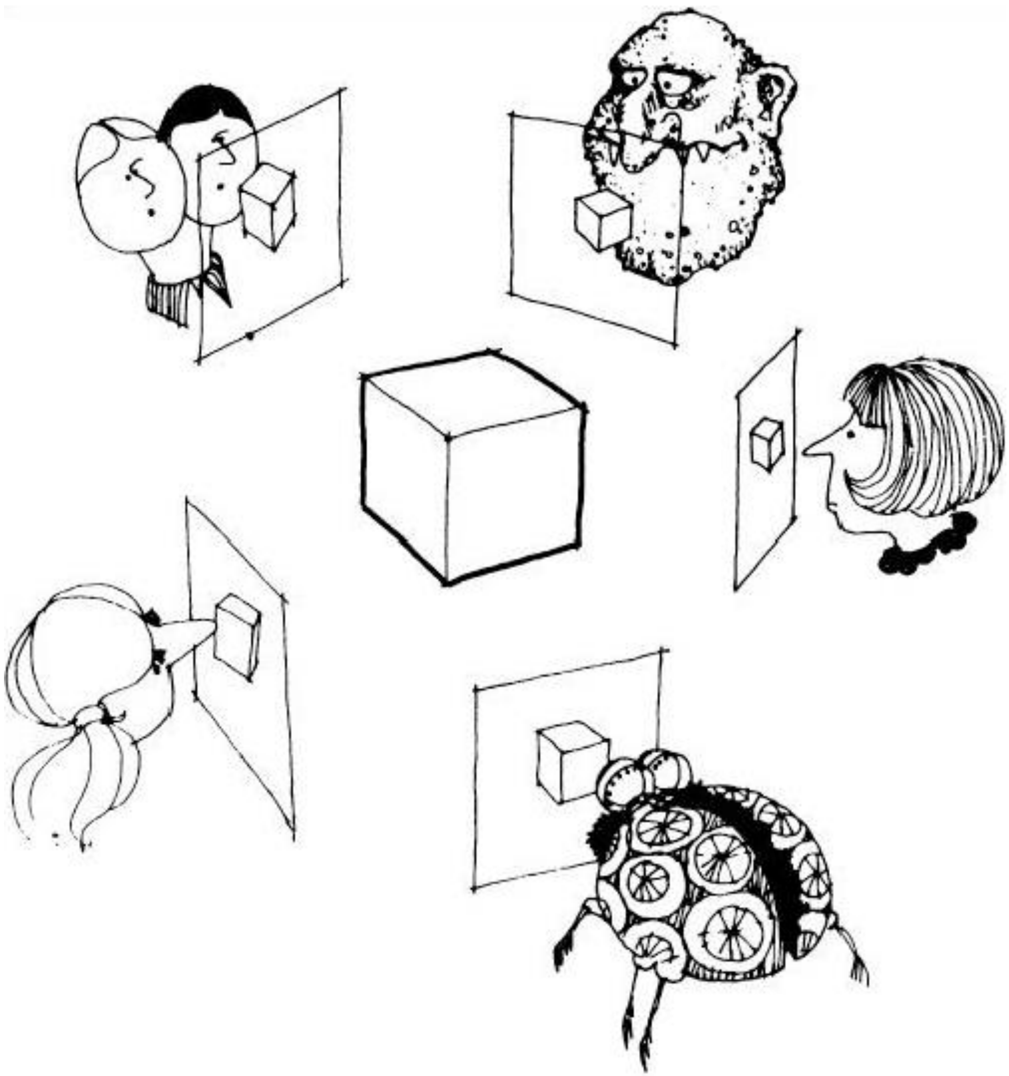
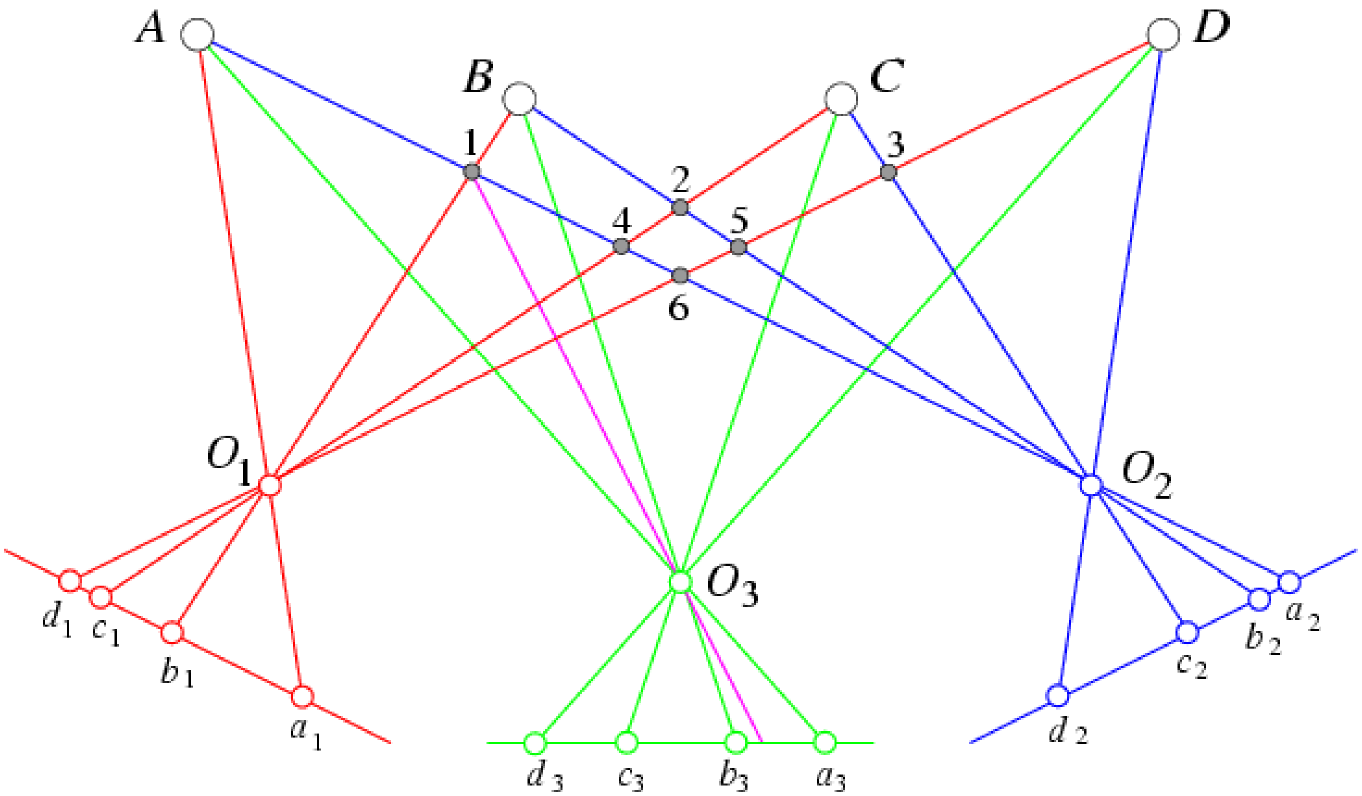
Multi-view stereo



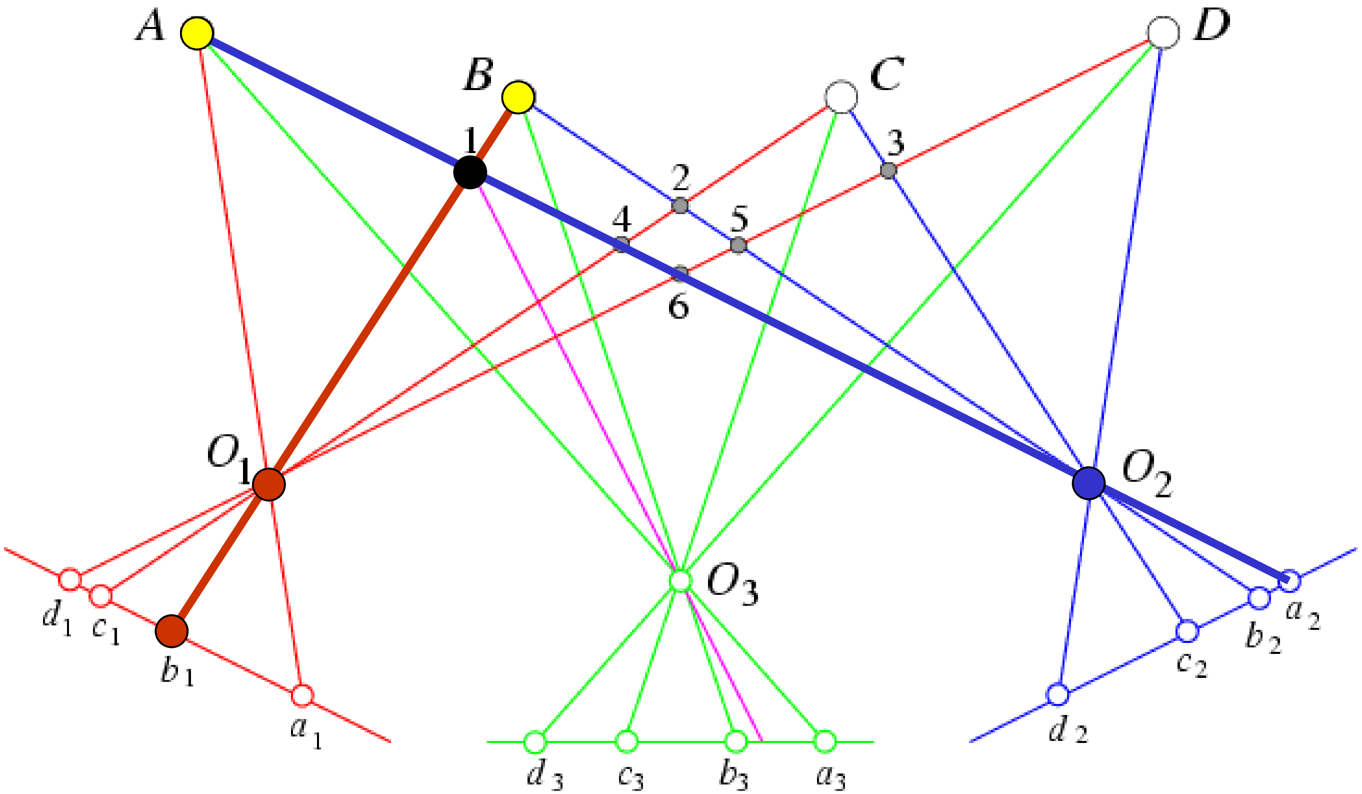
Many slides adapted from S. Seitz

Beyond two-view stereo



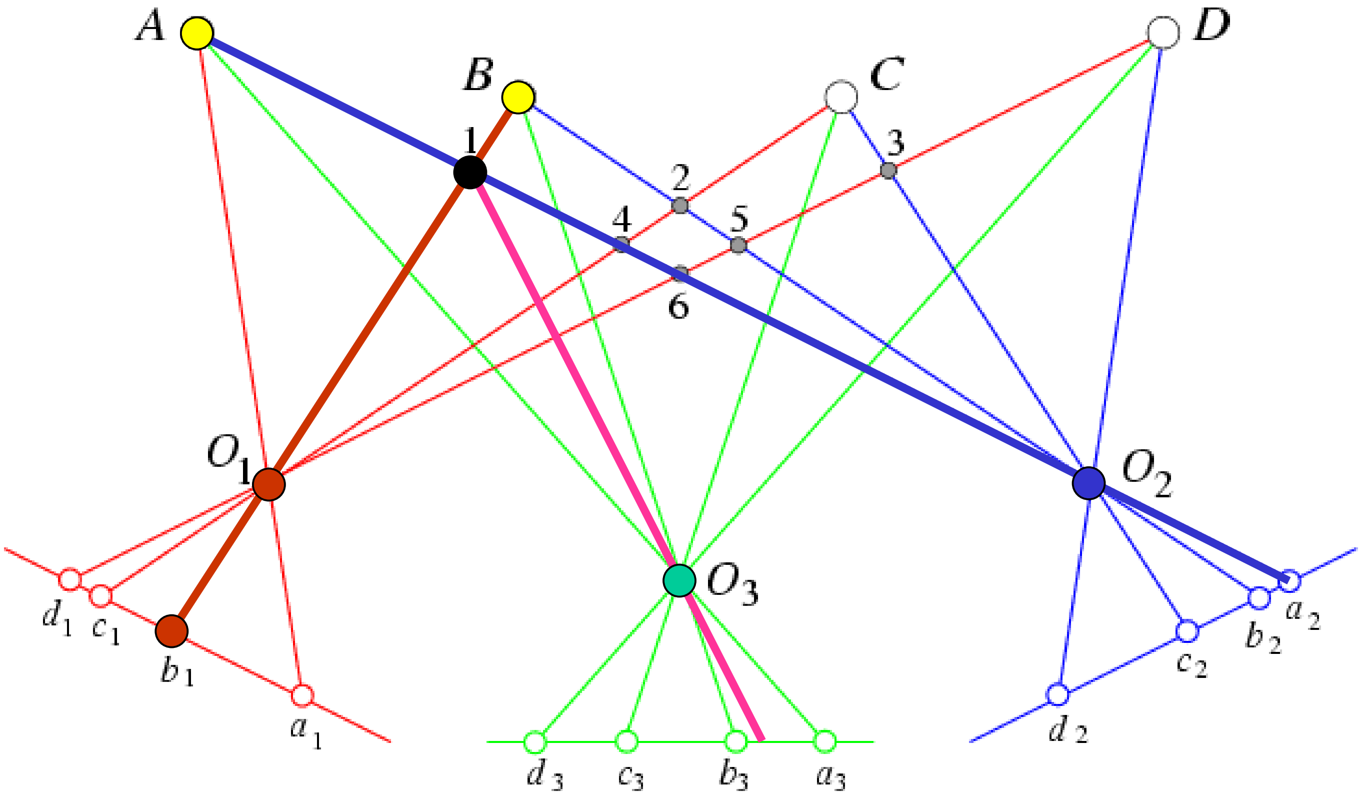
The third eye can be used for verification

Beyond two-view stereo



The third eye can be used for verification

Beyond two-view stereo

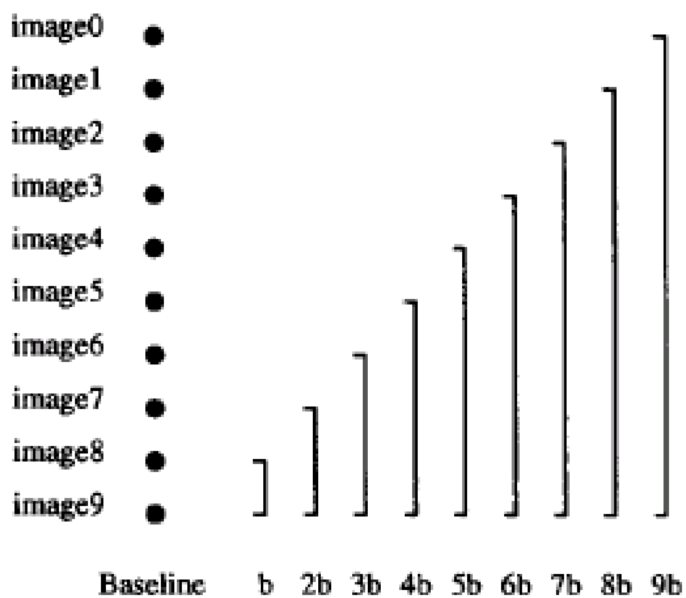


The third eye can be used for verification

Multiple-baseline stereo

- Pick a reference image, and slide the corresponding window along the corresponding epipolar lines of all other images, using inverse depth relative to the first image as the search parameter

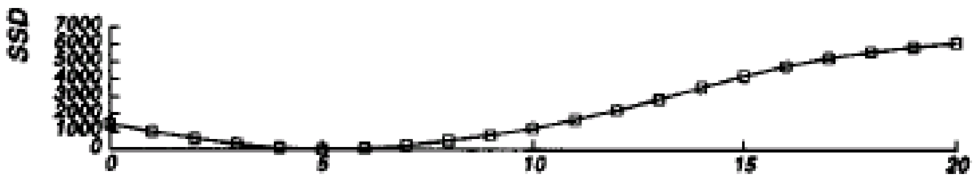
Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.



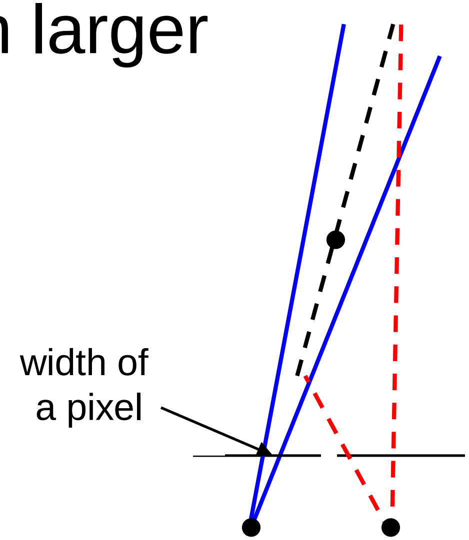
M. Okutomi and T. Kanade, [“A Multiple-Baseline Stereo System,”](#) IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(4):353-363 (1993).

Multiple-baseline stereo

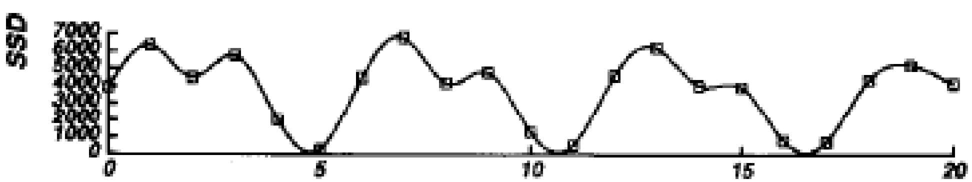
- For larger baselines, must search larger area in second image



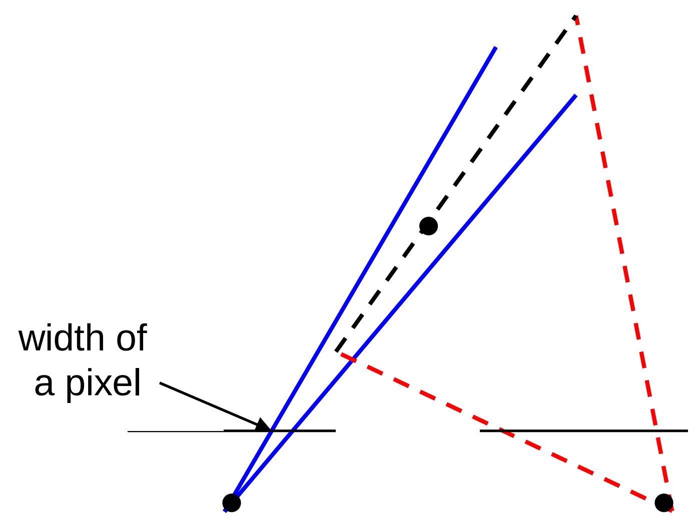
1/z



pixel matching score



1/z



Multiple-baseline stereo

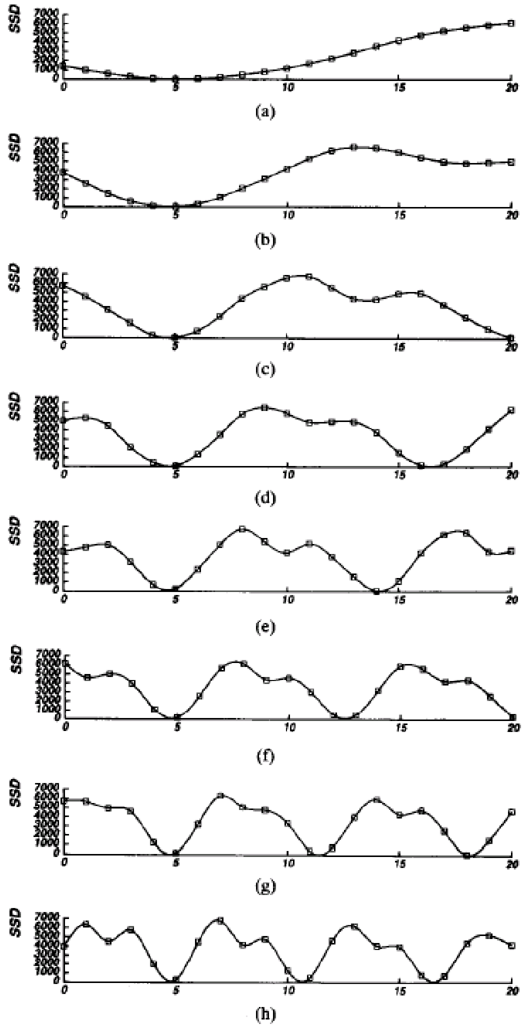


Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

Use the sum of correlation scores to rank matches

Multiple-baseline stereo

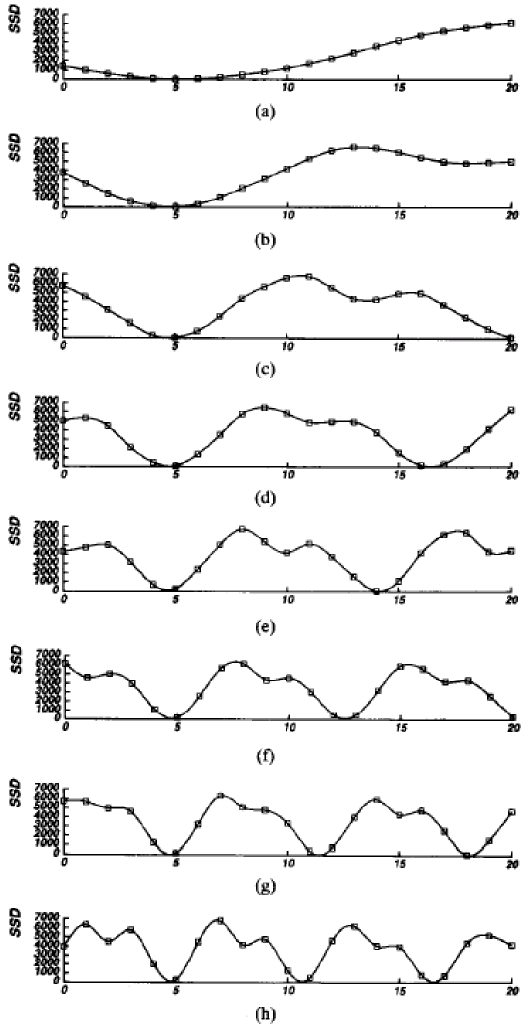


Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

Use the sum of correlation scores to rank matches

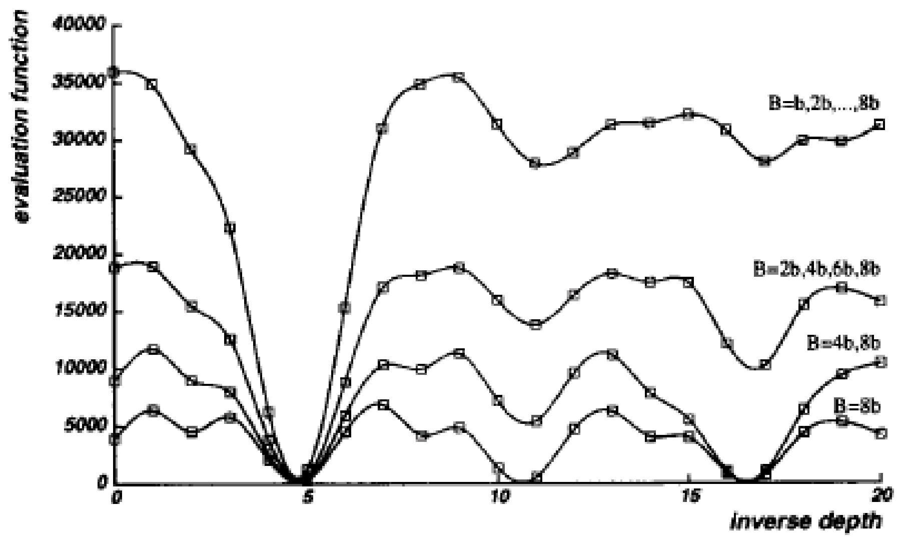


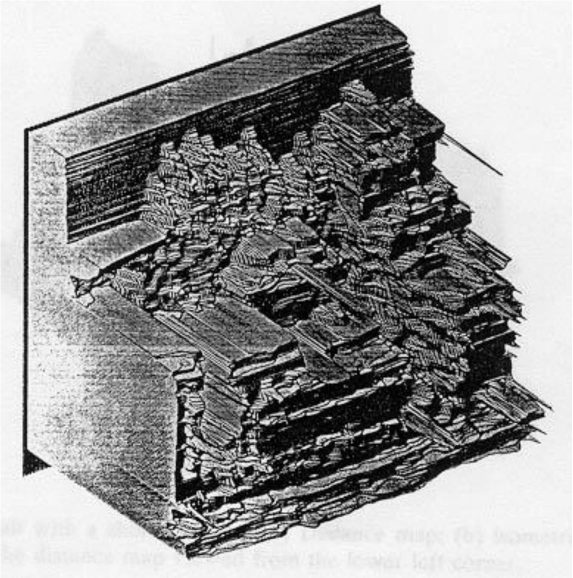
Fig. 7. Combining multiple baseline stereo pairs.

