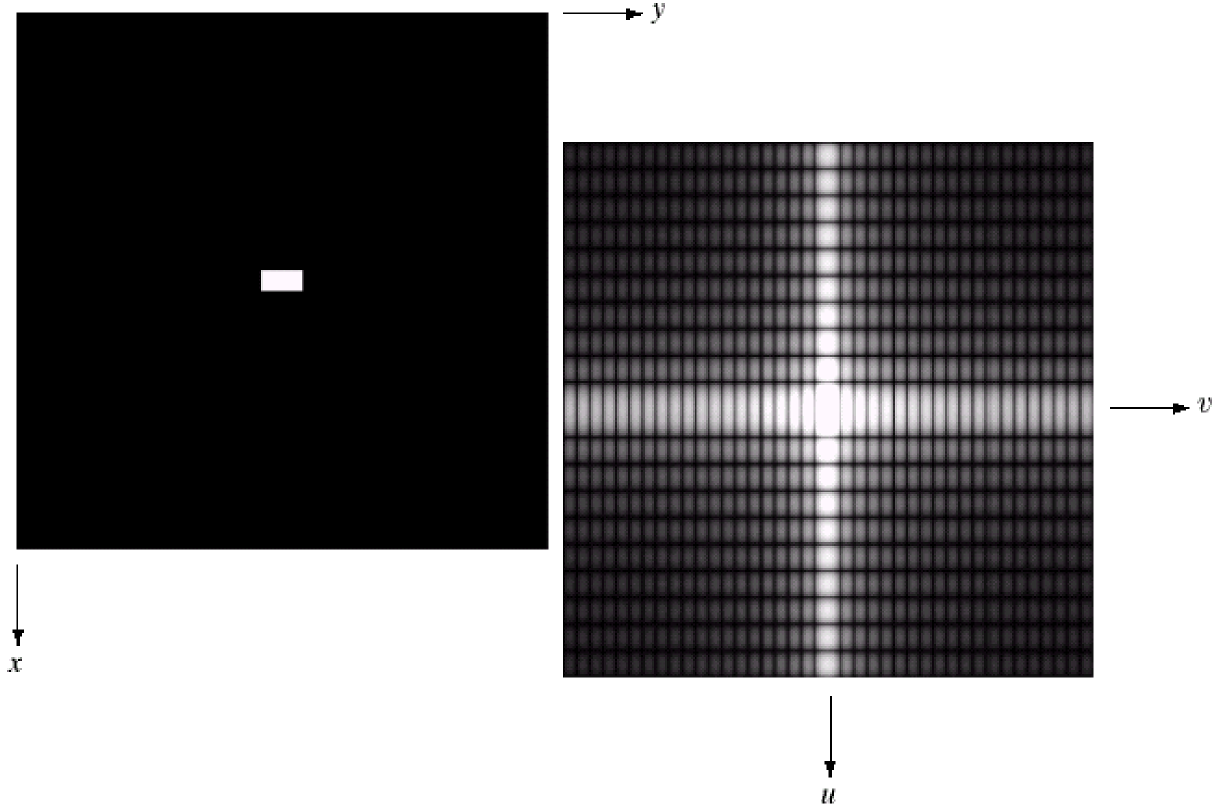
Image Enhancement in the Frequency Domain

Chang-Su Kim

Discrete Fourier Transform - Examples

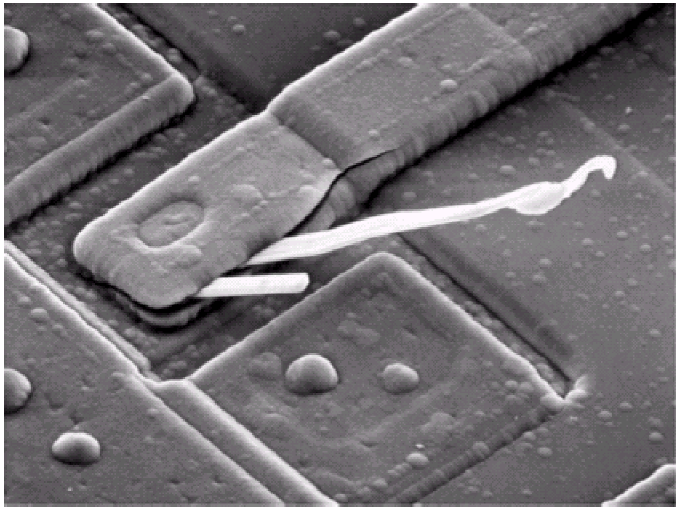


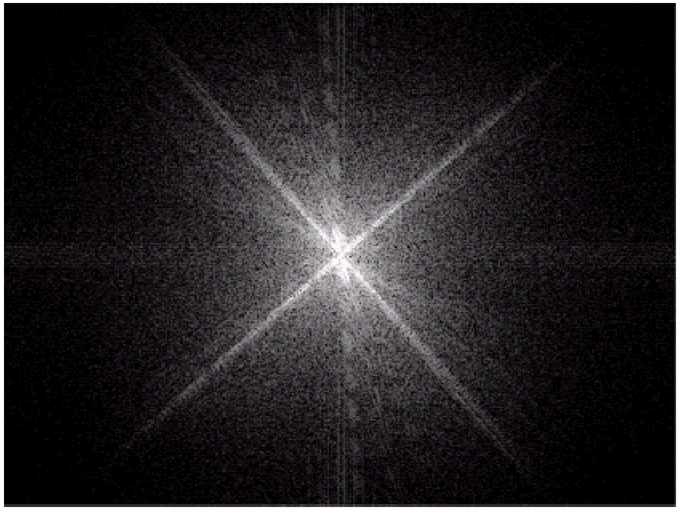
a b

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.

- Zero frequency is located at the center
- Inverse relationship between space and frequency
  - The separation of spectrum zeros in vertical direction is exactly twice that in horizontal direction

