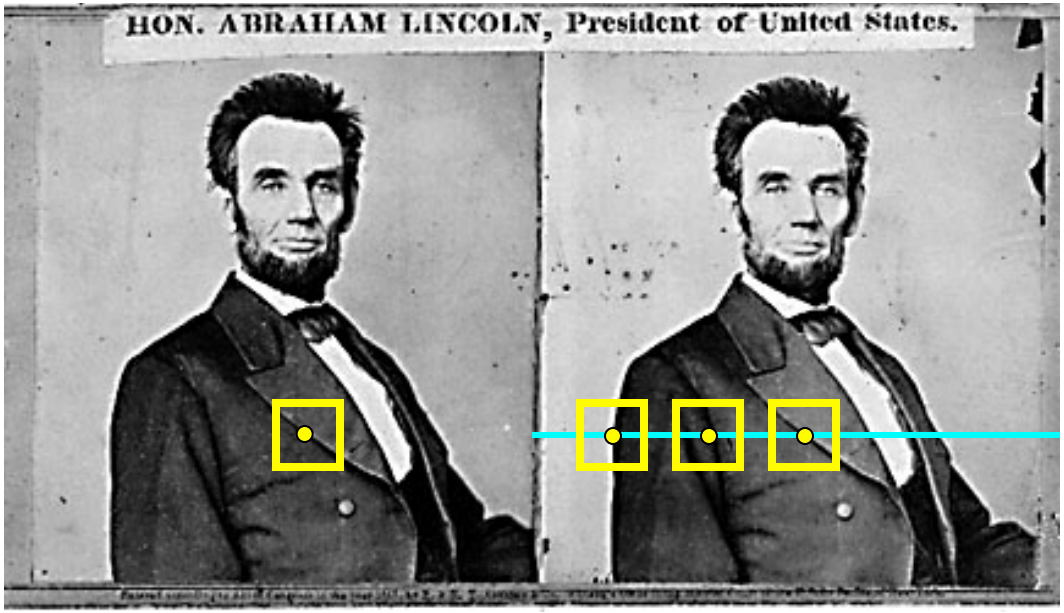
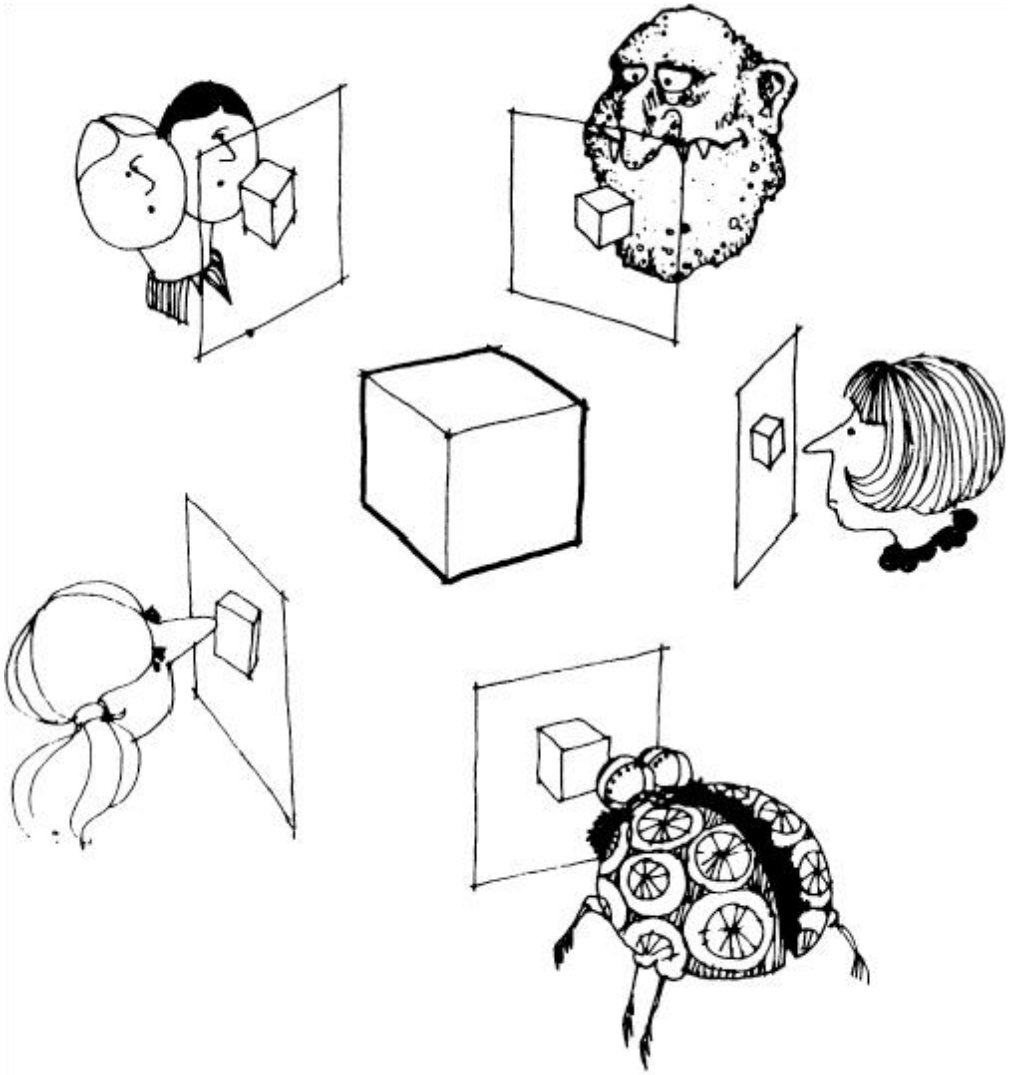
Review: Binocular stereo



- 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 depth(x) = B*f/(x-x')

Multi-view stereo



Many slides adapted from S. Seitz

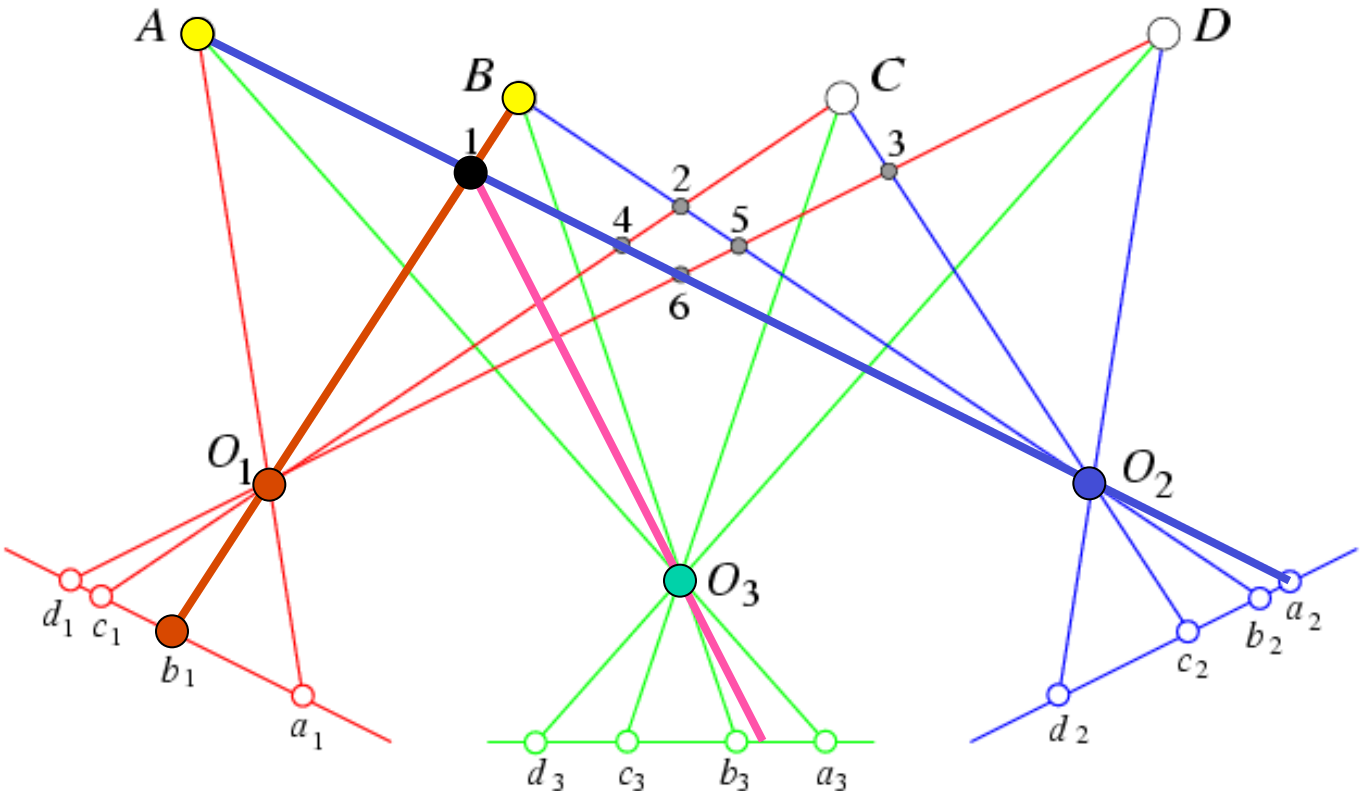
What is stereo vision?

