High-Dynamic-Range Imaging & Tone Mapping

Today’s Agenda

• Spatial color vision
• JPEG
• The dynamic range challenge
• Multiple exposures
• Estimating the response curve
• HDR merging: linear case
• Tone mapping

Contrast Sensitivity Function
Campbell-Robson Chart
Contrast Sensitivity Function (CSF)

- The sensitivity of the luminance channel is much higher than the sensitivities of Blue-Yellow and Red-Green opponent channels.
- The shape of Luminance CSF is different than those of the color channels.
- Chromatic channels have maximal sensitivity for uniform fields.

Photoshop demo

- Image > Mode > Lab color
- Go to channel panel, select Lightness
- Filter > Blur > Gaussian Blur, e.g. 4 pixel radius
  - very noticeable
- Undo, then select a & b channels
- Filter > Blur > Gaussian Blur, same radius
  - hardly visible effect

Today’s Agenda

- Spatial color vision
- JPEG
- The dynamic range challenge
- Multiple exposures
- Estimating the response curve
- HDR merging: linear case
- Tone mapping

Opponents and image compression

- JPEG, MPG, television
- Color opponents instead of RGB
  - YCrCB, similar to YUV
- Compress color more than luminance
  - downsample by factor of two for jpeg
  - less bandwidth for TV
- Exploit contrast sensitivity function
  - Compress high frequencies more
JPEG Compression

- Convert to YCbCr
  - half the resolution for Cr & Cb
- Perform Discrete Cosine Transform to work in frequency space
  - Local DCT, 8x8 pixel blocks
- Use CSF for quantization
  - more bits for frequencies with more sensitivity (medium)
- Other usual coding tricks
  - entropy coding, smart order of blocks

Example

- 800 x 533 image

Today’s Agenda

- Spatial color vision
- JPEG
- The dynamic range challenge
- Multiple exposures
- Estimating the response curve
- HDR merging: linear case
- Tone mapping

Light, exposure and dynamic range

- Inside is too dark
- Outside is too bright
- Sun overexposed
- Foreground too dark
Dynamic Range

- The "dynamic range" of a scene is the contrast ratio between its brightest and darkest parts.

- Typical images displayed on screen are 24-bits:
  - 8-bits per color component (RGB)
  - 256 different intensity levels

- Real-world dynamic range is far greater than 256 intensity levels!

Real world dynamic range

- Eye can adapt from $\sim 10^{-6}$ to $10^6$ cd/m$^2$.
- Often 1:100,000 in a scene.

The world is high dynamic range

- Relative brightness of different scenes, ranging from 1 inside a dark room lit by a monitor to 2,000,000 looking at the sun. (Photos courtesy of Paul Debevec)

Picture dynamic range: Guess!

- Pure black to pure white.
- Real world range: $10^{-6}$ to $10^6$.
- Picture range: $10^{-6}$ to $10^6$. 
The dynamic range problem

- **media (approximate and debatable)**
  - 10:1 photographic print (higher for glossy paper)
  - 20:1 artist’s paints
  - 200:1 slide film
  - 500:1 negative film
  - 1000:1 LCD display
  - 2000:1 digital SLR (~11 bits)

- **challenges**
  - choosing which 6-12 bits of the world to include in your photograph (cell phone to professional SLR, respectively)
  - metering the world to help you make this decision, since the world has more dynamic range than any light meter
  - compressing 12 bits into 4 bits for print, or 10 for LCD - this is the tone mapping problem

---

Picture dynamic range

- Typically 1:20 or 1:50
  - Black is ~50x darker than white

- Max 1:500

---

Real world

\[10^{-6} \text{ to } 10^6\]

Low contrast

---

Picture

\[10^{-6} \text{ to } 10^6\]

Low contrast
Problem 1: record the information

- The range of illumination levels that we encounter is 10 to 12 orders of magnitudes.
- Negatives/sensors can record 2 to 3 orders of magnitude.