Multiple-baseline stereo results

I1

I2

I10



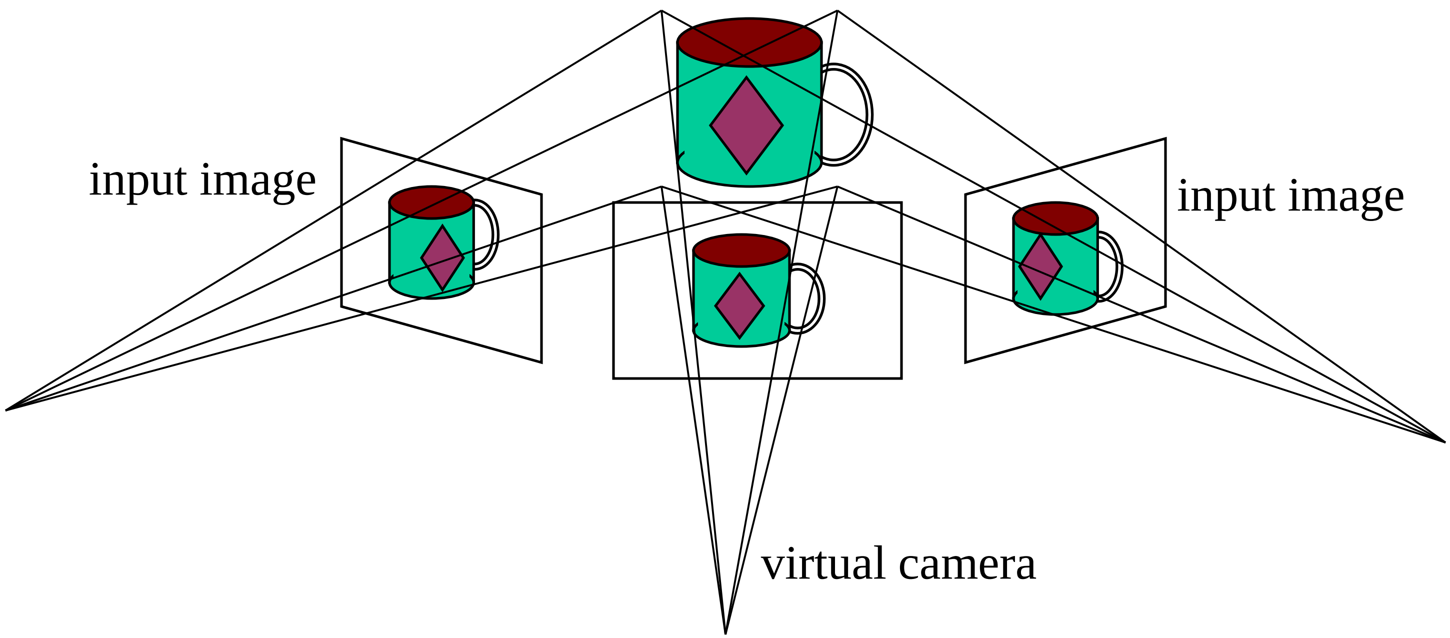
M. Okutomi and T. Kanade, "A Multiple-Baseline Stereo System," IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(4):353-363 (1993).

Summary: Multiple-baseline stereo

- Basic Approach
  - Choose a reference view
  - Plot SSD vs. inverse depth instead of disparity
  - Replace two-view SSD with sum of SSD over all baselines
- Limitations
  - Must choose a reference view
  - Occlusions become an issue for large baseline
- Possible solution: use a *virtual view*

Plane Sweep Stereo

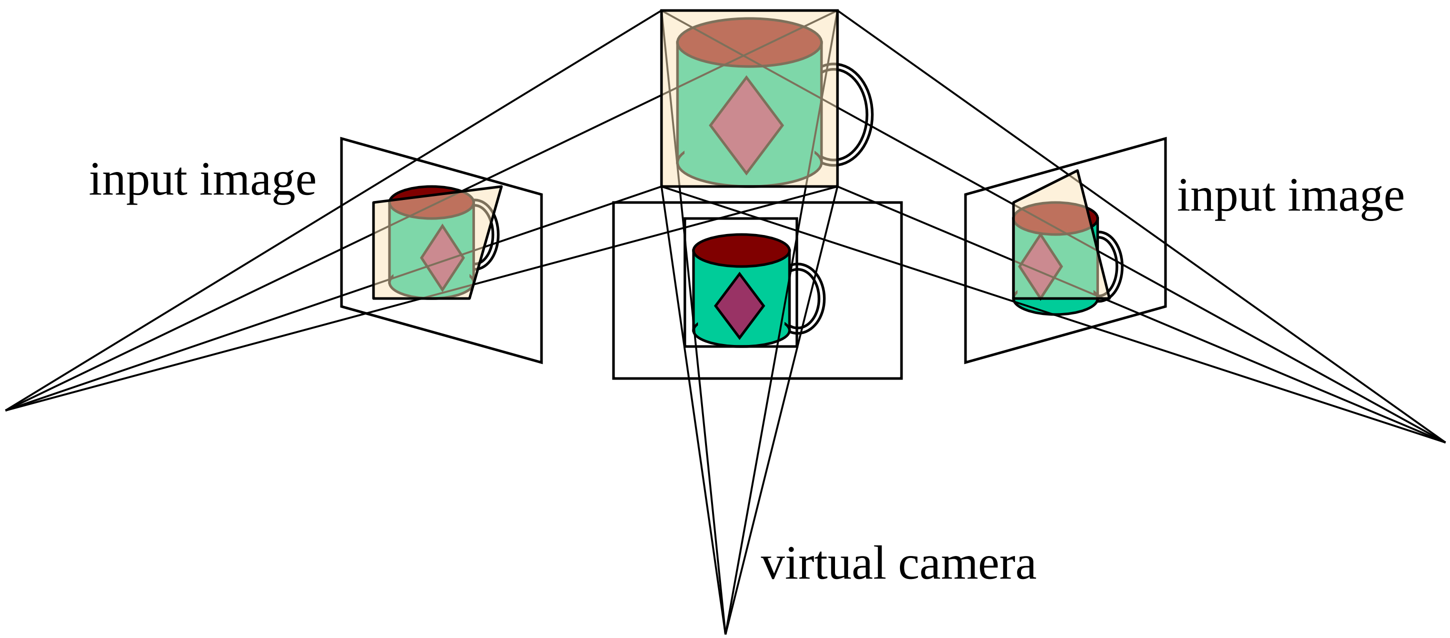
- Choose a virtual view
- Sweep family of planes at different depths with respect to the virtual camera



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

Plane Sweep Stereo

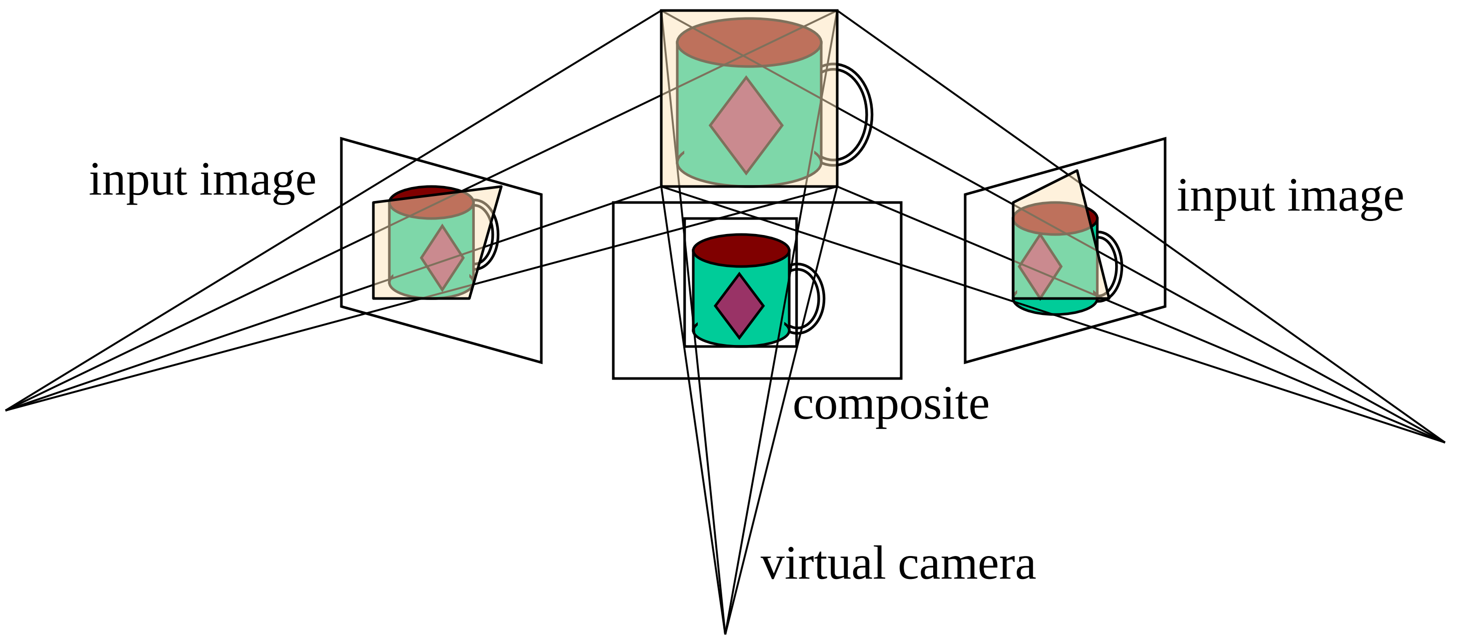
- Choose a virtual view
- Sweep family of planes at different depths with respect to the virtual camera



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

Plane Sweep Stereo

- Choose a virtual view
- Sweep family of planes at different depths with respect to the virtual camera

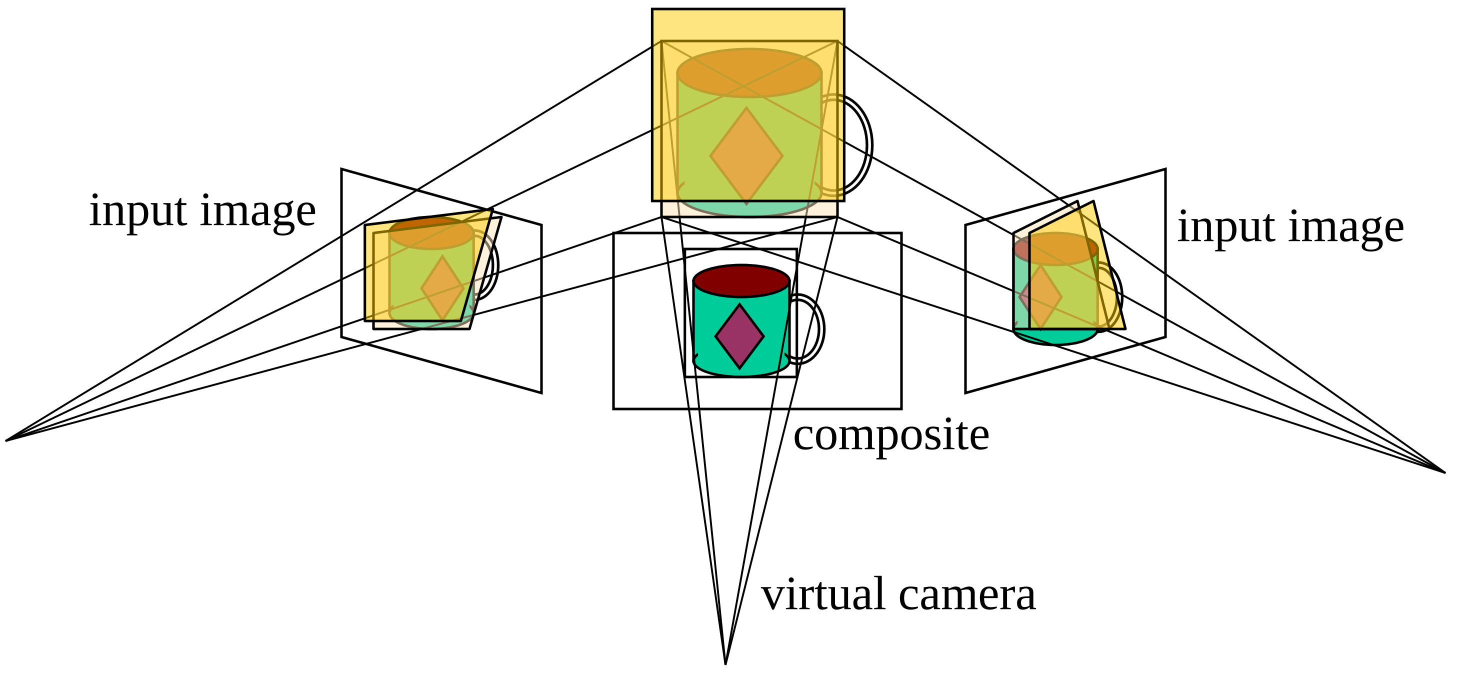


each plane defines an image ⇒ composite homography

R. Collins. [A space-sweep approach to true multi-image matching.]() CVPR 1996.

Plane Sweep Stereo

- Choose a virtual view
- Sweep family of planes at different depths with respect to the virtual camera

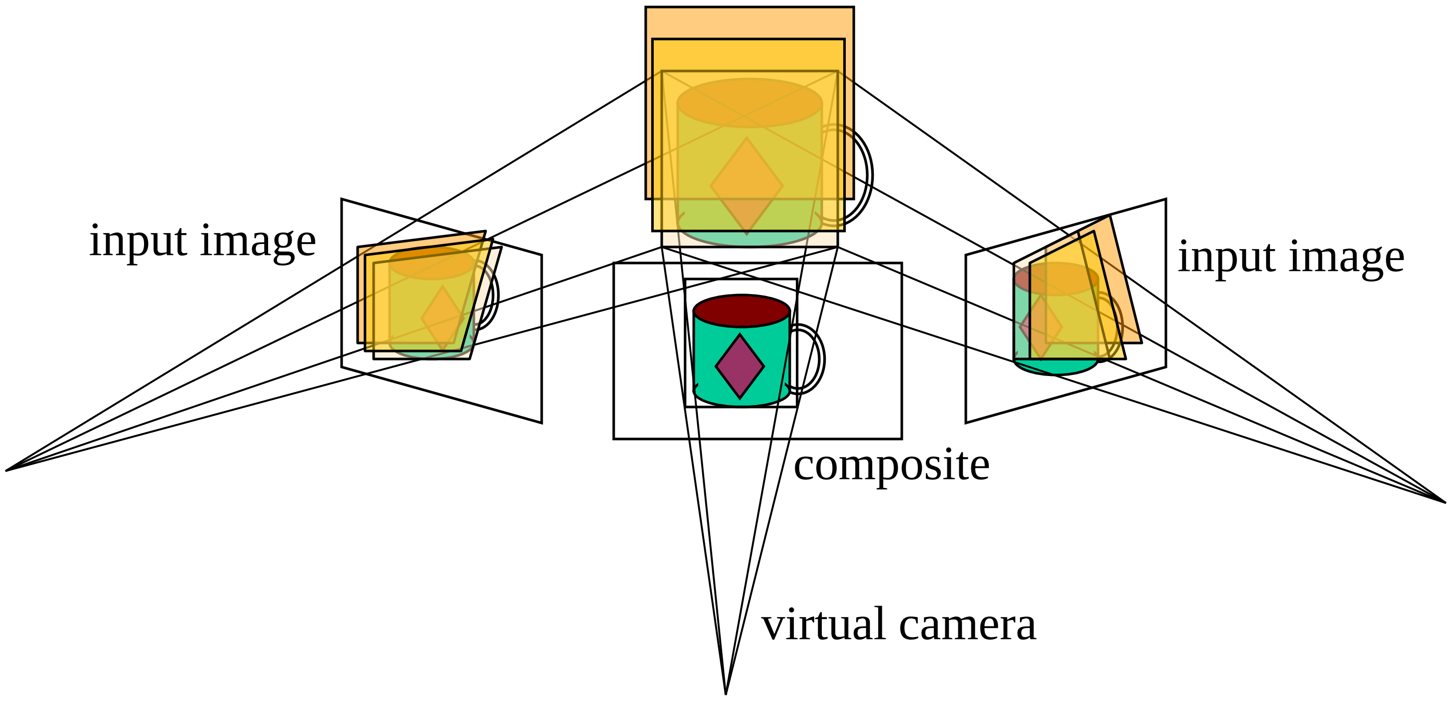


each plane defines an image $\Rightarrow$ composite homography

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

Plane Sweep Stereo

- Choose a virtual view
- Sweep family of planes at different depths with respect to the virtual camera



each plane defines an image ⇒ composite homography

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

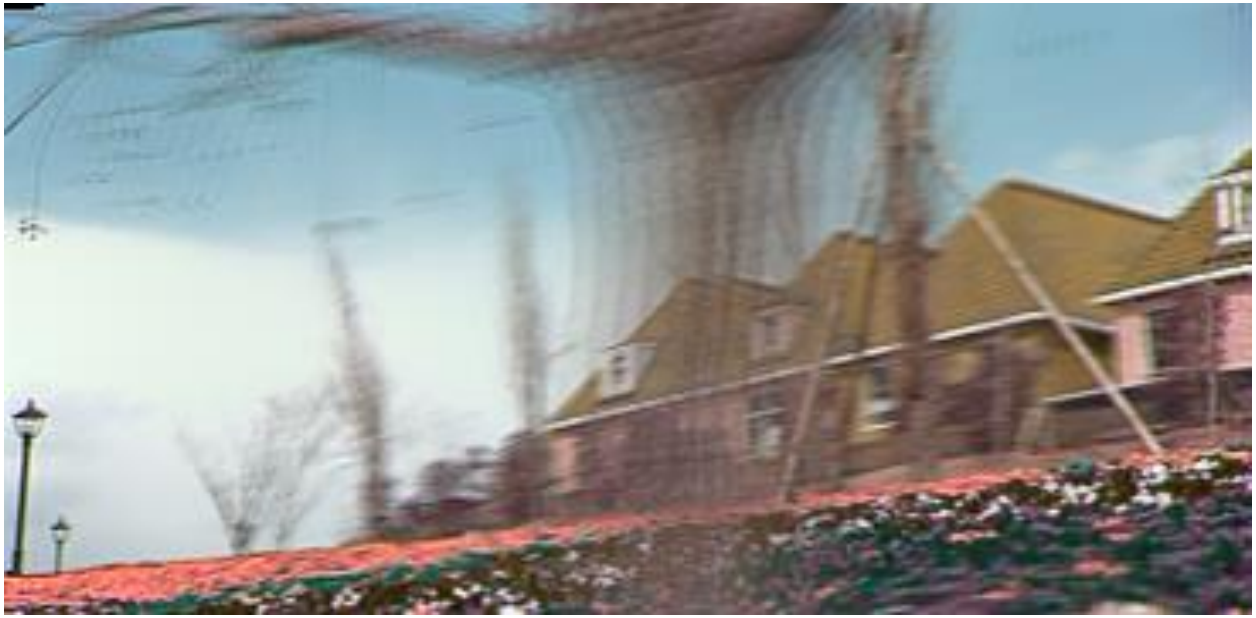
Plane Sweep Stereo

- For each depth plane
  - For each pixel in the composite image, compute the variance

