Intensity Transformations

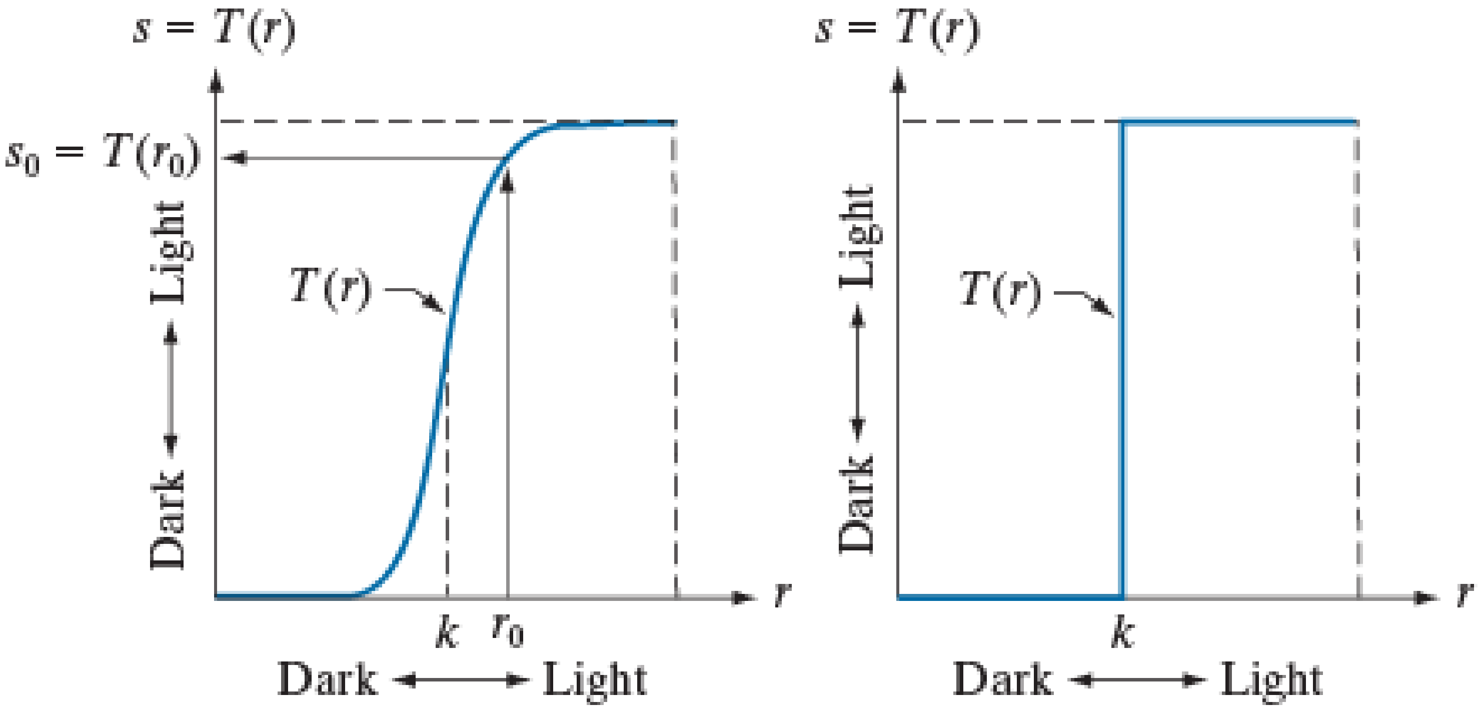
Intensity Transformation

- An intensity transformation is a function that takes the pixel intensities of the image as an input and creates a new image

$$g(x, y) = T[f(x, y)],$$

$$f(x, y) – \ input\ image,$$
$$g(x, y) - output\ image$$
$$T\ is\ a\ transformation$$

Intensity Transformation



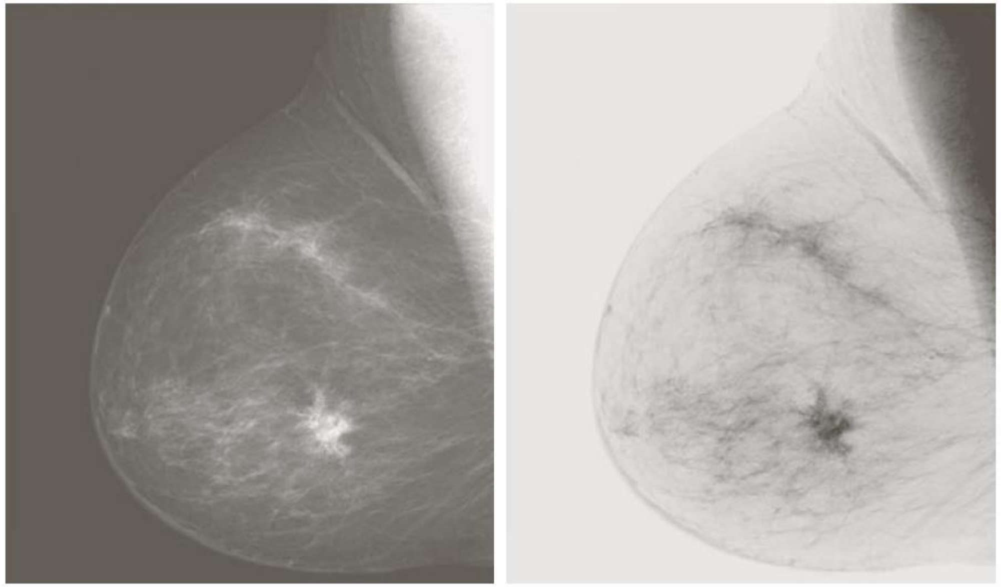
a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