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

What is stereo vision?

- Generic problem formulation: 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 (camera network or video sequence)
  - Calibration may be initially unknown
- “Representation of 3D shape”
  - Depth maps
  - Meshes
  - Point clouds
  - Patch clouds
  - Volumetric models
  - Layered models

Beyond two-view stereo

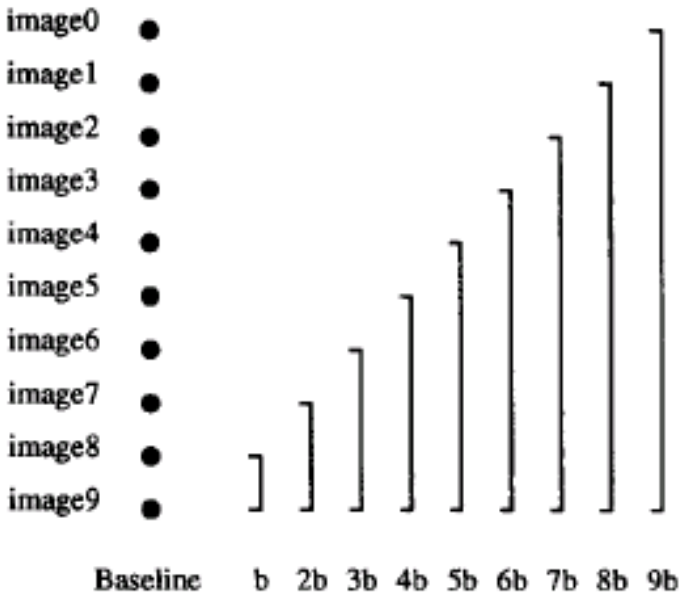


The third view can be used for verification

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

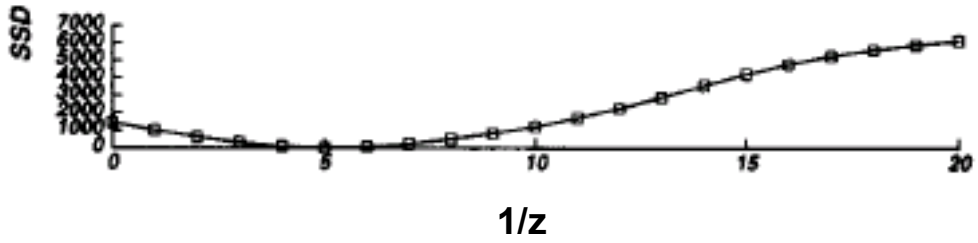
Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.



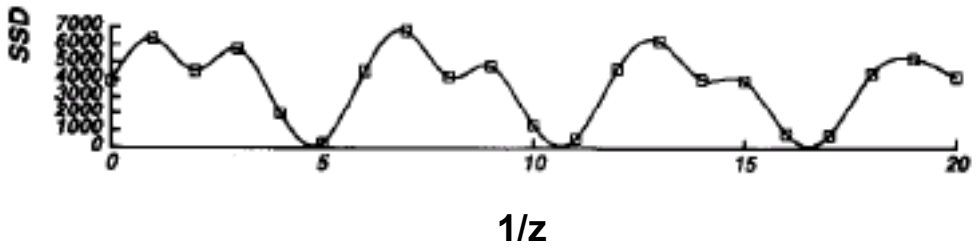
M. Okutomi and T. Kanade, "A Multiple-Baseline Stereo System," IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(4):353-363 (1993).

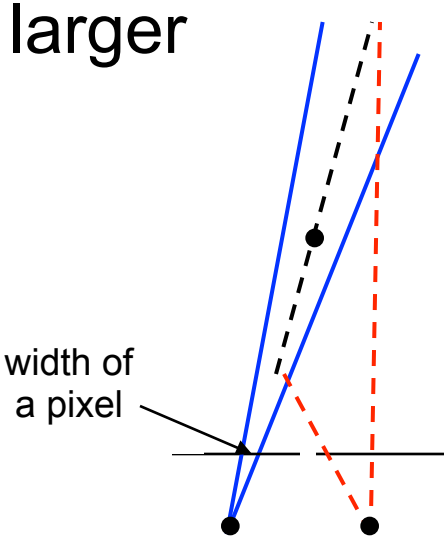
Multiple-baseline stereo

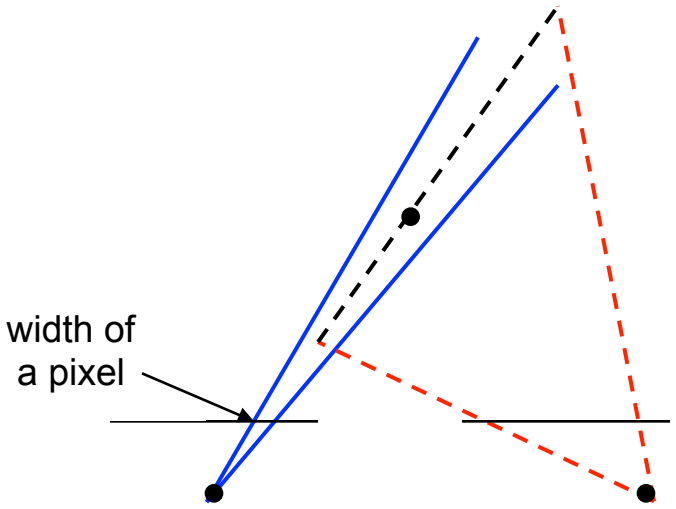
- For larger baselines, must search larger area in second image



