

More and More on Light Fields

Topics in Image-Based Modeling and Rendering
CSE291 J00
Lecture 4

Last Lecture

- Re-review with emphasis on radiometry
- Mosaics & Quicktime VR
- The Plenoptic function
- The main idea of the lumigraph/light field rendering.

Announcements

- Mailing list: cse291-j@cs.ucsd.edu

If you're not on it, and want to be added (e.g. auditing, and not on course list, send the email msg to majordomo@cs.ucsd.edu with body saying:

subscribe cse291-j my_email@something.something

- Class presentations: requests from

1/23 Diem Vu

1/28 Satya Malick

1/30 Jin-Su Kim

2/4 Sameer Agarwal

2/18 Jongwoo Lim

2/20 Yang Wu

2/25 Peter Schwer

3/4 Cindy Xin Wang

This lecture

S. J. Gortler, R. Grzeszczuk, R. Szeliski, M. F. Cohen. The Lumigraph, SIGGRAPH, pp 43--54, 1996

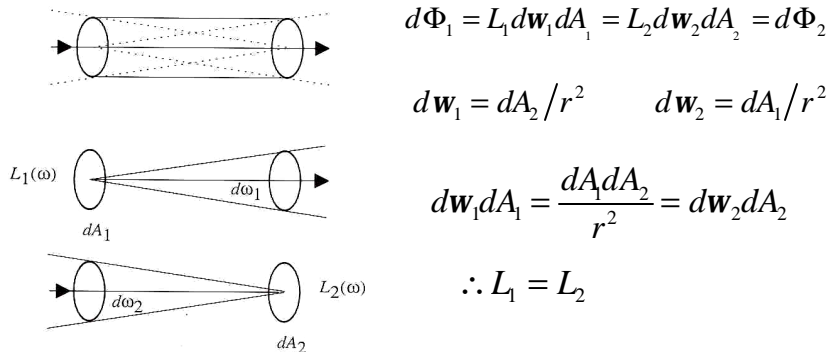
M. Levoy, P. Hanrahan, Light Field Rendering, SIGGRAPH, 1996

D. Wood, D. Azuma, W. Aldinger, B. Curless, T. Duchamp, D. Salesin, and W. Stutzle. Surface light fields for 3D photography, SIGGRAPH, 2000.

Aaron Isaksen, Leonard McMillan, Steven J. Gortler, Dynamically reparameterized light fields, SIGGRAPH 2000, pp 297 - 306

Radiance properties

- In free space, radiance is constant as it propagates along a ray
 - Derived from conservation of flux
 - Fundamental in Light Transport.



CS348, Fall 2001

© David Kriegman, 2001

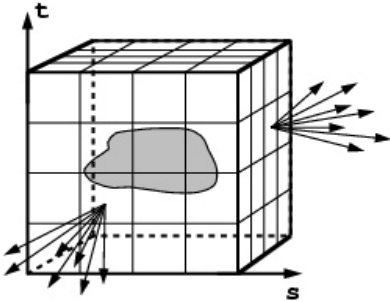
Light Field/Lumigraph Main Idea

- In free space, the 5-D plenoptic function can be reduced to a 4-D function (radiance) on the space of light rays.
- Camera images measure the radiance over a 2-D set – a 2-D subset of the 4-D light field.
- Rendered images are also a 2-D subset of the 4-D lumigraph.

CS348, Fall 2001

© David Kriegman, 2001

A cube of light



- All light from an object can be represented as if it were coming from a cube
- Each point on the cube has a 2-D set of rays emanating from the cube.

CS348, Fall 2001

© David Kriegman, 2001

4D Light Field

Modeling:
Move camera
center over a 2-D
surface.
 $2\text{-D} + 2\text{-D} \rightarrow 4\text{-D}$