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

Plane Sweep Stereo

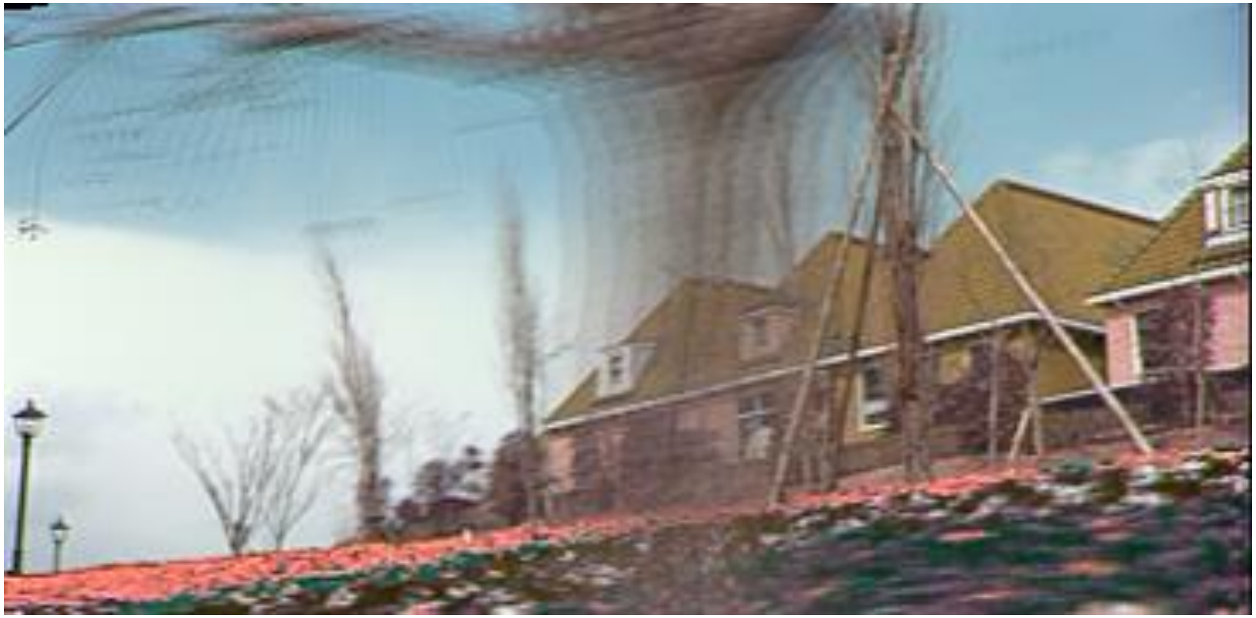
- For each depth plane
  - For each pixel in the composite image, compute the variance



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

Plane Sweep Stereo

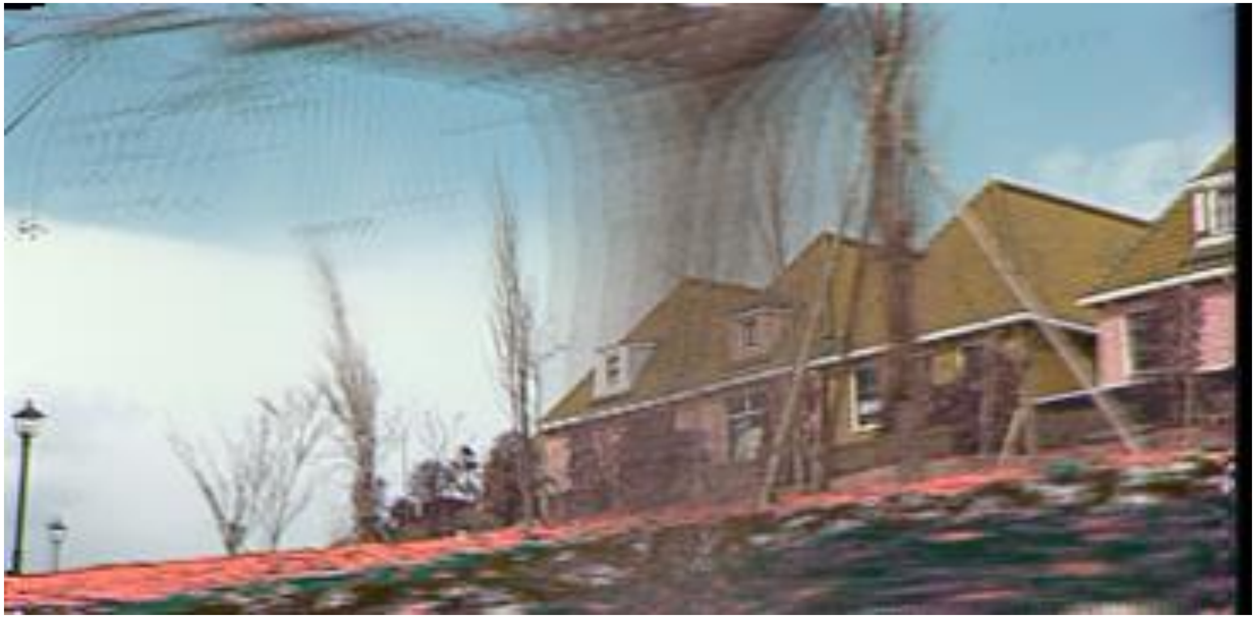
- For each depth plane
  - For each pixel in the composite image, compute the variance



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

Plane Sweep Stereo

- For each depth plane
  - For each pixel in the composite image, compute the variance



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

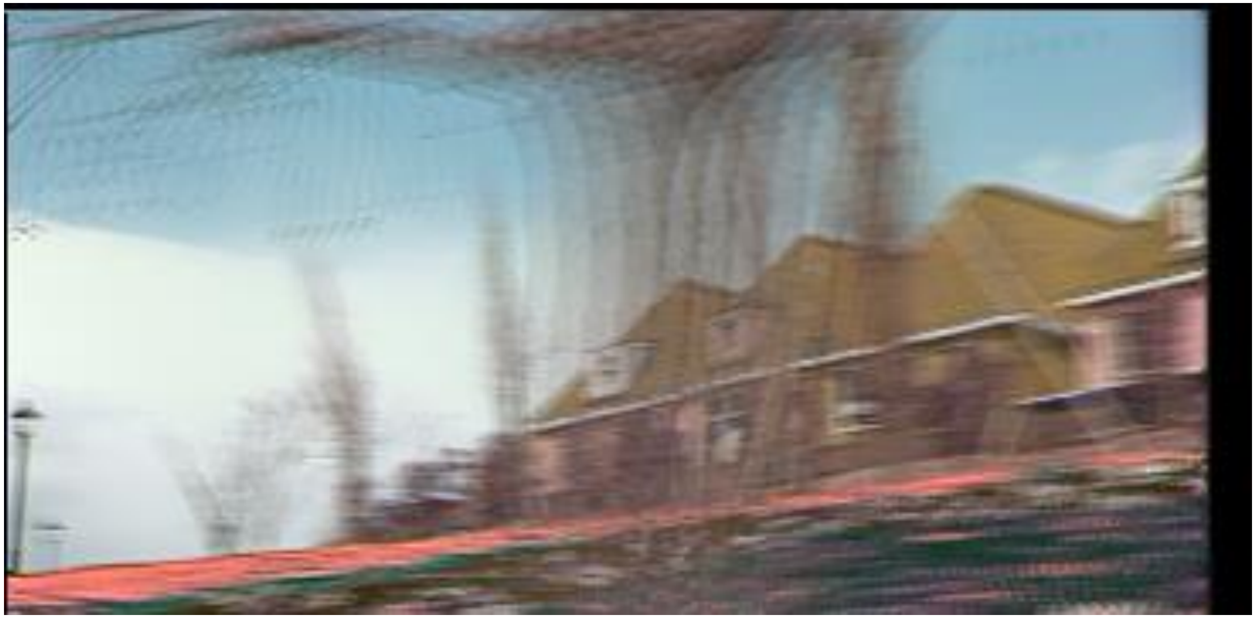
Plane Sweep Stereo

- For each depth plane
  - For each pixel in the composite image, compute the variance

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

Plane Sweep Stereo

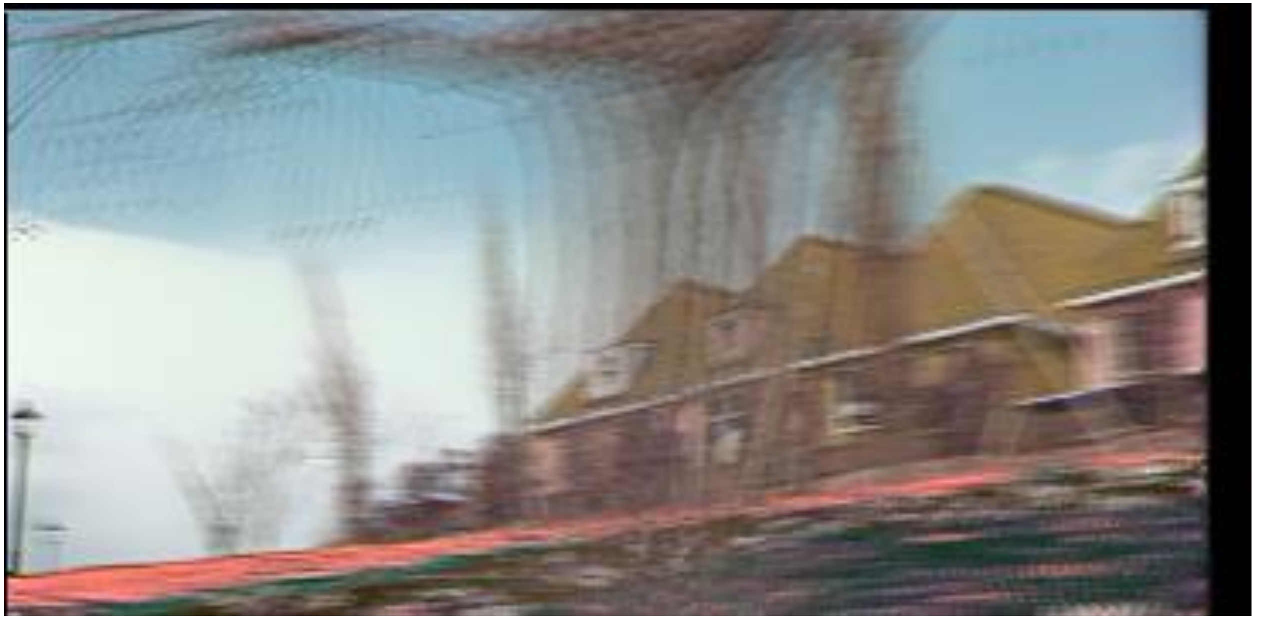
- For each depth plane
  - For each pixel in the composite image, compute the variance



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

Plane Sweep Stereo

- For each depth plane
  - For each pixel in the composite image, compute the variance



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

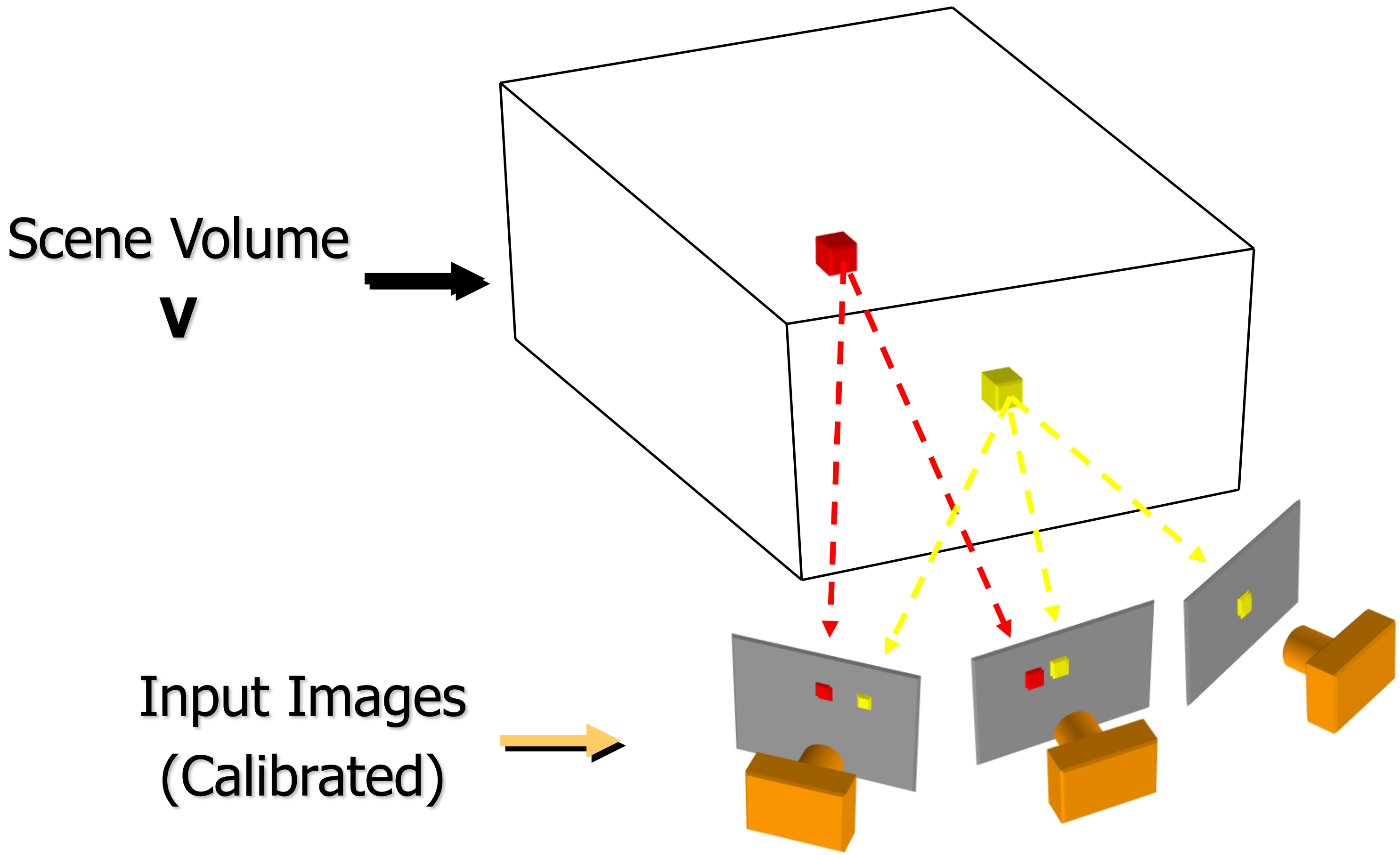
Can be accelerated using graphics hardware

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

Volumetric stereo

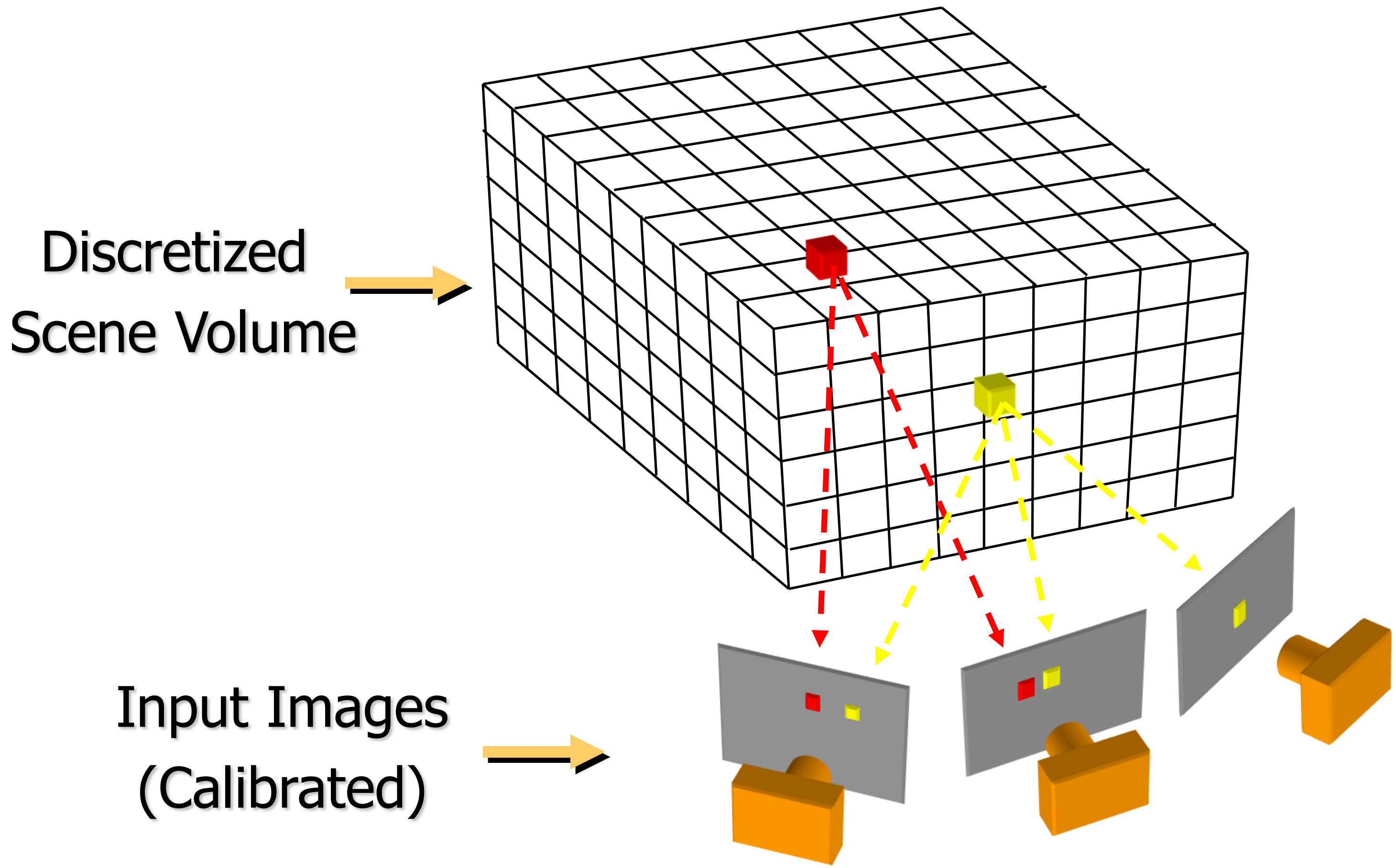
- In plane sweep stereo, the sampling of the scene still depends on the reference view
- We can use a voxel volume to get a view-independent representation

Volumetric stereo



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

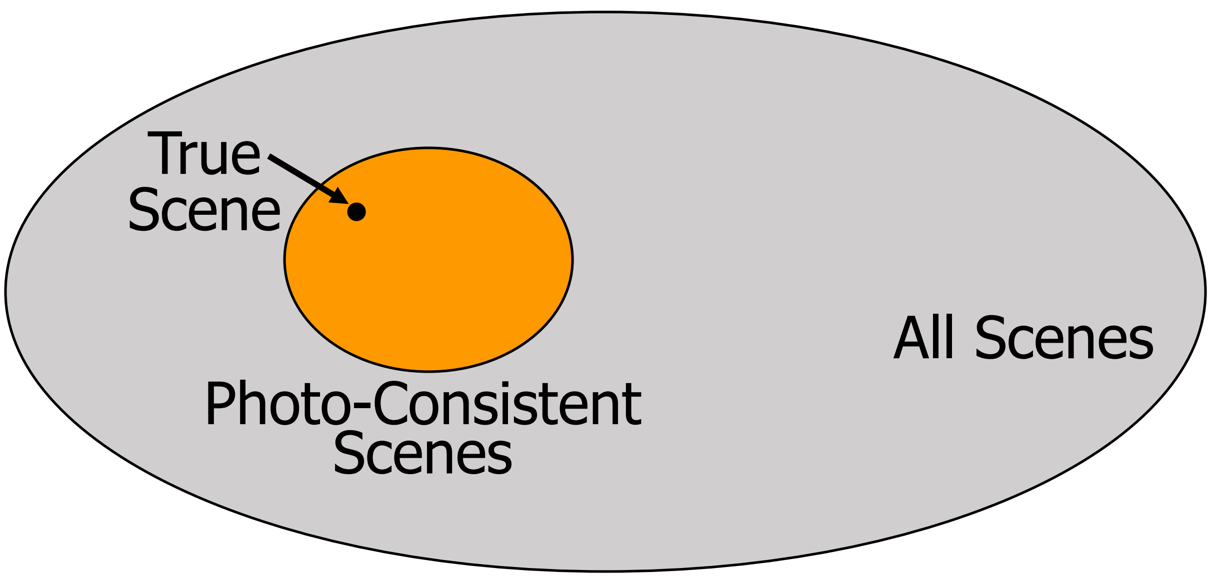
Discrete formulation: Voxel Coloring



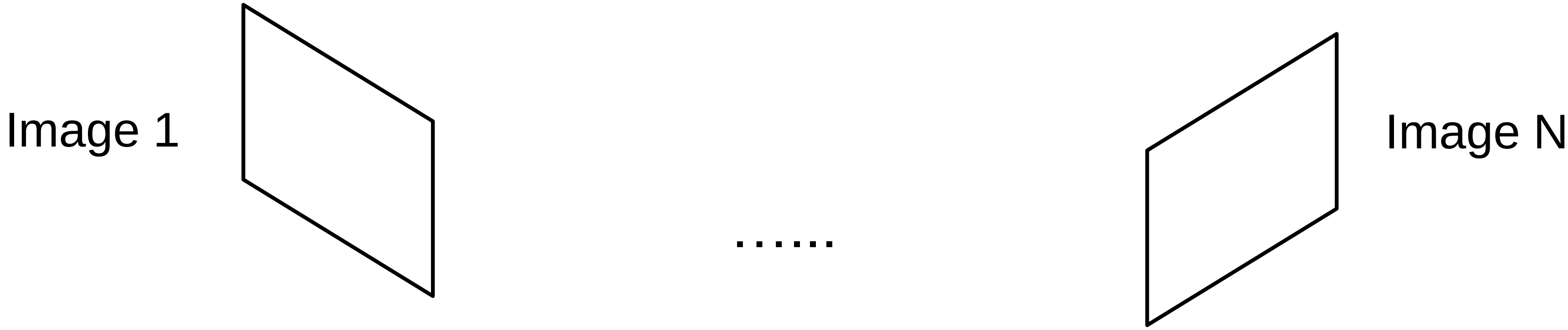
Goal: **Assign RGB values to voxels in V**
photo-consistent **with images**

Photo-consistency

- A *photo-consistent scene* is a scene that exactly reproduces your input images from the same camera viewpoints
- You can't use your input cameras and images to tell the difference between a photo-consistent scene and the true scene



