

# EE 5098 – Digital Image Processing

## 4. Filtering in the Frequency Domain

## Outline

- Preliminary Concepts
- Sampling and the Fourier Transform of Sampled Functions
- Discrete Fourier Transform
- The Basics of Filtering in the Frequency Domain
- Image Smoothing Using Lowpass Frequency-Domain Filters
- Image Sharpening Using Highpass Filters
- Selective Filtering

## Jean Baptiste Joseph Fourier

- French mathematician and physicist (03/21/1768-05/16/1830)

[http://en.wikipedia.org/wiki/Joseph\\_Fourier](http://en.wikipedia.org/wiki/Joseph_Fourier)

Orphaned: at nine

French campaign in  
Egypt with  
**Napoleon I:** 1798  
Governor of Lower  
Egypt



Permanent  
Secretary of the  
French Academy of  
Sciences: 1822

*Théorie analytique  
de la chaleur :*  
1822

**(The Analytic  
Theory of Heat)**

## Fourier Series and Fourier Transform

### Fourier Series

Any periodic function can be expressed as the sum of sines and /or cosines of different frequencies, each multiplied by a different coefficient

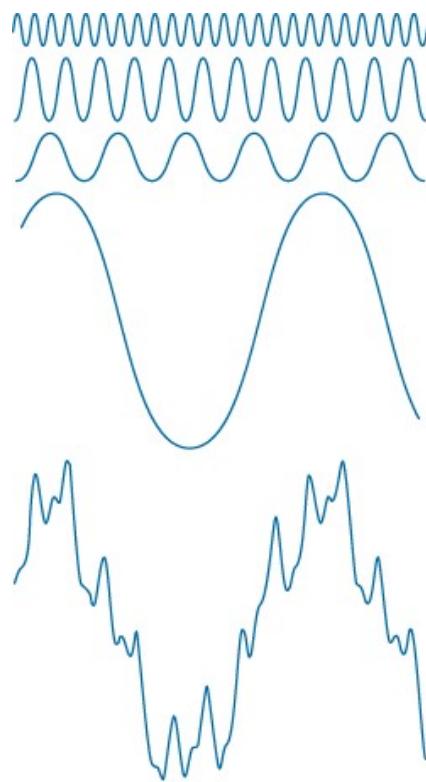
### Fourier Transform

Any function that is not periodic can be expressed as the integral of sines and /or cosines multiplied by a weighing function

## Fourier Series: Example

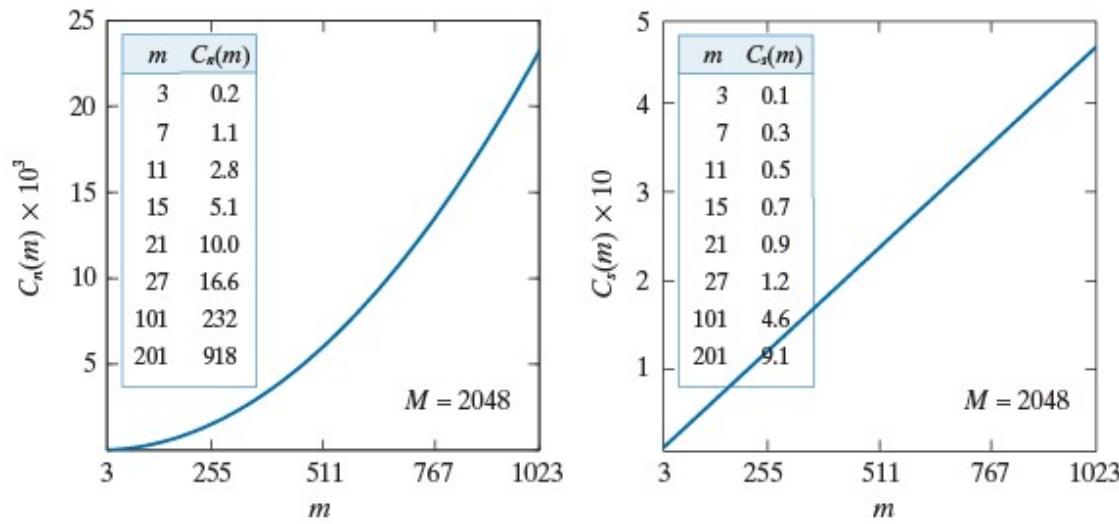
**FIGURE 4.1**

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



a b

**FIGURE 4.2**  
(a) Computational advantage of the FFT over non-separable spatial kernels.  
(b) Advantage over separable kernels.  
The numbers for  $C(m)$  in the inset tables are not to be multiplied by the factors of 10 shown for the curves.



## Preliminary Concepts

$j = \sqrt{-1}$ , a complex number

$$C = R + jI$$

the conjugate

$$C^* = R - jI$$

$$|C| = \sqrt{R^2 + I^2} \text{ and } \theta = \arctan(I / R)$$

$$C = |C|(\cos \theta + j \sin \theta)$$

Using Euler's formula,

$$C = |C| e^{j\theta}$$

## Fourier Series

A function  $f(t)$  of a continuous variable  $t$  that is periodic with period,  $T$ , can be expressed as the sum of sines and cosines multiplied by appropriate coefficients

$$1Dim: \quad f(t) = \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{T} t}$$

$\hookrightarrow$ 某種 weight

where

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-j \frac{2\pi n}{T} t} dt \quad \text{for } n = 0, \pm 1, \pm 2, \dots$$

## Impulses and the Sifting Property (1)

A *unit impulse* of a continuous variable  $t$  located at  $t=0$ , denoted  $\delta(t)$ , is defined by

$$\delta(t) = \begin{cases} \infty & \text{if } t = 0 \\ 0 & \text{if } t \neq 0 \end{cases}$$

and constrained to satisfy the identity

$$\int_{-\infty}^{\infty} \delta(t) dt = 1.$$

*Sifting property*

$$\int_{-\infty}^{\infty} f(t) \delta(t) dt = f(0)$$

$$\int_{-\infty}^{\infty} f(t) \delta(t - t_0) dt = f(t_0)$$

## Impulses and the Sifting Property (2)

A *unit impulse* of a discrete variable  $x$  located at  $x=0$ , denoted  $\delta(x)$ , is defined by

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x \neq 0 \end{cases}$$

and constrained also to satisfy the identity

$$\sum_{x=-\infty}^{\infty} \delta(x) = 1$$

*Sifting property*

$$\sum_{x=-\infty}^{\infty} f(x)\delta(x) = f(0)$$

$$\sum_{x=-\infty}^{\infty} f(x)\delta(x - x_0) = f(x_0)$$

## Impulses and the Sifting Property (3)

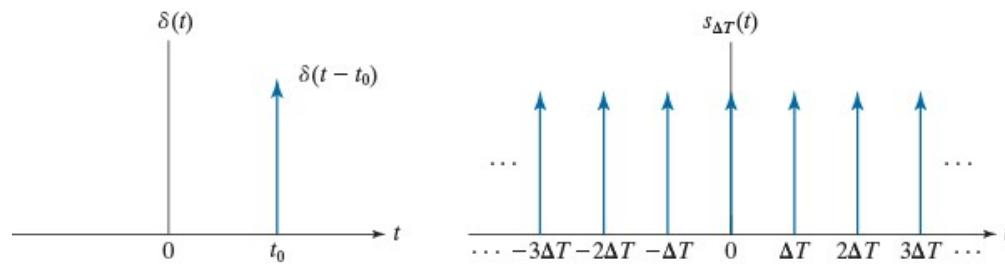
Denote an impulse train by  $s_{\Delta T}(t)$ ,

