2016 상반기 인턴 및 취업 설명회

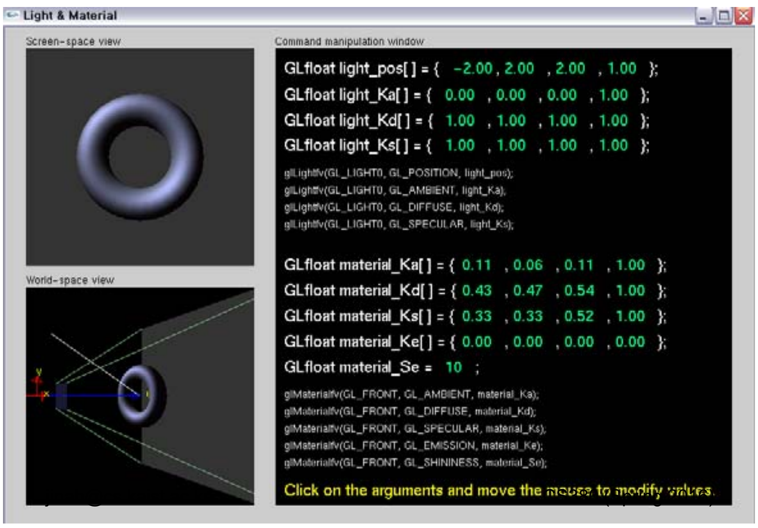
- 5월25일(수) 오후 1:00~6:00 (5시간)

- SW중심대학 사업의 일환으로
  산업협력 콘소시움을 조직하고
  학부학생 대상 인턴 및 취업 설명회 개최
- 목적:
  - 학생들에게 방학동안 인턴을 장려하고
    인턴을 수행할 회사의 바람직한 인턴 주제를 소개
  - 졸업 예정인 학생들의 취업 설명회

