

UNIVERSITY OF
CAMBRIDGE
COMPUTER LABORATORY



High dynamic range and tone mapping

Advanced Graphics and Image Processing

Rafał Mantiuk

Computer Laboratory, University of Cambridge

Cornell Box: need for tone-mapping in graphics



Rendering



Photograph

Real-world scenes are more challenging



- ▶ The match could not be achieved if the light source in the top of the box was visible
- ▶ The display could not reproduce the right level of brightness

Dynamic range



Luminance
↓
 $\frac{\max L}{\min L}$
↑
(for SNR > 3)

Dynamic range (contrast)

- ▶ As ratio:

$$C = \frac{L_{\max}}{L_{\min}}$$

- ▶ Usually written as C:1, for example 1000:1.

- ▶ As “orders of magnitude”
or log10 units:

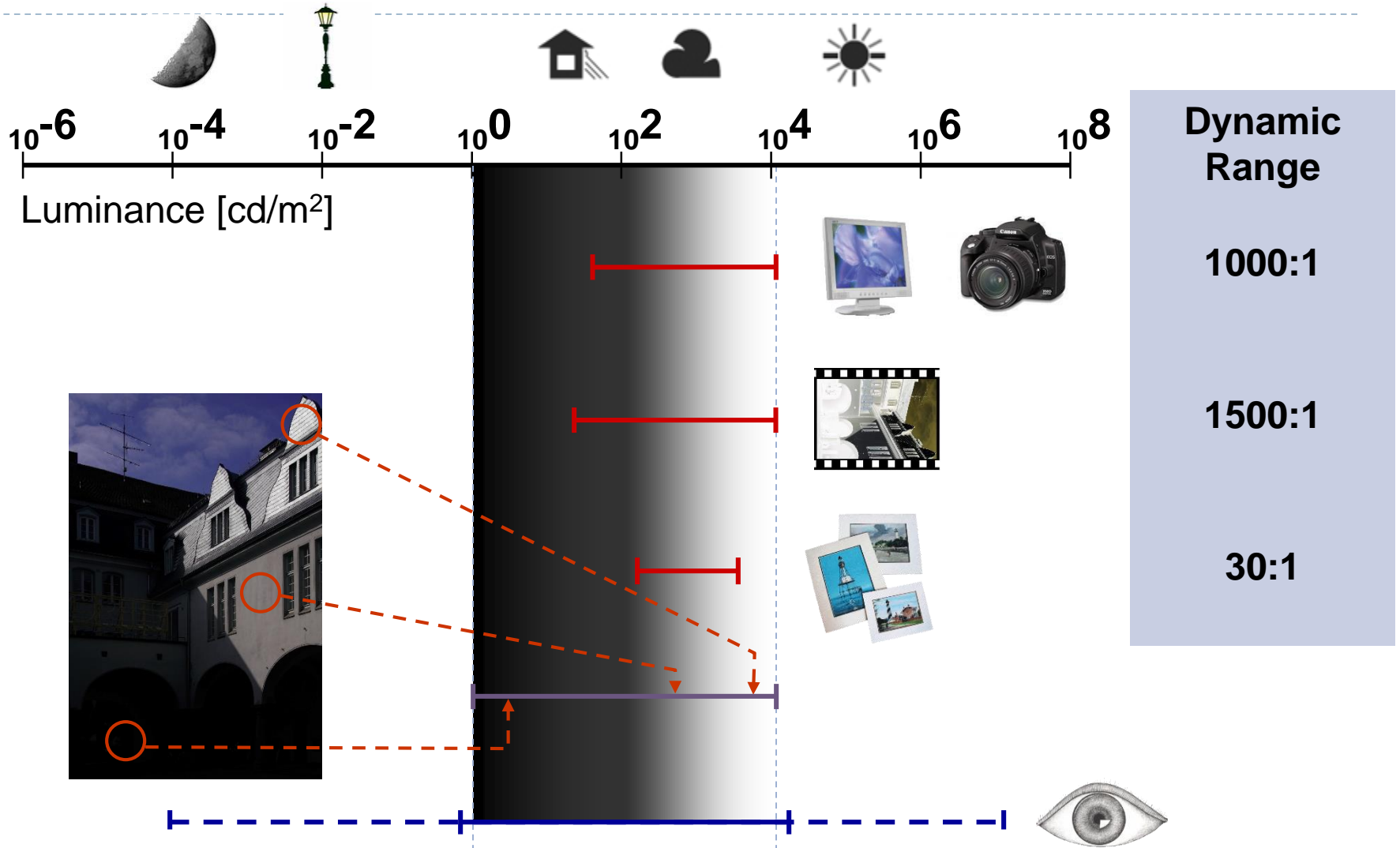
$$C_{10} = \log_{10} \frac{L_{\max}}{L_{\min}}$$

- ▶ As stops:

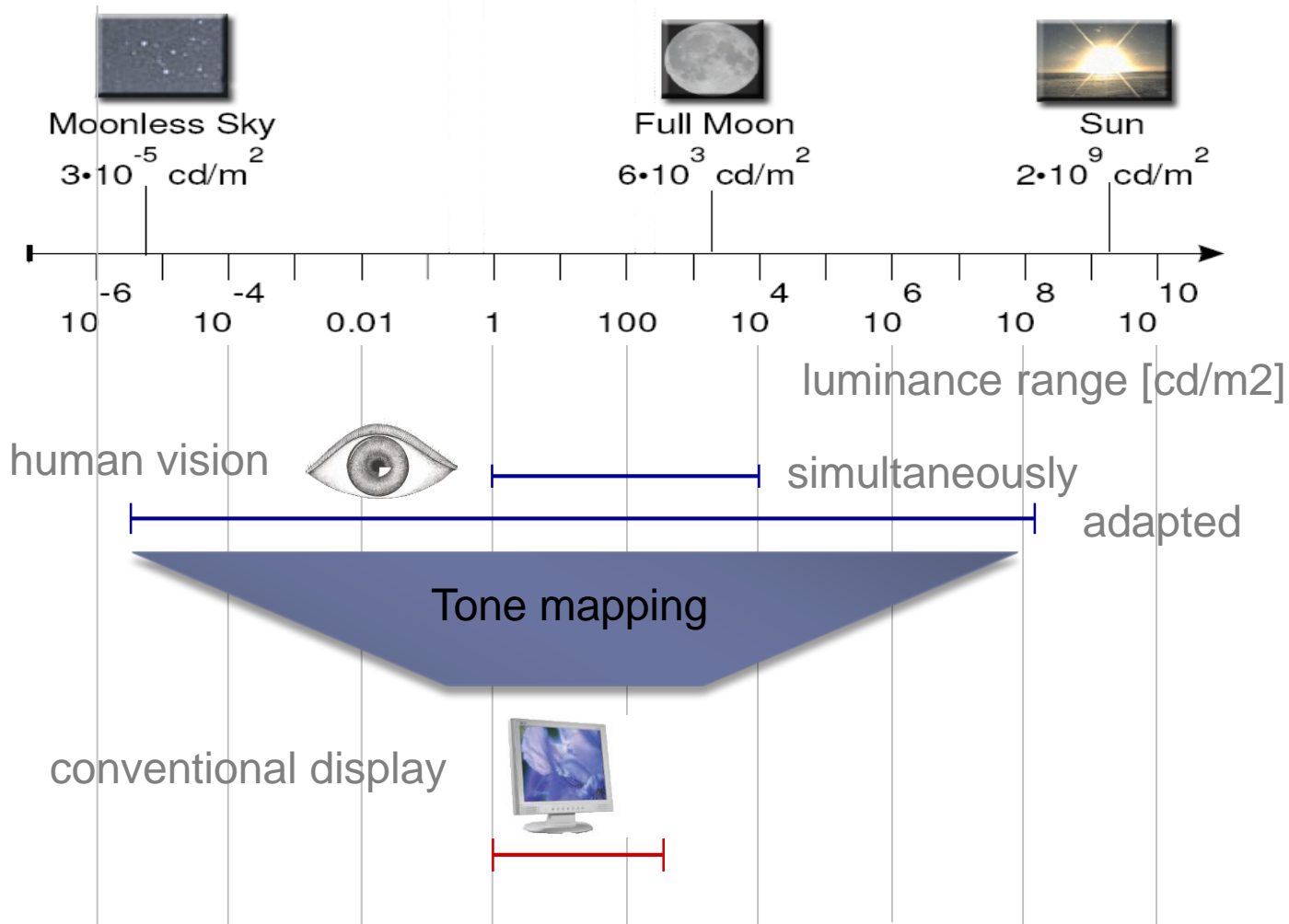
$$C_2 = \log_2 \frac{L_{\max}}{L_{\min}}$$

One stop is doubling
of halving the amount of light

High dynamic range (HDR)



Tone-mapping problem



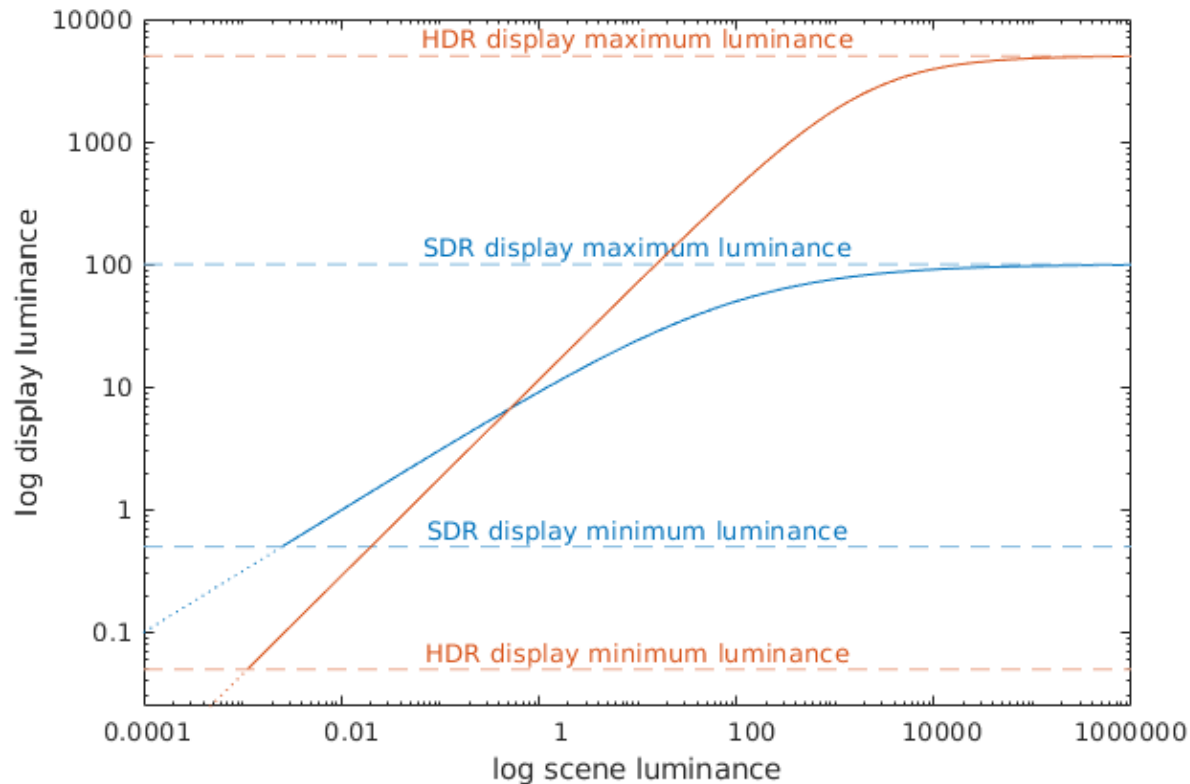
Why do we need tone mapping?

