Tone Reproduction

First Example: Linear Rescaling

- **Rescaling** is a point processing technique that alters the **contrast** and/or **brightness** of an image.
- In photography, **exposure** is a measure of how much light is projected onto the imaging sensor.
  - **Overexposure**: more light than what the sensor can measure.
  - **Underexposure**: sensor is unable to detect the light.
- Images which are underexposed or overexposed can frequently be improved by brightening or darkening them.
- The contrast of an image can be altered to bring out the internal structure of the image.

Rescaling Math

- Given a sample C_{in} of the source image, rescaling computes the output sample, C_{out}, using the scaling function

$$C_{out} = \alpha C_{in} + \beta$$

- α is a real-valued scaling factor known as **gain**
- β is a real-valued scaling factor known as **bias**

Effects of Rescaling

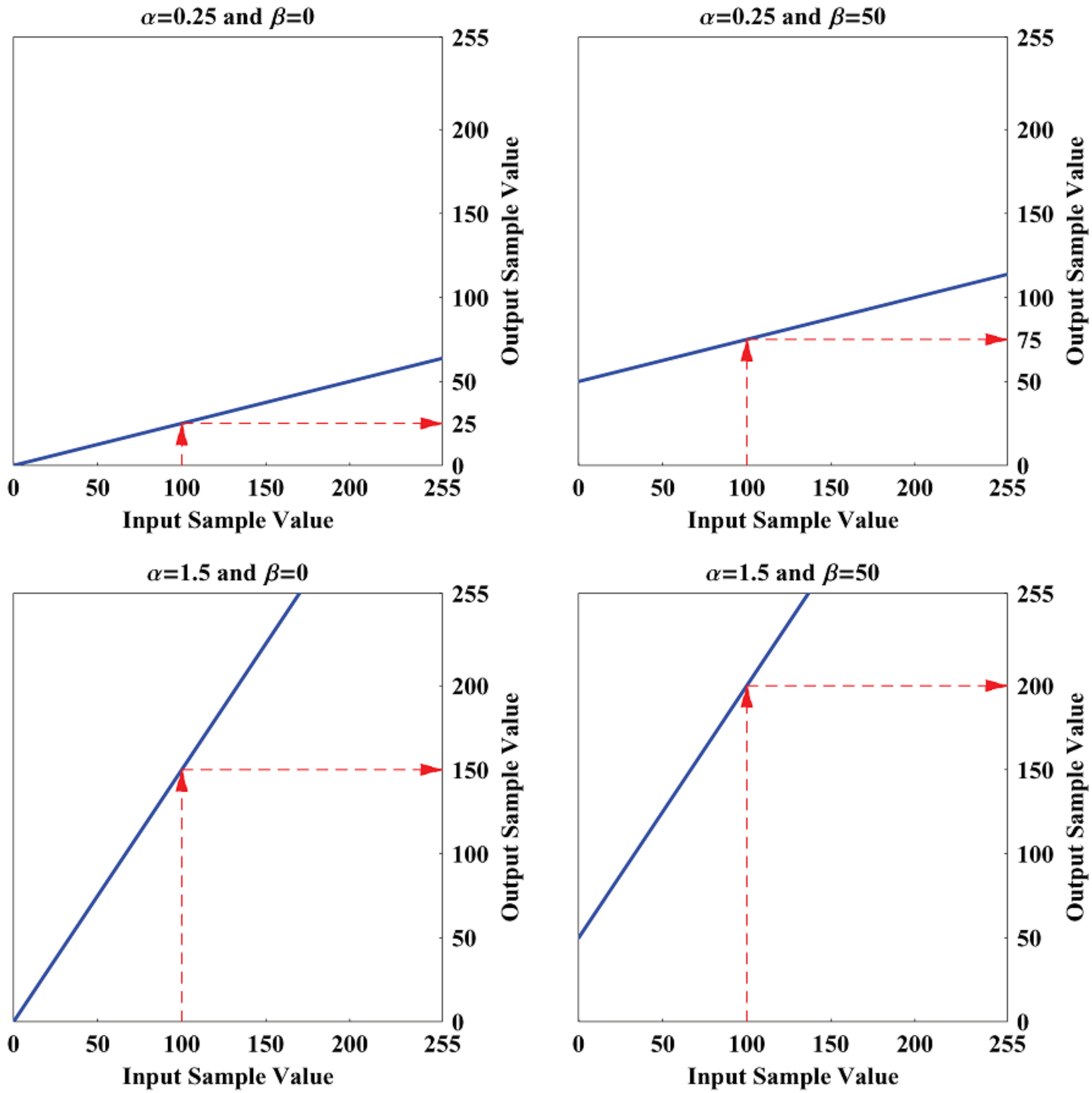


Figure 5.2. Graph of the linear scaling function with various gain and bias settings.

Why Use Both α, β?

- Consider two rescaled source samples of S rescaled to S'.

$$\begin{aligned} S_1' &= \alpha S_1 + \beta, \\ S_2' &= \alpha S_2 + \beta. \end{aligned}$$

- Calculate the **contrast** (the absolute difference) between the source and destination, called **Δ**S and **Δ**S'.

$$\begin{aligned} \triangle S' &= |S_1' - S_2'|, \\ \triangle S &= |S_1 - S_2|. \end{aligned}$$

- Now consider the relative change in contrast between the source and destination.

$$\frac{\triangle S'}{\triangle S} = \frac{|S_1' - S_2'|}{|S_1 - S_2|}.$$

Why Use Both α, β?

- The relative change in contrast can be simplified as

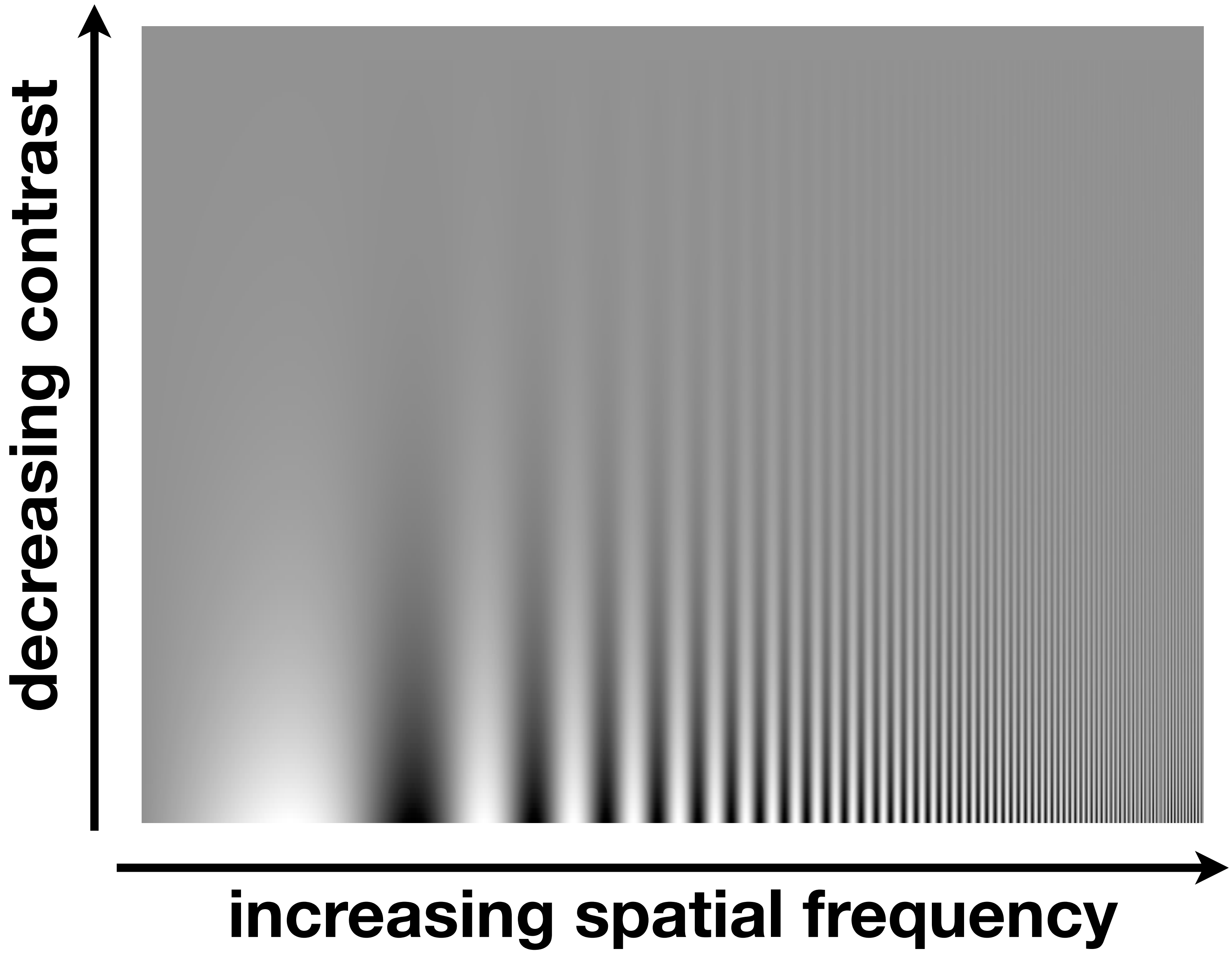
$$\boxed{\begin{aligned} \frac{\triangle S'}{\triangle S} &= \frac{|(\alpha S_1 + \beta) - (\alpha S_2 + \beta)|}{|S_1 - S_2|} \\ &= \frac{|\alpha| \cdot |S_1 - S_2|}{|S_1 - S_2|} \\ &= |\alpha|. \end{aligned}}$$

- Thus, gain (**α**) controls the change in contrast.
- Whereas bias (**β**) *does not* affect the contrast
- Bias, however, controls the final **brightness** of the rescaled image. Negative bias darkens and positive bias brightens the image

