Image-Based Rendering

COS 526, Fall 2016

Thomas Funkhouser

Acknowledgments: Dan Aliaga, Marc Levoy, Szymon Rusinkiewicz

What is Image-Based Rendering?

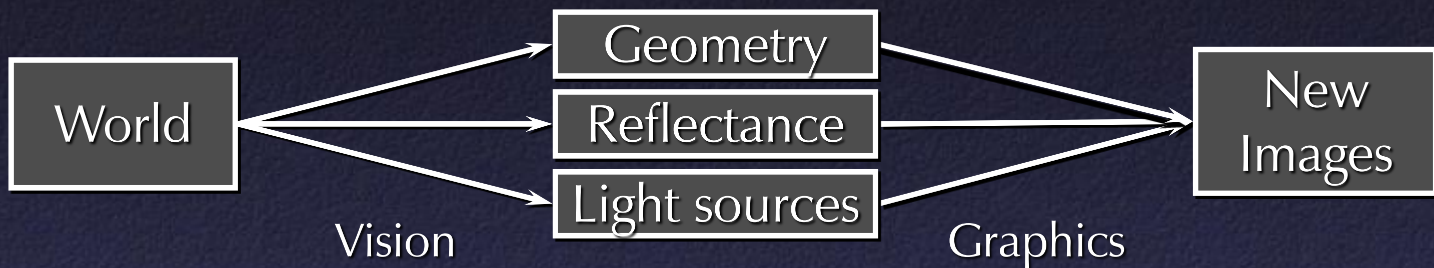
- Definition 1: the use of photographic imagery to overcome the limitations of traditional computer graphics

What is Image-Based Rendering?

- Definition 2: The use of computational techniques to overcome limitations of traditional photography

Image-Based Modeling and Rendering

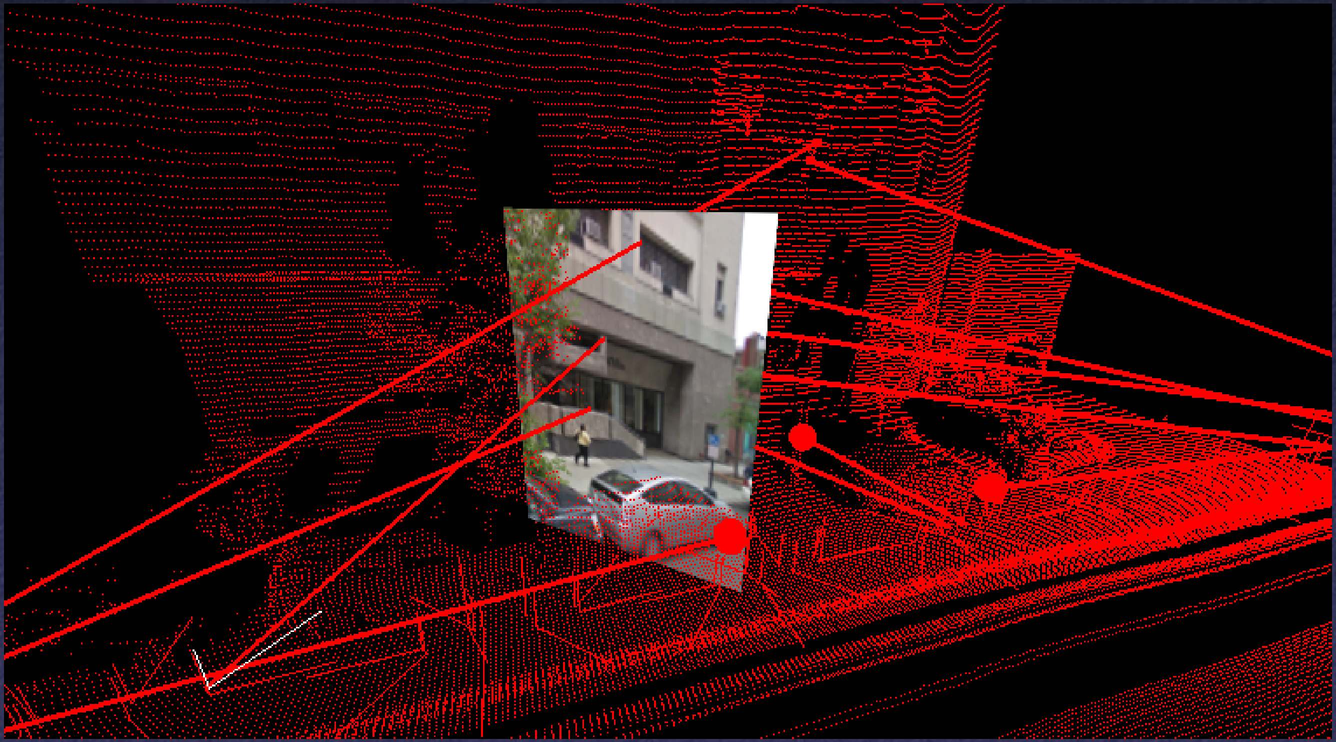
- Traditional vision / graphics pipelines:



- Image-based pipeline:

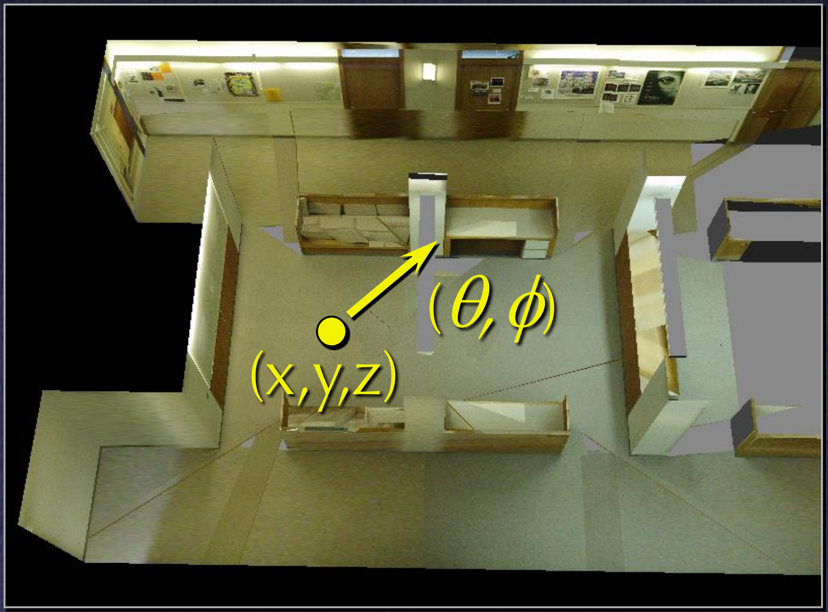
Image-Based Modeling and Rendering

- Generate new views of a scene directly from existing views



Plenoptic Function

- $L(x,y,z,\theta,\phi,t,\lambda)$
- Captures all light flow in a scene
  - to/from any point (x,y,z),
  - in any direction (θ,ϕ),
  - at any time (t),
  - at any frequency (λ)
- Enough information to construct any image of the scene at any time



[Funkhouser]

Plenoptic Function Simplifications

- Simplification from 7D to 3 × 5D
  - Represent color as RGB: eliminate λ
  - Static scenes: eliminate t

- Other simplfications?

Image-Based Representations

7D	Ideal
6D	Consider only 3 frequencies (RGB)
5D	Consider only one time instant (static scene)
4D	Consider only viewpoints inside/outside scene
3D	Consider one dimension fewer directions/positions
2D	Consider viewpoints at finite set points or angles

IBR Representations

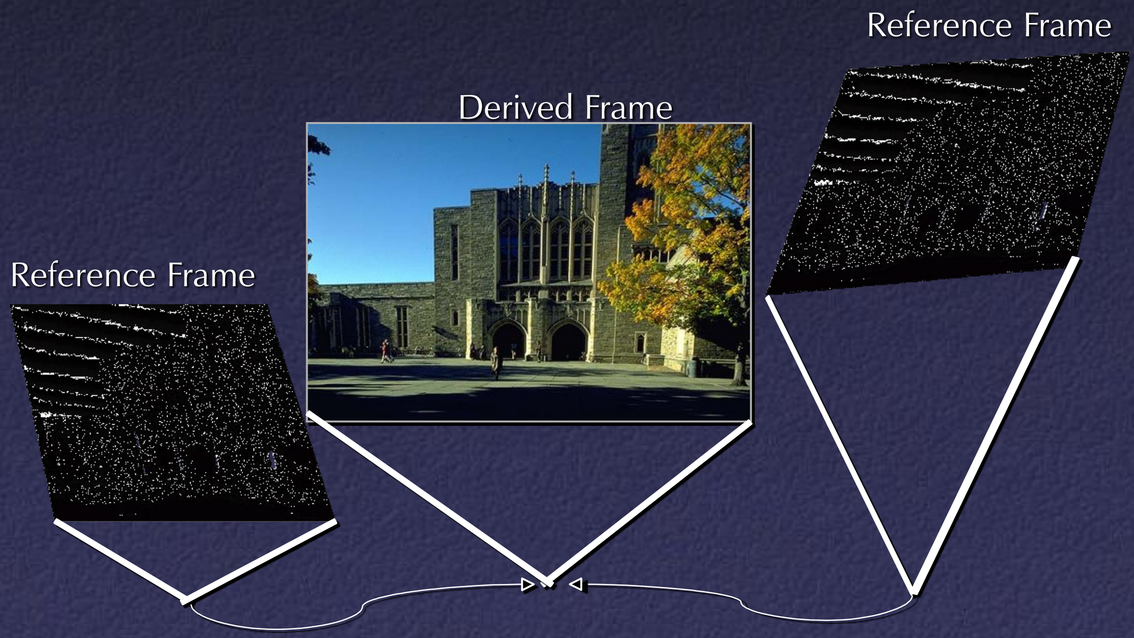
- Image pairs
- Sea of Images
- Lightfields / Lumigraphs

