Filtering in the Frequency Domain

Contents

- Background
- Introduction to the Fourier Transform and the Frequency Domain
  - 1D DFT
  - 2D DFT
- Frequency Domain Filtering Fundamentals
- Smoothing Frequency Domain Filters
- Sharpening Frequency Domain Filters

Background

- Jean Baptiste Joseph Fourier
  - The French mathematician, was born in 1786
- *Fourier series*
  - Any function (meets some mild mathematical conditions) that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient

Background

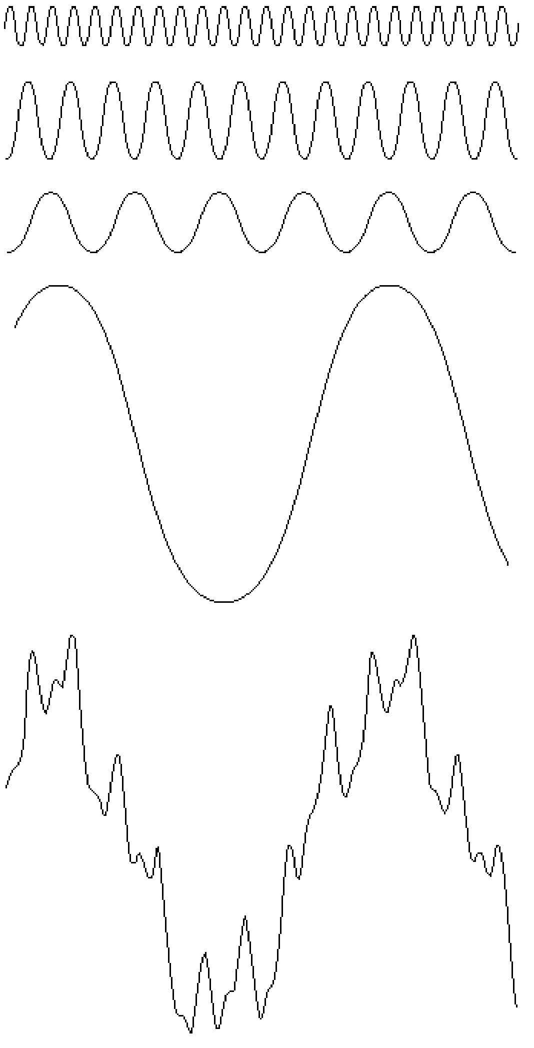


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

Background

- *Fourier transform*
  - Even functions that are not periodic (but whose area under the curve is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighting function
  - Its utility is even greater than the Fourier series in most practical problems

Background

- One of the most important characteristics of these representations
  - A function, expressed by either a Fourier series or transform, can be reconstructed (recovered) completely via an inverse process, with no loss of information
  - Allowing us to work in the “Fourier domain” and then return to the original domain of the function without losing any information

Background

- During the past century, and especially in the past 50 years, entire industries and academic disciplines have flourished as a result of Fourier's ideas
  - The advent of **digital computation** and the "discovery" of **fast Fourier transform (FFT)** algorithm in the late 1950s Revolutionized the fields of signal processing
  - These two core technologies allowed for the first time practical processing and meaningful interpretation of a host of signals of exceptional human and industrial importance, from medical monitors and scanners to modern electronic communications

Background

- We will dealing with functions (images) of finite duration, so the Fourier transform is the tool in which we are interested
  - Fourier techniques provide a meaningful and practical way to study and implement a host of image enhancement approaches

Introduction to the Fourier Transform

- Introduction to the Fourier transform
  - One dimensional
  - Two dimensional

- The focus is mostly on
  - Discrete formulation
  - Some properties

1-D Fourier Transform

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

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx \qquad \textbf{......(1)}$$

- Conversely, given F(u), we can obtain f(x) by means of *inverse Fourier transform*

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du \qquad \textbf{......(2)}$$

- (1) and (2) comprise the *Fourier transform pair*

1-D Discrete Fourier Transform

- ***DFT***: The Fourier transform of a discrete function of one variable

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

……(3)

- Given F(u), we can obtain the original function f(x) back using the ***inverse DFT***

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

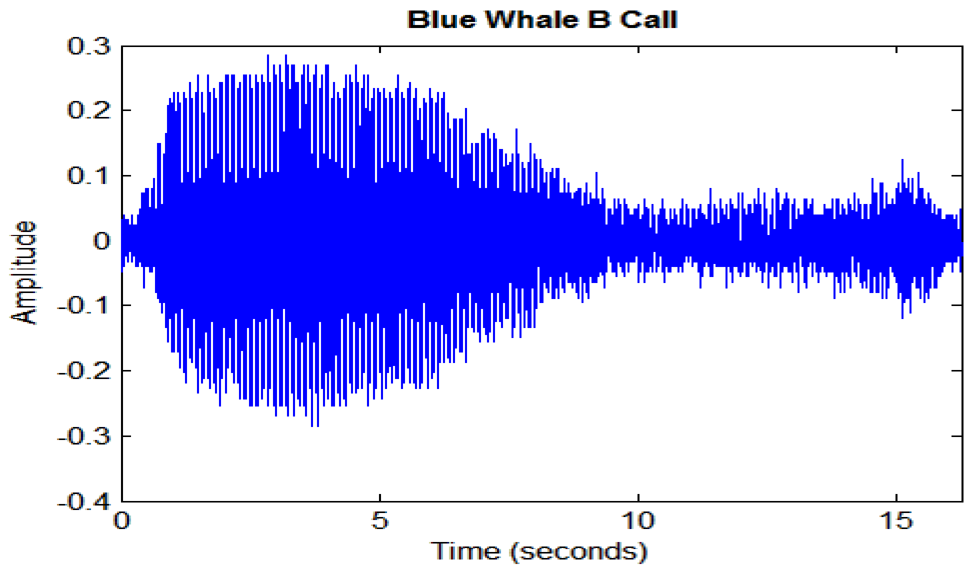
……(4)

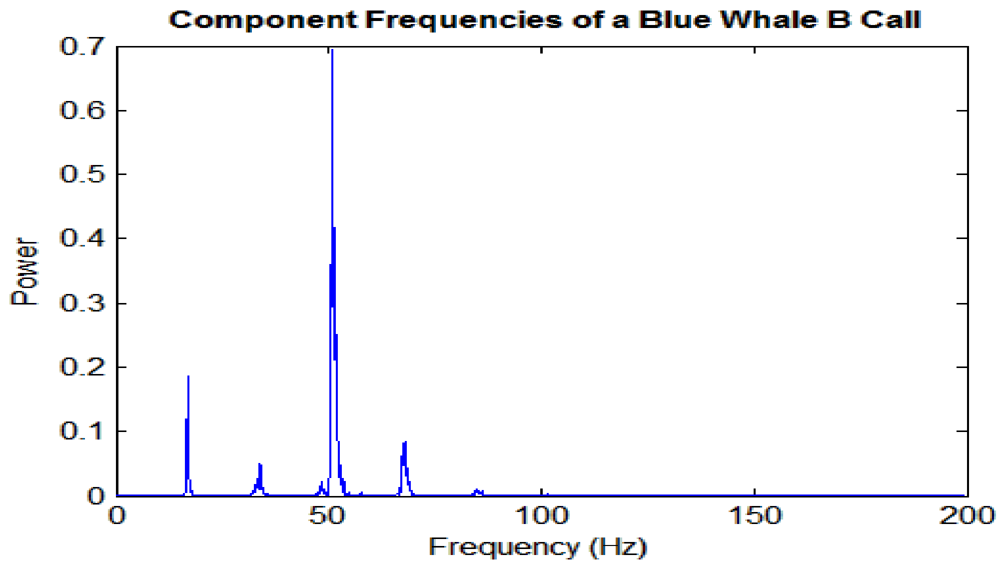
1-D Discrete Fourier Transform

- The domain (value of u) over which the values of $F(u)$ range is appropriately called *frequency domain*
  - Because u determines the frequency of the components of the transform

- Each of the M terms of $F(u)$ is called a *frequency component* of the transform

1-D Discrete Fourier Transform





1-D Discrete Fourier Transform

- A useful analogy is to compare the Fourier transform to a glass prism
  - The prism is a physical device that separates light into various color components, each depending on its wavelength (or frequency) content
  - The Fourier transform may be viewed as a "*mathematical prism*" that separates a function into various components, also based on frequency content

1-D Discrete Fourier Transform

- Euler's formula

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

- Substitute this expression into equation (3) and using the fact that cos(-θ)=cos(θ), give us

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)[\cos 2\pi ux/M - j\sin 2\pi ux/M]$$

for u=0,1,2,…,M-1.

1-D Discrete Fourier Transform

- Express *F*(*u*) in polar coordinates

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

  - *magnitude* or *spectrum*

$$|F(u)| = [R^2(u) + I^2(u)]^{1/2}$$

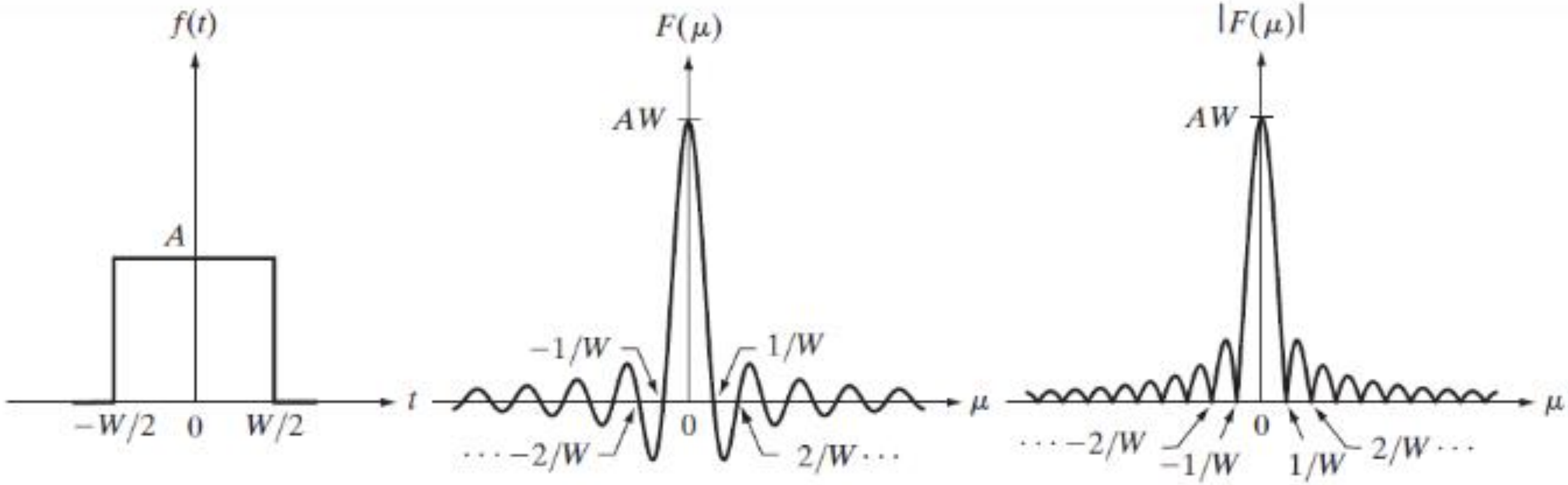
  - *phase angle* or *phase spectrum*

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

  - *Power spectrum*, or *power spectrum density*

$$\begin{aligned} P(u) &= |F(u)|^2 \\ &= R^2(u) + I^2(u) \end{aligned}$$

1-D Discrete Fourier Transform



$$f(x) = \begin{cases} A & x > -W/2 \& x < W/2 \\ 0 & x < -W/2 | x > W/2 \end{cases}$$

$$\begin{aligned} F(\mu) &= \int_{-\infty}^{\infty} f(t) e^{-j2\pi\mu t} dt = \int_{-W/2}^{W/2} A e^{-j2\pi\mu t} dt \\ &= \frac{-A}{j2\pi\mu} \left[e^{-j2\pi\mu t} \right]_{-w/2}^{W/2} = \frac{-A}{j2\pi\mu} \left[e^{-j\pi\mu W} - e^{j\pi\mu W} \right] \\ &= \frac{A}{j2\pi\mu} \left[e^{j\pi\mu W} - e^{-j\pi\mu W} \right] \\ &= AW \frac{\sin(\pi\mu W)}{(\pi\mu W)} \end{aligned}$$

$$|F(\mu)| = AW \left| \frac{\sin(\pi\mu W)}{(\pi\mu W)} \right|$$

1-D Discrete Fourier Transform

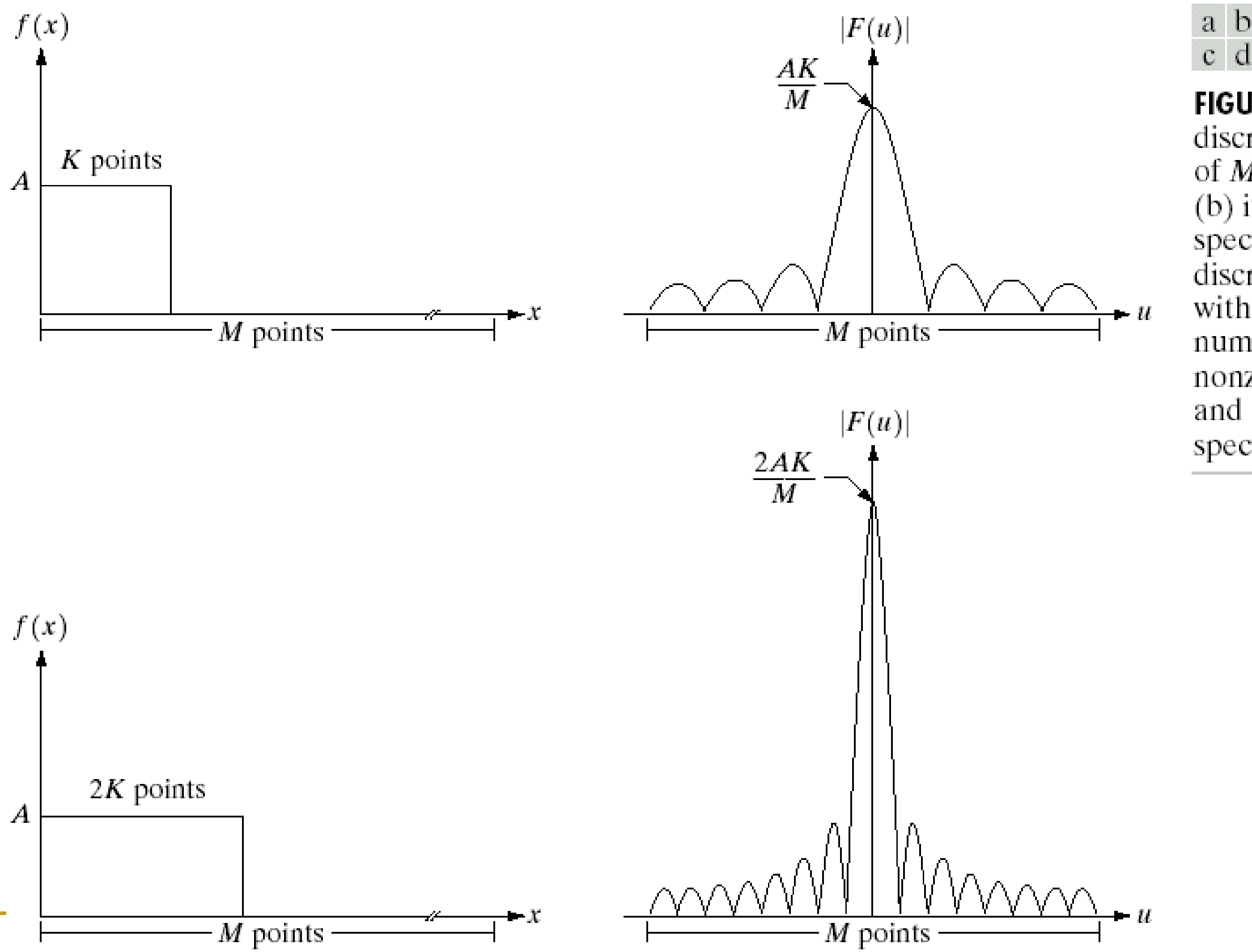


FIGURE 4.2 (a) A discrete function of M points, and (b) its Fourier spectrum. (c) A discrete function with twice the number of nonzero points, and (d) its Fourier spectrum.