- ▶ **To reduce dynamic range**
- ▶ **To customize the look**
 - ▶ colour grading
- ▶ **To simulate human vision**
 - ▶ for example night vision
- ▶ To adapt displayed images to a **display and viewing conditions**
- ▶ To make rendered images look **more realistic**
- ▶ To map from **scene- to display-referred** colours
- ▶ Different tone mapping operators achieve different goals



From scene- to display-referred colours

- ▶ The primary purpose of tone mapping is to transform an image from *scene-referred* to *display-referred* colours



Tone-mapping in rendering

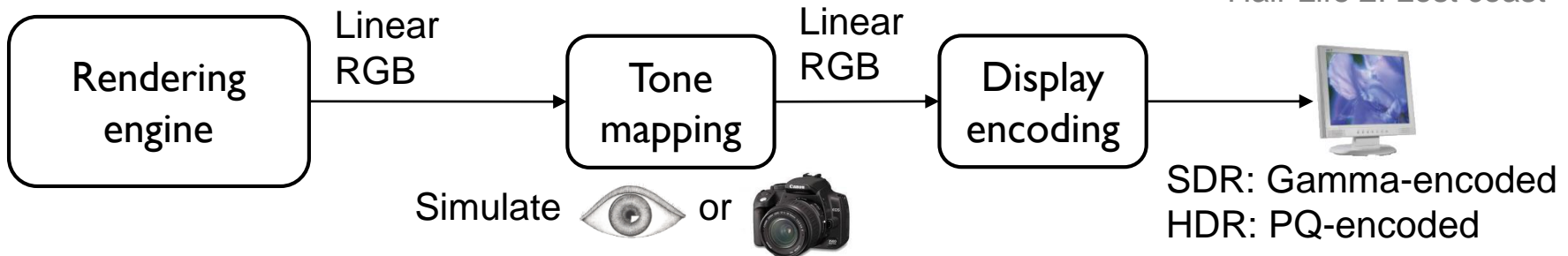
- ▶ Any physically-based rendering requires tone-mapping
- ▶ “HDR rendering” in games is pseudo-physically-based rendering
- ▶ Goal: to simulate a camera or the eye
- ▶ Greatly enhances realism

LDR illumination
No tone-mapping

HDR illumination
Tone-mapping



Half-Life 2: Lost coast



Basic tone-mapping and display coding

- ▶ The simplest form of tone-mapping is the exposure/brightness adjustment:

$$R_d = \frac{R_s}{L_{white}}$$

Display-referred red value

Scene-referred

Scene-referred luminance of white

- ▶ R for red, the same for green and blue
- ▶ No contrast compression, only for a moderate dynamic range
- ▶ The simplest form of display coding is the “gamma”

$$R' = (R_d)^{\frac{1}{\gamma}}$$

Prime (') denotes a gamma-corrected value

Typically $\gamma=2.2$

- ▶ For SDR displays only

Techniques

- ▶ **Arithmetic of HDR images**
- ▶ Display model
- ▶ Tone-curve
- ▶ Color transfer
- ▶ Base-detail separation
- ▶ Glare

Arithmetic of HDR images

- ▶ How do the basic arithmetic operations
 - ▶ Addition
 - ▶ Multiplication
 - ▶ Power function

affect the appearance of an HDR image?

- ▶ We work in the luminance space (NOT luma)
- ▶ The same operations can be applied to linear RGB
 - ▶ Or only to luminance and the colour can be transferred

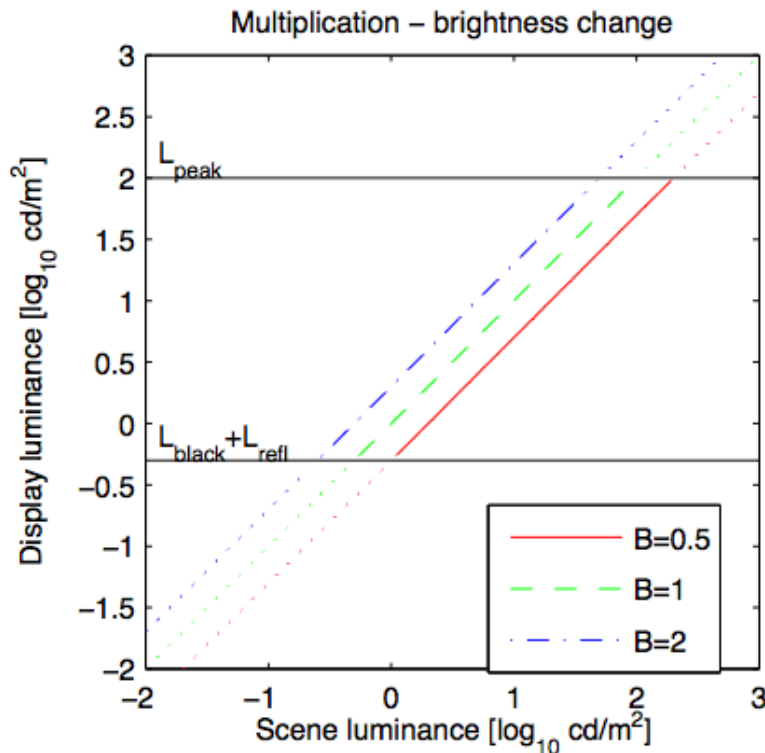
Multiplication – brightness change

Resulting
luminance

Input
luminance

$$T(L_p) = B \cdot L_p$$

Brightness change
parameter



- ▶ Multiplication makes the image brighter or darker
- ▶ It does not change the dynamic range!

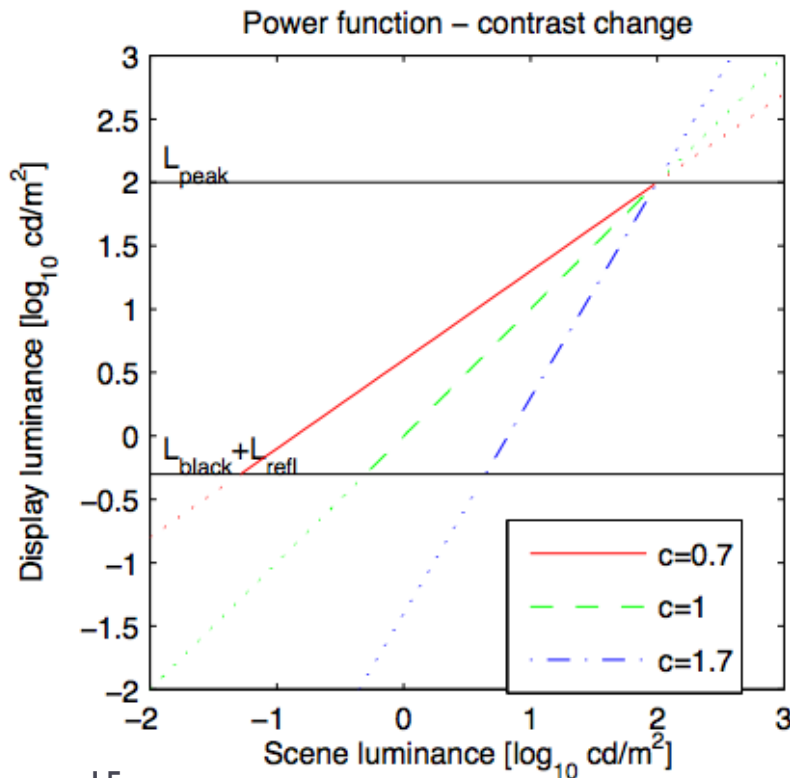
Power function – contrast change

Contrast change
(gamma)

Luminance of
white

$$T(L_p) = \left(\frac{L_p}{L_{white}} \right)^c$$

- ▶ Power function stretches or shrinks image dynamic range
- ▶ It is usually performed relative to a reference white colour/luminance
- ▶ Apparent brightness changes is the side effect of pushing tones towards or away from the white point

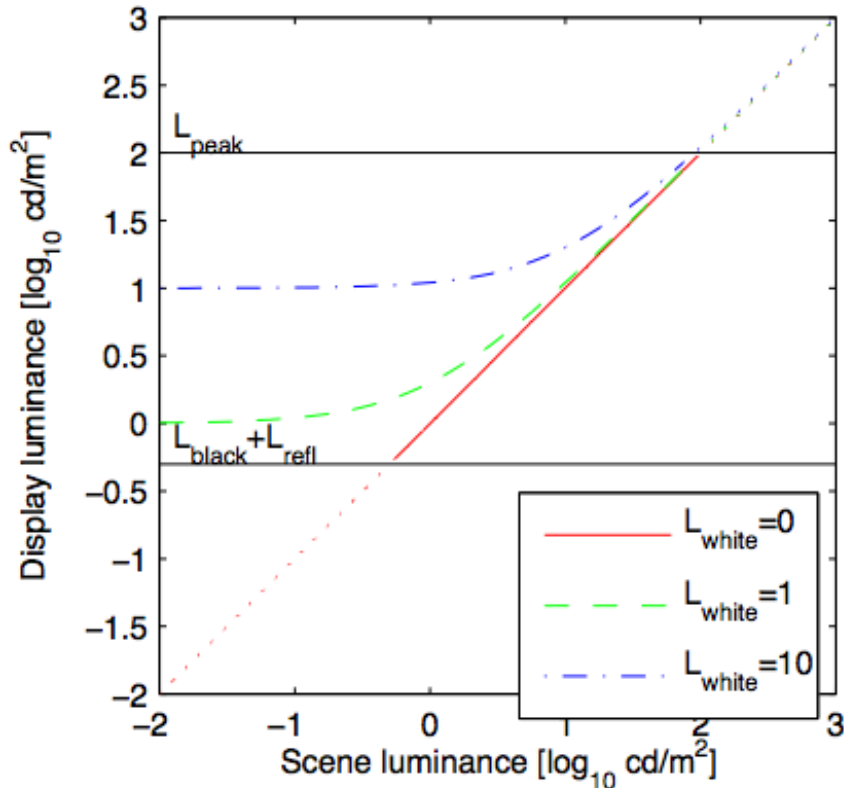


Addition – black level

$$T(L_p) = L_p + F$$

Black level
(flare, fog)

Addition – black level



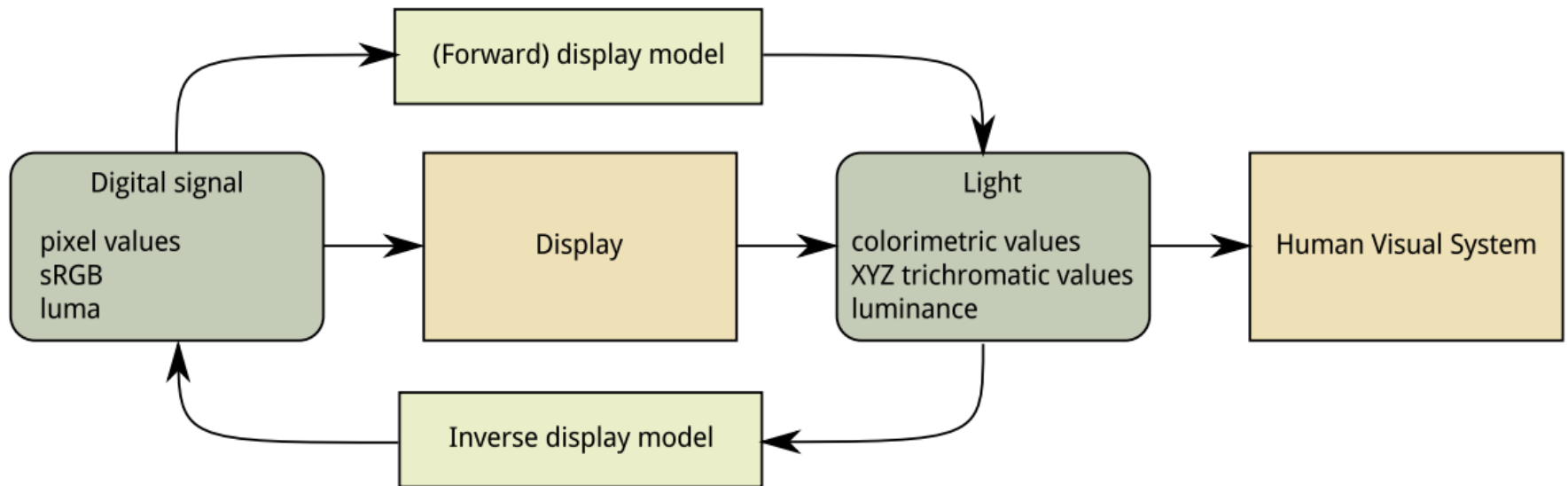
- ▶ Addition elevates black level, adds „fog” to an image
- ▶ It affects mostly darker tones
- ▶ It reduces image dynamic range

Techniques

- ▶ Arithmetic of HDR images
- ▶ **Display model**
- ▶ Tone-curve
- ▶ Color transfer
- ▶ Base-detail separation
- ▶ Glare

