Filtering in the Frequency Domain-2

2-D Discrete Fourier Transform and Its Inverse

DFT:

$$F(\mu,\nu)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(\mu x/M+\nu y/N)}$$

$$\mu=0,1,2,\ldots,M-1;\nu=0,1,2,\ldots,N-1;$$

$f(x,y)$ is a digital image of size M × N.

IDFT:

$$f(x,y)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(\mu,\nu)e^{j2\pi(\mu x/M+\nu y/N)}$$

Properties of the 2-D DFT

relationships between spatial and frequency intervals

Let ΔT and ΔZ denote the separations between samples, then the seperations between the corresponding discrete, frequency domain variables are given by

$$\Delta\mu = \frac{1}{M\Delta T}$$

and $$\Delta v = \frac{1}{N\Delta Z}$$

Properties of the 2-D DFT

translation and rotation

$$f(x,y)e^{j2\pi(\mu_0 x/M+\nu_0 y/N)} \Leftrightarrow F(\mu-\mu_0,\nu-\nu_0)$$

and

$$f(x-x_0,y-y_0) \Leftrightarrow F(\mu,\nu)e^{-j2\pi(\mu x_0/M+\nu y_0/N)}$$

Using the polar coordinates

$x = r\cos\theta \quad \mathrm{y{=}rsin}\theta \quad \mu{=}\omega\cos\varphi \quad \nu{=}\omega\sin\varphi$

results in the following transform pair:

$$f(r,\theta+\theta_0) \Leftrightarrow F(\omega,\varphi+\theta_0)$$

Properties of the 2-D DFT

periodicity

$2-D$ Fourier transform and its inverse are infinitely periodic

$$F(\mu,\nu)=F(\mu+k_1M,\nu)=F(\mu,\nu+k_2N)=F(\mu+k_1M,\nu+k_2N)$$

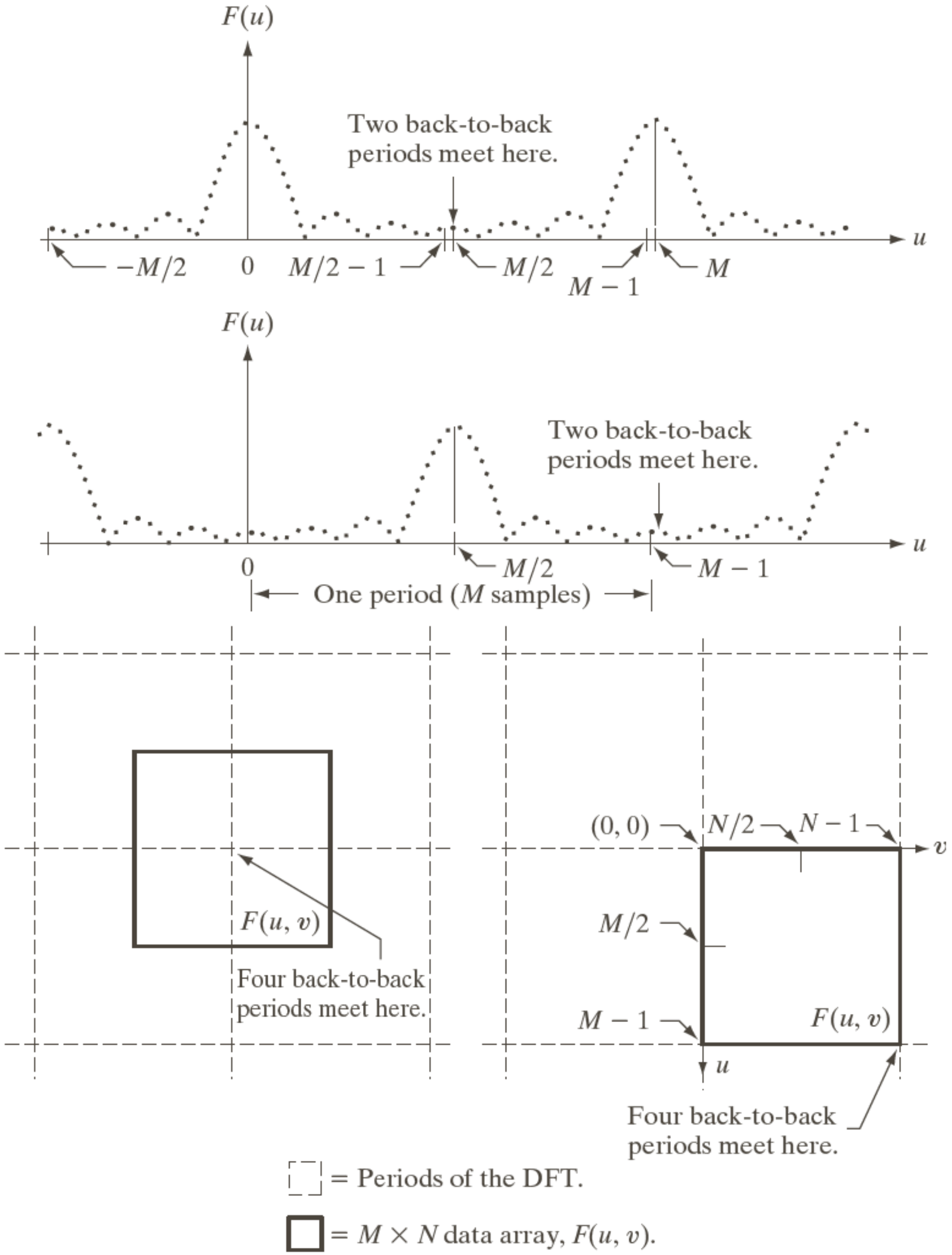
$$f(x,y)=f(x+k_1M,y)=f(x,y+k_2N)=f(x+k_1M,y+k_2N)$$

$$f(x)e^{j2\pi(\mu_0x/M)}\Leftrightarrow F(\mu-\mu_0)$$

$$\mu_0=M/2,\quad f(x)(-1)^x\Leftrightarrow F(\mu-M/2)$$

$$f(x,y)(-1)^{x+y}\Leftrightarrow F(\mu-M/2,\nu-N/2)$$

Prope
periodi

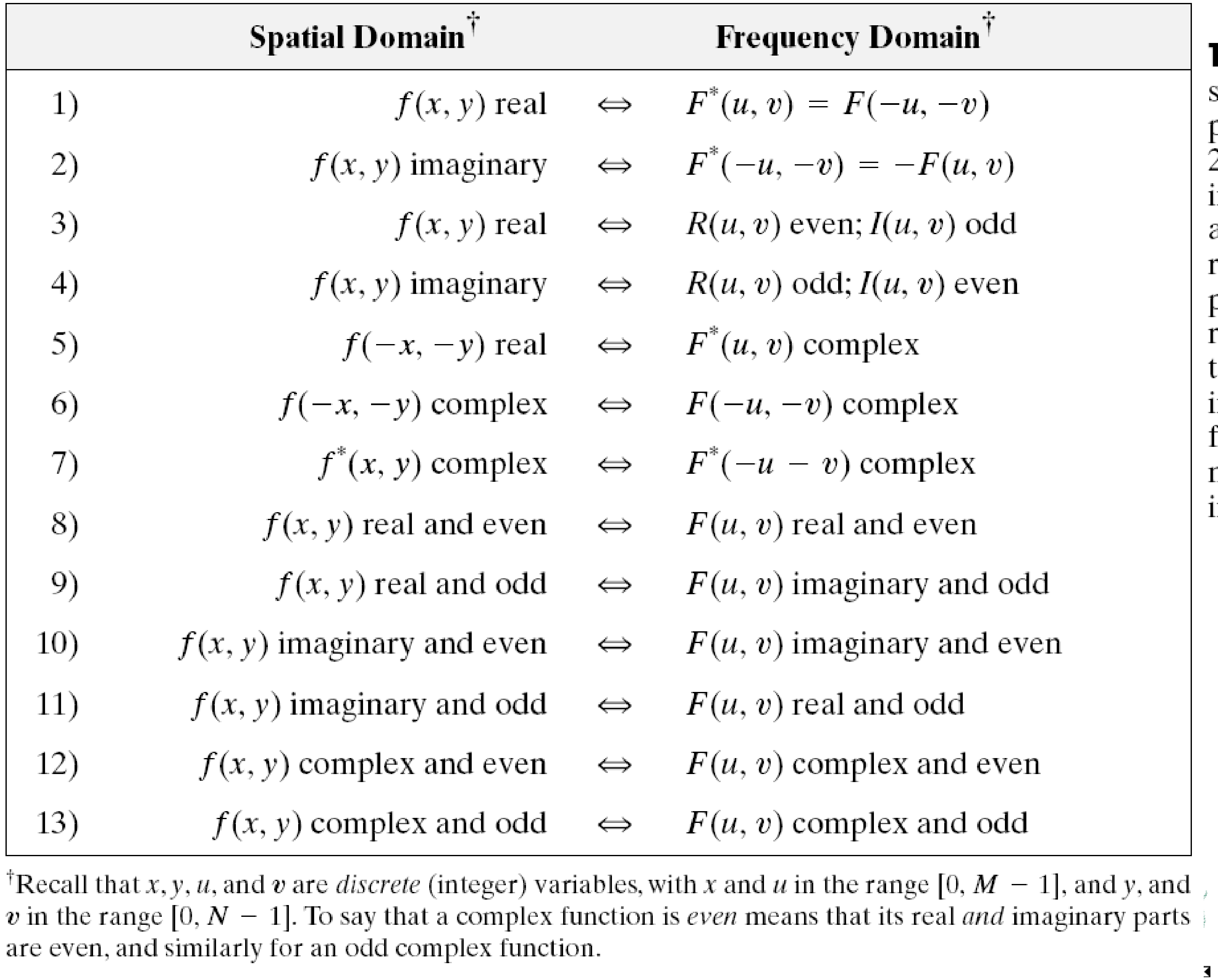

F(u)
Two back-to-back periods meet here.
u
−M/2
0
M/2 − 1
M/2
M − 1
M
F(u)
Two back-to-back periods meet here.
u
0
M/2
M − 1
One period (M samples)
F(u, v)
Four back-to-back periods meet here.
(0, 0)
N/2
N − 1
v
M/2
M − 1
F(u, v)
u
Four back-to-back periods meet here.
= Periods of the DFT.
= M × N data array, F(u, v).

a
b
c d

FIGURE 4.23 Centering the Fourier transform. (a) A 1-D DFT showing an infinite number of periods. (b) Shifted DFT obtained by multiplying $f(x)$ by $(-1)^x$ before computing $F(u)$. (c) A 2-D DFT showing an infinite number of periods. The solid area is the $M \times N$ data array, $F(u, v)$, obtained with Eq. (4.5-15). This array consists of four quarter periods. (d) A Shifted DFT obtained by multiplying $f(x, y)$ by $(-1)^{x+y}$ before computing $F(u, v)$. The data now contains one complete, centered period, as in (b).

Properties of the 2-D DFT

Symmetry



	Spatial Domain[†]		Frequency Domain[†]
1)	$f(x, y)$ real	⇔	$F^*(u, v) = F(-u, -v)$
2)	$f(x, y)$ imaginary	⇔	$F^*(-u, -v) = -F(u, v)$
3)	$f(x, y)$ real	⇔	$R(u, v)$ even; $I(u, v)$ odd
4)	$f(x, y)$ imaginary	⇔	$R(u, v)$ odd; $I(u, v)$ even
5)	$f(-x, -y)$ real	⇔	$F^*(u, v)$ complex
6)	$f(-x, -y)$ complex	⇔	$F(-u, -v)$ complex
7)	$f^*(x, y)$ complex	⇔	$F^*(-u - v)$ complex
8)	$f(x, y)$ real and even	⇔	$F(u, v)$ real and even
9)	$f(x, y)$ real and odd	⇔	$F(u, v)$ imaginary and odd
10)	$f(x, y)$ imaginary and even	⇔	$F(u, v)$ imaginary and even
11)	$f(x, y)$ imaginary and odd	⇔	$F(u, v)$ real and odd
12)	$f(x, y)$ complex and even	⇔	$F(u, v)$ complex and even
13)	$f(x, y)$ complex and odd	⇔	$F(u, v)$ complex and odd

TABLE 4.1 Some symmetry properties of the 2-D DFT and its inverse. $R(u, v)$ and $I(u, v)$ are the real and imaginary parts of $F(u, v)$, respectively. The term *complex* indicates that a function has nonzero real and imaginary parts.

[†] Recall that x, y, u, and v are *discrete* (integer) variables, with x and u in the range $[0, M - 1]$, and y, and v in the range $[0, N - 1]$. To say that a complex function is *even* means that its real *and* imaginary parts are even, and similarly for an odd complex function.

Properties of the 2-D DFT

Fourier Spectrum and Phase Angle

2-D DFT in polar form

$$F(u,v) = |F(u,v)|\, e^{j\phi(u,v)}$$

Fourier spectrum

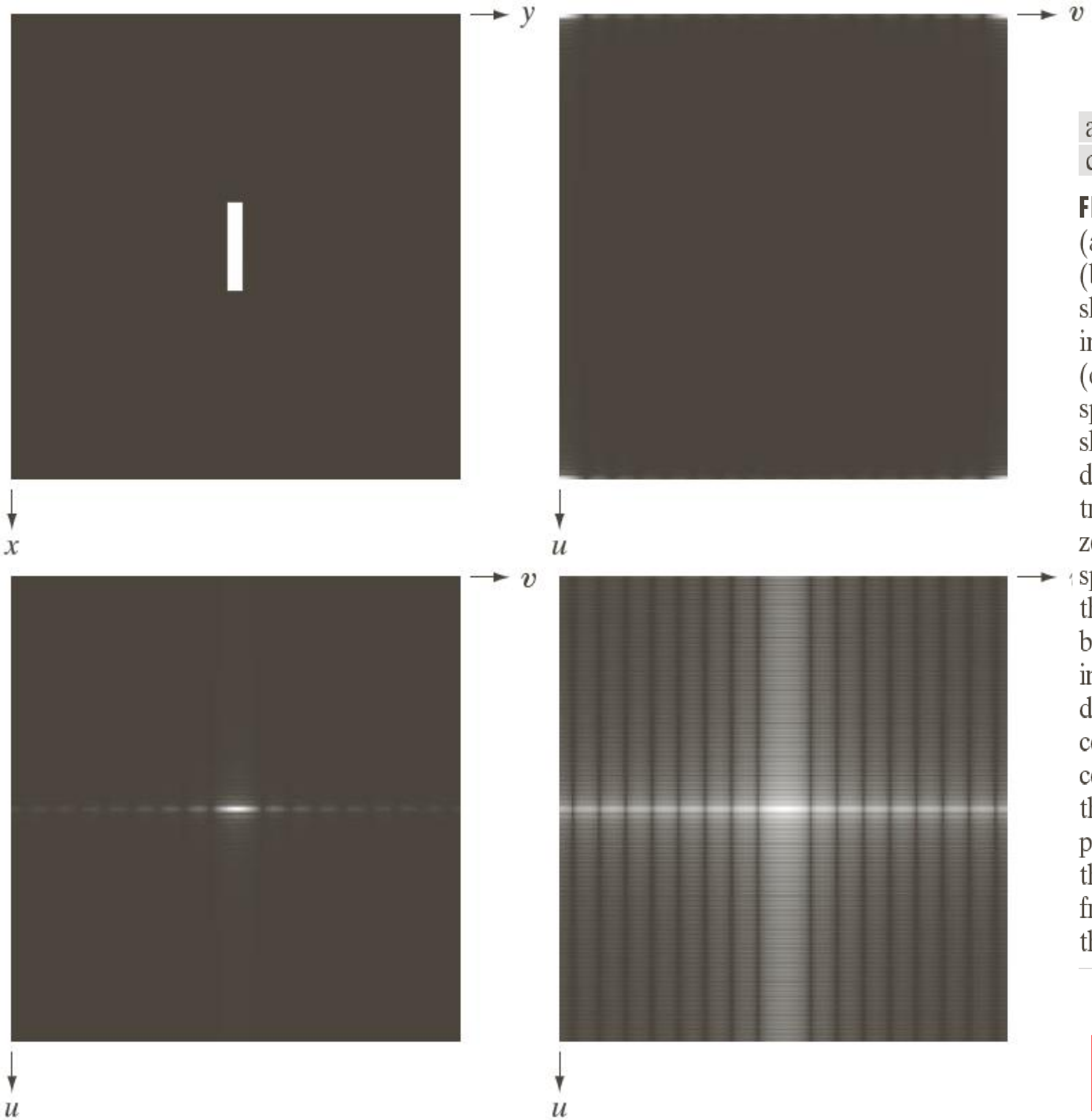
$$|F(u,v)| = \left[R^2(u,v) + I^2(u,v) \right]^{1/2}$$

Power spectrum

$$P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v)$$

Phase angle

$$\phi(\mathrm{u,v}) = \arctan\left[\frac{I(u,v)}{R(u,v)} \right]$$



a b
c d

FIGURE 4.24 (a) Image. (b) Spectrum showing bright spots in the four corners. (c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate convention used throughout the book places the origin of the spatial and frequency domains at the top left.

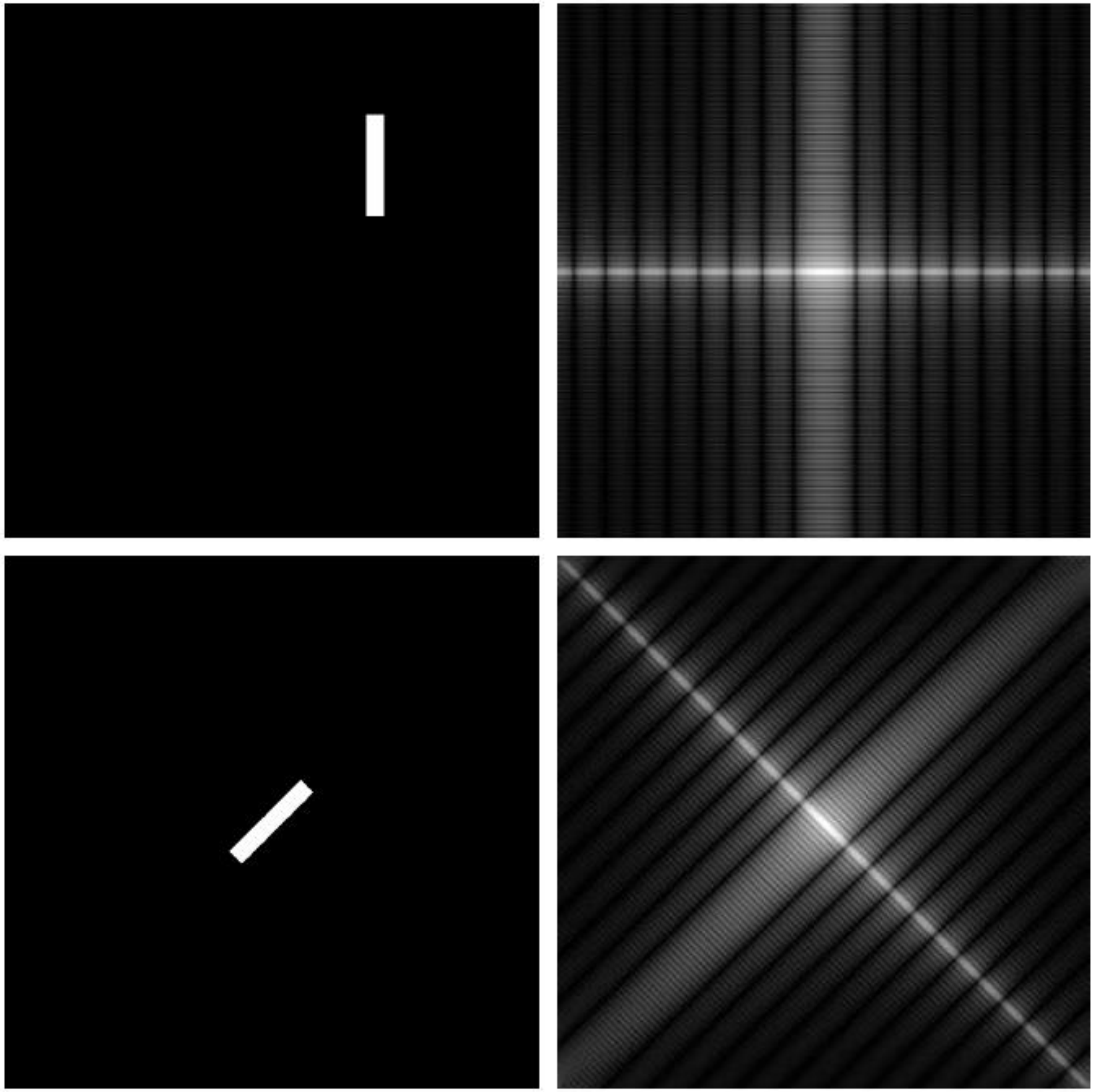


FIGURE 4.25 (a) The rectangle in Fig. 4.24(a) translated, and (b) the corresponding spectrum. (c) Rotated rectangle, and (d) the corresponding spectrum. The spectrum corresponding to the translated rectangle is identical to the spectrum corresponding to the original image in Fig. 4.24(a).

Example: Phase Angles

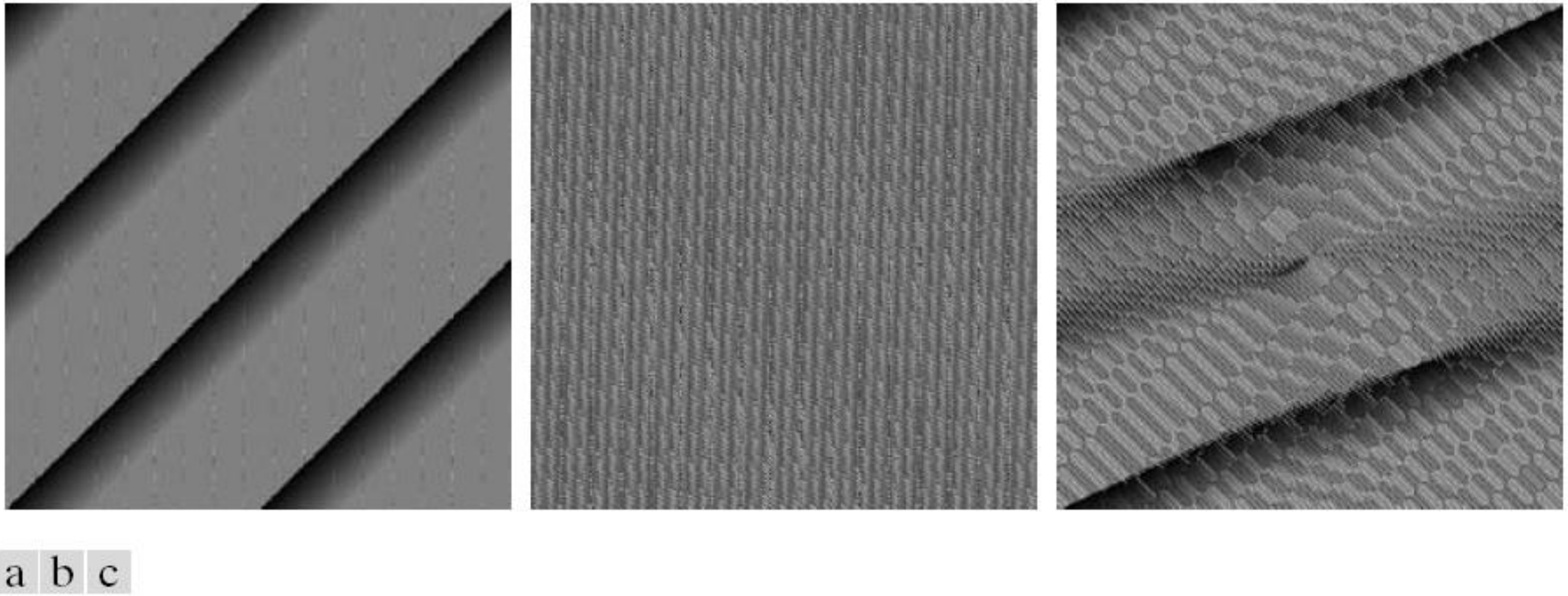


FIGURE 4.26 Phase angle array corresponding (a) to the image of the centered rectangle in Fig. 4.24(a), (b) to the translated image in Fig. 4.25(a), and (c) to the rotated image in Fig. 4.25(c).

