Recognizing faces

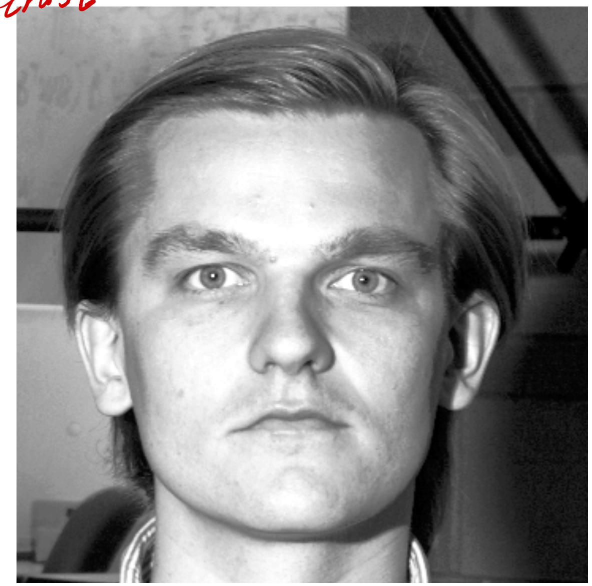
Histogram Equalization

Image normalization

- Due to lighting or shadow, intensity can vary significantly in an image.

make. image
more contrast

Histogram equalization



Histogram equalized

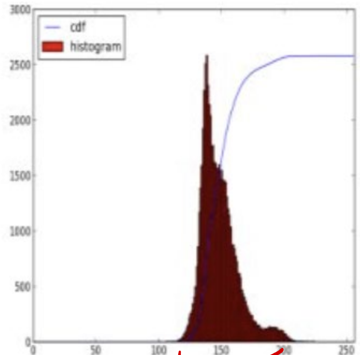
- Normalization of pixel intensity helps correct variations in imaging parameters in cameras as well as changes in illumination conditions.
- One widely used technique is histogram equalization, which is based on image histogram. It helps reduce extreme illumination.

[Li-Jain-05, Gonzalez-Woods-08, Yale]

Histogram Equalization

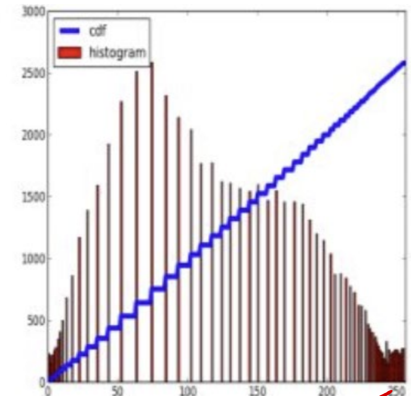
- Examples

low contrast



4 bits

Histogram equalization



8 bits

Histogram equalized

Source: https://docs.opencv.org/4.x/d5/daf/tutorial_py_histogram_equalization.html

Image Histogram

- Image Histogram
  - digital image with gray levels [0, L-1]
    255
  - $p(r_k) = n_k/N$, probability of occurrence of gray level r_k
  - r_k is the k^{th} gray level
  - n_k = number of pixels with k^{th} gray level
  - N = total number of pixels
  - $k = 0, 1, 2, 3, 4, 5 \ldots, L-1$

Image Histogram

- $p(r_k)$ is the probability of the occurrence of gray-level r_k
- $p_r(r_k)$ is the probability density function (PDF) of the variable r_k, $k = 0, \ldots, L-1$

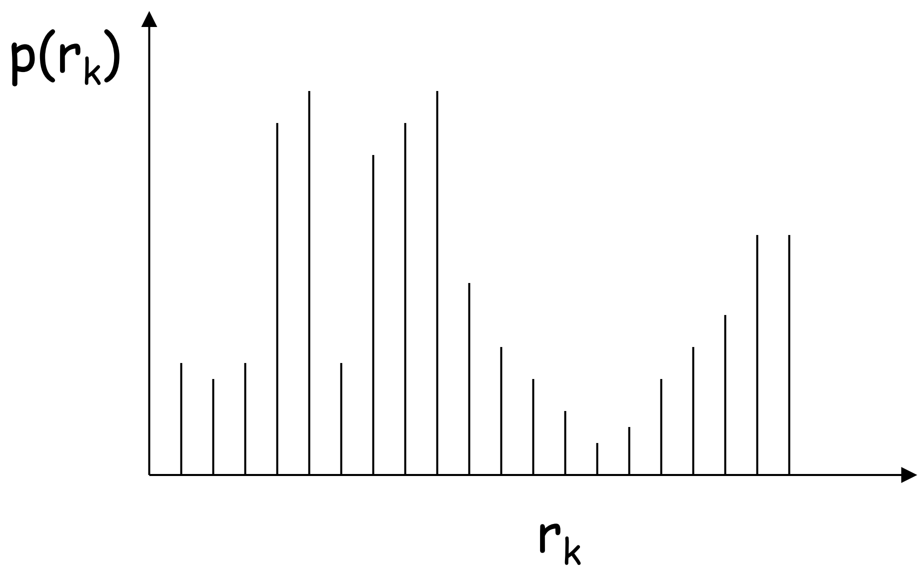


Image Histogram

- Histogram can tell you a lot about an image
- Gray level distribution
- It can be modeled by a statistical distribution (red line)

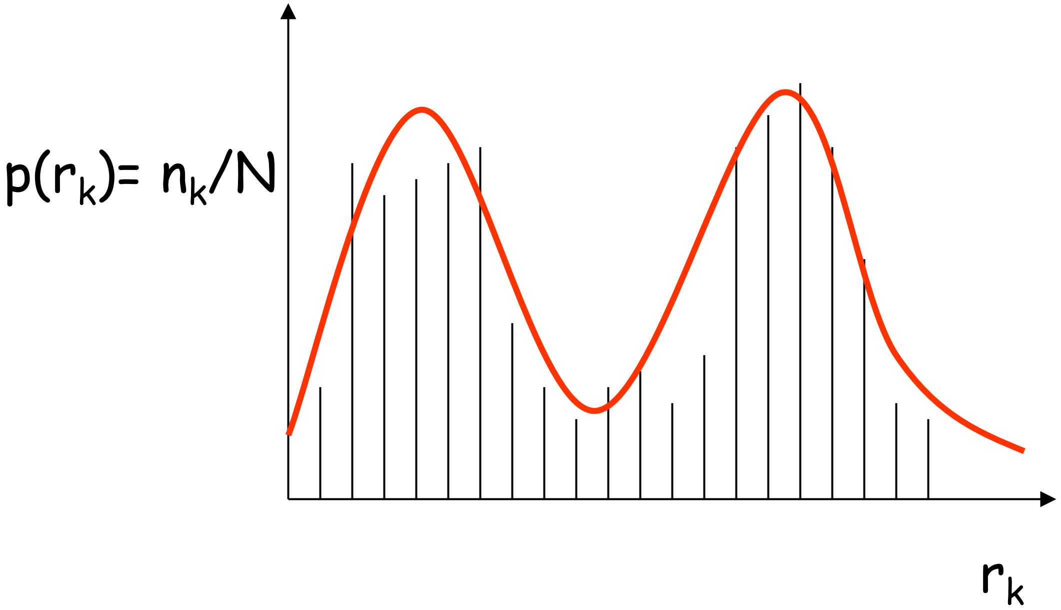


Image Histogram

- Gray level distribution helps define intensity threshold

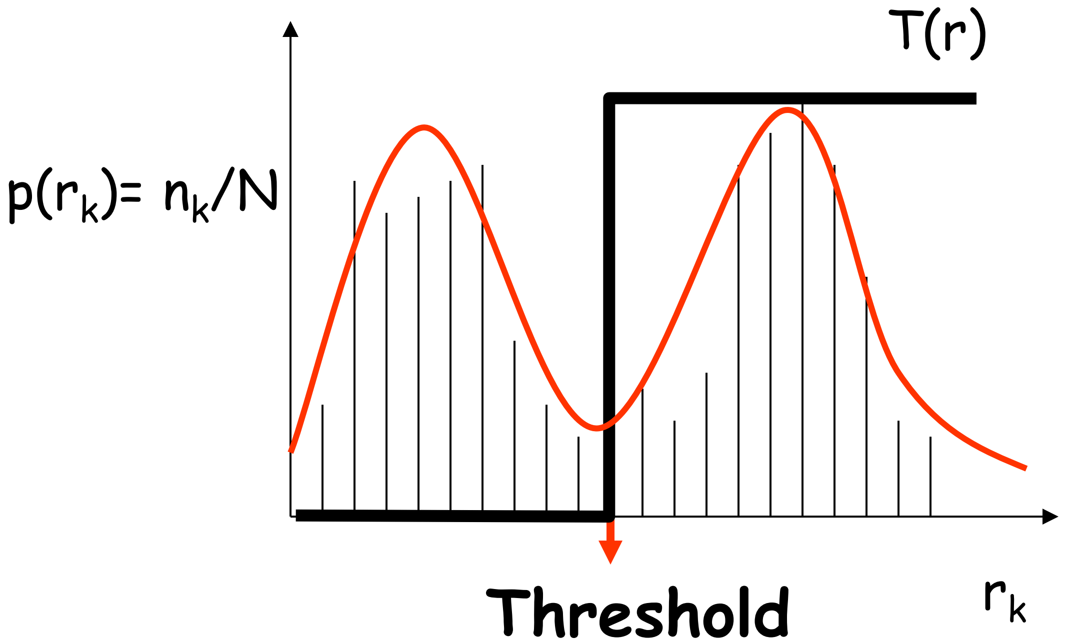


Image Histogram

- Histograms are not unique. Two images below give the same image histogram.
- No spatial information is captured