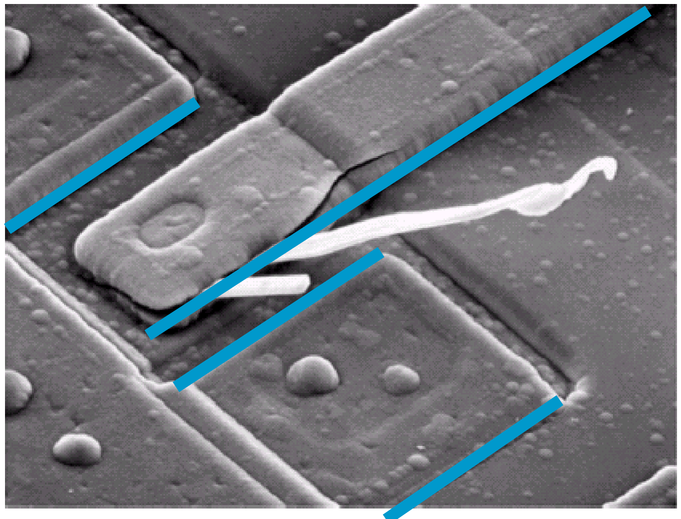
Discrete Fourier Transform - Examples

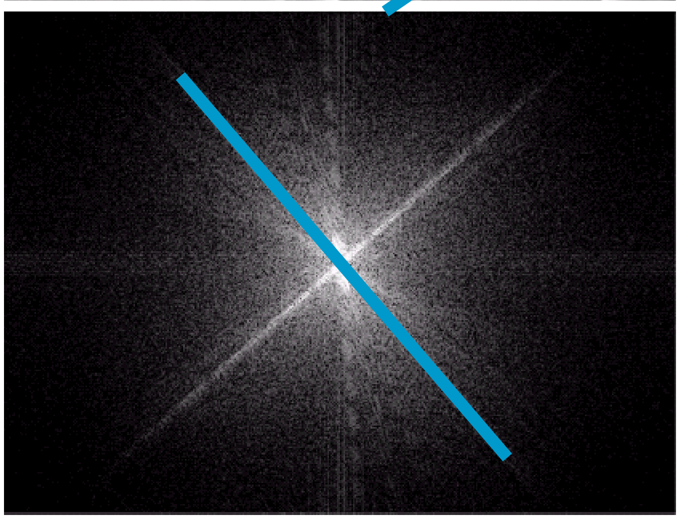




- Edge in space
  - Edge in frequency
  - 90^o rotated clockwise

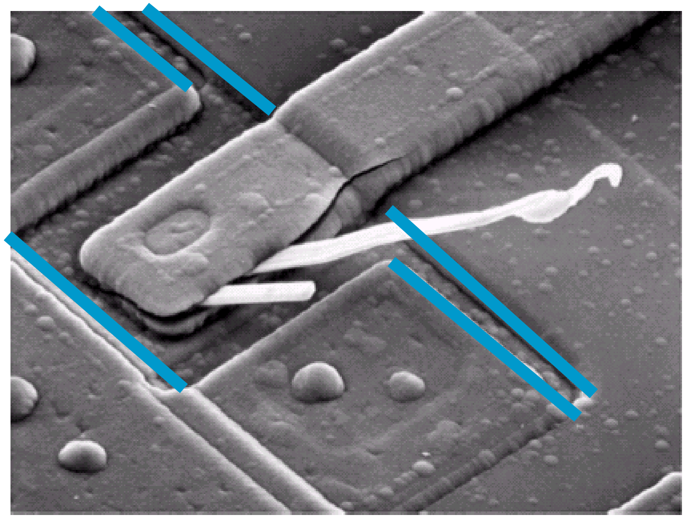
Discrete Fourier Transform - Examples

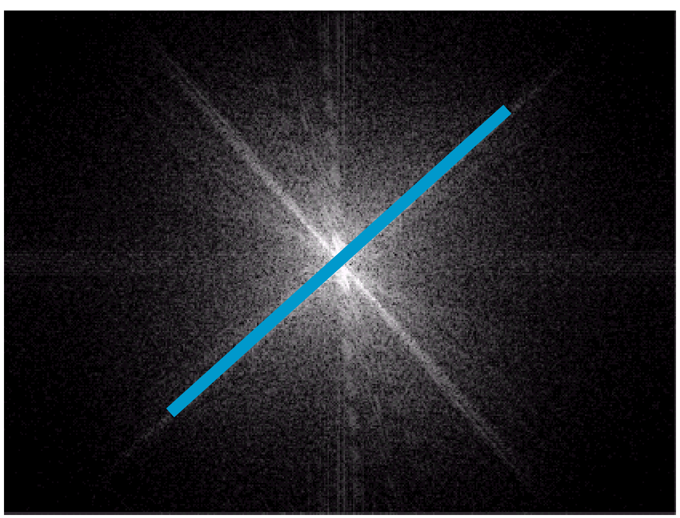




- Edge in space
  - Edge in frequency
  - 90^{o} rotated clockwise

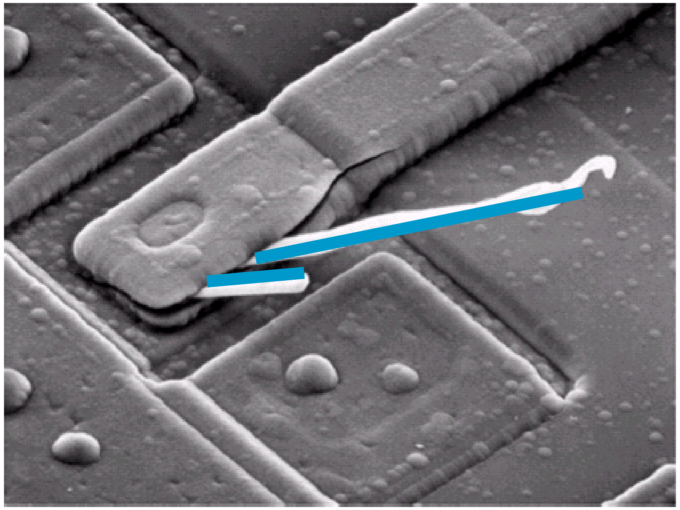
Discrete Fourier Transform - Examples

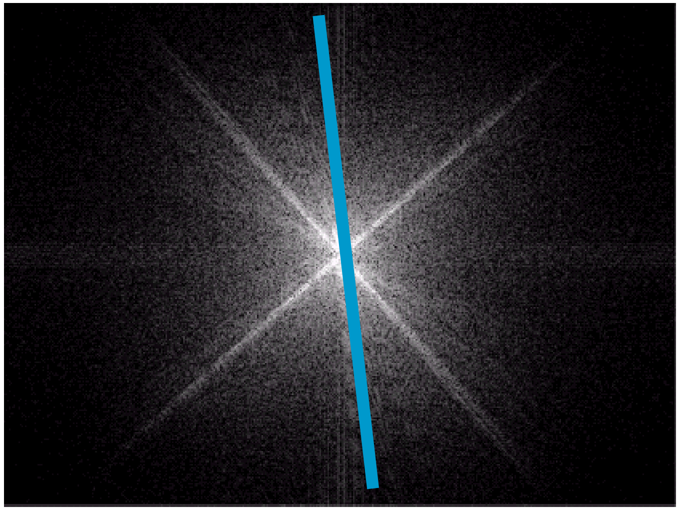




- Edge in space
  - Edge in frequency
  - 90^o rotated clockwise

Discrete Fourier Transform - Examples

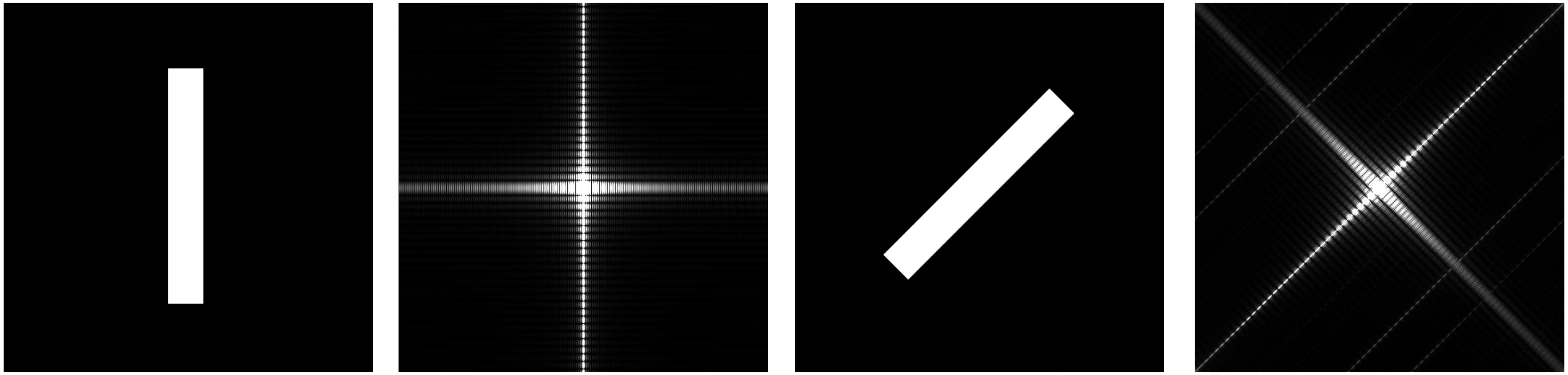




- Edge in space
  - Edge in frequency
  - 90^{o} rotated clockwise

Discrete Fourier Transform - Examples

- Rotation in space $\longrightarrow$ Rotation in frequency



(a) a sample image (b) its spectrum (c) rotated image (d) resulting spectrum

Filtering in Frequency Domain

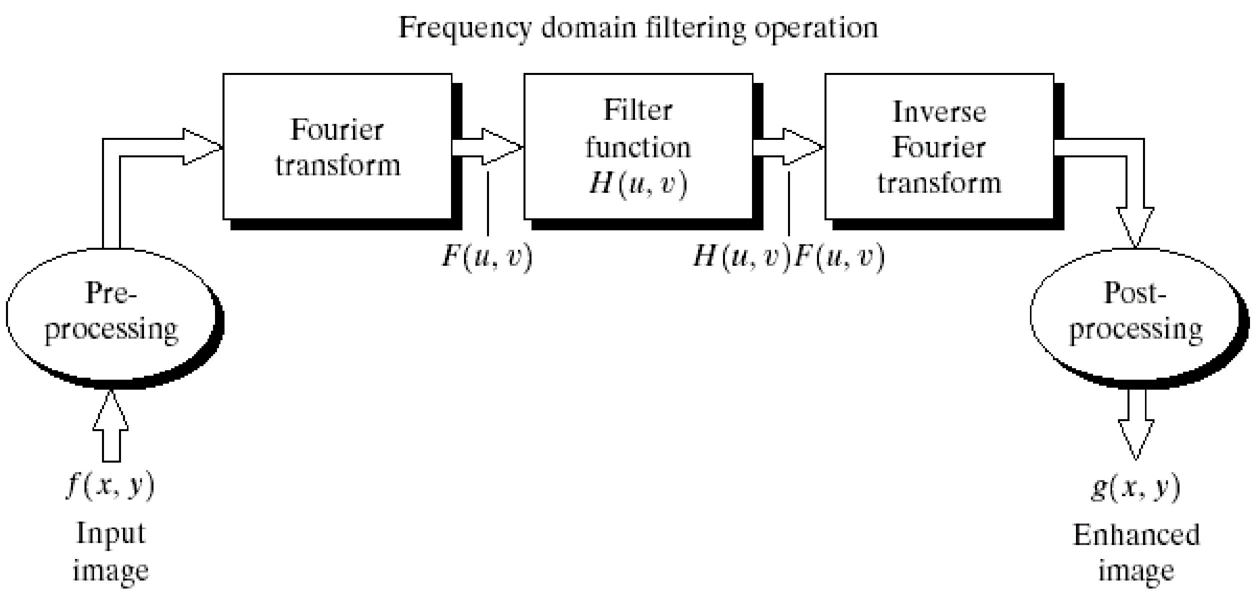


FIGURE 4.5 Basic steps for filtering in the frequency domain.

- $G(u, v) = H(u, v)F(u, v)$
- $g(x, y) = \text{IDFT}[G(u, v)]$

Filtering in Frequency Domain

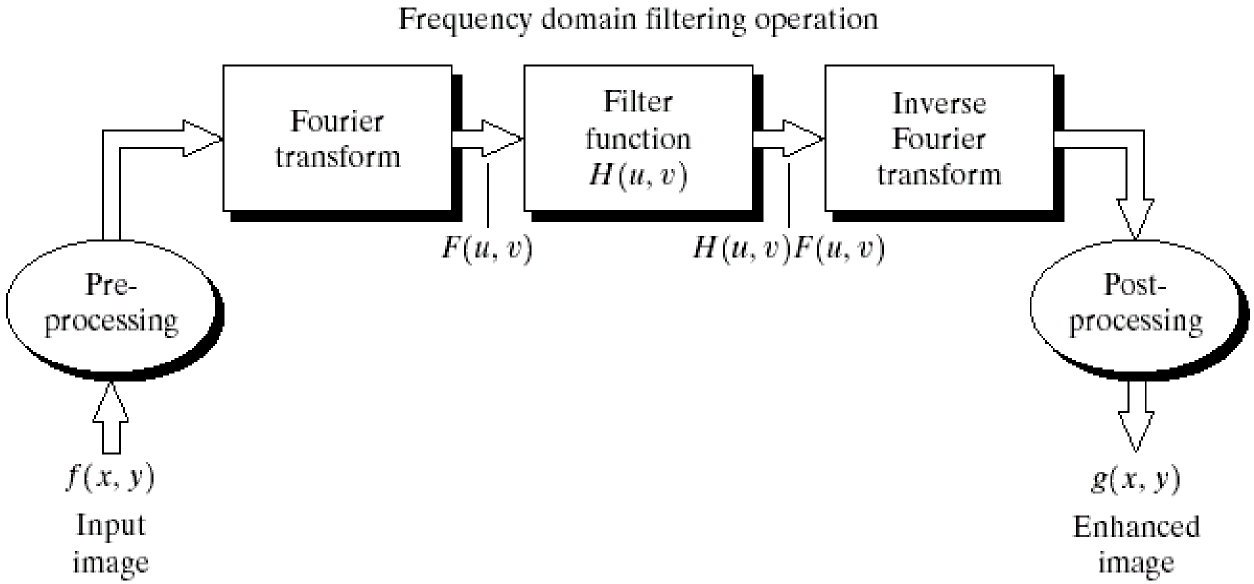


FIGURE 4.5 Basic steps for filtering in the frequency domain.

- F(u,v) is complex, although f(x,y) is a real image
- H(u,v)
  - Real : zero-phase-shift filter
  - radially symmetric about origin
- g(x,y) is a real image
  - Zero-forcing of imaginary parts to avoid round-off errors

Filtering in Frequency Domain

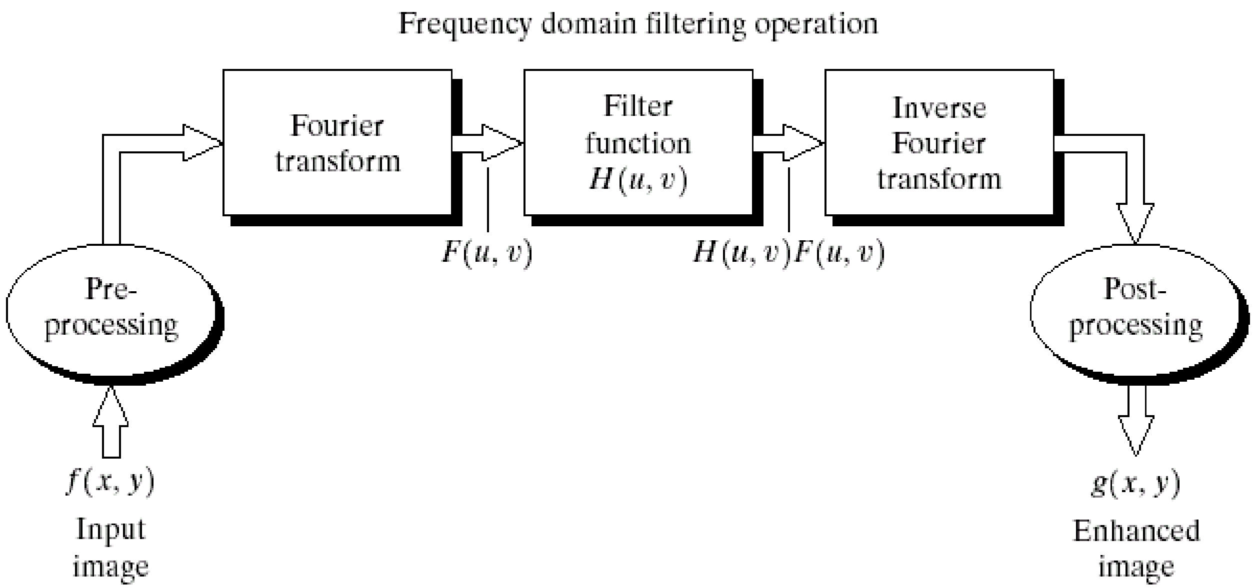
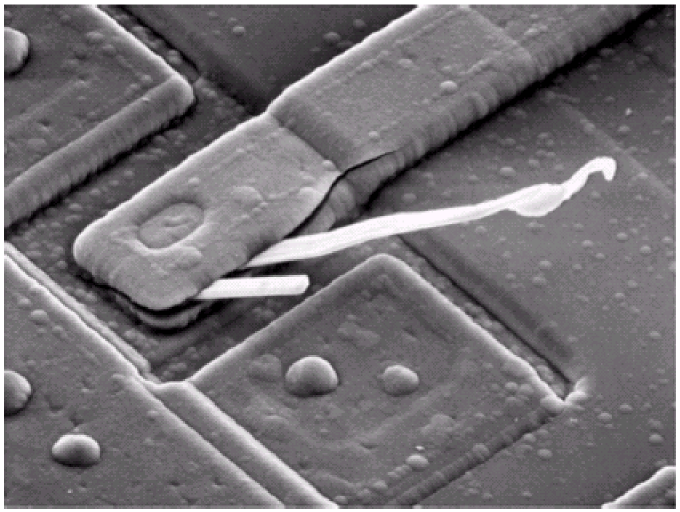
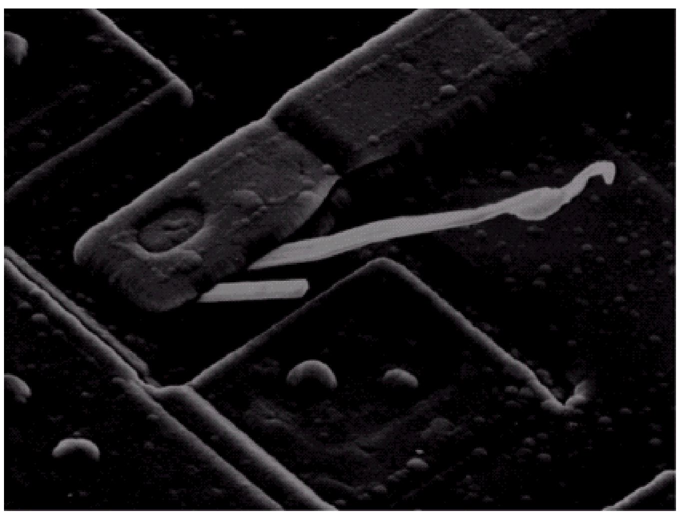


FIGURE 4.5 Basic steps for filtering in the frequency domain.