Display-adaptive tone mapping

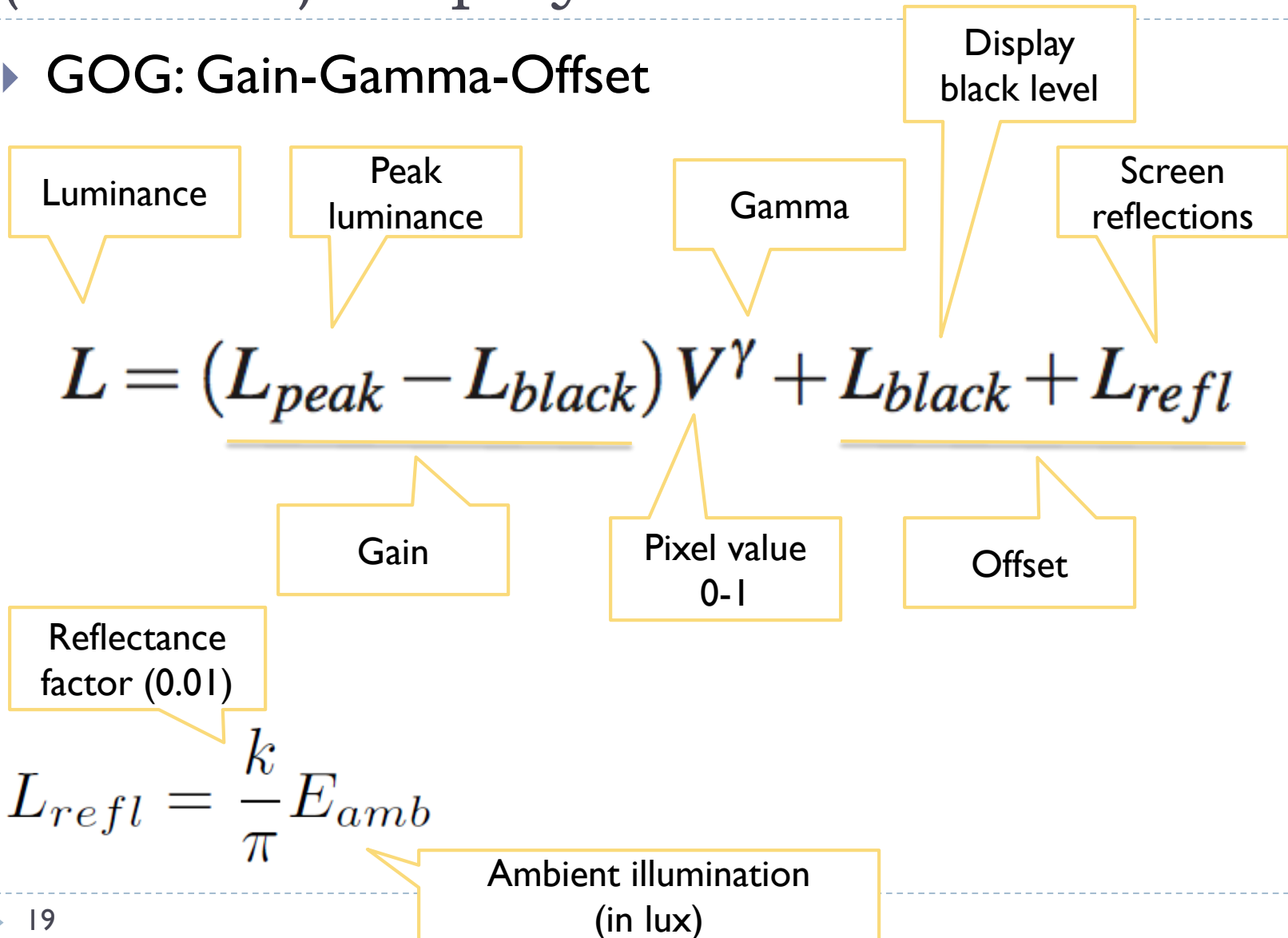
- ▶ Tone-mapping can account for the physical model of a display
 - ▶ How a display transforms pixel values into emitted light
 - ▶ Useful for ambient light compensation



Has a similar role as display encoding, but can account for viewing conditions

(Forward) Display model

► GOG: Gain-Gamma-Offset



Inverse display model

Symbols are the same as for the forward display model

$$V = \left(\frac{L - L_{black} - L_{refl}}{L_{peak} - L_{black}} \right)^{(1/\gamma)}$$

Note: This display model does not address any colour issues. The same equation is applied to red, green and blue color channels. The assumption is that the display primaries are the same as for the sRGB color space.

Ambient illumination compensation

Non-adaptive TMO

Display adaptive TMO



Ambient illumination compensation

Non-adaptive TMO

Display adaptive TMO



10²²

300

10 000

lux

Example: Ambient light compensation

- ▶ We are looking at the screen in bright light

$$L_{peak} = 100 [cd \cdot m^{-2}]$$

$$k = 0.005$$

Modern screens have reflectivity of around 0.5%

$$L_{black} = 0.1 [cd \cdot m^{-2}]$$

$$E_{amb} = 2000 [lux] \quad L_{refl} = \frac{0.005}{\pi} 2000 = 3.183 [cd \cdot m^{-2}]$$

- ▶ We assume that the dynamic of the input is 2.6 ($\approx 400:1$)

$$r_{in} = 2.6 \quad r_{out} = \log_{10} \frac{L_{peak}}{L_{black} + L_{refl}} = 1.77$$

- ▶ First, we need to compress contrast to fit the available dynamic range, then compensate for ambient light

$$L_{out} = \left(\frac{L_{in}}{L_{wp}} \right)^{\frac{r_{out}}{r_{in}}} - L_{refl}$$

The resulting value is in luminance, must be mapped to display luma / gamma corrected values (display encoded)

Simplest, but not the best tone mapping

Techniques

- ▶ Arithmetic of HDR images
- ▶ Display model
- ▶ **Tone-curve**
- ▶ Color transfer
- ▶ Base-detail separation
- ▶ Glare

Tone-curve

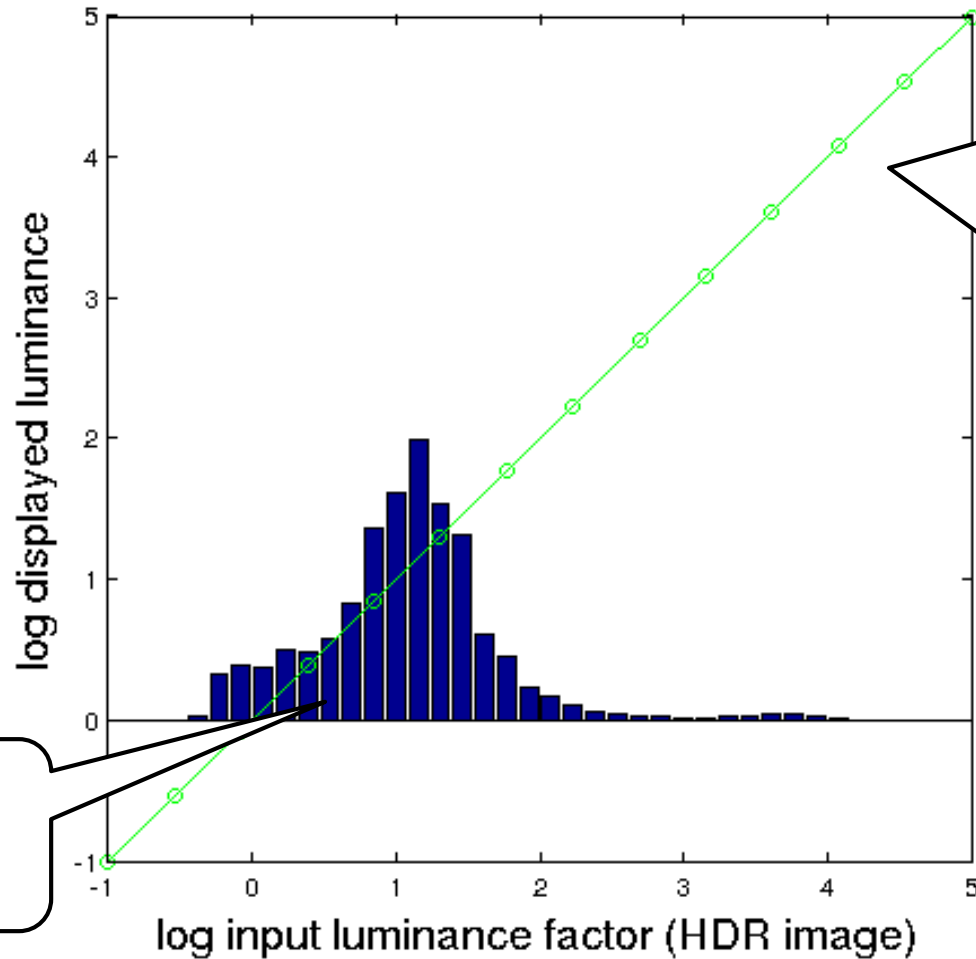
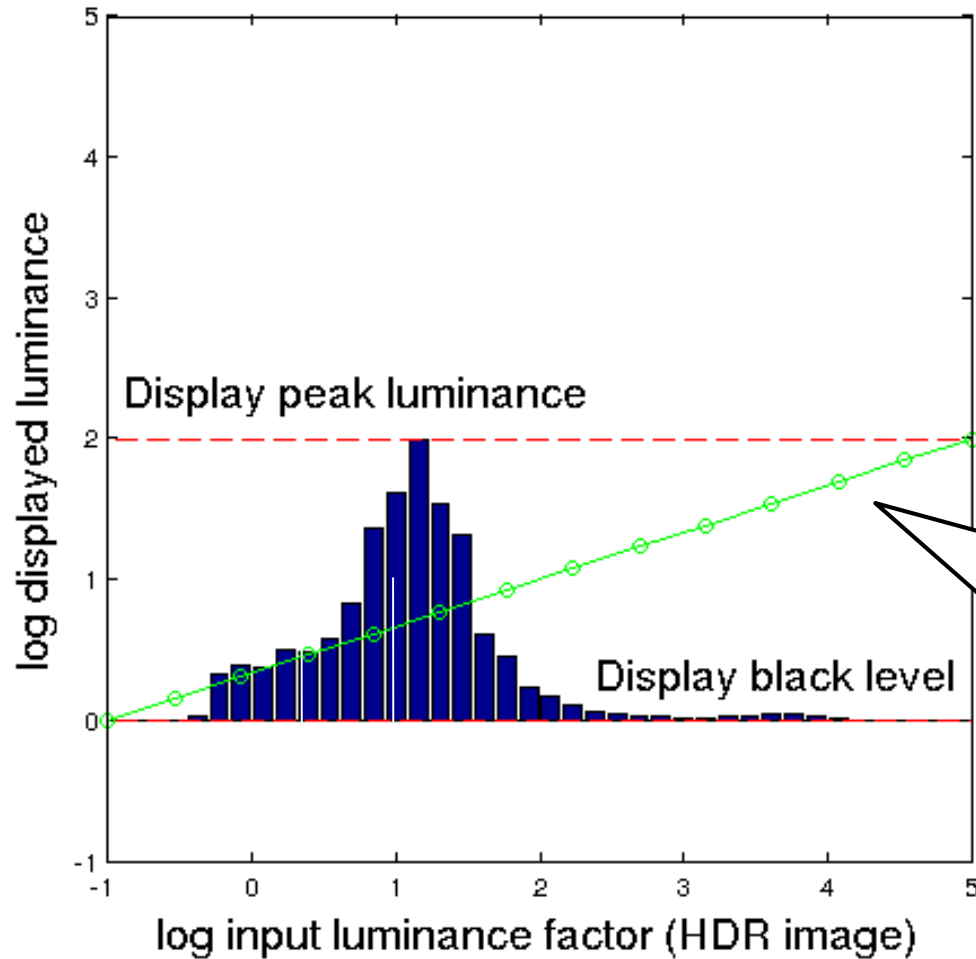