pixel matching score







Multiple-baseline stereo

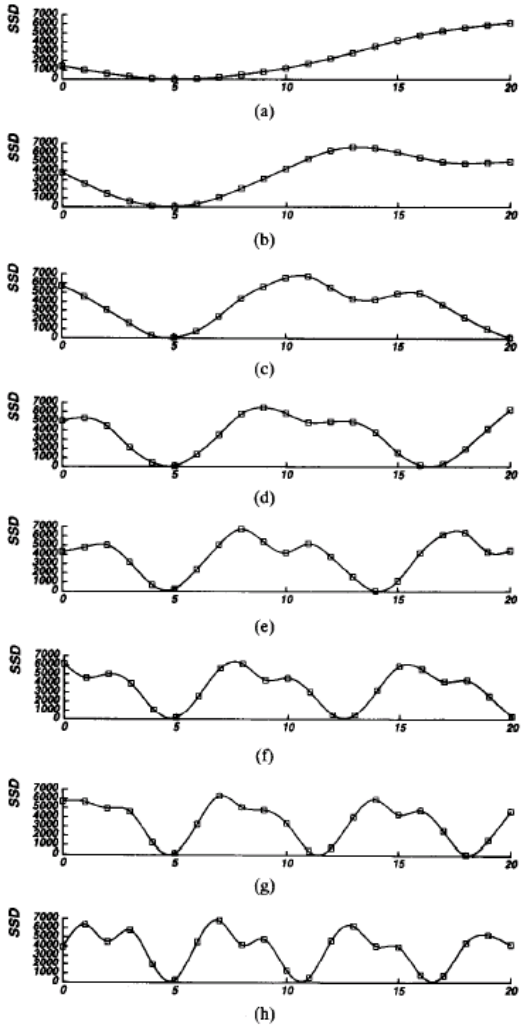


Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

Use the sum of SSD scores to rank matches

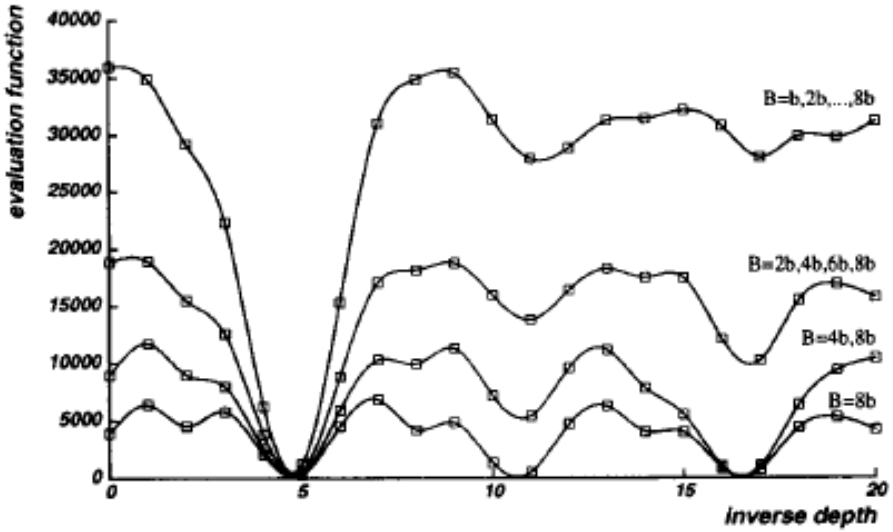


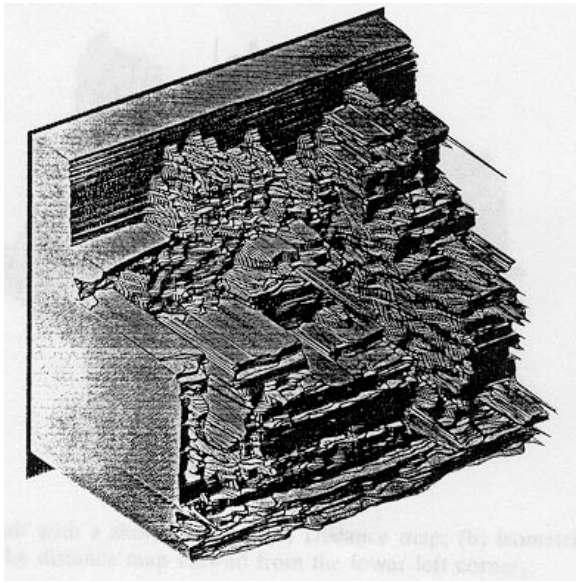
Fig. 7. Combining multiple baseline stereo pairs.