Example: Phase Angles and The Reconstructed

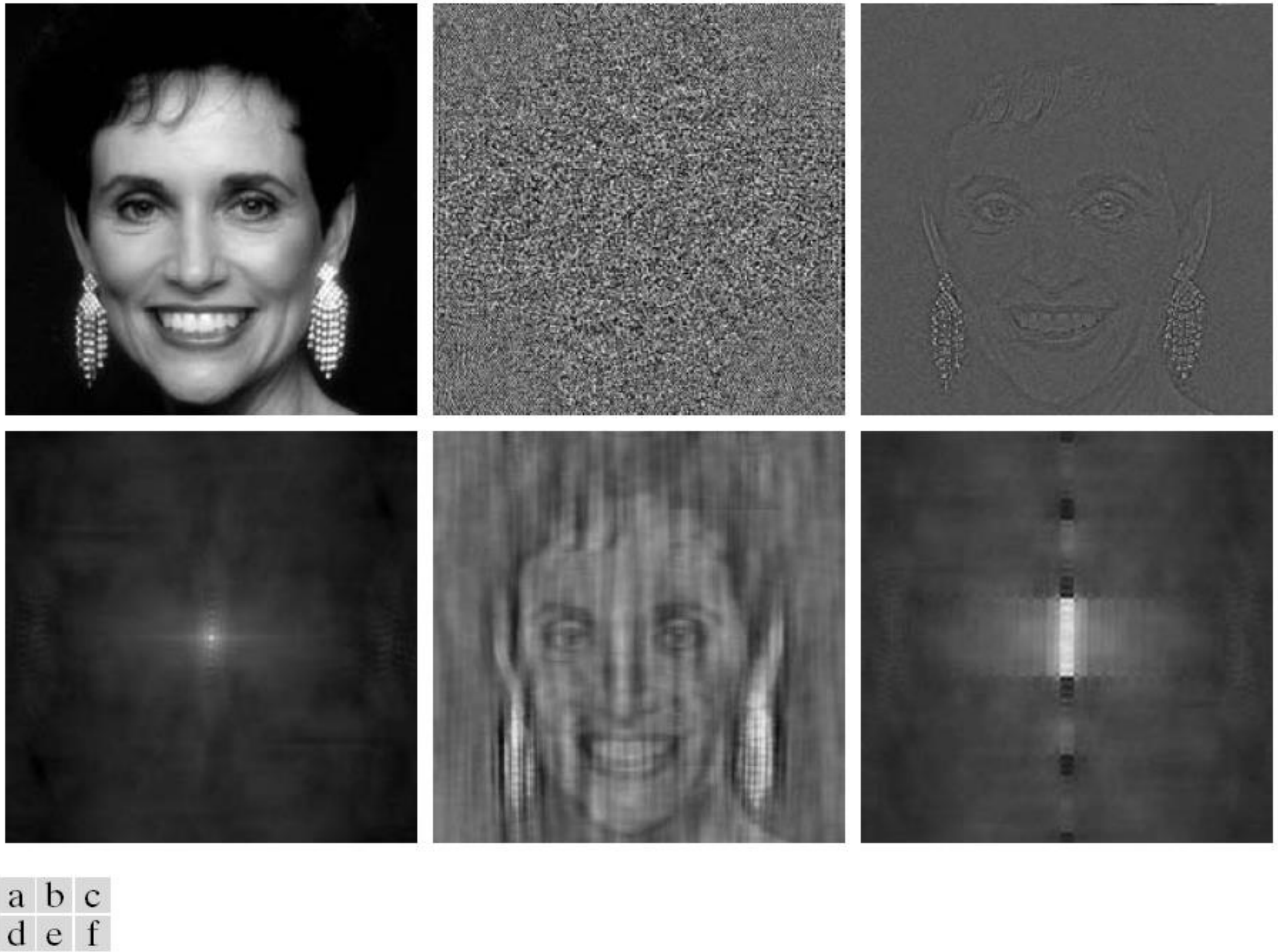


FIGURE 4.27 (a) Woman. (b) Phase angle. (c) Woman reconstructed using only the phase angle. (d) Woman reconstructed using only the spectrum. (e) Reconstruction using the phase angle corresponding to the woman and the spectrum corresponding to the rectangle in Fig. 4.24(a). (f) Reconstruction using the phase of the rectangle and the spectrum of the woman.

2-D Convolution Theorem

1-D convolution

$$f(x)\star h(x) = \sum_{m=0}^{M-1} f(m)h(x-m)$$

2-D convolution

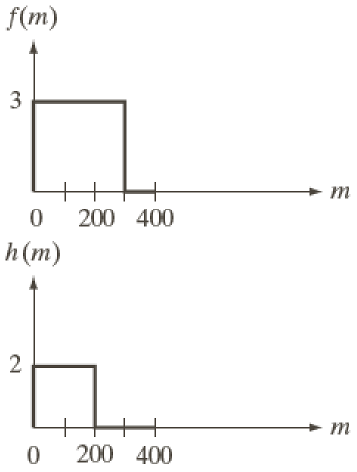
$$f(x,y)\star h(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)$$

$$x = 0,1,2,\ldots,M-1; y = 0,1,2,\ldots,N-1.$$

$$f(x,y)\star h(x,y) \Leftrightarrow F(u,v)H(u,v)$$

$$f(x,y)h(x,y) \Leftrightarrow F(u,v)\star H(u,v)$$

An Example of Convolution



Mirroring h about the origin

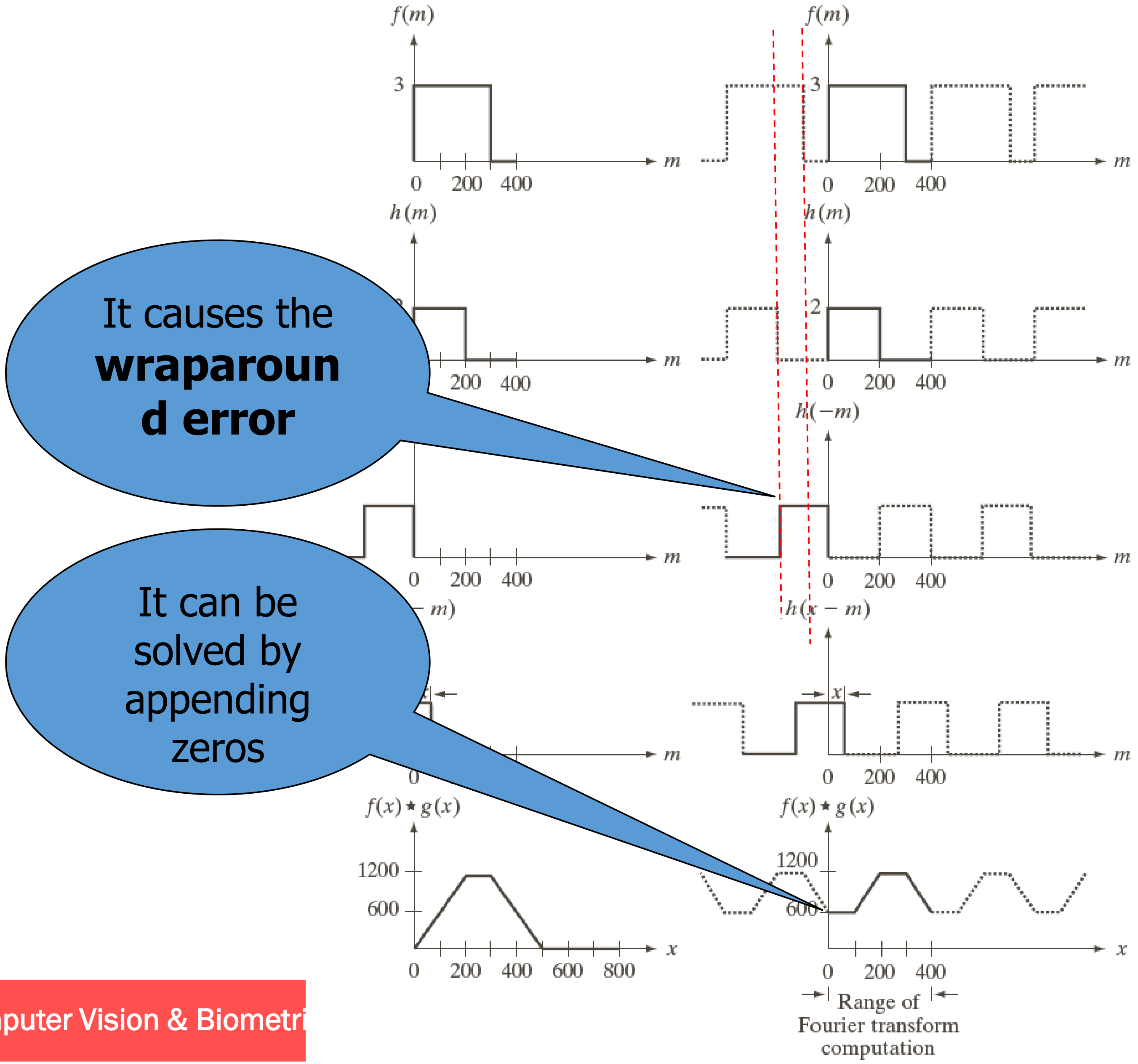
Translating the mirrored function by x

Computing the sum for each x

a	f
b	g
c	h
d	i
e	j

FIGURE 4.28 Left column: convolution of two discrete functions obtained using the approach discussed in Section 3.4.2. The result in (e) is correct. Right column: Convolution of the same functions, but taking into account the periodicity implied by the DFT. Note in (j) how data from adjacent periods produce wraparound error, yielding an incorrect convolution result. To obtain the correct result, function padding must be used.

An Example of Convolution



It causes the **wraparound error**

It can be solved by appending zeros

a	f
b	g
c	h
d	i
e	j

FIGURE 4.28 Left column: convolution of two discrete functions obtained using the approach discussed in Section 3.4.2. The result in (e) is correct. Right column: Convolution of the same functions, but taking into account the periodicity implied by the DFT. Note in (j) how data from adjacent periods produce wraparound error, yielding an incorrect convolution result. To obtain the correct result, function padding must be used.

Zero Padding

- Consider two functions f(x) and h(x) composed of A and B samples, respectively

- Append zeros to both functions so that they have the same length, denoted by P, then wraparound is avoided by choosing

$$P \geq A+B-1$$

Zero Padding

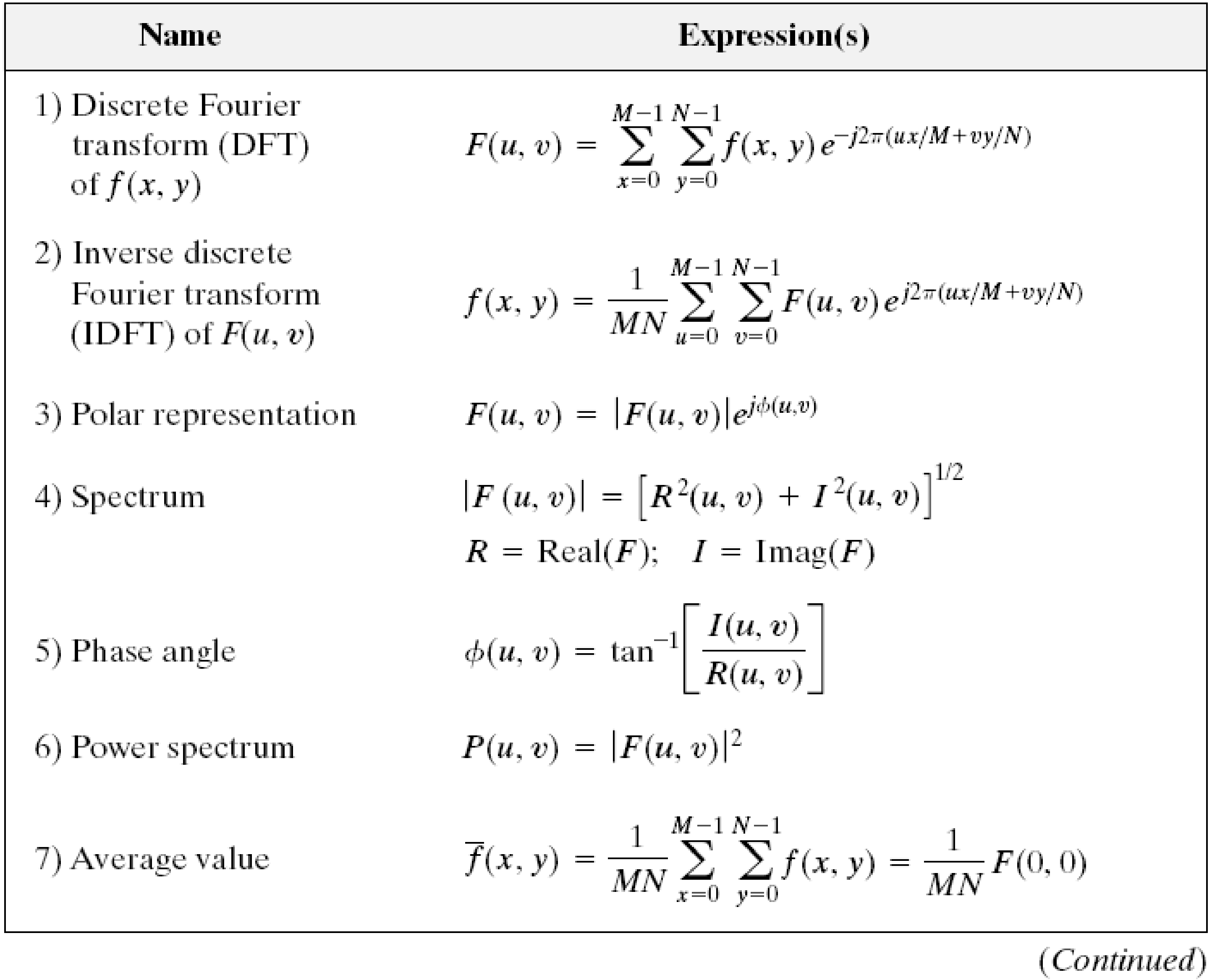
- Let f(x,y) and h(x,y) be two image arrays of sizes A×B and C×D pixels, respectively. Wraparound error in their convolution can be avoided by padding these functions with zeros

$$f_p(x,y) = \begin{cases} f(x,y) & 0 \le x \le A\text{-}1 \ and \ 0 \le y \le B\text{-}1 \\ 0 & A \le x \le P \ or \ B \le y \le Q \end{cases}$$

$$h_p(x,y) = \begin{cases} h(x,y) & 0 \le x \le C\text{-}1 \ and \ 0 \le y \le D\text{-}1 \\ 0 & C \le x \le P \ or \ D \le y \le Q \end{cases}$$

$$\text{Here } P \ge A + C - 1; Q \ge B + D - 1$$

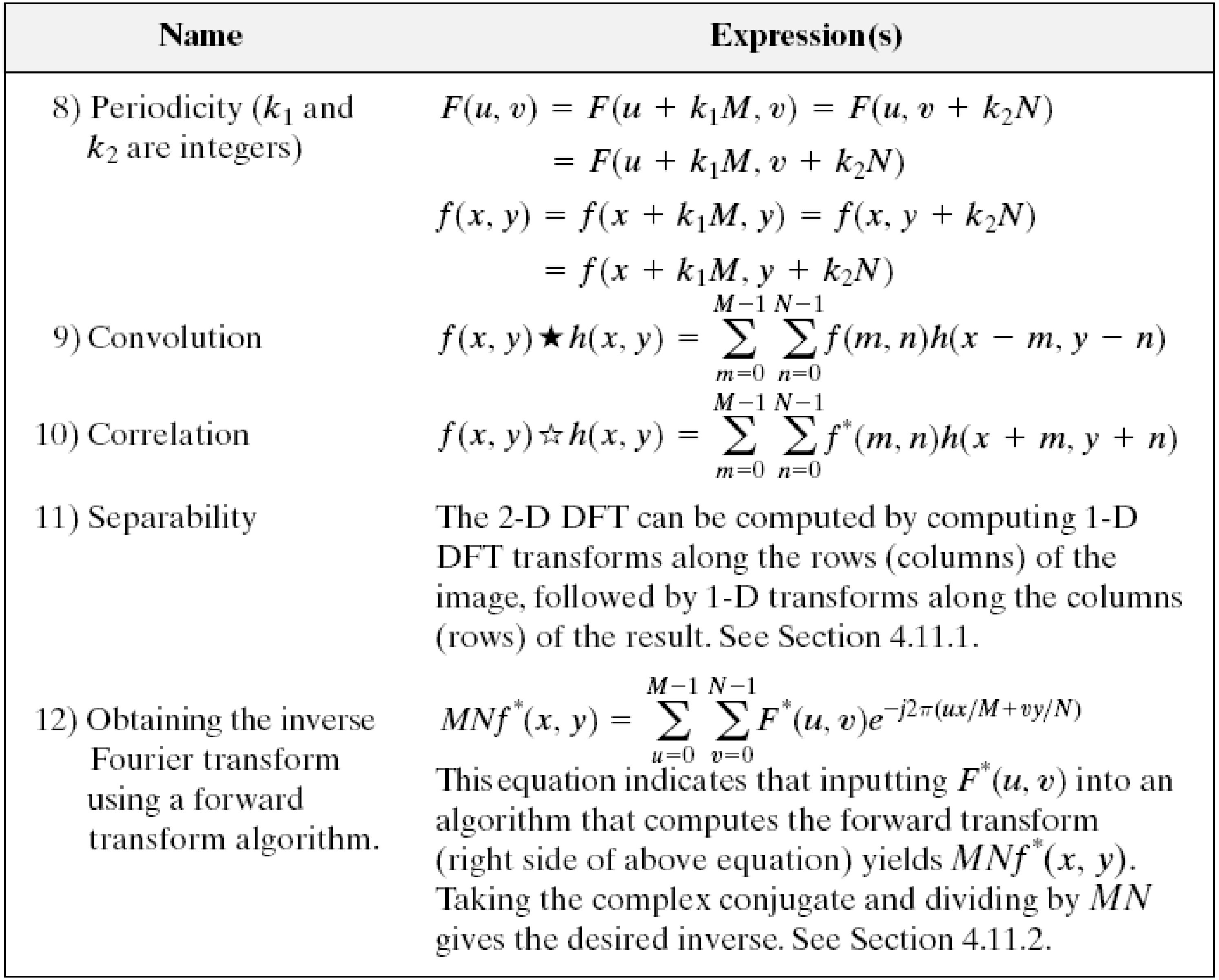
Summary



Name	Expression(s)
1) Discrete Fourier transform (DFT) of $f(x, y)$	$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$
2) Inverse discrete Fourier transform (IDFT) of $F(u, v)$	$f(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)}$
3) Polar representation	$F(u, v) = \lvert F(u, v) \rvert e^{j\phi(u,v)}$
4) Spectrum	$\lvert F(u, v) \rvert = \left[R^2(u, v) + I^2(u, v) \right]^{1/2}$ $R = \text{Real}(F); \quad I = \text{Imag}(F)$
5) Phase angle	$\phi(u, v) = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]$
6) Power spectrum	$P(u, v) = \lvert F(u, v) \rvert^2$
7) Average value	$\bar{f}(x, y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) = \frac{1}{MN} F(0, 0)$

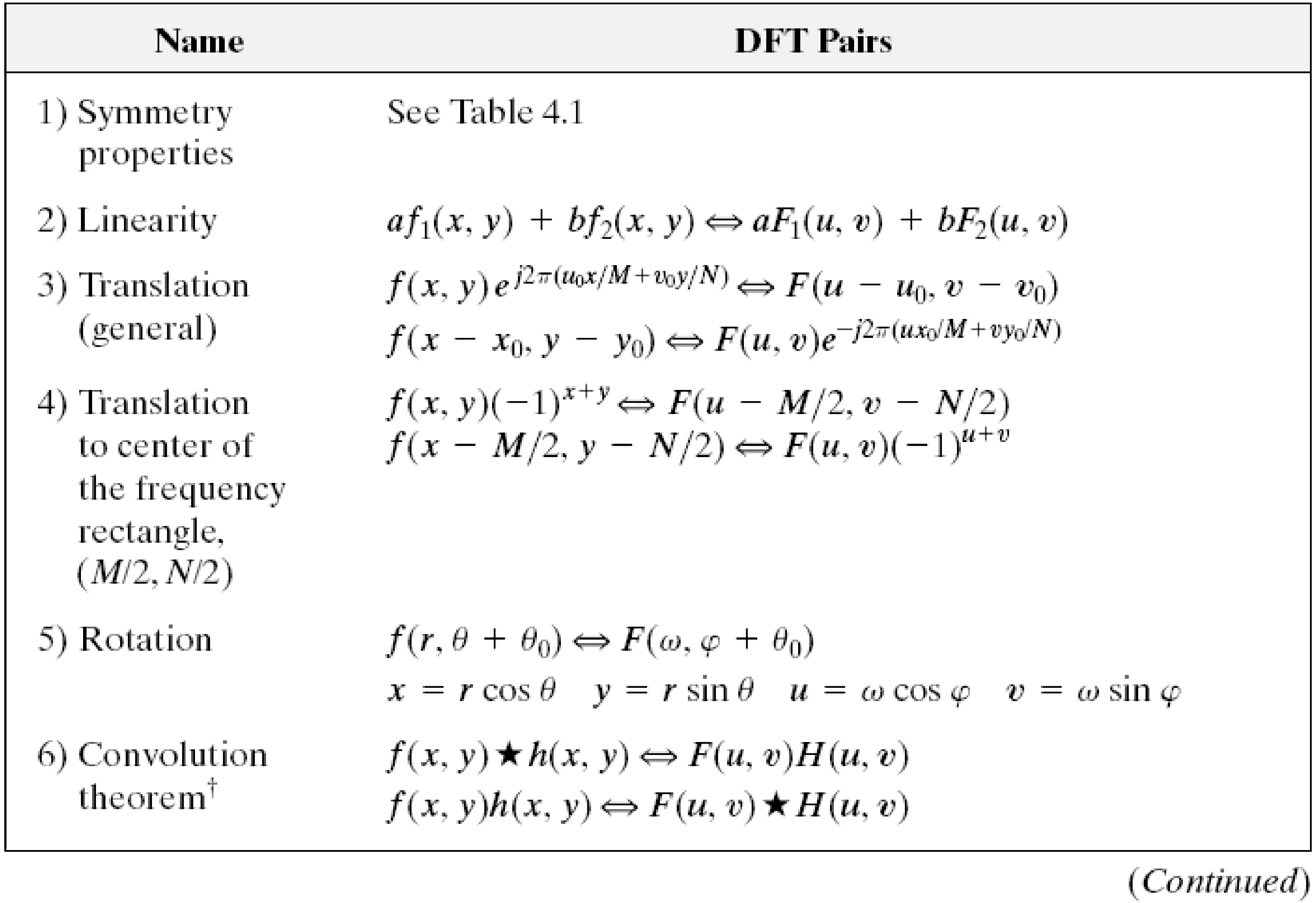
(Continued)

Summary



Name	Expression(s)
8) Periodicity (k_1 and k_2 are integers)	$F(u, v) = F(u + k_1M, v) = F(u, v + k_2N)$ $= F(u + k_1M, v + k_2N)$ $f(x, y) = f(x + k_1M, y) = f(x, y + k_2N)$ $= f(x + k_1M, y + k_2N)$
9) Convolution	$f(x, y) \bigstar h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)h(x - m, y - n)$
10) Correlation	$f(x, y) \star h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n)h(x + m, y + n)$
11) Separability	The 2-D DFT can be computed by computing 1-D DFT transforms along the rows (columns) of the image, followed by 1-D transforms along the columns (rows) of the result. See Section 4.11.1.
12) Obtaining the inverse Fourier transform using a forward transform algorithm.	$MNf^*(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v)e^{-j2\pi(ux/M + vy/N)}$ This equation indicates that inputting $F^*(u, v)$ into an algorithm that computes the forward transform (right side of above equation) yields $MNf^*(x, y)$. Taking the complex conjugate and dividing by MN gives the desired inverse. See Section 4.11.2.

