Chapter 4 Image Enhancement in the Frequency Domain

4.1 Background

- **Fourier series**
  - Any functions that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient
- **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

4.1 Background

- Fourier analysis

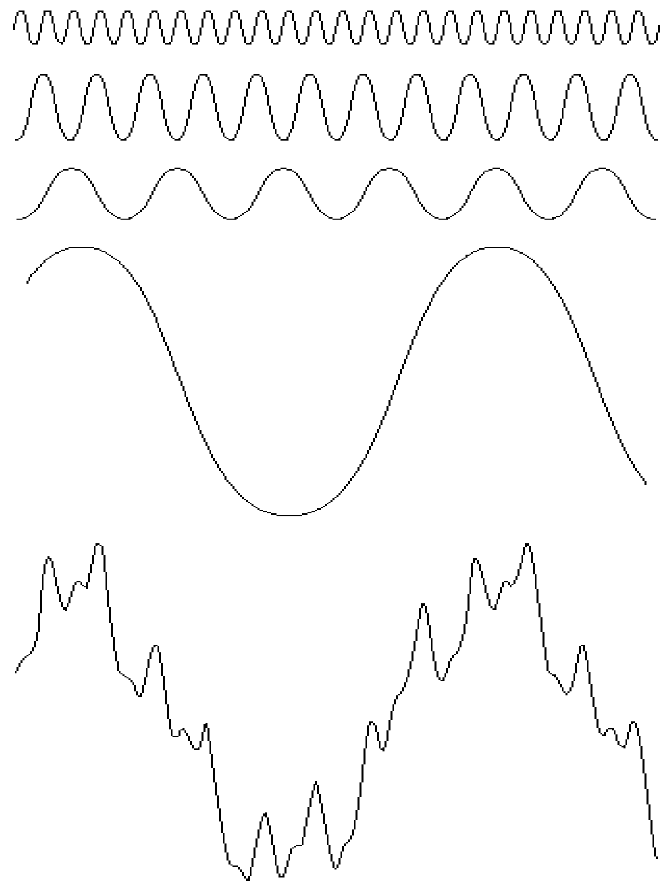


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

4.2 Fourier transform

4.2.1 1-D discrete Fourier transform

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

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

    - Note that $j = \sqrt{-1}$
  - We can obtain the original function back using the inverse DFT, given $F(u)$:

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

4.2.1 1-D discrete Fourier transform

- Magnitude spectrum, Phase spectrum

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{-j2\pi ux/M} \quad (\text{Euler's formula} : e^{j\theta} = \cos\theta + j\sin n\theta)$$

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

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

$$= R(u) + jI(u)$$

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

4.2.1 1-D discrete Fourier transform

- Magnitude spectrum, Phase spectrum
  - Magnitude (spectrum) of the transform

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

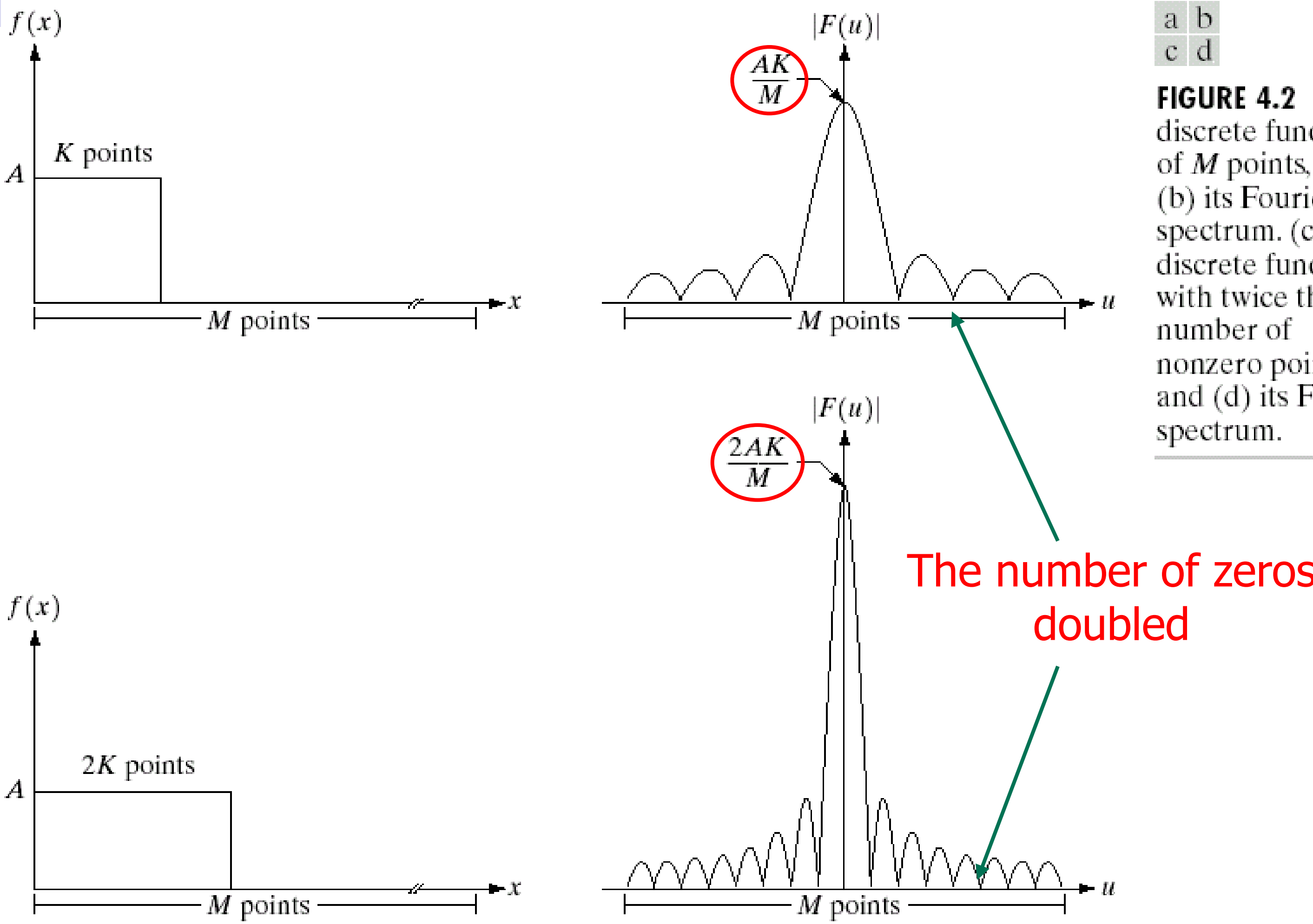
  - Phase angle (phase spectrum)

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

  - Power spectrum (spectral density)

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

4.2.1 1-D discrete Fourier 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.

4.2.1 1-D discrete Fourier transform

- $f(x) = f(x_0 + x \cdot Dx)$
  - x_0 : the first sample in the sequence
  - Δx: sampling period
  - $1/\Delta x$: sampling rate f_s
- $F(u) = F(u \cdot \Delta u)$
  - $\Delta u = f_s/M$
- The relationship between Δx and Δu :
  - $\Delta u = f_s/M = (1/\Delta x)/(M) = 1/(M\Delta x)$

4.2.2 2-D DFT

- 2-D discrete Fourier transform (2-D DFT)

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

$$u = 0,1,\ldots,M-1,\ v = 0,1,\ldots,N-1$$

  - u, v: transform (frequency) variables
  - x, y: spatial (image) variables
- 2-D inverse discrete Fourier transform (2-D IDFT)

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

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

4.2.2 2-D DFT

- Magnitude spectrum, Phase spectrum

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

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

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

- DC (direct current): F(0,0)

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

  - Average of f(x, y) (average gray level of the image)

4.2.1 1-D discrete Fourier transform

- Magnitude spectrum, Phase spectrum
  - Magnitude (spectrum) of the transform

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

  - Phase angle (phase spectrum)

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

  - Power spectrum (spectral density)

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

4.2.2 2-D DFT

- It can be shown that

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

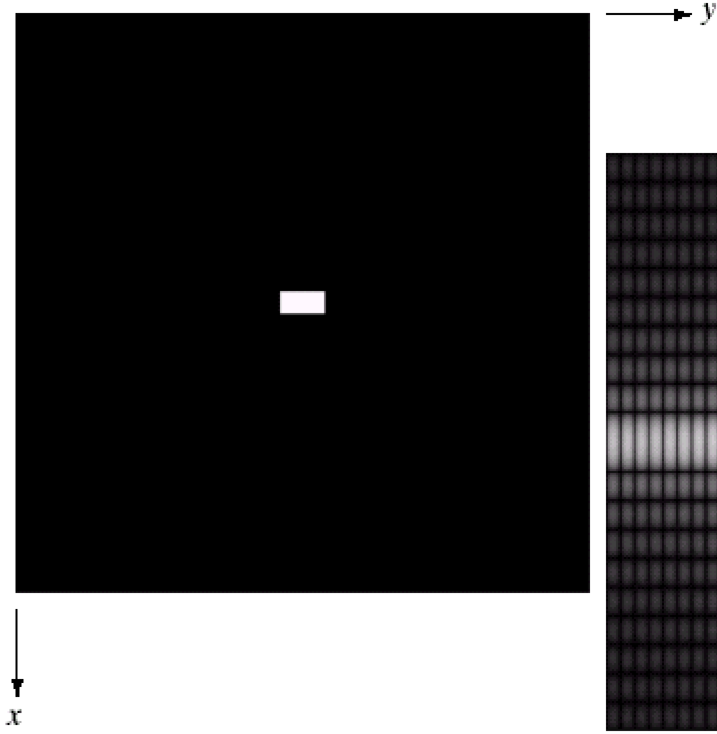
  - $\Im[\bullet]$: Fourier transform of the argument
  - The origin of the Fourier transform of $f(x, y)(-1)^{x+y}$ is located at $u = M/2$ and $v = N/2$
  - Multiplying $f(x, y)$ by $(-1)^{x+y}$ shifts the origin of $F(u, v)$ to frequency coordinates $(M/2, N/2)$

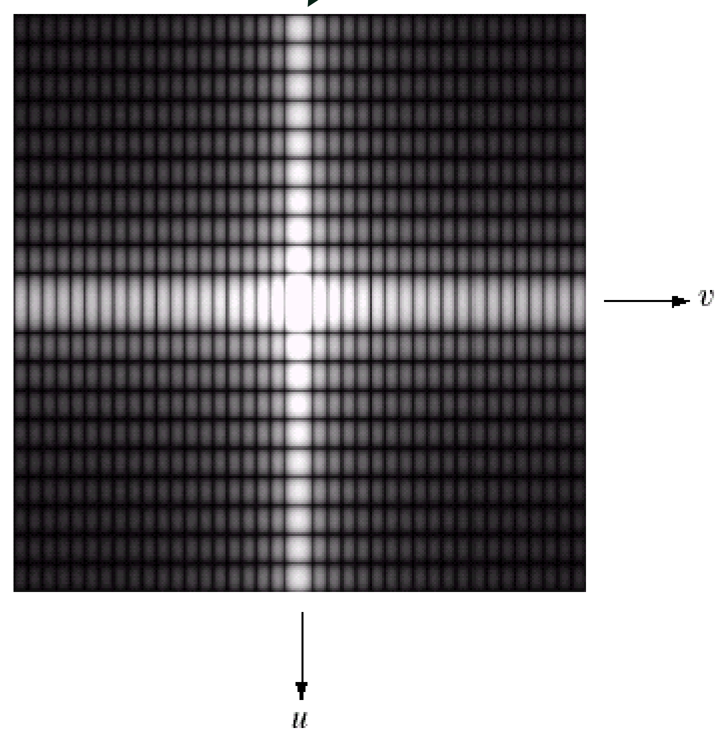
4.2.2 2-D DFT

- The relationship between Δx and Δu :
  - $f(x, y) = f(x\Delta x, y\Delta y)$,
  - $F(u,v) = F(u\,\Delta u, v\,\Delta v)$
  - $\Delta u = 1/(M\Delta x)$, $\Delta v = 1/(N\Delta y)$
- Implementation consideration for filtering processing: centering problem
  - $G(u,v) = F(u,v)H(u,v)$,
  - $F(u,v) = \mathfrak{J}[f(x,y)(-1)^{x+y}]$

4.2.2 2-D DFT

The separation of zeros in *u*-direction is twice the separation of zeros in *v*-direction

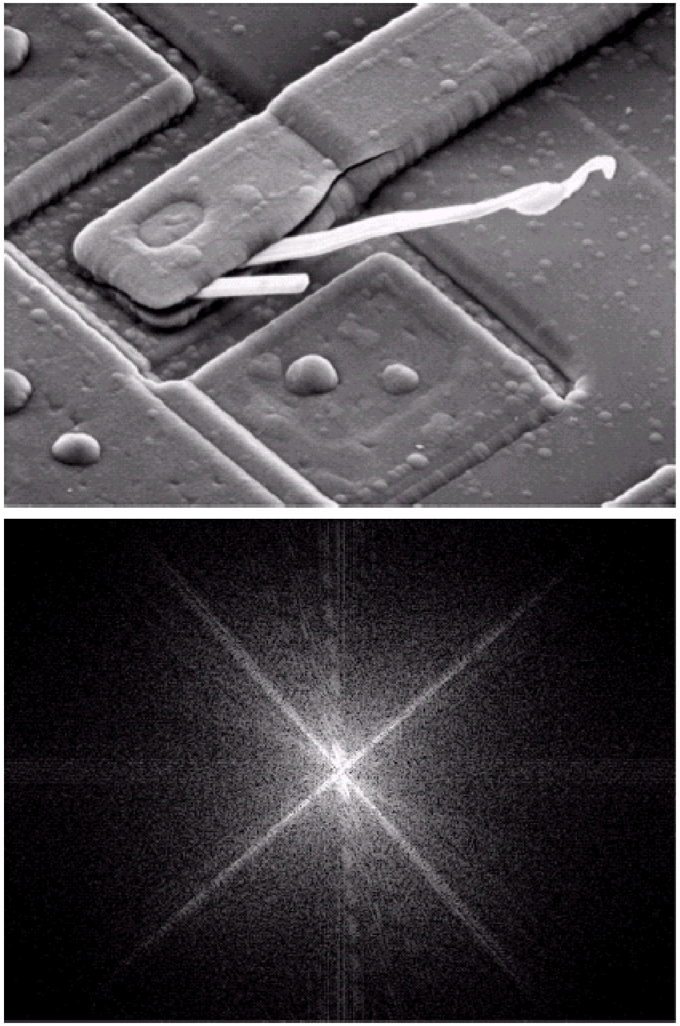




a b

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.

