Notice

Projects and Homework: please see course website:

http://tel.scholat.com/

或

http://www.scholat.com/

Chapter 3
Intensity Transformations and Spatial Filtering

第三章：灰度变换和空间滤波

浮雕

素描

1、将相机对着风景，放在三脚架上；

2、设置相机为每10秒拍一张，共拍15张左右；

3、用Photoshop的“文件(File)>脚本(Scripts)>数值(Statistics)”的“中间值(median)”打开这15张图片；

4、神器Photoshop将会找出这些图片的不同，然后轻松的去掉他们！给你一张纯风景！

1、将相机对着风景，放在三脚架上；
2、设置相机为每10秒拍一张，共拍15张左右；
3、用Photoshop的“文件(File)>脚本(Scripts)>数值(Statistics)”的“中间值(median)”打开这15张图片；
4、神器Photoshop将会找出这些图片的不同，然后轻松的去掉他们！给你一张纯风景！

Question：

- **Why?**
- **In what condition it works?**

Chapter 3
Image Enhancement in the Spatial Domain

It makes all the difference whether one sees darkness through the light or brightness through the shadows.

——*David Lindsay*

致谢: 本课件部分页面和图片来自台湾彭明辉教授的教学课件，特此表示感谢。

概 述

- 图像增强一直是图像处理中最具吸引力的领域
- 目的是提高图像在特定应用领域的视觉质量*
- 图象增强包括光滑、锐化、提取边缘、反转、去噪以及各种滤波等等处理。目的是经过处理后的图像更适合特定的应用（主观的观察分析和评价）
- 没有通用的理论和方法，主观评价为主
- 仍然有很多待解决的问题和新问题（如超分辨重建等）
- 共有两大类算法：空间域和频率域。
- 难点在于算法和图像视觉的对应

Preview

3.1 Background

3.2 Basic Grey Level Transformation

3.3 Histogram Processing

3.* Enhancement using Arithmetic/Logic Operation

3.4 Fundamentals of Spatial Filtering

3.5 Smoothing Spatial Filters

3.6 Sharpening Spatial Filters

3.7 Combing Spatial Enhancement Methods

Image Processing —— Image Understanding ——Computer Vision

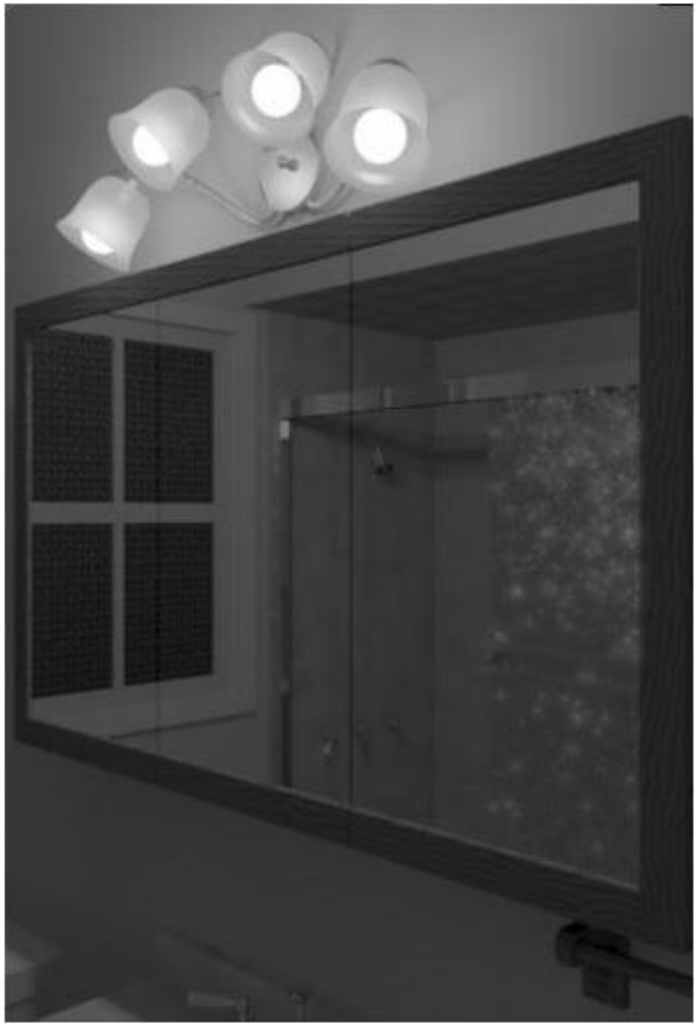
- **Low-level Processing: both input and output are images**
  - Noise Reduction
  - Image Enhancement
  - Image Sharpening
- **Mid-level Processing: input images, output attributes of those images**
  - Image Segmentation
  - Image Indexing (Feature Extraction)
- **High-level Processing: related to computer vision**
  - Image Analysis and Understanding

$f(x,y)$

T

$\rightarrow$

$g(x,y)=T\{f(x,y)\}$



3.1 Background (cont)

$f(x,y)$ $\qquad\qquad$ $g(x,y)=T\,f(x,y)$

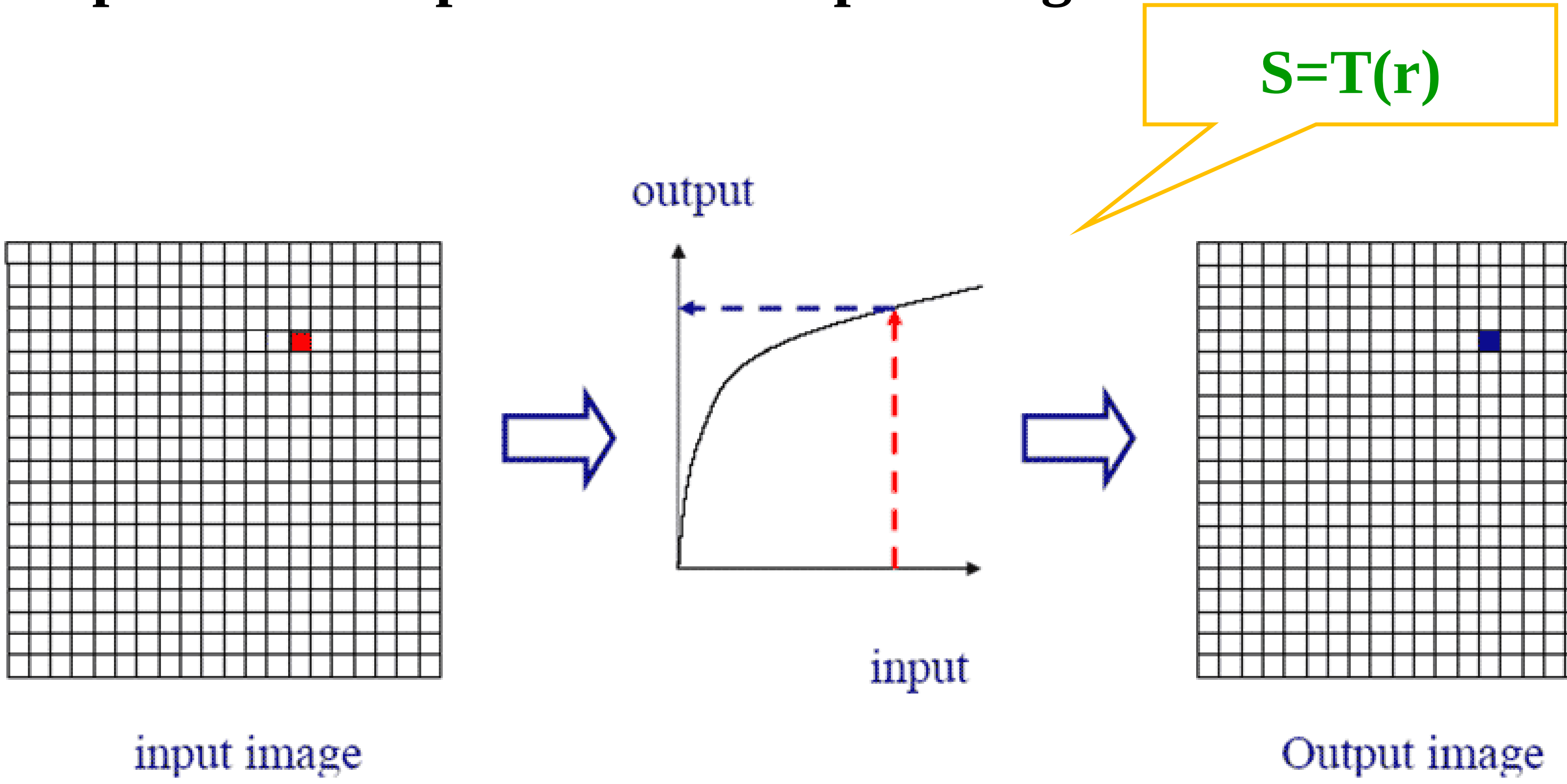
$\xrightarrow{T}$

显示器上显示一副数字图像需要哪些要素?

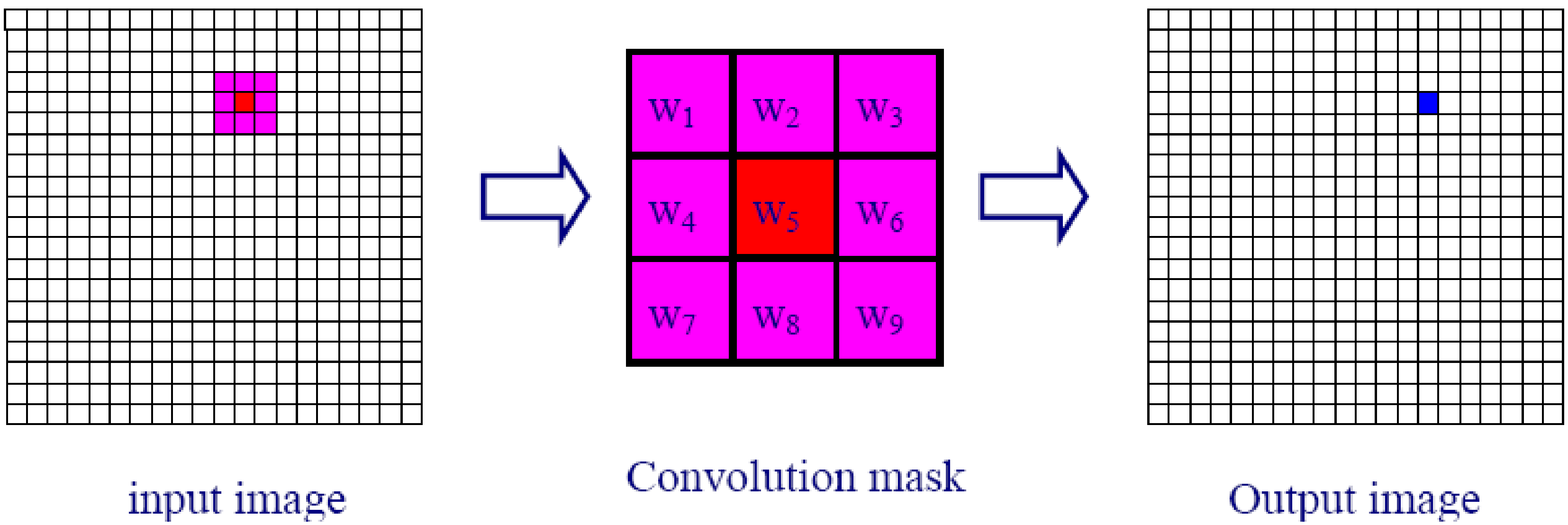
- 一个用矩阵表示的数字集合
- 数字和亮度的对应关系（规定好的显示方法）
- 最基础的数字图像处理就是根据规定好的显示方法，处理用矩阵表示的数字图像的像素值（灰度值）。

3.1 Background

- **Pixel-wise operation and convolution masks**
  - **Pixel-wise operation: one pixel in the original image to produce one pixel in the output image**



- **Convolution masks: It takes 3X3 (or more) pixels from the input image to produce one pixel in the output image**



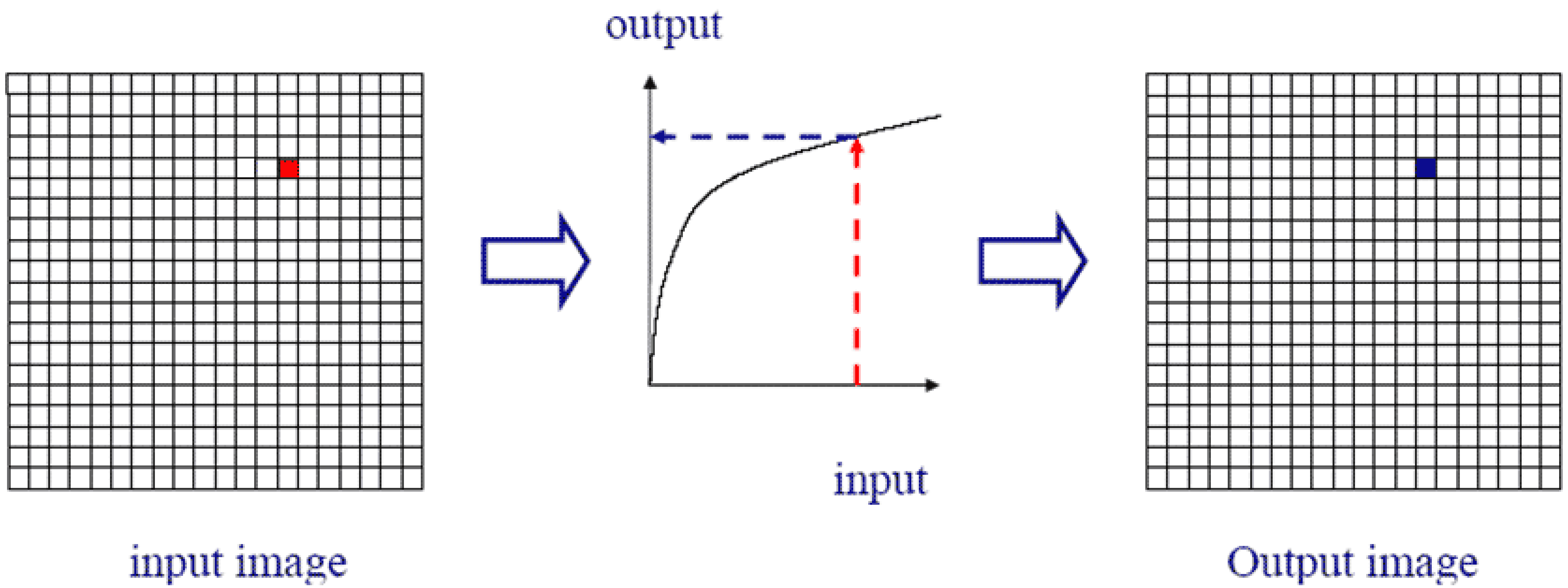
$$h(i,j) = w_1 p_1 + w_2 p_2 + w_3 p_3 + w_4 p_4 + w_5 p_5 + w_6 p_6 + w_7 p_7 + w_8 p_8 + w_9 p_9$$

3.2 Basic Grey Level Transformation

- **General formula**

r **— original (input) gray level,** ***s*** **—output**

$$s=T(r)$$



不同的函数*T*，给出不同的灰度变换方法，亦即图像增强方法。问题在于如何根据需要构造变换函数*T*

3.2 Basic Grey Level Transformation

●General formula

r **— original (input) gray level,** ***s*** **—output**

$$s=T(r)$$

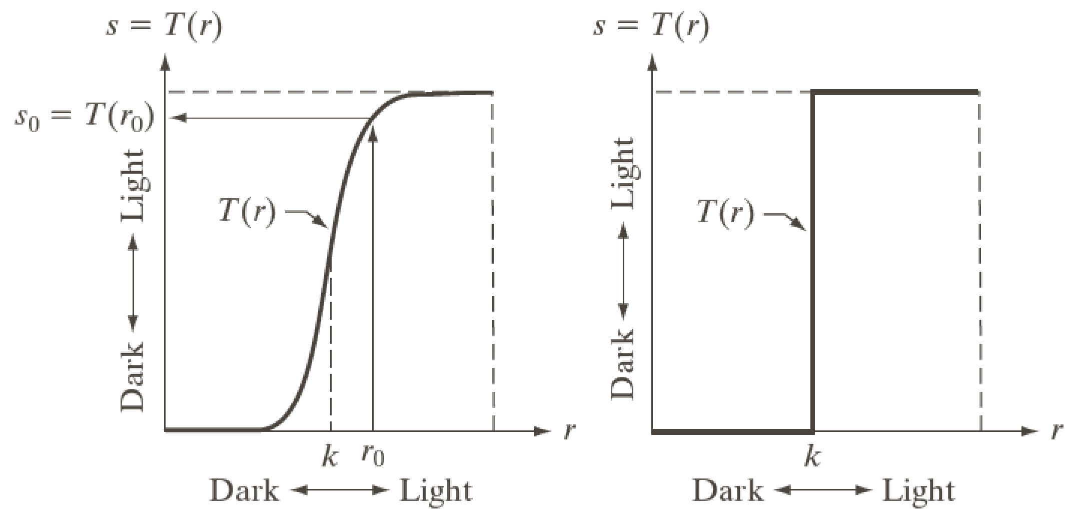
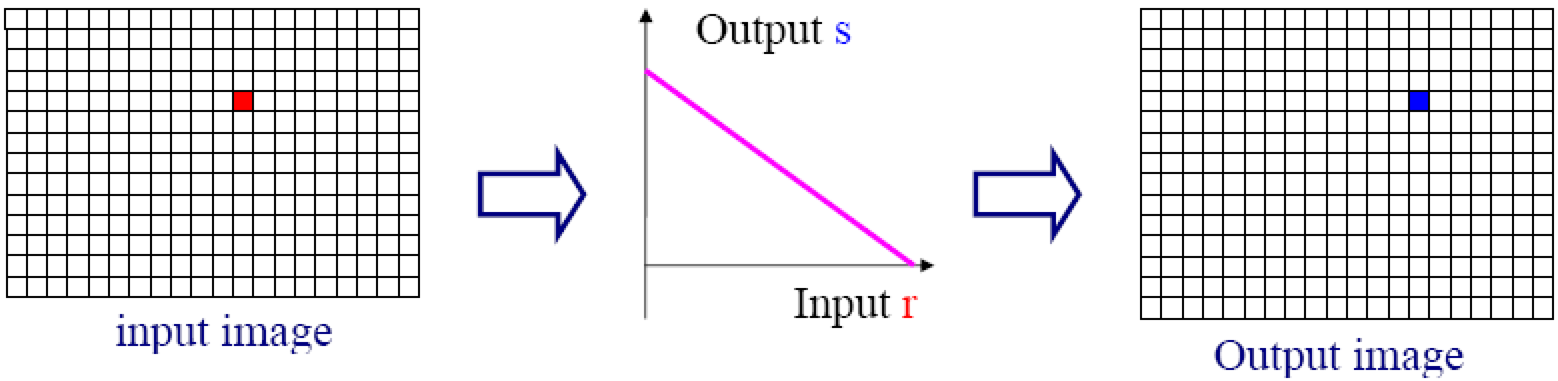


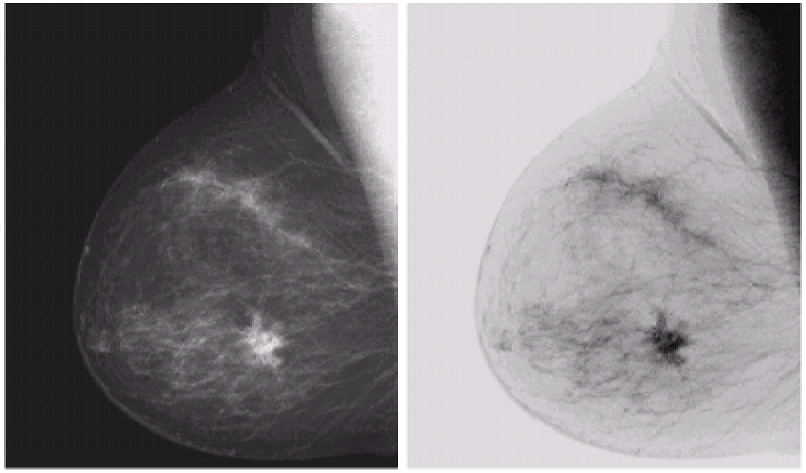
FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

不同的函数***T***，给出不同的灰度变换方法，亦即图像增强方法。问题在于如何根据需要构造变换函数***T***

3.2 Basic Grey Level Transformation

●Negative Image





The output negative image is produced by the following formula

$$\text{Output } s = 255 - \underset{\substack{\uparrow \\ \text{input}}}{r} = (L-1) - r$$

人眼的一个特点就是在背景相对光亮时对灰度层次有较好的分辨能力

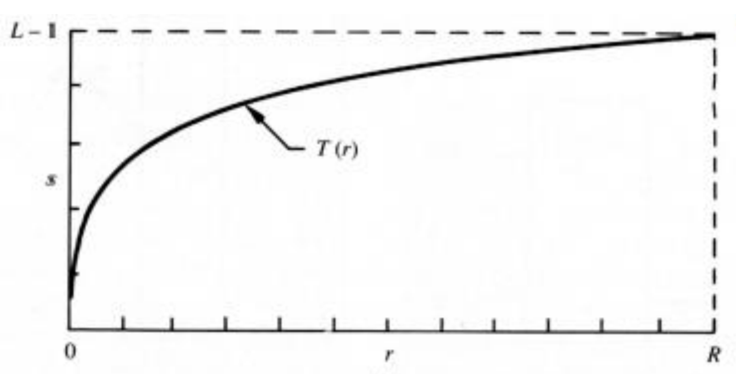
Log Transformations

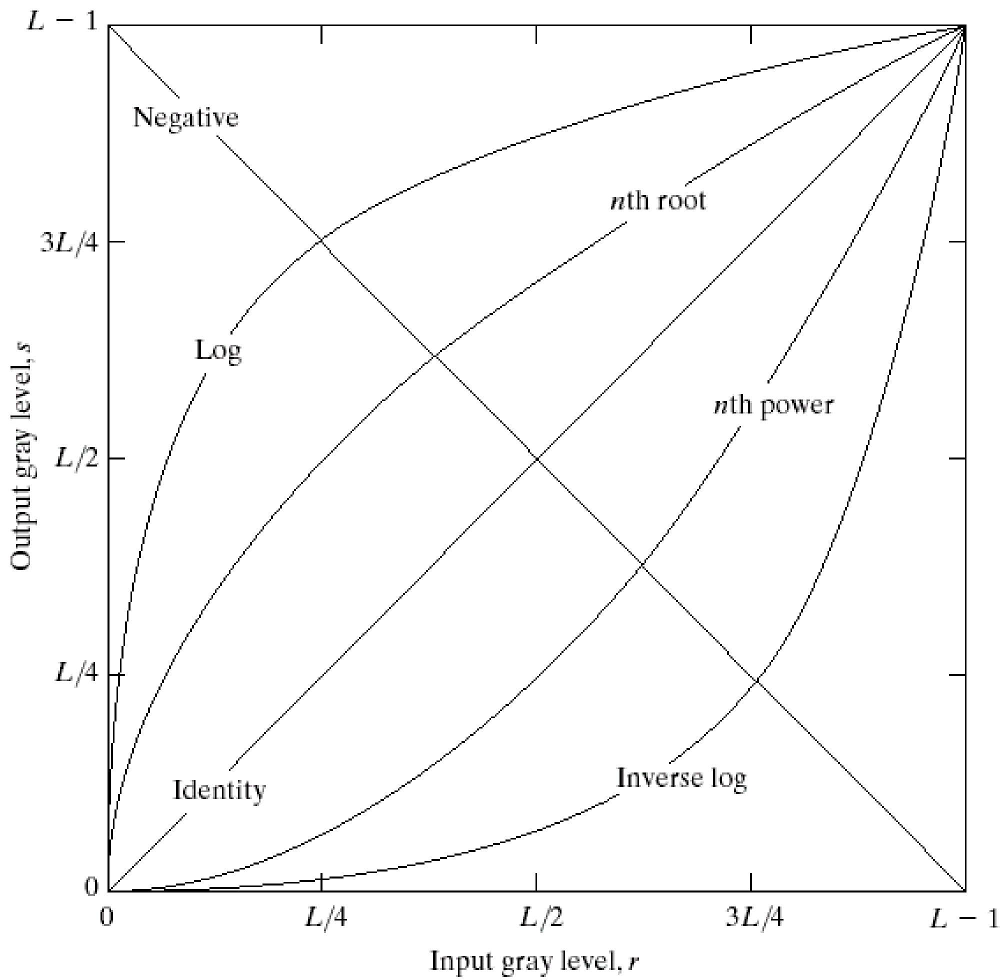
Figure 3.3

$$s = c \log(1+r)$$

Log transform is less popular than the power law discussed below. **Here *c* is a constant and** $r \geq 0$.

下列曲线更为合适：

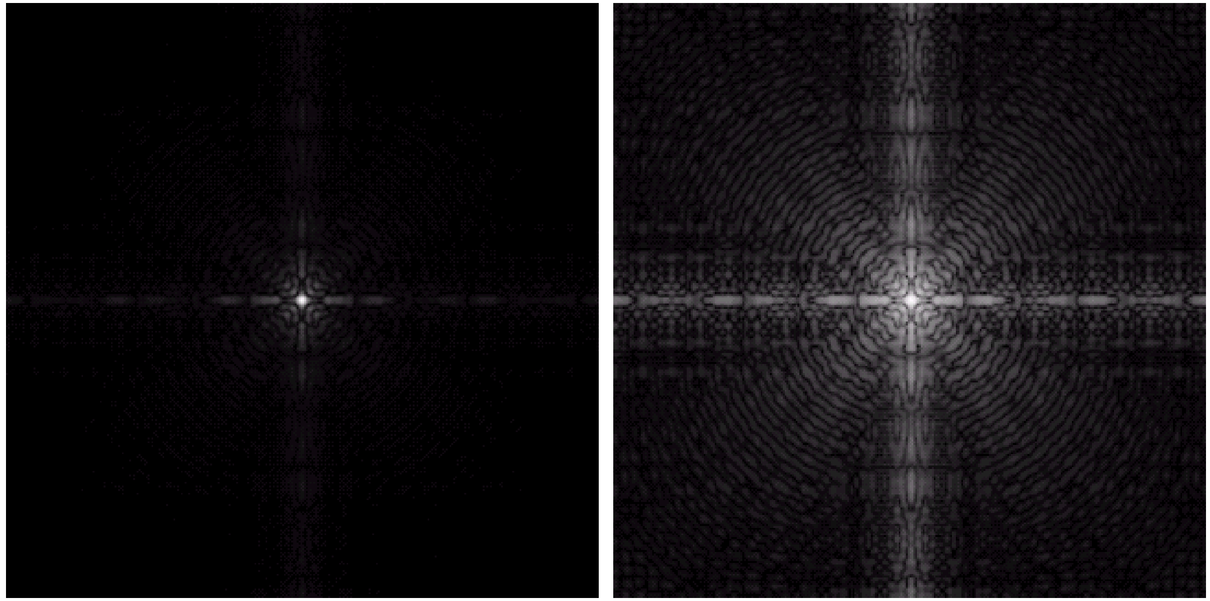




3.2 Some basic Gray level Transformation (cont.)

a b

FIGURE 3.5 (a) Fourier spectrum. (b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.