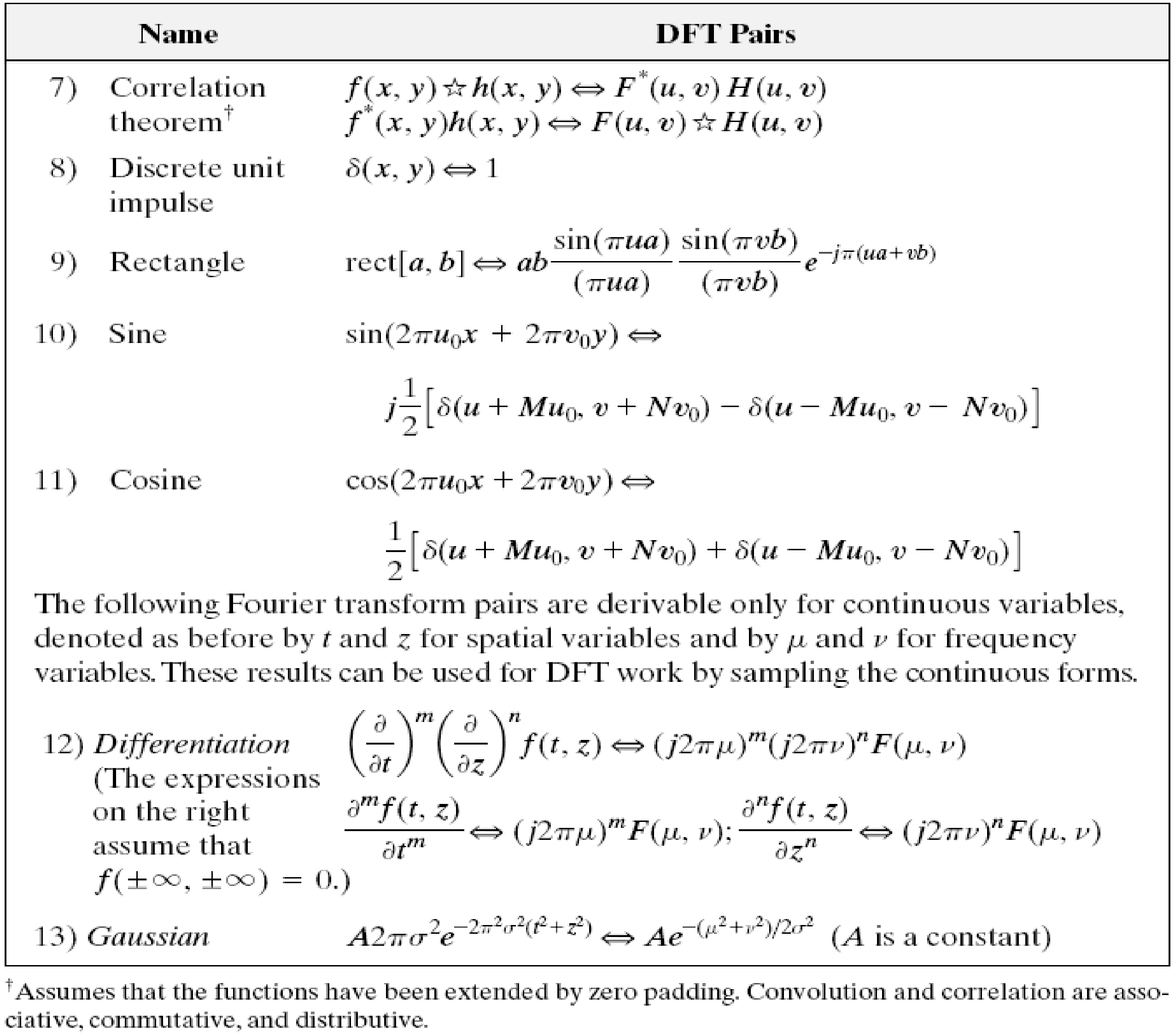
Summary



Name	DFT Pairs
1) Symmetry properties	See Table 4.1
2) Linearity	$af_1(x, y) + bf_2(x, y) \Leftrightarrow aF_1(u, v) + bF_2(u, v)$
3) Translation (general)	$f(x, y)e^{j2\pi(u_0x/M + v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v)e^{-j2\pi(ux_0/M + vy_0/N)}$
4) Translation to center of the frequency rectangle, $(M/2, N/2)$	$f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{u+v}$
5) Rotation	$f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$ $x = r\cos\theta \quad y = r\sin\theta \quad u = \omega\cos\varphi \quad v = \omega\sin\varphi$
6) Convolution theorem[†]	$f(x, y) \star h(x, y) \Leftrightarrow F(u, v)H(u, v)$ $f(x, y)h(x, y) \Leftrightarrow F(u, v) \star H(u, v)$

(Continued)

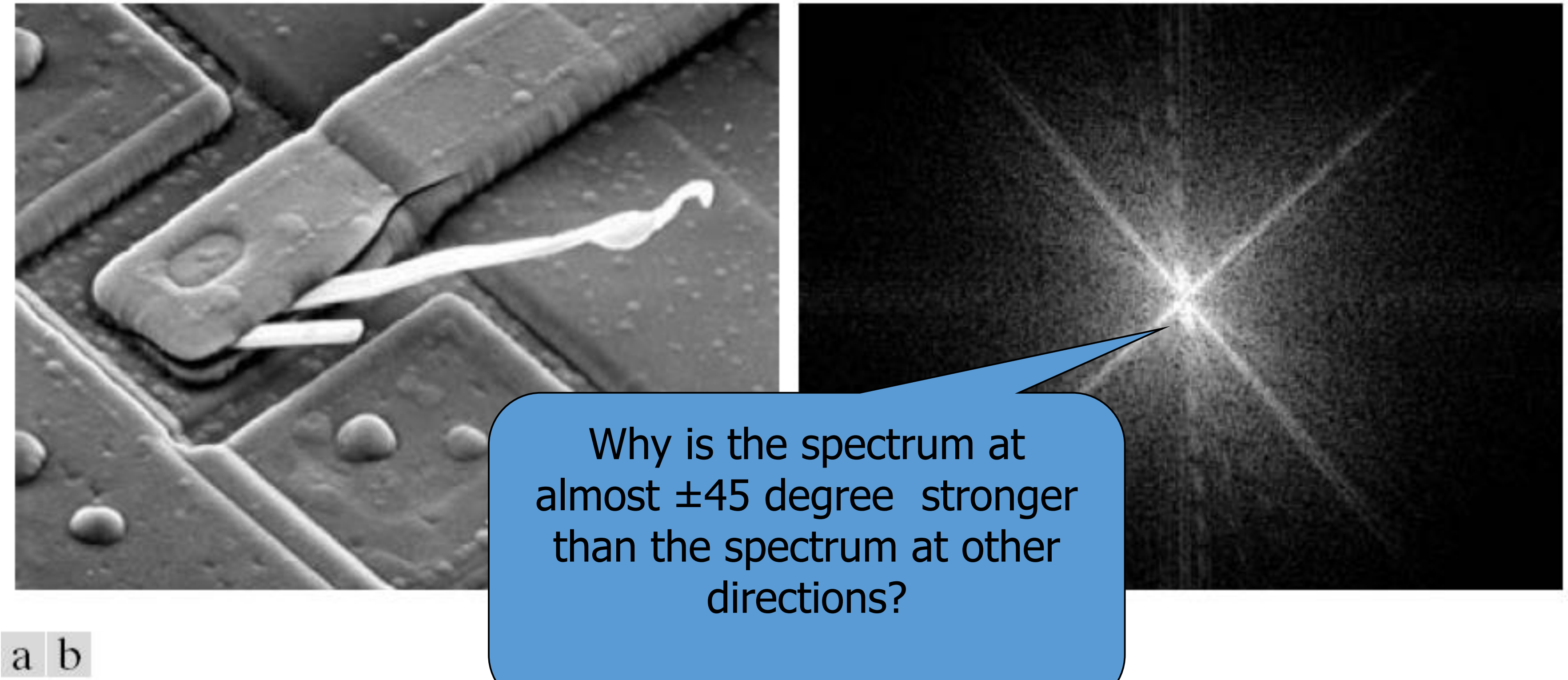
Summary



Name	DFT Pairs
7) Correlation theorem[†]	$f(x, y) \star h(x, y) \Leftrightarrow F^*(u, v)\, H(u, v)$ $f^*(x, y)h(x, y) \Leftrightarrow F(u, v) \star H(u, v)$
8) Discrete unit impulse	$\delta(x, y) \Leftrightarrow 1$
9) Rectangle	$\text{rect}[a, b] \Leftrightarrow ab\frac{\sin(\pi ua)}{(\pi ua)}\frac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
10) Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $j\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) - \delta(u - Mu_0, v - Nv_0)\right]$
11) Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) + \delta(u - Mu_0, v - Nv_0)\right]$
The following Fourier transform pairs are derivable only for continuous variables, denoted as before by t and z for spatial variables and by μ and ν for frequency variables. These results can be used for DFT work by sampling the continuous forms.	
12) *Differentiation* (The expressions on the right assume that $f(\pm\infty, \pm\infty) = 0$.)	$\left(\frac{\partial}{\partial t}\right)^m\left(\frac{\partial}{\partial z}\right)^n f(t, z) \Leftrightarrow (j2\pi\mu)^m(j2\pi\nu)^n F(\mu, \nu)$ $\frac{\partial^m f(t, z)}{\partial t^m} \Leftrightarrow (j2\pi\mu)^m F(\mu, \nu); \frac{\partial^n f(t, z)}{\partial z^n} \Leftrightarrow (j2\pi\nu)^n F(\mu, \nu)$
13) *Gaussian*	$A2\pi\sigma^2 e^{-2\pi^2\sigma^2(t^2+z^2)} \Leftrightarrow Ae^{-(\mu^2+\nu^2)/2\sigma^2}$ (A is a constant)

[†] Assumes that the functions have been extended by zero padding. Convolution and correlation are associative, commutative, and distributive.

The Basic Filtering in the Frequency Domain



Why is the spectrum at almost ±45 degree stronger than the spectrum at other directions?

FIGURE 4.29 (a) SEM image of a damaged integrated circuit. (b) Fourier spectrum of (a). (Original image courtesy of Dr. J. M. Hudak, Brockhouse Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada.)

The Basic Filtering in the Frequency Domain

- Modifying the Fourier transform of an image
- Computing the inverse transform to obtain the processed result

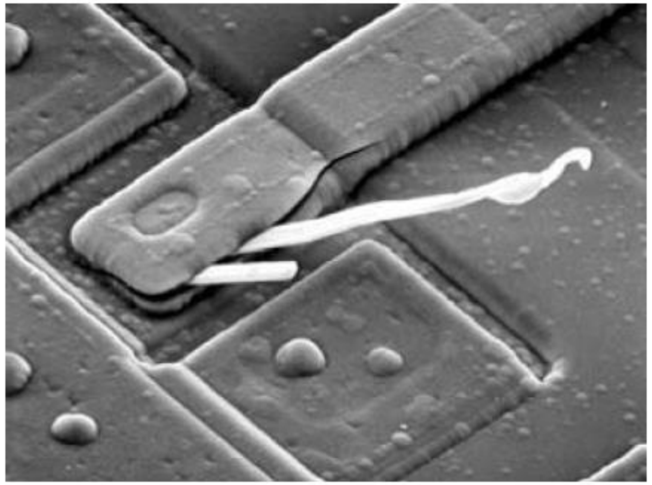
$$g(x, y) = \Im^{-1}\{H(u,v)F(u,v)\}$$

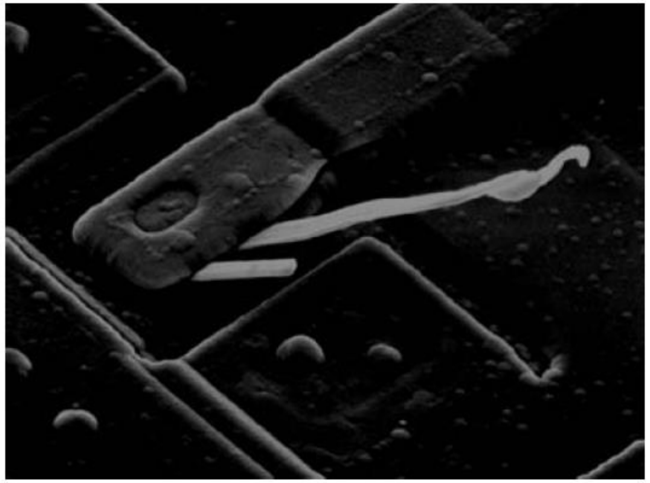
$F(u,v)$ is the DFT of the input image

$H(u,v)$ is a filter function.

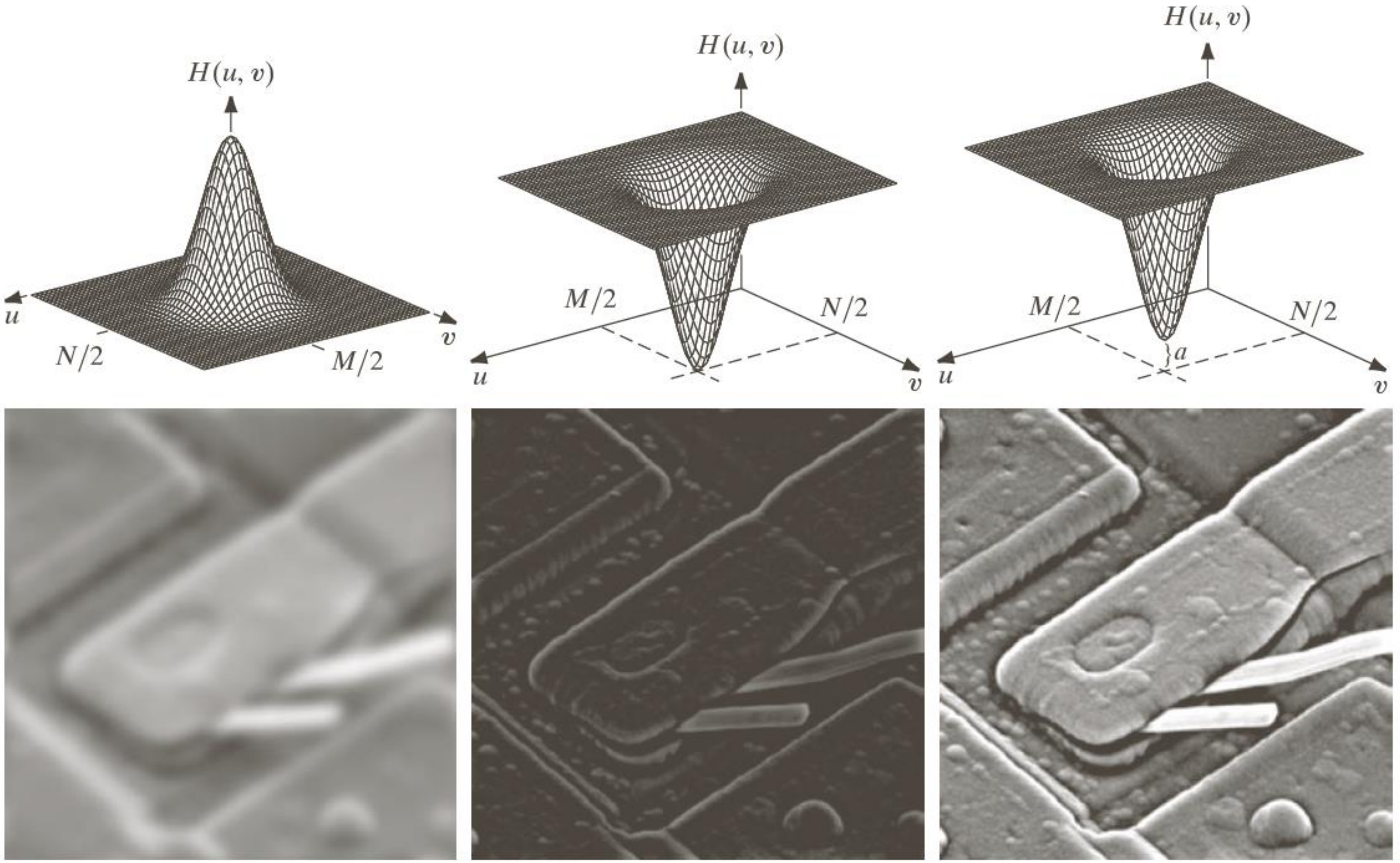
The Basic Filtering in the Frequency Domain

- In a filter H(u,v) that is 0 at the center of the transform and 1 elsewhere, what's the output image?





The Basic Filtering in the Frequency Domain



a	b	c
d	e	f

FIGURE 4.31 Top row: frequency domain filters. Bottom row: corresponding filtered images obtained using Eq. (4.7-1). We used $a = 0.85$ in (c) to obtain (f) (the height of the filter itself is 1). Compare (f) with Fig. 4.29(a).

The Basic Filtering in the Frequency Domain

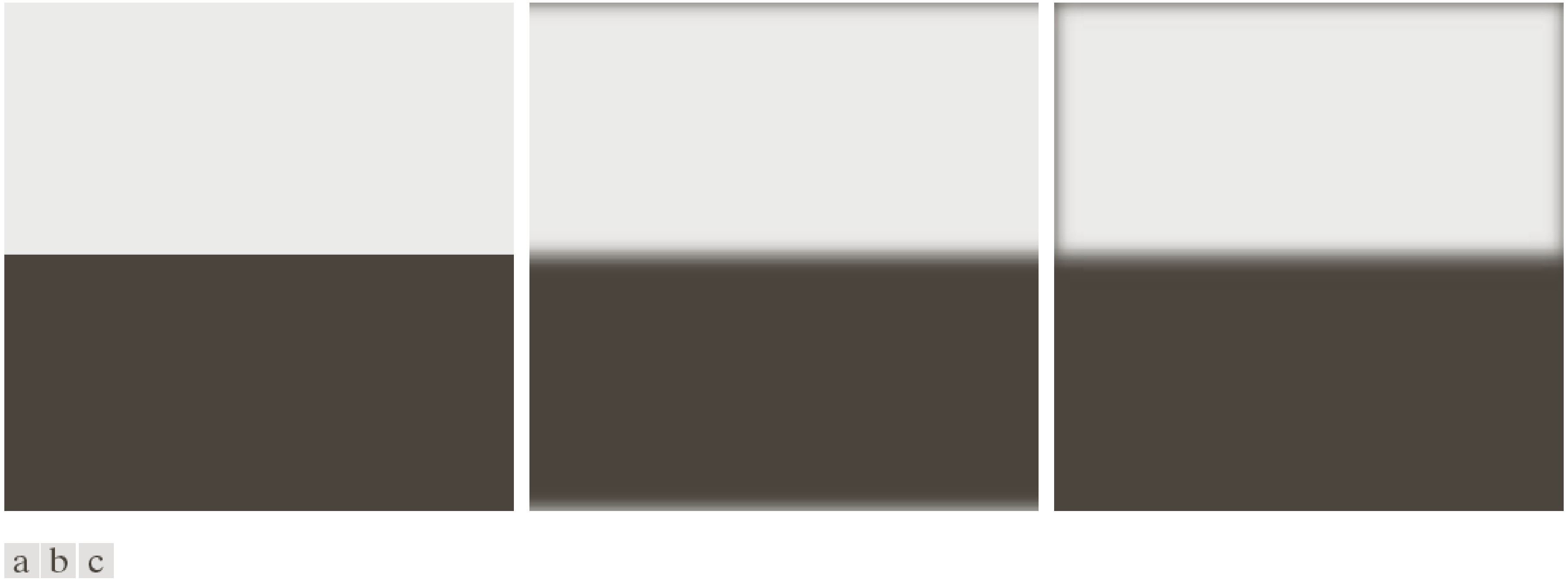


FIGURE 4.32 (a) A simple image. (b) Result of blurring with a Gaussian lowpass filter without padding. (c) Result of lowpass filtering with padding. Compare the light area of the vertical edges in (b) and (c).

Zero-Phase-Shift Filters

$$g(x,y) = \Im^{-1}\{H(u,v)F(u,v)\}$$

$$F(u,v) = R(u,v) + jI(u,v)$$

$$g(x,y) = \Im^{-1}\left[H(u,v)R(u,v) + jH(u,v)I(u,v)\right]$$

Filters affect the real and imaginary parts equally,

and thus no effect on the phase.

These filters are called **zero-phase-shift** filters

Examples: Nonzero-Phase-Shift Filters

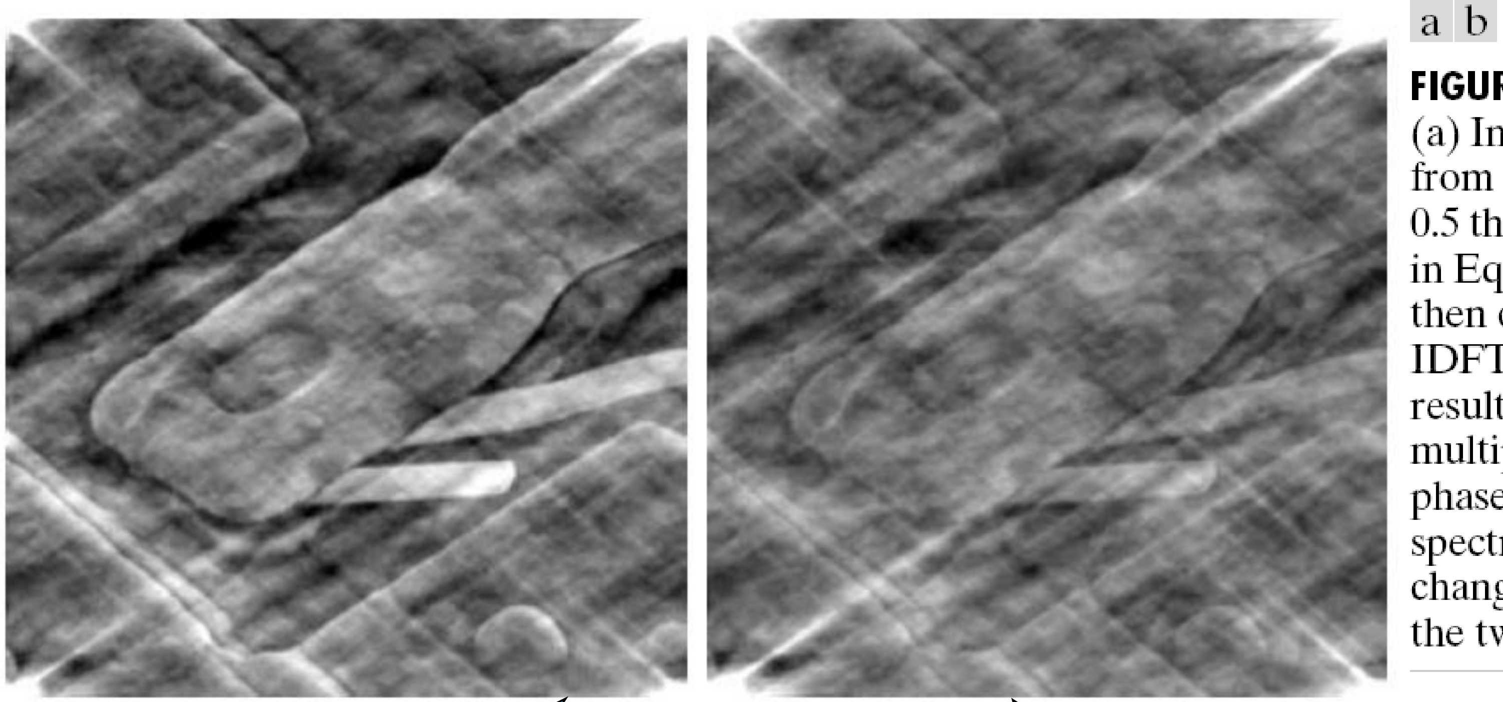


FIGURE 4.35 (a) Image resulting from multiplying by 0.5 the phase angle in Eq. (4.6-15) and then computing the IDFT. (b) The result of multiplying the phase by 0.25. The spectrum was not changed in either of the two cases.

Even s s in the phase ave drama ndesirable) ef filtered outpu

Phase angle is multiplied by 0.5

Phase angle is multiplied by 0.5

Summary:
Steps for Filtering in the Frequency Domain

1. Given an input image f(x,y) of size MxN, obtain the padding parameters P and Q. Typically, P = 2M and Q = 2N.

2. Form a padded image, $f_p(x,y)$ of size PxQ by appending the necessary number of zeros to f(x,y)

3. Multiply $f_p(x,y)$ by $(-1)^{x+y}$ to center its transform

4. Compute the DFT, F(u,v) of the image from step 3

5. Generate a real, symmetric filter function*, H(u,v), of size PxQ with center at coordinates (P/2, Q/2)

*generate from a given spatial filter, we pad the spatial filter, multiply the expadded array by $(-1)^{x+y}$, and compute the DFT of the result to obtain a centered H(u,v).

