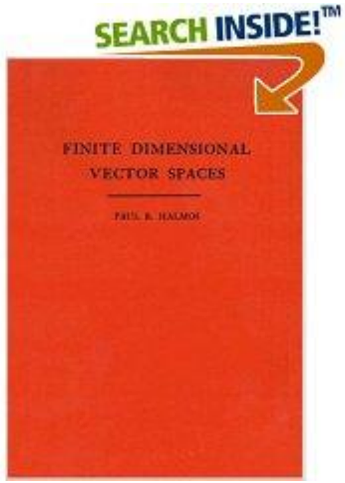
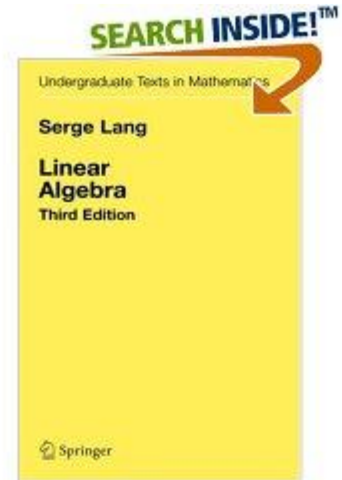
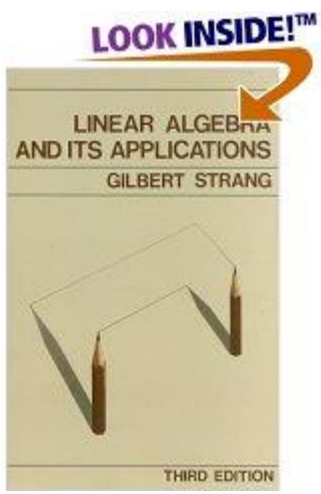
Some books on linear algebra



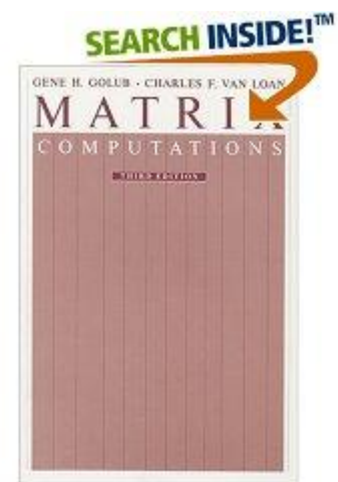
Finite Dimensional Vector Spaces, Paul R. Halmos, 1947



Linear Algebra, Serge Lang, 2004



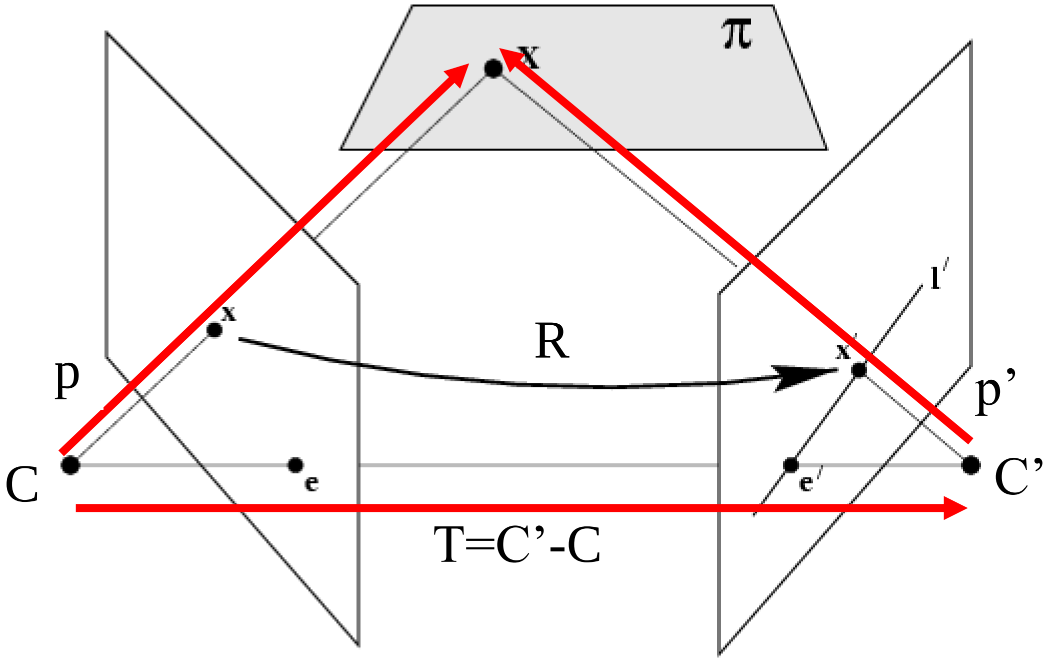
Linear Algebra and its Applications,
Gilbert Strang, 1988



Matrix Computation, Gene H. Golub,
Charles F. Van Loan, 1996

Last lecture

- 2-Frame Structure from Motion
- Multi-Frame Structure from Motion

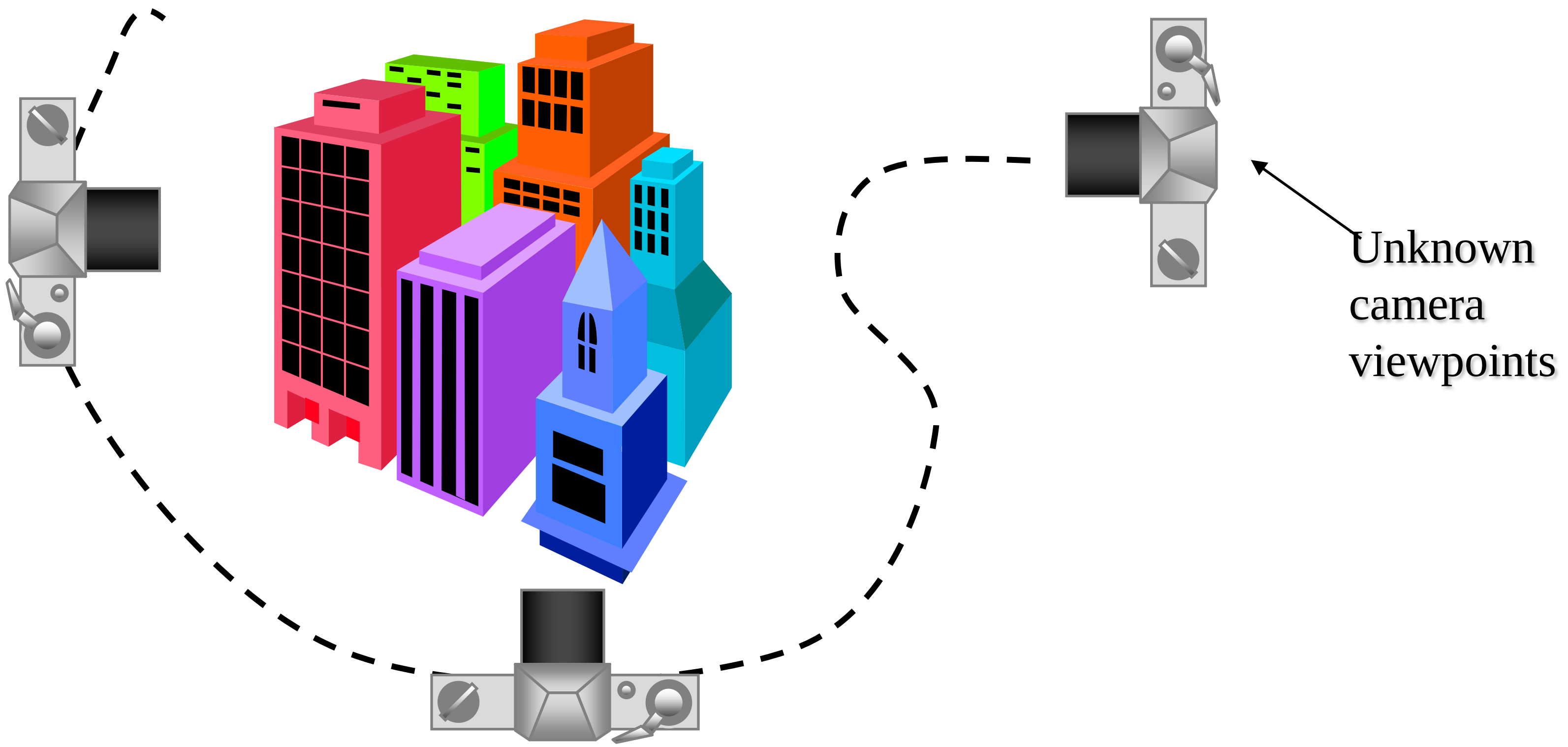


$$\mathbf{p}'^{T}\,\mathbf{E}\mathbf{p} = 0$$

$$\mathbf{x}'^{T}\,\mathbf{F}\mathbf{x} = 0$$

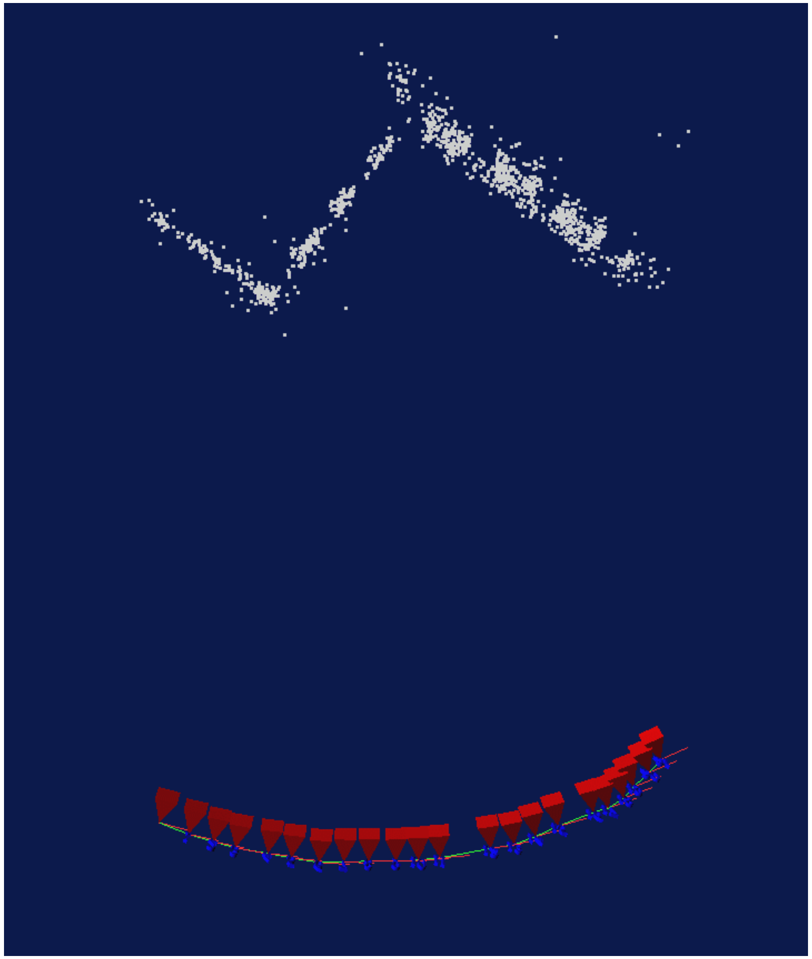
Today

- Continue on Multi-Frame Structure from Motion:
- Multi-View Stereo



Structure from Motion by Factorization

Problem statement



SFM under orthographic projection

2D image point — orthographic projection matrix — 3D scene point — Camera center

$$\mathbf{q} = \mathbf{\Pi}(\mathbf{p} - \mathbf{t})$$

$$2\times1 \quad 2\times3 \quad 3\times1 \quad 3\times1$$

For example, $\mathbf{\Pi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

In general, $\mathbf{\Pi} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}$ subject to $\mathbf{\Pi}\mathbf{\Pi}^{\mathrm{T}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

SFM under orthographic projection

2D image point

orthographic projection matrix

3D scene point

Camera center

$$\underset{2\times 1}{\mathbf{q}_n} = \underset{2\times 3}{\Pi}(\underset{3\times 1}{\mathbf{p}_n} - \underset{3\times 1}{\mathbf{t}})$$

- Choose scene origin to be the centroid of the 3D points

$$\sum \mathbf{p}_n = 0$$

- Choose image origin to be the centroid of the 2D points

$$\sum \mathbf{q}_n = 0$$

$$\mathbf{q} = \Pi \mathbf{p}$$

factorization (Tomasi & Kanade)

projection of *n* features in one image:

$$\underset{2\times n}{\begin{bmatrix}\mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n\end{bmatrix}} = \underset{2\times 3}{\prod} \underset{3\times n}{\begin{bmatrix}\mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n\end{bmatrix}}$$

projection of *n* features in *m* images

$$\underset{2m\times n}{\begin{bmatrix}\mathbf{q}_{11} & \mathbf{q}_{12} & \cdots & \mathbf{q}_{1n} \\ \mathbf{q}_{21} & \mathbf{q}_{22} & \cdots & \mathbf{q}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}_{m1} & \mathbf{q}_{m2} & \cdots & \mathbf{q}_{mn}\end{bmatrix}} = \underset{2m\times 3}{\begin{bmatrix}\mathbf{\Pi}_1 \\ \mathbf{\Pi}_2 \\ \vdots \\ \mathbf{\Pi}_m\end{bmatrix}} \underset{3\times n}{\begin{bmatrix}\mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_n\end{bmatrix}}$$

W measurement M motion S shape

Key Observation: *rank*(W) <= 3

Factorization

known

$$\underset{2m\times n}{\mathbf{W}} = \underset{2m\times 3}{\mathbf{M}}\ \underset{3\times n}{\mathbf{S}}$$

solve for

- Factorization Technique
  - W is at most rank 3 (assuming no noise)
  - We can use *singular value decomposition* to factor W:

$$\underset{2m\times n}{\mathbf{W}} = \underset{2m\times 3}{\mathbf{M}'}\ \underset{3\times n}{\mathbf{S}'}$$

  - S' differs from S by a linear transformation *A*:

$$\mathbf{W} = \mathbf{M}'\mathbf{S}' = (\mathbf{M}\mathbf{A}^{-1})(\mathbf{A}\mathbf{S})$$

  - Solve for A by enforcing *metric* constraints on *M*

Metric constraints

- Enforcing “Metric” Constraints
  - Compute **A** such that rows of **M** have these properties

$$\begin{bmatrix} \Pi_1 \\ \Pi_2 \\ \vdots \\ \Pi_m \end{bmatrix} = \mathbf{M} = \mathbf{M}'\mathbf{A} = \begin{bmatrix} \Pi_1' \\ \Pi_2' \\ \vdots \\ \Pi_m' \end{bmatrix} \mathbf{A}$$

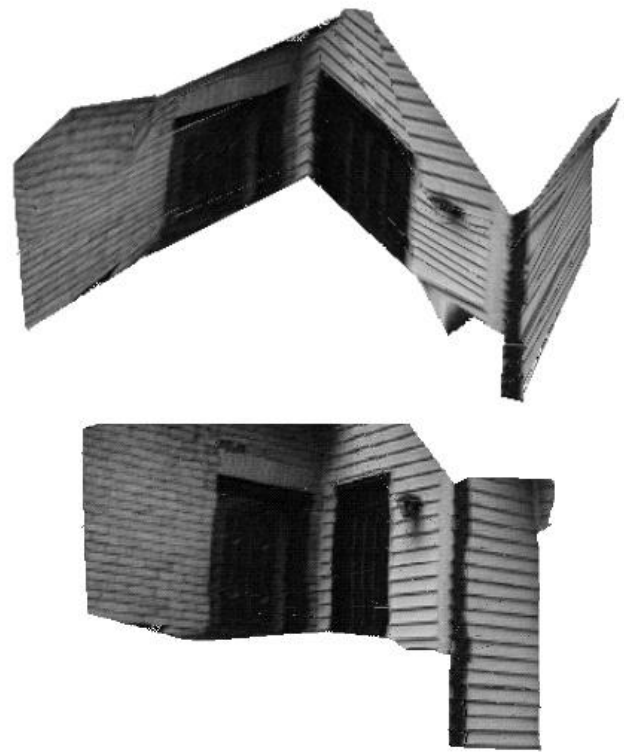
Trick (not in original Tomasi/Kanade paper, but in followup work)

- Constraints are linear in AA^T :

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \Pi_i \Pi_i^T = \Pi'_i \mathbf{A}\left(\Pi'^T_i \mathbf{A}\right)^T = \Pi'_i \mathbf{G} \Pi'^T_i \qquad \textit{where } \mathbf{G} = \mathbf{A}\mathbf{A}^T$$

- Solve for G first by writing equations for every Π_i in M
- Then $G = AA^T$ by SVD

Results



Extensions to factorization methods

- Paraperspective [Poelman & Kanade, PAMI 97]
- Sequential Factorization [Morita & Kanade, PAMI 97]
- Factorization under perspective [Christy & Horaud, PAMI 96] [Sturm & Triggs, ECCV 96]
- Factorization with Uncertainty [Anandan & Irani, IJCV 2002]

