Image-based Lighting (Part 2)

Computational Photography
Derek Hoiem, University of Illinois

Many slides from Debevec, some from Efros
Today

• Brief review of last class

• Show how to get an HDR image from several LDR images, and how to display HDR

• Show how to insert fake objects into real scenes using environment maps
How to render an object inserted into an image?
How to render an object inserted into an image?

Traditional graphics way

• Manually model BRDFs of all room surfaces
• Manually model radiance of lights
• Do ray tracing to relight object, shadows, etc.
How to render an object inserted into an image?

Image-based lighting

• Capture incoming light with a “light probe”
• Model local scene
• Ray trace, but replace distant scene with info from light probe
Key ideas for Image-based Lighting

• Environment maps: tell what light is entering at each angle within some shell
Cubic Map Example
Spherical Map Example
Key ideas for Image-based Lighting

• Light probes: a way of capturing environment maps in real scenes
Mirrored Sphere
One small snag

- How do we deal with light sources? Sun, lights, etc?
  - They are much, much brighter than the rest of the environment

- Use High Dynamic Range photography!
Key ideas for Image-based Lighting

• Capturing HDR images: needed so that light probes capture full range of radiance
Problem: Dynamic Range
Long Exposure

Real world

10^{-6} \quad 10^{6}

High dynamic range

Picture

10^{-6} \quad 10^{6}

0 to 255

Real world vs. Picture: The scale in the real world is much wider, covering from 10^{-6} to 10^{6}, whereas in a picture, it is limited to 0 to 255.
Short Exposure

Real world

Picture

High dynamic range

10^{-6} to 10^6

0 to 255
Recovering High Dynamic Range Radiance Maps from Photographs

Paul Debevec
Jitendra Malik

Computer Science Division
University of California at Berkeley

August 1997
Ways to vary exposure

- Shutter Speed (*)
- F/stop (aperture, iris)
- Neutral Density (ND) Filters
Shutter Speed

Ranges: Canon D30: 30 to 1/4,000 sec.
        Sony VX2000: ¼ to 1/10,000 sec.

Pros:
• Directly varies the exposure
• Usually accurate and repeatable

Issues:
• Noise in long exposures
The Approach

• Get pixel values $Z_{ij}$ for image with shutter time $\Delta t_j$ ($i^{th}$ pixel location, $j^{th}$ image)

• Exposure is radiance integrated over time:
  $$E_{ij} = R_i \cdot \Delta t_j \implies \ln E_{ij} = \ln R_i + \ln \Delta t_j$$

• To recover radiance $R_i$, we must map pixel values to log exposure: $\ln(E_{ij}) = g(Z_{ij})$

• Solve for $R, g$ by minimizing:
  $$\sum_{i=1}^{N} \sum_{j=1}^{P} w(Z_{ij}) \left[ \ln R_i + \ln \Delta t_j - g(Z_{ij}) \right]^2 + \lambda \sum_{z=Z_{min}}^{Z_{max}} w(z) g''(z)^2$$
The objective

Solve for radiance $R$ and mapping $g$ for each of 256 pixel values to minimize:

$$\sum_{i=1}^{N} \sum_{j=1}^{P} w(Z_{ij}) \left[ \ln R_i + \ln \Delta t_j - g(Z_{ij}) \right]^2 + \lambda \sum_{z=Z_{\text{min}}}^{Z_{\text{max}}} w(z) g''(z)^2$$

- give pixels near 0 or 255 less weight
- known shutter time for image $j$
- exposure should smoothly increase as pixel intensity increases
- radiance at particular pixel site is the same for each image
- exposure, as a function of pixel value
Matlab Code

function [g, IB] = gsolve(Z, B, l, w)

n = 256;
A = zeros(size(Z, 1) + size(Z, 2) + n + 1, n + size(Z, 1));
b = zeros(size(A, 1), 1);

% Include the data-fitting equations
k = 1;
for i = 1:size(Z, 1)
    for j = 1:size(Z, 2)
        wij = w(i, j);
        A(k, Z(i, j) + 1) = wij;
        A(k, n+i) = -wij;
        b(k, 1) = wij * B(i, j);
        k = k + 1;
    end
end

% Fix the curve by setting its middle value to 0
A(k, 129) = 1;
k = k + 1;

% Include the smoothness equations
for i = 1:n-2
    A(k, i) = 1 + w(i+1);
    A(k, i+1) = -2 + w(i+1);
    A(k, i+2) = 1 + w(i+1);
    k = k + 1;
end

