

数字图像处理

指导教师： 胡晓雁

电子邮件： huxy@bnu.edu.cn

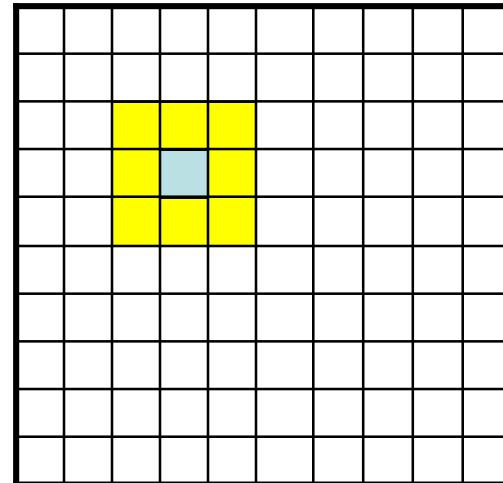
北京师范大学信息科学与技术学院

空间域增强

- ☒ 空间域增强是指增强构成图像的像素
- ☒ 空间域方法直接像素操作，输入一幅或多幅图像，然后输出处理的结果（图像）
 - ☒ $g(x,y) = T[f(x,y)]$
 - ☒ $g(x,y) = T[f_1(x,y), f_2(x,y), f_3(x,y) \dots]$
- ☒ T 的输入是像素 (x,y) 的邻域，输出是图像 g 中像素 (x,y) 的值

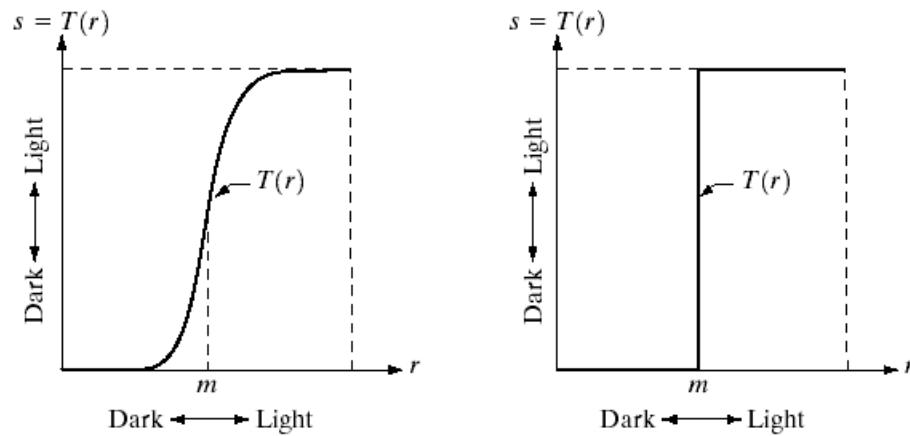
空间域增强

- 点 (x,y) 的邻域一般是以 (x,y) 为中心的正方形或者矩形子图像
- 该子图像的中心从一个像素向另一个像素移动， T 利用邻域中的像素值计算得到输出像素

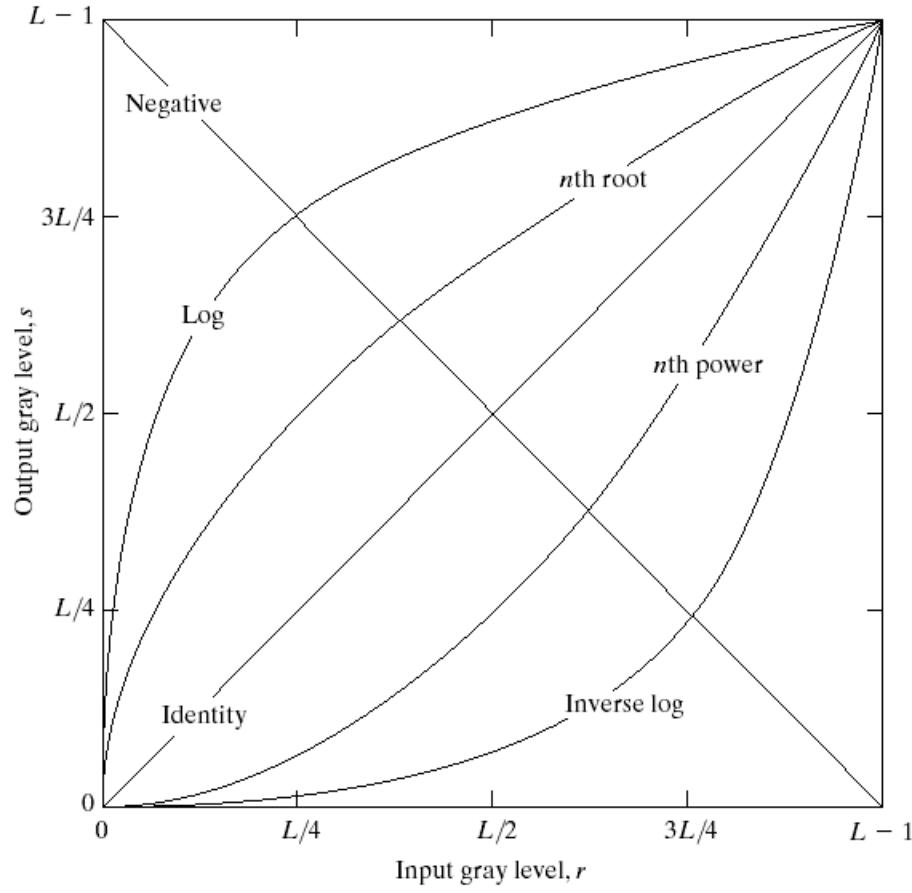


空间域增强

- 常用的是正方形邻域，一般以奇数大小为边长，如 1x1, 3x3, 5x5, 7x7
- $s = T(r)$



若干基本灰度变换



图像复制

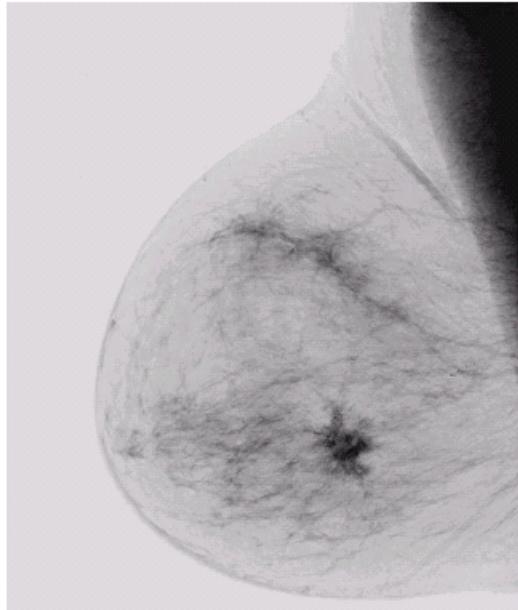
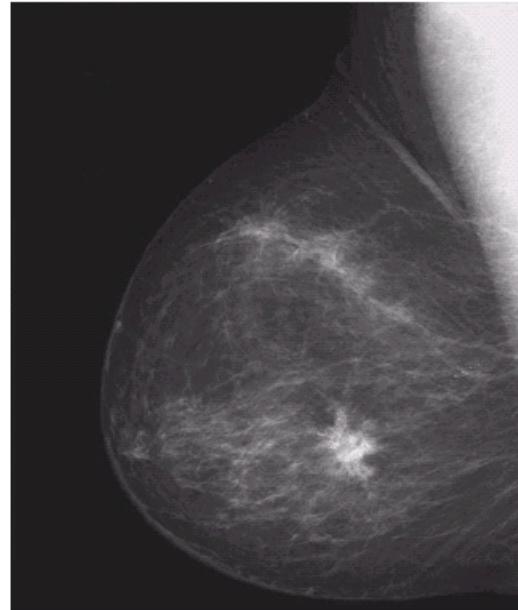
图像反转

幂次变换

对数变换

图像反转

- ✓ 灰度级范围为 $[0, L-1]$ 的图像反转
 - ✓ $s = L - 1 - r$
 - ✓ 使用于处理嵌入于图像暗色区域的白色或者灰色细节

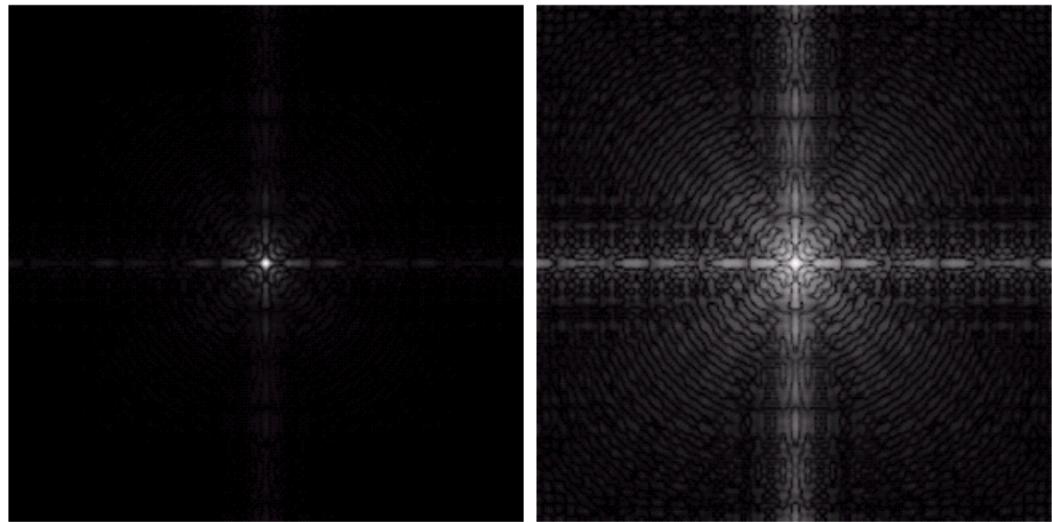


对数变换

- ✓ 对灰度级取对数，可以将高动态范围的图像压缩到低动态范围，以便更好地观察图像细节
- ✓ 高动态范围图像的一个问题是在一般的显示设备上无法显示如此大的动态范围，标准显示的动态范围只有255个灰度级别，而高动态范围能有多达 10^6 或更多的灰度级别

对数变换

- ✓ 下图是某图的傅里叶变换频谱，值范围是 $0 \sim 1.5 \times 10^6$
 - ✓ 左侧是线性变换到255范围内
 - ✓ 右侧是取对数变换



