

COL7655

Foundations of Visual Computing

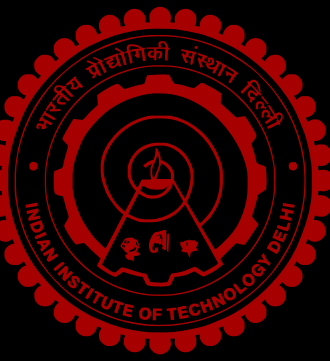
Subodh Kumar

subodh@cse.iitd.ac.in

COL7655

Transforming Intensity

Thanks to Rahul Narain for help with slides

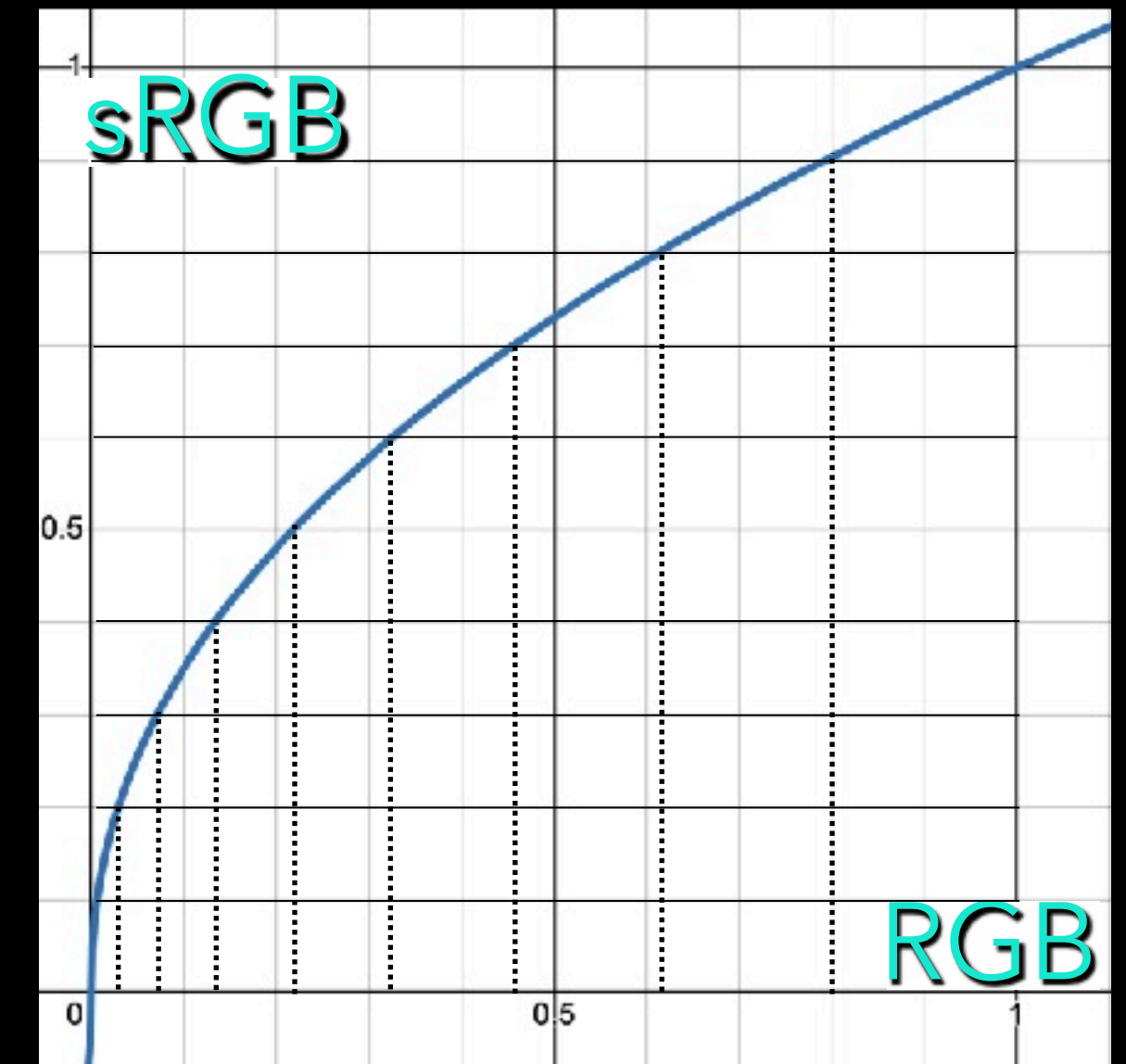


Operating on Intensities

- Intensities can be normalized, real number in $[0:1]$
 - Or, quantized in range $[0:L)$ (Power, or linear) Recall color space and quantization
- Operation may requires conversion between the two
- Operation could result in value outside the range
 - Usually clipped to range
 - Sometimes, we map back to another range by compressing or expanding it

Recall Intensity Quantization

- Monitors have a gamma power response, Number of Photons for value $v \in [0,1]$
 - $I \propto (I_{max} - I_{min})v^\gamma + I_{min}$
 - i.e., for $v = 0.5$, fewer than 50% of the maximum photons are produced
 - That appears perceptually halfway between *min* and *max*
- Images stored in files could be linear RGB or non-linear sRGB ($RGB^{\frac{1}{\gamma}}$)
 - Check image color profile



Intensity Transformation

Usually only at pixels

- Transform the intensity of each (x,y) : $g(x,y) = T(f(x,y))$
 - Used to adjust tone, contrast, brightness, saturation, etc.

e.g., $T(r) = r + b$



Monitor contrast c and brightness b adjustment: $T(r) = cr + b$

Example: Contrast control: $T(r) = cr$



$c = 0.5$



$c = 2$

Example: Brightness and (centered) contrast adjustment (scales values from selected anchor color k)

$$T(r) = c(r - k) + k$$

(e.g., $k = 0.5_{normalized}$ or $L/2$, where L is the largest intensity value)



$c = 0.5$



$c = 2$

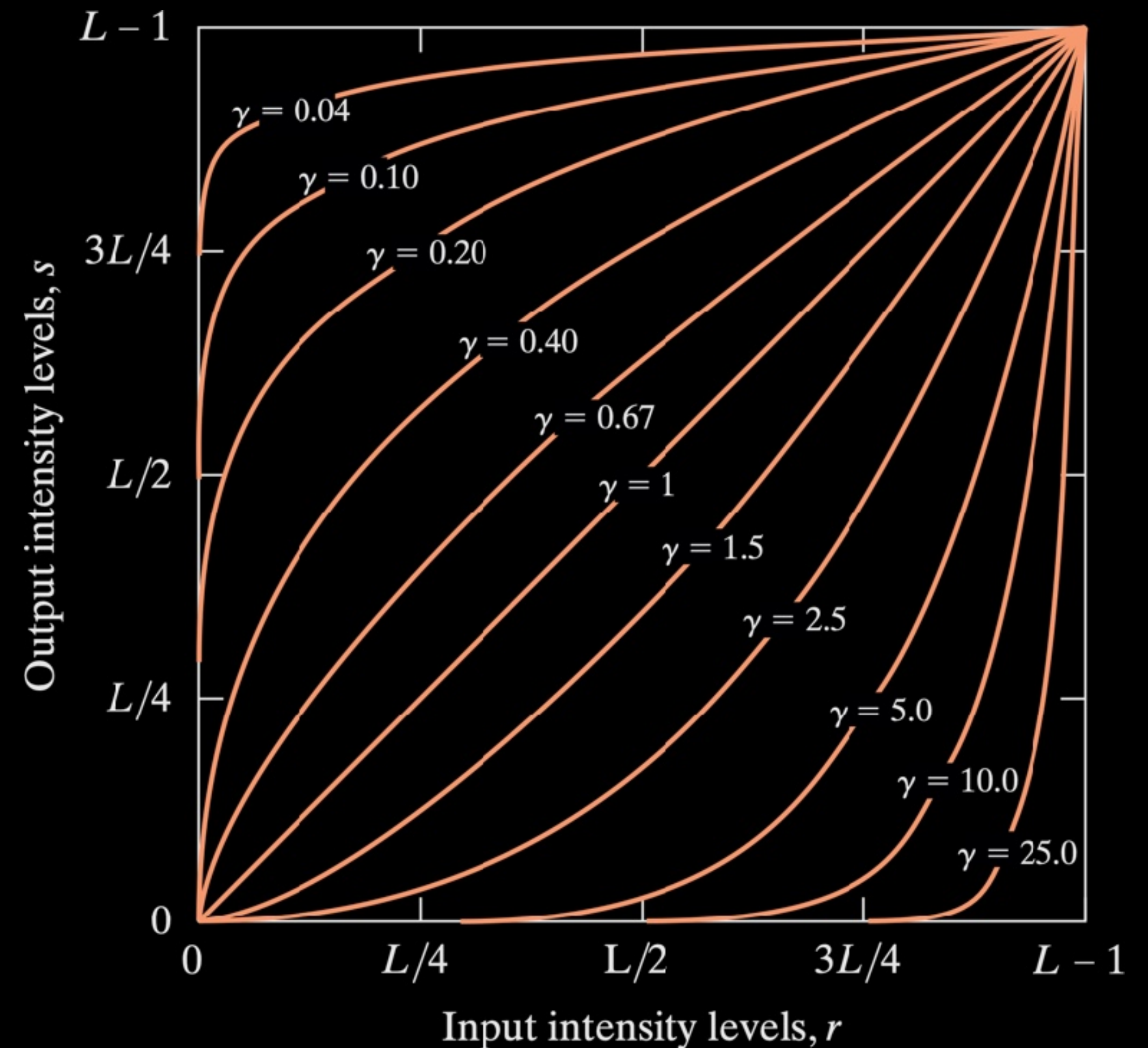
Power-law Transformation

Gamma correction

$$T(r) = r^\gamma$$

with intensities are in range $[0, 1]$

- Hardware (camera, monitor, printers) usually have a power law behavior
- $\gamma \approx 2.2$ perceptually more uniform
 - $\gamma \approx \frac{1}{2.2}$ allows better quantization of dark values
- Linear color assumed for filtering and other linear transformations





$\gamma = 1/3$



$\gamma = 1$ (identity)



$\gamma = 3$



Power-law Transformation of Color

Chromaticity can be affected

$R = 0.97$
 $G = 0.79$
 $B = 0.46$

$\gamma = 1/3$



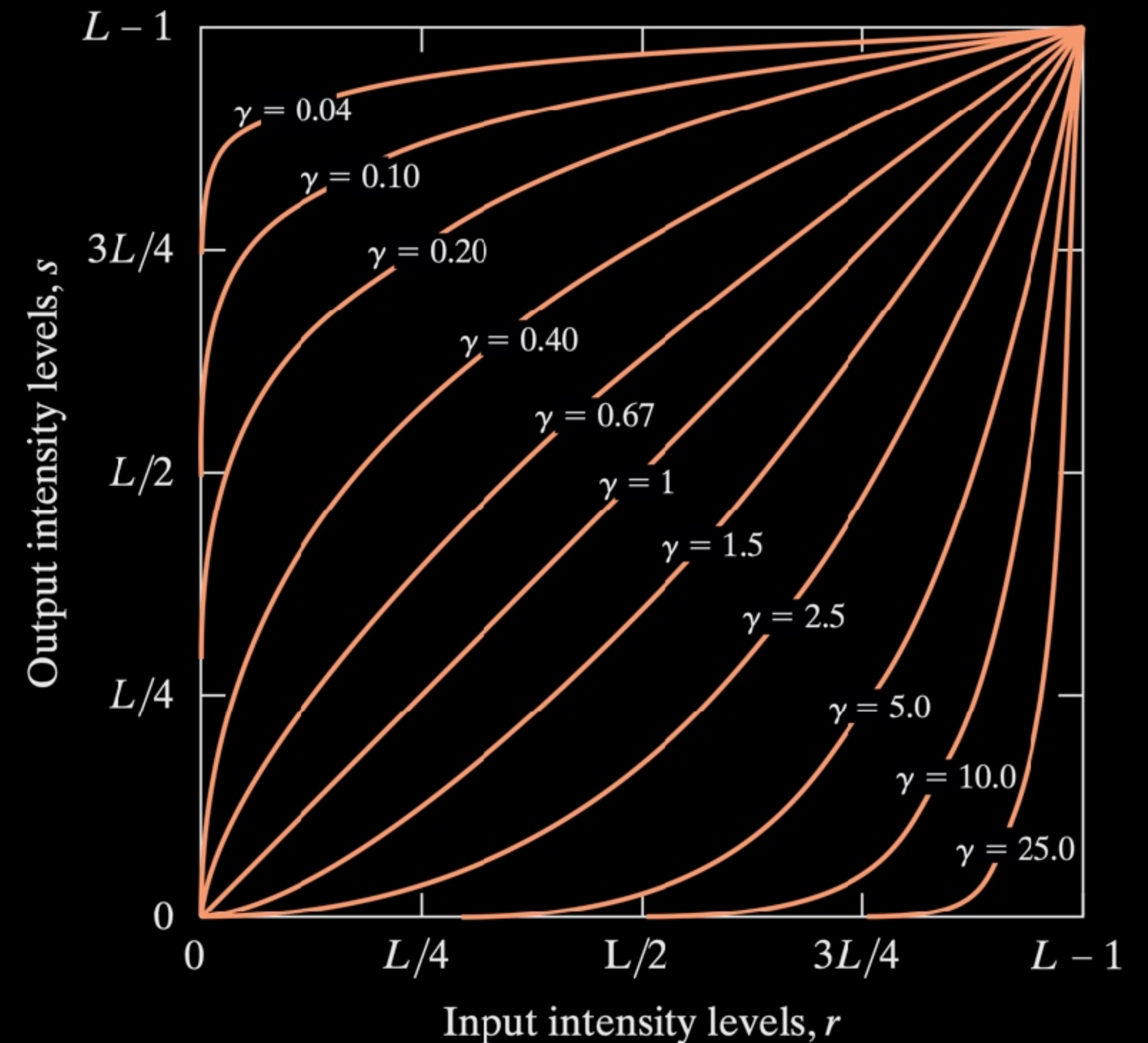
$R = 0.90$
 $G = 0.50$
 $B = 0.10$

$\gamma = 1$



$R = 0.73$
 $G = 0.13$
 $B = 0.00$

$\gamma = 3$



Power-law Transformation of Luminance

Chromaticity can be affected

- $T(Y) = Y^\gamma$, Luminance Y

- $T(r) = r * \frac{T(Y)}{Y}$

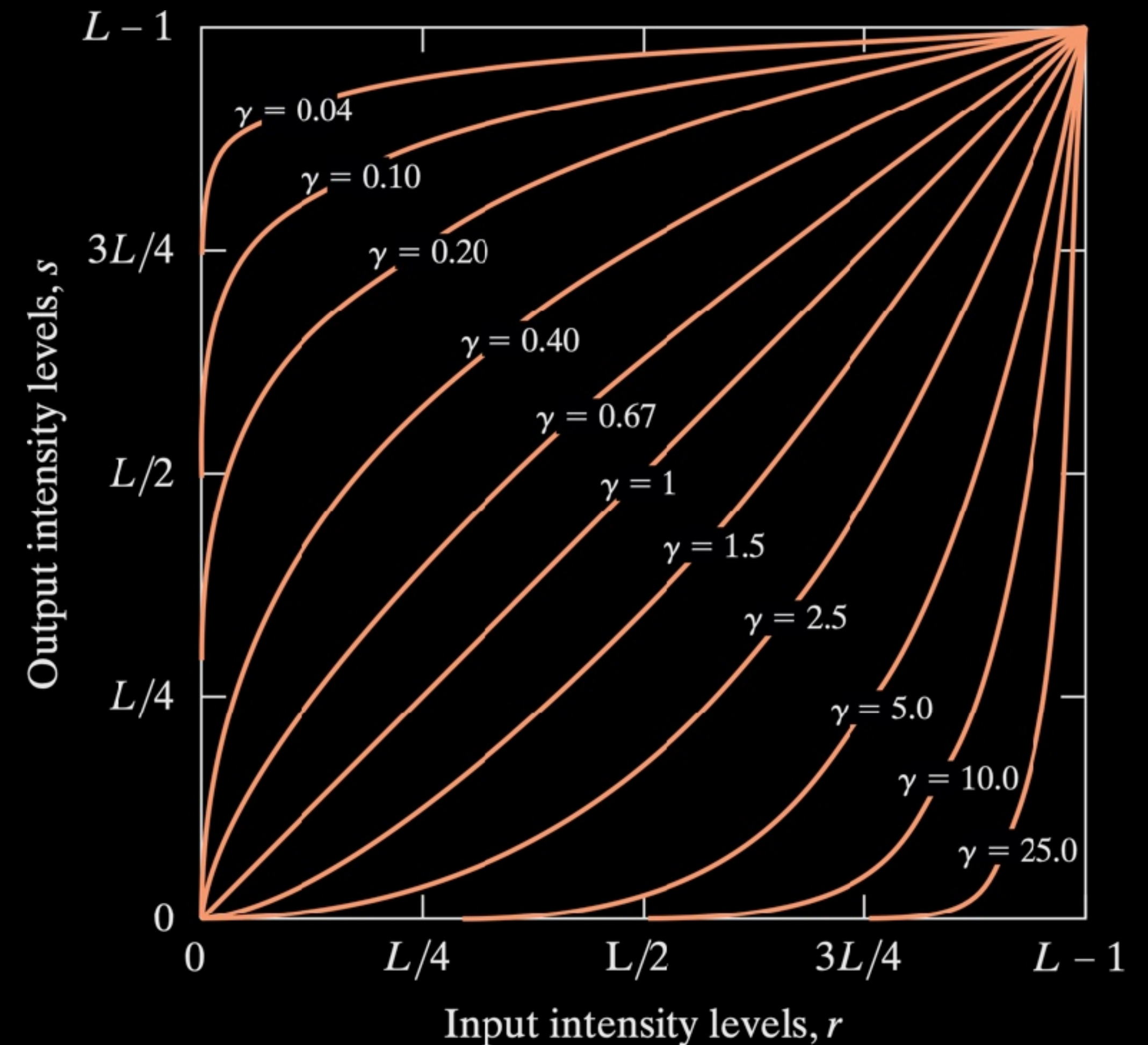
$\gamma = 1/3$



$\gamma = 1$



$\gamma = 3$



Same technique can also be used for contrast enhancement



Full Color Processing

- Component by components (Convolution, Laplacian)
 - OK for linear operations
- Only modify intensity/Luminance (Min, Max, Median, Scale, Sobel, Laplacian)
 - L of L*a*b is the most used for perceptual reasons, HSI is sometimes acceptable
 - Occasionally, also scale the chromaticity according to intensity scale
- Use color vector distances and gradients (Median, Bilateral, Laplacian)
 - Geometric median in color space $\approx$ one with the smallest total distance to the others

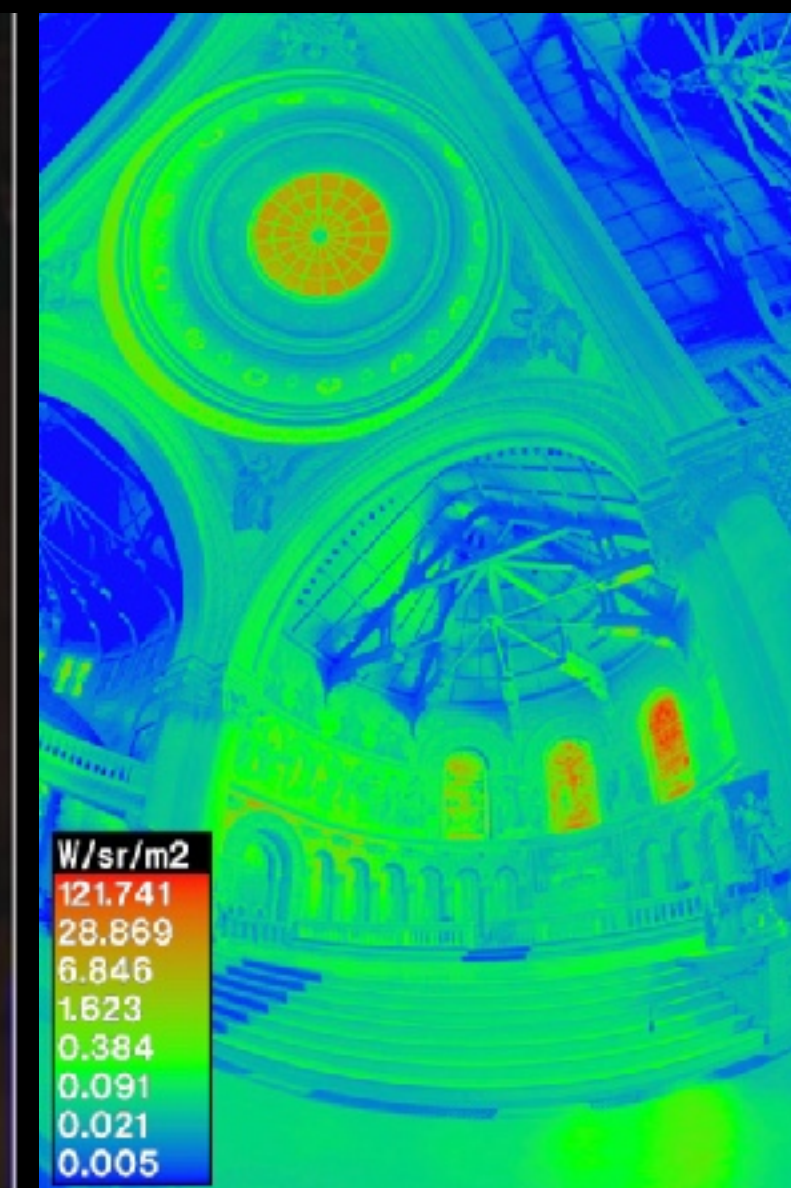
Log Transformation

- Compresses large range into small
 - High dynamic range (HDR) images
 - Fourier spectra
- Usually need to normalize first so intensities are in $[0, 1]$, then compute T , and finally rescale back to $[0, L)$