- Preprocessing
  - Image cropping
  - Zero padding
  - Data conversion to ‘float’
- Postprocessing
  - Image cropping
  - Zero-forcing of imaginary parts
  - Data conversion to ‘unsigned char’

Notch Filter





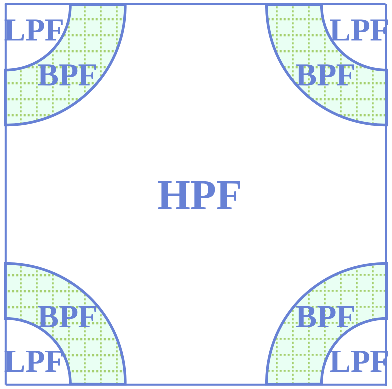
$$H(u,v) = \begin{cases} 0 & \text{if } (u,v) = (0,0) \\ 1 & \text{otherwise} \end{cases}$$

- Suppression of DC components
- linear scaling for display

- Notch filter
  - Suppress certain frequency components while preserving the other components
  - cf. bandpass filter
  - Notch filter is used to identify spatial image effects caused by specific, localized frequency domain components

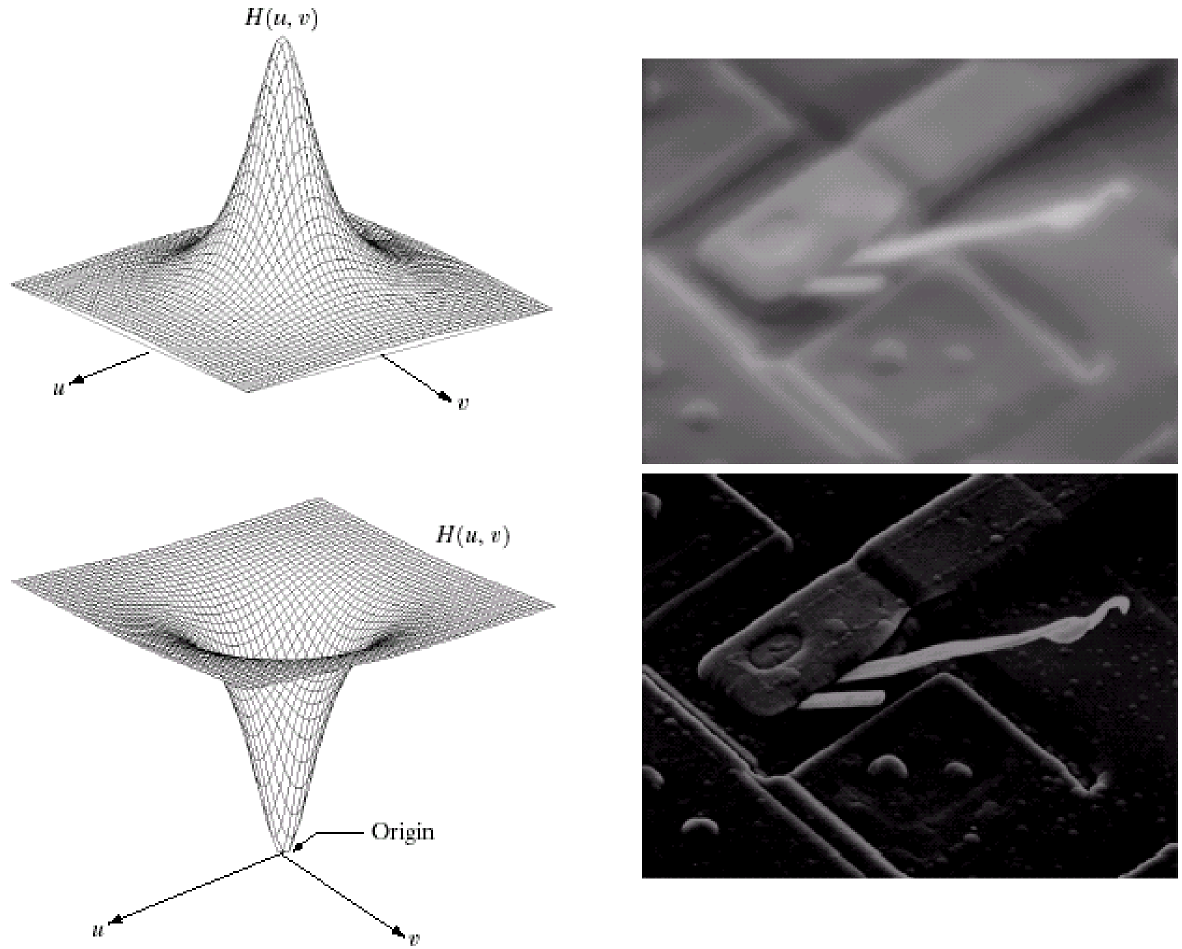
Lowpass Filter vs. Highpass Filter

- Low frequencies
  - General gray level appearance over smooth areas
- High frequencies
  - Details, edges and noise



- Lowpass filtered image
  - Less sharp details than original image
- Highpass filtered image
  - Less gray level variation over smooth areas with emphasized details

Lowpass Filter vs. Highpass Filter

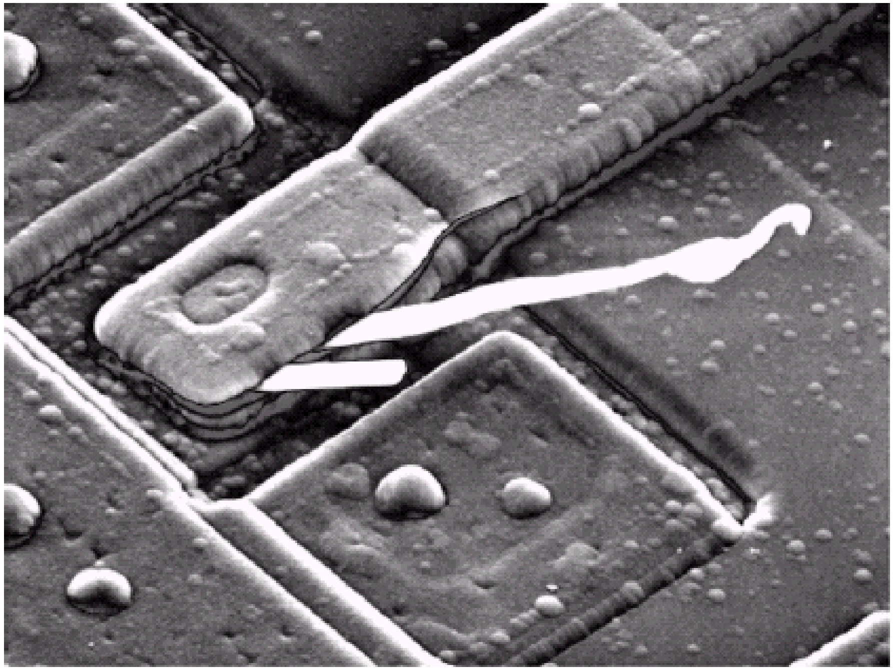


a b
c d

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

Modified Highpass Filter

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).



Space Domain vs. Frequency Domain

- Comparison of computations
  - N: number of pixels in the image
  - Frequency domain
    - $F(u,v) = FFT\,[f(x,y)]$ — $O(N \log N)$
    - $H(u,v) = FFT\,[h(x,y)]$ — $O(N \log N)$
    - $G(u,v) = F(u,v)\,H(u,v)$ — $O(N)$
    - $g(x,y) = IFFT\,[G(u,v)]$ — $O(N \log N)$
    - Total — $O(N \log N)$
  - Spatial domain
    - C: number of non-zero filter coefficients
    - $1 \leq C \leq N$
    - Requires CN multiplications and (C-1)N additions
  - Spatial domain approach is preferred when C is much less than N (typical cases)
  - Frequency domain approach is preferred when C is close to N

a	b	c
d	e	f
g	h	i

C=9

Space Domain vs. Frequency Domain

- General procedure to design filters
  1. Specify filter characteristics H(u,v) in frequency domain
     - Frequency domain is more intuitive (lowpass, highpass, bandpass, etc)
  2. Inverse transform H(u,v) to spatial domain h(x,y)
  3. Approximate h(x,y) with a small mask h'(x,y)
     - Computationally more efficient to use small mask in spatial domain

Space Domain vs. Frequency Domain

- Gaussian filter
  - Lowpass

$$
\begin{aligned}
H(u) &= Ae^{-\frac{u^2}{2\sigma^2}} \\
h(x) &= \sigma\sqrt{2\pi}Ae^{-2\pi^2\sigma^2x^2}
\end{aligned}
$$

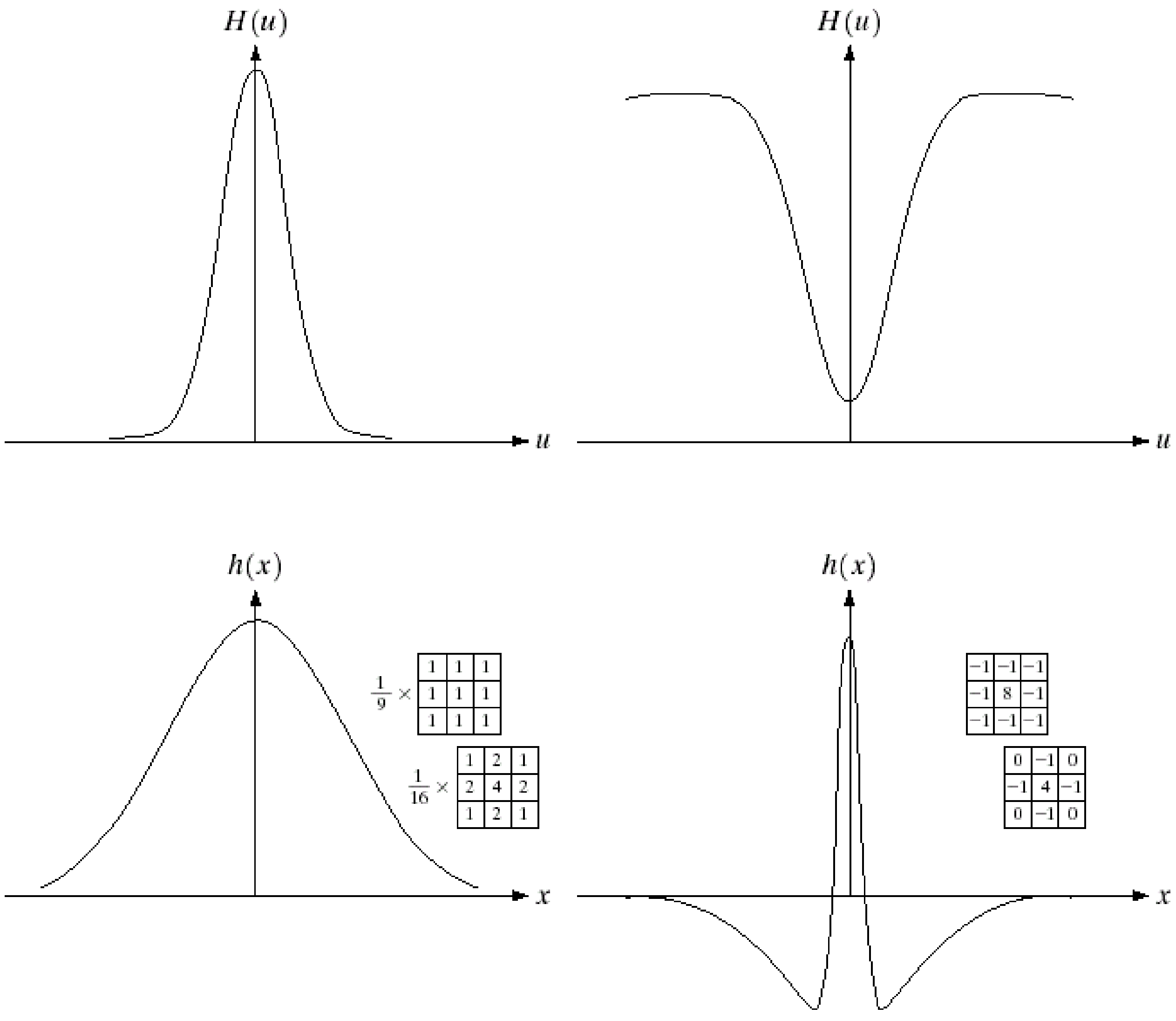
    - Note the reciprocal nature (inverse relationship) between space and frequency

