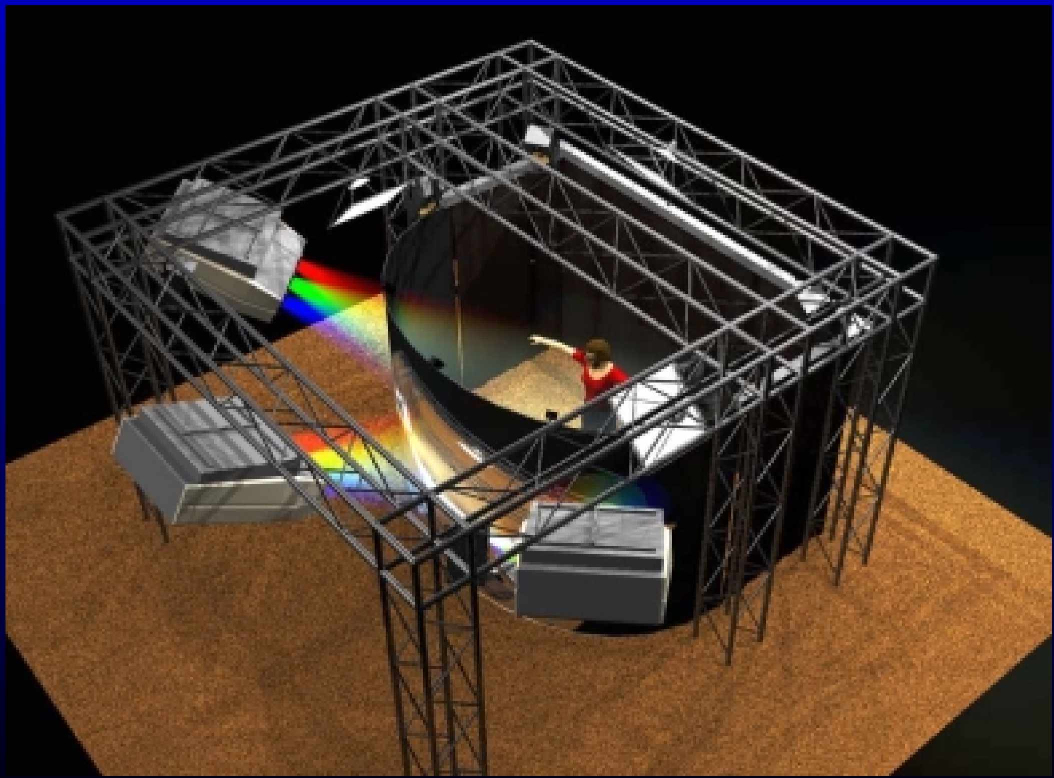
Hybrid Rendering for Collaborative, Immersive Virtual Environments

Stephan Würmlin
wuermlin@inf.ethz.ch



computer graphics group

ETH Eidgenössische Technische Hochschule Zürich

Outline

- Rendering techniques
  - GBR, IBR and HR
- From images to models
- Novel view generation
- Putting it all together
  - Image-based Visual Hulls
- Hybrid Rendering in VR?

Rendering techniques

- Geometry-based Rendering
  - simulation of light’s interaction with objects in a scene
  - based on *geometric models, lighting* …
- Image-based Rendering
  - (partial) reconstruction of the plenoptic function
  - based on *images*

Rendering techniques

Geometry-based Rendering

- Scene — Scene description in terms of geometry, surface properties, lighting conditions
- Discretization — Sampling (tesselation) optimized regarding *geometric properties*
- Representation — Set of *primitives* (polygons, textures, light sources)
- Image Synthesis — **Conventional Rendering**

Rendering techniques

GBR [2]

- Rendering cost dependent on scene complexity and lighting
- Visual realism due to texture mapping
  - different techniques available (see GDV I)
- Overhead if rendering primitives are smaller than screen resolution
- Requires 3D surface representation

