



Chapter-4

Filtering in the Frequency Domain

Jean Baptiste Joseph Fourier

Fourier was born in Auxerre, France in 1768.



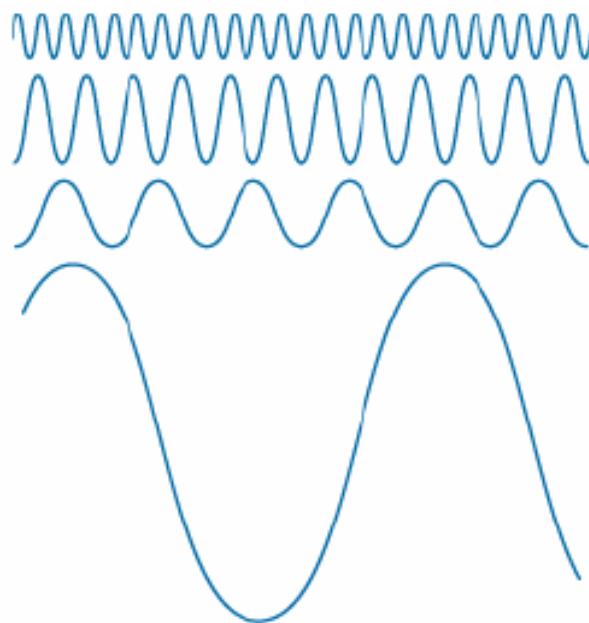
- Most famous of his work: “*La Théorie Analytique de la Chaleur*” published in 1822.
- Translated into English in 1878: “*The Analytic Theory of Heat*”.
- Nobody paid much attention when the work was initially published.
- One of the most important mathematical theories in modern engineering.

Filtering in the Frequency Domain

Filter: A device or material for suppressing or minimizing waves or oscillations of certain frequencies

Frequency: The number of times that a periodic function repeats the same sequence of values during a unit variation of the independent variable

The Big Idea



=

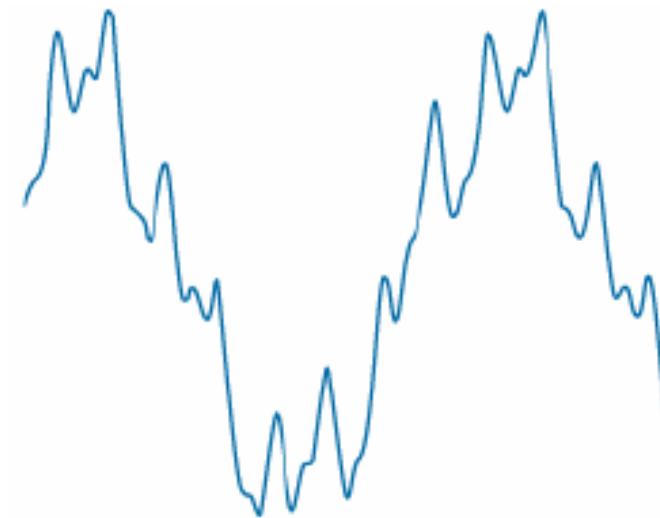
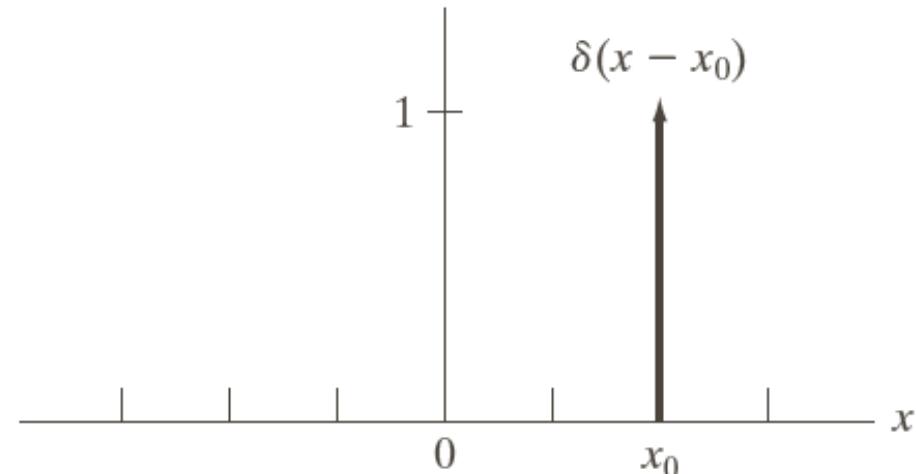


FIGURE 4.1
The function at the right is the sum of the four functions at left. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

Any function that periodically repeats itself can be expressed as a sum of sines and cosines of different frequencies each multiplied by a different coefficient - a *Fourier Series*

1D Continuous Signals

- **Impulse Function:**
 - May be considered both as continuous and discrete.
 - Useful for the representation of discrete signals through sampling of continuous signals.



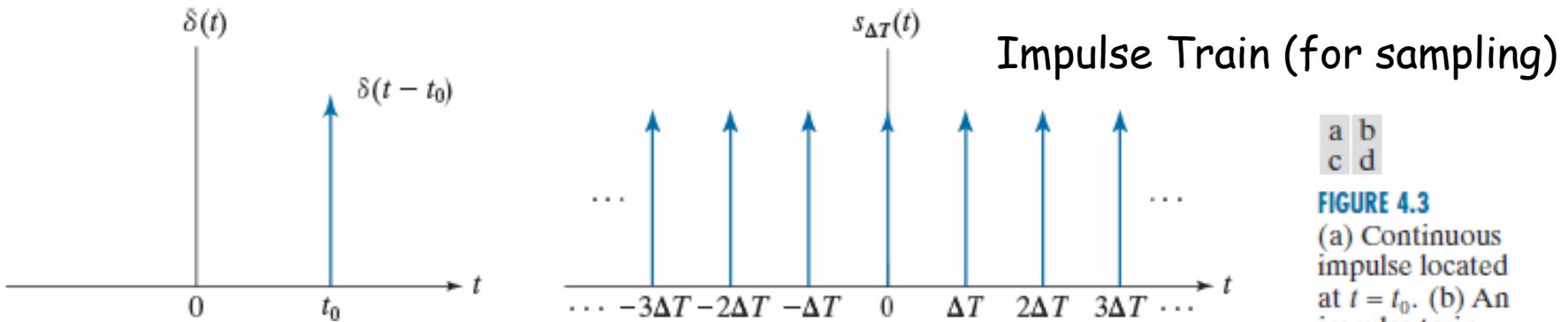
$$\delta(x - x_0) = \begin{cases} +\infty, & x = x_0 \\ 0 & \text{otherwise} \end{cases}$$

Sampling Property of Impulse

→ $\int_{-\infty}^{+\infty} f(x)\delta(x - x_0)dx = f(x_0)$

1D Continuous Delta

- Temporal Delta Function: $\delta(t - t_0) = \begin{cases} +\infty, & t = t_0 \\ 0 & \text{otherwise} \end{cases}$



- Impulse train: $s_{\Delta T}(t) = \sum_{n=-\infty}^{+\infty} \delta(t - n\Delta T)$
- Sampling of $f(t)$:

$$\tilde{f}(t) = f(t)s_{\Delta T}(t) = \sum_{n=-\infty}^{+\infty} f(t)\delta(t - n\Delta T) = \sum_{n=-\infty}^{+\infty} f(n\Delta T)\delta(t - n\Delta T)$$

a	b
c	d

FIGURE 4.3

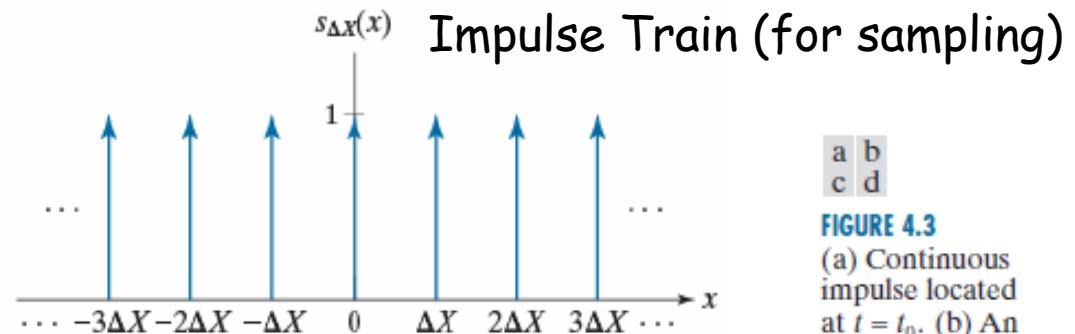
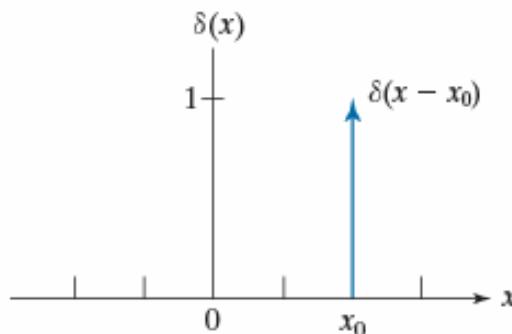
- (a) Continuous impulse located at $t = t_0$. (b) An impulse train consisting of continuous impulses. (c) Unit discrete impulse located at $x = x_0$. (d) An impulse train consisting of discrete unit impulses.



1D Continuous Signals

- Spatial Delta Function

$$\delta(x - x_0) = \begin{cases} +\infty, & x = x_0 \\ 0 & \text{otherwise} \end{cases}$$



a	b
c	d

FIGURE 4.3

- (a) Continuous impulse located at $t = t_0$. (b) An impulse train consisting of continuous impulses. (c) Unit discrete impulse located at $x = x_0$. (d) An impulse train consisting of discrete unit impulses.

Sampling Property of Impulse

$$\int_{-\infty}^{+\infty} f(x) \delta(x - x_0) dx = f(x_0)$$

4.2.2 1D Fourier Series

- The Fourier series expansion of a periodic signal $f(t)$.

$$f(t) = \sum_{n=-\infty}^{+\infty} c_n e^{j \frac{2\pi}{T} nt}$$

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-j \frac{2\pi}{T} nt} dt$$

1D Continuous Signals (cont.)

- Fourier transform of a continuous signal, $f(t)$

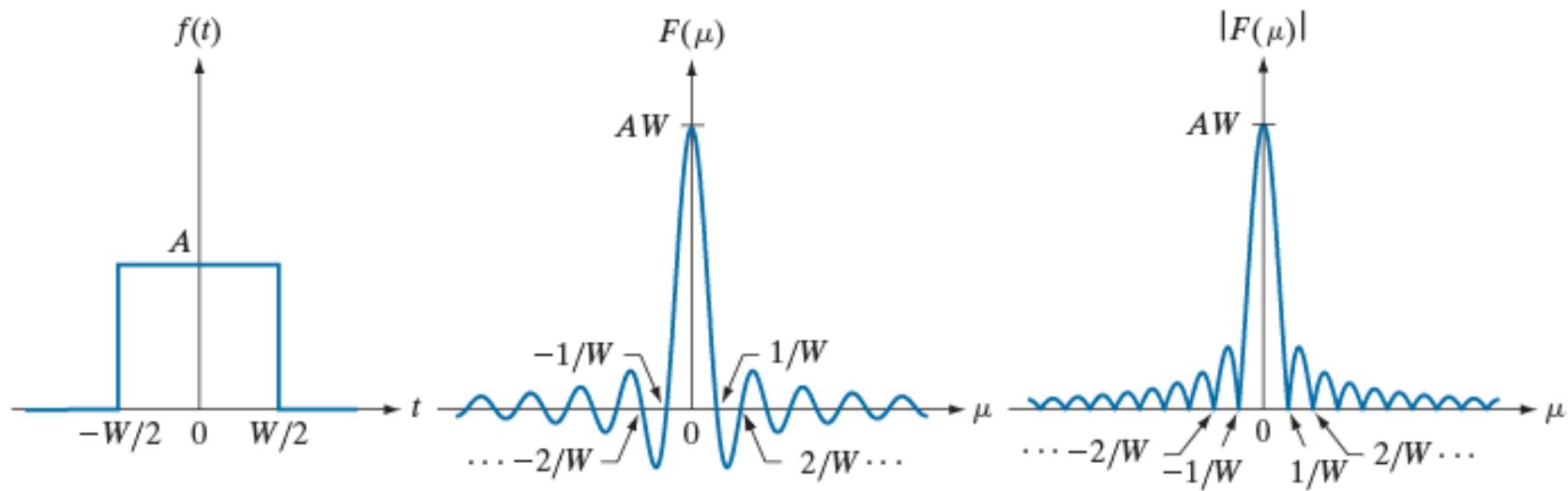
$$F(\mu) = \int_{-\infty}^{+\infty} f(t) e^{-j2\pi\mu t} dt$$

- Inverse Fourier

$$\begin{aligned} f(t) &= \int_{-\infty}^{+\infty} F(\mu) e^{j2\pi\mu t} d\mu \\ &= \int_{-\infty}^{+\infty} F(\mu) [\cos(2\pi\mu t) + j \sin(2\pi\mu t)] d\mu \end{aligned}$$

- Attention: The variable μ is the frequency (Hz) and is related to the radial frequency ($\Omega=2\pi\mu$) as in typical Signals and Systems courses

1D Continuous Signals (cont.)



a b c

FIGURE 4.4 (a) A box function, (b) its Fourier transform, and (c) its spectrum. All functions extend to infinity in both directions. Note the inverse relationship between the width, W , of the function and the zeros of the transform.

$$f(t) = AP_{W/2}(t) \leftrightarrow F(\mu) = AW \frac{\sin(\pi\mu W)}{(\pi\mu W)}$$



Convolution

- Convolution property of FTransform

$$f(t) \star h(t) = \int_{-\infty}^{+\infty} f(\tau)h(t - \tau)d\tau$$

$$f(t) \star h(t) \leftrightarrow F(\mu)H(\mu)$$

$$f(t)h(t) \leftrightarrow F(\mu)\star H(\mu)$$



4.3 Sampling

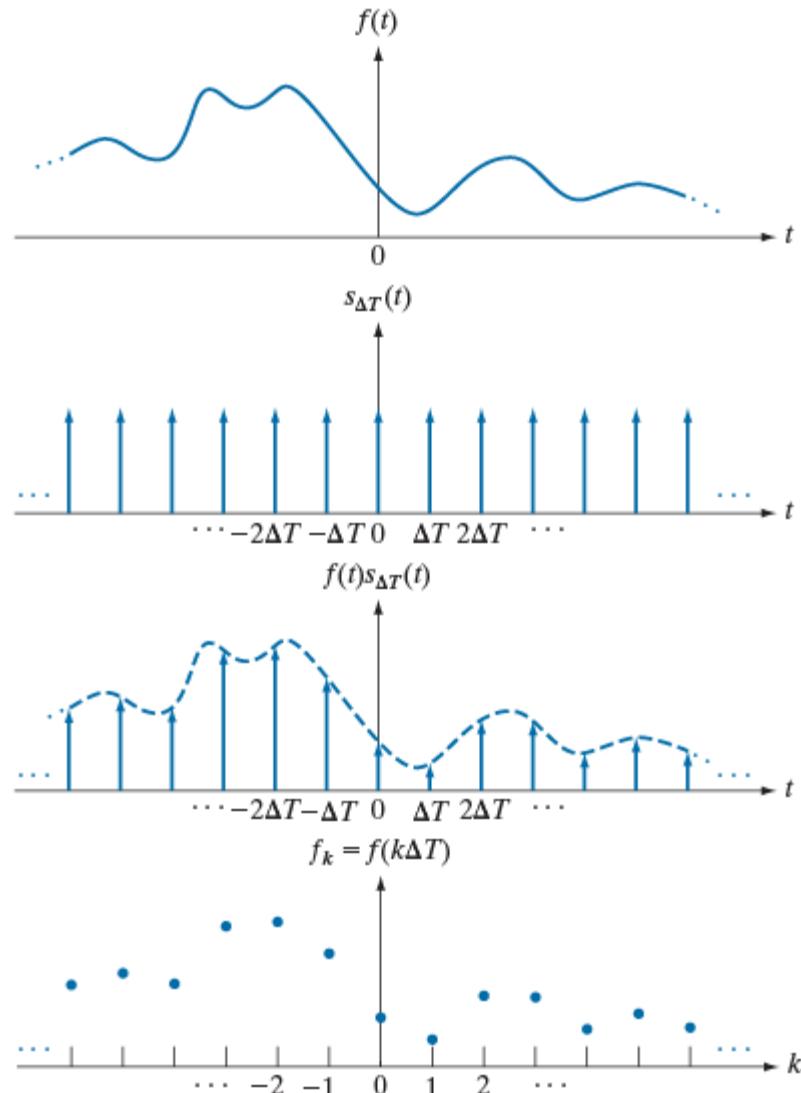
- Sampling of $f(t)$:

$$\tilde{f}(t) = f(t)s_{\Delta T}(t)$$

$$= \sum_{n=-\infty}^{+\infty} f(t)\delta(t - n\Delta T)$$

$$= \sum_{n=-\infty}^{+\infty} f(n\Delta T)\delta(t - n\Delta T)$$

$$f_k = \int_{-\infty}^{\infty} f(t)\delta(t - k\Delta T)dt$$



a
b
c
d

FIGURE 4.5

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

- The Fourier transform of the impulse train

$$s_{\Delta T}(t) \leftrightarrow S(\mu)$$

$$\sum_{n=-\infty}^{+\infty} \delta(t - n\Delta T) \leftrightarrow \frac{1}{\Delta T} \sum_{n=-\infty}^{+\infty} \delta\left(\mu - \frac{n}{\Delta T}\right)$$

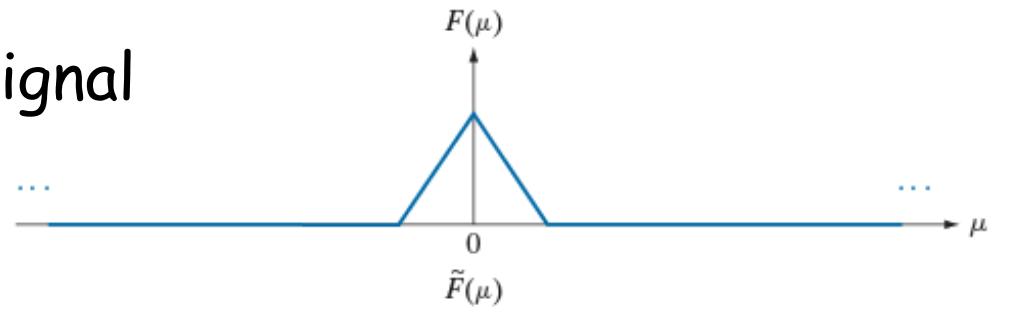
- $S(\mu)$: Also an impulse train in the frequency domain
- Impulses are equally spaced every $1/\Delta T$

- Sampling
 - The spectrum of the discrete signal consists of repetitions of the spectrum of the continuous signal every $1/\Delta T$.
 - Nyquist criterion should be satisfied.

$$f(t) \leftrightarrow F(\mu)$$

$$\tilde{f}(n\Delta T) \leftrightarrow \tilde{F}(\mu) = \frac{1}{\Delta T} \sum_{n=-\infty}^{+\infty} F\left(\mu - \frac{n}{\Delta T}\right)$$

FT of a continuous signal



a
b
c
d

Oversampling

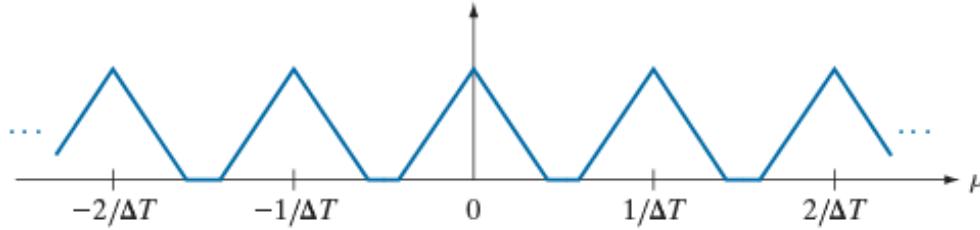
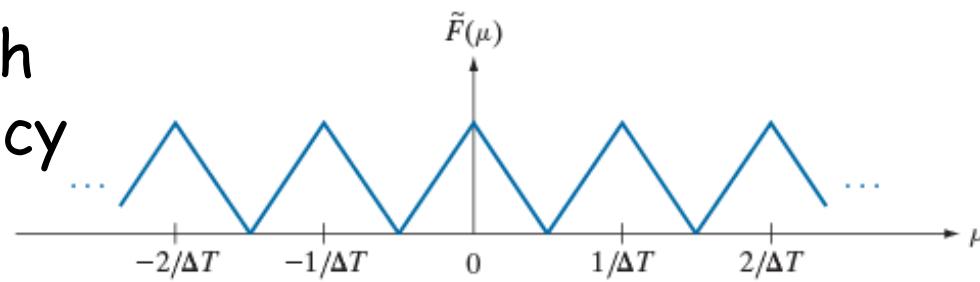


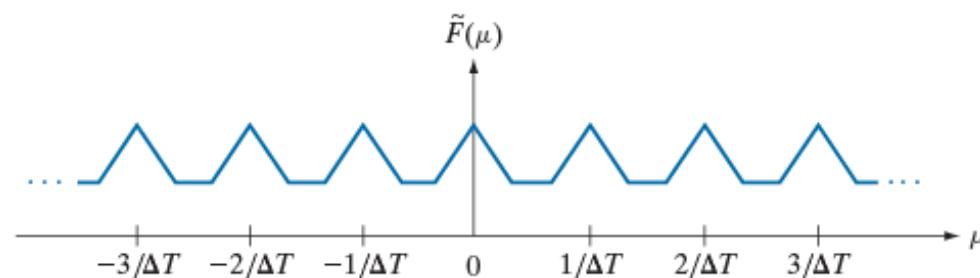
FIGURE 4.6
 (a) Illustrative sketch of the Fourier transform of a band-limited function.

Critical sampling with
the Nyquist frequency



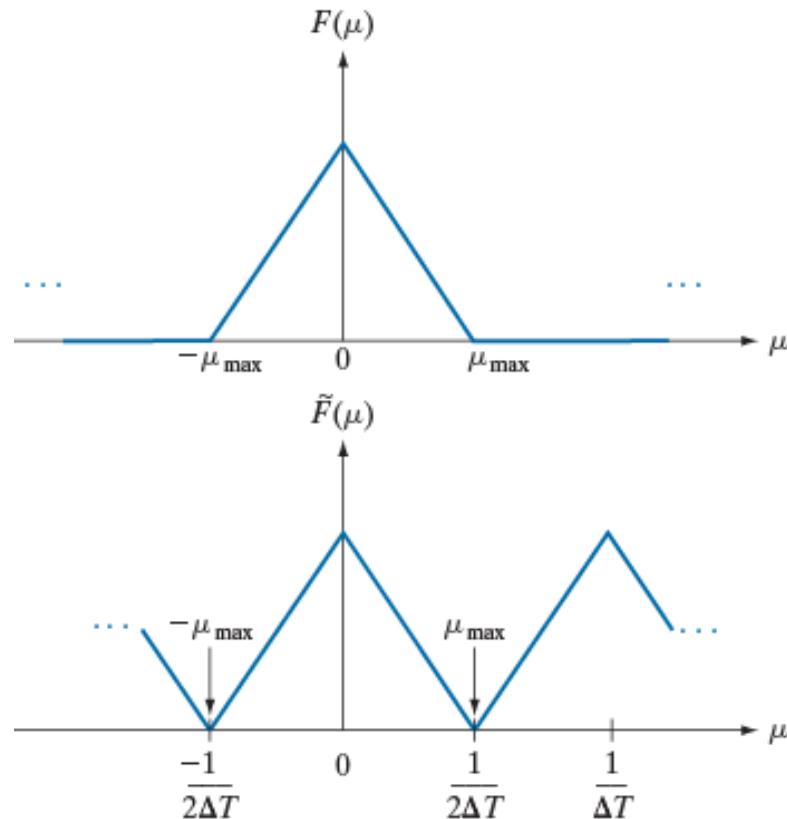
(b)–(d) Transforms of the corresponding sampled functions under the conditions of over-sampling, critically sampling, and under-sampling, respectively.

Undersampling
Aliasing appears



Must satisfy:
Nyquist theorem

$$\mu_s = \frac{1}{\Delta T} \geq 2\mu_{\max}$$



a
b

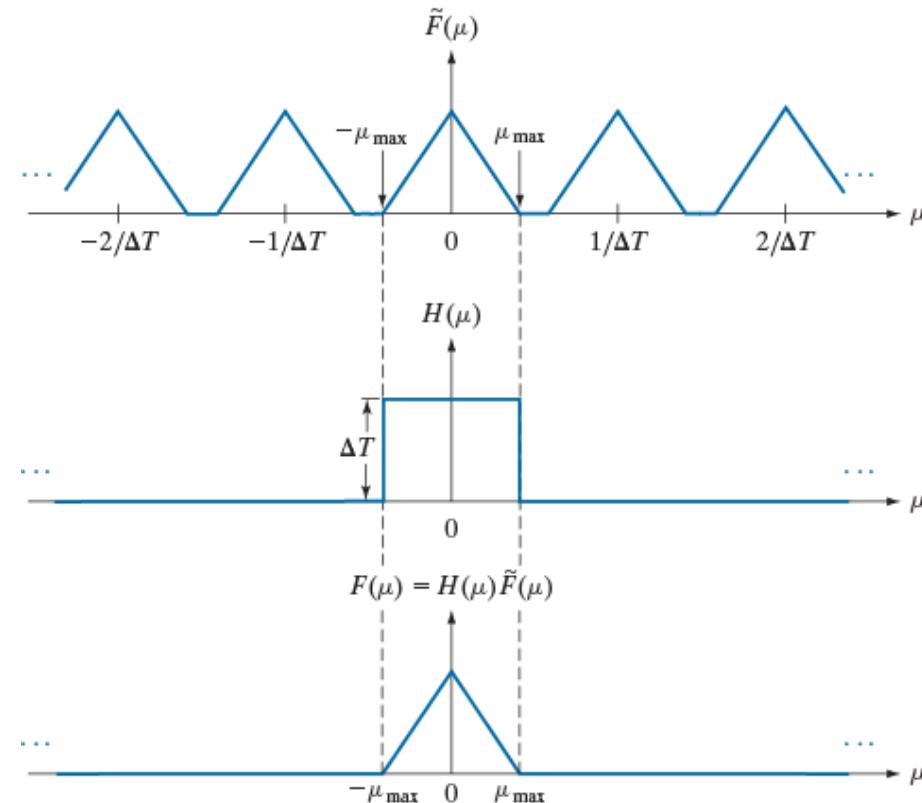
FIGURE 4.7
 (a) Illustrative sketch of the Fourier transform of a band-limited function.
 (b) Transform resulting from critically sampling that band-limited function.

- Reconstruction
 - Provided correctly sampled, the original continuous signal can be perfectly reconstructed from the samples

$$f(t) = \sum_{n=-\infty}^{+\infty} f(n\Delta T) \operatorname{sinc}\left[\frac{(t - n\Delta T)}{\Delta T}\right]$$

1D Continuous Signals: Reconstruction

- Reconstruction for Nyquist sampling:



a
b
c

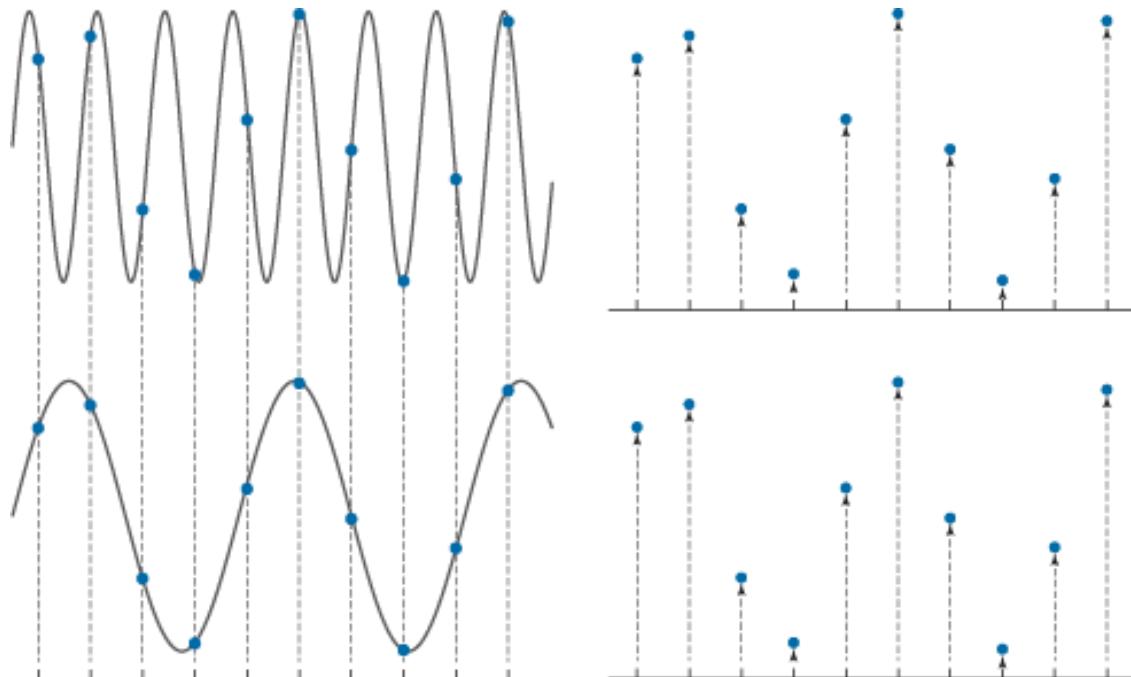
FIGURE 4.8
(a) Fourier transform of a sampled, band-limited function.
(b) Ideal lowpass filter transfer function.
(c) The product of (b) and (a), used to extract one period of the infinitely periodic sequence in (a).

$$F(\mu) = \tilde{F}(\mu)H(\mu)$$

$$f(t) = \tilde{f}(t) * T \operatorname{sinc}\left(\frac{t}{\Delta T}\right) = \sum_{n=-\infty}^{+\infty} f(n\Delta T) \operatorname{sinc}\left[\frac{(t-n\Delta T)}{\Delta T}\right]$$

4.3.4 Undersampling & Aliasing

- Undersampling causes aliasing
- If aliased, the reconstruction of the continuous signal is not correct



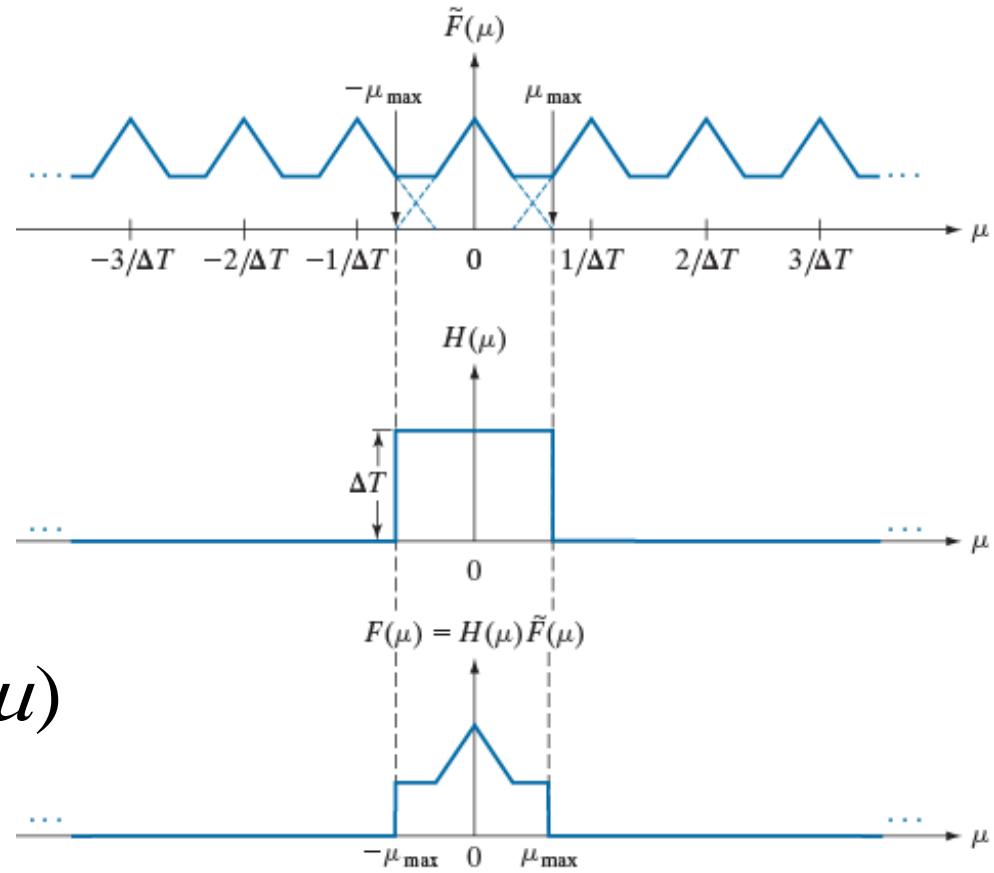
a b
c d

FIGURE 4.9
The functions in (a) and (c) are totally different, but their digitized versions in (b) and (d) are identical. Aliasing occurs when the samples of two or more functions coincide, but the functions are different elsewhere.

4.3.4 Undersampling & Aliasing

- If aliased, the reconstruction of the continuous signal is not correct

$$F(\mu) = \tilde{F}(\mu)H(\mu)$$



a
b
c

FIGURE 4.10 (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 $f(t)$.



Aliased signal

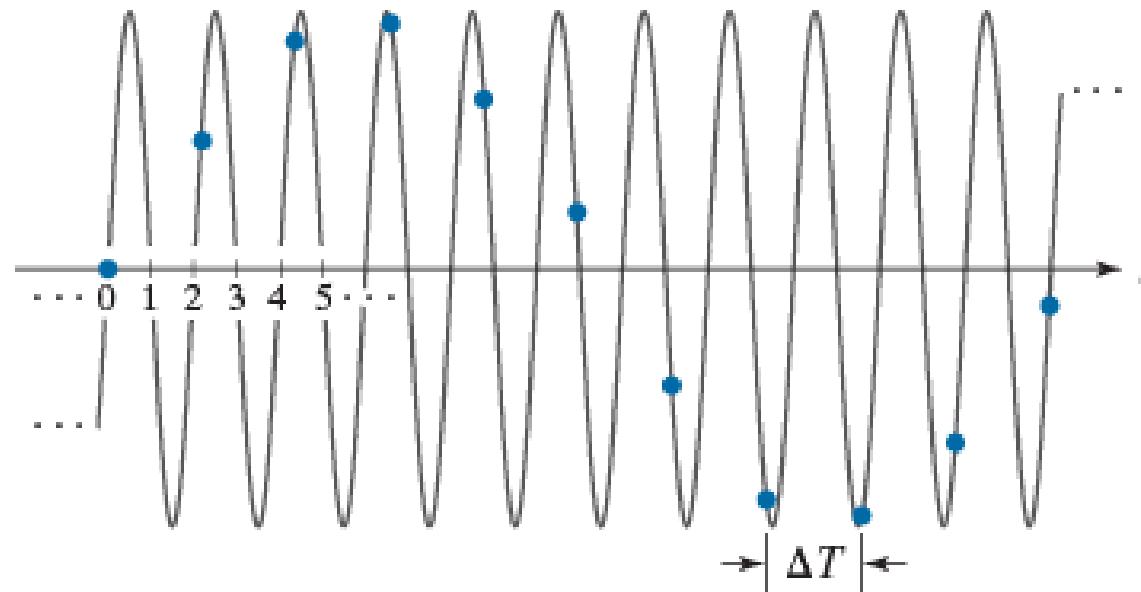


FIGURE 4.11 Illustration of aliasing. The under-sampled function (dots) looks like a sine wave having a frequency much lower than the frequency of the continuous signal. The period of the sine wave is 2 s, so the zero crossings of the horizontal axis occur every second. ΔT is the separation between samples.

4.4 The Discrete Fourier Transform

- The Fourier transform of a sampled (discrete) signal is a continuous function of the frequency.

$$\tilde{F}(\mu) = \frac{1}{\Delta T} \sum_{n=-\infty}^{+\infty} F\left(\mu - \frac{n}{\Delta T}\right)$$

- For a N -length discrete signal, taking N samples of its Fourier transform at frequencies:

$$\mu_k = \frac{k}{N\Delta T} = k \frac{\mu_s}{N} = k\Delta\mu, \quad k = 0, 1, \dots, N-1$$

provides the discrete Fourier transform (DFT) of the signal.

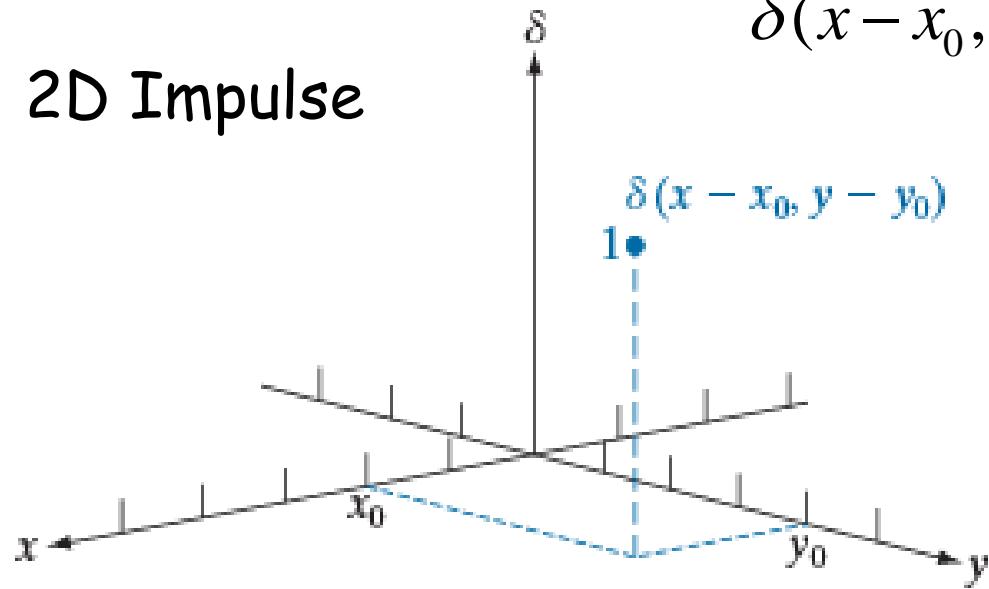
- DFT pair of signal $f(x)$ of length M :

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-j \frac{2\pi u x}{M}}, \quad 0 \leq u \leq M-1$$

$$f(x) = \frac{1}{M} \sum_{u=0}^{M-1} F(u) e^{j \frac{2\pi u x}{M}}, \quad 0 \leq x \leq M-1$$

4.5 2D Continuous Signals

2D Impulse



$$\delta(x - x_0, y - y_0) = \begin{cases} +\infty, & x = x_0, y = y_0 \\ 0 & \text{otherwise} \end{cases}$$

FIGURE 4.13
2-D unit discrete impulse. Variables x and y are discrete, and δ is zero everywhere except at coordinates (x_0, y_0) , where its value is 1.

