EE 4211 Computer Vision

Lecture 2A: Image enhancement (Spatial)

Semester B, 2021-2022

Schedules

Week	Date	Topics
1	Jan. 11 (face to face)	Introduction/Imaging
2	Jan. 18 (online)	Image enhancement in spatial domain
3	Jan. 25 (online)	Image enhancement in frequency domain (HW1 out)
4	Feb. 8	Morphological processing
5	Feb. 15	Image restoration (HW1 due)
6	Feb. 22	Midterm (no tutorials this week)
7	Mar. 1	Edge detection (HW2 out)
8	Mar. 8	Image segmentation
9	Mar. 15	Face recognition with PCA, LDA (tutorial on deep learning framework) (HW2 due)
10	Mar. 22	Face recognition based on deep learning Image segmentation based on deep learning (tutorial on coding)
11	Mar. 29	Object detection with traditional methods (Quiz) Object detection based on deep learning
12	Apr. 5	Events / Public Holidays
13	Apr. 12	Invited project presentation and Summary

Notes

- Zoom recording

  No recording after each lecture

  Will provide the recordings one week before the midterm and when we finish all lectures (for you to prepare the final exam)

- Project issues (illustrate it next week)
- PPT file update issues

  will upload one version before the lecture;

  will be updated after the lecture

Image Enhancement

- Image Enhancement – A set of image processing operations applied on images to produce good images useful for a specific application.
- The reasons for doing this include:
  - Make images more visually appealing
  - Highlight interesting detail in images
  - Remove noise from images

Examples

- Making Images More Visually Appealing

Gamma
Correction

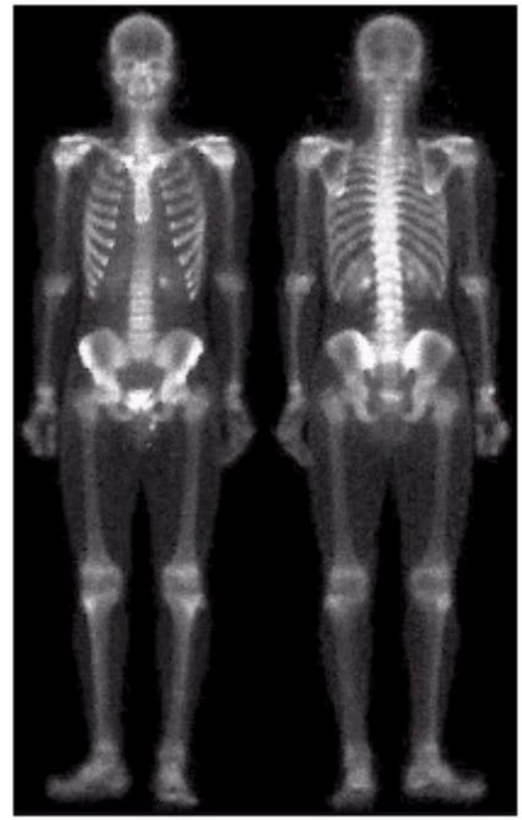
To boost
dark
values

Examples

- Highlighting Interesting Detail

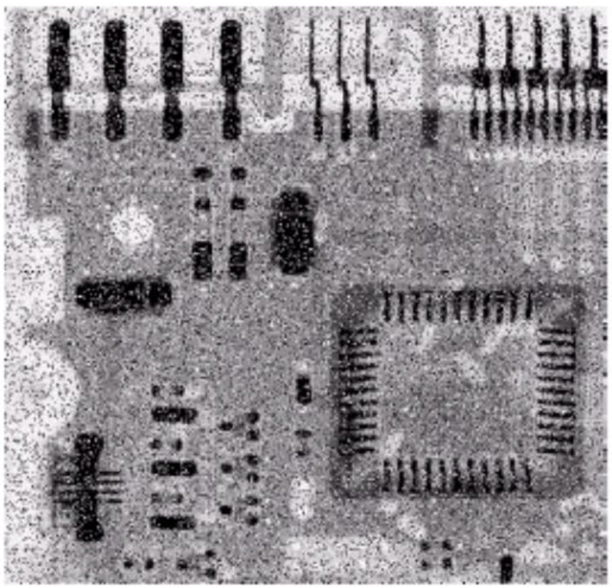
Combining Spatial Enchantment Techniques

Highlighting the skeleton



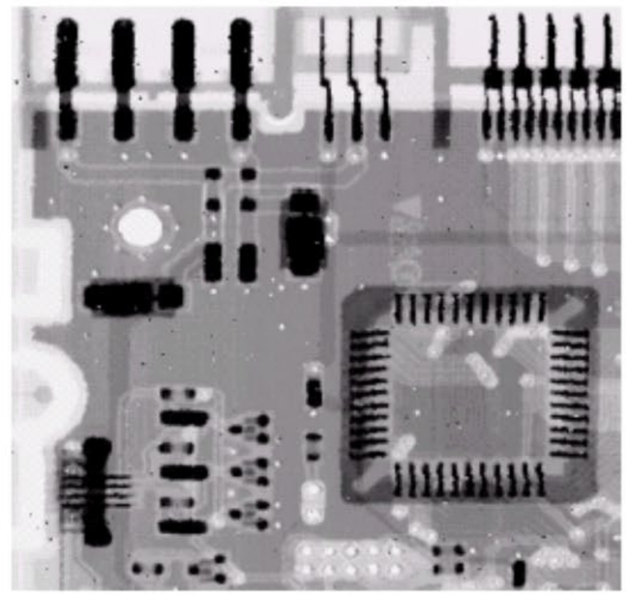
Examples

- Removing Noise From Image



Median
Filter

Remove
Slat and
Pepper
Noise



Spatial Domain vs. Frequency Domain

- Spatial Domain (image plane)
  - Techniques are based on direct manipulation of pixels in an image
- Frequency Domain
  - Techniques are based on modifying the spectral transform (in our course, we'll use Fourier transform) of an image
- There are some enhancement techniques based on various combinations of methods from these 2 domains

Spatial Domain Topics

- Point processing – Gray values change without any knowledge of its surroundings (Part I)
  - Log, power-law, piecewise linear
  - Histogram Equalization
- Neighborhood processing – Gray values change depending on the gray values in a small neighborhood of pixels around the given pixel (Part II)
  - Smoothing filters
  - Median filters
  - sharpening

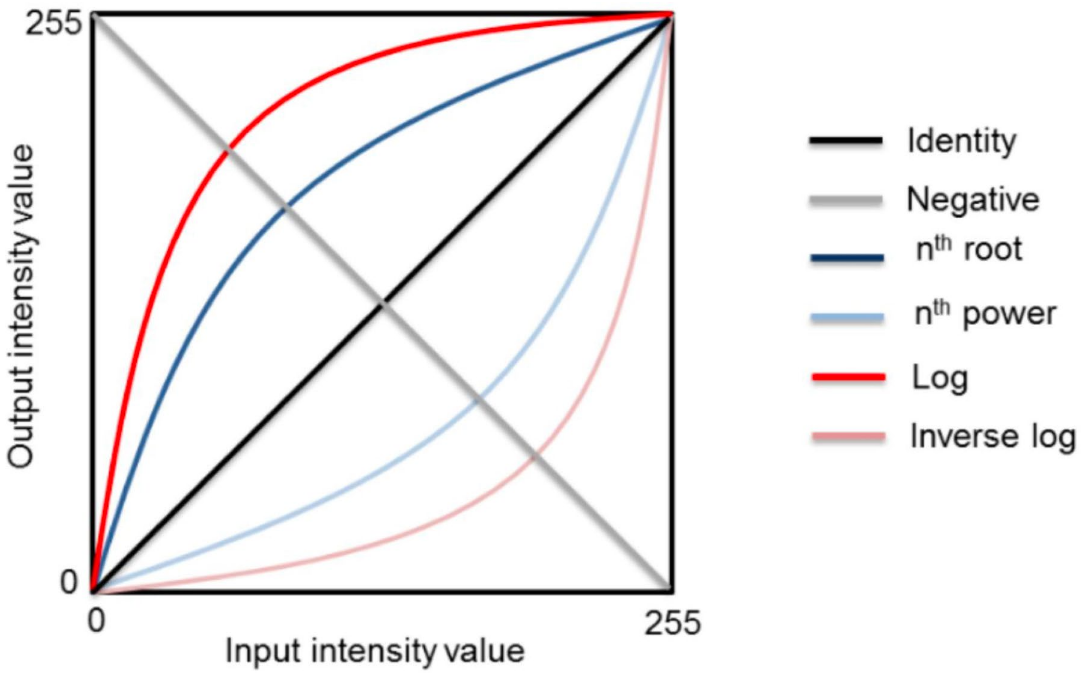
Spatial Domain Process

- $g(x,y) = T[f(x,y)]$
  - $f(x,y)$: input image
  - $g(x,y)$: output image
  - T: an operator

- The intensity of output $g(x,y)$ only depends on the intensity of the input $f(x,y)$ at the same coordinate of (x,y).

