Chapter 4: Image Enhancement in the Frequency Domain

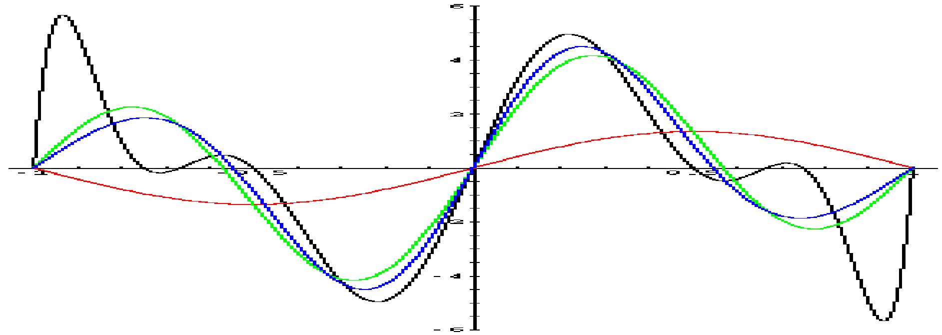
Chapter Overview:

- ✓ Brief outline of the origin of the Fourier transform.
- ✓ Introduce the fundamental concept of the Fourier transform for digital image processing in frequency domain.
- ✓ Apply and discuss the use of Fourier transform for digital image enhancement.

Origin of Fourier Transform

- French mathematician – Jean Baptiste Joseph Fourier (born 1768) introduced a *Fourier Series*.
- Fourier states that any function that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient.
- At it early stage, this idea was accepted by the scientist all over the world with skepticism.

Origin of Fourier Transform (cont’)



- Later, he introduced the concept of Fourier Transform
- Fourier transform – 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.
- Fourier transform becomes one of the very popular tools to solved many practical problems in many areas of applications as compared to the Fourier Series

Origin of Fourier Transform (cont')

- Fourier Series & Fourier Transform share important characteristic that a function, expressed in either a Fourier series or transform, *<u>can be reconstructed (recovered) completely via an inverse process</u>*, with no information loss.

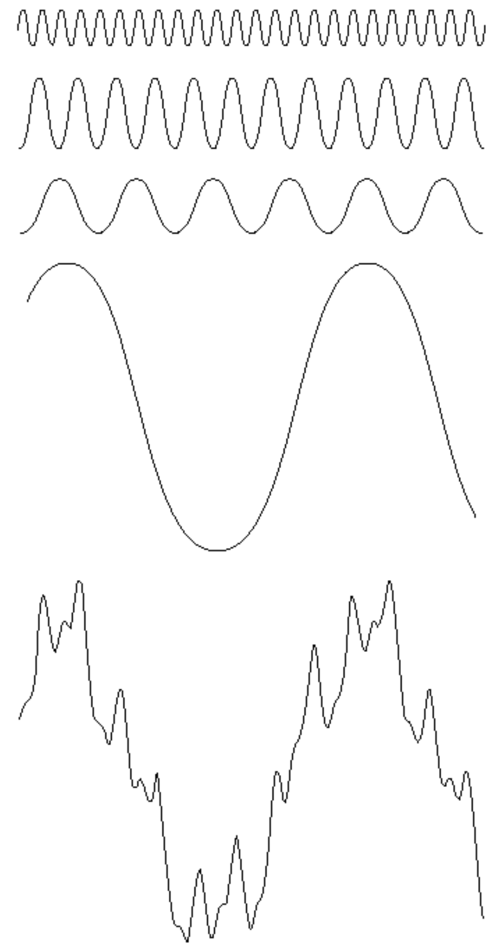


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

Fourier Transform in DIP

- DIP deals with functions with finite duration, thus we could utilized Fourier transform in frequency domain to provide a meaningful and practical image processing especially for image enhancement purposes.

Fourier Transform & Frequency Domain

One Dimensional Fourier Transform & its Inverse

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

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux}\, dx \qquad \text{where } j = \sqrt{-1}$$

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

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{-j2\pi ux}\, du \qquad \text{where } j = \sqrt{-1}$$

- This 2 equations comprise the *Fourier Transform Pair*

Fourier Transform & Frequency Domain

Two Dimensional Fourier Transform & its Inverse

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

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

– Similarly for the inverse Fourier transform

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

– Our interest is in discrete functions, since DIP is dealing with discrete data.

One Dimensional Fourier Transform & its Inverse

Discrete Fourier Transform (DFT):

– Fourier transform of a discrete function of one variable, *f(x)*, $x = 0,1,2,\ldots,M\text{-}1$ is expressed by:

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

– Similarly, inverse DFT could be obtained by:

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

– For computational purpose, we start substituting $u = 0$ in the exponential term and then summing all the values of x. Then, substitute $u=1$ in the exponential and repeat the summation over all the values of x. Repeat all process for all M values of u.

One Dimensional Fourier Transform & its Inverse

Integrating Frequency domain into DFT:

- The concept of frequency domain follows directly from Euler's formula:

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

- Thus, substituting this equation into DFT formula defined earlier gives us:

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

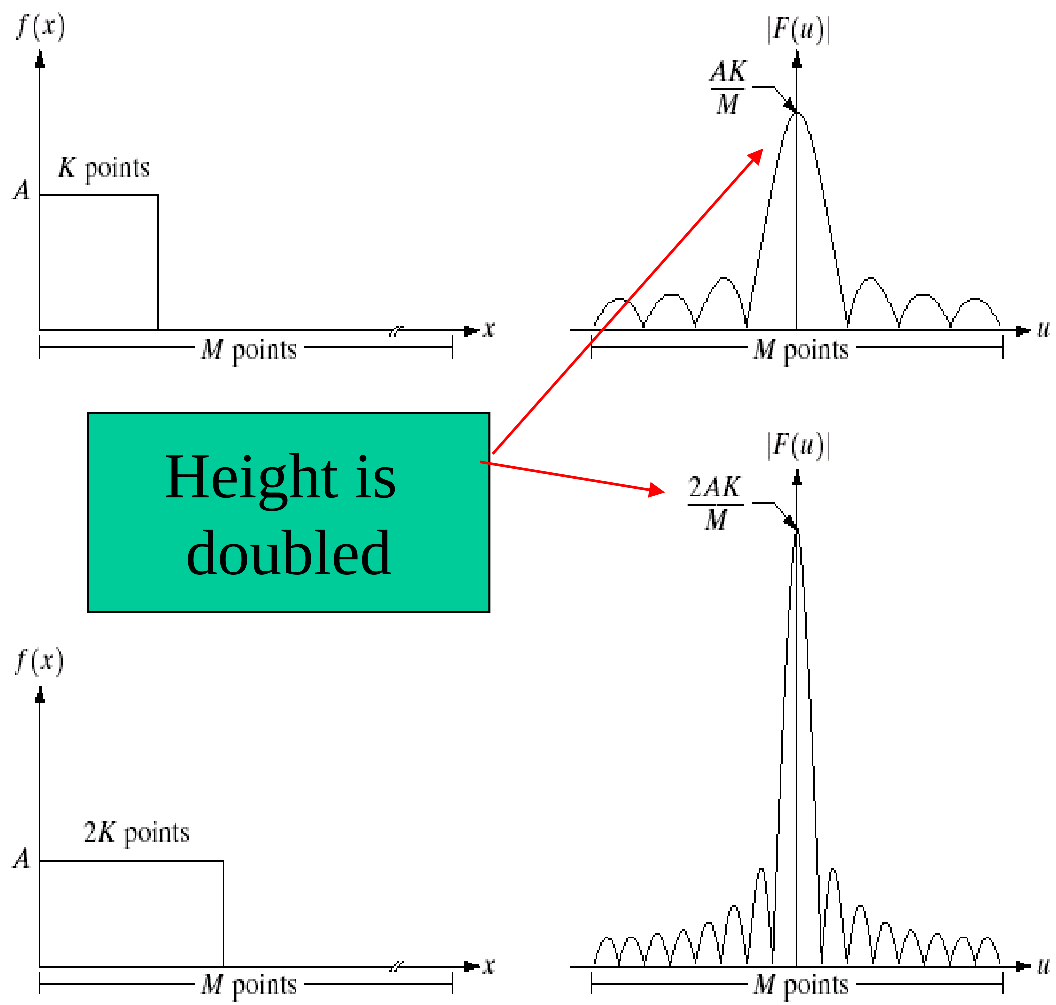
for $u = 0,1,2,\ldots, M-1.$

- The domain (values of *u*) over which the values of *F(u)* range is appropriately called the *frequency domain* because *u* determines the frequency of the components of the transform.
- The Fourier transform may be viewed as "mathematical prism" that separates a function into various components, also based on frequency content.

Example: Fourier Spectra of Two Simple 1-D Function

Consider,
M=1024, A=1 and K is only 8 points. Also, the spectrum is centered at u=0.

In order to center the transform, *f(x)* was multiplied by $(-1)^x$ before taking the transform.



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.

Two Dimensional Fourier Transform & its Inverse

Extending 1-D DFT to 2-D DFT :

– DFT of a function (image) *f(x,y)* of size *M x N* is given by the equation:

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

for all values of $u = 0,1,2,\ldots, M-1$ and $v = 0,1,2\ldots, N-1$.

– Similarly, 2-D inverse DFT may be expressed as:

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

for all values of $x = 0,1,2,\ldots, M-1$ and $y = 0,1,2\ldots, N-1$.

– The variables *u* and *v* are the *transform or frequency variables*, and *x* and *y* are the *spatial or image variables*.

Two Dimensional Fourier Transform & its Inverse (cont')

Shifting the Origin in DFT:

- It is a common practice to multiply the input image function by $(-1)^{x+y}$ prior to computing the Fourier Transform.
- Due to the exponential properties, it can be shown that:

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

where $\Im$ denotes the Fourier transform of the argument.

- This equation states that the origin of the Fourier Transform of $f(x,y)(-1)^{x+y}$ [that is, $F(0,0)$] is located at $u = M/2$ and $v = N/2$.
- In other words, multiplying $f(x,y)$ by $(-1)^{x+y}$ shifts the origin of $F(u,v)$ to the frequency coordinates $(M/2, N/2)$, which is the center of the M x N area occupied by the 2D DFT. – this domain is referred to as *frequency rectangle*.
- In order to ensure the operability, the M and N have to be even number integer

Two Dimensional Fourier Transform & its Inverse (cont')

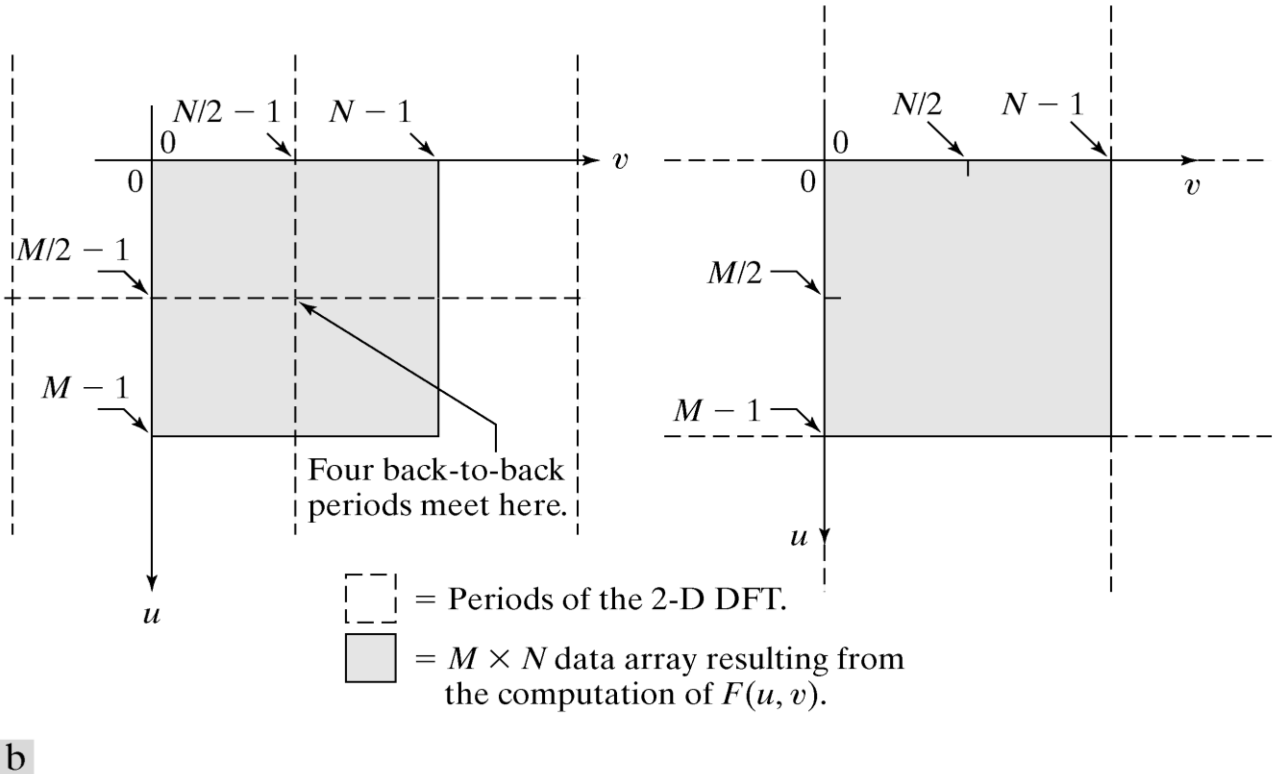


FIGURE 4.2 (a) $M \times N$ Fourier spectrum (shaded), showing four back-to-back quarter periods contained in the spectrum data. (b) Spectrum obtained by multiplying $f(x, y)$ by $(-1)^{x+y}$ prior to computing the Fourier transform. Only one period is shown shaded because this is the data that would be obtained by an implementation of the equation for $F(u, v)$.

Example: Centered Spectrum of a Simple 2-D Function

Consider image in (a):

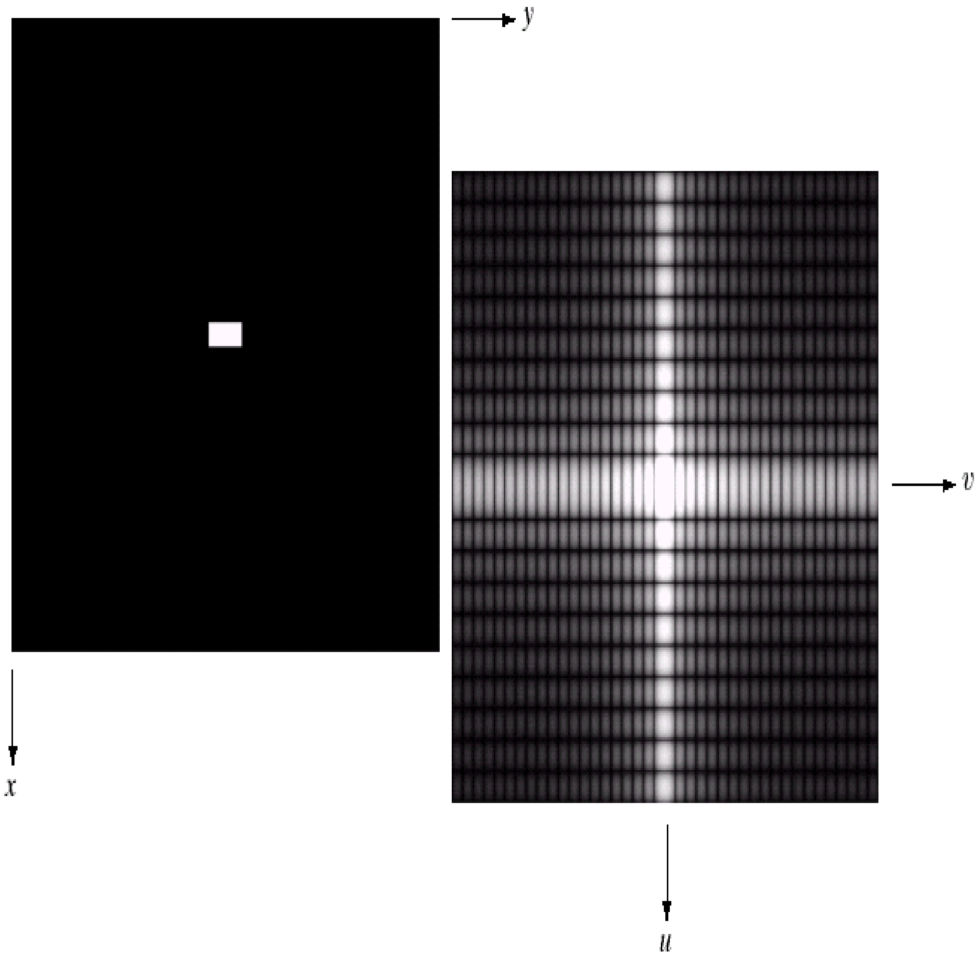
This image was multiplied by $(-1)^{x+y}$ prior to computing the Fourier Transform in order to center the spectrum.