IBR Representations

- Image pairs ←
- Sea of Images
- Lightfields / Lumigraphs

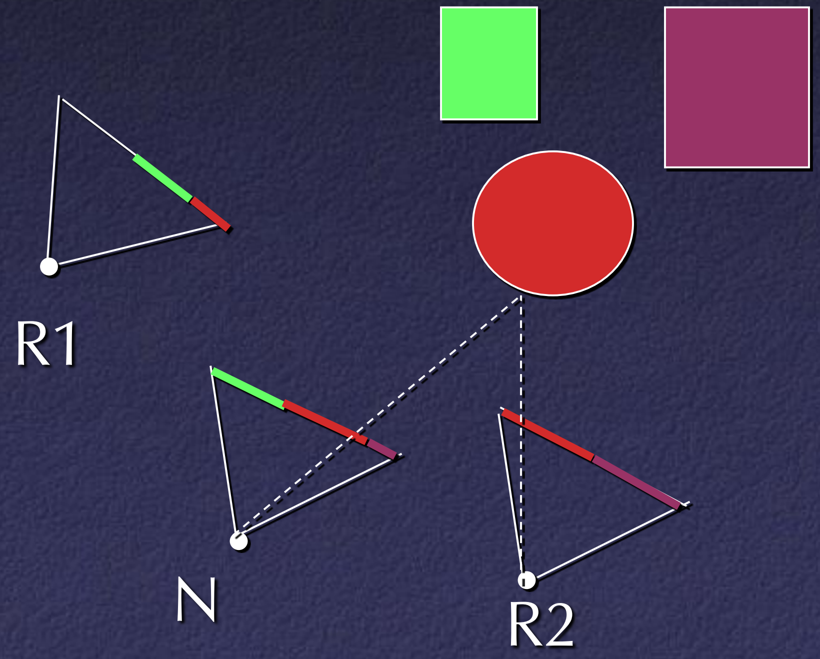
View Interpolation

- Create novel images by resampling photographs
  - Reference images sample 5D plenoptic function



View Interpolation

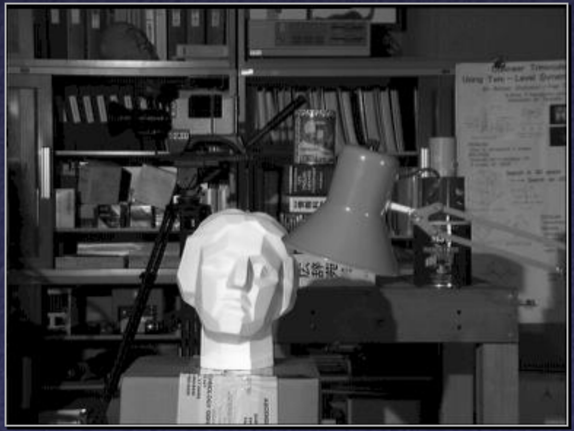
- Method:
  - Warp nearby reference images to novel viewpoint
  - Blend warped images



Morph with warp
defined by
reprojection or
pixel correspondences

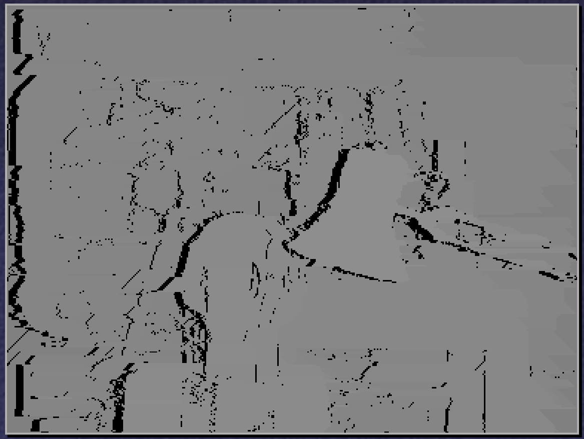
Pixel Correspondences

- Vision (e.g. stereo): disparity
- Feature matching: sparse
- 3D model: possibly coarse



Left

Right

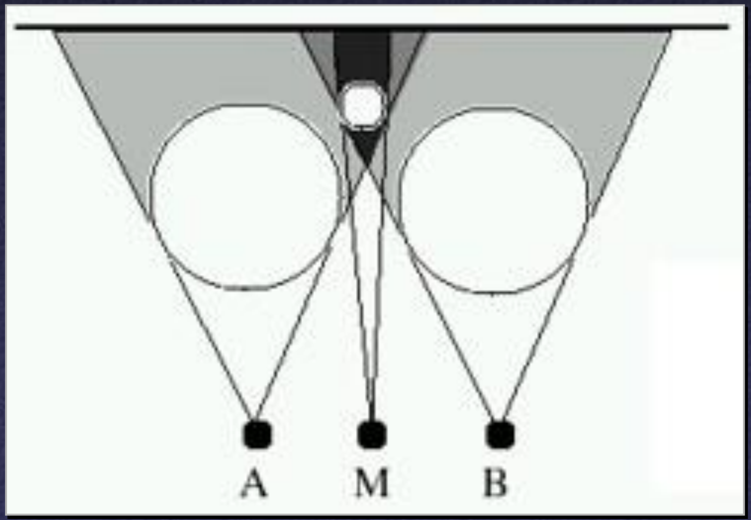


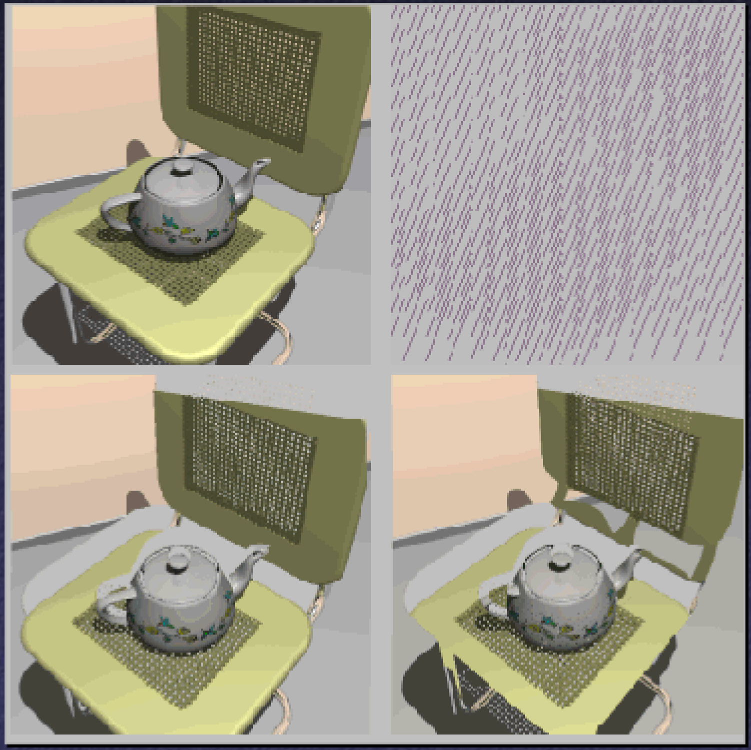
Disparity

[Szeliski]

View Interpolation

- Problem: changes in visibility
  - Disocclusions





[McMillan]