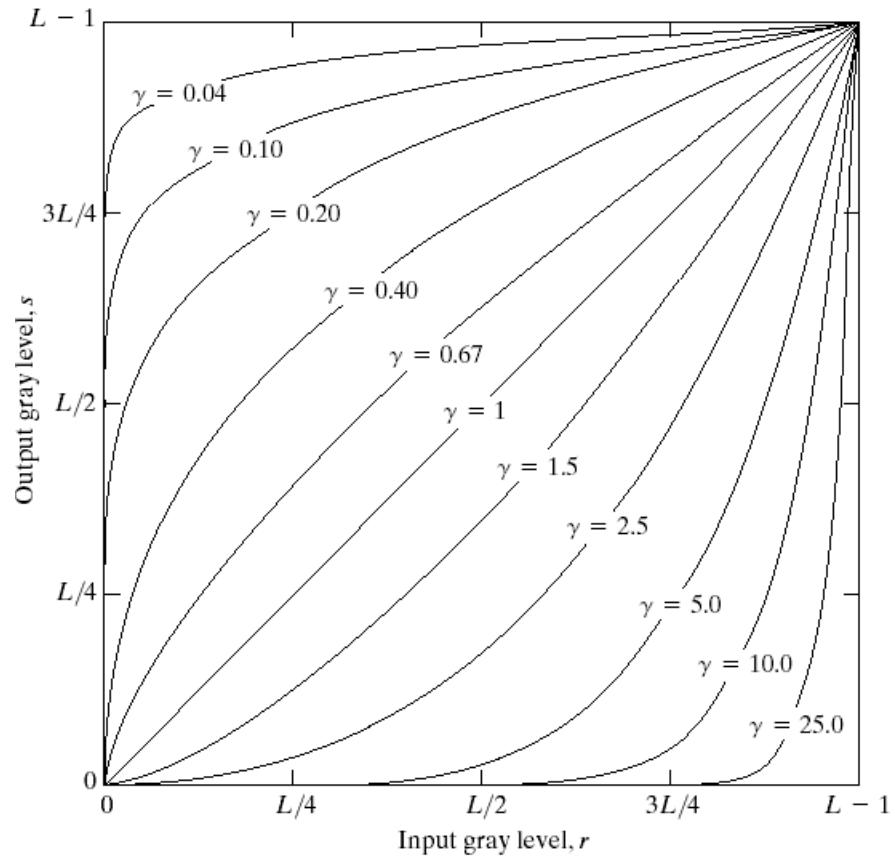
$$\checkmark S = c \log(1+r)$$

幂次变换

✓ 幂次变换也称为伽马变换

$$s = cr^\gamma \quad c, \gamma > 0$$

$$s = c(r + \varepsilon)^\gamma \quad c, \gamma > 0$$



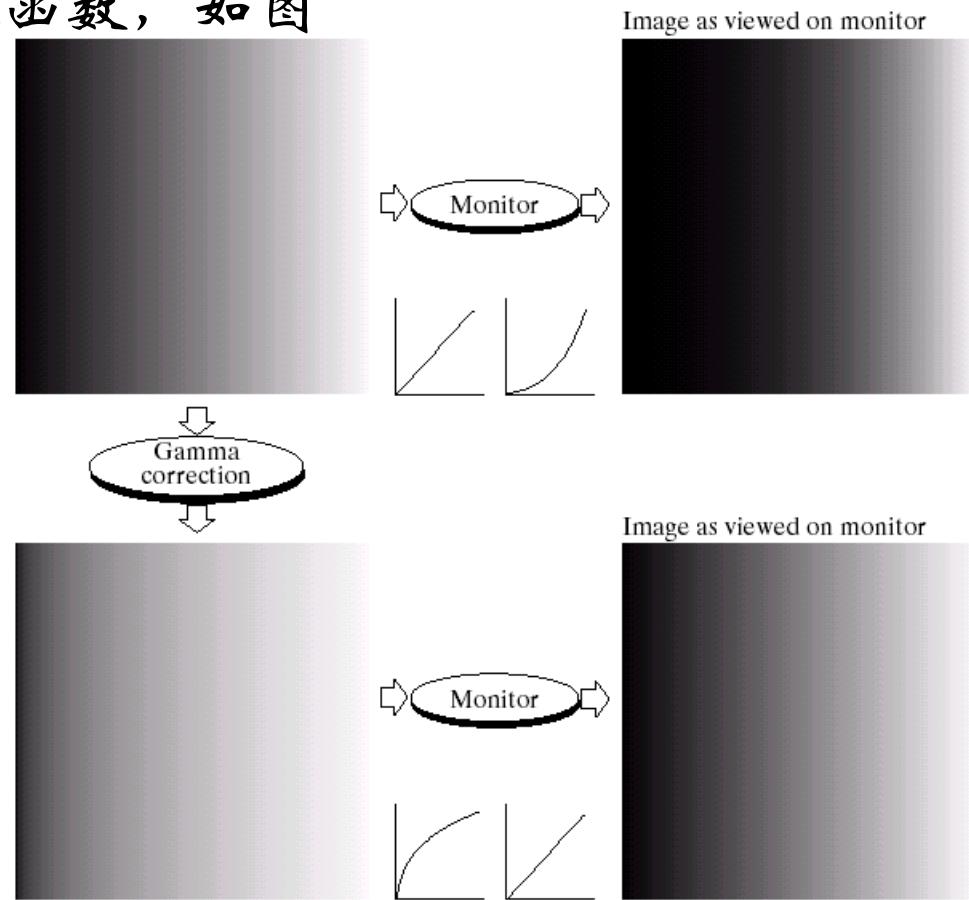
幂次变换

- ✓ 很多用于图像获取、打印和显示的各种装置根据幂次规律产生响应
- ✓ 不同的设备响应曲线不一样，修正幂次响应的过程称为伽玛校正

幂次变换

阴极射线管CRT装置的电压——强度响应，是一个指数变化范围在1.8~2.5的幂函数，如图

左边输入的是线性灰度图，CRT显示该图会较暗，如果在显示之前先进行伽玛校正，则CRT的显示会接近原图



幂次变换例1



a b
c d

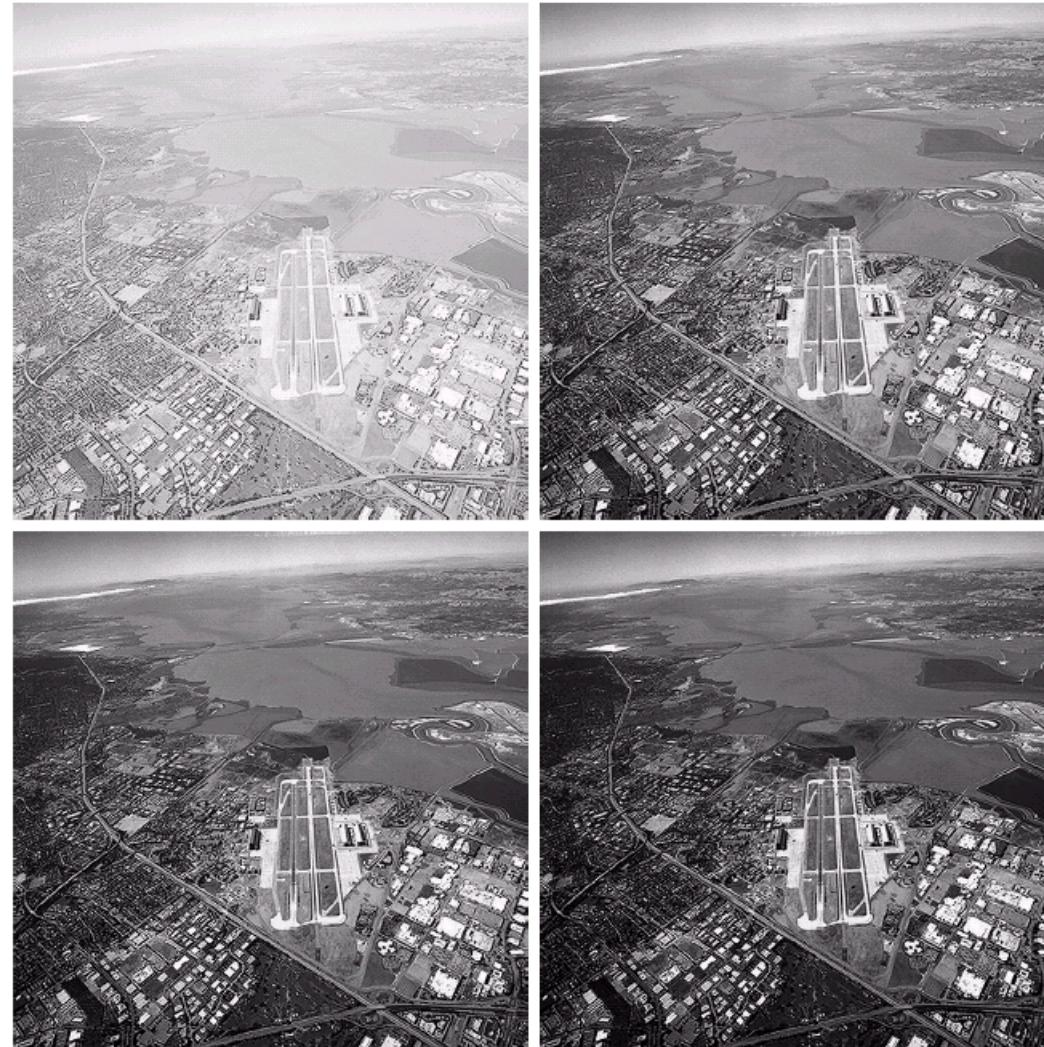
FIGURE 3.8
(a) Magnetic resonance (MR) image of a fractured human spine.
(b)-(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and $0.3,$ respectively.
(Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

幂次变换例2

a
b
c
d

FIGURE 3.9

(a) Aerial image.
(b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0$, and 5.0 , respectively. (Original image for this example courtesy of NASA.)



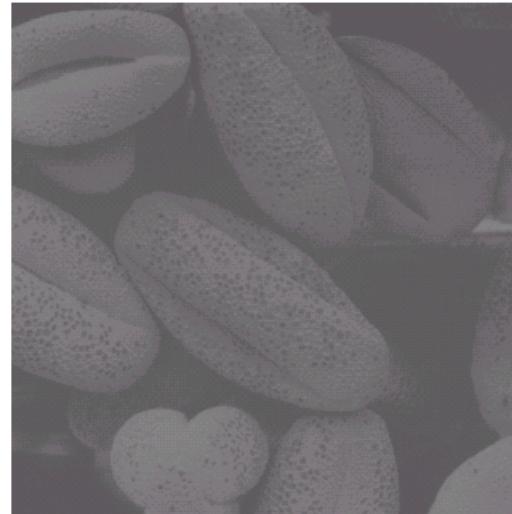
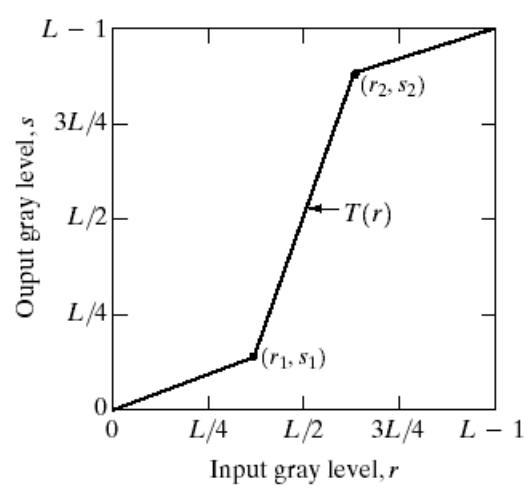
分段线性函数

- ✓ 对比拉伸
- ✓ 灰度切割
- ✓ 位图切割

对比拉伸

- ✓ 对比拉伸的思想是提高图像处理时灰度级的动态范围
- ✓ 点 (r_1, s_1) 和 (r_2, s_2) 的位置控制了变换函数的形状
- ✓ 若 $r_1=s_1, r_2=s_2$, 则变换为一线性函数, 恒等变换
- ✓ $r_1=r_2, s_1=0, s_2=L-1$, 则变为阈值函数

对比拉伸



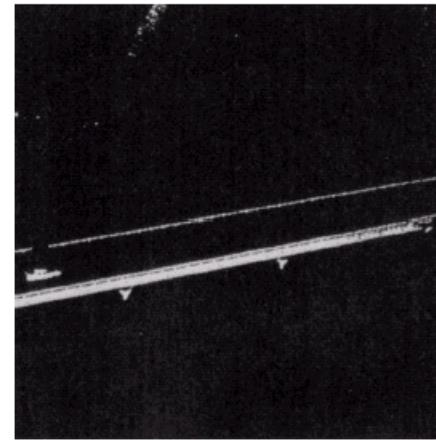
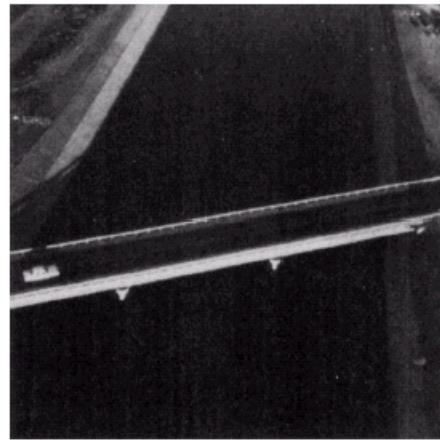
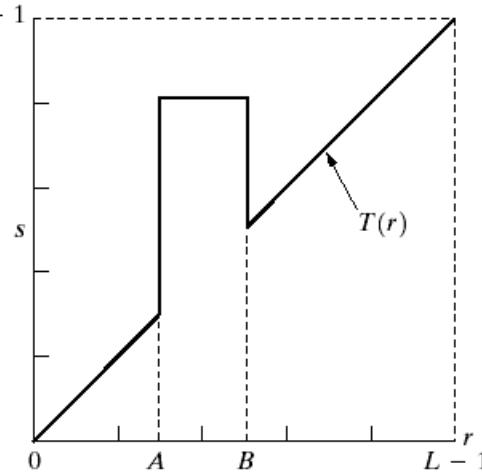
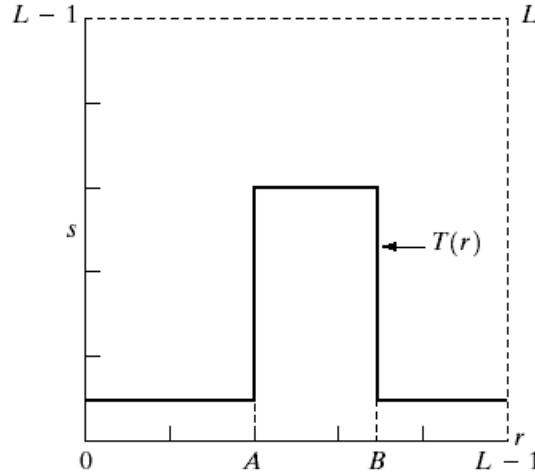
a
b
c
d

FIGURE 3.10
Contrast stretching.
(a) Form of transformation function. (b) A low-contrast image. (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.)

灰度切割

- ✓ 在图像中提高特定灰度范围的亮度，而同时降低其他灰度范围的亮度，以增强显示感兴趣的灰度范围区域（如卫星图像中的大量的水，或者是X射线图中的缺陷）

灰度切割



a	b
c	d

FIGURE 3.11
(a) This transformation highlights range $[A, B]$ of gray levels and reduces all others to a constant level.
(b) This transformation highlights range $[A, B]$ but preserves all other levels.
(c) An image.
(d) Result of using the transformation in (a).

位图切割

- ✓ 不提高灰度范围的亮度，而是强调特定位平面，来调整图像显示
- ✓ 假设一幅256级别灰度的图像，每个像素由8个位（一个字节）构成

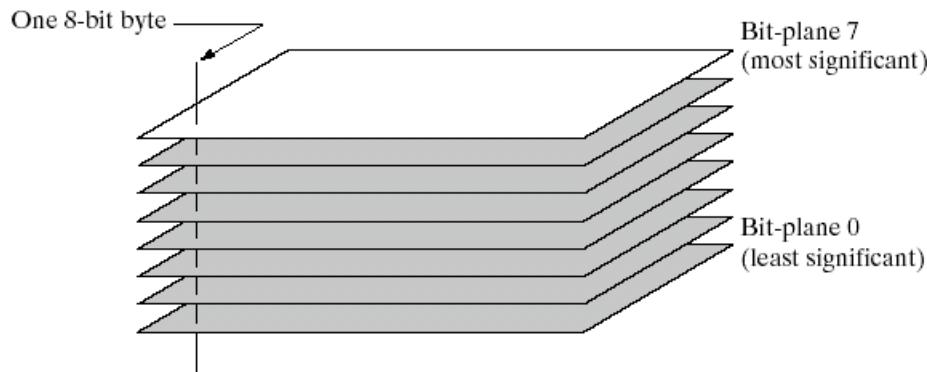


FIGURE 3.12
Bit-plane
representation of
an 8-bit image.