  - Highpass

$$
\begin{aligned}
H(u) &= Ae^{-\frac{u^2}{2\sigma_1^2}} - Be^{-\frac{u^2}{2\sigma_2^2}} \\
h(x) &= \sigma_1\sqrt{2\pi}Ae^{-2\pi^2\sigma_1^2x^2} - \sigma_2\sqrt{2\pi}Be^{-2\pi^2\sigma_2^2x^2}
\end{aligned}
$$

    - A>B and $\sigma_1 > \sigma_2$

Space Domain vs. Frequency Domain



a b
c d

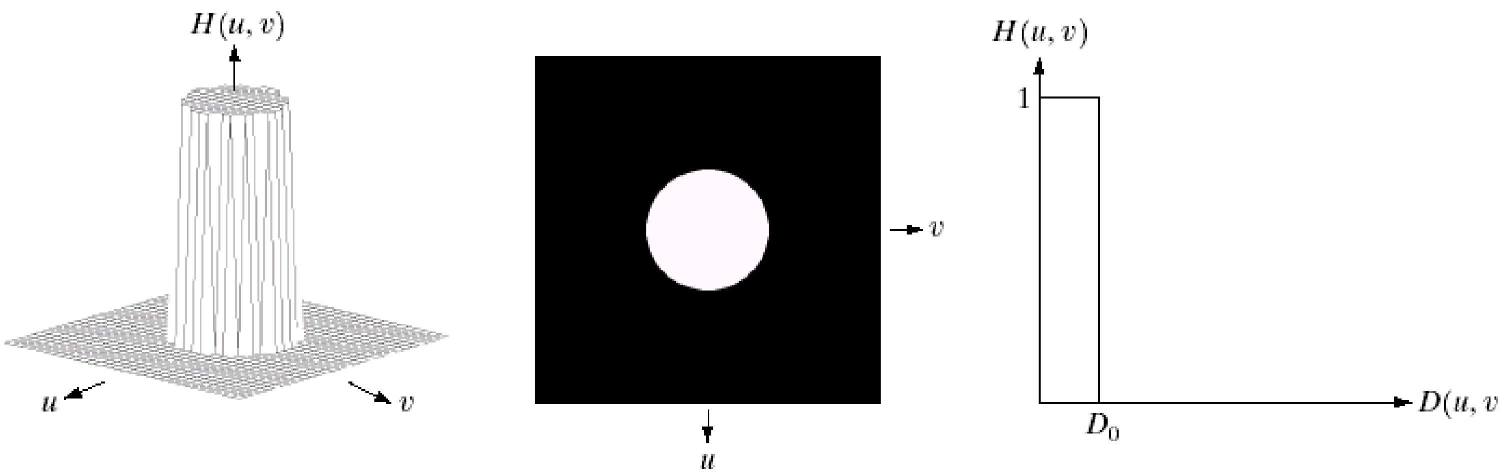
FIGURE 4.9
(a) Gaussian frequency domain lowpass filter.
(b) Gaussian frequency domain highpass filter.
(c) Corresponding lowpass spatial filter.
(d) Corresponding highpass spatial filter. The masks shown are used in Chapter 3 for lowpass and highpass filtering.

Ideal Lowpass Filter

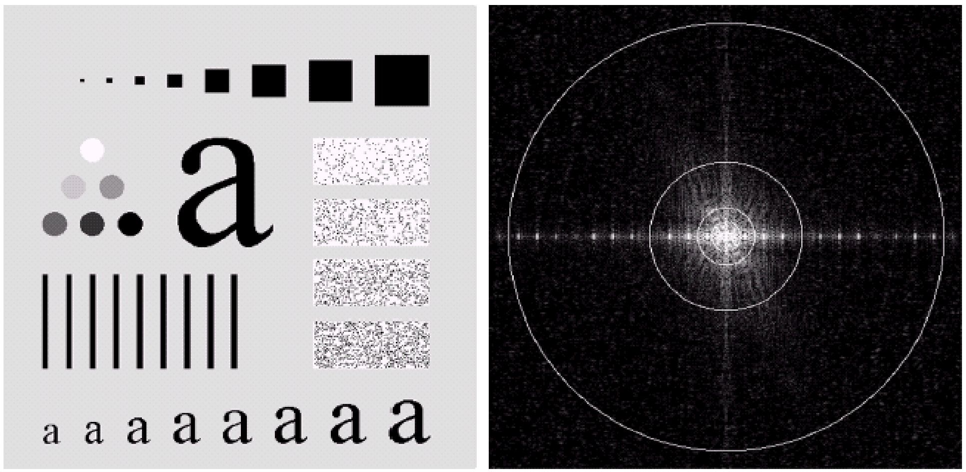
- Distance from the origin
  - $D(u,v) = (u^2+v^2)^{1/2}$

- Transfer function

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



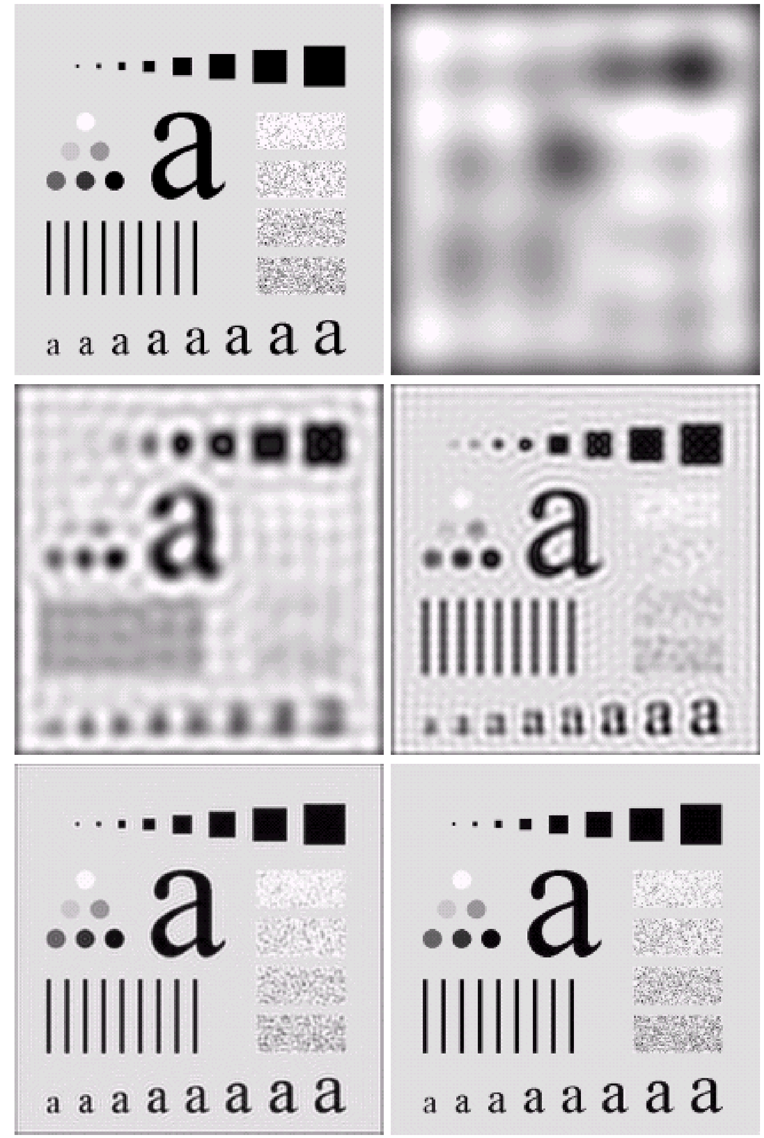
Ideal Lowpass Filter



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.

$$\text{Image Power} = \sum_u \sum_v |F(u,v)|^2$$
$$= \sum_u \sum_v P(u,v)$$



Ringing Artifacts

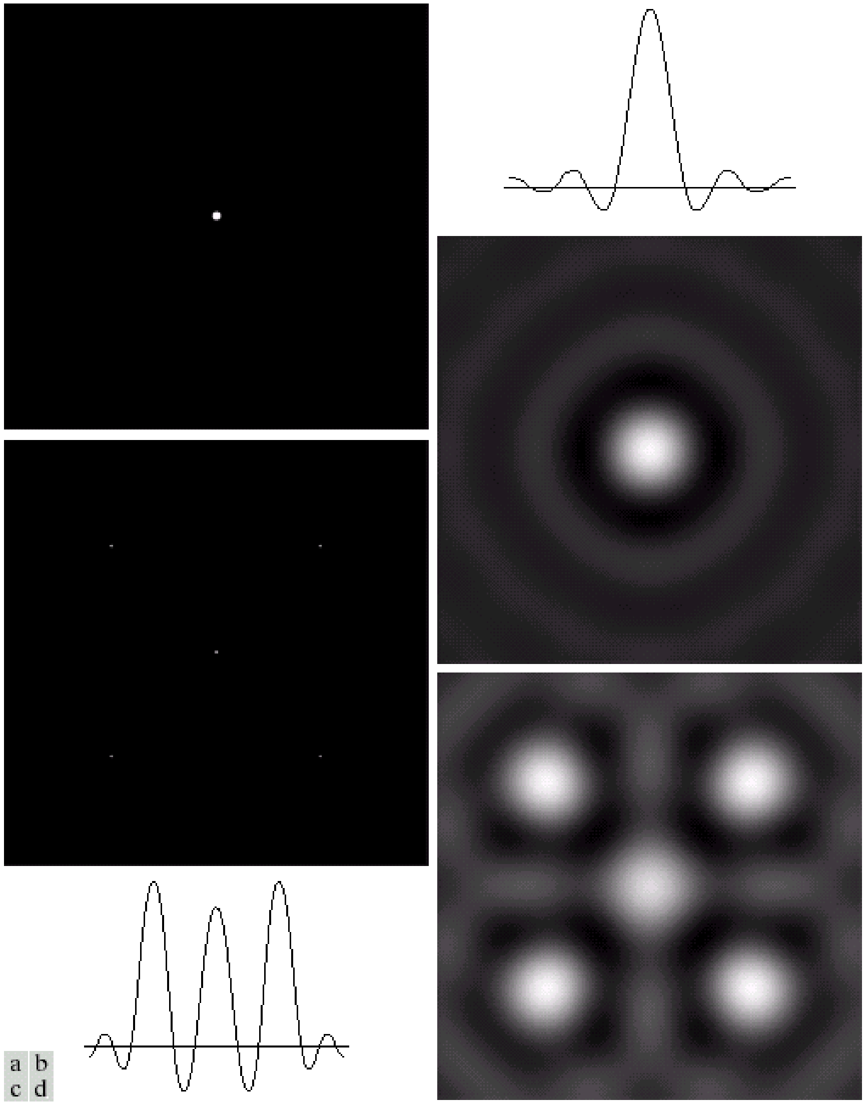


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

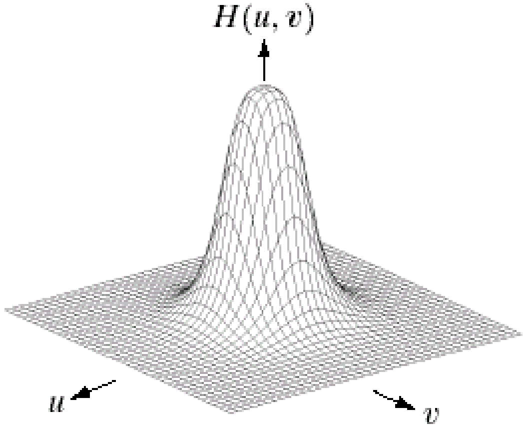
- sinc function in spatial domain
  - Dominant center component
    - blurring
  - Circular (+/-) components
    - ringing
- Reciprocal nature between space and frequency
  - Narrow lowpass ~ wider ringing

