Chapter 4
Image Enhancement in the Frequency Domain

Yinghua He

School of Computer Science and Technology
Tianjin University

- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

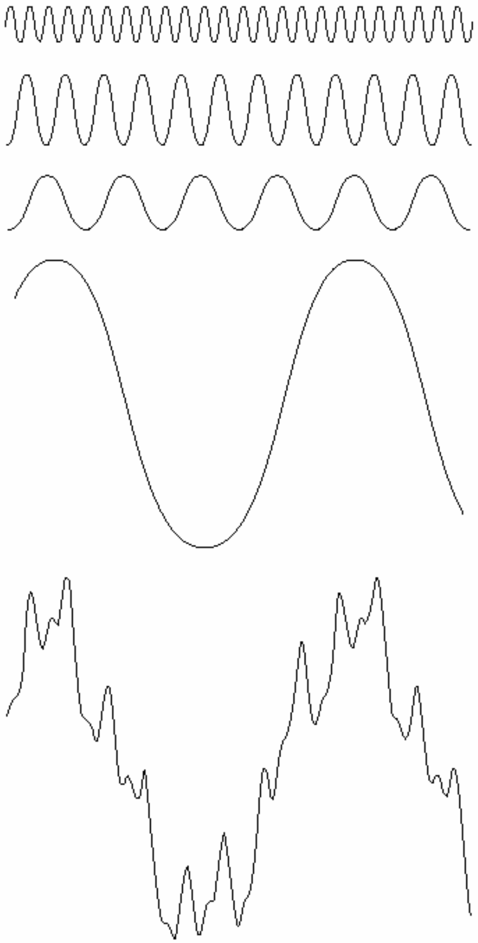


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

- The One-Dimensional Fourier Transform and its Inverse
- The Two-Dimensional Fourier Transform and its Inverse
- Filtering in the Frequency Domain
- Correspondence between Filtering in the Spatial and Frequency Domains

- The Fourier transform, F(u), of a single variable, continuous function, f(x), is defined by the equation

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx$$

- Where $j = \sqrt{-1}$
- Conversely, given F(u), we can obtain f(x) by means of the inverse Fourier transform

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

- These two equations comprise the Fourier transform pair.

- These equations are easily extended to two variables, u and v:

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

- And, similarly for the inverse transform,

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

- The Fourier transform of a discrete function of one variable, f(x), x=0,1,2,…,M-1, is given by the equation

$$F(u)=\frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{-j2\pi ux/M} \qquad for \quad u=0,1,2,\ldots,M-1 \qquad (4.2\text{-}5)$$

- Similarly, given F(u), we can obtain the original function back using the inverse DFT:

$$f(x)=\sum_{u=0}^{M-1} F(u)e^{-j2\pi ux/M} \qquad for \quad x=0,1,2,\ldots,M-1$$

- The concept of the frequency domain, follows directly from Euler’s formula:

$$e^{j\theta} = \cos\theta + j\sin\theta$$

- Substituting this expression into Eq. (4.2-5). and using the fact that $\cos(-\theta) = \cos\theta$, gives us

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)\left[\cos 2\pi ux/M \, - \, j\sin 2\pi ux/M\right] \qquad for \quad u = 0,1,2,\ldots,M-1$$

- In the analysis of complex numbers, we find it convenient sometimes to express F(u) in polar coordinates:

$$F(u) = \left|F(u)\right| e^{j\phi(u)}$$

- Where $\left|F(u)\right| = \left[R^2(u) + I^2(u)\right]^{1/2}$
- Is called the magnitude or spectrum of the Fourier transform, and

$$\phi(u) = \tan^{-1}\left[\frac{I(u)}{R(u)}\right]$$

- Is called the phase angle or phase spectrum of the transform.
- Another quantity that is used in this chapter is the power spectrum, defined as the square of the Fourier spectrum:

$$P(u) = \left|F(u)\right|^2 = R^2(u) + I^2(u)$$

- The first value of the sampled function is then $f(x_0)$.
- The kth sample gives us $f(x_0 + k\Delta x)$.
- When we write f(k), it is understood that we are utilizing shorthand notation that really means $f(x_0 + k\Delta x)$.

- f(x) is then understood to mean

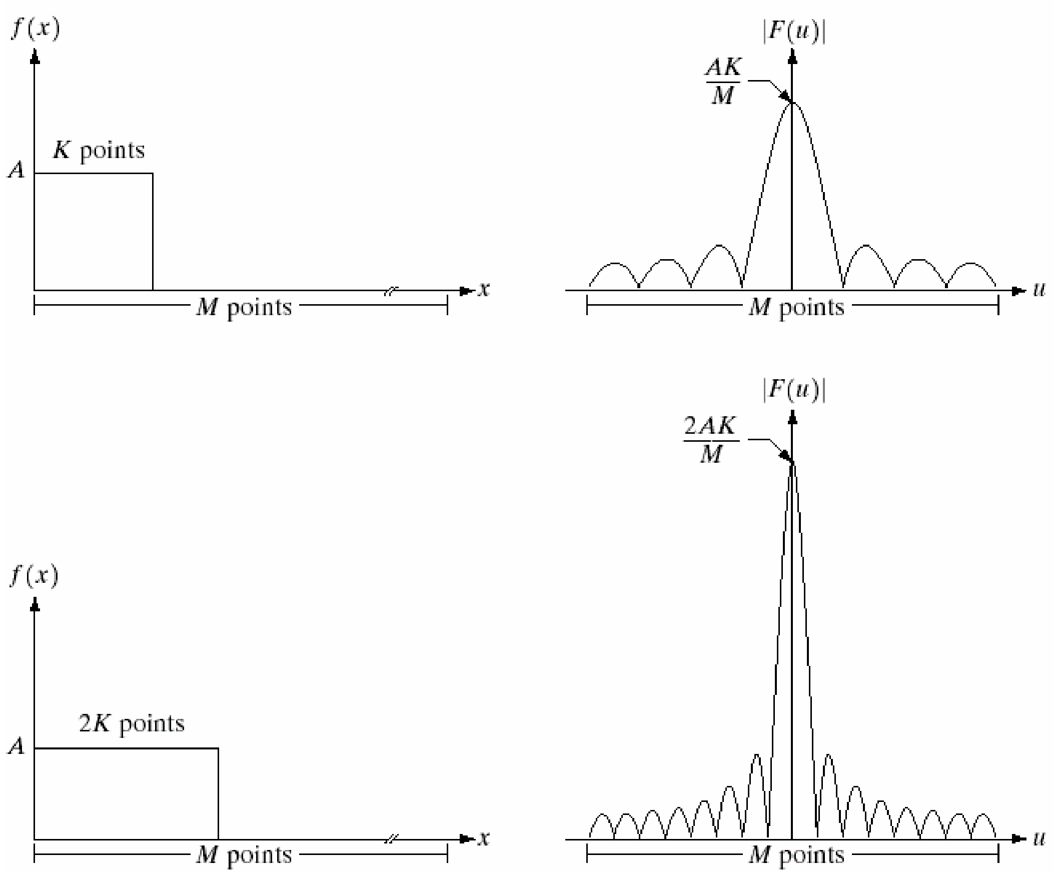
$$f(x) \overset{\Delta}{=} f(x_0 + x\Delta x)$$

- F(u) is then understood to mean

$$F(u) \overset{\Delta}{=} F(u\Delta u)$$

- Δx and Δu are inversely related by the expression

$$\Delta u = \frac{1}{M\Delta x}$$



a b
c d

FIGURE 4.2 (a) A discrete function of *M* points, and (b) its Fourier spectrum. (c) A discrete function with twice the number of nonzero points, and (d) its Fourier spectrum.

- The One-Dimensional Fourier Transform and its Inverse
- The Two-Dimensional Fourier Transform and its Inverse
- Filtering in the Frequency Domain
- Correspondence between Filtering in the Spatial and Frequency Domains

- The discrete Fourier transform of a function (image) f(x,y) of size M*N is given by the equation

$$F(u,v)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-2j\pi(ux/M+vy/N)}$$

- Given F(u,v), we obtain f(x,y) via the inverse Fourier transform, given by the expression

$$f(x,y)=\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{2j\pi(ux/M+vy/N)}$$

for x=0,1,2,…,M-1 and y=0,1,2,…,N-1.

- The Fourier spectrum, phase angle, and power spectrum:

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

$$\phi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right]$$

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

- Where R(u,v) and I(u,v) are the real and imaginary parts of F(u,v), respectively.

- It is common practice to multiply the input image function by $(-1)^{x+y}$ prior to computing the Fourier transform. Due to the properties of exponentials, it is not difficult to show that

$$\Im\left[f(x,y)(-1)^{x+y}\right] = F(u - M/2, v - N/2)$$

- Where $\Im[\cdot]$ denotes the Fourier transform of the argument.

- The value of the transform at (u,v)=(0,0) is

$$F(0,0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

- If f(x,y) is real, its Fourier transform is conjugate symmetric; that is

$$F(u,v) = F^*(-u,-v)$$

- From this, it follows that

$$|F(u,v)| = |F(-u,-v)|$$

- The following relationships between samples in the spatial and frequency domains:

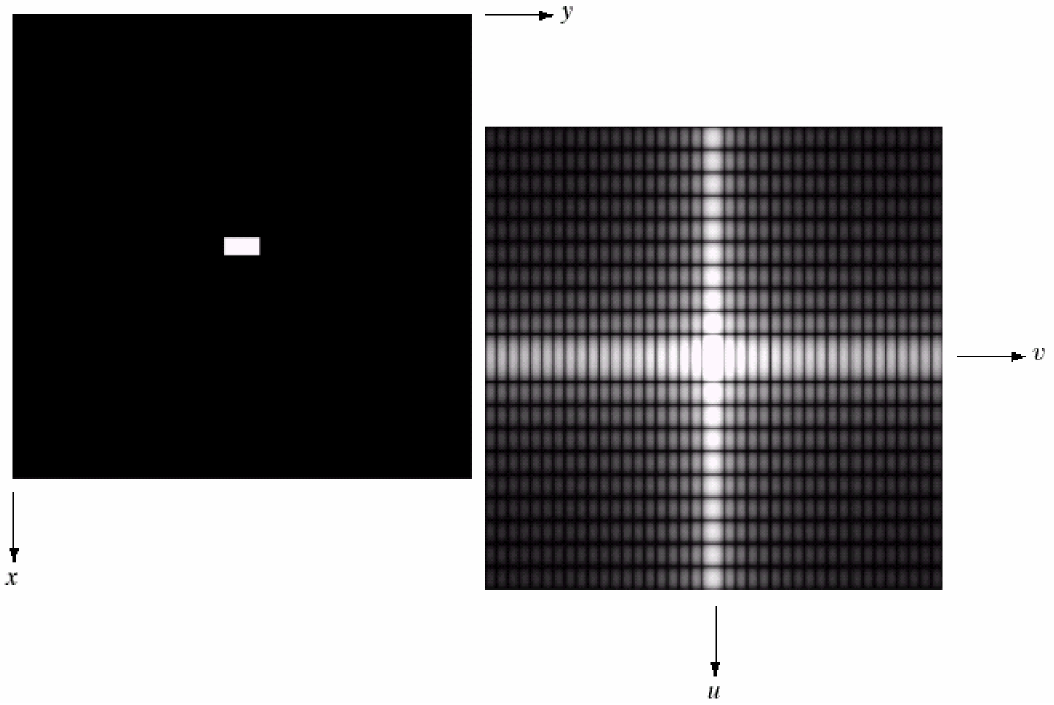
$$\Delta u = \frac{1}{M\Delta x}$$

and

$$\Delta v = \frac{1}{N\Delta y}$$

a b

FIGURE 4.3 (a) Image of a 20×40 white rectangle on a black background of size 512×512 pixels. (b) Centered Fourier spectrum shown after application of the log transformation given in Eq. (3.2-2). Compare with Fig. 4.2.



- The One-Dimensional Fourier Transform and its Inverse
- The Two-Dimensional Fourier Transform and its Inverse
- Filtering in the Frequency Domain
- Correspondence between Filtering in the Spatial and Frequency Domains

- It consists of the following steps:

(1) Multiply the input image by $(-1)^{x+y}$ to center the transform.

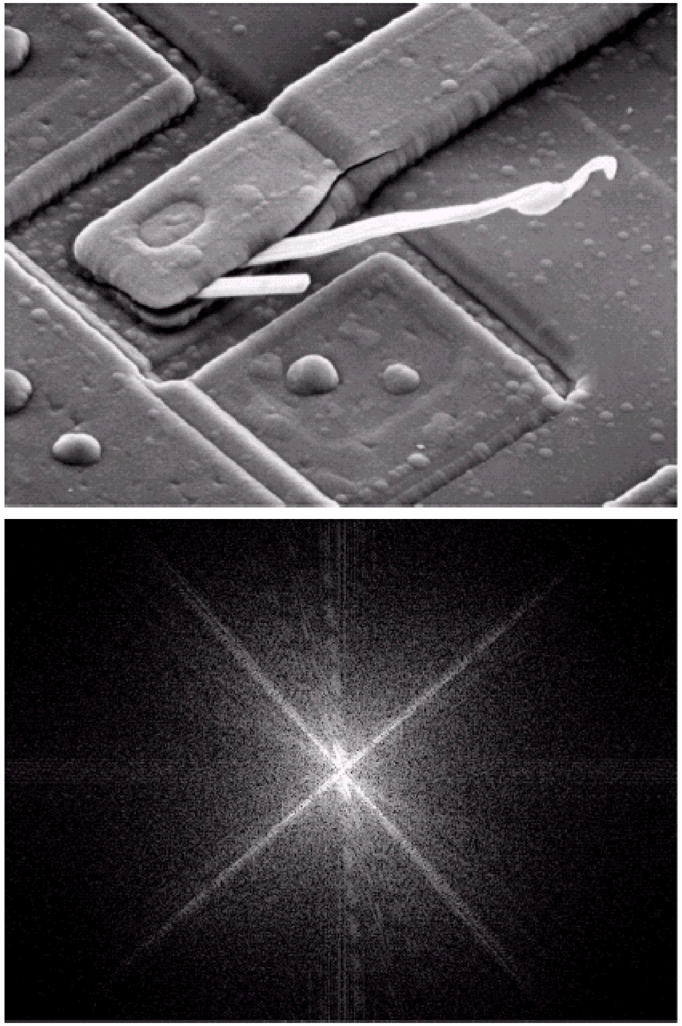
(2) Compute F(u,v), the DFT of the image from (1).

(3) Multiply F(u,v) by a filter function H(u,v).

(4) Compute the inverse DFT of the result in (3).

(5) Obtain the real part of the result in (4).

(6) Multiply the result in (5) by $(-1)^{x+y}$.



a
b

FIGURE 4.4 (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.)

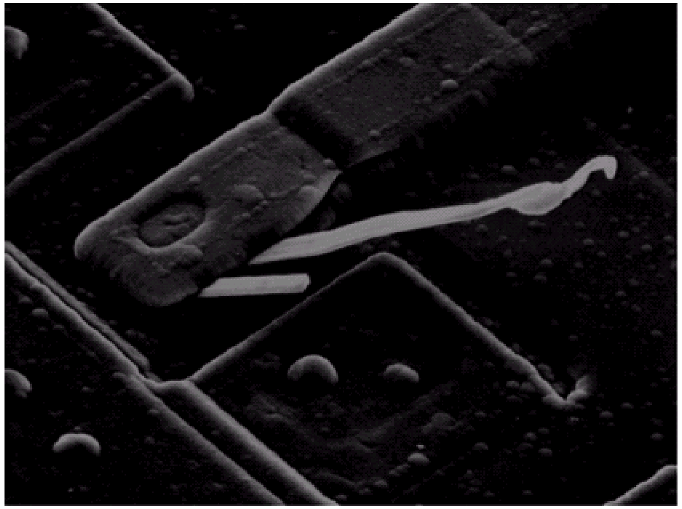
- H(u,v) is called a filter is because it suppresses certain frequencies in the transform while leaving others unchanged.
- The Fourier transform of the output image is given by

$$G(u,v) = H(u,v)F(u,v)$$

- The filtered image is obtained simply by taking the inverse Fourier transform of G(u,v):

Filtered Image= $\Im^{-1}[G(u,v)]$

FIGURE 4.6 Result of filtering the image in Fig. 4.4(a) with a notch filter that set to 0 the $F(0, 0)$ term in the Fourier transform.



- Assuming that the transform has been centered, we can do this operation by multiplying all values of F(u,v) by the filter function:

$$H(u,v) = \begin{cases} 0 & if \quad (u,v) = (M/2, N/2) \\ 1 & otherwise \end{cases}$$

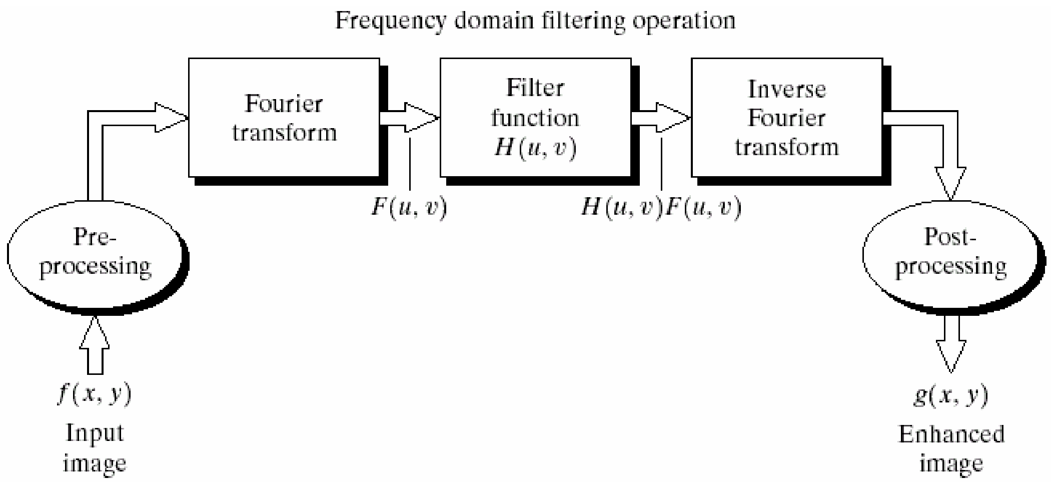
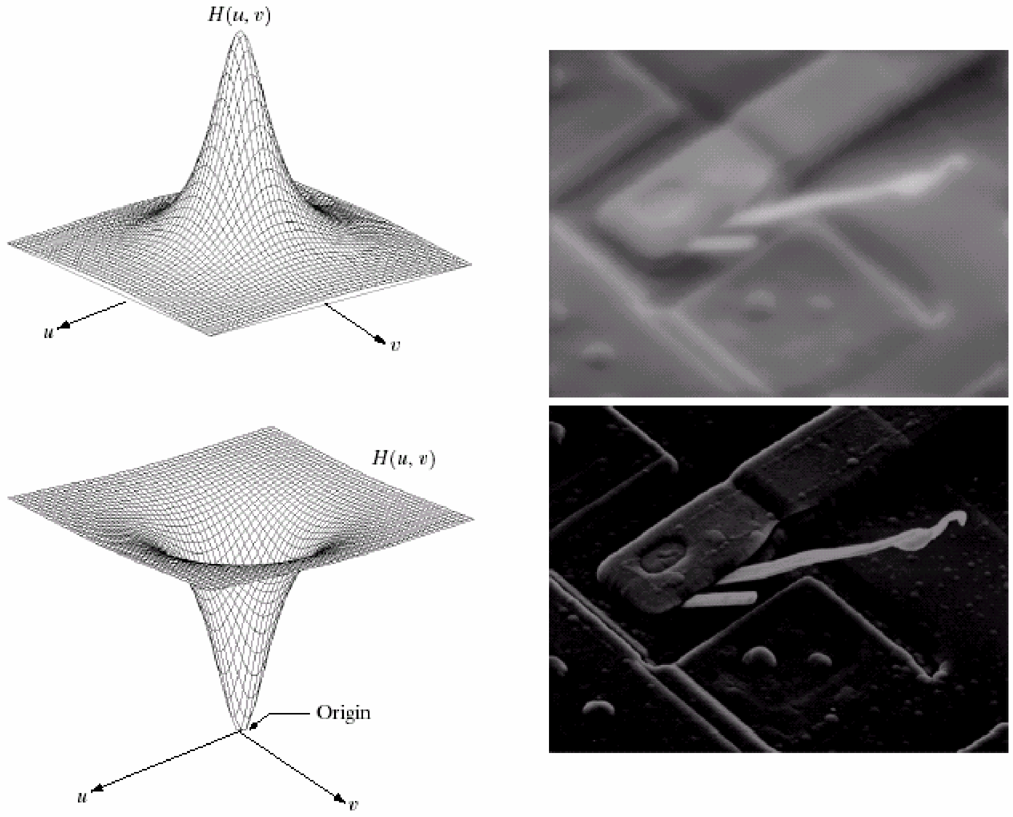


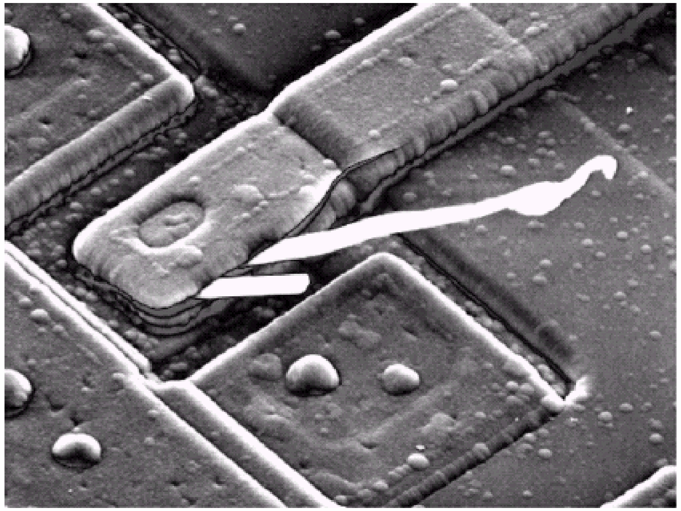
FIGURE 4.5 Basic steps for filtering in the frequency domain.



a b
c d

FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

FIGURE 4.8 Result of highpass filtering the image in Fig. 4.4(a) with the filter in Fig. 4.7(c), modified by adding a constant of one-half the filter height to the filter function. Compare with Fig. 4.4(a).