Perspective Bundle adjustment

Bundle Adjustment

$$\hat{u}_{ij} = f(\mathbf{K}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{x}_i)$$
$$\hat{v}_{ij} = g(\mathbf{K}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{x}_i)$$

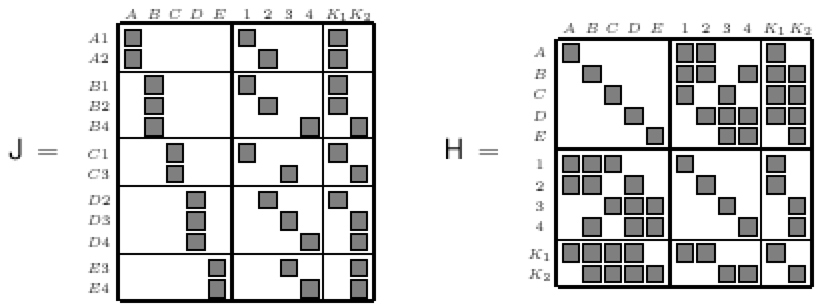
- How to initialize?
  - 2 or 3 views at a time, add more iteratively **[Hartley 00]**
- What makes this non-linear minimization hard?
  - many more parameters: potentially slow
  - poorer conditioning (high correlation)
  - potentially lots of outliers

Lots of parameters: sparsity

$$\hat{u}_{ij} = f(\mathbf{K}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{x}_i)$$
$$\hat{v}_{ij} = g(\mathbf{K}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{x}_i)$$

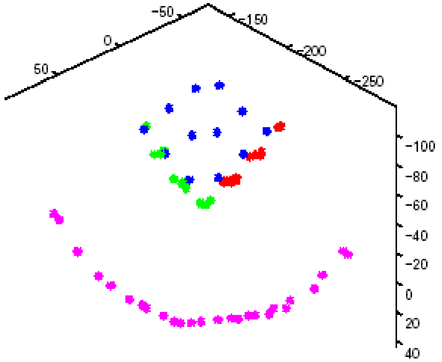
- Only a few entries in Jacobian are non-zero

$$\frac{\partial \hat{u}_{ij}}{\partial \mathbf{K}}, \quad \frac{\partial \hat{u}_{ij}}{\partial \mathbf{R}_j}, \quad \frac{\partial \hat{u}_{ij}}{\partial \mathbf{t}_j}, \quad \frac{\partial \hat{u}_{ij}}{\partial \mathbf{x}_i},$$

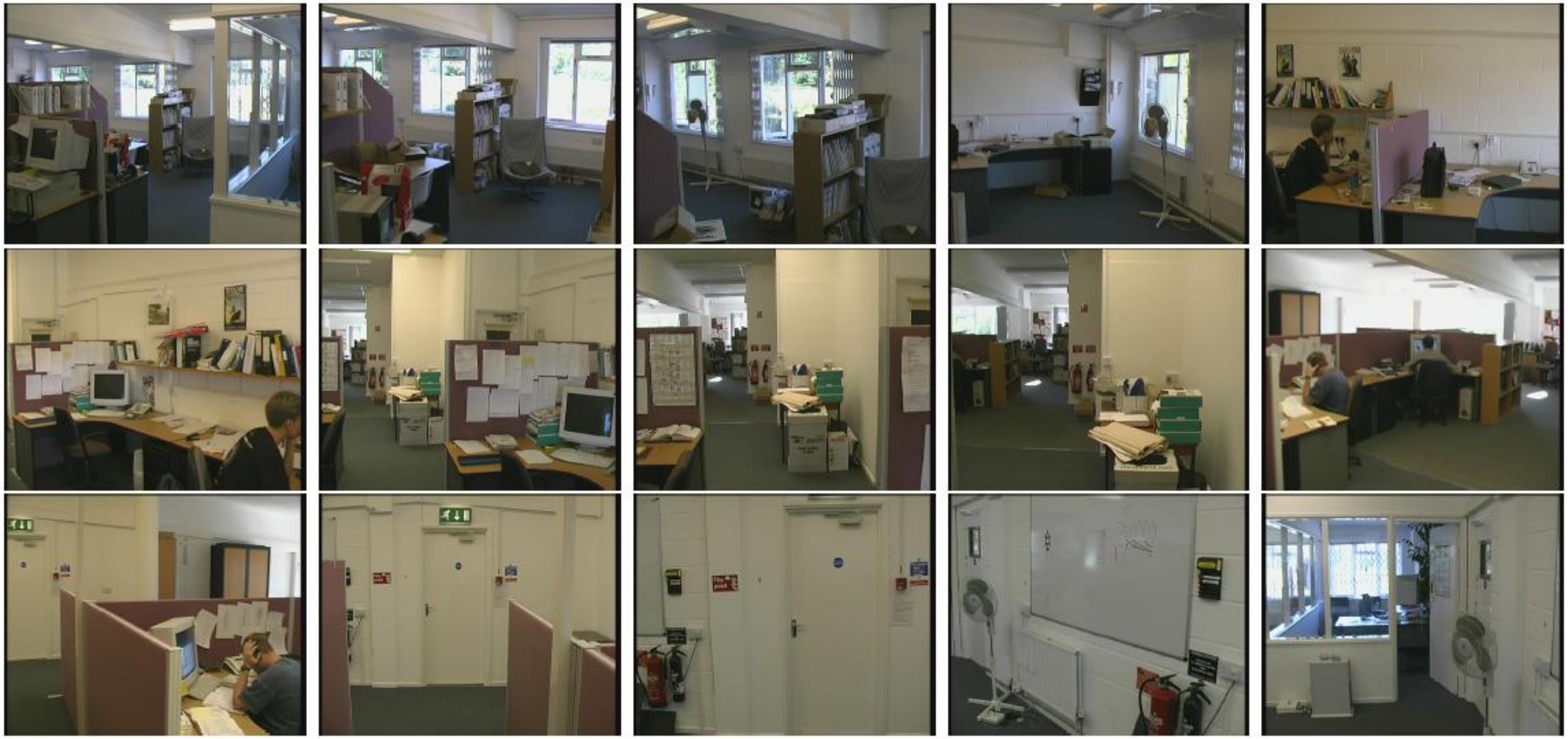


Structure from motion: limitations

- Very difficult to reliably estimate <u>*metric*</u> structure and motion unless:
  - large (*x* or *y*) rotation *or*
  - large field of view and depth variation
- Camera calibration important for Euclidean reconstructions
- Need good feature tracker
- Lens distortion

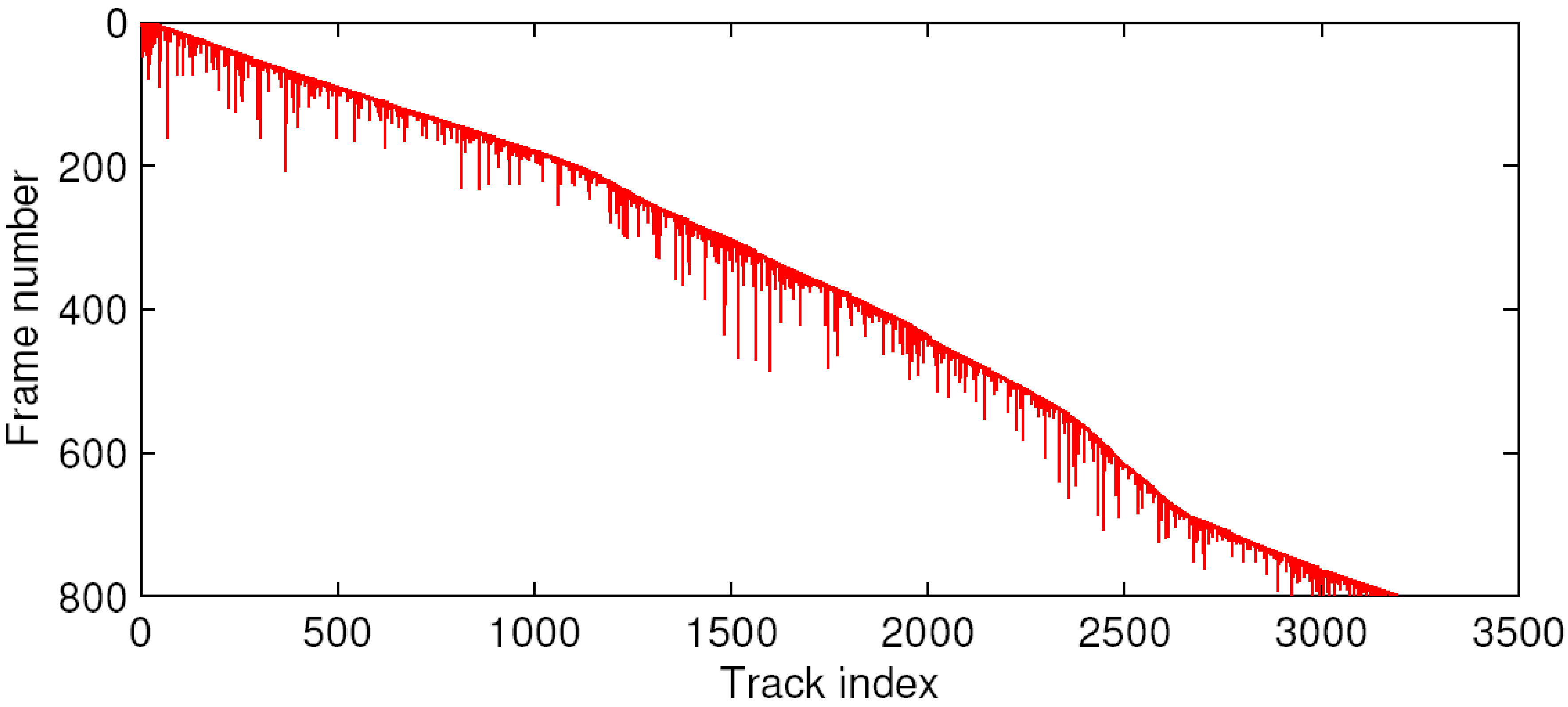


Track lifetime



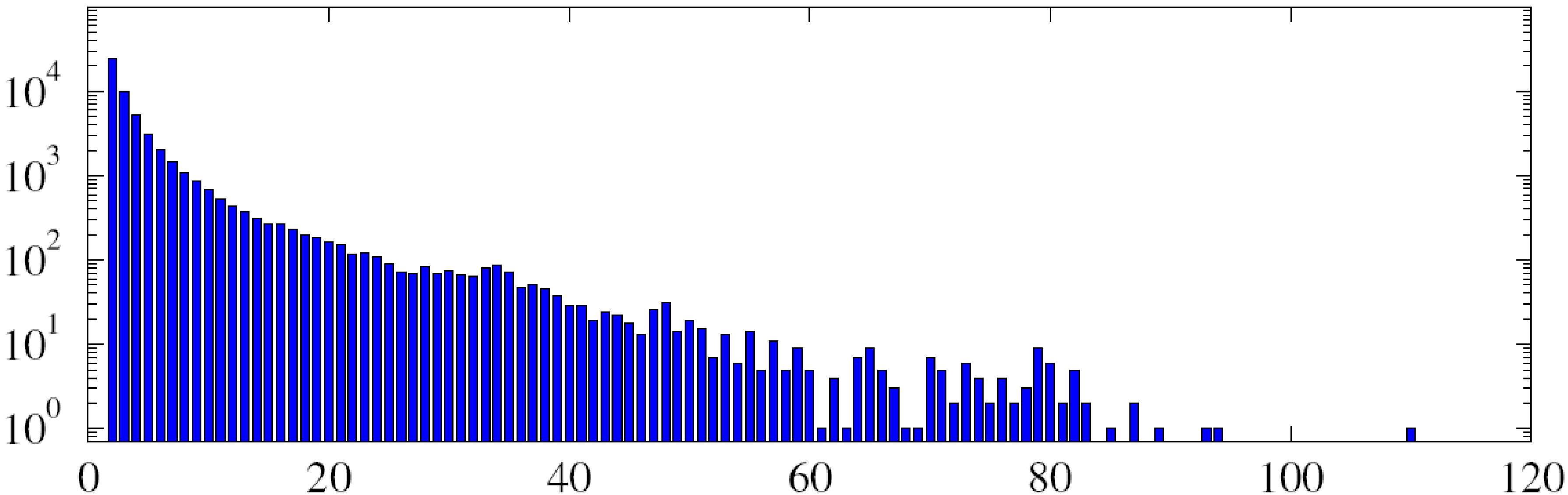
every 50th frame of a 800-frame sequence

Track lifetime



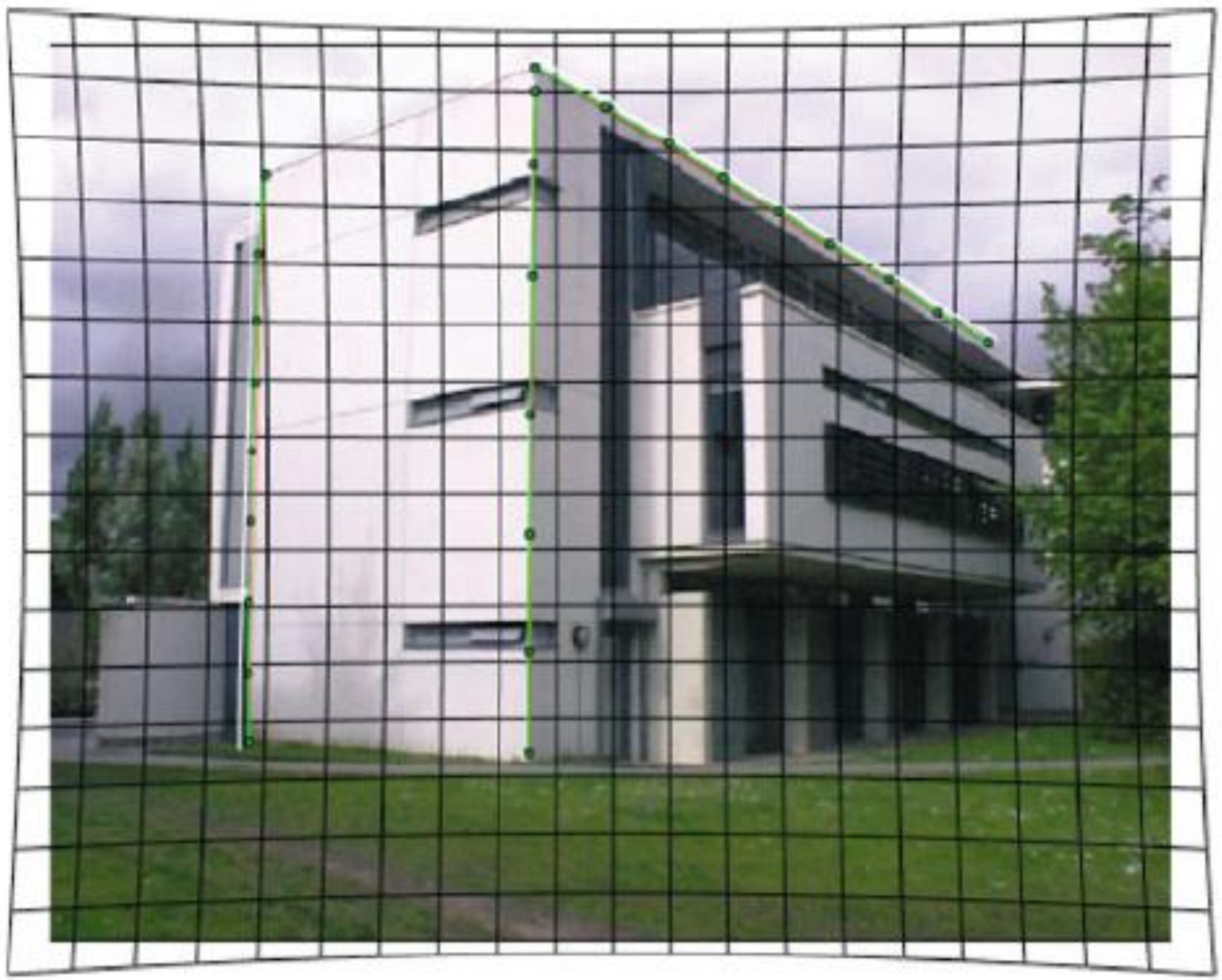
lifetime of 3192 tracks from the previous sequence