Common Intensity Transformations

- Three most common types of grey level transformations:
  - Linear: Negative/Identity
  - Logarithmic: Log/Inverse log
  - Power law: nth power/nth root



Negative Intensity Transformation

- Negatives Transformation:
  - $s = L\text{-}1 - r$
  - $s = 255 - r$
  - $g(x,y) = 255 - f(x,y)$

- Reversing the intensity levels of an image

If a pixel in an image is represented by 8bits, then every pixel can represent values from $0 \sim 2^8\text{-}1$

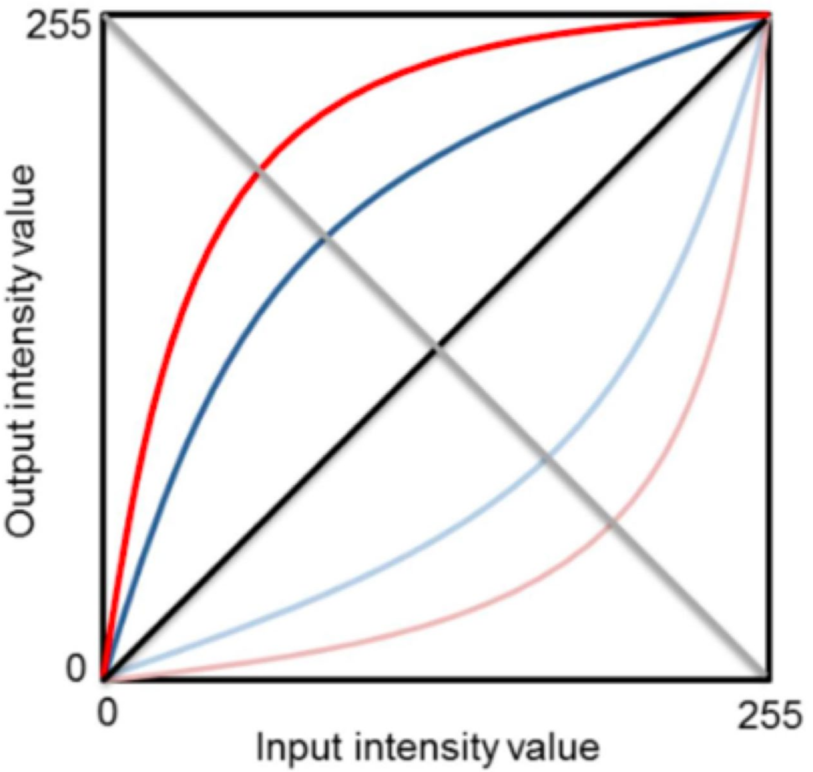
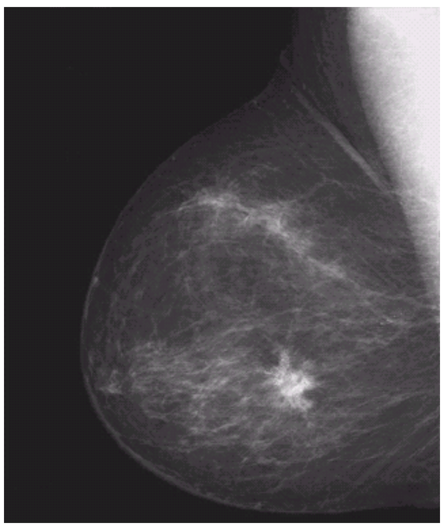
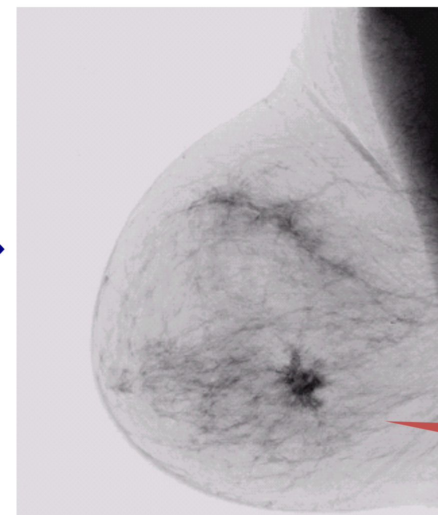


Image Negative Examples

- Suitable to enhance white or gray detail in dark regions of an image, especially when the black area is dominant in size

Original Image





Negative Image

Small lesion

```
%% Negative tranform %2A
clc;
clear all;
I=imread('Fig0304(a)(breast_digital_Xray).tif');
I_new=255-I;
figure;
subplot(1,2,1);imshow(I); title('original')
subplot(1,2,2);imshow(I_new);title('Image with negative transform');
```

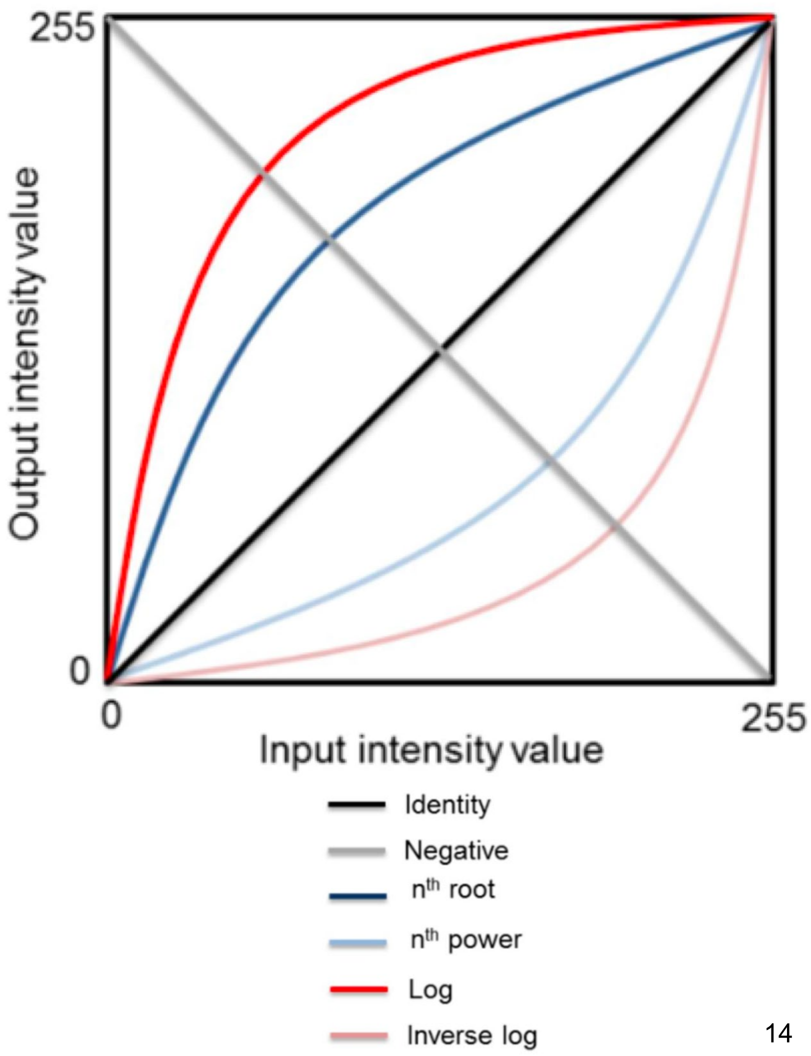
Log Transformation

- Log transformation:

  $s = c \cdot \text{Log}(r + 1)$

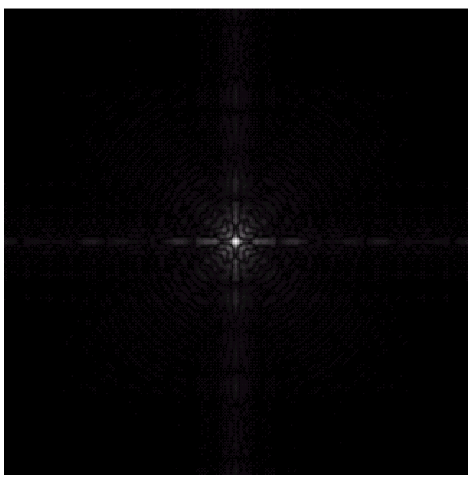
  $g(x,y) = c \cdot \text{Log}(f(x,y) + 1)$