4.2.3 Filtering in the 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.)

4.2.3 Filtering in the frequency domain

- Basics of filtering in the frequency domain
  1. Perform $f(x,y)\times(-1)^{x+y}$
  2. DFT: $F(u,v) = \mathfrak{I}[f(x,y)(-1)^{x+y}]$
  3. Multiply F(u,v) by a filter function:
     $G(u,v) = F(u,v)\,H(u,v)$
  4. IDFT: $g(x,y) = \mathfrak{I}^{-1}[G(u,v)]$
  5. Perform $g(x,y)\times(-1)^{x+y}$

- H(u,v): filter (filter transfer function)
  - suppresses certain frequencies in the transform while leaving others unchanged

4.2.3 Filtering in the frequency domain

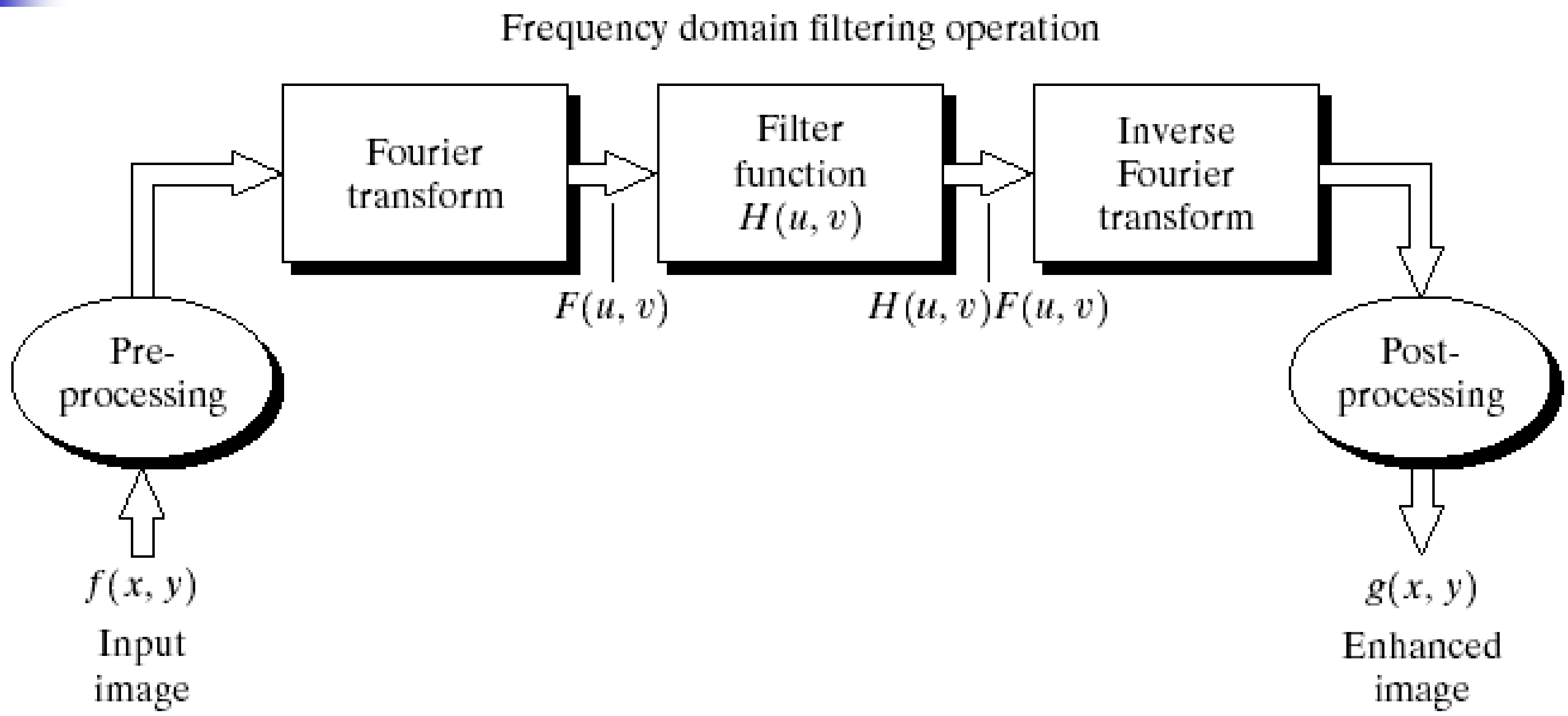


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Some basic filters

- Notch filter – a constant function with a hole (notch) at the origin

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

- Set F(0, 0) to zero (an image with zero average value)

Some basic filters

- Notch filter

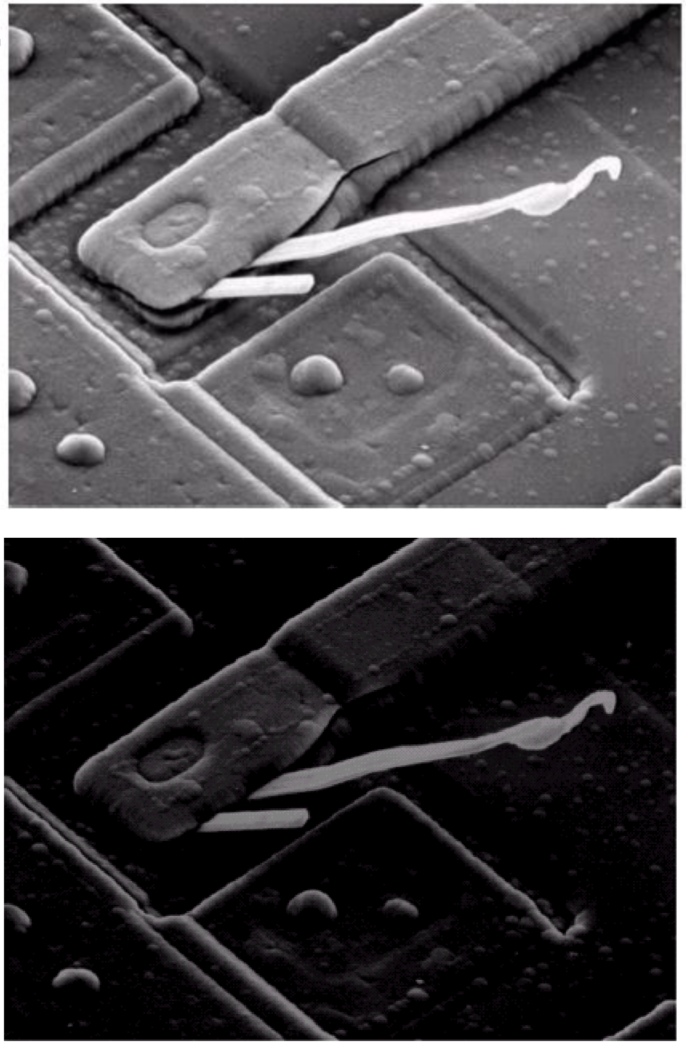


FIGURE 4.6 Result of filtering the image in Fig. 4.4(a) with a notch filter that set to 0 the $F(0, 0)$ term in the Fourier transform.

Some basic filters

1. Lowpass filter: attenuate high frequencies while "passing" low frequencies
   - Ideal Lowpass filter
   - Butterworth lowpass filter
   - Guassian lowpass filter
2. Highpass filter

Some basic filters

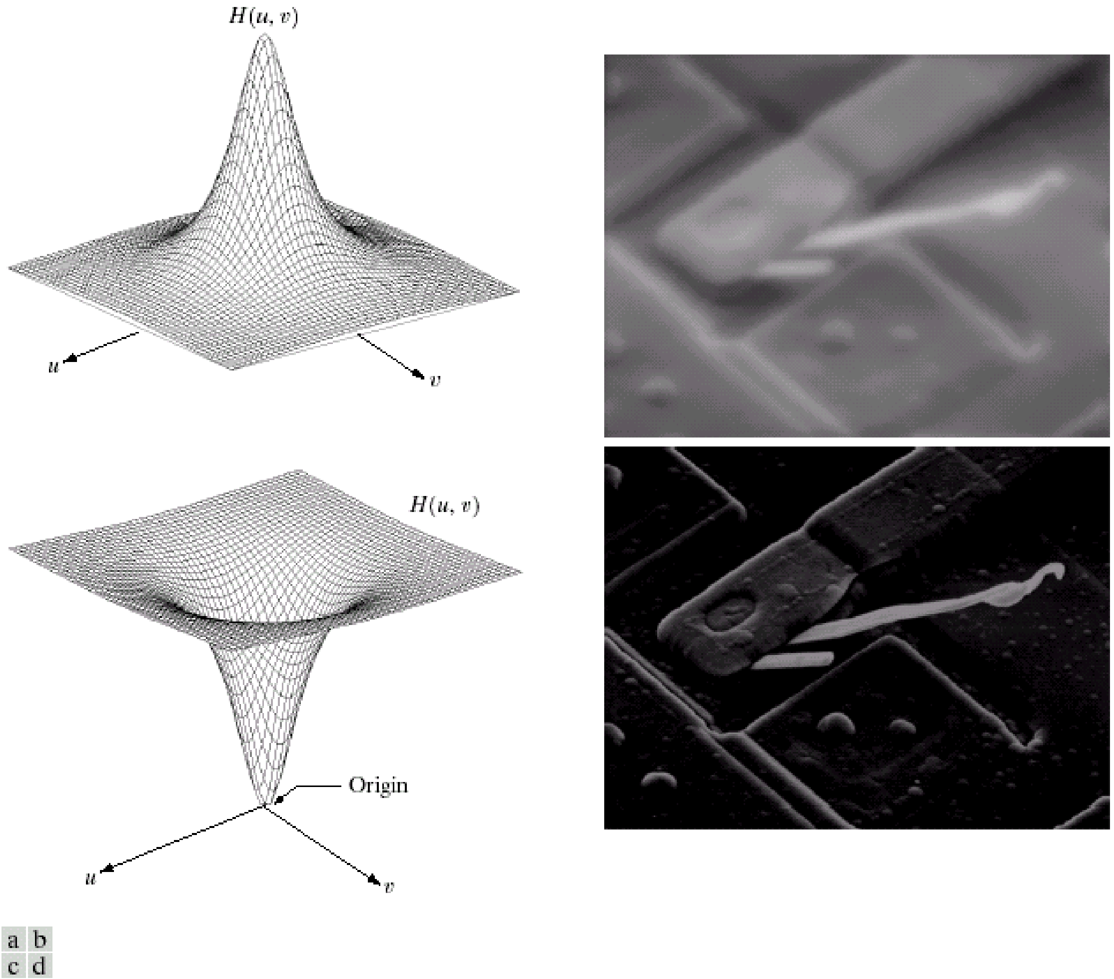
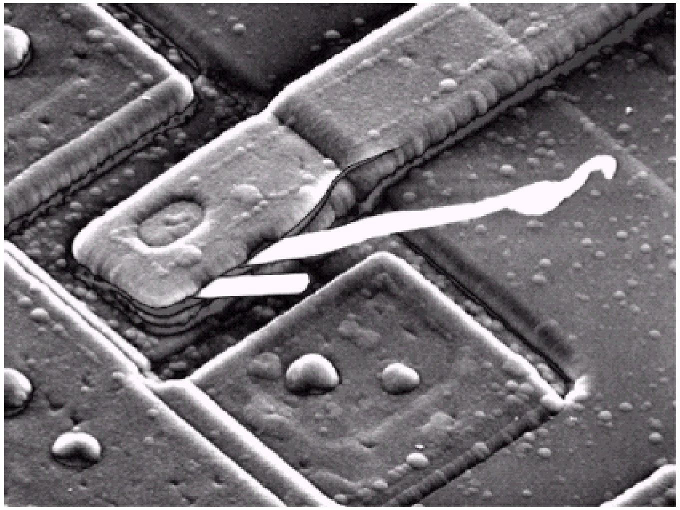


FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

Some basic filters

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).



4.2.4 Correspondence between filtering in the spatial and frequency domains

- **Convolution Theorem**
  - Discrete convolution

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

1. Flipping one function about the origin (h(m, n) $\rightarrow$ h(-m, -n))
2. Shifting that function with respect to the other by changing the values of (x, y) (h(-m, -n) $\rightarrow$ h(-m+x, -n+y))
3. Computing sum of products over all values of m and n, for each displacement (x, y)

4.2.4 Correspondence between filtering in the spatial and frequency domains

- **Convolution Theorem**
  - f(x,y) * h(x,y) <=> F(u,v) H(u,v)
  - f(x,y) h(x,y) <=> F(u,v) * H(u,v)

$$H(u) = Ae^{-\frac{u^2}{2\sigma^2}}, h(x) = \sqrt{2\pi}\sigma A e^{-2\pi^2\sigma^2 x^2}$$

$$H(u) = Ae^{-\frac{u^2}{2\sigma_1^2}} - Be^{-\frac{u^2}{2\sigma_2^2}}, A \geq B, \sigma_1 > \sigma_2$$

4.2.4 Correspondence between filtering in the spatial and frequency domains

- Ex. LPF, HPF
  - LPF:

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

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

  - HPF:

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

$$A \geq B, \sigma_1 > \sigma_2$$

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

4.2.4 Correspondence between filtering in the spatial and frequency domains

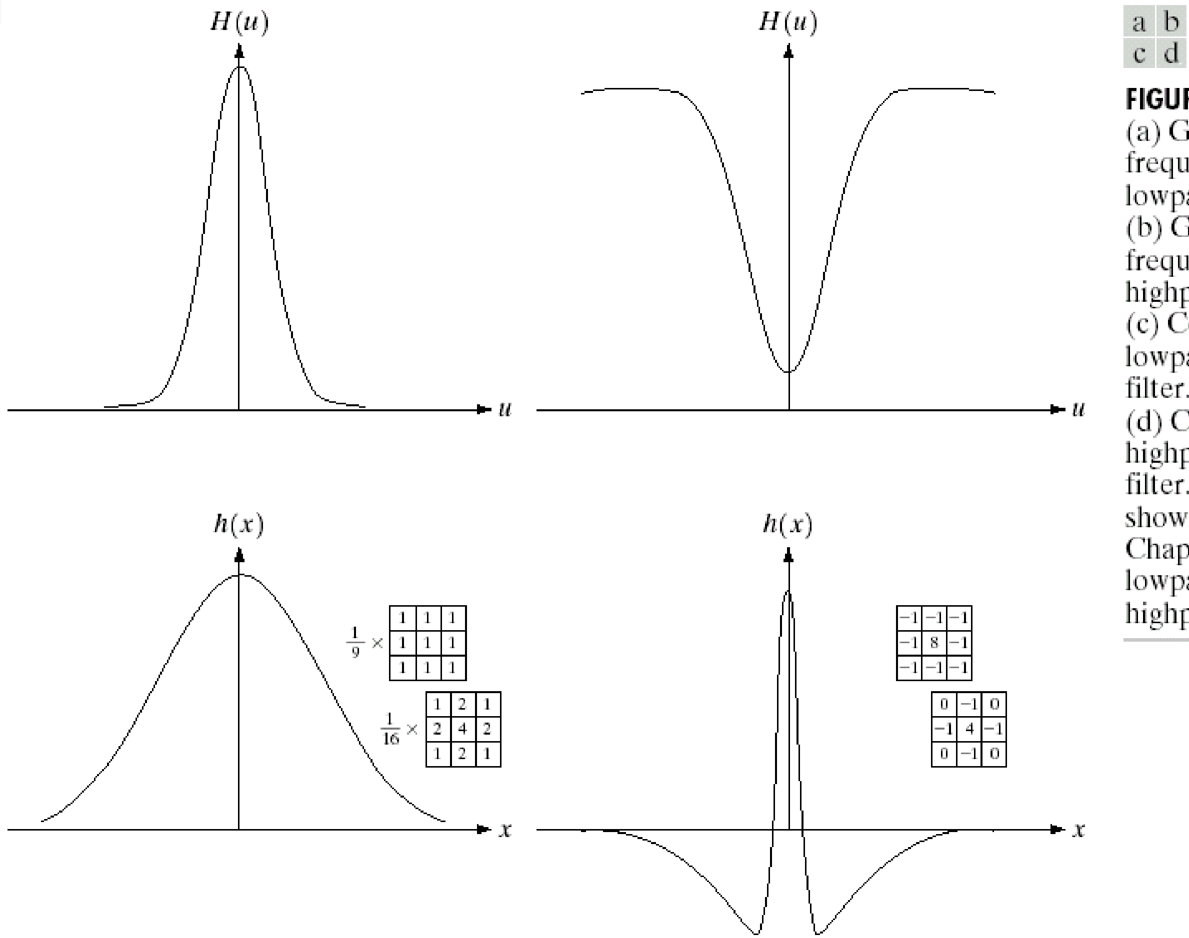


FIGURE 4.9 (a) Gaussian frequency domain lowpass filter. (b) Gaussian frequency domain highpass filter. (c) Corresponding lowpass spatial filter. (d) Corresponding highpass spatial filter. The masks shown are used in Chapter 3 for lowpass and highpass filtering.

