Multiview Reconstruction

COS 429: Computer Vision

PRINCETON UNIVERSITY

Acknowledgments: T. Funkhouser,, S. Seitz

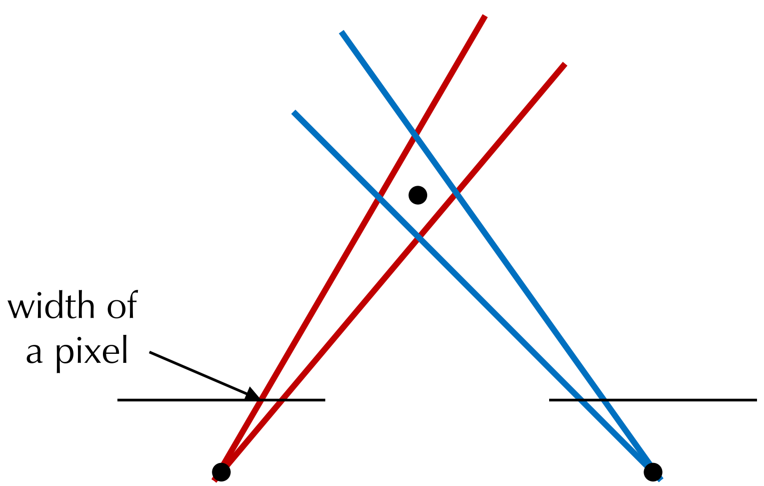
Multiview Stereo

- Given multiple images of the same object or scene, compute a representation of its 3D shape

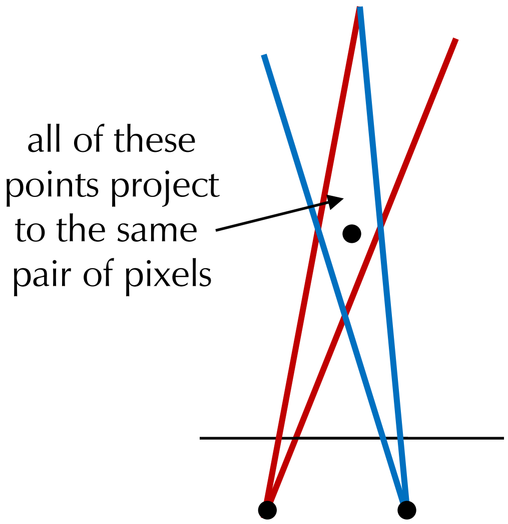


Why More Than 2 Views?

- Choosing a good baseline is hard
  - Too short – low accuracy
  - Too long – matching becomes hard



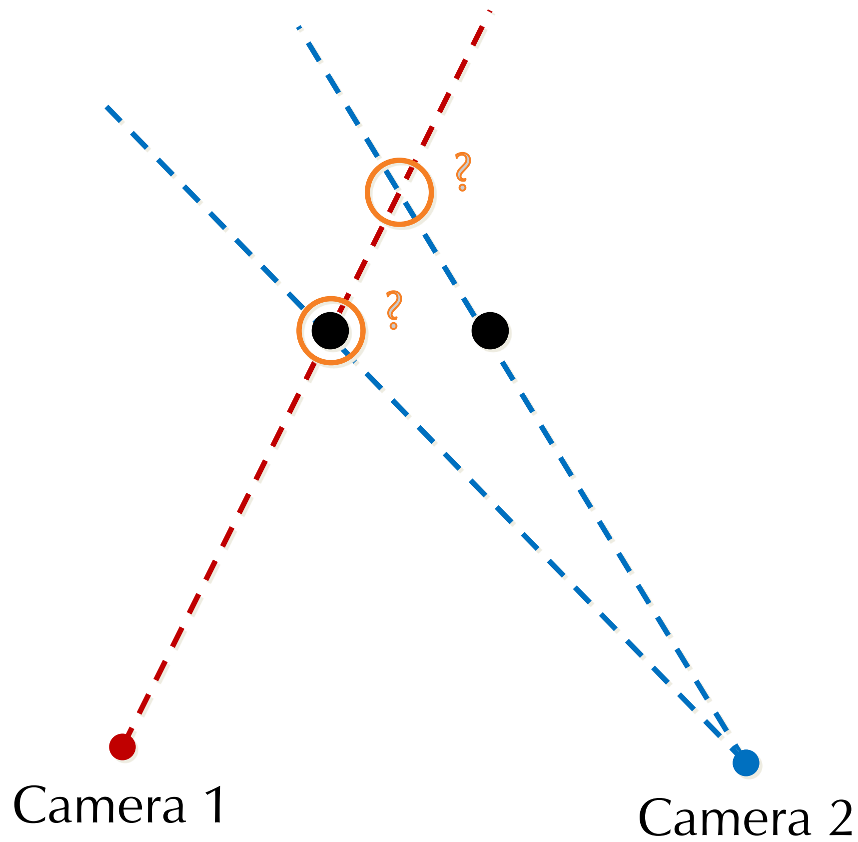
Large Baseline



Small Baseline

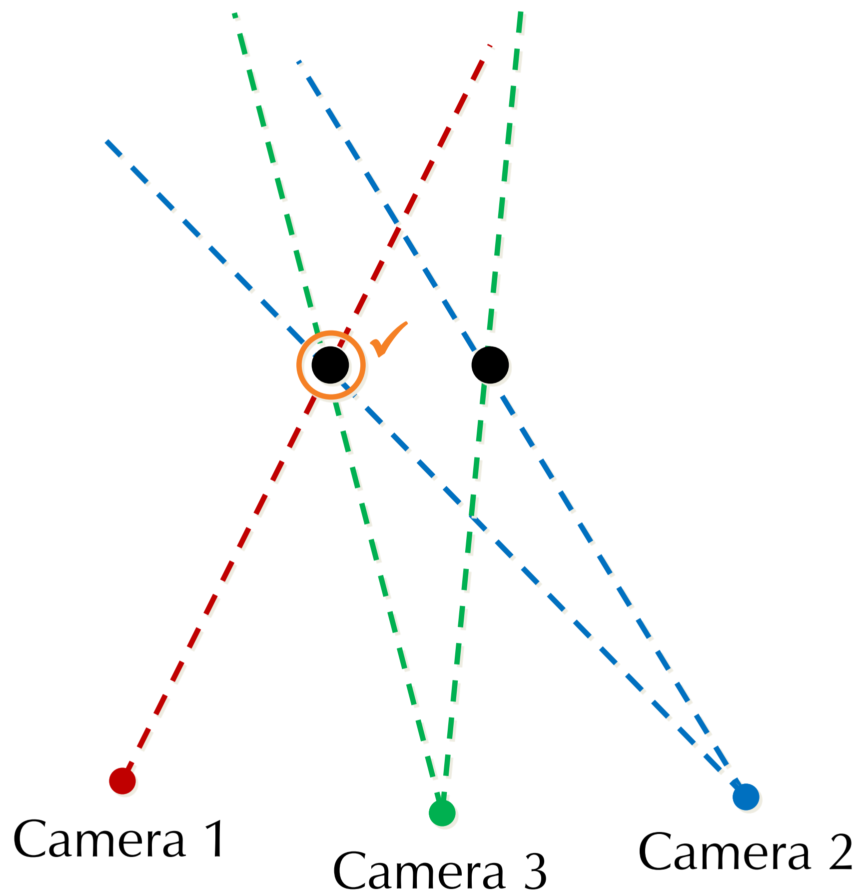
Why More Than 2 Views?

- Ambiguity with 2 views



Why More Than 2 Views?

- Ambiguity with 2 views – disambiguated by additional view



Outline

- Image-centric approaches
  - Multibaseline stereo
  - Plane-sweep stereo
- Volume-centric approaches
  - Silhouette carving
  - Voxel coloring
  - Space carving
- Surface-centric approaches
  - Feature detection + expansion/filtering
  - Mesh refinement

Outline

- Image-centric approaches
  - Multibaseline stereo
  - Plane-sweep stereo
- Volume-centric approaches
  - Silhouette carving
  - Voxel coloring
  - Space carving
- Surface-centric approaches
  - Feature detection + expansion/filtering
  - Mesh refinement

Trinocular Stereo

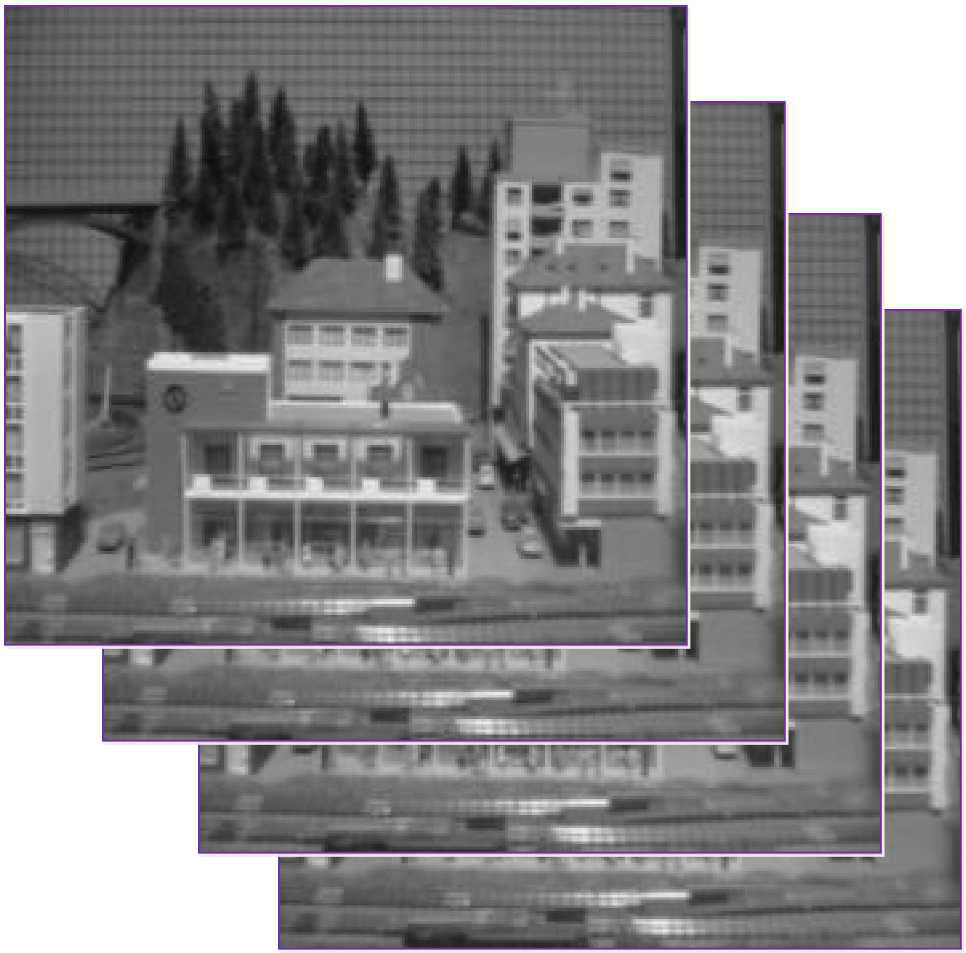
- Straightforward approach: use third view to eliminate bad correspondences
  - Pick 2 views, find correspondences
  - For each matching pair, reconstruct 3D point
  - Project point into 3rd image
  - If can't find correspondence near predicted location, reject

Multibaseline Stereo

More generally, for N views …

- Pick one reference view
- For each candidate depth
  - Compute sum of squared differences to all other views, assuming correct disparity for view
- Resolves ambiguities: only correct depths will "constructively interfere"

Multibaseline Stereo



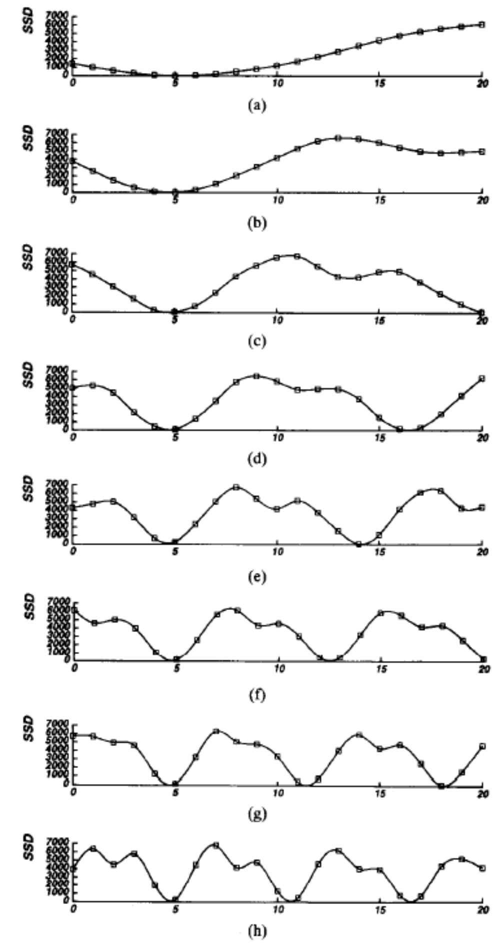
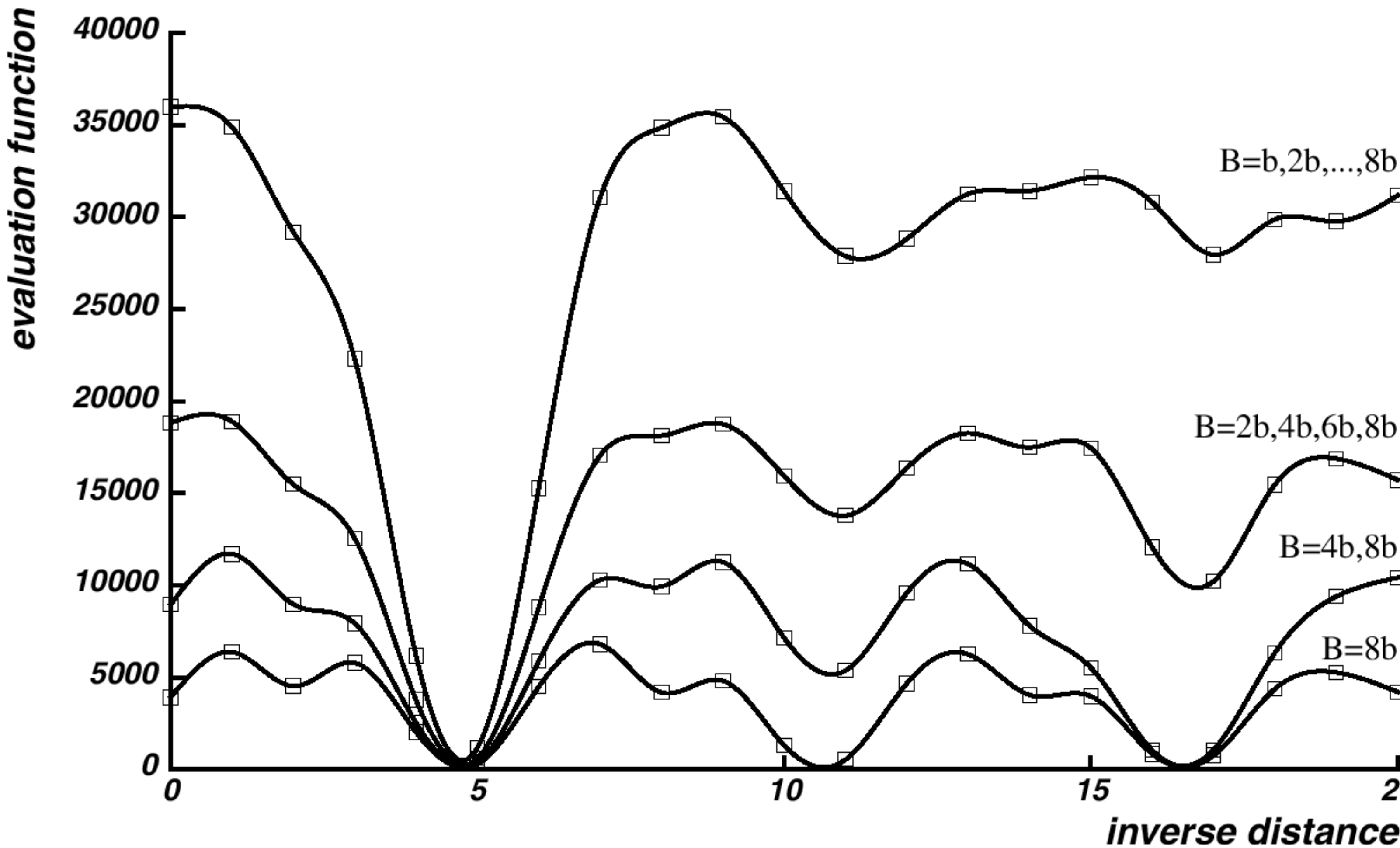