1-D Discrete Fourier Transform

- In the discrete case, when we write f(x), it is understood that we are utilizing shorthand notation that really means

$$f(x) \triangleq f(x_0 + x\Delta x)$$

- The sequence for the values of u is 0,Δu,…,[M-1]Δu. F(u) is understood to mean

$$F(u) \triangleq F(u\Delta u)$$

1-D Discrete Fourier Transform

- Δx and Δu are inversely related by the expression

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

  - this relationship is useful when measurements are an issue in the images being processed

1-D Discrete Fourier Transform

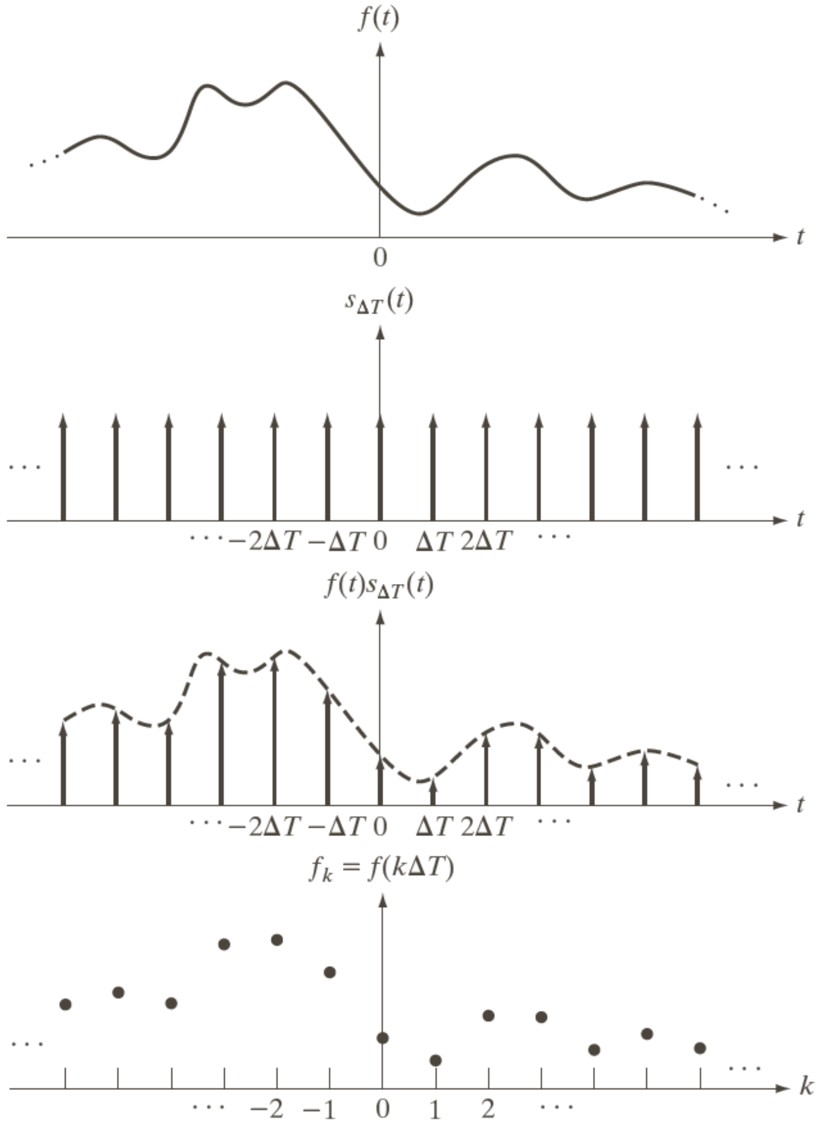


FIGURE 4.5
(a) A continuous function. (b) Train of impulses used to model the sampling process. (c) Sampled function formed as the product of (a) and (b). (d) Sample values obtained by integration and using the sifting property of the impulse. (The dashed line in (c) is shown for reference. It is not part of the data.)

1-D Discrete Fourier Transform

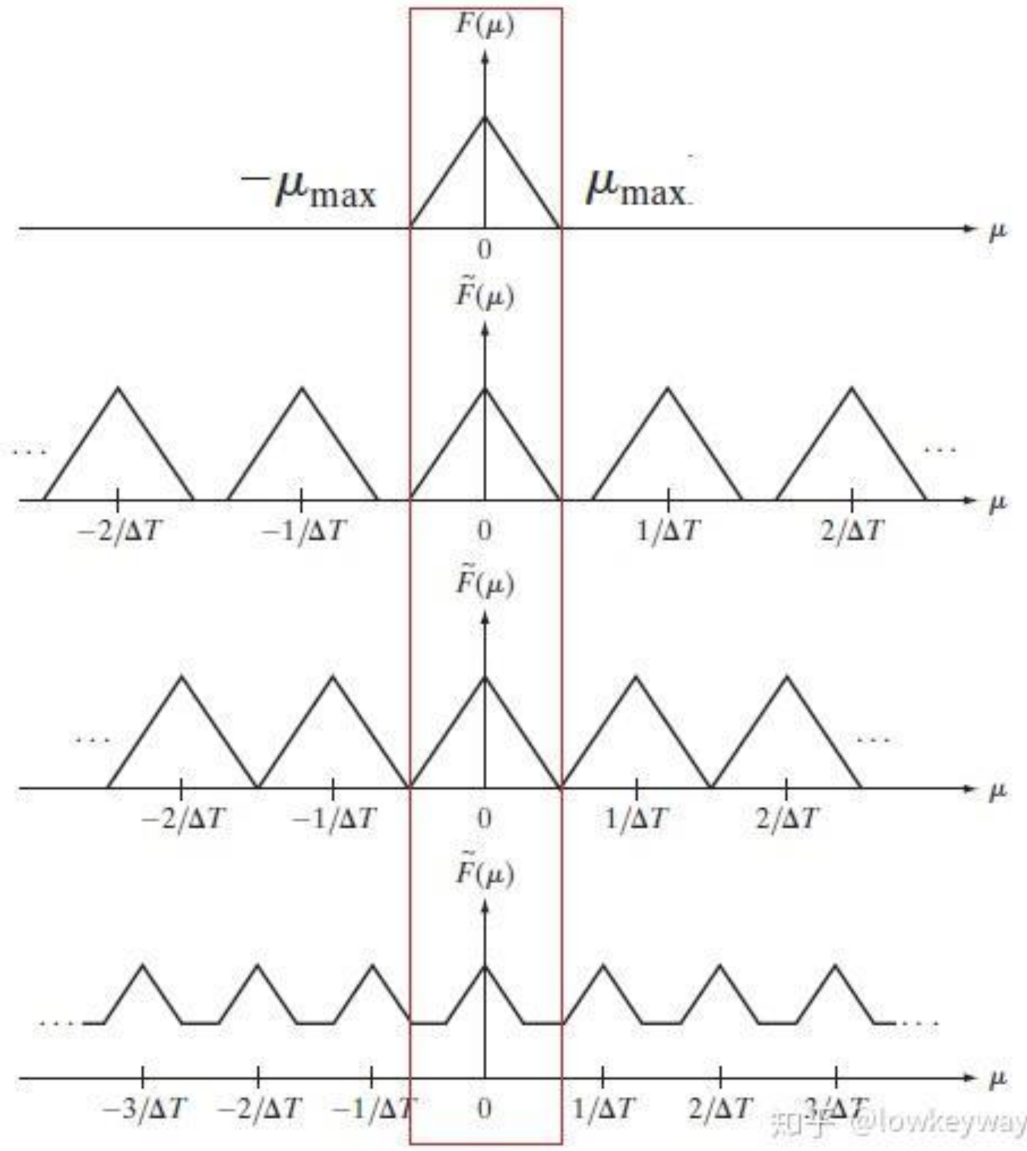
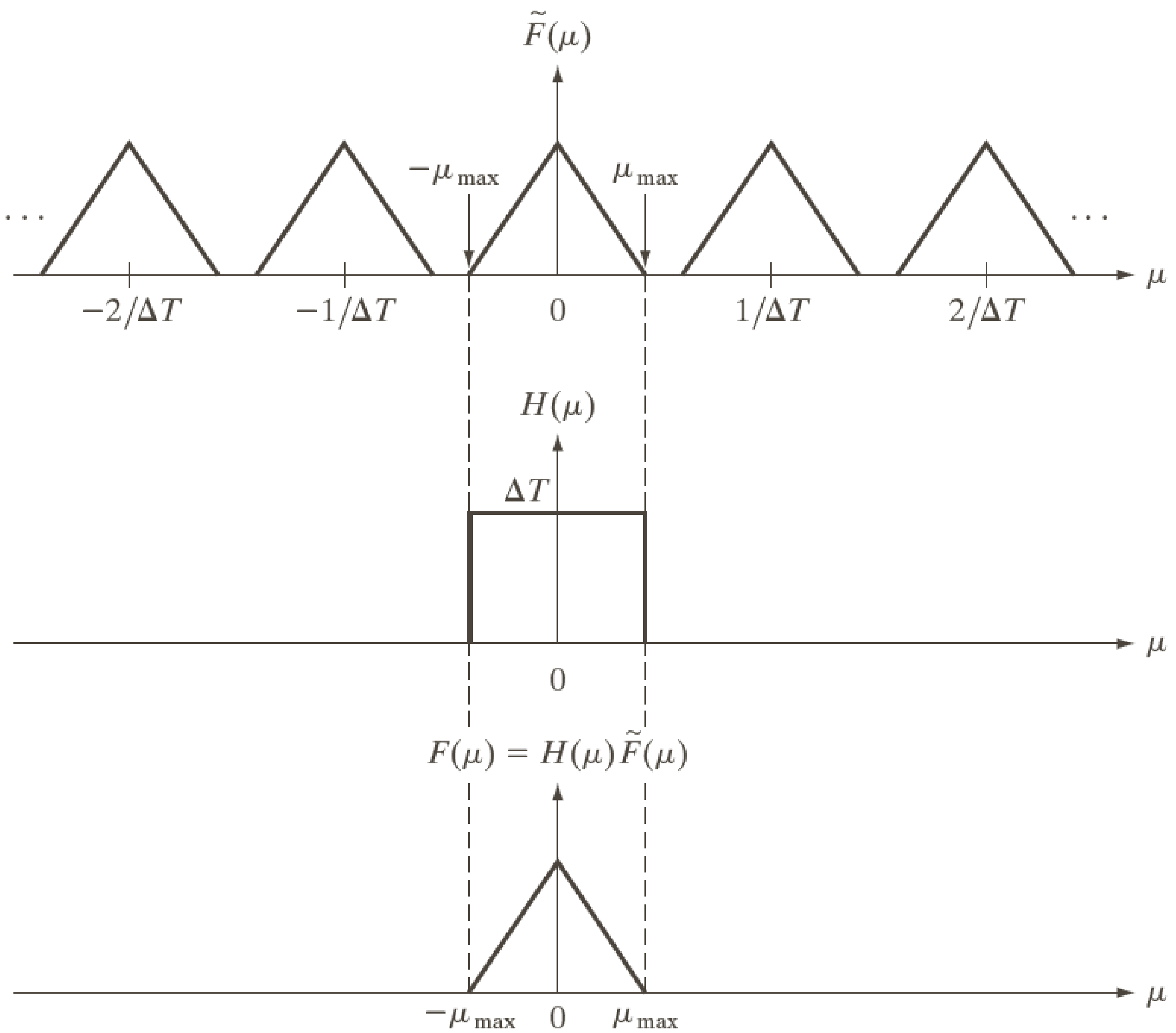


FIGURE 4.6
(a) Fourier transform of a band-limited function.
(b)–(d) Transforms of the corresponding sampled function under the conditions of over-sampling, critically-sampling, and under-sampling, respectively.

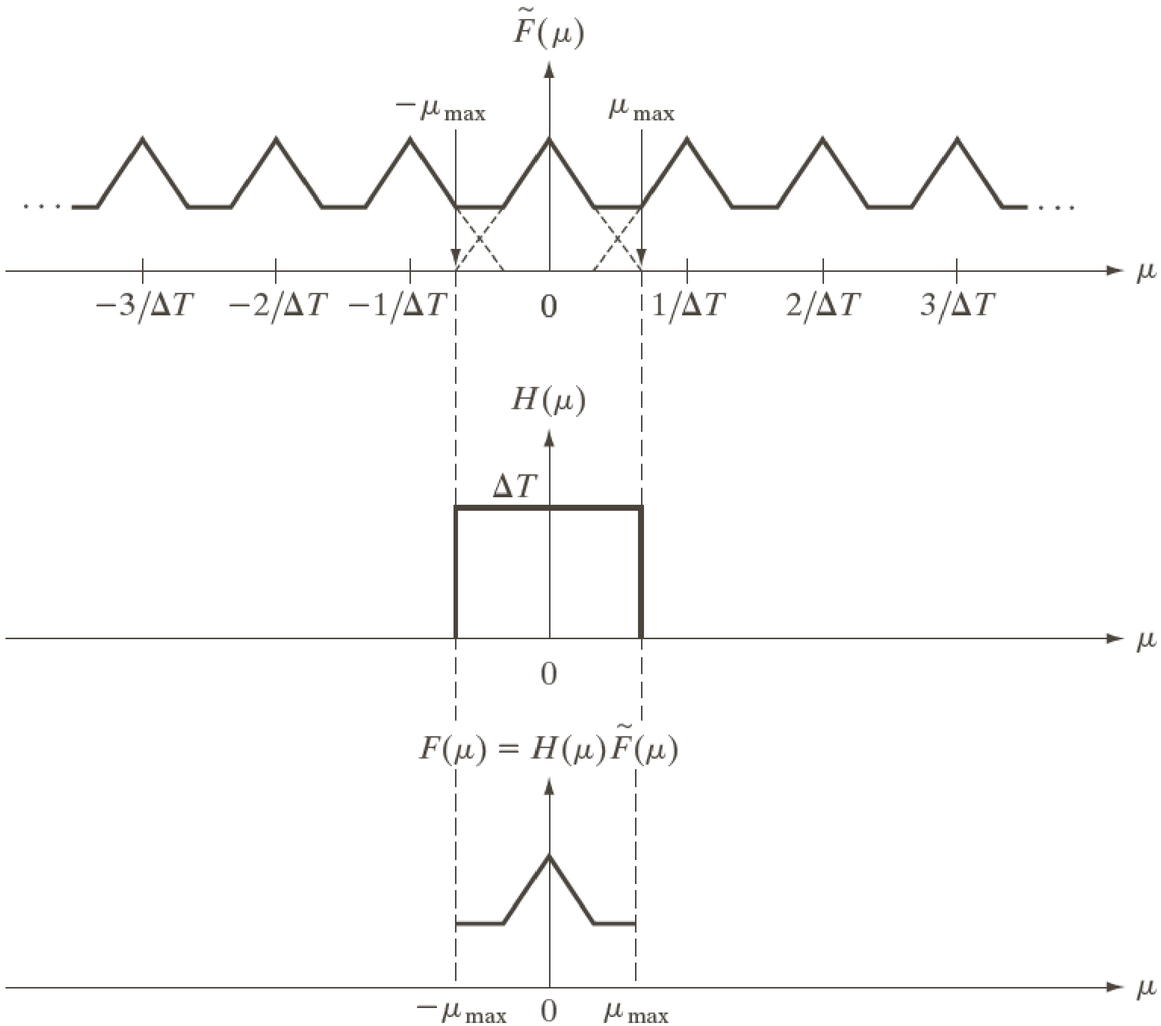
1-D Discrete Fourier Transform



a
b
c

FIGURE 4.8 Extracting one period of the transform of a band-limited function using an ideal lowpass filter.

1-D Discrete Fourier Transform



(a) Fourier transform of an under-sampled, band-limited function. (Interference between adjacent periods is shown dashed). (b) The same ideal lowpass filter used in Fig. 4.8 . (c) The product of (a) and (b).The interference from adjacent periods results in aliasing that prevents perfect recovery of $F(\mu)$ and, consequently, of $f(t)$.

1-D Discrete Fourier Transform

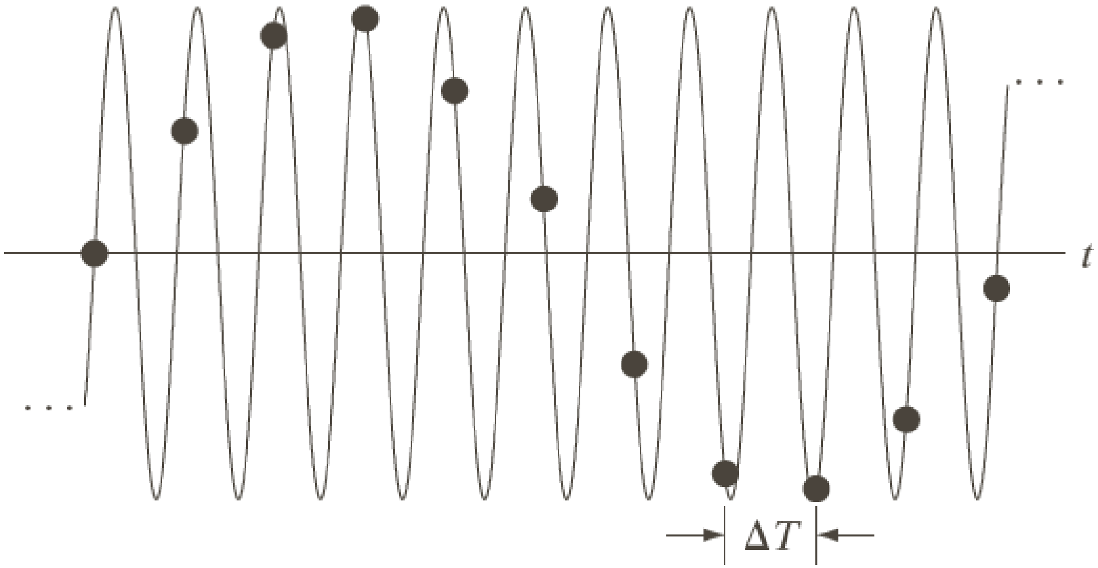


FIGURE 4.10 Illustration of aliasing. The under-sampled function (black dots) looks like a sine wave having a frequency much lower than the frequency of the continuous signal. The period of the sine wave is 2 s, so the zero crossings of the horizontal axis occur every second. ΔT is the separation between samples.

1-D Discrete Fourier Transform

a b c

FIGURE 4.17 Illustration of aliasing on resampled images. (a) A digital image with negligible visual aliasing. (b) Result of resizing the image to 50% of its original size by pixel deletion. Aliasing is clearly visible. (c) Result of blurring the image in (a) with a 3×3 averaging filter prior to resizing. The image is slightly more blurred than (b), but aliasing is not longer objectionable. (Original image courtesy of the Signal Compression Laboratory, University of California, Santa Barbara.)

2-D DFT

- **2D DFT** of an image f(x,y) of size M×N is given by

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

……(5)

for u=0,1,2,…,M-1, and v=0,1,2,…,N-1.

- Given F(u,v), we obtain f(x,y) by **inverse 2D DFT**

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

……(6)

for x=0,1,2,…,M-1, and y=0,1,2,…,N-1.