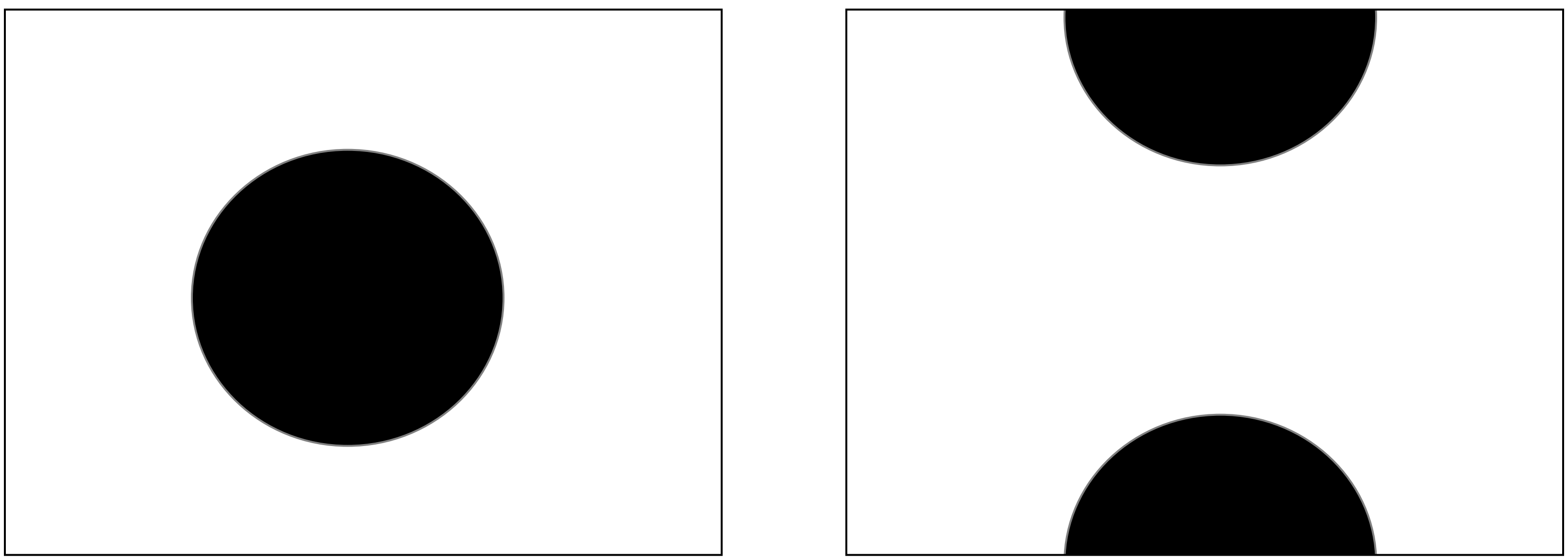
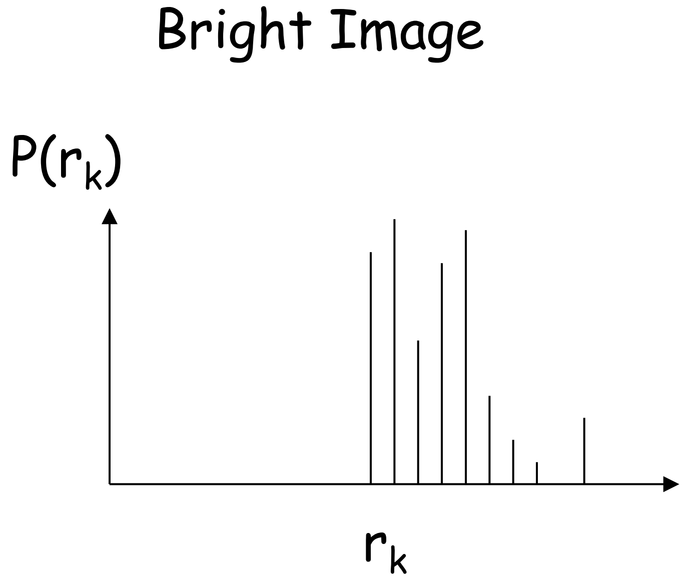


Image Histogram

- Image types



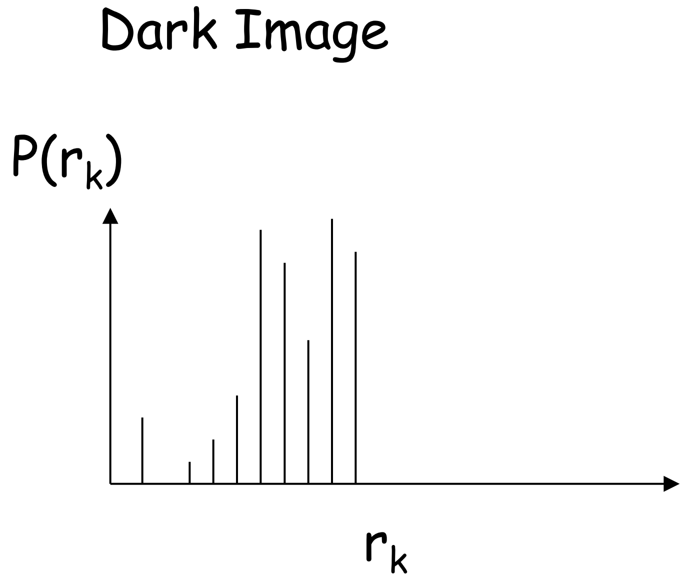
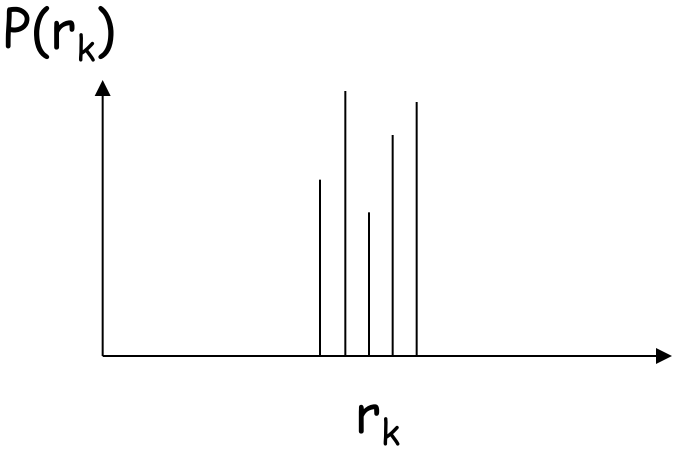