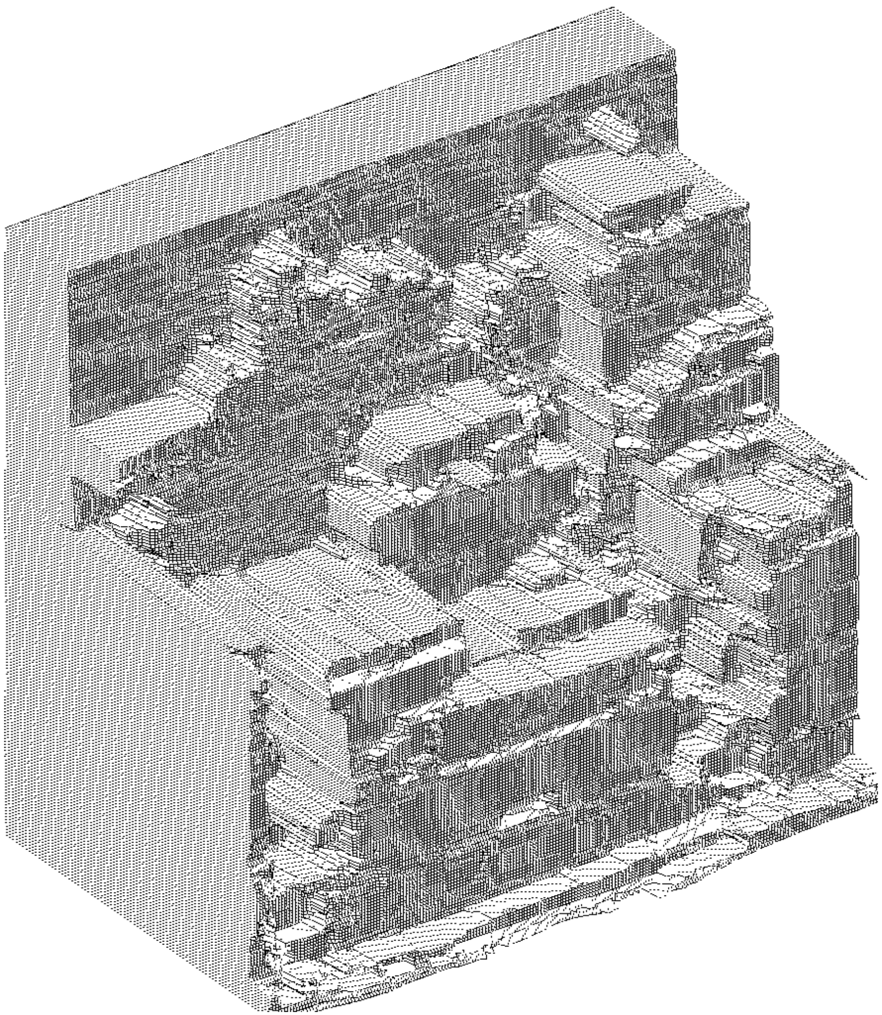
Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

Multibaseline Stereo

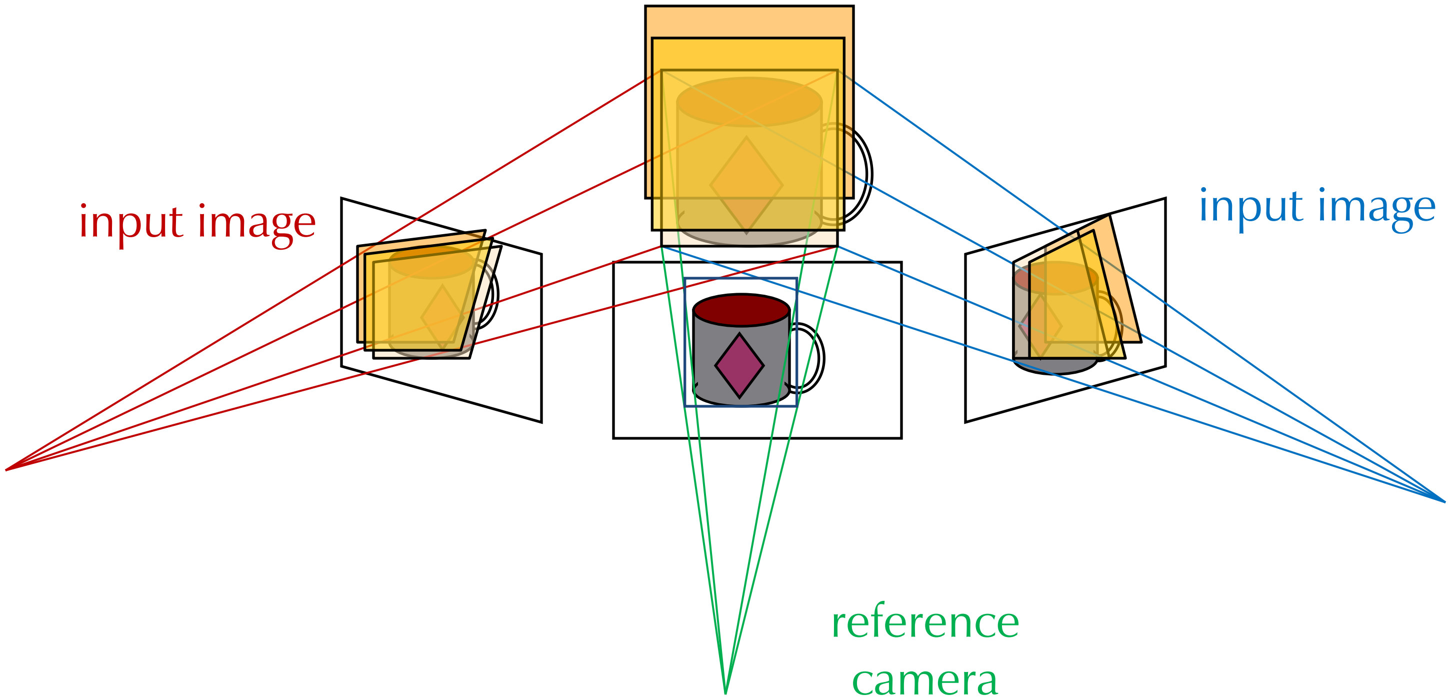


[Okutami & Kanade]

Multibaseline Stereo Results



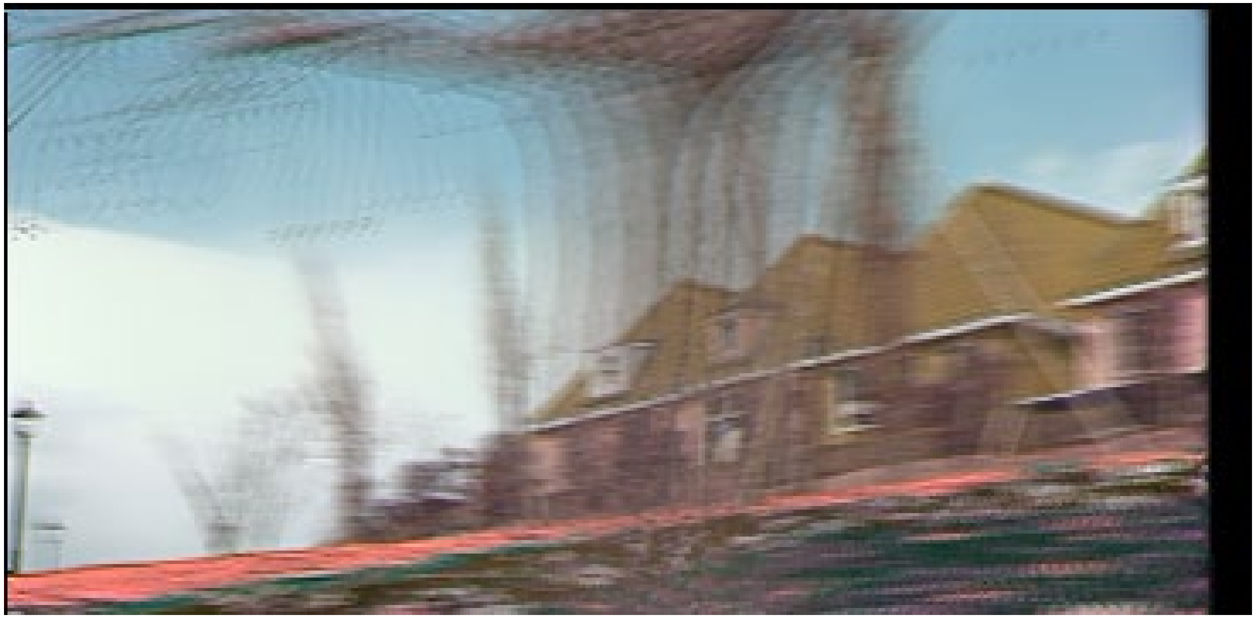
Plane Sweep Stereo



Each plane defines a homography warping each input image into the reference view

Plane Sweep Stereo

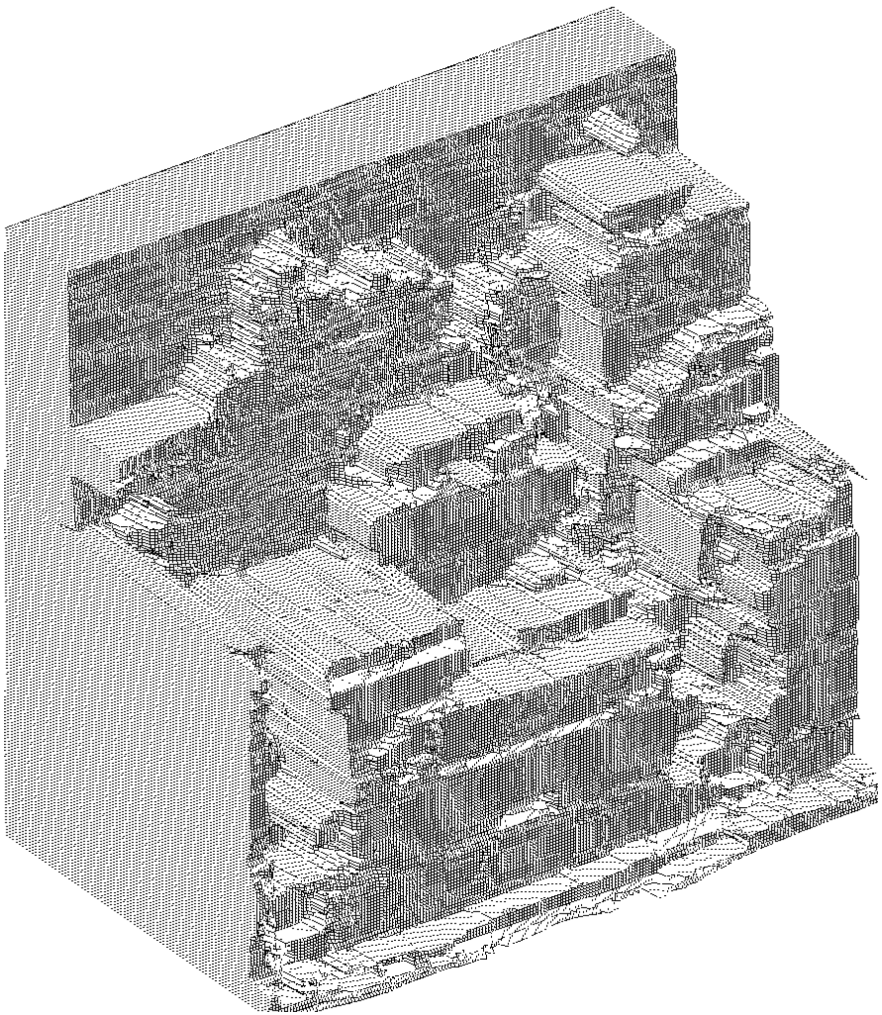
For each pixel, select the depth that gives the lowest variance



Problems with these approaches

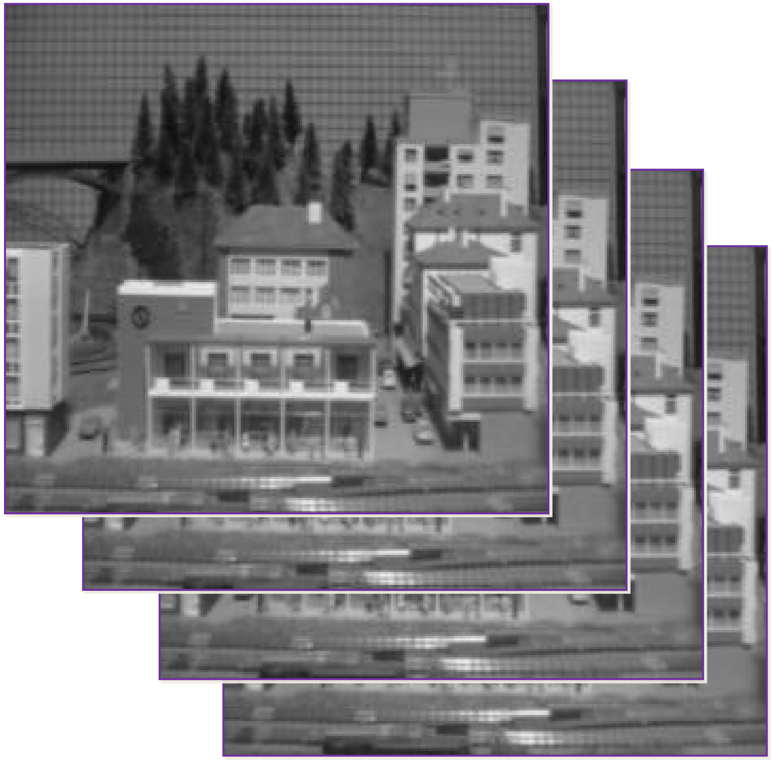
- Limited types of 3D surfaces
  - Have to pick a reference view
- No consideration for visibility
  - With many cameras / large baseline, occlusion becomes likely
  - Contributes incorrect values to error function

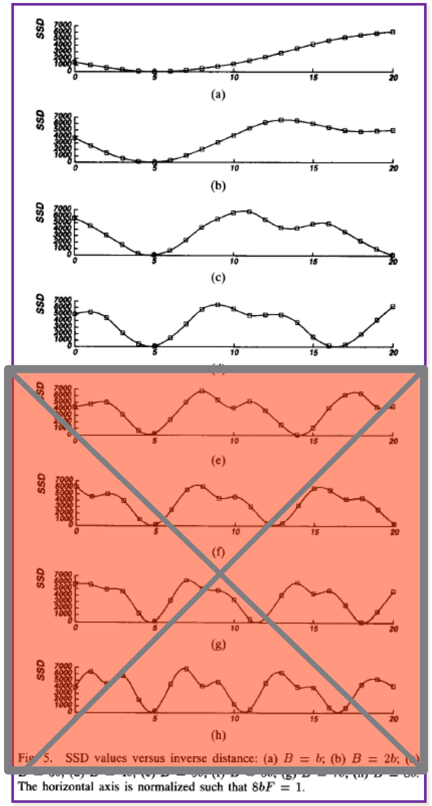
Reference View Problem



Visibility Problem

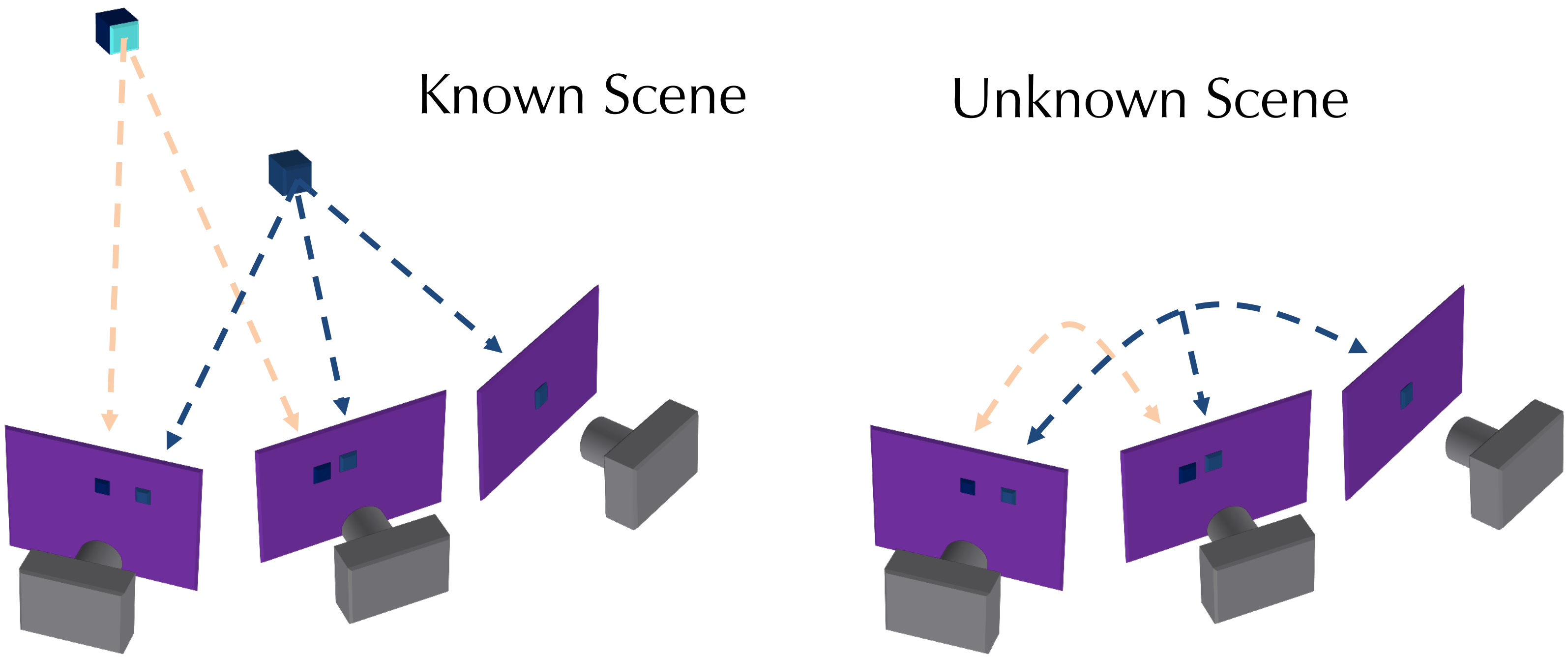
- For larger baselines, occlusion is common





Visibility Problem

- Which scene points are seen in which images?



