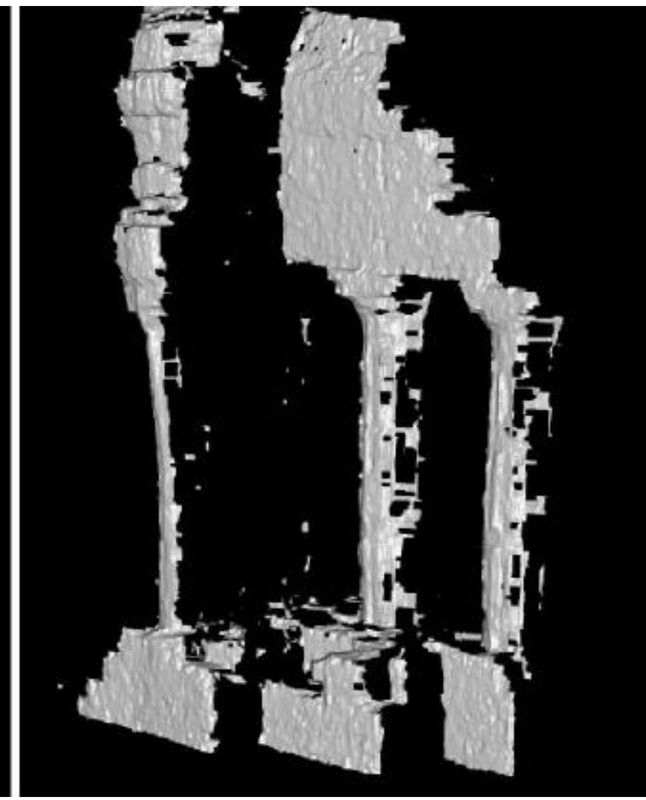
VRIP


depth map 1
depth map 2
combination
isosurface extraction
signed distance function
+

Depthmap Merging

Depthmap 1

Depthmap 2



Merging Depth Maps: Temple Model

[Goesele et al. 06]

input image

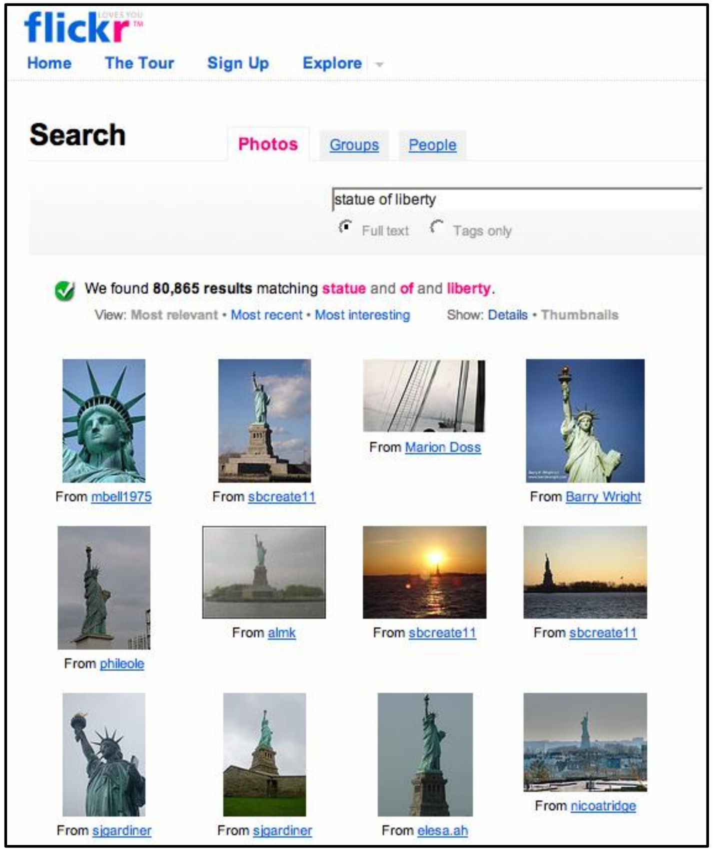
317 images
(hemisphere)

ground truth model

State-of-The-Art

Multi-View Stereo from Internet Collections

[Goesele et al. 07]