% Solve the system using SVD
x = A \\ b;

g = x(1:n);
IB = x(n+1:size(x, 1));
function [g,lE]=gsolve(Z,B,l,w)

n = 256;
A = zeros(size(Z,1)*size(Z,2)+n+1,n+size(Z,1));
b = zeros(size(A,1),1);

k = 1;                %% Include the data-fitting equations
for i=1:size(Z,1)
    for j=1:size(Z,2)
        wij = w(Z(i,j)+1);
        A(k,Z(i,j)+1) = wij; A(k,n+i) = -wij; b(k,1) = wij * B(i,j);
        k=k+1;
    end
end

A(k,129) = 1;         %% Fix the curve by setting its middle value to 0

for i=1:n-2           %% Include the smoothness equations
    A(k,i)=l*w(i+1); A(k,i+1)=-2*l*w(i+1); A(k,i+2)=l*w(i+1);
    k=k+1;
end

x = A\b;              %% Solve the system using pseudoinverse

g = x(1:n);
lE = x(n+1:size(x,1));
Illustration

Image series

Pixel Value $Z = f(\text{Exposure})$

Exposure = Radiance $\cdot \Delta t$

$log \text{ Exposure} = log \text{ Radiance} + log \Delta t$
Response Curve

Assuming unit radiance for each pixel

After adjusting radiances to obtain a smooth response curve
Results: Digital Camera

Kodak DCS460
1/30 to 30 sec

Recovered response curve

Pixel value

log Exposure
Reconstructed radiance map
Results: Color Film

- Kodak Gold ASA 100, PhotoCD
Recovered Response Curves

Red

Green

Blue

RGB
How to display HDR?

Linearly scaled to display device
Global Operator (Reinhart et al)

\[ L_{\text{display}} = \frac{L_{\text{world}}}{1 + L_{\text{world}}} \]
Global Operator Results
Reinhart Operator

Darkest 0.1% scaled to display device
Acquiring the Light Probe
Assembling the Light Probe
Real-World HDR Lighting Environments

Lighting Environments from the Light Probe Image Gallery:
http://www.debevec.org/Probes/
Illumination Results
Comparison: Radiance map versus single image

HDR

LDR
CG Objects Illuminated by a Traditional CG Light Source
Illuminating Objects using Measurements of Real Light

Environment assigned “glow” material property in Greg Ward’s RADIANCE system.

http://radsite.lbl.gov/radiance/
Rendering with Natural Light

SIGGRAPH 98 Electronic Theater
Movie

- [http://www.youtube.com/watch?v=EHBgkeXH9lU](http://www.youtube.com/watch?v=EHBgkeXH9lU)
Illuminating a Small Scene
We can now illuminate **synthetic objects** with **real light**.

- Environment map
- Light probe
- HDR
- Ray tracing

How do we add synthetic objects to a **real scene**?
Real Scene Example

*Goal: place synthetic objects on table*
Light Probe / Calibration Grid
Modeling the Scene

light-based model

real scene
The Light-Based Room Model
Modeling the Scene

light-based model

synthetic objects

local scene

real scene
The Lighting Computation

- distant scene (light-based, unknown BRDF)
- synthetic objects (known BRDF)
- local scene (estimated BRDF)
Rendering into the Scene

Background Plate
Rendering into the Scene

Objects and Local Scene matched to Scene
Differential Rendering
Difference in local scene
IMAGE-BASED LIGHTING IN Fiat Lux

Paul Debevec, Tim Hawkins, Westley Sarokin, H. P. Duiker, Christine Cheng, Tal Garfinkel, Jenny Huang

SIGGRAPH 99 Electronic Theater
Fiat Lux

• http://ict.debevec.org/~debevec/FiatLux/movie/
• http://ict.debevec.org/~debevec/FiatLux/technology/
HDR Image Series

2 sec

1/4 sec

1/30 sec

1/250 sec

1/2000 sec

1/8000 sec
Stp1 Panorama
Assembled Panorama
Light Probe Images
Capturing a Spatially-Varying Lighting Environment
What if we don’t have a light probe?

Zoom in on eye

Insert Relit Face

Environment map


-- Nishino Nayar 2004
Environment Map from an Eye
Can Tell What You are Looking At

Eye Image:

Computed Retinal Image:
Video
Summary

• Real scenes have complex geometries and materials that are difficult to model

• We can use an environment map, captured with a light probe, as a replacement for distance lighting

• We can get an HDR image by combining bracketed shots

• We can relight objects at that position using the environment map
Have a good break!

• Next class – computational cameras