Multiple-baseline stereo results

I1

I2

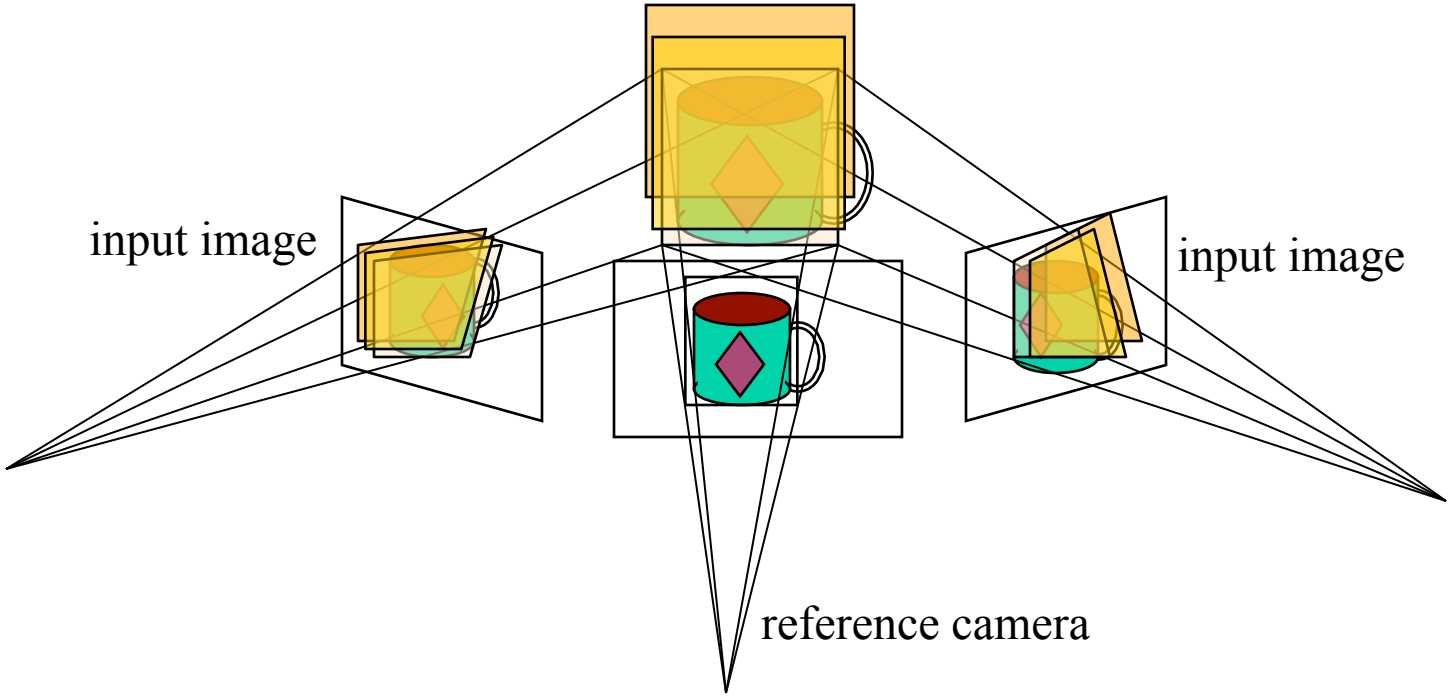
I10



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

Plane Sweep Stereo

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

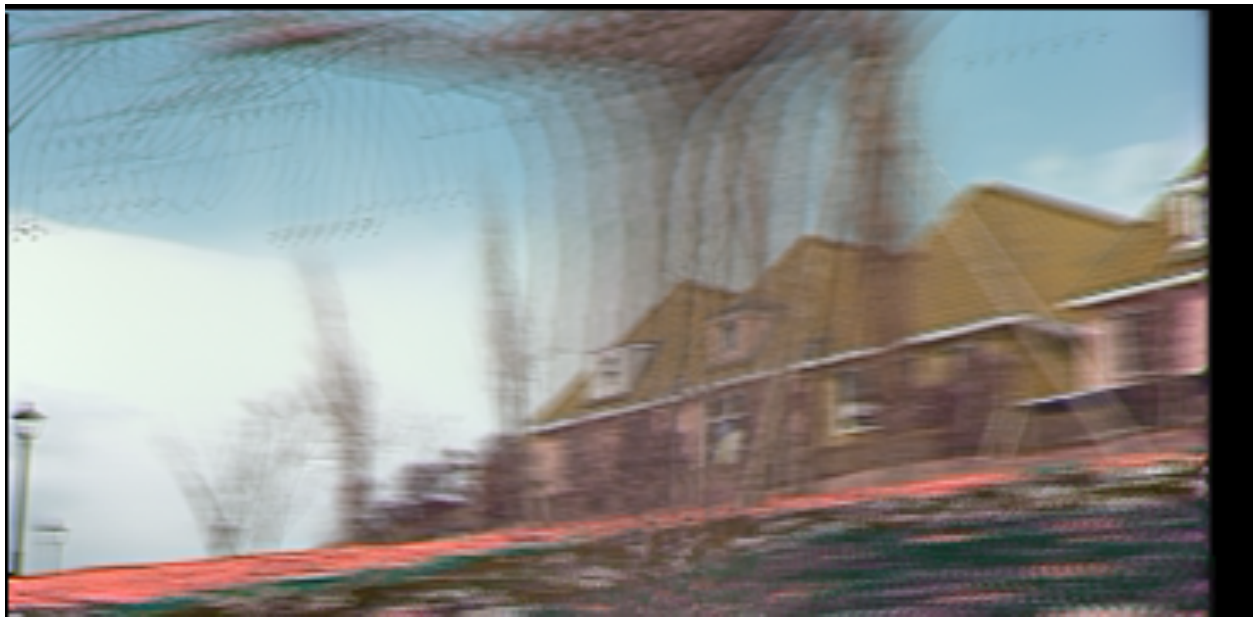


Each plane defines a homography warping each input image into the reference view

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

Plane Sweep Stereo

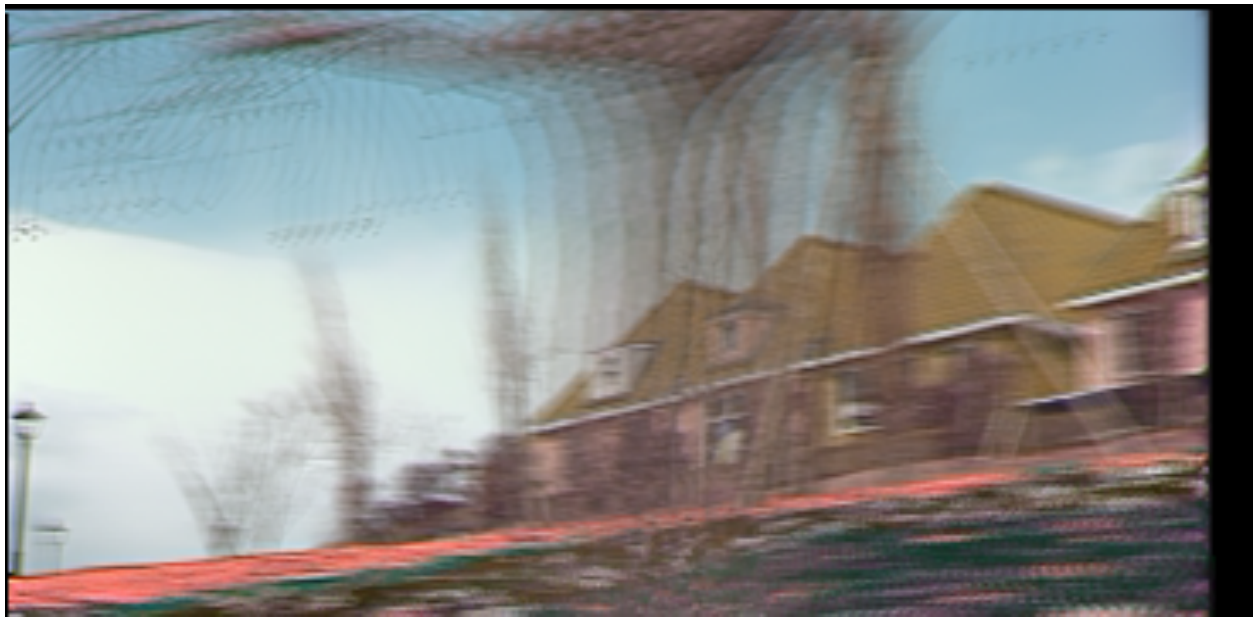
- For each depth plane
  - For each pixel in the composite image stack, compute the variance



- 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 stack, compute the variance



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

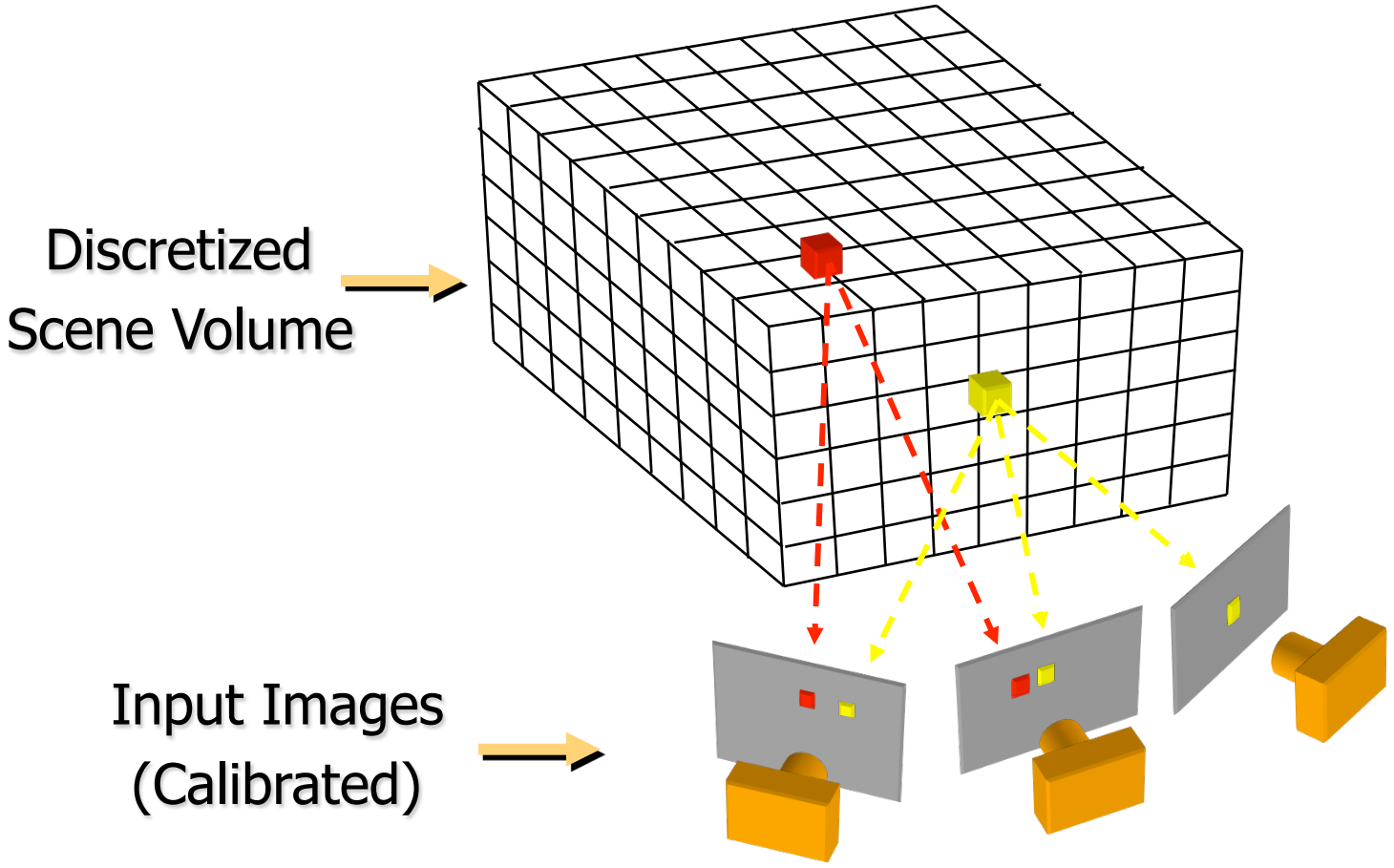
Can be accelerated using graphics hardware

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 depends on the reference view
- We can use a voxel volume to get a view-independent representation