Rendering techniques

Image-based Rendering

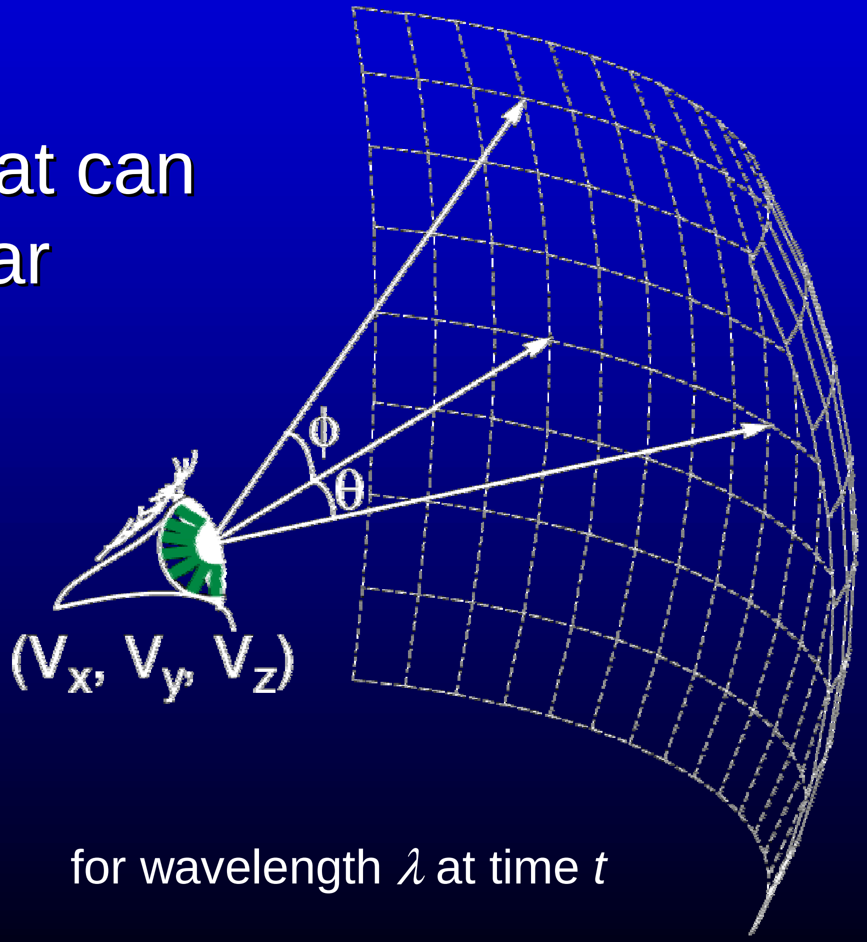
- Scene — Scene description in terms of the plenoptic function
- Discretization — Sampling optimized regarding *screen resolution*
- Representation — Set of *n-dimensional samples*
- Image Synthesis — **Reconstruction**

Rendering techniques

IBR [2]

- The plenoptic function
  - describes everything that can be seen from a particular viewing position

$$I=P(x,y,z,\theta,\rho,\lambda,t)$$



for wavelength λ at time t

Rendering techniques
IBR [3]

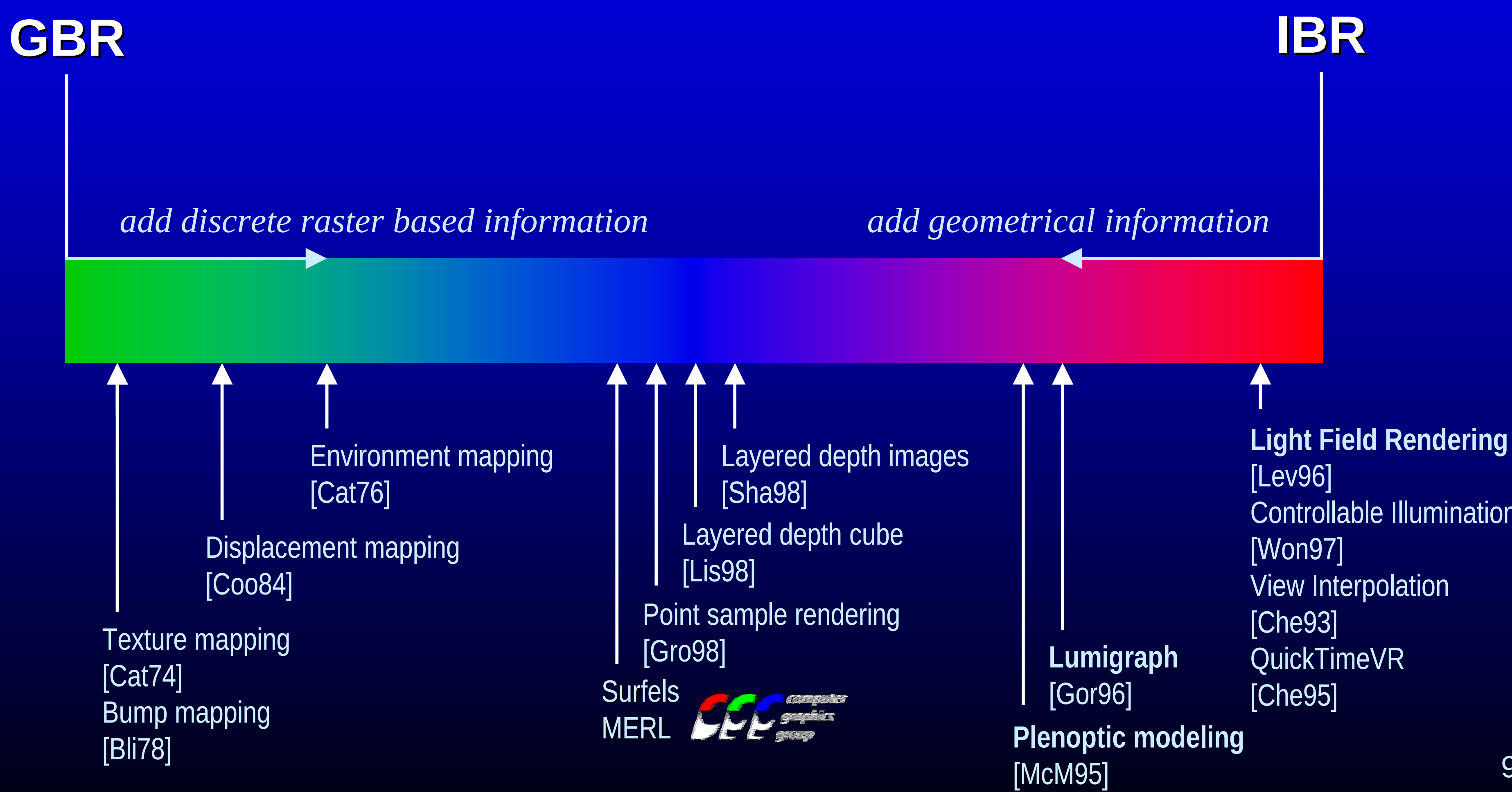
- Rendering cost independent of scene complexity
- No geometric model needed
- Improved realism through real world data