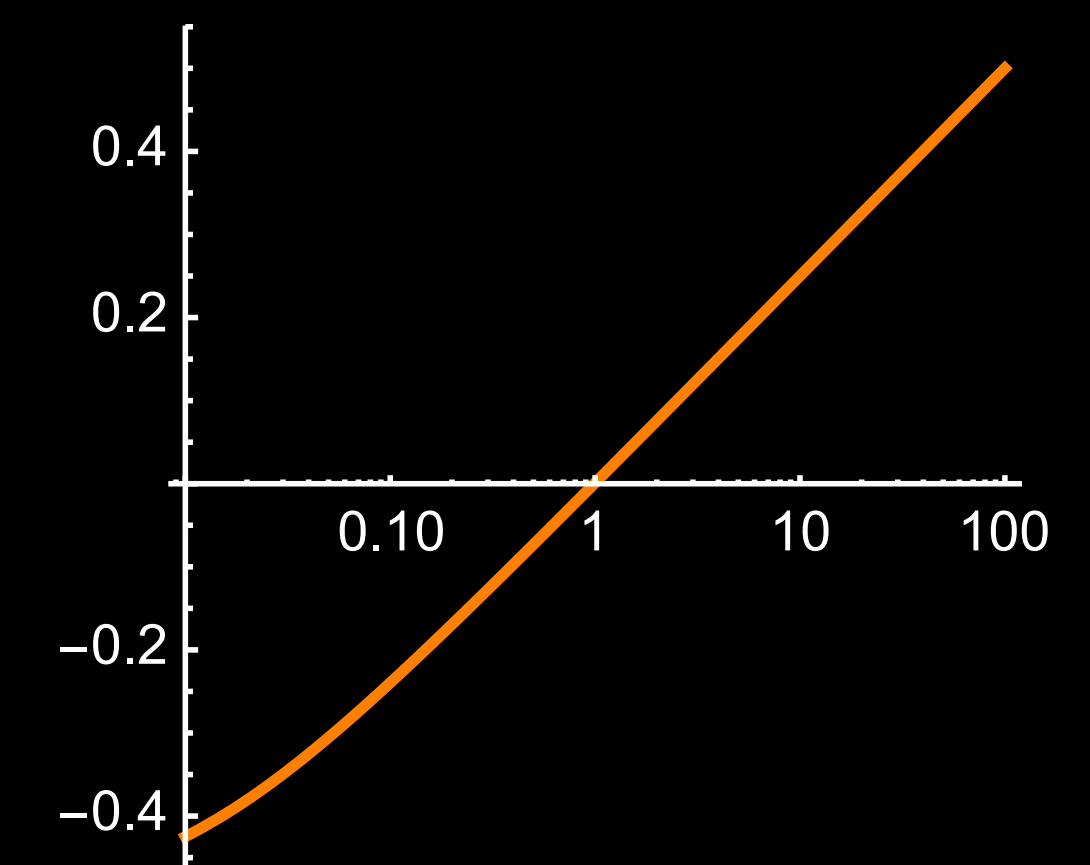
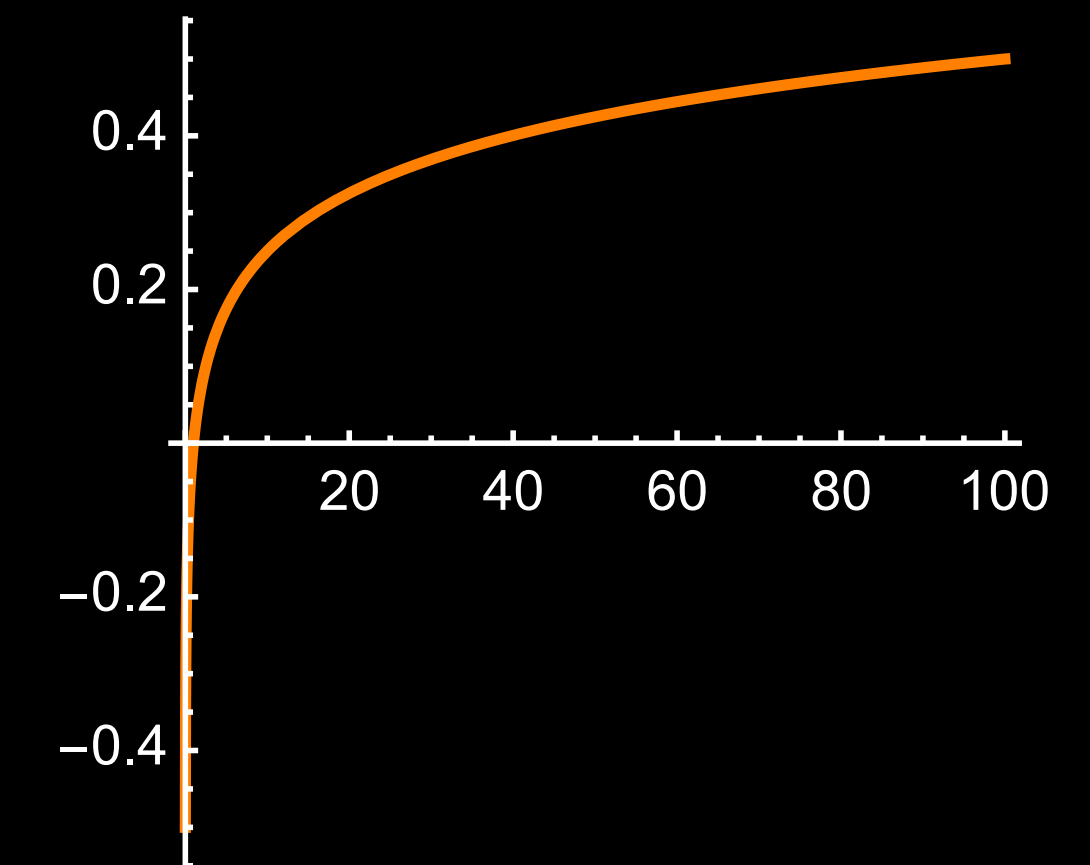
$$T(r) = c \log(r + \epsilon)$$

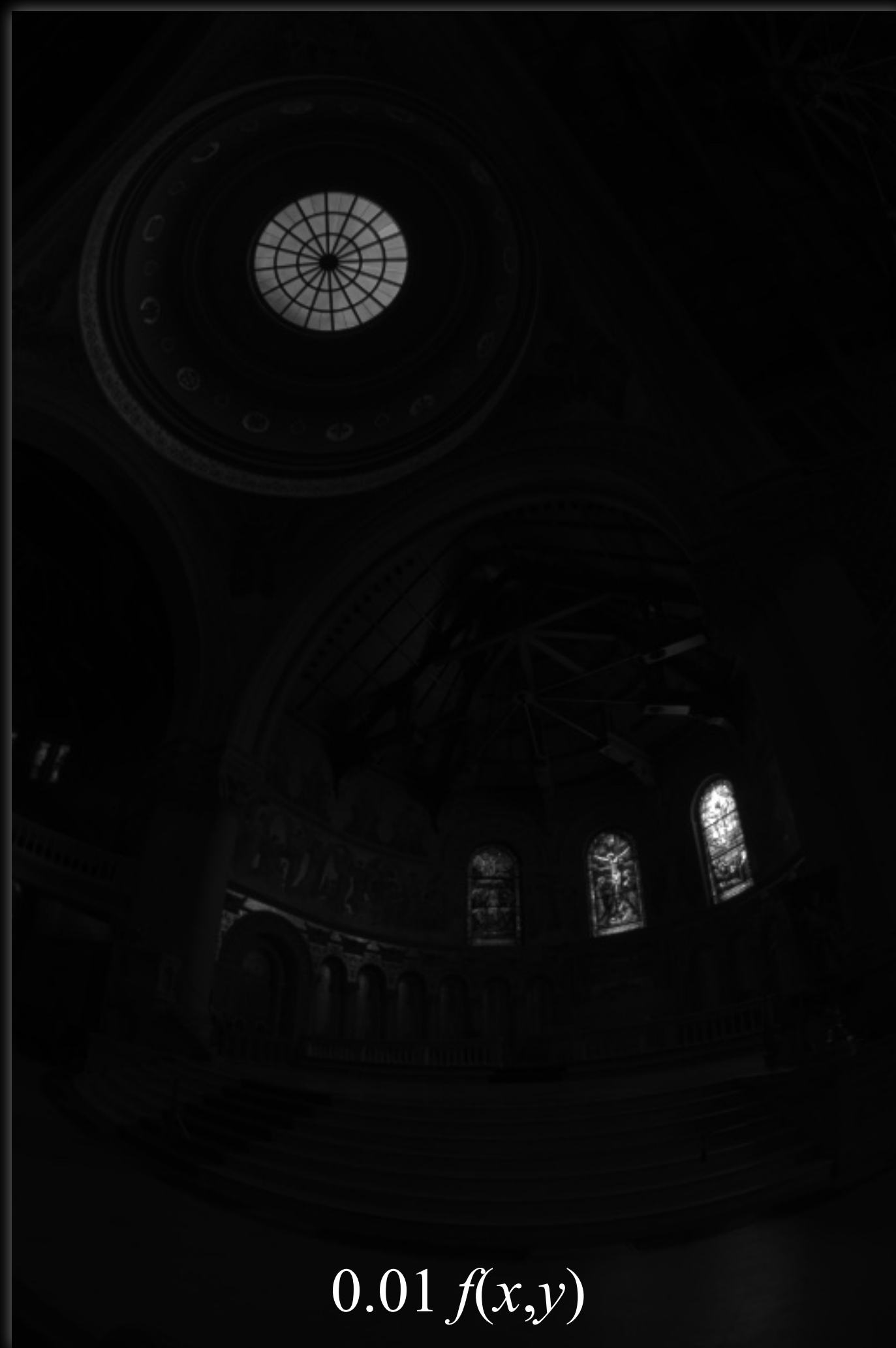


Rendered Image



Heat-map of Radiance





$$\min f(x,y) = 6.85 \times 10^{-4}, \max f(x,y) = 2.06 \times 10^2$$

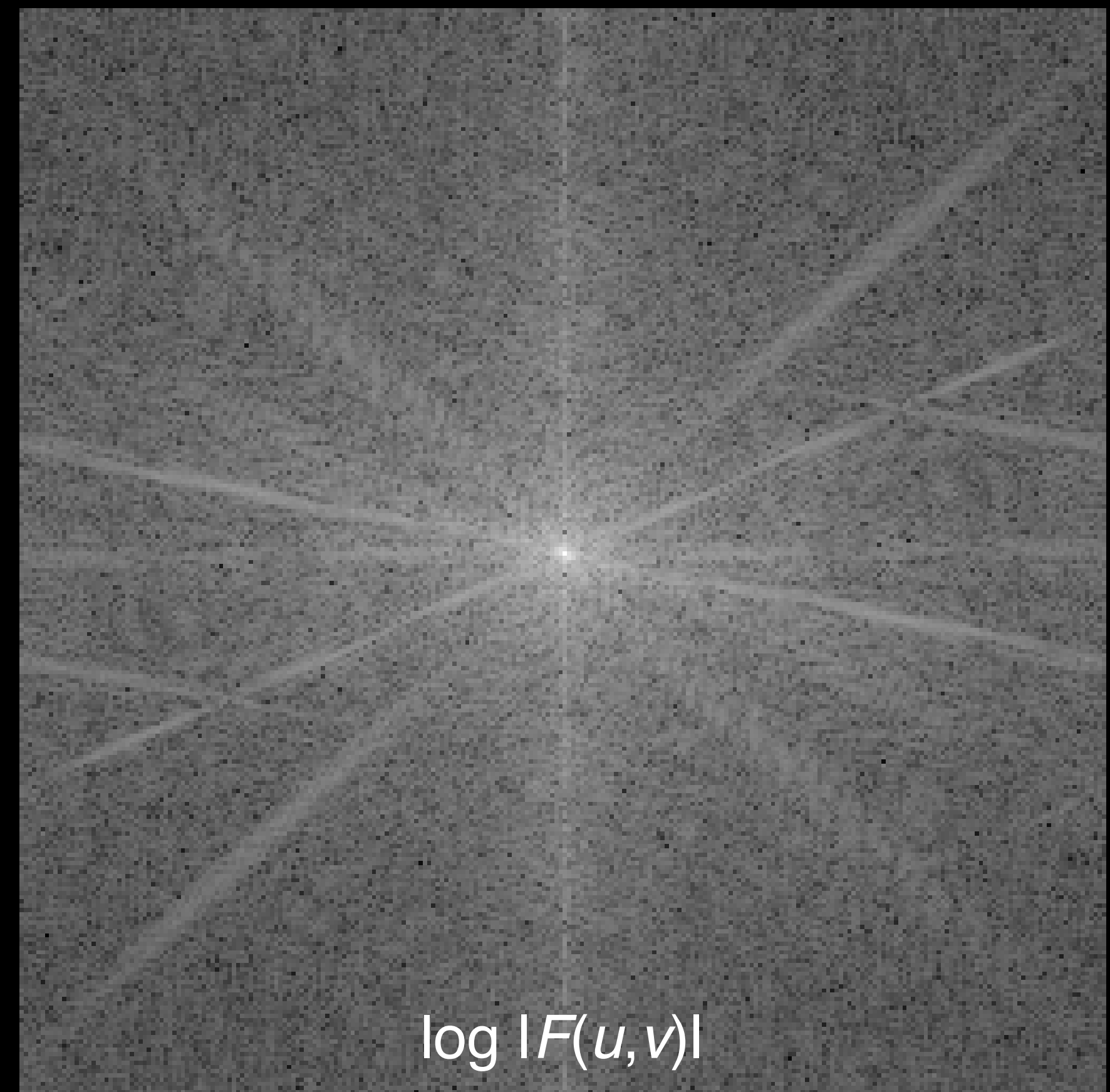


$f(x,y)$



$\log f(x,y)$
(after rescaling)

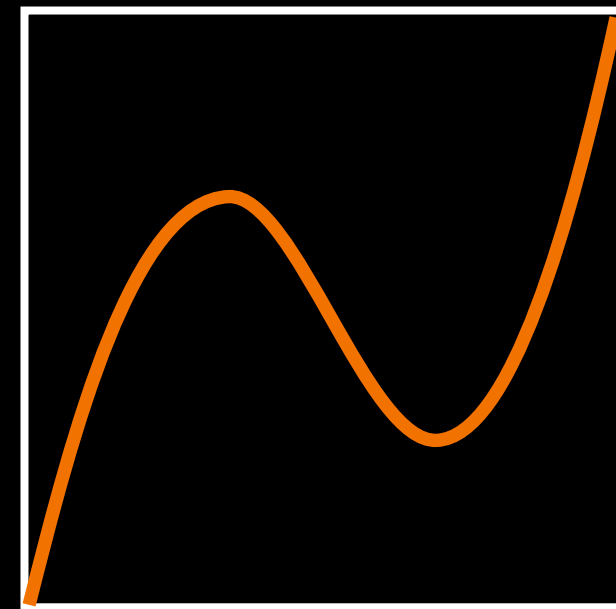
Example: Fourier spectrum of an image



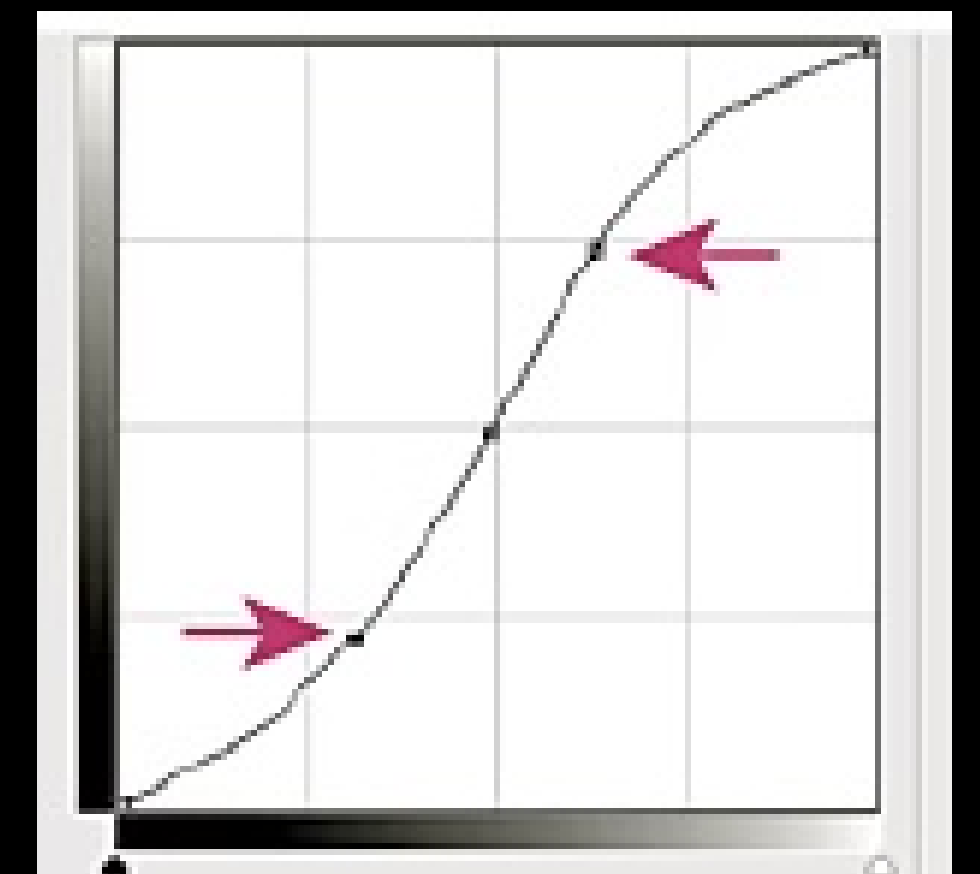
$$\min |F(u,v)| = 2.21 \times 10^{-3}, \max |F(u,v)| = 9.31 \times 10^8$$

General Transformation Curves

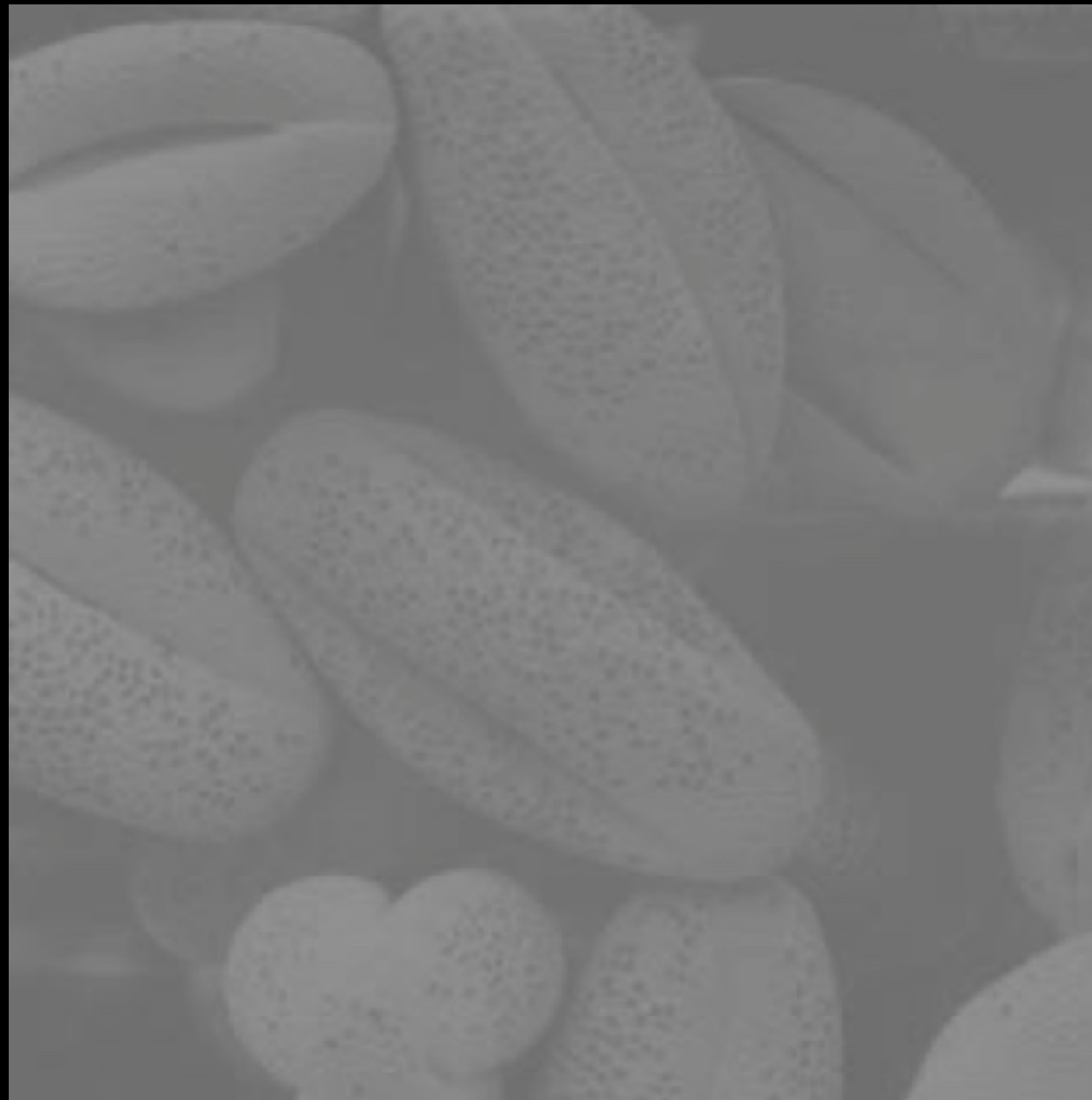
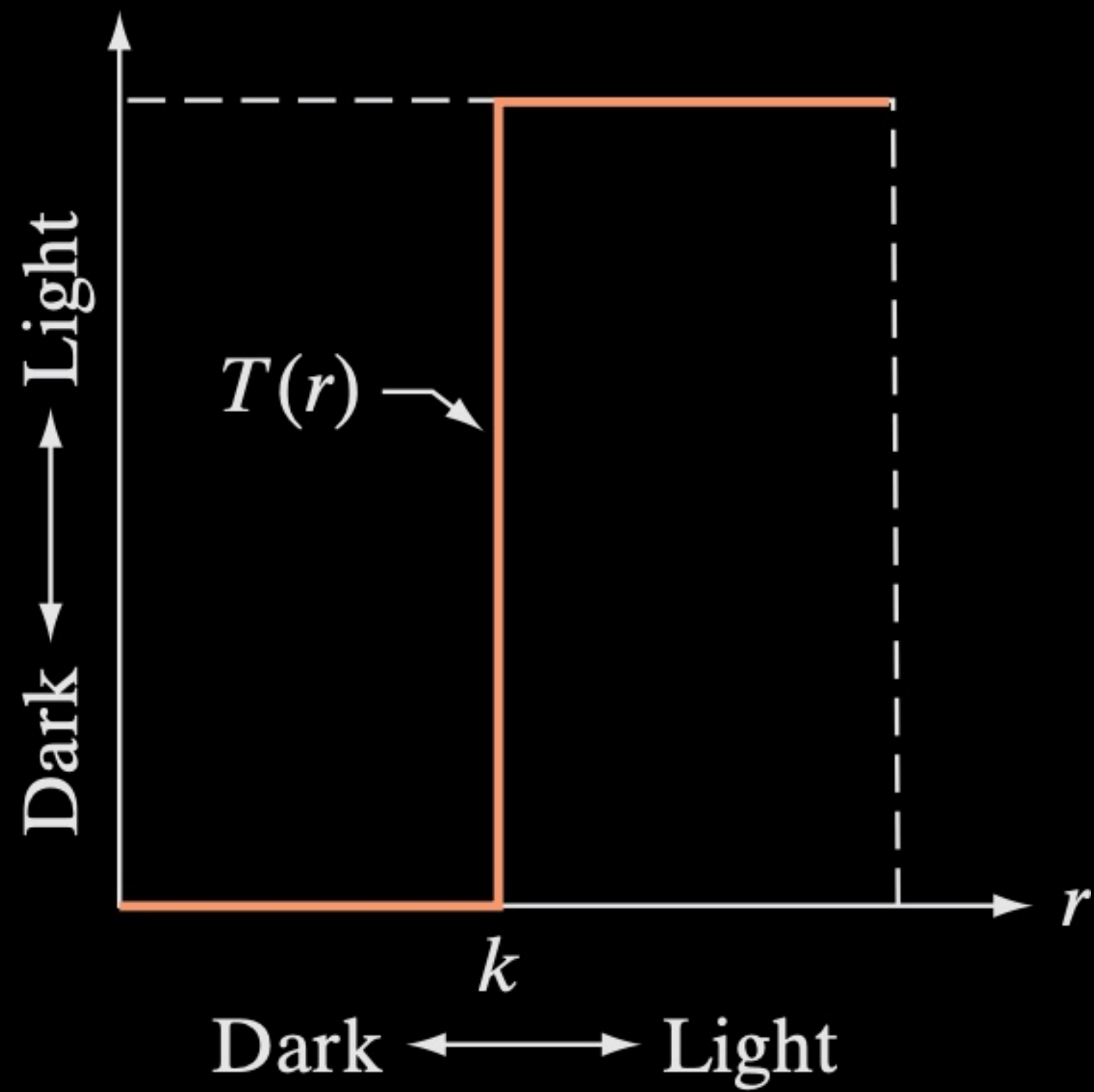
- $T(r)$ can be any function from $[0, L)$ to $[0, L)$, but usually should be monotonically increasing
- Otherwise, inversion artifacts can occur