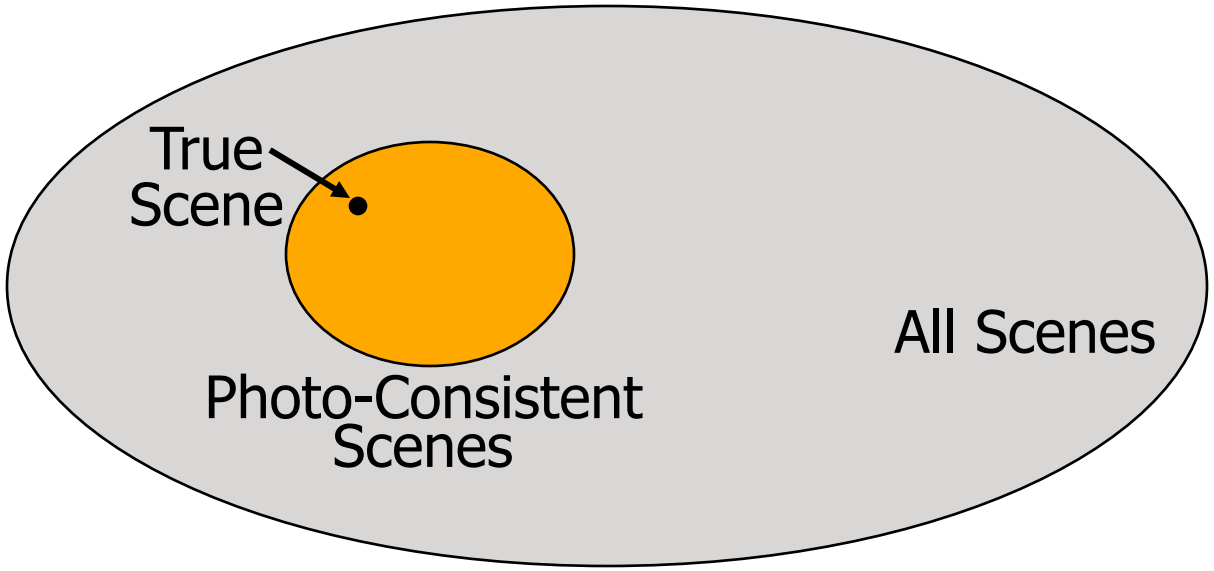
Volumetric Stereo / Voxel Coloring



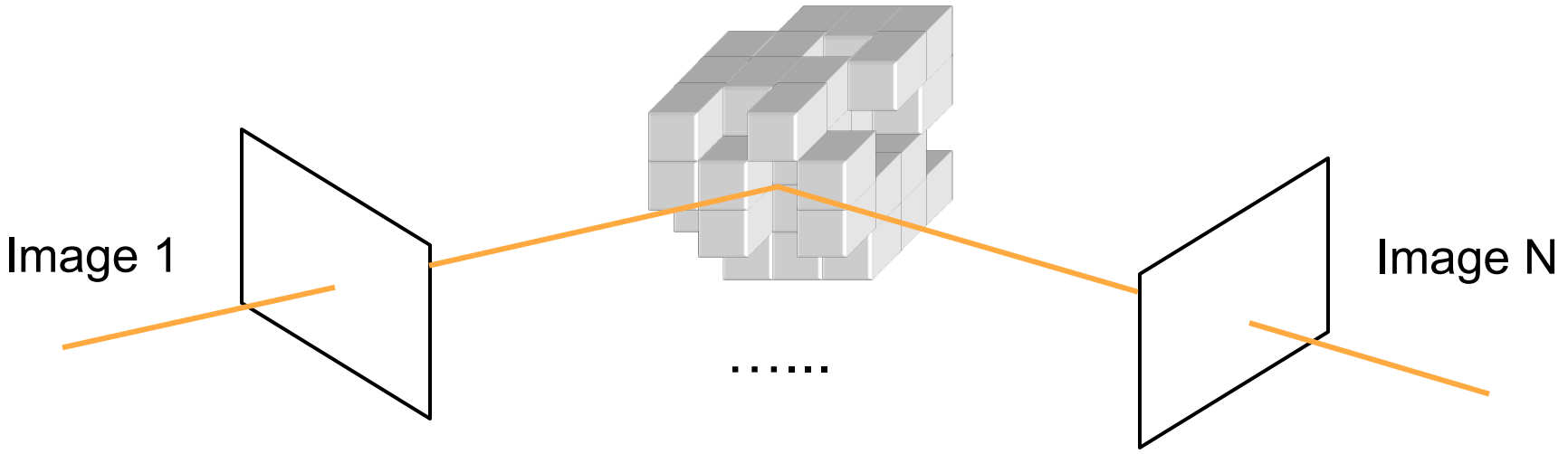
Goal: **Assign RGB values to voxels in V**
photo-consistent **with images**

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

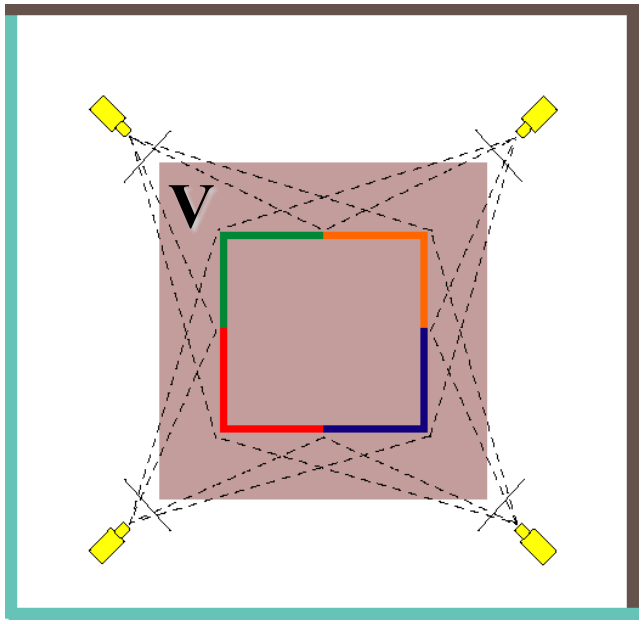


Space Carving Algorithm

- Initialize to a volume V containing the true scene
- Choose a voxel on the outside of the volume
- 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

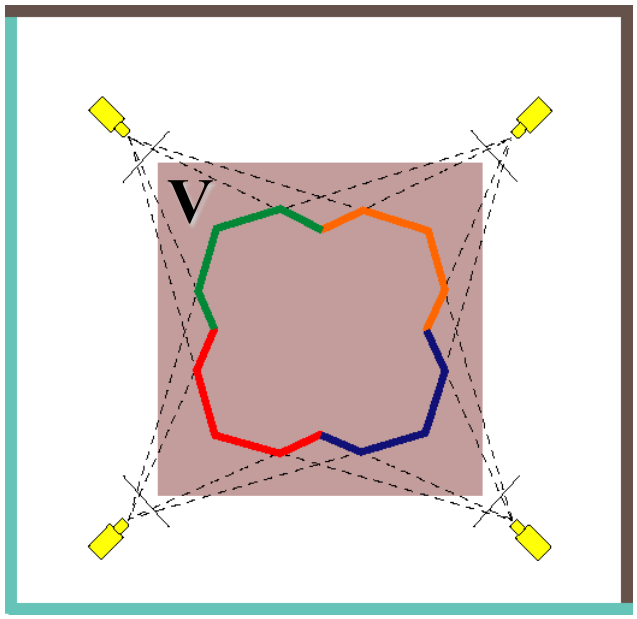


Photo Hull

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

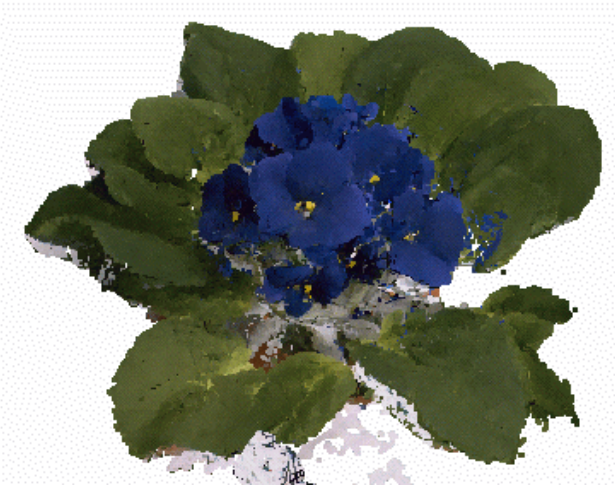
- It is a photo-consistent scene reconstruction
- Tightest possible bound on the true scene