4.3 Smoothing Frequency-Domain Filters

- 4.3.1 Ideal LPF (ILPF)

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

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

  - D_0: cutoff frequency

4.3 Smoothing Frequency-Domain Filters

- 4.3.1 Ideal LPF

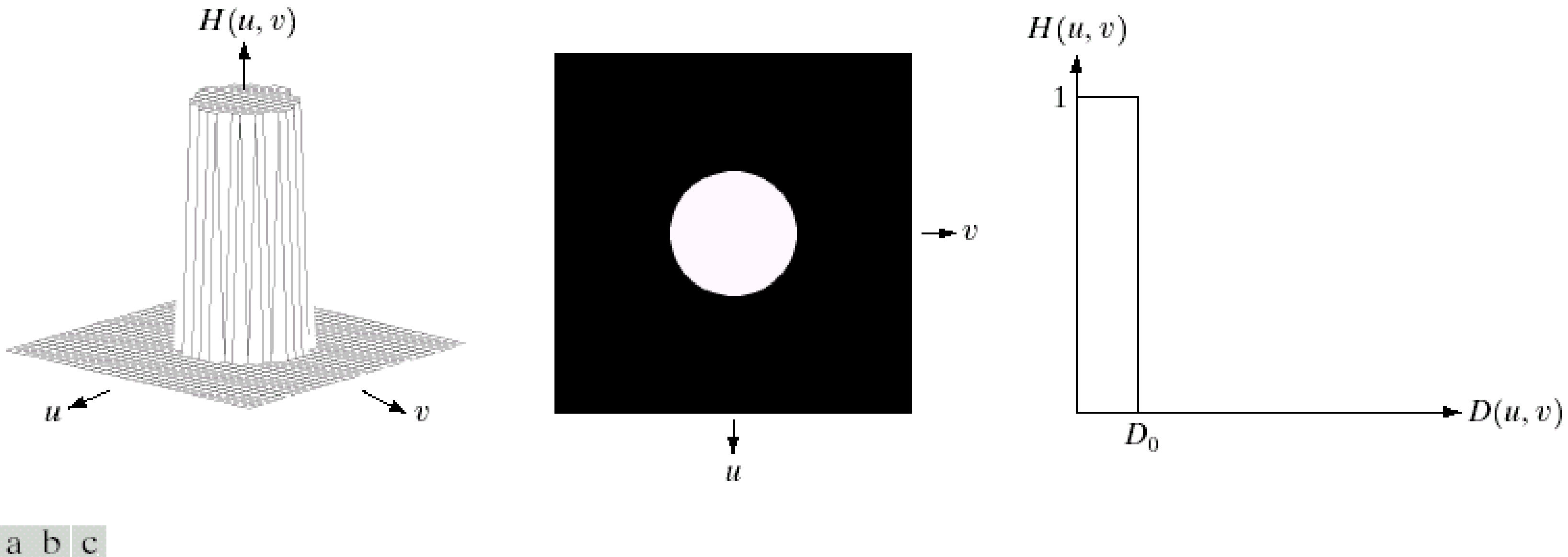
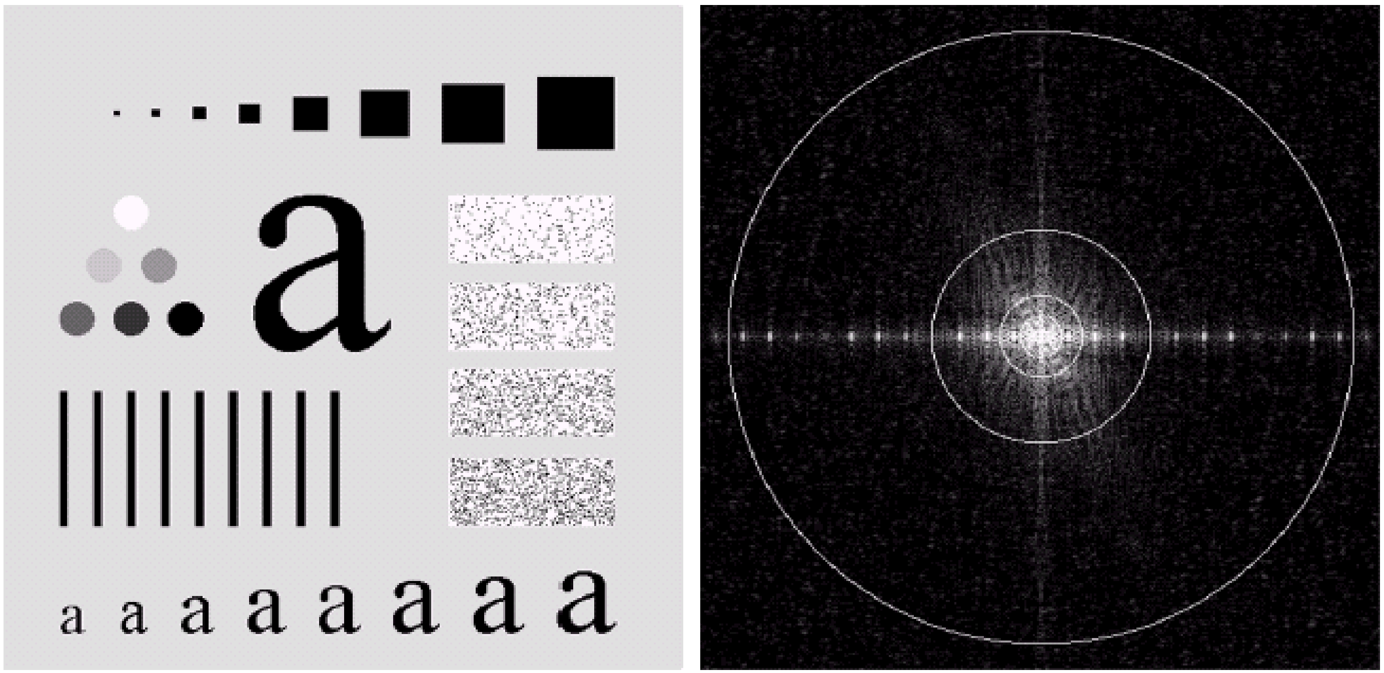


FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

4.3 Smoothing Frequency-Domain Filters

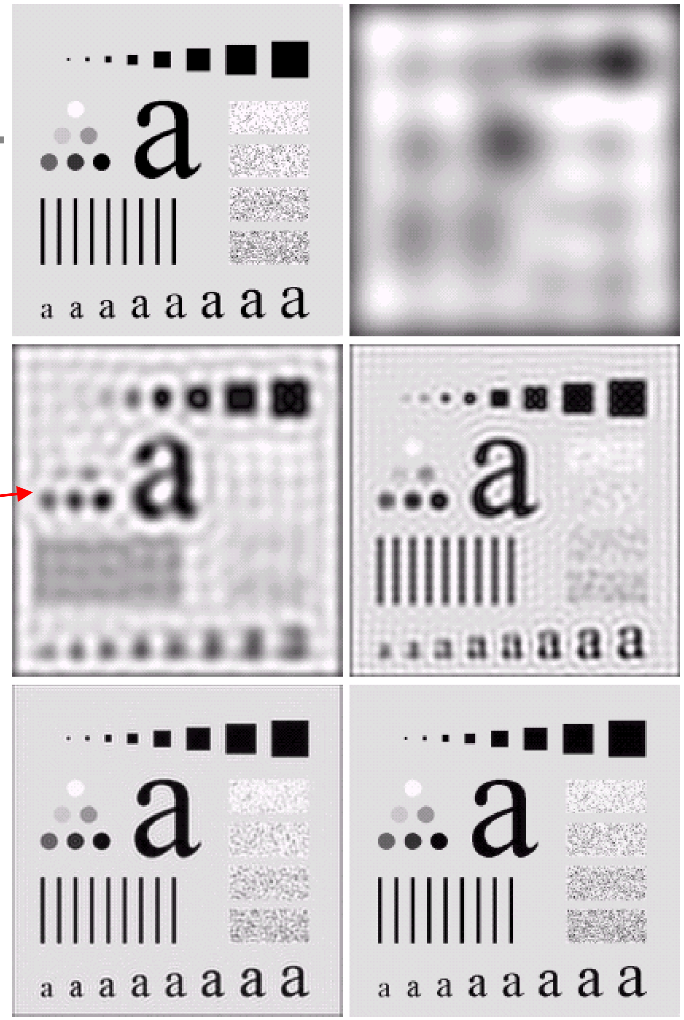


a b

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.

4.3 Smoothing Frequency-Domain Filters

ringing effect

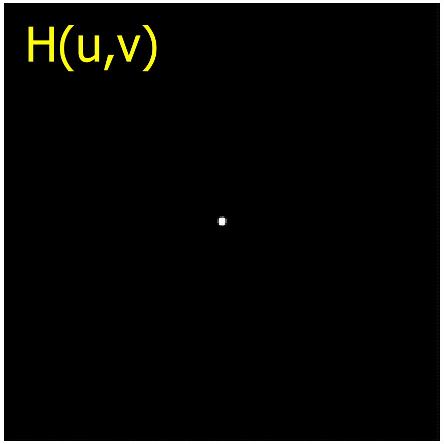


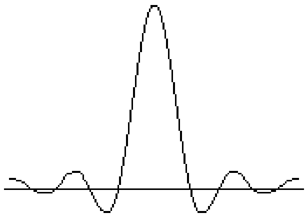
a b
c d
e f

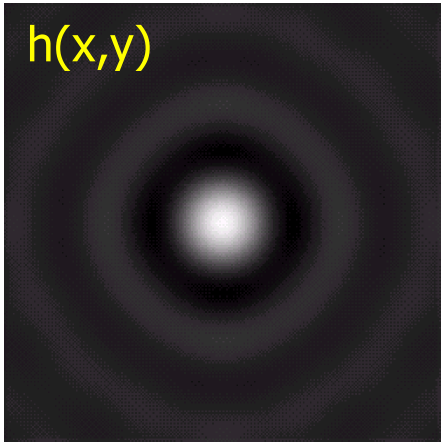
FIGURE 4.12 (a) Original image. (b)–(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). The power removed by these filters was 8, 5.4, 3.6, 2, and 0.5% of the total, respectively.

4.3 Smoothing Frequency-Domain Filters

- Why ringing effect?
  - Perspective from spatial filter function
    1. H(u,v) was multiplied by $(-1)^{u+v}$
    2. Perform 2D IDFT
    3. The real part of the inverse DFT was multiplied by $(-1)^{x+y}$ to get h(x,y)
  - h(x,y) has two distinctive properties:
    - **Dominant component** – responsible for blurring
    - **Concentric, circular component** – responsible for ringing characteristic







4.3 Smoothing Frequency-Domain Filters

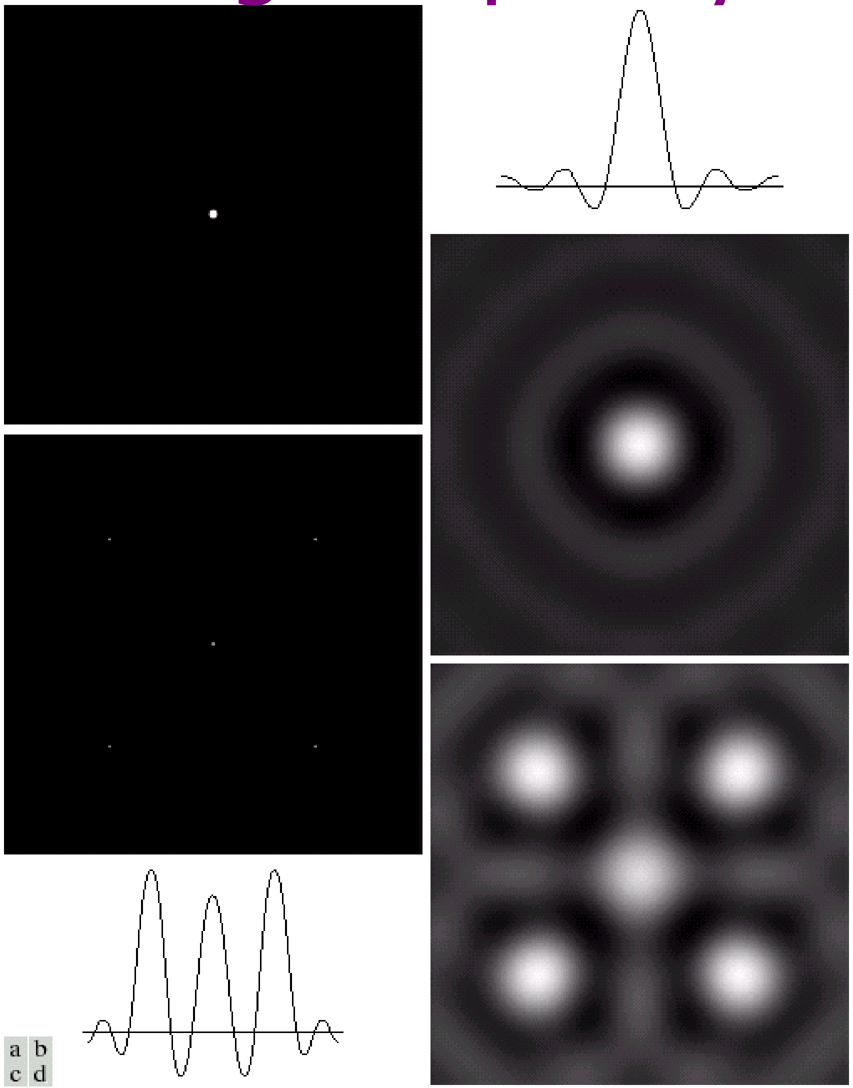


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

4.3.2 Butterworth LPF

- Butterworth LPF (BLPF)

$$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}} \quad n: \text{order of BLPF}$$