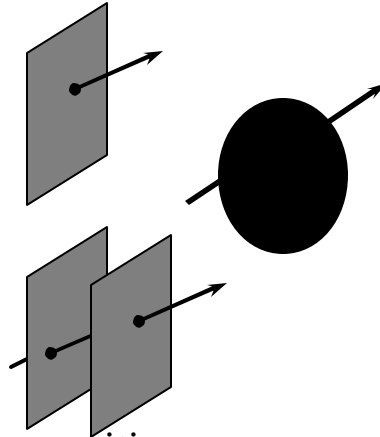


CS348, Fall 2001

© David Kriegman, 2001

Parameterization ray space

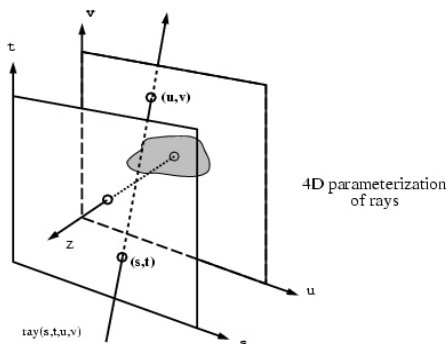
- Point / angle
- Two points on a sphere
- Points on two planes
- Original images and camera positions
- Plucker Coordinates



CS348, Fall 2001

© David Kriegman, 2001

Two plane parameterization



- Rays are parameterized by where they intersect these planes.
- Any point in the 4D Light field/Lumigraph is identified by its coordinates (s,t,u,v)
- Continuous light field
 $L(s,t,u,v)$
- **Cube around object – six slabs.**

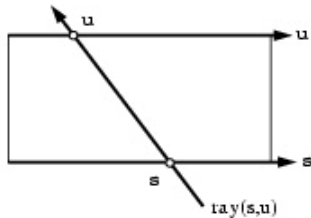
Note that the lumigraph and light field papers interchange roles of (s,t) and (u,v)

CS348, Fall 2001

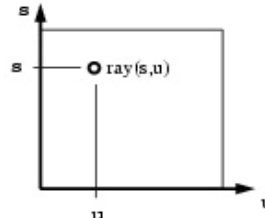
© David Kriegman, 2001

The Ray Space

2D Slice of 4D Space



ray space in 2D



Bear in mind that a ray in \mathbb{R}^3 is represented as a point in the 4-D ray space (s,t,u,v) .

Constructing the light field

Move camera over 2-D surface and sample

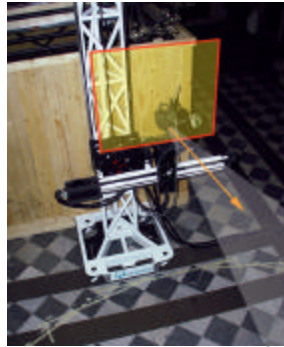
1. Build a gantry (light field)
2. Use vision techniques to estimate camera position – non-uniform sampling (lumigraph)
3. Rendered images (both)

Light Field Rig

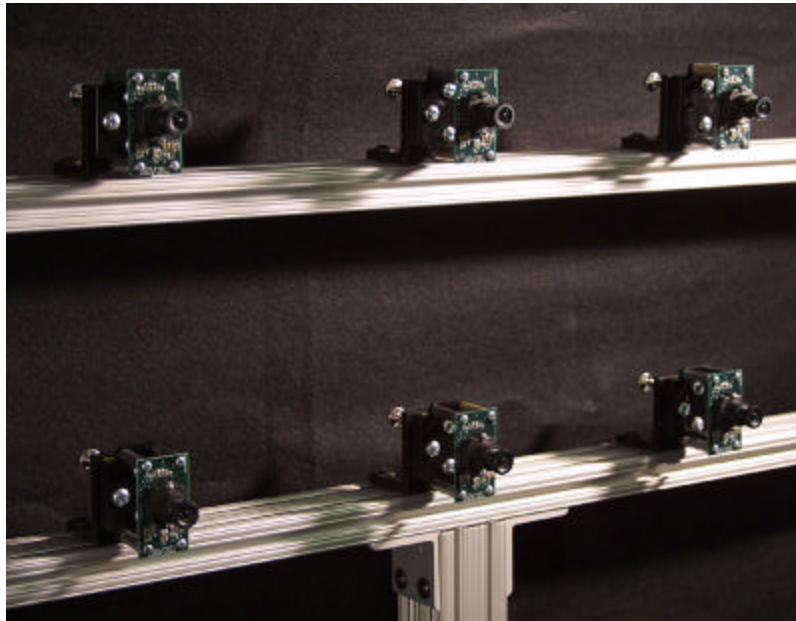
- Capture:
 - Computer-controlled camera rig
 - Move camera to grid of locations on a plane