2-D DFT

- Fourier spectrum, phase angle, and power spectrum are defined as

Fourier spectrum:

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

Phase angle:

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

Power Spectrum:

$$\begin{aligned} P(u,v) &= |F(u,v)|^2 \\ &= R^2(u,v) + I^2(u,v) \end{aligned}$$

2-D DFT

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

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

  - From this, it follows that

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

  - Which says that the spectrum of the Fourier transform is symmetric

2-D DFT

- It is common practice to multiply the input image function by $(-1)^{x+y}$ prior to computing the Fourier transform

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

  - Multiplying f(x,y) by $(-1)^{x+y}$ shifts the origin of F(u,v) [that is, F(0,0)] to frequency coordinates (M/2,N/2), which is the center of the $M \times N$ area
  - When the limits of summations are from u=1 to M and v=1 to N, the actual center of the transform will then be at u=(M/2)+1 and v=(N/2)+1

2-D DFT

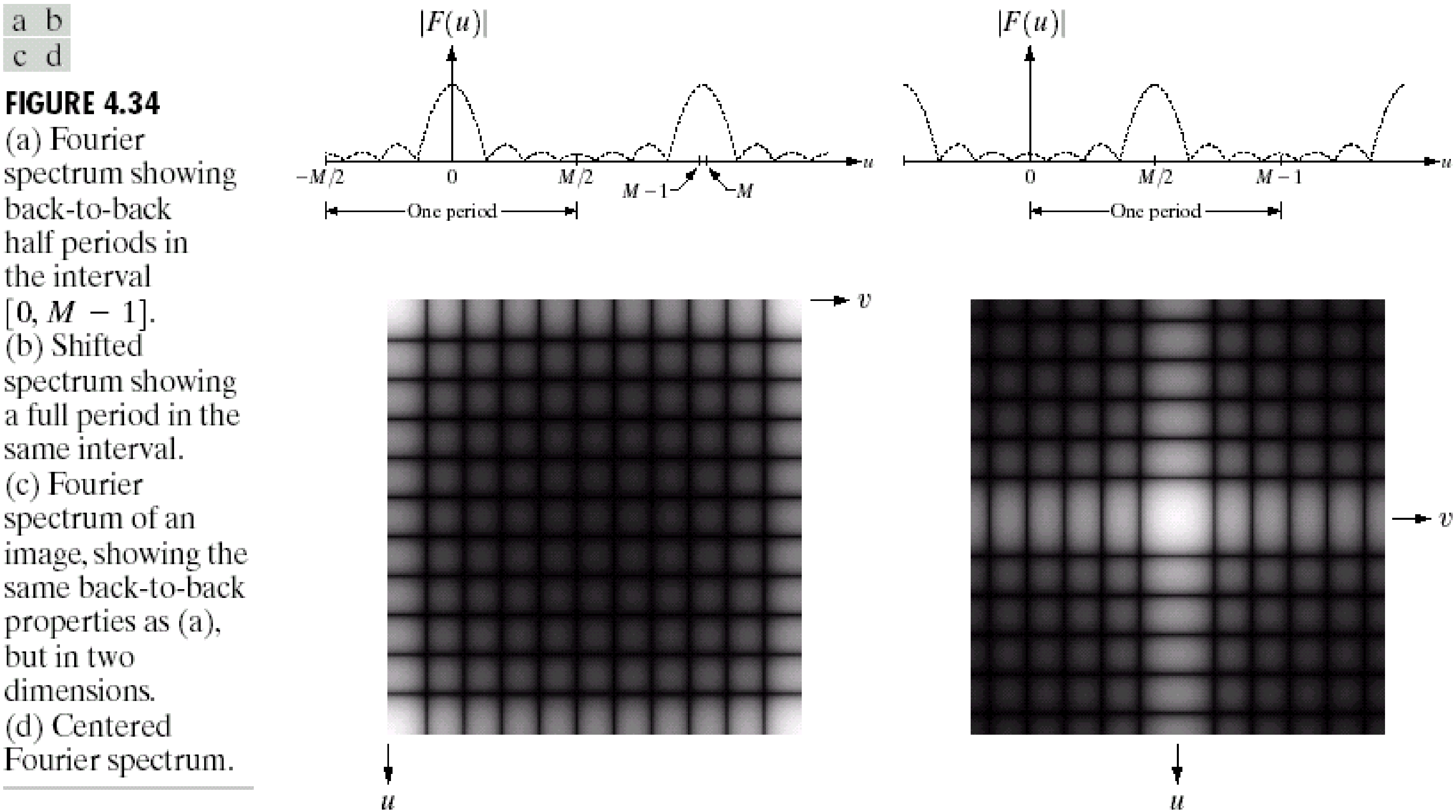


FIGURE 4.34 (a) Fourier spectrum showing back-to-back half periods in the interval $[0, M-1]$. (b) Shifted spectrum showing a full period in the same interval. (c) Fourier spectrum of an image, showing the same back-to-back properties as (a), but in two dimensions. (d) Centered Fourier spectrum.

2-D DFT

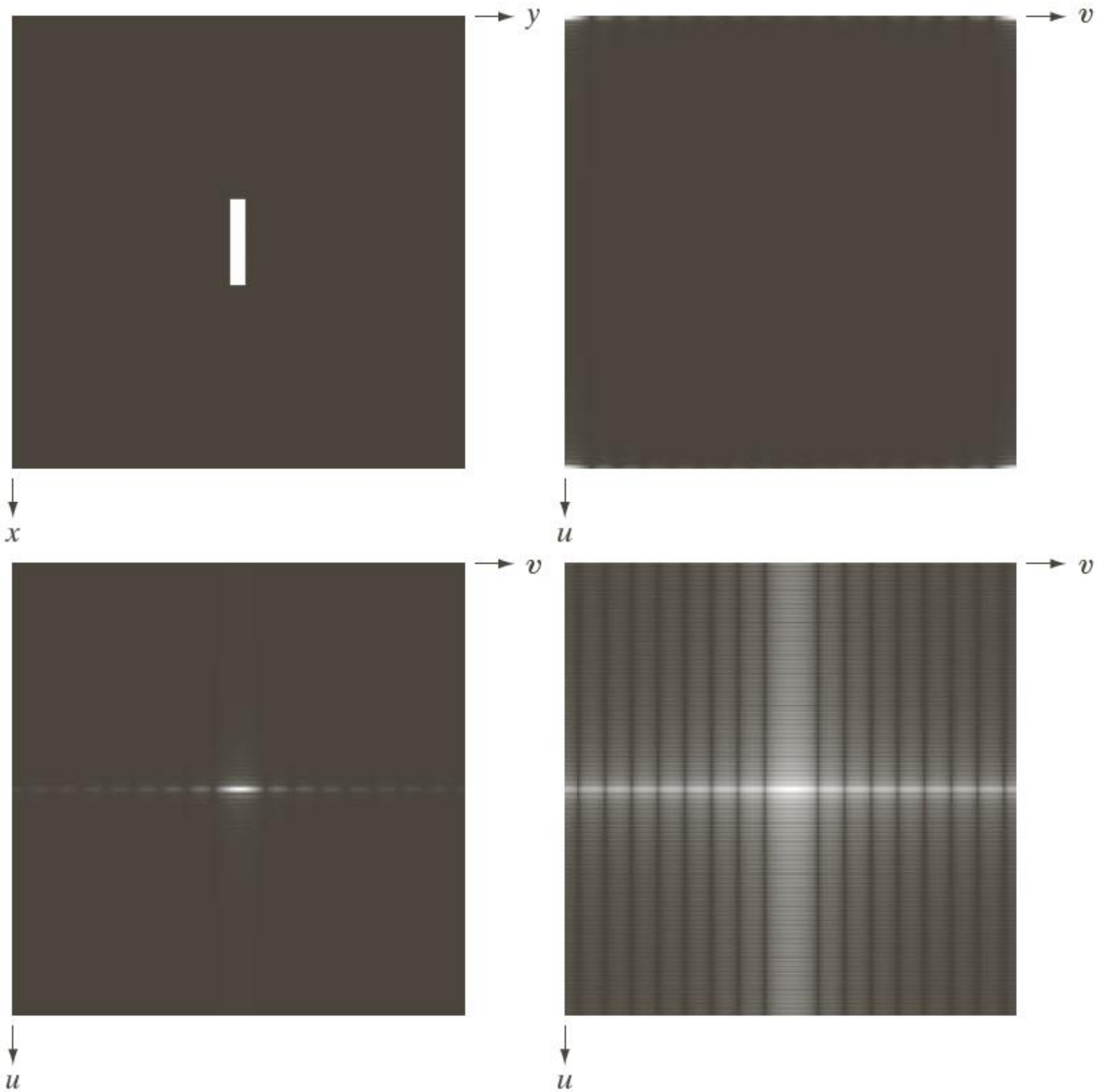


FIGURE 4.24
(a) Image.
(b) Spectrum showing bright spots in the four corners.
(c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate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

2-D DFT

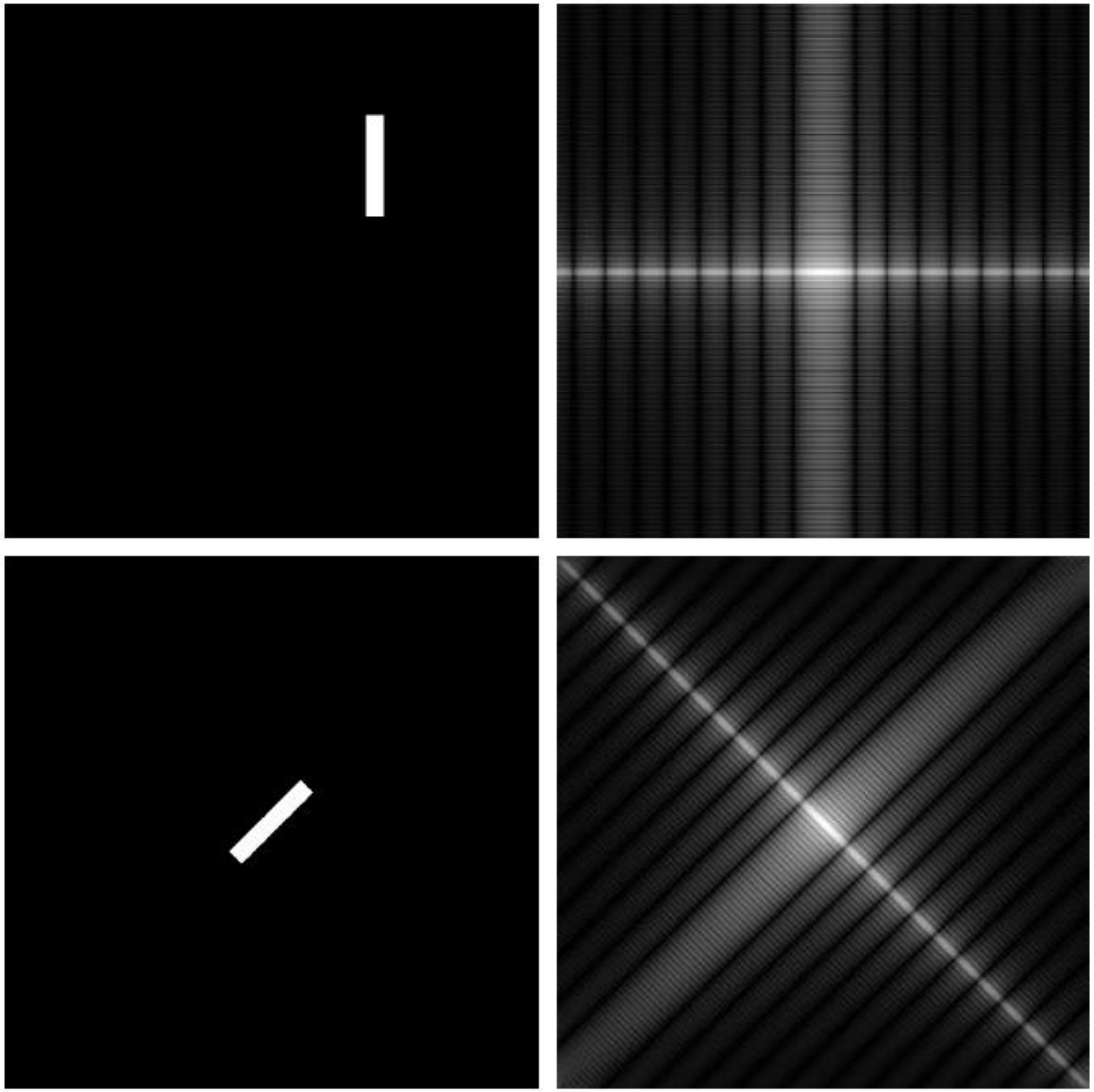
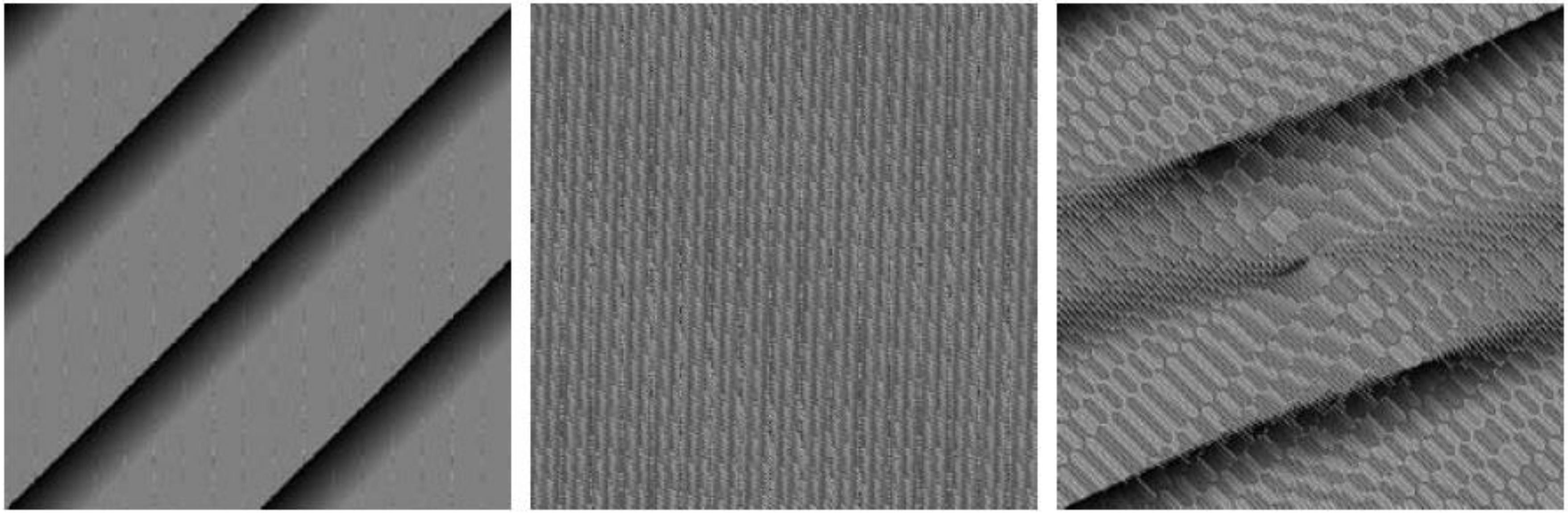


FIGURE 4.25 (a) The rectangle in Fig. 4.24(a) translated, and (b) the corresponding spectrum. (c) Rotated rectangle, and (d) the corresponding spectrum. The spectrum corresponding to the translated rectangle is identical to the spectrum corresponding to the original image in Fig. 4.24(a).

2-D DFT



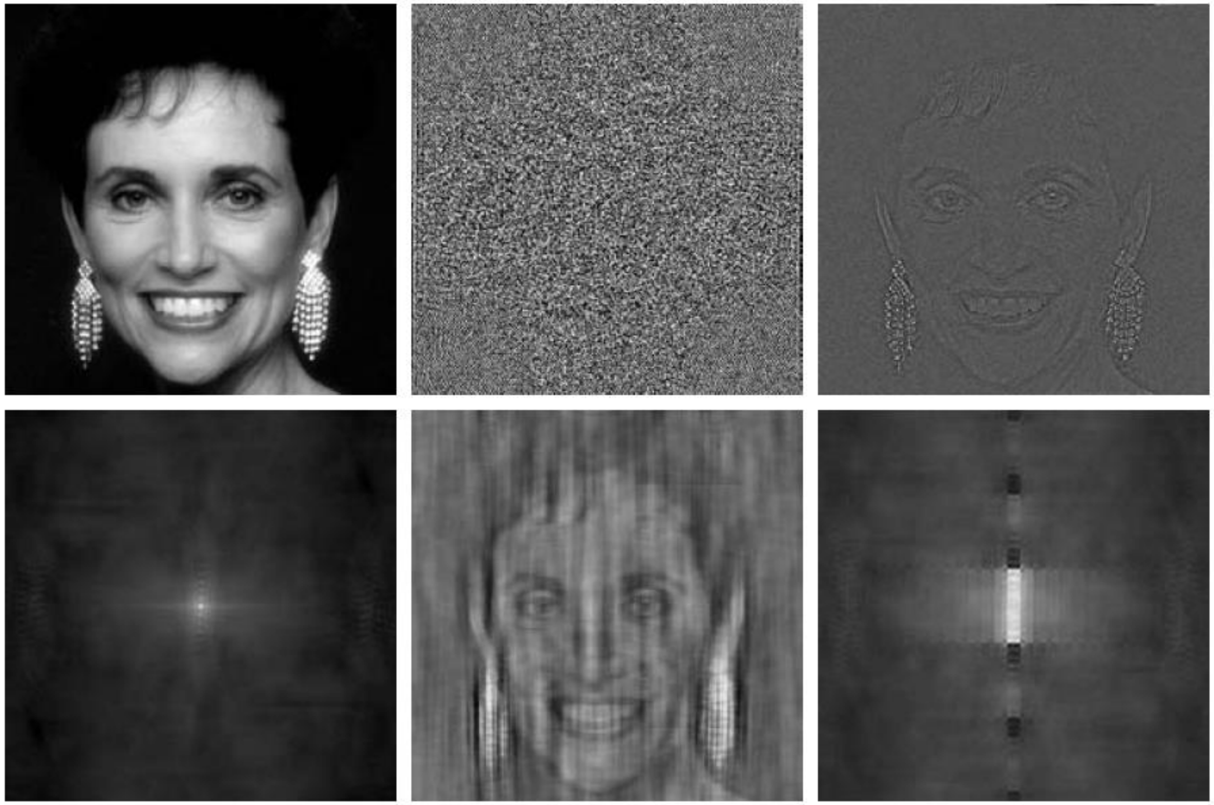
a b c

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).