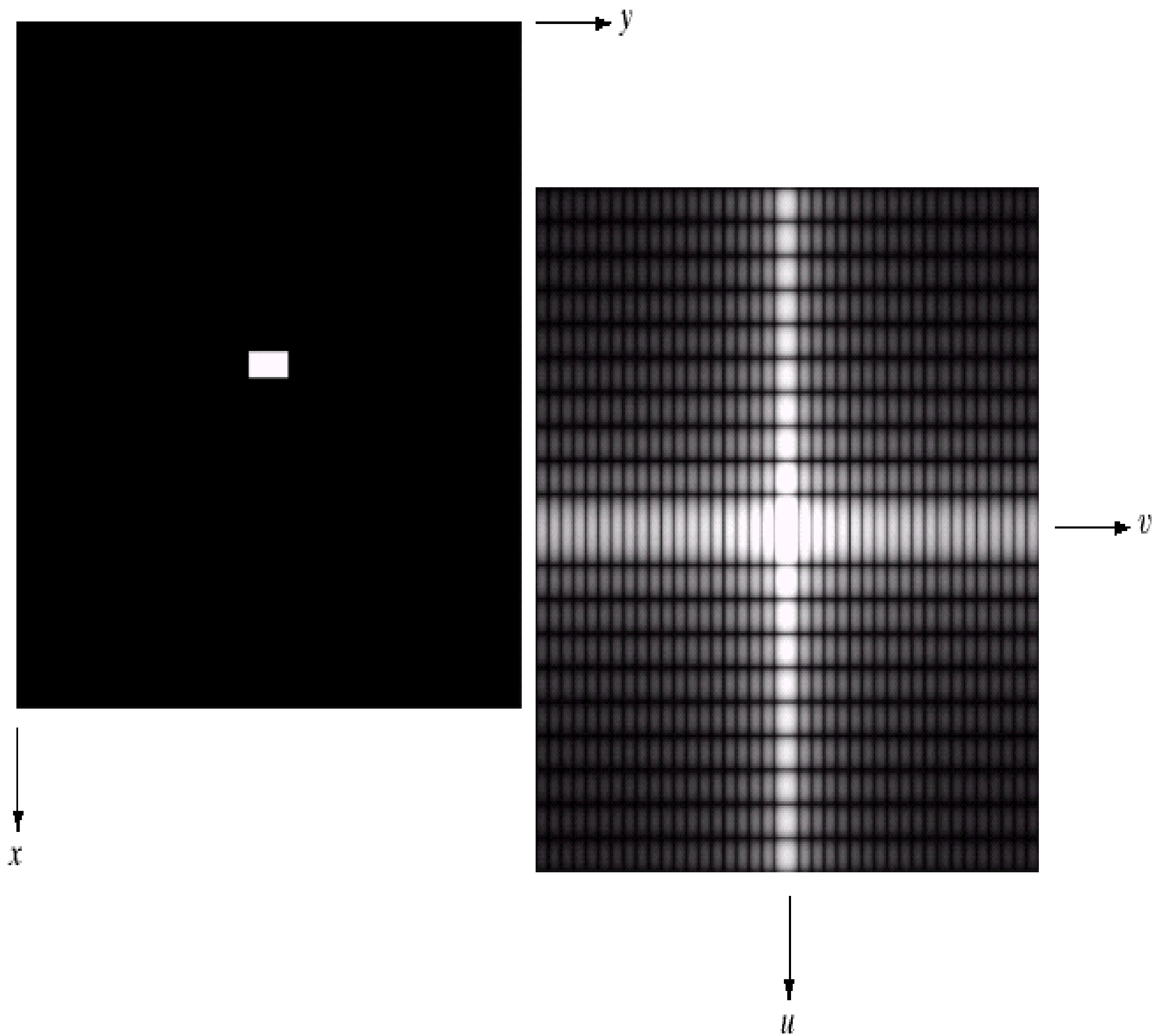
Note: the separation of spectrum zeros in the *u* direction is exactly twice the size of zeros in *v* direction – compare to the image size.

.

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.

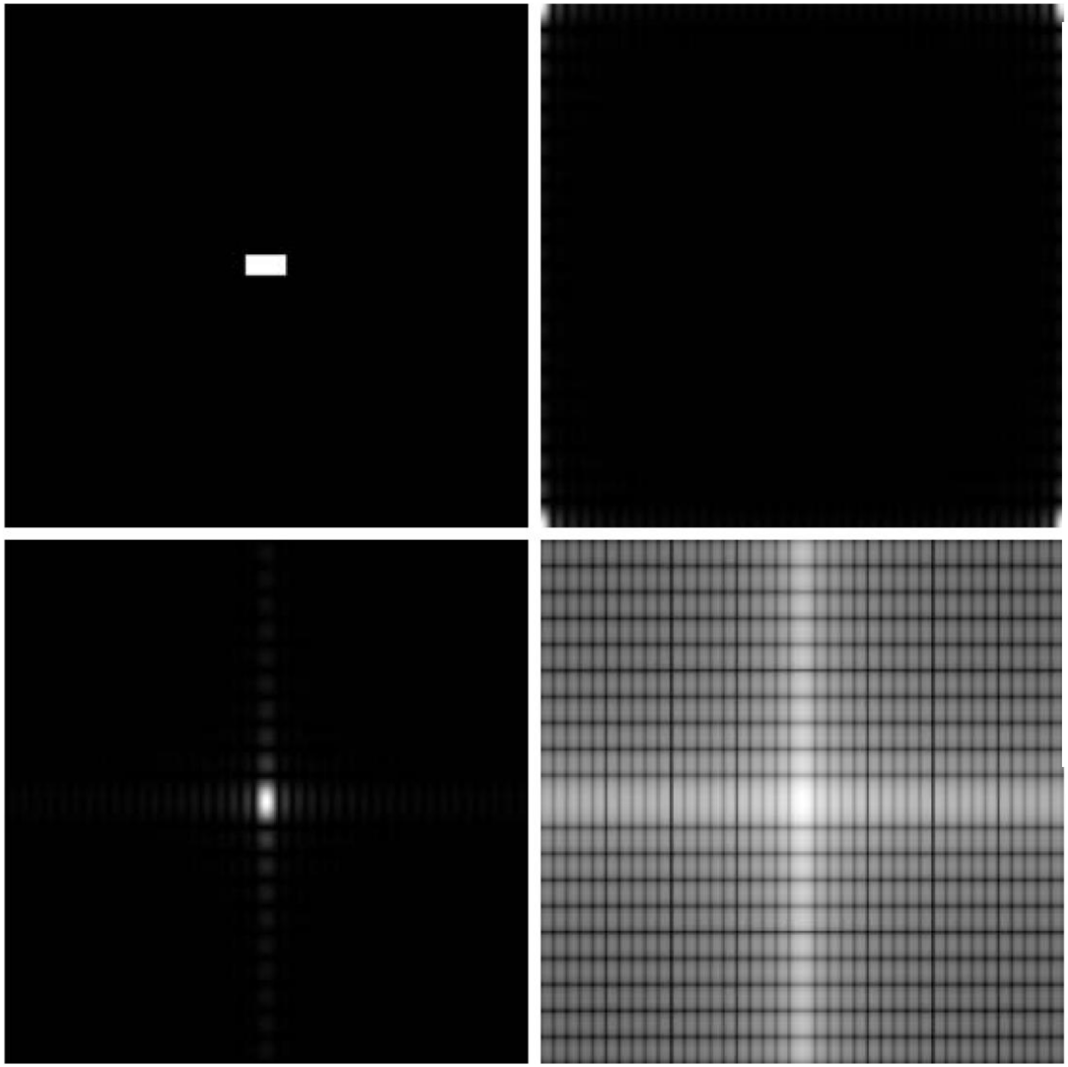




a b

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.

Example: Further Processing in DFT

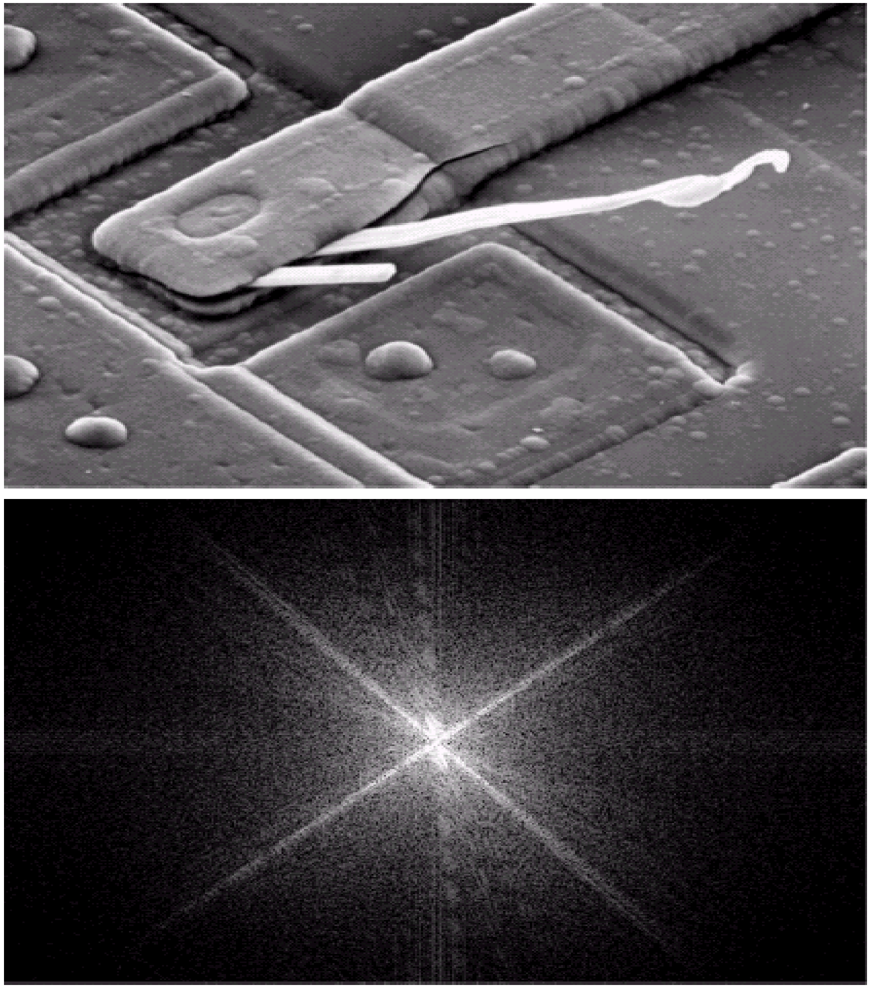


a	b
c	d

FIGURE 4.3
(a) A simple image.
(b) Fourier spectrum.
(c) Centered spectrum.
(d) Spectrum visually enhanced by a log transformation.

Example: An Image and Its Fourier Spectrum

2500 times magnification



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.)

Basis Steps for Filtering in the Frequency Domain

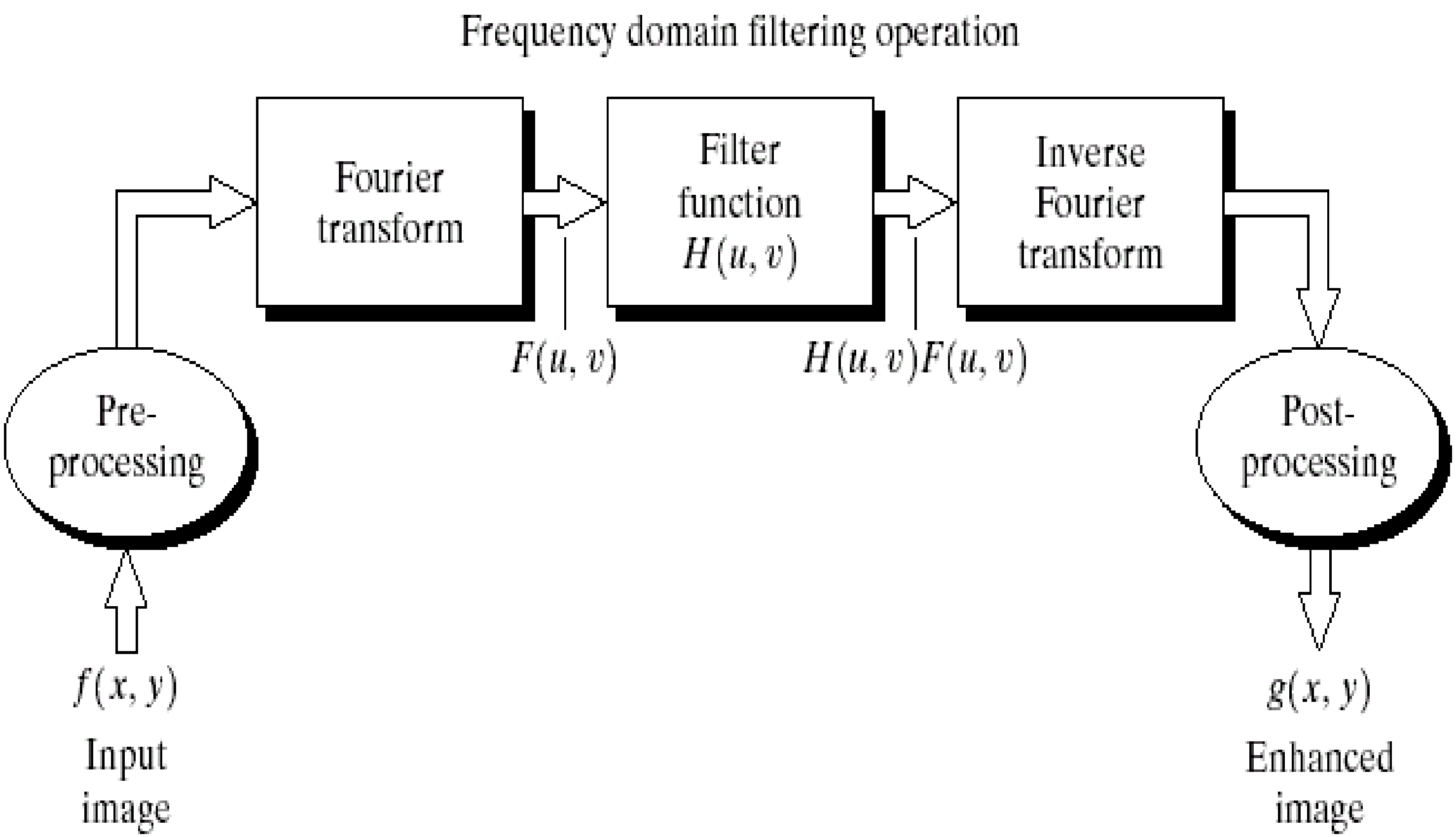


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Basics of filtering in the frequency domain

Filter transfer function

1. multiply the input image by $(-1)^{x+y}$ to center the transform to $u = M/2$ and $v = N/2$ (if M and N are even numbers, then the shifted coordinates will be integers)
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 $(-1)^{x+y}$ to cancel the multiplication of the input image.

Basis Steps for Filtering in the Frequency Domain (cont')

H(u,v) – suppresses certain frequency in the transform while leaving other unchanged.

e.g. $G(u,v) = H(u,v)F(u,v)$

Multiplication of a 2-D function defined by element by element basis.

The components of *F* are complex quantities but he filters that we deal in this course are real.

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

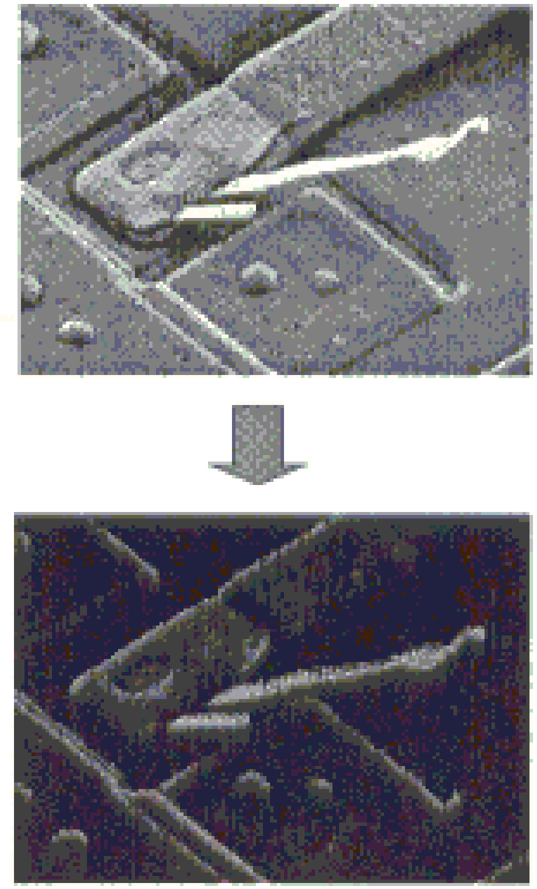
$$\text{Filtered image} = \mathfrak{I}^{-1}[G(u,v)]$$

Basis Steps for Filtering in the Frequency Domain (cont')

The pre and post processing includes:

- multiplication $(-1)^{x+y}$ process
- cropping of the input image to its closest even dimension
- gray level scaling
- conversion to floating point on input
- conversion to an 8-bit integer format on the output,
- etc.

Notch filter



- this filter is to force the F(0,0) which is the average value of an image (dc component of the spectrum)
- the output has prominent edges
- in reality the average of the displayed image can't be zero as it needs to have negative gray levels. the output image needs to scale the gray level

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

Effect of a Notch Filter

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.

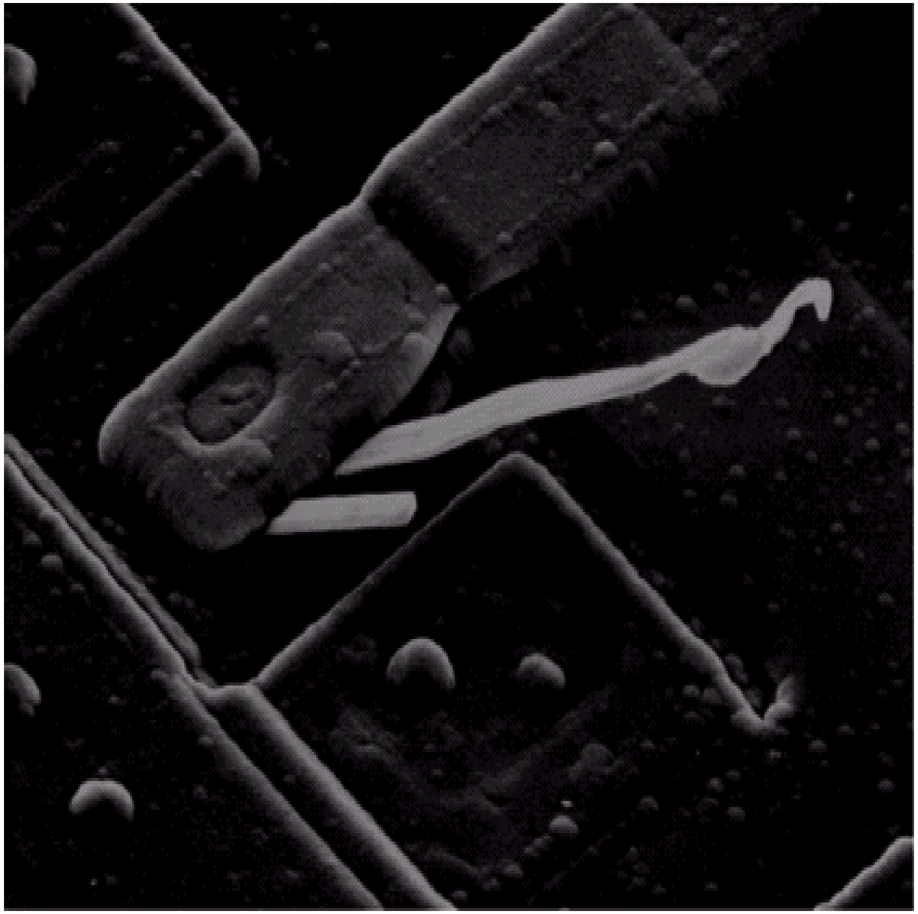


Image appears darker

Lowpass & Highpass Filtering

– Low frequencies in the Fourier Transform are responsible for general gray level appearance of an image over smooth area.