(a) Shows a Fourier spectrum with value in the range 0--1.5×10^6, which are scaled linearly (线性标定) for display in a 8-bit system.

(b) Applying Log transform to (a), the resulting range of values become 0 to 6.2. And then, scaling linearly.

关于图像显示及线形标定(scaling linearly)!!!

- **Power-law Transformation:** $s = cr^{\gamma}$

where both c and γ are positive constant.

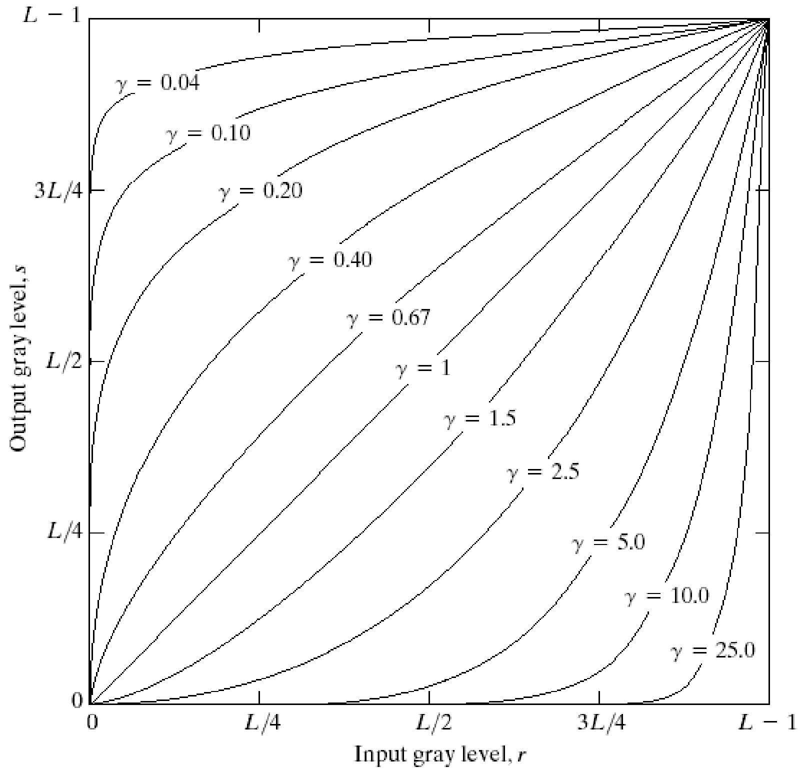


Figure 3.6 power-law

- **For $\gamma<1$, image in the dark area is enhanced.**
- **For $\gamma>1$, image in the bright area is enhanced.**

当幂指数 γ 变化时，得到一系列不同的曲线，对应于不同效果的变换。这是和对数变换不同和方便的地方。

伽玛校正：大量的图像设备如捕捉卡、打印机、数码相机以及显示装置的响应（输出）就对应一个幂函数，通常称这个幂函数的指数为“伽玛” （gamma） （不同厂家的设备这个指数也不同）。纠正这个幂次响应的处理称为伽玛校正（gamma correction）。

- 阴极射线管（CRT）装置中有一个电压-强度响应，这个响应就是指数γ变换范围在1.8-2.5的幂函数。

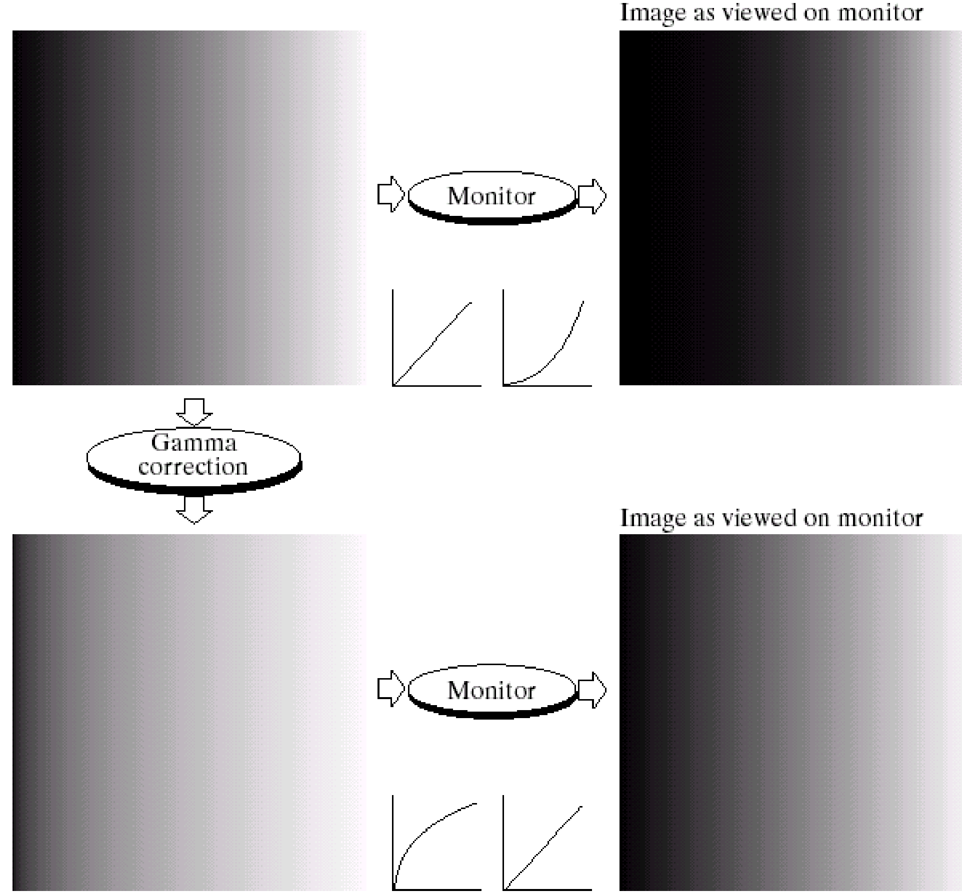


FIGURE 3.7 (a) Linear-wedge gray-scale image. (b) Response of monitor to linear wedge. (c) Gamma-corrected wedge. (d) Output of monitor.

- **Some key points for gamma correction**
- 伽马校正对在计算机屏幕上精确显示图像也很重要。
- 幂函数还可以用于调整图像对比度。
- 在一般的图像处理软件中，几乎都有伽玛校正的功能。这个功能可用于调整图像的对比度。如果图像偏暗，有些低灰度值的细节被掩盖时，可考虑用指数 $\gamma<1$ 的伽玛校正；反之， $\gamma>1$ 的校正对那些被“漂白”的细节会起作用。
- 对图像像素灰度的调整最好在象素值被量化到0—255的整数之前

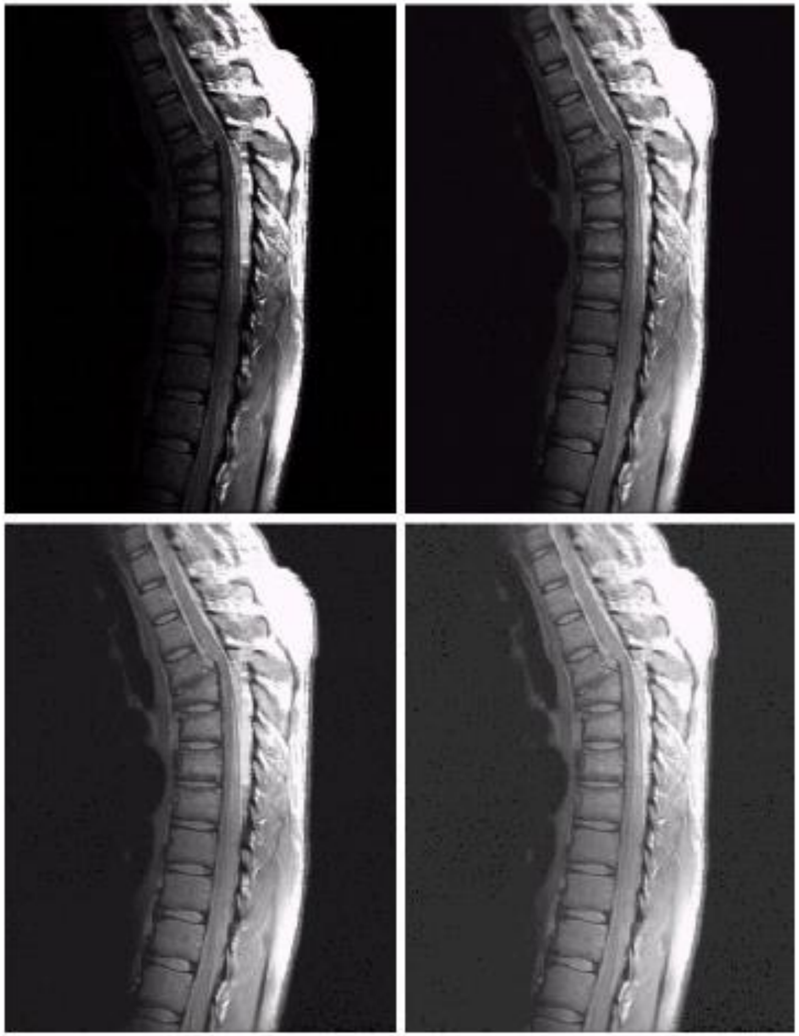


Figure 3.8 power-law

Image in the dark area is enhanced by the power law

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

with

$$\gamma = 0.6$$

$$\gamma = 0.4$$

$$\gamma = 0.3$$

这个例子和前一个的效果刚好相反，对应的伽马值分别为3.0，4.0和5.0

Image in the bright area is enhanced by the power law

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

with

$\gamma = 3.0$

$\gamma = 4.0$

$\gamma = 5.0$

Looking carefully at the shape of these transformation *curves*, what do we see? And conclusion?

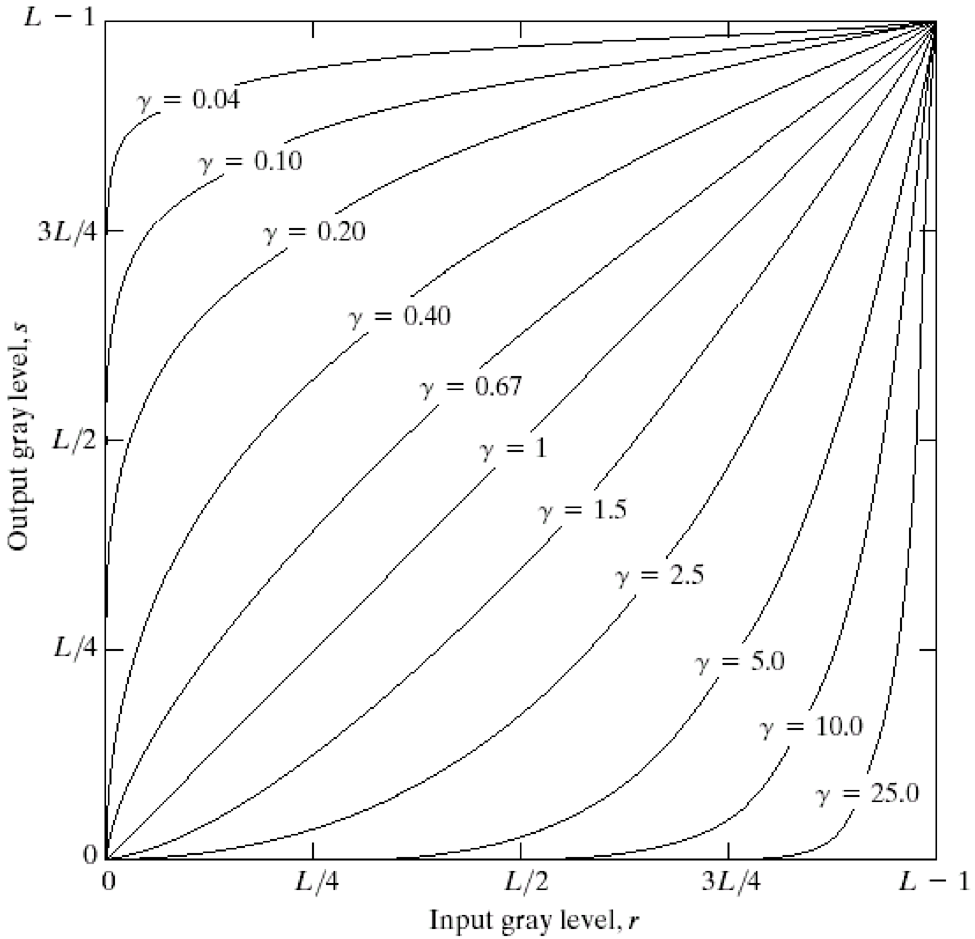


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

Basically, gray level transformation is

$$\begin{bmatrix} 0 \\ 1 \\ \vdots \\ L-1 \end{bmatrix} \xrightarrow{\quad T \quad} \begin{bmatrix} T(0) \\ T(1) \\ \vdots \\ T(L-1) \end{bmatrix} \xrightarrow{T=r^2} \begin{bmatrix} 0 \\ 1 \\ \vdots \\ (L-1)^2 \end{bmatrix}$$

不同的T给出不同的图像处理效果

- **Piece-wise linear transform**

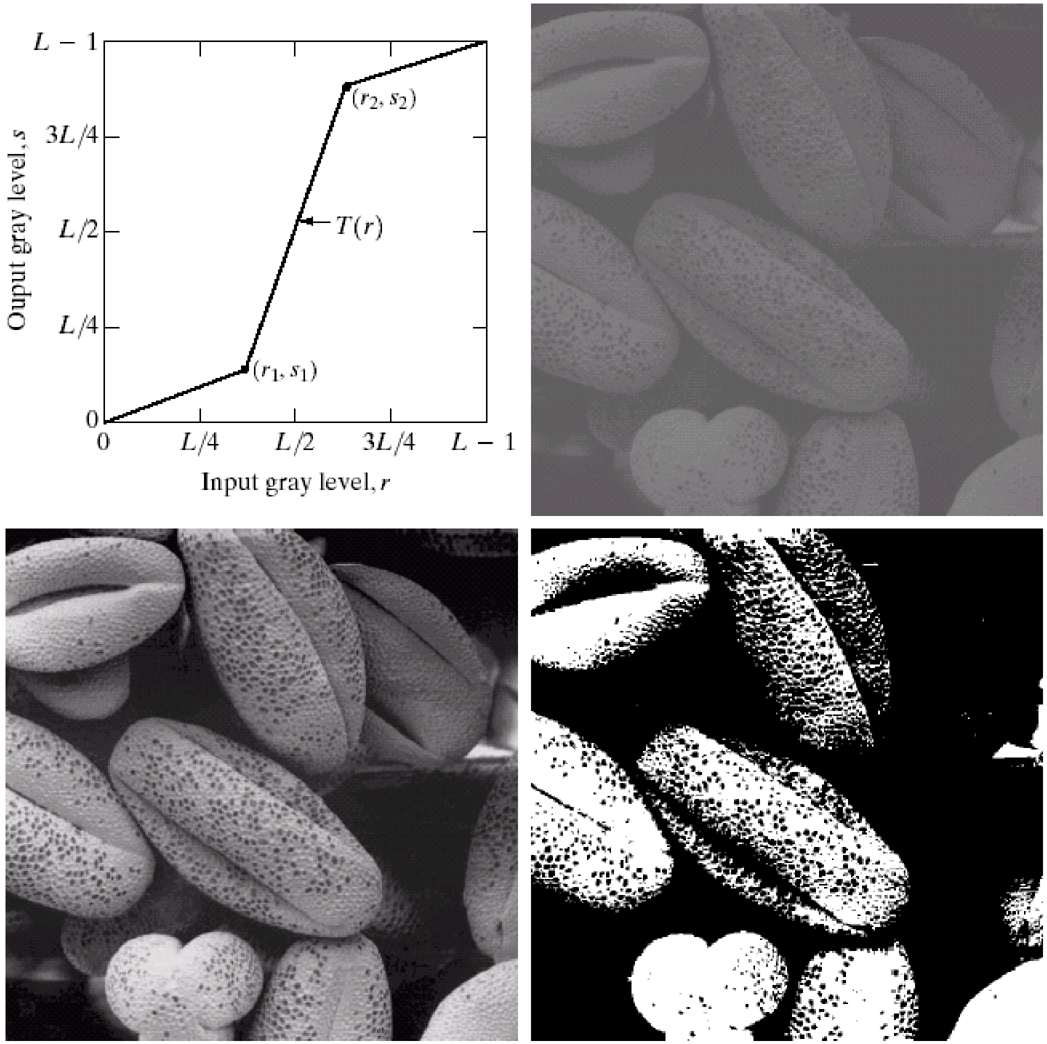


Figure 3.10 a b

c d

a. Bilinear transform

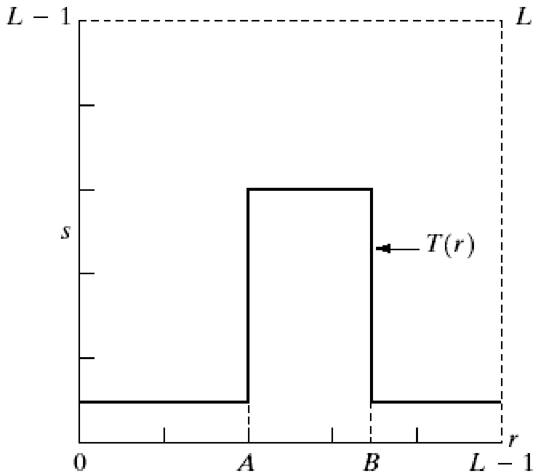
b. Original image (low contrast)

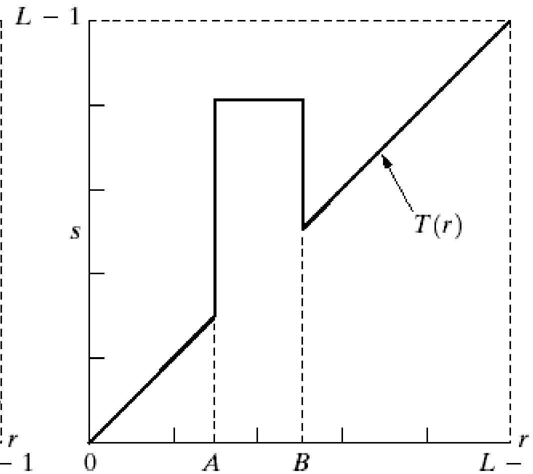
c. Output of bilinear transform

d. thresholding after bilinear transform

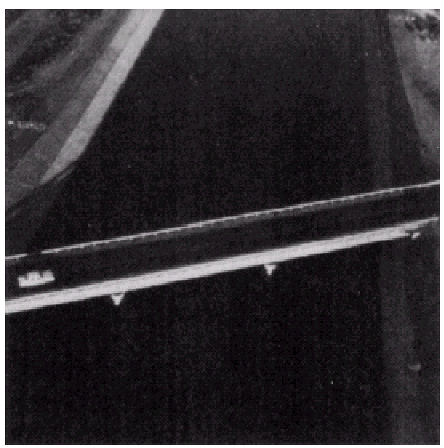
- **Grey level slicing：在图像中提高特定灰度的亮度**

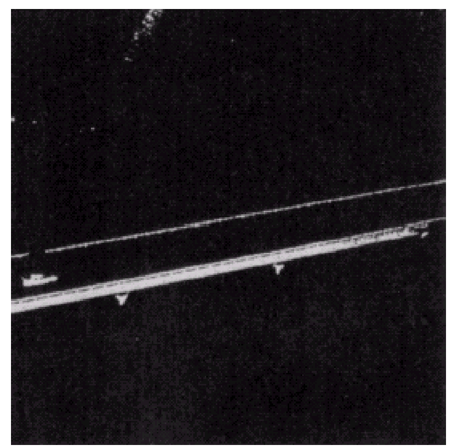
The left mapping (transform) The right mapping (transform) highlights brightness in the level [A, B] to results in a binary image





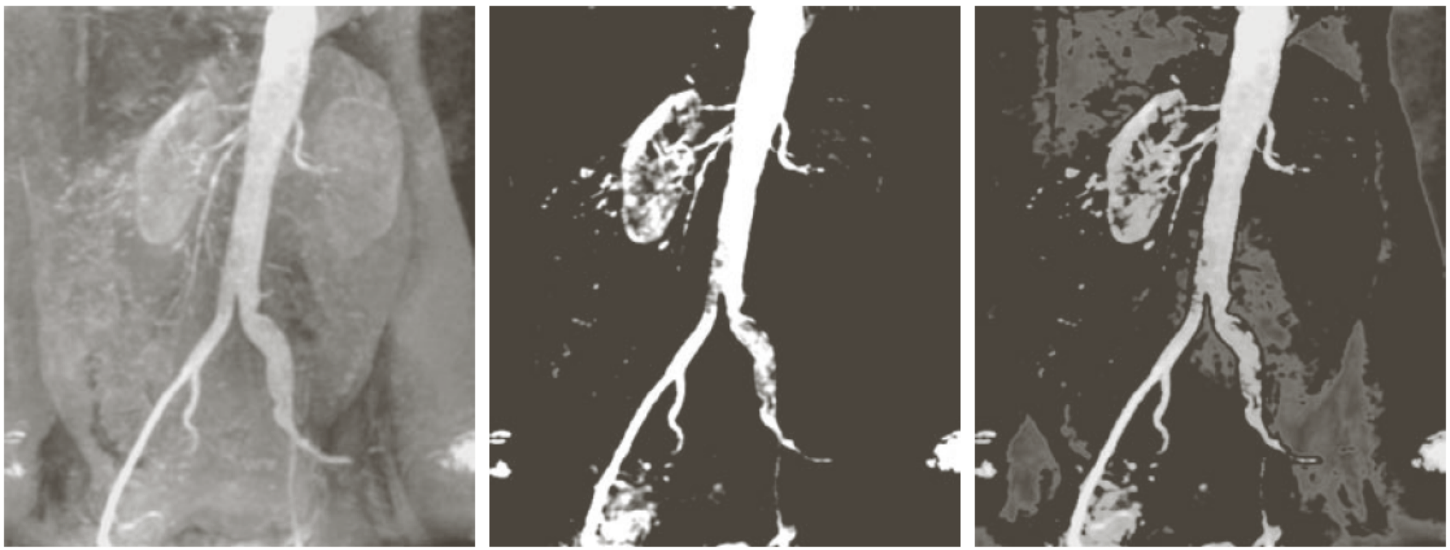
The right mapping (transform) highlights brightness in the level [A, B] but preserve details in other brightness level





Original image **binary image by left mapping**

- **Grey level slicing：在图像中提高特定灰度的亮度**



a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

- **Bit-plane slicing:** Instead of highlighting gray-level range, highlighting the contribution made to total appearance by specific bits might be desired.

  8-bit image can be divided into eight 1-bit planes, ranging from bit-plane 0 for the least significant bit to bit-plane 7 for the most significant bit.

  We may change one of the eight 1-bit plane to enhance the image.

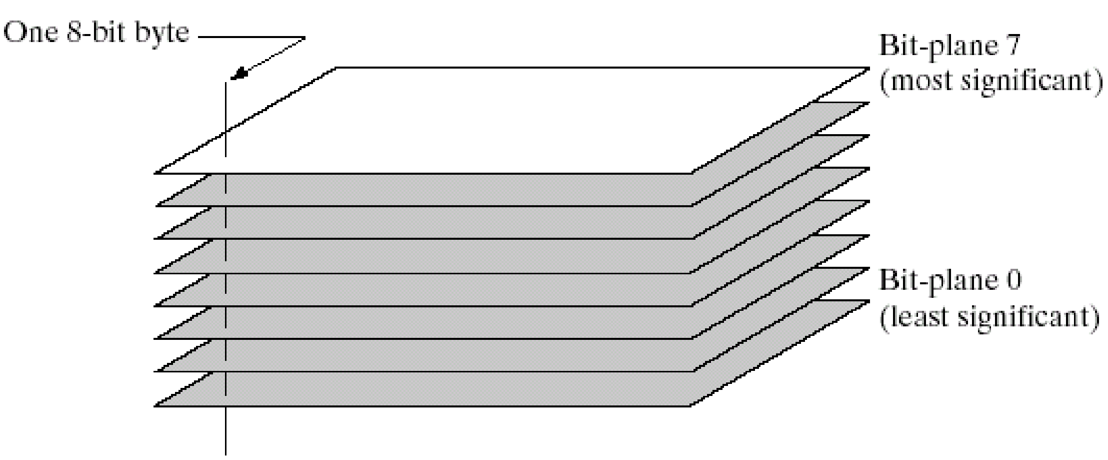


FIGURE 3.12 Bit-plane representation of an 8-bit image.