Track lifetime

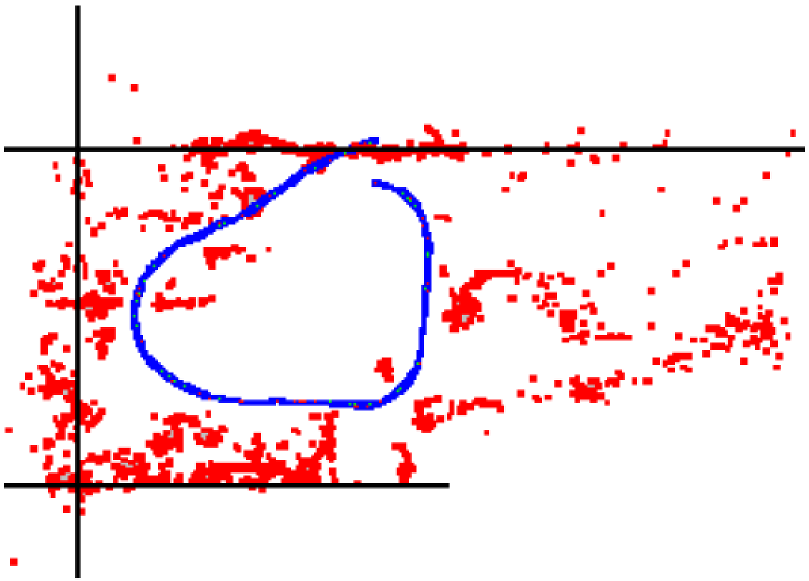


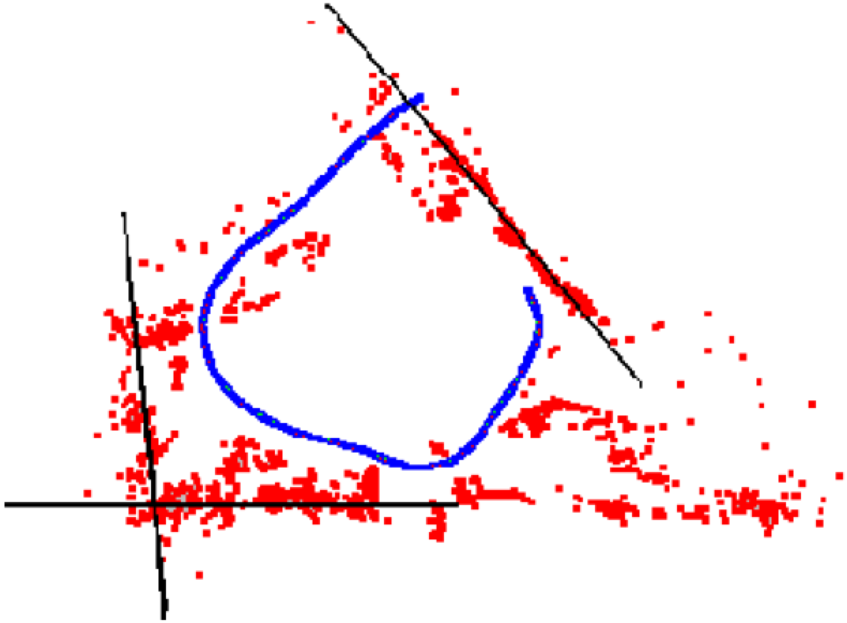
track length histogram

Nonlinear lens distortion



Nonlinear lens distortion



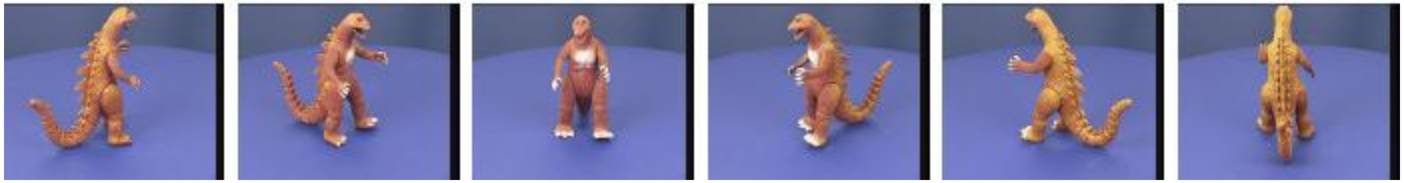


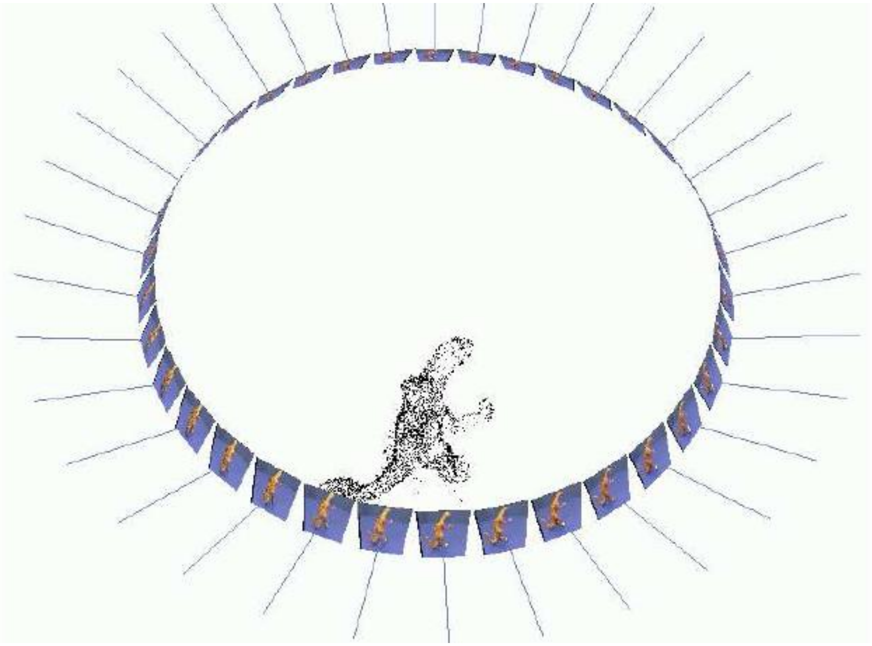
effect of lens distortion

Prior knowledge and scene constraints

add a constraint that several lines are parallel

Prior knowledge and scene constraints





add a constraint that it is a turntable sequence

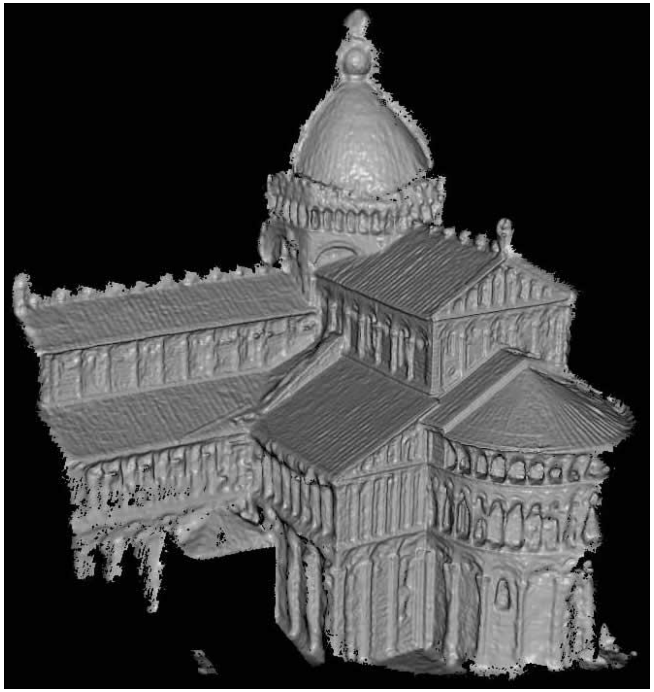
Applications of Structure from Motion

Jurassic park

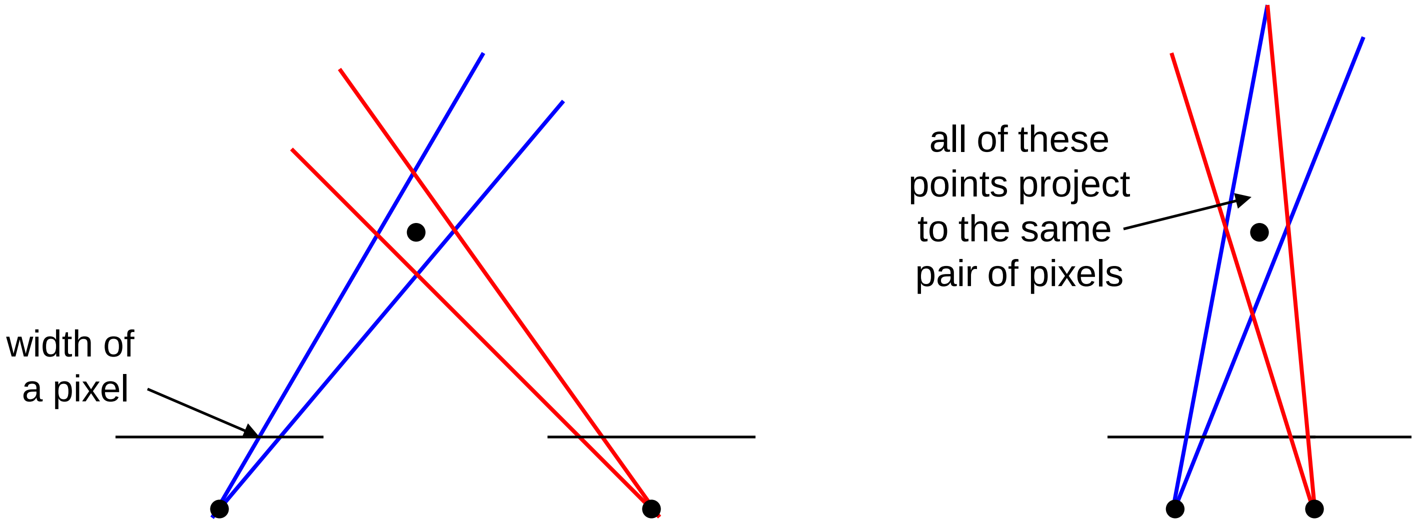
PhotoSynth

http://labs.live.com/photosynth/

Multiview Stereo



Choosing the stereo baseline



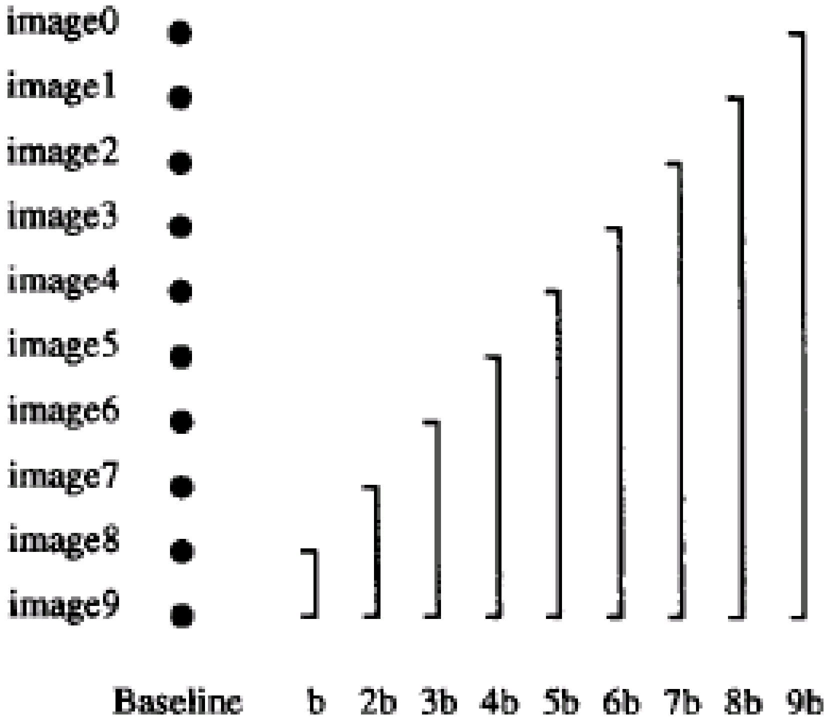
Large Baseline **Small Baseline**

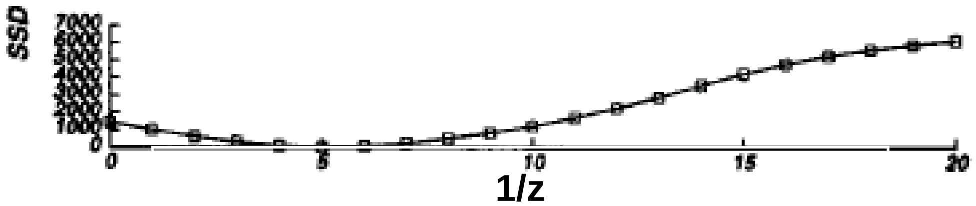
What's the optimal baseline?

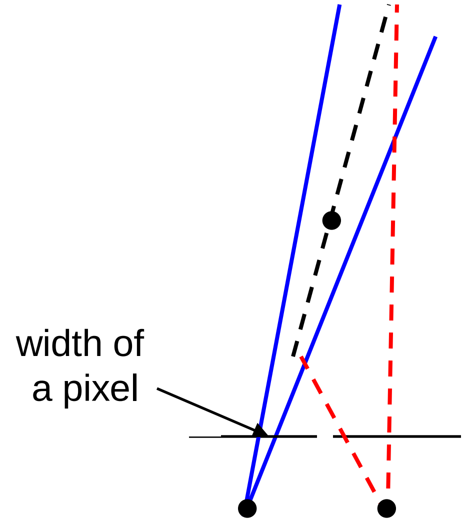
- Too small: large depth error
- Too large: difficult search problem

The Effect of Baseline on Depth Estimation

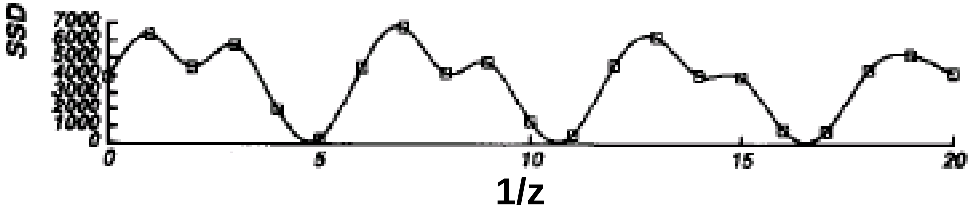
Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.

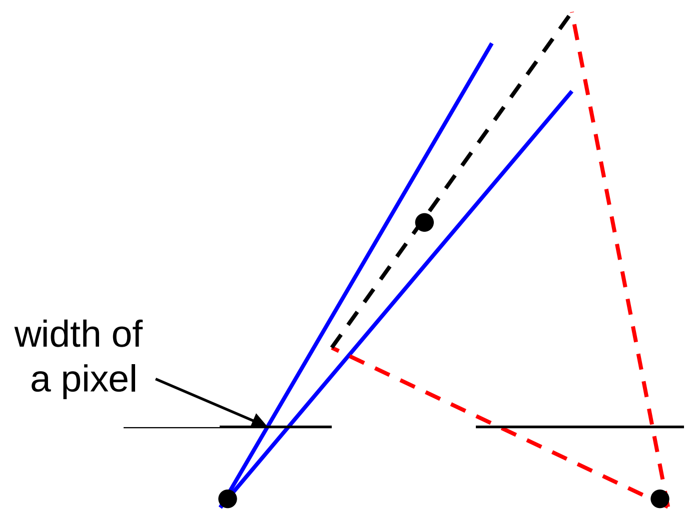






pixel matching score





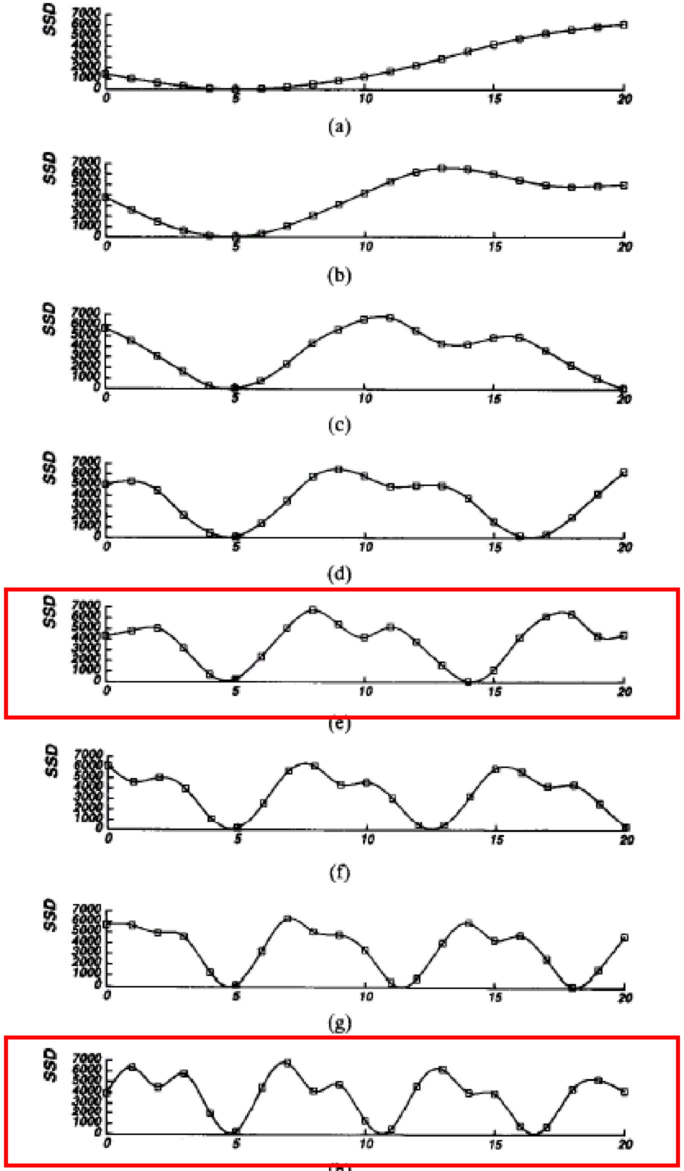


Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

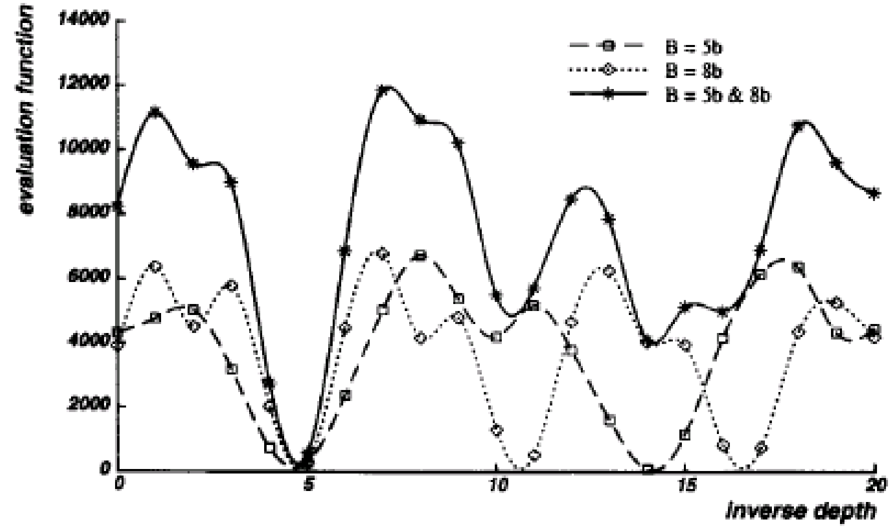


Fig. 6. Combining two stereo pairs with different baselines.