Challenges

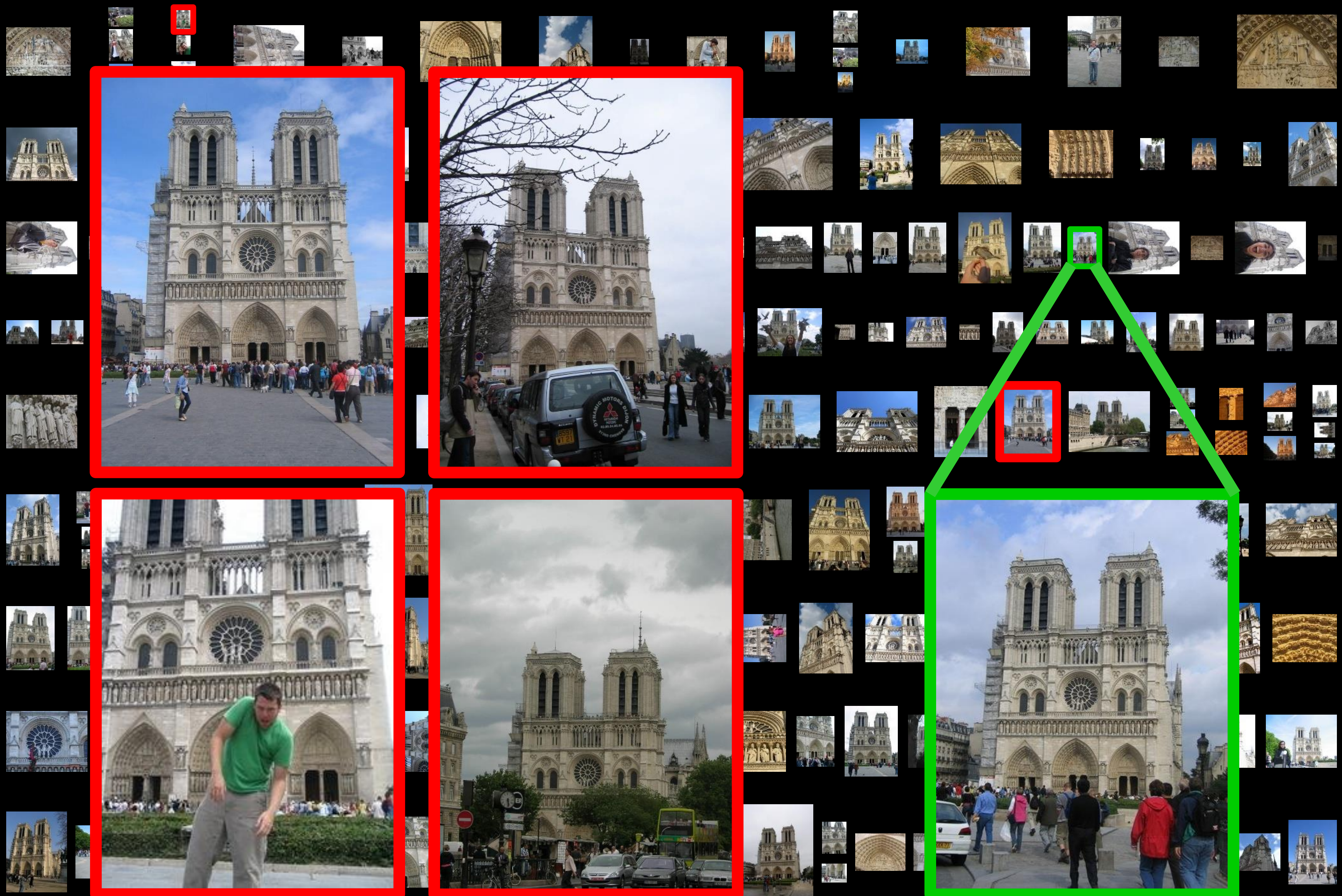
- Appearance variation

- Resolution

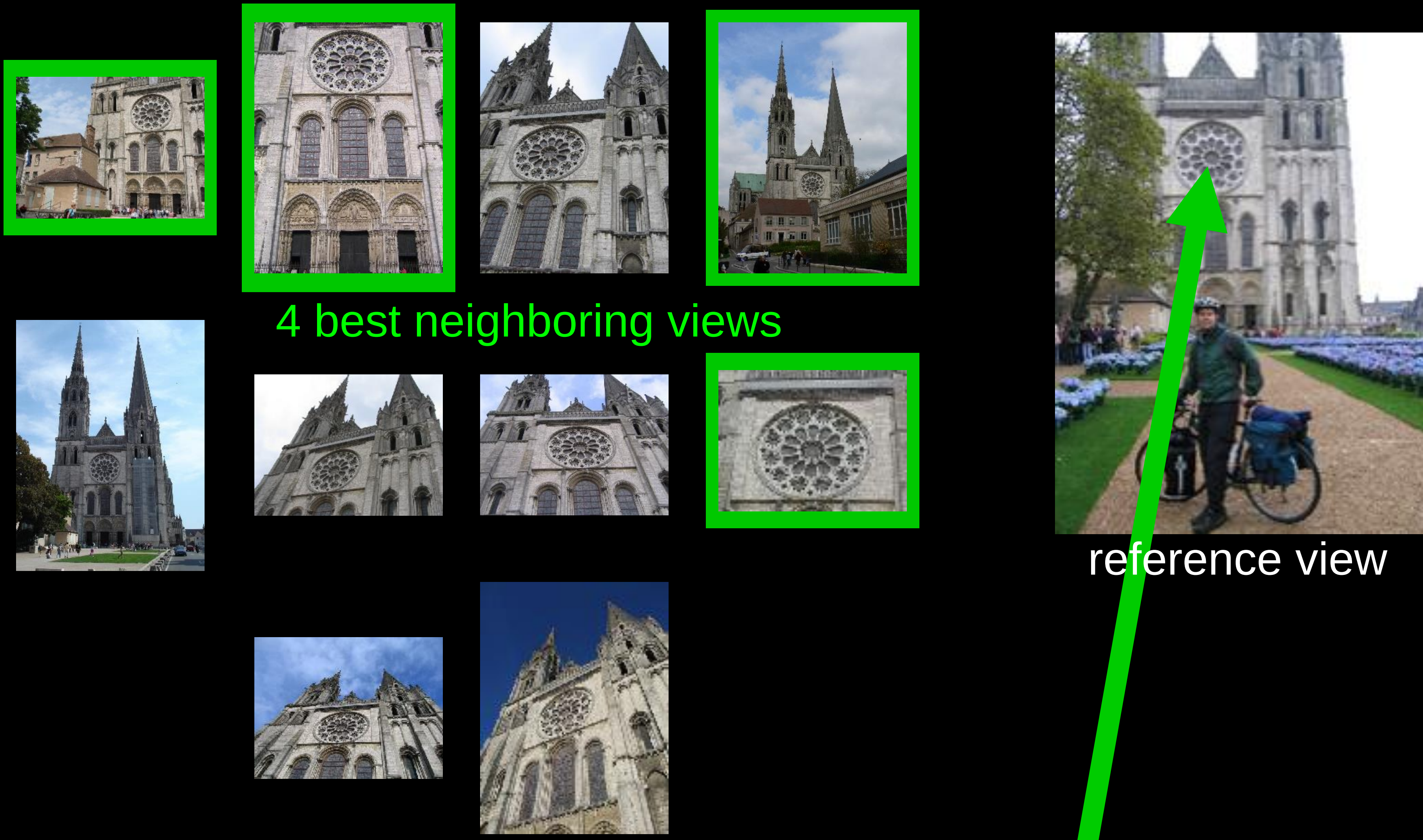
- Massive collections