Sidebar: Relating Contrast Sensitivities to Signal Processing

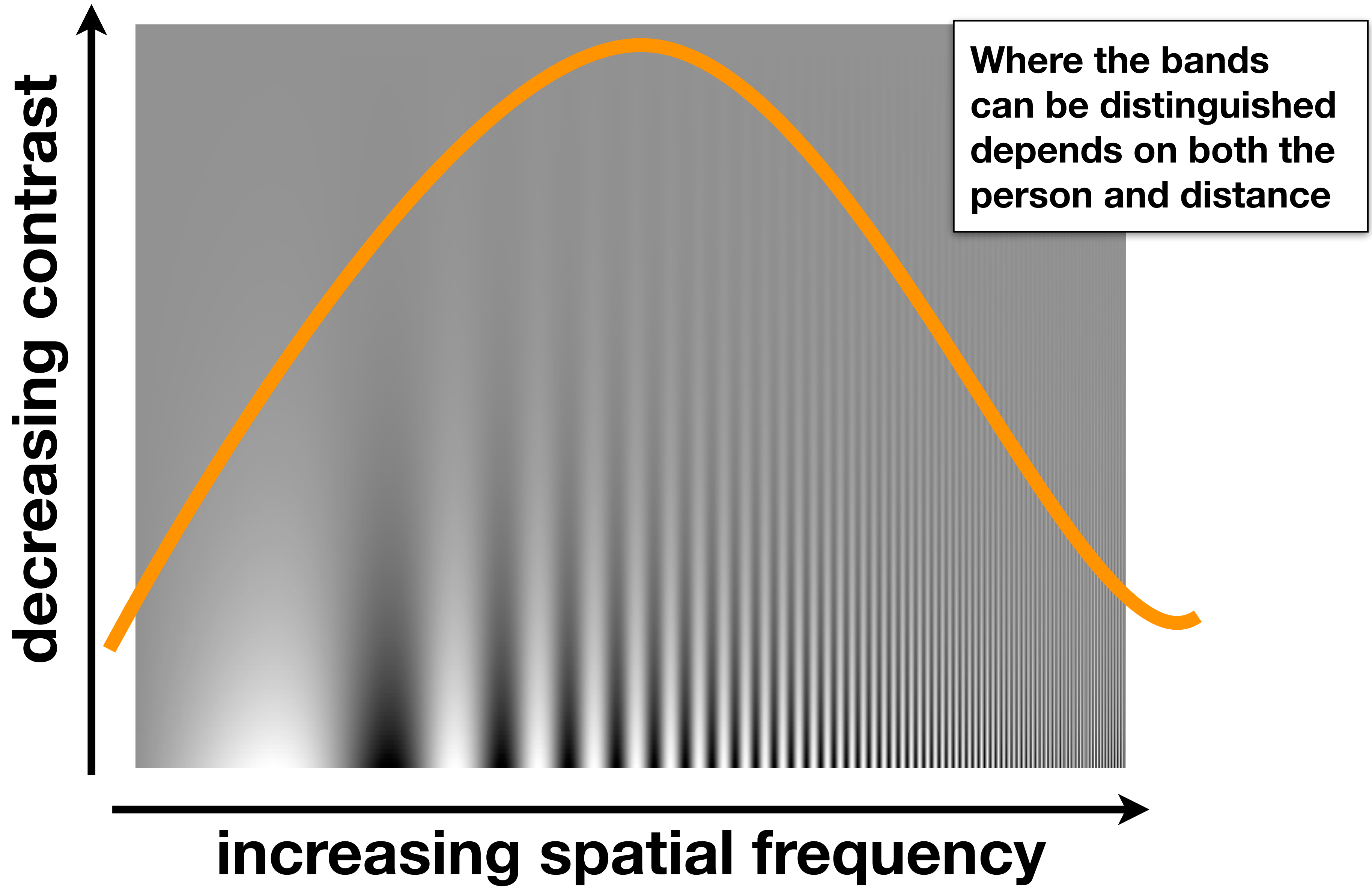
Contrast Sensitivity Function

Campbell-Robson Chart



Contrast Sensitivity Function

Campbell-Robson Chart



Contrast Sensitivities Vary by Channel

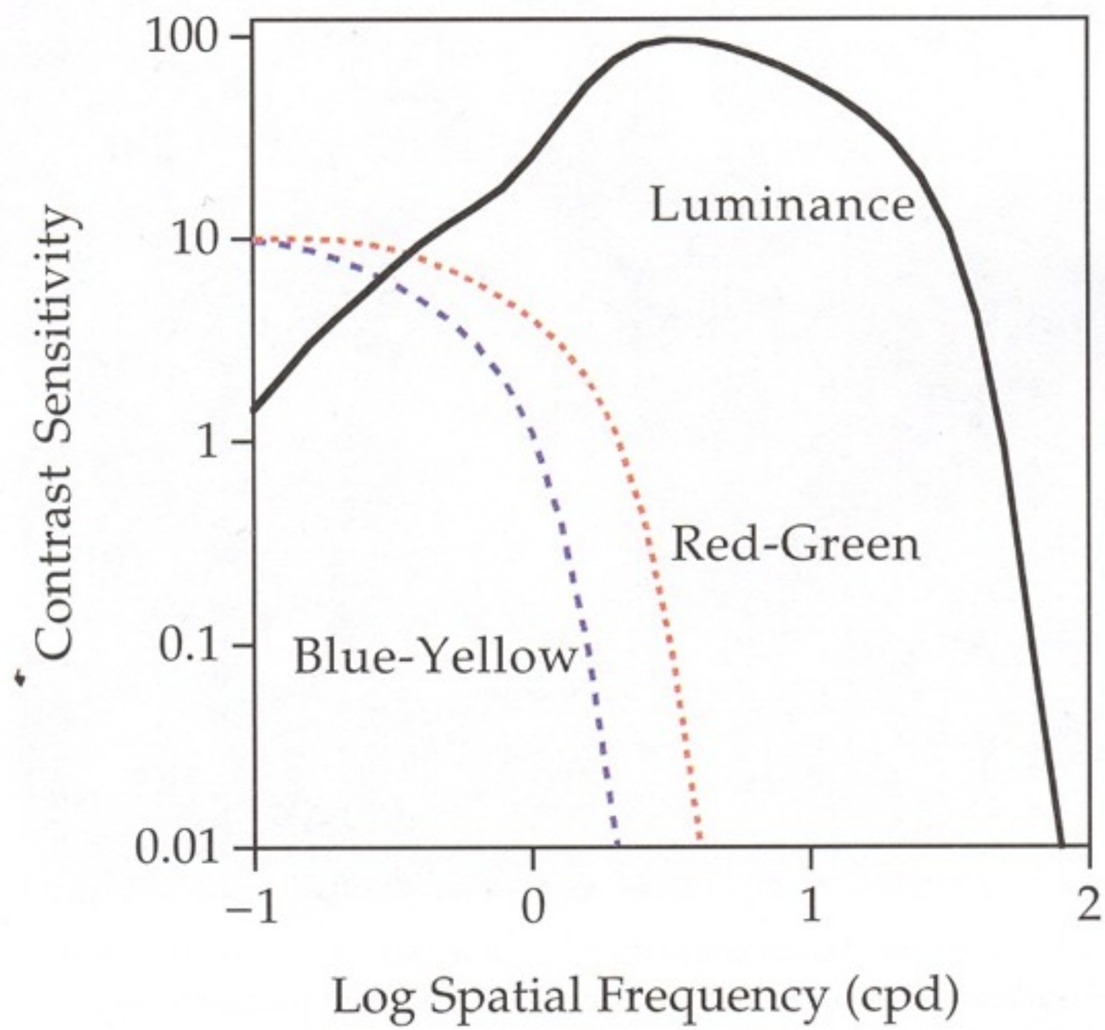


Figure 1-18. Spatial contrast sensitivity functions for luminance and chromatic contrast.

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

Original

Blur Lightness

Blur a & b

Important: Clamping

- Rescaling may produce samples that lie outside of the output images (e.g. below 0 or above 255 in 8-bit images)
- **Clamping** the output values ensures that the output samples are truncated to the 8-bit dynamic range limit
- Note that clamping does 'lose' information, since it truncates.

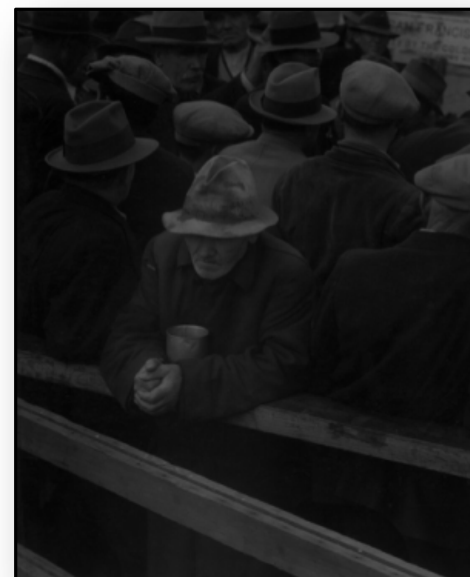
$$clamp(x, min, max) = \begin{cases} \text{min} & \text{if } \lfloor x \rfloor \leq \text{min}, \\ \text{max} & \text{if } \lfloor x \rfloor \geq \text{max}, \\ \lfloor x \rfloor & \text{otherwise.} \end{cases}$$

Rescaling Examples

gain = 1, bias = 55

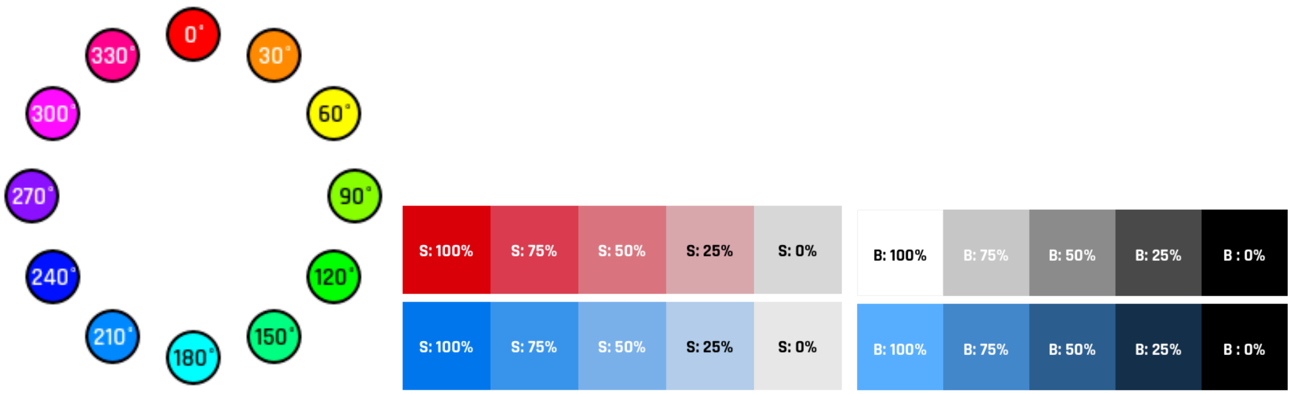
gain = 1, bias = -55

gain = 2, bias=0



gain = .5, bias=0

Rescaling Color Images

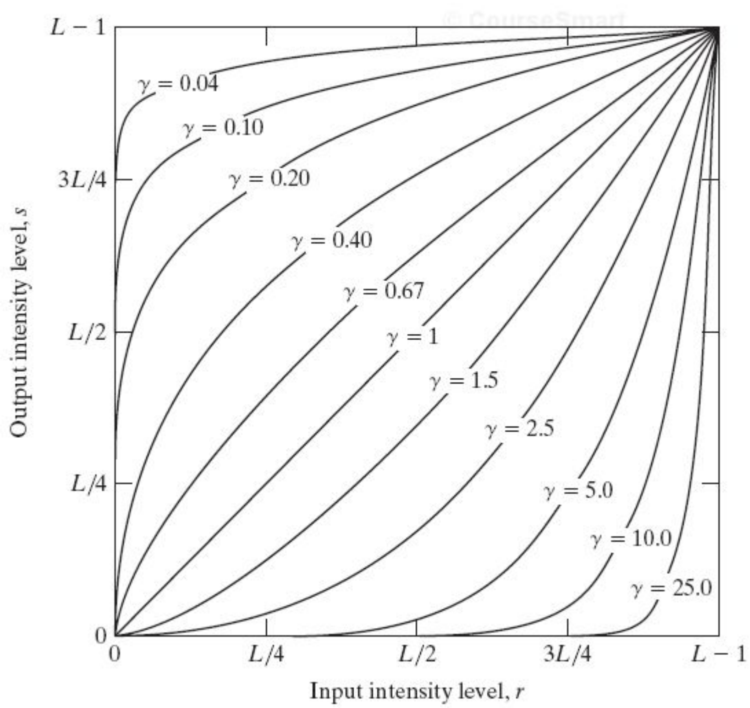
- Often, it is desirable to apply different gain and bias values to each channel of a color image separately
  - Example: A color image that utilizes the HSB (Hue-Saturation-Brightness) color model.
  - 

Credit "Learn Ui Design Blog"

  - Since all color information is contained in the H and S channels, it may be useful to adjust ONLY the brightness, encoded in channel B, without altering the color of the image in any way.
- Rescaling the channels of a color image in a non-uniform manner is also possible rescaling each color channel separately.

Example: Gamma Correction

$$s = r^{\gamma}$$



a b
c d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively. (Original image for this example courtesy of NASA.)

Putting it all together: Gain, Bias, and Gamma

- $C_{out} = (\boldsymbol{\alpha} C_{in} + \boldsymbol{\beta})^{\gamma}$
- $\boldsymbol{\alpha}$ is known as **gain** (exposure)
- $\boldsymbol{\beta}$ is known as **bias** (offset)
- $\boldsymbol{\gamma}$ maps to a non-linear curve (**gamma** correction)

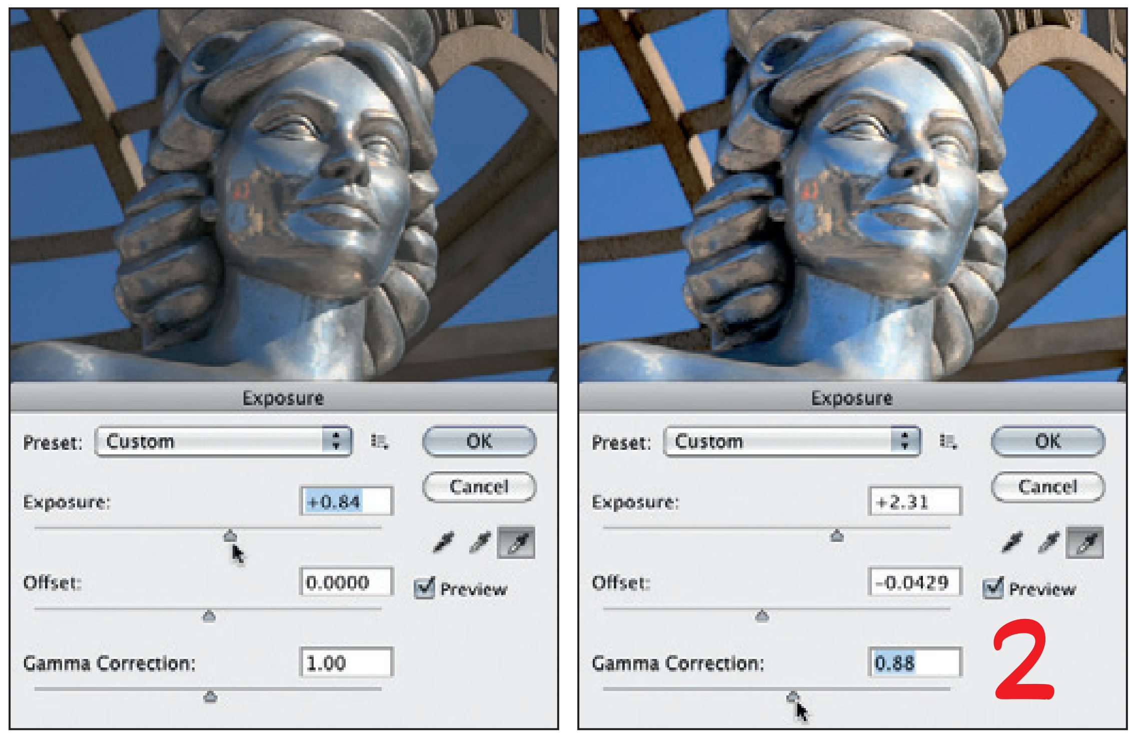
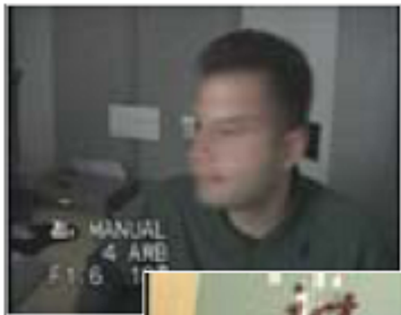


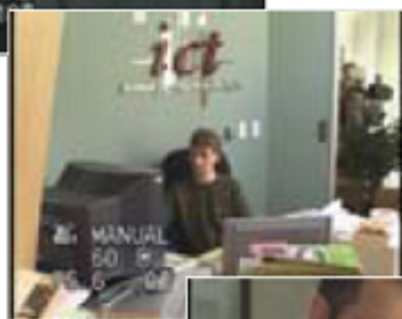
Image of Photoshop from Christian Bloch - The HDRI Handbook 2.0

Dynamic Range

The World is a High Dynamic Range (HDR)