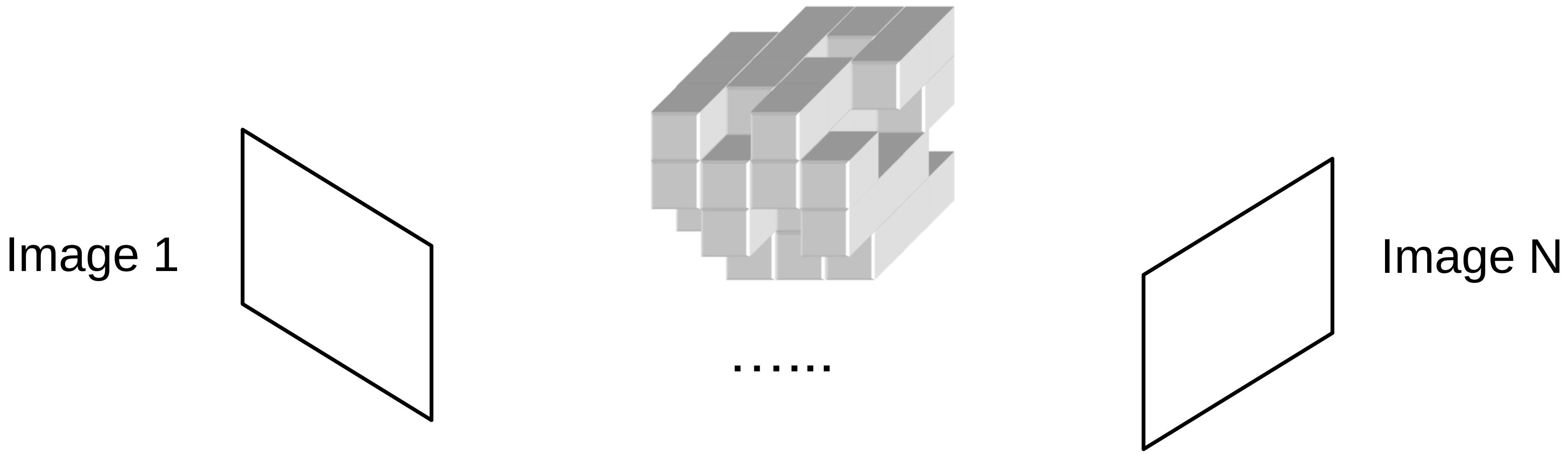
Space Carving



Space Carving Algorithm

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

Space Carving

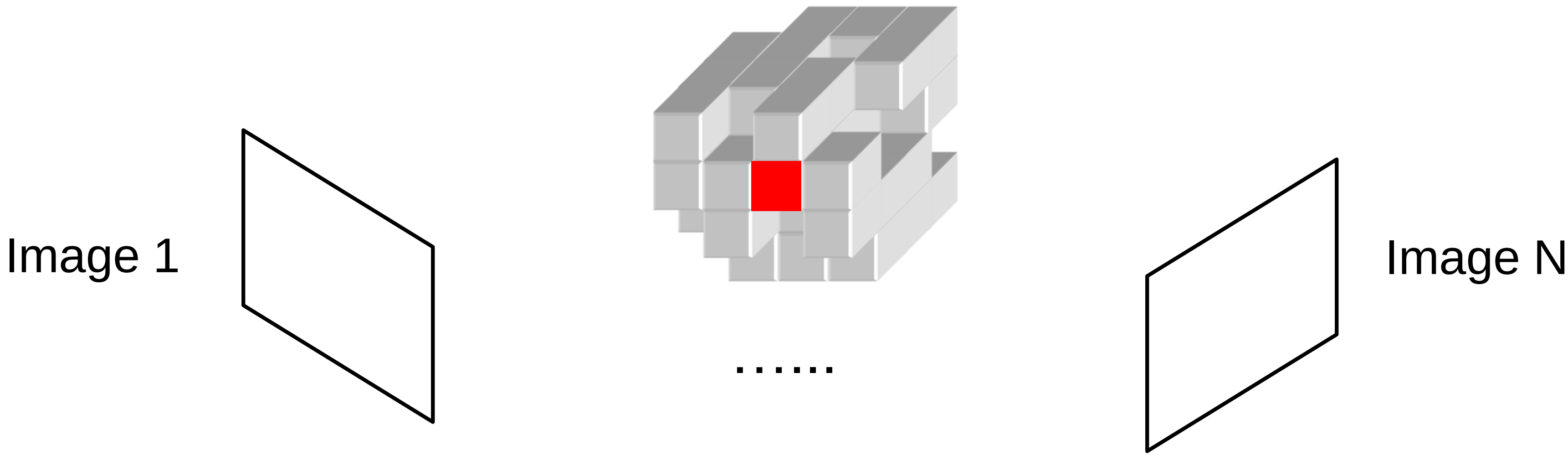


Space Carving Algorithm

- Initialize to a volume V containing the true scene

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

Space Carving

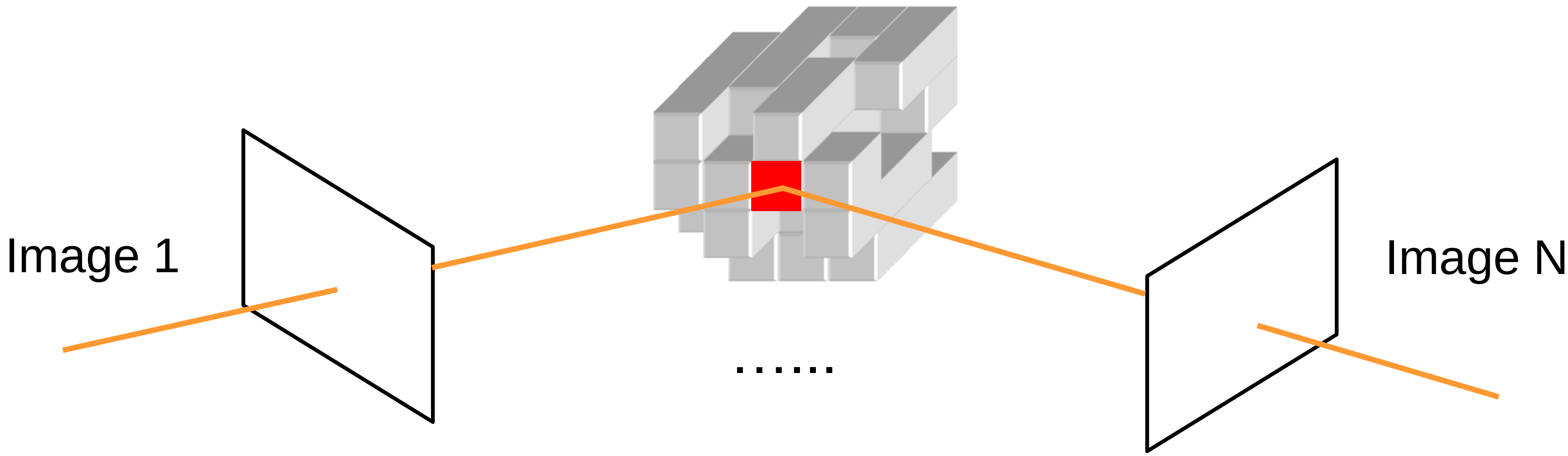


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

Space Carving

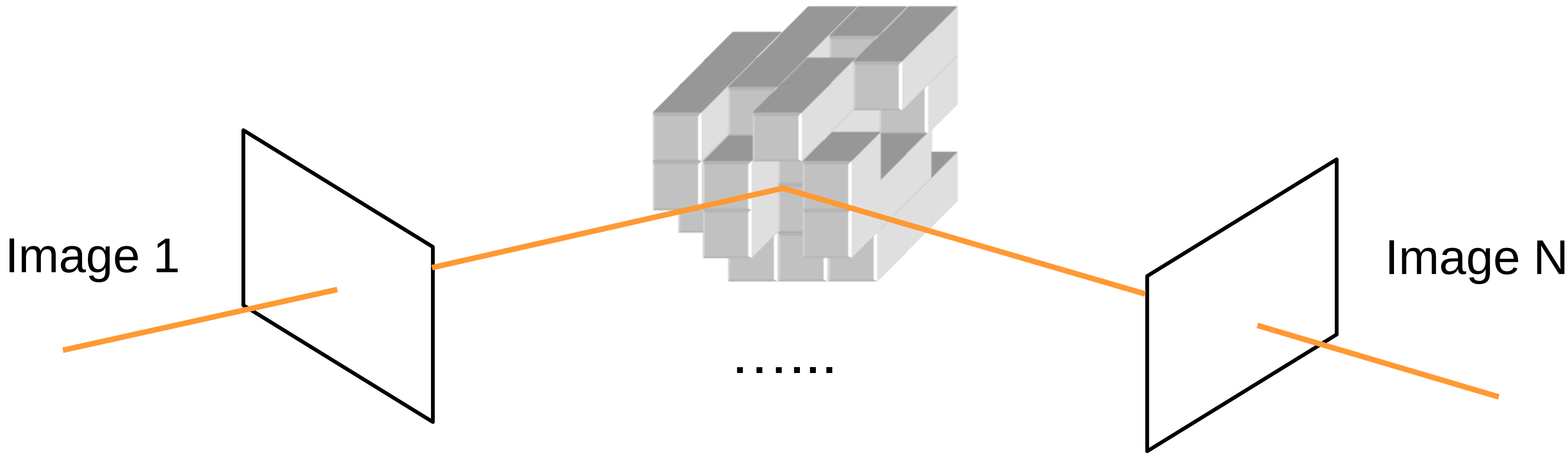


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface
- Project to visible input images

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

Space Carving

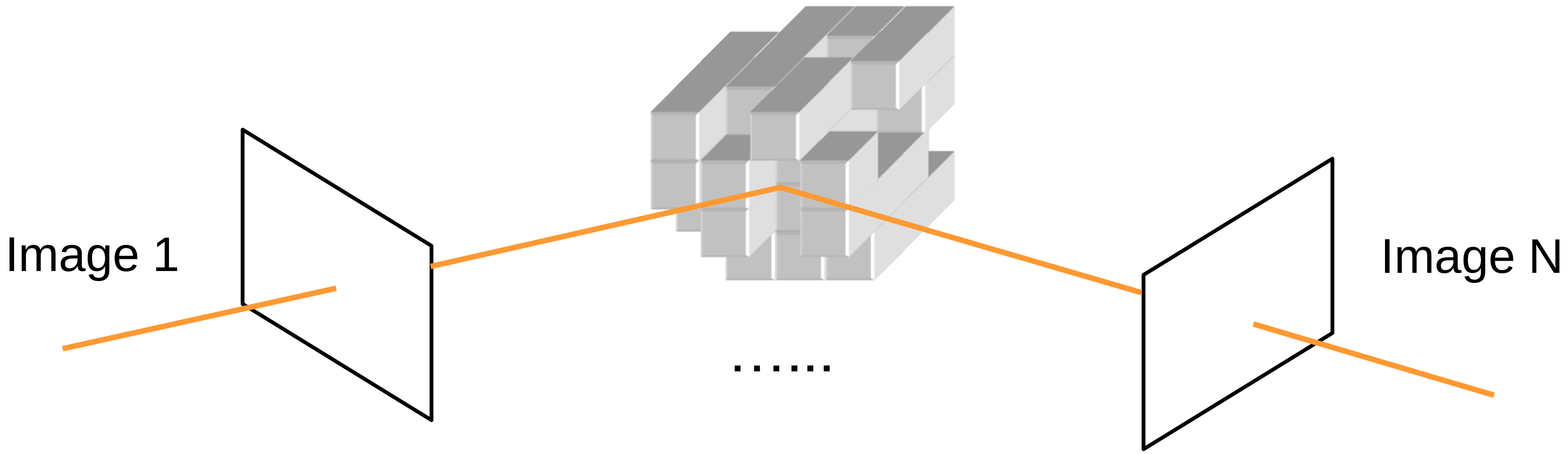


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface
- Project to visible input images
- Carve if not photo-consistent

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

Space Carving

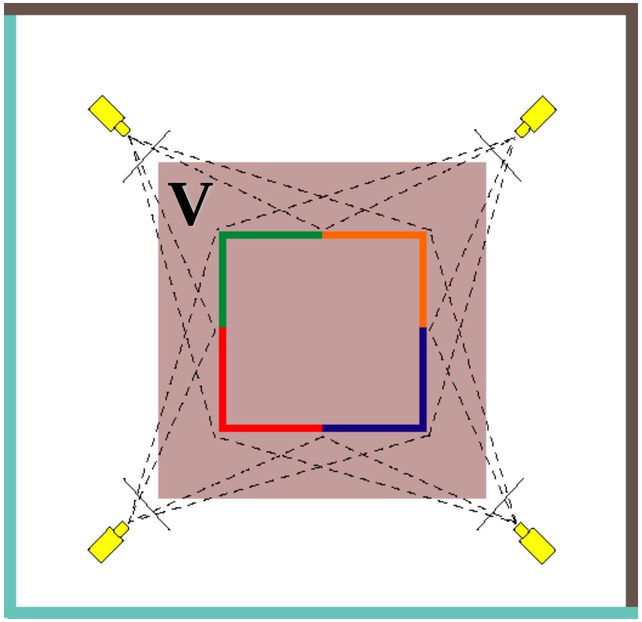


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the current surface
- Project to visible input images
- Carve if not photo-consistent
- Repeat until convergence

K. N. Kutulakos and S. M. Seitz, **A Theory of Shape by Space Carving**, *ICCV* 1999

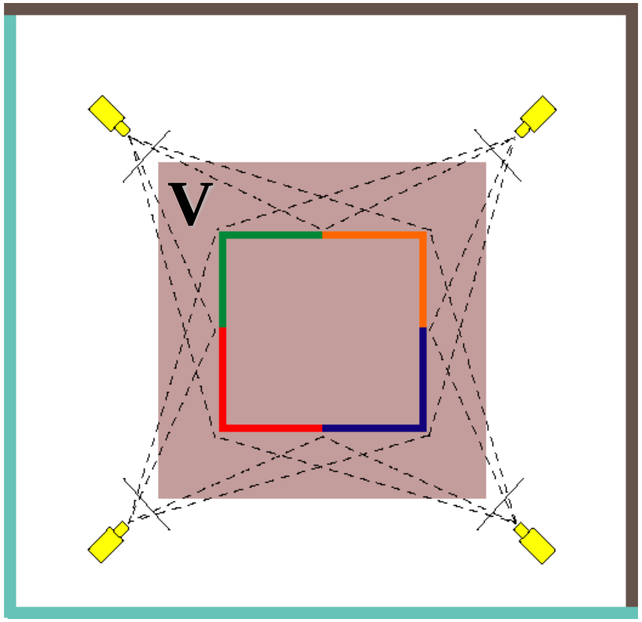
Which shape do you get?



True Scene

Source: S. Seitz

Which shape do you get?



True Scene

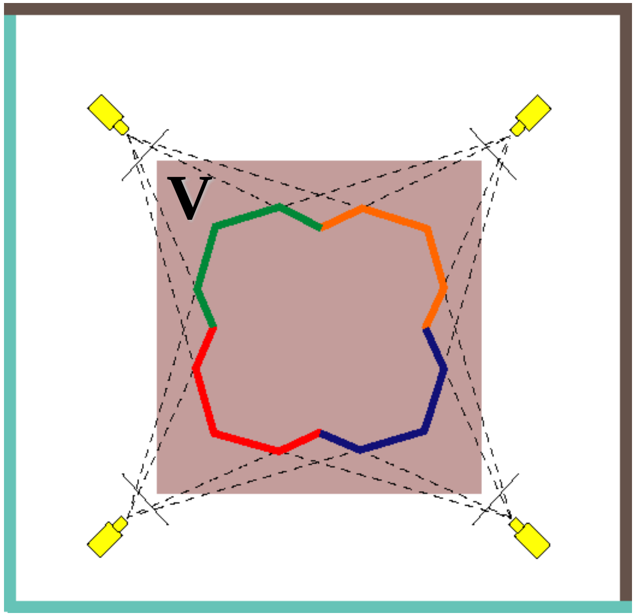
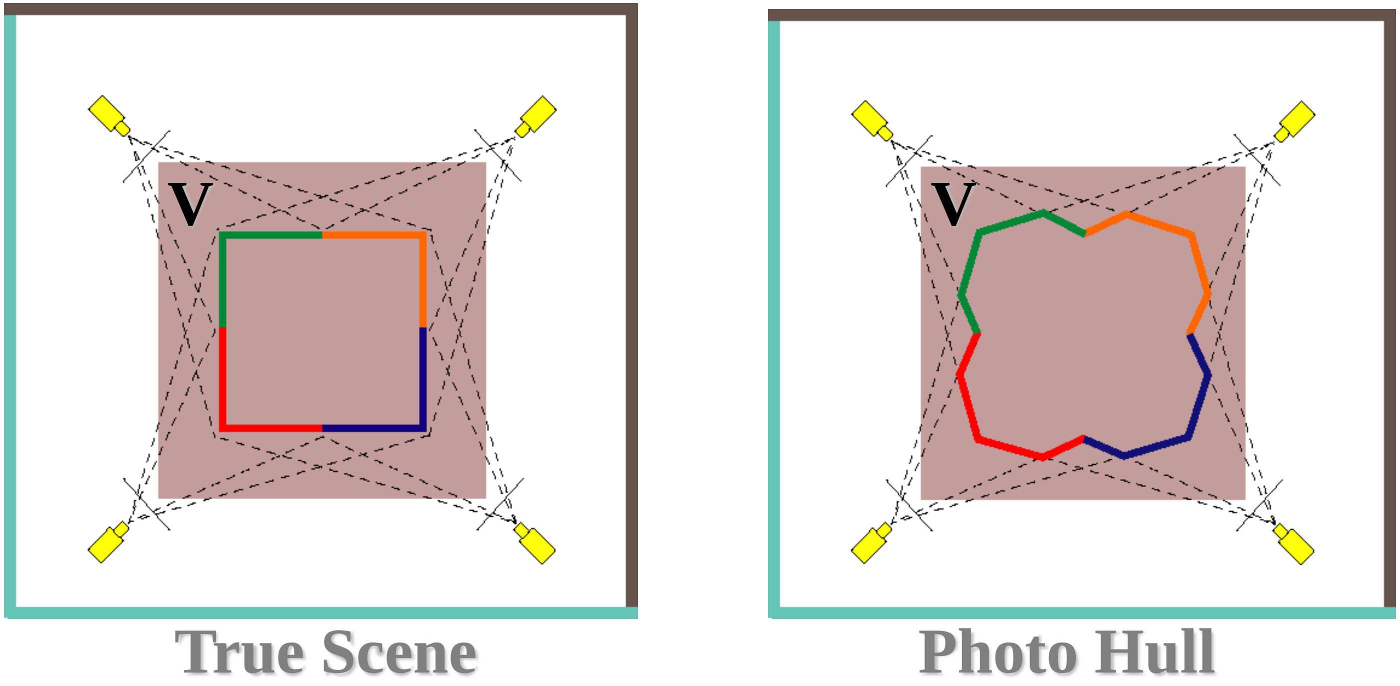


Photo Hull

Source: S. Seitz

Which shape do you get?