- Typical way to construct:
 - User specifies input-output table for some i , $T(r_i) = s_i$
 - Compute all T by interpolation



Example: Thresholding



Example: Contrast stretching

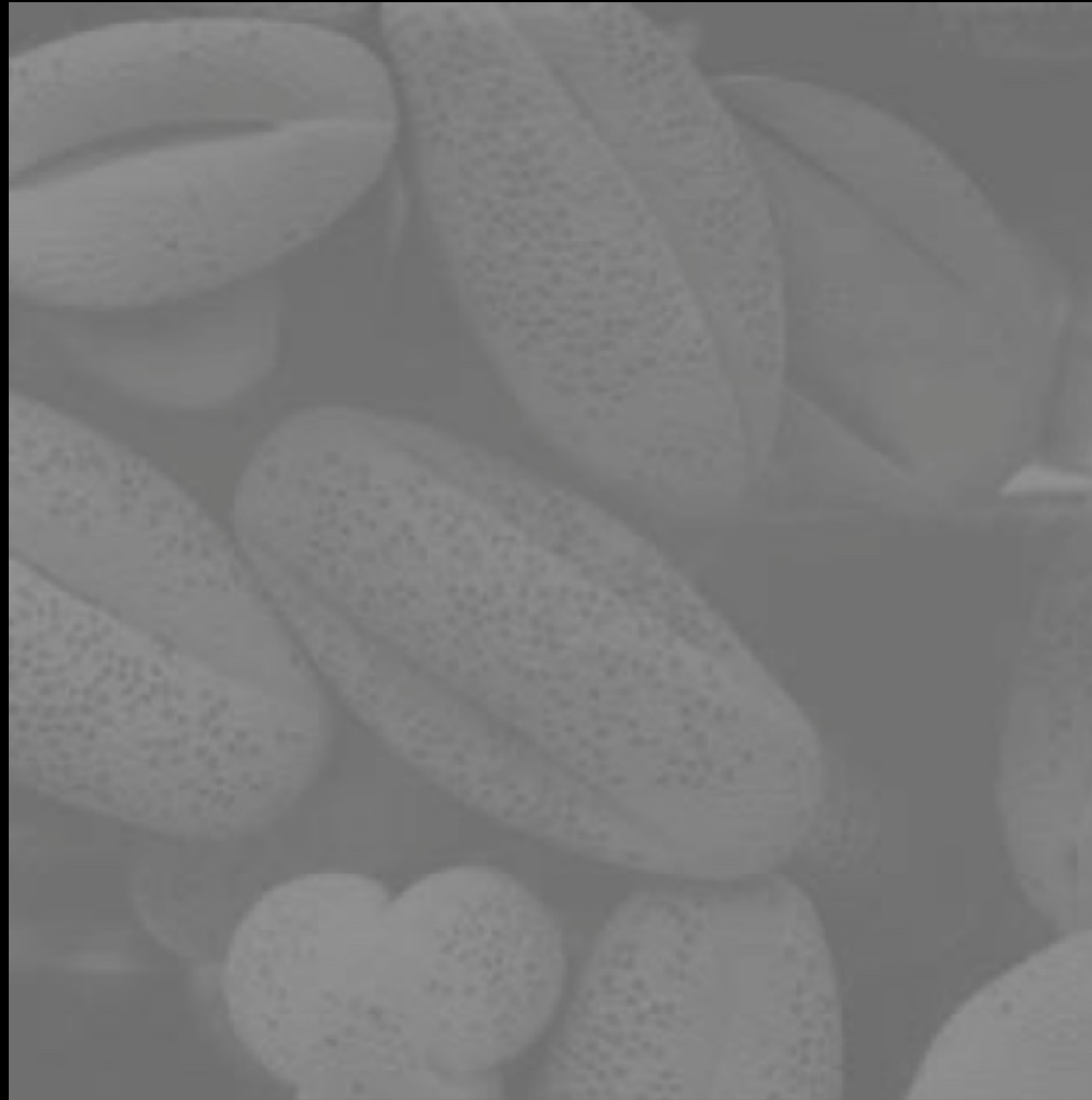
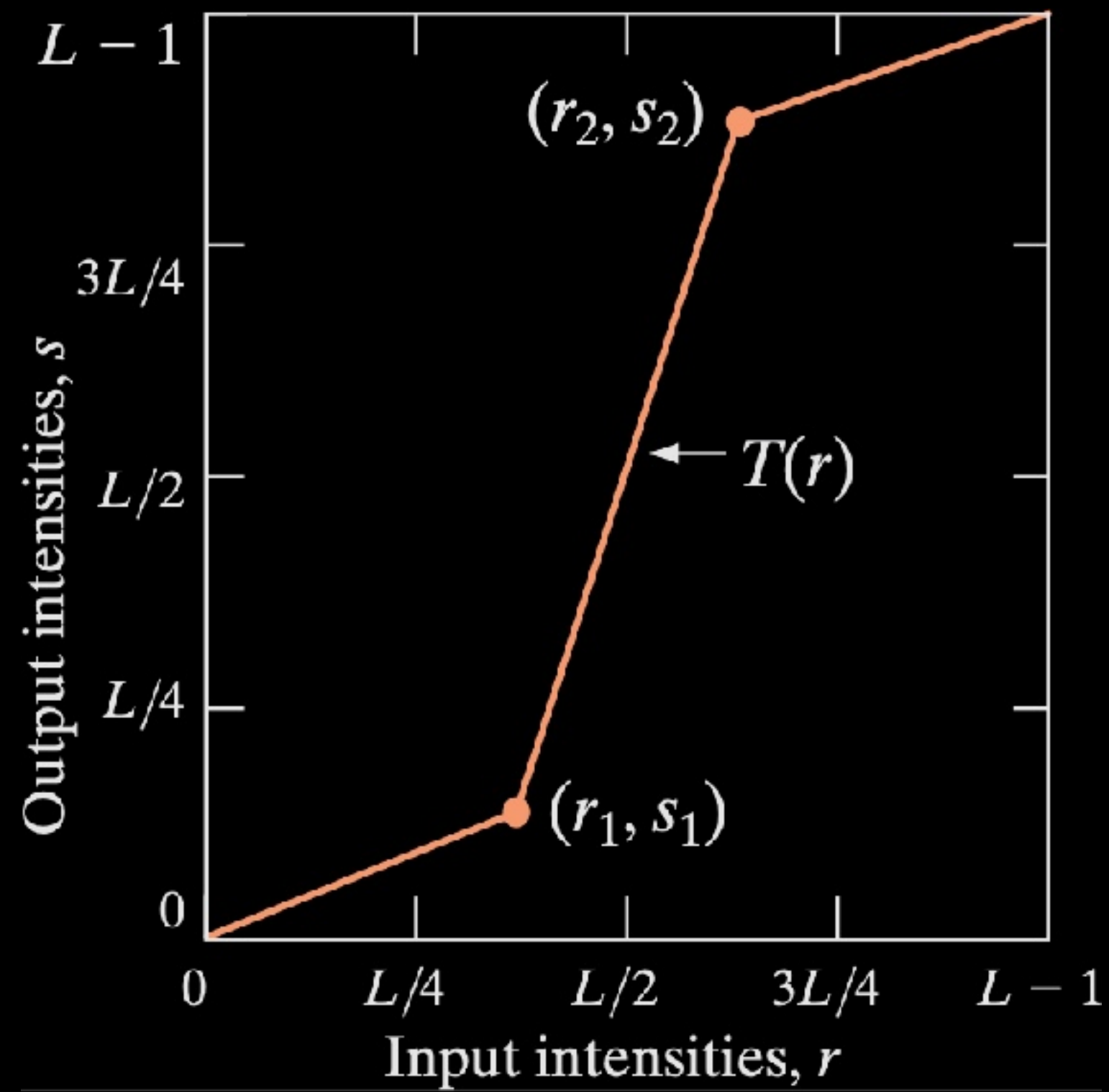
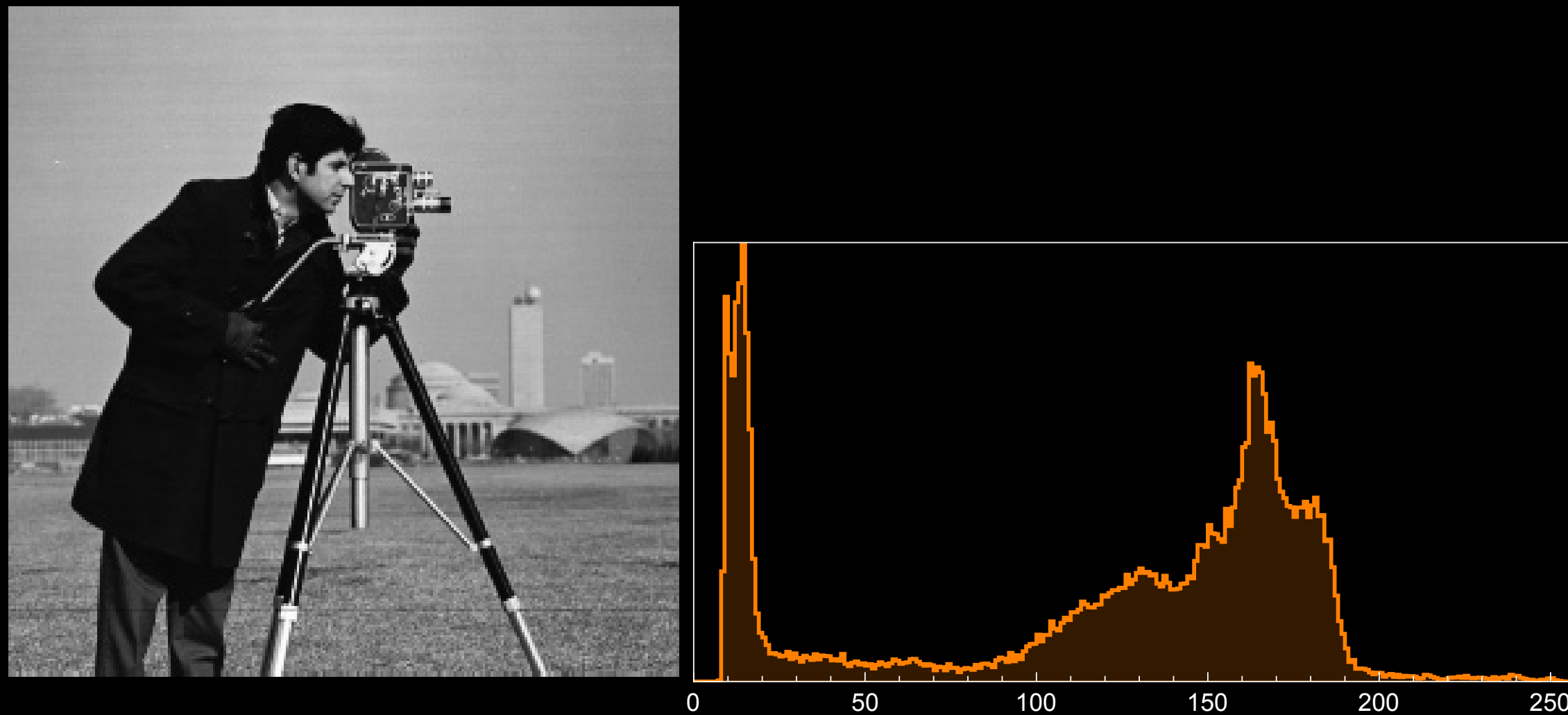