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

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 Fourier Transform

The *Fourier Transform* of a continuous function  $f(t)$

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

The *Inverse Fourier Transform* of  $F(\mu)$

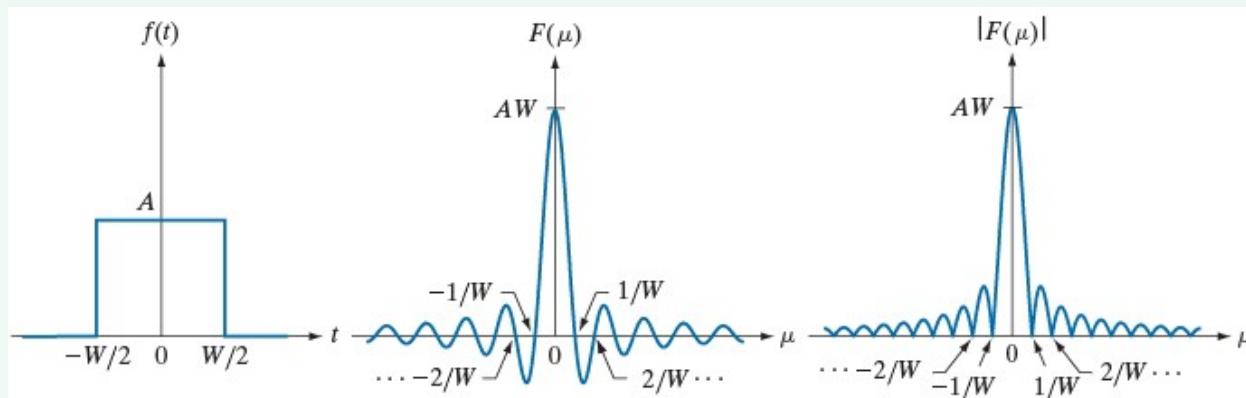
$$f(t) = \Im^{-1}\{F(\mu)\} = \int_{-\infty}^{\infty} F(\mu) e^{j2\pi\mu t} d\mu$$

Fourier transform pair

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

*freq domain*

## Fourier Transform of 1-D Signals



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.

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

## Fourier Transform of an Impulse

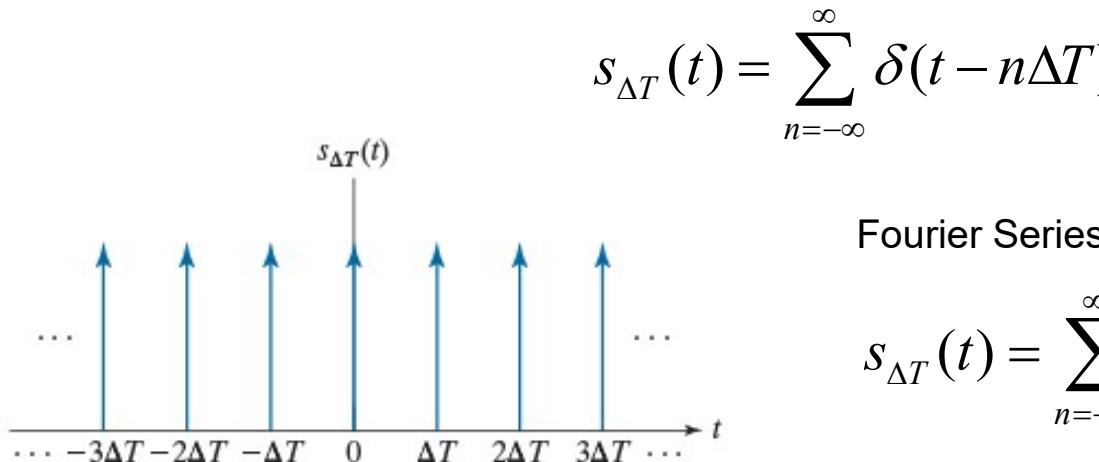
The Fourier transform of a unit impulse located at the origin:

$$\begin{aligned} F(\mu) &= \int_{-\infty}^{\infty} \delta(t) e^{-j2\pi\mu t} dt \\ &= e^{-j2\pi\mu 0} \\ &= 1 \end{aligned}$$

The Fourier transform of a unit impulse located at  $t = t_0$ :

$$\begin{aligned} F(\mu) &= \int_{-\infty}^{\infty} \delta(t - t_0) e^{-j2\pi\mu t} dt \\ &= e^{-j2\pi\mu t_0} \\ &= \cos(2\pi\mu t_0) - j \sin(2\pi\mu t_0) \end{aligned}$$

## Fourier Transform of an Impulse Train (1)



Fourier Series representation:

$$s_{\Delta T}(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\Delta T) = \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{\Delta T} t}$$
$$c_n = \frac{1}{\Delta T} \int_{-\Delta T/2}^{\Delta T/2} s_{\Delta T}(t) e^{-j \frac{2\pi n}{\Delta T} t} dt$$

$$= \frac{1}{\Delta T} \int_{-\Delta T/2}^{\Delta T/2} \delta(t) e^{-j \frac{2\pi n}{\Delta T} t} dt$$
$$= \frac{1}{\Delta T} e^0 = \frac{1}{\Delta T}$$

## Fourier Transform of an Impulse Train (2)

$$\begin{aligned}
 s_{\Delta T}(t) &= \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{\Delta T} t} = \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} e^{j \frac{2\pi n}{\Delta T} t} \\
 \Im \left\{ e^{j \frac{2\pi n}{\Delta T} t} \right\} &= \int_{-\infty}^{\infty} e^{j \frac{2\pi n}{\Delta T} t} e^{-j 2\pi \mu t} dt = \int_{-\infty}^{\infty} e^{j \frac{2\pi n}{\Delta T} t} e^{-j 2\pi \mu t} dt \\
 &= \int_{-\infty}^{\infty} e^{-j 2\pi (\mu - \frac{n}{\Delta T}) t} dt = \delta(\mu - \frac{n}{\Delta T}) \\
 S(\mu) \triangleq \Im \{S_{\Delta T}(t)\} &= \Im \left\{ \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} e^{j \frac{2\pi n}{\Delta T} t} \right\} = \frac{1}{\Delta T} \Im \left\{ \sum_{n=-\infty}^{\infty} e^{j \frac{2\pi n}{\Delta T} t} \right\} \\
 &= \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \delta(\mu - \frac{n}{\Delta T}) \Rightarrow \text{also an impulse train}
 \end{aligned}$$

## Fourier Transform and Convolution

The convolution of two functions is denoted by the operator  $\star$ . Then,

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

$$\begin{aligned}\Im\{f(t) \star h(t)\} &= \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} f(\tau)h(t - \tau)d\tau \right] e^{-j2\pi\mu t} dt \\ &= \int_{-\infty}^{\infty} f(\tau) \left[ \int_{-\infty}^{\infty} h(t - \tau)e^{-j2\pi\mu t} dt \right] d\tau \\ &= \int_{-\infty}^{\infty} f(\tau) \left[ H(\mu)e^{-j2\pi\mu\tau} \right] d\tau \\ &= H(\mu) \int_{-\infty}^{\infty} f(\tau) e^{-j2\pi\mu\tau} d\tau \\ &= H(\mu)F(\mu)\end{aligned}$$

