CSE168: Rendering Algorithms
Tone Mapping

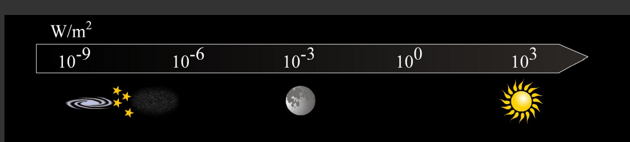
Henrik Wann Jensen

henrik@cs.ucsd.edu

Tone Mapping

- High Dynamic Range
- Tone mapping operators
- Bloom
- Night rendering

High Dynamic Range



Human Eye Technology

- Can see detail in scenes with 5 orders of magnitude in difference between darkest and brightest element

Display Technology

- Typically 8 bit (0-255 discrete intensity values)
- Printed media is even less

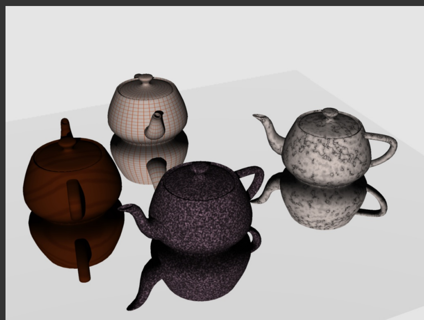
Gamma Correction

Not tone-mapping, but compensates for non-linear intensity mapping within the monitor.

$$L_d = L^{1/g}$$

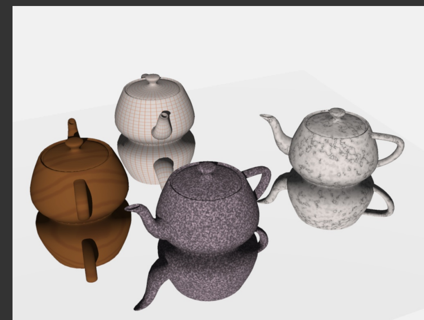
g is the gamma value (often 1.7-2.2).

Gamma Correction



No gamma

Gamma Correction



Gamma ($g = 2$)

Tone Mapping

The process of displaying a high dynamic range image on a low dynamic range display is called *tone mapping* or sometimes *tone reproduction*.

- Compressing scale of intensities
- Possibly modifying colors
- Possibly applying some spatial filter

Luminance

Luminance: "Radiance as seen by an observer"

$$L_v(x, \vec{\omega}) = \int_\lambda L(x, \vec{\omega}) V(\lambda) d\lambda$$

V is the spectral response curve for a standard observer.

RGB to Luminance

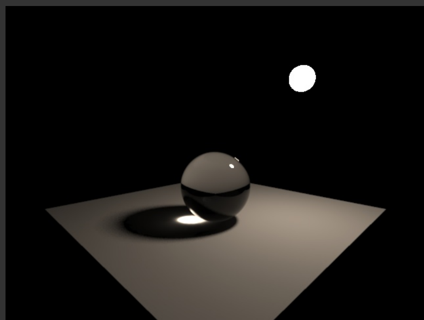
$$L_v = 683 * (L_r * 0.21 + L_g * 0.72 + L_b * 0.07)$$

Normalizing the Intensities

Computer L_{min} and L_{max}.

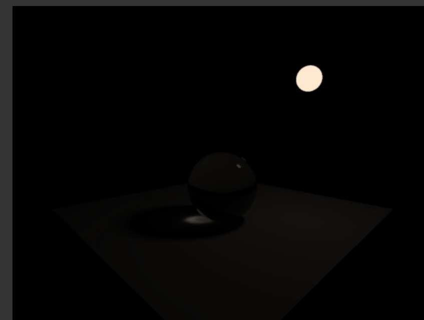
$$L_d(x, y) = L(x, y)/L_{max}$$

Normalizing the Intensities



Not normalized

Normalizing the Intensities



Normalized

Normalizing the Intensities

- Looses too much detail
- Only for scenes with low dynamic range

Contrast Based Scale Factor

- Greg Ward 1994
- Capture smallest *visible* change in luminance

Contrast Based Scale Factor

Blackwell found in 1970s that:

$$\Delta L(L_a) = 0.0594 * (1.219 + L_a^{0.4})^{2.5}$$

where L_a is the adaptation luminance.

Ward seeks a constant m such that:

$$L_d = m * L_v$$

Contrast Based Scale Factor

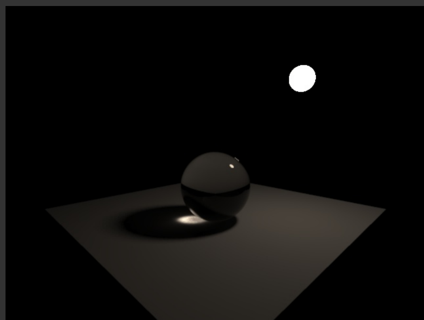
Assume that

$$\Delta L(L_{da}) = m \Delta L(L_w a)$$

$$m = \frac{1}{L_{d,max}} \left(\frac{1.219 + L_{da}^{0.4}}{1.219 + L_{wa}^{0.4}} \right)^{2.5}$$

Ward suggest $L_{da} = L_{dmax}/2$. World adaptation can be log average image luminance.