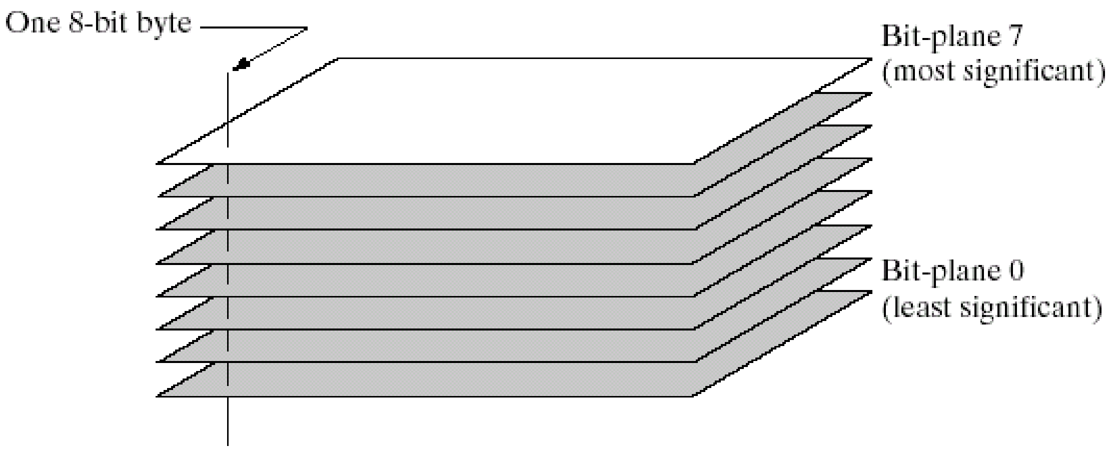


FIGURE 3.12 Bit-plane representation of an 8-bit image.

$$\begin{bmatrix} 255 & 130 \\ 80 & 24 \end{bmatrix} \xrightarrow{\text{Bit-plane}}$$

255	130	80	24
1	1	0	0
1	0	1	0
1	0	0	0
1	0	1	1
1	0	0	1
1	0	0	0
1	1	0	0
1	0	0	0

$$\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Fractal image generated by computer

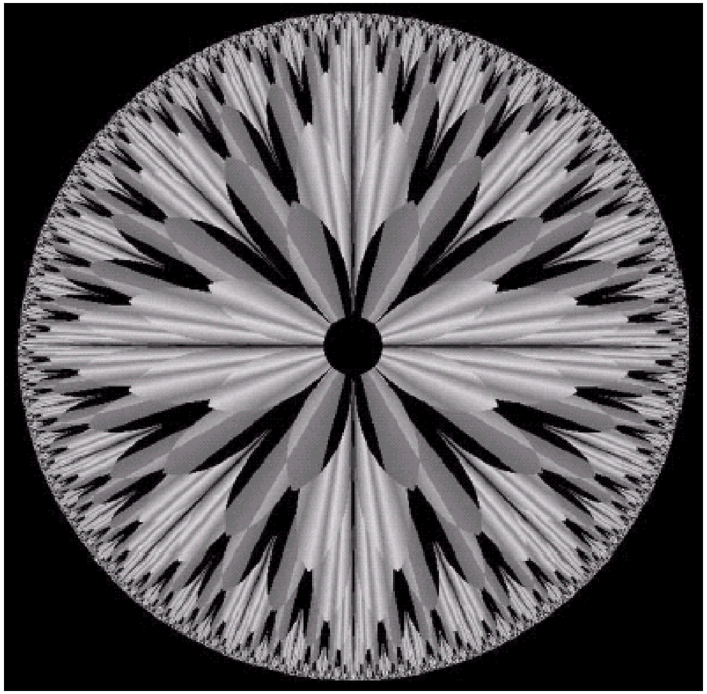


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

8-bit planes from most significant on top：

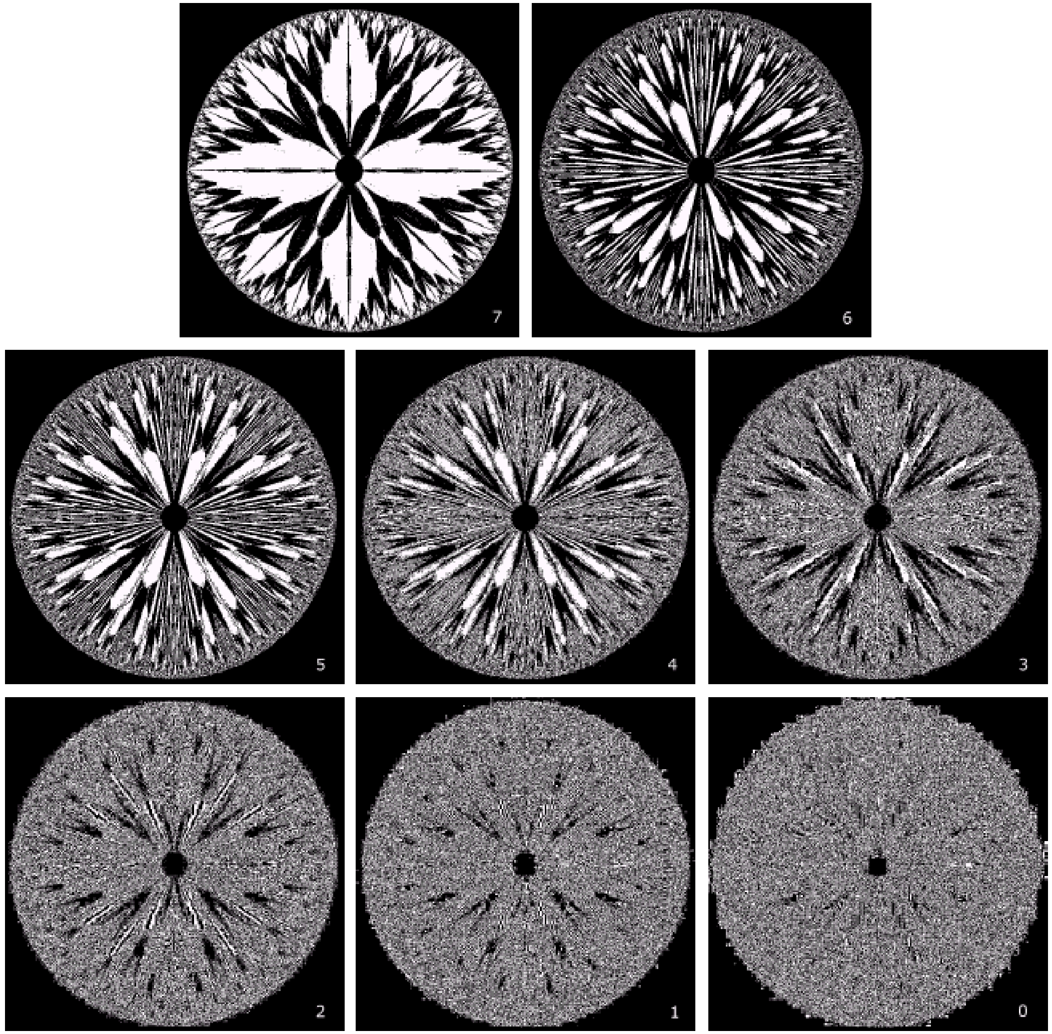


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

8-bit planes from most significant on right of bottom line:

a	b	c
d	e	f
g	h	i

FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.

a b c

FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

$$C(t)=tA+(1-t)B$$

问题：去掉低位的bit-plane，相当于对灰度值做了什么处理？

Question

So far, do we have any criteria for the image quality?

Could we automatically enhance images?

3.3 Histogram Processing

- **Concept of image histogram (直方图的概念)**

➢ **Grey level: [0, L-1]**

➢ The **histogram of a digital image** is a discrete function $h(r_k)=n_k$, where r_k is k^{th} grey level and n_k is the number of pixels in the image having gray level r_k. It is common practice to normalize a histogram by dividing each of its values by the total number of pixels in the image, denoted by n :

$$p(r_k)=n_k/n.$$

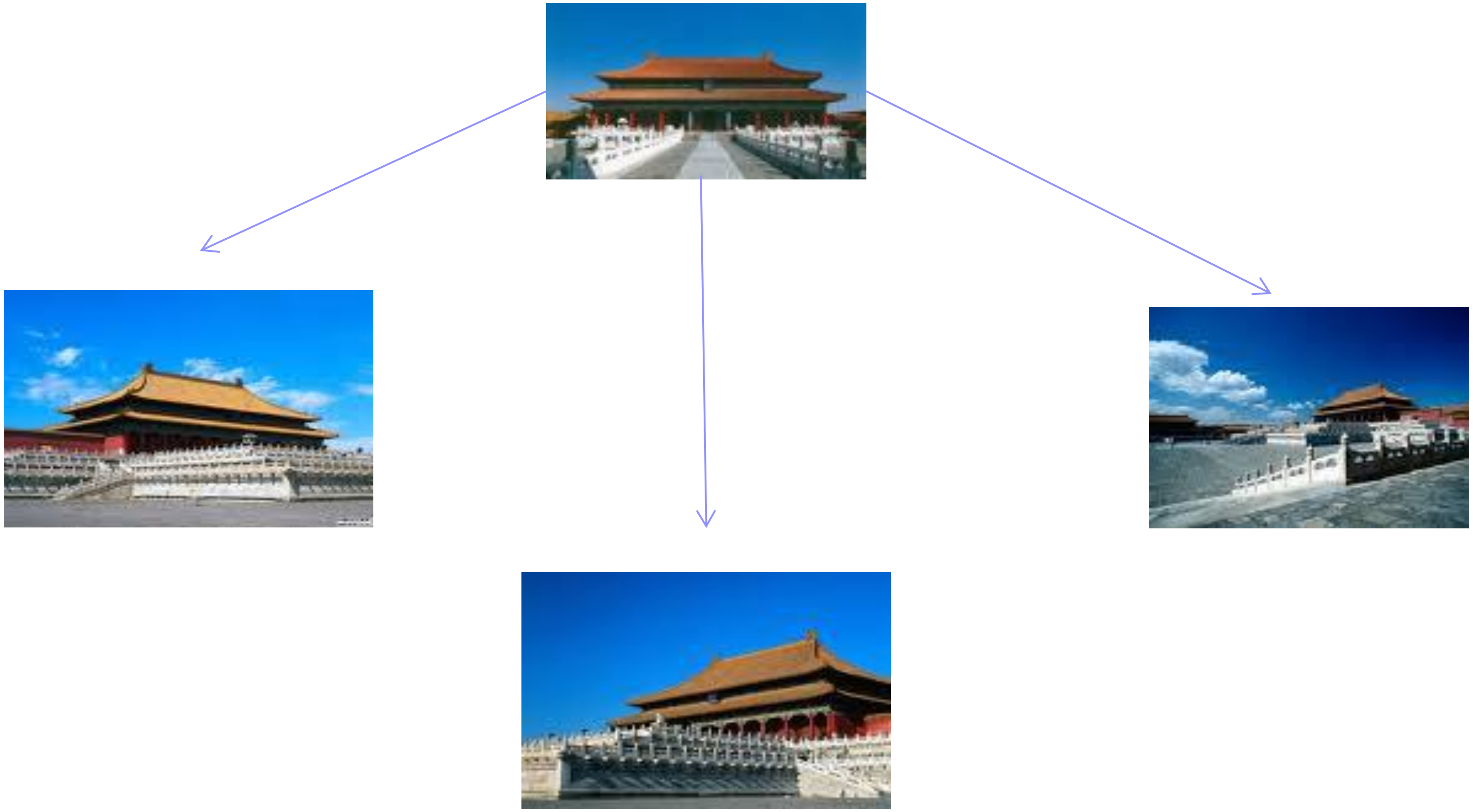
➢ **Normalized histogram** satisfy: $0 \le P(r_k) \le 1$ (k=0, 1,…L-1), and

$$\sum_k P(r_k) = 1$$

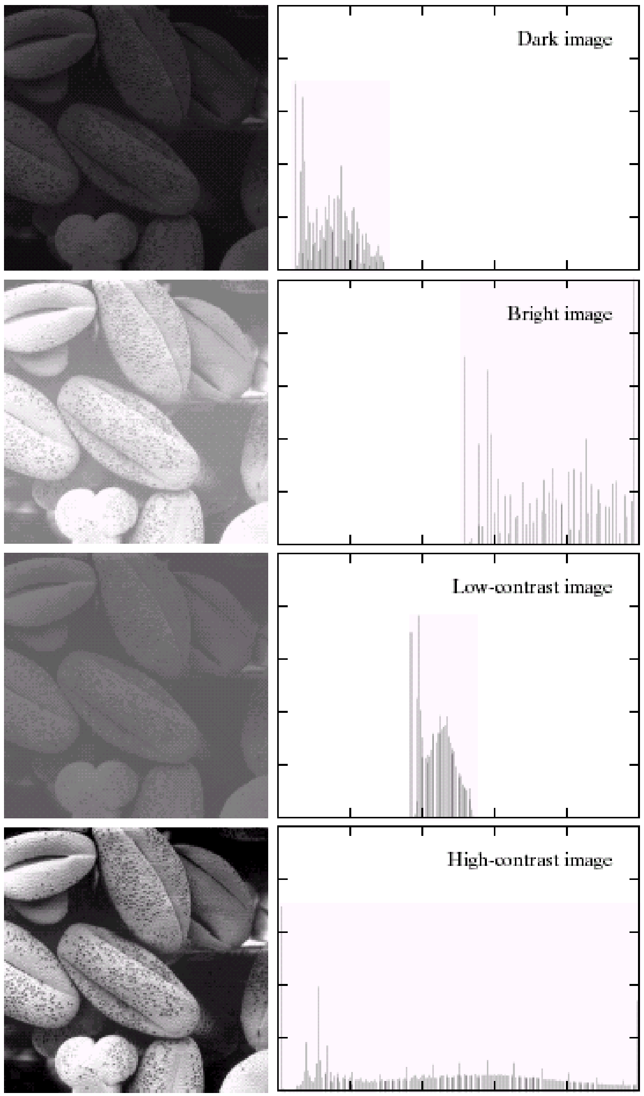
L=8

$$\begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 2 & 0 & 2 \\ 0 & 5 & 7 & 0 \\ 2 & 2 & 5 & 2 \end{bmatrix} \Longleftrightarrow \begin{bmatrix} 0 & 0 & 1/7 & 2/7 \\ 1/7 & 2/7 & 0 & 2/7 \\ 0 & 5/7 & 7/7 & 0 \\ 2/7 & 2/7 & 5/7 & 2/7 \end{bmatrix}$$

r	0	1	2	3	4	5	6	7
n	5	2	6	0	0	2	0	1
p	5/16	1/8	3/8	0	0	1/8	0	1/16

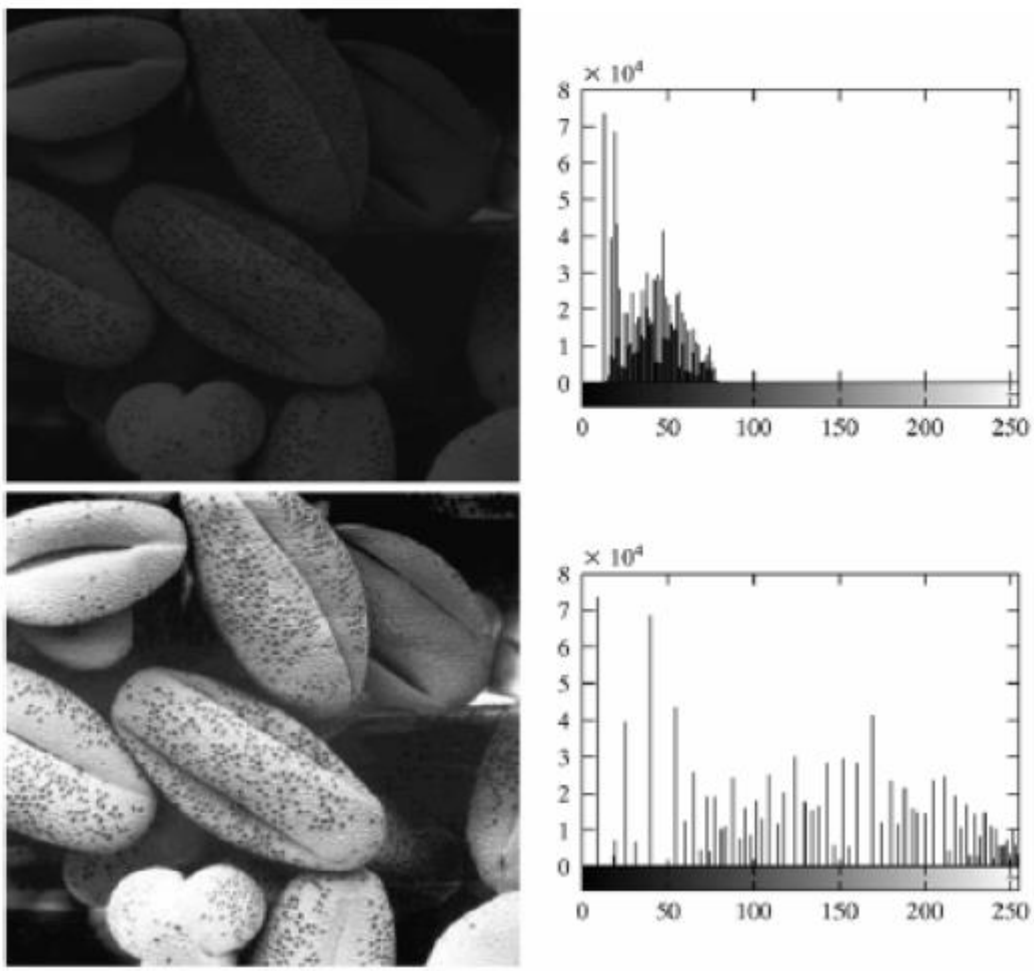


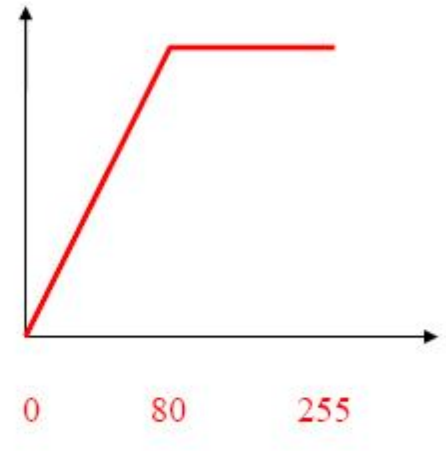
Relationship between Histogram and image visual quality



a b

FIGURE 3.15 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)





The equalization enhances brightness in the range of [0, 80] and squeeze the brightness in the range of [80, 255]

How can we get a proper transformation to make the resulting histogram equally distributed automatically?

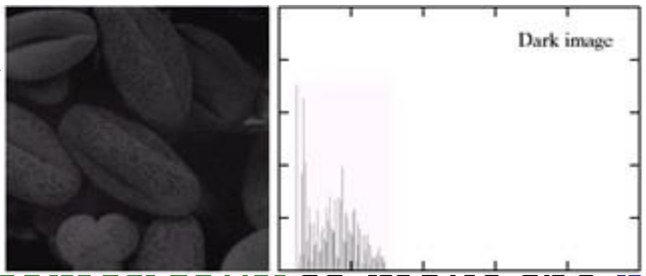
Question

Once we have an image, its histogram can be easily calculated, as well as its "visual quality".

Could we automatically get a transformation to achieve an output image with equally distributed histogram?

- **Histogram equalization（直方图均衡化）：**

r=*f*(*x*, *y*) is a image, where *r* is the gray level of the pixel in the position (*x*, *y*).

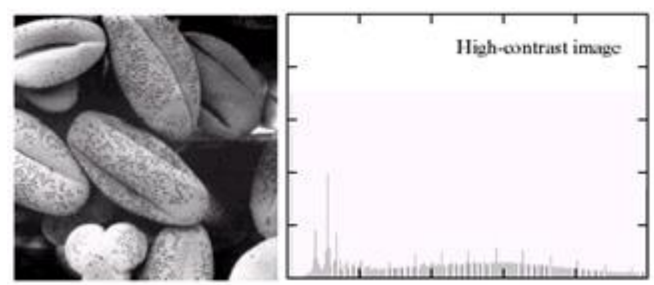
Hi 

find a transform

$T(r),\ 0 \le r \le 1$

automatically to make the new histogram of the output image ***s***=T{*f*(*x*,*y*)} close to be equally distributed

S=T(r)

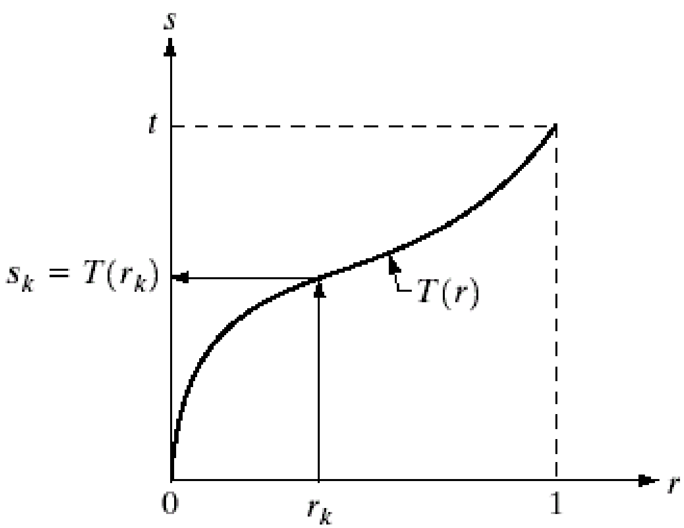


- **Histogram equalization（直方图均衡化）：**

Suppose that Histogram Equalization (**HE**) is performed by the following transformation

$$s = T(r), \quad 0 \le r \le 1$$

Here, all the (256) grey level has been normalized into $0 \le r \le 1$.



Let's see what the function (transformation) T *is* required to be

(a)(single-valued*) monotonic function

(b)$0 \le T(r) \le 1$, for $0 \le r \le 1$

* the inverse transform exists

$$r = T^{-1}(s), \quad 0 \le s \le 1$$

What dose the transform *T(r)* look like?

Histogram and the probability density function

When both the grey level r and the variable s in the histogram are normalized to $0 \le r \le 1$ and $0 \le s \le 1$, the histogram becomes the probability density function $p_r(r)$.

Let r is a variable and $s = T(r)$ denotes the cumulative probability distribution function of r

$$T(r) \triangleq \int_0^r p_r(r)dr$$

That is, the area under the histogram for the normalized grey level in the range [0, r]

$$\mathbf{s=T(r)}$$

can be proved later to be the transform we need for the case of continuous random variables.

Histogram and the probability density function

In the discrete case, the probability density function $p_r(r_k)$ becomes

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

probability of r_k → $p_r(r_k)$; r_k: a grey level; n_k: number of pixels with intensity r_k; n: total number of pixels

and the cumulative probability distribution function becomes

$$s_k = T(r_k) = \sum_{j=0}^{k} P_r(r_j) = \sum_{j=0}^{k} \frac{n_k}{n}, \qquad k = 0, 1, 2, \cdots, L-1$$

This is just the **Histogram Equalization** for image $r = f(x, y)$.

先看连续随机变量的情形：注意

$p_r(r)$ and $T(r)$ are known and $T^{-1}(s)$ satisfies (a)

$$s=T(r) \rightarrow r=T^{-1}(s)$$

若把 s 和 r 都看成是 [0，1] 区间的随机变量，$P_r(r)$ 和 $P_s(s)$ 分别表示随机变量 r 和 s 的概率密度函数。由基本的概率统计理论，在一定的条件下，两个概率密度函数之间有如下关系：

$$P_s(s) = P_r(r)\left|\frac{dr}{ds}\right|$$

考虑如何寻找满足要求的函数变换。先看连续随机变量的情况

定义图像处理中一个重要的变换函数

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

易于证明，由此变换得到的随机变量 s 的概率密度函数满足

$$P_s(s) = \begin{cases} 1, & 0 \le s \le 1 \\ 0, & \text{otherwise} \end{cases}$$

也就是说随机变量 s 具有均匀分布的密度函数。

考虑离散数据的情况：这时，采用的变换为

$$s_k = T(r_k) = \sum_{j=0}^{k} P_r(r_j) = \sum_{j=0}^{k} \frac{n_k}{n}, \quad k = 0,1,2,\cdots,L-1$$

这个式子可以看成是连续情形的“近似”，它给出的变换称为“直方图均衡化”或者“直方图线性化”

Example of Histogram and Histogram Equalization

$$\begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 2 & 0 & 2 \\ 0 & 5 & 7 & 0 \\ 2 & 2 & 5 & 2 \end{bmatrix}$$

设一幅图像$f(x,y)$如上所示，背景化为8个灰度级别。给出上图的规范化直方图（Normalized Histogram），并计算利用直方图均衡化（Histogram Equalization）方法对上图作变换，问灰度值等于1的像素，变换后的灰度值是多少？

r	0	1	2	3	4	5	6	7
P(r)	5/16	2/16	6/16	0	0	2/16	0	1/16

作直方图均衡化变换时，注意灰度值是规范在[0，1]范围之内的。因此，若要恢复到整数灰度值，如0—255，需要再做一下相应的变换。

设一幅图像灰度级别是L，则完整的直方图均衡化(Histogram Equalization)公式为：

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} P_r(r_j)$$

$$= (L-1)\sum_{j=0}^{k} \frac{n_k}{n}, \qquad k = 0, 1, 2, \cdots, L-1$$

再假设图像（矩阵）的行数和列数分别为M和N，则n=MN, 上式就是3.3-8：

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} P_r(r_j)$$

$$= (L-1)\sum_{j=0}^{k} \frac{n_k}{MN}, \qquad k = 0, 1, 2, \cdots, L-1 \tag{3.3-8}$$

k

Example 3.5 A 3-bit image (L=8) of size 64×64 pixels (MN=4096) has the intensity distribution shown in Table 3.1, where the intensity levels are integers in the range [0, L-1]=[0, 7]. Suppose its histogram is as following table

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.

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

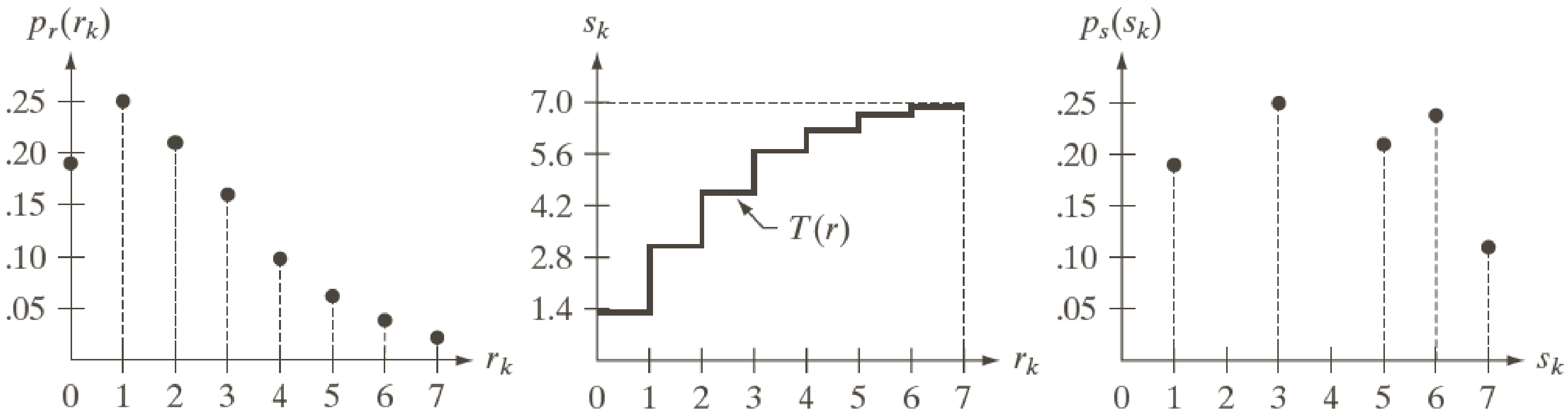
$$s_0 = T(r_0) = 7\sum_{j=0}^{0} P_r(r_j) = 1.33$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} P_r(r_j) = 7P_r(r_0) + 7P_r(r_1) = 3.08$$

Similarly, s_2=4.55, s_3=5.67, s_4=6.23, s_5=6.65, s_6=6.86, s_7=7.00. After round them to the nearest integer, we have:

s_0=1, s_1=3, s_2=5, s_3=6, s_4=6, s_5=7, s_6=7, s_7=7.

k



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.