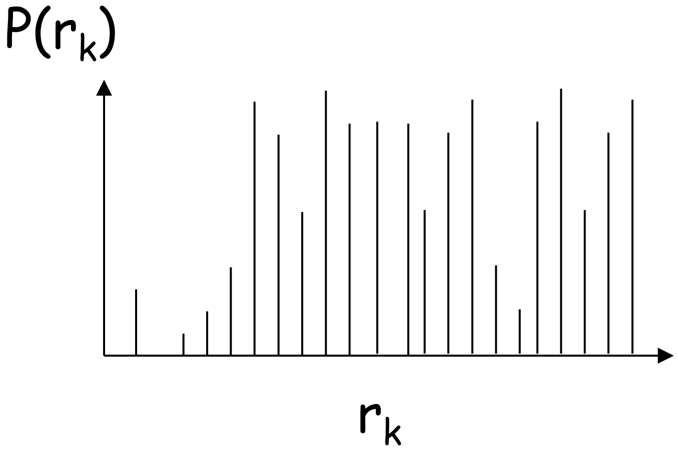
Image Histogram

- Image types

Low Contrast



High Contrast



Some Examples

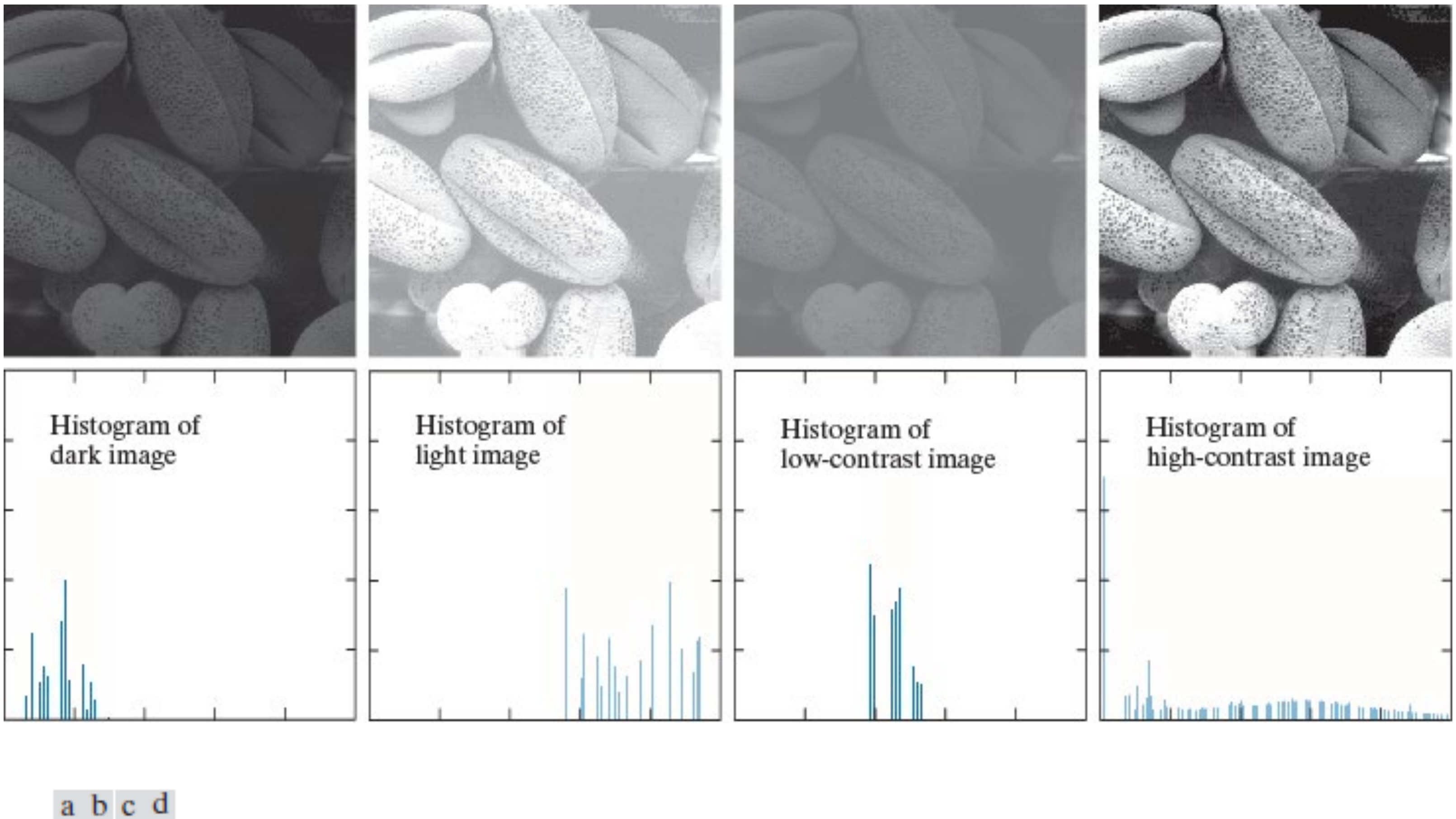


FIGURE 3.16 Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$.

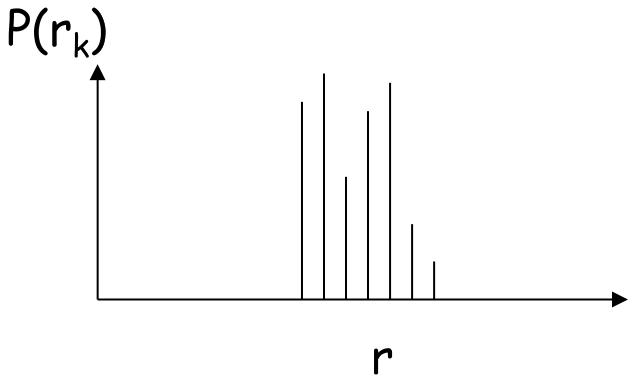
The figure is provided by Pearson Education, Digital Image Processing, Gonzalez & Woods, www.ImageProcessingPlace.com

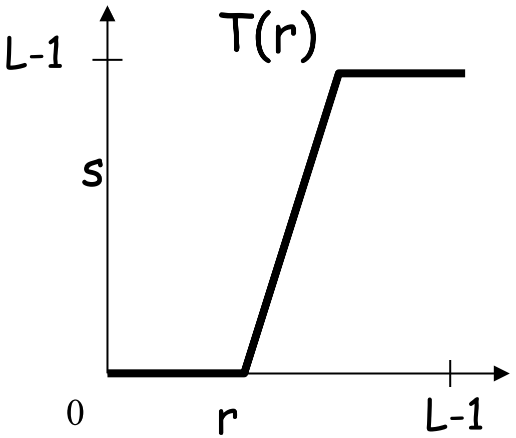
Contrast Stretching

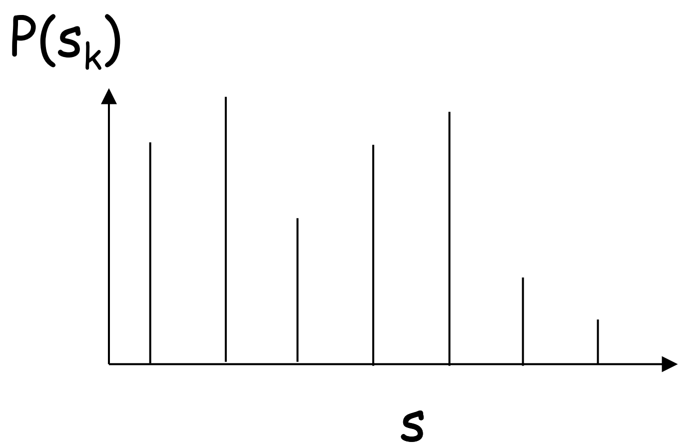
28:00

?