Image Negative (Inverse)

$$g(x, y) = L - 1 - f(x, y),$$

$f(x, y)$ - input image with intensity levels in the range $[0, L-1]$



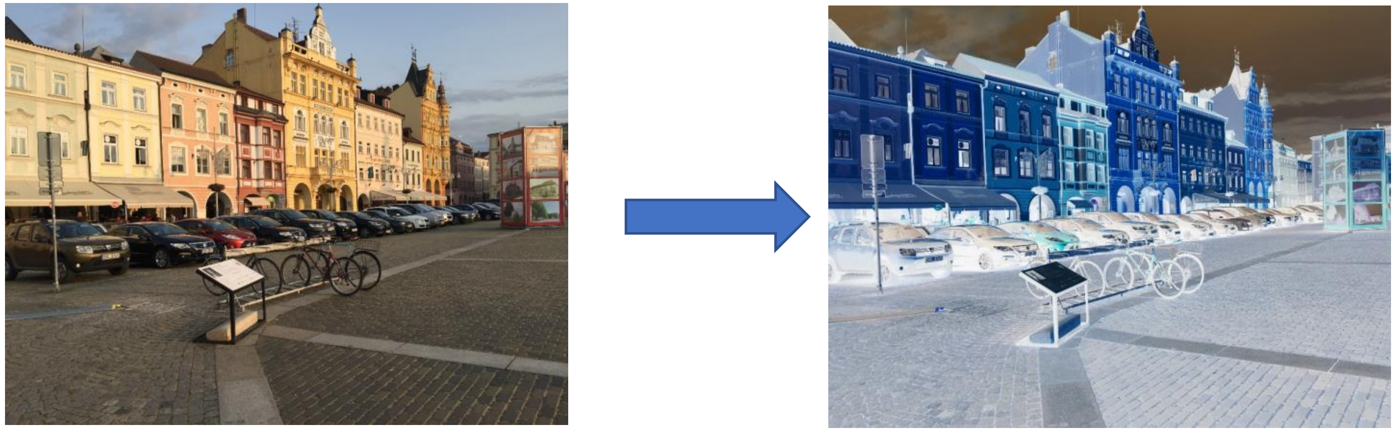
a b

FIGURE 3.4 (a) A digital mammogram. (b) Negative image obtained using Eq. (3-3). (Image (a) Courtesy of General Electric Medical Systems.)

Image Negative (Inverse)

```
import cv2

# Opening the image.
im = cv2.imread('../figures/FamilyTrip.JPG')
# Performing the inversion operation
imnegative = 255 - im
# Saving the image as imageinverse_output.png in Figures folder.
cv2.imwrite('../figures/FamilyTrip_negative.JPG', imnegative)
```



Power law (Gamma) transformation

$$g(x, y) = c \cdot f(x, y)^{\gamma},$$

c, γ are positive constants

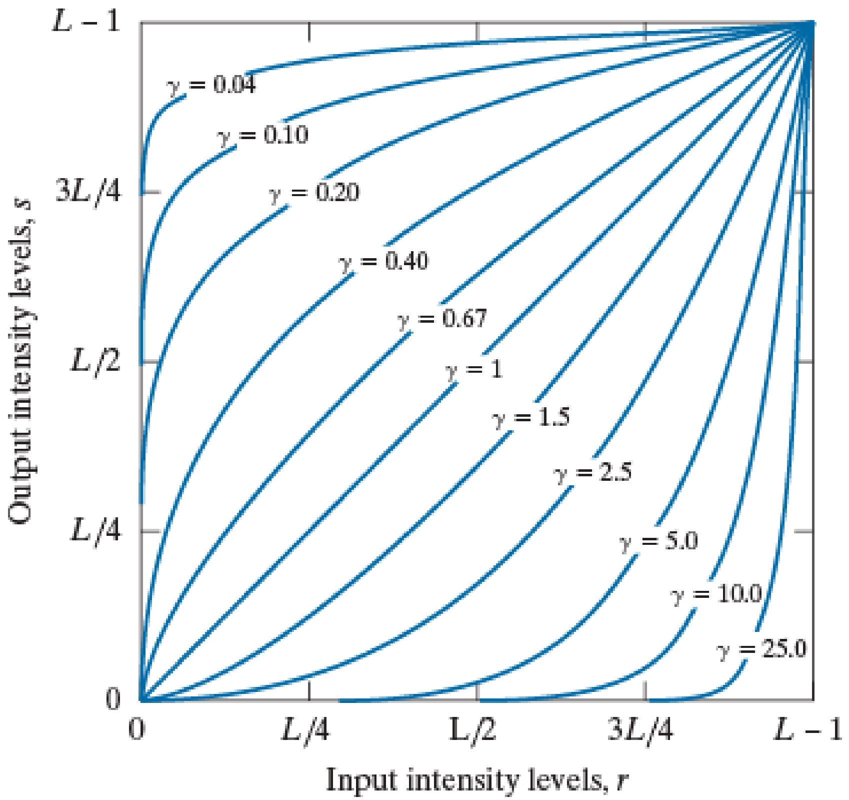
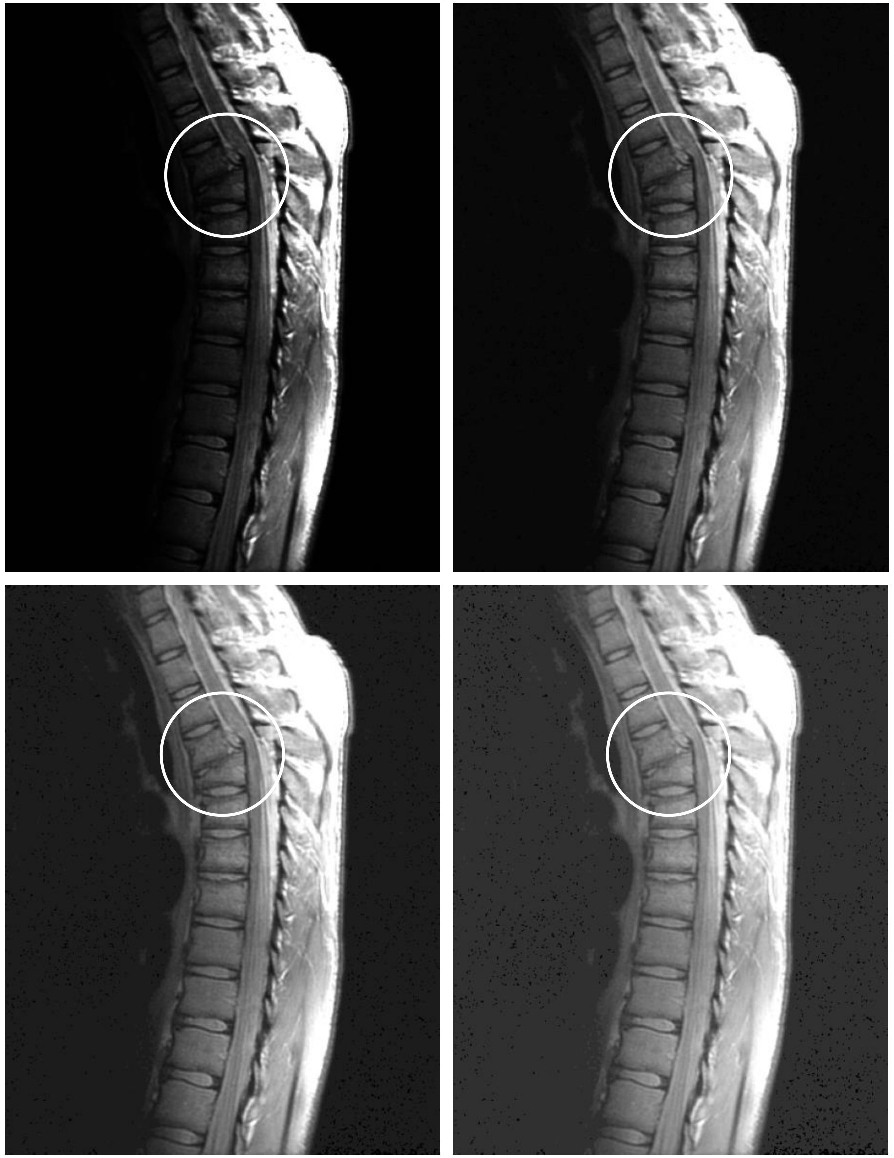