Summary:
Steps for Filtering in the Frequency Domain

6. Form the product G(u,v) = H(u,v)F(u,v) using array multiplication

7. Obtain the processed image

$$g_p(x,y) = \left\{ real\left[\Im^{-1}\left[G(u,v) \right] \right] \right\}(-1)^{x+y}$$

8. Obtain the final processed result, g(x,y), by extracting the MxN region from the top, left quadrant of $g_p(x,y)$

An
Ste

cessing

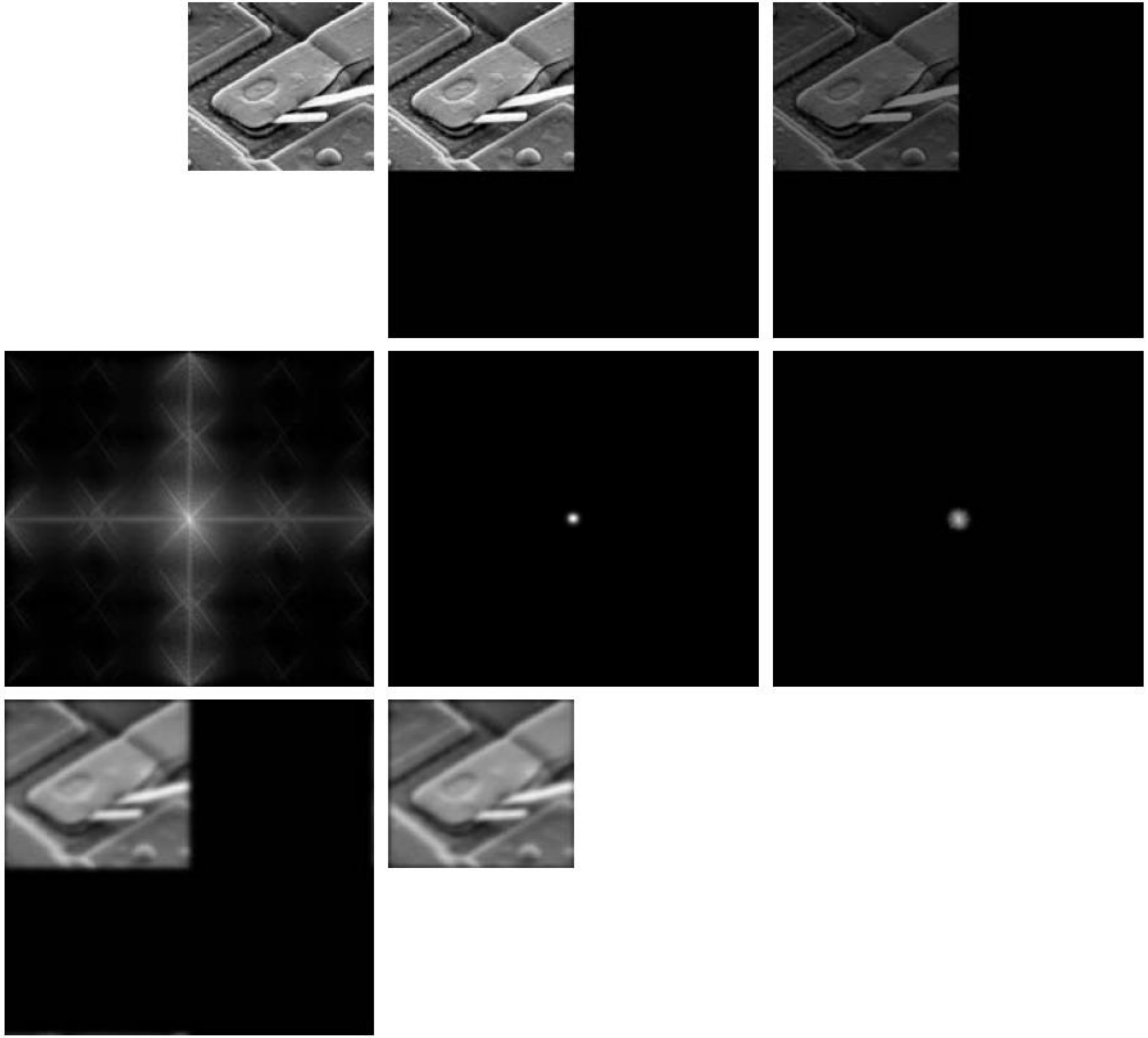


FIGURE 4.36 (a) An $M \times N$ image, f. (b) Padded image, f_p of size $P \times Q$. (c) Result of multiplying f_p by $(-1)^{x+y}$. (d) Spectrum of F_p. (e) Centered Gaussian lowpass filter, H, of size $P \times Q$. (f) Spectrum of the product HF_p. (g) g_p, the product of $(-1)^{x+y}$ and the real part of the IDFT of HF_p. (h) Final result, g, obtained by cropping the first M rows and N columns of g_p.

Correspondence Between Filtering in the Spatial and Frequency Domains (1)

Let H(u) denote the 1-D frequency domain Gaussian filter

$$H(u) = Ae^{-u^2/2\sigma^2}$$

The corresponding filter in the spatial domain

$$h(x) = \sqrt{2\pi}\sigma Ae^{-2\pi^2\sigma^2x^2}$$

1. Both components are Gaussian and real
2. The functions behave reciprocally

Correspondence Between Filtering in the Spatial and Frequency Domains (2)

Let $H(u)$ denote the difference of Gaussian filter

$$H(u) = Ae^{-u^2/2\sigma_1^2} - Be^{-u^2/2\sigma_2^2}$$

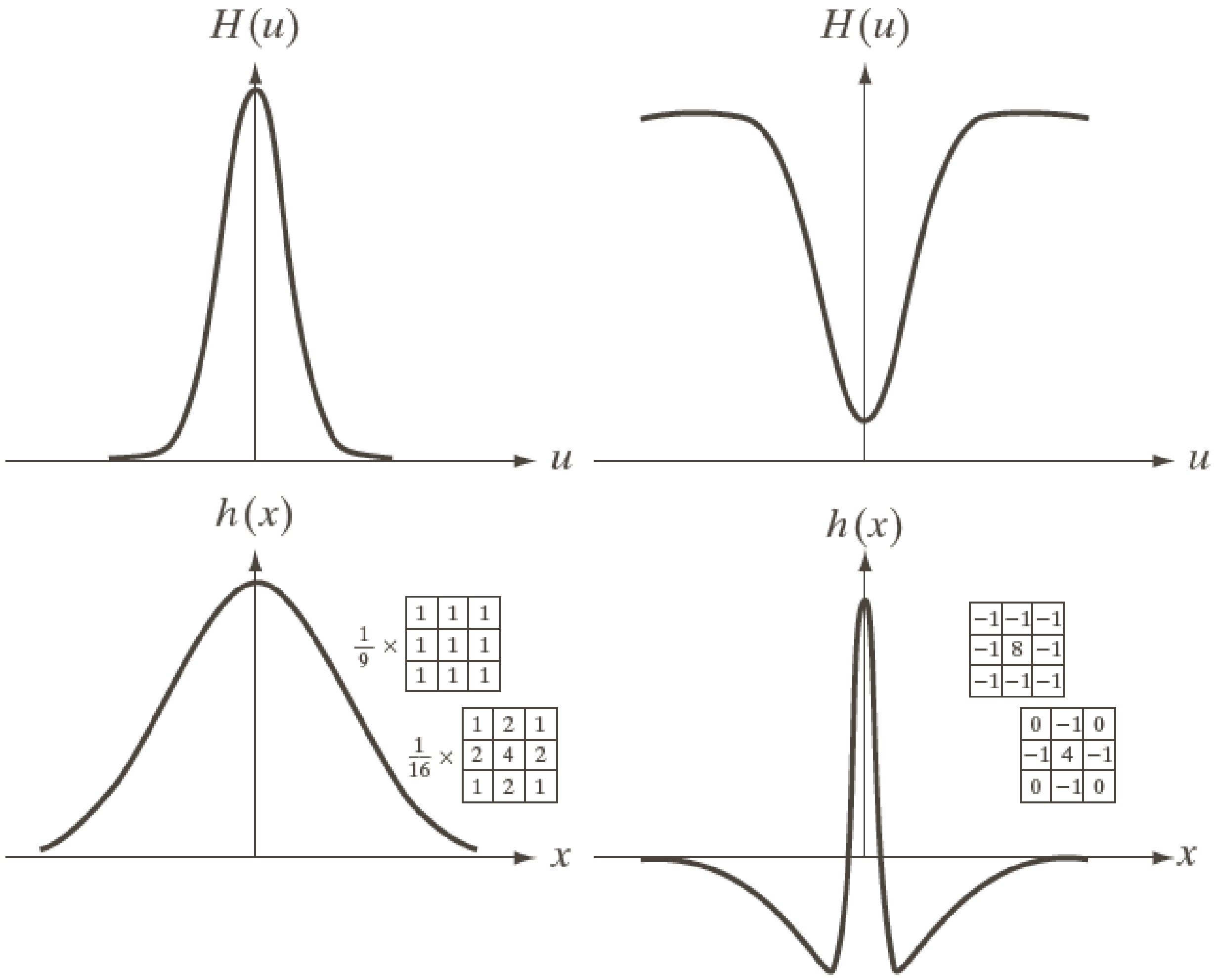
$$\text{with } A \geq B \text{ and } \sigma_1 \geq \sigma_2$$

The corresponding filter in the spatial domain

$$h(x) = \sqrt{2\pi}\sigma_1 Ae^{-2\pi^2\sigma_1^2x^2} - \sqrt{2\pi}\sigma_2 Ae^{-2\pi^2\sigma_2^2x^2}$$

High-pass filter or low-pass filter ?

Correspondence Between Filtering in the Spatial and Frequency Domains (3)



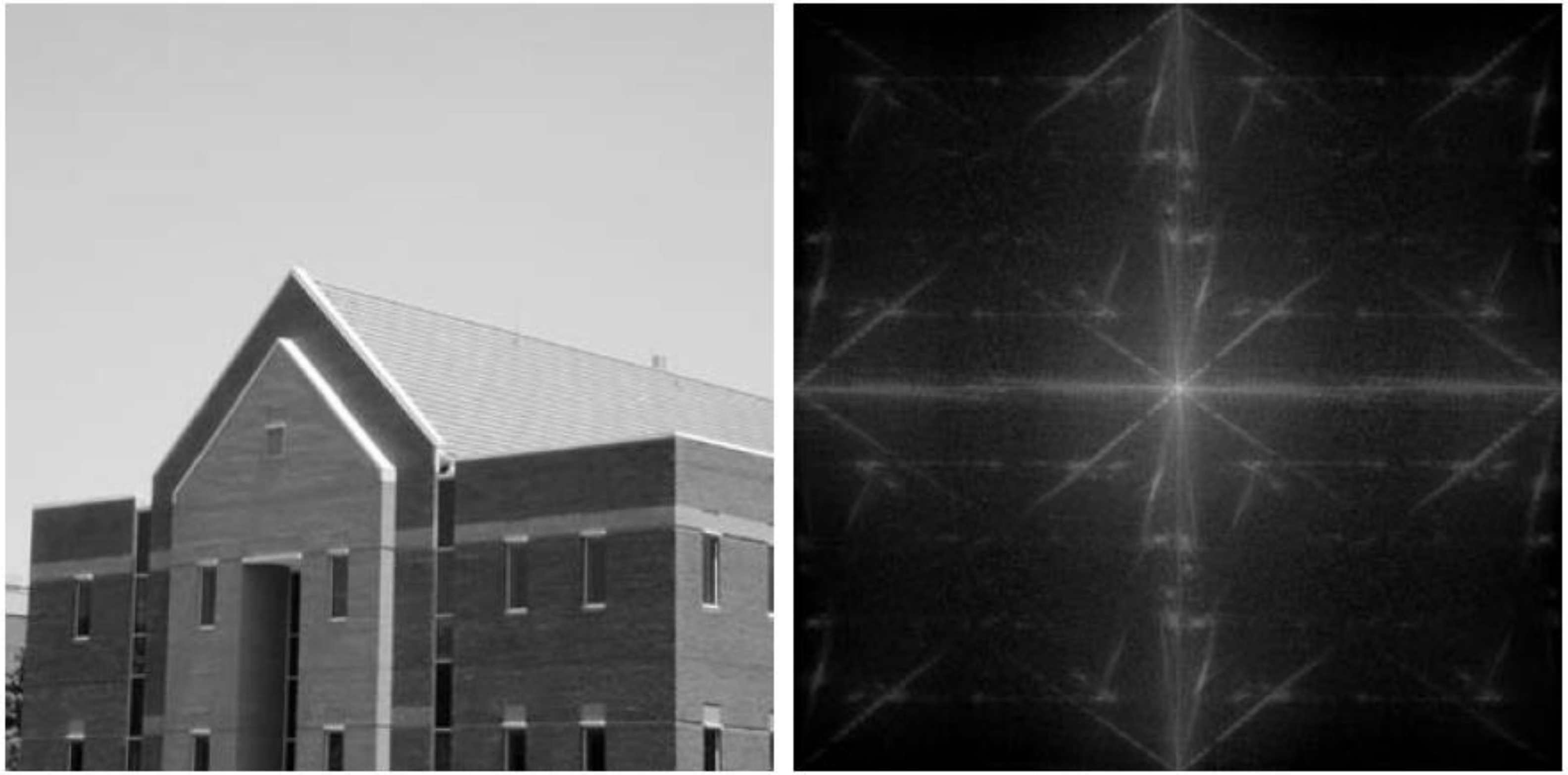
$$\frac{1}{9}\times\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}\qquad \frac{1}{16}\times\begin{bmatrix}1&2&1\\2&4&2\\1&2&1\end{bmatrix}$$

$$\begin{bmatrix}-1&-1&-1\\-1&8&-1\\-1&-1&-1\end{bmatrix}\qquad \begin{bmatrix}0&-1&0\\-1&4&-1\\0&-1&0\end{bmatrix}$$

a	c
b	d

FIGURE 4.37 (a) A 1-D Gaussian lowpass filter in the frequency domain. (b) Spatial lowpass filter corresponding to (a). (c) Gaussian highpass filter in the frequency domain. (d) Spatial highpass filter corresponding to (c). The small 2-D masks shown are spatial filters we used in Chapter 3.

Correspondence Between Filtering in the Spatial and Frequency Domains: Example

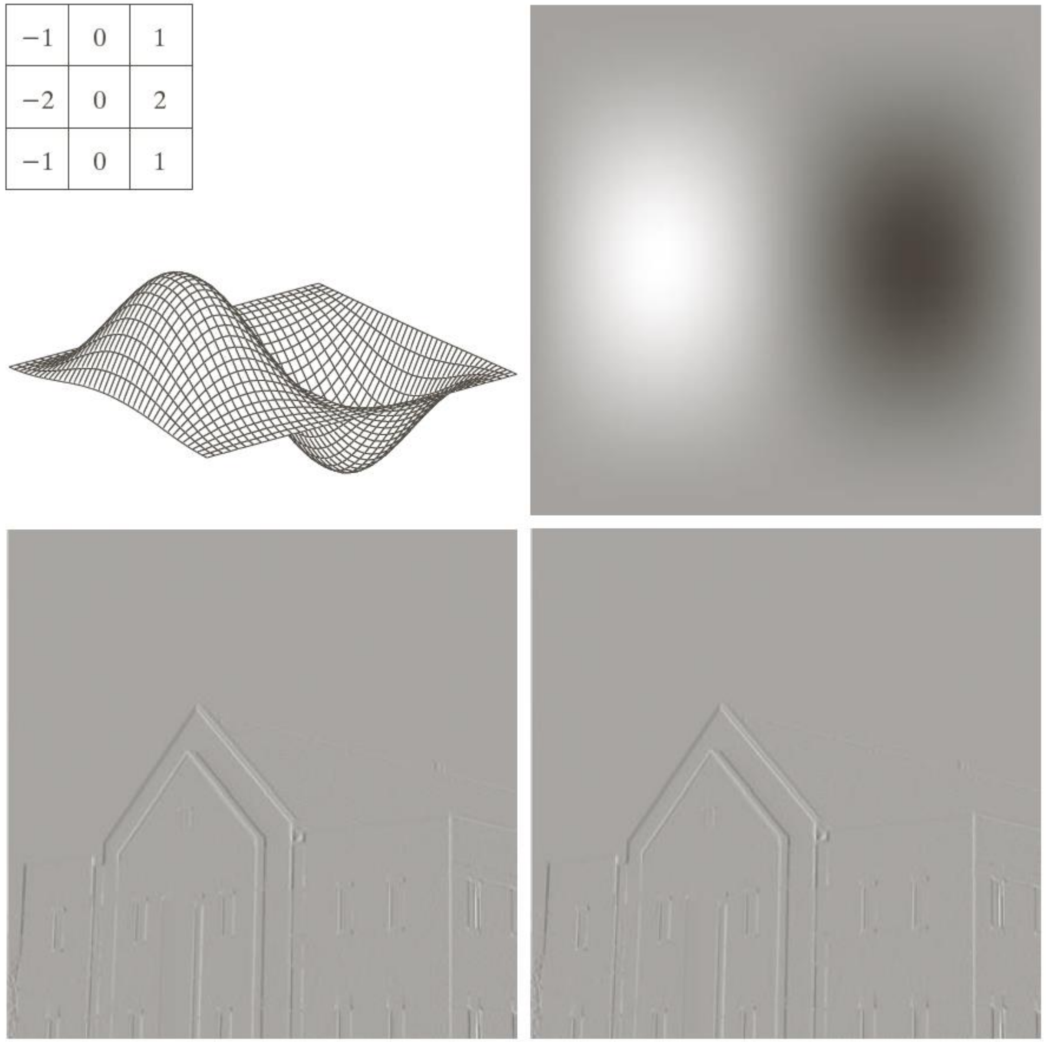


a b

FIGURE 4.38 (a) Image of a building, and (b) its spectrum.

600x600

Correspondence Between Filtering in the Spatial and Frequency Domains: Example



−1	0	1
−2	0	2
−1	0	1

FIGURE 4.39 (a) A spatial mask and perspective plot of its corresponding frequency domain filter. (b) Filter shown as an image. (c) Result of filtering Fig. 4.38(a) in the frequency domain with the filter in (b). (d) Result of filtering the same image with the spatial filter in (a). The results are identical.

Generate H(u,v)

$$f_p(x,y)=\begin{cases} f(x,y) & 0\le x\le 599 \ and \ 0\le y\le 599 \\ 0 & 600\le x\le 602 \ or \ 600\le y\le 602 \end{cases}$$

$$h_p(x,y)=\begin{cases} h(x,y) & 0\le x\le 2 \ and \ 0\le y\le 2 \\ 0 & 3\le x\le 602 \ or \ 3\le y\le 602 \end{cases}$$

$$\text{Here } P\ge A(600)+C(3)-1=602;$$

$$Q\ge B(600)+D(3)-1=602.$$

Generate H(u,v)

1. Multiply $h_p(x, y)$ by $(\text{-}1)^{x+y}$ to center the frequency domain filter

2. Compute the forward DFT of the result in (1)

3. Set the real part of the resulting DFT to 0 to account for parasitic real parts

4. Multiply the result by $(\text{-}1)^{u+v}$, which is implicit when $h(x, y)$ was moved to the center of $h_p(x, y)$.

Image Smoothing Using Filter Domain Filters: ILPF

Ideal Lowpass Filters (ILPF)

$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

D_0 is a positive constant and $D(u,v)$ is the distance between a point (u,v) in the frequency domain and the center of the frequency rectangle

$$D(u,v) = \left[(u - P/2)^2 + (v - Q/2)^2 \right]^{1/2}$$

Image Smoothing Using Filter Domain Filters: ILPF

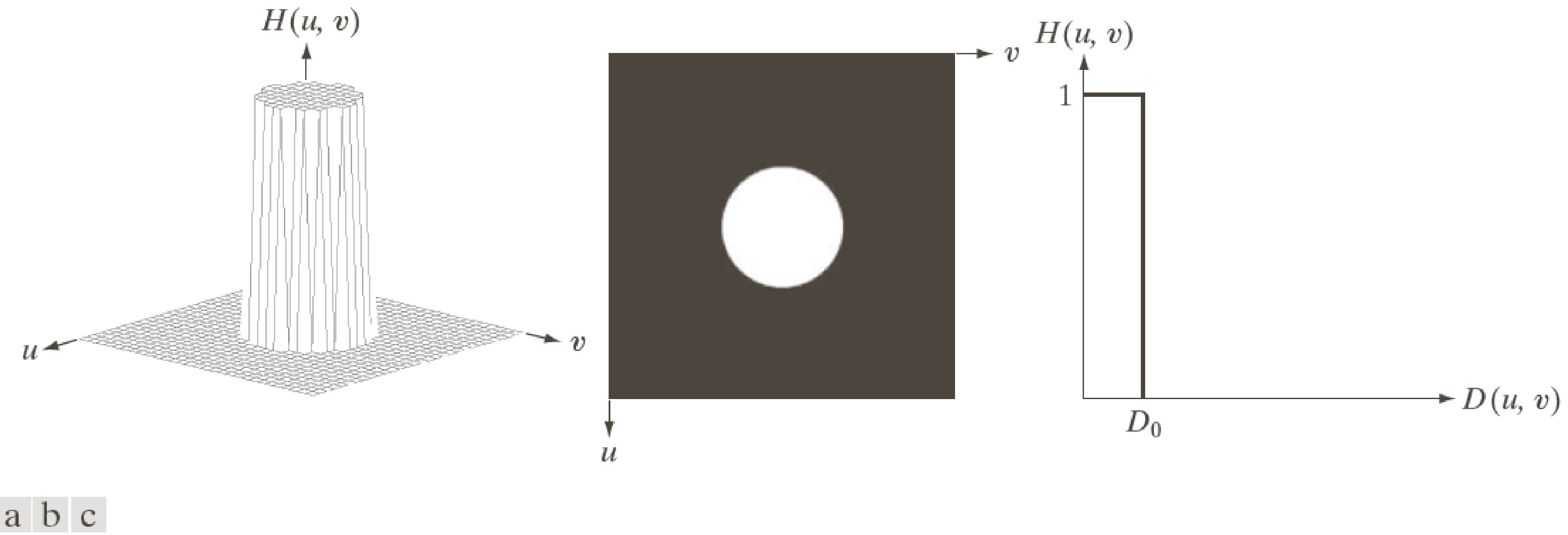


FIGURE 4.40 (a) Perspective plot of an ideal lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

ILPF Filtering Example

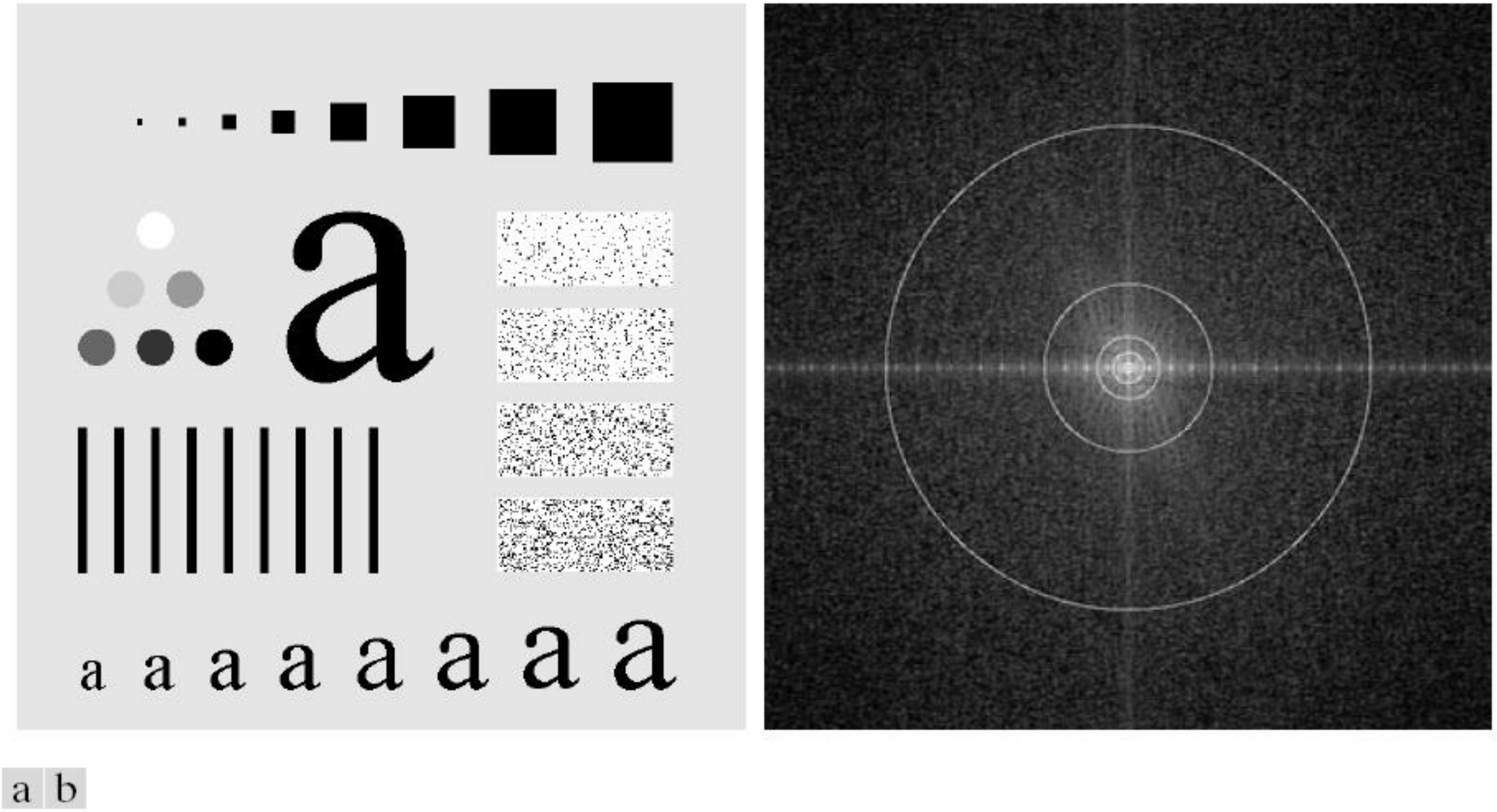


FIGURE 4.41 (a) Test pattern of size 688 × 688 pixels, and (b) its Fourier spectrum. The spectrum is double the image size due to padding but is shown in half size so that it fits in the page. The superimposed circles have radii equal to 10, 30, 60, 160, and 460 with respect to the full-size spectrum image. These radii enclose 87.0, 93.1, 95.7, 97.8, and 99.2% of the padded image power, respectively.