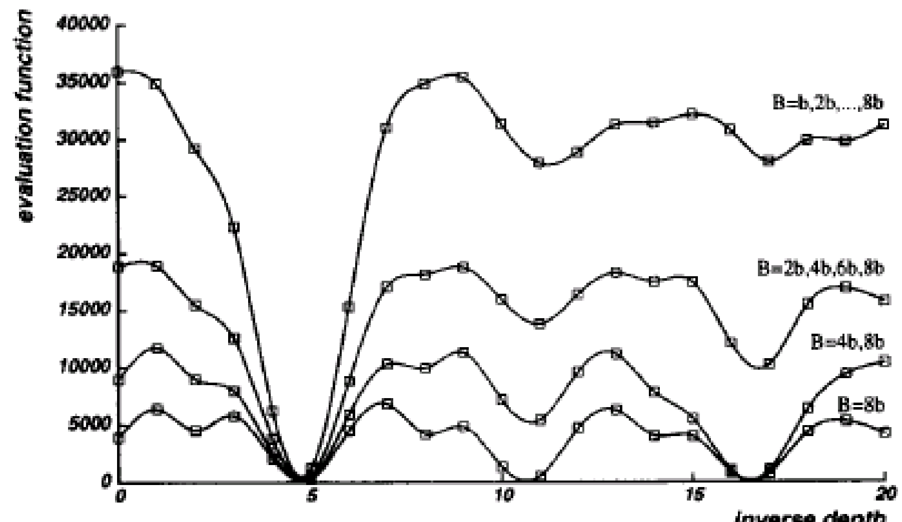


Fig. 7. Combining multiple baseline stereo pairs.

Multibaseline Stereo

Basic Approach

- Choose a reference view
- Use your favorite stereo algorithm BUT
  - replace two-view SSD with SSD over all baselines

Limitations

- Must choose a reference view (bad)
- Visibility!

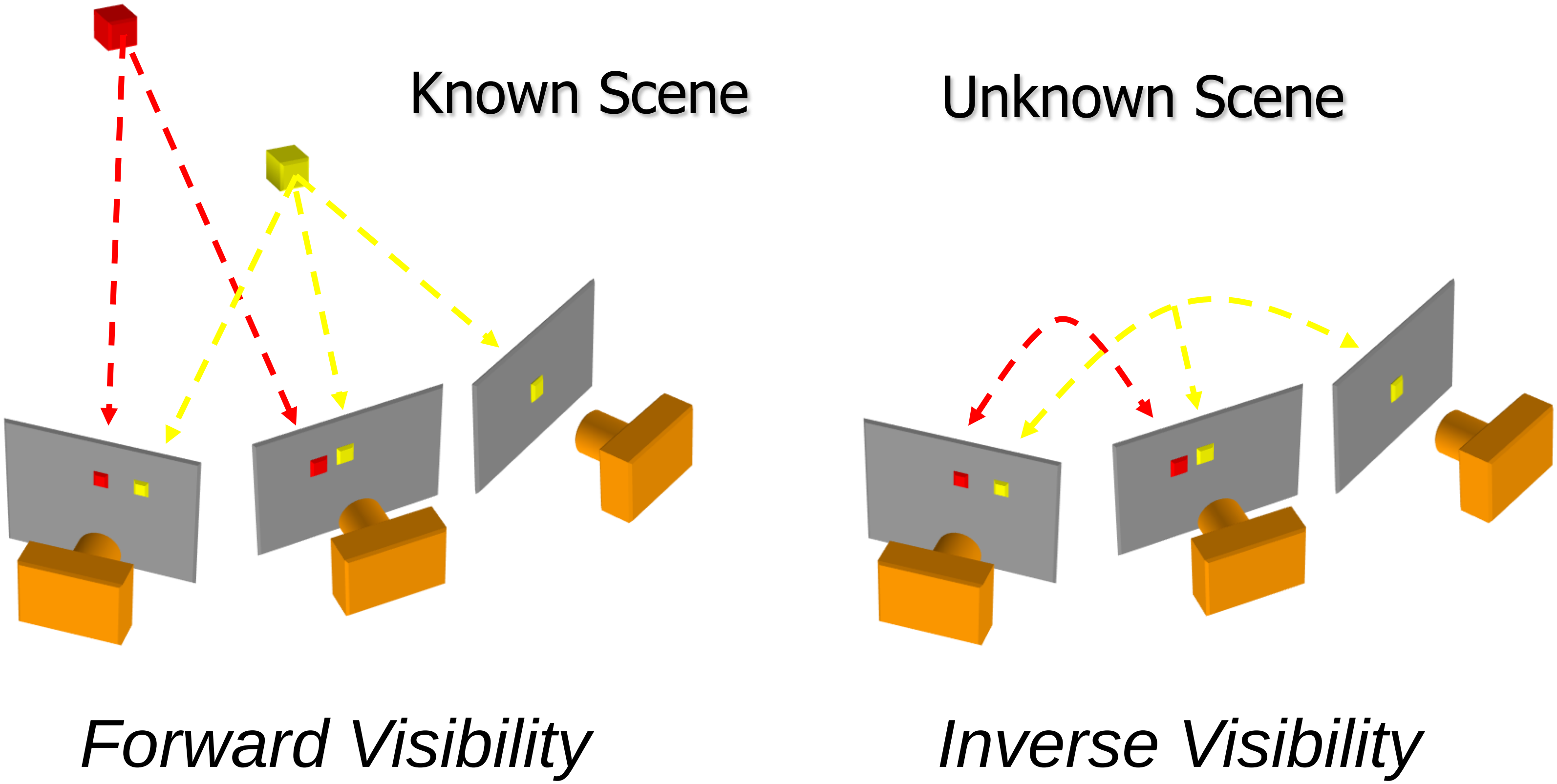
MSR Image based Reality Project

http://research.microsoft.com/~larryz/videoviewinterpolation.htm

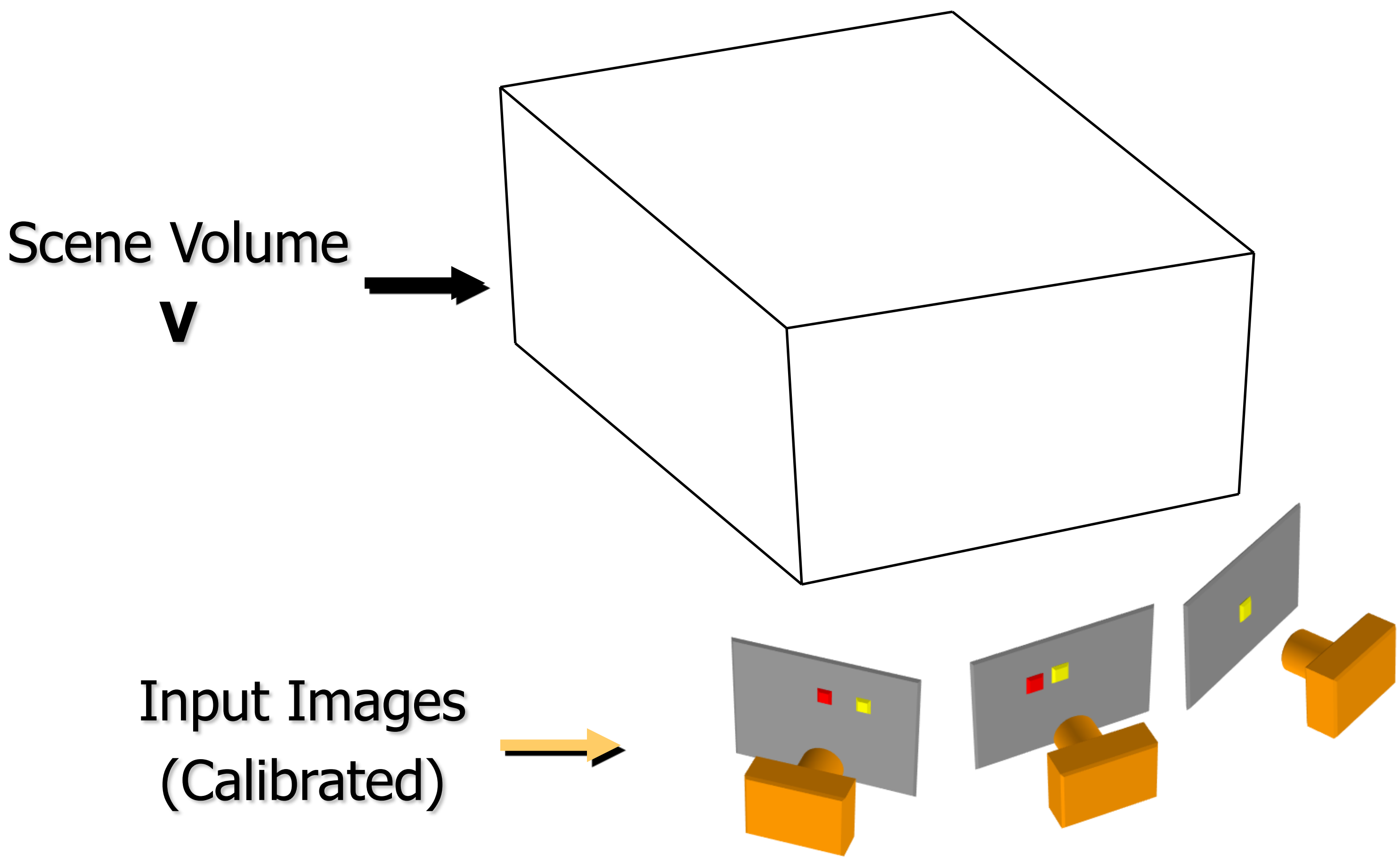
  … 

The visibility problem

Which points are visible in which images?



Volumetric stereo

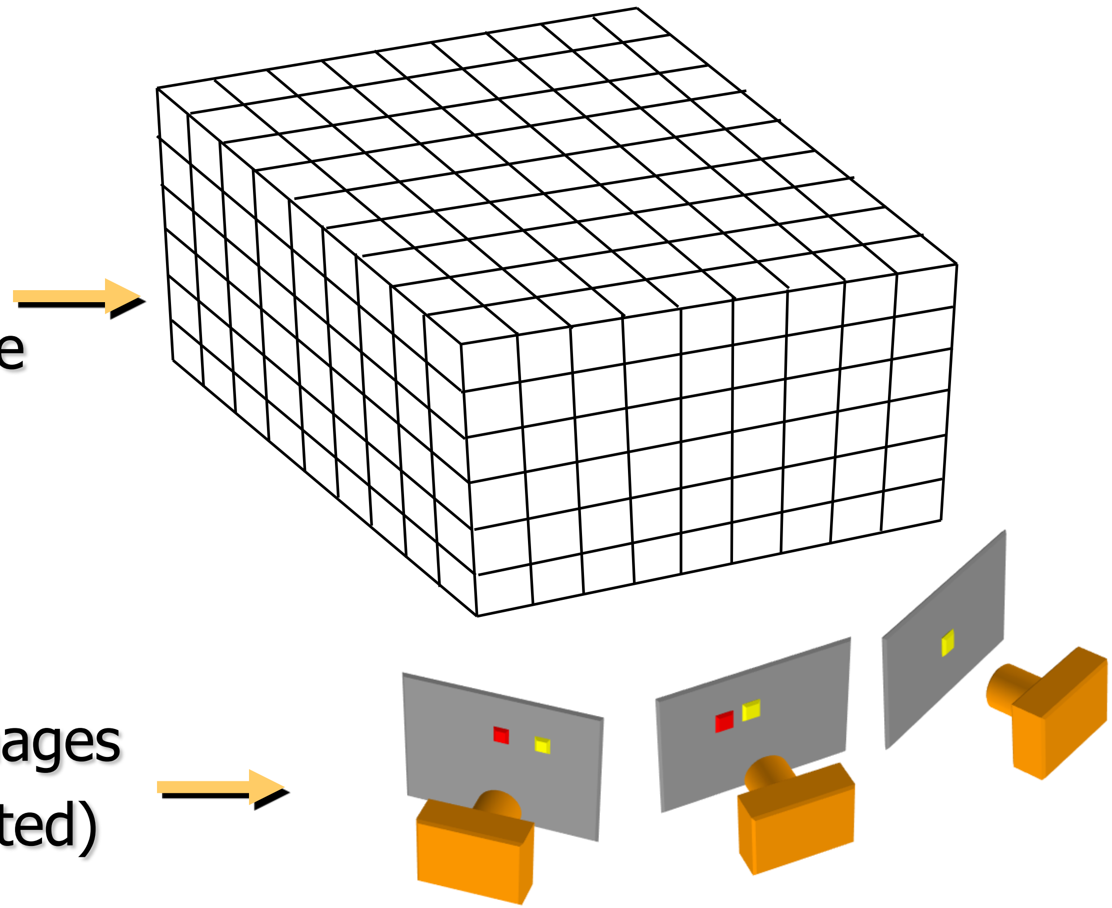


Goal: **Determine occupancy, "color" of points in V**

Discrete formulation: Voxel Coloring

Discretized Scene Volume

Input Images (Calibrated)



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

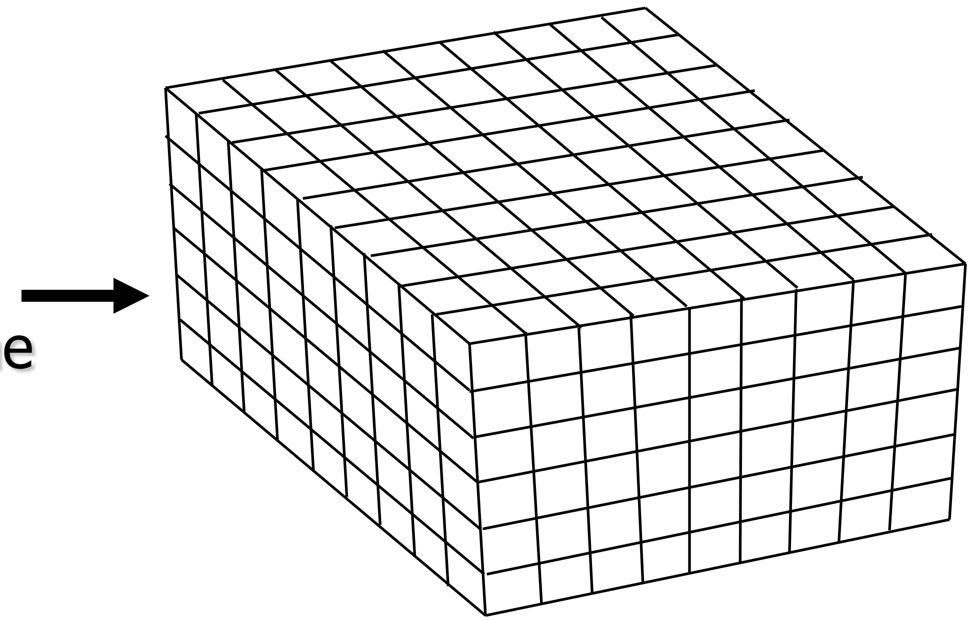
Complexity and computability

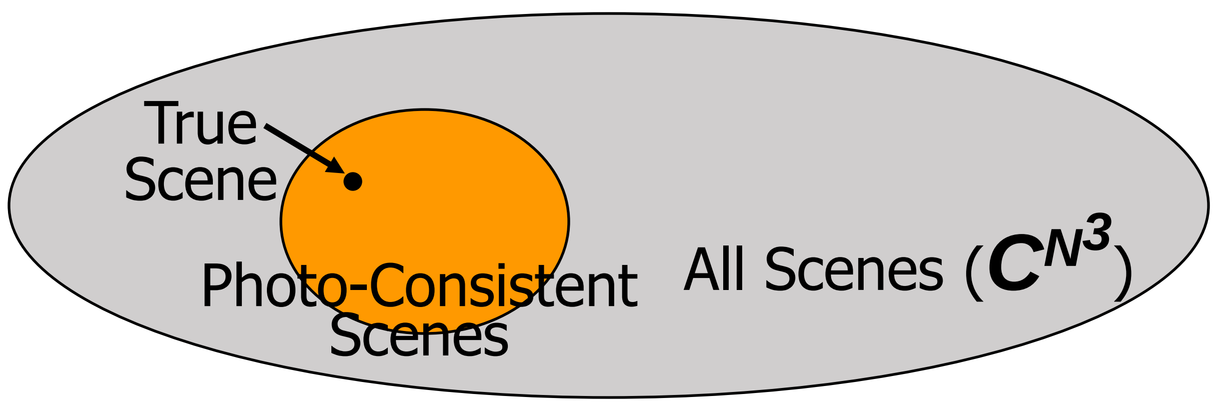
Discretized

Scene Volume

N^3 voxels

C colors





Issues

Theoretical Questions

- Identify class of *all* photo-consistent scenes

Practical Questions

- How do we compute photo-consistent models?

