

# Lecture 13: High Dynamic Range Imaging and Tone Mapping

---

**Yuhao Zhu**

<http://yuhaozhu.com>

**CSC 292/572, Fall 2022**  
**Mobile Visual Computing**

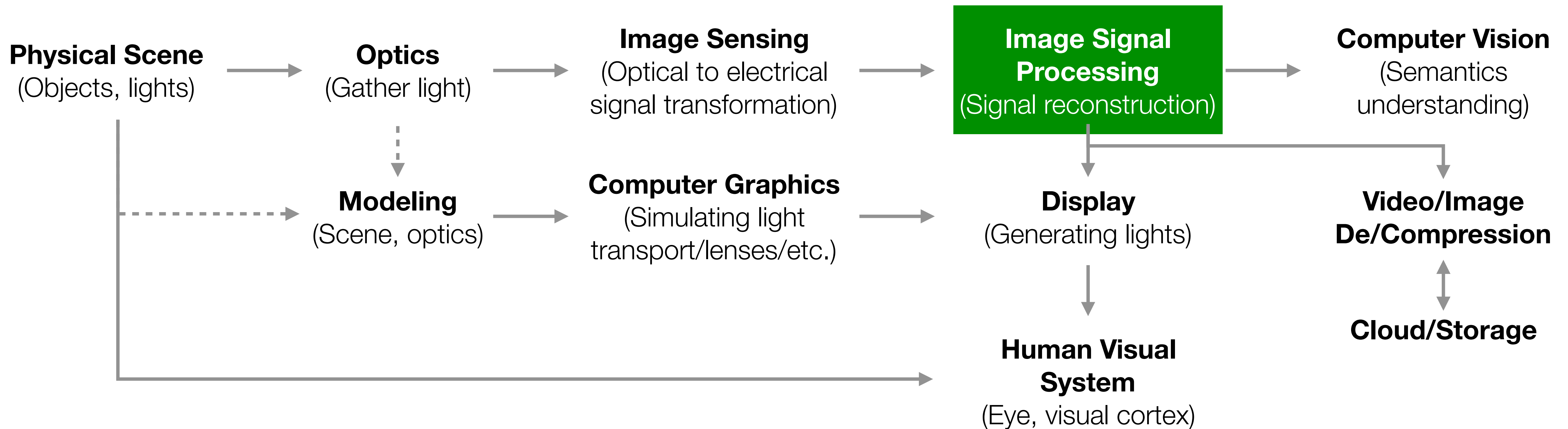


# Logistics

WA3 grades posted.

Will post WA4 and PA2 soon. Both are on camera imaging.

# Where Are We



# The Roadmap

Theoretical Preliminaries

Human Visual Systems

Color in Nature, Arts, Tech  
(a.k.a., the birth, life, and death of light)

**Digital Camera Imaging**

Modeling and Rendering

Applications



Optics in Camera

Image Sensor

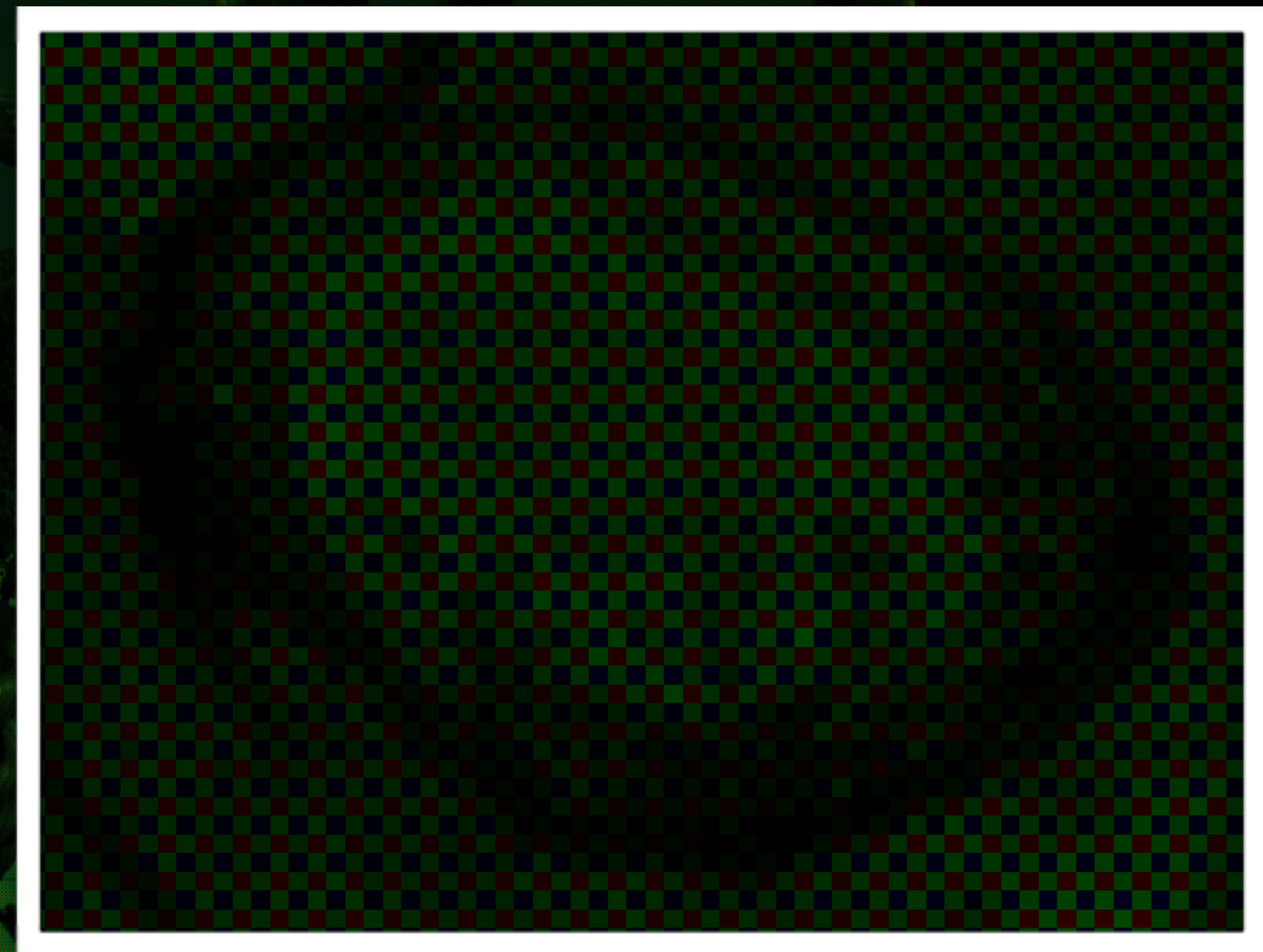
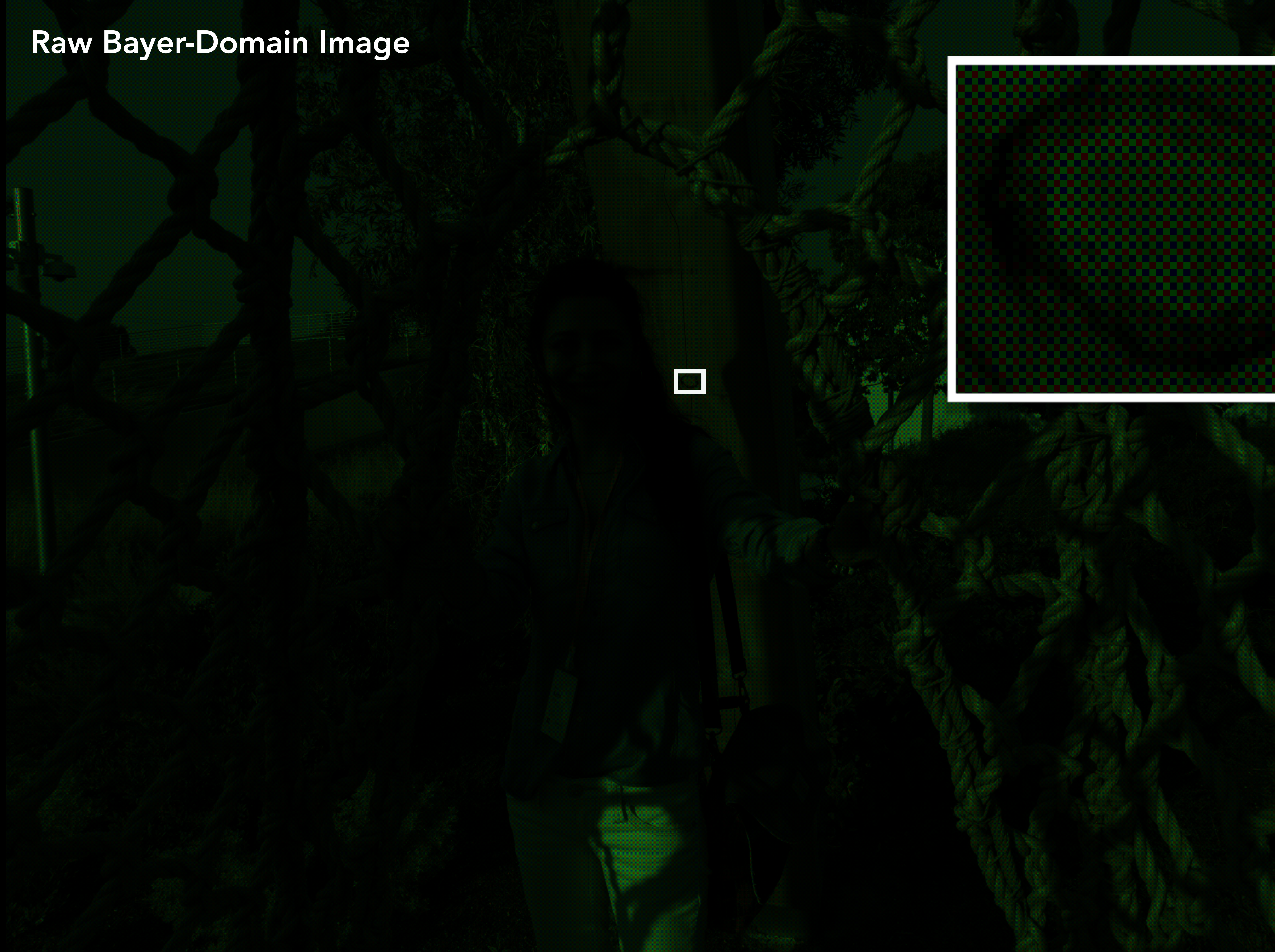
**Image Signal Processing**

Image/Video Compression

Immersive Content



Raw Bayer-Domain Image





Demosaic-ed Image





With Color Correction





With Simple Tone Mapping





# What Pixel Photo Generates





Raw Bayer-Domain Image



Demosaiced Image





With Color Correction



With Simple Tone Mapping





What Pixel Phone Generates

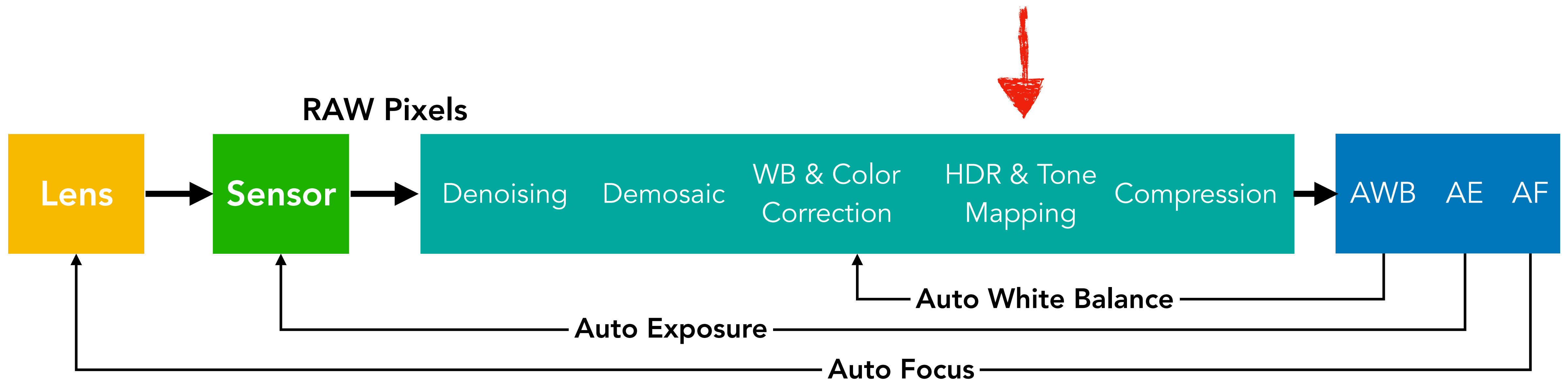


With Simple Tone Mapping





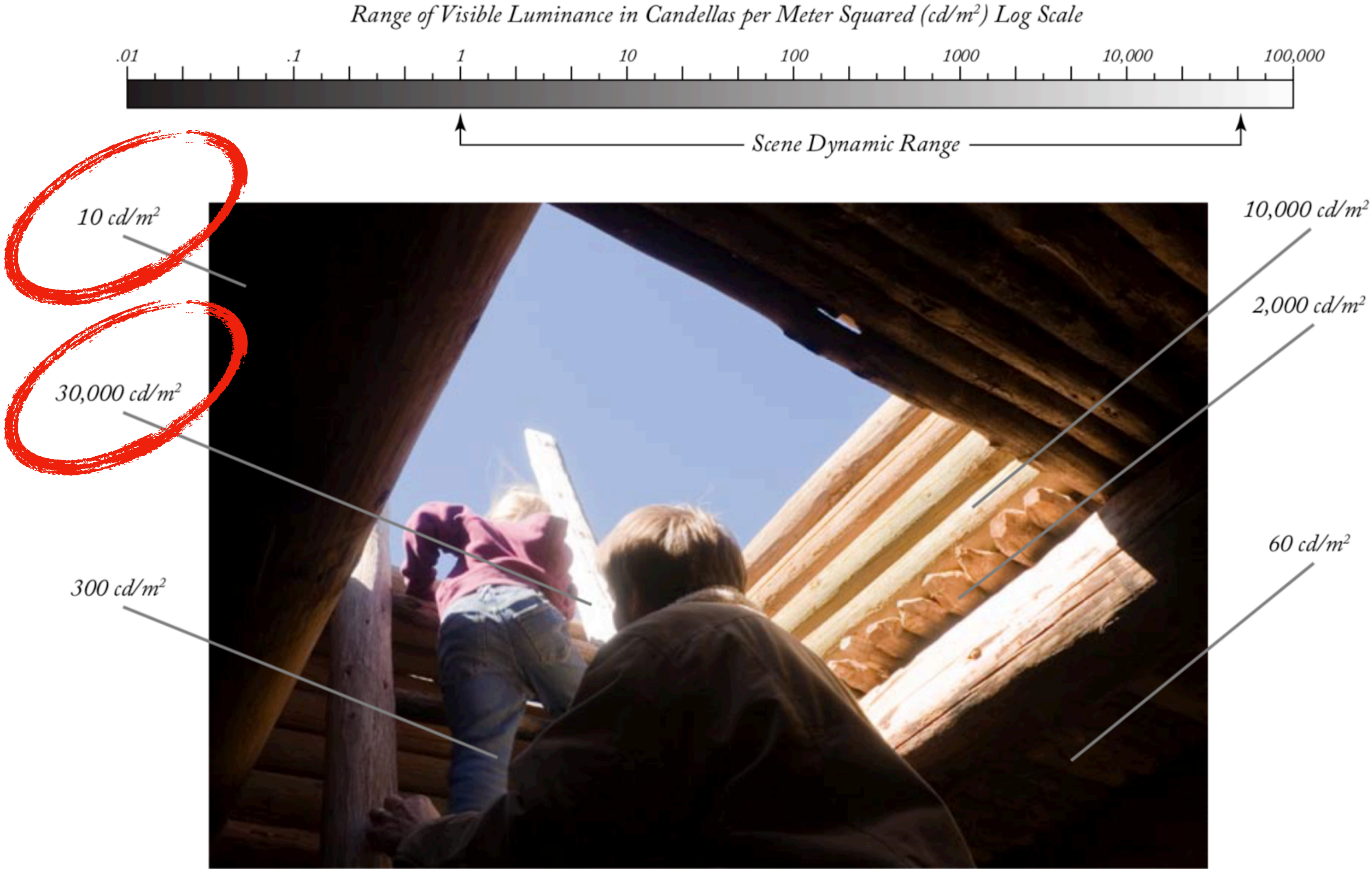
# HDR & Tone Mapping



# Natural Scenes Have High Dynamic Range

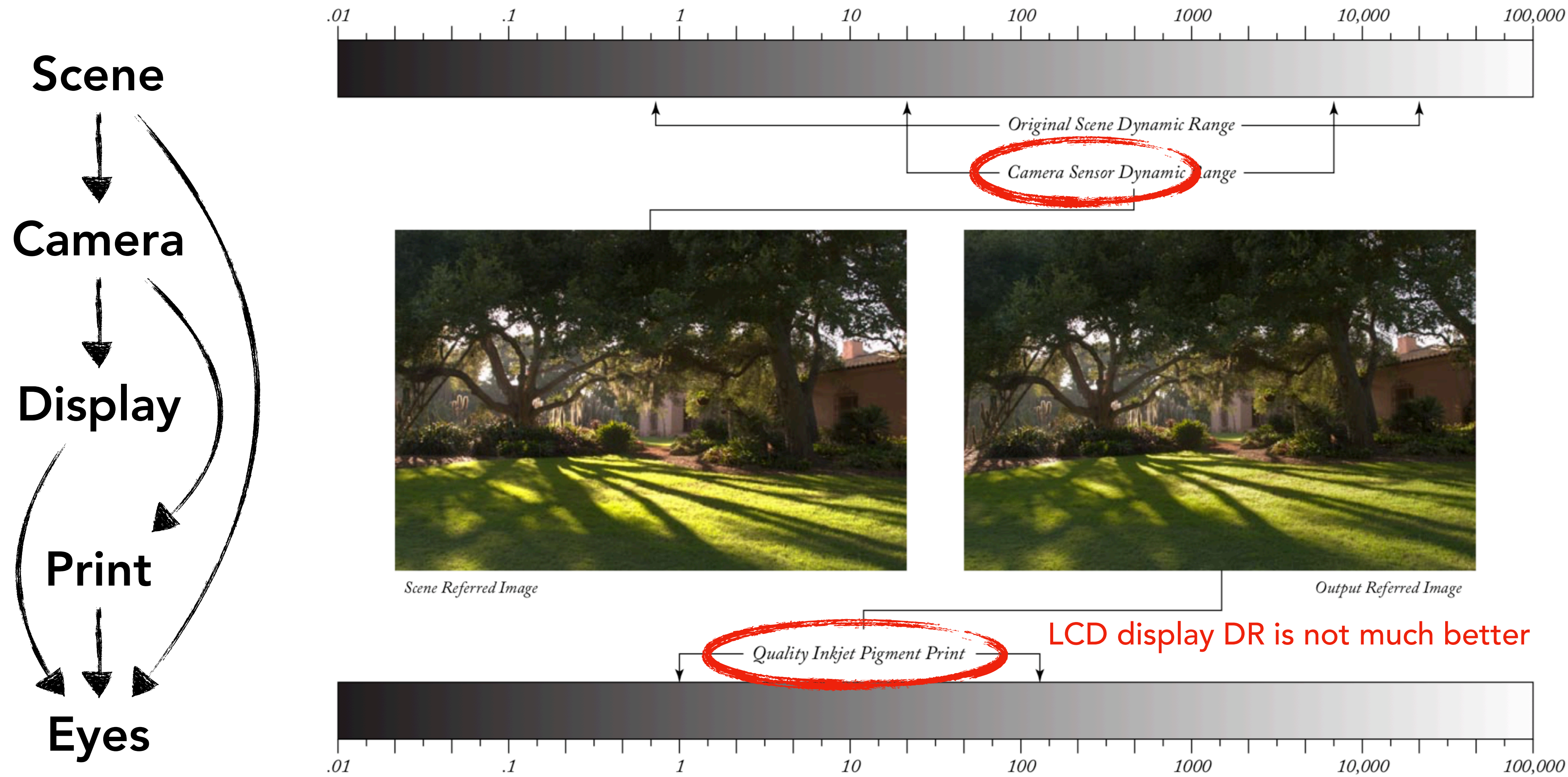
Scene

Eyes



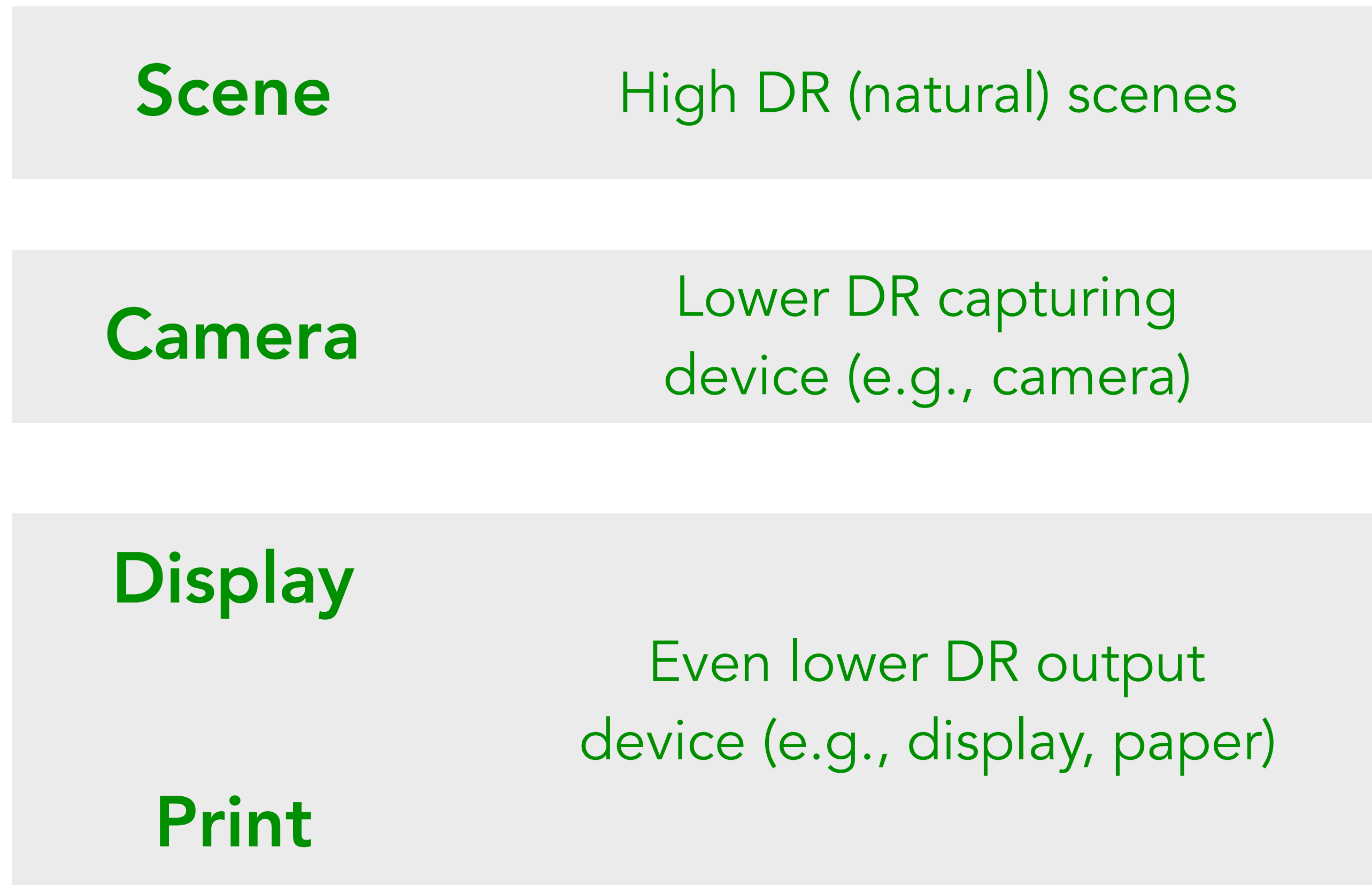


# The Problem





# Two Related Tasks in Sensor Signal Processing

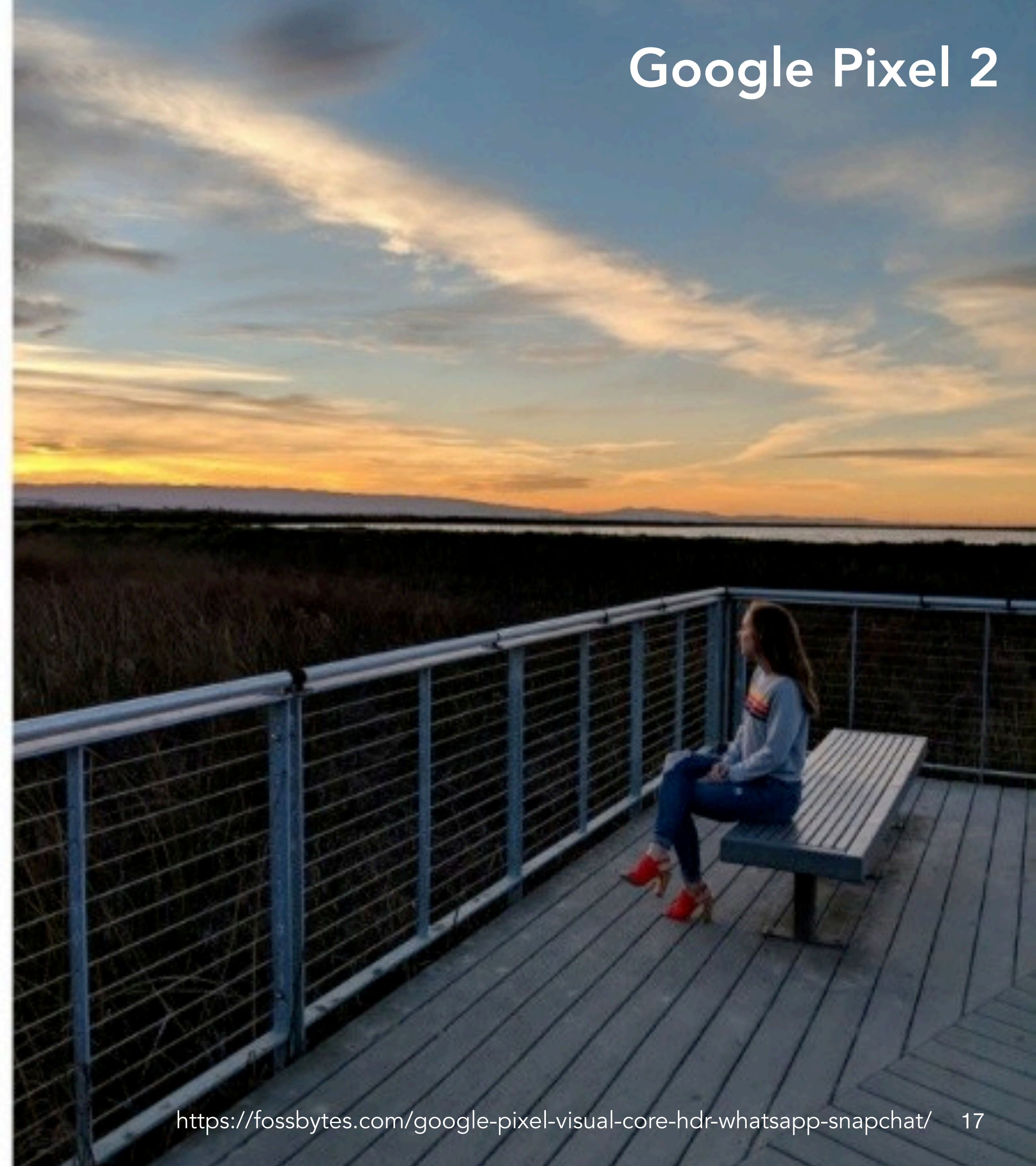


- 1. HDR Imaging:** how to capture an HDR scene with a lower DR capturing device. Limited by the camera.
- 2. Tone Mapping:** how to display HDR images on lower DR display devices. Limited by the display medium.

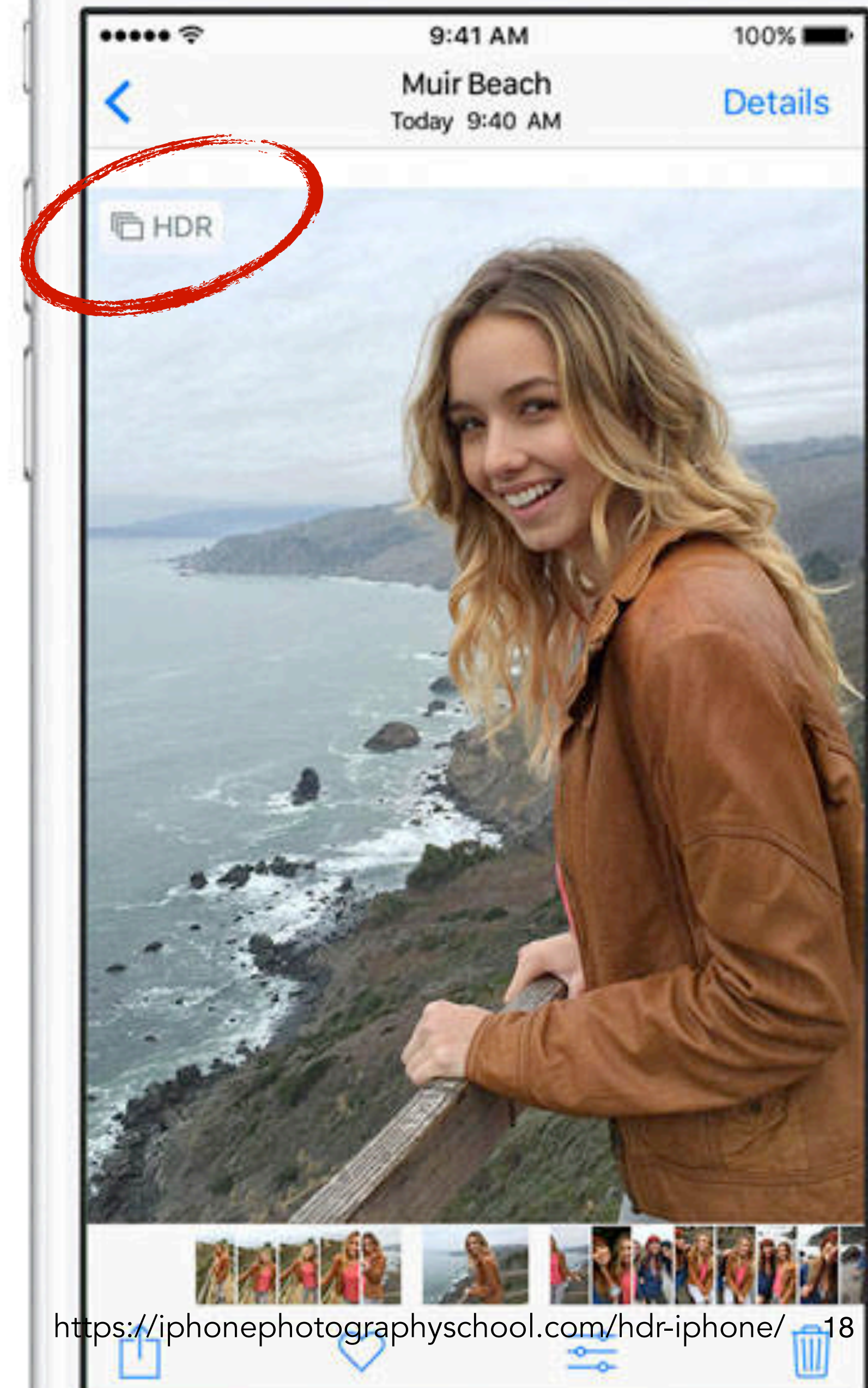
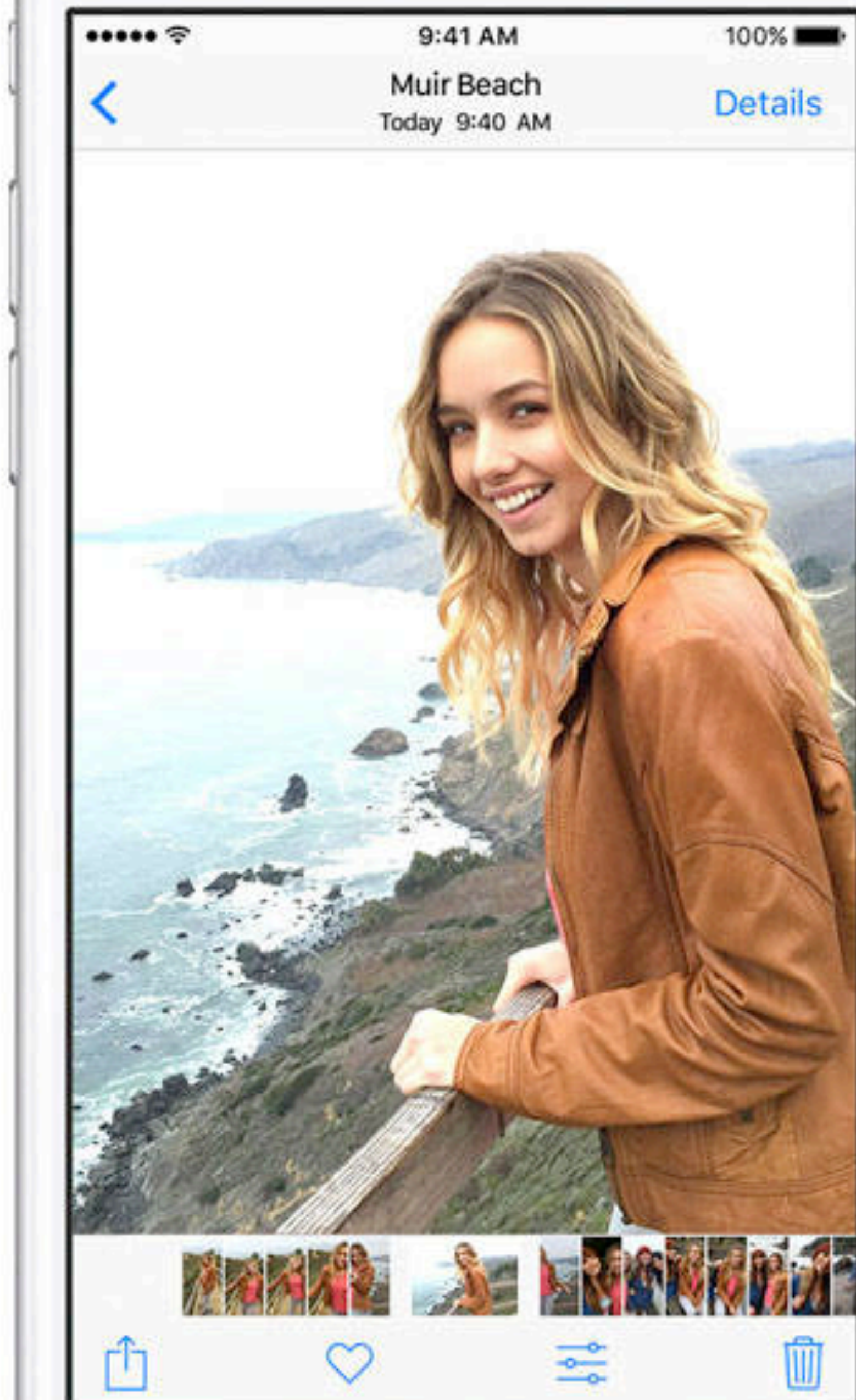
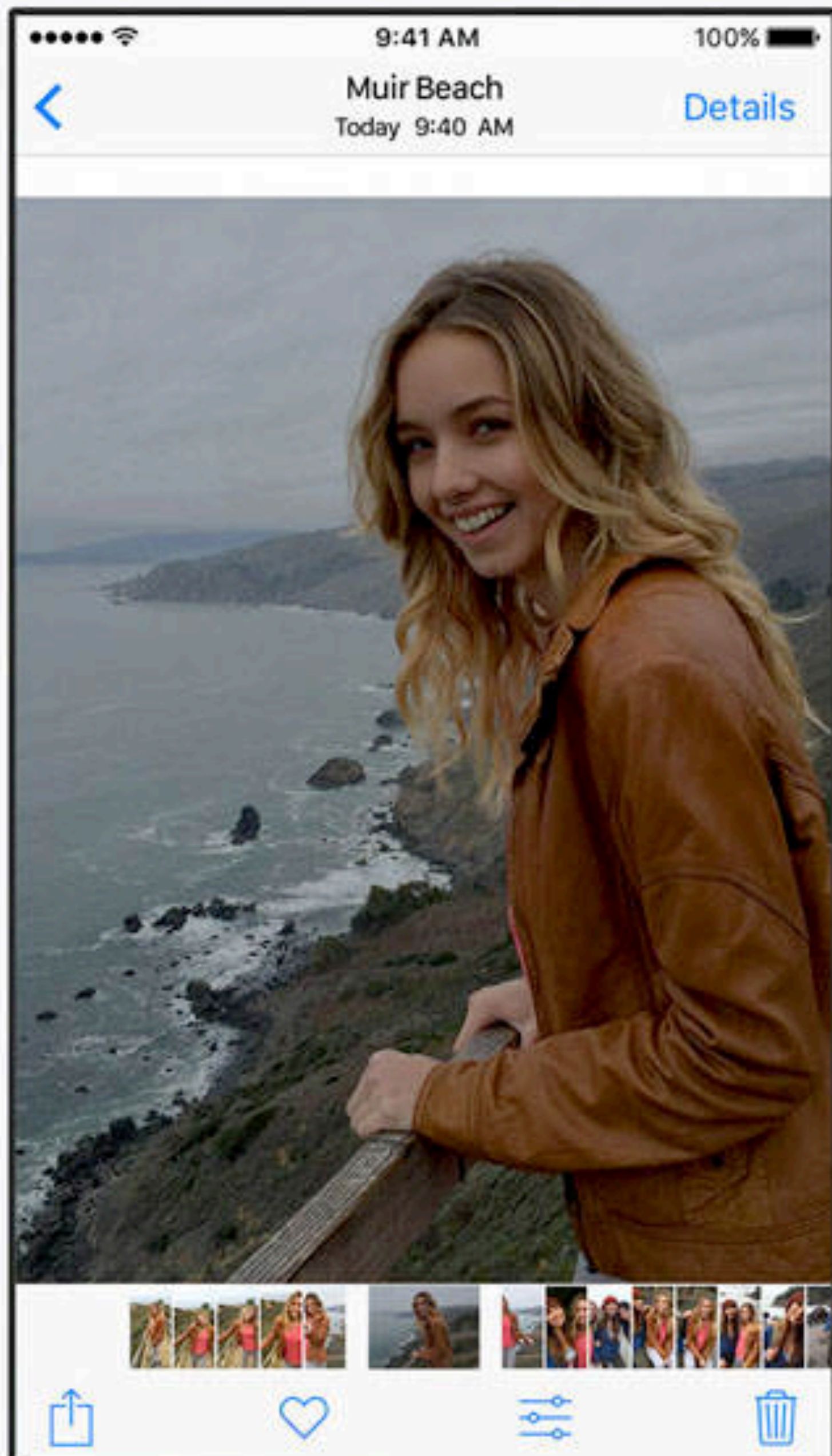




Google Pixel 2



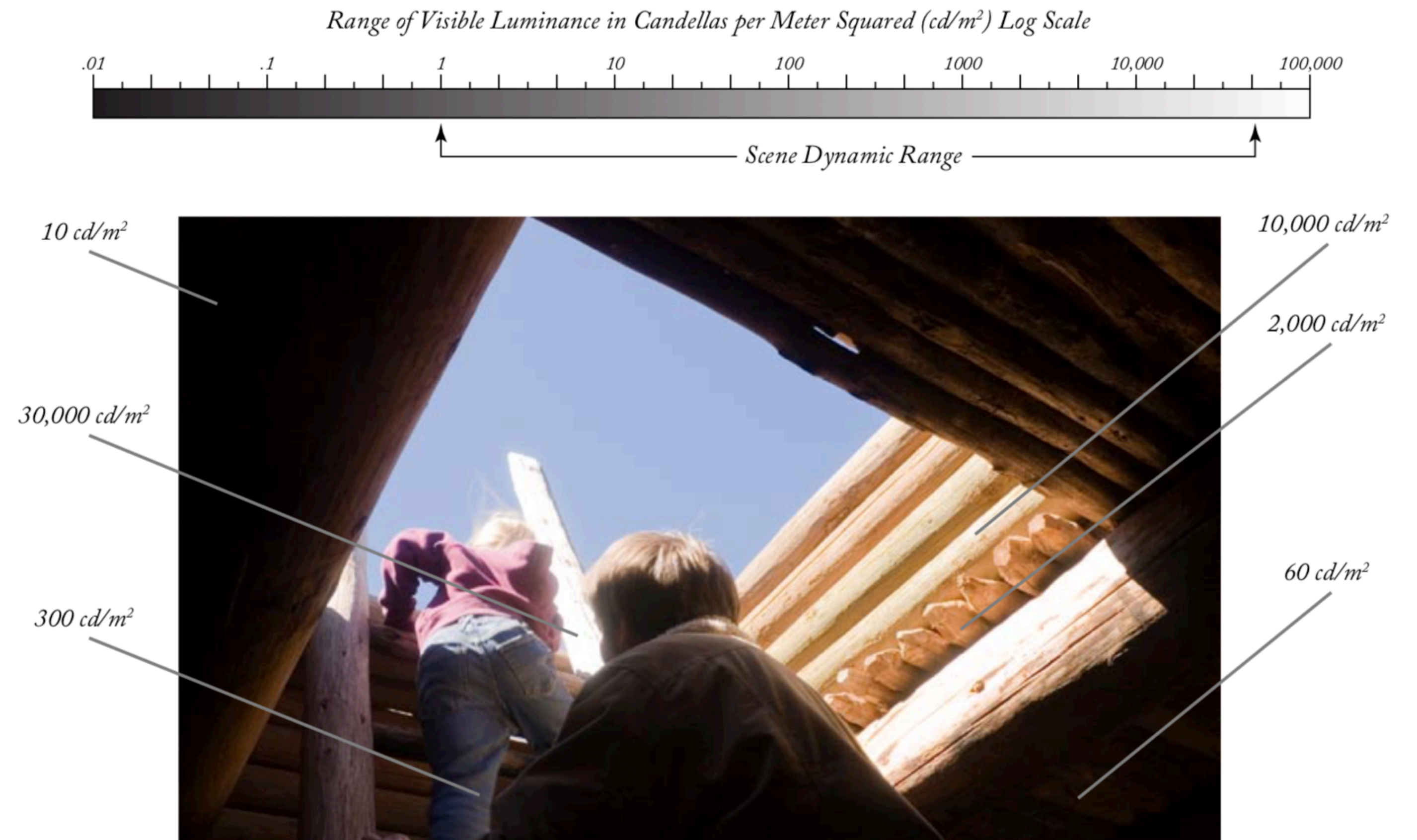






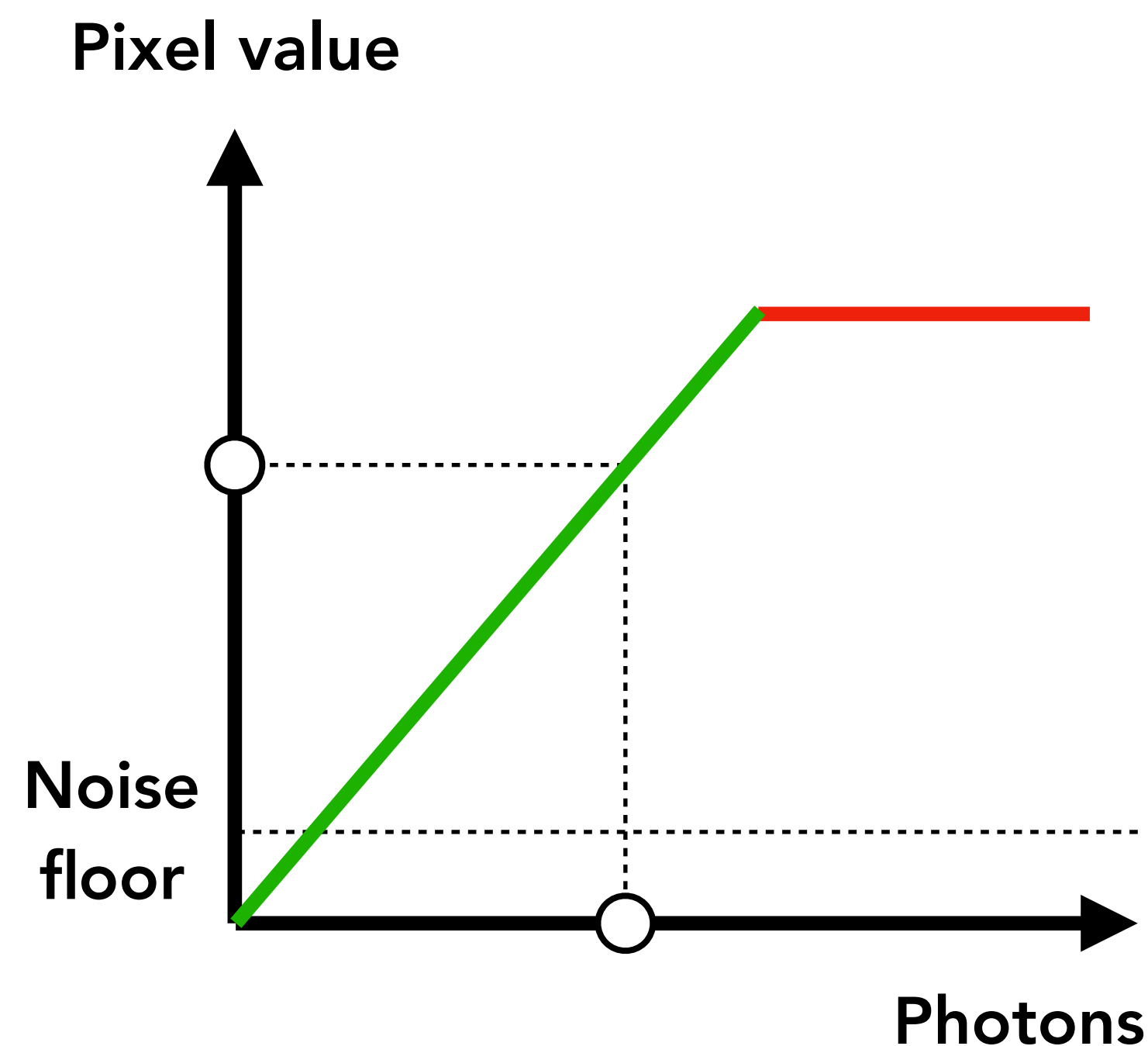
# Goal of HDR Imaging

1. Pixel values should represent the full range of scene luminance, i.e., no saturation.
2. The relative ratio of luminance should ideally be maintained (for now).
  - So that a simple scaling later, if needed, can recover the true luminance in the scene.



