Lecture 7 – Spatial Filtering（空间滤波）

This lecture will cover:

– Correlation(相关) and Convolution (卷积)

– Spatial Filtering（空间滤波器）

✓ Smoothing（平滑）

✓ Sharpening（锐化）

– Color spatial filtering（彩色滤波）

Spatial Filtering

A Spatial filter

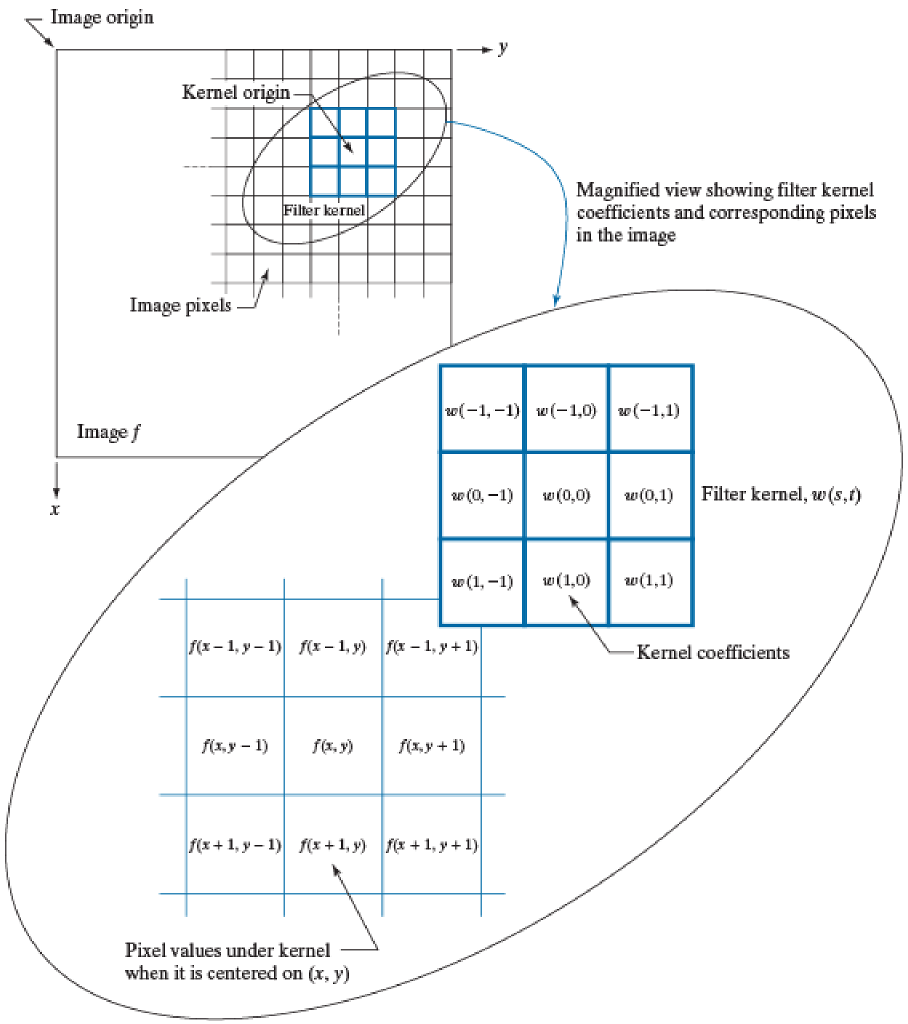
- **is directly applied on the image**
- **is also called spatial masks(掩模), kernels(核), templates(模板), windows(窗口)**
- **consists of**

  1) neighborhood 2) a predefined operation

- **can be linear and nonlinear**
  - Linear spatial filter corresponds to spectral filter in frequency domain
  - Nonlinear spatial filter cannot be accomplished in frequency domain

Spatial Filter

FIGURE 3.34 The mechanics of linear spatial filtering using a 3×3 kernel. The pixels are shown as squares to simplify the graphics. Note that the origin of the image is at the top left, but the origin of the kernel is at its center. Placing the origin at the center of spatially symmetric kernels simplifies writing expressions for linear filtering.



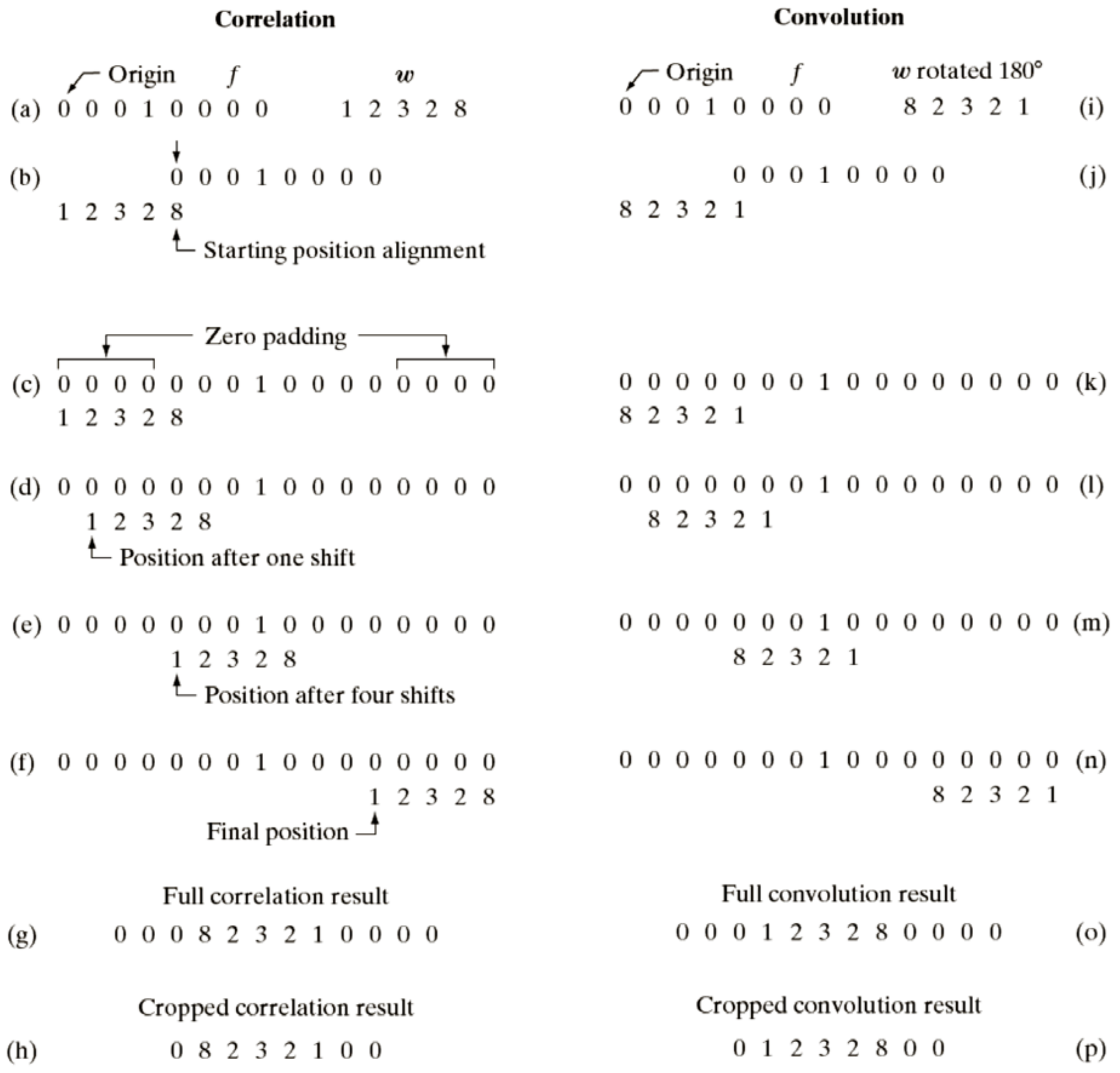
$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