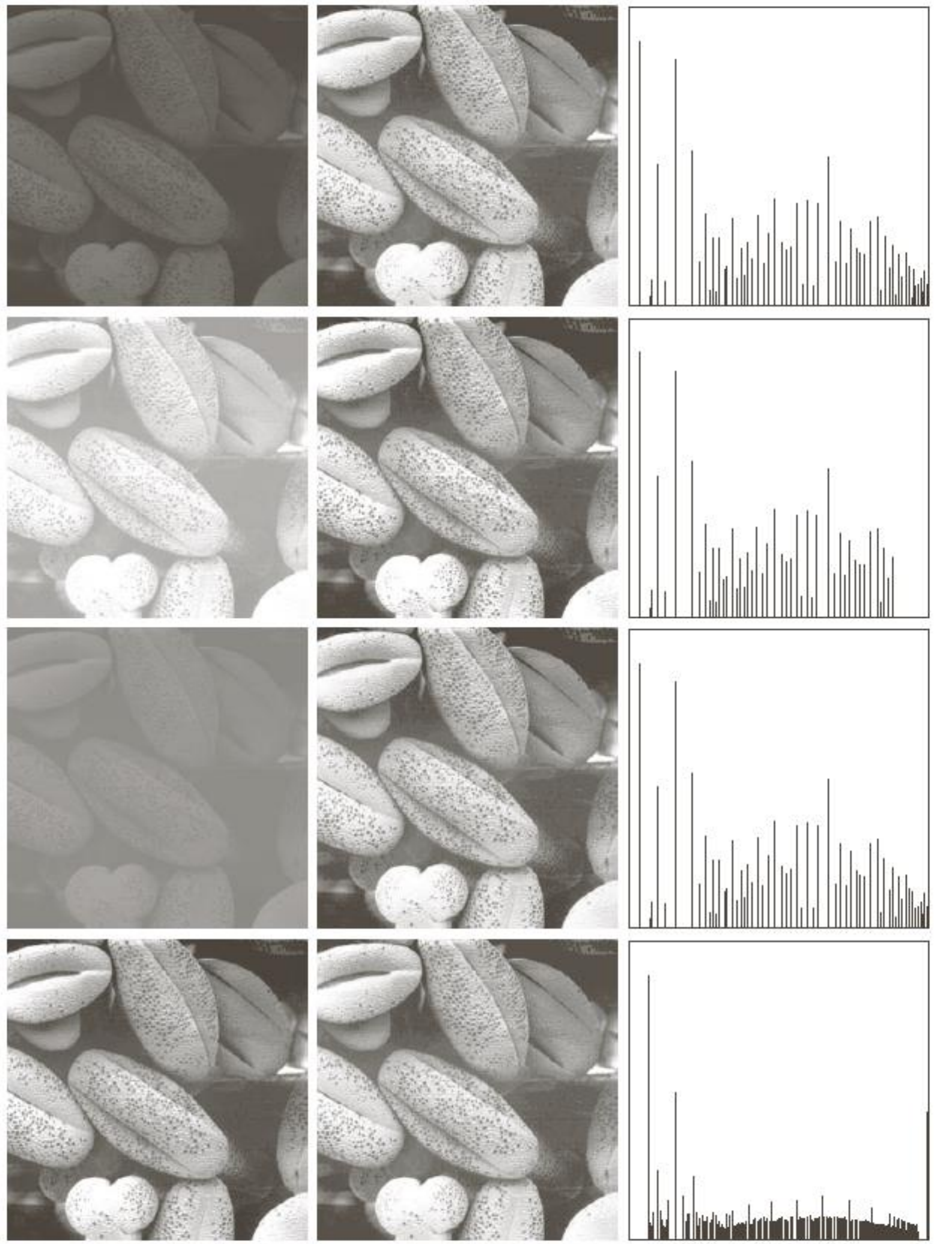


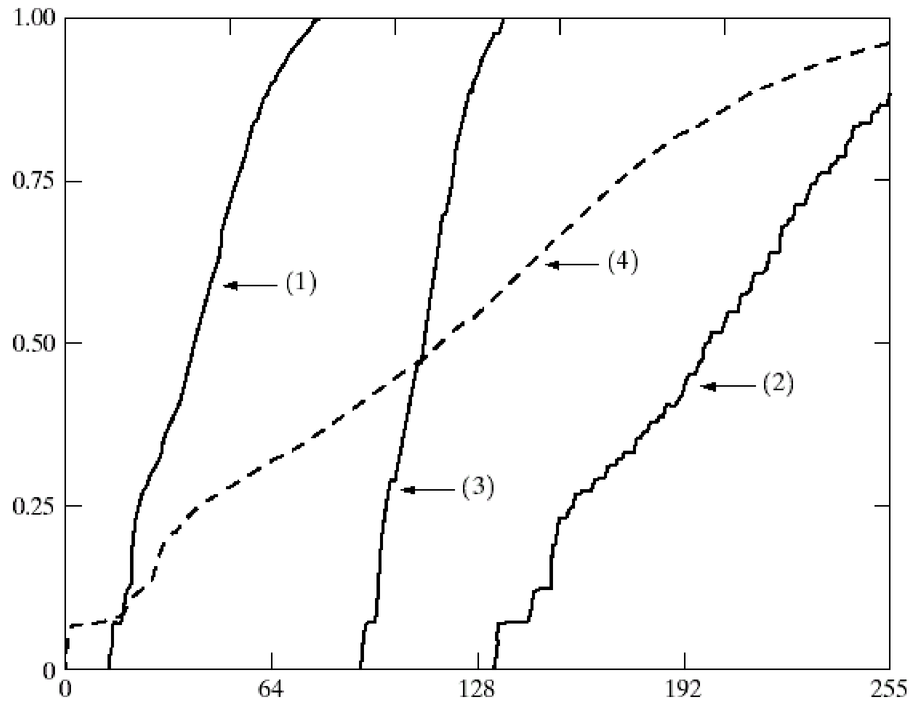
FIGURE 3.20 Left column: images from Fig. 3.16. Center column: corresponding histogram-equalized images. Right column: histograms of the images in the center column.

One example of Histogram Equalization (3.6)。

直方图均衡化方法用于图像增强有一个最大的特点：自动化。有强大的适应性强的功能。

对应的变换函数（1）——（4）

FIGURE 3.18 Transformation functions (1) through (4) were obtained from the histograms of the images in Fig.3.17(a), using Eq. (3.3-8).

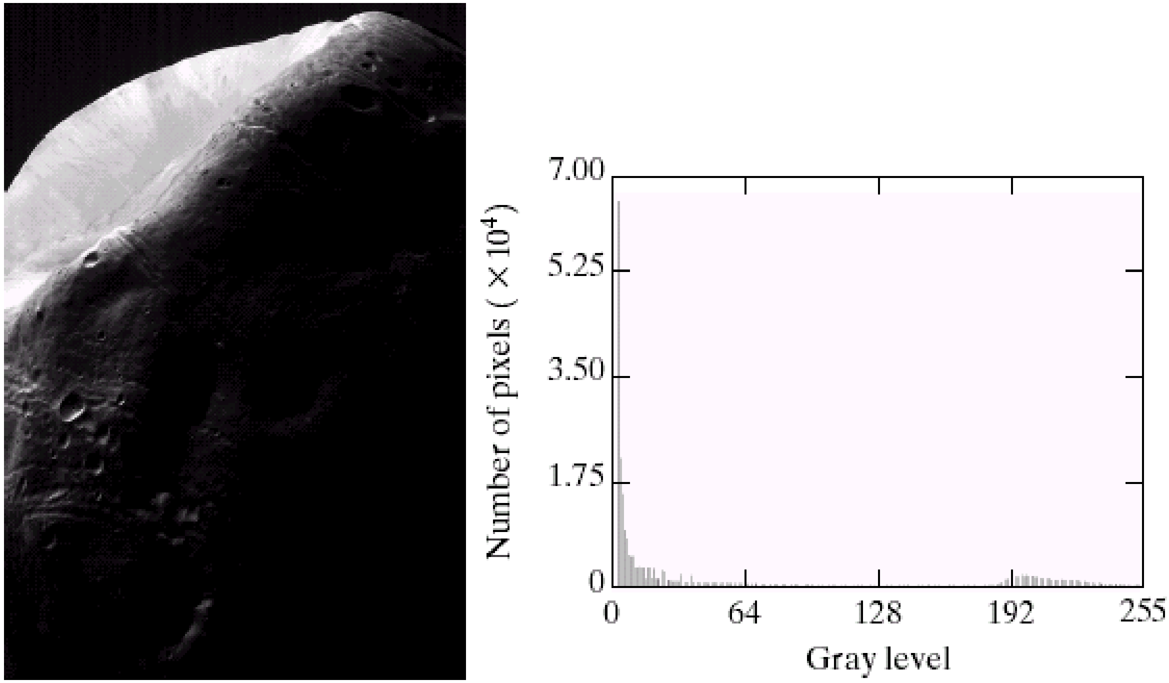


Problems：

- **Histogram equalization is not suitable for all images. For some images, the results are not satisfactory. Sometimes it even makes an image looks worse.**

- **"Gray Level Jump"： If the proportion of a certain gray level in the original image is quite large, there will be a big difference between the two adjacent gray level in the result image and the display impression may be poor.**

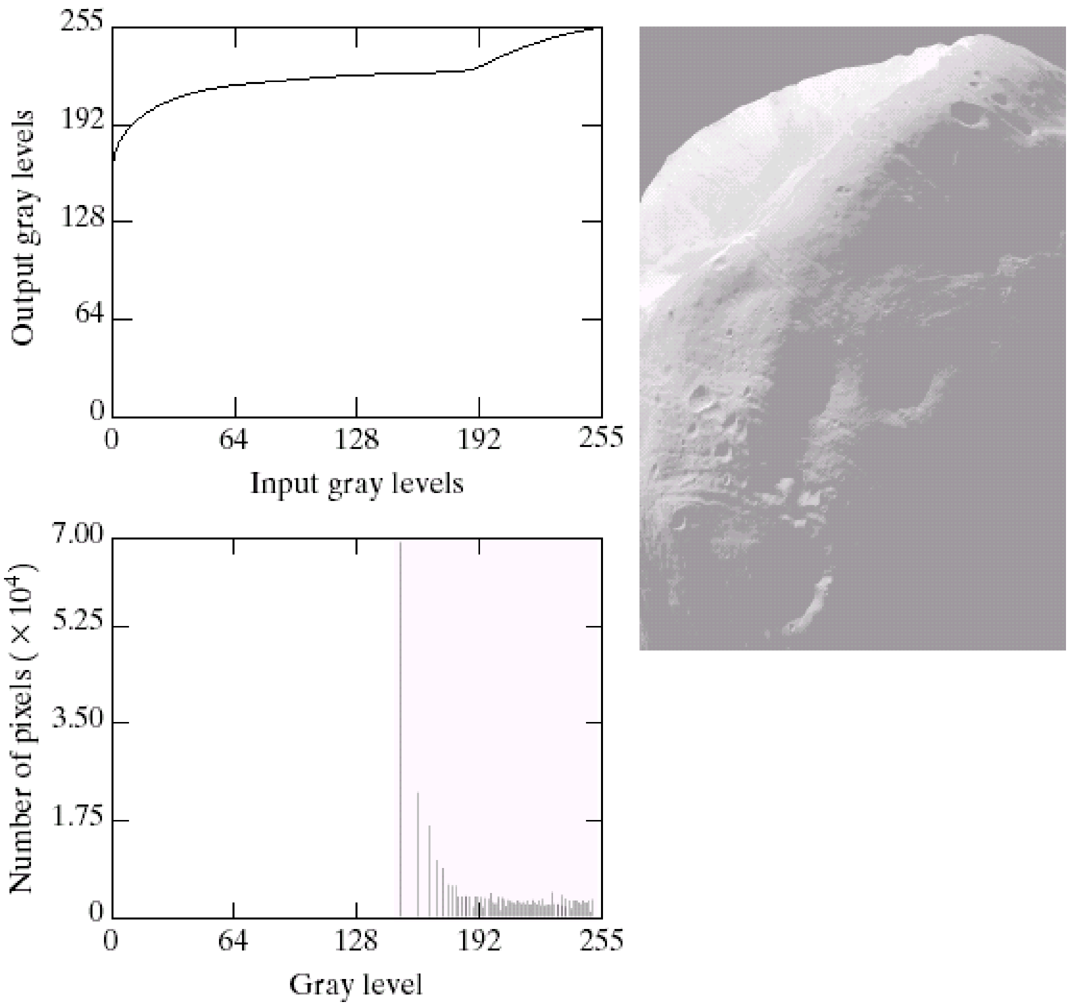
原因：对离散情形，我们无法得到类似连续情形的理论结果。反例亦的确存在。



a b

FIGURE 3.20 (a) Image of the Mars moon Photos taken by NASA's *Mars Global Surveyor*. (b) Histogram. (Original image courtesy of NASA.)

3.3 Histogram Processing （cont.)



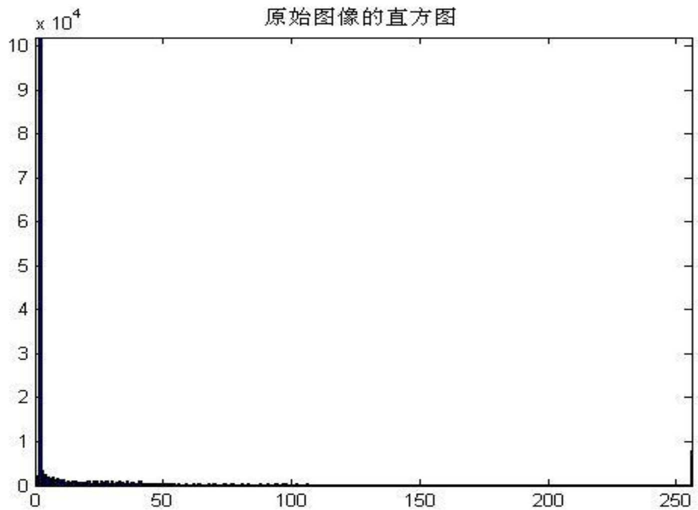
a b
c

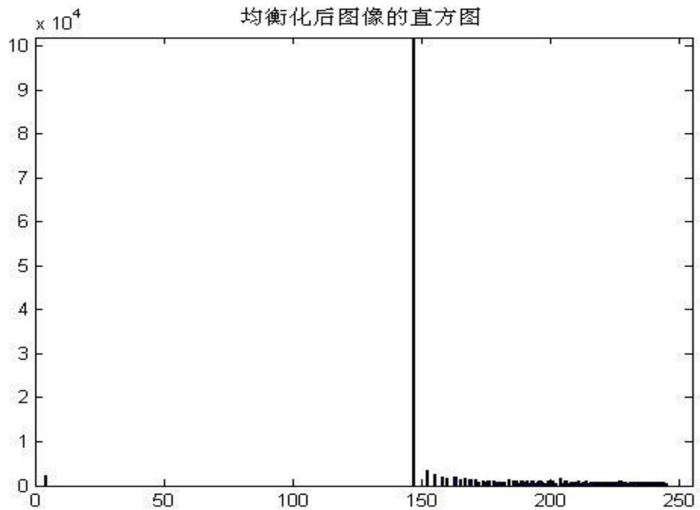
FIGURE 3.21 (a) Transformation function for histogram equalization. (b) Histogram-equalized image (note the washed-out appearance). (c) Histogram of (b).

3.3 Histogram Processing （cont.)

原始图像

均衡化后的图像


原始图像的直方图
x 10^4
10
9
8
7
6
5
4
3
2
1
0
0
50
100
150
200
250


均衡化后图像的直方图
x 10^4
10
9
8
7
6
5
4
3
2
1
0
0
50
100
150
200
250

Histogram matching

Histogram matching is an automatic histogram equalization method that adjusts the histogram to match a specified probability distribution (such as equal distribution).

$p_r(r_k)$——the original histogram (probability density function)

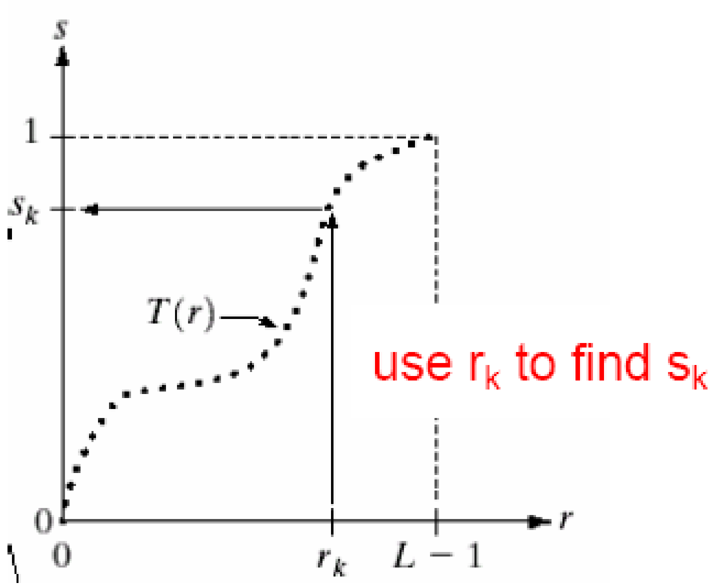
$p_z(z_j)$——the final histogram, which is required

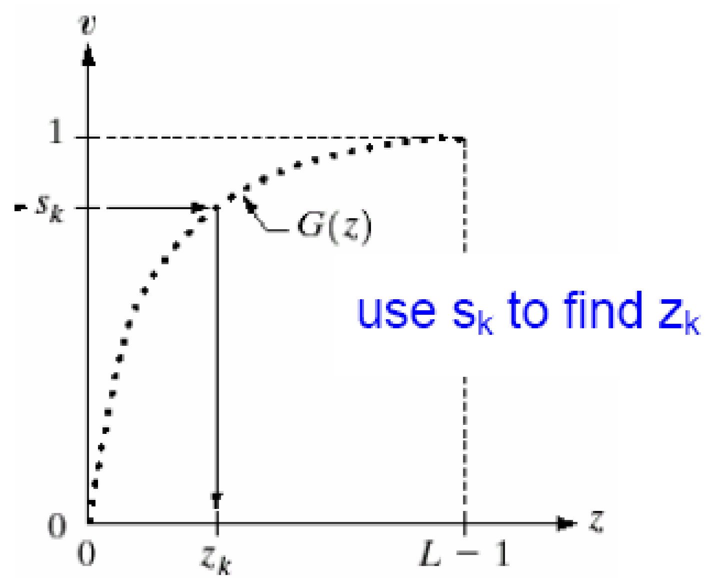
Histogram matching problem: For each grey level r_k in original image, find the suitable grey level z_k for the output image.

This task can be down in the following manner:

(1) Step 1: Calculate $s_k \triangleq T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}$ and

$$v_j \triangleq G(z_j) = \sum_{l=0}^{j} p_z(z_l) = \sum_{l=0}^{j} \frac{n_l}{n}$$





(2) Step 2 & 3, use r_k to find s_k, then use s_k to find z_k as shown above.

- 直方图匹配（寻找一个灰度变换，使输出图像有指定的直方图）　（略）

基本方法（利用连续情形推导离散数据的近似公式）：

- 计算出原图的直方图，定义一变量s满足

$$s = T(r) = \int_0^r p_r(w)dw \qquad (3.3.10)$$

- 给出输出图像期望的直方图 $p_z(z)$ ，并强令

$$G(z) = \int_0^z p_z(t)dt = s \qquad (3.3.11)$$

- 得到理论上的变换公式

$$z = G^{-1}(s) = G^{-1}T(r) \qquad (3.3.12)$$

离散图像数据的近似公式：

- 计算出原图的直方图，对*k=0, ……, L−1* 定义

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n} \qquad (3.3.13)$$

- 给出输出图像期望的直方图 $p_z(z)$，并强令

$$v_k = G(z_k) = \sum_{i=0}^{k} p_z(z_i) = s_k \qquad (3.3.14)$$

- 得到离散情形的变换公式

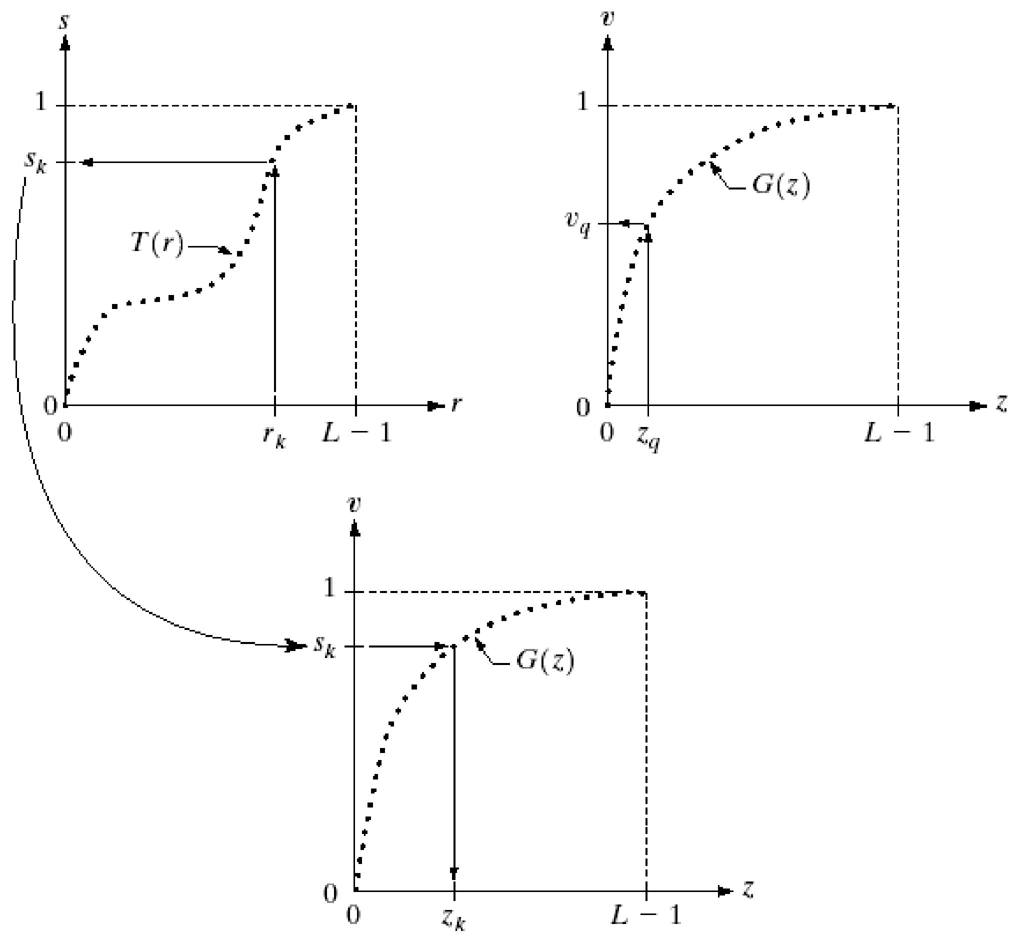
$$z_k = G^{-1}\left(T(r_k)\right) \qquad (3.3.15)$$

$$z_k = G^{-1}\left(s_k\right) \qquad (3.3.16)$$

Implementing steps(注意下述公式的意义)：

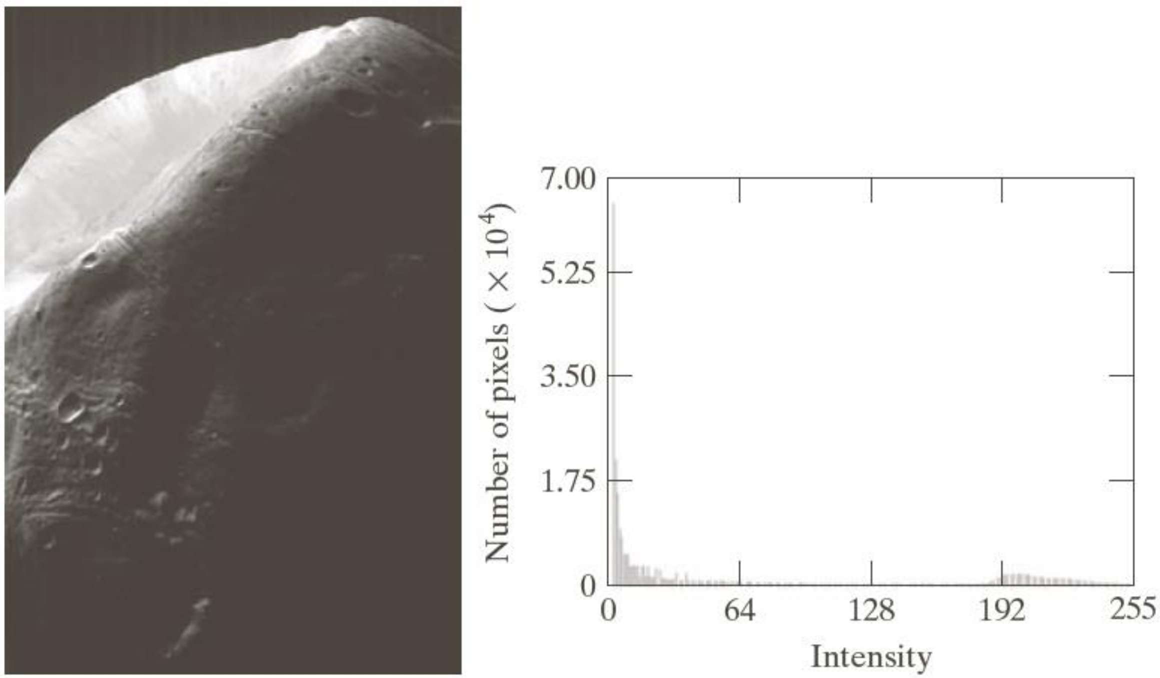
a b
c

FIGURE 3.19
(a) Graphical interpretation of mapping from r_k to s_k via $T(r)$.
(b) Mapping of z_q to its corresponding value v_q via $G(z)$.
(c) Inverse mapping from s_k to its corresponding value of z_k.