Low Contrast

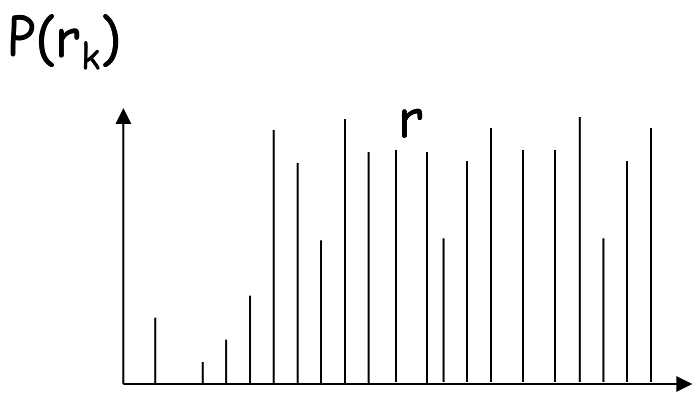


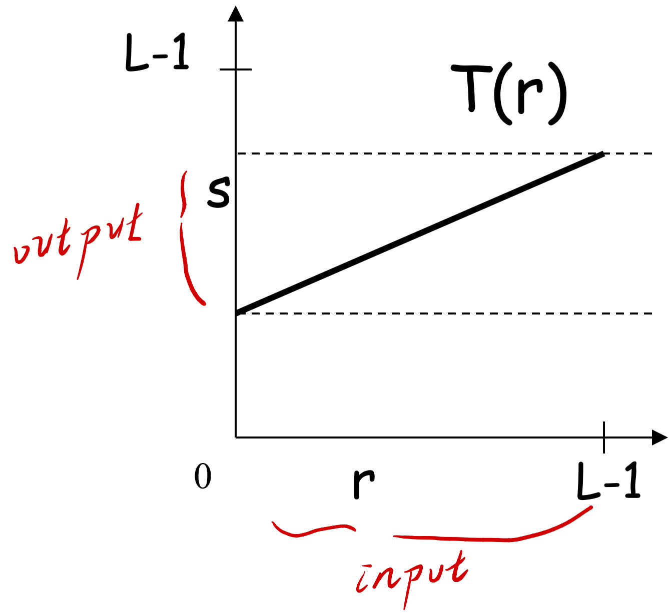


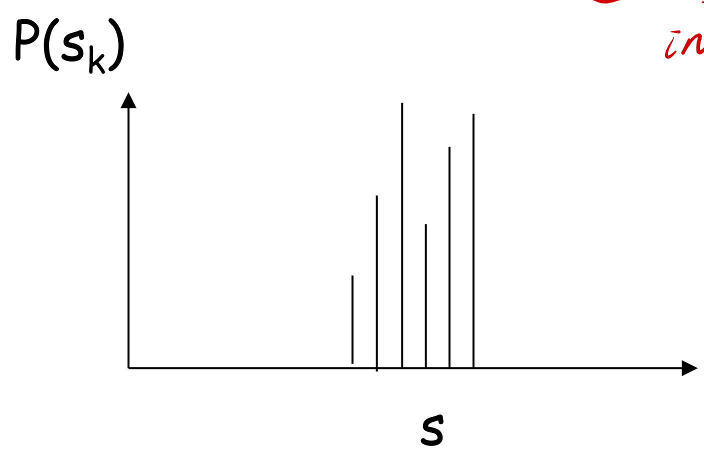


Contrast Compressing

High Contrast







Histogram Equalization

- We want an image with equally many pixels at every gray level, or the output intensity approximately follows a *uniform distribution*.

- That is, a flat histogram, where each gray level, r_k, appears (N/r_m) times

  - where "r_m" is the maximum gray level
  - N is number of pixels in the image

- Example
  - https://demonstrations.wolfram.com/HistogramEqualization/

Histogram Equalization

- The weighted cumulative distribution function (CDF) of r is represented by (in continuous form), (T is the transformation function)

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

- This transformation function seeks to generate an output image with a uniform (flat) probability density function (PDF), independently of the form of $p_r(r)$

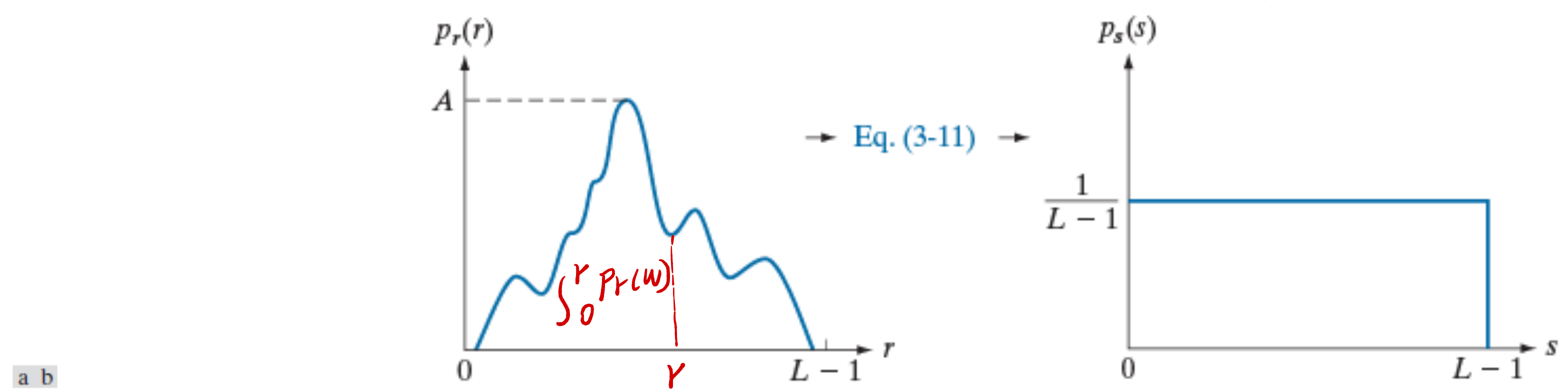


FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying Eq. (3-11) to the input PDF. The resulting PDF is always uniform, independently of the shape of the input.

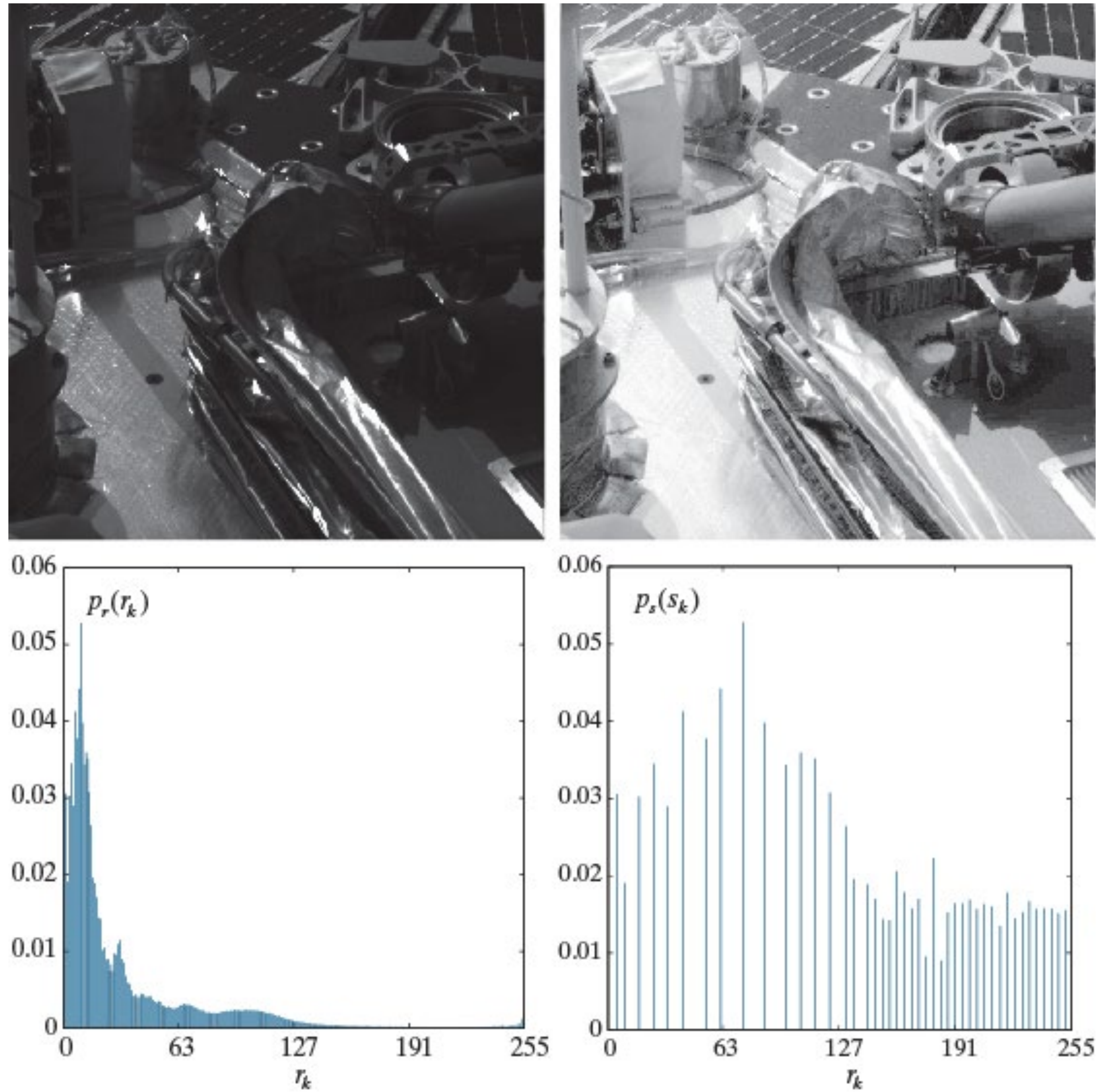


FIGURE 3.22 (a) Image from Phoenix Lander. (b) Result of histogram equalization. (c) Histogram of image (a). (d) Histogram of image (b). (Original image courtesy of NASA.)

The figure is provided by Pearson Education, Digital Image Processing, Gonzalez & Woods, www.ImageProcessingPlace.com

Use T(r) to equalize the histogram

- Assume the input variable r has been normalized between [0,1]

- s = T(r), there are two properties
  - (a) T(r) is single-valued in the interval 0 <= r <= 1
  - (b) 0 <= T(r) <= 1 for 0 <= r <= 1

Conditions (a) and (b)

- (a) T(r) is single-valued and non-decreasing in the interval 0 <= r <= 1
  - Can preserve the order from black to white in the gray scale.
  - Can preserve the basic appearance of an image.

- (b) for 0 <= r <= 1, 0 <= T(r) <= 1
  - Can guarantee a mapping that is consistent with the allowed range of pixel values. No intensity rescaling is needed.