FIGURE 3.6 Plots of the gamma equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.

Power law (Gamma) transformation



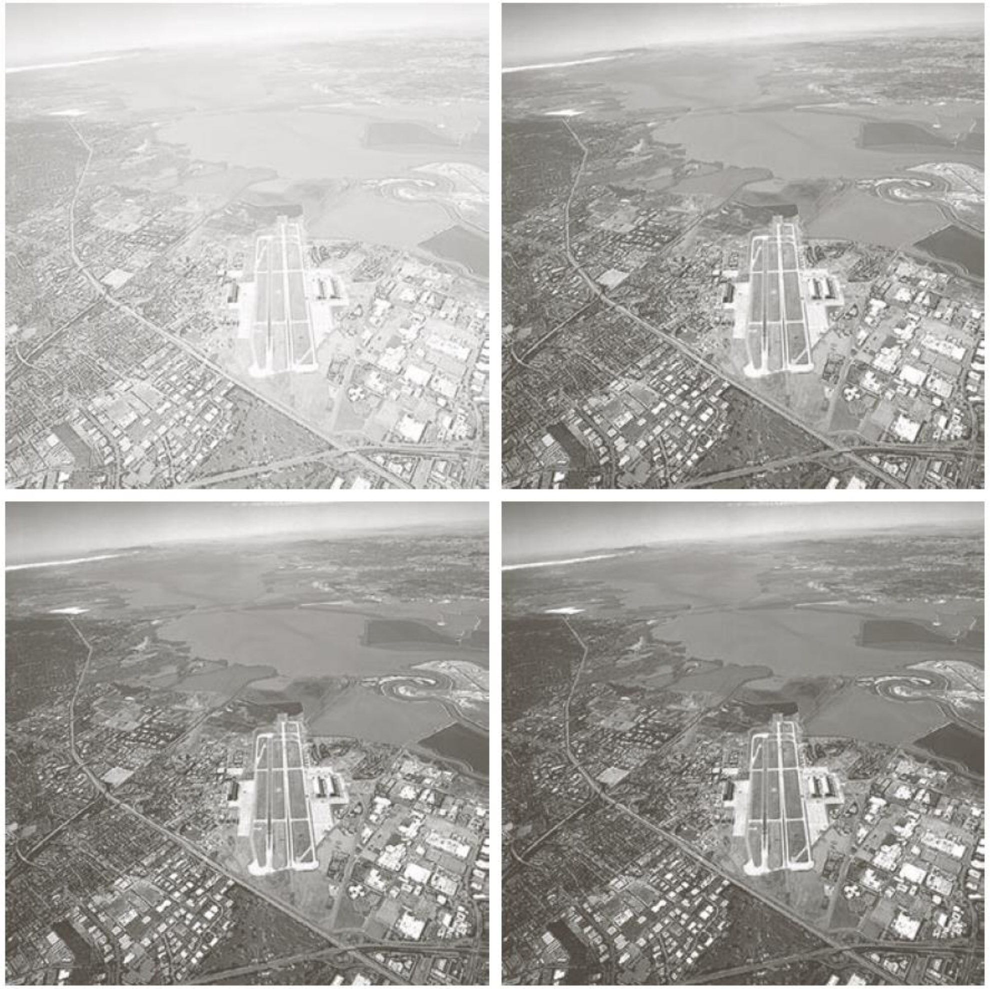
a b
c d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) with $c = 1$ and $\gamma = 0.6, 0.4$, and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

Power law (Gamma) transformation

a b
c d

FIGURE 3.9
(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3-5) with $\gamma = 3.0, 4.0$, and 5.0, respectively. ($c = 1$ in all cases.) (Original image courtesy of NASA.)