- The One-Dimensional Fourier Transform and its Inverse
- The Two-Dimensional Fourier Transform and its Inverse
- Filtering in the Frequency Domain
- Correspondence between Filtering in the Spatial and Frequency Domains

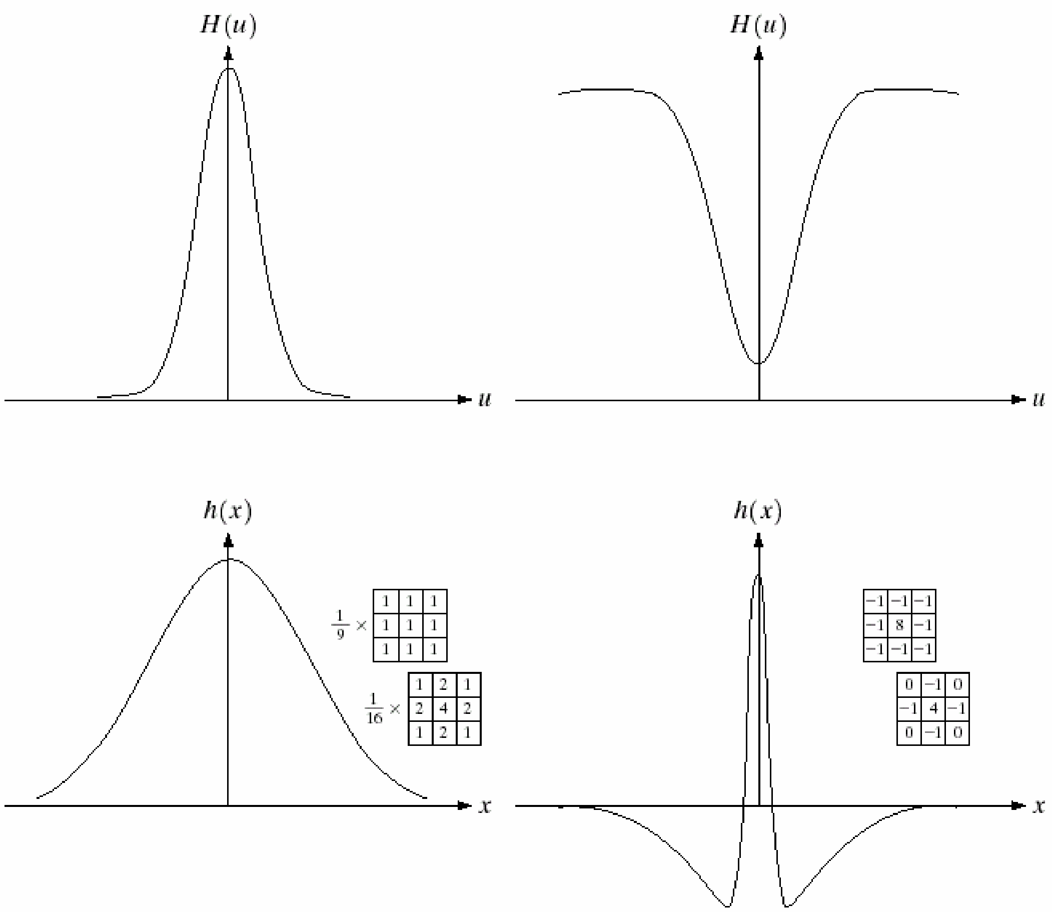
- The discrete convolution of two functions f(x,y) and h(x,y) of size M*N is denoted by f(x,y)*h(x,y) and is defined by the expression

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

- Letting F(u,v) and H(u,v) denote the Fourier transforms of f(x,y) and h(x,y), the following result holds:

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

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



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

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

a b
c d

FIGURE 4.9 (a) Gaussian frequency domain lowpass filter. (b) Gaussian frequency domain highpass filter. (c) Corresponding lowpass spatial filter. (d) Corresponding highpass spatial filter. The masks shown are used in Chapter 3 for lowpass and highpass filtering.

- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

- Basic model for filter in the frequency domain is given by the following equation

$$G(u,v) = H(u,v)F(u,v)$$

Where F(u,v) is the Fourier transform of the image to be smoothed.

- The objective is to select a filter transfer function H(u,v) that yields G(u,v) by attenuating the high-frequency components of F(u,v).

- Ideal Lowpass Filter
- Butterworth Lowpass Filters
- Gaussian Lowpass Filter
- Additional Examples of Lowpass Filtering

- The transfer function of a two-dimensional(2-D) ideal lowpass filter(ILPF) is:

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

- The distance from any point (u,v) to the center of the Fourier transform is given by

$$D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2}.$$

Ideal Lowpass Filters

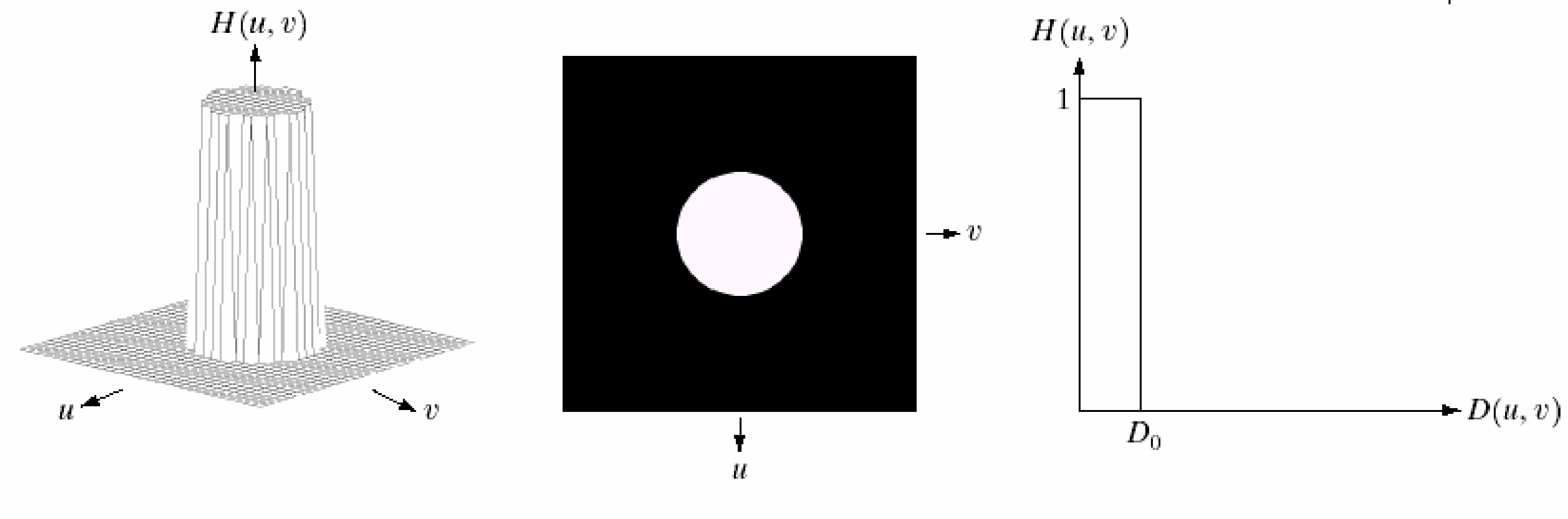


FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

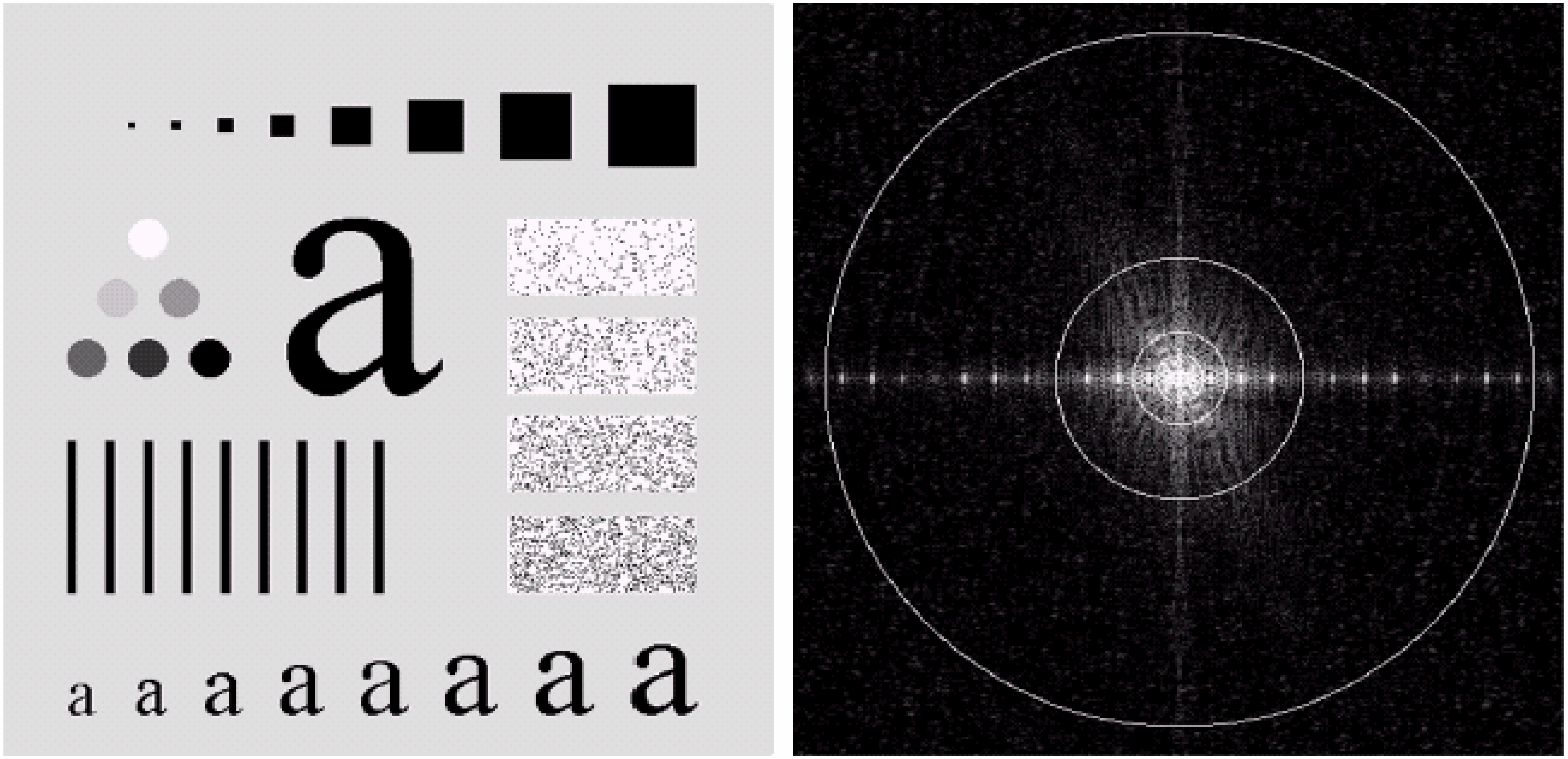
Ideal Lowpass Filters

- **Image power**
  - Image power (P_T) is obtained by summing the component of the power spectrum at each point *(u,v)*, for $u = 0,1,2,\ldots M\text{-}1$ and $v = 0,1,2,\ldots N\text{-}1$

$$\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} P(u,v)$$

  - A circle of radius r with origin at the center of the frequency rectangle encloses α = 100 [$\Sigma\Sigma P(u,v)/P_T$]

Ideal Lowpass Filters



a b

FIGURE 4.11 (a) An image of size 500 × 500 pixels and (b) its Fourier spectrum. The superimposed circles have radii values of 5, 15, 30, 80, and 230, which enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power, respectively.

Ideal Lowpass Filters

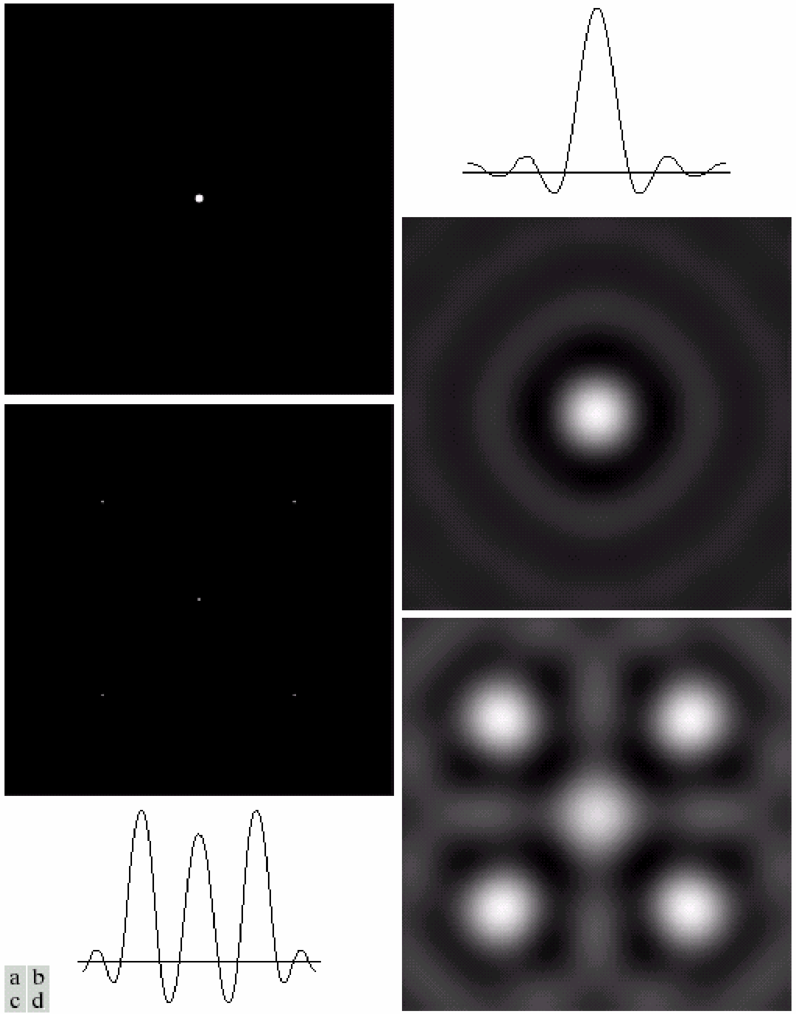


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

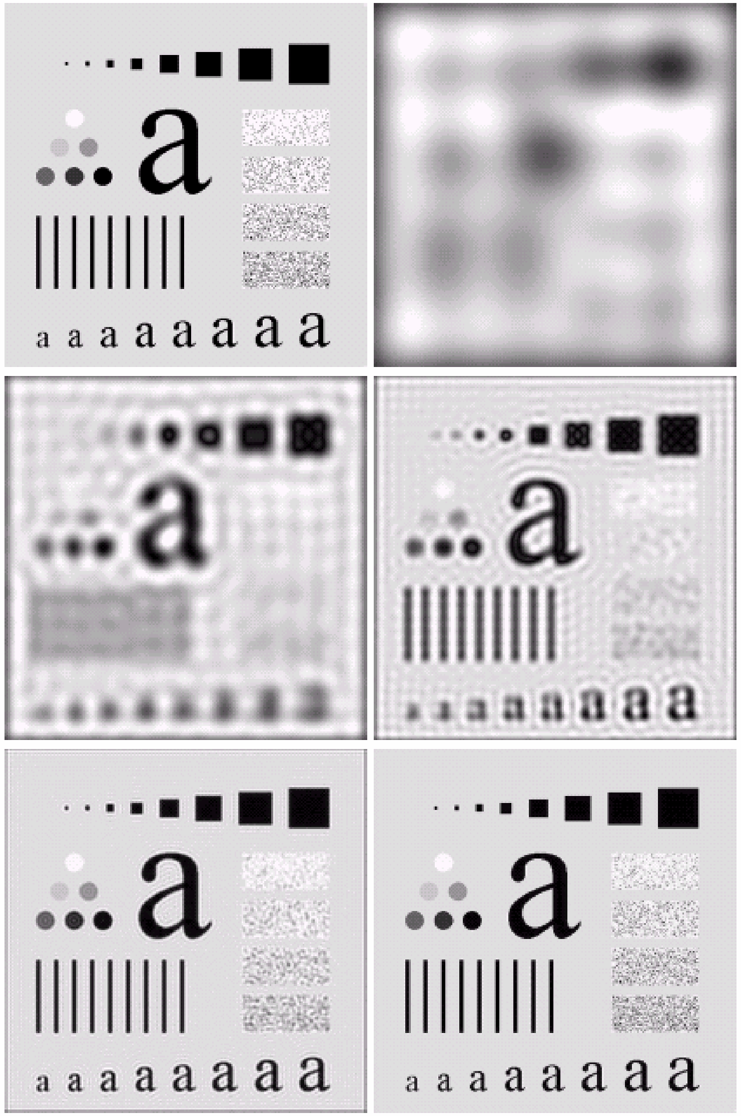


FIGURE 4.12 (a) Original image. (b)–(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). The power removed by these filters was 8, 5.4, 3.6, 2, and 0.5% of the total, respectively.

Ideal Lowpass Filters

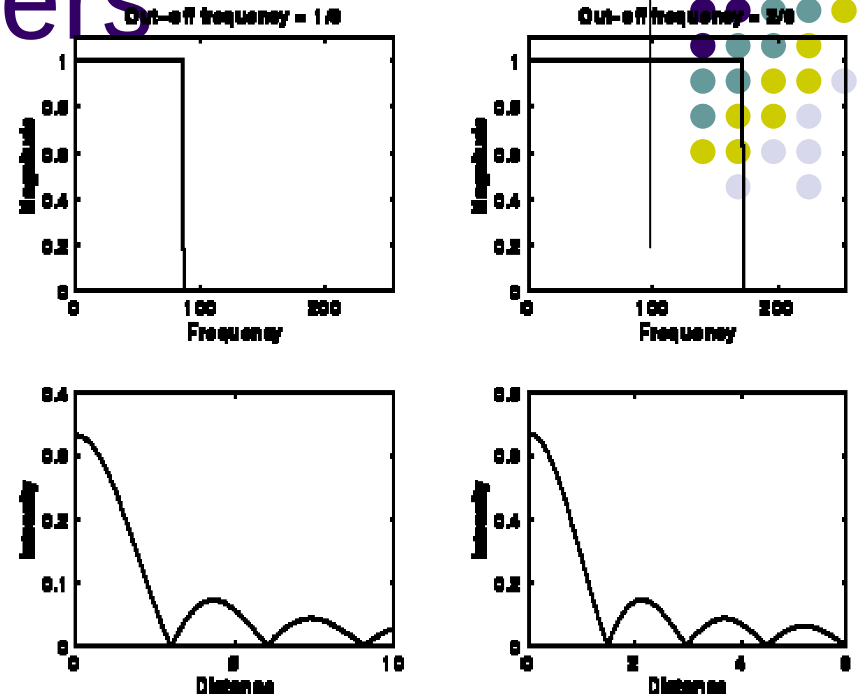
- As the filter radius increases, less and less power is removed, resulting in less severe blurring.
- Fig 4.12(c) through (e) are characterized by “ringing” which becomes finer in texture as the amount of high frequency content removed decreases.

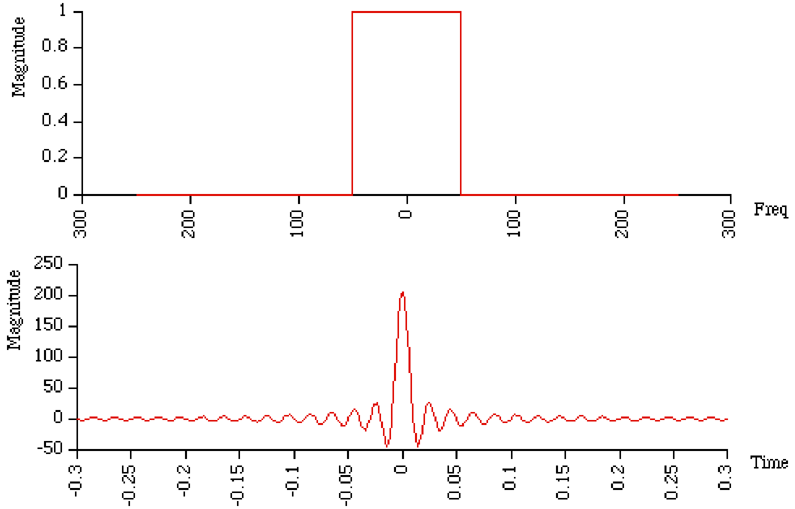
Ideal Lowpass Filters

- Ringing effect occurs along the edges of the filtered spatial domain image (illustrated in a Figure).
- Next slide figure shows the shape of the one-dimensional filter in both the frequency and spatial domains for two different values of D_0.
- We obtain the shape of the two-dimensional filter by rotating these functions about the *y-axis*.

Ideal Lowpass Filters

- Multiplication in the Fourier domain corresponds to a convolution in the spatial domain.
- Due to the multiple peaks of the ideal filter in the spatial domain, the filtered image produces ringing along intensity edges in





- Ideal Lowpass Filter
- Butterworth Lowpass Filters
- Gaussian Lowpass Filter
- Additional Examples of Lowpass Filtering

Butterworth Lowpass Filters

- Ideal filtering simply cuts off the Fourier transform. It is easy to implement, however, it has the disadvantage of introducing unwanted artifacts (ringing) into the result.
- One way of avoiding these artifacts is to use a filter matrix a circle with a cutoff that is less sharp.