– High frequencies are responsible for details such as edges and noise.

Lowpass Filter

- Attenuates high frequencies while "passing" low frequencies
- expecting filtered image to have less sharp detail than the original
- image will appear blur

Highpass Filter

- Attenuates low frequencies while "passing" high frequencies
- expecting less gray level variation in smooth areas and emphasized transitional (e.g. edges) gray level detail.
- image will appear sharper.

Lowpass & Highpass Filter Effects

Lowpass

Highpass

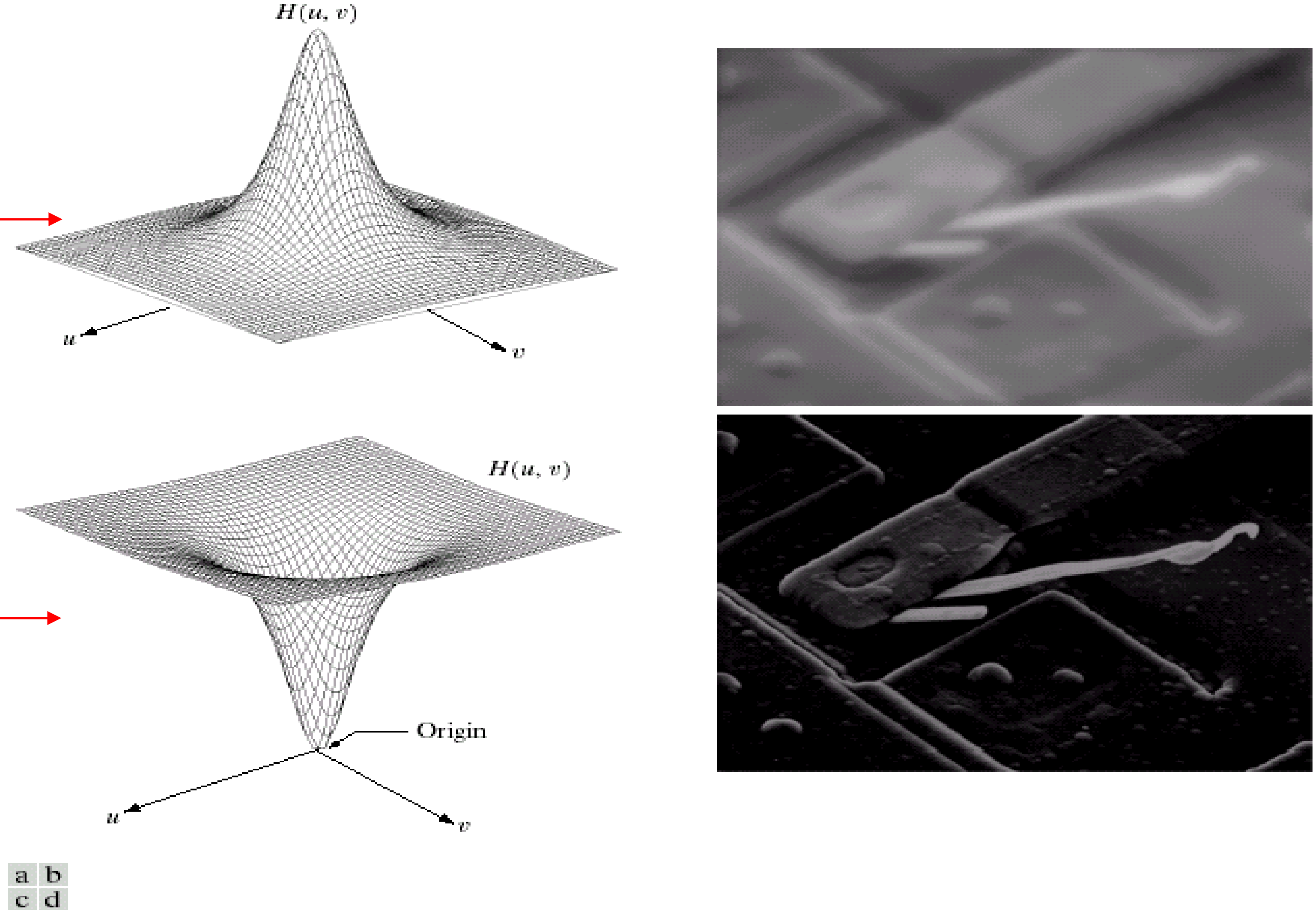
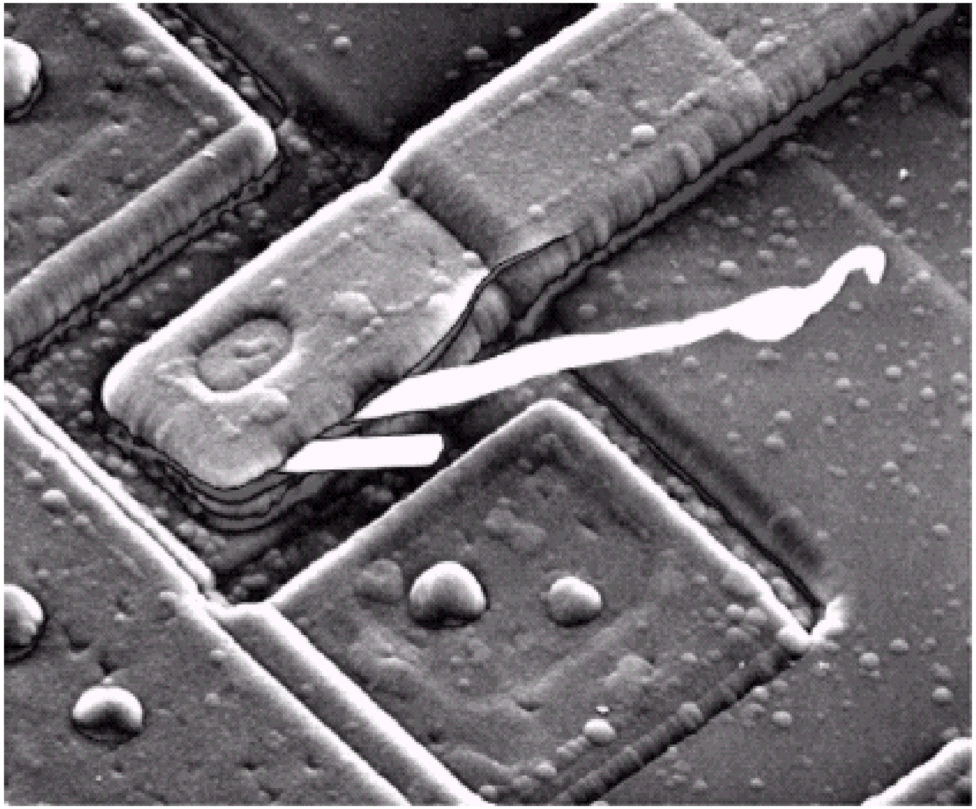


FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

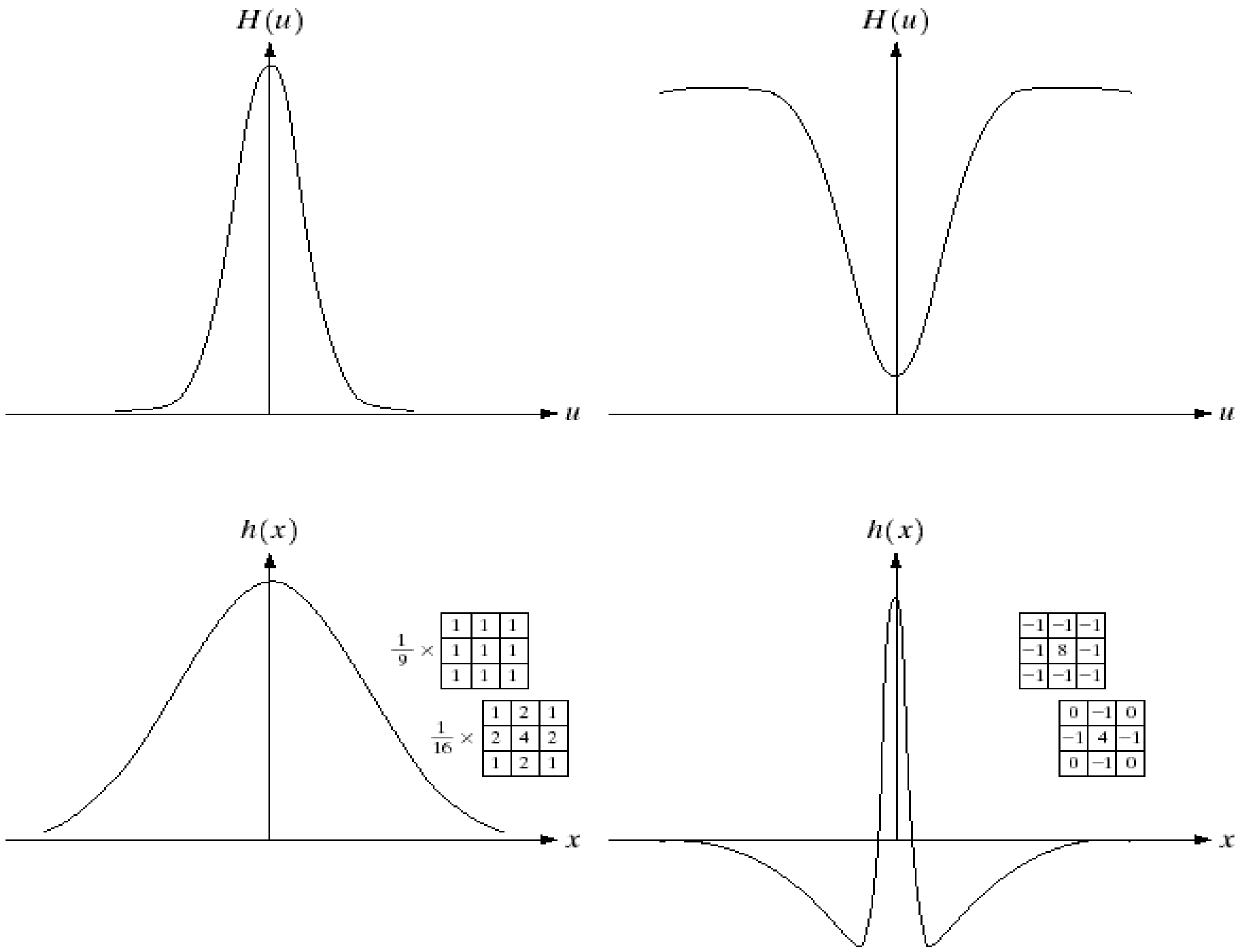
Effect of Addition of a Constant to Earlier Filtered Image

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).



Purpose – not to completely eliminate F(0,0)

Relation Between the Filtering in Spatial & Frequency Domains



a b
c d

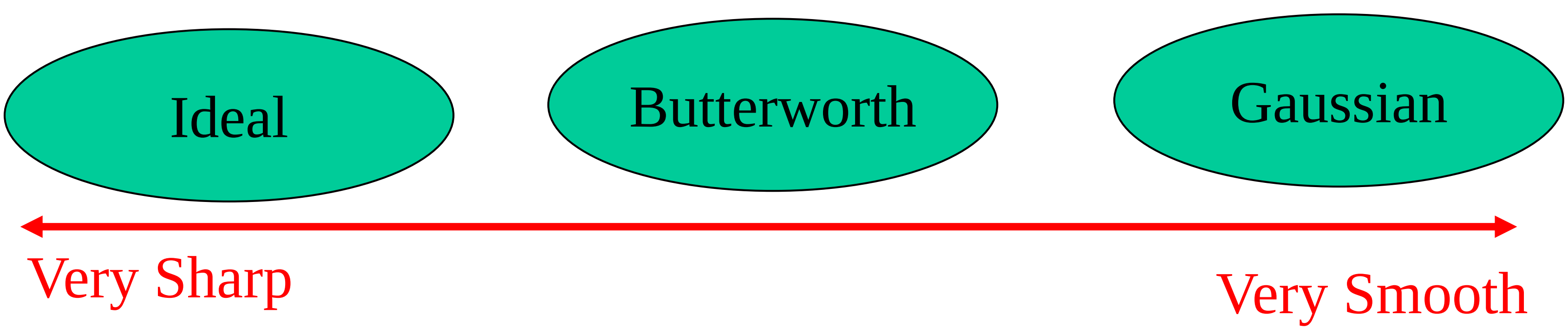
FIGURE 4.9 (a) Gaussian frequency domain lowpass filter. (b) Gaussian frequency domain highpass filter. (c) Corresponding lowpass spatial filter. (d) Corresponding highpass spatial filter. The masks shown are used in Chapter 3 for lowpass and highpass filtering.

Smoothing Frequency Domain Filters

As mentioned earlier, edges and other sharp transitions (e.g. noise) are the results from high frequency contents – hence, smoothing (blurring) can be achieved by attenuating a specified range of frequency components In the Fourier transform of a given image.

i.e $G(u,v) = H(u,v)F(u,v)$

Thus, consider 3 types of filters:



Ideal Lowpass Filters (ILPF)

- "Cut off" all high frequency components of the Fourier Transform :

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

Cutoff frequency

-the point of transition between H(u,v)=1 and H(u,v)=0.

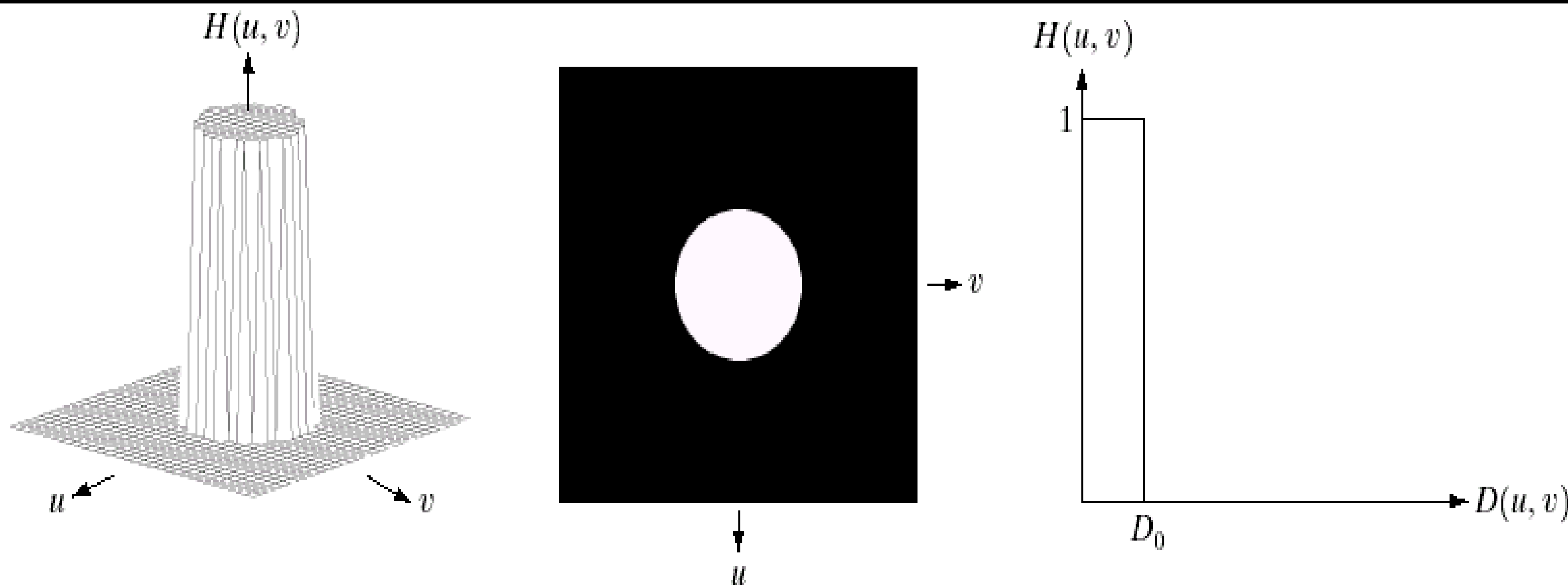
where D_0 is a specified distance from the origin of the centered transform, $D(u,v)$ is the distance from the point (u,v) to the origin of the frequency rectangle.

- If the image is of size M x N, thus

The center of the freq. rectangle

$$D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{0.5}$$

Ideal Lowpass Filters (ILPF) (cont')



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.

Ideal Lowpass Filters (ILPF) (cont')

One way to establish a set of standard *cutoff frequency* loci is to compute Circle that enclose specified amounts of total image power, P_T.