Source: S. Seitz

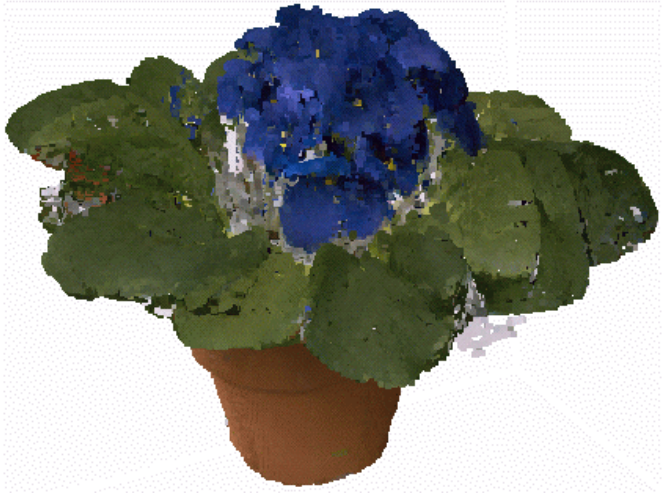
Space Carving Results: African Violet

Input Image (1 of 45)

Reconstruction



Reconstruction



Reconstruction

Source: S. Seitz

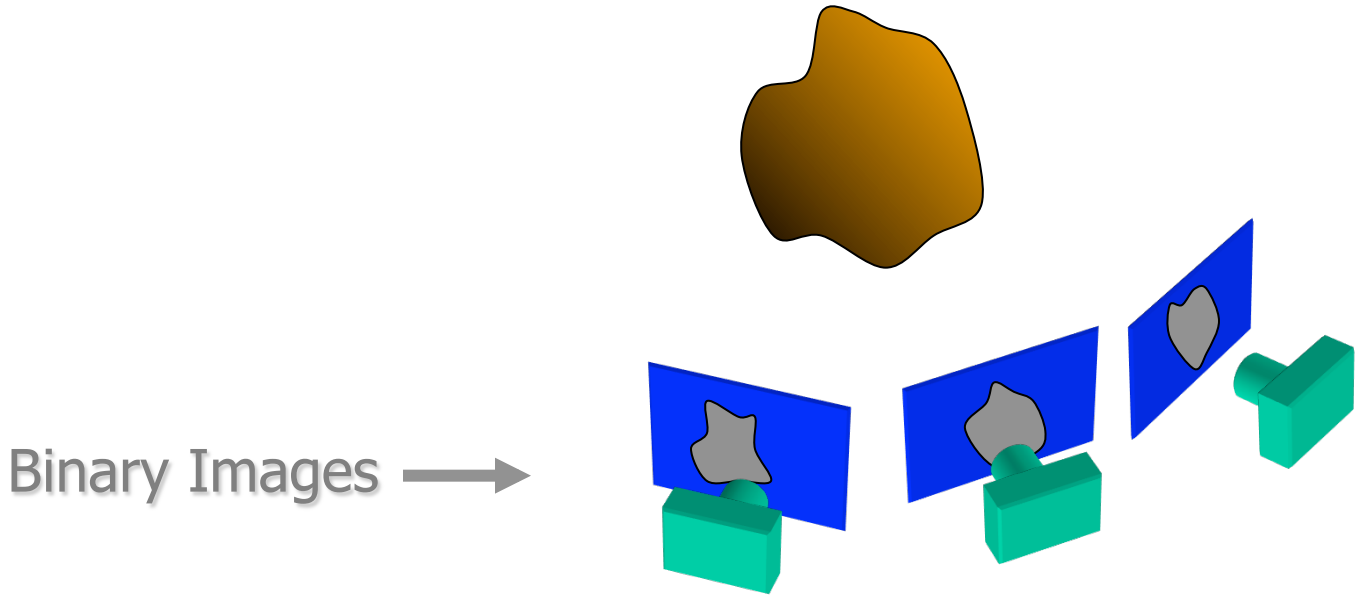
Space Carving Results: Hand

Input Image
(1 of 100)

Views of Reconstruction

Reconstruction from Silhouettes

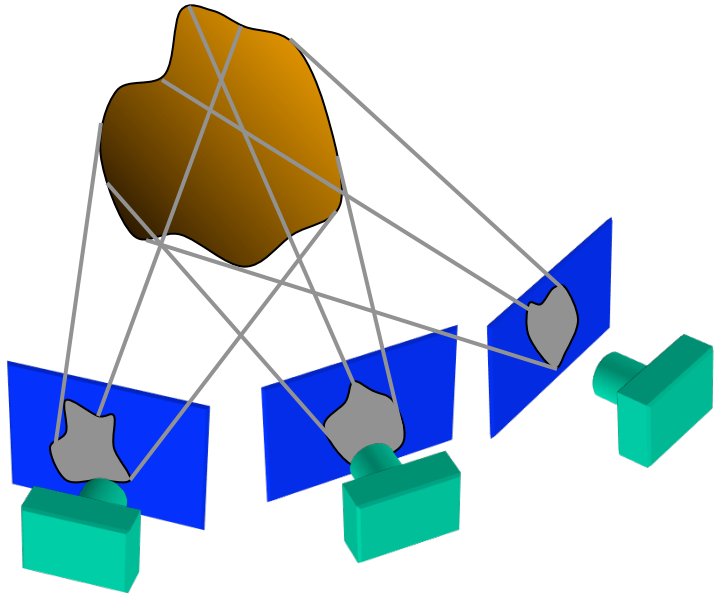
- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views