3.3 Histogram Processing （cont.)

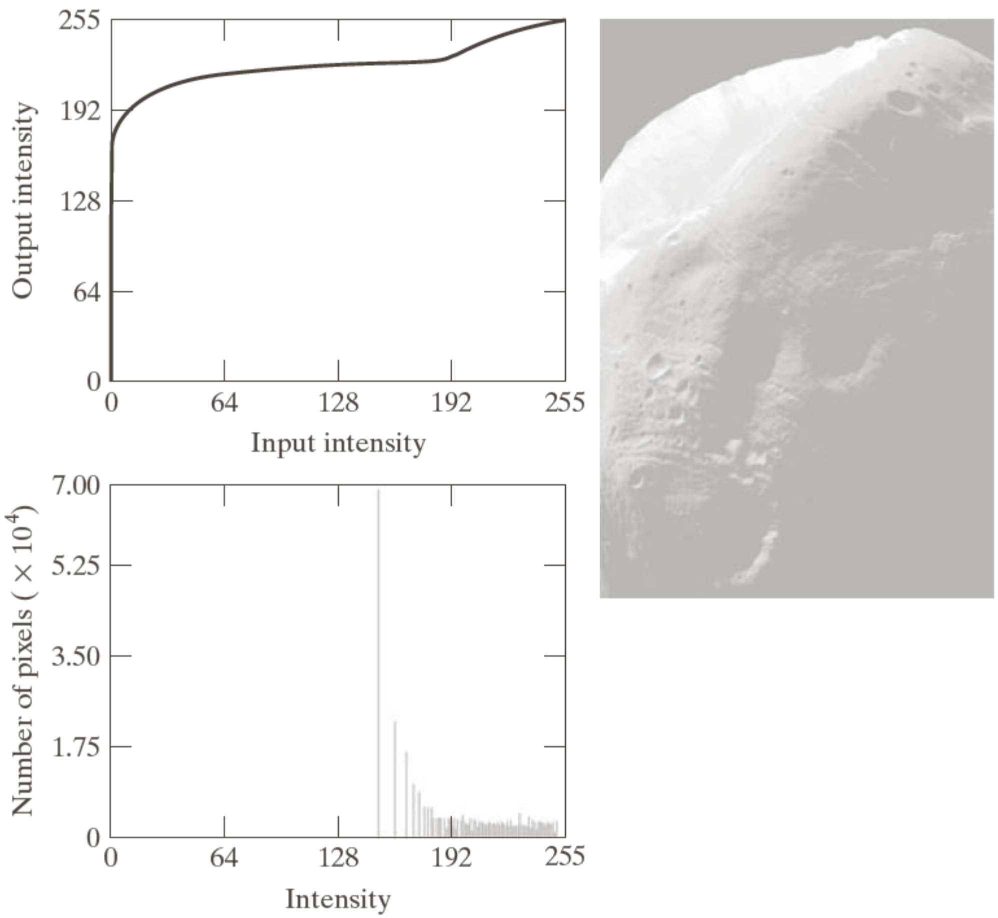
Example: (comparison between traditional histogram equalization and histogram matching) :



a b

FIGURE 3.23 (a) Image of the Mars moon Phobos taken by NASA's *Mars Global Surveyor.* (b) Histogram. (Original image courtesy of NASA.)

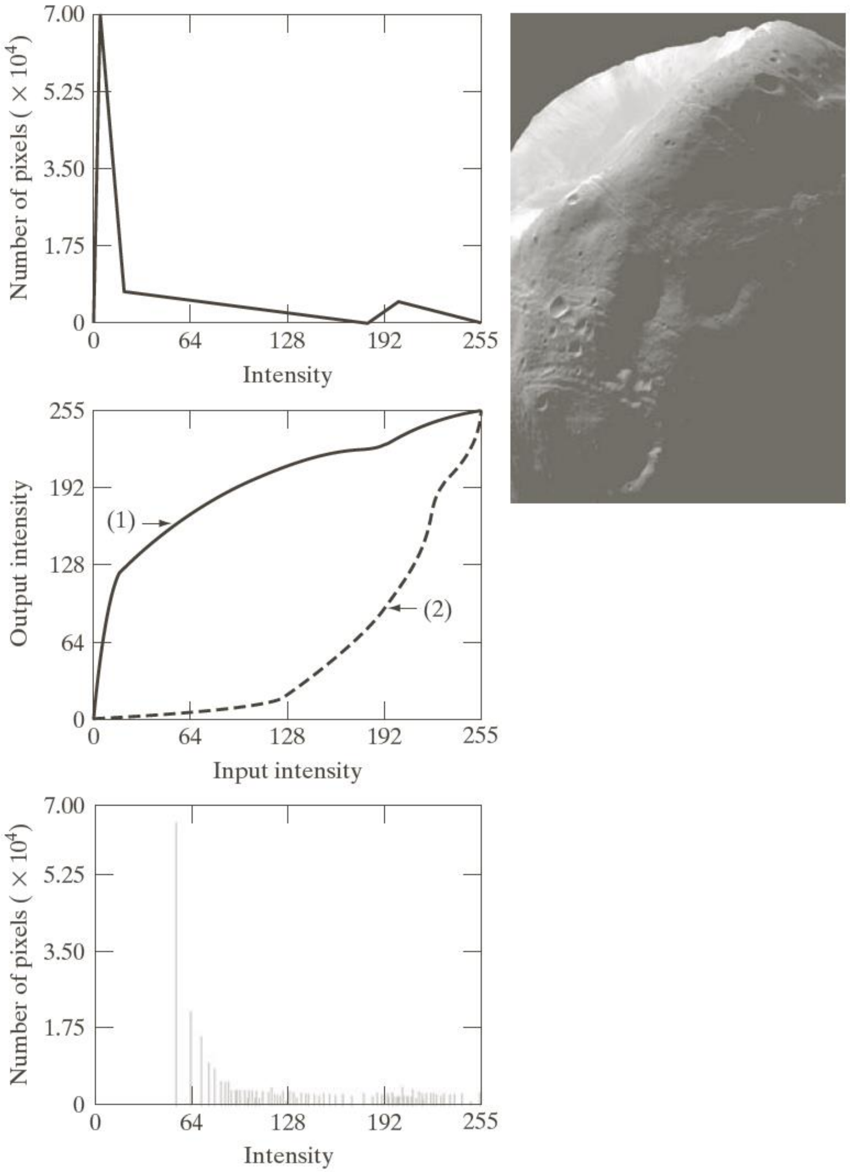
Original Histogram Equalization result



a b
c

FIGURE 3.24
(a) Transformation function for histogram equalization. (b) Histogram-equalized image (note the washed-out appearance). (c) Histogram of (b).

Histogram Matching result



a	c
b	
d	

FIGURE 3.25
(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

Stop and think: What have we learned so far?

- Image quality criteria ? Subjective? (图像质量的标准？ 主观？ 客观？)
- Objective criteria or quantitative criteria? (有没有客观或者定量标准？ ）
- Could histogram be a criteria? （直方图的分布）
- What are the factors of measurement of image quality（衡量图像好坏有哪些因素）: Contrast, others?
- Contrast ratio (对比度) is important: what is the specific definition of contrast ratio?

Prof. Xiaolin Wu's Talk

Stop and think: new research results from Prof. WU, Xiaolin

- 从信息熵的角度，“直方图均衡化”方法试图使处理后输出图像的信息熵尽可能大。因此，直方图均衡化有一定的道理；
- 图像的视觉质量包括两个方面，对比度和色调失真度（不仅仅是对比度）。
- 尽管对比度增强是经典图像处理的主要内容之一，但至今为止，图像对比度并没有很确切的定义。武教授尝试给出了一个定义（见讲稿公式（1）），该定义相当于图像对比度的期望值。
- 经典图像处理教科书在介绍图像增强方法时，都只是罗列一下现有的算法，很难对增强算法的效果好坏给出定量的描述。武教授针对这一点，对一个给定的灰度变换 **T**，定义了变换后输出图像对比度增益的度量（见课件的公式（4））；

Prof. Xiaolin Wu‘s Talk （2）

Stop and think: new research results from Prof. WU, Xiaolin

- 给出对比度增益的度量后，举例说明了对比度的增益并不是越大越好。人眼的视觉心理学表明，除了对比度之外，人眼还关注色调的光滑程度。如果色调光滑性破坏较多，增强后的图像会出现等高线状的块状效果；
- 为了解决色调失真度的问题，武教授给出了一个图像灰度变换后色调失真度的定义（公式（6））。简单说，这个定义相当于不同灰度值变换成相同灰度值的个数，有些像量化的限制。例如，如果原图上灰度值为5、6、7、8、9、10的像素在变换后灰度全部变成了8，这时变换的色调失真度大于等于6。
- 在以上的定义和分析基础上，构造了在对色调失真度作一定限制的前提下，最大化“对比度增益”的算法Optimal Contrast-Tone Mapping（OCTM）。此算法最后归结为线性规划问题（公式（10)）。

Prof. Xiaolin Wu‘s Talk （2）

Stop and think: any other methods (transformation) which can do better jobs than HE?

s=T(r)

$$T : \{0, 1, \cdots, L-1\} \rightarrow \{0, 1, \cdots, \text{Ł}-1\}$$

$$T(j) \geq T(i) \text{ if } j > i$$

$$T(i) = \sum_{0 \leq j \leq i} s_j, \; 0 \leq i < L$$

$$s_j \in \{0, 1, \cdots, \text{Ł}-1\}$$

$$\sum_{0 \leq j < L} s_j < \text{Ł}.$$

$$C(\mathrm{s}) = \sum_{0 \leq j < L} p_j s_j \qquad \Phi(\mathrm{s}) = \max_{1 \leq i \leq \text{Ł}} \{T^{-1}(i) - T^{-1}(i-1)\} \qquad CTR = \frac{C(\mathrm{s})}{\Phi(\mathrm{s})}$$

$$T^{-1}(i) = \min\{j : T(j) = i\}$$

Prof. Xiaolin Wu's Talk

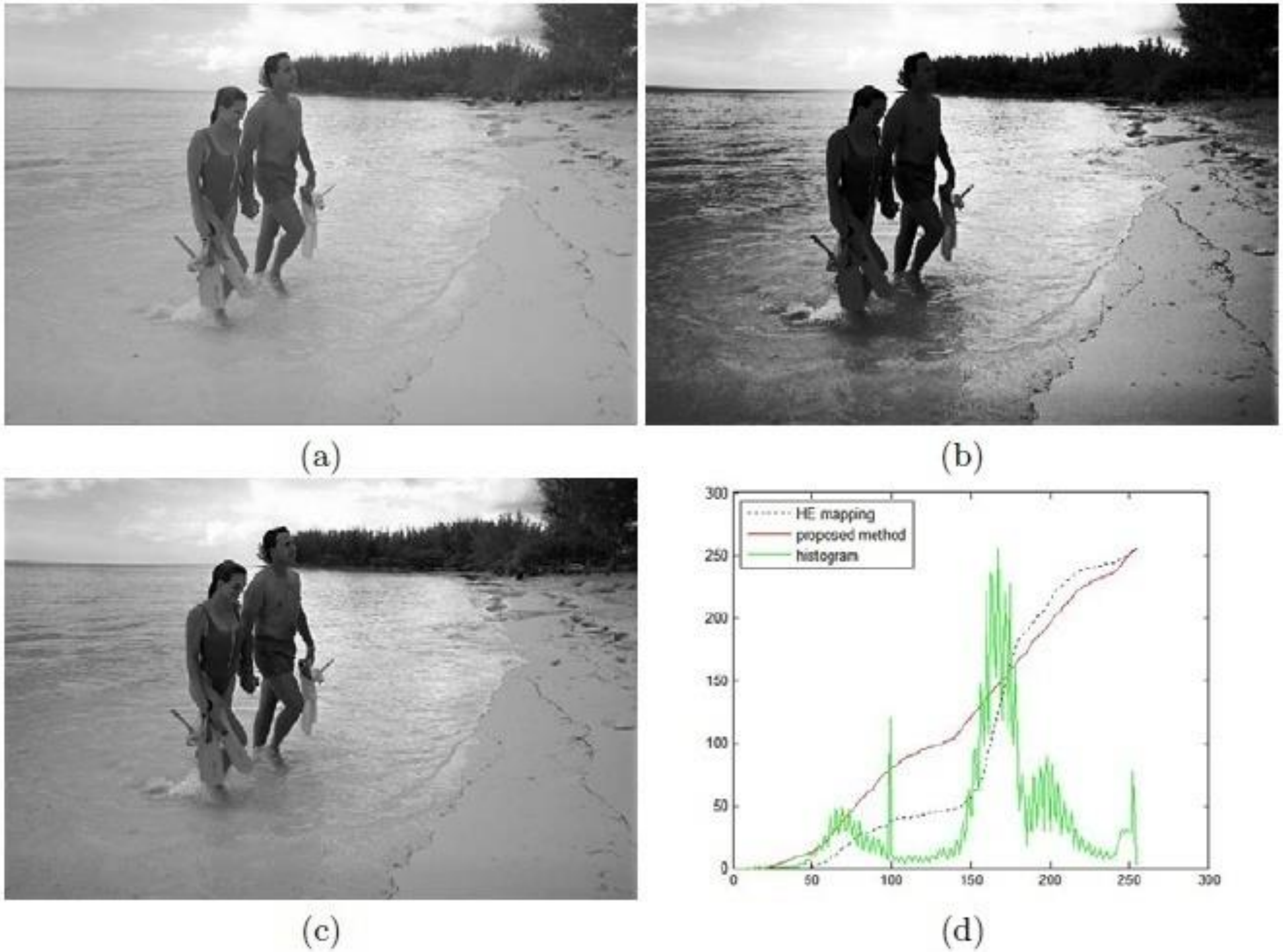


Fig. 1. (a) the original, (b) the output of histogram equalization, (c) the output of the proposed method, and (d) the transfer functions and the original histogram

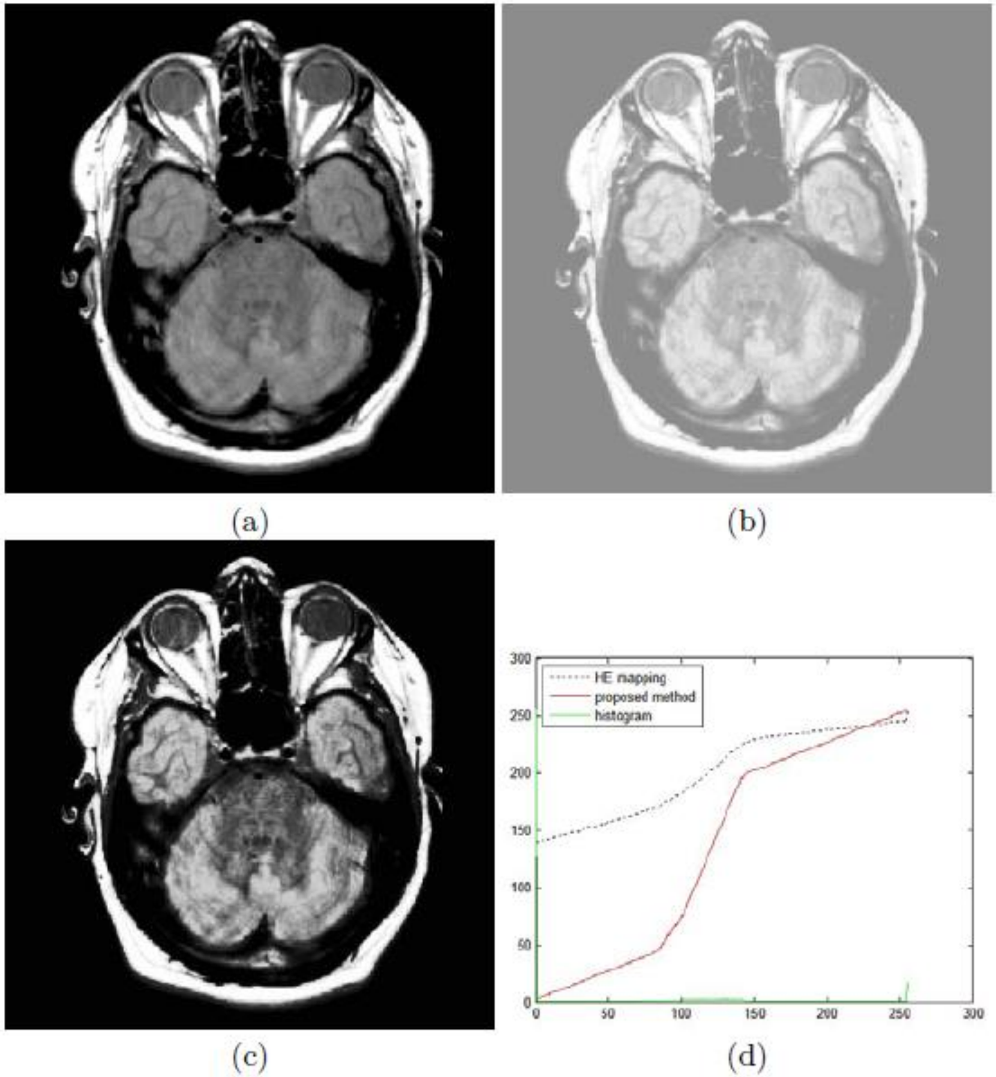


Fig. 2. (a) the original, (b) the output of histogram equalization, (c) the output of the proposed method, and (d) the transfer functions and the original histogram

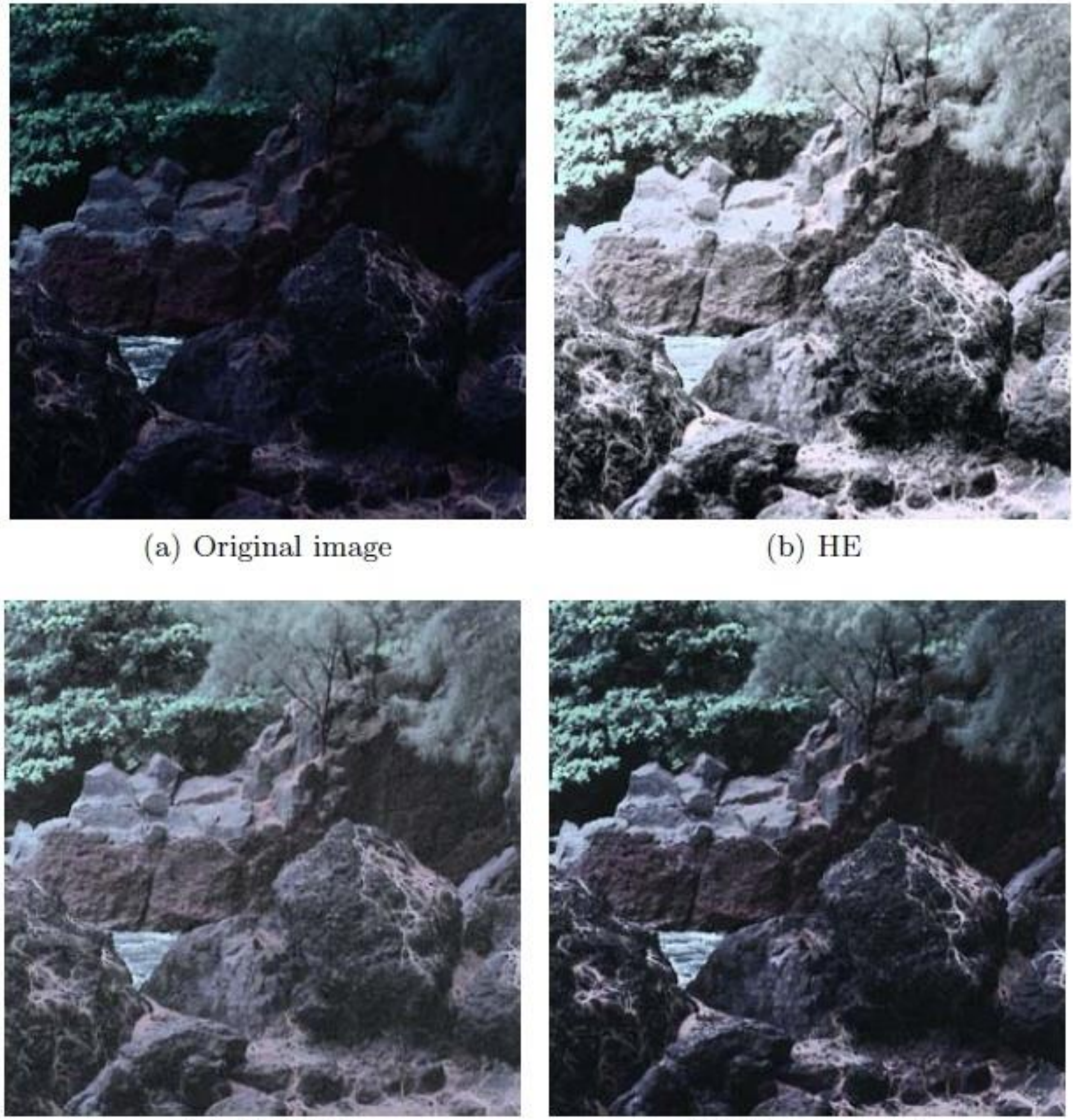
(a) (b) (c) (d)

Fig. 3. (a) the original, (b) the output of histogram equalization, (c) the output of the proposed method, and (d) the transfer functions and the original histogram

3.3 Histogram Processing （cont.)

Fig. 4. (a) the original image before Gamma correction, (b) after Gamma correction, (c) Gamma correction followed by histogram equalization, and (d) joint Gamma correction and contrast-tone optimization by the proposed method

3.3 Histogram Processing （cont.)

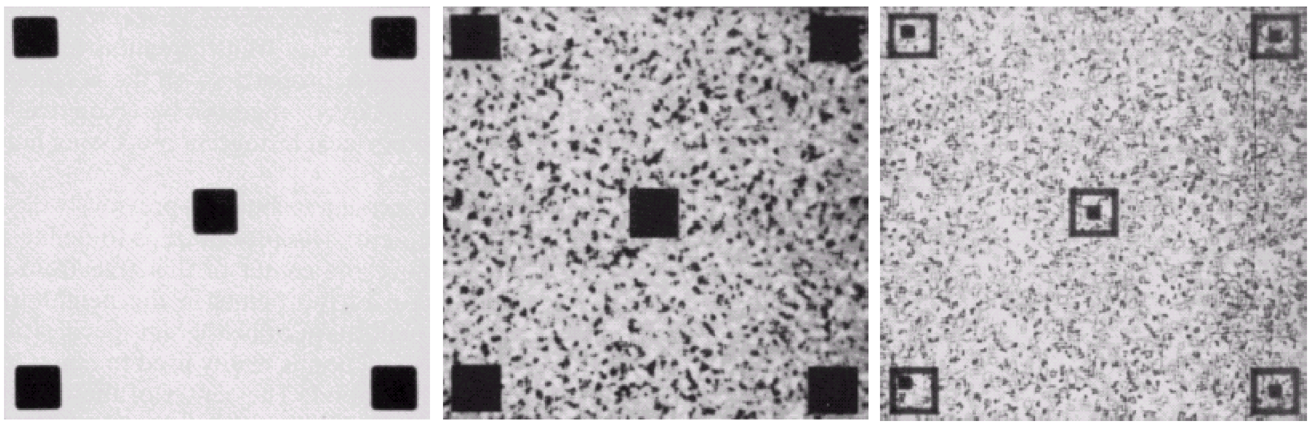


(a) Original image (b) HE

(c) CLAHE (d) The proposed

Fig. 5. Comparison of different methods on image Rocks

- **Local Enhancement** (skip)



a b c

FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

- **Statistic properties of histogram** (skip)

Recall the probability density function (histogram). For details, see subsection 3.3.4.

probability of r_k $\longrightarrow p_r(r_k) = \frac{n_k}{n}$ number of pixels with intensity r_k / total number of pixels

The mean value (the first moment) is defined as

$$m = \sum_{j=0}^{L-1} r_j \bullet p_r(r_j) = \sum_{j=0}^{k} r_j \bullet \frac{n_j}{n}$$

The standard deviation (square root of the second moment) is defined as

$$\sigma^2 \triangleq \mu_2 = \sum_{j=0}^{L-1} (r_j - m)^2 p_r(r_j) = \sum_{j=0}^{L-1} (r_j - m)^2 \frac{n_j}{n}$$

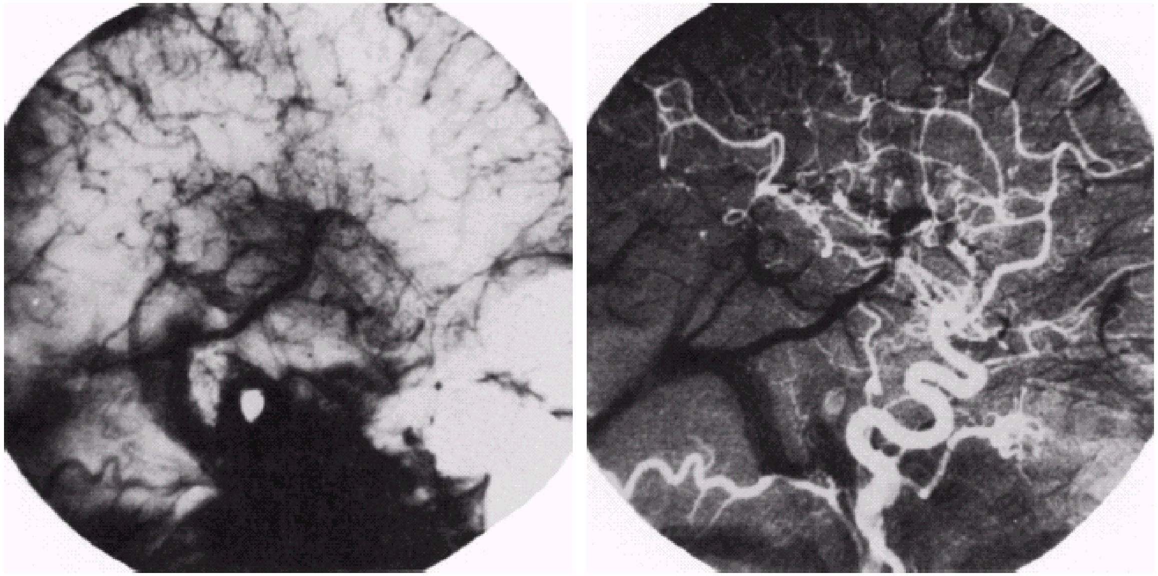
The nth moment is defined as

$$\mu_n \triangleq \sum_{j=0}^{L-1} (r_j - m)^n p_r(r_j) = \sum_{j=0}^{L-1} (r_j - m)^n \frac{n_j}{n}$$

3.* Enhancement using Arithmetic/Logic operation

- Addition (+)
- Subtraction (-)
- AND, OR, NOT

ALL ARE pixel to pixel operation



a b

FIGURE 3.29 Enhancement by image subtraction. (a) Mask image. (b) An image (taken after injection of a contrast medium into the bloodstream) with mask subtracted out.