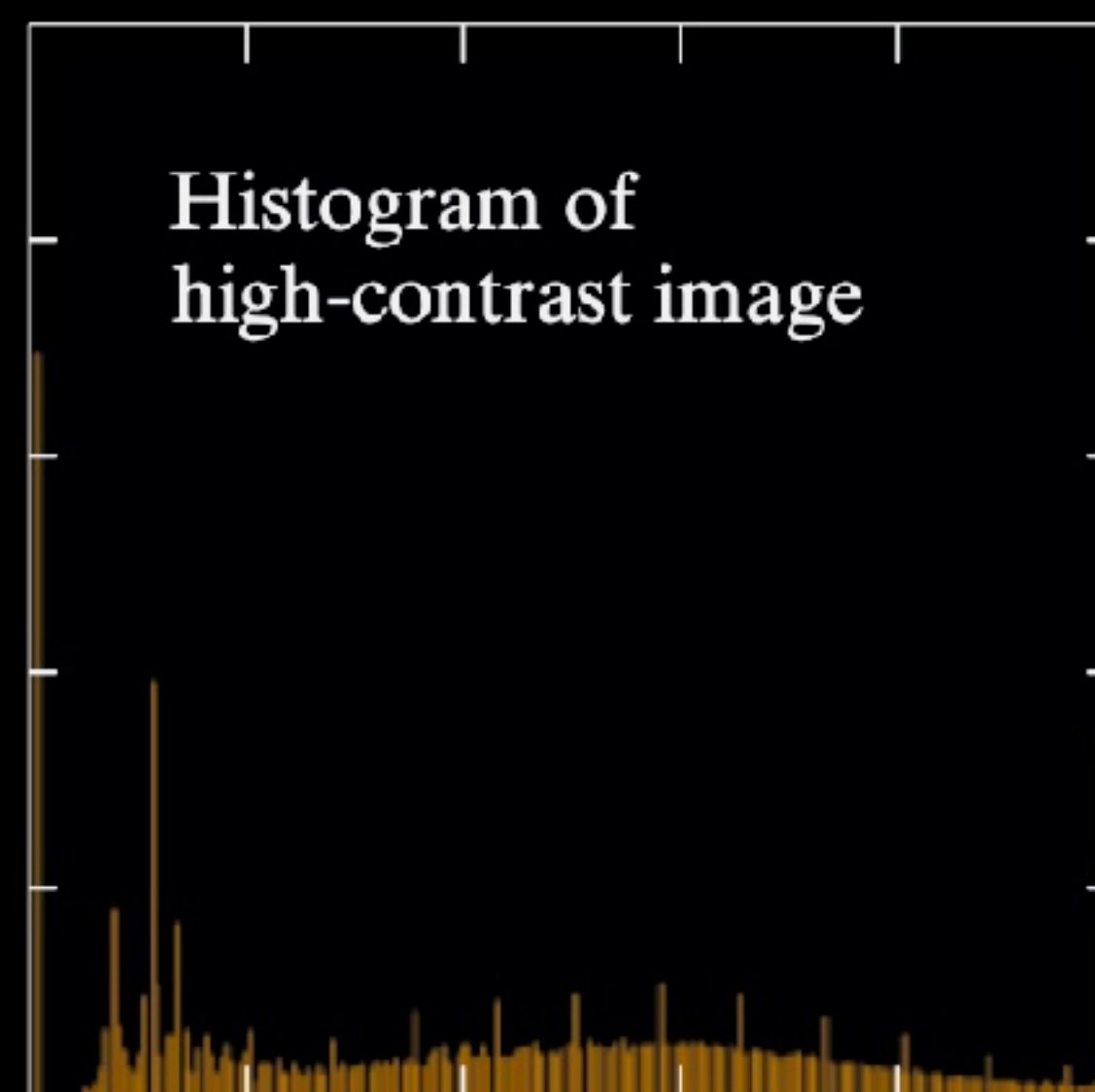
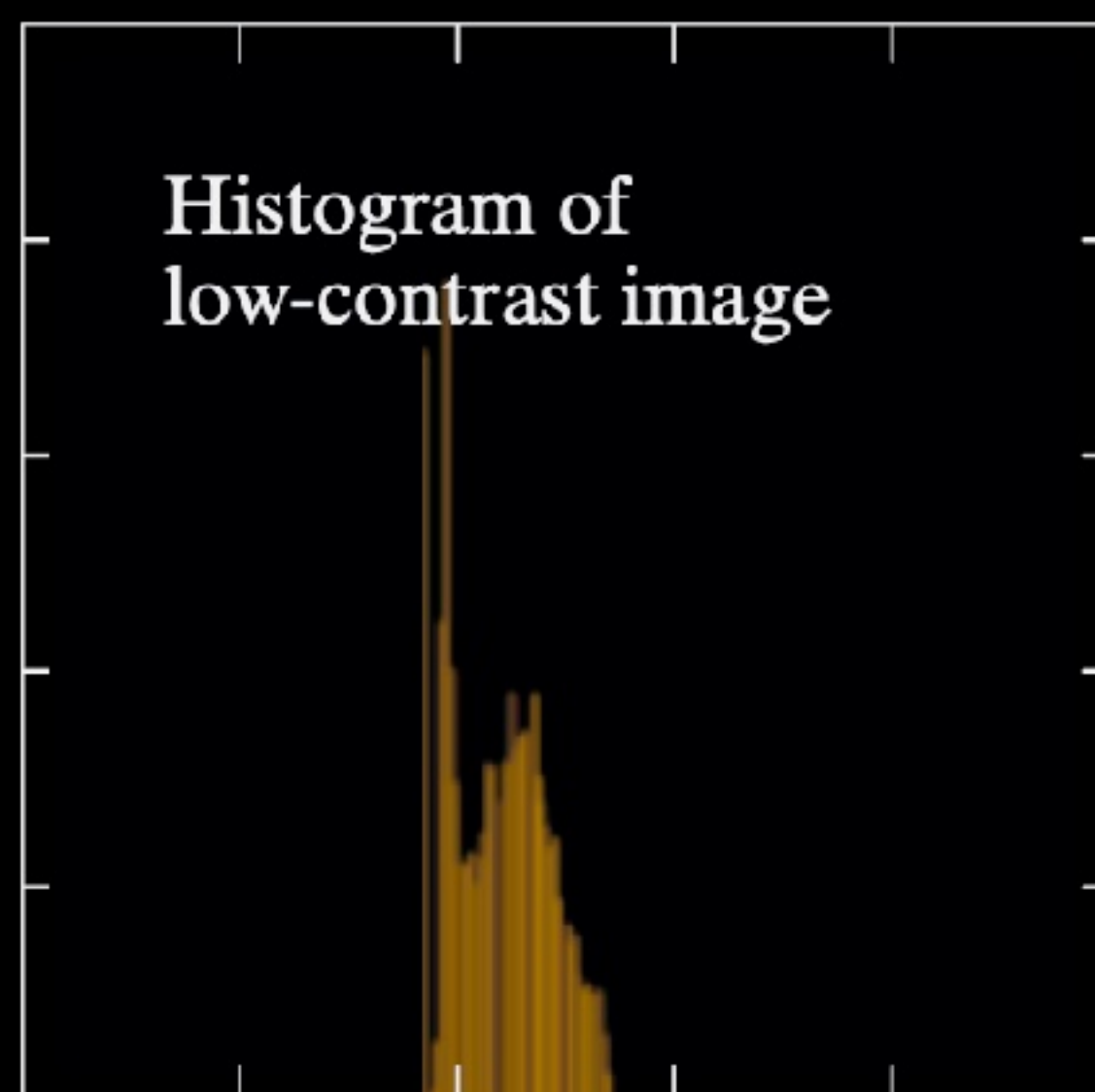
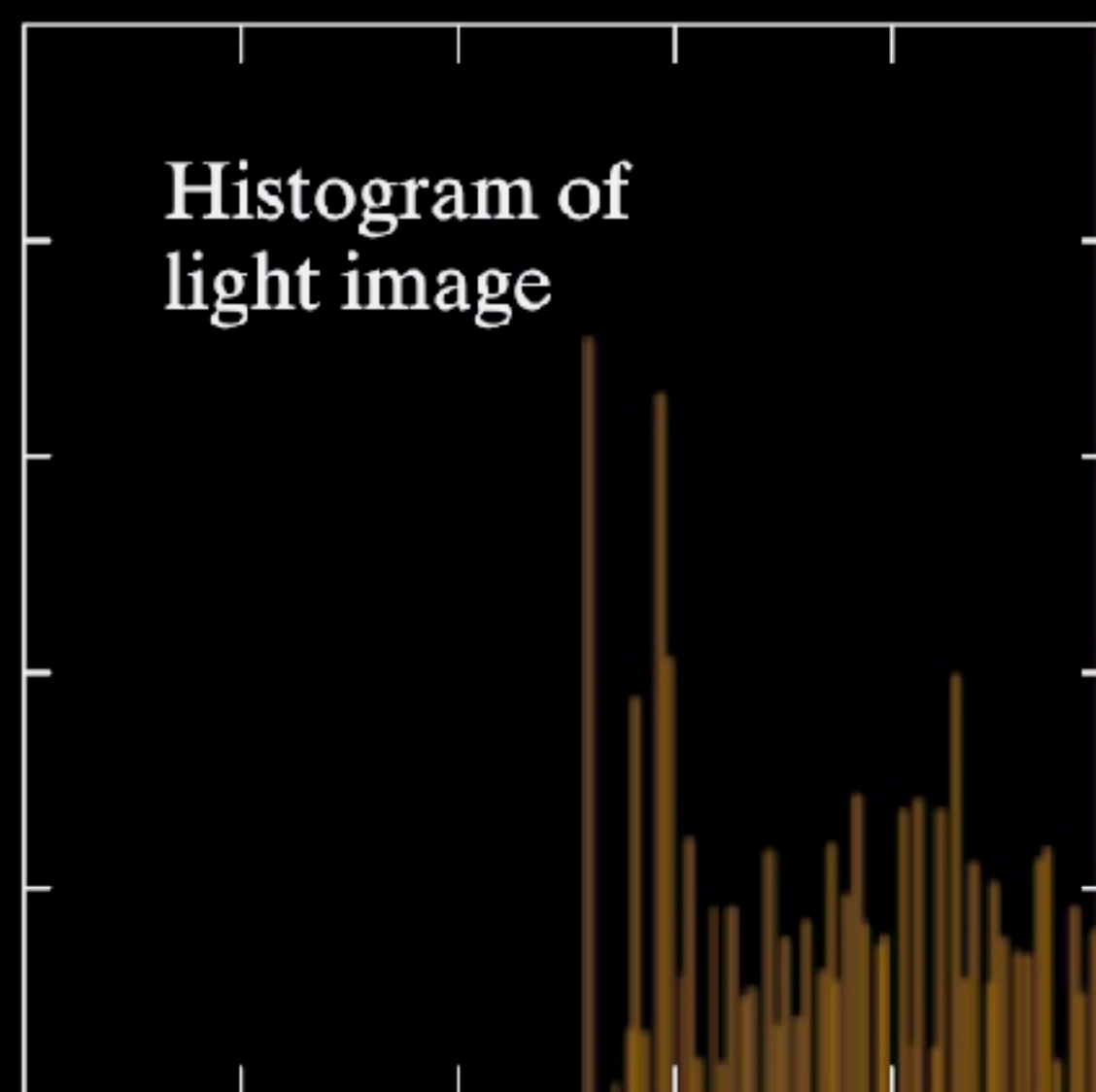
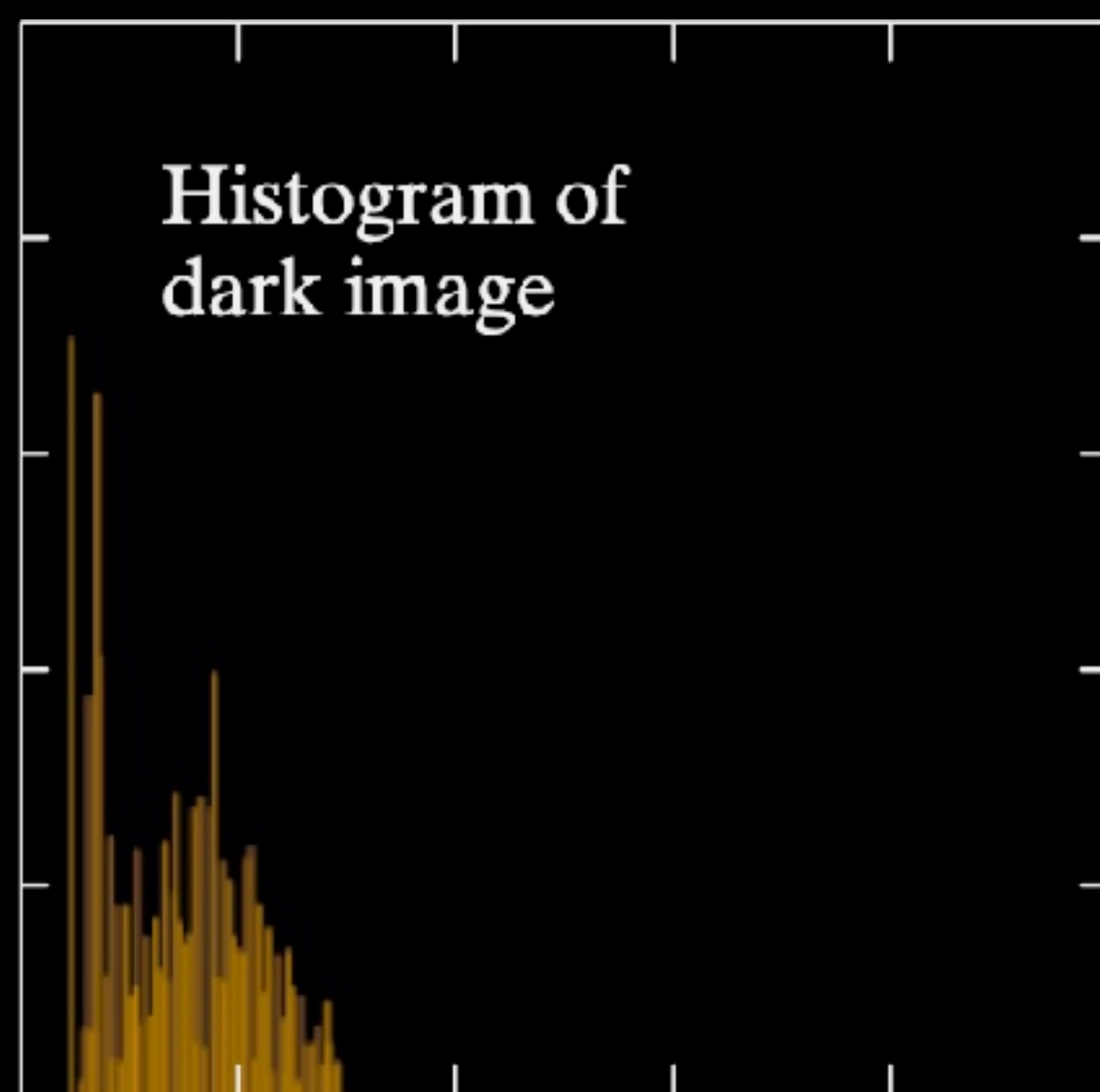
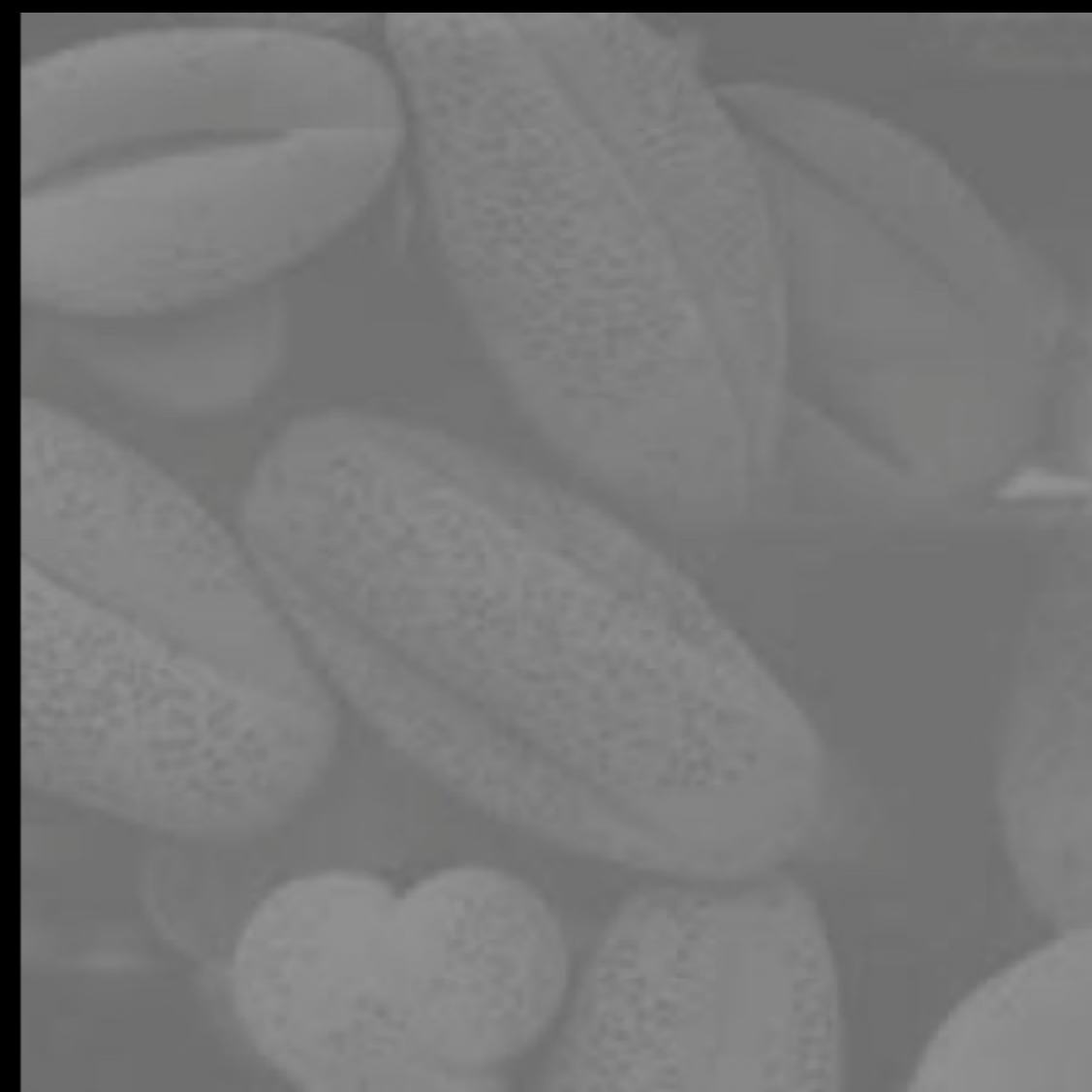
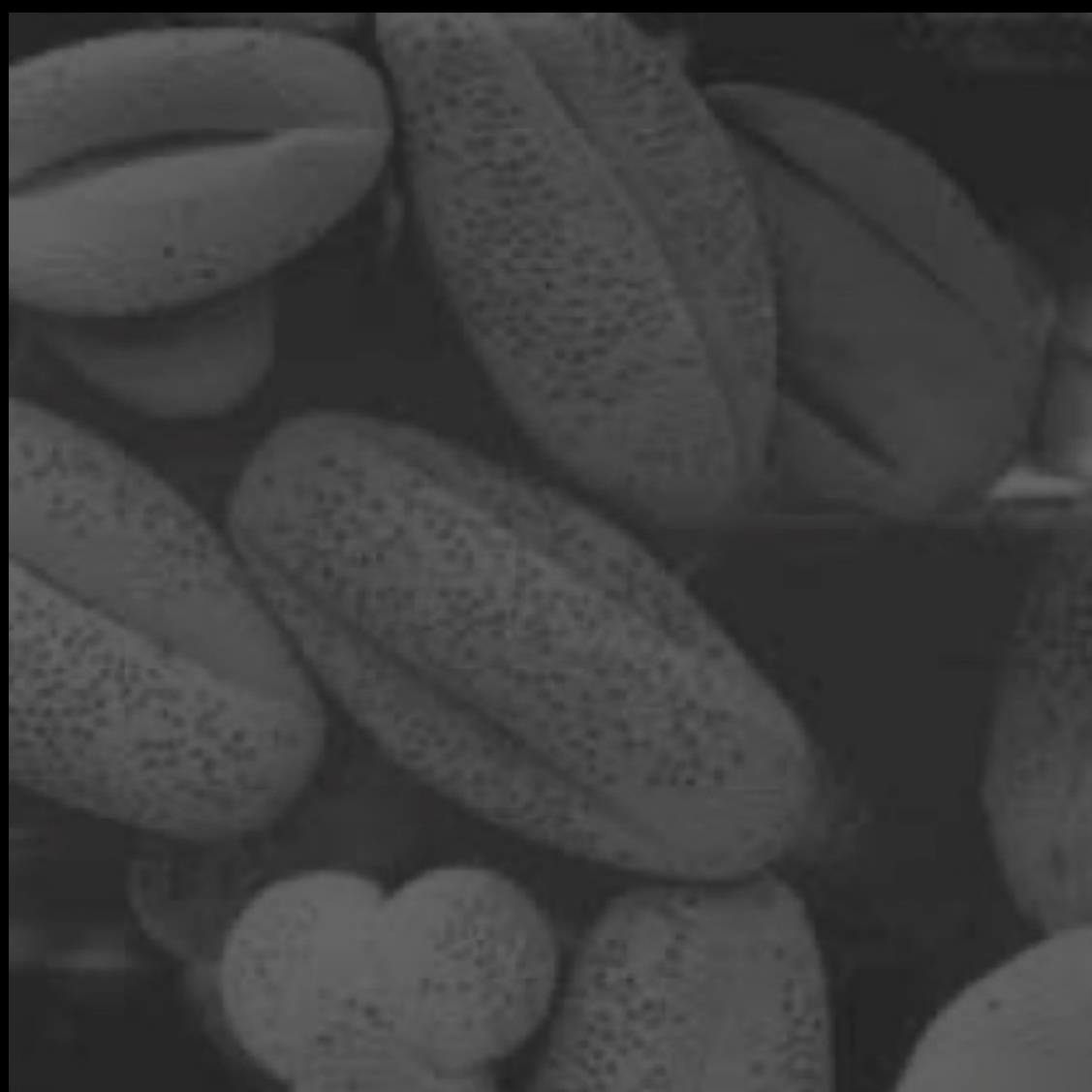
Image Histogram

- **Unnormalized:** $h(r)$ = number of pixels (x, y) with intensity r
- **Normalized:** $p(r) = \frac{1}{MN} h(r)$, for $M \times N$ image
- $p(r)$ = probability that randomly chosen pixel (x, y) has intensity r



$$\text{mean } m = \sum_r r p(r)$$

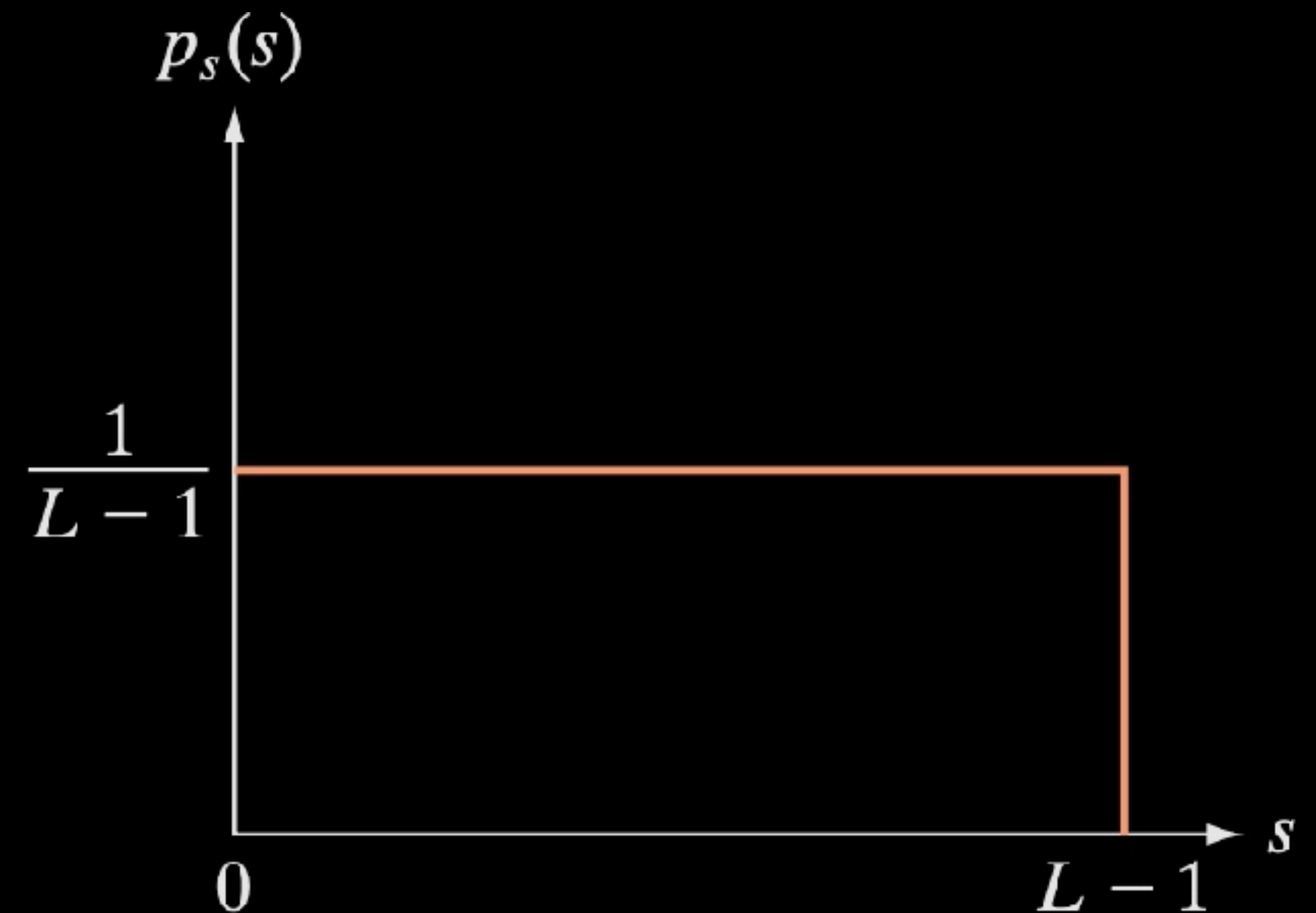
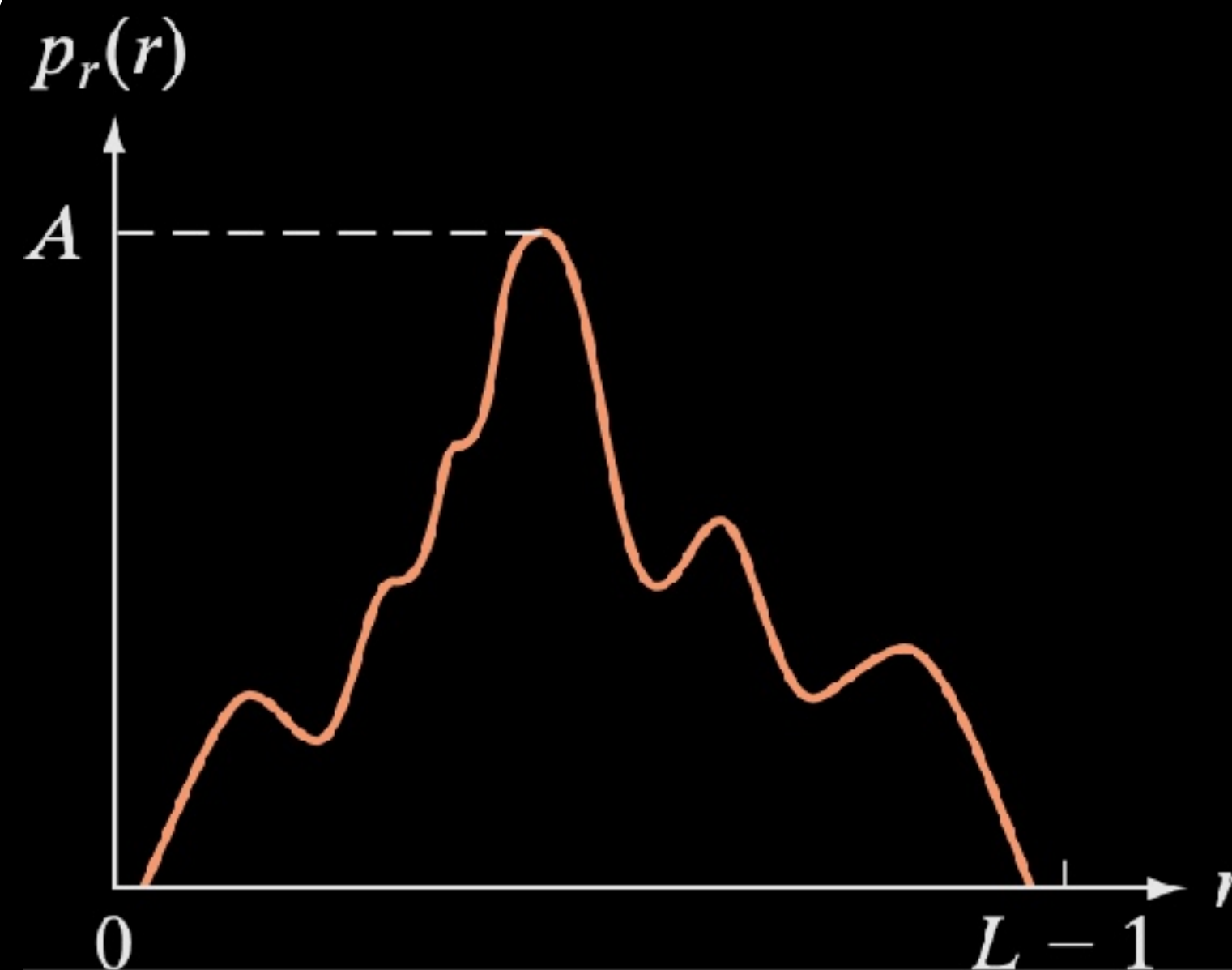
$$\text{variance } \sigma^2 = \sum_r (r - m)^2 p(r)$$



Histogram Equalization

Enhance contrast, for example

- **Goal:** “Spread out” distribution of intensities equally over entire range $[0, L)$
- Find intensity transformation $s = T(r)$ so that histogram becomes uniform:
 - $p_s(s) = 1/(L - 1)$



For now, ignore quantization and suppose the histogram is a continuous distribution $p_r : [0, L-1]_{\mathbb{R}} \rightarrow \mathbb{R}$, and

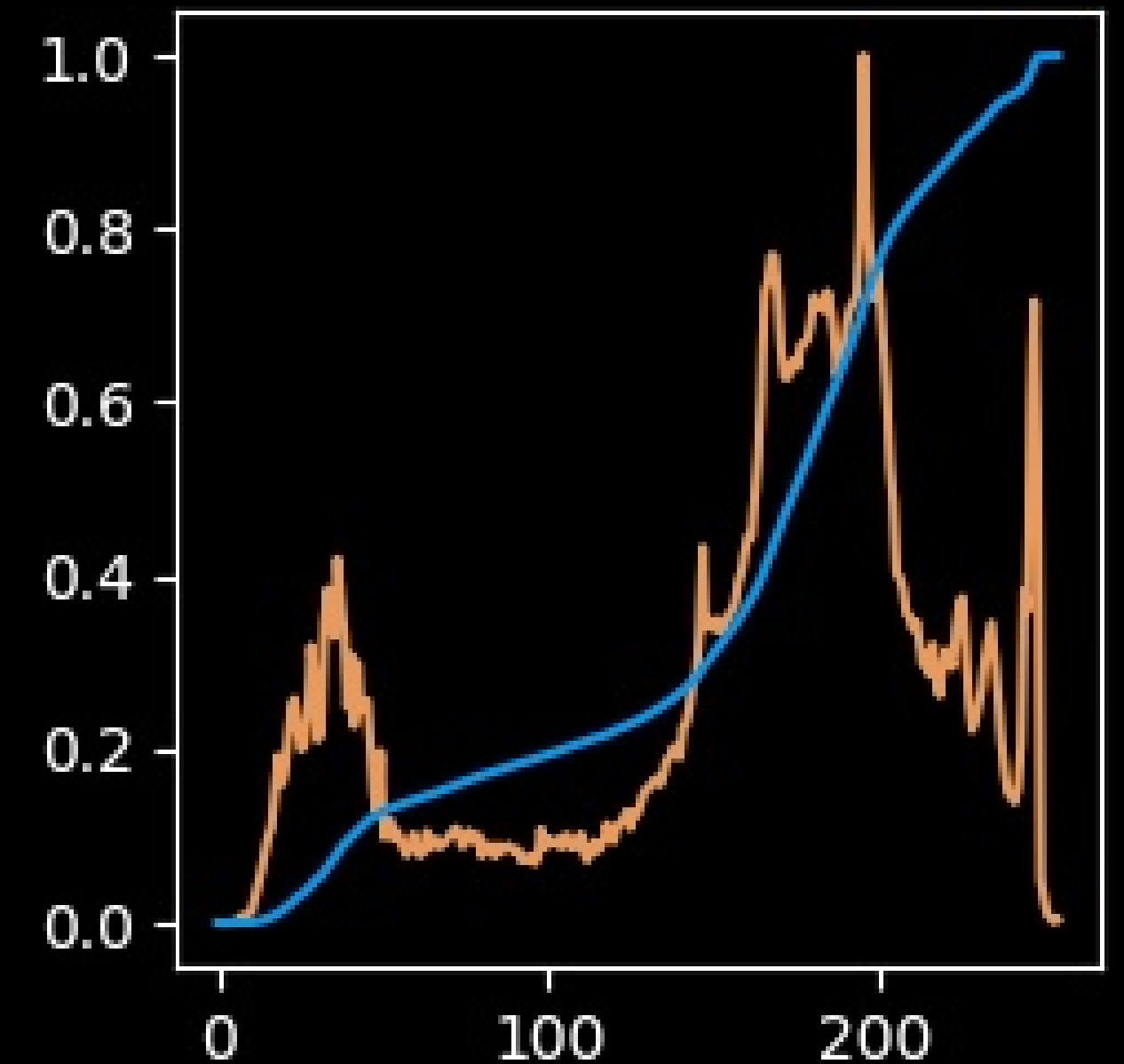
Consider the cumulative distribution function (CDF):