The Photo Hull *is the UNION of all photo-consistent scenes in V*

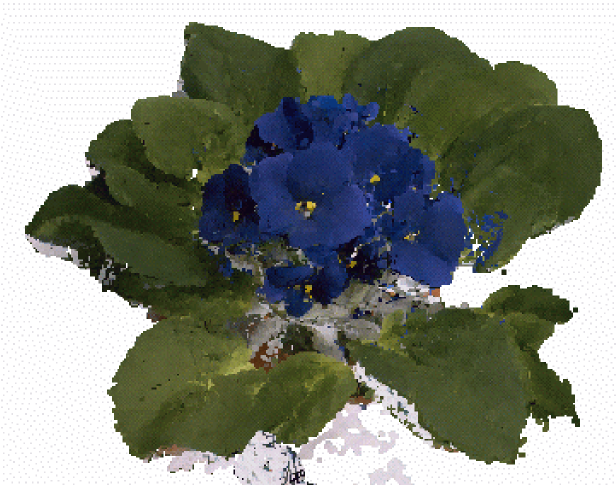
- It is a photo-consistent scene reconstruction
- Tightest possible bound on the true scene

Source: S. Seitz

Space Carving Results: African Violet

Input Image (1 of 45)

Reconstruction



Reconstruction

Reconstruction

Source: S. Seitz

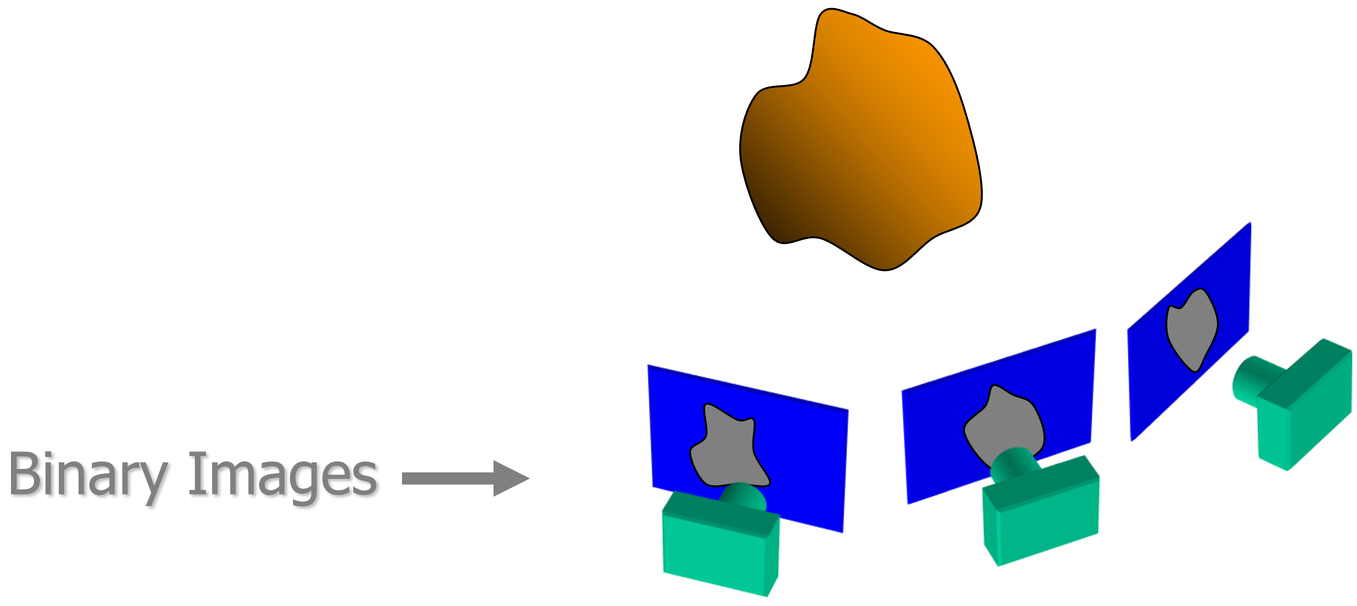
Space Carving Results: Hand

Input Image (1 of 100)

Views of Reconstruction

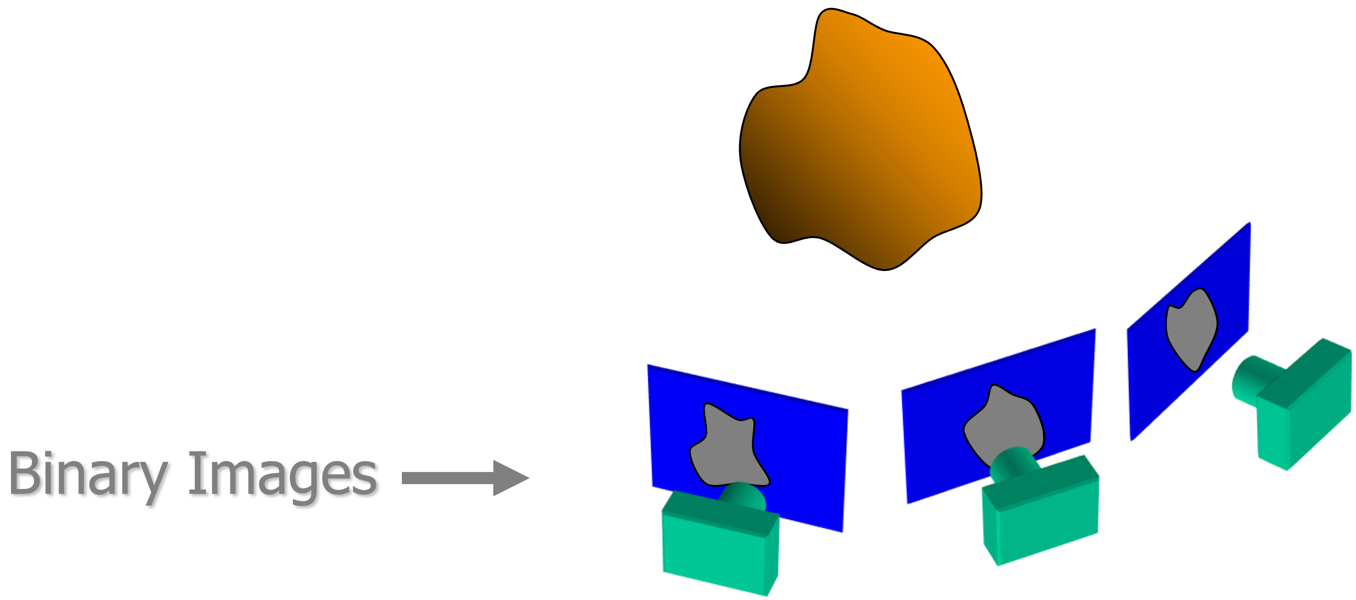
Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object’s silhouette in all views



Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

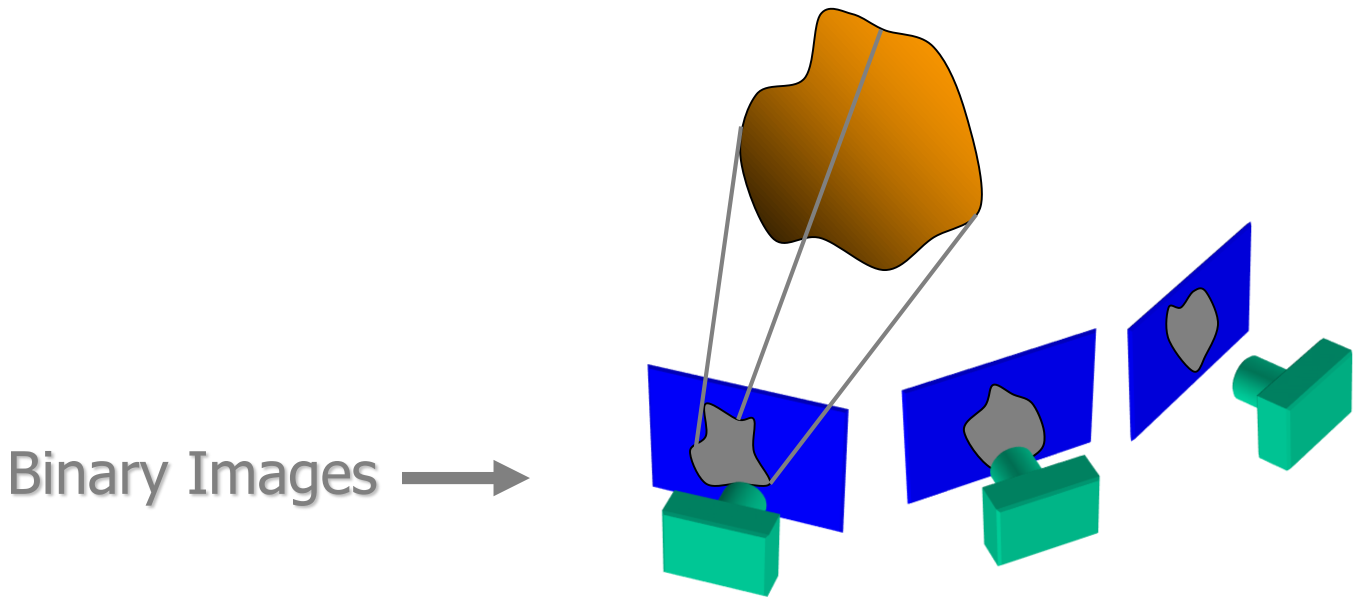


Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

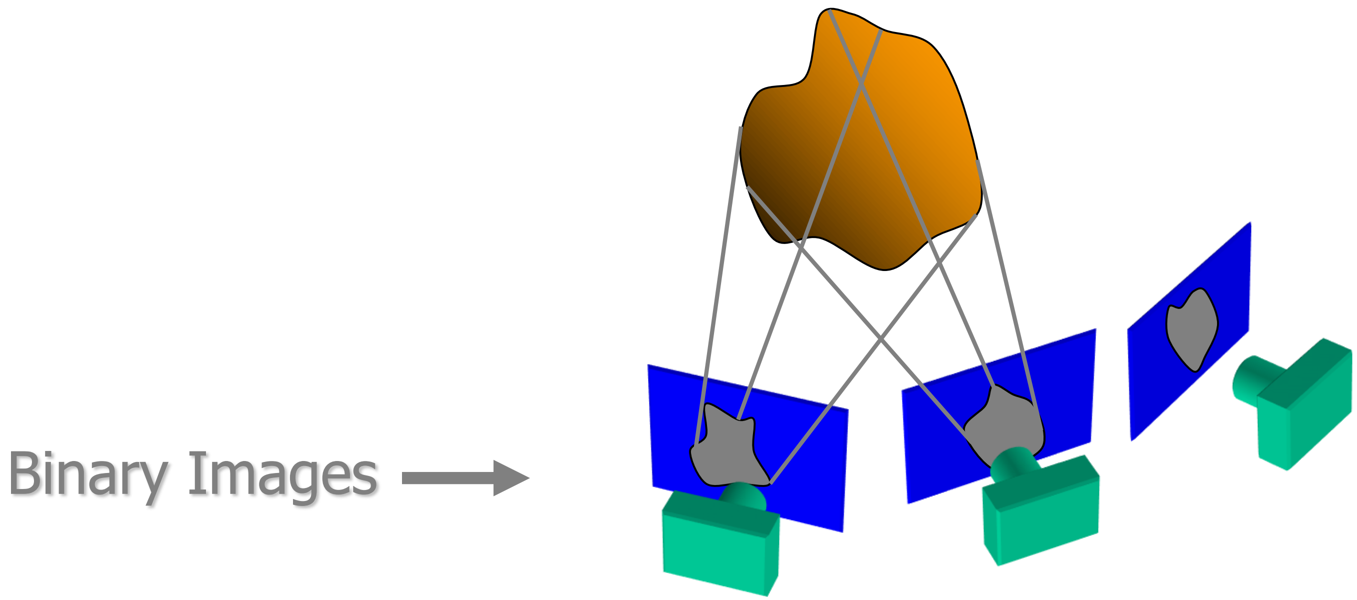


Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views



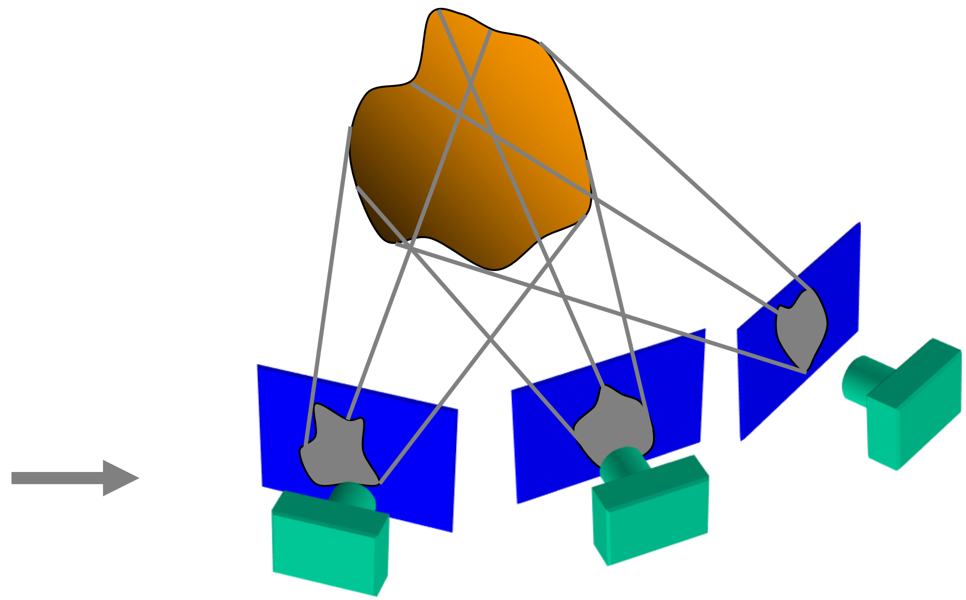
Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

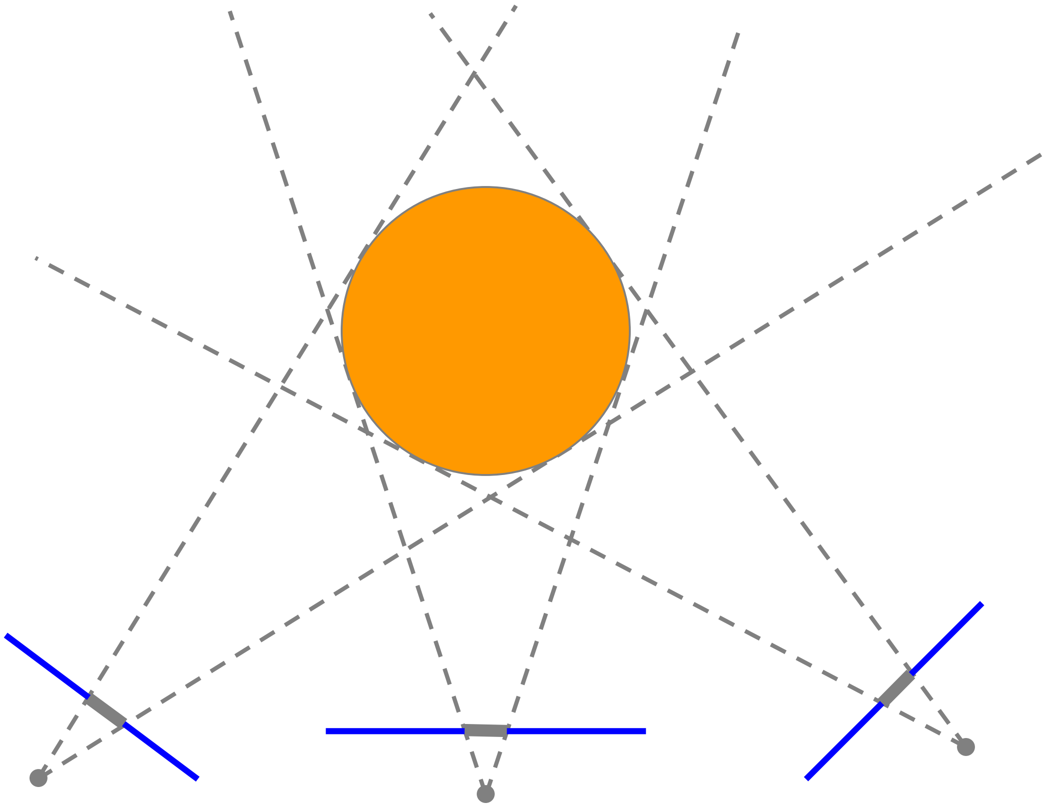
Binary Images



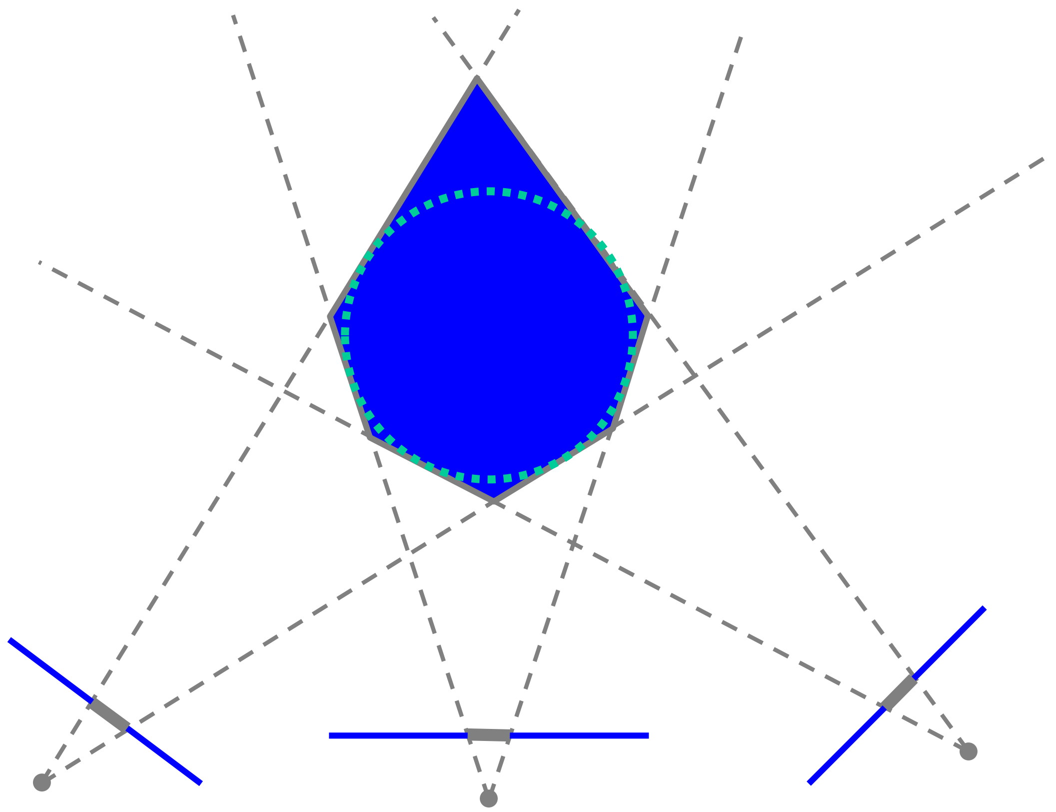
Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

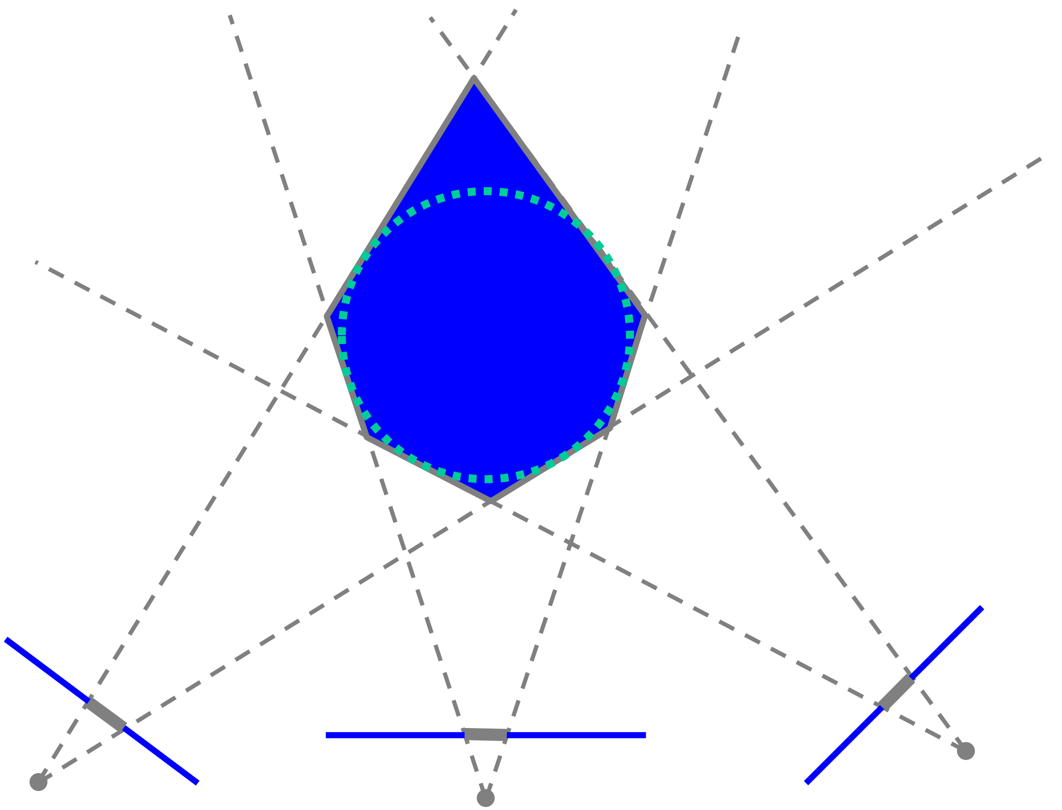
Volume intersection



Volume intersection



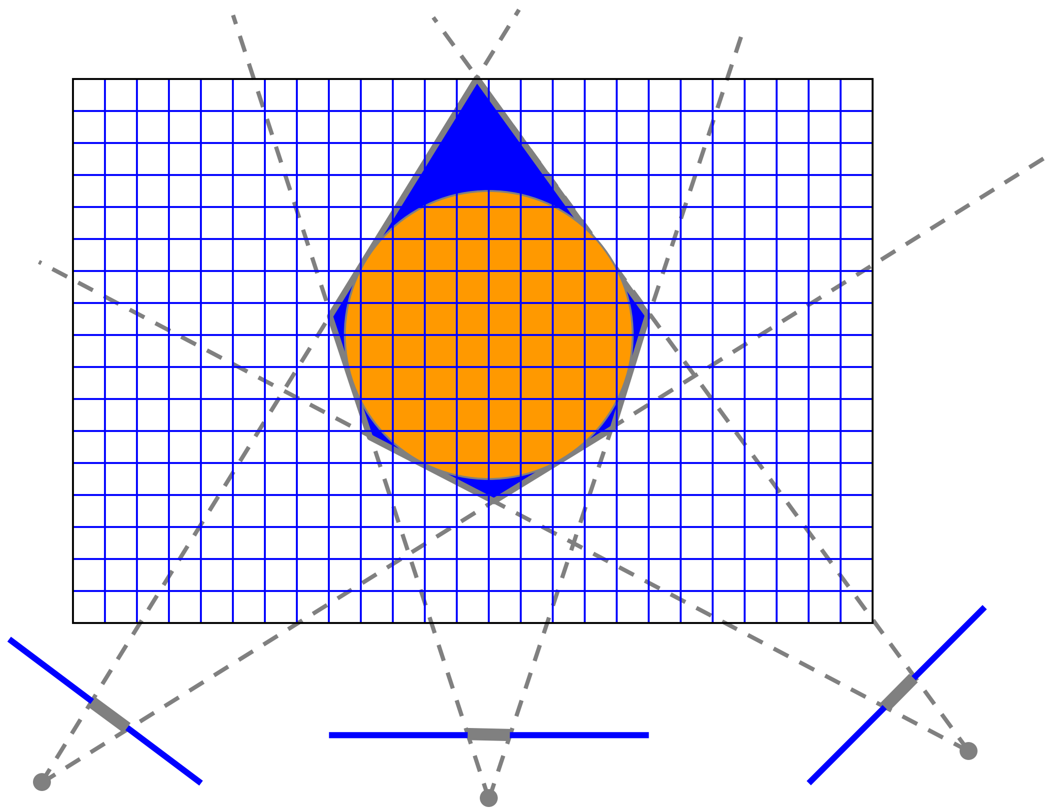
Volume intersection



Reconstruction Contains the True Scene

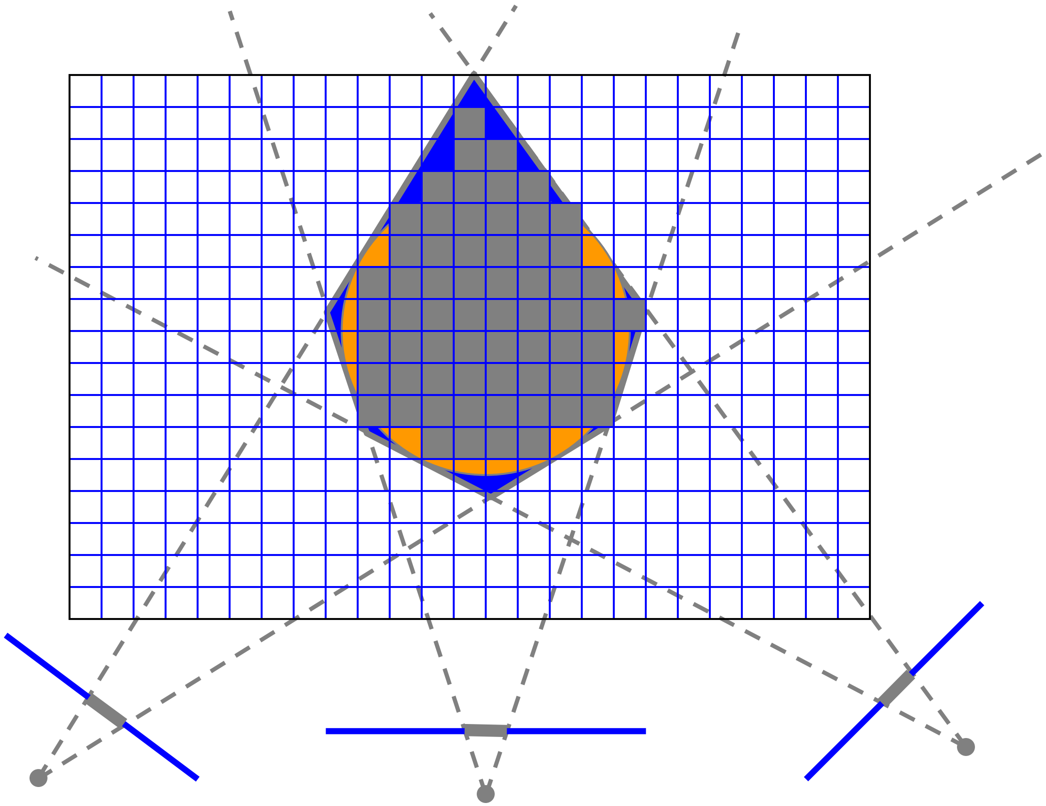
- But is generally not the same

Voxel algorithm for volume intersection



Color voxel black if on silhouette in every image

Voxel algorithm for volume intersection



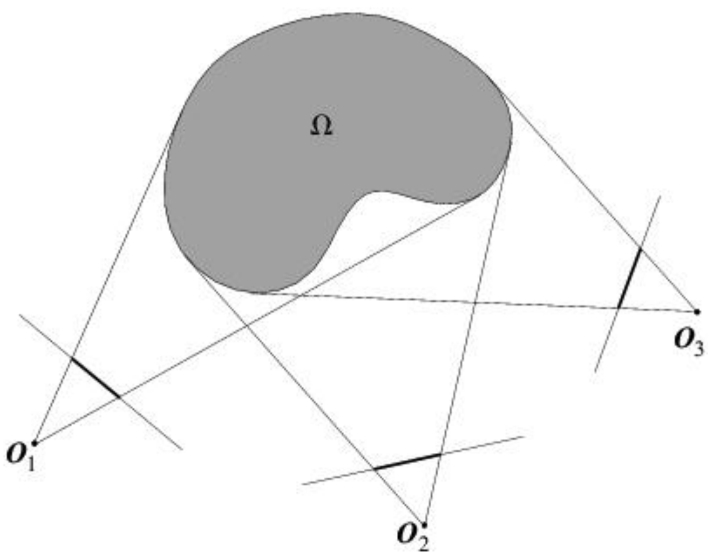
Color voxel black if on silhouette in every image