CS348, Fall 2001

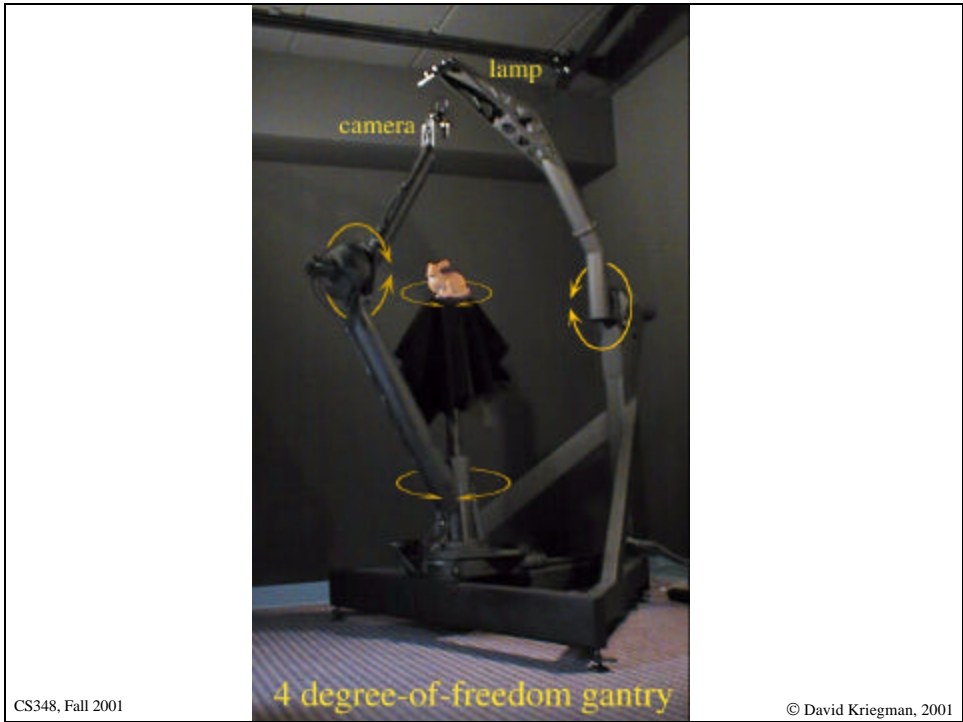


David Kriegman, 2001

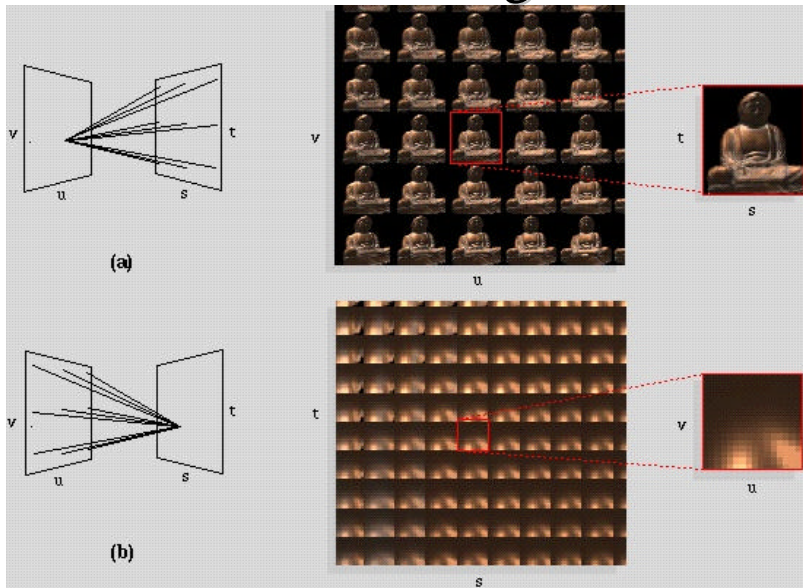


CS348, Fall 2001

© David Kriegman, 2001



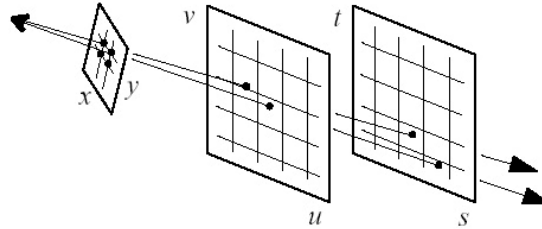
Two views of Light Field



CS348, Fall 2001

© David Kriegman, 2001

Rendering



- For each desired ray:
 - Compute intersection with (u,v) and (s,t) planes
 - Take radiance at closest ray
- Can be computed using texture mapping hardware
- Variants: interpolation
 - Bilinear in (u,v) only
 - Bilinear in (s,t) only
 - Quadrilinear in (u,v,s,t)
- Apply band pass filter to remove HF noise that may cause aliasing