ILPF Filtering Example

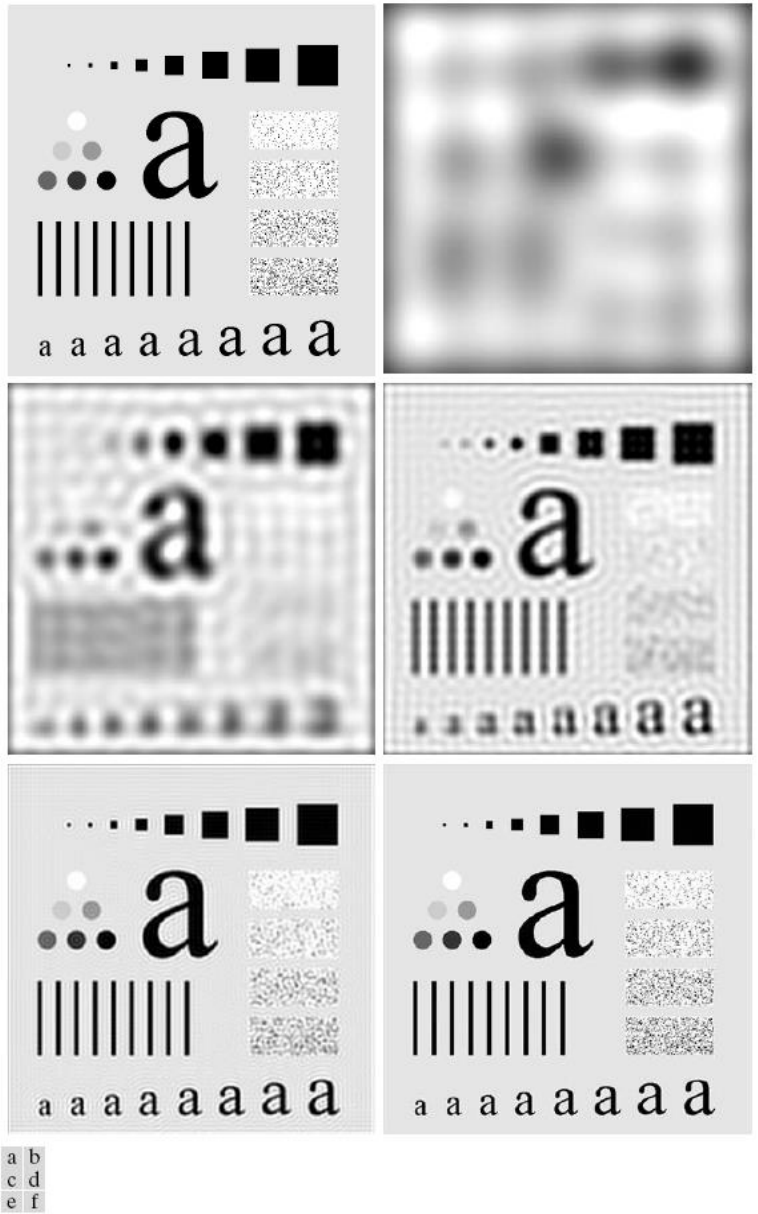
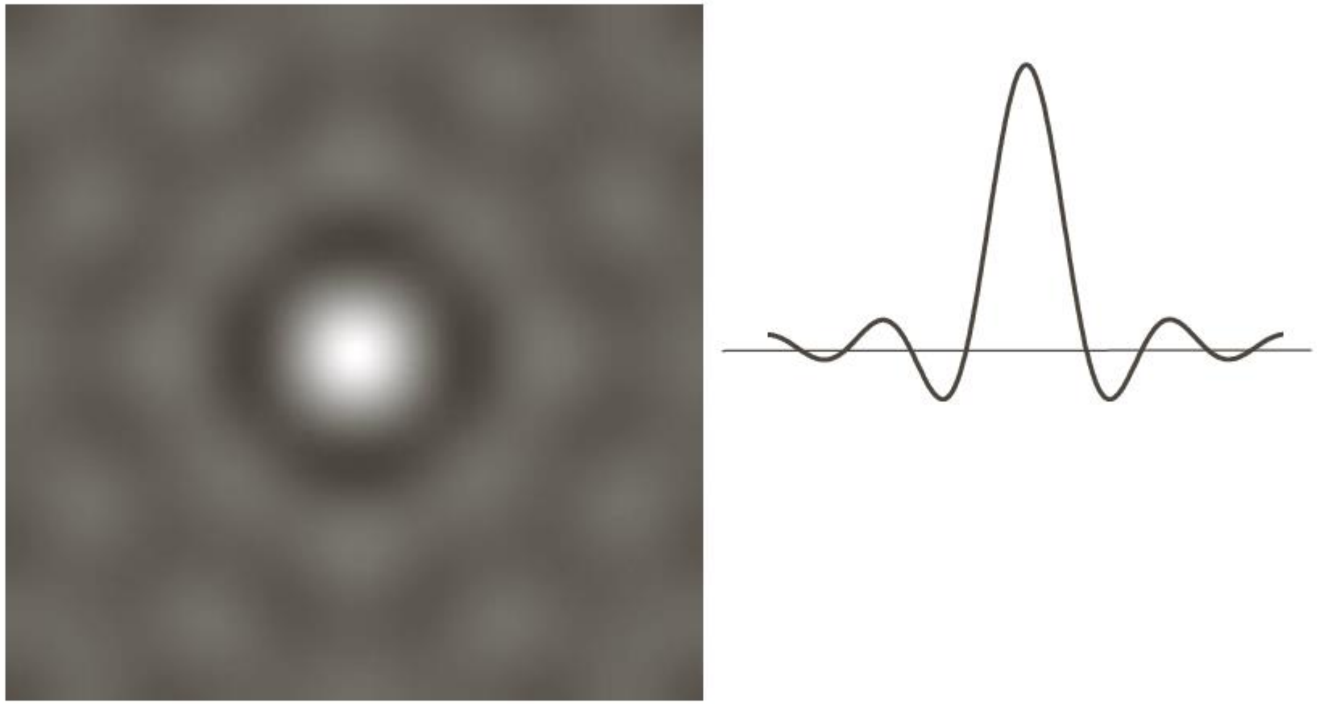


FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

The Spatial Representation of ILPF



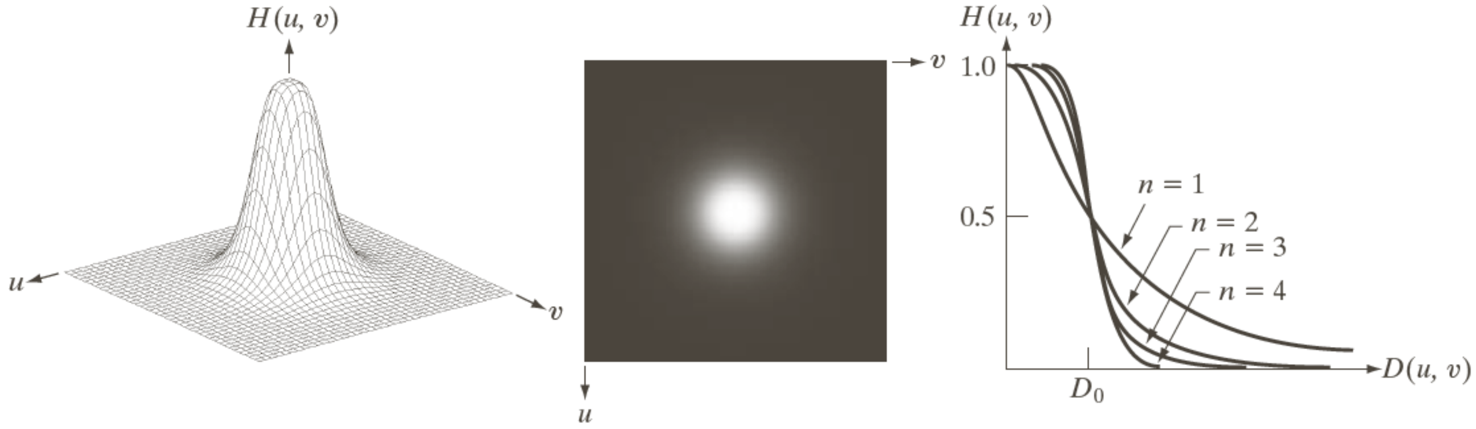
a b

FIGURE 4.43 (a) Representation in the spatial domain of an ILPF of radius 5 and size 1000 × 1000. (b) Intensity profile of a horizontal line passing through the center of the image.

Image Smoothing Using Filter Domain Filters: BLPF

Butterworth Lowpass Filters (BLPF) of order n and with cutoff frequency D_0

$$H(u,v)=\frac{1}{1+\left[D(u,v)/D_0\right]^{2n}}$$



a b c

FIGURE 4.44 (a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

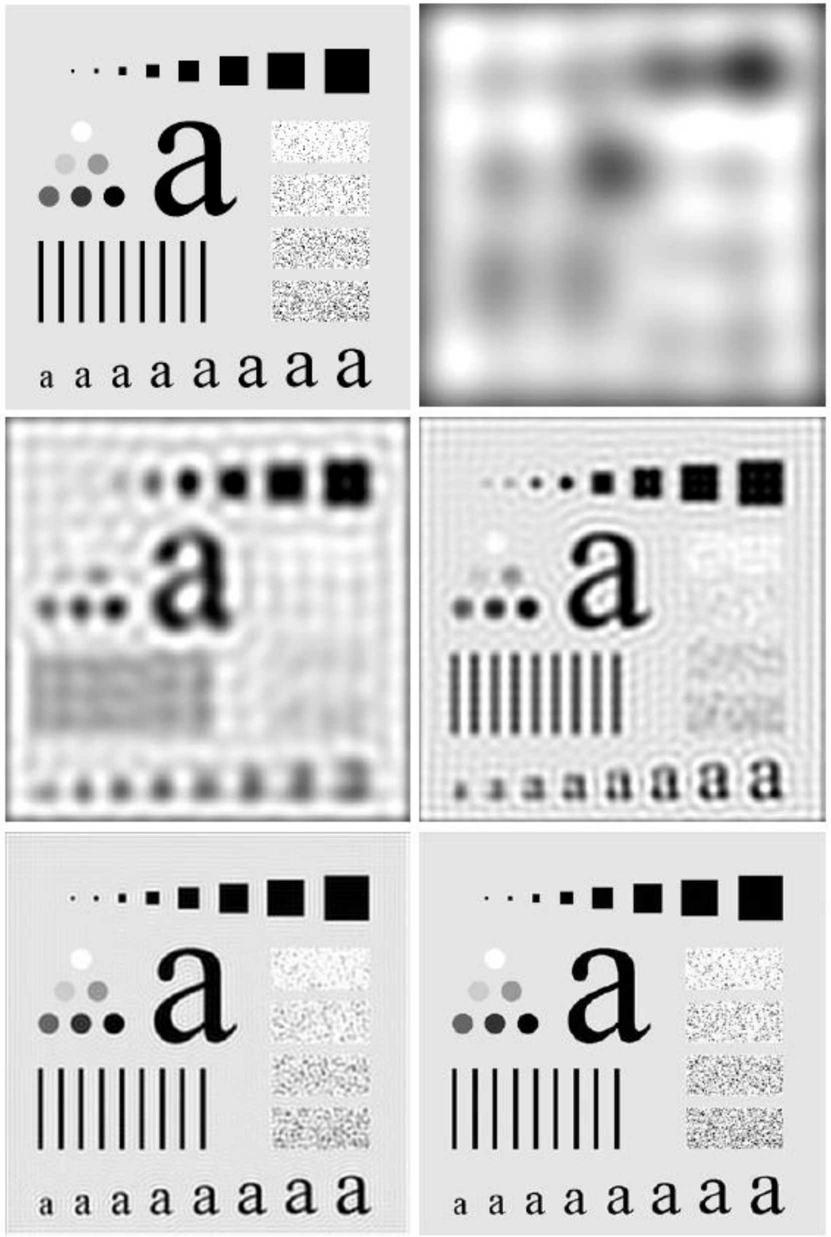


FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

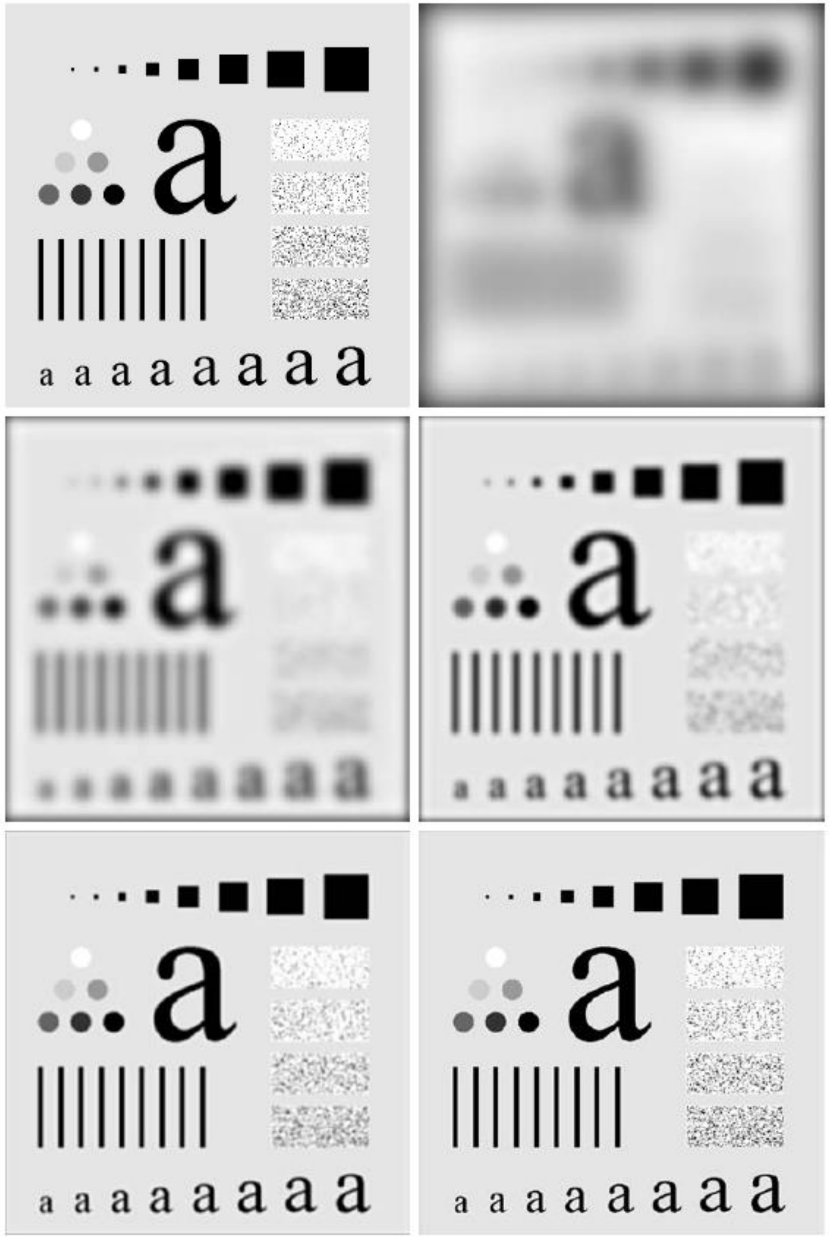


FIGURE 4.45 (a) Original image. (b)–(f) Results of filtering using BLPFs of order 2, with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Fig. 4.42.

The Spatial Representation of BLPF

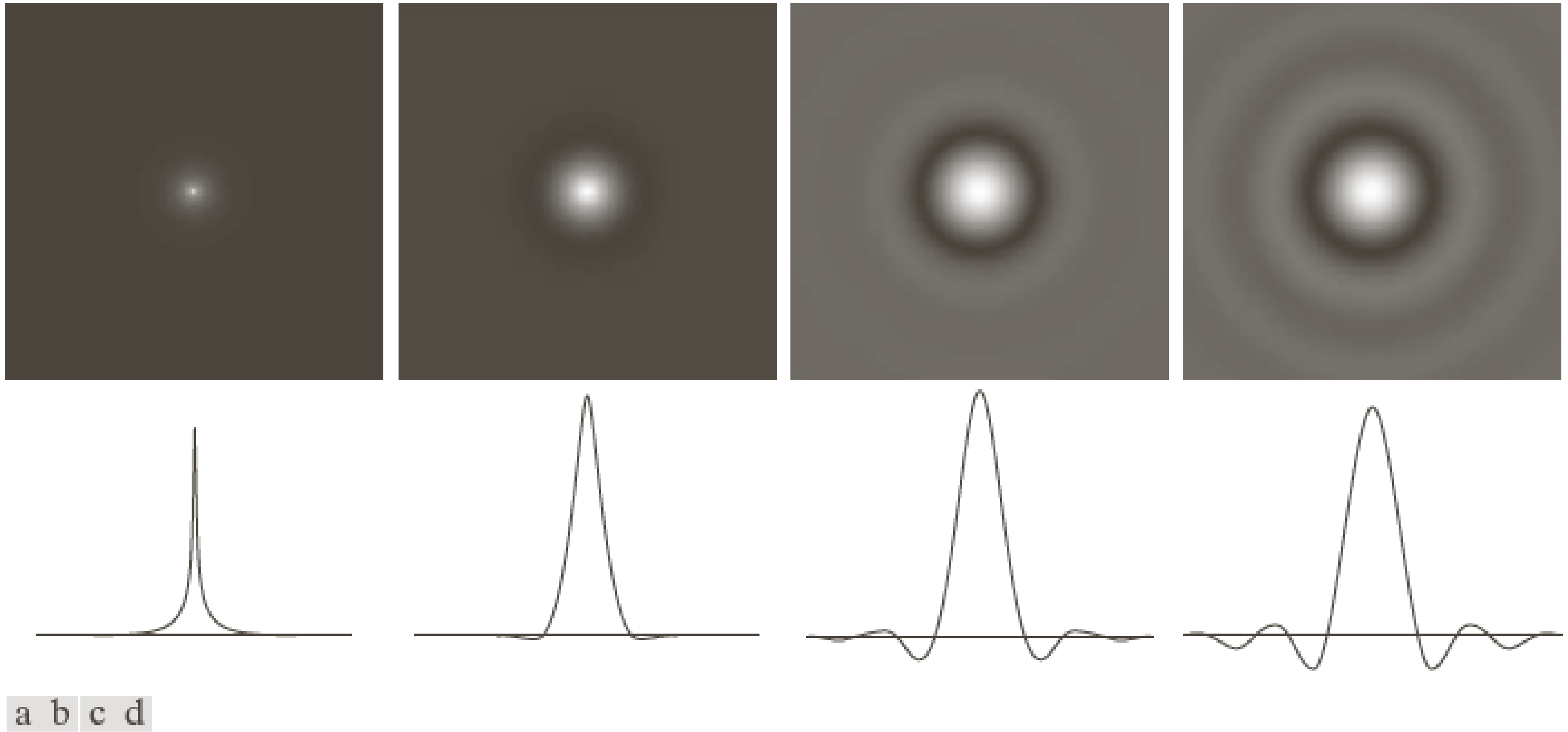


FIGURE 4.46 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is 1000 × 1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