Image histogram

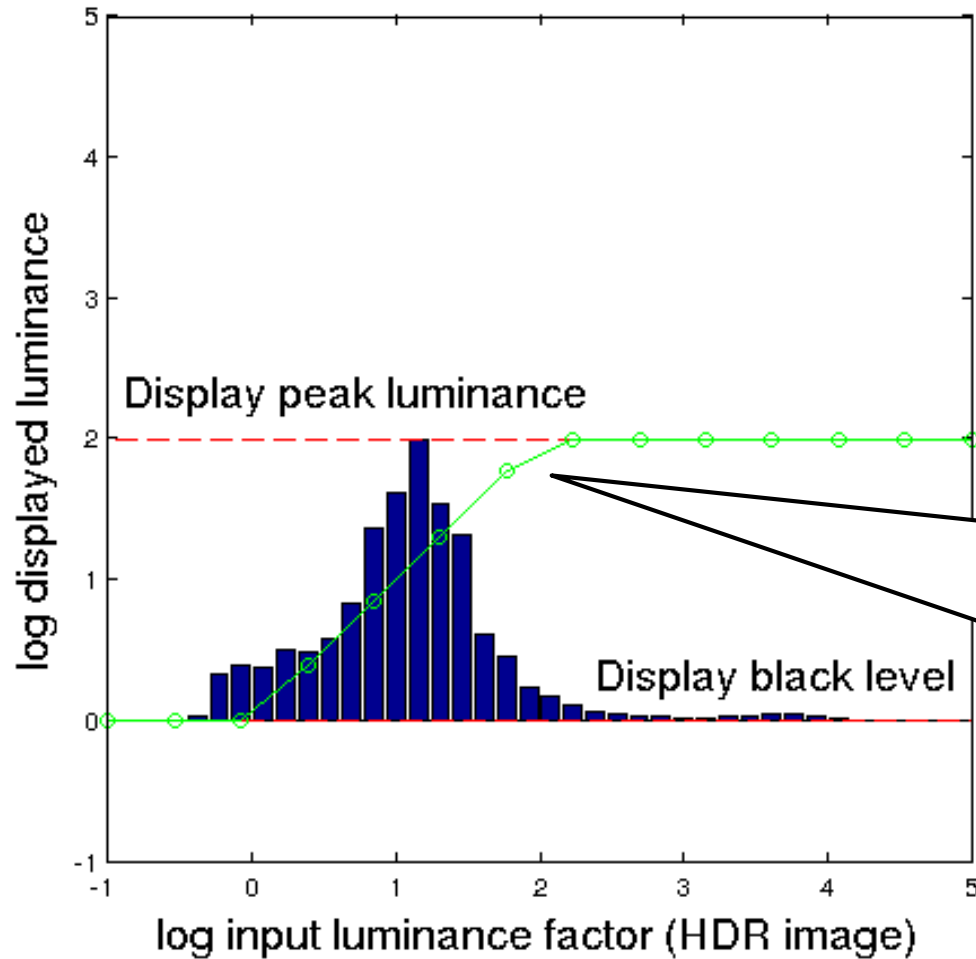
Best tone-mapping is the one which does not do anything, i.e. slope of the tone-mapping curves is equal to 1.

Tone-curve



But in practice contrast (slope) must be limited due to display limitations.

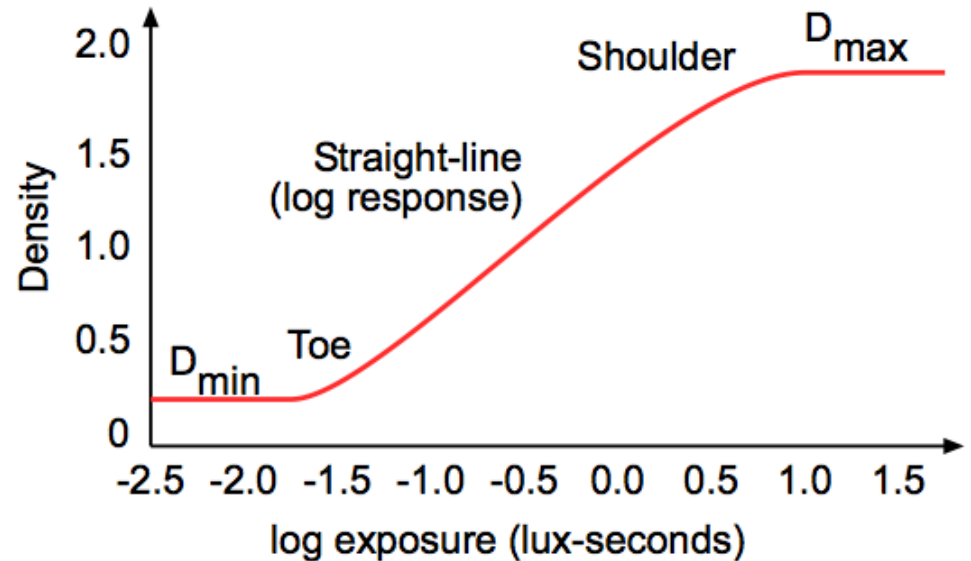
Tone-curve



Global tone-mapping is a compromise between clipping and contrast compression.

Sigmoidal tone-curves

- ▶ Very common in digital cameras
 - ▶ Mimic the response of analog film
 - ▶ Analog film has been engineered over many years to produce good tone-reproduction
- ▶ Fast to compute



Sigmoidal tone mapping

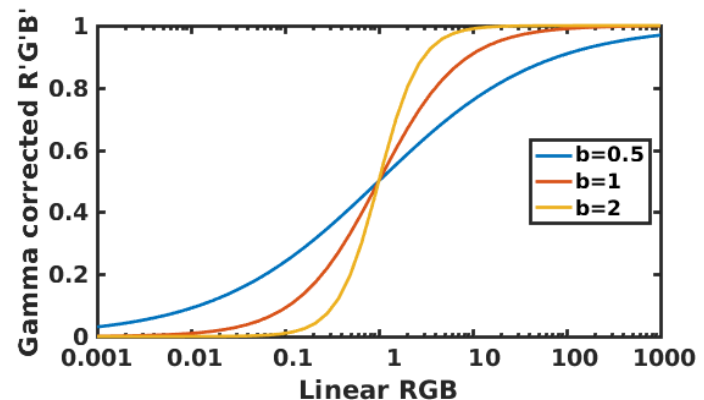
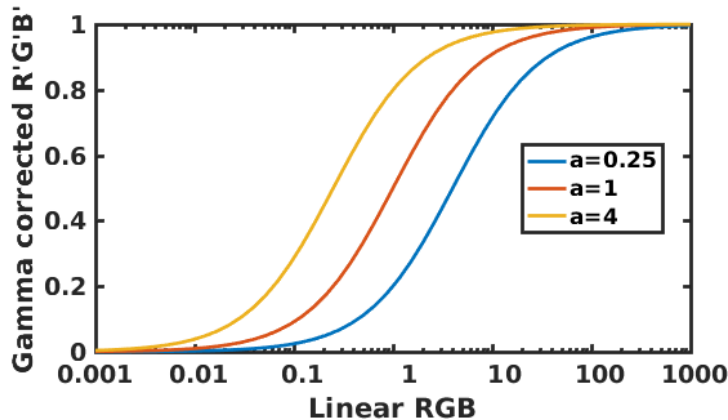
- ▶ Simple formula for a sigmoidal tone-curve:

$$R'(x, y) = \frac{R(x, y)^b}{\left(\frac{L_m}{a}\right)^b + R(x, y)^b}$$

where L_m is the geometric mean (or mean of logarithms):

$$L_m = \exp\left(\frac{1}{N} \sum_{(x,y)} \ln(L(x, y))\right)$$

and $L(x, y)$ is the luminance of the pixel (x, y) .



Sigmoidal tone mapping example

$a=0.25$



$a=1$



$a=4$



$b=0.5$

$b=1$

$b=2$



Histogram equalization

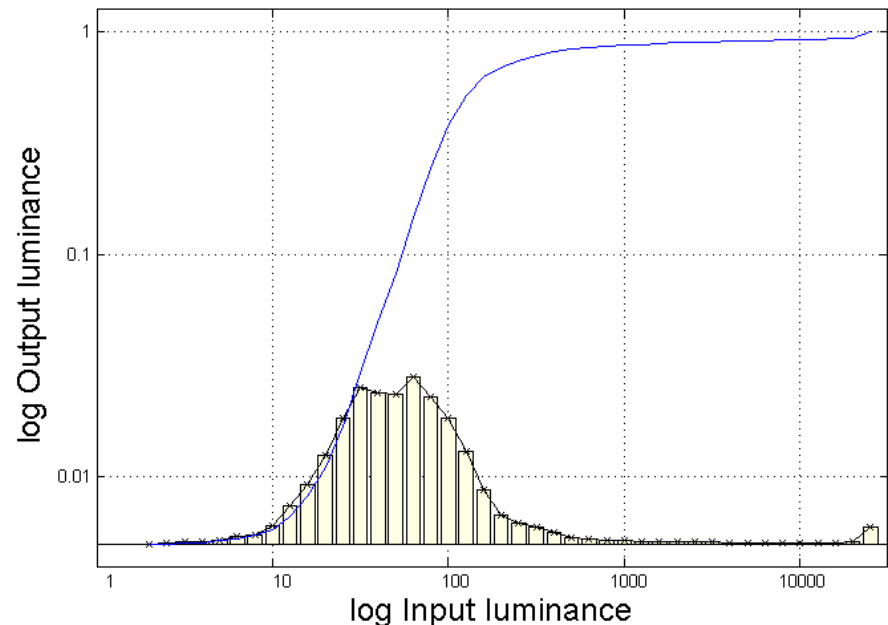
- ▶ 1. Compute normalized cumulative image histogram

$$c(I) = \frac{1}{N} \sum_{i=0}^I h(i) = c(I - 1) + \frac{1}{N} h(I)$$

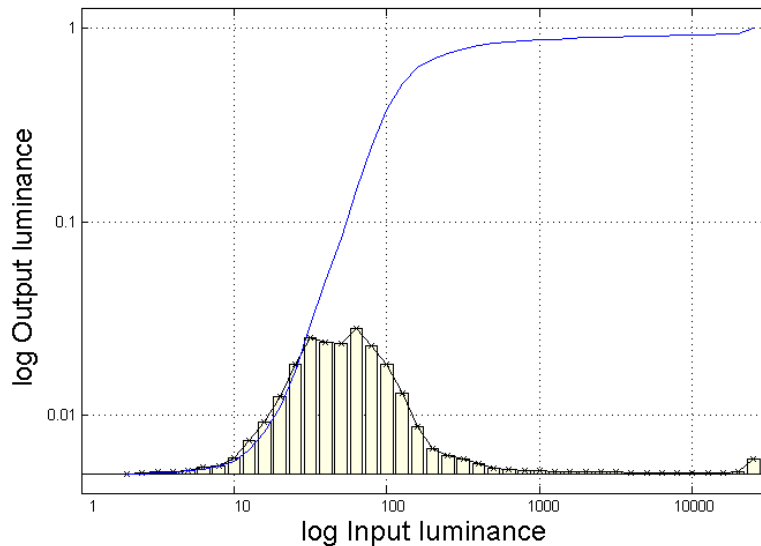
- ▶ For HDR, operate in the log domain
- ▶ 2. Use the cumulative histogram as a tone-mapping function

$$Y_{out} = c(Y_{in})$$

- ▶ For HDR, map the log-I0 values to the $[-dr_{out}; 0]$ range
 - ▶ where dr_{out} is the target dynamic range (of a display)



Histogram equalization



- ▶ Steepest slope for strongly represented bins
 - ▶ If many pixels have the same value - enhance contrast
 - ▶ Reduce contrast, if few pixels
- ▶ Histogram Equalization distributes contrast distortions relative to the “importance” of a brightness level

Histogram adjustment with a linear ceiling

- ▶ [Larson et al. 1997, IEEE TVCG]