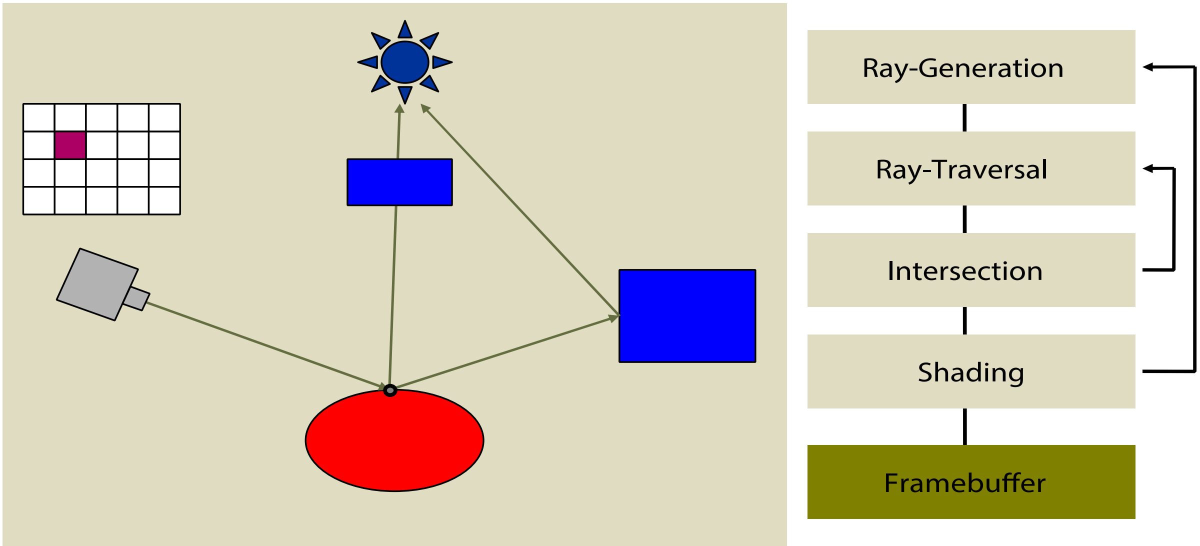
<Last lecture>

Demo

- Light-material
- Light-position

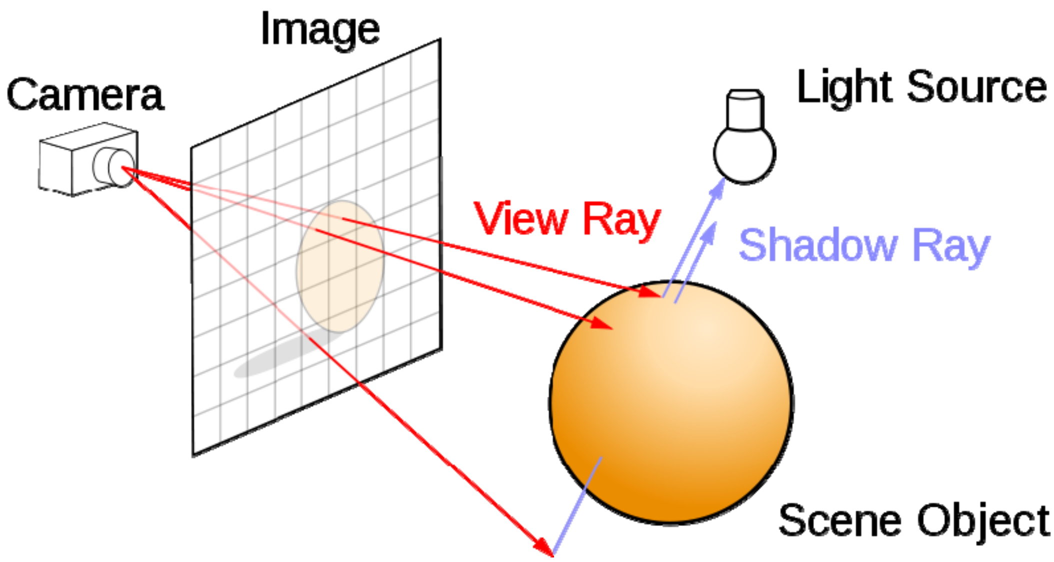


Ray tracing pipeline

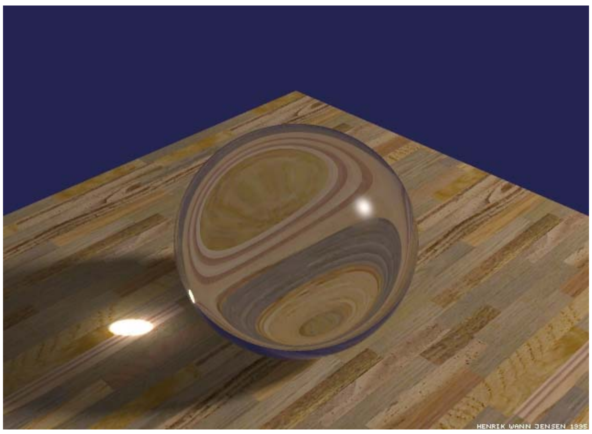


Courtesy of Hendrik Lensch

Ray tracing

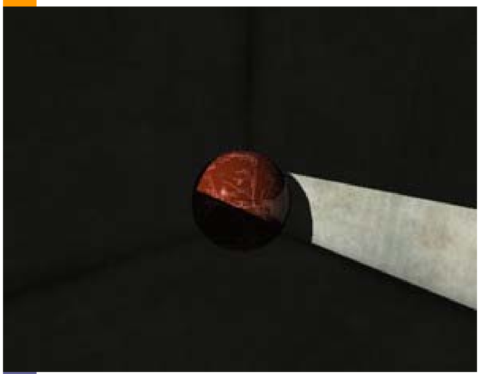


Caustics



Images created by Prof. Henrik Jensen @UCSD

Photon Mapping



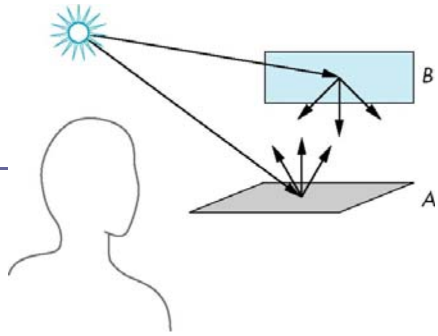
SoftImage (XSI)

