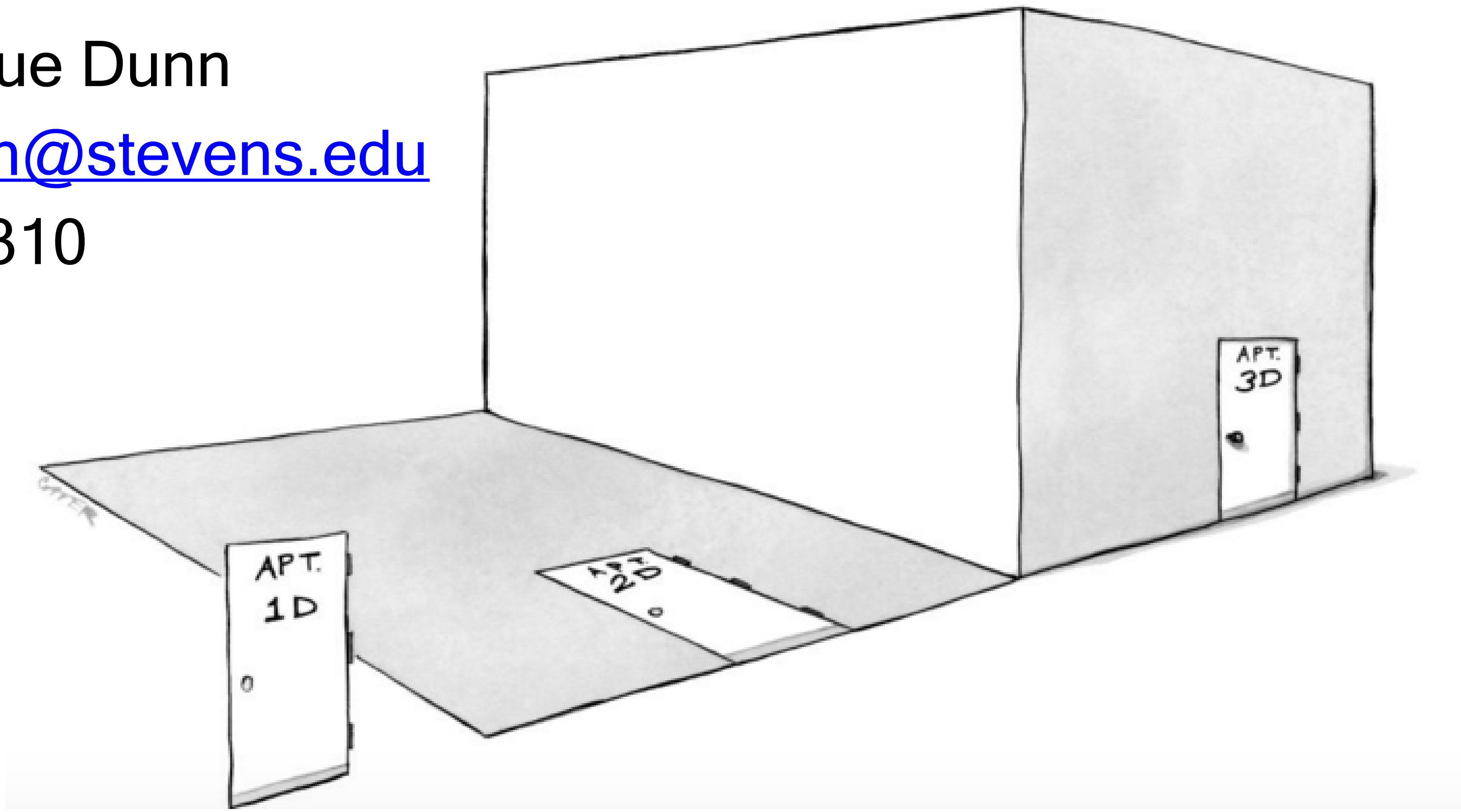
CS 532: 3D Computer Vision
Lecture 8

Enrique Dunn

edunn@stevens.edu

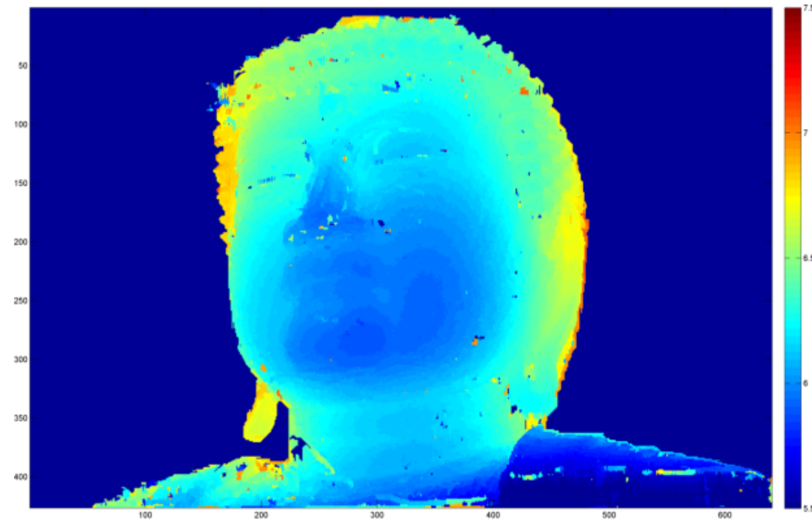
Lieb 310



Joint Depth Estimation and View Selection

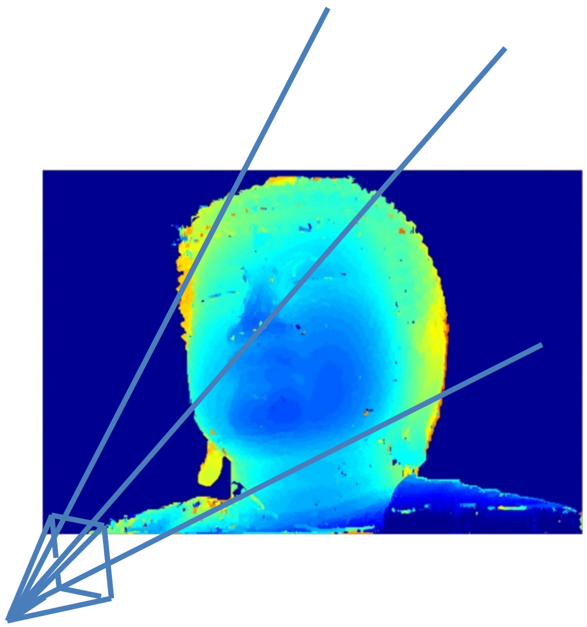
Depthmap Estimation

Image



Depthmap

Depthmap Estimation

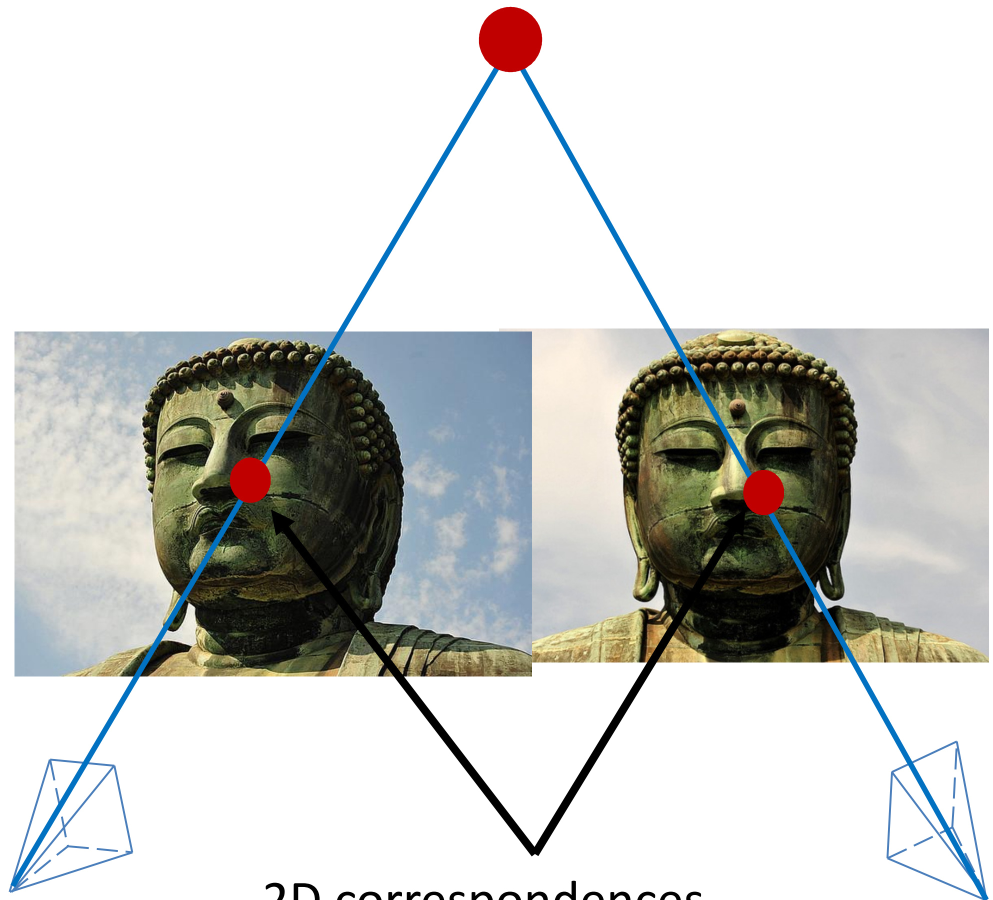


Projection into 3D space

3D point cloud

Background

Reference image

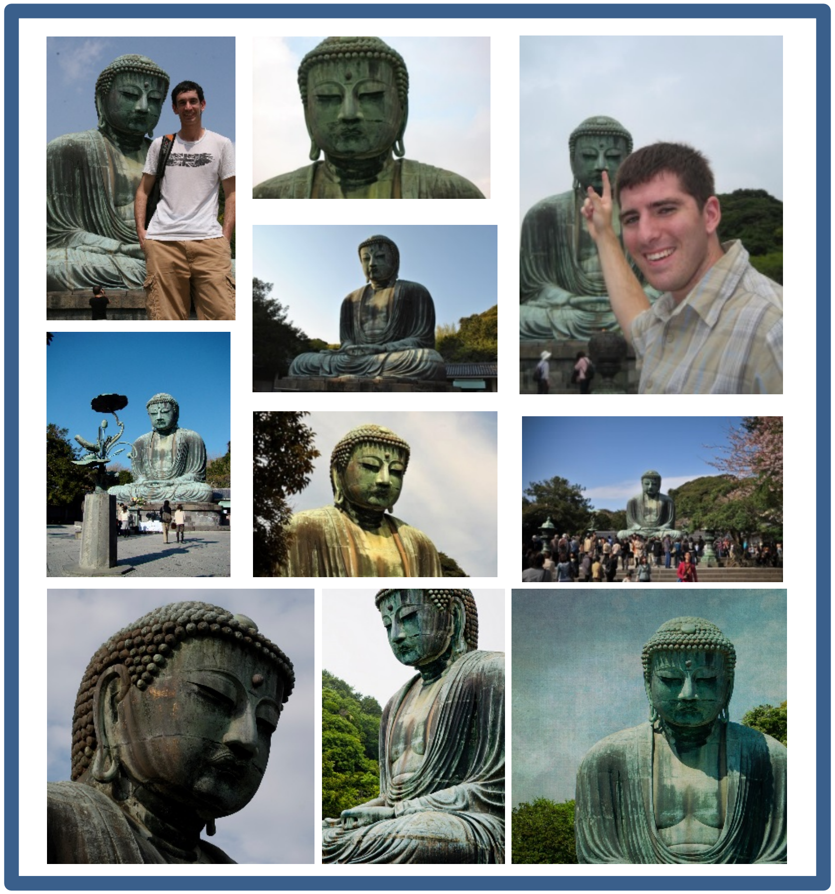


Source image

2D correspondences

Challenges

Reference image



211 Source images (only 10 are shown)

Challenges

Reference image

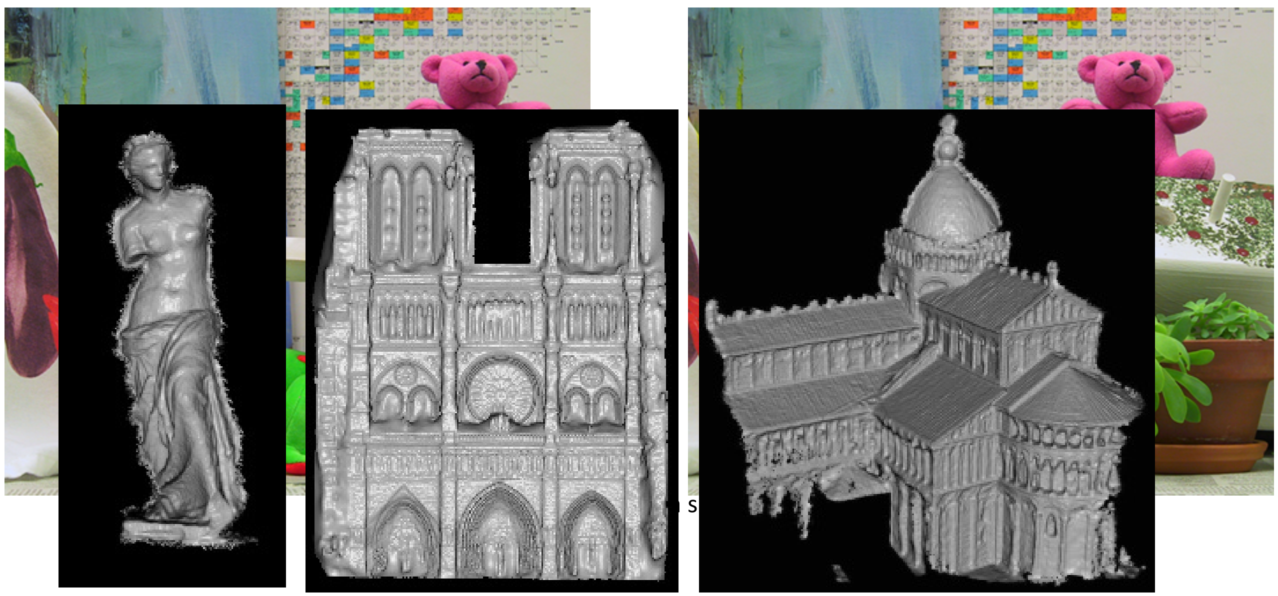
Illumination variation

Scene resolution difference

Occlusion

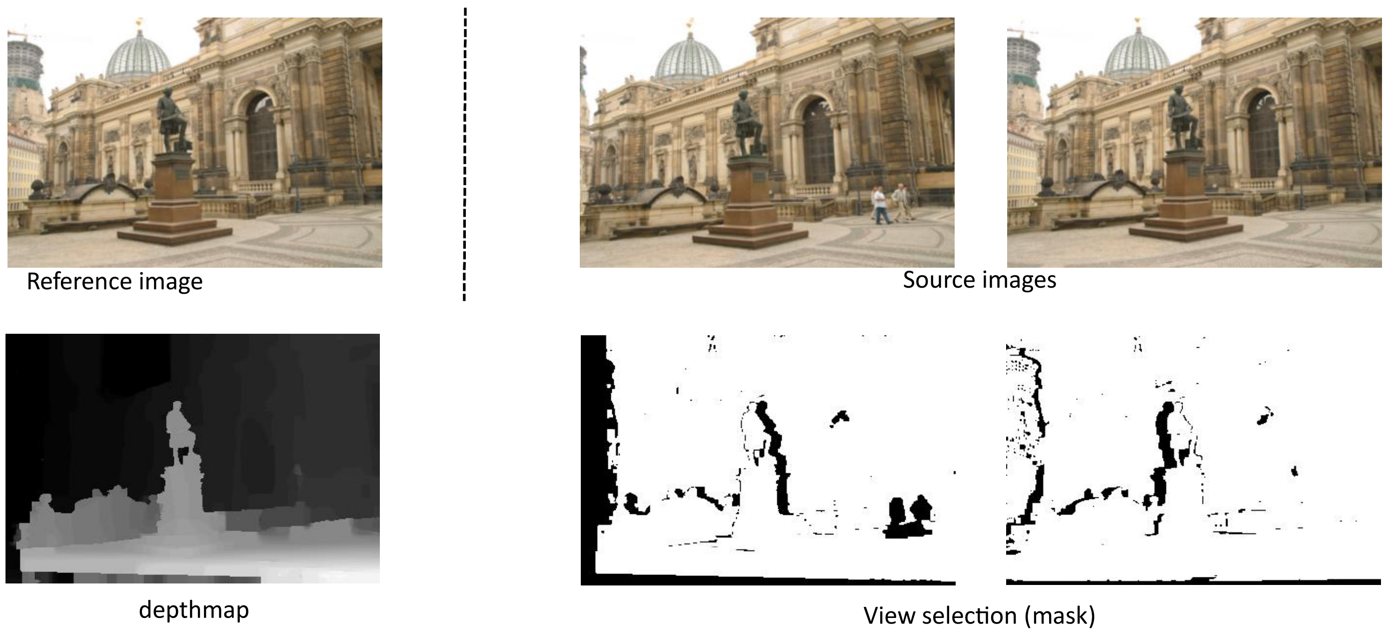
Prior Works

- Use images taken under a controlled environment
  - Yoon et al. PAMI 2006, Hirschmüller et al. CVPR 2005, Bleyer et al. BMVC 2012, Lu et al. CVPR 2013, Wang et al. CVPR 2014, …
- Reconstruction from Internet photo collections
  - Goesele et al. ICCV 2007



Prior Works

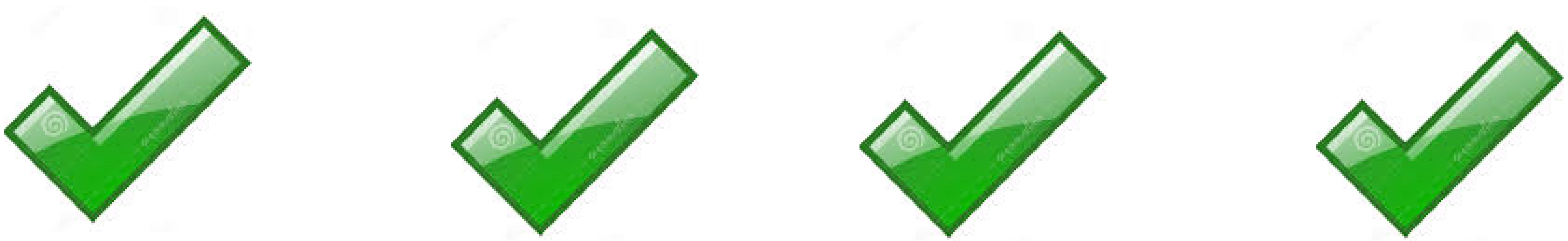
- Use images taken under a controlled environment
  - Yoon et al. PAMI 2006, Hirschmüller et al. CVPR 2005, Bleyer et al. BMVC 2012, Lu et al. CVPR 2013, Wang et al. CVPR 2014, …
- Reconstruction from Internet photo collections
  - Goesele et al. ICCV 2007
- Pixel level view selection and depth estimation
  - Strecha et al., CVPR 2006



Pixel Level Image Selection

Reference image

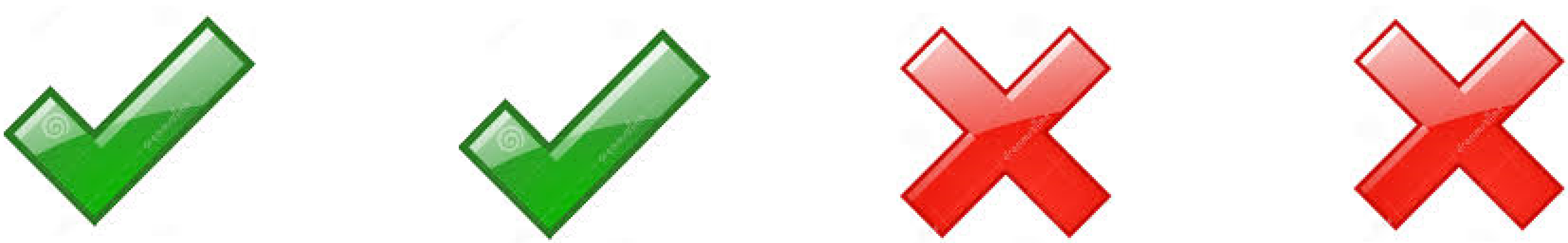
Source images



Pixel Level Image Selection

Reference image

Source images



Pixel Level Image Selection

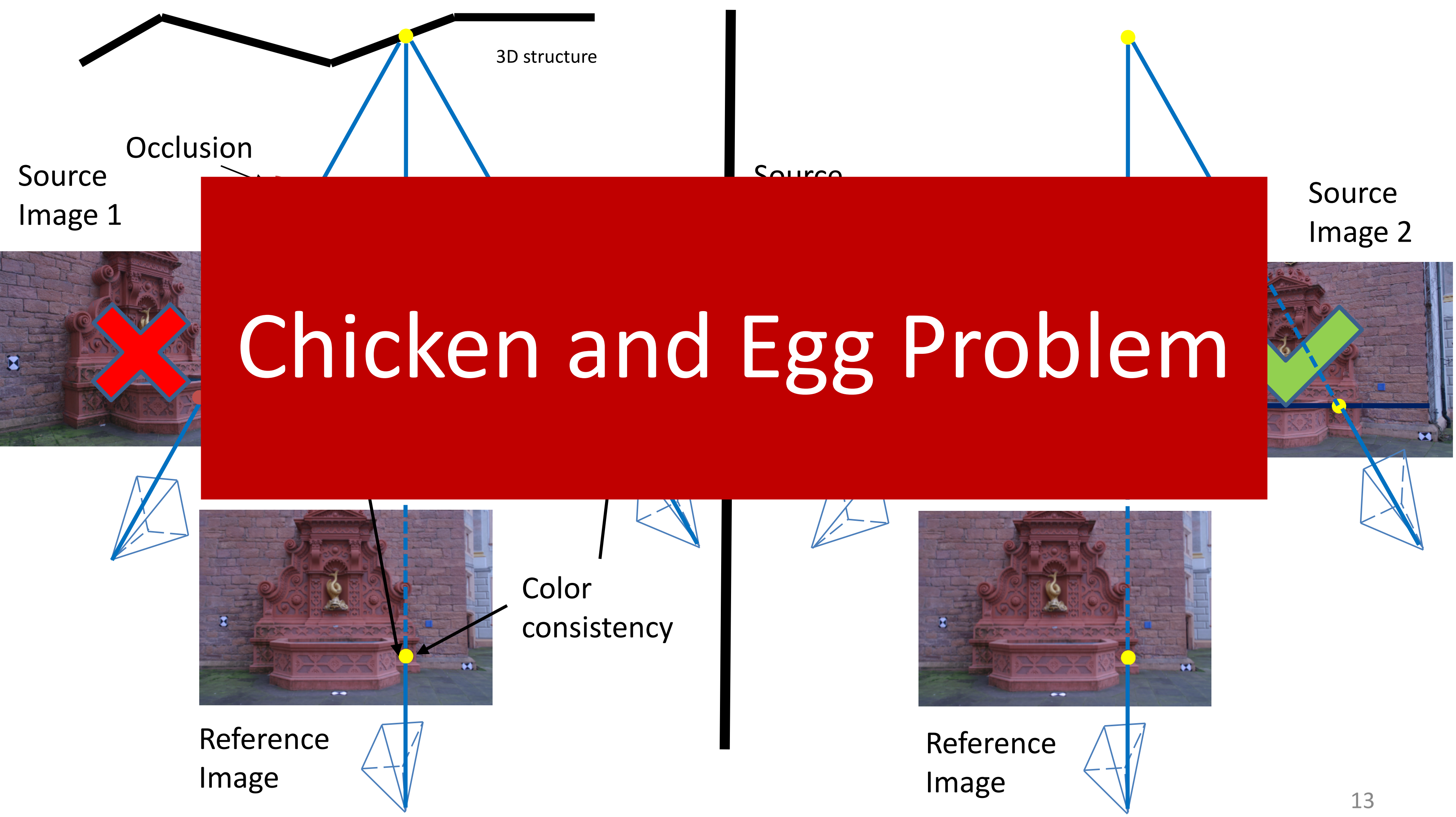
Reference image

Source images

Image Selection vs Depth Estimation

Assume known 3D structure

Assume known image selection



Our Method Overview

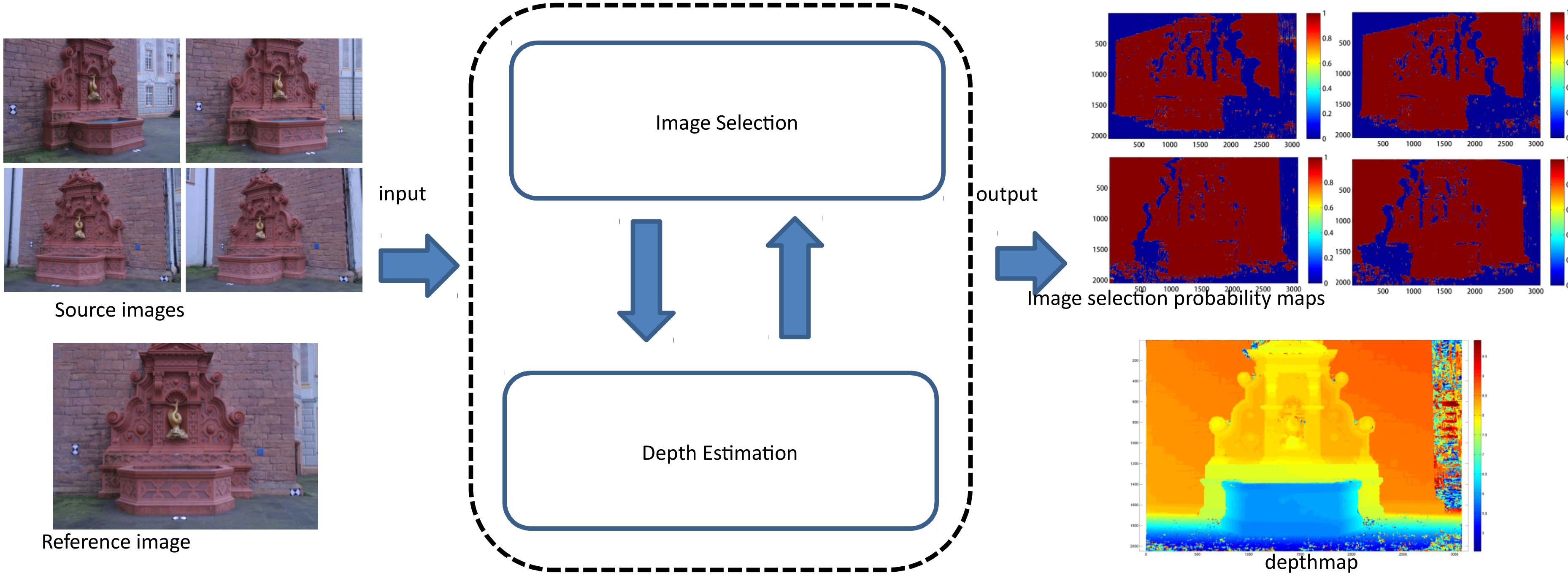


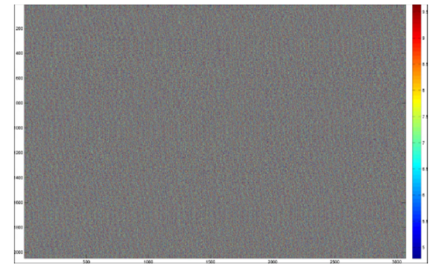
Image Selection–Smoothness Term

Reference image

Source image

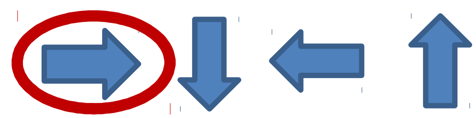
Our Method

- Random depth initialization



Random depth

- Depth propagations
  - EM style depth estimation and view selection
  - Propagations in four directions