Linear mapping



Histogram equalization

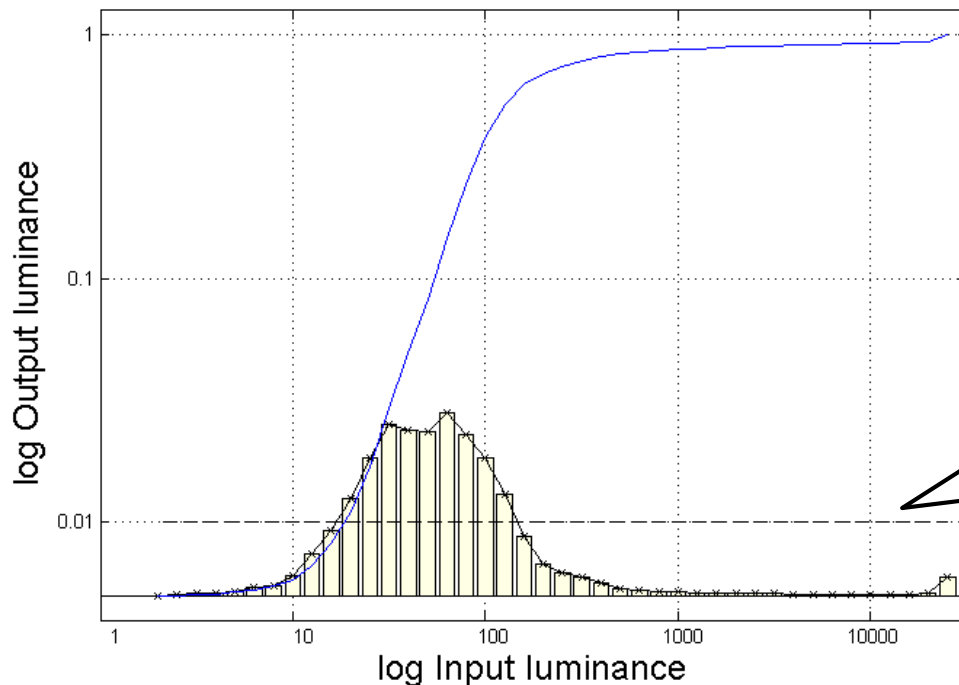


Histogram equalization
with a ceiling



Histogram adjustment with a linear ceiling

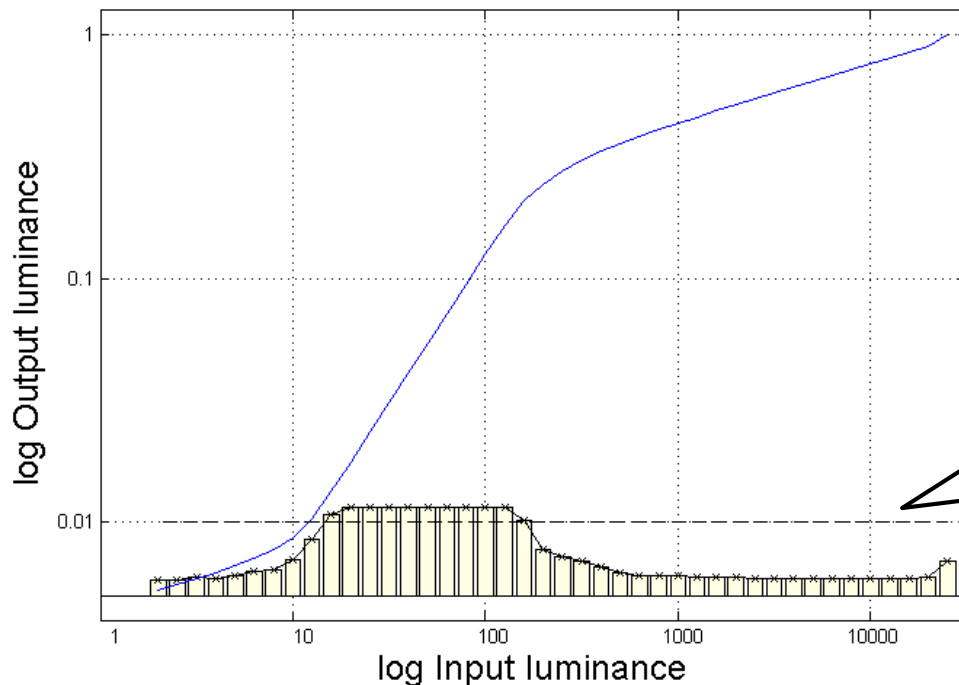
- ▶ Truncate the bins that exceed the ceiling;
- ▶ Distribute the removed counts to all bins;
- ▶ Repeat until converges



Ceiling, based on
the maximum
permissible
contrast

Histogram adjustment with a linear ceiling

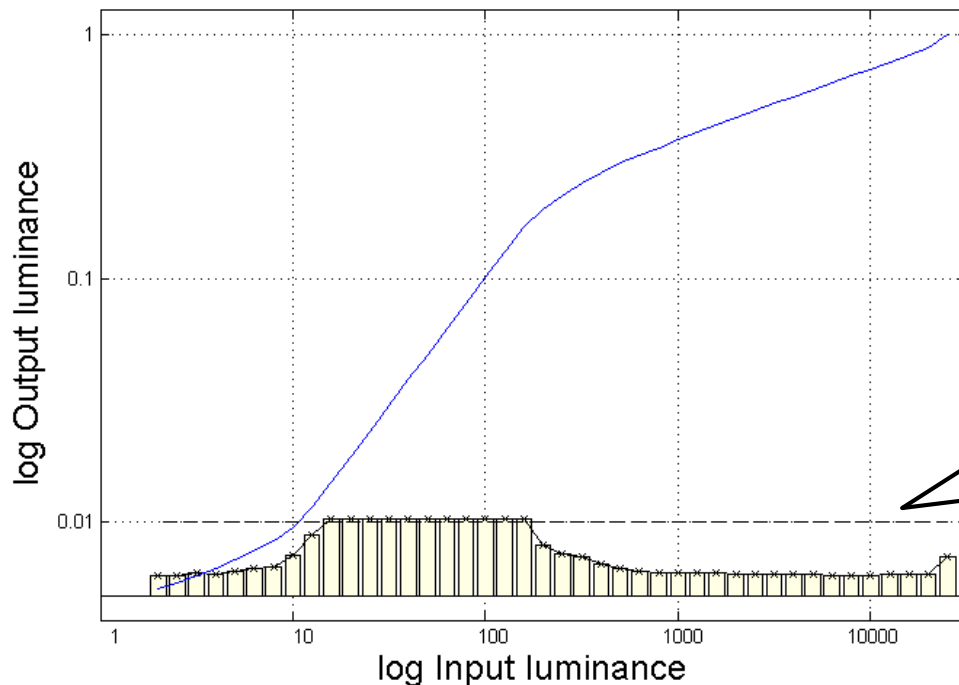
- ▶ Truncate the bins that exceed the ceiling;
- ▶ Distribute the removed counts to all bins;
- ▶ Repeat until converges



Ceiling, based on
the maximum
permissible
contrast

Histogram adjustment with a linear ceiling

- ▶ Truncate the bins that exceed the ceiling;
- ▶ Distribute the removed counts to all bins;
- ▶ Repeat until converges



Techniques

- ▶ Arithmetic of HDR images
- ▶ Display model
- ▶ Tone-curve
- ▶ **Color transfer**
- ▶ Base-detail separation
- ▶ Glare

Colour transfer in tone-mapping

- ▶ Many tone-mapping operators work on luminance
 - ▶ For speed
 - ▶ To avoid colour artefacts
- ▶ Colours must be transferred later from the original image
- ▶ Colour transfer in the linear RGB colour space:

The diagram illustrates the formula for the output red channel value, R_{out} . A box on the left labeled "Output color channel (red)" has an arrow pointing to the R_{out} term in the equation. The equation is $R_{out} = \left(\frac{R_{in}}{L_{in}} \right)^s \cdot L_{out}$. A box on the right labeled "Saturation parameter" has an arrow pointing to the exponent s . Another box on the right labeled "Resulting luminance" has an arrow pointing to the L_{out} term.

$$R_{out} = \left(\frac{R_{in}}{L_{in}} \right)^s \cdot L_{out}$$

- ▶ The same formula applies to green (G) and blue (B) linear colour values

Colour transfer: out-of-gamut problem

- ▶ Colours often fall outside the colour gamut when contrast is compressed



Original image

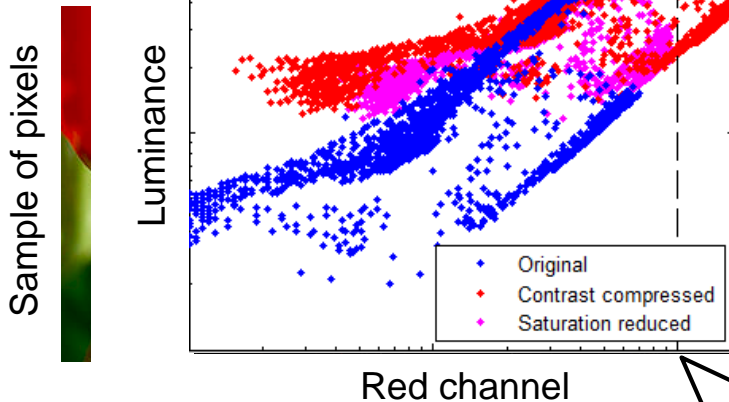


↓ Contrast reduced ($s=1$)



Saturation reduced ($s=0.6$)

Colours before/after processing

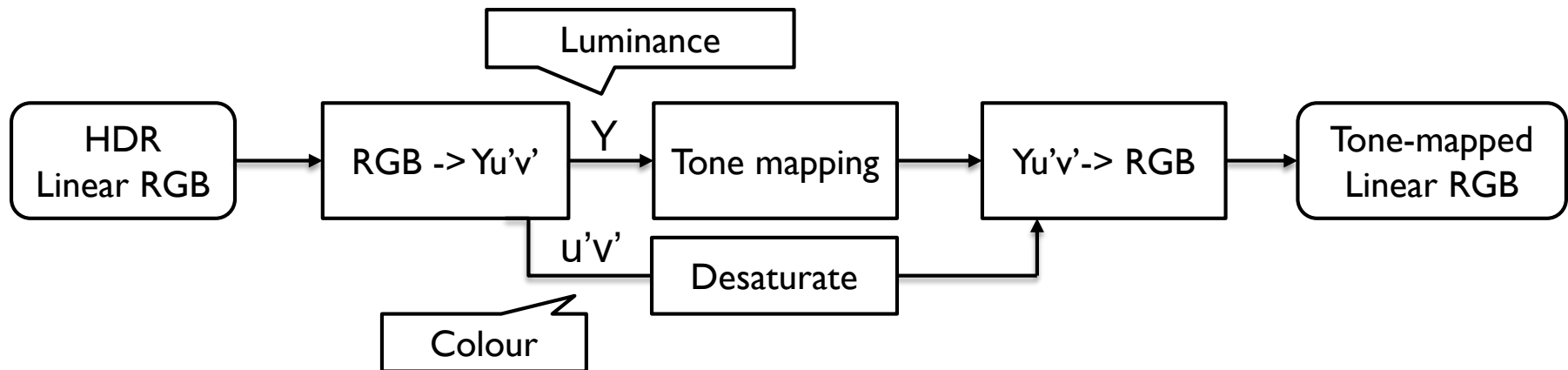


- ▶ Reduction in saturation is needed to bring the colors into gamut

Gamut boundary

Colour transfer: alternative method

- ▶ Colour transfer in linear RGB will alter resulting luminance
- ▶ Colours can be also transferred and saturation adjusted using CIE $u'v'$ chromatic coordinates



- ▶ To correct saturation: $u'_{out} = (u'_{in} - u'_w) \cdot s + u'_w$ Chroma of the white
 $u'_w = 0.1978$
- $v'_{out} = (v'_{in} - v'_w) \cdot s + v'_w$ $v'_w = 0.4683$

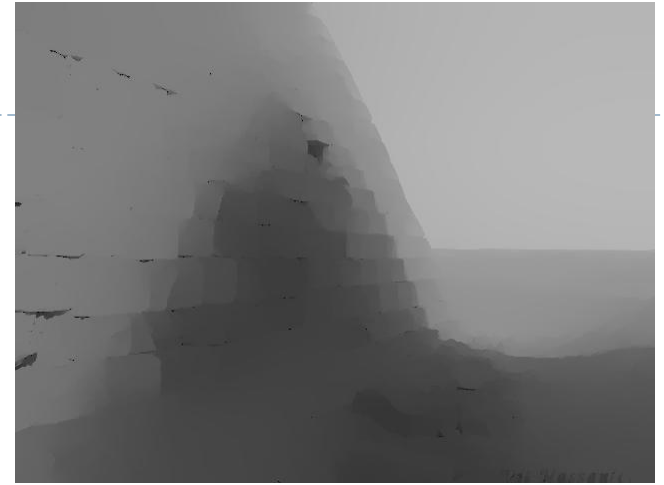
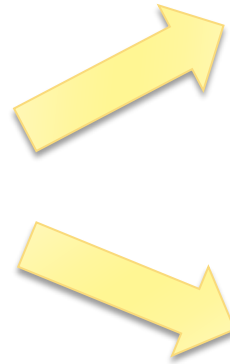
Techniques

- ▶ Arithmetic of HDR images
- ▶ Display model
- ▶ Tone-curve
- ▶ Color transfer
- ▶ **Base-detail separation**
- ▶ Glare