# Scene Luminance vs. Raw Pixel Value

Barring noise and quantization error, the RAW pixel value **n** and the incident photon quantity **P** are close to **linear** between the noise floor and saturation.



$$Q = P \times SSF \times QE$$

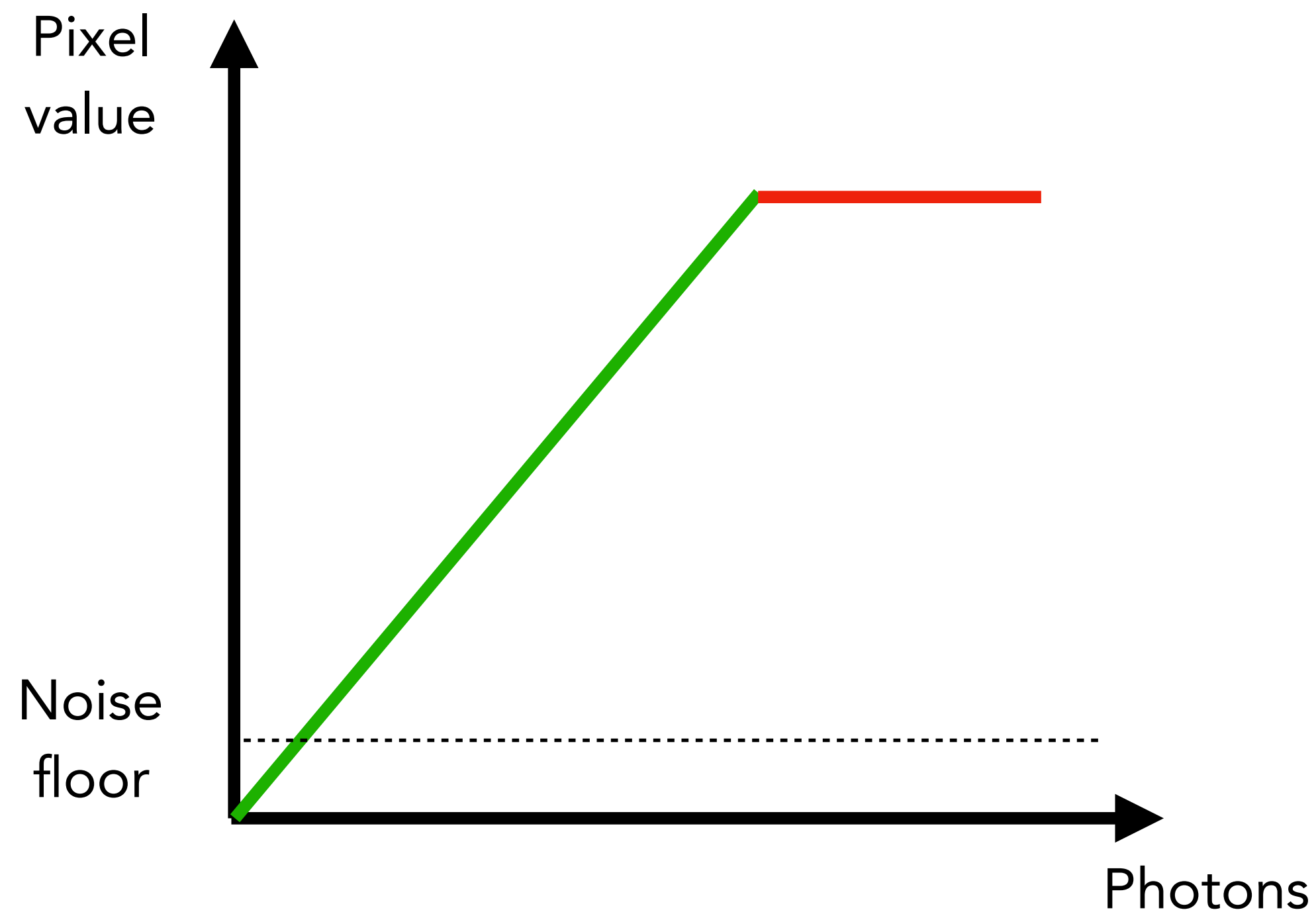
# of photons/pixel

$$V = \frac{Q}{C} g$$

$$n = \left\lfloor \frac{V - V_{min} + s/2}{s} \right\rfloor$$

# Dynamic Range Issue

**Problem:** full luminance range isn't *accurately* captured.



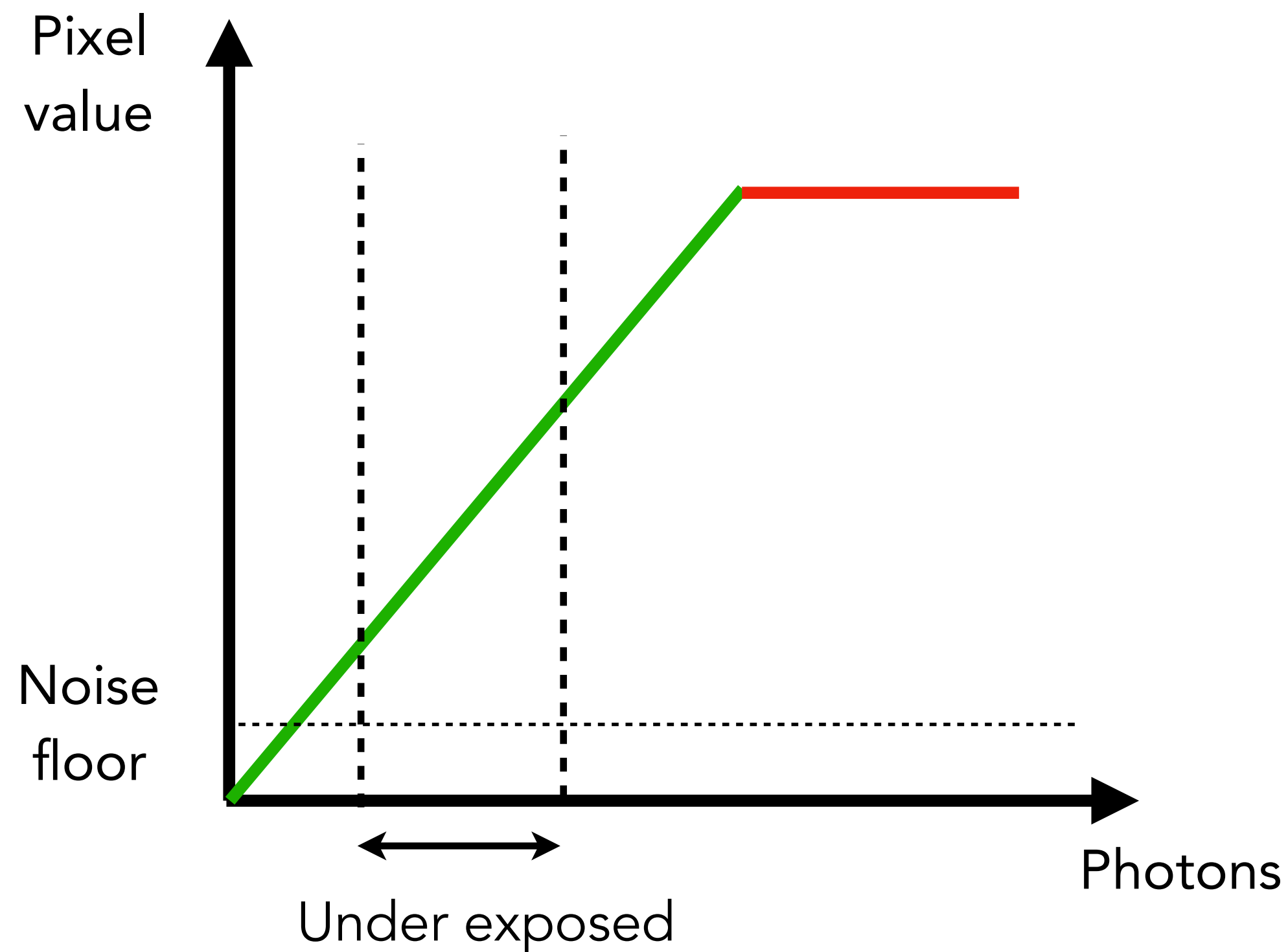
When pixels are under-exposed, we in theory could recover the luminance, but the SNR would be very low, so the luminance values are noisy.

- Simply amplifying all the pixel values will not work because the noise gets amplified too. The SNR is same (i.e., still very noisy).
- Denoising has the effect of enhancing DR.

When pixels are over-exposed, pixel values don't reflect scene luminance.

# Dynamic Range Issue

**Problem:** full luminance range isn't *accurately* captured.



When pixels are under-exposed, we in theory could recover the luminance, but the SNR would be very low, so the luminance values are noisy.

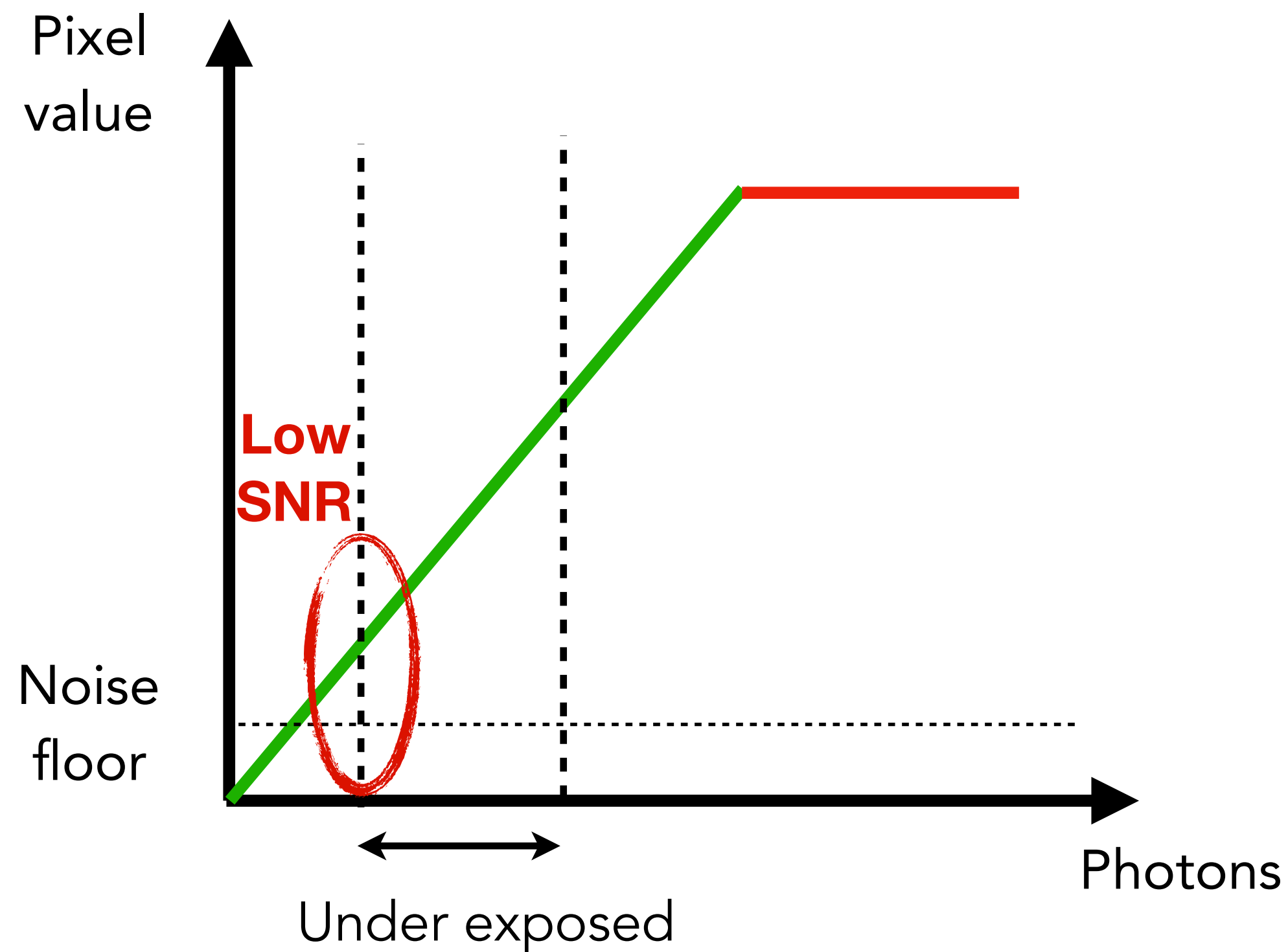
- Simply amplifying all the pixel values will not work because the noise gets amplified too. The SNR is same (i.e., still very noisy).
- Denoising has the effect of enhancing DR.

When pixels are over-exposed, pixel values don't reflect scene luminance.



# Dynamic Range Issue

**Problem:** full luminance range isn't *accurately* captured.



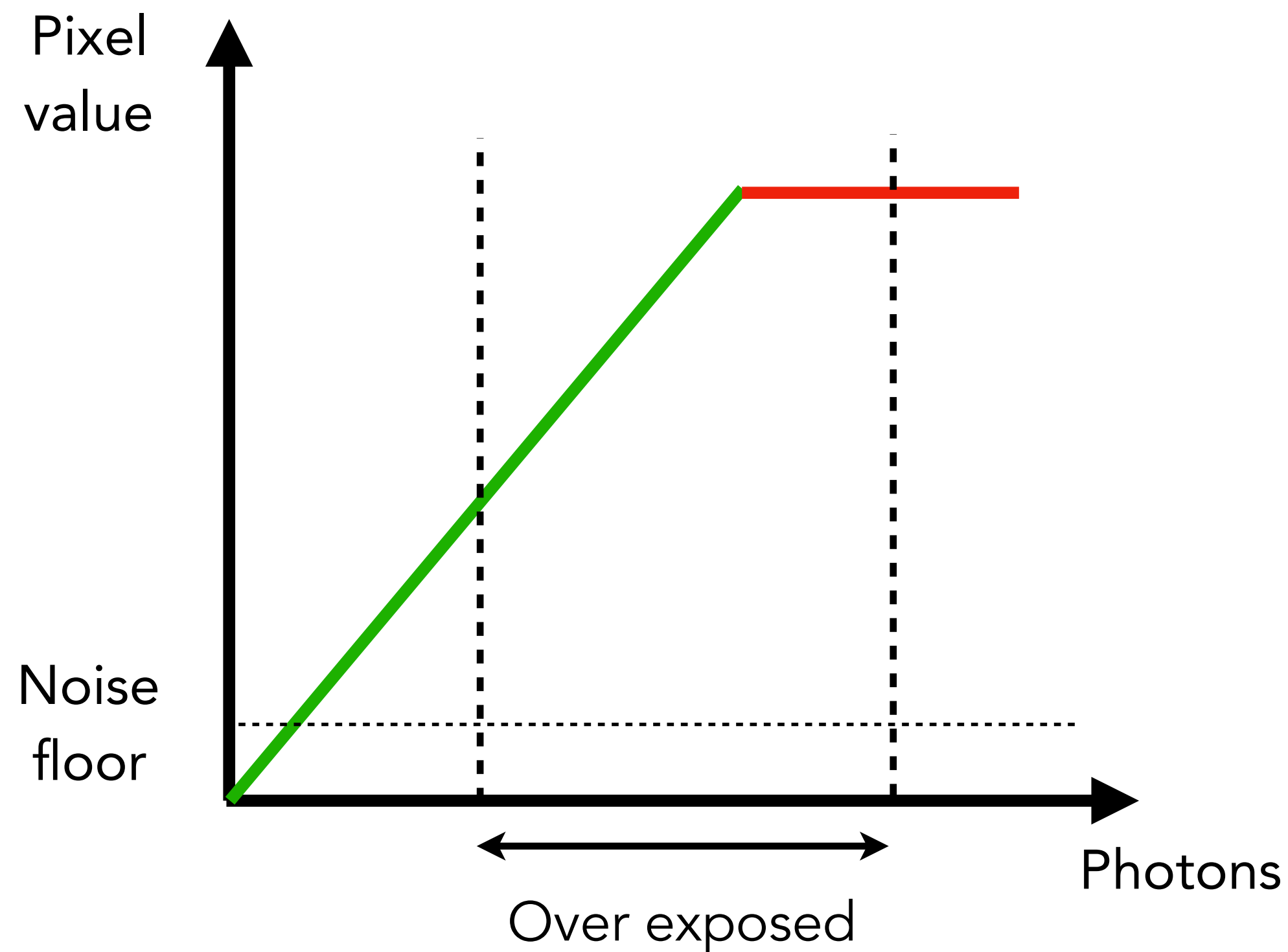
When pixels are under-exposed, we in theory could recover the luminance, but the SNR would be very low, so the luminance values are noisy.

- Simply amplifying all the pixel values will not work because the noise gets amplified too. The SNR is same (i.e., still very noisy).
- Denoising has the effect of enhancing DR.

When pixels are over-exposed, pixel values don't reflect scene luminance.

# Dynamic Range Issue

**Problem:** full luminance range isn't *accurately* captured.



When pixels are under-exposed, we in theory could recover the luminance, but the SNR would be very low, so the luminance values are noisy.

- Simply amplifying all the pixel values will not work because the noise gets amplified too. The SNR is same (i.e., still very noisy).
- Denoising has the effect of enhancing DR.

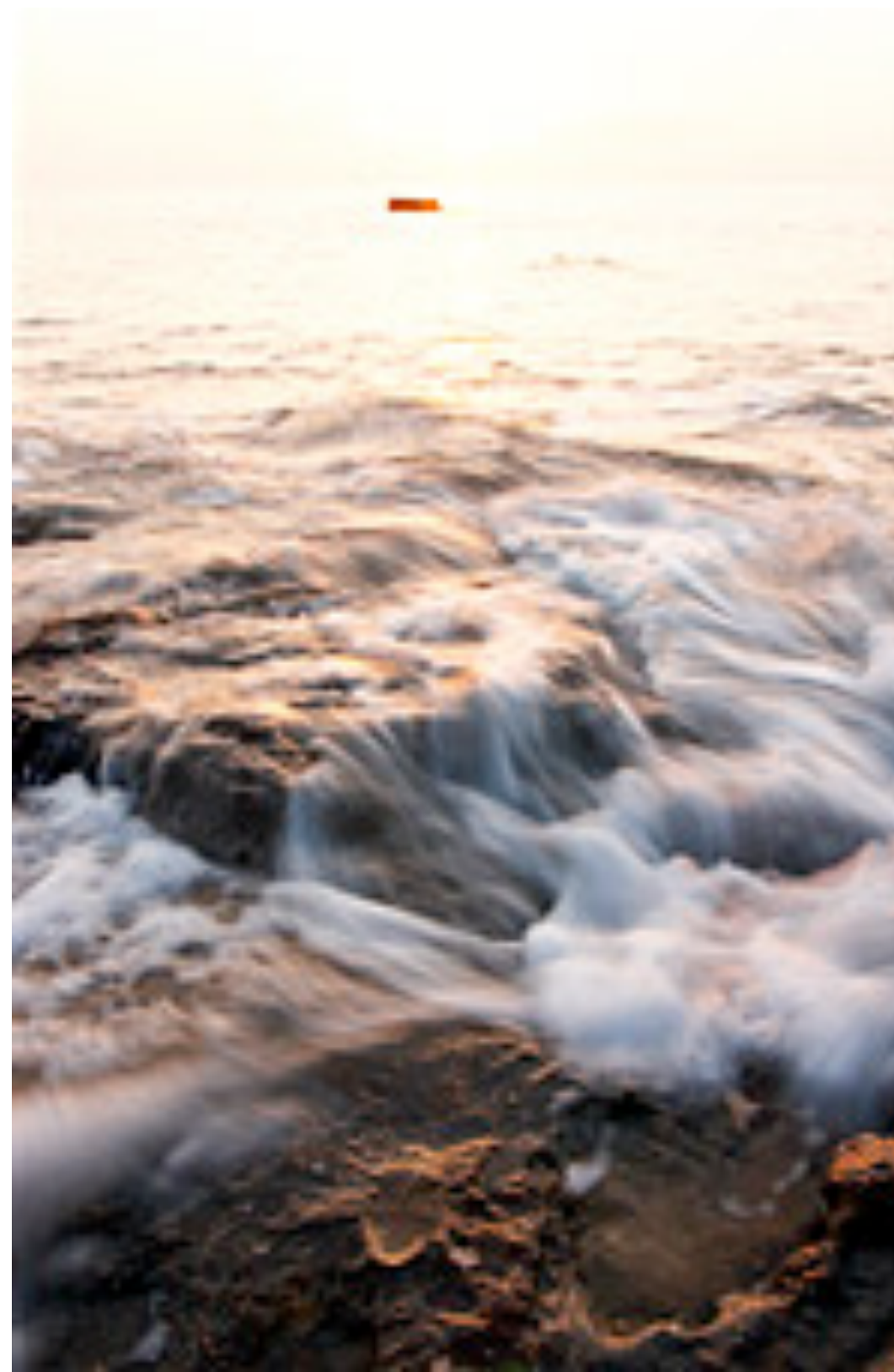
When pixels are over-exposed, pixel values don't reflect scene luminance.

# HDR Using Optics

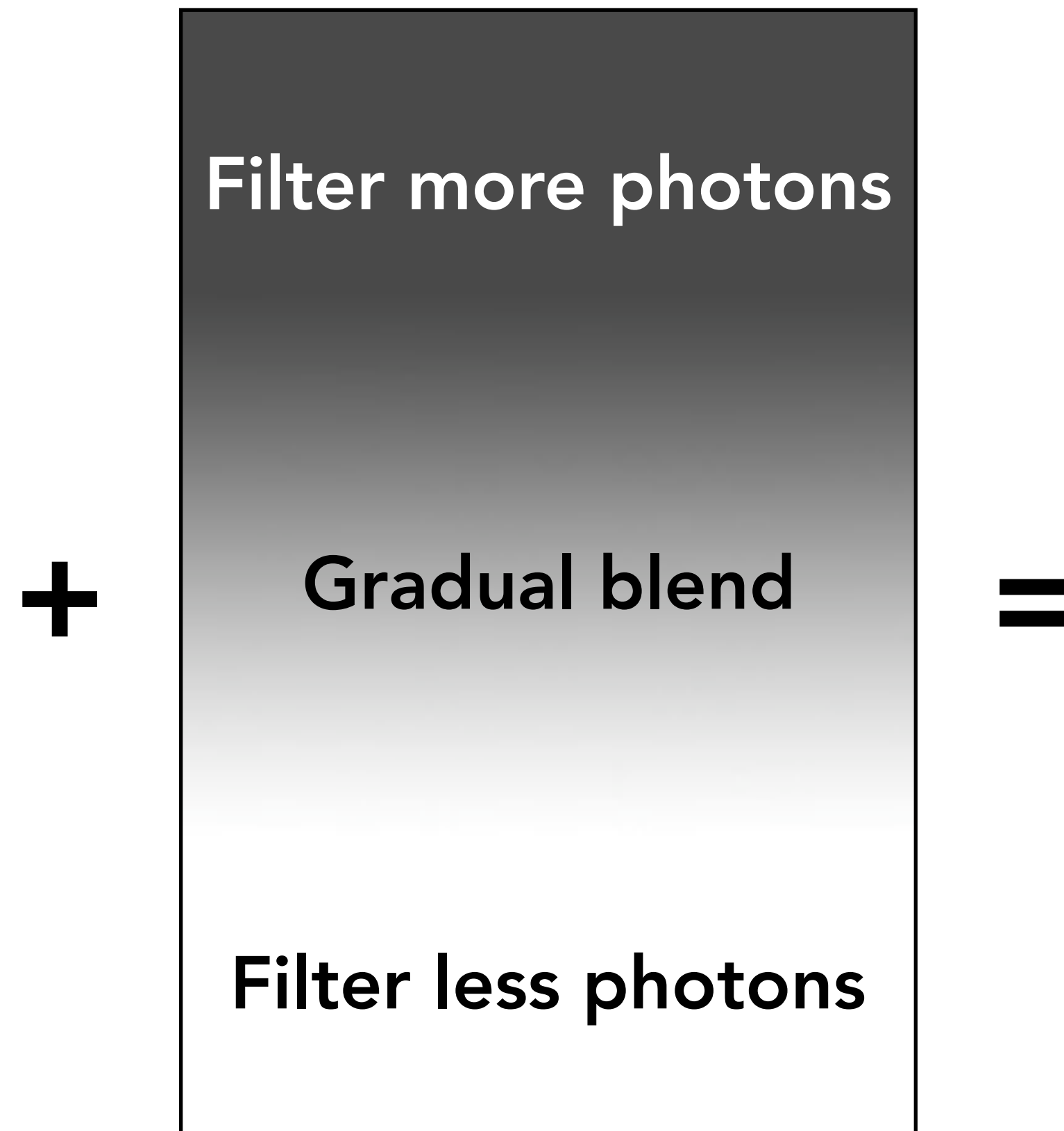


# HDR Imaging Through Neutral Density (ND) Filter

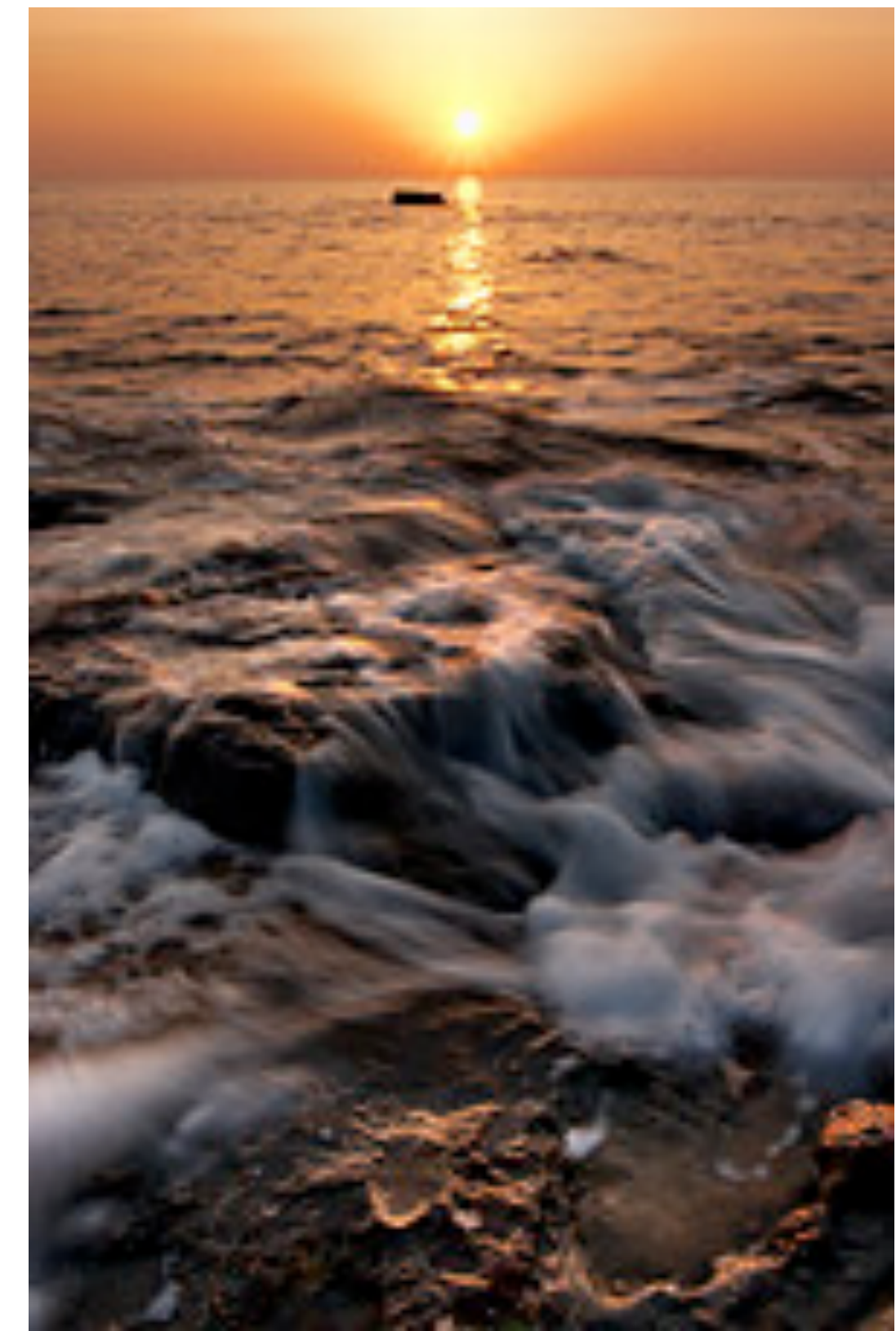
Original Scene



Graduated ND Filter



Output Image



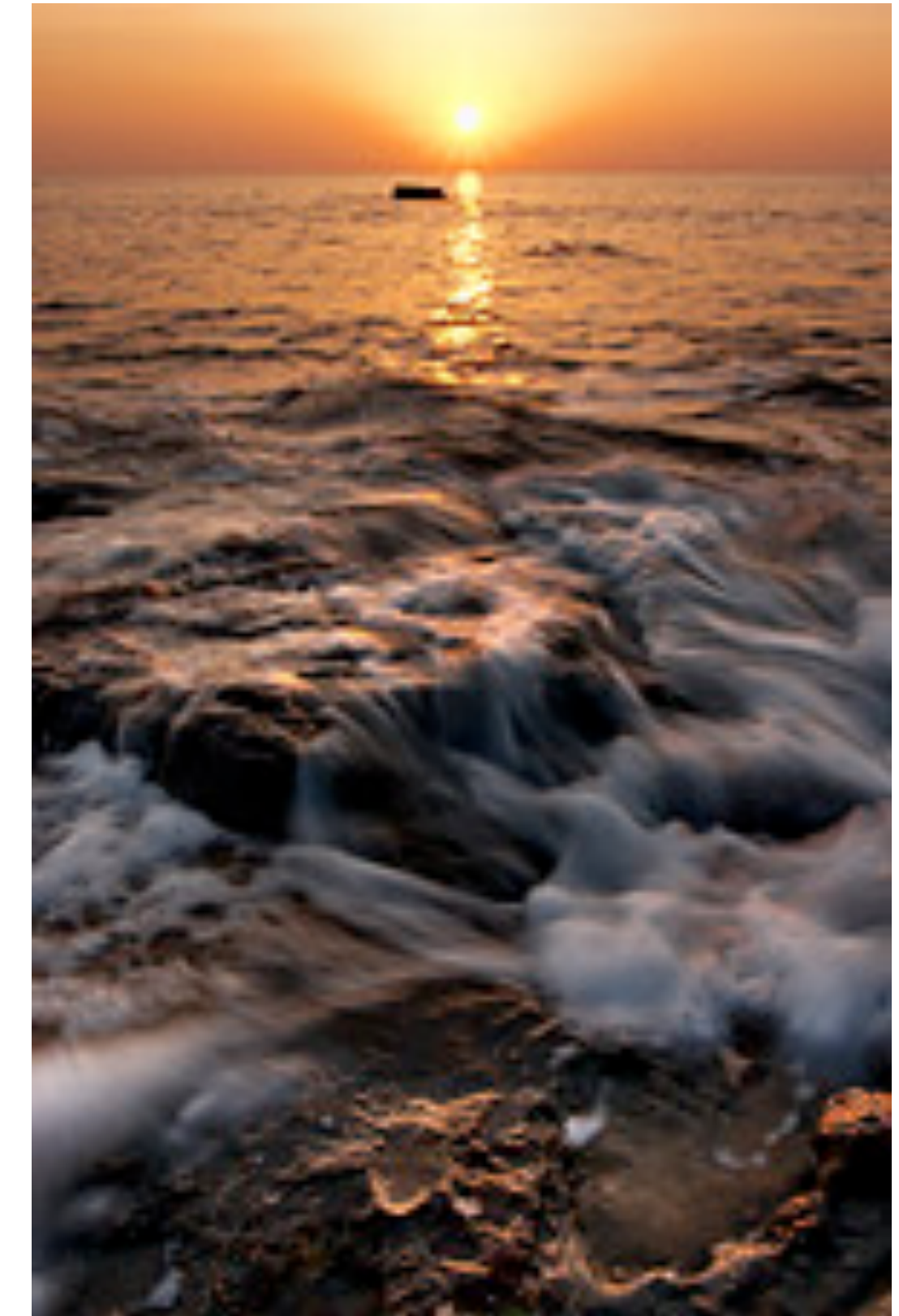
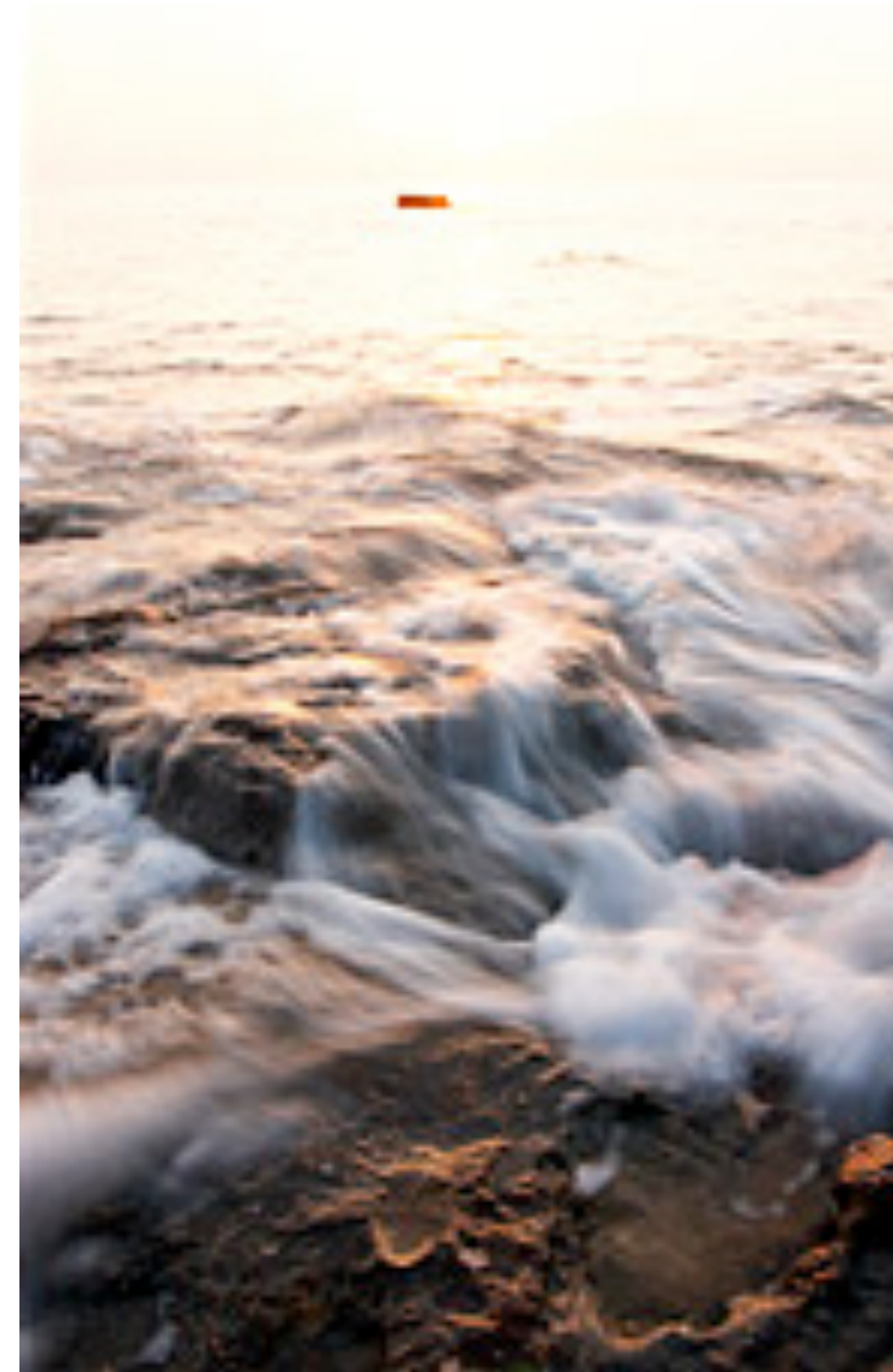
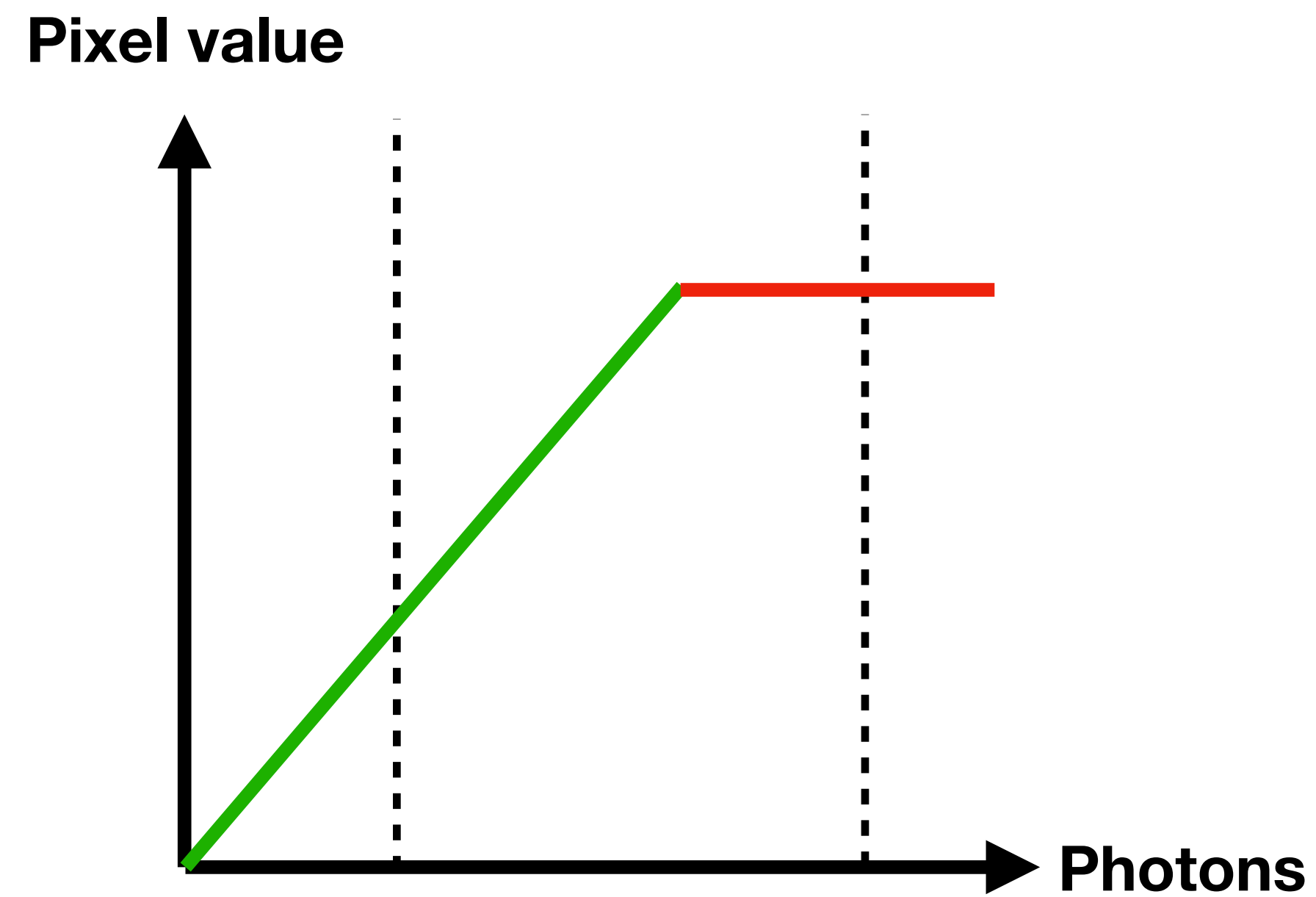


Tiffen 77CGND6 77mm Graduated 0.6 ND Filter.  
4X exposure difference. High-end GND filters  
provide ~4 orders of magnitude of difference.