- Map a narrow range of low input grey level values into a wider range of output values
  - Expand the values of dark pixels
  - Compress higher values of lighter pixels
- The inverse log transformation performs the opposite transformation

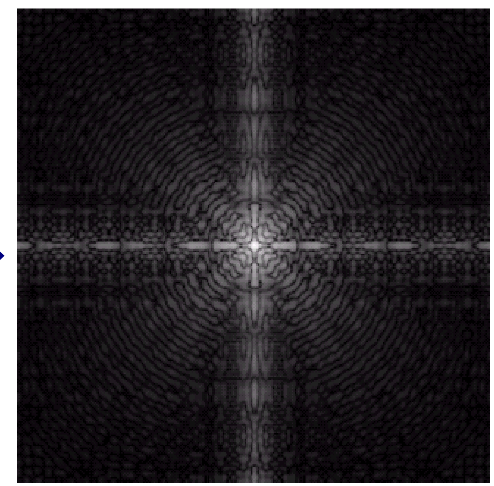


Log Transformation Example

- Log functions are particularly useful when the input grey level values may have an extremely large range of values
- In the following example: the Fourier transform of an image is put through a log transform to reveal more detail



Magnitude of Fourier Spectrum



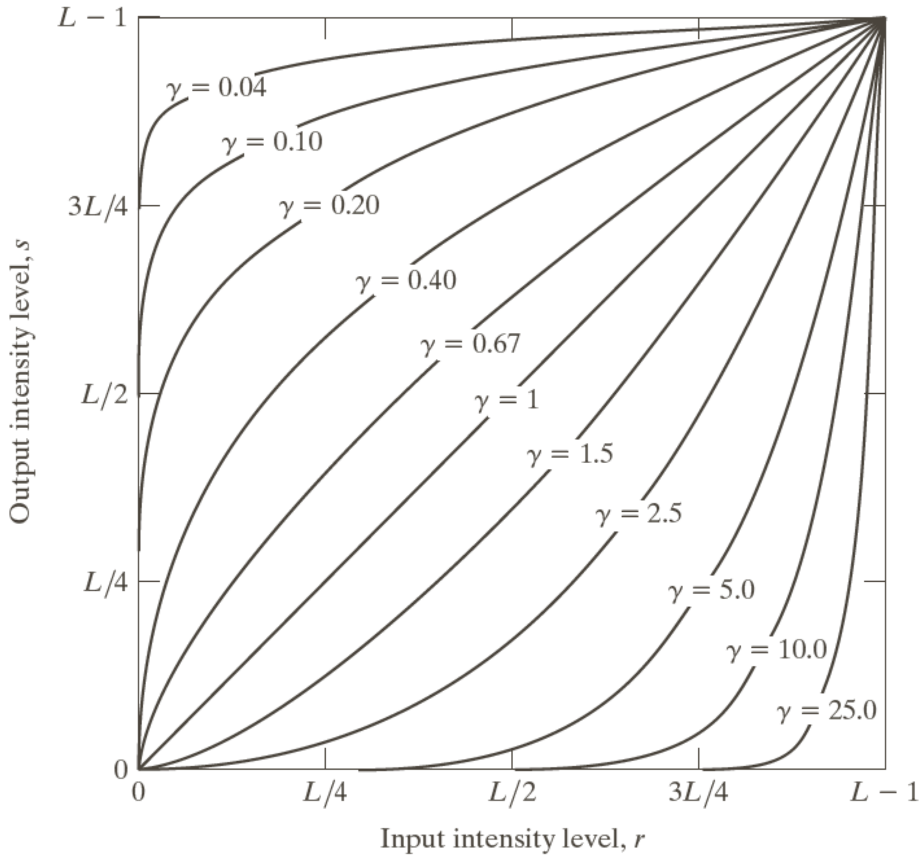
Log-transform of Fourier Spectrum Magnitude

Power-Law (Gamma) Transformations

- Power law transformations:

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

- Varying γ gives a whole family of curves
  - Map a narrow range of dark input values into a wider range of output values (γ <1)
  - Map a large range of dark input values into a smaller range of output values (γ >1)



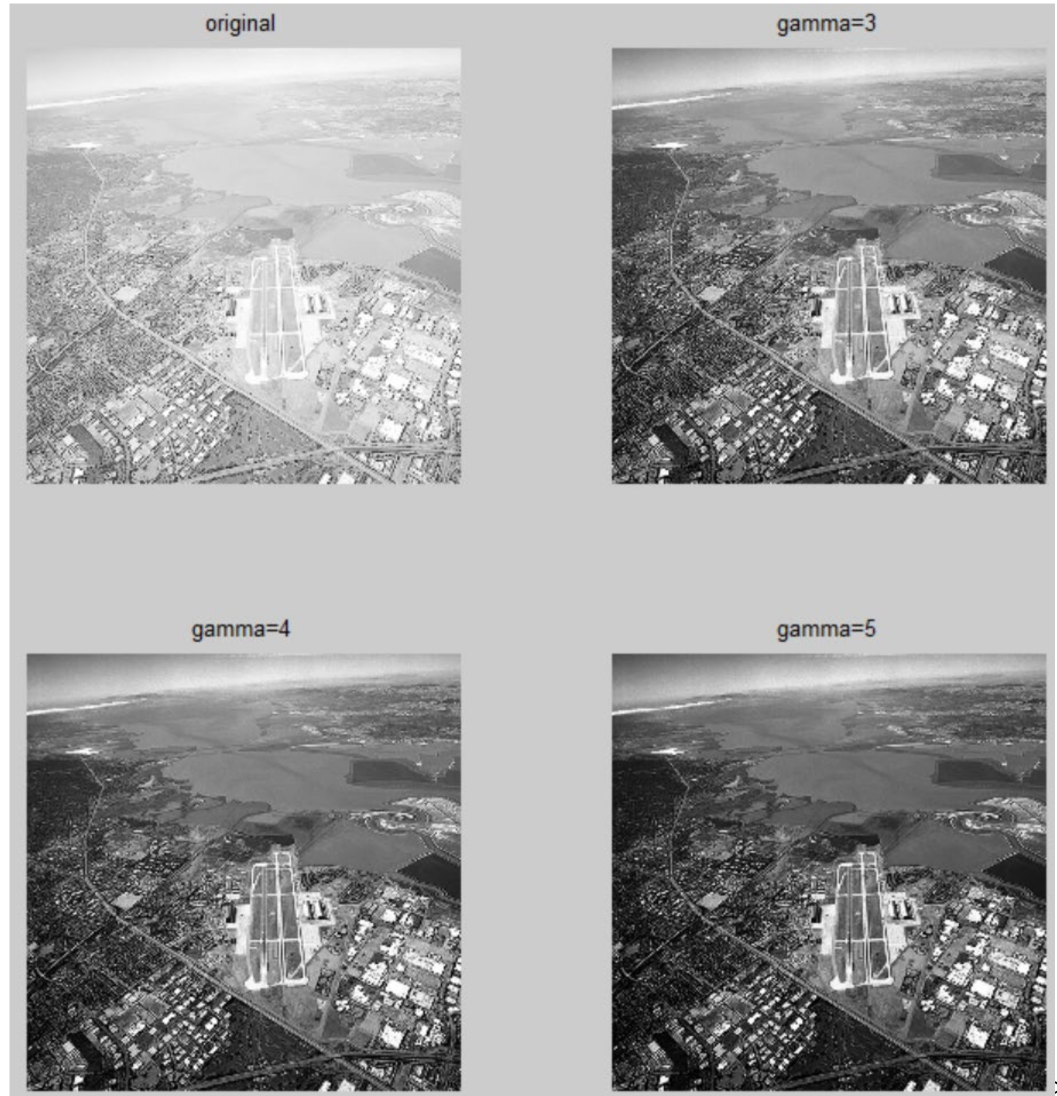
Power-Law Examples

- The original image shows a magnetic resonance (MR) image of a fractured human spine
- **Problem**: picture is too dark
- **Solution**: expansion of lower levels is desirable, $\gamma < 1$
- Different curves highlight different details

```
%% gamar transform with gamma<1 %2A
I=imread('Fig0308(a)(fractured_spine).tif');
gamma1=0.6;
gamma2=0.3;
gamma3=0.4;

I_new1=imadjust(I,[],[],gamma1);% help imadjust to fir
I_new2=imadjust(I,[],[],gamma2);
I_new3=imadjust(I,[],[],gamma3);

figure;
subplot(2,2,1);imshow(I);title('original');
subplot(2,2,2);imshow(I_new1);title('gamma=0.6');
subplot(2,2,3);imshow(I_new2);title('gamma=0.3');
subplot(2,2,4);imshow(I_new3);title('gamma=0.4');
```