## Fourier Transform and Convolution

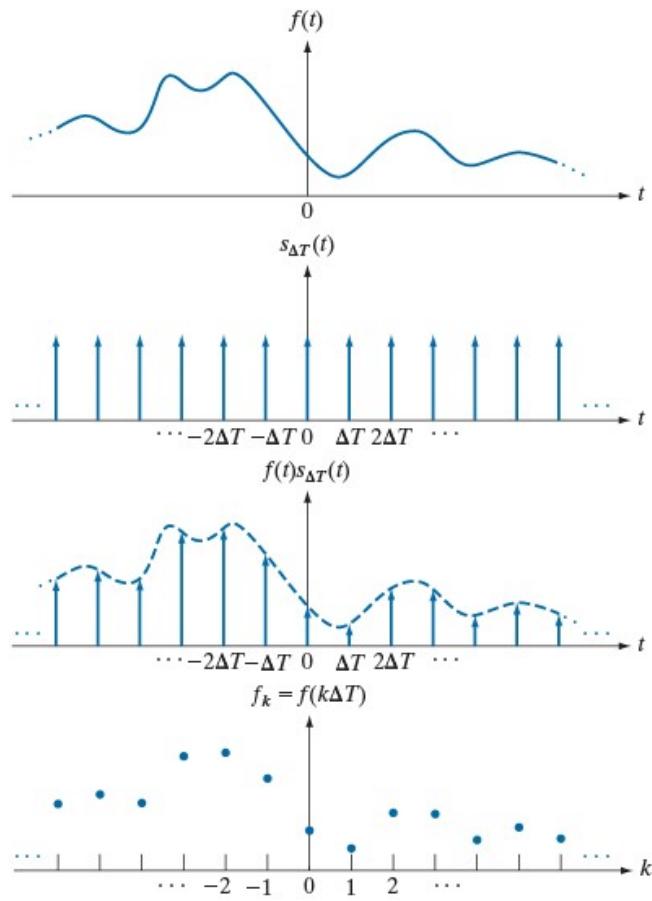
Fourier Transform Pairs

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

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

# Fourier Transform of Sampled Functions (1)

$$\begin{aligned}\tilde{f}(t) &= f(t)s_{\Delta T}(t) \\ &= \sum_{n=-\infty}^{\infty} f(t)\delta(t-n\Delta T)\end{aligned}$$



**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.)

## Fourier Transform of Sampled Functions (2)

$$\tilde{F}(\mu) = \Im\left\{\tilde{f}(t)\right\} = \Im\left\{f(t)s_{\Delta T}(t)\right\} = F(\mu) \star S(\mu)$$

$$S(\mu) = \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \delta(\mu - \frac{n}{\Delta T})$$

$$\tilde{F}(\mu) = F(\mu) \star S(\mu) = \int_{-\infty}^{\infty} F(\tau) S(\mu - \tau) d\tau$$

$$= \frac{1}{\Delta T} \int_{-\infty}^{\infty} F(\tau) \sum_{n=-\infty}^{\infty} \delta(\mu - \tau - \frac{n}{\Delta T}) d\tau$$

$$= \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} F(\tau) \delta(\mu - \tau - \frac{n}{\Delta T}) d\tau$$

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

## Fourier Transform of Sampled Functions (3)

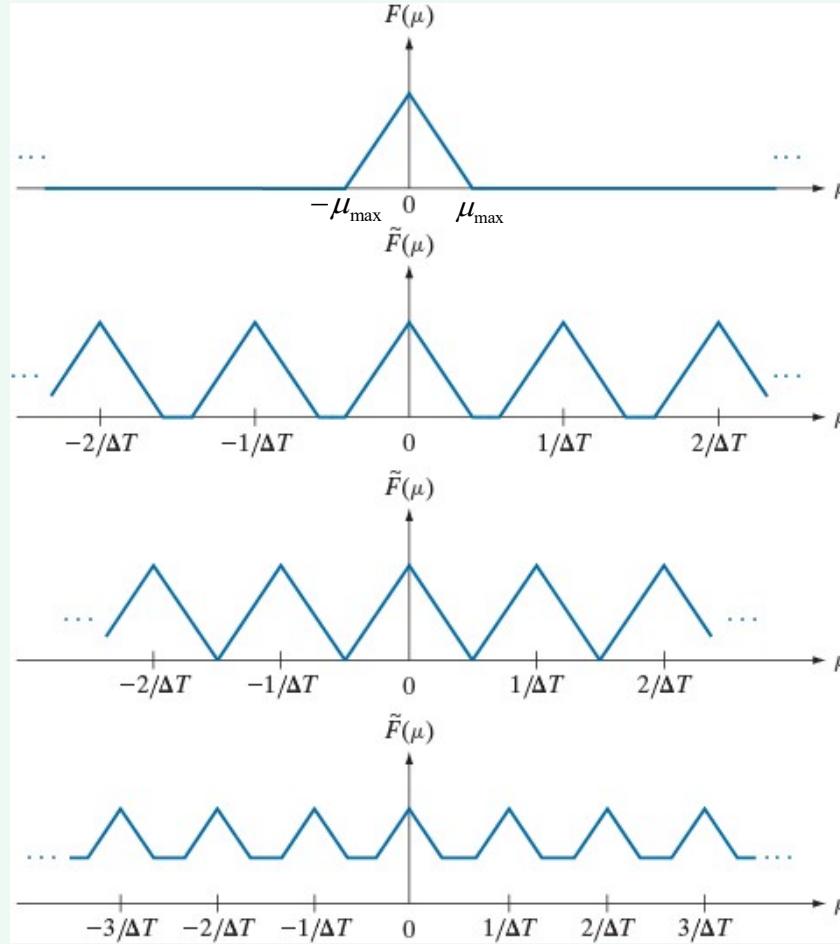
- ▶ A **bandlimited** signal is a signal whose Fourier transform is zero above a certain finite frequency. In other words, if the Fourier transform has finite support then the signal is said to be bandlimited.