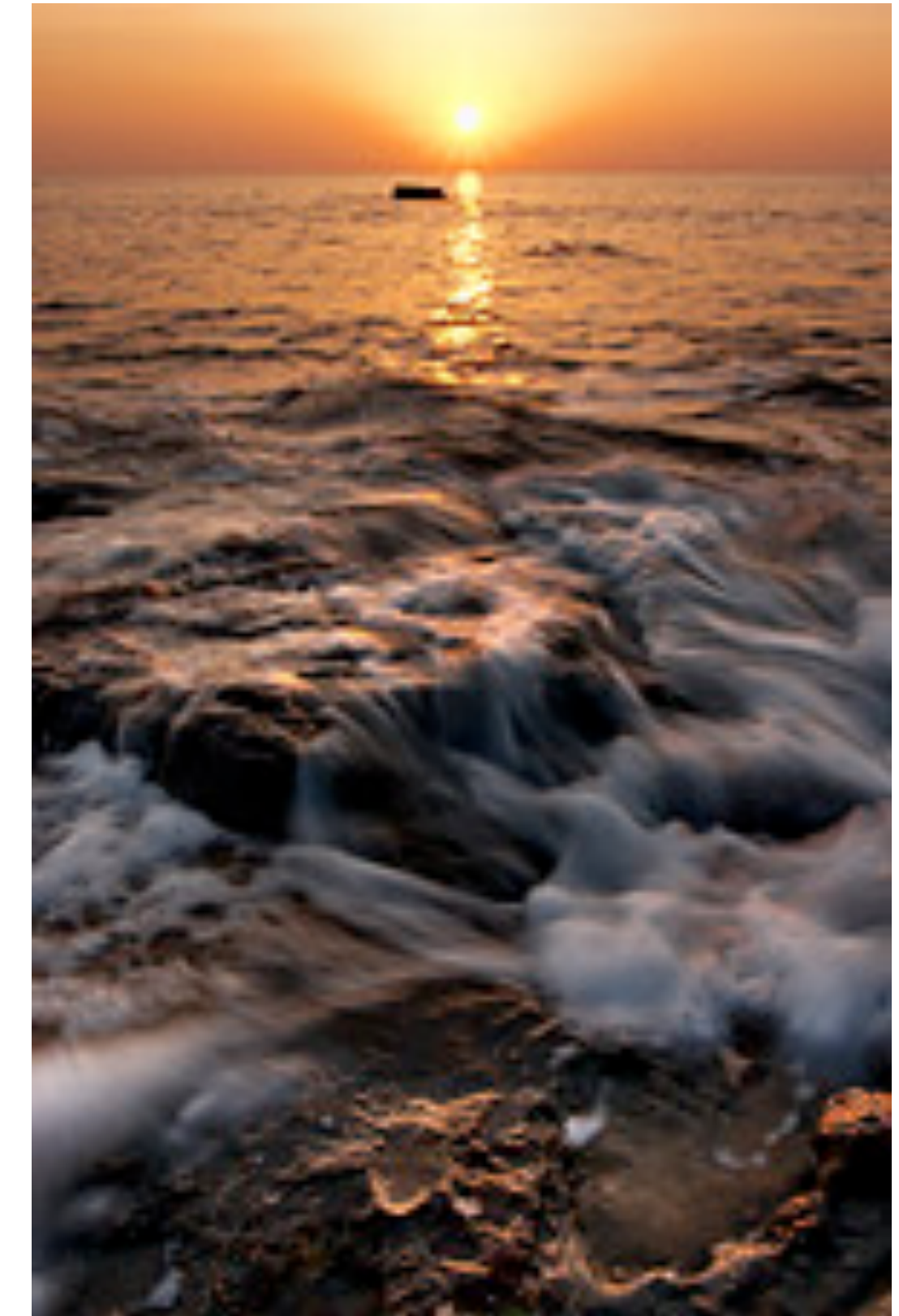
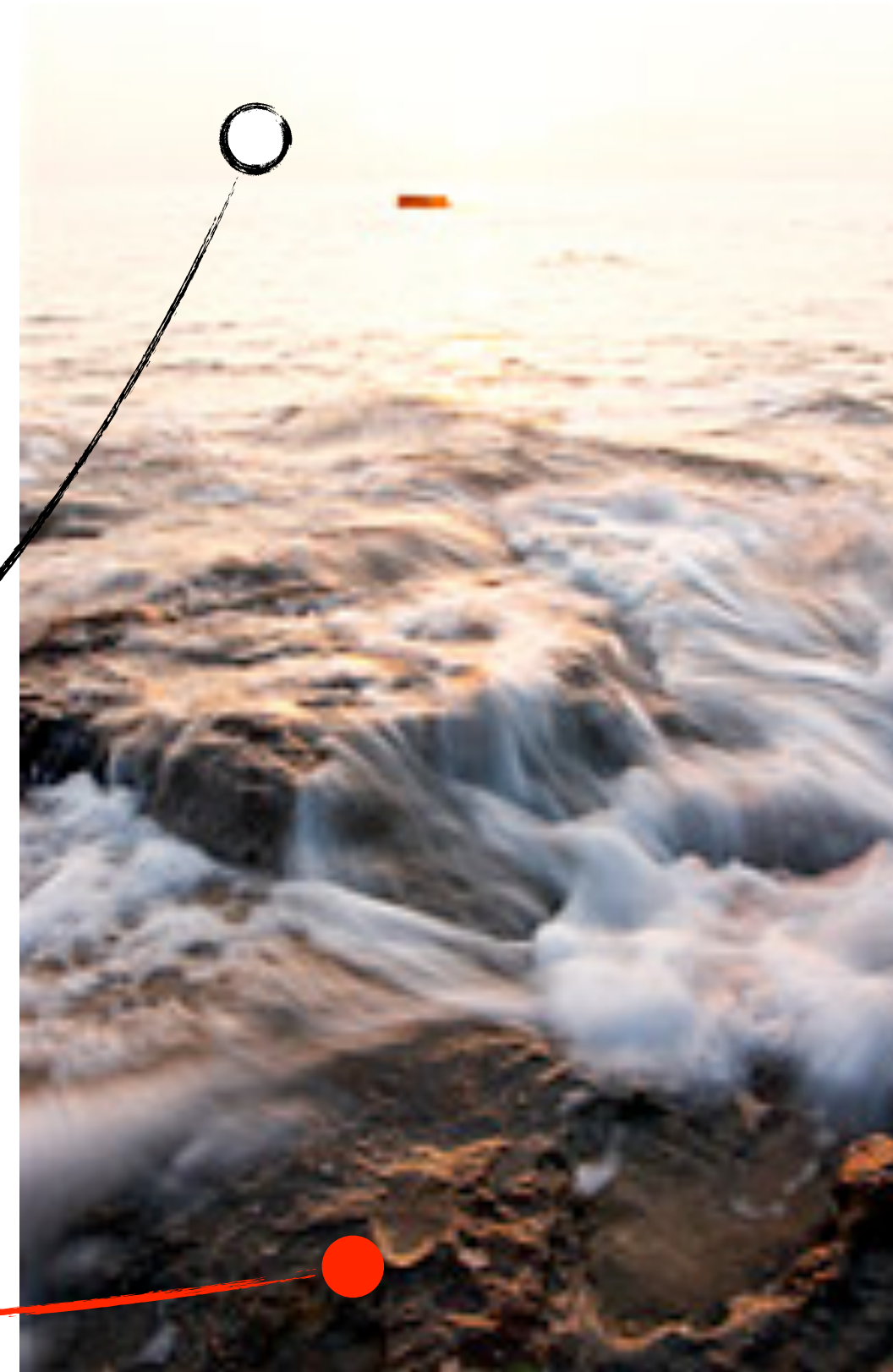
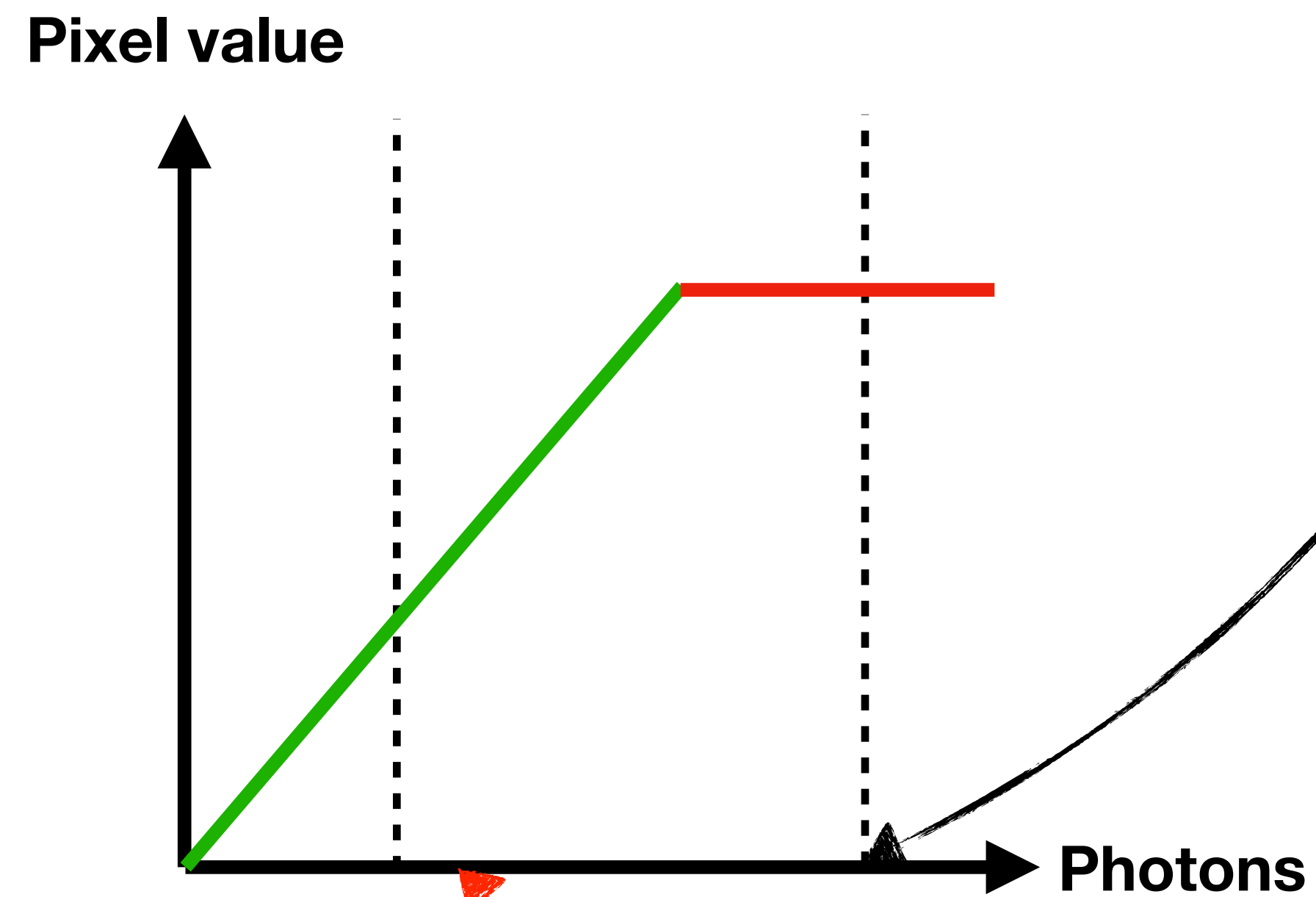


# What's Happening When Using a GDN Filter?



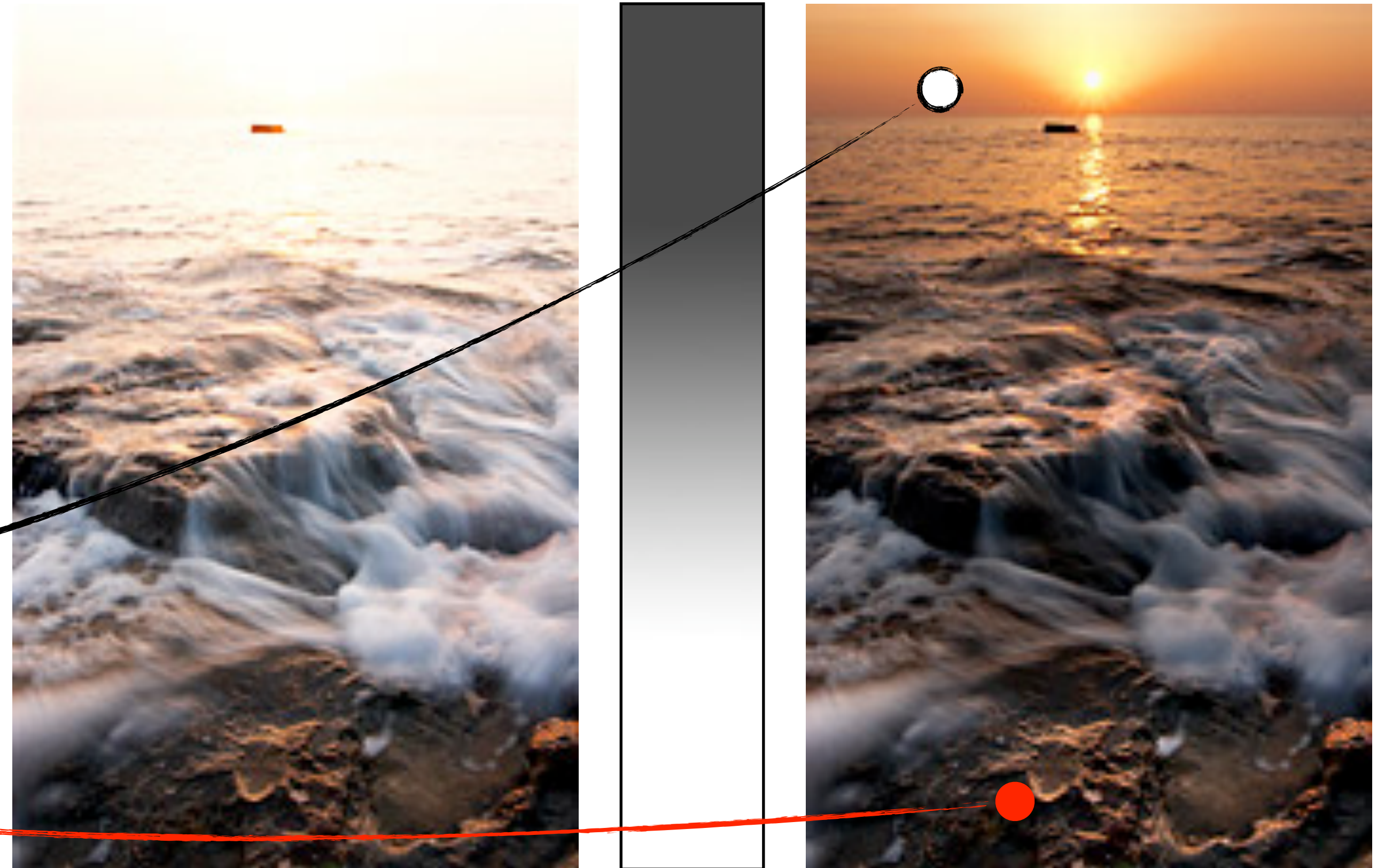
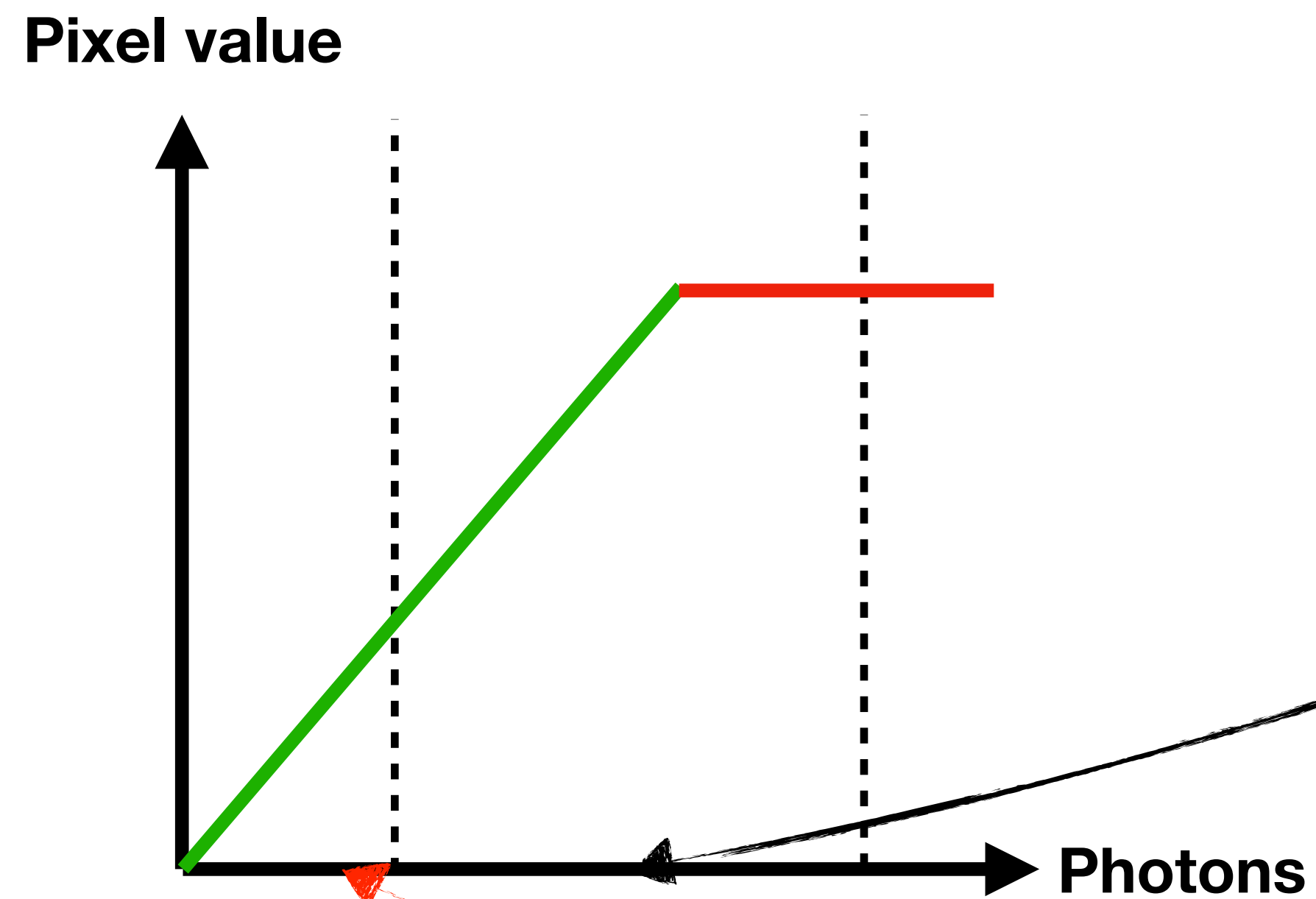


# What's Happening When Using a GDN Filter?



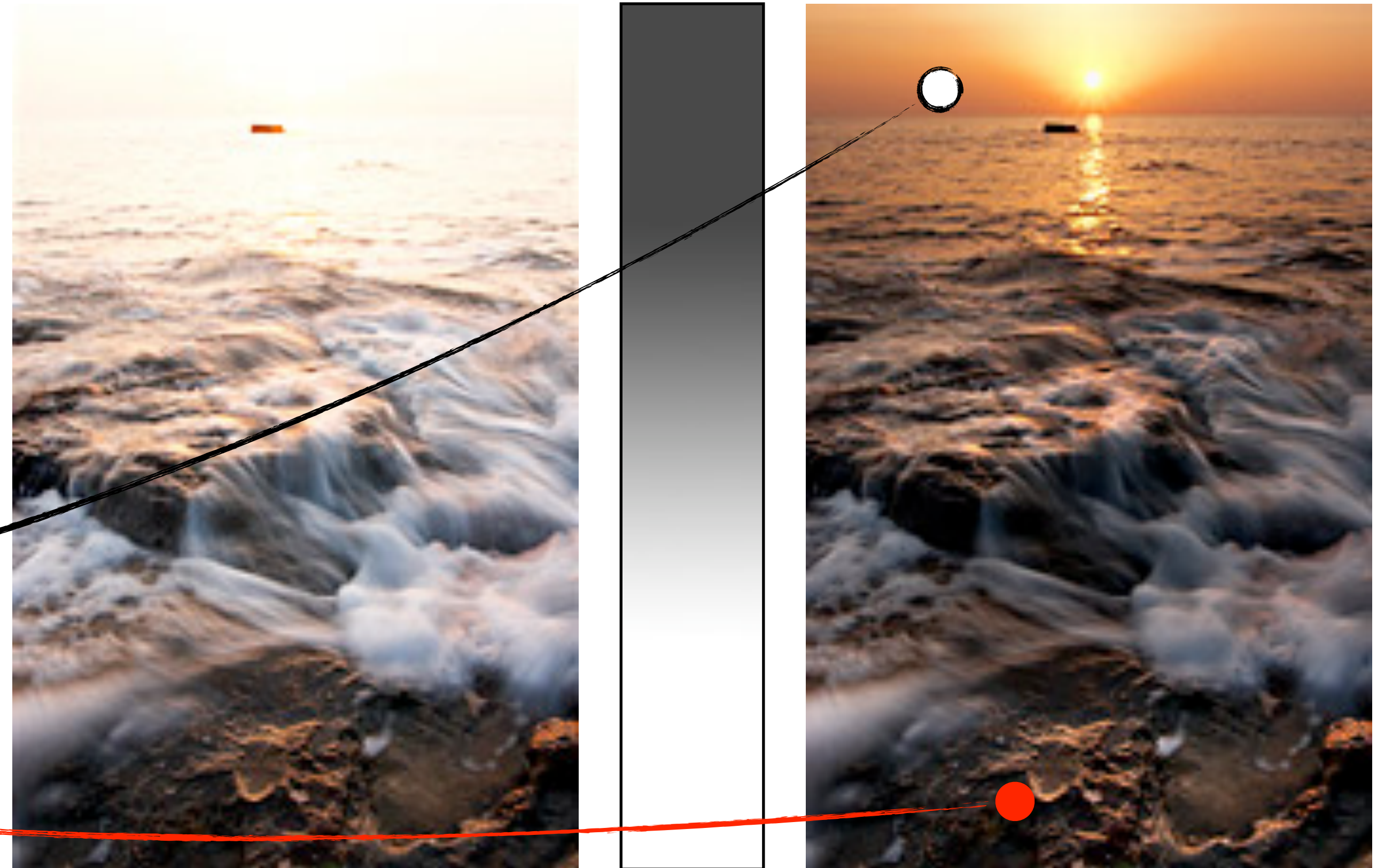
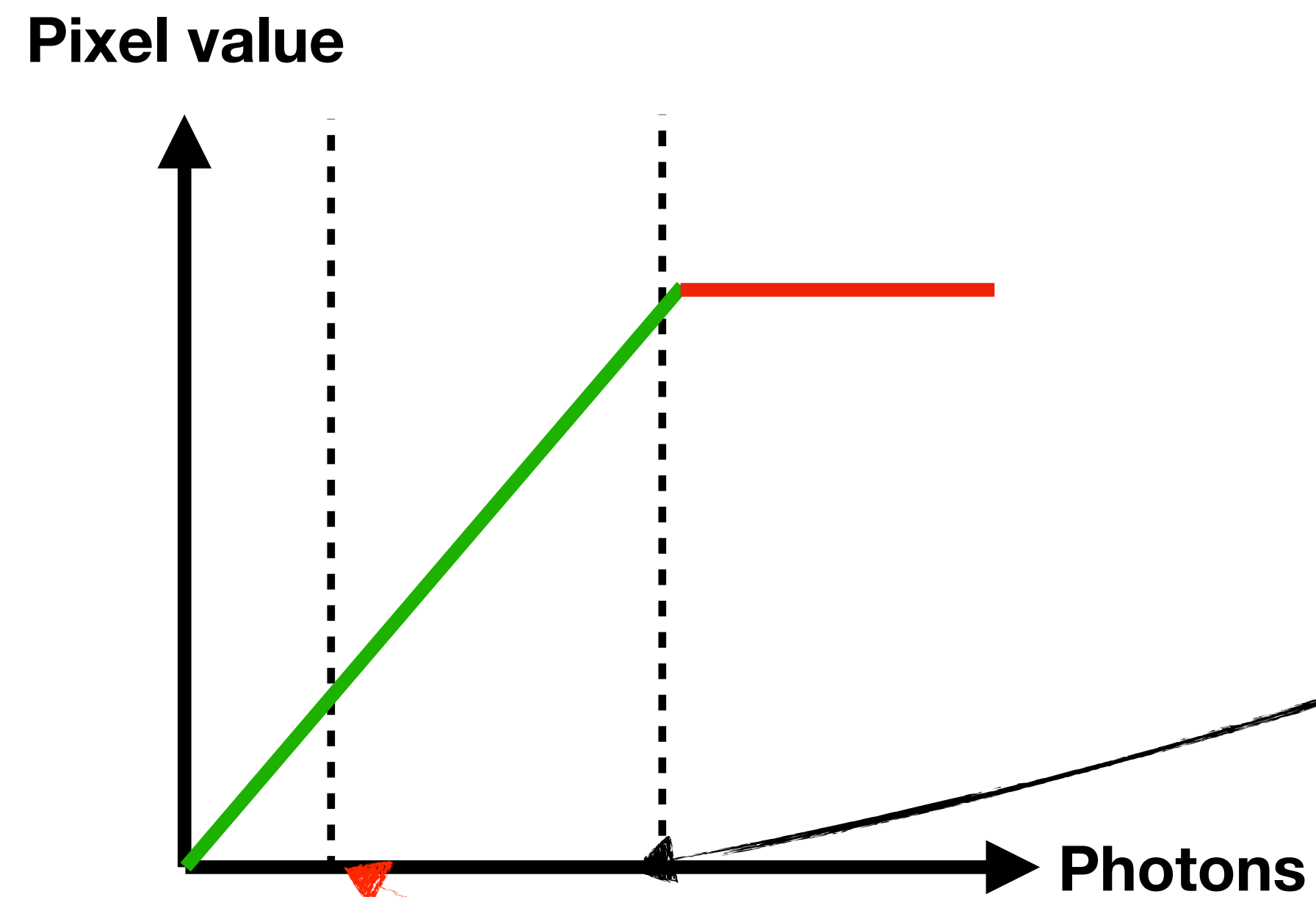


# What's Happening When Using a GDN Filter?



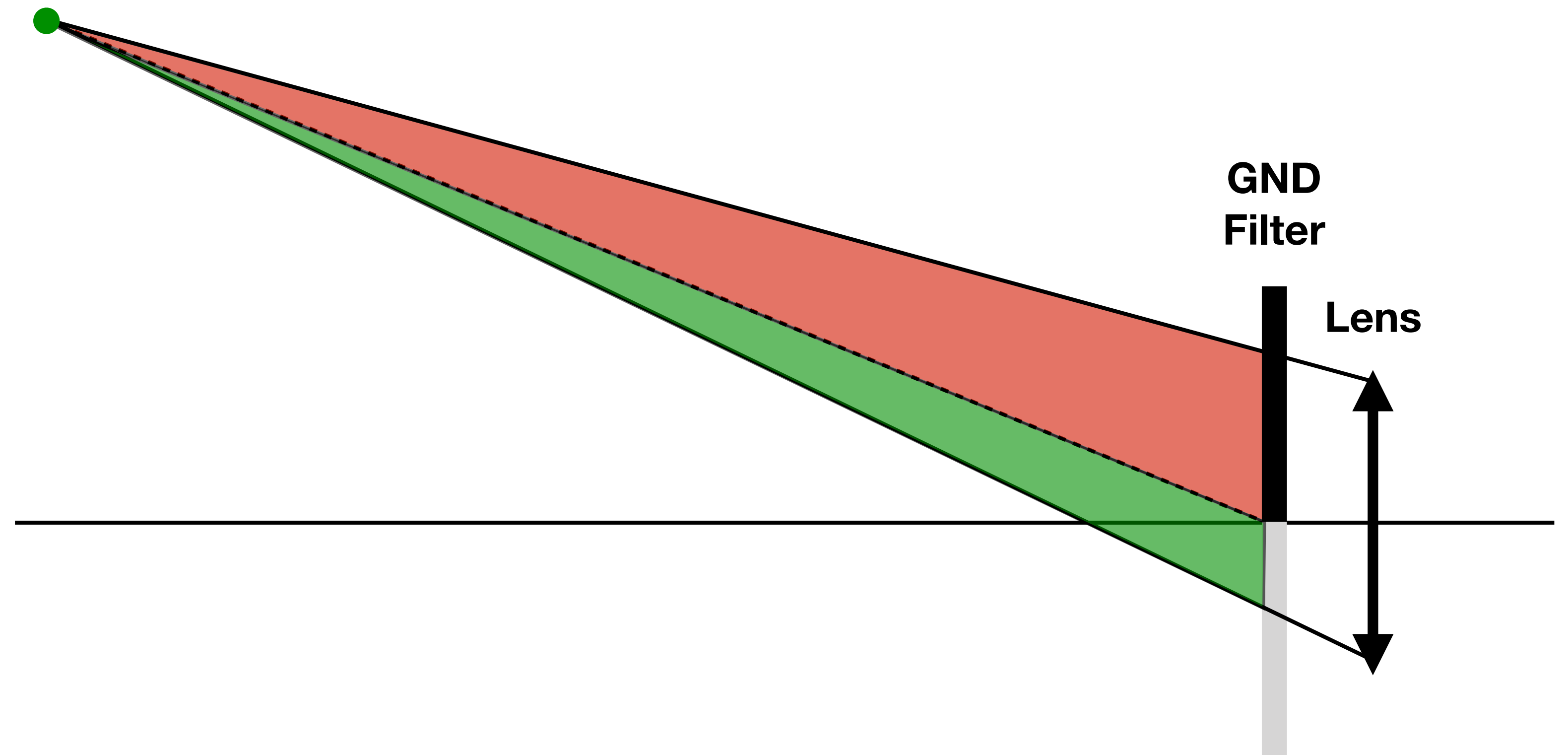
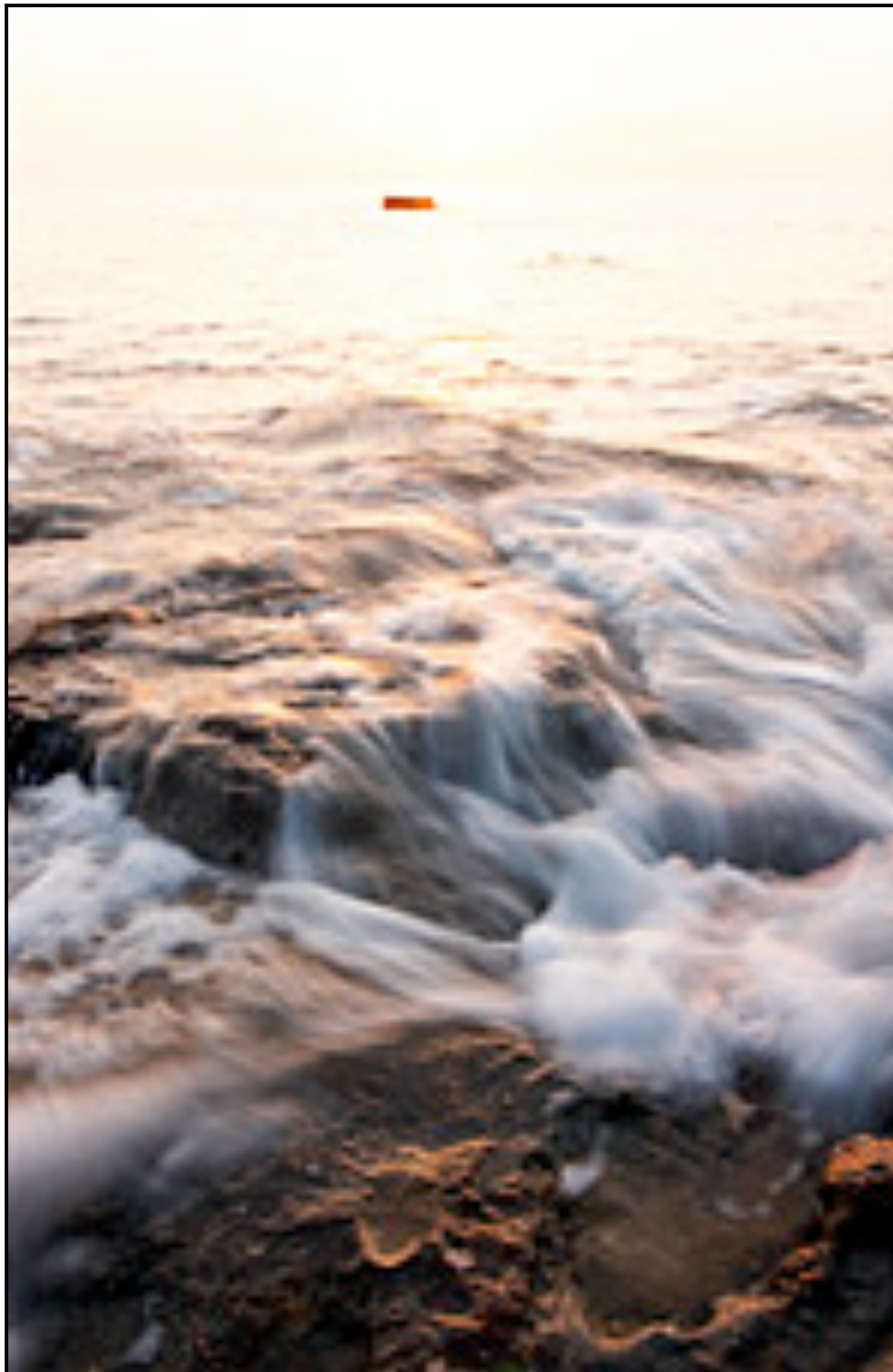