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

where $P(u,v) = |F(u,v)|$

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

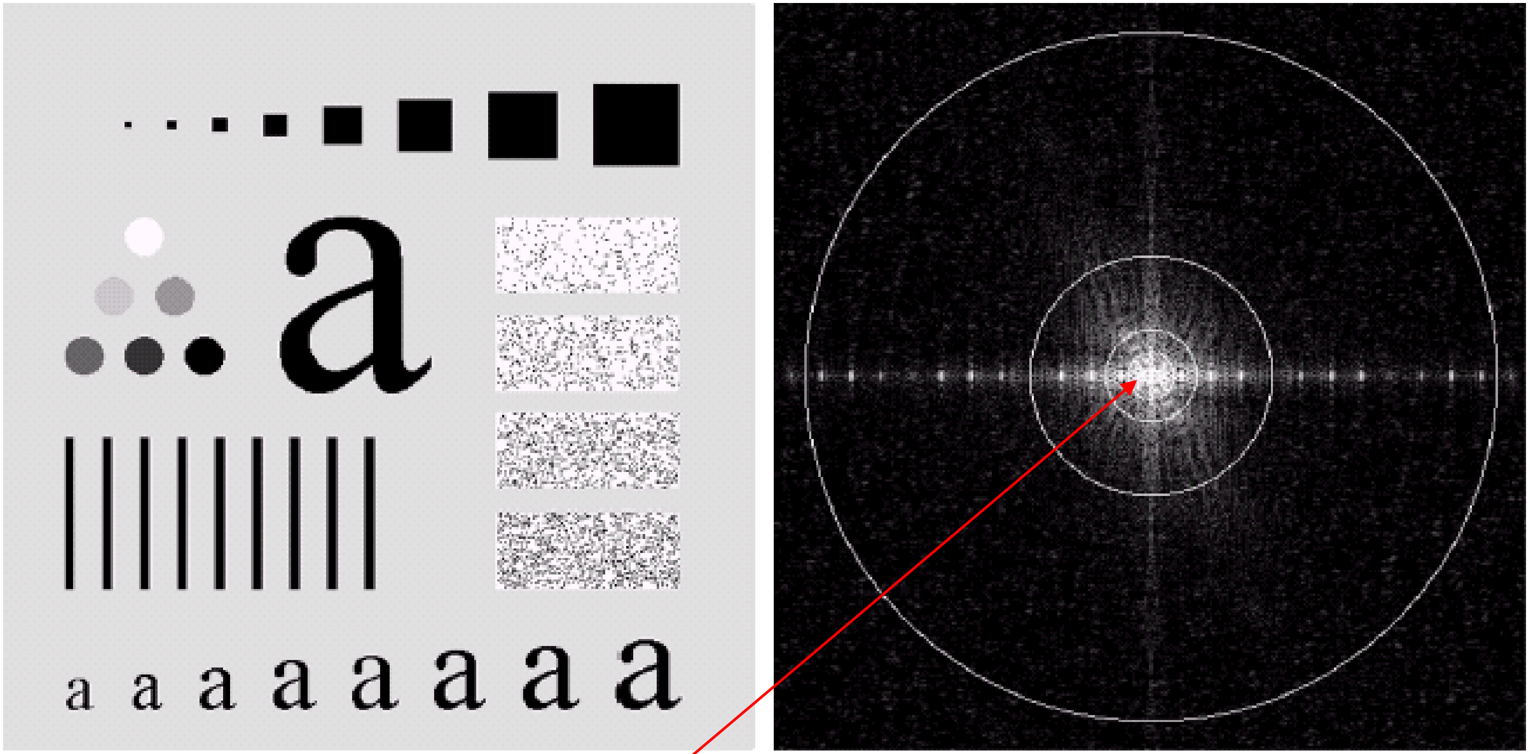
where $R(u,v)$ *and* $I(u,v)$ are real and imaginary part of $F(u,v)$, respectively.

If the transform has been centered, a circle of radius r with the origin at the center of the frequency rectangle encloses α percent of the power, where

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

and the summation is taken over the values of (u,v) that lie inside the circle.

Example: Image Power Circle



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.

α

92% of the total image power being enclosed by a relatively small circle of radius 5.

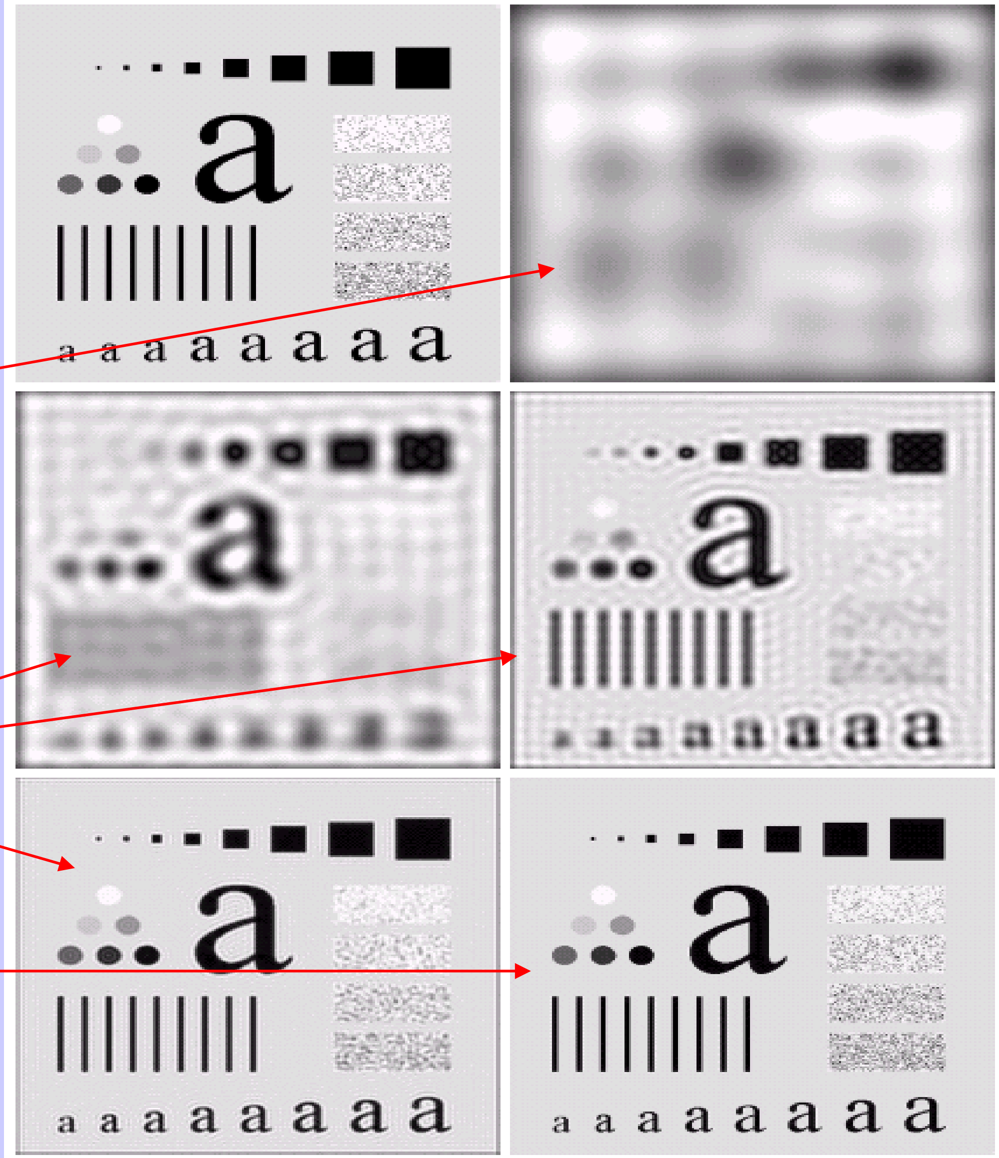
Results of ILPF using earlier defined Total Image Power

8% image power is removed - useless

"Ringing" effect

Little "ringing" effect

Thus, ILPF not very practical – suitable for computer simulation & as analysis tools



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.

Butterworth Lowpass Filters (BLPF)

The transfer function of a BLPF of order n with cutoff frequency at a distance D_0 from the origin, is defined as:

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

$$where \quad D(u,v) = [(u-\tfrac{M}{2})^2 + (v-\tfrac{N}{2})^2]^{0.5}$$

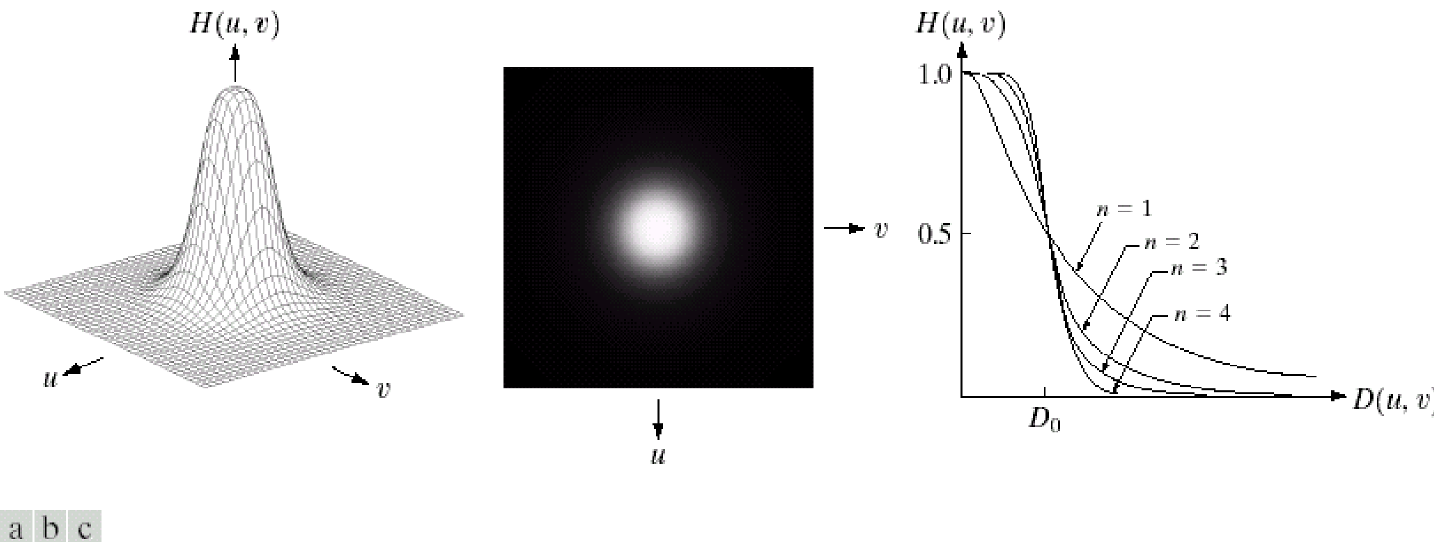


FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

BLPF (cont')

Note: smooth transition in blurring as a function of increasing cutoff freq. compared to ILPF.

No ringing appears!

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.

Gaussian Lowpass Filters (GLPF)

GLPF is defined as:

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

where $D(u,v)$ is the distance from the origin of the Fourier Transform and σ is a measure of the spread of the Gaussian curve.

By letting $\sigma = D_0$, we get:

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

where D_0 is the cutoff frequency

GLPF (cont')

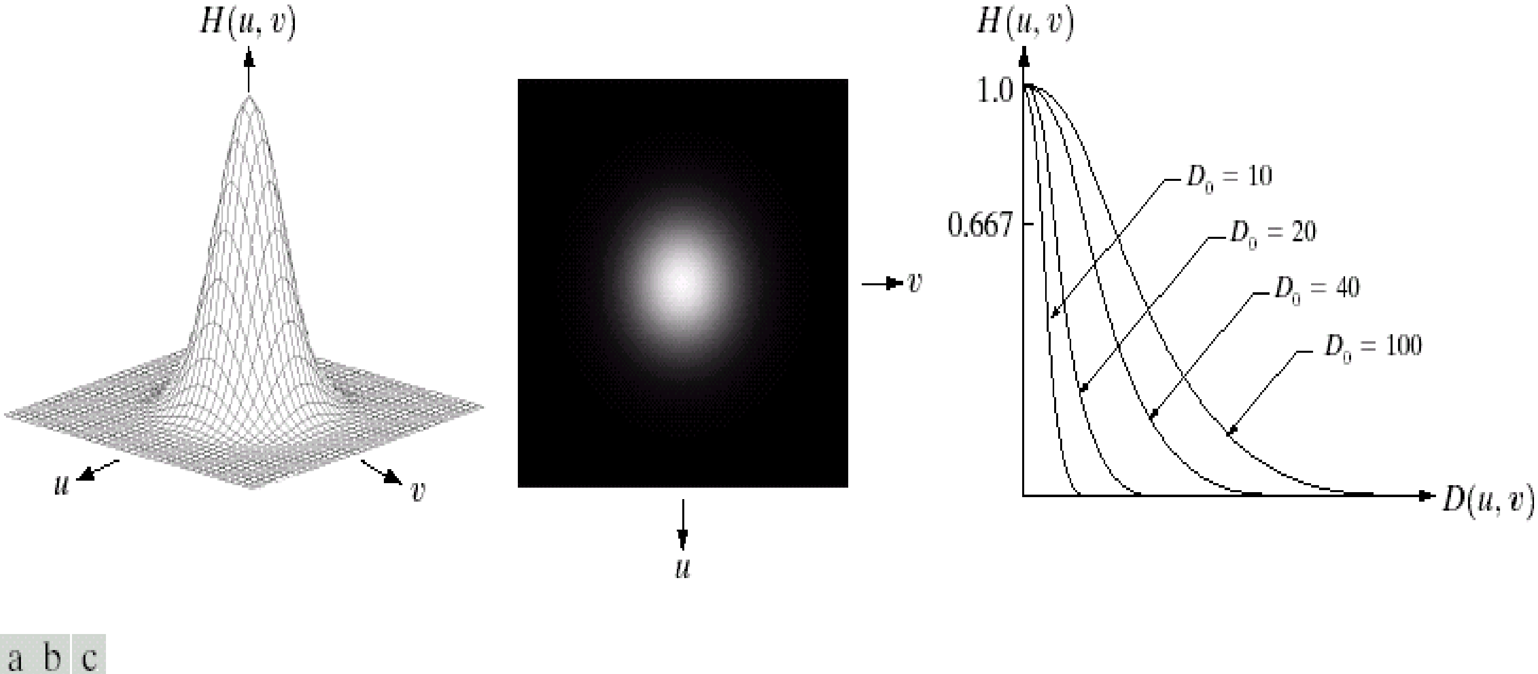


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

GLPF (cont')

Smooth transition in blurring as a function of increasing cutoff frequency.

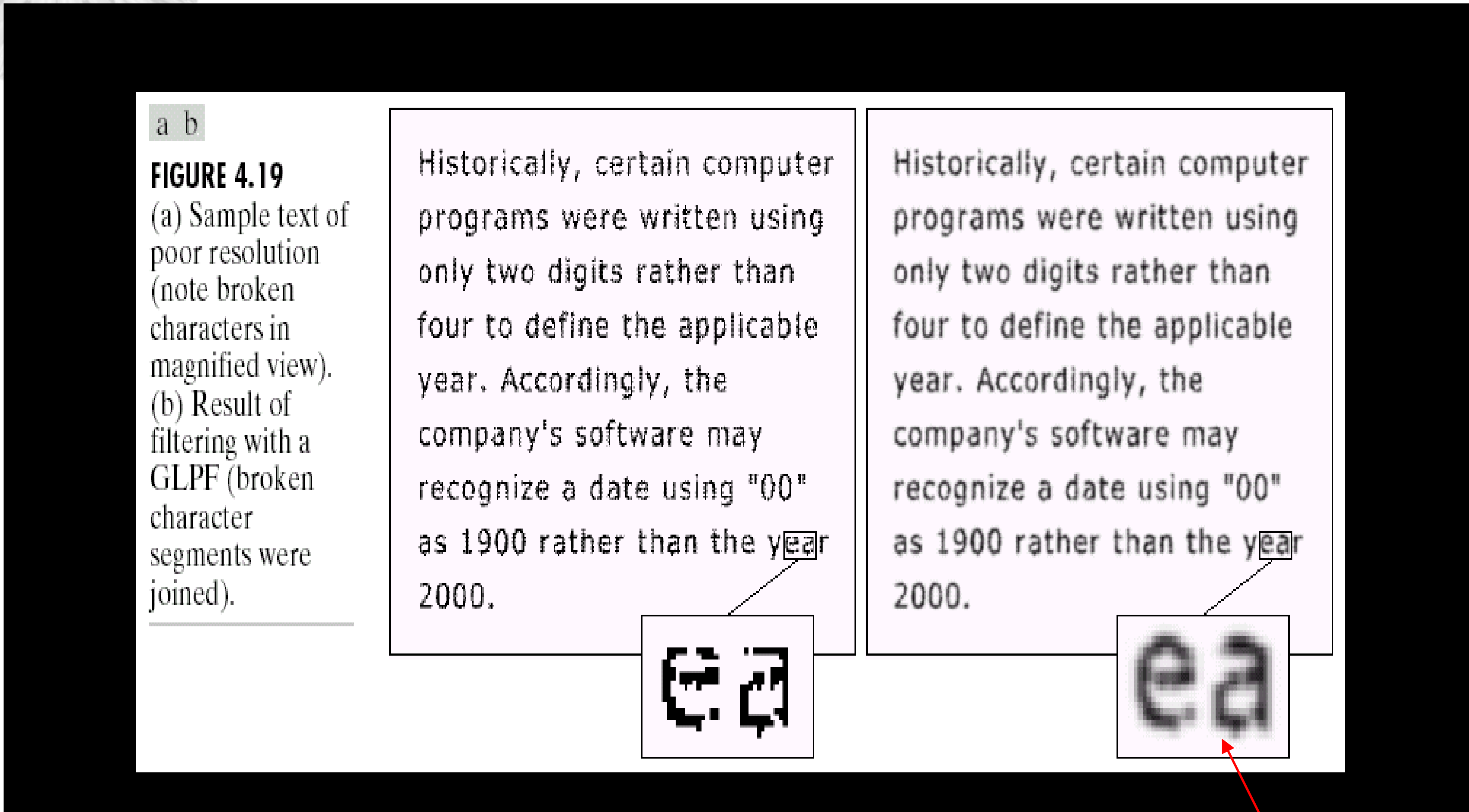
No ringing appears

-Suitable as an analysis tool

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

Example: GLPF in Character Recognition



using D_0=80

Such problem may occur in fax transmission, duplicated materials, historical records, etc.