82754 results for photos matching **notre** and **dame** and **paris**

Law of Nearest Neighbors

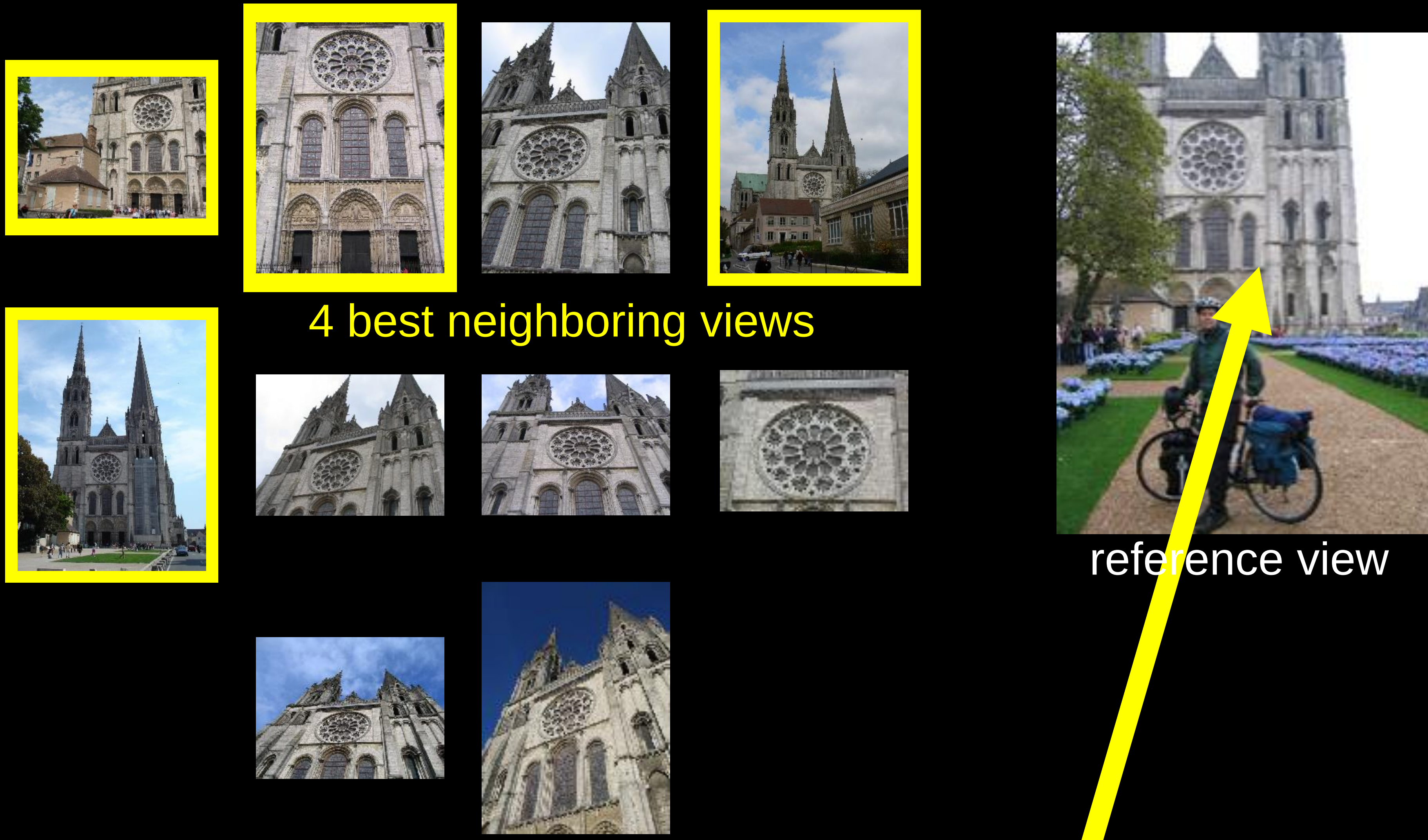


206 *Flickr* images taken by 92 photographers