Histogram Equalization

- A histogram is a graphical depiction of the distribution of pixel value in an image

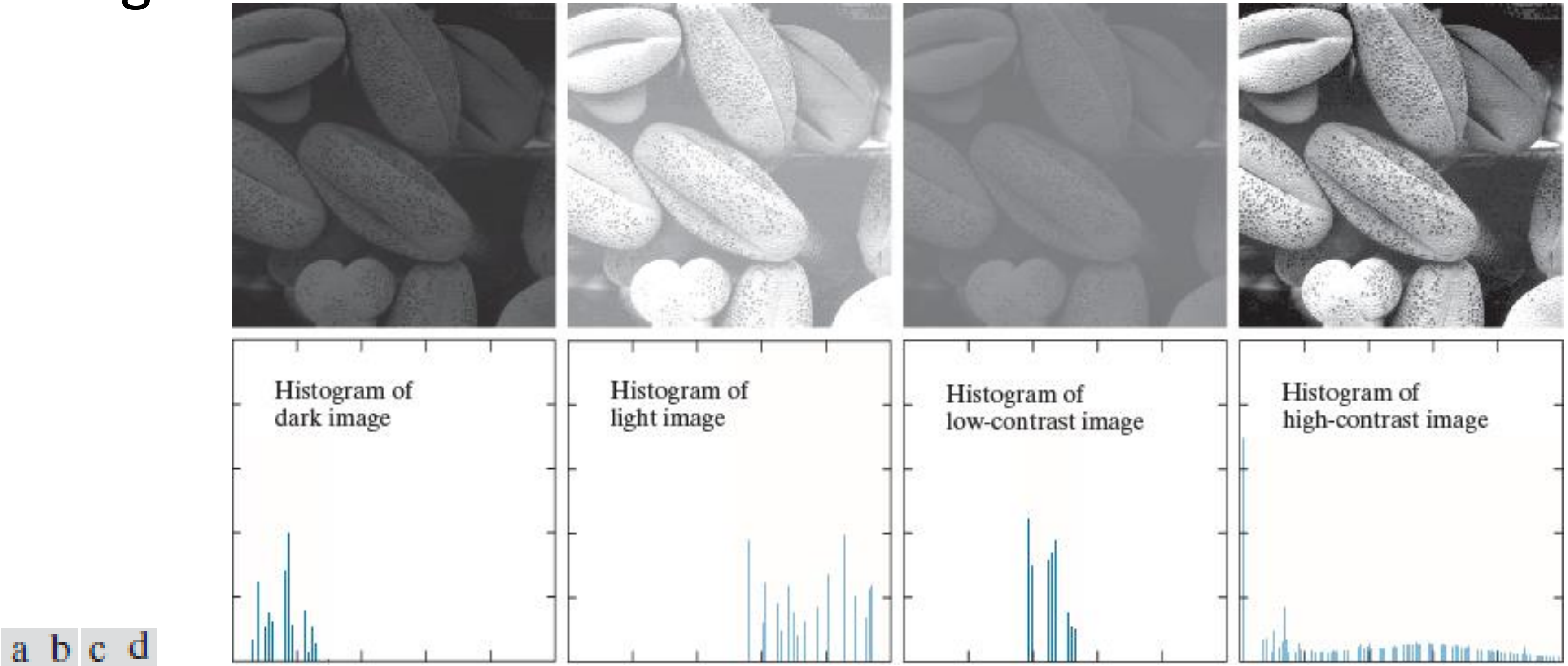


FIGURE 3.16 Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$.

Histogram Equalization

- The histogram of the input image is normalized so that the range of the normalized histogram is [0, 1]

$$p_x(i) = \frac{number\ of\ pixels\ with\ a\ gray\ level\ i}{total\ number\ of\ the\ pixels\ in\ the\ image}$$

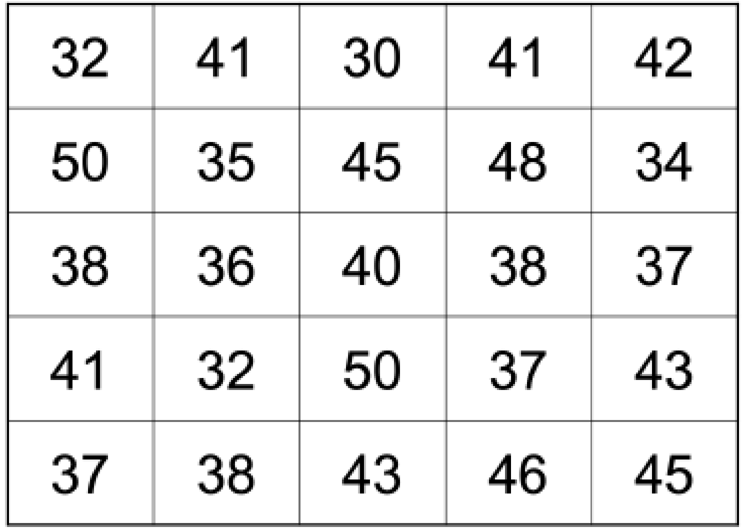
- The cumulative distribution function (CDF) $C(i) = \sum_{j=0}^{i} p_x(i)$