CS348, Fall 2001

© David Kriegman, 2001

Compression

- Two-stage decompression
 - Decode entropy coding while loading file
 - End up with codebook and code indices packed into 16-bit words.
 - De-quantization
 - Engine requests samples of light field
 - Subscript index calculated
 - Look up index in codebook
 - Get vector of sample values
 - Subscript again for requested sample
- Digitization
 - Slabs 4
 - Images/Slab 32x16
 - Pixels/Image 256x256
 - Total Size 402 MB
- Compression
 - VQ coding 24:1
 - Entropy coding 5:1
 - Total 120:1
 - Comp. Size 3.4 MB

CS348, Fall 2001

© David Kriegman, 2001

Compression Example



Original

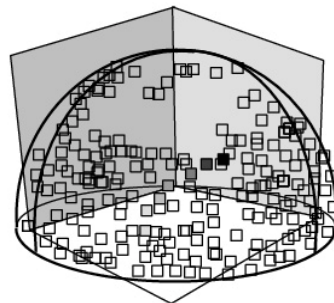
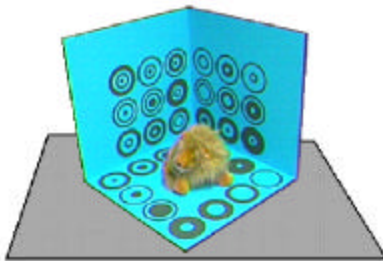


Compressed 120:1

CS348, Fall 2001

© David Kriegman, 2001

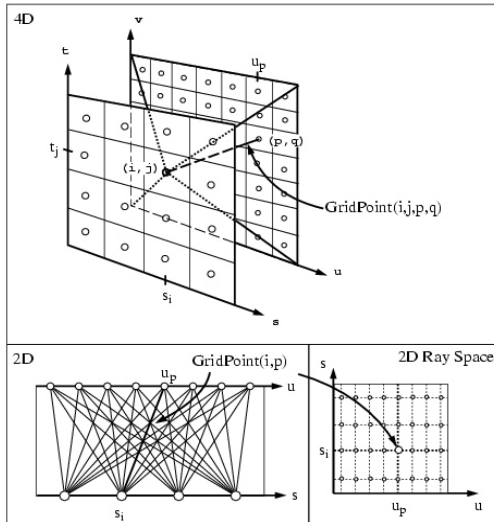
Lumigraph: Non-uniform sampling



CS348, Fall 2001

© David Kriegman, 2001

Sampling of rays is discrete



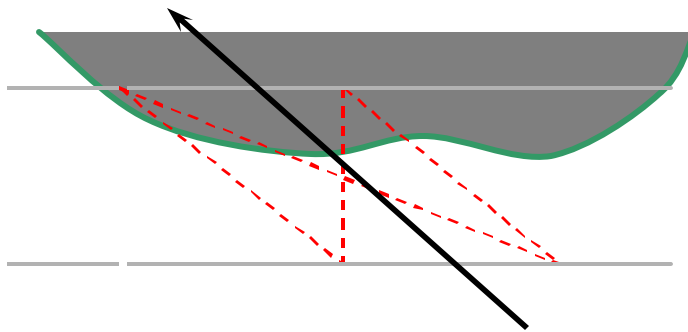
- Samples in the image (pixels) are not continuous
- The camera location is not continuous
- We have gaps in our lumigraph