- $f(x,y)$: input image
- $g(x,y)$: output filtered image
- $w(s,t)$: $m \times n$ spatial filter, where $m = 2a+1, n = 2b+1$

上海科技大学 ShanghaiTech University

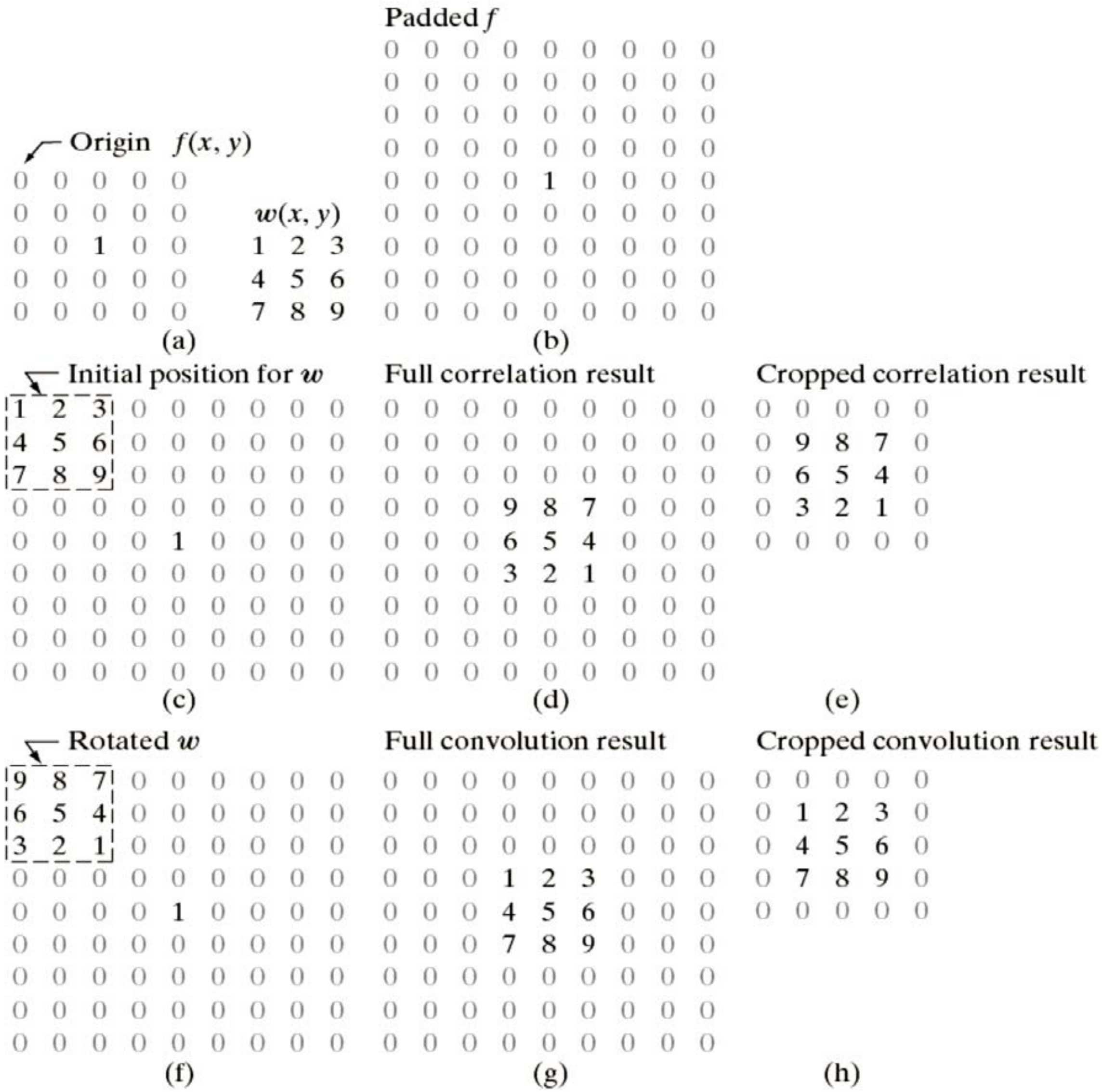
Correlation(相关) and Convolution(卷积) (1D)

FIGURE 3.35 Illustration of 1-D correlation and convolution of a kernel, *w*, with a function *f* consisting of a discrete unit impulse. Note that correlation and convolution are functions of the variable *x*, which acts to *displace* one function with respect to the other. For the extended correlation and convolution results, the starting configuration places the right-most element of the kernel to be coincident with the origin of *f*. Additional padding must be used.