- The histogram equalization transformation is defined

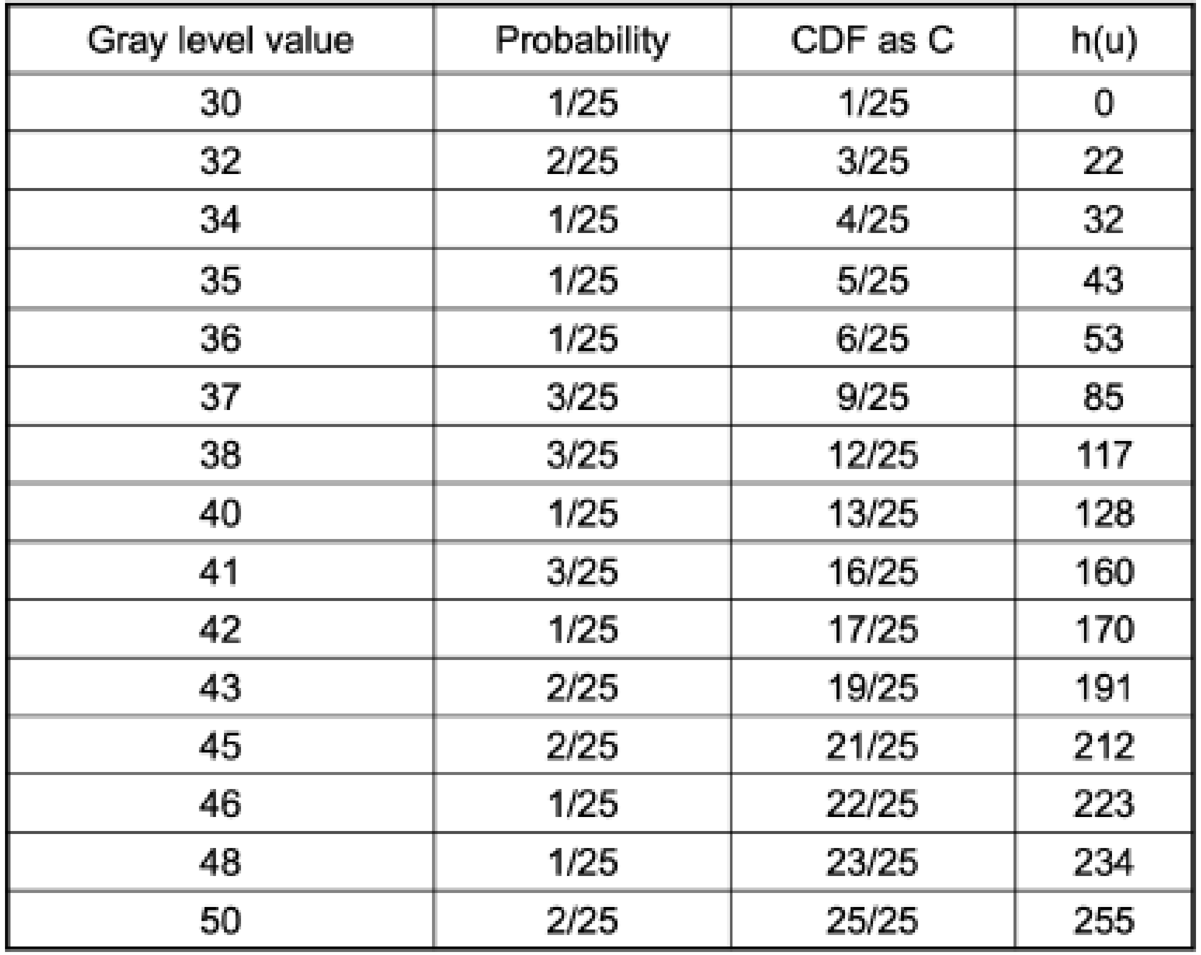
$$h(u) = round(\frac{C(u) - C_{min}}{1 - C_{min}} * (L - 1))$$

C_{min} is the minimum value in the cumulative distribution

Histogram Equalization



32	41	30	41	42
50	35	45	48	34
38	36	40	38	37
41	32	50	37	43
37	38	43	46	45



Gray level value	Probability	CDF as C	h(u)
30	1/25	1/25	0
32	2/25	3/25	22
34	1/25	4/25	32
35	1/25	5/25	43
36	1/25	6/25	53
37	3/25	9/25	85
38	3/25	12/25	117
40	1/25	13/25	128
41	3/25	16/25	160
42	1/25	17/25	170
43	2/25	19/25	191
45	2/25	21/25	212
46	1/25	22/25	223
48	1/25	23/25	234
50	2/25	25/25	255

$$C(i) = \sum_{j=0}^{i} p_x(i)$$

$$h(u) = round(\frac{C(u) - C_{min}}{1 - C_{min}} * (L - 1))$$

Histogram Equalization

```
img = cv2.imread(‘ifilename',cv2.IMREAD_GRAYSCALE)

equ = cv2.equalizeHist(img)

cv2.imwrite(‘ofilename,equ)
```