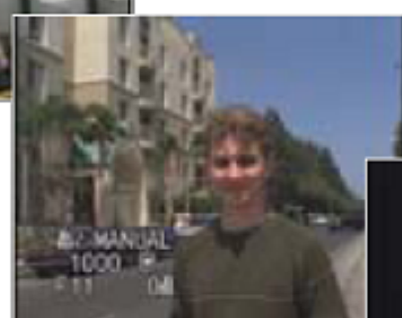


1:1



1:1,500

1:25,000

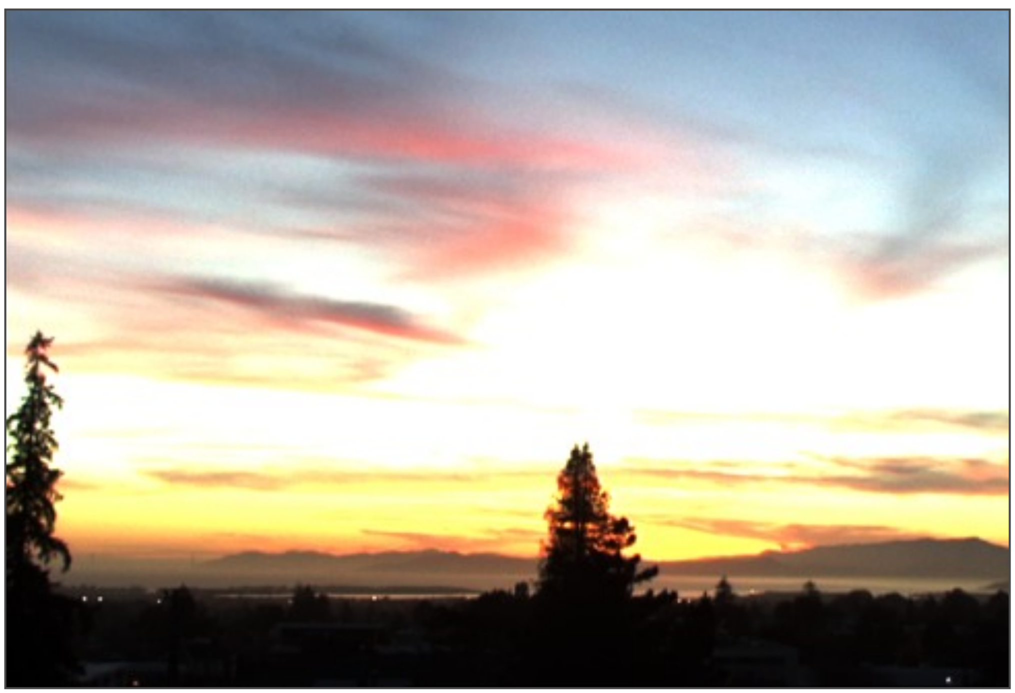


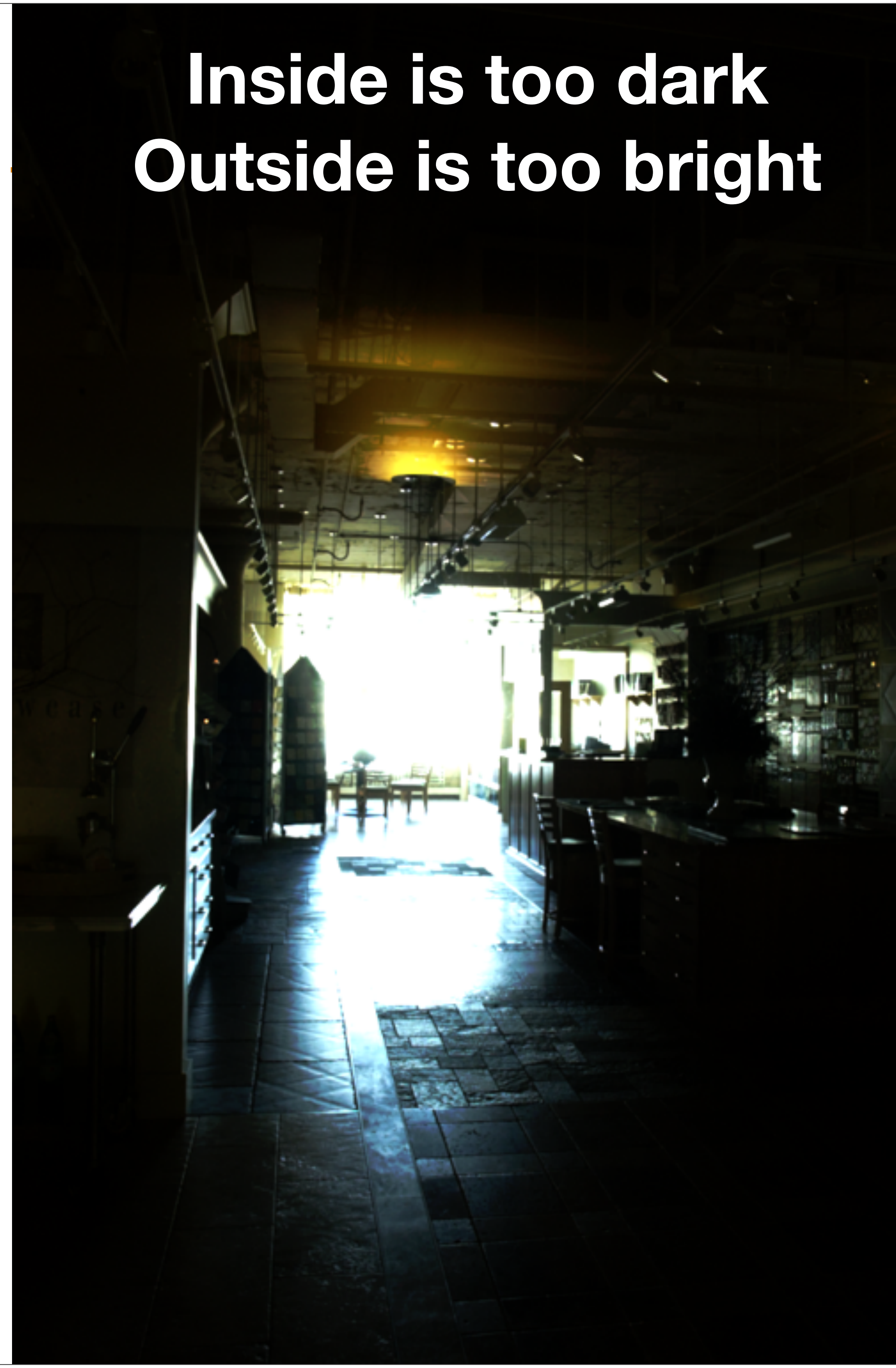
1:400,000

1:2,000,000,000

Examples

Sun overexposed
Foreground too dark



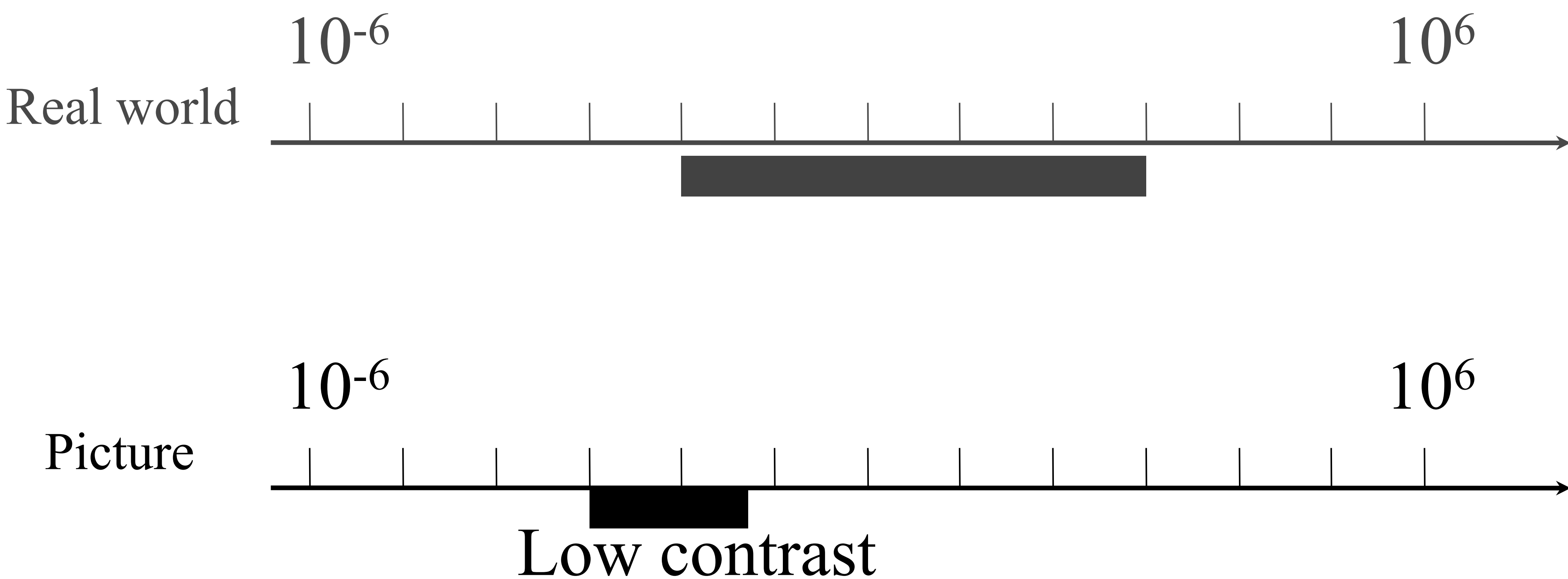


Dynamic Range in Displays?

- Range of pure black vs. pure white?

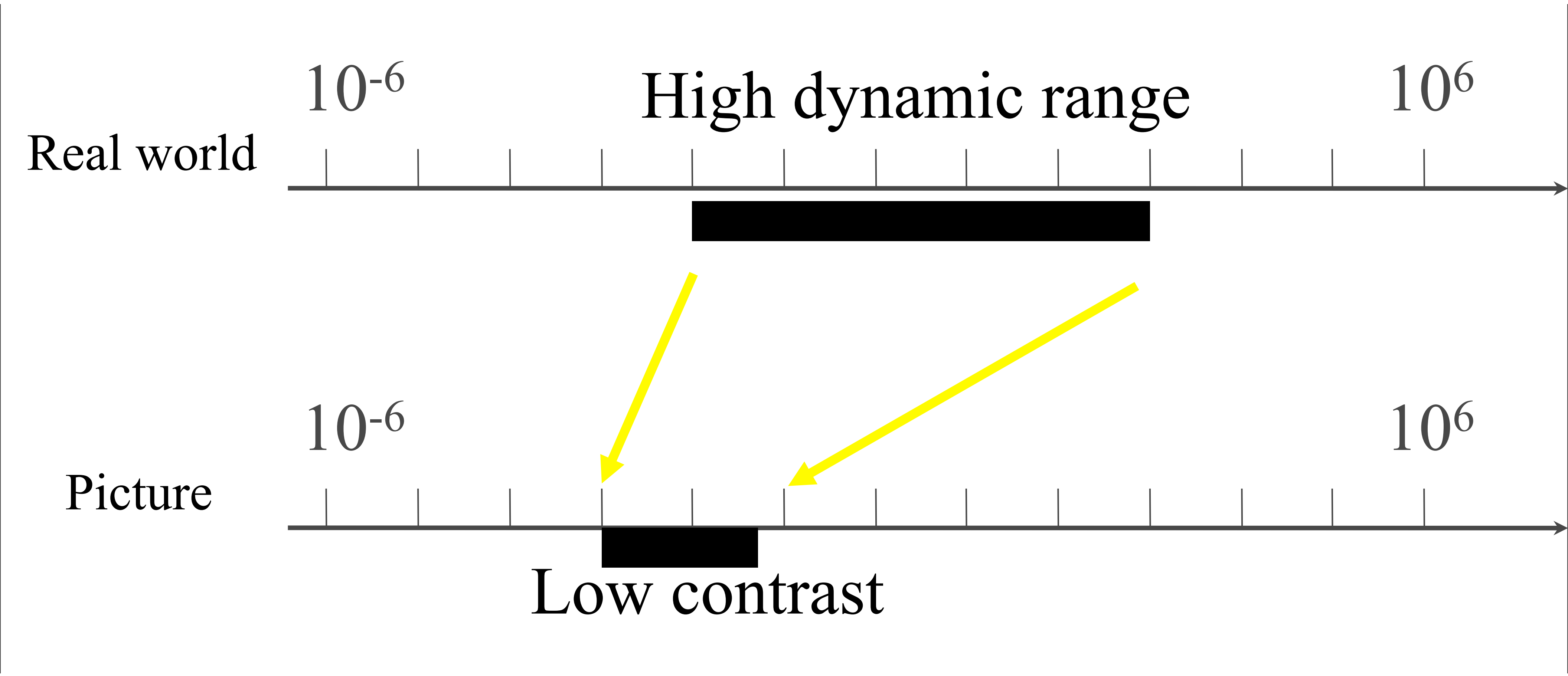
Dynamic Range in Displays?

- **Typically 1: 20 or 1:50**
  - Black is ~ 50x darker than white
- **Max 1:500**



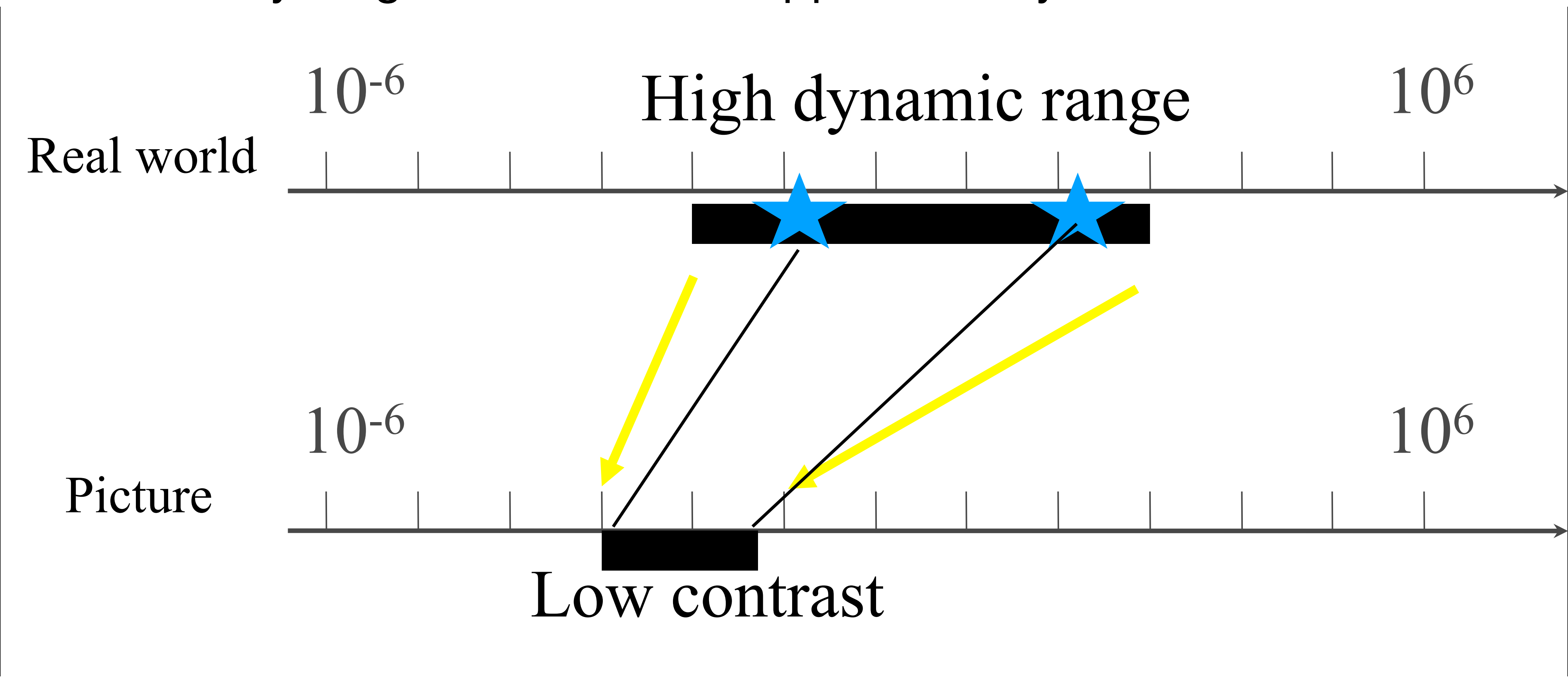
Problem: Displaying the Information

- Problem: How should we map scene radiances (up to 1:100,000) to display radiances (only around 1:100) to produce a satisfactory image?
- Goal: match limited contrast of the display medium while preserving details
- Solution: **Tone Mapping**



First solution: Linear mapping

- We will find the pixels with min and max intensity in the input image.
- Map them to the min and max intensities of the display
- Everything in between is mapped linearly.