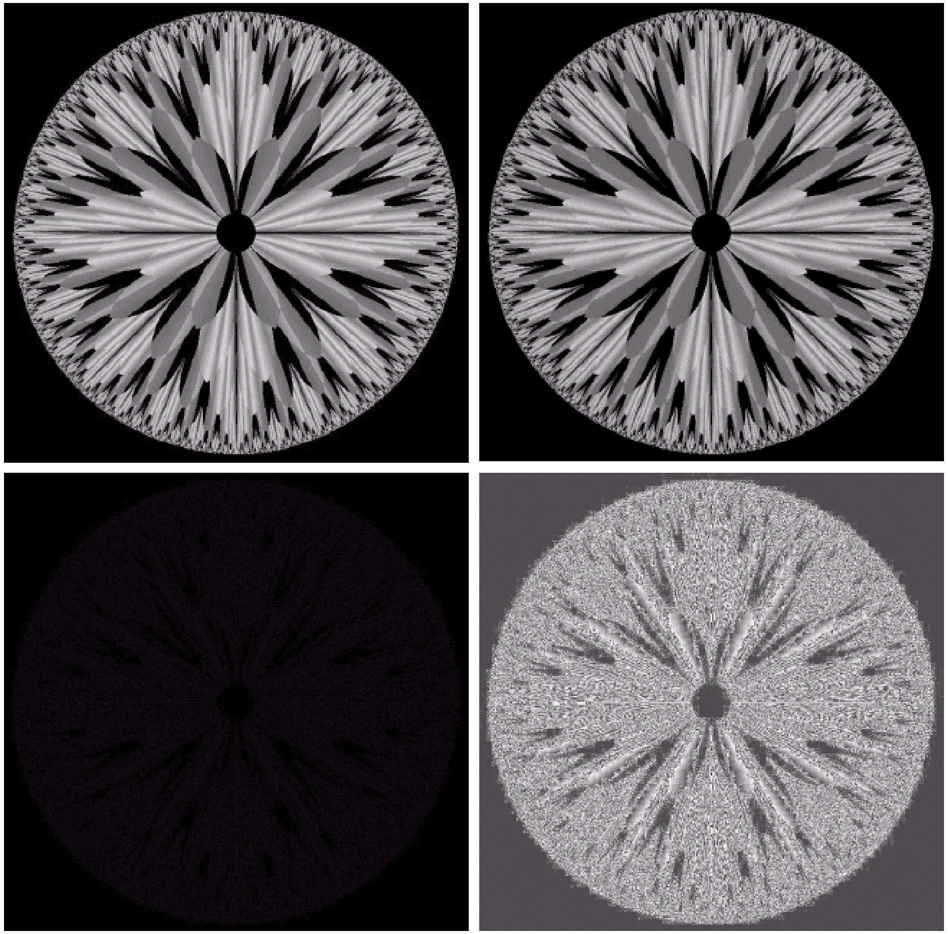
Using subtraction to remove mask image in medical applications.

图像减法处理后“差”的显示要注意数值的范围，标定处理后的数据。

- 图像减法处理：其主要作用就是增强两幅图之间的差异 。

a b
c d

FIGURE 3.28
(a) Original fractal image.
(b) Result of setting the four lower-order bit planes to zero.
(c) Difference between (a) and (b).
(d) Histogram-equalized difference image.
(Original image courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA).



- **Averaging K images to remove zero-mean Gaussian noise**

➢ **Theoretical Background**

$f(x,y)$——原始图像

$\eta(x,y)$——随机噪音，在各个坐标点（x, y）上的噪音不相关，且均值为零

$g(x,y)=f(x,y)+\eta(x,y)$——带噪音的图像

对 K 幅带噪音的图像取平均

$$\bar{g}(x,y)=\frac{1}{K}\sum_{k=1}^{K}g_k(x,y)$$

则有

$$E\{\overline{g}(x,y)\} = f(x,y)$$

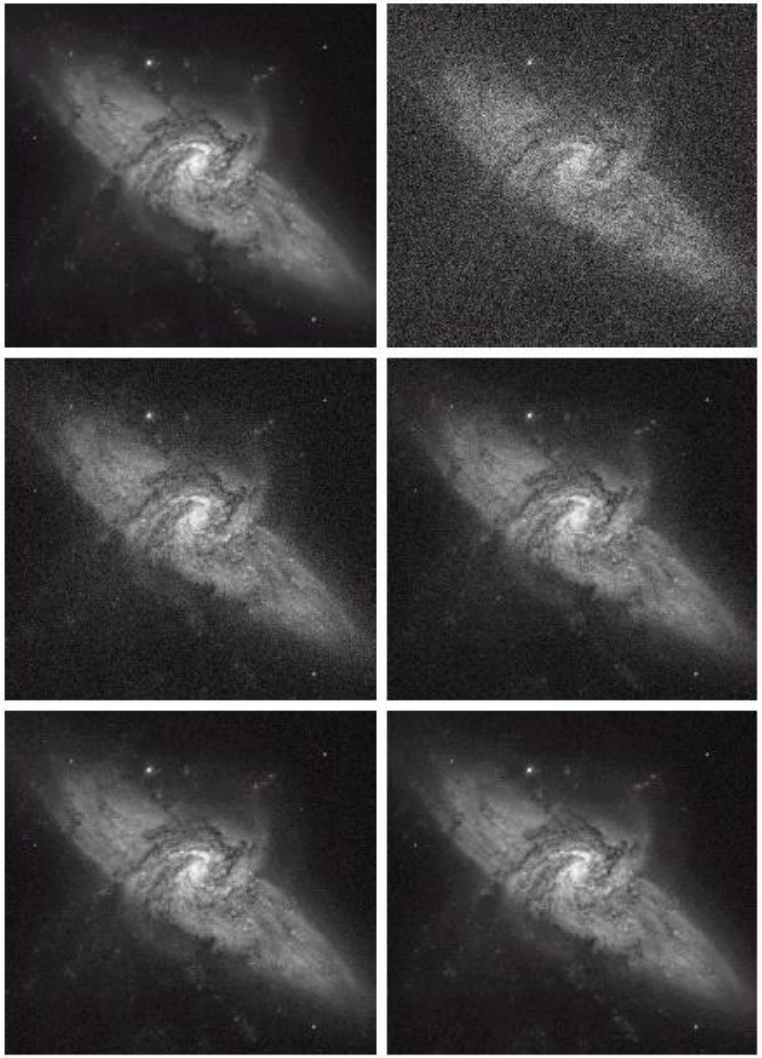
$$\sigma^2_{\overline{g}(x,y)} = \frac{1}{K}\sigma^2_{\eta(x,y)}$$

其中，$E\{\overline{g}(x,y)\}$是$\overline{g}$的期望值，$\sigma^2_{\overline{g}(x,y)}$和$\sigma^2_{\eta(x,y)}$分别是$\overline{g}$和$\eta$的方差。在平均图像中，任何一点的标准差为：

$$\sigma_{\overline{g}(x,y)} = \frac{1}{\sqrt{K}}\sigma_{\eta(x,y)}$$

这是平均处理方法去噪的理论背景。

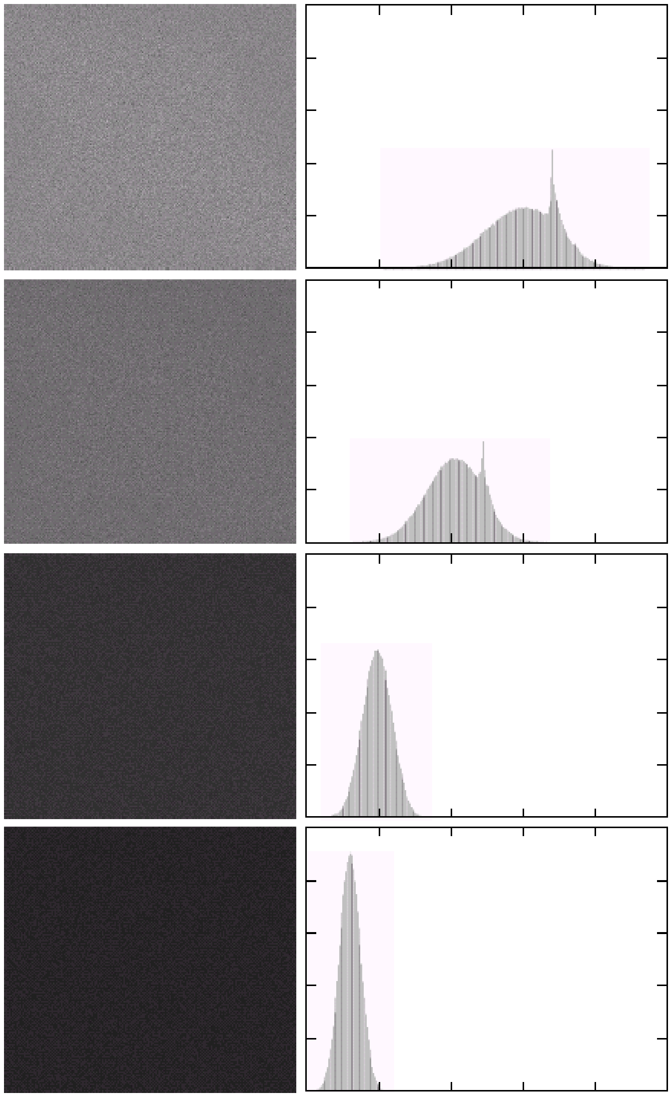
Example 3.8: Image averaging in the field of astronomy.



A clear original image is shown on the left, and a noisy image on the right.

The four images below are averaged from 8, 16, 64, 128 noisy images.

The more noisy images used, the better result.



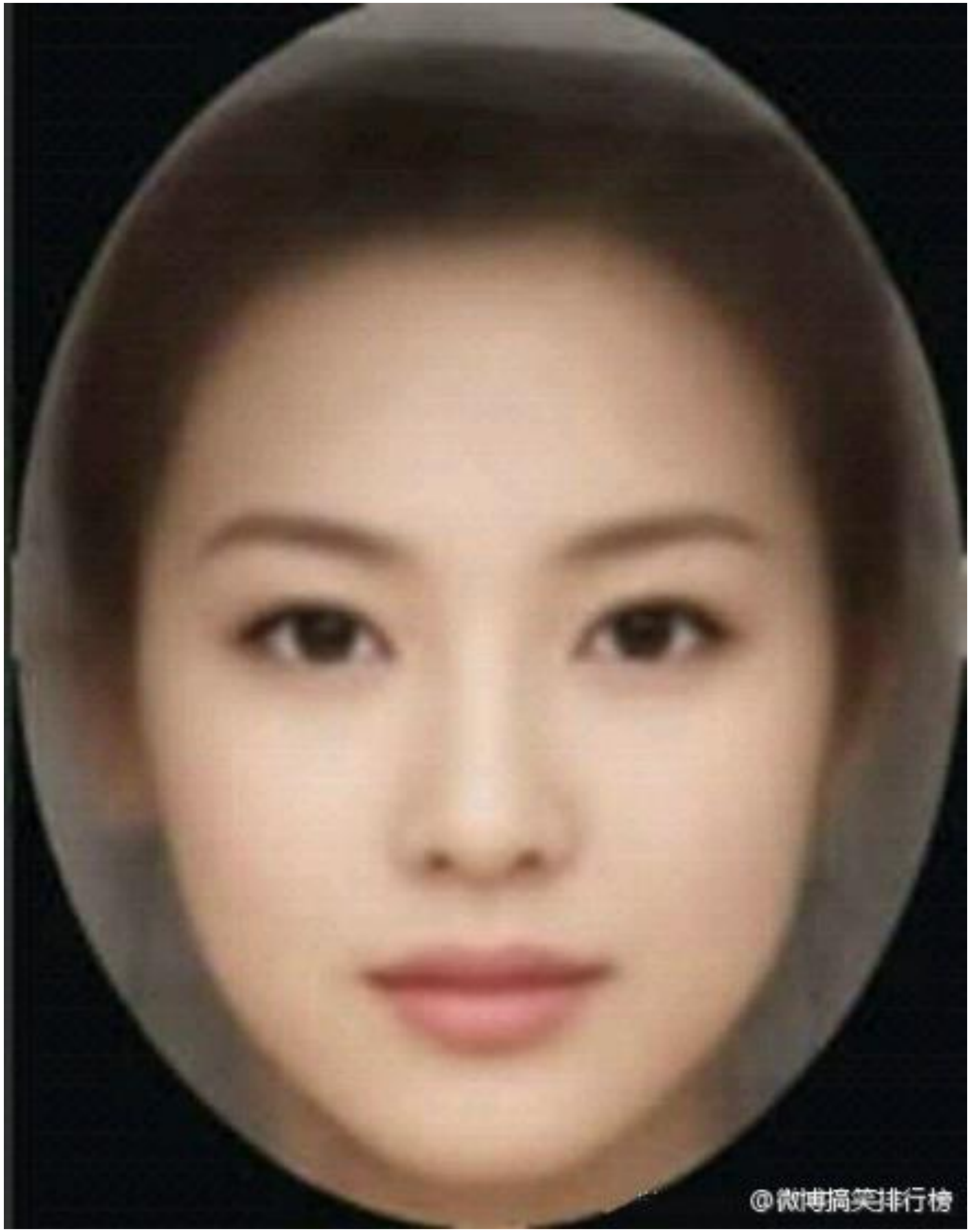
a b

FIGURE 3.31 (a) From top to bottom: Difference images between Fig. 3.30(a) and the four images in Figs. 3.30(c) through (f), respectively. (b) Corresponding histograms.

- ✓ 左边一列从上到下，是图(c)-(f)每一幅均值处理后的图像与原始图像(a)的差值图像。右边是相应的差值图像直方图。
- ✓ 平均处理后要注意规范或标定图像数据

- ✓ 希腊夜景

- ✓ 希腊夜景



- ✓ 这张脸就是用东方人的脸平均出来的。其实，用任何一组足够多的人脸合成出来的平均脸都很漂亮
- ✓ 平均脸网站：
- ✓ TA处有数据

3.* Enhancement Using Arithmetic/Logic Operation (cont.)

软件学院09级男生

软件学院09级女生

软件学院09级数媒男生

软件学院09级计应男生

软件学院09级数媒女生

软件学院09级计应女生

25个男明星的平均脸

25个女明星的平均脸

3.* Enhancement Using Arithmetic/Logic Operation (cont.)

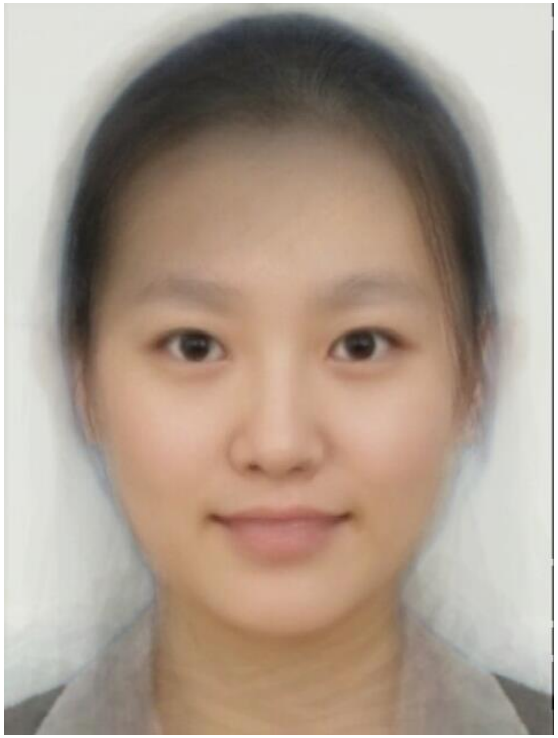
人像原图

美化图

参考图

3.* Enhancement Using Arithmetic/Logic Operation (cont.)

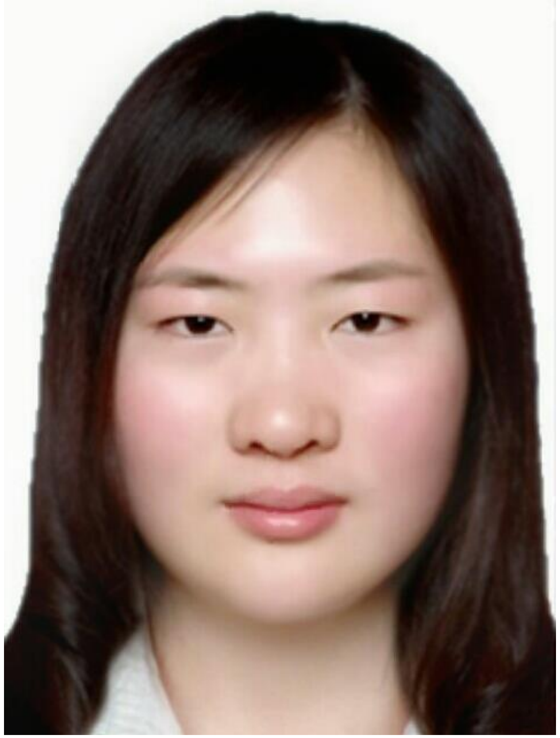
人像原图



美化图

参考图

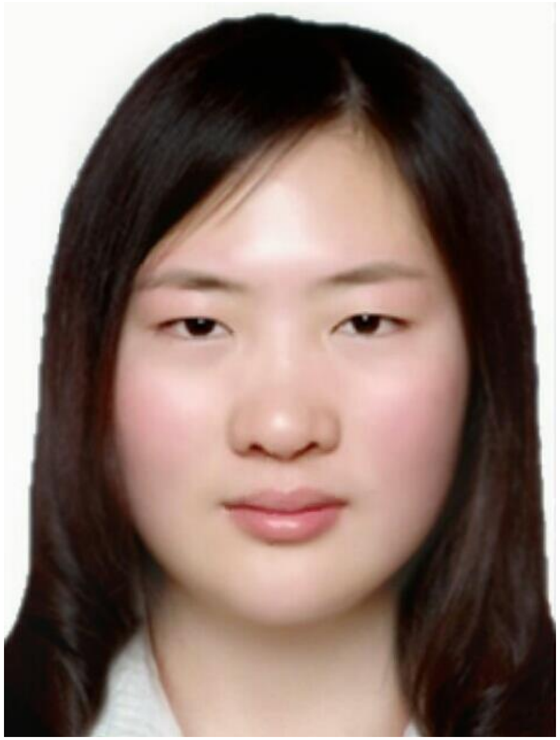
3.* Enhancement Using Arithmetic/Logic Operation (cont.)



人像原图

美化图

参考图



人像原图

美化图

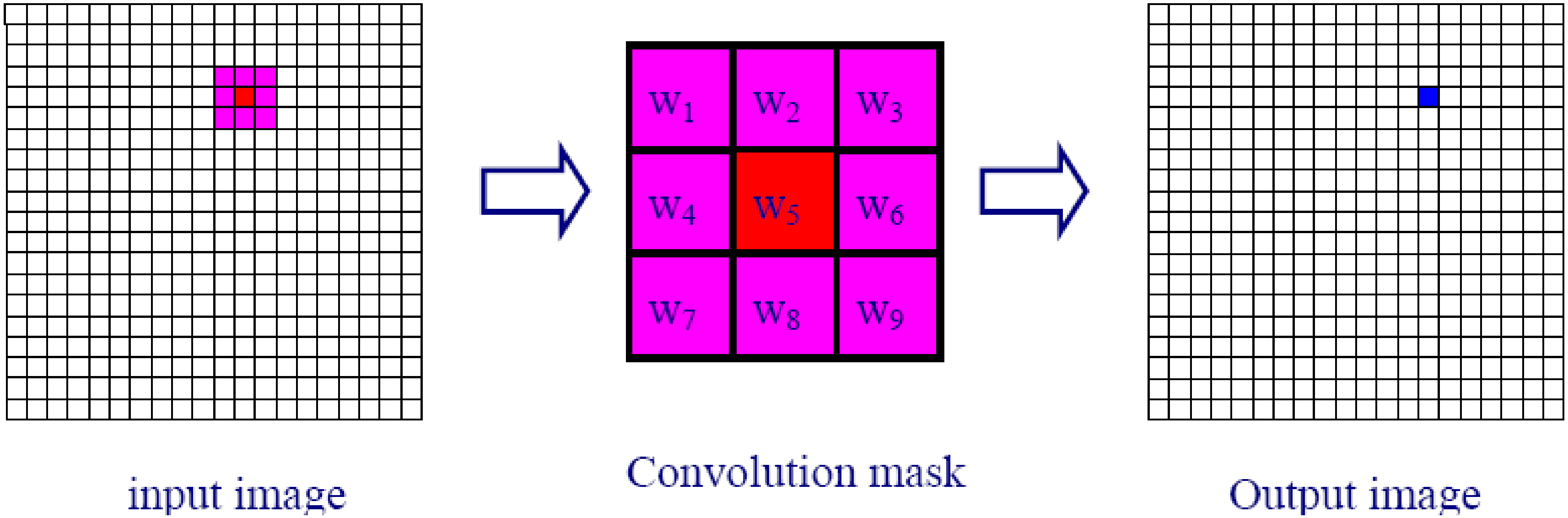
参考图

3.* Enhancement Using Arithmetic/Logic Operation (cont.)

3.4 Basics of Spatial Filtering（空间滤波基础）

- 滤波的概念来自信号处理中的傅里叶分析
- 空间滤波指的是直接对图像像素进行处理的操作
- 大致分为线性和非线性两种情形
- 对像素灰度值的调整要利用该像素周围的像素信息
- 滤波器（filter）的概念。滤波器有时也叫掩模（mask）、核（kernel）、模板（template）或窗口（window）

- **Most filters can be represented by convolution masks of dimension 3X3, 5X5, 7X7, 9X9, 11X11, etc. But 3X3 is the most common form.**



$$h(i,j) = w_1 p_1 + w_2 p_2 + w_3 p_3 + w_4 p_4 + w_5 p_5 + w_6 p_6 + w_7 p_7 + w_8 p_8 + w_9 p_9$$

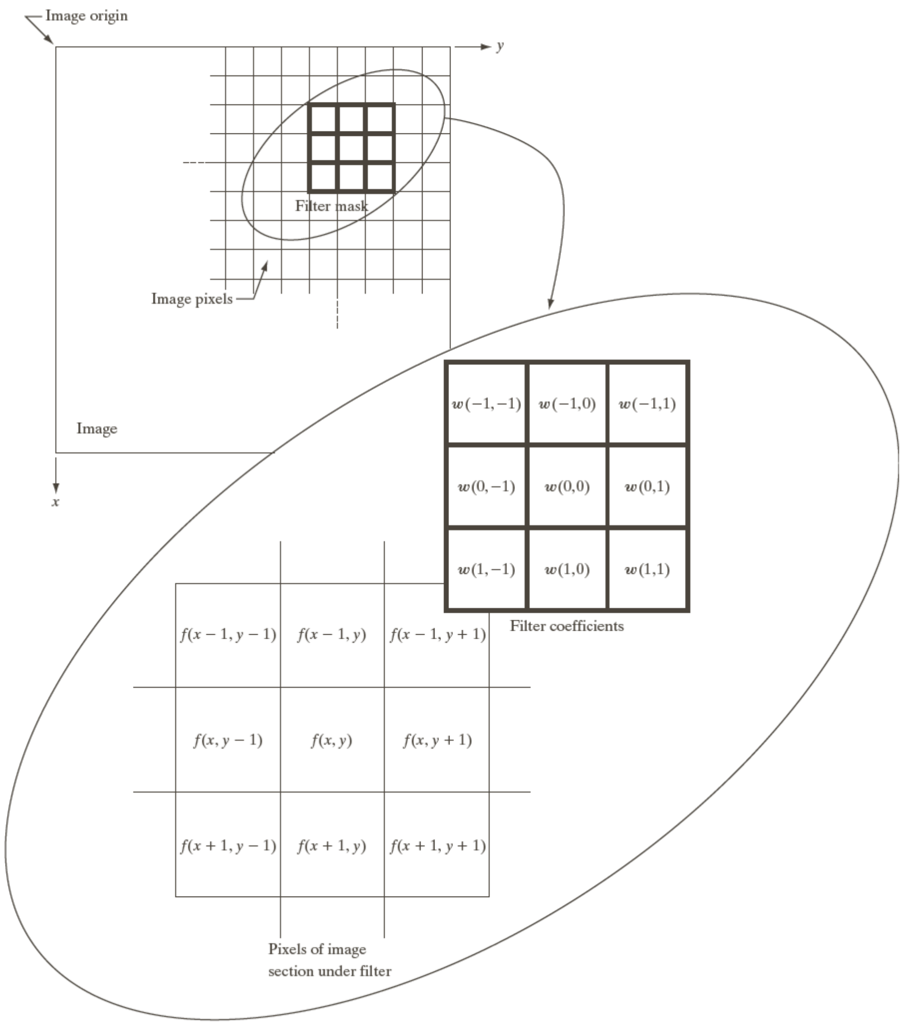


FIGURE 3.28 The mechanics of linear spatial filtering using a 3×3 filter mask. The form chosen to denote the coordinates of the filter mask coefficients simplifies writing expressions for linear filtering.

浮雕

$$\begin{bmatrix} -1 & 0 & -1 \\ 0 & 4 & 0 \\ -1 & 0 & -1 \end{bmatrix}$$

➢ Spatial Correlation and Convolution

similar but different

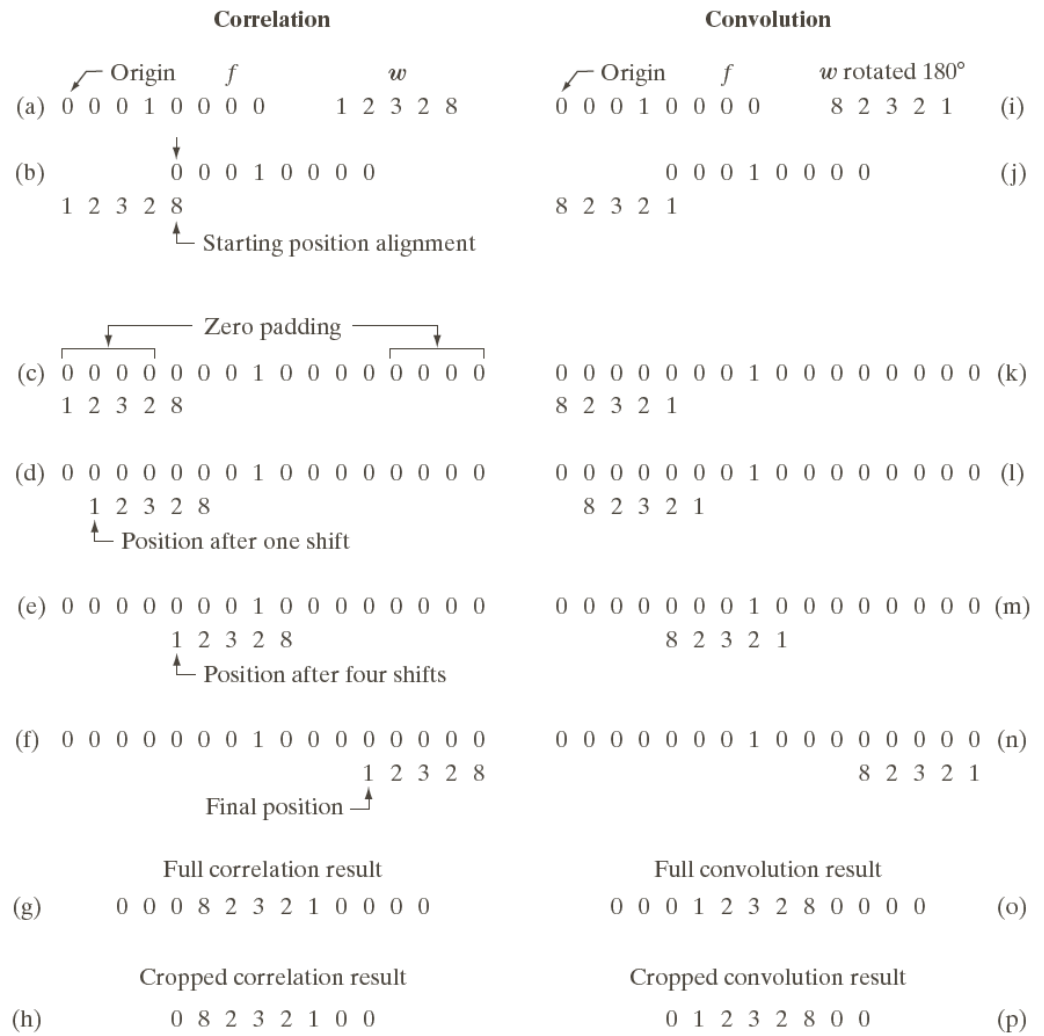


FIGURE 3.29 Illustration of 1-D correlation and convolution of a filter with a discrete unit impulse. Note that correlation and convolution are functions of *displacement*.