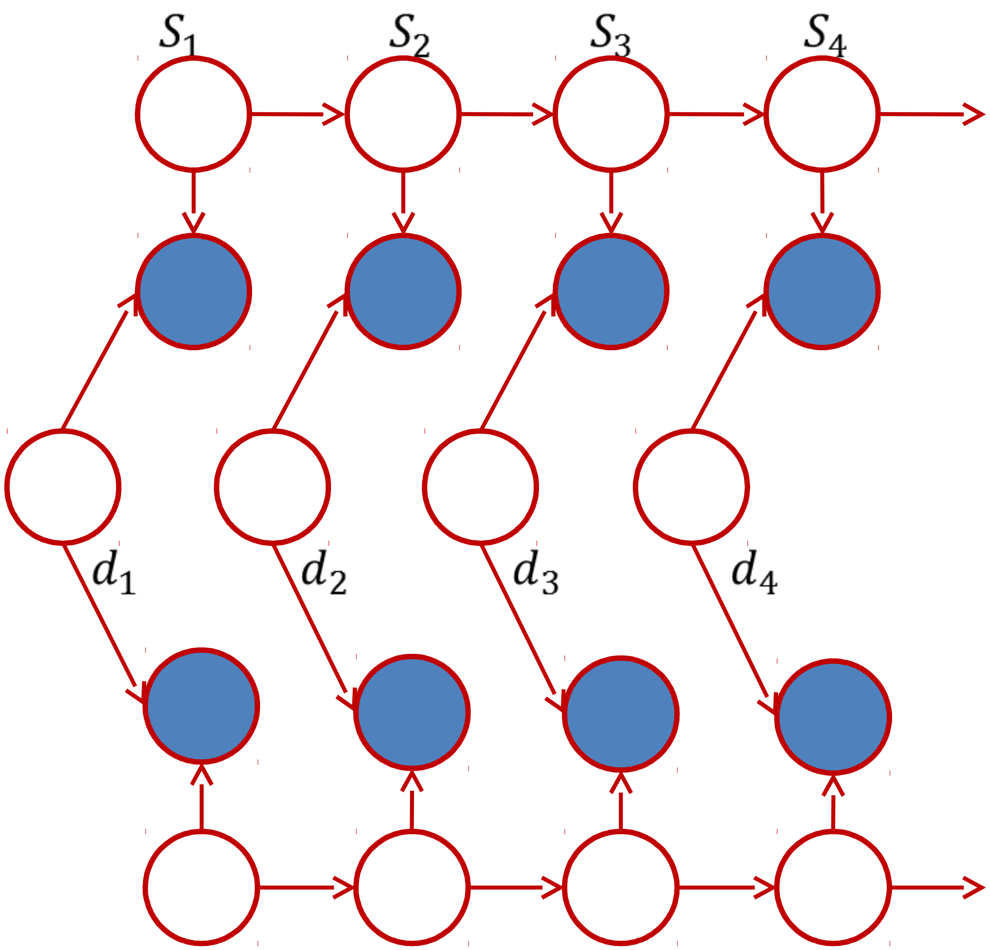
  - Each row/column is independent

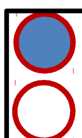
Our Method

Image Selection

Source Image

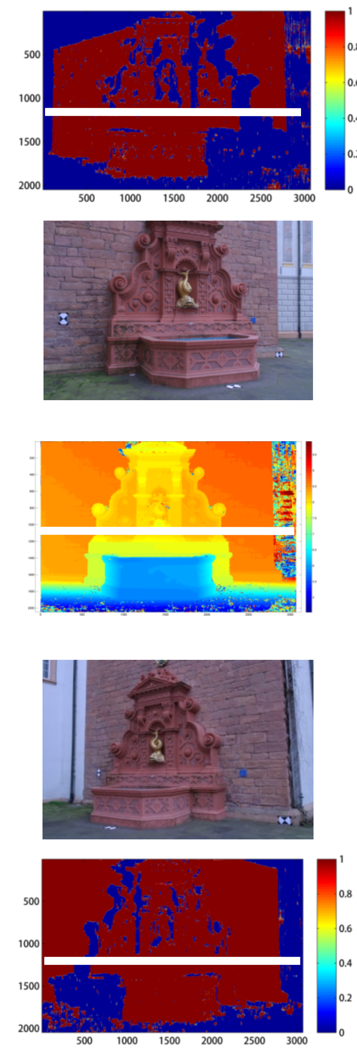
Depth





-- Observation

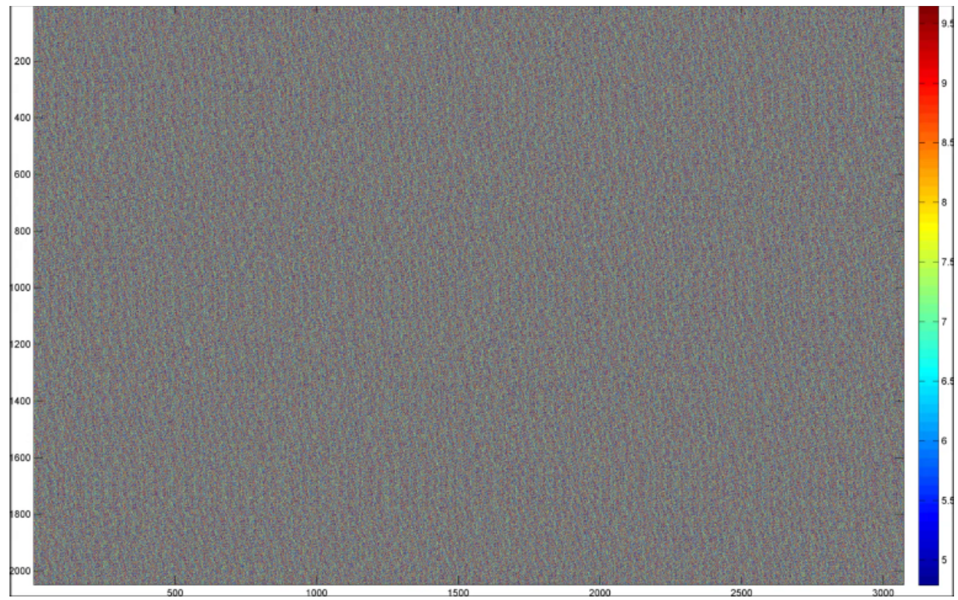
-- Unknown variable



Our Method

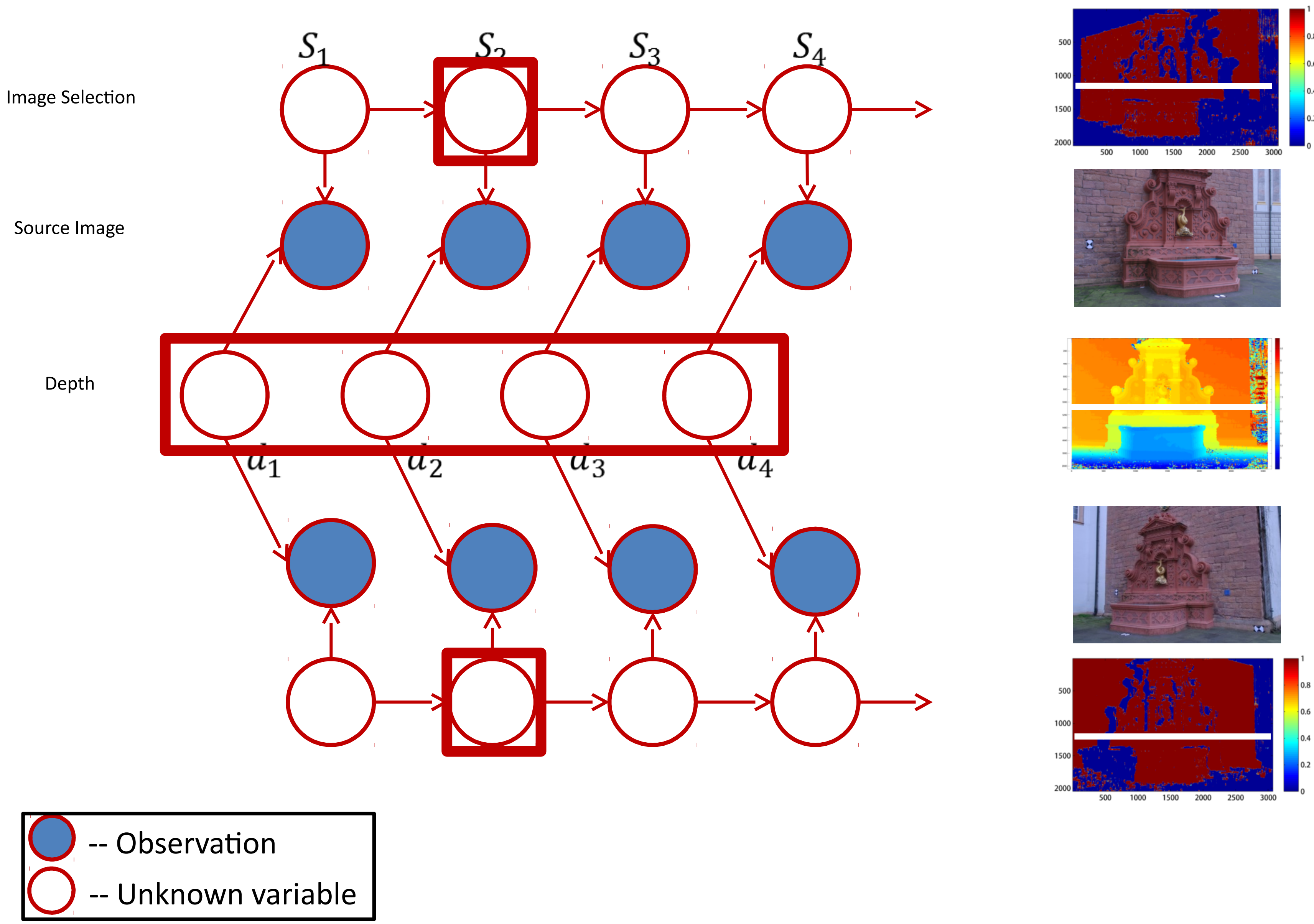
- Random depthmap initialization

Reference image



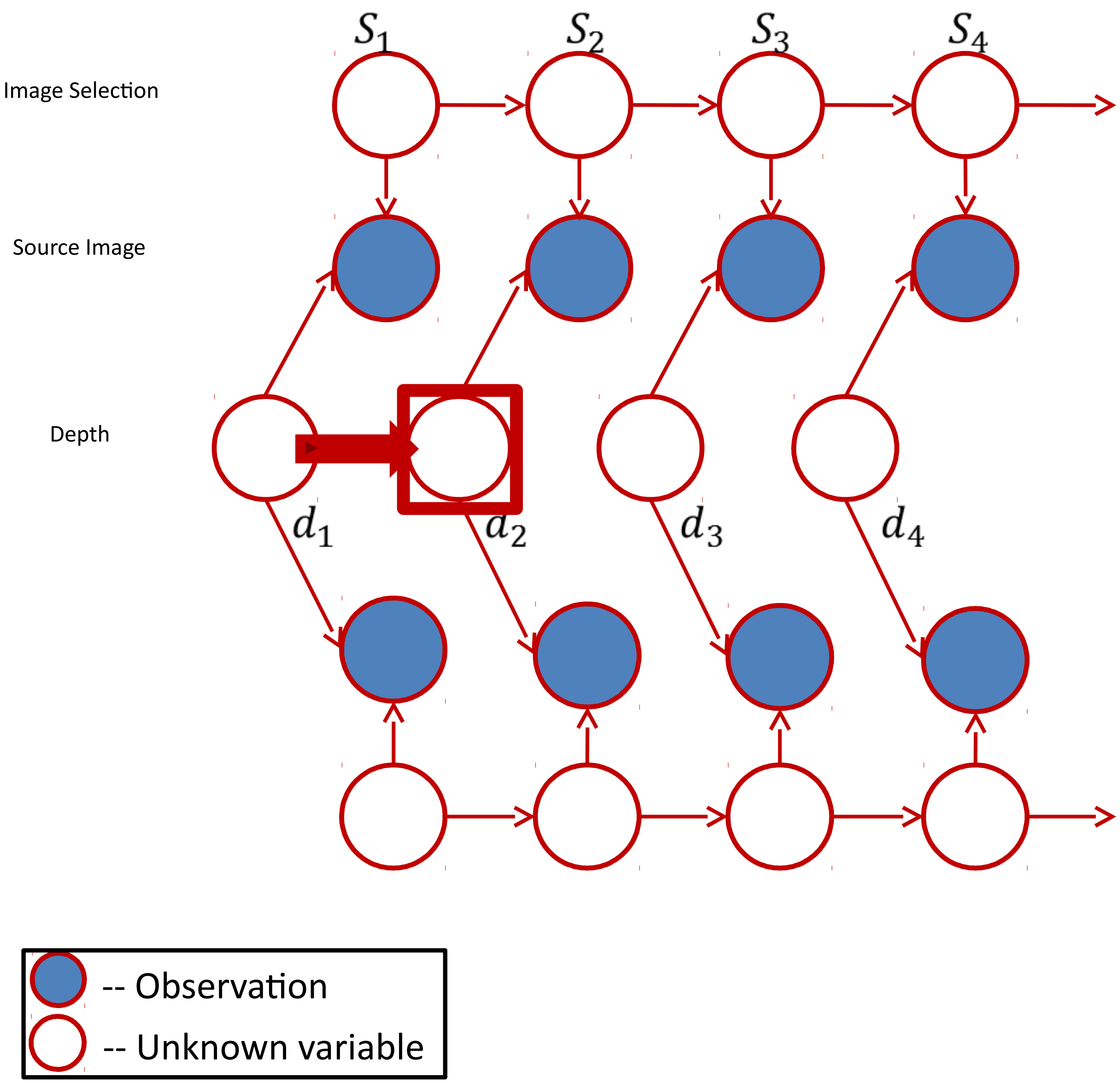
Random depthmap

Our Method

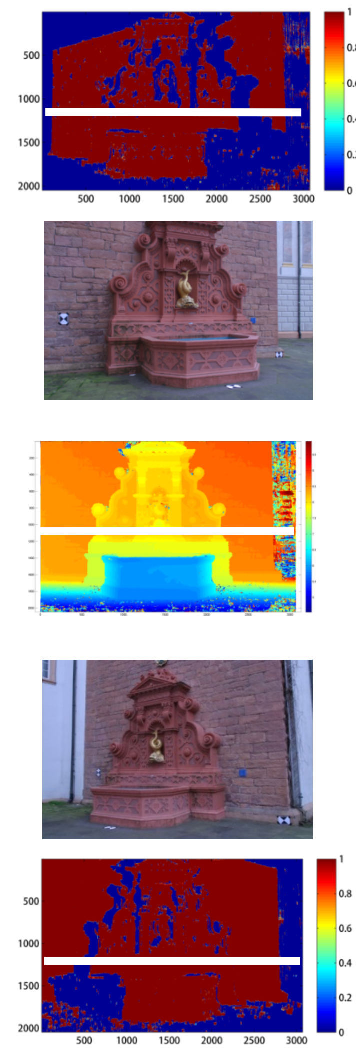


S_1
S_2
S_3
S_4
Image Selection
Source Image
Depth
d_1
d_2
d_3
d_4
-- Observation
-- Unknown variable

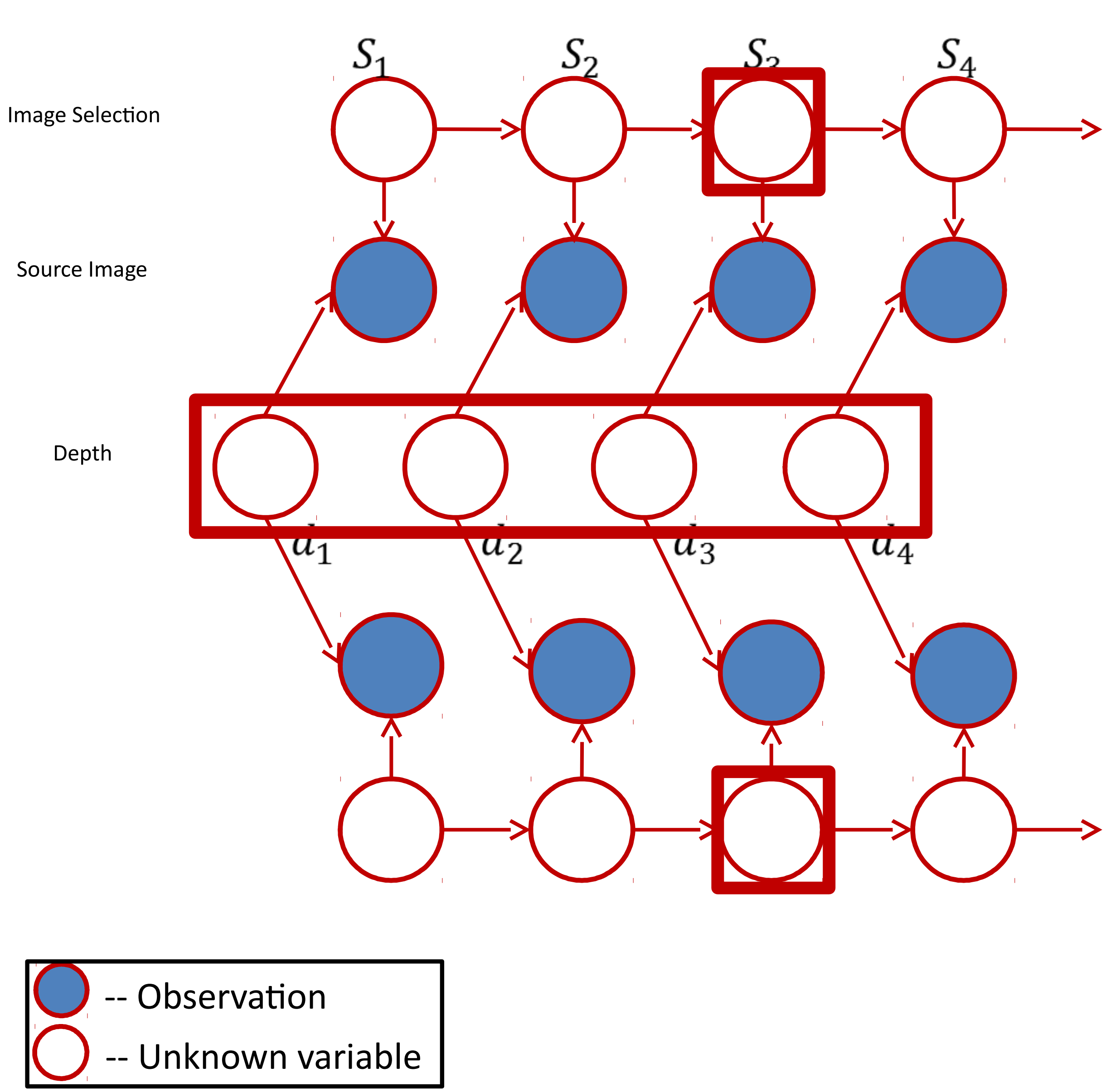
Our Method

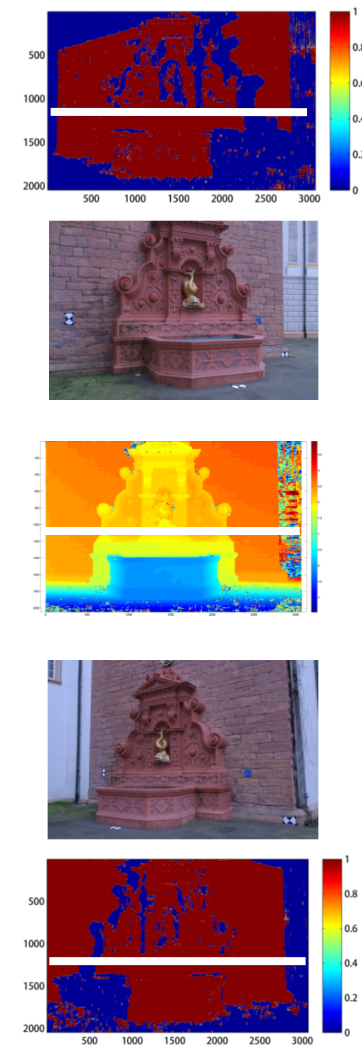


S_1
S_2
S_3
S_4
Image Selection
Source Image
Depth
d_1
d_2
d_3
d_4
-- Observation
-- Unknown variable