位图切割

- ✓ 在这些位平面中，较高位的数据包含了在视觉上很重要的大多数数据，而较低位平面给出了图像的细微细节
- ✓ 应用举例，灰度阈值变换：
 - ✓ 1) 0 - 127 灰度级的像素统一映射为灰度级 0
 - ✓ 2) 128 - 255 灰度级的像素统一映射为灰度级 255
 - ✓ 直接使用位平面 7 的二值图像即可

位图切割

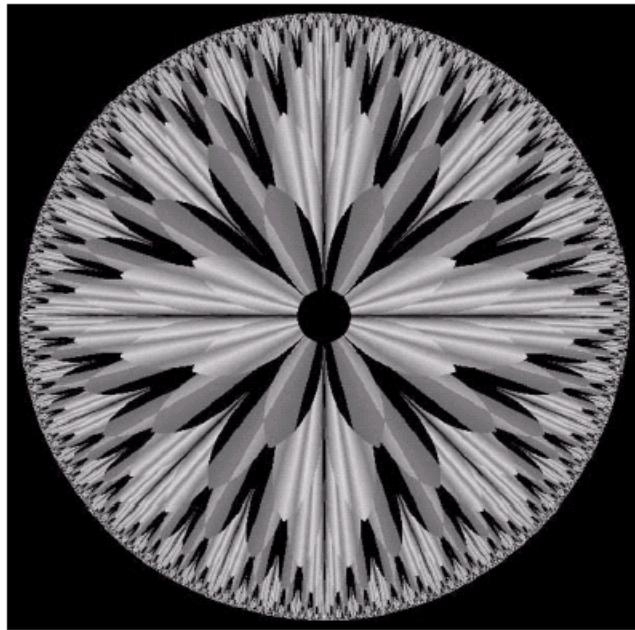


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Sv

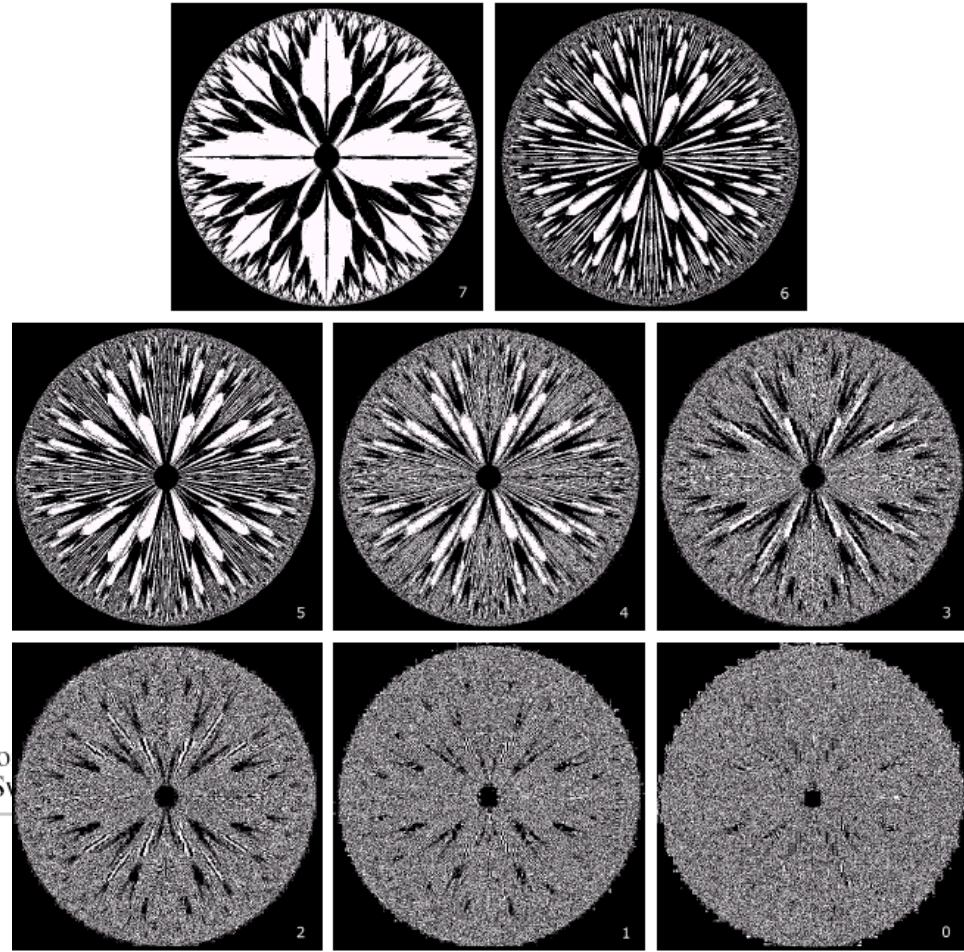
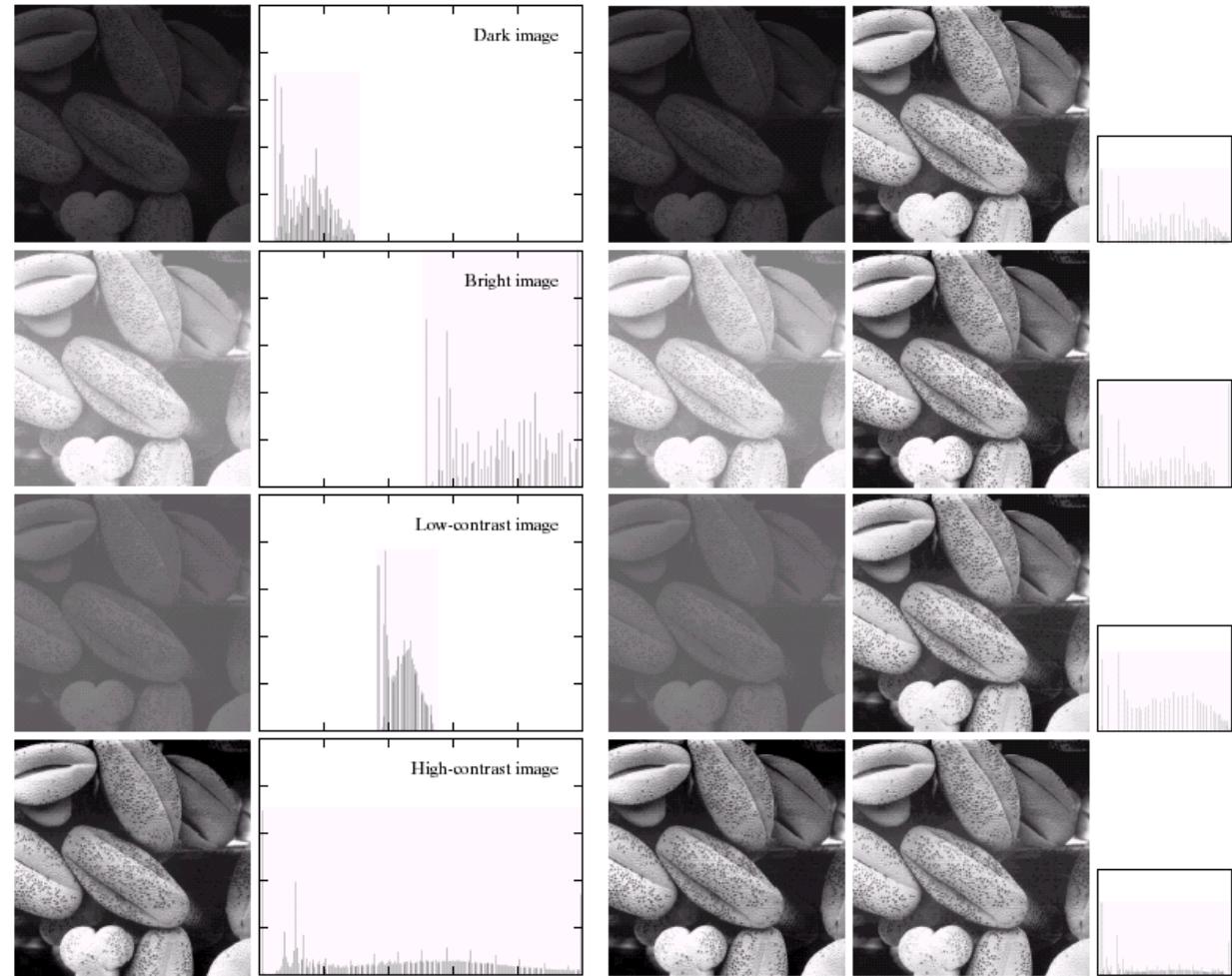
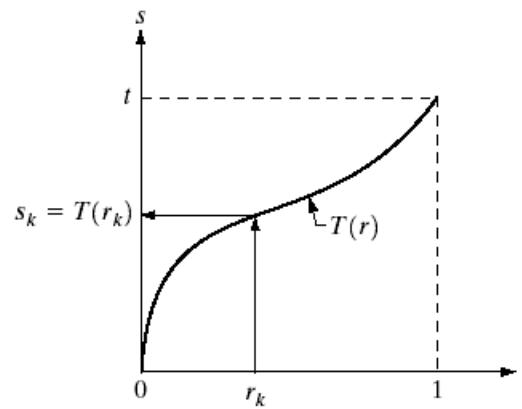


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom right of each image identifies the bit plane.

直方图处理

直方图均衡化



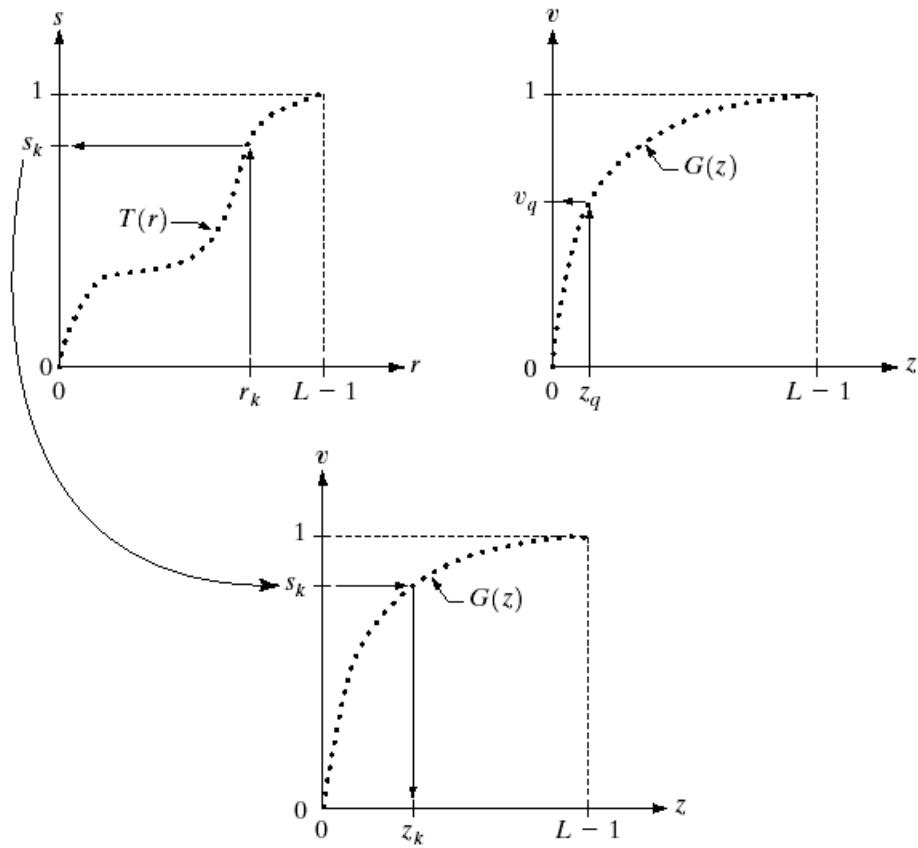
直方图处理

直方图规范化

a
b
c

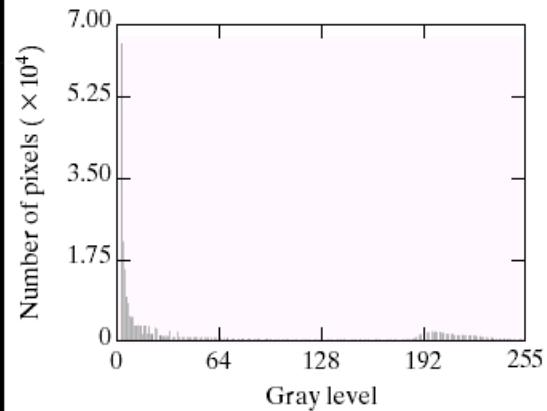
FIGURE 3.19

- (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 .



$$z = G^{-1}[T(r_k)]$$

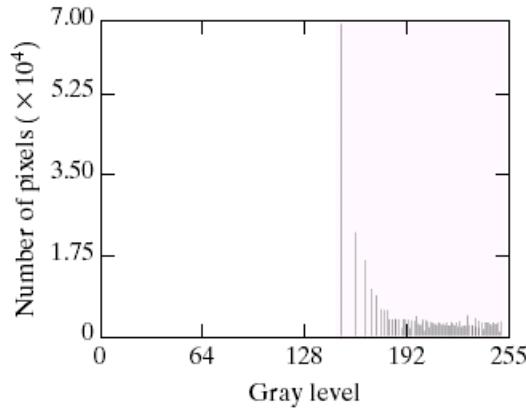
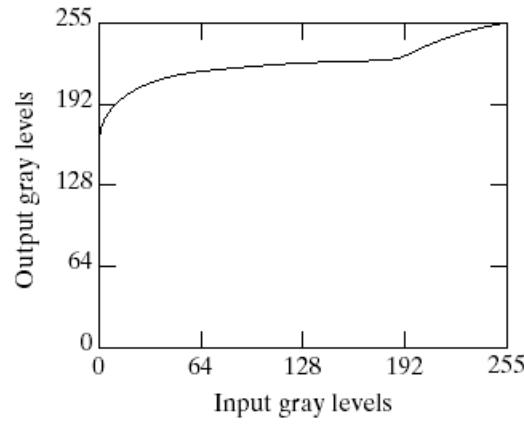
直方图处理



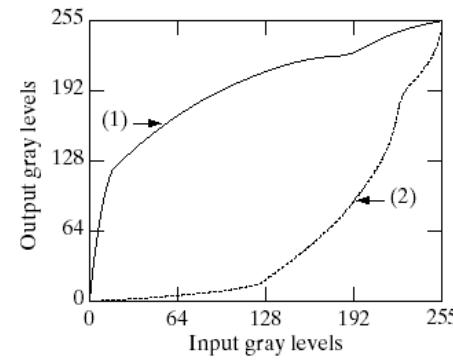
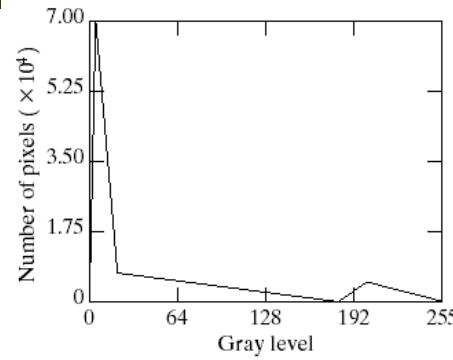
a b

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

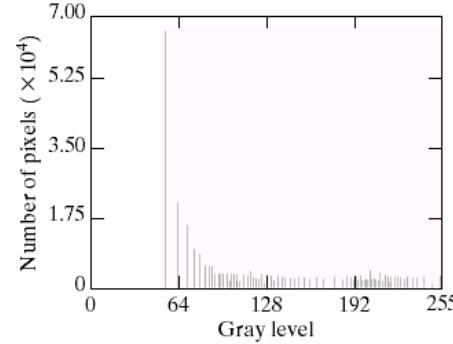
直方图处理



均衡化



规范化



直方图处理

- ✓ 局部增强技术：前面讨论的直方图均衡化和规范化是全局性的方法
- ✓ 有时，图像只是在某些局部区域的细节需要增强处理，此时可以将直方图的计算区从整幅图像调整到像素的方形邻域