Subsurface Scattering

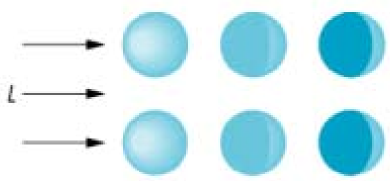
Siggraph 2001

Images created by Prof. Henrik Jensen @UCSD

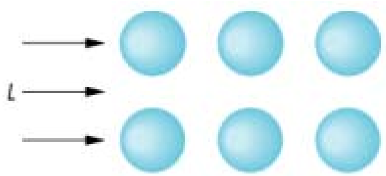
Light and Matter



- Rendering equation
  — very complex
- Ray tracing / Radiocity
  - global model



- Phong reflection model
  - local model
  - Rasterization-based



Light (Technical)

Chapter 21

May 19, 2016

Chapter 21

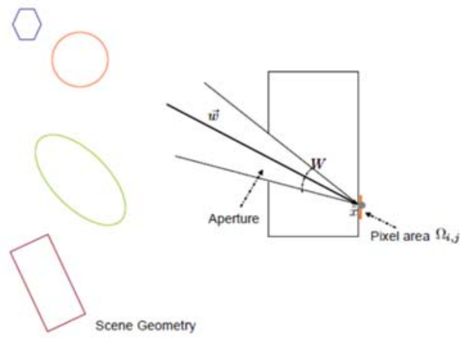
- More detail on how light and reflections are properly ‘measured’ and ‘represented’
- High quality rendering
  - Not with the rendering pipeline (as OpenGL)
- More fundamental aspect of deriving the appropriate equations
- Photon
- Irradiance, radiance
- Reflection, BRDF
- Light (ray) simulation

- Rendering equation

$$L^t(\tilde{x}, \vec{v}) \quad = \quad L^e(\tilde{x}, \vec{v}) + \int_H dw \; f_{\tilde{x},\vec{n}}(\vec{w}, \vec{v}) \; L^t(\vec{w}, \tilde{x}) \cos(\theta)$$

Figure 21.13: By computing multiple bounces, we can compute the correct distribution of light and darkness in an environment. From [72], ©ACM.

Sensors



- Camera ..:
  - finite aperture and finite shutter
  - Photon count at pixel (i,j)

$$\int_T dt \int_{\Omega_{i,j}} dA \int_W dw\ F_{i,j}(\tilde{x})\ L^t(\vec{w}, \tilde{x}) \cos(\theta)$$

Integration over the lens creates focus effects.

One of the first published images rendered using lots of rays per pixel.