# What's Happening When Using a GDN Filter?



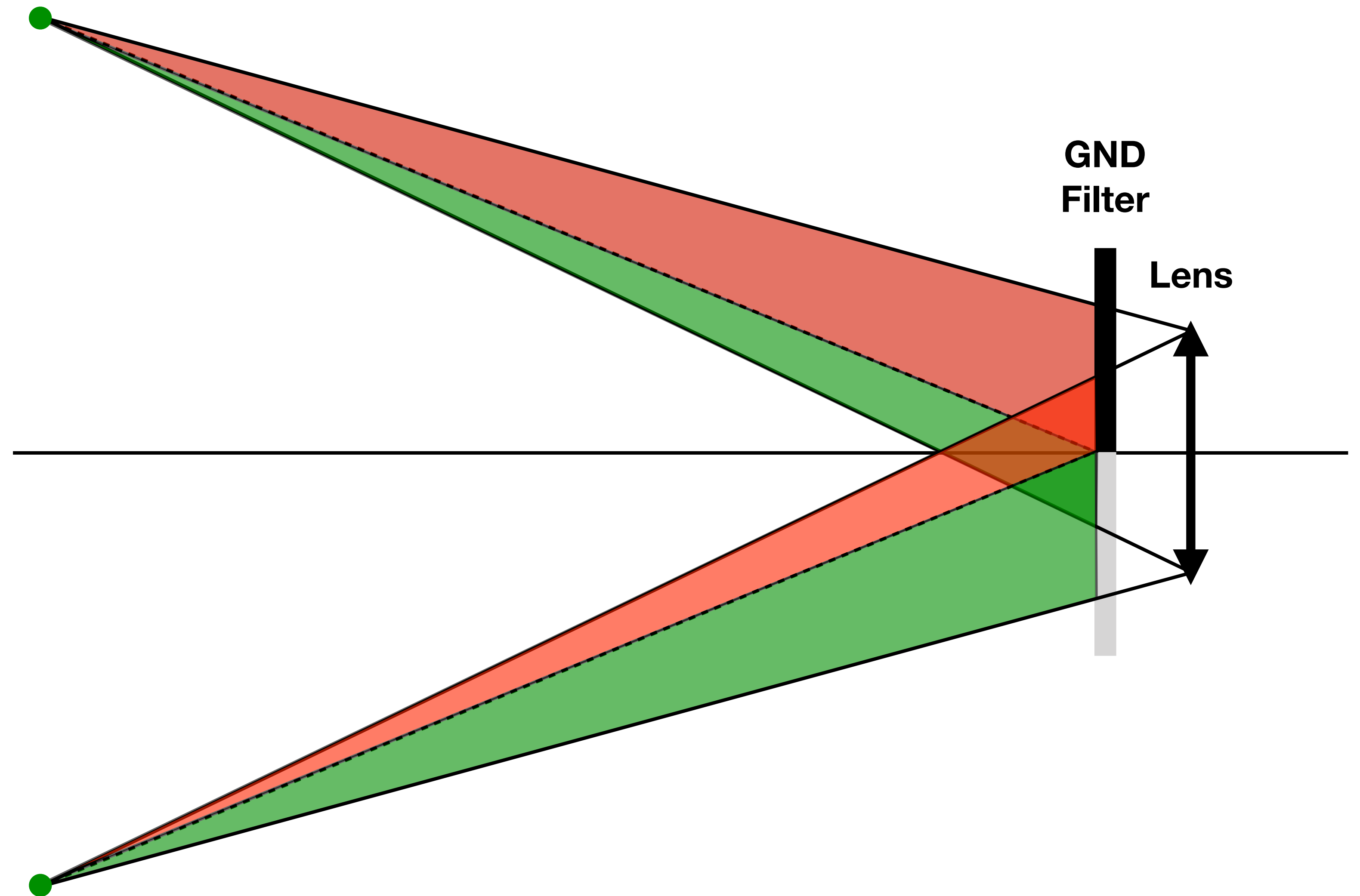
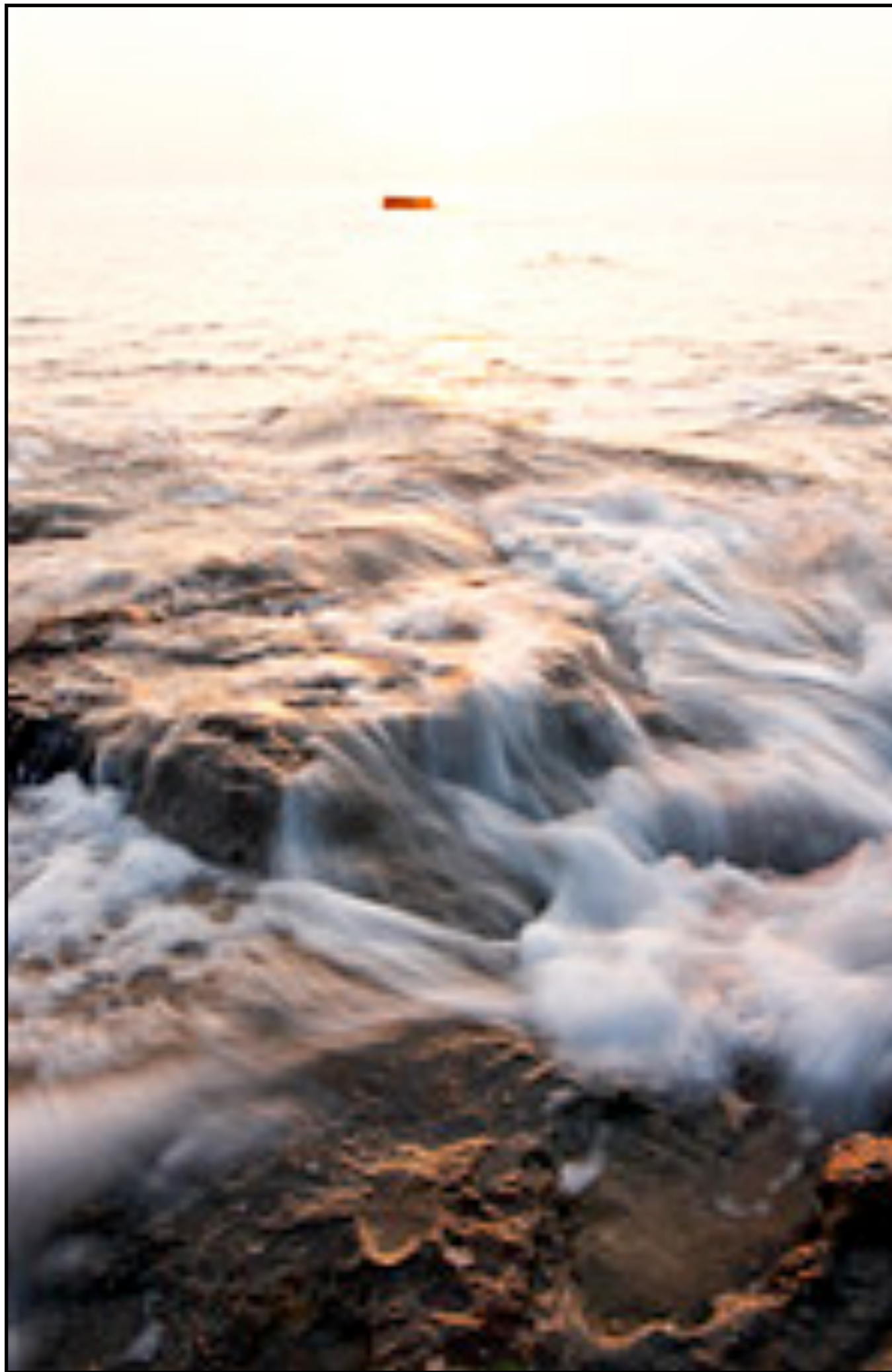


# What's Happening When Using a GDN Filter?



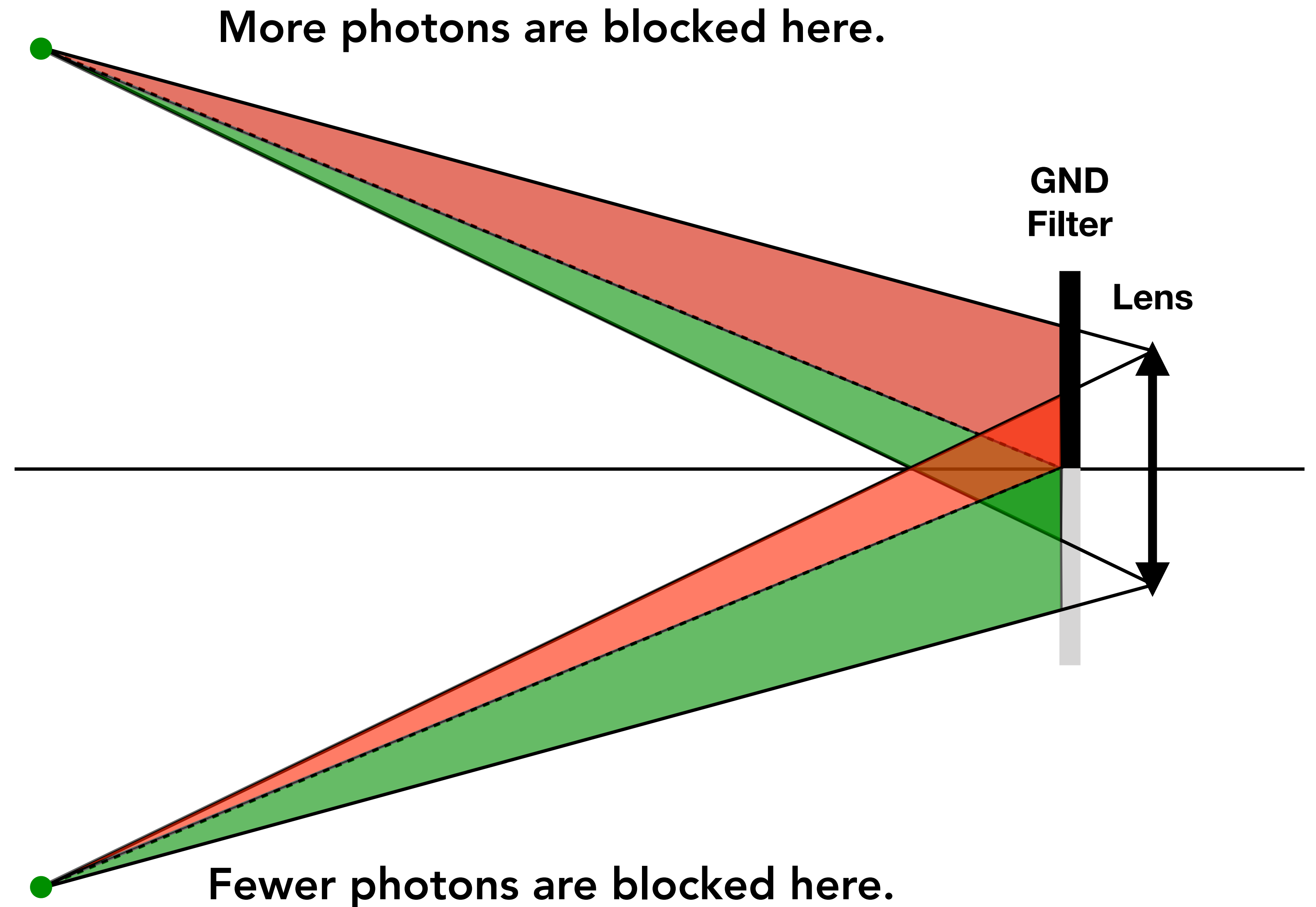
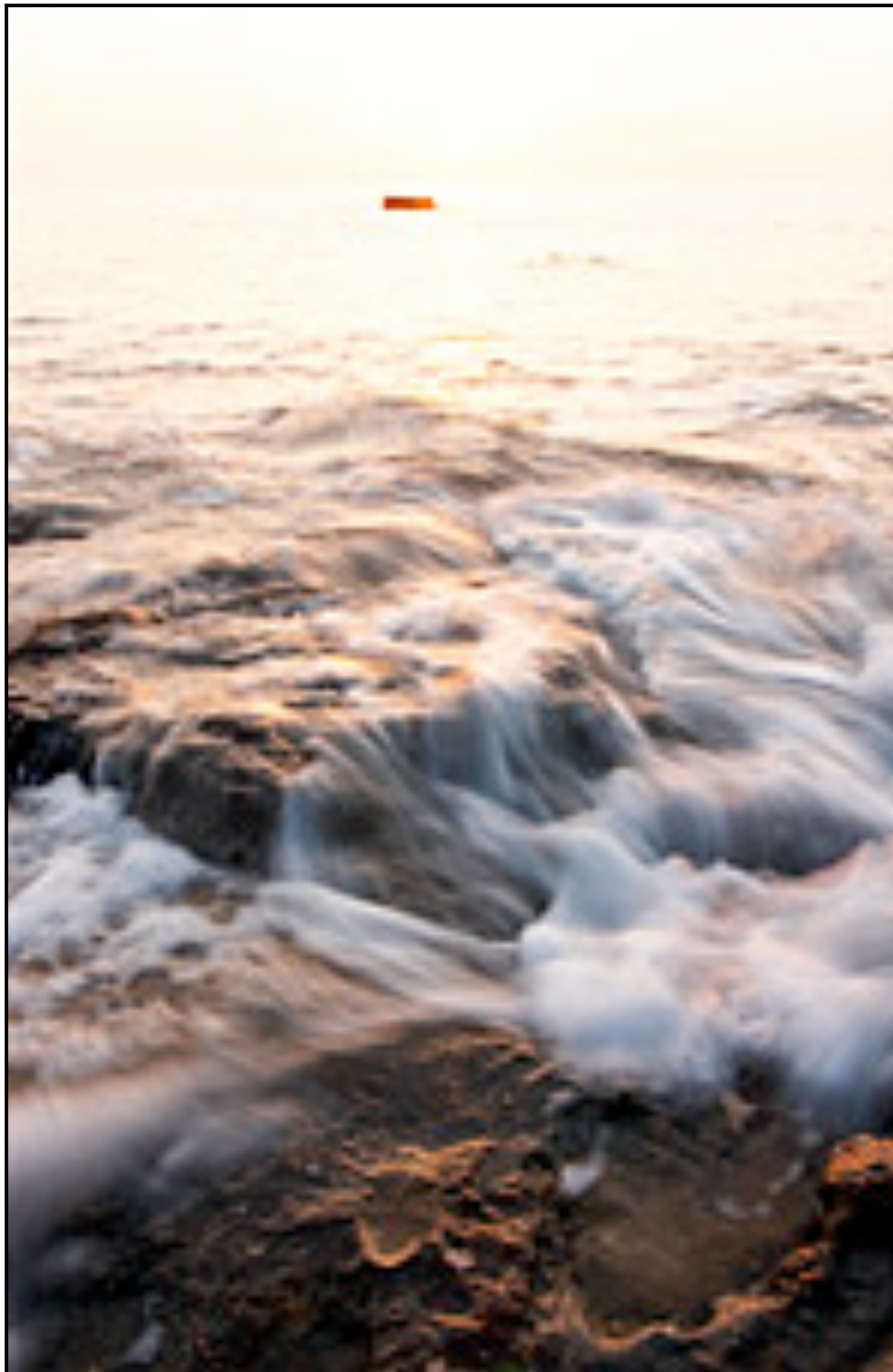


# What's Happening When Using a GDN Filter?





# What's Happening When Using a GDN Filter?





# What's Happening When Using a GDN Filter?

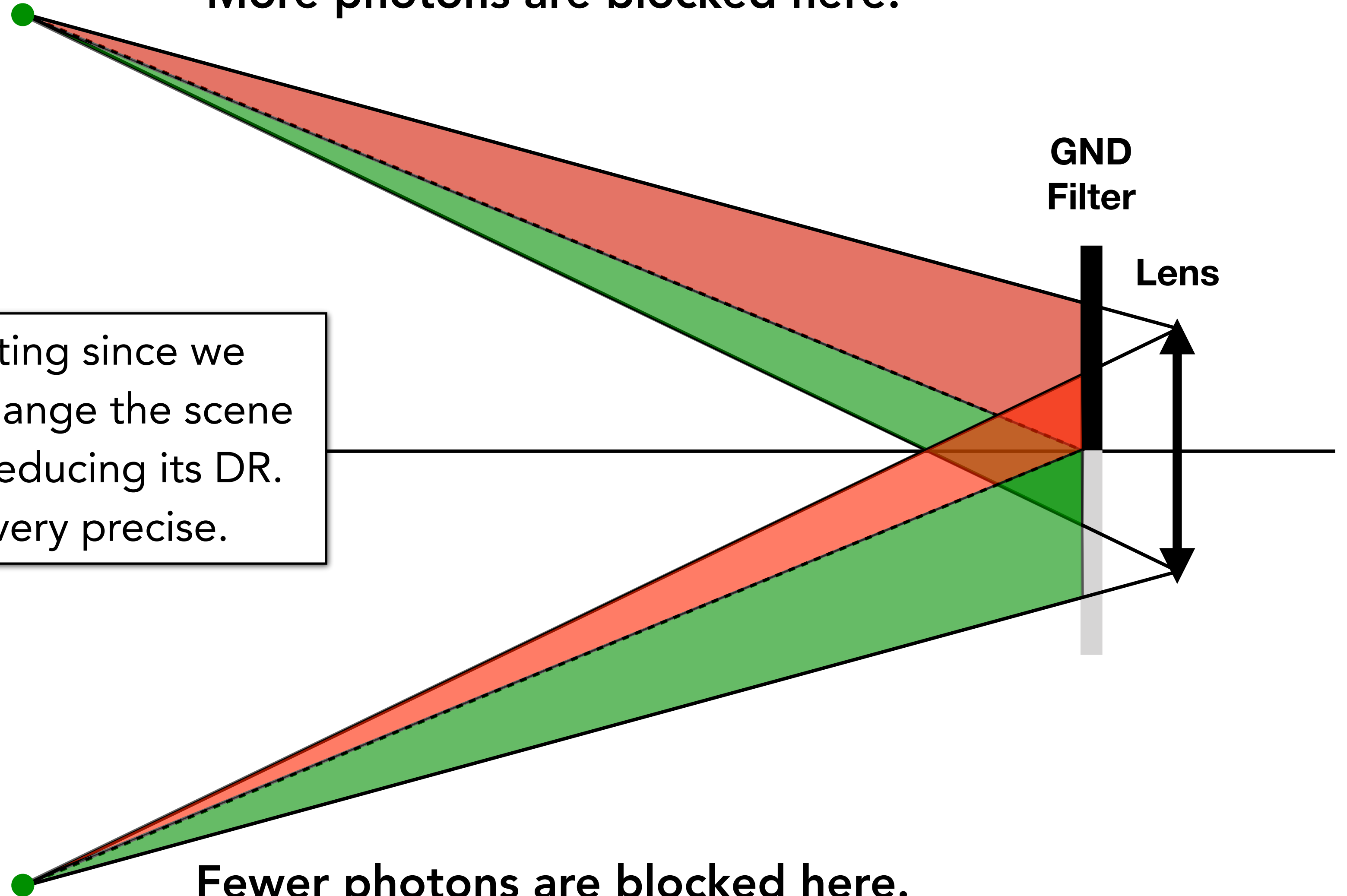
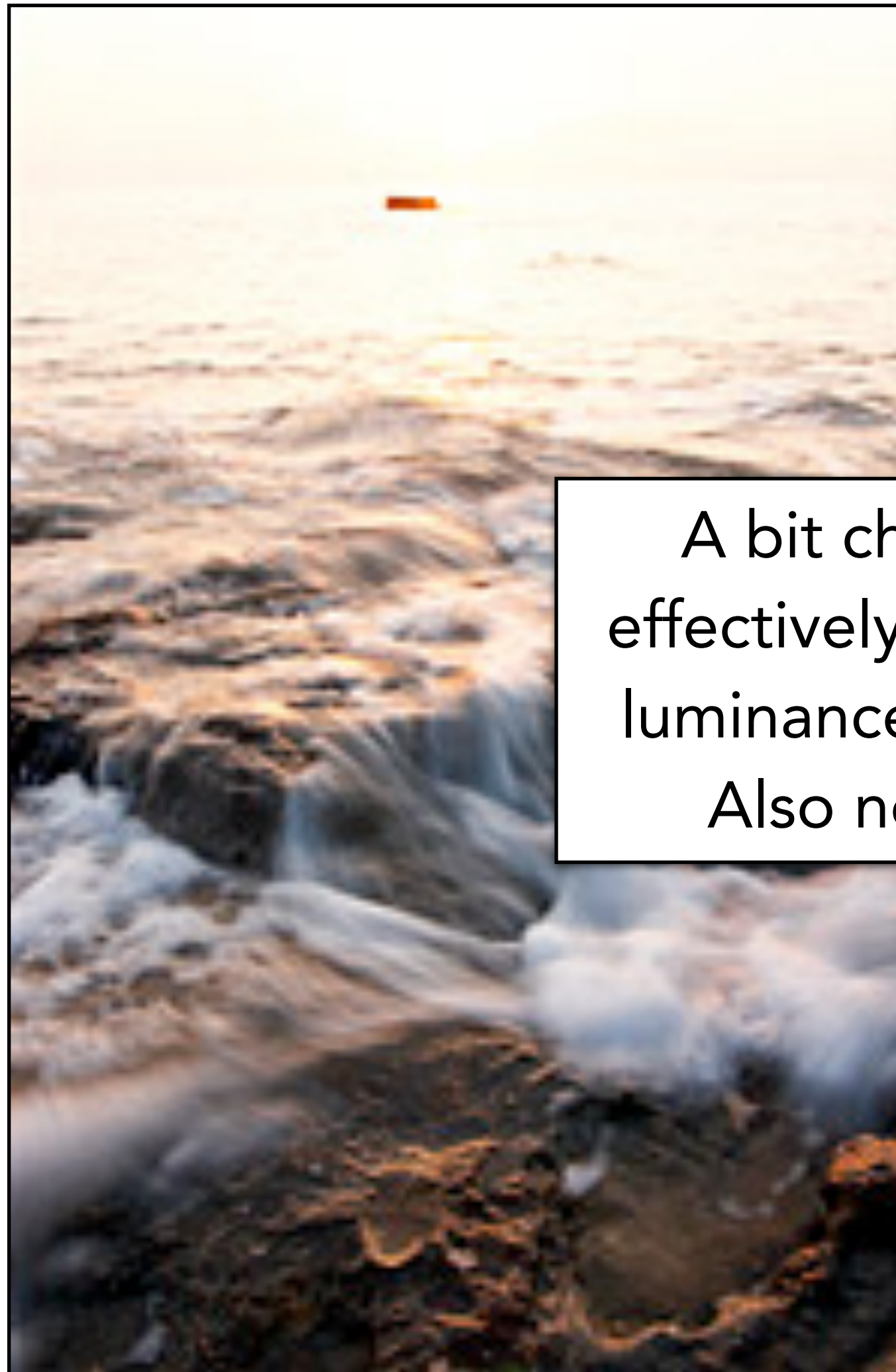
More photons are blocked here.

GND  
Filter

Lens

A bit cheating since we  
effectively change the scene  
luminance, reducing its DR.  
Also not very precise.

Fewer photons are blocked here.





# HDR Using Computation



# HDR Imaging Through Multiple Exposures

**Basic idea:** take multiple images of the scene with varying exposure times, align them, and combine them together to generate the final image.

Each exposure provides accurate luminance for *some*, not all, scene points.

Each scene point will be mapped to a pixel in each exposure. The most accurate luminance information of the point is from a pixel that's well exposed but not over-exposed.

Better yet, we can average across images to reduce noise.





HYATT

Grand Plaza



















# A Simple Algorithm (in Linear RGB Space)



Since raw value and photon quantity are linear, can think of 80 as 80 photons.

● Photon/second:  $160/2 = 80$

★ Photon/second:  $((38*4+162)/8)/2 = 19.6$

● New pixel value = 80, ignore 255 from the second image since it's saturated and can't be used to derive the luminance information.

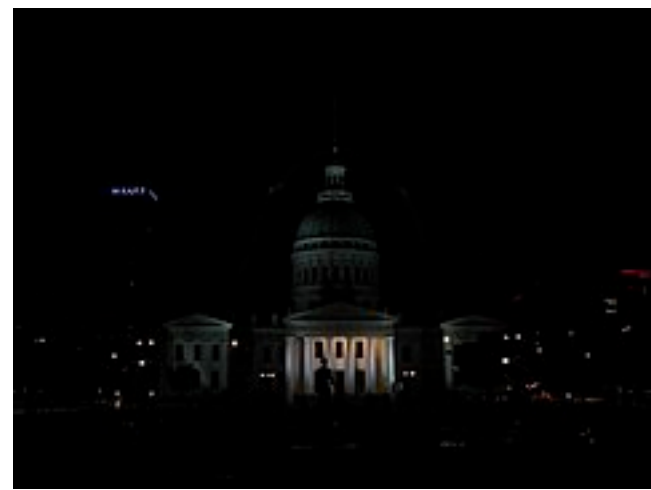
★ New pixel Value =  $((38*4+162)/8)/2 = 19.6$ . Pixel values from both images can be used to derive the scene luminance; averaging them reduces noise.

For each pixel, if not black nor saturated, convert to photon/second, and then average across frames. In the end, we get the relative luminance for all pixels.



# A Simple Algorithm (in Linear RGB Space)

1s — 2X → 2s — 4X exposure time → 8s — 2X → 16s



160 255  
38 162



Can conceptually extend to multiple exposures, and apply the same process to all three color channels.

● New pixel value = 80, ignore 255 from the second image since it's saturated and can't be used to derive the luminance information.

★ New pixel Value =  $((38 \cdot 4 + 162) / 8) / 2 = 19.6$ . Pixel values from both images can be used to derive the scene luminance; averaging them reduces noise.

For each pixel, if not black nor saturated, convert to photon/second, and then average across frames. In the end, we get the relative luminance for all pixels.



# Multiple Exposures on sRGB Images

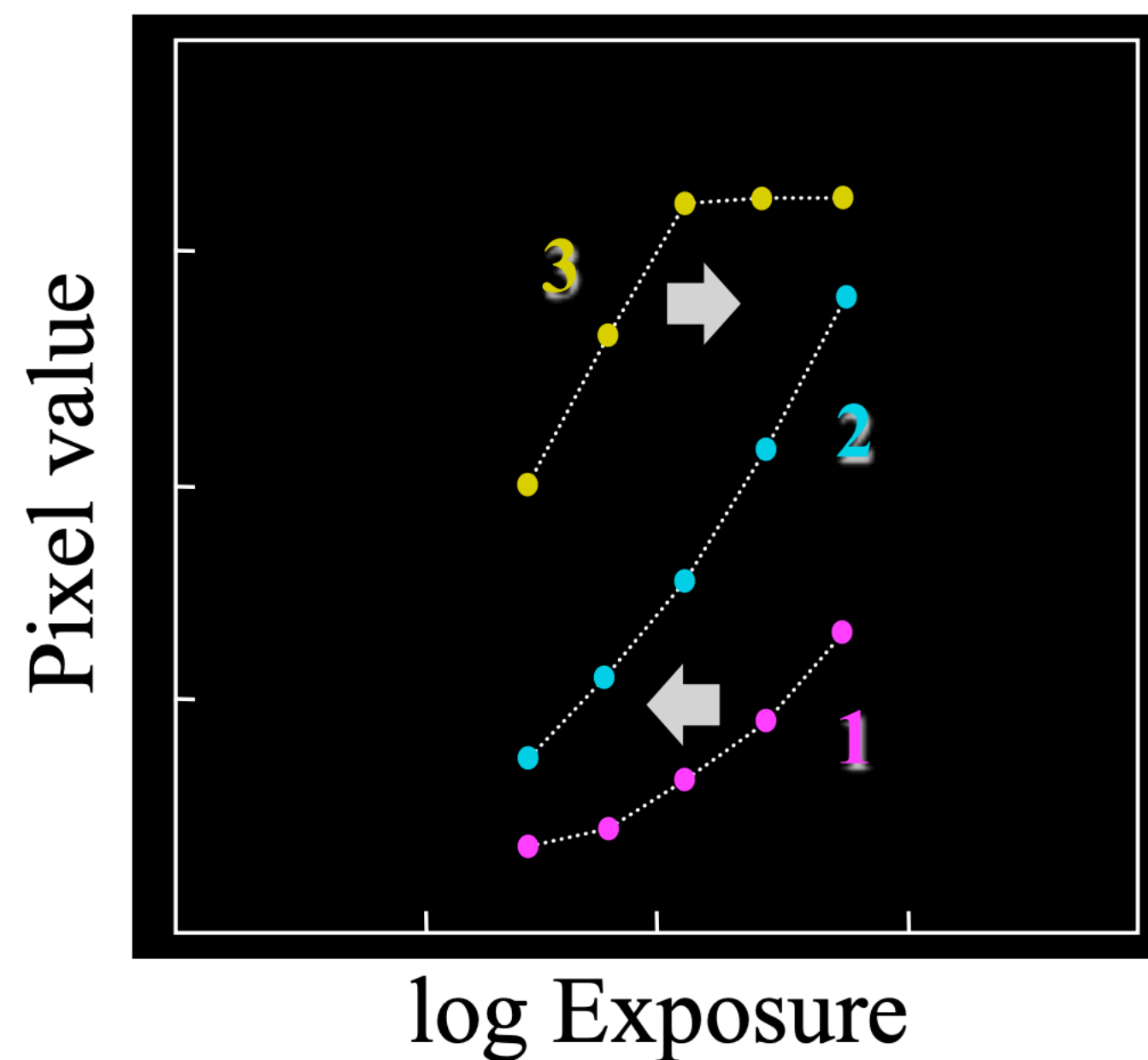
When working with sRGB images the luminance to pixel value curve is non-linear (lots of signal processing in-between). Fit a curve to find the relationship.

Other issues:

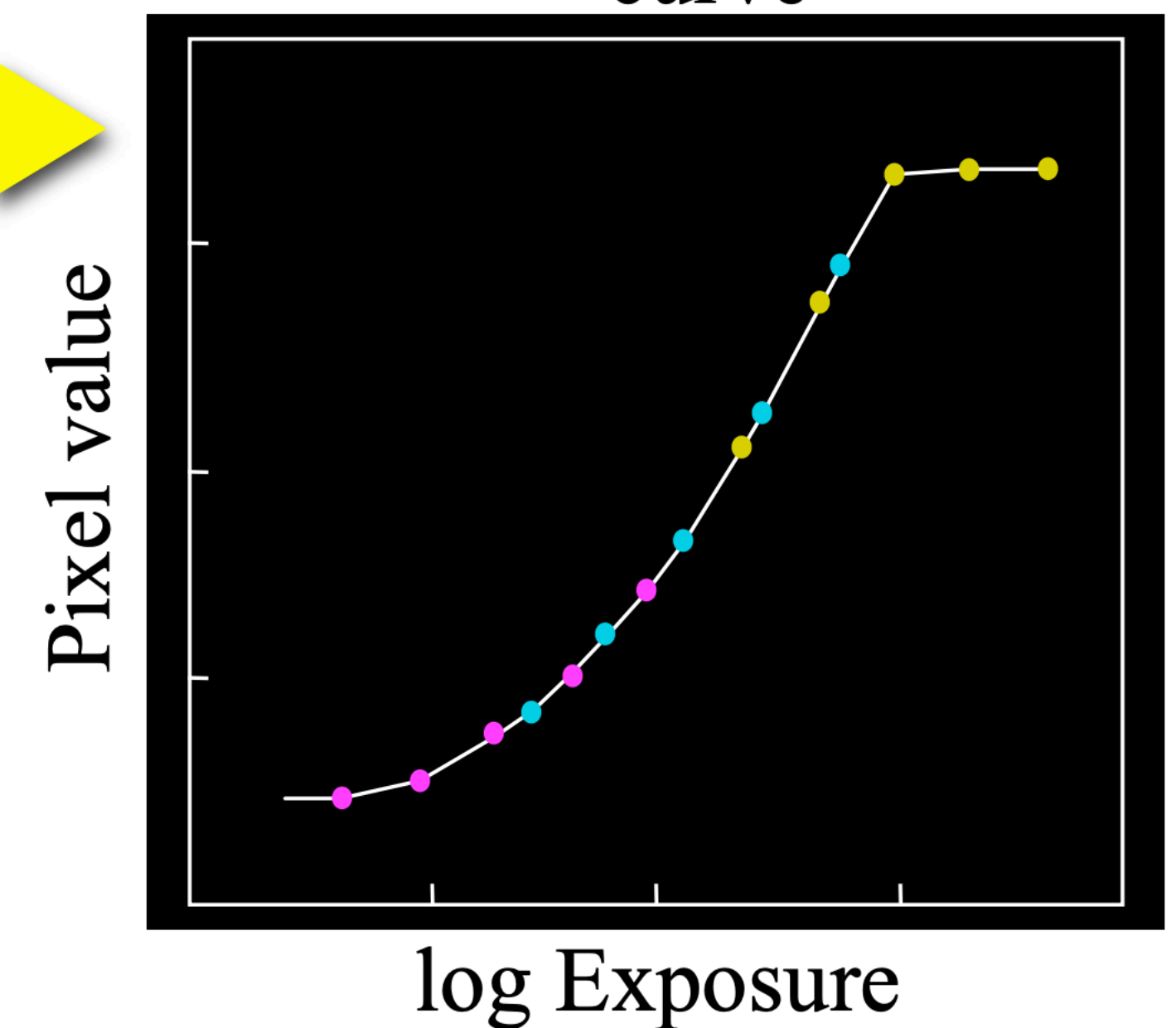
- Need to align images.
- Subjective to motion blur.

- **Exposure is unknown, fit to find a smooth curve**

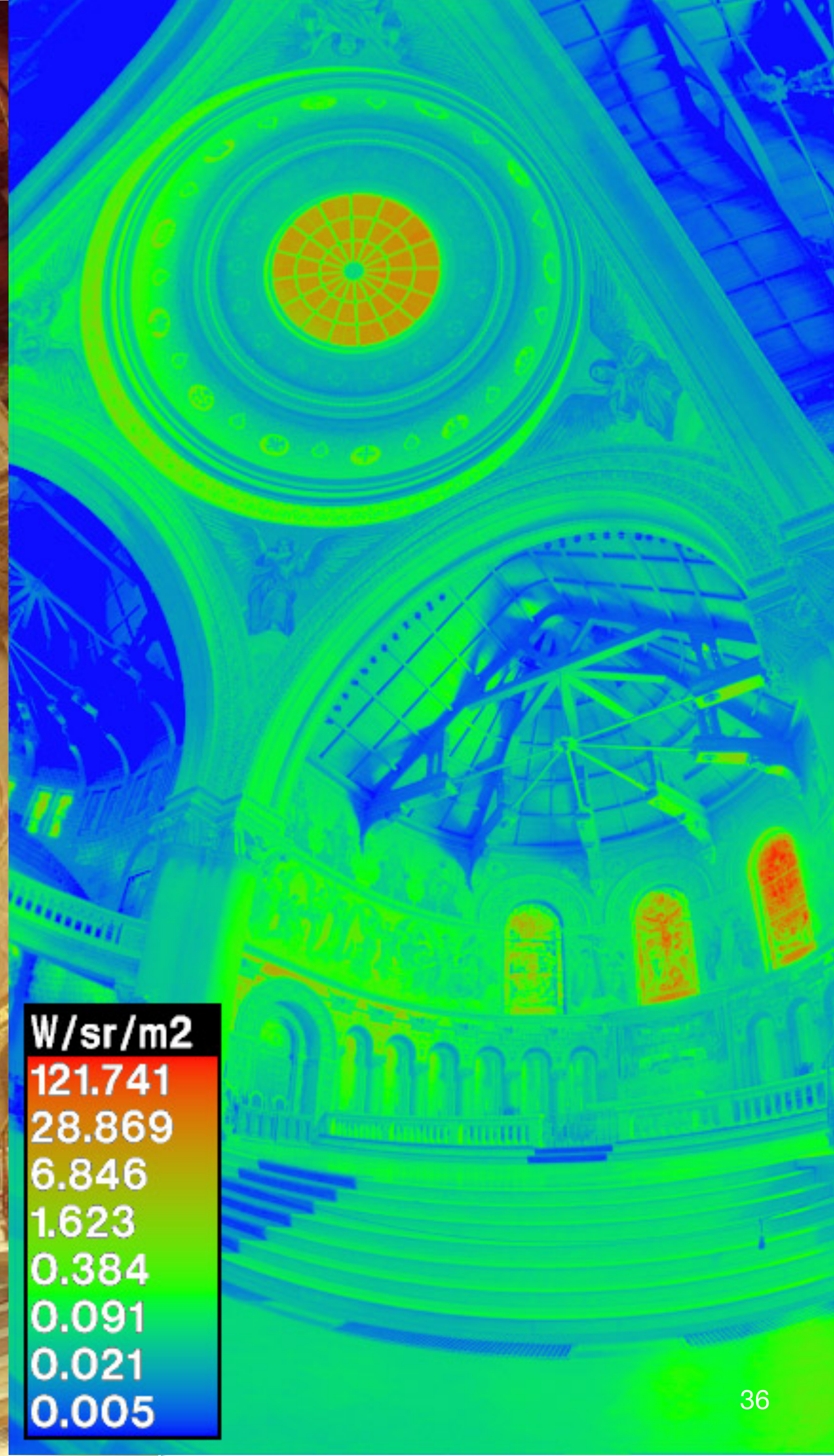
Assuming unit radiance for each pixel



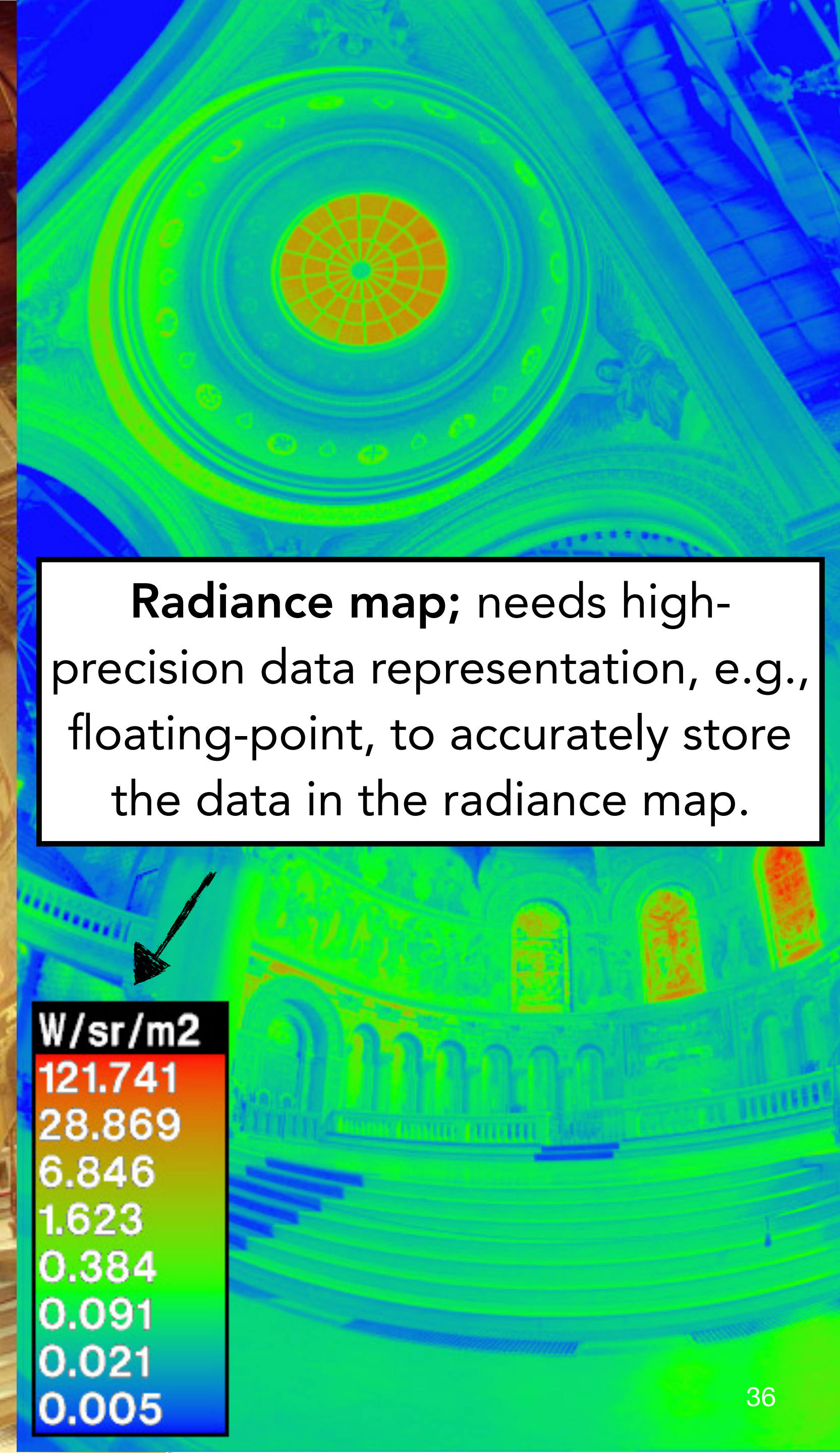
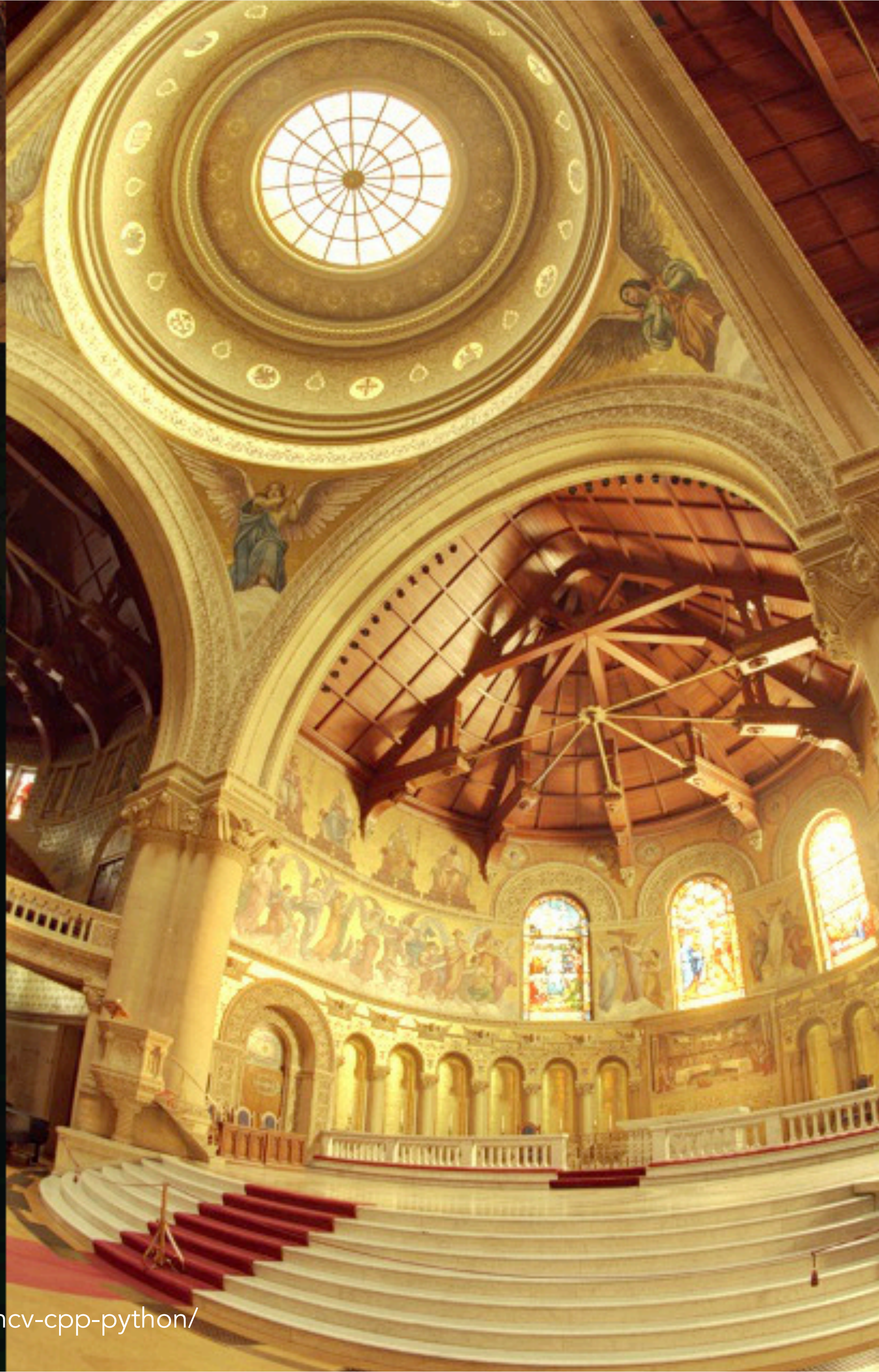
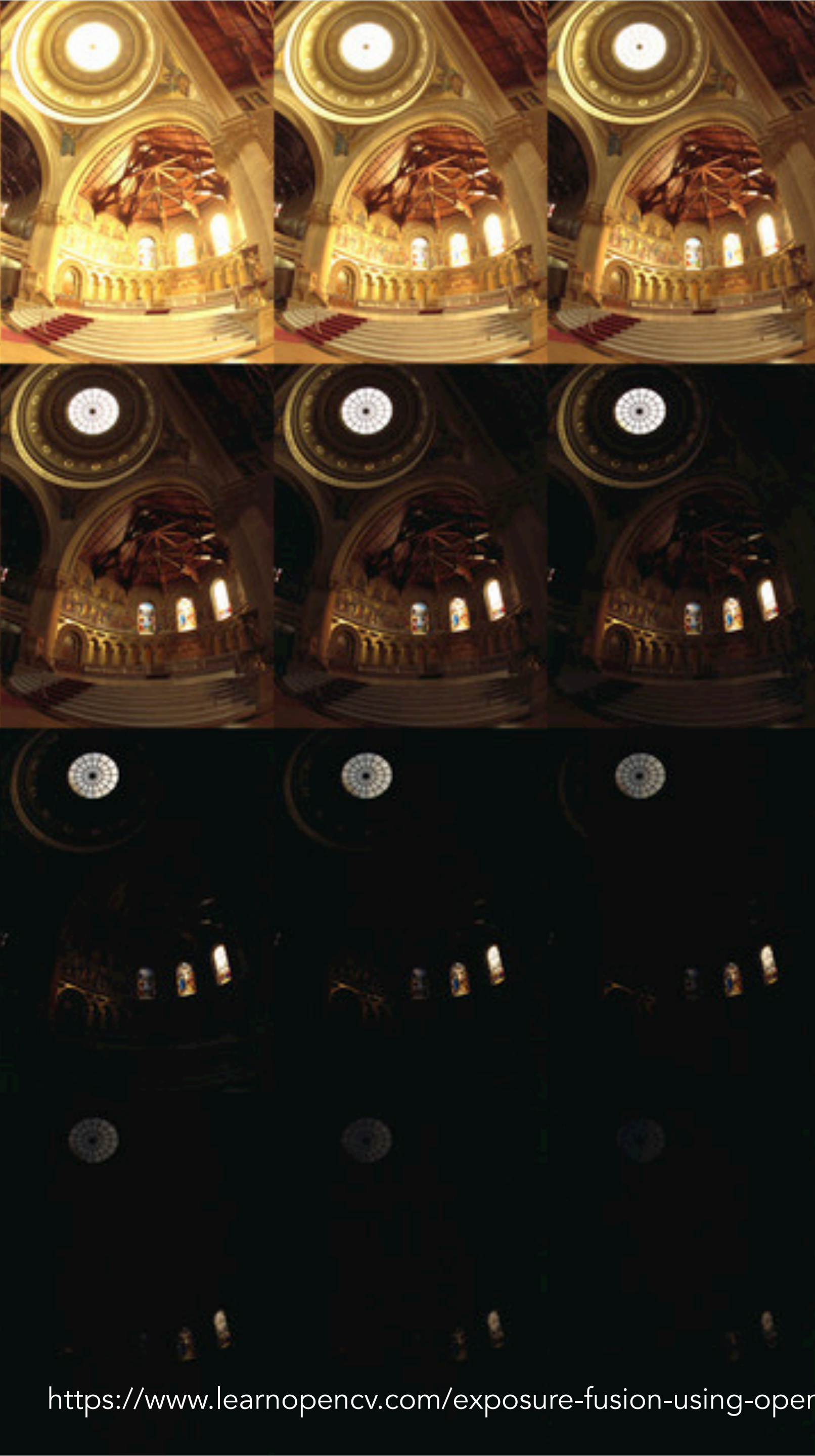
After adjusting radiances to obtain a smooth response curve