Example: GLPF in Printing & Publishing Industry

Results in a smoother, softer-looking from a sharp original.

For human faces, the objective is to reduce the sharpness of fine skin lines and small blemishes.

Significant reduction in fine skin lines around the eyes

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).

Example: GLPF in Satelite & Arial Images

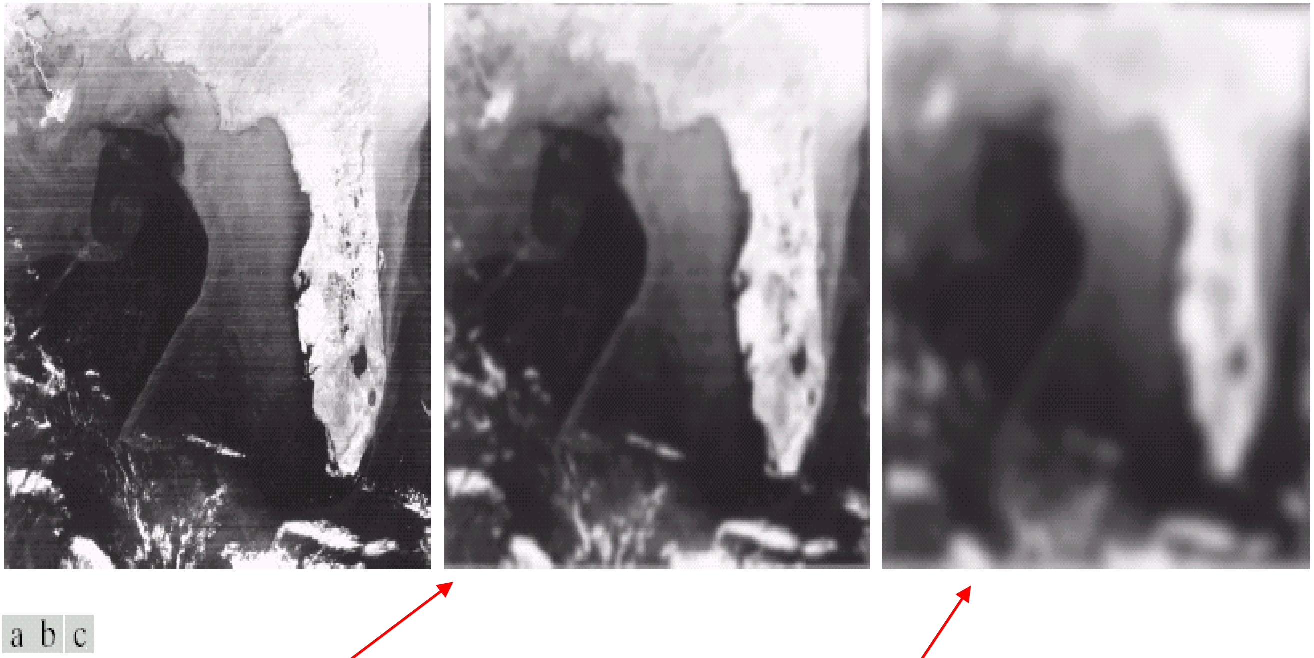


FIGURE 4.21 (a) Image showing prominent scan lines. (b) Result of using a GLPF with $D_0 = 30$. (c) Result of using a GLPF with $D_0 = 10$. (Original image courtesy of NOAA.)

Objective is to simplify the detection of features like the interface boundaries between the ocean currents

Objective is to blur as much as possible while leaving large features recognisable, e.g. lake sizes (dark region)

Sharpening Frequency Domain Filters

Edges, noises and other abrupt changes in gray level are associated with high frequency component.

Thus, image sharpening can be accomplished in frequency domain by a *high pass filtering* process, which attenuates the low frequency components without disturbing the high frequency information in Fourier Transform.

In other words, performing the reverse operation of ILPF:

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

Perspective Plot of High Pass Filters

Ideal High Pass

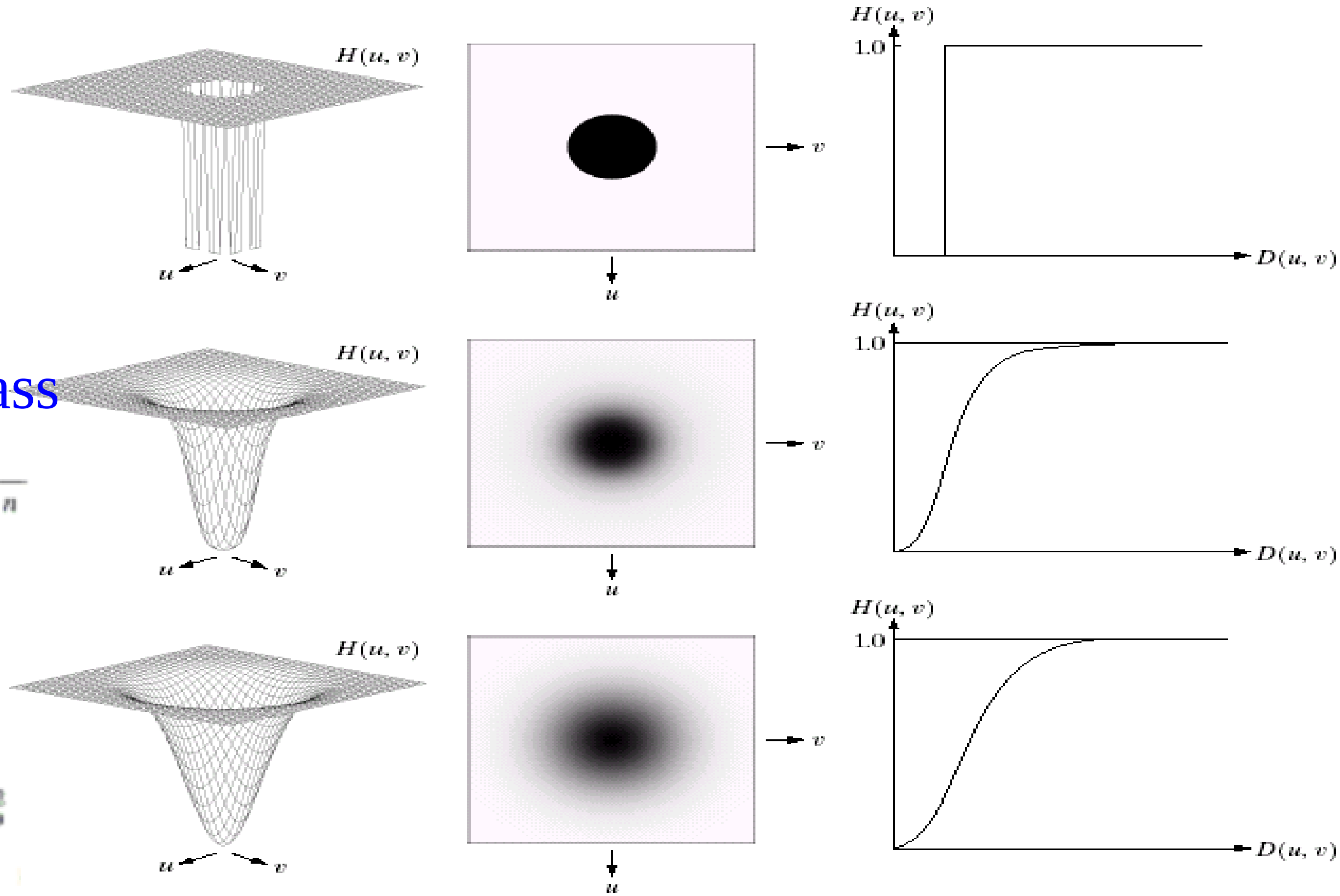
$$H(u,v)=\begin{cases}0 \text{ if } D(u,v)\le D_0\\ 1 \text{ if } D(u,v)>D_0\end{cases}$$

Butterworth High Pass

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

Gaussian High Pass

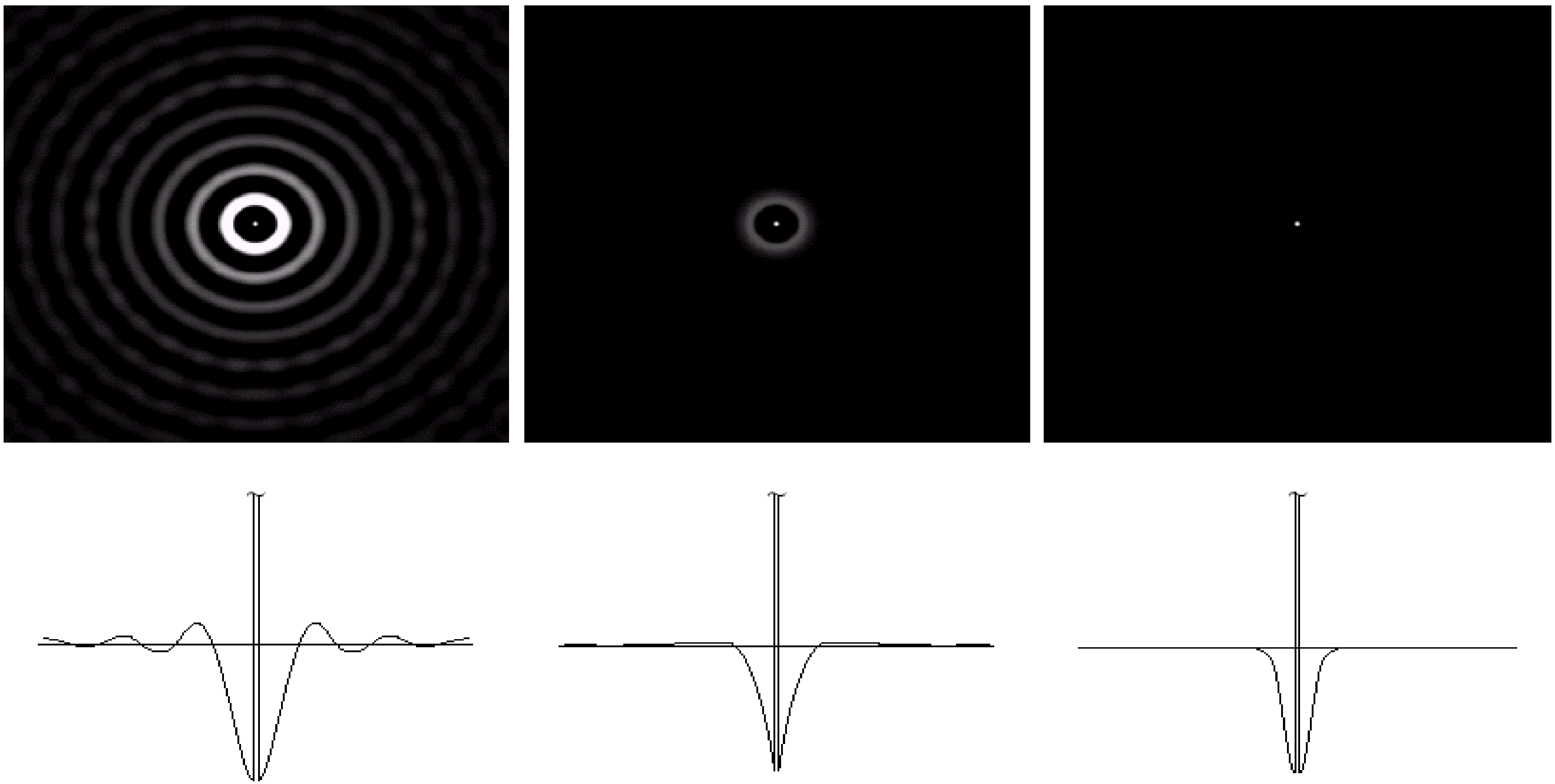
$$H(u,v)=1-e^{-D^2(u,v)/2D_0^2}$$



a	b	c
d	e	f
g	h	i

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.

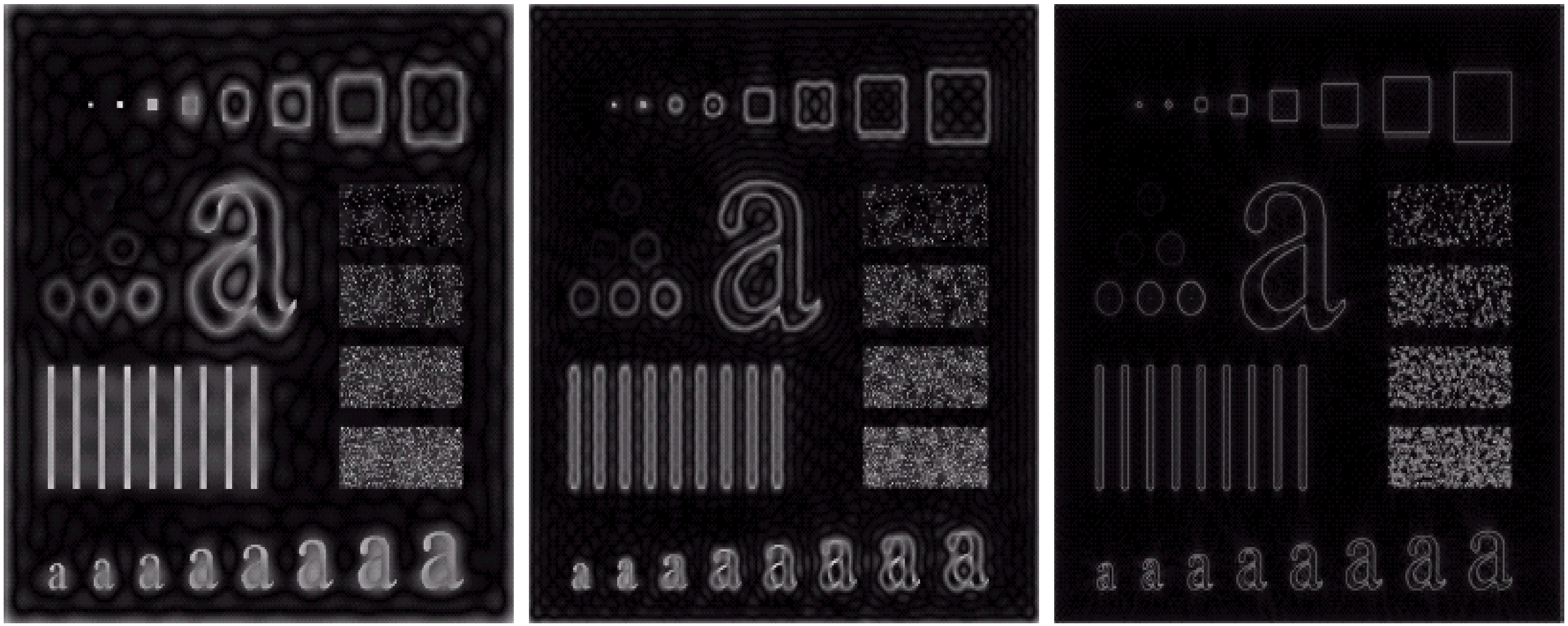
Spatial Domain Representation of an Ideal, Butterworth & Gaussian High Pass Filters



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.

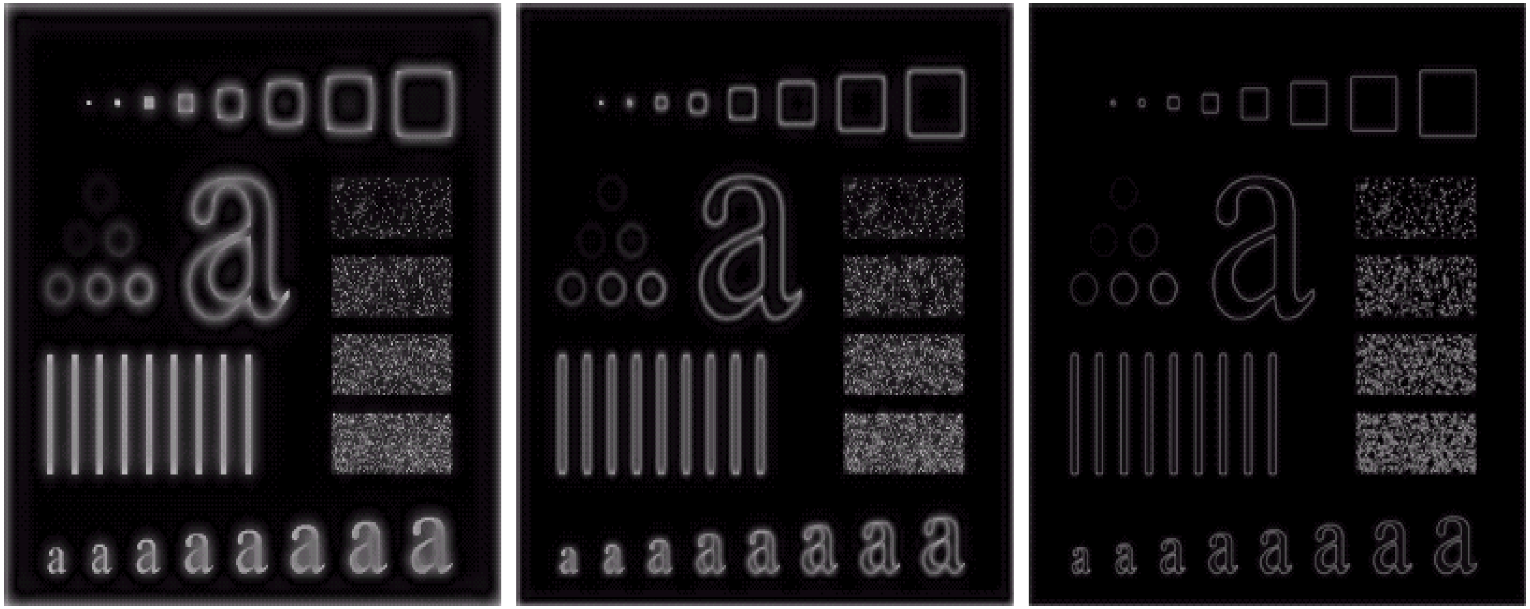
Example: Results from IHPF



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).