Spatial Correlation and Convolution

2D

Origin $f(x, y)$

0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0

$w(x, y)$

1	2	3
4	5	6
7	8	9

(a)

Padded f

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(b)

Initial position for w

1	2	3	0	0	0	0	0	0
4	5	6	0	0	0	0	0	0
7	8	9	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(c)

Full correlation result

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	9	8	7	0	0	0
0	0	0	6	5	4	0	0	0
0	0	0	3	2	1	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(d)

Cropped correlation result

0	0	0	0	0
0	9	8	7	0
0	6	5	4	0
0	3	2	1	0
0	0	0	0	0

(e)

Rotated w

9	8	7	0	0	0	0	0	0
6	5	4	0	0	0	0	0	0
3	2	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(f)

Full convolution result

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	1	2	3	0	0	0
0	0	0	4	5	6	0	0	0
0	0	0	7	8	9	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(g)

Cropped convolution result

0	0	0	0	0
0	1	2	3	0
0	4	5	6	0
0	7	8	9	0
0	0	0	0	0

(h)

FIGURE 3.30 Correlation (middle row) and convolution (last row) of a 2-D filter with a 2-D discrete, unit impulse. The 0s are shown in gray to simplify visual analysis.

Spatial Correlation and Convolution

2D

Correlation operation

$$w(x,y) \star f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

Convolution operation

$$w(x,y) \bigstar f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)$$

3.5 Smoothing Spatial Filtering

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

3X3 box filter

$$\frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

weighted averaging filter

- **Smoothing filter is also called an 'averaging filter' or a 'low-pass filter'.**
- **The sum of all elements in the mask is "1" to avoid dc bias.**
- **When all elements in the mask are equal, it is called a 'box filter'.**
- **Averaging filter has the side effect of blurring.**

- 平滑的主要作用是去除细小的细节（噪声）提取大的目标，从而得到感兴趣物体的一个大致描述
- 平滑滤波器分线性和非线性两种

- **Averaging filter**

“平均”包括加权平均是最常见的平滑方式，例如有一组点

0，4，5，1，0，0

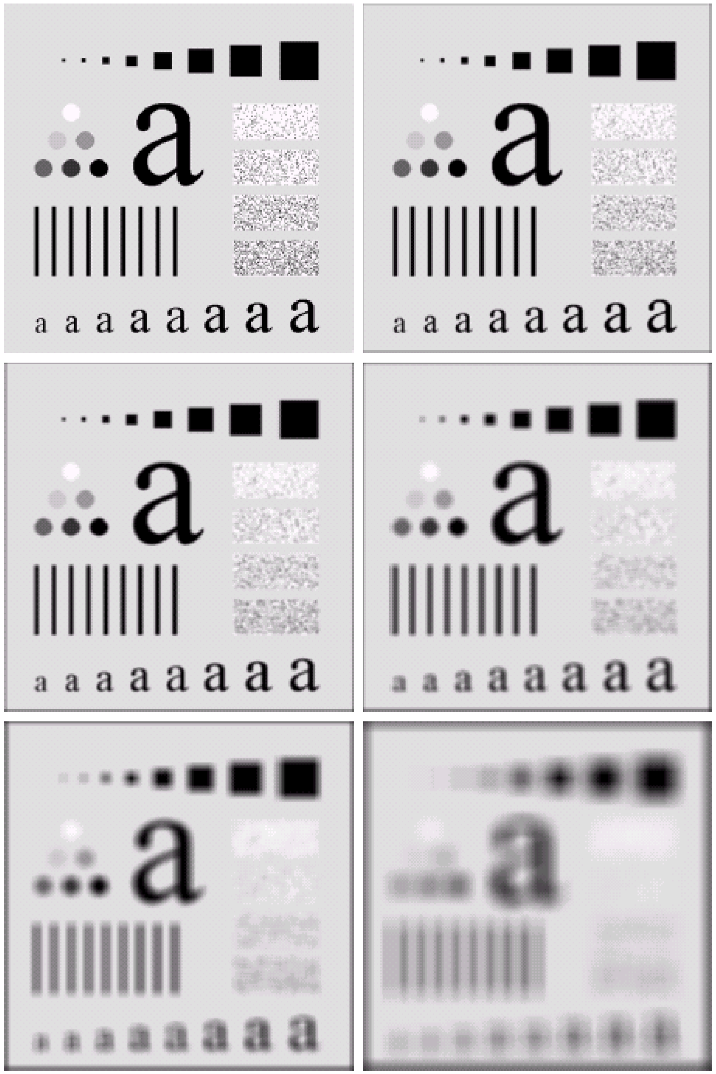
经过最简单的平均算法，例如每一点和相邻的两点做平均后替换原来的值，得到：

2，3，10/3，2，1/3，0

加权平均（1/4，1/2，1/4）：

1，3.25，3.75，1.75，0.25，0

➢Example of Averaging filter



Left: Original image,

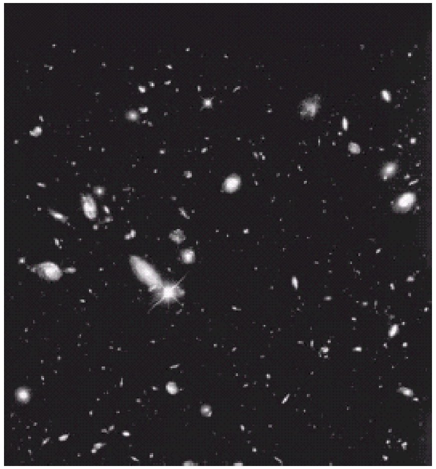
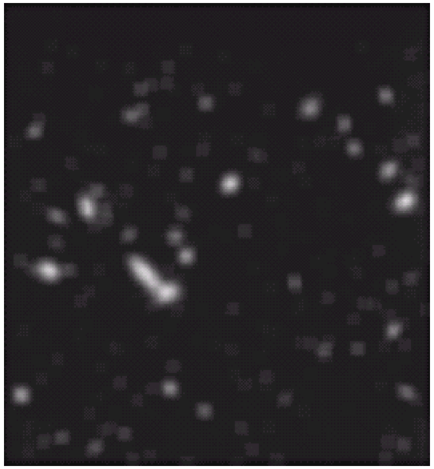
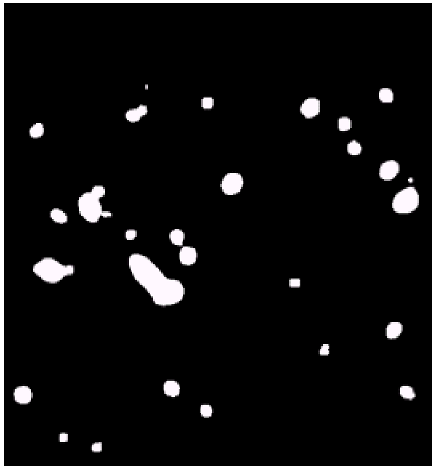
Right: Smoothing by 3×3 mask,

Below: Smoothing by masks of

5×5, 9×9, 15×15, 35×35,

In these pictures, the size of the black square on the top is 3, 5, 9, 15, 25, 35, 45, and 55 pixels.

Figure 3.36 Image from Hubble space telescope: 提取感兴趣目标的例子

Original image Averaging by 15x15 After thresholding

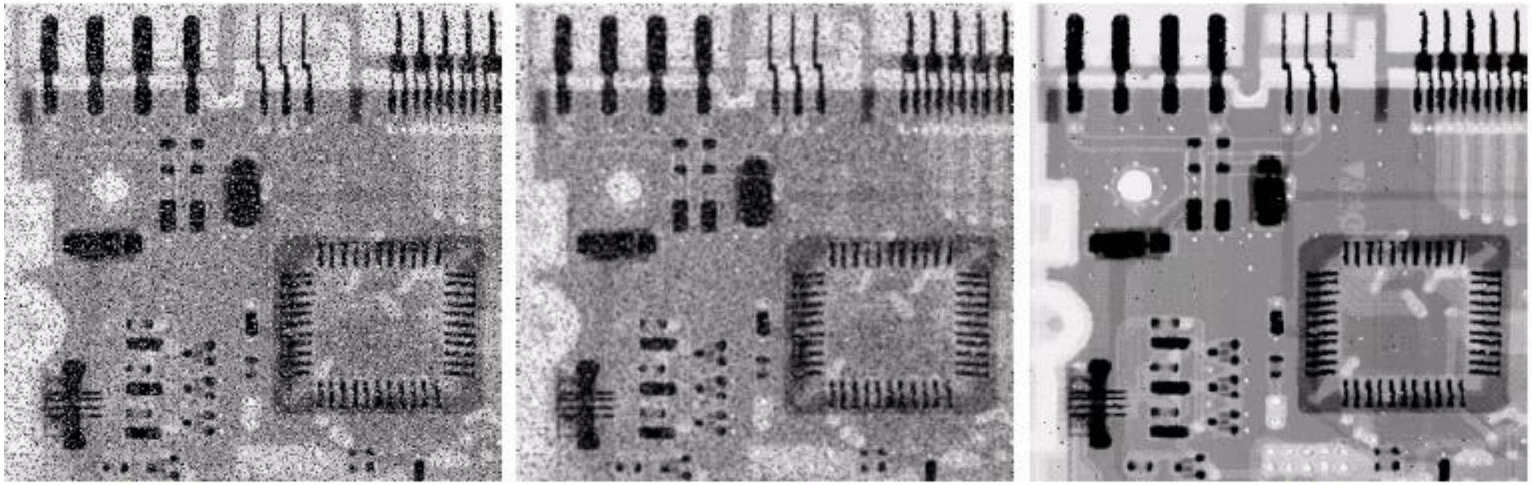
- **Median Filter (Order statistic filter)**

➢ The output intensity is the middle value of all pixel values in the mask.

10	20	20
20	100	20
20	25	15

$$\{z_k\}=\{10,20,20,20,100,20,20,25,15\}$$
$$=\{10,15,20,20,20,20,20,25,100\}$$
$$median\{z_k\}=20$$

➢ Median filter can remove salt-and-pepper noise without blurring.

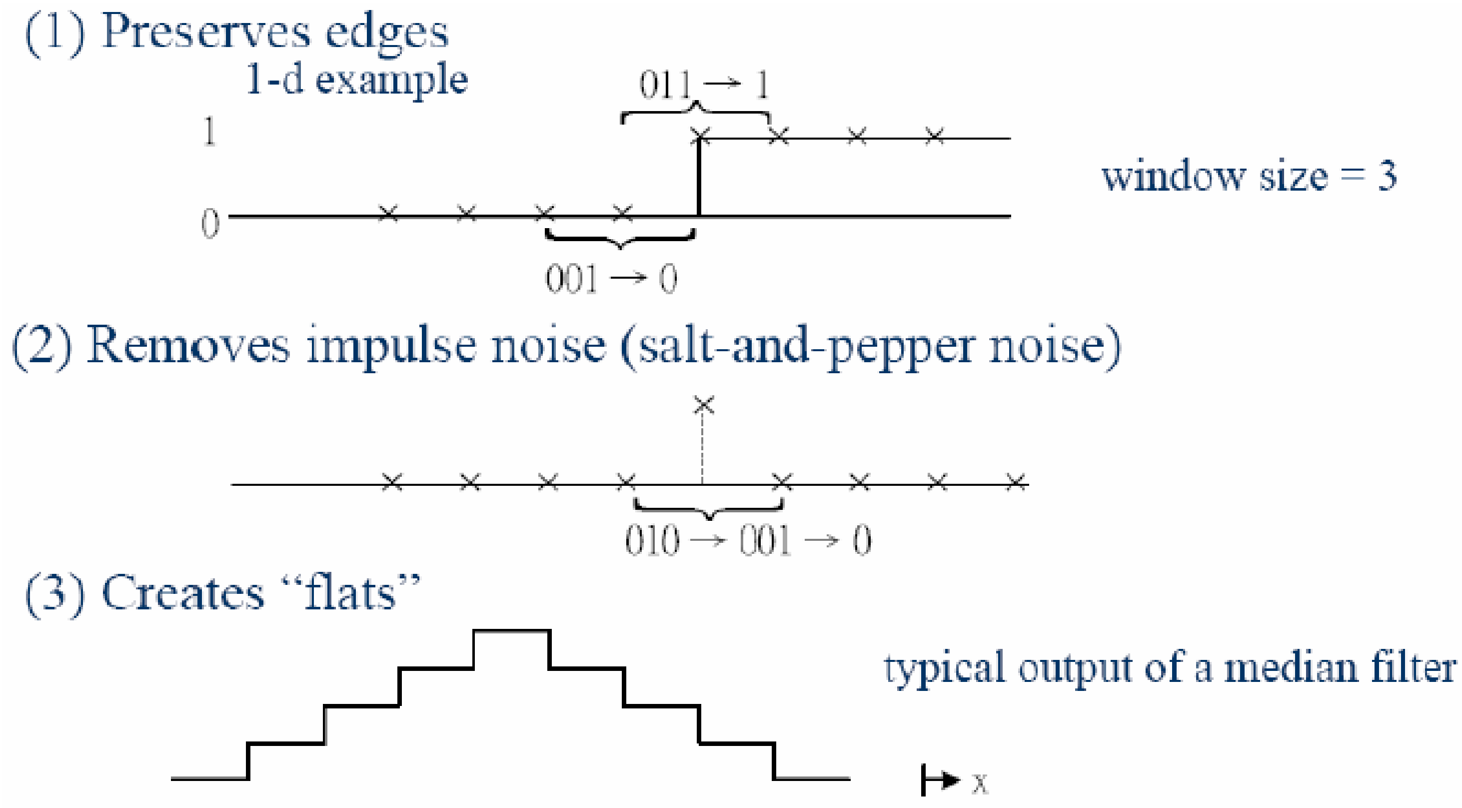


Original image averaging filtering median filtering


@微摄友
weibo.com/weisheyou


@一坨吃饱喝足的猫先森
weibo.com/u/2683130623

➢ Properties of median filter



➢ Max Filter and min filter (Other order statistic filter)

✓ The output intensity is the max (min) value of all pixel values in the mask.

1、将相机对着风景，放在三脚架上；

2、设置相机为每10秒拍一张，共拍15张左右；

3、用Photoshop的“文件(File)>脚本(Scripts)>数值(Statistics)”的“中间值(median)”打开这15张图片；

4、神器Photoshop将会找出这些图片的不同，然后轻松的去掉他们！给你一张纯风景！

Question：

- **Why?**
- **In what condition it works?**

3.6 Sharpening Spatial Filters

Sharpening is achieved by highlighting details (edges) using derivatives or edge detectors.

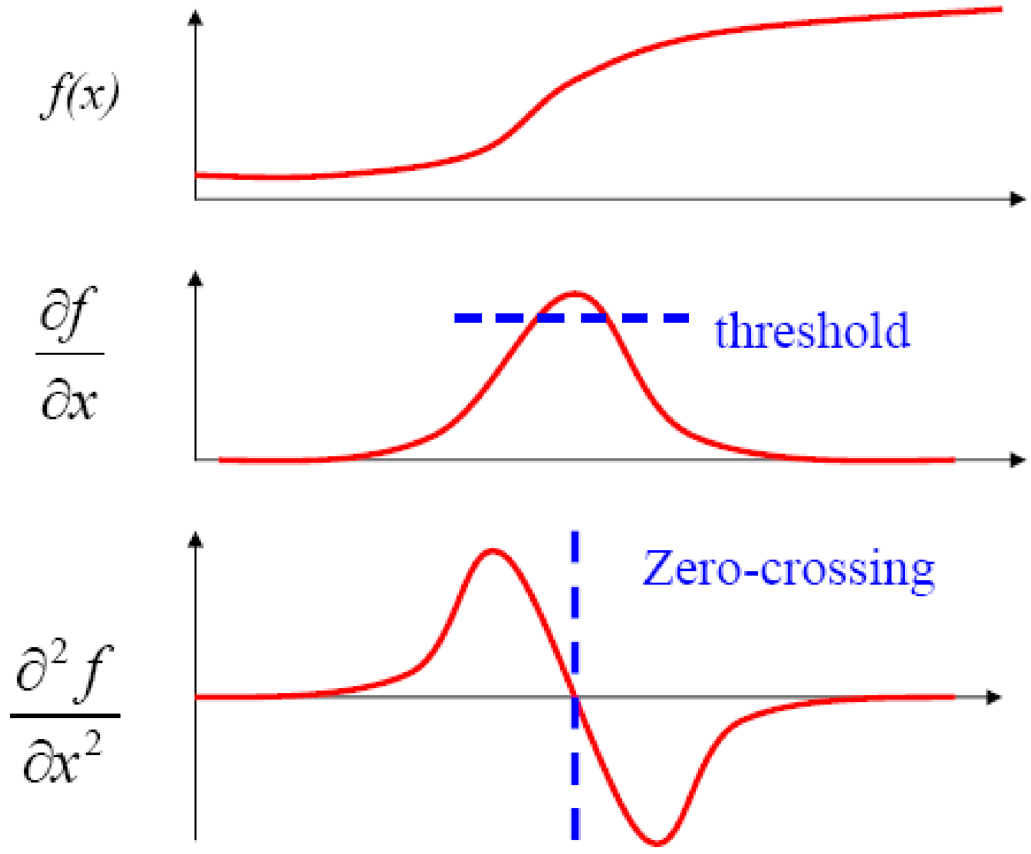
● **Foundation**——Finite difference approximation to derivatives

✓ Finite difference approximation to first order derivative

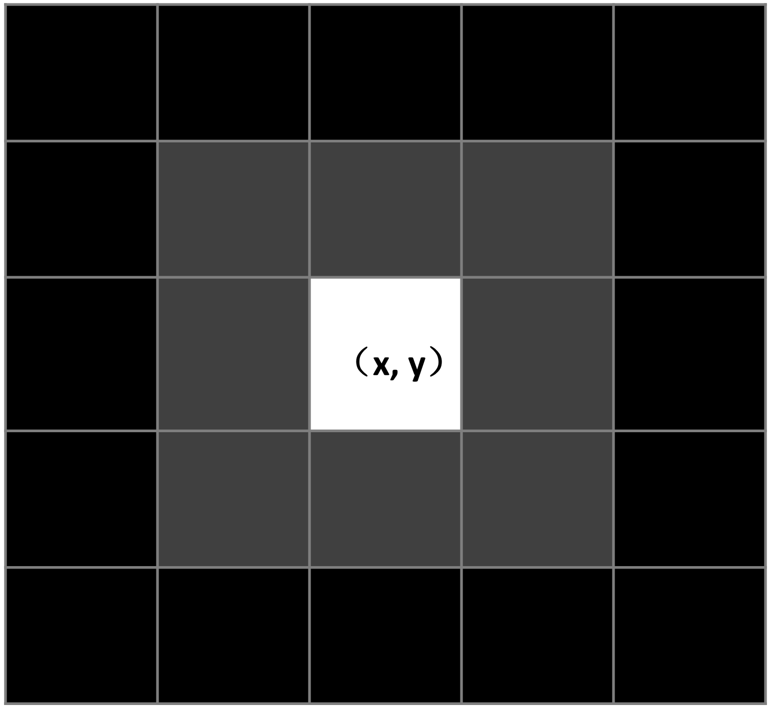
$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

✓ Finite difference approximation to second order derivative

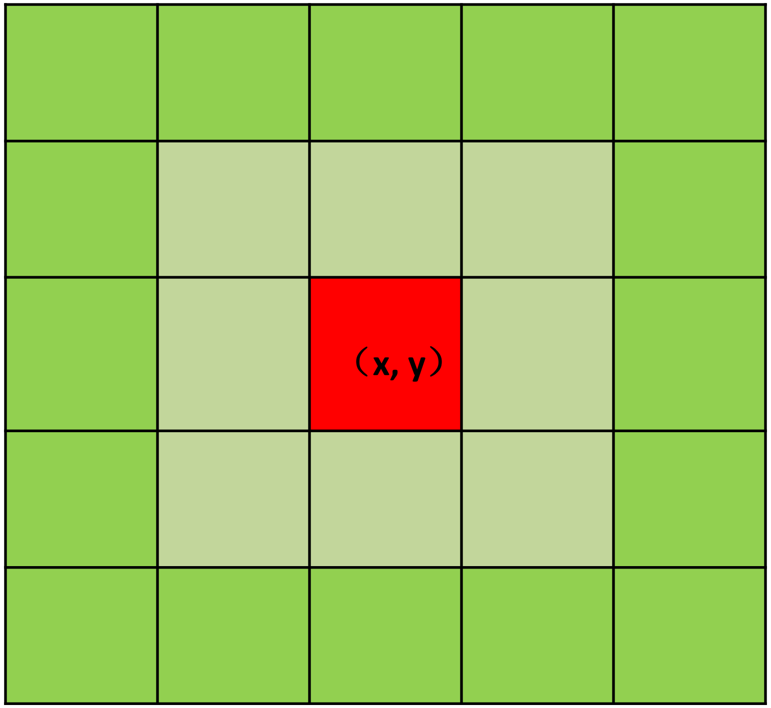
$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$



$f(x)$
$\frac{\partial f}{\partial x}$
threshold
$\frac{\partial^2 f}{\partial x^2}$
Zero-crossing



The value difference between pixel (x, y) and other pixels give us different meaning according to the sharpness. Different ways to subtract the values of pixels also makes differences.

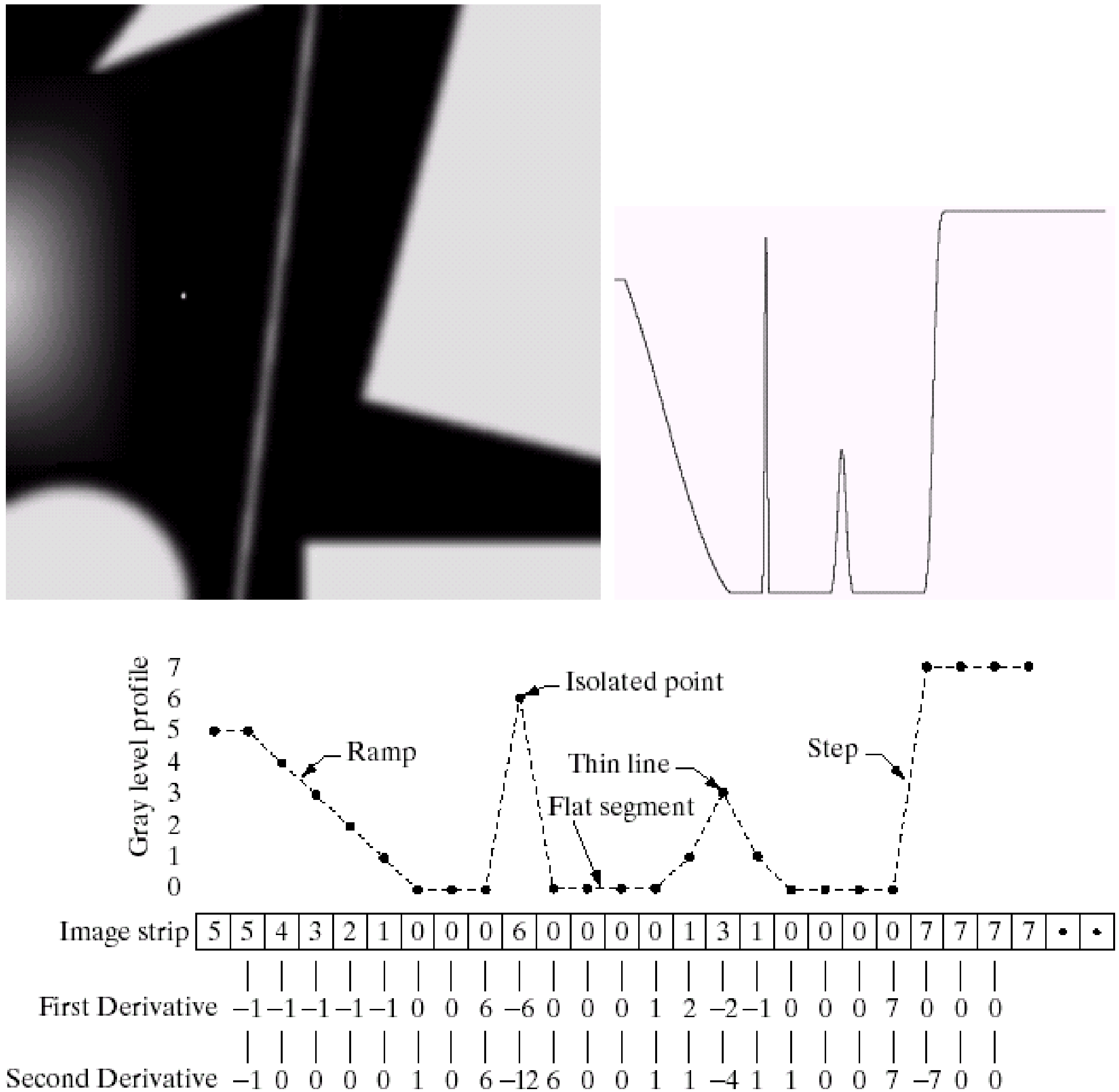


The value difference between pixel (x, y) and other pixels give us different meaning according to the sharpness. Different ways to subtract the values of pixels also makes differences.