Image Smoothing Using Filter Domain Filters: GLPF

Gaussian Lowpass Filters (GLPF) in two dimensions is given

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2}$$

By letting $\sigma = D_0$

$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

Image Smoothing Using Filter Domain Filters: GLPF

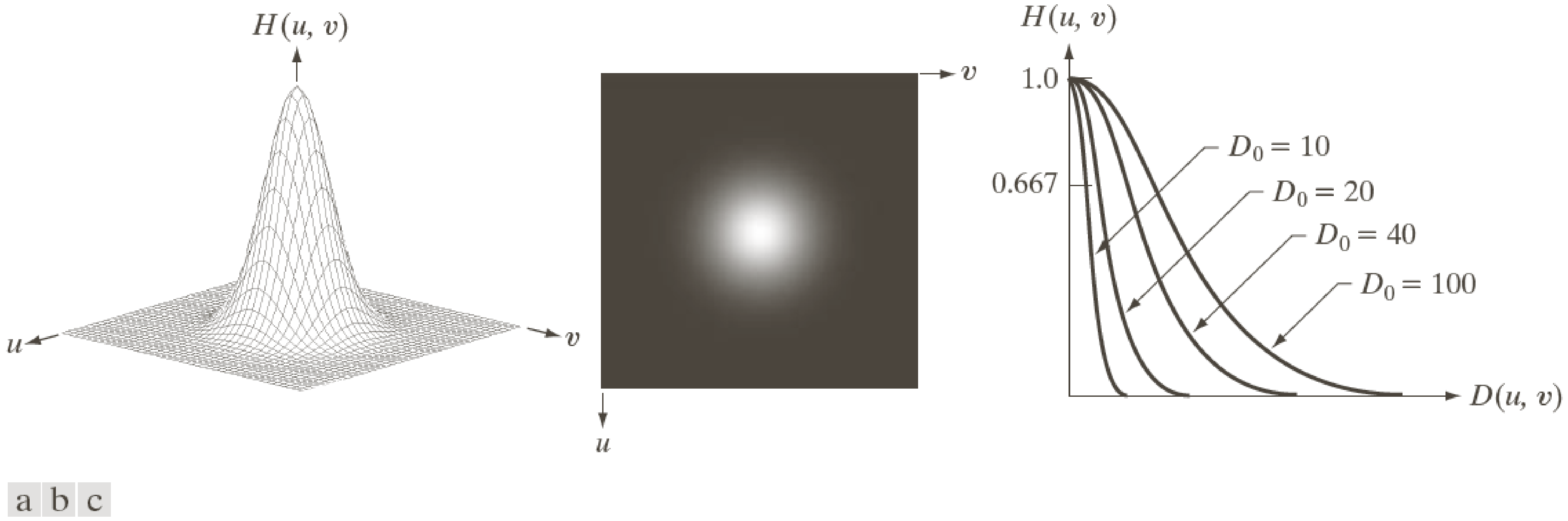


FIGURE 4.47 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

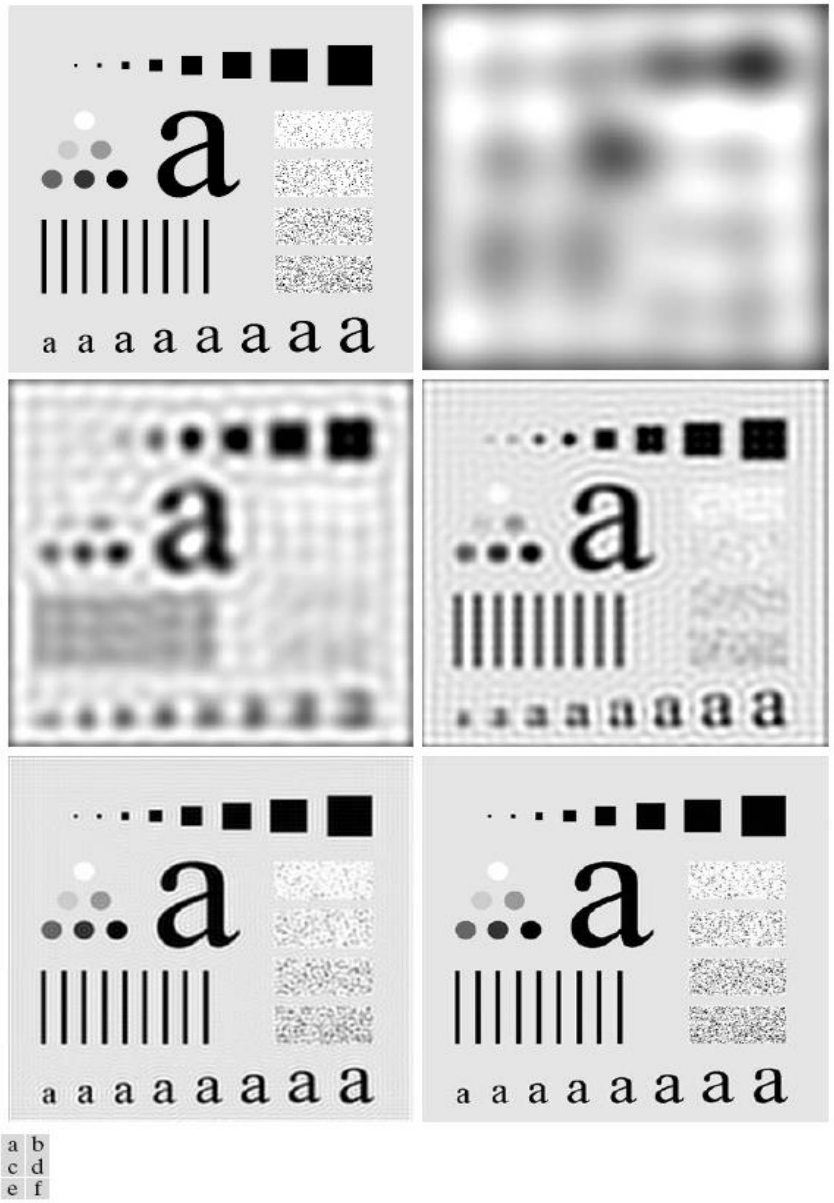


FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

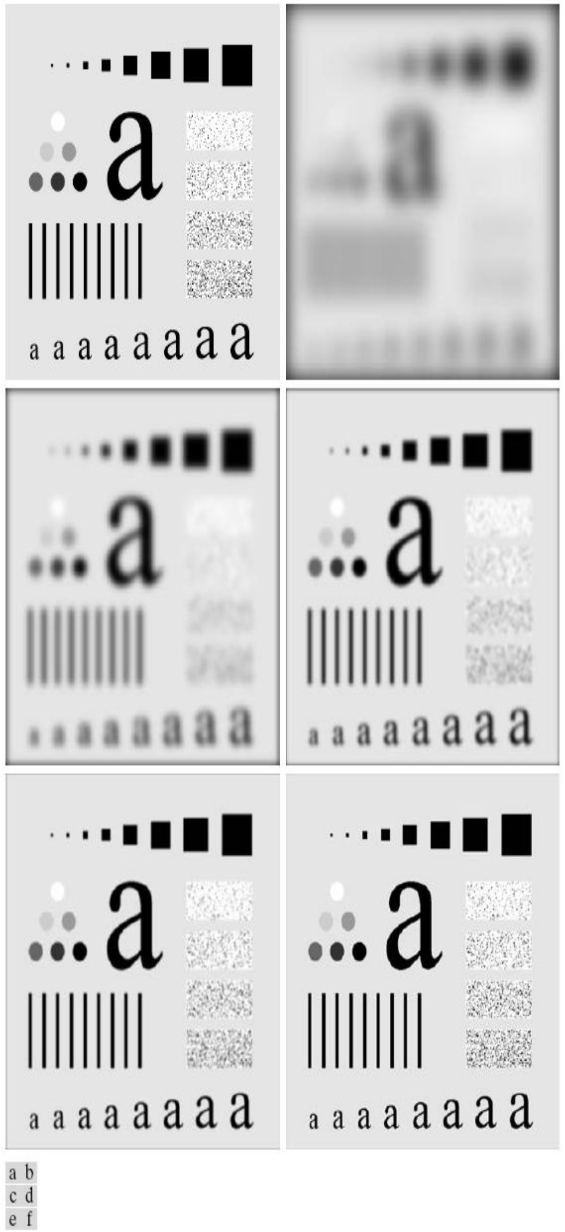
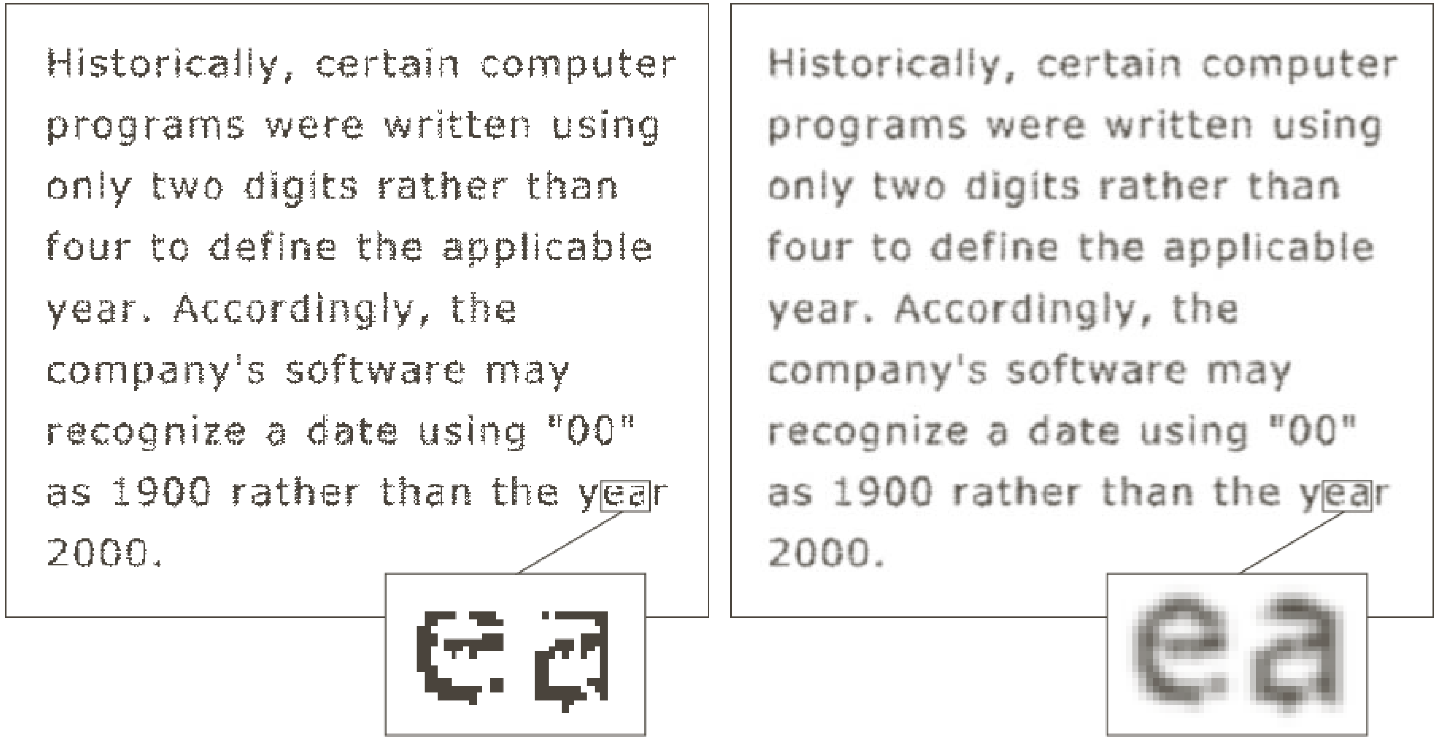


FIGURE 4.48 (a) Original image. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Figs. 4.42 and 4.45.

Examples of smoothing by GLPF (1)



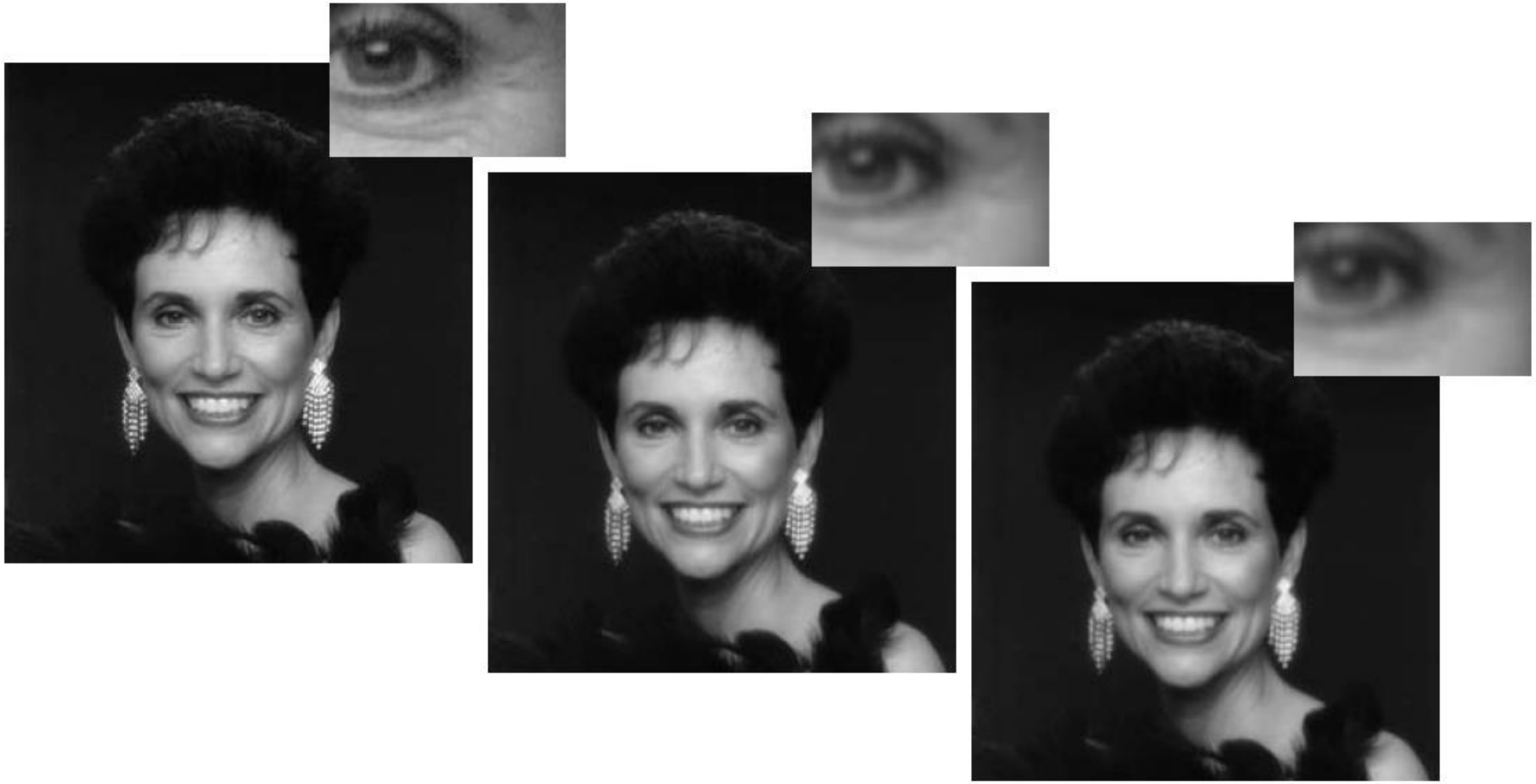
Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

a b

FIGURE 4.49
(a) Sample text of low resolution (note broken characters in magnified view).
(b) Result of filtering with a GLPF (broken character segments were joined).

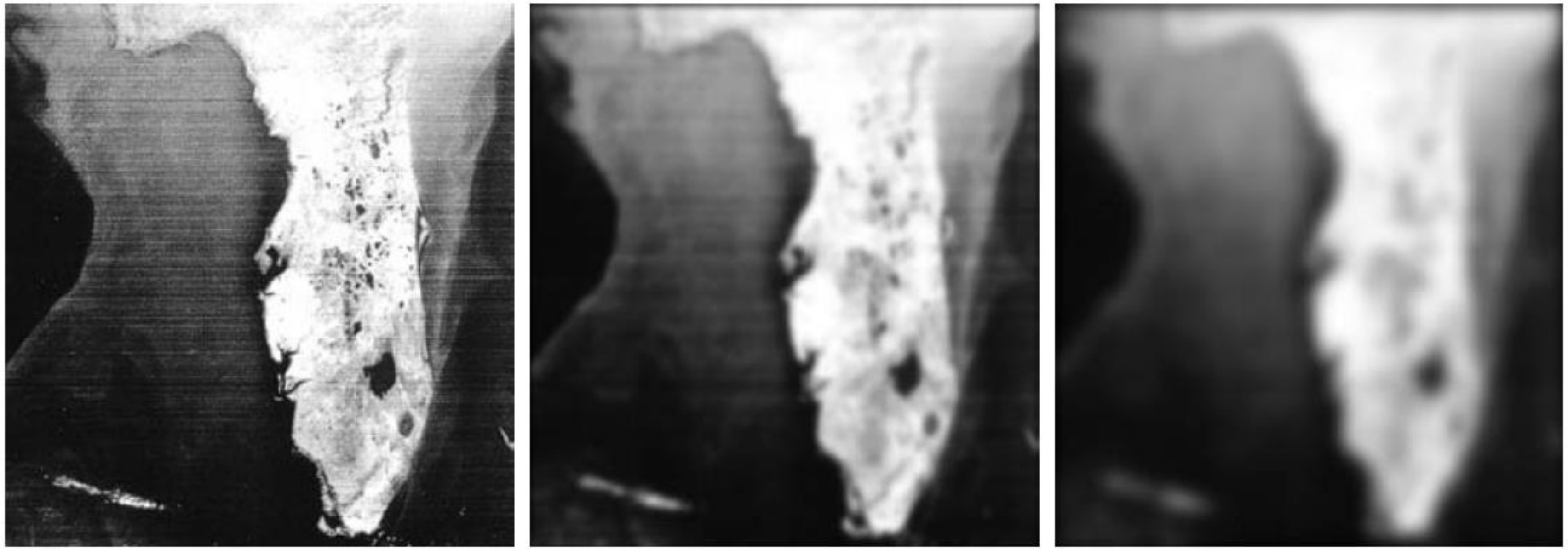
Examples of smoothing by GLPF (2)



a b c

FIGURE 4.50 (a) Original image (784 × 732 pixels). (b) Result of filtering using a GLPF with $D_0 = 100$. (c) Result of filtering using a GLPF with $D_0 = 80$. Note the reduction in fine skin lines in the magnified sections in (b) and (c).

Examples of smoothing by GLPF (3)



a b c

FIGURE 4.51 (a) Image showing prominent horizontal scan lines. (b) Result of filtering using a GLPF with $D_0 = 50$. (c) Result of using a GLPF with $D_0 = 20$. (Original image courtesy of NOAA.)

Image Sharpening Using Frequency Domain Filters

A highpass filter is obtained from a given lowpass filter using

$$H_{HP}(u,v) = 1 - H_{LP}(u,v)$$

A 2-D ideal highpass filter (IHPL) is defined as

$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

Image Sharpening Using Frequency Domain Filters

A 2-D Butterworth highpass filter (BHPL) is defined as

$$H(u,v) = \frac{1}{1+\left[D_0 / D(u,v)\right]^{2n}}$$

A 2-D Gaussian highpass filter (GHPL) is defined as

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

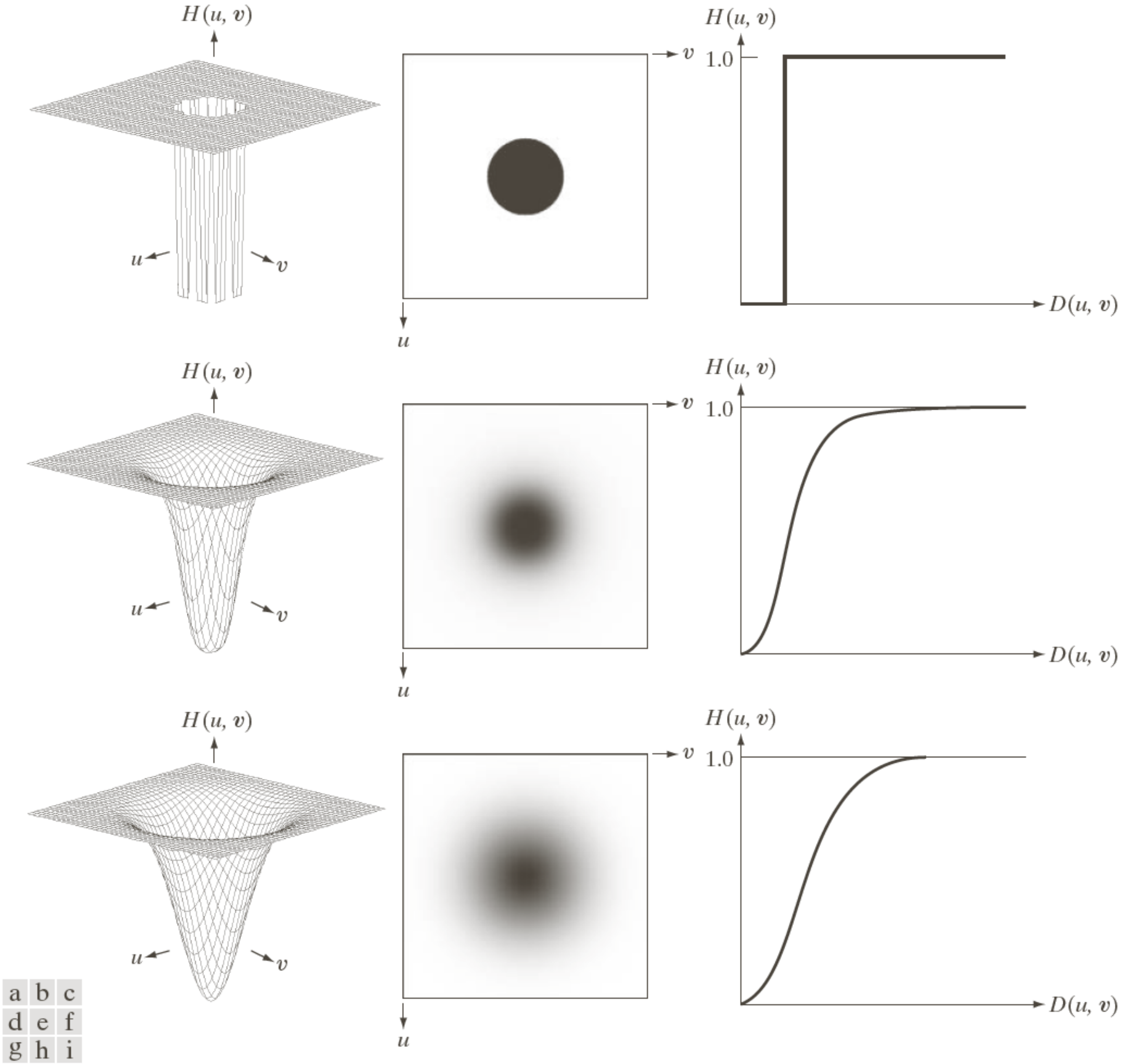


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

The Spatial Representation of Highpass Filters

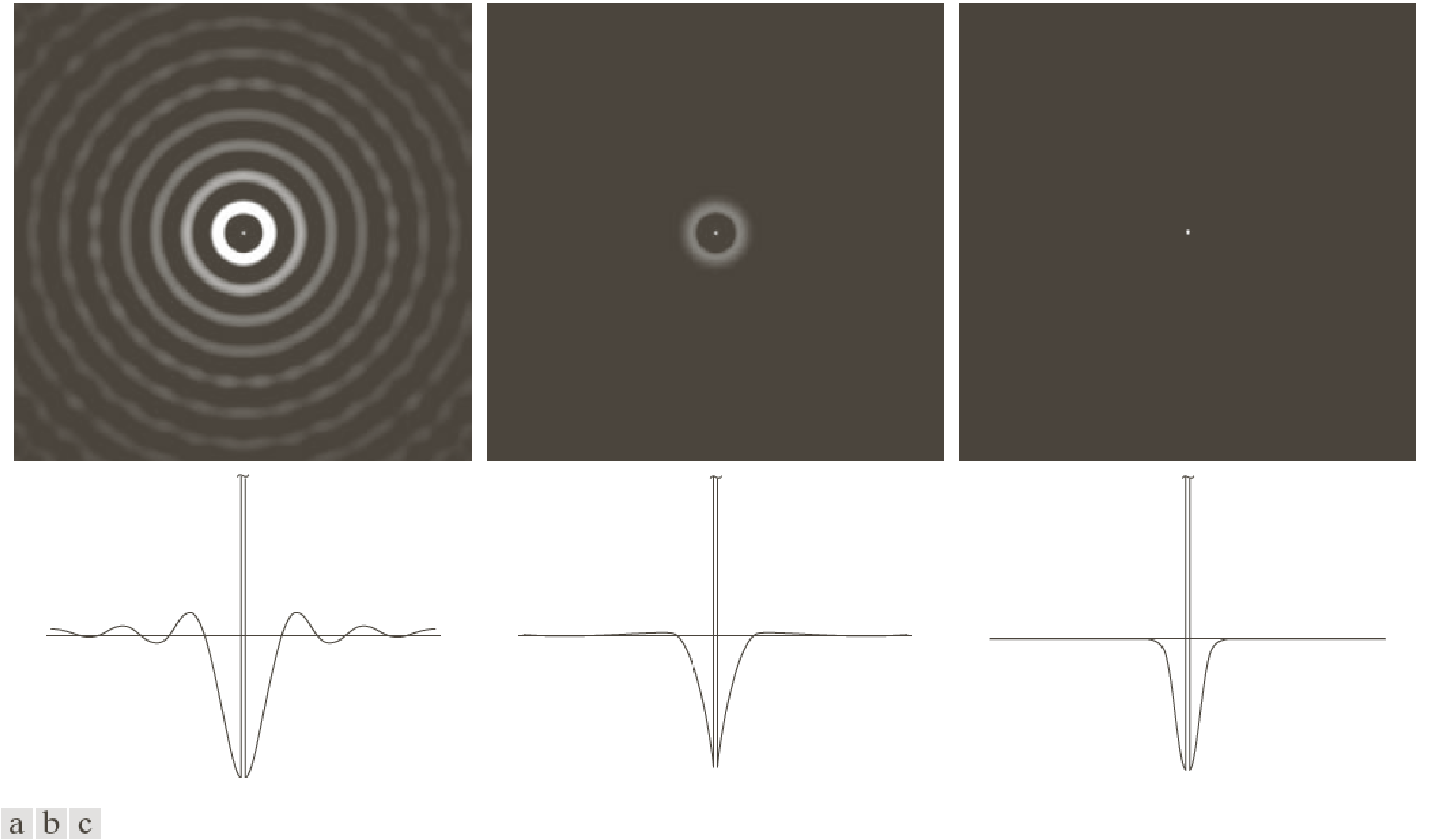


FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

Filtering Results by IHPF

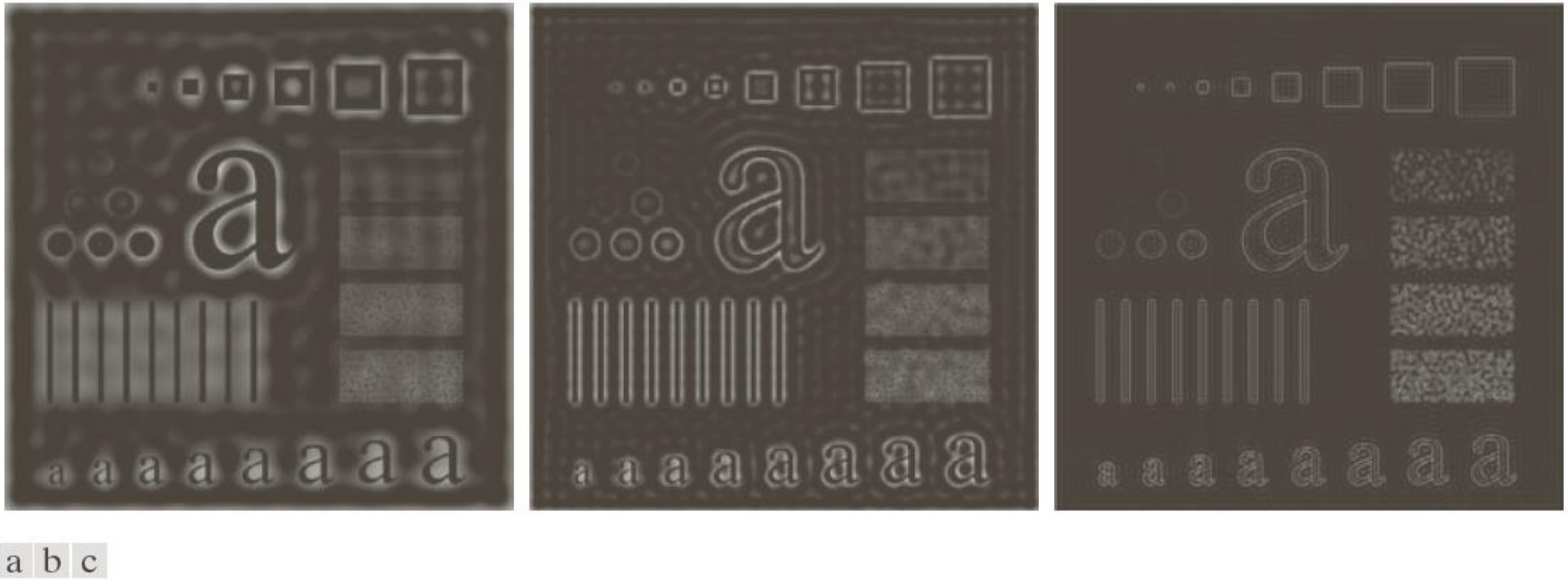


FIGURE 4.54 Results of highpass filtering the image in Fig. 4.41(a) using an IHPF with $D_0 = 30$, 60, and 160.