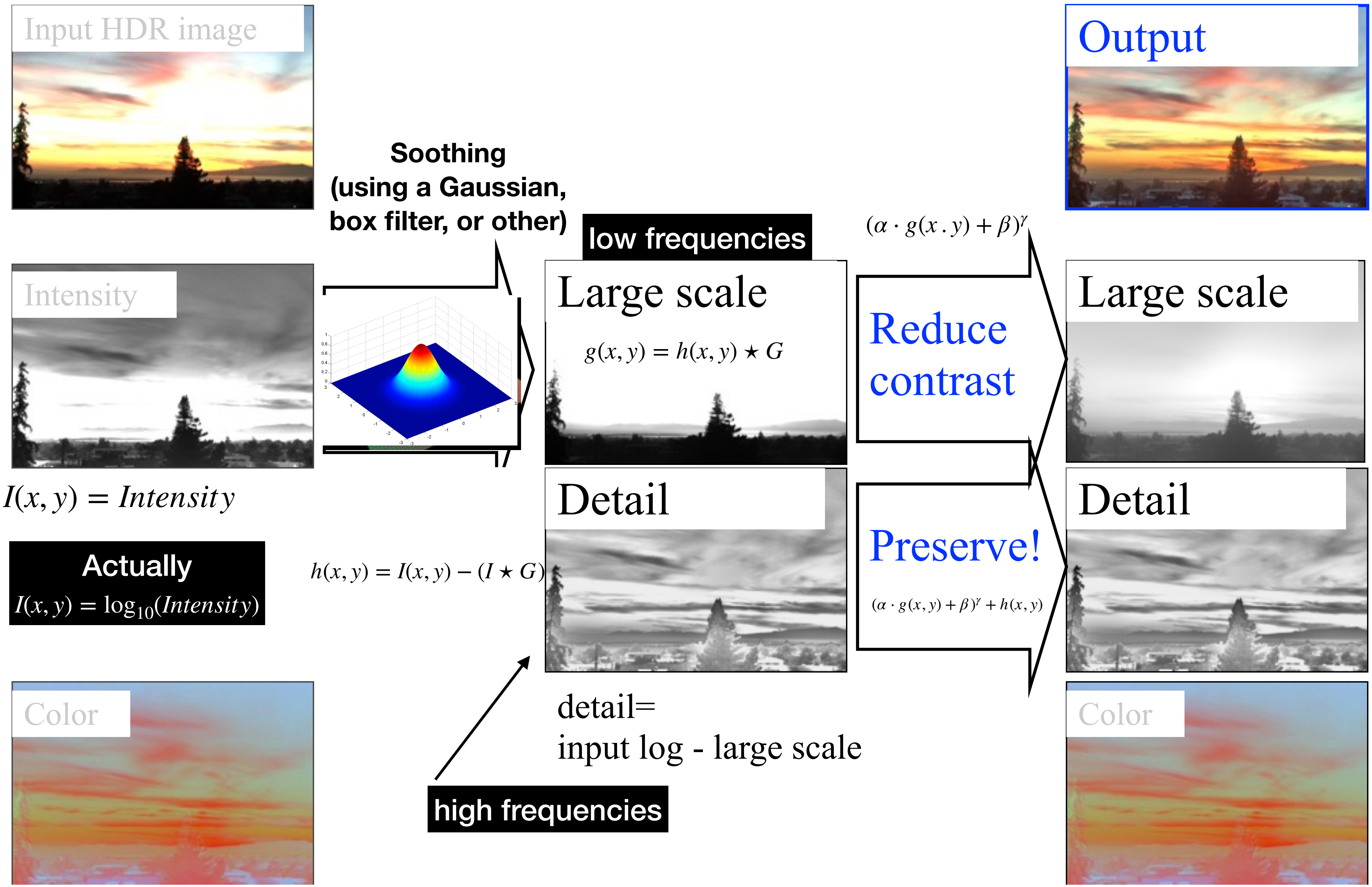


Without HDR + Tone Mapping

From Durand and Dorsey. No single global exposure can preserve both the colors of the sky and the details of the landscape, as shown on the rightmost images.

With HDR + Tone Mapping

Recap



Input HDR image
Intensity
$I(x, y) = Intensity$
Actually
$I(x, y) = \log_{10}(Intensity)$
Color
Soothing
(using a Gaussian,
box filter, or other)
low frequencies
Large scale
$g(x, y) = h(x, y) \star G$
$h(x, y) = I(x, y) - (I \star G)$
Detail
detail=
input log - large scale
high frequencies
$(\alpha \cdot g(x . y) + \beta)^{\gamma}$
Reduce contrast
Preserve!
$(\alpha \cdot g(x, y) + \beta)^{\gamma} + h(x, y)$
Output
Large scale
Detail
Color

Before

http://abduzeedo.com/20-beautiful-hdr-pictures

Not All Tone Mapping Produces Extreme Results, Sometimes Just Beautiful Ones

Check (recommended)

http://luminous-landscape.com/essays/hdr-plea.shtml

Tone Mapping

Question: But why do we need more than 100 levels of intensity (luminance) if in the input file we only have 256 values of intensities (RGB) ?

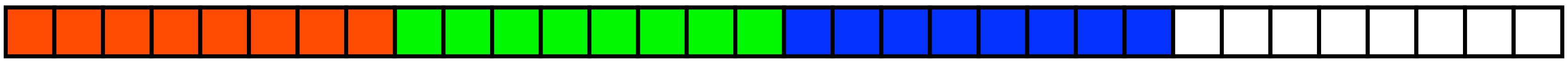
Answer: Not all file format has so few levels.

Even PPM could have 2 bytes per channel, so 256^2=65536 levels per channel.

Other formats gives much wider range:

Radiance RGBE Format (.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) =

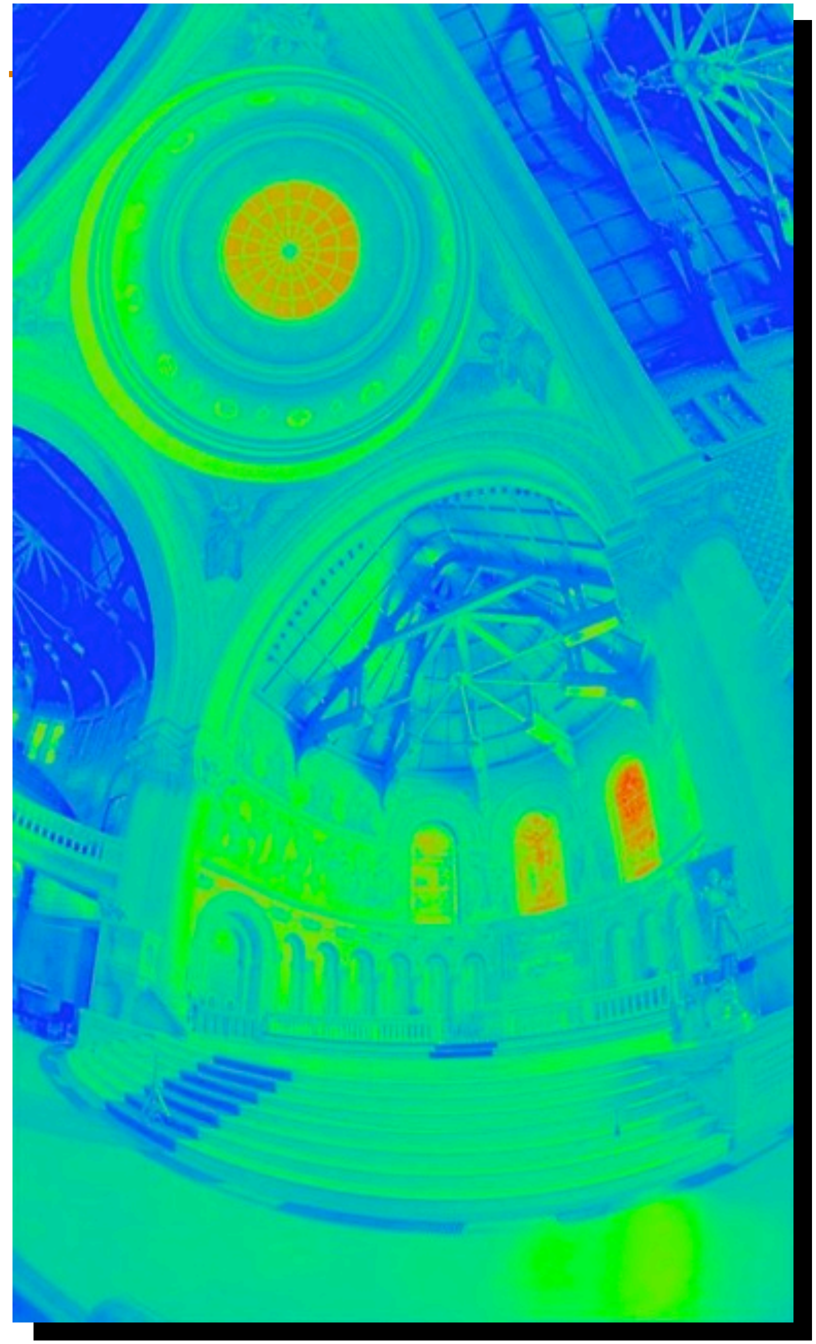
(145, 215, 87) * 2^(103-128) =

0.00000432 0.00000641 0.00000259

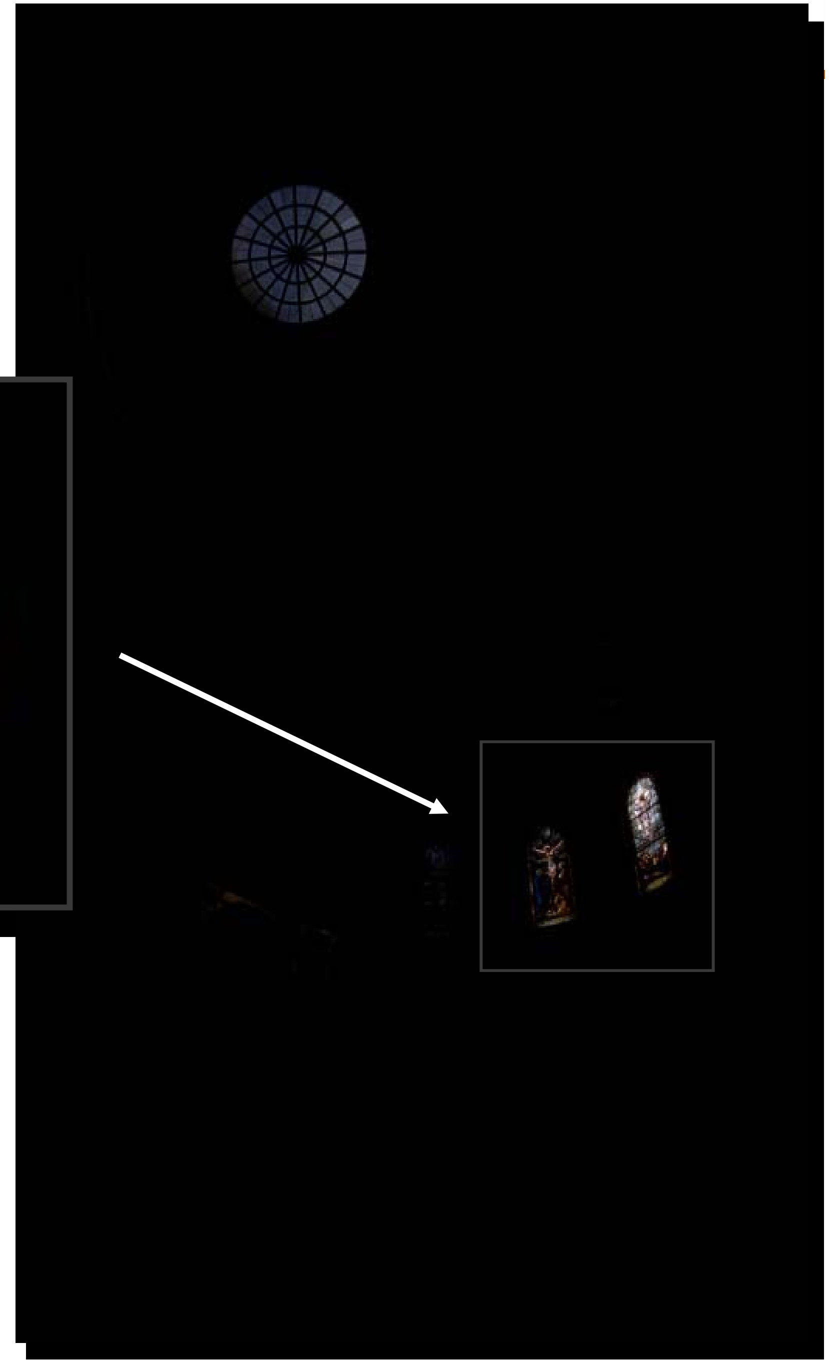
Ward, Greg. "Real Pixels," in Graphics Gems IV, edited by James Arvo, Academic Press, 1994

The Radiance Map

Radiance Definition: much of the power (in watts) is emitted by a cm^2 surface will be received by an optical system looking at that surface from a specified angle of view.