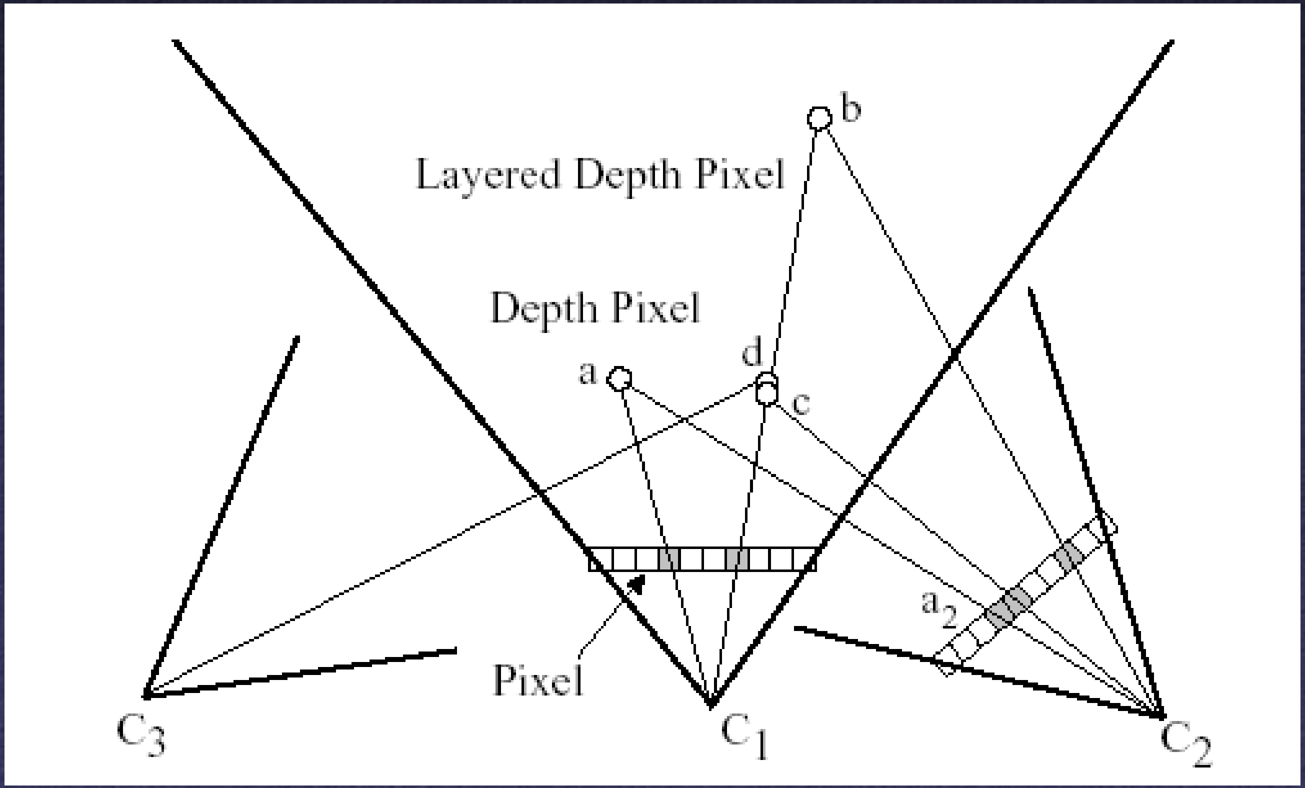
Disocclusions

- Partial solutions:
  - Fill holes by interpolating nearby pixels
  - Fill holes with texture synthesis

[McMillan]

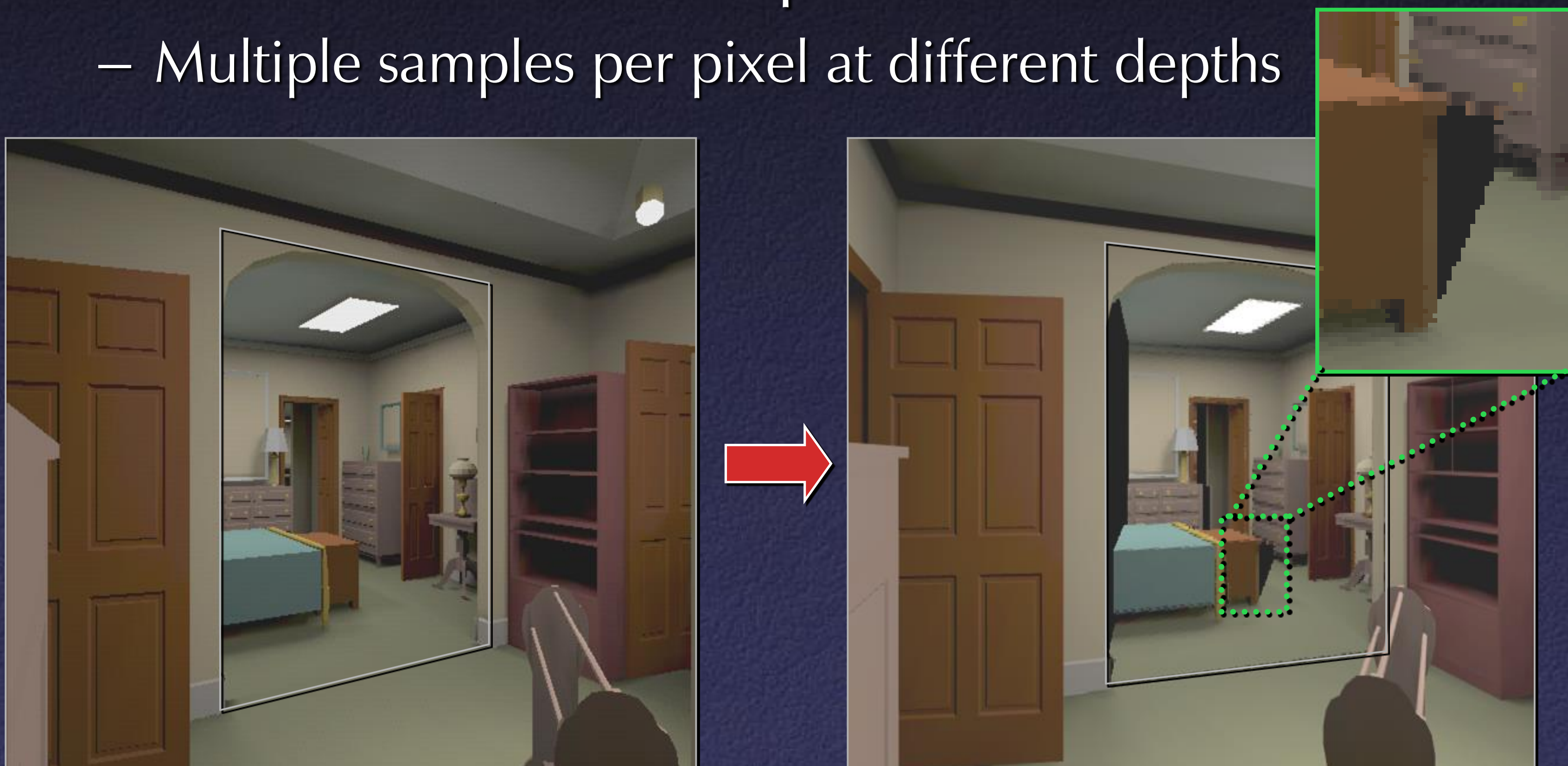
Disocclusions

- Another solution (when possible):
  - Multiple samples per pixel at different depths



Disocclusions

- Another solution (when possible):
  - Multiple samples per pixel at different depths



Reference Image   Warped Depth Image

[Popescu]

Disocclusions

- Another solution (when possible):
  - Multiple samples per pixel at different depths

Reference Image

Warped Layered Depth Image

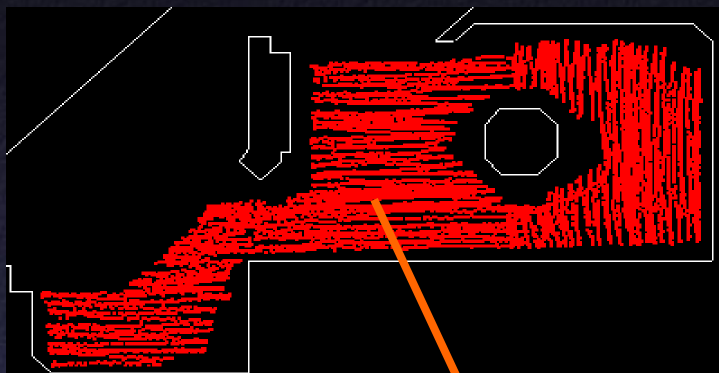
View Interpolation Challenges

- Capture
  - How do we obtain a dense set of calibrated images over a large area in a practical manner?
- Data Management
  - How do we store and access the large amount of data?
- Rendering
  - How do we create novel views from a dense sampling of images in real-time?