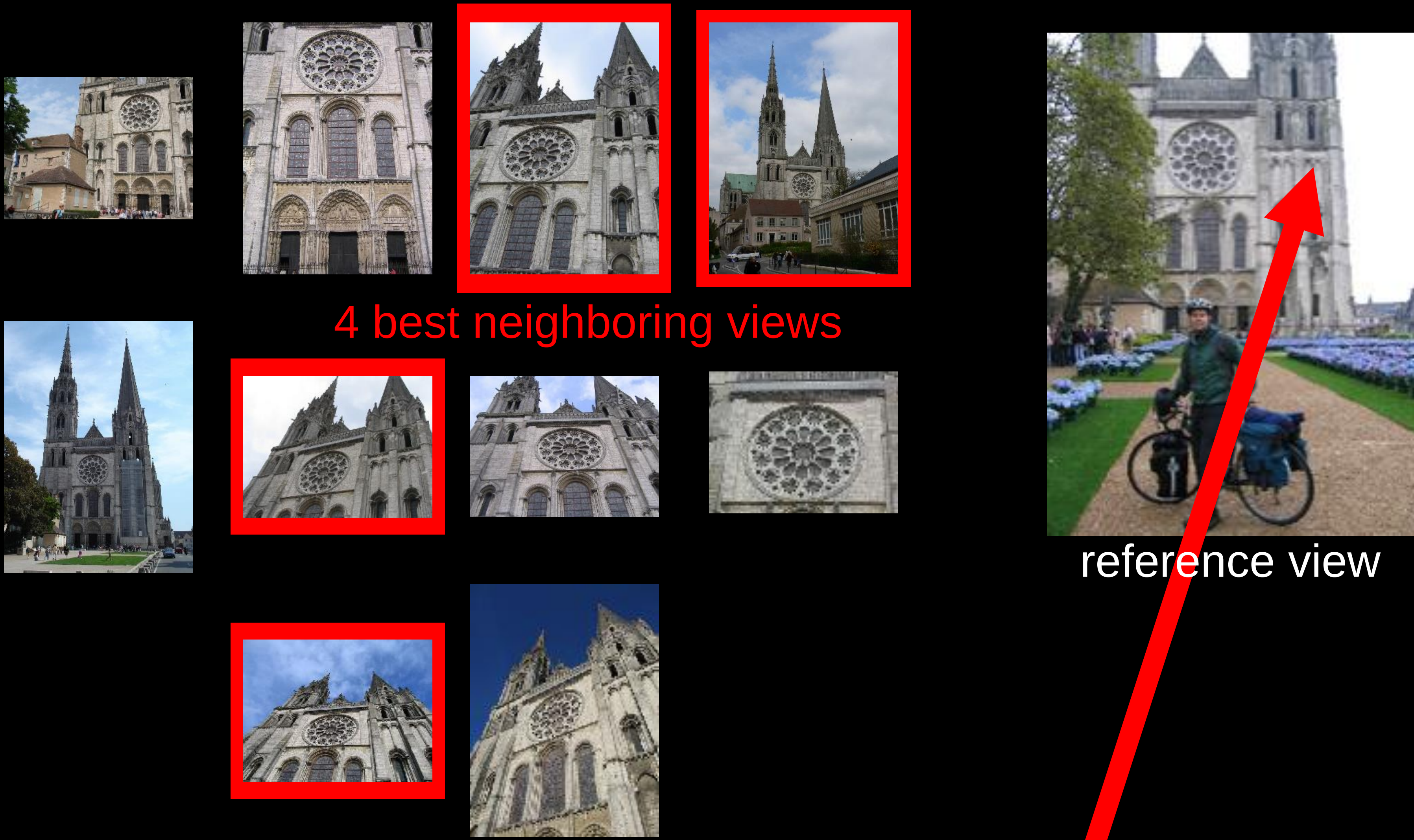
Local view selection

- Automatically select neighboring views for each point in the image
- Desiderata: good matches AND good baselines



Local view selection

- Automatically select neighboring views for each point in the image
- Desiderata: good matches AND good baselines