Correlation and Convolution (2D)

FIGURE 3.36 Correlation (middle row) and convolution (last row) of a 2-D kernel with an image consisting of a discrete unit impulse. The 0's are shown in gray to simplify visual analysis. Note that correlation and convolution are functions of x and y. As these variable change, they *displace* one function with respect to the other. See the discussion of Eqs. (3-45) and (3-46) regarding full correlation and convolution.



Equations

Correlation:

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

Convolution

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

Vector Operation

$$R = w_1 z_k + w_2 z_2 + \cdots + w_{mn} z_{mn} = \sum_{k=1}^{mn} w_k z_k = w^T z$$

TABLE 3.5
Some fundamental properties of convolution and correlation. A dash means that the property does not hold.

Property	Convolution	Correlation
Commutative	$f \bigstar g = g \bigstar f$	—
Associative	$f \bigstar (g \bigstar h) = (f \bigstar g) \bigstar h$	—
Distributive	$f \bigstar (g+h) = (f \bigstar g) + (f \bigstar h)$	$f \star (g+h) = (f \star g) + (f \star h)$

Separable filter kernels:

$$w = VW^T = V \bigstar W^T$$

or $\quad w = VV^T = V \bigstar V^T$

Where V and W are vectors of size m*1 and n*1

The convolution can be written as

$$\begin{aligned} w \bigstar f &= \left(V \bigstar W^T\right) \bigstar f \\ &= \left(W^T \bigstar V\right) \bigstar f \\ &= W^T \bigstar (V \bigstar f) \\ &= (V \bigstar f) \bigstar W^T \end{aligned}$$

The computational advantage:

$$C = \frac{MNmn}{MN(m+n)} = \frac{mn}{(m+n)}$$

Spatial Filter Masks

- **Linear Spatial Filter (线性滤波器):**
  define the coefficients by
  - Constant: $R = \frac{1}{9}\sum_{k=1}^{9} z_k$
  - Coordinate: $h(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$

a b

FIGURE 3.37 Examples of smoothing kernels: (a) is a *box* kernel; (b) is a *Gaussian* kernel.

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{4.8976} \times \begin{bmatrix} 0.3679 & 0.6065 & 0.3679 \\ 0.6065 & 1.0000 & 0.6065 \\ 0.3679 & 0.6065 & 0.3679 \end{bmatrix}$$

- **Nonlinear Spatial Filter (非线性滤波器):**
  determine the operation
  - Max filter (最大值滤波)
  - Median filter （中值滤波）

f_1

1	8	2
2	4	7
3	5	6

f_2

9	3	8
2	6	7
4	0	3

f_1+f_2

10	11	10
4	10	14
7	5	9

$\max(f_1)+\max(f_2) \neq \max(f_1+f_2)$

Smoothing Filters (平滑滤波器)

- **Smoothing filters are used for**
  - *Blurring for preprocessing tasks:* removing small details before object extraction, bridging of small gaps, etc
  - *Noise deduction:*
- **Types of filters**
  - *Linear filter:* average filtering – lowpass filter in frequency domain
  - *Nonlinear filter*

Smoothing Filters (平滑滤波器)

- General form of box filters

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

$\frac{1}{9}\times$

1	1	1
1	1	1
1	1	1

$\frac{1}{16}\times$

1	2	1
2	4	2
1	2	1

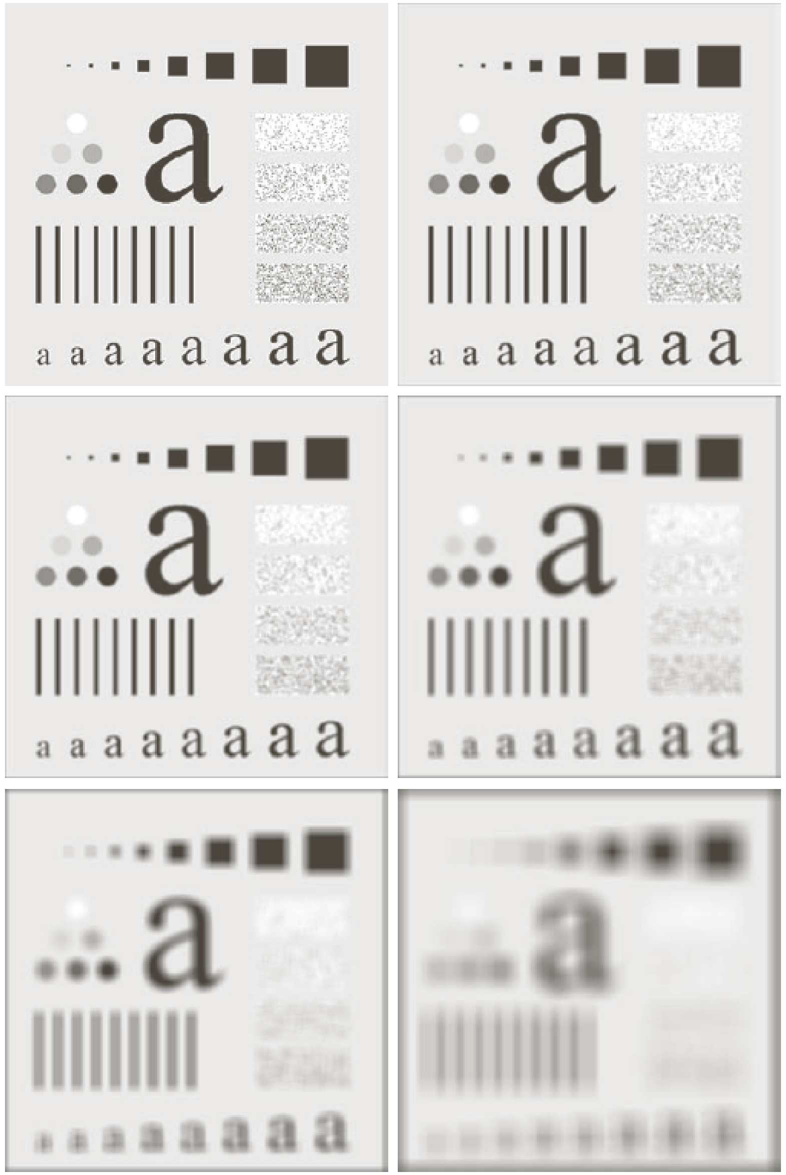


FIGURE 3.33 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes m = 3, 5, 9, 15, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50 × 120 pixels.