2-D DFT

- Questions
  - Which has more information, the phase or the magnitude?
  - What happens if you take the phase from one image and combine it with the magnitude from another image?

2-D DFT



a b c
d e f

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 DFT

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

2-D DFT

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	$\Im\left[f_1(x,y) + f_2(x,y)\right] = \Im\left[f_1(x,y)\right] + \Im\left[f_2(x,y)\right]$ $\Im\left[f_1(x,y)\cdot f_2(x,y)\right] \neq \Im\left[f_1(x,y)\right]\cdot\Im\left[f_2(x,y)\right]$
Scaling	$af(x,y) \Leftrightarrow aF(u,v),\ f(ax,by) \Leftrightarrow \dfrac{1}{\lvert ab\rvert} F(u/a, v/b)$
Rotation	$x = r\cos\theta \quad y = r\sin\theta \quad u = \omega\cos\varphi \quad v = \omega\sin\varphi$ $f(r, \theta+\theta_0) \Leftrightarrow F(\omega, \varphi+\theta_0)$
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.

2-D DFT

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

2-D DFT

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 $\text{rect}[a, b] \Leftrightarrow ab\dfrac{\sin(\pi ua)}{(\pi ua)}\dfrac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$

Cosine $\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$

$$\frac{1}{2}\left[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)\right]$$

Sine $\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$

$$j\frac{1}{2}\left[\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)\right]$$

† Assumes that functions have been extended by zero padding.

Convolution Theorem

- Objective: establish a direct link between some of those spatial filters and their frequency domain counterparts

- The most fundamental relationship between the spatial and frequency domains is established by *convolution theorem*

Convolution Theorem

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

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

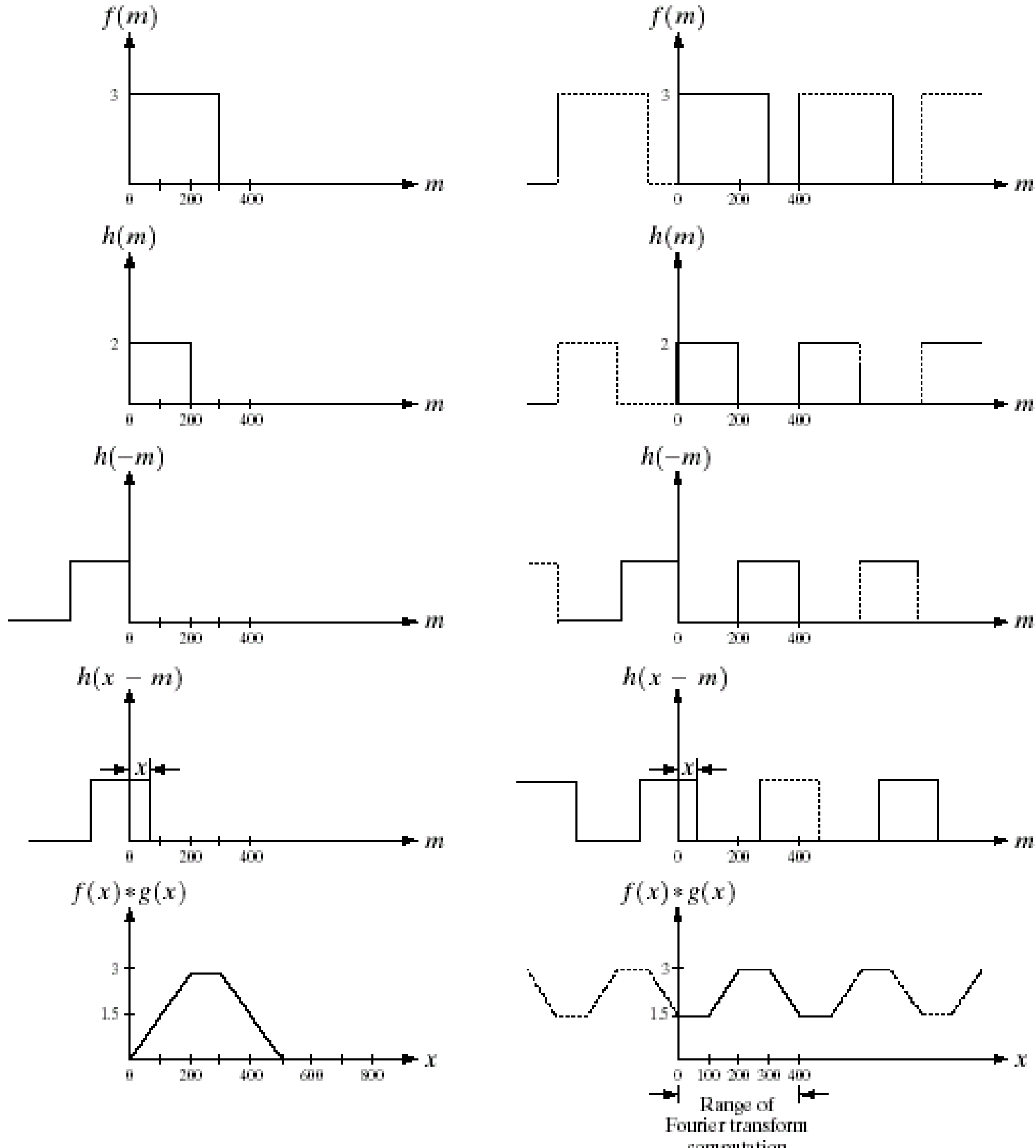
  - The minus sigh simply mean that function h(x,y) is mirrored about the origin

Convolution Theorem

- The definition of convolution is really nothing more than an implementation for
  - (1) flipping one function about the origin
  - (2) shifting that function with respect to the other by changing the values of (x,y)
  - (3) computing a sum of products over all values of *m* and *n*, for each displacement (x,y).
    - the displacements (x,y) are integer increments that stop when the functions no longer overlap

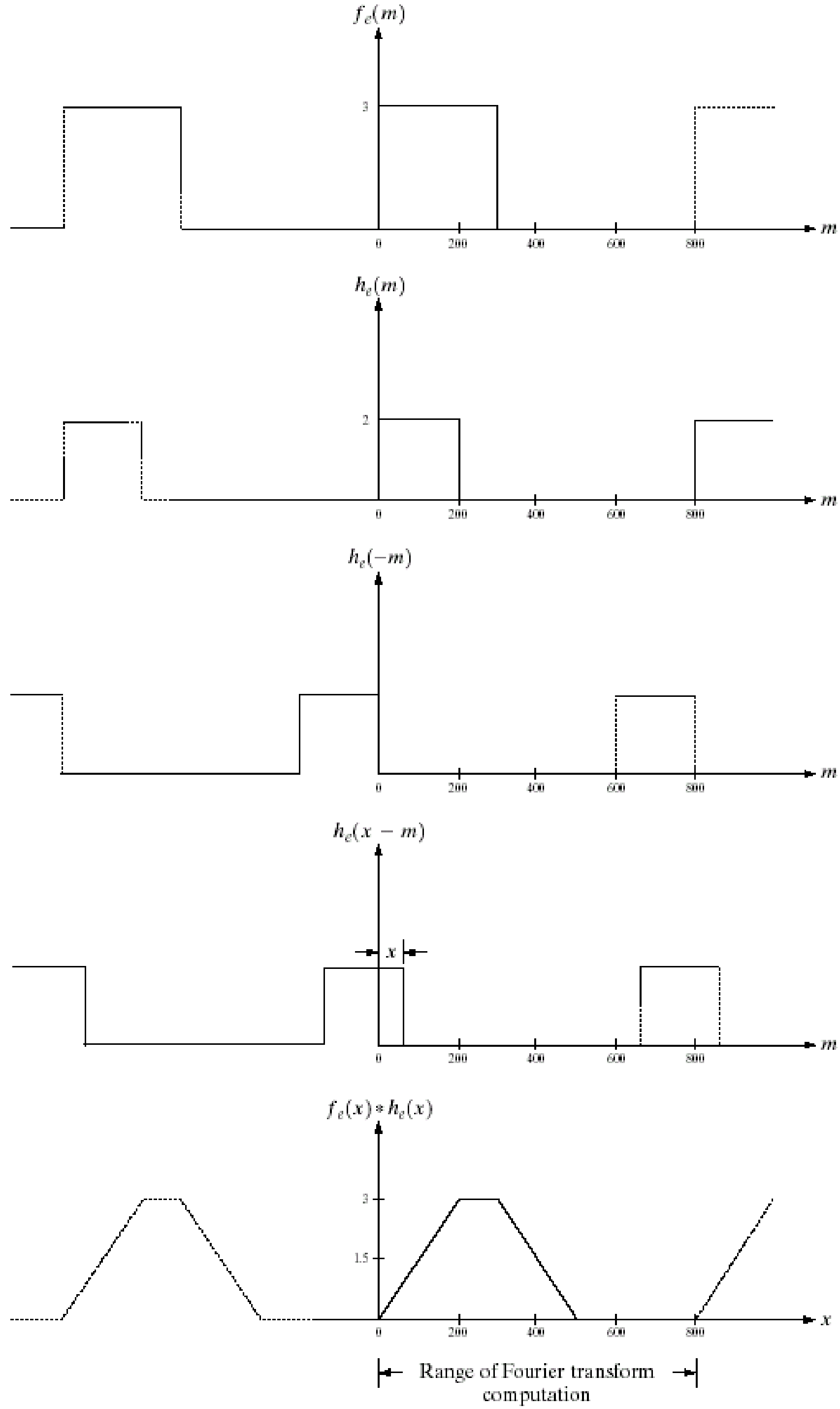
a	f
b	g
c	h
d	i
e	j

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.



a
b
c
d
e

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



Convolution Theorem

- Convolution theorem
  - Let F(u,v) and H(u,v) denote the Fourier transforms of f(x,y) and h(x,y) respectively.
  - One-half of convolution theorem simply states that f(x,y)*h(x,y) and F(u,v)H(u,v) constitute a Fourier transform pair

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

Convolution Theorem

- Filters in the spatial and frequency domains constitute a Fourier transform pair

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

  - We can specify filters in the frequency domain, take their inverse transform, and then use the resulting filter in the spatial domain as a guide for constructing smaller spatial filter masks

Contents

- Background
- Introduction to the Fourier Transform and the Frequency Domain
  - 1D DFT
  - 2D DFT
- Smoothing Frequency Domain Filters
- Sharpening Frequency Domain Filters

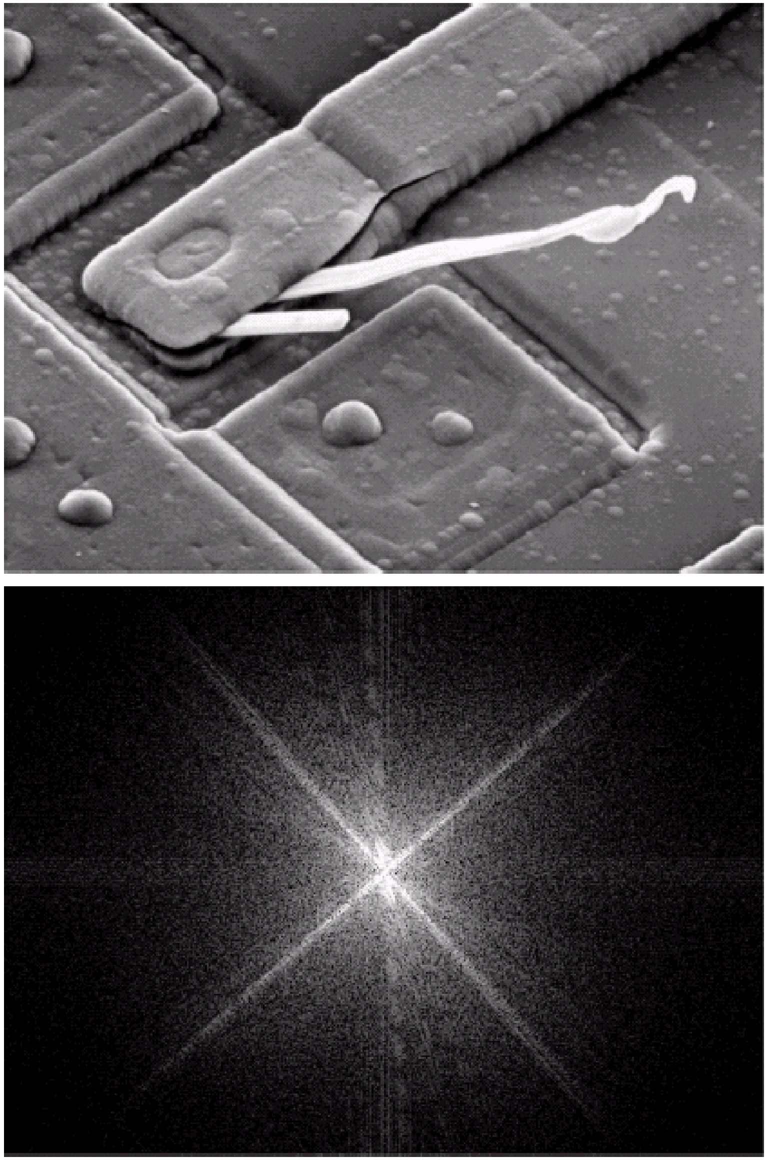
Basics of Filtering in Frequency Domain

- Some basic properties of the frequency domain
  - Some general statements can be made about the relationship between the frequency components of the Fourier transform and the spatial characteristics of an image
  - It is not difficult intuitively to *associate the frequencies in the Fourier transform with patterns of intensity variations in an image*, since frequency is directly related to rate of change

Basics of Filtering in Frequency Domain

- ***The slowest varying frequency component (u=0, v=0)*** corresponds to the average gray level of an image
- ***The low frequencies*** correspond to slowly varying components of an image
  - In an image of a room, for example, smooth gray-level variations on the walls and floor
- ***The higher frequencies*** correspond to faster gray level changes in the image
  - Such as edges and noise

Basics of Filtering in Frequency Domain



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.)

Frequency Domain Filtering Fundamentals

- H(u,v) is called a *filter*, or *filter transfer function*
  - Because it suppresses certain frequencies in the transform while leaving others unchanged

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

  - Element-by-element multiplication
  - H(u,v) typically are real (*zero-phase-shift filters*)

Frequency Domain Filtering Fundamentals

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

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

The final image is obtained by:

- 1). taking the real part of this result and
- 2). multiplying it by $(-1)^{x+y}$

Frequency Domain Filtering Fundamentals

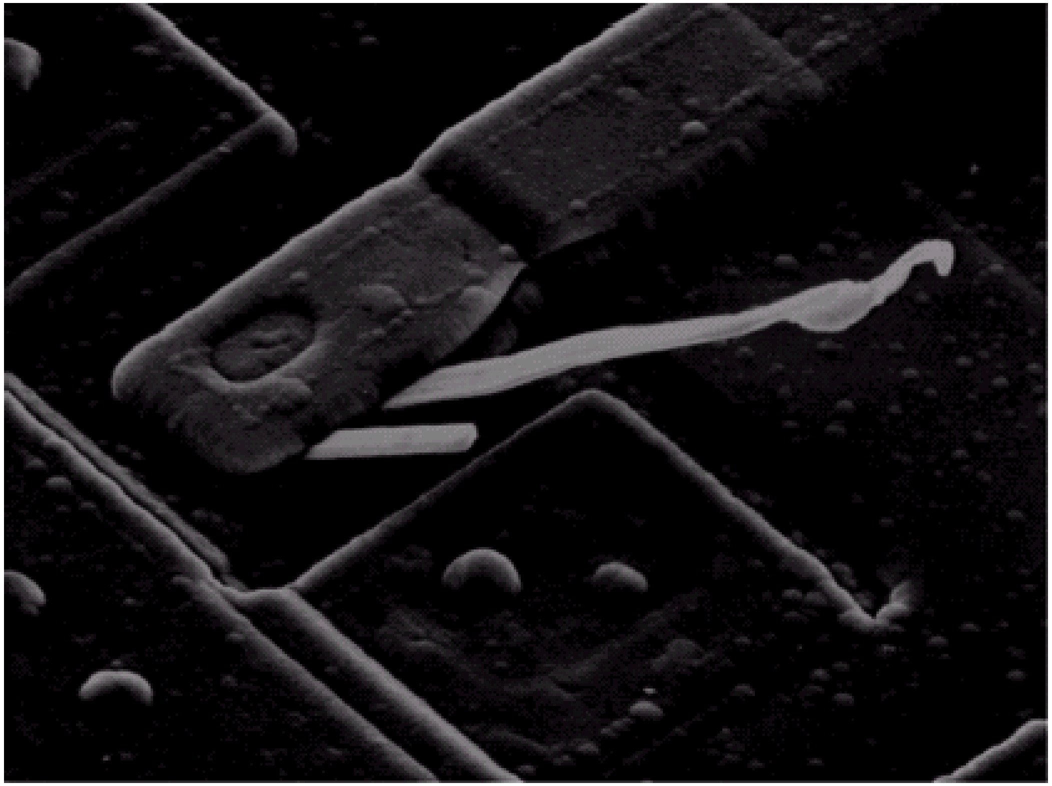
- Some basic filters and their properties
- *Notch filter*

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

  - Set F(0,0) to zero and leave all other frequency components of the Fourier transform untouched
    - Force the average value to zero
    - Make prominent edges stand out (byproduct)

Frequency Domain Filtering Fundamentals

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.