- Large storage requirements
- Limited flexibility (eg. lighting, reflectance)

by Matthias Zwicker

Rendering techniques

Hybrid Rendering



Rendering techniques

Hybrid Rendering [2]

- Combination of IBR and GBR
  - Extraction of **geometric** information from images
  - Extraction of **texture** information from images

- Rendering from novel views using available information

From images to models

- Geometric information from multiple images
  - (partial) reconstruction of 3D models

- Number of different techniques:
  - volumetric models from silhouettes
  - 3D surface points from stereo correspondence

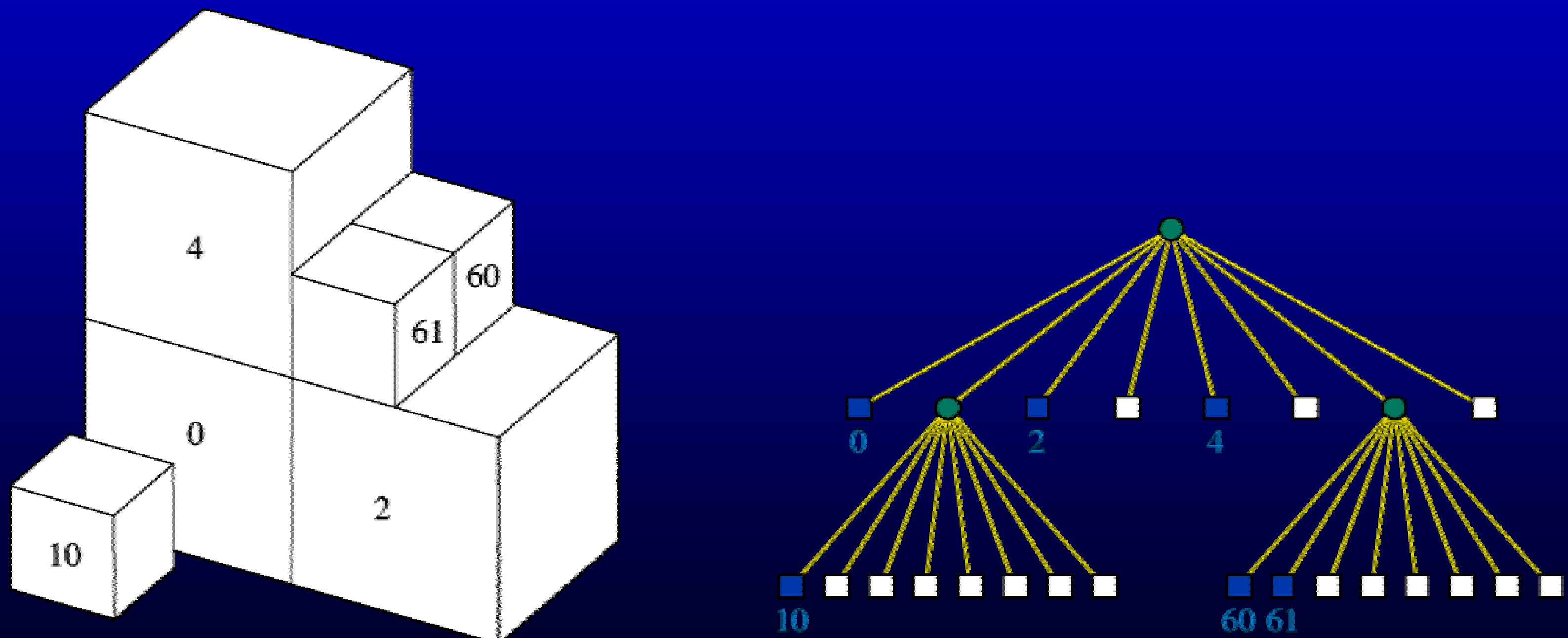
Volumetric models

- 3D volumetric model from silhouettes
  - extract binary silhouette using blue-screen (chroma-keying) technique
  - each silhouette + camera center defines enclosing conic region of space
  - intersection of cones ➡ bounding volume
  - octree construction in hierarchical coarse-to-fine fashion: *cube subdivisions*

From images to models

Volumetric models [2]

- Octree example and representation