Illumination & reflectance separation



Input



Illumination



Reflectance

$$Y = I \times R$$

Image

Illumination

Reflectance

Illumination and reflectance

Reflectance

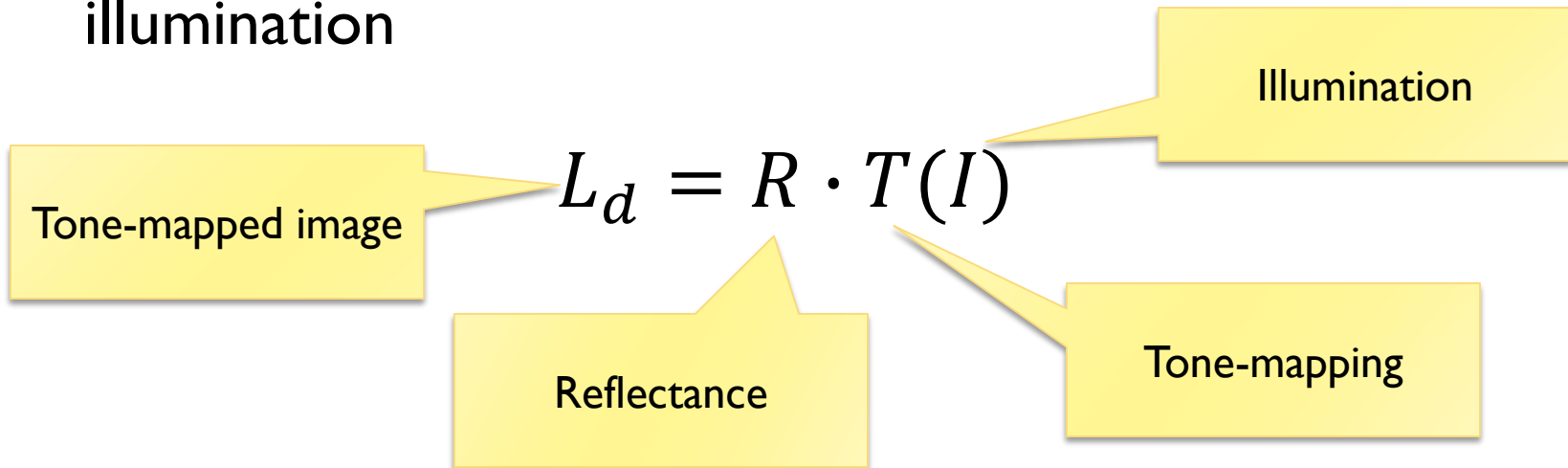
- ▶ White $\approx 90\%$
- ▶ Black $\approx 3\%$
- ▶ Dynamic range $< 100:1$
- ▶ Reflectance critical for object & shape detection

Illumination

- ▶ Sun $\approx 10^9 \text{ cd/m}^2$
- ▶ Lowest perceivable luminance $\approx 10^{-6} \text{ cd/m}^2$
- ▶ Dynamic range 10,000:1 or more
- ▶ Visual system partially discounts illumination

Reflectance & Illumination TMO

- ▶ Hypothesis: *Distortions in reflectance are more apparent than the distortions in illumination*
- ▶ Tone mapping could preserve reflectance but compress illumination



- ▶ for example:

$$L_d = R \cdot (I / L_{white})^c \cdot L_{white}$$

How to separate the two?

- ▶ (Incoming) illumination – slowly changing
 - ▶ except very abrupt transitions on shadow boundaries
- ▶ Reflectance – low contrast and high frequency variations



Gaussian filter

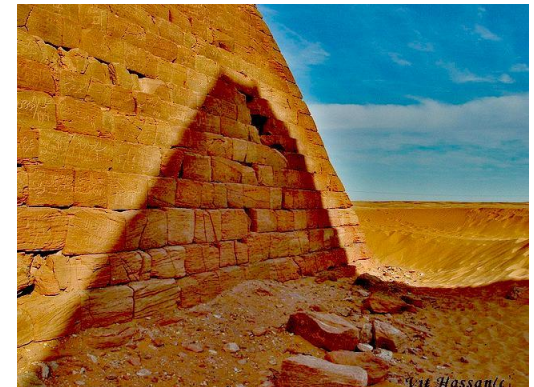
$$f(x) = \frac{1}{\sqrt{2\pi}\sigma_s} e^{-\frac{x^2}{2\sigma_s^2}}$$

- ▶ First order approximation



- ▶ Blurs sharp boundaries
- ▶ Causes halos

Tone mapping
result



Bilateral filter

$$I_p \approx \frac{1}{k_s} \sum_{t \in \Omega} f(p-t) g(L_p - L_t) L_p$$

- ▶ Better preserves sharp edges



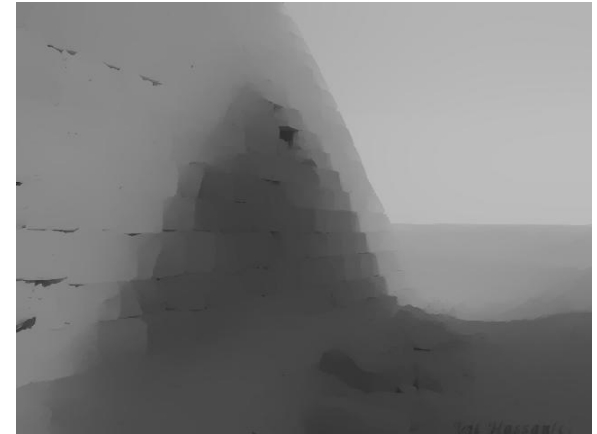
Tone mapping result

- ▶ Still some blurring on the edges
- ▶ Reflectance is not perfectly separated from illumination near edges



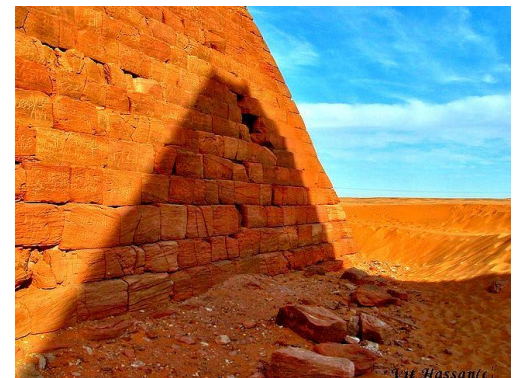
Weighted-least-squares (WLS) filter

- ▶ Stronger smoothing and still distinct edges



Tone mapping result

- ▶ Can produce stronger effects with fewer artifacts
- ▶ See „Advanced image processing” lecture

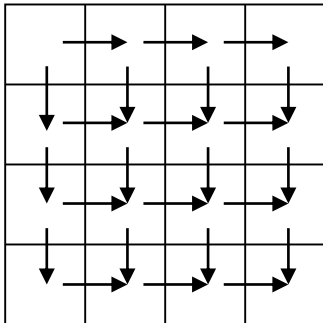


[Farbman et al., SIGGRAPH 2008]

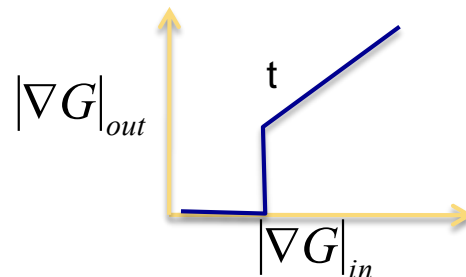
Retinex

- ▶ Retinex algorithm was initially intended to separate reflectance from illumination [Land 1964]
- ▶ There are many variations of Retinex, but the general principle is to eliminate from an image small gradients, which are attributed to the illumination

1 step: compute gradients in log domain



2nd step: set to 0 gradients less than the threshold



3rd step: reconstruct an image from the vector field

$$\nabla^2 I = \text{div } G$$

For example by solving the Poisson equation

Retinex examples

From: <http://dragon.larc.nasa.gov/retinex/757/>



From: http://www.ipol.im/pub/algo/lmps_retinex_poisson_equation/#ref_1

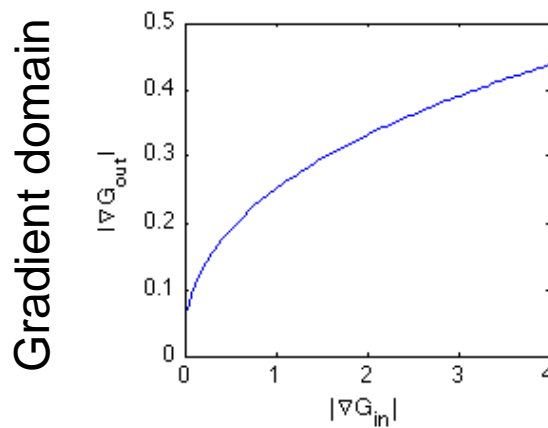
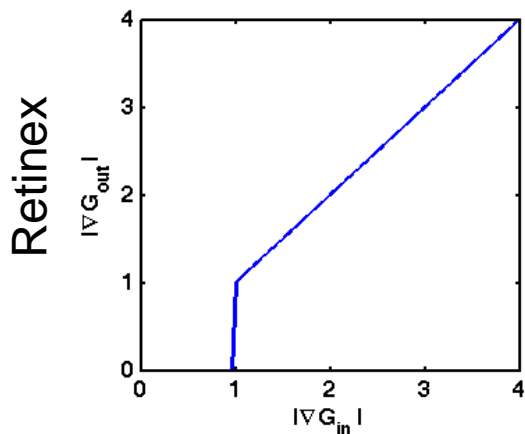


Gradient domain HDR compression



[Fattal et al.,
SIGGRAPH 2002]

- ▶ Similarly to Retinex, it operates on log-gradients
- ▶ But the function amplifies small contrast instead of removing it



- Contrast compression achieved by global contrast reduction
 - Enhance reflectance, then compress everything

Techniques

- ▶ Arithmetic of HDR images
- ▶ Display model
- ▶ Tone-curve
- ▶ Color transfer
- ▶ Base-detail separation
- ▶ **Glare**

Glare



"Alan Wake" © Remedy Entertainment

Glare Illusion



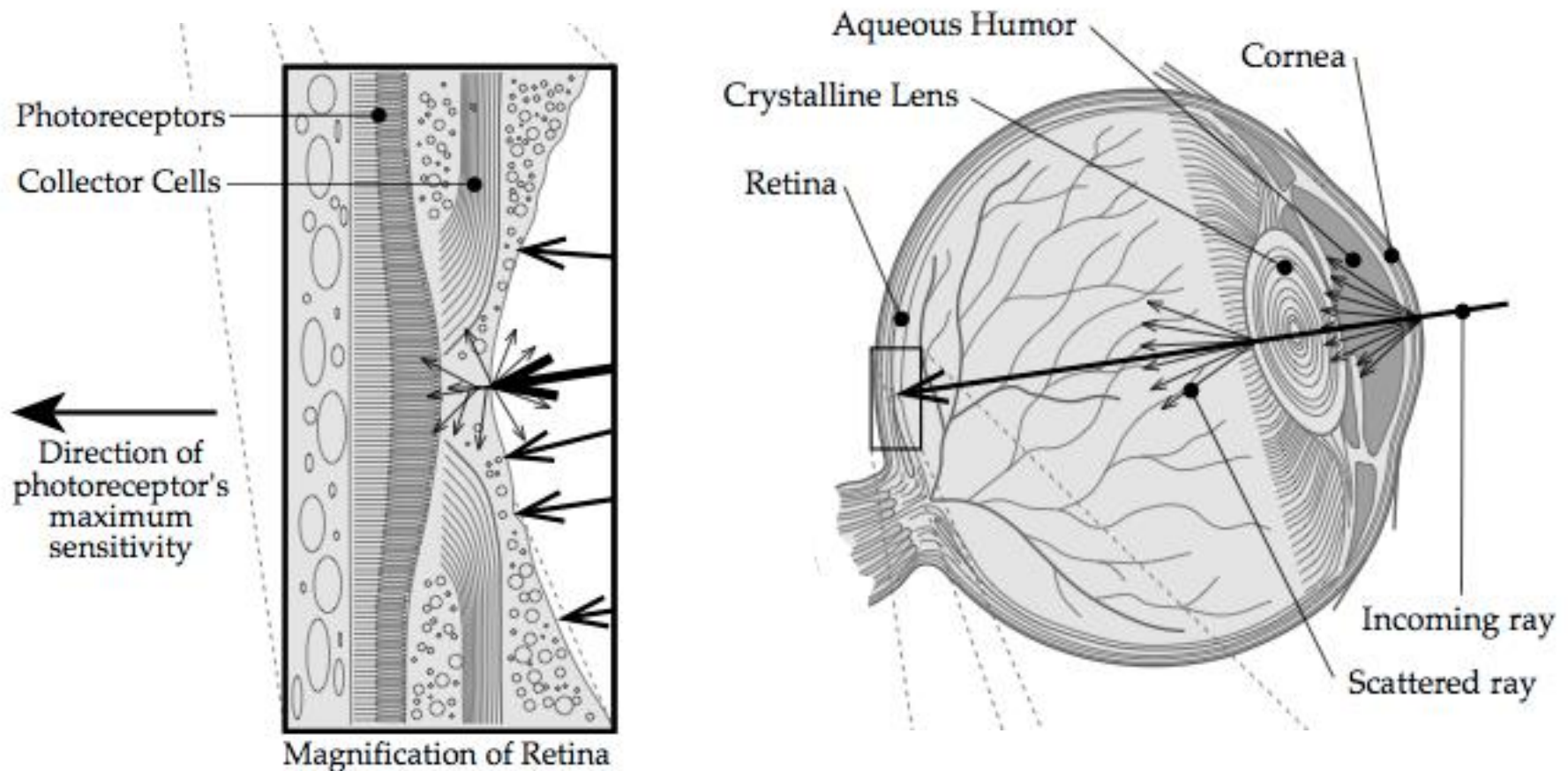
Photography



Painting



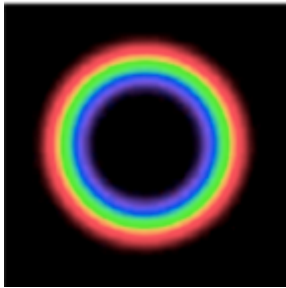
Scattering of the light in the eye



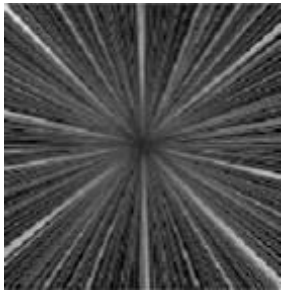
From: Sekuler, R., and Blake, R. Perception, second ed. McGraw- Hill, New York, 1990