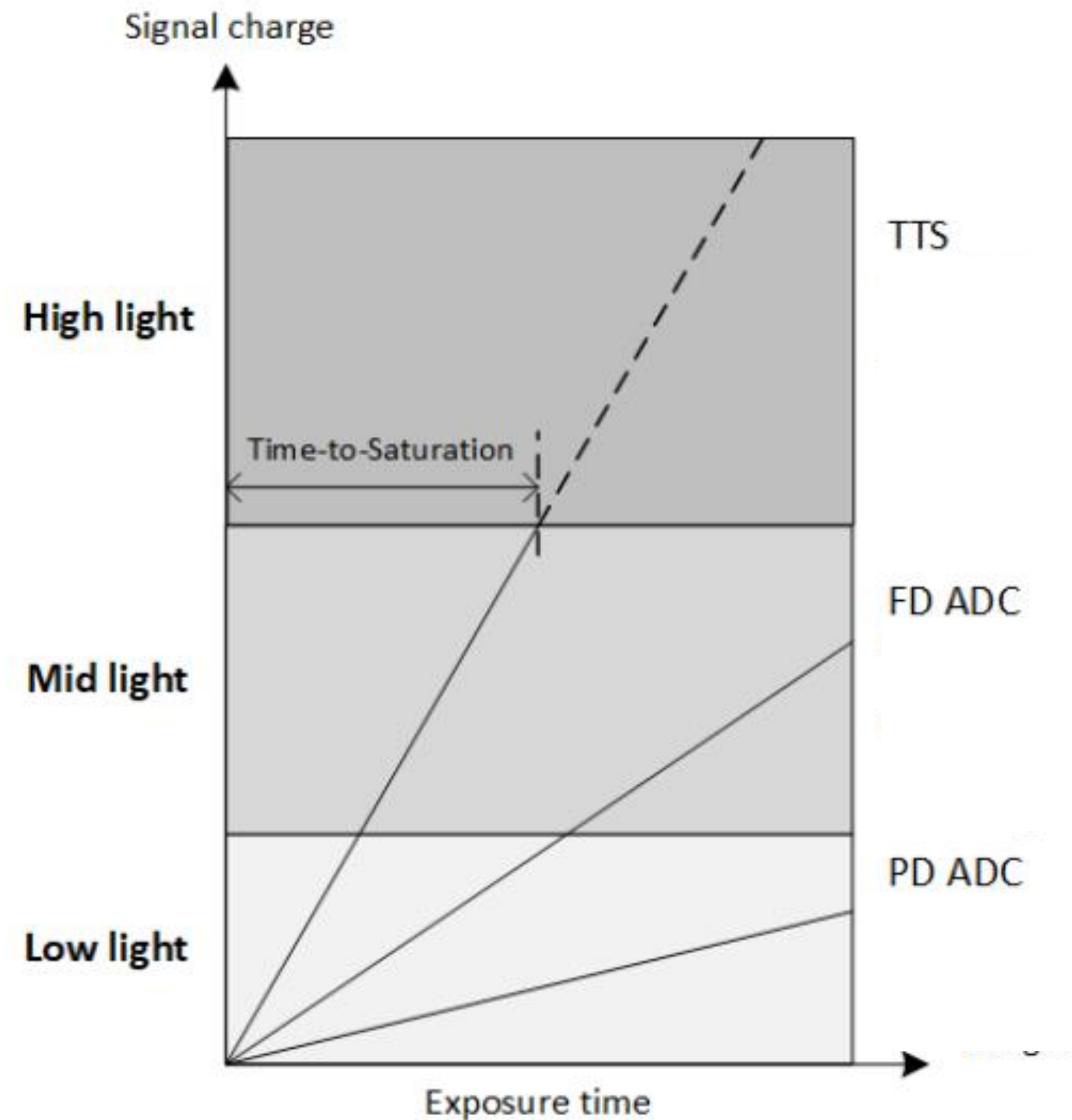




# HDR in Sensor



# Sensor Support for HDR



## Three main ideas:

- Use bigger pixels. Trade-off between spatial resolution, noise, and DR.
- Time-to-saturation (illustration on the right): measure the time to saturation and extrapolate the photon counts.
- Staggered HDR (next slide): multi-exposure fusion in hardware.



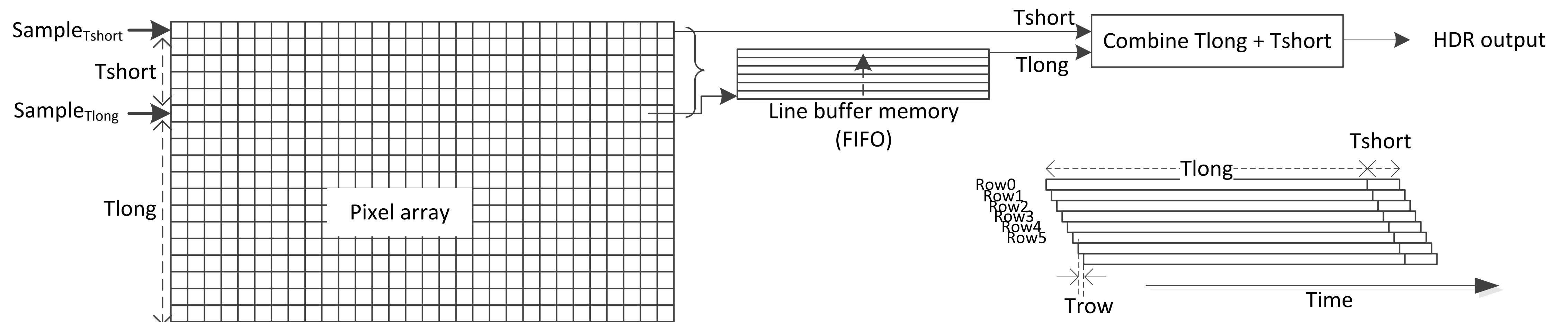
# Staggered HDR

Exposes each row in **two exposures** in succession, **long** first then **short**.

- Store long-exposed pixel lines in a temporary buffer.

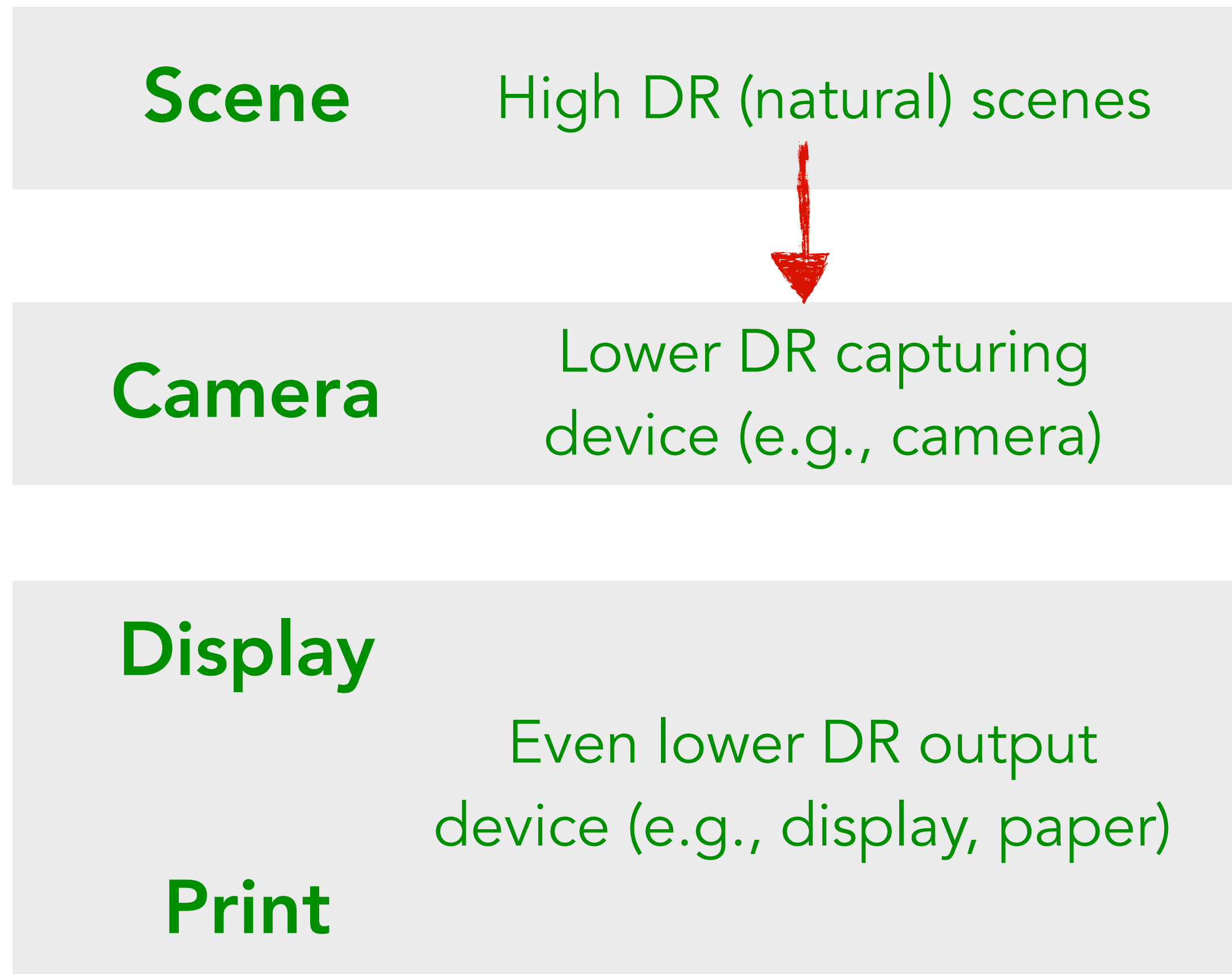
As soon as the short exposure of a row finishes, combine it with the long-exposure data of the same row from the buffer.

Shorted gap between multiple (2 here) exposures.





# HDR Imaging Recap



Key goal is to represent the relative luminance of scene points accurately across the entire luminance range.

In principle, it's easy to combine multiple exposures on raw pixels (linear w.r.t. luminance). Harder on sRGB images (not linear).

- Averaging across exposures also serves **denoising**. GND can be used when accurate relative luminance isn't required.



# HDR Imaging Recap

**Scene** High DR (natural) scenes

**Camera** Lower DR capturing device (e.g., camera)

**Display**



**Tone Mapping**

Even lower DR output device (e.g., display, paper)

**Print**

Key goal is to represent the relative luminance of scene points accurately across the entire luminance range.

In principle, it's easy to combine multiple exposures on raw pixels (linear w.r.t. luminance). Harder on sRGB images (not linear).

- Averaging across exposures also serves **denoising**. GND can be used when accurate relative luminance isn't required.



# Tone Mapping



# Tone Mapping: Problem 1

HDR imaging allows us to accurately capture the full dynamic range of the scene, which could be ~20 stops.

Typical Display: ~10 stops (LG 27UK850): the ratio between the brightness pixel and the darkest pixel is about 1024 under the same display power. You can dim the display, but that weakens all pixels and the DR is still the same.

- HDR Display: 14.2 stops (UHD Alliance Certified HDR LCD display)
- Paper: 6-8 stops (typical value); related to pigment density

So we need to compress the contrast of ~100,000:1 to ~1000:1.

- Can we simply linearly compress the range?



# Tone Mapping: Problem 2

Natural scenes contain *continuous* luminance levels, whereas digital images inherently quantize continuous luminance levels to *discrete* levels.

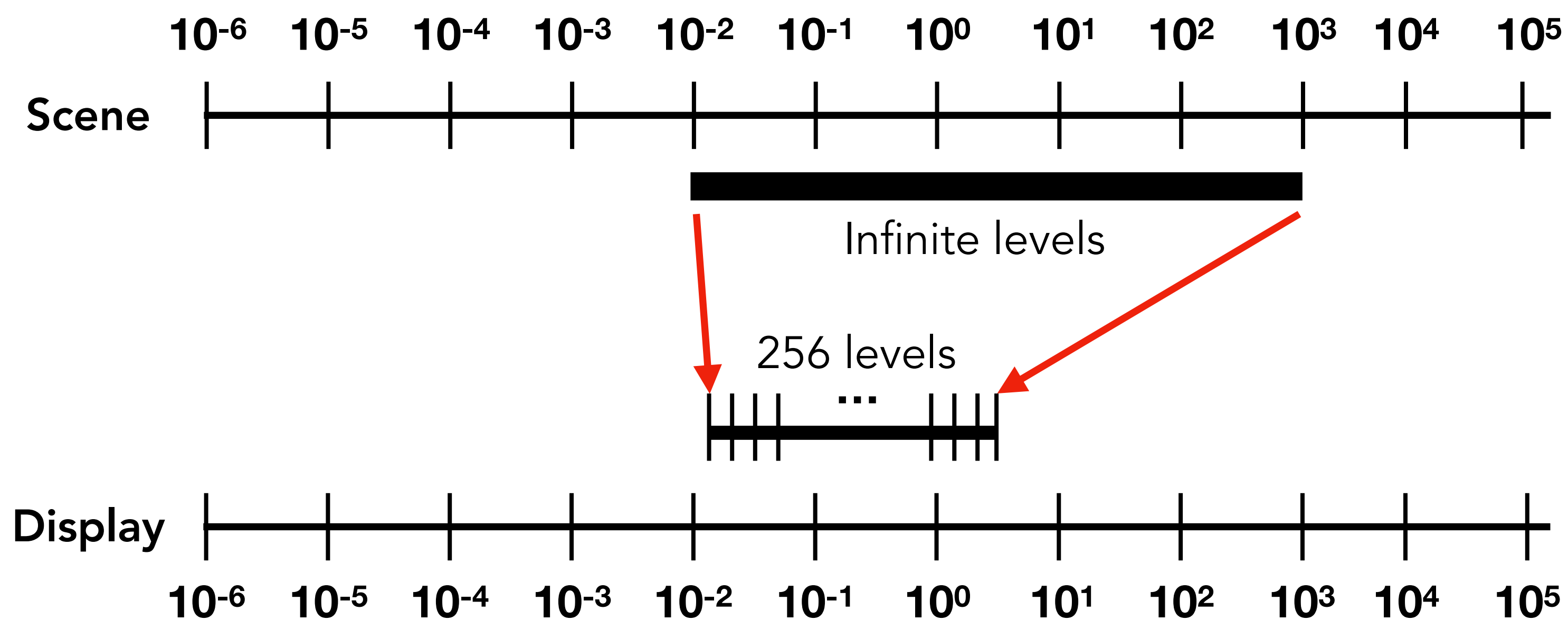
If an image is encoded in 8-bit sRGB, only 256 levels are available to represent the luminance levels.

- As a result, subtle contrast (luminance difference) might end up being quantized to the same level, losing the details when being displayed.

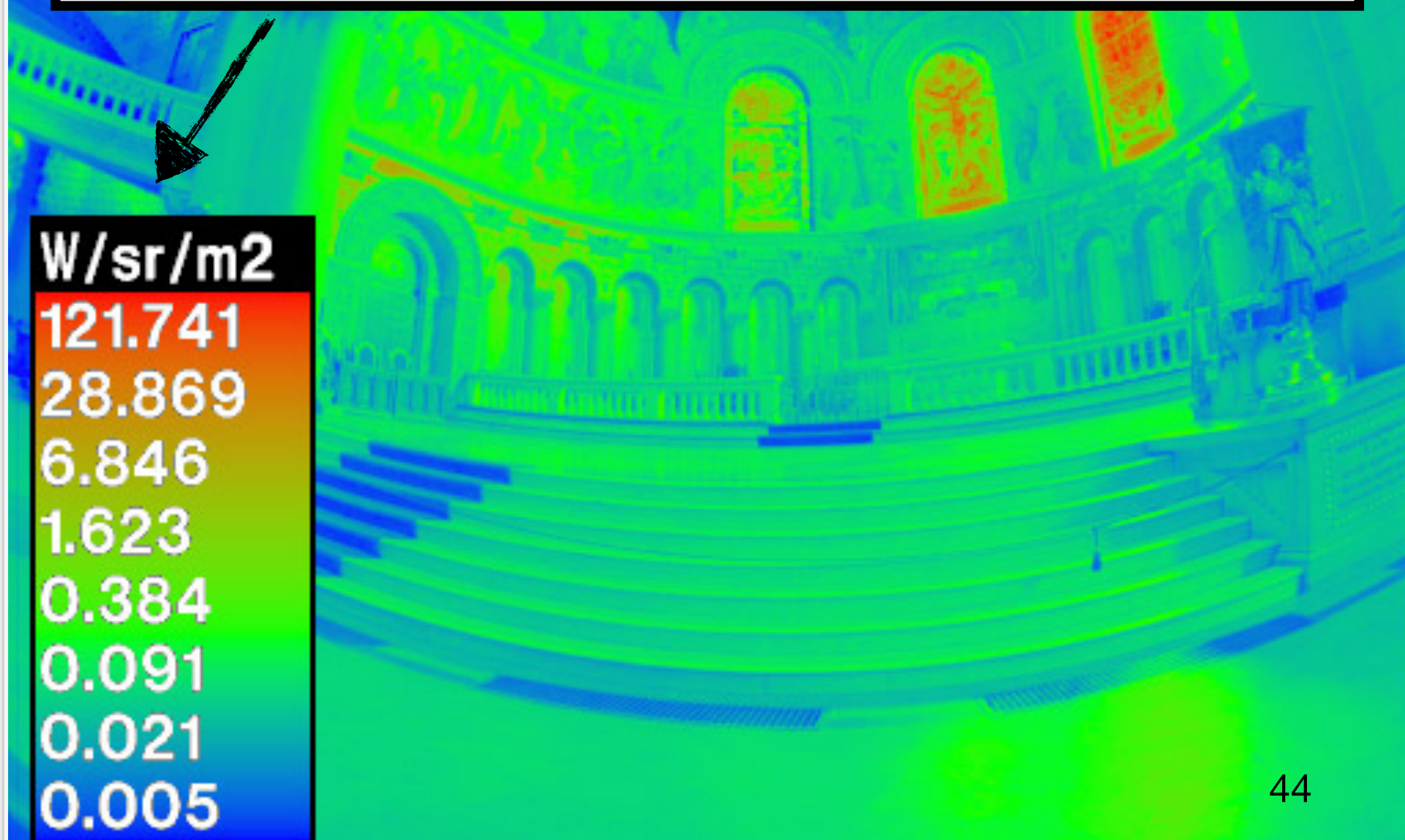
Encoding color using higher bit-depth, e.g., 10 bits, alleviates the second problem but doesn't affect/solve the first problem at all.



Some information is guaranteed to be lost. The goal is to exploit characteristics of human visual system to generate images that mimic human's expected visual response.



How to represent the full range of these numbers using only 8 bits per color channel, as required by sRGB (and most displays), while being visually pleasing?





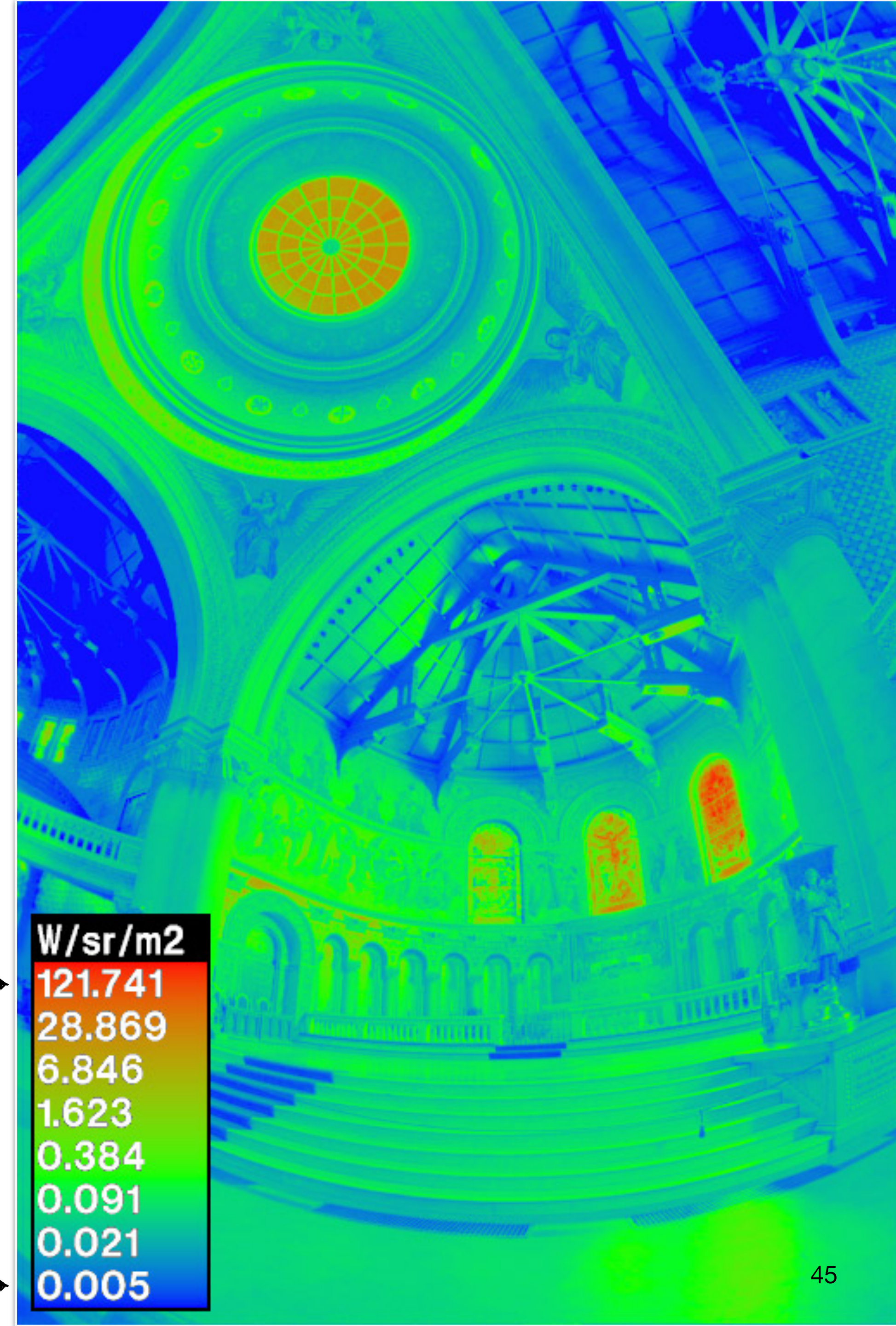
# Linear Mapping

Map the highest luminance pixel to 255 and lowest to 0. What do you think will happen for this image?

255



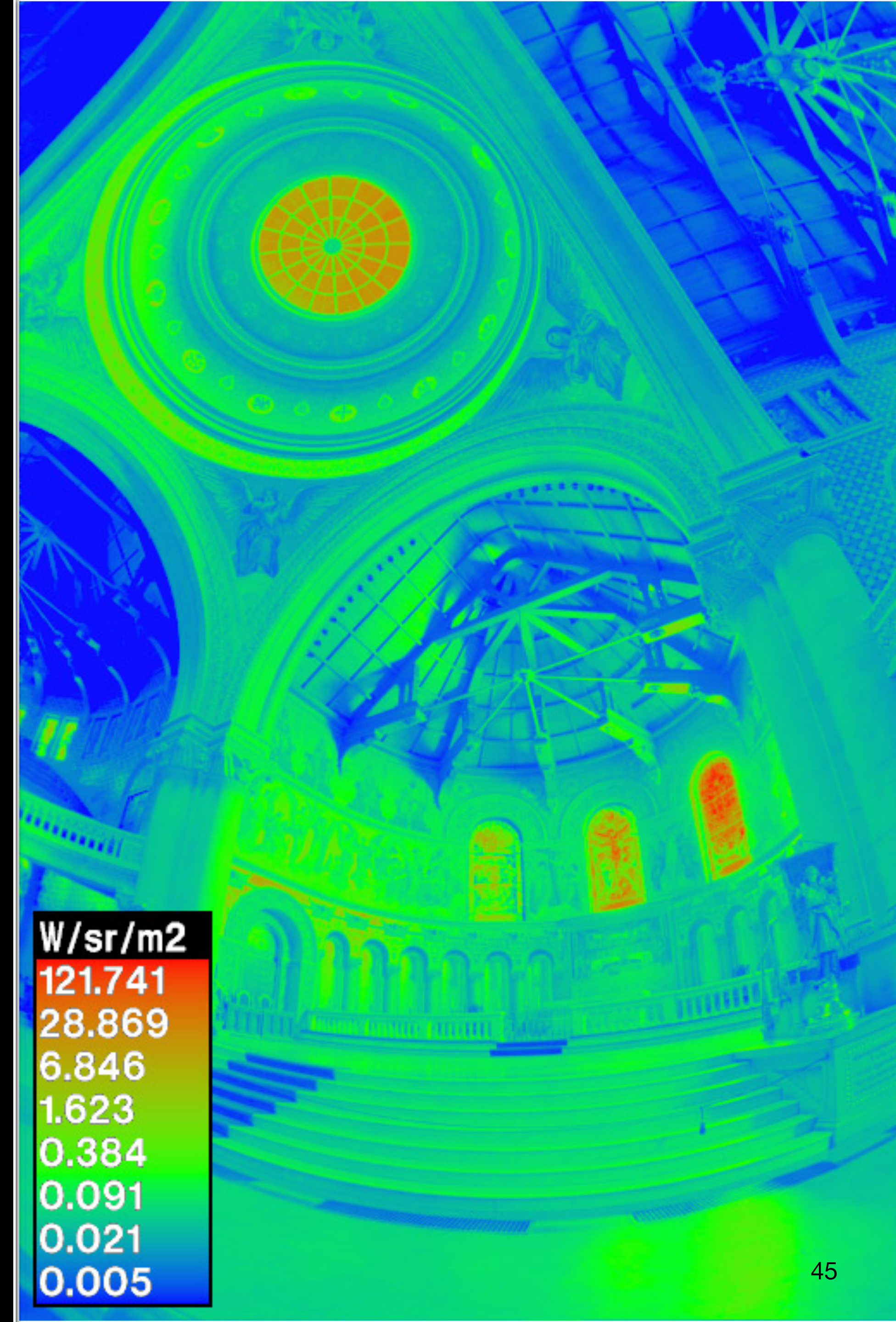
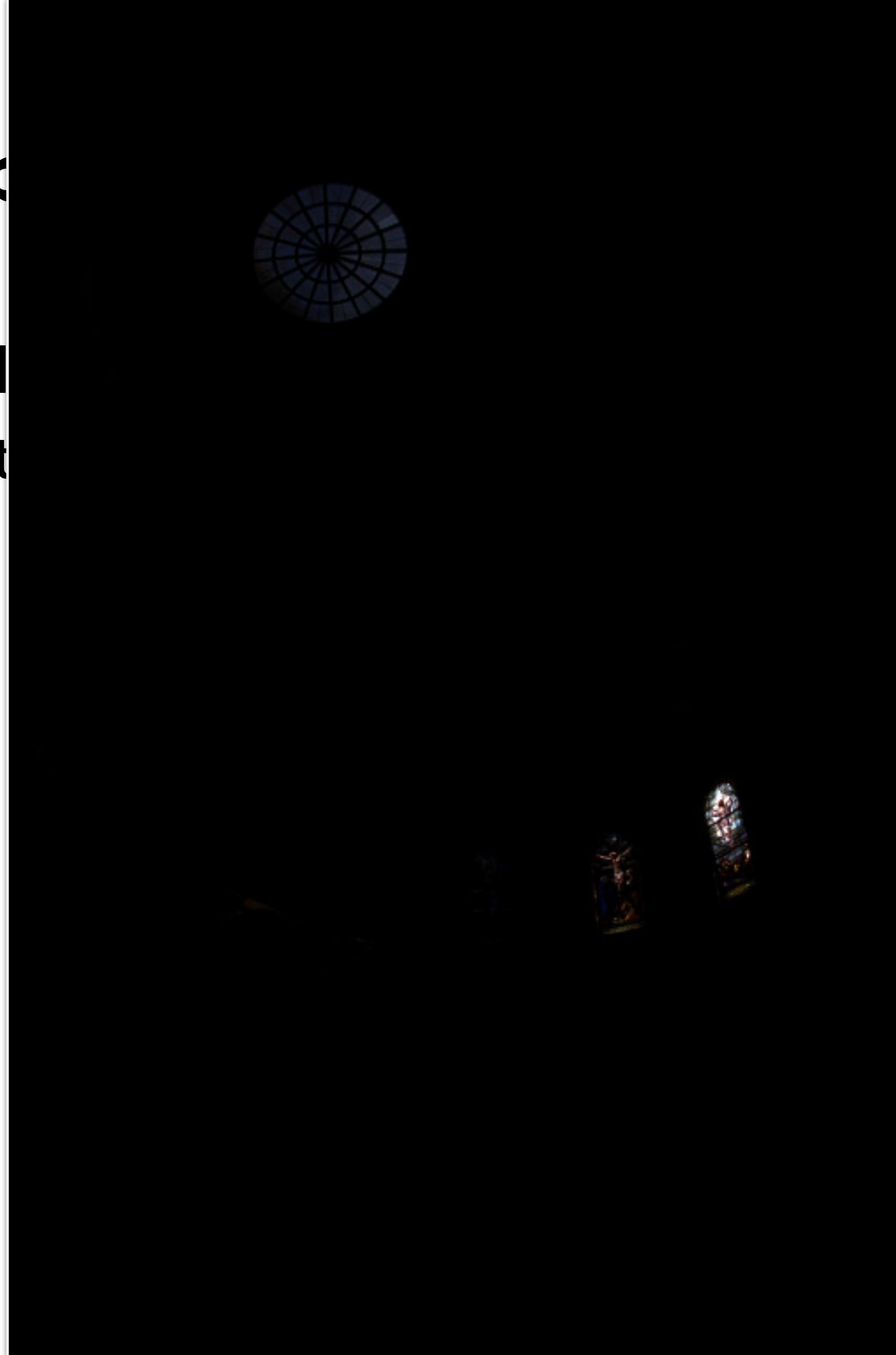
0





# Linear Mapping

Map the highest to lowest to 0. What for this image?

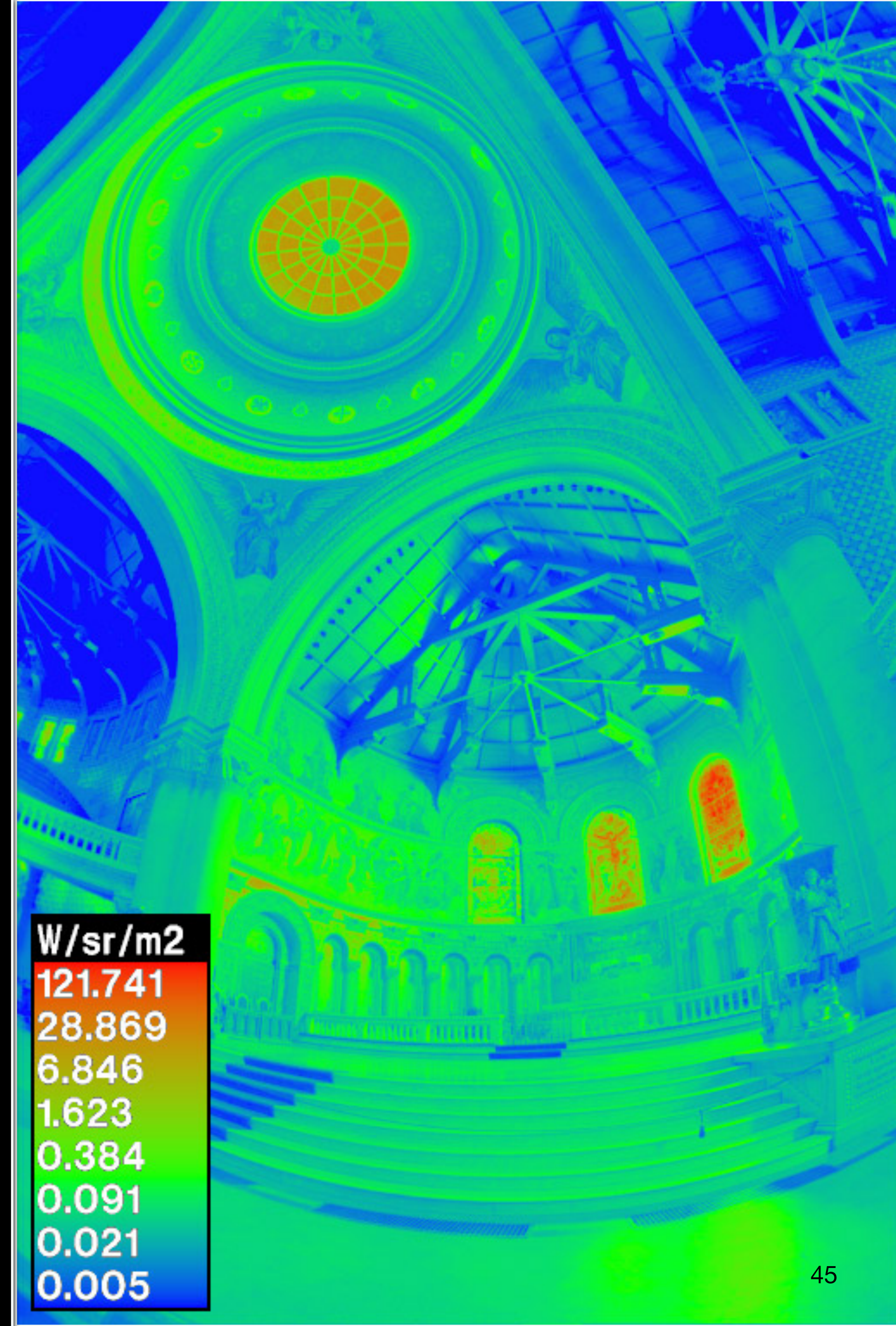




# Linear Mapping

Map the highest luminance to 1 and the lowest to 0. What does this look like for this image?