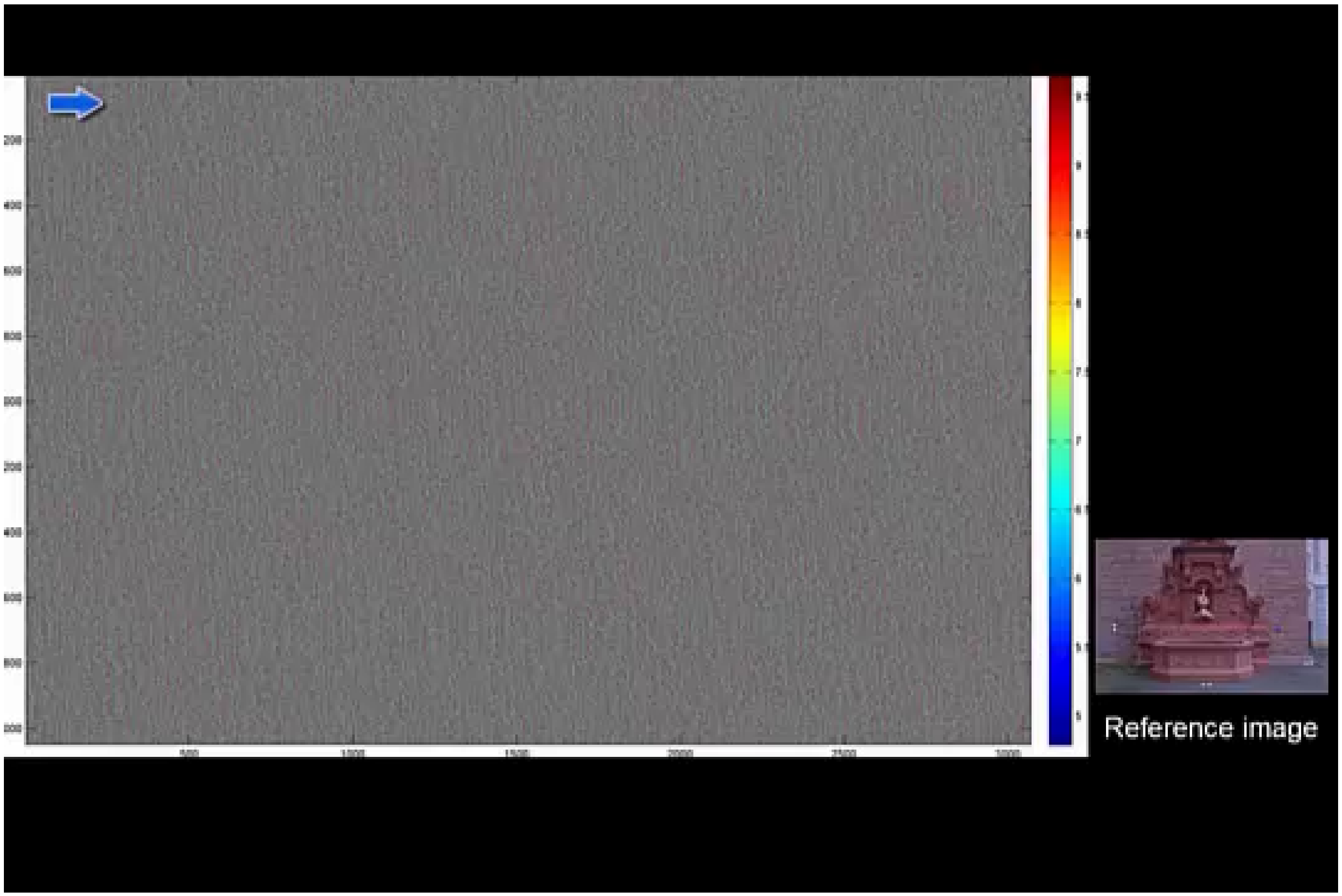
Our Method


S_1
S_2
S_3
S_4
Image Selection
Source Image
Depth
d_1
d_2
d_3
d_4
-- Observation
-- Unknown variable



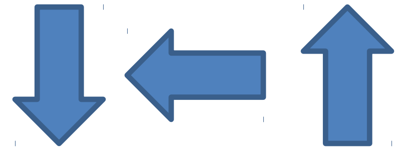
Our method

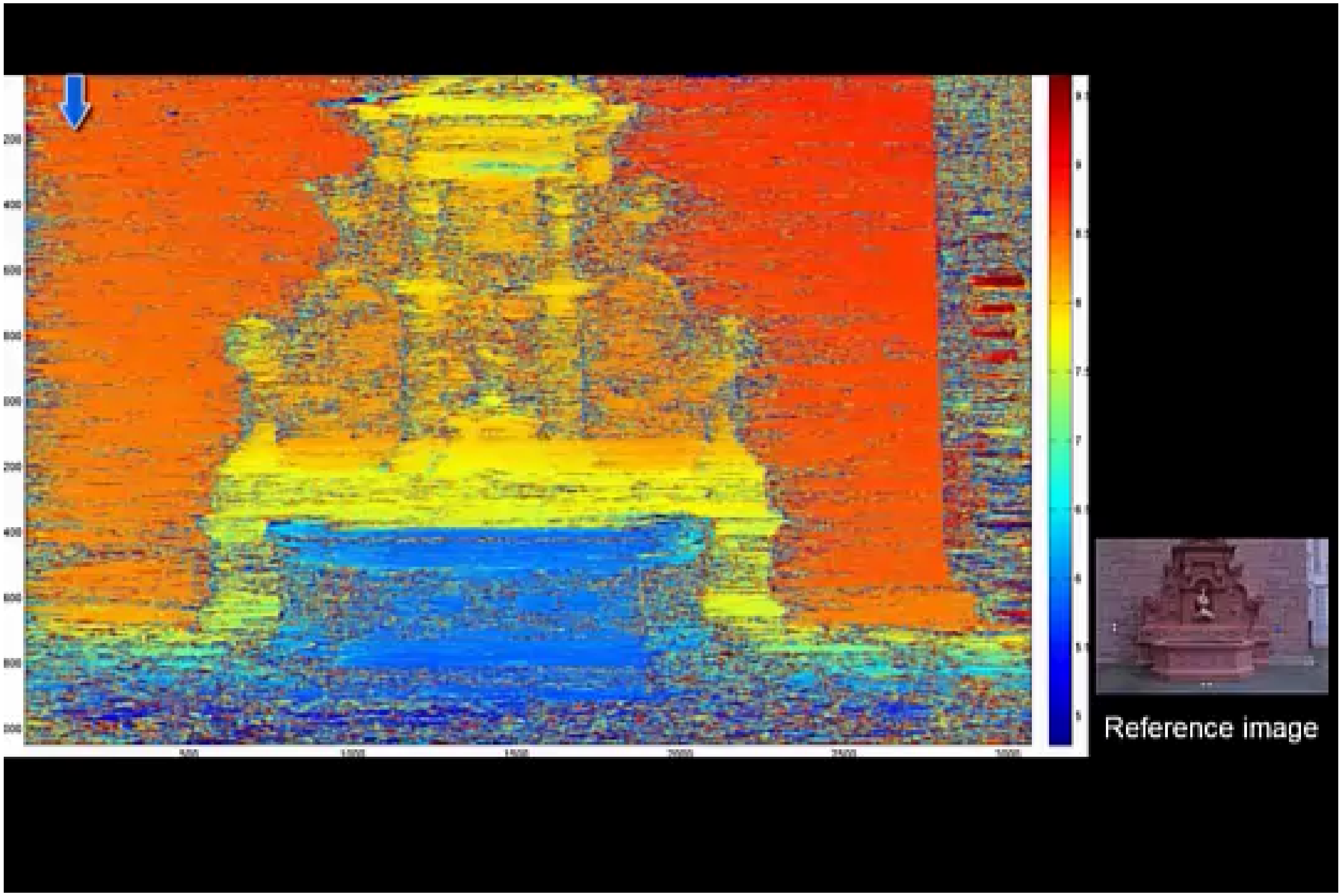
- Left to right propagation



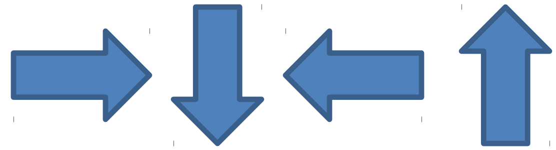
Our method

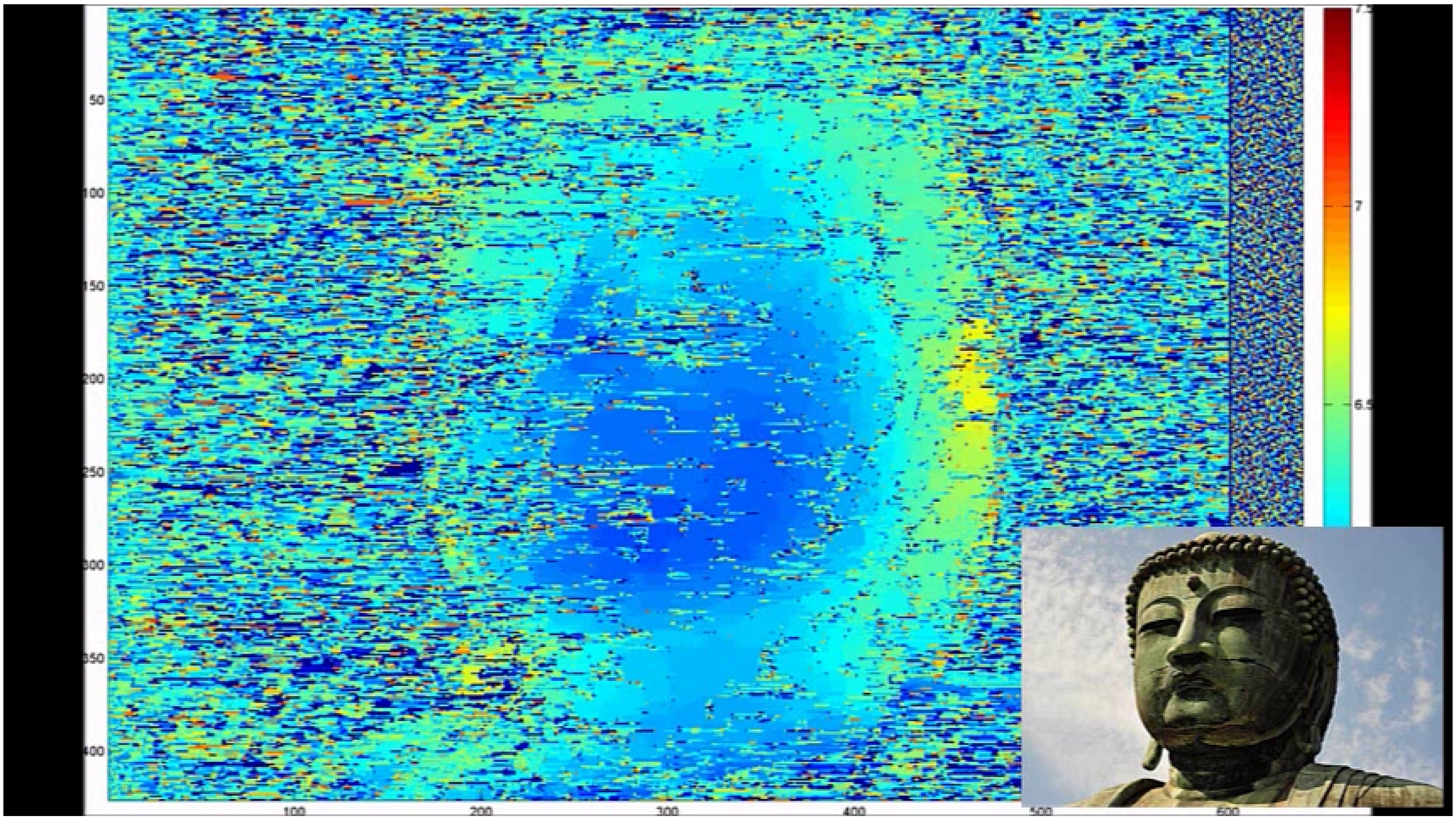
- Propagations in other directions





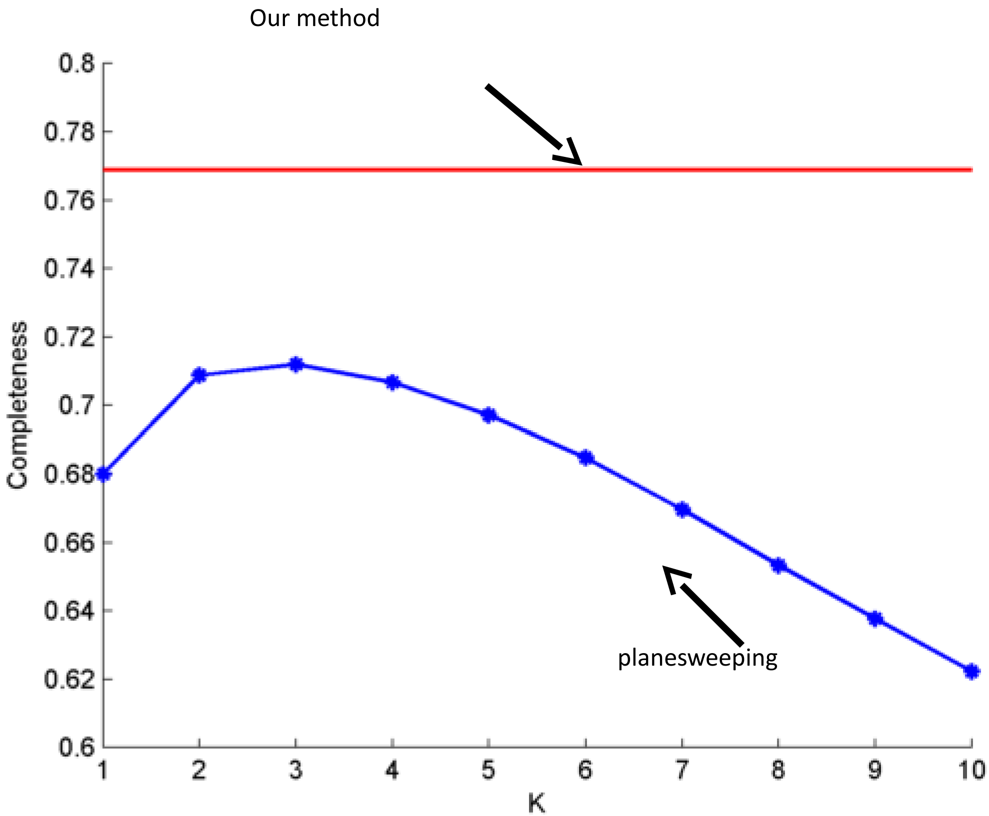
Our method

- Four kinds of propagations to generate depthmap: 

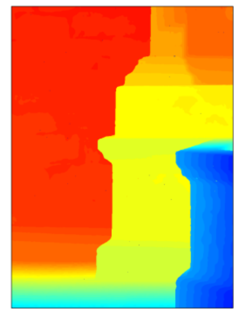


Experiments

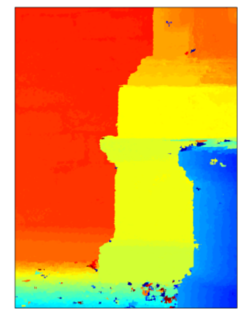
- Compare results with best-K stereo



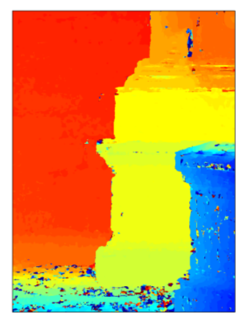
Completeness: percentage of pixels with errors less than 2 cm



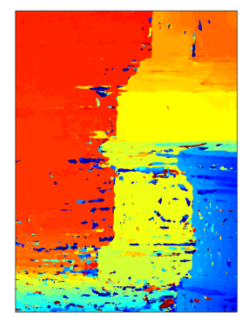
Ground truth



Our result



K=3



K=10

Experiments

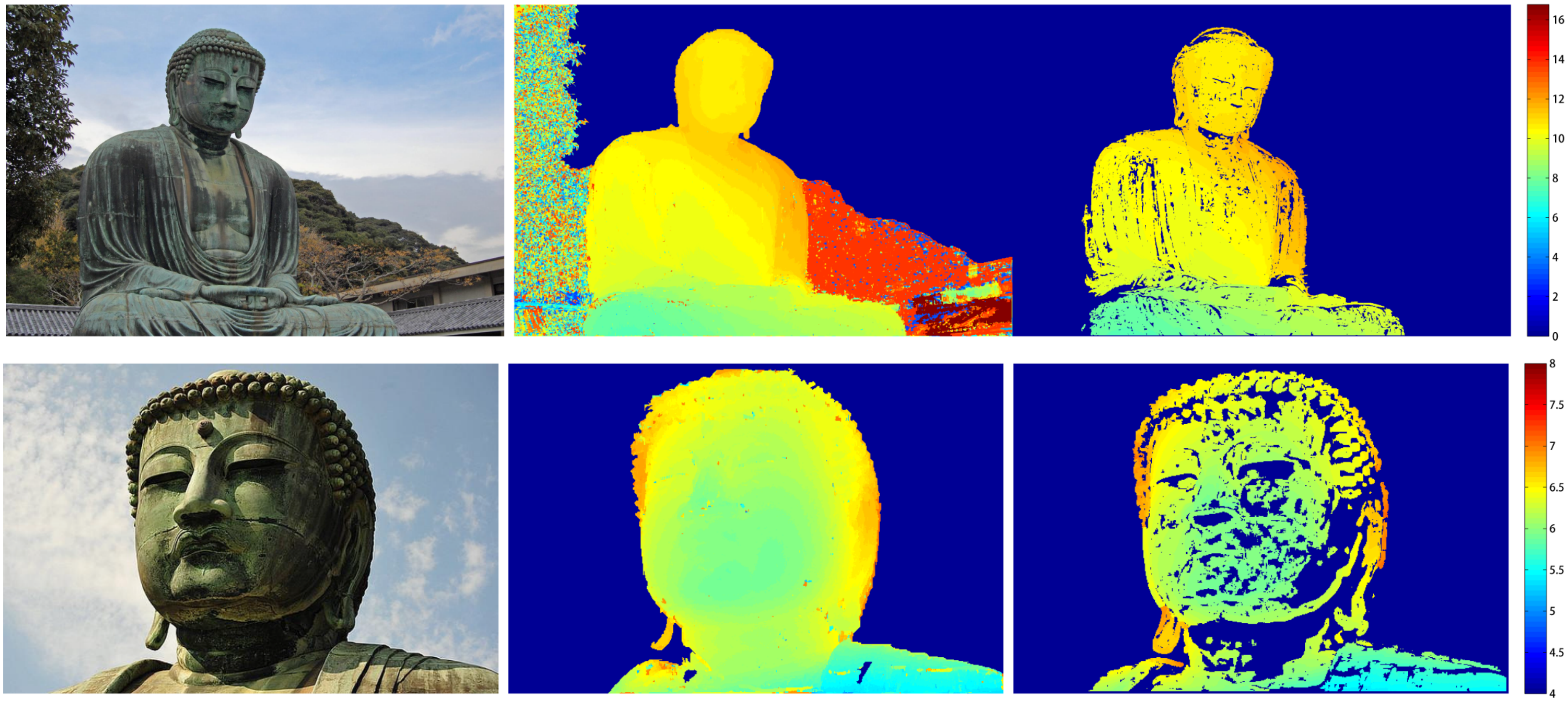
- Compare with other methods

Dataset	Fountain-P11		Herzjesu-P9	
Error threshold	2cm	10cm	2cm	10cm
Our method	**0.769**	0.929	**0.650**	0.844
LC	0.754	**0.930**	0.649	0.848
FUR	0.731	0.838	0.646	0.836
ZAH	0.712	0.832	0.220	0.501
TYL	0.732	0.822	0.658	**0.852**

Percentage of pixels with errors less than a threshold (2 or 10 cm)

LC: Hu et al. 3DIMPVT 2012
FUR: Furkawa et al. PAMI, 2010
ZAH: Zaharescu et al. PAMI 2011
TYL: Tylecek et al. Int'l Journal of VR 2010

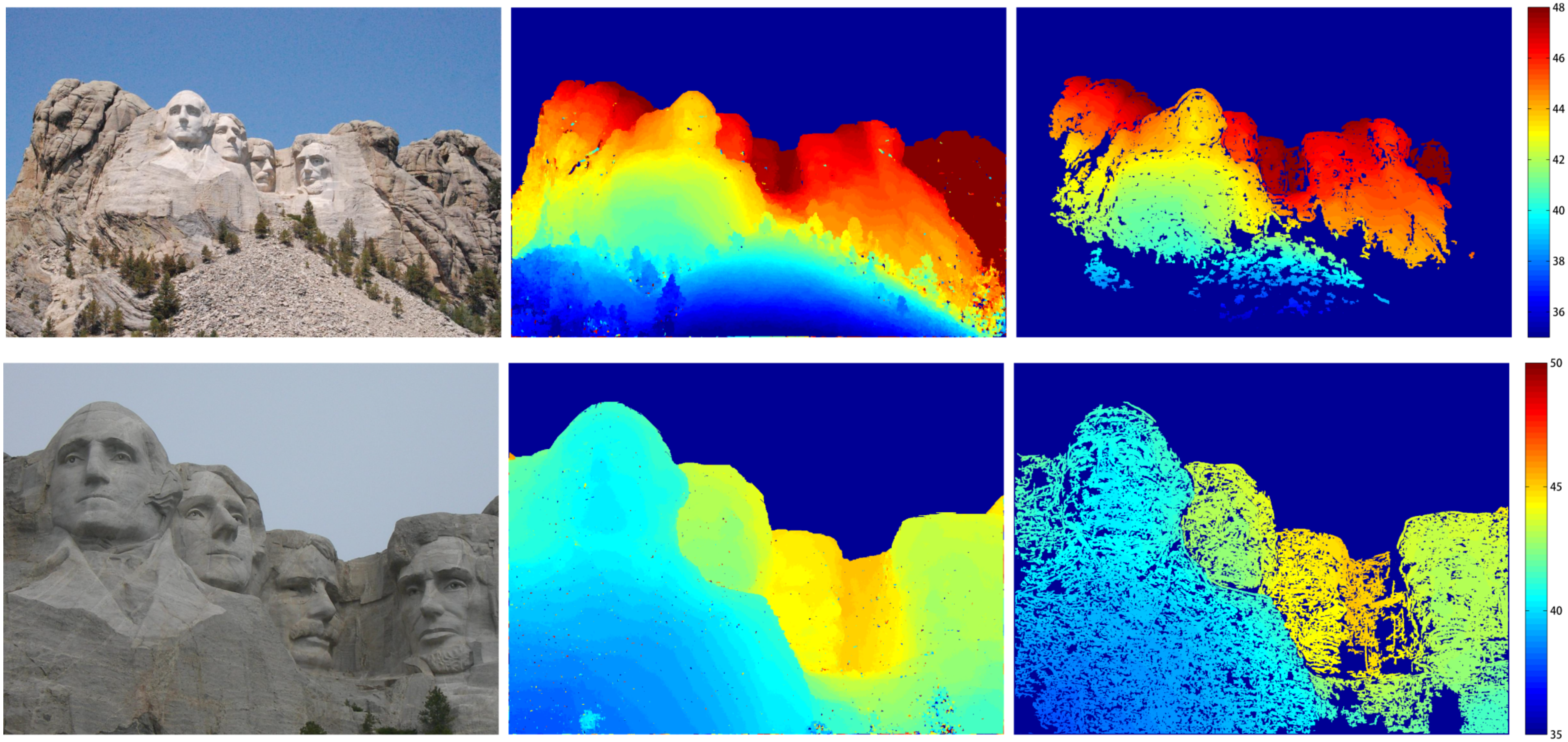
Buddha (212 images)



Source images | Our method | Goesele's method

Runtime: on average 47.3 secs/image

Mt. Rushmore (206 images)