IBR Representations

- Image pairs
- Sea of Images ←
- Lightfields / Lumigraphs

Sea of Images



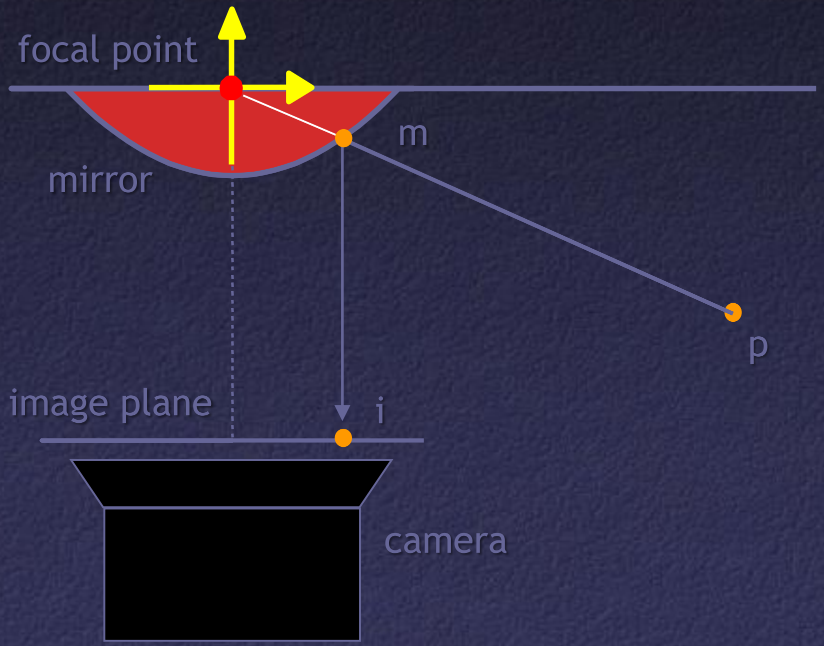
Large-scale Dense Capture

Data Management
and
Rendering Algorithms

Interactive

Sea of Images Capture

- Use a hemispherical FOV camera driven on cart



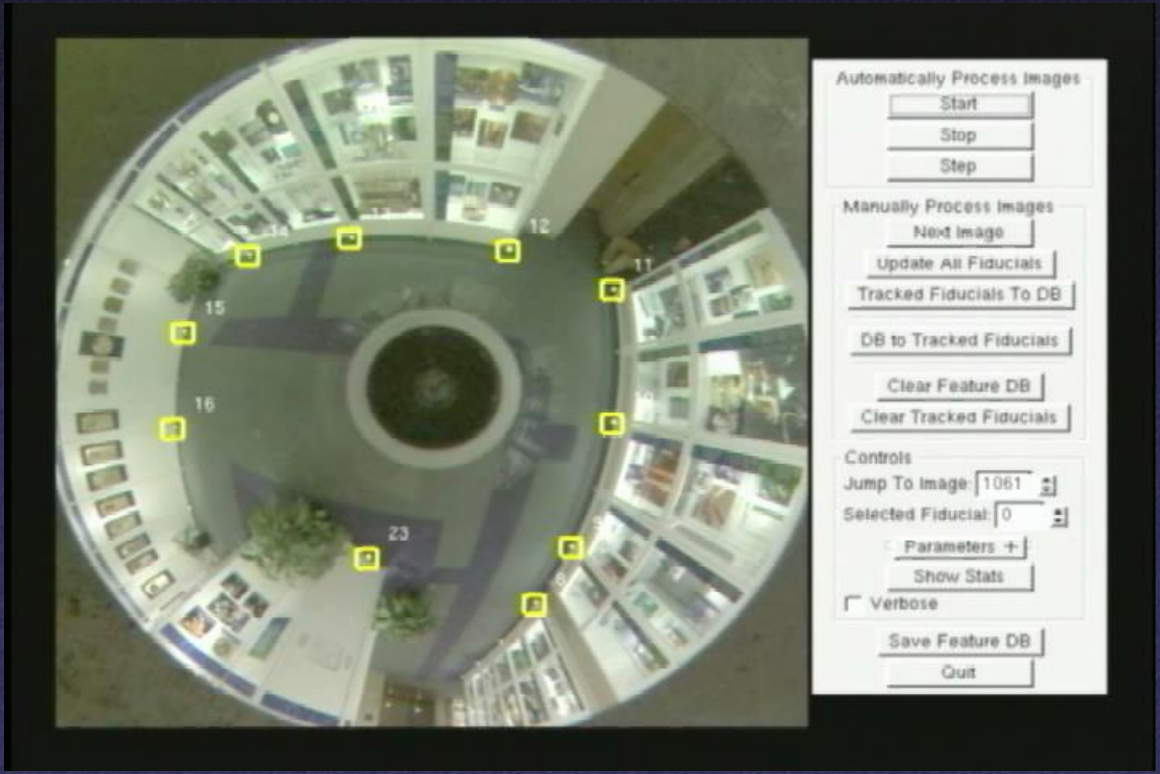
Paraboloidal Catadioptric Camera
[Nayar97]