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

Butterworth Lowpass Filters

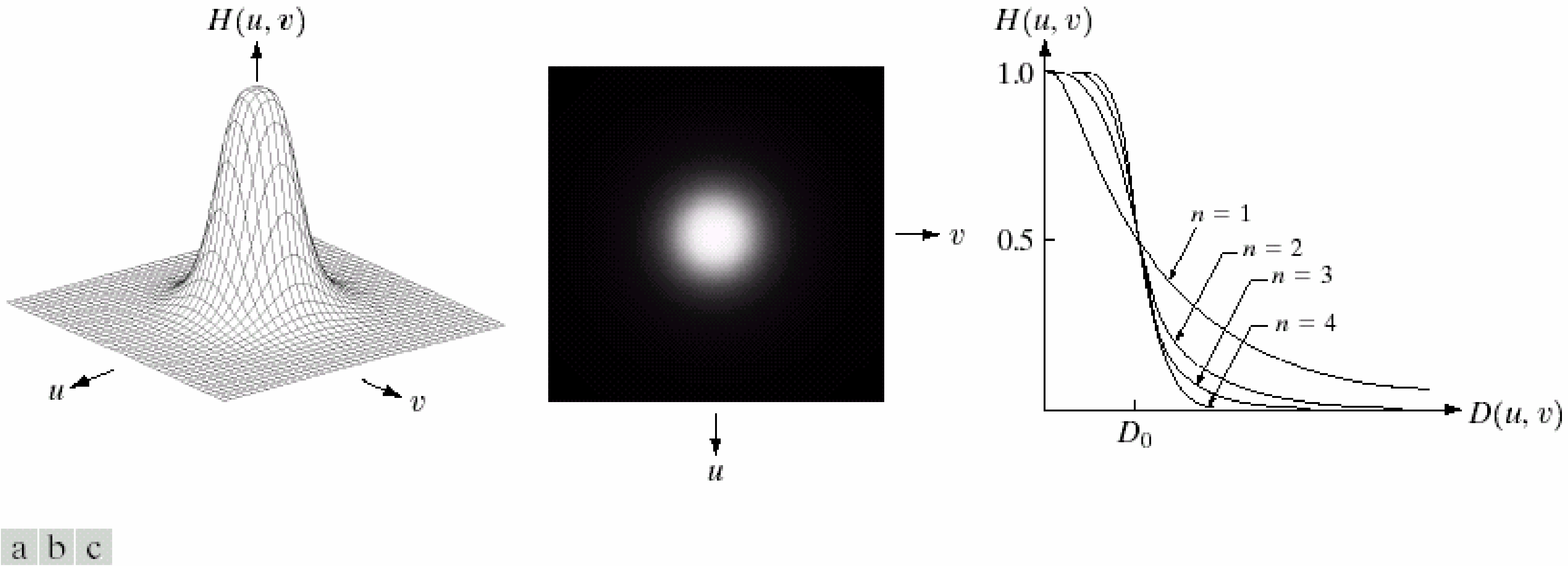


FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

Lowpass Filters

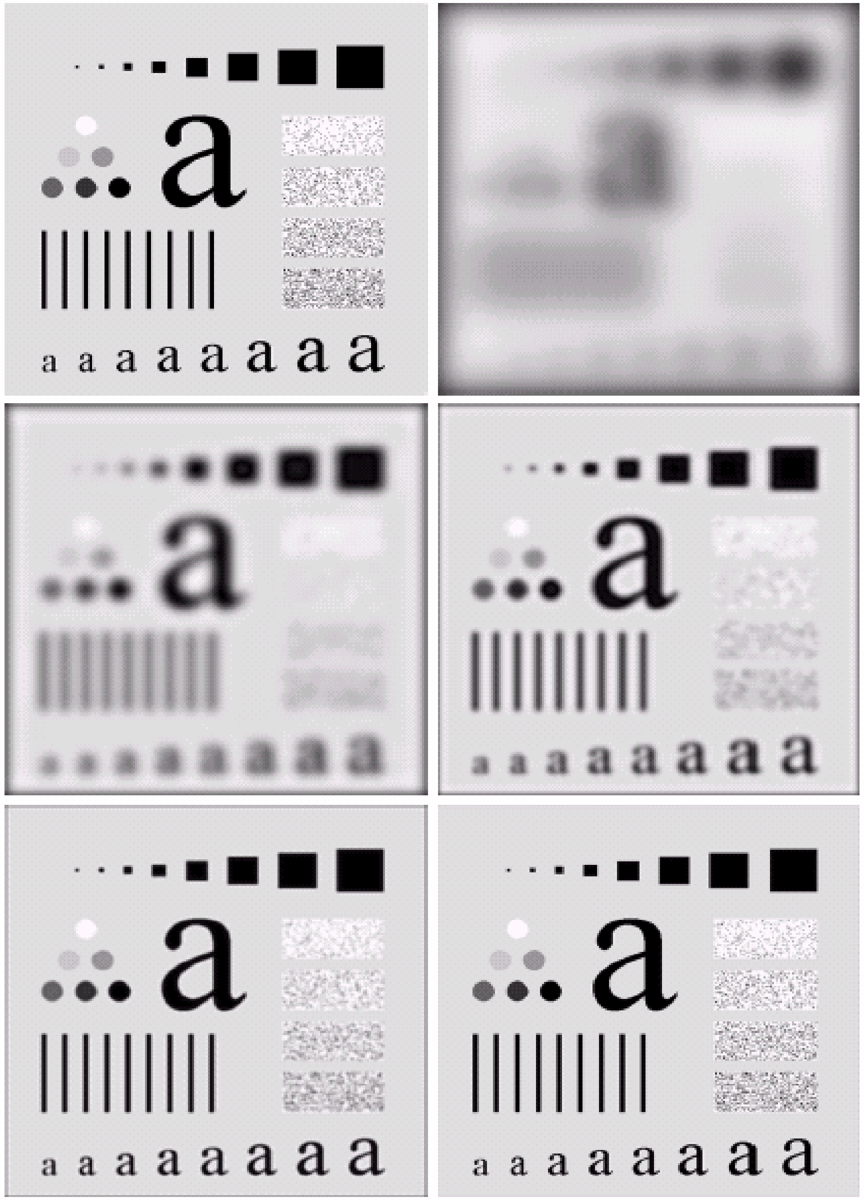
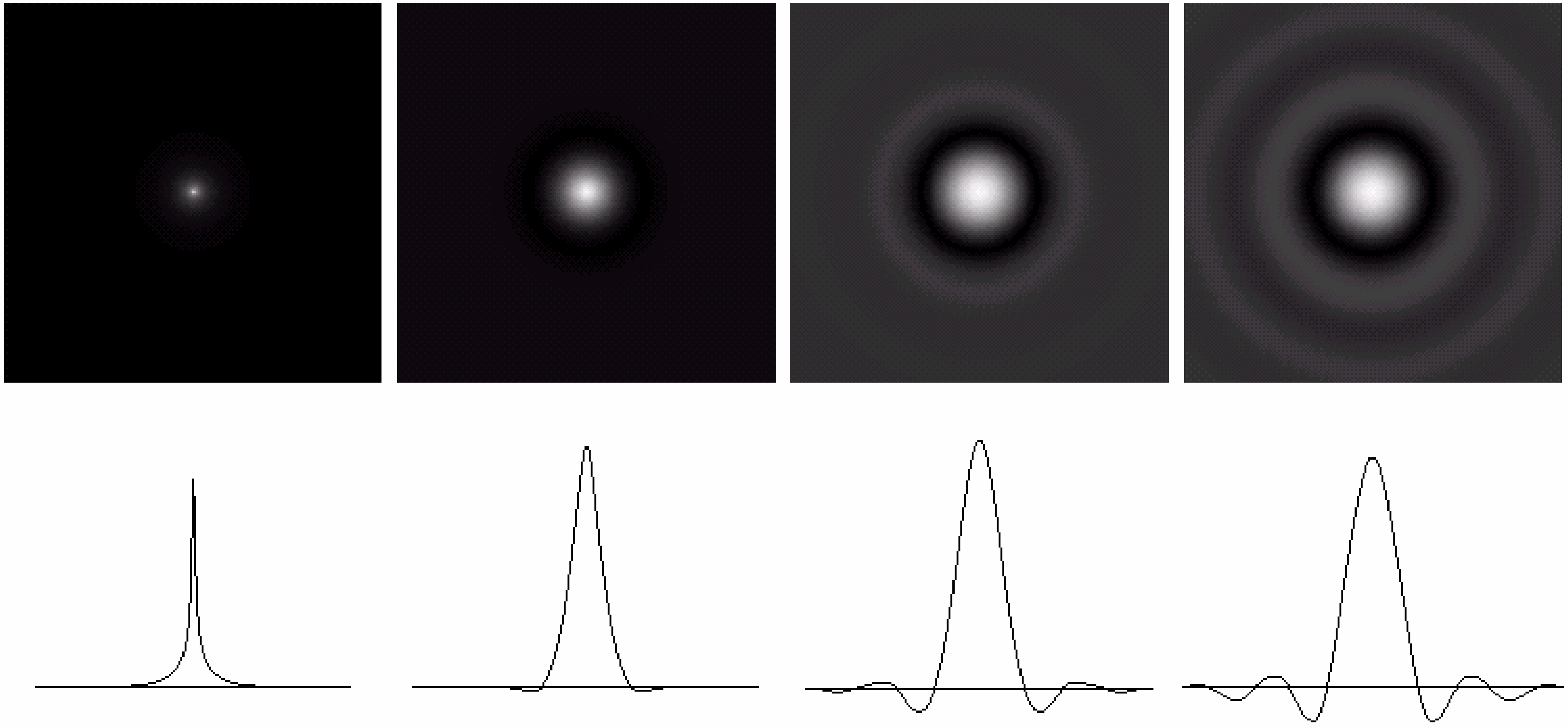


FIGURE 4.15 (a) Original image. (b)–(f) Results of filtering with BLPFs of order 2, with cutoff frequencies at radii of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Fig. 4.12.

Butterworth Lowpass Filters



a b c d

FIGURE 4.16 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding gray-level profiles through the center of the filters (all filters have a cutoff frequency of 5). Note that ringing increases as a function of filter order.

- Ideal Lowpass Filter
- Butterworth Lowpass Filters
- Gaussian Lowpass Filter
- Additional Examples of Lowpass Filtering

Gaussian Lowpass Filters

- The form of these filters in two dimensions is given by

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

$$\downarrow$$

$$H(u,v) = \exp(\frac{-D^2(u,v)}{2D^2{}_0})$$

- *D(u,v)* is the distance from the origin of the Fourier transform, which we assume has been shifted to the center of the frequency rectangle

Gaussian Lowpass Filters

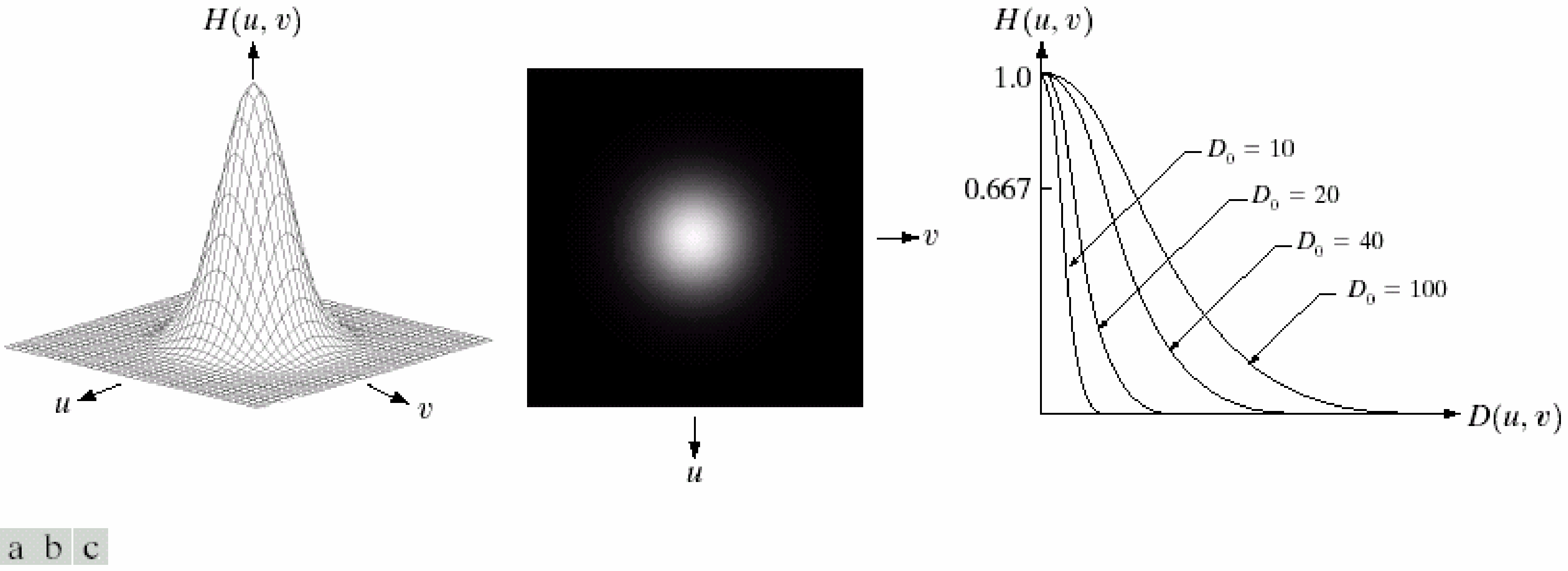


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

Lowpass Filters

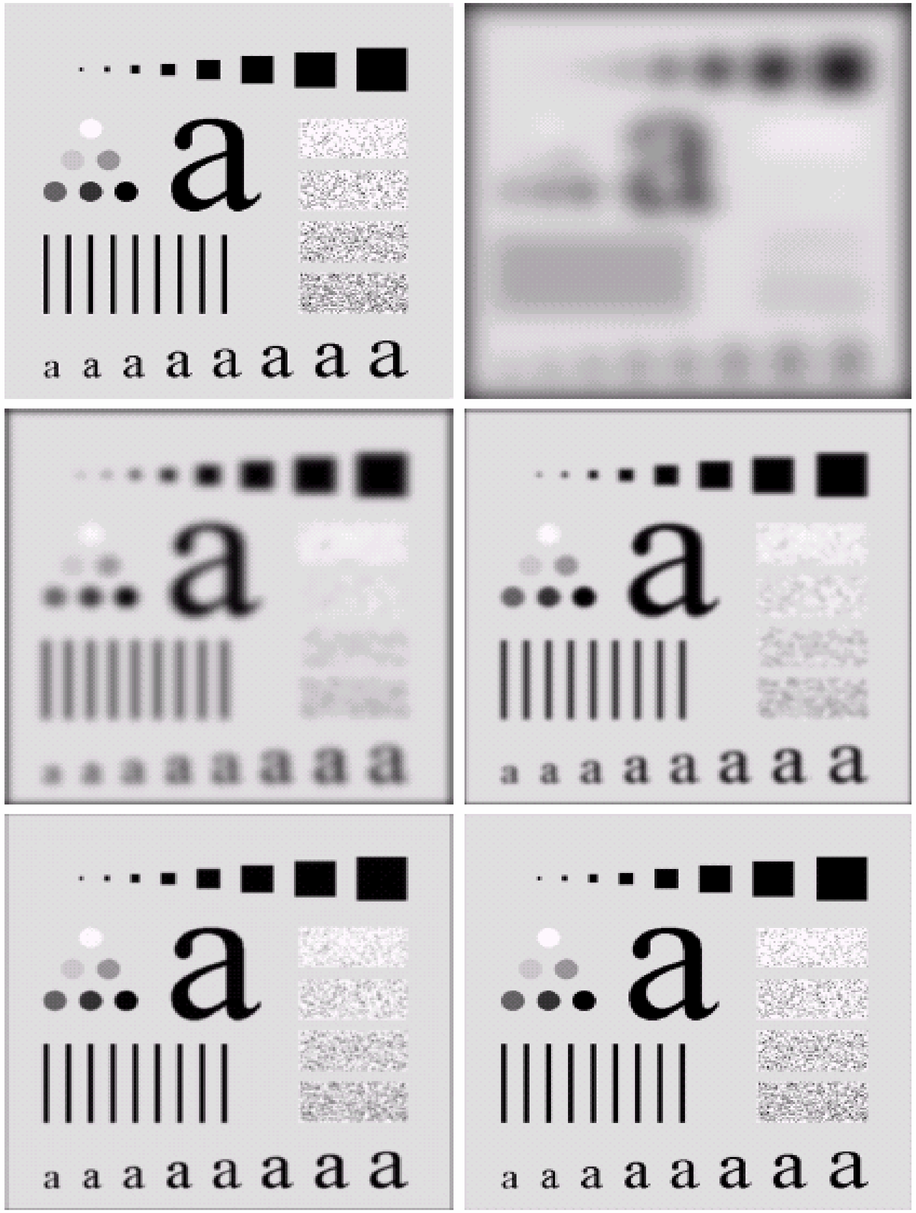


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

a b
c d
e f

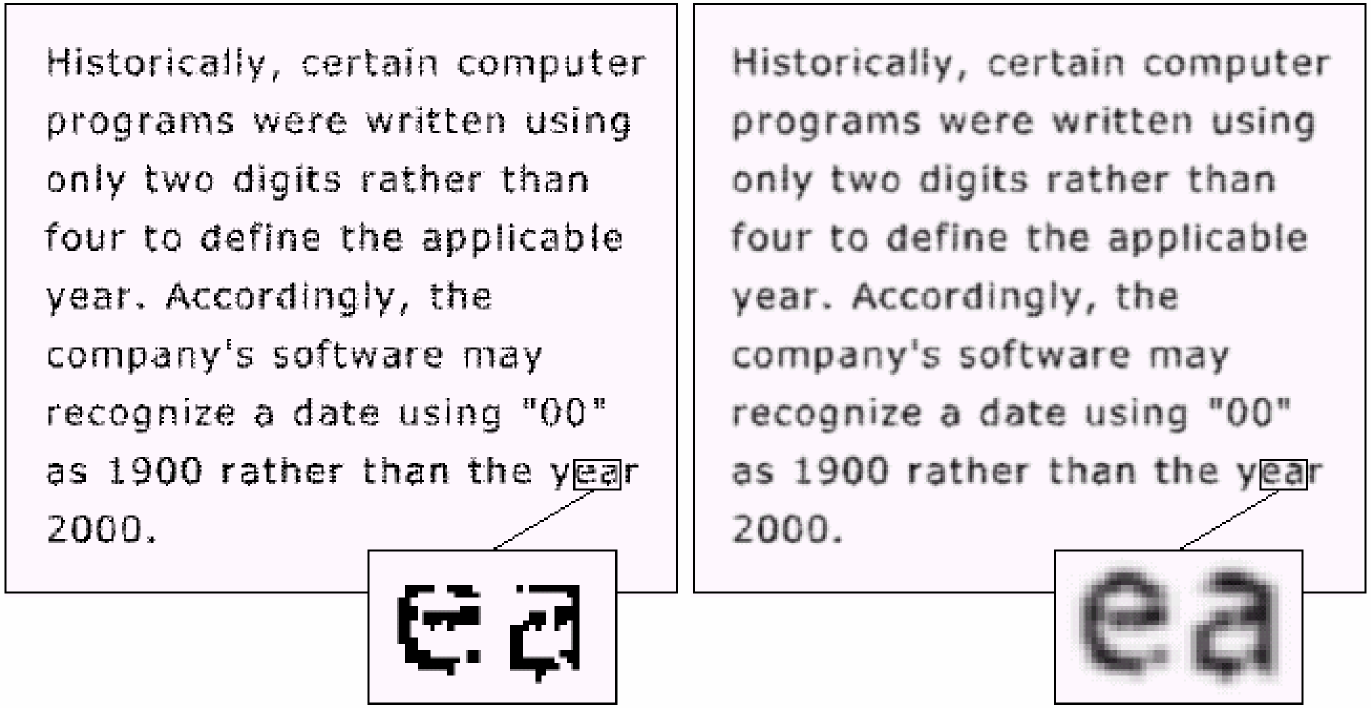
Lowpass Filtering

- Fig. 4.19 shows a sample of text of poor resolution that may be occurred from fax transmission, duplicated material, and historical records.
- Fig 4.19(a) shows characters in a document have distorted shapes. Many characters are broken.
- Fig 4.19(b) shows “repaired” characters by this simple process using a Gaussian lowpass filter with $D_0 = 80$

Gaussian Lowpass Filters

a b

FIGURE 4.19 (a) Sample text of poor resolution (note broken characters in magnified view). (b) Result of filtering with a GLPF (broken character segments were joined).



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.

- Ideal Lowpass Filter
- Butterworth Lowpass Filters
- Gaussian Lowpass Filter
- Additional Examples of Lowpass Filtering

Additional Examples of Lowpass Filtering

- Fig. 4.20 shows an application of lowpass filtering to produce a smoother, softer-looking result from a sharp original. The smoothed images look quite soft and pleasing
- Fig. 4.21 shows images with prominent scan line.
- Lowpass filtering a crude but simple way to reduce the effect of these lines.
- Fig. 4.21(b) shows an image with $D_0 = 30$
- Fig. 4.21(c) shows an image with $D_0 = 10$

Gaussian Lowpass Filters

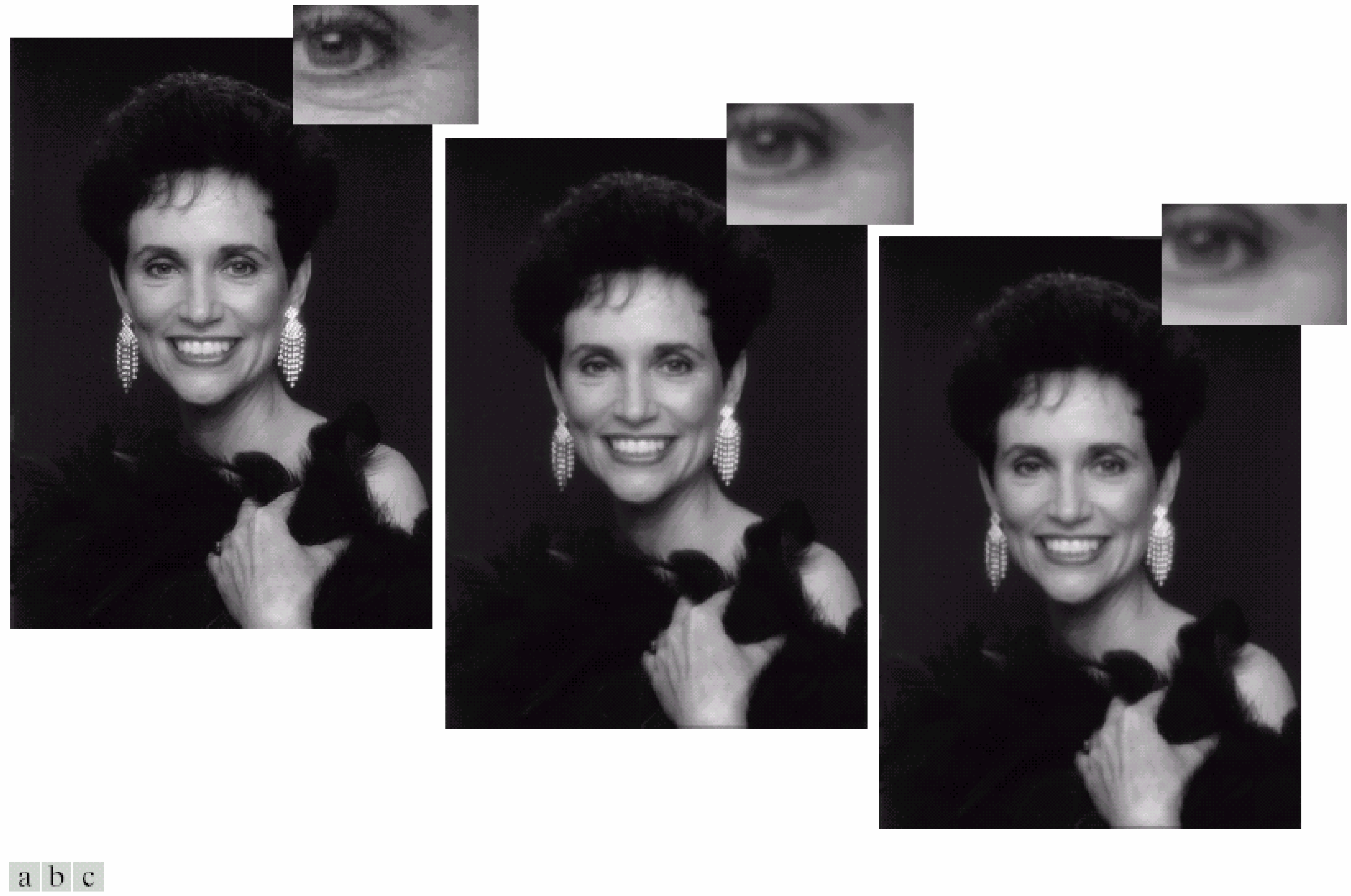


FIGURE 4.20 (a) Original image (1028 × 732 pixels). (b) Result of filtering with a GLPF with $D_0 = 100$. (c) Result of filtering with a GLPF with $D_0 = 80$. Note reduction in skin fine lines in the magnified sections of (b) and (c).