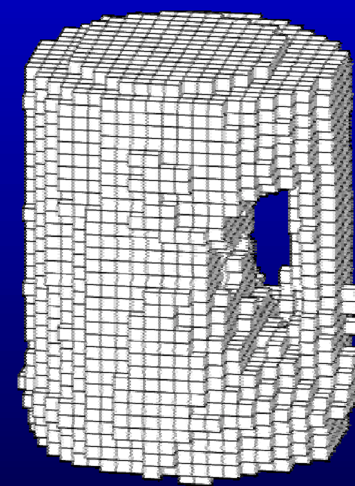
From images to models

Volumetric models [3]

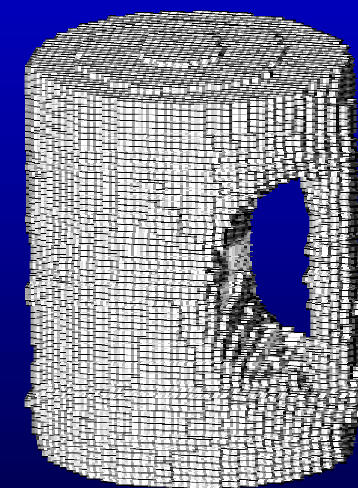
- Results

Input image

silhouette image
(non-white areas are considered part of object)



lower-res octree model
(5 levels)



higher-res octree model
(6 levels)

Volumetric models [4]

- Advantages
  - simple to implement
  - fast execution
  - complete (closed) surface
- Disadvantages
  - only produces *line hull* (no cavities)
  - possible misclassification in silhouette extraction
  - limited resolution

From images to models

3D surface points

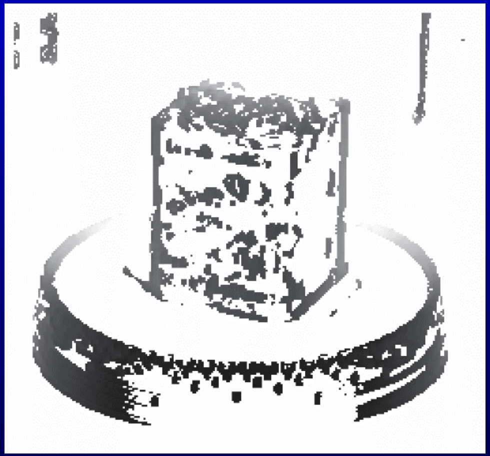
- Integration of depth maps into a complete 3D model
  - Computation of dense *disparity maps* from pairs of images using correlation
  - Backprojection into object space
- Convert to 3D points lying on the surface of the object

From images to models

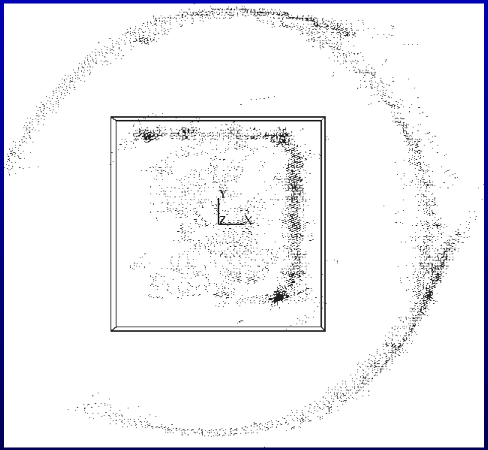
3D surface points [2]

- Results

first image
in sequence



depth map
from flow
(darker is nearer)



top view of
3D point cloud

From images to models

3D surface points [3]