Reconstruction from Silhouettes

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

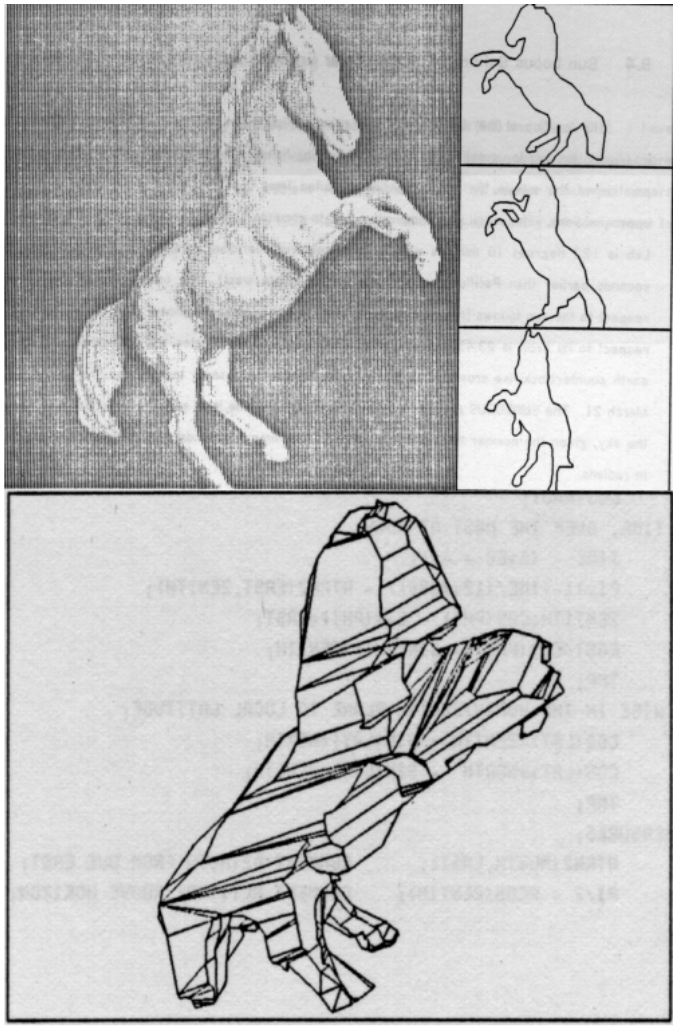
Binary Images →



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

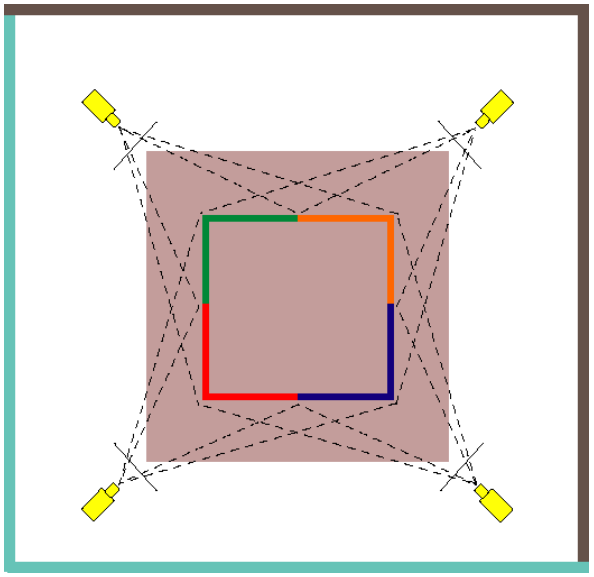
- *Backproject* each silhouette
- Intersect backprojected volumes

Volume intersection



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

Photo-consistency vs. silhouette-consistency



True Scene

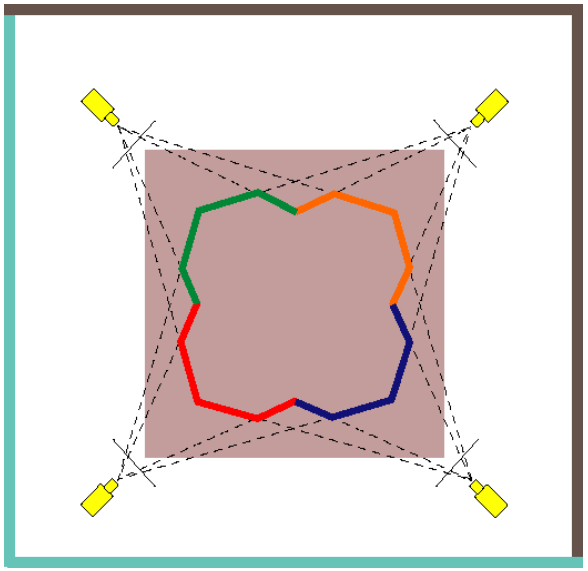
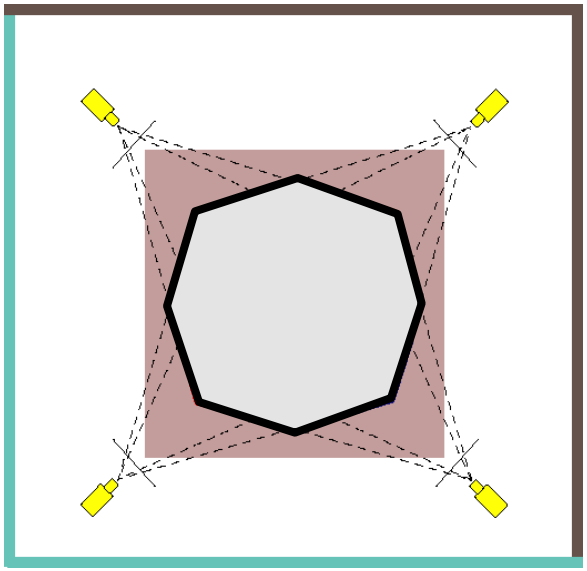


Photo Hull



Visual Hull

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 (photo-consistent parts of visual hull)
3. Carve the visual hull to optimize photo-consistency keeping the rims fixed

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

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.

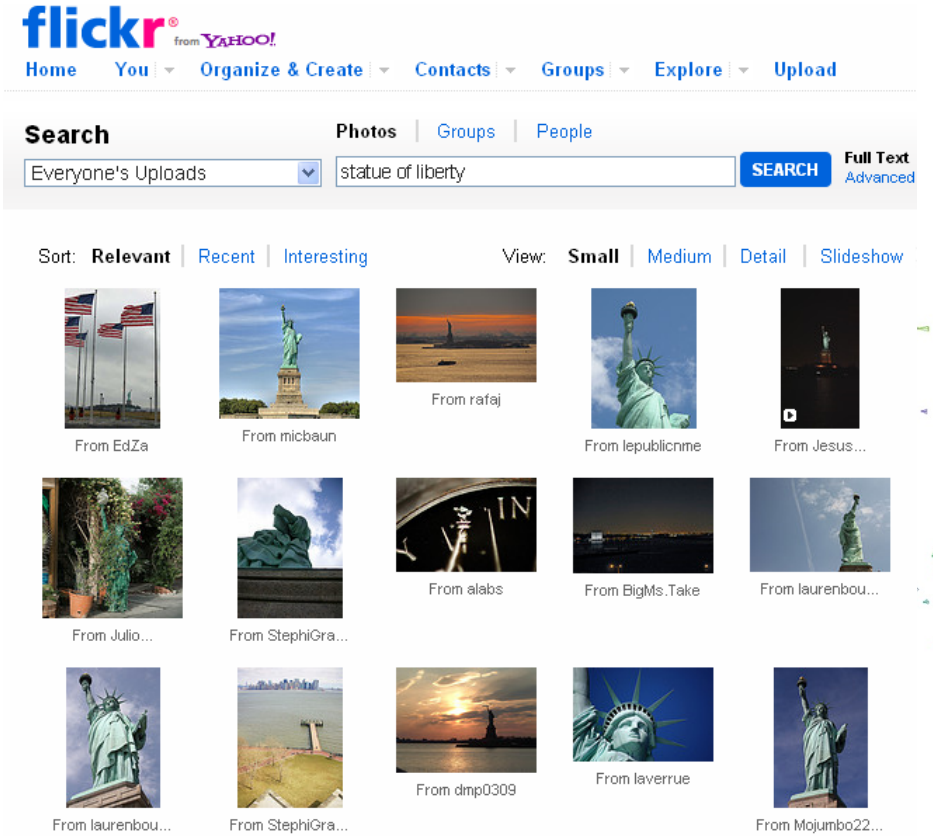
From feature matching to dense stereo

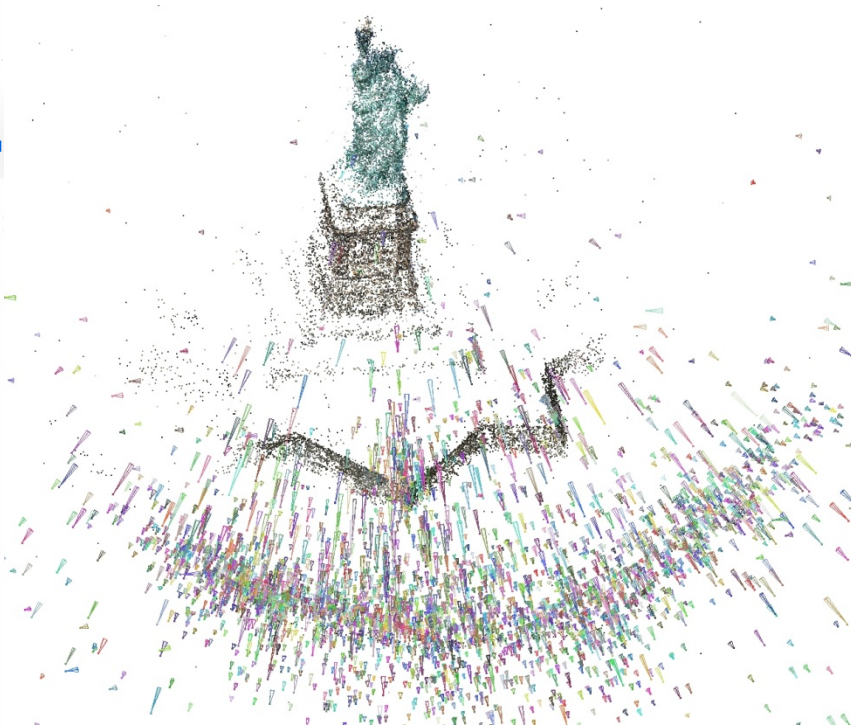
http://www.cs.washington.edu/homes/furukawa/gallery/

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





Towards Internet-Scale Multi-View Stereo

YouTube video, high-quality video

Yasutaka Furukawa, Brian Curless, Steven M. Seitz and Richard Szeliski, Towards Internet-scale Multi-view Stereo,CVPR 2010.