CS348, Fall 2001

© David Kriegman, 2001

Lumigraph Rendering

- Use rough depth information to improve rendering quality

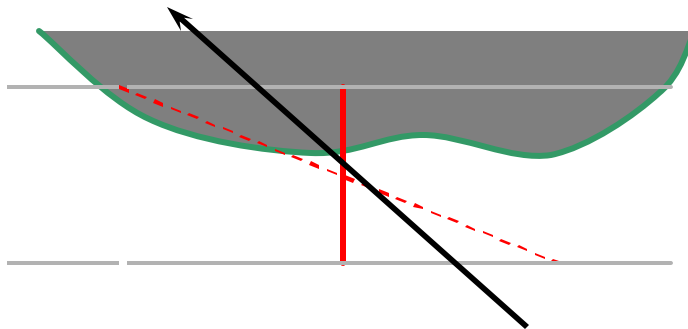


CS348, Fall 2001

© David Kriegman, 2001

Lumigraph Rendering

- Use rough depth information to improve rendering quality



CS348, Fall 2001

© David Kriegman, 2001

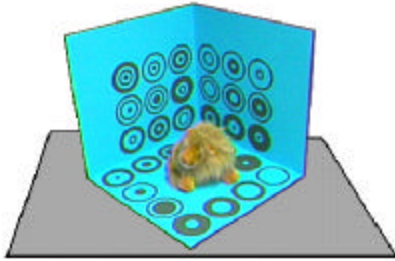
Lumigraph Construction

- Obtain rough geometric model
 - Chroma keying (blue screen) to extract silhouettes
 - Octree-based space carving
- Resample images to two-plane parameterization

CS348, Fall 2001

© David Kriegman, 2001

Camera Calibration and Pose Estimation

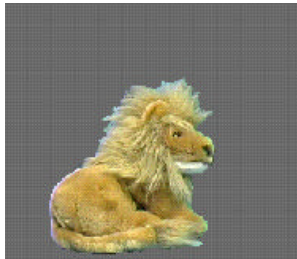


- A fixed lens is used to keep intrinsic parameters constant throughout the process
- Extrinsic parameters change constantly
- A previously developed algorithm in combination with a special capture stage is used to compute the Extrinsic parameters

CS348, Fall 2001

© David Kriegman, 2001

3D Shape Approximation



- Only a rough estimate of the shape is needed
- The octree construction algorithm is used
- The resulting volume construction is pictured above

CS348, Fall 2001

© David Kriegman, 2001

Lumigraph Rendering



Without using
geometry



Using approximate
geometry

CS348, Fall 2001

© David Kriegman, 2001

Rendered stereo pair using lumigraph



CS348, Fall 2001

© David Kriegman, 2001

Play light field movie

CS348, Fall 2001

© David Kriegman, 2001

**Dynamically Reparameterized
Light Fields**

CS348, Fall 2001

© David Kriegman, 2001

New parameterizations

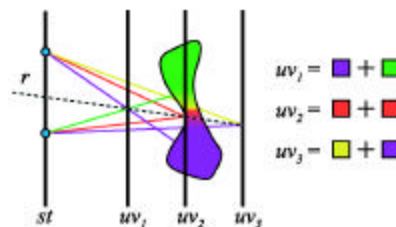
- Interactive Rendering
 - Moderately sampled light fields
 - Significant, unknown depth variation
- Passive autostereoscopic viewing
 - “fly’s eye” lens array
 - Computation for synthesizing a new view comes directly from optics of display device
- To achieve desired flexibility, do not perform aperture filter from original paper.

CS348, Fall 2001

© David Kriegman, 2001

Light Slabs (Again)

- Two-plane parameterization impacts reconstruction filters
- Aperture filtering removes high frequency data
- But if plane is wrong place, only stores blurred version of scene

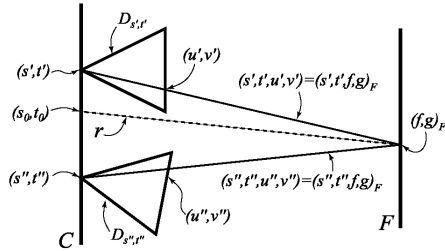


CS348, Fall 2001

© David Kriegman, 2001

Ray Construction

- Given a ray r , can find rays (s', t', u', v') and (s'', t'', u'', v'') in $D_{s', t'}$ and $D_{s'', t''}$ that intersect at the same $(f, g)_F$
- Combine values with a filter
- In practice, use more rays and weight the filter accordingly