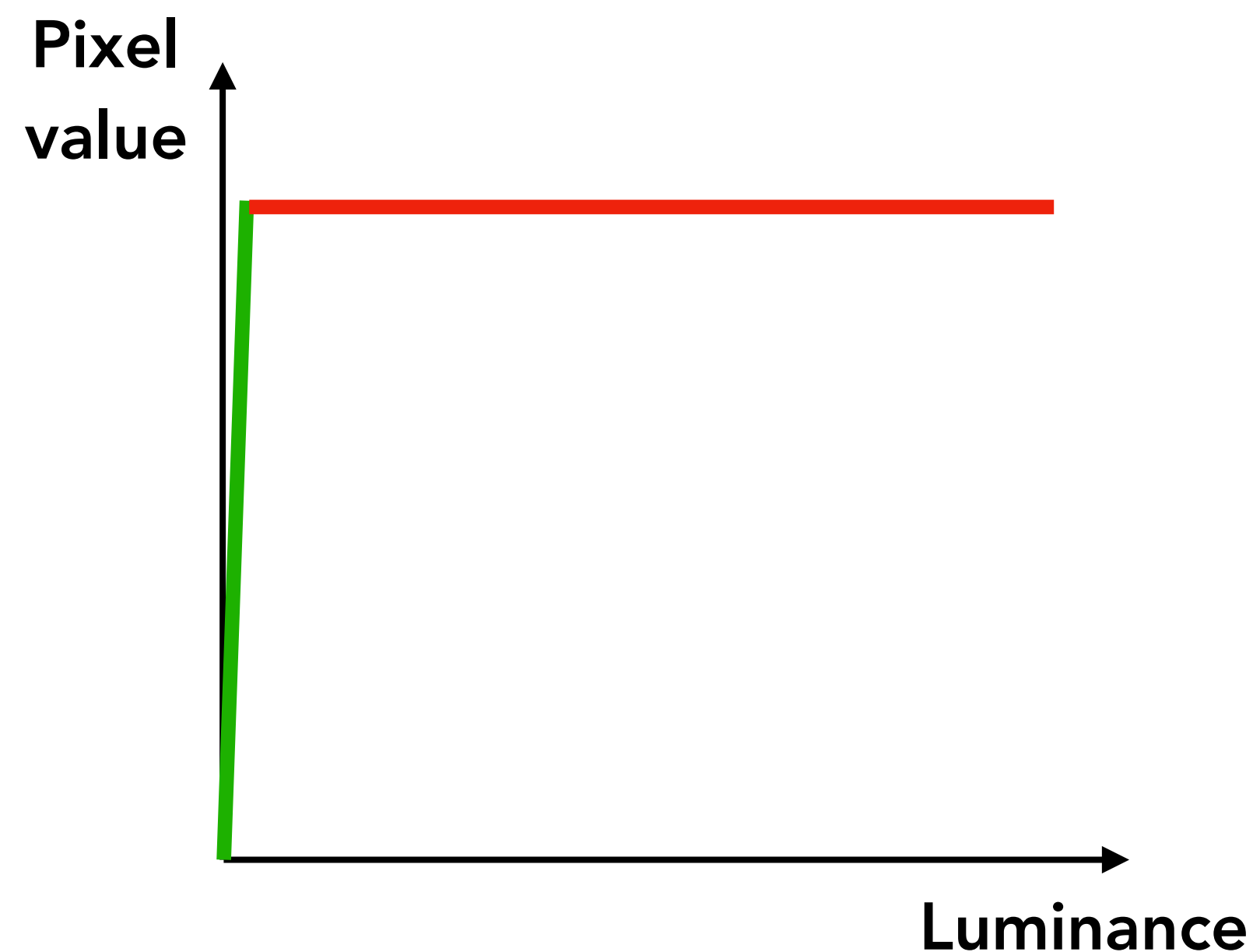
Why? Most of the luminance levels are clustered toward the low end.



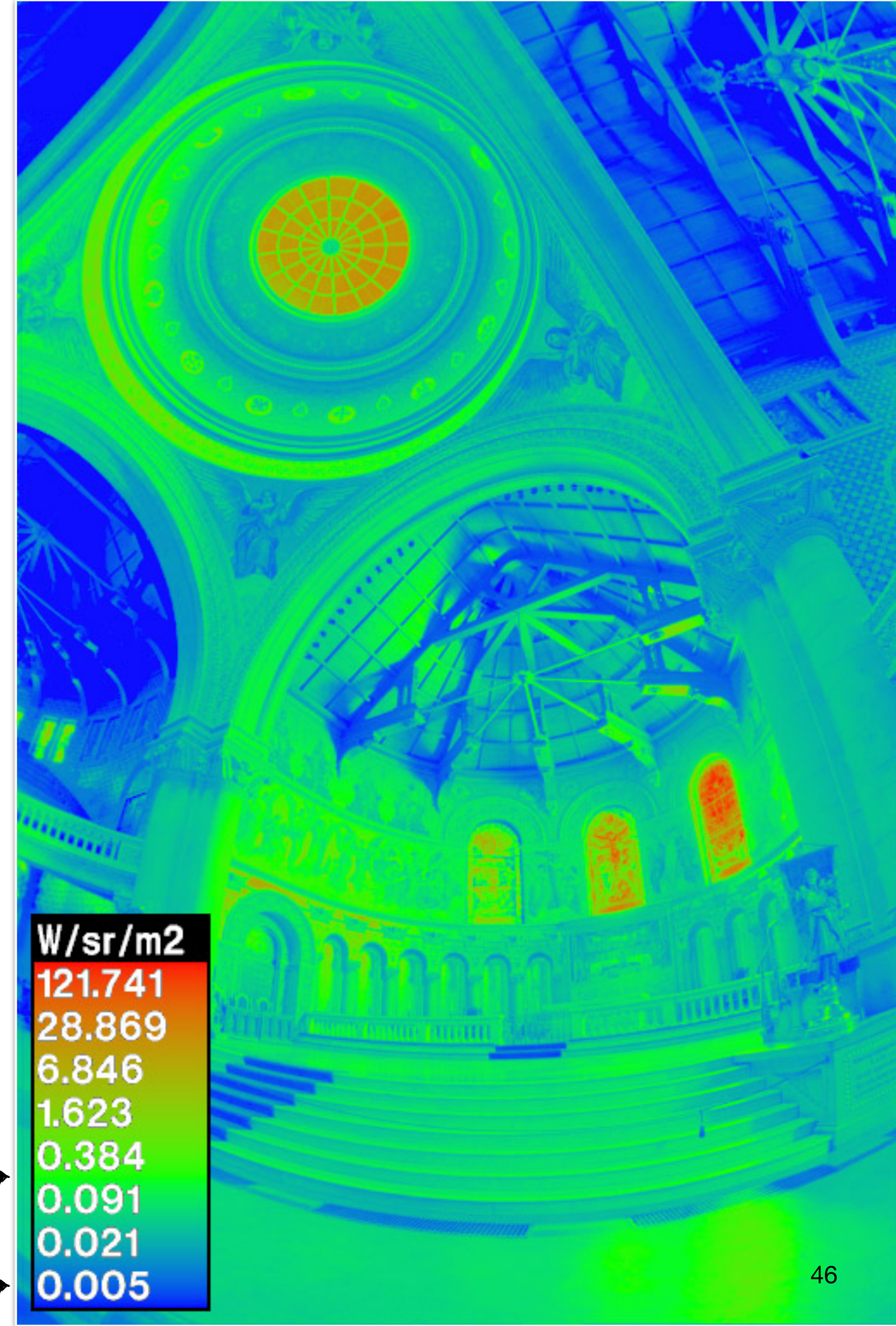


# Another Try of Linear Mapping

Linearly map the darkest 0.1% luminance in the DR to the  $[0, 255]$  range, and blow out the rest (i.e., 255 for the rest). What do you think will happen for this image?



255 →  
0 →

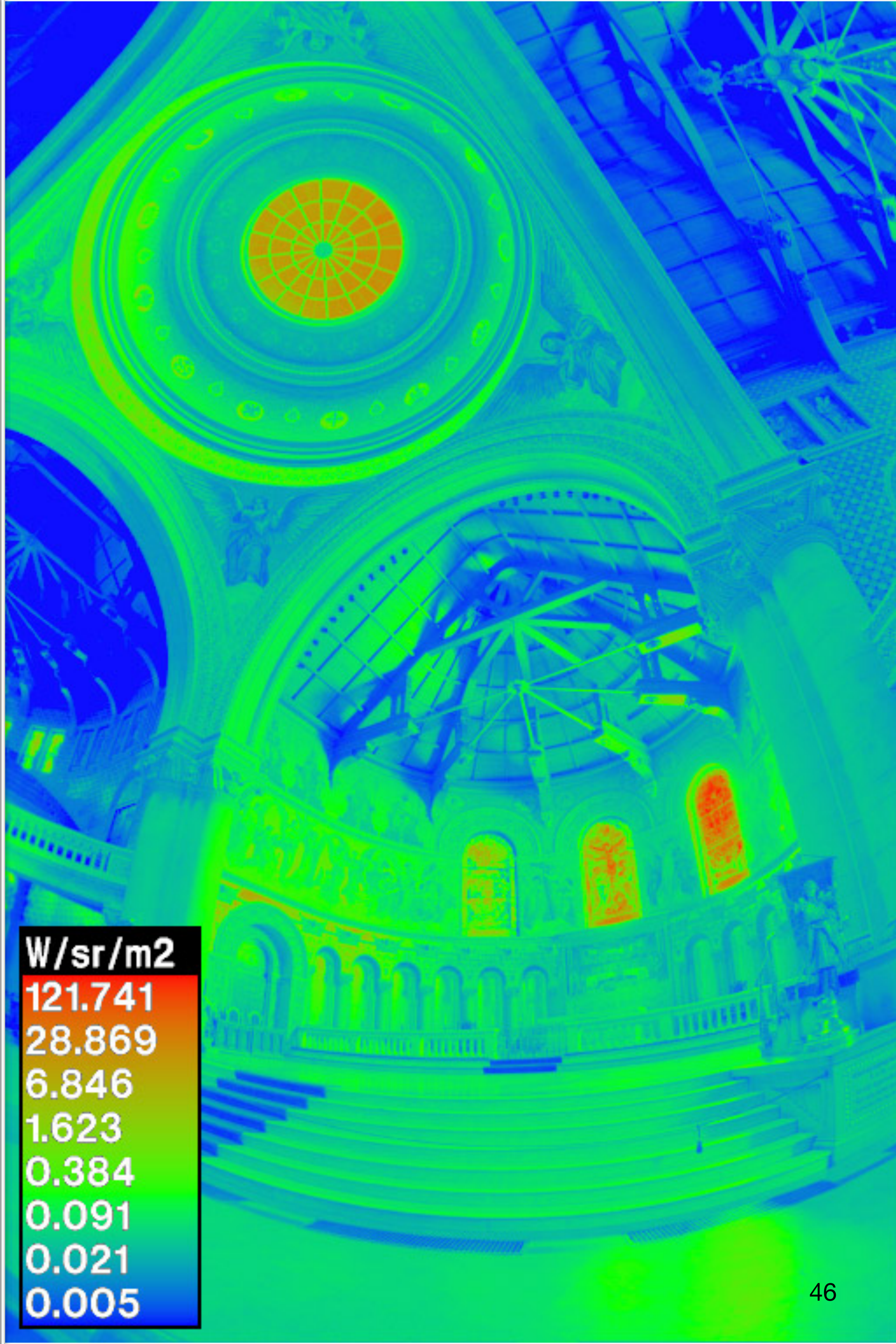
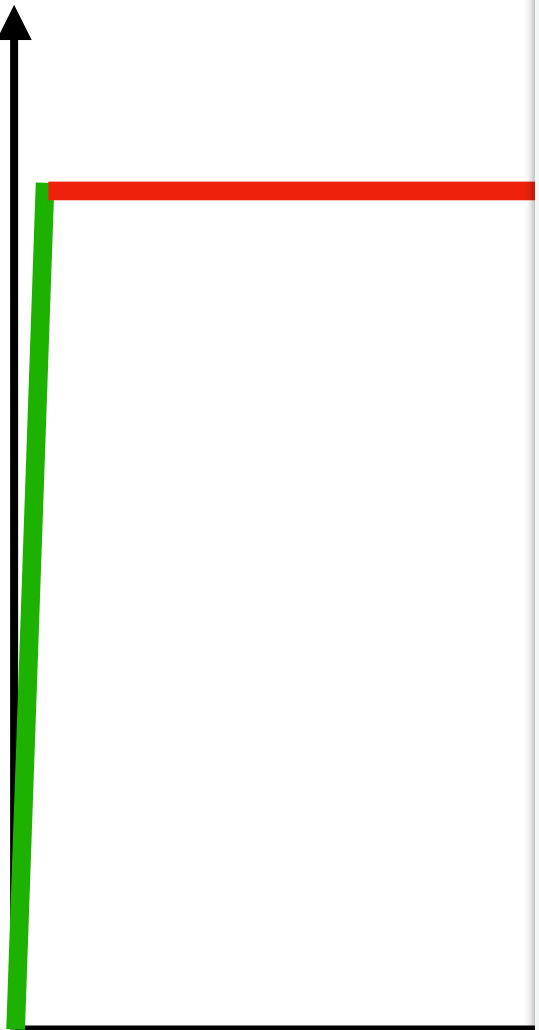




# Another Try

Linearly map the  
the DR to the [0,  
rest (i.e., 255 for  
will happen for th

Pixel  
value



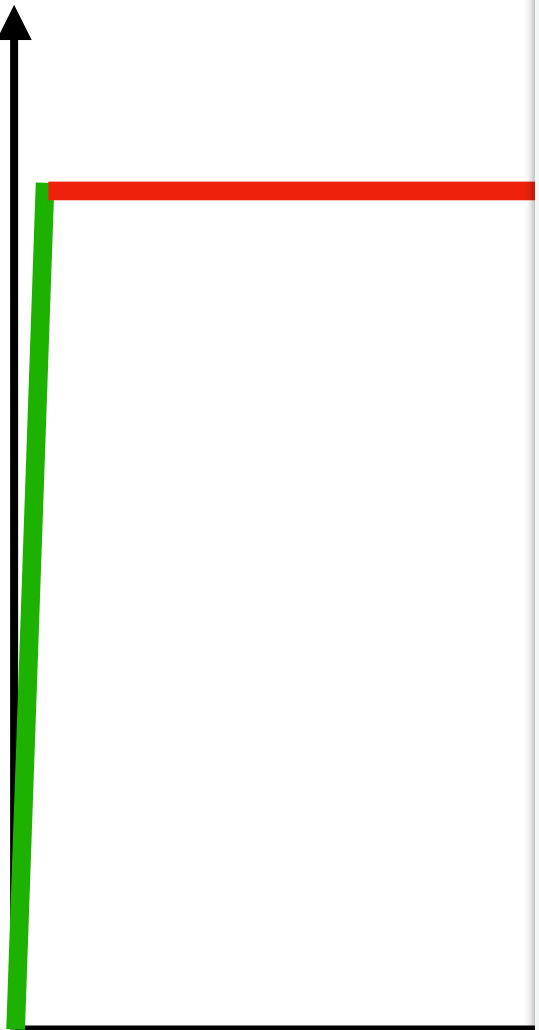
W/sr/m2
121.741
28.869
6.846
1.623
0.384
0.091
0.021
0.005



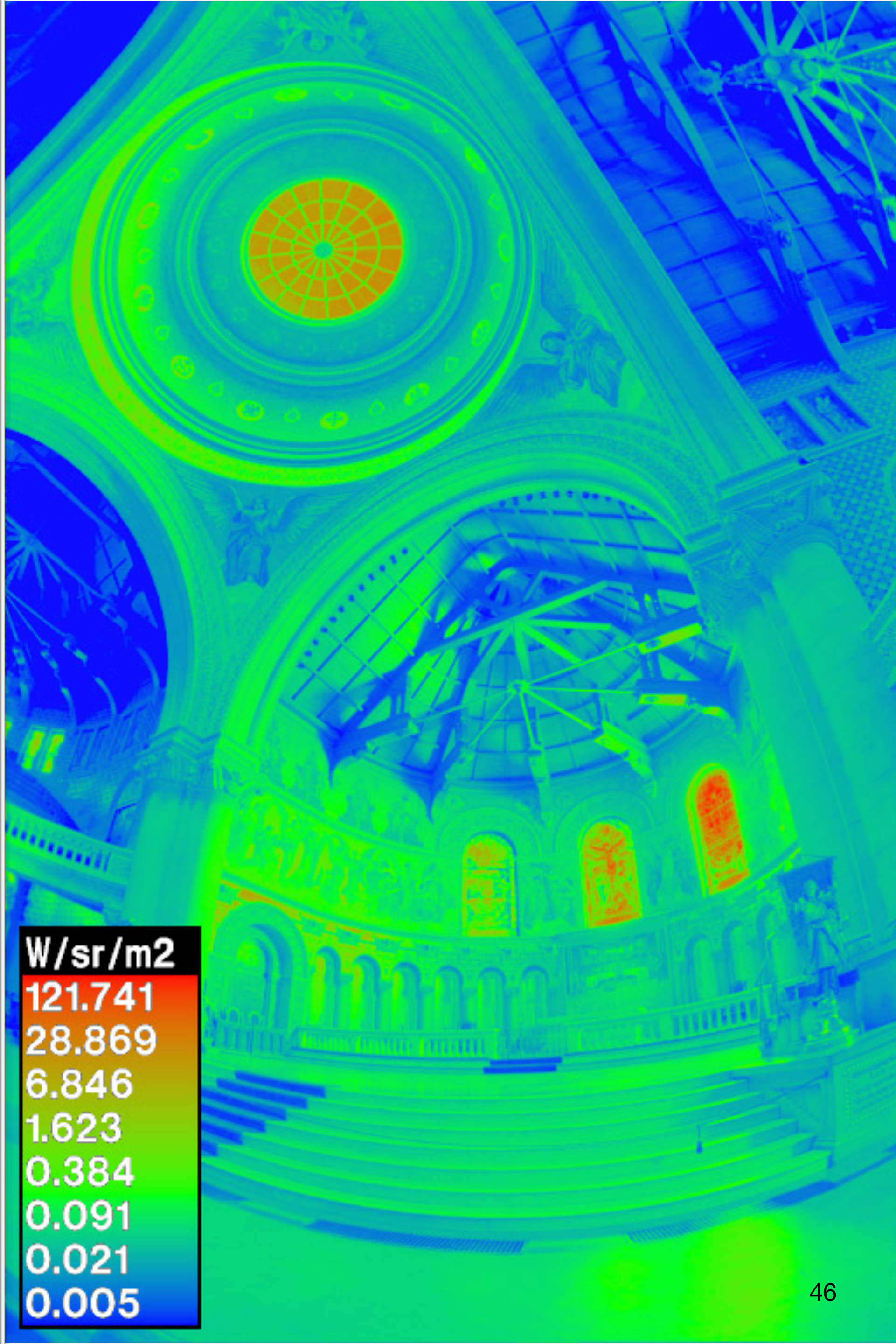
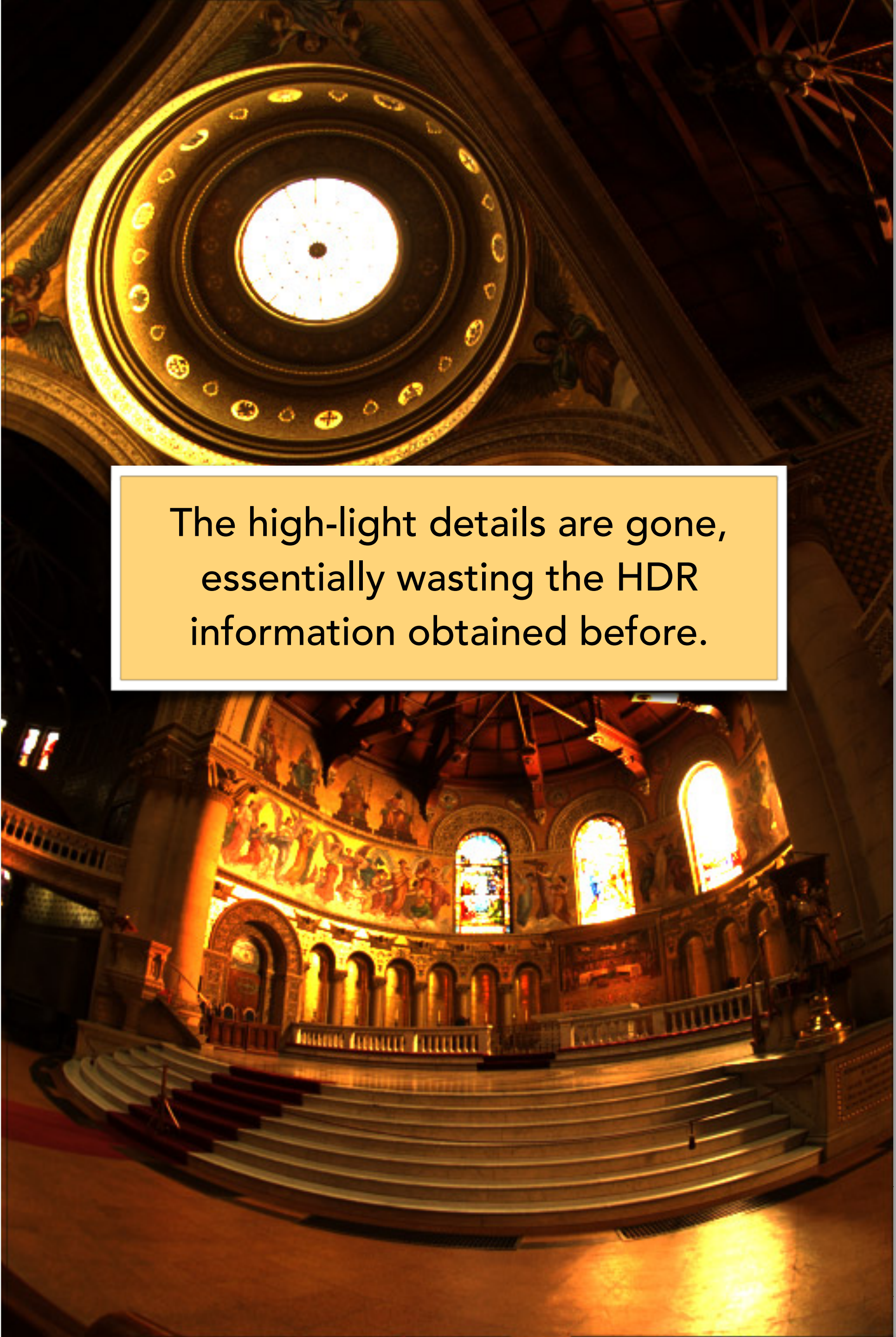
# Another Try

Linearly map the  
the DR to the [0,  
rest (i.e., 255 for  
will happen for th

Pixel  
value



The high-light details are gone,  
essentially wasting the HDR  
information obtained before.



W/sr/m2
121.741
28.869
6.846
1.623
0.384
0.091
0.021
0.005



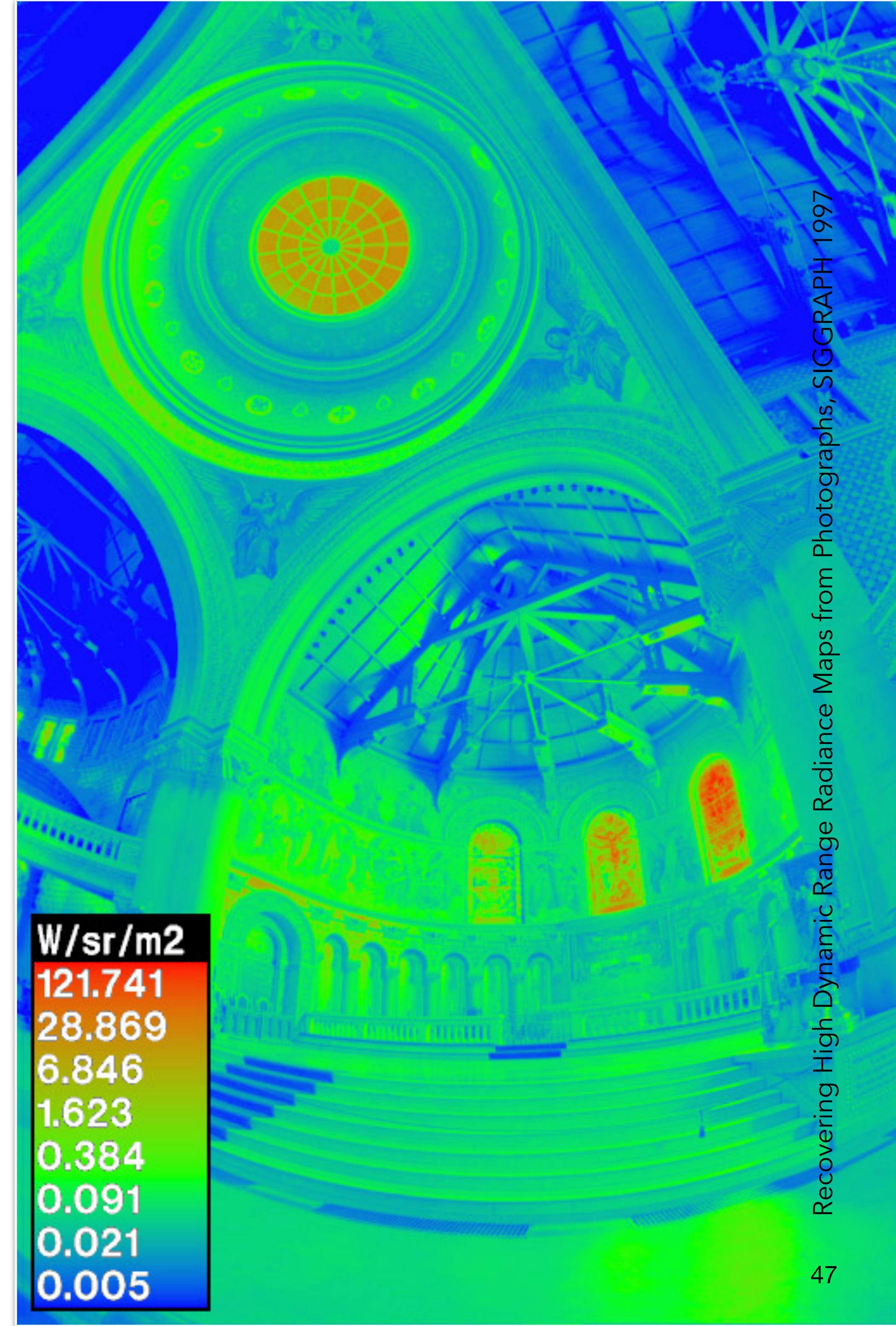
# Non-Linear Mapping

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$

Then scale to pixel values.

$$L_{display} = AL_{world}^{\gamma}$$

Gamma encoding. A is chosen to scale to desired pixel values.





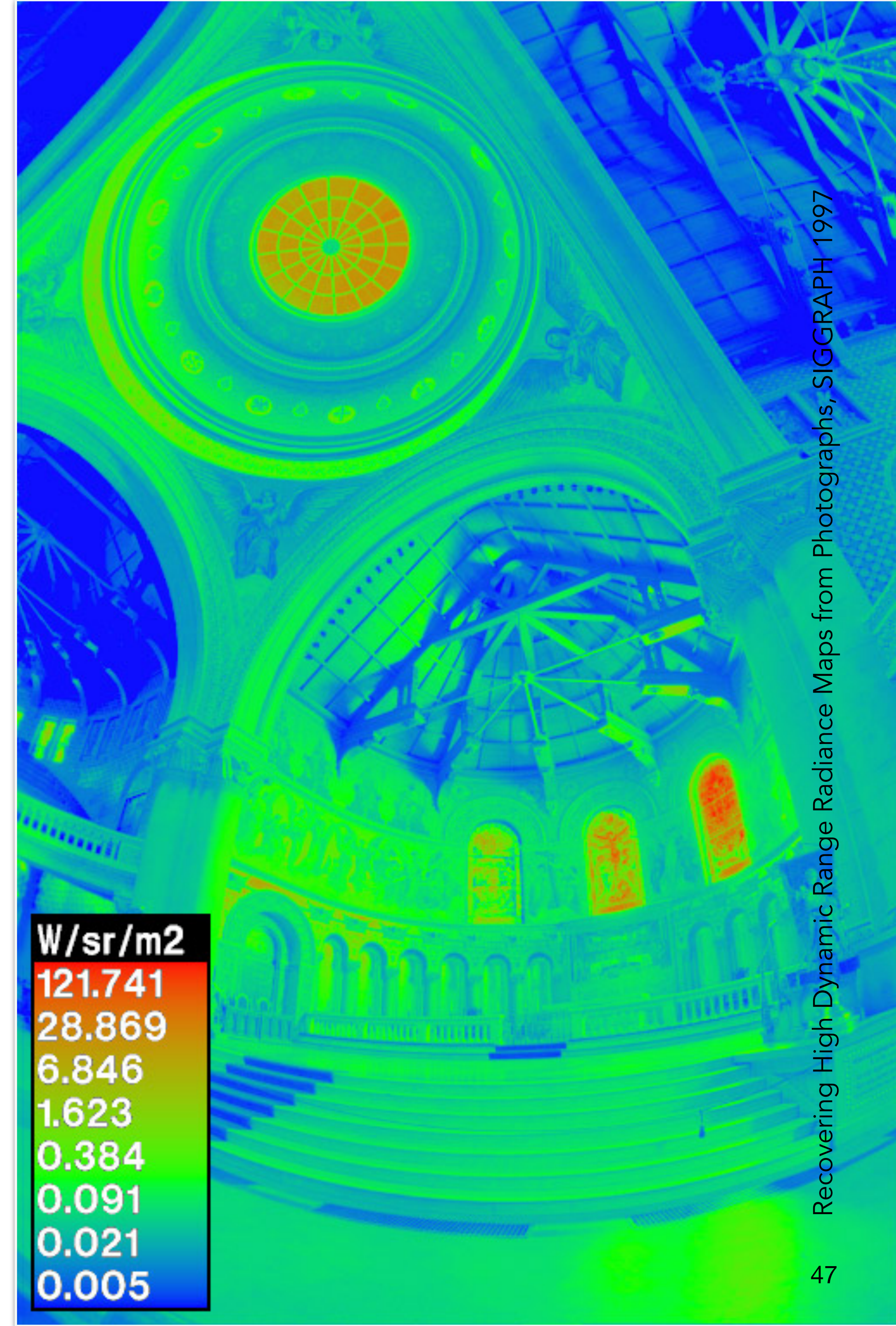
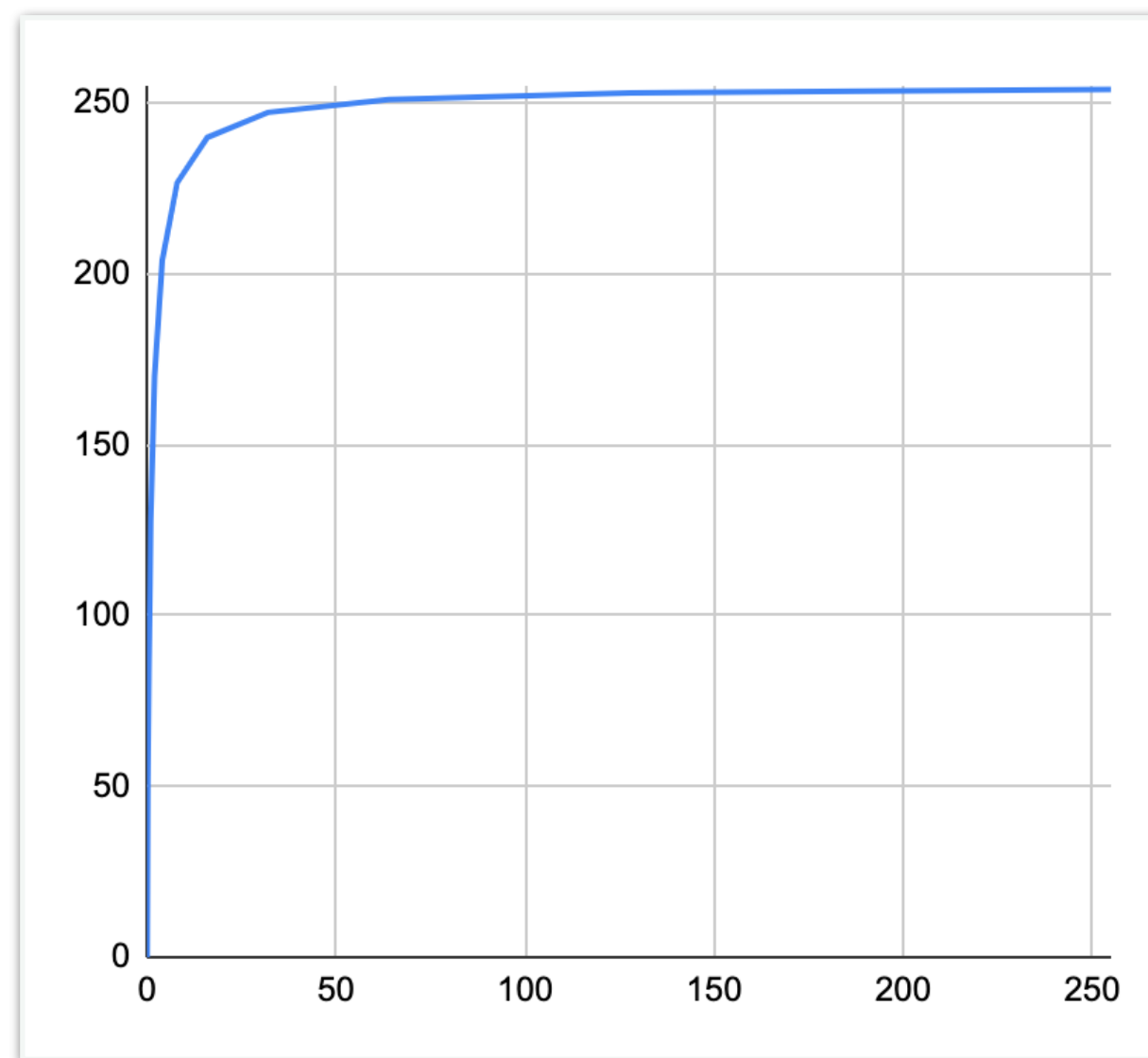
# Non-Linear Mapping

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$

Then scale to pixel values.

$$L_{display} = AL_{world}^{\gamma}$$

Gamma encoding. A is chosen to scale to desired pixel values.





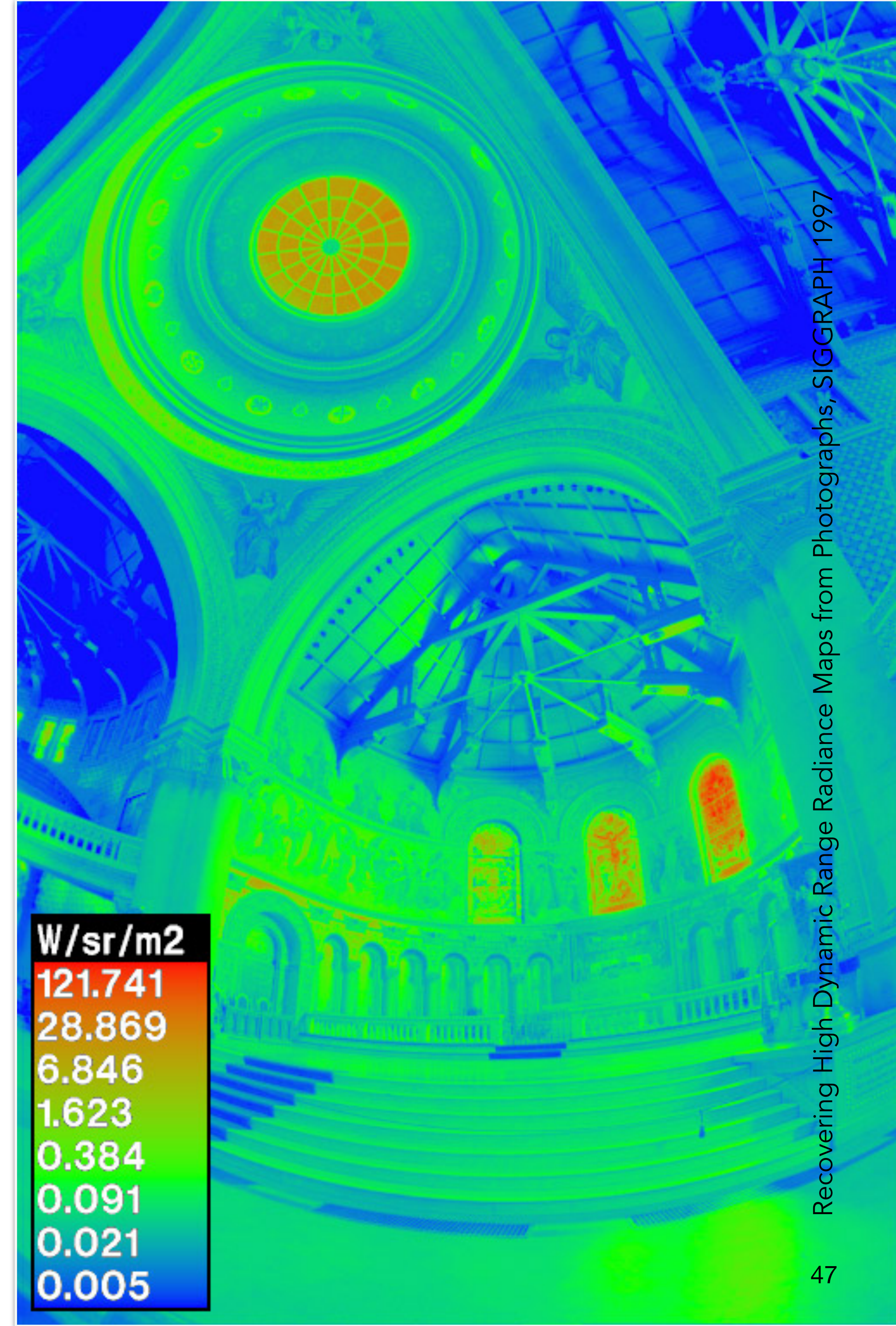
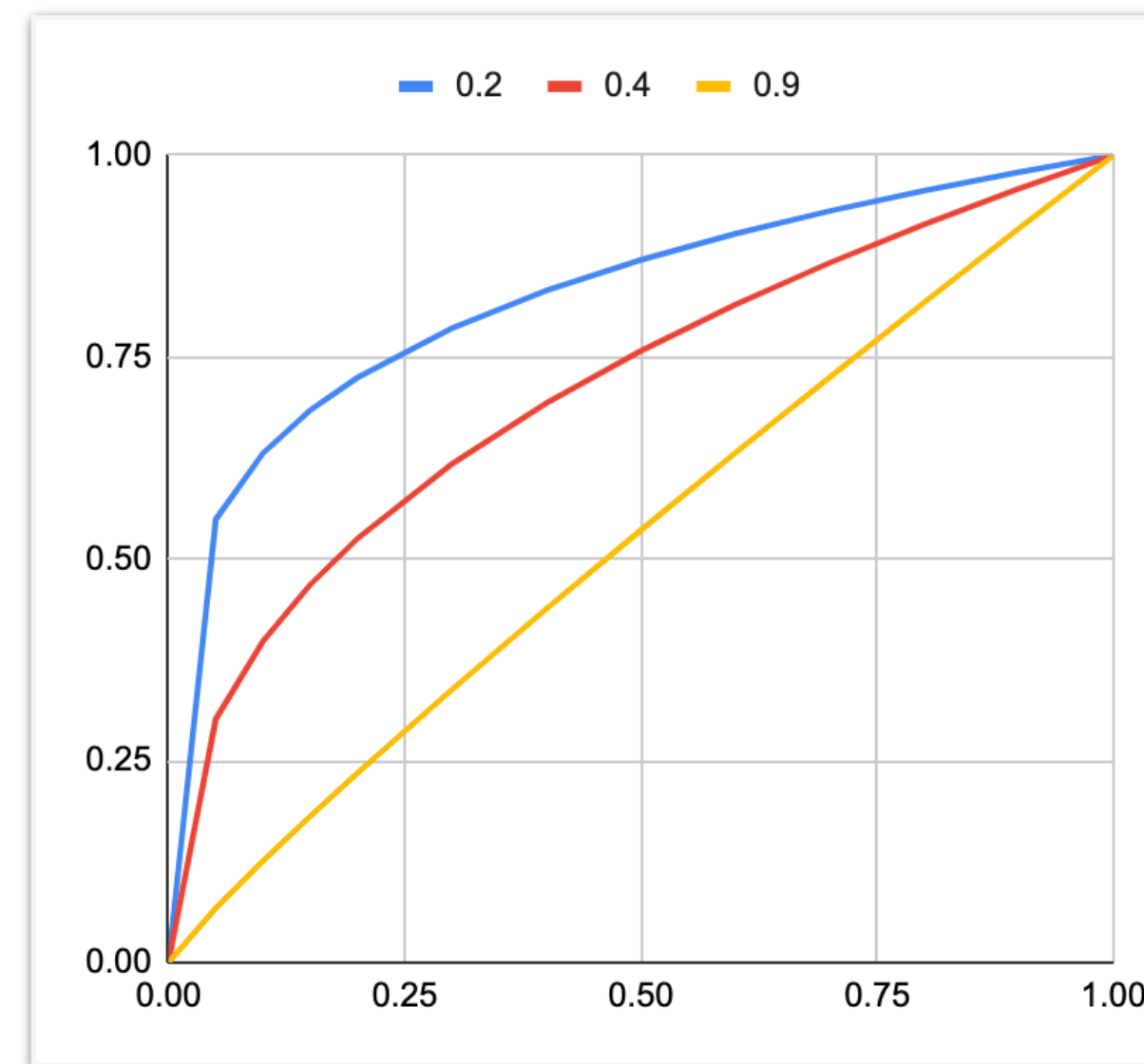
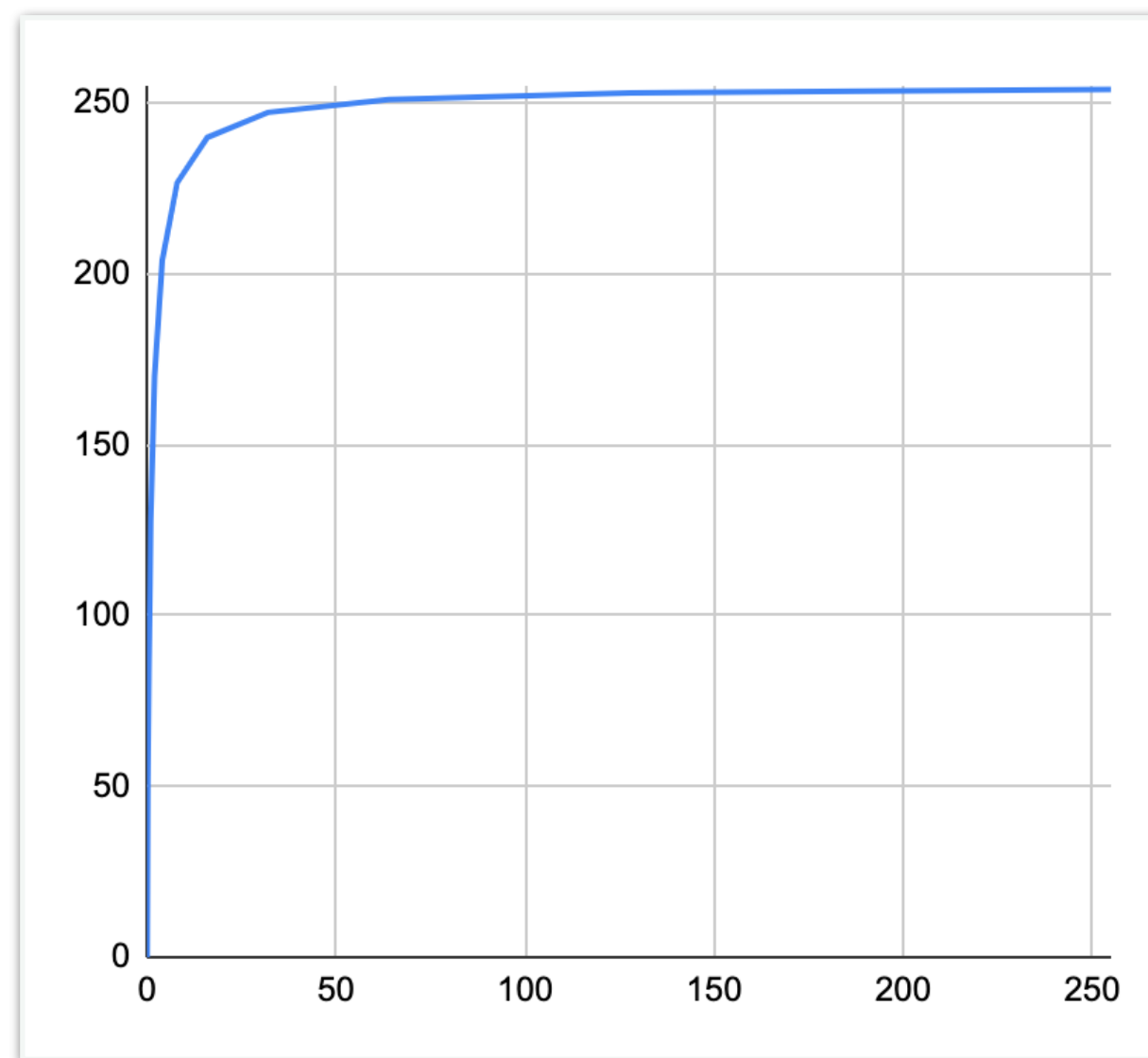
# Non-Linear Mapping

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$

Then scale to pixel values.

$$L_{display} = AL_{world}^{\gamma}$$

Gamma encoding. A is chosen to scale to desired pixel values.