Sampling Property:

$$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) \delta(x - x_0, y - y_0) dy dx = f(x_0, y_0)$$

Separable: $\delta(x - x_0, y - y_0) = \delta(x - x_0) \delta(y - y_0)$

- The Fourier transform of a continuous 2D signal $f(t, z)$.

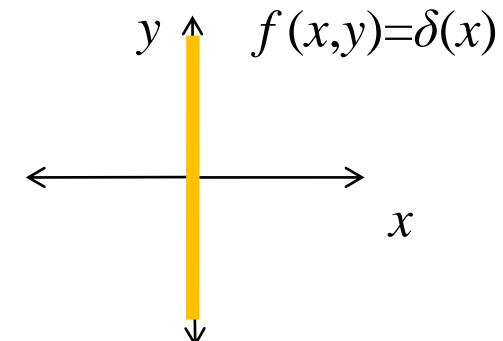
$$F(\mu, \nu) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(t, z) e^{-j2\pi(\mu t + \nu z)} dt dz$$

$$f(t, z) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F(\mu, \nu) e^{j2\pi(\mu t + \nu z)} d\nu d\mu$$

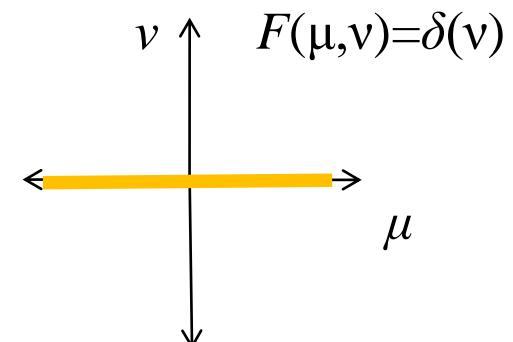
2D Continuous Signals (cont.)

- Example: FT of $f(x, y) = \delta(x)$

$$F(\mu, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \delta(x) e^{-j2\pi(\mu x + vy)} dy dx$$



$$= \int_{-\infty}^{+\infty} \delta(x) e^{-j2\pi\mu x} dx \int_{-\infty}^{+\infty} e^{-j2\pi vy} dy$$



$$= \int_{-\infty}^{+\infty} e^{-j2\pi vy} dy = \delta(v)$$

Fourier of a constant (or DC) signal is a delta

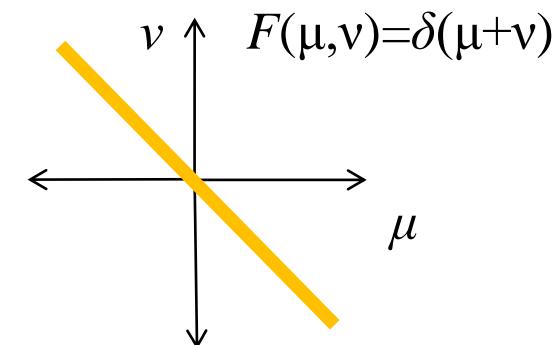
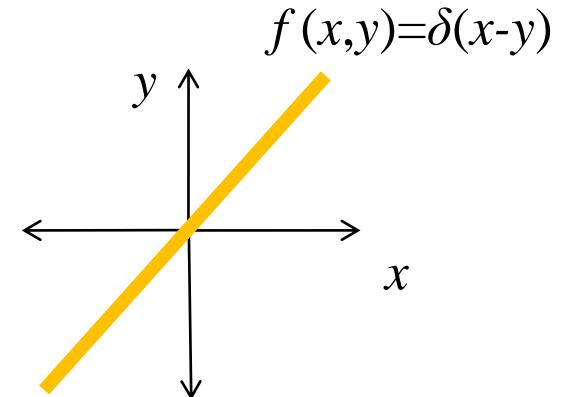
- Example: FT of $f(x, y) = \delta(x-y)$

$$F(\mu, \nu) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \delta(x-y) e^{-j2\pi(\mu x + \nu y)} dy dx$$

$$= \int_{-\infty}^{+\infty} \left[\int_{-\infty}^{+\infty} \delta(x-y) e^{-j2\pi\mu x} dx \right] e^{-j2\pi\nu y} dy$$

$$= \int_{-\infty}^{+\infty} e^{-j2\pi\mu y} e^{-j2\pi\nu y} dy = \int_{-\infty}^{+\infty} e^{-j2\pi(\mu+\nu)y} dy$$

$$= \delta(\mu + \nu)$$

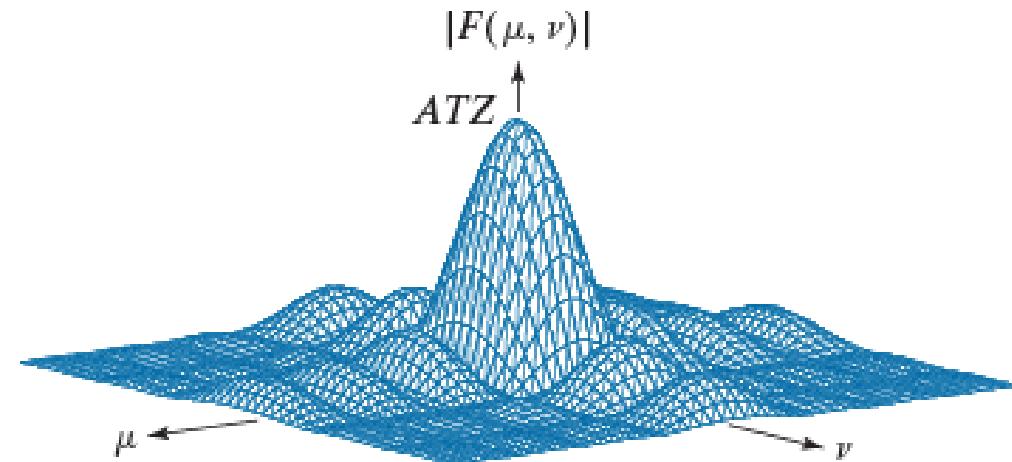
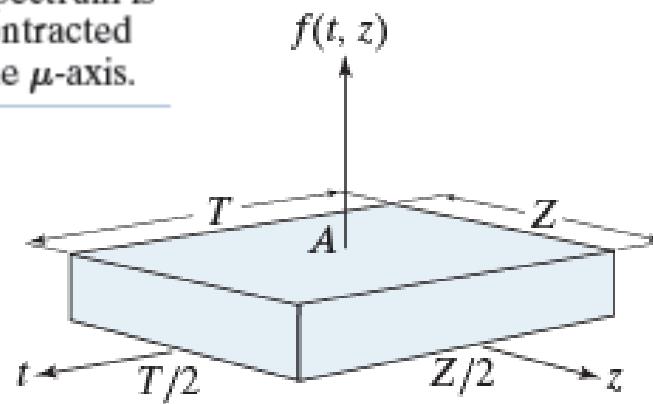


2D Continuous Signals (cont.)

a b

FIGURE 4.14

(a) A 2-D function and (b) a section of its spectrum.
 The box is longer along the t -axis, so the spectrum is more contracted along the μ -axis.



$$f(t, z) = A P_{T/2, Z/2}(t, z) \Leftrightarrow F(\mu, \nu) = ATZ \frac{\sin(\pi\mu T)}{(\pi\mu T)} \frac{\sin(\pi\nu Z)}{(\pi\nu Z)}$$

- 2D continuous convolution

$$f(x, y) \star h(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x - \alpha, y - \beta) h(\alpha, \beta) d\alpha d\beta$$

- We will examine the discrete convolution in more detail.
- Convolution property

$$f(x, y) \star h(x, y) \leftrightarrow F(\mu, \nu) H(\mu, \nu)$$

2D Continuous Signals (cont.)

- 2D sampling is accomplished by

$$s_{\Delta X \Delta Y}(x, y) = \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} \delta(x - n\Delta X, y - m\Delta Y)$$

- The FT of the sampled 2D signal consists of repetitions of the spectrum of the 1D continuous signal.

$$\tilde{F}(\mu, \nu) = \frac{1}{\Delta X} \frac{1}{\Delta Y} \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} F\left(\mu - \frac{m}{\Delta X}, \nu - \frac{n}{\Delta Y}\right)$$

2D Continuous Signals (cont.)

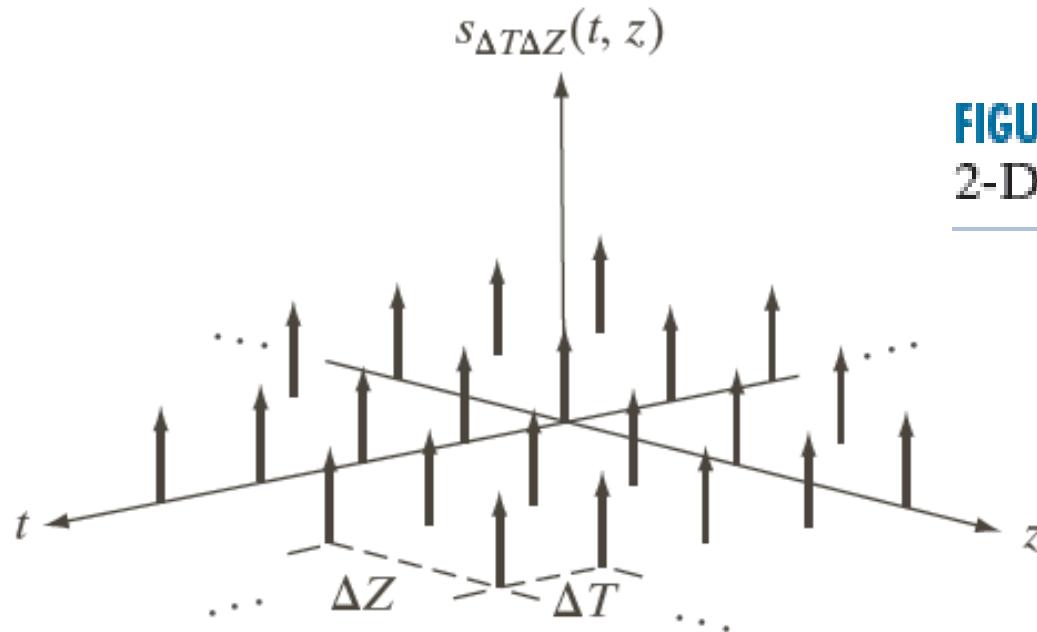


FIGURE 4.15
2-D impulse train.

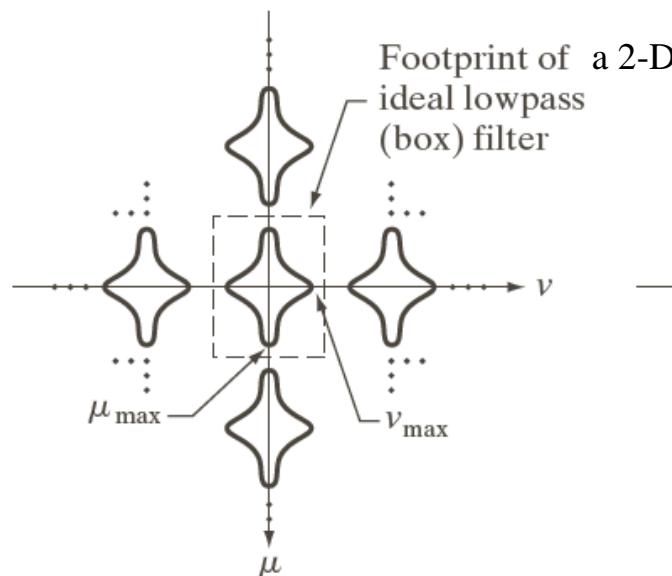
- The 2D impulse train is also separable:

$$s_{\Delta T \Delta Z}(t, z) = s_{\Delta T}(t)s_{\Delta Z}(z) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \delta(t - m\Delta T, z - n\Delta Z)$$

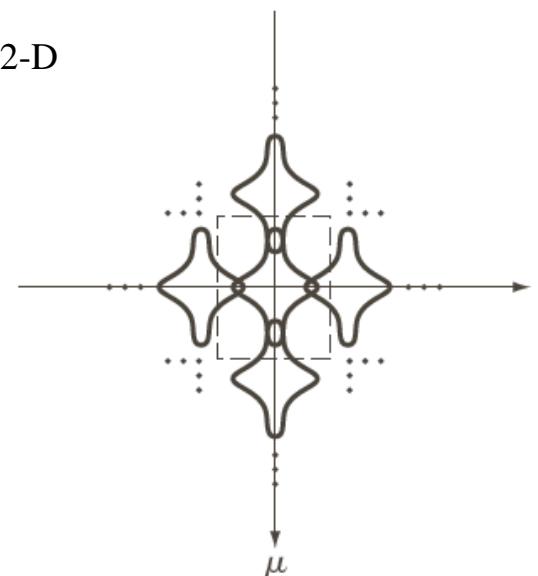
4.5.3 2-D Sampling and 2-D Sampling Theorem

- The Nyquist theorem involves both the horizontal and vertical frequencies.

$$\frac{1}{\Delta X} \geq 2\mu_{\max}, \quad \frac{1}{\Delta Y} \geq 2\nu_{\max}$$



Over-sampled



Under-sampled

a b

FIGURE 4.16
Two-dimensional Fourier transforms of (a) an over-sampled, and (b) an under-sampled, band-limited function.

2-D Sampling and Aliasing Effects

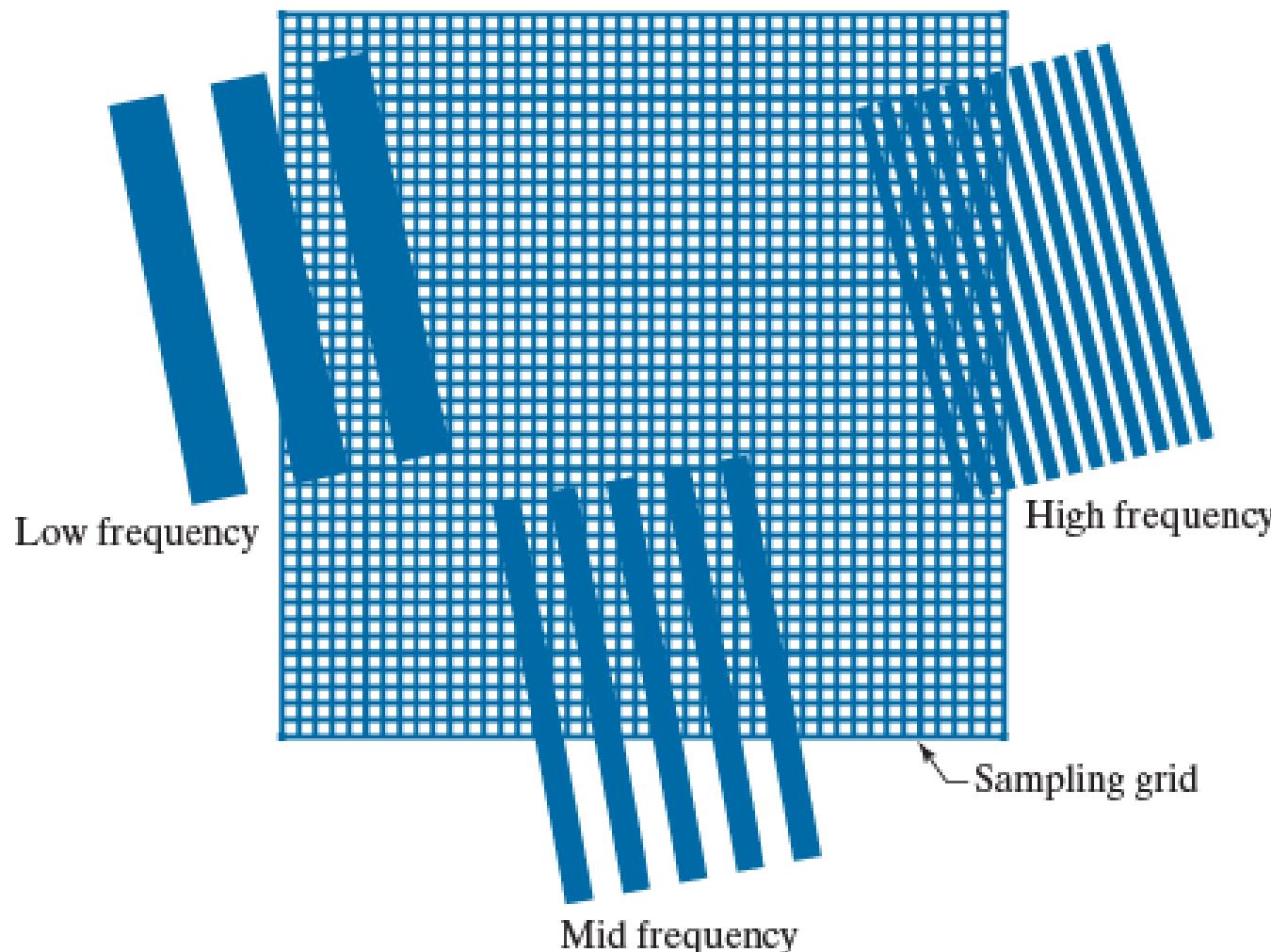


FIGURE 4.17
Various aliasing effects resulting from the interaction between the frequency of 2-D signals and the sampling rate used to digitize them. The regions outside the sampling grid are continuous and free of aliasing.

Aliasing in Images: Example

In an image system, the number of samples is fixed at 96x96 pixels. If we use this system to digitize checkerboard patterns ...

a b
c d

FIGURE 4.18

Aliasing. In (a) and (b) the squares are of sizes 16 and 6 pixels on the side. In (c) and (d) the squares are of sizes 0.95 and 0.48 pixels, respectively. Each small square in (c) is one pixel. Both (c) and (d) are aliased. Note how (d) masquerades as a “normal” image.

Under-sampling

Aliasing in Images: Example

- 1x1 pixel sampling
- Number of samples is fixed at 96x96 pixels

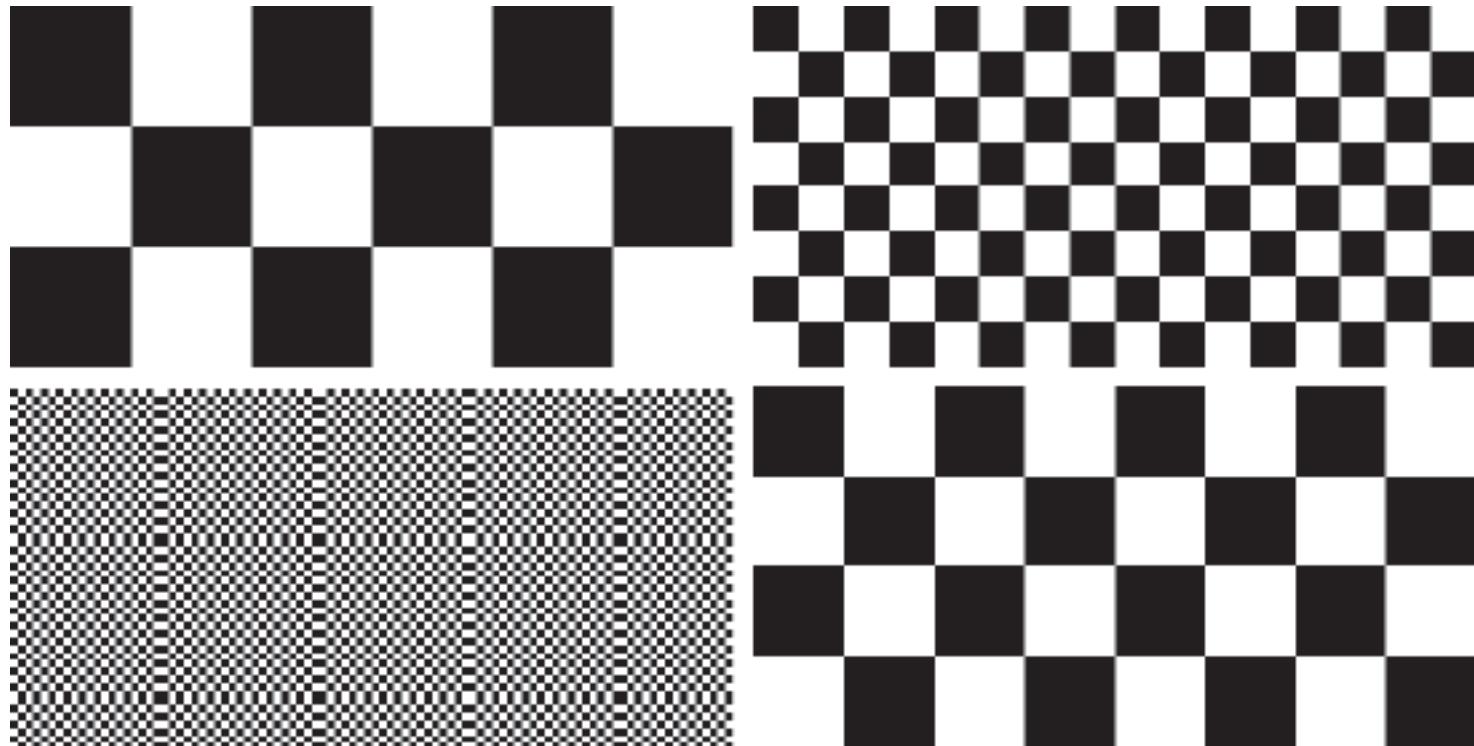


FIGURE 4.18

Aliasing. In (a) and (b) the squares are of sizes 16 and 6 pixels on the side. In (c) and (d) the squares are of sizes 0.95 and 0.48 pixels, respectively. Each small square in (c) is one pixel. Both (c) and (d) are aliased. Note how (d) masquerades as a “normal” image.

Under-sampling



Aliasing in Images: Example



Anti-aliasing Filter + Re-sampling

a b c

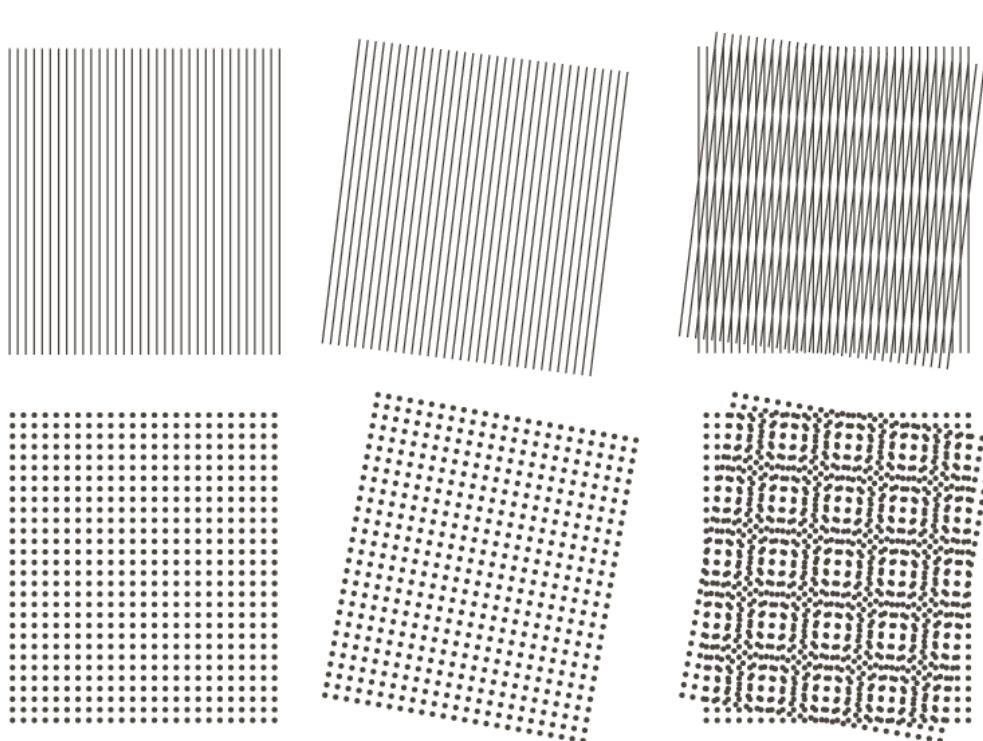
FIGURE 4.19 Illustration of aliasing on resampled natural images. (a) A digital image of size 772×548 pixels with visually negligible aliasing. (b) Result of resizing the image to 33% of its original size by pixel deletion and then restoring it to its original size by pixel replication. Aliasing is clearly visible. (c) Result of blurring the image in (a) with an 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.)

Aliasing - Moiré Patterns

- Undesired artifact of images produced by various digital imaging and computer graphics techniques
 - Effect of sampling a scene with periodic or nearly periodic components (e.g. overlapping grids, TV raster lines and striped materials in the scene).
- In image processing the problem arises when scanning media prints (e.g. magazines, newspapers).
- The problem is more general than sampling artifacts.
- http://en.wikipedia.org/wiki/Moiré_pattern

Aliasing - Moiré Patterns (cont.)

- Superimposed ink drawings (not digitized)
- Produces effect of new frequencies not existing in the original components.



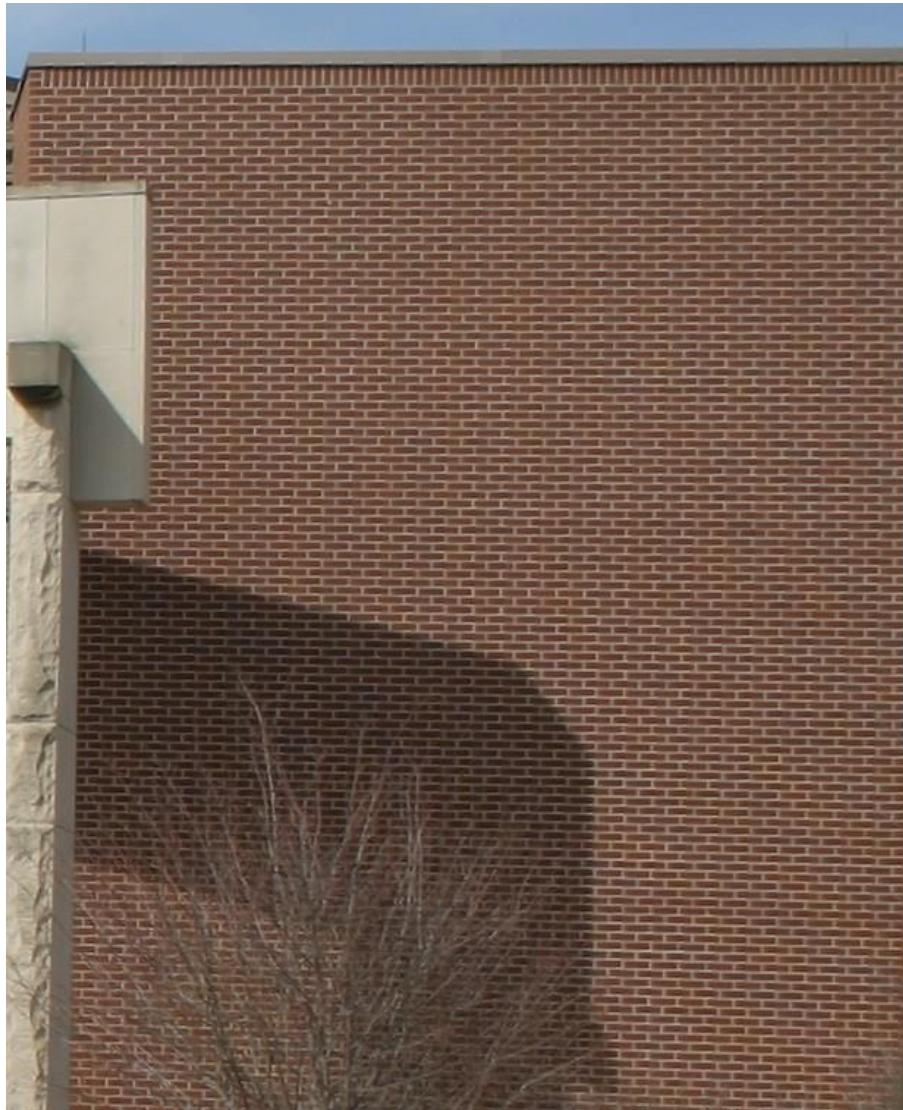
a	b	c
d	e	f

FIGURE 4.20
Examples of the moiré effect.
These are vector drawings, not digitized patterns.
Superimposing one pattern on the other is analogous to multiplying the patterns.





Moiré Patterns



Original Image



Moiré Patterns



A moiré pattern formed by incorrectly down-sampling the former image

Aliasing - Moiré Patterns (cont.)

- In printing industry the problem occurs when scanning photographs from the superposition of:
 - The sampling lattice (usually horizontal and vertical).
 - Dot patterns on the newspaper image.

Aliasing - Moiré Patterns (cont.)



FIGURE 4.21
A newspaper image digitized at 75 dpi. Note the moiré-like pattern resulting from the interaction between the $\pm 45^\circ$ orientation of the half-tone dots and the north-south orientation of the sampling elements used to digitized the image.

4.5.5 The 2D DFT

- 2D DFT pair of image $f(x, y)$ of size $M \times N$.

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

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

$$\begin{cases} 0 \leq u \leq M - 1 \\ 0 \leq v \leq N - 1 \end{cases}, \quad \begin{cases} 0 \leq x \leq M - 1 \\ 0 \leq y \leq N - 1 \end{cases}$$



Let ΔT and ΔZ denote the separations between spatial samples, then the separations between the corresponding discrete, frequency domain variables are given by

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

and $\Delta v = \frac{1}{N \Delta Z}$

- **Translation**

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

and

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

- **Rotation**

Fourier Pair Using polar coordinates: $f(r, \theta) \Leftrightarrow F(\omega, \varphi)$

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

results in the following transform pair:

$\Rightarrow f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$: Same Rotation in both domains

Properties of the 2-D DFT: Periodicity

2 – D Fourier transform and its inverse are infinitely periodic

$$F(u, v) = F(u + k_1 M, v) = F(u, v + k_2 N) = F(u + k_1 M, v + k_2 N)$$

$$f(x, y) = f(x + k_1 M, y) = f(x, y + k_2 N) = f(x + k_1 M, y + k_2 N)$$

$$f(x)e^{j2\pi(u_0x/M)} \Leftrightarrow F(u - u_0)$$

$$\text{For } u_0 = M / 2, \quad f(x)(-1)^x \Leftrightarrow F(u - M / 2)$$

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

→ Used for Centering Fourier Transform of an Image

Periodicity of the DFT (cont...)

- For display and computation purposes it is convenient to shift the DFT and have a complete period in $[0, M-1]$.
- From DFT properties:

$$f(x)e^{j2\pi(u_0x/M)} \Leftrightarrow F(u - u_0)$$

Letting $u_0=M/2$: $f(x)(-1)^n \Leftrightarrow F(u - M/2)$

- $F(0)$ is now located at $M/2$.

Periodicity of the DFT (cont...)

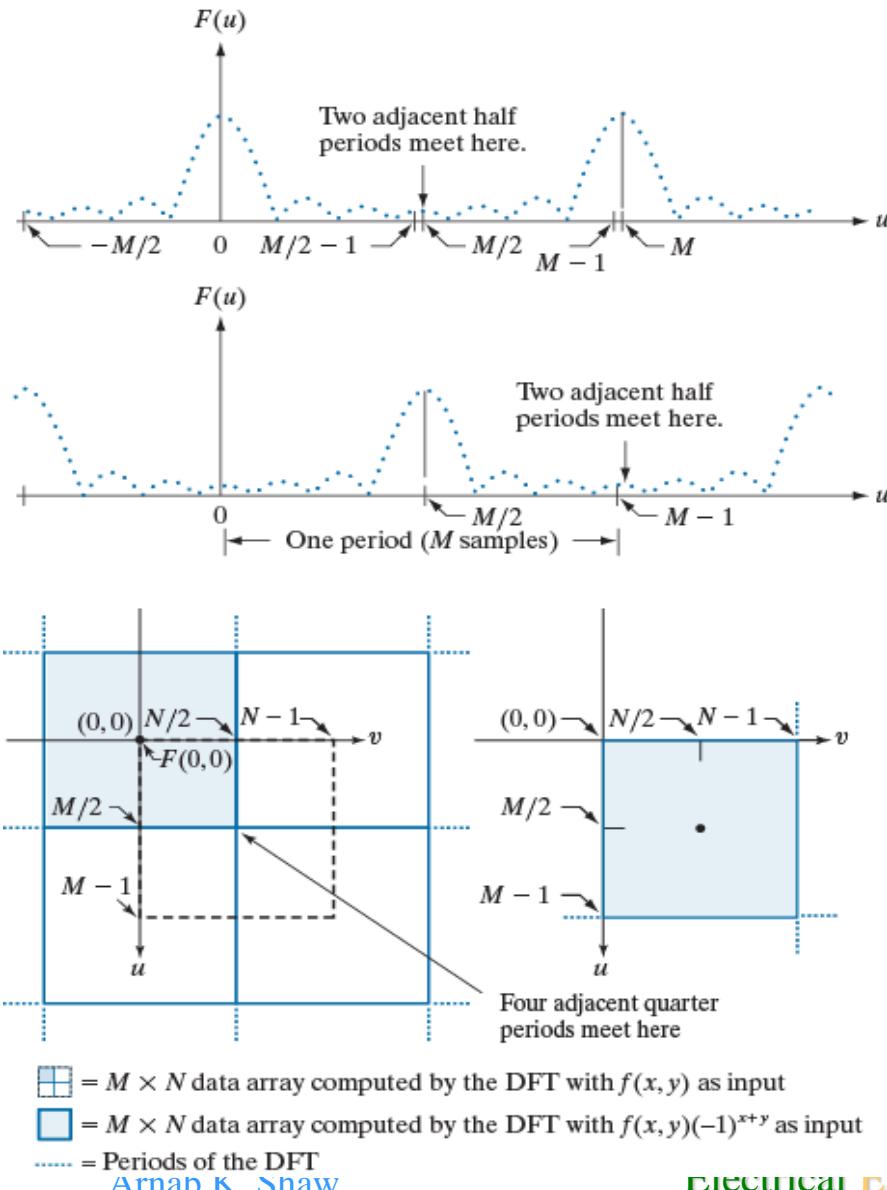
- In two dimensions:

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

and $F(0,0)$ is now located at $(M/2, N/2)$.

Periodicity Properties of the $(e^{-j\frac{2\pi}{M}ux})^N$ DFT

- The range of frequencies of the signal is between $[-M/2, M/2]$.
- The DFT covers two back-to-back half periods of the signal as it covers $[0, M-1]$.





Properties of the 2-D DFT Symmetry

TABLE 4.1

Some symmetry properties of the 2-D DFT and its inverse. $R(u,v)$ and $I(u,v)$ are the real and imaginary parts of $F(u,v)$, respectively. Use of the word *complex* indicates that a function has nonzero real and imaginary parts.

	Spatial Domain [†]	Frequency Domain [†]
1)	$f(x,y)$ real	$\Leftrightarrow F^*(u,v) = F(-u,-v)$
2)	$f(x,y)$ imaginary	$\Leftrightarrow F^*(-u,-v) = -F(u,v)$
3)	$f(x,y)$ real	$\Leftrightarrow R(u,v)$ even; $I(u,v)$ odd
4)	$f(x,y)$ imaginary	$\Leftrightarrow R(u,v)$ odd; $I(u,v)$ even
5)	$f(-x,-y)$ real	$\Leftrightarrow F^*(u,v)$ complex
6)	$f(-x,-y)$ complex	$\Leftrightarrow F(-u,-v)$ complex
7)	$f^*(x,y)$ complex	$\Leftrightarrow F^*(-u,-v)$ complex
8)	$f(x,y)$ real and even	$\Leftrightarrow F(u,v)$ real and even
9)	$f(x,y)$ real and odd	$\Leftrightarrow F(u,v)$ imaginary and odd
10)	$f(x,y)$ imaginary and even	$\Leftrightarrow F(u,v)$ imaginary and even
11)	$f(x,y)$ imaginary and odd	$\Leftrightarrow F(u,v)$ real and odd
12)	$f(x,y)$ complex and even	$\Leftrightarrow F(u,v)$ complex and even
13)	$f(x,y)$ complex and odd	$\Leftrightarrow F(u,v)$ complex and odd

[†]Recall that x , y , u , and v are *discrete* (integer) variables, with x and u in the range $[0, M-1]$, and y and v in the range $[0, N-1]$. To say that a complex function is *even* means that its real *and* imaginary parts are even, and similarly for an *odd* complex function. As before, “ \Leftrightarrow ” indicates a Fourier transform pair.