On today's End-note

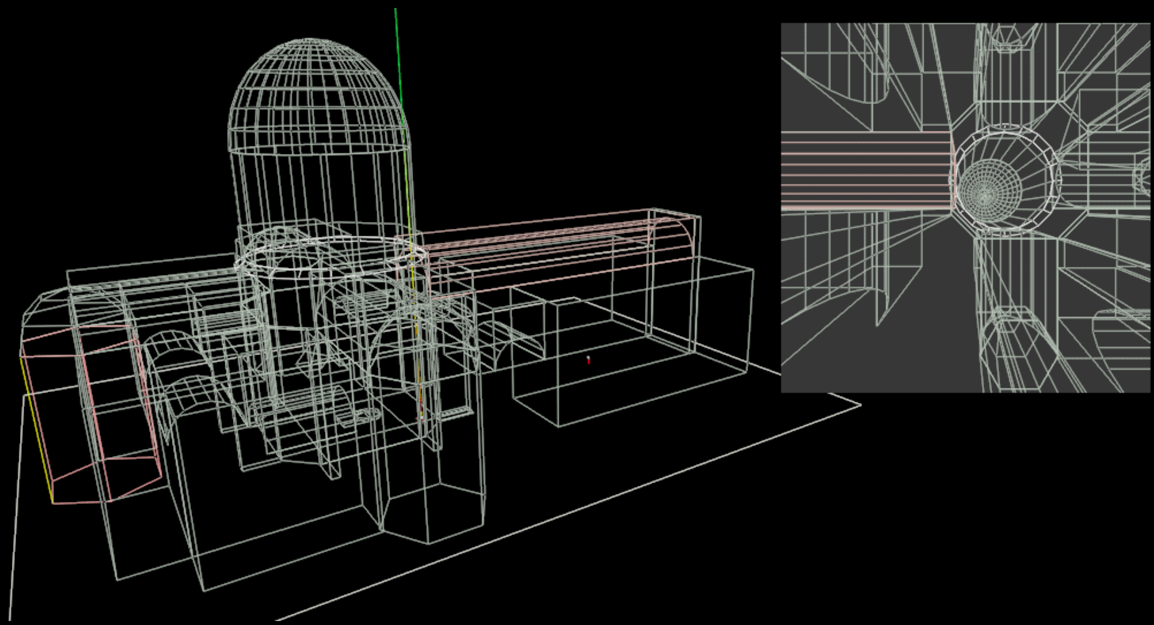
- Please write some comments on the technical animations you would watch today.
  - You may compare any pair or choose one
  - What the animation about
  - Any impressive parts?
  - Any disappointing aspects?
  - Etc …

Image-Based Rendering

- Image-based rendering (as well as image-based modeling & image-based lighting) works by extracting geometry, image texture, and global illumination information from analyzing photographic data.
- Some of the earliest experiments with IBR used photographs of real environments to extract depth maps of the environment and reproject pixels in 3D space.
  - The quality of these early examples of image-based renderings was very dependent on the number of views taken of the environment. But newer techniques can calculate the missing information and fill in approximate data.
- Gathering the information to be used in the IBR is an important step because it will define the quality of the final results.

Image-Based Rendering

- Fiat Lux (Paul Debevec UC at Berkley)
  - Siggragph Electronic Theater 1999
  - The film features a variety of dynamic objects realistically rendered into real environments, including St. Peter's Basilica in Rome. The geometry, appearance, and illumination of the environments were acquired through digital photography and augmented with the synthetic objects to create the animation.



The 3D model of St. Peter's Basilica derived from the panoramic imagery using Façade (a photogrammetric modeling system). The model was used to create 3D camera moves, determine the origin of illumination, and determine shadow and reflection placement.

View Interpolation

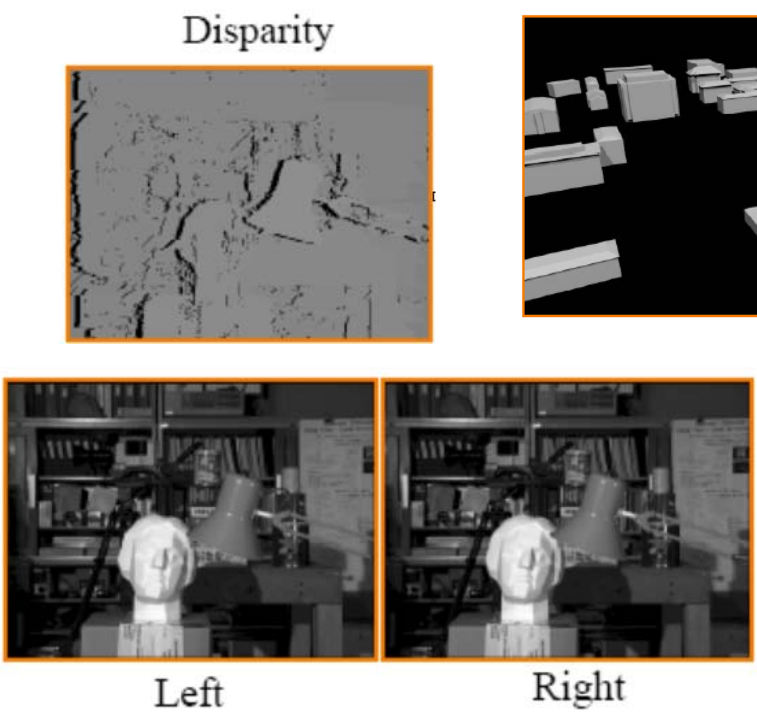


Image-based Modeling

- Image-based Tree Modeling
  - Ping Tan et al. (Hong Kong University of Science and Technology, Microsoft)