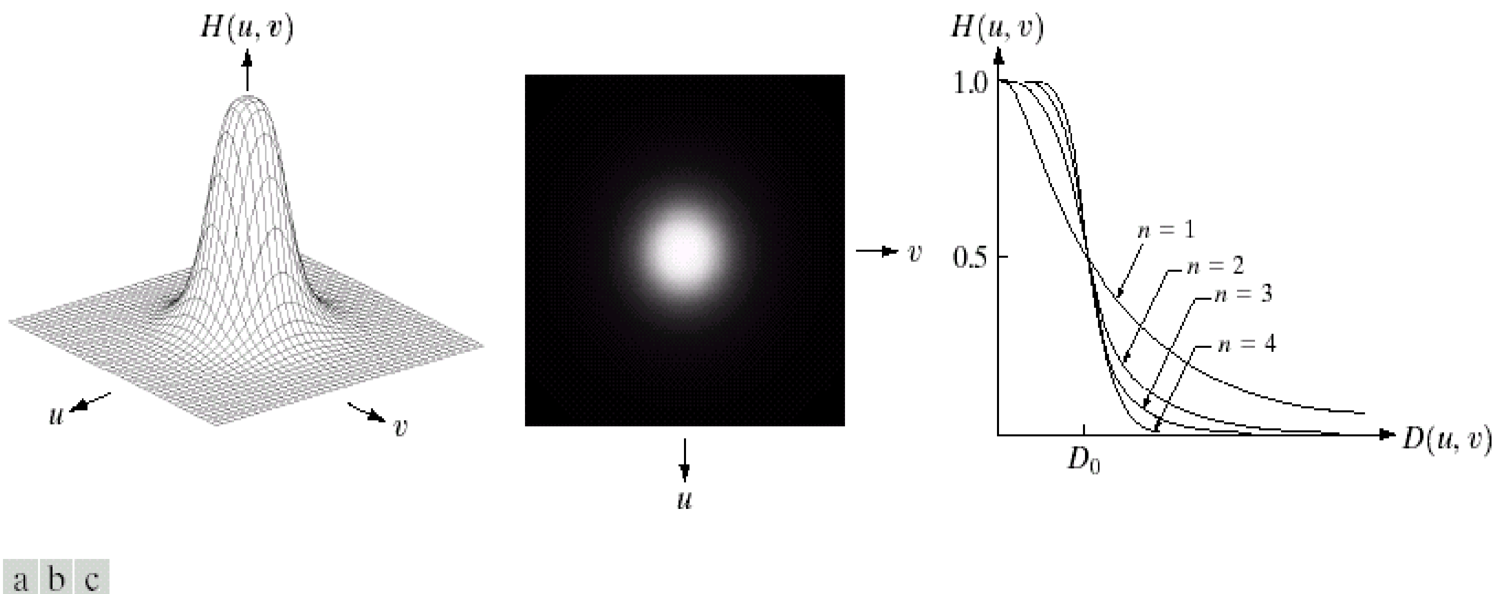
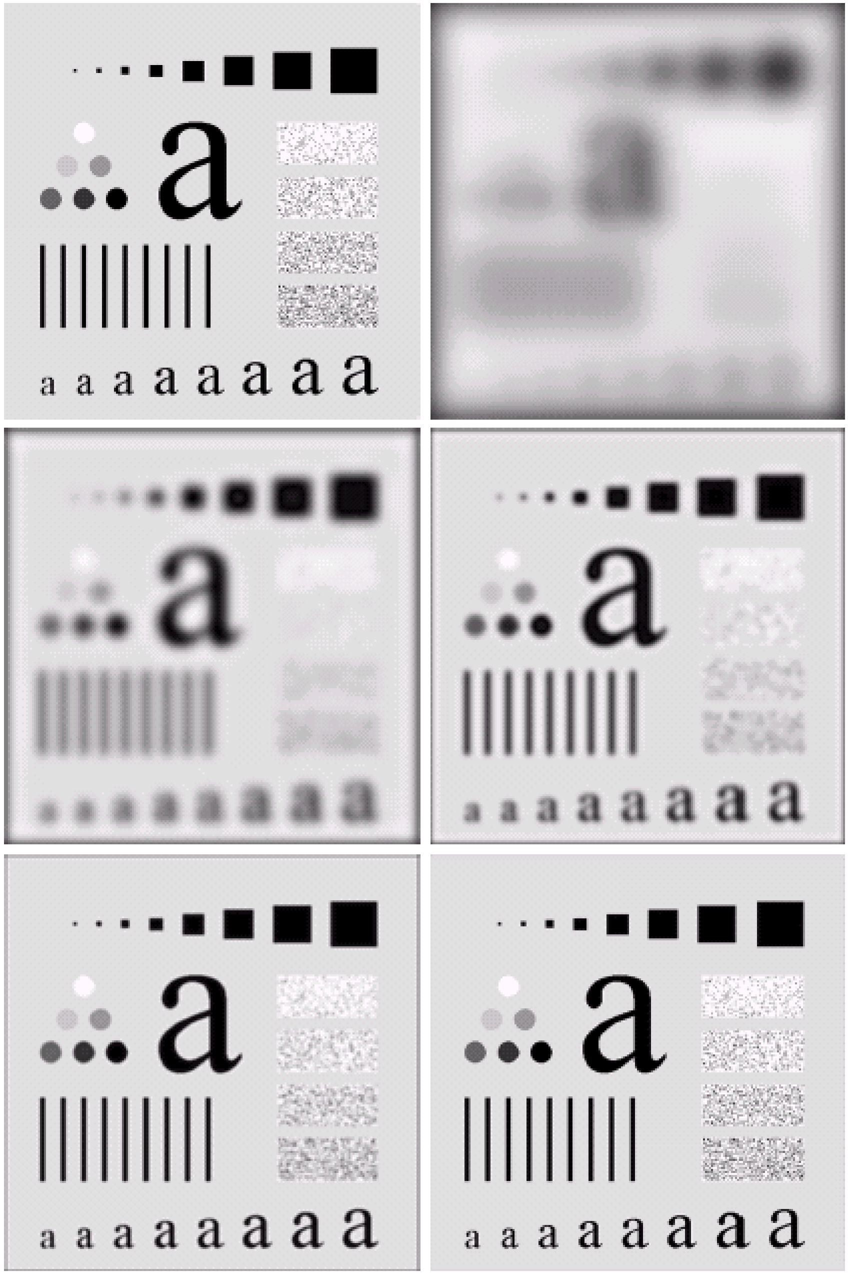


FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

4.3.2 Butterworth LPF



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.

4.3.2 Butterworth LPF

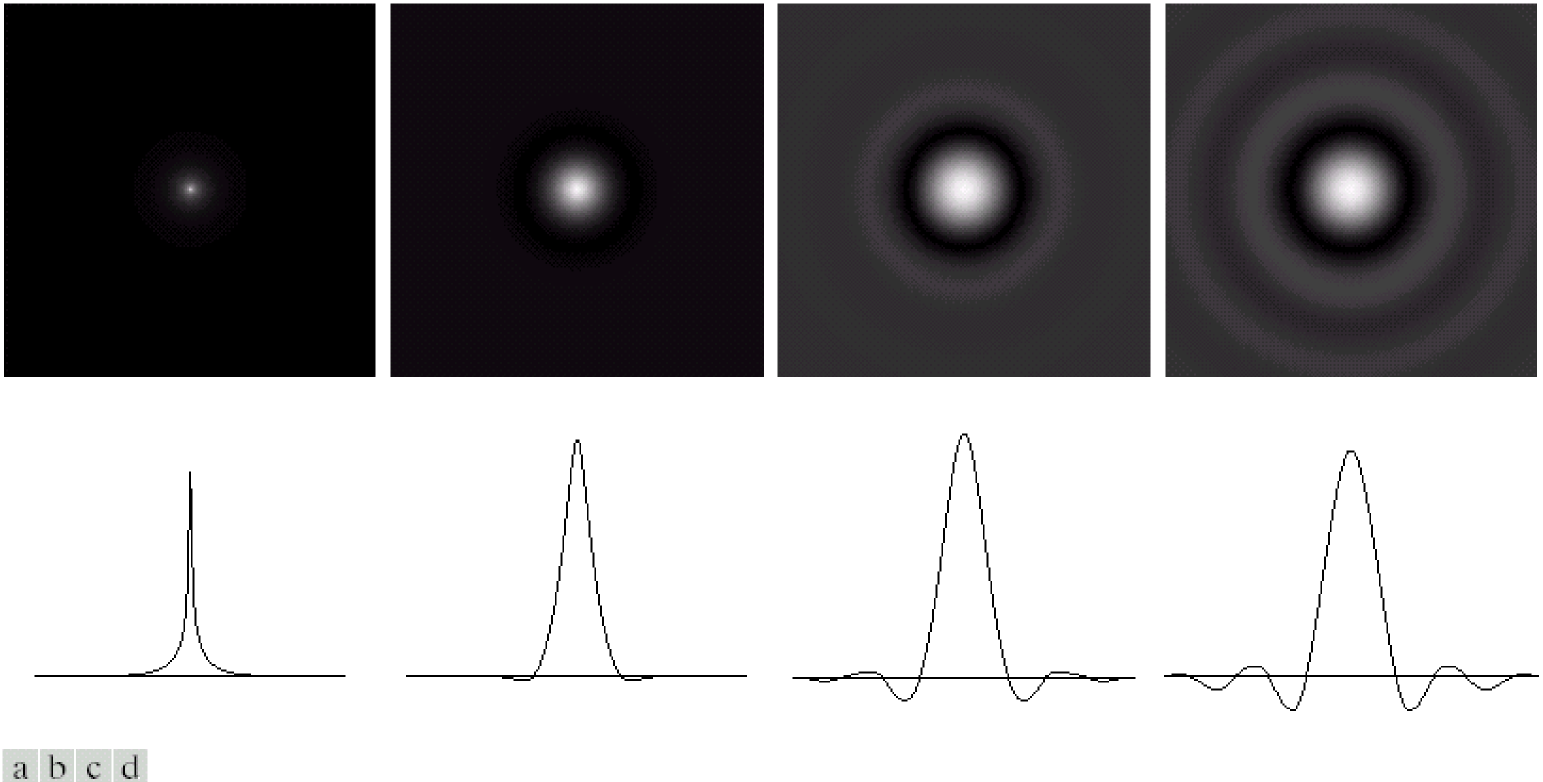


FIGURE 4.16 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding gray-level profiles through the center of the filters (all filters have a cutoff frequency of 5). Note that ringing increases as a function of filter order.

4.3.3 Gaussian LPF

- Gaussian LPF (GLPF)

$$H(u,v) = e^{-\frac{D^2(u,v)}{2\sigma^2}} \xrightarrow{\text{Let } \sigma = D_0} H(u,v) = e^{-\frac{D^2(u,v)}{2D_0^2}}$$

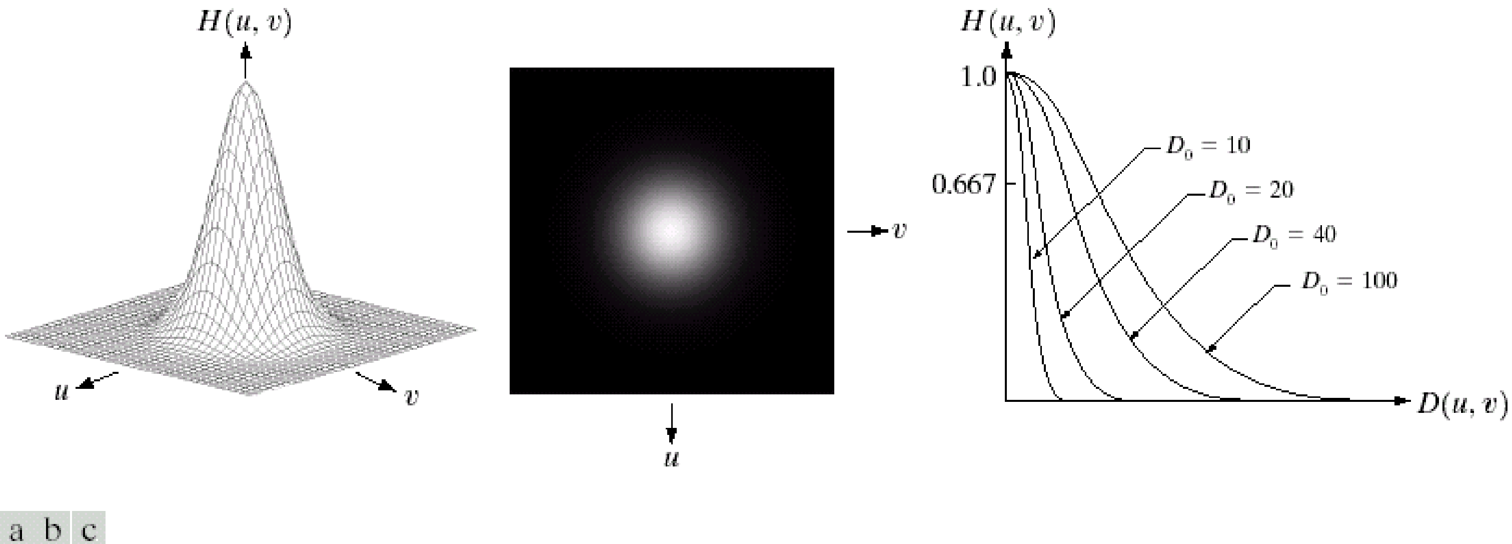
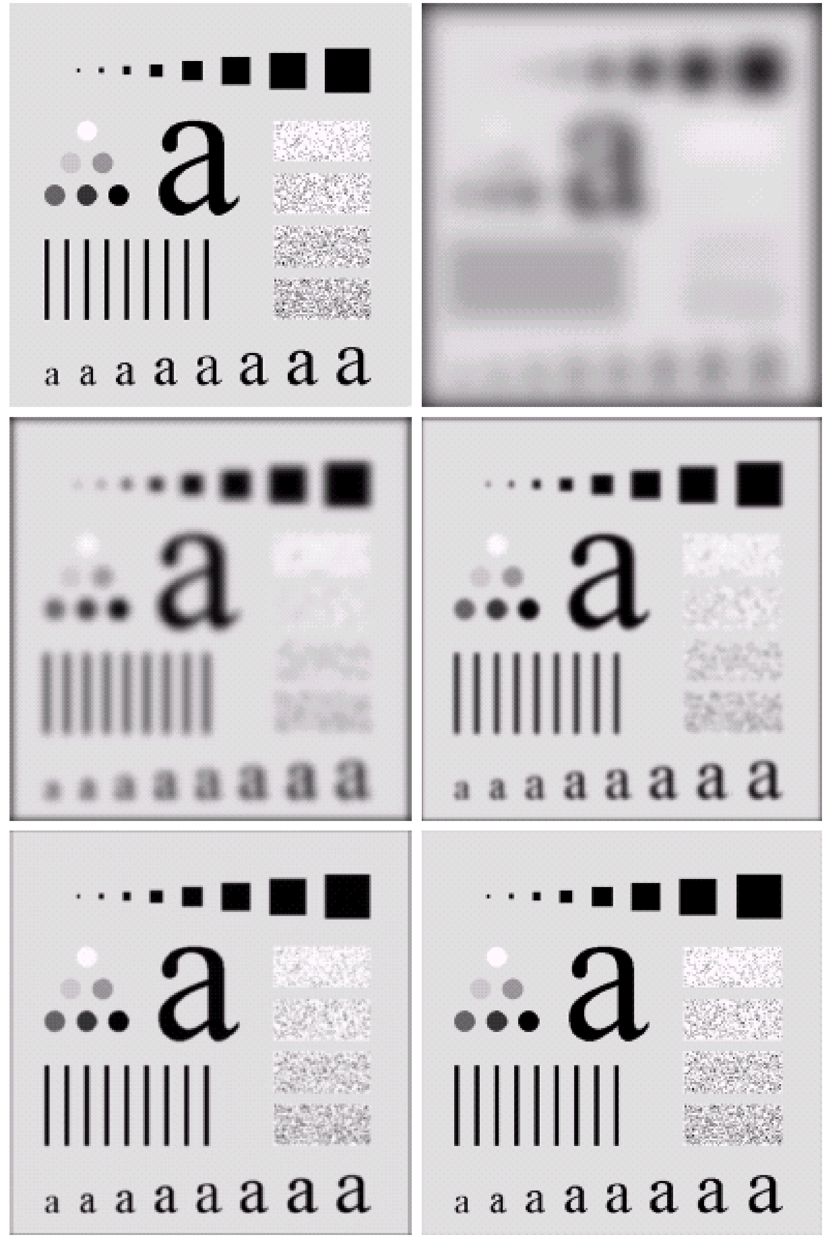


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

4.3.3 Gaussian LPF



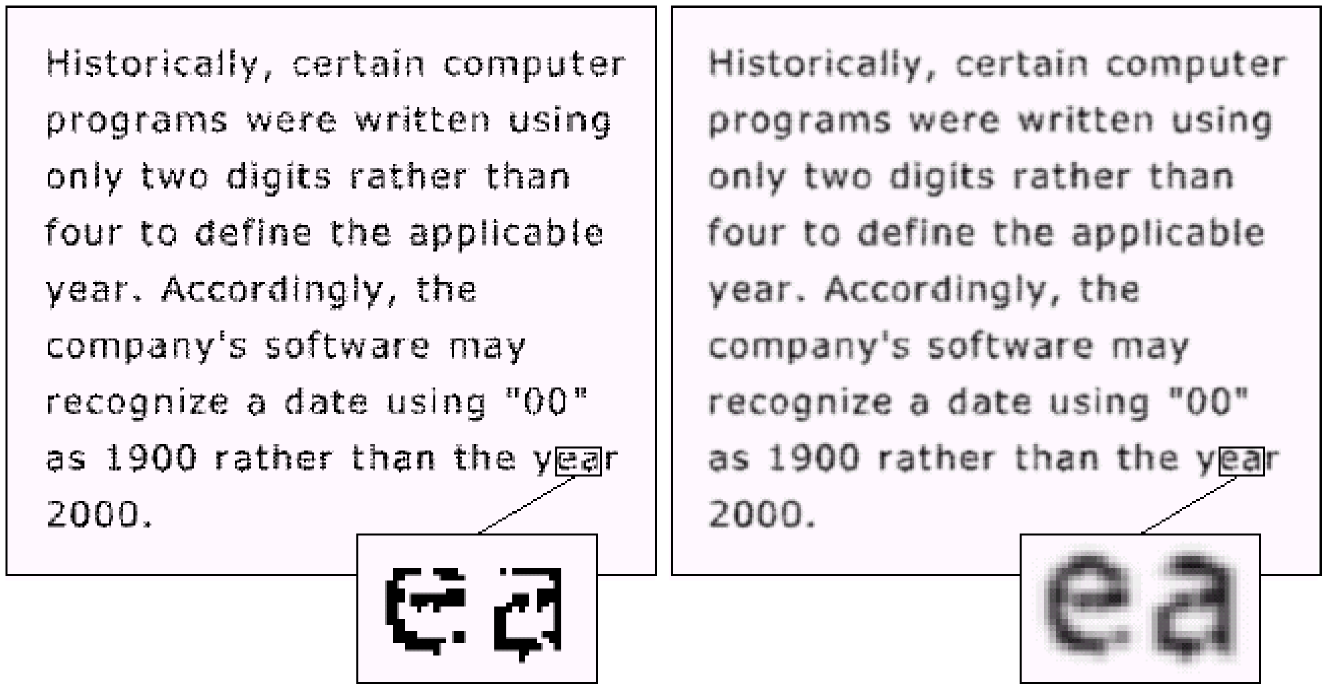
a b
c d
e f

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.