2-D DFT in polar form

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

Fourier spectrum (*Magnitude*)

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

Power spectrum

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

Phase angle

$$\phi(u, v) = \arctan \left[\frac{I(u, v)}{R(u, v)} \right]$$

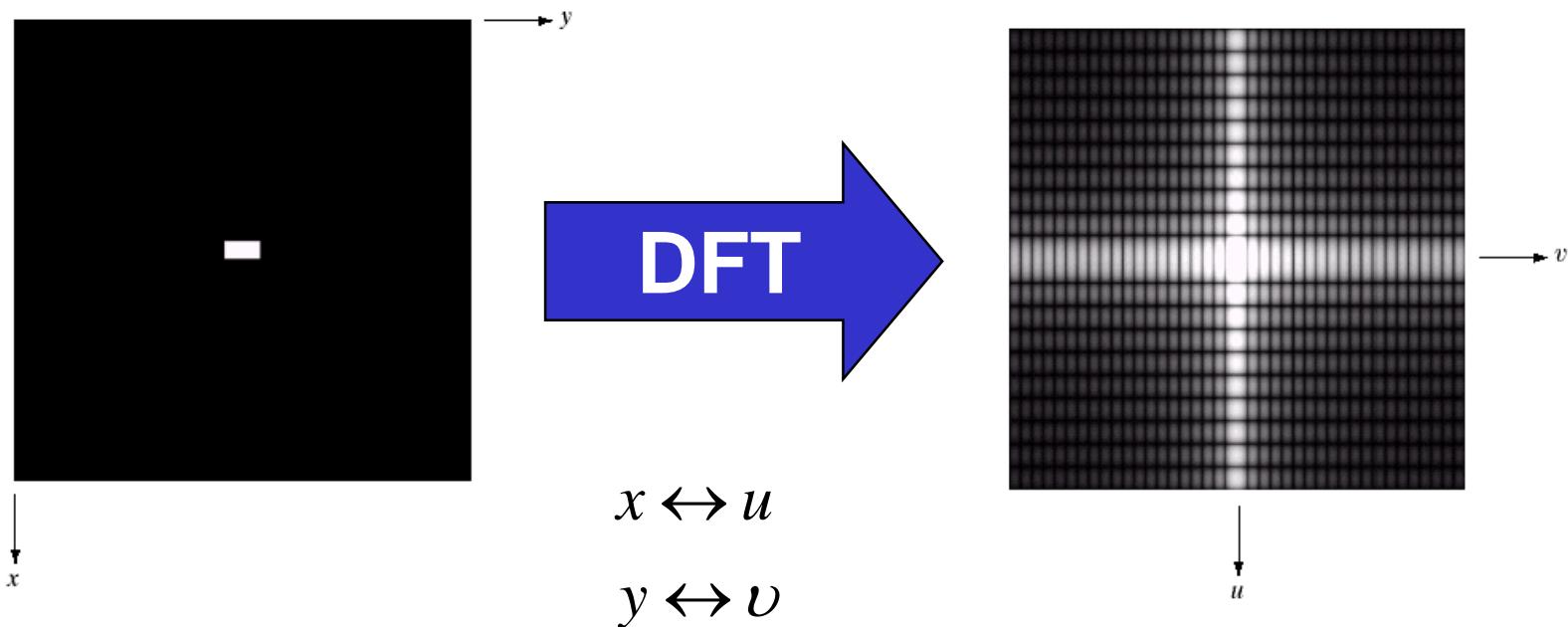
Basics of Filtering in the Frequency Domain

- Each term of $F(u, v)$ contains all values of $f(x, y)$
- Usually, it is impossible to make direct associations between specific components of an image and its transform
- Some general statements can be made about the relationship between the frequency components of the FT and spatial features of an image
 - Associate frequencies in the FT with patterns of intensity variations in an image

- **Slowest varying frequency component ($u = v = 0$) is proportional to the average intensity of image**
- As we move away from the origin of the transform, the low frequencies correspond to the slowly varying intensity components of an image
 - e.g., Smooth intensity variations on walls and floor in an image of room or a cloudless sky in an outdoor scene
- As we move further away from the origin, the higher frequencies begin to correspond to faster and sharper intensity changes in the image
 - e.g. Sharp change in intensity such as edges of objects and other components of an image and noise

- Filtering techniques in the frequency domain are based on modifying the FT to achieve a specific objective and then computing the inverse DFT to get back to the image domain
- Two components of the FT to process
 - Transform magnitude (Spectrum): Useful
 - Phase angle: Useful in some cases

The DFT of a two dimensional image can be visualised by showing the spectrum of the image component frequencies





DFT & Images

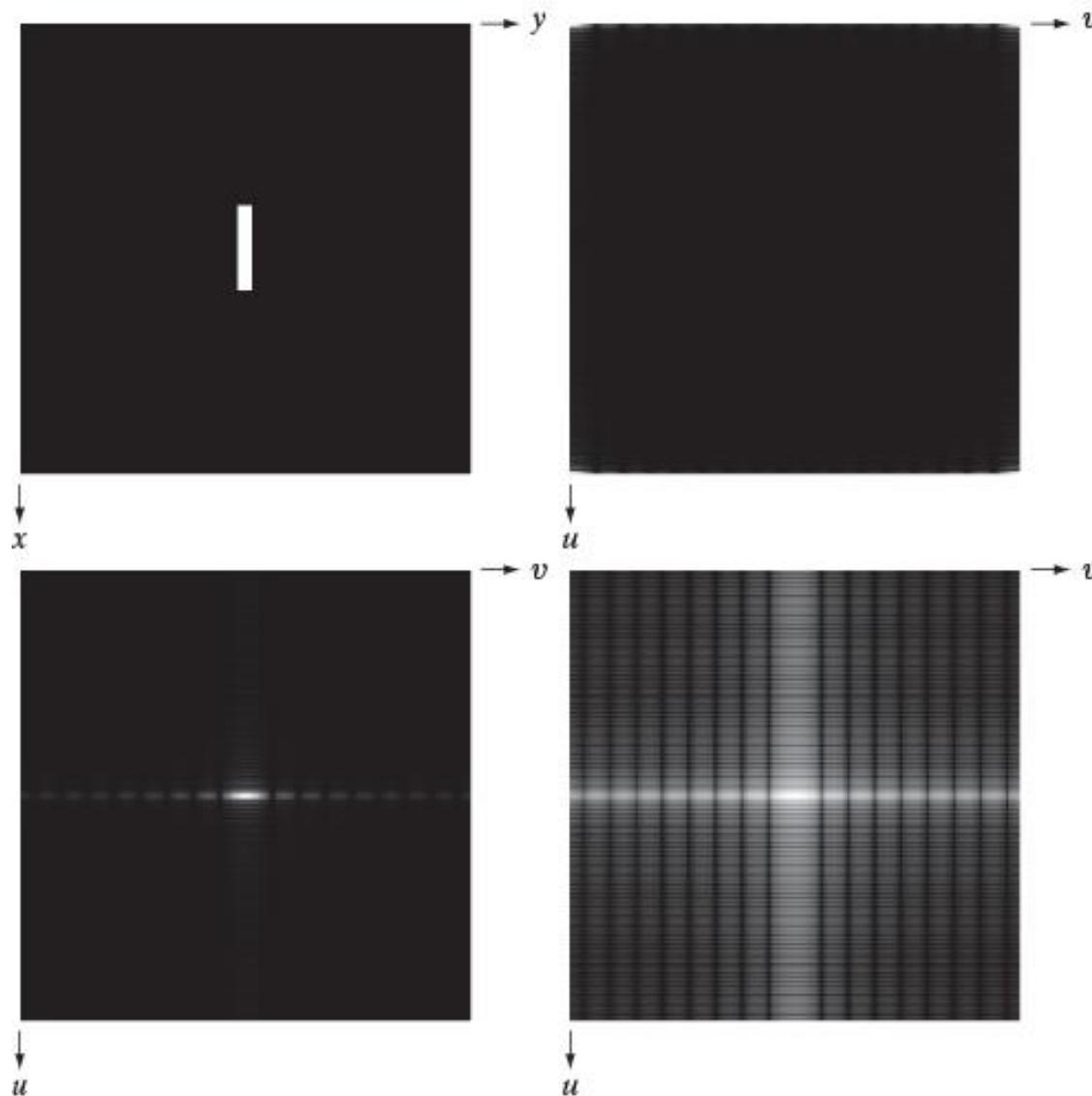
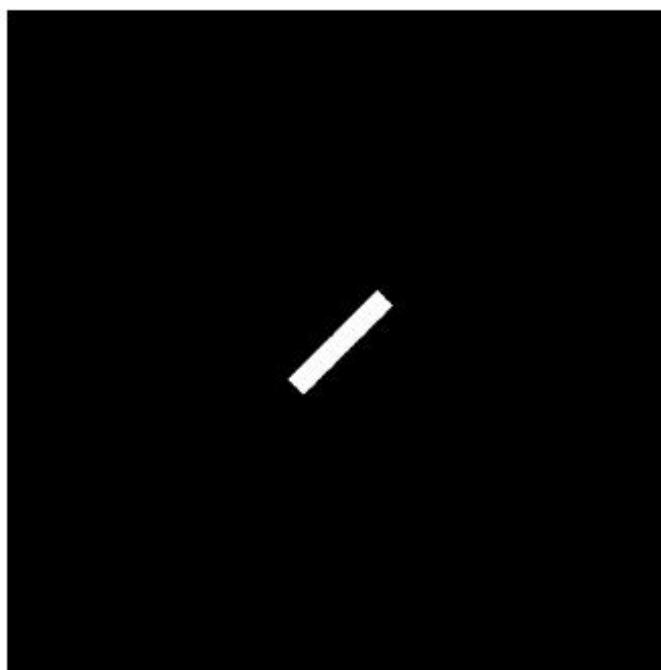
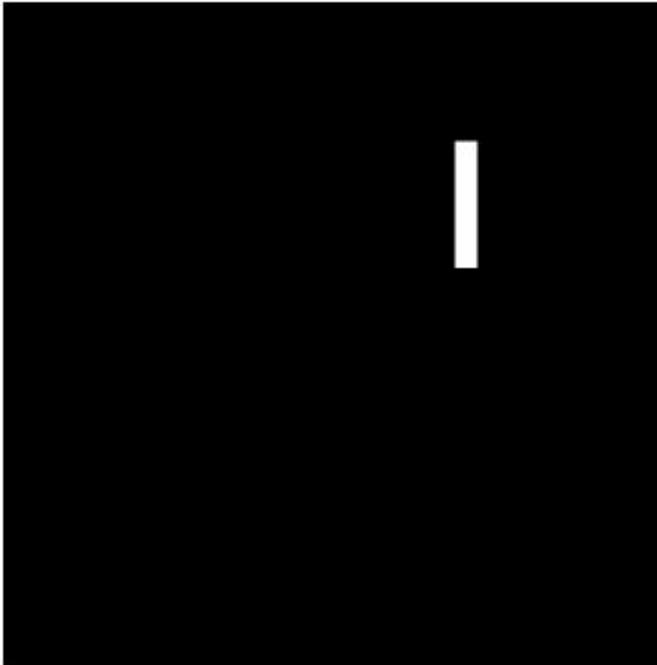


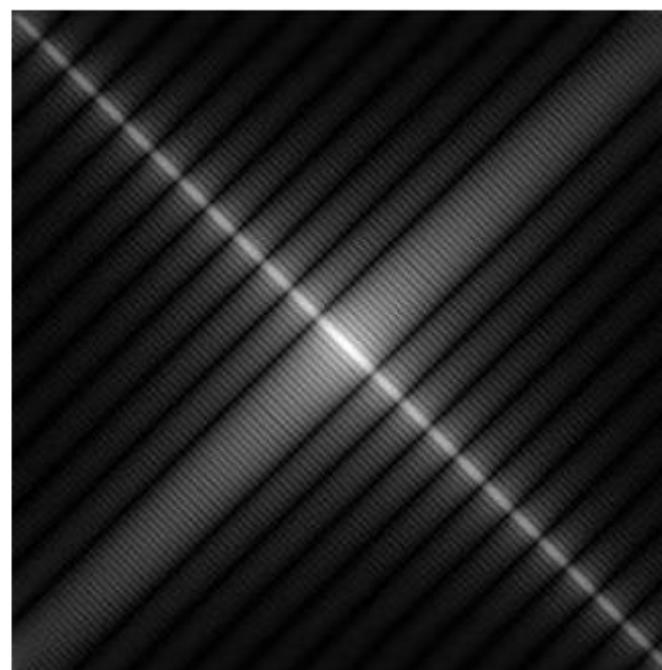
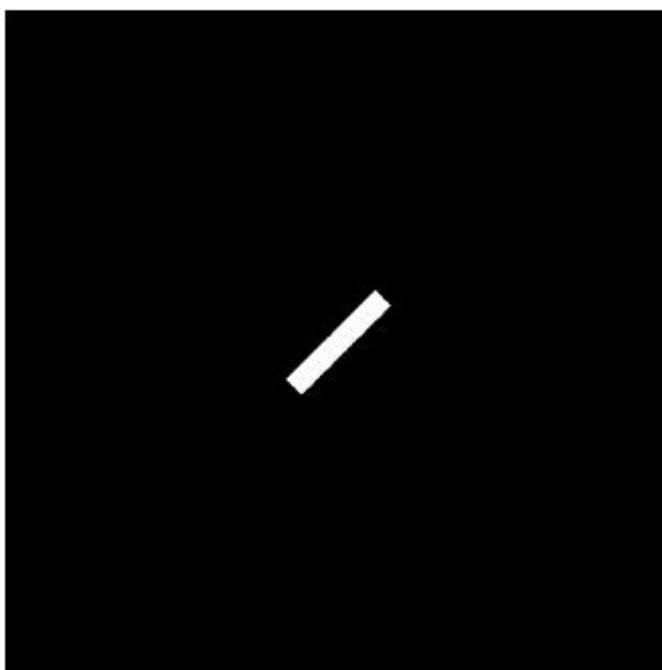
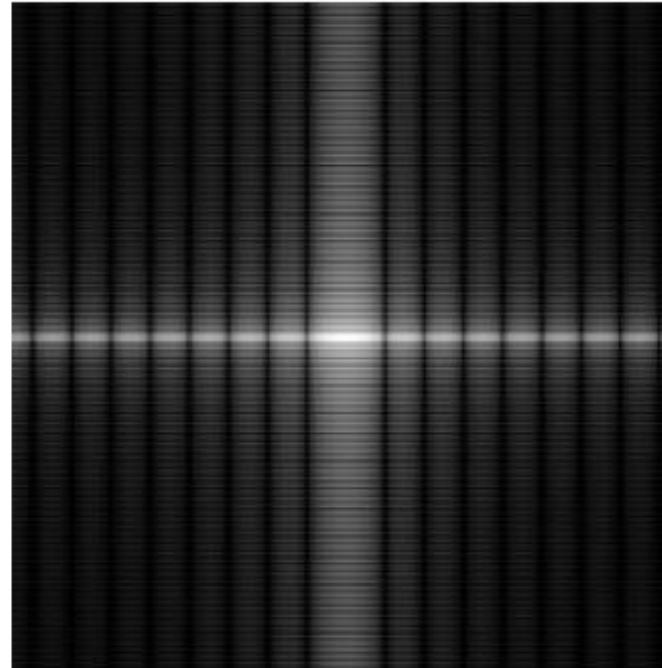
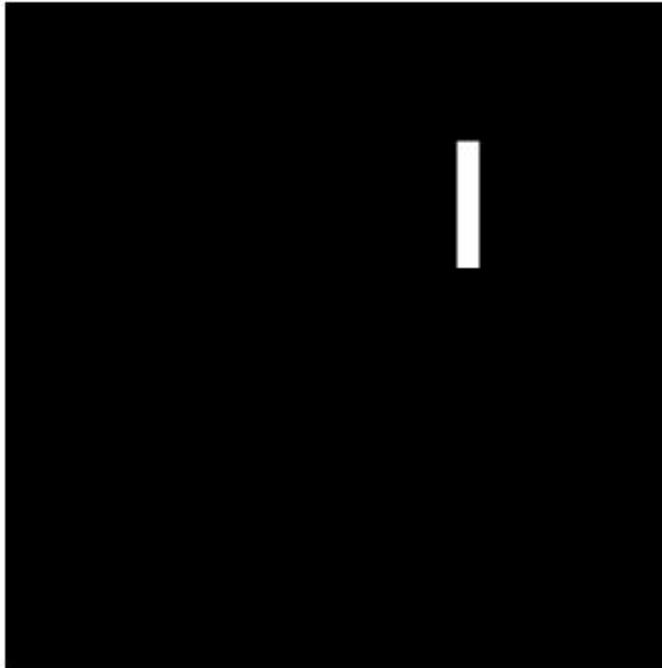
FIGURE 4.23

(a) Image.
(b) Spectrum,
showing small,
bright areas in the
four corners (you
have to look care-
fully to see them).
(c) Centered
spectrum. $f(x,y)(-1)^{x+y} \Leftrightarrow F(u-M/2, v-N/2)$
(d) Result after a
log transformation.
The zero crossings
of the spectrum
are closer in the
vertical direction
because the rectan-
gle in (a) is longer
in that direction.
The right-handed
coordinate
convention used in
the book places the
origin of the spatial
and frequency
domains at the top
left (see Fig. 2.19).



WR
U



a b
c d**FIGURE 4.24**

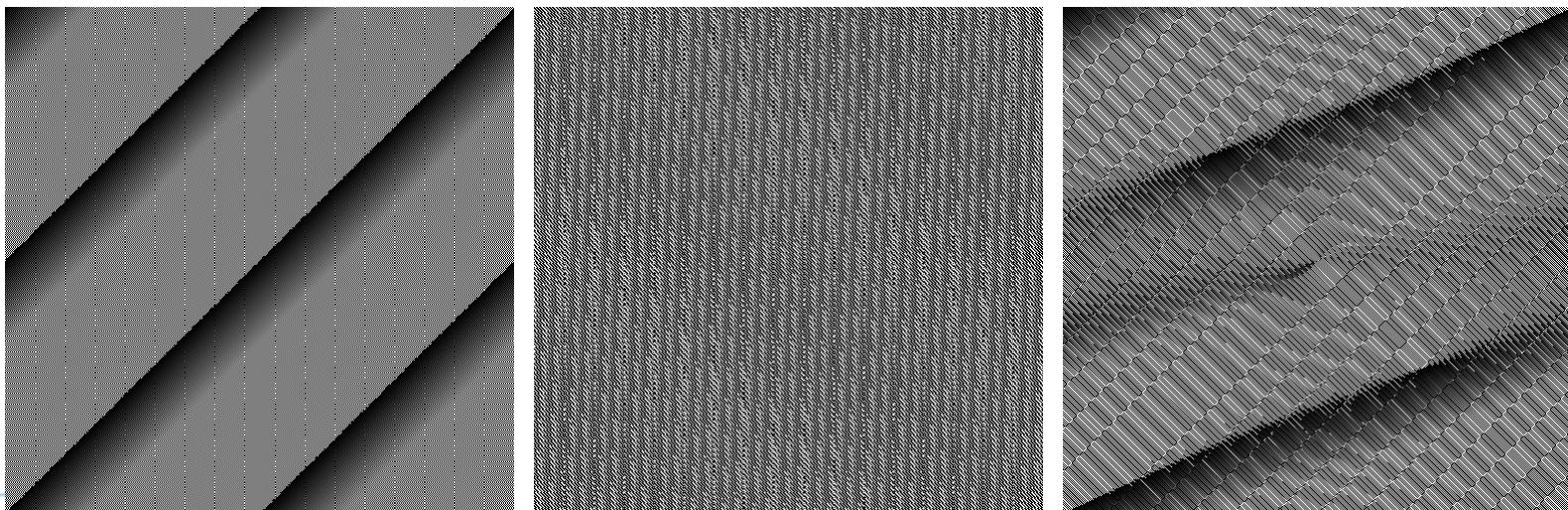
- (a) The rectangle in Fig. 4.23(a) translated.
(b) Corresponding spectrum.
(c) Rotated rectangle.
(d) Corresponding spectrum.
The spectrum of the translated rectangle is identical to the spectrum of the original image in Fig. 4.23(a).



Example: Phase Angles

a b c

FIGURE 4.25
Phase angle
images of
(a) centered,
(b) translated,
and (c) rotated
rectangles.



- Although the images differ by a simple geometric transformation **no intuitive information** may be extracted **from their phases** regarding their relation.



Example: Phase Angles and The Reconstructed

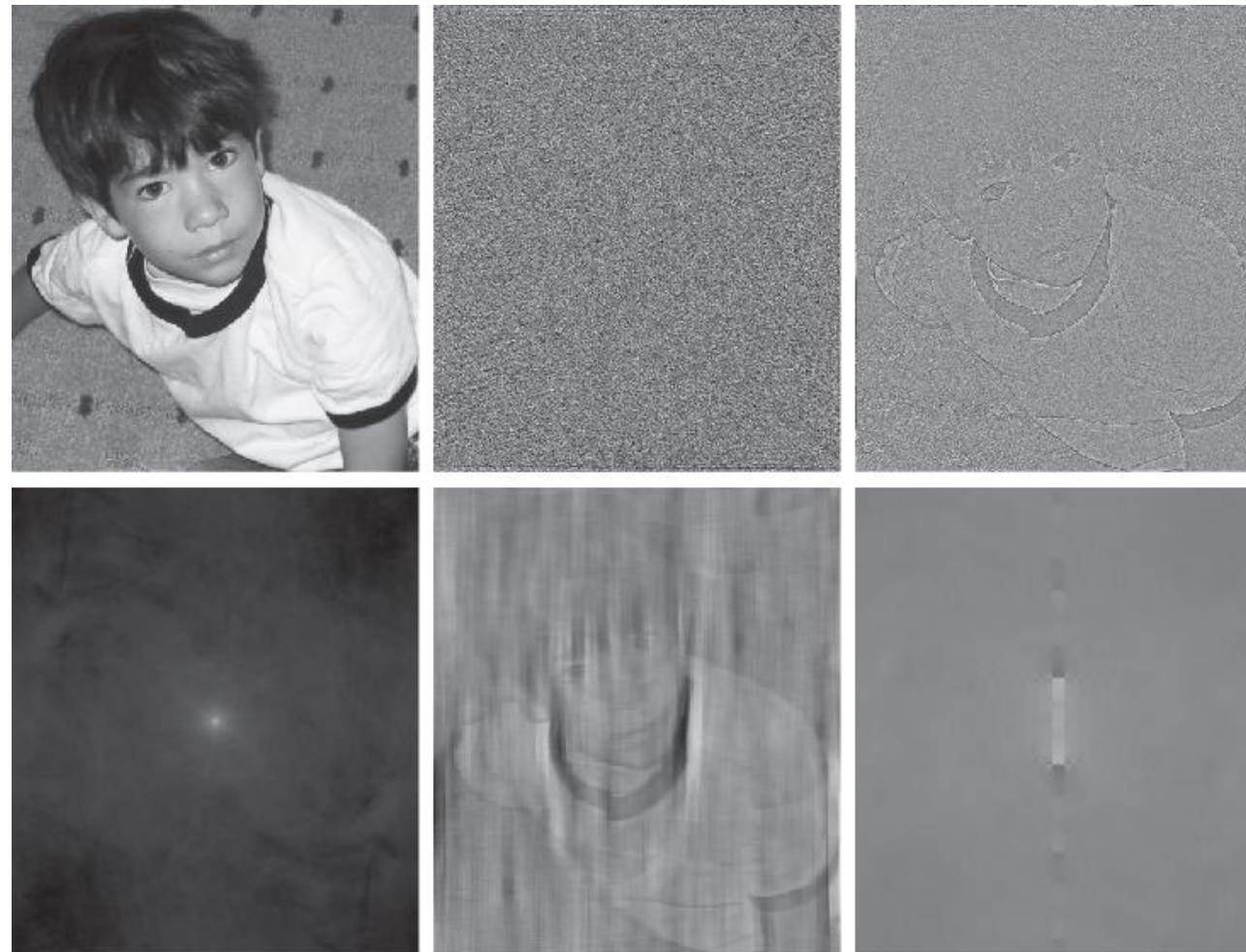


FIGURE 4.26 (a) Boy image. (b) Phase angle. (c) Boy image reconstructed using only its phase angle (all shape features are there, but the intensity information is missing because the spectrum was not used in the reconstruction). (d) Boy image reconstructed using only its spectrum. (e) Boy image reconstructed using its phase angle and the spectrum of the rectangle in Fig. 4.23(a). (f) Rectangle image reconstructed using its phase and the spectrum of the boy's image.



Example: Phase Angles and The Reconstructed

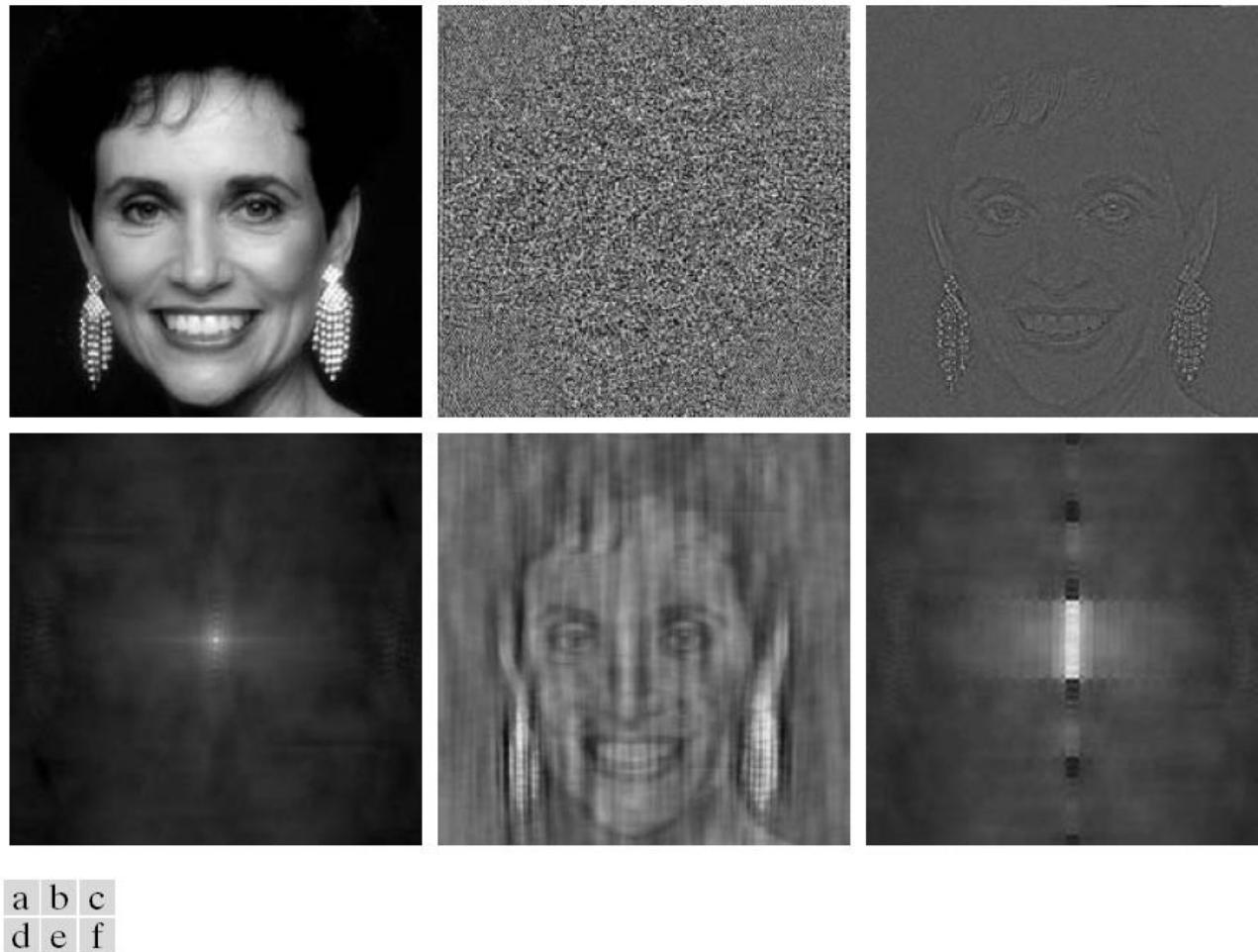


FIGURE 4.27 (a) Woman. (b) Phase angle. (c) Woman reconstructed using only the phase angle. (d) Woman reconstructed using only the spectrum. (e) Reconstruction using the phase angle corresponding to the woman and the spectrum corresponding to the rectangle in Fig. 4.24(a). (f) Reconstruction using the phase of the rectangle and the spectrum of the woman.

3rd Edition

4.6.6 2-D Convolution Theorem

1-D convolution

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

2-D convolution

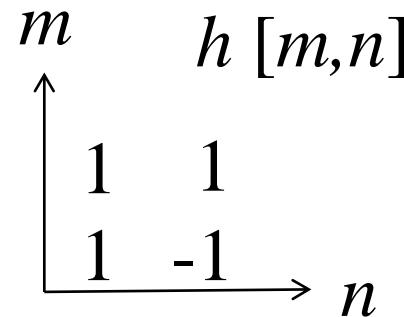
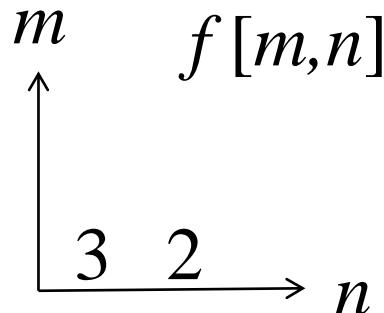
$$f(x, y) \star h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)h(x-m, y-n)$$

$$x = 0, 1, 2, \dots, M-1; y = 0, 1, 2, \dots, N-1.$$

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

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

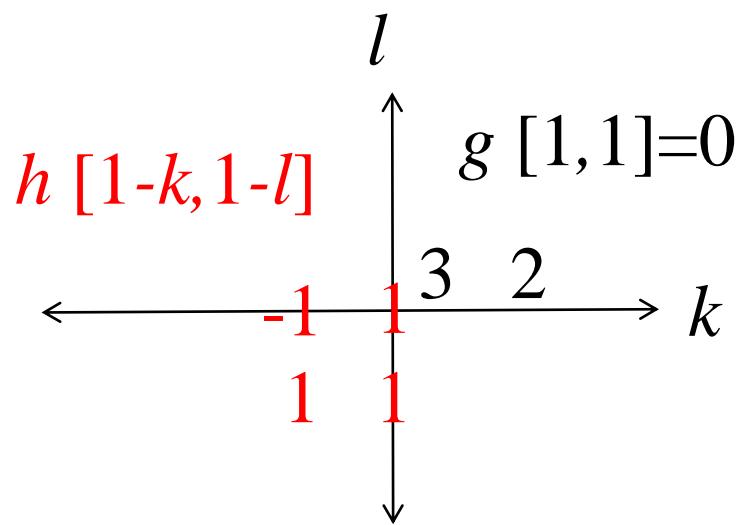
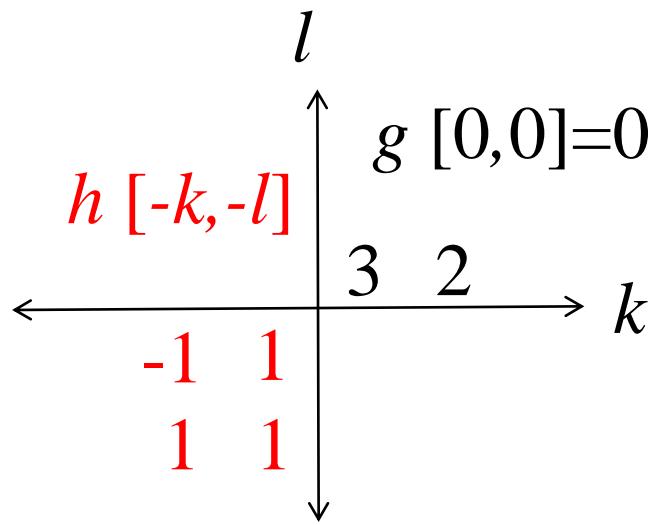
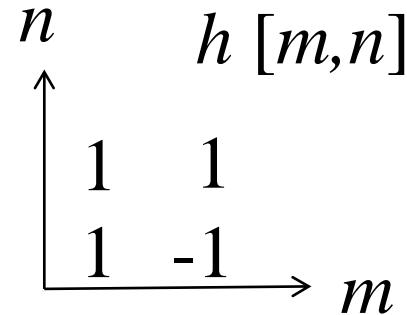
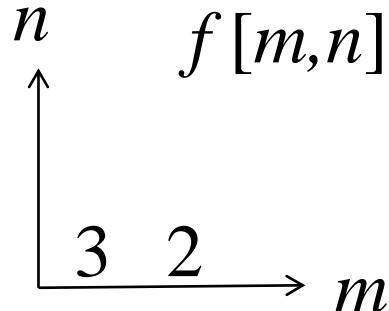
2D Discrete Convolution



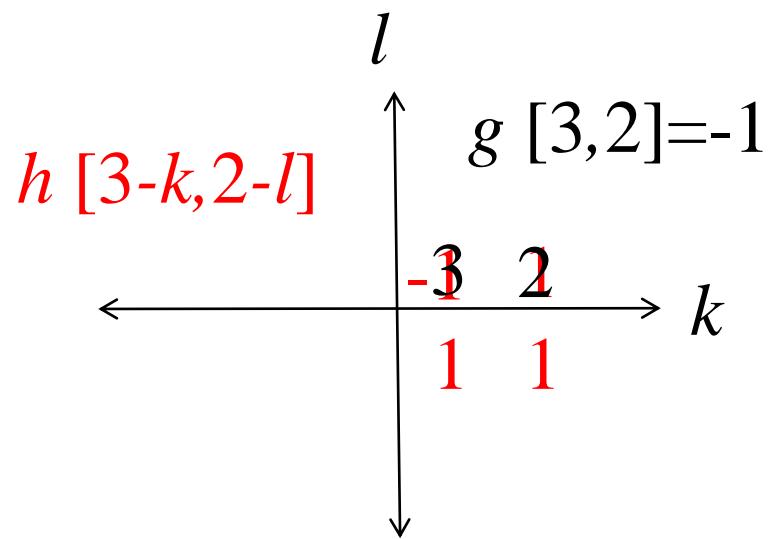
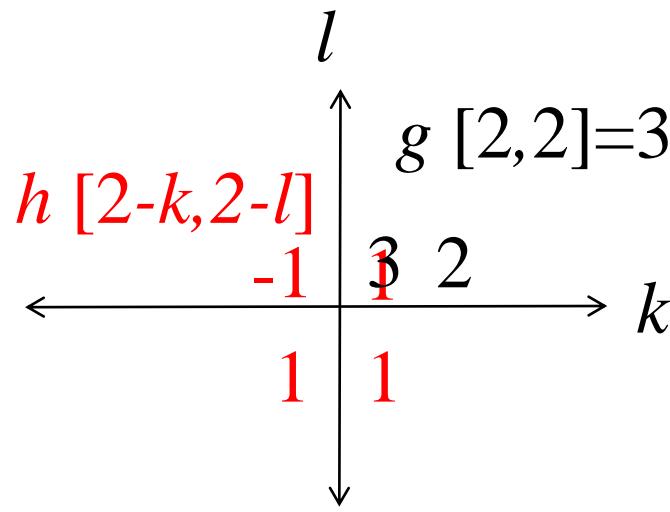
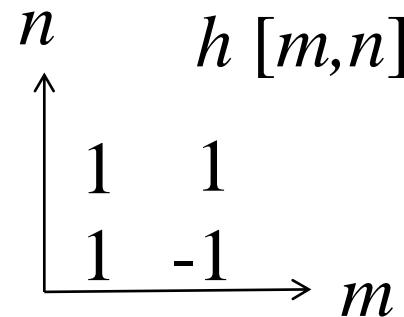
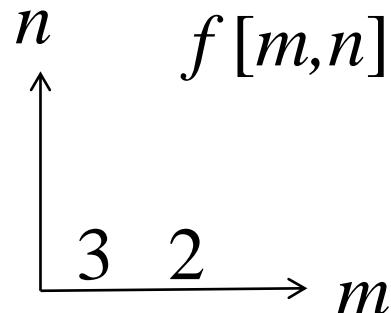
$$g[m, n] = f[m, n] \star h[m, n] = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} f[k, l] h[m-k, n-l]$$

- Take the symmetric mirror image of one of the signals with respect to the origin.
- Shift it and compute the sum at every position $[m, n]$.

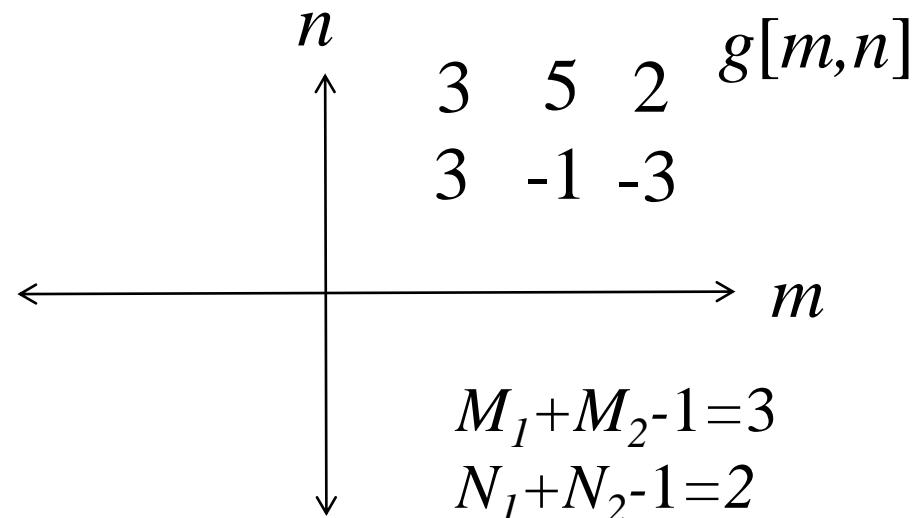
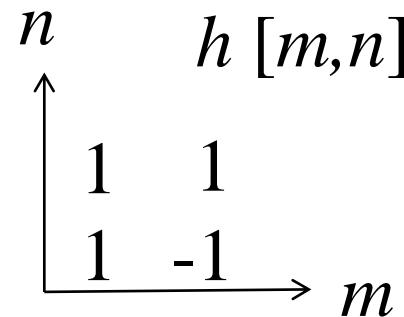
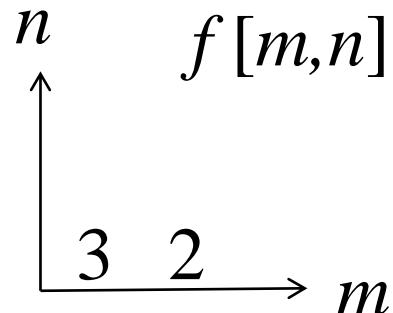
2D Discrete Convolution (cont.)