- Advantages
  - gives detailed surface estimates
  - multi-view aggregation improves accuracy
- Disadvantages
  - fails in textureless areas
  - sparse, incomplete surface

Novel view generation

- Ray Databases
  - Light Field Rendering [Levoy et al.]
  - The Lumigraph [Gortler et al.]
- Image Warping
  - transforming the points of one image to their appropriate position in another image [McMillan & Bishop]

Novel view generation

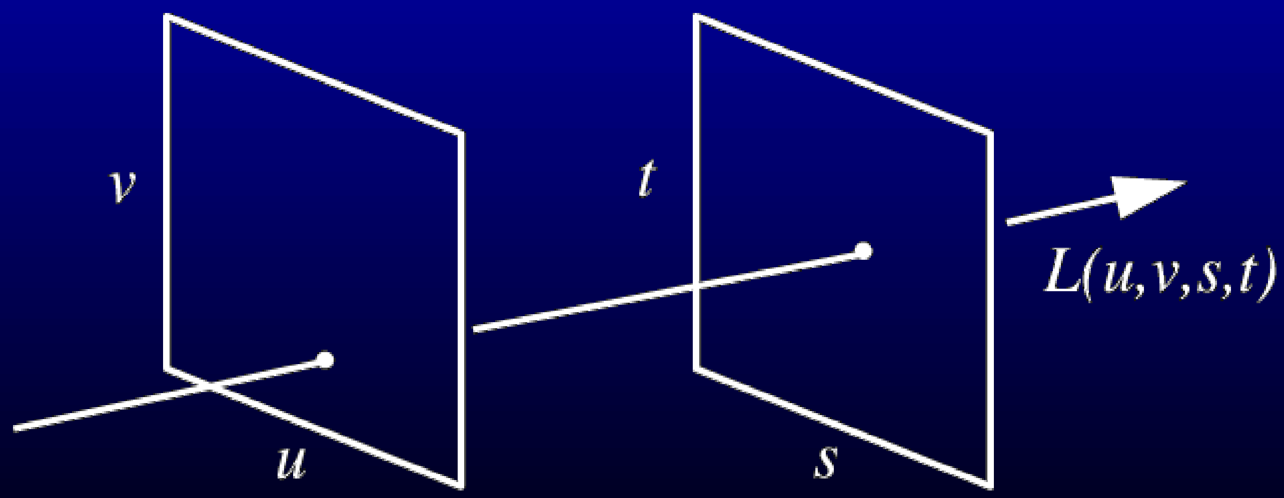
Light Field Rendering

- Pure IBR technique
  - no depth information
- 4D approximation of plenoptic function
- Database of rays
- Quadrilinear interpolation for novel views
- Large arrays of images needed

Novel view generation

Light Field Rendering [2]

- Sampling on rays
  - light slab
  - regular grid of intersection points



Novel view generation

Light Field Rendering [3]