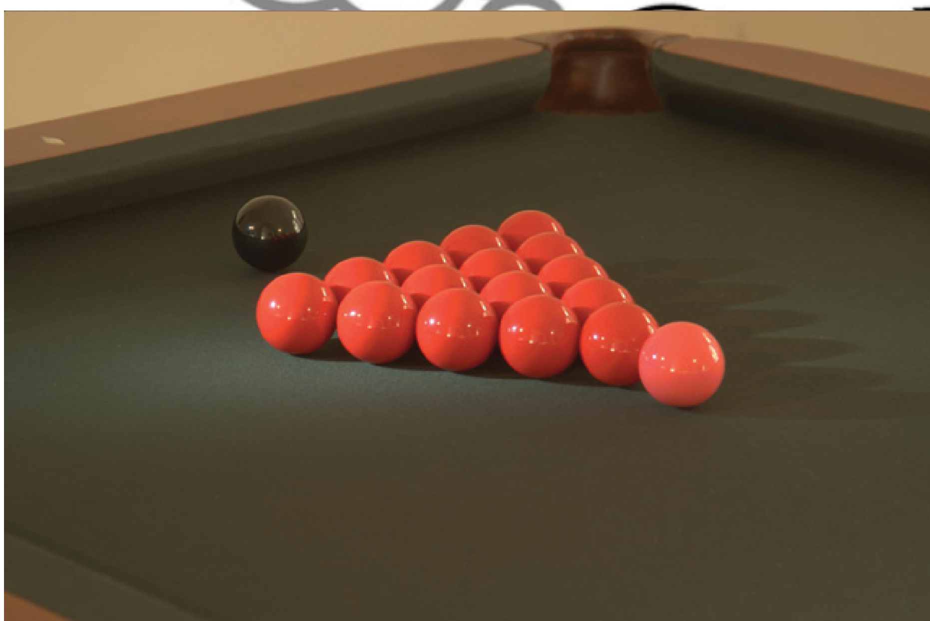


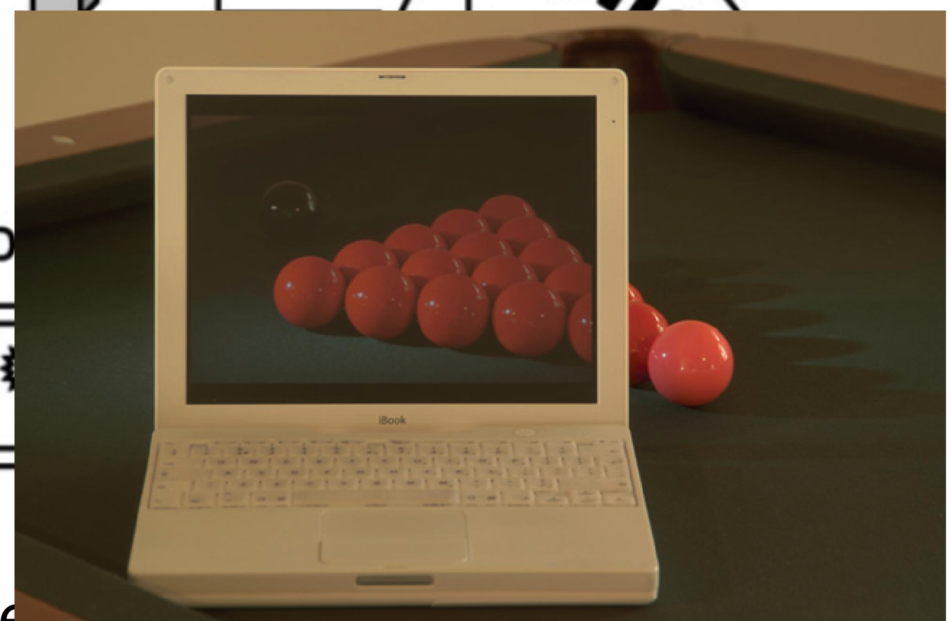
The Radiance Map



Linearly scaled to display device

Approach: Visual Matching

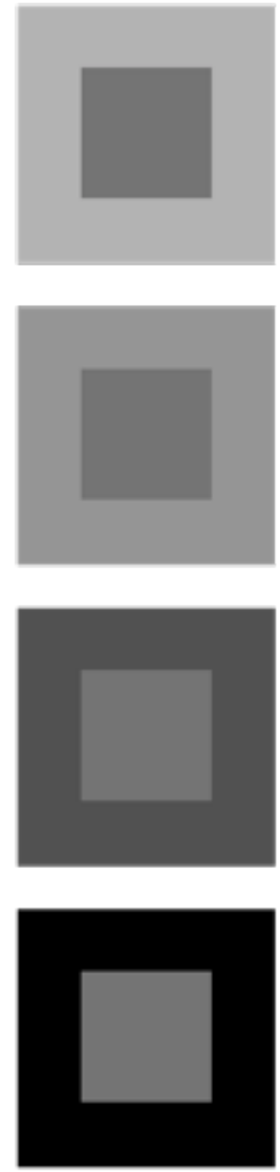




We do not need to reproduce the true radiance as long as it gives us a visual match.

Eyes and Dynamic Range

- We're sensitive to change (multiplicative)
  - A ratio of 1:2 is perceived as the same contrast as a ratio of 100 to 200
  - Use the log domain as much as possible
- But, eyes are **not** photometers
  - Dynamic adaptation (very local in retina)
  - Different sensitivity to spatial frequencies



Headlights are ON in both photos

Can we just scale? Maybe!

- For a color image, try to convert the input (world) luminance $\mathbf{L_w}$ to a target display luminance $\mathbf{L_d}$

- This type of scaling works (sometimes). In particular, it works best in the log and/or exponential domains

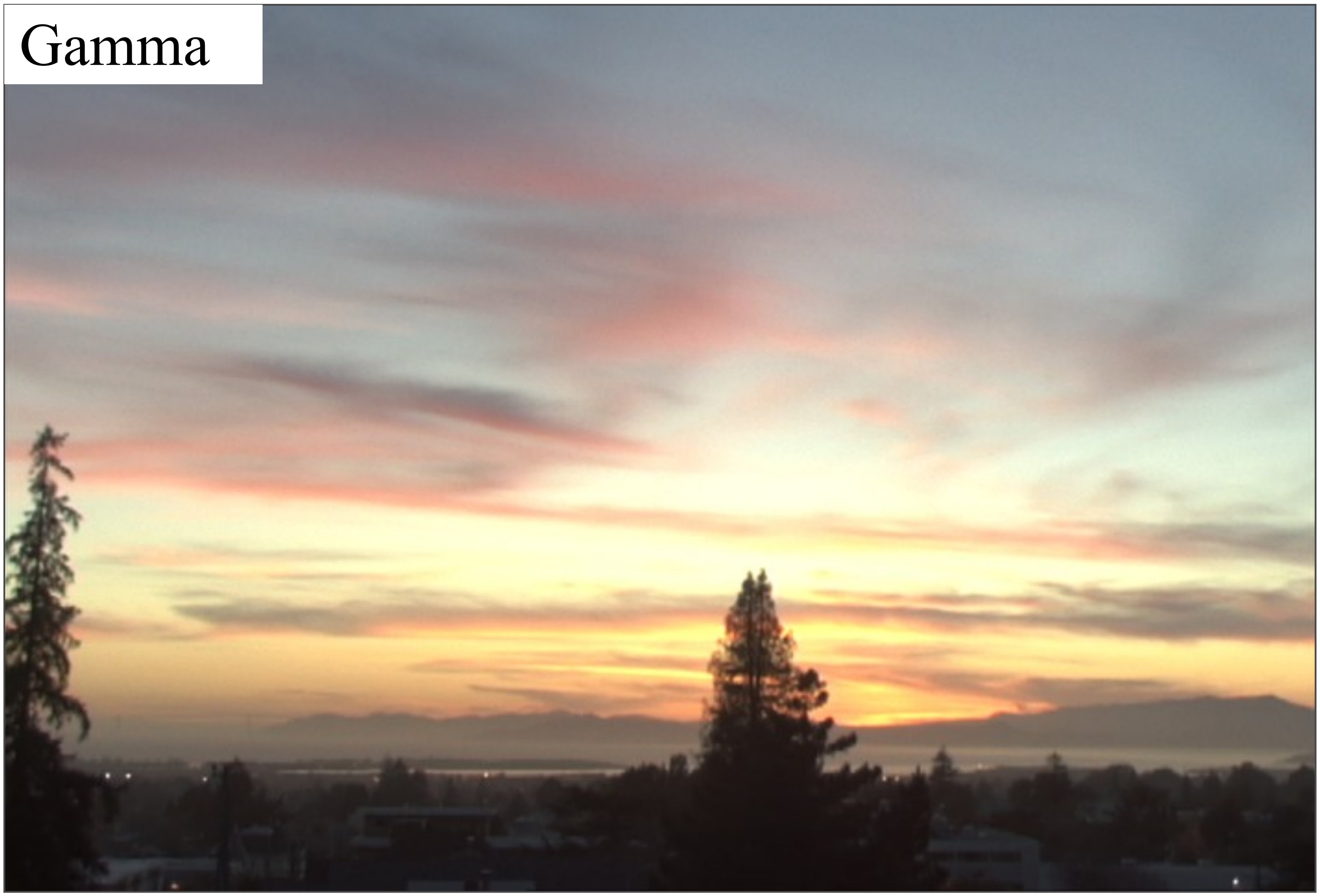
- $log_{10}(x)=1+log_{10}(y)$ means $x=10y$

- The base of the log is not important, as long as we are consistent in the mapping

$$\begin{bmatrix} R_d \\ G_d \\ B_d \end{bmatrix} = \begin{bmatrix} L_d \dfrac{R_w}{L_w} \\ L_d \dfrac{G_w}{L_w} \\ L_d \dfrac{B_w}{L_w} \end{bmatrix}$$

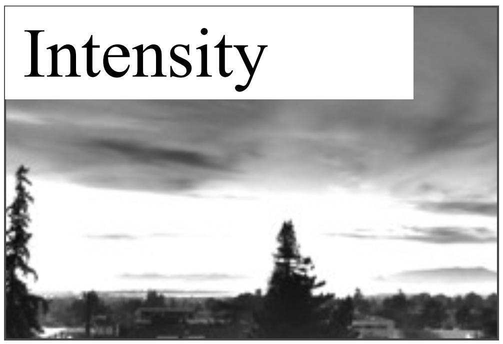
What scale value to use?
How about Gamma compression

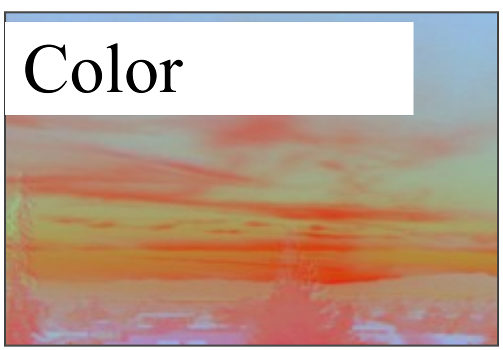
- $C_{out} = C_{in}^{\gamma}$, where $0 < \gamma < 1$ applied to each R,G,B channel
- Colors are washed out, why?