Filtering Results by BHPF

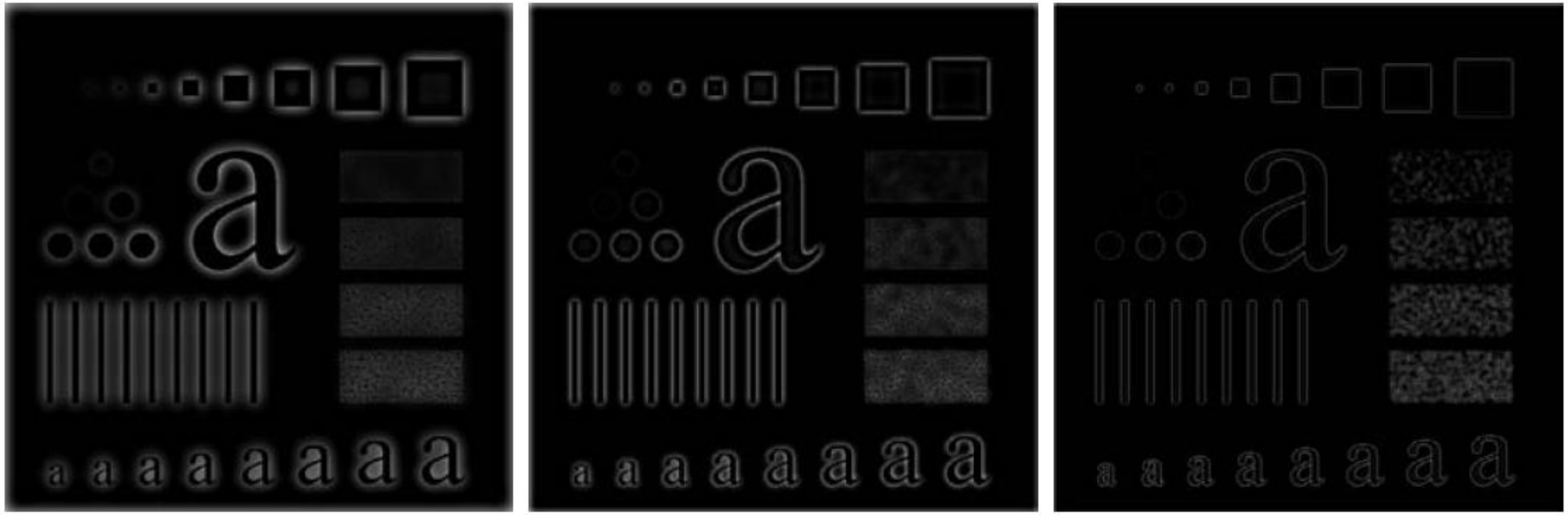


FIGURE 4.55 Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with $D_0 = 30, 60$, and 160, corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPF.

Filtering Results by GHPF

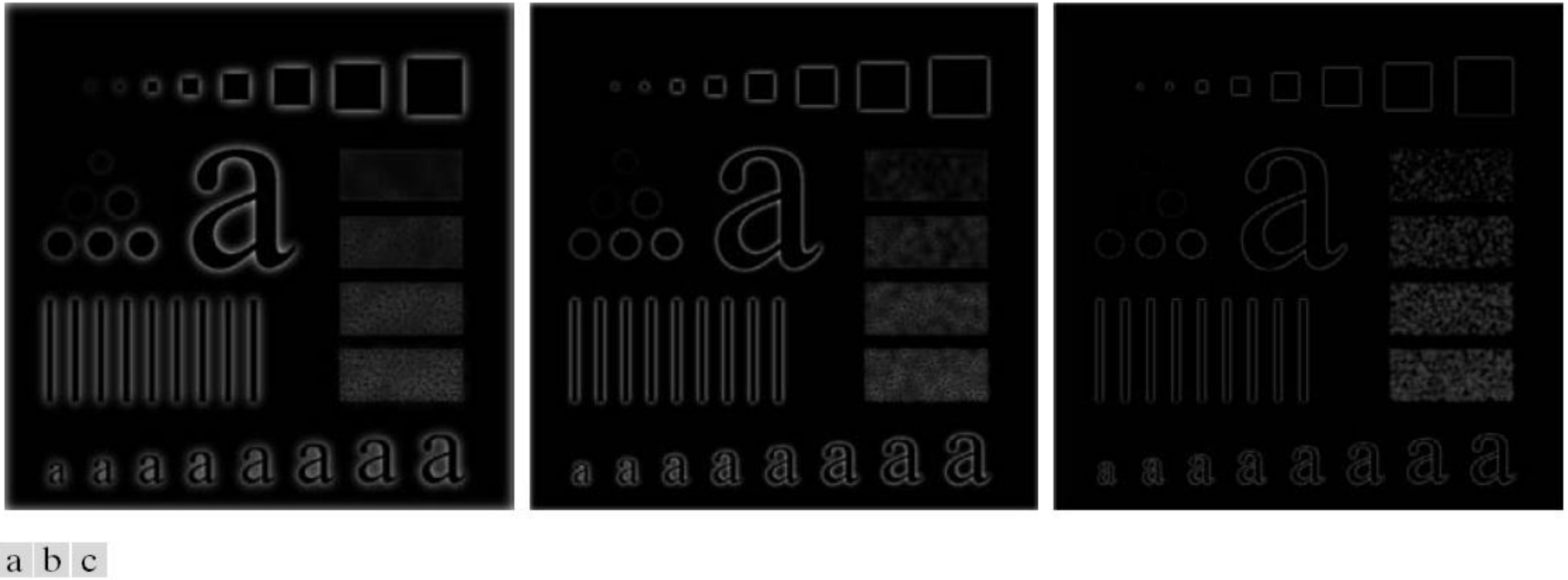
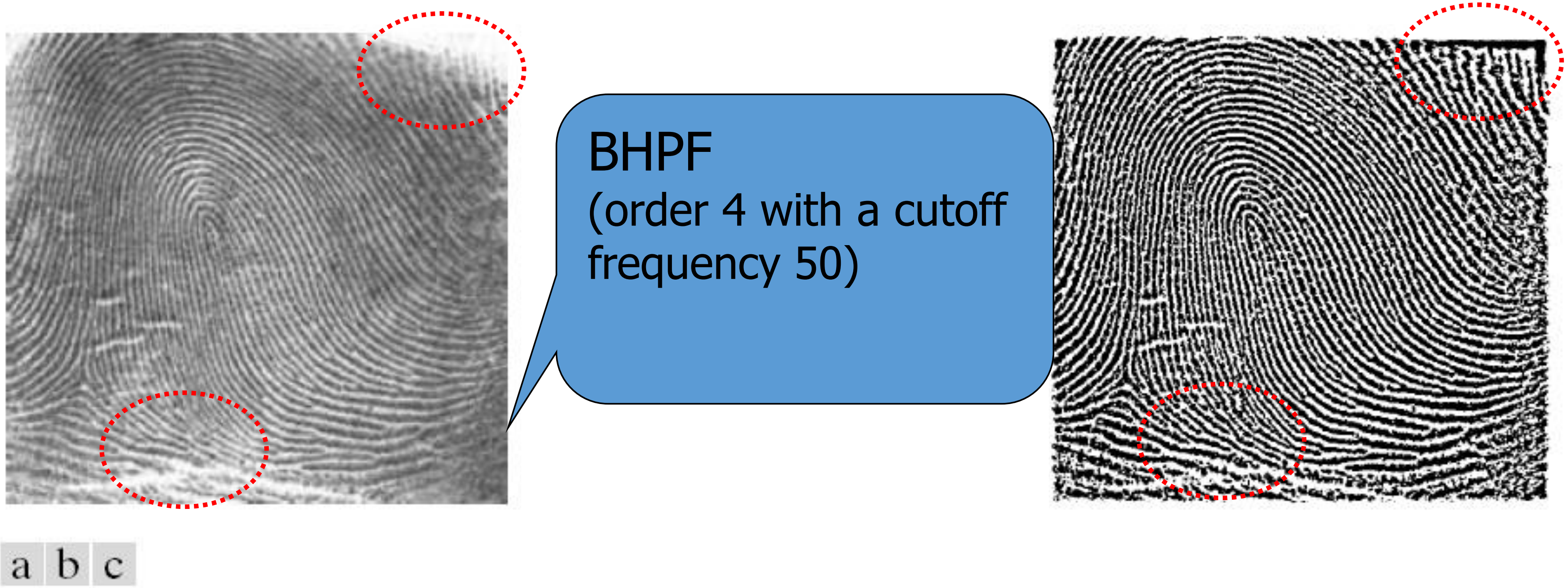


FIGURE 4.56 Results of highpass filtering the image in Fig. 4.41(a) using a GHPF with $D_0 = 30$, 60, and 160, corresponding to the circles in Fig. 4.41(b). Compare with Figs. 4.54 and 4.55.

Using Highpass Filtering and Threshold for Image Enhancement



BHPF
(order 4 with a cutoff frequency 50)

FIGURE 4.57 (a) Thumb print. (b) Result of highpass filtering (a). (c) Result of thresholding (b). (Original image courtesy of the U.S. National Institute of Standards and Technology.)

The Laplacian in the Frequency Domain

$$H(u,v) = -4\pi^2(u^2+v^2)$$

$$H(u,v) = -4\pi^2\left[(u-P/2)^2+(v-Q/2)^2)\right]$$

$$= -4\pi^2 D^2(u,v)$$

The Laplacian image

$$\nabla^2 f(x,y) = \mathfrak{I}^{-1}\left\{H(u,v)F(u,v)\right\}$$

Enhancement is obtained

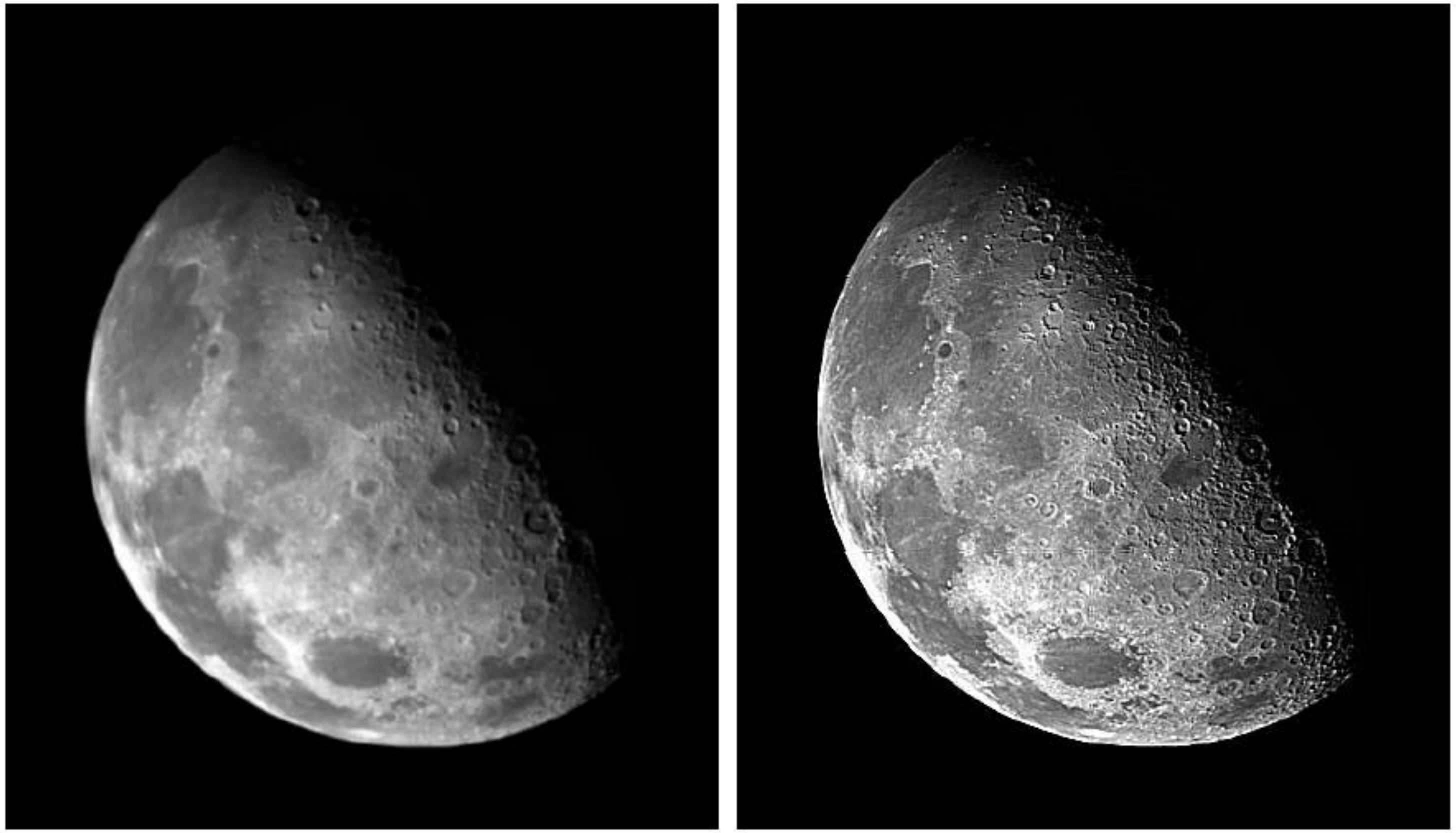
$$g(x,y) = f(x,y) + c\nabla^2 f(x,y) \qquad c = \text{-}1$$

The Laplacian in the Frequency Domain

The enhanced image

$$
\begin{aligned}
g(x,y) &= \Im^{-1}\{F(u,v) - H(u,v)F(u,v)\} \\
&= \Im^{-1}\{[1 - H(u,v)]F(u,v)\} \\
&= \Im^{-1}\{[1 + 4\pi^2 D^2(u,v)]F(u,v)\}
\end{aligned}
$$

The Laplacian in the Frequency Domain



a b

FIGURE 4.58 (a) Original, blurry image. (b) Image enhanced using the Laplacian in the frequency domain. Compare with Fig. 3.38(e).

Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Fitering

$$g_{mask}(x,y) = f(x,y) - f_{LP}(x,y)$$

$$f_{LP}(x,y) = \Im^{-1}\left[H_{LP}(u,v)F(u,v)\right]$$

Unsharp masking and highboost filtering

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

$$\begin{aligned} g(x,y) &= \Im^{-1}\left\{\left[1 + k * \left[1 - H_{LP}(u,v)\right]\right]F(u,v)\right\} \\ &= \Im^{-1}\left\{\left[1 + k * H_{HP}(u,v)\right]F(u,v)\right\} \end{aligned}$$

Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Fitering

$$g(x,y)=\Im^{-1}\left\{\left[k_1+k_2*H_{HP}(u,v)\right]F(u,v)\right\}$$

$$k_1\geq 0 \ \ \text{and} \ \ k_2\geq 0$$

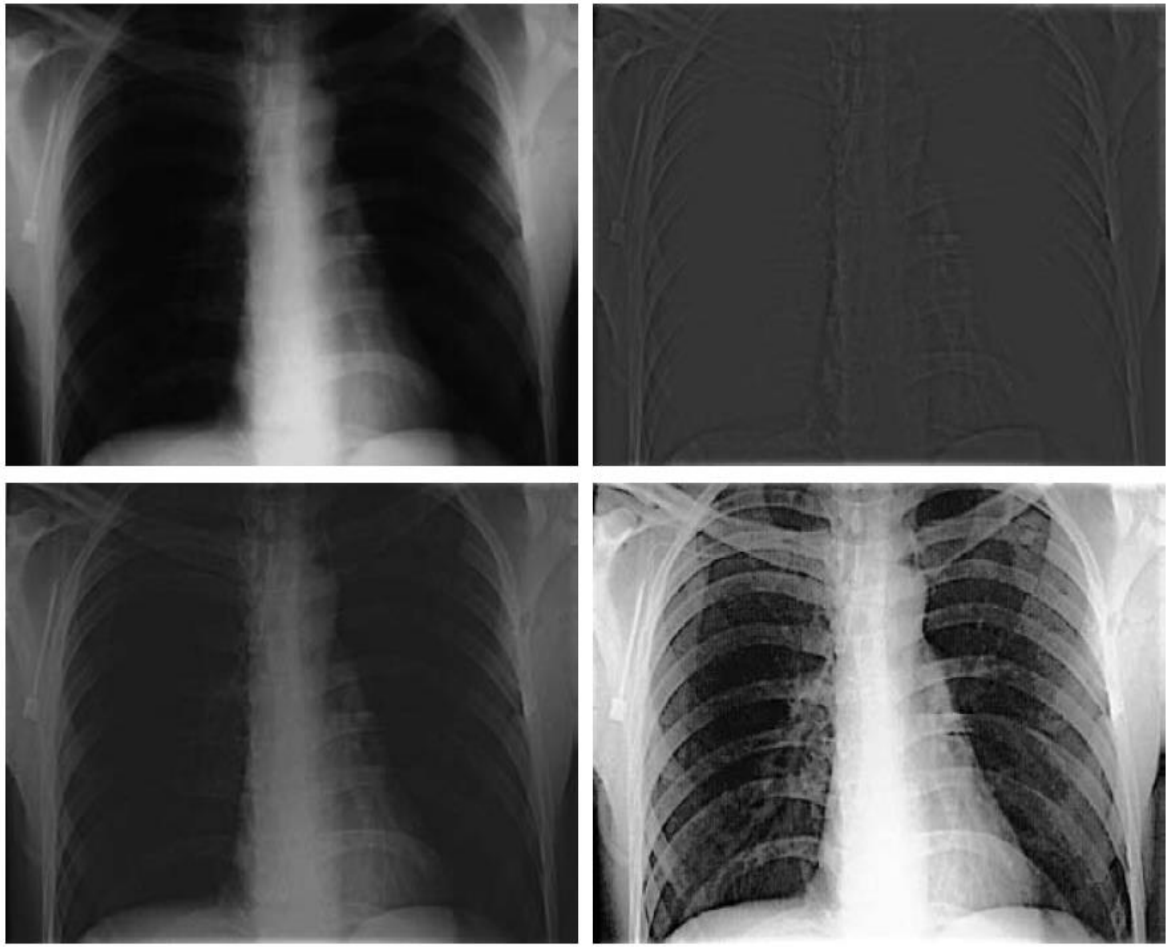


FIGURE 4.59 (a) A chest X-ray image. (b) Result of highpass filtering with a Gaussian filter. (c) Result of high-frequency-emphasis filtering using the same filter. (d) Result of performing histogram equalization on (c). (Original image courtesy of Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

Homomorphic Filtering

$$f(x,y) = i(x,y)r(x,y)$$

$$\Im\left[f(x,y)\right] \neq \Im\left[i(x,y)\right]\Im\left[r(x,y)\right]$$?

$$z(x,y) = \ln f(x,y) = \ln i(x,y) + \ln r(x,y)$$

$$\Im\{z(x,y)\} = \Im\{\ln f(x,y)\} = \Im\{\ln i(x,y)\} + \Im\{\ln r(x,y)\}$$

$$Z(u,v) = F_i(u,v) + F_r(u,v)$$

Homomorphic Filtering

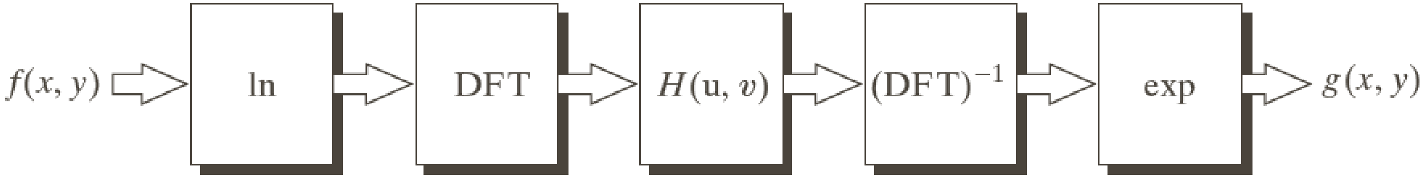
$$\begin{aligned} S(u,v) &= H(u,v)Z(u,v) \\ &= H(u,v)F_i(u,v) + H(u,v)F_r(u,v) \end{aligned}$$

$$\begin{aligned} s(x,y) &= \Im^{-1}\left\{S(u,v)\right\} \\ &= \Im^{-1}\left\{H(u,v)F_i(u,v) + H(u,v)F_r(u,v)\right\} \\ &= \Im^{-1}\left\{H(u,v)F_i(u,v)\right\} + \Im^{-1}\left\{H(u,v)F_r(u,v)\right\} \\ &= i'(x,y) + r'(x,y) \end{aligned}$$

$$g(x,y) = e^{s(x,y)} = e^{i'(x,y)}e^{r'(x,y)} = i_0(x,y)r_0(x,y)$$

Homomorphic Filtering

FIGURE 4.60 Summary of steps in homomorphic filtering.

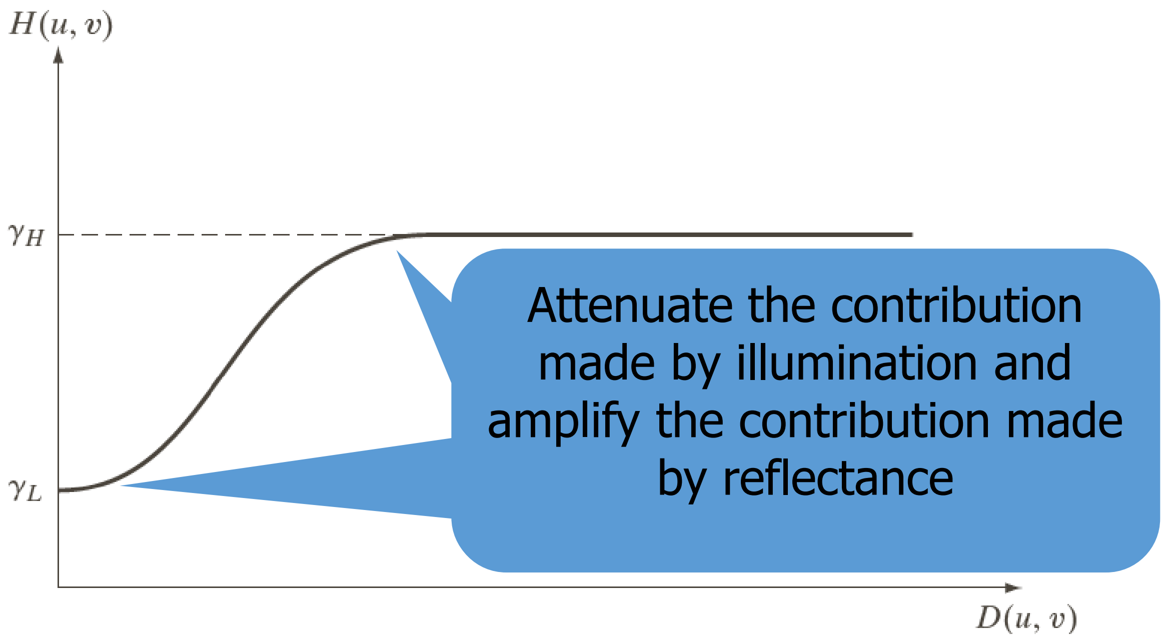


The illumination component of an image generally is characterized by slow spatial variations, while the reflectance component tends to vary abruptly

These characteristics lead to associating the low frequencies of the Fourier transform of the logarithm of an image with illumination the high frequencies with reflectance.

Homomorphic Filtering

$$H(u,v) = (\gamma_H - \gamma_L)\left[1 - e^{-c\left[D^2(u,v)/D_0^2\right]}\right] + \gamma_L$$



Attenuate the contribution made by illumination and amplify the contribution made by reflectance

FIGURE 4.61 Radial cross section of a circularly symmetric homomorphic filter function. The vertical axis is at the center of the frequency rectangle and $D(u,v)$ is the distance from the center.