Gaussian Lowpass Filters

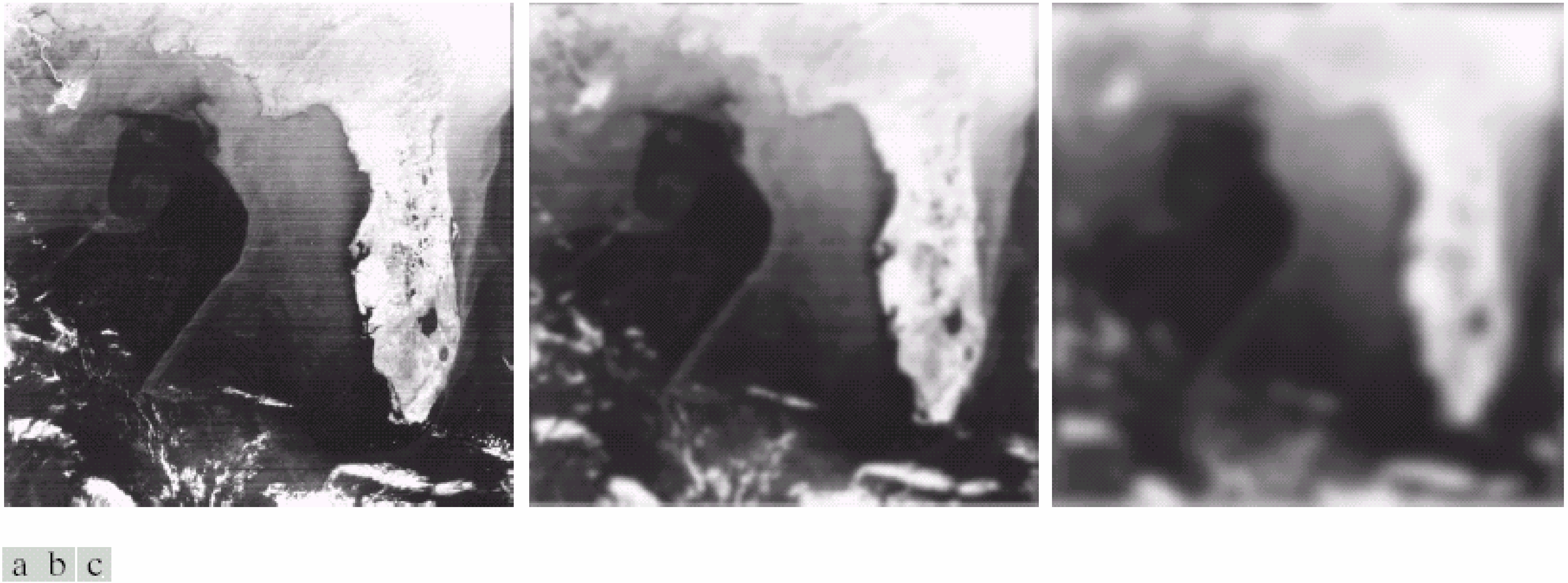


FIGURE 4.21 (a) Image showing prominent scan lines. (b) Result of using a GLPF with $D_0 = 30$. (c) Result of using a GLPF with $D_0 = 10$. (Original image courtesy of NOAA.)

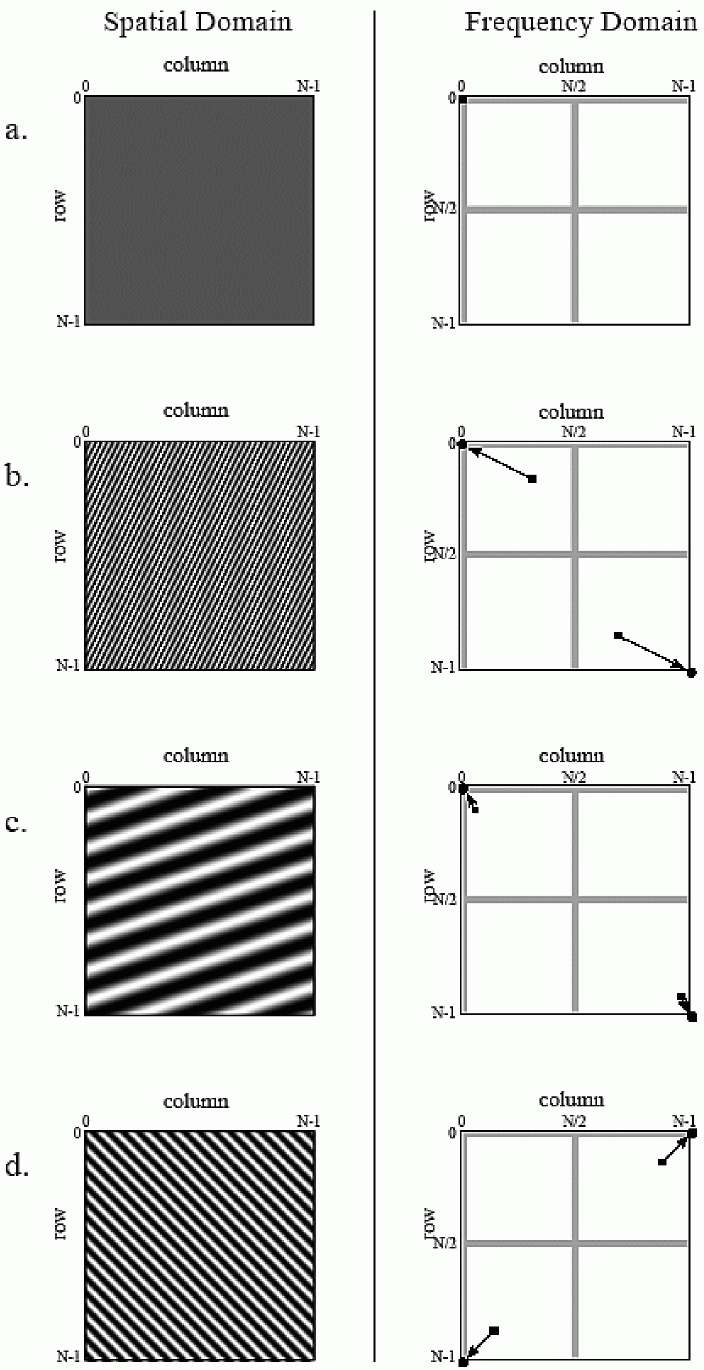


FIGURE 24-10
Two-dimensional sinusoids. Image sine and cosine waves have both a *frequency* and a *direction*. Four examples are shown here. These spectra are displayed with the low-frequencies at the corners. The circles in these spectra show the location of zero frequency.

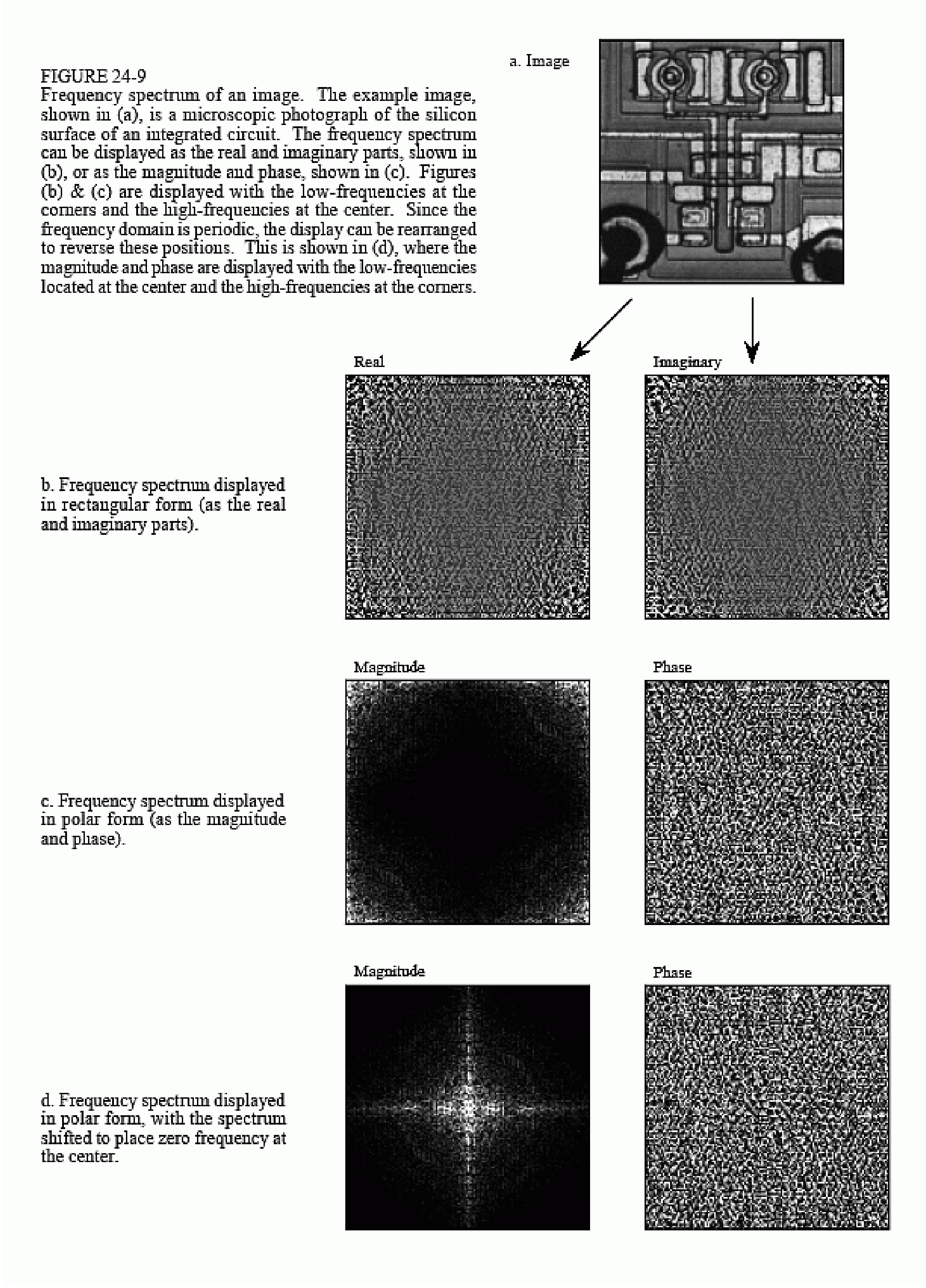


FIGURE 24-9
Frequency spectrum of an image. The example image, shown in (a), is a microscopic photograph of the silicon surface of an integrated circuit. The frequency spectrum can be displayed as the real and imaginary parts, shown in (b), or as the magnitude and phase, shown in (c). Figures (b) & (c) are displayed with the low-frequencies at the corners and the high-frequencies at the center. Since the frequency domain is periodic, the display can be rearranged to reverse these positions. This is shown in (d), where the magnitude and phase are displayed with the low-frequencies located at the center and the high-frequencies at the corners.

b. Frequency spectrum displayed in rectangular form (as the real and imaginary parts).

c. Frequency spectrum displayed in polar form (as the magnitude and phase).

d. Frequency spectrum displayed in polar form, with the spectrum shifted to place zero frequency at the center.

- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

Sharpening Frequency Domain Filters

- Image sharpening can be achieved by a highpass filtering process, which attenuates the low-frequency components without disturbing high-frequency information.
- Zero-phase-shift filters: radially symmetric and completely specified by a cross section.

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

Sharpening Frequency Domain Filters

- Fig. 4.22 shows typical 3-D plots image representations and cross sections for these filters (IHPF, BHPF, GHPF).

- Fig. 4.23 illustrates what these filters look like in the spatial domain. A spatial representation of a frequency domain filter is obtained by (1)multiplying $H(u,v)$ by $(-1)^{u+v}$ for centering (2)computing the inverse DFT (3) multiplying the real part of the inverse DFT by $(-1)^{x+y}$

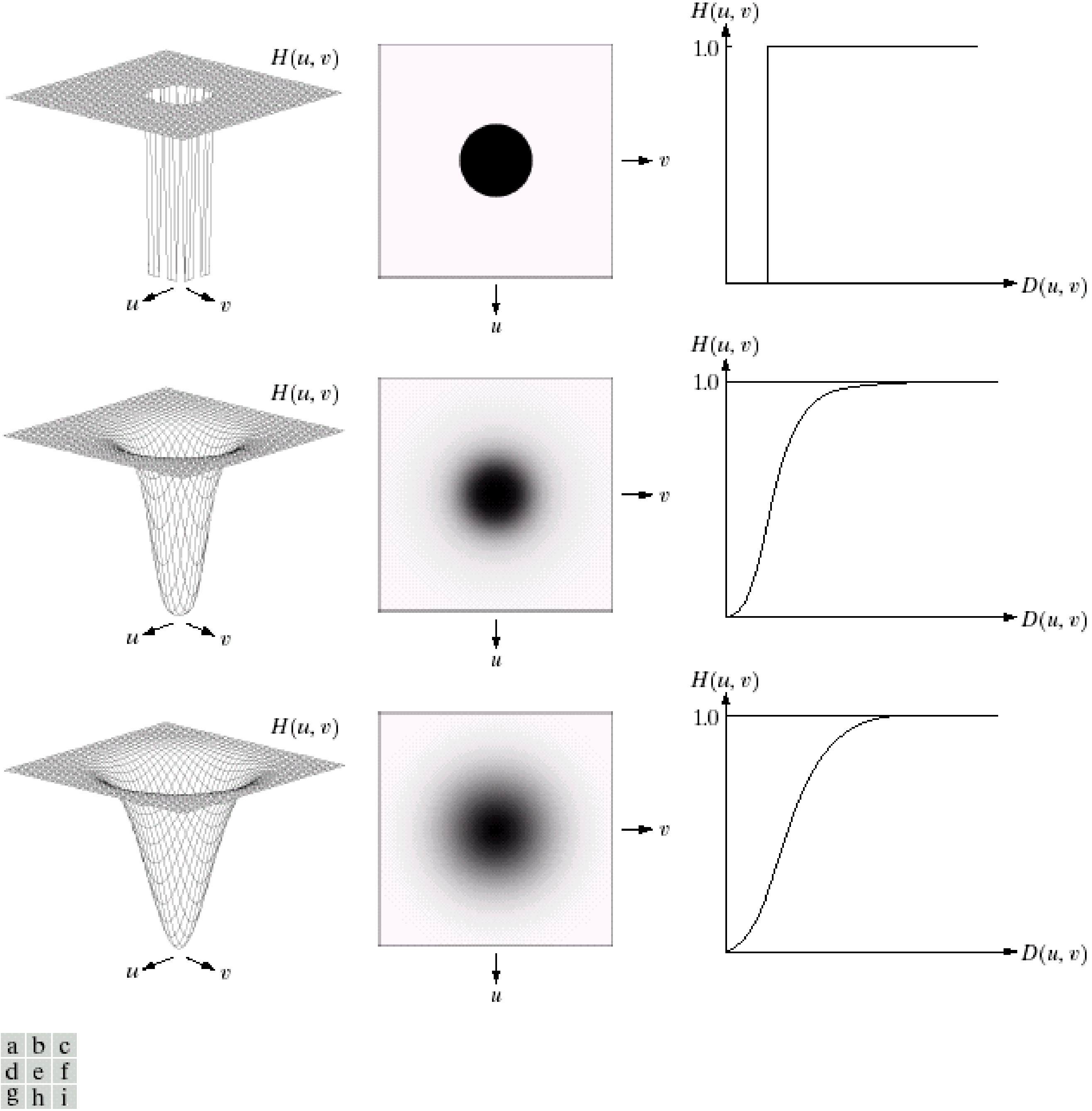


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

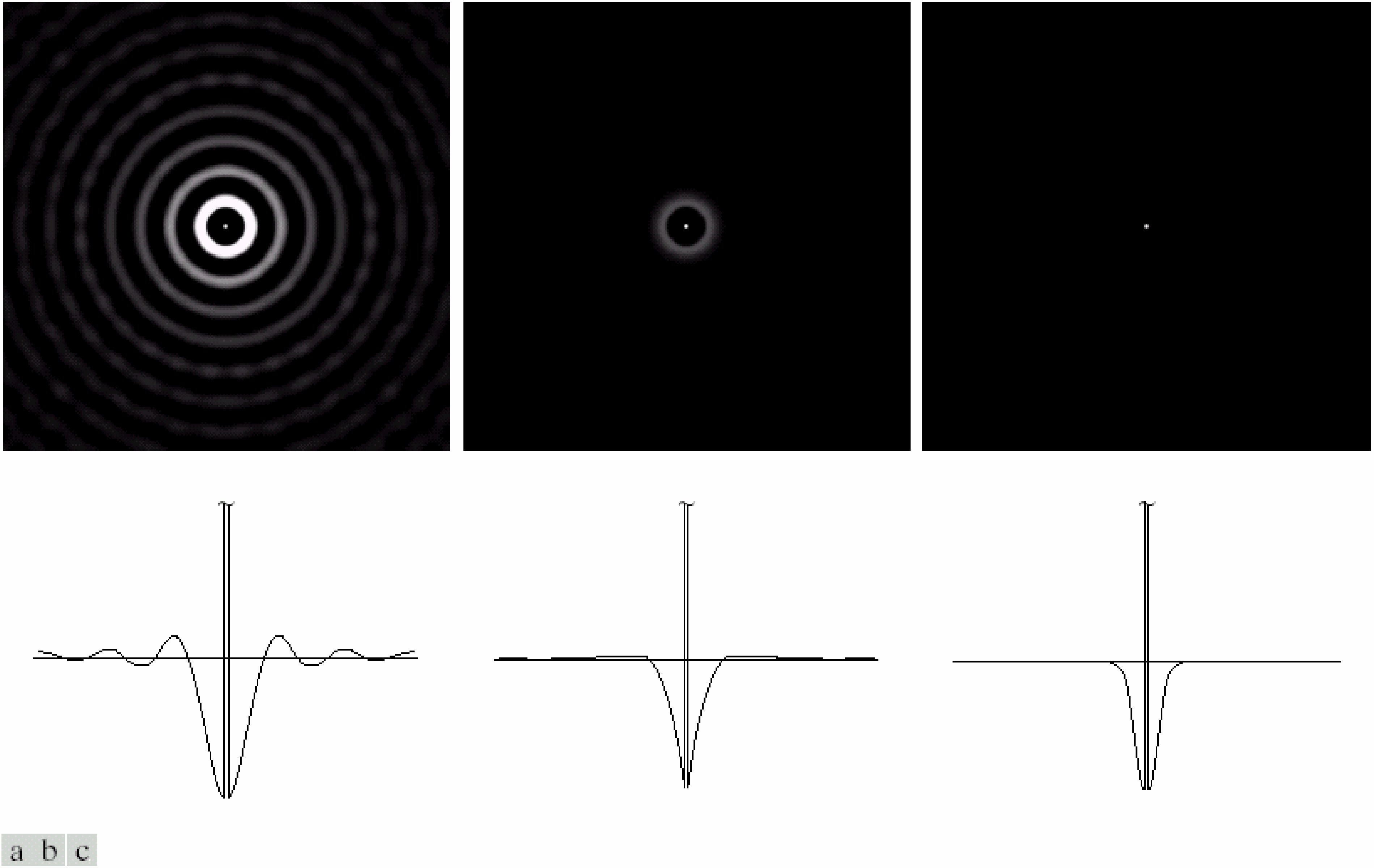


FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

Sharpening Frequency Domain Filters

- Ideal Highpass Filters
- Butterworth Highpass Filters
- Gaussian Highpass Filters
- The Laplacian in the Frequency Domain
- Unsharp Masking, High-Boost Filtering, and High-Frequency Emphasis Filtering

Ideal Highpass Filters

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

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

- D_0 is the cutoff distance measured.
- This filter is the opposite of the ideal lowpass filter.

Ideal Highpass Filters

- Fig. 4.24(a) is so severe that it produced distorted, thickened object boundaries. Edges on the top three circles do not show well.

- The result for D_0 = 80 is more of what a high pass-filtered image should look like. The edges are much cleaner and less distorted, and the smaller objects have been filtered properly.

Ideal Highpass Filters

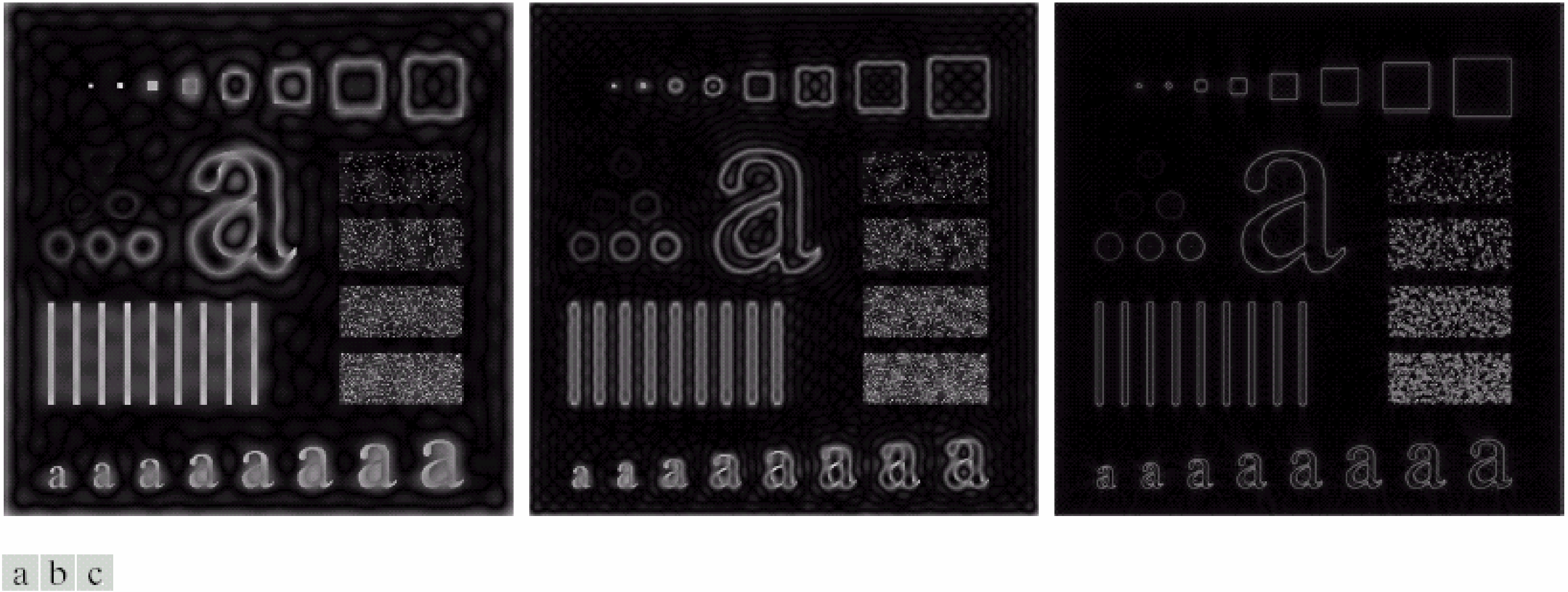


FIGURE 4.24 Results of ideal highpass filtering the image in Fig. 4.11(a) with $D_0 = 15$, 30, and 80, respectively. Problems with ringing are quite evident in (a) and (b).