a b
c d
e f

Gaussian smoothing Filters (高斯平滑滤波器)

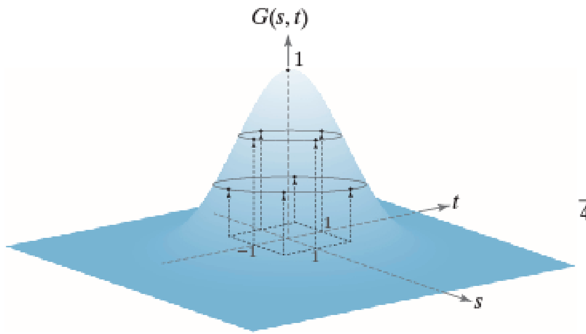
- General form of Gaussian filters

$$w(s,t) = G(s,t) = Ke^{-\frac{s^2+t^2}{2\sigma^2}}$$

$$G(r) = Ke^{-\frac{r^2}{2\sigma^2}}$$

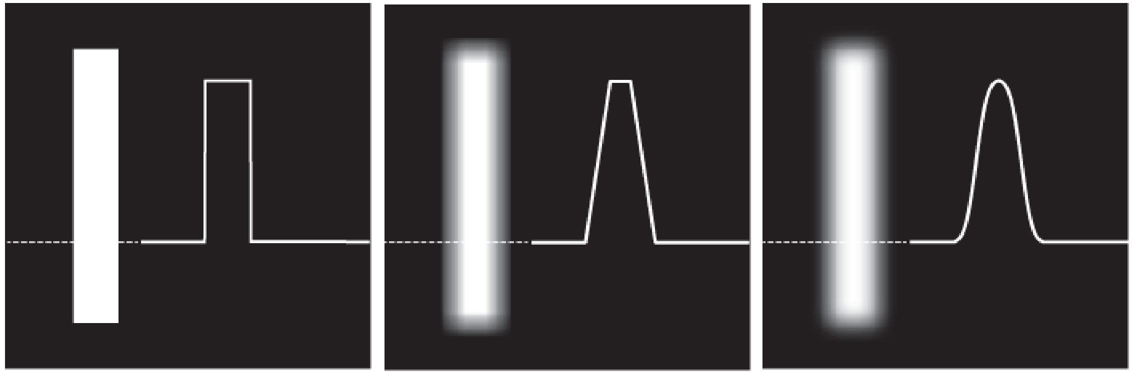
a b

FIGURE 3.41 (a) Sampling a Gaussian function to obtain a discrete Gaussian kernel. The values shown are for $K = 1$ and $\sigma = 1$. (b) Resulting 3×3 kernel [this is the same as Fig. 3.37(b)].



$\frac{1}{4.8976} \times$

0.3679	0.6065	0.3679
0.6065	1.0000	0.6065
0.3679	0.6065	0.3679



a b c

FIGURE 3.44 (a) Image of a white rectangle on a black background, and a horizontal intensity profile along the scan line shown dotted. (b) Result of smoothing this image with a box kernel of size 71×71, and corresponding intensity profile. (c) Result of smoothing the image using a Gaussian kernel of size 151×151, with $K = 1$ and $\sigma = 25$. Note the smoothness of the profile in (c) compared to (b). The image and rectangle are of sizes 1024×1024 and 768×128 pixels, respectively.

Smoothing Filter and Thresholding(阈值处理)

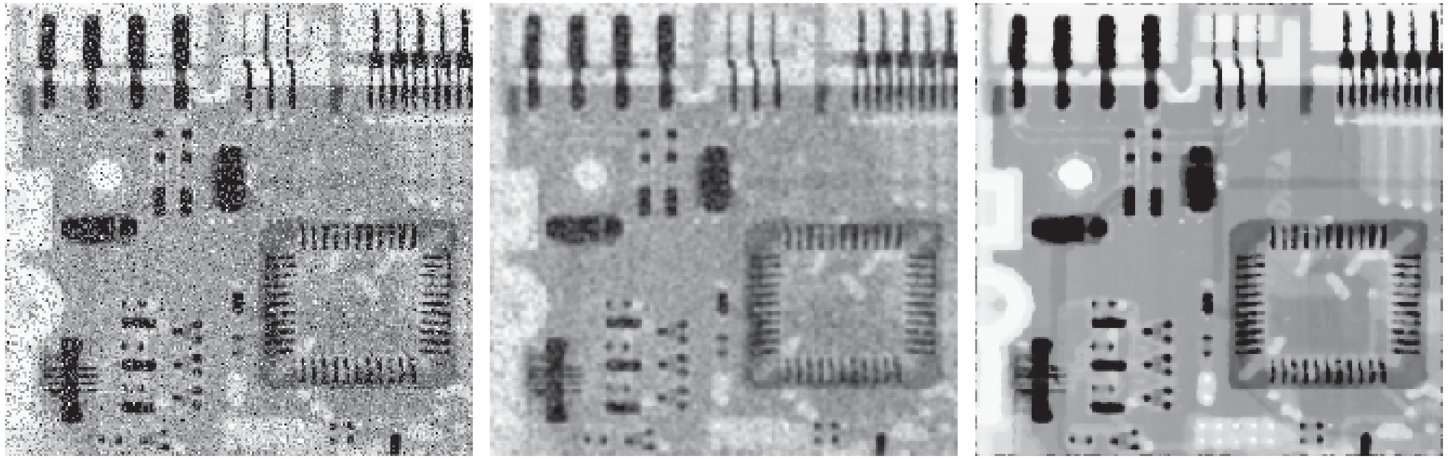
a b c

FIGURE 3.47 (a) A 2566 × 2758 Hubble Telescope image of the *Hickson Compact Group*. (b) Result of lowpass filtering with a Gaussian kernel. (c) Result of thresholding the filtered image (intensities were scaled to the range [0, 1]). The Hickson Compact Group contains dwarf galaxies that have come together, setting off thousands of new star clusters. (Original image courtesy of NASA.)