Frequency Domain Filtering Fundamentals

- *Lowpass filter*
  - Attenuates high frequencies while “passing” low frequencies
  - Filtered image will show less sharp detail

- *Highpass filter*
  - Attenuates low frequencies while “passing” high frequencies
  - Filtered image will appear sharper

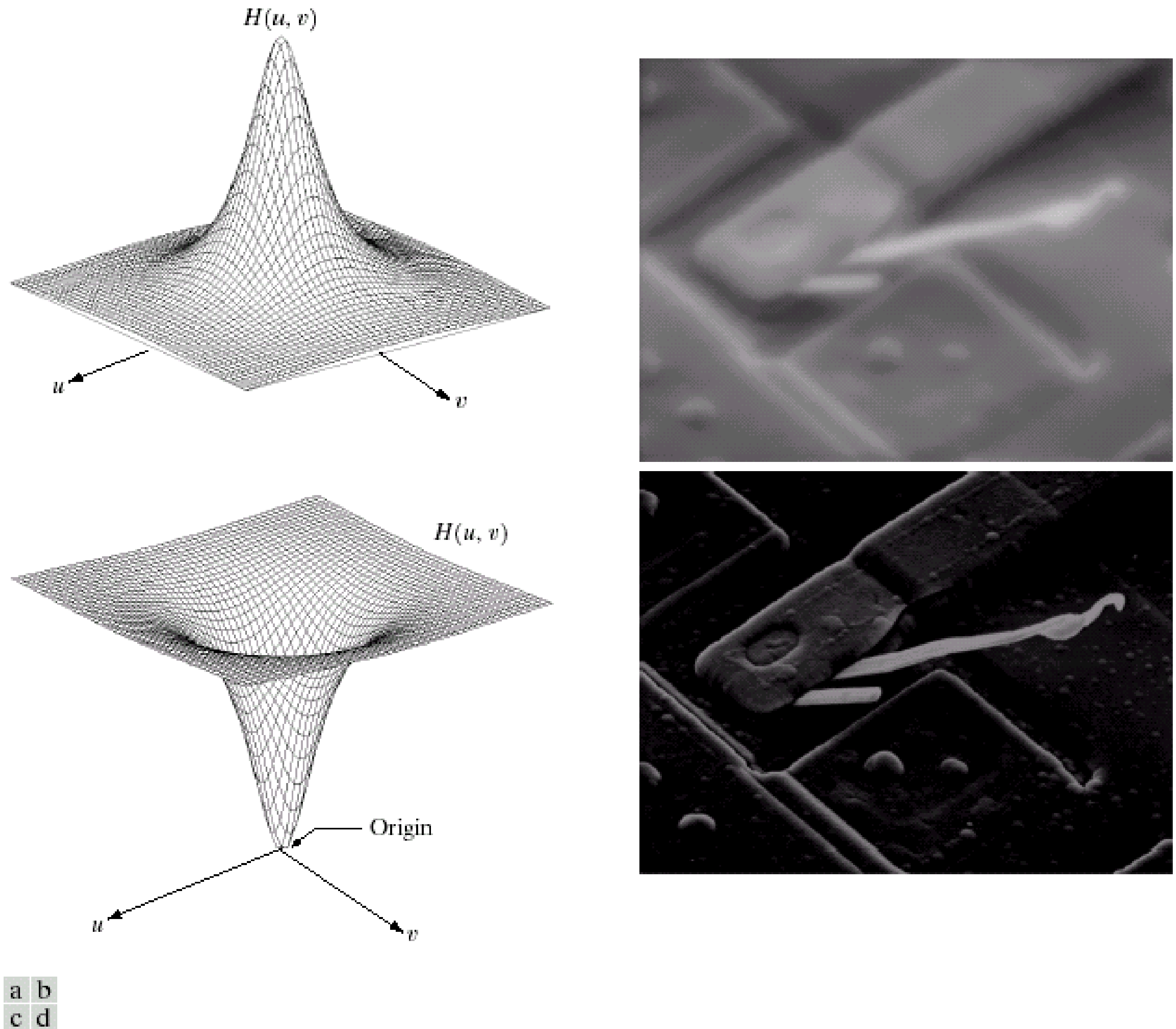


FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

Frequency Domain Filtering Fundamentals

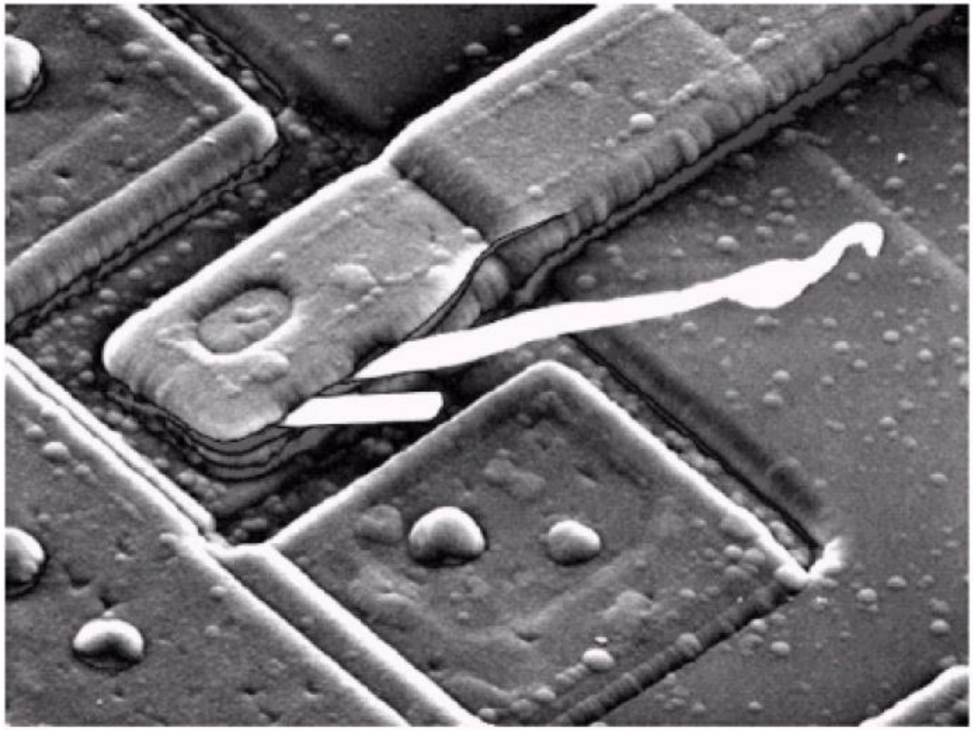


FIGURE 4.8 Result of highpass filtering the image in Fig. 4.4(a) with the filter in Fig. 4.7(c), modified by adding a constant of one-half the filter height to the filter function. Compare with Fig. 4.4(a).

Steps for Filtering in the Frequency Domain

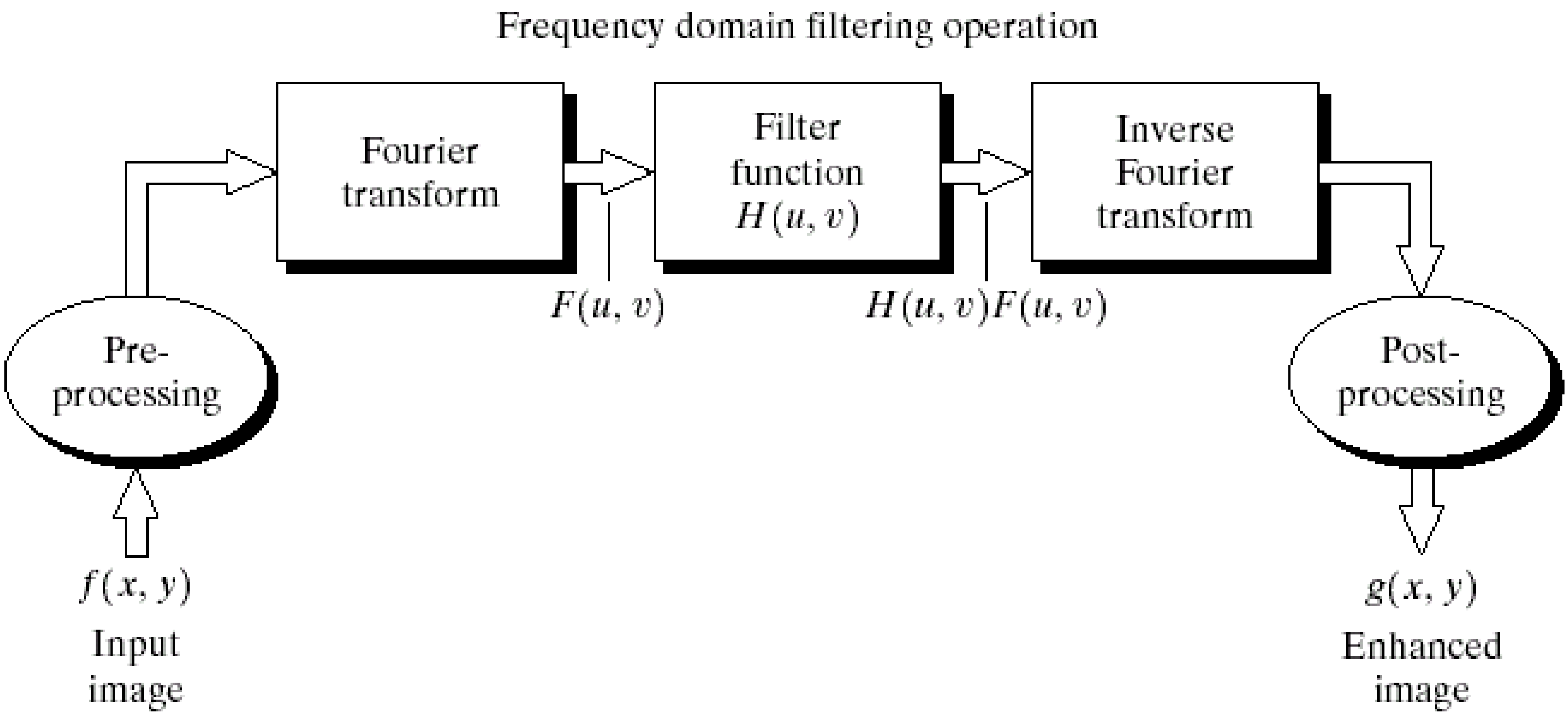


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Steps for Filtering in the Frequency Domain

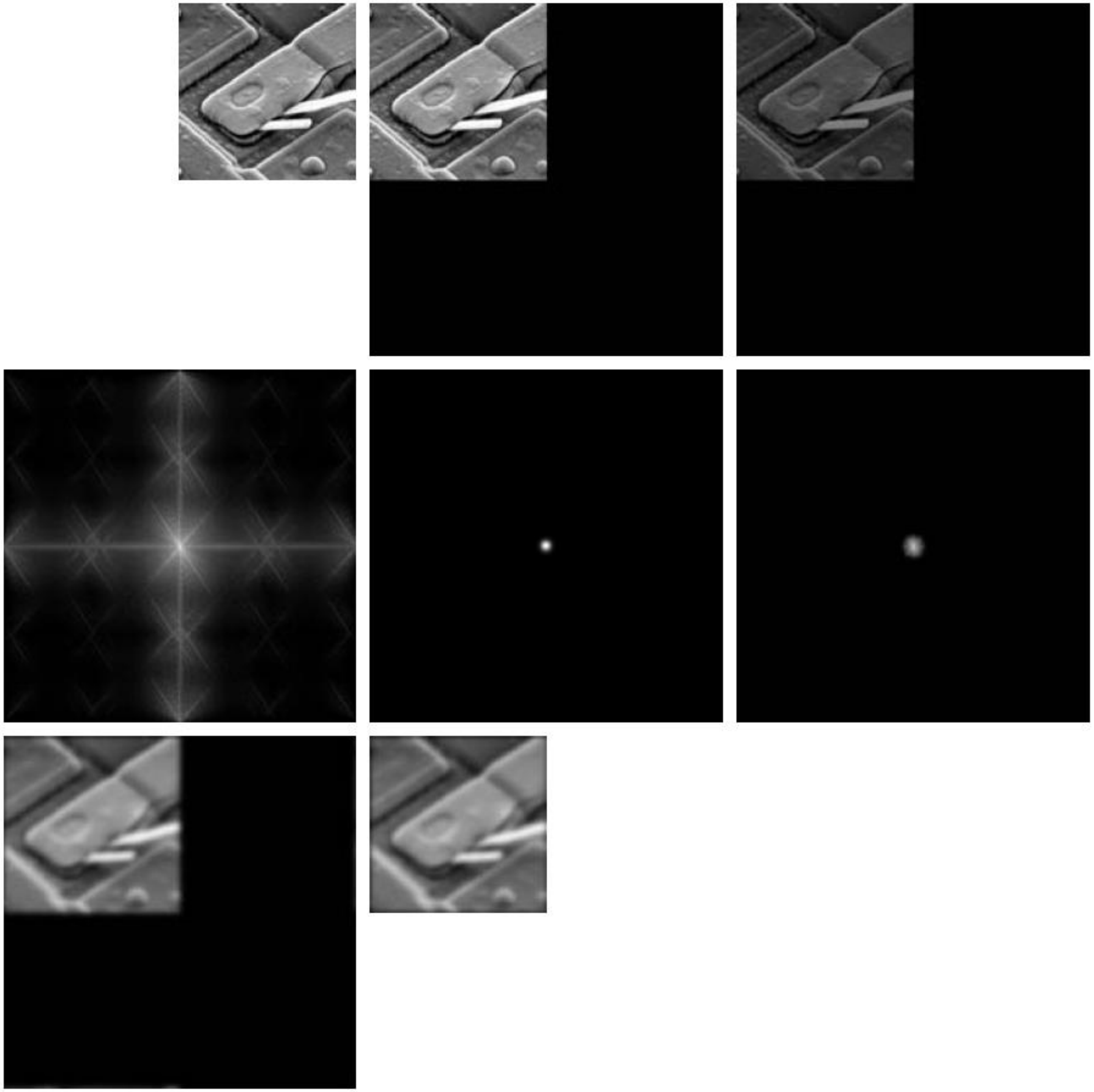
- **Summary of steps for filtering in Frequency domain**
  - 1. Given an input image $f(x,y)$ of size $M \times N$, form a padded image $f_p(x,y)$ of size $P \times Q$. Typically, we select $P=2M$ and $Q=2N$
  - 2. Multiply $f_p(x,y)$ by $(-1)^{x+y}$ to center its transform
  - 3. Compute $F(u,v)$, the DFT of the image from step 2

Steps for Filtering in the Frequency Domain

- 4. Generate a real, symmetric filter function, $H(u,v)$, of size $P \times Q$ with center at $(P/2, Q/2)$. Form the product $G(u,v) = H(u,v)F(u,v)$
- 5. Obtain the processed image:

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

- 6. Obtain the final processed result, $g(x,y)$, by extracting the $M \times N$ region from the top left quadrant of $g_p(x,y)$



a b c
d e f
g h

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.

Filtering in the Frequency Domain

- In conclusion, filtering in the frequency domain is based on
  - modifying the transform of an image in some way via a filter function,
  - and then taking the inverse of the result to obtain the processed output image

Correspondence between Filtering in the Spatial and Frequency Domains

- Filters based on Gaussian functions are of particular importance
  - Their shapes are easily specified (standard deviation *σ*)
  - Both the forward and inverse Fourier transforms of a Gaussian fu unctions

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

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

  - When H(u) has abroad profile (large value of *σ*), h(x) has a narrow profile, and vice versa

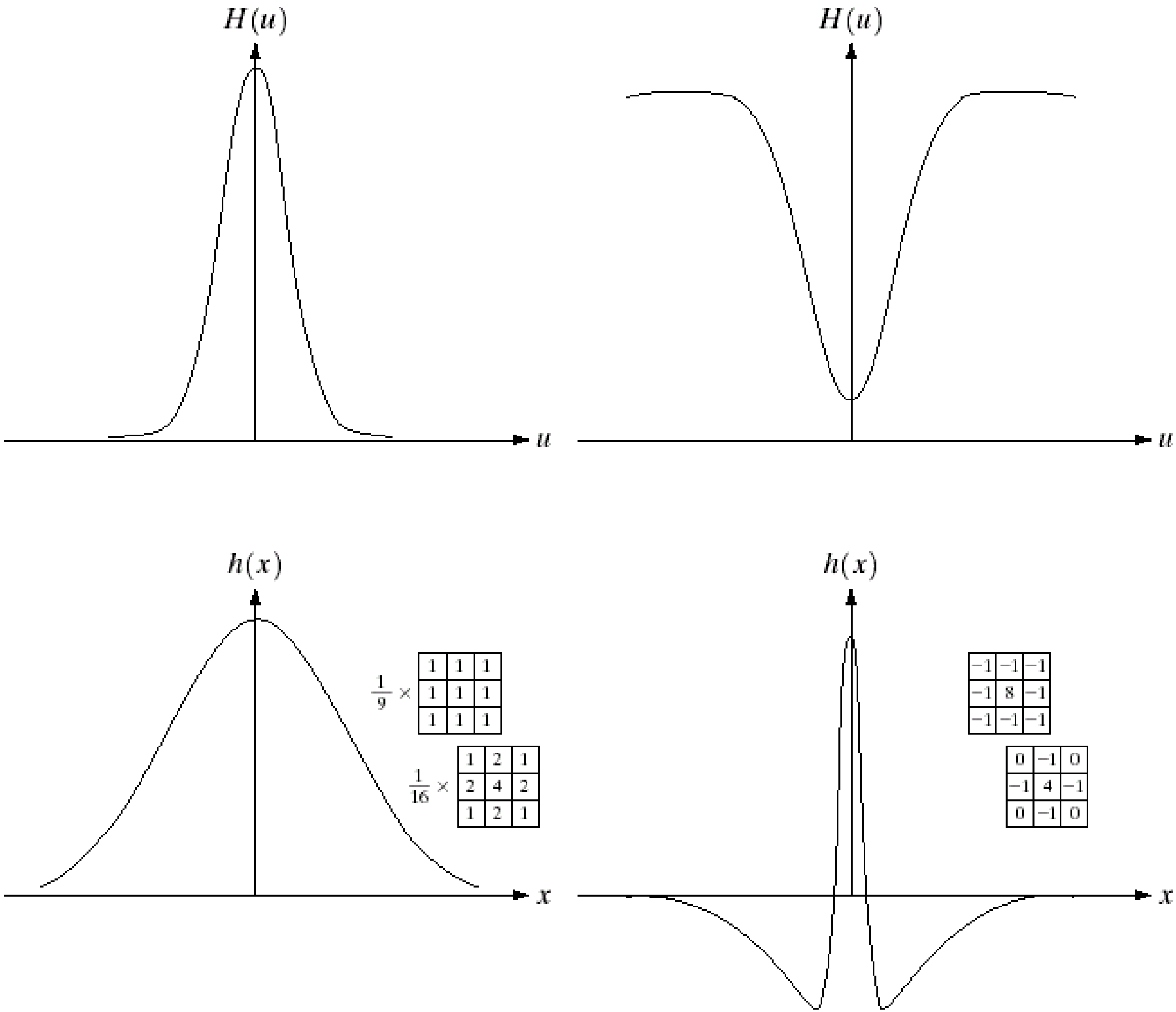
Correspondence between Filtering in the Spatial and Frequency Domains

- We can construct a highpass filter as a difference of Gaussian (DOG), as follows:

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

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

where A≥B and $\sigma_1 > \sigma_2$.