Forward Visibility
known scene

Inverse Visibility
known images

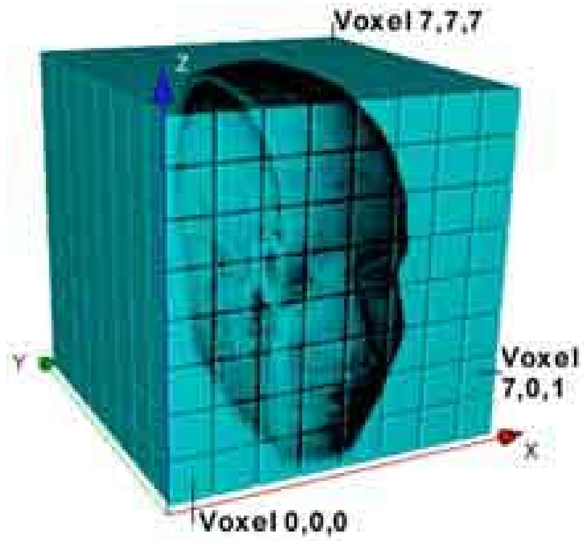
Snavely

Outline

- Image-centric approaches
  - Multibaseline stereo
  - Plane-sweep stereo
- Volume-centric approaches
  - Silhouette carving
  - Voxel coloring
  - Space carving
- Surface-centric approaches
  - Feature detection + expansion/filtering
  - Mesh refinement

Volume-Centric Multiview Approaches

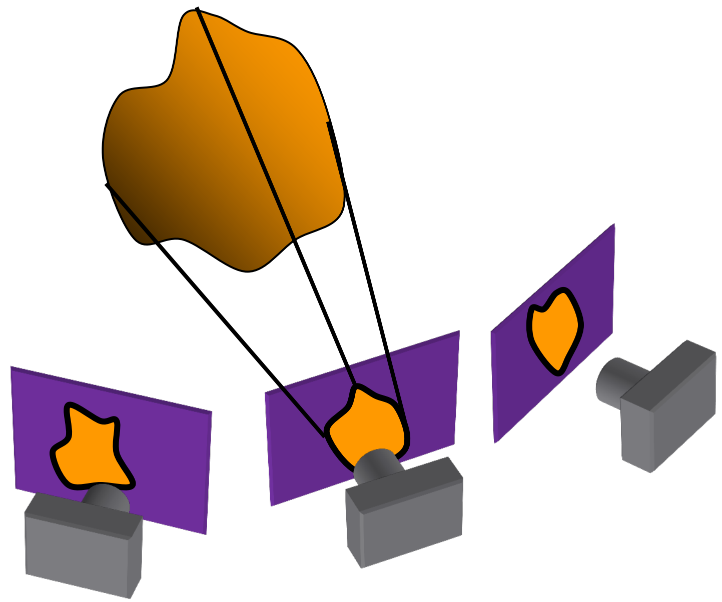
- Compute photoconsistency at 3D points
- Typically use discretized volume (voxel grid)
- For each voxel, predict whether
  3D point is on surface,
  or inside/outside object



Silhouette Carving

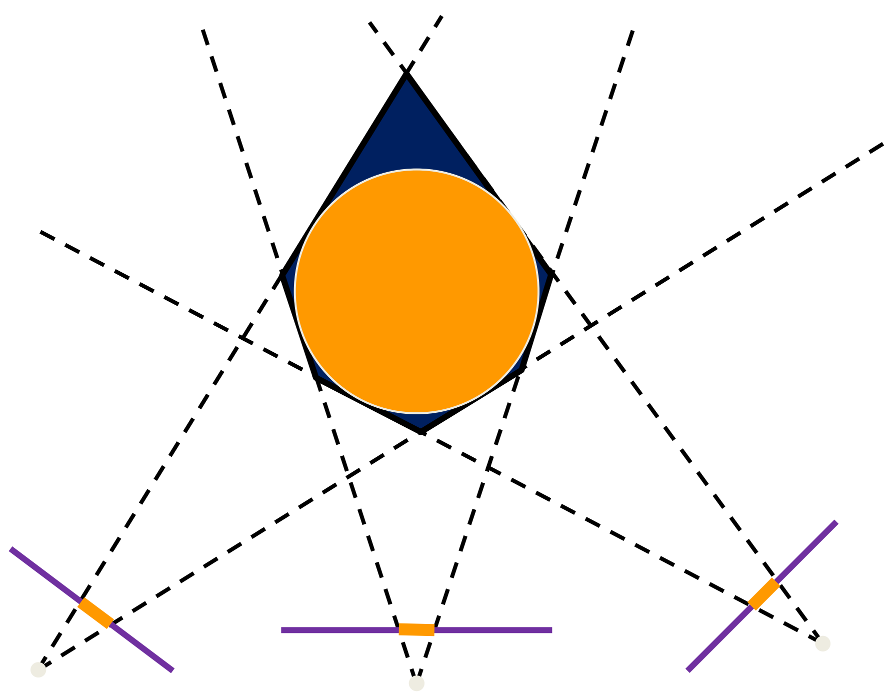
- Find silhouettes in all images
- Exact version:
  - Back-project all silhouettes, find intersection

Binary Images →



Silhouette Carving

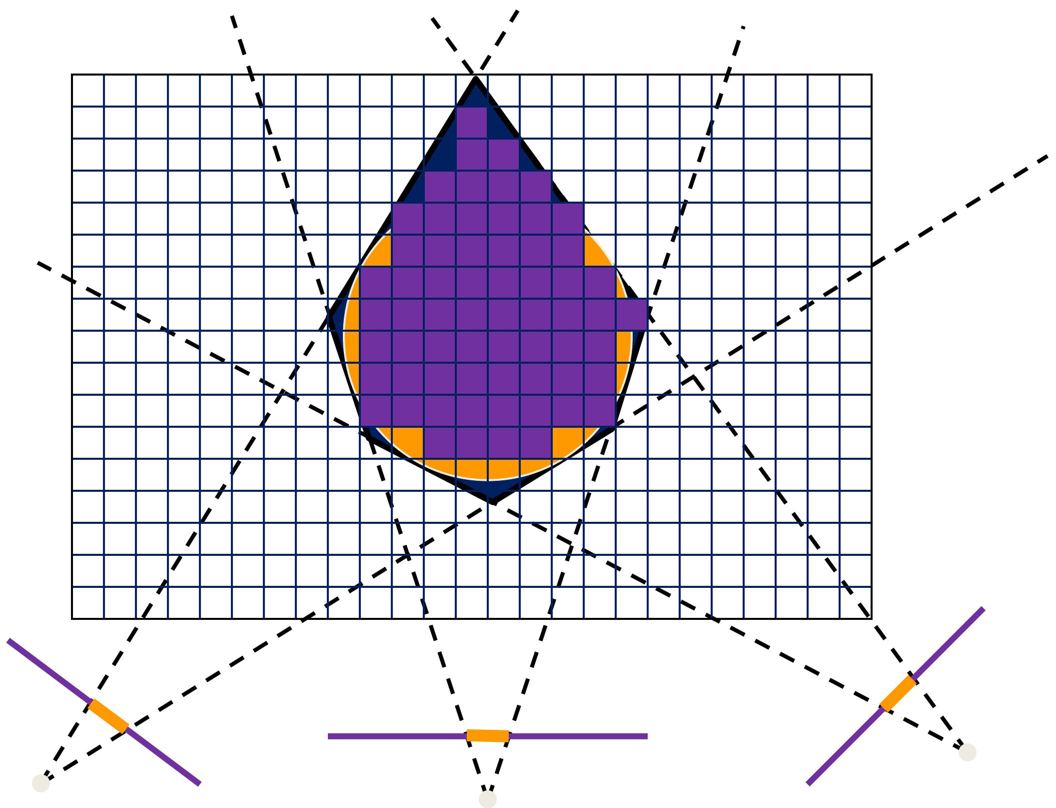
- Find silhouettes in all images
- Exact version:
  - Back-project all silhouettes, find intersection



Silhouette Carving

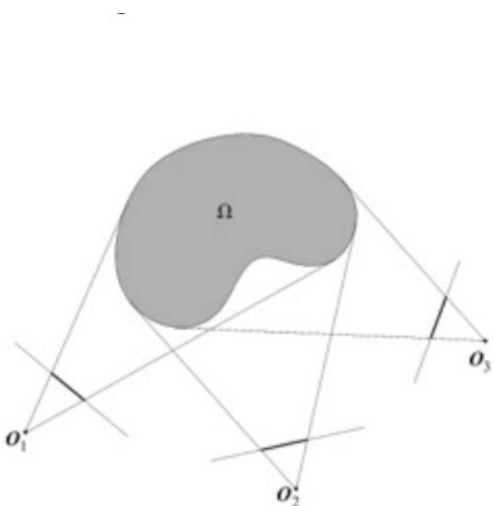
- Discrete version:
  - Loop over all voxels in some volume
  - If projection into images lies inside all silhouettes, mark as occupied
  - Else mark as free

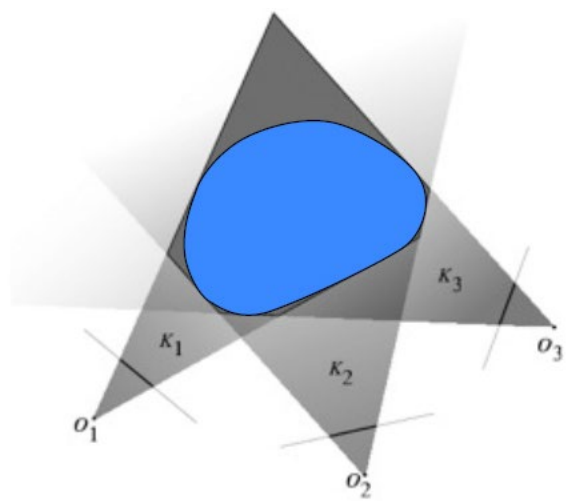
Silhouette Carving



Silhouette Carving

- Limit of silhouette carving is *visual hull*
- In general not the same as object
  - Can't recover "pits" in object
- Not the same as convex hull





Silhouette Carving

- The visual hull is a good starting point for better algorithms (that consider photo-consistency)
  - Easy to compute (if segmentation is good!)
  - Tight outer boundary of the object
  - Parts of the visual hull (rims) already lie on the surface and are already photo-consistent

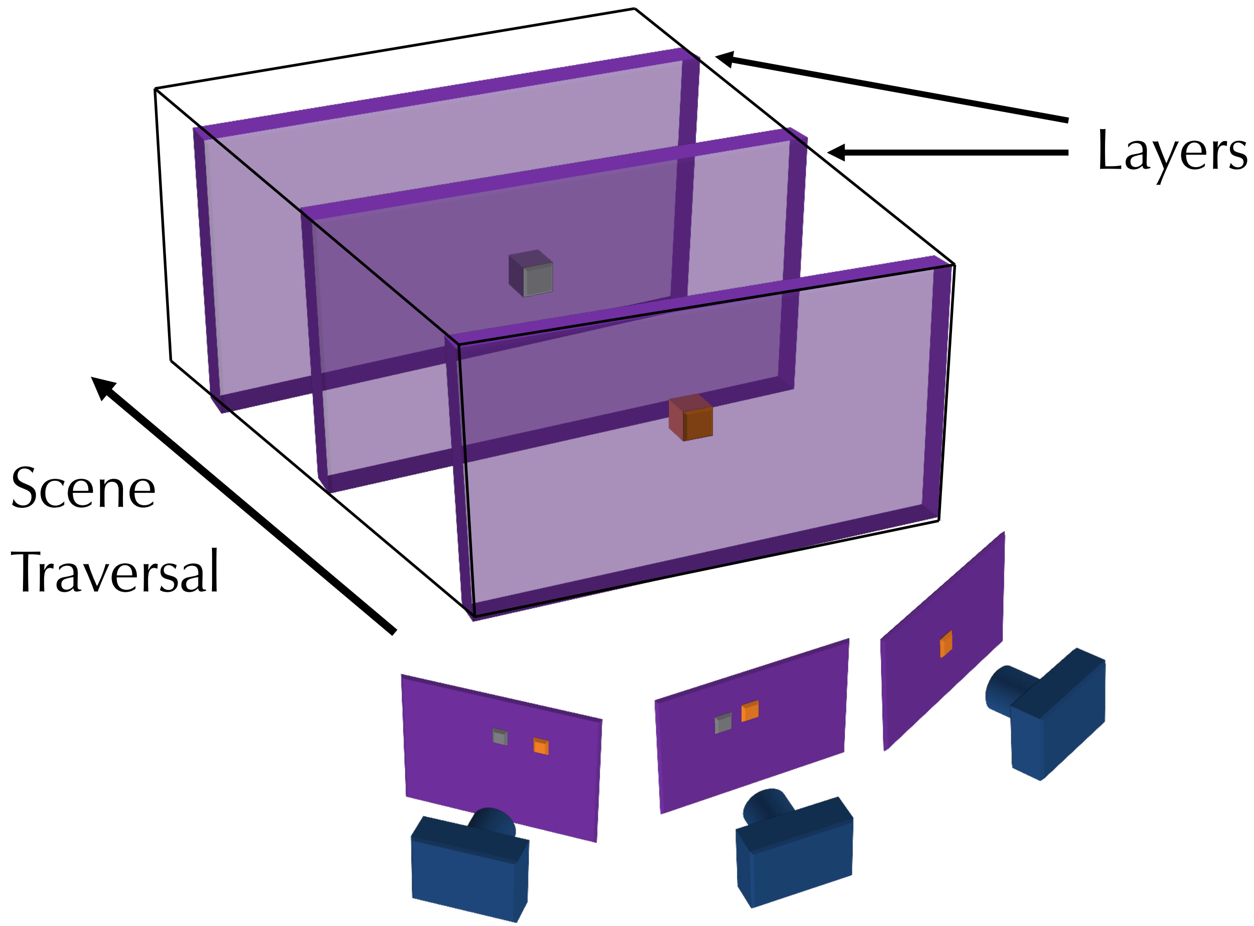
Voxel Coloring

- Basic idea:
  - Project each voxel into each image in which it is visible
  - If colors in images agree, mark voxel with color
  - Else, mark voxel as empty
- Agreement of colors based on comparing standard deviation of colors to threshold