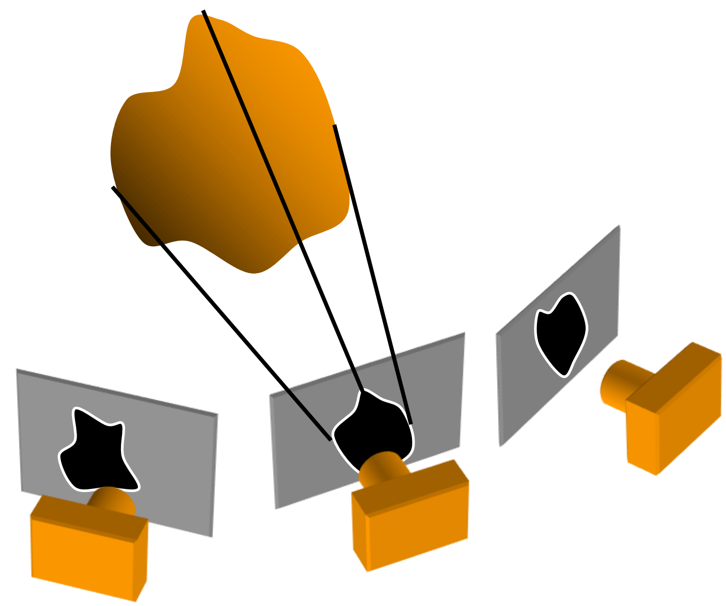
Voxel coloring solutions

1. C=2 (shape from silhouettes)
   - Volume intersection [Baumgart 1974]
     > For more info: *Rapid octree construction from image sequences.* R. Szeliski, CVGIP: Image Understanding, 58(1):23-32, July 1993. (this paper is apparently not available online) or

     > W. Matusik, C. Buehler, R. Raskar, L. McMillan, and S. J. Gortler, *Image-Based Visual Hulls*, SIGGRAPH 2000 (pdf 1.6 MB)
2. C unconstrained, viewpoint constraints
   - Voxel coloring algorithm [Seitz & Dyer 97]
3. General Case
   - Space carving [Kutulakos & Seitz 98]

Reconstruction from Silhouettes (C = 2)

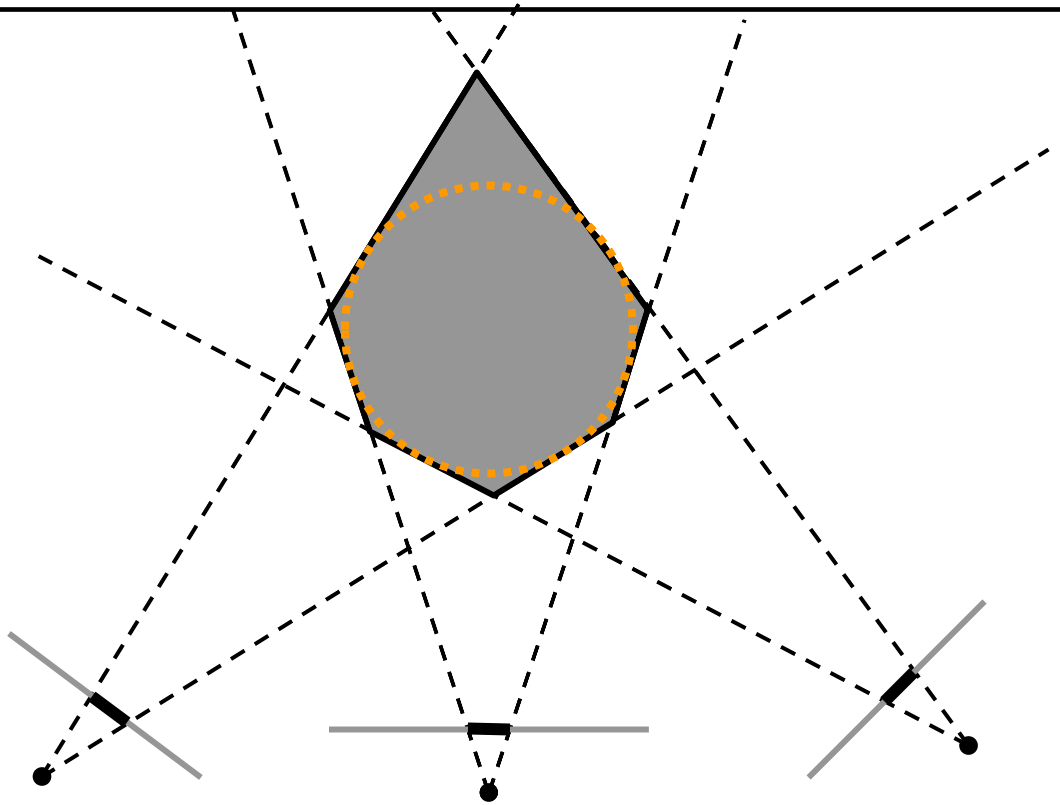
Binary Images →



Approach:

- *Backproject* each silhouette
- Intersect backprojected volumes

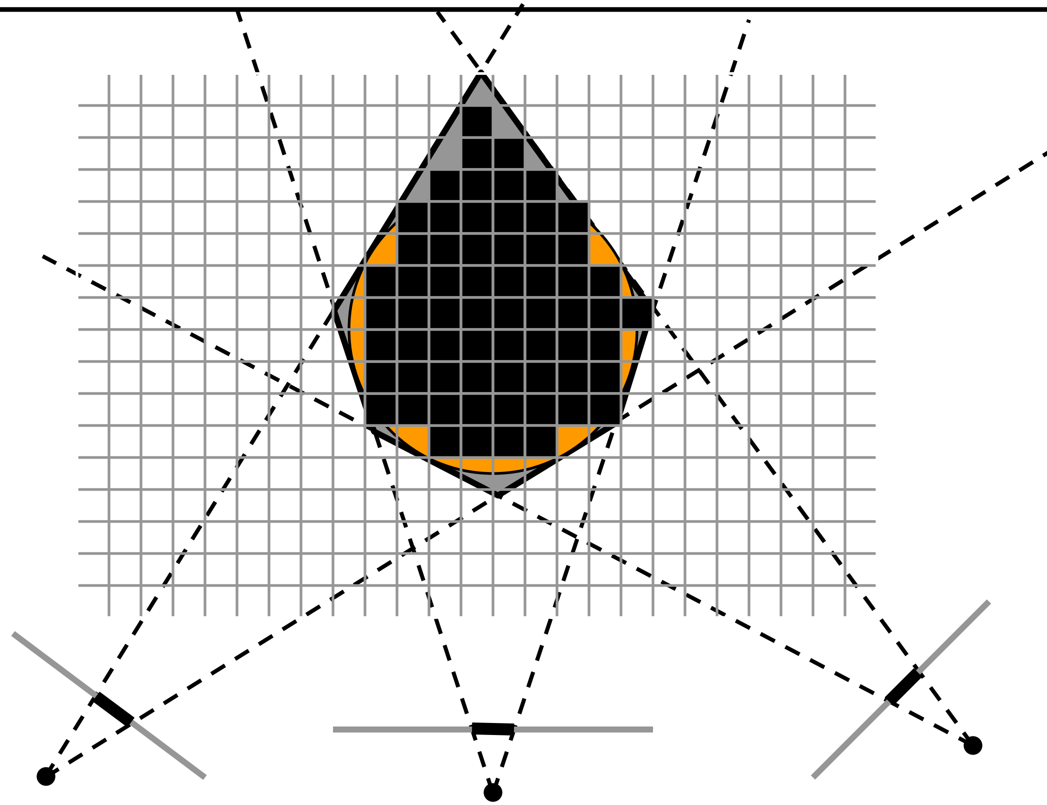
Volume intersection



Reconstruction Contains the True Scene

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

Voxel algorithm for volume intersection



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!

Properties of Volume Intersection

Pros

- Easy to implement, fast
- Accelerated via octrees [Szeliski 1993] or interval techniques [Matusik 2000]

Cons

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

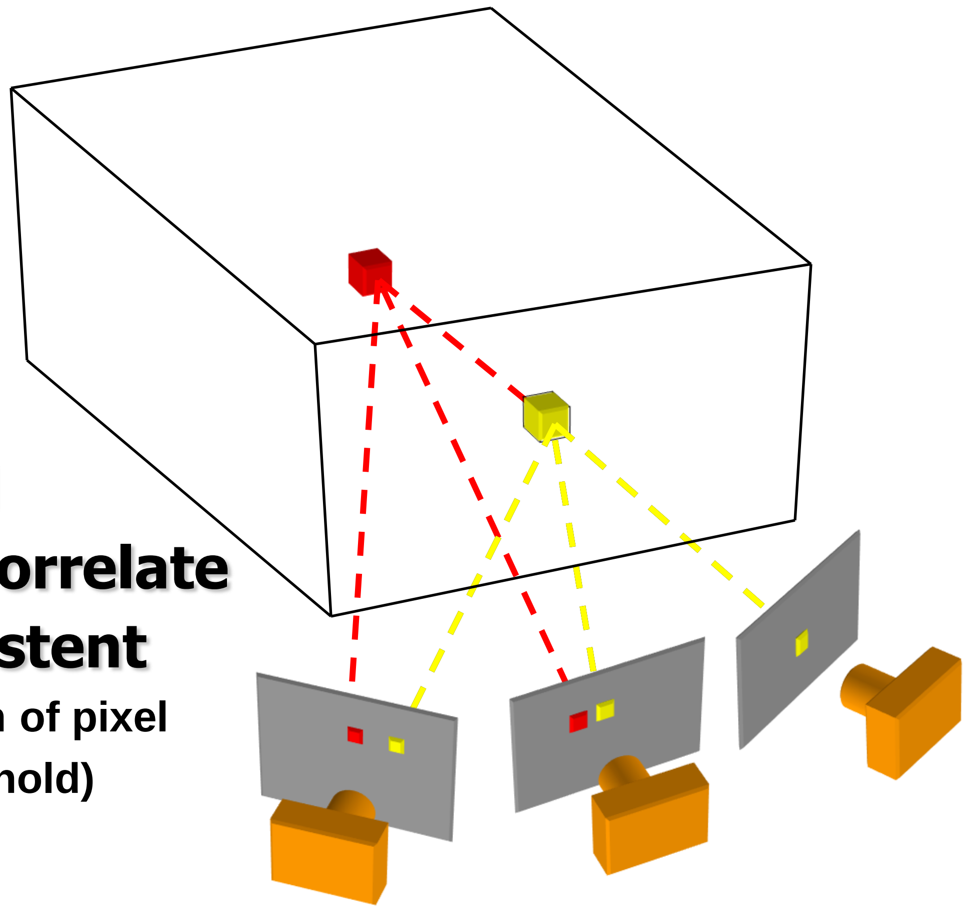
Voxel Coloring Solutions

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

2. C unconstrained, viewpoint constraints
   - Voxel coloring algorithm [Seitz & Dyer 97]
     - For more info: http://www.cs.washington.edu/homes/seitz/papers/ijcv99.pdf

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

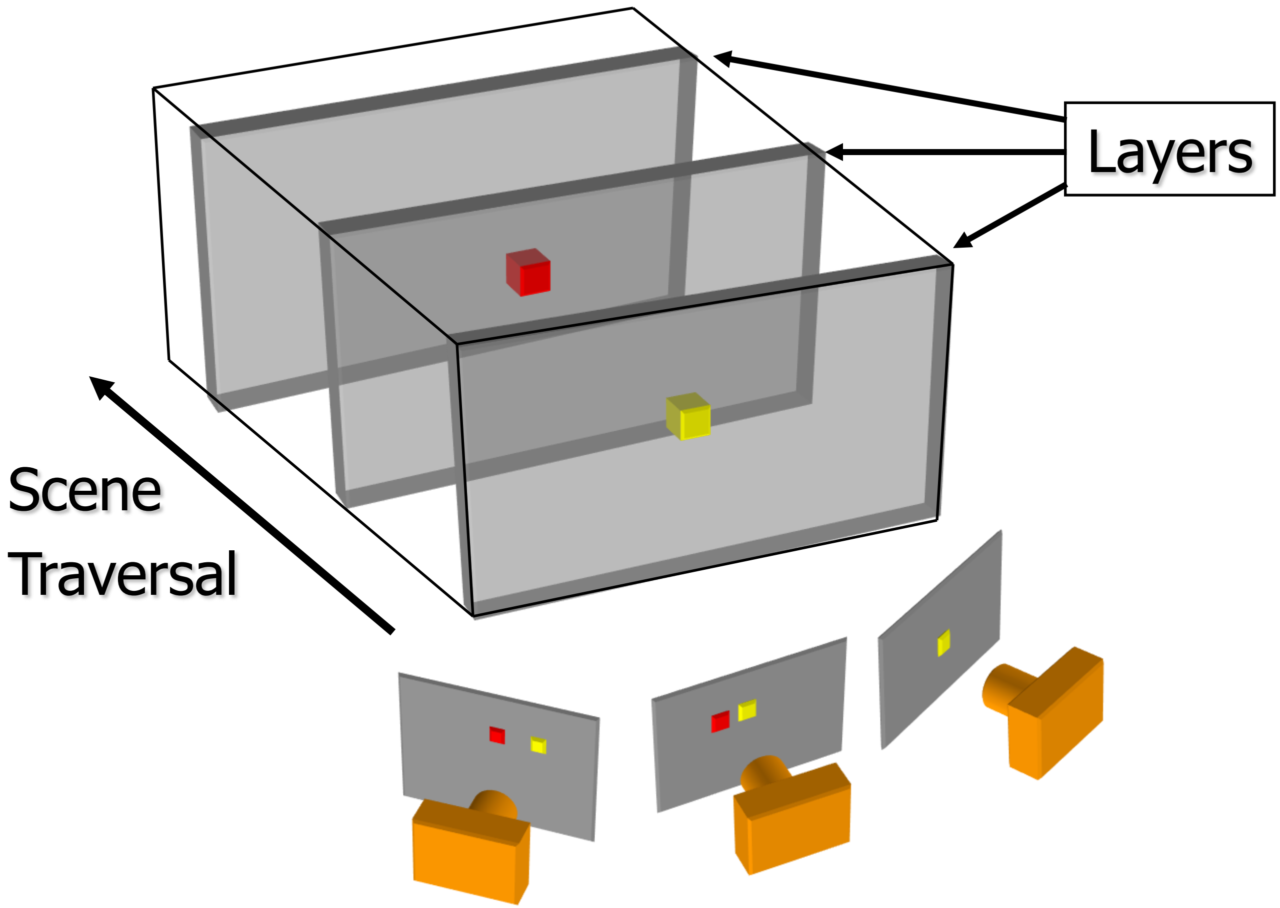
Voxel Coloring Approach



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

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

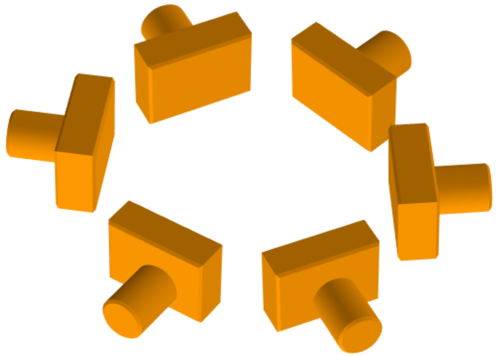
Depth Ordering: visit occluders first!