Sea of Images Capture

- Use a hemispherical FOV camera driven on cart

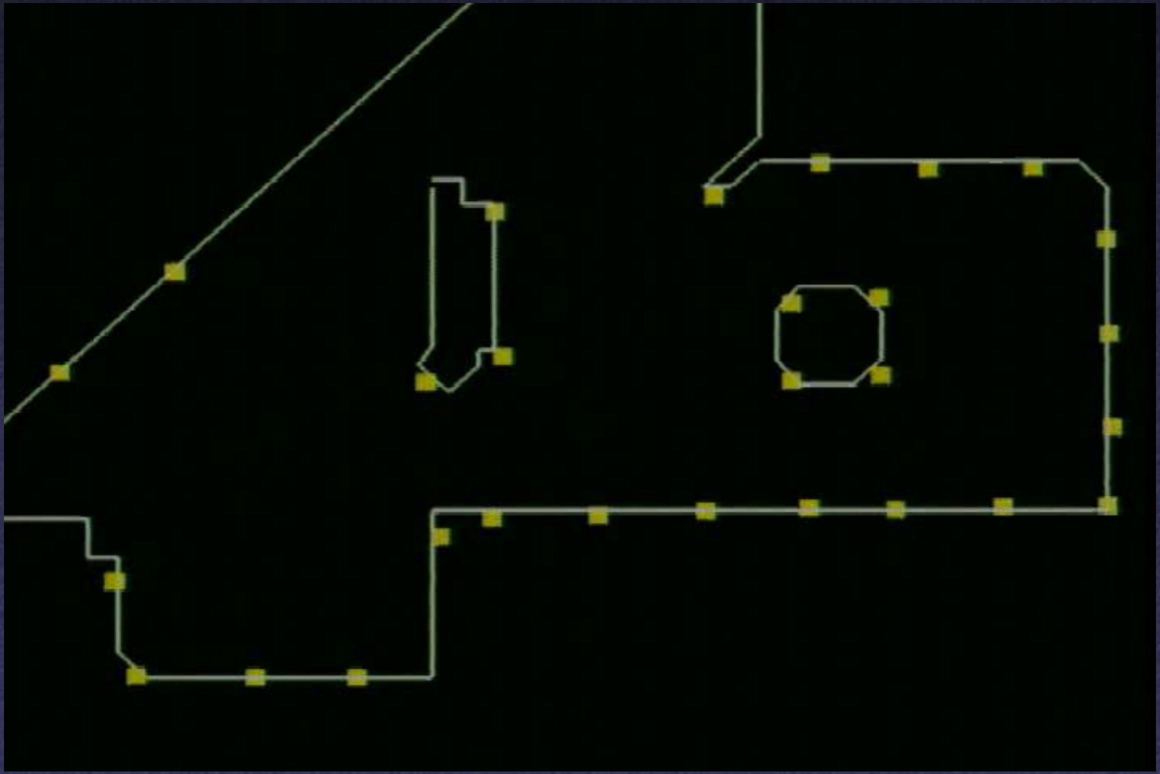
Sea of Images Capture

- Locate camera by tracking fiducials



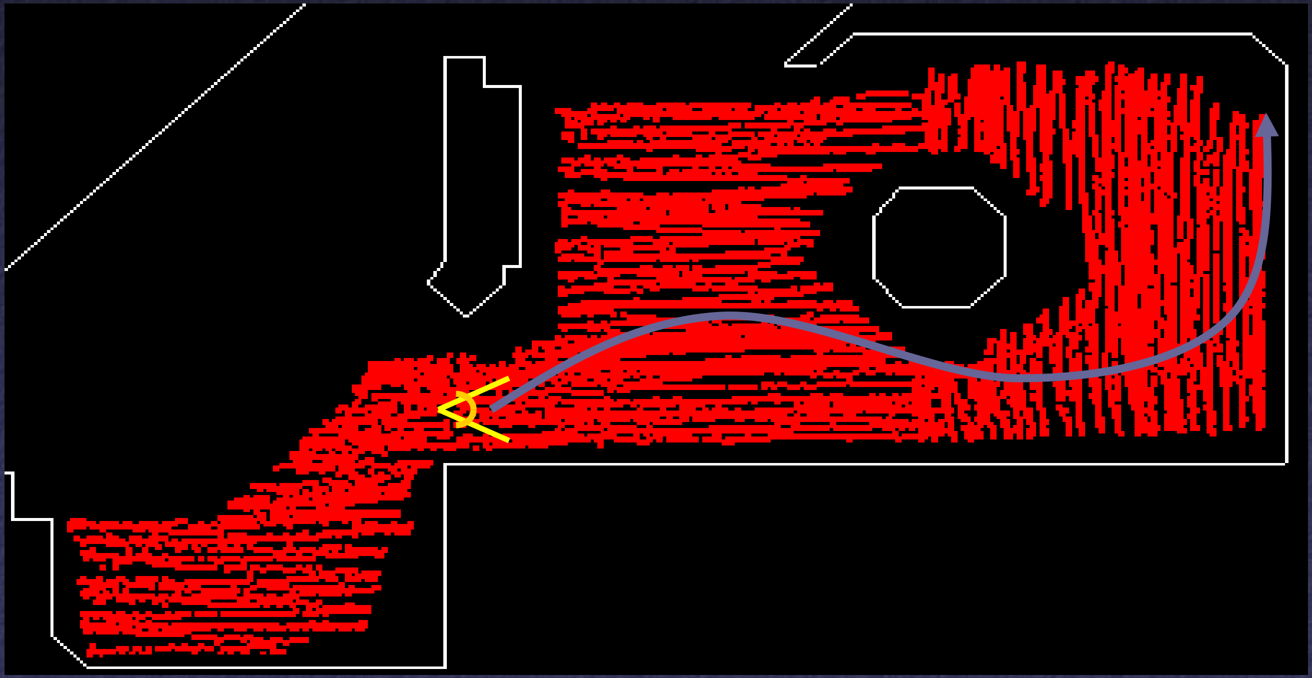
Sea of Images Capture

- Result is a "sea of images" spaced a few inches apart



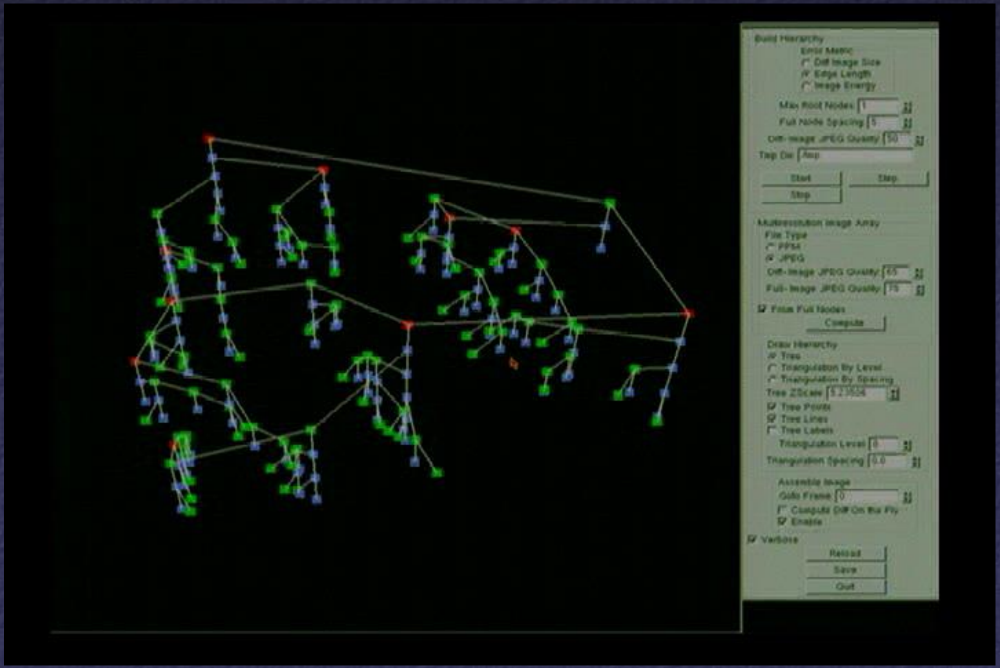
Sea of Images Compression

- Goal: provide access to images along arbitrary viewpoint paths in real-time



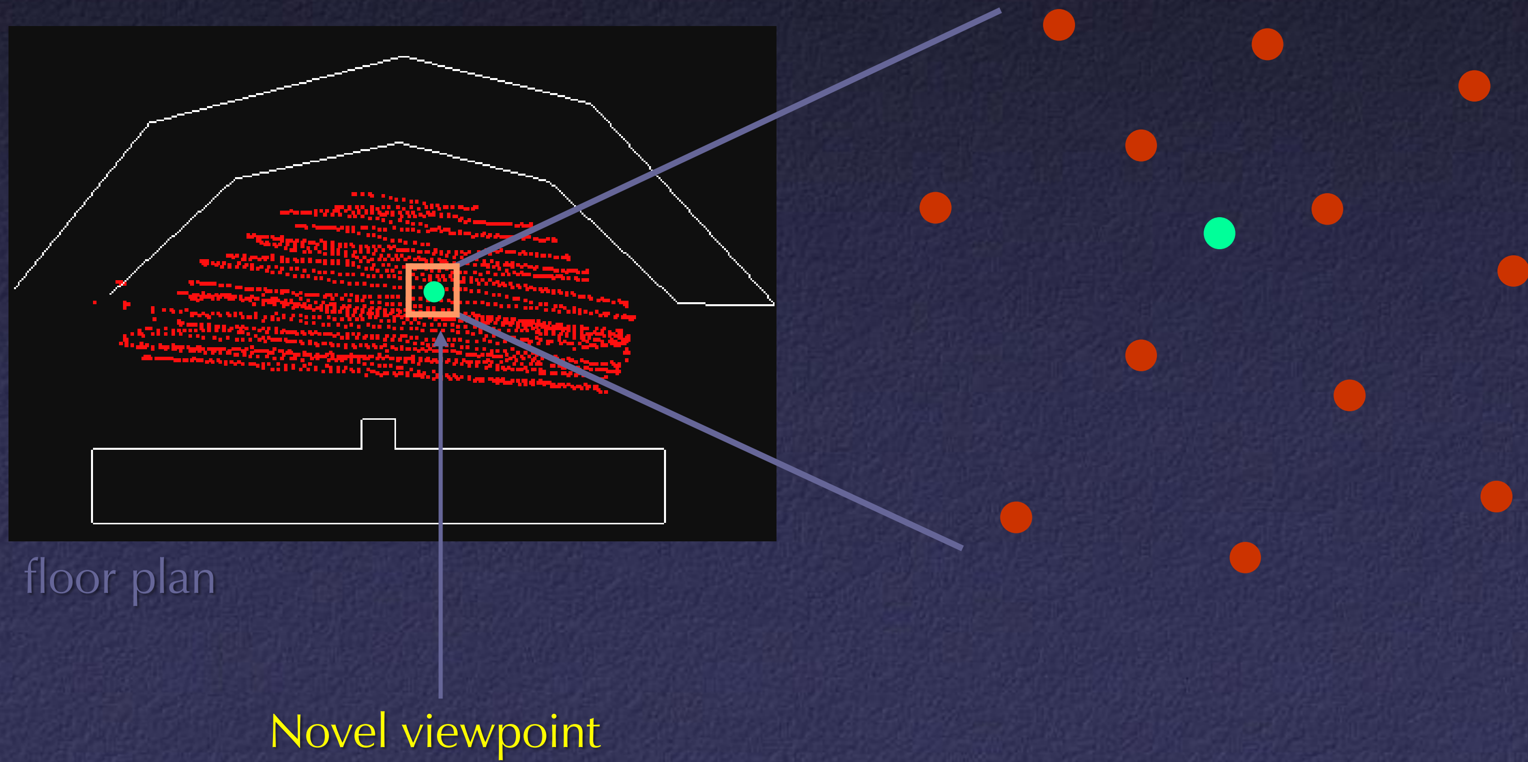
Sea of Images Compression

- Approach: create a multiresolution spatial hierarchy of compressed original images and compressed difference images