Voxel Coloring and Occlusion

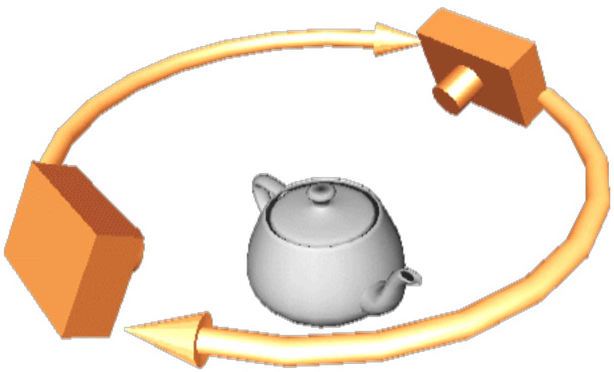
- Problem: which voxels are visible?
- Solution, part 1: constrain camera views
  - When a voxel is considered, necessary occlusion information must be available
  - Sweep occluders before occludees
  - Constrain camera positions to allow this sweep

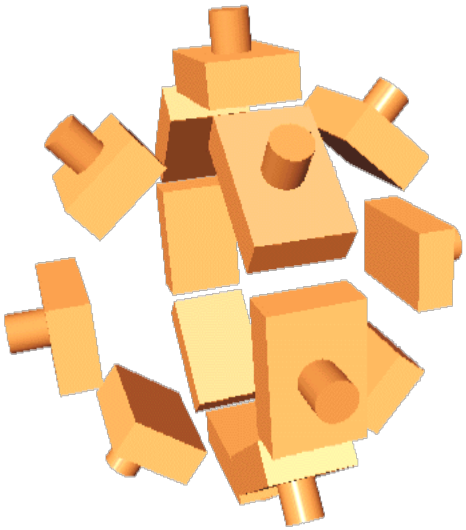
Voxel Coloring Sweep Order



Seitz

Voxel Coloring Camera Positions



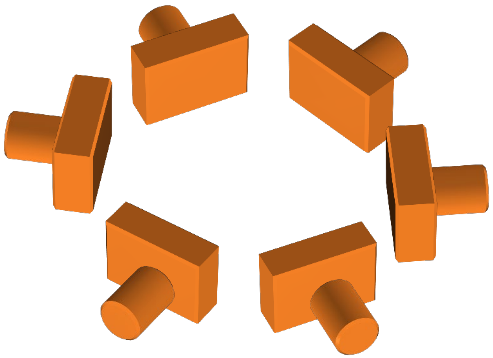


Inward-looking
    Cameras above scene

Outward-looking
    Cameras inside scene

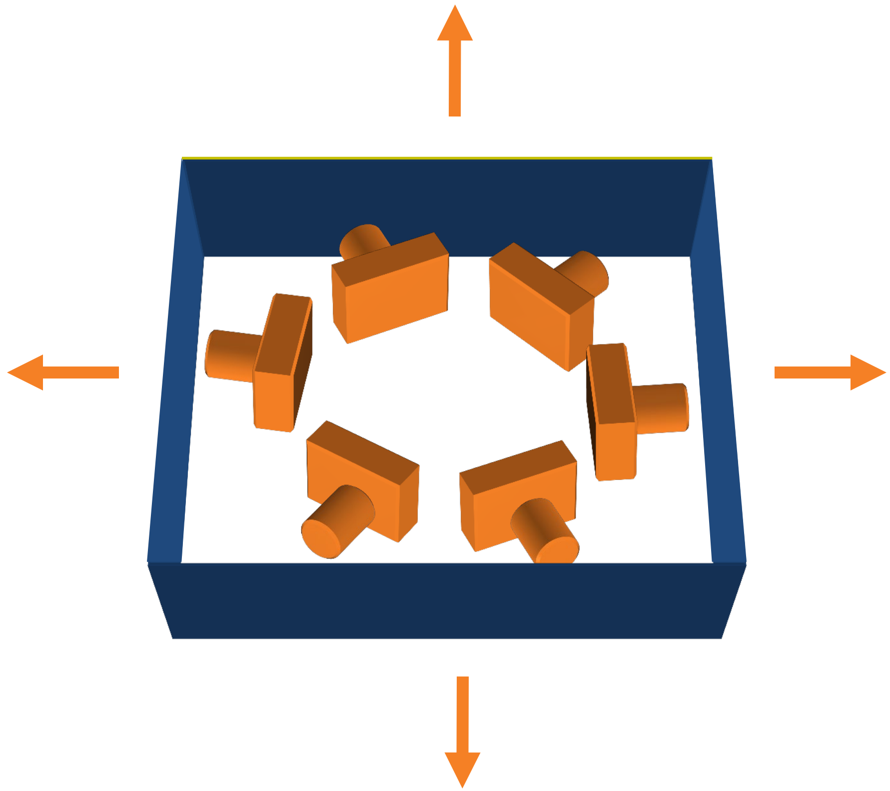
Seitz

Panoramic Depth Ordering



- Cameras oriented in many different directions
- Planar depth ordering does not apply

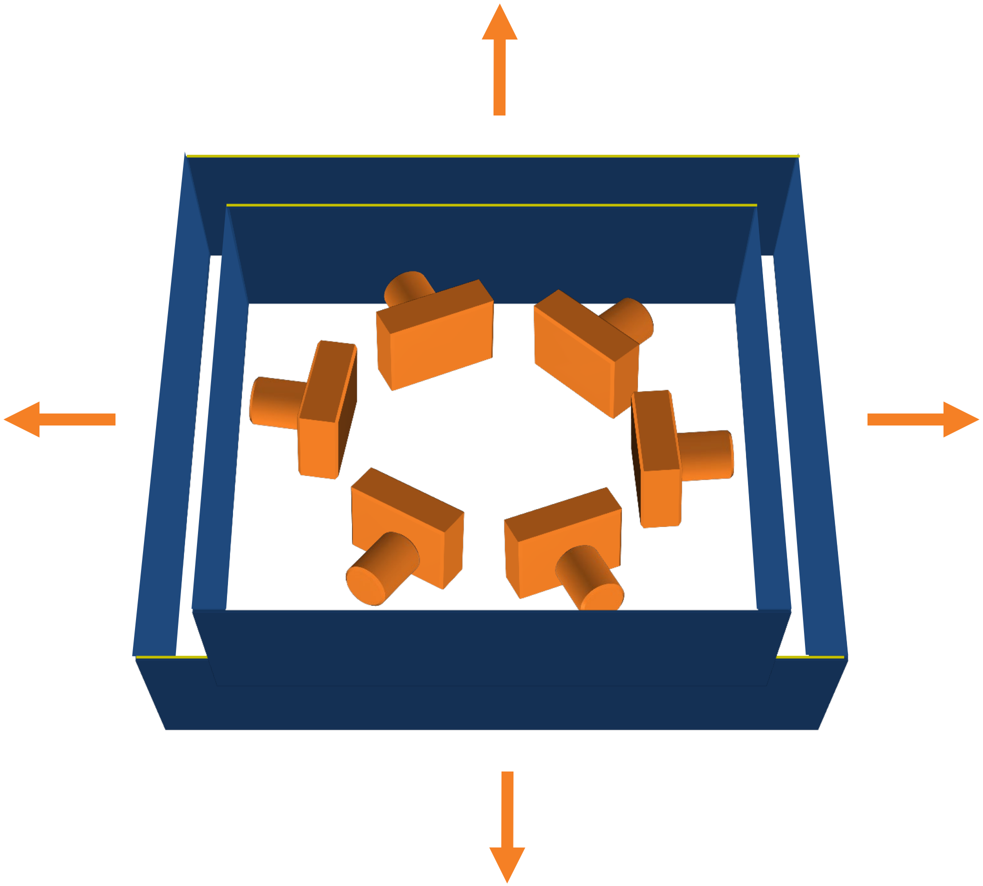
Panoramic Depth Ordering



Layers radiate outwards from cameras

Seitz

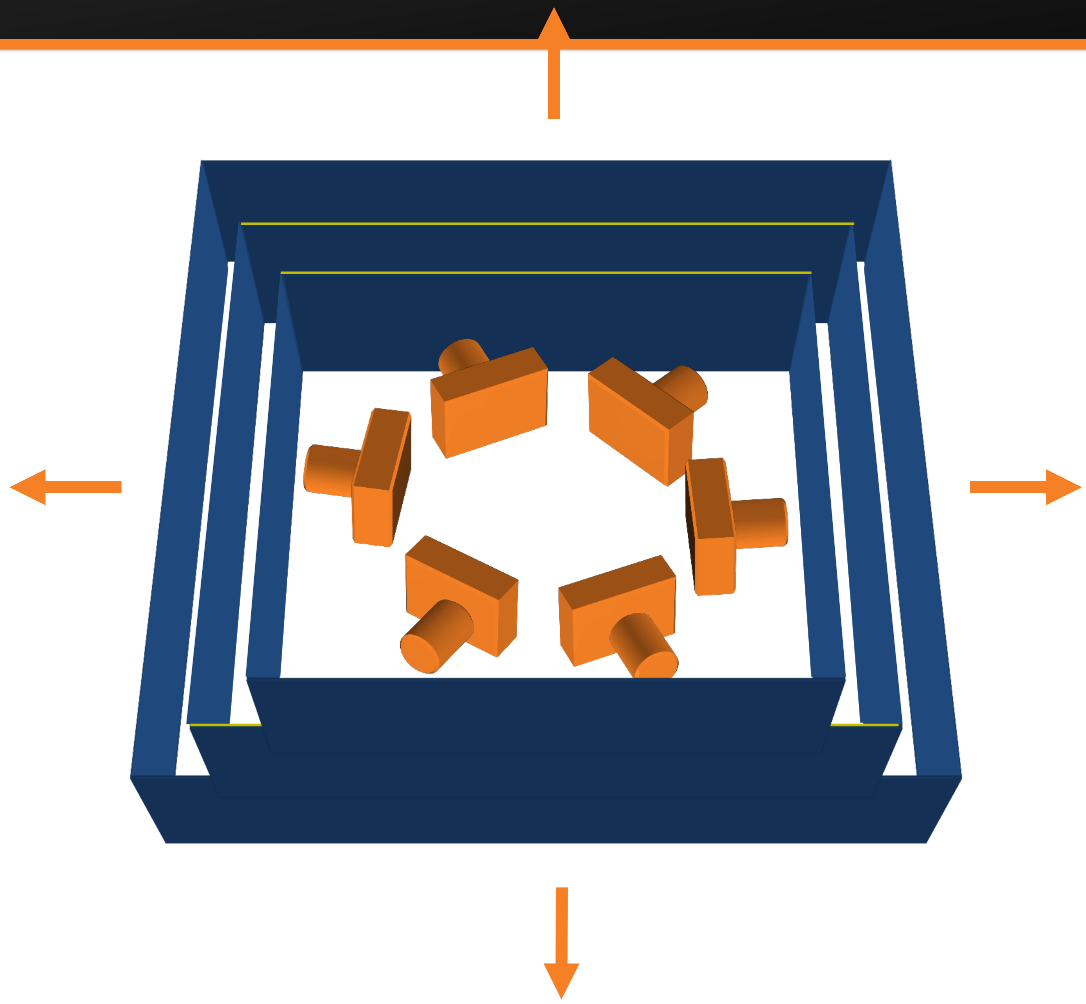
Panoramic Depth Ordering



Layers radiate outwards from cameras

Seitz

Panoramic Depth Ordering



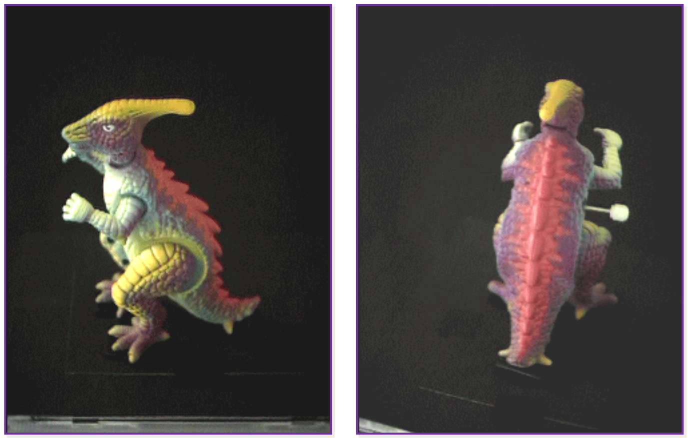
Layers radiate outwards from cameras

Seitz

Voxel Coloring and Occlusion

- Solution, part 2: per-image mask of which pixels have been used
  - Each pixel only used once
  - Mask filled in as sweep progresses

Voxel Coloring Results



Selected Dinosaur Images

Selected Flower Images

- Calibrated Turntable
- 360° rotation (21 images)

Seitz

Voxel Coloring Results

Dinosaur Reconstruction
72 K voxels colored
7.6 M voxels tested

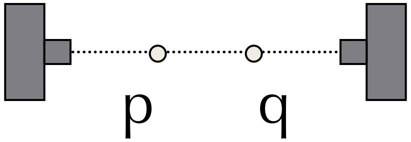
Flower Reconstruction
70 K voxels colored
7.6 M voxels tested

Seitz

Voxel Coloring Results

- With texture: good results
- Without texture: regions tend to "bulge out"
  - Voxels colored at earliest time at which projection into images is consistent
  - Model good for re-rendering: image will look correct for viewpoints near the original ones

Limitations of Voxel Coloring



- A view-independent depth order may not exist
- Need more powerful general-case algorithms
  - Unconstrained camera positions
  - Unconstrained scene geometry/topology

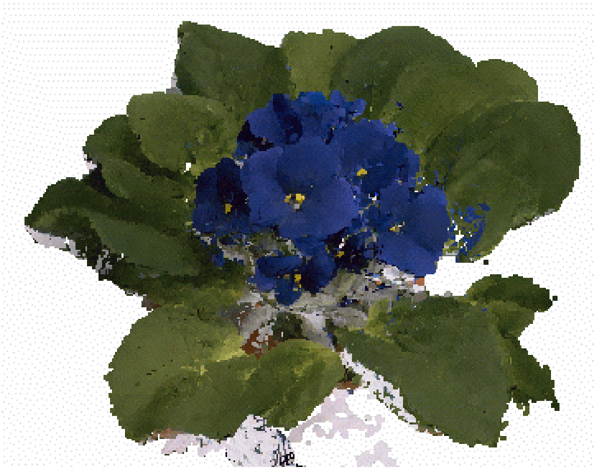
Multi-Pass Plane Sweep

- Sweep planes in each of 6 principal directions
- Consider cameras on only one side of plane
- Repeat until convergence
  - Carved voxels stay carved
  - New voxels may be carved away on future passes

Space Carving Results: African Violet

Input Image (1 of 45)