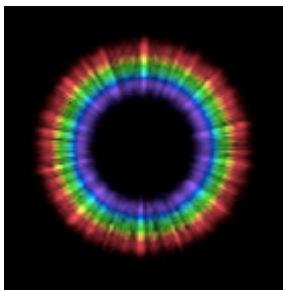
Ciliary corona and lenticular halo



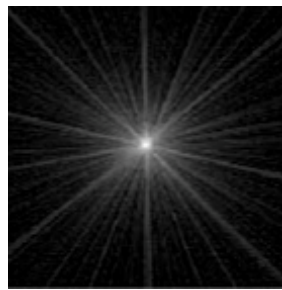
*



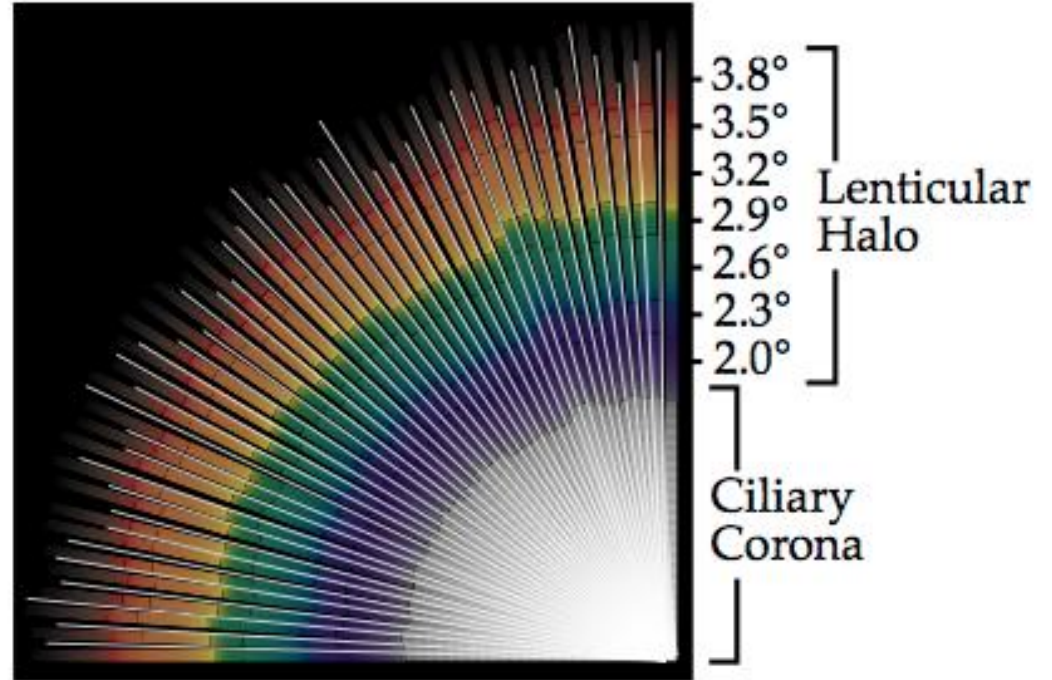
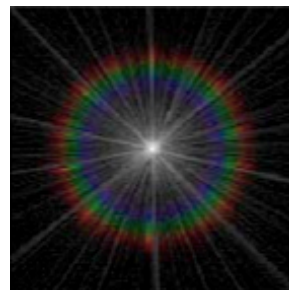
=



+

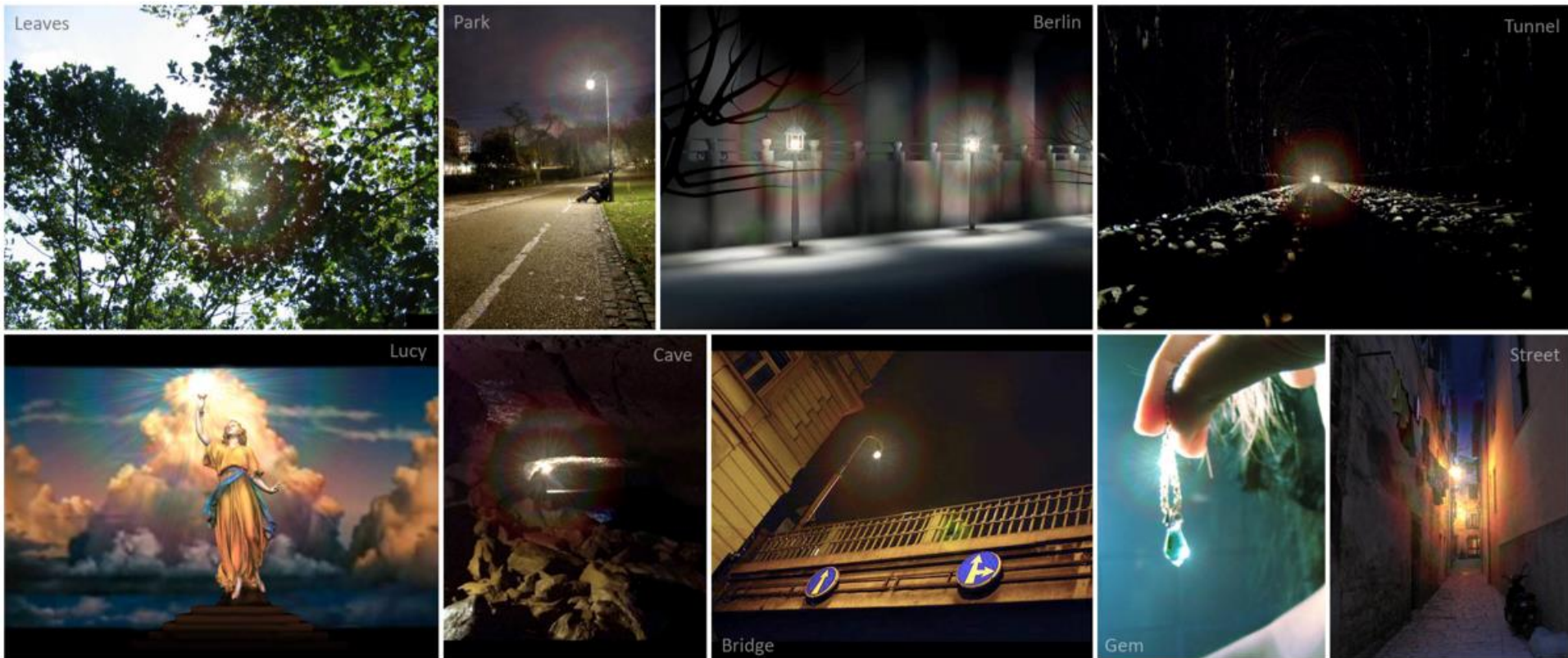


=



From: Spencer, G. et al.
1995. Proc. of
SIGGRAPH. (1995)

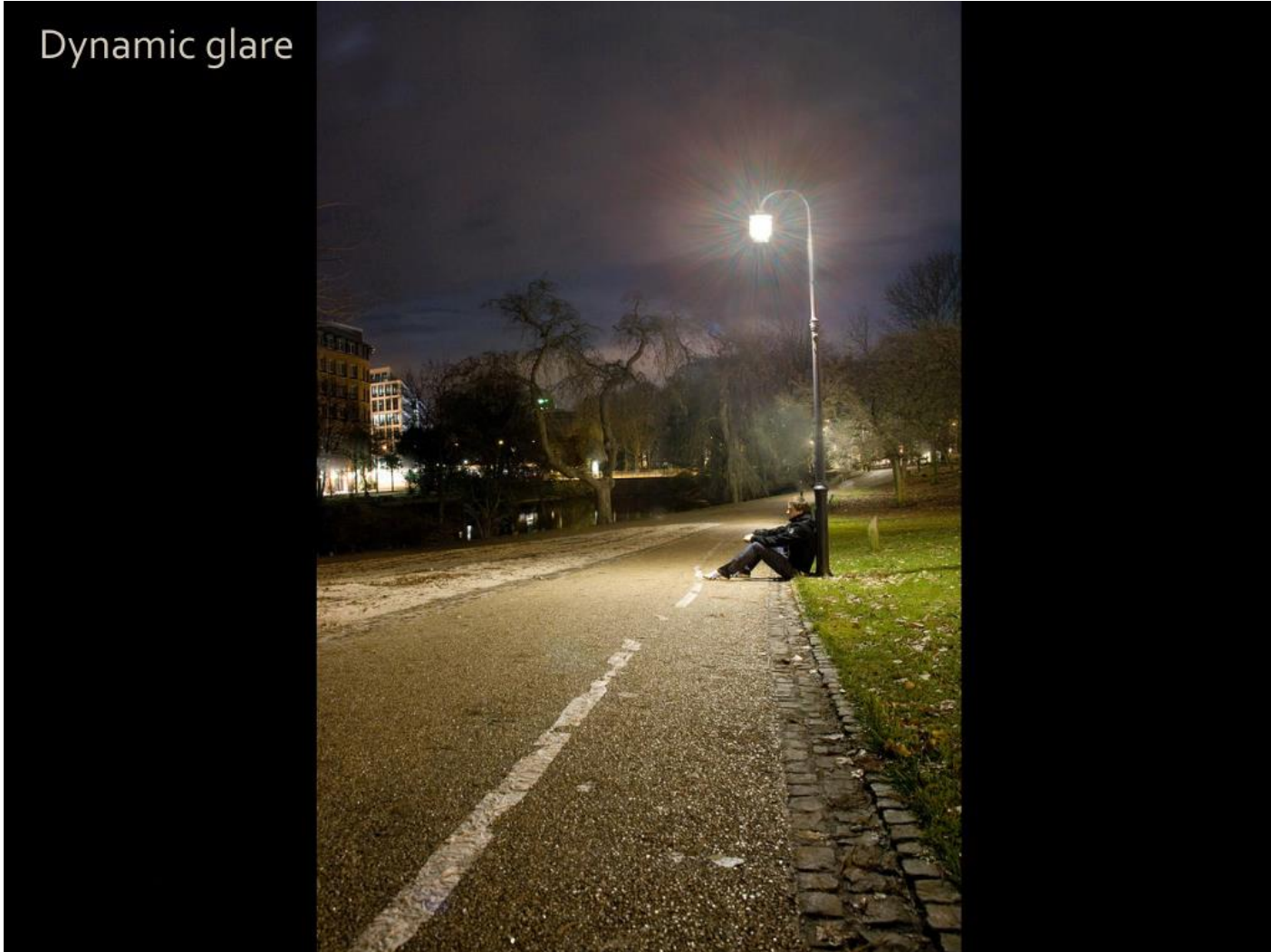
Examples of simulated glare



[From Ritschel et al, Eurographics 2009]