直方图处理

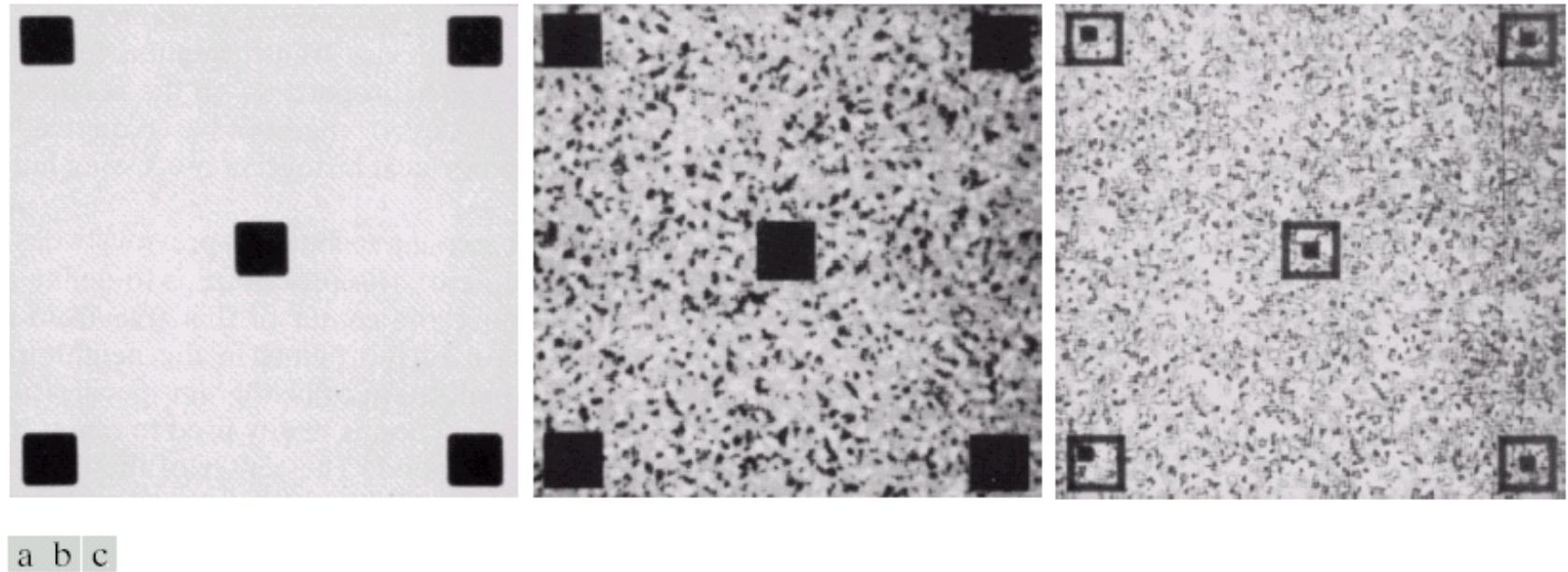


FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

直方图处理

- ✓ 使用统计法增强图像：从直方图获得本幅图像或者本区域像素值的统计参数
- ✓ 令 r 是像素灰度值， $p(r)$ 是归一化直方图分量，可以把 $p(r)$ 看成是灰度级 r 出现的频率，并近似看作概率估计值

直方图处理

✓ 均值： $m = \sum_{i=0}^{L-1} r_i p(r_i)$

✓ n阶矩： $\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i)$

✓ 二阶矩（方差）： $\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^2 p(r_i)$

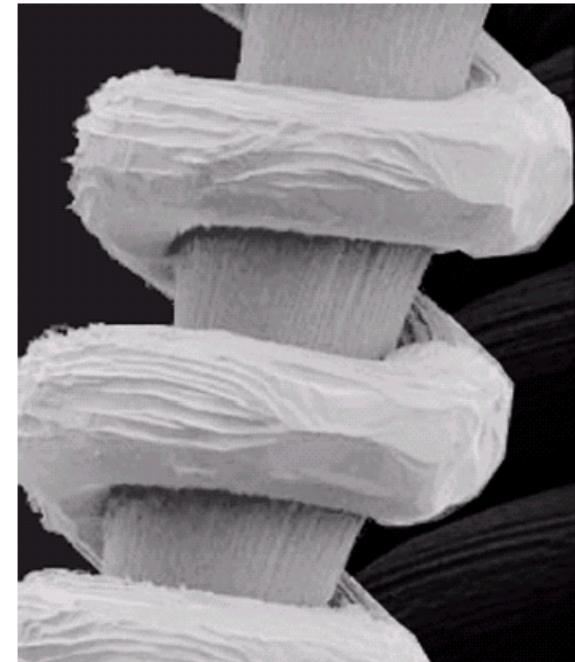
✓ 邻域计算：

$$m_{S_{xy}} = \sum_{(s,t) \in S_{xy}} r_{s,t} p(r_{s,t}) \quad \sigma_{S_{xy}}^2 = \sum_{(s,t) \in S_{xy}} [r_{s,t} - m_{S_{xy}}]^2 p(r_{s,t})$$

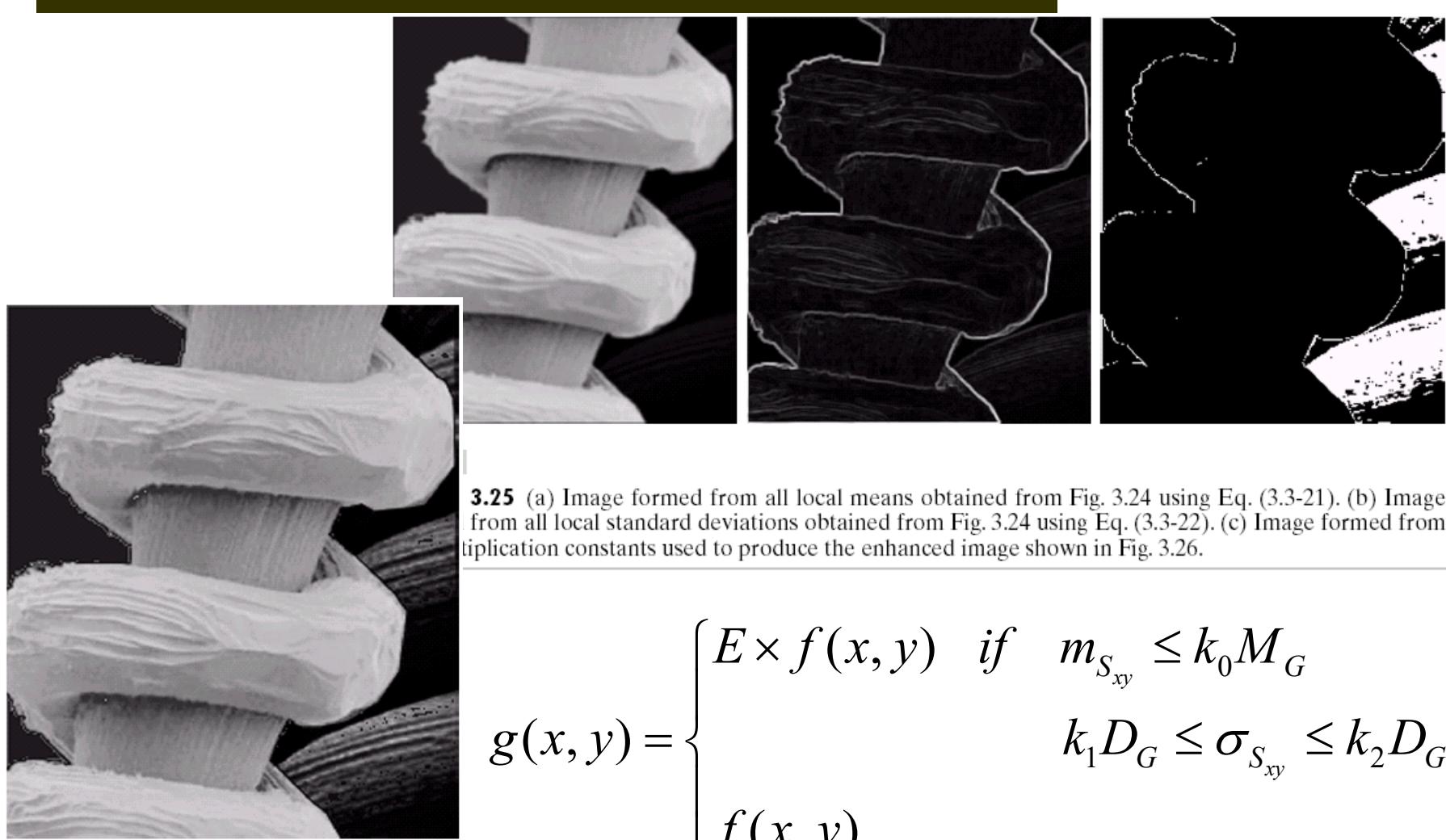
直方图处理

- ✓ 基于局部统计量的增强
- ✓ 像素均值：给出了局部的像素亮度
- ✓ 像素方差：给出了局部的亮度对比

方案：增强较暗的区域，
具有适当对比度的像素，
同时保持亮的区域



直方图处理

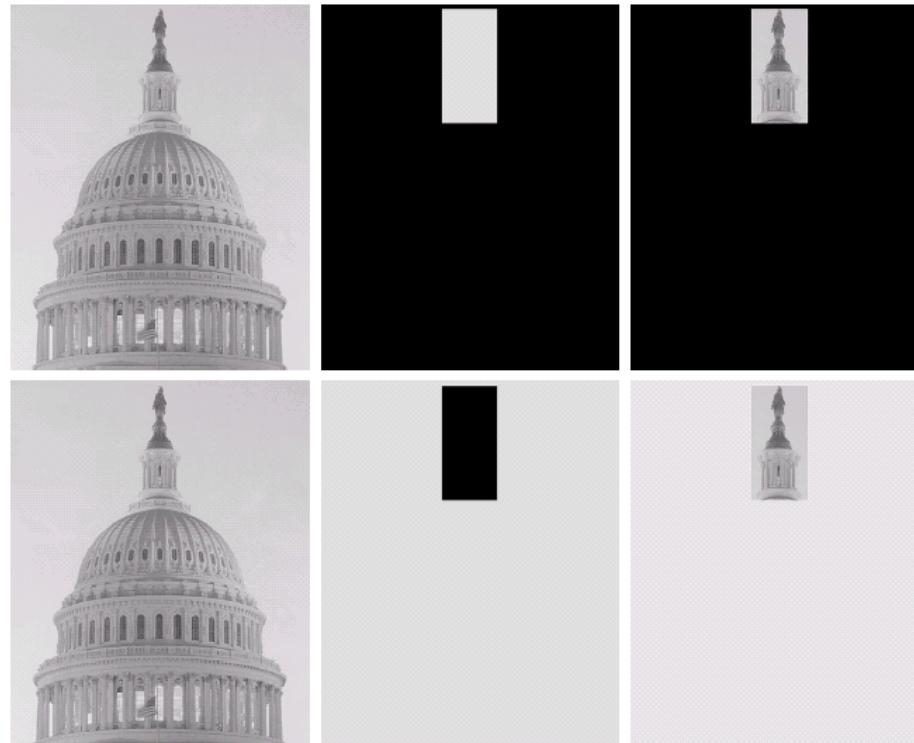


算术逻辑操作增强图像

- ✓ 在图像处理中，算术运算和逻辑操作主要是以像素对像素的方式在两幅或者多幅图像之间进行（逻辑非除外）
- ✓ 将像素值视作二进制串，使用逻辑运算来处理：
 - ✓ 逻辑非可以实现图像反转功能
 - ✓ 与、或操作通常作为模板，从一幅图像中提取子图像

算术逻辑操作增强图像

- 下图中的模板，亮的为1，黑的为0，从左图中提取感兴趣的区域子图像



a b c
d e f

FIGURE 3.27
(a) Original image. (b) AND image mask.
(c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask.
(f) Result of operation OR on images (d) and (e).

图像减法

- 设有两幅图像 $f(x,y)$ 和 $h(x,y)$, 两者的差图像定义为:

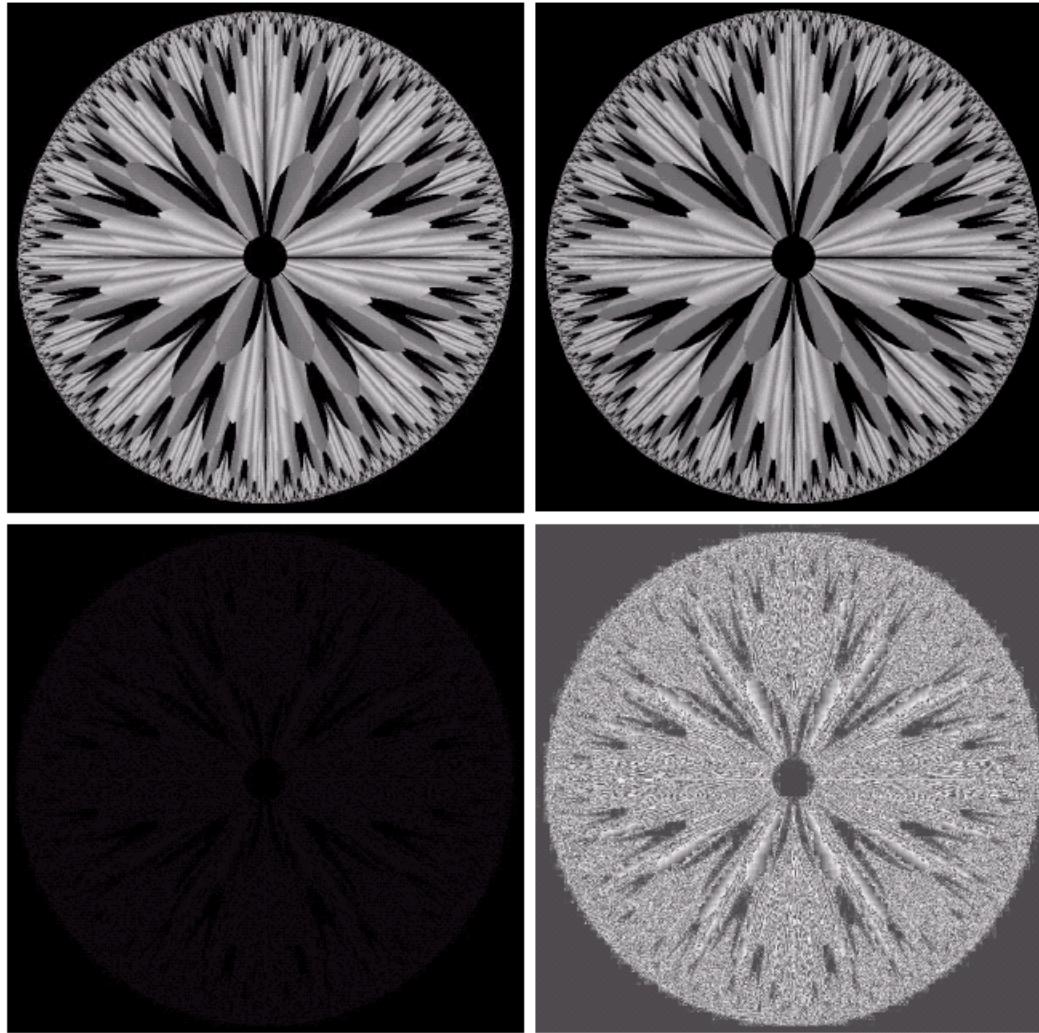
$$g(x,y) = f(x,y) - h(x,y)$$

- 当用一幅包含目标的图像减去背景图像, 则差图像就是只包含目标的图像了, 这有助于我们去除不太重要的背景信息, 而提取出比较感兴趣的目标物

图像减法

a
b
c
d

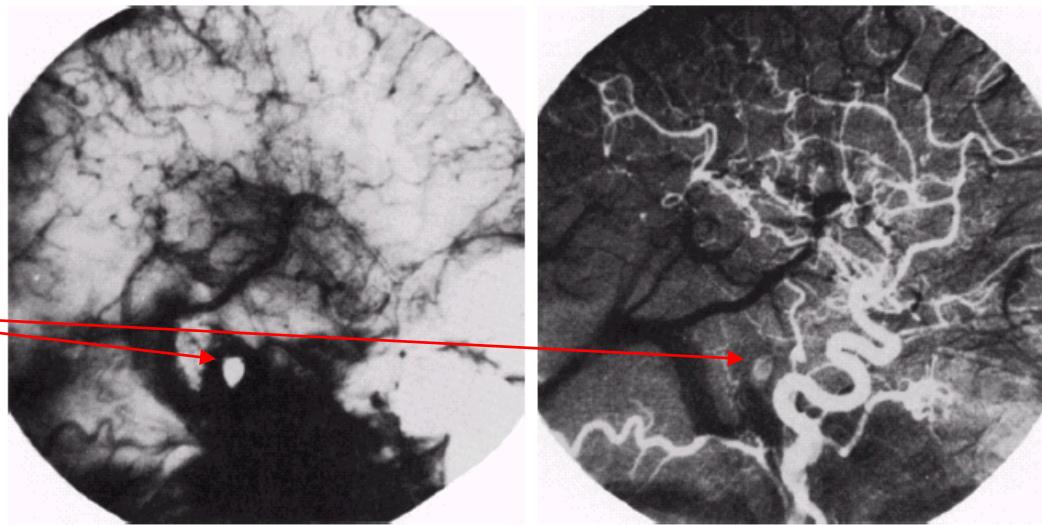
FIGURE 3.28
(a) Original fractal image.
(b) Result of setting the four lower-order bit planes to zero.
(c) Difference between (a) and (b).
(d) Histogram-equalized difference image.
(Original image courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA).



图像减法

- 掩膜式X光成像法：在注入碘元素之前拍摄的背景图像，注入碘元素后视频拍摄的图像减去掩膜图像

脊柱



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.

图像减法

- 减法处理中的问题：出现负数
- 处理办法，使用对比拉伸

$$g(x, y) = f(x, y) - h(x, y)$$

$$g(x, y) \in [-255, 255]$$

$$g(x, y) = (g(x, y) + 255) / 2$$

图像减法

- ✓ 减法处理中的问题：出现负数
- ✓ 有效利用灰度级范围

$$g(x, y) = f(x, y) - h(x, y)$$

$$g(x, y) \in [min, max]$$

$$g(x, y) = (g(x, y) - min) \times 255 / max$$

图像加法

- ❑ 图像加法可以在两幅或者多幅图像中进行，不过在执行加法后需要取平均值

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

- ❑ 注意上式中左右图像均为 g ，只是下标不一样，如此安排是因为：图像加法更多是在相同场景的不同时刻的图像之间进行

图像加法

- ✓ 设对一场景进行拍摄，但结果图像可能会被噪声污染，这可以通过图像加法解决
- ✓ 设图像被噪声污染，该噪声均值为0，方差为 $\sigma_{\eta(x,y)}$ ，显然方差越大，噪声越强

$$g(x, y) = f(x, y) + \eta(x, y)$$

图像加法

✓ 根据式 $\bar{g}(x, y) = \frac{1}{K} \sum_{i=1}^k g_i(x, y)$

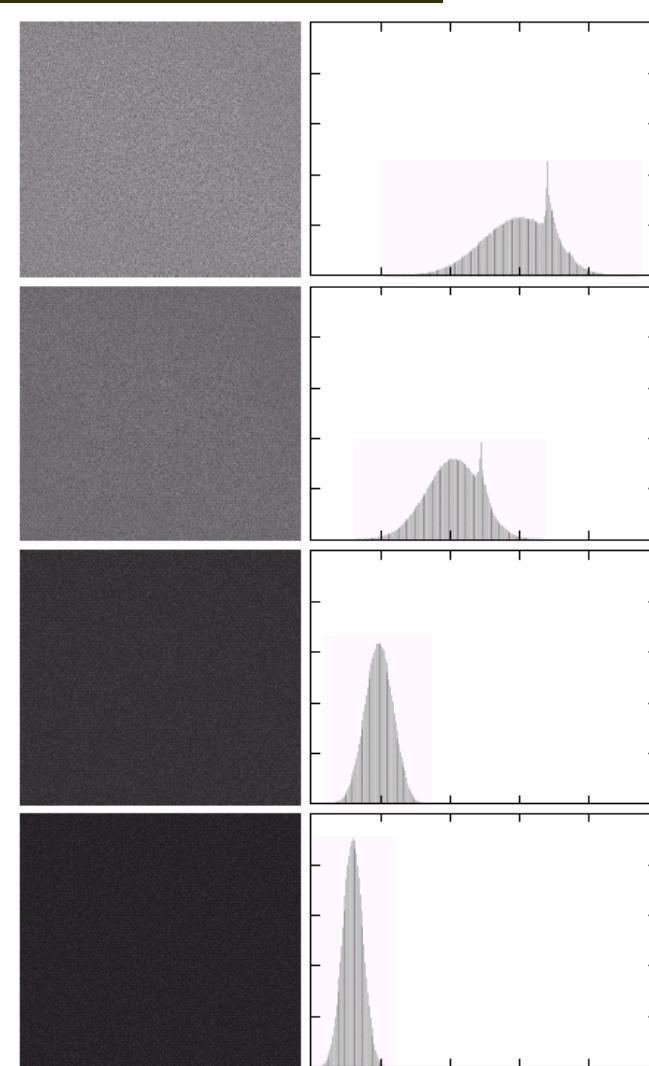
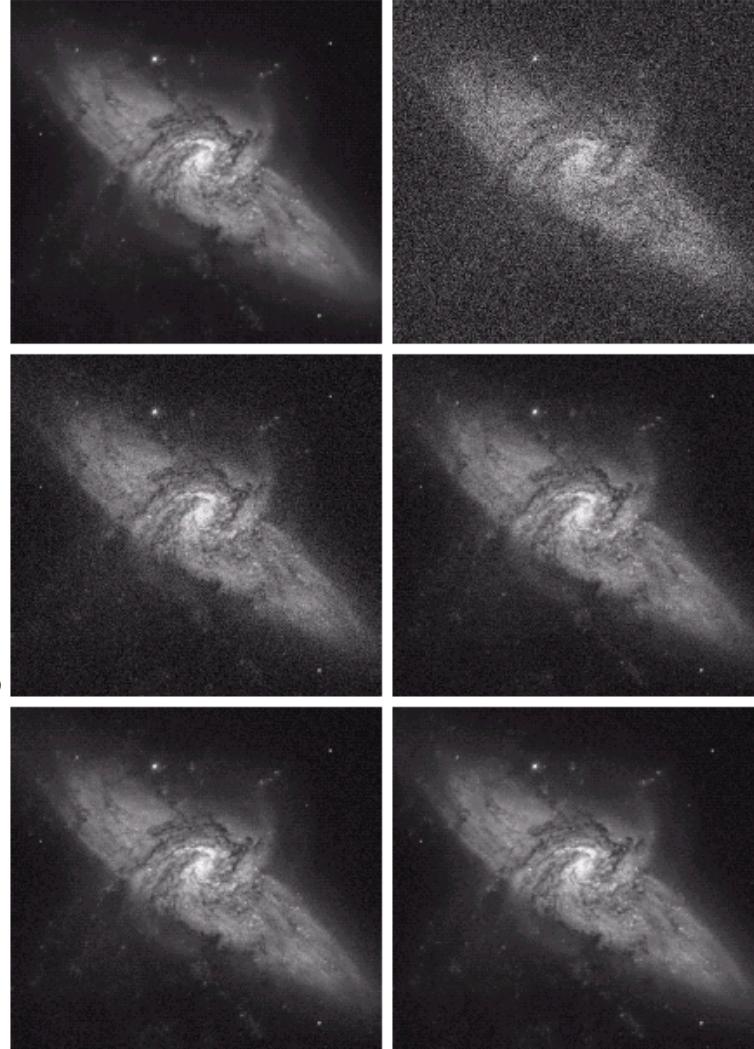
$$E\{\bar{g}(x, y)\} = f(x, y)$$

$$\sigma_{\bar{g}(x, y)}^2 = \frac{1}{K} \sigma_{\eta(x, y)}^2$$

当K增加时，在结果图像中各像素点的
噪声影响会越来越小，因此会越来越趋
近于原始图像 $f(x, y)$

图像加法

原图像和噪声图像
8
16
64
128
幅图像取平均



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.

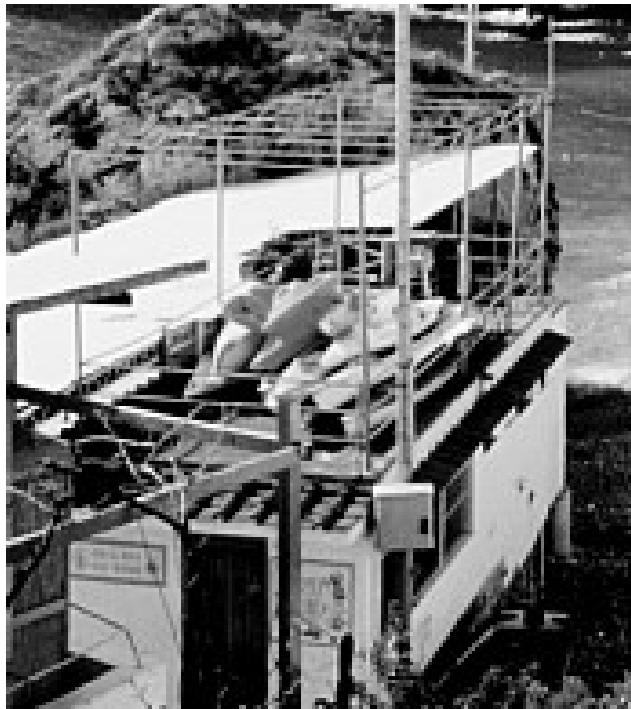
平均后图像与原图像的差值及差图像直方图

空间域平滑

- 任何一幅图像，在获取和传输的过程中，会受到各种噪声的干扰，使图像质量下降，图像特征被淹没，对图像分析造成影响，为了抑制噪声，改善图像质量所进行的处理称为图像噪声去除，简称去噪
- 因为去噪的过程往往使图像中的尖锐信号被修整或抹平，因此大部分去噪算法又称为图像平滑算法

空间域平滑

- 为了抑制噪声改善图像质量所进行的处理称图像平滑或去噪。它可以在空间域和频率域中进行



模 板

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



空间域平滑

- ✓ 本节介绍空间域的几种平滑法
 - ✓ 局部平滑法
 - ✓ 超限像素平滑法
 - ✓ 灰度最相近的K个邻点平均法
 - ✓ 梯度倒数加权平滑法
 - ✓ 最大均匀性平滑
 - ✓ 有选择保持边缘平滑法
 - ✓ 空间低通滤波法
 - ✓ 多幅图像平均法
 - ✓ 中值滤波

局部平滑法

- ✓ 局部平滑法是一种直接在空间域上进行平滑处理的技术。假设图像是由许多灰度恒定的小块组成，相邻像素间存在很高的空间相关性，而噪声则是统计独立的。因此，可用邻域内各像素的灰度平均值代替该像素原来的灰度值，实现图像的平滑

局部平滑法

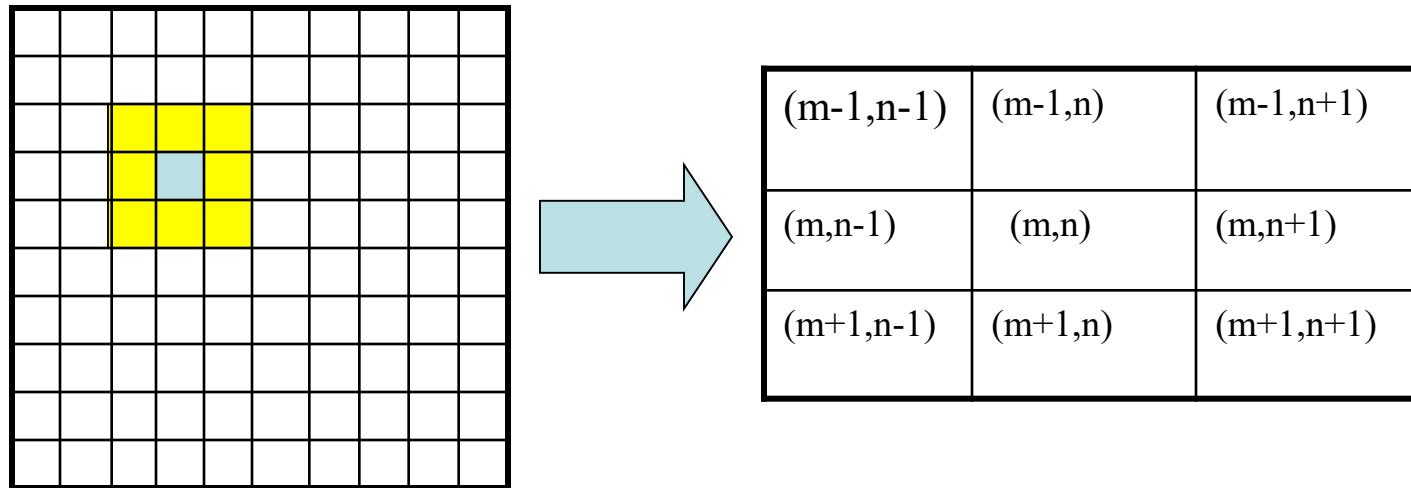
- 设有一幅 $N \times N$ 的图像 $f(x,y)$, 若平滑图像为 $g(x,y)$, 则有

$$g(x, y) = \frac{1}{M} \sum_{i, j \in S} f(i, j)$$

- 式中 $x, y = 0, 1, \dots, N-1$;
- S 为 (x, y) 邻域内像素坐标的集合;
- M 表示集合 S 内像素的总数。
- 可见邻域平均法就是将当前像素邻域内各像素的灰度平均值作为其输出值的去噪方法