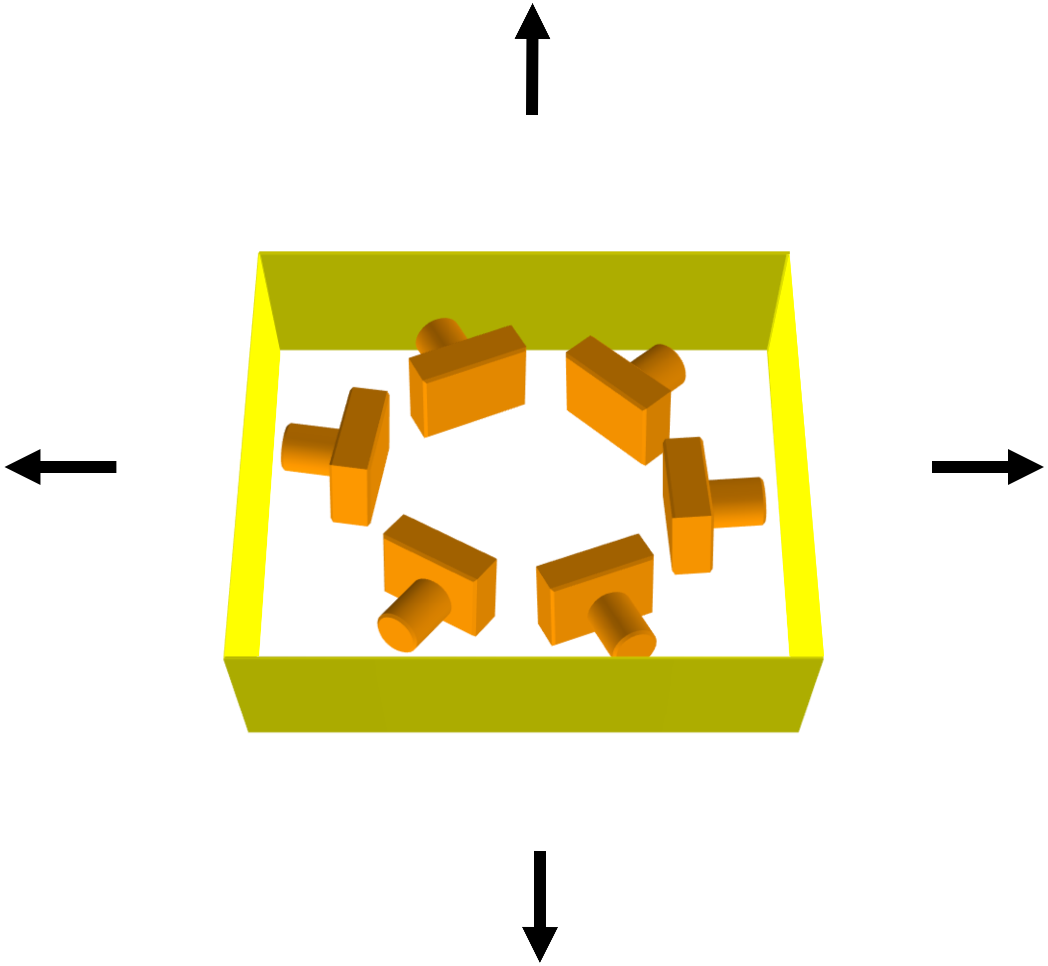
Condition: **depth order is the** ***same for all input views***

Panoramic Depth Ordering

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

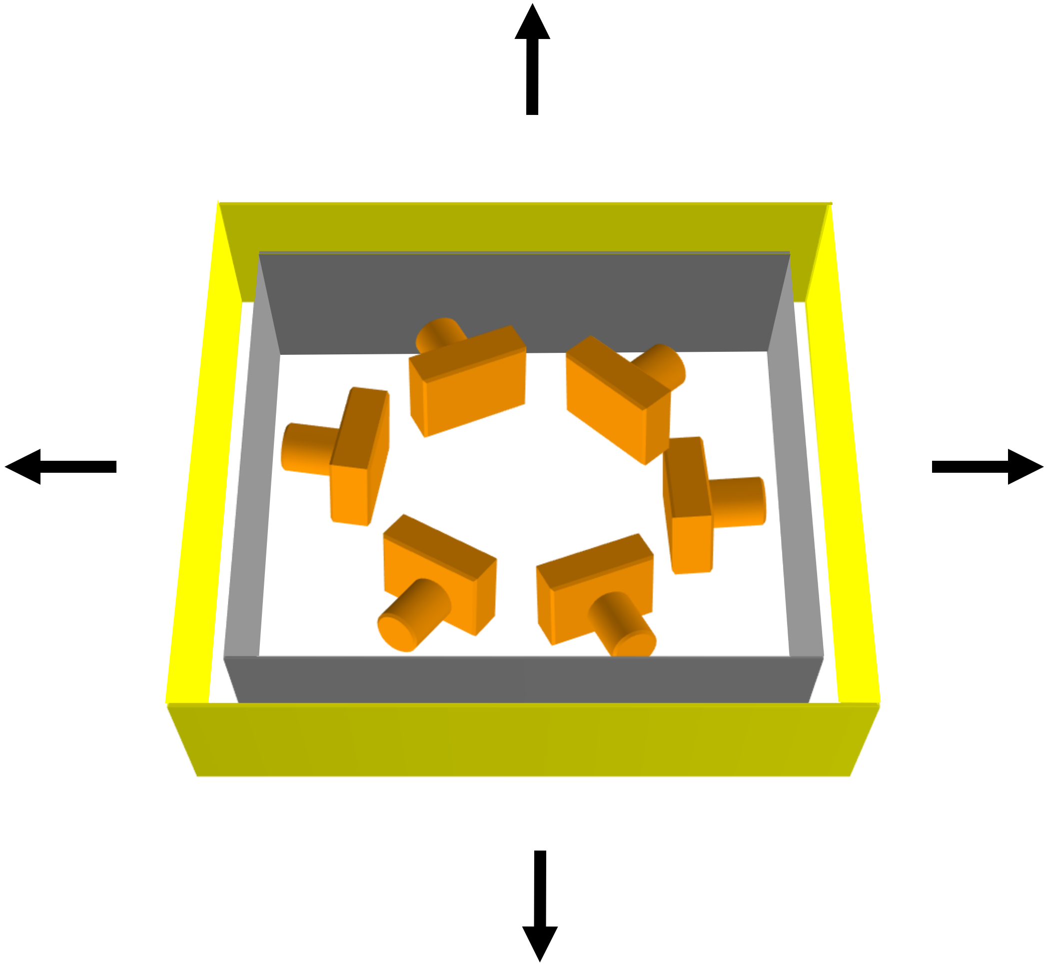


Panoramic Depth Ordering



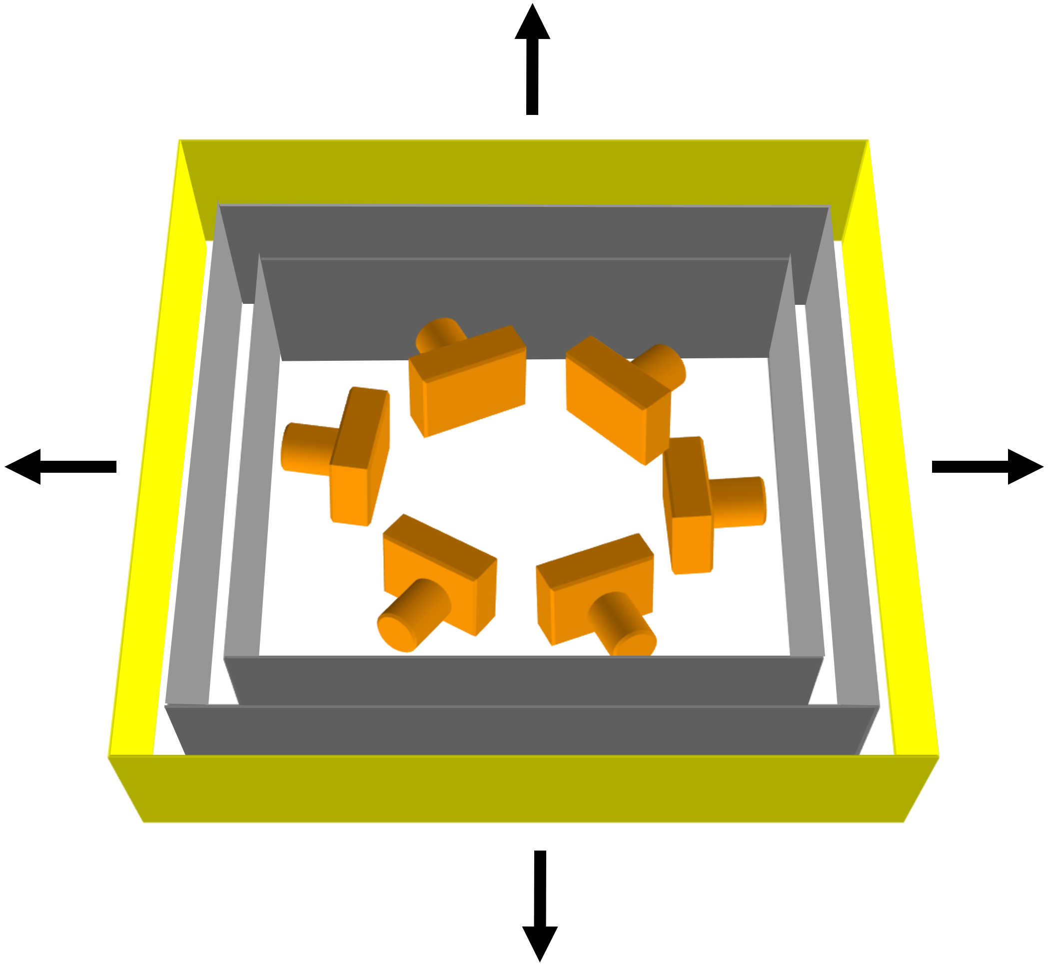
Layers radiate outwards from cameras

Panoramic Layering



Layers radiate outwards from cameras

Panoramic Layering

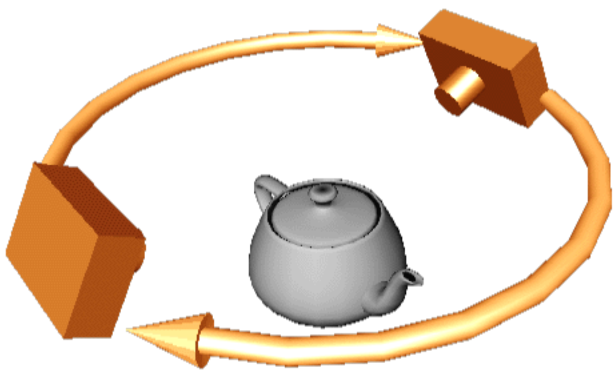


Layers radiate outwards from cameras

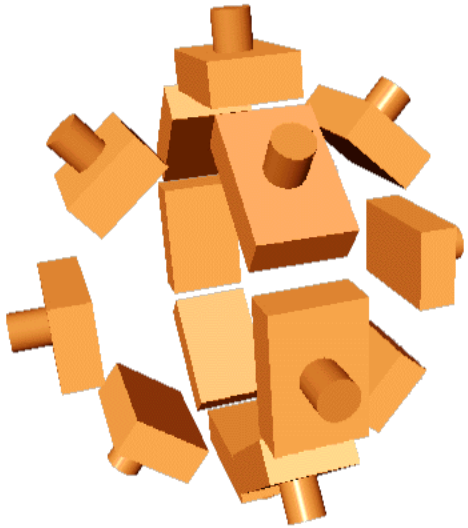
Compatible Camera Configurations

Depth-Order Constraint

- Scene outside convex hull of camera centers



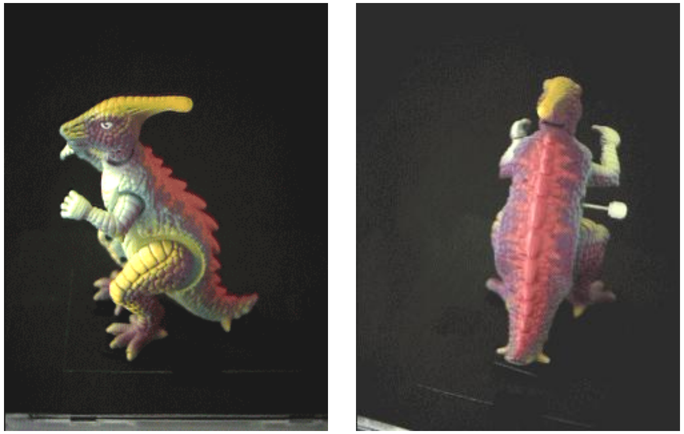
Inward-Looking



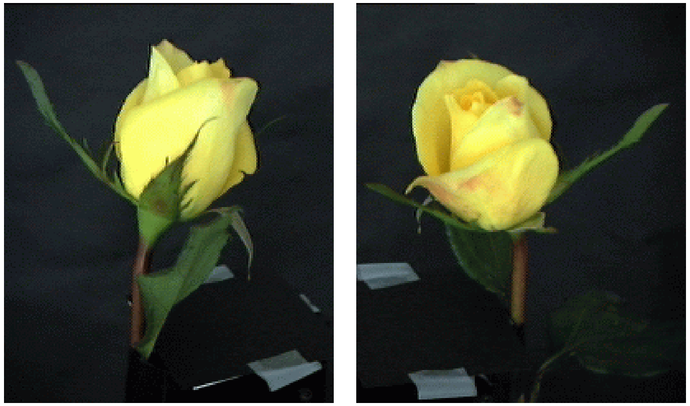
Outward-Looking

Calibrated Image Acquisition

Calibrated Turntable



Selected Dinosaur Images



Selected Flower Images

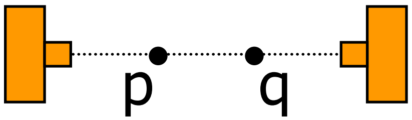
Voxel Coloring Results (Video)

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

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

Limitations of Depth Ordering

A view-independent depth order may not exist



Need more powerful general-case algorithms

- Unconstrained camera positions
- Unconstrained scene geometry/topology

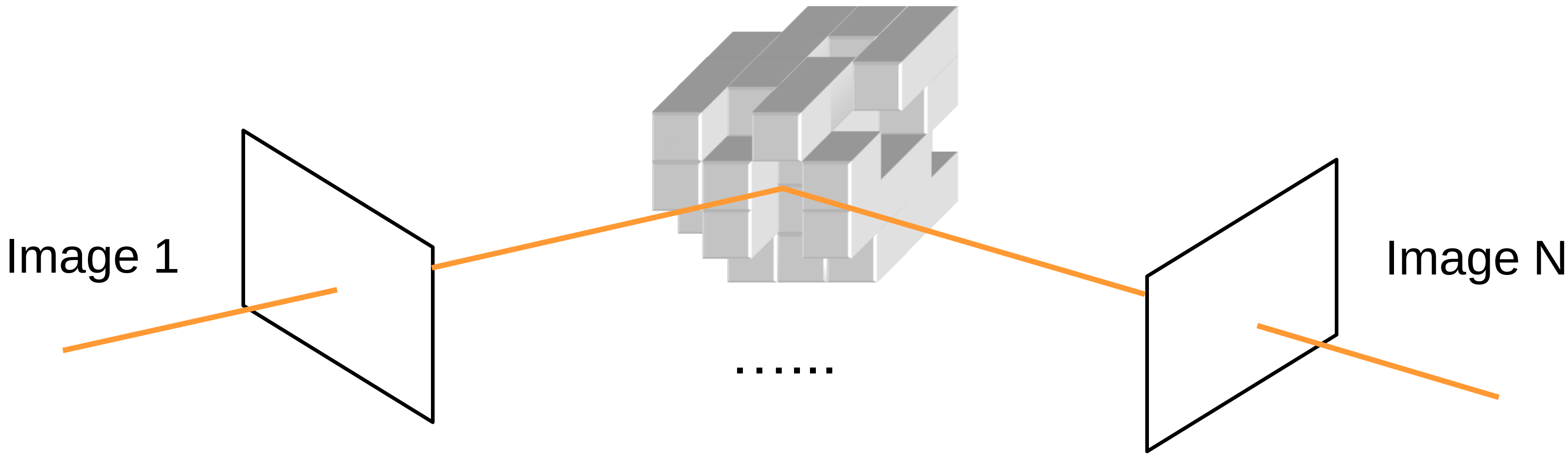
Voxel Coloring Solutions

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

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

3. General Case
   - Space carving [Kutulakos & Seitz 98]
     - > For more info: http://www.cs.washington.edu/homes/seitz/papers/kutu-ijcv00.pdf