Power-Law Examples

- An aerial photo of a runway is shown on right
- **Problem**: Image has "washout" appearance
- **Solution**: Compression of higher gray levels is desirable, $\gamma > 1$



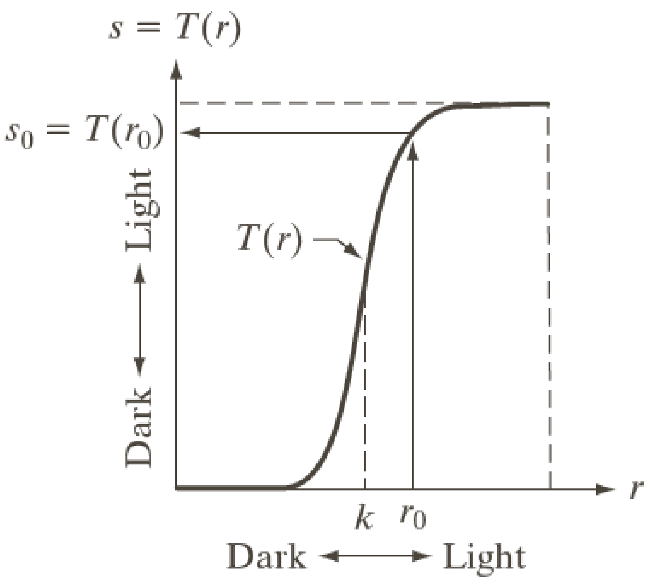
```matlab
%% gamar transform with gamma>1 %2A
clc
clear
I=imread('aerial.tif');

gamma1=3;
gamma2=4;
gamma3=5;

I_new1=imadjust(I,[],[],3);
I_new2=imadjust(I,[],[],4);
I_new3=imadjust(I,[],[],5);


figure;
subplot(2,2,1);imshow(I);title('original');
subplot(2,2,2);imshow(I_new1);title('gamma=3');
subplot(2,2,3);imshow(I_new2);title('gamma=4');
subplot(2,2,4);imshow(I_new3);title('gamma=5');
```

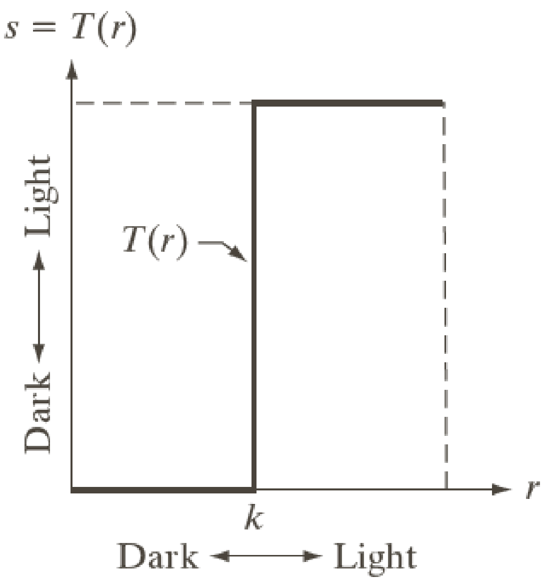
Piecewise-Linear Transformation

- Rather than using a well defined mathematical function, we can use arbitrary user-defined transforms
- Contrast Stretching
  - Expand the range of intensity levels in an image so that it spans the full intensity range of the recording medium or display device.
- Intensity-level Slicing
  - Highlighting a specific range of intensities in an image

Contrast Stretching

- Intensity transformation function: s=T(r)
- Produce higher contrast than the original by
  - Darkening the levels below *k* in the original image
  - Brightening the levels above *k* in the original
  - Thresholding: produce a binary image

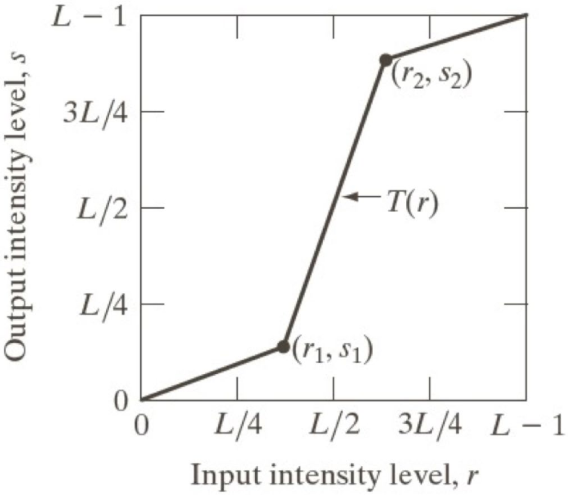




Contrast Stretching Examples

- Problem:
  - Low contrast image
  - result of poor illumination
  - lack of dynamic range
- Solution: Contrast stretching using the given transformation function