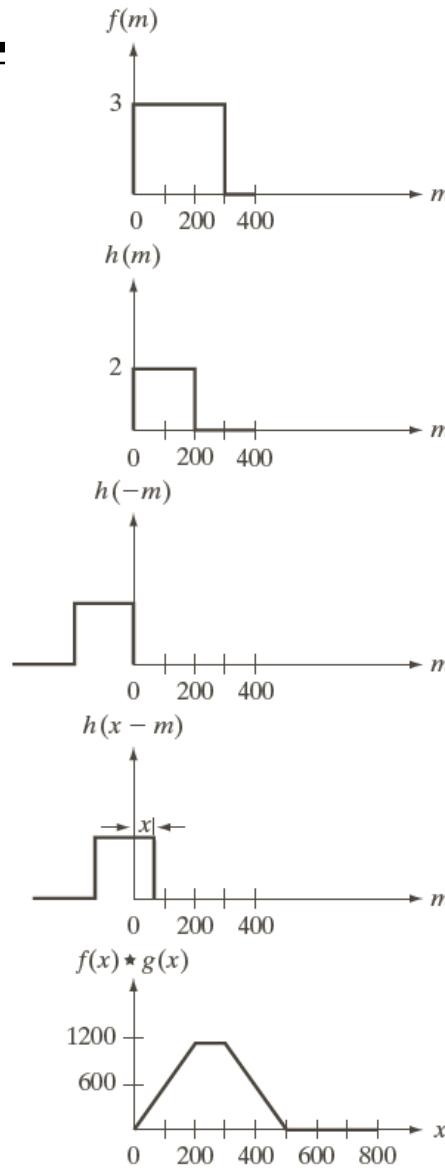
2D Discrete Convolution (cont.)



2D Discrete Convolution (cont.)



An Example of 1-D Convolution



Mirroring h about the origin

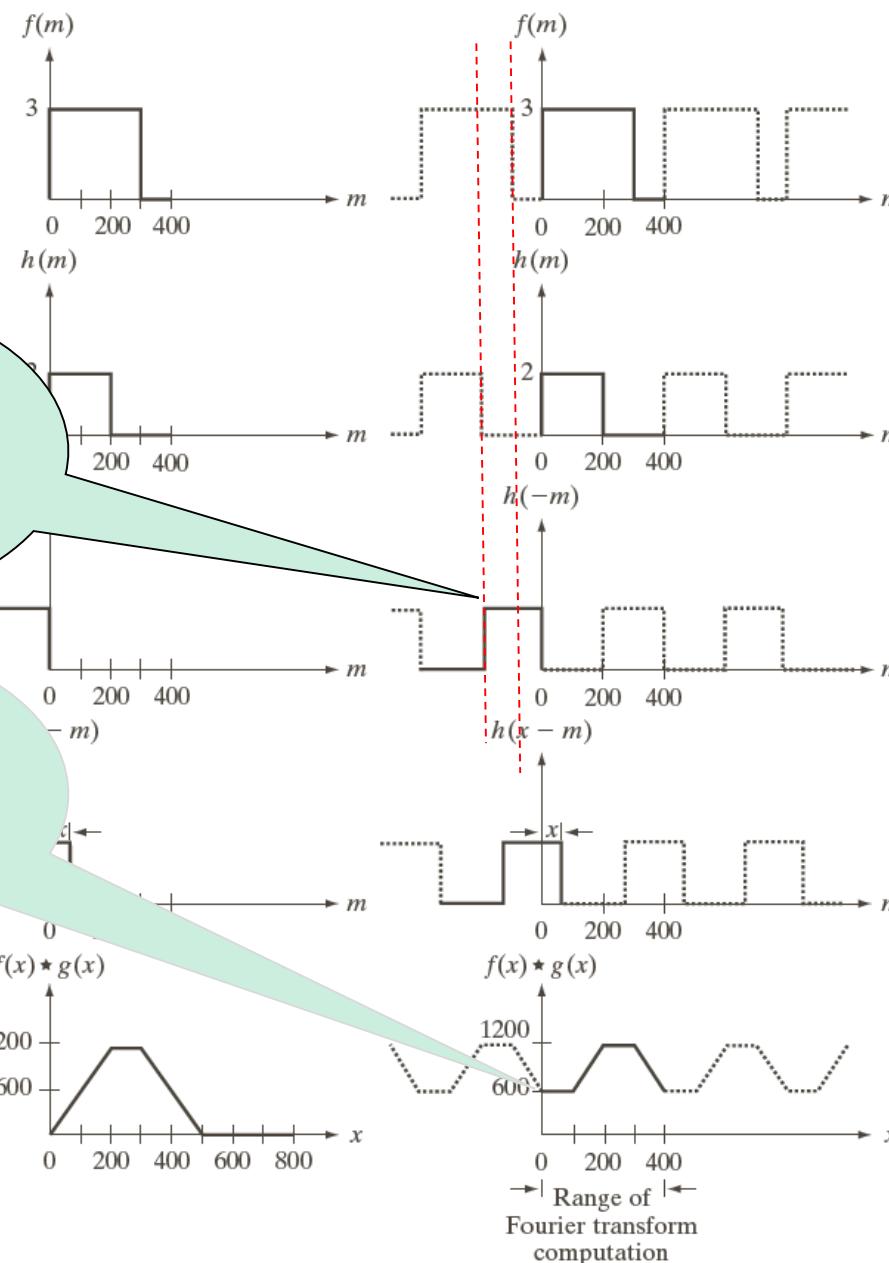
Translating the mirrored function by x

Computing the sum for each x

a	f
b	g
c	h
d	i
e	j

FIGURE 4.27
 Left column: Spatial convolution computed with Eq. (3-44), using the approach discussed in Section 3.4.
 Right column: Circular convolution. The solid line in (j) is the result we would obtain using the DFT, or, equivalently, Eq. (4-48). This erroneous result can be remedied by using zero padding.

Example of Circular Convolution



a	f
b	g
c	h
d	i
e	j

FIGURE 4.27
 Left column: Spatial convolution computed with Eq. (3-44), using the approach discussed in Section 3.4.
 Right column: Circular convolution. The solid line in (j) is the result we would obtain using the DFT, or, equivalently, Eq. (4-48). This erroneous result can be remedied by using zero padding.

It causes the wraparound error

It can be solved by appending zeros



Summary

TABLE 4.3
Summary of DFT
definitions and
corresponding
expressions.

Name	Expression(s)
1) Discrete Fourier transform (DFT) of $f(x, y)$	$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M+vy/N)}$
2) Inverse discrete Fourier transform (IDFT) of $F(u, v)$	$f(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M+vy/N)}$
3) Polar representation	$F(u, v) = F(u, v) e^{j\phi(u, v)}$
4) Spectrum	$ F(u, v) = [R^2(u, v) + I^2(u, v)]^{1/2}$ $R = \text{Real}(F); \quad I = \text{Imag}(F)$
5) Phase angle	$\phi(u, v) = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]$
6) Power spectrum	$P(u, v) = F(u, v) ^2$
7) Average value	$\bar{f}(x, y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) = \frac{1}{MN} F(0, 0)$



Summary

TABLE 4.3

Summary of DFT definitions and corresponding expressions.

Name	Expression(s)
8) Periodicity (k_1 and k_2 are integers)	$\begin{aligned} F(u, v) &= F(u + k_1M, v) = F(u, v + k_2N) \\ &= F(u + k_1M, v + k_2N) \end{aligned}$ $\begin{aligned} f(x, y) &= f(x + k_1M, y) = f(x, y + k_2N) \\ &= f(x + k_1M, y + k_2N) \end{aligned}$
9) Convolution	$f(x, y) \star h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)h(x - m, y - n)$
10) Correlation	$f(x, y) \star h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n)h(x + m, y + n)$
11) Separability	The 2-D DFT can be computed by computing 1-D DFT transforms along the rows (columns) of the image, followed by 1-D transforms along the columns (rows) of the result. See Section 4.11.1.
12) Obtaining the inverse Fourier transform using a forward transform algorithm.	$MNf^*(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v)e^{-j2\pi(ux/M+vy/N)}$ <p>This equation indicates that inputting $F^*(u, v)$ into an algorithm that computes the forward transform (right side of above equation) yields $MNf^*(x, y)$. Taking the complex conjugate and dividing by MN gives the desired inverse. See Section 4.11.2.</p>



Summary

TABLE 4.4

Summary of DFT pairs. The closed-form expressions in 12 and 13 are valid only for continuous variables. They can be used with discrete variables by sampling the continuous expressions.

Name	DFT Pairs
1) Symmetry properties	See Table 4.1
2) Linearity	$af_1(x, y) + bf_2(x, y) \Leftrightarrow aF_1(u, v) + bF_2(u, v)$
3) Translation (general)	$f(x, y)e^{j2\pi(u_0x/M+v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v)e^{-j2\pi(ux_0/M+vy_0/N)}$
4) Translation to center of the frequency rectangle, $(M/2, N/2)$	$f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{u+v}$
5) Rotation	$f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$ $x = r \cos \theta \quad y = r \sin \theta \quad u = \omega \cos \varphi \quad v = \omega \sin \varphi$
6) Convolution theorem [†]	$f(x, y) \star h(x, y) \Leftrightarrow F(u, v)H(u, v)$ $f(x, y)h(x, y) \Leftrightarrow F(u, v) \star H(u, v)$

(Continued)



Summary

TABLE 4.4

Summary of DFT pairs. The closed-form expressions in 12 and 13 are valid only for continuous variables. They can be used with discrete variables by sampling the continuous expressions.

Name	DFT Pairs
7) Correlation theorem [†]	$f(x, y) \star h(x, y) \Leftrightarrow F^*(u, v) H(u, v)$ $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \star H(u, v)$
8) Discrete unit impulse	$\delta(x, y) \Leftrightarrow 1$
9) Rectangle	$\text{rect}[a, b] \Leftrightarrow ab \frac{\sin(\pi ua)}{(\pi ua)} \frac{\sin(\pi vb)}{(\pi vb)} e^{-j\pi(ua+vb)}$
10) Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $j \frac{1}{2} [\delta(u + Mu_0, v + Nv_0) - \delta(u - Mu_0, v - Nv_0)]$
11) Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $\frac{1}{2} [\delta(u + Mu_0, v + Nv_0) + \delta(u - Mu_0, v - Nv_0)]$
The following Fourier transform pairs are derivable only for continuous variables, denoted as before by t and z for spatial variables and by μ and ν for frequency variables. These results can be used for DFT work by sampling the continuous forms.	
12) Differentiation (The expressions on the right assume that $f(\pm\infty, \pm\infty) = 0$)	$\left(\frac{\partial}{\partial t}\right)^m \left(\frac{\partial}{\partial z}\right)^n f(t, z) \Leftrightarrow (j2\pi\mu)^m (j2\pi\nu)^n F(\mu, \nu)$ $\frac{\partial^m f(t, z)}{\partial t^m} \Leftrightarrow (j2\pi\mu)^m F(\mu, \nu); \frac{\partial^n f(t, z)}{\partial z^n} \Leftrightarrow (j2\pi\nu)^n F(\mu, \nu)$
13) Gaussian	$A 2\pi \sigma^2 e^{-2\pi^2 \sigma^2 (t^2 + z^2)} \Leftrightarrow A e^{-(\mu^2 + \nu^2)/2\sigma^2}$ (A is a constant)

[†] Assumes that the functions have been extended by zero padding. Convolution and correlation are associative, commutative, and distributive.

Zero Padding - 1D

- ▶ Consider two functions $f(x)$ and $h(x)$ composed of A and B samples, respectively
- ▶ Append zeros to both functions so that they have the same length, denoted by P , then wraparound is avoided by choosing

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

Zero Padding - 2D

- ▶ Let $f(x, y)$ and $h(x, y)$ be two image arrays of sizes $A \times B$ and $C \times D$ pixels, respectively. Wraparound error in their convolution can be avoided by padding these functions with zeros

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

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

Here, $P \geq A + C - 1$; $Q \geq B + D - 1$

4.7.2 The Basic Filtering in the Frequency Domain

- ▶ Filtering in the frequency domain consists of modifying the FT of an image then computing the inverse transform
- ▶ Given an image $f(x, y)$ of size $M \times N$, the basic filtering is
 - Multiply the Fourier Transforms of the image and the filter
 - Compute the inverse transform to obtain the processed result
(see next page for a block diagram)

$$g(x, y) = \text{Real} \left\{ \mathfrak{F}^{-1} [H(u, v)F(u, v)] \right\}$$

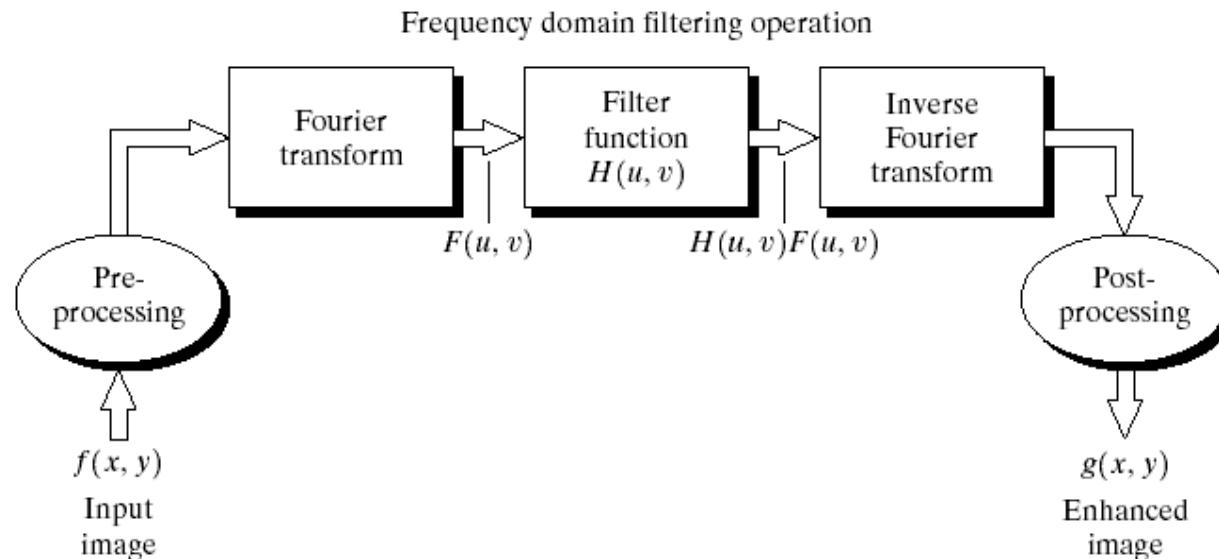
$F(u, v)$ is the DFT of the input image

$H(u, v)$ is a filter function.

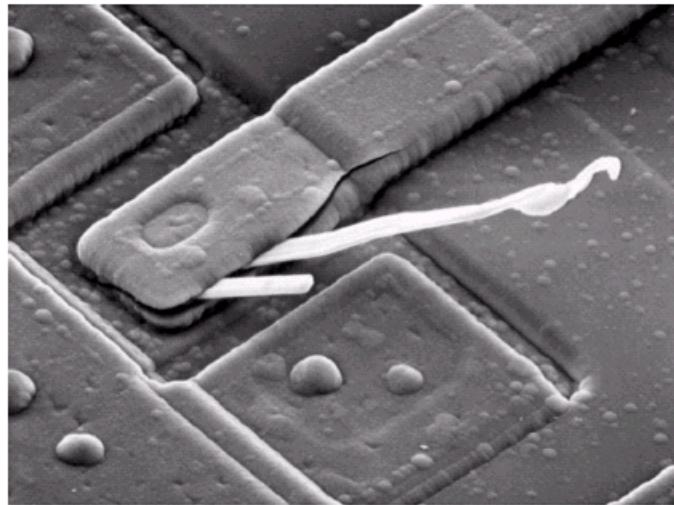
The DFT and Image Processing

To filter an image in the frequency domain:

1. Compute $F(u, v)$, the DFT of the image
2. Multiply $F(u, v)$ by a filter function $H(u, v)$
3. Compute the inverse DFT of the result



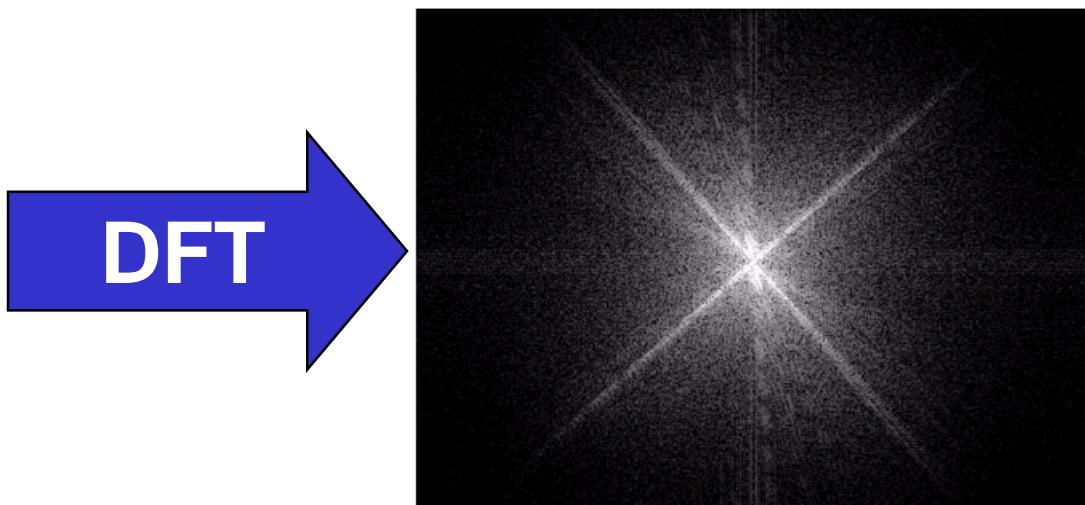
DFT & Images (cont...)



Scanning electron
microscope image of an
integrated circuit
magnified ~2500 times

a b

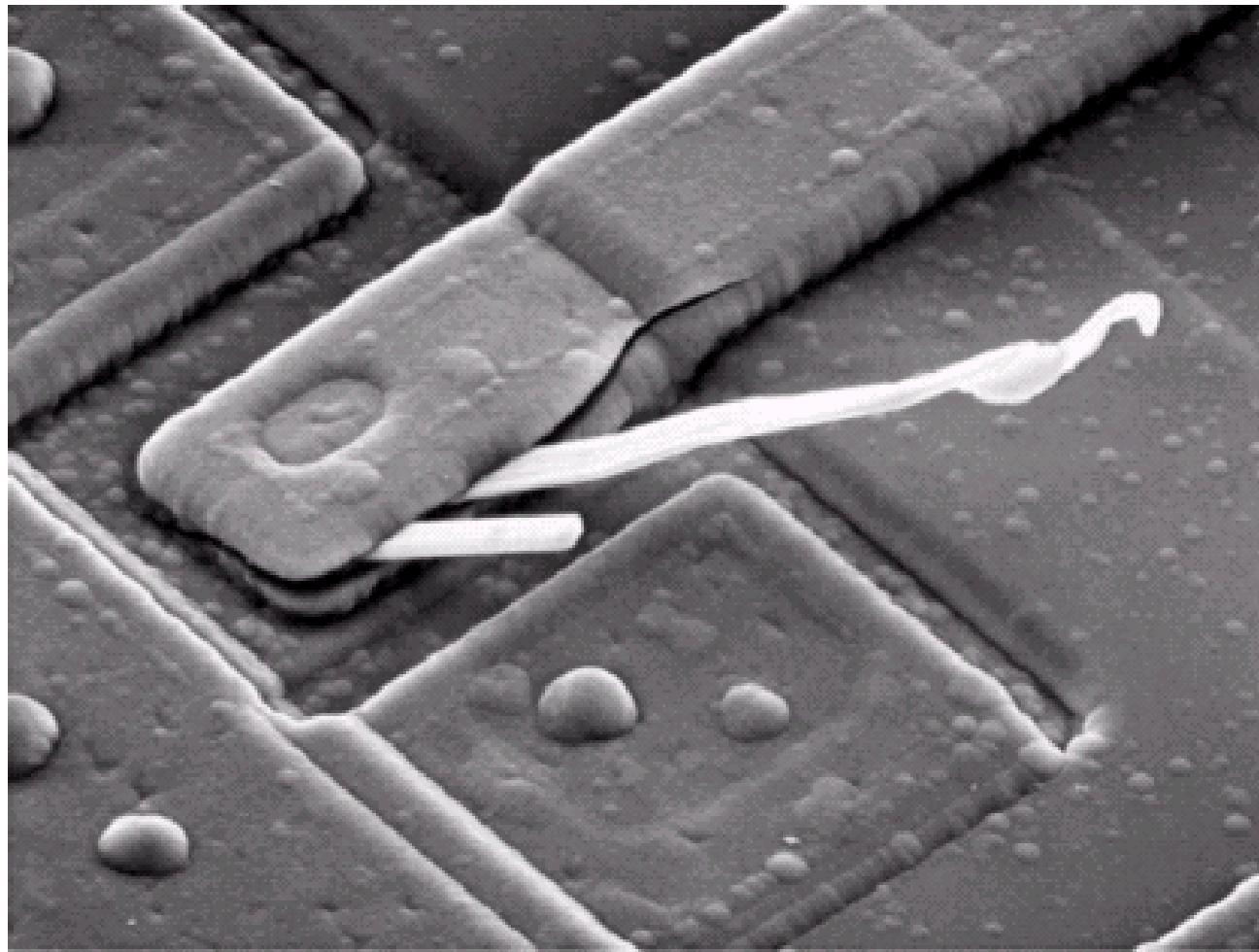
FIGURE 4.28 (a) SEM image of a damaged integrated circuit. (b) Fourier spectrum of (a). (Original image courtesy of Dr. J. M. Hudak, Brockhouse Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada.)



Fourier spectrum
of the image



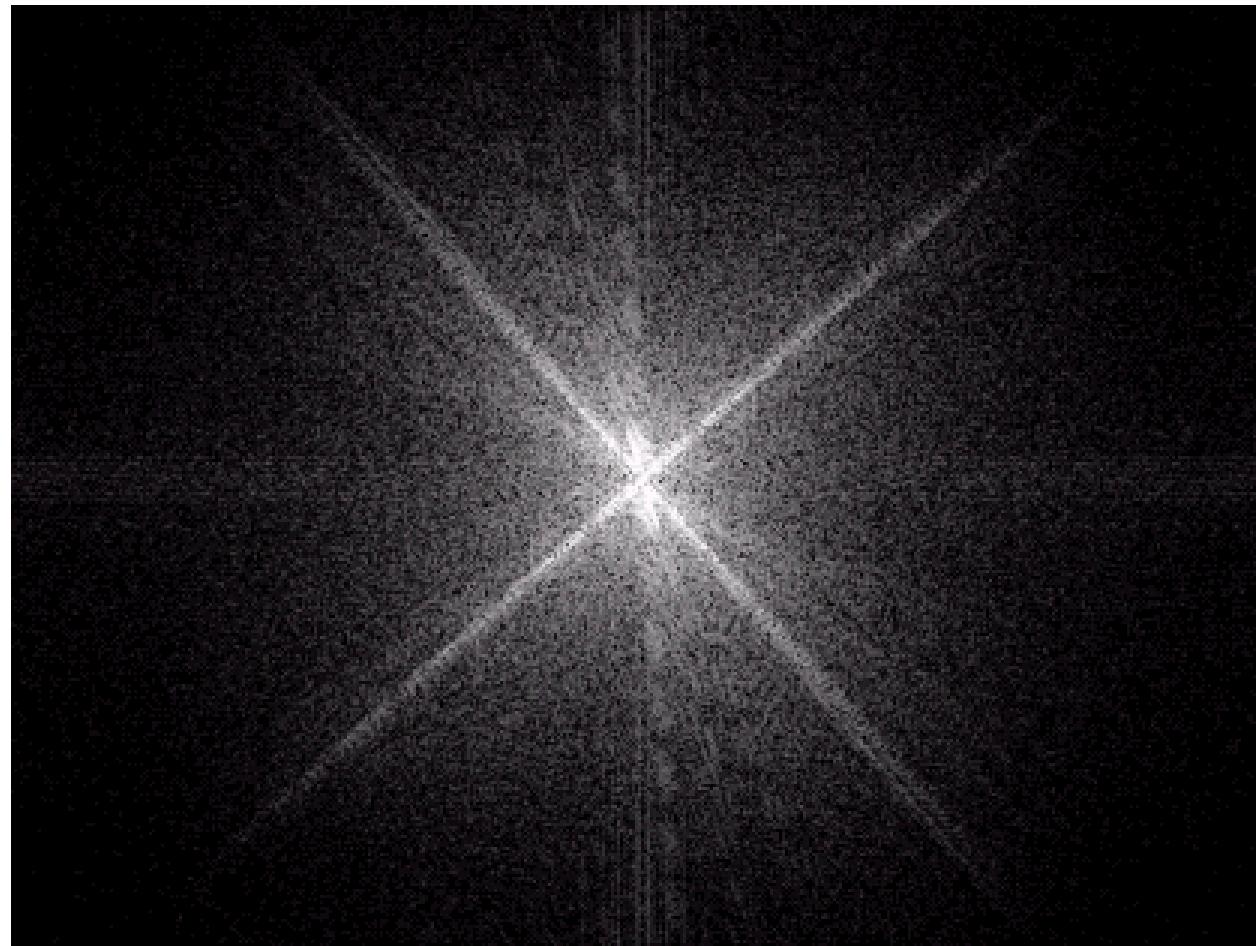
DFT & Images (cont...)





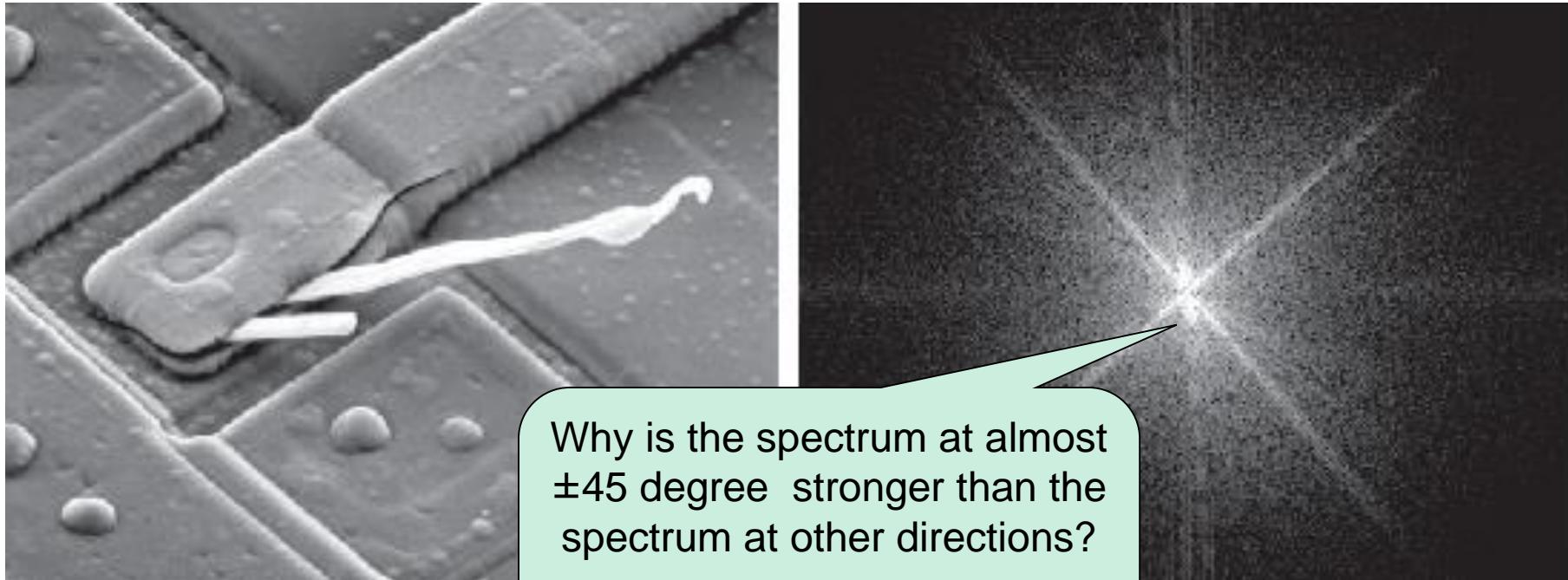
WRIGHT STATE
UNIVERSITY

DFT & Images (cont...)





4.7 The Basic Filtering in the Frequency Domain



a b

FIGURE 4.28 (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.)

Basic Filtering in the Frequency Domain - Effect of DC-term

- ▶ Fig 4.28 and 4.29: In a filter $H(u, v)$ that is 0 at the center of the transform and 1 elsewhere, what's the output image?

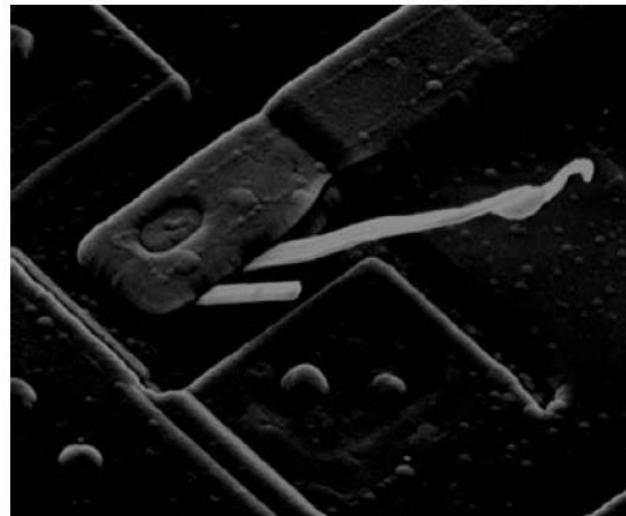
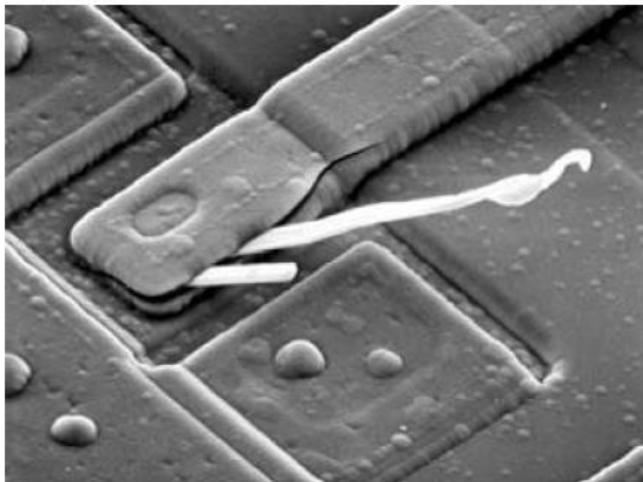


FIGURE 4.29
Result of filtering the image in Fig. 4.28(a) with a filter transfer function that sets to 0 the dc term, $F(P/2, Q/2)$, in the centered Fourier transform, while leaving all other transform terms unchanged.

- Reject DC term and Pass all other terms
- Image becomes much darker

- **Lowpass filter:** Filter $H(u, v)$
 - Attenuates high frequencies
 - Pass low frequencies - would blur an image
- **Highpass filter:**
 - Would enhance sharp details
 - But cause reduction in contrast in the image

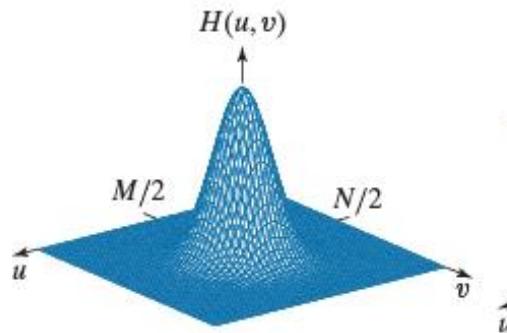


The Basic Filtering in the Frequency Domain

Lowpass

Highpass

Raised Highpass



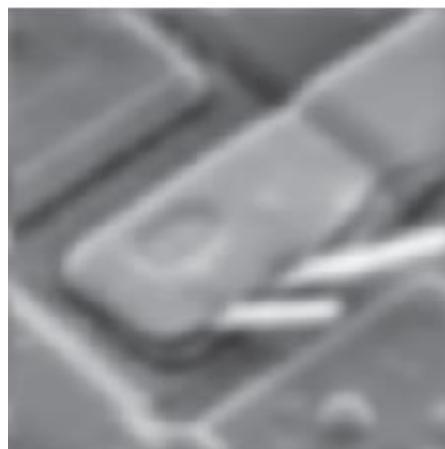
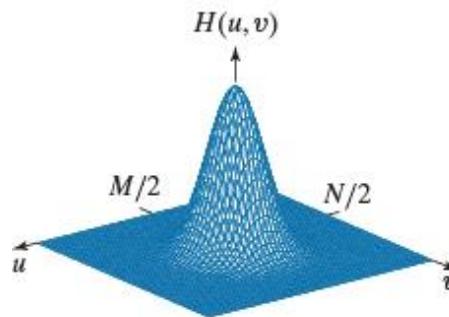
a	b	c
d	e	f

FIGURE 4.30 Top row: Frequency domain filter transfer functions of (a) a lowpass filter, (b) a highpass filter, and (c) an offset highpass filter. Bottom row: Corresponding filtered images obtained using Eq. (4-104). The offset in (c) is $a = 0.85$, and the height of $H(u,v)$ is 1. Compare (f) with Fig. 4.28(a).

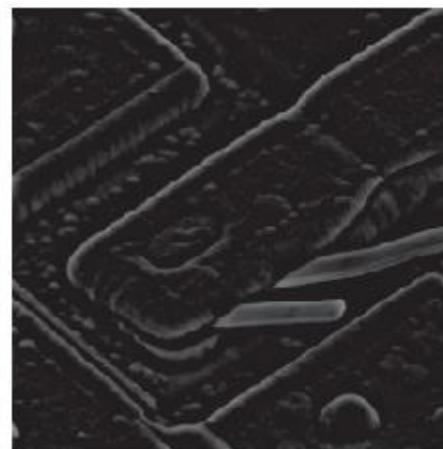
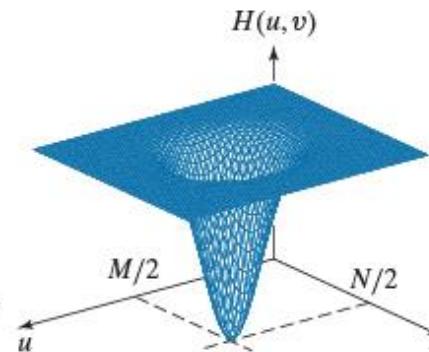


The Basic Filtering in the Frequency Domain

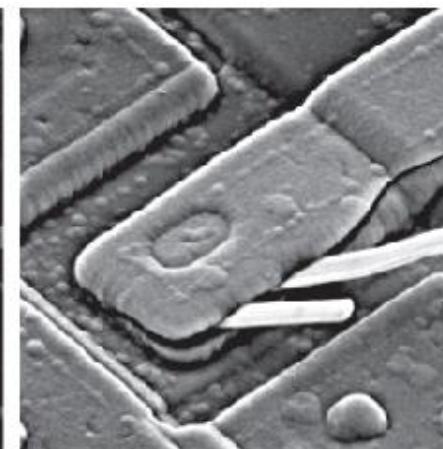
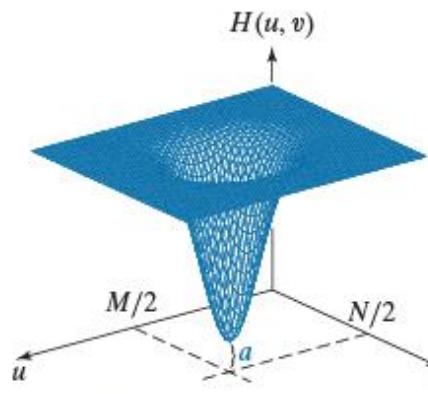
Lowpass



Highpass



Raised Highpass



a	b	c
d	e	f

FIGURE 4.30 Top row: Frequency domain filter transfer functions of (a) a lowpass filter, (b) a highpass filter, and (c) an offset highpass filter. Bottom row: Corresponding filtered images obtained using Eq. (4-104). The offset in (c) is $a = 0.85$, and the height of $H(u,v)$ is 1. Compare (f) with Fig. 4.28(a).



Filtering in the Frequency Domain: Effect of Zero-Padding



a | b | c

FIGURE 4.31 (a) A simple image. (b) Result of blurring with a Gaussian lowpass filter without padding. (c) Result of lowpass filtering with zero padding. Compare the vertical edges in (b) and (c).

- See Figure 3.45 and its discussion regarding dark borders caused by zero-padding.

Zero-Phase-Shift Filters

$$g(x, y) = \mathcal{I}^{-1}\{H(u, v)F(u, v)\}$$

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

$$g(x, y) = \mathcal{I}^{-1}[H(u, v)R(u, v) + jH(u, v)I(u, v)]$$

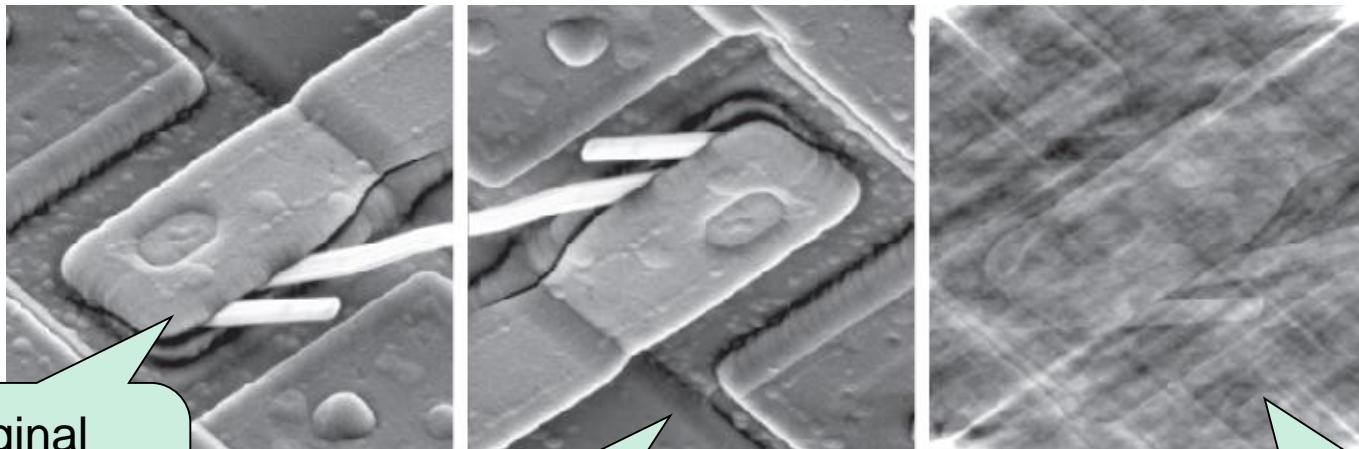
- ▶ Filters affect the real and imaginary parts equally
- ▶ No effect on the phase
- ▶ These filters are called **zero-phase-shift** filters



Example: Nonzero-Phase-Shift Filters

a b c

FIGURE 4.34 (a) Original image. (b) Image obtained by multiplying the phase angle array by -1 in Eq. (4-86) and computing the IDFT. (c) Result of multiplying the phase angle by 0.25 and computing the IDFT. The magnitude of the transform, $|F(u,v)|$, used in (b) and (c) was the same.