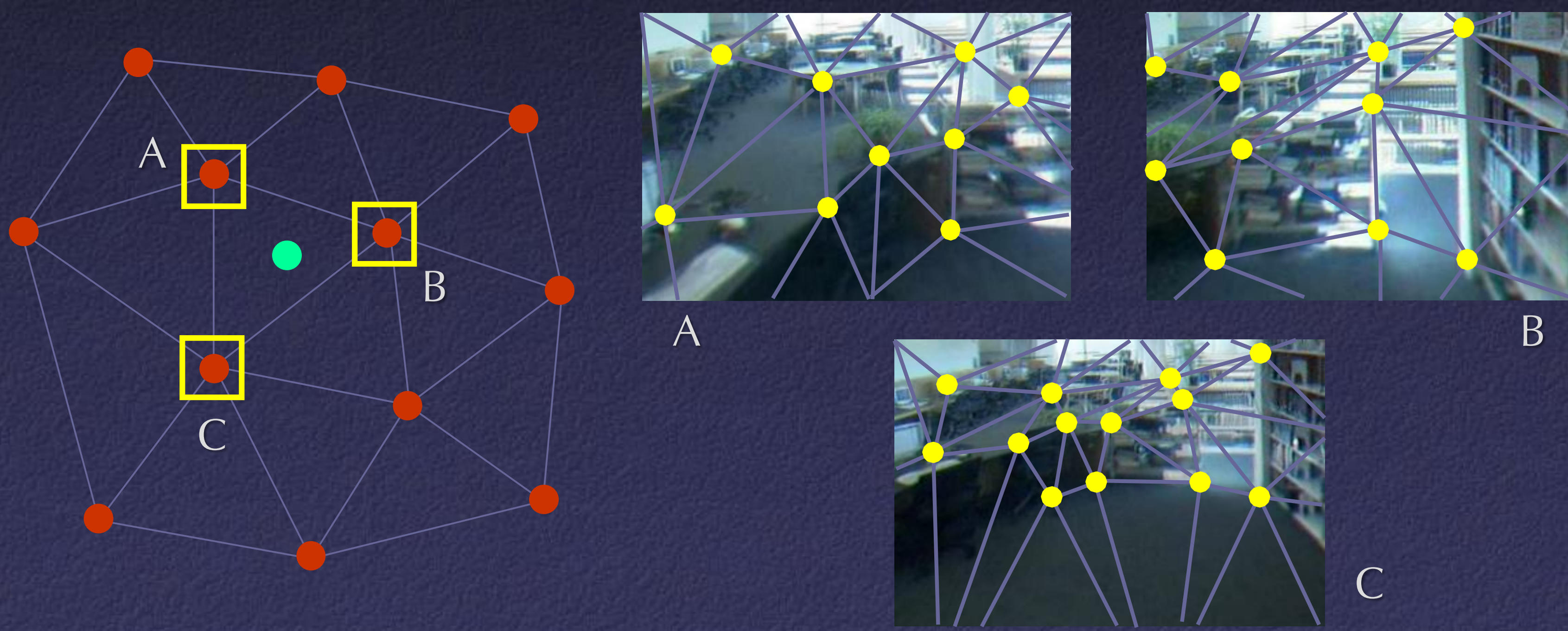
Sea of Images Rendering

- Use captured images near the novel viewpoint to create new views



Sea of Images Rendering

- Interpolate three nearest views using detected feature correspondences



Sea of Images Results

- Bell Labs Museum
  - 900 square ft
  - 9832 images
  - 2.2 inch spacing
- Princeton Library
  - 120 square ft
  - 1947 images
  - 1.6 inches
- Personal Office
  - 30 square feet
  - 3475 images
  - 0.7 inches

[Aliaga02]

Sea of Images Results

- Times
  - Setup: ~15 minutes
  - Capture: ~30-60 minutes
  - Preprocessing time: 4 to 17 hours
- Frame rate
  - 1024x1024 @ 20Hz, 512x512 @ 30Hz

Sea of Images Results

Sea of Images Results

Sea of Images Results

Sea of Images Results

- Render complex light effects (specular highlights)

cylindrical projection

Sea of Images Results

- Multiresolution pre-filtering: far-to-near image sequence

Sea of Images Results

captured omnidirectional image

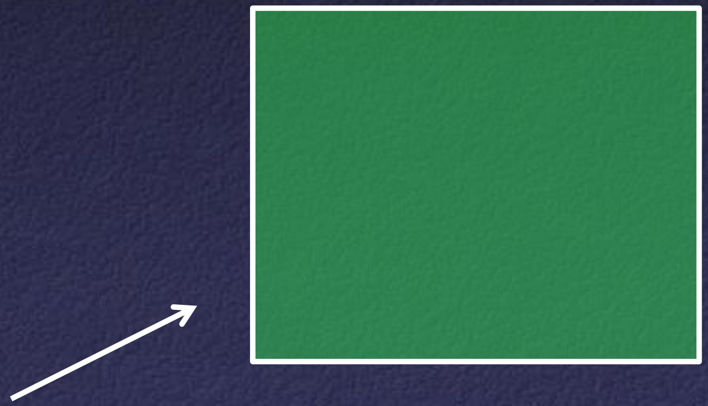
reconstructed omnidirectional image

IBR Representations

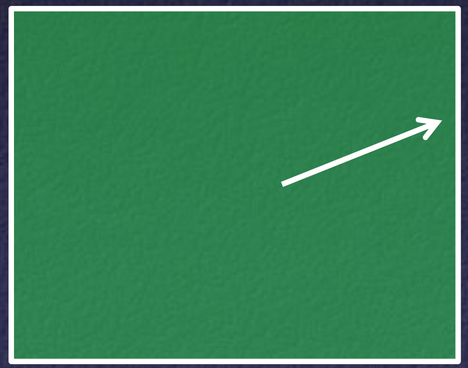
- Image pairs
- Sea of Images
- Lightfields / Lumigraphs

Lightfields

- In unoccluded space, can reduce plenoptic function to 4D



Outside looking in



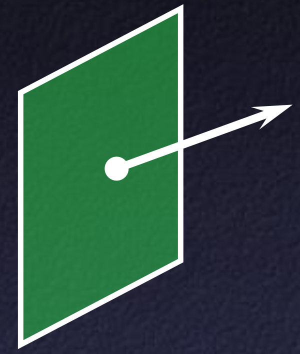
Inside looking out

Using Lightfields

- Obtain 2D slices of 4D data set
- Arbitrary views: take other 2D slices
- Challenges:
  - – Parameterization
  - – Capture
  - – Compression
  - – Rendering