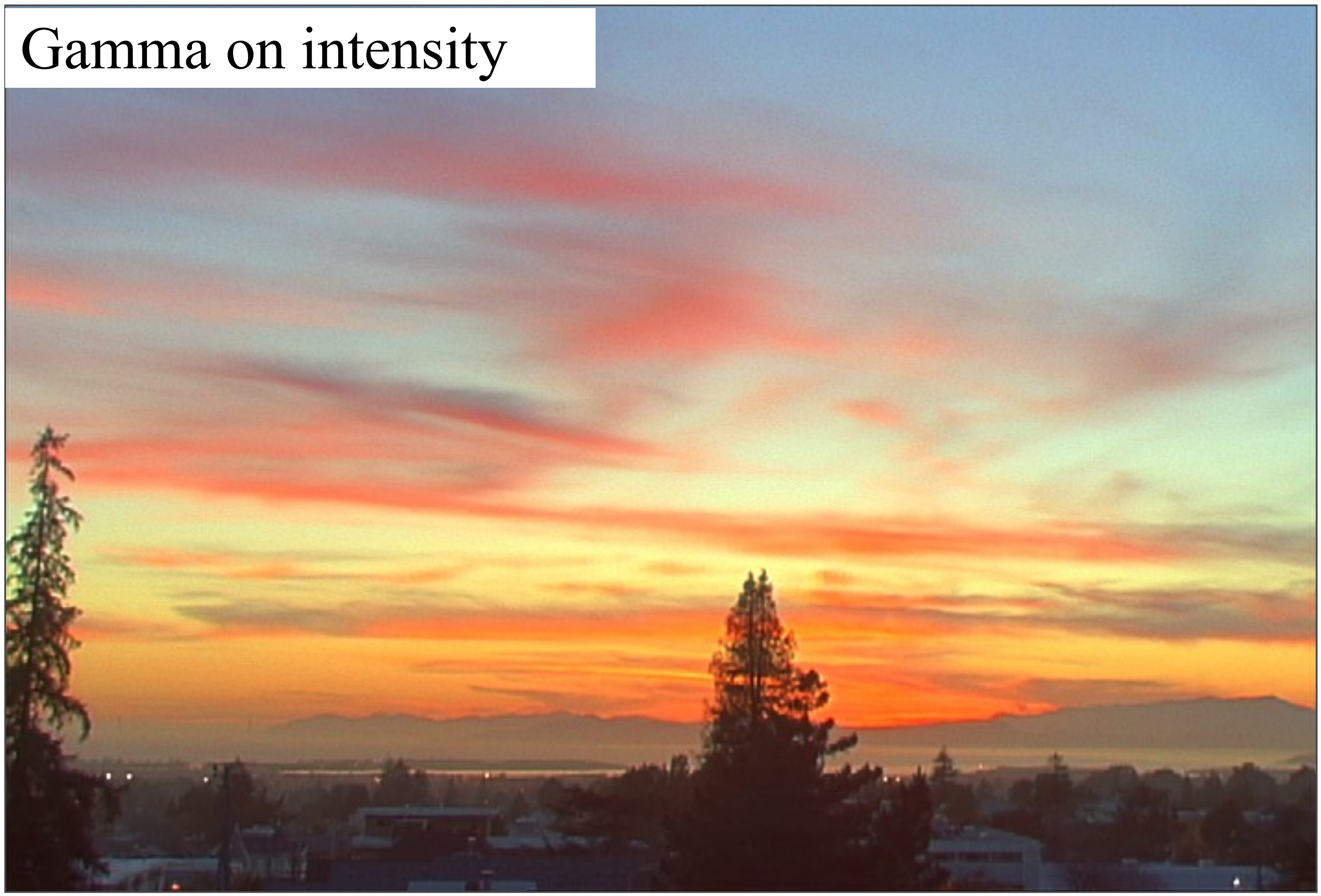


Gamma compression on Intensity

- Colors ok, but details in intensity are blurry

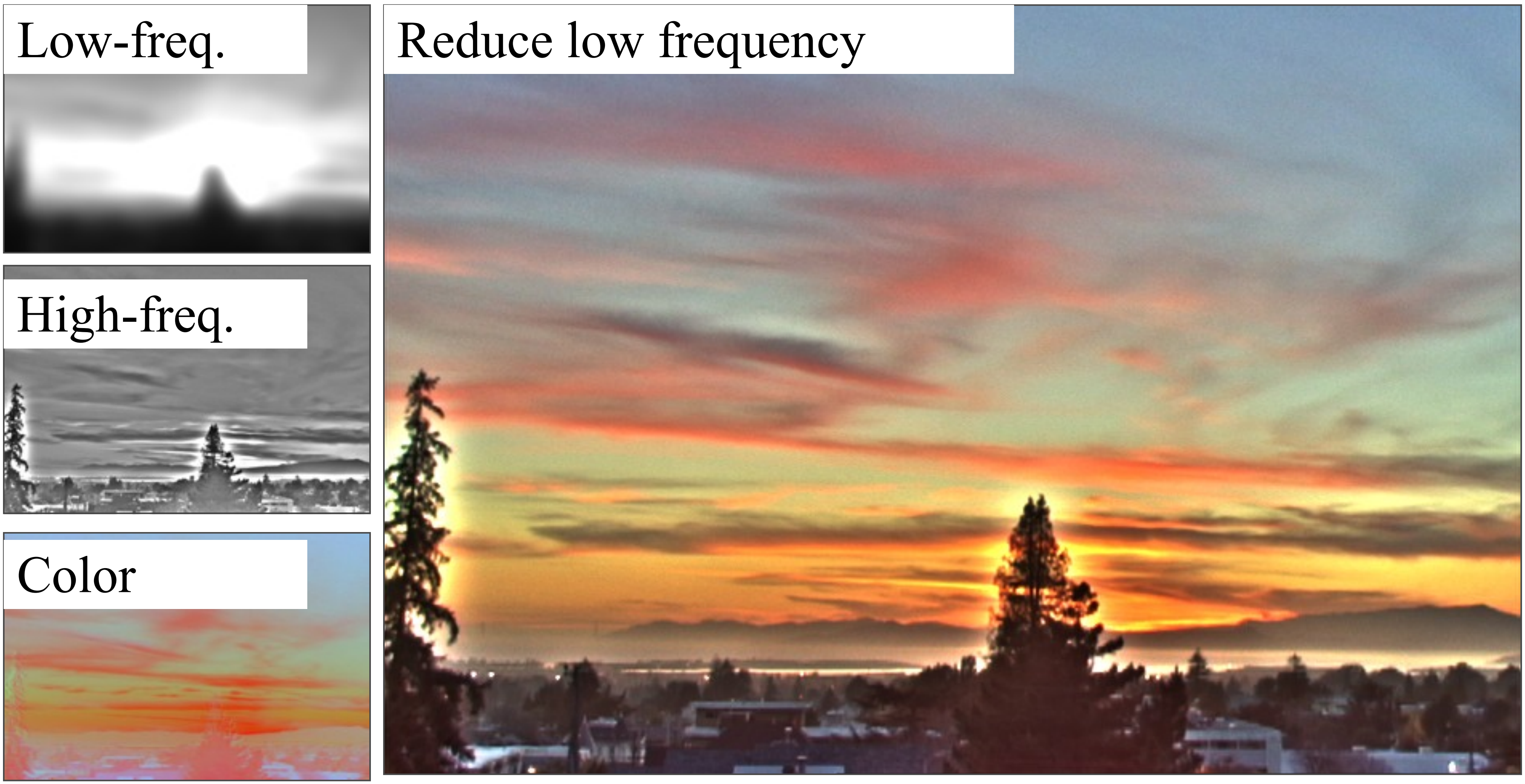






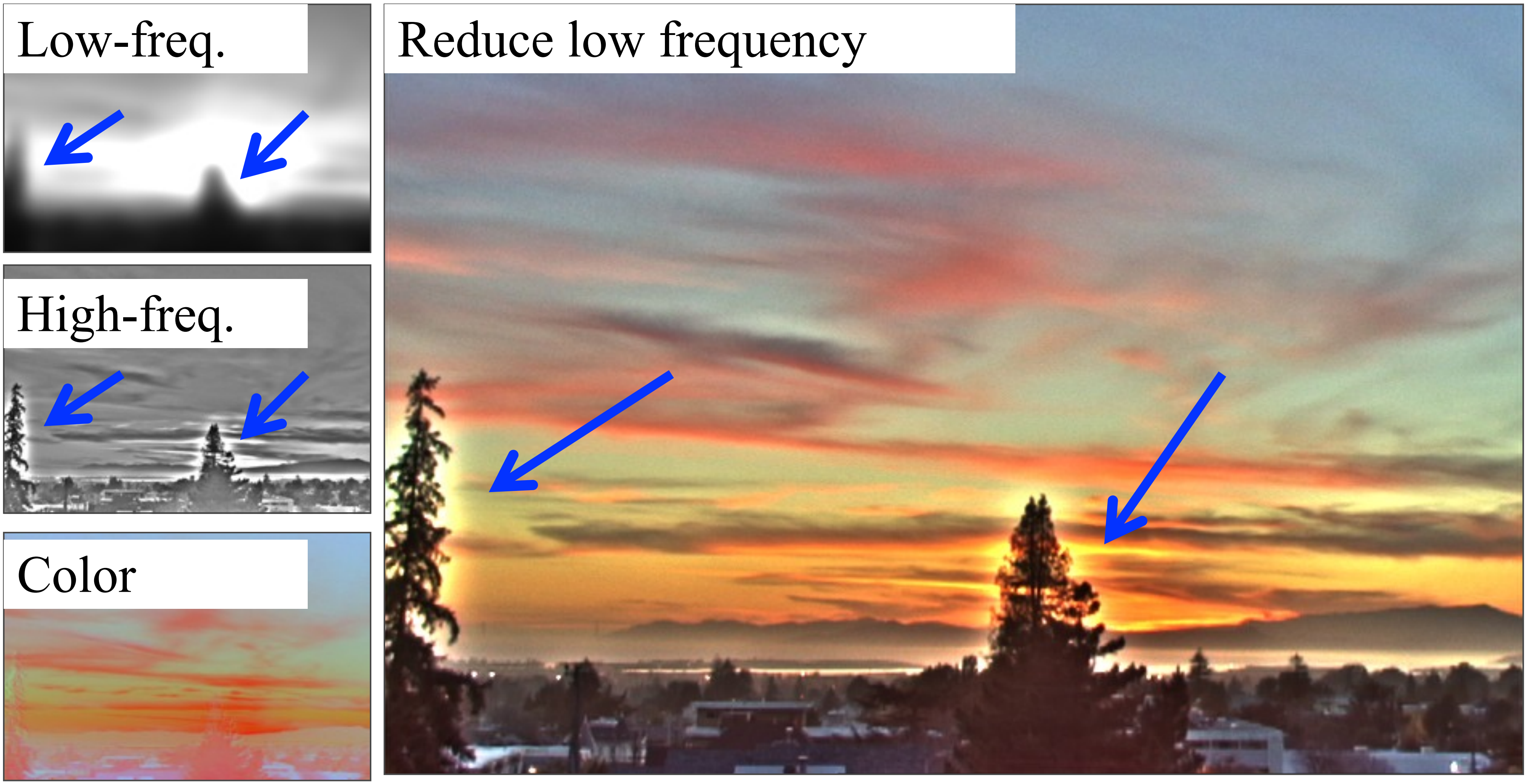
Oppenheim 1968, Chiu et al. 1993

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



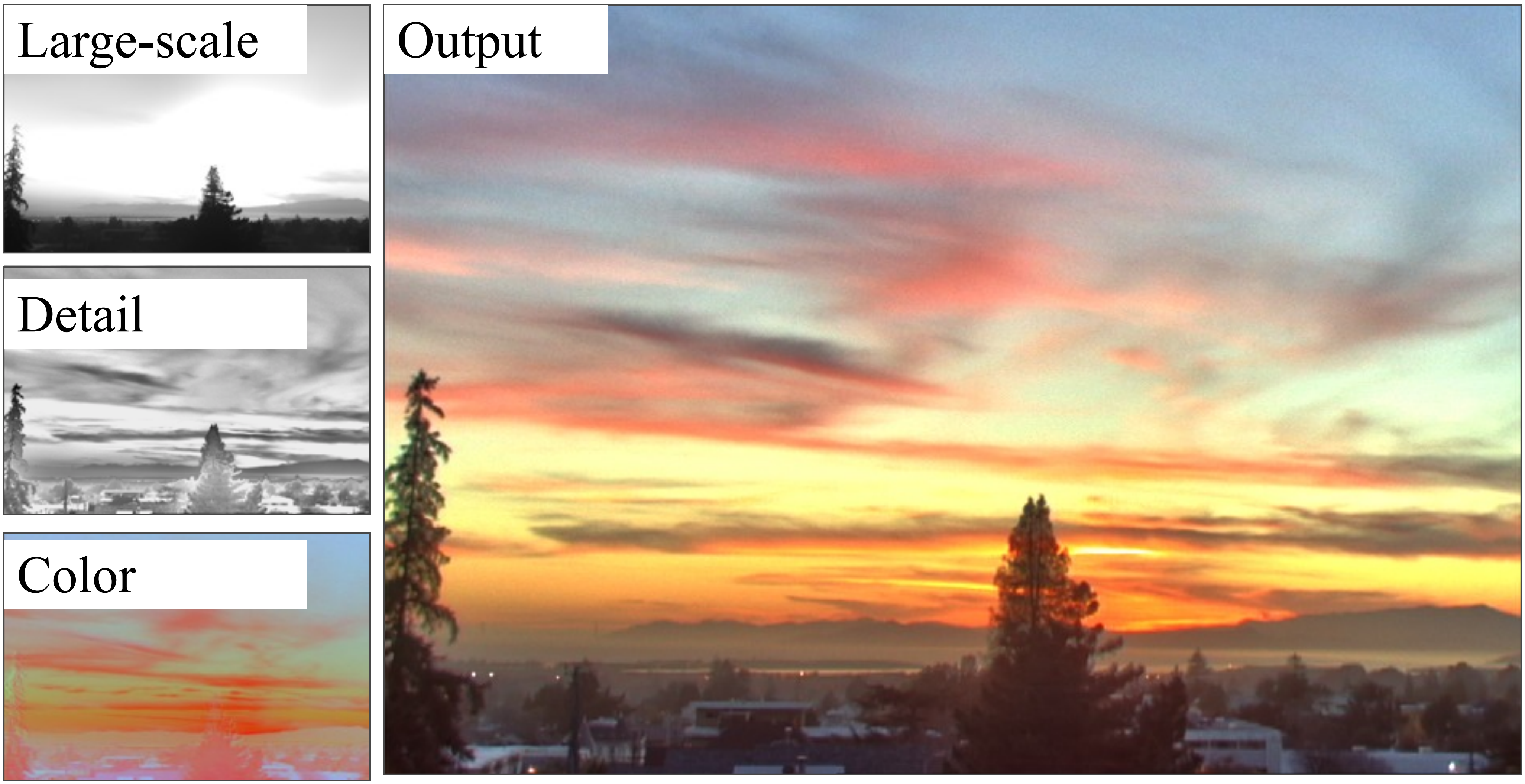
The halo nightmare

- **For strong edges**
- **Because they contain high frequency**



Our approach

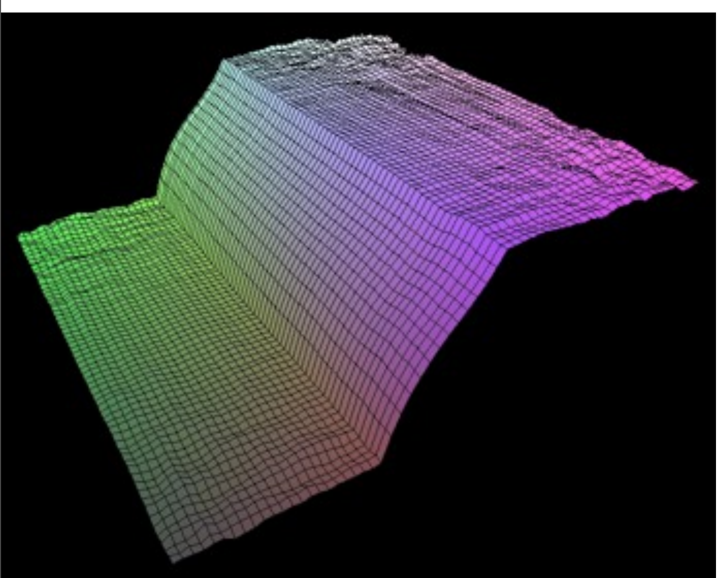
- **Do not blur across edges**
- **Non-linear filtering**