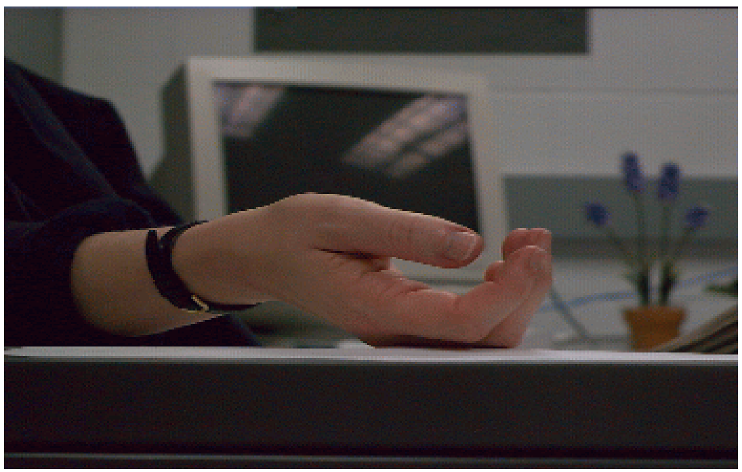
Reconstruction



Reconstruction

Reconstruction

Space Carving Results: Hand

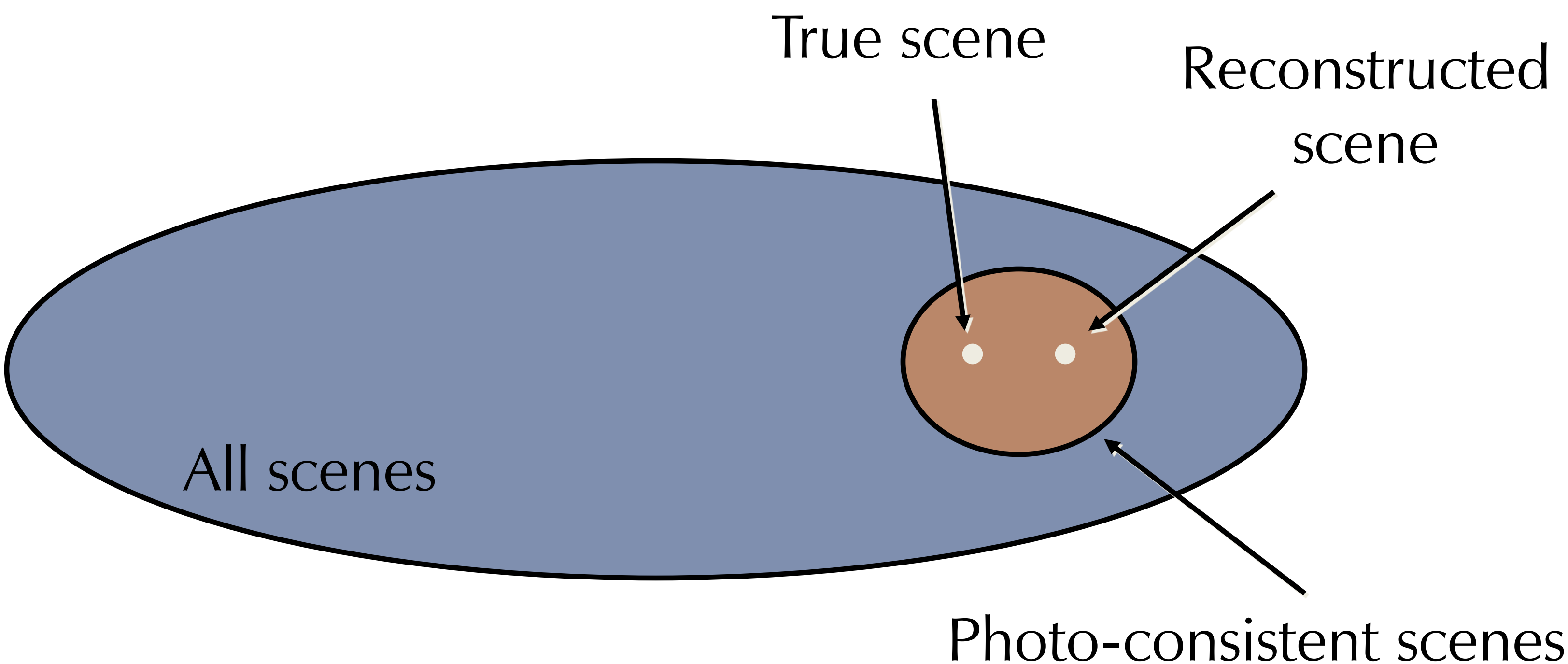


Input Image
(1 of 100)

Views of Reconstruction

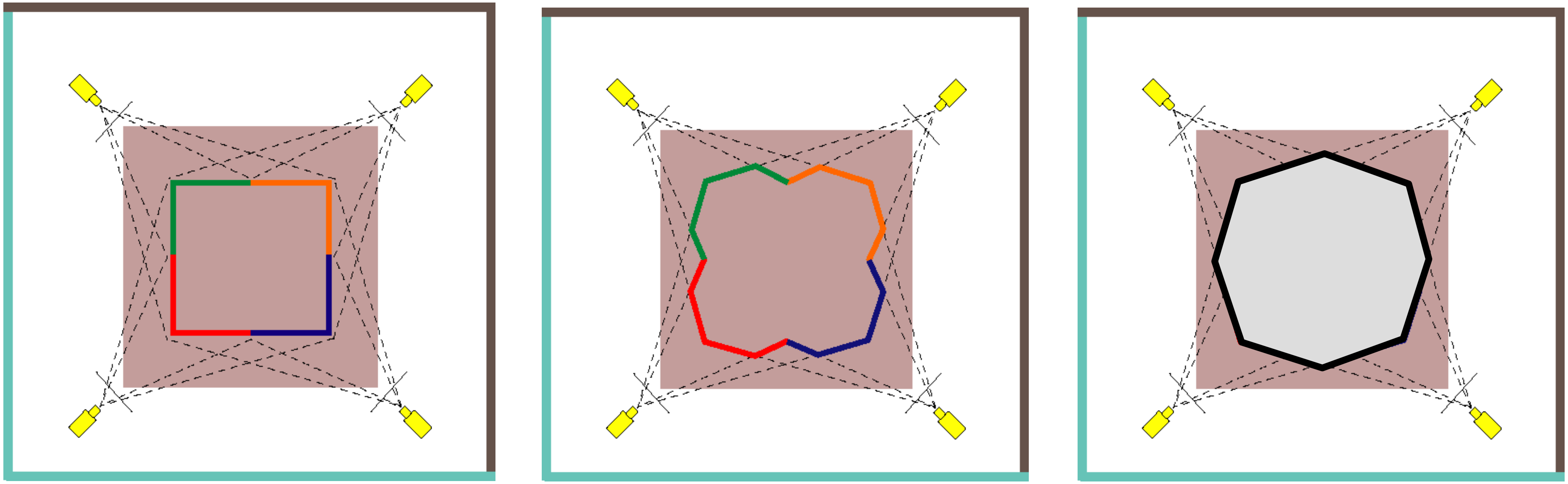
Caveat

- Result: not necessarily correct scene
- Many scenes may be photoconsistent with the input images



Caveat

- Photo-consistency vs. silhouette-consistency

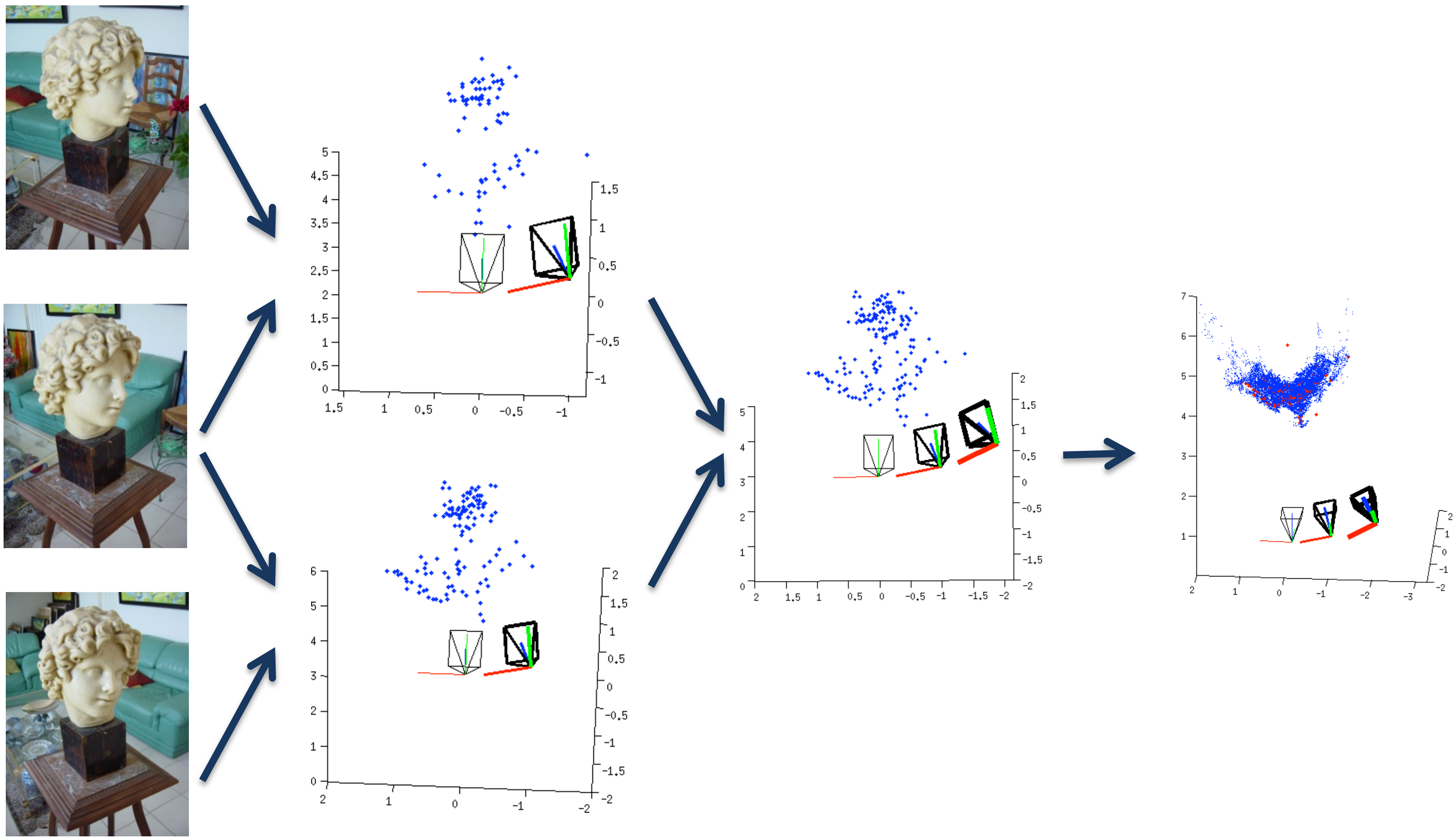


True Scene Photo Hull Visual Hull

Outline

- Image-centric approaches
  - Multibaseline stereo
  - Plane-sweep stereo
- Volume-centric approaches
  - Silhouette carving
  - Voxel coloring
  - Space carving
- Surface-centric approaches
  - Feature detection + expansion/filtering
  - Mesh refinement

Patch-Based Approaches



Structure from Motion (SFM) Multi-view Stereo (MVS)

Patch-Based Approaches

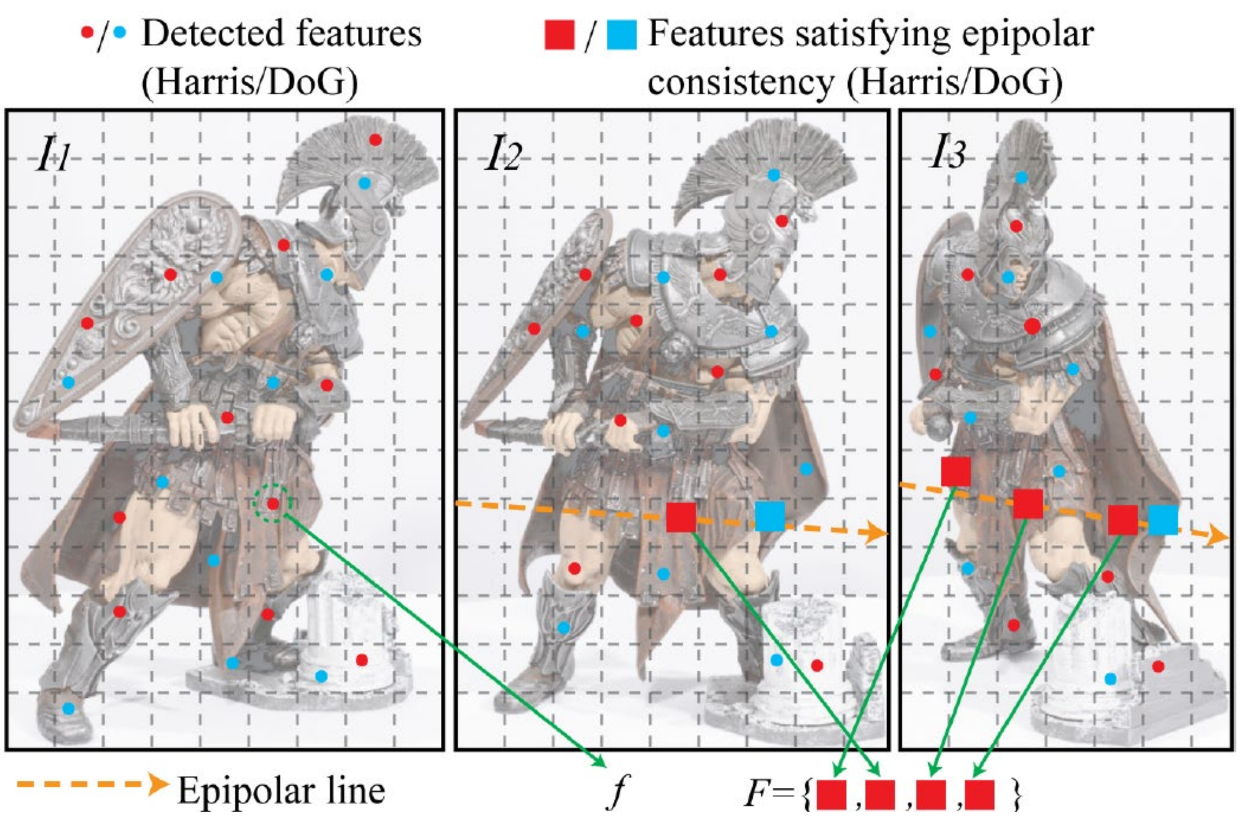
- Detect feature correspondences, and then expand/filter based on consistency in other views

One Input Features Correspondences Expansion Final Surface

Furukawa

Patch-Based Approaches

1) Detect features, find correspondences



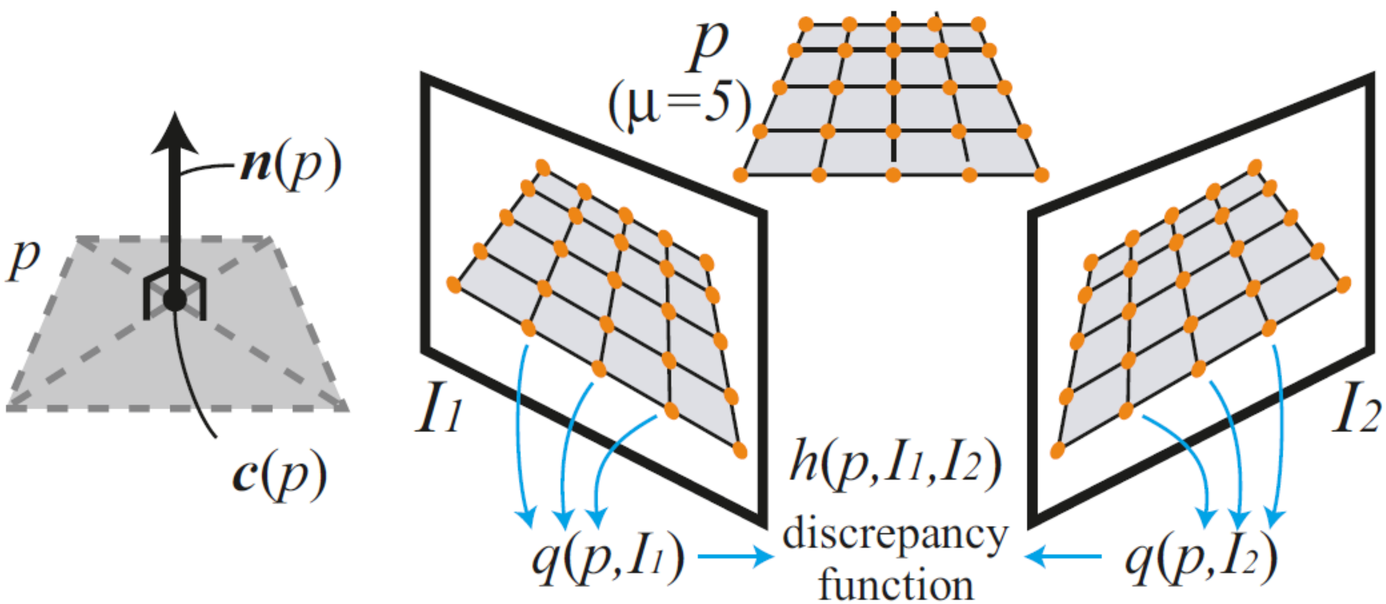
Furukawa

Structure from Motion

- "Limiting case" of multibaseline stereo
- Track features in a video sequence
- For n frames and f features, have $2{\cdot}n{\cdot}f$ knowns, $6{\cdot}n+3{\cdot}f$ unknowns
  - Can solve for feature positions and camera extrinsics