Sharpening Frequency Domain Filters

- Ideal Highpass Filters
- Butterworth Highpass Filters
- Gaussian Highpass Filters
- The Laplacian in the Frequency Domain
- Unsharp Masking, High-Boost Filtering, and High-Frequency Emphasis Filtering

Butterworth Highpass Filters

- The transfer function of the Butterworth highpass filter (BHPF) of order n and will cutoff frequency locus at distance D_0 from the origin is given by

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

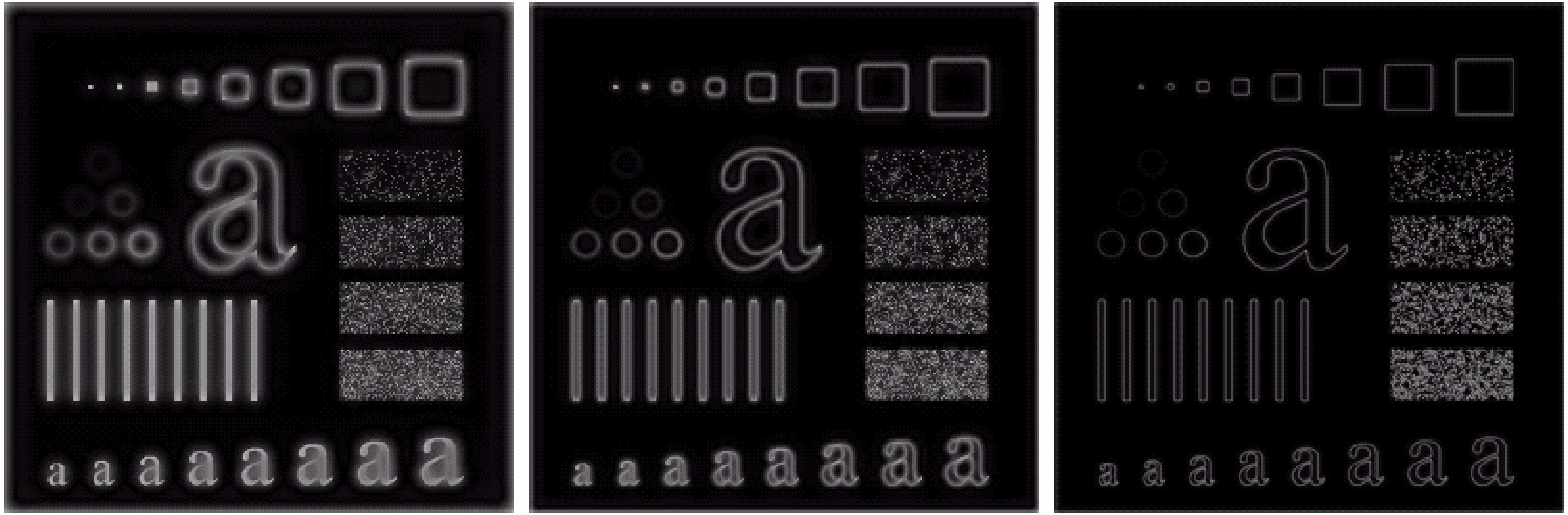
- High-frequency emphasis: Adding a constant to a highpass filter to preserve the low-frequency components.

Butterworth Highpass Filters

- Fig. 4.25: The boundary is much less distorted than in Fig. 4.24, even for the smallest value of cut off frequency.

- Since the center spot sizes of the IHPF and the BHPF are similar, the performance of the two filters I terms of filtering the smaller objects is comparable. The transition into higher values of cutoff frequencies is much smoother with the BHPF

Butterworth Highpass Filters



a b c

FIGURE 4.25 Results of highpass filtering the image in Fig. 4.11(a) using a BHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. These results are much smoother than those obtained with an ILPF.

Sharpening Frequency Domain Filters

- Ideal Highpass Filters
- Butterworth Highpass Filters
- Gaussian Highpass Filters
- The Laplacian in the Frequency Domain
- Unsharp Masking, High-Boost Filtering, and High-Frequency Emphasis Filtering

Gaussian Highpass Filters

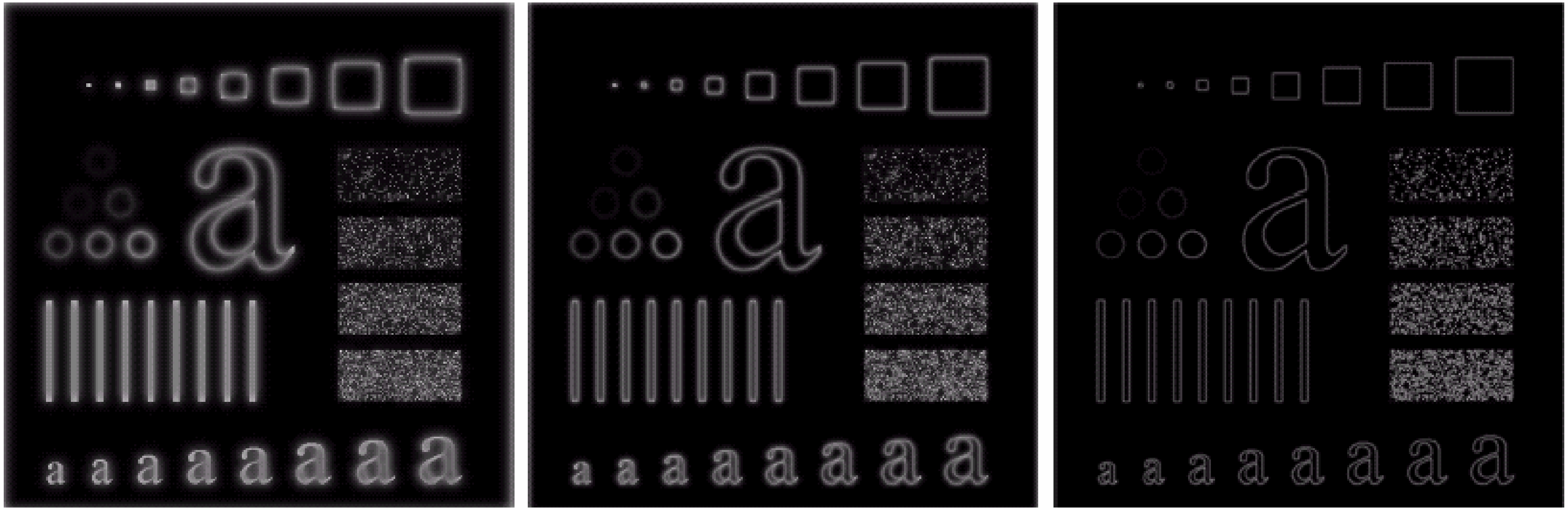
- The transfer function of the Gaussian Highpass Filters (GHPF) with cutoff frequency locus at distance D_0 from the origin is given by

$$H(u,v) = 1 - \exp\left(\frac{-D^2(u,v)}{2{D_0}^2}\right)$$

Gaussian Highpass Filters

- Fig. 4.26: As expected, the results obtained are smoother than with the previous two filters. Even the filtering of the smaller objects and thin bars cleaner with the Gaussian filter.

Gaussian Highpass Filters



a b c

FIGURE 4.26 Results of highpass filtering the image of Fig. 4.11(a) using a GHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. Compare with Figs. 4.24 and 4.25.

Sharpening Frequency Domain Filters

- Ideal Highpass Filters
- Butterworth Highpass Filters
- Gaussian Highpass Filters
- The Laplacian in the Frequency Domain
- Unsharp Masking, High-Boost Filtering, and High-Frequency Emphasis Filtering

Laplacian in the Frequency Domain

- It can be shown that:

$$\Im\left[\nabla^2 f(x,y)\right] = -(u^2+v^2)F(u,v)$$

- The Laplacian can be implemented in the frequency domain by using the filter (Shift to center)

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

$$= -[(u-M/2)^2+(v-N/2)^2)].$$

Laplacian in the Frequency Domain

- The laplacian-filtered image in the spatial domain is obtain by computing the inverse Fourier Transform of *H(u,v)F(u,v)*

$$\nabla^2 f(x,y) = \Im^{-1}\{-[(u-M/2)^2+(v-N/2)^2]F(u,v)\}.$$

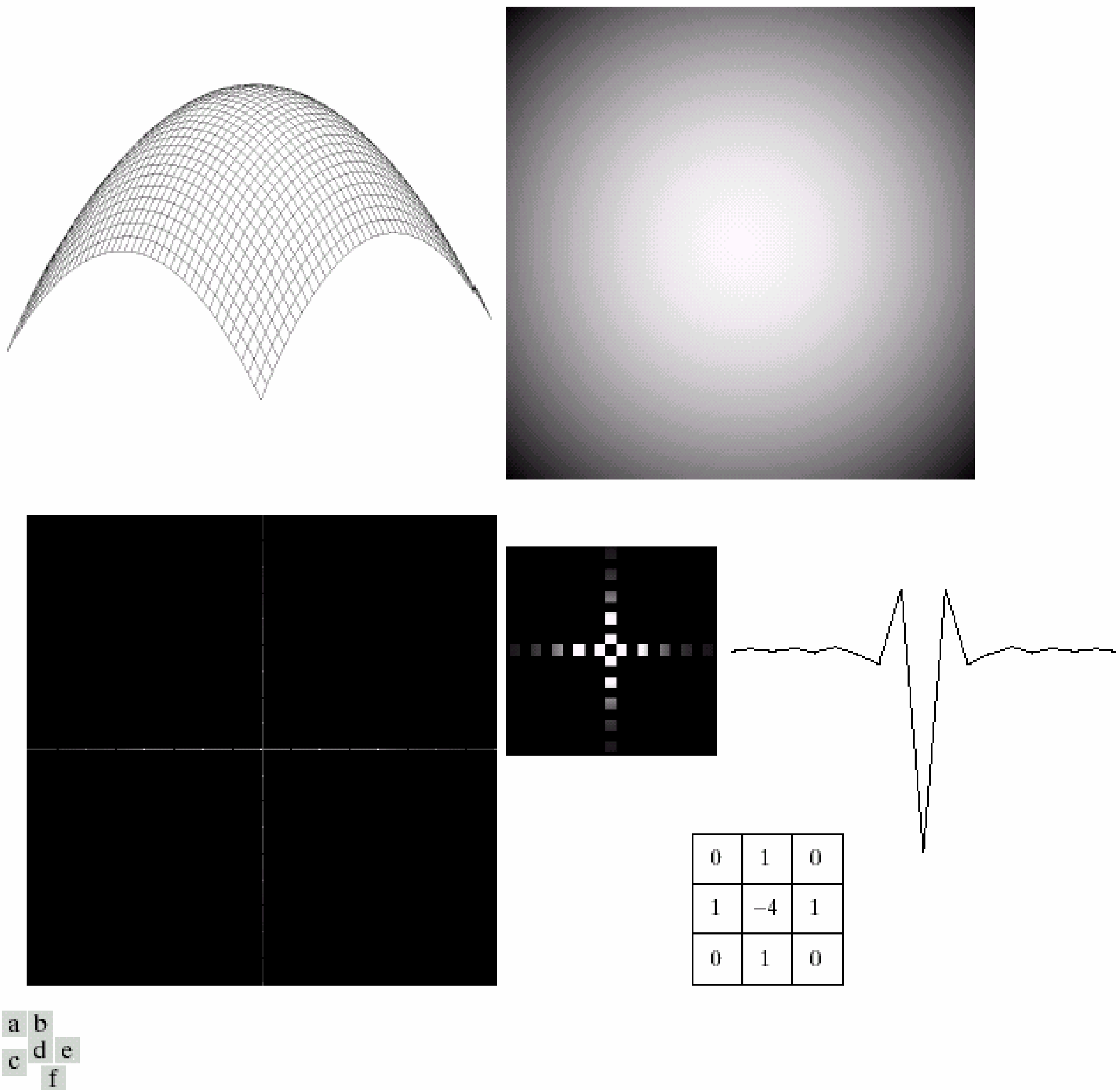
Laplacian in the Frequency Domain

- Fig. 4.27(a) is a 3-D perspective plot of

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

$$= -[(u - M/2)^2 + (v - N/2)^2)].$$

- The function is center at (*M/2*,*N/2*), and its value at the top of the dome is zero. All other values are negative.
- Fig. 4.27(b) shows *H(u,v)* as an image, also centered.
- Fig. 4.27(c) is the Laplacian in the spatial domain.



0	1	0
1	–4	1
0	1	0

FIGURE 4.27 (a) 3-D plot of Laplacian in the frequency domain. (b) Image representation of (a). (c) Laplacian in the spatial domain obtained from the inverse DFT of (b). (d) Zoomed section of the origin of (c). (e) Gray-level profile through the center of (d). (f) Laplacian mask used in Section 3.7.

Laplacian in the Frequency Domain

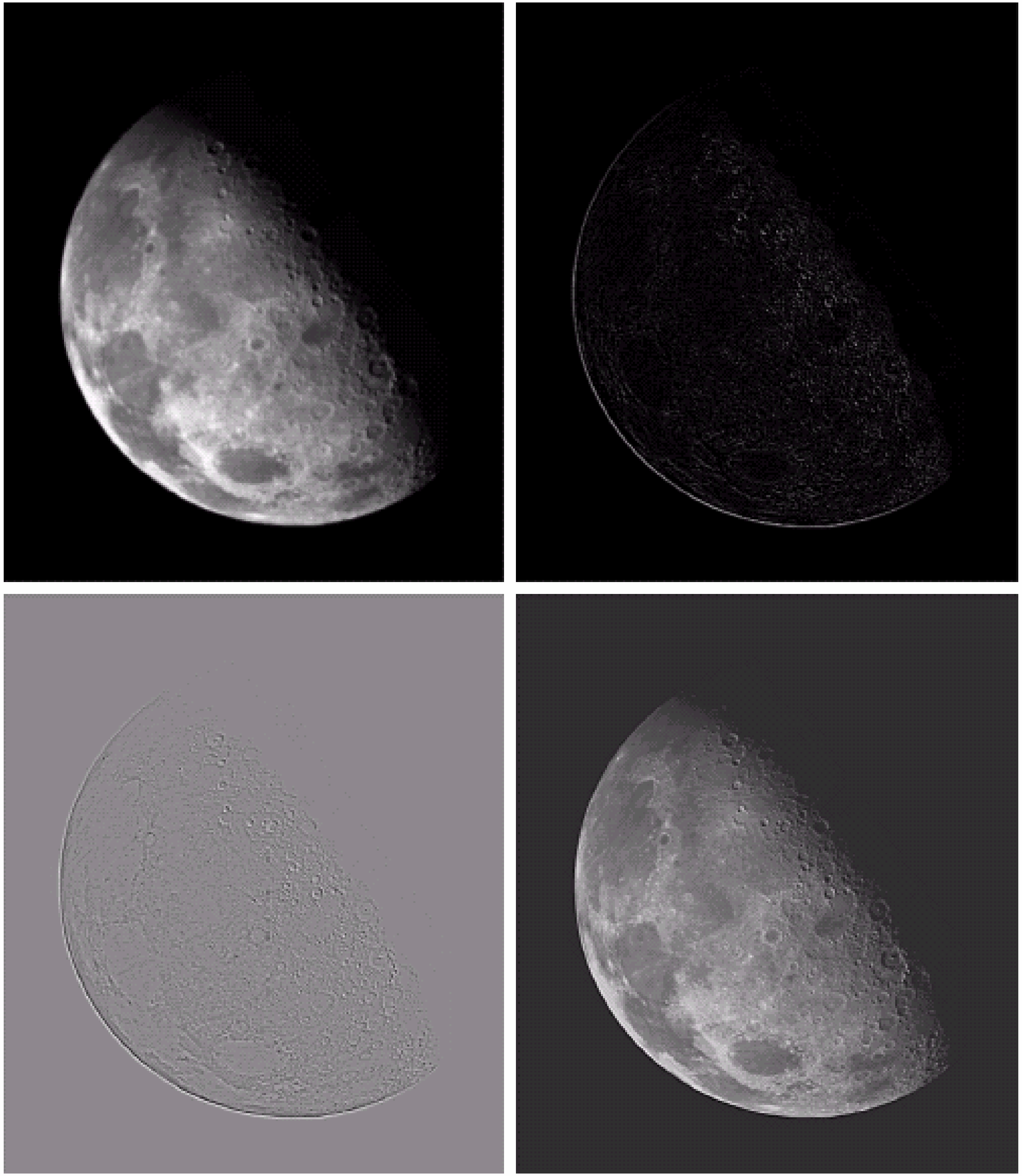
- Fig. 4.28(a) is the same image in as Fig. 3.40(a). Fig. 4.28(b) shows the result of filtering this image in the frequency domain using

$$\nabla^2 f(x,y) = \Im^{-1}\{-[(u-M/2)^2+(v-N/2)^2]F(u,v)\}.$$

- Fig. 4.28(c) show the scaled image (for display only)
- Fig. 4.28(d) should be compared with Fig. 3.40, which shows exactly the same sequence of steps but computed using only spatial domain techniques. The results are identical for all practical purposes.

a b
c d

FIGURE 4.28 (a) Image of the North Pole of the moon. (b) Laplacian filtered image. (c) Laplacian image scaled. (d) Image enhanced by using Eq. (4.4-12). (Original image courtesy of NASA.)



Sharpening Frequency Domain Filters

- Ideal Highpass Filters
- Butterworth Highpass Filters
- Gaussian Highpass Filters
- The Laplacian in the Frequency Domain
- Unsharp Masking, High-Boost Filtering, and High-Frequency Emphasis Filtering

Unsharp Masking, High-Boost Filtering

- Unsharp masking:

$$f_{hp}(x,y) = f(x,y) - f_{lp}(x,y)$$

- High boost filtering:

$$f_{hb}(x,y) = Af(x,y) - f_{lp}(x,y)$$

$$f_{hb}(x,y) = (A-1)f(x,y) + f_{hp}(x,y)$$

$$H_{hb}(u,v) = (A-1) + H_{hp}(u,v)$$

Unsharp Masking, High-Boost Filtering

- Fig. 4.29 (b) is a highpass filterd iamge.
- The image in Fig. 4.29(c) was obtained using

  $f_{hb}(x,y) = (A-1)f(x,y) + f_{hp}(x,y)$

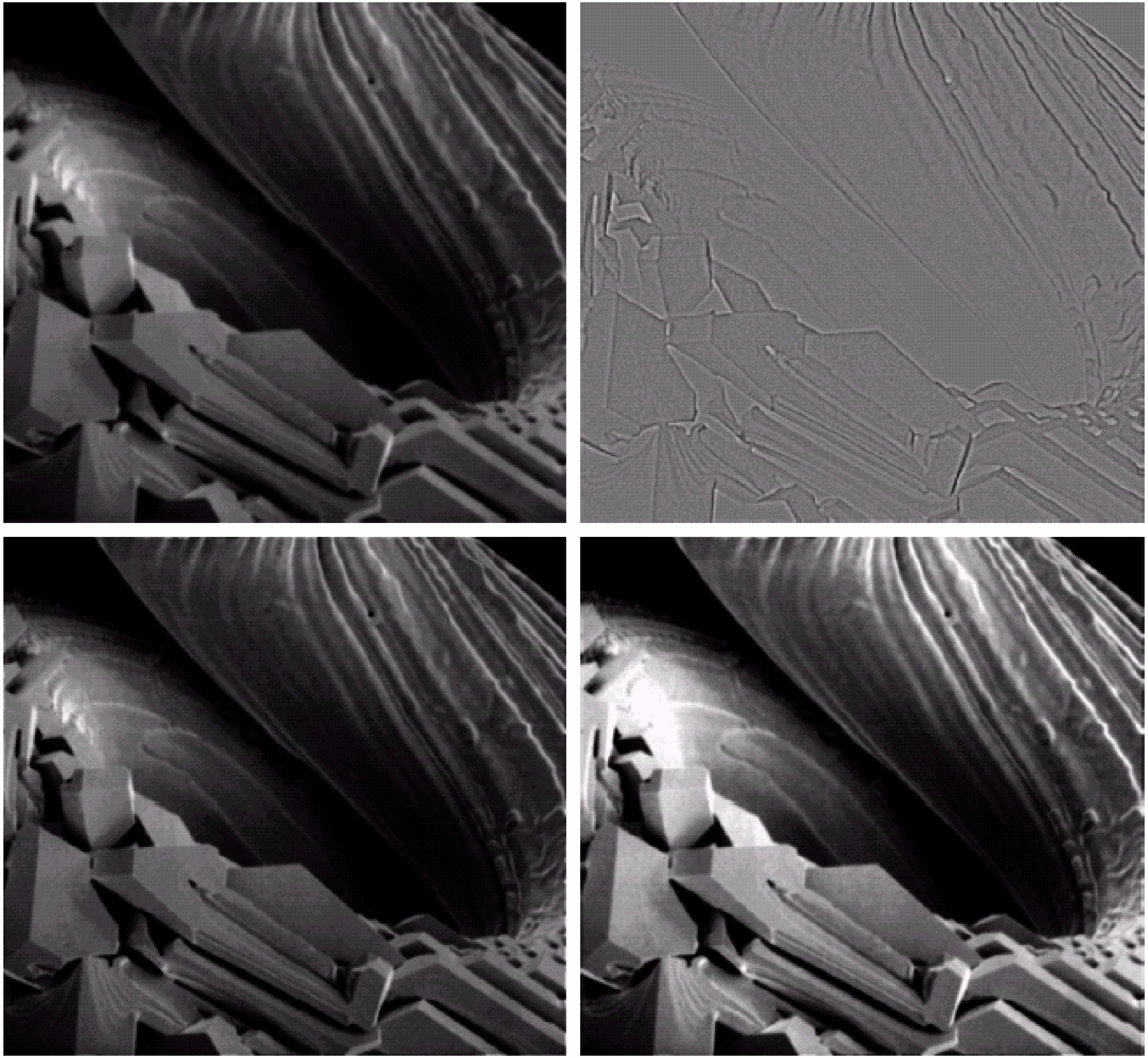
  with A = 2.