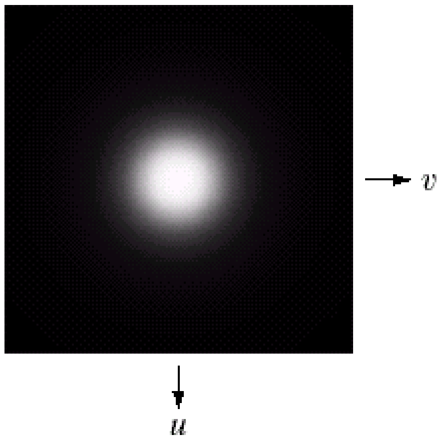
Butterworth Lowpass Filters

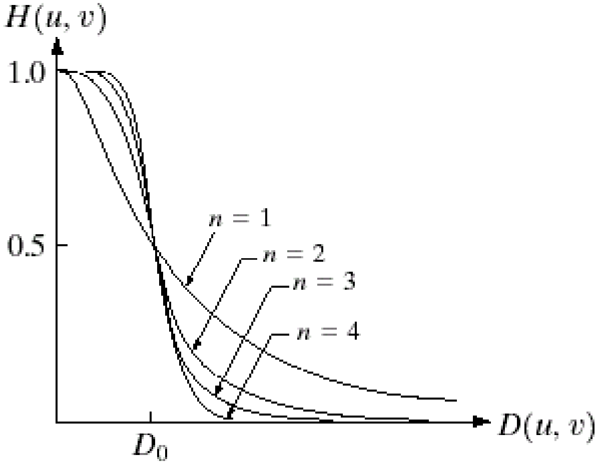
- Perform blurring without ringing
- Transfer function of a Butterworth filter of order n

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

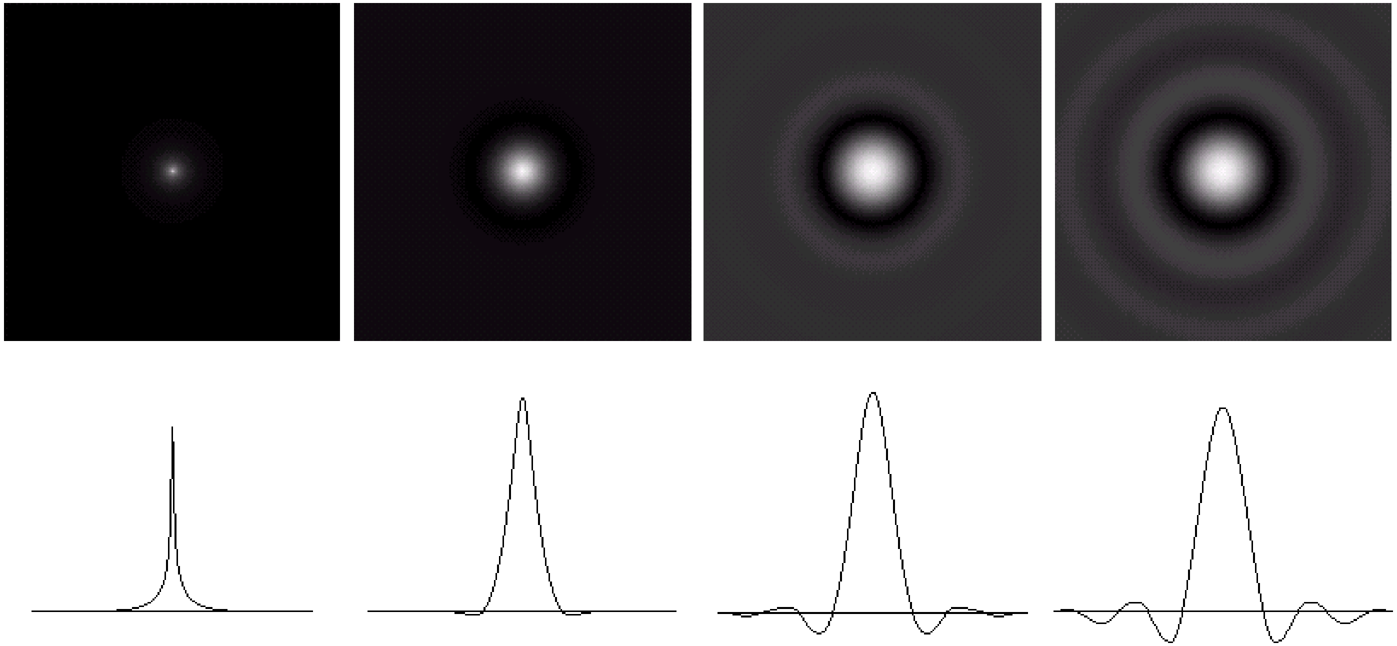
  - D_0: cutoff frequency distance
    - $H(u,v) = 0.5$ when $D(u,v) = D_0$







Butterworth Lowpass Filter

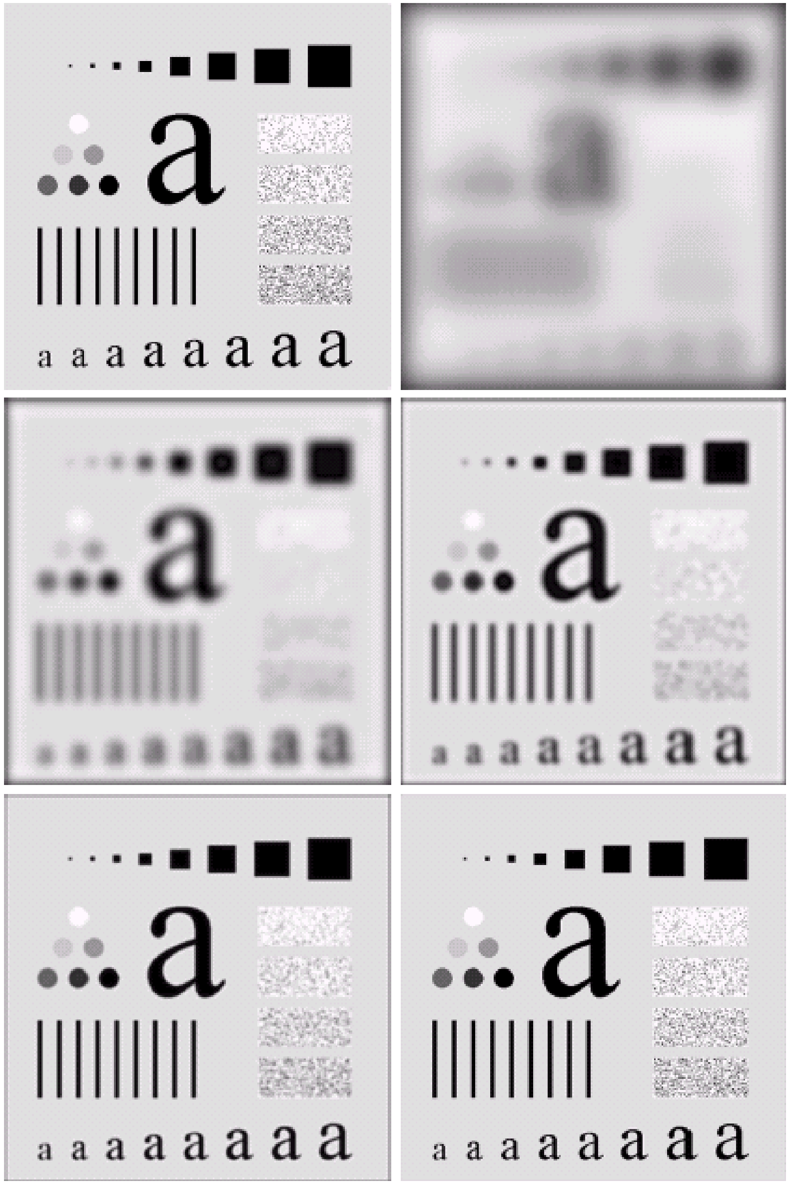


a b c d

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.

- Ringing can be observed as order n increases
- In the extreme case of n= ∞, the Butterworth filter is identical to the ideal filter

Butterworth Lowpass Filter (order 2)



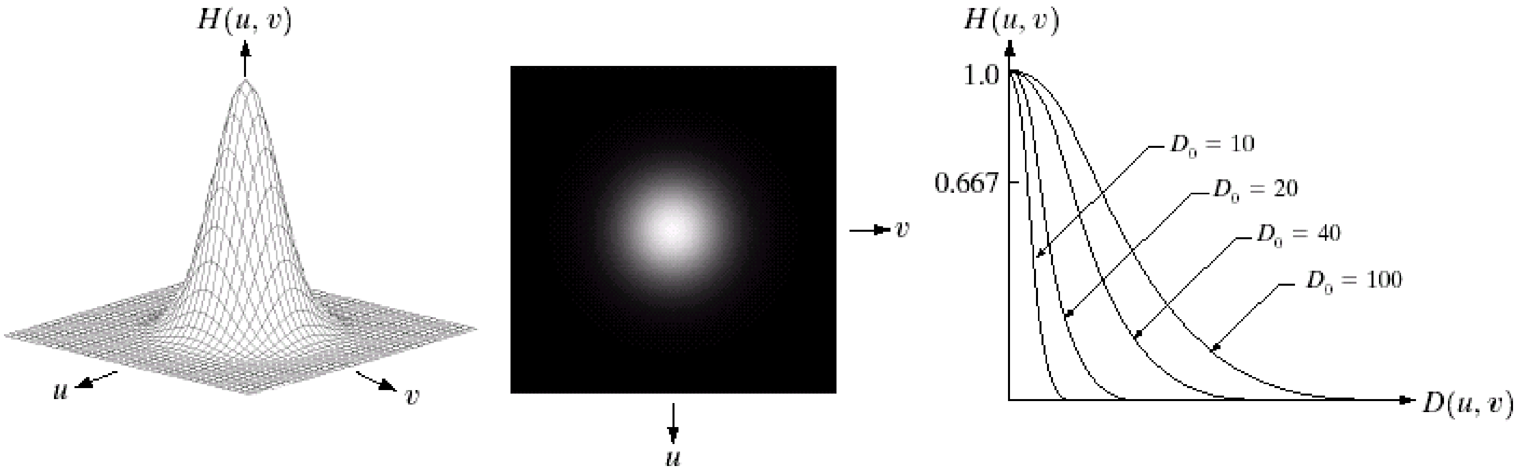
Butterworth Lowpass Filter (order 2)

Gaussian Lowpass Filters

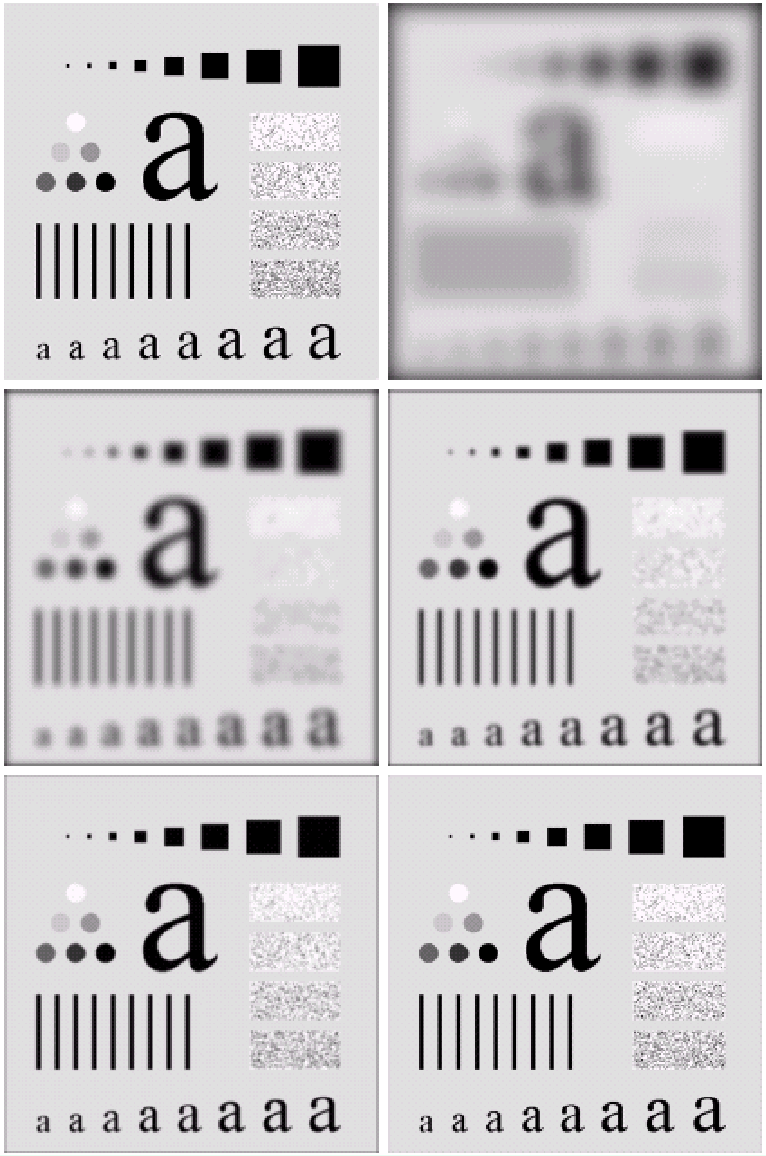
- Transfer Function

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

  - D_0: cutoff frequency distance
    - $H(u,v) = 0.667$ when $D(u,v) = D_0$



Gaussian Lowpass Filters



- Spatial response is also Gaussian
- No ringing is guaranteed
- Less sharp transition around cutoff frequency than Butterworth filters