Nonlinear Smoothing Filters

- **Order-statistic filter (统计排序滤波器)**

$$R = H\{z_k | k = 1,2,\cdots mn\}$$



a b c

FIGURE 3.49 (a) X-ray image of a circuit board, corrupted by salt-and-pepper noise. (b) Noise reduction using a 19×19 Gaussian lowpass filter kernel with $\sigma = 3$. (c) Noise reduction using a 7×7 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening Filter

- To highlight transitions in intensity

- Accomplished by spatial differentiation

  - First-order derivative: $\frac{\partial f}{\partial x} = f(x+1) - f(x)$

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

- Sharpening filter

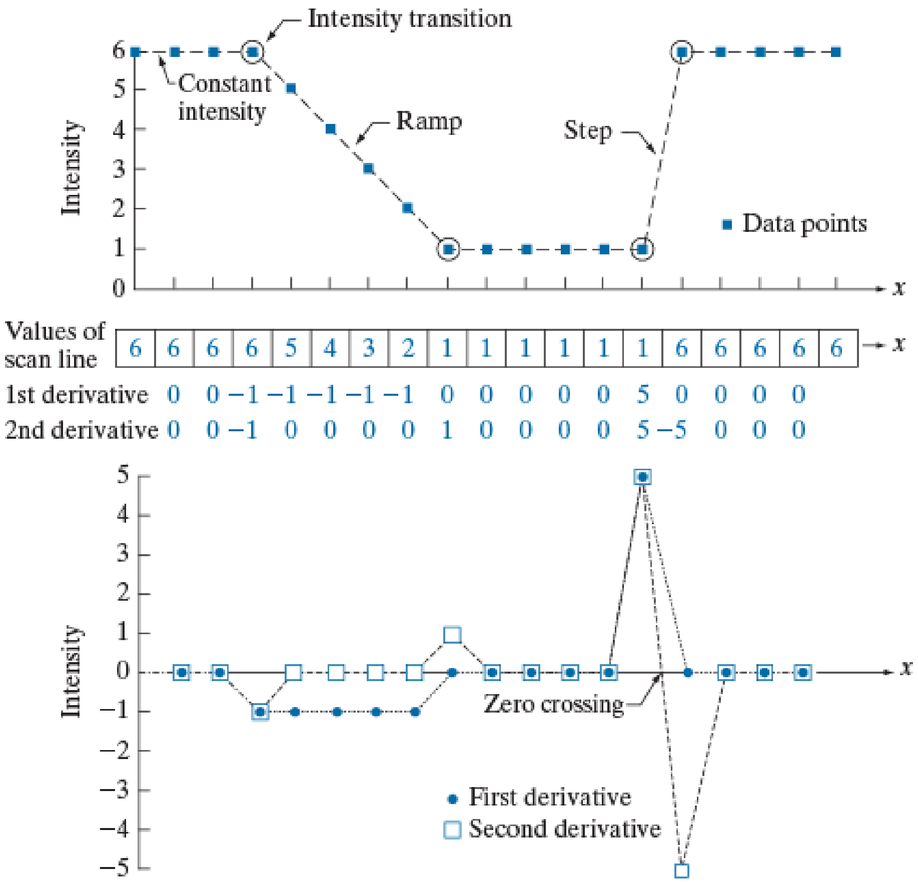
  - Laplacian filtering（拉普拉斯算子）
  - Unsharp Masking（非锐化掩蔽）
  - Gradient filtering（梯度算子）

Sharpening Filter

The definition used for derivative:

1. Zero in area of constant intensity
2. Nonzero at the onset of intensity step or ramp
3. (1) Nonzero along intensity ramp – 1st order derivative

   (2) Zero along intensity ramp with constant slope – 2nd order derivative



a
b
c

FIGURE 3.50
(a) A section of a horizontal scan line from an image, showing ramp and step edges, as well as constant segments.
(b)Values of the scan line and its derivatives.
(c) Plot of the derivatives, showing a zero crossing. In (a) and (c) points were joined by dashed lines as a visual aid.

Laplacian(拉普拉斯算子)

For an image function $\boldsymbol{f(x, y)}$,

X direction： $\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$

Y direction： $\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) + f(x+1, y) + f(x-1, y) - 4f(x, y)$$