局部平滑法

- 例如，对图像采用 3×3 的邻域平均法，对于像素 (m,n) ，其邻域像素如下：



则有：

$$g(m,n) = \frac{1}{9} \sum_{i \in Z} \sum_{j \in Z} f(m+i, n+j)$$

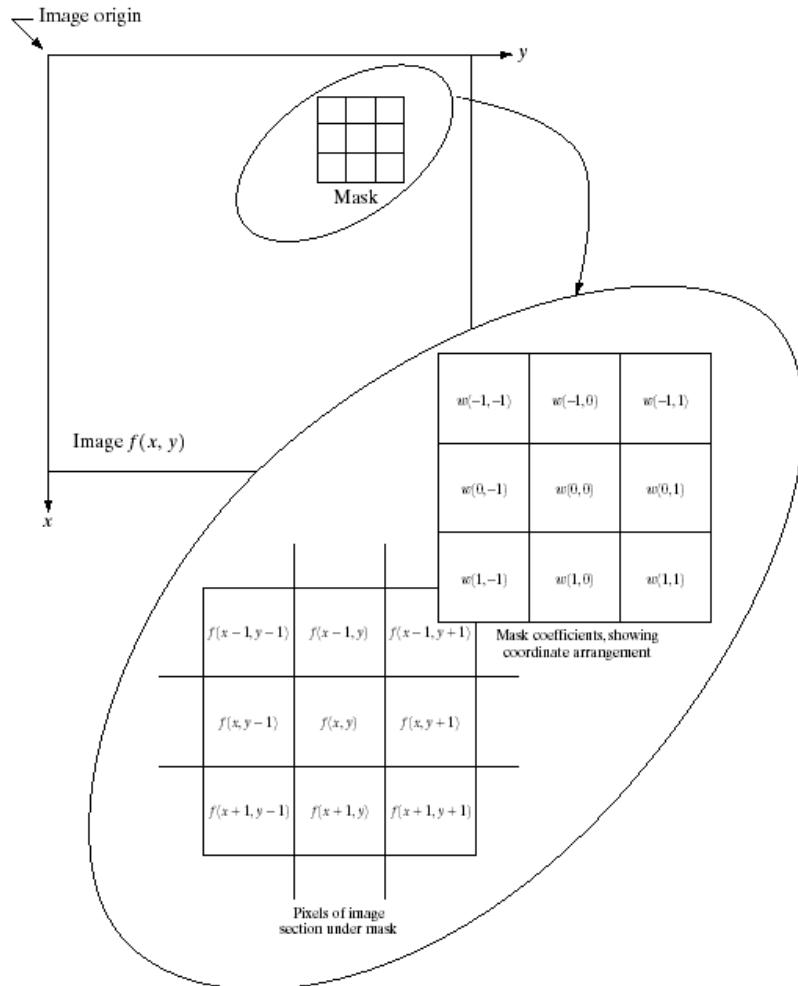
局部平滑法

- 其作用相当于用这样的模板同图像卷积。

$$H = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- 设图像中的噪声是随机不相关的加性噪声，窗口内各点噪声是独立同分布的，经过上述平滑后，信号与噪声的方差比可望提高M倍

空间域图像处理模板

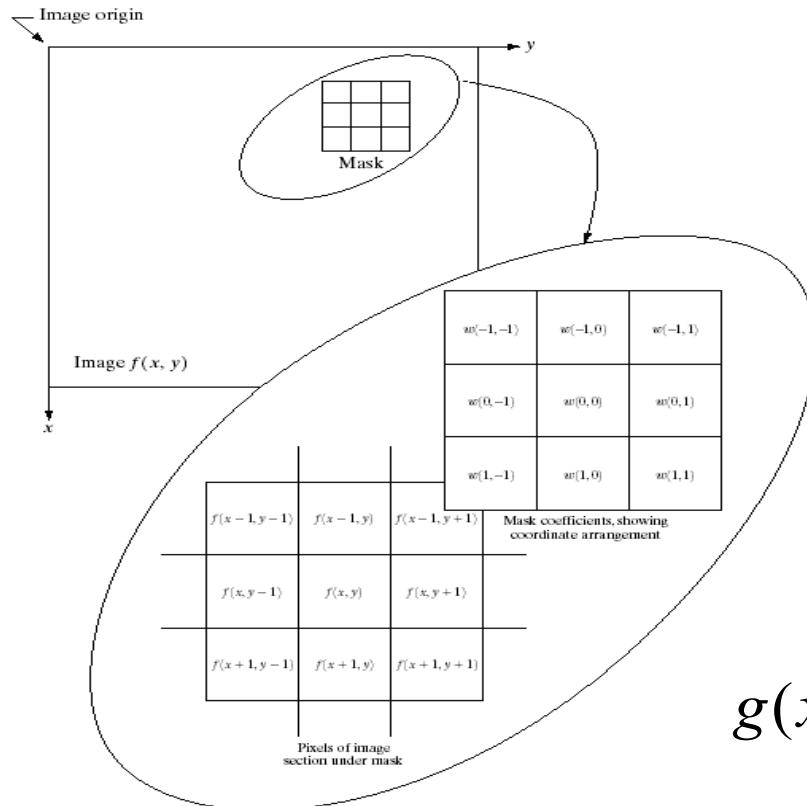


在待处理图像的每个像素上进行处理，处理的依据是其周边像素值的线性组合

周边像素集合作为邻域，取出各像素的值 $f(i, j)$ ，乘以相应的加权系数 $w(i, j)$ ，最后求和得到处理后的像素值

空间域图像处理模板

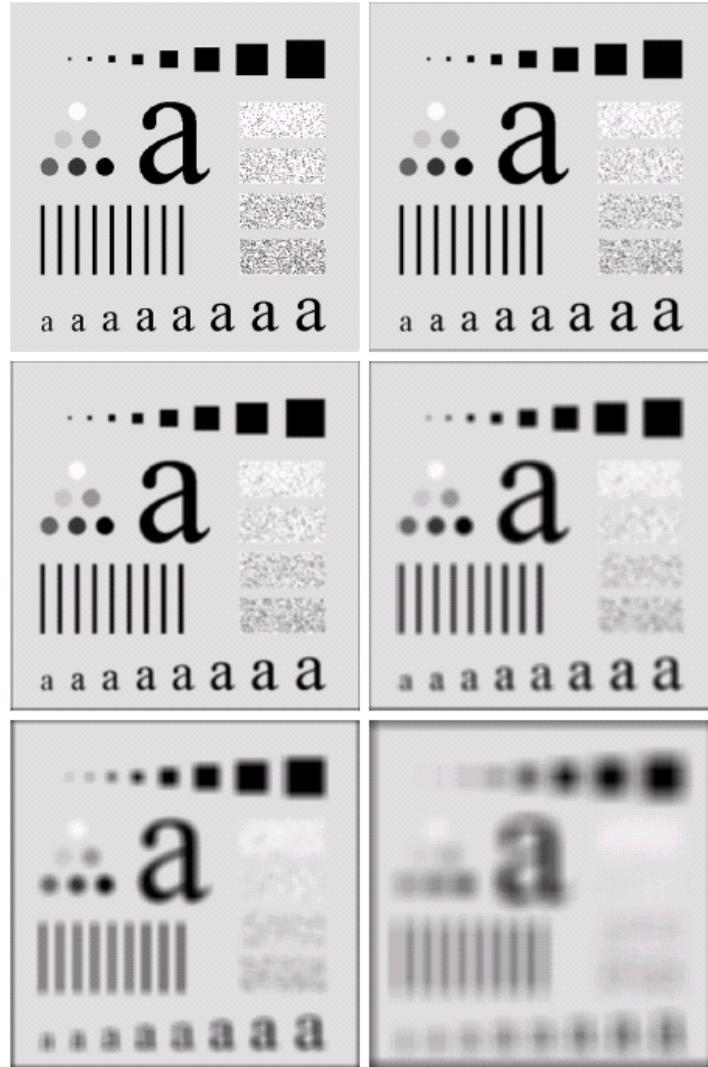
加权系数值构成的权值矩阵常被称为：



{ 模板： template
掩膜： mask
卷积核： kernel
滤波器： filter

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

局部平滑法



$n=3$, 图像中有轻微模糊, 尺寸比滤波器小或者接近的细节受影响较大, 如较小的字母a和细的颗粒比其他部分模糊
(噪声明显减少字母锯齿状边界与圆圈都平滑了)

$n=9$, 图像更模糊, 某些圆圈与背景融合较多, 不易与相似灰度的背景区分开, $n=15$ 和 35 时, 小尺寸物体就被抹去了, 由于边界补0, 当滤波器尺寸变大而引起了黑色边界问题

局部平滑法

- ✓ 这种算法简单，但它的主要缺点是在降低噪声的同时使图像产生模糊，特别是在边缘和细节处。而且邻域越大，在去噪能力增强的同时模糊程度越严重
- ✓ 平滑算法对白噪声（高斯噪声）有效，对椒盐噪声效果一般甚至较差，对这类噪声，非线性的处理技术更有效

局部平滑法



(a)



(b)



(c)



(d)

(a) 原图像

(b) 加椒盐噪声的图像

(c) 3×3 邻域平滑

(d) 5×5 邻域平滑

改进方法

- 为克服简单局部平均法的弊病，目前已提出许多保边缘、细节的局部平滑算法。它们的出发点都集中在如何选择邻域的大小、形状和方向、参加平均的点数以及邻域各点的权重系数等

超限像素平滑法

- 对邻域平均法稍加改进，可导出超限像素平滑法。它是将 $f(x,y)$ 和邻域平均 $g(x,y)$ 差的绝对值与选定的阈值进行比较，根据比较结果决定点 (x,y) 的最后灰度 $g'(x,y)$ 。其表达式为

$$g'(x,y) = \begin{cases} g(x,y), & \text{当 } |f(x,y) - g(x,y)| > T \\ f(x,y), & \text{否则} \end{cases}$$

超限像素平滑法

- ✓ 这算法对抑制椒盐噪声比较有效，对保护仅有微小灰度差的细节及纹理也有效。可见随着邻域增大，去噪能力增强，但模糊程度也大。同局部平滑法相比，超限像元平滑法去椒盐噪声效果更好



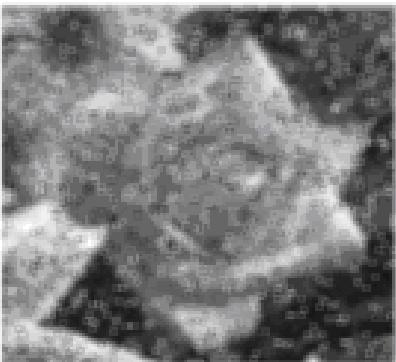
(a)



(b)

(a) 原图像

(b) 对 (a) 加椒盐噪声的图像



(c)



(d)

(c) 3×3 邻域平滑

(d) 5×5 邻域平滑



(e)



(f)

(e) 3×3 超限像素平滑 ($T=64$)

(f) 5×5 超限像素平滑 ($T=48$)

灰度最相近的K个邻点平均法

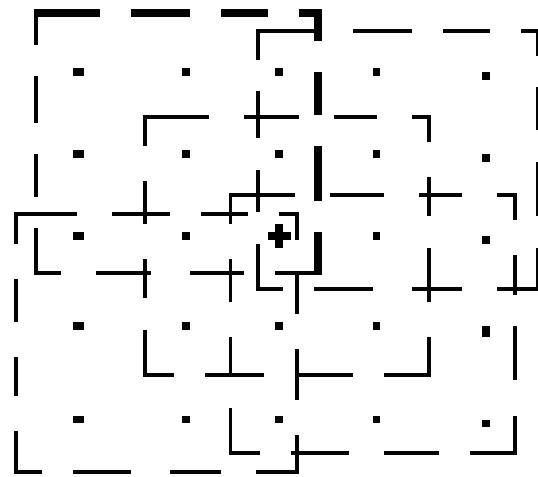
- 算法的出发点是：在 $n \times n$ 的窗口内，属于同一集合体的像素，它们的灰度值将高度相关。因此，可用窗口内与中心像素的灰度最接近的K个邻像素的平均灰度来代替窗口中心像素的灰度值。此即灰度最相近K邻点平均法

灰度最相近的K个邻点平均法

- 较小的K值使噪声方差下降较小，但保持细节效果较好；而较大的K值平滑噪声较好，但会使图像边缘模糊。
- 实验证明，对于 3×3 的窗口，取K=6为宜

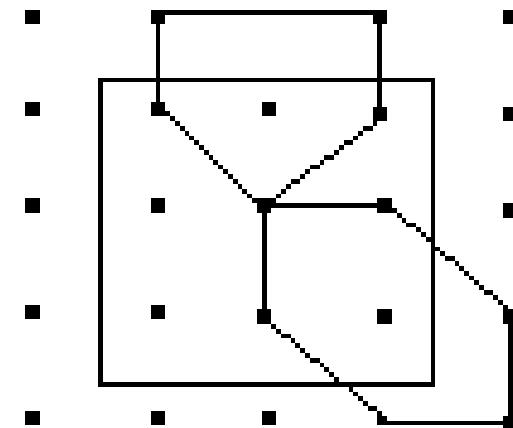
最大均匀性平滑

- 为避免消除噪声引起边缘模糊，该算法先找出环绕图像中每像素的最均匀区域，然后用这区域的灰度均值代替该像素原来的灰度值



有选择保边缘平滑法

- 该方法对图像上任一像素 (x,y) 的 5×5 邻域，采用9个掩模，其中包括一个 3×3 正方形、4个五边形和4个六边形。计算各个掩模的均值和方差，对方差进行排序，最小方差所对应的掩模区的灰度均值就是像素 (x,y) 的输出值



有选择保边缘平滑法

- 该方法以方差作为各个区域灰度均匀性的测度。若区域含有尖锐的边缘，它的灰度方差必定很大，而不含边缘或灰度均匀的区域，它的方差就小，那么最小方差所对应的区域就是灰度最均匀区域。因此有选择保边缘平滑法既能够消除噪声，又不破坏区域边界的细节。另外，五边形和六边形在 (x,y) 处都有锐角，这样，即使像素 (x,y) 位于一个复杂形状区域的锐角处，也能找到均匀的区域。从而在平滑时既不会使尖锐边缘模糊，也不会破坏边缘形状