Patch-Based Approaches

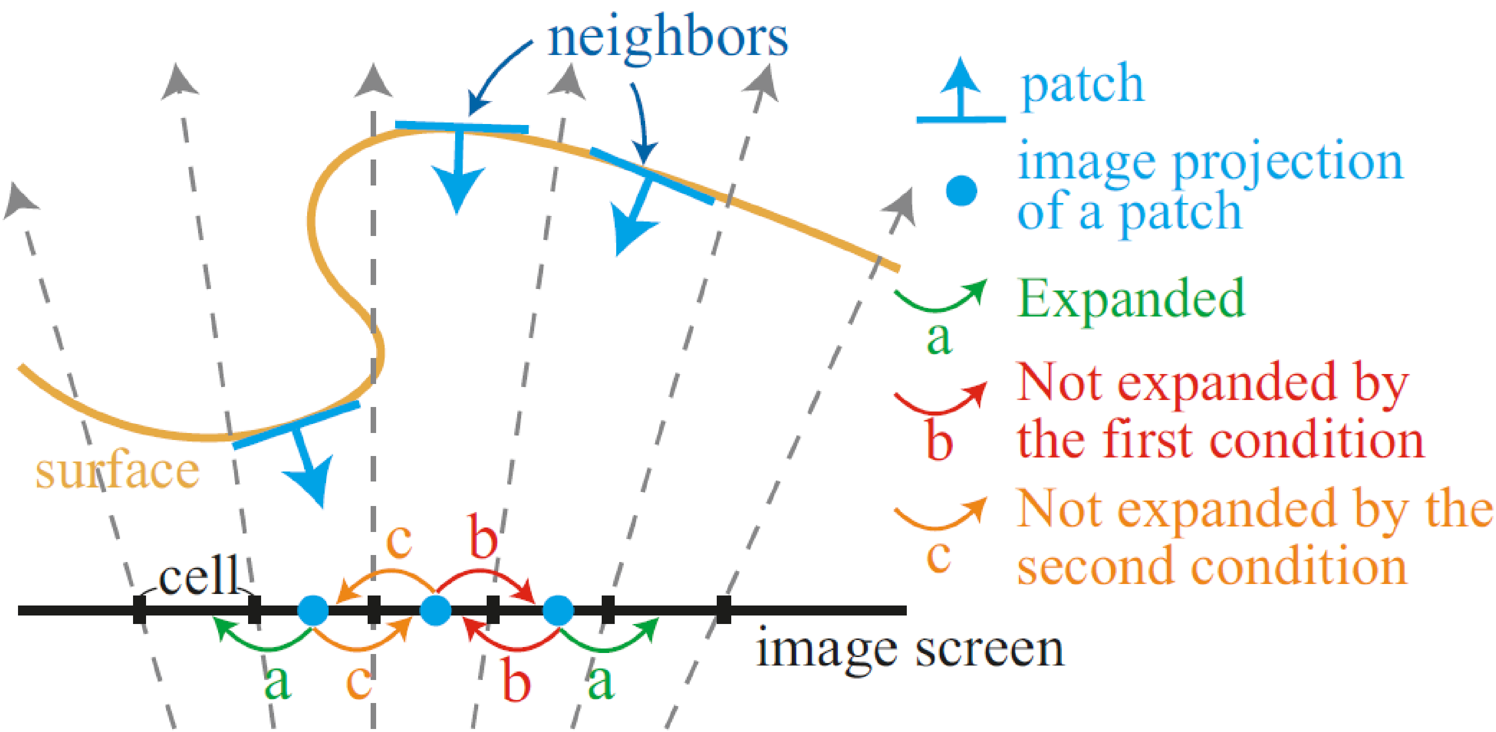
2) Construct seed patches



Furukawa

Patch-Based Approaches

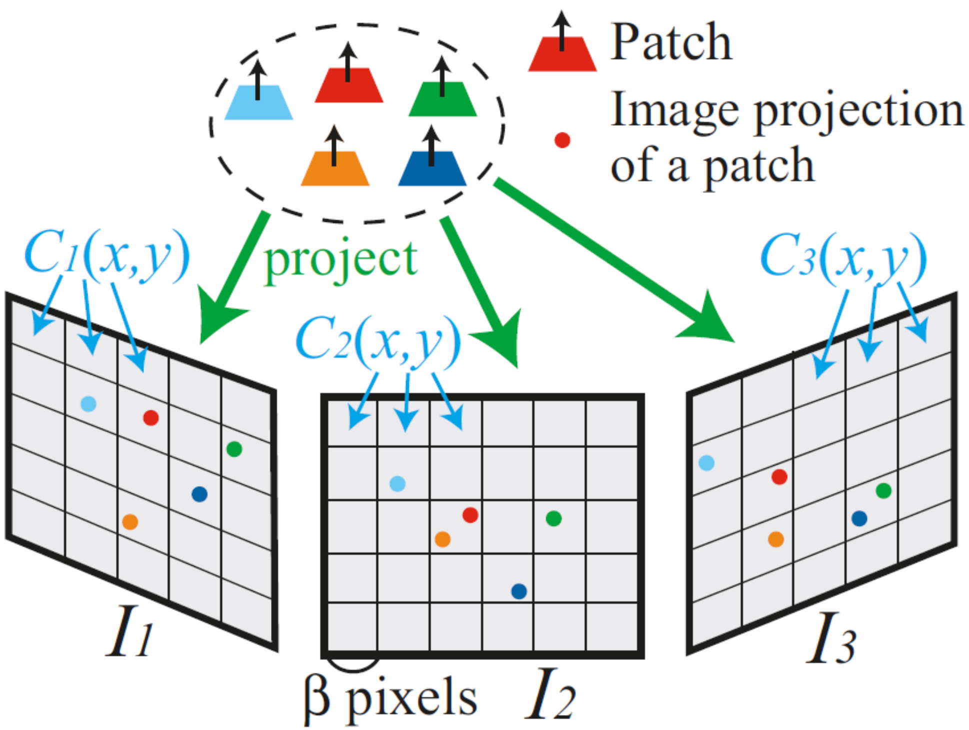
3b) Expand patches to neighbors



Furukawa

Patch-Based Approaches

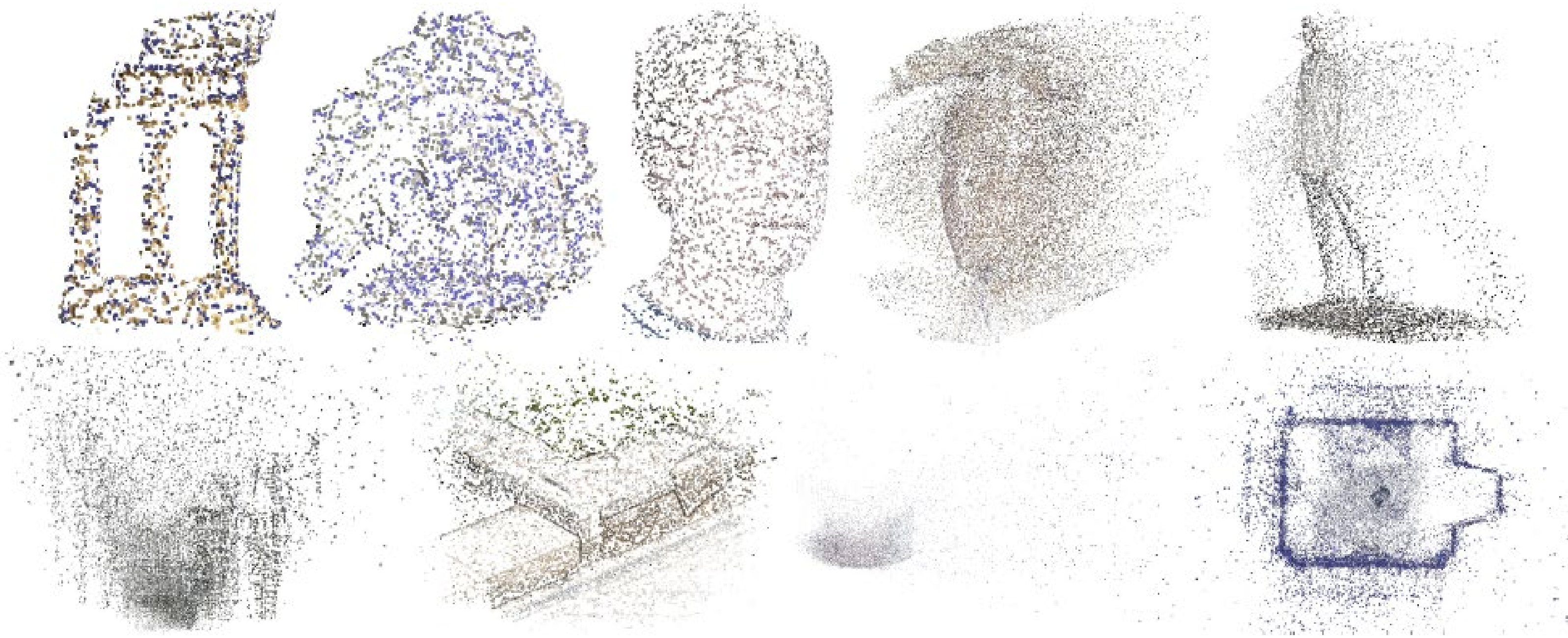
3a) Filter inconsistent patches



Furukawa

Patch-Based Results

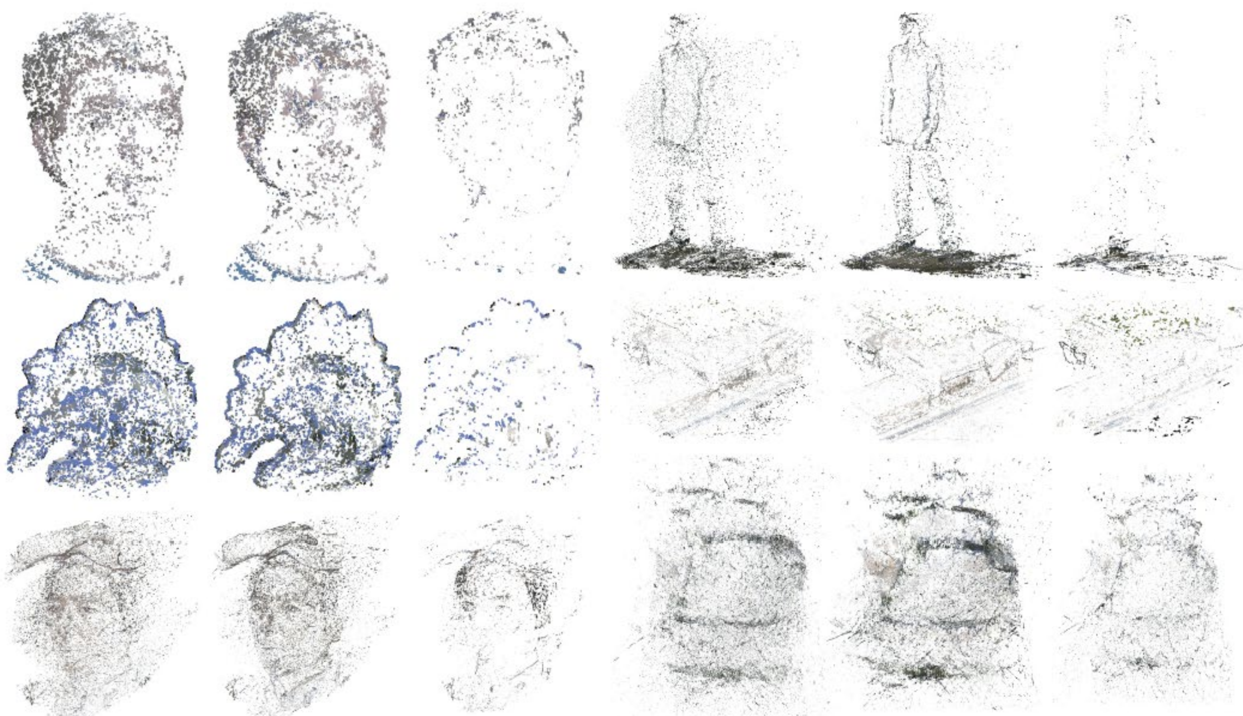
Initial set of seed patches



Furukawa

Patch-Based Results

After filtering inconsistent patches

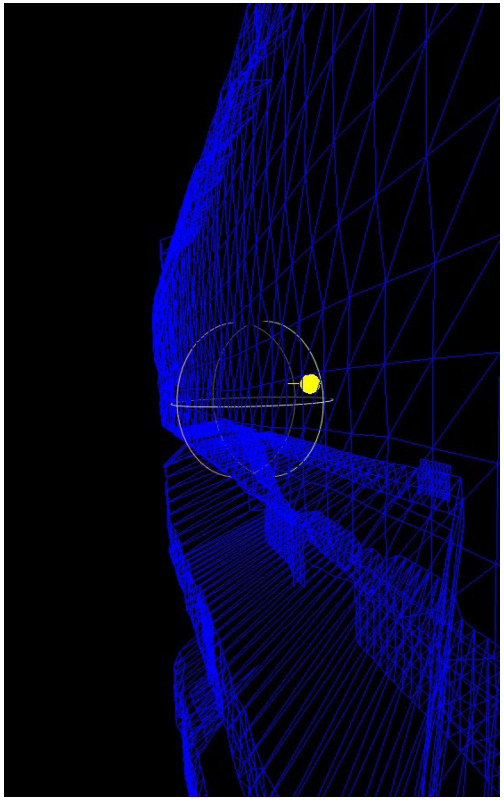


Filter 1 (visibility) | Filter 2 (visibility) | Filter 3 (regularization) | Filter 1 (visibility) | Filter 2 (visibility) | Filter 3 (regularization)

Furukawa

Mesh-Based Methods

- Optimize vertices of a 3D triangle mesh surface to maximize photoconsistency
  - Generate initial mesh (e.g., connecting patches)
  - Move vertices along normal direction to improve photoconsistency (e.g., NCC)