\[10^{-6} \quad 10^0 \quad 10^3 \quad 10^6\]

\[\text{Real scenes} \quad \text{Negative/sensor}\]

Problem 2: Display the information

- Match limited contrast of the medium.
- Preserve details.

\[10^{-6} \quad 10^0 \quad 10^3 \quad 10^6\]

\[\text{Real world} \quad \text{High dynamic range} \quad \text{Low contrast}\]

Without HDR & tone mapping

With HDR & tone mapping
• http://www.stuckincustoms.com/2011/04/30/hdr-before-after/

Can be extreme

Not always cheesy

Not always cheesy

Today’s Agenda

• Multiple-exposure High-Dynamic-Range imaging
• Tone mapping using the bilateral filter

Today’s Agenda

• Spatial color vision
• JPEG
• The dynamic range challenge
• Multiple exposures
• Estimating the response curve
• HDR merging: linear case
• Tone mapping

Multiple exposure photography

• Sequentially measure all segments of the range

Real world

10^-6
High dynamic range
10^6

Picture

10^-6
Low contrast
10^6
Multiple exposure photography

• Sequentially measure all segments of the range

Real world

$10^{-6}$

High dynamic range

$10^6$

Picture

$10^{-6}$

Low contrast

$10^6$
Multiple exposure photography

- Sequentially measure all segments of the range

Real world

\[10^{-6} \quad \text{High dynamic range} \quad 10^6\]

Picture

\[10^{-6} \quad \text{Low contrast} \quad 10^6\]

How do we vary exposure?

- Options:
  - Shutter speed
  - Aperture
  - ISO
  - Neutral density filter

Tradeoffs

- **Shutter speed**
  - Range: ~30 sec to 1/4000sec (6 orders of magnitude)
  - Pros: reliable, linear
  - Cons: sometimes noise for long exposure

- **Aperture**
  - Range: ~f/1.4 to f/22 (2.5 orders of magnitude)
  - Cons: changes depth of field
  - Useful when desperate

- **ISO**
  - Range: ~100 to 1600 (1.5 orders of magnitude)
  - Cons: noise
  - Useful when desperate

- **Neutral density filter**
  - Range: up to 4 densities (4 orders of magnitude) & can be stacked
  - Cons: not perfectly neutral (color shift), not very precise, need to touch camera (shake)
  - Pros: works with strobe/flash, good complement when desperate

Today’s Agenda

- Spatial color vision
- JPEG
- The dynamic range challenge
- Multiple exposures
- **Estimating the response curve**
- HDR merging: linear case
- Tone mapping
Image formation

- Scene radiance $L(x, y)$ reaches the sensor at a pixel $x$, $y$
- For each image $i$, radiance gets multiplied by exposure factor $k_i$
  - depends on shutter speed, ISO, etc
- Noise $n$ gets added
- Values above 1 get clipped
  - depends on photosite well capacity

$$I(x, y) = \text{clip}(k_i L(x, y) + n)$$

Dynamic range

- In the highlights, we are limited by clipping
- In the shadows, we are limited by noise

Merging HDR

- For each pixel
  - figure out which images are useful
  - scale values appropriately according, ideally, to $k_i$
  - Voila!

$$I(x, y) = \text{clip}(k_i L(x, y) + n)$$

Which images are useful

- Eliminate clipped pixels
  - e.g. $>0.99$
- Eliminate pixels that are too dark / too noisy
  - e.g. $<0.002$
Assembling HDR

- Figure out scale factor between images
  - From exposure data
  - Or by looking at ratios $I_i(x,y)/I_j(x,y)$
    - but only when both are good
- Compute weight map $w_i$ for each image
- Reconstruct full image using weighted combination

$$\text{out}(x,y) = \frac{1}{\sum w_i(x,y)} \sum w_i(x,y) \frac{1}{k_i} I_i(x,y)$$