Discrete approximation of *T*

- Let's look at the discrete approximation

intensity level.

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k}\frac{n_j}{N} = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

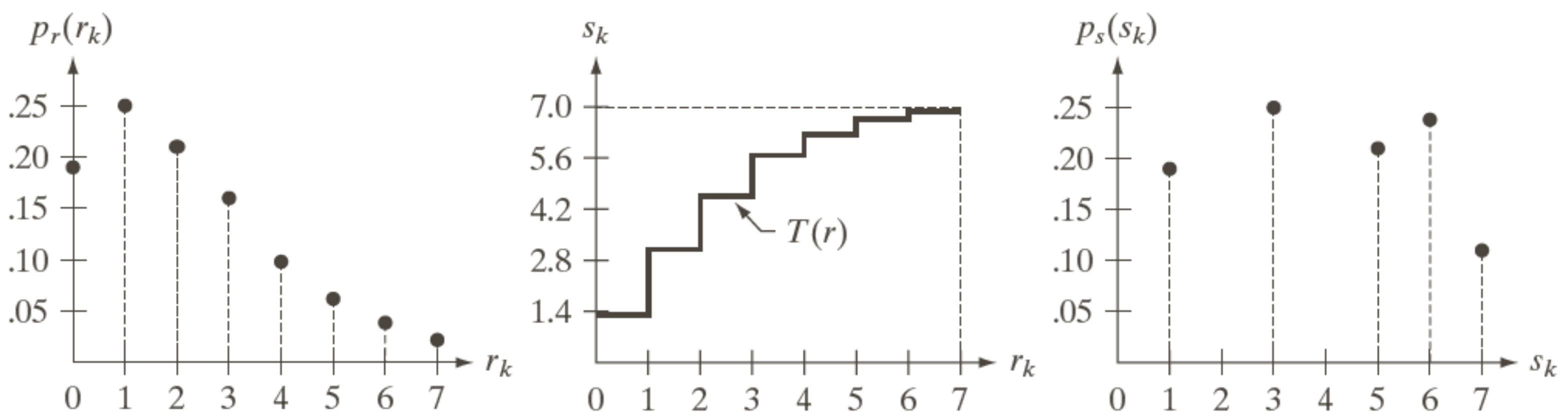
s_k is the output continuous intensity value, $0 <= s_k <= L-1$
r_k is the input continuous intensity value, $0 <= r_k <= L-1$
n_j is number of pixels with the j^{th} gray level
k = 0, 1, 2, 3, . . L-1 (gray levels)

Example 1

3 bits

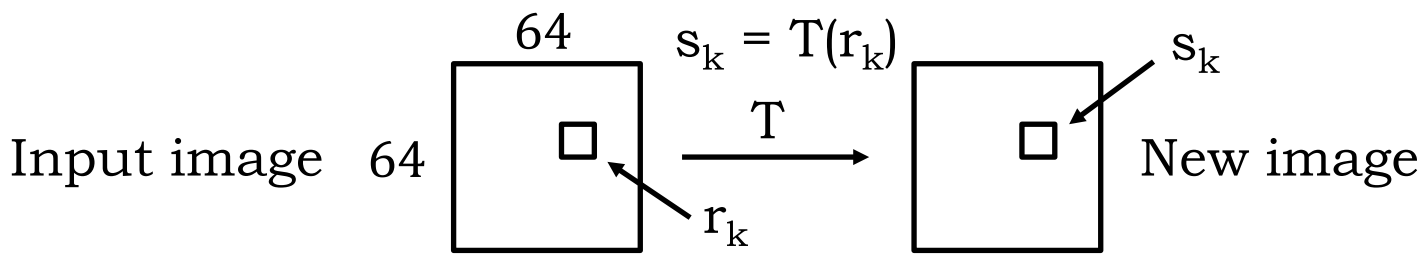
r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

TABLE 3.1 Intensity distribution and histogram values for a 3-bit, 64 × 64 digital image.



a b c

FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.



Total number of pixels = 64*64 = 4096 pixels

$$\frac{2^3-1=7}{2^8-1=255}$$

k	0	1	2	3	4	5	6	7
r_k	0	1	2	3	4	5	6	7
$Pr(r_k)$	$\frac{790}{4096}$ 0.19	$\frac{1023}{4096}$ 0.25	$\frac{850}{4096}$ 0.21	$\frac{656}{4096}$ 0.16	$\frac{329}{4096}$ 0.08	$\frac{245}{4096}$ 0.06	$\frac{122}{4096}$ 0.03	$\frac{81}{4096}$ 0.02
S_k*(1/7)	0.19	0.44	0.65	0.81	0.89	0.95	0.98	1
Gray levels	0.19×7 1	0.44×7 3	5	6	6	7	7	7

$$\sum Pr(r_k) = 1$$

Stretching — Compressing — Compressing

$0 \rightarrow 1$, $1 \rightarrow 3$, $2 \rightarrow 5$

$3, 4 \rightarrow 6$

$5, 6, 7 \rightarrow 7$

The discrete formulation of *histogram equalization* is given by

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j), \; k = 0,1,2,\ldots,L-1$$

where r_k and s_k represent the original and transformed grey levels respectively, L denotes the number of discrete grey levels, T is the transformation function, p_r is the probability density function $p_r(r_j) = n_j/n$, n is the total number of pixels and n_j is the number of pixels having intensity level r_j.

(a) Perform histogram equalization on a 3×3 input image, as shown below. Show all steps.

Answers:

Input image

0	0.33	0.67
0.33	1	0.33
0.67	0.33	0.67

Histogram equalized image

0.11	0.56	0.89
0.56	1	0.56
0.89	0.56	0.89

Round to 2 decimal places

r_k	0	0.33	0.67	1
$P_r(r_k)$	1/9	4/9	3/9	1/9
Transformation function: S_k	1/9 = 0.11	5/9 = 0.56	8/9 = 0.89	9/9 = 1

(b) Explain why the transformation function $T(r_j)$ is increasing in the interval $0 \le r_j \le 1$, where $j = 0, 1, 2, \ldots, L-1$.

Answers:

$$S_k = T(r_k) = \sum_{j=0}^{k} P_r(r_j)$$

$P_r(r_j)$ is non-negative

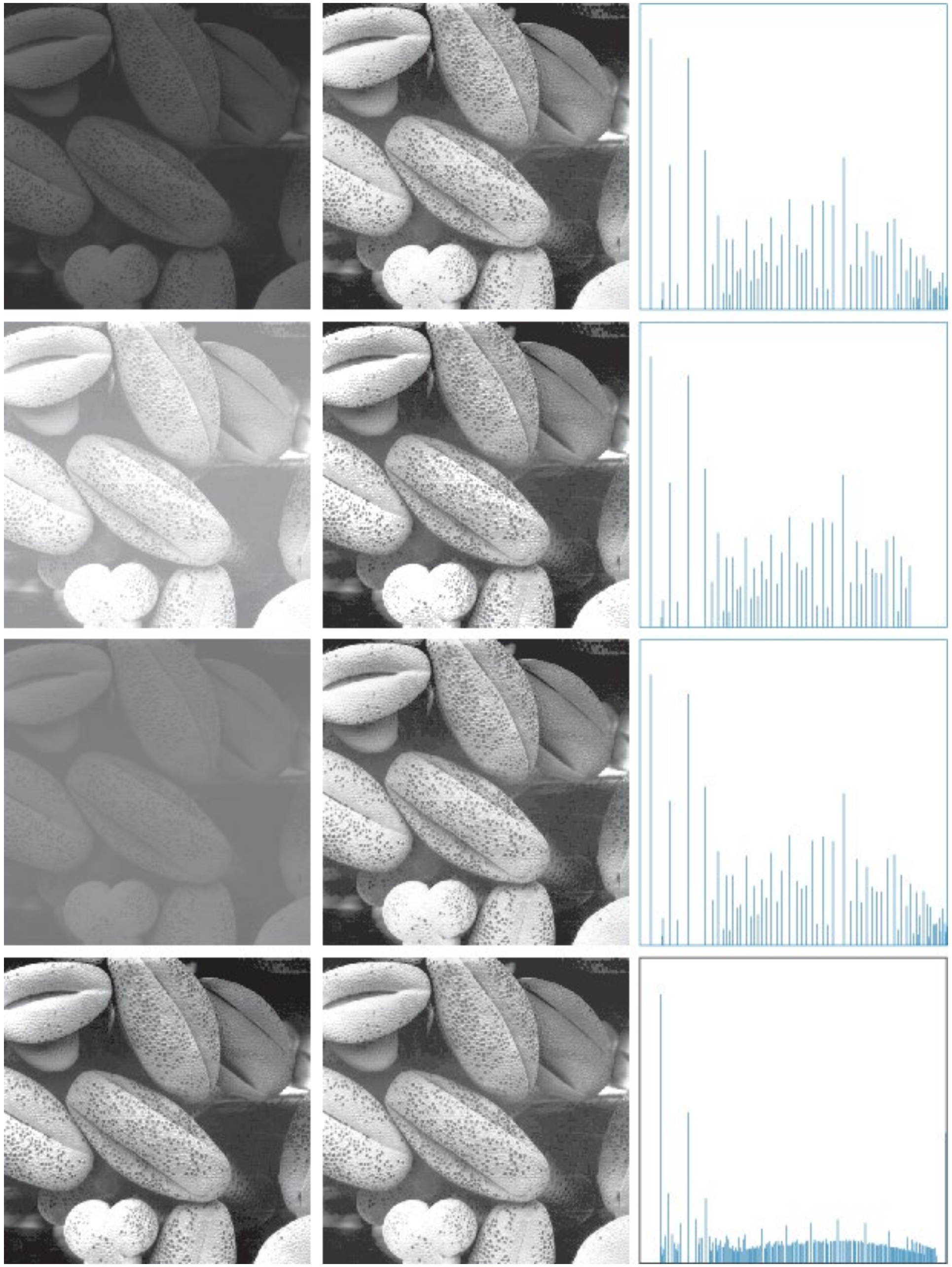
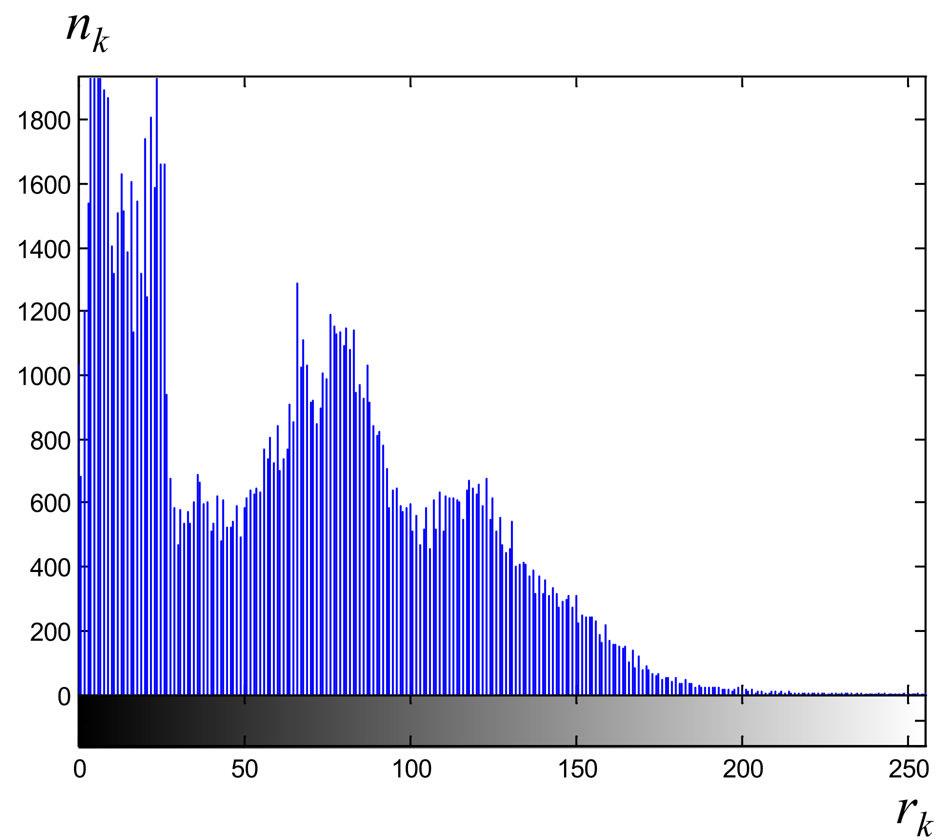


FIGURE 3.20 Left column: Images from Fig. 3.16. Center column: Corresponding histogram-equalized images. Right column: histograms of the images in the center column (compare with the histograms in Fig. 3.16).