- Results

Demo

Interactive
Light Field Viewer

program and light field data set:
Stanford Computer Graphics Laboratory

Novel view generation

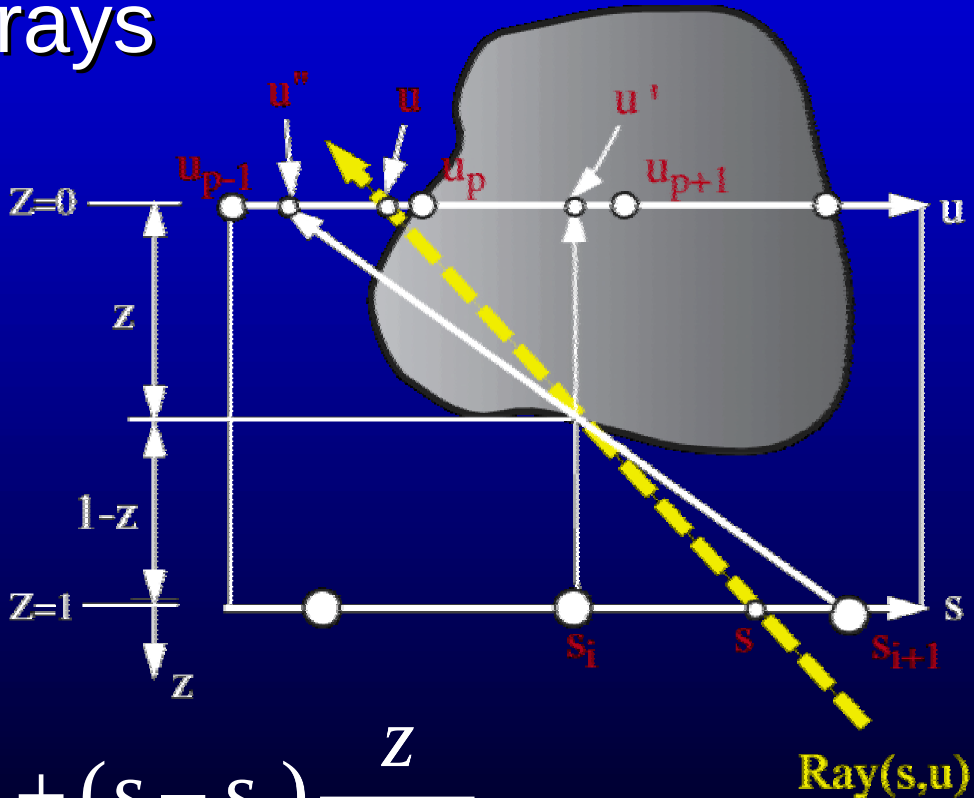
The Lumigraph

- Close to the Light Field approach
  - but geometric information for depth correcting images
    - ➡volumetric model
- Real-world acquisition with hand-held camera
- Discretization with coeff. and basis functions
- Lumigraph construction similar to multi-dimensional *scattered data approximation*
  - hierarchical algorithm: **splat - pull - push**

Novel view generation

The Lumigraph [2]

- Depth correction of rays



intersection at: $u' = u + (s - s_i)\dfrac{z}{1-z}$

Novel view generation
The Lumigraph [3]

- Results

no depth correction

with depth correction

Novel view generation

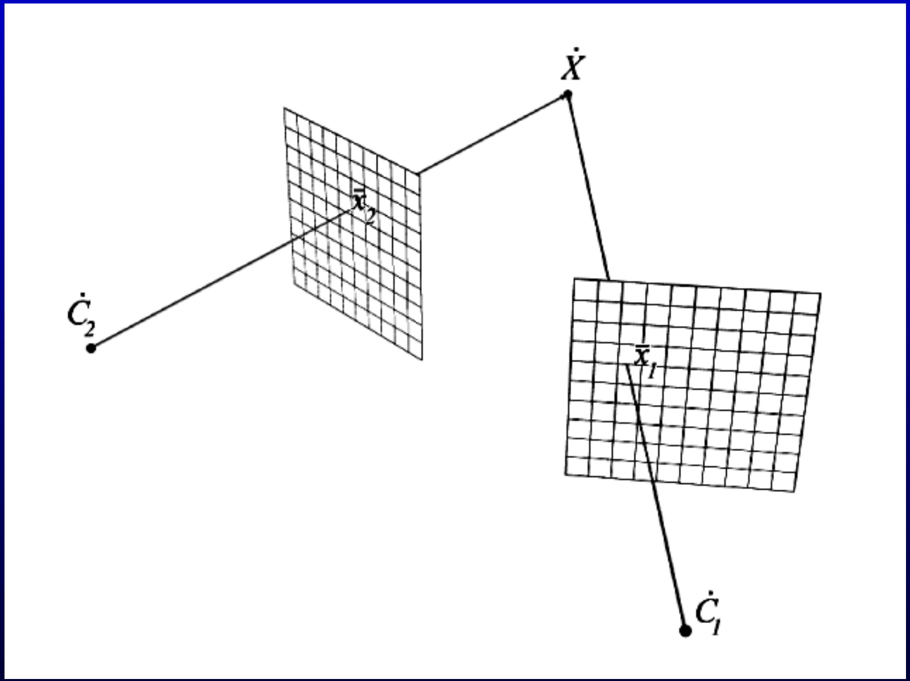
Image Warping

- Mapping points from a reference image to a desired image
  - depth information or correspondences needed
- Not a one-to-one mapping
  - thus resolving *visibility* is needed
- Reconstruction so that transformed points appear continuously in desired image

Novel view generation

Image Warping [2]

- Specification of a 3D point



$$\dot{X} = \dot{C}_1 + t_1 \mathbf{P}_1 \overline{x}_1 = \dot{C}_2 + t_2 \mathbf{P}_2 \overline{x}_2$$

Novel view generation

Image Warping [2]

- Planar image-warping equation

generalized disparity

$$\bar{x}_2 \doteq \delta(\bar{x}_1)\mathbf{P}_2^{-1}\left(\dot{C}_1 - \dot{C}_2\right) + \mathbf{P}_2^{-1}\mathbf{P}_1\bar{x}_1$$

per pixel shift proportional to disparity in direction of epipole

planar projective transformation

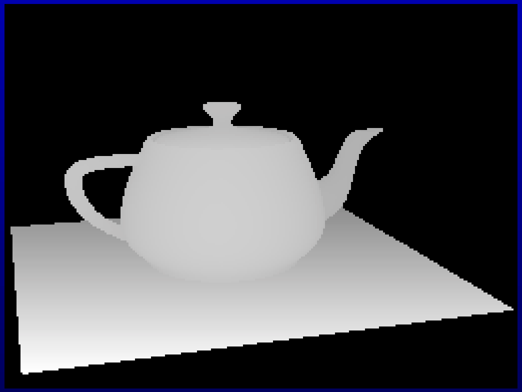
Novel view generation

Image Warping [3]