An example of a simple bandlimited signal is a sinusoid of the form,

$$x(t) = \sin(2\pi ft + \theta)$$

## Fourier Transform of Sampled Functions (4)

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



Over-sampling

$$\frac{1}{\Delta T} > 2\mu_{\max}$$

Critically-sampling

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

under-sampling

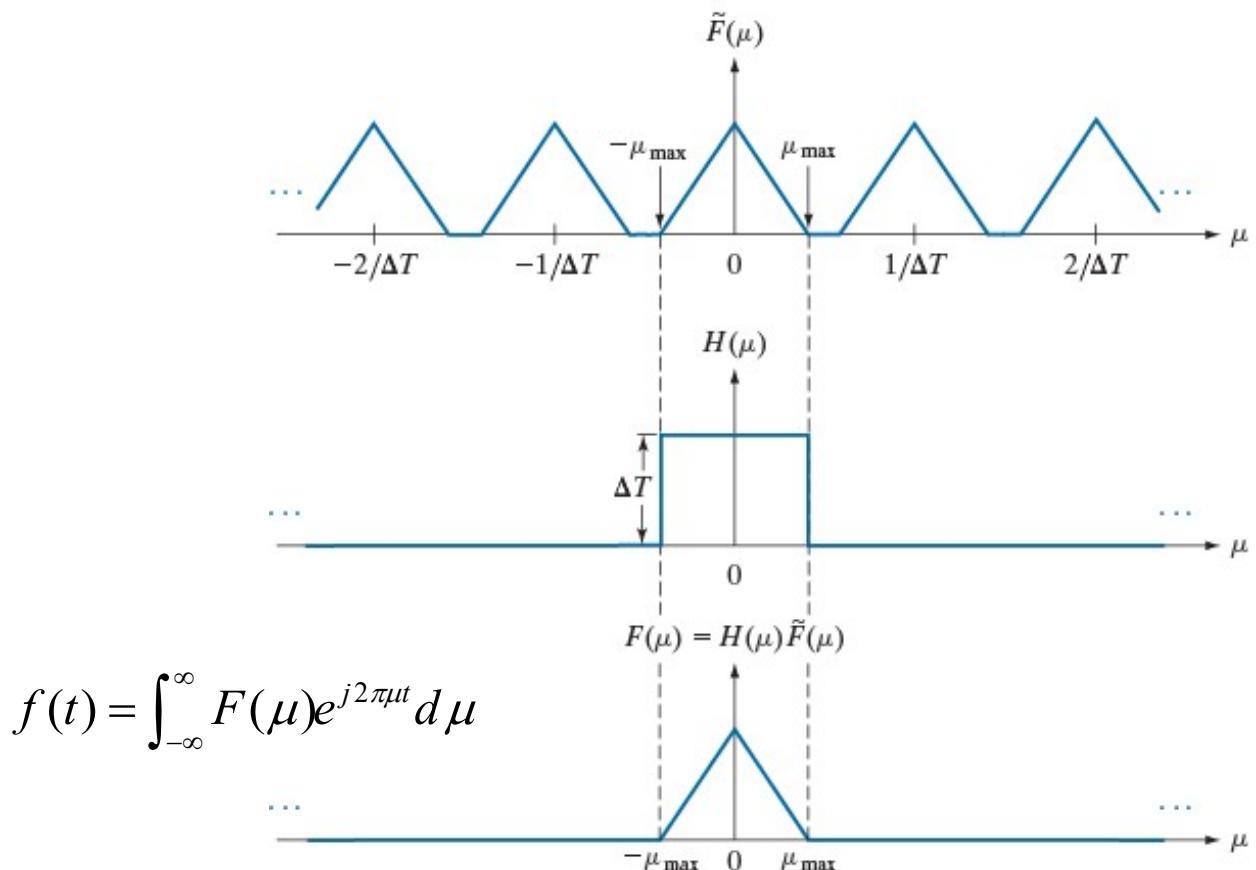
$$\frac{1}{\Delta T} < 2\mu_{\max}$$

aliasing

## Nyquist–Shannon Sampling Theorem

- We can recover  $f(t)$  from its sampled version if we can isolate a copy of  $F(\mu)$  from the periodic sequence of copies of this function contained in  $\tilde{F}(\mu)$ , the transform of the sampled function  $\tilde{f}(t)$ .
  - Sufficient separation is guaranteed if  $\frac{1}{\Delta T} > 2\mu_{\max}$ .
- **Sampling theorem:** A continuous, band-limited function can be recovered completely from a set of its samples if the samples are acquired at a rate exceeding twice the highest frequency content of the function.

## Nyquist–Shannon Sampling Theorem



a  
b  
c

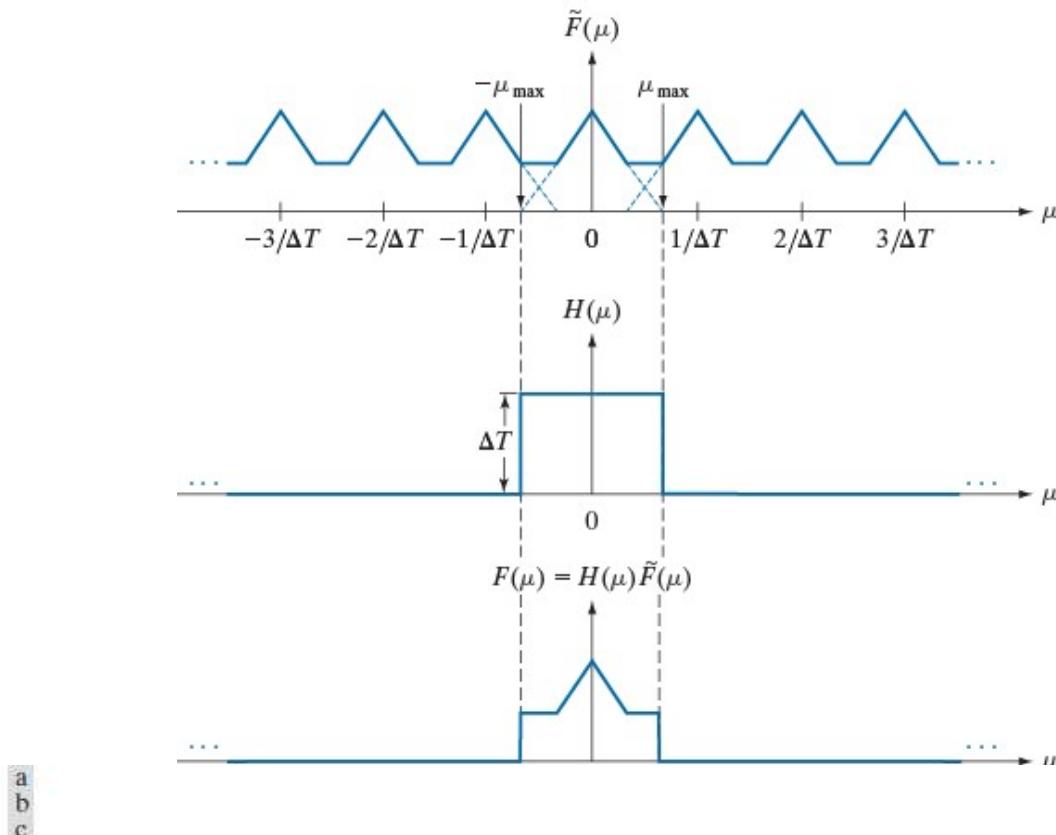
**FIGURE 4.8**  
 (a) Fourier transform of a sampled, band-limited function.  
 (b) Ideal lowpass filter transfer function.  
 (c) The product of (b) and (a), used to extract one period of the infinitely periodic sequence in (a).

## Aliasing

If a band-limited function is sampled at a rate that is less than twice its highest frequency?

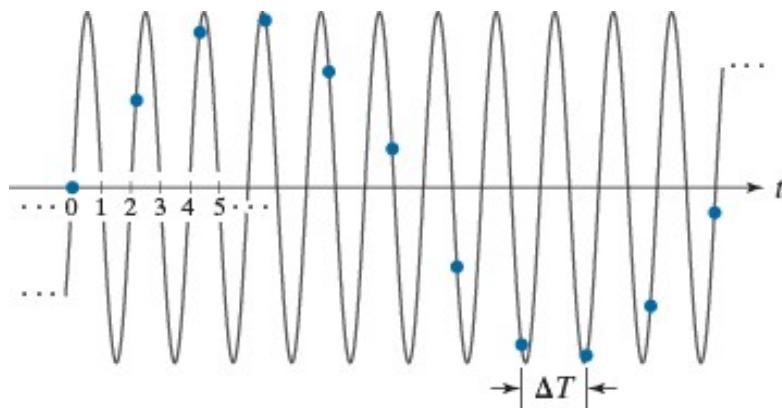
The inverse transform will yield a corrupted function. This effect is known as *frequency aliasing* or simply as *aliasing*.

## Aliasing



**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  $f(t)$ .

## Aliasing



**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.  $\Delta T$  is the separation between samples.

## Function Reconstruction from Sampled Data

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

$$\begin{aligned} f(t) &= \mathcal{I}^{-1} \{ F(\mu) \} \\ &= \mathcal{I}^{-1} \{ H(\mu)\tilde{F}(\mu) \} \\ &= h(t) \star \tilde{f}(t) \end{aligned}$$

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

## The Discrete Fourier Transform (DFT) of One Variable

$$F(\mu) = \sum_{x=0}^{M-1} f(x) e^{-j2\pi\mu x/M}, \quad \mu = 0, 1, \dots, M-1$$

$$f(x) = \frac{1}{M} \sum_{\mu=0}^{M-1} F(\mu) e^{j2\pi\mu x/M}, \quad x = 0, 1, 2, \dots, M-1$$

## 2-D Impulse and Sifting Property: Continuous

The impulse  $\delta(t, z)$ ,

$$\delta(t, z) = \begin{cases} \infty & \text{if } t = z = 0 \\ 0 & \text{otherwise} \end{cases}$$

and  $\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(t, z) dt dz = 1$

The sifting property

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t, z) \delta(t, z) dt dz = f(0, 0)$$

and

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t, z) \delta(t - t_0, z - z_0) dt dz = f(t_0, z_0)$$

## 2-D Impulse and Sifting Property: Discrete

The impulse  $\delta(x, y)$ ,

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y = 0 \\ 0 & \text{otherwise} \end{cases}$$

The sifting property

$$\sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} f(x, y) \delta(x, y) = f(0, 0)$$

and

$$\sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} f(x, y) \delta(x - x_0, y - y_0) = f(x_0, y_0)$$

## 2-D Fourier Transform: Continuous

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

and

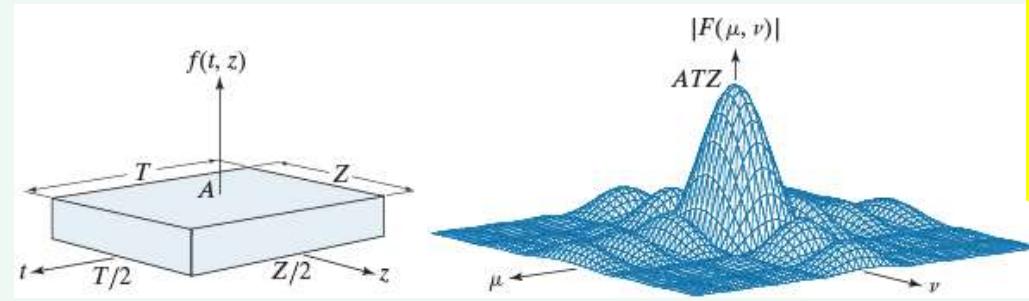
$$f(t, z) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\mu, \nu) e^{j2\pi(\mu t + \nu z)} d\mu d\nu$$

## 2-D Fourier Transform: Continuous

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  $\mu$ -axis.