Example 2

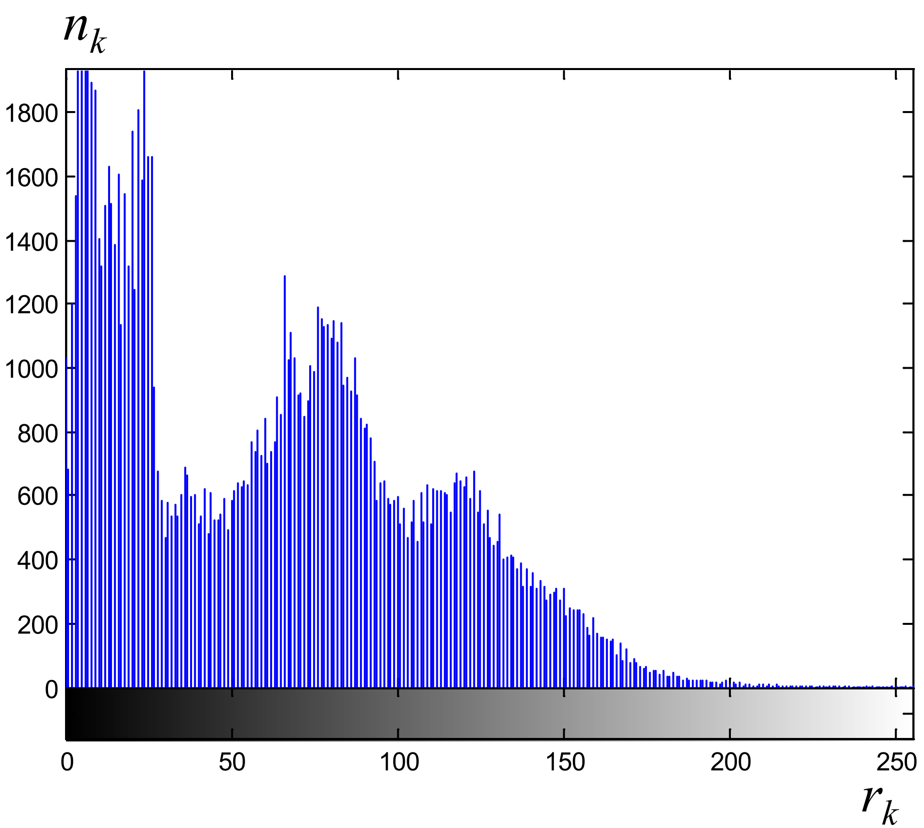
Image

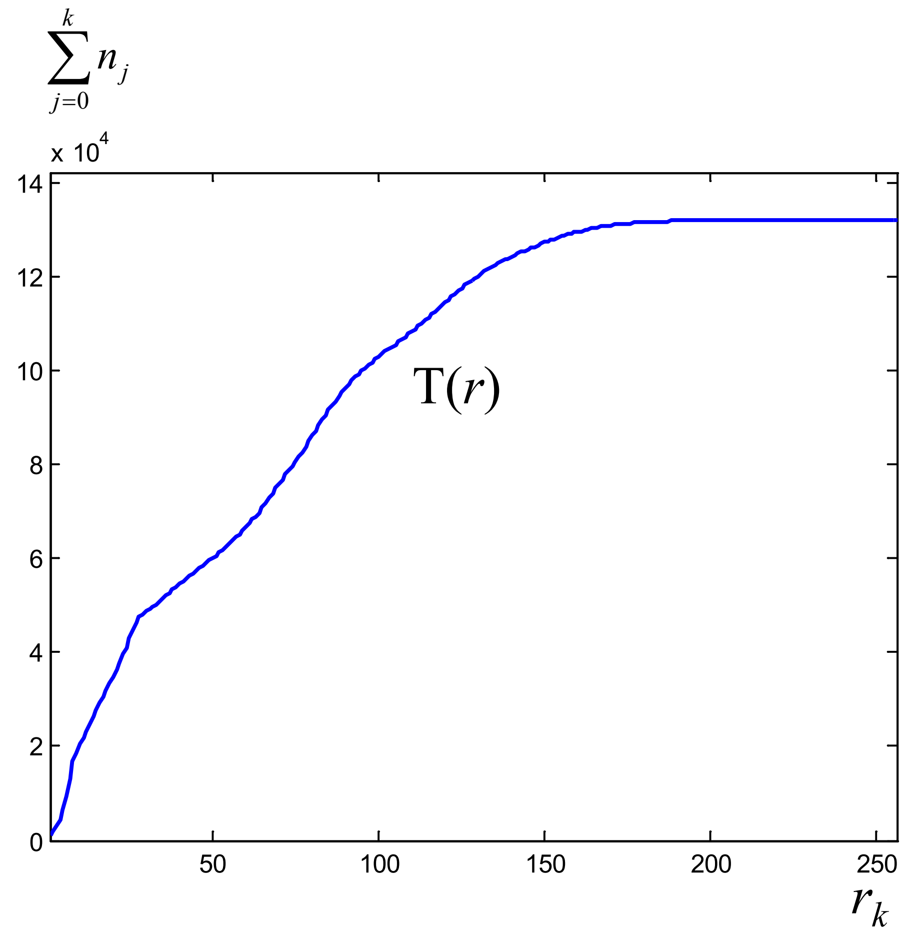


Histogram is not normalized, *y* axis is n_k

To normalize, let y axis = $p_r(r_k) = n_k / N$

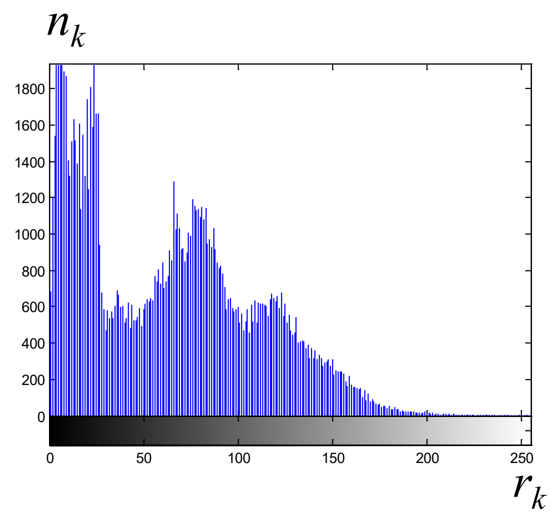
[Yale]

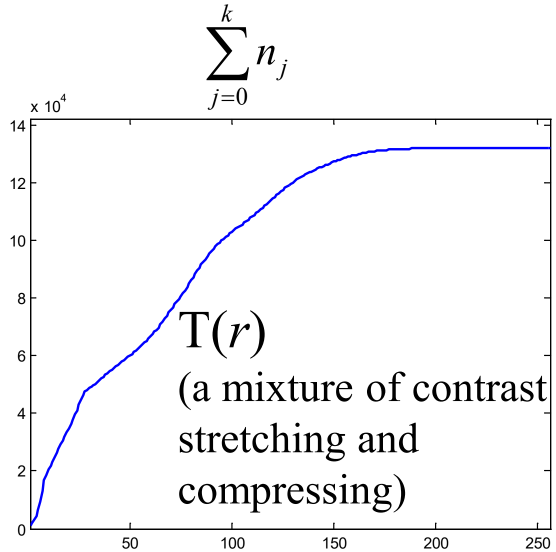


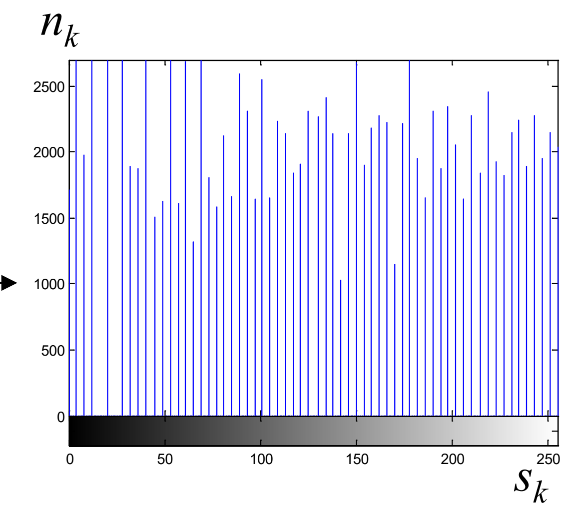


Histogram is not normalized, *y* axis is n_k

To normalize, let y axis = $p_r(r_k) = n_k / N$







Histogram equalization

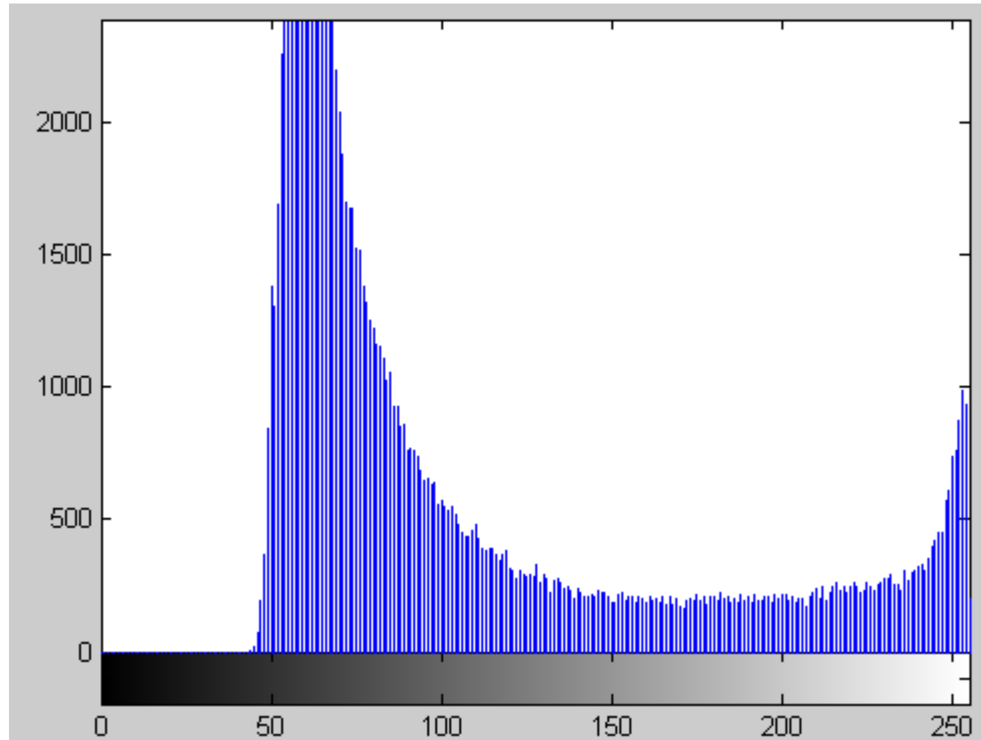
Histogram equalized

[Yale]

