High Dynamic Range Images

...with a lot of slides stolen from Paul Debevec and Yuanzhen Li,

© Alyosha Efros

15-463: Computational Photography
Alexei Efros, CMU, Fall 2007
The Grandma Problem
Problem: Dynamic Range

The real world is high dynamic range.
pixel (312, 284) = 42

42 photos?
Long Exposure

Real world

Picture

0 to 255

High dynamic range

$10^{-6}$ to $10^6$
Short Exposure

Real world

High dynamic range

Picture

0 to 255

10^{-6} to 10^{6}
Camera Calibration

- **Geometric**
  - How pixel *coordinates* relate to *directions* in the world

- **Photometric**
  - How pixel *values* relate to *radiance* amounts in the world
The Image Acquisition Pipeline

- **Lens**: scene radiance ($W/sr/m^2$)
- **Shutter**: sensor irradiance
- **Film**: sensor exposure ($\Delta t$)
- **Electronic Camera**: latent image

The Image Acquisition Pipeline
\[
\log \text{Exposure} = \log (\text{Radiance} \times \Delta t)
\]

(CCD photon count)
Varying Exposure
Camera is not a photometer!

- **Limited dynamic range**
  ⇒ Perhaps use multiple exposures?
- **Unknown, nonlinear response**
  ⇒ Not possible to convert pixel values to radiance
- **Solution:**
  - Recover response curve from multiple exposures, then reconstruct the *radiance map*
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
Shutter Speed

- **Note**: shutter times usually obey a power series – each “stop” is a factor of 2

- $\frac{1}{4}, \frac{1}{8}, \frac{1}{15}, \frac{1}{30}, \frac{1}{60}, \frac{1}{125}, \frac{1}{250}, \frac{1}{500}, \frac{1}{1000}$ sec

- Usually really is:

- $\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \frac{1}{64}, \frac{1}{128}, \frac{1}{256}, \frac{1}{512}, \frac{1}{1024}$ sec
The Algorithm

Image series

\[ \Delta t = \frac{1}{64} \text{ sec} \]

\[ \Delta t = \frac{1}{16} \text{ sec} \]

\[ \Delta t = \frac{1}{4} \text{ sec} \]

\[ \Delta t = 1 \text{ sec} \]

\[ \Delta t = 4 \text{ sec} \]

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

Exposure = Radiance \( \times \) \( \Delta t \)

\( \log \) Exposure = \( \log \) Radiance + \( \log \) \( \Delta t \)
Response Curve

Assuming unit radiance for each pixel

After adjusting radiances to obtain a smooth response
The Math

• Let \( g(z) \) be the *discrete* inverse response function.

• For each pixel site \( i \) in each image \( j \), want:

\[
\ln \text{Radiance}_i + \ln \Delta t_j = g(Z_{ij})
\]

• Solve the overdetermined linear system:

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

fitting term

smoothness term
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
k=k+1;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 SVD

g = x(1:n);
lE = x(n+1:size(x,1));
Results: Digital Camera

Kodak DCS460
1/30 to 30 sec

Recovered response curve

Pixel value

log Exposure
Results: Color Film

- Kodak Gold ASA 100, PhotoCD
Recovered Response Curves

Red

Green

Blue

RGB
The Radiance Map
The Radiance Map

Linearly scaled to display device
Portable FloatMap (.pfm)

- 12 bytes per pixel, 4 for each channel

![Color bands representing pixel values]

- Sign, exponent, mantissa

Text header similar to Jeff Poskanzer’s .ppm image format:

```
PF
768 512
1
<binary image data>
```

Floating Point TIFF similar
Radiance Format
(.pic, .hdr)

32 bits / pixel

Red Green Blue Exponent

(145, 215, 87, 149) =
(145, 215, 87) * 2^(149-128) =
(1190000, 1760000, 713000)

(145, 215, 87, 103) =
(145, 215, 87) * 2^(103-128) =
(0.00000432, 0.00000641, 0.00000259)

**ILM’s OpenEXR (.exr)**

- 6 bytes per pixel, 2 for each channel, compressed
- Several lossless compression options, 2:1 typical
- Compatible with the “half” datatype in NVidia's Cg
- Supported natively on GeForce FX and Quadro FX
- Available at [http://www.openexr.net/](http://www.openexr.net/)
Now
What?
Tone Mapping

• How can we do this?
  Linear scaling?, thresholding? Suggestions?

Real World
Ray Traced
World (Radiance)

Display/
Printer

High dynamic range

0 to 255
Simple Global Operator

• Compression curve needs to
  – Bring everything within range
  – Leave dark areas alone

• In other words
  – Asymptote at 255
  – Derivative of 1 at 0
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
What do we see?

Vs.
What does the eye sees?

The eye has a huge dynamic range. Do we see a true radiance map?

Figure 1: The range of luminances in the natural environment and associated visual parameters. After Hood (1986).
Metamores

Can we use this for range compression?
Compressing Dynamic Range
Compressing and Companding High Dynamic Range Images with Subband Architectures

Yuanzhen Li, Lavanya Sharan, Edward Adelson
Massachusetts Institute of Technology
Dynamic Range Problem

Source: Shree Nayar
Range Compression

Method: Gamma or log on intensities.
Problem: loss of detail.

Solution: filtering.
Problem: halos.

Halos!!
Multiscale Subband Decomposition

Choice of filters: Wavelets, QMFs, Laplacian, etc. They all worked.
Point Nonlinearity on Subbands

Original subband → point nonlinearity limits range → Modified subband

Problem: Nonlinear distortion.
Smooth Gain Control

gain(x) = \frac{b'(x)}{b(x)} =\  =

\text{smooth}
Smooth Gain Control Reduces Distortion

Point nonlinearity

Smooth gain control

Distorted.

Distortion reduced.
Smooth Gain Control on Subbands

rectify -> blur
activity map -> gain map