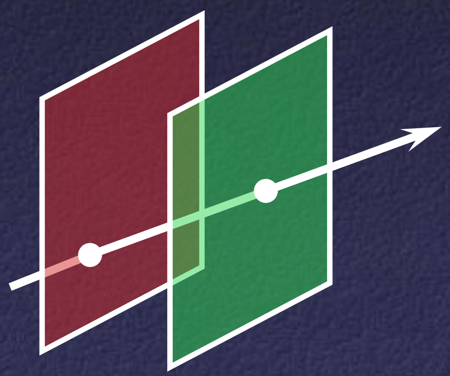
Lightfield Parameterization

- Point / angle



- Two points on a sphere

- Points on two planes

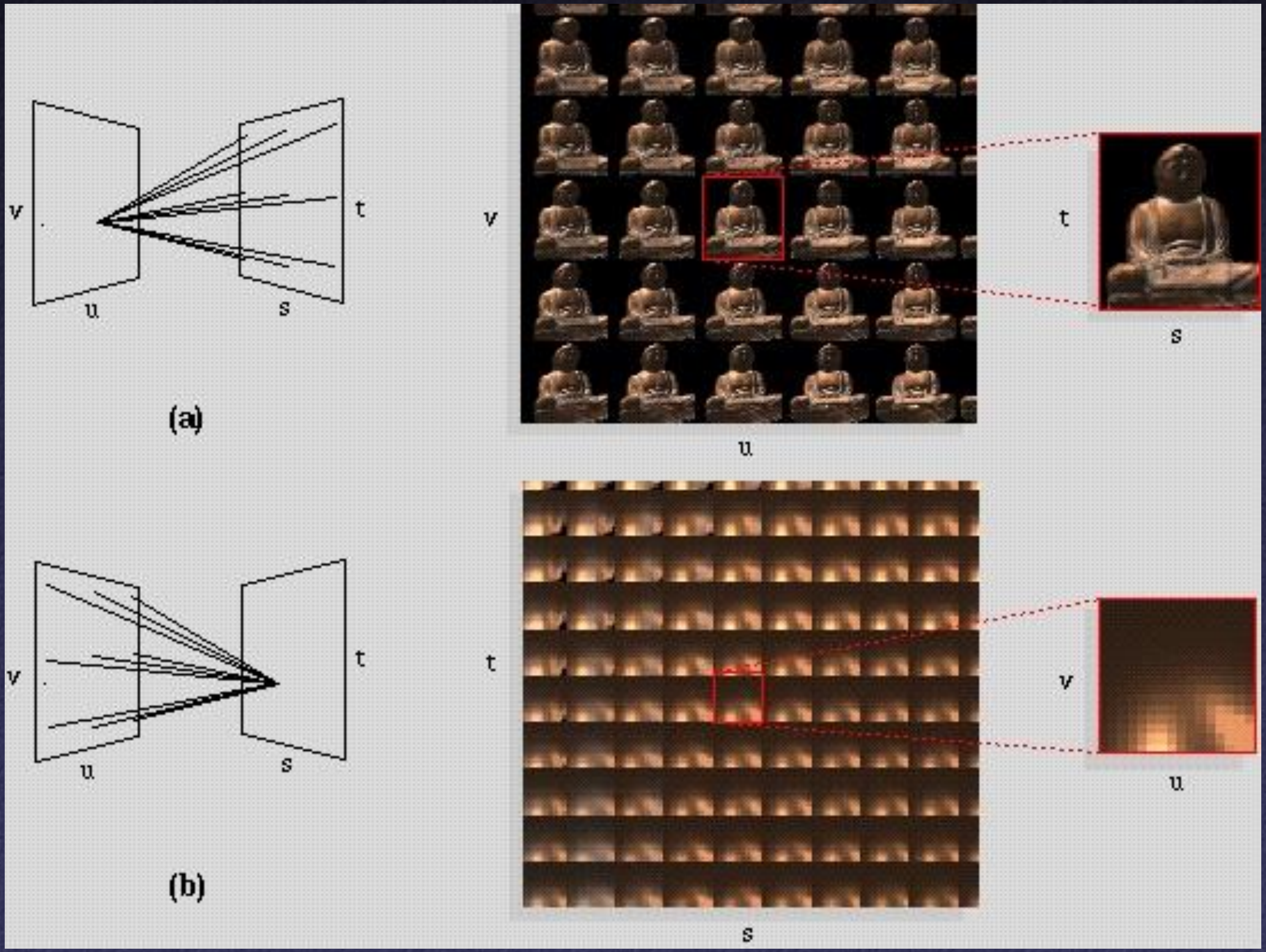


- Original images and camera positions

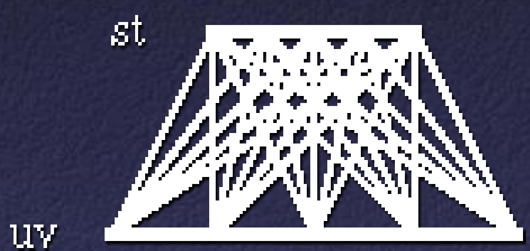
Light Field Two-Plane Parameterization

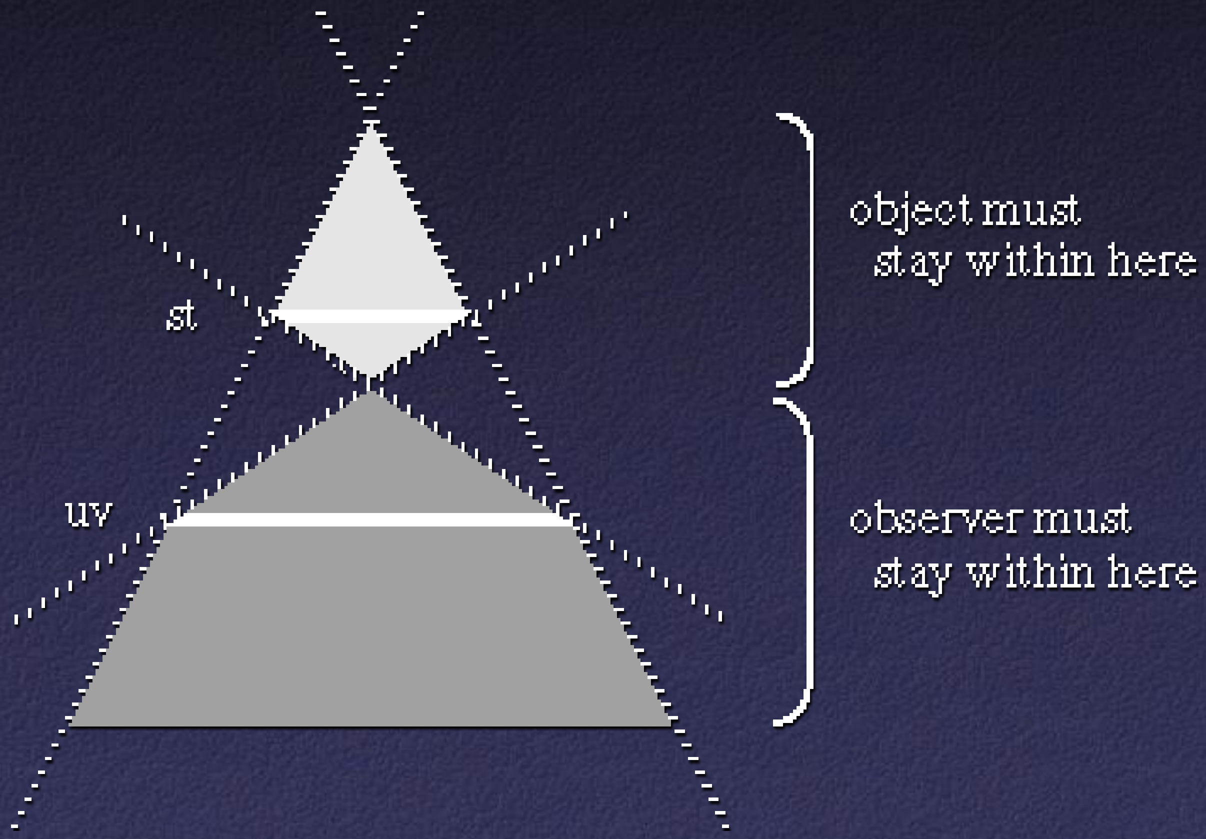
- Two planes, evenly sampled: “light slab”
- In general, planes in arbitrary orientations
- In practice, one plane = camera locations
  - Minimizes resampling

Light Field Two-Plane Parameterization

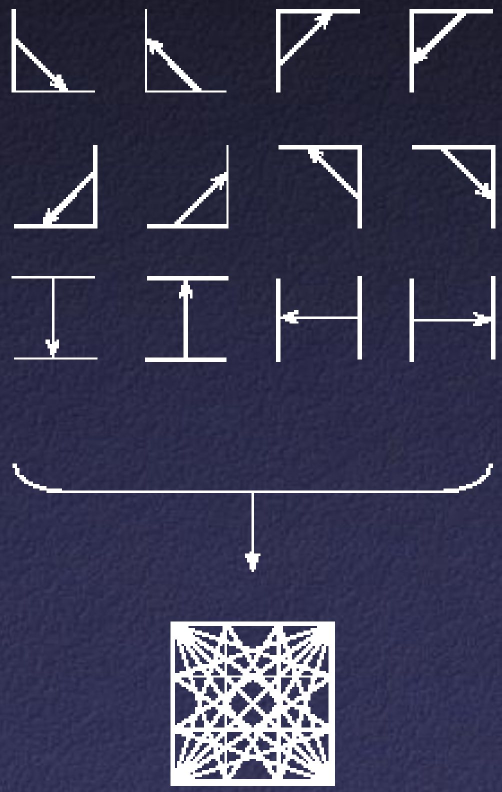


Light Field Coverage





Multi-Slab Light Fields

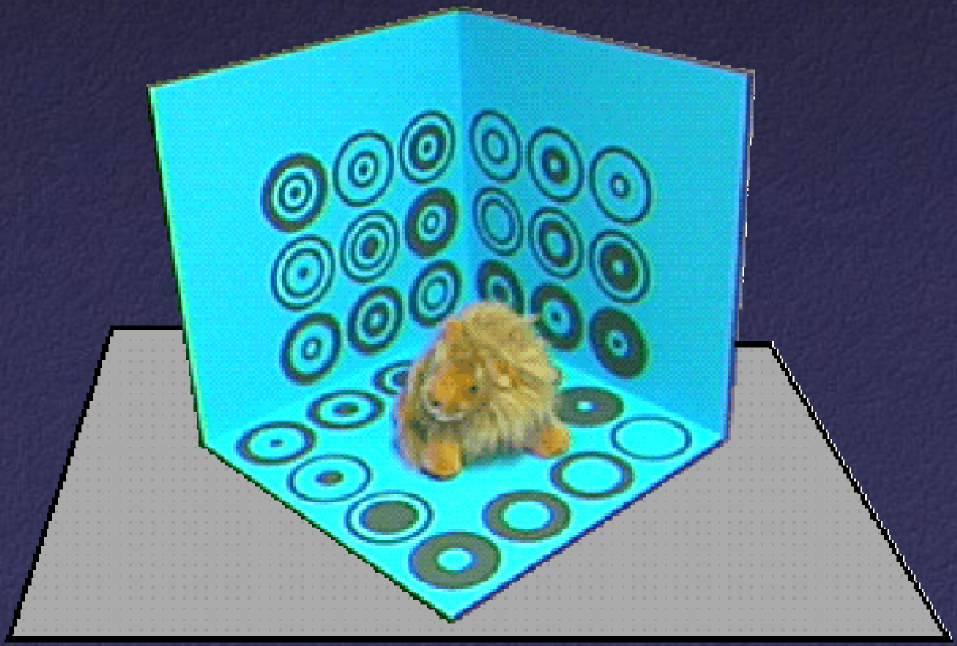


Lightfield Capture

- Capture a 2D set of (2D) images
- Choices:
  - Camera motion: human vs. computer
  - Constraints on camera motion
  - Coverage and sampling uniformity
  - Aliasing

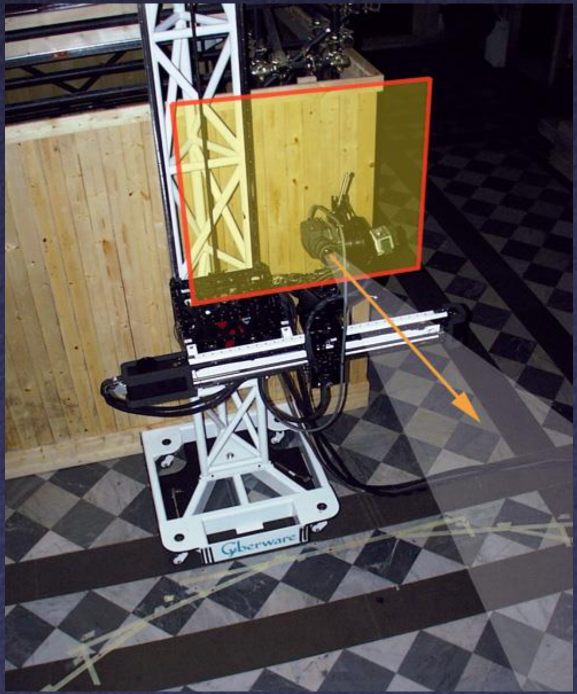
Lumigraph Capture

- Capture: move camera by hand
- Camera intrinsics assumed calibrated
- Camera pose recovered from markers