Example 3

Image

n_k

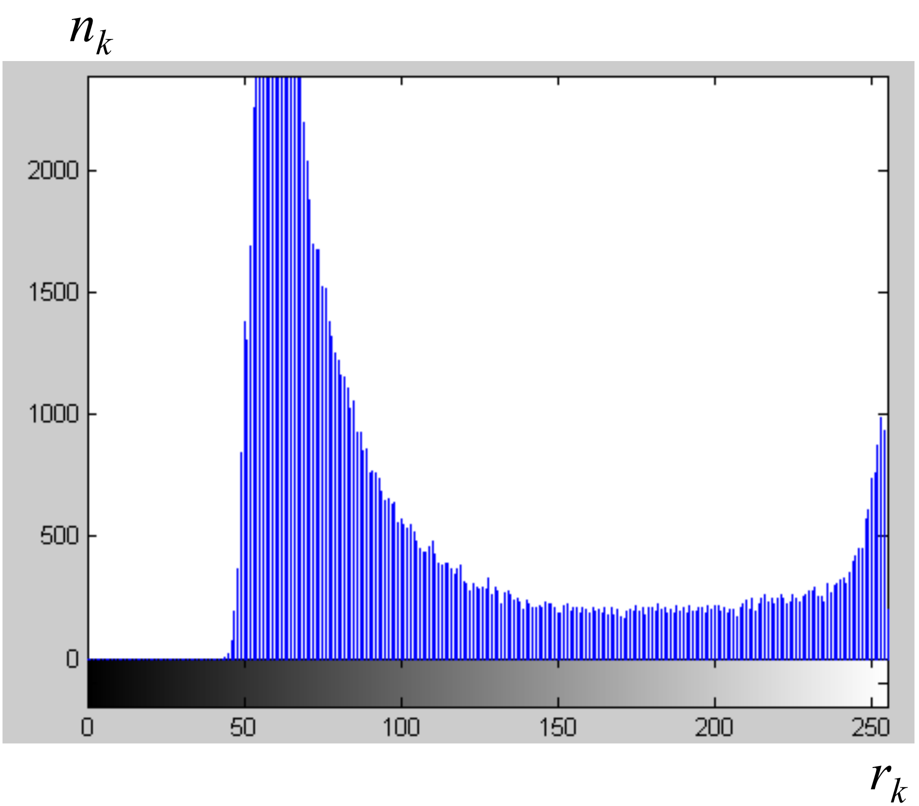


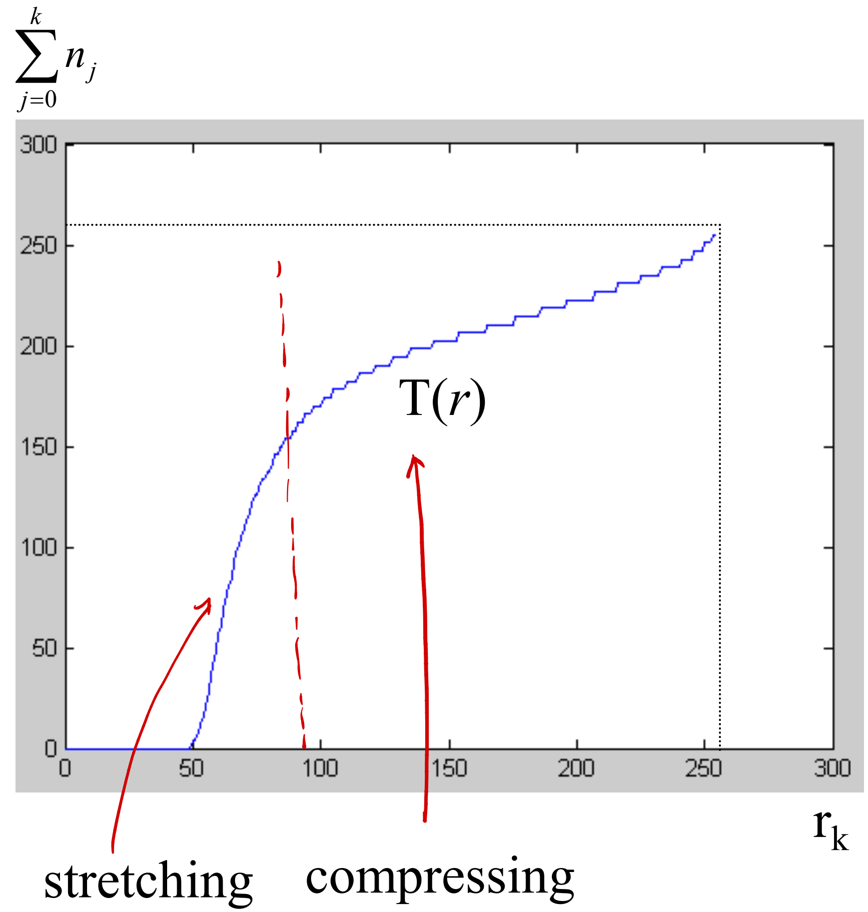
r_k

Histogram

Notice, this is not normalized, y axis is n_k.

To normalize, let y axis = $p_r(r_k)= n_k /N$



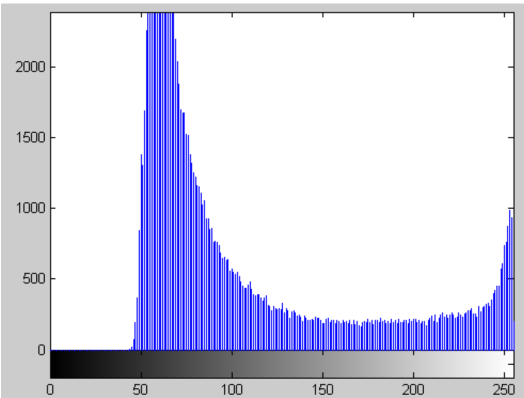


Histogram

Notice, this is not normalized, y axis is n_k.

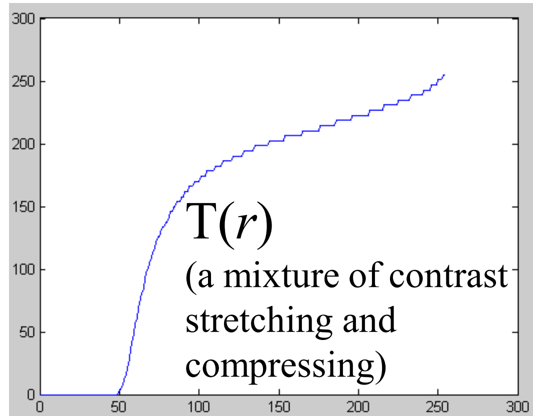
To normalize, let y axis = $p_r(r_k) = n_k / N$

n_k

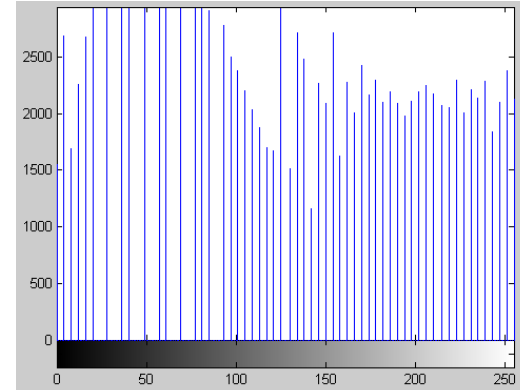


r_k

$$\sum_{j=0}^{k} n_j$$



n_k



s_k

× improve.
lose details

Histogram
equalization

Histogram equalized

Histogram equalization

- Histogram equalization can significantly improve image appearance
  - Automatic
  - User doesn’t have to perform windowing

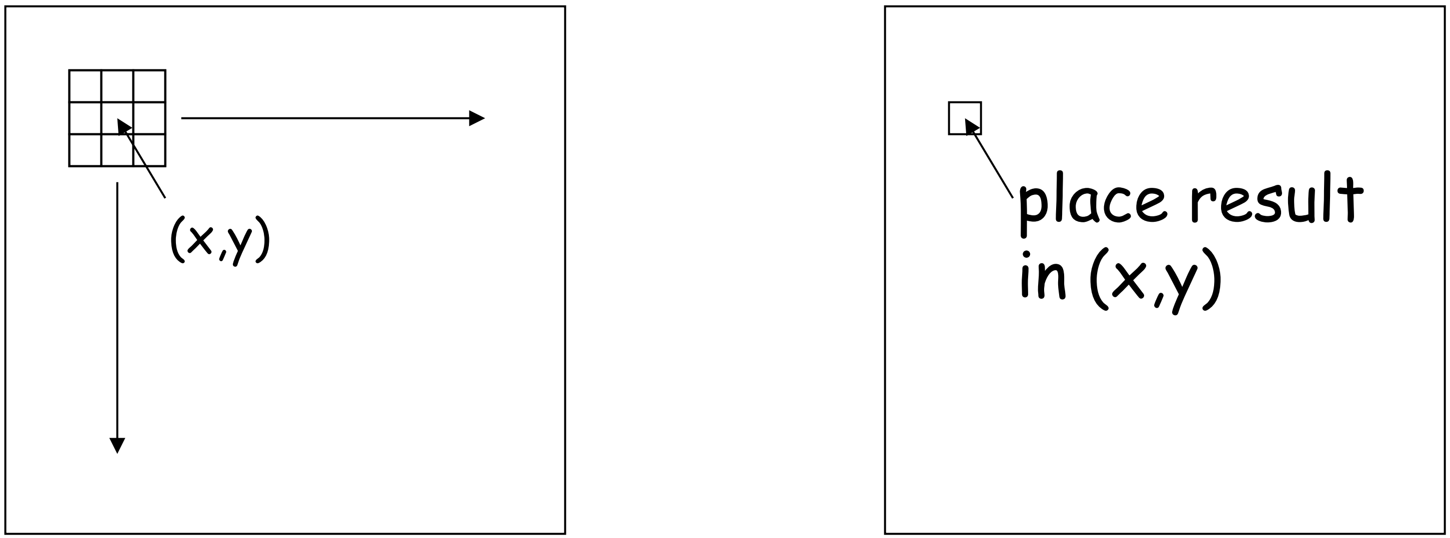
- Nice pre-processing step before face detection
  - Account for different lighting conditions
  - Account for different camera/device properties

Local Enhancement

- Histogram equalization is a global operation
  - Each pixel is processed based on information of the entire image
  - Often enhances global details

- We would like to enhance details over small areas
  - Each pixel is processed based on information of a small area/sub-image

Local Histogram Equalization



Result

calculate histogram
using neighborhood of mxm
about (x,y) each pixel has a table.

Local Histogram

- Apply histogram equalization about a neighborhood around (x,y)
- Transform the gray level for pixel (x,y)
- Move the neighborhood over the rest of the image

Local Histogram

Reveals detail in local areas

Original Global Histogram Local Histogram

Adaptive Histogram Equalization

Figure 4: *Left:* Basic histogram equalization. *Right:* Adaptive histogram equalization.

Source: https://pyimagesearch.com/2021/02/01/opencv-histogram-equalization-and-adaptive-histogram-equalization-clahe/