有选择保边缘平滑法

- 例如，某像素 5×5 邻域的灰度分布如图 4.2.4，经计算9个掩模区的均值和方差为

均值	3	4	4	2	2	4	3	3	3	4	7	3	2	4
对应的方差	11	30	37	13	20	26	12	26	32	8	4	1	4	3

最小方差为11，对应的灰度均值3，

采用有选择保边缘平滑，该像素的输出值为3。

空间低通滤波法

- 邻域平均法可看作一个掩模作用于图像 $f(x,y)$ 的低通空间滤波，掩模就是一个滤波器，它的响应为 $H(r,s)$ ，于是滤波输出的数字图像 $g(x,y)$ 用离散卷积表示为

$$g(x,y) = \sum_{r=-k}^k \sum_{s=-l}^l f(x-r, y-s) H(r,s)$$

空间低通滤波法

常用的掩模有

$$H_1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad H_2 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad H_3 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$H_4 = \frac{1}{8} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad H_5 = \frac{1}{2} \begin{bmatrix} 0 & \frac{1}{4} & 0 \\ \frac{1}{4} & 1 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 \end{bmatrix}$$

空间低通滤波法

- 掩模不同，中心点或邻域的重要程度也不相同，因此，应根据问题的需要选取合适的掩模。但不管什么样的掩模，必须保证全部权系数之和为单位值，这样可保证输出图像灰度值在许可范围内，不会产生“溢出”现象

统计排序滤波器

- ✓ 统计滤波器是一种非线性的空间滤波器，它的响应基于滤波器包围的图像区域中像素的排序，然后用统计排序结果决定的值代替中心像素的值

中值滤波

- ✓ 中值滤波是对一个滑动窗口内的诸像素灰度值排序，用中值代替窗口中心像素的原来灰度值，因此它是一种非线性的图像平滑法
- ✓ 这里中值指的是一个数值集合中的某个值，在集合中有一半值小于或等于这个值，也有一半值大于或等于这个值

中值滤波

例：采用 1×3 窗口进行中值滤波

原图像为：

2 2 6 2 1 2 4 4 4 2 4

处理后为：

2 2 2 2 2 2 4 4 4 4 4

中值滤波

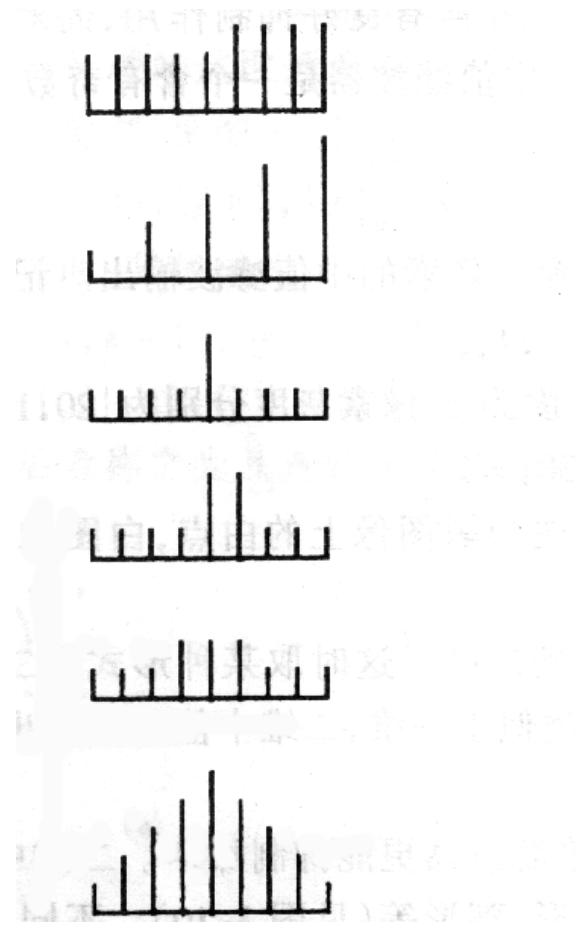
- ✓ 它对脉冲干扰及椒盐噪声的抑制效果好，在抑制随机噪声的同时能有效保护边缘少受模糊。但它对点、线等细节较多的图像却不太合适
- ✓ 对中值滤波法来说，正确选择窗口尺寸的大小是很重要的环节。一般很难事先确定最佳的窗口尺寸，需通过从小窗口到大窗口的中值滤波试验，再从中选取最佳的

一维中值滤波的几个例子 (N=5)

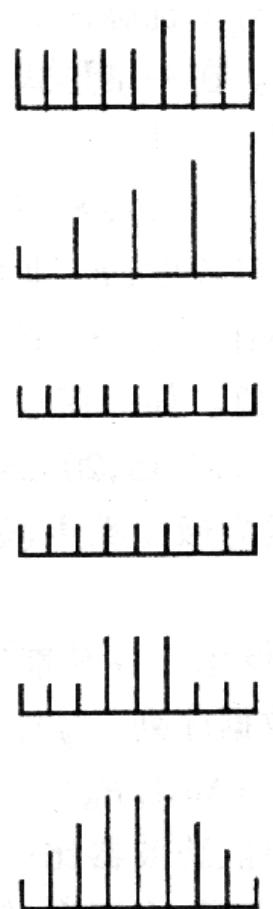
- ✓ 离散阶跃信号、斜升信号没有受到影响。
离散三角信号的顶部则变平了。对于离散的脉冲信号，当其连续出现的次数小于窗口尺寸的一半时，将被抑制掉，否则将不受影响

一维中值滤波的几个例子 (N=5)

原图像

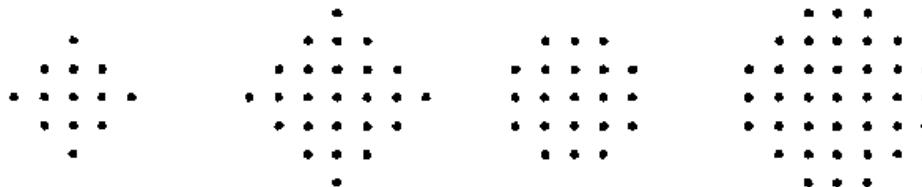


中值滤波



二维中值滤波

- ✓ 一维中值滤波的概念很容易推广到二维。
一般来说，二维中值滤波器比一维滤波器更能抑制噪声
- ✓ 二维中值滤波器的窗口形状可以有多种，
如线状、方形、十字形、圆形、菱形等



二维中值滤波

- ✓ 不同形状的窗口产生不同的滤波效果，使用中必须根据图像的内容和不同的要求加以选择。从以往的经验看，方形或圆形窗口适宜于外轮廓线较长的物体图像，而十字形窗口对有尖顶角状的图像效果好

二维中值滤波



(a)



(b)

图(a)为原图像；图(b)为加椒盐噪声的图像；图(c)和图(d)分别为 3×3 、 5×5 模板进行中值滤波的结果



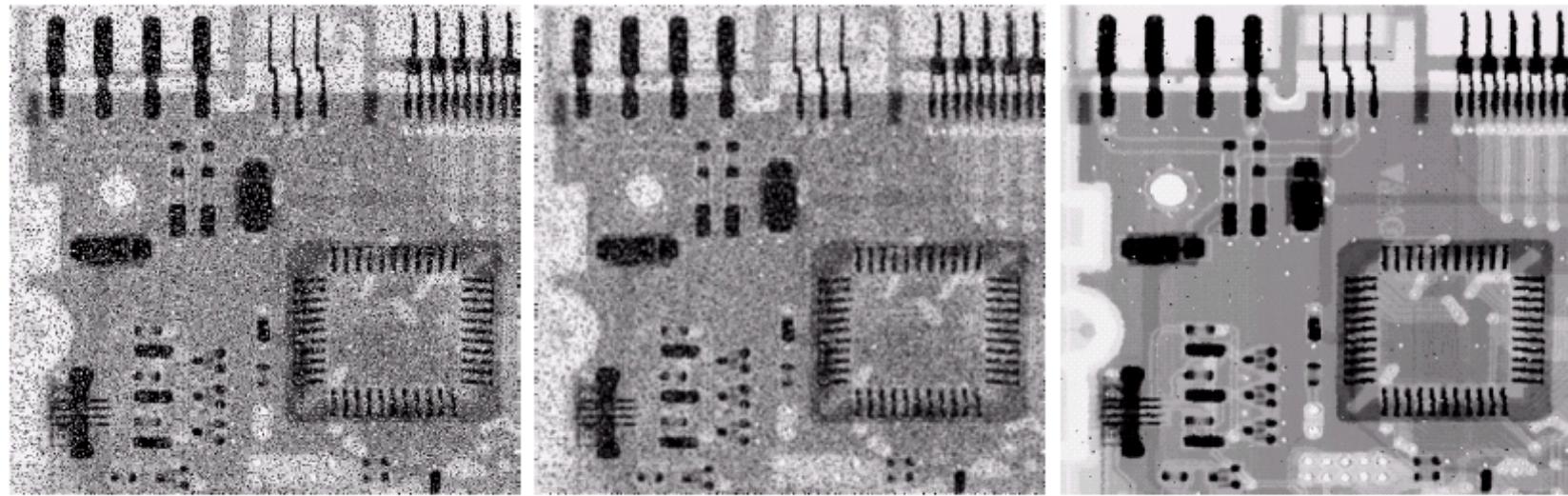
(c)



(d)

可见中值滤波法能有效削弱椒盐噪声，且比邻域、超限像素平均法更有效

二维中值滤波



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

图像空间域锐化

- ✓ 在图像的识别中常需要突出边缘和轮廓信息。图像锐化就是增强图像的边缘或轮廓。
- ✓ 图像平滑通过积分过程使得图像边缘模糊，图像锐化则通过微分而使图像边缘突出、清晰。

图像中的微分

- 对于一元函数 $f(x)$, 一阶微分是其导数

$$f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

- 二阶微分是导数的导数

$$f''(x) = \lim_{\Delta x \rightarrow 0} \frac{f'(x + \Delta x) - f'(x)}{\Delta x}$$

图像中的微分

- 对于一元函数 $f(x)$, 一阶微分是其导数

$$f'(x) \approx \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

- 二阶微分是导数的导数

$$\begin{aligned} f''(x) &\approx \frac{f'(x + \Delta x) - f'(x)}{\Delta x} \\ &\approx \frac{f(x + 2\Delta x) - 2f(x + \Delta x) + f(x)}{\Delta x^2} \end{aligned}$$

图像中的微分

- 图像中的像素间隔为1，因此取 $\Delta x = 1$

$$f'(x) \approx \frac{f(x + \Delta x) - f(x)}{\Delta x} = f(x + 1) - f(x)$$

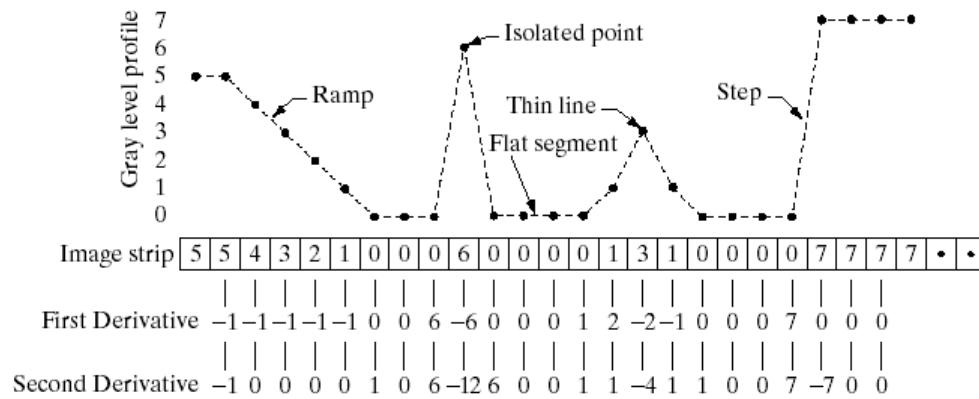
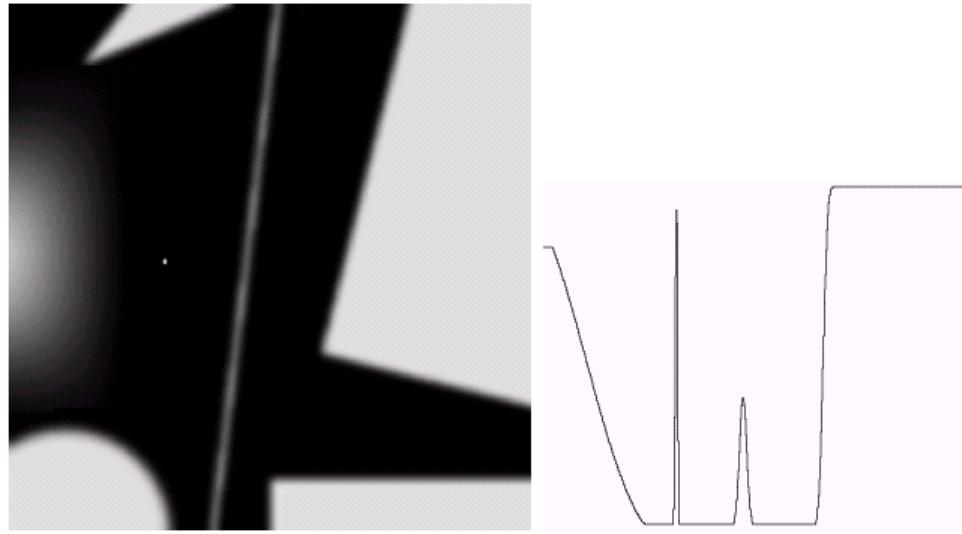
- 二阶微分是导数的导数

$$f''(x) \approx \frac{f'(x + \Delta x) - f'(x)}{\Delta x} = f(x + 1) + f(x - 1) - 2f(x)$$

图像中的微分

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).