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

a b c d

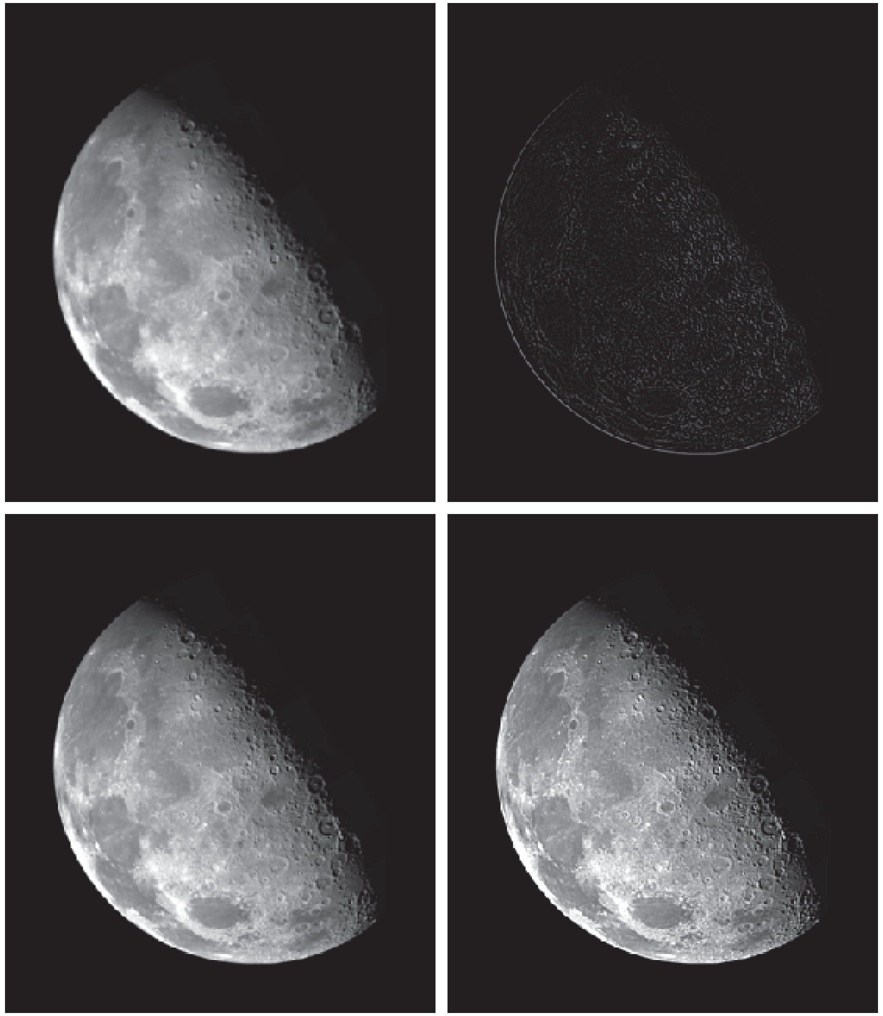
FIGURE 3.51 (a) Laplacian kernel used to implement Eq. (3-62). (b) Kernel used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other Laplacian kernels.

Laplacian Filter Masks

$$g(x, y) = f(x, y) + c\nabla^2 f(x, y), \text{where } c = \pm 1$$

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

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



a b
c d

FIGURE 3.52 (a) Blurred image of the North Pole of the moon. (b) Laplacian image obtained using the kernel in Fig. 3.51(a). (c) Image sharpened using Eq. (3-63) with $c = -1$. (d) Image sharpened using the same procedure, but with the kernel in Fig. 3.51(b). (Original image courtesy of NASA.)

上海科技大学 ShanghaiTech University

Unsharp Masking (非锐化掩蔽)

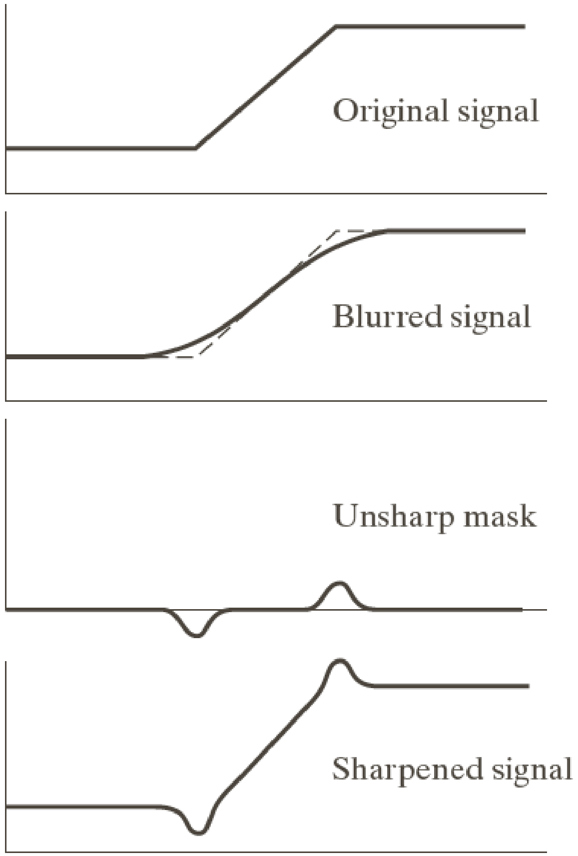
- **The steps:**

1. Blur the original image;
2. Subtract the blurred image from the original (the resulting difference is called the **mask**);

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

3. Add the mask to the original

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



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).

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.

上海科技大学 ShanghaiTech University

Gradient (梯度)

- The first-order derivative of $f(x,y)$: $\nabla f \equiv \mathrm{grad}(f) \equiv \begin{cases} g_x \\ g_y \end{cases} = \begin{cases} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{cases}$

- The amplitude: $M(x,y) = \mathrm{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2}$ or $M(x,y) \approx |g_x| + |g_y|$

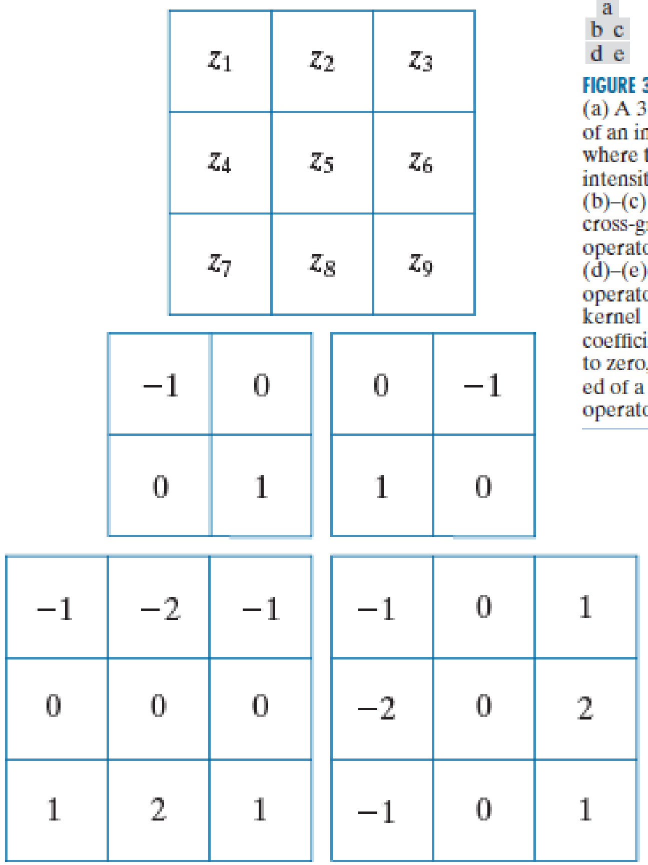
- Definition of the first order derivative :