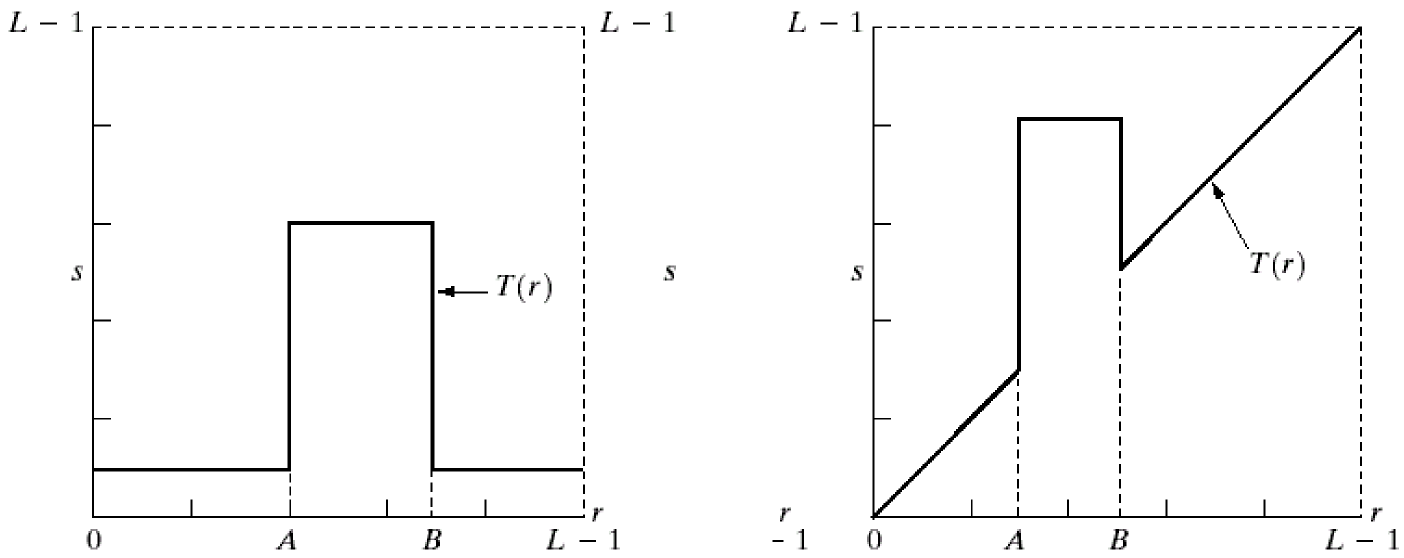
Low contrast image



Contrast Stretched image with piecewise linear transform

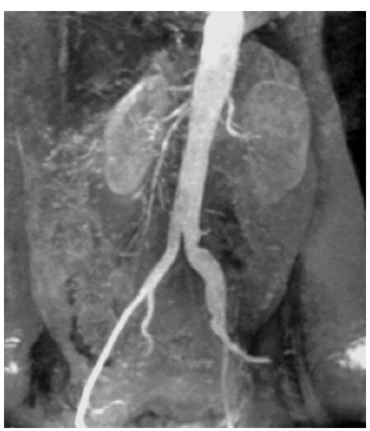
Intensity-Level Slicing

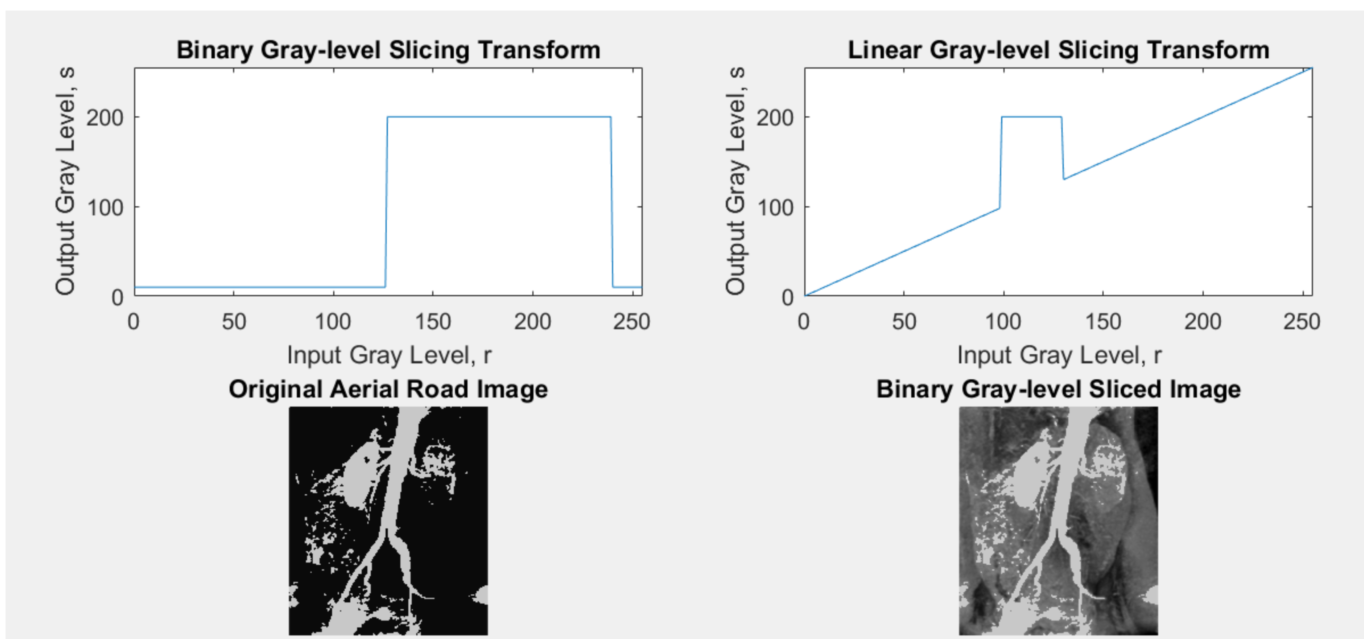
- Highlights a specific range of grey levels
  - Display high value for gray levels in the range of interest and low value for all other gray levels
  - Useful for highlighting features in an image



Intensity-Level Slicing Examples

- Binary gray level slicing transform
  - Highlights range [A,B] and reduces all others to a constant level
- Linear gray level slicing transform
  - Highlights range [A,B] but preserves all other levels





Histogram

- What is Histogram?
- Histogram Equalization

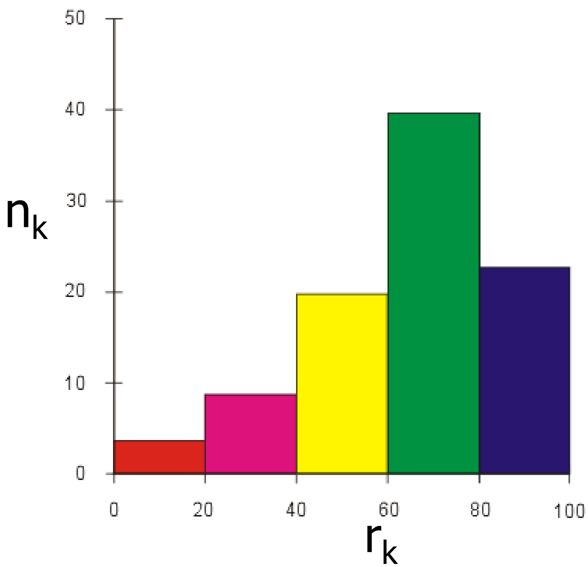
Image Histograms

- Given an image with gray levels from 0 to L-1, the histogram of the image is a representation of the frequency of occurrence of each gray level in the image
- Histogram $h(r_k) = n_k$
  - r_k is the kth intensity value
  - n_k is the number of pixels in the image with intensity r_k
- Probability Density Function (pdf): Normalized histogram

$$p(r)=n_k/(MN)$$

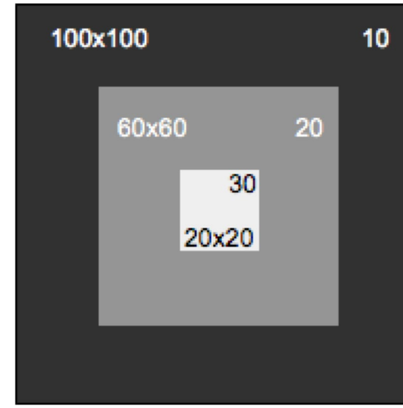
- Cumulative Density Function (cdf)

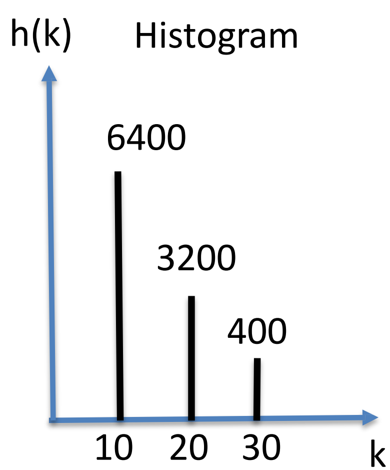
$$P_r(r_k) = \sum_{j=0}^{k} p_r(r_j)$$

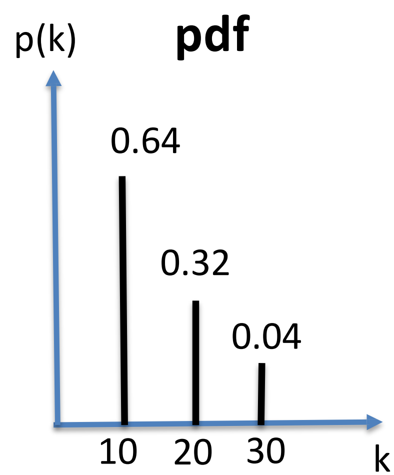


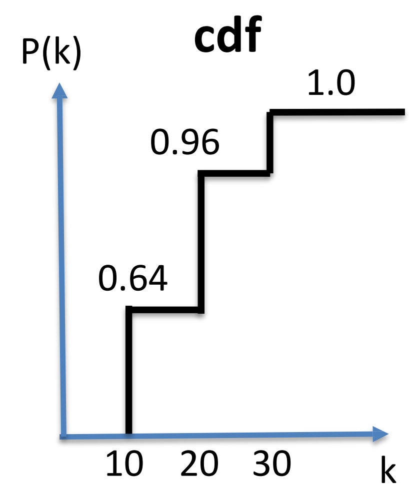
Example

- Find histogram
- Find Probability Density Function (pdf)
- Find Cumulative Density Function (cdf)









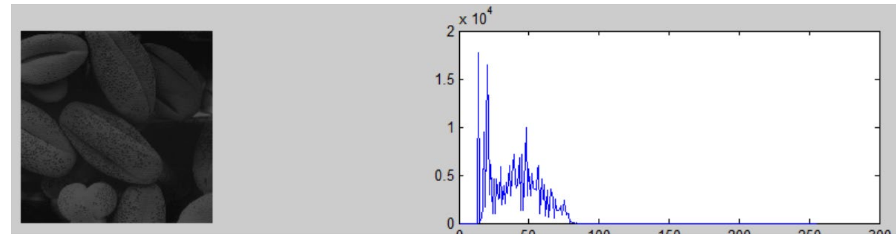
P(r1)=p(r1)=0.64

P(r2)=p(r1)+p(r2)=0.64+0.32=0.96

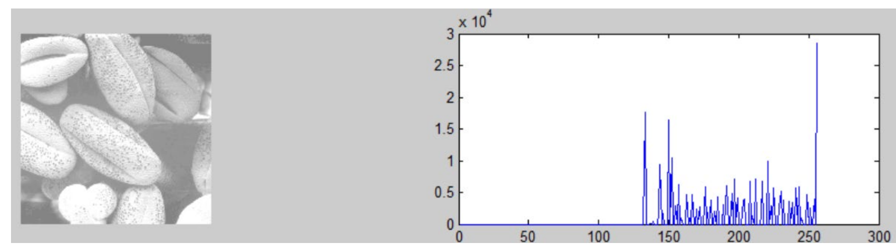
P(r3)=p(r1)+p(r2)+p(r3)=1

Histogram Examples

- Dark Image
  - Components of histogram are concentrated on low side of gray scale