Source images

Our method

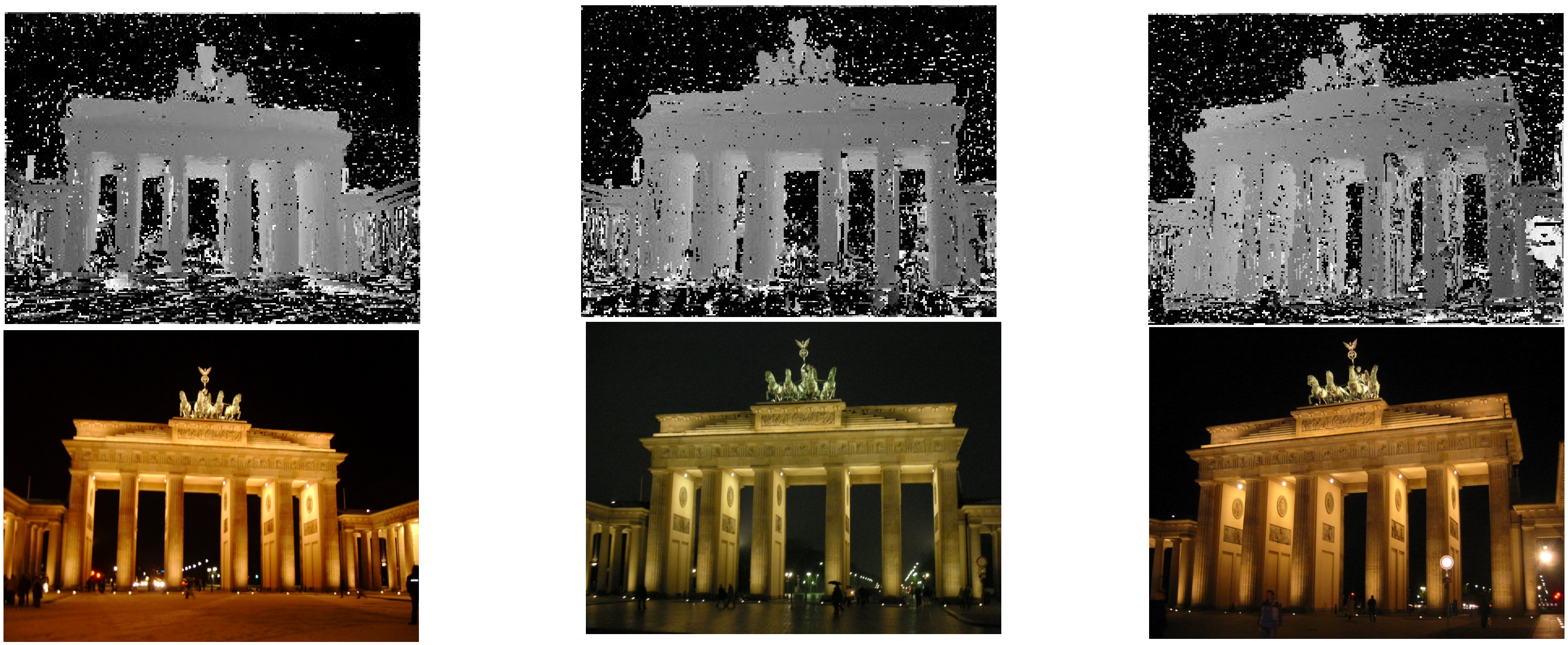
Goesele's method

Heightmap Representation for Depthmap Fusion

Real-time Stereo Estimation

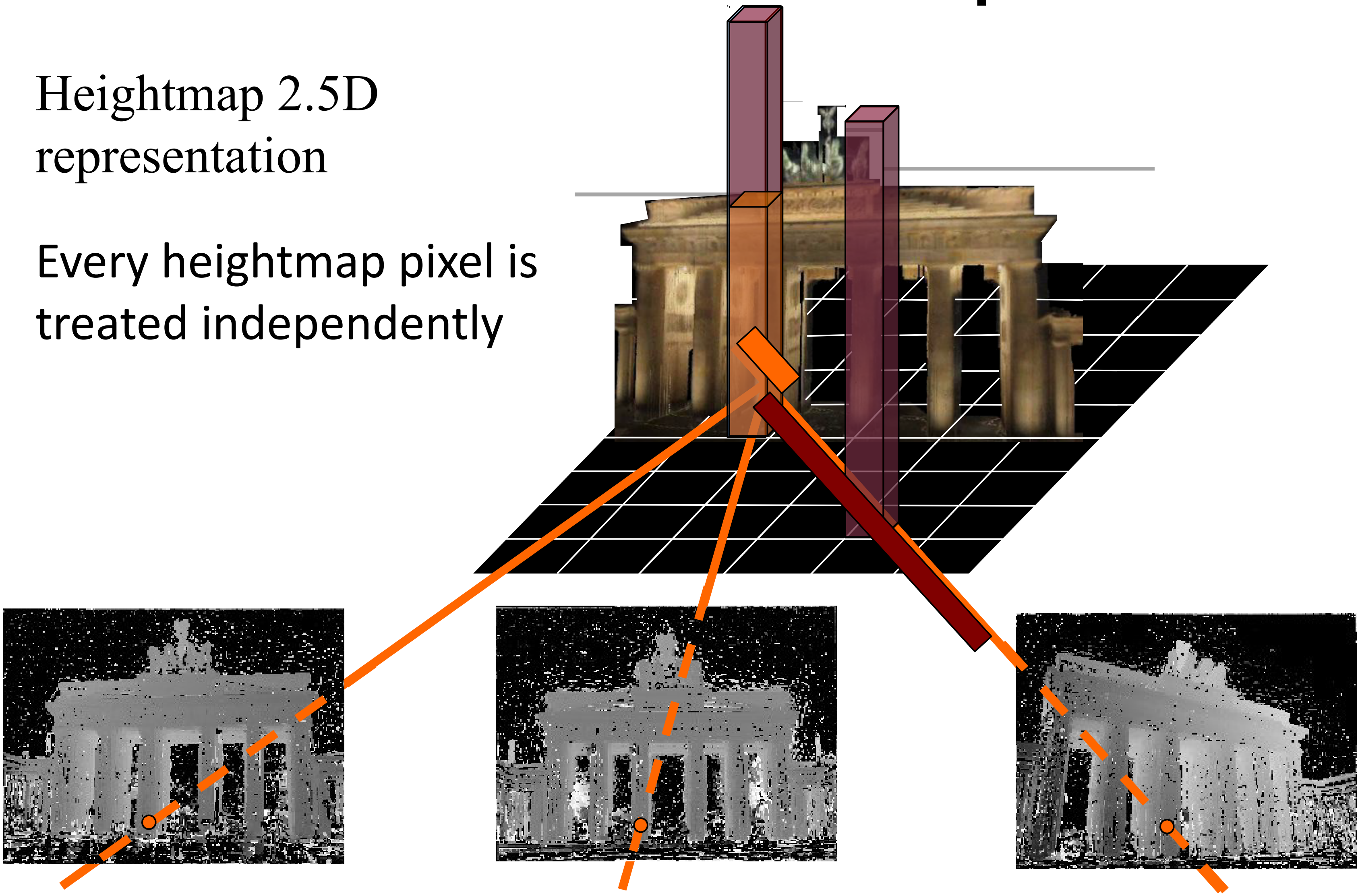
Problem:

- highly noisy depth estimates
- no depth estimates

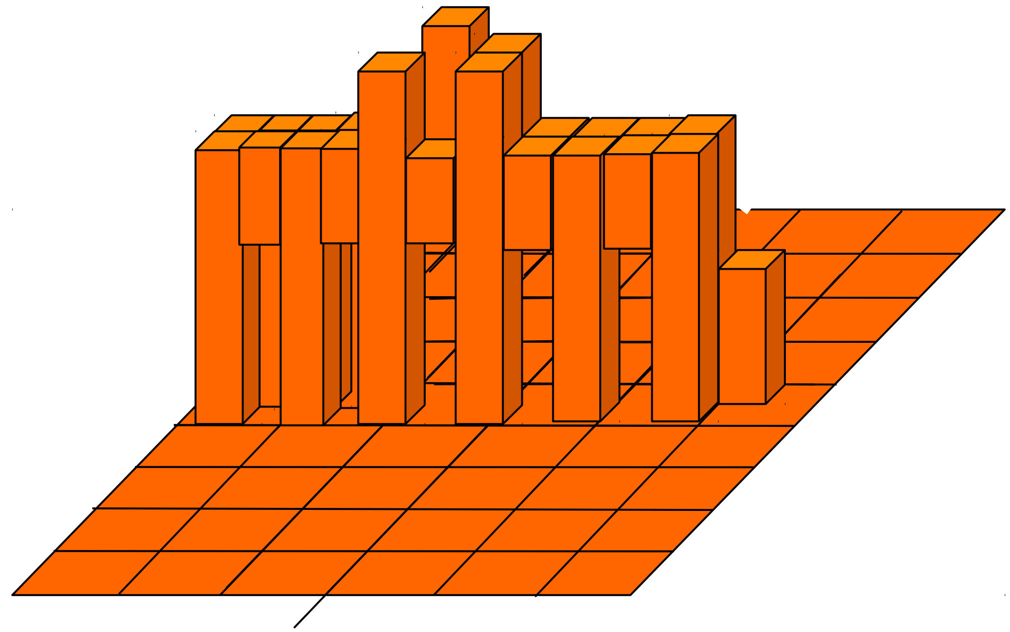


Robust stereo fusion process

- Heightmap 2.5D representation
- Every heightmap pixel is treated independently



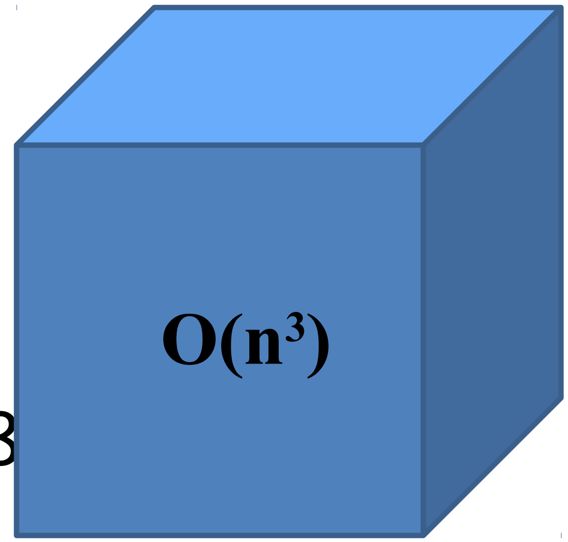
Robust stereo fusion process



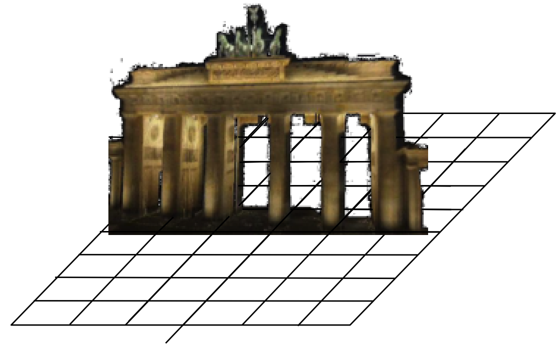
- Enforces vertical facades
- One continuous surface, no holes
- Fast to compute, easy to store $O(n^2)$ instead of $O(n^3)$

Related Work

- Space carving
  - Kutulakos and Seitz 2000
- Graph cuts
  - Vogiatzis et al. 2005
- Level set
  - Faugeras and Keriven 1998
- Convex
  - Zach et al. 2007



Layout



- Vertical direction
  - vanishing point
  - camera constraints (Szeliski 2005, Snavely et al. 2006)

Video

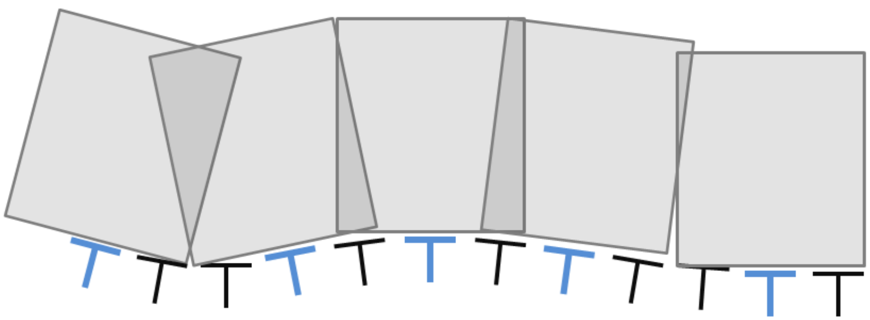
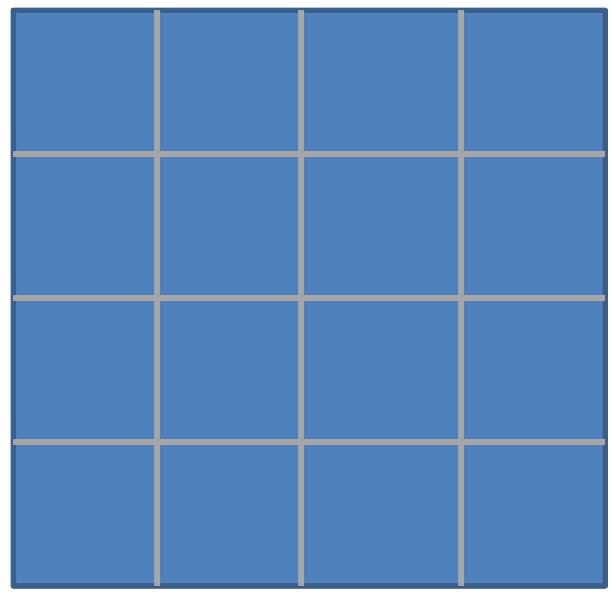


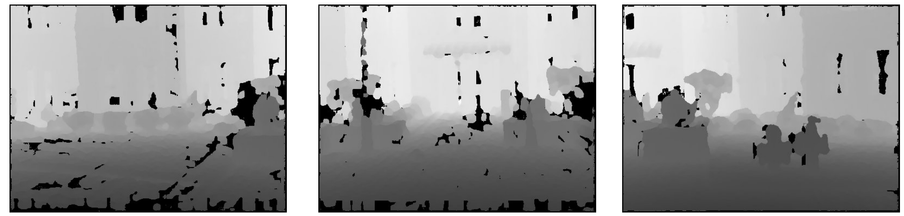
Photo Collections



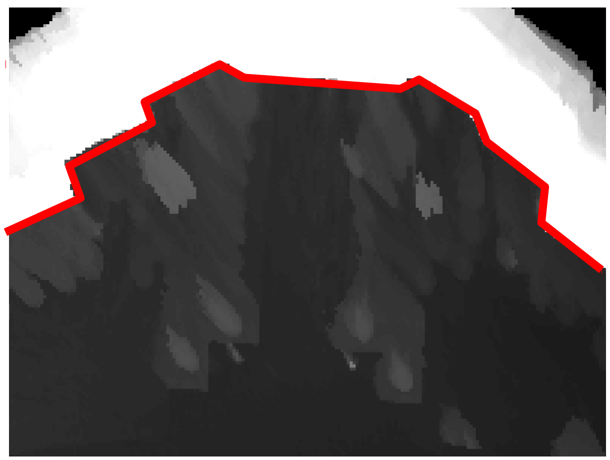
Vertical Surfaces

Gallup et al. 3DPVT 2010

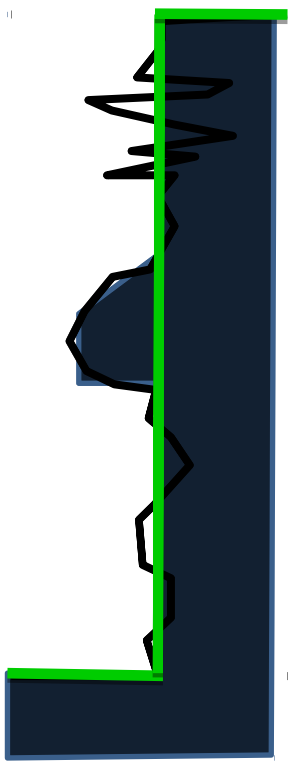
Input Images



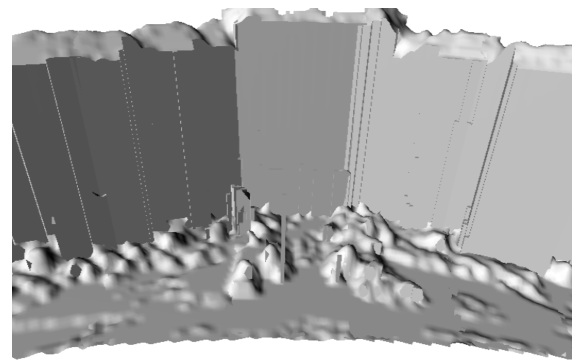
Input Depthmaps



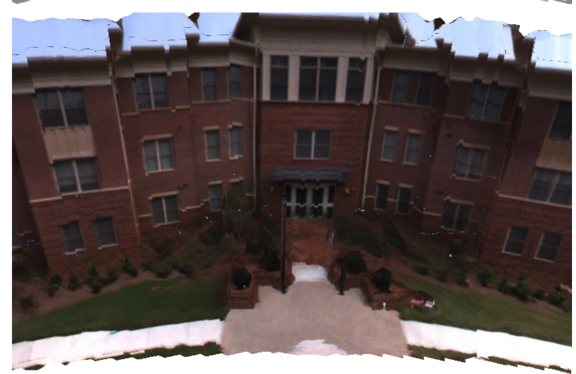
Heightmap
(Top-Down View)
the brighter the higher



3D Model
Geometry

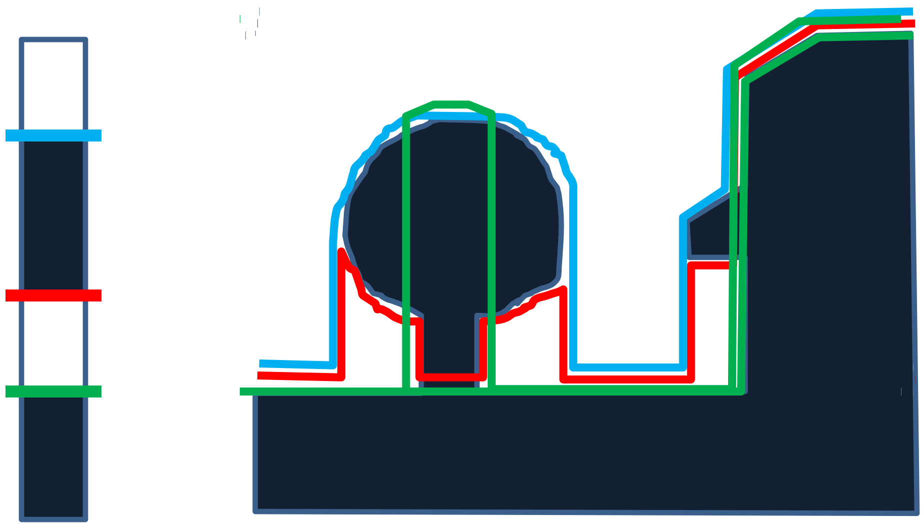


Textured 3D
Model



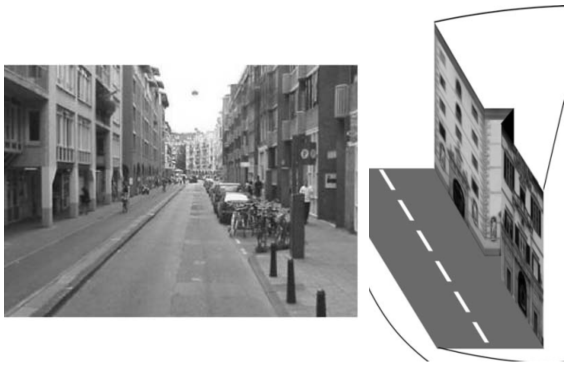
N-Layer Heightmap

- Generalize to n-layer heightmap
- Each layer is a transition from full/empty or empty/full

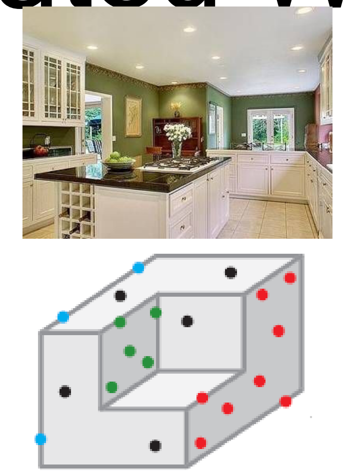


- Compute layer positions with dynamic programming
- Use model selection (BIC) to determine number of layers

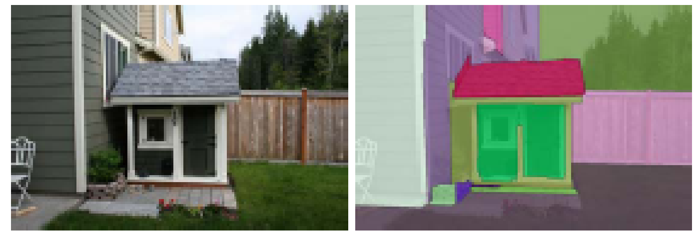
Related Work



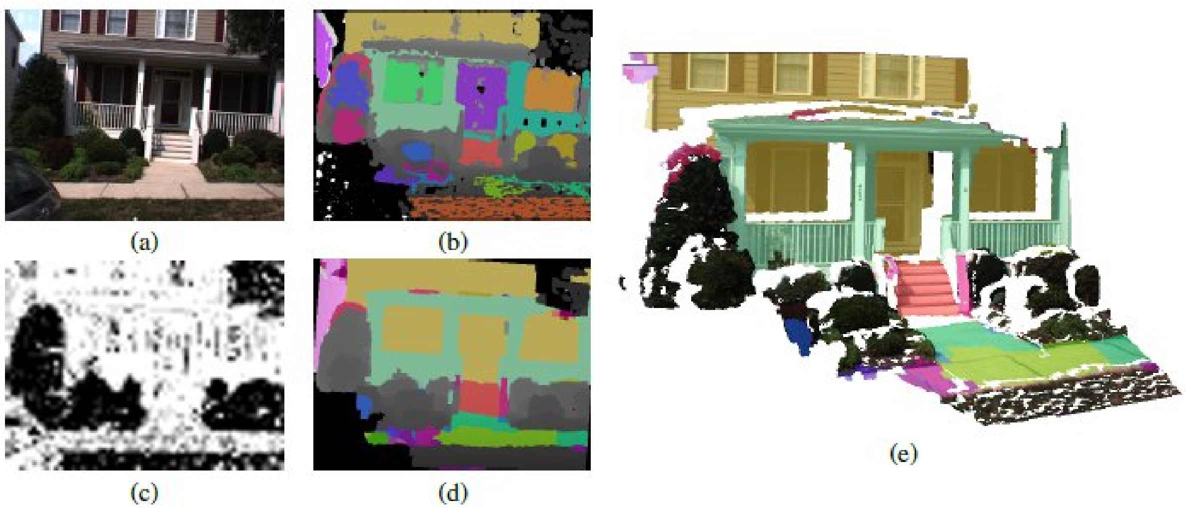
Cornelis et al. 2006



Furukawa et al. 2010



Sinha et al. 2009

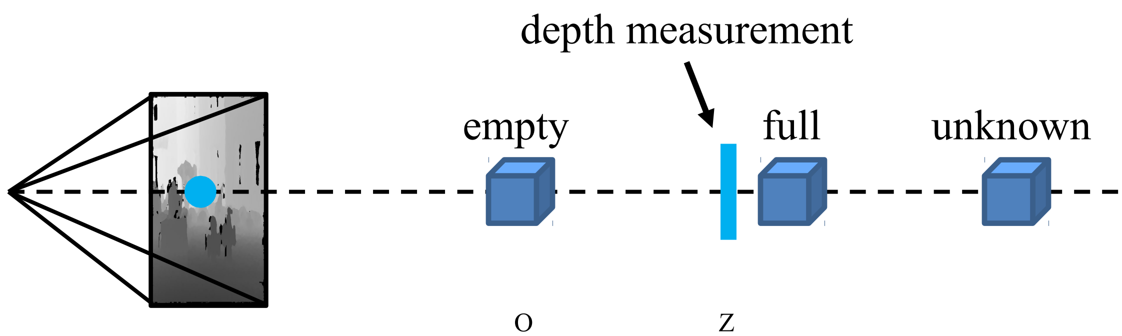


Gallup et al. 2010

Probabilistic Occupancy

Introduced in robotics: Margaritis and Thrun 1998

Follows derivation of Guan et al. 2008



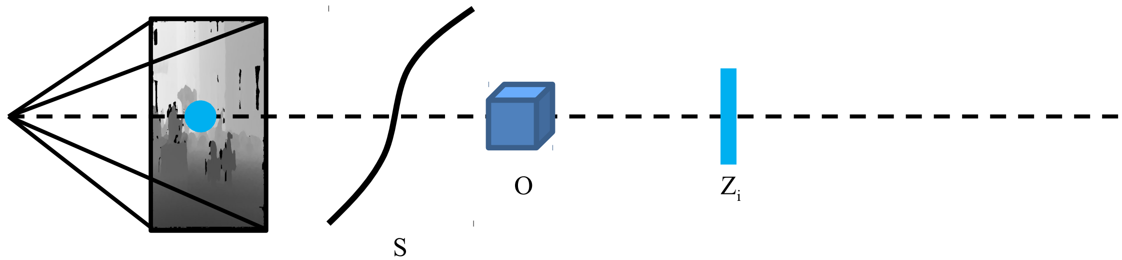
Random Variables

O – binary, occupancy of voxel

Z={Z_i} – continuous, depth measurements for i=1…k depth maps

$$P(O|Z) \qquad \underbrace{\hphantom{xxxxxxxx}}$$

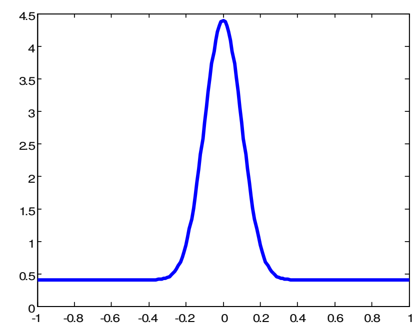
Probabilistic Occupancy



$$P(Z_i|O)$$

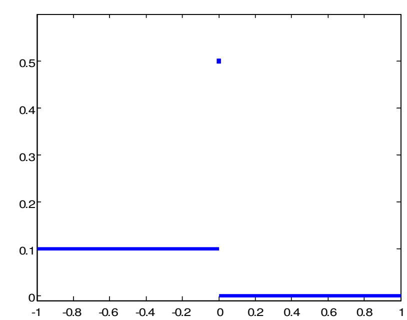
Measurement Model

$$P(Z_i|S,O) = P(Z_i|S) = \begin{cases} \mathcal{N}(S,\sigma)|_{Z_i} & \text{if inliner} \\ \mathcal{U}(z_{min}, z_{max})|_{Z_i} & \text{if outlier} \end{cases}$$
$$= \rho\mathcal{N}(S,\sigma)|_{Z_i} + (1-\rho)\mathcal{U}(z_{min}, z_{max})|_{Z_i}$$