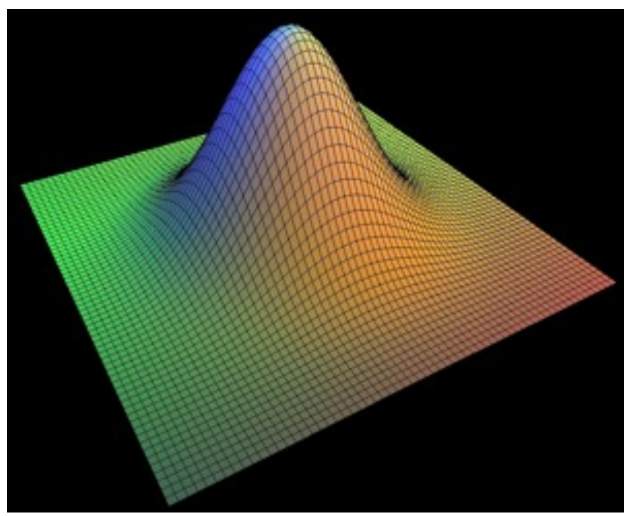


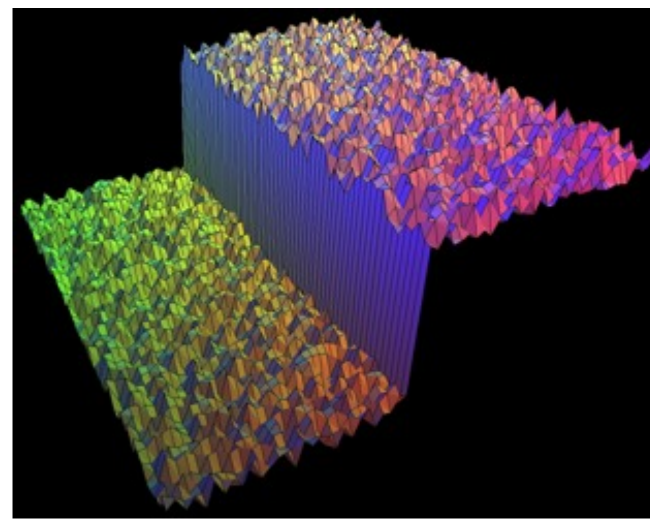
Start with Gaussian filtering

- **Here, input is a step function + noise**

$$J = f \otimes I$$





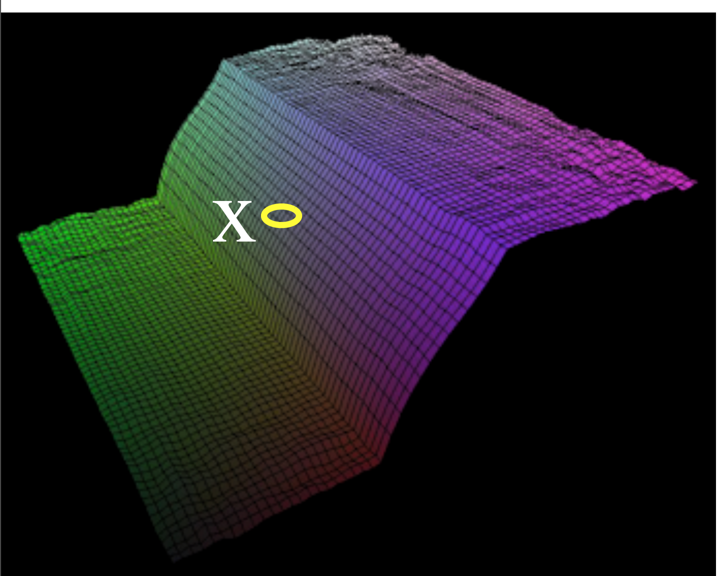


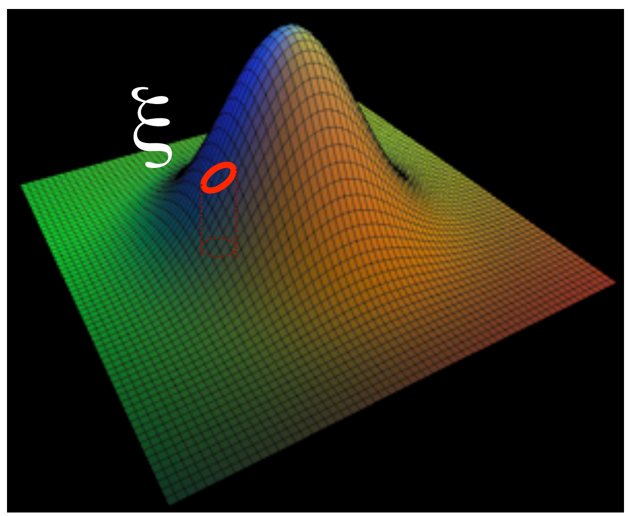
output ⟸ input

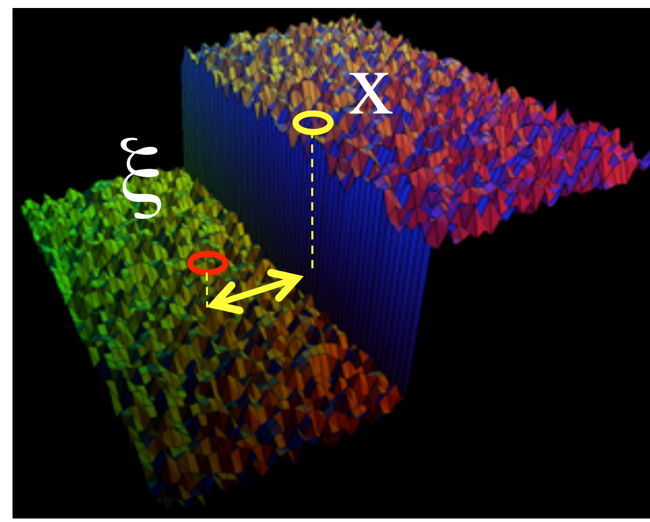
Gaussian filter as weighted average

- **Weight of ξ depends on distance to x**

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







output ⟵ input

The importance of convex combinations

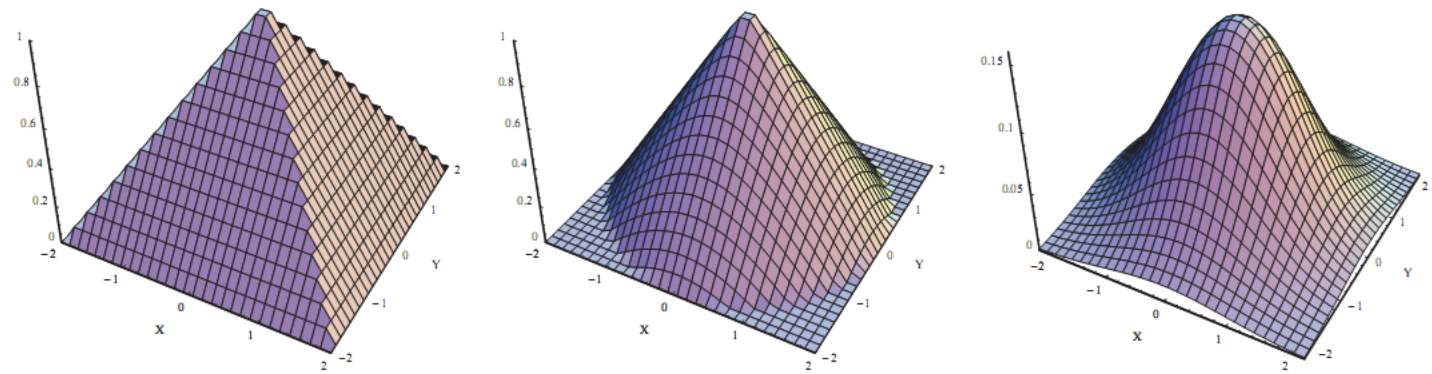
$$J(\mathbf{x}) = \sum \overbrace{f(\mathbf{x},\xi)}^{\text{weight function}} \underbrace{I(\xi)}_{\text{Intensity}}$$

x is the point where we need the answer
ξ is nearby point

When we smooth, or interpolate we usually use wighted average.

$$G(\mathbf{x}, \xi\,.\,x, \xi\,.\,y) = \frac{1}{2\pi\sigma} e^{-\frac{1}{2\sigma^2}(x-\xi.x)^2+(y-\xi.y)^2}$$

Which functions could f(x,ξ) be -



(a) Pyramid. (b) Cone. (c) Gaussian.

$$f(\mathbf{x},\ \xi) = \max\left\{0, \frac{3}{\alpha^2} - \frac{3}{\alpha^3}\max(|x-\xi\,.\,x|, |y-\xi\,.\,y|)\right\}$$

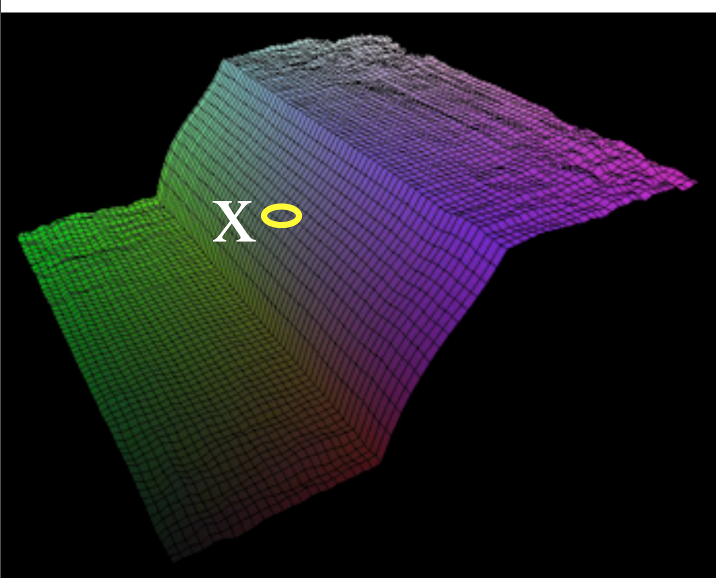
α is the **width** of the base of the pyramid. So in Fig(a), α=4

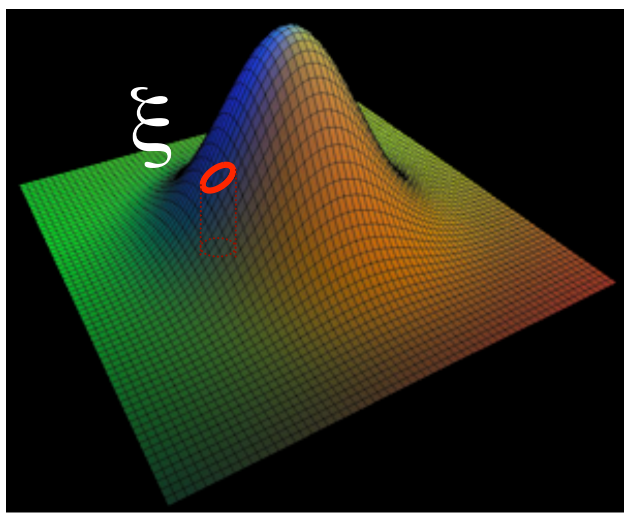
We will try to make sure that sum of weights =1 (this is called **convex combination**)
See example of bilinear interpolation on the whiteboard

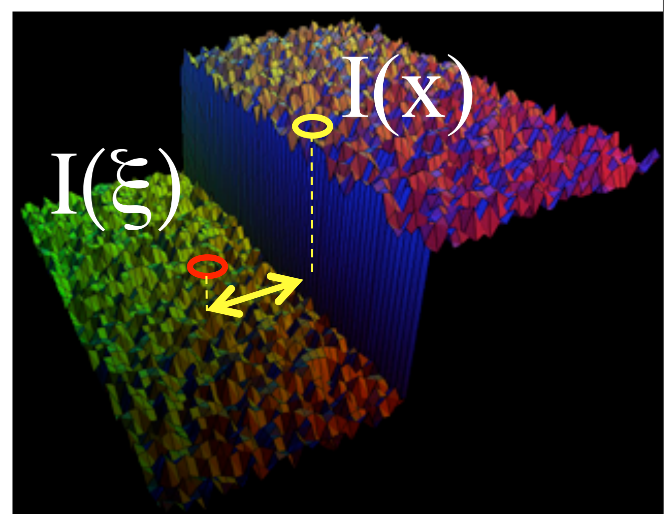
The problem of edges

- **Here, $I(\xi)$ "pollutes" our estimate J(x)**
- **It is too different**
  - To correct it, we will have to change the averaging.
  - Remember that during the smoothing, we will first sum $\sum I(\xi)$ for all point ξ near ***x.***
  - To resolve the **halo problem**, We will avoid summing $I(\xi)$ if $I(\xi)$ is very far from $I(x)$

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







output ⟵ input

Principle of Bilateral filtering

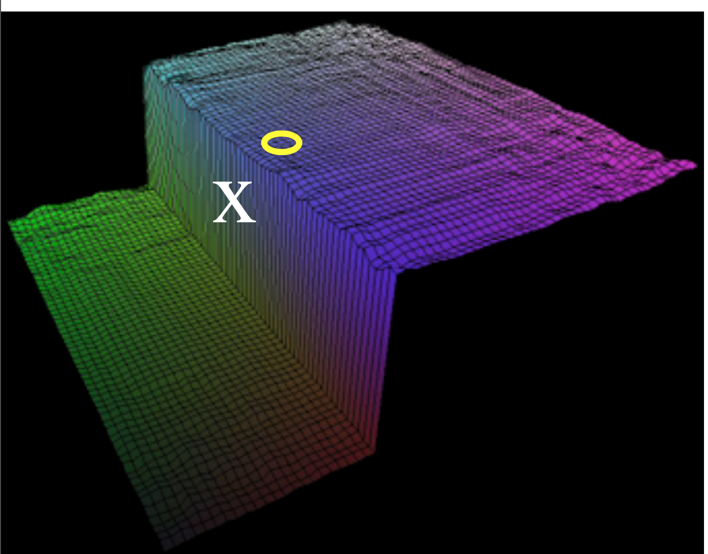
[Tomasi and Manduchi 1998]

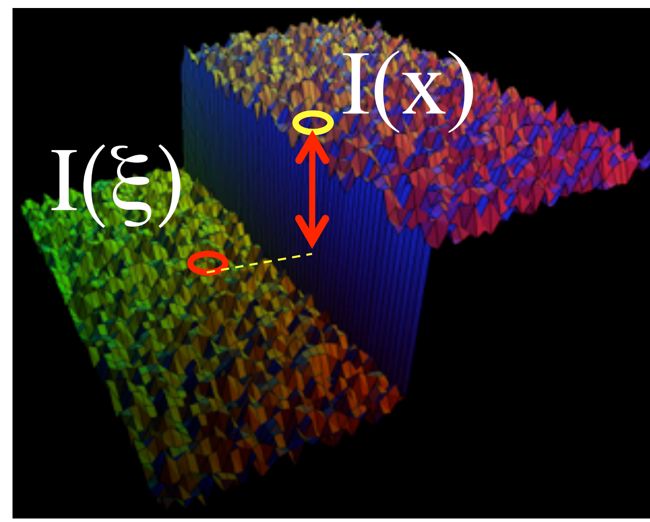
- **Penalty g on the intensity difference**