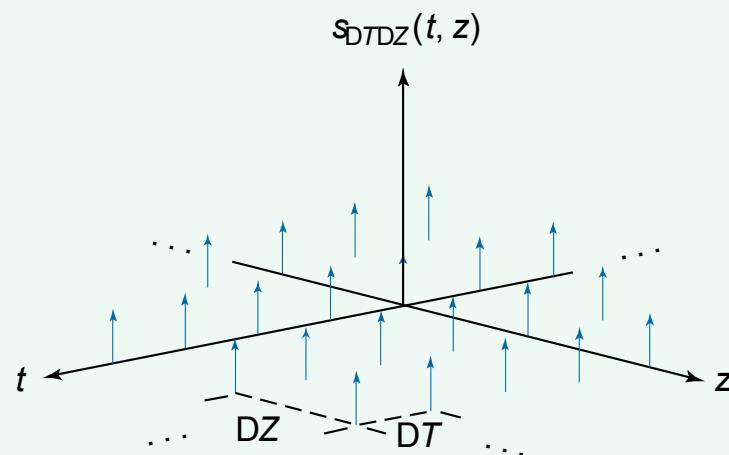
The locations of zeros in the spectrum are inversely proportional to  $T$  and  $Z$ . So the spectrum is more contracted along the  $\mu$ -axis.

$$\begin{aligned}
 F(\mu, \nu) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t, z) e^{-j2\pi(\mu t + \nu z)} dt dz \\
 &= \int_{-T/2}^{T/2} \int_{-Z/2}^{Z/2} A e^{-j2\pi(\mu t + \nu z)} dt dz \\
 &= ATZ \left[ \frac{\sin(\pi\mu T)}{\pi\mu T} \right] \left[ \frac{\sin(\pi\nu Z)}{\pi\nu Z} \right]
 \end{aligned}$$

## 2-D Sampling and 2-D Sampling Theorem

2-D impulse train:

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



**FIGURE 4.15**  
2-D impulse train.

## 2-D Sampling and 2-D Sampling Theorem

Function  $f(t, z)$  is said to be band-limited if its Fourier transform is 0 outside a rectangle established by the intervals  $[-\mu_{\max}, \mu_{\max}]$  and  $[-\nu_{\max}, \nu_{\max}]$ , that is

$$F(\mu, \nu) = 0 \text{ for } |\mu| \geq \mu_{\max} \text{ and } |\nu| \geq \nu_{\max}$$

Two-dimensional sampling theorem:

A continuous, band-limited function  $f(t, z)$  can be recovered with no error from a set of its samples if the sampling intervals are

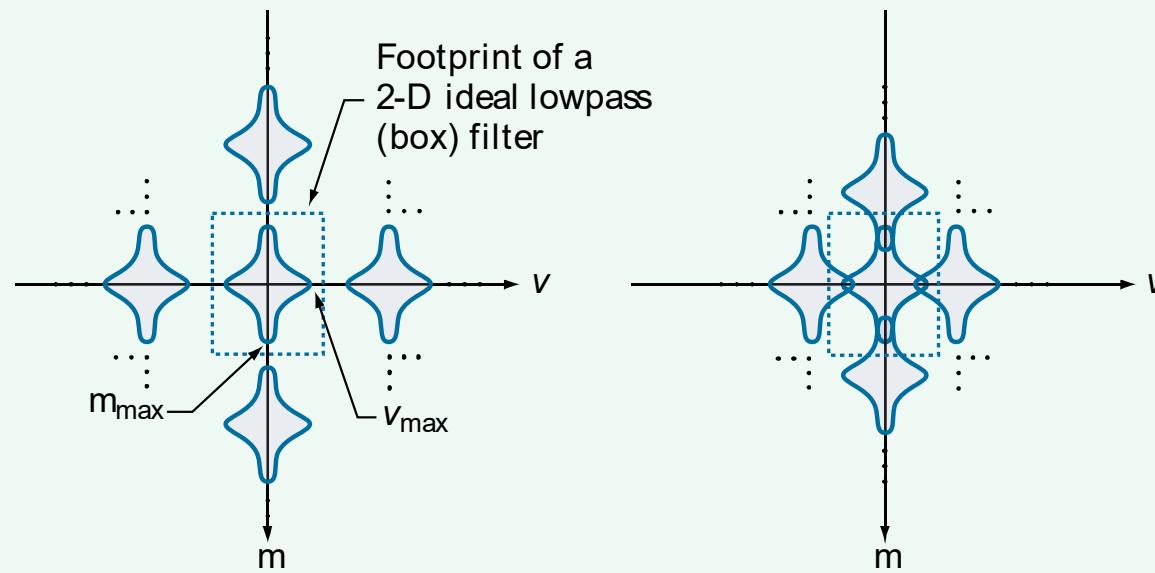
$$\Delta T < \frac{1}{2\mu_{\max}} \text{ and } \Delta Z < \frac{1}{2\nu_{\max}}$$

## 2-D Sampling and 2-D Sampling Theorem

a b

**FIGURE 4.16**

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



# Summary of FS and FT Expressions

	Continuous time		Discrete time	
	Time domain	Frequency domain	Time domain	Frequency domain
Fourier Series	$x(t) = \sum_{k=-\infty}^{+\infty} a_k e^{jk\omega_0 t}$	$a_k = \frac{1}{T_0} \int_{T_0} x(t) e^{-jk\omega_0 t} dt$	$x[n] = \sum_{k=(N)} a_k e^{jk(2\pi/N)n}$	$a_k = \frac{1}{N} \sum_{k=(N)} x[n] e^{-jk(2\pi/N)m}$
	continuous time periodic in time	discrete frequency aperiodic in frequency	discrete time periodic in time	discrete frequency periodic in frequency
Fourier Transform	$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(j\omega) e^{j\omega t} d\omega$	$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt$	$x[n] = \frac{1}{2\pi} \int_{2\pi} X(e^{j\omega}) e^{j\omega n} d\omega$	$X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x[n] e^{-j\omega n}$
	continuous time aperiodic in time	continuous frequency aperiodic in frequency	discrete time aperiodic in time	continuous frequency periodic in frequency

# Summary

Transform	Input Signal		Output Spectrum	
	Periodicity	Image domain	Periodicity	Spectral domain
CT FS	periodic	continuous	aperiodic	discrete
DT FS	periodic	discrete	periodic	discrete
CT FT	aperiodic	continuous	aperiodic	continuous
DT FT	aperiodic	discrete	periodic	continuous
DFT	aperiodic	discrete	periodic	discrete



## 2-D Discrete Fourier Transform and Its Inverse

DFT:

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

$\mu, \nu = 0, 1, 2, \dots, M-1$ ;

$f(x, y)$ : an  $M \times N$  digital image.