Original
Image

Even small changes
dramatic (usually
negative) effects on the final
output

Phase angle
is multiplied
by -1

phase angle can have
dramatic (usually
negative) effects on the final
output

Phase angle
is multiplied
by 0.25

$$F(u,v) = |F(u,v)| e^{j\phi(u,v)} \quad (4.6-15)$$

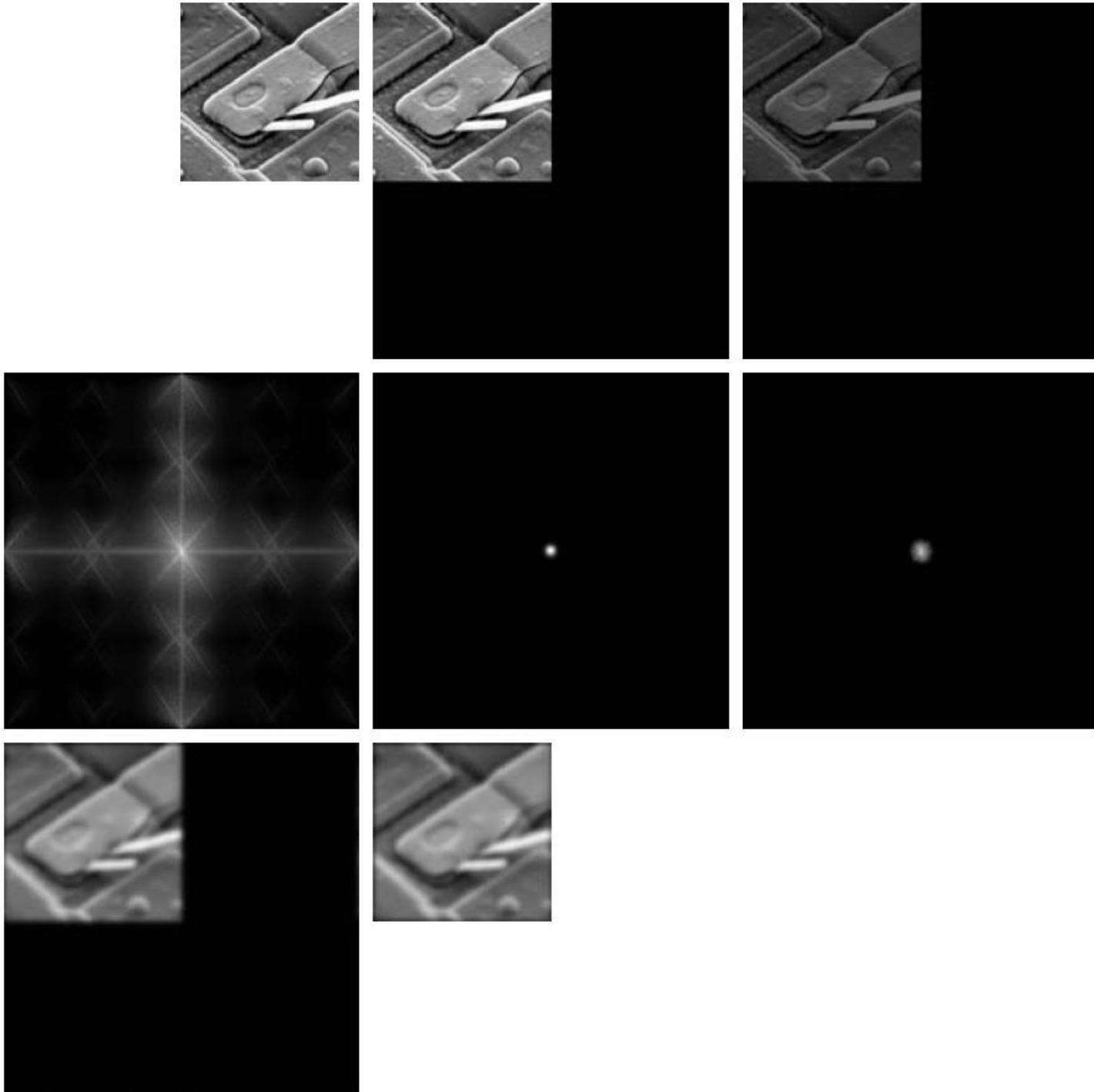
4.7.3 Summary: Steps for Filtering in the Frequency Domain

1. Given an input image $f(x, y)$ of size $M \times N$, obtain the padding parameters P and Q . Typically, $P = 2M$ and $Q = 2N$.
2. Form a padded image, $f_p(x, y)$ of size $P \times Q$ by appending the necessary number of zeros to $f(x, y)$
3. Multiply $f_p(x, y)$ by $(-1)^{x+y}$ to center its transform
4. Compute the DFT, $F(u, v)$ of the image from step 3
5. Generate a real, symmetric filter function*, $H(u, v)$, of size $P \times Q$ with center at coordinates $(P/2, Q/2)$

*generated from a given spatial filter: Pad the spatial filter, multiply the expanded array by $(-1)^{x+y}$, and compute the DFT of the result to obtain a centered $H(u, v)$.

Summary: Steps for Filtering in the Frequency Domain

6. Form the product $G(u,v) = H(u,v)F(u,v)$ using array multiplication
 7. Obtain the processed image
- $$g_p(x, y) = \left\{ \text{real} \left[\mathfrak{I}^{-1} [G(u, v)] \right] \right\} (-1)^{x+y}$$
- p → padded Real part used to ignore computational inaccuracies
8. Obtain the final processed result, $g(x, y)$, by extracting the $M \times N$ region from the top, left quadrant of $g_p(x, y)$



Frequency Domain

a b c
d e f
g h

FIGURE 4.35

- (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 .
- (e) Centered Gaussian lowpass filter transfer function, H , of size $P \times Q$.
- (f) Spectrum of the product HF .
- (g) Image g_p , the real part of the IDFT of HF , multiplied by $(-1)^{x+y}$.
- (h) Final result, g , obtained by extracting the first M rows and N columns of g_p .

Let $H(u)$ denote the 1-D frequency domain Gaussian filter

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

The corresponding filter in the spatial domain

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

- Both functions are Gaussian and real
- The functions behave reciprocally
 - If $H(u)$ is broad, then $h(x)$ is narrow & vice versa

Let $H(u)$ denote the difference of Gaussian filters

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

with $A \geq B$ and $\sigma_1 \geq \sigma_2$

The corresponding filter in the spatial domain

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

Is this a high-pass filter or low-pass filter ?



Complex Filtering by Combining Multiple Gaussians

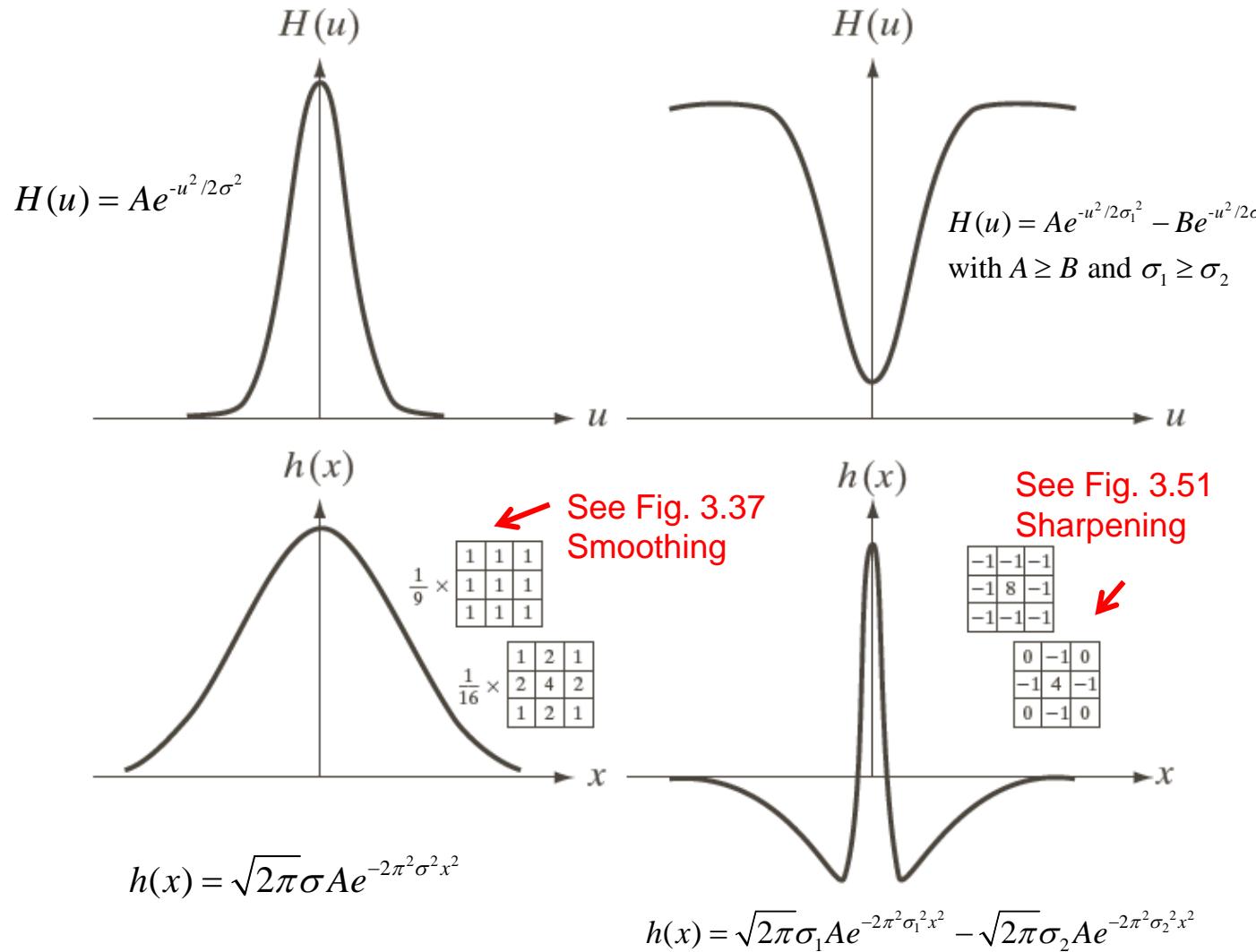
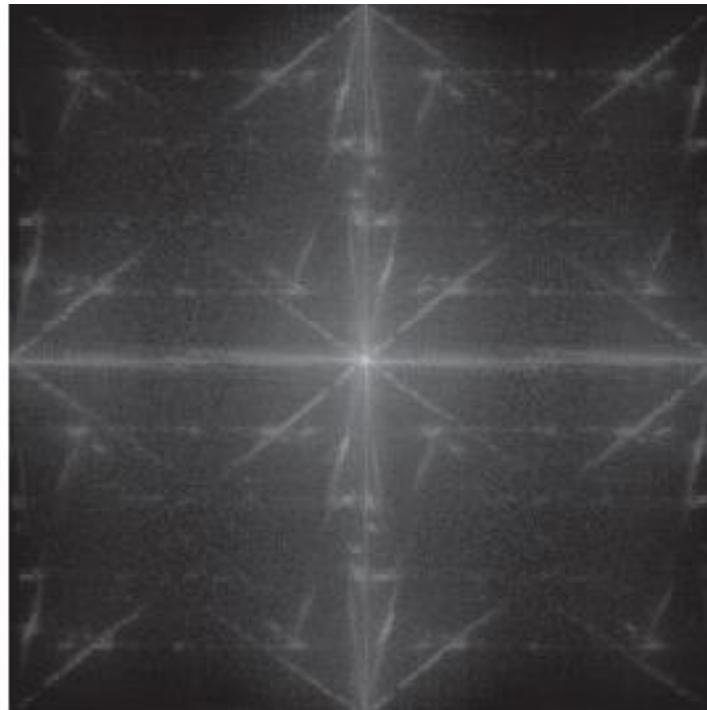


FIGURE 4.36

- (a) A 1-D Gaussian lowpass transfer function in the frequency domain.
(b) Corresponding kernel in the spatial domain. (c) Gaussian highpass transfer function in the frequency domain.
(d) Corresponding kernel. The small 2-D kernels shown are kernels we used in Chapter 3.



Example: Correspondence Between Filtering in the Spatial and Frequency Domain



a b

FIGURE 4.37

(a) Image of a building, and (b) its Fourier spectrum.

600x600



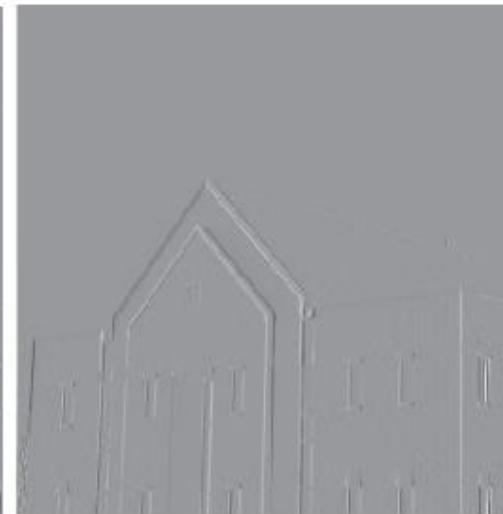
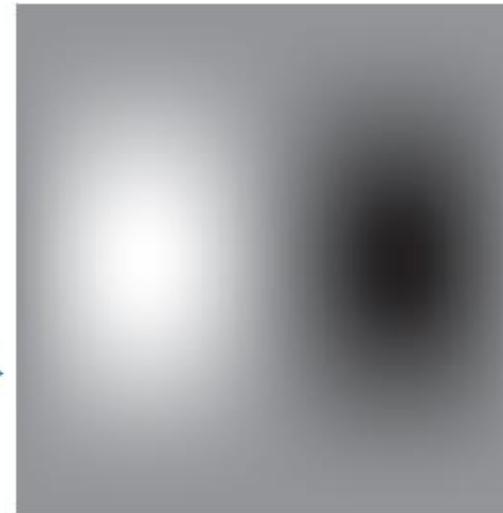
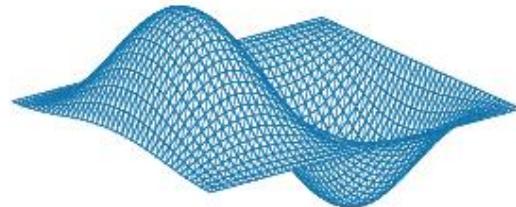
Correspondence Between Filtering in the Spatial and Frequency Domains: Example

Sobel mask

See Fig. 3.56(e)

Note the anti-symmetry (odd) in both domains.
See property-9 of Table 4.1

-1	0	1
-2	0	2
-1	0	1



a b
c d

FIGURE 4.38

- (a) A spatial kernel and perspective plot of its corresponding frequency domain filter transfer function.
- (b) Transfer function shown as an image.
- (c) Result of filtering Fig. 4.37(a) in the frequency domain with the transfer function in (b).
- (d) Result of filtering the same image in the spatial domain with the kernel in (a). The results are identical.

Zero-Pad the Image and the filter to same sizes

Image: $f_p(x, y) = \begin{cases} f(x, y) & 0 \leq x \leq 599 \text{ and } 0 \leq y \leq 599 \\ 0 & 600 \leq x \leq 602 \text{ or } 600 \leq y \leq 602 \end{cases}$

Filter: $h_p(x, y) = \begin{cases} h(x, y) & 0 \leq x \leq 2 \text{ and } 0 \leq y \leq 2 \\ 0 & 3 \leq x \leq 602 \text{ or } 3 \leq y \leq 602 \end{cases}$

$$\text{Here } P \geq A(600) + C(3) - 1 = 602;$$

$$Q \geq B(600) + D(3) - 1 = 602.$$

Then take forward DFT and multiply

Ideal Lowpass Filters (ILPF)

- Pass all frequencies within a circle of radius D_0 from the origin without attenuation and cut off all frequencies outside this circle
- The point of transition between $H(u, v) = 1$ and $H(u, v) = 0$ is called the cutoff frequency

Ideal Lowpass Filters (ILPF)

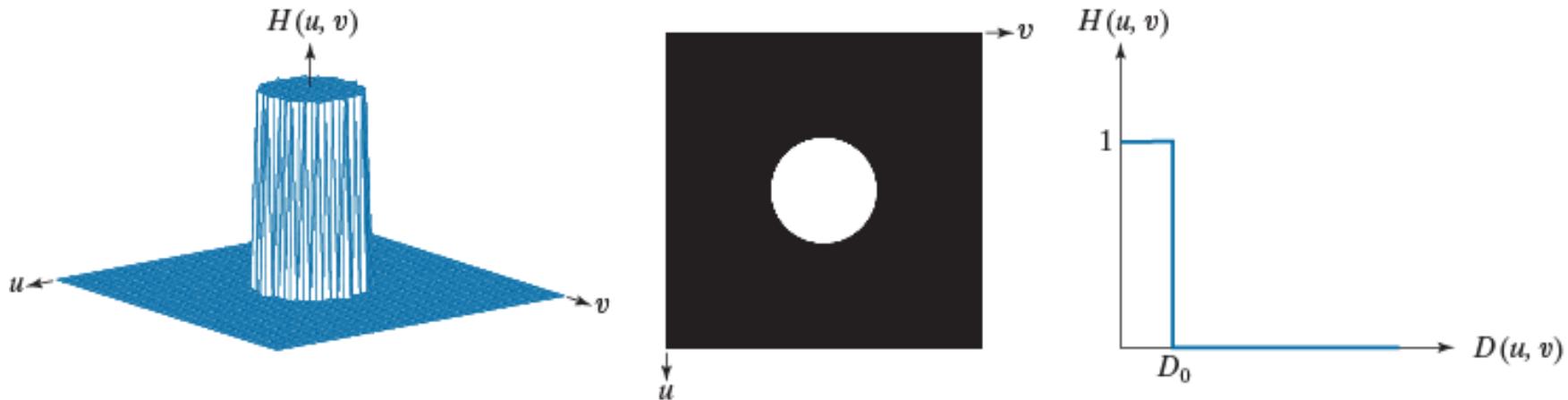
$$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$$

D_0 is a positive constant and $D(u, v)$ is the distance between a point (u, v) in the frequency domain and the center of the frequency rectangle

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



Image Smoothing Using Frequency Domain Filters: ILPF



a b c

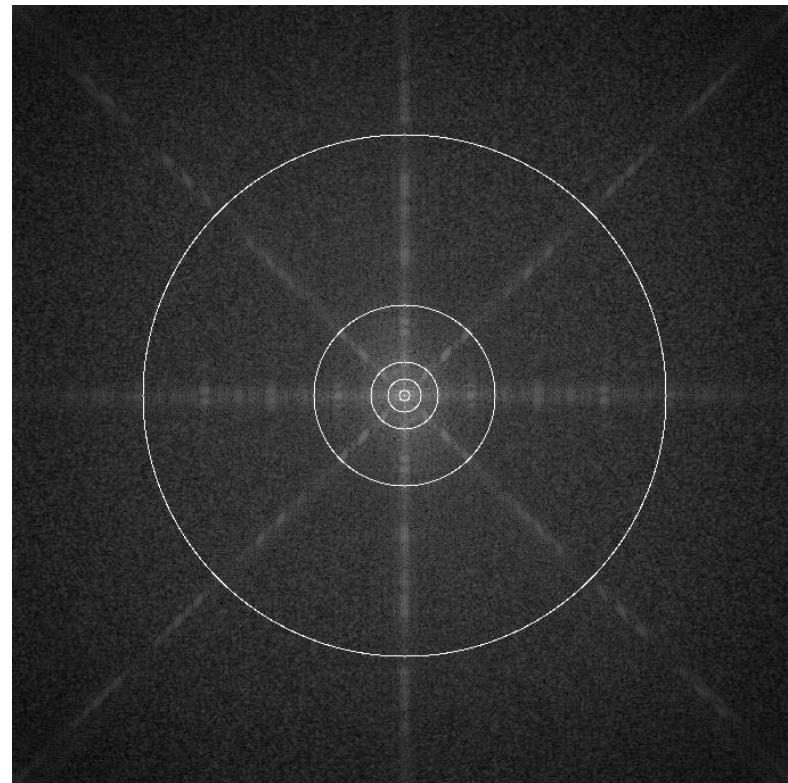
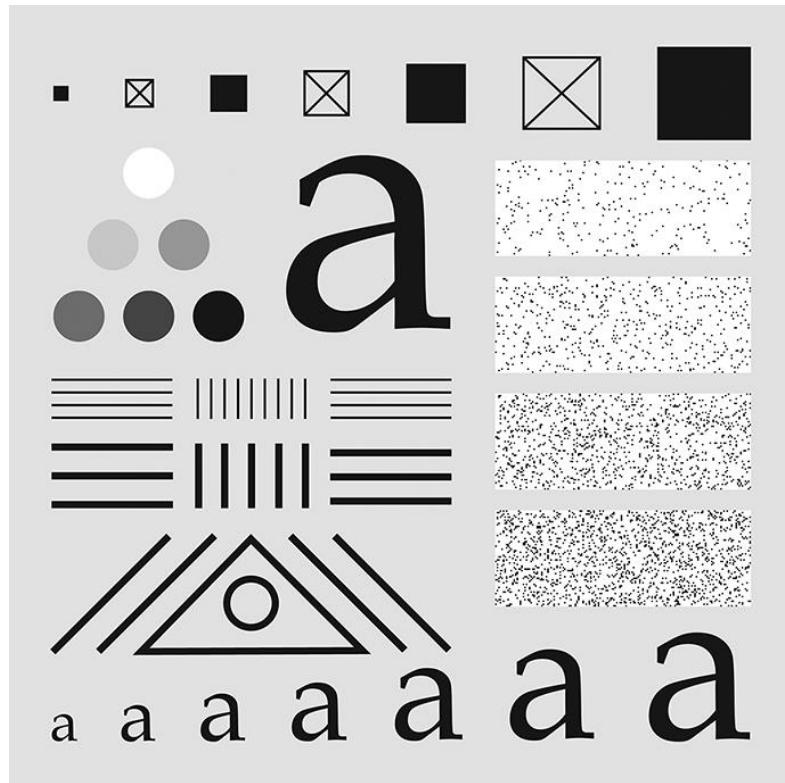
FIGURE 4.39 (a) Perspective plot of an ideal lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross section.

- One way to establish a set of standard cutoff frequency loci is to compute circles that enclose specified amounts of total image power P_T
- A circle of radius D_0 encloses percent of the power

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

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

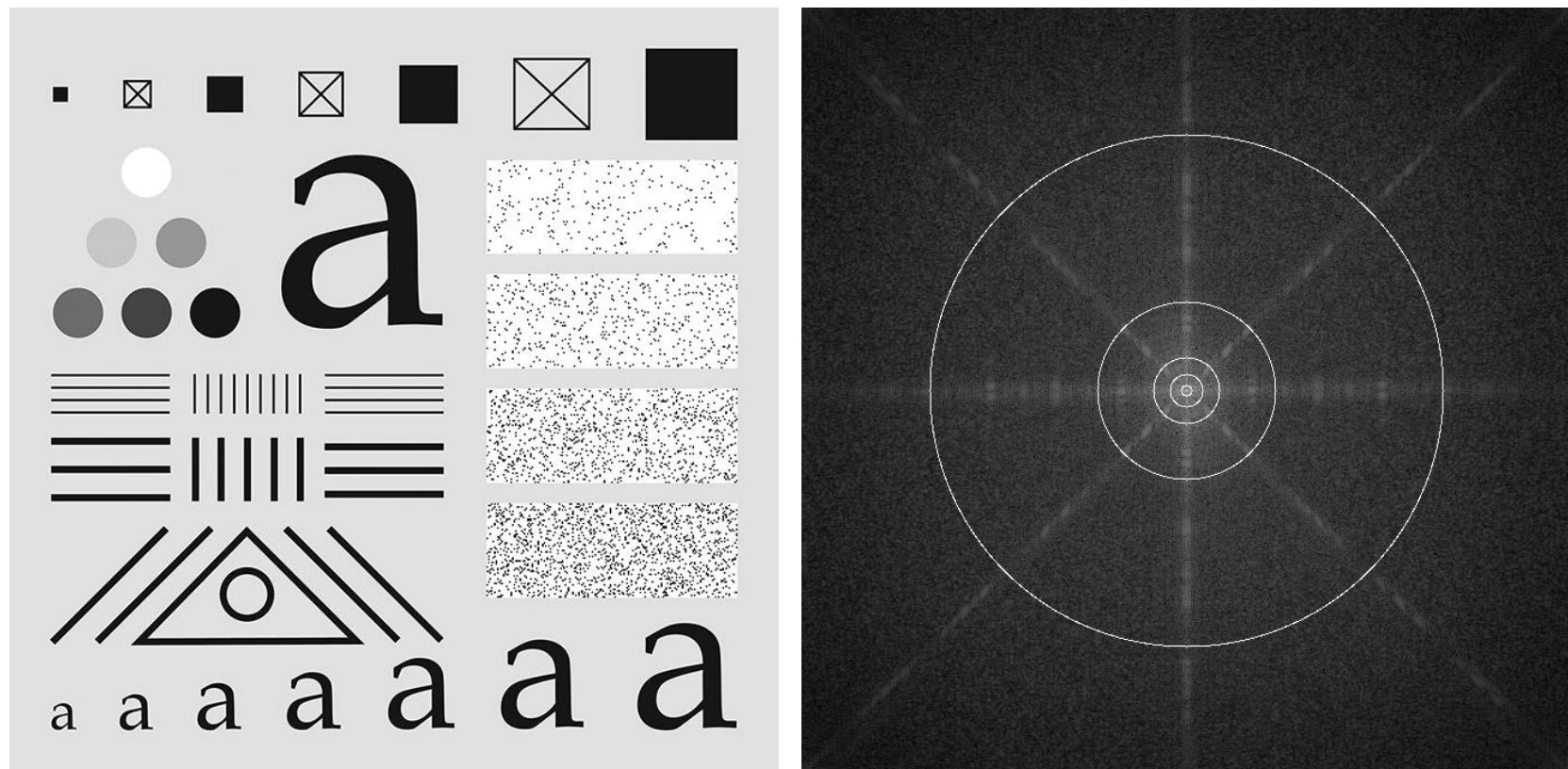
Ideal Low Pass Filter (cont...)



An image, its Fourier spectrum and a series of ideal low pass filters of radii = 10, 30, 60, 160 and 460 superimposed on top of it.



ILPF Filtering Example



a b

FIGURE 4.40 (a) Test pattern of size 688×688 pixels, and (b) its spectrum. The spectrum is double the image size as a result of padding, but is shown half size to fit. The circles have radii of 10, 30, 60, 160, and 460 pixels with respect to the full-size spectrum. The radii enclose 86.9, 92.8, 95.1, 97.6, and 99.4% of the padded image power, respectively.

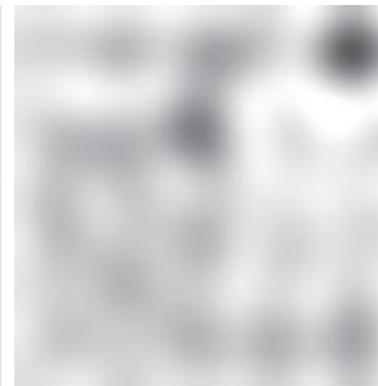


Ideal Low Pass Filter

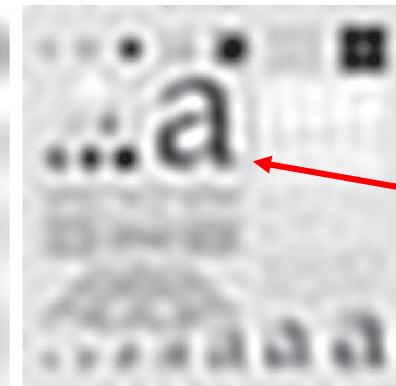
Original
image



ILPF of
radius 10



ILPF of
radius 30



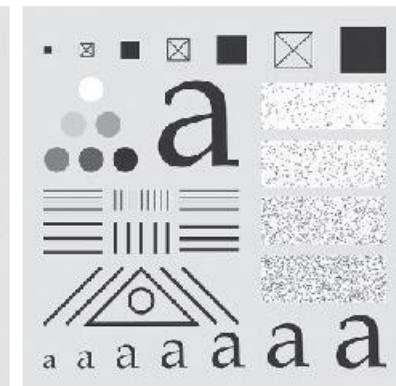
- Notice the ringing effect
- Why? → See Next Slide

ILPF of
radius 60



ILPF of radius 160

a b c
d e f



ILPF of
radius 460

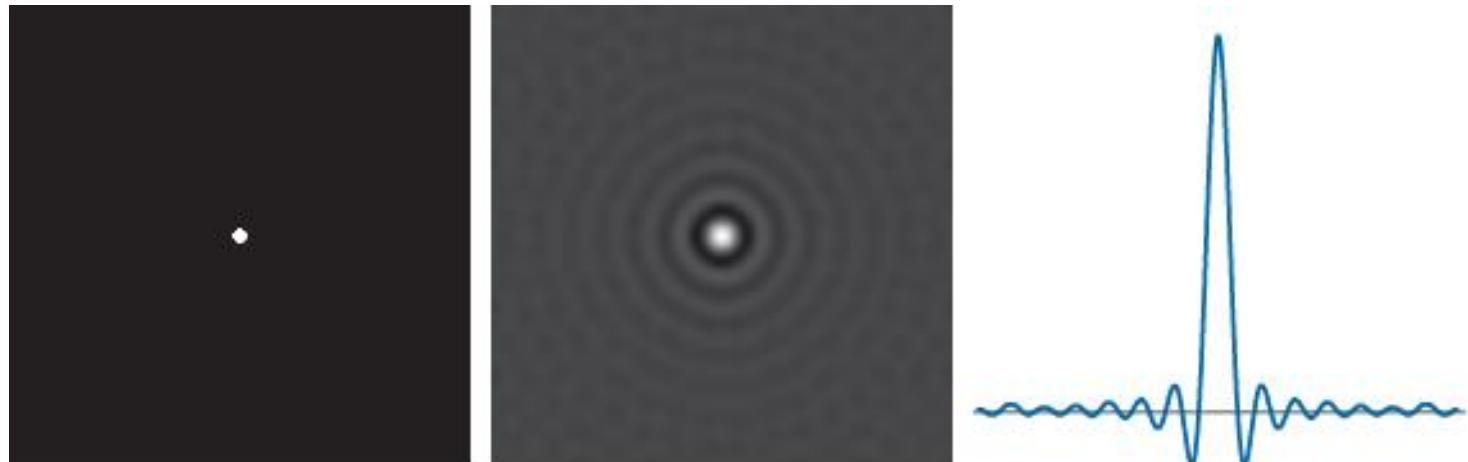
FIGURE 4.41 (a) Original image of size 688×688 pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively. We used mirror padding to avoid the black borders characteristic of zero padding, as illustrated in Fig. 4.31(c).

- ILPF in the spatial domain is a sinc function that has to be truncated → produces ringing effect
 - The main lobe causes blurring
 - Side lobes cause ringing

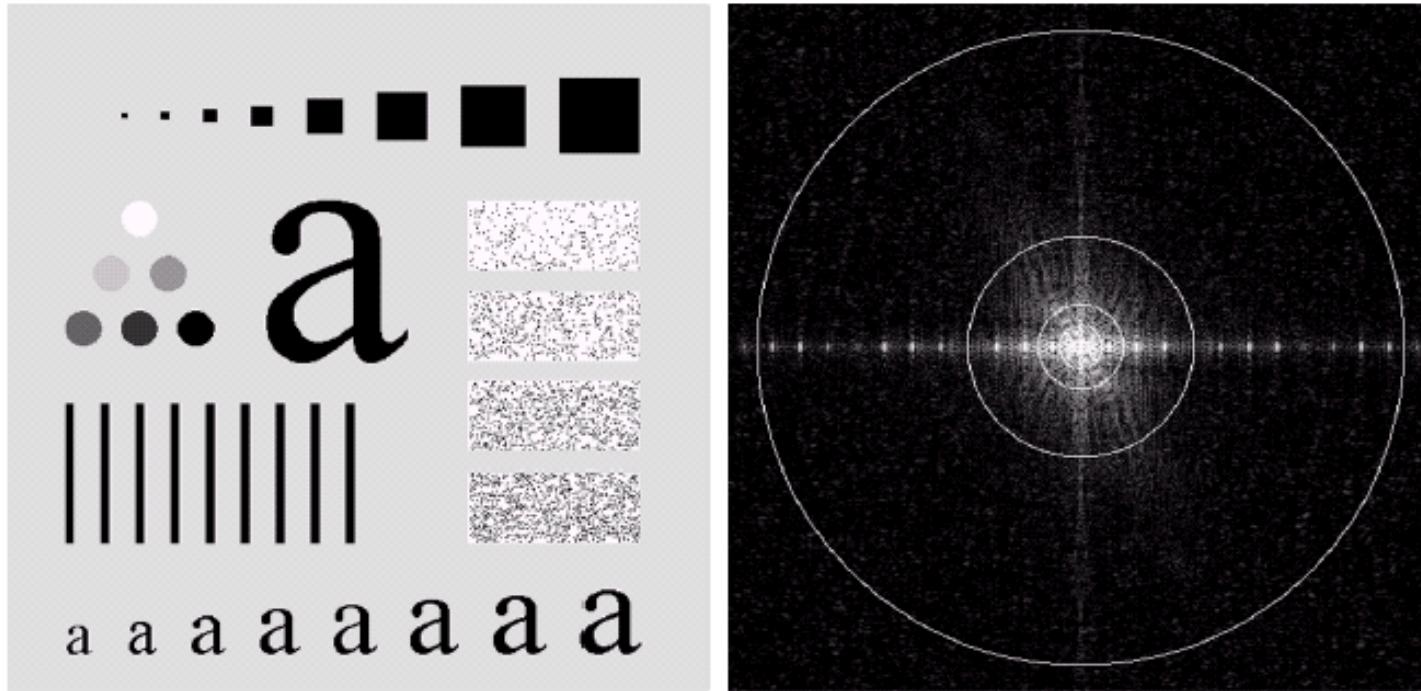
a b c

FIGURE 4.42

- (a) Frequency domain ILPF transfer function.
(b) Corresponding spatial domain kernel function.
(c) Intensity profile of a horizontal line through the center of (b).



Ideal Low Pass Filter (cont...)

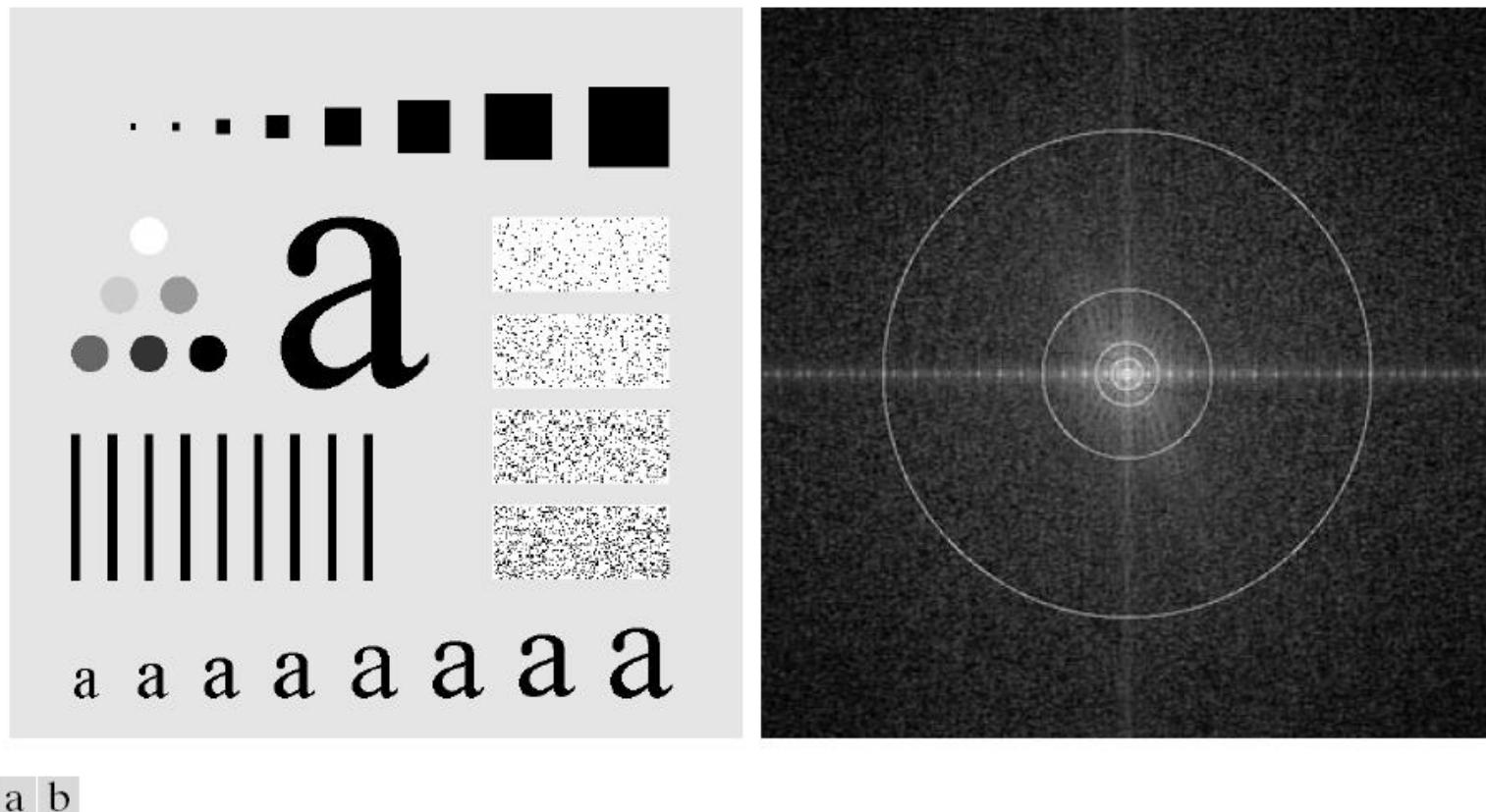


An image, its Fourier spectrum and a series of ideal low pass filters of radii = 10, 30, 60, 160 and 460 superimposed on top of it.