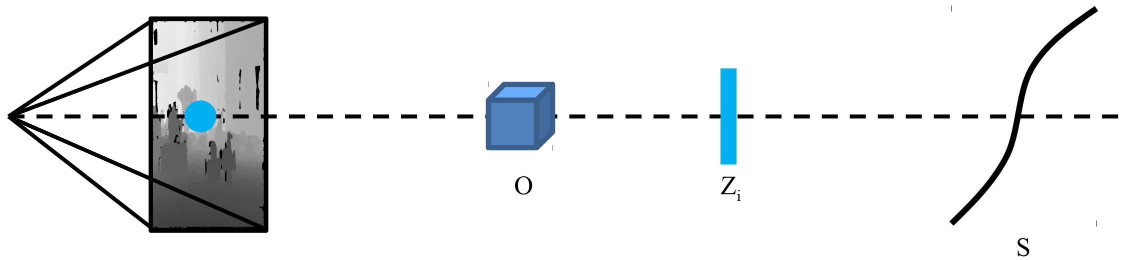


Surface Formation Model

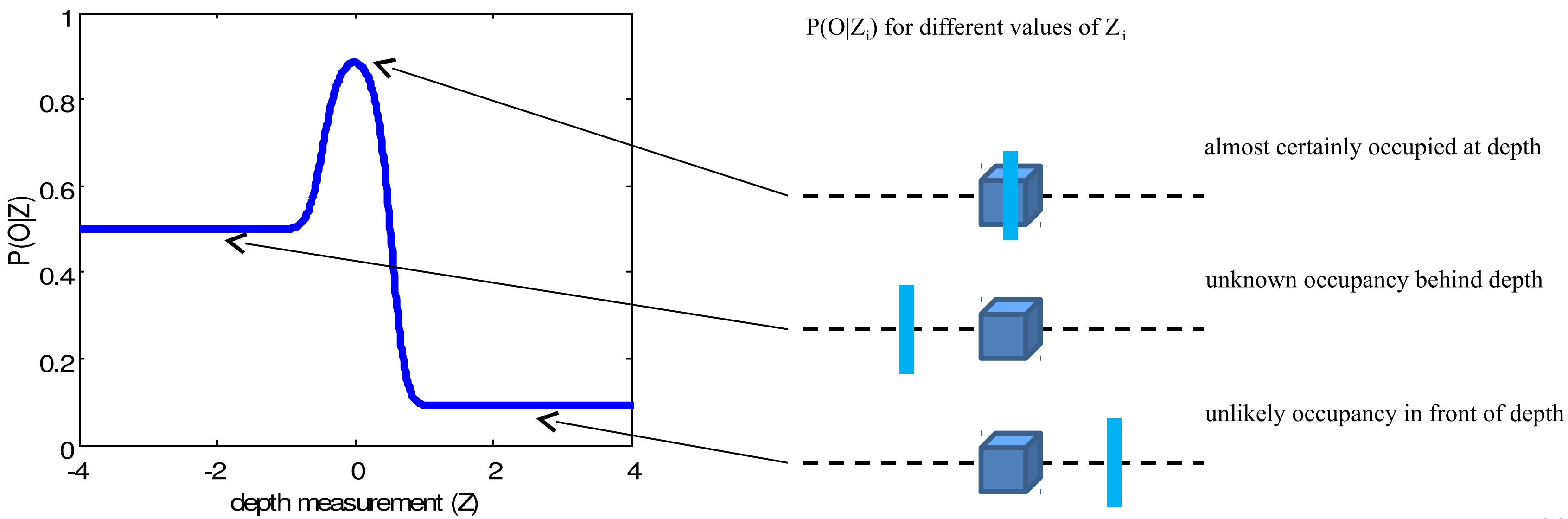
$$P(S|O) = \begin{cases} 1/(z_{max} - z_{min}) & \text{if } S < z_p - \epsilon \\ (1 - z_p/(z_{max} - z_{min})/\epsilon & \text{if } z_p - \epsilon \leq S \leq z_p \\ 0 & \text{if } S > z_p \end{cases}$$



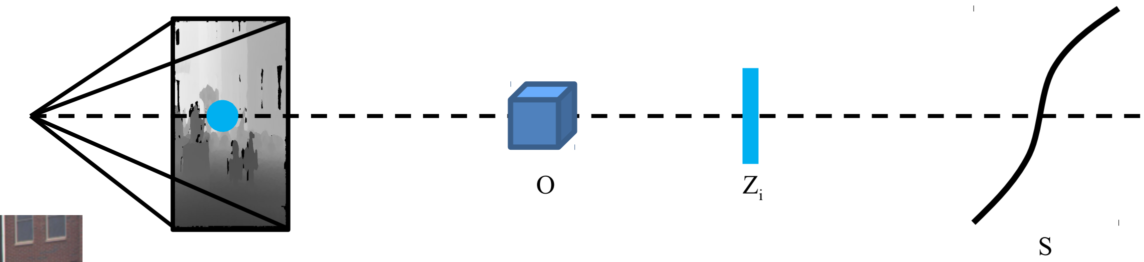
Probabilistic Occupancy

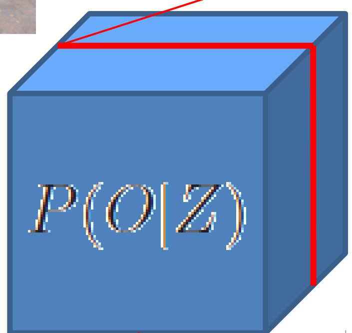


$$P(Z_i|O) = \int_{z_{min}}^{z_{max}} P(Z_i|S,O)P(S|O)dS$$

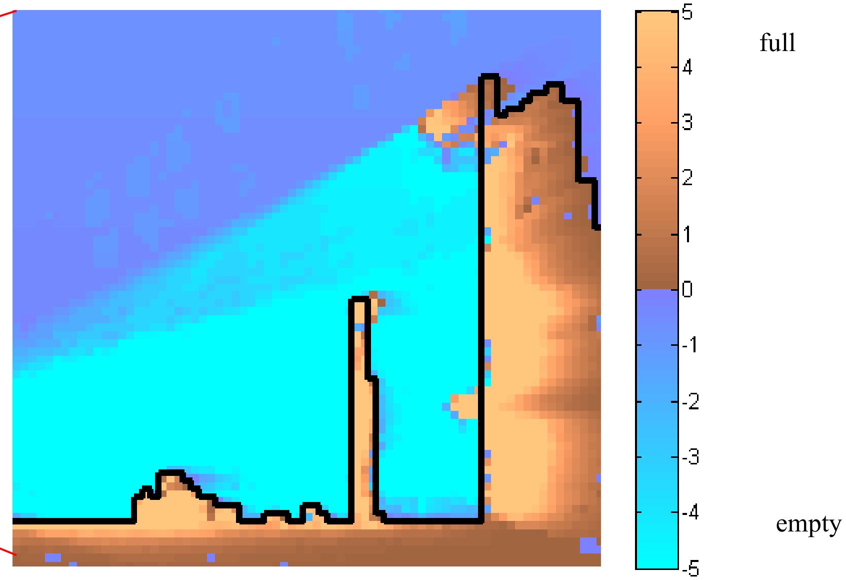


Probabilistic Occupancy



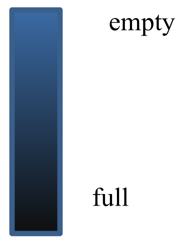


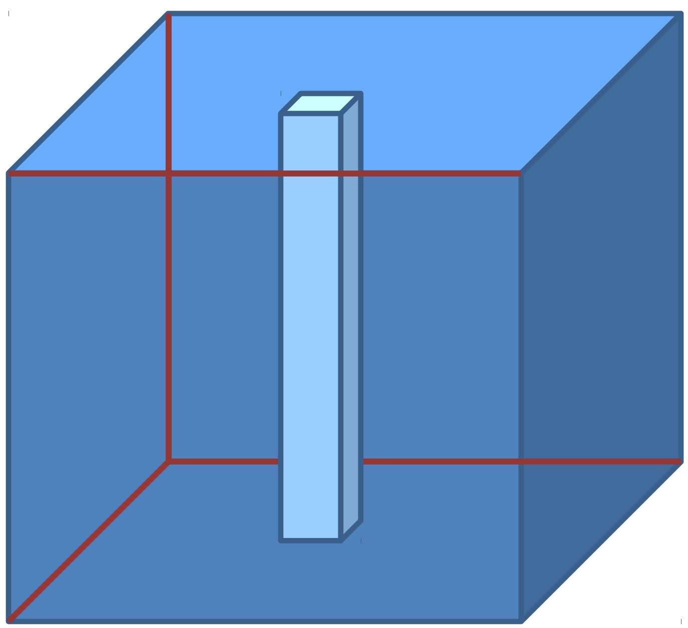
probabilistic
occupancy
volume

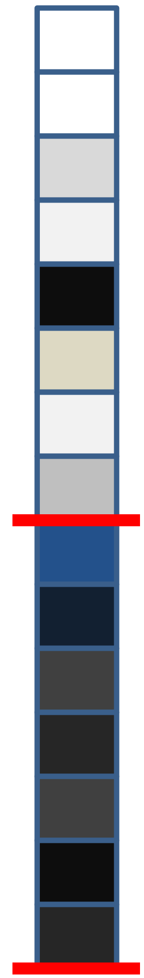


occupancy likelihood slice

Heightmap Computation







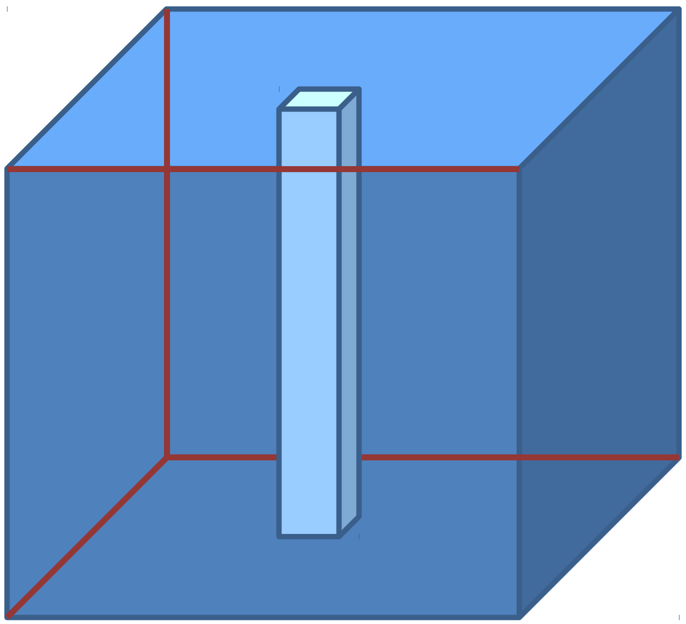
$L(O_m|Z)$

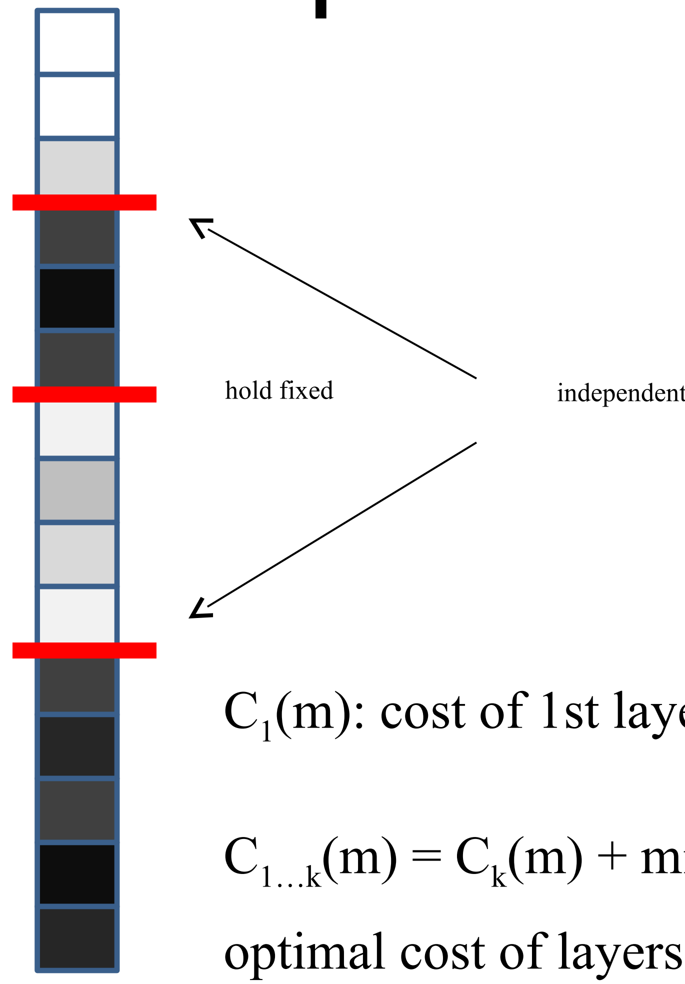
cost of voxels 1…m being full

$$C(m) = \sum_{1}^{m} L(Z \mid O_m) + \sum_{m+1}^{M} L(Z \mid \neg O_m)$$

cost of voxels m+1…M being empty

Heightmap Computation





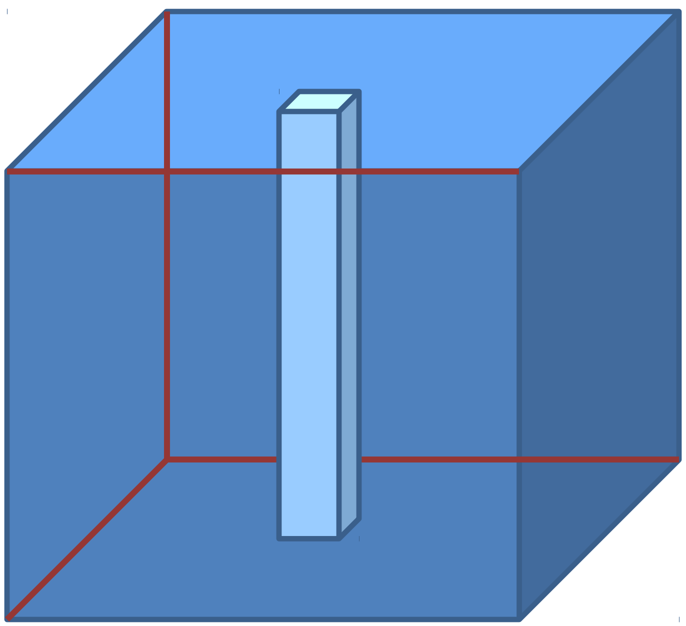
$L(O_m|Z)$

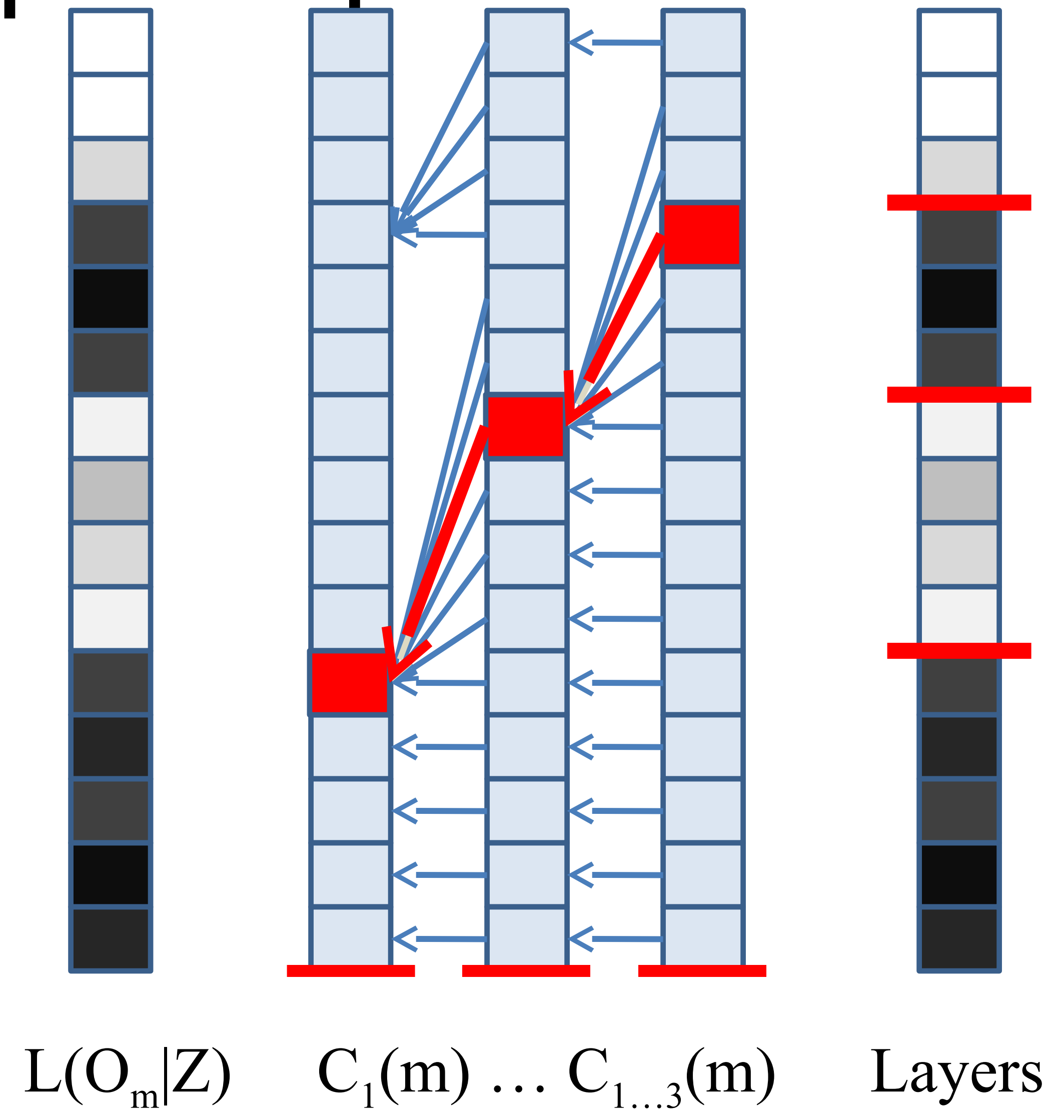
$C_1(m)$: cost of 1st layer at height m

$C_{1\ldots k}(m) = C_k(m) + \min C_{1\ldots k-1}(m')$:

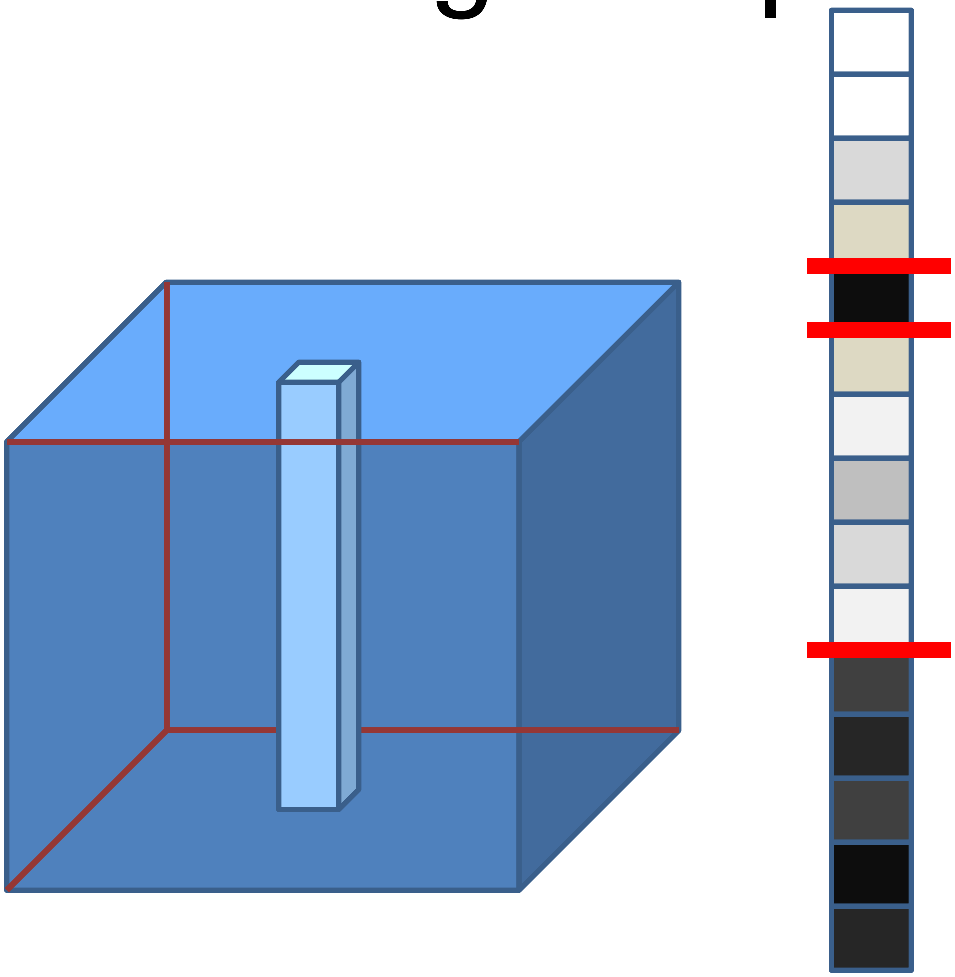
optimal cost of layers 1…k with layer k fixed at height m

Heightmap Computation





Heightmap Computation



$$C_{1\ldots k}(m) = C_k(m) + \min C_{1\ldots k-1}(m')$$

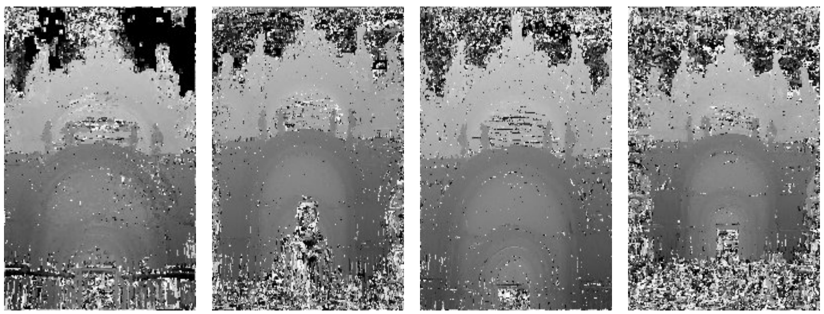
$+ \frac{1}{2} \ln |Z|$ if $m \neq m'$

User sets maximum number of layers.
Model selection determines if less are needed.