Remember that the sum of weights must be 1.
What to do if we skip some terms ?
We will divide the total sum by k(x)

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

$g(|a-b|) = 1$ if a is close to b, and zero otherwise





output

input

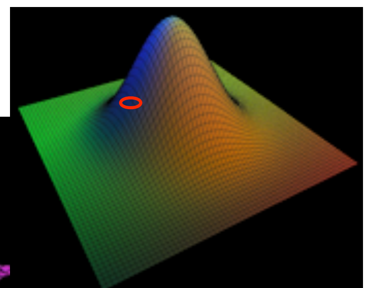
Bilateral filtering

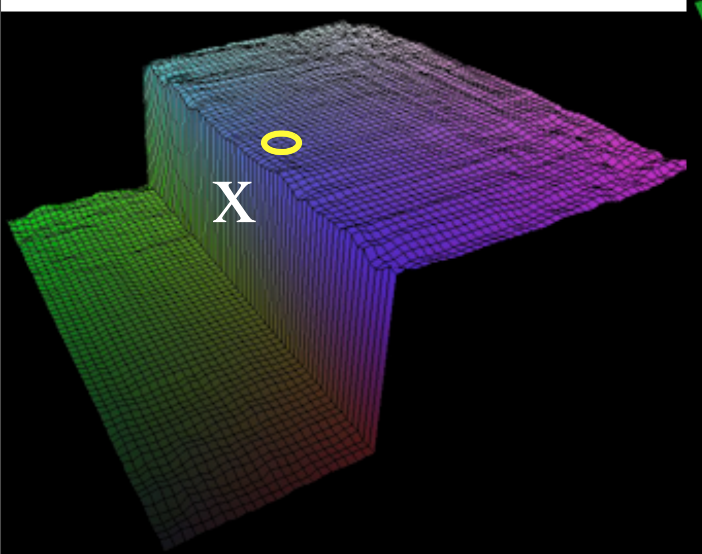
[Tomasi and Manduchi 1998]

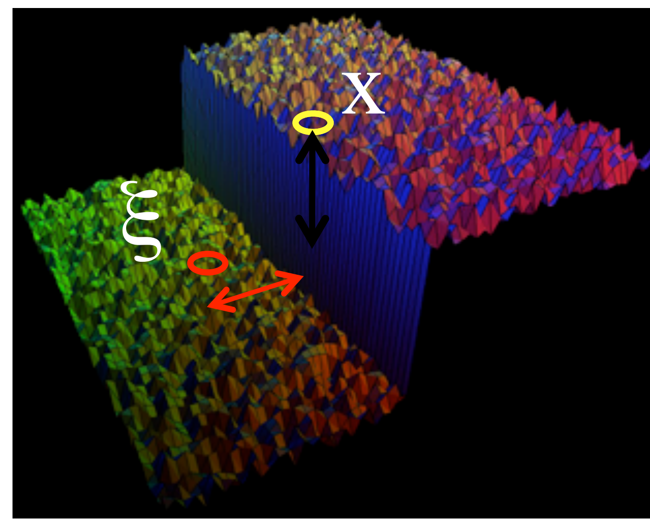
- **Spatial Gaussian f**

Remember that the sum of weights must be 1.
What to do if we skip some terms ?
We will divide the total sum by k(x)

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







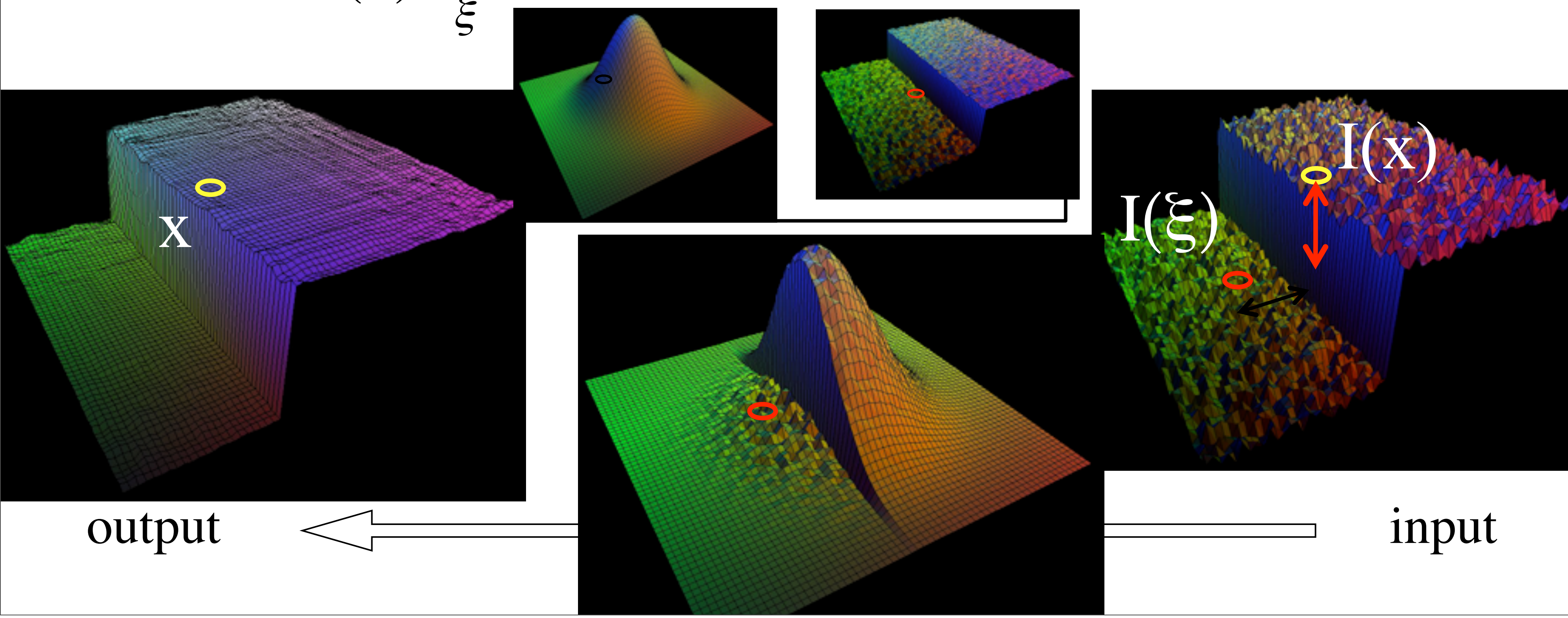
output ⟵ input

Bilateral filtering

[Tomasi and Manduchi 1998]

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

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

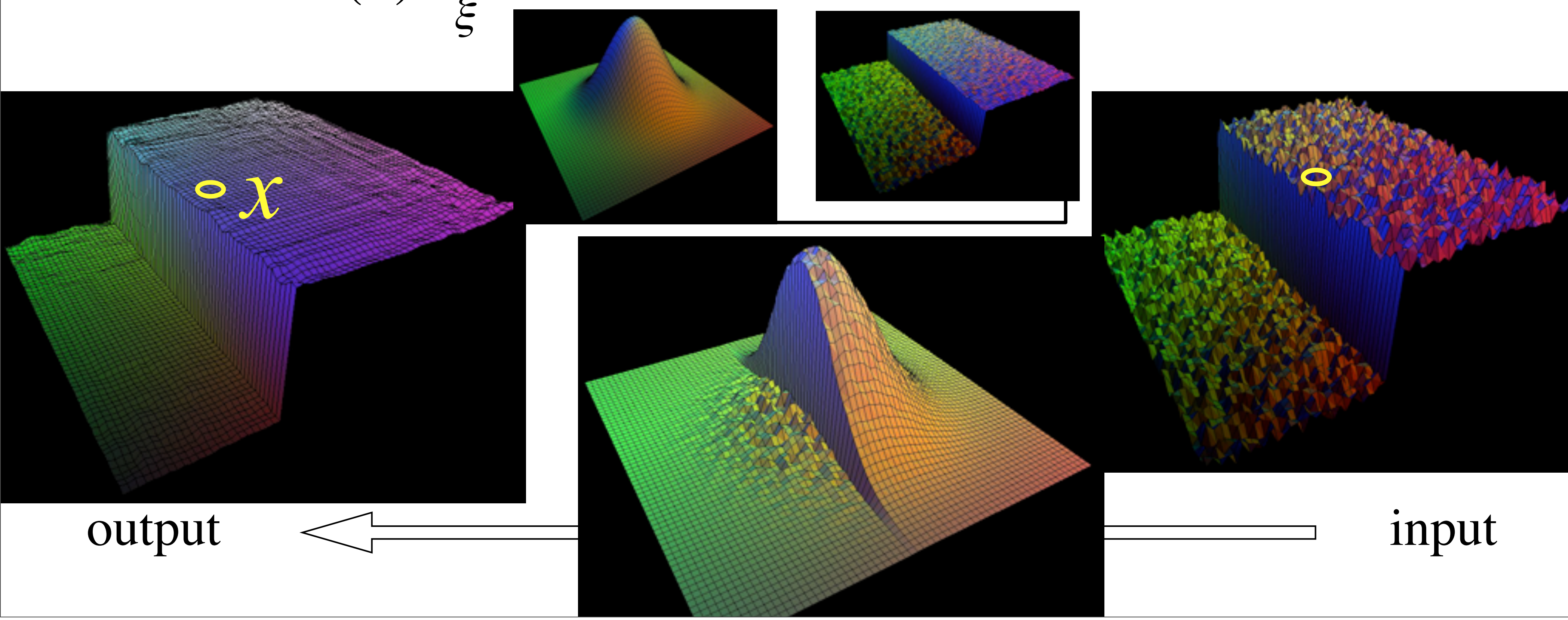


Normalization factor

[Tomasi and Manduchi 1998]

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

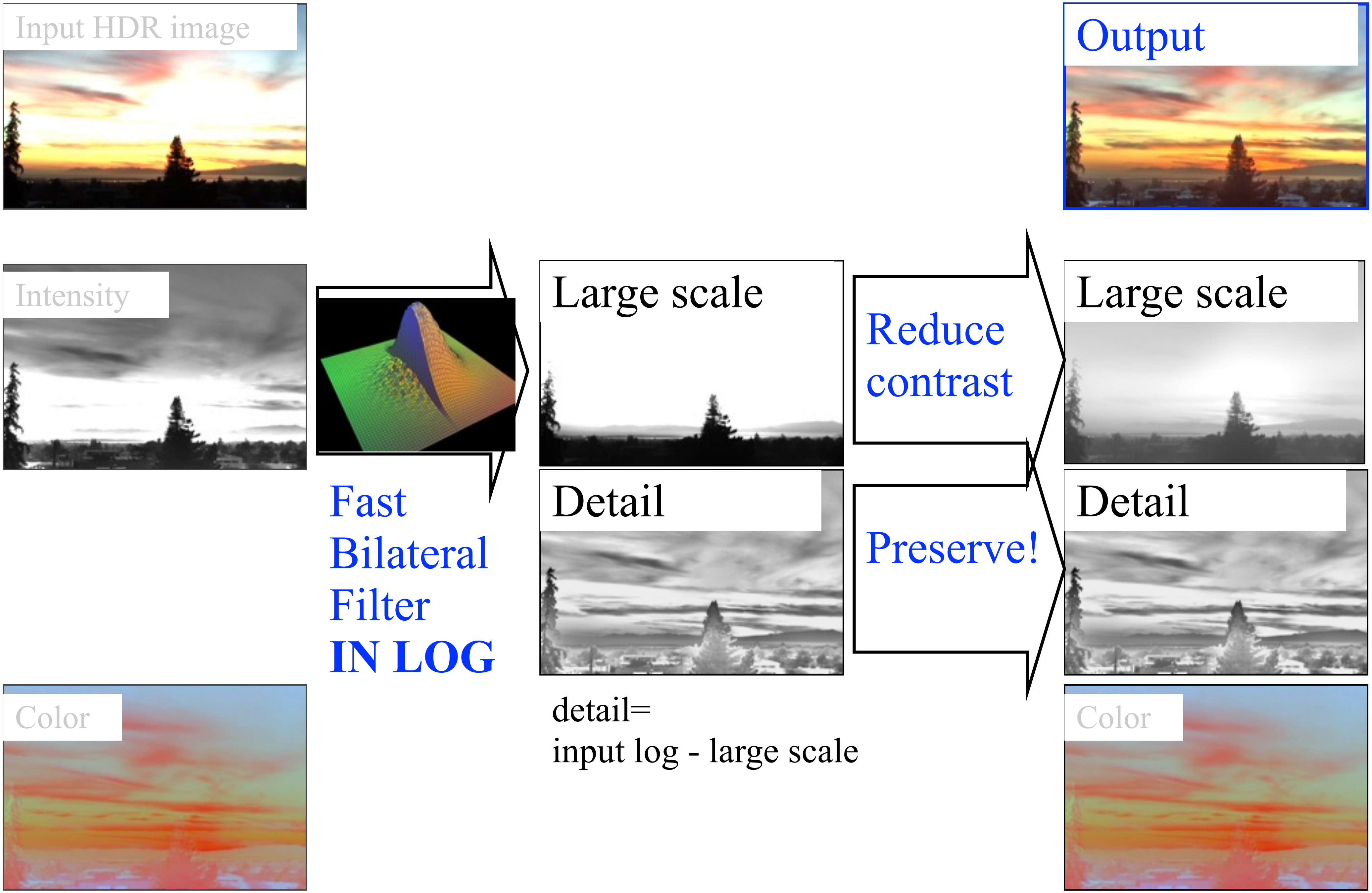
$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x,\xi) \quad g(I(\xi)-I(x)) \quad I(\xi)$$



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

Recap


Input HDR image
Output
Intensity
Large scale
Reduce contrast
Large scale
Fast Bilateral Filter IN LOG
Detail
Preserve!
Detail
detail= input log - large scale
Color
Color