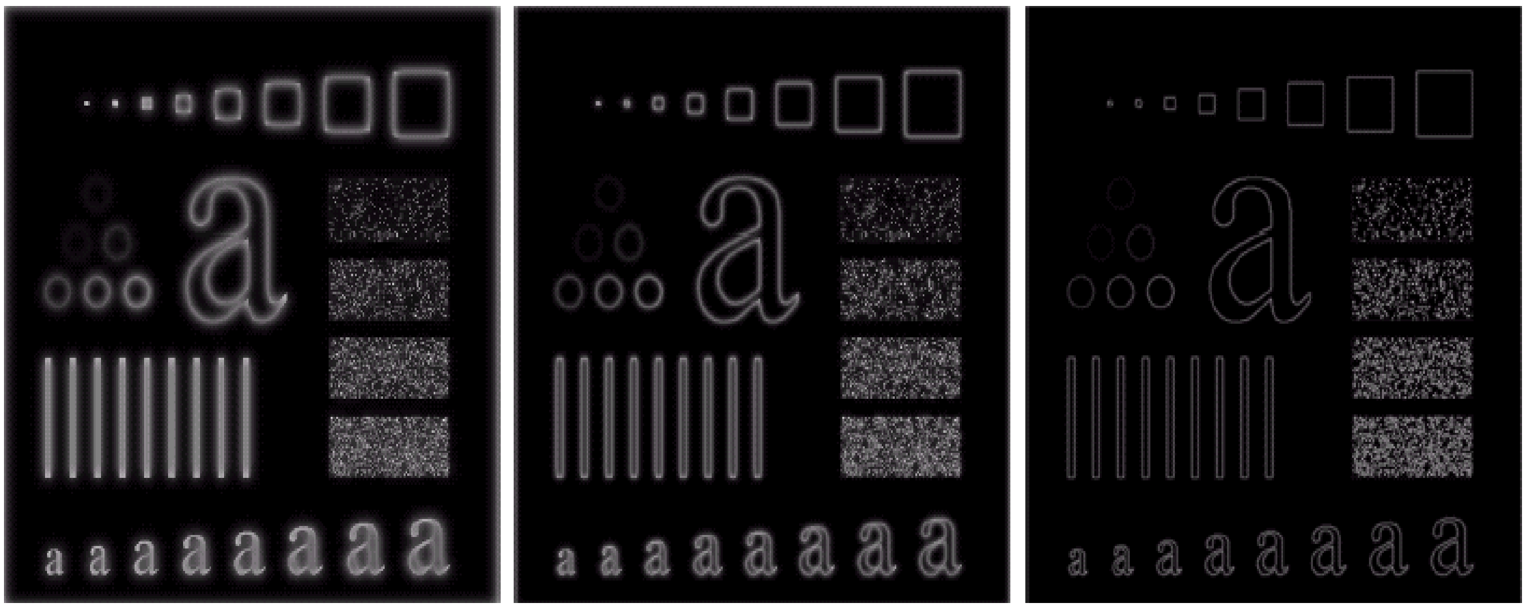
Example: Results from BHPF



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.

Example: Results from GHPF



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.

The Laplacian in the Frequency Domain

It can be shown that:

$$\Im\left[\frac{d^n f(x)}{dx^n}\right] = (ju)^n F(u)$$

From this equation, it can be shown that

$$\Im\left[\frac{d^2 f(x,y)}{dx^2} + \frac{d^2 f(x,y)}{dy^2}\right] = (ju)^2 F(u,v) + (jv)^2 F(u,v)$$

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

Thus, the Laplacian of f(x, y) could be implemented inside Fourier Transform :

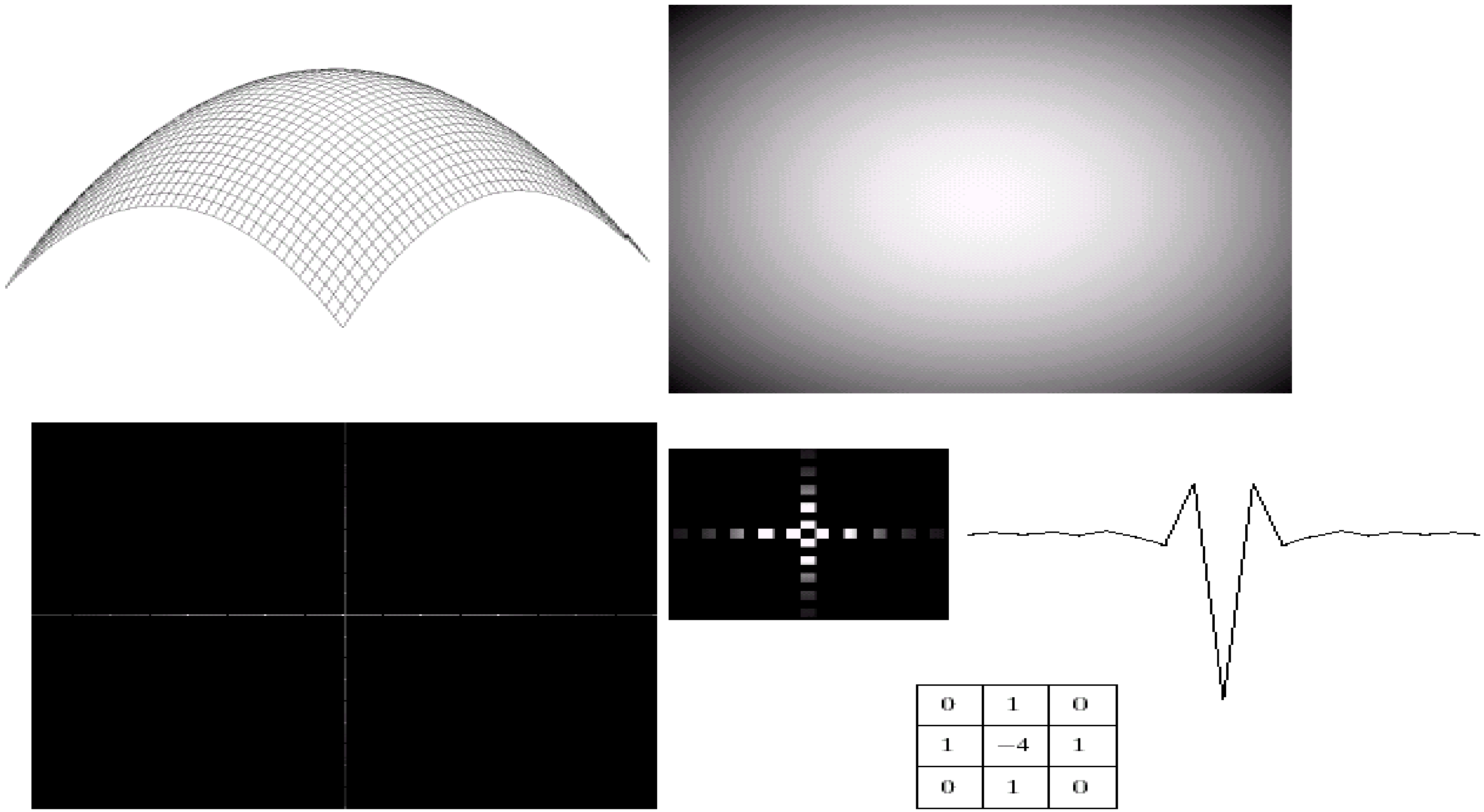
$$\Im\left[\nabla^2 f(x,y)\right] = -(u^2 + v^2) F(u,v)$$

which simply said that the Laplacian can be implemented in the Frequency domain by :

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

All the filtering operations is assumed (in this chapter) to be carried out after the origin of the *F(u,v)* has been centered by performing the $f(x,y)(-1)^{x+y}$ *multiplication*.

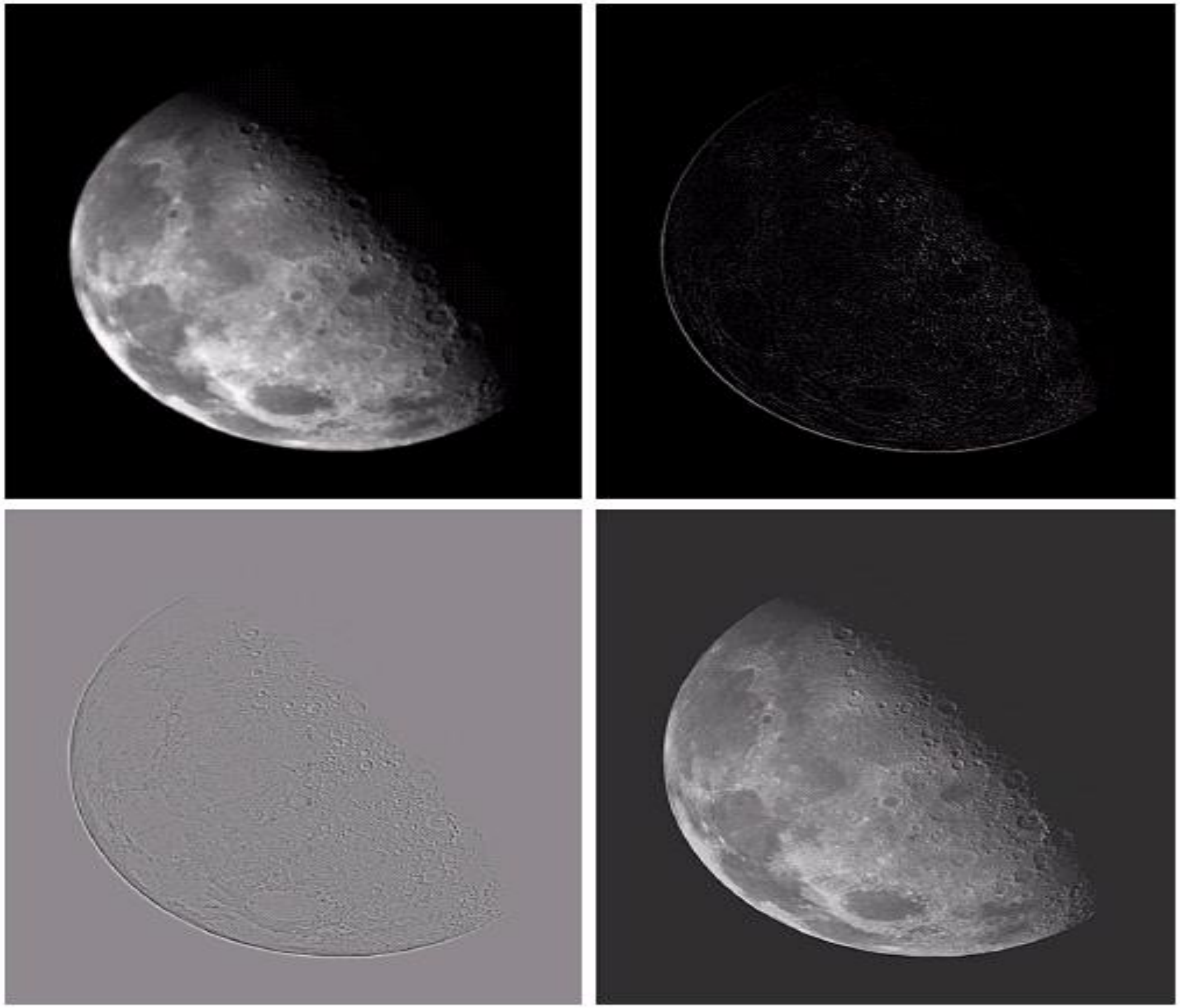
The Laplacian in the Frequency Domain



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.

Example: Laplacian Filtered Image



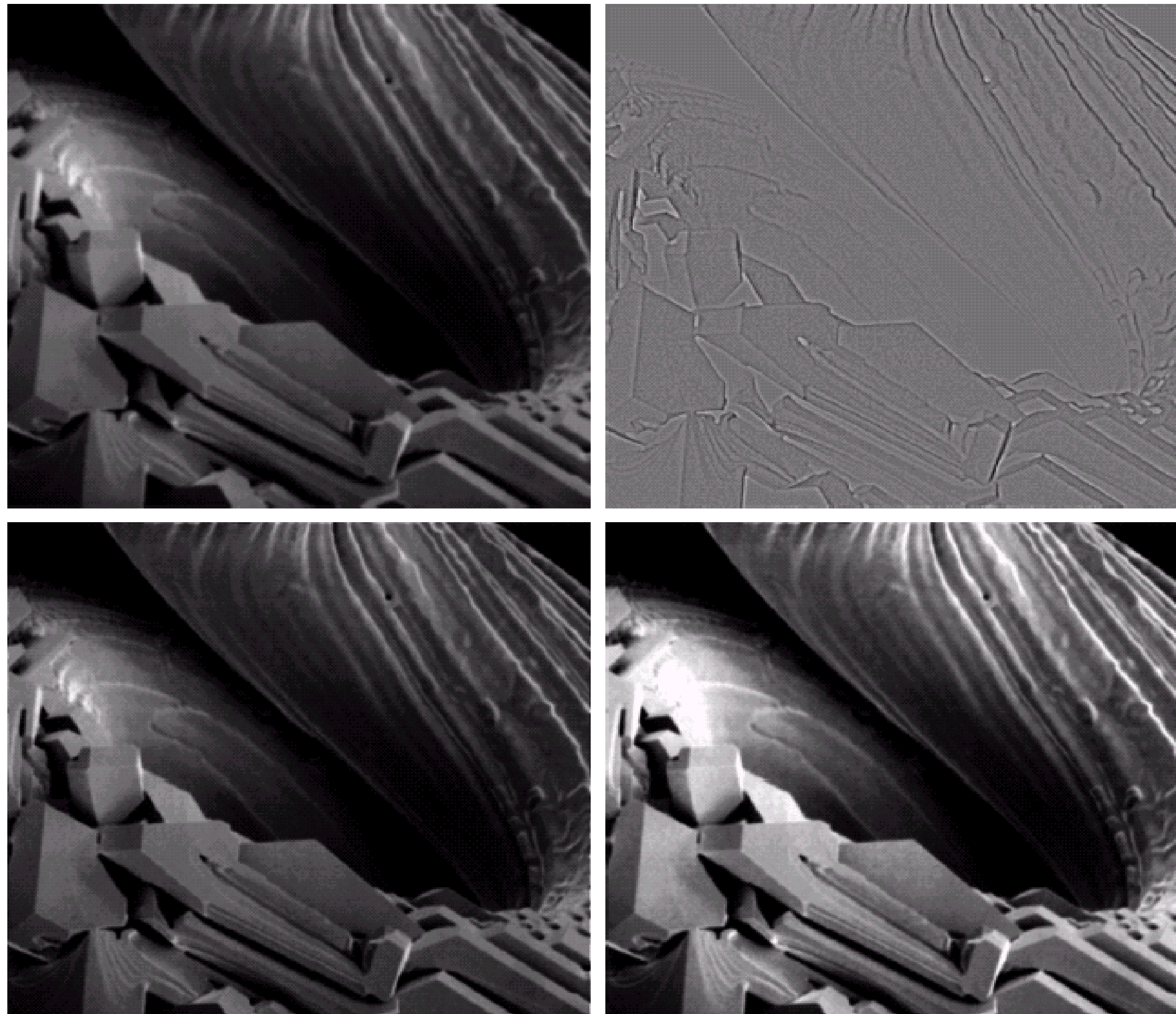
a b
c d

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.)

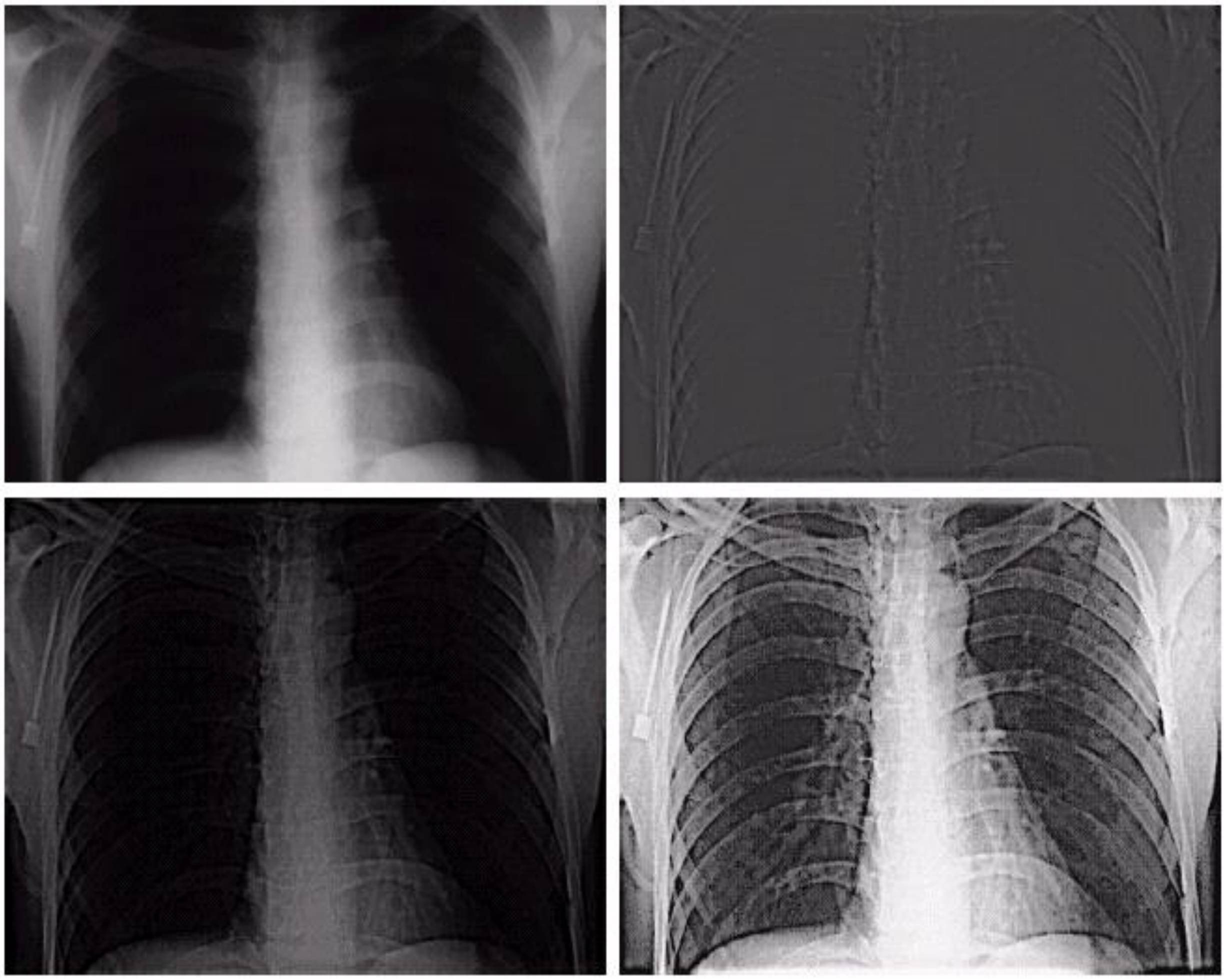
Example: Filtered Image

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.)