Fast stereo for Internet photo collections

- Start with a cluster of registered views
- Obtain a depth map for every view using plane sweeping stereo with normalized cross-correlation

Frahm et al., "Building Rome on a Cloudless Day," ECCV 2010.

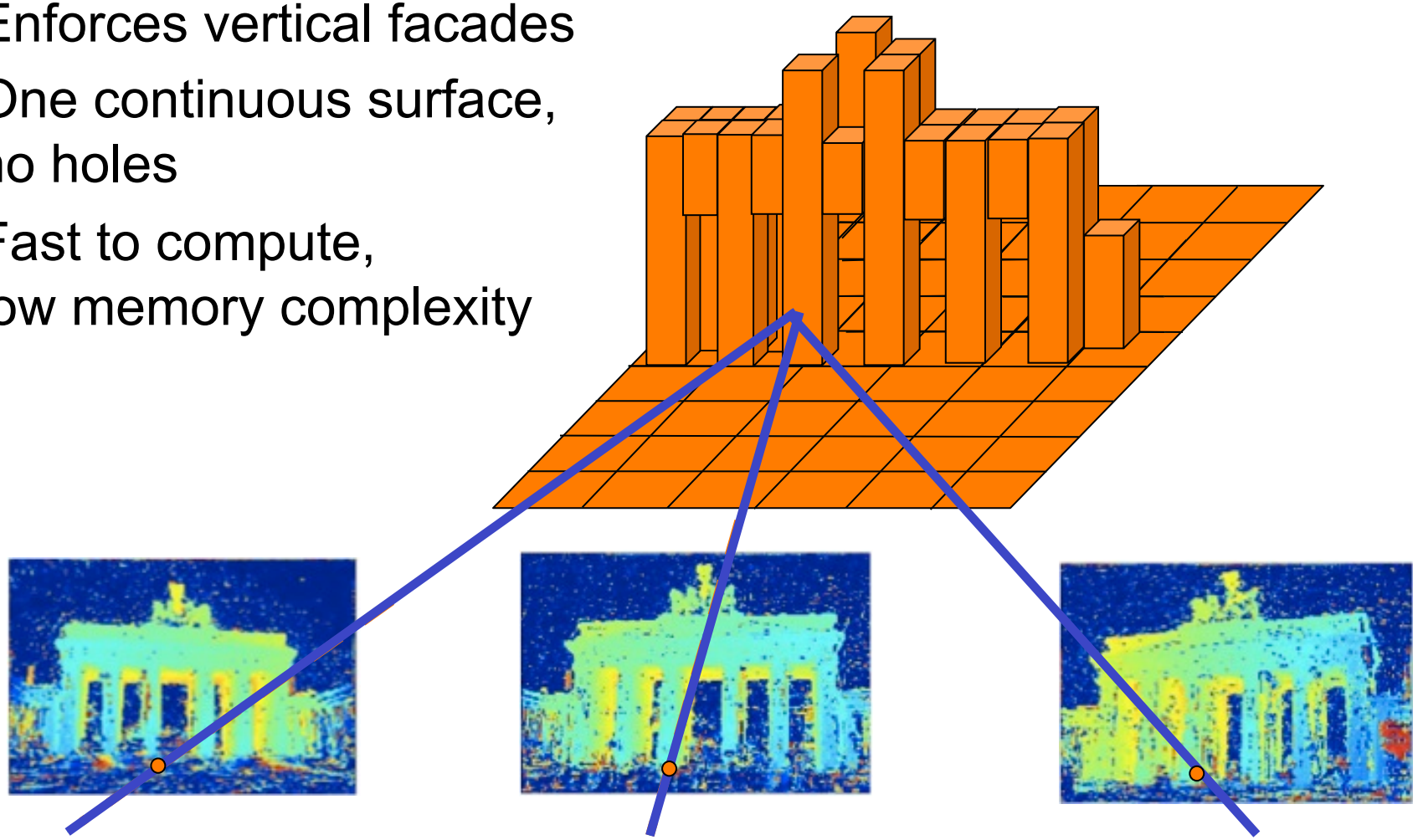
Plane sweeping stereo

- Need to register individual depth maps into a single 3D model
- Problem: depth maps are very noisy

Frahm et al., "Building Rome on a Cloudless Day," ECCV 2010.

Robust stereo fusion using a heightmap

- Enforces vertical facades
- One continuous surface, no holes
- Fast to compute, low memory complexity



David Gallup, Marc Pollefeys, Jan-Michael Frahm, “3D Reconstruction using an n-Layer Heightmap”, DAGM 2010

Results

YouTube Video

Frahm et al., "Building Rome on a Cloudless Day," ECCV 2010.

Kinect: Structured infrared light

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

KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera*

Shahram Izadi[1], *David Kim*[1,3], *Otmar Hilliges*[1], *David Molyneaux*[1,4], *Richard Newcombe*[2], *Pushmeet Kohli*[1], *Jamie Shotton*[1], *Steve Hodges*[1], *Dustin Freeman*[1,5], *Andrew Davison*[2], *Andrew Fitzgibbon*[1]

[1]Microsoft Research Cambridge, UK [2]Imperial College London, UK
[3]Newcastle University, UK [4]Lancaster University, UK [5]University of Toronto, Canada

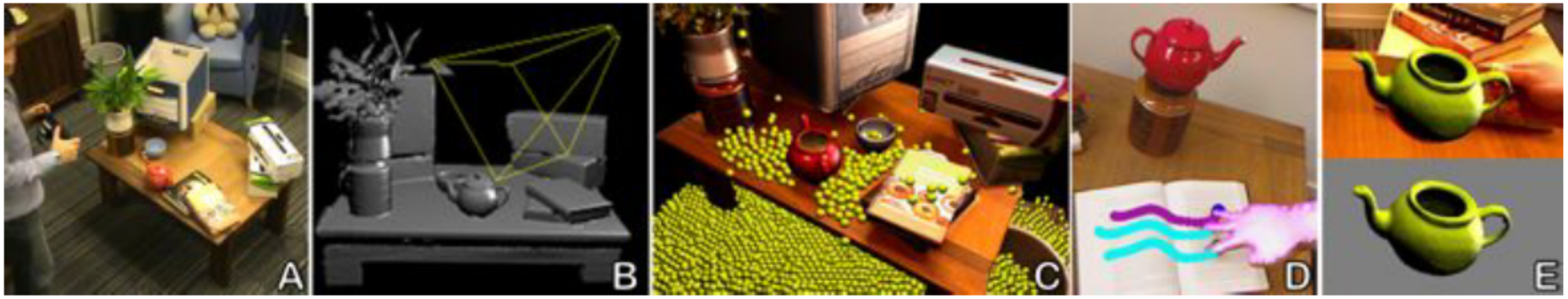


Figure 1: KinectFusion enables real-time detailed 3D reconstructions of indoor scenes using only the depth data from a standard Kinect camera. A) user points Kinect at coffee table scene. B) Phong shaded reconstructed 3D model (the wireframe frustum shows current tracked 3D pose of Kinect). C) 3D model texture mapped using Kinect RGB data with real-time particles simulated on the 3D model as reconstruction occurs. D) Multi-touch interactions performed on any reconstructed surface. E) Real-time segmentation and 3D tracking of a physical object.

Paper link (ACM Symposium on User Interface Software and Technology, October 2011)

YouTube Video