- Image-based procedural modeling of facades
  - Pascal Muller et al (ETH Zurich)

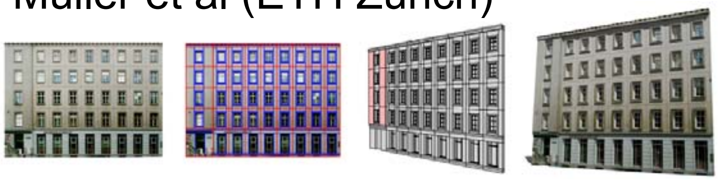


Image-Based Rendering

- IBR
  - Images are the primary data used for rendering
  - Motivation
    - Rendering is proportional to the number of pixels rendered, not to the number of vertices in a geometrical model. [more efficient]
    - Many objects (clouds, fur) are difficult to represent with polygons.
    - Actual photos can be used to enhance the level or realism.

Image-Based Rendering

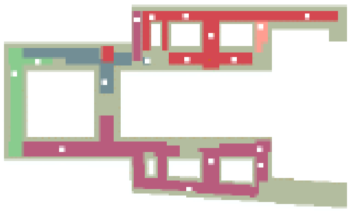
- IBR
  - Images are the primary data used for rendering
  - Motivation

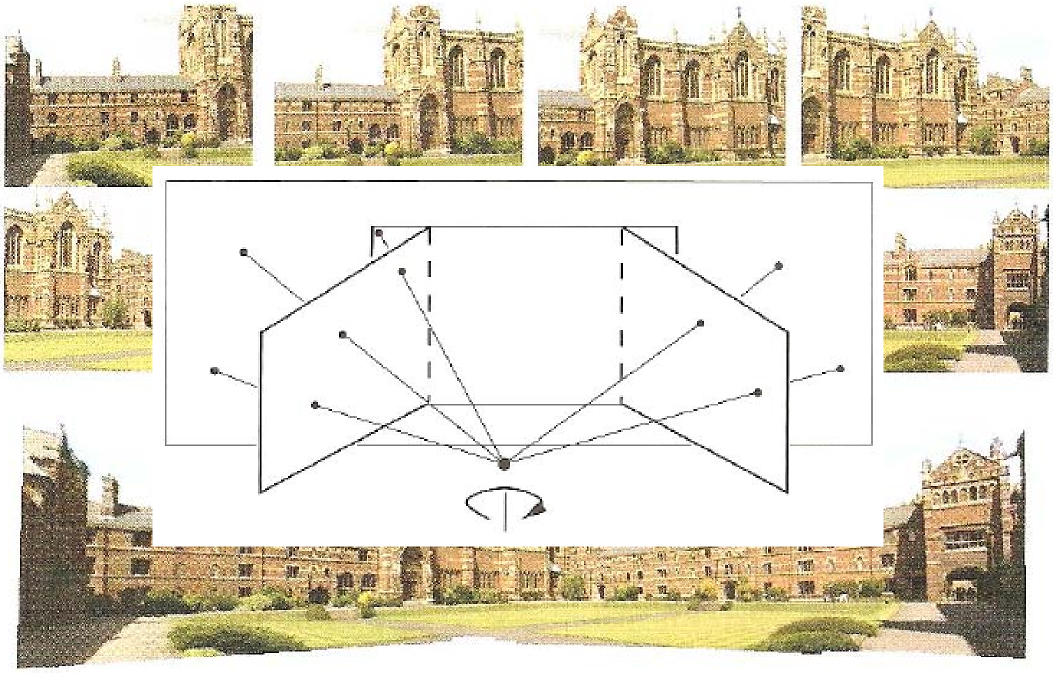
    …

  - *But on the other hand,*
    - Polygons have the advantage of representing the object in a reasonable fashion from any view.
      - As the camera moves, the representation of the object does not have to change (except perhaps LOD)