IDFT:

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

## Spatial and Frequency Intervals

Let  $\Delta T$  and  $\Delta Z$  denote the separations between samples, then the separations between the corresponding discrete, frequency domain variables are given by

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

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



## Translation and Rotation

$$\underbrace{f(x, y)e^{j2\pi(\mu_0x/M + \nu_0y/N)}}_{\text{做 rotate}} \Leftrightarrow F(\mu - \mu_0, \nu - \nu_0)$$
$$f(x - x_0, y - y_0) \Leftrightarrow F(\mu, \nu)e^{-j2\pi(\mu x_0/M + \nu y_0/N)}$$

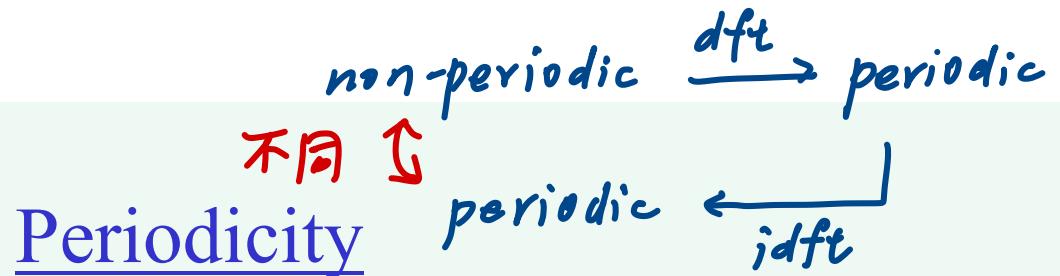
*做 shift*

Using the polar coordinates

$$x = r \cos \theta \quad y = r \sin \theta \quad \mu = \omega \cos \varphi \quad \nu = \omega \sin \varphi$$

results in the following transform pair:

$$f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$$



2-D DFT and its inverse are infinitely **periodic**:

$$F(\mu, \nu) = F(\mu + k_1 M, \nu) = F(\mu, \nu + k_2 N) = F(\mu + k_1 M, \nu + k_2 N)$$

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

**Shifting** the Fourier transform:

$$f(x)e^{j2\pi(\mu_0 x/M)} \Leftrightarrow F(\mu - \mu_0)$$

$$\mu_0 = M/2, \quad f(x)(-1)^x \Leftrightarrow F(\mu - M/2)$$

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

# Symmetry Properties of 2-D DFT

**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 <sup>†</sup>	Frequency Domain <sup>†</sup>
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

<sup>†</sup>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.

# Summary of DFT Definitions

**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) Spectrum	$ F(u, v)  = [R^2(u, v) + I^2(u, v)]^{1/2} \quad R = \text{Real}(F); I = \text{Imag}(F)$
4) Phase angle	$\phi(u, v) = \tan^{-1} \left[ \frac{I(u, v)}{R(u, v)} \right]$
5) Polar representation	$F(u, v) =  F(u, v)  e^{j\phi(u, v)}$
6) Power spectrum	$P(u, v) =  F(u, v) ^2$
7) Average value	$\bar{f} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) = \frac{1}{MN} F(0, 0)$
8) Periodicity ( $k_1$ and $k_2$ are integers)	$F(u, v) = F(u + k_1 M, v) = F(u, v + k_2 N) \\ = F(u + k_1, v + k_2 N)$ $f(x, y) = f(x + k_1 M, y) = f(x, y + k_2 N) \\ = f(x + k_1 M, y + k_2 N)$

## Summary of DFT Definitions

9) Convolution

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

10) Correlation

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

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.

12) Obtaining the IDFT using a DFT algorithm

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

This equation indicates that inputting  $F^*(u, v)$  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.

## Summary of DFT pairs

**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	$f(r,\theta + \theta_0) \Leftrightarrow F(\omega,\varphi + \theta_0)$ $r = \sqrt{x^2 + y^2} \quad \theta = \tan^{-1}(y/x) \quad \omega = \sqrt{u^2 + v^2} \quad \varphi = \tan^{-1}(v/u)$
6) Convolution theorem <sup>†</sup>	$f \star h)(x,y) \Leftrightarrow (F \bullet H)(u,v)$ $(f \bullet h)(x,y) \Leftrightarrow (1/MN)[(F \star H)(u,v)]$
7) Correlation theorem <sup>†</sup>	$(f \diamondsuit h)(x,y) \Leftrightarrow (F^* \bullet H)(u,v)$ $(f^* \bullet h)(x,y) \Leftrightarrow (1/MN)[(F \diamondsuit H)(u,v)]$

## Summary of DFT pairs

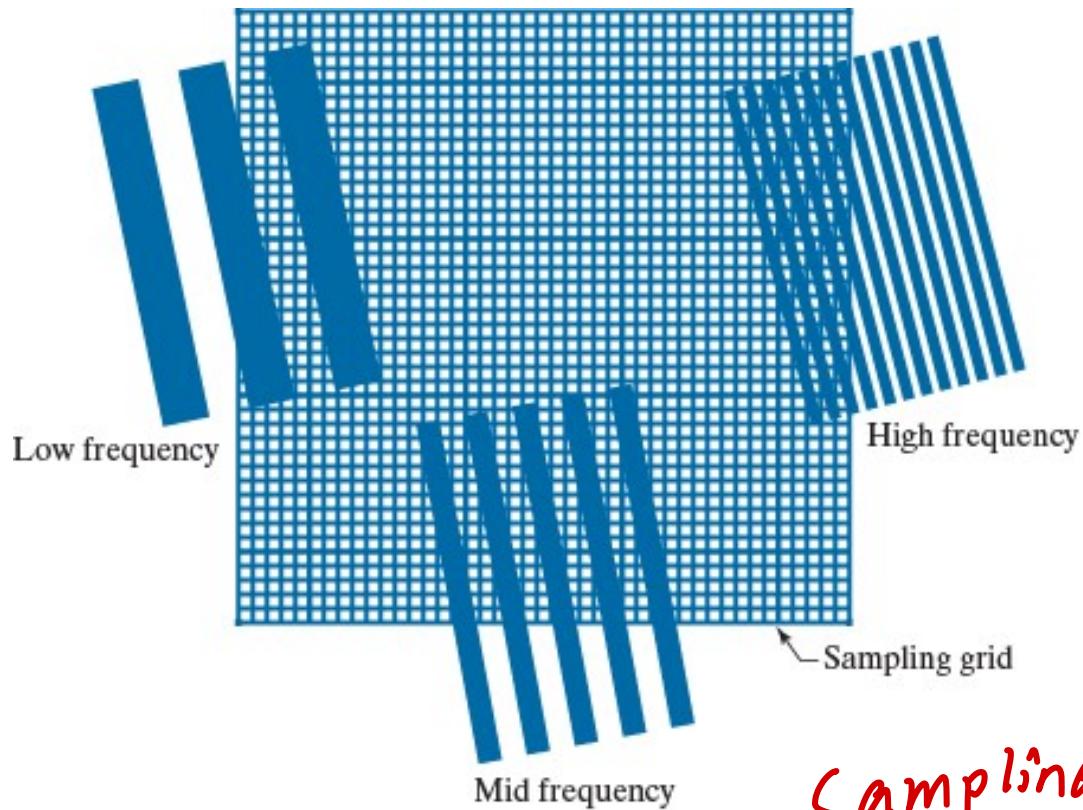
8)	Discrete unit impulse	$\delta(x,y) \Leftrightarrow 1$ $1 \Leftrightarrow MN\delta(u,v)$
9)	Rectangle	$\text{rec}[a,b] \Leftrightarrow ab \frac{\sin(\pi ua)}{(\pi ua)} \frac{\sin(\pi vb)}{(\pi vb)} e^{-j\pi(ua+vb)}$
10)	Sine	$\sin(2\pi u_0 x/M + 2\pi v_0 y/N) \Leftrightarrow \frac{jMN}{2} [\delta(u+u_0, v+v_0) - \delta(u-u_0, v-v_0)]$
11)	Cosine	$\cos(2\pi u_0 x/M + 2\pi v_0 y/N) \Leftrightarrow \frac{1}{2} [\delta(u+u_0, v+v_0) + \delta(u-u_0, v-v_0)]$
12)	Differentiation (the expressions on the right assume that $f(\pm\infty, \pm\infty) = 0$ .	$\left(\frac{\partial}{\partial t}\right)^m \left(\frac{\partial}{\partial z}\right)^n f(t,z) \Leftrightarrow (j2\pi\mu)^m (j2\pi\nu)^n F(\mu,\nu)$ $\frac{\partial^m f(t,z)}{\partial t^m} \Leftrightarrow (j2\pi\mu)^m F(\mu,\nu); \quad \frac{\partial^n f(t,z)}{\partial z^n} \Leftrightarrow (j2\pi\nu)^n F(\mu,\nu)$
13)	Gaussian	$A2\pi\sigma^2 e^{-2\pi^2\sigma^2(t^2+z^2)} \Leftrightarrow Ae^{-(\mu^2+\nu^2)/2\sigma^2}$ ( $A$ is a constant)