Example: Filtered Image



a b
c d

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.)

Homomorphic Filter

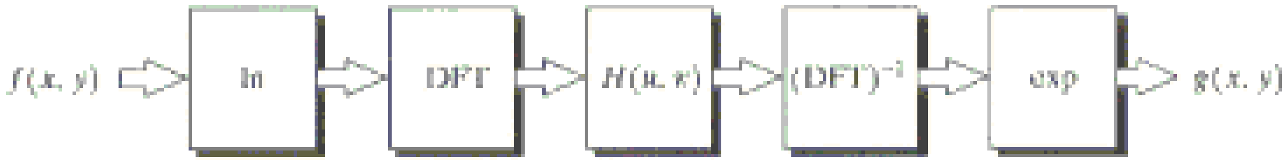


FIGURE 4.31 Homomorphic filtering approach for image enhancement.

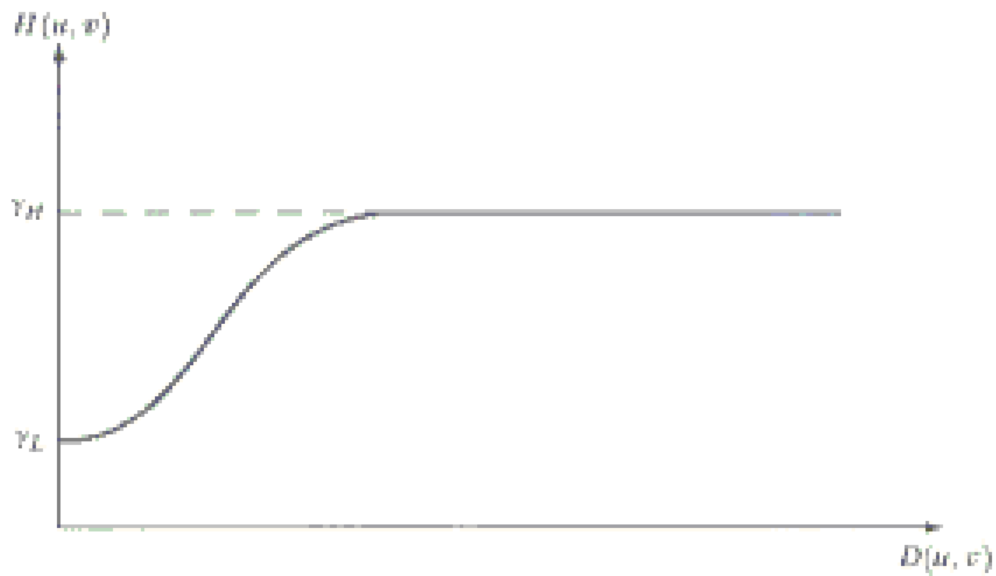


FIGURE 4.32 Cross section of a circularly symmetric filter function. $D(u, v)$ is the distance from the origin of the centered transform.

Result of Homomorphic Filter

a b

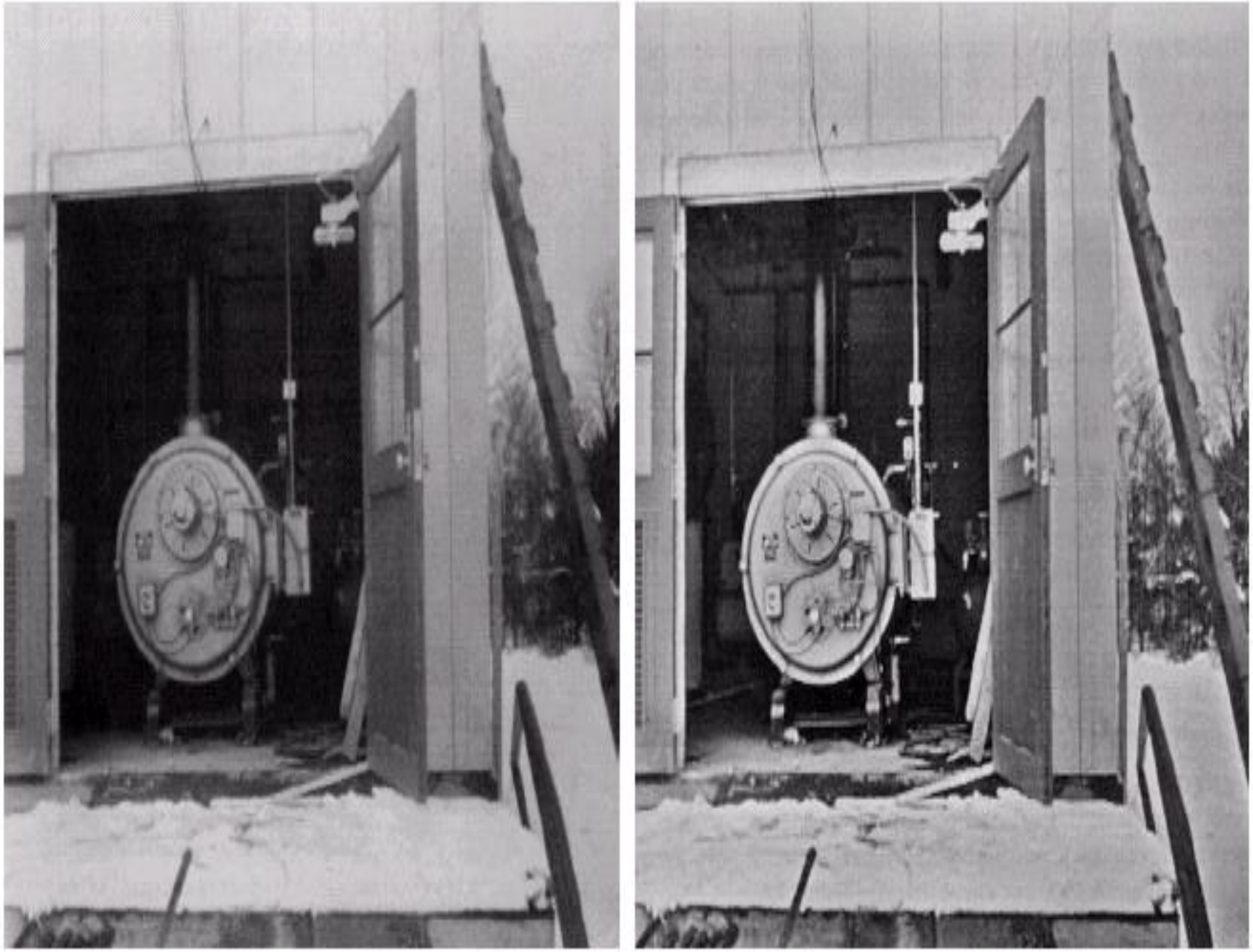
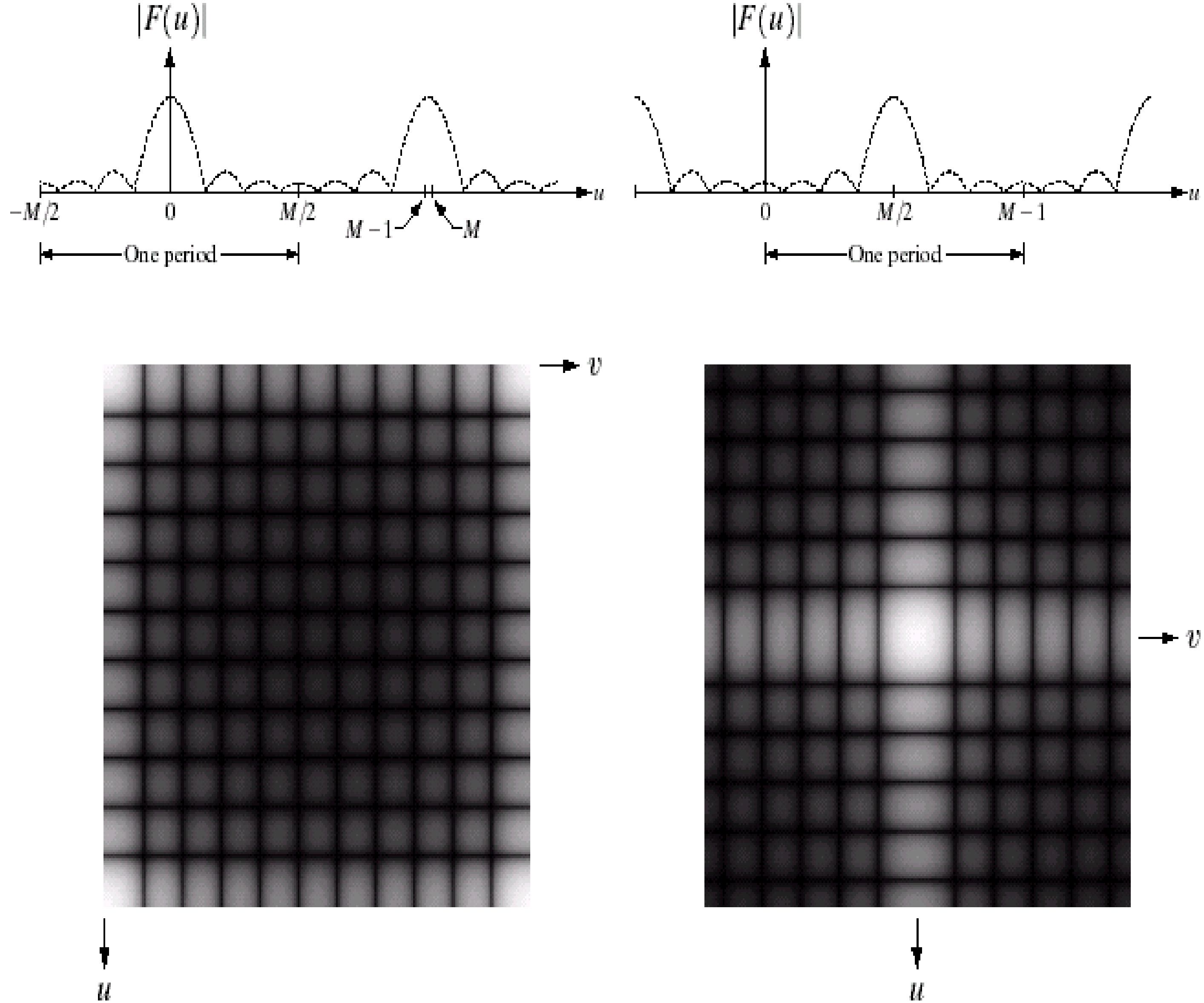


FIGURE 4.33 (a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)

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.



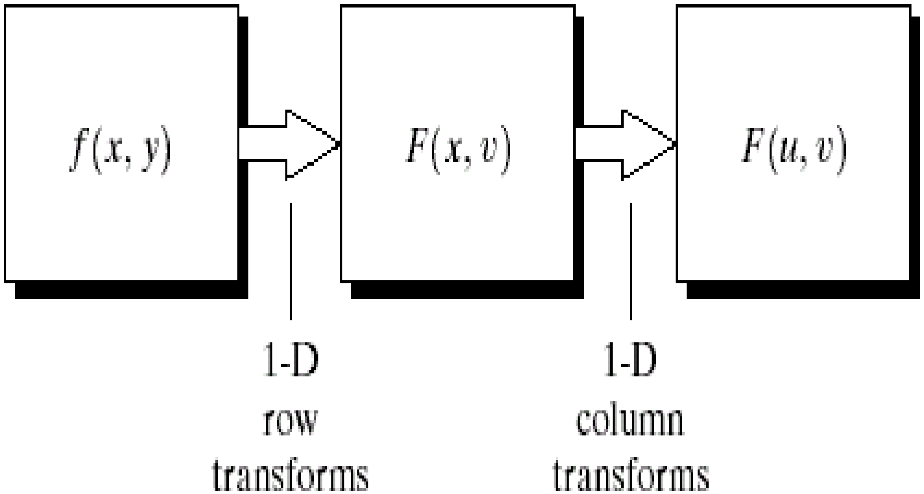


FIGURE 4.35 Computation of the 2-D Fourier transform as a series of 1-D transforms.

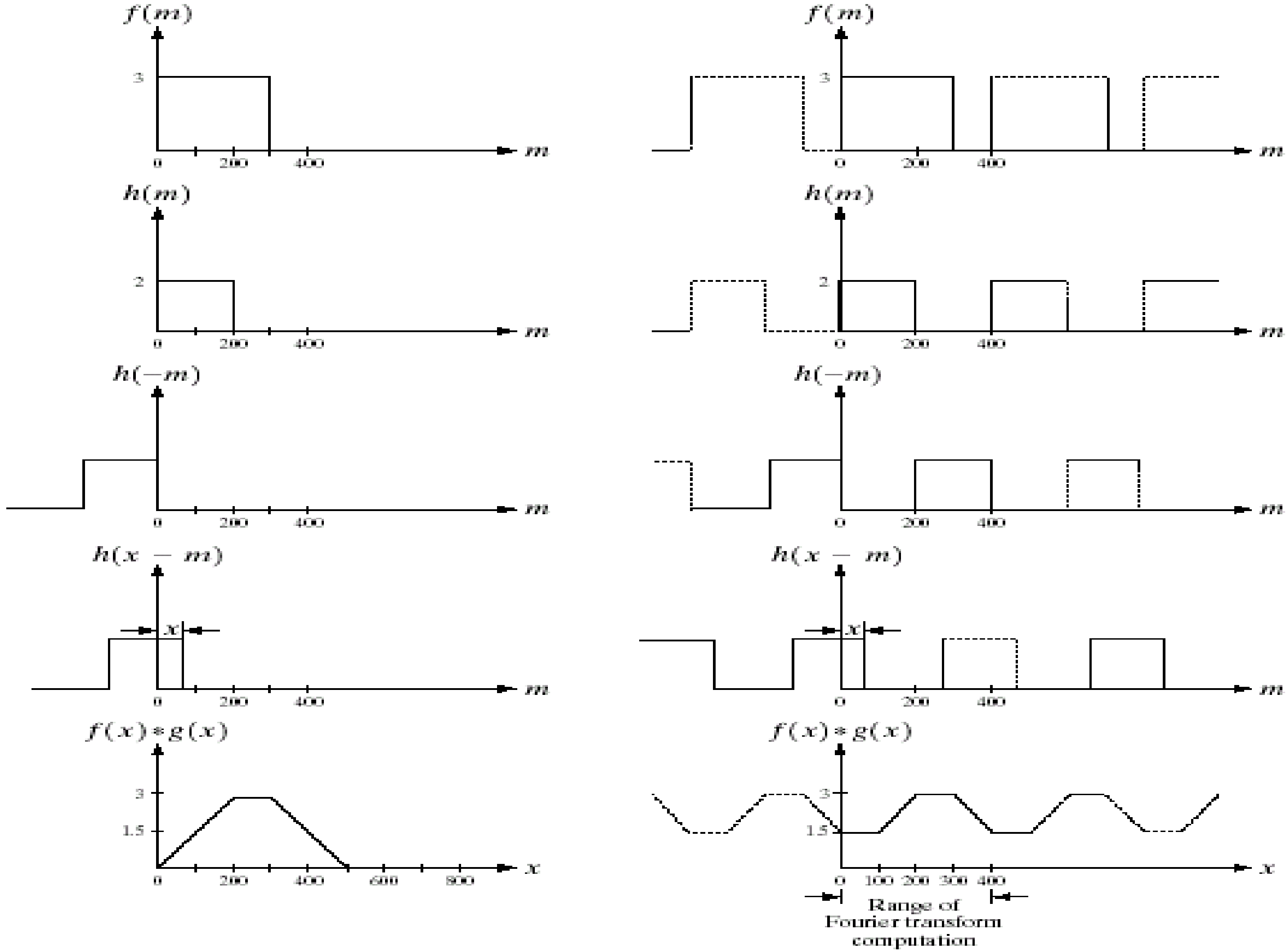


FIGURE 4.36 Left: convolution of two discrete functions. Right: convolution of the same functions, taking into account the implied periodicity of the DFT. Note in (j) how data from adjacent periods corrupt the result of convolution.