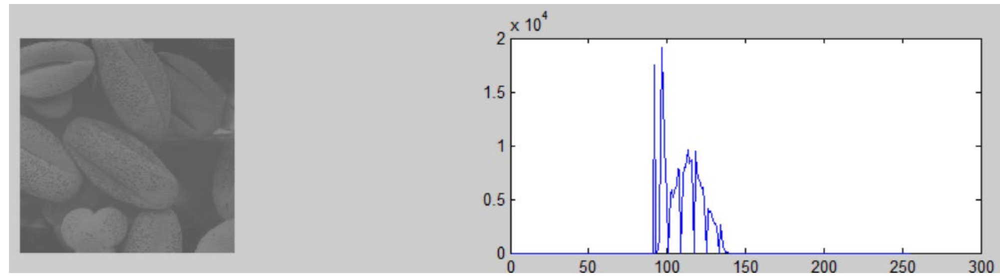


- Bright image
  - Components of histogram are concentrated on the high side of the gray scale

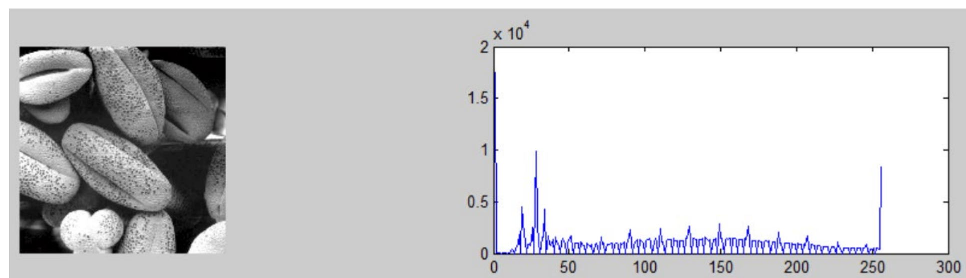


Histogram Examples

- Low-contrast Image
  - Histogram is narrow and centred towards the middle of the gray scale

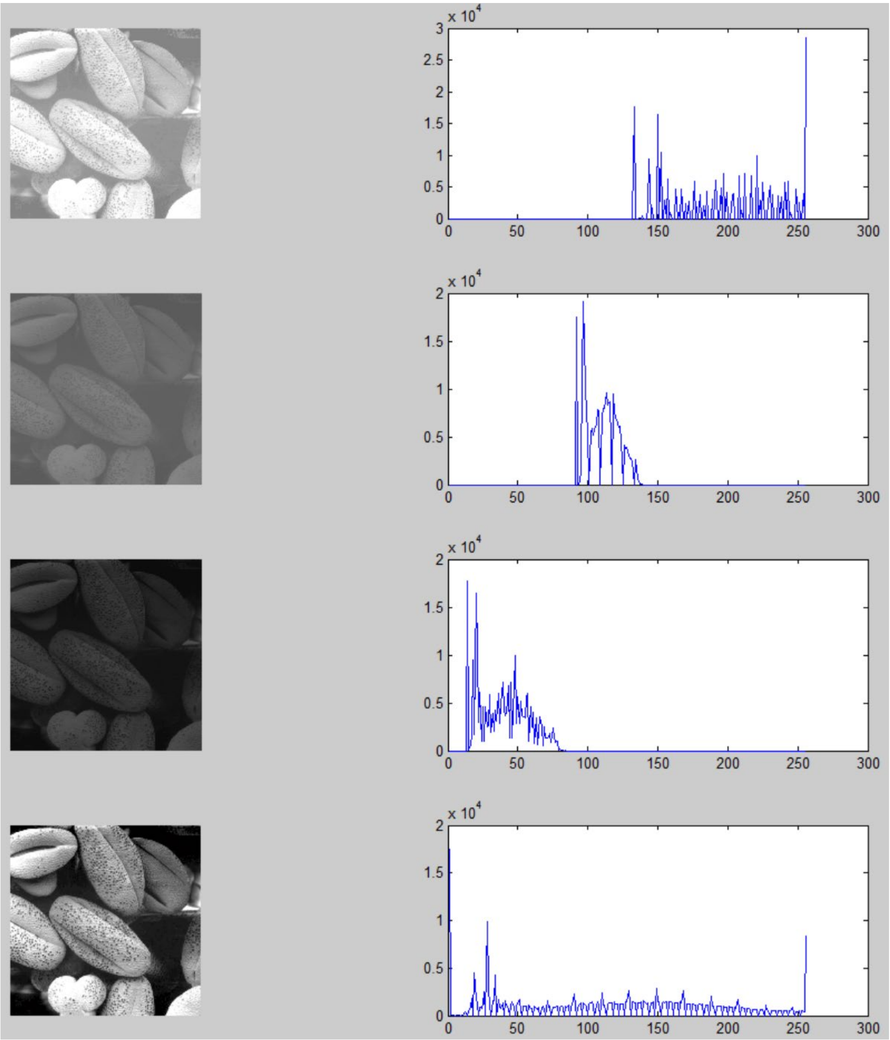


- High-contrast Image
  - Histogram covers a broad range of the gray scale
  - The distribution of pixels is not too far from uniform, with very few vertical lines being much higher than others



Histogram Examples

- Images and their histograms
- Notice the relationships between the images and their histograms
- Note that the high contrast image has the most evenly spaced histogram



```matlab
%% Plot image histogram %2A_PP30
I1=imread('hist1.tif');
I2=imread('hist2.tif');
I3=imread('hist3.tif');
I4=imread('hist4.tif');

figure;
subplot(4,2,1);imshow(I1);
subplot(4,2,2);plot(imhist(I1));
subplot(4,2,3);imshow(I2);
subplot(4,2,4);plot(imhist(I2));

subplot(4,2,5);imshow(I3);
subplot(4,2,6);plot(imhist(I3));
subplot(4,2,7);imshow(I4);
subplot(4,2,8);plot(imhist(I4));
```

Histogram Equalization

- Histogram EQUALization
  - Aim: To “equalize” the histogram, to “flatten”, “distrubute as uniform as possible”
- As the low-contrast image's histogram is narrow and centered towards the middle of the gray scale, by distributing the histogram to a wider range will improve the quality of the image
- Adjust probability density function of the original histogram so that the probabilities spread equally

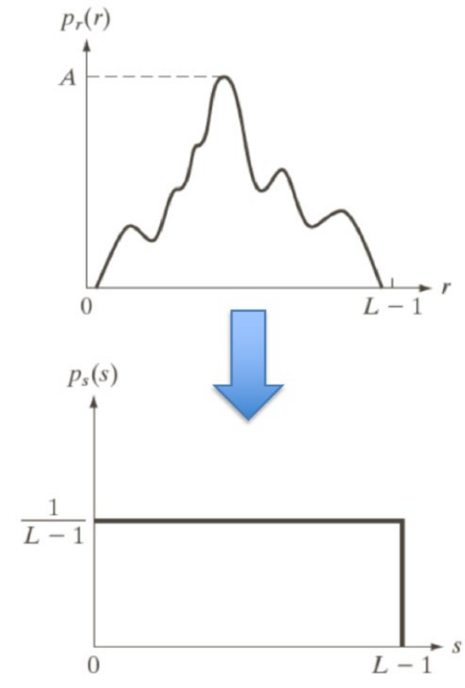
Histogram Equalization Process

- Objective: Obtain a flat histogram
- Let $p_r(r)$ and $p_s(s)$ denote the probability density function (PDF) of random variables r and s

$$p_r(r_k) = \frac{n_k}{MN} \qquad 0 \leq r \leq L\text{-}1.$$

$$\sum_{k=0}^{L-1} p_r(r_k) = 1$$

$$p_s(s) = \frac{1}{L-1} \qquad 0 \leq s \leq L\text{-}1.$$



Doing Histogram Equalization by Hand

- Procedure

Get histogram of *MxN* input image $H_r(r) = n_r$. Gray levels range from 0..*L*-1.

Determine Probability Density Function (PDF)

$$p_r(r_k) = \frac{n_k}{MN}$$

Determine Cumulative Probability Distribution (CDF)

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

Scale *T(r)* to desired range of output gray levels

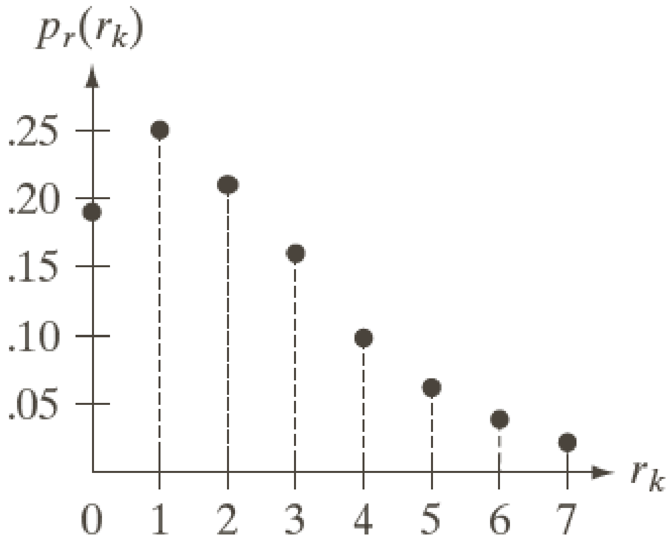
$$T(r) = (L-1)P_r(r)$$

Apply the transformation *s = T(r)* to compute the output values

Example: Histogram Equalization

- Suppose that a 3-bit image (L=8) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in following table.
- Get the histogram equalization transformation function and give the $p_s(s_k)$ for each s_k.