  - Roberts cross-gradient operator (罗伯特交叉梯度算子)

$$M(x,y) = |z_9 - z_5| + |z_8 - z_6|$$

  - Sobel operator (Sobel算子)

$$M(x,y) = |(z_7+2z_8 + z_9) - (z_1+2z_2+z_3)| \\ +|(z_3+2z_6 + z_9) - (z_1+2z_4+z_7)|$$



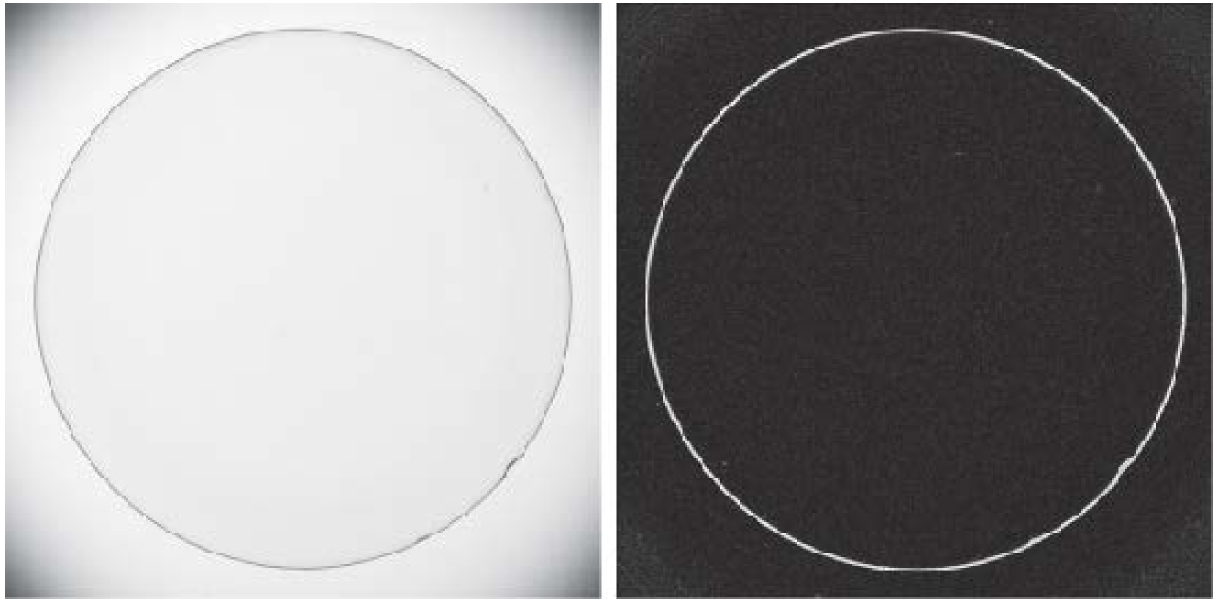
a
b c
d e

FIGURE 3.56 (a) A 3×3 region of an image, where the zs are intensity values. (b)–(c) Roberts cross-gradient operators. (d)–(e) Sobel operators. All the kernel coefficients sum to zero, as expected of a derivative operator.

Edge Enhancement

a b

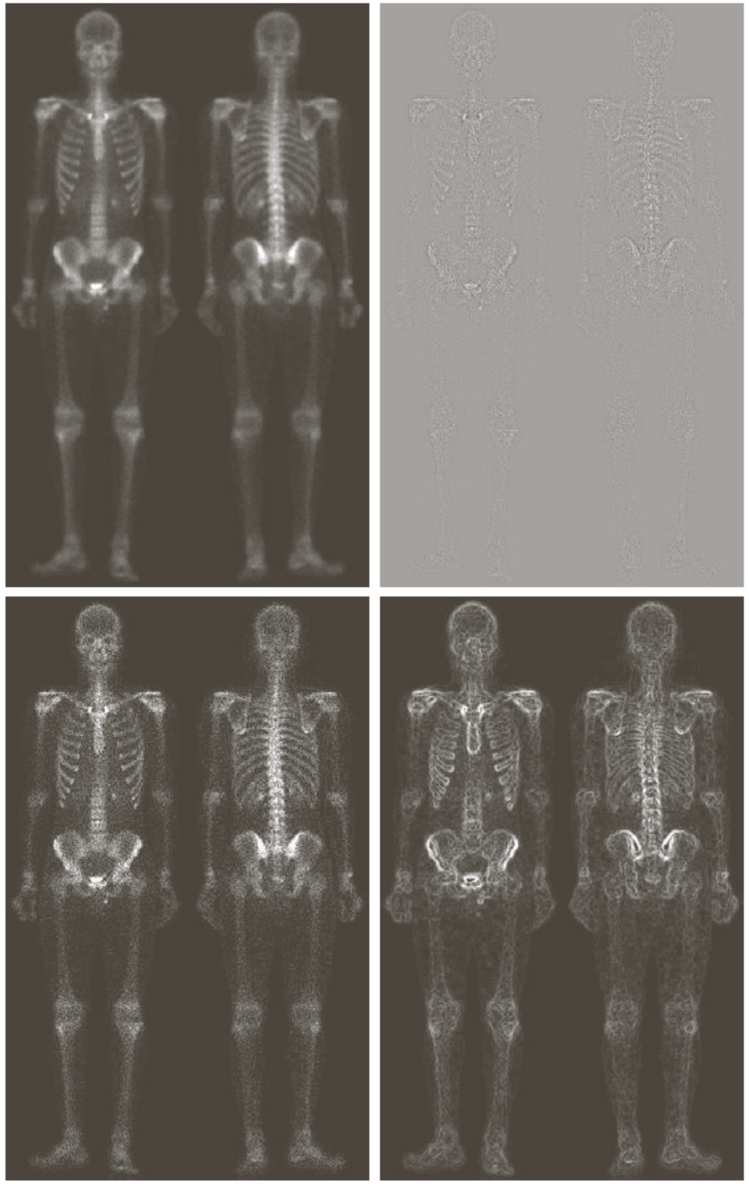
FIGURE 3.57
(a) Image of a contact lens (note defects on the boundary at 4 and 5 o'clock).
(b) Sobel gradient.
(Original image courtesy of Perceptics Corporation.)