Image panorama

Cylindrical Panorama





Multiple View Geometry in Computer Vision (Hartley & Zisserman)

Lumigraph

- Lumigraph
  - Light-field rendering techniques
  - Instead of viewing much of an environment from a single location, a single object is viewed from a set of viewpoints.
  - Given a new viewpoint, these techniques perform an interpolation process between stored views in order to create the new view.
  - “A next generation photograph that allows you to see an image from any angle you can think of.”
    - This is a digital equivalent to a hologram.
  - ‘reverse side’ of Global Illumination in the rendering spectrum

Lumigraph / Light Field

- Levoy & Hanrahan (Siggraph '96)
  - must stay outside convex hull of the object
  - like rebinning in computed tomography

Light field is an array of images

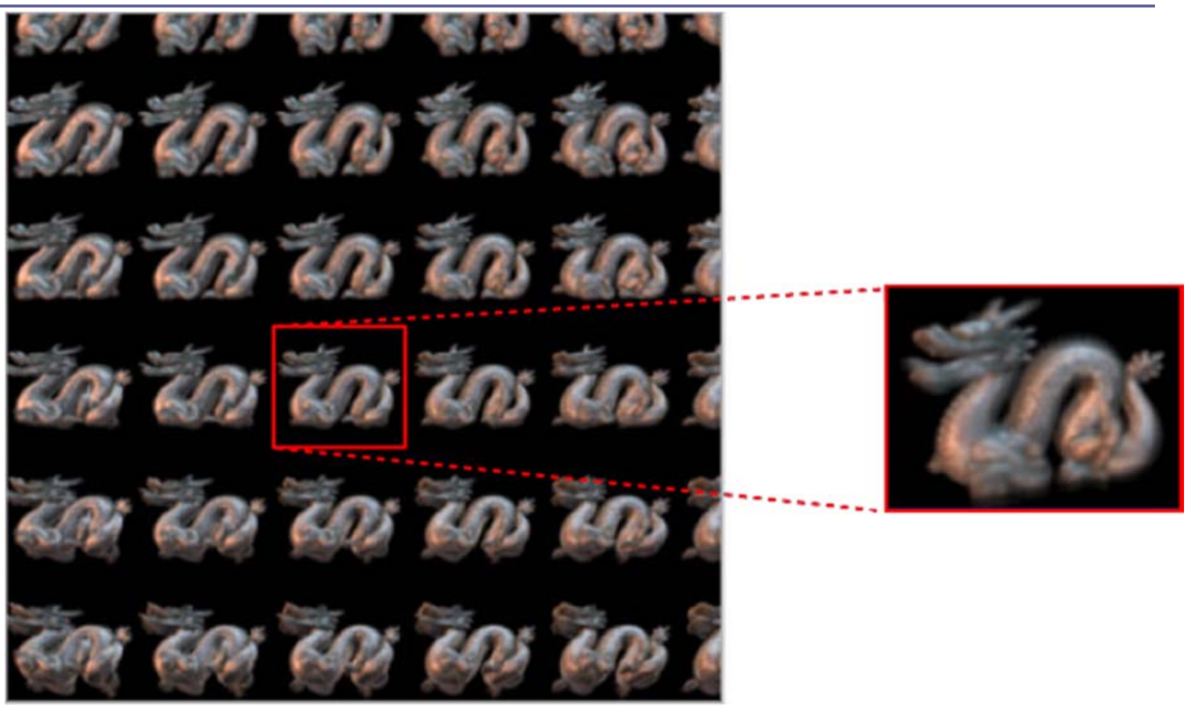


Image-based Rendering

- Image-based Rendering
- Image-based Modeling
- Image-based Lighting
  - 2000 UC Berkeley (DVD)