Contrast Based Scale Factor

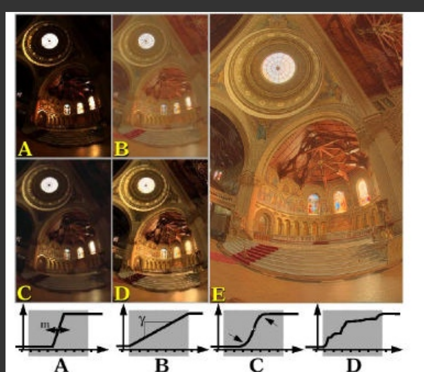


Contrast based scale factor

Contrast Based Scale Factor

- Global tone operator
- Can be made local by computing a local adaptation radiance

Other Intensity Mappings



Photographic Tonemapping

Reinhard et al. SIGGRAPH 2002

Photographic Tonemapping

Compute log average image luminance:

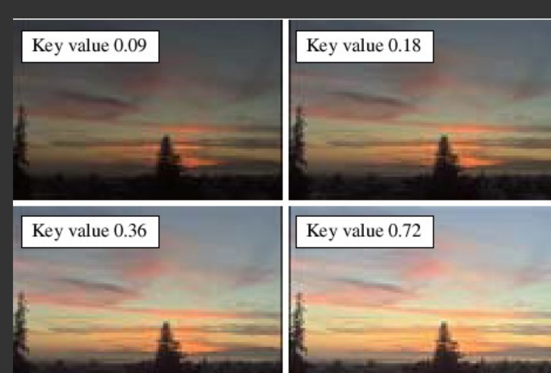
$$\tilde{L}_w = exp\left(\frac{1}{N}\sum_{pixels} log(\epsilon + L_w(x,y))\right)$$

Compute scaled luminance

$$L(x,y) = \frac{a}{\tilde{L}_w} L_w(x,y)$$

a is the key value.

Photographic Intensity Mapping



Photographic Tonemapping

Compress scaled luminance into 0-1

$$L_d(x,y) = \frac{L(x,y)}{1 + L(x,y)}$$

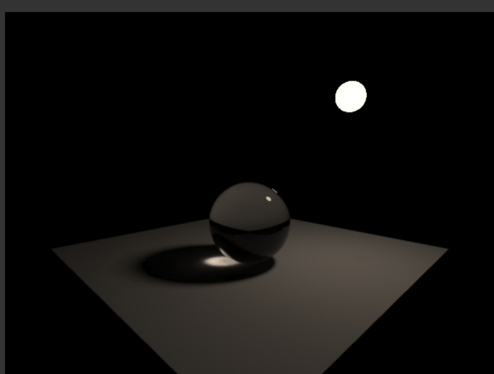
Compresses bright values more.

Photographic Tonemapping

Enable burn out areas

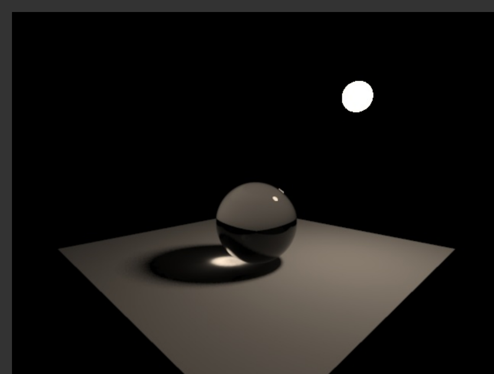
$$L_d(x,y) = \frac{L(x,y)}{1 + L(x,y)} \left(1 + \frac{L(x,y)}{L_{white}^2}\right)$$

Photographic Intensity Mapping



Key = 0.18

Photographic Intensity Mapping



Key = 0.5

More Tone Mapping

- Perceptual operators
- Gradient compression
- Histogram techniques

Bloom

Make lights look brighter...

Bloom

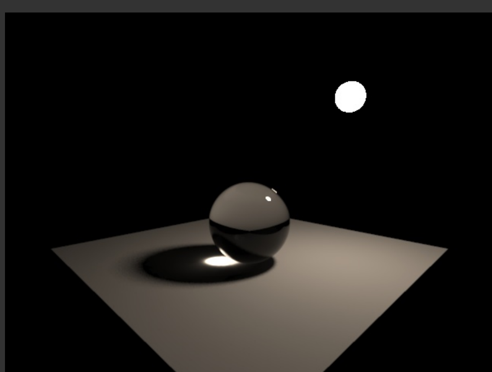
$$f(x,y) = \left(1 - \frac{\sqrt{x^2+y^2}}{r}\right)^4$$

where r is the filter radius.

Bloom Algorithm

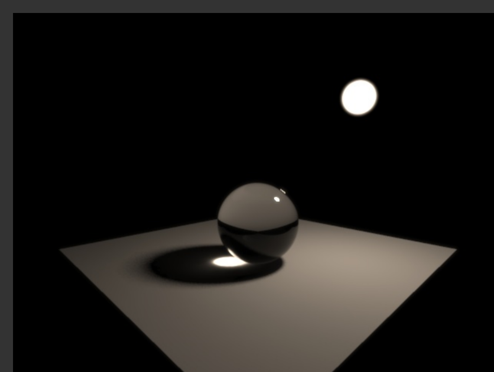
```
for each pixel (x,y) in the input image
  add f(x,y) to output image
  update weight of each pixel in output image

normalize output image
```


No Bloom



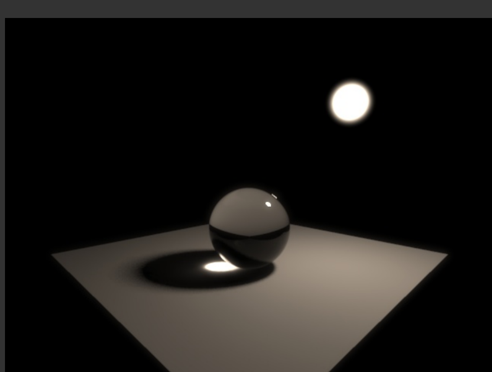
No bloom

Bloom



Bloom radius is 8 pixels

Bloom



Bloom radius is 32 pixels

Night Rendering

Next Time

- Non-Photorealistic rendering