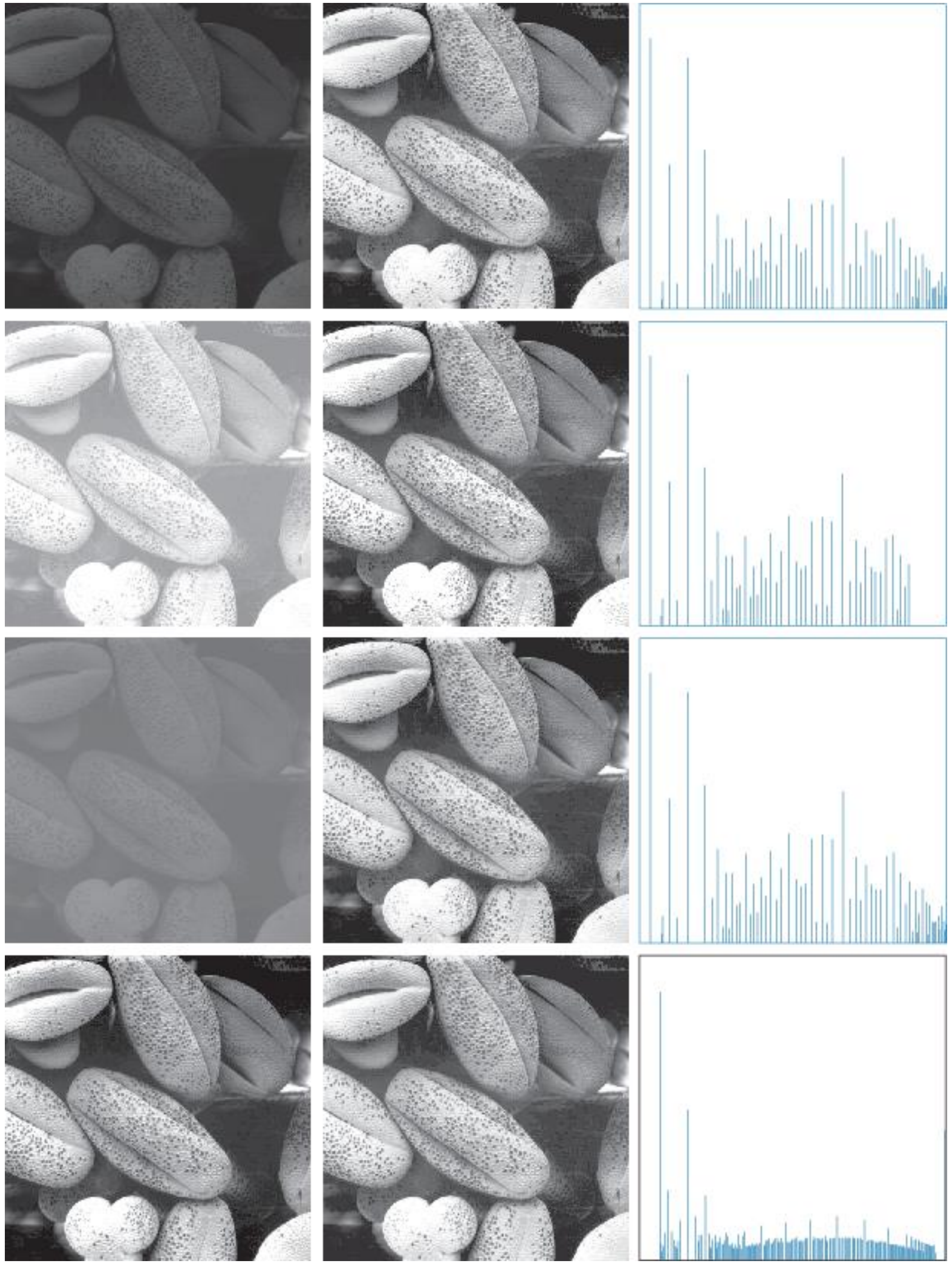


FIGURE 3.20 Left column: Images from Fig. 3.16. Center column: Corresponding histogram-equalized images. Right column: histograms of the images in the center column (compare with the histograms in Fig. 3.16).

Contrast Limiting Adaptive Histogram Equalization (CLAHE)

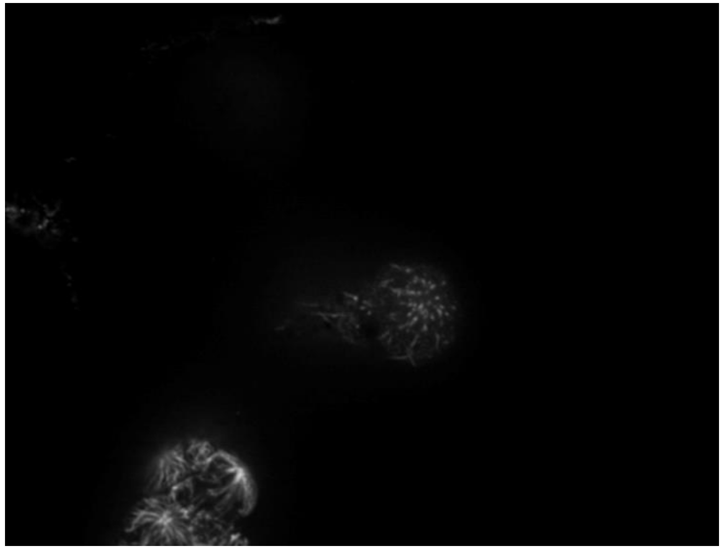
- Histogram equalization works well when the distribution of the pixel values is similar throughout the image.
- BUT, when some regions are significantly different (e.g. lighter or darker) from other parts of the image, the contrast in some regions might not be enhanced correctly.
- Solution: divide the images into cells and perform histogram equalization in each cell.
    - A bit more details ☺

Contrast Limiting Adaptive Histogram Equalization (CLAHE)

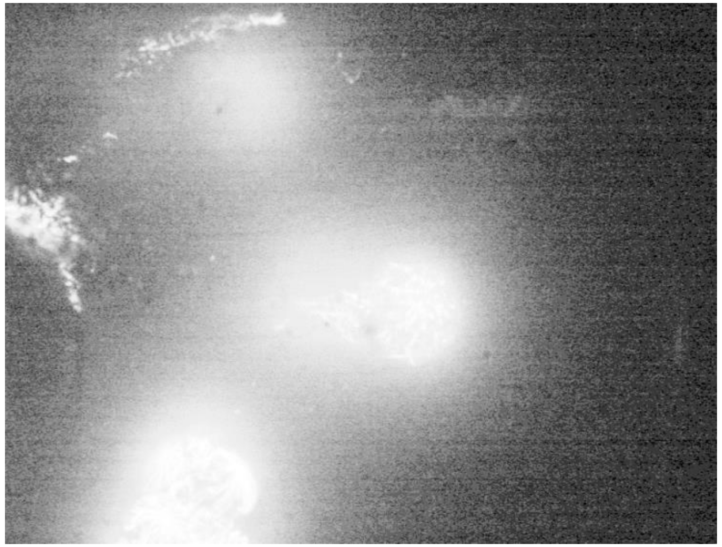
- Instantiate the CLAHE algorithm using `cv2.createCLAHE`
- Call the `.apply` method on the CLAHE object to apply histogram equalization

```
clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(20,20))
cl1 = clahe.apply(im)
```