4.3.3 Gaussian LPF

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).



4.3.3 Gaussian LPF

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).

4.3.3 Gaussian LPF

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.)

4.4 Sharpening Frequency Domain Filters

- Highpass filter

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

  - Attenuate low-frequency components without disturbing high-frequency information in the frequency domain

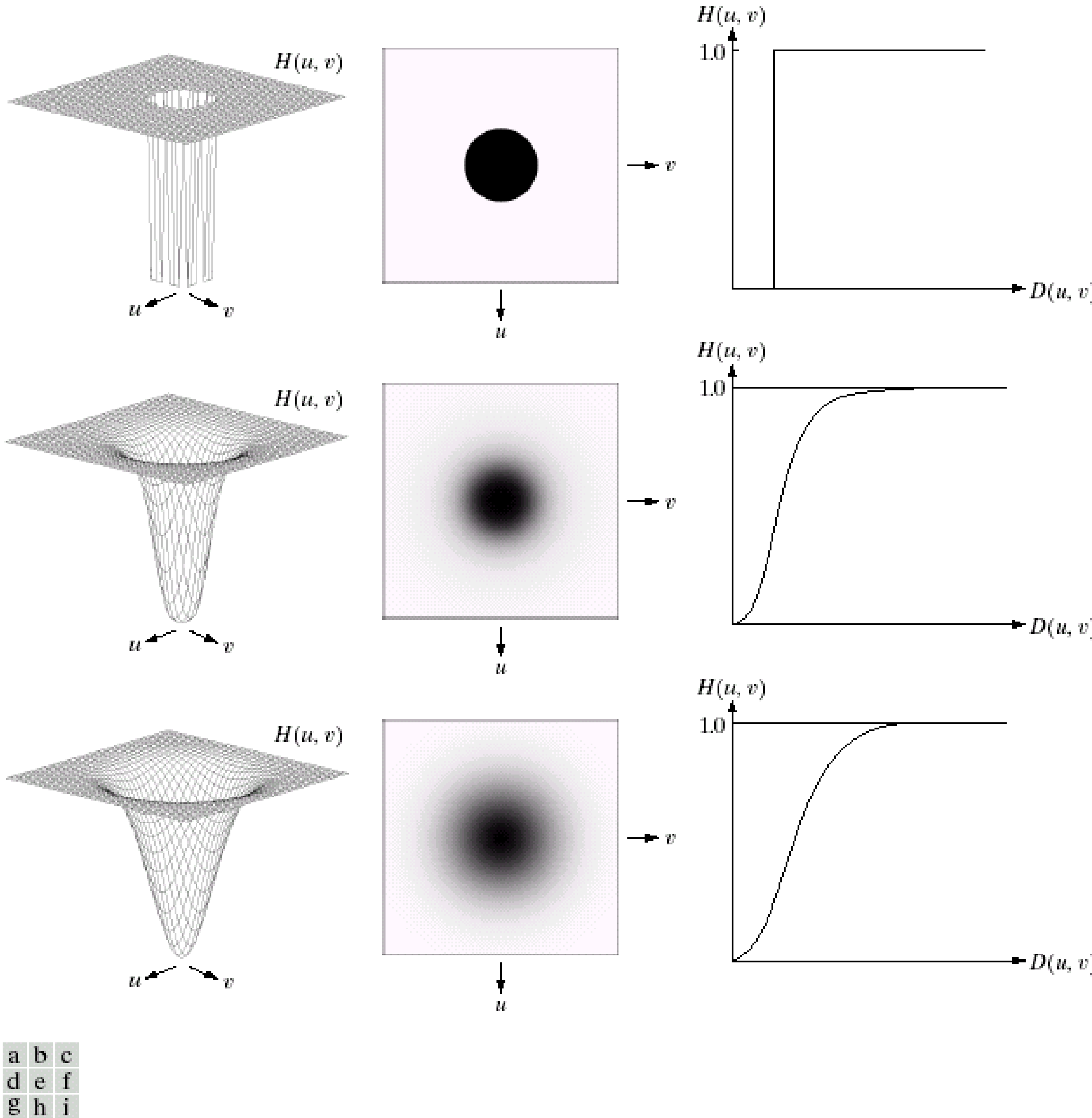


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

4.3 Smoothing Frequency-Domain Filters

- Spatial representation of a highpass filter
  1. $H_{hp}(u,v)$ was multiplied by $(-1)^{u+v}$
  2. Perform 2D IDFT
  3. The real part of the inverse DFT was multiplied by $(-1)^{x+y}$ to get $h_{hp}(x,y)$

4.4 Sharpening Frequency Domain Filters

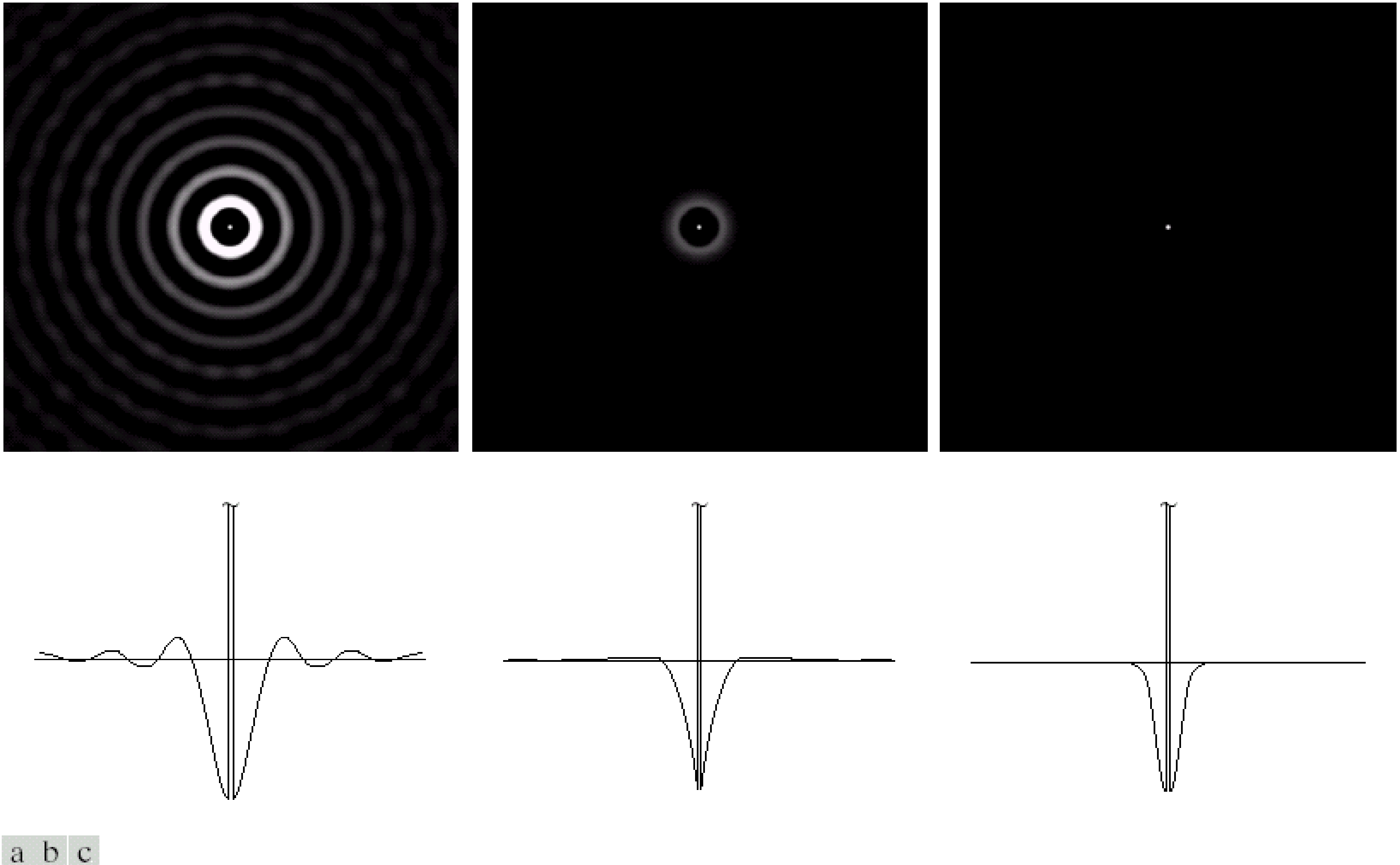
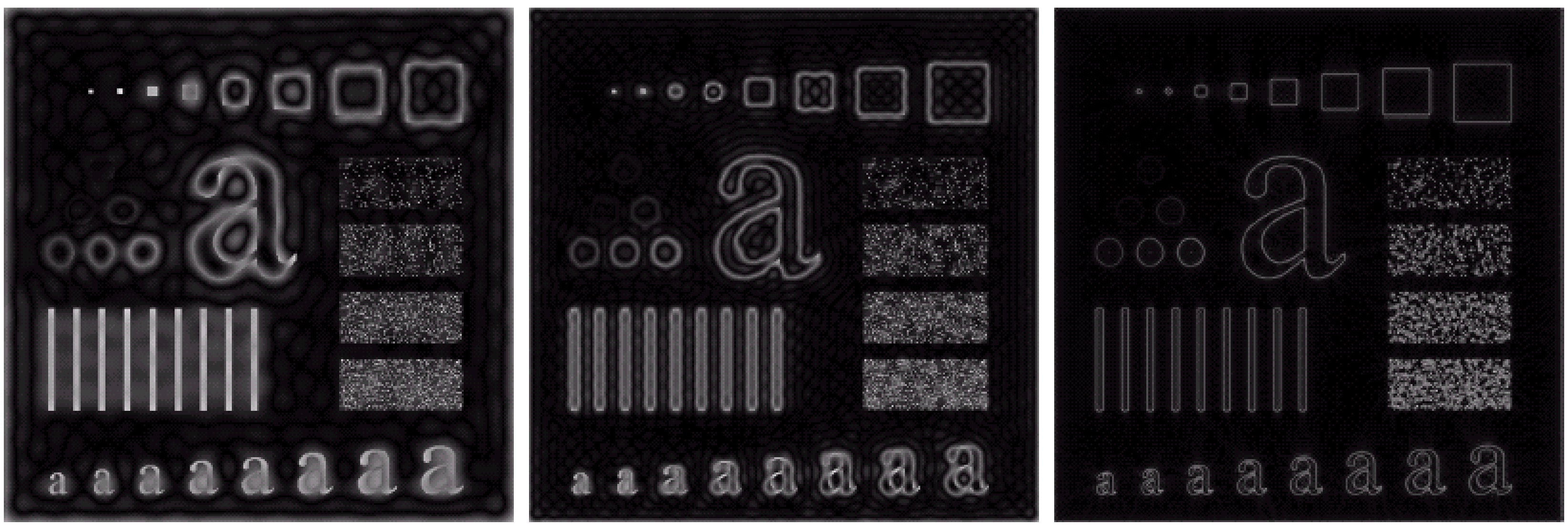


FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

4.4.1 Ideal Highpass filters

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

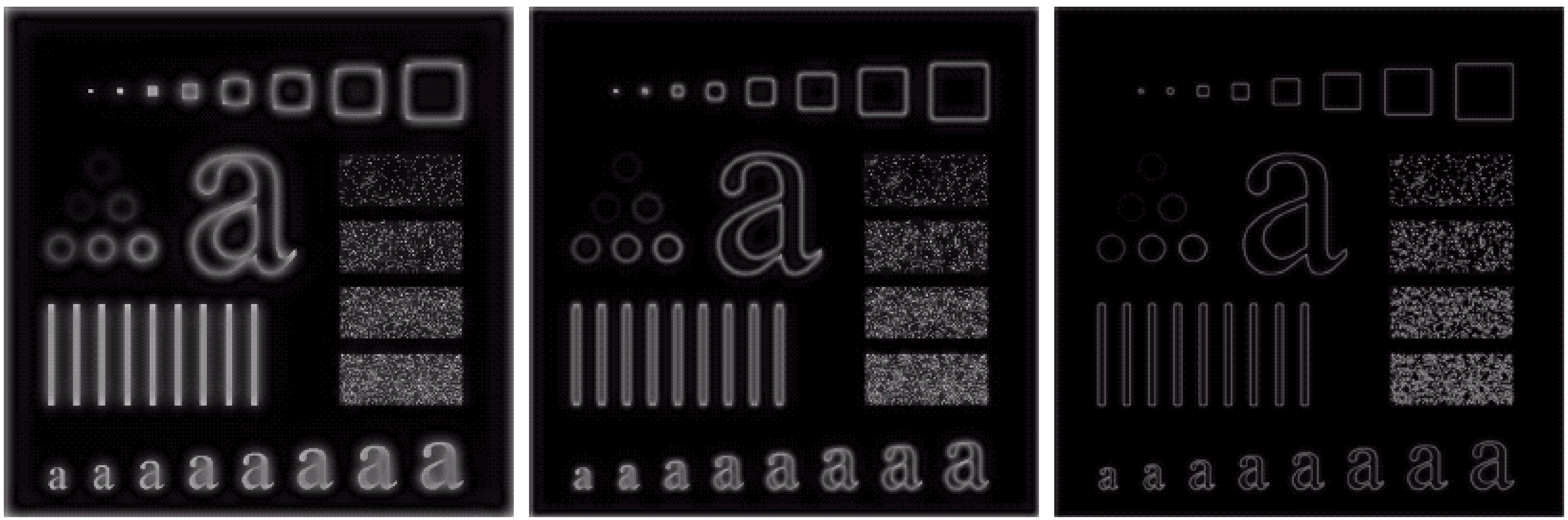


a b c

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).