3rd Edition



ILPF Filtering Example



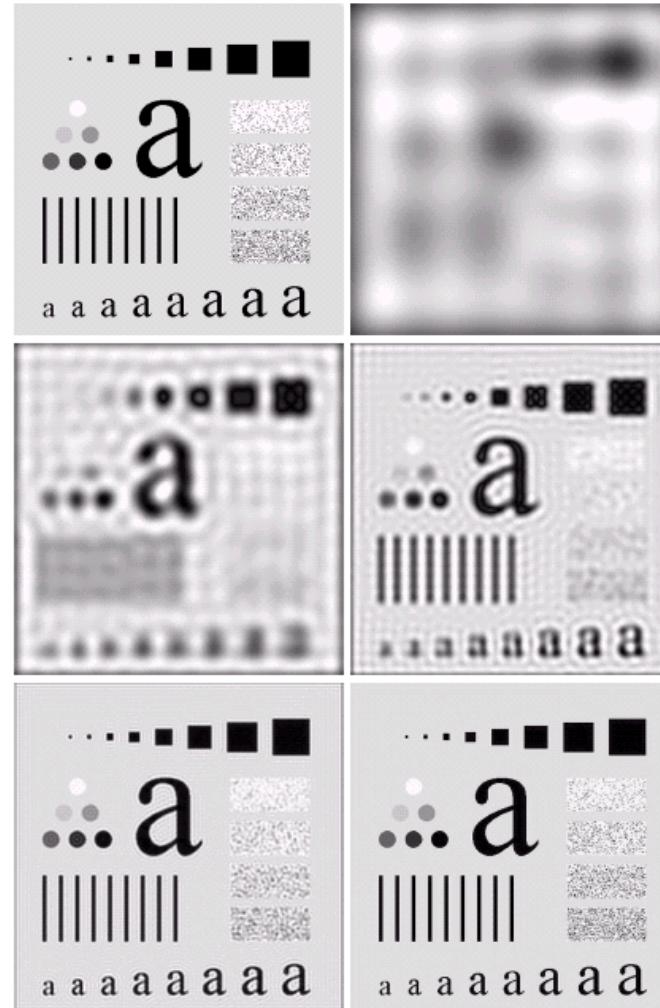
a | b

FIGURE 4.41 (a) Test pattern of size 688×688 pixels, and (b) its Fourier spectrum. The spectrum is double the image size due to padding but is shown in half size so that it fits in the page. The superimposed circles have radii equal to 10, 30, 60, 160, and 460 with respect to the full-size spectrum image. These radii enclose 87.0, 93.1, 95.7, 97.8, and 99.2% of the padded image power, respectively.

3rd Edition

Ideal Low Pass Filter

Original
image



ILPF of
radius 10

ILPF of
radius 30

ILPF of
radius 60

ILPF of
radius 160

ILPF of
radius 460

FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

3rd Edition

- **Gaussian Lowpass Filters (GLPF)**

- GLPF with cutoff frequency at a distance D_0
- Unlike ILPF, the GLPF transfer function **does not have a sharp discontinuity** that gives a clear cutoff between passed and filtered frequencies
- **Smooth transition** between low and high frequencies
- **Less Ringing than Ideal Low Pass Filter (ILPF)**

Gaussian Lowpass Filters (GLPF) in two dimensions,

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

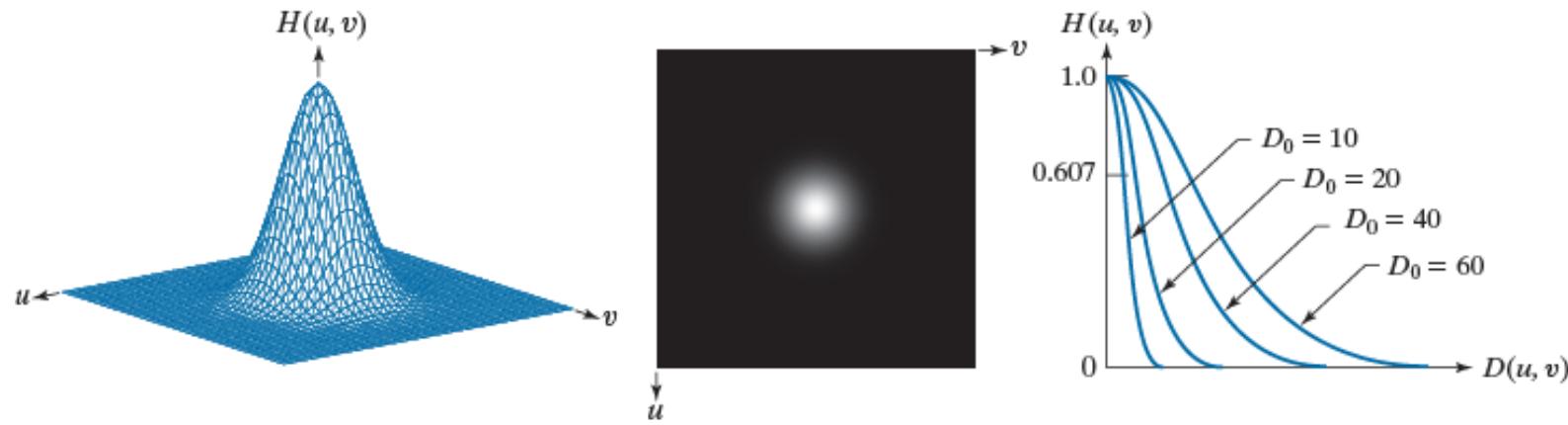
$D(u, v)$: Distance from center

By letting $\sigma = D_0$

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



Image Smoothing Using Frequency Domain Filters: GLPF

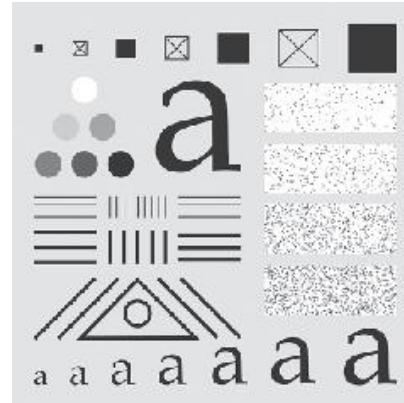


a b c

FIGURE 4.43 (a) Perspective plot of a GLPF transfer function. (b) Function displayed as an image. (c) Radial cross sections for various values of D_0 .

Gaussian Lowpass Filters (cont...)

Original image



Gaussian $D_0 = 10$

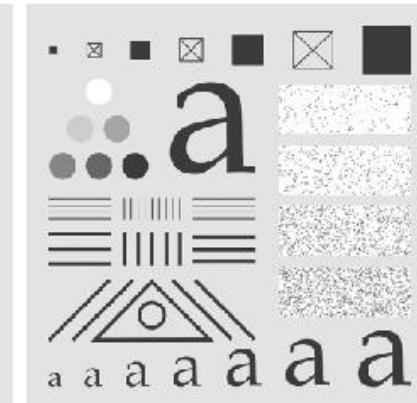


Gaussian $D_0 = 30$

Gaussian $D_0 = 60$



Gaussian $D_0 = 460$



Gaussian $D_0 = 160$

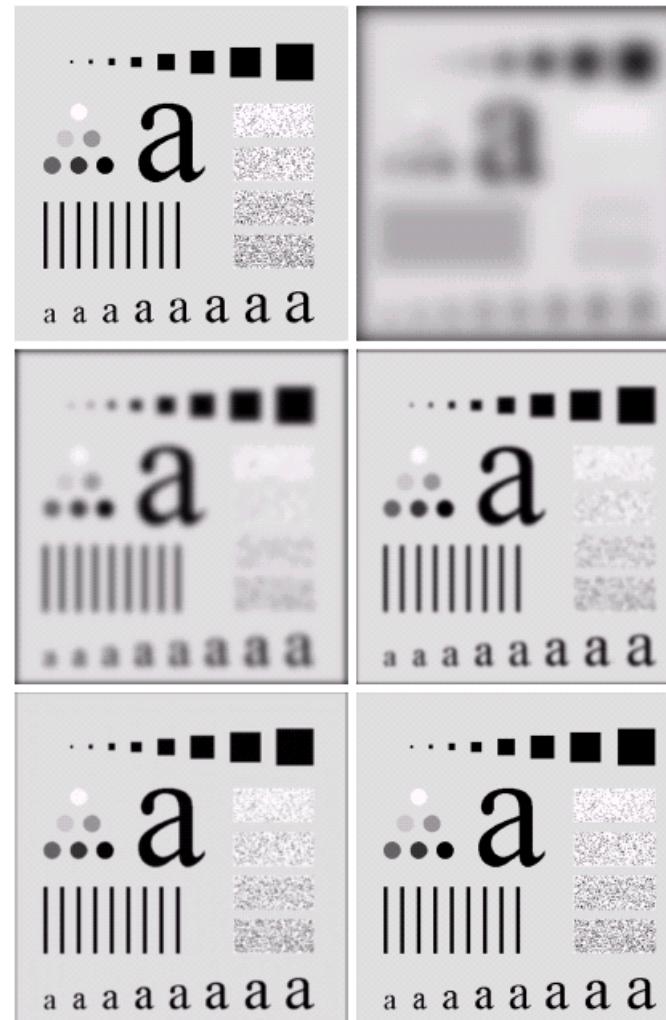
a	b	c
d	e	f

Less ringing than ILPF but also less smoothing

FIGURE 4.44 (a) Original image of size 688×688 pixels. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.40. Compare with Fig. 4.41. We used mirror padding to avoid the black borders characteristic of zero padding.

Gaussian Lowpass Filters (cont...)

Original image



Gaussian $D_0=30$

Gaussian $D_0=10$

Gaussian $D_0=60$

Gaussian $D_0=460$

Gaussian $D_0=160$

Less ringing than
ILPF but also less
smoothing

FIGURE 4.48 (a) Original image. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Figs. 4.42 and 4.45.

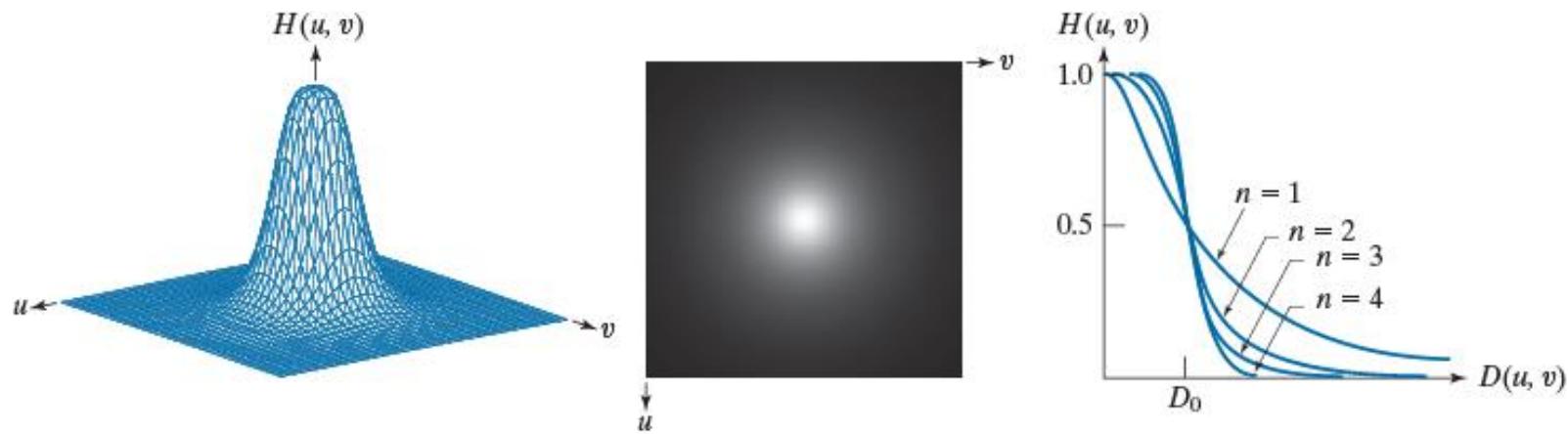
3rd Edition

4.8.2 Image Smoothing Using Frequency Domain Filters: BLPF

Butterworth Lowpass Filters (BLPF) of order n and with cutoff frequency D_0

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

- Monotonic Decreasing
- No Ripples



a b c

FIGURE 4.45 (a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross sections of BLPFs of orders 1 through 4.

- Butterworth Lowpass Filters (BLPF)
 - BLPF of order n , and with cutoff frequency at a distance D_0
 - Unlike ILPF, the BLPF transfer function **does not have a sharp discontinuity** that gives a clear cutoff between passed and filtered frequencies
 - Smooth transition between low and high frequencies
 - No Ringing for order = 1 and 2 (preferable)



- Butterworth Lowpass Filters (BLPF)

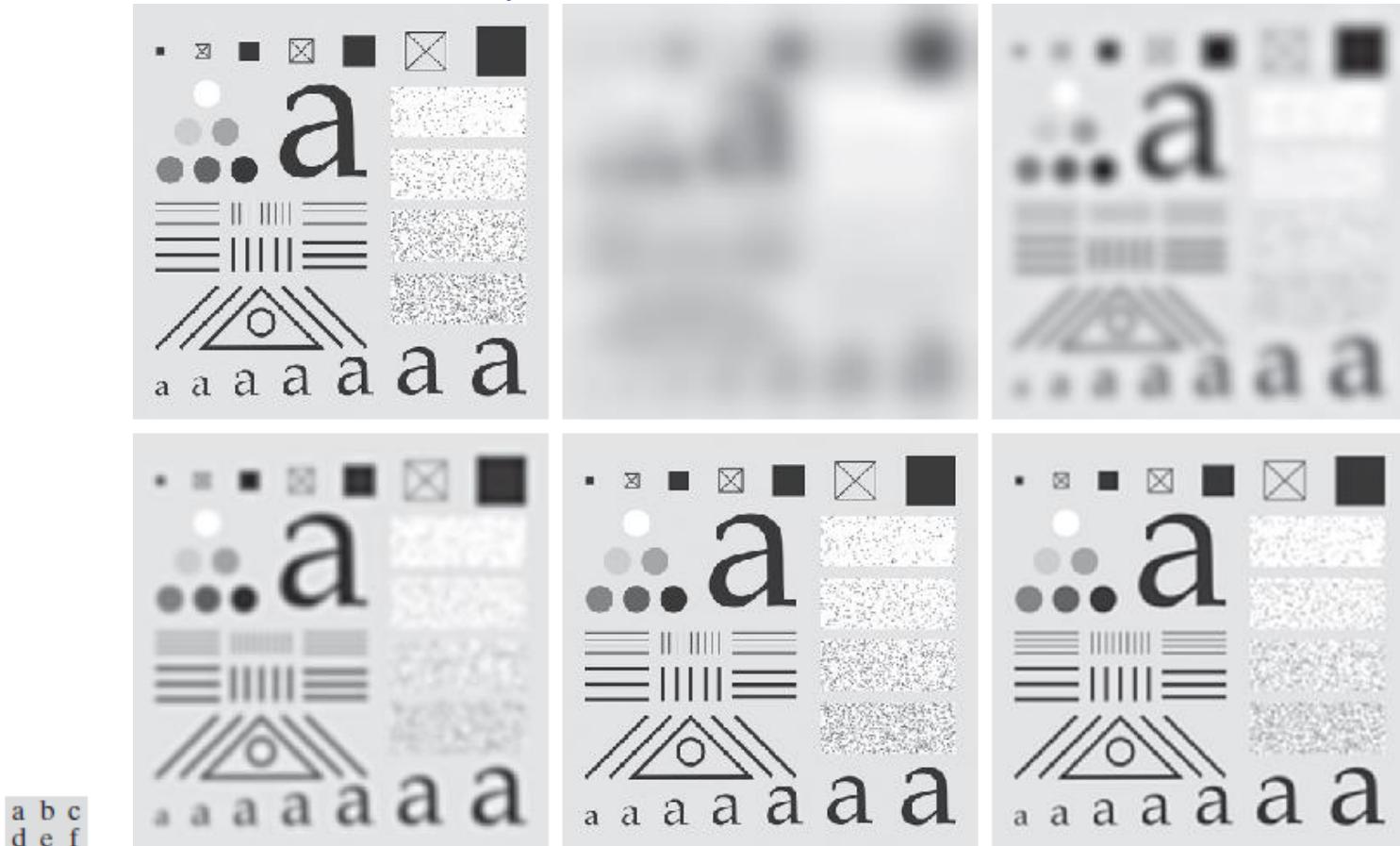


FIGURE 4.46 (a) Original image of size 688×688 pixels. (b)–(f) Results of filtering using BLPFs with cutoff frequencies at the radii shown in Fig. 4.40 and $n = 2.25$. Compare with Figs. 4.41 and 4.44. We used mirror padding to avoid the black borders characteristic of zero padding.

- See Figure 3.45 and its discussion regarding mirror-padding.

The Spatial Representation of BLPF

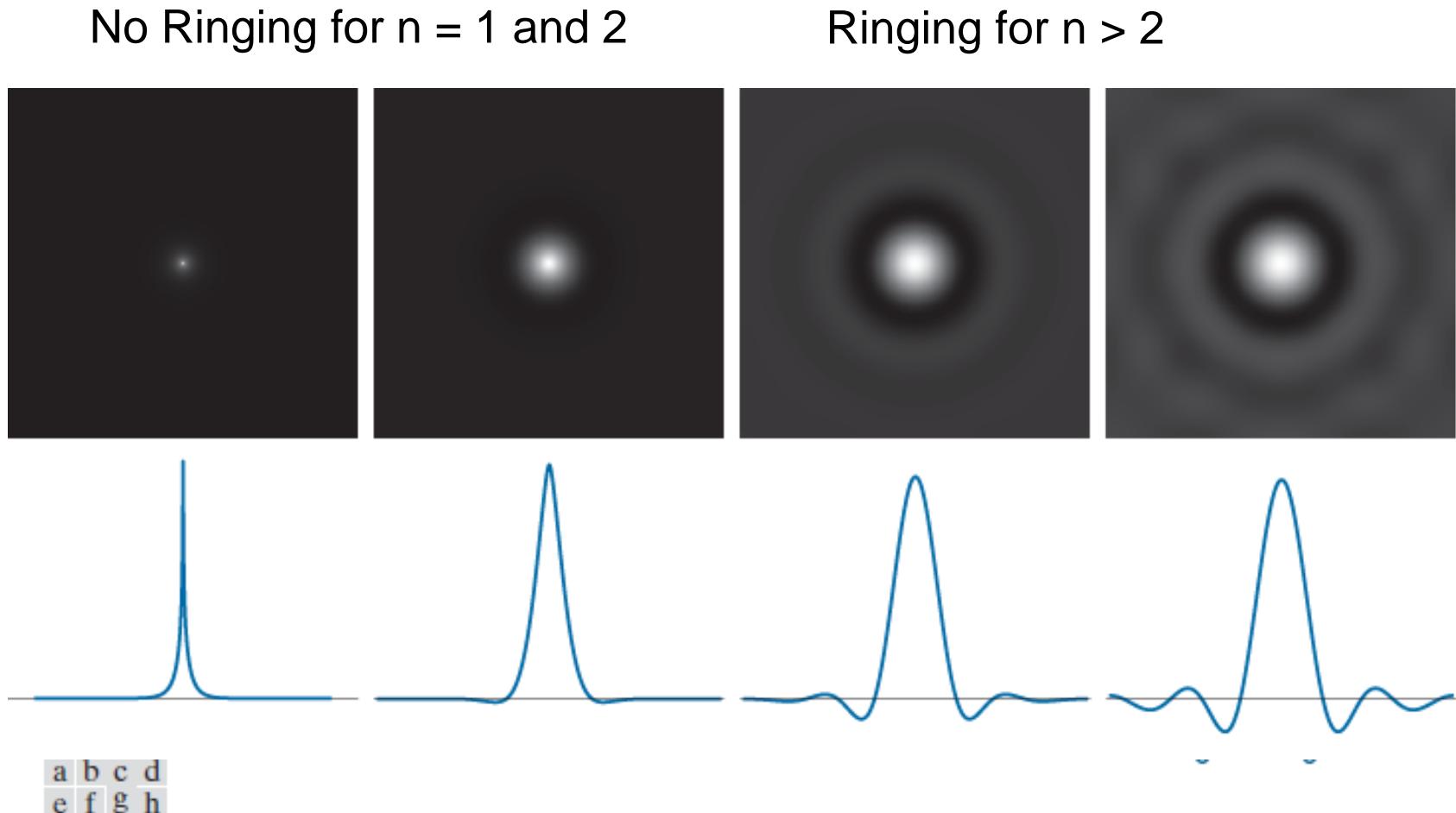


FIGURE 4.47 (a)–(d) Spatial representations (i.e., spatial kernels) corresponding to BLPF transfer functions of 1000×1000 pixels, cut-off frequency of 5, and order 1, 2, 5, and 20, respectively. (e)–(h) Corresponding intensity profiles through the center of the filter functions.



Lowpass Filters Equations

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

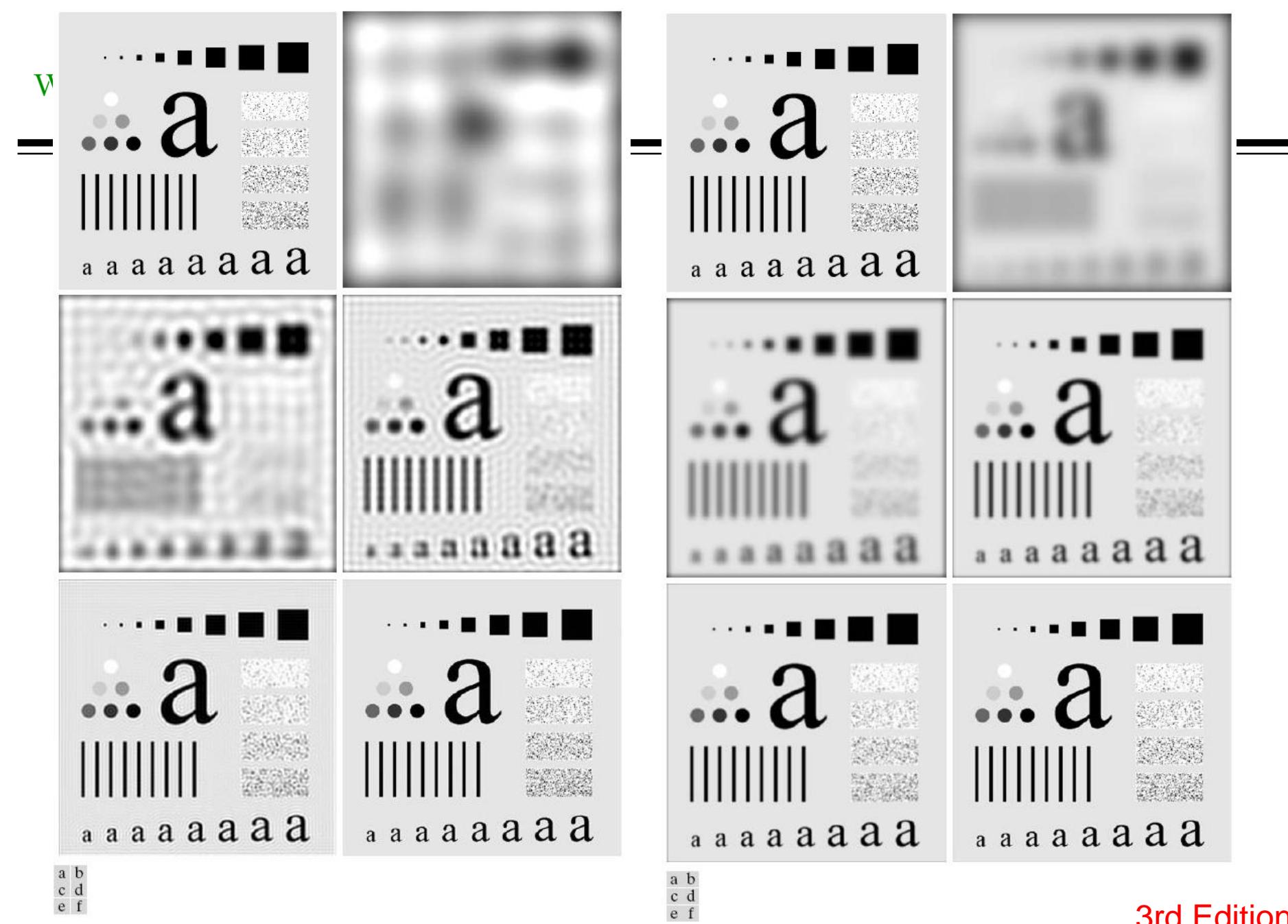


FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

EE-'

ib K. Shaw

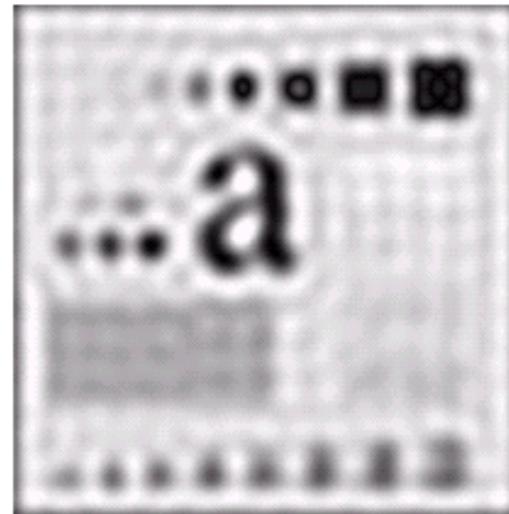
3rd Edition

FIGURE 4.48 (a) Original image. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Figs. 4.42 and 4.45.

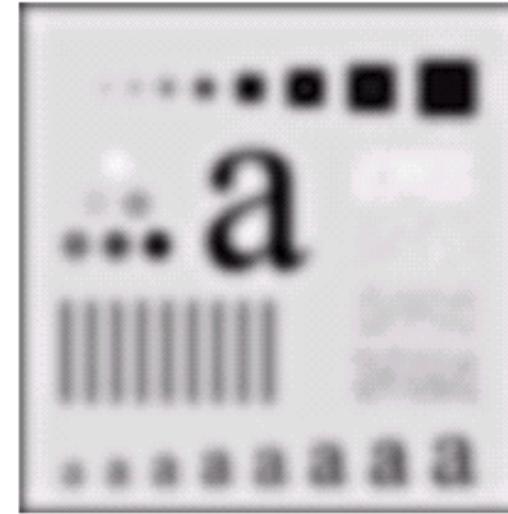
Engineering

Lowpass Filters Compared

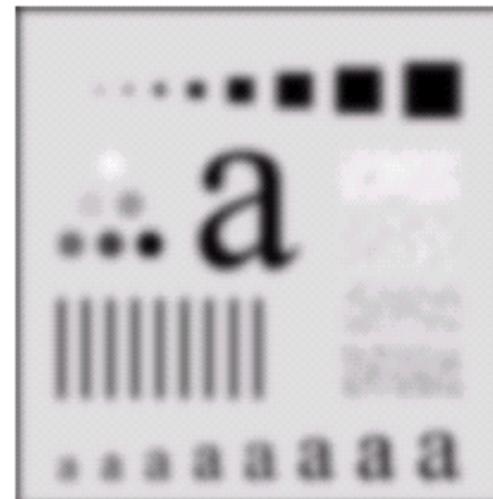
ILPF: $D_0=30$



BLPF: $n=2$, $D_0=30$



Gaussian: $D_0= 30$



3rd Edition

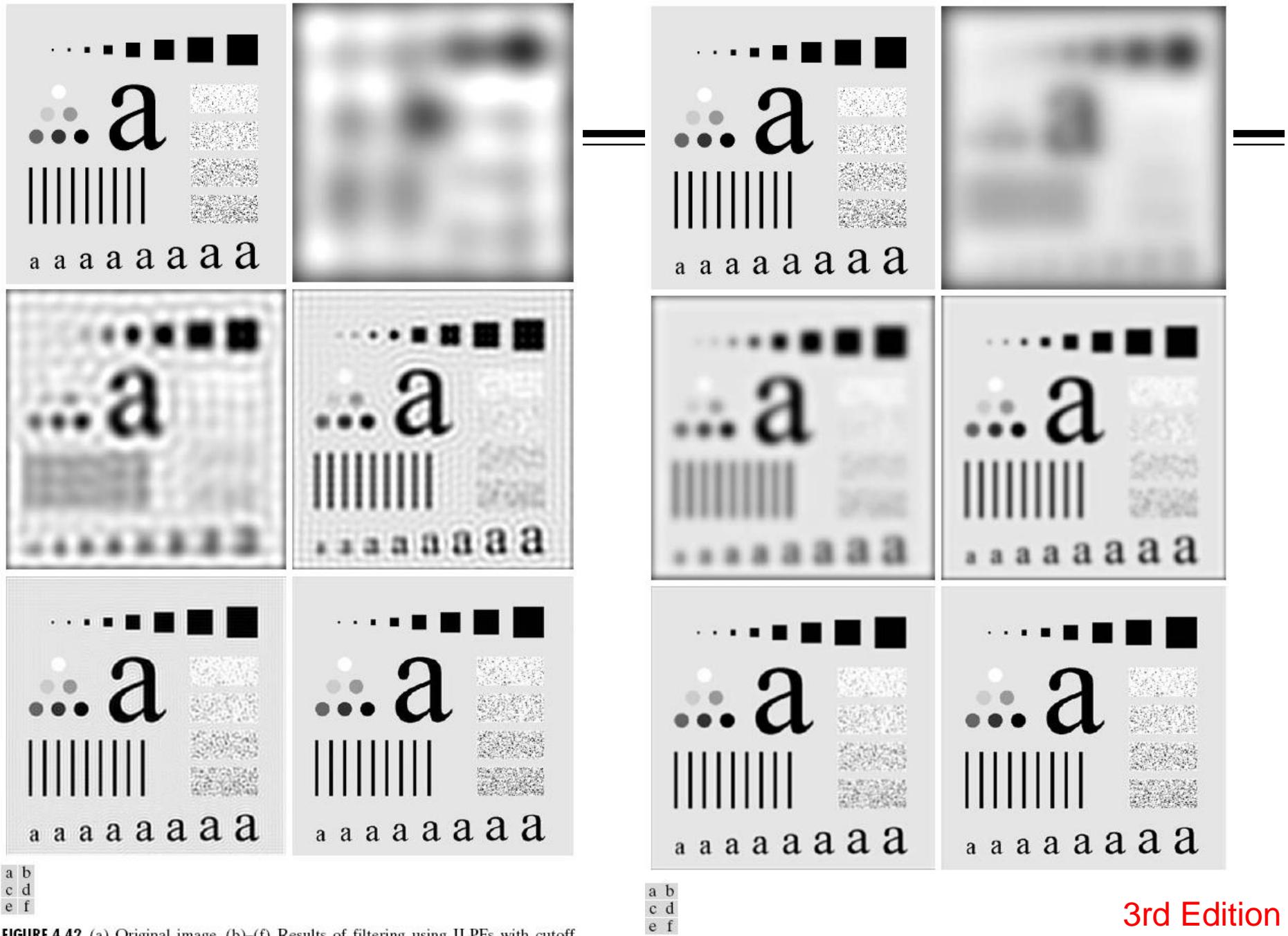


FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

3rd Edition

FIGURE 4.45 (a) Original image. (b)–(f) Results of filtering using BLPFs of order 2, with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Fig. 4.42.

Examples of Smoothing by GLPF (1)

a b

FIGURE 4.48

(a) Sample text of low resolution (note the broken characters in the magnified view).
(b) Result of filtering with a GLPE, showing that gaps in the broken characters were joined.

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.



Examples of smoothing by GLPF (2)

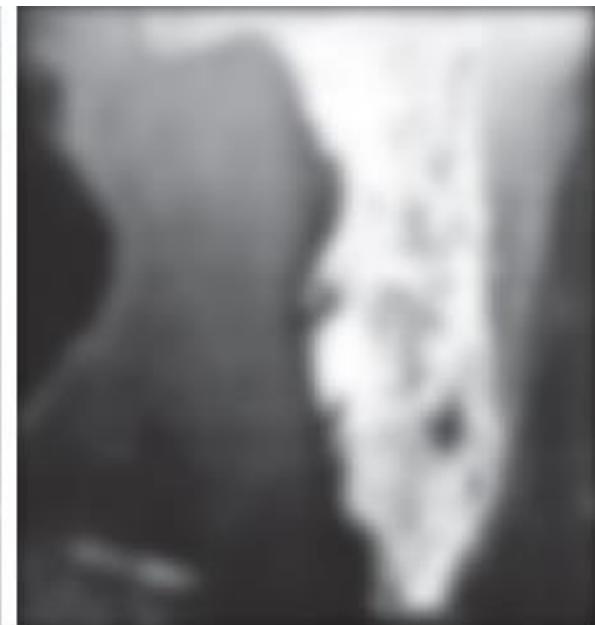
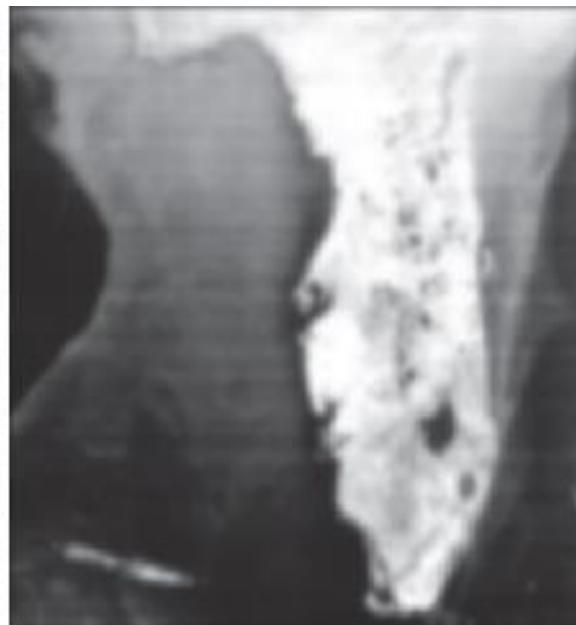
- Different lowpass Gaussian filters used to remove blemishes in a photograph.



FIGURE 4.49 (a) Original 785×732 image. (b) Result of filtering using a GLPF with $D_0 = 150$. (c) Result of filtering using a GLPF with $D_0 = 130$. Note the reduction in fine skin lines in the magnified sections in (b) and (c).



Examples of smoothing by GLPF (3)



a b c

FIGURE 4.50 (a) 808×754 satellite image showing prominent horizontal scan lines. (b) Result of filtering using a GLPF with $D_0 = 50$. (c) Result of using a GLPF with $D_0 = 20$. (Original image courtesy of NOAA.)

4.9 Image Sharpening using Highpass Filters

- Edges and fine details in images are associated with high frequency components
- **Highpass filters** - Only pass the high frequencies, and suppress the low frequencies
- High frequency regions are precisely the reverse of low pass filter regions in frequency domain. So:

$$H_{HP}(u, v) = 1 - H_{LP}(u, v)$$

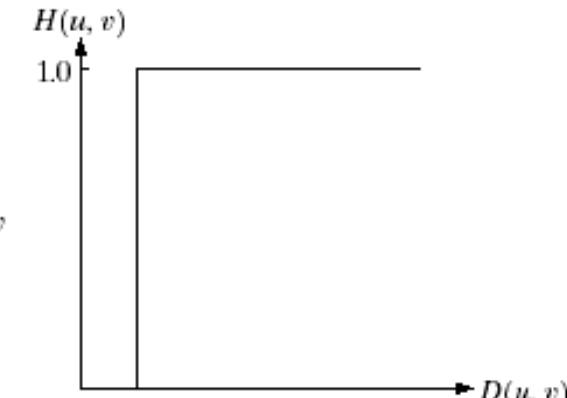
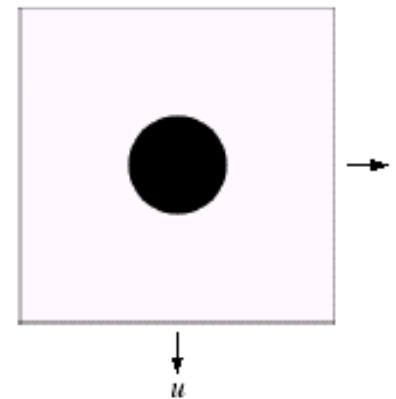
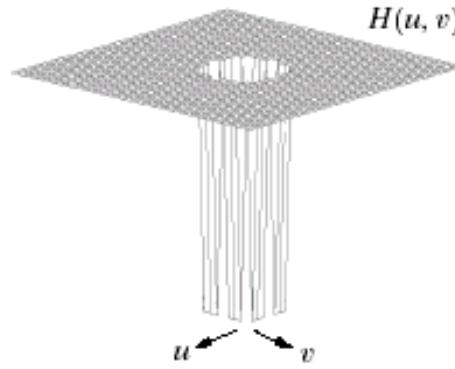
- Three types (Similar to Lowpass)
 - Ideal
 - Butterworth
 - Gaussian

4.9.1 Ideal High Pass Filters

- The ideal high pass filter is given by:

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

- D_0 is the cut off distance as before.