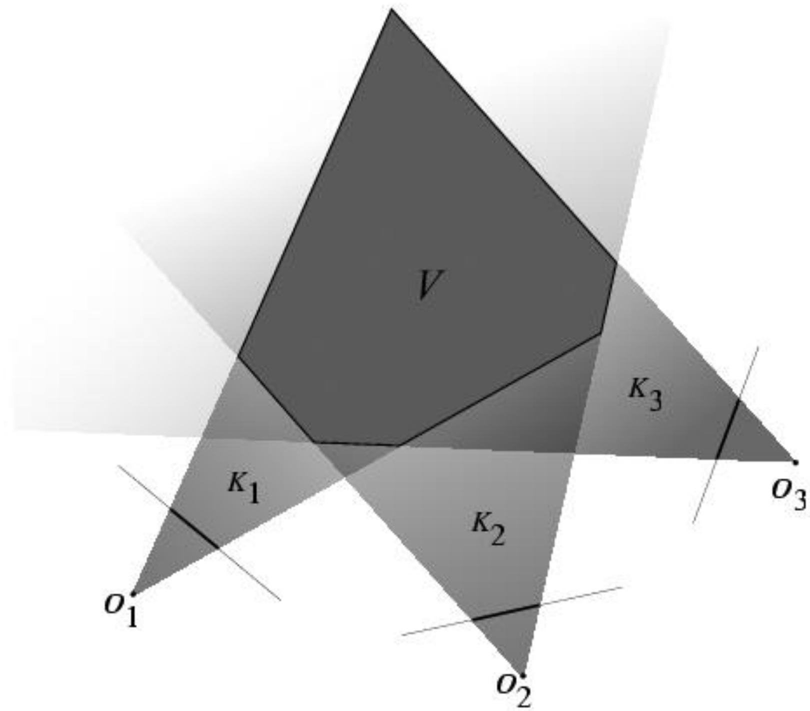
Properties of Volume Intersection

- Pros
  - Easy to implement, fast
- Cons
  - No concavities

Properties of Volume Intersection

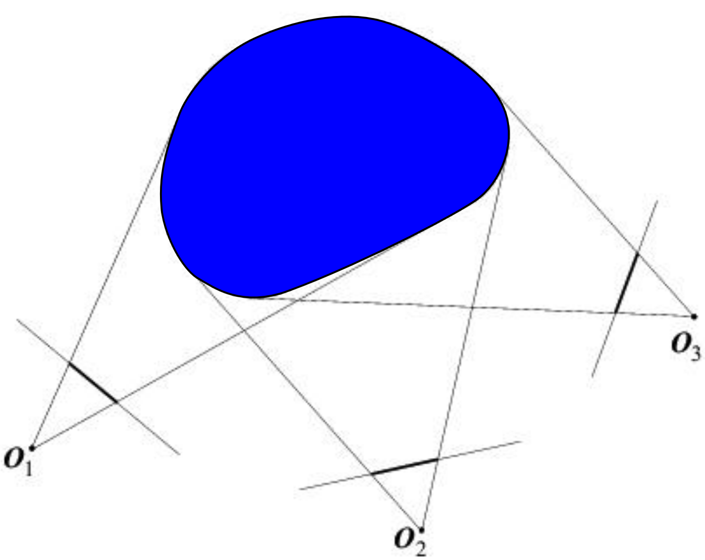
- Pros
  - Easy to implement, fast
- Cons
  - No concavities

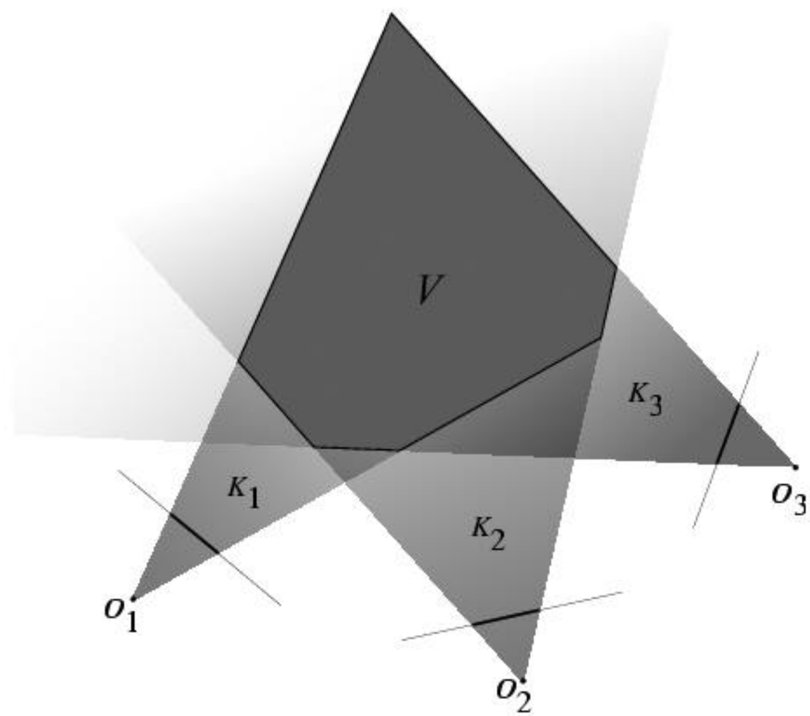




Properties of Volume Intersection

- Pros
  - Easy to implement, fast
- Cons
  - No concavities

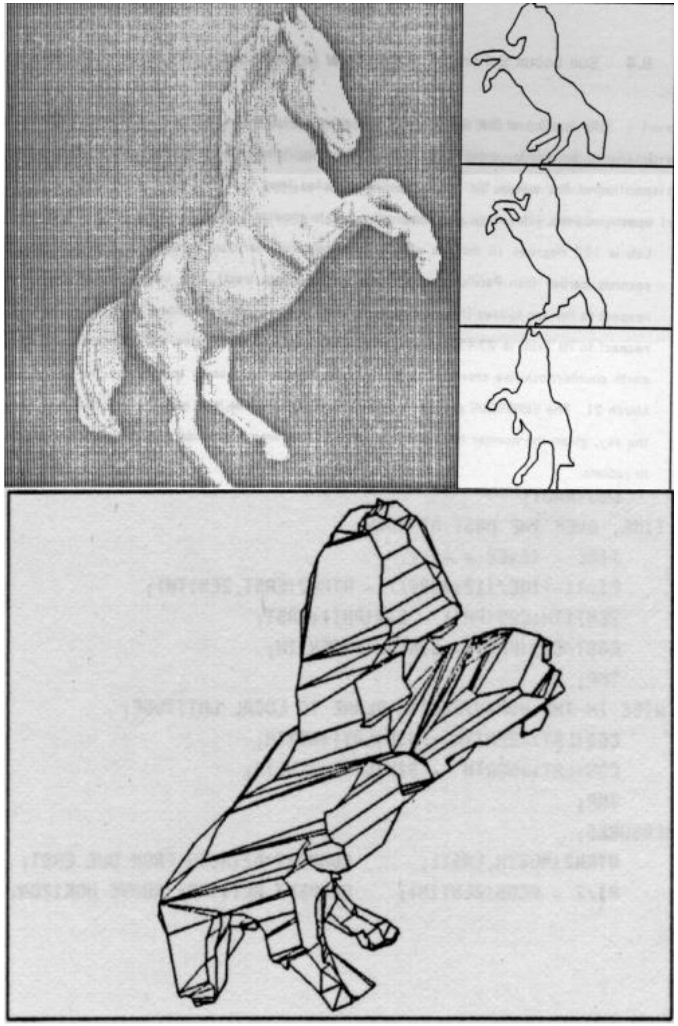




Properties of Volume Intersection

- Pros
  - Easy to implement, fast
- Cons
  - No concavities
  - Reconstruction is not photo-consistent if texture information is available
  - Requires silhouette extraction

Polyhedral volume intersection

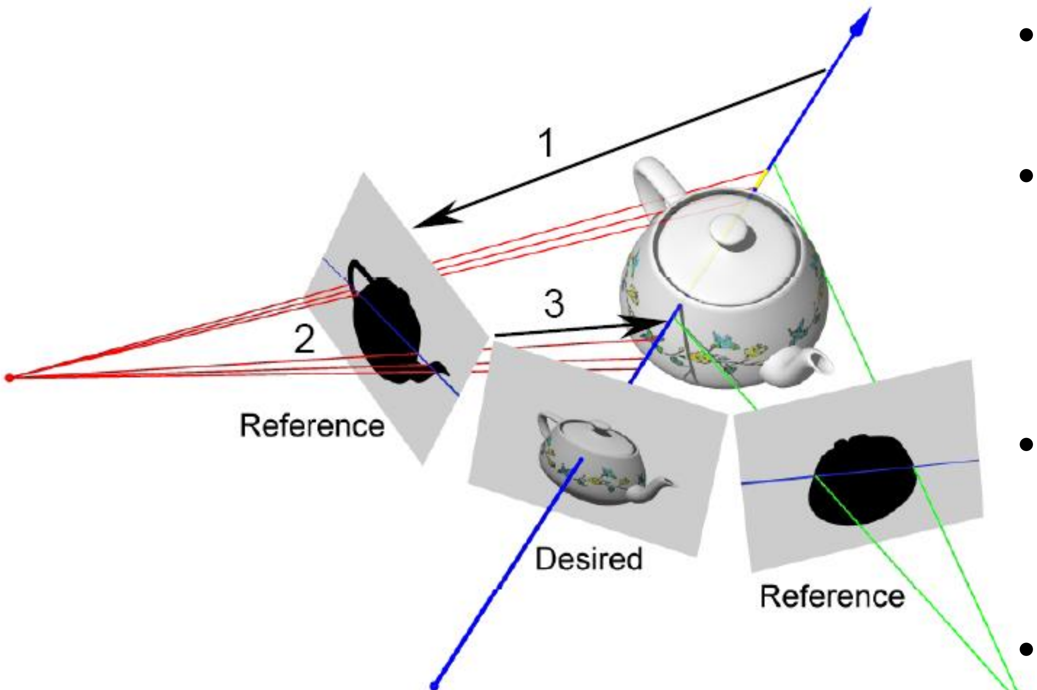


B. Baumgart, *Geometric Modeling for Computer Vision*, Stanford Artificial Intelligence Laboratory, Memo no. AIM-249, Stanford University, October 1974.

Polyhedral volume intersection: Pros and cons

- Pros
  - No voxelization artifacts
- Cons
  - Depends on discretization of outlines
  - Numerical problems when polygons to be intersected are almost coplanar
  - Does not take advantage of epipolar geometry

Image-based visual hulls

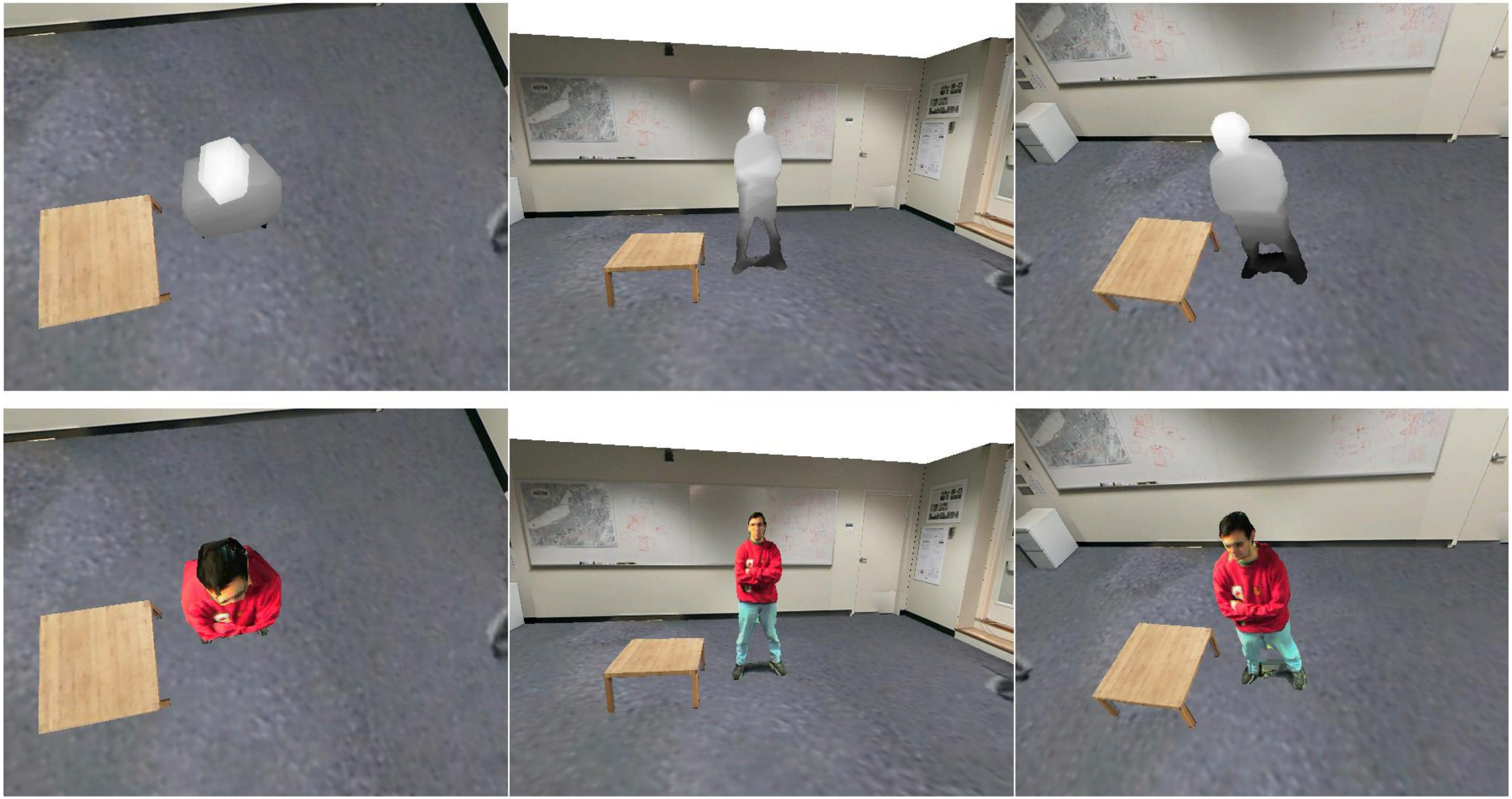


Virtual view

- Pick a pixel in the virtual view
- Project corresponding visual ray into every other view to get a set of epipolar lines
- Find intervals where epipolar lines overlap with silhouettes
- Lift intervals back onto the 3D ray and find their intersection

Wojciech Matusik, Christopher Buehler, Ramesh Raskar, Steven Gortler, and Leonard McMillan. [Image-based Visual Hulls.]() In SIGGRAPH 2000

Image-based visual hulls



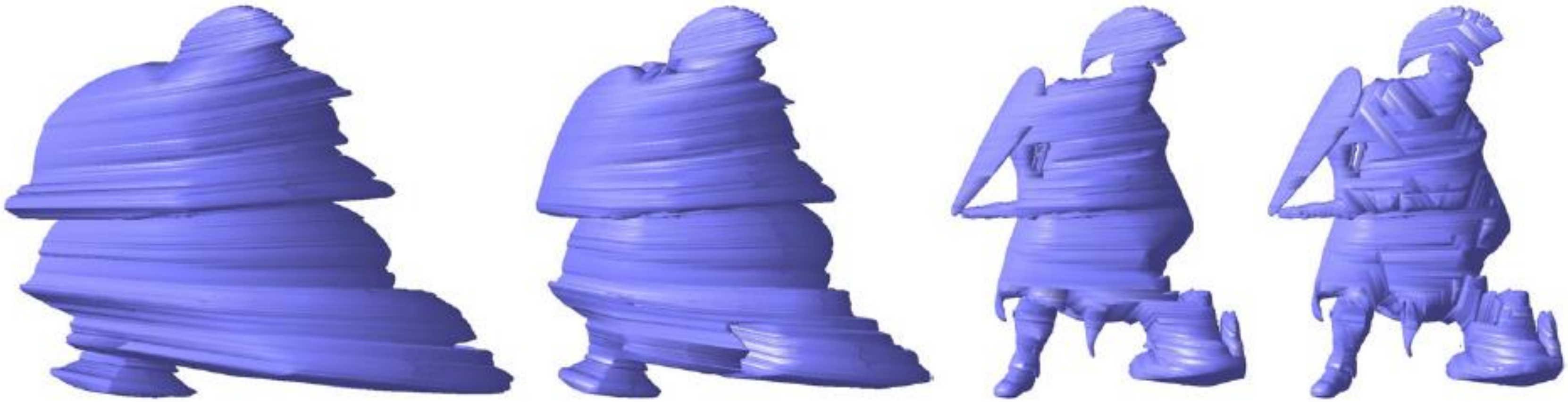
Wojciech Matusik, Christopher Buehler, Ramesh Raskar, Steven Gortler, and Leonard McMillan. Image-based Visual Hulls. In SIGGRAPH 2000

Image-based visual hulls: Pros and cons

- Pros
  - Can work in real time
  - Takes advantage of epipolar geometry
- Cons
  - Need to recompute the visual hull every time the virtual view is changed

Understanding the shape of visual hulls

- In principle, we can use epipolar geometry to compute an *exact* representation of the visual hull in 3D



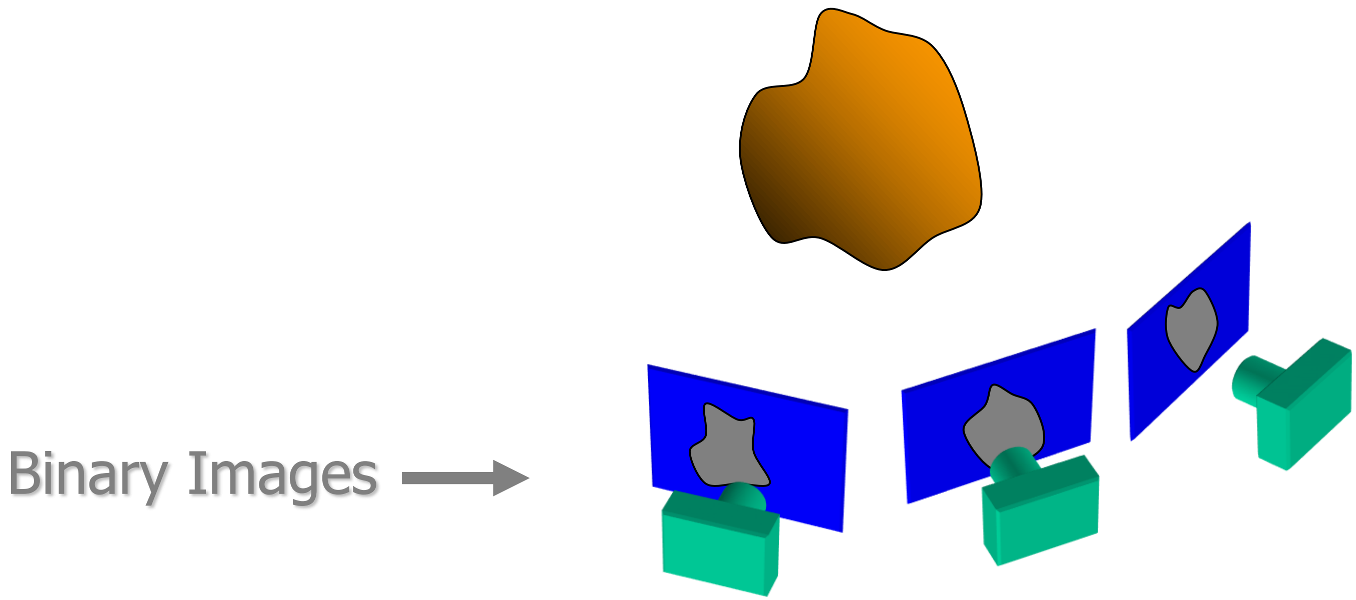
S. Lazebnik, Y. Furukawa, and J. Ponce, "Projective Visual Hulls", IJCV 2007.