Computing $k_i$

$$l(x,y) = \text{clip}(k_i \cdot L(x,y) + n)$$

- Only up to global scale factor
- Actually compute $k_i/k_j$ for pairs of images
- Focus on pixels where
  - no clipping occurs & noise is negligible
- $I_i(x, y) = k_i \cdot L(x, y)$
- get $k_i/k_j$ by considering $I_i/I_j$
- If linearity holds, should be the same for all pixels
- Use median for extra robustness

Special cases

- Some pixels might be underexposed or overexposed in all images
- Simple solution: don’t eliminate dark pixels in the brightest image or bright pixels in the darkest one.
HDR combination papers

- Steve Mann
  http://genesis.eecg.toronto.edu/wyckoff/index.html
- Paul Debevec
  http://www.debevec.org/Research/HDR/
- Mitsunaga, Nayar, Grossberg
- http://people.csail.mit.edu/hasinoff/hdrnoise/

Today’s Agenda

- Spatial color vision
- JPEG
- The dynamic range challenge
- Multiple exposures
- Estimating the response curve
- HDR merging: linear case
- **Tone mapping**

Problem 2: Display the information

- Match limited contrast of the medium
- Preserve details

Real world: $10^{-6}$ High dynamic range: $10^{6}$

Picture: $10^{-6}$ Low contrast: $10^{6}$
The second half: contrast reduction

- **Input:** high-dynamic-range image
  - (floating point per pixel)

Naïve technique

- Scene has 1:10,000 contrast, display has 1:100
- Simplest contrast reduction?

Naïve: Gamma compression

- $X \rightarrow X^\gamma$ (where $\gamma = 0.5$ in our case)
- But... colors are washed-out. Why?

Gamma compression on intensity

- Colors are OK, but details (intensity high-frequency) are muddy
Oppenheim 1968, Chiu et al. 1993

- Reduce contrast of low-frequencies
- Keep mid and high frequencies

Homomorphic filtering

- Oppenheim, in the sixties
- Images are the product of illumination and albedo
  - Similarly, many sounds are the product of an envelope and a modulation
  - Illumination is usually slow-varying
  - Perform albedo-illumination using low-pass filtering of the log image

- See also Koenderink "Image processing done right"
  - http://www.springerlink.com/content/s1533/journal/1538.3333/linkingpublicationresults.1:106633.1
The halo nightmare

- For strong edges
- Because they contain high frequency

Low-freq.  
Reduce low frequency

High-freq.  

Instead use bilateral filter

- Do not blur across edges
- Non-linear filtering

Start with Gaussian filtering

- Here, input is a step function + noise

\[ J = f \otimes I \]

Bilateral filter

- Tomasi and Manduci 1998]  
  - http://www.cse.ucsc.edu/~manduchi/Papers/ICCV98.pdf
- Developed for denoising
- Related to
  - SUSAN filter [Smith and Brady 95]  
    http://citeseer.ist.psu.edu/smith95susan.html
  - Digital-TV [Chan, Osher and Chen 2001]  
    http://citeseer.ist.psu.edu/chan01digital.html
  - sigma filter  
- Full survey:  
Gaussian filter as weighted average

• Weight of $\xi$ depends on distance to $x$

$$J(x) = \sum_{\xi} f(x,\xi) \ I(\xi)$$

The problem of edges

• Here, $I(\xi)$ “pollutes” our estimate $J(x)$
• It is too different

$$J(x) = \sum_{\xi} f(x,\xi) \ I(\xi)$$

Principle of Bilateral filtering

[Tomasi and Manduchi 1998]

• Penalty $g$ on the intensity difference

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \ g(I(\xi) - I(x)) \ I(\xi)$$

Bilateral filtering

[Tomasi and Manduchi 1998]

• Spatial Gaussian $f$

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) g(I(\xi) - I(x)) \ I(\xi)$$
Bilateral filtering

- Spatial Gaussian $f$
- Gaussian $g$ on the intensity difference

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \cdot g(I(\xi) - I(x)) \cdot I(\xi)$$