4.9.2 Image Sharpening Using Frequency Domain Filters

A 2-D Butterworth highpass filter (BHPL) is defined as

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

A 2-D Gaussian highpass filter (GHPL) is defined as

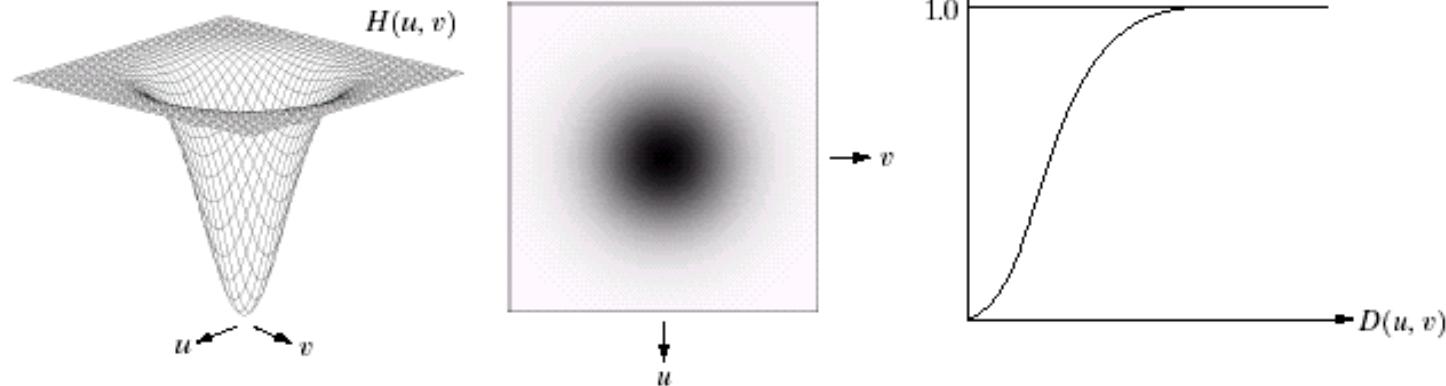
$$H(u, v) = 1 - e^{-D^2(u, v)/2D_0^2}$$

4.9.3 Gaussian High Pass Filters

- The Gaussian high pass filter is given as:

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

- D_0 is the cut off distance as before.



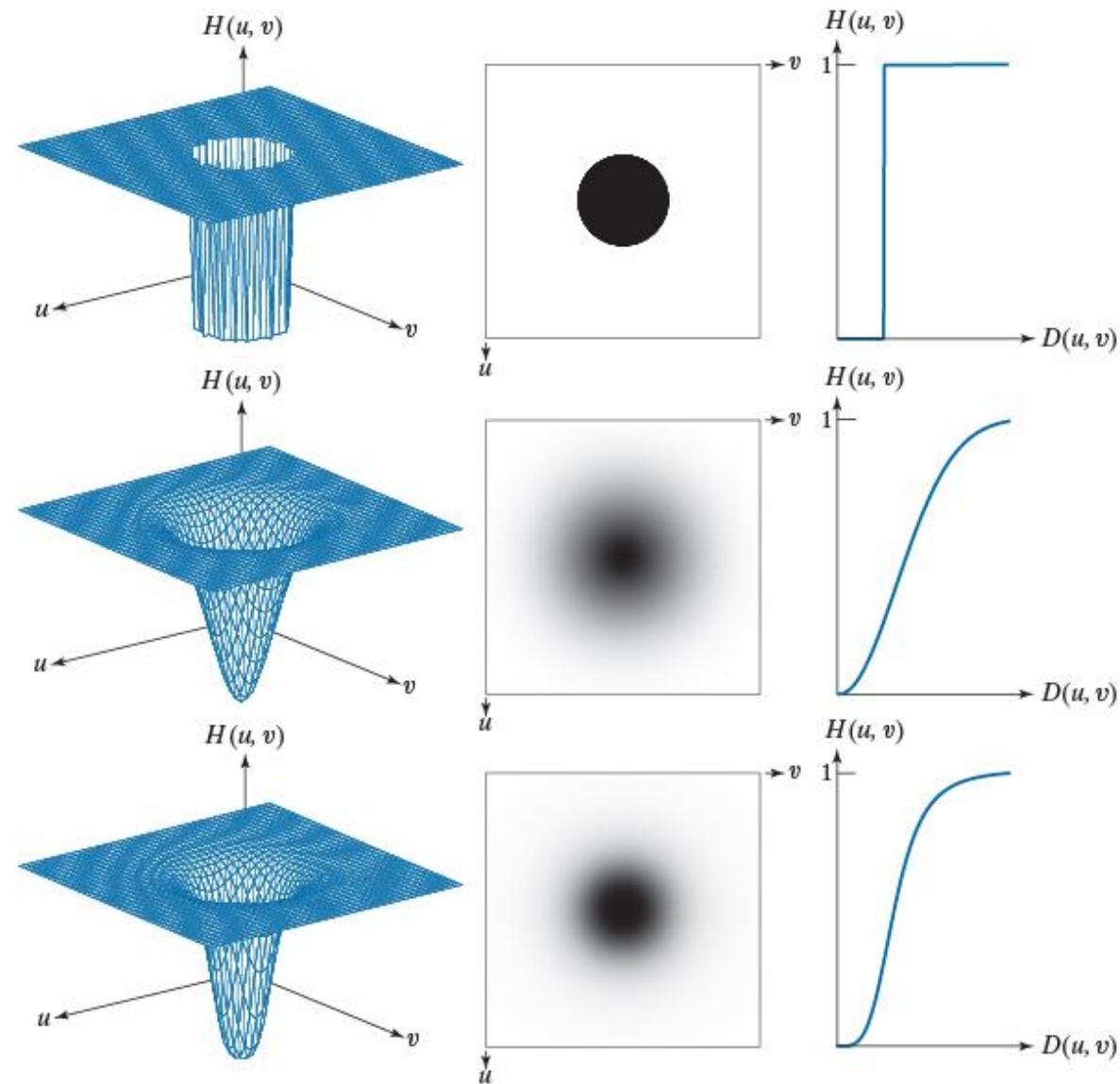


Highpass Filters

a | b c
d | e f
g | h i

FIGURE 4.51

Top row:
Perspective plot,
image, and, radial
cross section of
an IHPF transfer
function. Middle
and bottom
rows: The same
sequence for
GHPF and BHPF
transfer functions.
(The thin image
borders were
added for clarity.
They are not part
of the data.)



Highpass Filter Spatial Kernels

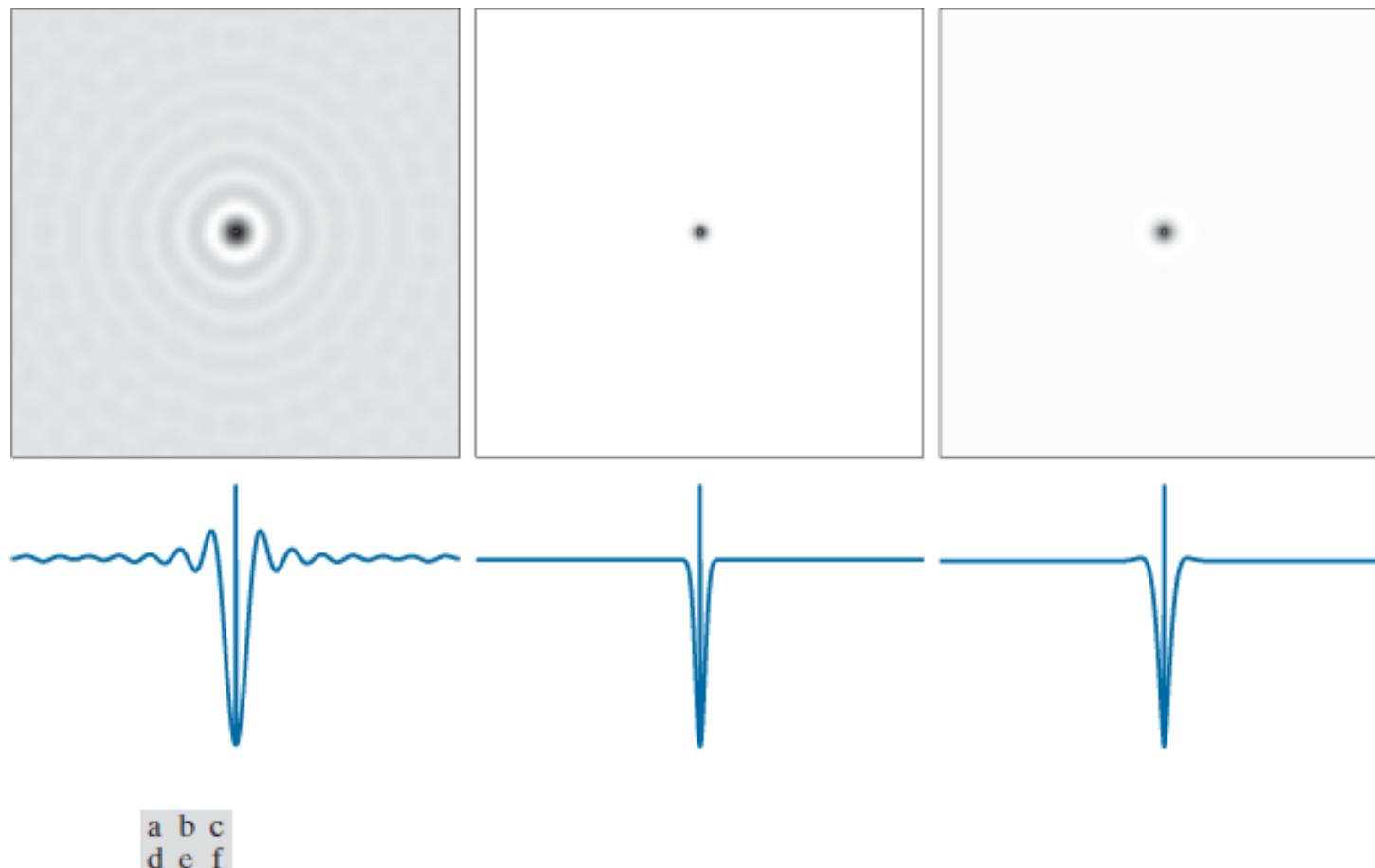


FIGURE 4.52 (a)–(c): Ideal, Gaussian, and Butterworth highpass spatial kernels obtained from IHPF, GHPF, and BHPF frequency-domain transfer functions. (The thin image borders are not part of the data.) (d)–(f): Horizontal intensity profiles through the centers of the kernels.



- Highpass Filters Equations

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



Highpass Filter Comparison

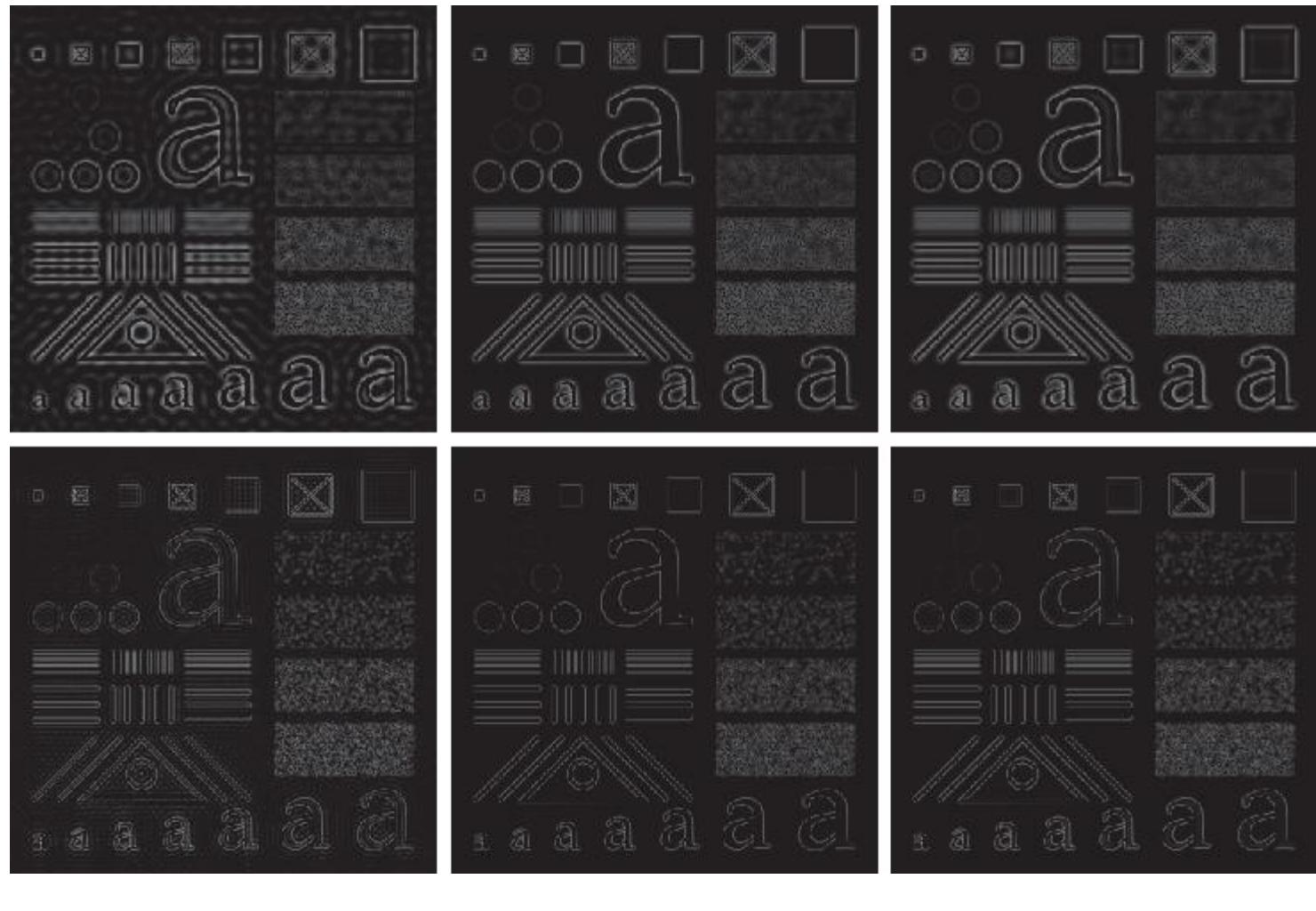


FIGURE 4.53 Top row: The image from Fig. 4.40(a) filtered with IHPF, GHPF, and BHPF transfer functions using $D_0 = 60$ in all cases ($n = 2$ for the BHPF). Second row: Same sequence, but using $D_0 = 160$.



Highpass Filter Comparison



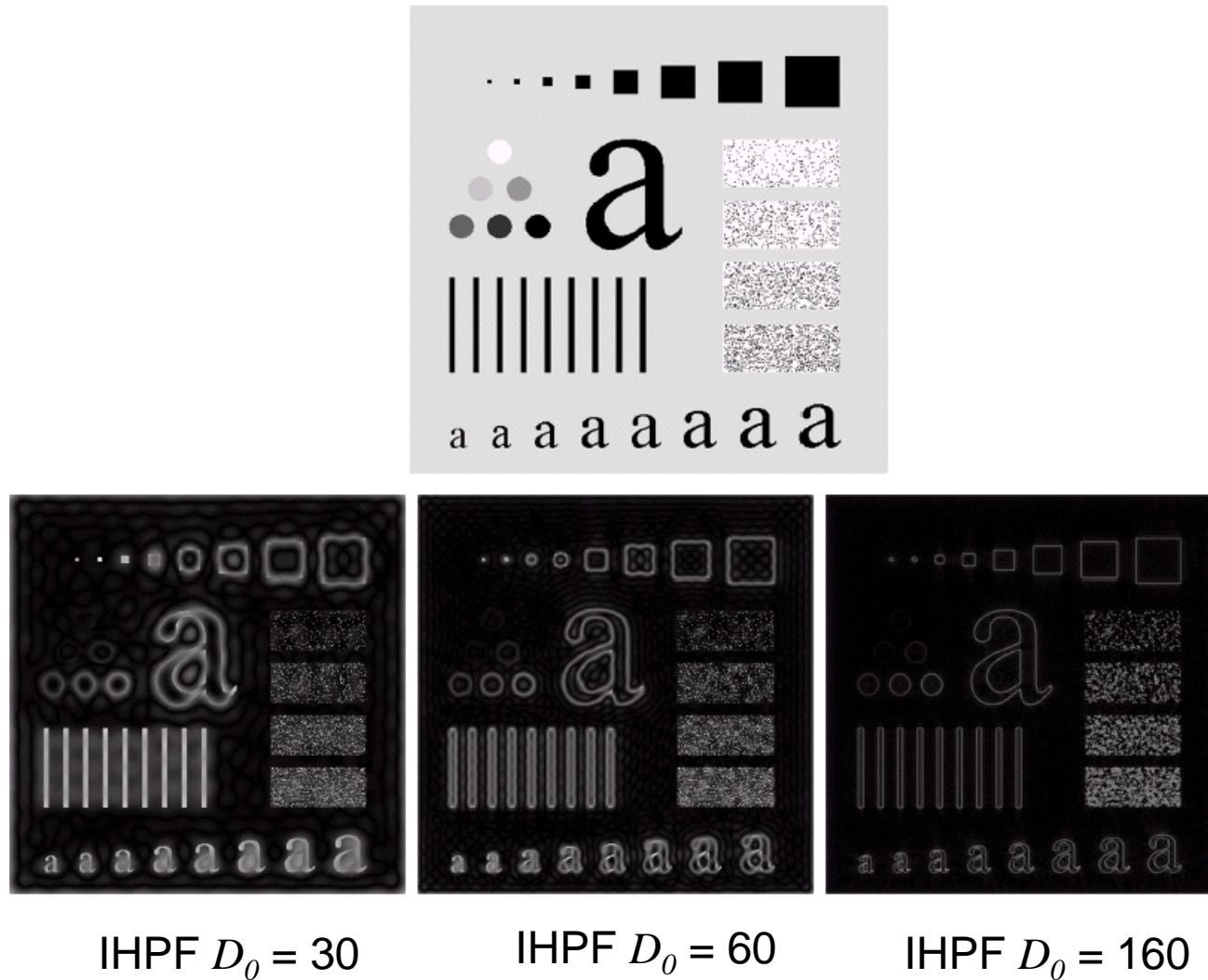
a b c

FIGURE 4.54 The images from the second row of Fig. 4.53 scaled using Eqs. (2-31) and (2-32) to show both positive and negative values.

$$g_m = g - \min(f) \quad (2-31)$$

$$g_s = K [g_m / \max(g_m)] \quad (2-32)$$

Ideal High Pass Filters (cont...)



a b c

FIGURE 4.54 Results of highpass filtering the image in Fig. 4.41(a) using an IHPF with $D_0 = 30, 60$, and 160 .

3rd Edition

Filtering Results by BHPF

BHPF n=2, $D_0 = 30$



BHPF n=2, $D_0 = 60$



BHPF n=2, $D_0 = 160$



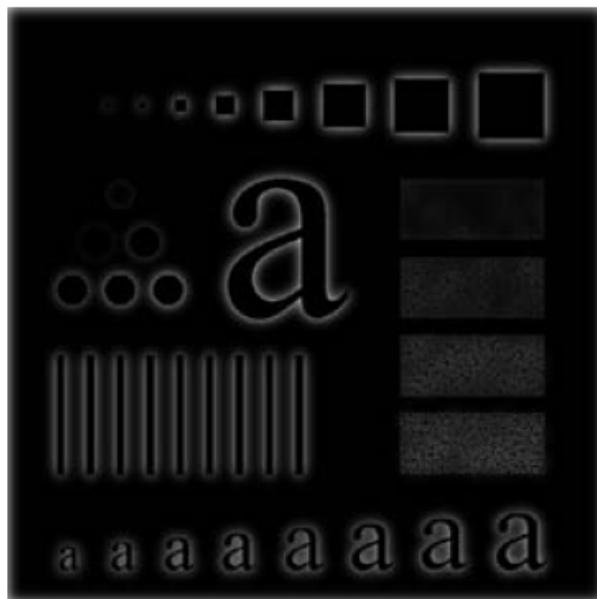
a b c

FIGURE 4.55 Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with $D_0 = 30, 60$, and 160, corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPE.

3rd Edition

Filtering Results by GHPF

Gaussian HPF, $D_0 = 30$



Gaussian HPF, $D_0 = 60$



Gaussian HPF, $D_0 = 160$

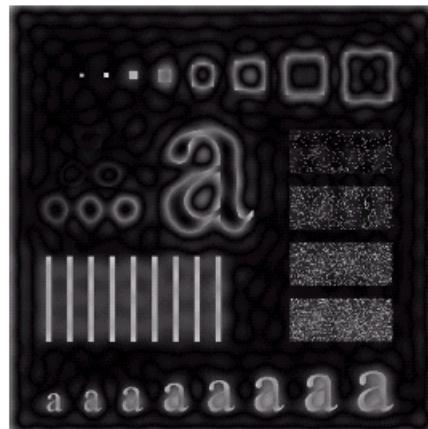
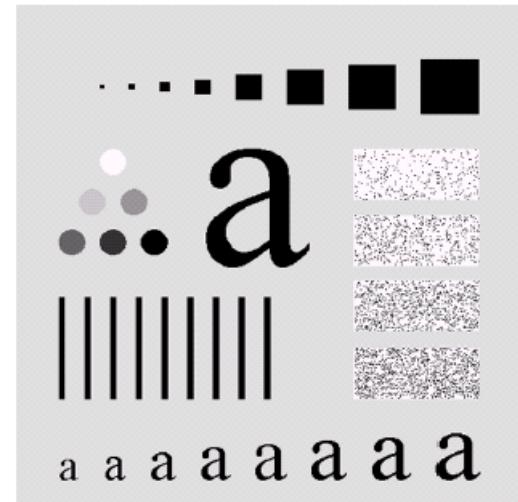


a b c

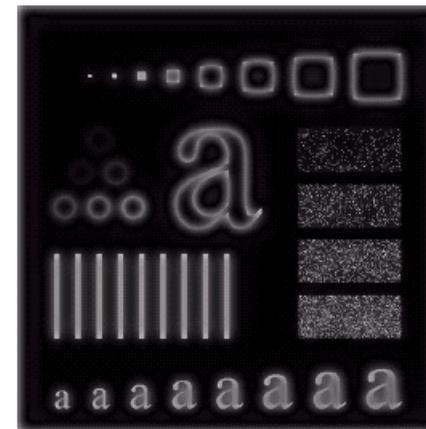
FIGURE 4.56 Results of highpass filtering the image in Fig. 4.41(a) using a GHPF with $D_0 = 30, 60$, and 160 , corresponding to the circles in Fig. 4.41(b). Compare with Figs. 4.54 and 4.55.

3rd Edition

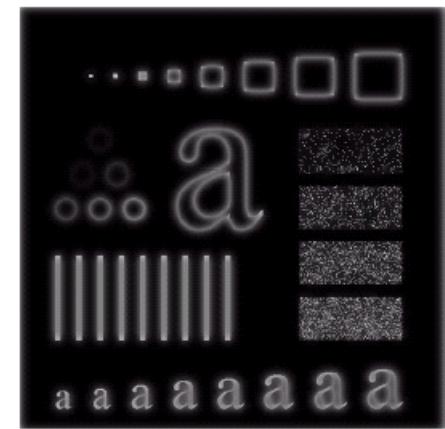
Highpass Filter Comparison



IHPF $D_0 = 30$

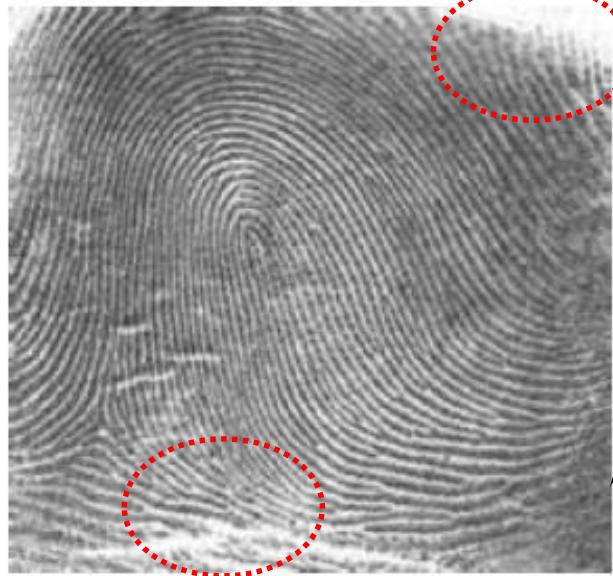


BHPF $n=2, D_0 = 30$

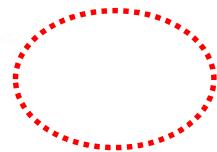
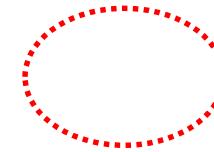


Gaussian HPF, $D_0 = 30$
3rd Edition

Using Highpass Filtering and Threshold for Image Enhancement



BHPF
(order 4 with a cutoff
frequency 50)



a b c

FIGURE 4.55 (a) Smudged thumbprint. (b) Result of highpass filtering (a). (c) Result of thresholding (b). (Original image courtesy of the U.S. National Institute of Standards and Technology.)

Using Highpass Filtering and Threshold for Image Enhancement

BHPF

(order 4 with a cutoff
frequency 50)



a b c

FIGURE 4.55 (a) Smudged thumbprint. (b) Result of highpass filtering (a). (c) Result of thresholding (b). (Original image courtesy of the U.S. National Institute of Standards and Technology.)

4.9.4 The Laplacian in the Frequency Domain

- Laplacian can be implemented in the frequency domain using the filter

$$H(u, v) = -4\pi^2(u^2 + v^2)$$

$$\begin{aligned} H(u, v) &= -4\pi^2 \left[(u - P/2)^2 + (v - Q/2)^2 \right] \\ &= -4\pi^2 D^2(u, v) \end{aligned}$$

The Laplacian image

$$\nabla^2 f(x, y) = \mathcal{I}^{-1} \{ H(u, v) F(u, v) \}$$

Enhancement is obtained

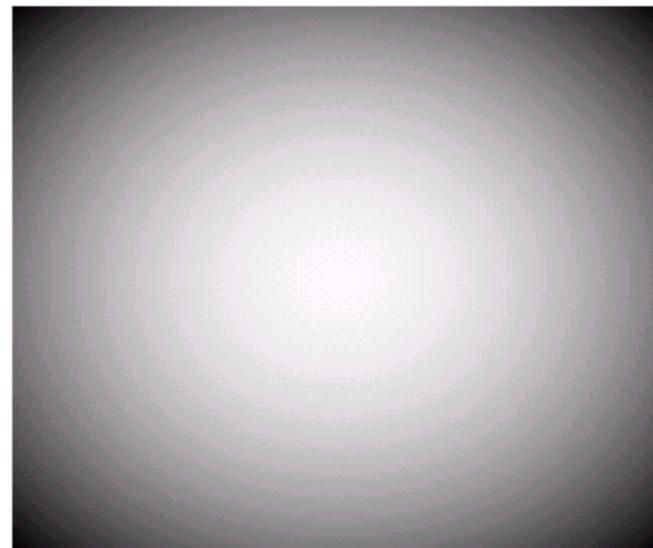
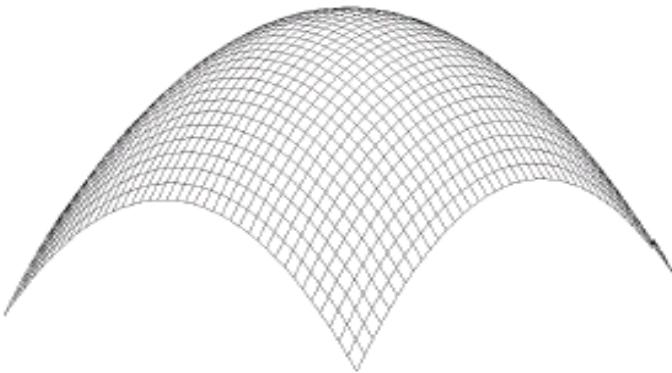
$$g(x, y) = f(x, y) + c \nabla^2 f(x, y) \quad c = -1$$

The enhanced image

$$\begin{aligned}g(x, y) &= \mathcal{I}^{-1} \left\{ F(u, v) - H(u, v)F(u, v) \right\} \\&= \mathcal{I}^{-1} \left\{ [1 - H(u, v)]F(u, v) \right\} \\&= \mathcal{I}^{-1} \left\{ [1 + 4\pi^2 D^2(u, v)]F(u, v) \right\}\end{aligned}$$

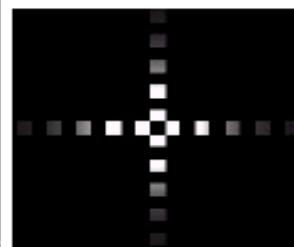
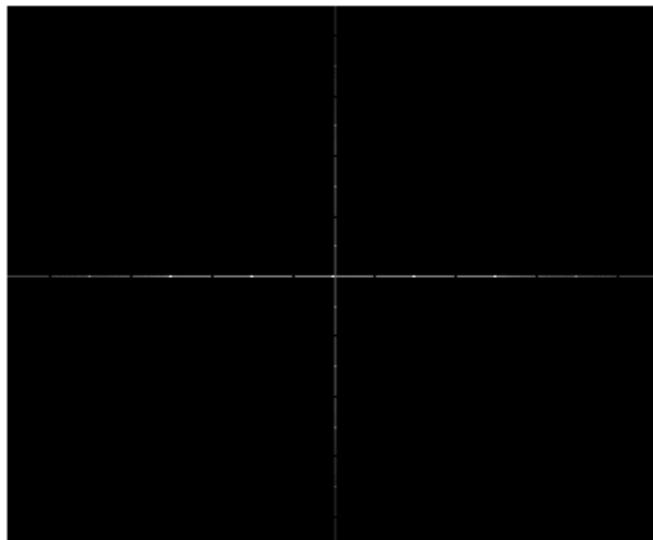
Laplacian in the Frequency Domain

Laplacian in the frequency domain
(not centered)



2-D image of Laplacian in the frequency domain
(not centered)

Inverse DFT of Laplacian in the frequency domain



0	1	0
1	-4	1
0	1	0



Zoomed section of the image on the left compared to spatial filter



Laplacian in the Frequency Domain



a b

FIGURE 4.56
(a) Original, blurry image.
(b) Image enhanced using the Laplacian in the frequency domain.
Compare with Fig. 3.52(d).
(Original image courtesy of NASA.)

4.9.5 Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Filtering

$$g_{mask}(x, y) = f(x, y) - f_{LP}(x, y)$$

$$f_{LP}(x, y) = \mathcal{F}^{-1} [H_{LP}(u, v) F(u, v)]$$

Unsharp masking and highboost filtering

$$g(x, y) = f(x, y) + k * g_{mask}(x, y)$$

$$\begin{aligned} g(x, y) &= \mathcal{F}^{-1} \left\{ \left[1 + k * [1 - H_{LP}(u, v)] \right] F(u, v) \right\} \\ &= \mathcal{F}^{-1} \left\{ \left[1 + k * H_{HP}(u, v) \right] F(u, v) \right\} \end{aligned}$$

- A General formulation: More control parameters

$$g(x, y) = \mathcal{I}^{-1} \left\{ \underbrace{\left[k_1 + k_2 H_{HP}(u, v) \right]}_{k_1 \geq 0 \text{ and } k_2 \geq 0} F(u, v) \right\}$$

High-Frequency-Emphasis
Filtering



High-Frequency-Emphasis Filtering
Gaussian Filter
 $K_1=0.5, K_2=0.75$

a b
c d

Gaussian Filter
 $D_0=40$

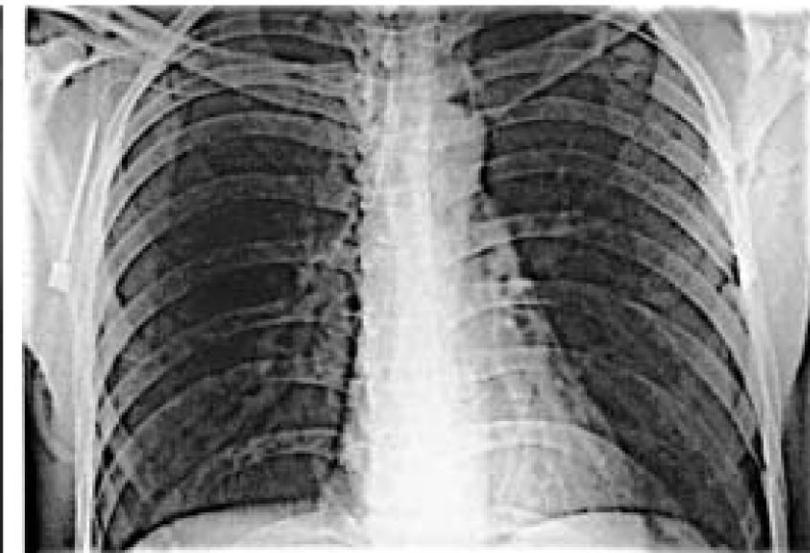
FIGURE 4.57 (a) A chest X-ray image. (b) Result of highpass filtering with a Gaussian filter. (c) Result of high-frequency-emphasis filtering using the same filter. (d) Result of performing histogram equalization on (c). (Original image courtesy of Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

Highpass Filtering Example

Original image



High frequency
emphasis result



Highpass filtering result

After histogram
equalization

4.9.6 Homomorphic Filtering

Illumination

$$f(x, y) = i(x, y)r(x, y) : \text{Multiplicative}$$

Reflectance

$$\Im[f(x, y)] \neq \Im[i(x, y)]\Im[r(x, y)] : \text{Does not work}$$

Define: $z(x, y) = \ln f(x, y) = \ln i(x, y) + \ln r(x, y)$

Then take Fourier Transform:

$$\Im\{z(x, y)\} = \Im\{\ln f(x, y)\} = \Im\{\ln i(x, y)\} + \Im\{\ln r(x, y)\}$$

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

Homomorphic Filtering

Perform Filtering in Fourier Domain:

$$\begin{aligned} S(u, v) &= H(u, v)Z(u, v) \\ &= H(u, v)F_i(u, v) + H(u, v)F_r(u, v) \end{aligned}$$

Take Inverse Fourier Transform:

$$\begin{aligned} s(x, y) &= \mathcal{I}^{-1}\{S(u, v)\} \\ &= \mathcal{I}^{-1}\{H(u, v)F_i(u, v) + H(u, v)F_r(u, v)\} \\ &= \mathcal{I}^{-1}\{H(u, v)F_i(u, v)\} + \mathcal{I}^{-1}\{H(u, v)F_r(u, v)\} \\ &= i'(x, y) + r'(x, y) \end{aligned}$$

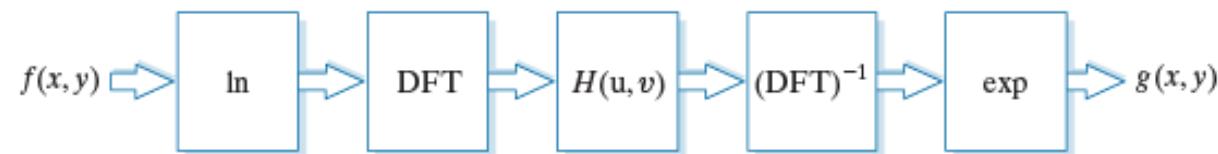
Back to Image Domain using Anti-Logarithm:

$$g(x, y) = e^{s(x, y)} = e^{i'(x, y)}e^{r'(x, y)} = i_0(x, y)r_0(x, y)$$

Homomorphic Filtering

FIGURE 4.58

Summary of steps in homomorphic filtering.



Illumination component, $i_0(x, y)$

- Characterized by slow spatial variations

Reflectance component, $r_0(x, y)$

- Tends to vary abruptly

Associate Fourier transform of logarithm of Image

- Low frequencies → Illumination: $i_0(x, y)$
- High frequencies → Reflectance: $r_0(x, y)$

Homomorphic Filtering

- To Gain Control over Illumination and Reflectance

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

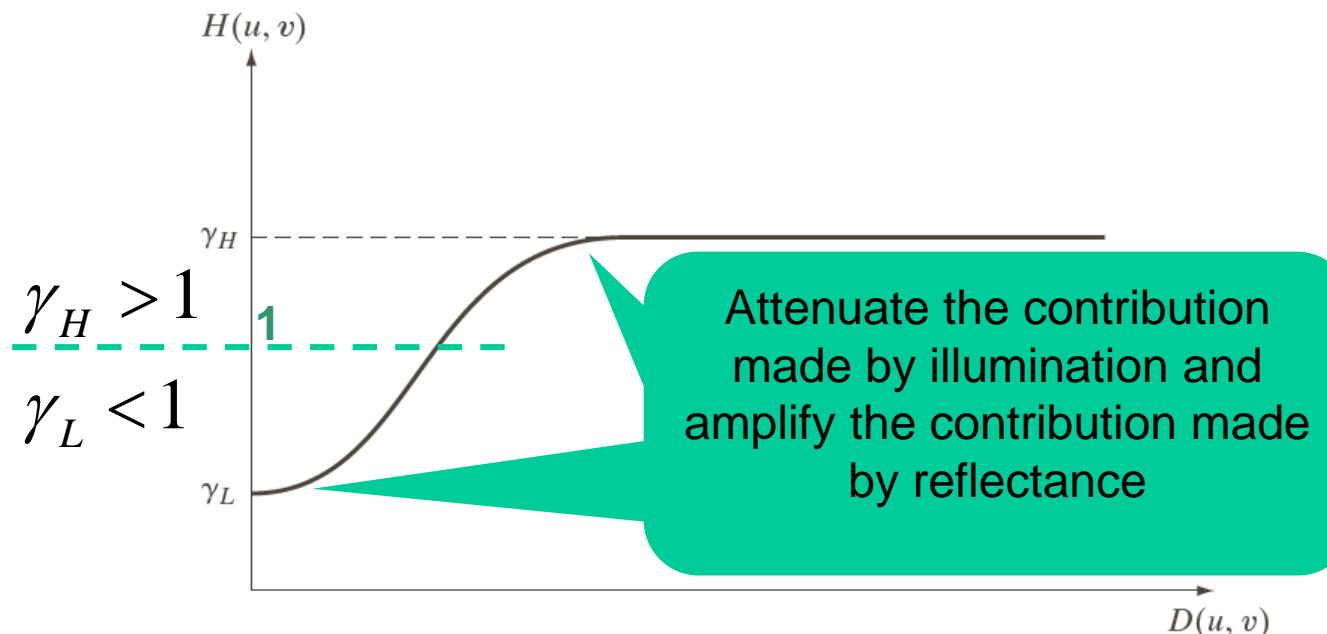


FIGURE 4.59
Radial cross section of a homomorphic filter transfer function.