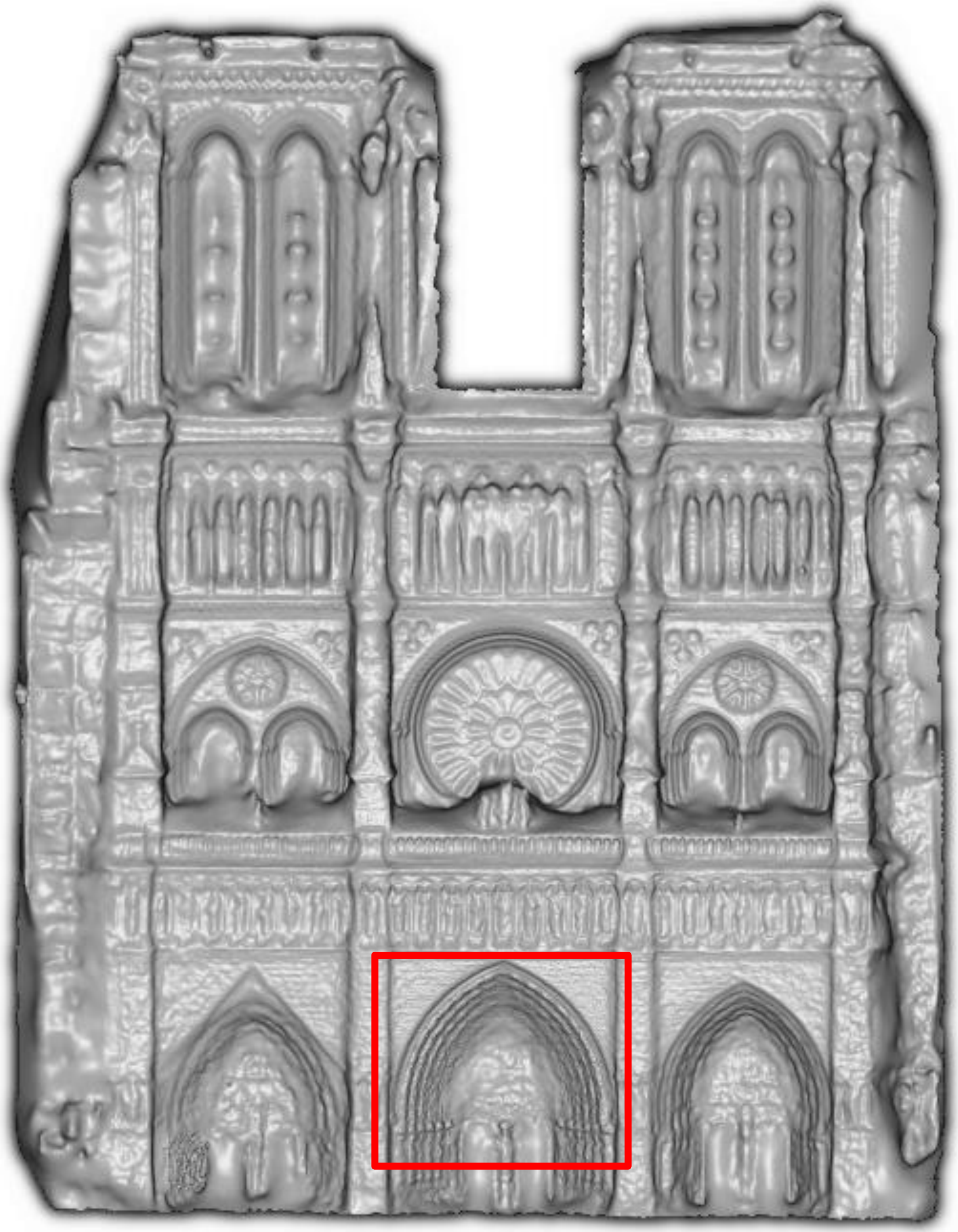


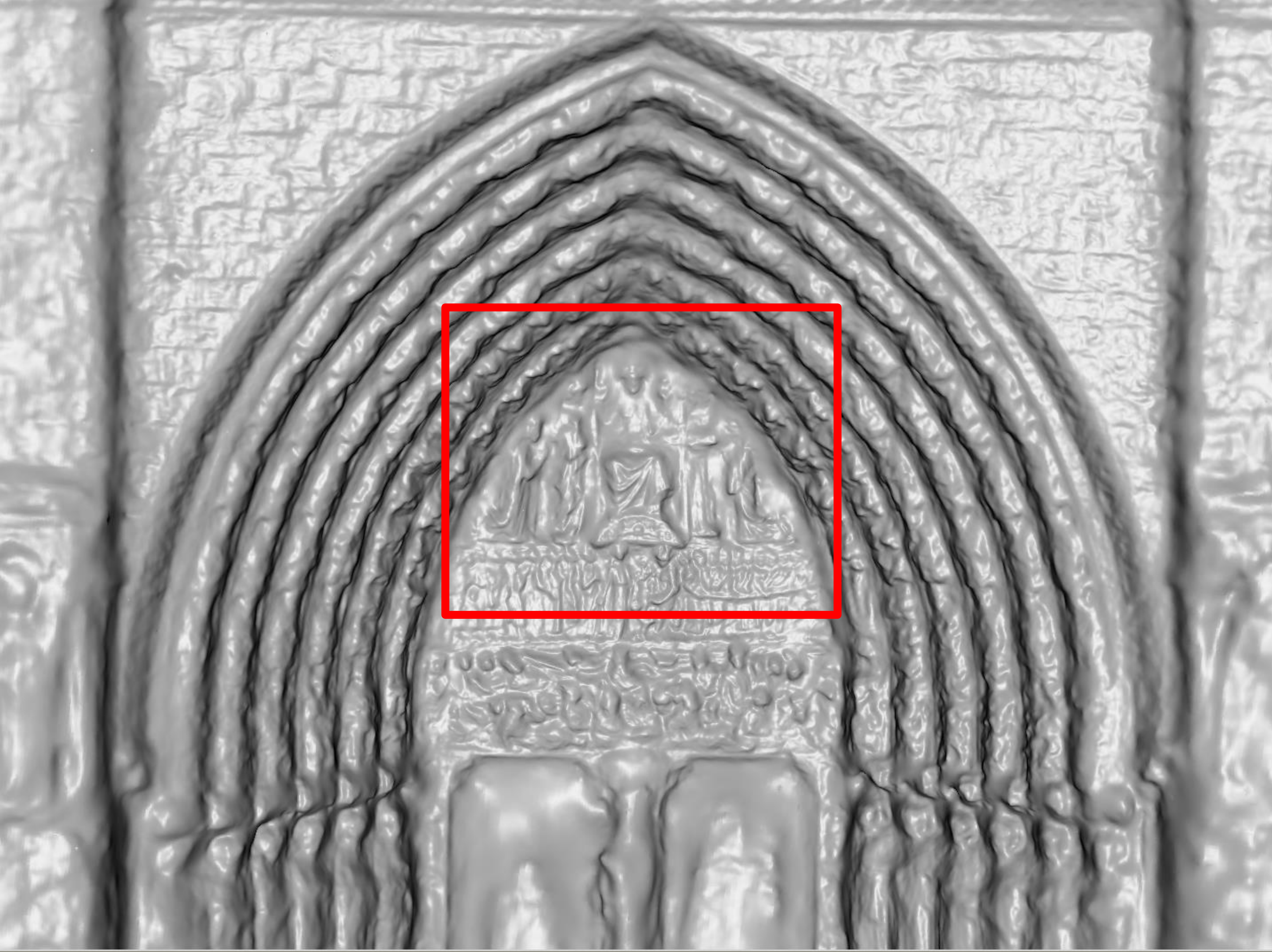
Local view selection

- Automatically select neighboring views for each point in the image
- Desiderata: good matches AND good baselines