图像中的微分

- ✓ 根据上图计算知：
 - ✓ 一阶微分只有在平坦的区域为0或接近0
 - ✓ 二阶微分在平坦处以及斜坡处为0，只在斜坡的起始和终点处才非零
 - ✓ 二阶微分对孤立噪声点及其周边点的响应比一阶微分强的多
- ✓ 结论：
 - ✓ 一阶微分处理通常产生较宽边缘
 - ✓ 二阶微分处理对细节有较强的响应如细线条和孤立点
 - ✓ 一阶微分处理一半对灰度阶梯有较强响应
 - ✓ 二阶微分处理对灰度阶梯变化产生双响应

梯度锐化法

- 图像锐化法最常用的是梯度法。对于图像 $f(x, y)$, 在 (x, y) 处的梯度定义为

$$grad(x, y) = \begin{bmatrix} f'_x \\ f'_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix}$$

- 梯度是一个矢量，其大小和方向为

$$|grad(x, y)| = \sqrt{f'_x^2 + f'_y^2} = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2}$$

$$\theta = \operatorname{tg}^{-1}(f'_y / f'_x) = \operatorname{tg}^{-1}\left(\frac{\partial f(x, y)}{\partial y} / \frac{\partial f(x, y)}{\partial x}\right)$$

梯度锐化法

- 对于离散图像处理而言，常用到梯度的大小，因此把梯度的大小习惯称为“梯度”。并且一阶偏导数采用一阶差分近似表示，即

$$f'_x = f(x+1, y) - f(x, y)$$

$$f'_y = f(x, y+1) - f(x, y)$$

为简化梯度的计算，经常使用以下两种近似计算值

$$\nabla f = \max(|f'_x|, |f'_y|)$$

$$\nabla f = |f'_x| + |f'_y|$$

图像空间域锐化

- 除梯度算子以外，还可采用Roberts、Prewitt和Sobel算子计算梯度，来增强边缘

梯度算子模板

$$f_x' = f(x+1, y) - f(x, y) = z_8 - z_5$$

$$f_y' = f(x, y+1) - f(x, y) = z_6 - z_5$$

-1	1
----	---

-1
1

梯度模板

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

-1	0
0	1
1	0

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

图像空间域锐化

- ✓ Roberts[1965]使用交叉差分算法
- ✓ 差分计算式如下

$$f_x' = f(x+1, y+1) - f(x, y) = z_9 - z_5$$

$$f_y' = f(x, y+1) - f(x+1, y) = z_8 - z_6$$

图像空间域锐化

- 为在锐化边缘的同时减少噪声的影响，Prewitt从加大边缘增强算子的模板大小出发，由 2×2 扩大到 3×3 来计算差分，如图(a)所示。

$$\begin{array}{ccc} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{array} \quad \begin{array}{ccc} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{array}$$

(a) Prewitt 算子

$$\begin{array}{ccc} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{array} \quad \begin{array}{ccc} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{array}$$

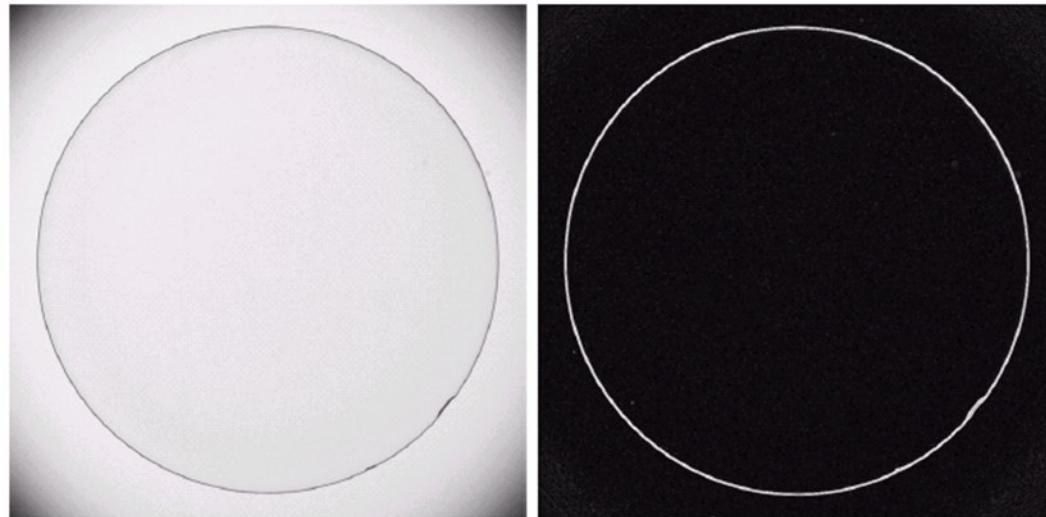
(b) Sobel 算子

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

图像空间域锐化

- ✓ Sobel在Prewitt算子的基础上，对4-邻域采用带权的方法计算差分，对应的模板如图(b)
- ✓ 根据梯度计算式就可以计算Roberts、Prewitt和Sobel梯度。一旦梯度算出后，就可根据不同的需要生成不同的梯度增强图像

例：隐形眼镜检测



a b

FIGURE 3.45
Optical image of
contact lens (note
defects on the
boundary at 4 and
5 o'clock).
(b) Sobel
gradient.
(Original image
courtesy of
Mr. Pete Sites,
Perceptics
Corporation.)

4点、5点钟处两个边缘缺陷，去除了灰度平坦区域，梯度处理突出了小斑点

图像空间域锐化

- ☒ 第一种输出形式
 - ☒ $g(x,y) = \text{grad}(x,y)$
- ☒ 此法的缺点是增强的图像仅显示灰度变化比较陡的边缘轮廓，而灰度变化比较平缓或均匀的区域则呈黑色

图像空间域锐化

☒ 第二种输出形式

$$g(x, y) = \begin{cases} grad(x, y), & grad(x, y) \geq T \\ f(x, y), & \text{其它} \end{cases}$$

图像空间域锐化

- ✓ 第三种输出形式

$$g(x, y) = \begin{cases} L_G & , \quad \text{grad}(x, y) \geq T \\ f(x, y), & \text{其他} \end{cases}$$

- ✓ 它将明显边缘用一固定的灰度级 L_G 来表现

图像空间域锐化

- ✓ 第四种输出形式

$$g(x, y) = \begin{cases} \text{grad}(x, y) & , \text{grad}(x, y) \geq T \\ L_B, & \text{其他} \end{cases}$$

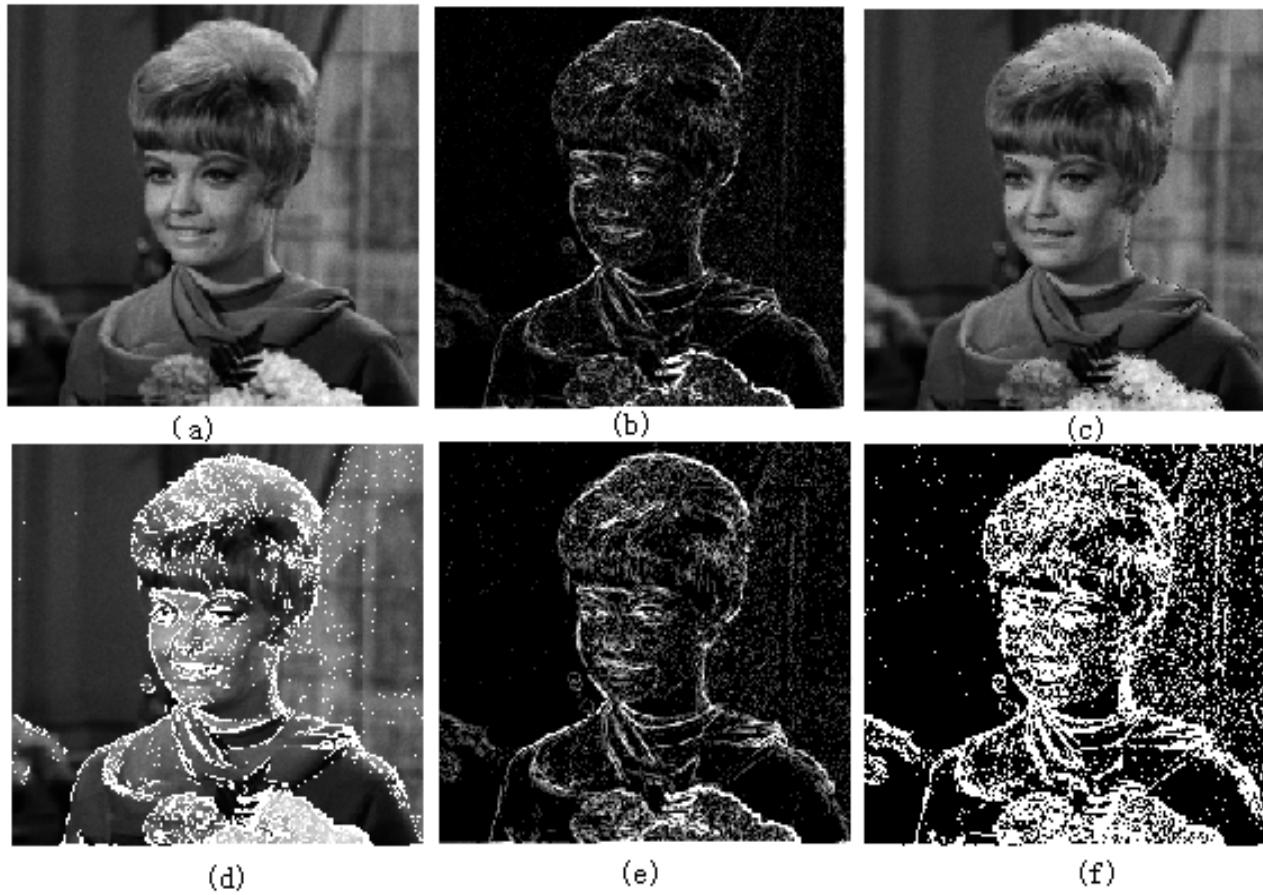
- ✓ 此方法将背景用一个固定的灰度级 L_B 来表现，便于研究边缘灰度的变化

图像空间域锐化

☒ 第五种输出形式

$$g(x, y) = \begin{cases} L_G & , grad(x, y) \geq T \\ L_B & , \text{其他} \end{cases}$$

- ☒ 这种方法将明显边缘和背景分别用灰度级 L_G 和 L_B 表示，生成二值图像，便于研究边缘所在位置



Laplacian 算子

- ✓ Laplacian 算子是线性二阶微分算子。即

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

- ✓ 对离散的数字图像而言，二阶偏导数可用二阶差分近似，可推导出Laplacian算子表达式为

$$\begin{aligned}\nabla^2 f(x, y) = & \\ & f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)\end{aligned}$$

Laplacian 算子 模板

0	1	0
1	-4	1
0	1	0
0	-1	0
-1	4	-1
0	-1	0

a b
c d

FIGURE 3.39

(a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4).

(b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Laplacian增强算子

- ✓ Laplacian算子作为二阶微分算子，强调的是图像中灰度的突变，并且淡化图像中灰度变化缓慢的区域
- ✓ 因此拉普拉斯算子产生的图像是以浅色显示强边缘图像中的边缘和突变点，而以黑色或暗灰色显示原图中的其他背景区域
- ✓ 将拉普拉斯算子产生的图像叠加到原图像中可以使得图像中的边界和突变细节得到锐化，同时也保留了原图像中的信息

Laplacian增强算子

- ✓ Laplacian增强算子：将拉普拉斯算子产生的图像叠加到原图像

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) & \text{Laplace掩膜中心系数为负} \\ f(x, y) + \nabla^2 f(x, y) & \text{Laplace掩膜中心系数为正} \end{cases}$$

拉普拉斯增强算子模板

0	-1	0
-1	5	-1
0	-1	0

$$g(x, y) =$$

$$5f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)]$$

拉普拉斯增强

a
b
c
d

FIGURE 3.40

(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)



a) 月球北极图像

b) Laplace算子产生的图像可能产生负数

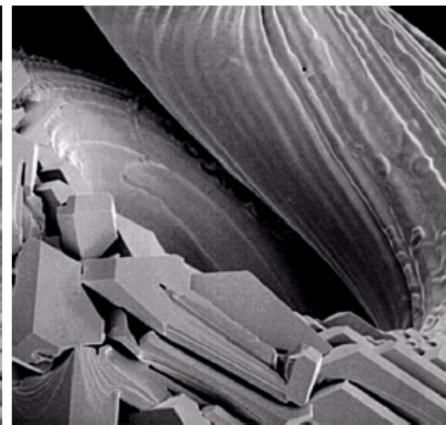
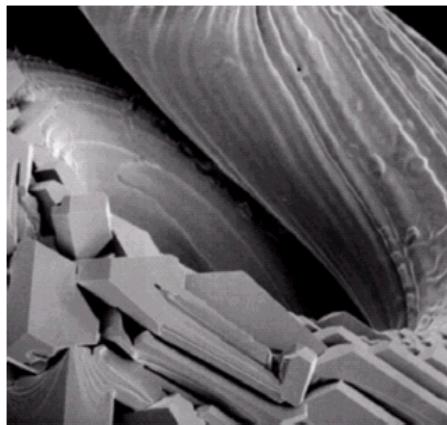
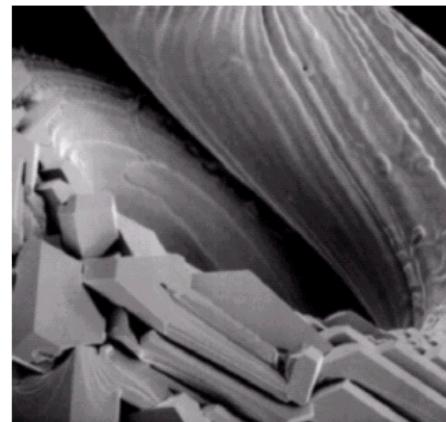
c) 图像重新标定后的图像

d) 增强后的图像

合成拉普拉斯掩膜

0	-1	0
-1	5	-1
0	-1	0

-1	-1	-1
-1	9	-1
-1	-1	-1



a b c
d e

FIGURE 3.41 (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

高频提升滤波

0	-1	0
-1	$A + 4$	-1
0	-1	0

-1	-1	-1
-1	$A + 8$	-1
-1	-1	-1

a b

FIGURE 3.42 The high-boost filtering technique can be implemented with either one of these masks, with $A \geq 1$.

设锐化图像： $f_s(x, y) = f(x, y) - \bar{f}(x, y) \quad A \geq 1$

$$\begin{aligned}f_{hb}(x, y) &= Af(x, y) - \bar{f}(x, y) \\&= (A-1)f(x, y) + f(x, y) - \bar{f}(x, y) \\&= (A-1)f(x, y) + f_s(x, y)\end{aligned}$$

高频提升滤波

0	-1	0
-1	$A + 4$	-1
0	-1	0

-1	-1	-1
-1	$A + 8$	-1
-1	-1	-1

a b

FIGURE 3.42 The high-boost filtering technique can be implemented with either one of these masks, with $A \geq 1$.

选择拉普拉斯增强选择作为锐化图像：

$$f_s(x, y) = f(x, y) - \nabla^2 f(x, y)$$

$$f_{hb}(x, y) = Af(x, y) - \nabla^2 f(x, y)$$

提升滤波应用之一是处理较暗的输入图像，
通过设置不同的A，起到增亮原图像且锐化细节

a b
c d

FIGURE 3.43

(a) Same as Fig. 3.41(c), but darker.

(a) Laplacian of (a) computed with the mask in Fig. 3.42(b) using $A = 0$.

(c) Laplacian enhanced image using the mask in Fig. 3.42(b) with $A = 1$. (d) Same as (c), but using $A = 1.7$.