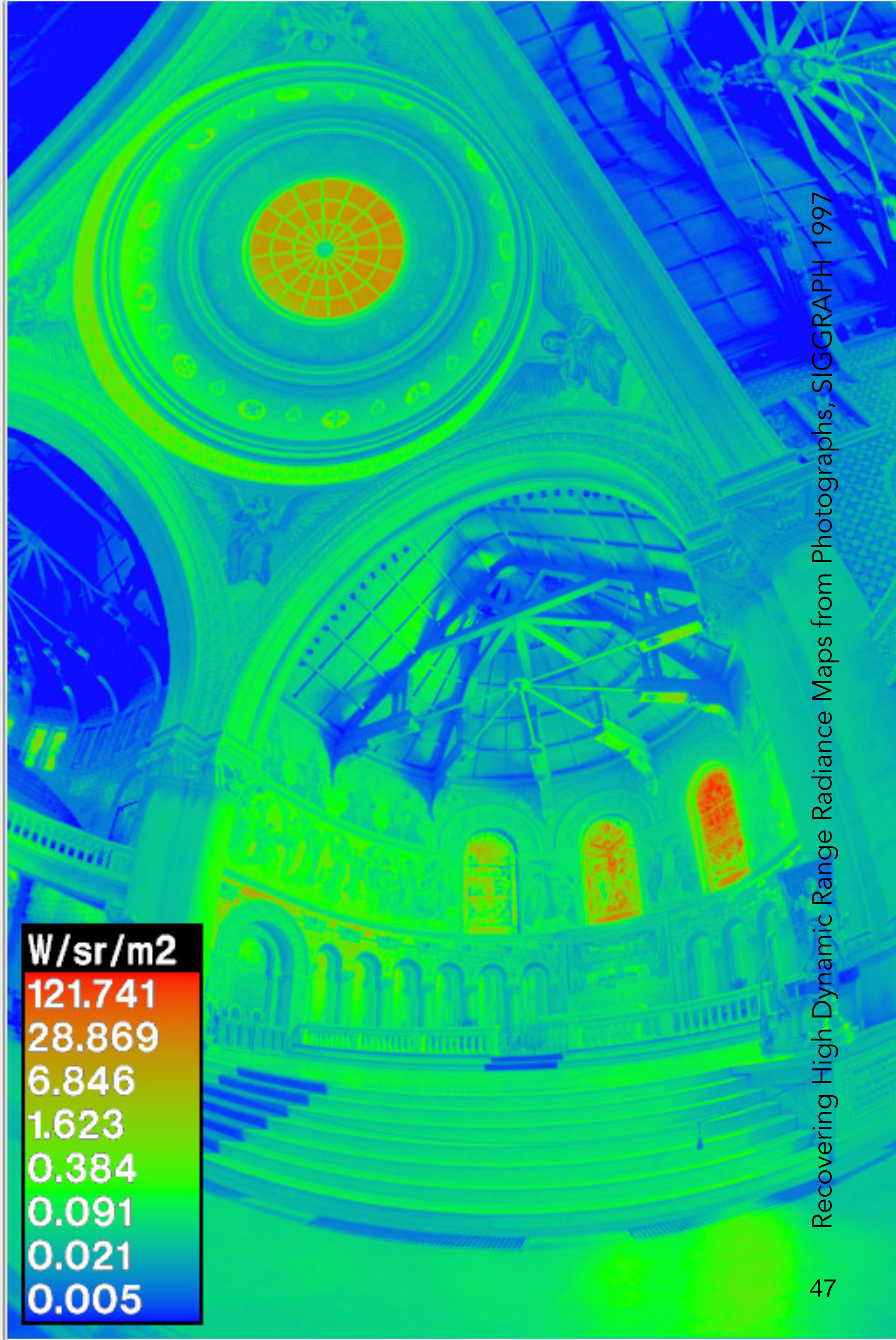
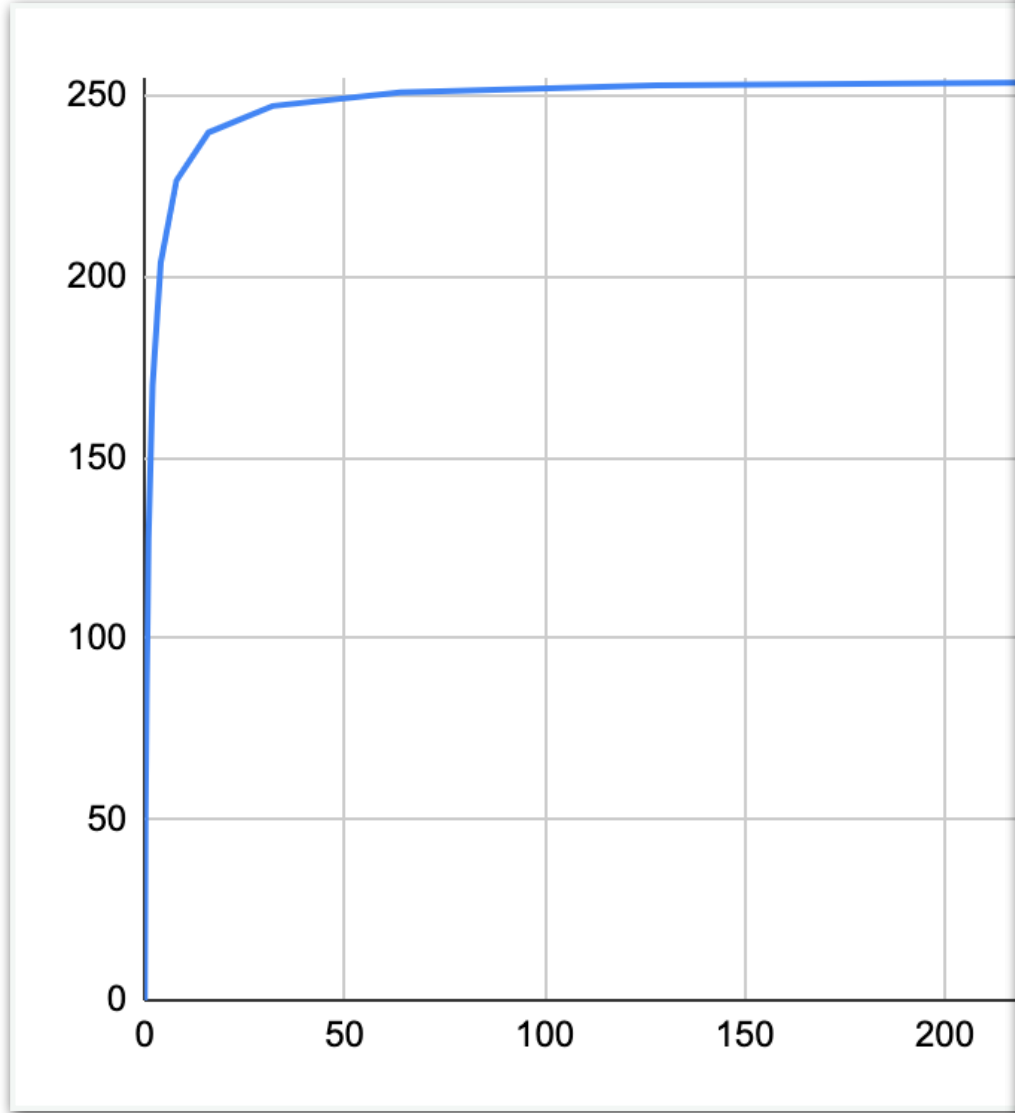




# Non-Linear

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$

$$L_{display} = AL^{\gamma}_{world}$$



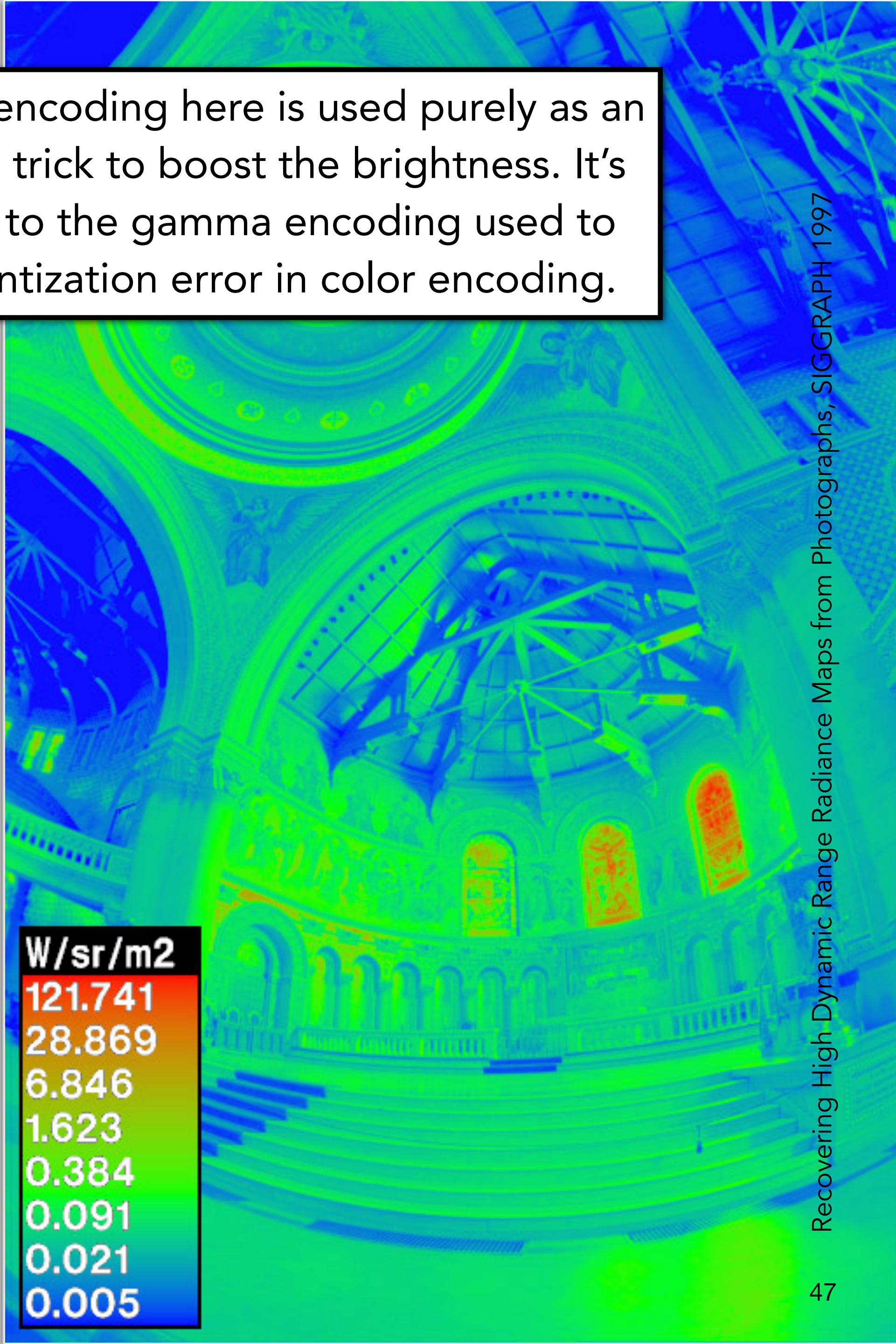
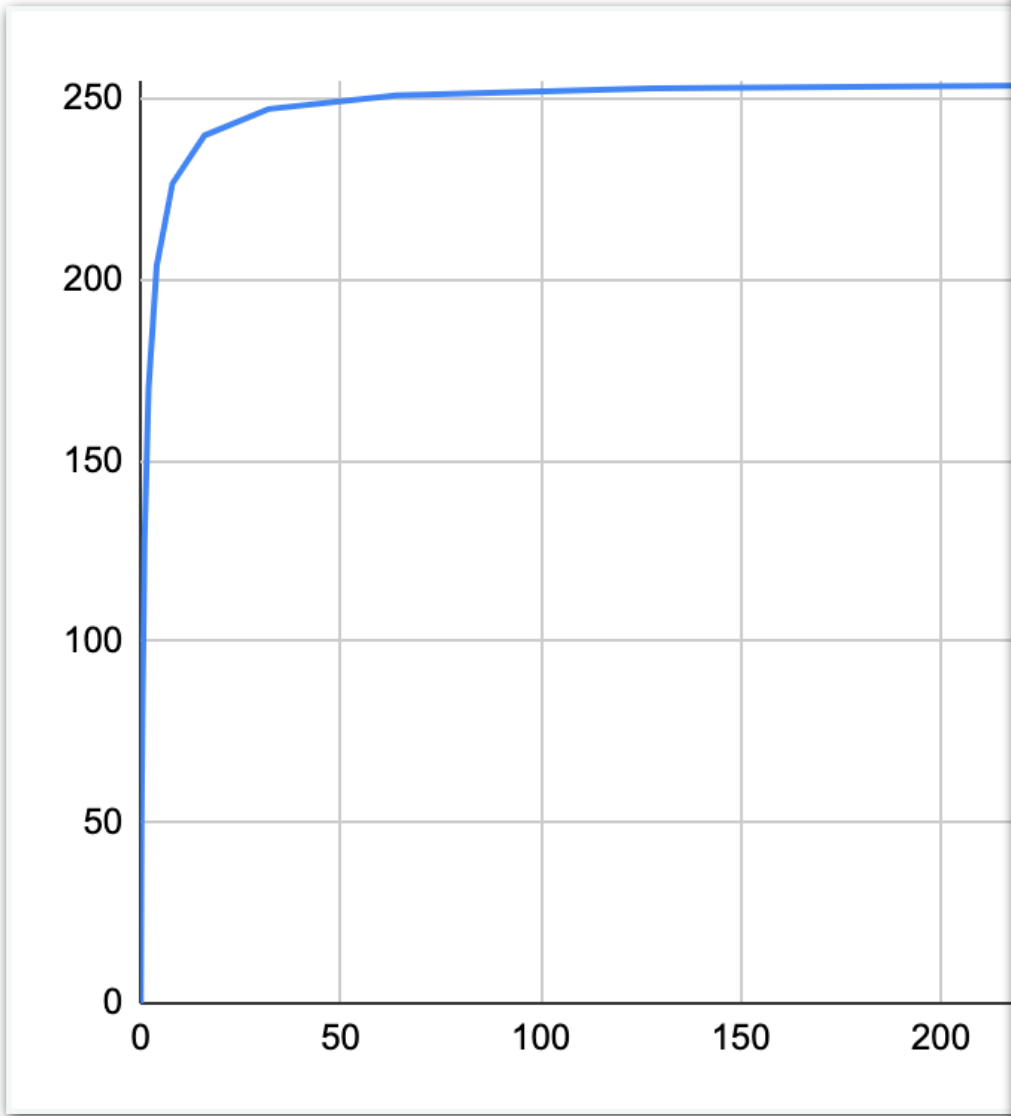


# Non-Linear

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$

$$L_{display} = AL_{world}^{\gamma}$$

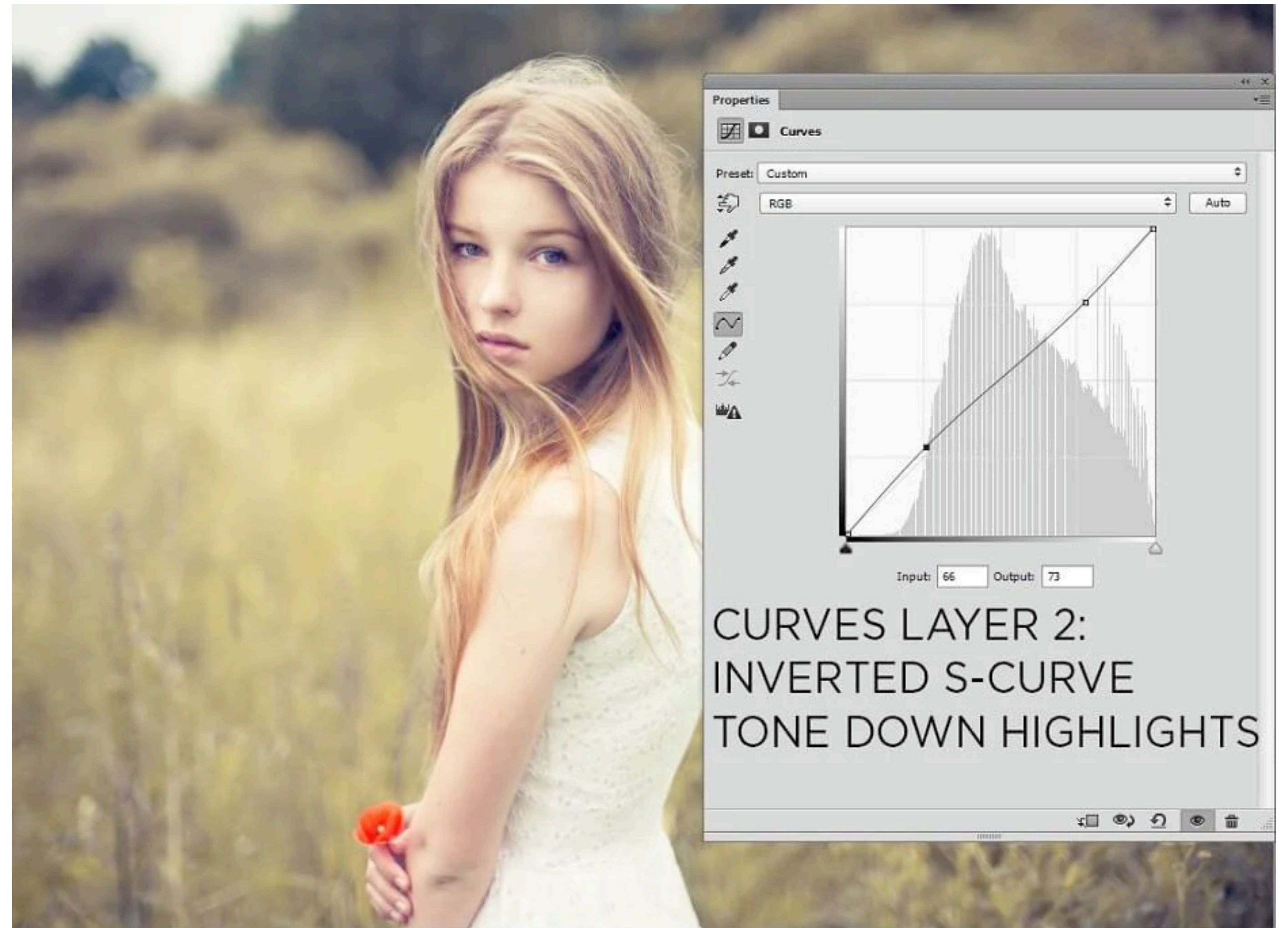
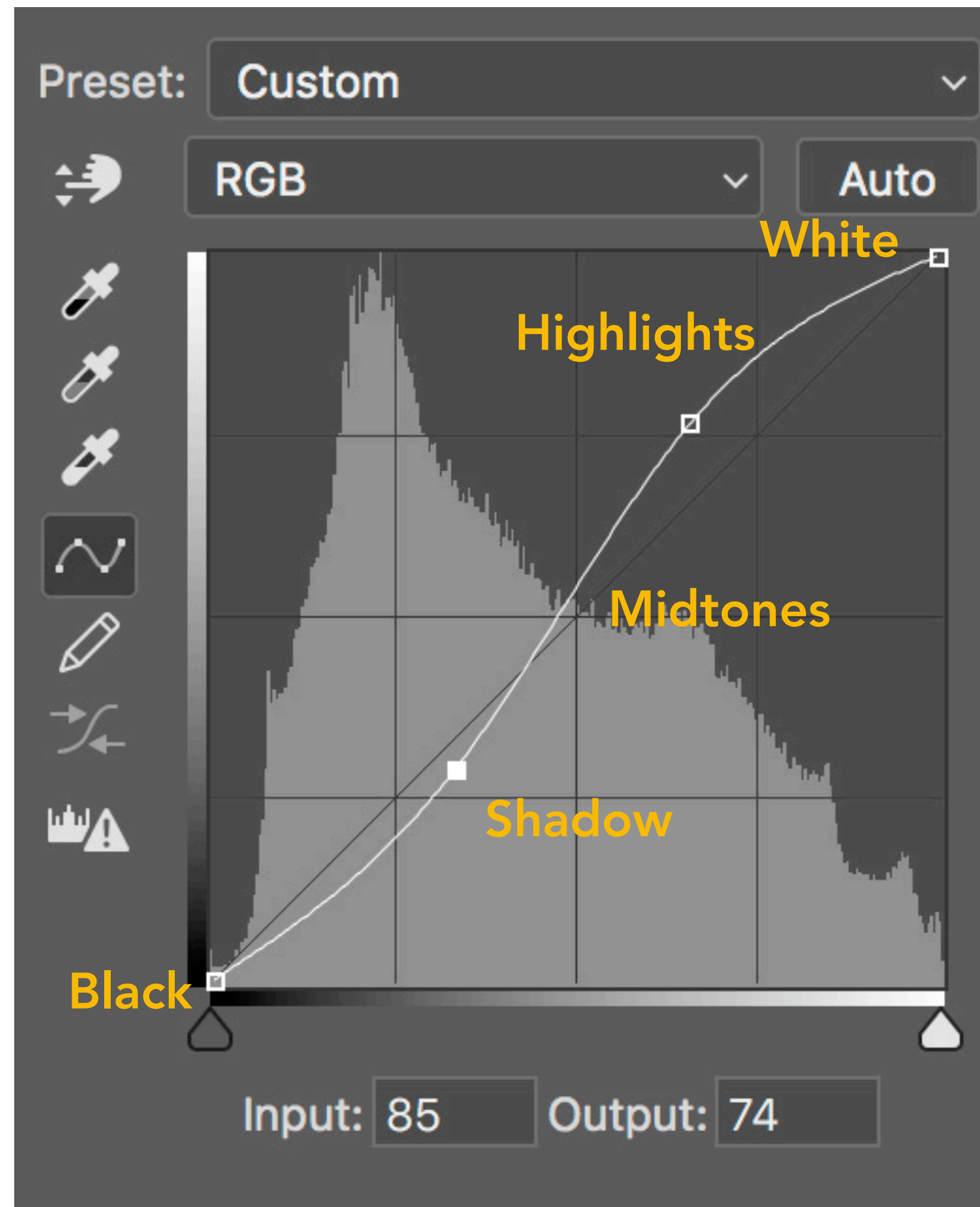
The gamma encoding here is used purely as an engineering trick to boost the brightness. It's not related to the gamma encoding used to reduce quantization error in color encoding.



W/sr/m2	
121.741	
28.869	
6.846	
1.623	
0.384	
0.091	
0.021	
0.005	



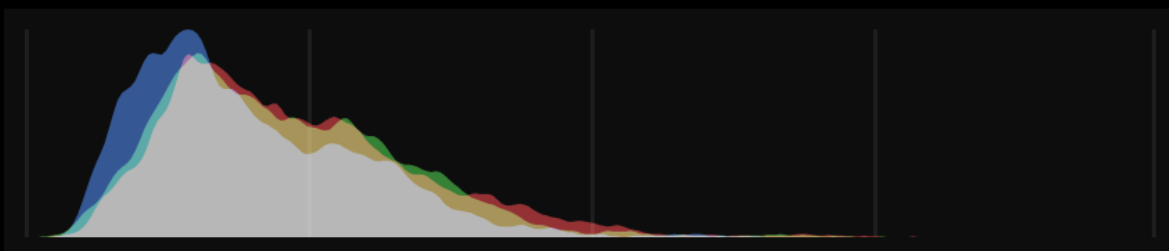
# S-Shaped Curve in Software





11:17

LTE

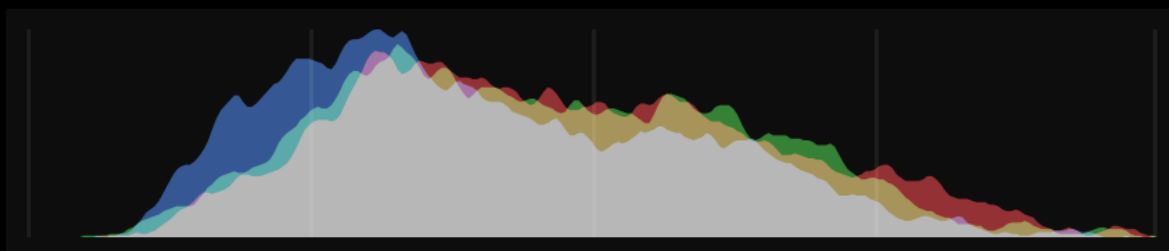


Curve

Done



11:17

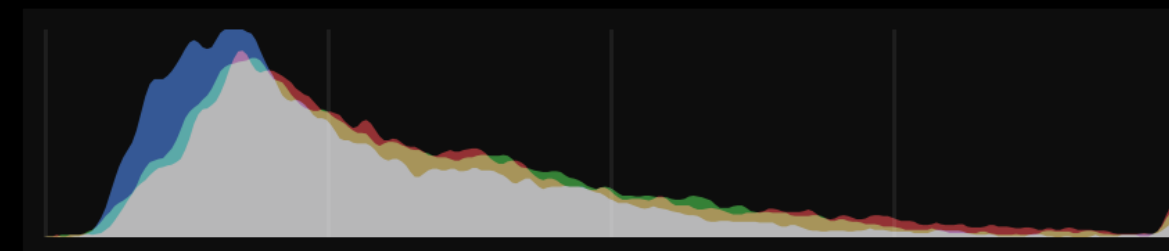


Curve

Done



11:17



Curve

Done





# Issue with Global Tone Mapping

All schemes so far are *global*; they apply the same operator to all pixels.

- Different pixels, regardless of where they are in the scene, are mapped in the same way if they have the same radiance.

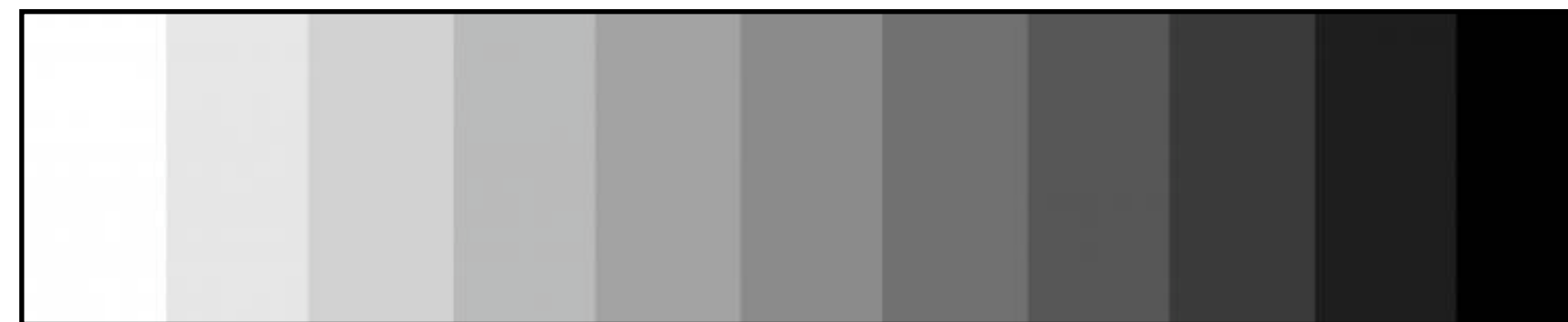
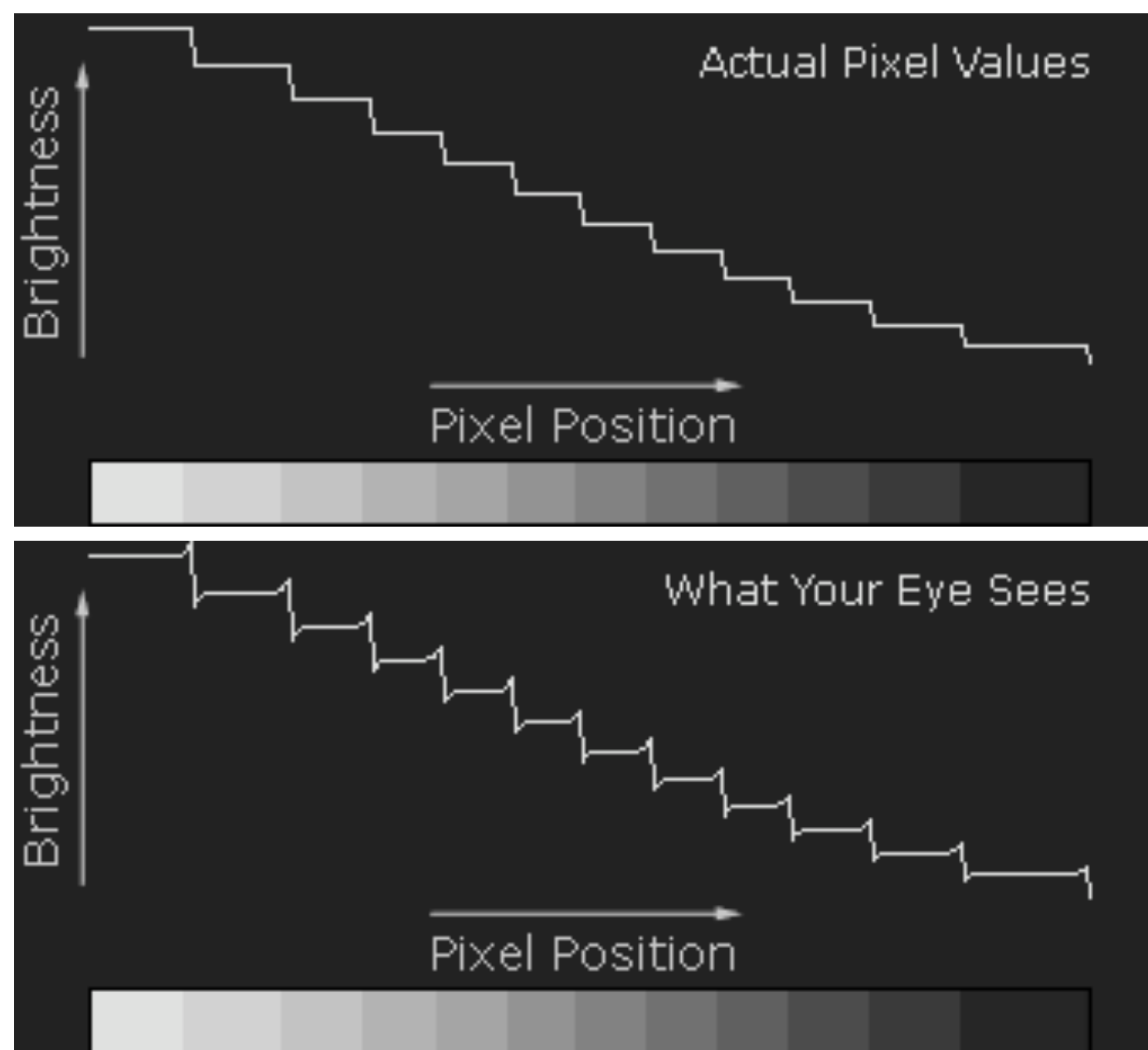
**Problem:** since contrasts are globally reduced, local contrasts are diminished.

Why preserve local contrasts? Human eyes adapt to local field around the current fixation.

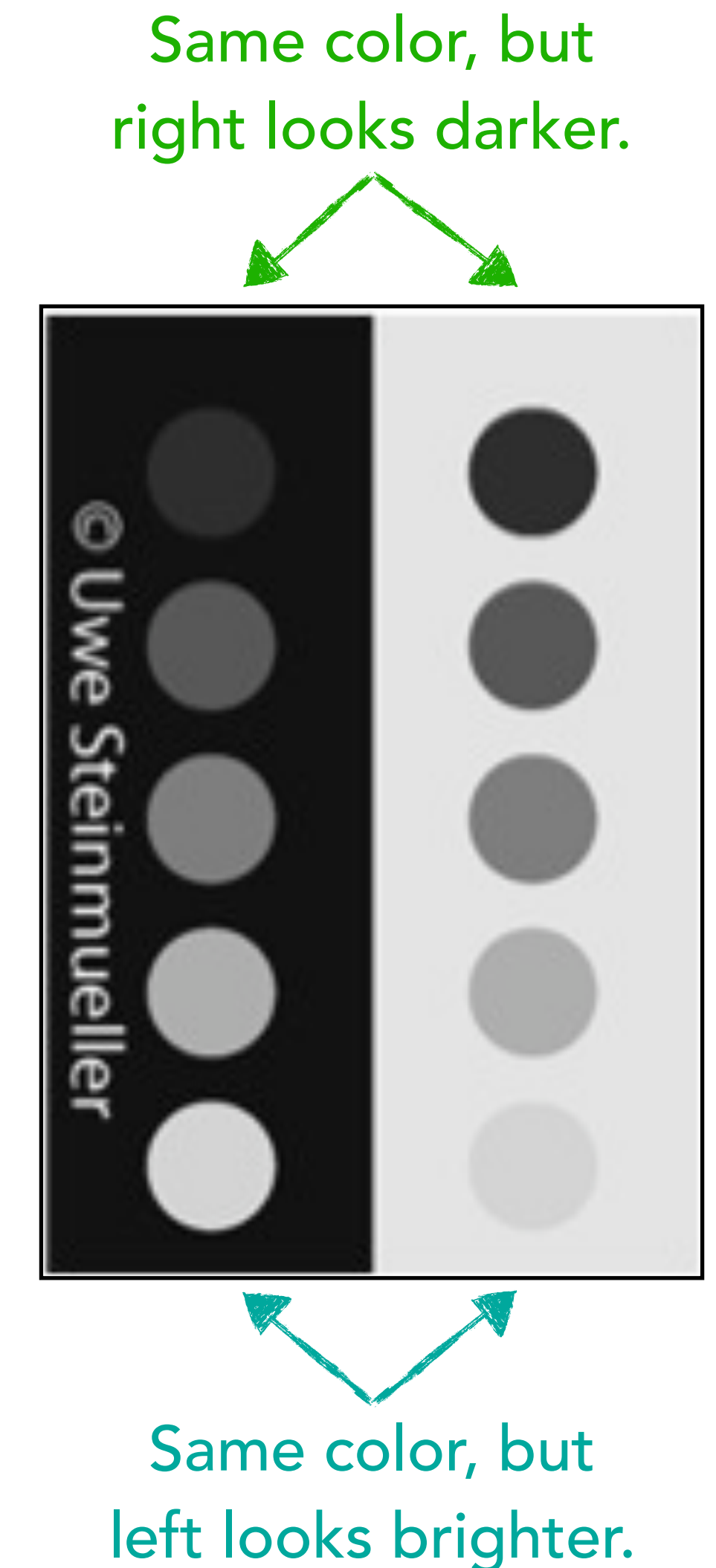


# Local Contrast is Important

Intuitively, at an edge brighter color becomes brighter and dark color becomes darker. Our brain does this “contrast enhancement” so we can better discern details at edges.



Mach band illusion





# Preserve Local Contrast

What does it mean to preserve local contrast?

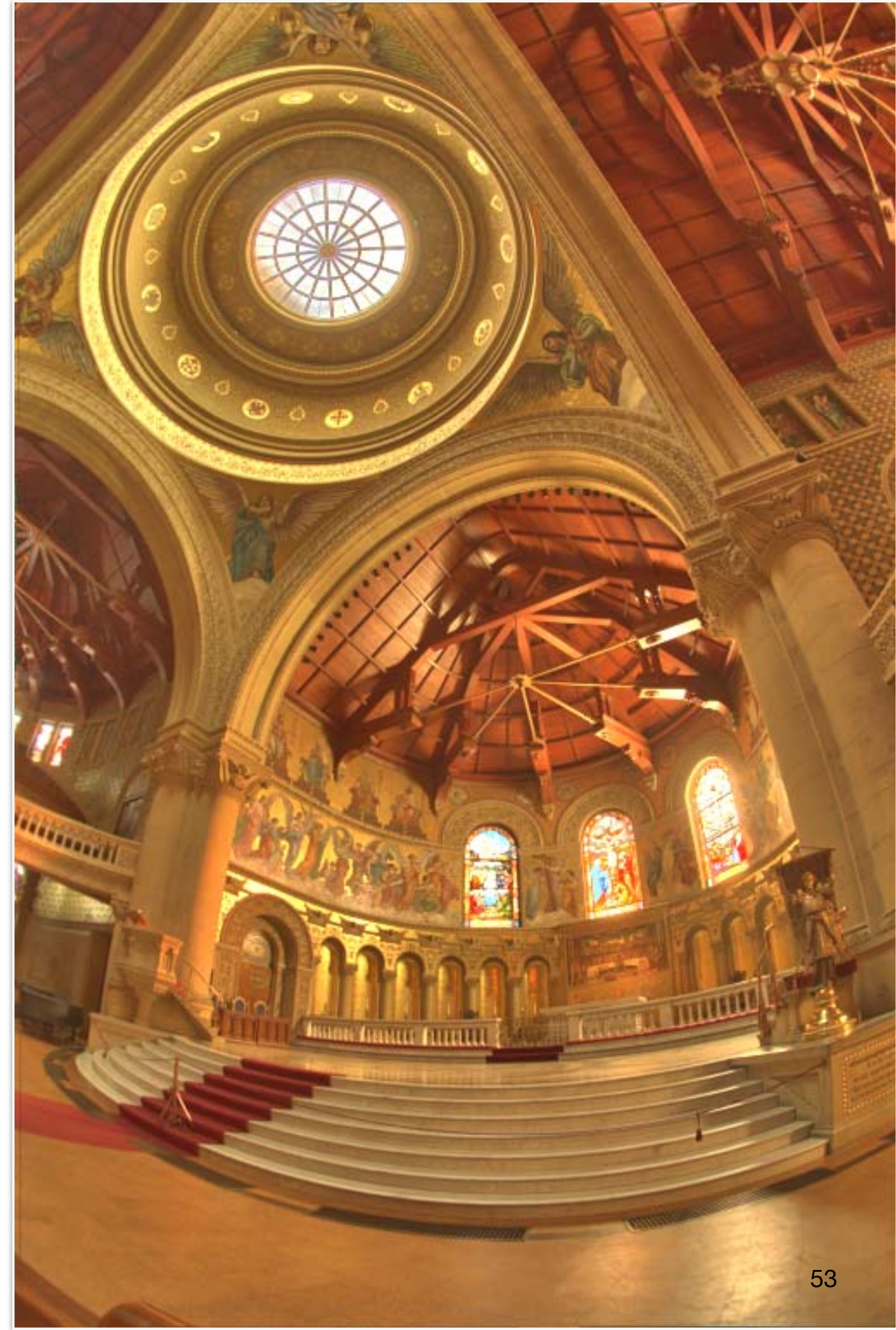
Make sure in local regions brighter pixels look (much) brighter than dark pixels. Doesn't mean the exactly same luminance difference or ratio needs to be maintained.