Original color $c_r(r) = P[f(x, y) \leq r] = \int_0^r p_r(r') dr'$

If the intensities are transformed by a *strictly increasing* (unknown) function $s = T(r)$. What's the CDF of the new intensities s ?

$$\begin{aligned} c_s(s) &= P[g(x, y) \leq s] = P[T(f(x, y)) \leq T(r)] \\ &= P[f(x, y) \leq r] = c_r(r) \end{aligned}$$

Color after
equalization



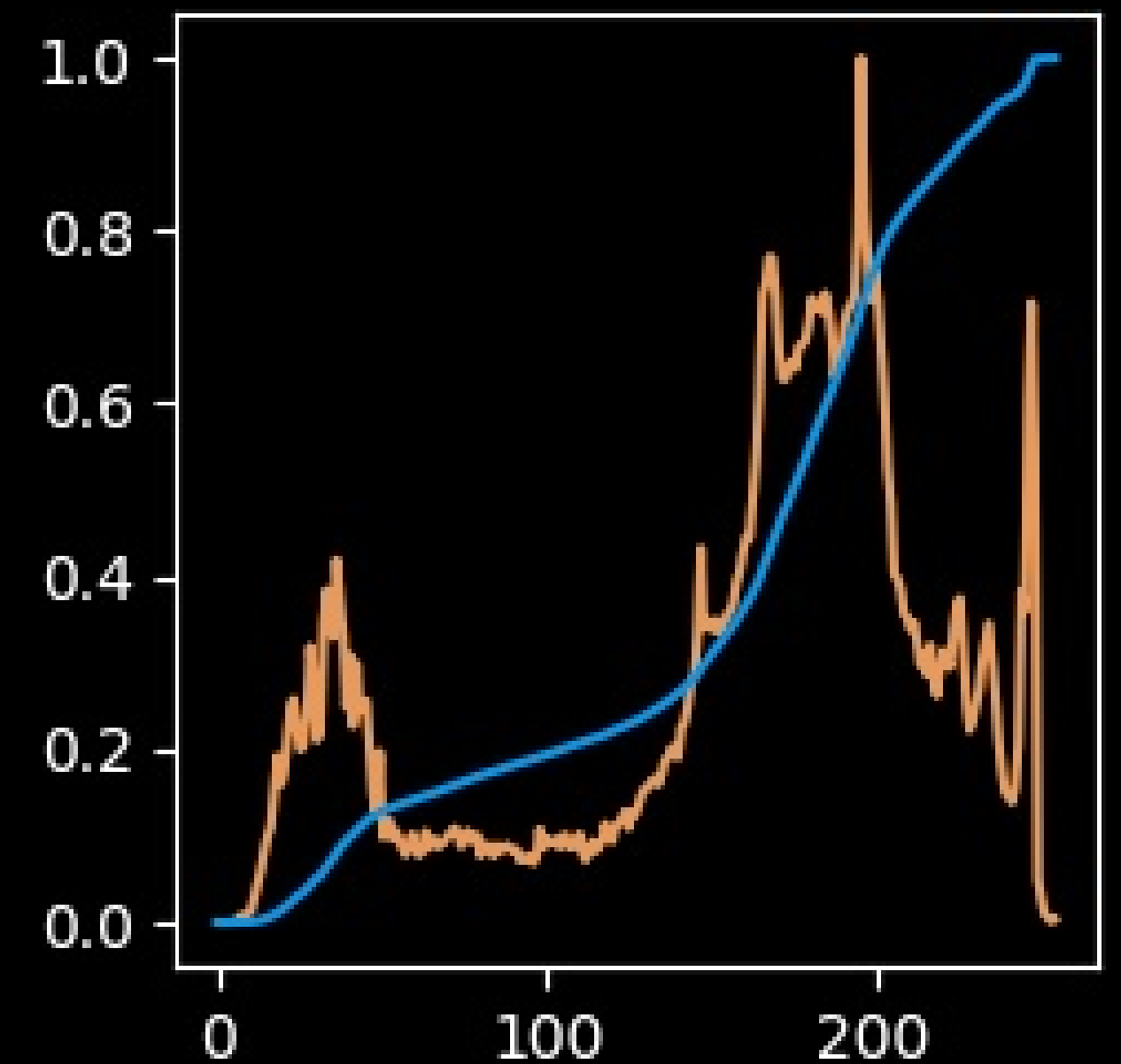
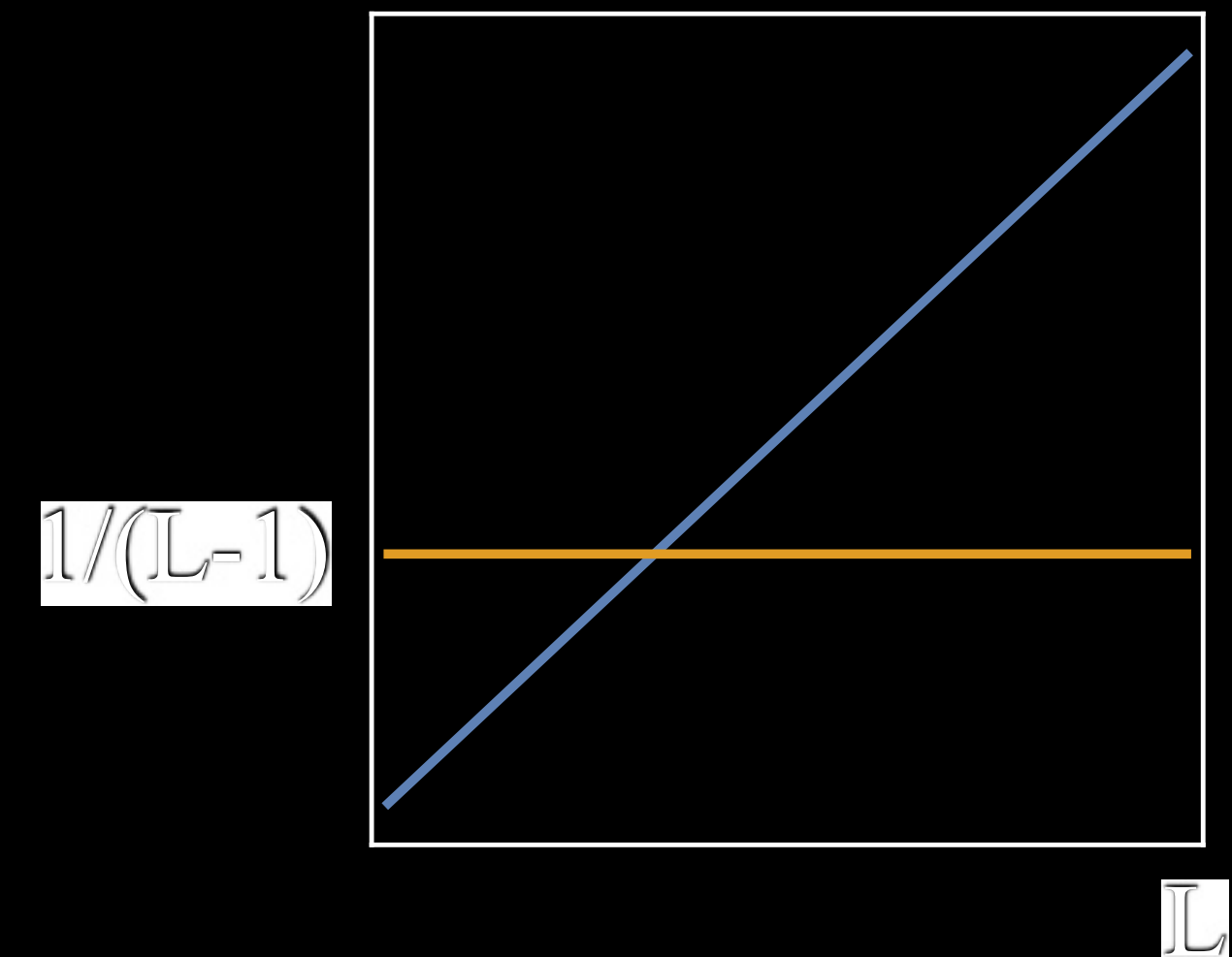
Thus, If $s = T(r)$ is a strictly increasing, $c_s(s) = c_r(r)$

For equalization, we want $p_s(s) = \frac{1}{L-1}$

$$\Rightarrow c_s(s) = \frac{s}{L-1}$$

$$\Rightarrow s = (L-1)c_r(r) = (L-1) \int_0^r p_r(r') dr'$$

Returning to the discrete case, $s = (L-1) \sum_{r'=0}^r p_r(r')$



r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

$$c(0) = 0.19$$

$$T(0) = 7 \quad c(0) = 1.33 \rightarrow 1$$

$$c(1) = c(0) + 0.25 = 0.44$$

$$T(1) = 7 \quad c(1) = 3.08 \rightarrow 3$$

$$\vdots$$

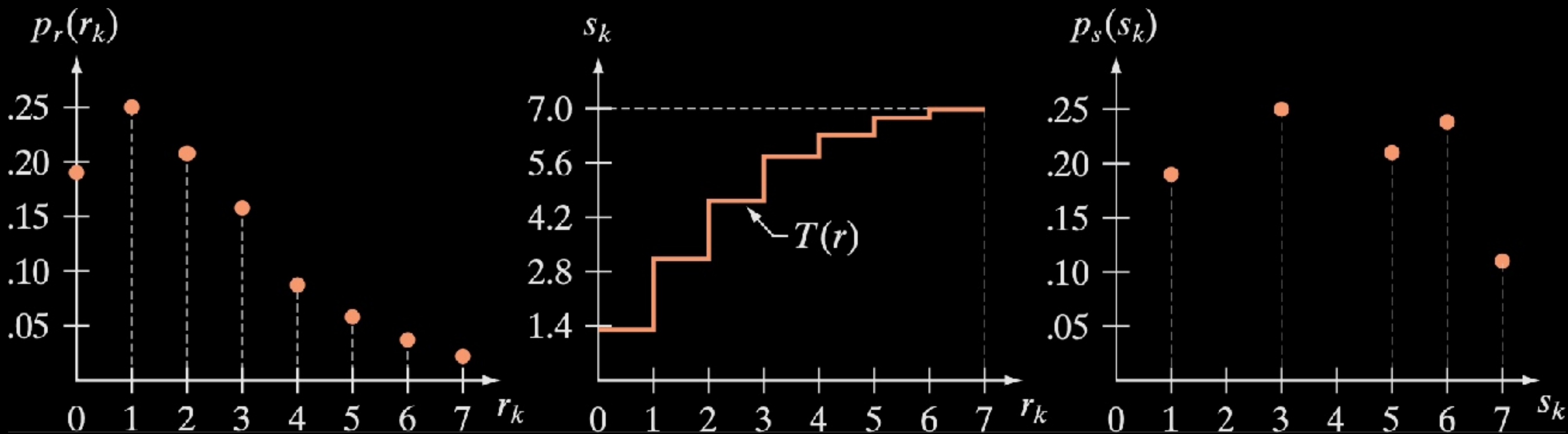
$$\vdots$$

$$c(6) = c(5) + 0.03 = 0.98$$

$$T(6) = 7 \quad c(6) = 6.65 \rightarrow 7$$

$$c(7) = c(6) + 0.02 = 1$$

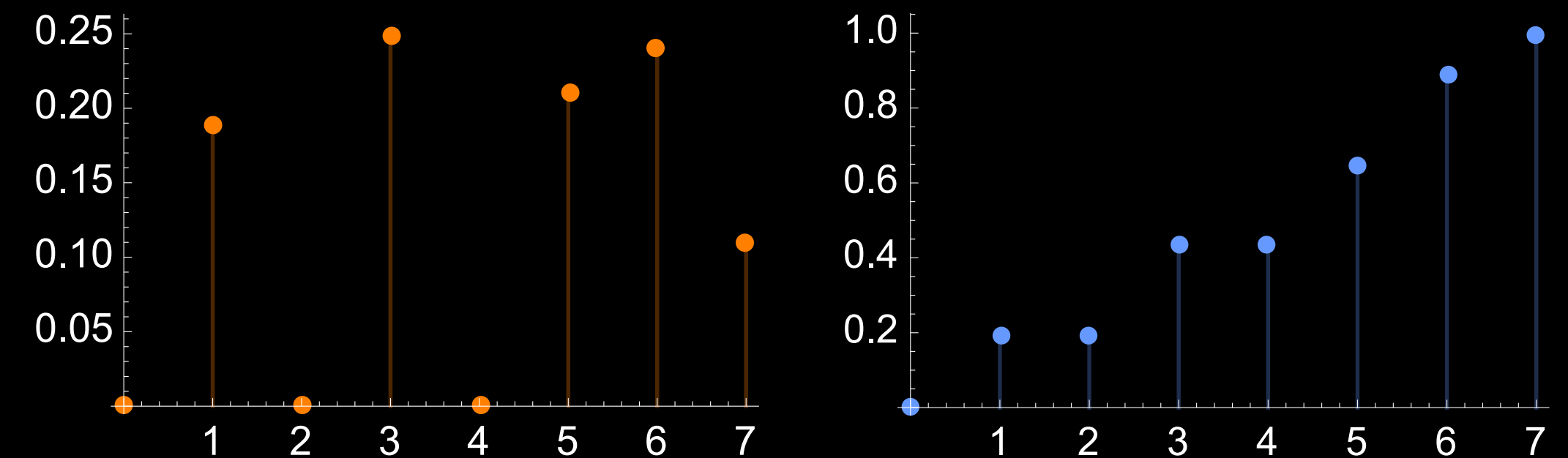
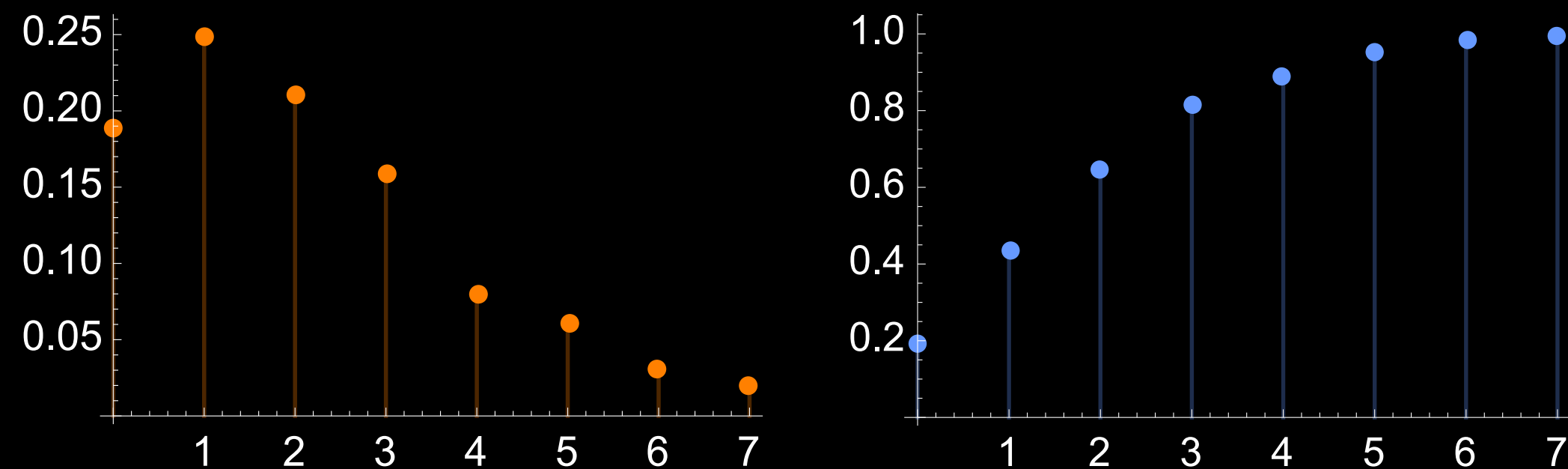
$$T(7) = 7 \quad c(7) = 7$$