  This image is sharper but still too dark.
- Fig. 4.29(d) was obtained with A = 2.7, which in effect means that the input image was multiplied by 1.7 before the Laplacian was subtracted from it.

Unsharp Masking, High-Boost Filtering

- Fig. 4.29(d) is not as sharp as Fig. 3.43(d). The reason for this that a frequency domain representation of the Laplacian is closer to the mask that excludes the diagonal neighbors [Fig. 4.27(f)].

- It is known that a mask that includes the diagonal neighbors produces slightly sharper results. They do become evident for images with larger features.



a b
c d

FIGURE 4.29 Same as Fig. 3.43, but using frequency domain filtering. (a) Input image. (b) Laplacian of (a). (c) Image obtained using Eq. (4.4-17) with $A = 2$. (d) Same as (c), but with $A = 2.7$. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

High-Frequency Emphasis Filtering

- Sometimes it is advantageous to accentuate the contribution to enhancement made by the high-frequency component of an image.
- We multiply a high pass filter function by a constant and add an offset so that the zero frequency term is not eliminated by the filter.

$$H_{hfe}(u,v) = a+bH_{hp}(u,v),$$

where a>=0 and b>a.

[Typical a = [0.25,0.5] and b = [1.5,2.0]]

High-Frequency Emphasis Filtering

- Fig. 4.30(a) shows a chest X-ray with a narrow range of gray levels. Our objective is to sharpen the image.
- Fig 4.30(c) shows image using HFE (with a = 0.5 and b = 2.0). Although the image is still dark, the gray level tonality due to the low frequency components was not lost.
- Fig. 4.30(d) shows image that is been performing histogram equalization.

High-Frequency Emphasis Filtering

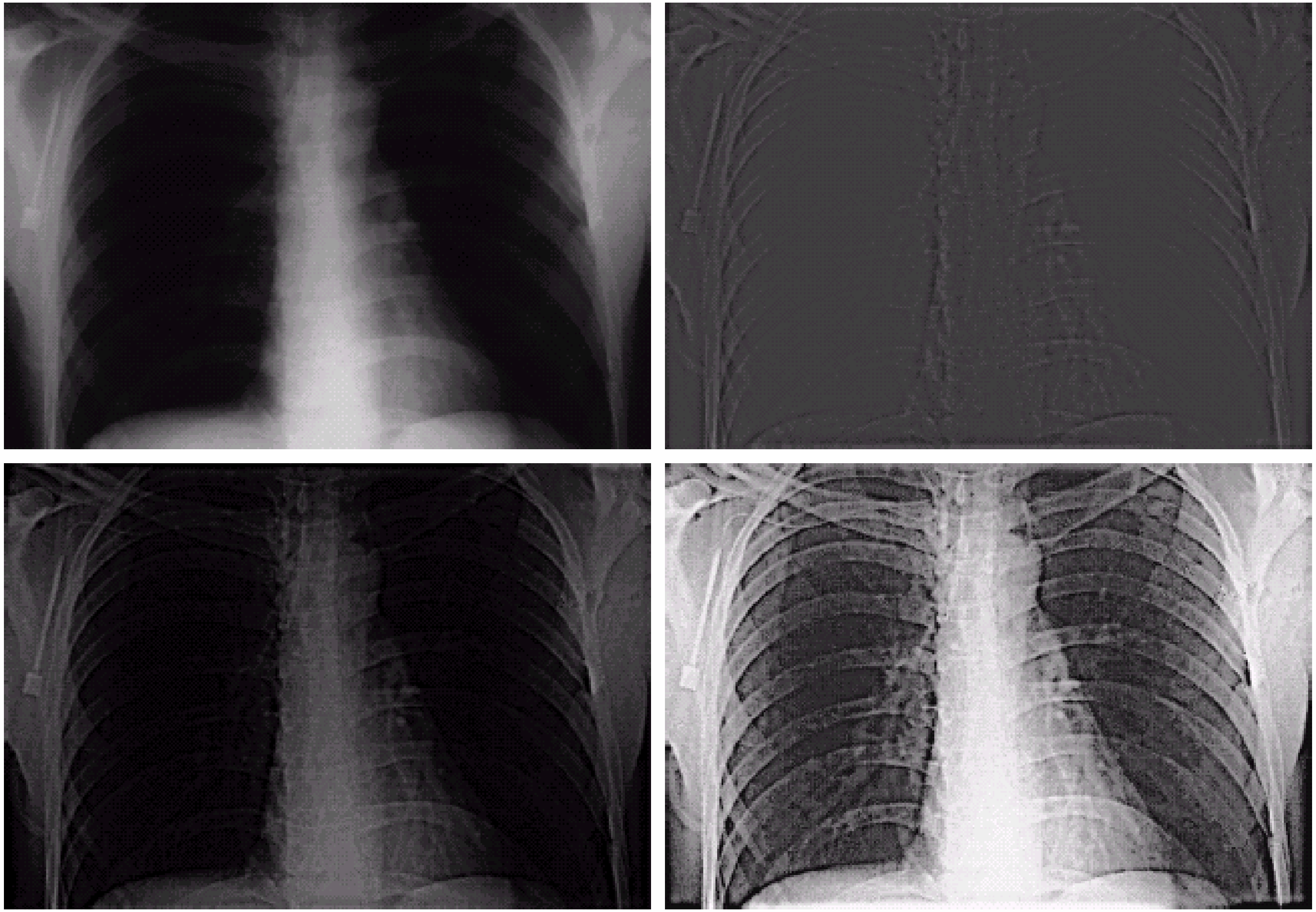


FIGURE 4.30 (a) A chest X-ray image. (b) Result of Butterworth highpass filtering. (c) Result of high-frequency emphasis filtering. (d) Result of performing histogram equalization on (c). (Original image courtesy Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

Homomorphic Filtering

- We can view an image f(x,y) as a product of two components:

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

$$0 < i(x,y) < \infty$$

$$0 < r(x,y) < 1$$

- i(x,y): illumination. It is determined by the illumination source.
- r(x,y): reflectance (or transmissivity). It is determined by the characteristics of imaged objects.

Homomorphic Filtering

- In some images, the quality of the image has reduced because of non-uniform illumination.
- Homomorphic filtering can be used to perform illumination correction.

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

- The above equation cannot be used directly in order to operate separately on the frequency components of illumination and reflectance.

Homomorphic Filtering

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

H(u,v) : $S(u,v) = H(u,v)Z(u,v)$

(DFT)$^{-1}$: $s(x,y) = i'(x,y) + r'(x,y)$

exp : $g(x,y) = \exp(s(x,y)) = i_0(x,y)r_0(x,y)$

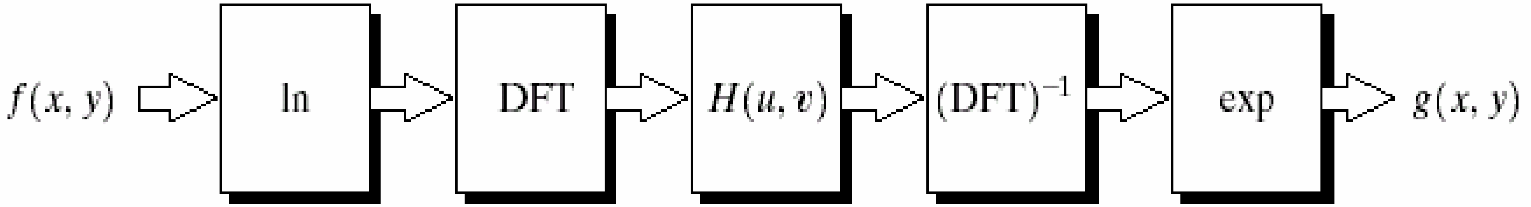


FIGURE 4.31 Homomorphic filtering approach for image enhancement.

Homomorphic Filtering

- By separating the illumination and reflectance components, homomorphic filter can then operate on them separately.
- Illumination component of an image generally has slow variations, while the reflectance component vary abruptly.
- By removing the low frequencies (highpass filtering) the effects of illumination can be removed .

Homomorphic Filtering

- A good idea of control can be gained over the illumination and reflectance components with a homomorphic filter. This control requires specification of a filter function *H(u,v)* that affects the low and high frequency components of the Fourier transform in different ways.
- Fig. 4.32 shows a cross section of such a filter. If the parameters γ_L and γ_H are chosen so that γ_L <1and γ_H >1.
- The curve in Fig. 4.32 can be approximated using modified Gaussian highpass filter:

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

Homomorphic Filtering

- Fig. 4.33 is typical of the results that can be obtained with the homomorphic filtering function in Fig. 4.32.
- Fig. 4.33(b) shows the result of processing this image by homomorphic filtering, with γ_L = 0.5 γ_H =2.0 in the filter function of Fig. 4.32.
- A reduction of dynamic range in the brightness, together with an increase in contrast, brought out the details of objects inside the shelter and balanced the gray levels of the outside wall. The enhanced image also is sharper.

Homomorphic Filtering

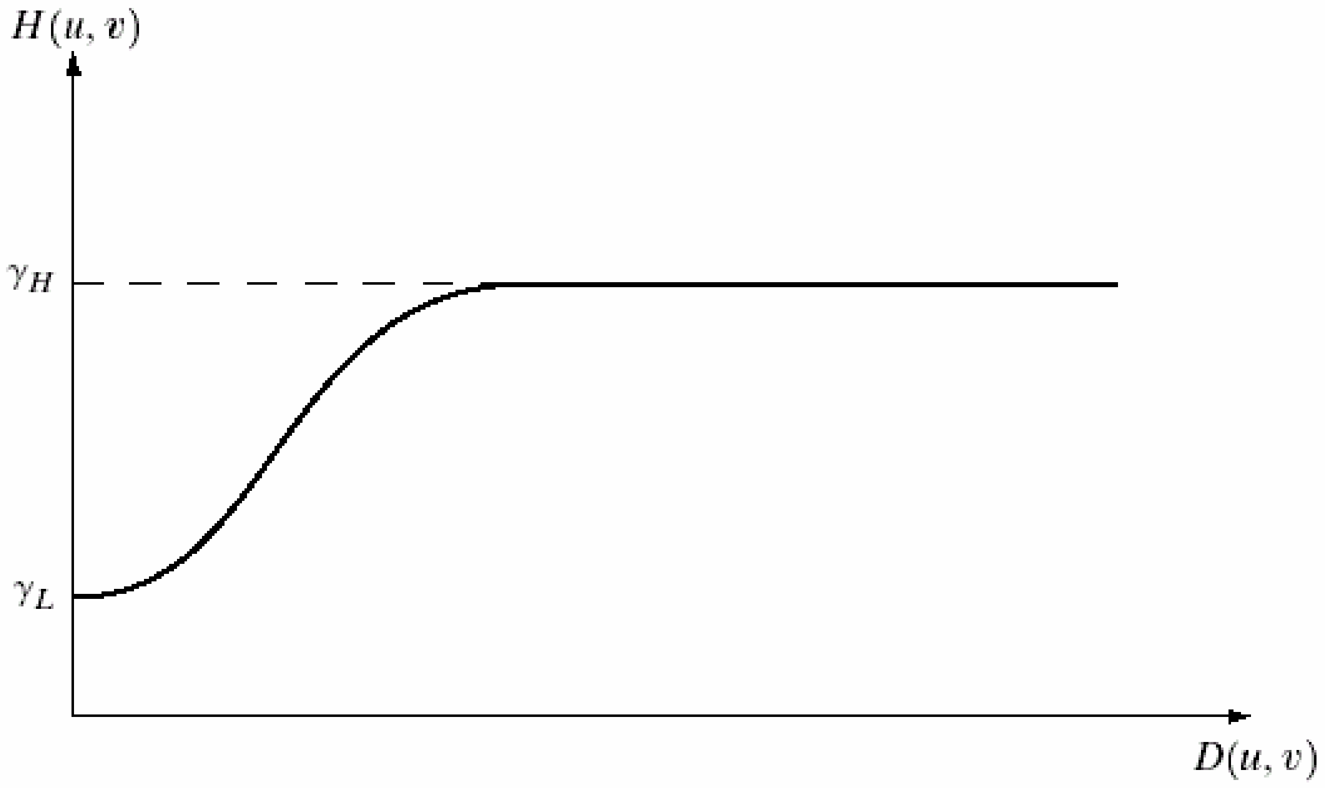
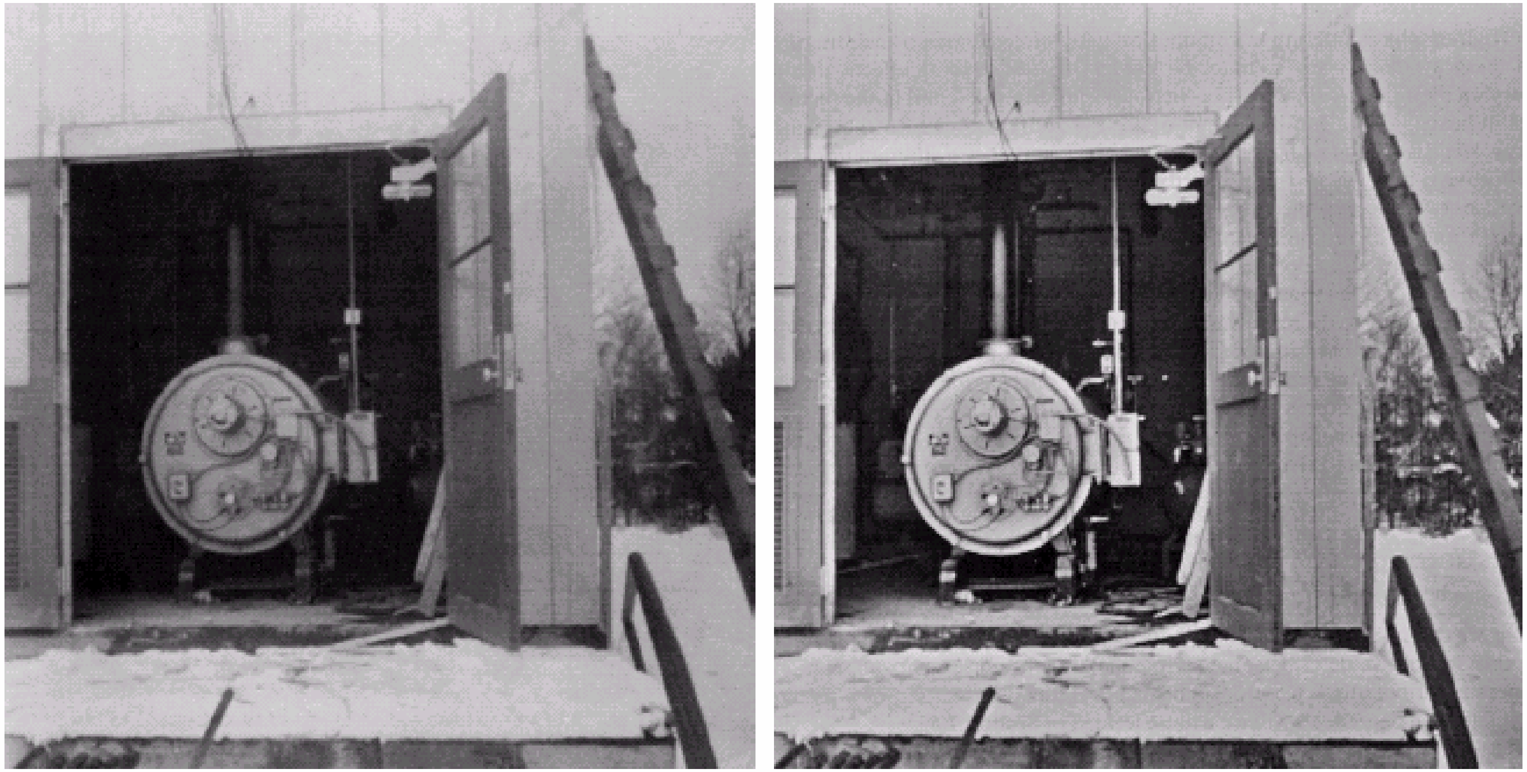


FIGURE 4.32 Cross section of a circularly symmetric filter function. $D(u, v)$ is the distance from the origin of the centered transform.

Homomorphic Filtering

a b

FIGURE 4.33 (a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)



- Background
- Introduction to the Fourier Transform and the Frequency Domain
- Smoothing Frequency-Domain Filters
- Sharpening Frequency Domain Filters
- Homomorphic Filtering
- Implementation

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- The Fourier transform pair has the following translation properties:

$$f(x,y)e^{j2\pi(u_0x/M+v_0y/N)} \Leftrightarrow F(u-u_0,v-v_0) \quad (4.6\text{-}1)$$

and

$$f(x-x_0,y-y_0) \Leftrightarrow F(u,v)e^{-j2\pi(ux_0/M+vy_0/N)} \quad (4.6\text{-}2)$$

- When $u_0 = M/2$ and $v_0 = N/2$, it follows that

$$e^{j2\pi(u_0 x/M + v_0 y/N)} = e^{j\pi(x+y)} = (-1)^{x+y}$$

In this case, Eq. (4.6-1) becomes

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

and, similarly

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

Distributivity snd scaling

- From the definition of the Fourier transform it follows that

- And, in general, that

- The Fourier transfoem is distributive over addition, but nor over multiplication.

Distributivity snd scaling

- For two scalars a and b

$$af(x,y) \Leftrightarrow aF(u,v)$$

and

$$f(ax,by) \Leftrightarrow \frac{1}{|ab|}F(u/a,v/b)$$

Rotation

- If we introduction the polar coordinates

$$x = r\cos\theta \qquad y = r\sin\theta \qquad u = \omega\cos\varphi \qquad v = \omega\sin\varphi$$

- Then $f(x,y)$ and $F(u,v)$ become $f(\gamma,\theta)$ and $F(\omega,\varphi)$.
- Direct substitution into definition of the Fourier transform yields

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

Periodicity and conjugate symmetry

- The discrete Fourier transform has the following periodicity properties:

$$F(u,v) = F(u+M,v) = F(u,v+N) = F(u+M,v+N)$$

- The inverse transform also is periodic:

$$f(x,y) = f(x+M,y) = f(x,y+N) = f(x+M,y+N)$$

- Conjugate symmetry

$$F(u,v) = F^*(-u,-v)$$

- The spectrum also is symmetric

$$\left|F(u,v)\right| = \left|F(-u,-v)\right|$$

Separability

- The discrete Fourier transform in Eq. (4.2-16) can be expressed in the separable form

$$F(u,v)=\frac{1}{M}\sum_{x=0}^{M-1}e^{-j2\pi ux/M}\;\frac{1}{N}\sum_{y=0}^{N-1}f(x,y)e^{-j2\pi vy/N}$$

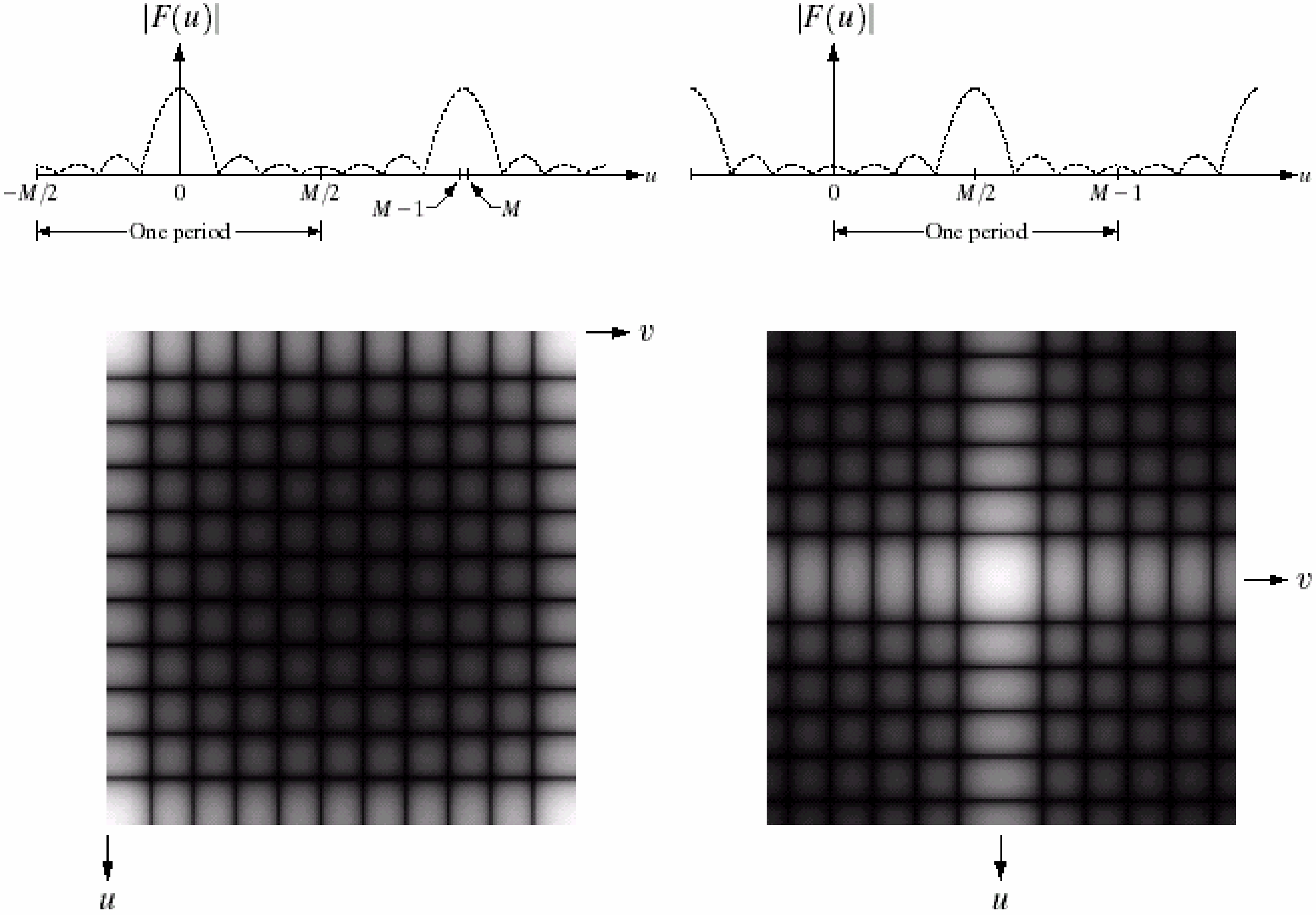
$$=\frac{1}{M}\sum_{x=0}^{M-1}F(x,v)e^{-j2\pi ux/M}$$

where

$$F(x,v)=\frac{1}{N}\sum_{y=0}^{N-1}f(x,y)e^{-j2\pi vy/N}$$

a b
c d

FIGURE 4.34 (a) Fourier spectrum showing back-to-back half periods in the interval $[0, M - 1]$. (b) Shifted spectrum showing a full period in the same interval. (c) Fourier spectrum of an image, showing the same back-to-back properties as (a), but in two dimensions. (d) Centered Fourier spectrum.