Applications of Lowpass Filtering

- Preprocessing before machine recognition
  - Removal of small gaps

a b

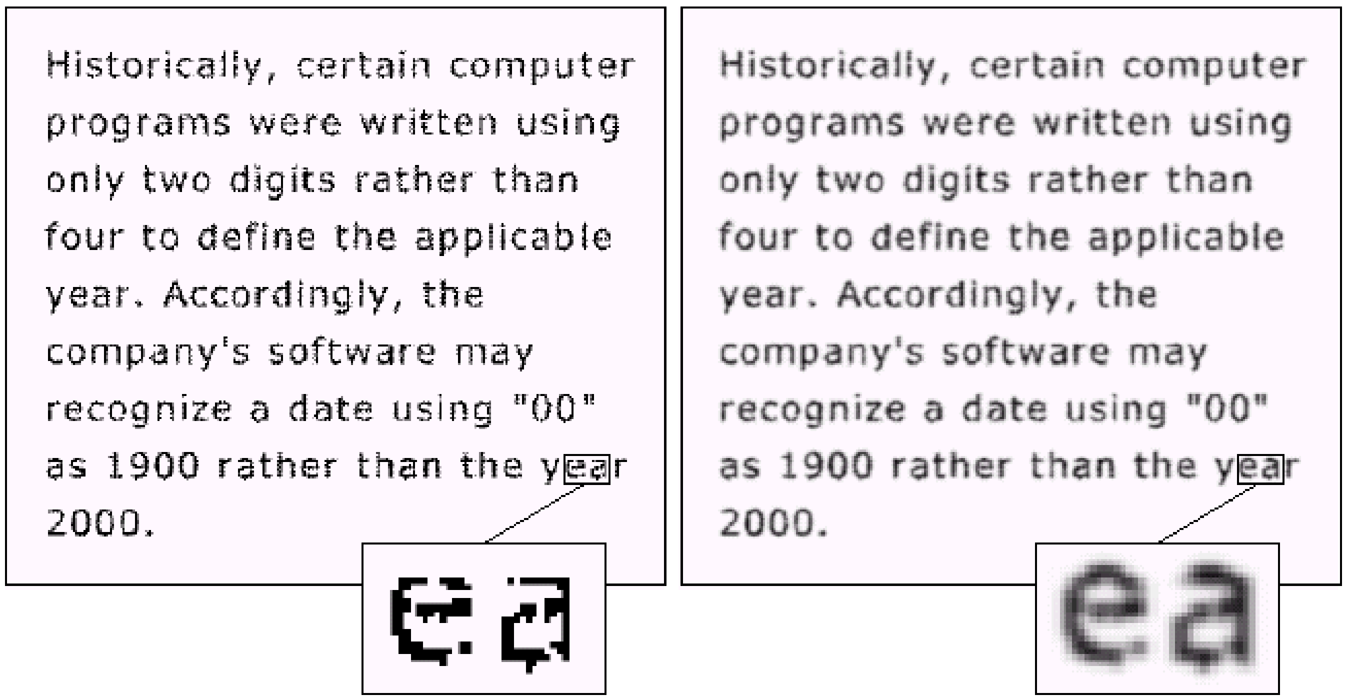


FIGURE 4.19 (a) Sample text of poor resolution (note broken characters in magnified view). (b) Result of filtering with a GLPF (broken character segments were joined).

Applications of Lowpass Filtering

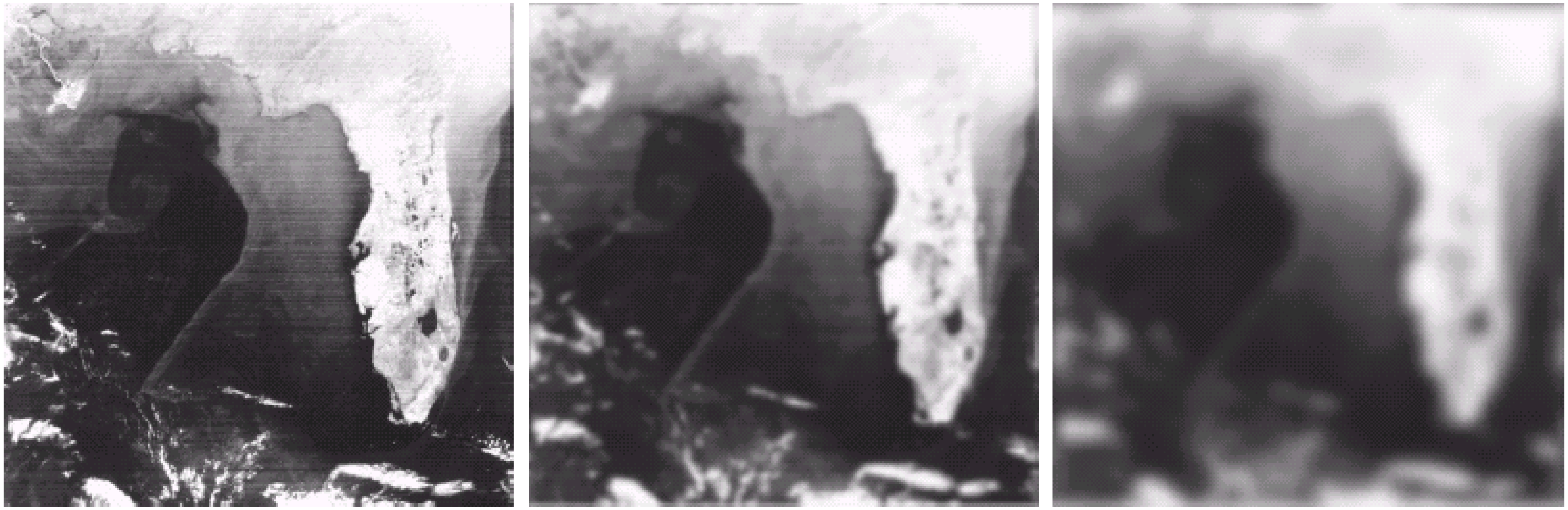
- Cosmetic processing of photos

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).

Applications of Lowpass Filtering

- Enhancement of Satellite Images
  - Removal of horizontal sensor scan lines



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.)

Highpass Filters

- Reverse operation of lowpass filters
- Transfer function
  - $H_{hp}(u,v) = 1 - H_{lp}(u,v)$
  - A highpass filter $H_{hp}(u,v)$ can be designed using a lowpass filter $H_{lp}(u,v)$
- Impulse response
  - $h_{hp}(x,y) = \delta(x,y) - h_{lp}(x,y)$
- We also consider only zero-phase-shift radially symmetric filters