- “Video-based” Modeling and Rendering

- Video Tooning
  - http://www.youtube.com/watch?v=gGfL20agFeQ (4.32)

- Video-based Rendering
  - Unstructured VBR (2010)
    - http://www.youtube.com/watch?v=gqSLsPrA7RE (5:50)

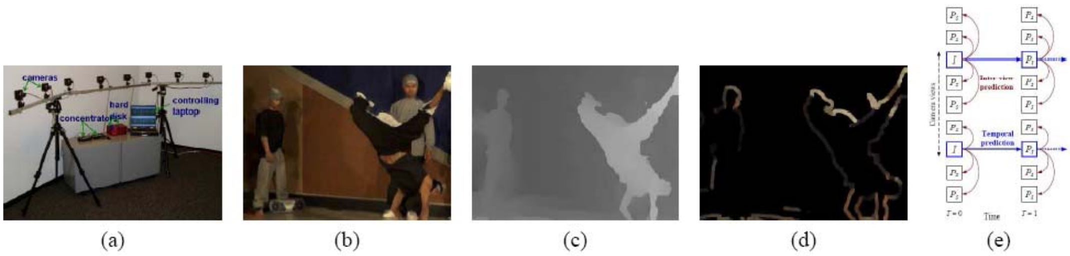


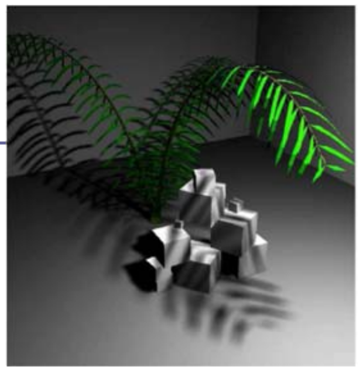
Figure 6: Video View Interpolation: (a) capture rig consisting of 8 cameras; (b) recovered background color image; (c) recovered depth map; (d) recovered boundary image (color channel); (e) multi-viewpoint video compression scheme.

Siggraph …

- Technical Papers Preview Trailer (3:11)
  - 2013: http://www.youtube.com/watch?v=JAFhkdGtHck
  - **2016:** https://youtu.be/dQBJ0r5Pj5s

- Real-time graphics
  - **Dawn** (siggraph 2003) ,, shader demo
  - 2012 A new Dawn (Tech demo)
    - https://www.youtube.com/watch?v=bI1_quVr_3w
  - Real time PC graphics (Demo by Nvidia 2000~2013) 2:13:06
    - https://youtu.be/2bHpUljLVrc
  - Skin Stretch (Siggraph 2015)
    - https://youtu.be/7RbOsYFioHg