4.4.2 Butterworth Highpass Filter

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

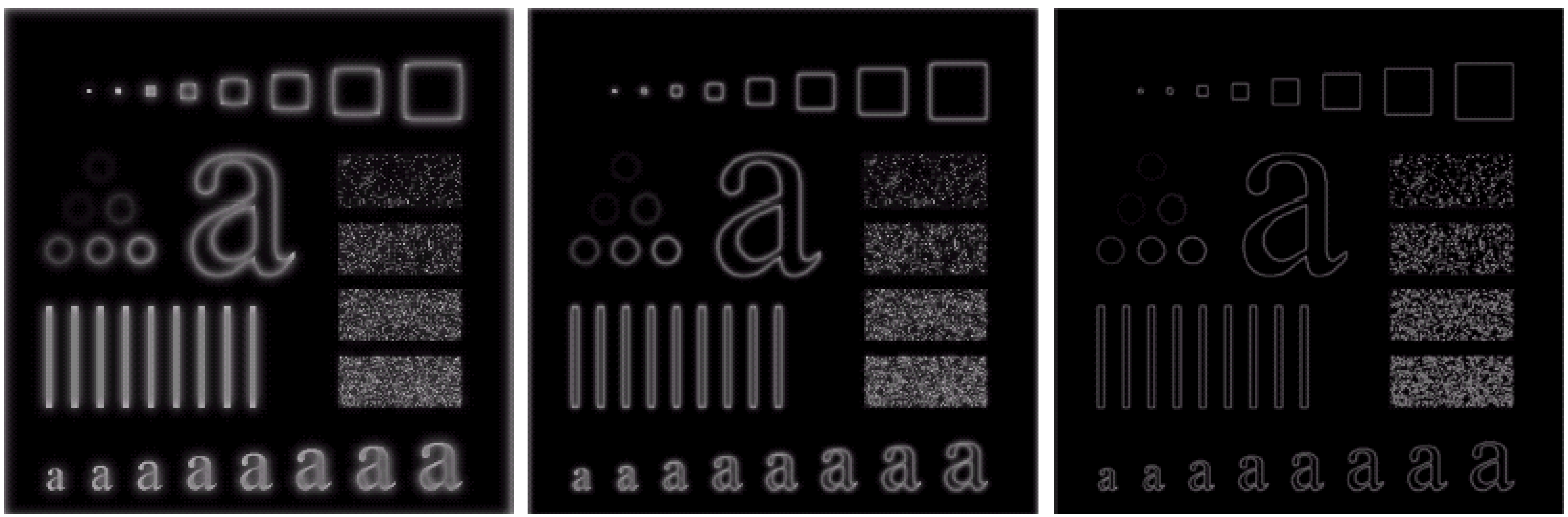


a b c

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.

4.4.3 Gaussian HPF

- Gaussian HPF: $H(u,v) = 1 - e^{-\frac{D^2(u,v)}{2D_0^2}}$



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.

4.4.4 The Laplacian in the Frequency Domain

- It can be shown that $\Im\left[\frac{d^n f(x)}{dx^n}\right] = (ju)^n F(u)$

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

Laplacian of *f*(*x*, *y*)

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

> The Laplacian can be implemented in the frequency domain by using the filter

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

4.4.4 The Laplacian in the Frequency Domain

- Derivation of spatial representation of a Laplacian filter
  1. Shifting the center of the filter function

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

  2. Perform 2D IDFT

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

  3. The real part of the inverse DFT was multiplied by $(-1)^{x+y}$ to get h (x,y)

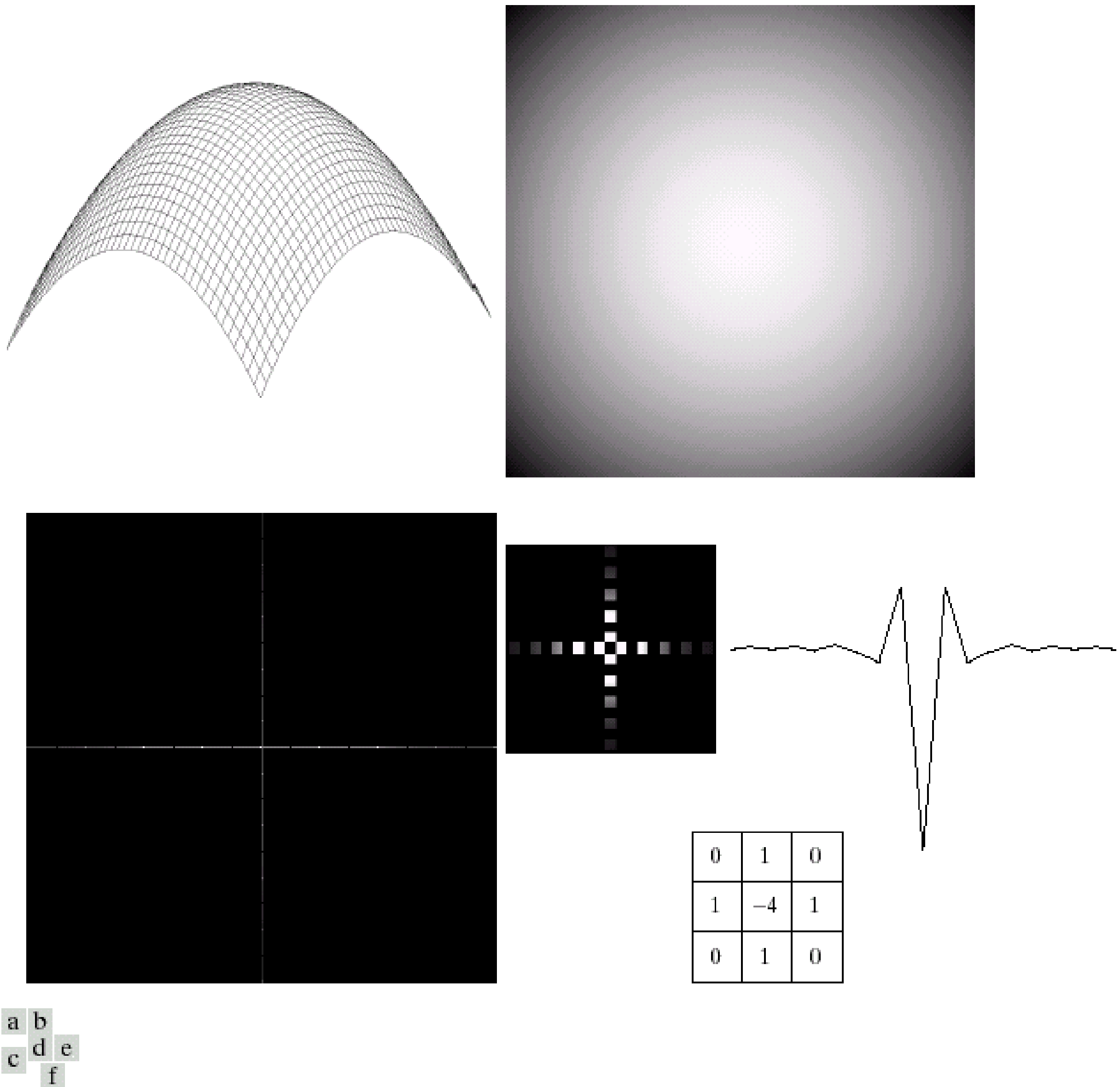


FIGURE 4.27 (a) 3-D plot of Laplacian in the frequency domain. (b) Image representation of (a). (c) Laplacian in the spatial domain obtained from the inverse DFT of (b). (d) Zoomed section of the origin of (c). (e) Gray-level profile through the center of (d). (f) Laplacian mask used in Section 3.7.

4.4.4 The Laplacian in the Frequency Domain

- Enhancement using Laplacian

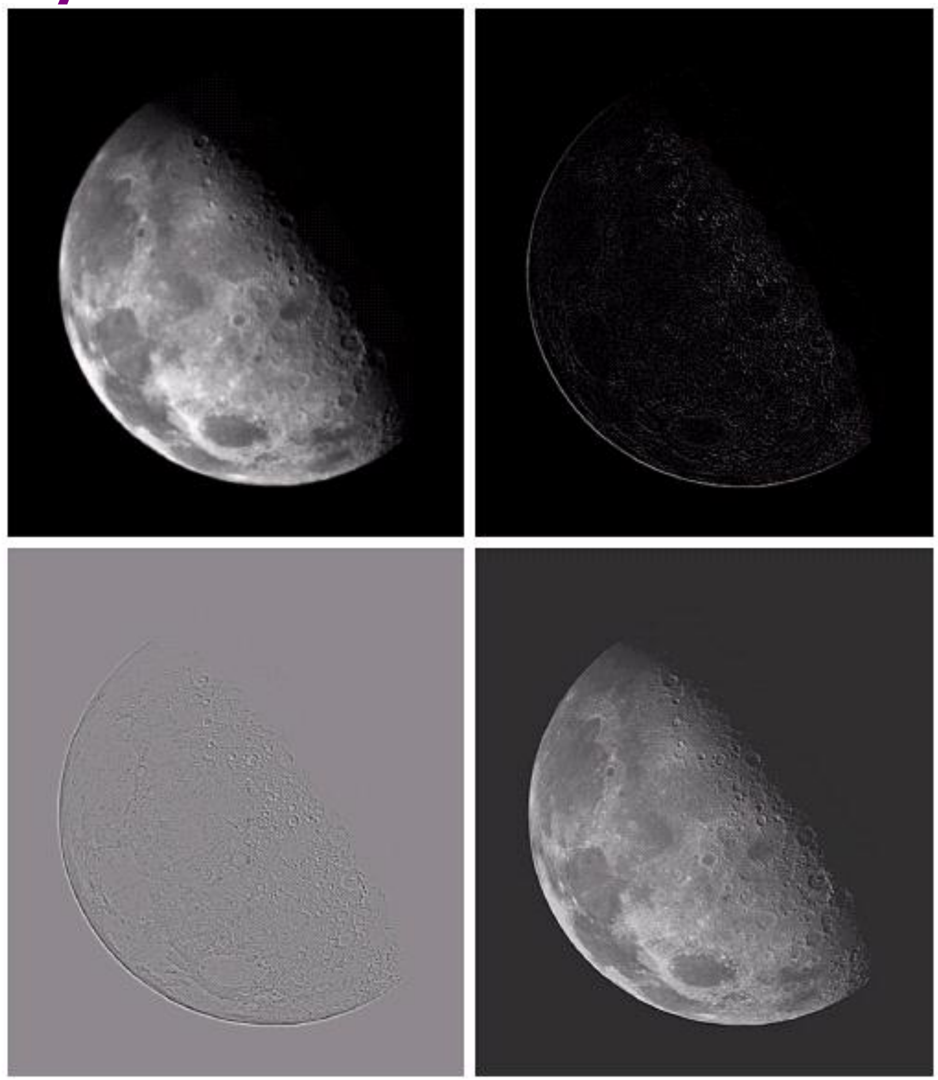
$$g(x, y) = f(x, y) - \nabla^2 f(x, y)$$

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

4.4.4 The Laplacian in the Frequency Domain

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.)



4.4.5 Unsharp Masking, High Boost Filtering

- Highpass filter: $f_{hp}(x,y) = f(x,y) - f_{lp}(x,y)$
- High-boost filter: $f_{hb}(x,y) = Af(x,y) - f_{lp}(x,y)$, $A \geq 1$
  - $f_{hb}(x,y) = (A-1)f(x,y) + f(x,y) - f_{lp}(x,y)$
    $= (A-1)f(x,y) + f_{hp}(x,y)$
- $F_{hp}(u,v) = F(u,v) - F_{lp}(u,v)$ $(F_{lp}(u,v) = H_{lp}(u,v)F(u,v))$
  $= F(u,v) - H_{lp}(u,v)F(u,v)$
- $H_{hp}(u,v) = 1 - H_{lp}(u,v)$
- Similarly, $H_{hb}(u,v) = (A-1) + H_{hp}(u,v)$

4.4.5 Unsharp Masking, High Boost Filtering

a b
c d

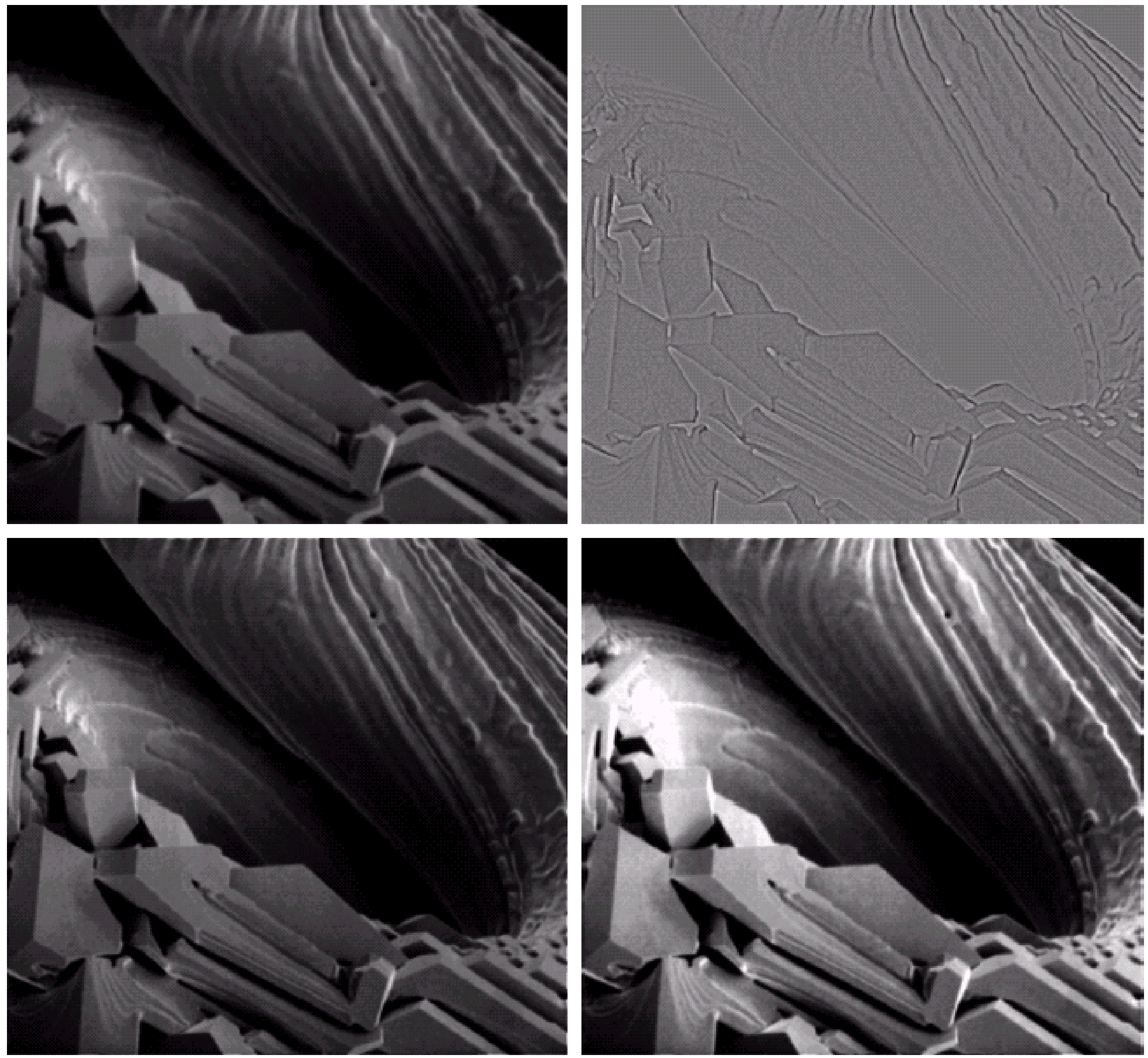


FIGURE 4.29 Same as Fig. 3.43, but using frequency domain filtering. (a) Input image. (b) Laplacian of (a). (c) Image obtained using Eq. (4.4-17) with $A = 2$. (d) Same as (c), but with $A = 2.7$. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon,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

4.4.5 Unsharp Masking, High Boost Filtering

- **High-frequency emphasis filter**:

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

$$a \geq 0 \ (0.25 \leq a \leq 0.5),\ b > a \ (1.5 \leq b \leq 2.0)$$

4.4.5 Unsharp Masking, High Boost Filtering

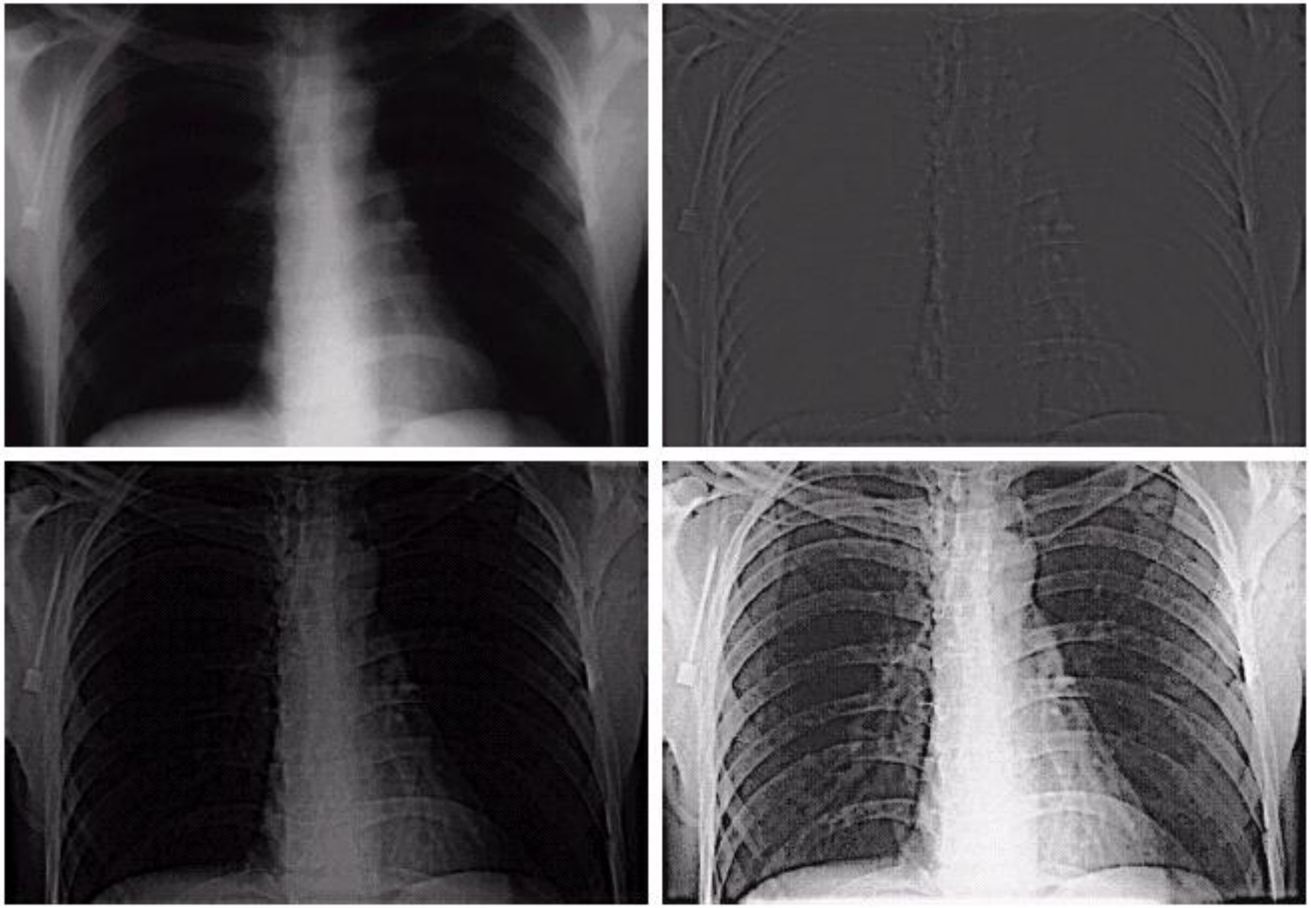


FIGURE 4.30 (a) A chest X-ray image. (b) Result of Butterworth highpass filtering. (c) Result of high-frequency emphasis filtering. (d) Result of performing histogram equalization on (c). (Original image courtesy Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

4.5 Homomorphic Filtering

- Homophorphic filtering is designed by using the illumination-reflectance model:

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

- $z(x,y) = \ln f(x,y) = \ln i(x,y) + \ln r(x,y)$
- $\Im[z(x,y)] = \Im[\ln i(x,y) + \ln r(x,y)]$
- $Z(u,v) = F_i(u,v) + F_r(u,v)$
- Filtering process

  $S(u,v) = H(u,v)\, Z(u,v) = H(u,v)\, F_i(u,v) + H(u,v)\, F_r(u,v)$

4.5 Homomorphic Filtering

- Inverse Fourier transform

$$s(x,y) = \mathfrak{I}^{-1}\{S(u,v)\}$$
$$= \mathfrak{I}^{-1}\{H(u,v)\, F_i(u,v)\} + \mathfrak{I}^{-1}\{H(u,v)\, F_i(u,v)\}$$

- Let $i'(x,y) = \mathfrak{I}^{-1}\{H(u,v)\, F_i(u,v)\}$

  $r'(x,y) = \mathfrak{I}^{-1}\{H(u,v)\, F_r(u,v)\}$
- $s(x,y) = i'(x,y) + r'(x,y)$
- $g(x,y) = e^{s(x,y)} = e^{i'(x,y)} + e^{r'(x,y)} = i_0(x,y) + r_0(x,y)$

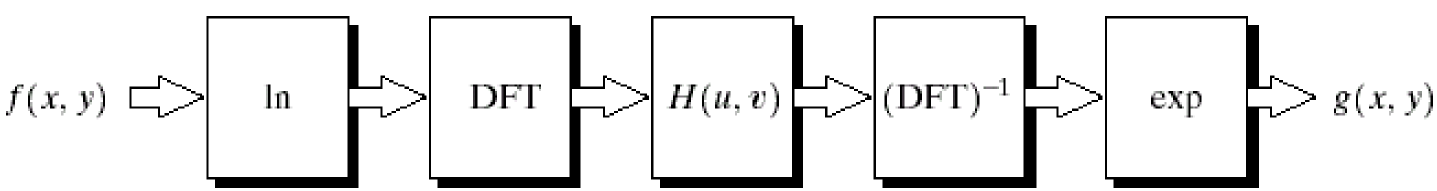


FIGURE 4.31 Homomorphic filtering approach for image enhancement.

4.5 Homomorphic Filtering

- Since $Z(u,v) = F_i(u,v) + F_r(u,v)$, the homomorphic filter function $H(u,v)$ can operate on these components separately
  - Illumination component – slow spatial variations
  - Reflectance component – vary abruptly, particularly at the junctions of dissimilar objects
  - Associating the low frequencies of the Fourier transform of the logarithm of the image with illumination and the high frequencies with reflectance

4.5 Homomorphic Filtering

- Simultaneous dynamic range compression and contrast enhancement
  - If $\gamma_L < 1$: decrease the contribution made by the low frequencies (illumination)
  - If $\gamma_H > 1$: amplify the contribution made by the high frequencies (reflectance)

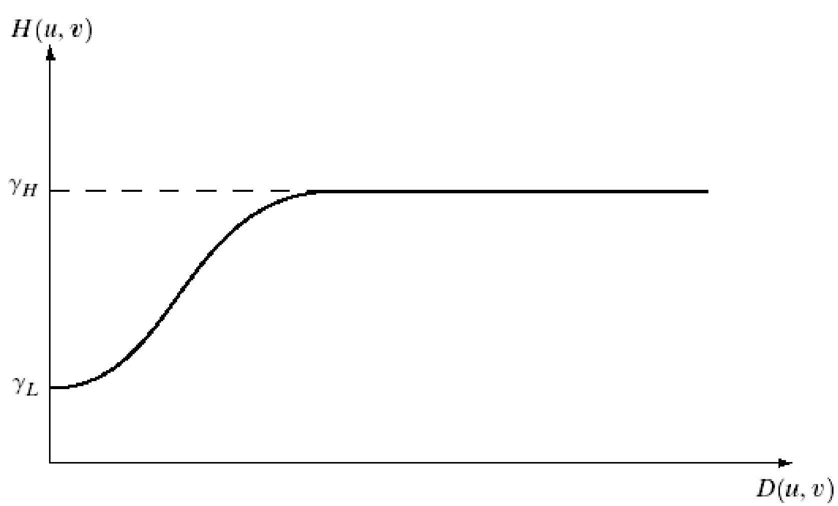


FIGURE 4.32 Cross section of a circularly symmetric filter function. $D(u, v)$ is the distance from the origin of the centered transform.

4.5 Homomorphic Filtering

- Approximating homomorphic filter using modified Gaussian highpass filter:

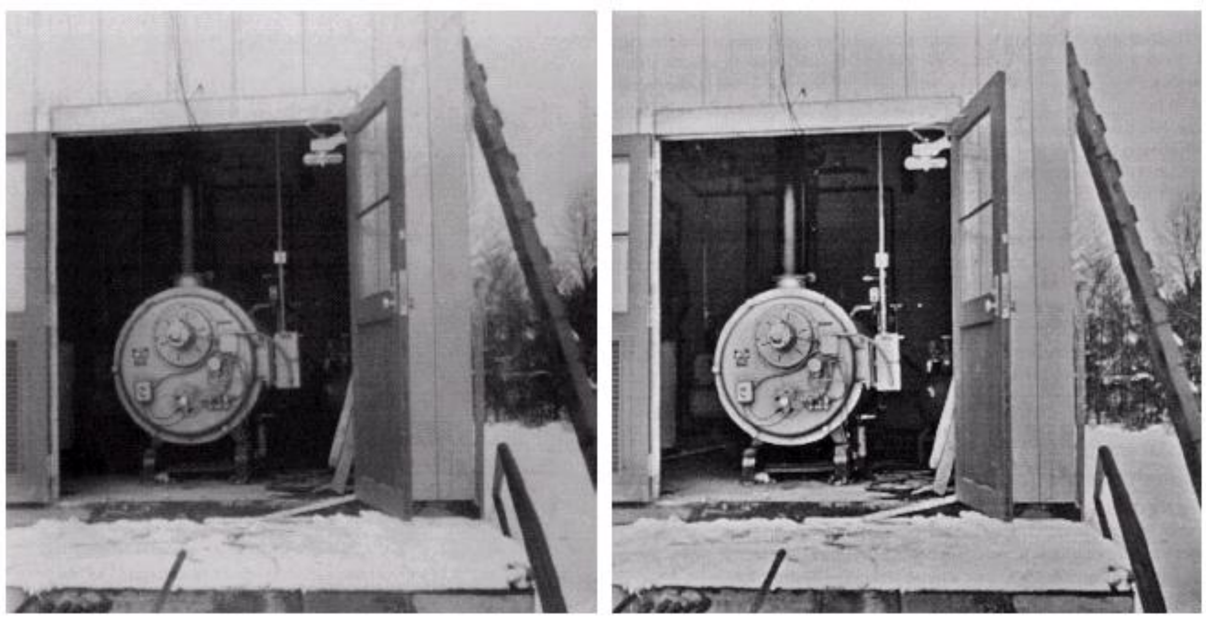
$$H(u,v) = (\gamma_H - \gamma_L)\left(1 - e^{-c(D^2(u,v)/D_0^2)}\right) + \gamma_L$$

a b

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

$\gamma_L = 0.5$

$\gamma_H = 2.0$



4.6 Implementation

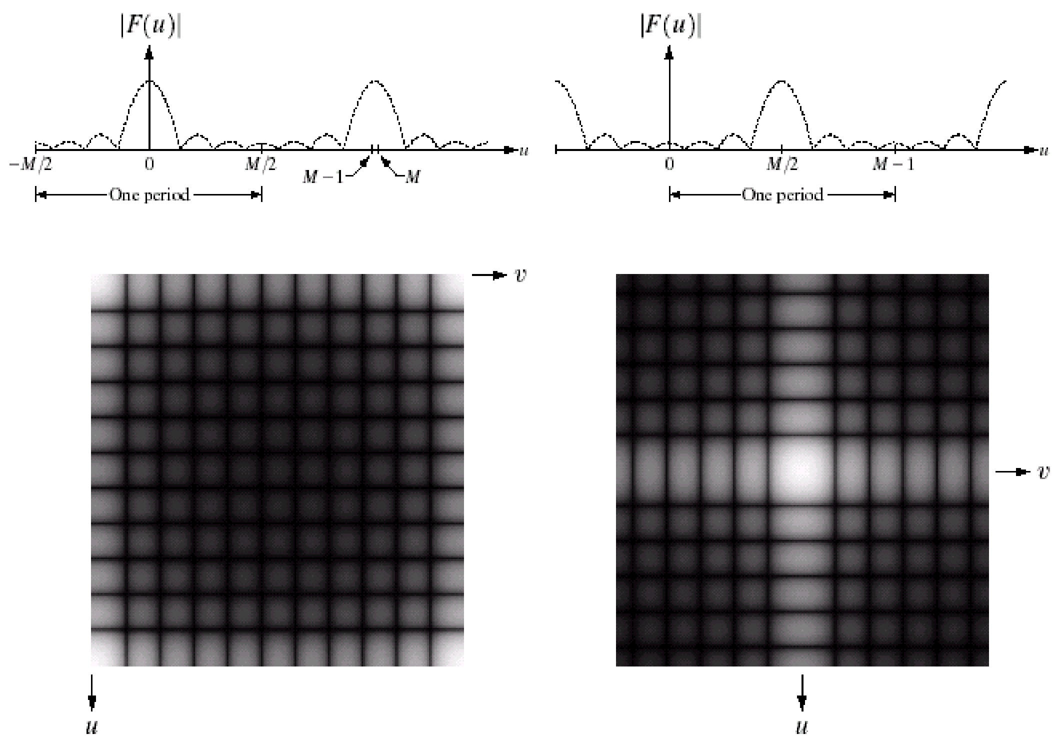


FIGURE 4.34
(a) Fourier spectrum showing back-to-back half periods in the interval $[0, M-1]$.
(b) Shifted spectrum showing a full period in the same interval.
(c) Fourier spectrum of an image, showing the same back-to-back properties as (a), but in two dimensions.
(d) Centered Fourier spectrum.

4.6 Implementation

- Prosperities of Fourier transform

  1. Shifting

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

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

  2. Scaling

     $f(ax,by) \Longleftrightarrow 1/|ab|F(u/a,v/b)$

  3. Rotation

     $f(r,\theta+\theta_0) \Longleftrightarrow F(\omega,\phi+\theta_0)$

  4. Periodicity

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

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

4.6 Implementation

5. Separability

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

$$= \frac{1}{M}\sum_{x=0}^{M-1}[\frac{1}{N}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi\frac{vy}{N}}]e^{-j2\pi\frac{ux}{M}}$$

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

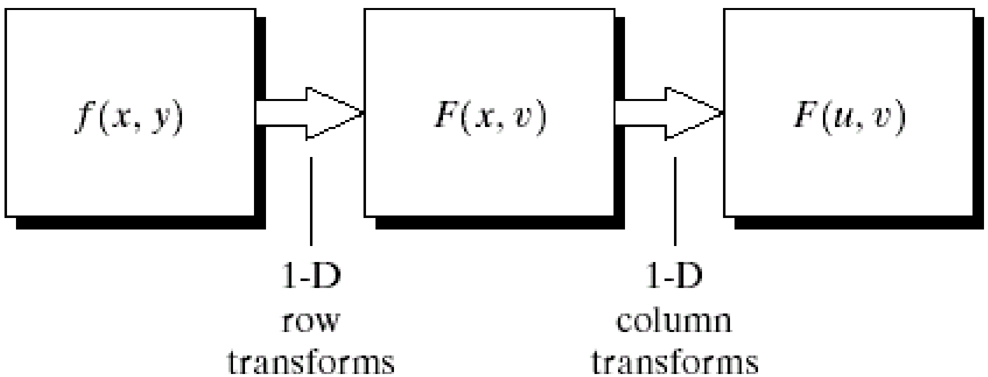


FIGURE 4.35 Computation of the 2-D Fourier transform as a series of 1-D transforms.