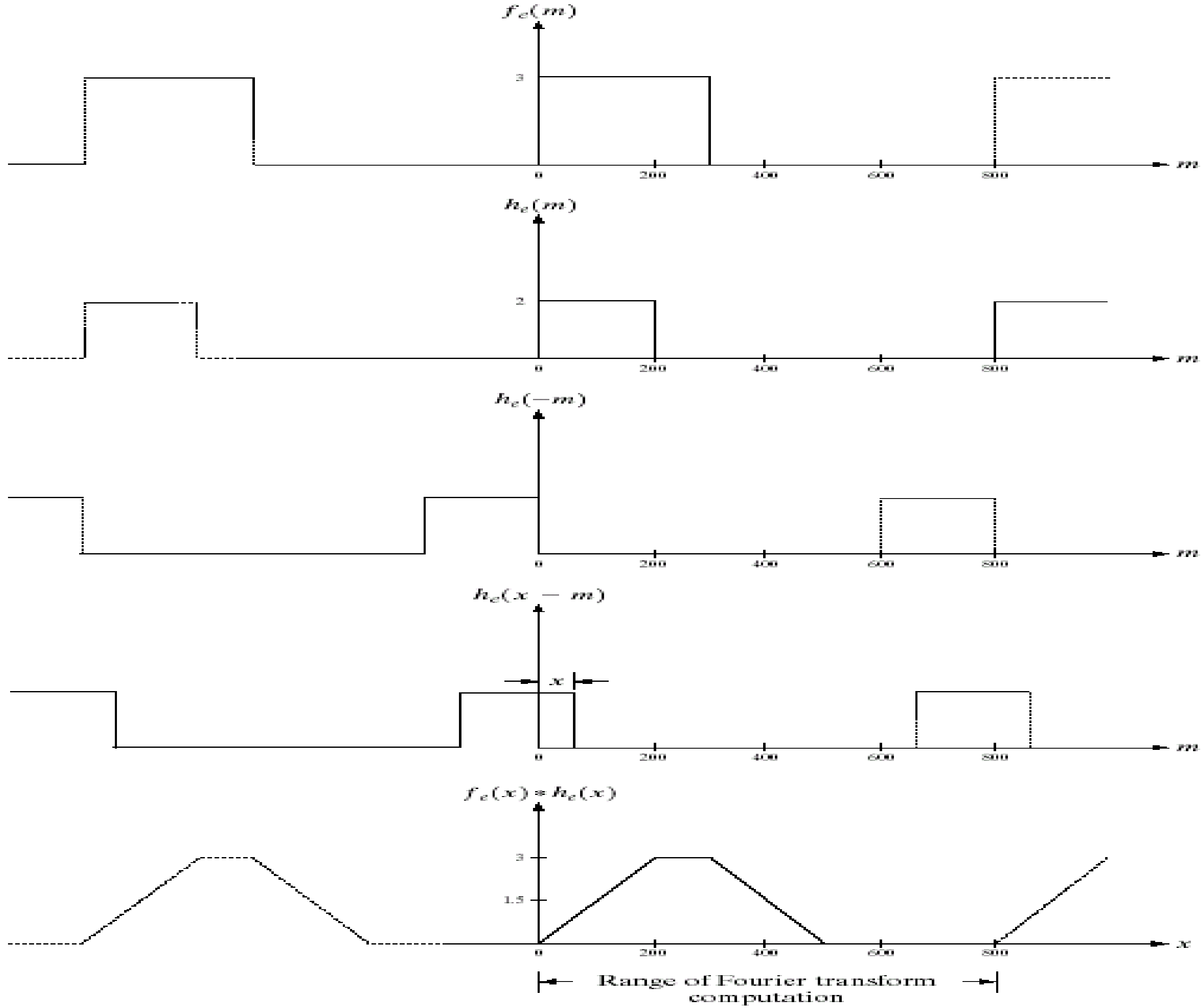
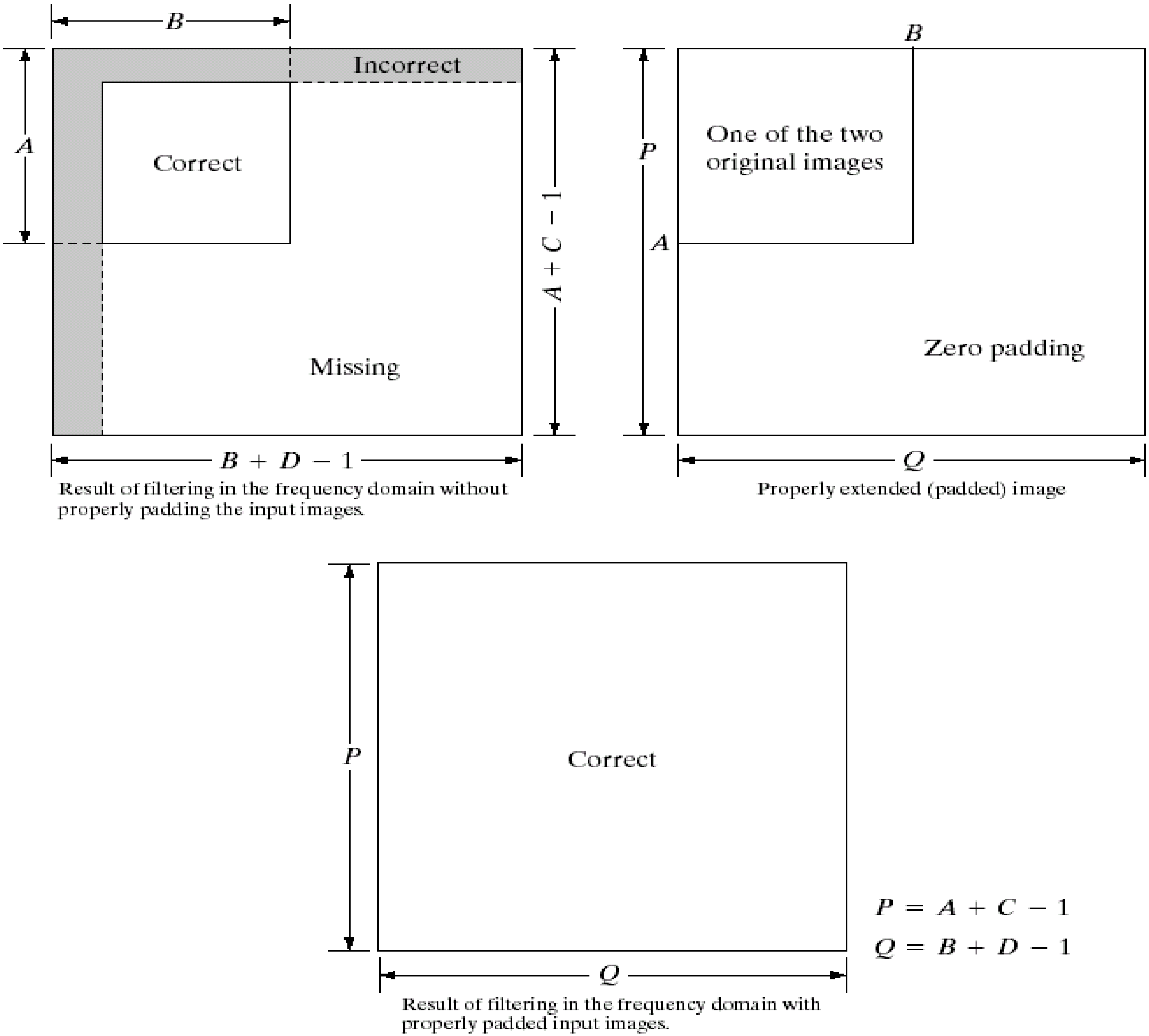


FIGURE 4.37 Result of performing convolution with extended functions. Compare Figs. 4.37(e) and 4.36(e).



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.

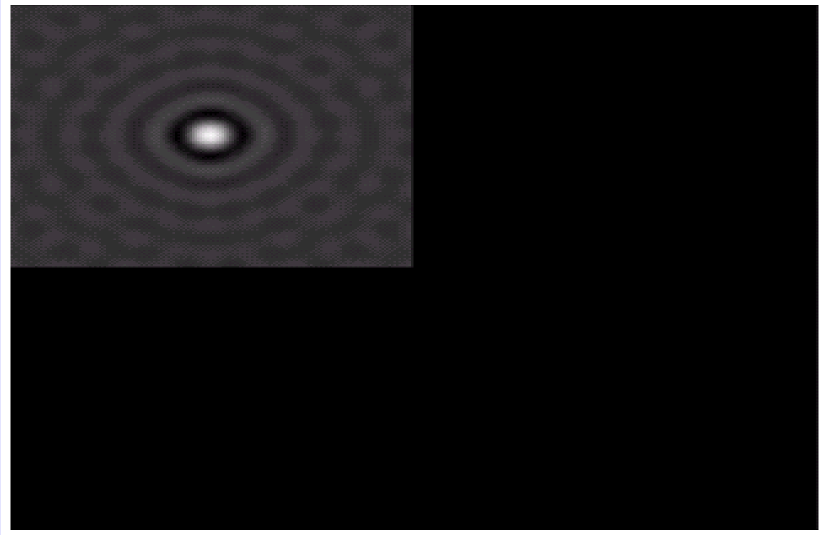


FIGURE 4.39 Padded lowpass filter is the spatial domain (only the real part is shown).

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.

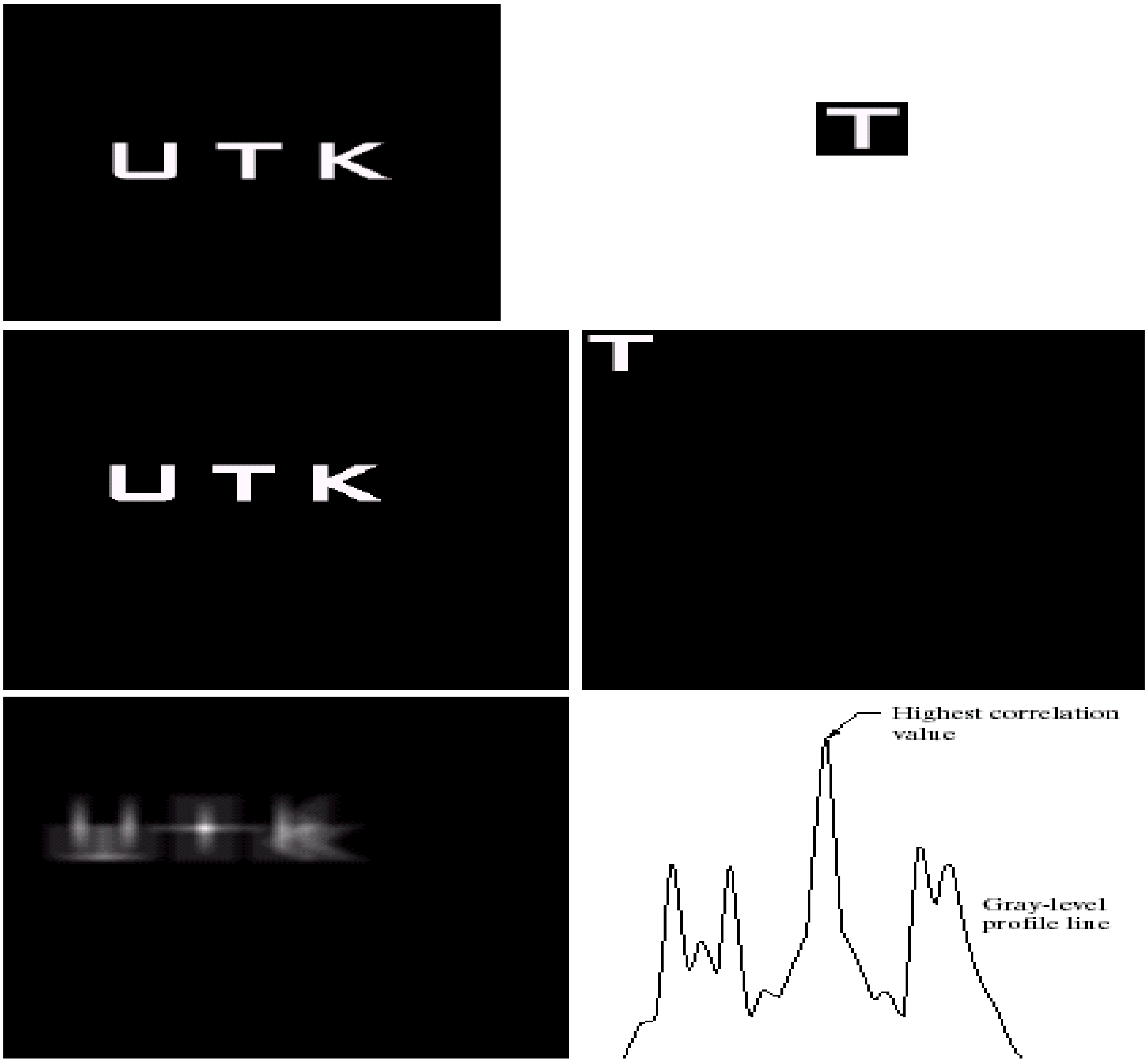


FIGURE 4.41 (a) Image. (b) Template. (c) and (d) Padded images. (e) Correlation function displayed as an image. (f) Horizontal profile line through the highest value in (e), showing the point at which the best match took place.

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	$f(x,y)e^{j2\pi(u_0x/M+v_0y/N)} \Leftrightarrow F(u-u_0, v-v_0)$ $f(x-x_0, y-y_0) \Leftrightarrow F(u,v)e^{-j2\pi(ux_0/M+vy_0/N)}$ When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then $f(x,y)(-1)^{x+y} \Leftrightarrow F(u-M/2, v-N/2)$ $f(x-M/2, y-N/2) \Leftrightarrow F(u,v)(-1)^{u+v}$

TABLE 4.1 *(continued)*

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	$x = r\cos\theta \quad y = r\sin\theta \quad u = \omega\cos\varphi \quad v = \omega\sin\varphi$ $f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$
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.

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	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow \frac{1}{2}\left[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)\right]$
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.

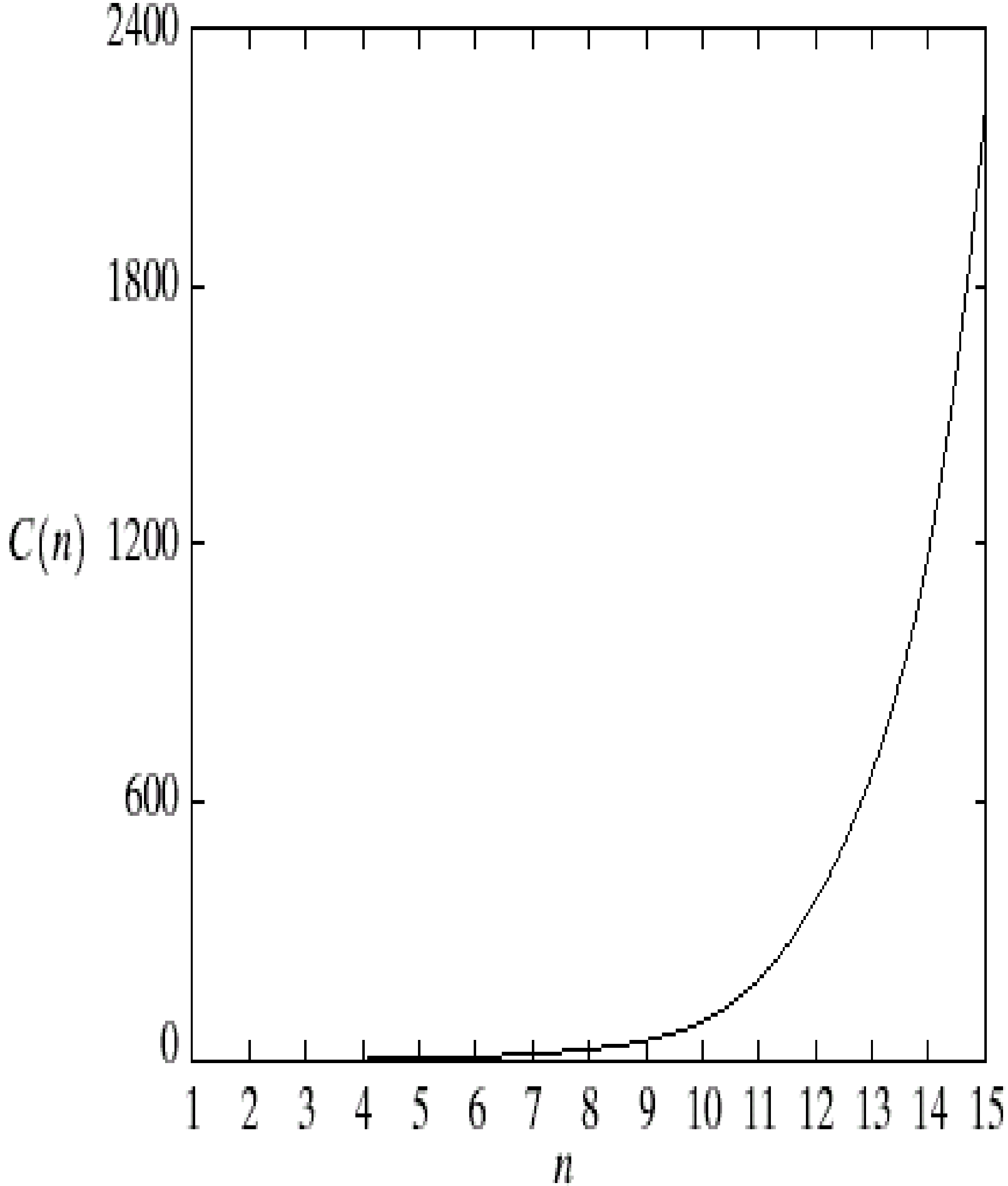


FIGURE 4.42 Computational advantage of the FFT over a direct implementation of the 1-D DFT. Note that the advantage increases rapidly as a function of n.

END

CHAPTER 4