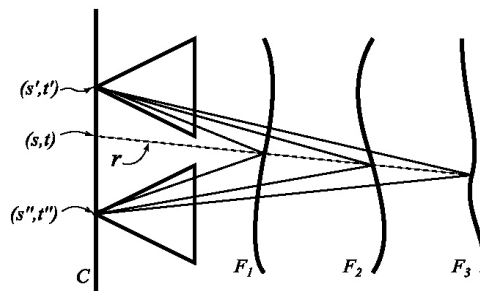
CS348, Fall 2001

© David Kriegman, 2001

Variable Focus

- Can create variable focus effects
- Multiple focal surfaces
- Can use one or many simultaneously

Rendering ray r



CS348, Fall 2001

© David Kriegman, 2001

Focal plane at multiple depths



Figure 8: A focal plane has been placed through the tree.

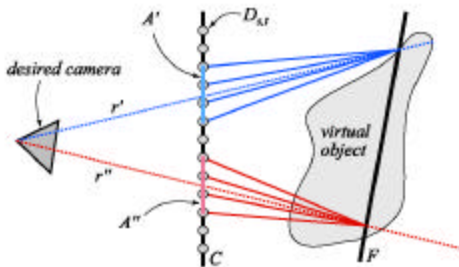


Figure 9: The same scene as Figure 8, but with the focal plane passing through the hills behind the island.

CS348, Fall 2001

© David Kriegman, 2001

Variable Aperture



- Real camera aperture produce depth-of-field effects
- Emulate depth-of-field effects by combining rays from several cameras
- Combine samples for each $D_{s,t}$ in the synthetic aperture to produce image

1. Render r' with aperture A' : in focus
2. Render r'' with aperture A'' : out of focus

CS348, Fall 2001

© David Kriegman, 2001

“Aperture” Size

Small aperture



Figure 9: The same scene as Figure 8, but with the focal plane passing through the hills behind the island.

Large aperture



Figure 6: By making the aperture very large, we are able to look through objects. In this case, there is a tree and island occluding the hills where the slight haze appears. Figures 8, 9, and 10 are the same scene with a smaller aperture; the tree is visible.

CS348, Fall 2001

© David Kriegman, 2001

Moving Toys animation: Focal plane, camera orientation, and camera position change



CS348, Fall 2001

© David Kriegman, 2001

By using the largest possible aperture, images are created with dramatic (limited) depth of field. These images use an aperture that incorporates all 256 cameras evenly. Note that we can see through the tree when the focal plane is at the hills in the background.

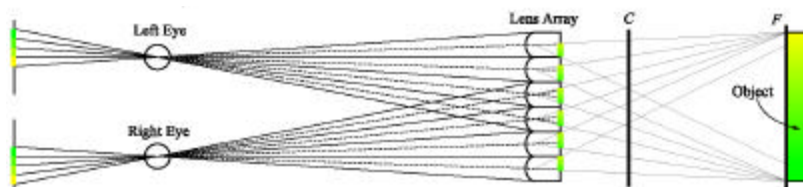


CS348, Fall 2001

© David Kriegman, 2001

Autostereoscopic Light Fields

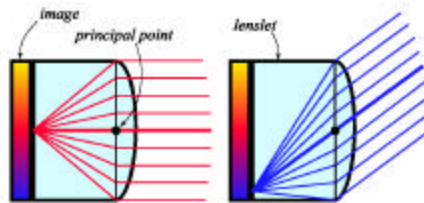
- Can think of the parameterization as a integral photograph, or the “fly’s eye view”
- Each lenslet treated as single pixel



CS348, Fall 2001

© David Kriegman, 2001

View-Dependent Color



- Each lenslet can have view dependent color
- Draw a ray through the principle point of the lenslet
- Use paraxial approximation to determine which color will be seen from a direction

CS348, Fall 2001

© David Kriegman, 2001

Unstructured Lumigraph Rendering

Unstructured lumigraph rendering Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, Michael Cohen SIGGRAPH 2001, pp: 425 – 432

- Further enhancement of lumigraphs:
 - do not use two-plane parameterization
- Store original pictures: no resampling
- Hand-held camera, moved around an environment

Plenoptic sampling , Jin-Xiang Chai, Shing-Chow Chan, Heung-Yeung Shum, Xin Tong, SIGGRAPH 2000, pp. 307 – 318

CS348, Fall 2001

© David Kriegman, 2001