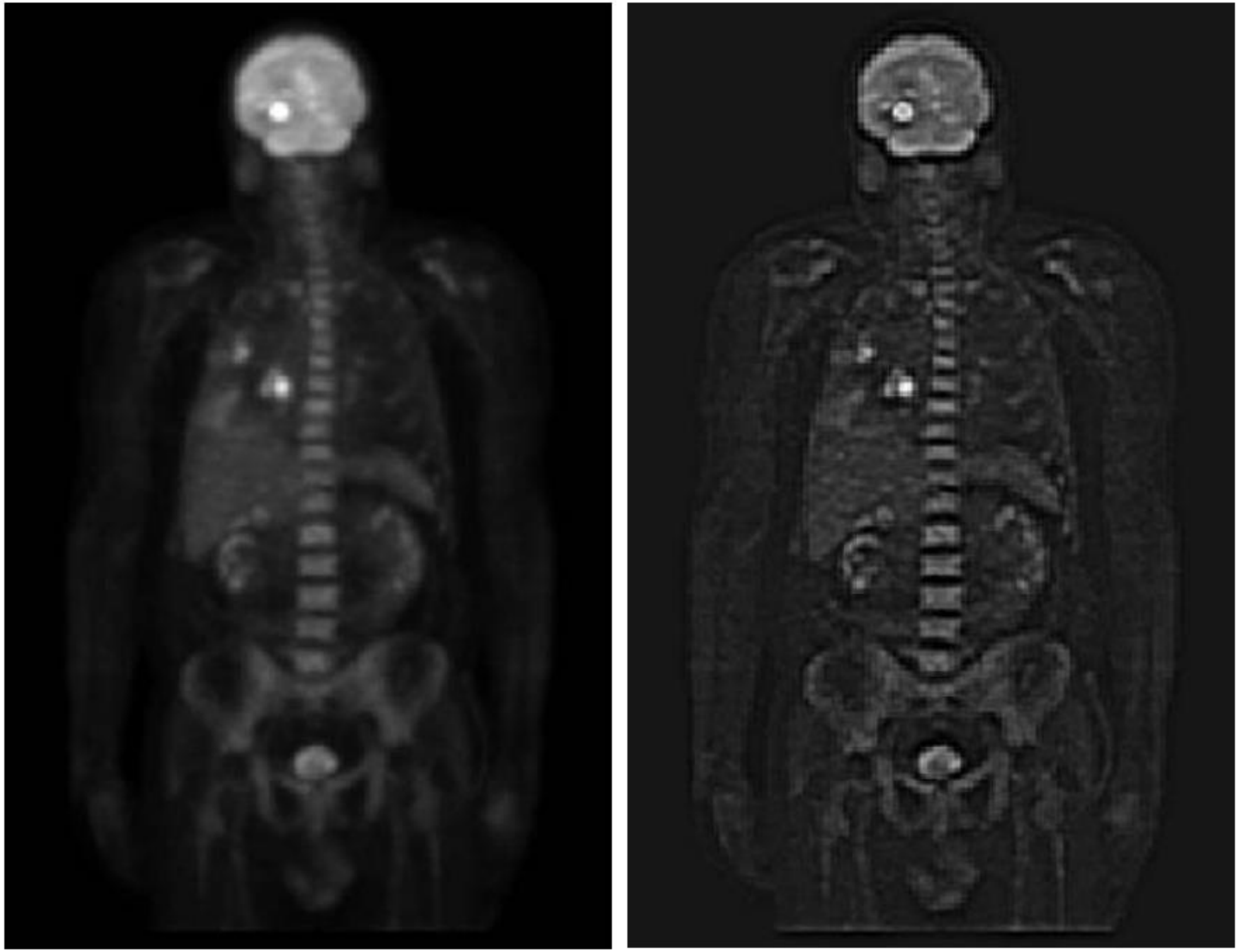
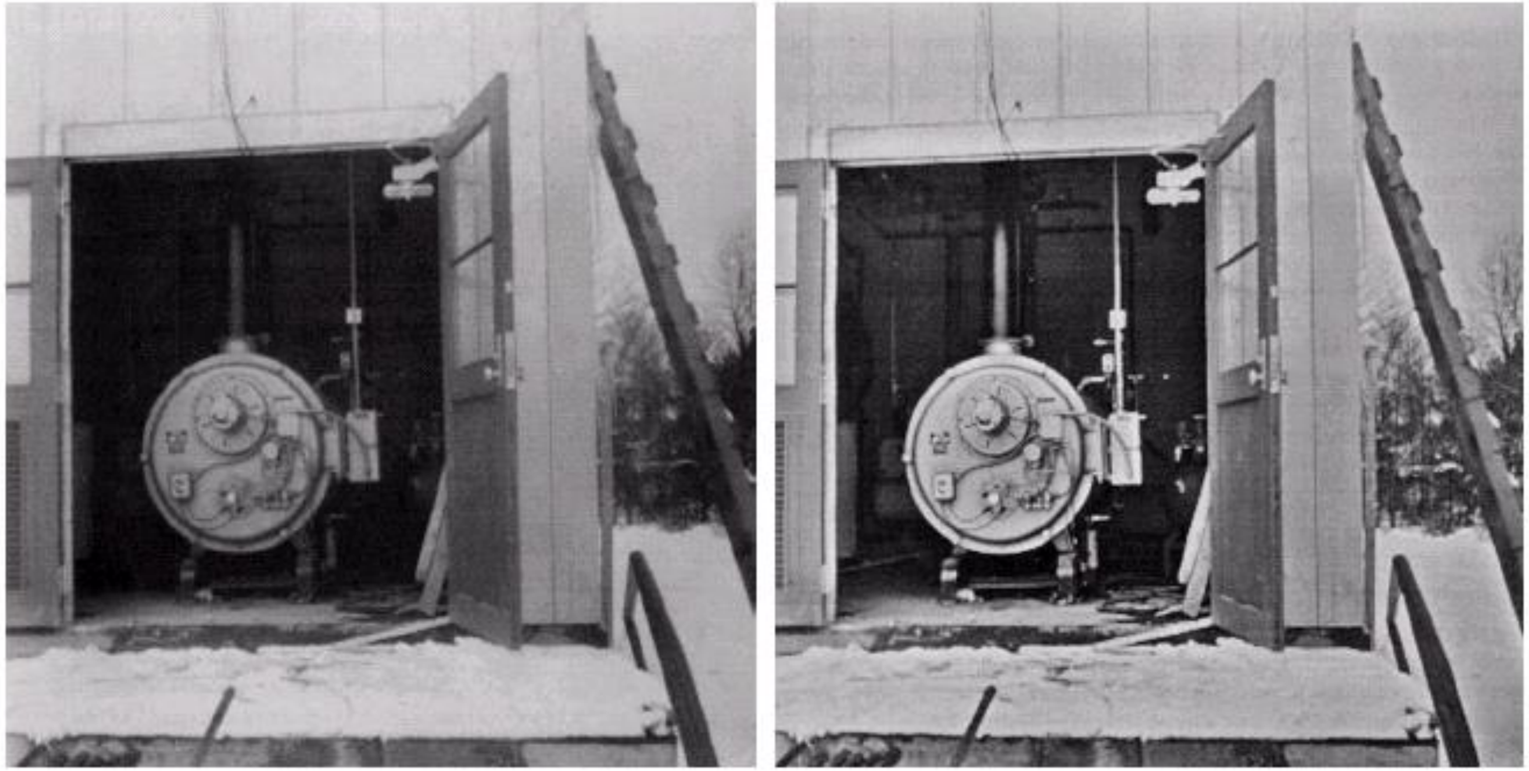


FIGURE 4.62 (a) Full body PET scan. (b) Image enhanced using homomorphic filtering. (Original image courtesy of Dr. Michael E. Casey, CTI PET Systems.)

Homomorphic Filtering

a b

FIGURE (a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)



Selective Filtering

Non-Selective Filters:
operate over the entire frequency rectangle

Selective Filters
operate over some part, not entire frequency rectangle

- **bandreject or bandpass**: process specific bands
- **notch filters**: process small regions of the frequency rectangle

Selective Filtering:
Bandreject and Bandpass Filters

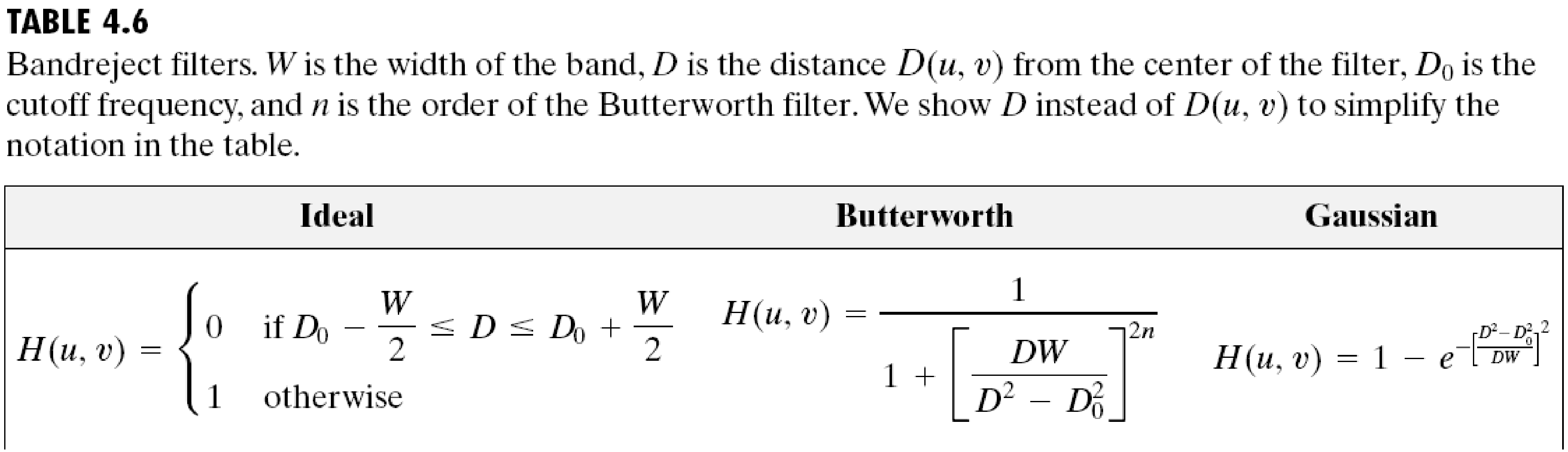


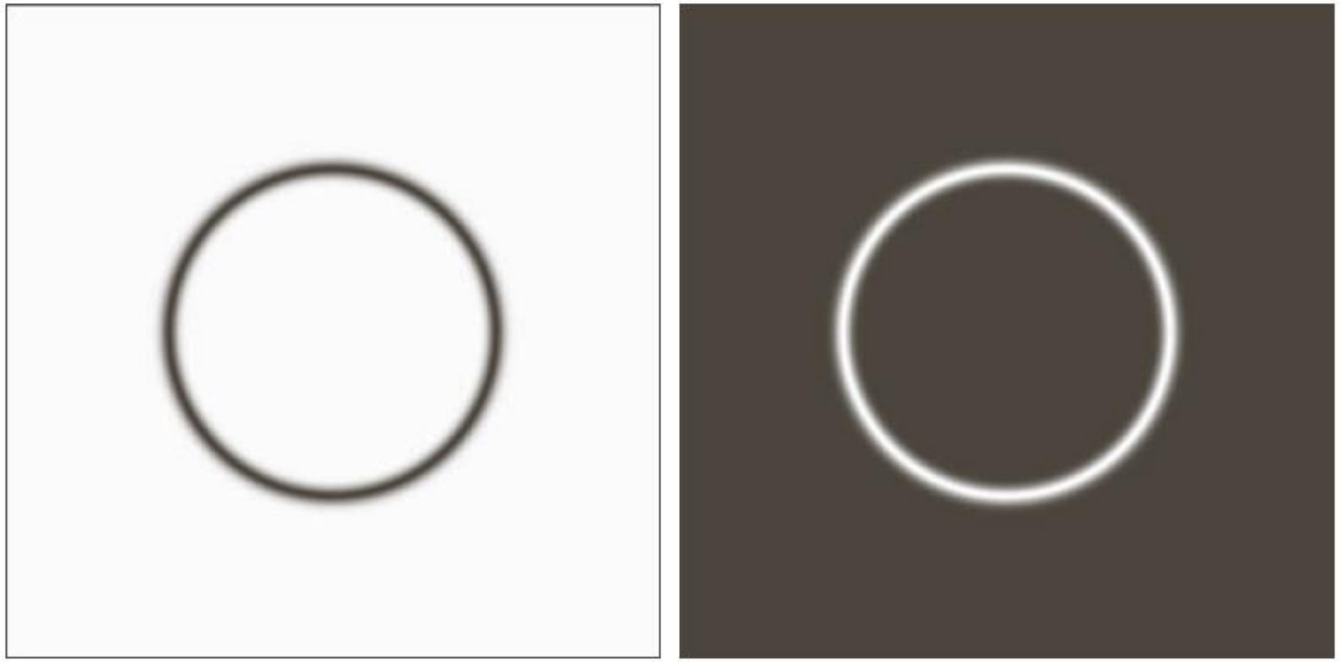
TABLE 4.6

Bandreject filters. W is the width of the band, D is the distance $D(u, v)$ from the center of the filter, D_0 is the cutoff frequency, and n is the order of the Butterworth filter. We show D instead of $D(u, v)$ to simplify the notation in the table.

Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \le D \le D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = \dfrac{1}{1 + \left[\dfrac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u,v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

Selective Filtering:
Bandreject and Bandpass Filters



a b

FIGURE 4.63 (a) Bandreject Gaussian filter. (b) Corresponding bandpass filter. The thin black border in (a) was added for clarity; it is not part of the data.

Selective Filtering: Notch Filters

Zero-phase-shift filters must be symmetric about the origin. A notch with center at (u_0, v_0) must have a corresponding notch at location ($-u_0$,$-v_0$).

Notch reject filters are constructed as products of highpass filters whose centers have been translated to the centers of the notches.

$$H_{NR}(u,v) = \prod_{k=1}^{Q} H_k(u,v)H_{-k}(u,v)$$

where $H_k(u,v)$ and $H_{\text{-}k}(u,v)$ are highpass filters whose centers are at (u_k, v_k) and $(\text{-}u_k, \text{-}v_k)$, respectively.

Selective Filtering:
Notch Filters

$$H_{NR}(u,v) = \prod_{k=1}^{Q} H_k(u,v) H_{-k}(u,v)$$

where $H_k(u,v)$ and $H_{-k}(u,v)$ are highpass filters whose centers are at (u_k, v_k) and $(-u_k, -v_k)$, respectively.

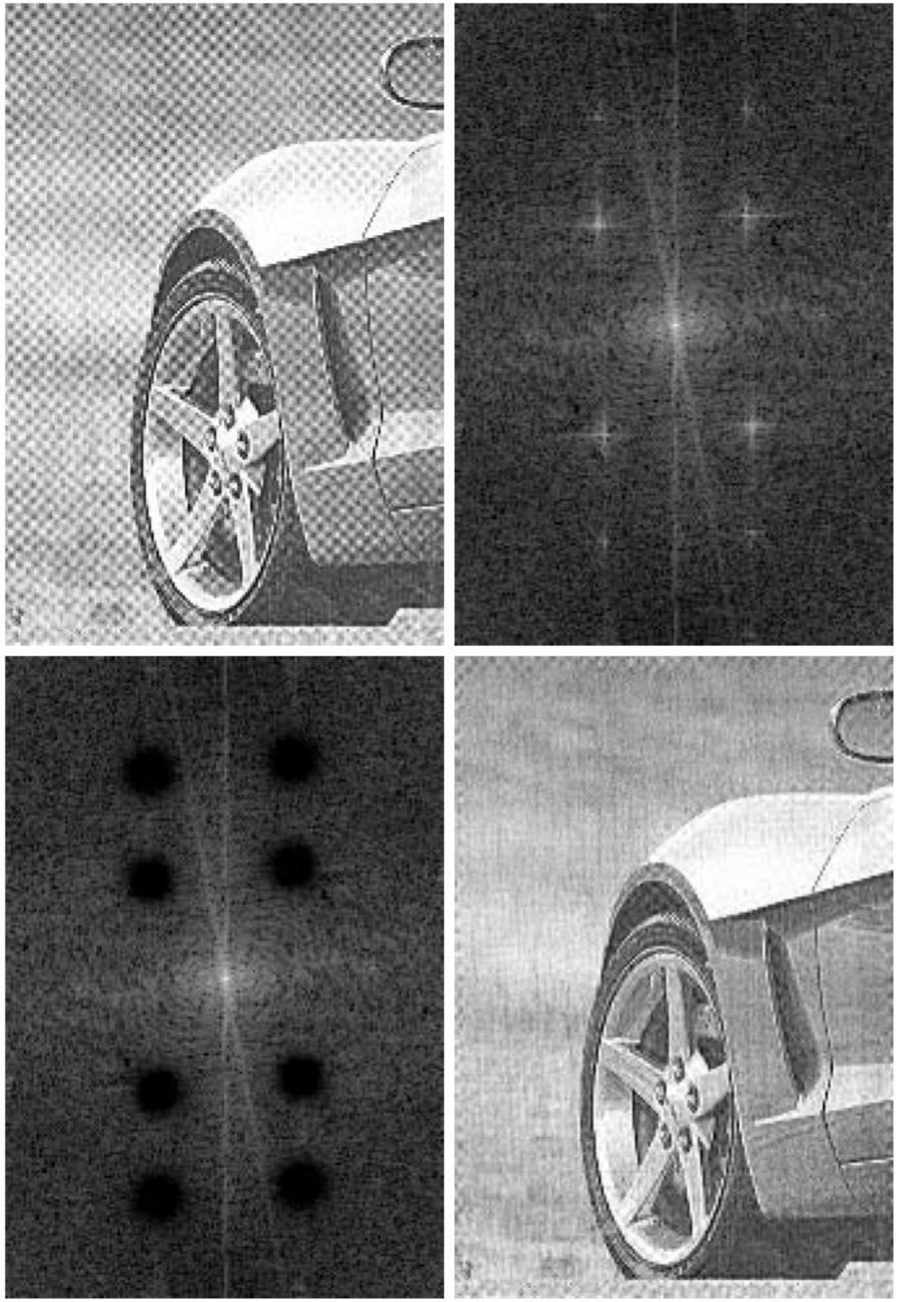
A Butterworth notch reject filter of order n

$$H_{NR}(u,v) = \prod_{k=1}^{3} \left[\frac{1}{1 + \left[D_{0k} / D_k(u,v) \right]^{2n}} \right] \left[\frac{1}{1 + \left[D_{0k} / D_{-k}(u,v) \right]^{2n}} \right]$$

$$D_k(u,v) = \left[(u - M/2 - u_k)^2 + (v - N/2 - v_k)^2 \right]^{1/2}$$

$$D_{-k}(u,v) = \left[(u - M/2 + u_k)^2 + (v - N/2 + v_k)^2 \right]^{1/2}$$

Examples: Notch Filters (1)

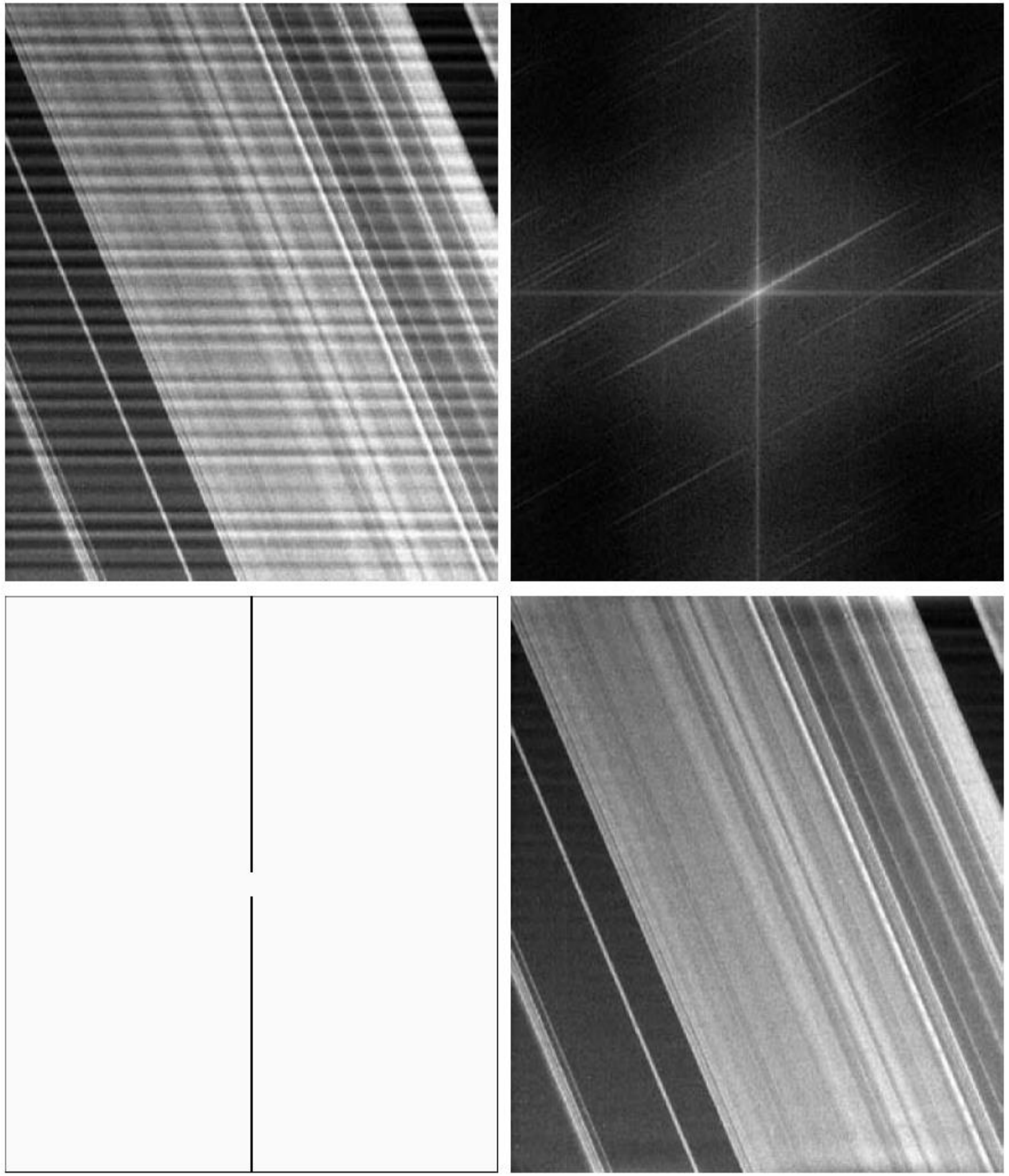


a b
c d

FIGURE 4.64 (a) Sampled newspaper image showing a moiré pattern. (b) Spectrum. (c) Butterworth notch reject filter multiplied by the Fourier transform. (d) Filtered image.

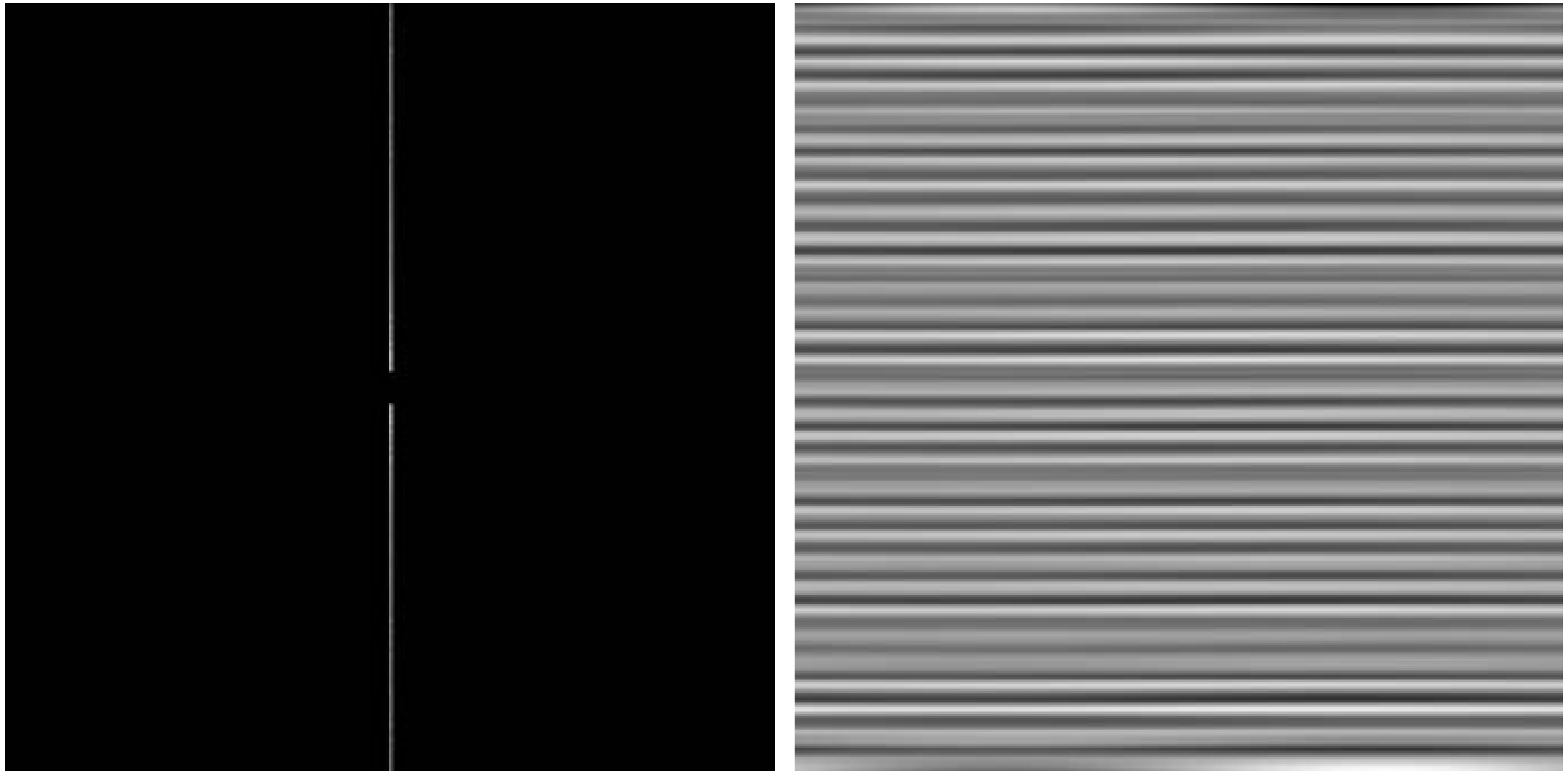
A Butterworth notch reject filter D_0=3 and n=4 for all notch pairs

Examples: Notch Filters (2)



a	b
c	d

FIGURE 4.65 (a) 674 × 674 image of the Saturn rings showing nearly periodic interference. (b) Spectrum: The bursts of energy in the vertical axis near the origin correspond to the interference pattern. (c) A vertical notch reject filter. (d) Result of filtering. The thin black border in (c) was added for clarity; it is not part of the data. (Original image courtesy of Dr. Robert A. West, NASA/JPL.)



a b

FIGURE 4.66 (a) Result (spectrum) of applying a notch pass filter to the DFT of Fig. 4.65(a). (b) Spatial pattern obtained by computing the IDFT of (a).