<sup>†</sup> Assumes that  $f(x,y)$  and  $h(x,y)$  have been properly padded. Convolution is associative, commutative, and distributive. Correlation is distributive (see Table 3.5). The products are elementwise products (see Section 2.6).

# Aliasing

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.



Sampling by freq  
不够高

## Aliasing in Images: Example

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.

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

Under-sampling

## Aliasing in Images: Example



**FIGURE 4.19** Illustration of aliasing on resampled natural images. (a) A digital image of size  $772 \times 548$  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.)

## *a special case of aliasing*

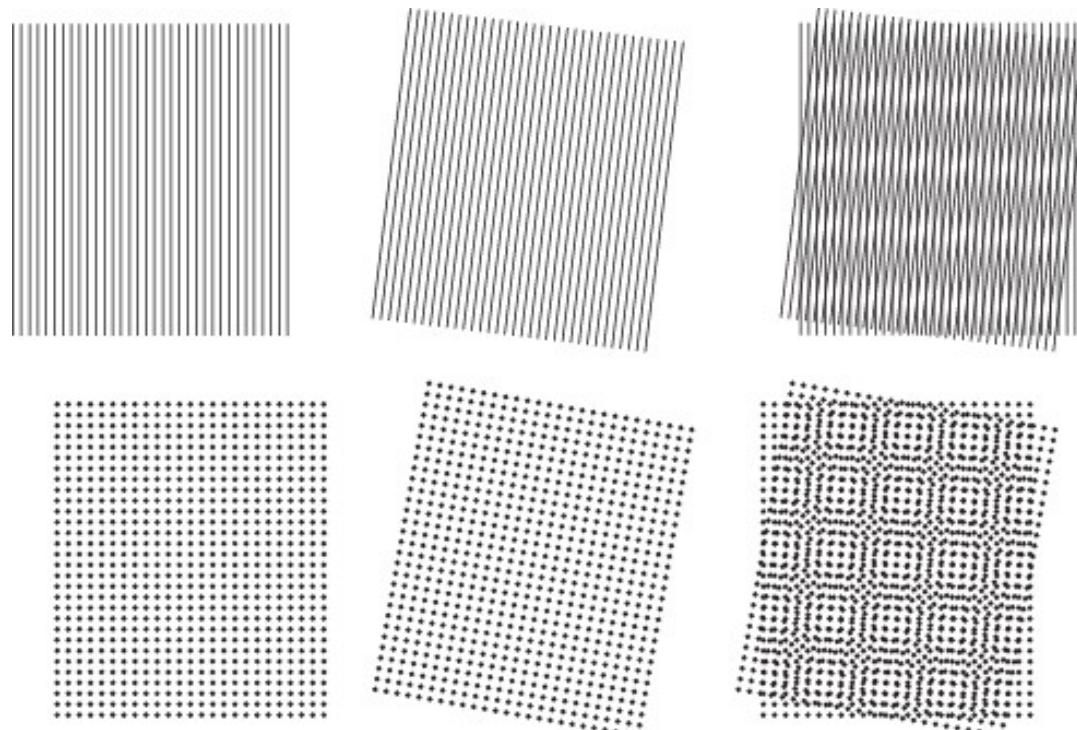
### Moiré Effect

- Moiré patterns are often produced by various digital imaging and computer graphics techniques, e. g., when scanning a halftone picture or ray tracing a checkered plane. **This cause of moiré is a special case of aliasing**, due to under-sampling a fine regular pattern. --Wikipedia

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.