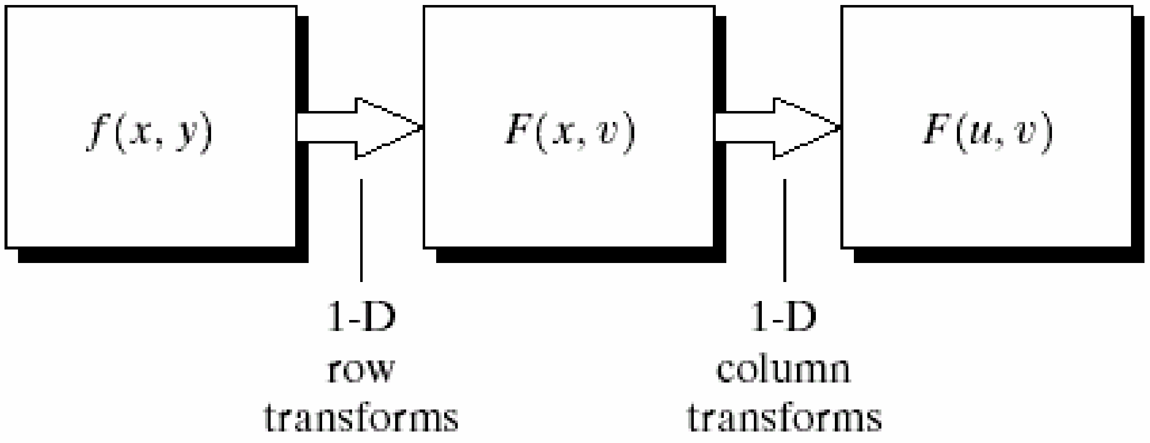


FIGURE 4.35 Computation of the 2-D Fourier transform as a series of 1-D transforms.

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- The 1-DFourier transforms:

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{-j2\pi ux/M} \qquad (4.6\text{-}16)$$

and

$$f(x) = \sum_{x=0}^{M-1} F(u)e^{j2\pi ux/M} \qquad (4.6\text{-}17)$$

- Taking the complex conjugate of Eq. (4.6-17) and dividing both sides by M yields

$$\frac{1}{M} f^*(x) = \frac{1}{M} \sum_{u=0}^{M-1} F^*(u) e^{-j2\pi ux/M}$$

- A similar analysis for two variables yields:

$$\frac{1}{M} f^*(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u,v) e^{-j2\pi(ux/M+vy/N)}$$

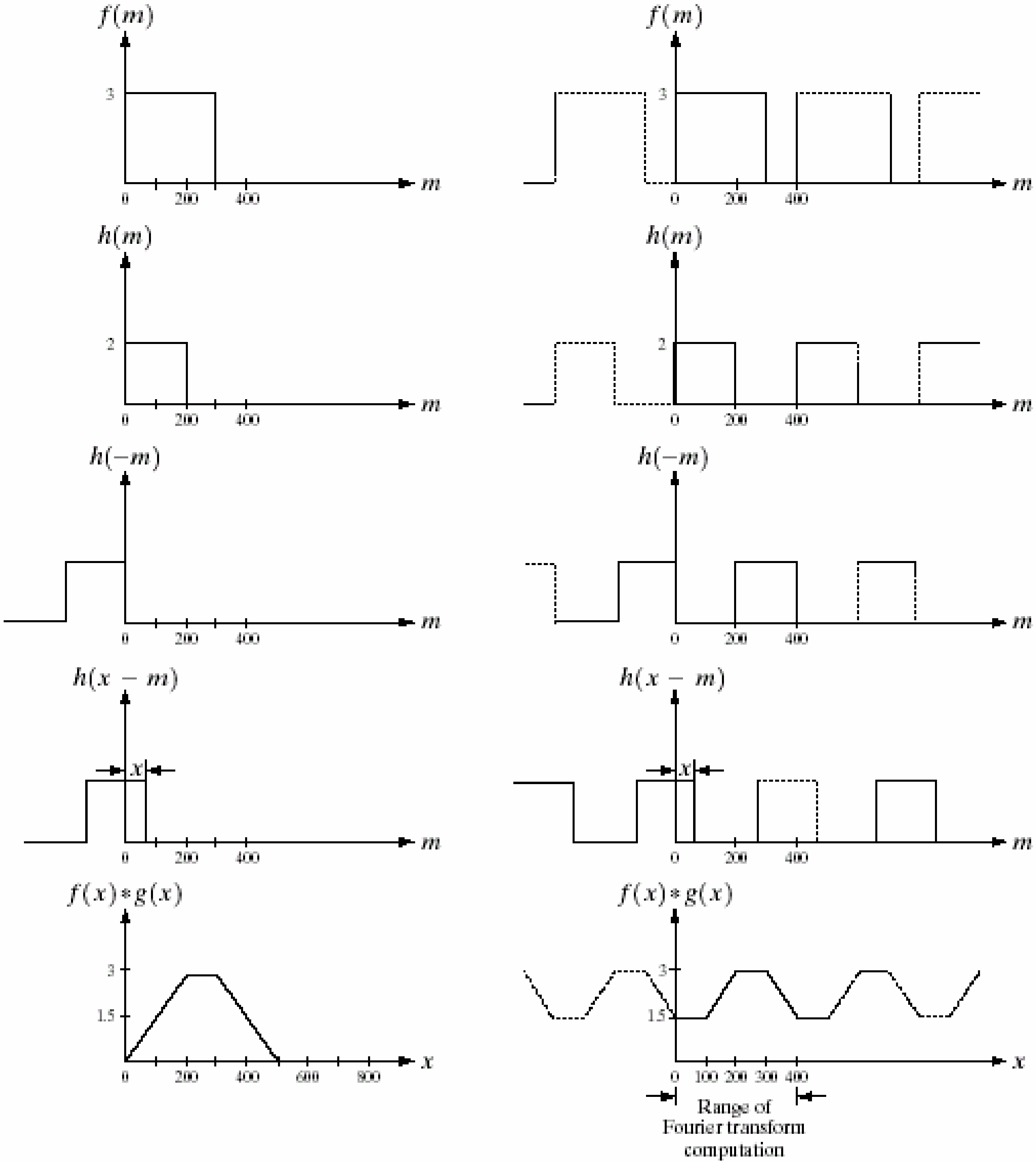
- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- Figure 4.36 illustrates the significance of periodicity. The left column of this figure shows convolution computed using the 1-D version of Eq. (4.2-30):

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

a f
b g
c h
d i
e j

FIGURE 4.36 Left: convolution of two discrete functions. Right: convolution of the same functions, taking into account the implied periodicity of the DFT. Note in (j) how data from adjacent periods corrupt the result of convolution.



- This procedure yields extended, or padded, functions given by

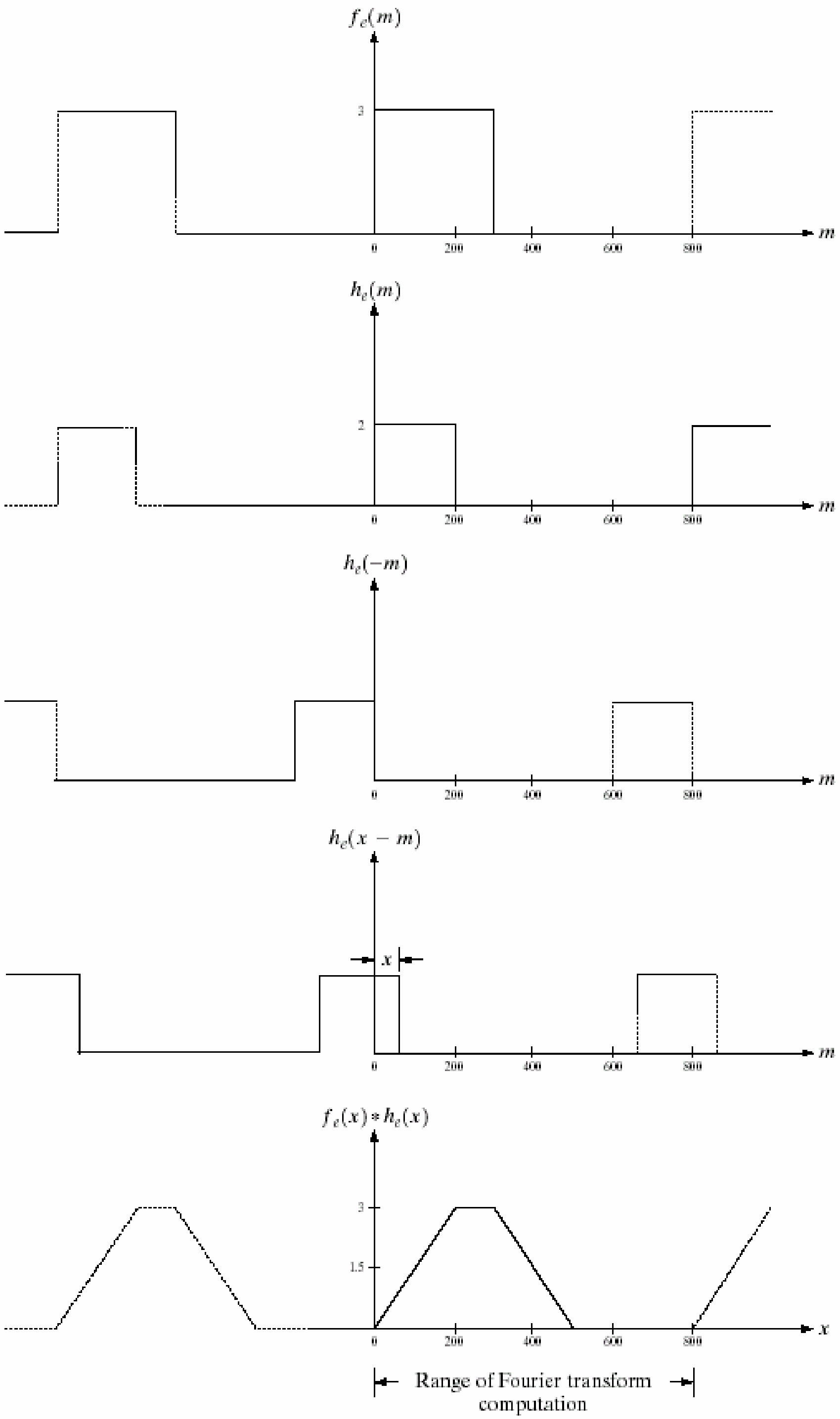
$$f_e(x) = \begin{cases} f(x) & 0 \leq x \leq A-1 \\ 0 & A \leq x \leq P \end{cases}$$

and

$$g_e(x) = \begin{cases} g(x) & 0 \leq x \leq B-1 \\ 0 & B \leq x \leq P \end{cases}$$

a
b
c
d
e

FIGURE 4.37 Result of performing convolution with extended functions. Compare Figs. 4.37(e) and 4.36(e).



- Suppose that we have two images f(x,y) and h(x,y) of sizes A*B and C*D, respectively. Wraparound error in 2-D convolution is avoided by choosing:

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

and

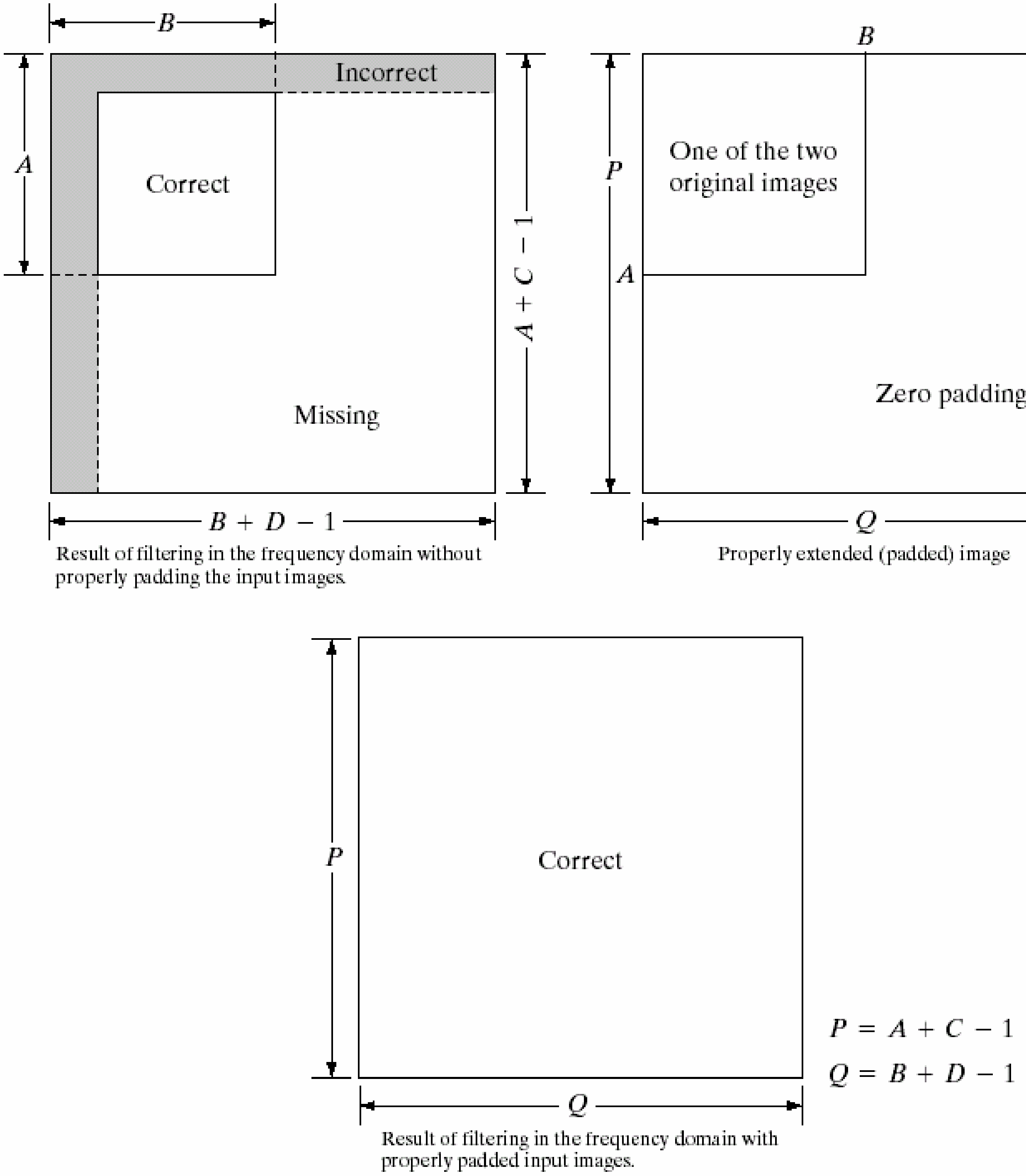
$$Q \geq B + D - 1$$

- The periodic sequences are formed by extending f(x,y) and h(x,y) as follows:

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

and

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



Result of filtering in the frequency domain without properly padding the input images.

Properly extended (padded) image

Result of filtering in the frequency domain with properly padded input images.

a b
c

FIGURE 4.38 Illustration of the need for function padding. (a) Result of performing 2-D convolution without padding. (b) Proper function padding. (c) Correct convolution result.

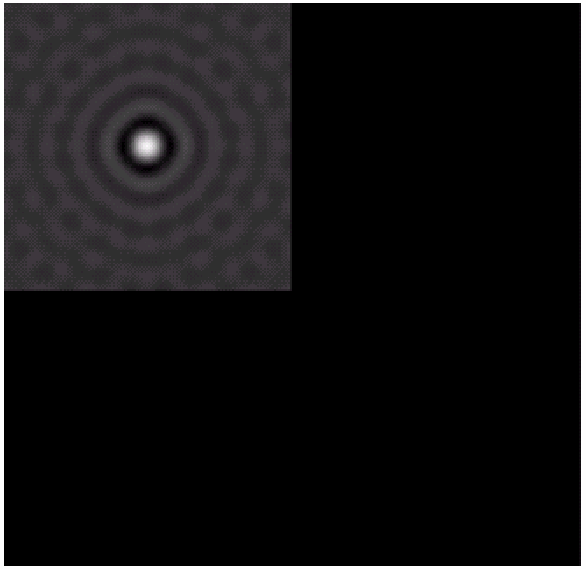


FIGURE 4.39 Padded lowpass filter is the spatial domain (only the real part is shown).

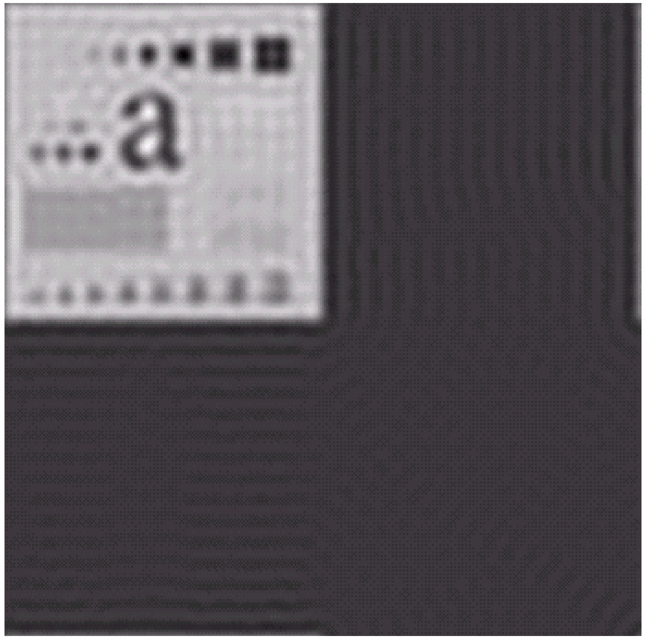


FIGURE 4.40 Result of filtering with padding. The image is usually cropped to its original size since there is little valuable information past the image boundaries.

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- The discrete convolution of two functions f(x,y) and h(x,y) of size M*N is denoted by f(x,y)*h(x,y) and is defined by the expression:

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

- The convolution theorem consists of the following relationships between the two functions and their Fourier transforms:

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

and

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

- The correlation of two function f(x,y) and h(x,y) is defined as:

$$f(x,y) \circ h(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^{*}(m,n) h(x+m, y+n)$$

where f* denotes the complex conjugate of f.

- There is a correlation theorem, analogous to the convolution theorem. Let F(u,v) and H(u,v) denote the Fourier transforms of f(x,y) and h(x,y).

$$f(x,y) \circ h(x,y) \Leftrightarrow F^{*}(u,v)H(u,v)$$

- An analogous result is that correlation in the frequency domain reduces to multiplication in the spatial domain; that is

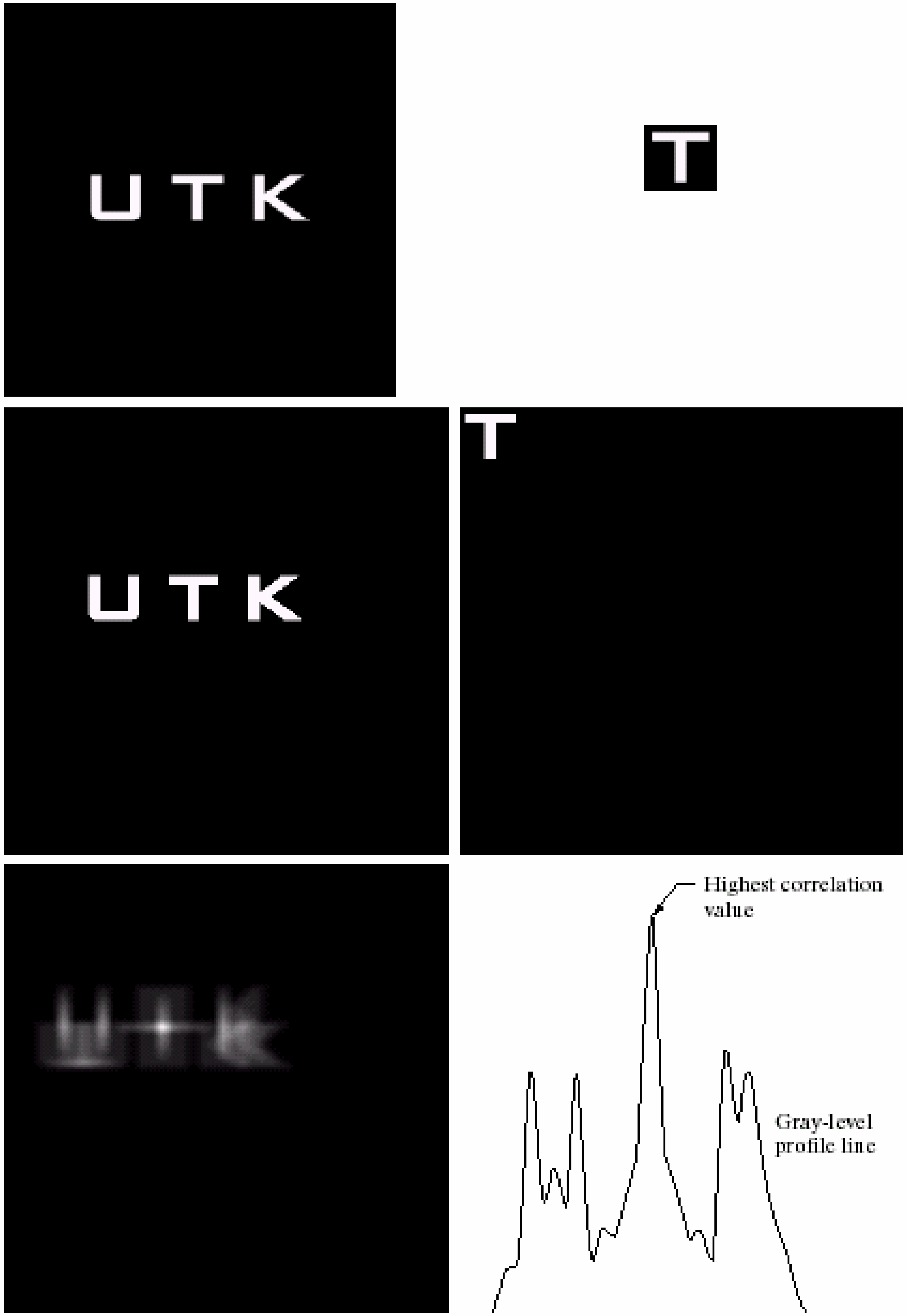
$$f^{*}(x,y)h(x,y) \Leftrightarrow F(u,v) \circ H(u,v)$$

- The autocorrelation theorem:

$$f(x,y) \circ f(x,y) \Leftrightarrow \left|F(u,v)\right|^2$$

- Similarly,

$$\left|f(x,y)\right|^2 \Leftrightarrow F(u,v) \circ F(u,v)$$



a	b
c	d
e	f

FIGURE 4.41 (a) Image. (b) Template. (c) and (d) Padded images. (e) Correlation function displayed as an image. (f) Horizontal profile line through the highest value in (e), showing the point at which the best match took place.