Result: Simultaneous dynamic Range Compression and Contrast Enhancement



$$\gamma_L = 0.25$$

$$\gamma_H = 2$$

$$c = 1$$

$$D_0 = 80$$

a b

FIGURE 4.60

(a) Full body PET scan. (b) Image enhanced using homomorphic filtering. (Original image courtesy of Dr. Michael E. Casey, CTI Pet Systems.)



Homomorphic Filtering Example



a b

FIGURE 4.60

(a) Full body PET scan. (b) Image enhanced using homomorphic filtering. (Original image courtesy of Dr. Michael E. Casey, CTI Pet Systems.)

$$\gamma_L = 0.25$$

$$\gamma_H = 2$$

$$c = 1$$

$$D_0 = 80$$

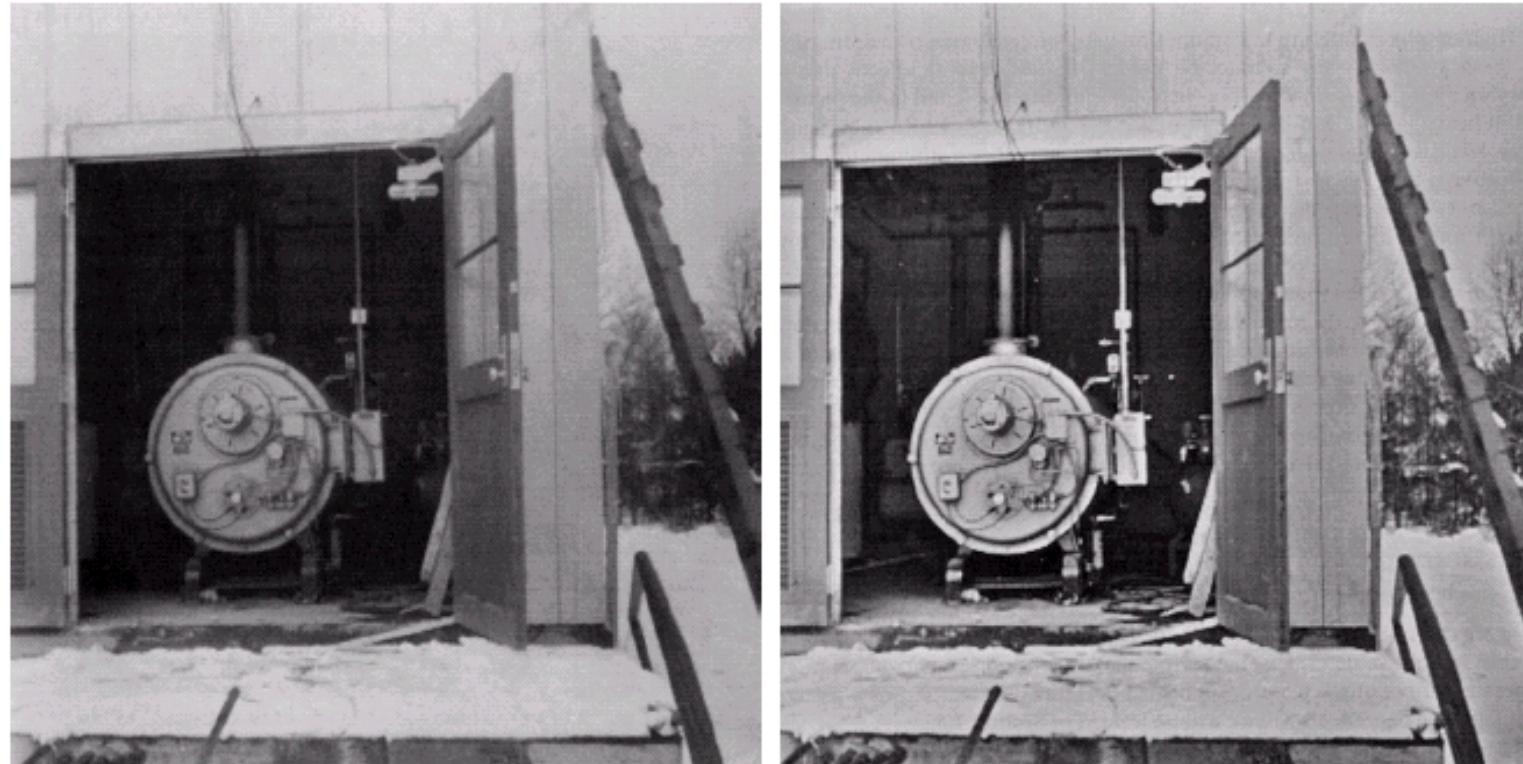


Homomorphic Filtering

a b

FIGURE

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



2nd Edition

Non-Selective Filters (studied so far):

- Operate over the entire frequency rectangle

Selective Filters

- Operate over some part/s, as needed, not the entire frequency rectangle

Bandreject or bandpass

- Process specific bands

Notch filters

- Process small regions of the frequency rectangle



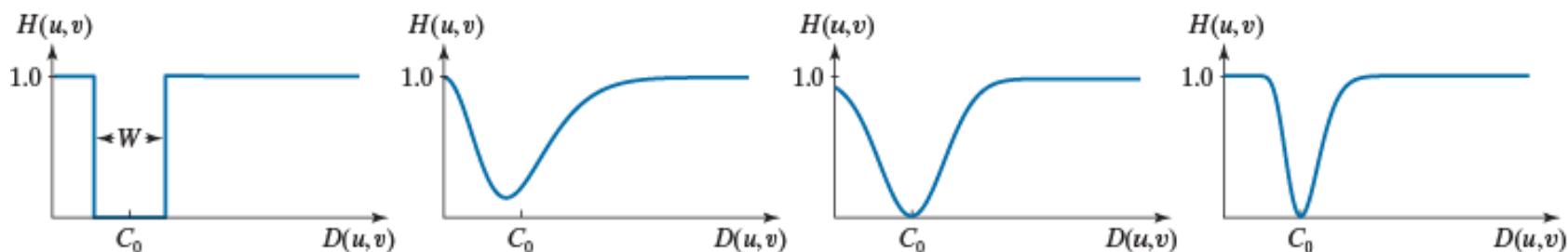
Selective Filtering: Bandreject and Bandpass Filters

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

TABLE 4.7

Bandreject filter transfer functions. C_0 is the center of the band, W is the width of the band, and $D(u,v)$ is the distance from the center of the transfer function to a point (u,v) in the frequency rectangle.

Ideal (IBRF)	Gaussian (GBRF)	Butterworth (BBRF)
$H(u,v) = \begin{cases} 0 & \text{if } C_0 - \frac{W}{2} \leq D(u,v) \leq C_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = 1 - e^{-\left[\frac{D^2(u,v) - C_0^2}{D(u,v)W}\right]^2}$	$H(u,v) = \frac{1}{1 + \left[\frac{D(u,v)W}{D^2(u,v) - C_0^2}\right]^{2n}}$



a b c d

FIGURE 4.61 Radial cross sections. (a) Ideal bandreject filter transfer function. (b) Bandreject transfer function formed by the sum of Gaussian lowpass and highpass filter functions. (The minimum is not 0 and does not align with C_0 .) (c) Radial plot of Eq. (4-149). (The minimum is 0 and is properly aligned with C_0 , but the value at the origin is not 1.) (d) Radial plot of Eq. (4-150); this Gaussian-shape plot meets all the requirements of a bandreject filter transfer function.

Selective Filtering: Bandreject Filters

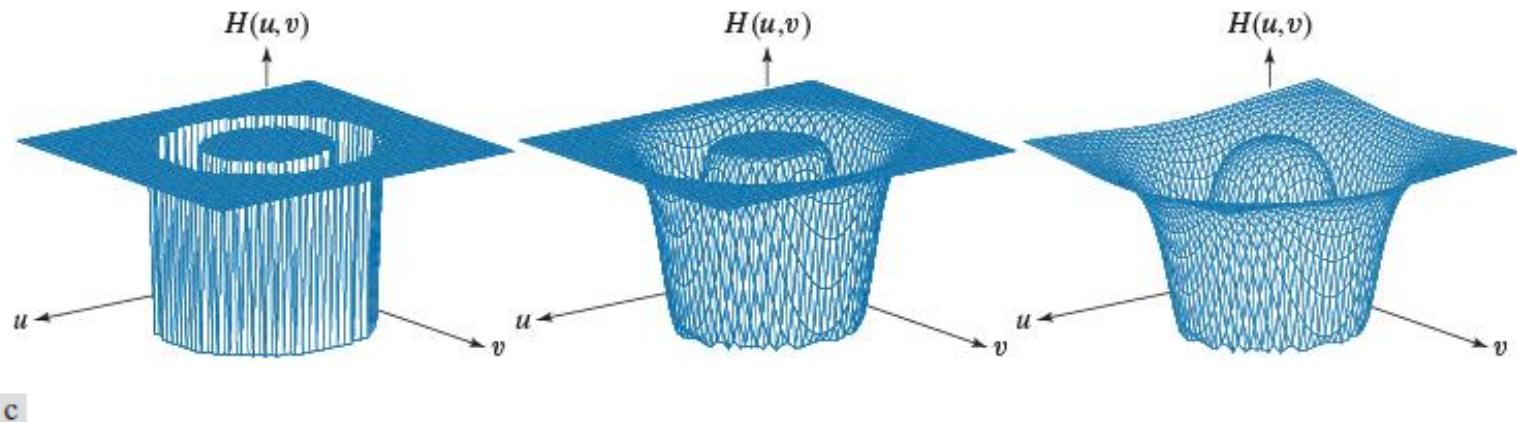
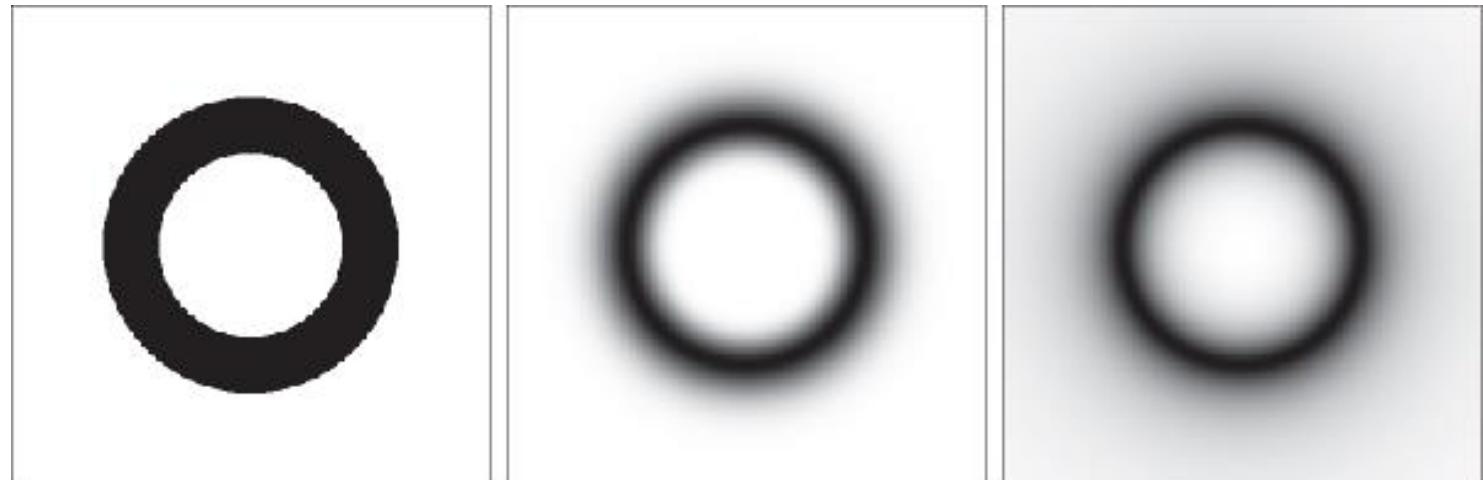


FIGURE 4.62 Perspective plots of (a) ideal, (b) modified Gaussian, and (c) modified Butterworth (of order 1) bandreject filter transfer functions from Table 4.7. All transfer functions are of size 512×512 elements, with $C_0 = 128$ and $W = 60$.

a b c

FIGURE 4.63

(a) The ideal,
(b) Gaussian, and
(c) Butterworth
bandpass transfer
functions from
Fig. 4.62, shown
as images. (The
thin border lines
are not part of the
image data.)



Selective Filtering: Notch Filters

- Zero-phase-shift filters must be symmetric about origin.
- A notch with center at (u_0, v_0) must have a corresponding notch at location $(-u_0, -v_0)$
- Notch reject filters are constructed as product of highpass filters whose centers have been translated to the centers of the notches

$$H_{NR}(u, v) = \prod_{k=1}^Q H_k(u, v) H_{-k}(u, v)$$

where $H_k(u, v)$ and $H_{-k}(u, v)$ are highpass filters whose centers are at (u_k, v_k) and $(-u_k, -v_k)$, respectively.

Selective Filtering: Notch Filters

A Butterworth notch reject filter of order n

$$H_{NR}(u, v) = \prod_{k=1}^3 \left[\frac{1}{1 + [D_{0k} / D_k(u, v)]^{2n}} \right] \left[\frac{1}{1 + [D_{0k} / D_{-k}(u, v)]^{2n}} \right]$$

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

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



Examples: Notch Filters (1)



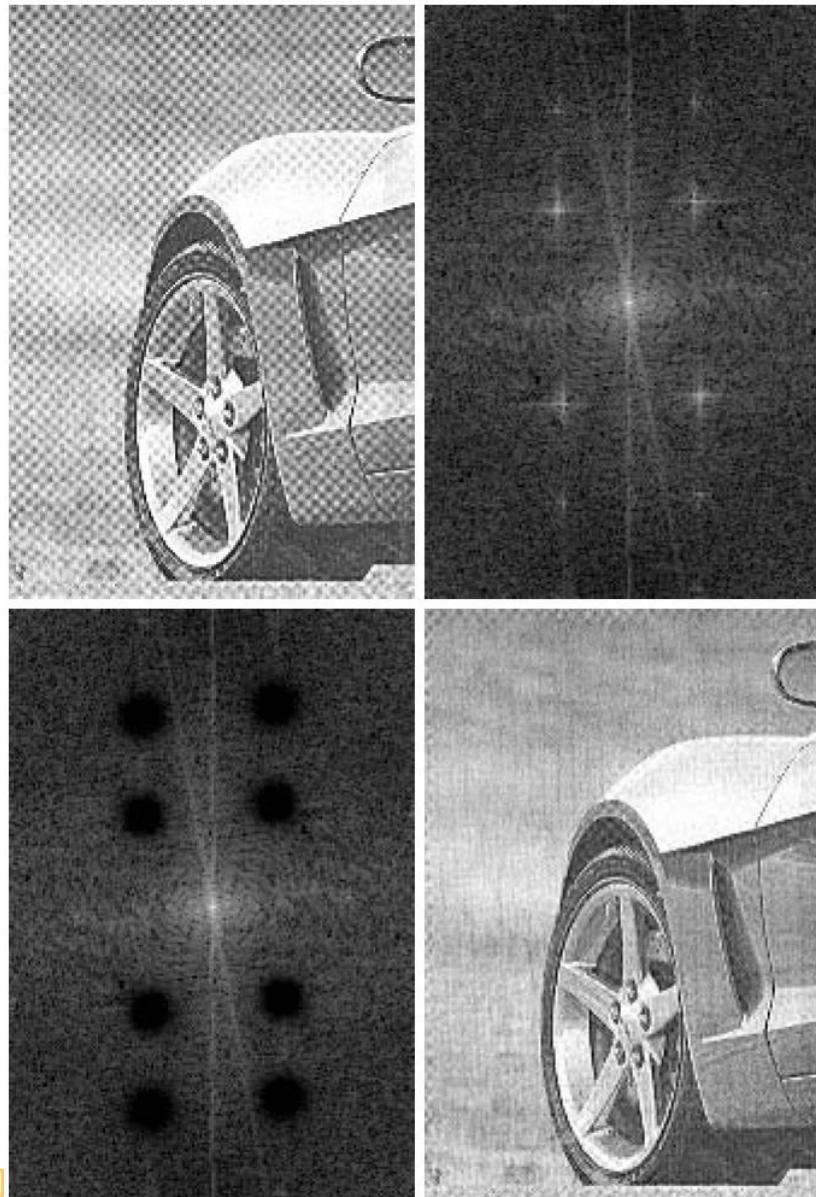
a	b
c	d

FIGURE 4.64

- (a) Sampled newspaper image showing a moiré pattern.
- (b) Spectrum.
- (c) Butterworth notch reject filter multiplied by the Fourier transform.
- (d) Filtered image.

A Butterworth notch reject filter $D_0=3$ and $n=4$ for all notch pairs

Notch Reject Filters (cont...)



a b
c d

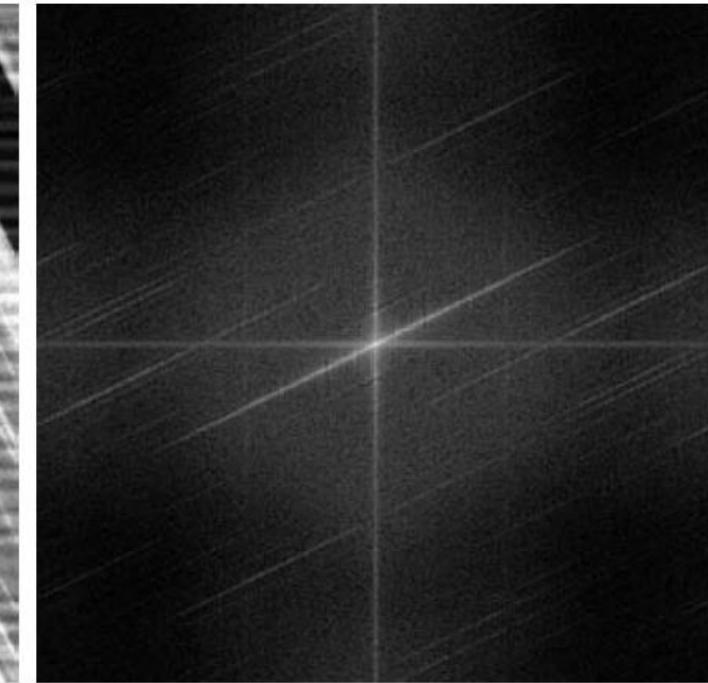
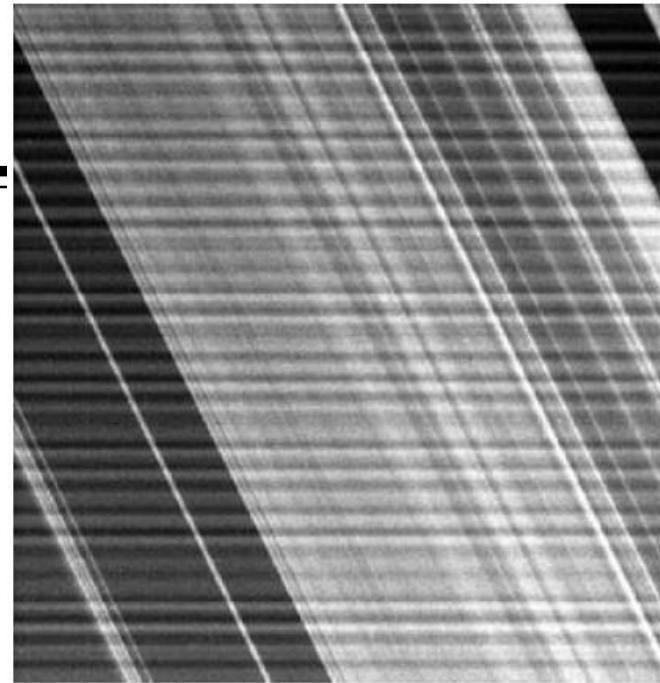
FIGURE 4.64
(a) Sampled newspaper image showing a moiré pattern.
(b) Spectrum.
(c) Fourier transform multiplied by a Butterworth notch reject filter transfer function.
(d) Filtered image.

Examples: Notch Filters WRIGHT STATE UNIVERSITY (2)

a b
c d

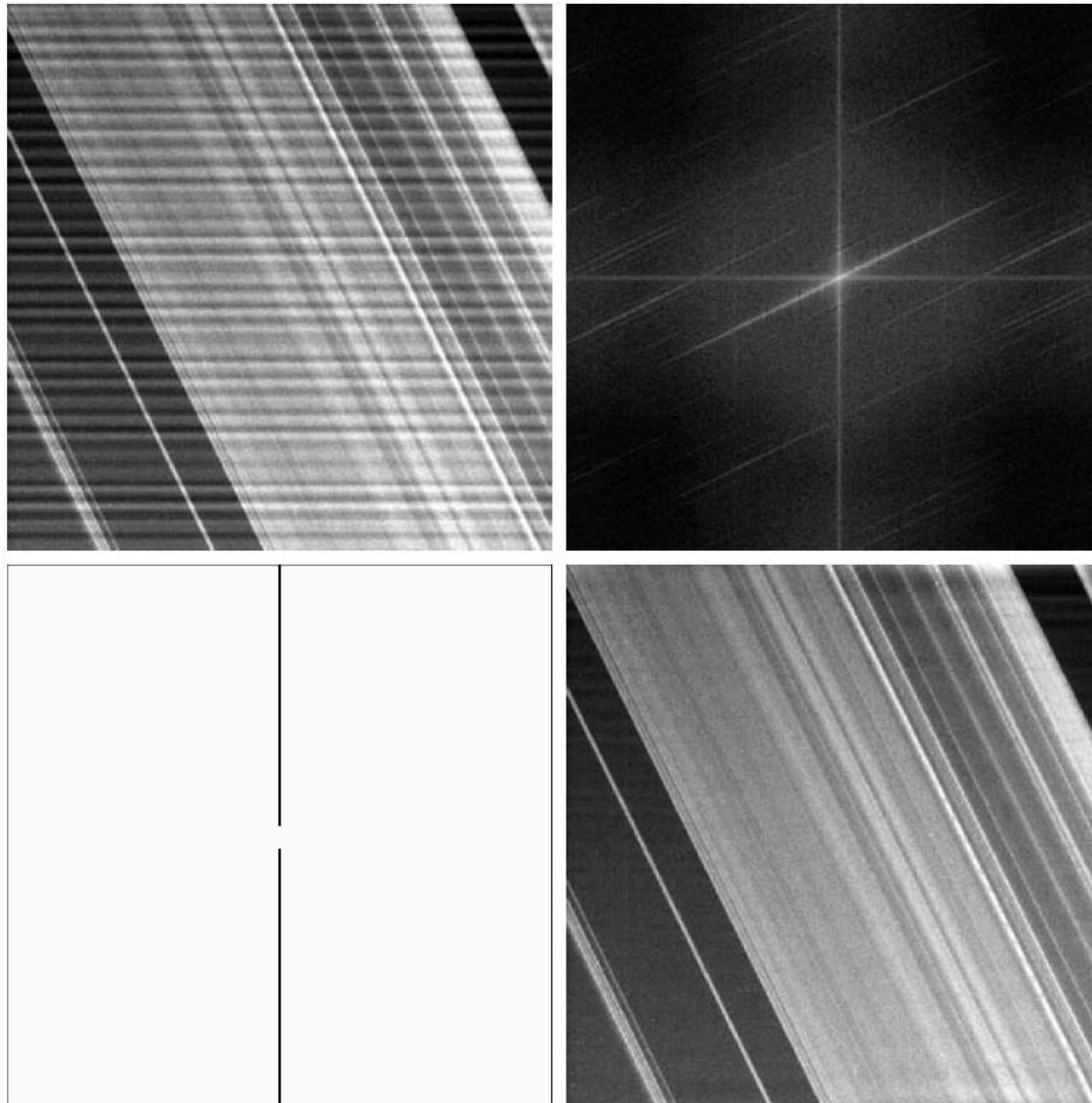
FIGURE 4.65

- (a) Image of Saturn rings showing nearly periodic interference.
(b) Spectrum. (The bursts of energy in the vertical axis near the origin correspond to the interference pattern).
(c) A vertical notch reject filter transfer function.
(d) Result of filtering.
(The thin black border in (c) is not part of the data.) (Original image courtesy of Dr. Robert A. West, NASA/JPL.)





Notch Filters (cont...)



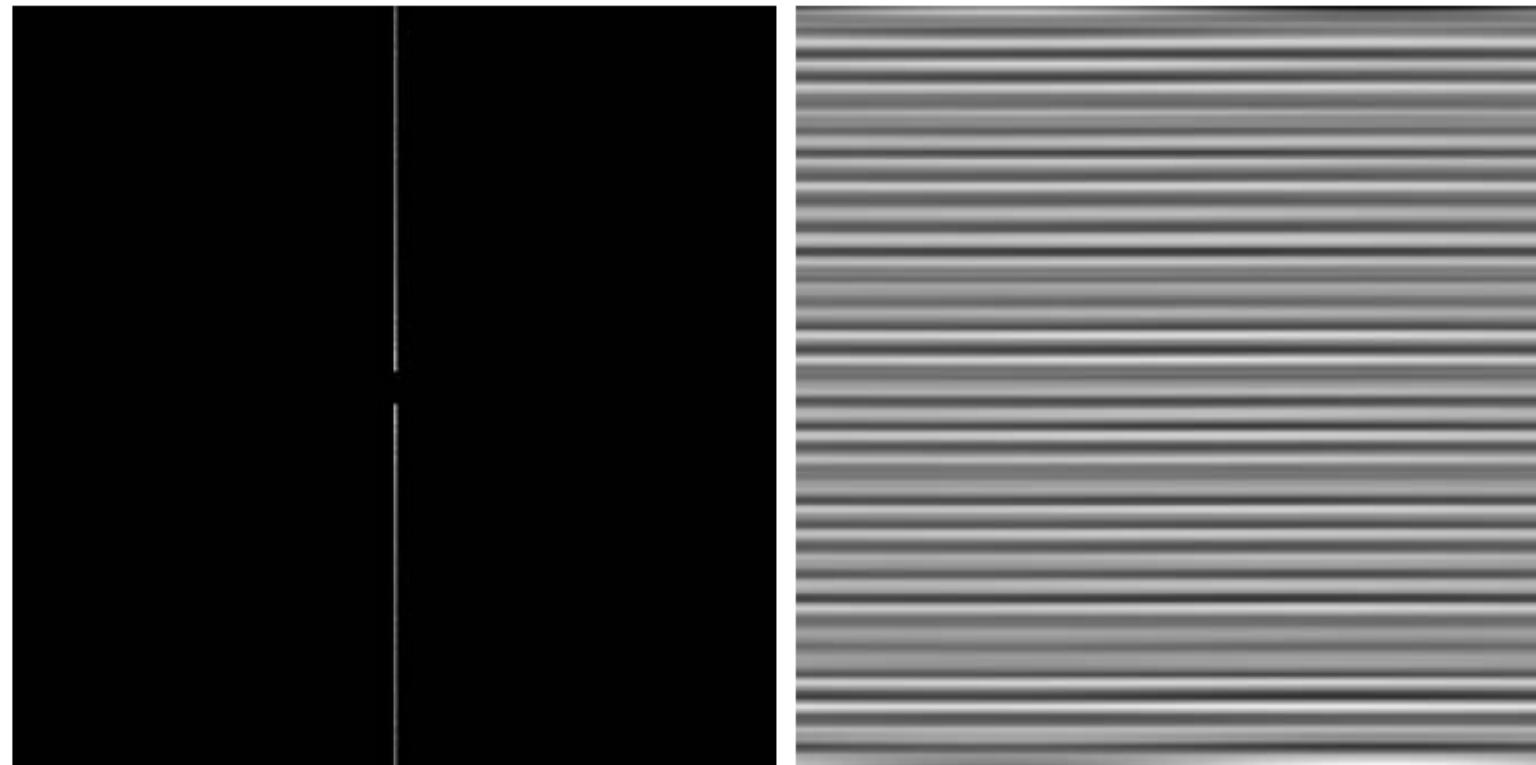
a
b
c
d

FIGURE 4.65

- (a) Image of Saturn rings showing nearly periodic interference.
(b) Spectrum. (The bursts of energy in the vertical axis near the origin correspond to the interference pattern).
(c) A vertical notch reject filter transfer function.
(d) Result of filtering.
(The thin black border in (c) is not part of the data.) (Original image courtesy of Dr. Robert A. West, NASA/JPL.)



Notch Filters (cont...)



a b

FIGURE 4.66

- (a) Notch pass filter function used to isolate the vertical axis of the DFT of Fig. 4.65(a).
(b) Spatial pattern obtained by computing the IDFT of (a).

Shows the Interference only

Fast Fourier Transform

- The reason that Fourier based techniques have become so popular is the development of the *Fast Fourier Transform (FFT)* algorithm
- FFT allows the Fourier transform to be carried out in a reasonable amount of time
- Reduces the complexity from $O(N^4)$ to $O(N^2\log N^2)$.
- The amount of time required to perform a Fourier transform is reduced by a factor of 100 - 600 times!

- Similar tasks can be done in the spatial and frequency domains
- Filtering in the spatial domain can be easier to understand
- Filtering in the frequency domain can be much faster - especially for large images

Summary

- In Chapter-4 Image Enhancement in the frequency domain was studied
 - The Fourier series & the Fourier transform
 - Image Processing in the frequency domain
 - Image smoothing
 - Image sharpening
 - Fast Fourier Transform
- Chapter-5 will examine image restoration using the spatial and frequency based techniques we have been looking at

Acknowledgements

The slides are primarily based on the figures and images in the Digital Image Processing textbook by Gonzalez and Woods:

- http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

In addition, slides have been adopted and modified from the following excellent sources:

- http://www.cs.uoi.gr/~cnikou/Courses/Digital_Image_Processing
- <http://www.comp.dit.ie/bmacnamee/gaip.htm>
- <http://baggins.nottingham.edu.my/~hs00ihock/G52IIP/>
- <http://gear.kku.ac.th/~nawapak/178353.html>
- <https://cs.nmt.edu/~ip/index.html>

- Property
 - The DFT of a N -length $f[n]$ signal is periodic with period N .

$$F[k + N] = F[k]$$

- This is due to the periodicity of the complex exponential:

$$\left(w_N\right)^{n+kN} = \left(e^{-j\frac{2\pi}{N}}\right)^{n+kN} = w_N^n \left(w_N\right)^{kN} = w_N^n = e^{-j\frac{2\pi n}{N}}$$

- Property: Sum of complex exponentials

$$\frac{1}{N} \sum_{n=0}^N w_N^{kn} = \begin{cases} 1, & k = rN, r \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}$$

$$w_N = e^{-j \frac{2\pi}{N}}$$

- DFT pair of signal $f[n]$ of length N may be expressed in matrix-vector form.

$$F[k] = \sum_{n=0}^{N-1} f[n] w_N^{nk}, \quad 0 \leq k \leq N-1$$

$$f[n] = \frac{1}{N} \sum_{n=0}^{N-1} F[k] w_N^{-nk}, \quad 0 \leq n \leq N-1$$

$$w_N = e^{-j \frac{2\pi}{N}}$$

$$\mathbf{F} = \mathbf{A}\mathbf{f}$$

$$\mathbf{A} = \begin{bmatrix} \left(w_N^0\right)^0 & \left(w_N^0\right)^1 & \left(w_N^0\right)^2 & \dots & \left(w_N^0\right)^{N-1} \\ \left(w_N^1\right)^0 & \left(w_N^1\right)^1 & \left(w_N^1\right)^2 & \dots & \left(w_N^1\right)^{N-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \left(w_N^{N-1}\right)^0 & \left(w_N^{N-1}\right)^1 & \left(w_N^{N-1}\right)^2 & \dots & \left(w_N^{N-1}\right)^{N-1} \end{bmatrix}$$

$$\mathbf{f} = [f[0], f[1], \dots, f[N-1]]^T, \quad \mathbf{F} = [F[0], F[1], \dots, F[N-1]]^T$$

Example for $N=4$

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

The inverse DFT is then expressed by:

$$\mathbf{f} = \mathbf{A}^{-1}\mathbf{F}$$

$$\mathbf{A}^{-1} = \frac{1}{N}(\mathbf{A}^*)^T = \frac{1}{N} \left(\begin{bmatrix} (w_N^0)^0 & (w_N^0)^1 & (w_N^0)^2 & \dots & (w_N^0)^{N-1} \\ (w_N^1)^0 & (w_N^1)^1 & (w_N^1)^2 & \dots & (w_N^1)^{N-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (w_N^{N-1})^0 & (w_N^{N-1})^1 & (w_N^{N-1})^2 & \dots & (w_N^{N-1})^{N-1} \end{bmatrix}^* \right)^T$$

This is derived by the complex exponential sum property.



Linear Convolution

$$f[n] = \{1, 2, 2\}, \quad h[n] = \{1, -1\}, \quad N_1 = 3, N_2 = 2$$

$$g[n] = f[n] * h[n] = \sum_{m=-\infty}^{+\infty} f[m]h[n-m]$$

Length of Convolution: $N=N_1+N_2-1=4$

Linear Convolution (cont.)

$$f[n] = \{1, 2, 2\}, \quad h[n] = \{1, -1\}, \quad N_1 = 3, N_2 = 2$$

$$g[n] = f[n] * h[n] = \sum_{m=-\infty}^{+\infty} f[m]h[n-m]$$

	$f[m]$	1	2	2		$g[n]$		
$n = 0$	$h[0-m]$	-1	1		\rightarrow	1		
$n = 1$	$h[1-m]$		-1	1	\rightarrow	1		
$n = 2$	$h[2-m]$			-1	1	\rightarrow	0	
$n = 3$	$h[3-m]$				-1	1	\rightarrow	-2

$$g[n] = \{1, 1, 0, -2\}$$

DFT Domain Shifts are Circular Shifts

- Signal $x[n]$ of length N .
- A circular shift ensures that the resulting signal will keep its length N .
- It is a shift modulo- N denoted by

$$x[(n-m)_N] = x[(n-m) \bmod N]$$

- Example: $x[n]$ is of length $N=8$.

$$x[(-2)_N] = x[(-2)_8] = x[6]$$

$$x[(10)_N] = x[(10)_8] = x[2]$$

Circular Convolution

$$f[n] = \{1, 2, 2\}, \quad h[n] = \{1, -1\}, \quad N_1 = 3, N_2 = 2$$

$$g[n] = f[n] * h[n] = \sum_{m=-\infty}^{+\infty} f[m]h[(n-m)_N]$$

Circular shift modulo- N

The result is of length: $N = \max \{N_1, N_2\} = 3$

Circular Convolution (cont.)

$$f[n] = \{ \underline{1}, 2, 2 \}, \quad h[n] = \{ \underline{1}, -1 \}, \quad N_1 = 3, N_2 = 2$$

$$g[n] = f[n] * h[n] = \sum_{m=-\infty}^{+\infty} f[m]h[(n-m)_N]$$

	$f[m]$	1	2	2	$g[n]$
$n = 0$	$h[(0-m)_N]$	-1	1	-1	-1
$n = 1$	$h[(1-m)_N]$		-1	1	1
$n = 2$	$h[(2-m)_N]$			-1	0

$g[n] = \{-\underline{1}, 1, 0, \}$ **Result has wrong length and values**

$$g[n] = f[n] * h[n] \leftrightarrow G[k] = F[k]H[k]$$

- The property holds for the circular convolution.
- In signal processing we are interested in linear convolution.
- Is there a similar property for the linear convolution?

DFT and Convolution (cont.)

$$g[n] = f[n] * h[n] \leftrightarrow G[k] = F[k]H[k]$$

- Let $f[n]$ be of length N_1 and $h[n]$ be of length N_2 .
- Then $g[n] = f[n]*h[n]$ is of length N_1+N_2-1 .
- If the **signals are zero-padded** to length $N=N_1+N_2-1$ then their circular convolution will be the same as their linear convolution:

$$\tilde{g}[n] = \tilde{f}[n]\tilde{h}[n] \leftrightarrow \tilde{G}[k] = \tilde{F}[k]\tilde{H}[k]$$

Zero-padded signals

DFT and Convolution (cont.)

$$f[n] = \{1, 2, 2\}, \quad h[n] = \{1, -1\}, \quad N_1 = 3, N_2 = 2$$

- Zero-padding to length: $N=N_1+N_2-1=4$

$$\tilde{f}[n] = \{1, 2, 2, 0\}, \quad \tilde{h}[n] = \{1, -1, 0, 0\}$$

$f[m]$					$g[n]$
$h[(n-0)_4]$	0	0	-1	1	1
$h[(n-1)_4]$		0	0	-1 1 0 0	1
$h[(n-2)_4]$			0	0 -1 1 0	0
$h[(n-3)_4]$				0 0 -1 1	-2

$$\tilde{g}[n] = \{1, 1, 0, -2\} = g[n]$$

- The result is the same as that for linear convolution

- Verification using DFT

$$\tilde{\mathbf{F}} = \mathbf{A}\tilde{\mathbf{f}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ -1-j2 \\ 1 \\ -1+j2 \end{bmatrix}$$

$$\tilde{\mathbf{H}} = \mathbf{A}\tilde{\mathbf{h}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1+j \\ 2 \\ 1-j \end{bmatrix}$$

$$\tilde{G}[k] = \tilde{F}[k] \tilde{H}[k]$$

- Element-wise multiplication

$$\tilde{\mathbf{G}} = \tilde{\mathbf{F}} \times \tilde{\mathbf{H}} = \begin{bmatrix} 5 \times 0 \\ (-1 - j2) \times (1 + j) \\ 1 \times 2 \\ (-1 + j2) \times (1 - j) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 - j3 \\ 2 \\ 1 + j3 \end{bmatrix}$$

- Inverse DFT of the result

$$\tilde{\mathbf{g}} = \mathbf{A}^{-1}\tilde{\mathbf{G}} = \frac{1}{4}(\mathbf{A}^*)^T \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 0 \\ 1-j3 \\ 2 \\ 1+j3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ -2 \end{bmatrix}$$

- The same result as their linear convolution.