Perfectly uniform histogram not possible for digital images

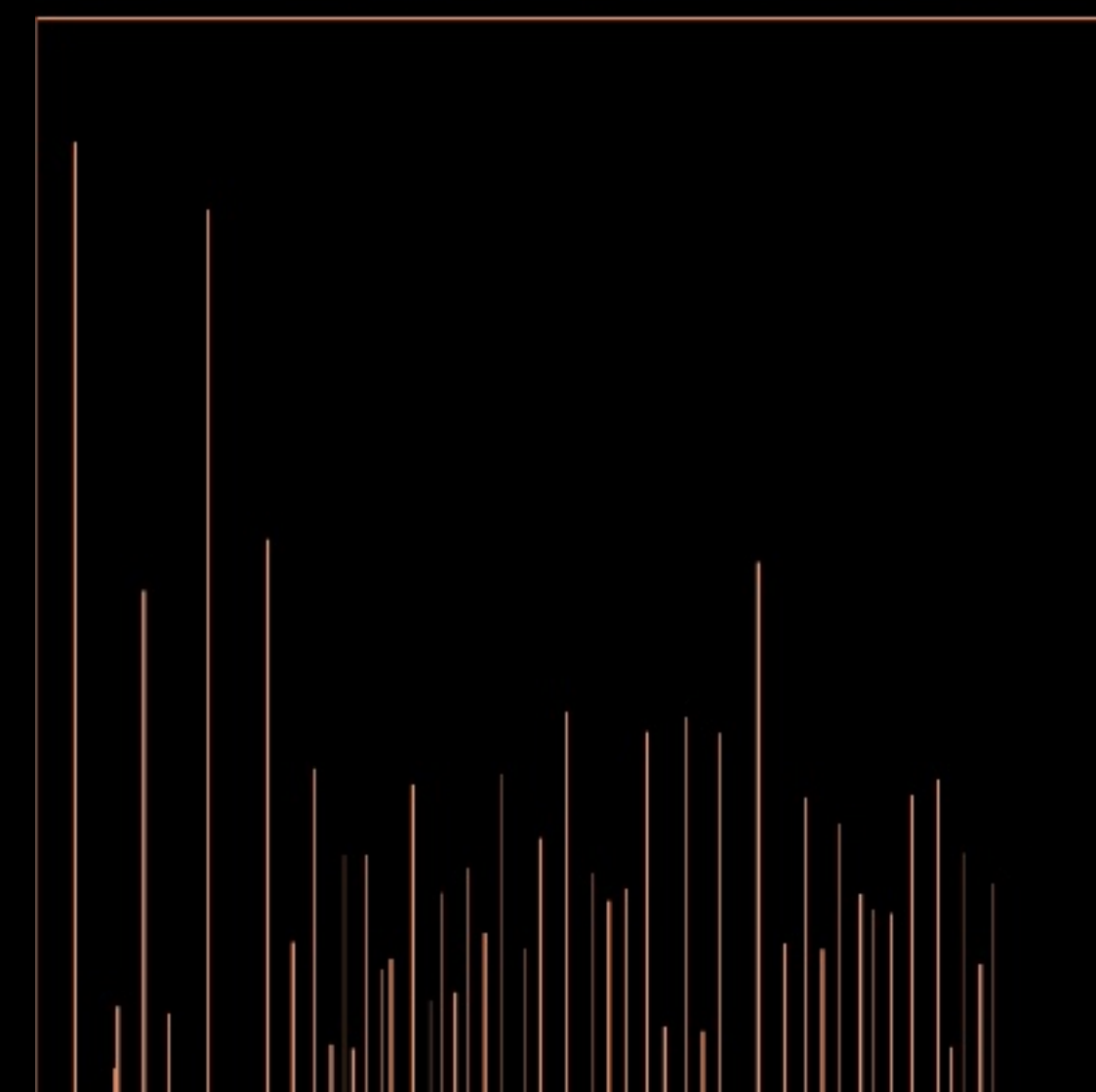
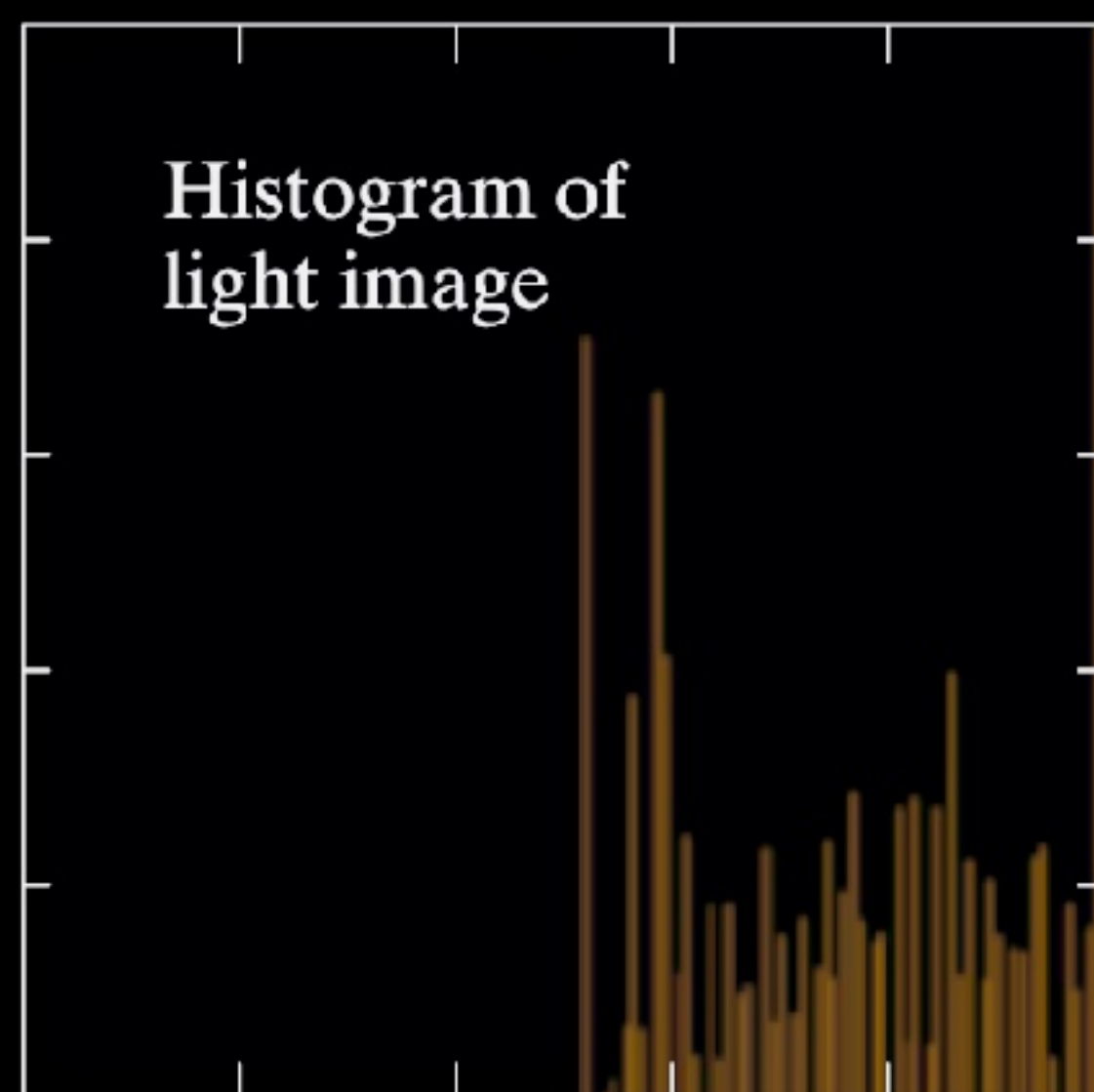
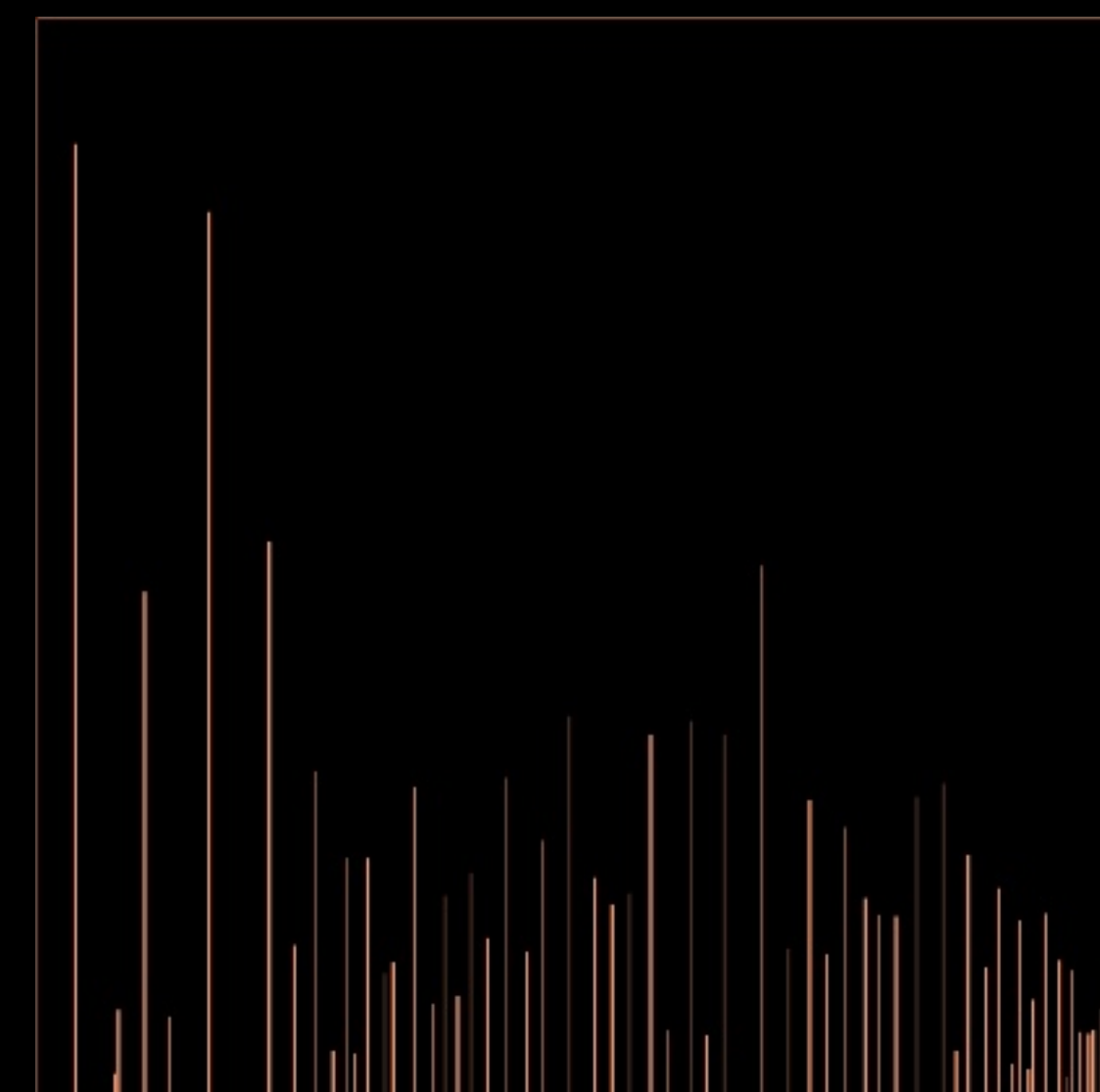
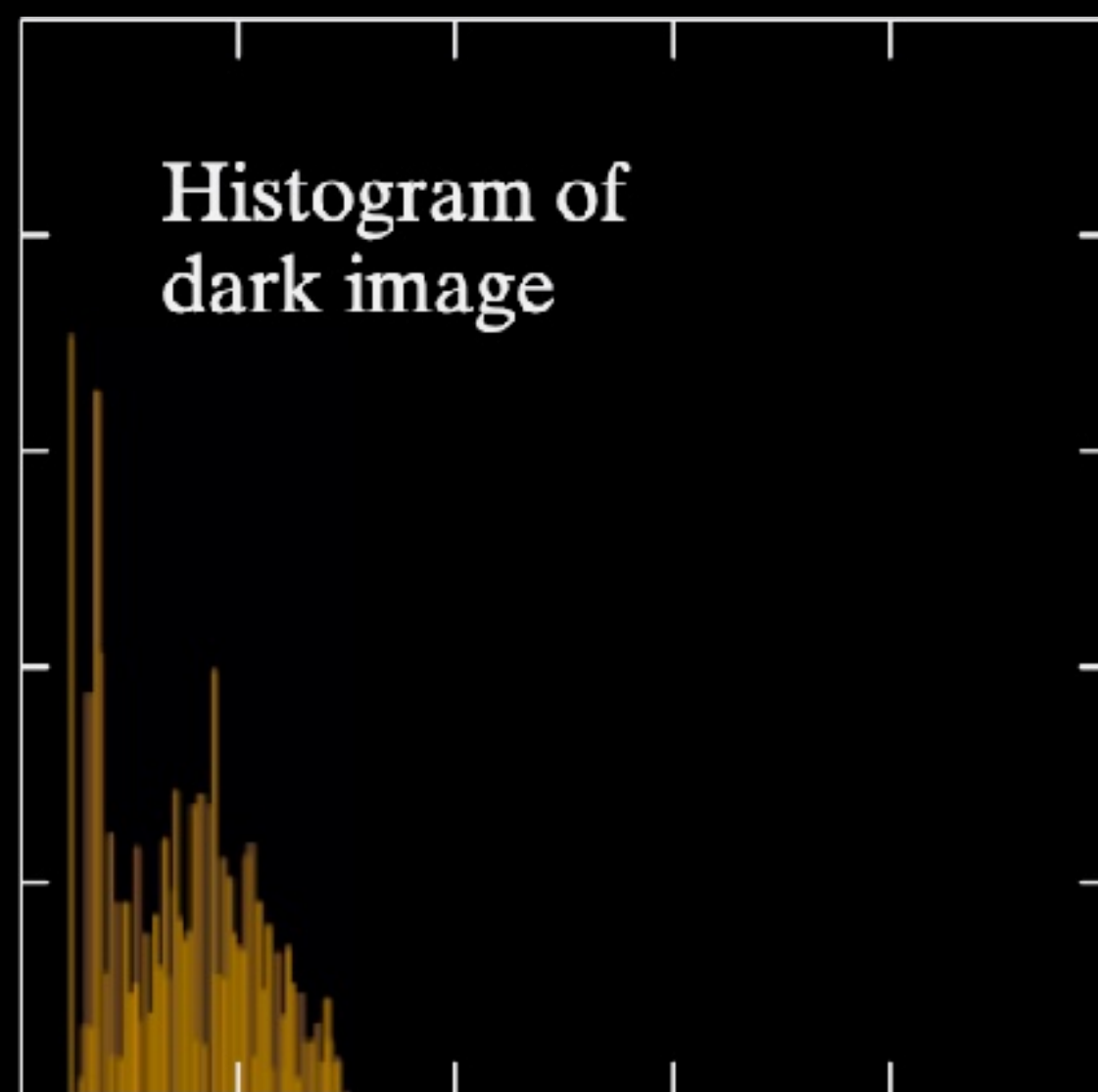
- Finite number of intensity levels $\{0, \dots, L - 1\} \Rightarrow T$ cannot be *strictly* increasing

Histogram equalization does its best:

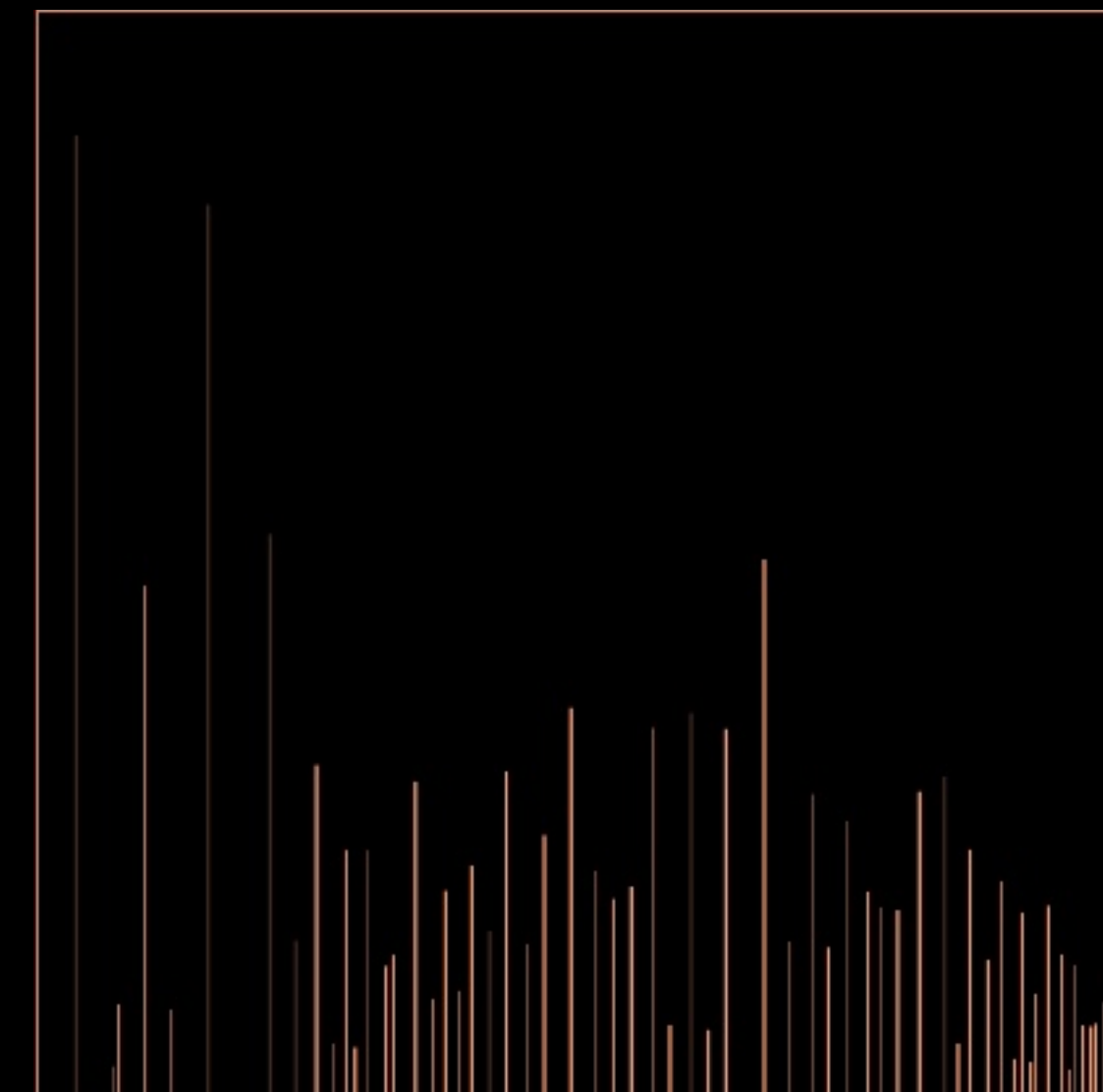
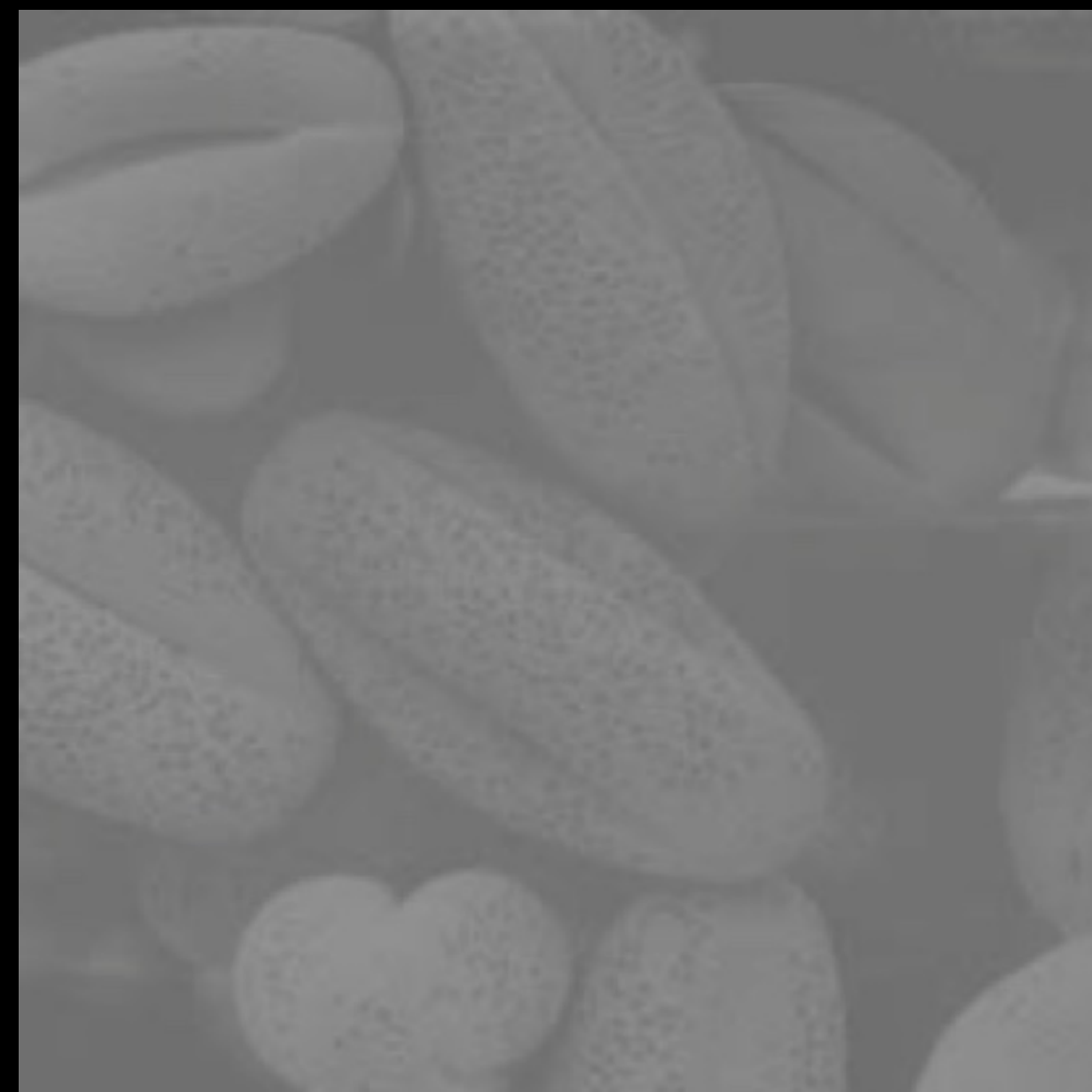
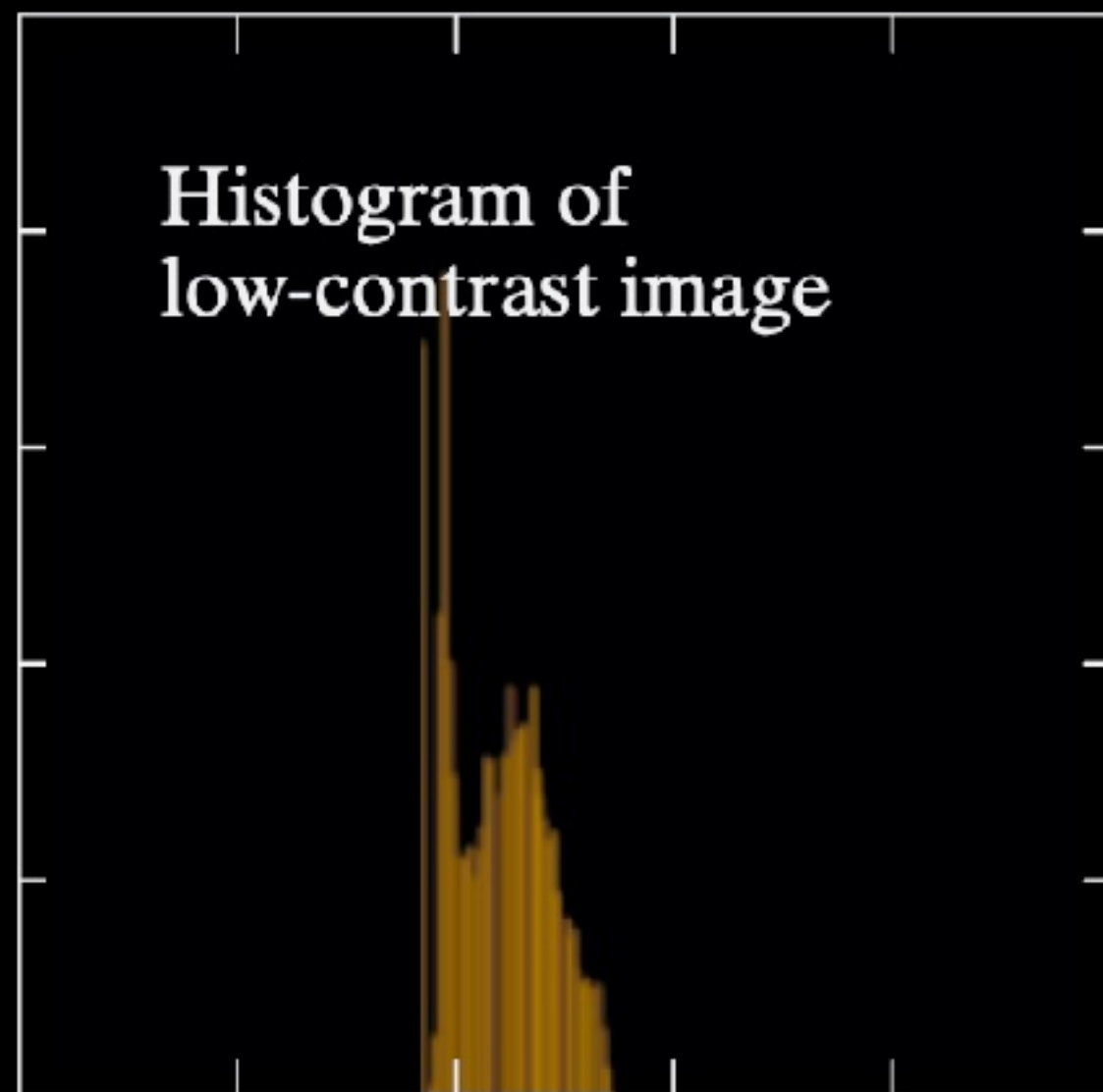


Another interpretation:

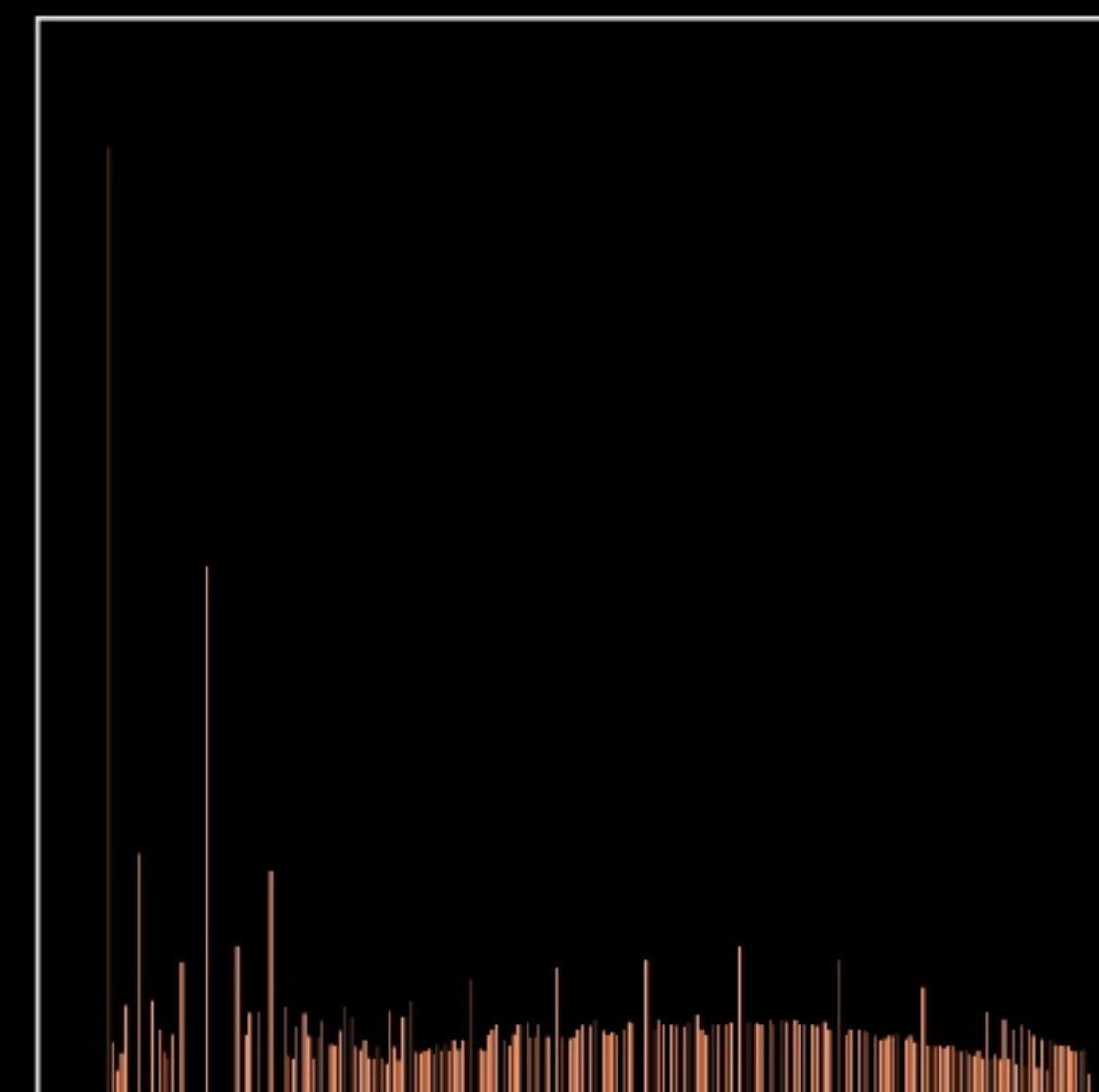
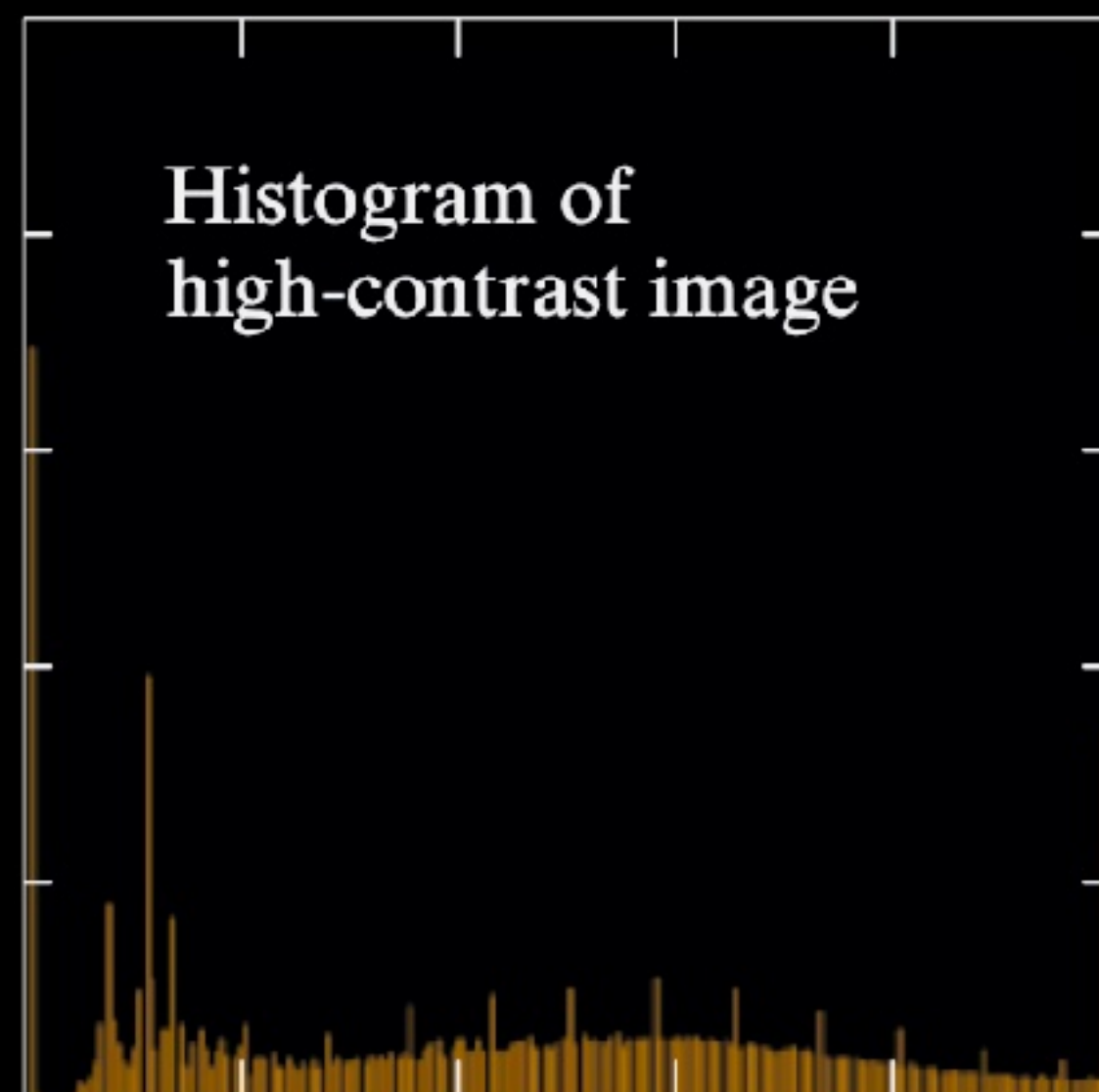
- min intensity $\rightarrow 0$, max intensity $\rightarrow L - 1$
- k th percentile $\rightarrow k\%$ intensity



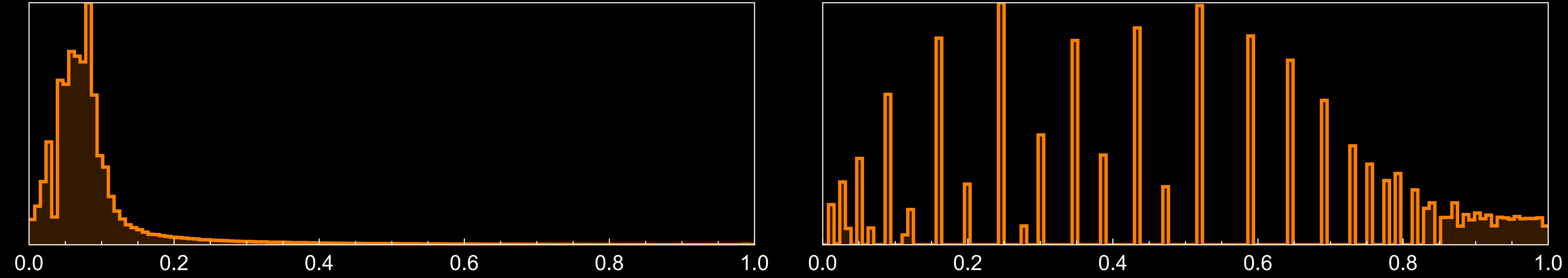
Histogram of
low-contrast image



Histogram of
high-contrast image

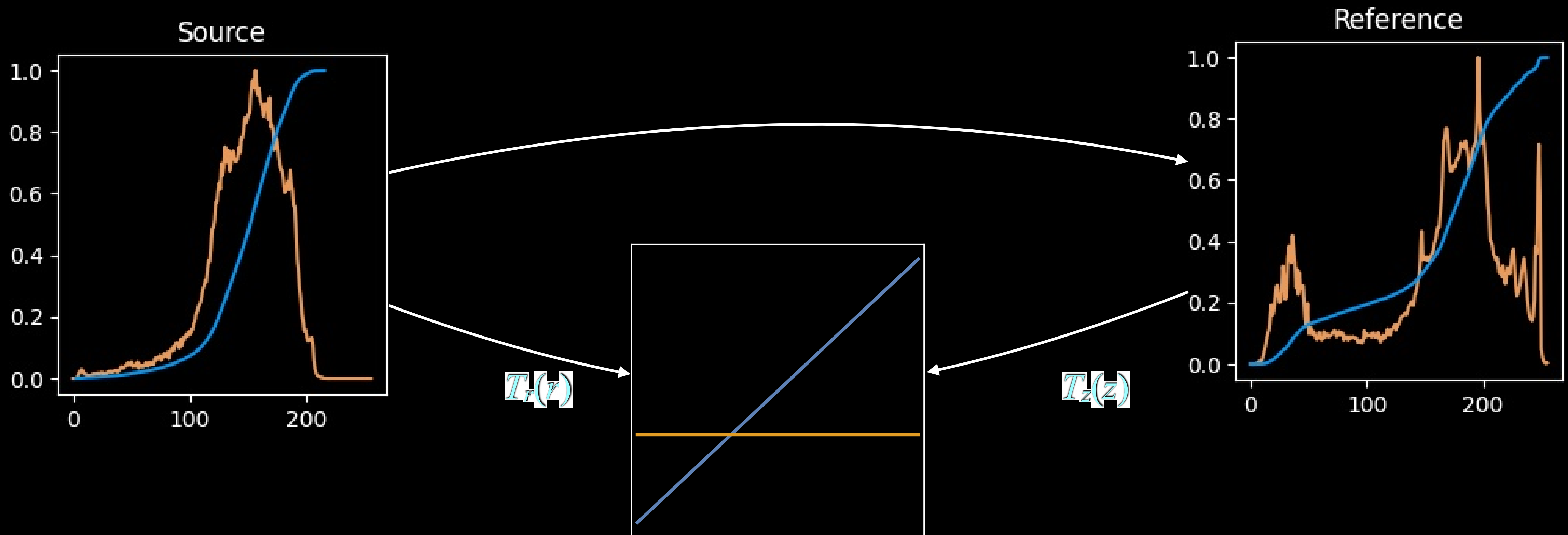


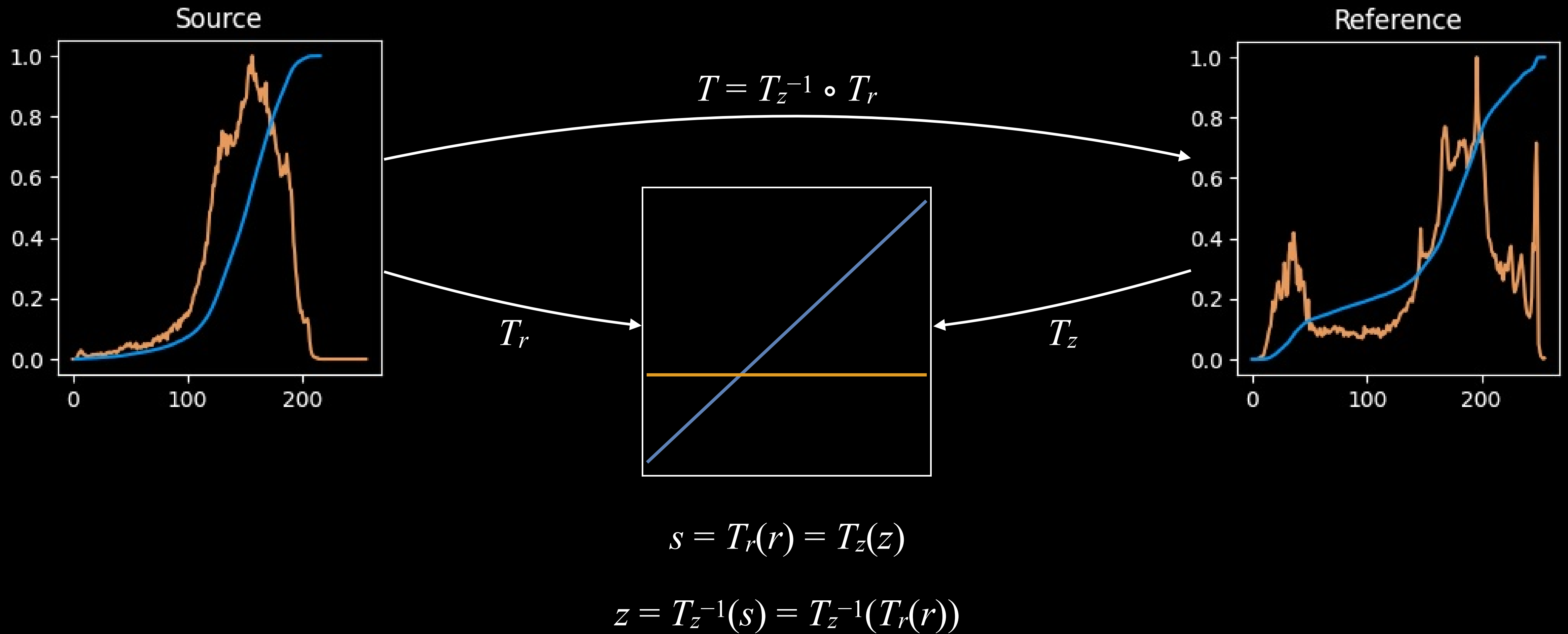
Histogram equalization is not always a good idea!



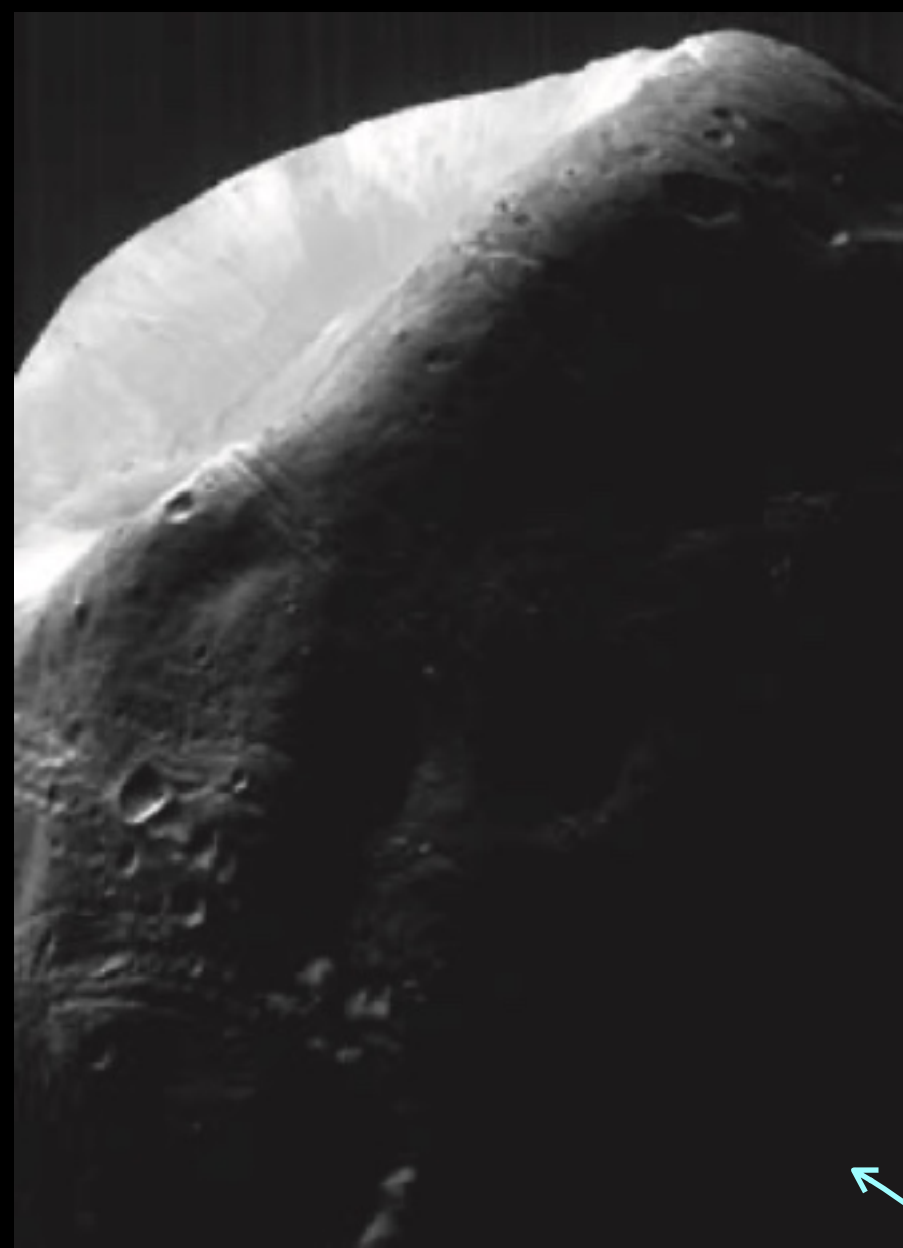
Histogram matching

- Find $s = T(r)$ such that s has a desired histogram $p_z(z)$, not necessarily uniform





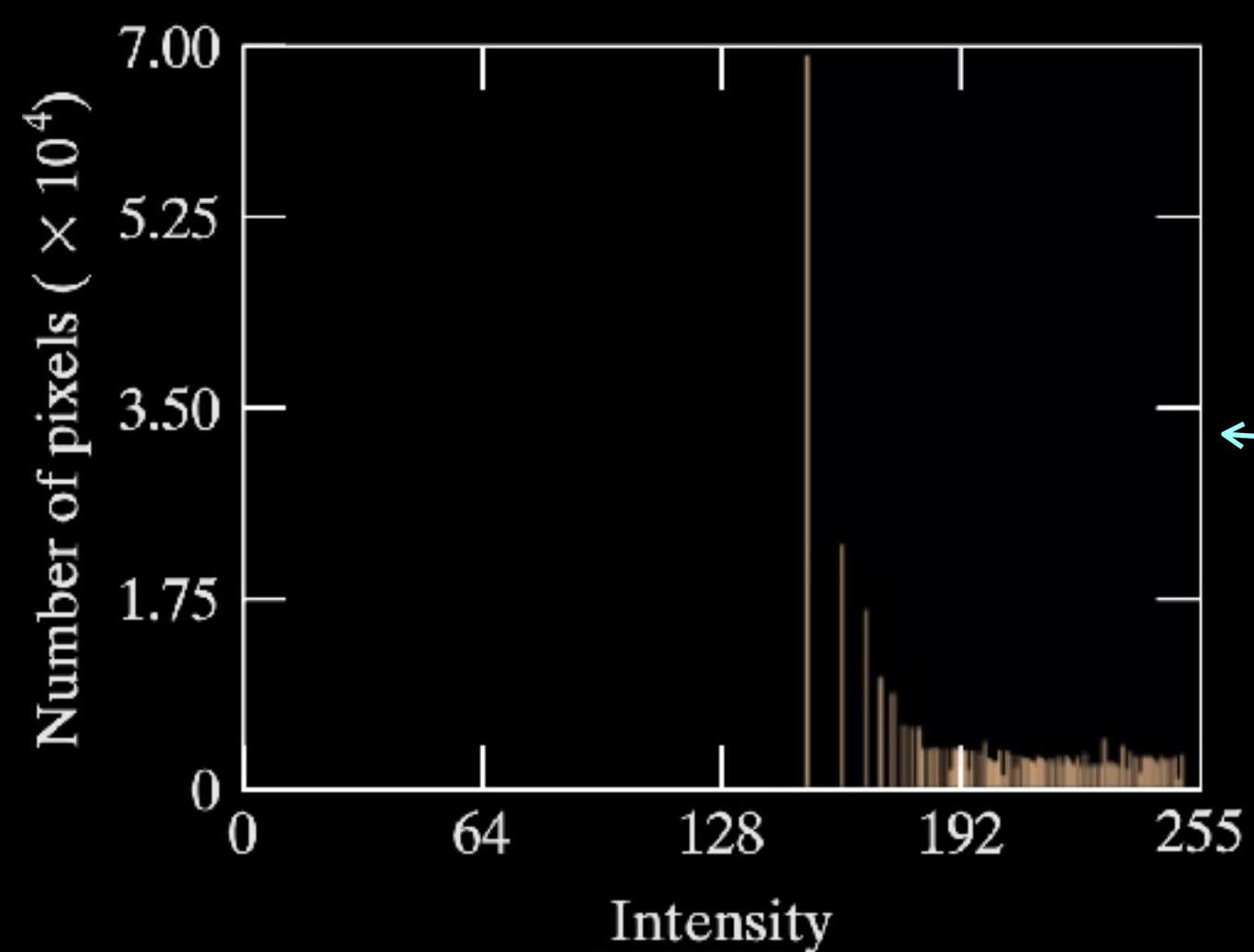
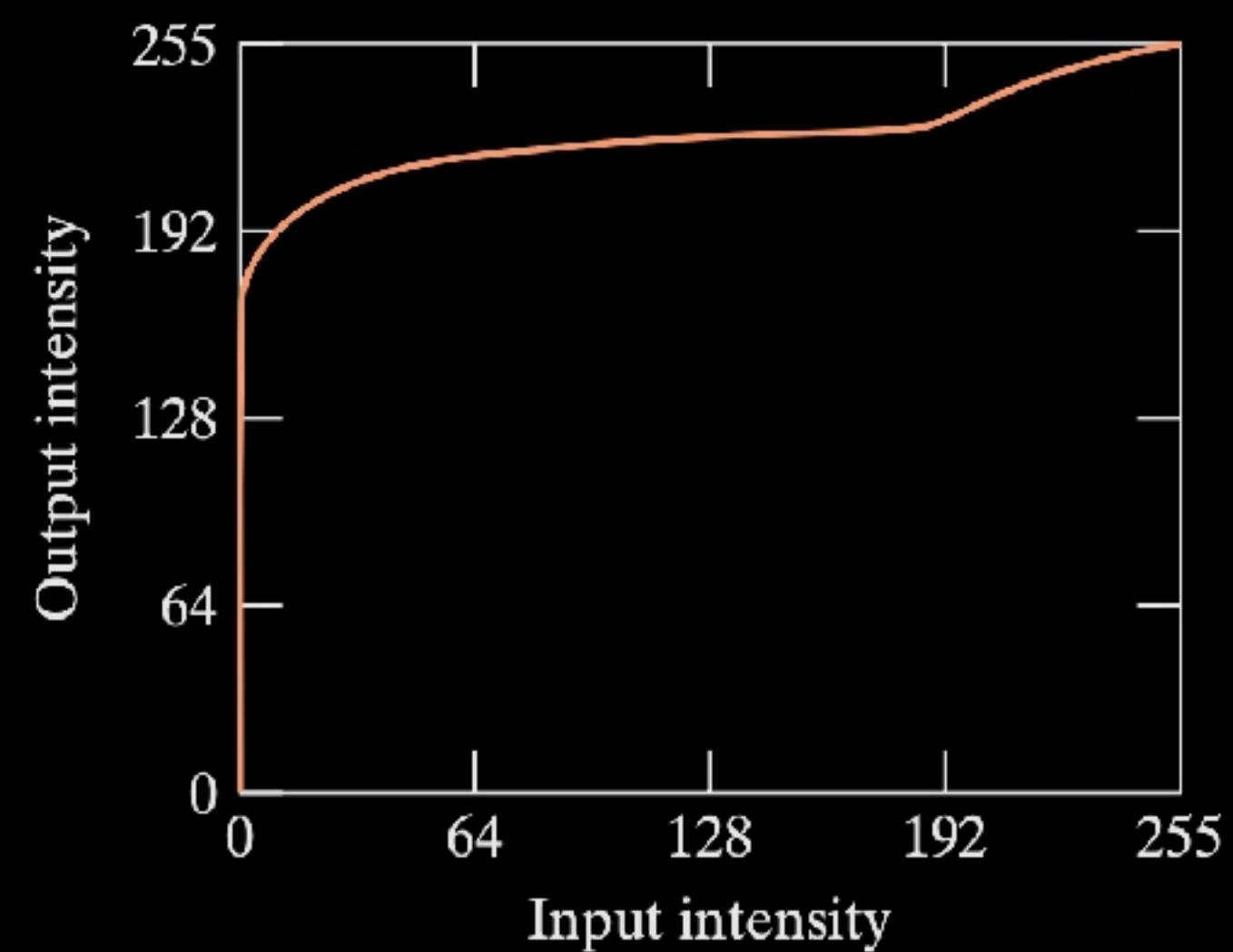
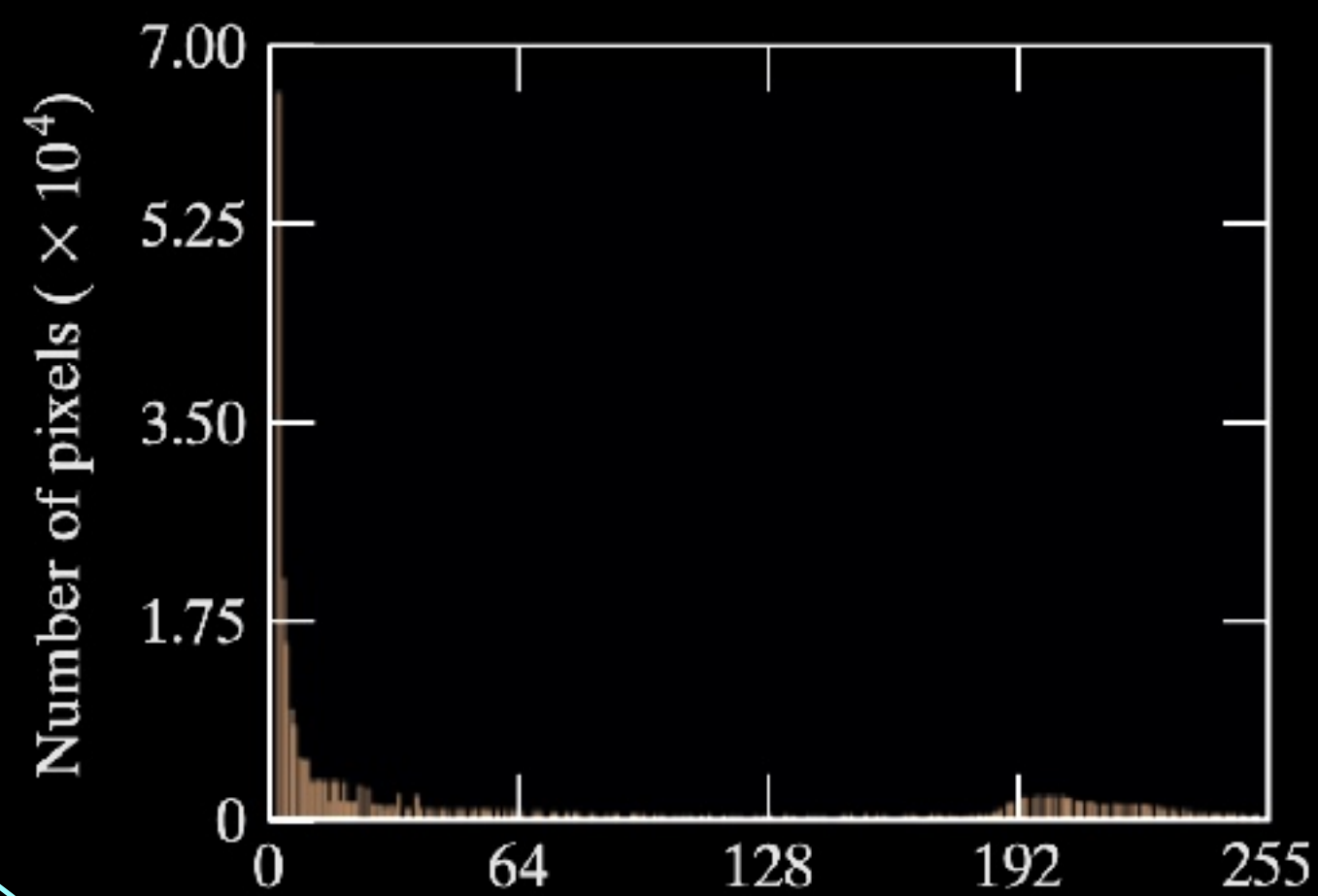
To compute $T_z^{-1}(s)$: store T_z in lookup table, do binary search for closest match to s