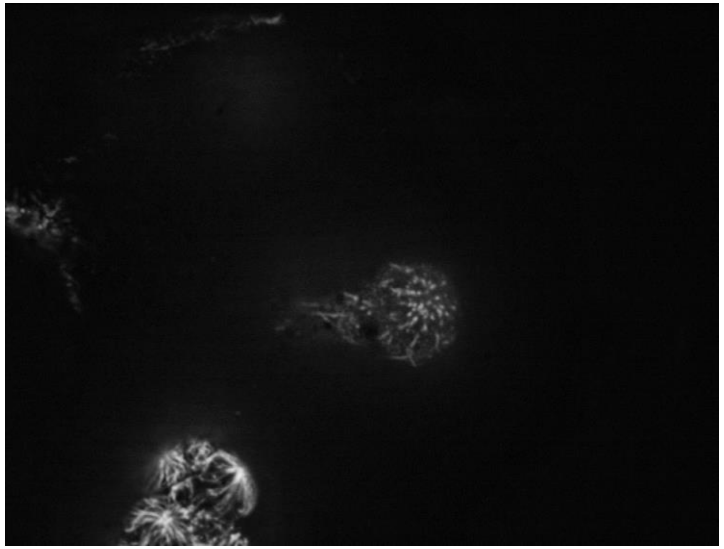
Contrast Limiting Adaptive Histogram Equalization (CLAHE)