Highpass Filters – Transfer Functions

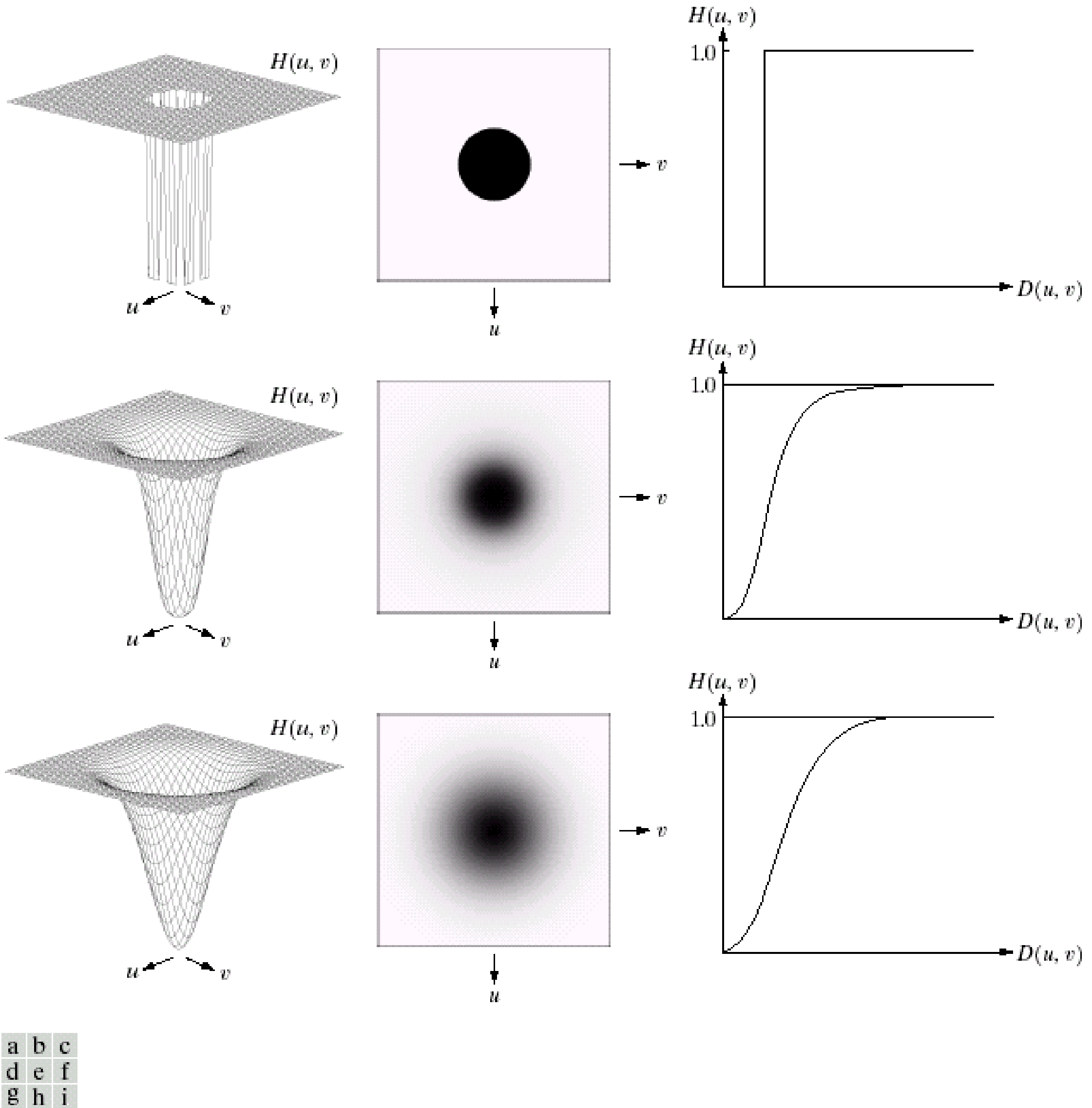


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

Highpass Filters – Impulse Responses

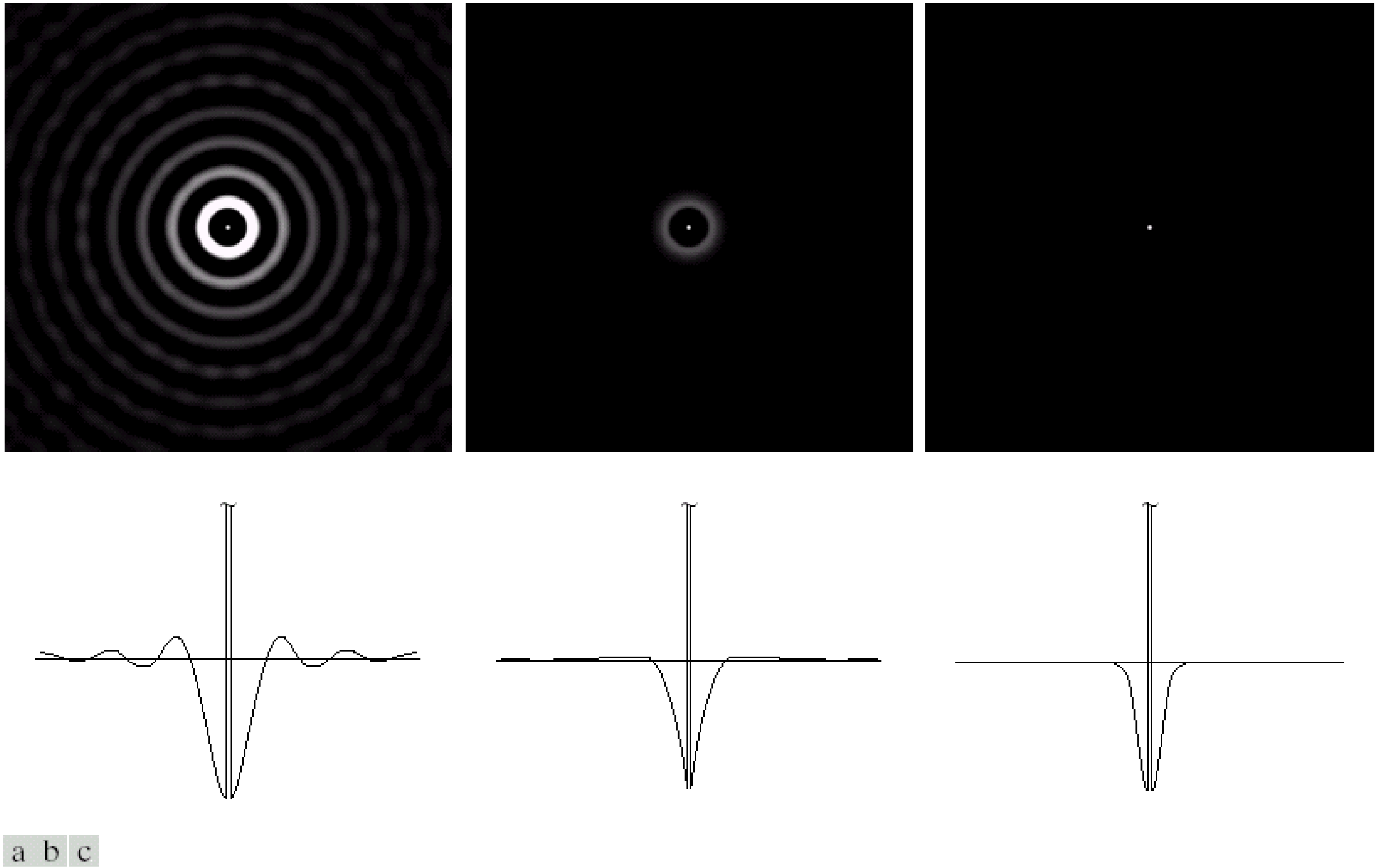
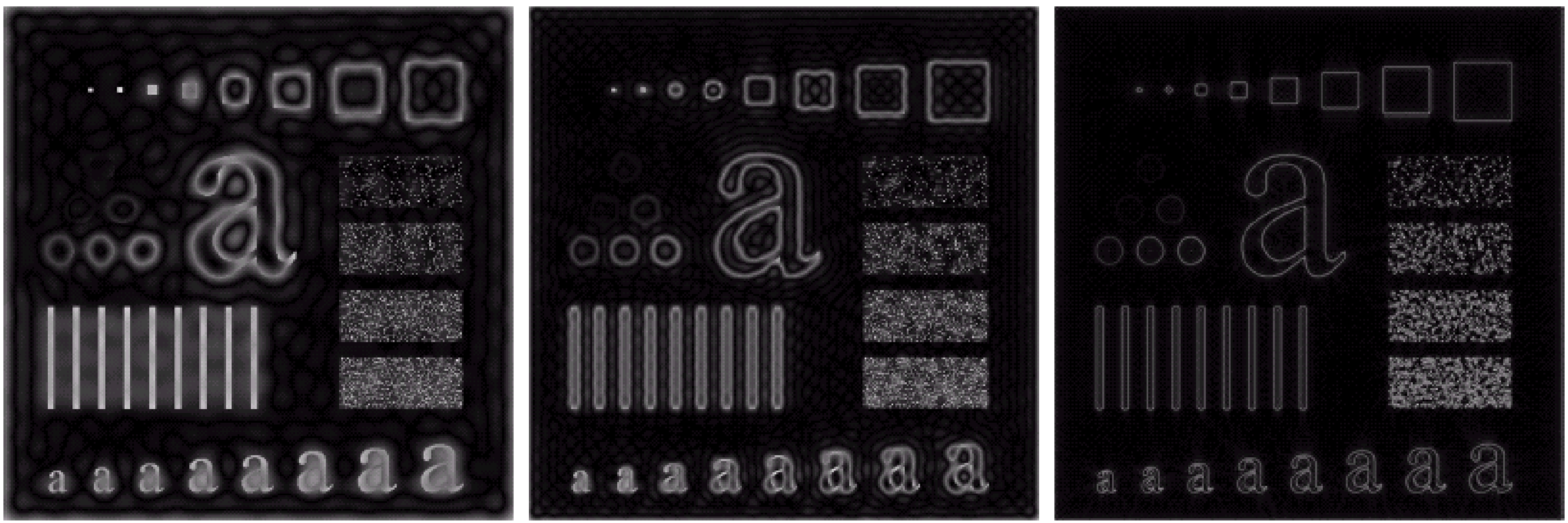


FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

Ideal Highpass Filters

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

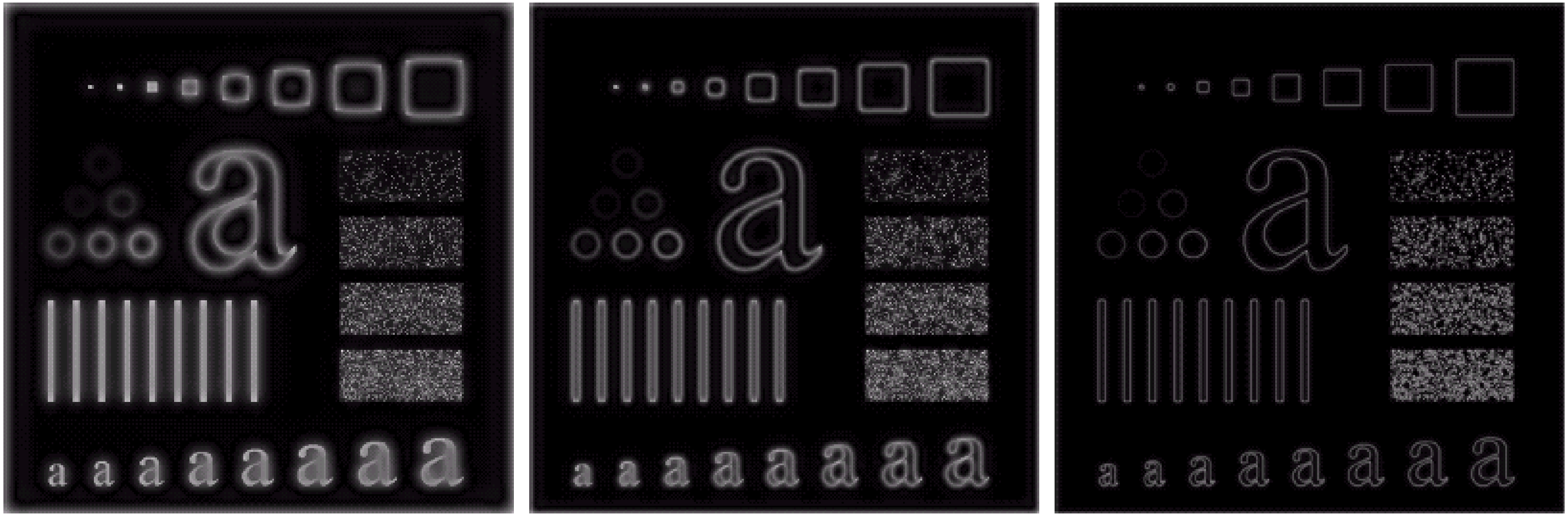


a b c

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).

Butterworth Highpass Filters

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

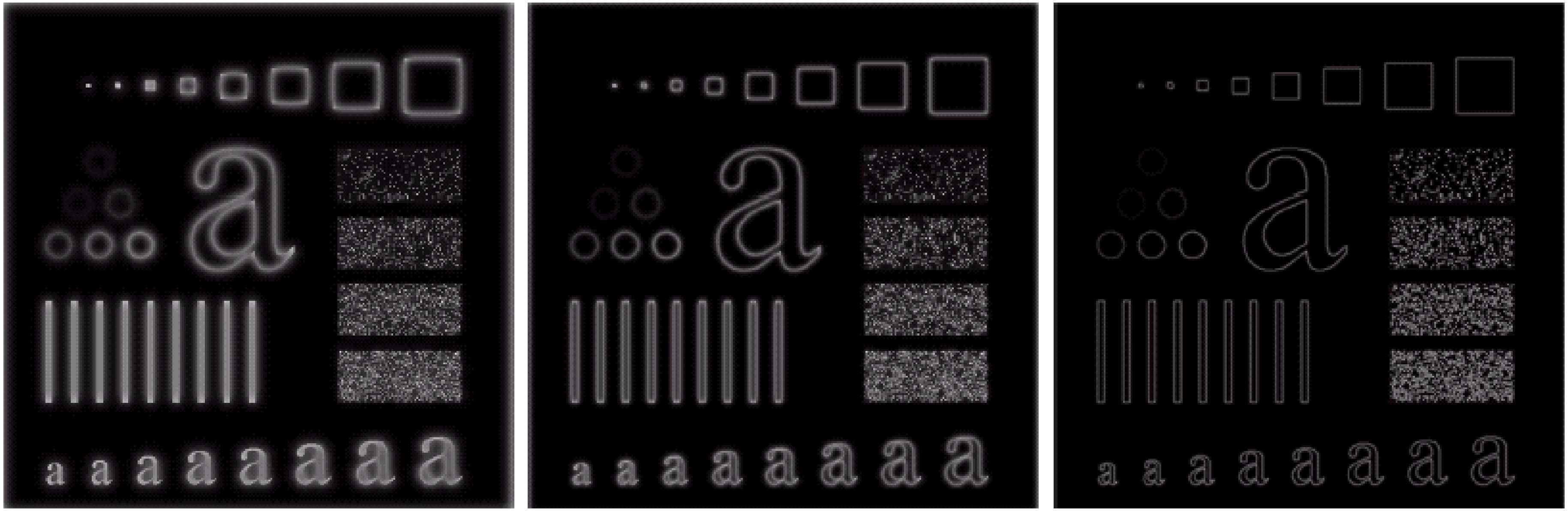


a b c

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.

Gaussian Highpass Filters

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



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.

Laplacian in Frequency Domain

- In continuous domain

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

- Laplacian in FD

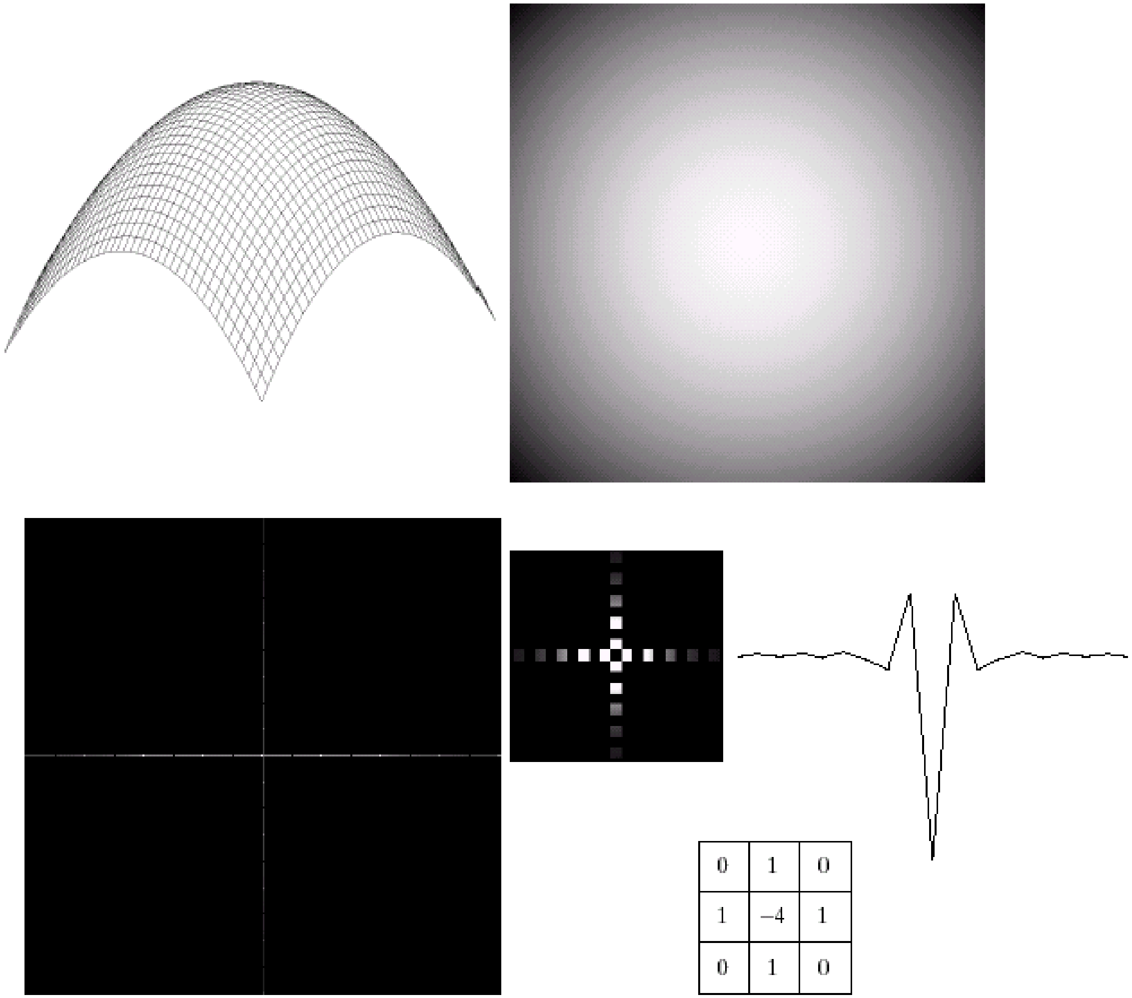
$$\begin{aligned}\mathcal{F}[\frac{d^2 f(x,y)}{dx^2} + \frac{d^2 f(x,y)}{dy^2}] &= (ju)^2 F(u,v) + (jv)^2 F(u,v) \\ &= -(u^2 + v^2) F(u,v)\end{aligned}$$