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

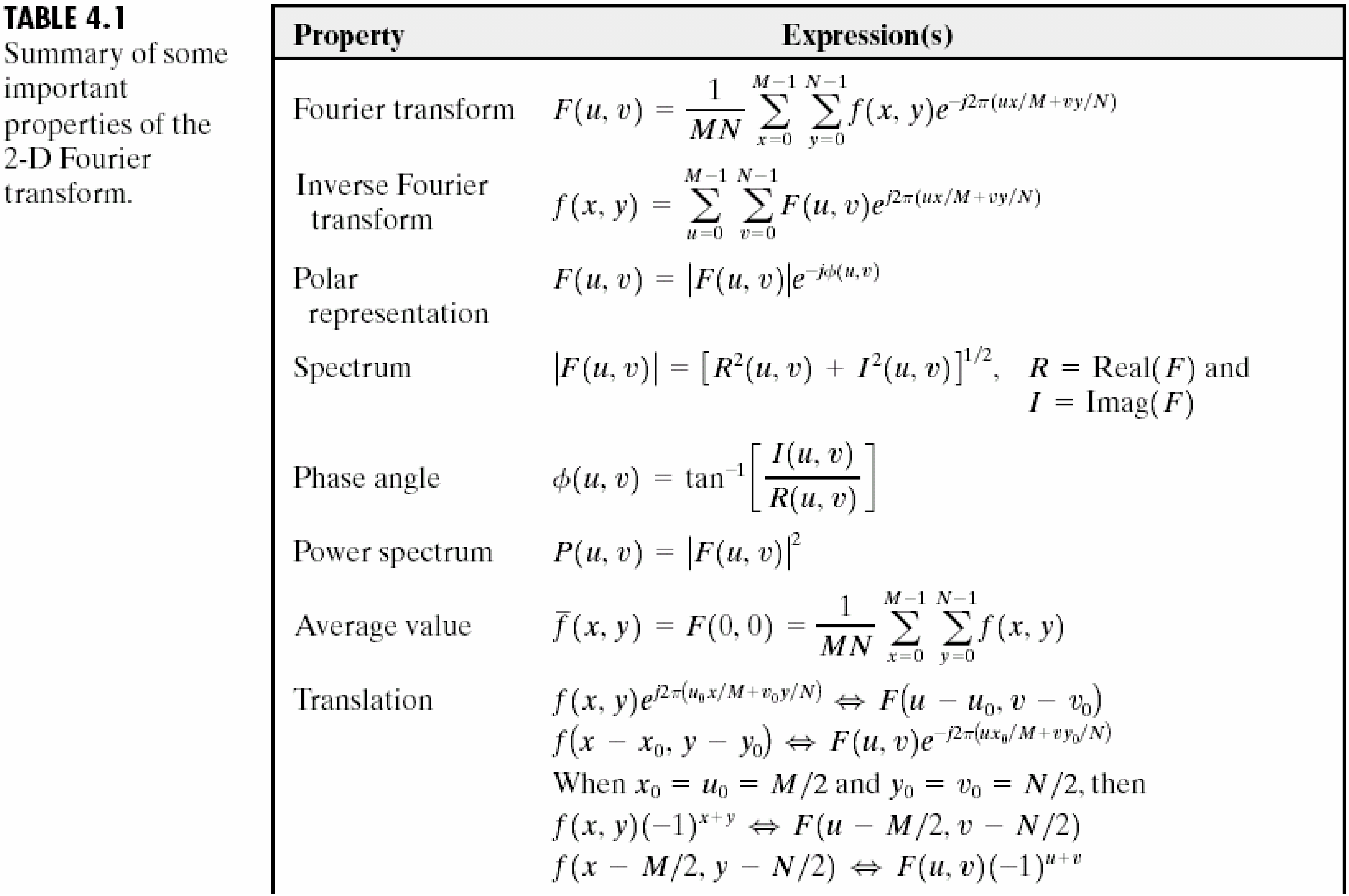
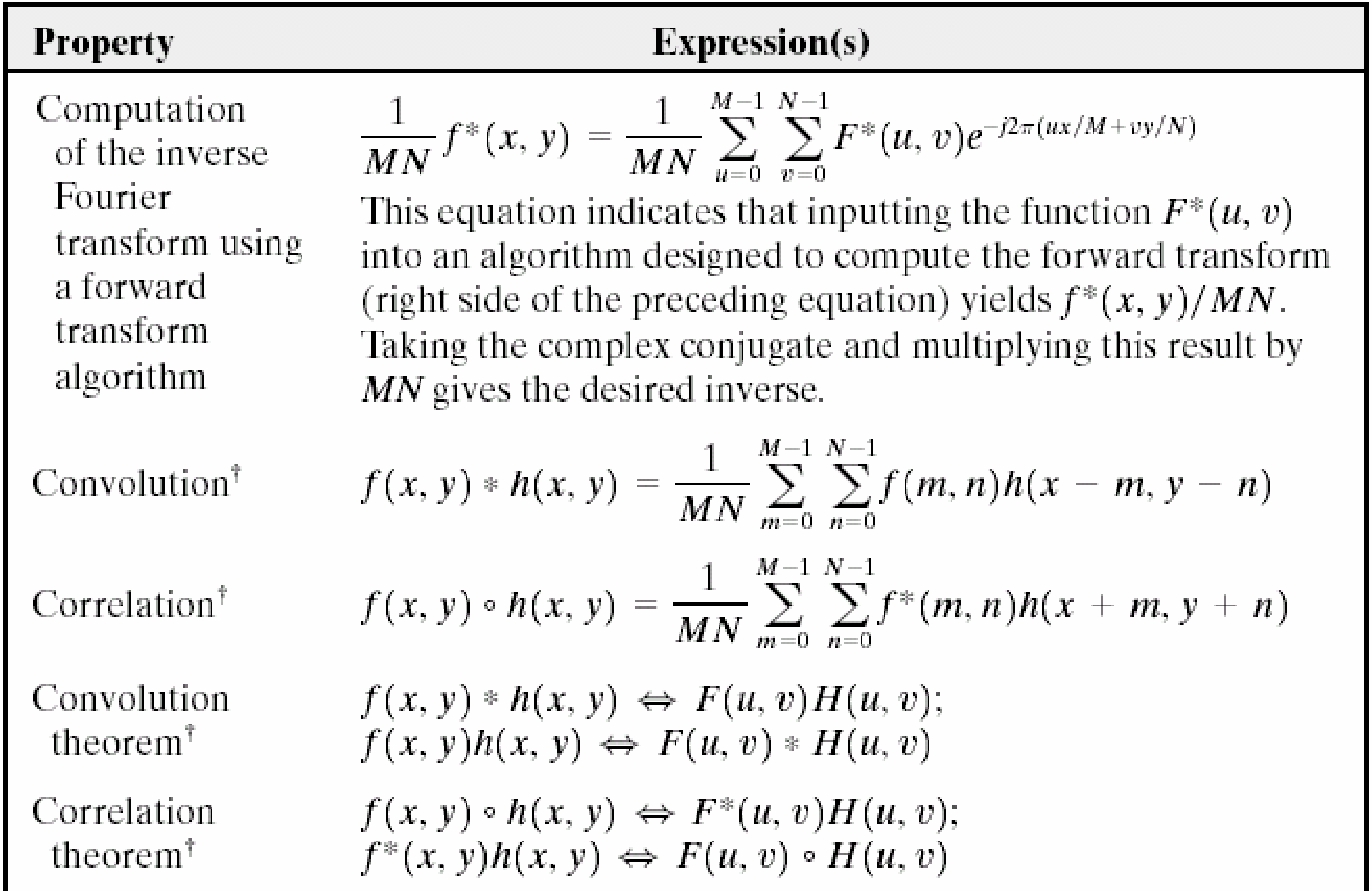


TABLE 4.1 Summary of some important properties of the 2-D Fourier transform.

Property	Expression(s)
Fourier transform	$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$
Inverse Fourier transform	$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)}$
Polar representation	$F(u, v) = \lvert F(u, v) \rvert e^{-j\phi(u,v)}$
Spectrum	$\lvert F(u, v) \rvert = \left[R^2(u, v) + I^2(u, v) \right]^{1/2}$, $R = \text{Real}(F)$ and $I = \text{Imag}(F)$
Phase angle	$\phi(u, v) = \tan^{-1}\left[\frac{I(u, v)}{R(u, v)} \right]$
Power spectrum	$P(u, v) = \lvert F(u, v) \rvert^2$
Average value	$\bar{f}(x, y) = F(0, 0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$
Translation	$f(x, y) e^{j2\pi(u_0 x/M + v_0 y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v) e^{-j2\pi(ux_0/M + vy_0/N)}$ When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then $f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{u+v}$

Conjugate symmetry	$F(u, v) = F^*(-u, -v)$ $\lvert F(u, v)\rvert = \lvert F(-u, -v)\rvert$
Differentiation	$\dfrac{\partial^n f(x, y)}{\partial x^n} \Leftrightarrow (ju)^n F(u, v)$ $(-jx)^n f(x, y) \Leftrightarrow \dfrac{\partial^n F(u, v)}{\partial u^n}$
Laplacian	$\nabla^2 f(x, y) \Leftrightarrow -(u^2 + v^2)F(u, v)$
Distributivity	$\Im[f_1(x, y) + f_2(x, y)] = \Im[f_1(x, y)] + \Im[f_2(x, y)]$ $\Im[f_1(x, y) \cdot f_2(x, y)] \neq \Im[f_1(x, y)] \cdot \Im[f_2(x, y)]$
Scaling	$af(x, y) \Leftrightarrow aF(u, v), f(ax, by) \Leftrightarrow \dfrac{1}{\lvert ab\rvert} F(u/a, v/b)$
Rotation	$x = r\cos\theta \qquad y = r\sin\theta \qquad u = \omega\cos\varphi \qquad v = \omega\sin\varphi$ $f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$
Periodicity	$F(u, v) = F(u + M, v) = F(u, v + N) = F(u + M, v + N)$ $f(x, y) = f(x + M, y) = f(x, y + N) = f(x + M, y + N)$
Separability	See Eqs. (4.6-14) and (4.6-15). Separability implies that we can compute the 2-D transform of an image by first computing 1-D transforms along each row of the image, and then computing a 1-D transform along each column of this intermediate result. The reverse, columns and then rows, yields the same result.

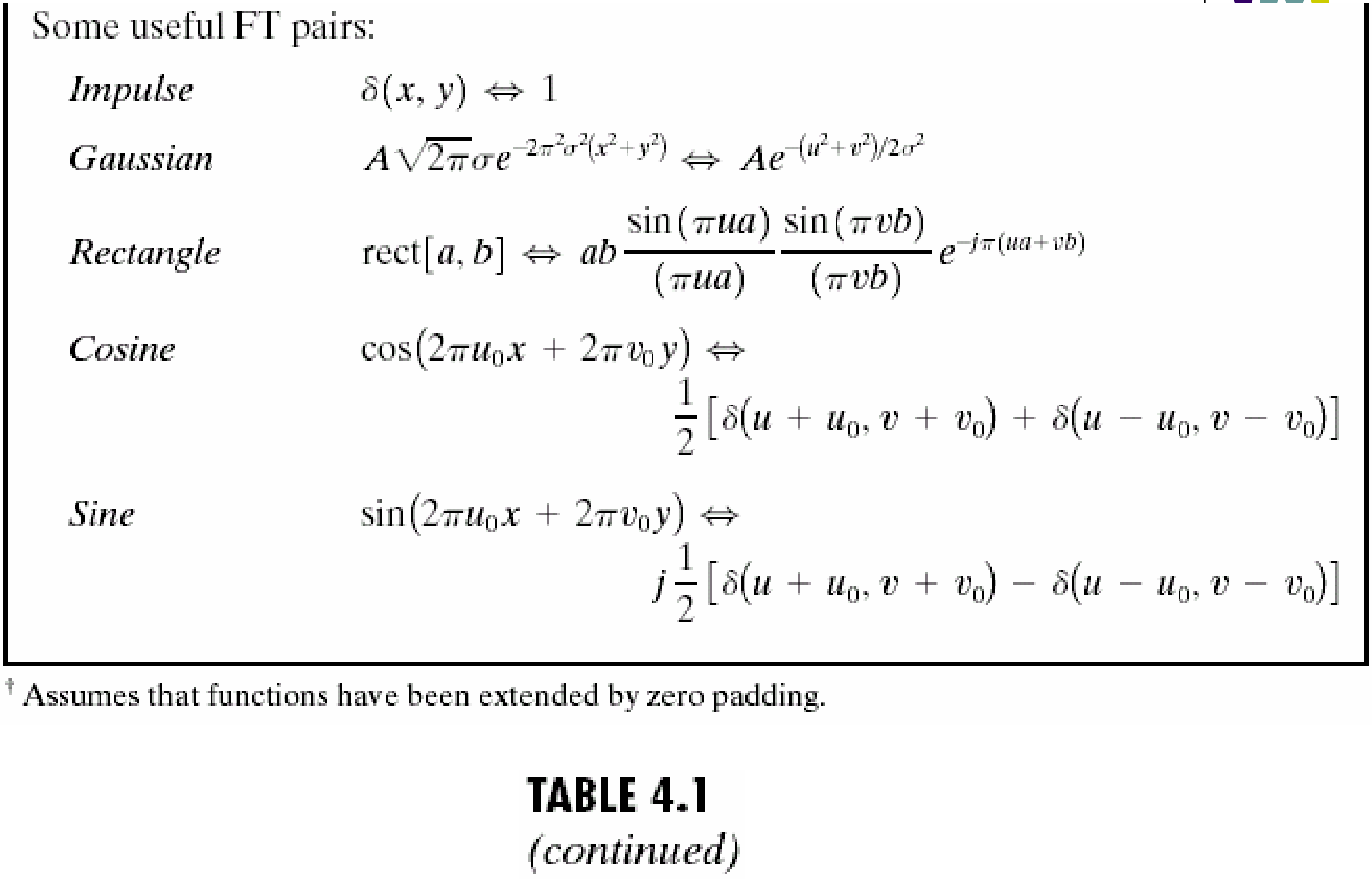
TABLE 4.1
(continued)



Property	Expression(s)
Computation of the inverse Fourier transform using a forward transform algorithm	$\frac{1}{MN} f^*(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v) e^{-j2\pi(ux/M + vy/N)}$ This equation indicates that inputting the function $F^*(u, v)$ into an algorithm designed to compute the forward transform (right side of the preceding equation) yields $f^*(x, y)/MN$. Taking the complex conjugate and multiplying this result by MN gives the desired inverse.
Convolution[†]	$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n)$
Correlation[†]	$f(x, y) \circ h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n) h(x + m, y + n)$
Convolution theorem[†]	$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v)$; $f(x, y) h(x, y) \Leftrightarrow F(u, v) * H(u, v)$
Correlation theorem[†]	$f(x, y) \circ h(x, y) \Leftrightarrow F^*(u, v) H(u, v)$; $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$

TABLE 4.1

(continued)



Some useful FT pairs:

Impulse	$\delta(x, y) \Leftrightarrow 1$
Gaussian	$A\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2(x^2+y^2)} \Leftrightarrow Ae^{-(u^2+v^2)/2\sigma^2}$
Rectangle	$\text{rect}[a, b] \Leftrightarrow ab\frac{\sin(\pi ua)}{(\pi ua)}\frac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $\frac{1}{2}[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)]$
Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $j\frac{1}{2}[\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)]$

† Assumes that functions have been extended by zero padding.

TABLE 4.1
(continued)

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- The FFT algorithm developed in this section is based on the so-called successive doubling method. For notational convenience we express Eq. (4.2-5) in the form

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)W_M^{ux}$$

Where $W_M = e^{-j2\pi/M}$

And M is assumed to be of the form

$M = 2^n$ or $M = 2K$

- Substitution of Eq. (4.6-38) into Eq. (4.6-35) yields

$$F(u) = \frac{1}{2K}\sum_{x=0}^{2K-1} f(x)W_{2K}^{ux}$$

$$= \frac{1}{2}\left[\frac{1}{K}\sum_{x=0}^{K-1} f(2x)W_{K}^{u(2x)} + \frac{1}{K}\sum_{x=0}^{K-1} f(2x+1)W_{2K}^{u(2x+1)}\right]$$

- Using $W_{2K}^{2ux} = W_{K}^{ux}$

$$F(u) = \frac{1}{2}\left[\frac{1}{K}\sum_{x=0}^{K-1} f(2x)W_{K}^{ux} + \frac{1}{K}\sum_{x=0}^{K-1} f(2x+1)W_{K}^{ux}W_{2K}^{ux}\right]$$

- Defining $$F_{even}(u) = \frac{1}{K}\sum_{x=0}^{K-1} f(2x)W_K^{ux}$$

  and $$F_{odd}(u) = \frac{1}{K}\sum_{x=0}^{K-1} f(2x+1)W_K^{ux}$$

$$F(u) = \frac{1}{2}\left[F_{even}(u) + F_{odd}(u)W_{2K}^{ux}\right]$$

Because $W_M^{u+M} = W_M^u$ and $W_{2M}^{u+M} = -W_{2M}^u$

$$F(u+K) = \frac{1}{2}\left[F_{even}(u) - F_{odd}(u)W_{2K}^{u}\right]$$

- Continuing this argument for any positive integer value of n leads to recursive expressions for the number of multiplications and additions required to implement the FFT:

$$m(n) = 2m(n-1) + 2^{n-1}, \qquad n \geq 1$$

and

$$a(n) = 2a(n-1) + 2^{n}, \qquad n \geq 1$$

- The computational advantage of the FFT over a direct implementation of the 1-D DFT is defined as

$$C(M) = \frac{M^2}{M \log_2 M} = \frac{M}{\log_2 M}$$

- Because it is assumed that $M = 2^n$, we can express Eq. (4.6-49) in terms of n:

$$C(n) = \frac{2^n}{n}$$

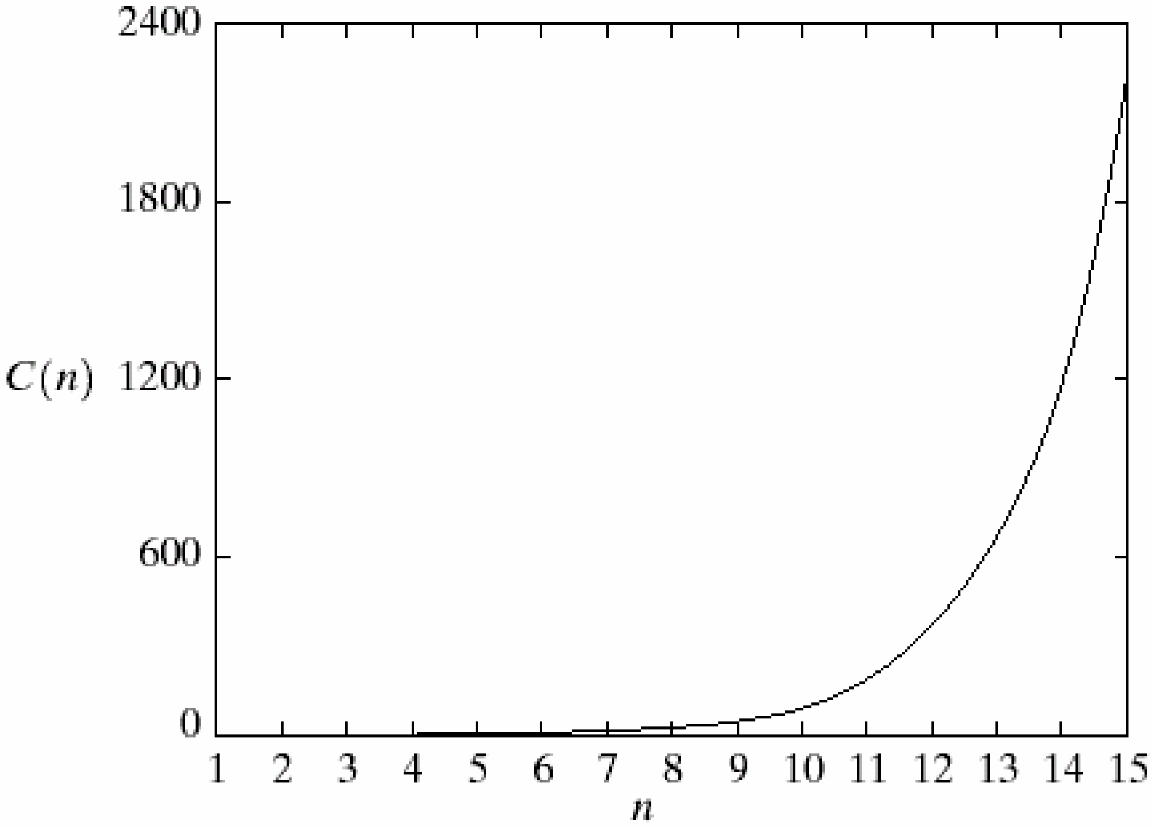


FIGURE 4.42 Computational advantage of the FFT over a direct implementation of the 1-D DFT. Note that the advantage increases rapidly as a function of ***n***.

- Some Additional Properties of the 2-D Fourier Transform
- Computing the Inverse Fourier Transform Using a Forword Transform Algorithm
- More on Periodicity: the Need for Padding
- The Convolution and Correlation Theorems
- Summary of Properties of the 2-D Fourier Tandform
- The Fast Fourier Transform
- Some Comments on Filter Design

- All the filters discussed in this chapter are specified in equation form.
- In order to use the filters, we simply sample the equation for the desired values of (u,v). This process results in the filter function H(u,v).
- In all our examples, this function was multiplied by the DFT of the input image, and the inverse DFT was computed.
- All forward and inverse Fourier transforms in this chapter were computed with an FFT algorithm.

- The approach to filtering discussed in this chapter is focused strictly on fundamentals, the focus being specifically to explain the effects of filtering in the frequency domain as clearly as possible.
- We know of no better way to do that than to treat filtering the way we did here. One can view this development as the basis for "prototyping" a filter. In other words, given a problem for which we want to find a filter, the frequency domain approach is an ideal tool for experimenting, quickly and with full control over filter parameters.
- Once a filter for a specific application has been found, it often is of interest to implement the filter directly in the spatial domain, using firmware and/or hardware.