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



Example: Histogram Equalization

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

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7 \times 0.19 = 1.33 \quad \rightarrow 1$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7 \times (0.19 + 0.25) = 3.08 \quad \rightarrow 3$$

$$s_2 = 4.55 \quad \rightarrow 5 \qquad s_3 = 5.67 \quad \rightarrow 6$$

$$s_4 = 6.23 \quad \rightarrow 6 \qquad s_5 = 6.65 \quad \rightarrow 7$$

$$s_6 = 6.86 \quad \rightarrow 7 \qquad s_7 = 7.00 \quad \rightarrow 7$$

Example: Discrete Histogram Equalization (3)

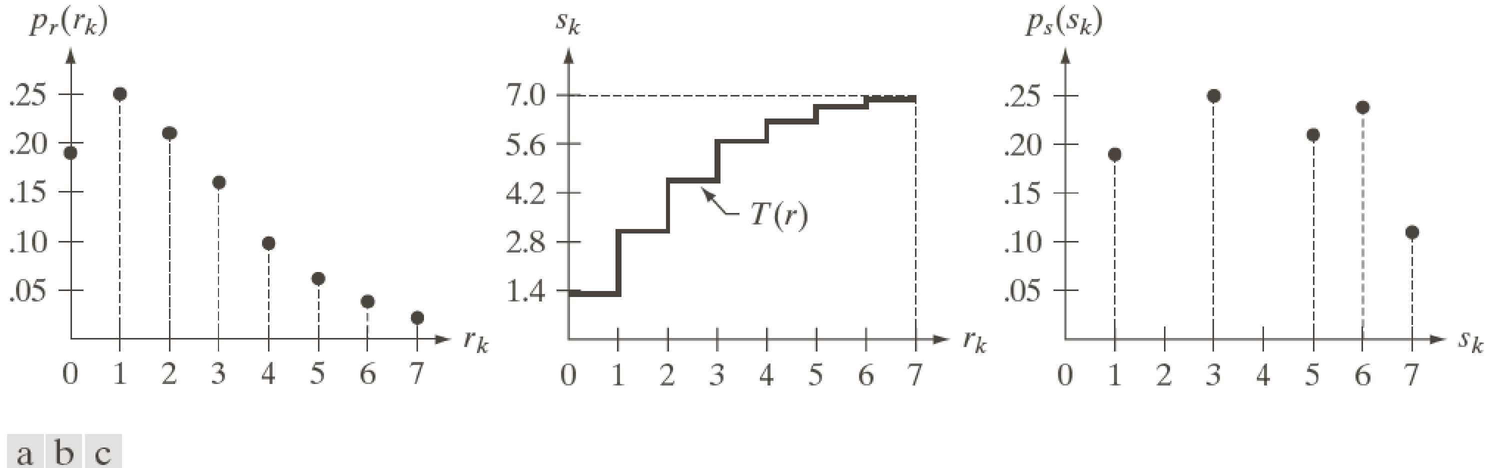


FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Numeric Example of Equalization

0	1	3	4
1	2	2	3
1	3	4	4
3	2	5	2

(a)

i	$\hat{h}_i$	$\hat{C}_i$	$7\hat{C}_i$
0	1/16	1/16	0
1	3/16	4/16	2
2	4/16	8/16	4
3	4/16	12/16	5
4	3/16	15/16	7
5	1/16	16/16	7
6	0/16	16/16	7
7	0/16	16/16	7

(b)

0	2	5	7
2	4	4	5
2	5	7	7
5	4	7	4

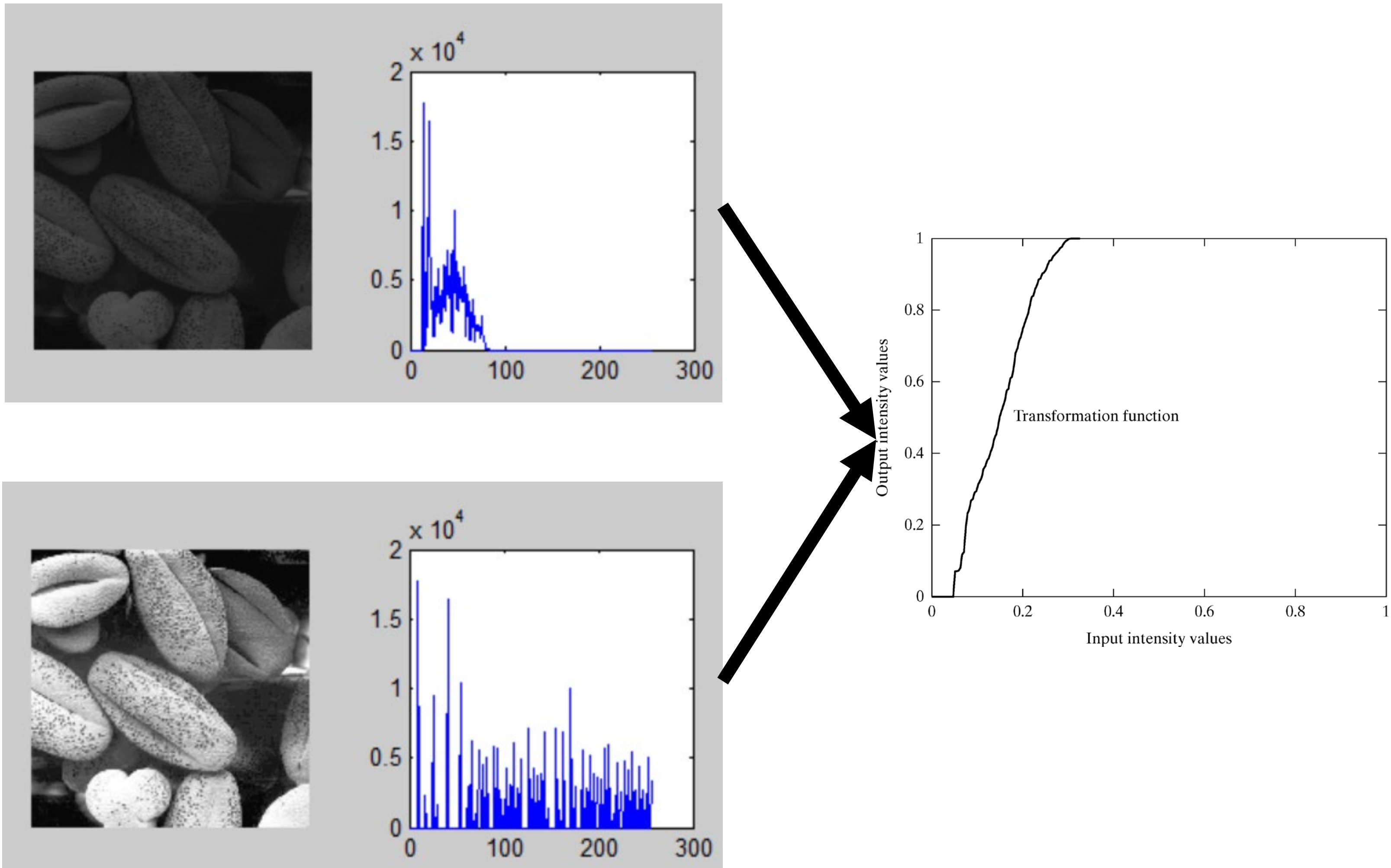
(c)