Lightfield Capture

- Levoy 06:
  - Computer-controlled camera rig
  - Move camera to grid of locations on a plane



Lightfield Capture

- Acquire an entire light field at once
- Video rates
- Integrated MPEG2 compression for each camera

(Bennett Wilburn, Michal Smulski, Mark Horowitz)

Lightfield Capture

Lytro

Lightfield Compression

- Compress individual images (JPEG, etc.)
- Adapt video compression to 2D arrays
- Decomposition into basis functions
- Vector quantization

Lightfield Rendering

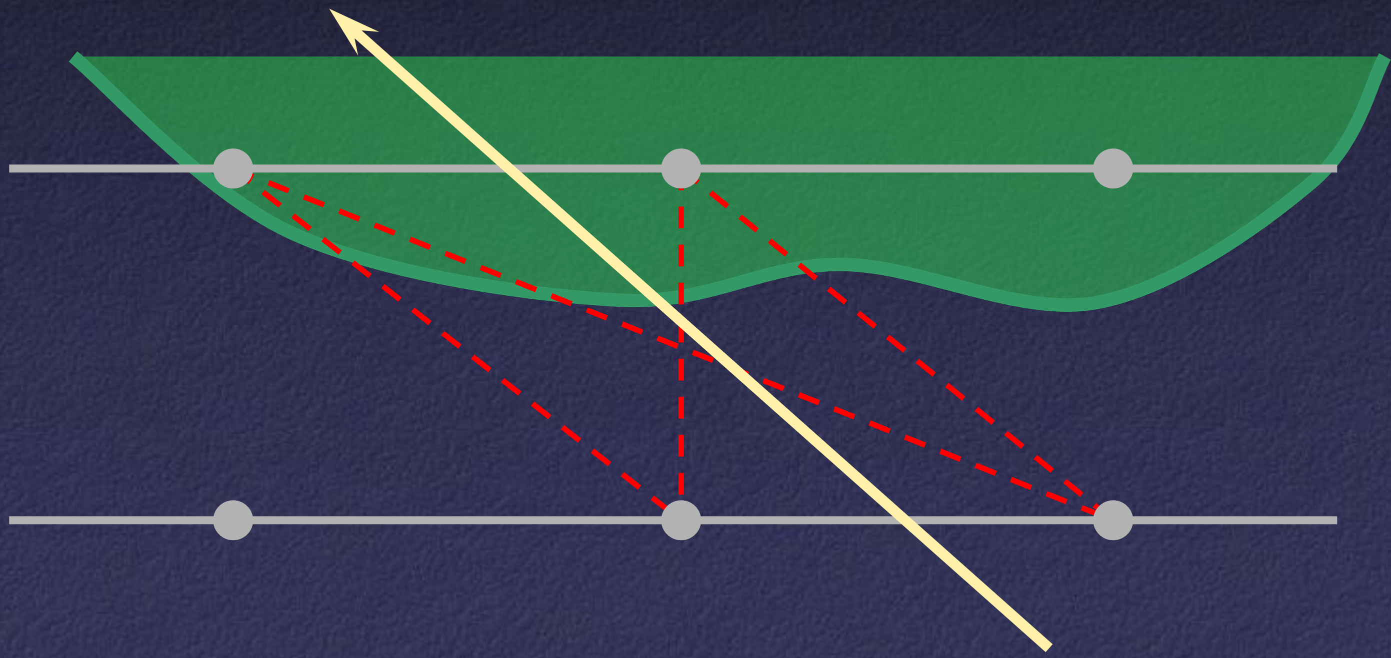
- How to select rays?
- How to interpolate

Lightfield Rendering

- For each desired ray:
  - Compute intersection with (u,v) and (s,t) planes
  - Take closest ray
- Variants: interpolation
  - Bilinear in (u,v) only
  - Bilinear in (s,t) only
  - Quadrilinear in (u,v,s,t)

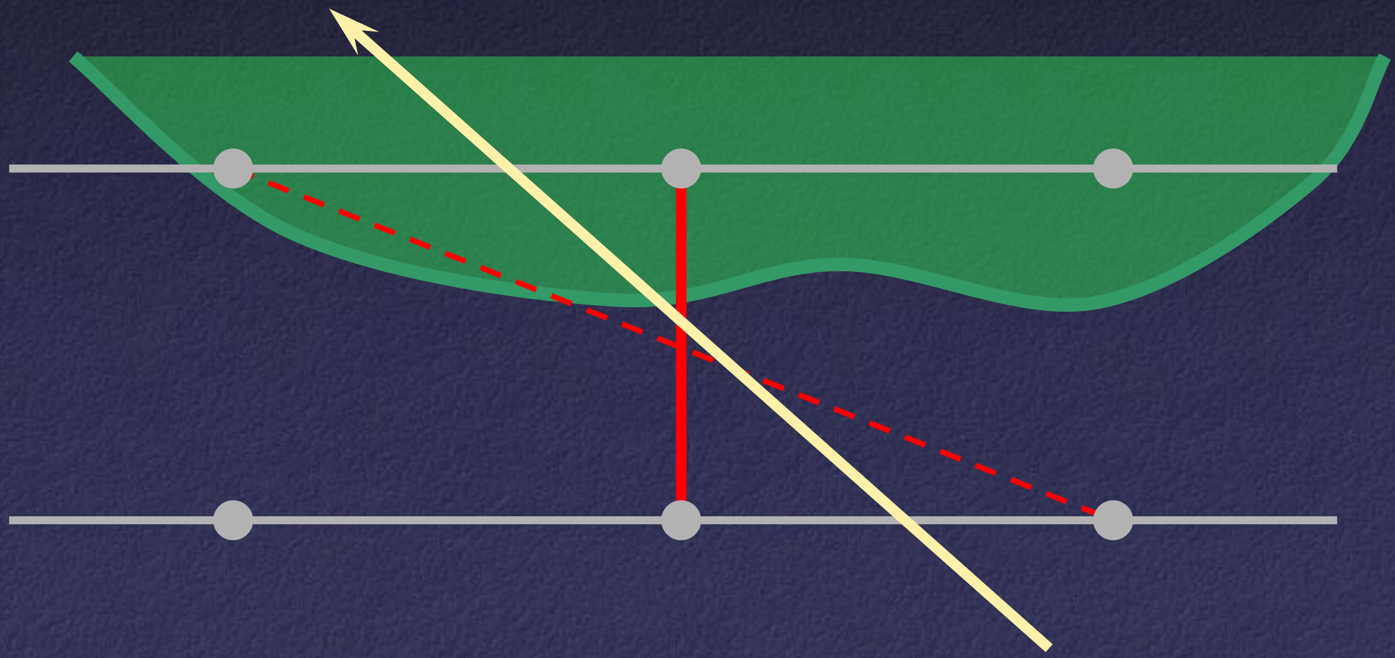
Lumigraph Rendering

- Use rough depth information to improve rendering quality



Lumigraph Rendering

- Use rough depth information to improve rendering quality



Lumigraph Rendering

Without using geometry

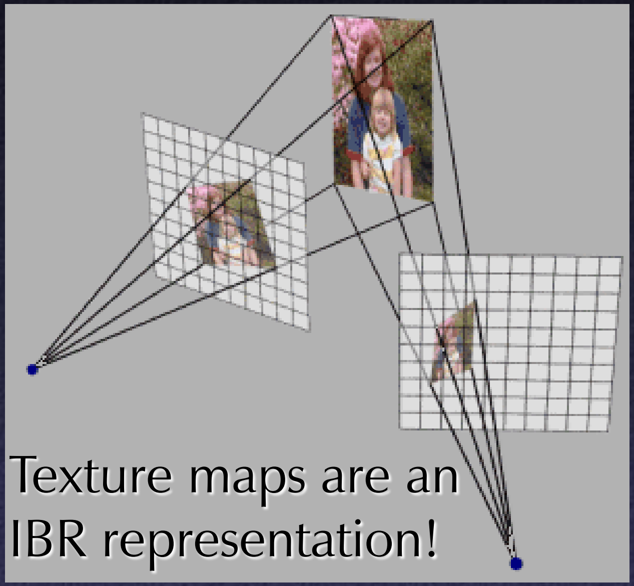
Using approximate geometry

Lightfields

- Advantages:
  - Simpler computation vs. traditional CG
  - Cost independent of scene complexity
  - Cost independent of material properties and other optical effects
  - Avoid hard vision problems
- Disadvantages:
  - Static geometry
  - Fixed lighting
  - High storage cost

Other IBR Representations

- Texture maps
- VDTMs
- Surface lightfields
- Unstructured lightfields
- Concentric mosaics
- Panorama
- Etc.



[McMillan]

IBR Summary

- Advantages
  - Photorealistic - by definition
  - Do not have to create 3D detailed model
  - Do not have to do lighting simulation
  - Performance independent of scene
- Disadvantages
  - Static scenes only
  - Real-world scenes only
  - Difficult for scenes with specularities, etc.
  - Limited range of viewpoints
  - Limited resolution