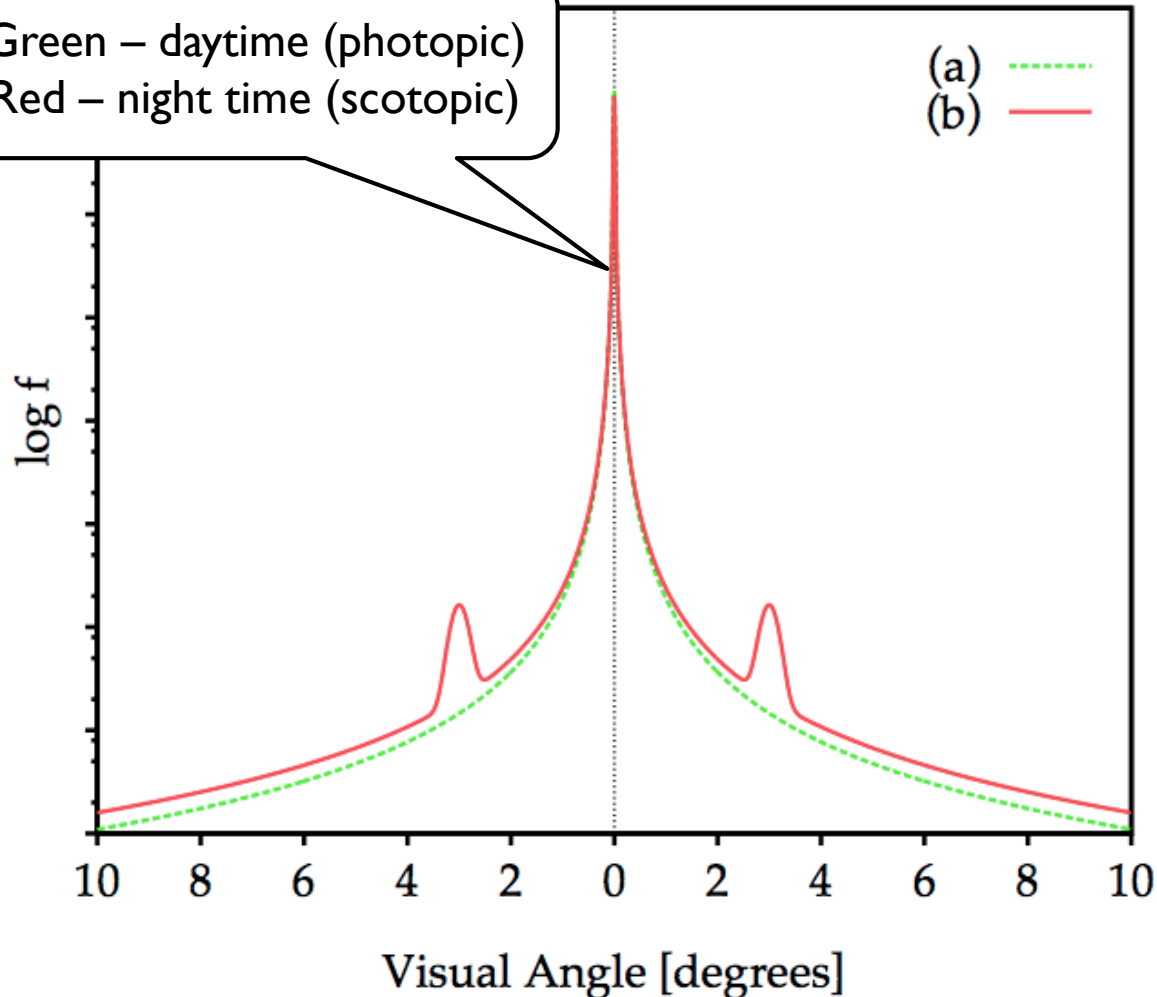
Temporal glare

Dynamic glare



Point Spread Function of the eye

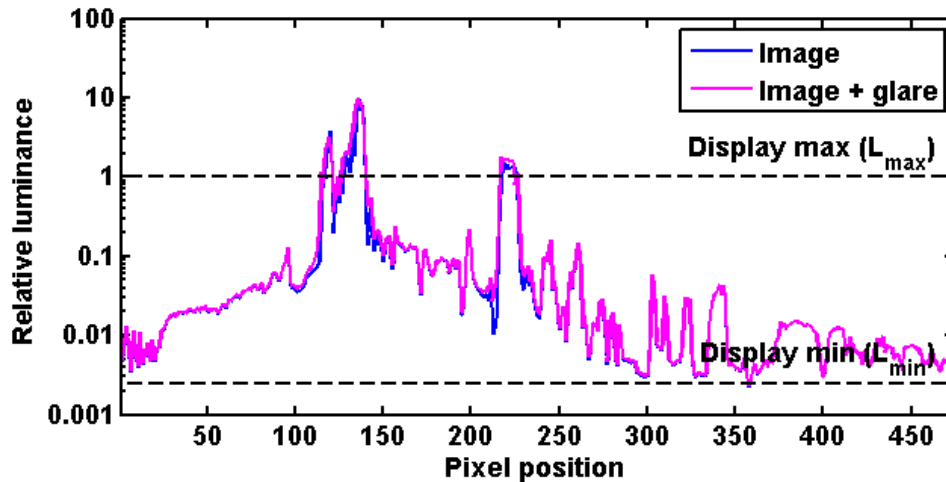
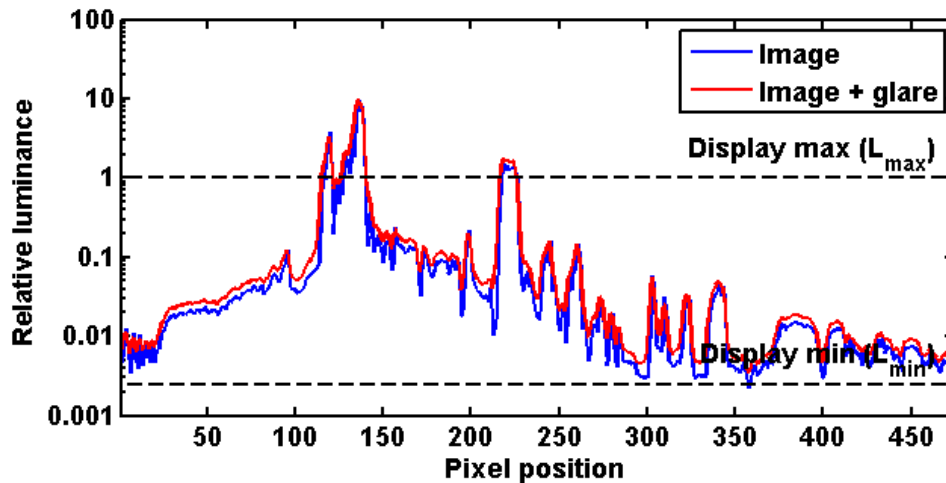
Green – daytime (photopic)
Red – night time (scotopic)



- ▶ What portion of the light is scattered towards a certain visual angle
- ▶ To simulate:
 - ▶ construct a digital filter
 - ▶ convolve the image with that filter

From: Spencer, G. et al. 1995.
Proc. of SIGGRAPH. (1995)

Selective application of glare



- ▶ A) Glare applied to the entire image

$$I_g = I * G$$

Glare kernel (PSF)

- ▶ Reduces image contrast and sharpness

- B) Glare applied only to the clipped pixels

$$I_g = I + I_{clipped} * G - I_{clipped}$$

$$\text{where } I_{clipped} = \begin{cases} I & \text{for } I > 1 \\ 0 & \text{otherwise} \end{cases}$$

Better image quality

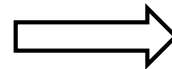
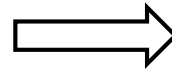


Selective application of glare

A) Glare applied to the entire image



Original image

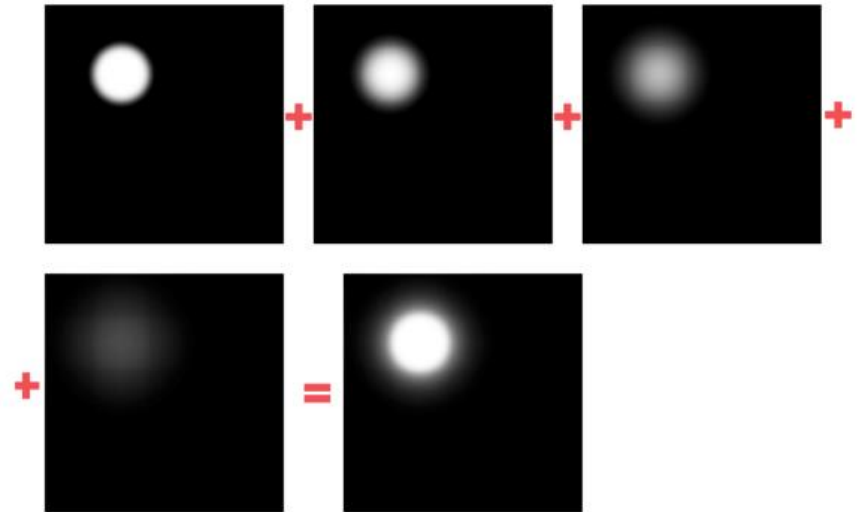


B) Glare applied to clipped pixels only



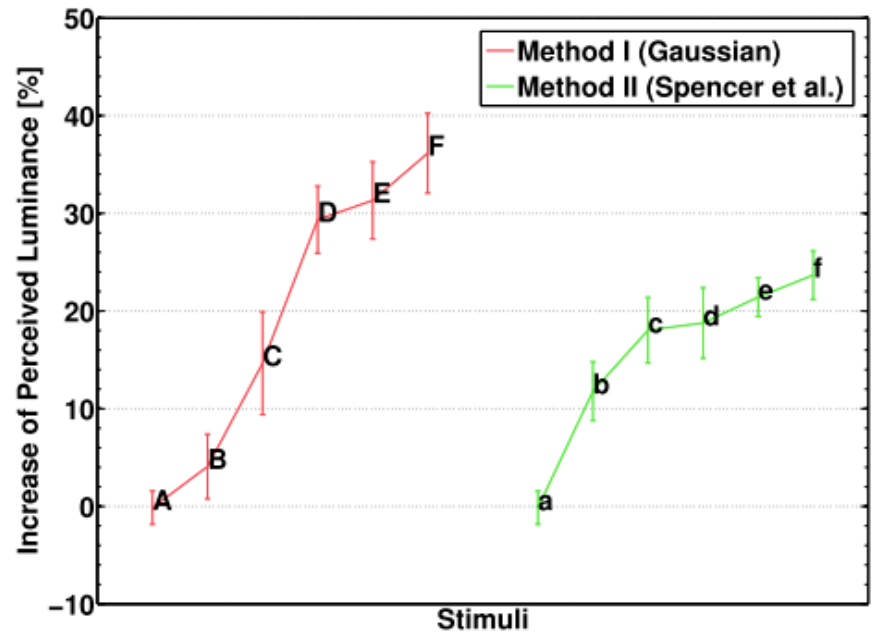
Glare (or bloom) in games

- ▶ Convolution with large, non-separable filters is too slow
- ▶ The effect is approximated by a combination of Gaussian filters
 - ▶ Each filter with different “sigma”
- ▶ The effect is meant to look good, not be an accurate model of light scattering
- ▶ Some games simulate camera rather than the eye



Does the exact shape of the PSF matter?

- ▶ The illusion of increased brightness works even if the PSF is very different from the PSF of the eye



red - Gaussian



green - accurate



[Yoshida et al., APGV 2008]

HDR rendering – motion blur



From LDR pixels

From HDR pixels

References

- ▶ **Comprehensive book on HDR Imaging**
 - ▶ E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*, 2nd editio. Morgan Kaufmann, 2010.
- ▶ **Overview of HDR imaging & tone-mapping**
 - ▶ http://www.cl.cam.ac.uk/~rkm38/hdri_book.html
- ▶ **Review of recent video tone-mapping**
 - ▶ A comparative review of tone-mapping algorithms for high dynamic range video
Gabriel Eilertsen, Rafal K. Mantiuk, Jonas Unger, Eurographics State-of-The-Art Report 2017.
- ▶ **Selected papers on tone-mapping:**
 - ▶ G.W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Trans. Vis. Comput. Graph.*, vol. 3, no. 4, pp. 291–306, 1997.
 - ▶ R. Wanat and R. K. Mantiuk, "Simulating and compensating changes in appearance between day and night vision," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 33, no. 4, p. 147, 2014.
 - ▶ Spencer, G. et al. 1995. Physically-Based Glare Effects for Digital Images. Proceedings of SIGGRAPH. (1995), 325–334
 - ▶ Ritschel, T. et al. 2009. Temporal Glare: Real-Time Dynamic Simulation of the Scattering in the Human Eye. *Computer Graphics Forum*. 28, 2 (Apr. 2009), 183–192
 - ▶ ...