Patch + Mesh Results

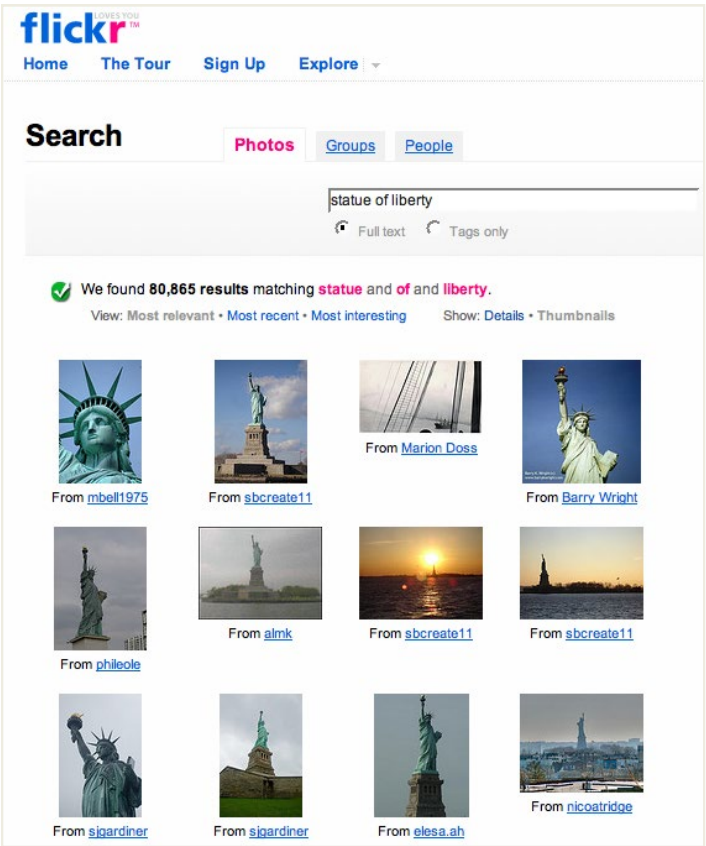
Furukawa

Patch + Mesh Results

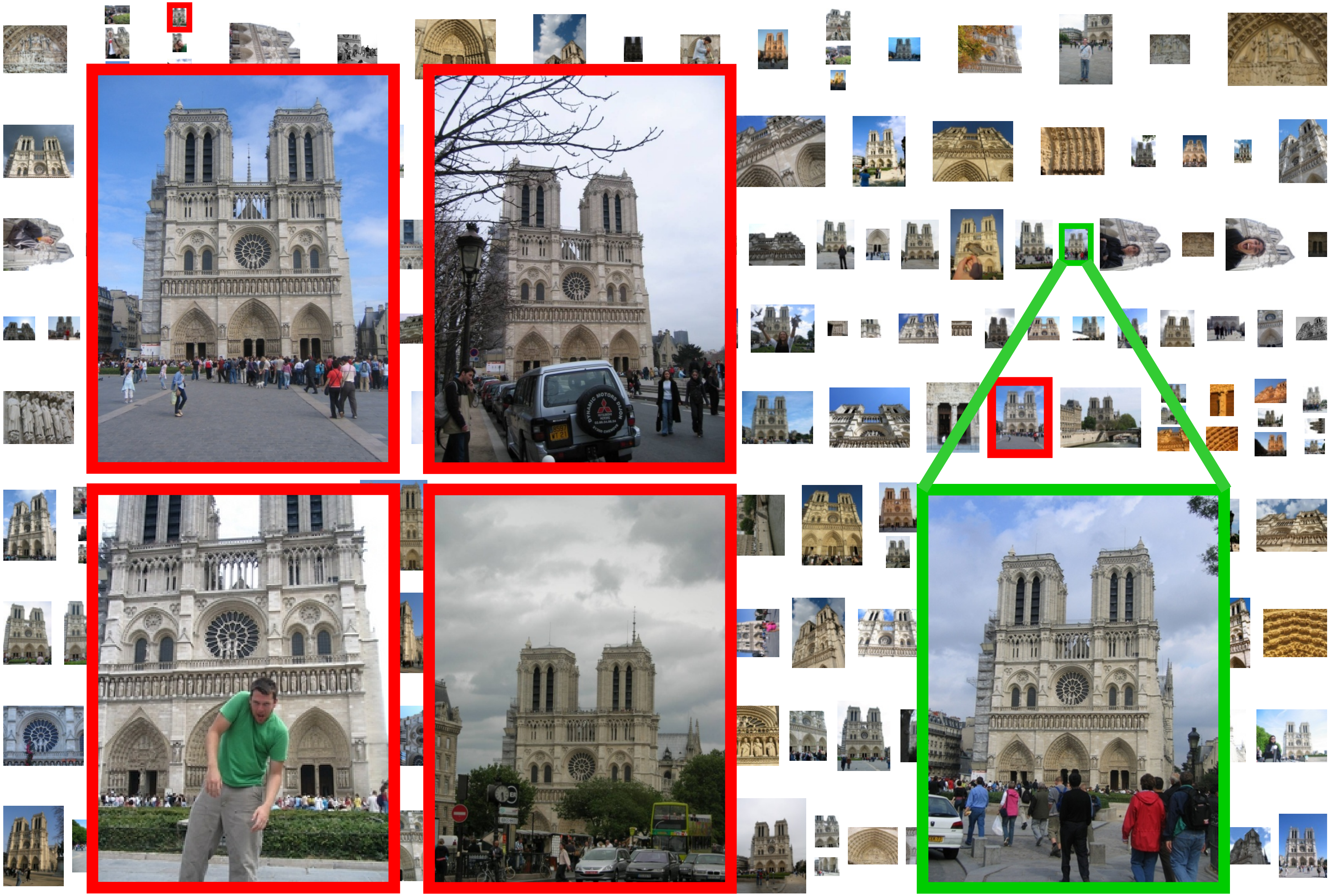
Furukawa

Multi-View Stereo from Internet Collections

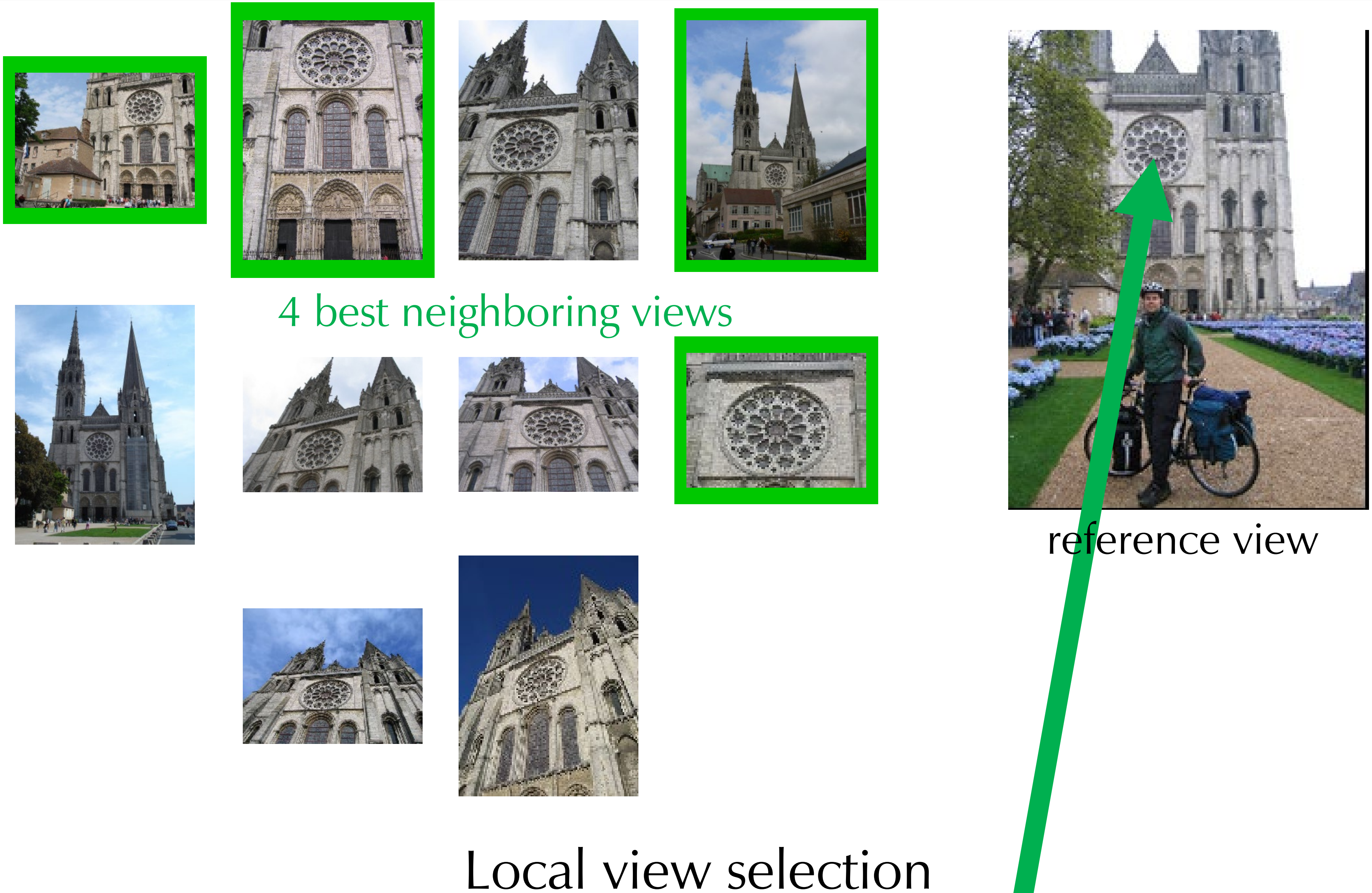
[Goesele, Snavely, Curless, Hoppe, Seitz, ICCV 2007]



Law of Large Image Collections



206 *Flickr* images taken by 92 photographers

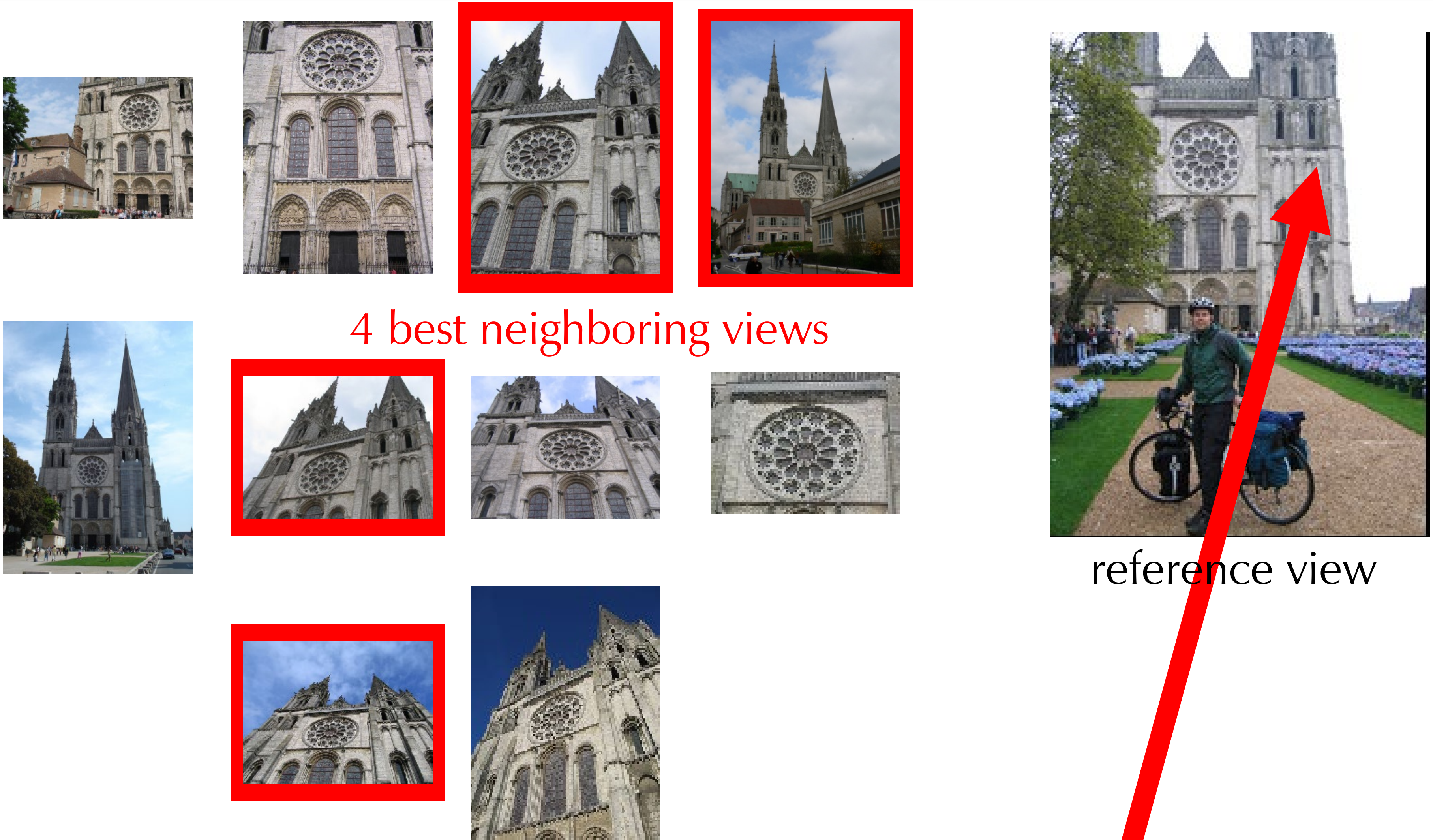


Local view selection

- 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

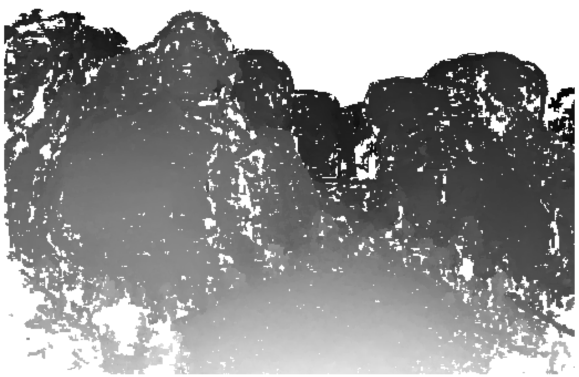
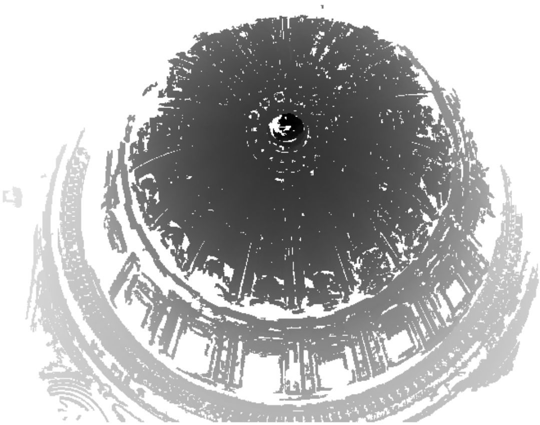
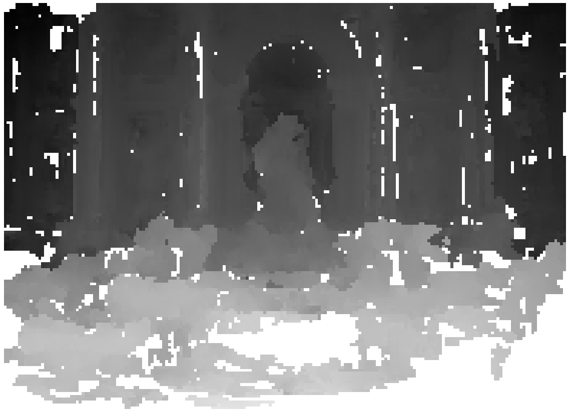


Local view selection

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

Results



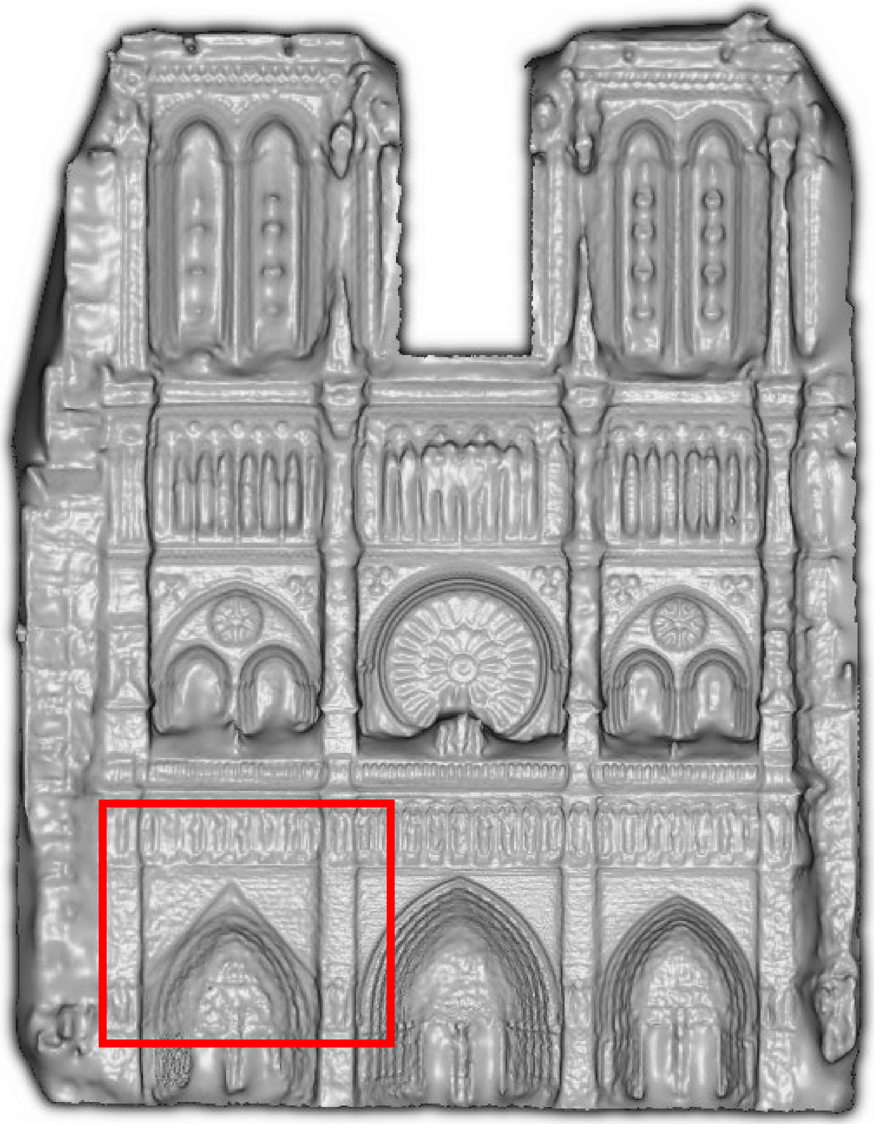
Mt. Rushmore
160 images
60 photographers