- Results

reference image
(note: only *one* image)

disparity data
(stored as gray scale image)



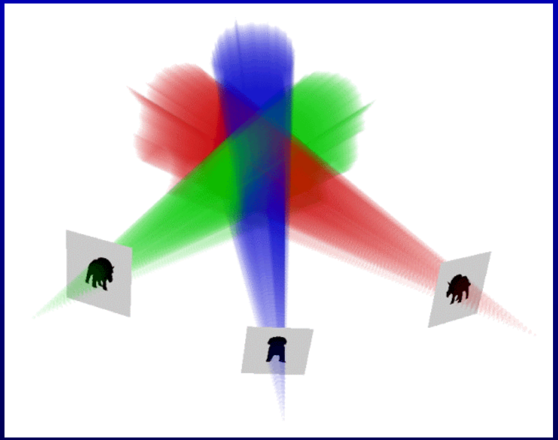
Demo

Image Warping Applet
MIT, Leonard McMillan

Putting it all together:

Image-based Visual Hulls

- The Visual Hull
  - intersection of silhouette volumes seen from multiple points of view



- Image-based representation
  - silhouette image with occupancy intervals at every pixel
  - stored as pairs of real numbers
    - third dimension is continuous

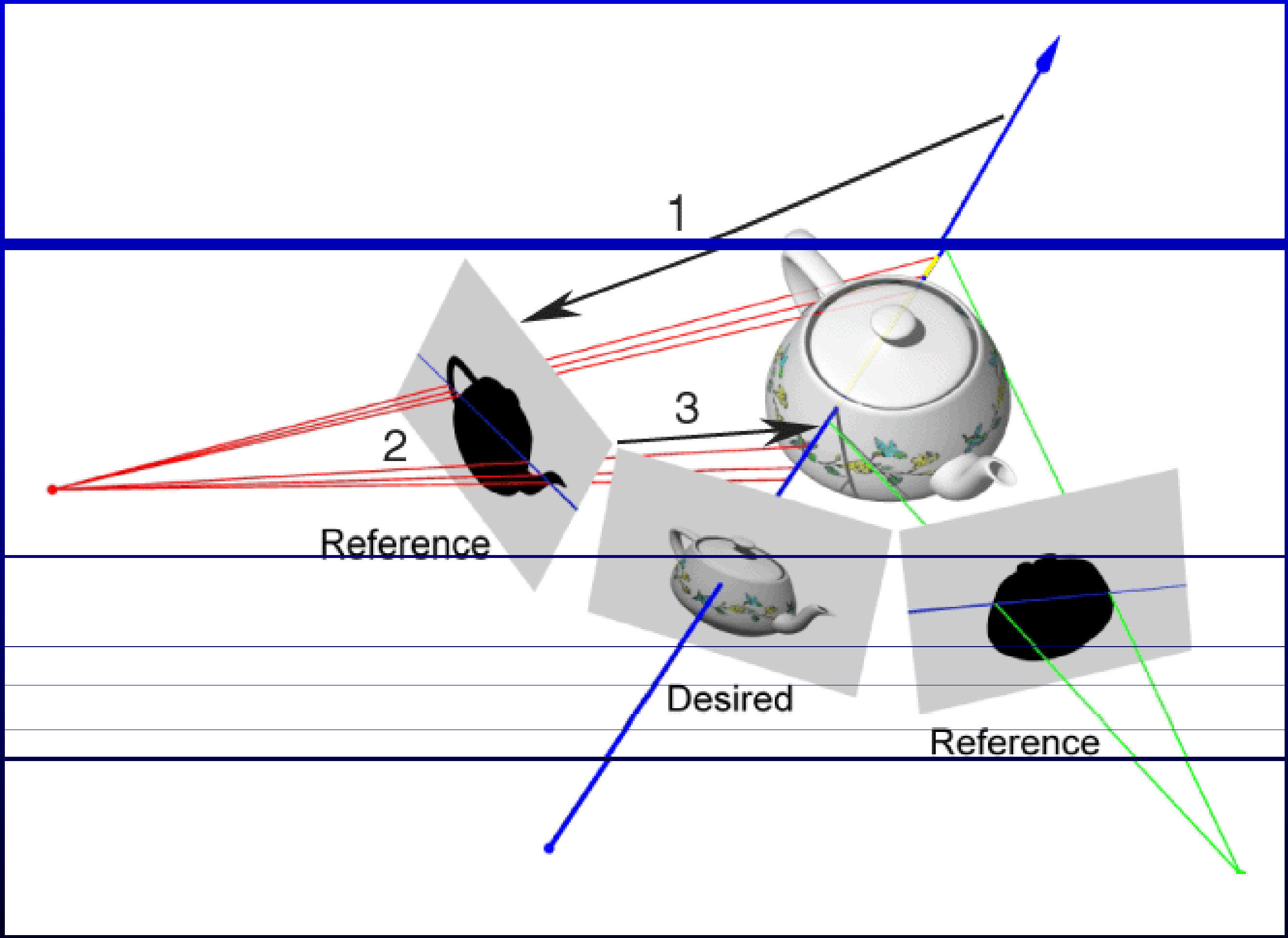
Putting it all together:

Image-based Visual Hulls [2]

- Creating Image-based Visual Hulls
  1. projection of the desired 3D viewing ray onto a reference image
  2. determination of the intervals where the projected ray crosses the silhouette
  3. lift back these intervals onto the desired ray where they can be intersected with intervals from other reference images

Putting it all together:

Image-based Visual Hulls [3]

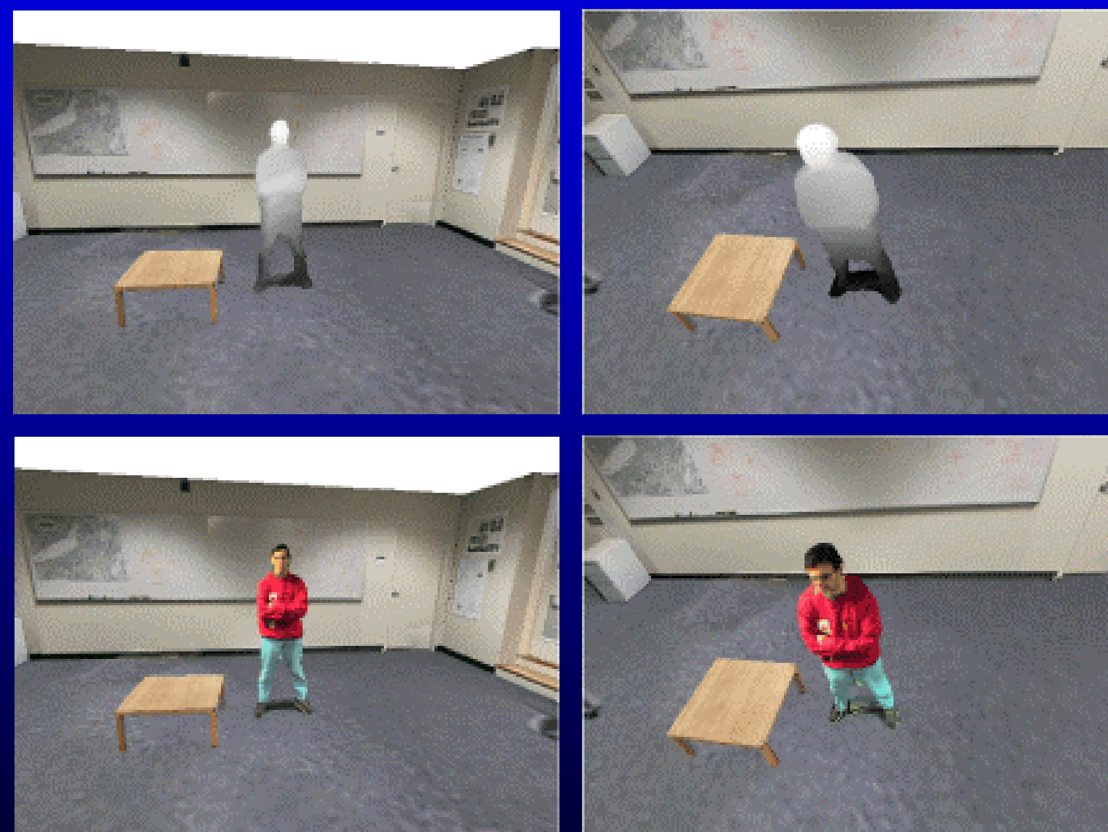


Putting it all together:

Image-based Visual Hulls [4]

- Results

base: 4 segmented reference images



upper: **depth maps** of the computed visual hulls
lower: **shaded renderings** from the same viewpoint

Putting it all together:

Image-based Visual Hulls [5]

MIT Visual Hull Project

Hybrid Rendering in VR?

- Rendering humans in VR?
- Two possibilities:
  - video-based avatars
    - billboarding
    - no occlusion detection
  - real humans
    - texture and geometric information from video streams
- …and we want real humans!