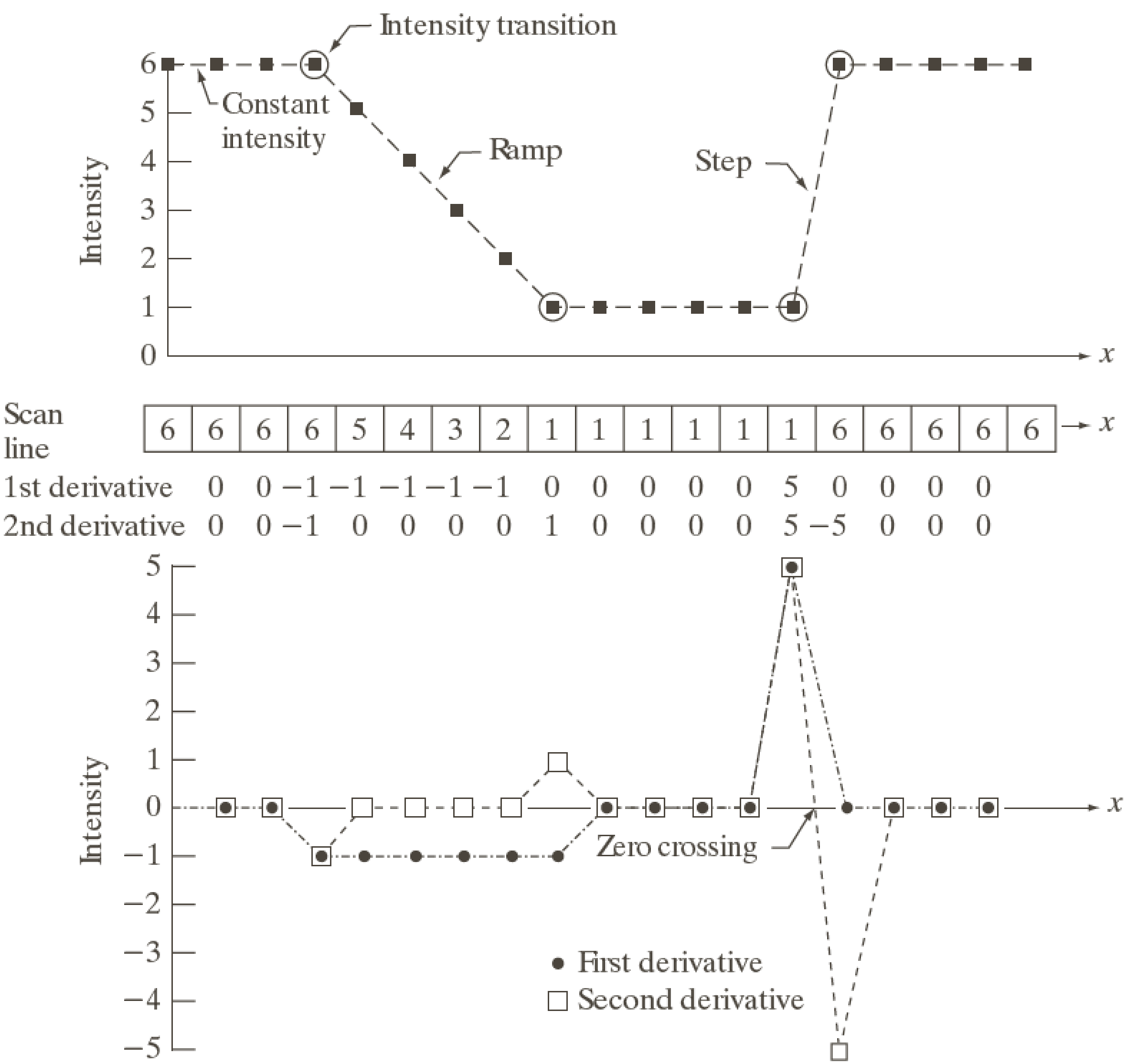
一个简单例子说明图像不同部分对微分操作的响应

a b
c

FIGURE 3.38
(a) A simple image. (b) 1-D horizontal gray-level profile along the center of the image and including the isolated noise point. (c) Simplified profile (the points are joined by dashed lines to simplify interpretation).



一个简单例子说明图像不同部分对微分操作的响应



a
b
c

FIGURE 3.36 Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image. In (a) and (c) data points are joined by dashed lines as a visualization aid.

结论

- 一阶微分产生较“宽”的边界，二阶微分产生较“细”的边界；
- 二阶微分处理对细节有较强的响应，如细线和孤立点；
- 一阶微分对阶梯状的灰度变化有较强的响应；
- 二阶微分在处理阶梯状灰度变化时产生双响应
- 如果灰度的变化相似，二阶微分对线的反应比对阶梯强，对点的反应比对线强。

由于形成增强细节的能力较强，对一般图像处理的应用来说，二阶微分处理要比一阶微分处理好一些。但不绝对。

- **Laplacian filter (2nd derivative):**

The Laplacian of a 2D function *f(x, y)* is

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Its finite difference approximation is given by

$$\begin{aligned}\nabla^2 f(x,y) &= \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \\ &\cong \left[f(x+1,y) - 2f(x,y) + f(x-1,y)\right] \\ &\quad + \left[f(x,y+1) - 2f(x,y) + f(x,y-1)\right] \\ &= f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)\end{aligned}$$

which can be implemented in the following convolution mask

$$\nabla^2 f(x,y) \cong \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\nabla^2 f(x,y) \cong \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

A higher-order approach

- **Laplacian filter (2nd derivative):**

The Laplacian operator can be represented by one of the following filters:

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

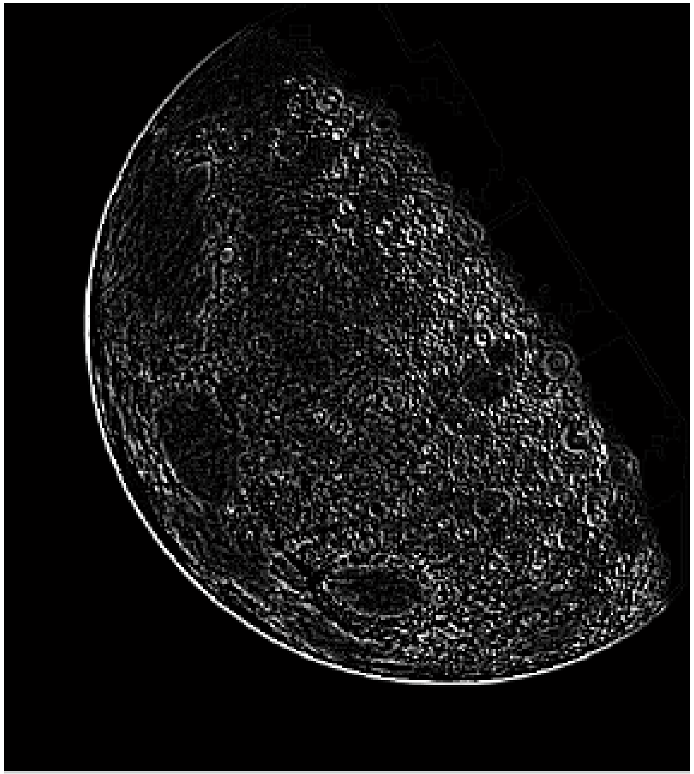
−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37
(a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

- **Edge detection by Laplacian filter (2nd derivative)**

- **Comparison of high order and low order Laplacian filter**



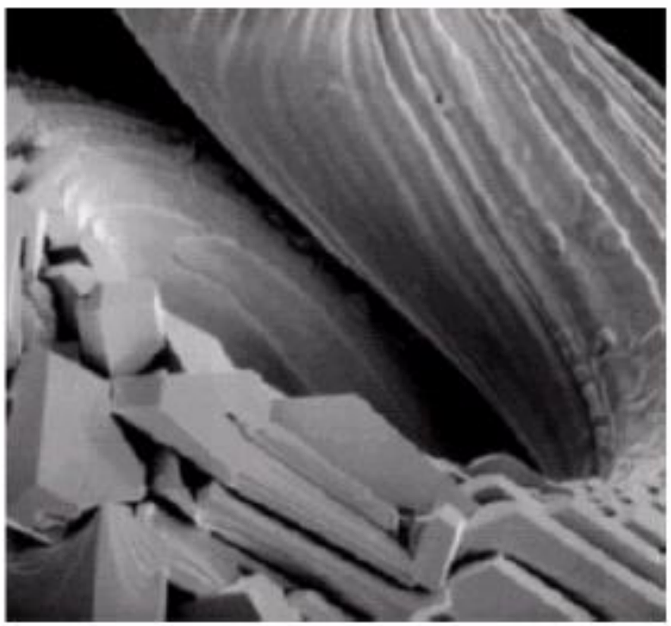
$$\nabla^2 f(x,y) \cong \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\nabla^2 f(x,y) \cong \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

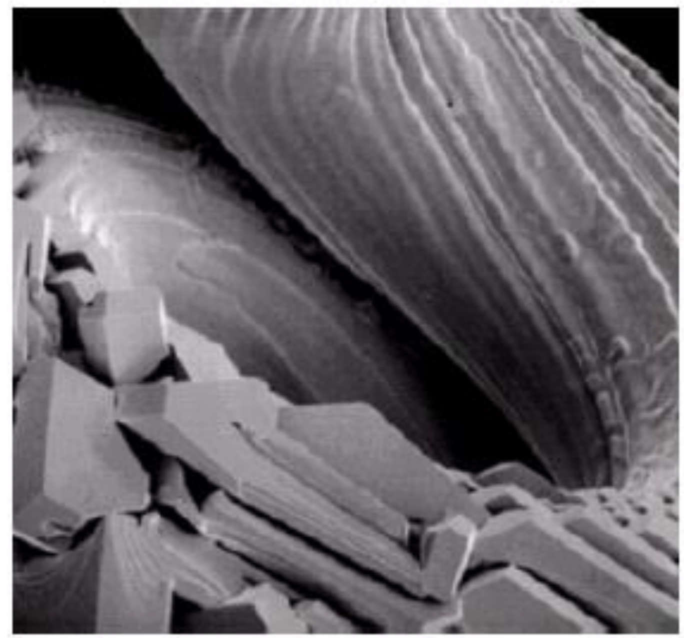
●Enhancement by Laplacian filter

Enhanced image = Original image + Edge image

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



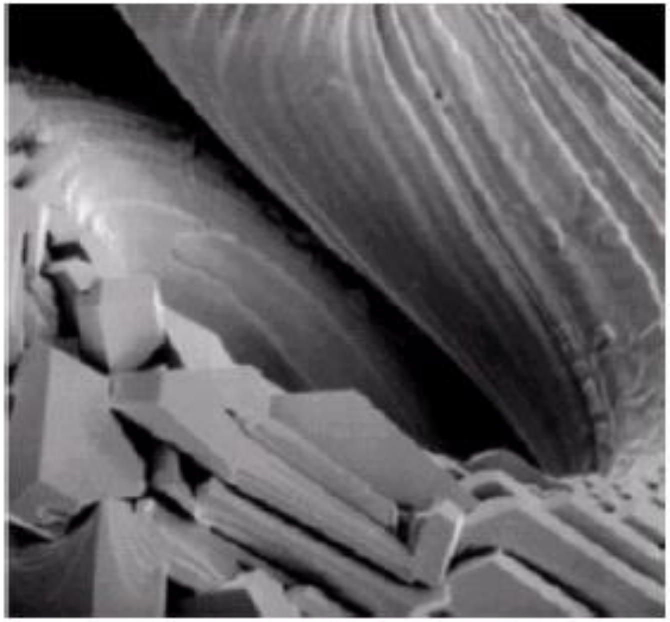
Original image



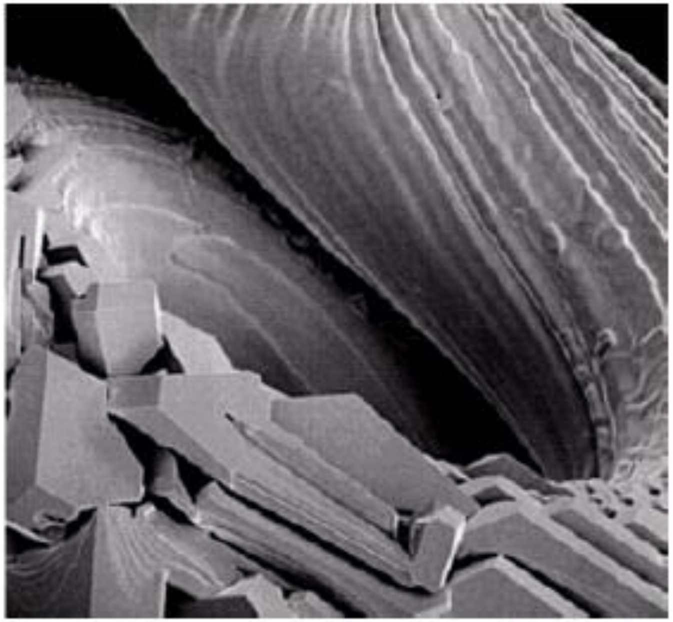
Enhanced image

When using a second order approximation of Laplacian, it becomes

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



Original image



Enhanced image

●Unsharp Masking and High-boost Filter

Unsharp masking processing steps:

1. Blur the original image
2. Subtract the blurred image from the original (the resulting difference is called mask
3. Add the mask to the original

Blurred image

$$g_{mask}(x,y) = f(x,y) - \bar{f}(x,y)$$

Add a weighted portion of mask back to the original image

$$g(x,y) = f(x,y) + k\, g_{mask}(x,y)$$

Here $k \geq 0$. When $k = 1$, we have unsharp masking as defined above. If $k > 1$, we got so called high-boost filter. k<1 de-emphasizes the contribution of the unsharp mask.

另一个角度看：Unsharp masking and high-boost filter

High boost filter = B • Original image – low-pass filtered image

= (B-1) • Original image + (Original image – low-pass filtered image)

= (B-1) • Original image –high-pass filtered image

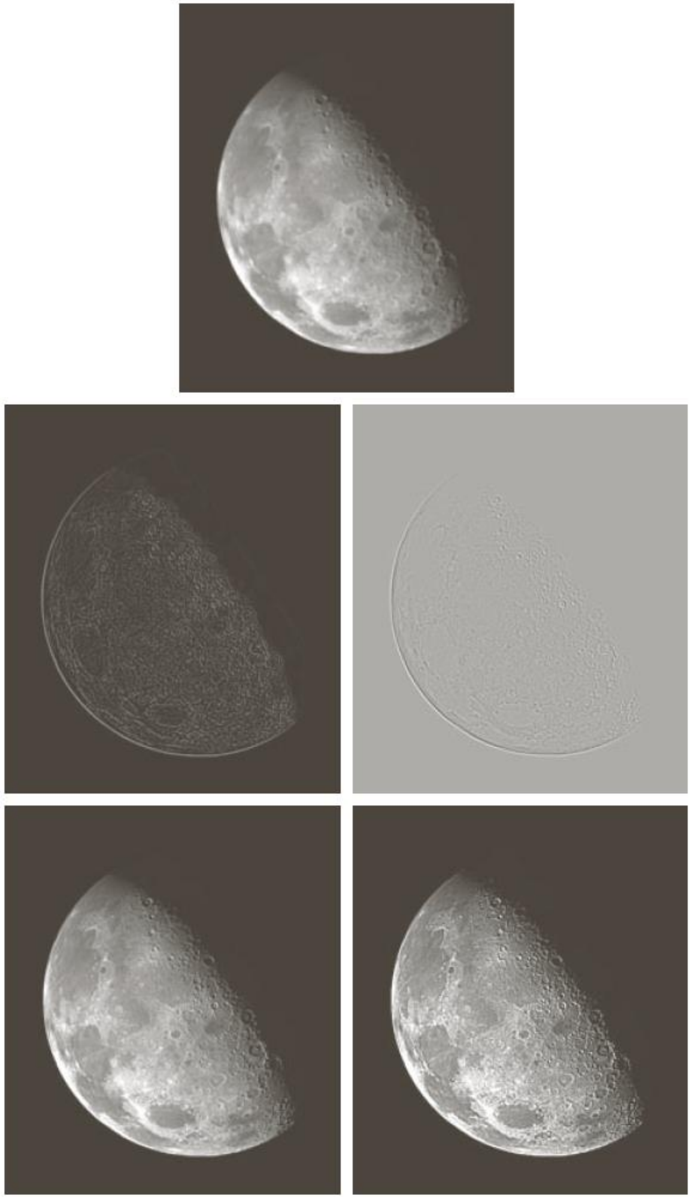
Therefore, it has the following general form

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & A+4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & A+8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

that is equivalent to Laplacian enhancement but with a weighting on the original Image.

公式的物理意义就是把原图的一个模糊过的图像从原图中减去，从而得到一个相对清晰的图像（就象一个模糊的负片和一个正片放在一起冲洗出相对清晰的照片）。更进一步的普遍形式就是所谓的高提升滤波处理。

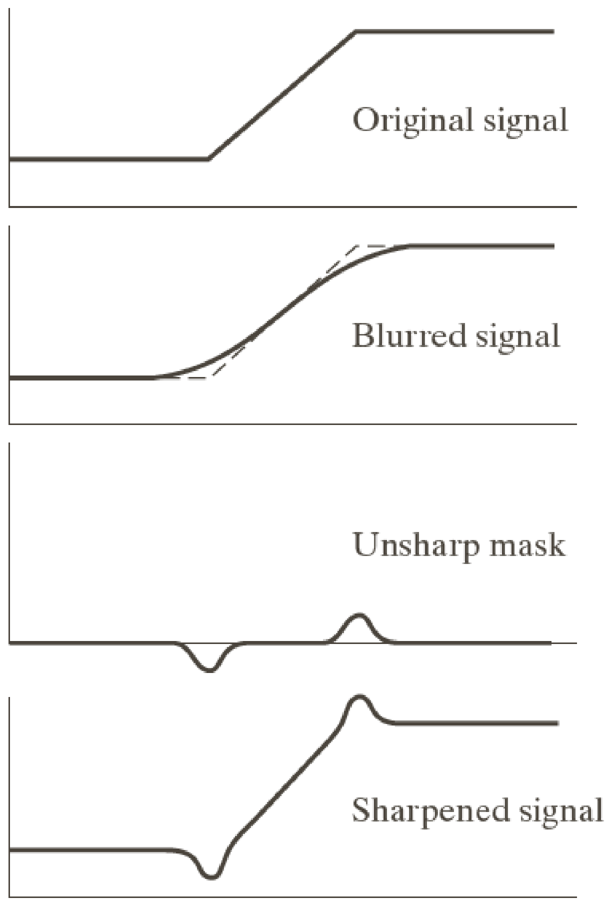
Image enhancement with high-boost filtering based on Laplacian



- Blur the original image
- Subtract the Laplacian

a
b c
d e

FIGURE 3.38 (a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the mask in Fig. 3.37(a). (e) Result of using the mask in Fig. 3.37(b). (Original image courtesy of NASA.)



a
b
c
d

FIGURE 3.39 1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

- Blur the original image
- Subtract the blurred image from the original
- Add the resulting mask to the original

- $g(x, y) = f(x, y) + k\, g_{mask}$

- $k = 1$ unsharp masking
- $k > 1$ highboost filtering

3.6 Sharpening Spatial Filters (cont.)

a
b
c
d
e

FIGURE 3.40 (a) Original image. (b) Result of blurring with a Gaussian filter. (c) Unsharp mask. (d) Result of using unsharp masking. (e) Result of using highboost filtering.

input

blurred

input - blurred

input + (input - blurred)

input + 4.5 (input - blurred)

● **Enhancement using First Derivatives (Sobel edge detector)**

$$\nabla f(x,y) = \left[\frac{\partial f(x,y)}{\partial x} \quad \frac{\partial f(x,y)}{\partial y} \right]^T$$

and its magnitude is often approximated by

$$|\nabla f(x,y)| = \sqrt{\left[\frac{\partial f(x,y)}{\partial x} \right]^2 + \left[\frac{\partial f(x,y)}{\partial y} \right]^2}$$

◆ In order to overcome the computational complexity, we may in some occasion to use L_1 norm to reduce the load without any effects on the results.

$$\boxed{M(x,y) = |\nabla f| \approx |g_x| + |g_y|}$$

注意：通常在不引起混淆的情况下，把梯度的模称为梯度。

另一个问题：如何计算离散图像数据的梯度？

一种算法是2×2的掩模上的交叉差分（Robert交叉梯度算子）：

$$g_x = (z_9 - z_5), \text{ and } g_y = (z_8 - z_6)$$

其梯度

$$\nabla f = |z_9 - z_5| + |z_8 - z_6|$$

或者在3×3的掩模上计算（Sobel算子）

$$\nabla f = |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$$

3.6 Sharpening Spatial Filters (cont.)

a
b c
d e

FIGURE 3.44 A 3 × 3 region of an image (the z's are gray-level values) and masks used to compute the gradient at point labeled z_5. All masks coefficients sum to zero, as expected of a derivative operator.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

3×3掩模上计算中心点z_5的梯度

−1	0
0	1

0	−1
1	0

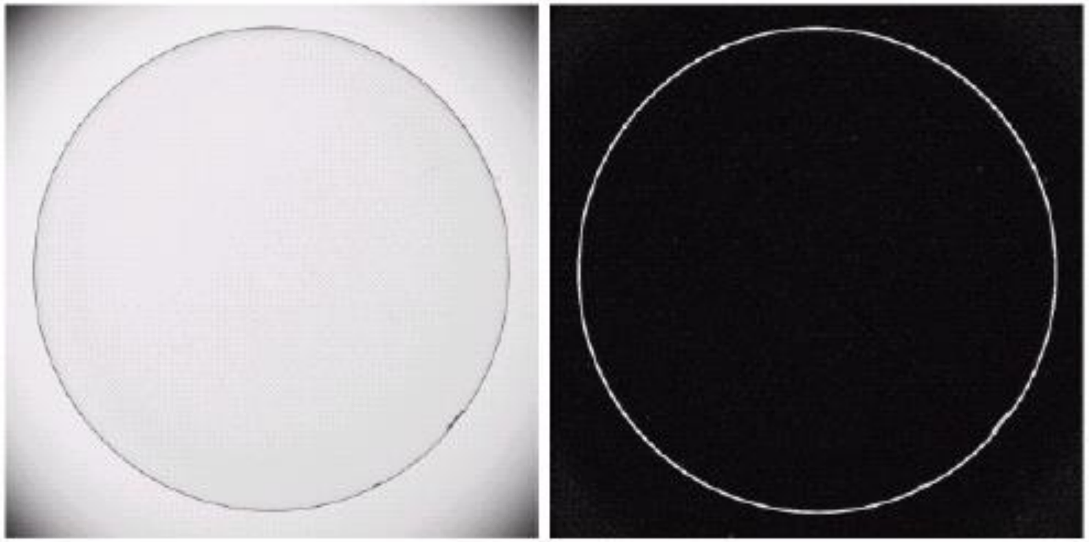
Robert梯度算子滤波器

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel edge detector

Please notice that, in G_X , G_Y and $|\nabla f(x,y)| \cong |G_X| + |G_Y|$, elements of masks always sum up to zero, which is essential for any edge detector.



Edge detection by Sobel operator

用于产品的检验，突出了边沿缺陷和灰度平坦区域突变（小斑点）。

3.7 混合空间增强法 （略）

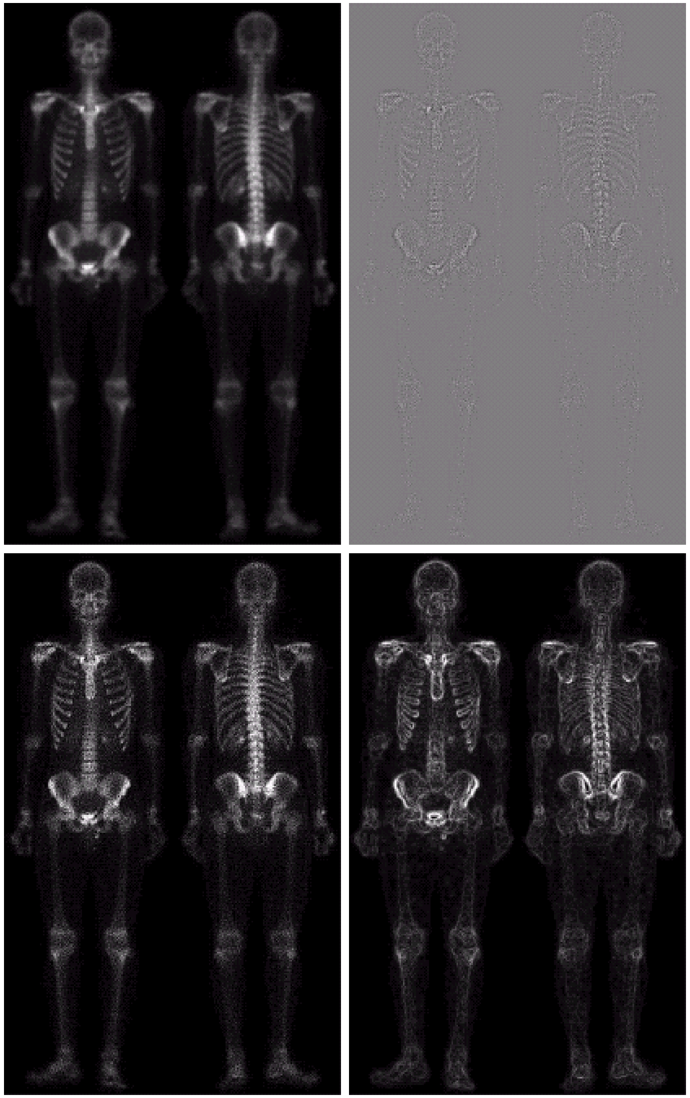
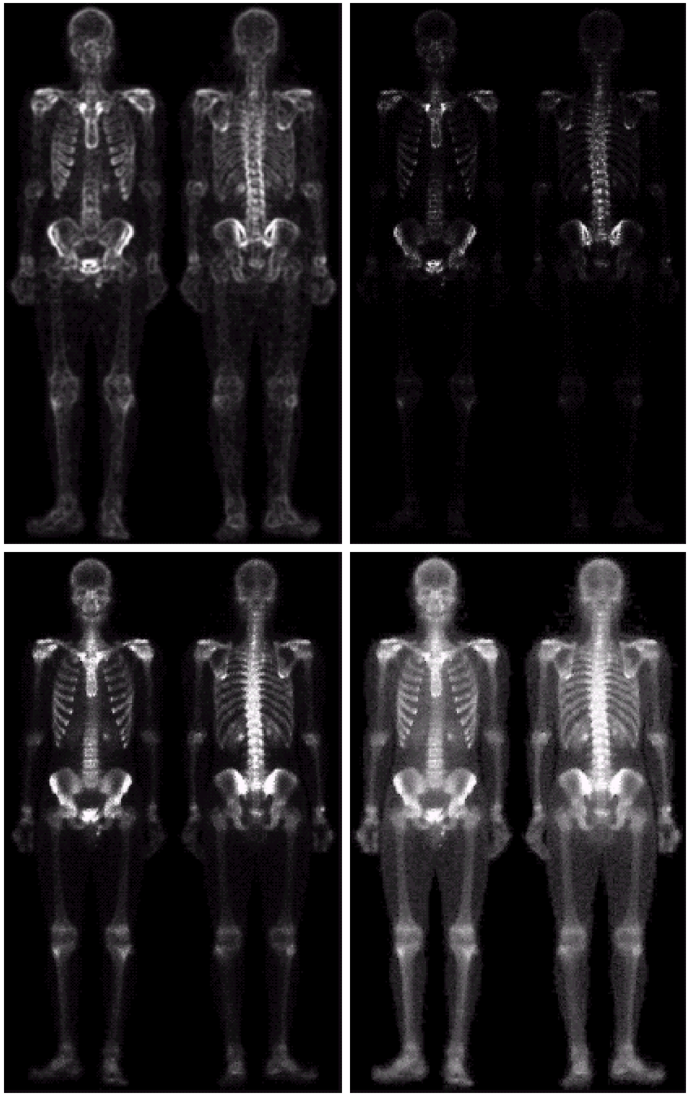


FIGURE 3.46
(a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel of (a).



e	f
g	h

FIGURE 3.46 *(Continued)* (e) Sobel image smoothed with a 5 × 5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

End of Chapter 3