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.

Correspondence between Filtering in the Spatial and Frequency Domains

- Importance of frequency domain
  - **Significant degree of intuitiveness regarding how to specify filters**
  - A guide for constructing smaller spatial masks
  - **Some enhancement tasks that would be exceptionally difficult or impossible to formulate directly in the spatial domain become almost trivial in the frequency domain**

Smoothing Frequency-Domain Filters

- Smoothing (blurring) is achieved by **attenuating a specified range of high-frequency components** in Fourier transform
  - Edges and other sharp transitions (such as noise) contribute significantly to the high-frequency content
- Basic “model” for filtering in the frequency domain

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

Smoothing Frequency-Domain Filters

- Consider three types of lowpass filters, which cover the range of from very sharp to very smooth filter functions
  - Ideal (very sharp)
  - Butterworth (has a parameter called ***filter order***)
  - Gaussian (very smooth)

Smoothing Frequency-Domain Filters

- **Ideal Lowpass Filters**
  - 2-D ideal lowpass filter (ILPF) has the transfer function

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

where D_0 is a specified nonnegative quantity, and $D(u,v)$ is the distance from point (u,v) to the origin of the frequency rectangle

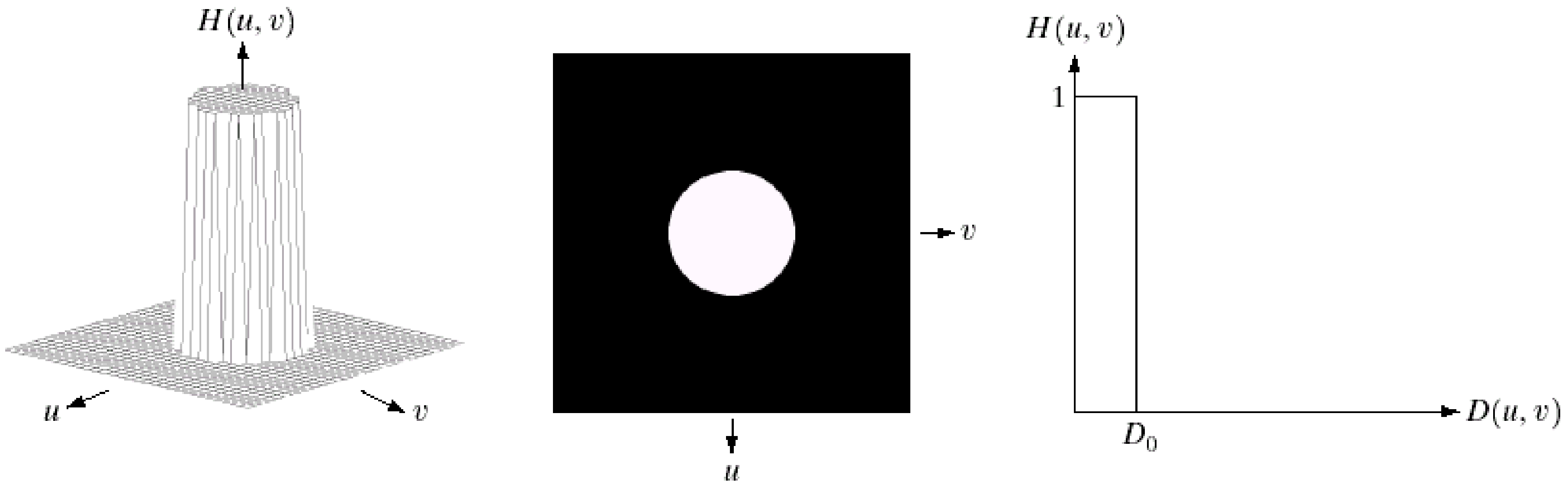
Smoothing Frequency-Domain Filters

- If the size of padded image is M$\times$N
  - The center of the frequency rectangle is at (u, v) = (M/2, N/2) due to the fact that the transform has been centered.
  - In this case, the distance from any point (u,v) to the center (origin) of the Fourier transform is:

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

- Smoothing Frequency-Domain Filters

- D_0 is called ***cutoff frequency***: the point of transition between H(u,v)=1 and H(u,v)=0.



a b c

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

Smoothing Frequency-Domain Filters

- One way to establish a set of standard cutoff frequency loci is to compute circles that enclose specified amount of total image power P_T

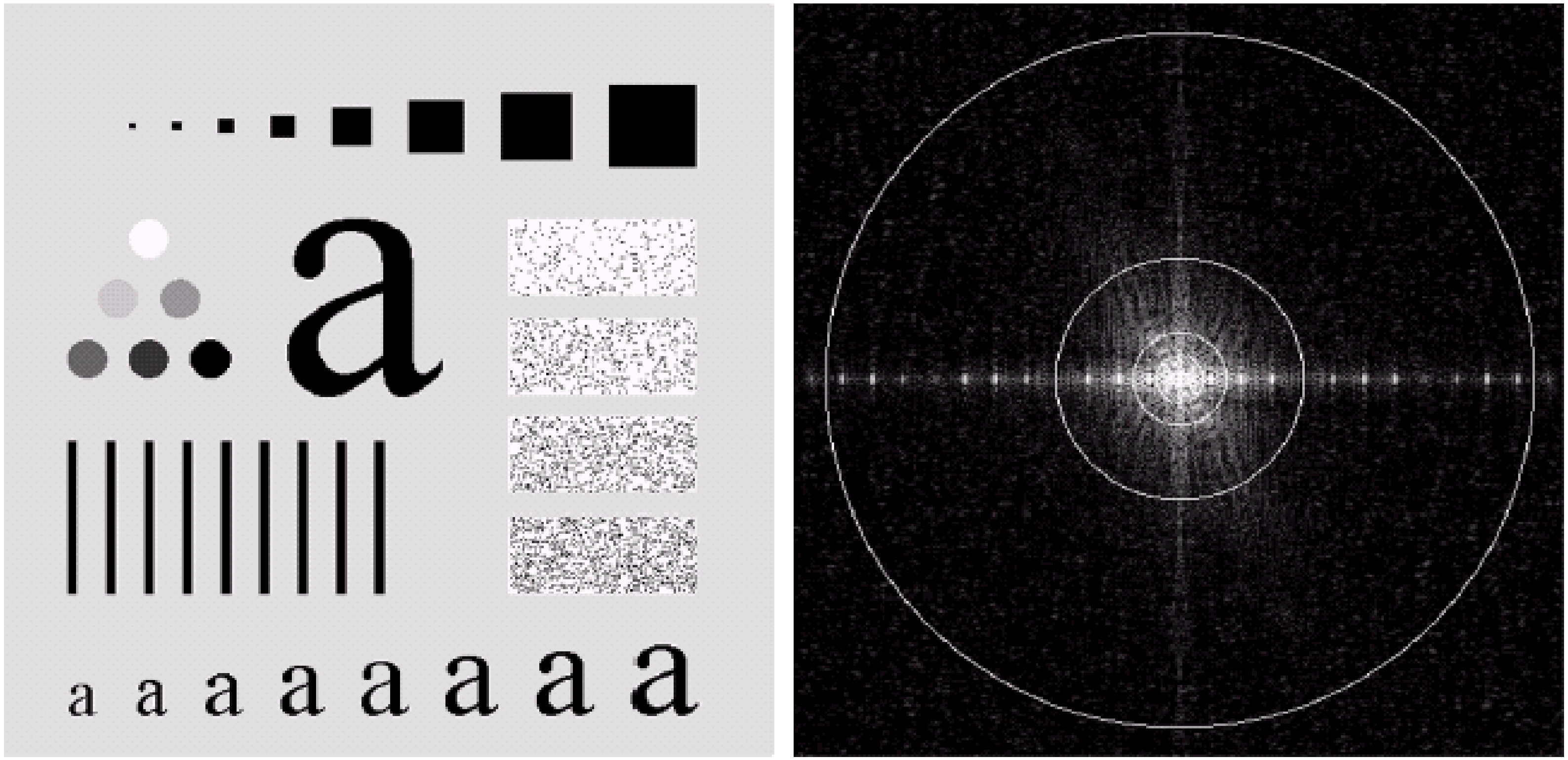
$$P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u,v)$$

  - α percent of the power

$$\alpha = 100\left[\sum_u \sum_v P(u,v) / P_T\right]$$

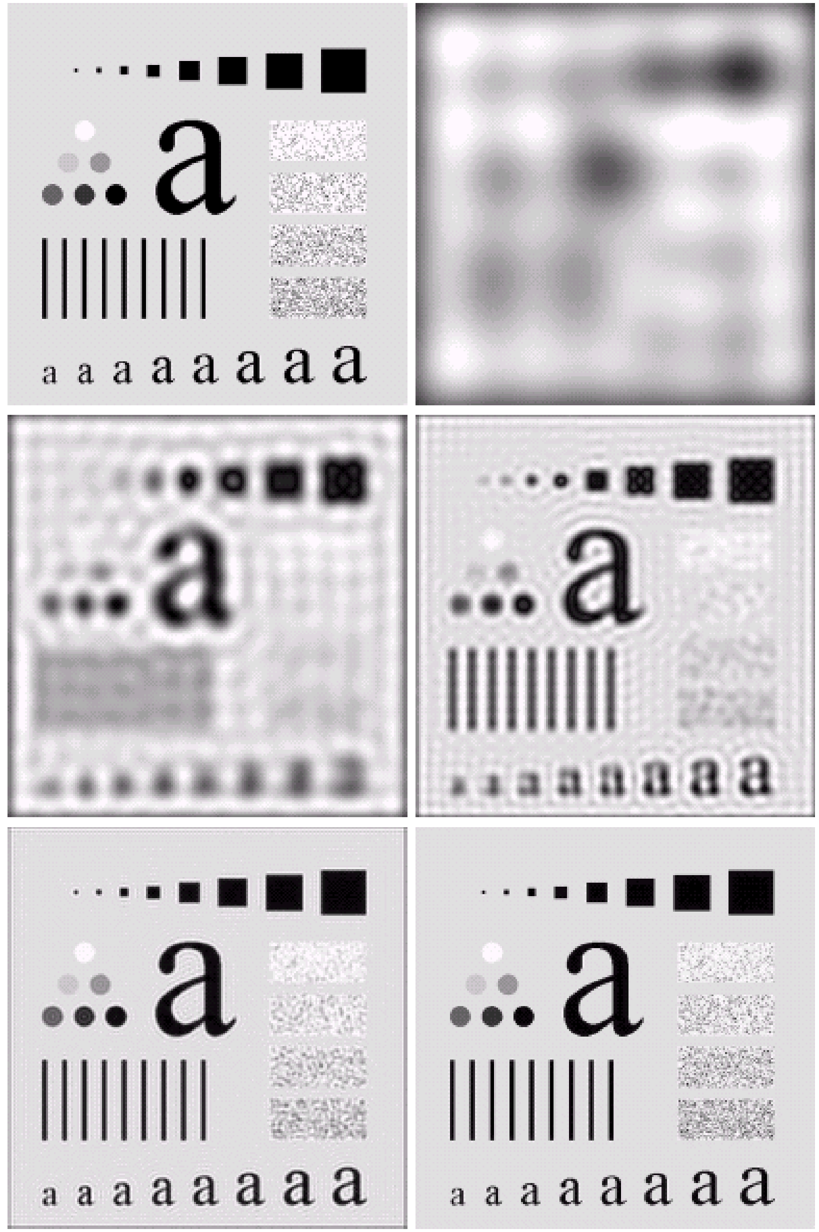
  - Summation is taken over the values of (u,v) that lie inside the circle or on its boundary

- Smoothing Frequency-Domain 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.



a b
c d
e f

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.

- The sharp cutoff frequencies of an ideal lowpass filter cannot be realized with electronic components (**“nonphysical” filters**)
- However, because ideal filters can be implemented in a computer, it is useful to study their behavior as part of our development of filtering concepts
- **The blurring and ringing properties of the ILPF can be explained by reference to the convolution theorem**

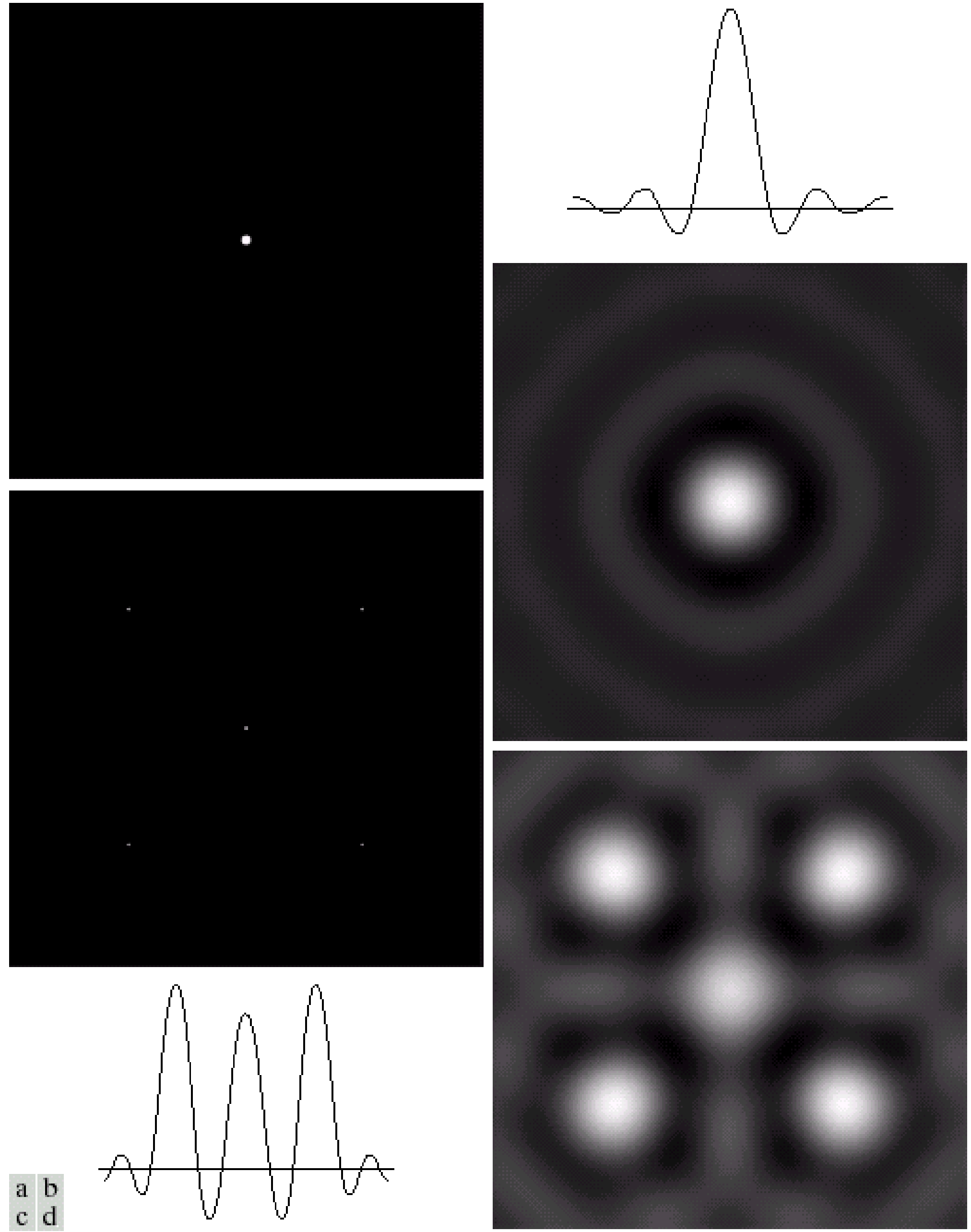


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

Smoothing Frequency-Domain Filters

- **Butterworth Lowpass Filters**
  - The transfer function of a Butterworth lowpass filter (BLPF) of order *n*, and with cutoff frequency at a distance D_0 from the origin, is defined as