i	$\hat{h}_i$
0	1/16
1	0/16
2	3/16
3	0/16
4	4/16
5	4/16
6	0/16
7	4/16

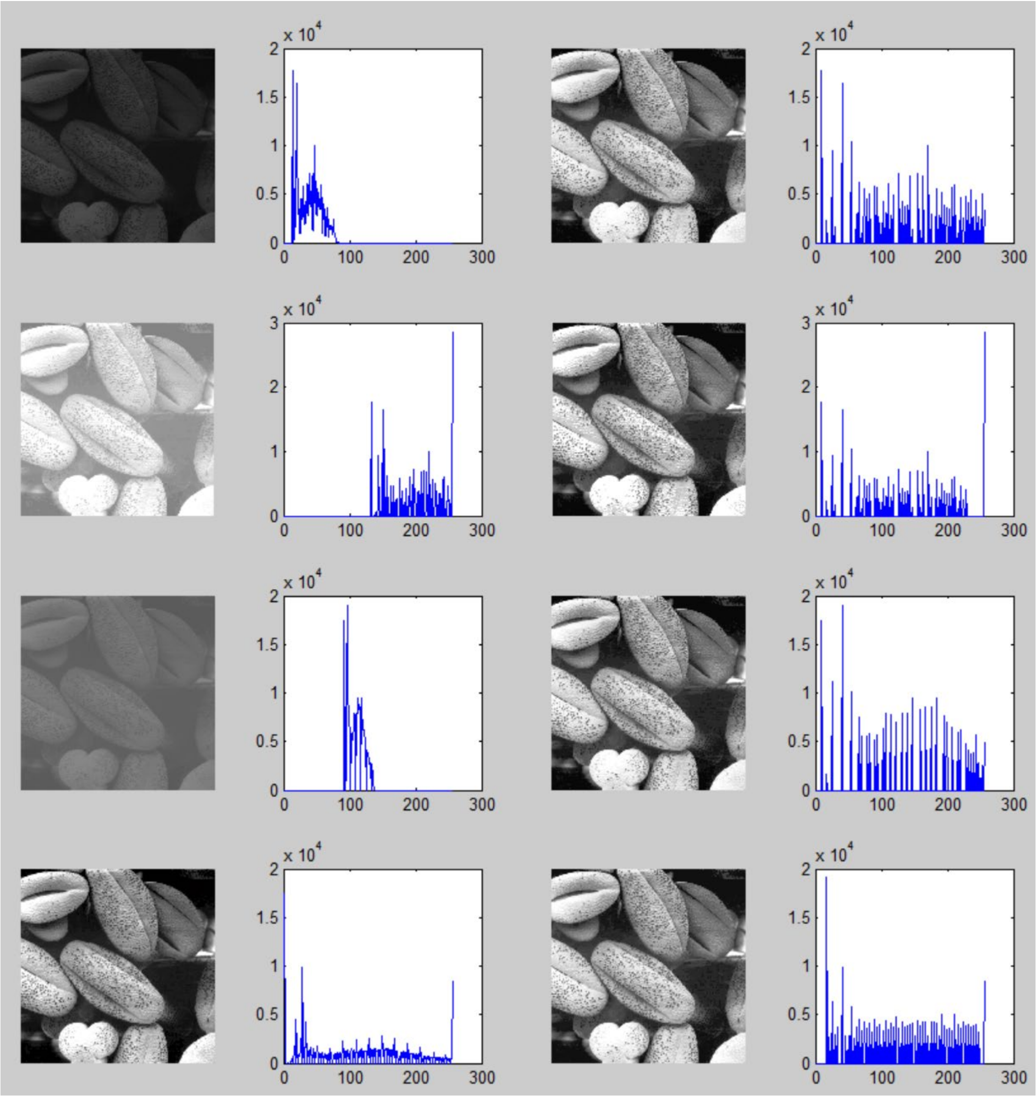
(d)

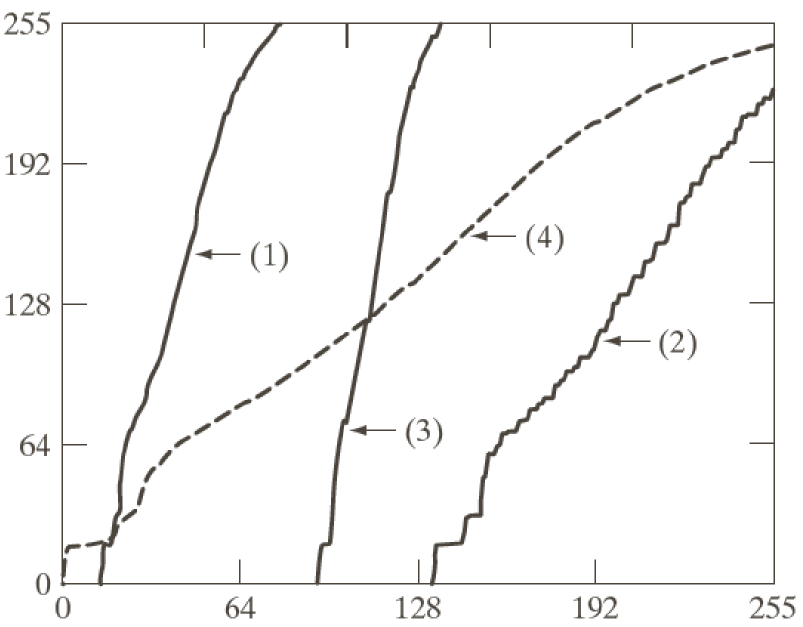
Figure 5.9. Numerical example of histogram equalization: (a) a 3-bit image, (b) normalized histogram and CDF, (c) the equalized image, and (d) histogram of the result.

Equalization Transformation Function



Histogram Equalization Examples





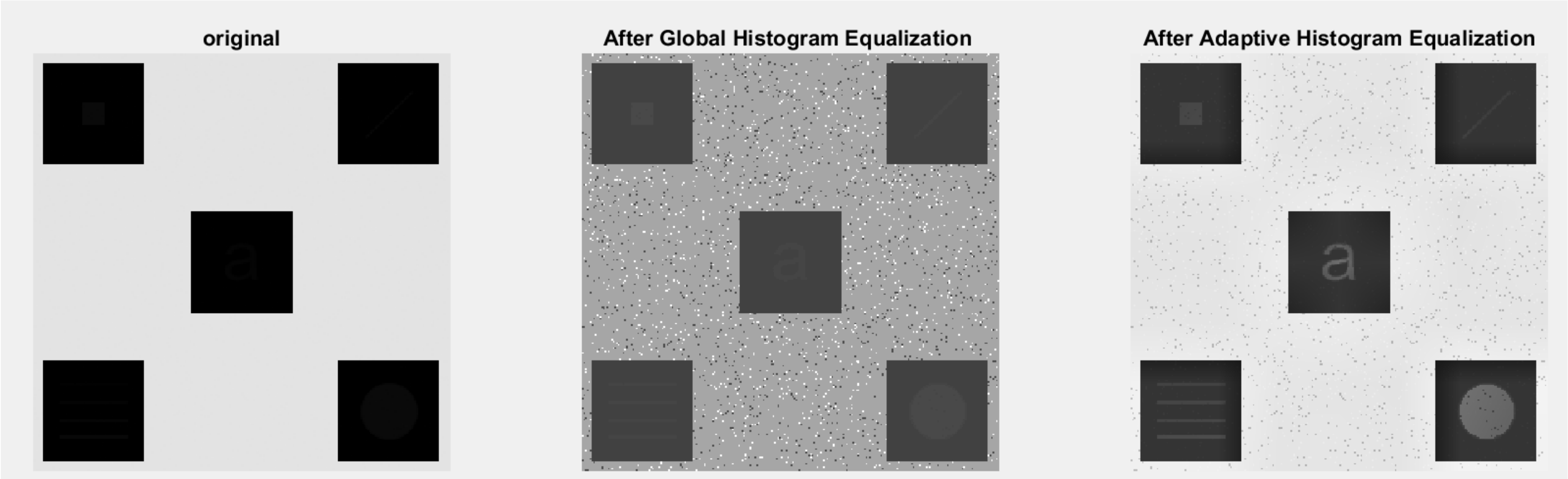
Local Histogram Equalization

- Histogram equalization can be performed on a “local” level.
  - Compute the ***histogram and CDF*** of a ***local*** region about each pixel and then use that CDF as a lookup table for ***that pixel alone***.
  - Has the (possibly negative) effect of eliminating global contrast

Original Image Equalized Image Locally Equalized Image

Local Histogram Equalization

- Dealing with things locally




```
%% local hist equalization %2A_PP42
img=imread('square.tif');
clipLimit = 0.2;
numTiles = [7 7];
I_new=histeq(img);
imgLocHistEq = adapthisteq(img, 'ClipLimit', clipLimit, 'NumTiles', numTiles);
figure;
subplot(1,3,1); imshow(img); title('original');
subplot(1,3,2); imshow(I_new); title('After Global Histogram Equalization ');
subplot(1,3,3); imshow(imgLocHistEq); title('After Adaptive Histogram Equalization');
```