Review: Multi-view stereo

- Multiple-baseline stereo
  - Pick one input view as reference
  - Inverse depth instead of disparity
- Plane sweep stereo
  - Virtual view
- Volumetric stereo
  - Photo-consistency
  - Space carving
- Shape from silhouettes
  - Visual hull: intersection of visual cones

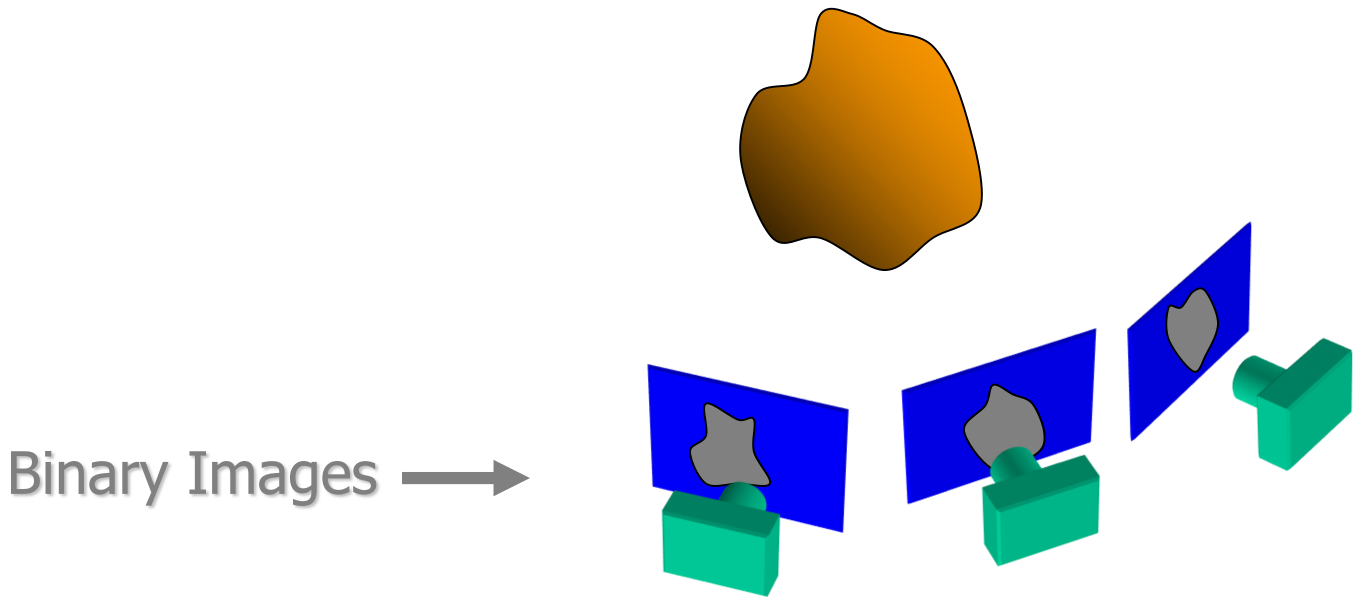
Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views



Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

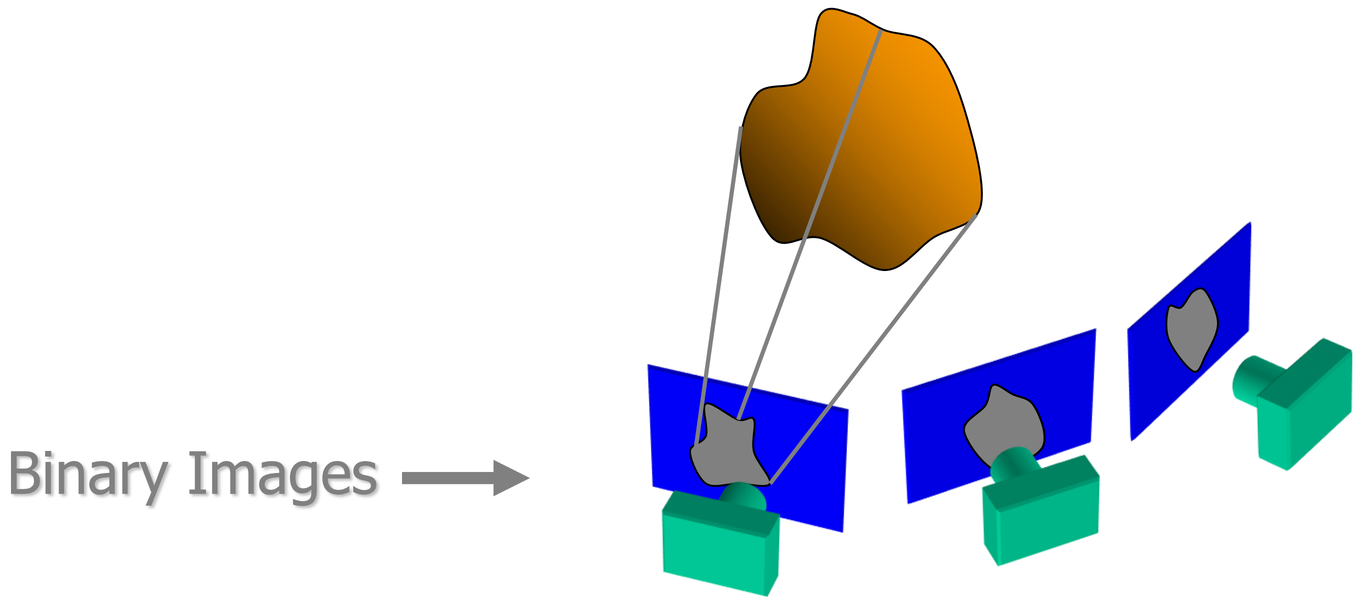


Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

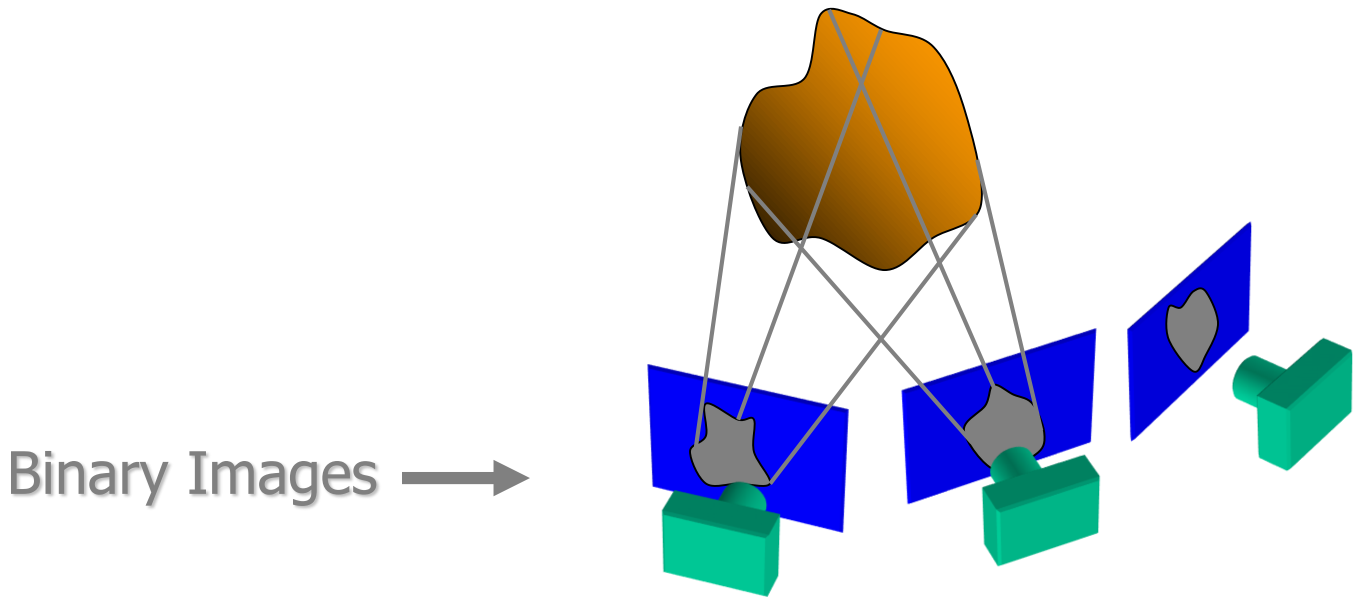


Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views

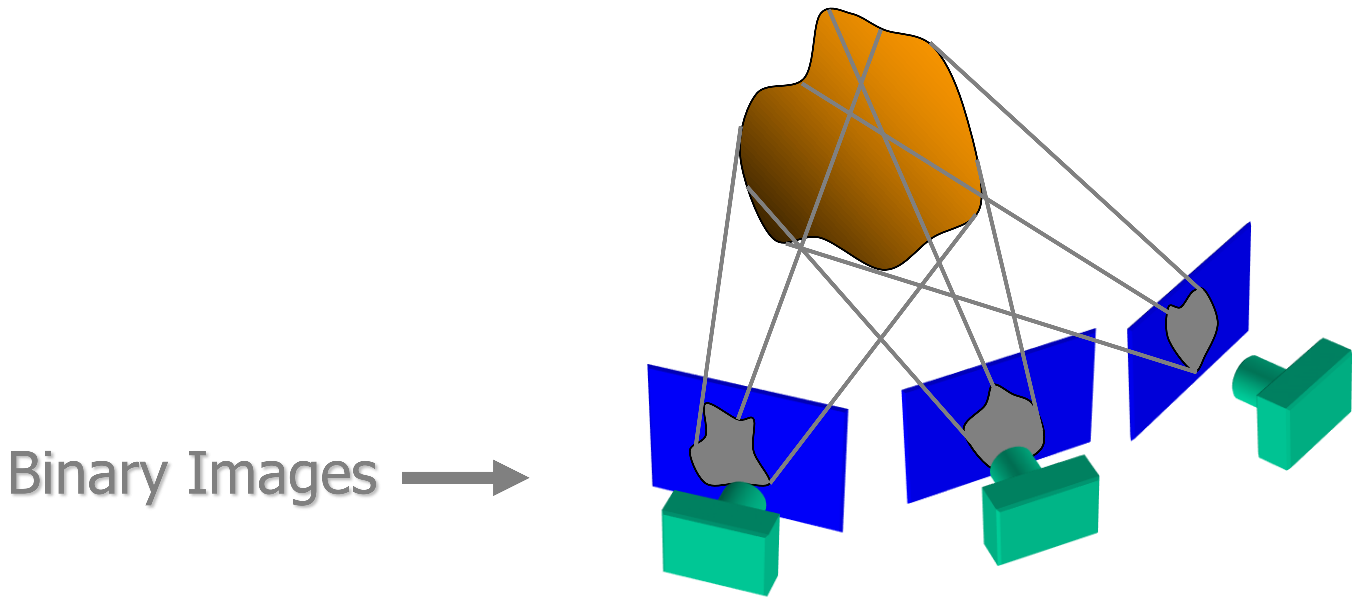


Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

Reconstruction from Silhouettes

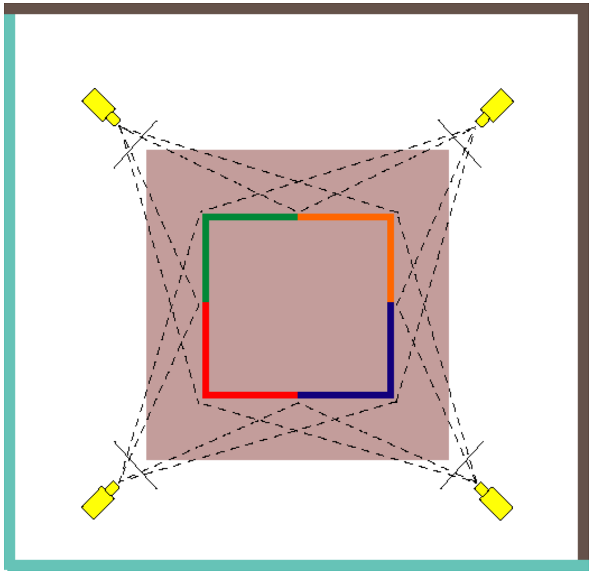
- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views



Finding the silhouette-consistent shape (*visual hull*):

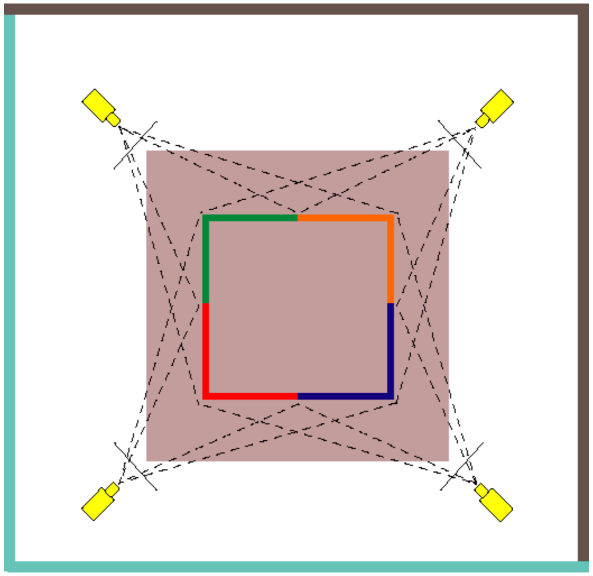
- *Backproject* each silhouette
- Intersect backprojected volumes

Photo-consistency vs. silhouette-consistency



True Scene

Photo-consistency vs. silhouette-consistency



True Scene

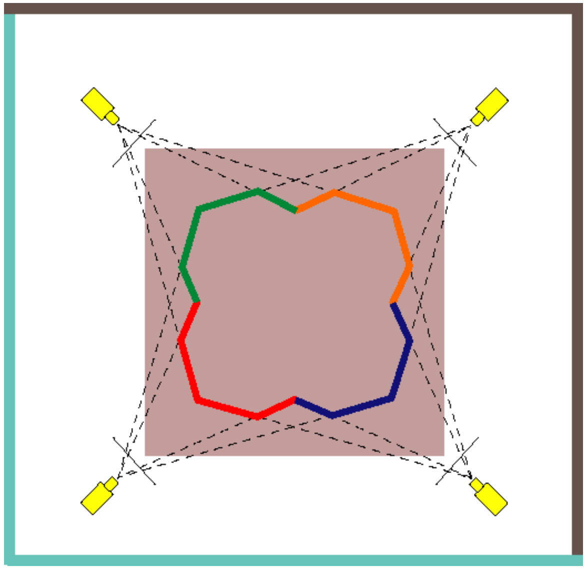
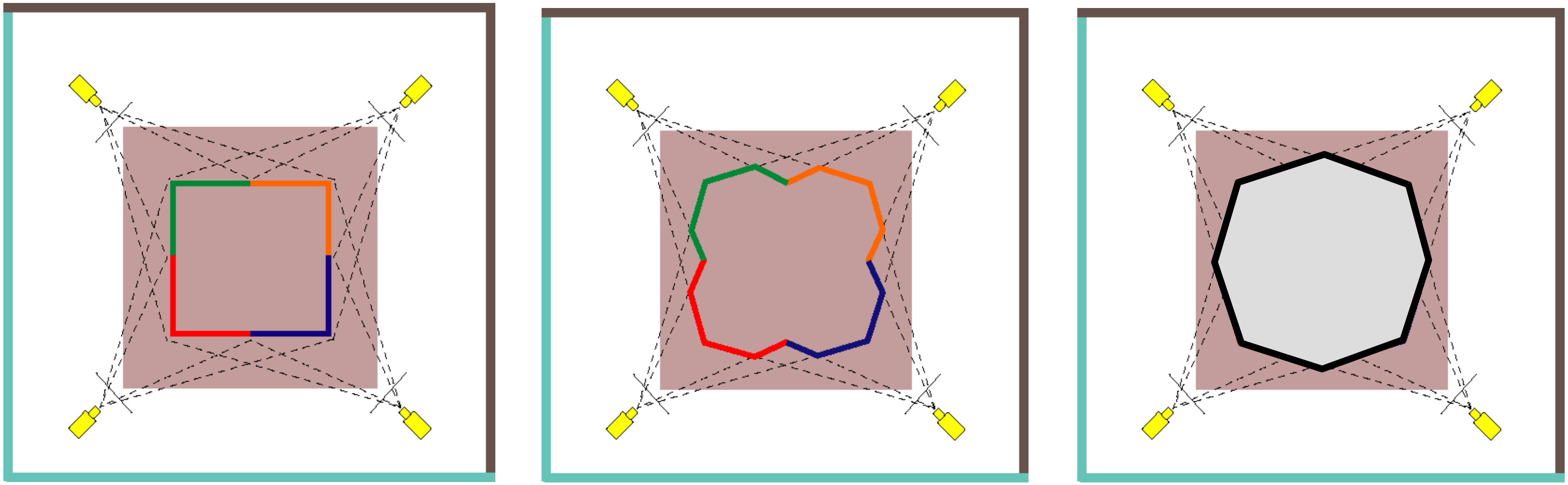


Photo Hull

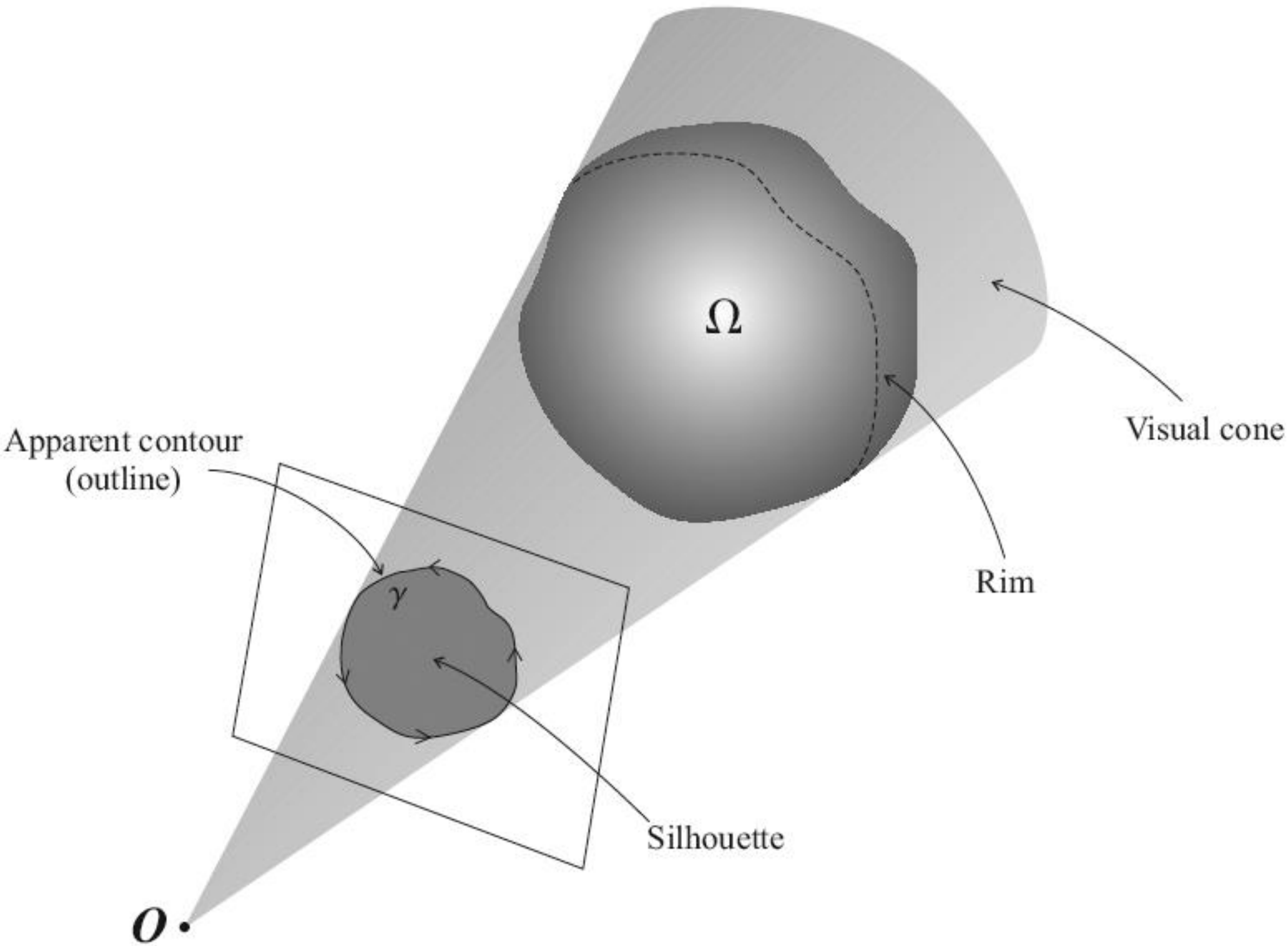
Photo-consistency vs. silhouette-consistency



True Scene **Photo Hull** **Visual Hull**

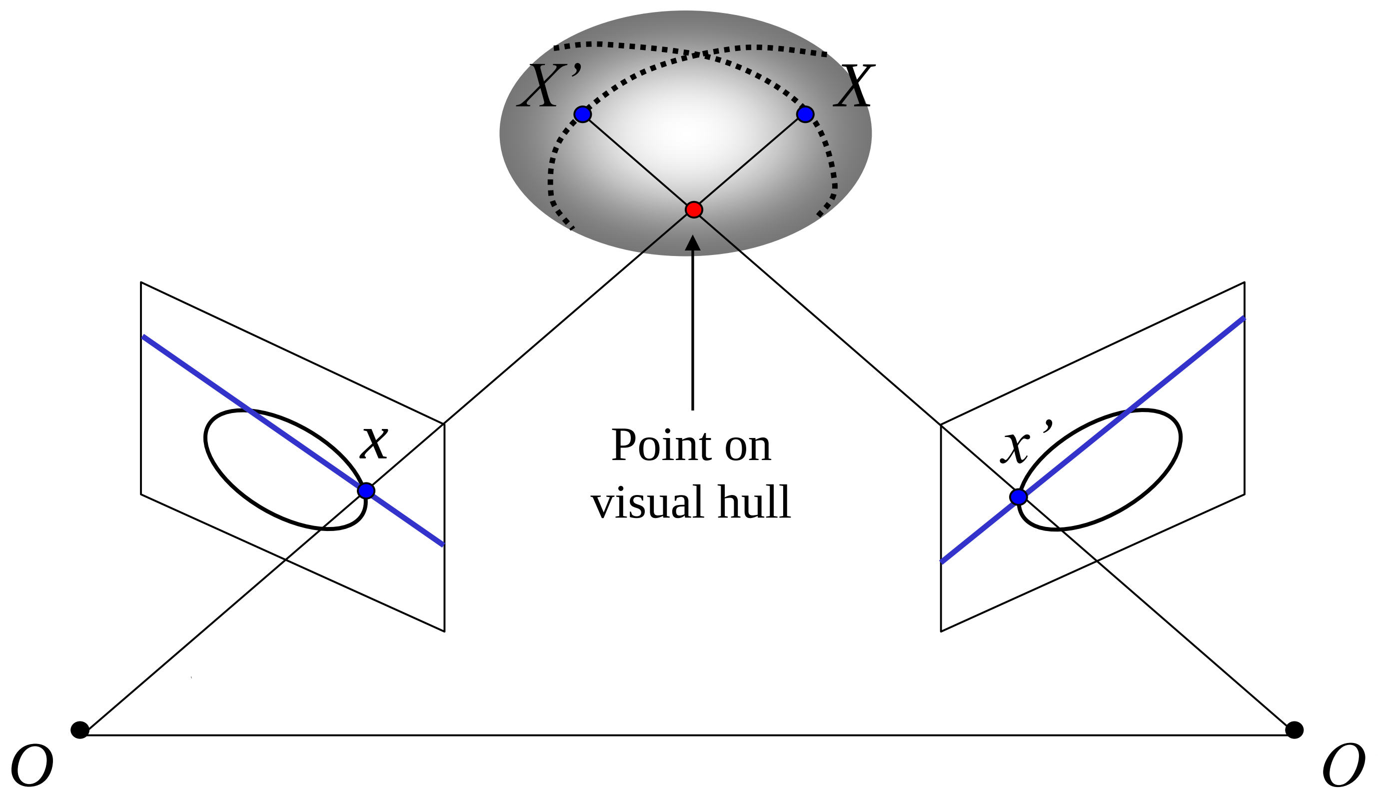
Understanding the shape of visual hulls

- What part of the visual hull belongs to the surface?