Notre Dame de Paris

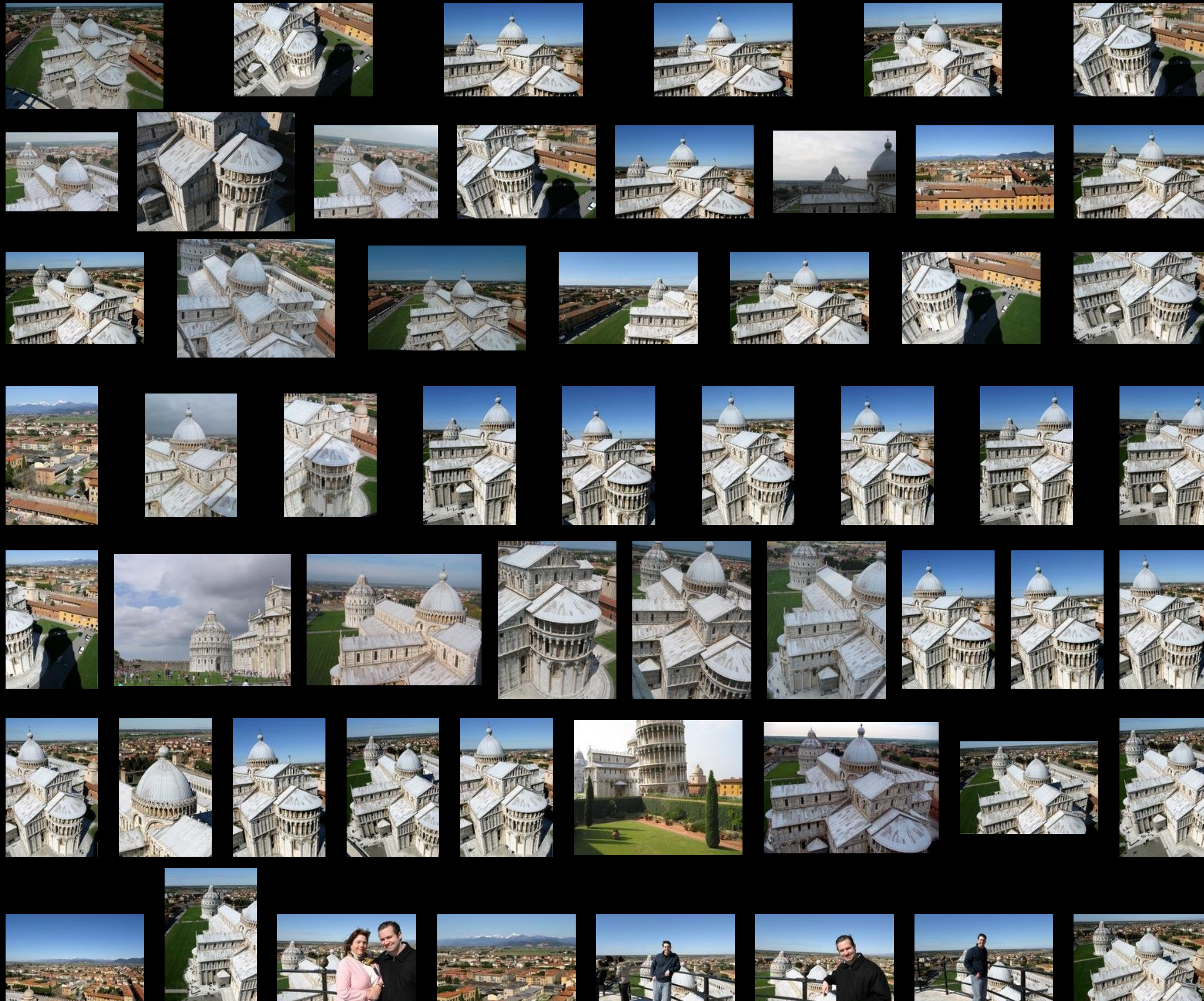
653 images
313 photographers



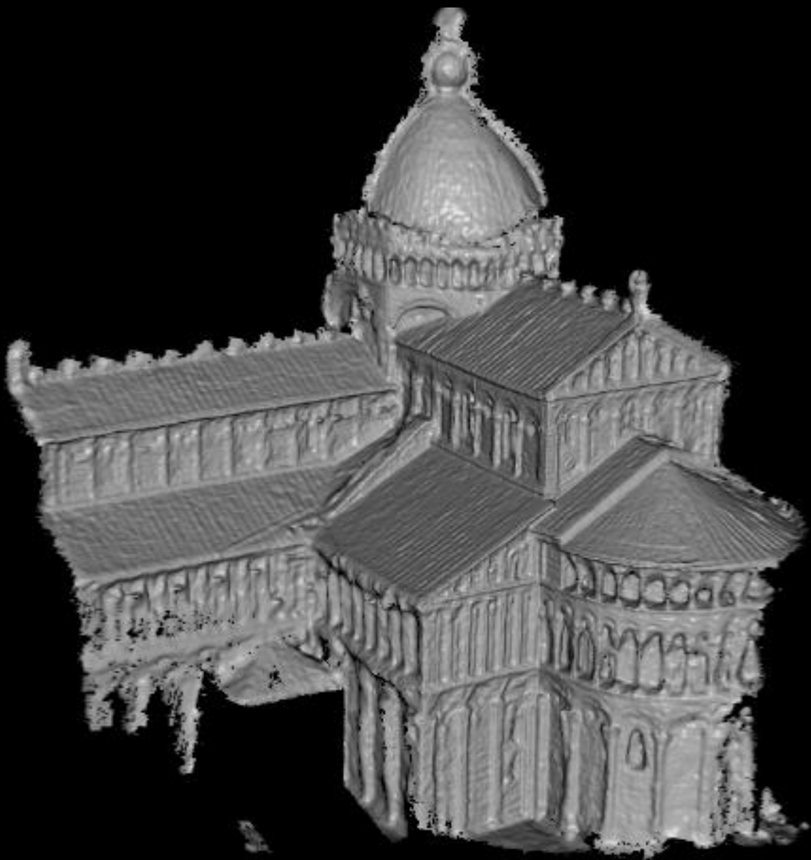


129 *Flickr* images taken by 98 photographers

merged model of Venus de Milo



56 *Flickr* images taken by 8 photographers



merged model of Pisa Cathedral

Accuracy compared to laser scanned model:
90% of points within *0.25%* of ground truth

How can Deep Learning Help?