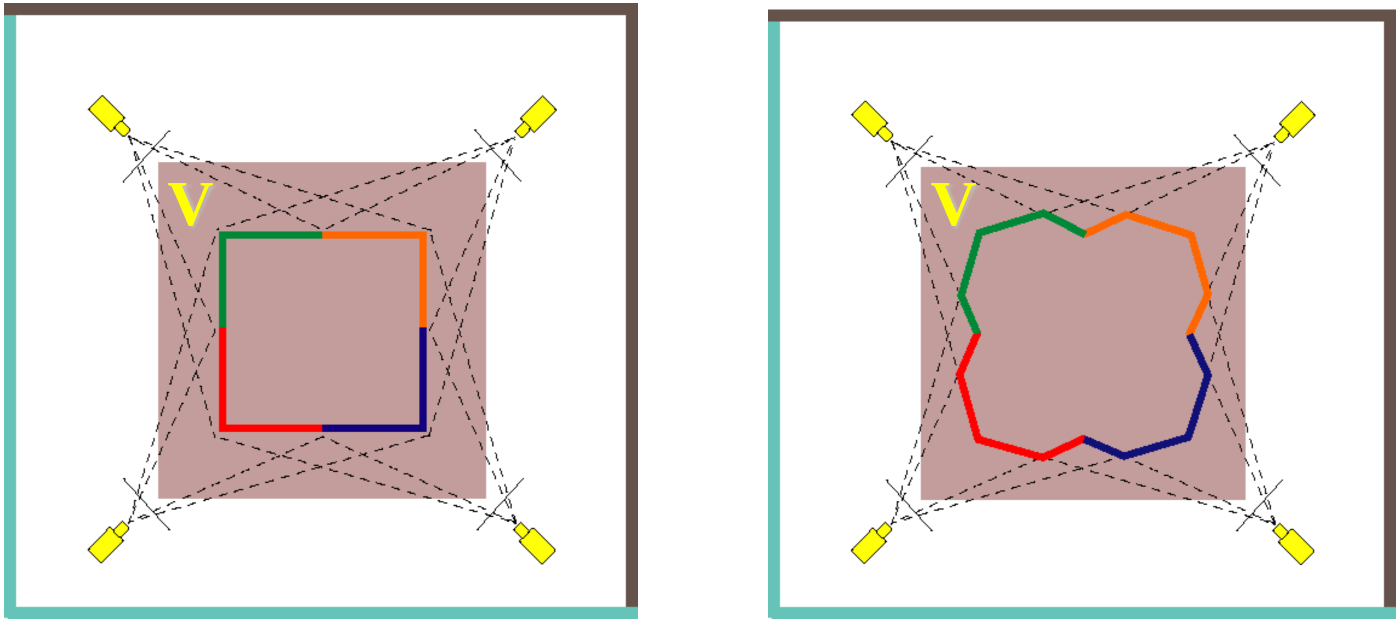
Space Carving Algorithm



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

Space Carving Algorithm

The Basic Algorithm is Unwieldy

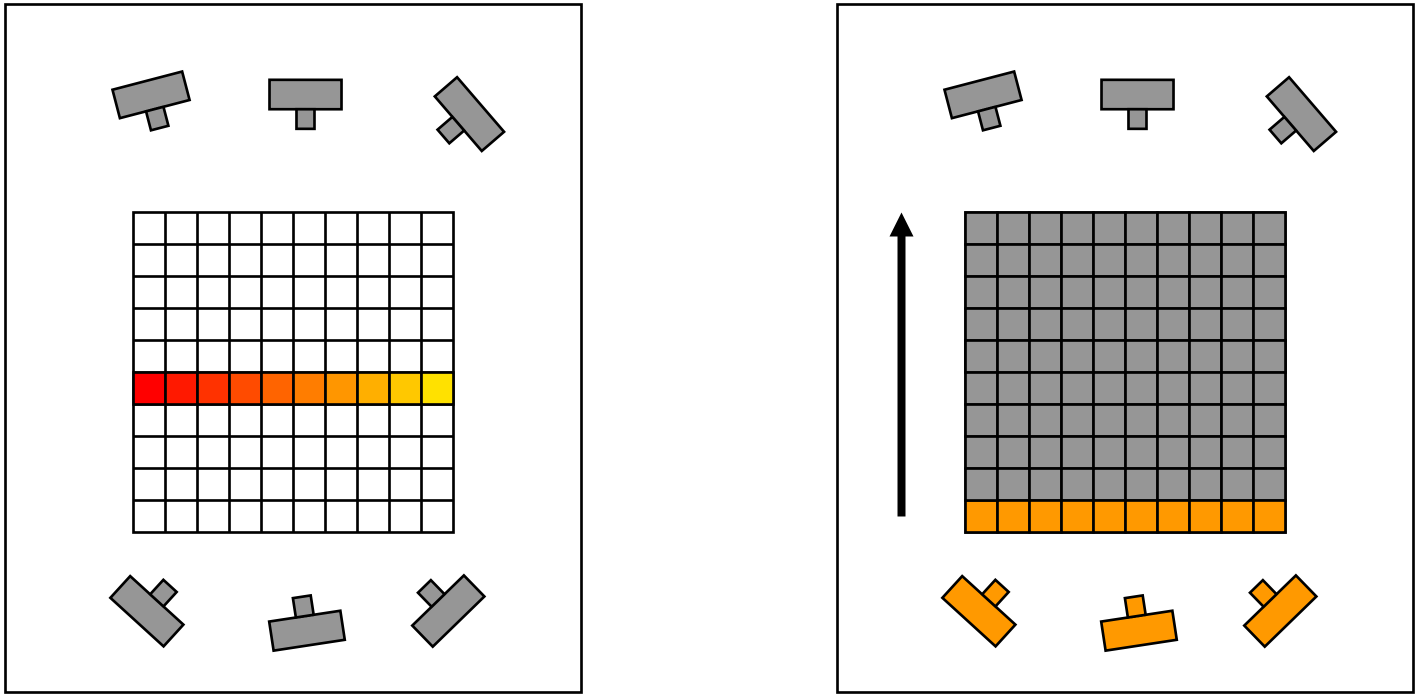
- Complex update procedure

Alternative: Multi-Pass Plane Sweep

- Efficient, can use texture-mapping hardware
- Converges quickly in practice
- Easy to implement

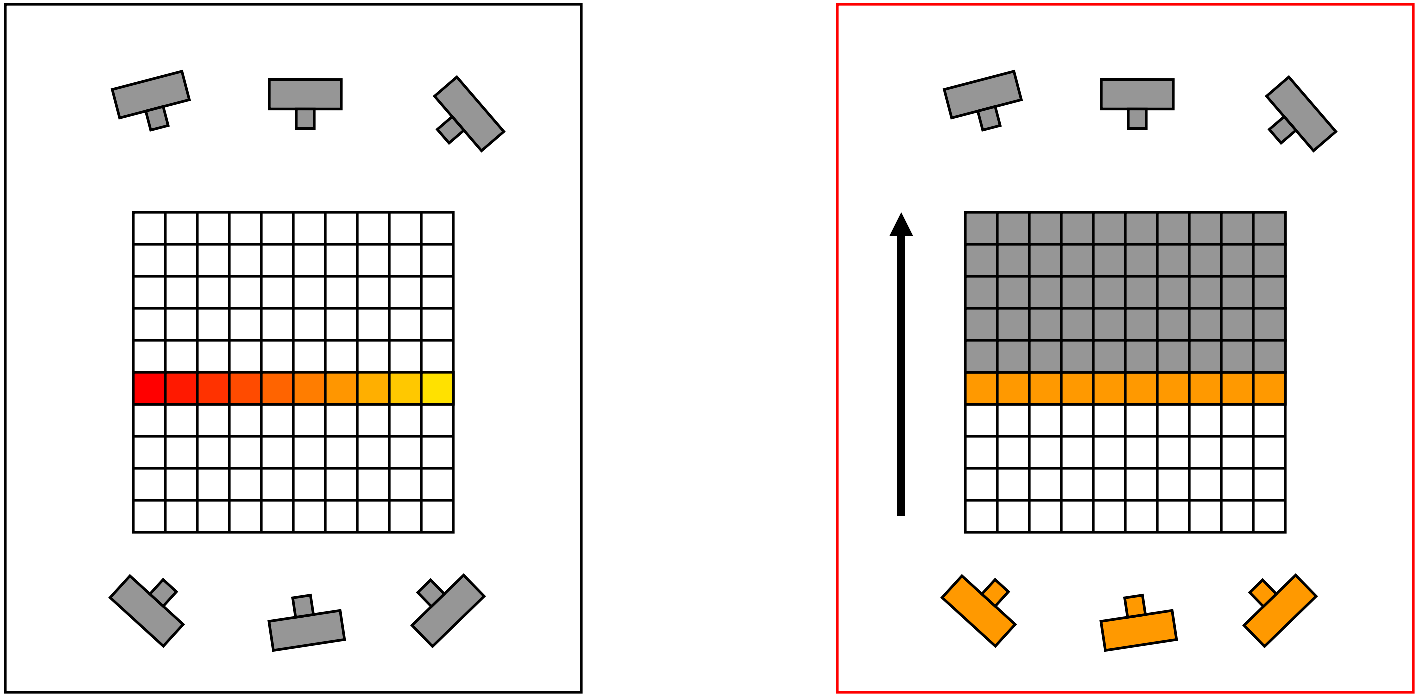
Multi-Pass Plane Sweep

- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence



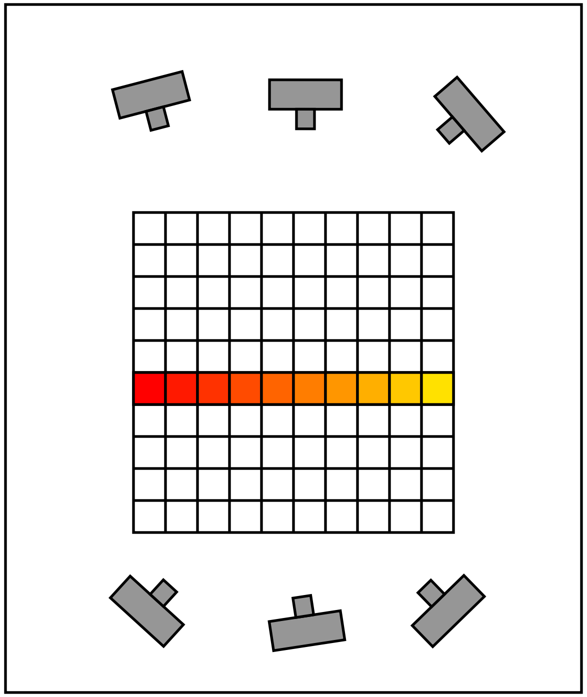
Multi-Pass Plane Sweep

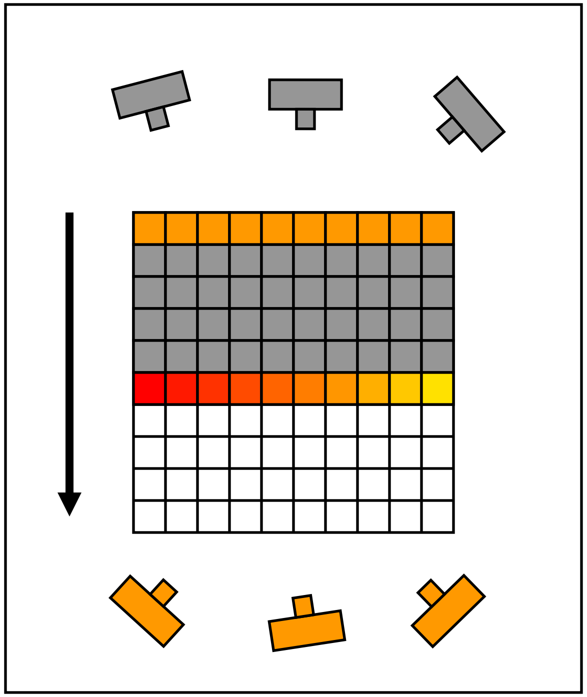
- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence



Multi-Pass Plane Sweep

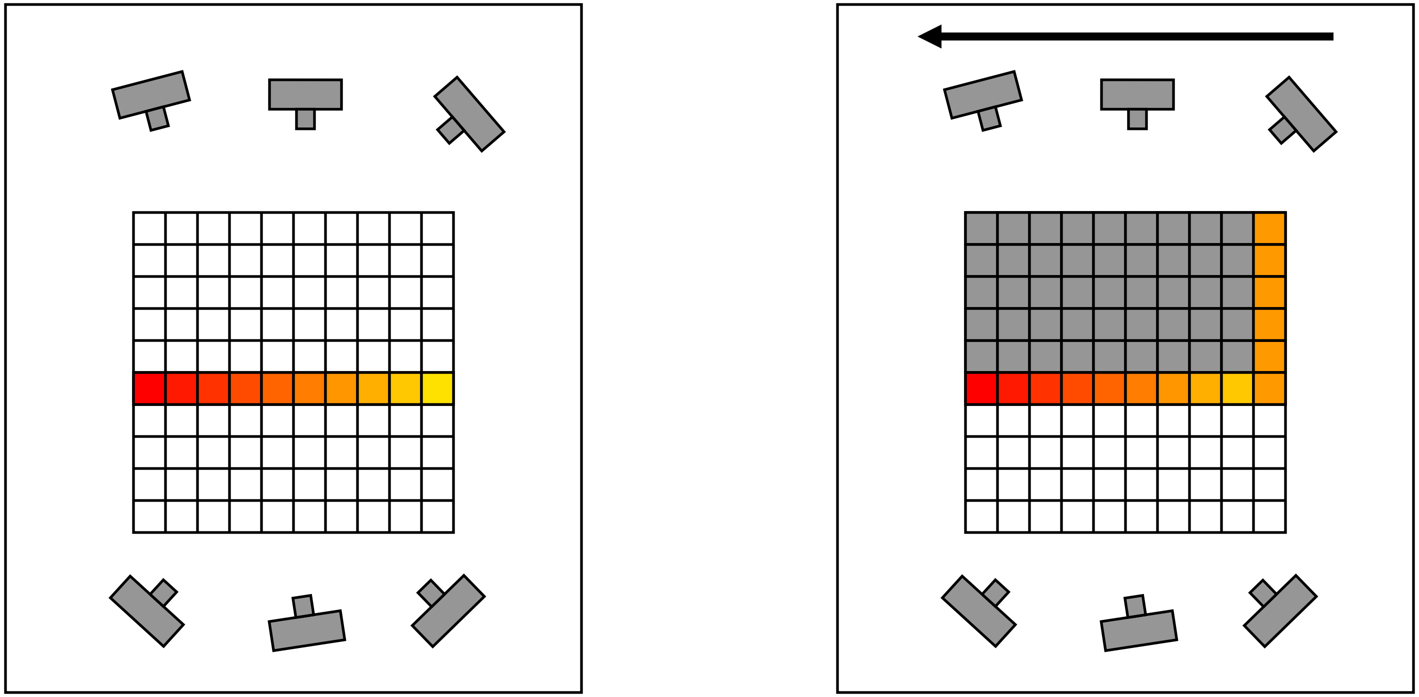
- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence





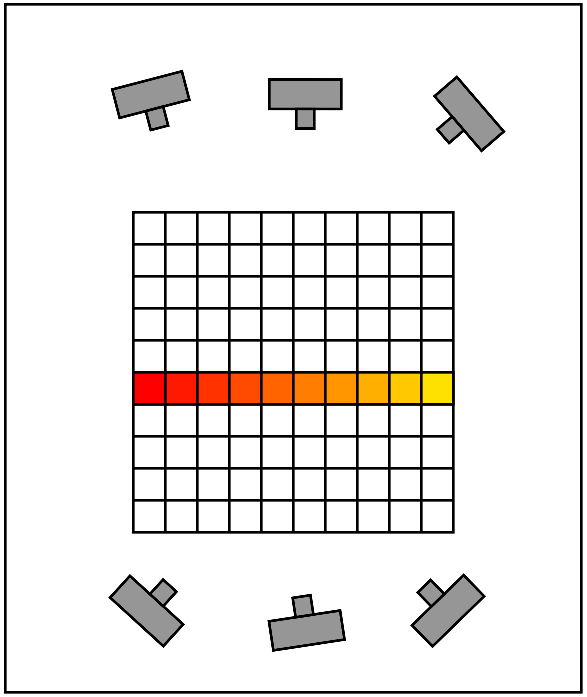
Multi-Pass Plane Sweep

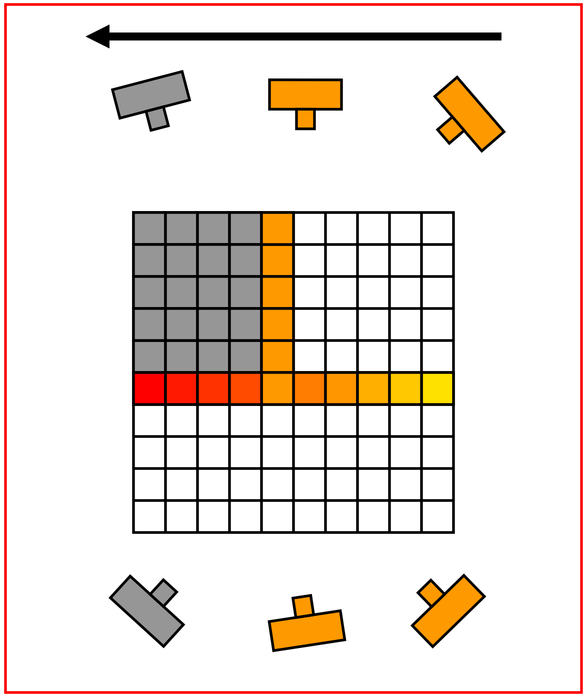
- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence