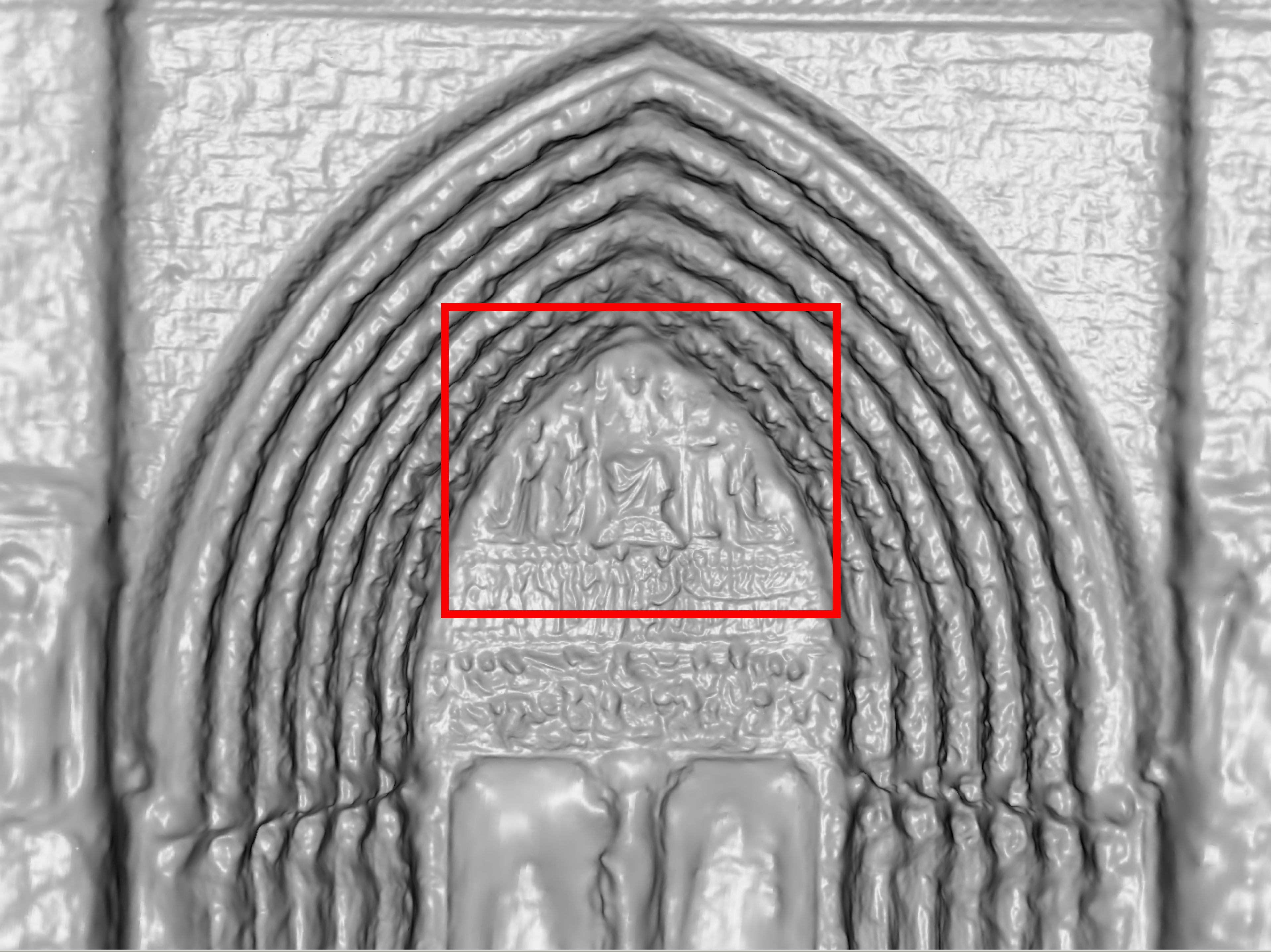
St. Peter
151 images
50 photographers

Trevi Fountain
106 images
51 photographers

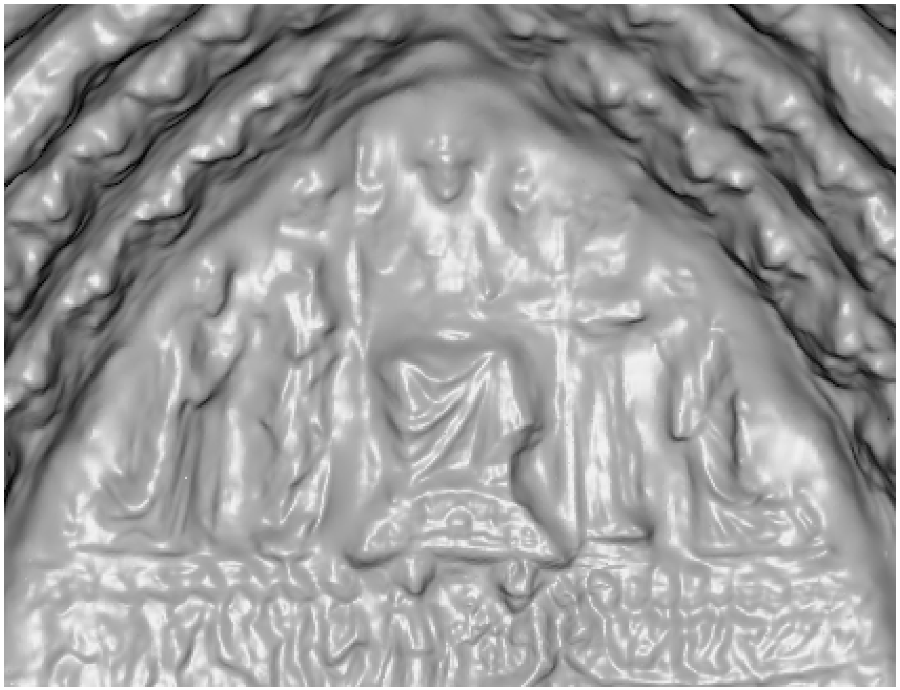
Notre Dame de Paris

653 images taken by 313photographers



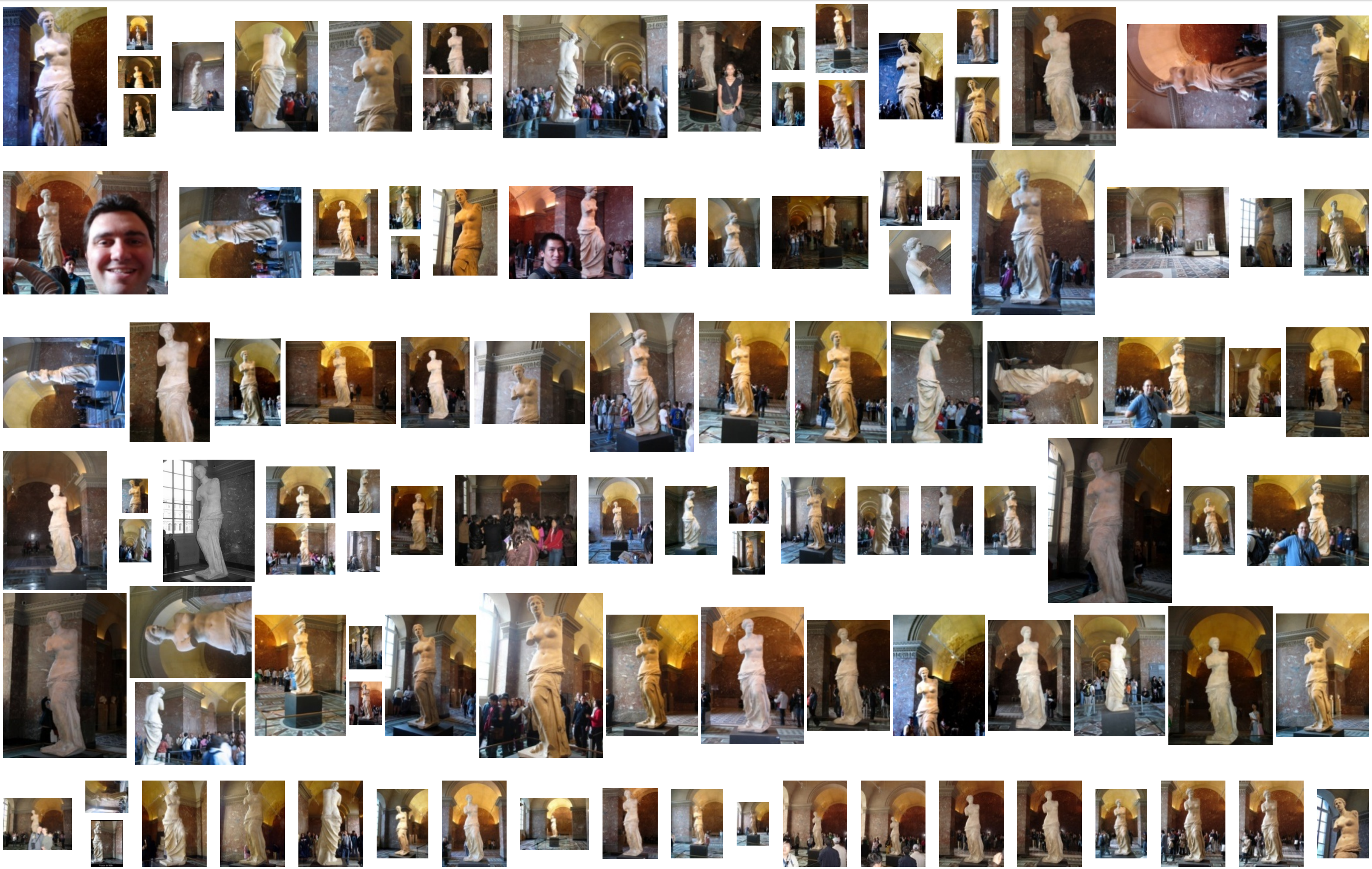


Notre Dame de Paris



Venus de Milo

129 *Flickr* images taken by 98 photographers

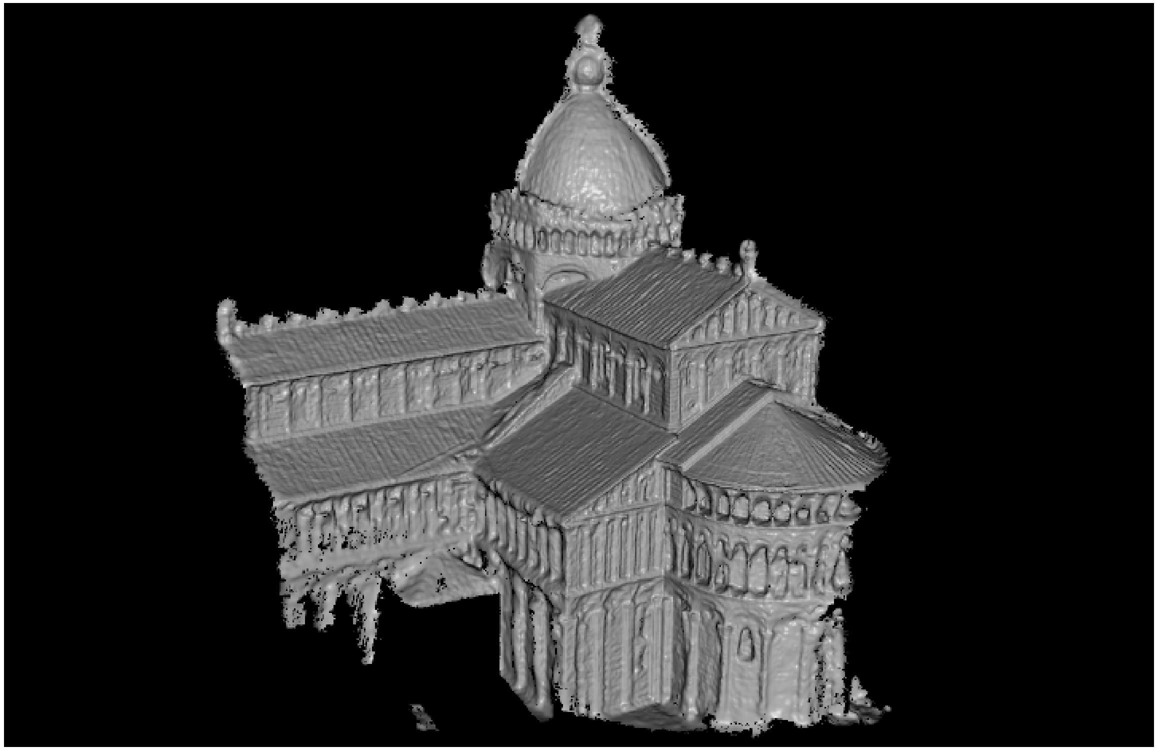


Venus de Milo

Pisa Cathedral

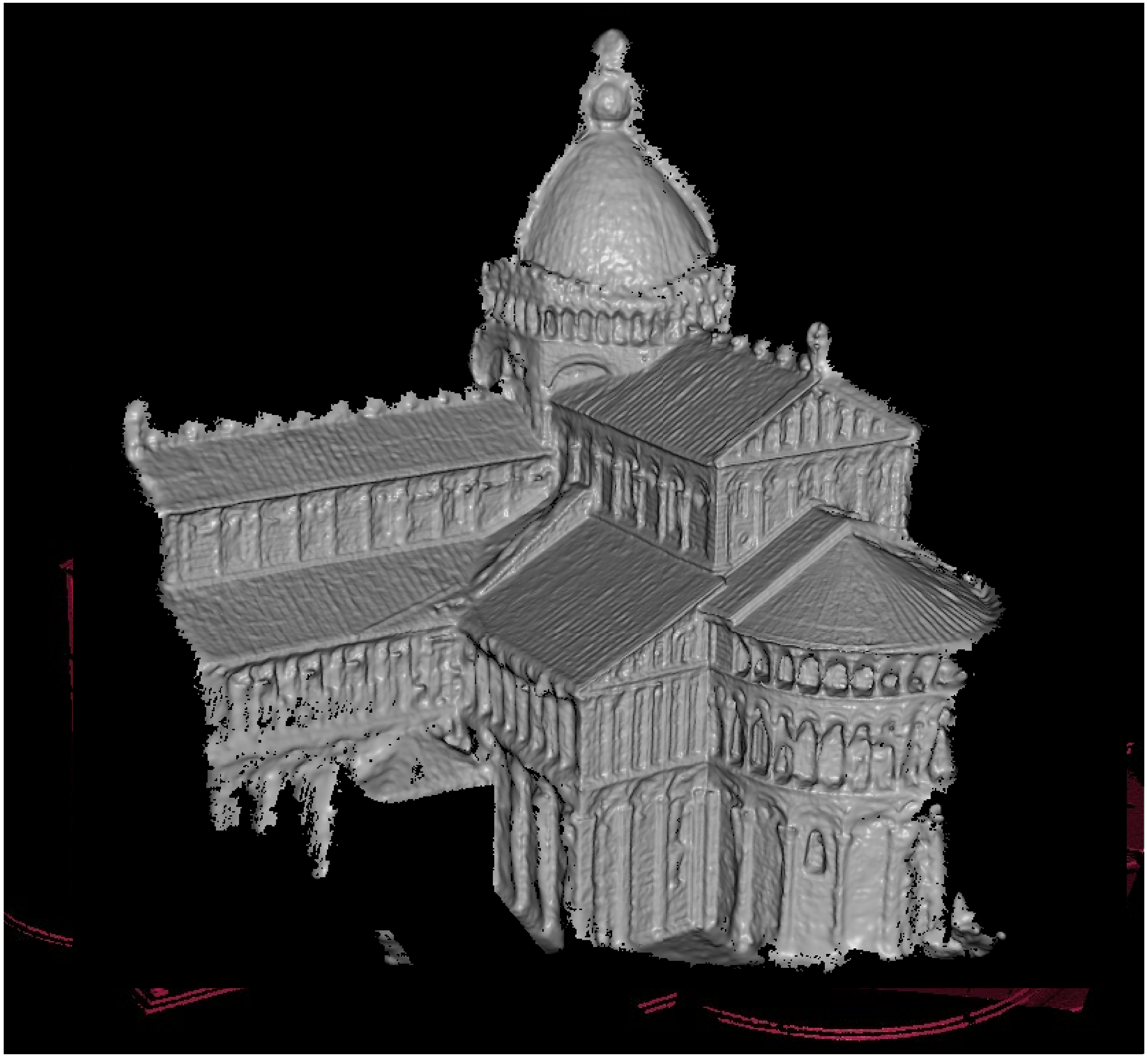
56 *Flickr* images taken by 8 photographers

Pisa Cathedral



Pisa Cathedral

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



Summary

- Image-centric approaches
  - Multibaseline stereo
  - Plane-sweep stereo
- Voxel-centric approaches
  - Silhouette carving
  - Voxel coloring
  - Space carving
- Surface-centric approaches
  - Feature detection + expansion/filtering
  - Mesh refinement