21.3 Light Simulation

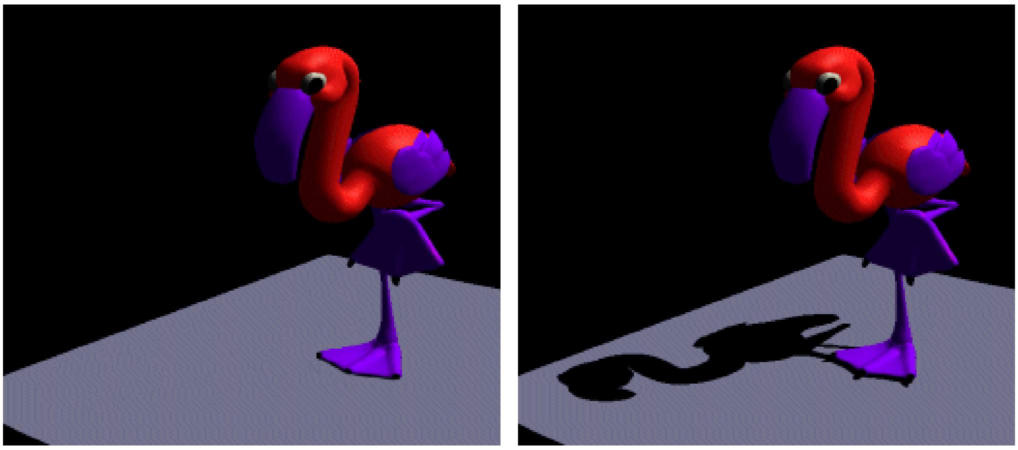


- 21.3.1. Direct point lights
- 21.3.2. Direct area lights
- 21.3.3. Two bounces
- 21.3.4. And so on
  - .. In software rendering, such integrals can be computed using sampling and summing. Successful methods here include distribution ray tracing, path tracing, radiance algorithm and photon mapping.
  - *In OpenGL,* most of these effects are simply **hacked** using a combination of **multiple pass rendering** and **precomputed texture.**
- 21.3.5. The Rendering equation

$$L^t(\tilde{x},\vec{v}) \quad = \quad L^e(\tilde{x},\vec{v}) + \int_H dw\ f_{\tilde{x},\vec{n}}(\vec{w},\vec{v})\ L^t(\vec{w},\tilde{x})\cos(\theta)$$

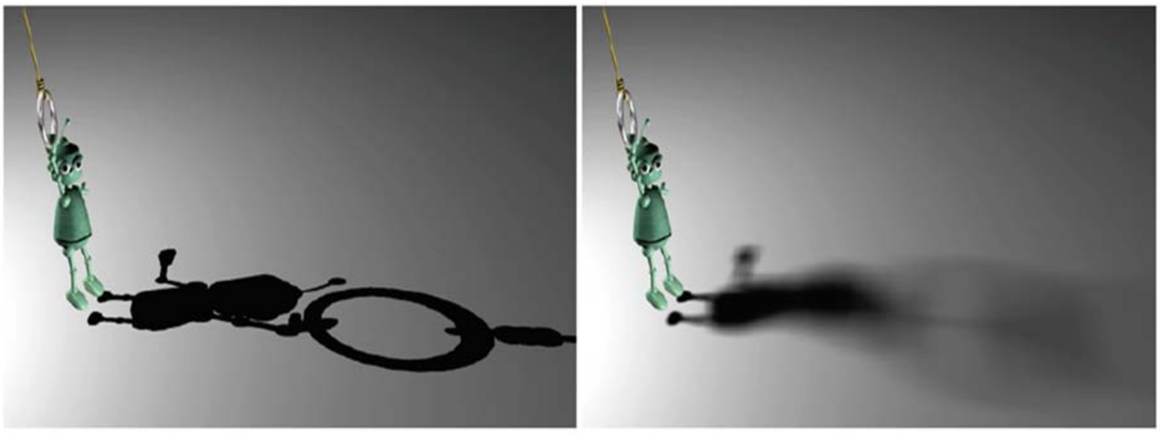
Shadows?

- Provide additional cues about the shapes and relative positions of objects in 3D world
- Makes 3D Graphics more believable



Han-Wei Shen (Ohio Univ)

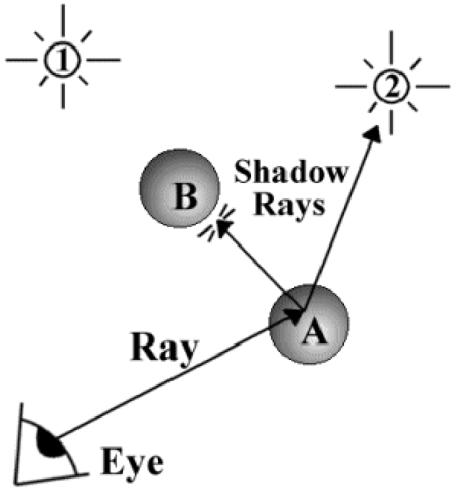
"Hard" and "Soft" Shadows



- Hard shadow
  – *point* light source
- Soft shadow
  – *area* light source

Simple Approach: Raytracing

- Cast ray to light (*shadow rays*)
- Surface point in shadow if the shadow rays hits an occluder object.
- Raytracing is slow, can we do better (perhaps at the cost of quality)



Shadow Algorithms

- We will focus on hard shadows
  - Planar Shadows
  - Shadow Maps
  - Shadow Volume