Normalization factor

- $k(x) = \sum_{\xi} f(x,\xi) \cdot g(I(\xi) - I(x))$

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \cdot g(I(\xi) - I(x)) \cdot I(\xi)$$

Bilateral filtering is non-linear

- The weights are different for each output pixel

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \cdot g(I(\xi) - I(x)) \cdot I(\xi)$$

Bilateral denoising

Bilateral filter

Median 3x3
### Basic denoising

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Image 1</th>
<th>Image 2</th>
<th>Image 3</th>
<th>Image 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bilateral filter</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>Median 5x5</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
</tbody>
</table>

### Basic denoising

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Image 1</th>
<th>Image 2</th>
<th>Image 3</th>
<th>Image 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bilateral filter</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>Bilateral filter – lower sigma</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
</tbody>
</table>

### Contrast reduction

- **Input HDR image**: ![Image](image5.png)
- **Contrast too high!**
Contrast reduction

Input HDR image

Intensity

intensity = 0.4R + 0.7G + 0.01B

Color

R' = R / intensity
G' = G / intensity
B' = B / intensity

important to use ratios
(makes it luminance invariant)

Detail = log intensity - large scale
(residual)

Large scale

Bilateral
Filter in log

Spatial sigma: 2 to 5% image size
Range sigma: 0.4 (in log 10)

Reduce contrast
Contrast reduction

Log domain

- Very important to work in the log domain
- Recall: humans are sensitive to multiplicative contrast
- With log domain, our notion of “strong edge” always corresponds to the same contrast

Contrast reduction in log domain

- Set target large-scale contrast (e.g. $\log_{10} 100$)
  - i.e. in linear output, we want 1:100 contrast for large scale
- Compute range of input large scale layer:
  - $\text{largeRange} = \max(\text{inLogLarge}) - \min(\text{inLogLarge})$
- Scale factor $k = \log_{10} (100) / \text{largeRange}$
- Normalize so that the biggest value is 0 in log
- Optional: amplify detail by detailAmp

$$\text{outLog} = \text{detailAmp} \times \text{inLogDetail} + k(\text{inLogLarge} - \max(\text{inLogLarge}))$$
Final output

- From last slide:
  \[ \text{outLog} = \text{detailAmp} \times \text{inLogDetail} + \]
  \[ k(\text{inLogLarge} - \max(\text{inLogLarge})) \]
  \[ \text{outIntensity} = 10^{\text{outLog}} \]
- Recall that R’, G’, B’ is the intensity-normalized RGB color
  \[ \text{outR} = \text{outIntensity} \times R’ \]
  \[ \text{outG} = \text{outIntensity} \times G’ \]
  \[ \text{outB} = \text{outIntensity} \times B’ \]
Bells and whistles: increase detail

Input HDR image

Intensity

Color

Fast Bilateral Filter
IN LOG

Large scale

Reduce contrast

Large scale

Amplify

Input log - large scale

Output

What matters

- Spatial sigma: not very important
- Range sigma: quite important
- Use of the log domain for range: critical
  - Because HDR and because perception sensitive to multiplicative contrast
  - CIELab might be better for other applications
- Luminance computation
  - Not critical, but has influence
  - See Flash/no-flash paper [Eisemann 2004] for smarter function

Speed

- Direct bilateral filtering is slow (minutes)
- Fast algorithm: bilateral grid
  - http://groups.csail.mit.edu/graphics/bilagrid/
  - http://graphics.stanford.edu/papers/gkdtrees/
What have we learnt?

- Log is good
- Luminance is different from chrominance
- Separate components:
  - Low and high frequencies
- Strong edges are important

Alternative: exposure fusion

- One single step for both multiple-exposure merging & tone mapping

Figure 2. Exposure fusion is guided by weight maps for each input image. A high weight means that a pixel should appear in the final image. These weights reflect desired image qualities, such as high contrast and saturation. Image courtesy of Jacques Jofre.