$L(O_m|Z)$

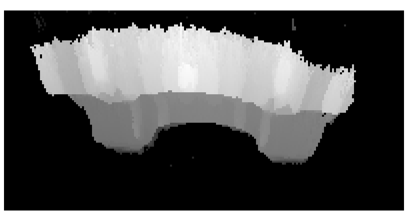
Heightmap Result

photos and depthmaps

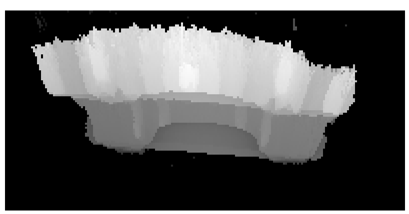


1 vs 3 Layers

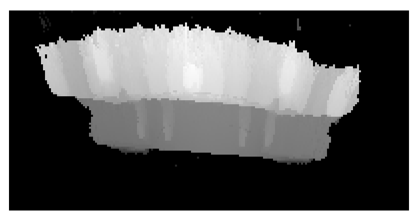
Heightmap Layers



Layer 1



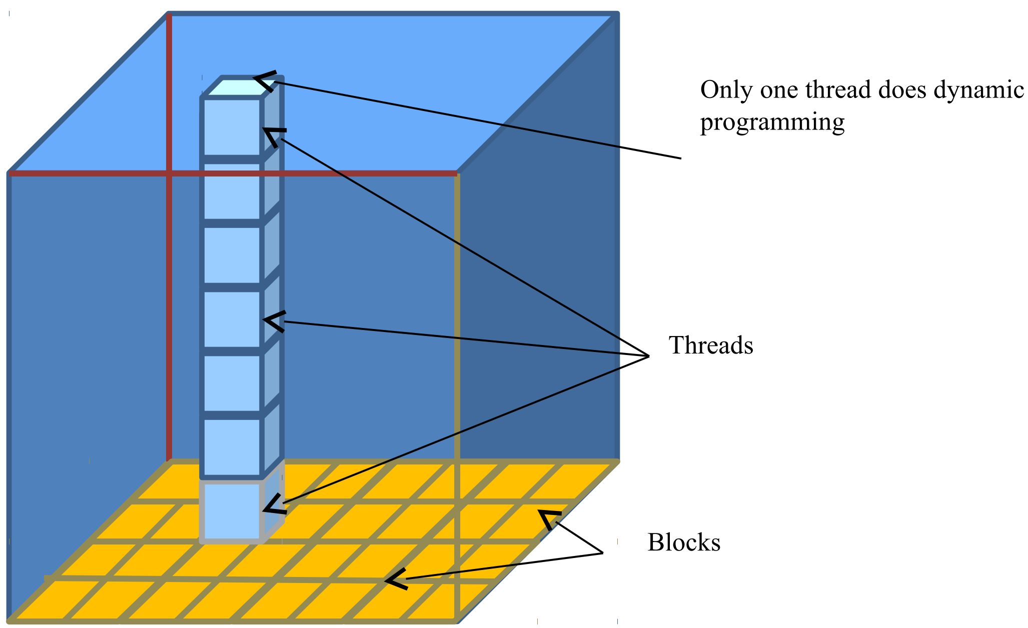
Layer 2



Layer 3

Implementation

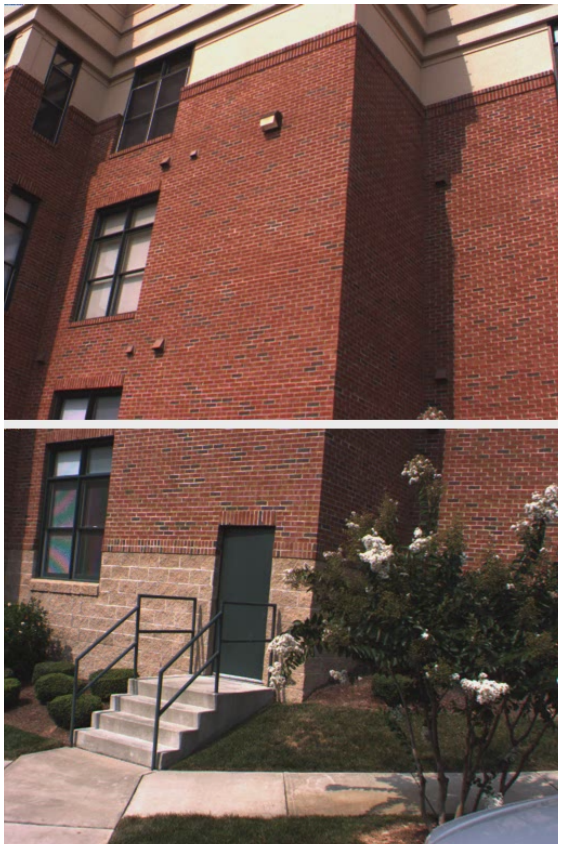
- Implemented in CUDA



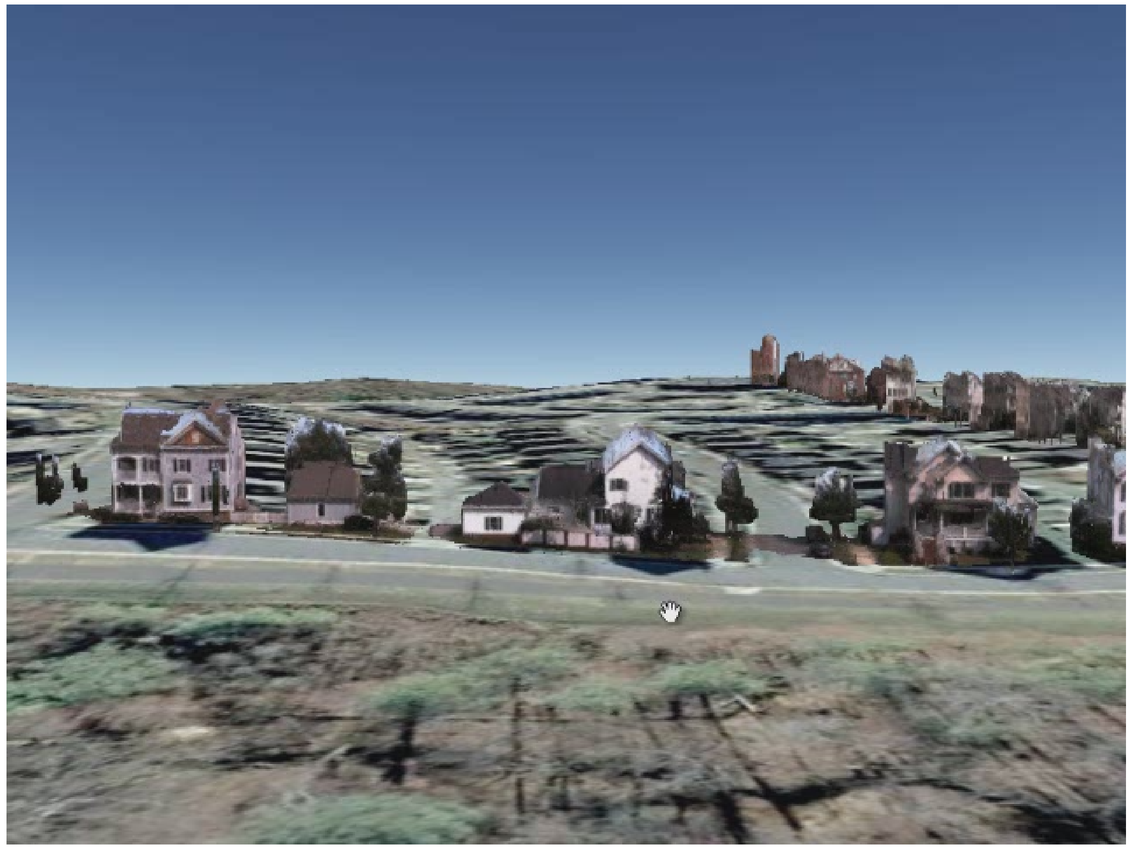
- Computes 100x100(x100) heightmap
  from 50 depthmaps in 70ms on Nvidia GTX 285

Results from Video

Input Videos



3D Reconstruction



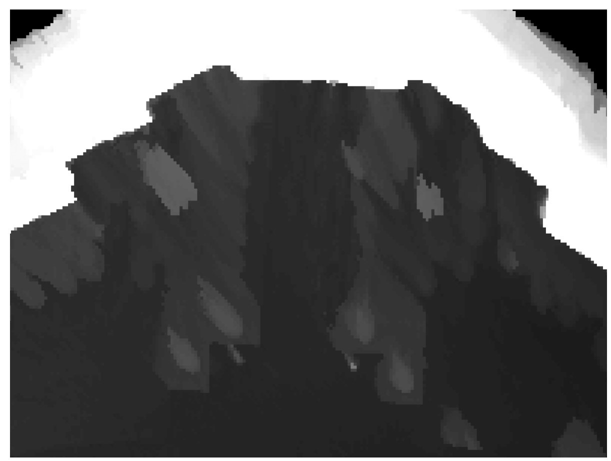
Results from Photo Collections

Input Photos

3D Reconstruction

Conclusion

- Heightmap Advantages
  - less memory: $O(n^2)$ instead of $O(n^3)$
  - parallel: each heightmap pixel independent
  - compact storage
  - vertical surfaces
- Disadvantages
  - Cannot capture all detail



Lecture Outline

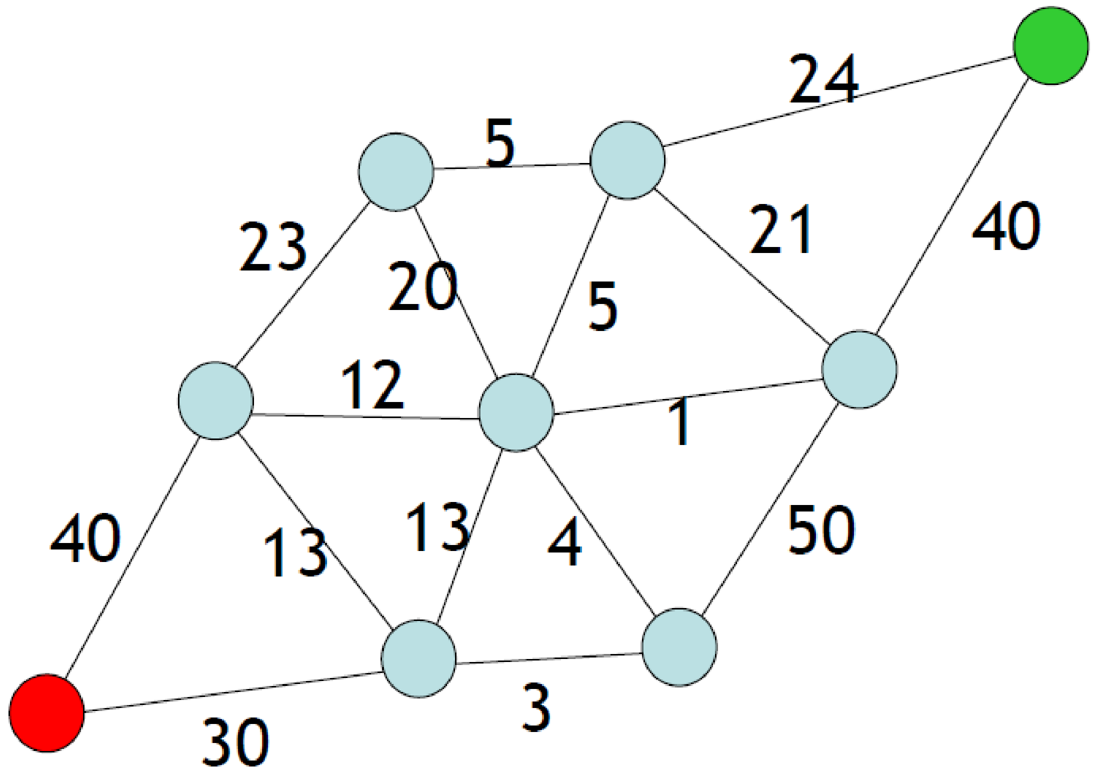
- Multi-view Stereo part II
  - Slides by G. Vogiatzis and L. Zhang
  - Paper by A. Collet et al. (2015)

- Introduction to Computational Geometry
- Convex Hulls
  - David M. Mount, CMSC 754: Computational Geometry lecture notes, Department of Computer Science, University of Maryland, Spring 2012
  - Slides by:
    - B. Gartner, M. Hoffman and E. Welzl (ETH)
    - M. van Kreveld (Utrecht University)
    - P. Indyk and J.C. Yang (MIT)

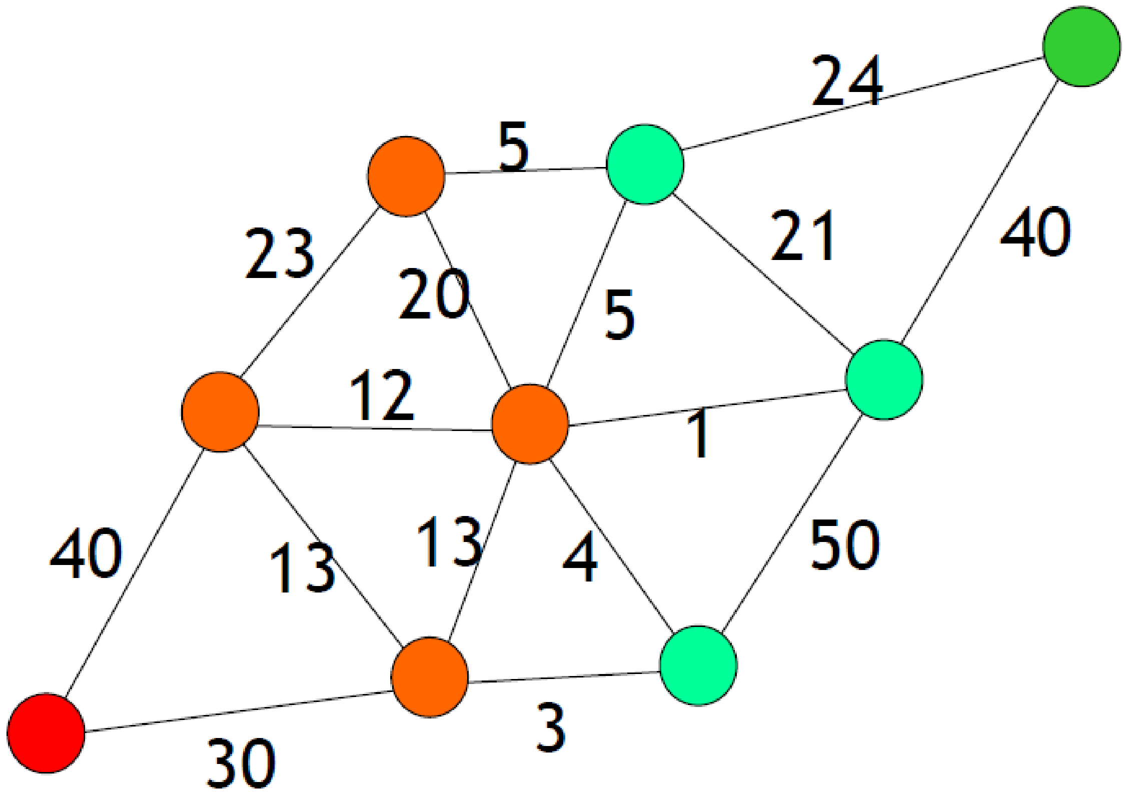
Extracting a Surface from Photo-consistency

- Vogiatzis et al. (PAMI 2007)
- Divide the space in voxels
- Compute the photo-consistency of each voxel
  - By robustly combining all pairwise NCC scores
- Problem: **find a minimum cost surface that separates interior from exterior of the object**
- Add term that favors large volume, otherwise solution collapses to a point

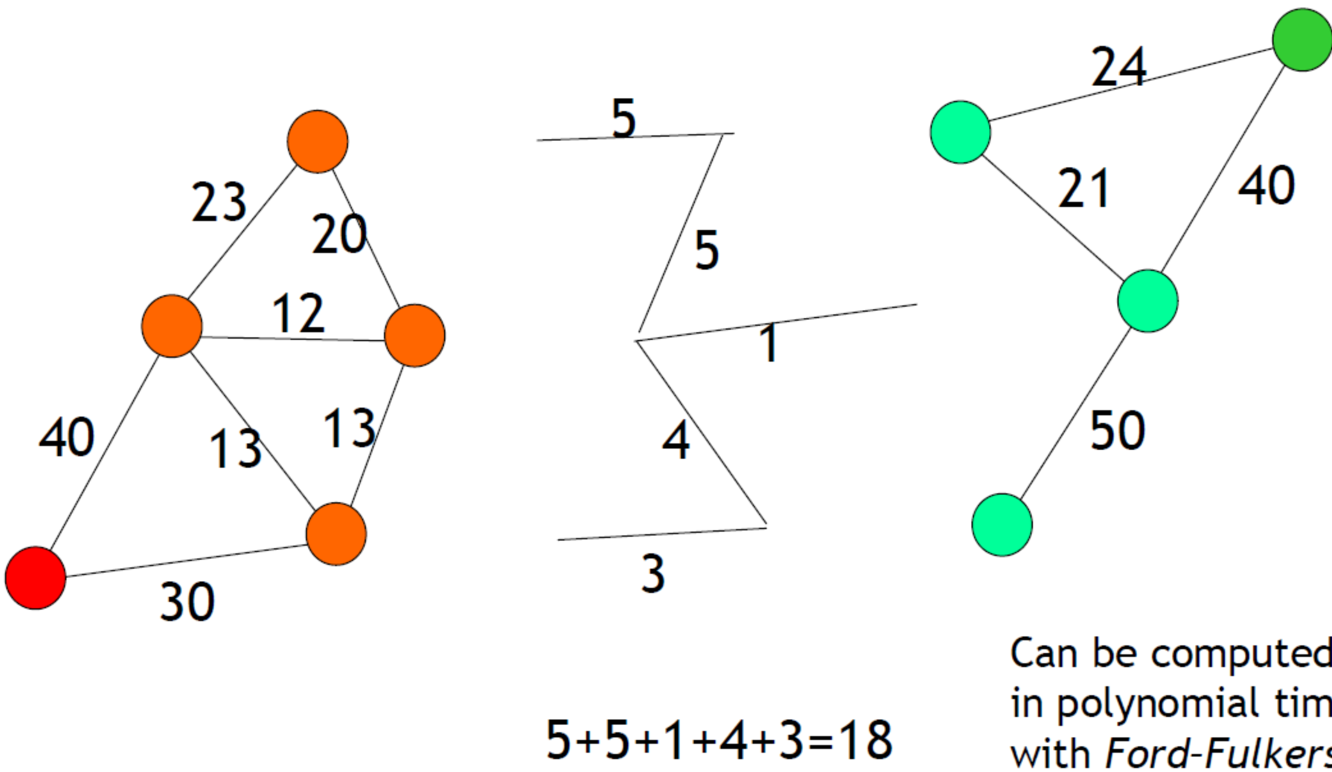
How to Solve?



Graph Cut



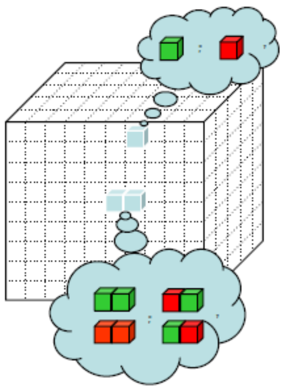
Minimum Cut



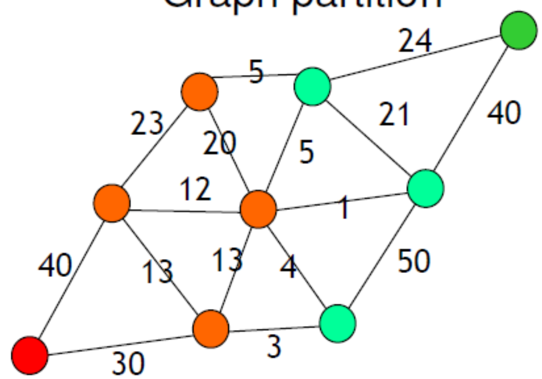
Can be computed
in polynomial time
with *Ford-Fulkerson (1956)*
algorithm

Three Equivalent Representations

Binary labelling



Graph partition

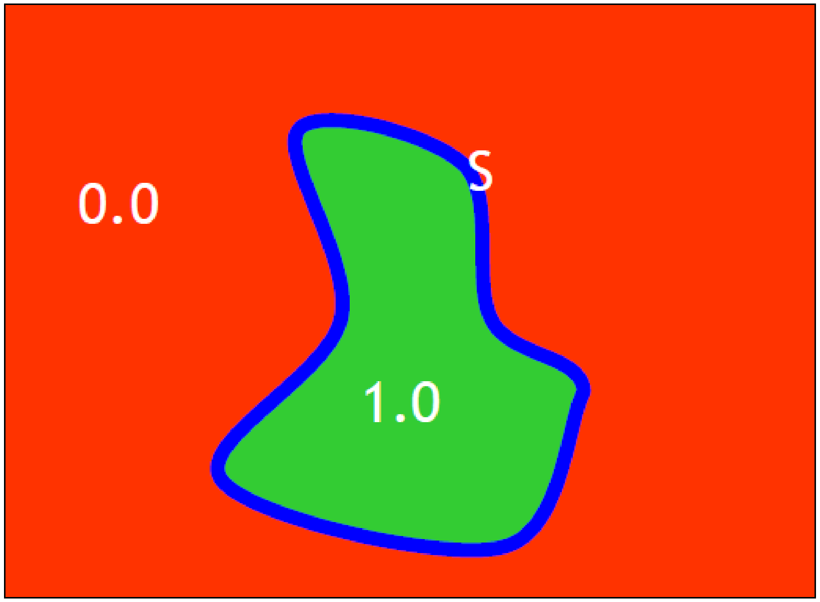


Continious functional

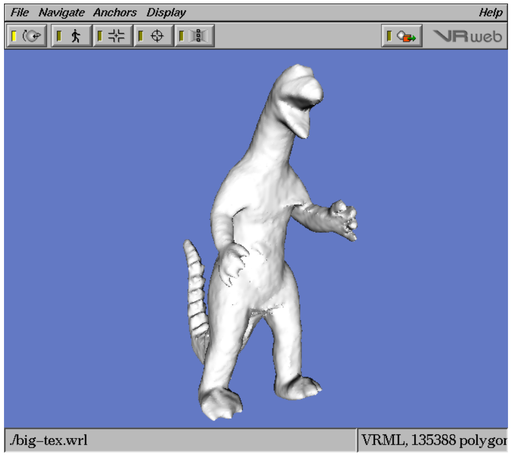
$$E[S] = \iint_S \boxed{\rho(x)dS} + \iiint_{V(S)} \boxed{\sigma(x)dV}$$

Extracting the Surface

- Marching cubes algorithm can extract isosurfaces
  - Matlab: [tri, pts] = isosurface(V)
  - Where V is a binary volume of 0s and 1s