Multi-Pass Plane Sweep

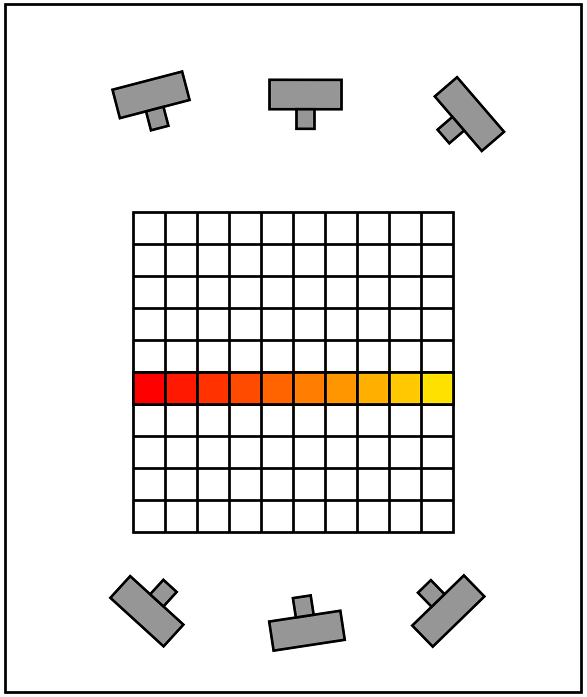
- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence

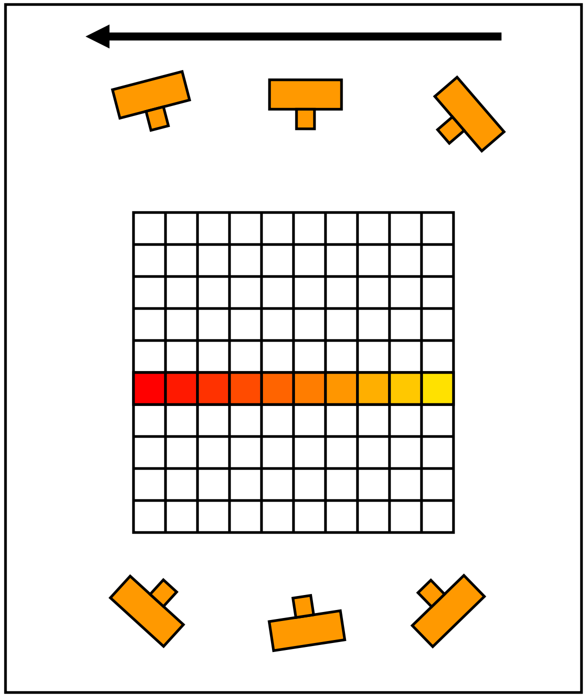




Multi-Pass Plane Sweep

- Sweep plane in each of 6 principle directions
- Consider cameras on only one side of plane
- Repeat until convergence



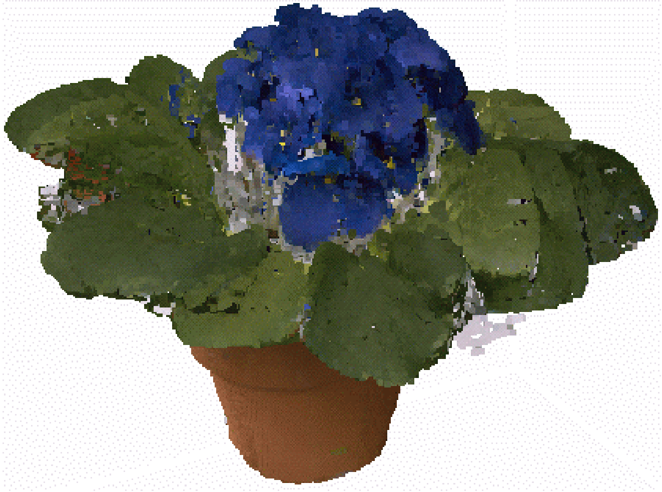


Space Carving Results: African Violet

Input Image (1 of 45)

Reconstruction

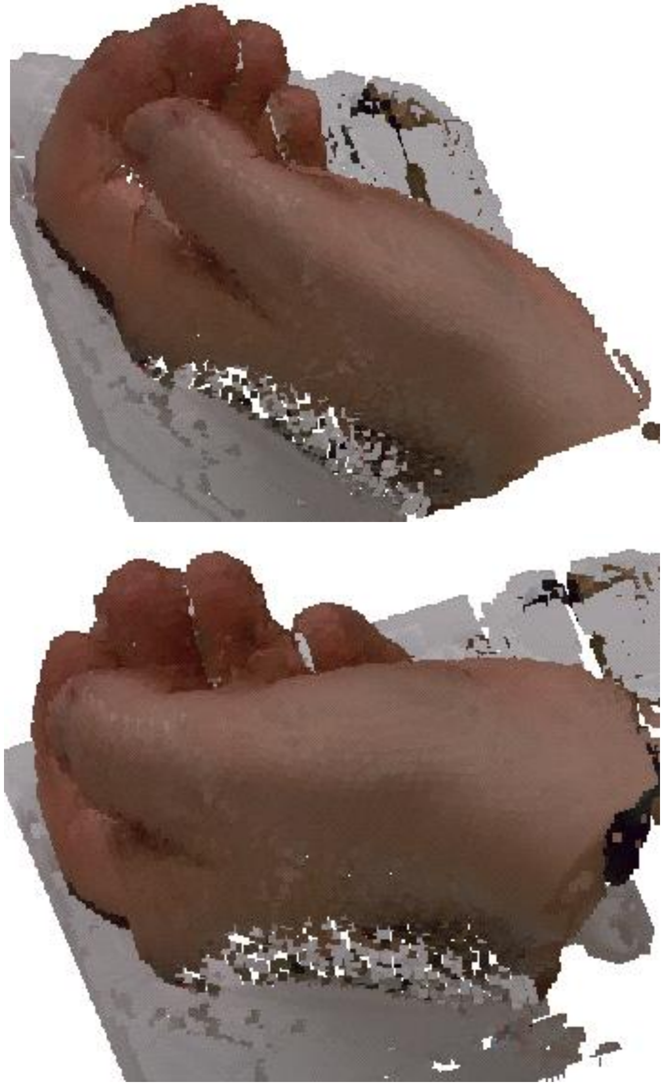
Reconstruction



Reconstruction

Space Carving Results: Hand

Input Image
(1 of 100)



Views of Reconstruction

Properties of Space Carving

Pros

- Voxel coloring version is easy to implement, fast
- Photo-consistent results
- No smoothness prior

Cons

- Bulging
- No smoothness prior

Alternatives to space carving

Optimizing space carving

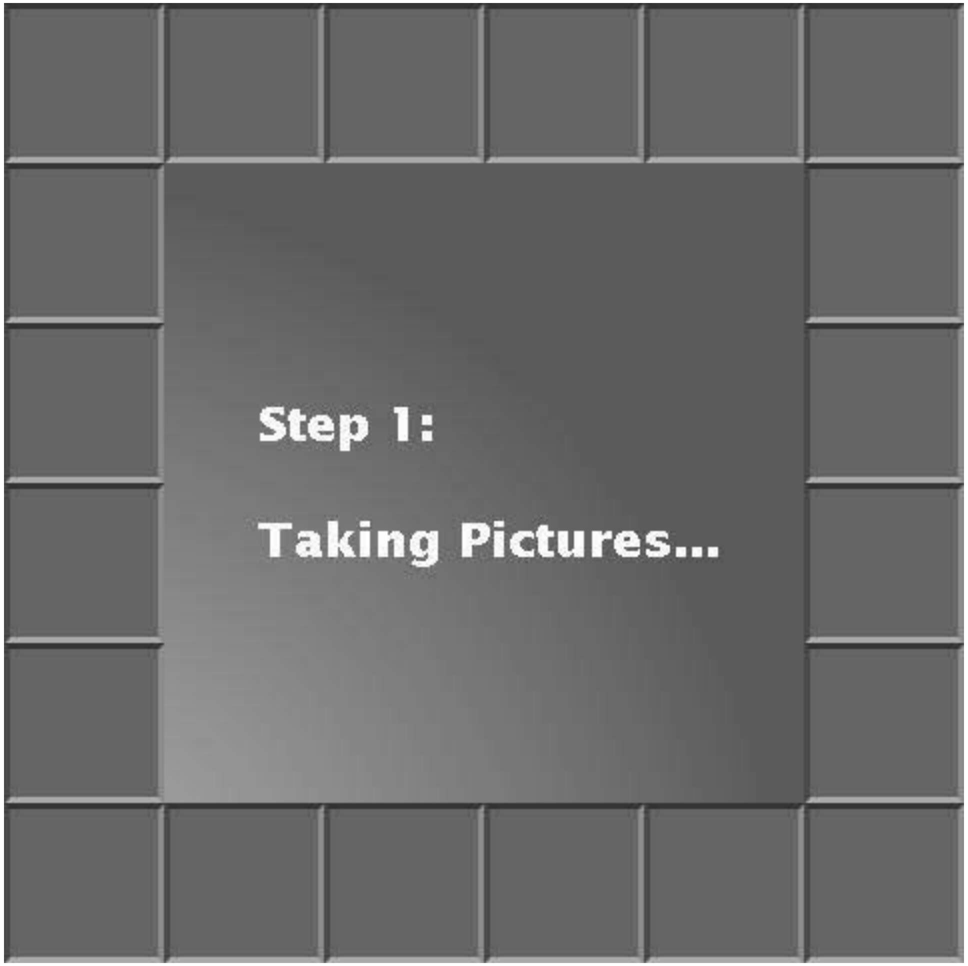
- recent surveys
  - >Slabaugh et al., 2001
  - >Dyer et al., 2001
- many others...

Graph cuts

- Kolmogorov & Zabih

Level sets

- introduce smoothness term
- surface represented as an implicit function in 3D volume
- optimize by solving PDE’s



Alternatives to space carving

Optimizing space carving

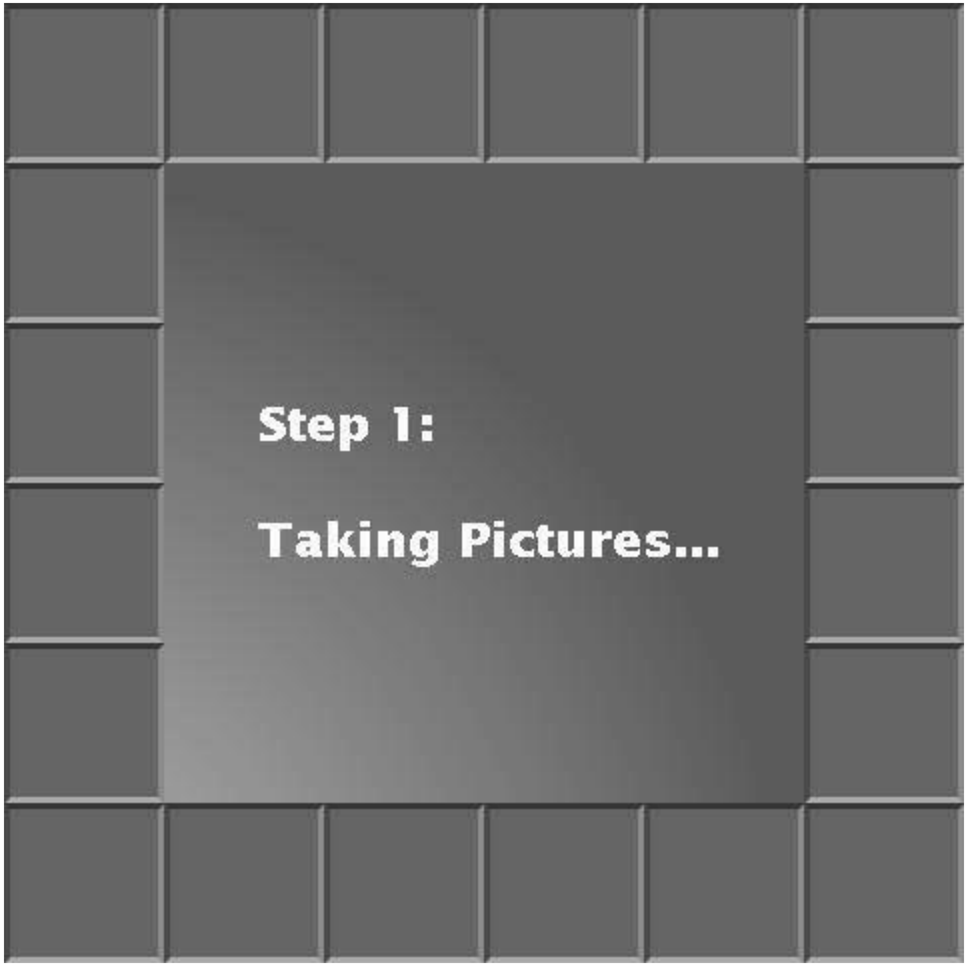
- recent surveys
  - >Slabaugh et al., 2001
  - >Dyer et al., 2001
- many others...

Graph cuts

- Kolmogorov & Zabih

Level sets

- introduce smoothness term
- surface represented as an implicit function in 3D volume
- optimize by solving PDE's



Level sets vs. space carving

Advantages of level sets

- optimizes consistency with images + smoothness term
- excellent results for smooth things
- does not require as many images

Advantages of space carving

- much simpler to implement
- runs faster (orders of magnitude)
- works better for thin structures, discontinuities

For more info on level set stereo:

- Renaud Keriven's page:
  - http://cermics.enpc.fr/~keriven/stereo.html

References

Volume Intersection

- Martin & Aggarwal, “Volumetric description of objects from multiple views”, Trans. Pattern Analysis and Machine Intelligence, 5(2), 1991, pp. 150-158.
- Szeliski, “Rapid Octree Construction from Image Sequences”, Computer Vision, Graphics, and Image Processing: Image Understanding, 58(1), 1993, pp. 23-32.
- Matusik, Buehler, Raskar, McMillan, and Gortler , “Image-Based Visual Hulls”, Proc. SIGGRAPH 2000, pp. 369-374.

Voxel Coloring and Space Carving

- Seitz & Dyer, “Photorealistic Scene Reconstruction by Voxel Coloring”, Intl. Journal of Computer Vision (IJCV), 1999, 35(2), pp. 151-173.
- Kutulakos & Seitz, “A Theory of Shape by Space Carving”, International Journal of Computer Vision, 2000, 38(3), pp. 199-218.
- **Recent surveys**
  - Slabaugh, Culbertson, Malzbender, & Schafer, “A Survey of Volumetric Scene Reconstruction Methods from Photographs”, Proc. workshop on Volume Graphics 2001, pp. 81-100. http://users.ece.gatech.edu/~slabaugh/personal/publications/vg01.pdf
  - Dyer, “Volumetric Scene Reconstruction from Multiple Views”, Foundations of Image Understanding, L. S. Davis, ed., Kluwer, Boston, 2001, 469-489. ftp://ftp.cs.wisc.edu/computer-vision/repository/PDF/dyer.2001.fia.pdf

References

Other references from this talk

- **Multibaseline Stereo**: Masatoshi Okutomi and Takeo Kanade. A multiple-baseline stereo. IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 15(4), 1993, pp. 353--363.
- **Level sets:** Faugeras & Keriven, "Variational principles, surface evolution, PDE's, level set methods and the stereo problem", IEEE Trans. on Image Processing, 7(3), 1998, pp. 336-344.
- **Mesh based**: Fua & Leclerc, "Object-centered surface reconstruction: Combining multi-image stereo and shading", IJCV, 16, 1995, pp. 35-56.
- **3D Room:** Narayanan, Rander, & Kanade, "Constructing Virtual Worlds Using Dense Stereo", Proc. ICCV, 1998, pp. 3-10.
- **Graph-based**: Kolmogorov & Zabih, "Multi-Camera Scene Reconstruction via Graph Cuts", Proc. European Conf. on Computer Vision (ECCV), 2002.
- **Helmholtz Stereo**: Zickler, Belhumeur, & Kriegman, "Helmholtz Stereopsis: Exploiting Reciprocity for Surface Reconstruction", IJCV, 49(2-3), 2002, pp. 215-227.