- Laplacian transfer function

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

Laplacian in Frequency Domain

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



0	1	0
1	–4	1
0	1	0

a b
c d e
f

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

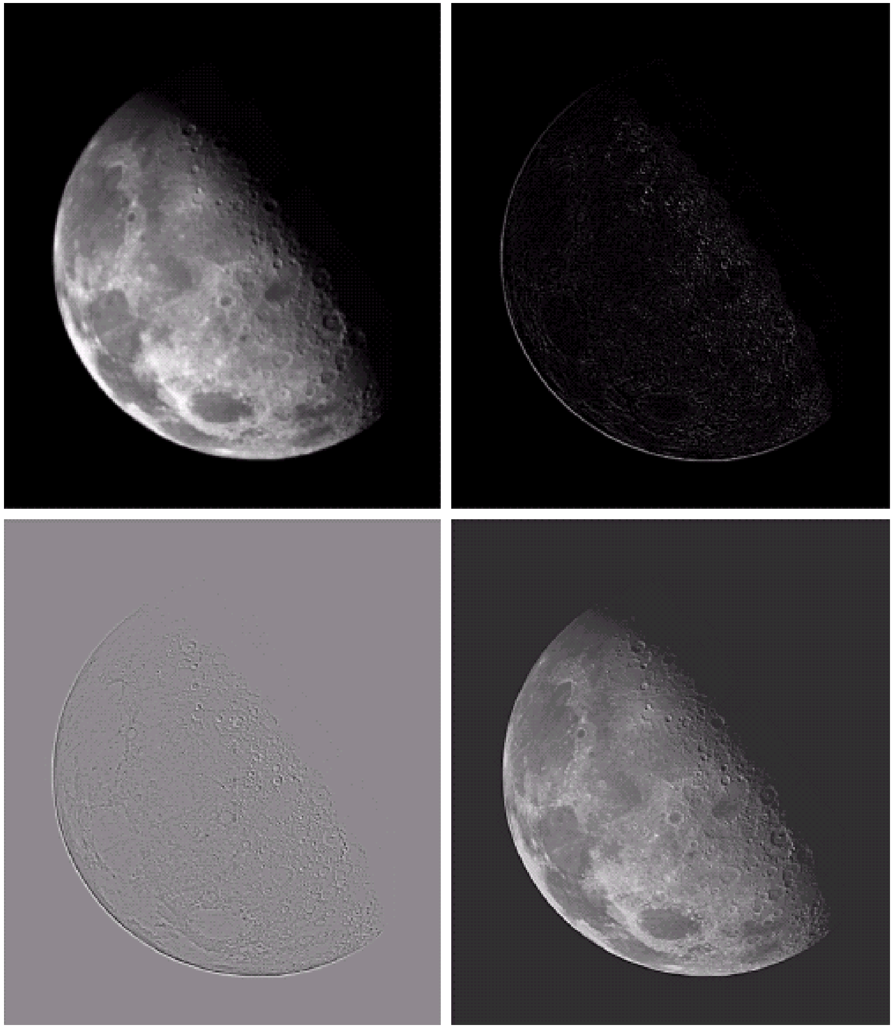
Laplacian Sharpening Filter

- Positive definition of Laplacian
  - $H(u,v) = u^2+v^2$
- Recall

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

0	-1	0
-1	5	-1
0	-1	0

- Transfer function of Laplacian sharpening filter
  - $H(u,v) = 1+ (u^2+v^2)$

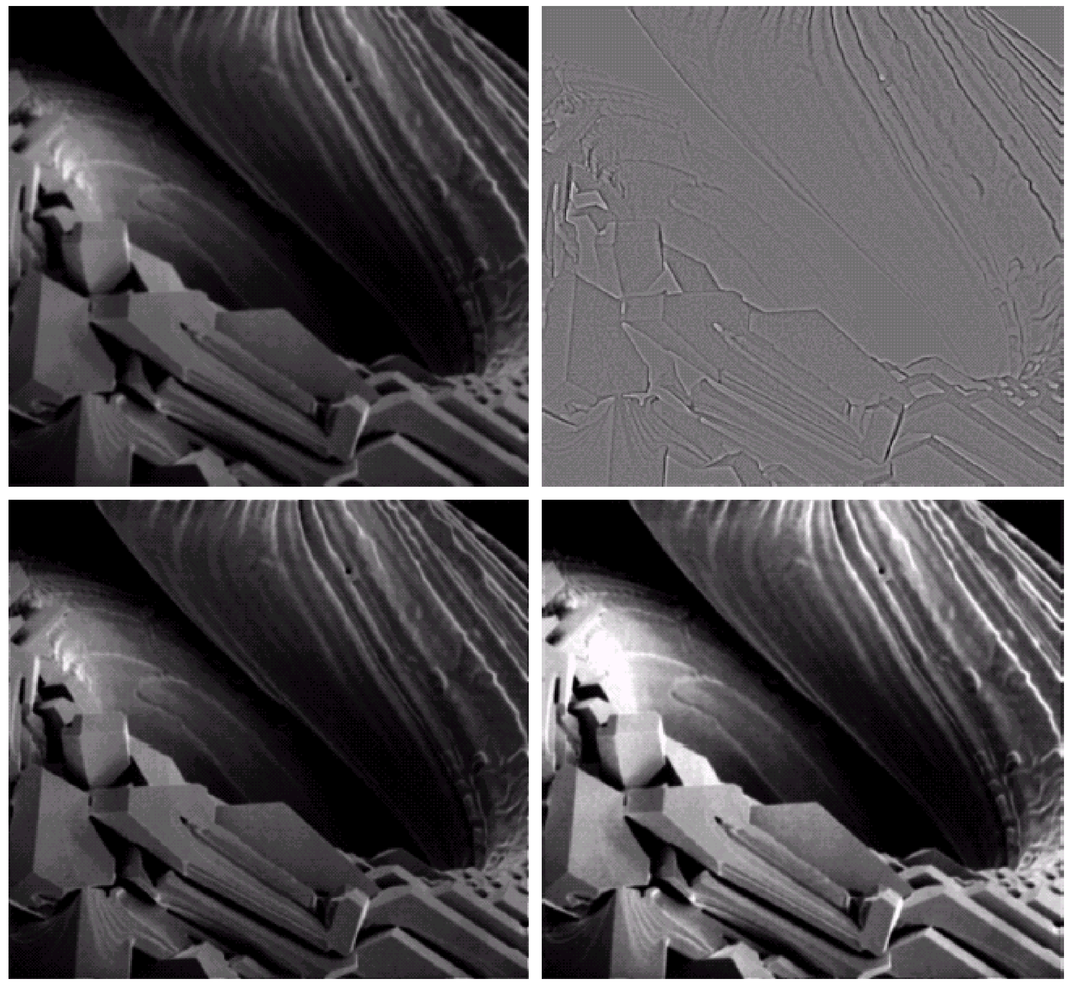


High-Boost Filters

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

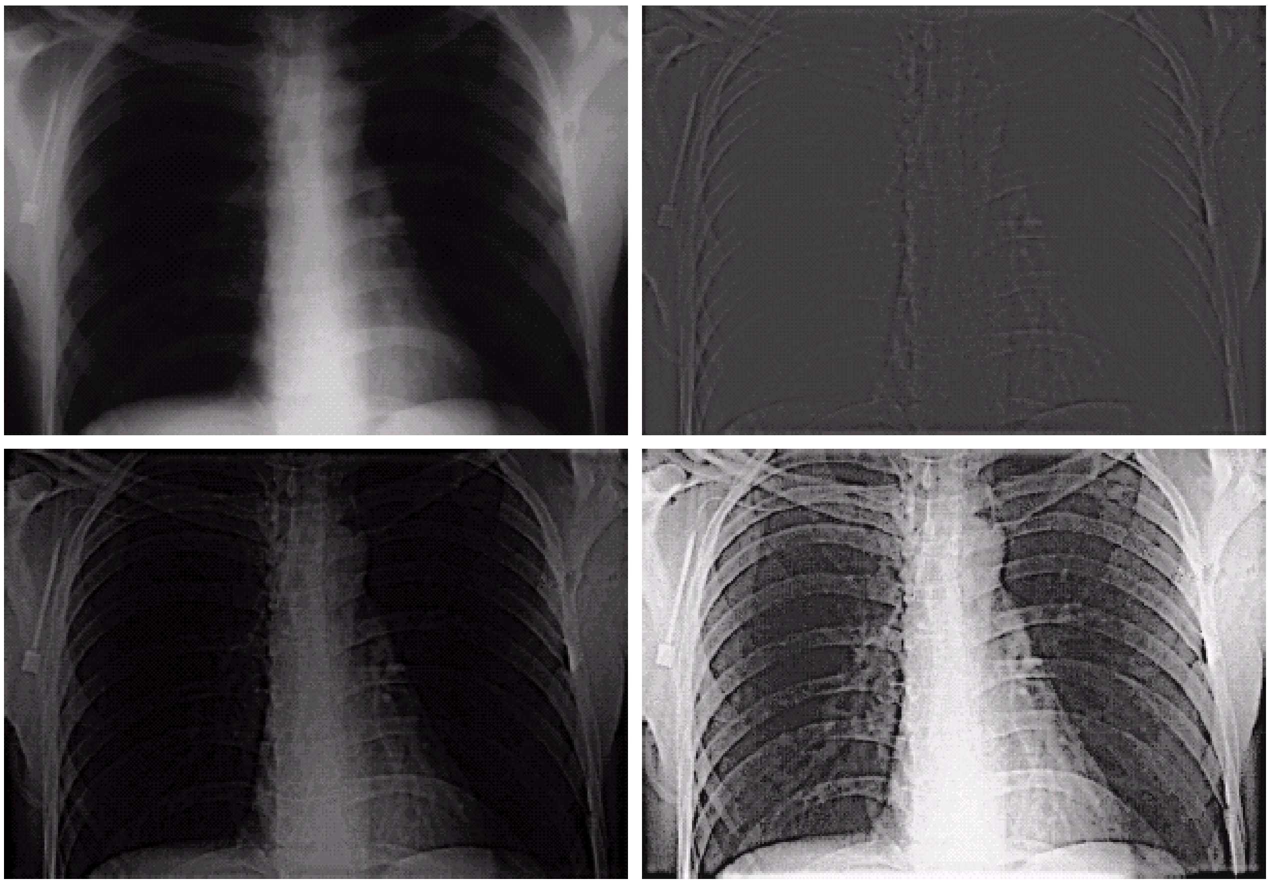
a b
c d

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



High Frequency Emphasis Filters

- $H_{hfe}(u,v) = a + b\,H_{hp}(u,v)$
  - $b > a > 0, \quad 0.25 < a < 0.5, \quad 1.5 < b < 2.0$



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.)