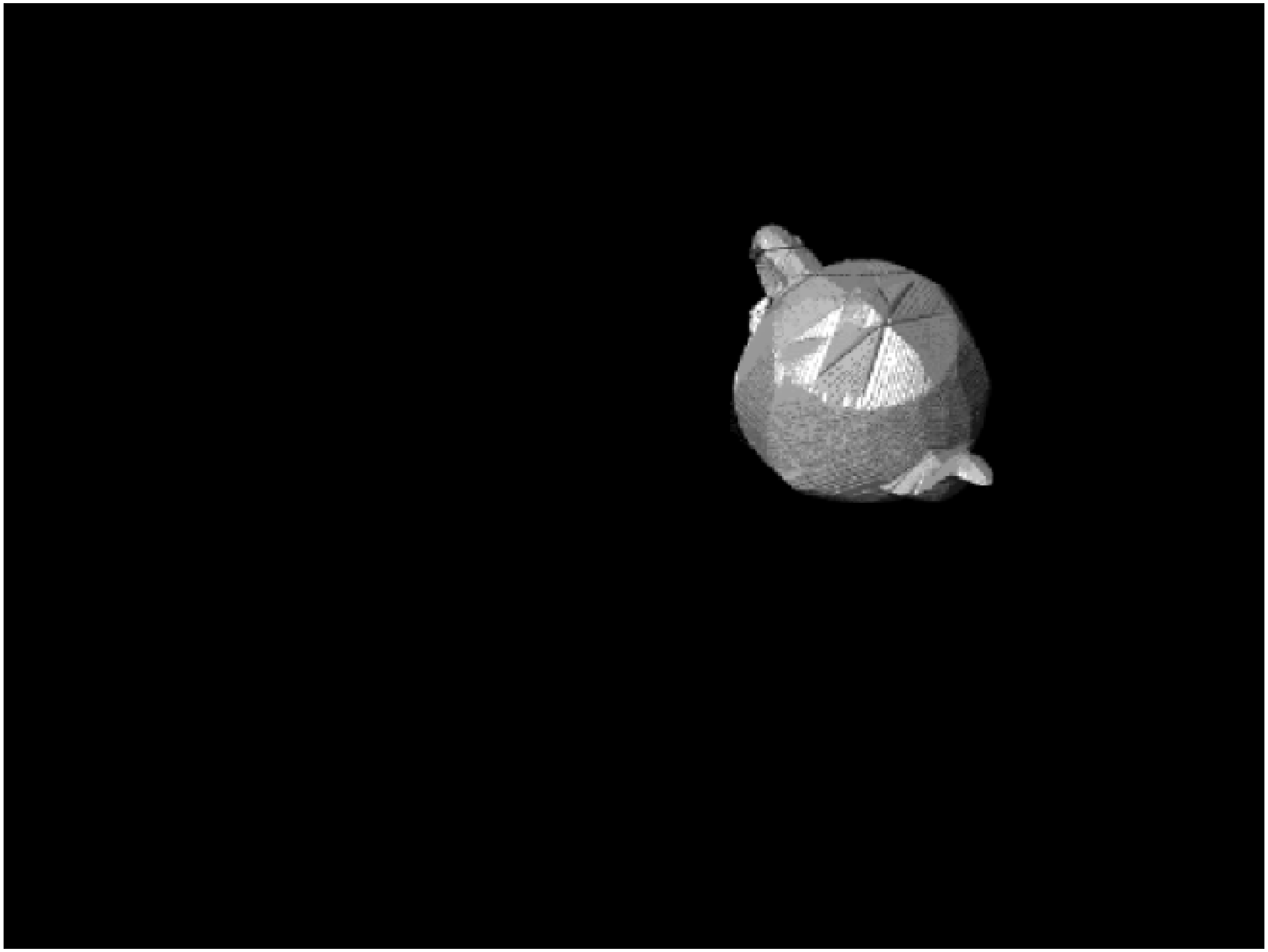


Shape from silhouettes

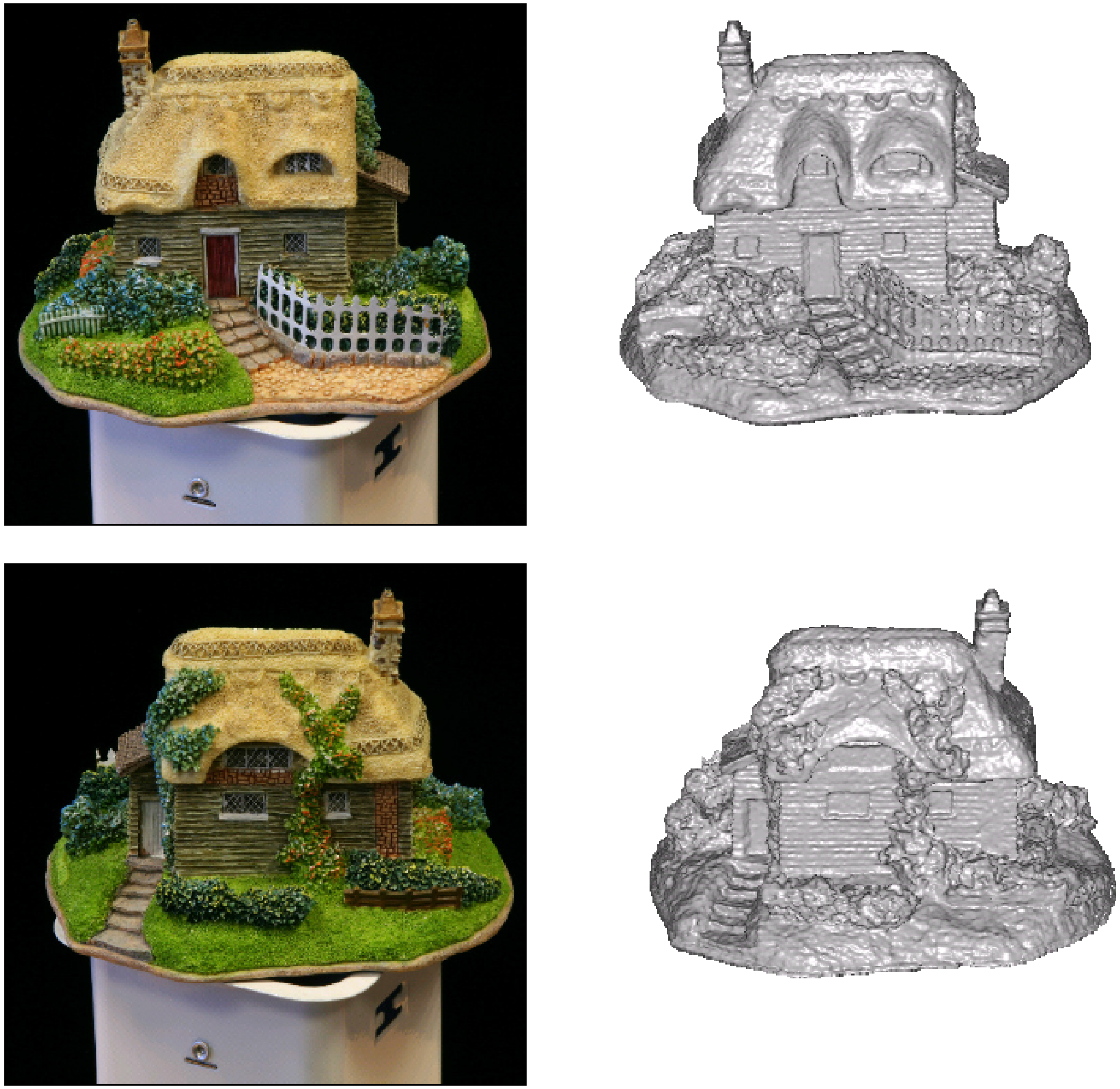


Automatic 3D Model Construction for Turn-Table Sequences,
A.W. Fitzgibbon, G. Cross, and A. Zisserman, SMILE 1998

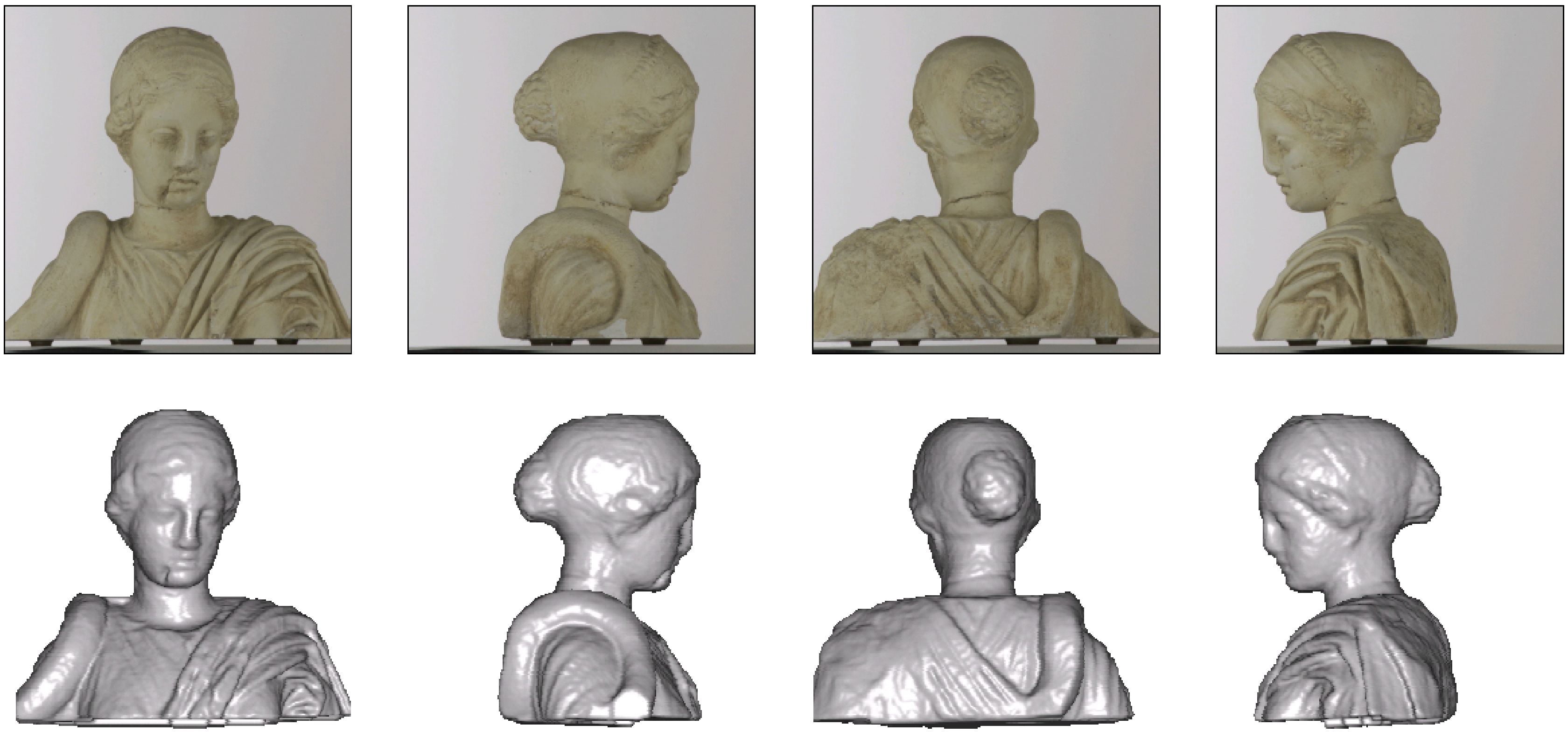
Visual Hull: A 3D Example



Results

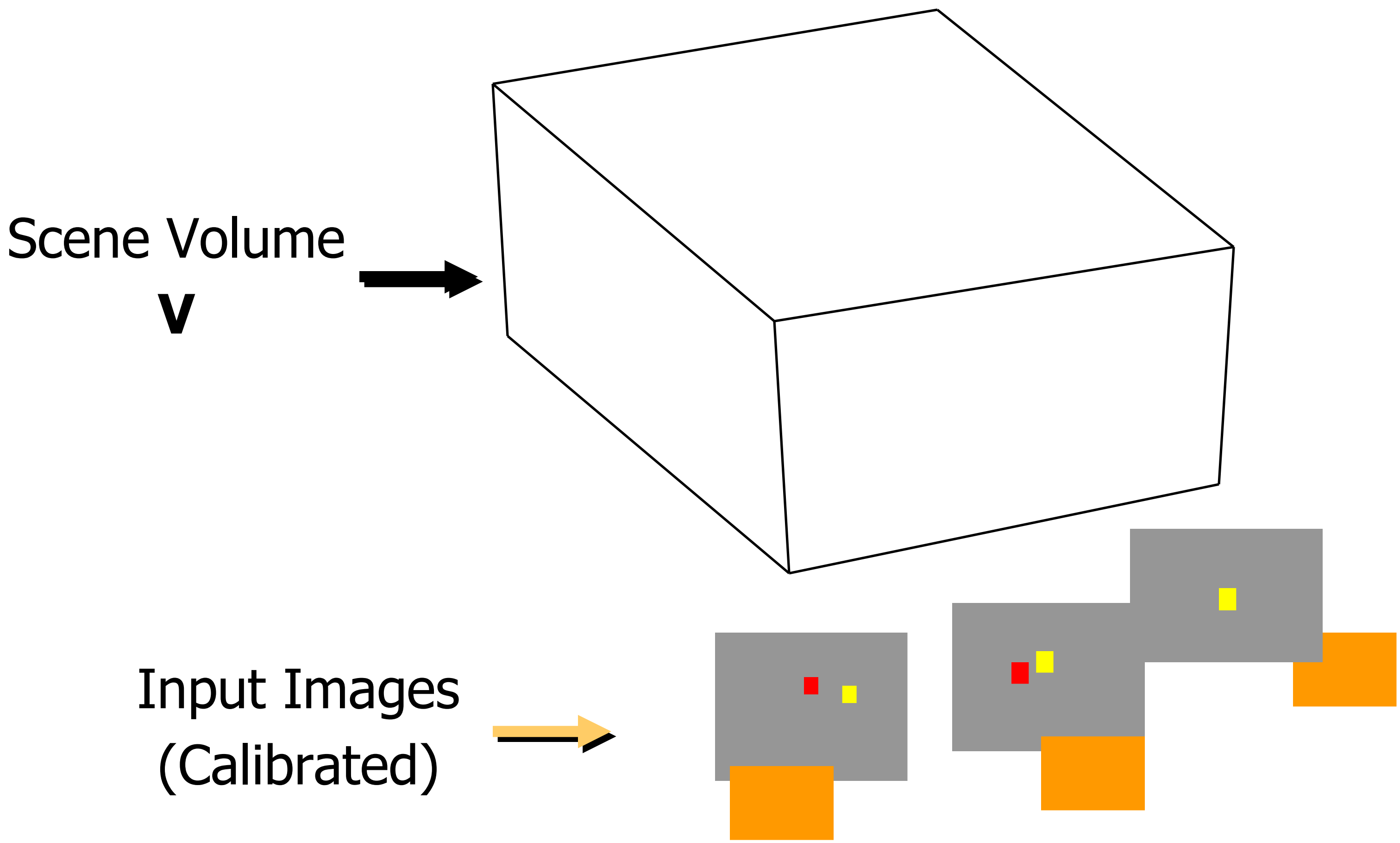


Results



Volumetric Stereo

- Determine occupancy, “color” of points in V
- Slides by L. Zhang

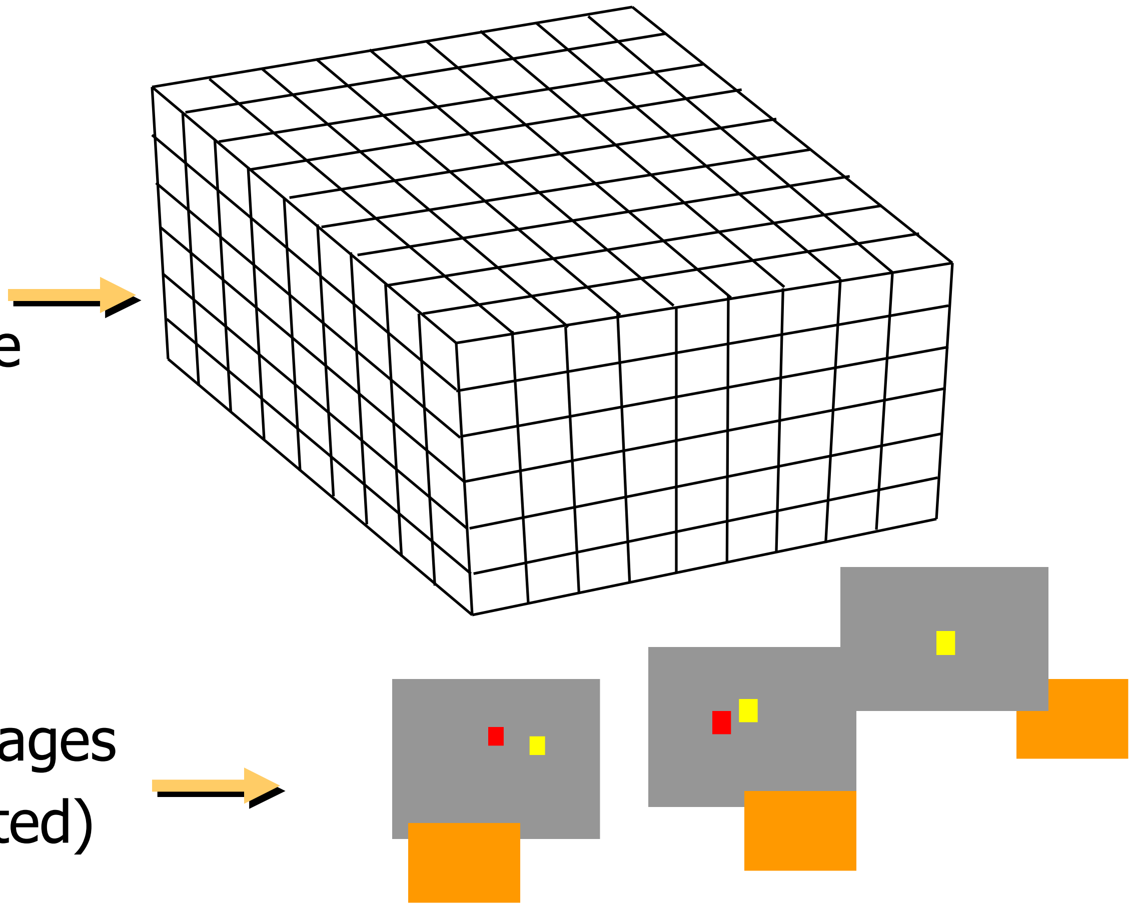


Discrete Formulation: Voxel Coloring

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

Discretized
Scene Volume

Input Images
(Calibrated)

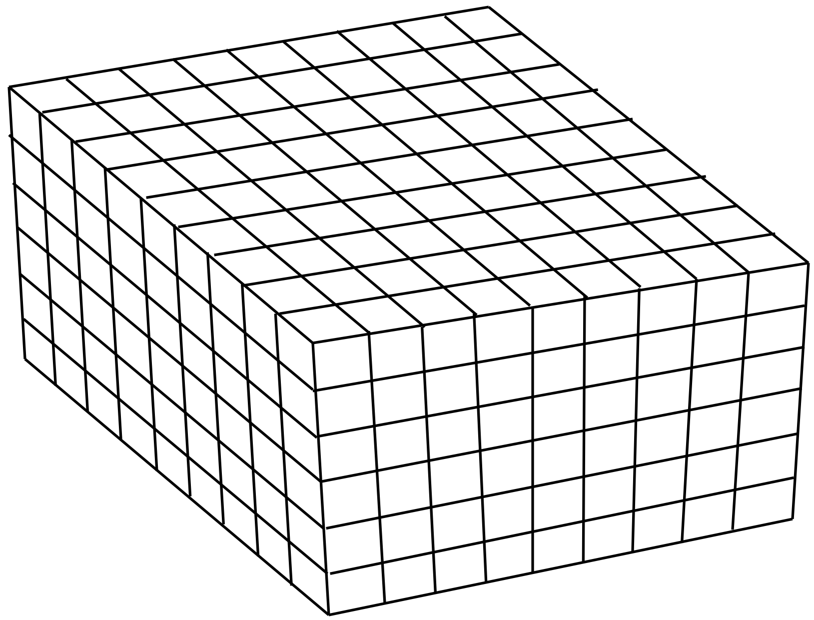


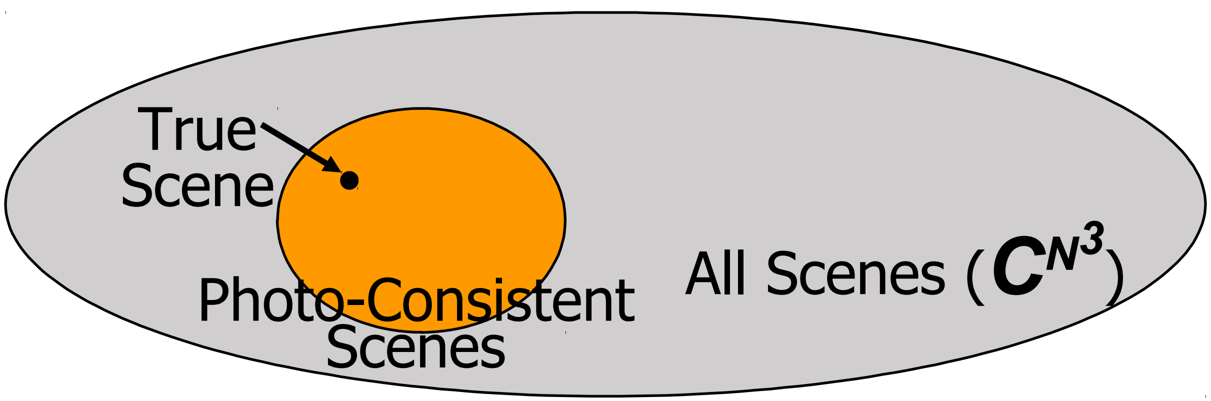
Complexity and Computability

Discretized Scene Volume

N^3 voxels

C colors

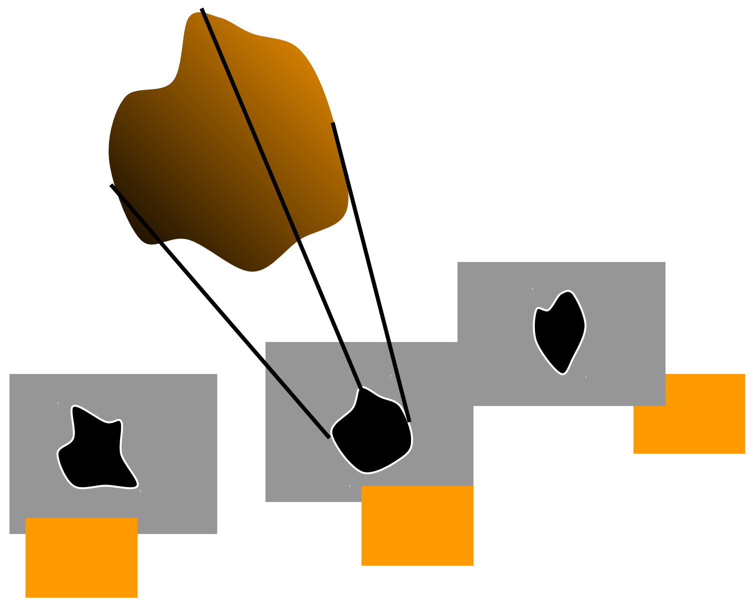




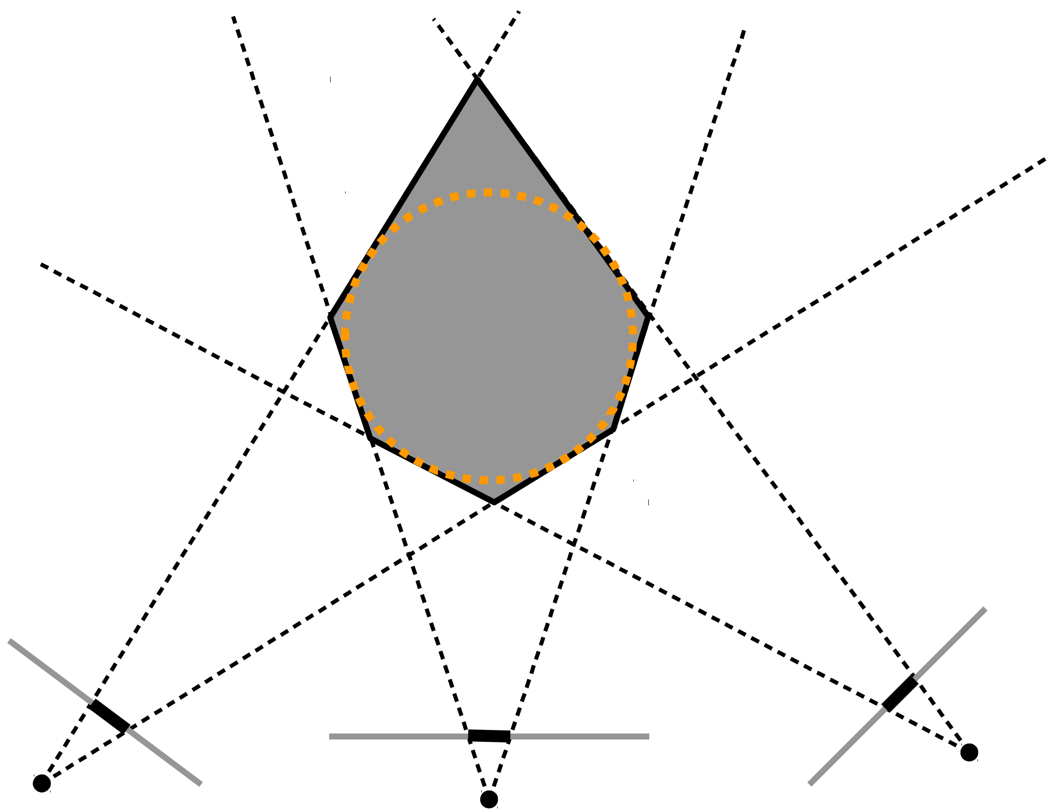
Reconstruction from Silhouettes (C=2)

- Approach:
  - *Back-project* each silhouette
  - Intersect back-projected volumes