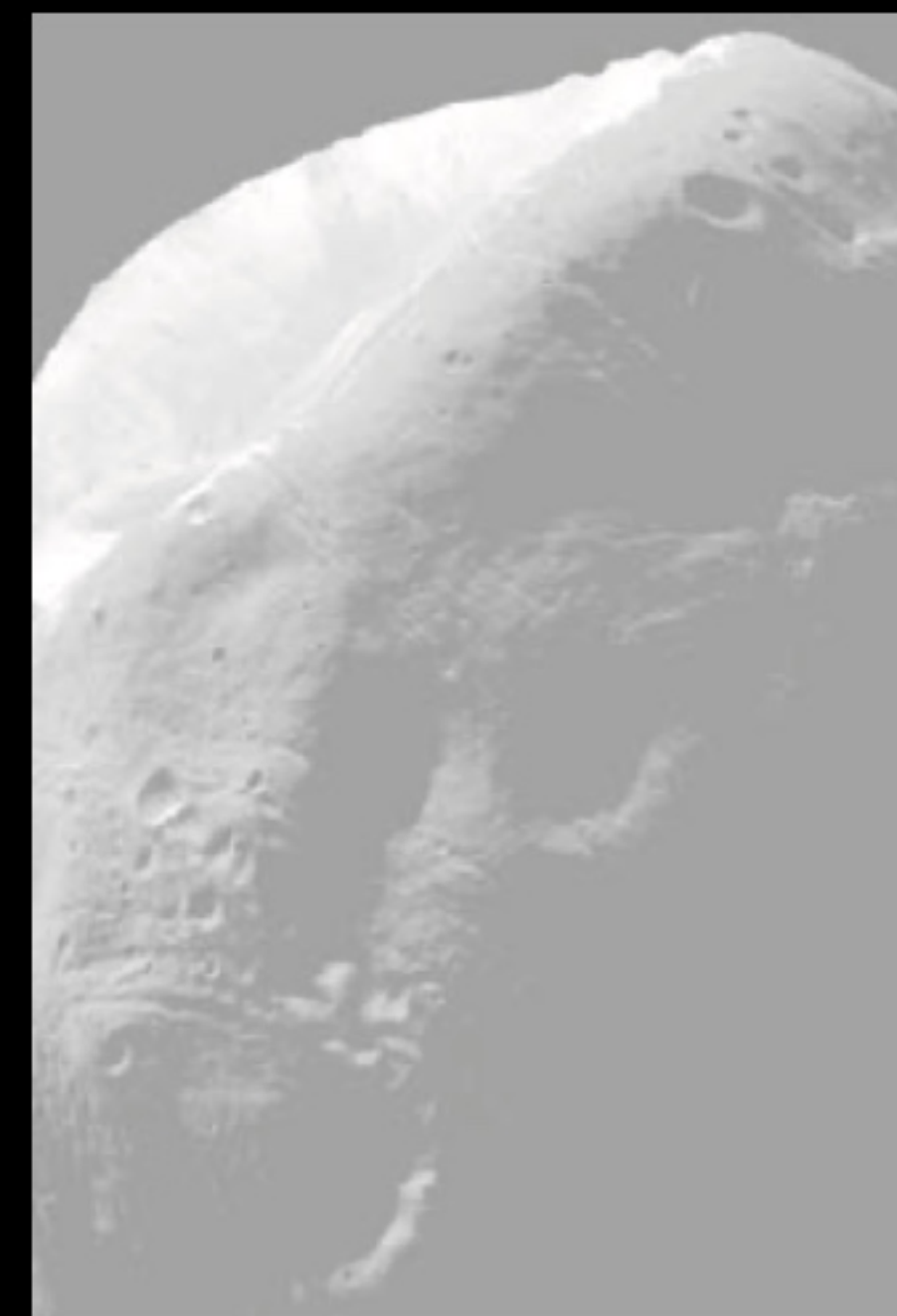
Input Image
(Mars Moon [NASA])

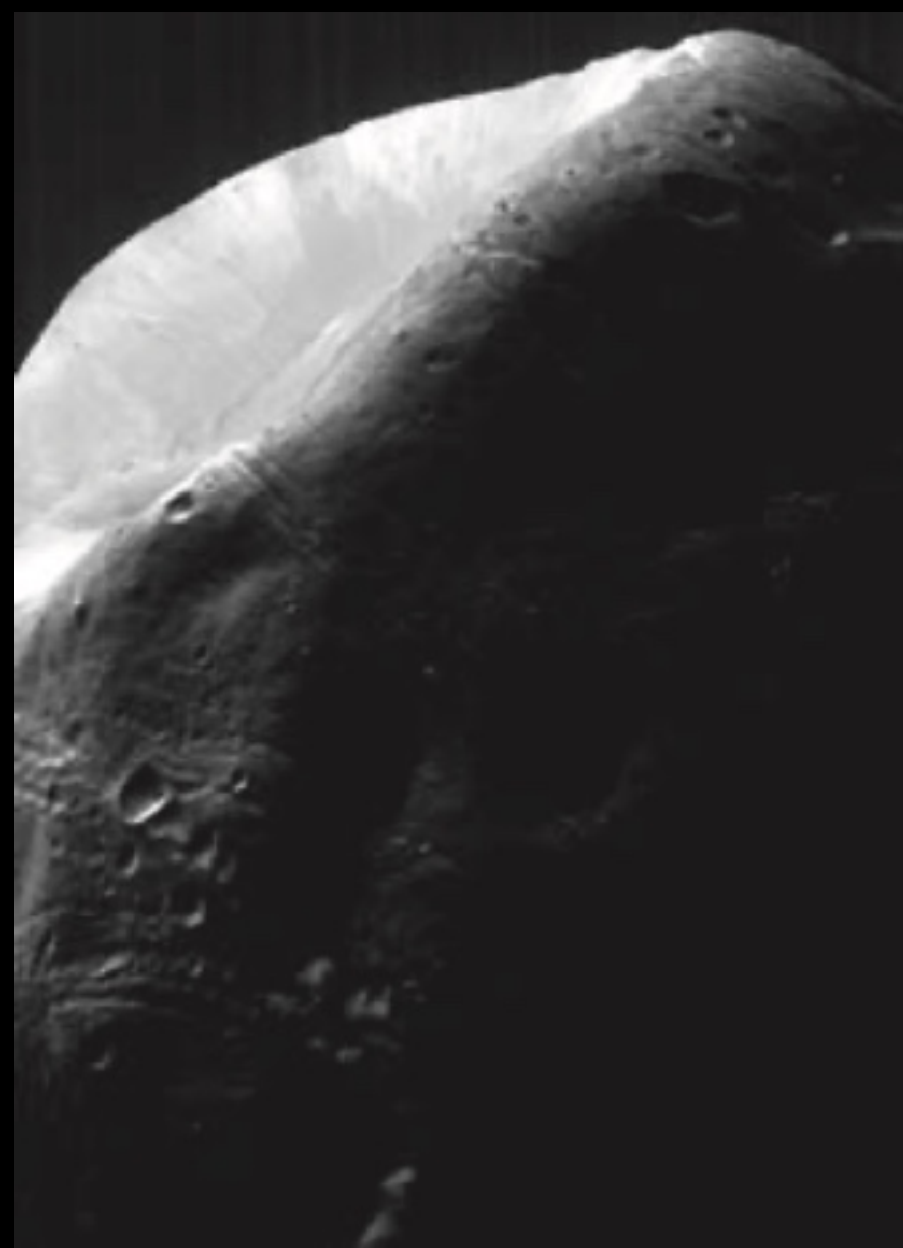
Many dark pixels

Histogram

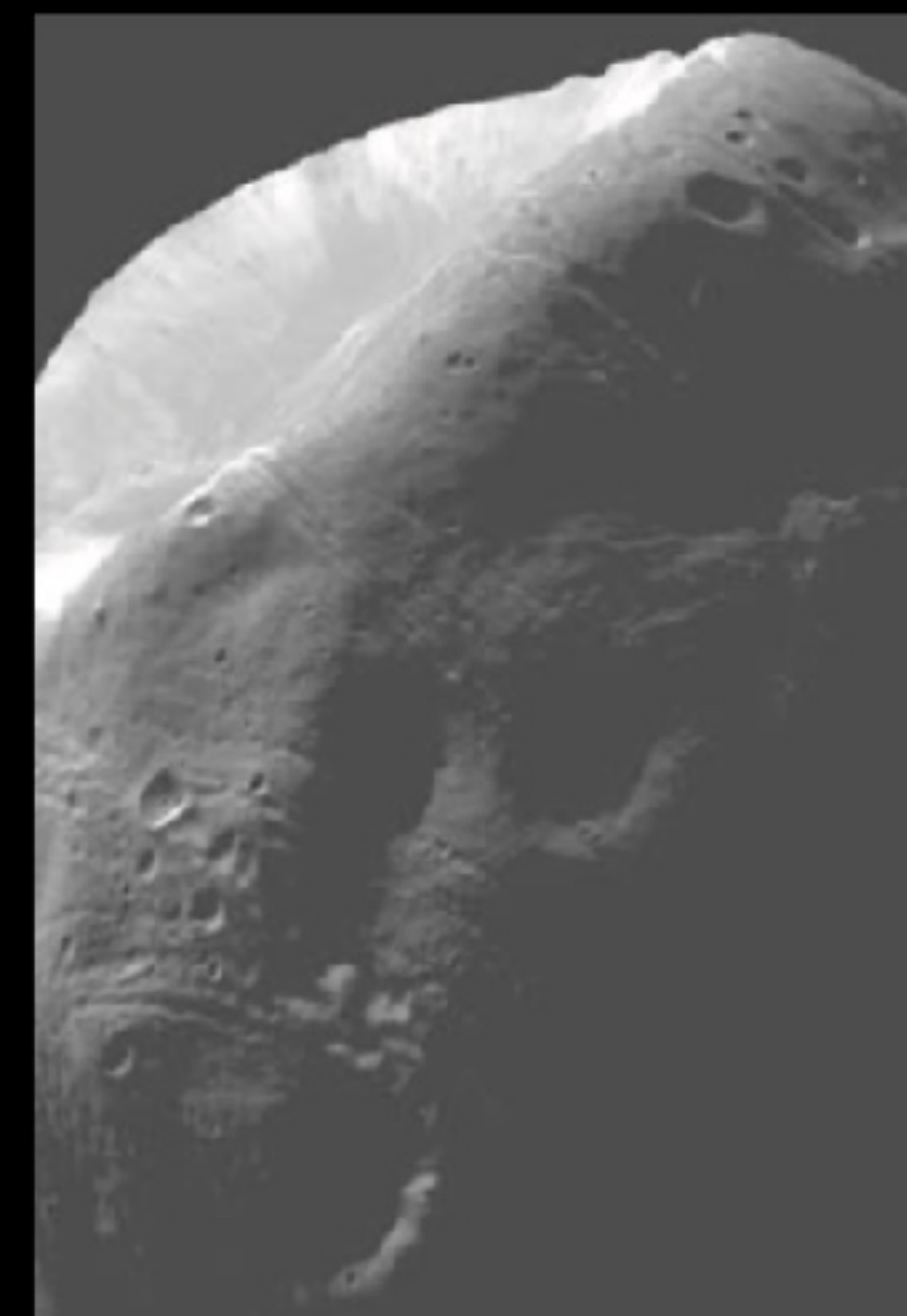
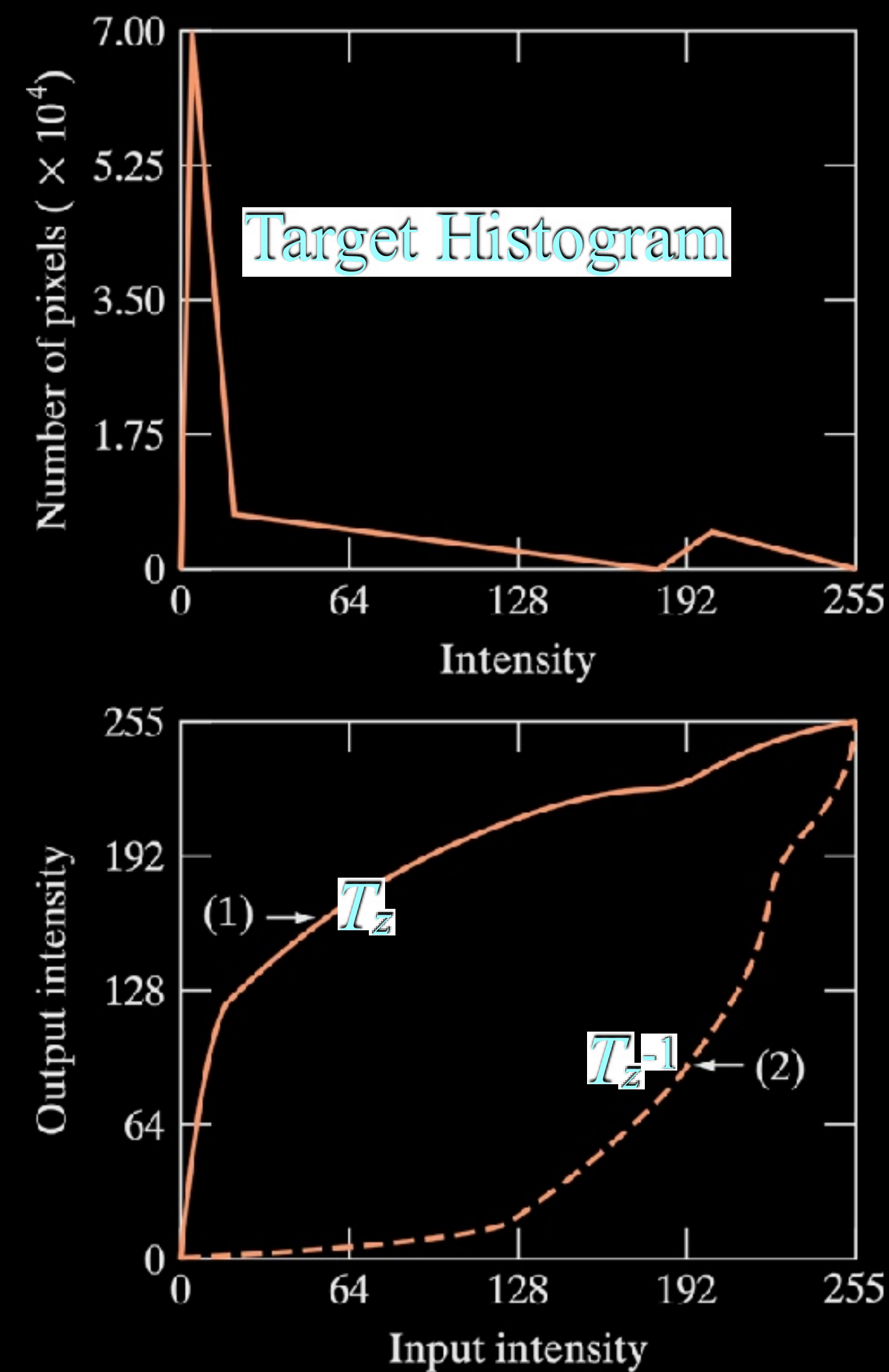
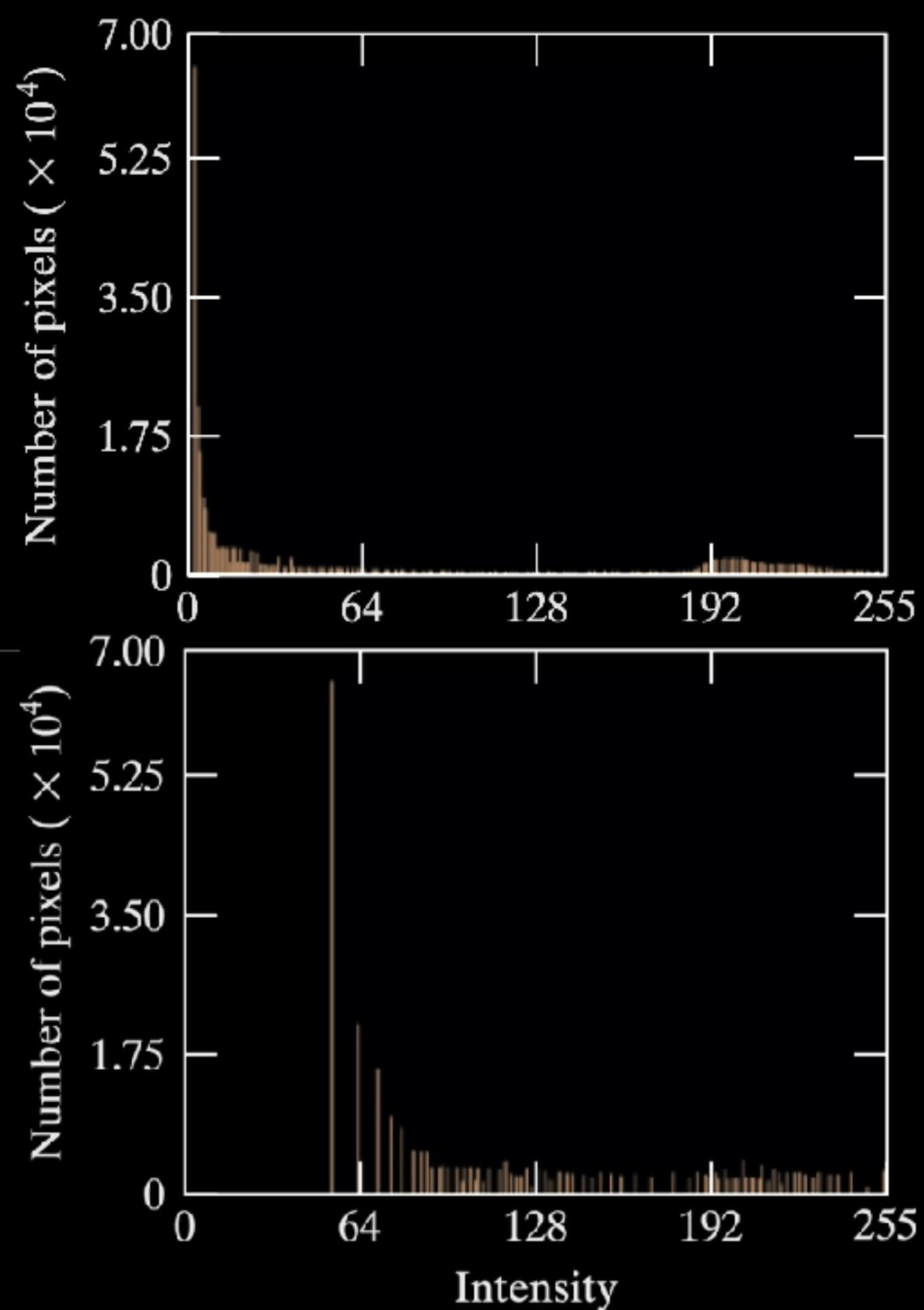


Histogram
Equalized





Input Image
(Mars Moon [NASA])



Targeted Histogram