Many local tone mapping schemes strive to preserve, sometimes even enhance, local contrast.



# Local Tone Mapping

Advanced tone mapping schemes maintain local contrast (while reducing global contrast).

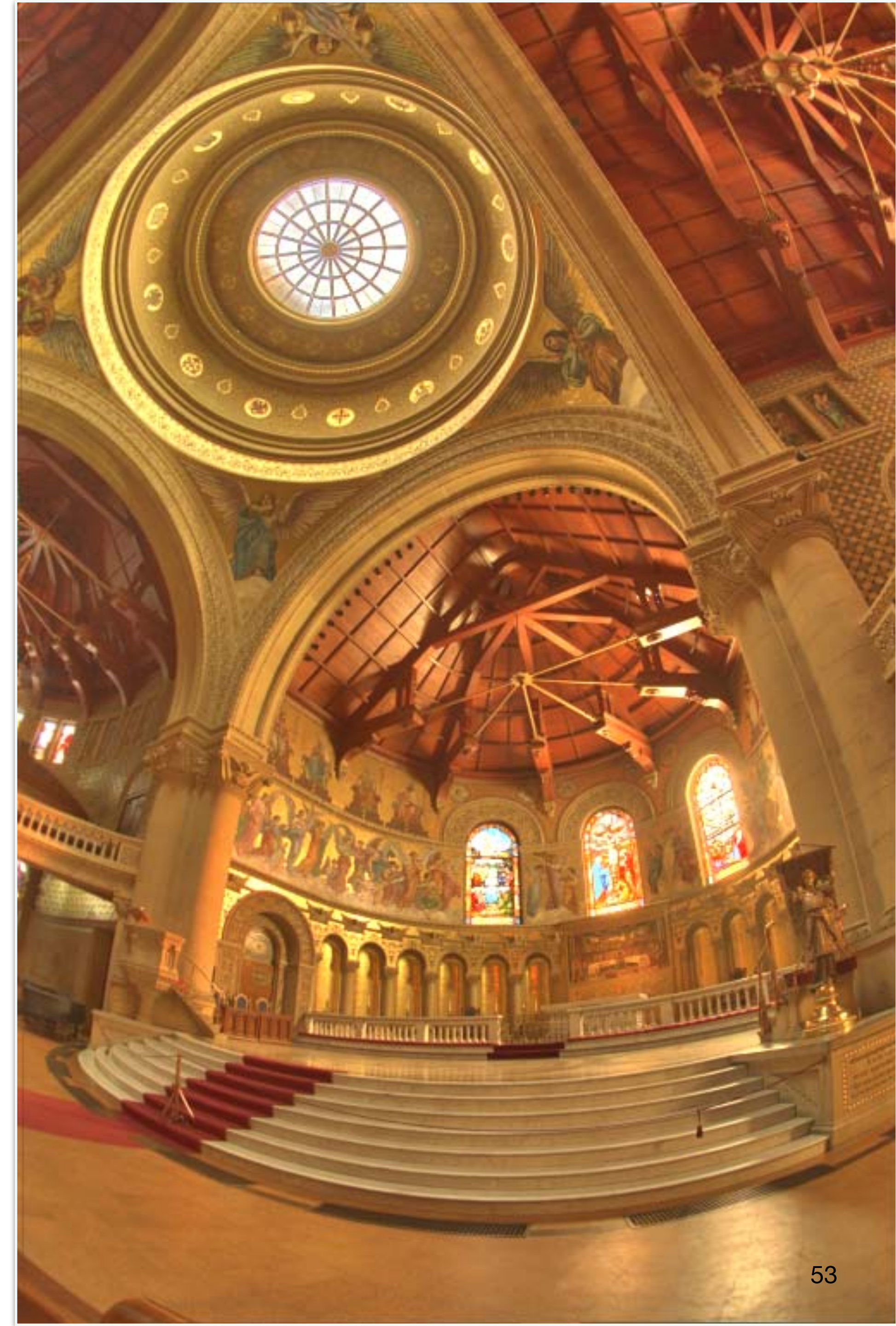




# Local Tone Mapping

Advanced tone mapping schemes maintain local contrast (while reducing global contrast).

Sometimes people take a standard DR image and enhance its local contrast, which gives an illusion of HDR as we feel that details are more pronounced. But the actual DR is unchanged.





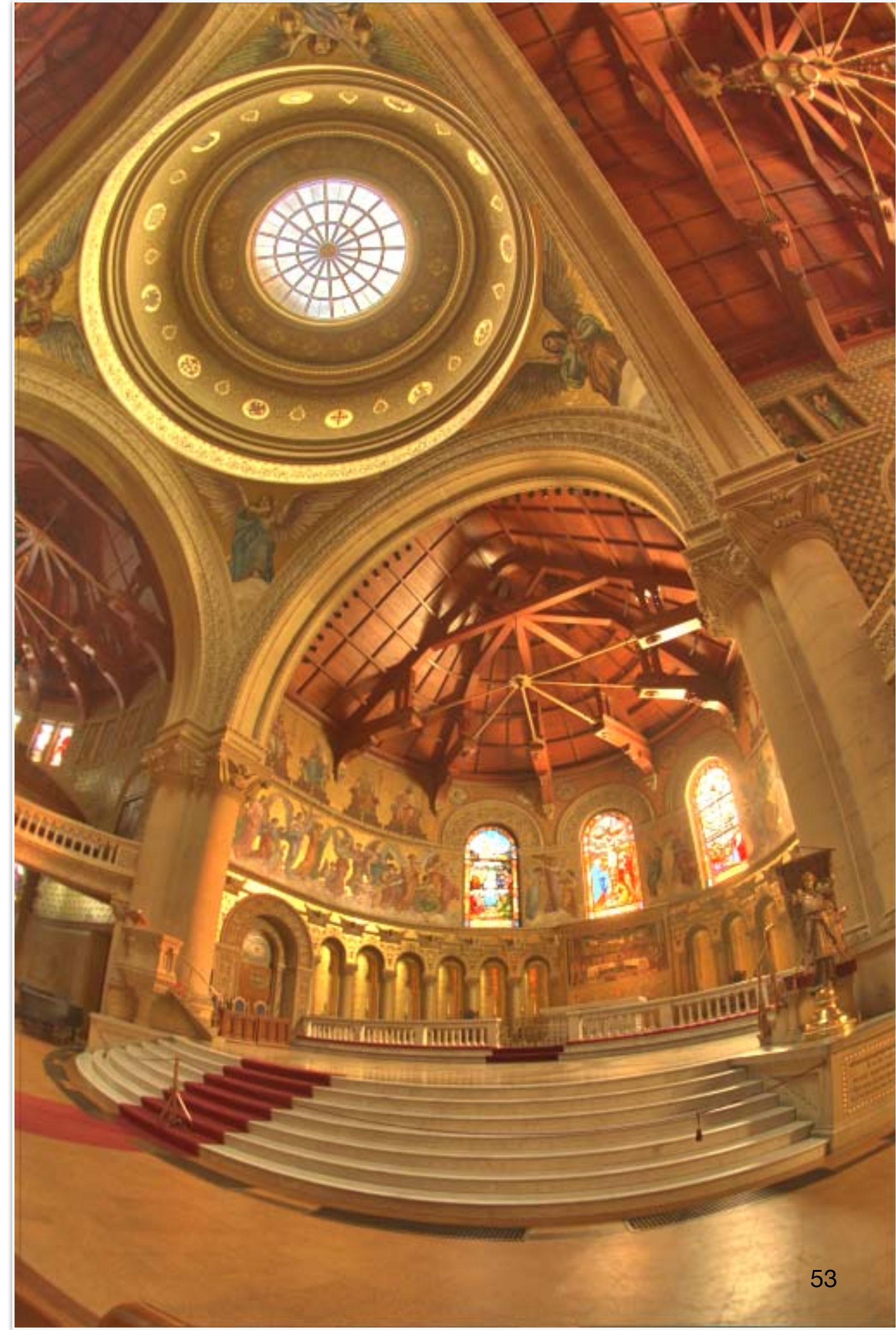
# Local Tone Mapping

Advanced tone mapping schemes maintain local contrast (while reducing global contrast).

Sometimes people take a standard DR image and enhance its local contrast, which gives an illusion of HDR as we feel that details are more pronounced. But the actual DR is unchanged.

But be careful not to enhance local contrast too much. Otherwise we get the “HDR look”

- images look very sharp and details are artificially exaggerated but overall the contrast feels “flat”.







“The HDR Look”





“The HDR Look”



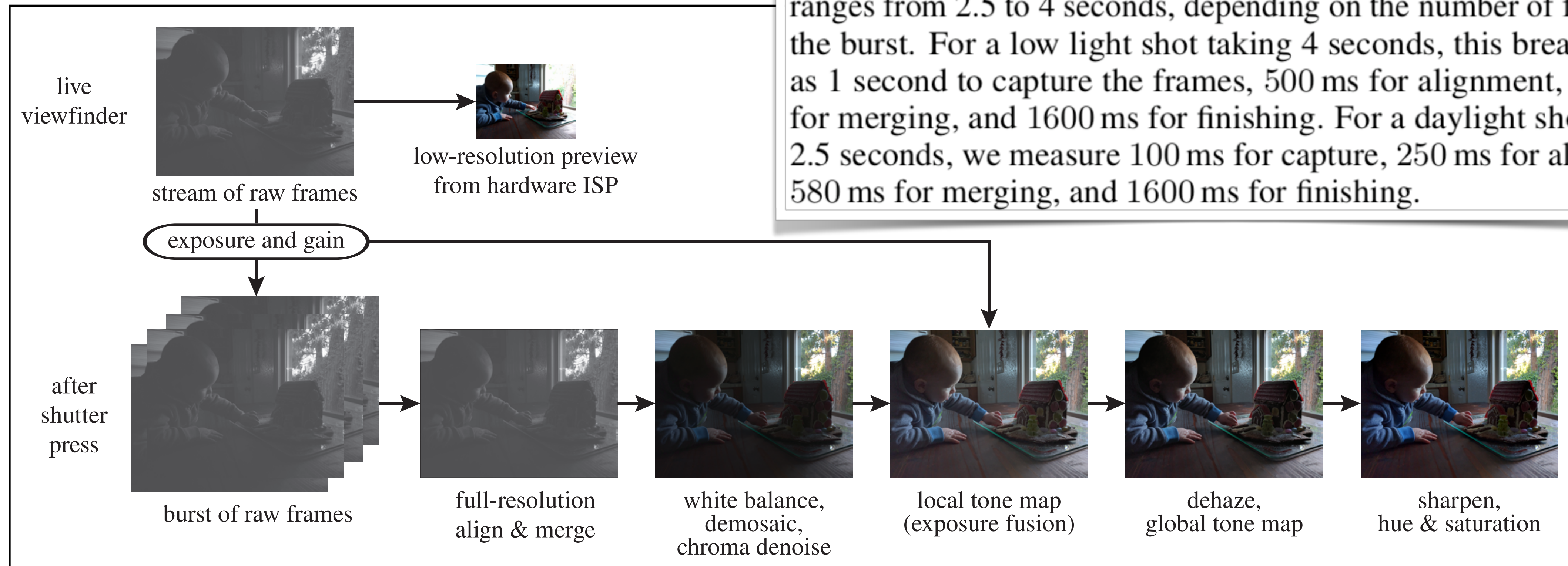
# "The HDR Look"





# Google Pixel 2's HDR Pipeline (HDR+)

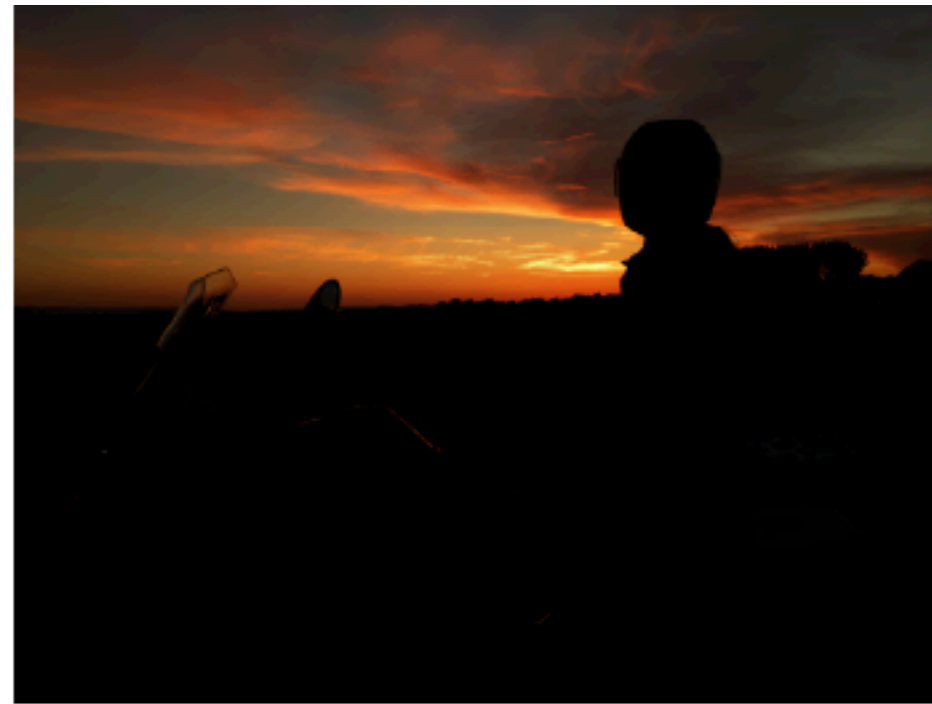
Summarizing, on a Qualcomm Snapdragon 810 not subject to thermal throttling the time required to produce an output photograph ranges from 2.5 to 4 seconds, depending on the number of frames in the burst. For a low light shot taking 4 seconds, this breaks down as 1 second to capture the frames, 500 ms for alignment, 1200 ms for merging, and 1600 ms for finishing. For a daylight shot taking 2.5 seconds, we measure 100 ms for capture, 250 ms for alignment, 580 ms for merging, and 1600 ms for finishing.



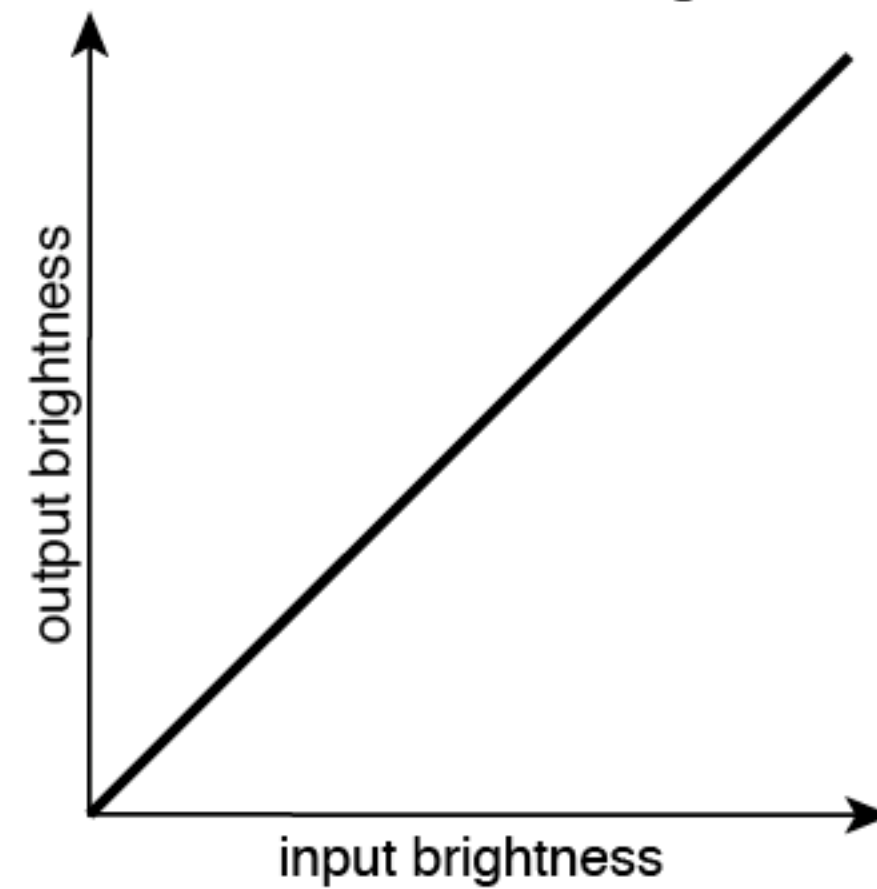
Viewfinder executed on hardware ISP. Real-time but no HDR. Main HDR pipeline written in Halide and executes in SIMD/multi-core CPUs. Viewfinder image different from the final output.



# Tone Mapping in HDR+



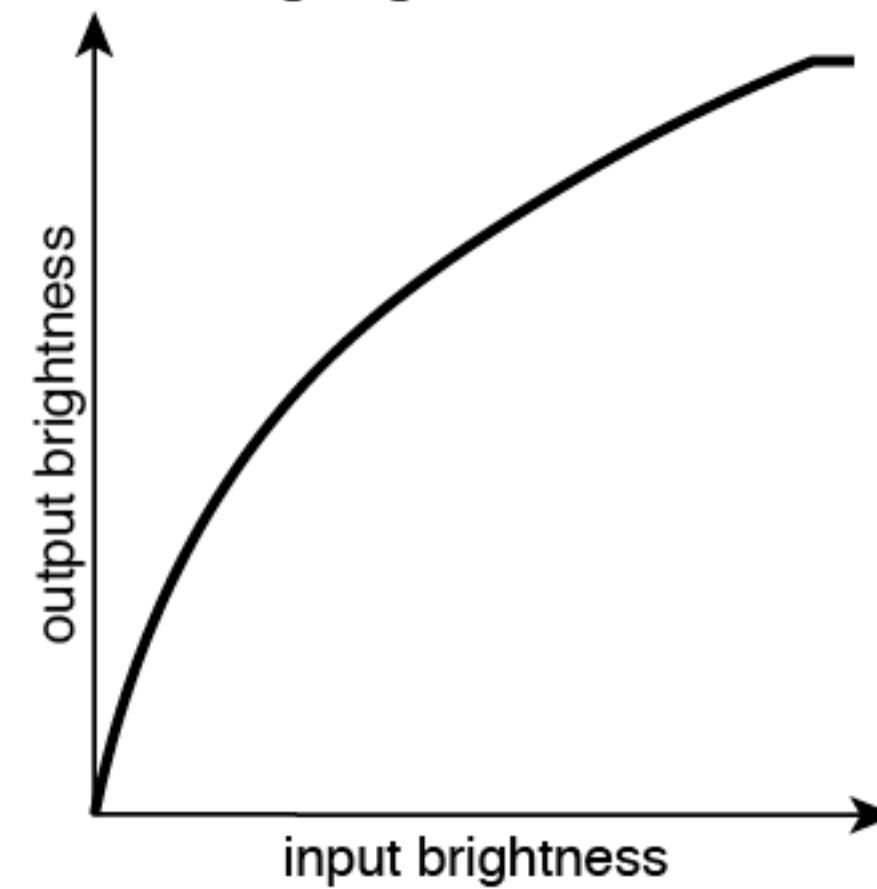
Linear RGB image



(a)



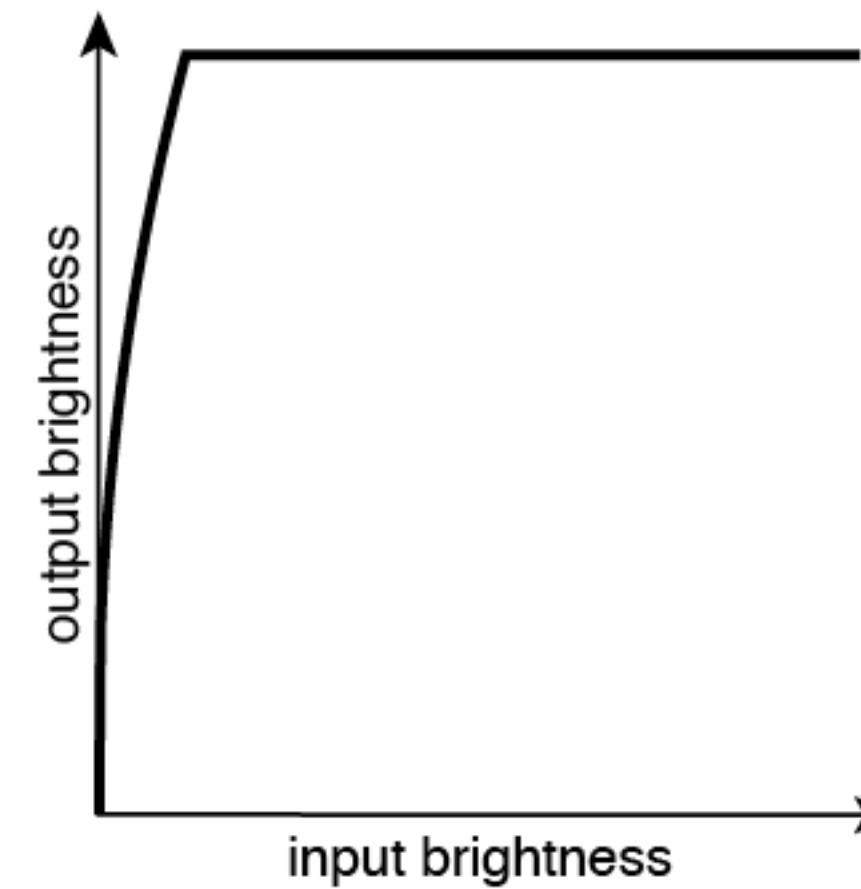
Highlights visible



(b)



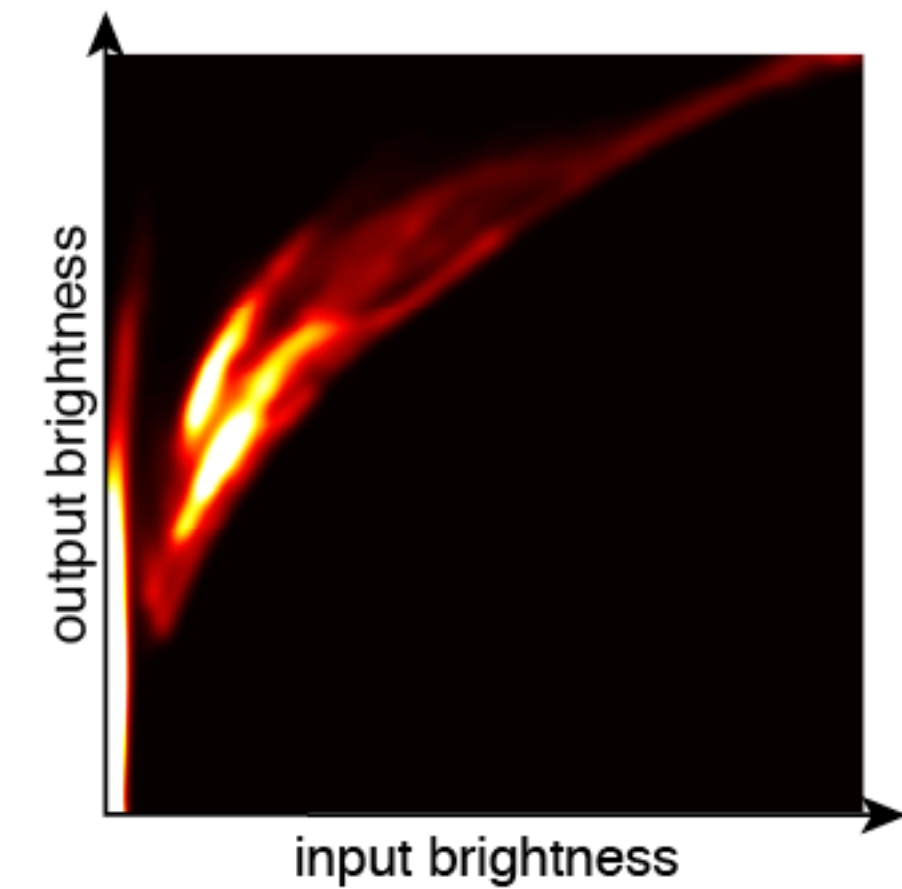
Shadows visible



(c)



HDR+



(d)



# Tone Mapping in HDR+



Linear RGB image



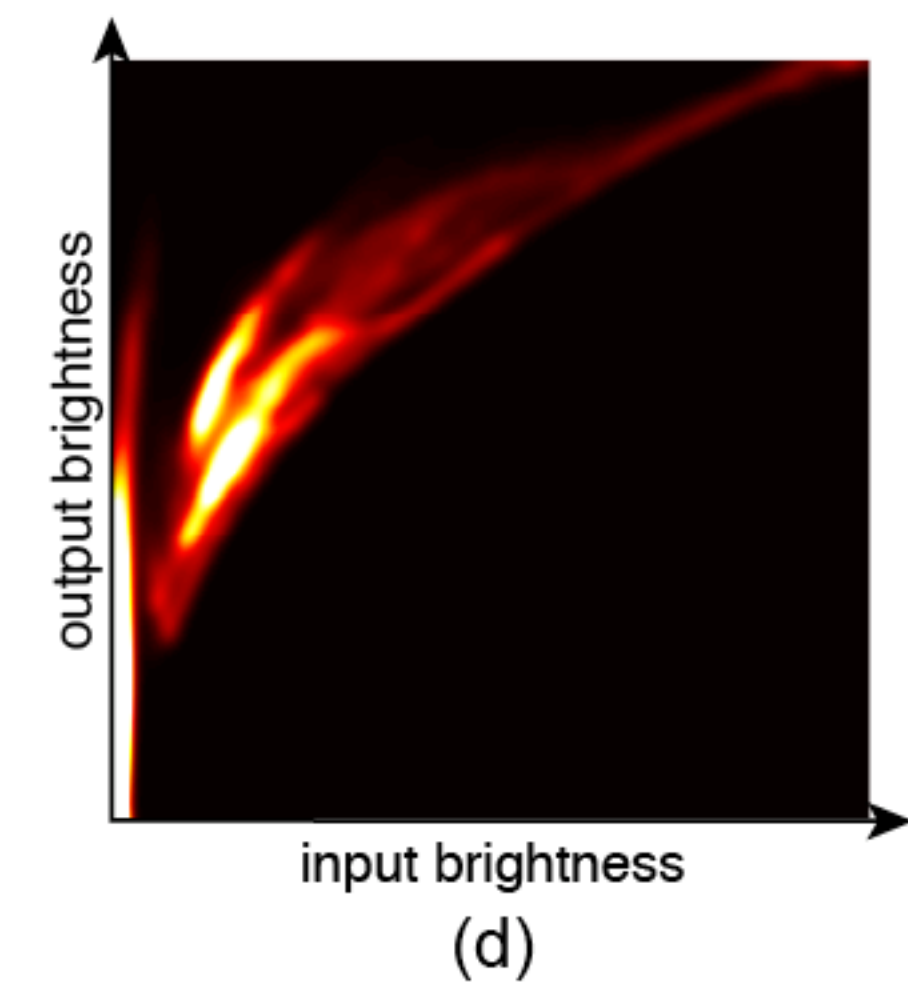
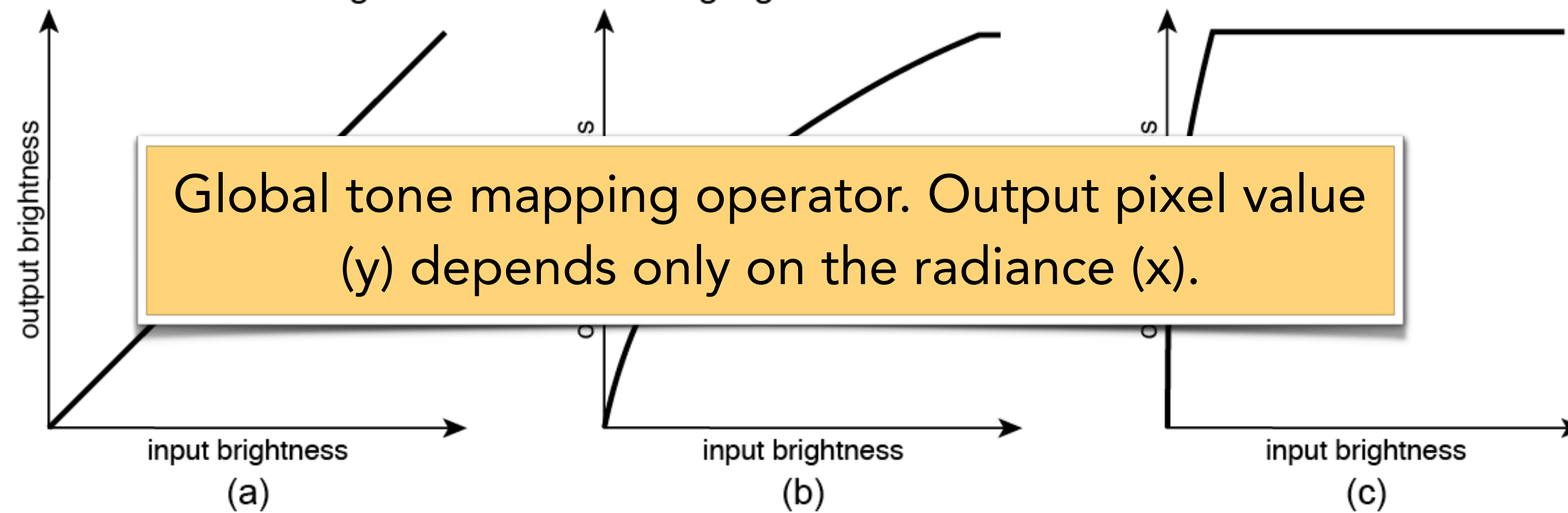
Highlights visible



Shadows visible



HDR+





# Tone Mapping in HDR+



Linear RGB image



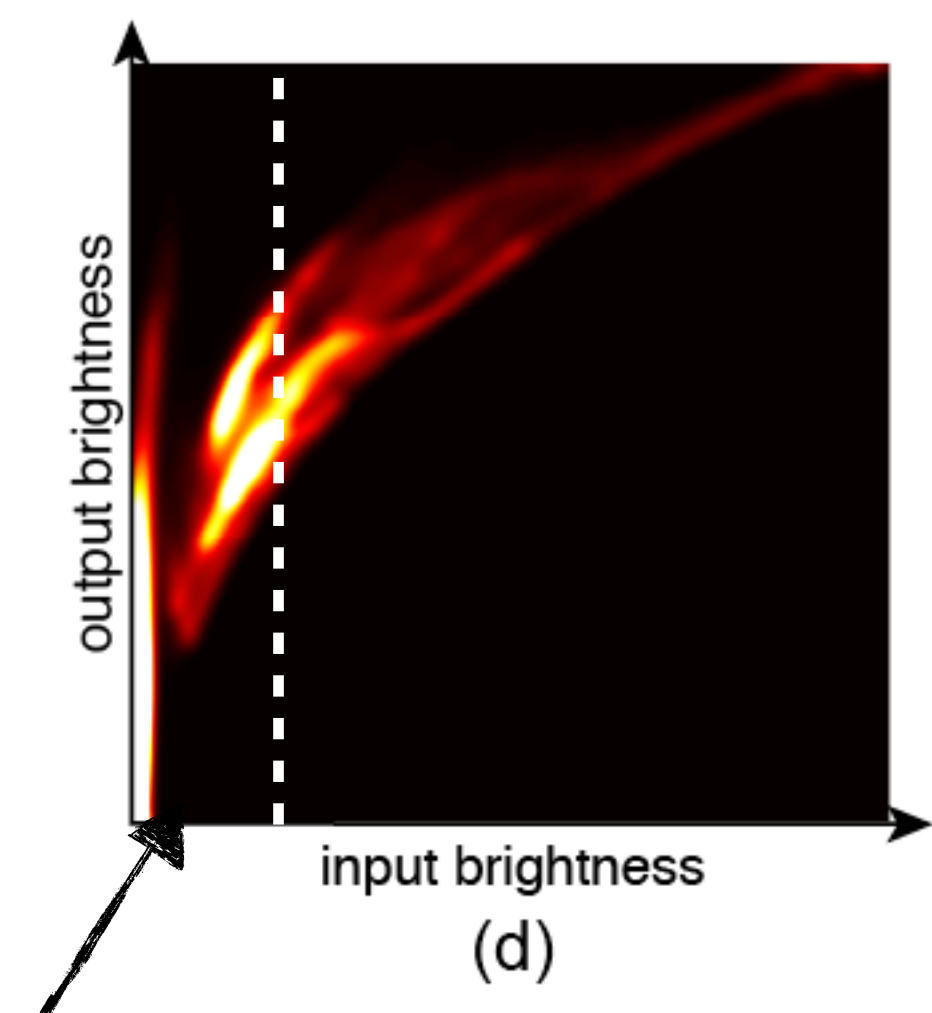
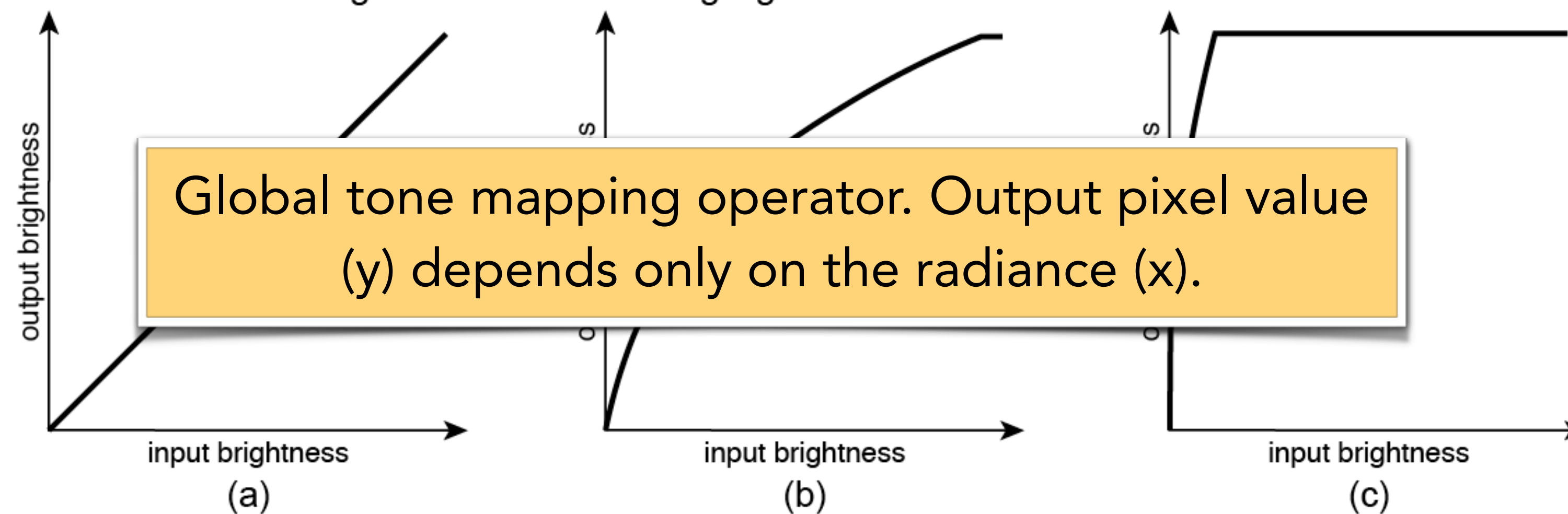
Highlights visible



Shadows visible



HDR+

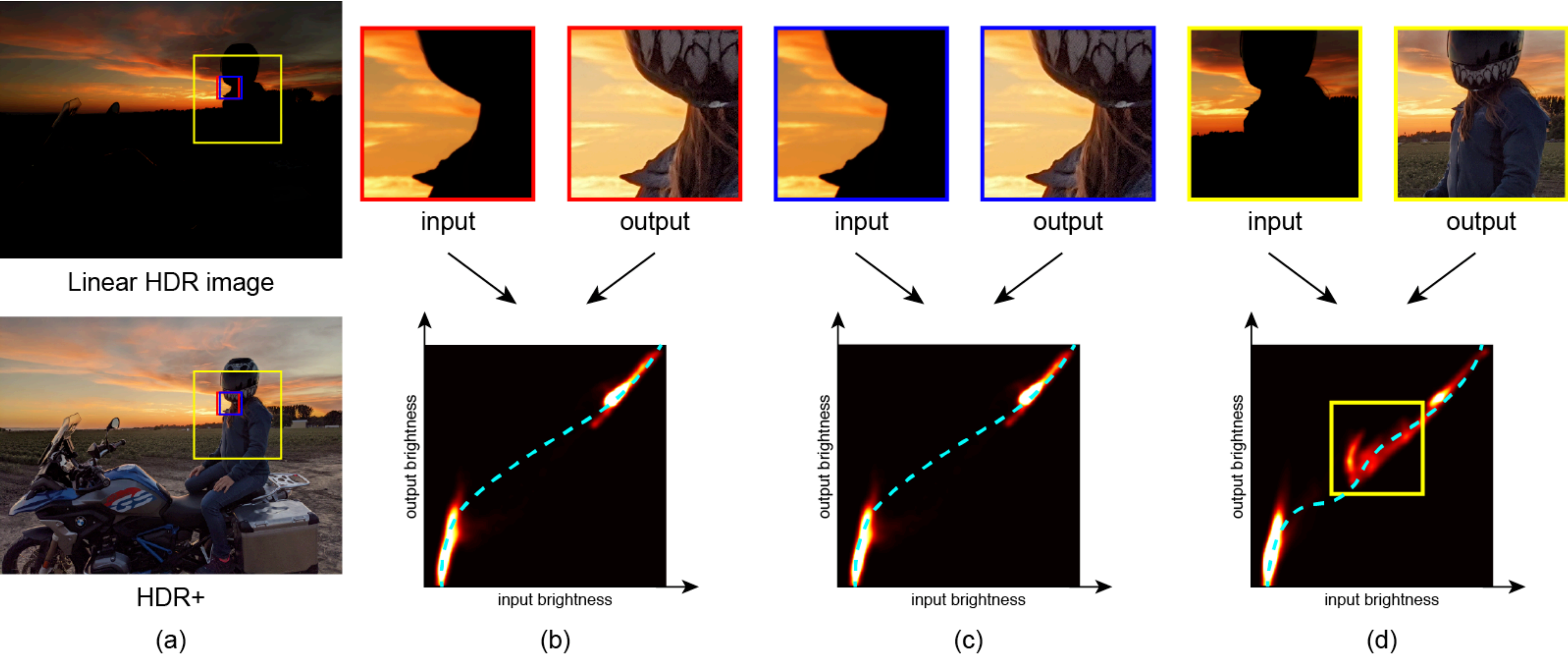


Local tone mapping. Pixels with same radiance could be mapped to different values. Heatmap shows the mapping distribution.



# Live HDR+ on Pixel 4 and Pixel 4a

Goal: produce HDR image in live viewfinder. But full HDR+ algorithm is slow for live. Idea: use (fast) global tone mapping operator in individual small tiles. Per-tile curve is generated by a neural network.





# HDR and Tone Mapping Recap

Two related but distinct issues:

- HDR imaging: reproducing the full range of the (relative) radiance in the scene; limited by cameras. HDR imaging let us preserve the full DR of the scene.
- Tone mapping: how to present HDR images, either generated from HDR imaging or graphics renders, in LDR media (e.g., display/paper).

Typical camera captures, even without HDR imaging, have higher DR than your display; some amount of tone mapping is usually needed.

Tone mapping predates HDR imaging, and was initially used in computer graphics. With global illumination techniques, renders could generate HDR images earlier than HDR imaging algorithms were popularized.



# Deep Learning For Computational Photography

Conventional image signal processing algorithms are empirical: HDR, TM, denoising, demosaicking, color correction, deblurring, supersampling, white balance, etc. Our quest for visually-pleasing photos is endless.

Let a DNN learn to generate high-quality photos.