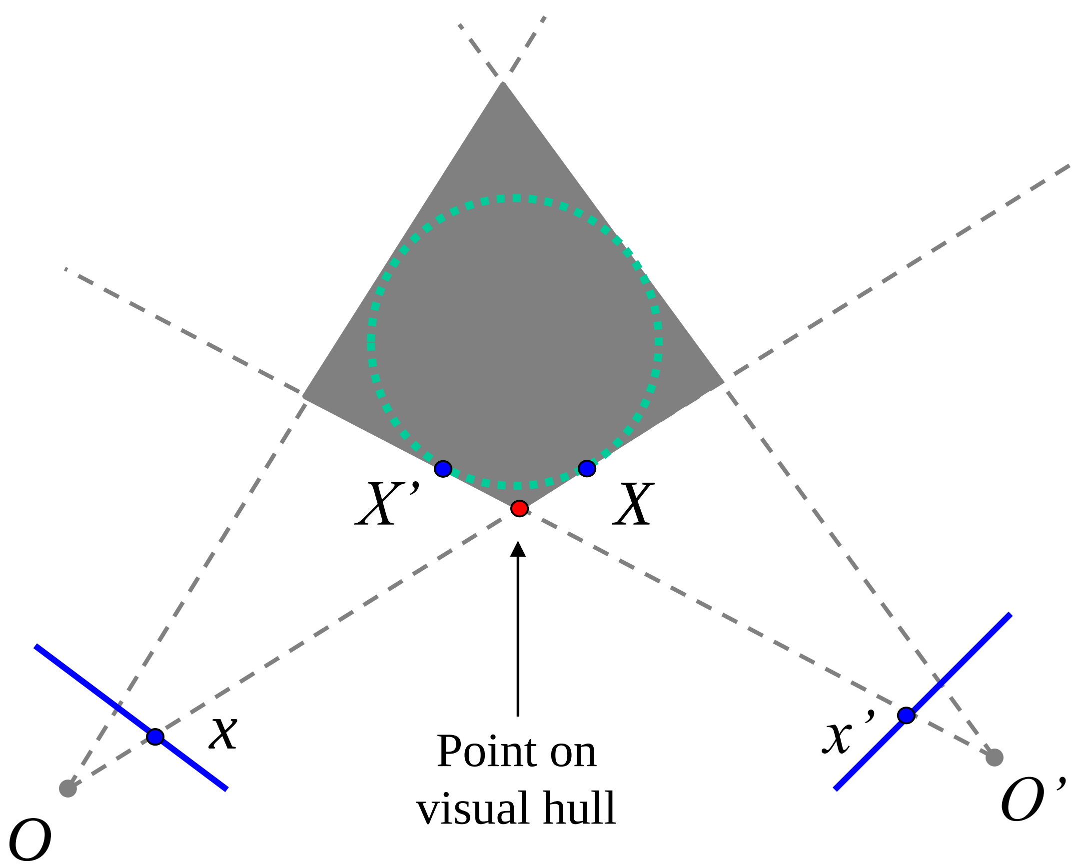
Understanding the shape of visual hulls

- The visual hull does not correspond to the true surface because the epipolar constraint is not valid for silhouette points



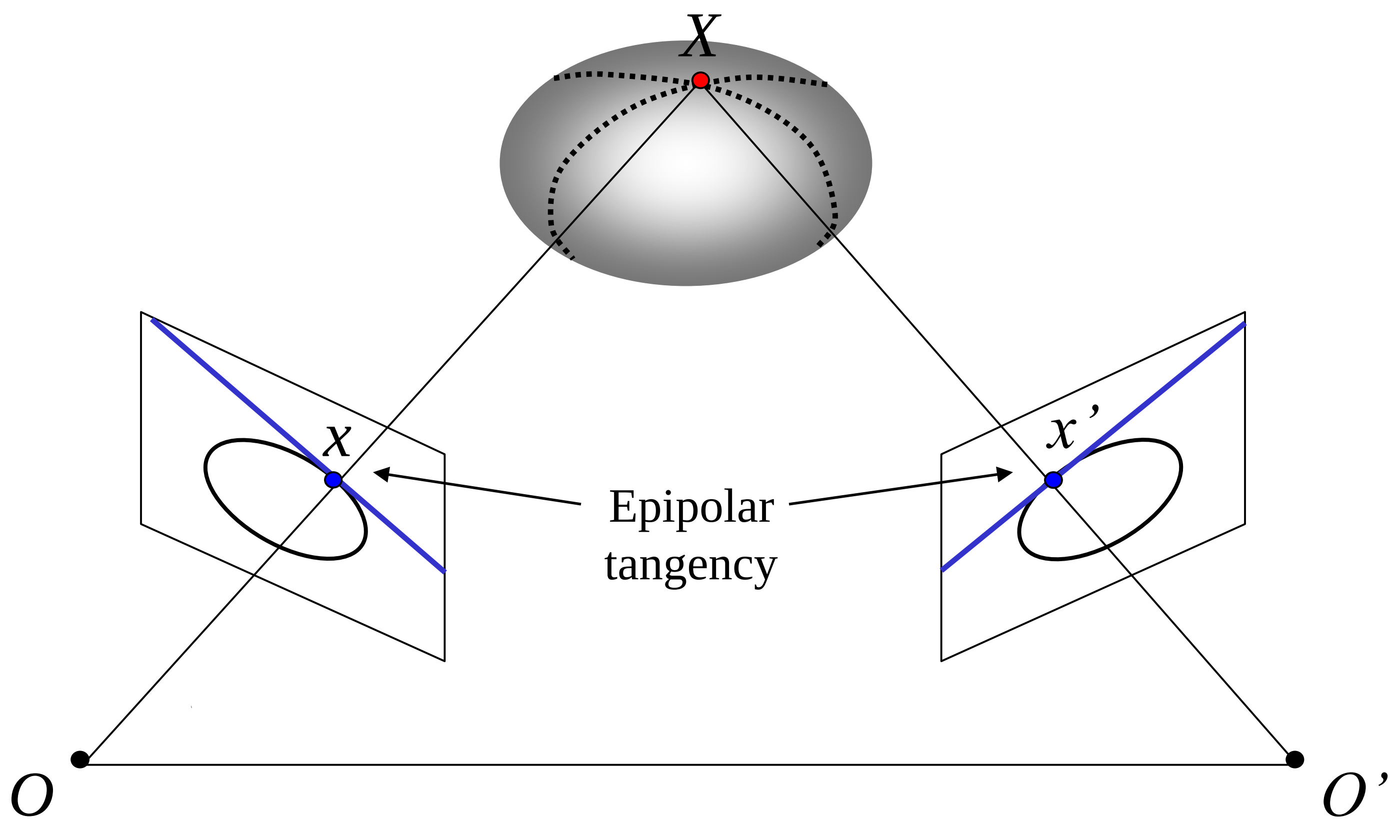
Understanding the shape of visual hulls

- The visual hull does not correspond to the true surface because the epipolar constraint is not valid for silhouette points



Understanding the shape of visual hulls

- The visual hull does not correspond to the true surface because the epipolar constraint is not valid for silhouette points
- Exception: *frontier points*



Carved visual hulls

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

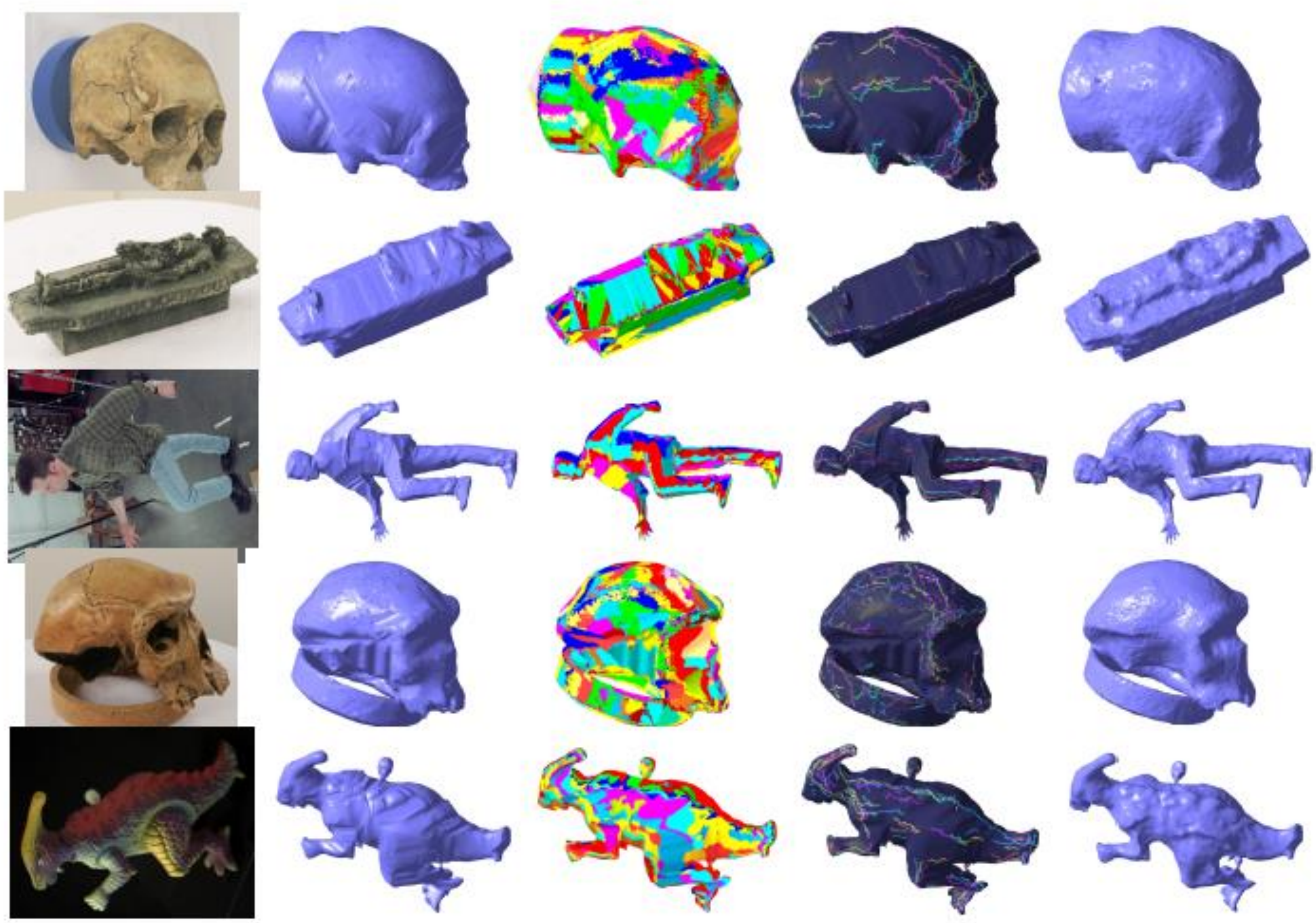
Yasutaka Furukawa and Jean Ponce, **Carved Visual Hulls for Image-Based Modeling**, ECCV 2006.

Carved visual hulls

1. Compute visual hull
2. Use dynamic programming to find rims and constrain them to be fixed
3. Carve the visual hull to optimize photo-consistency

Yasutaka Furukawa and Jean Ponce, **Carved Visual Hulls for Image-Based Modeling**, ECCV 2006.

Carved visual hulls



Yasutaka Furukawa and Jean Ponce, **[Carved Visual Hulls for Image-Based Modeling]()**, ECCV 2006.

Carved visual hulls: Pros and cons

- Pros
  - Visual hull gives a reasonable initial mesh that can be iteratively deformed
- Cons
  - Need silhouette extraction
  - Have to compute a lot of points that don't lie on the object
  - Finding rims is difficult
  - The carving step can get caught in local minima

- Possible solution: use sparse feature correspondences as initialization

Feature-based stereo matching

T. Tuytelaars and L. Van Gool, [***"Matching Widely Separated Views based on Affine Invariant Regions"***]() Int. Journal on Computer Vision, 59(1), pp. 61-85, 2004.

Feature-based stereo matching

- Pros
  - Robust to clutter and occlusion
  - Only find matches at reliable points
  - Can use invariant local features to deal with foreshortening, scale change, wide baselines
- Cons
  - You only get a sparse cloud of points (or oriented patches), not a dense depth map or a complete surface

From feature matching to dense stereo

1. Extract features
2. Get 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

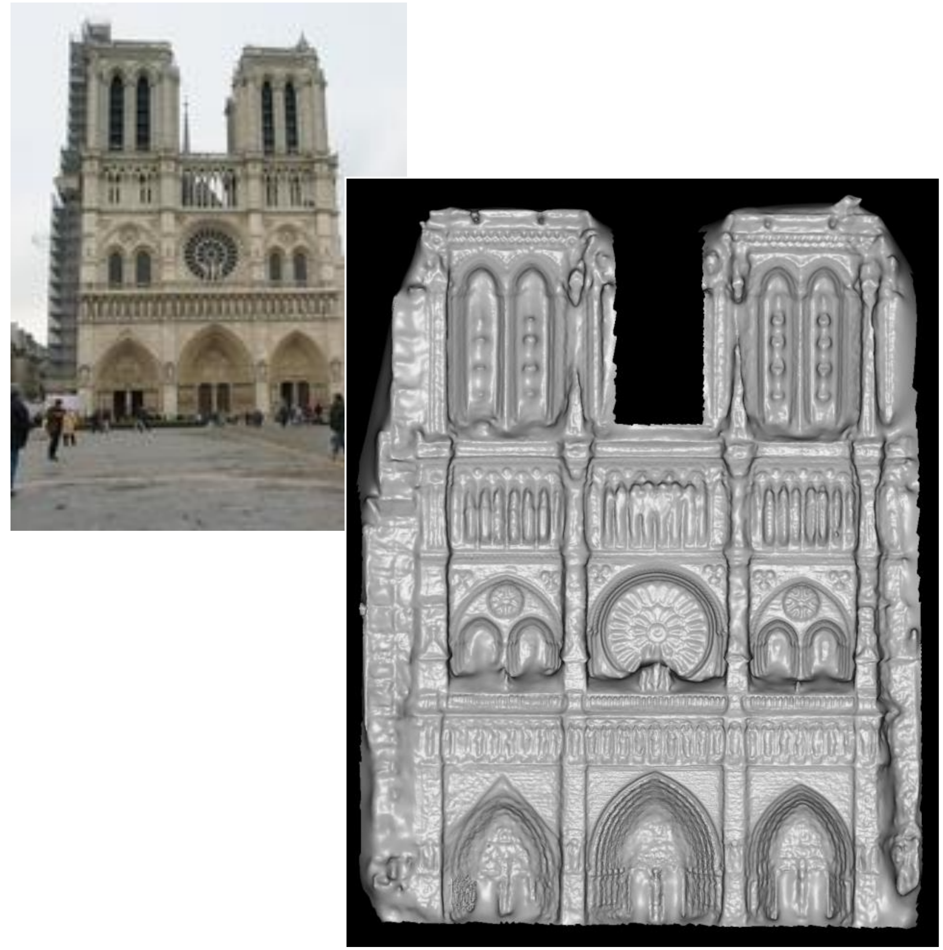
Yasutaka Furukawa and Jean Ponce, **Accurate, Dense, and Robust Multi-View Stereopsis**, CVPR 2007.

From feature matching to dense stereo

http://www-cvr.ai.uiuc.edu/~yfurukaw/

Yasutaka Furukawa and Jean Ponce, **Accurate, Dense, and Robust Multi-View Stereopsis**, CVPR 2007.

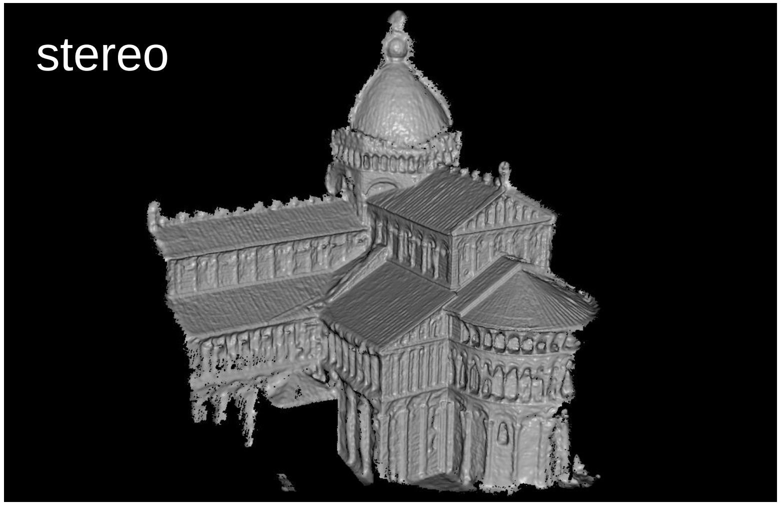
Stereo from community photo collections

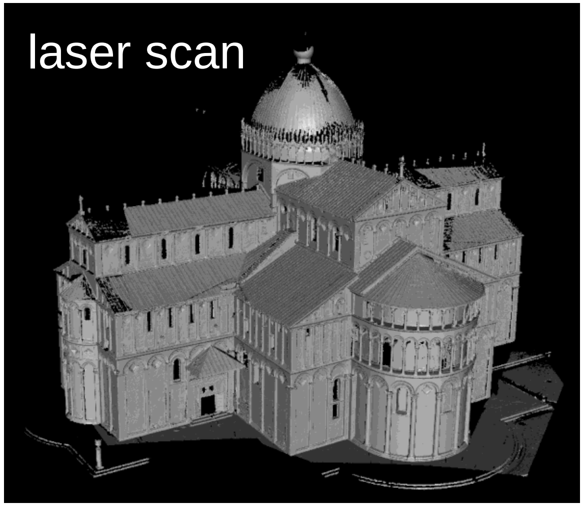


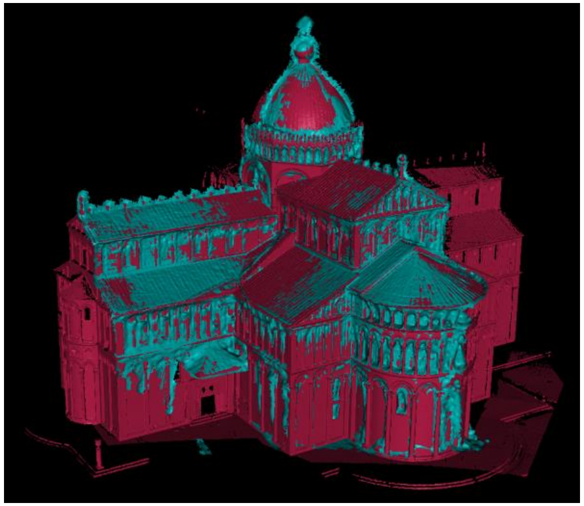
M. Goesele, N. Snavely, B. Curless, H. Hoppe, S. Seitz, Multi-View Stereo for Community Photo Collections, ICCV 2007

http://grail.cs.washington.edu/projects/mvscpc/

Stereo from community photo collections







Comparison: 90% of points within 0.128 m of laser scan (building height 51m)

M. Goesele, N. Snavely, B. Curless, H. Hoppe, S. Seitz, Multi-View Stereo for Community Photo Collections, ICCV 2007

Stereo from community photo collections

- Up to now, we’ve always assumed that camera calibration is known
- For photos taken from the Internet, we need *structure from motion* techniques to reconstruct both camera positions and 3D points

Multi-view stereo: Summary

- Multiple-baseline stereo
  - Pick one input view as reference
  - Inverse depth instead of disparity
- Plane sweep stereo
  - Virtual view
- Volumetric stereo
  - Photo-consistency
  - Space carving
- Shape from silhouettes
  - Visual hull: intersection of visual cones
  - Volumetric, polyhedral, image-based
- Carved visual hulls
- Feature-based stereo
  - From sparse to dense correspondences