Binary Images



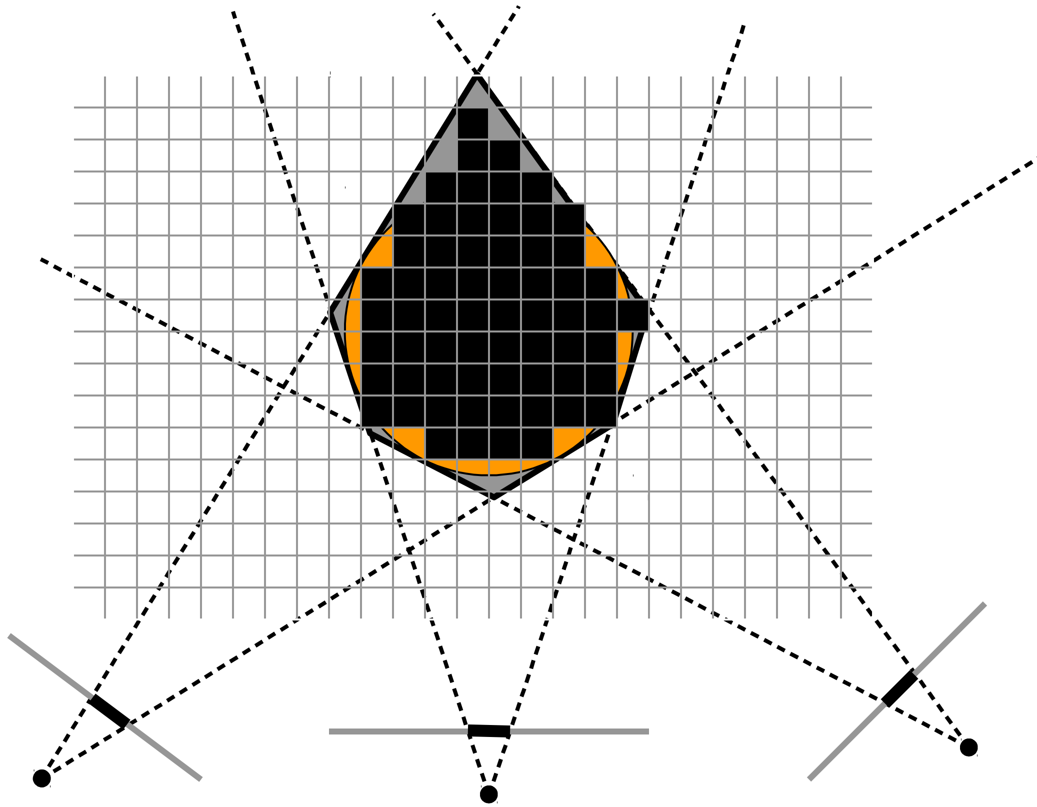
Volume Intersection



Reconstruction Contains the True Scene

- But is generally not the same
- In the limit (all views) we get *visual hull*
  - > Complement of all lines that do not intersect S

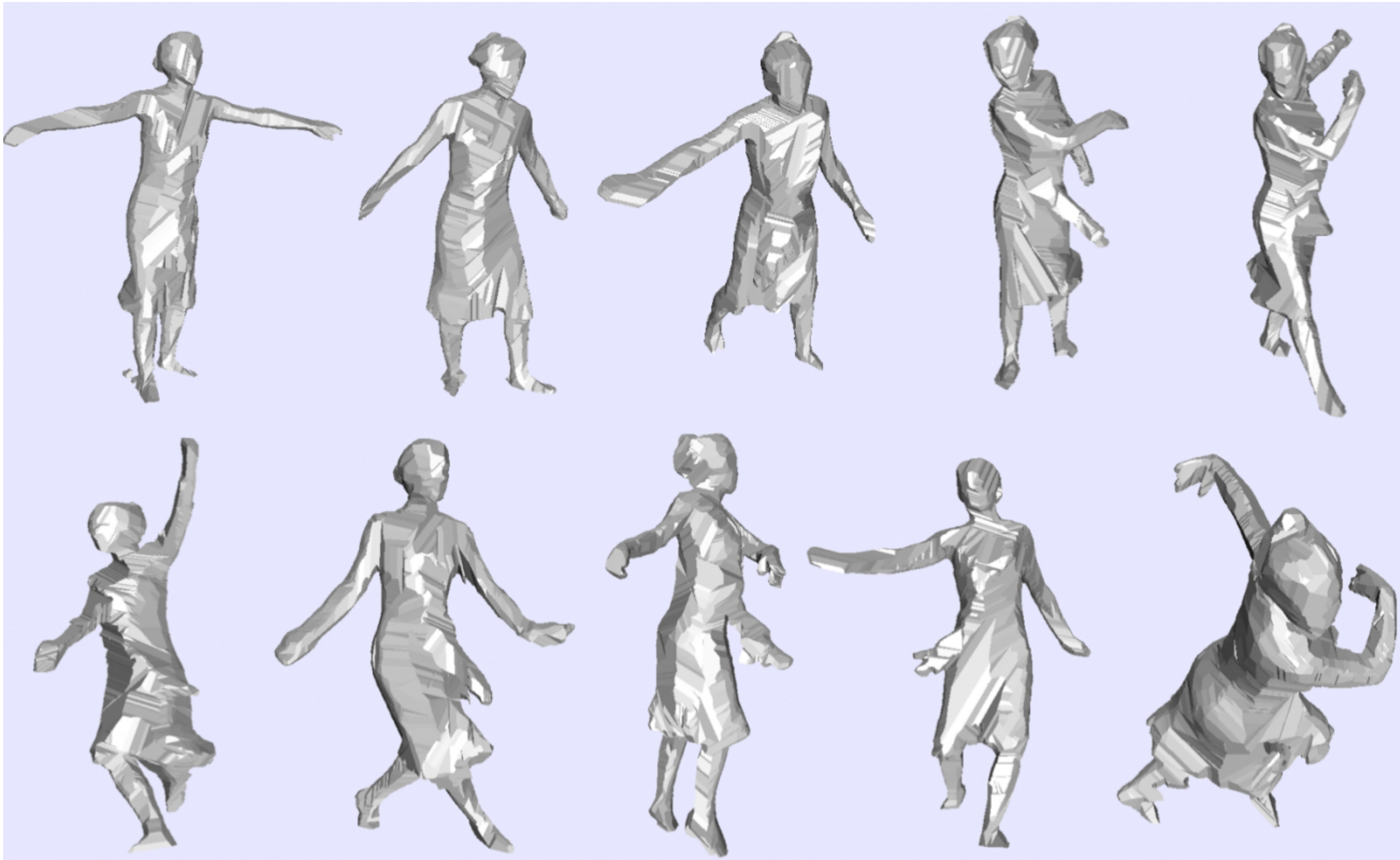
Voxel-based Algorithm



Color voxel black if on silhouette in every image

- O(?), for M images, N^3 voxels
- Don't have to search 2^{N^3} possible scenes!

Results (Franco and Boyer, PAMI 2009)



Properties of Volume Intersection

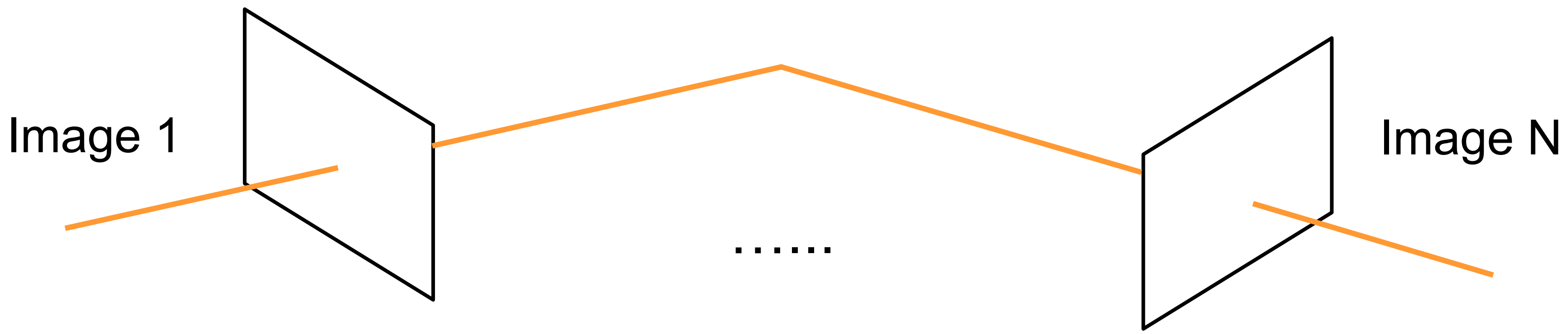
Pros

- Easy to implement, fast
- Accelerated via octrees

Cons

- No concavities
- Reconstruction is not photo-consistent
- Requires identification of silhouettes

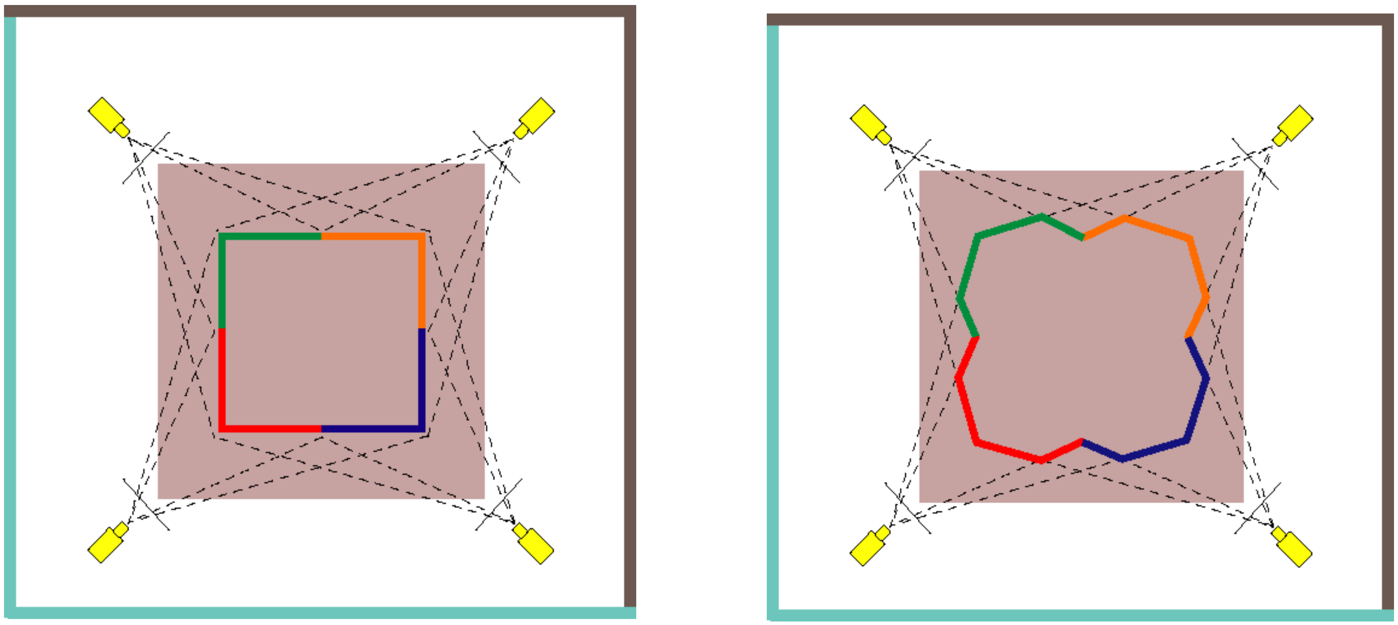
Space Carving



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

Which Shape do You Get?

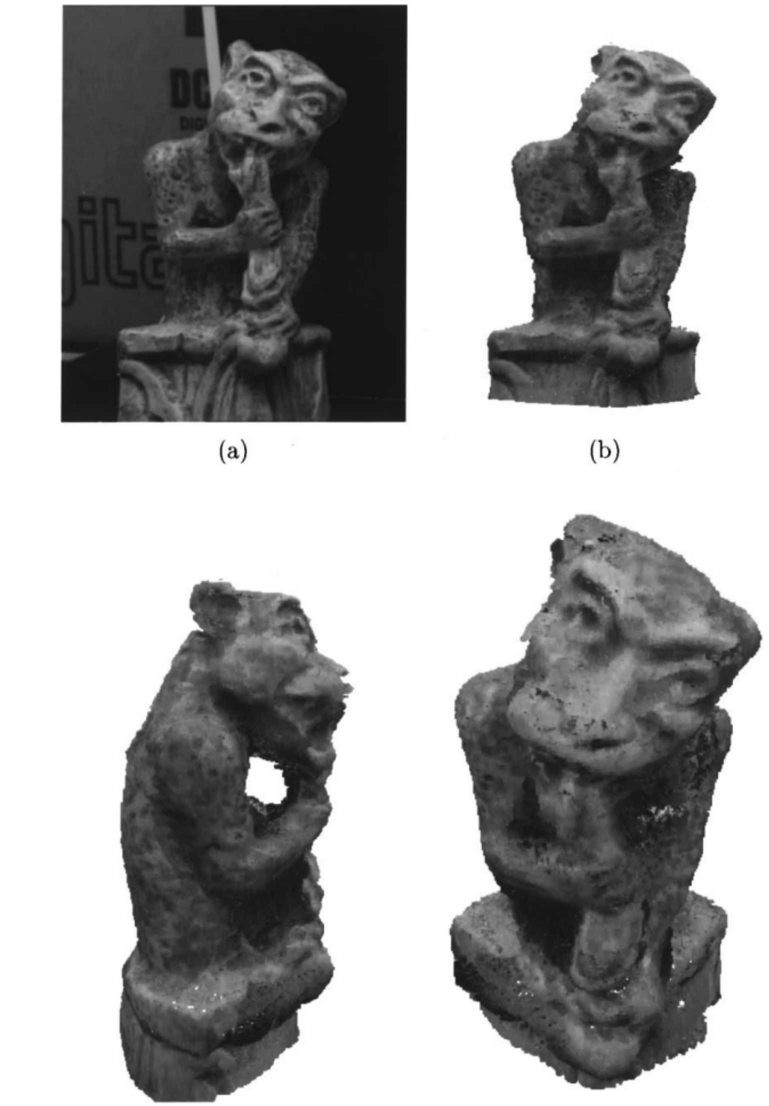


True Scene **Photo Hull**

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

Results (Kutulakos and Seitz, IJCV 2000)

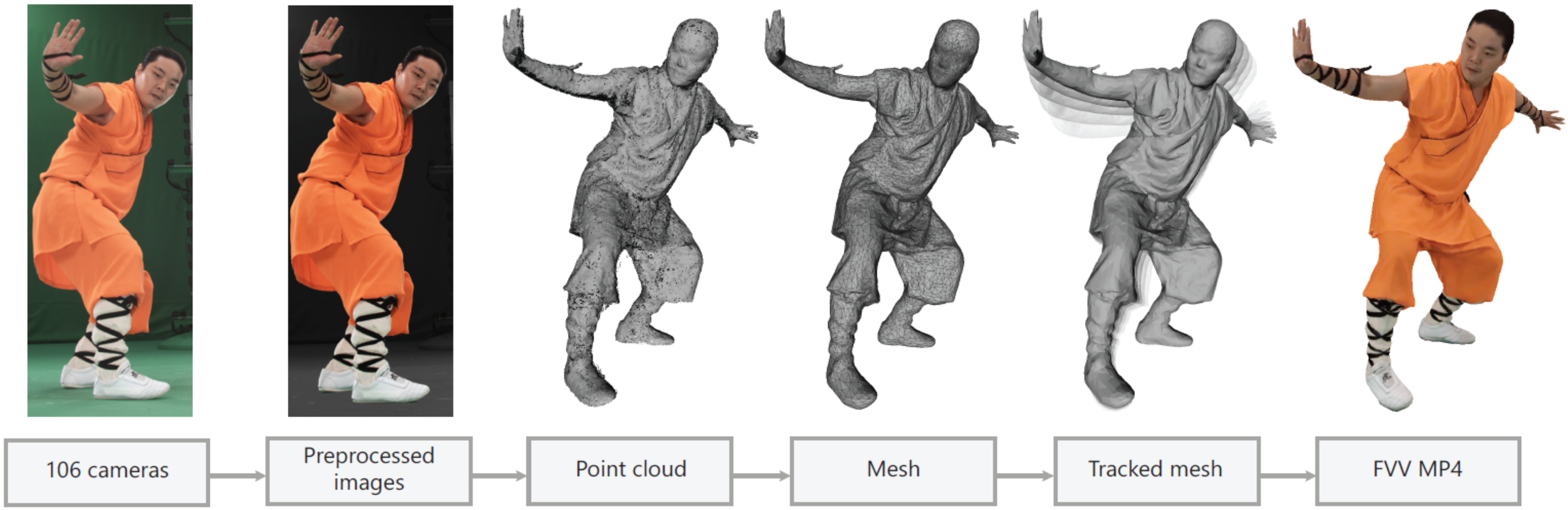


Free-Viewpoint Video

Alvaro Collet, Ming Chuang, Pat Sweeney,
Don Gillett, Dennis Evseev, David Calabrese,
Hugues Hoppe, Adam Kirk, Steve Sullivan
Microsoft Corporation

SIGGRAPH 2015

Pipeline



Capture + Preprocessing
- Calibration
- Bias correction
- Background subtraction

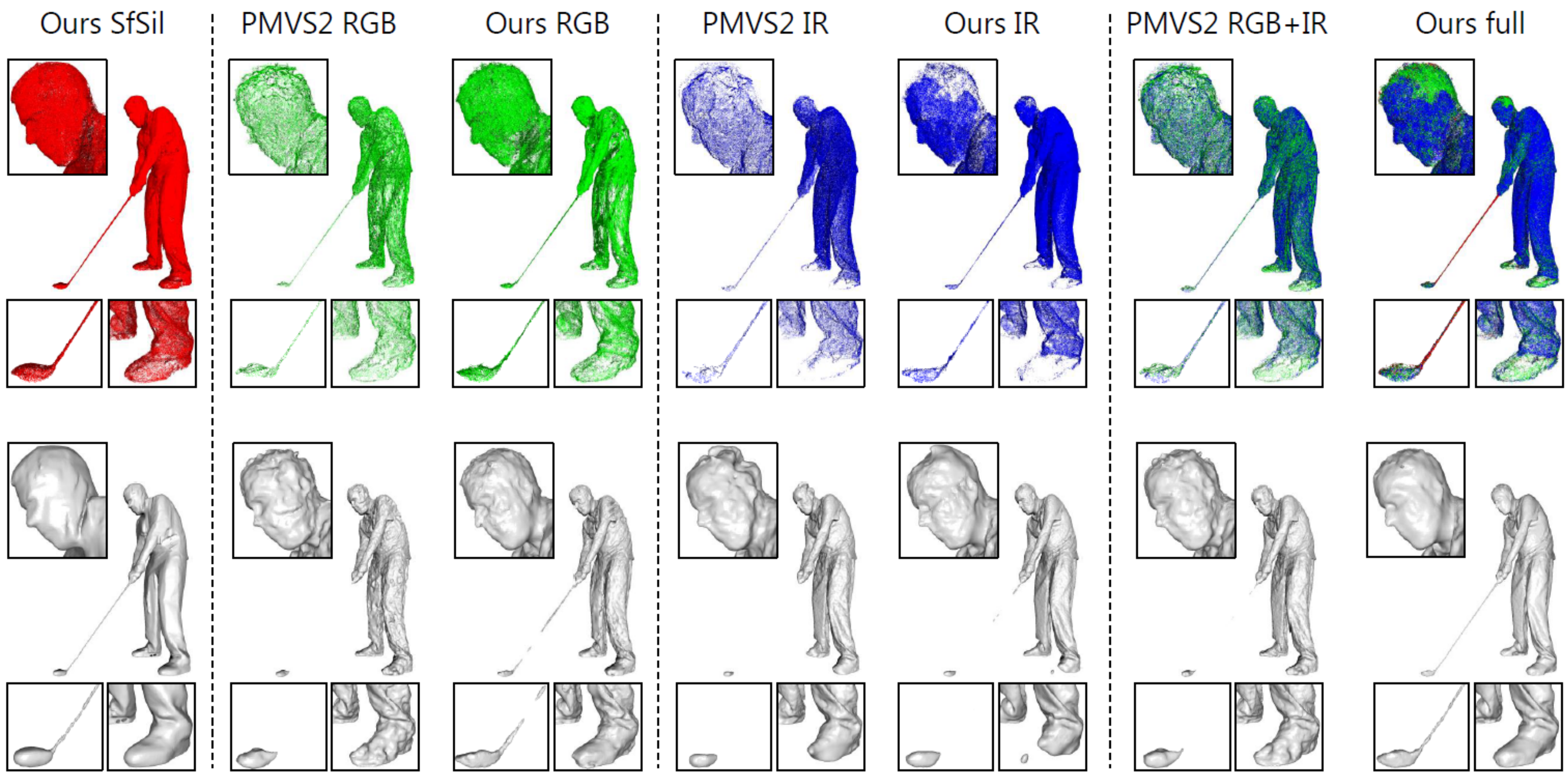
Point Generation
- Depth maps (IR, RGB, SfS)
- Multimodal MVS
- Iterative surface estimation (point-based)

Meshing + Texturing
- Topology denoising
- Island removal
- Hull-constrained PSR
- Occlusion detection
- Texturing

Temporal Mesh Processing
- Mesh tracking
- Importance detection
- Mesh decimation
- Texture atlas generation

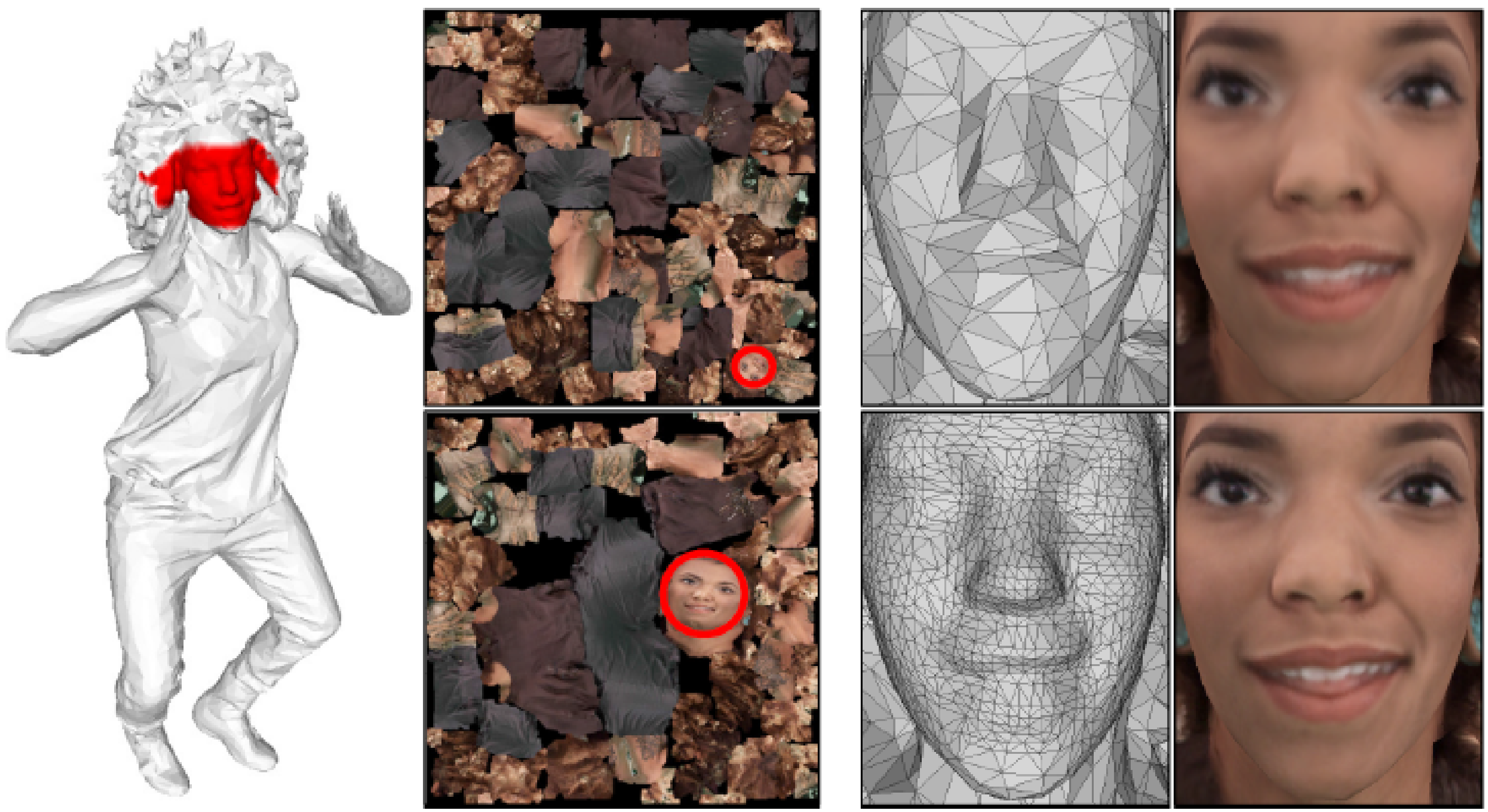
Compression + Encoding
- Video encode H.264
- Audio encode
- Mesh compression

Reconstruction



PMVS: system by Furukawa and Ponce, we saw last week
PSR: Poisson Surface Reconstruction (last slide last week)

Adaptive Level of Detail



Synthesized Viewpoints

Synthesized Viewpoints