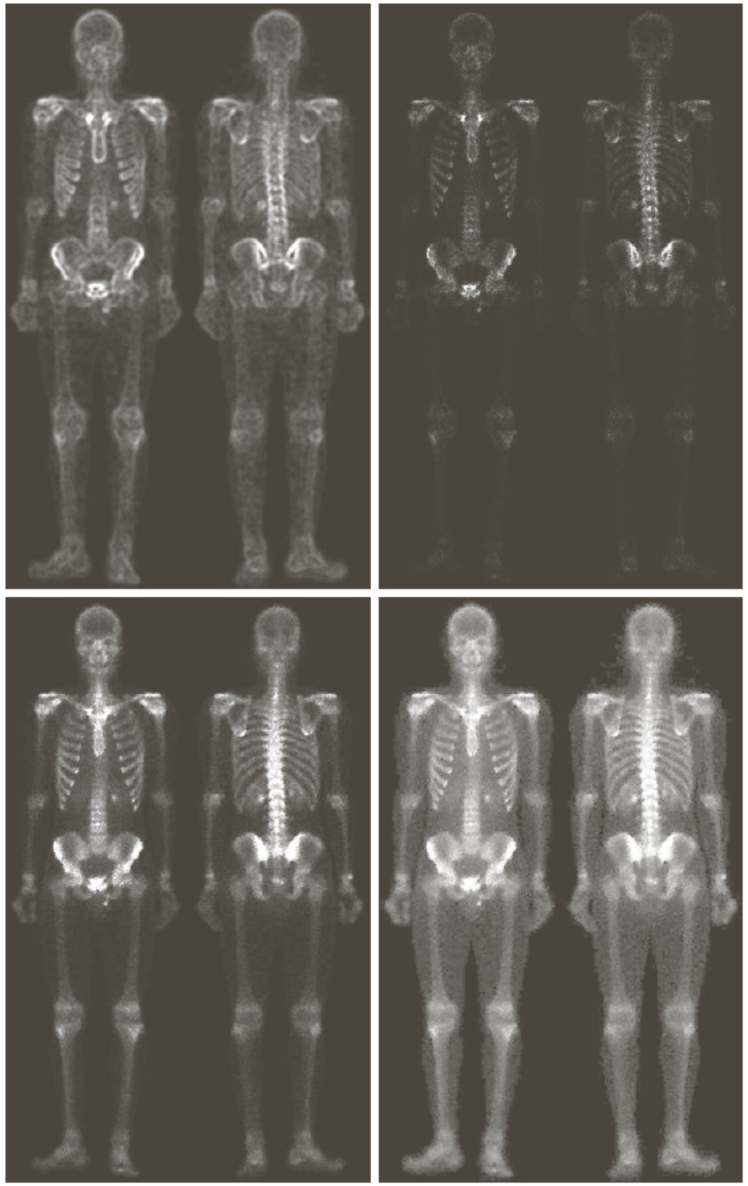


Combined Enhancement Methods

a b
c d

FIGURE 3.63
(a) Image of whole body bone scan.
(b) Laplacian of (a).
(c) Sharpened image obtained by adding (a) and (b).
(d) Sobel gradient of image (a). (Original image courtesy of G.E. Medical Systems.)





e f
g h

FIGURE 3.63
(Continued)
(e) Sobel image smoothed with a 5×5 box filter.
(f) Mask image formed by the product of (b) and (e).
(g) Sharpened image obtained by the adding images (a) and (f).
(h) Final result obtained by applying a power-law transformation to (g). Compare images (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

上海科技大学 ShanghaiTech University

Color image smoothing

- Neighborhood averaging by color components

$$\bar{c}(x,y)=\frac{1}{K}\sum_{(s,t)\in S_{xy}} c(s,t)$$

$$=\begin{bmatrix}\frac{1}{K}\sum_{(s,t)\in S_{xy}} R(s,t)\\ \frac{1}{K}\sum_{(s,t)\in S_{xy}} G(s,t)\\ \frac{1}{K}\sum_{(s,t)\in S_{xy}} B(s,t)\end{bmatrix}$$

a b
c d

FIGURE 7.36
(a) RGB image.
(b) Red component image.
(c)Green component.
(d) Blue component.

Color image smoothing

➢ Neighborhood averaging by intensity components

$$\bar{c}(x,y) = \bar{I}(x,y) = \frac{1}{K}\sum_{(s,t)\in S_{xy}} I(s,t)$$

a b c

FIGURE 7.37 HSI components of the RGB color image in Fig. 7.36(a). (a) Hue. (b) Saturation. (c) Intensity.

Color image smoothing

- HIS color model decouples intensity and color information, and is more suitable for many gray-scale processing techniques.

a b c

FIGURE 7.38 Image smoothing with a 5×5 averaging kernel. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

上海科技大学 ShanghaiTech University

Color image sharpening

- Neighborhood averaging by color components or intensity

$$\nabla^2 c(x,y) = \begin{bmatrix} \nabla^2 R(x,y) \\ \nabla^2 G(x,y) \\ \nabla^2 B(x,y) \end{bmatrix} \quad \text{or} \quad \nabla^2 c(x,y) = \nabla^2 I(x,y)$$

a b c

FIGURE 7.39 Image sharpening using the Laplacian. (a) Result of processing each RGB channel. (b) Result of processing the HSI intensity component and converting to RGB. (c) Difference between the two results.