Original

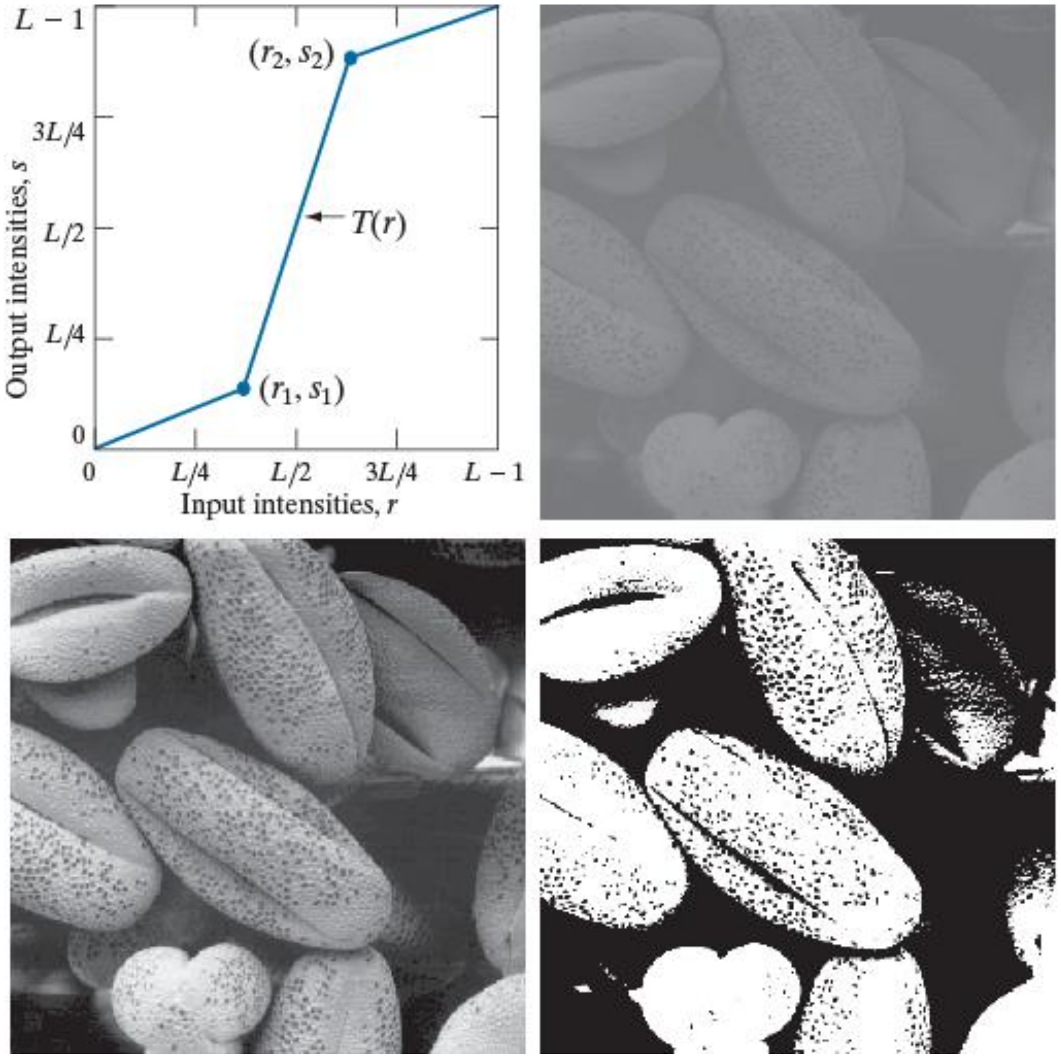


Histogram Equalization



CLAHE

Contrast Stretching

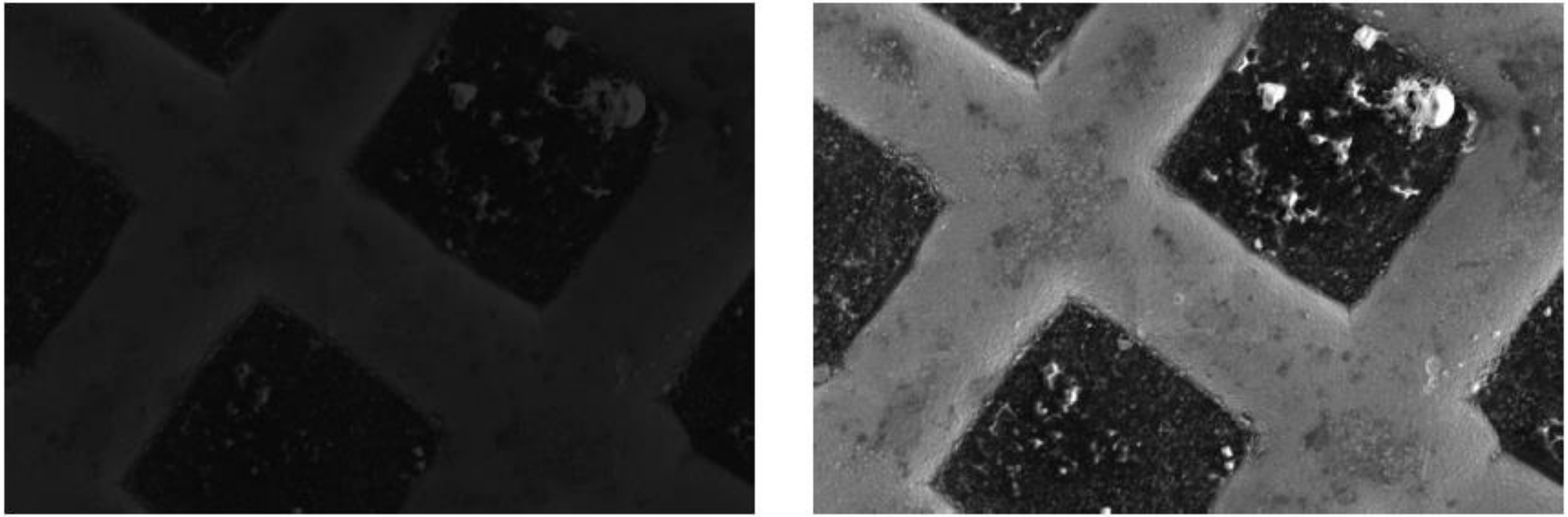


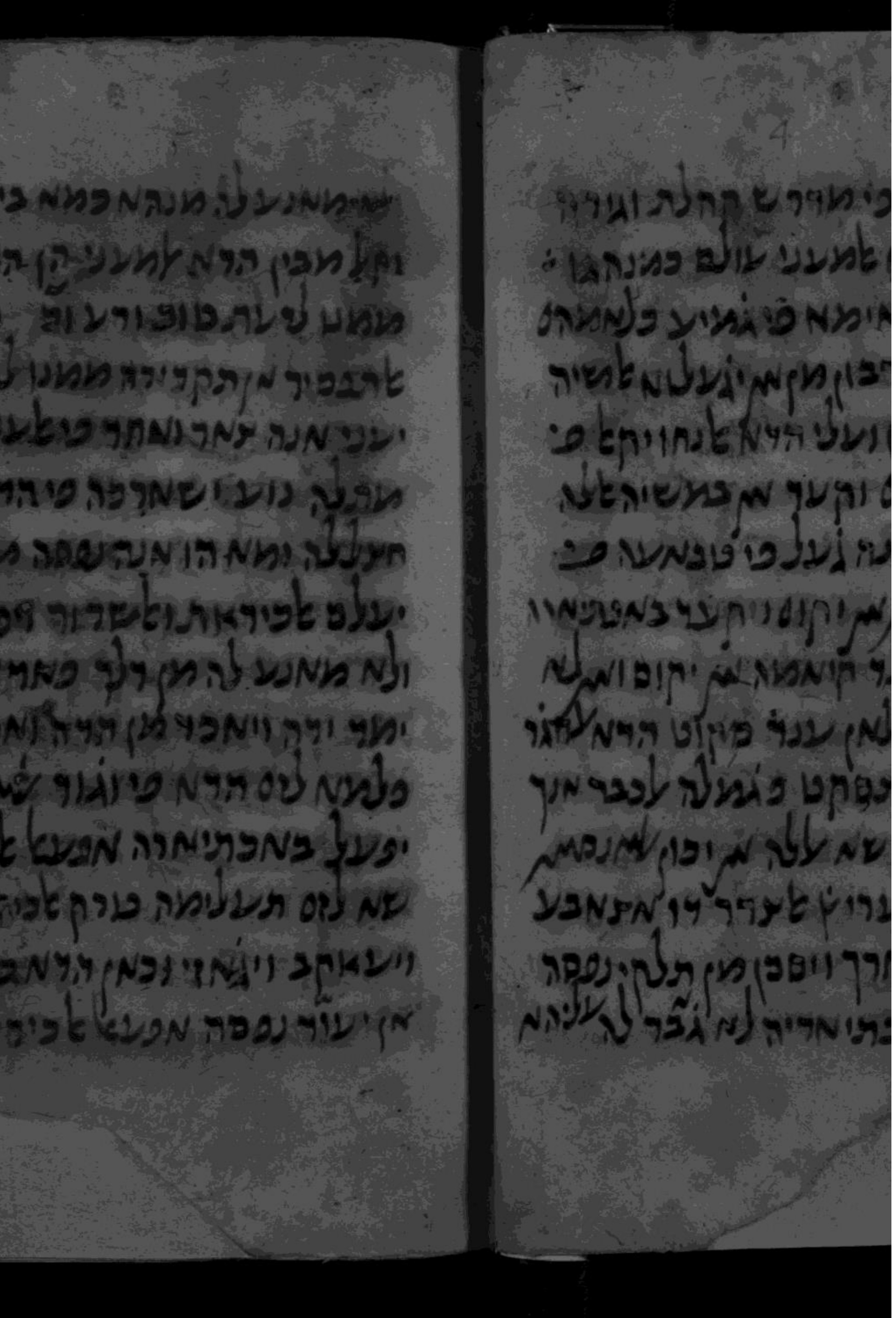
a b
c d

FIGURE 3.10 Contrast stretching. (a) Piecewise linear transformation function. (b) A low-contrast electron microscope image of pollen, magnified 700 times. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

Min-Max Contrast Stretching

$$g(x,y) = 255 * \frac{f(x,y) - x_{min},}{x_{max} - x_{min}}$$





Practice

The focus of this exercise is to experiment with intensity transformations to enhance an image of an ancient manuscript using:

1. Histogram equalization
2. CLAHE
3. Gamma correction (experiment with different values of gamma)
4. Contrast stretching

Once (according to your judgment) you have the best visual result for each transformation, try to explain the reasons for the major differences between them.