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

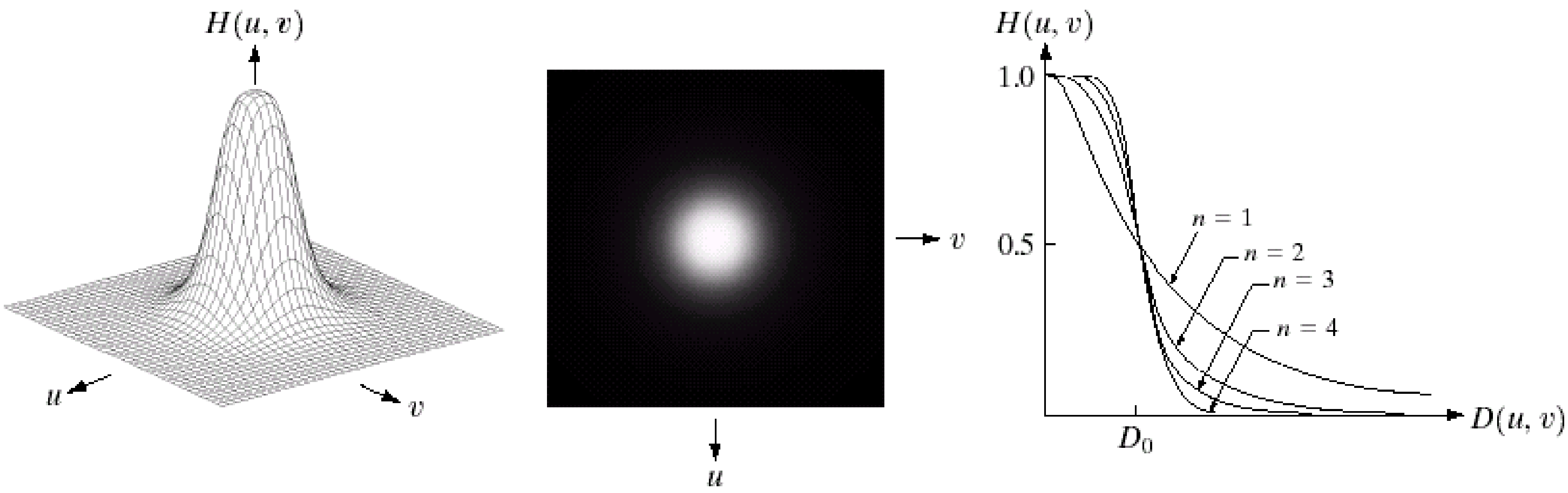
where

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

Smoothing Frequency-Domain Filters

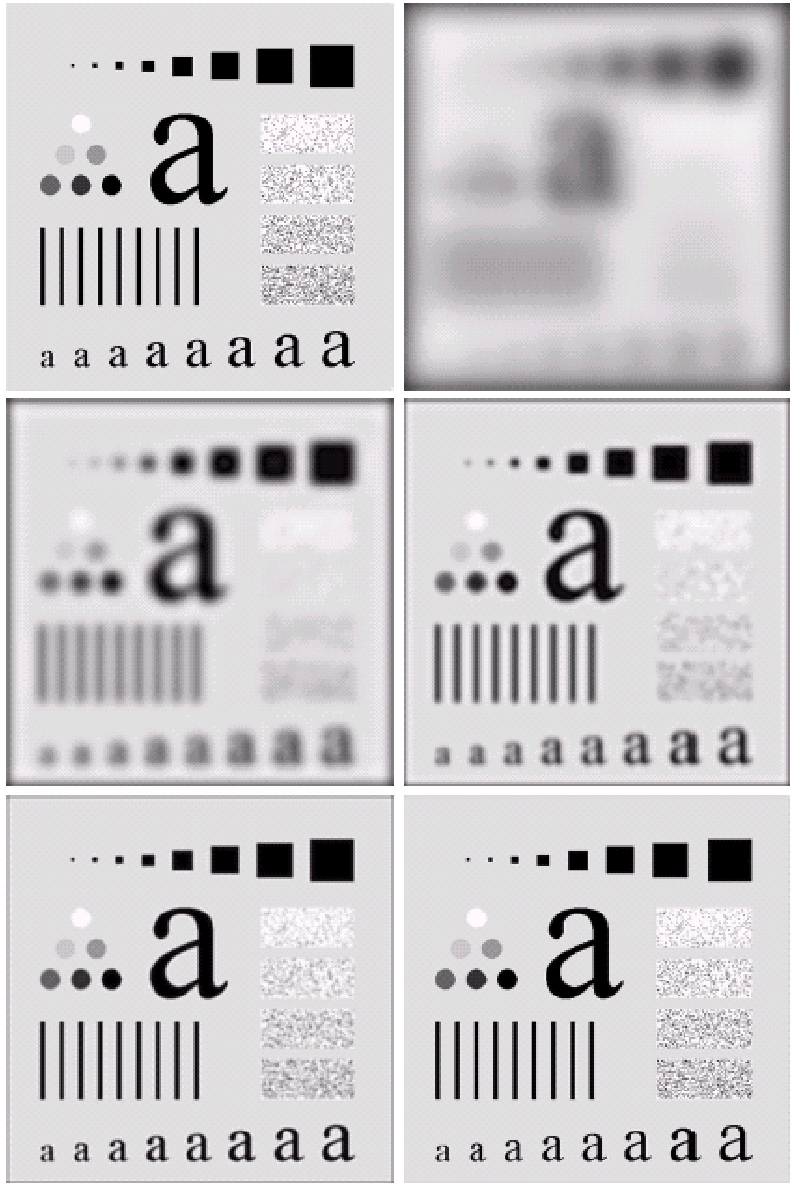
- For filters with smooth transfer functions, defining a cutoff frequency locus at points for which H(u,v) is down to a certain fraction of its maximum value is customary
  - In this case, H(u,v) = 0.5 (down 50% from its maximum value of 1) when D(u,v) = D_0

Smoothing Frequency-Domain Filters



a b c

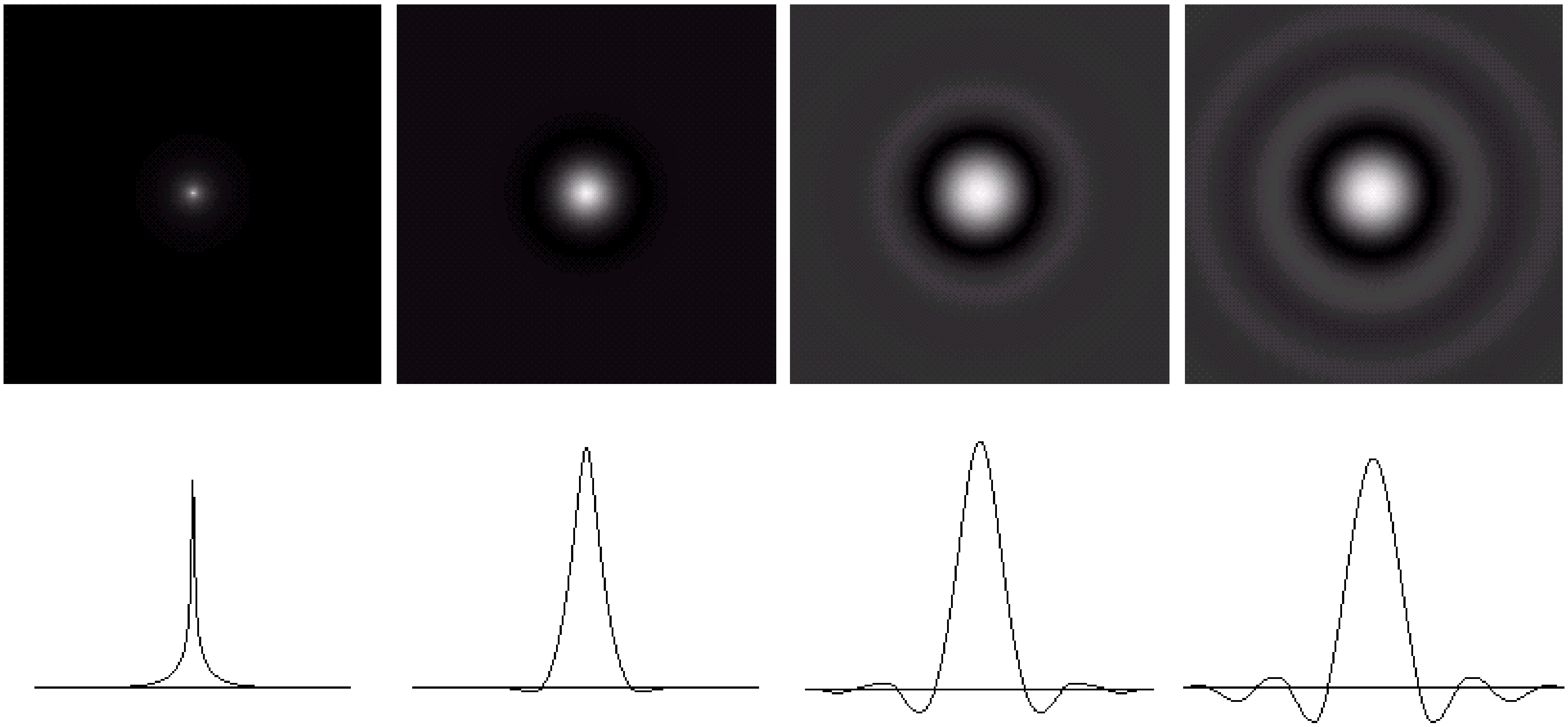
FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.



a b
c d
e f

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.

Smoothing Frequency-Domain 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.

Smoothing Frequency-Domain Filters

- **Gaussian Lowpass Filters**
  - 2-D Gaussian lowpass filters (GLPFs)

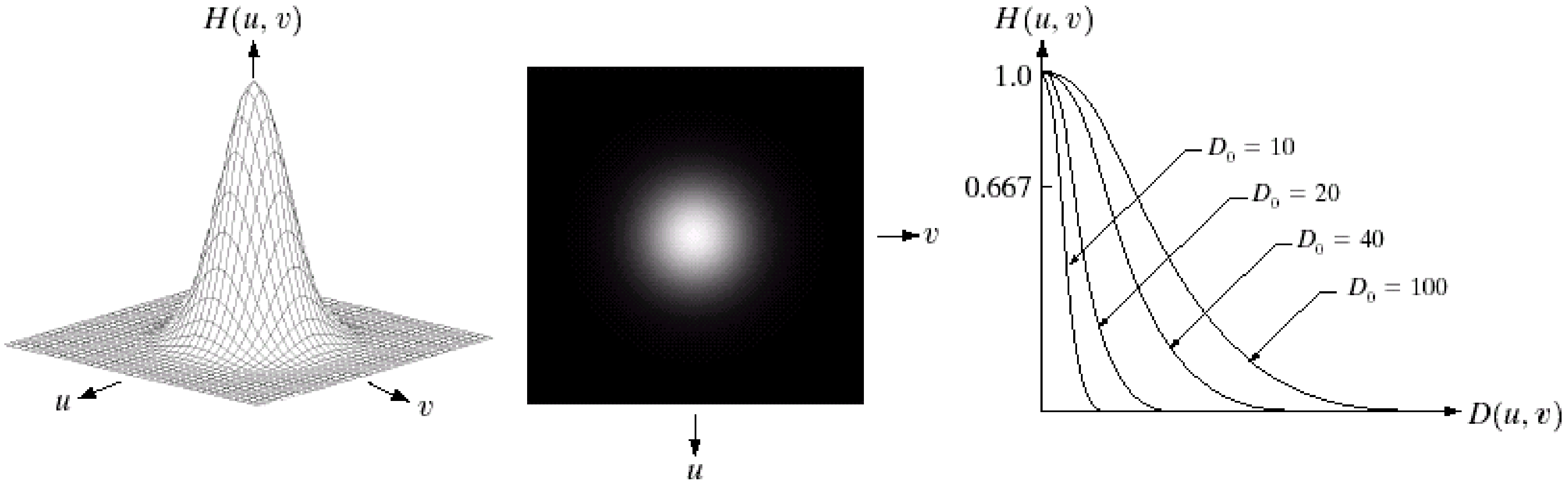
$$H(u,v) = e^{-D^2(u,v)/2\sigma^2}$$

  - By letting σ= D_0 (cutoff frequency):

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

when $D(u,v) = D_0$, the filter is down to 0.607 of its maximum value

Smoothing Frequency-Domain Filters



a b c

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.

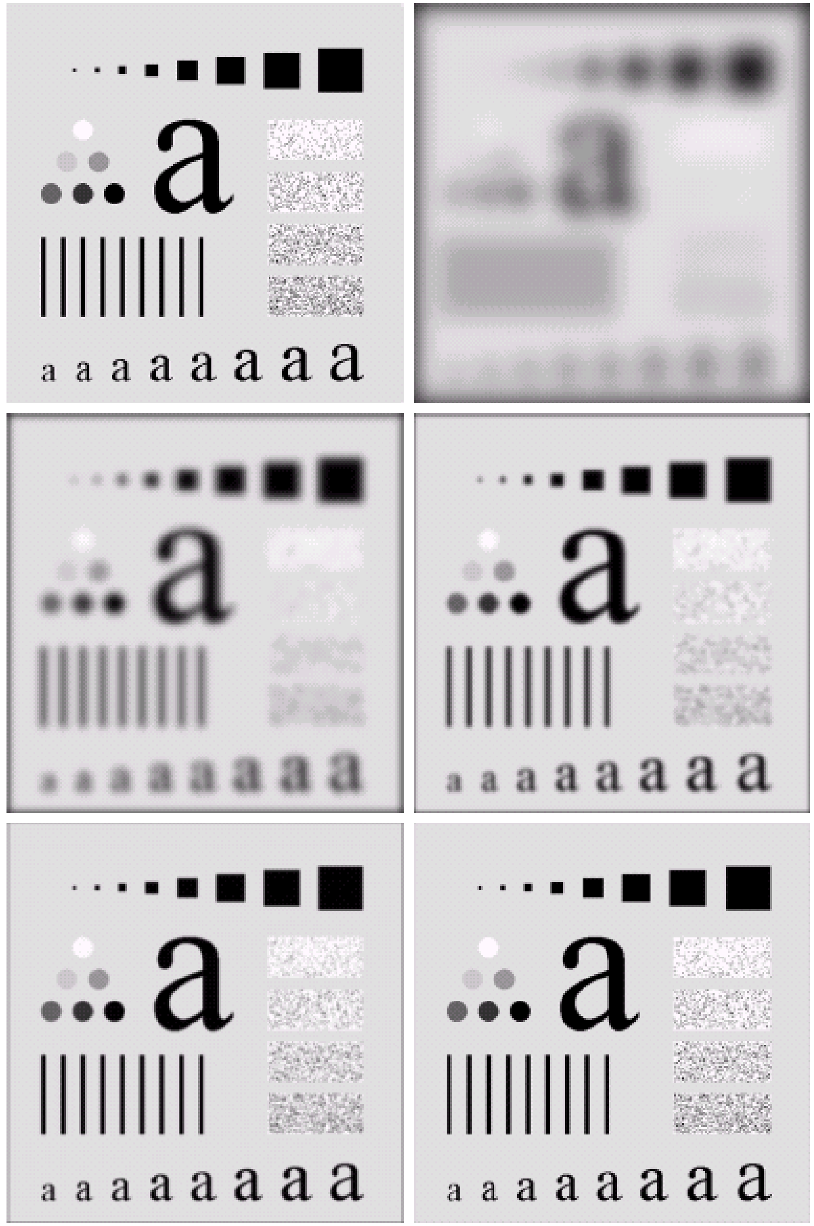


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

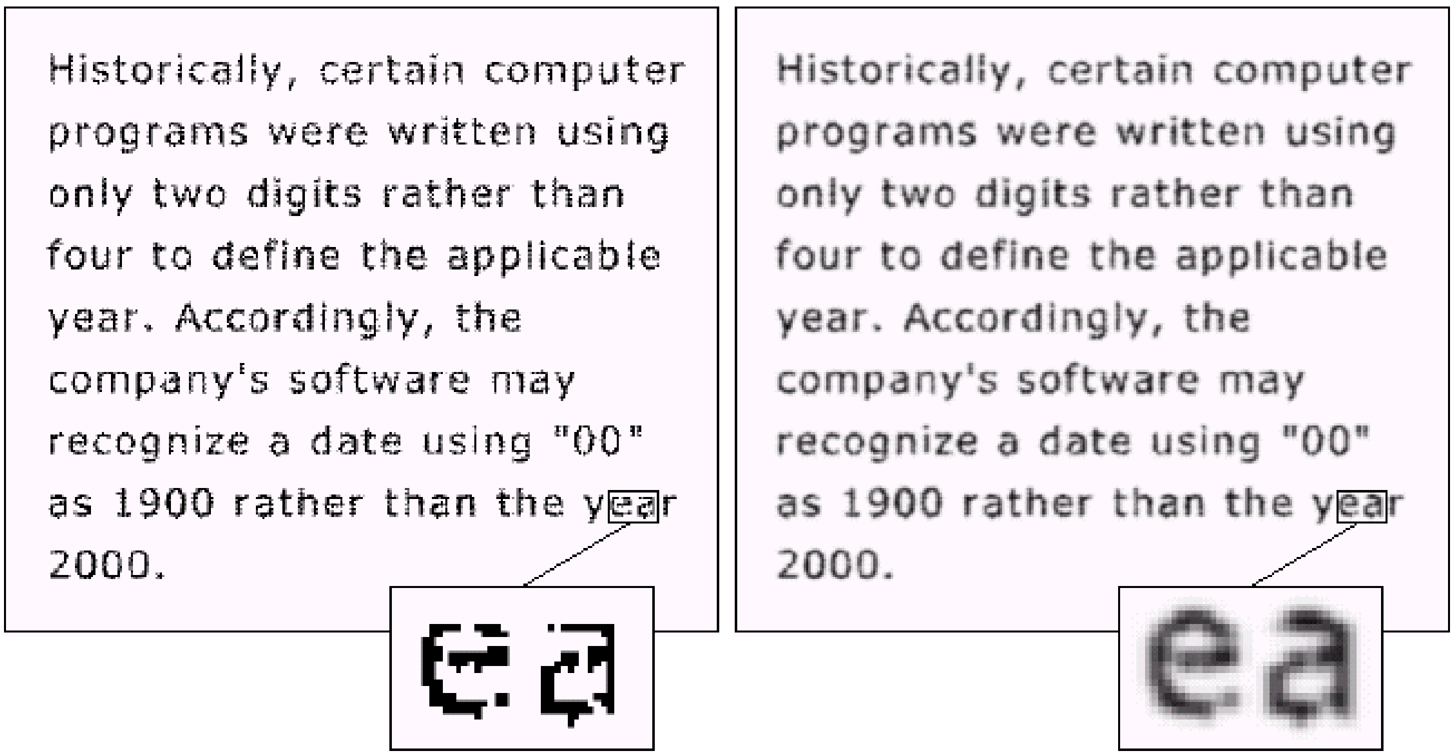
a	b
c	d
e	f

Examples of Lowpass filtering

Character recognition

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).