4.6.3 Padding

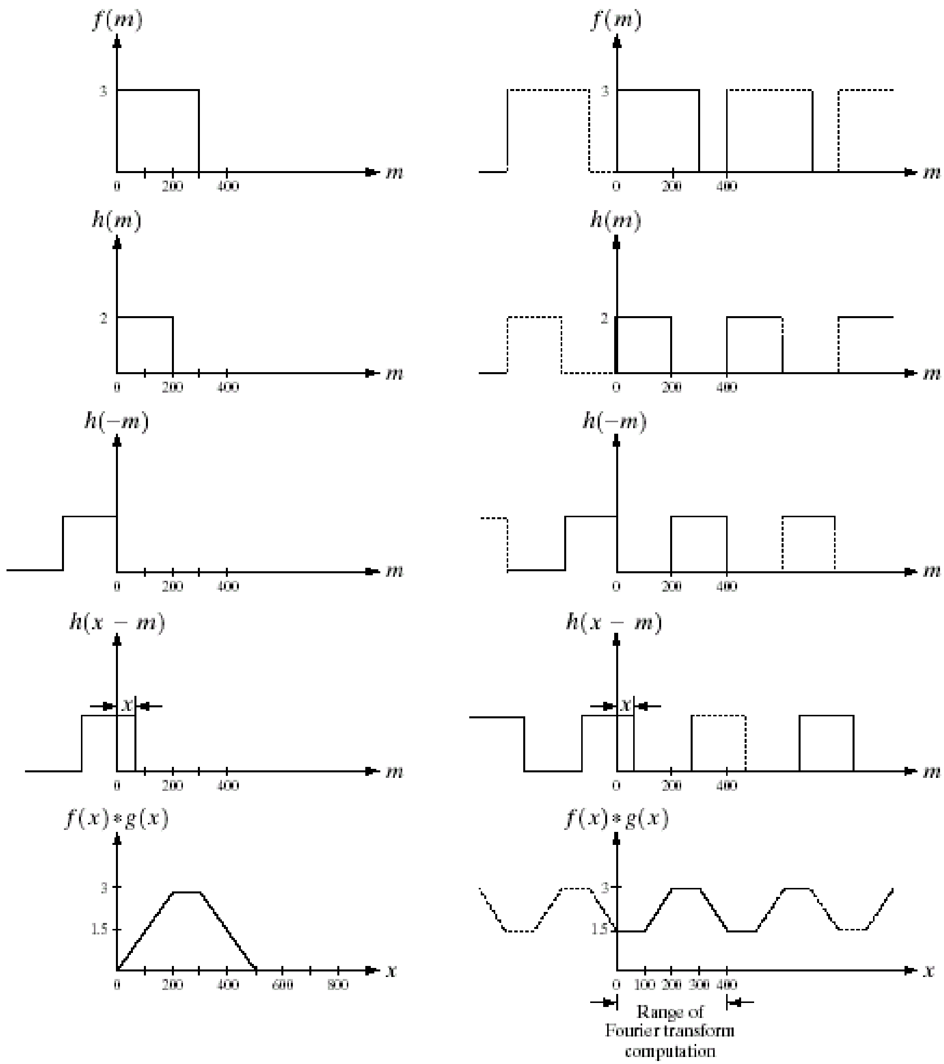
- The relationship between the linear convolution and circular convolution.
- Circular convolution of M-points 1-D signal

  $\Longleftrightarrow$ M-point multiplication of DFT coefficients.
- Linear convolution

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

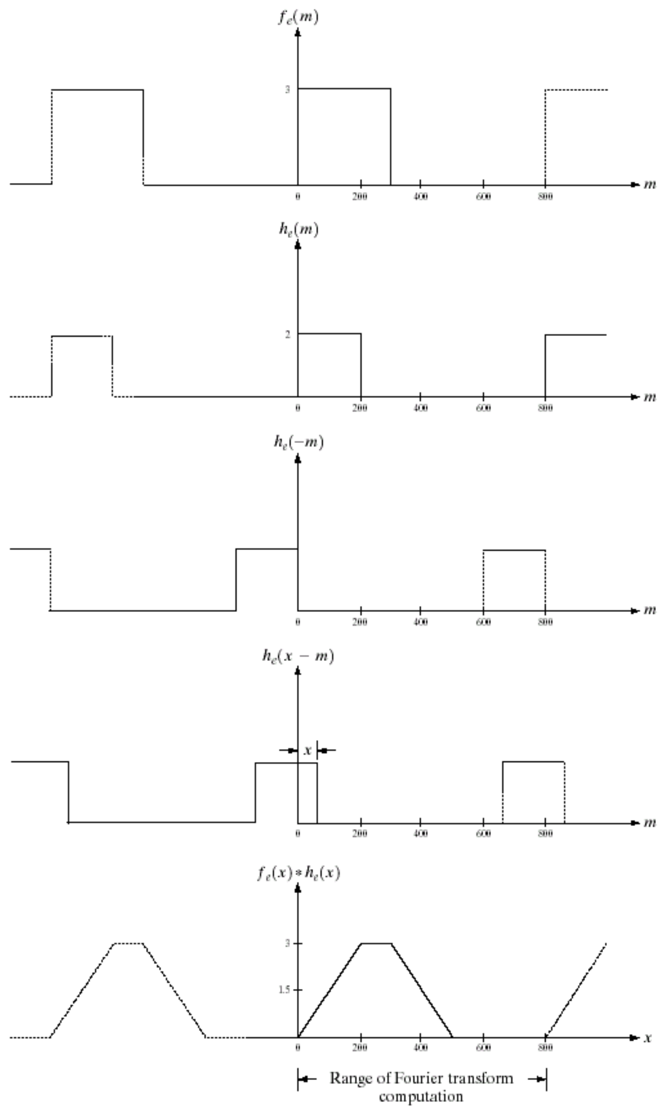
a	f
b	g
c	h
d	i
e	j

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.



a
b
c
d
e

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

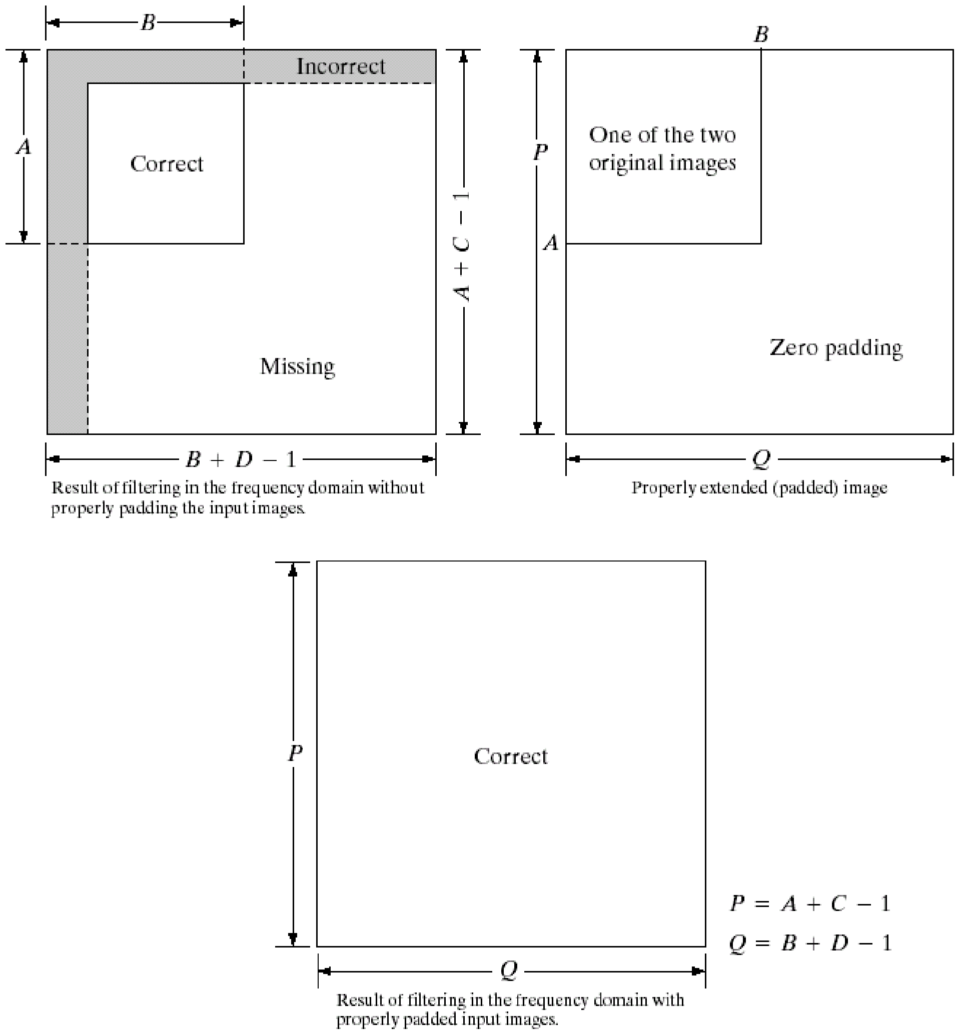


- To avoid the wraparound error, the padding process is applied.

$$f_e(x) = \begin{cases} f(x) & 0 \le x \le A-1 \\ 0 & \mathrm{A} \le x \le p \end{cases}$$

$$g_e(x) = \begin{cases} f(x) & 0 \le x \le B-1 \\ 0 & \mathrm{B} \le x \le p \end{cases}$$

- The value of P should not be less than A+B-1.



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.

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

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

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

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

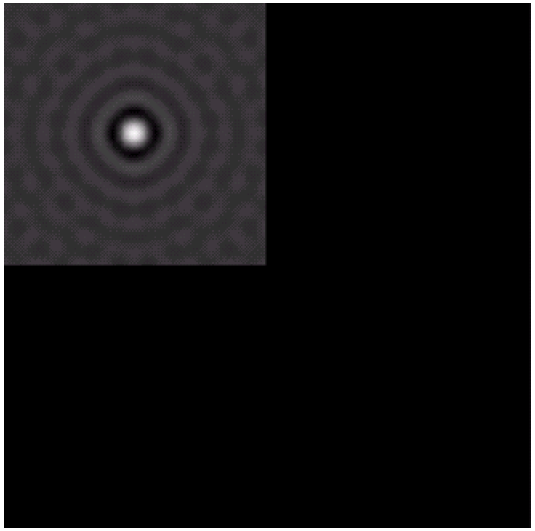


FIGURE 4.39 Padded lowpass filter is the spatial domain (only the real part is shown).

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.

4.6.4 The Convolution and Correlation Theorems

- 2-D Convolution

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

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

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

- 2-D correlation

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

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

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

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

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

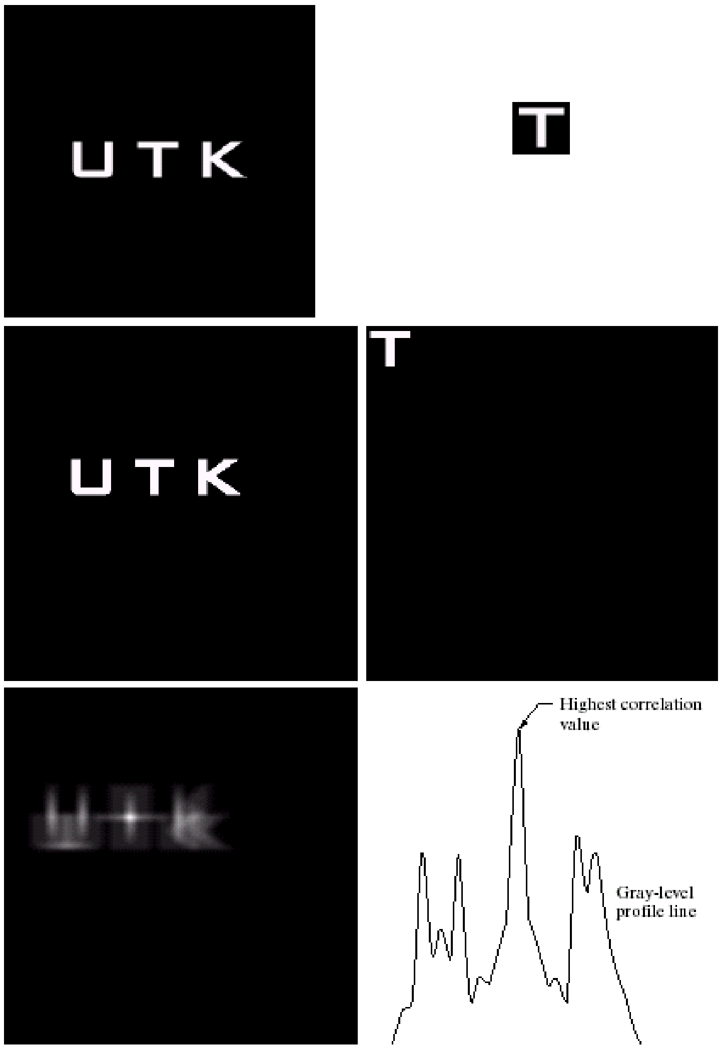


FIGURE 4.41 (a) Image. (b) Template. (c) and (d) Padded images. (e) Correlation function displayed as an image. (f) Horizontal profile line through the highest value in (e), showing the point at which the best match took place.

4.6.5 Summary of Prosperities of the 2-D Fourier Transform

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

Property	Expression(s)
Fourier transform	$F(u,v) = \dfrac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)}$
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	$\phi(u,v) = \tan^{-1}\left[\dfrac{I(u,v)}{R(u,v)}\right]$
Power spectrum	$P(u,v) = \lvert F(u,v)\rvert^2$
Average value	$\bar{f}(x,y) = F(0,0) = \dfrac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$
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	$\text{rect}[a, b] \Leftrightarrow ab\dfrac{\sin(\pi ua)}{(\pi ua)}\dfrac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow \frac{1}{2}[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)]$
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.

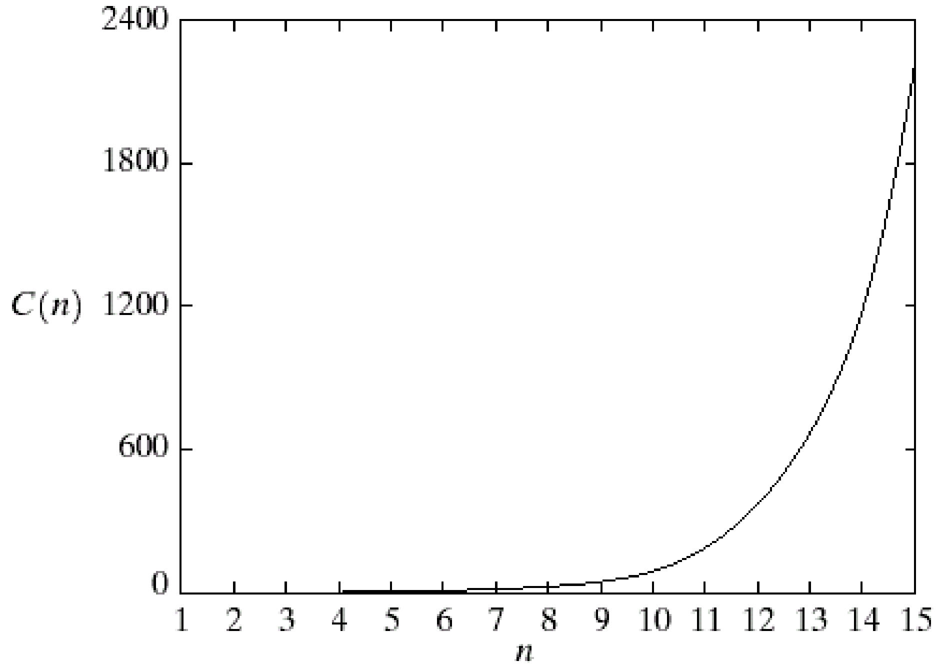


FIGURE 4.42 Computational advantage of the FFT over a direct implementation of the 1-D DFT. Note that the advantage increases rapidly as a function of n.