Examples of Lowpass filtering

Printing and publishing industry

a b c

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).

Examples of Lowpass filtering

Satellite and aerial images

a b c

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.)

Sharpening Frequency-Domain Filters

- Image sharpening can be achieved in the frequency domain by highpass filtering
  - **Attenuates low-frequency components without disturbing high-frequency information**
  - The transfer function of the highpass filters can be obtained using the relation

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

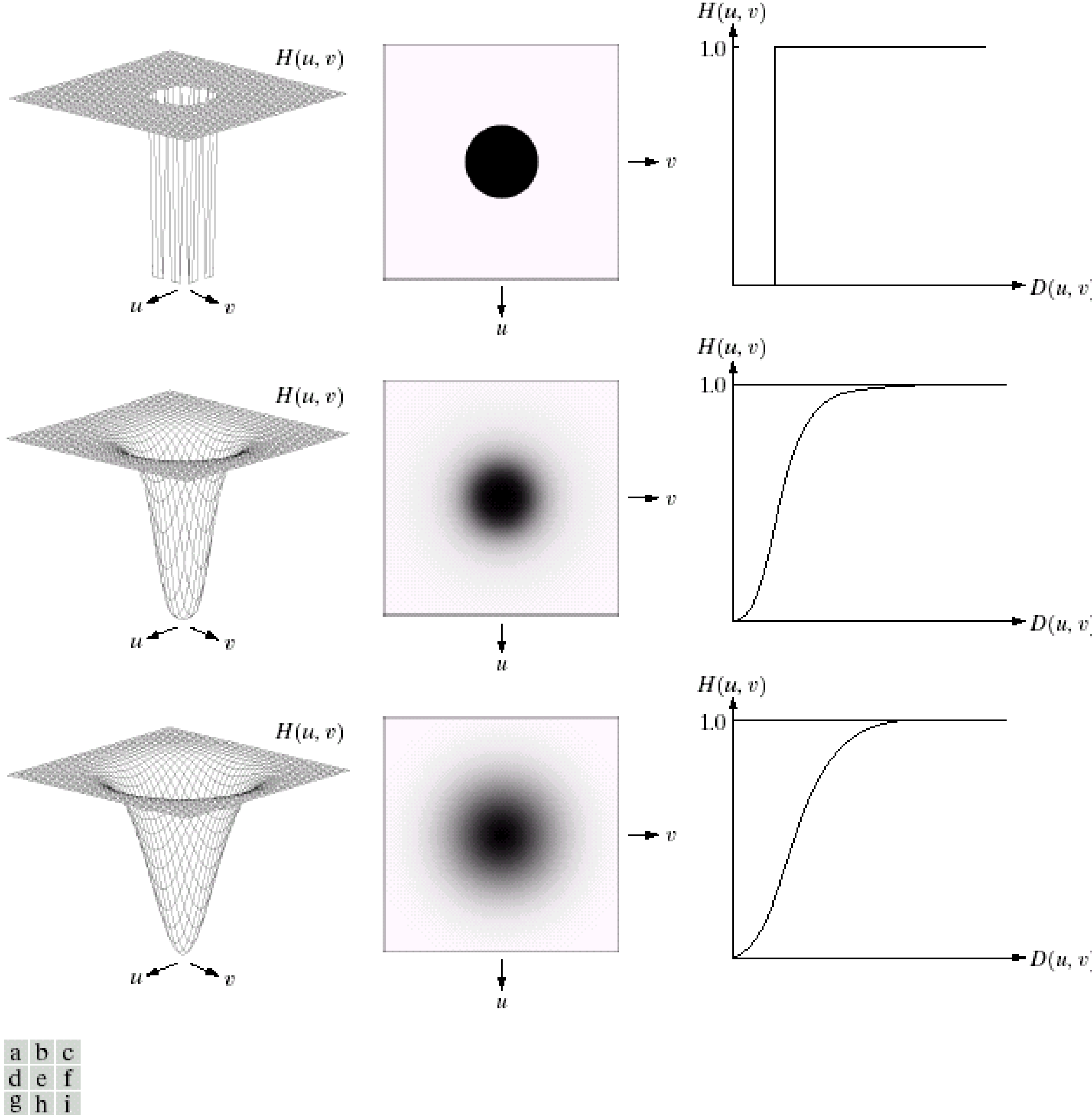
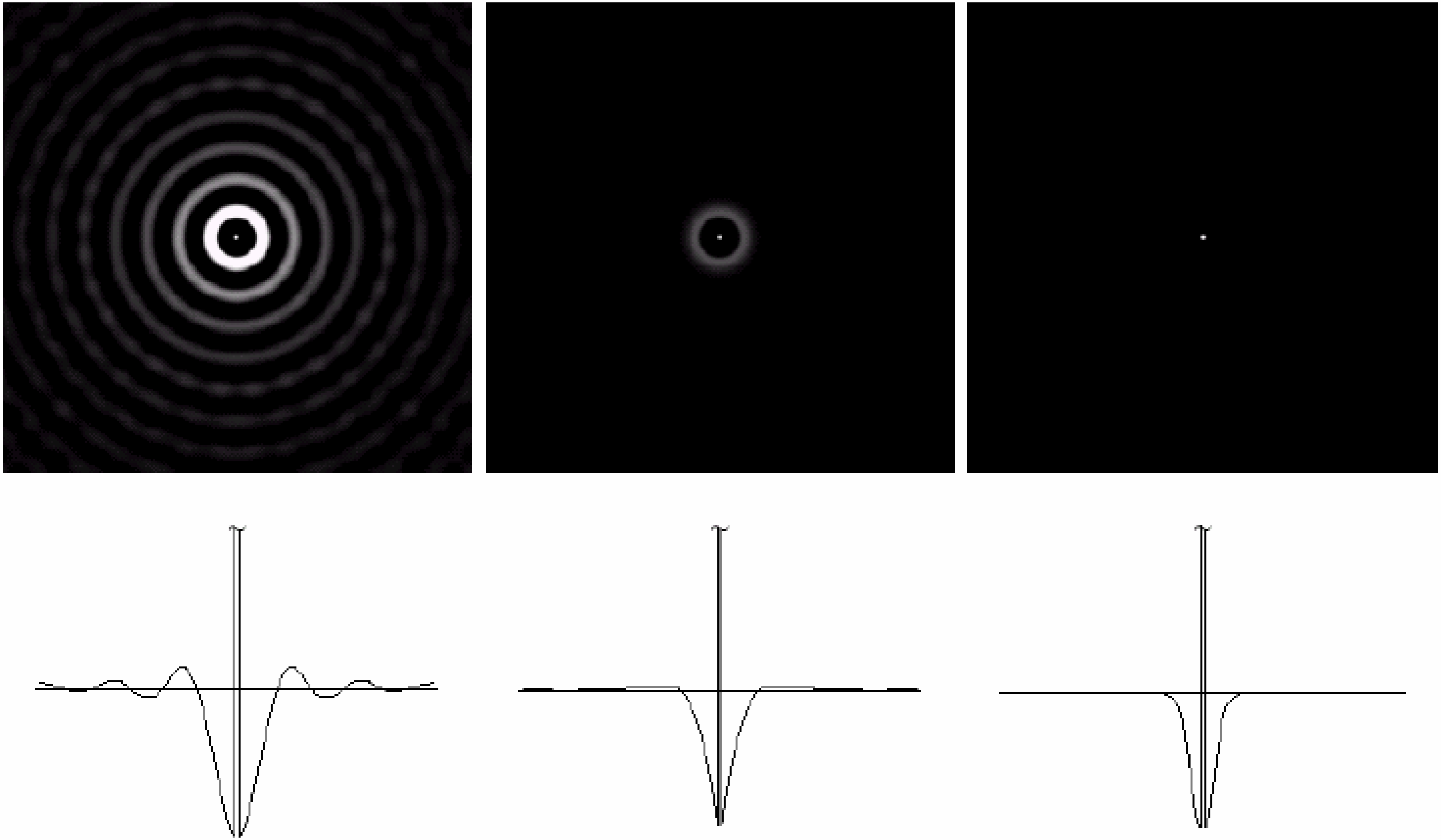


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.



a b c

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

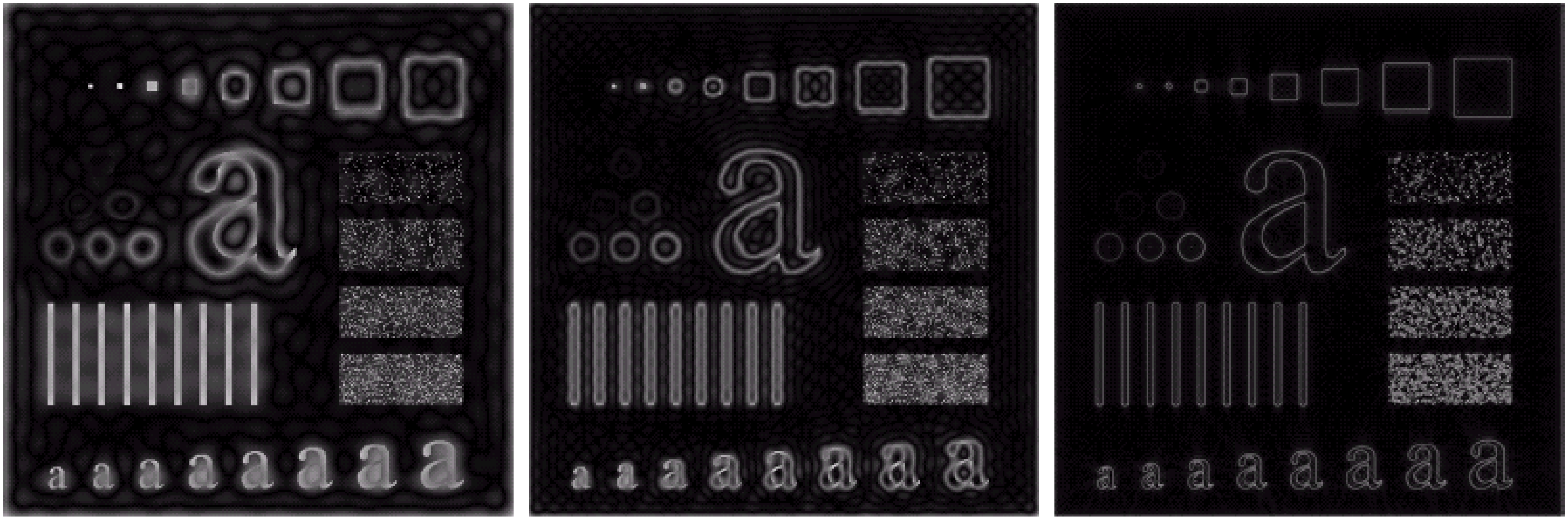
- **Ideal Highpass Filters**
  - A 2-D ideal highpass filter (IHPF) is defined as

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

  where D_0 is the cutoff distance measured from the origin of the frequency rectangle

  - As in the case of the ideal lowpass filter, the IHPF is not physically realizable with electronic components

Sharpening Frequency-Domain Filters



a b c

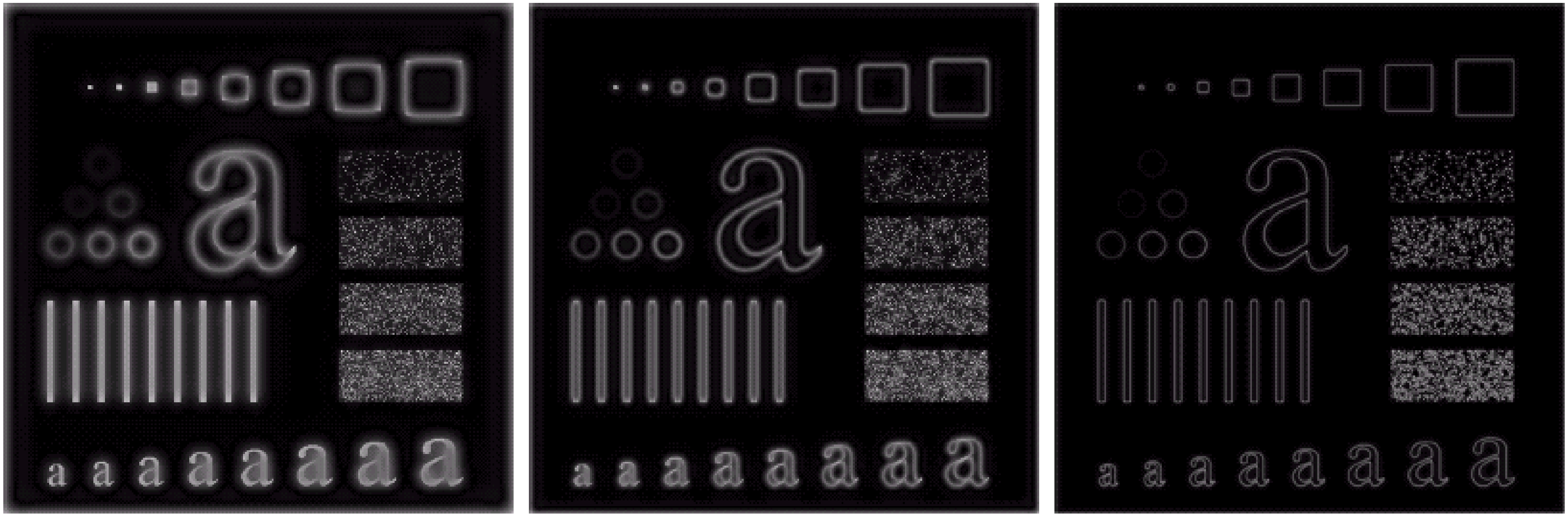
FIGURE 4.24 Results of ideal highpass filtering the image in Fig. 4.11(a) with $D_0 = 15$, 30, and 80, respectively. Problems with ringing are quite evident in (a) and (b).

Sharpening Frequency-Domain Filters

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

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

Sharpening Frequency-Domain Filters



a b c

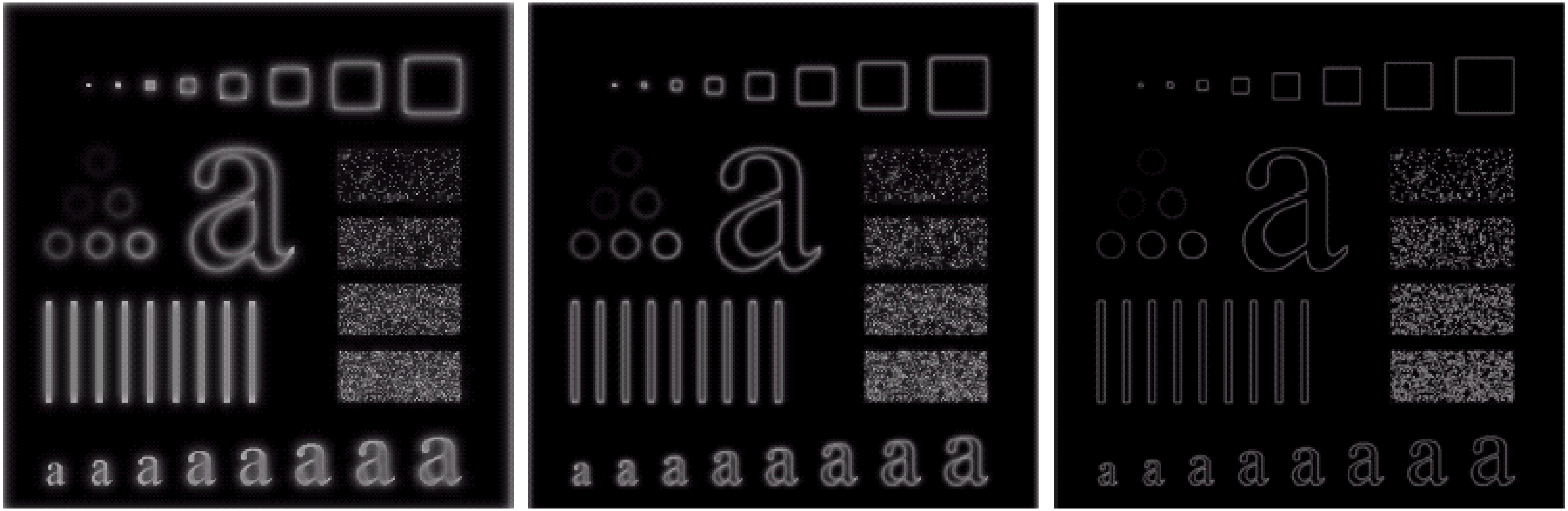
FIGURE 4.25 Results of highpass filtering the image in Fig. 4.11(a) using a BHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. These results are much smoother than those obtained with an ILPF.

Sharpening Frequency-Domain Filters

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

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

Sharpening Frequency-Domain 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

EXAMPLE : Using highpass filtering and thresholding for image enhancement

a b c

FIGURE 4.55
(a) Smudged thumbprint. (b) Result of highpass filtering (a). (c) Result of thresholding (b).

Summary

TABLE 4.5

Lowpass filter transfer functions. D_0 is the cutoff frequency, and n is the order of the Butterworth filter.

Ideal	Gaussian	Butterworth
$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \leq D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$	$H(u,v) = e^{-D^2(u,v)/2D_0^2}$	$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}}$

TABLE 4.6

Highpass filter transfer functions. D_0 is the cutoff frequency and n is the order of the Butterworth transfer function.

Ideal	Gaussian	Butterworth
$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \leq D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$	$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$	$H(u,v) = \frac{1}{1+[D_0/D(u,v)]^{2n}}$

Summary

- Background
- Introduction to the Fourier Transform and the Frequency Domain
  - 1D DFT
  - 2D DFT
- Frequency Domain Filtering Fundamentals
- Smoothing Frequency Domain Filters
- Sharpening Frequency Domain Filters

谢谢大家！