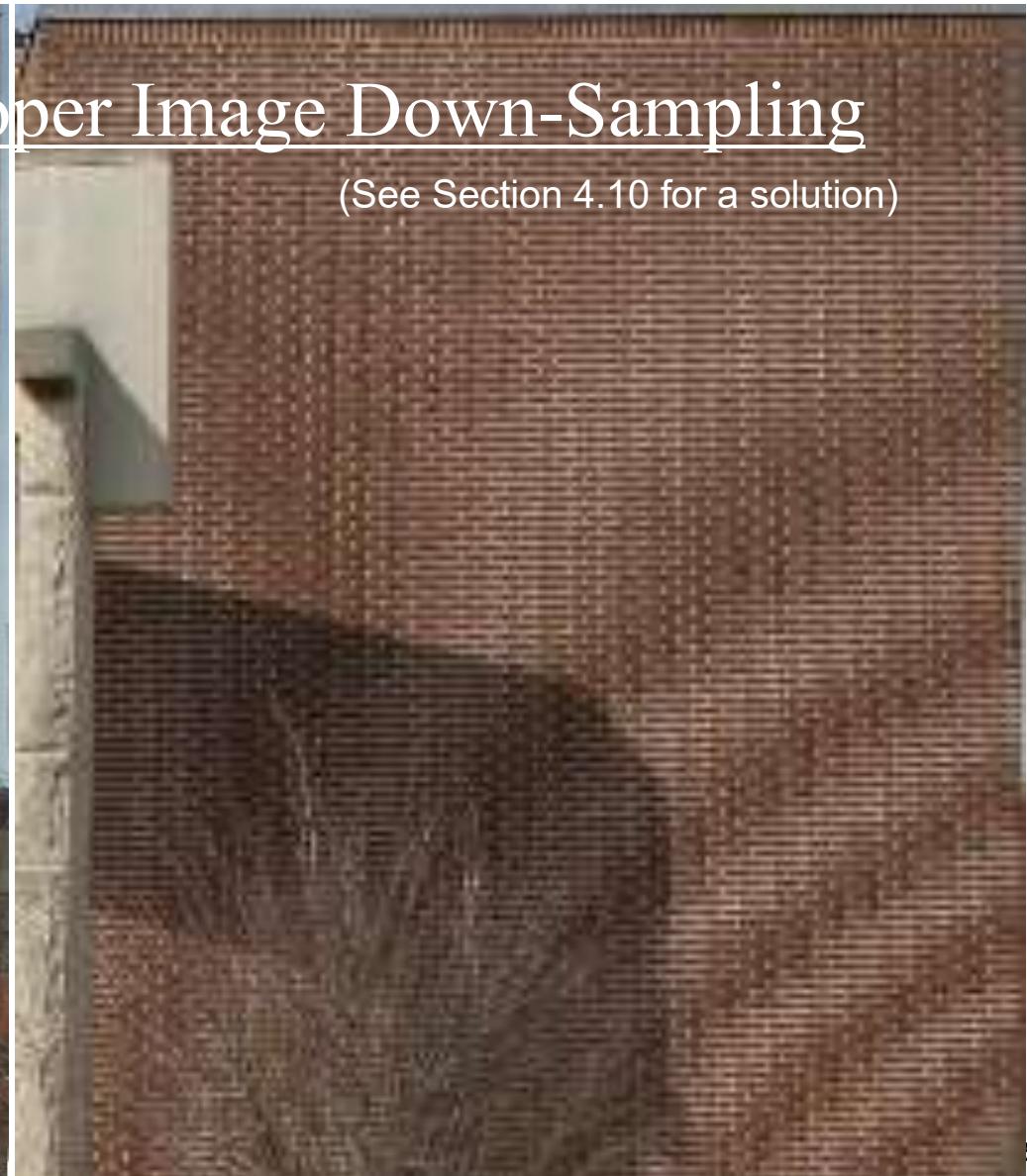


# Moiré Effect Due to Improper Image Down-Sampling



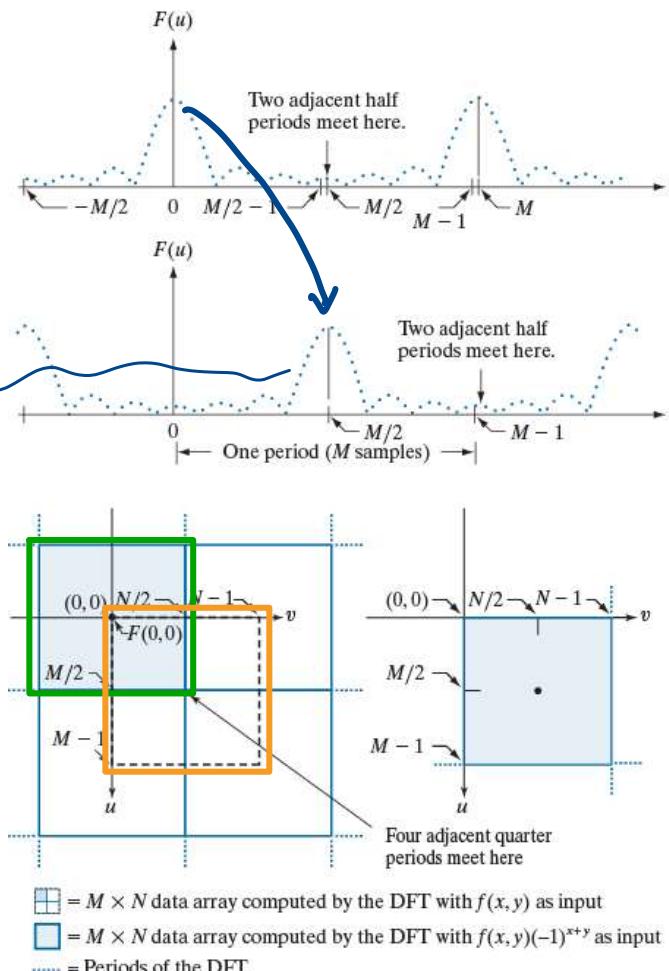
## Moiré Effect Due to Improper Image Down-Sampling

(See Section 4.10 for a solution)



# Centering the Fourier Transform

何點該改哪？  
希望 origin 在中央



a  
b  
c d

**FIGURE 4.22**  
Centering the Fourier transform.  
 (a) A 1-D DFT showing an infinite number of periods.  
 (b) Shifted DFT obtained by multiplying  $f(x)$  by  $(-1)^x$  before computing  $F(u)$ .  
 (c) A 2-D DFT showing an infinite number of periods. The area within the dashed rectangle is the data array,  $F(u,v)$ , obtained with Eq. (4-67) with an image  $f(x,y)$  as the input. This array consists of four quarter periods.  
 (d) Shifted array obtained by multiplying  $f(x,y)$  by  $(-1)^{x+y}$  before computing  $F(u,v)$ . The data now contains one complete, centered period, as in (b).

希望能  $\square$  shift 到  $\square$   
而不是直接  $\square$  (而在 freq domain shift, 相當於原本的  
domain flip rotate)

## Even and Odd Symmetry

- Any real or complex function can be expressed as the sum of an even and an odd part

$$w(x, y) = w_e(x, y) + w_o(x, y)$$

$$w_e(x, y) \triangleq \frac{w(x, y) + w(-x, -y)}{2}, \quad w_e(x, y) = w_e(-x, -y)$$

$$w_o(x, y) \triangleq \frac{w(x, y) - w(-x, -y)}{2}, \quad w_o(x, y) = -w_o(-x, -y)$$

- Even functions are symmetric, odd functions are antisymmetric



## Even and Odd Symmetry for Images

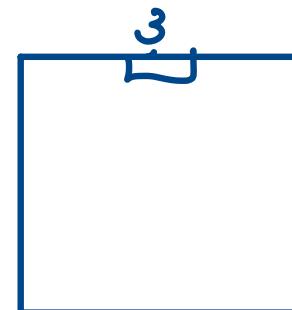
- Note: All indices in the DFT and IDFT are non-negative integers, so the symmetry for images is defined w.r.t. the center point.

$$w_e(x, y) = w_e(M - x, N - y)$$

$$w_o(x, y) = -w_o(M - x, N - y)$$

- Product of two even or two odd functions is even
- Product of an even and an odd function is odd
- A discrete function can be odd only if all its samples sum to zero
- Thus

$$\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} w_e(x, y) w_o(x, y) = 0$$



哪個是？

### Example 4.10: Even and Odd Functions

Consider  $f = \{f(0), f(1), f(2), f(3)\} = \{2, 1, 1, 1\}$ .

*when M=4* To be even,  $f(x) = f(4-x)$ ,  $x = 0, 1, 2, 3$ , which requires

$$f(0) = f(4), f(1) = f(3), f(2) = f(2), f(3) = f(1).$$

$f(0)$  is immaterial because  $f(4)$  is outside the range of consideration.

Therefore, the function  $f$  is even.

When  $M$  is odd, for  $f$  to be even,  $f(0)$  is arbitrary but the others must form pairs with equal values.

ex:  
 $\{2, 1, 1, 1, 1\}$

$M=5$ , 是 even

$$w_e(x, y) = w_e(M - x, N - y)$$

$$w_o(x, y) = -w_o(M - x, N - y)$$

## Example 4.10: Even and Odd Functions

The first term  $w_o(0)$  of an odd sequence  $w_o$  must be zero, because

$$w_o(0) = \frac{w(0) - w(-0)}{2} = 0.$$

↓      ↓  
0       $\frac{M}{2}$

Consider the 1-D sequence  $\{g(0), g(1), g(2), g(3)\} = \{0, -1, 0, 1\}$ .

To be odd,  $g(x) = -g(4-x)$ ,  $x = 0, 1, 2, 3$ , which requires

$$g(0) = 0, \quad g(1) = -g(3), \quad g(2) = -g(2), \quad g(3) = -g(1).$$

So any 4-point odd sequence has the form  $\{0, a, 0, -a\}$ .

In general, a 1-D odd sequence has the property that the points at 0 and  $M/2$  are always zero. When  $M$  is odd, the first point still has to be zero, but the others must form pairs with equal value but opposite signs.

## Example 4.10: Even and Odd Functions

Whether a function is odd or even plays a key role in interpreting the results based on DFTs. Consider the  $6 \times 6$  array (a Sobel kernel) with center at (3,3)

$$\begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -2 & \mathbf{0} & 2 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

You can prove it is odd using Eq. 4-83. However, adding another row or column of 0's would make the matrix neither even nor odd.

In general, inserting a 2-D array of odd/even dimensions into a larger array of zeros of odd/even symmetry preserves the symmetry of the smaller array.

## Fourier Spectrum and Phase Angle

2-D DFT in polar form

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

Fourier spectrum

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

Power spectrum

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

Phase angle

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

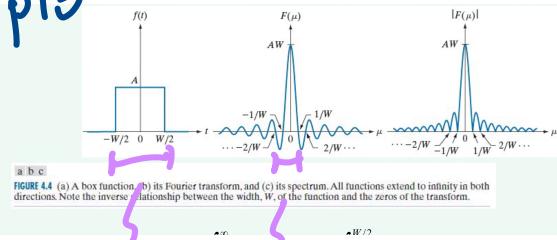


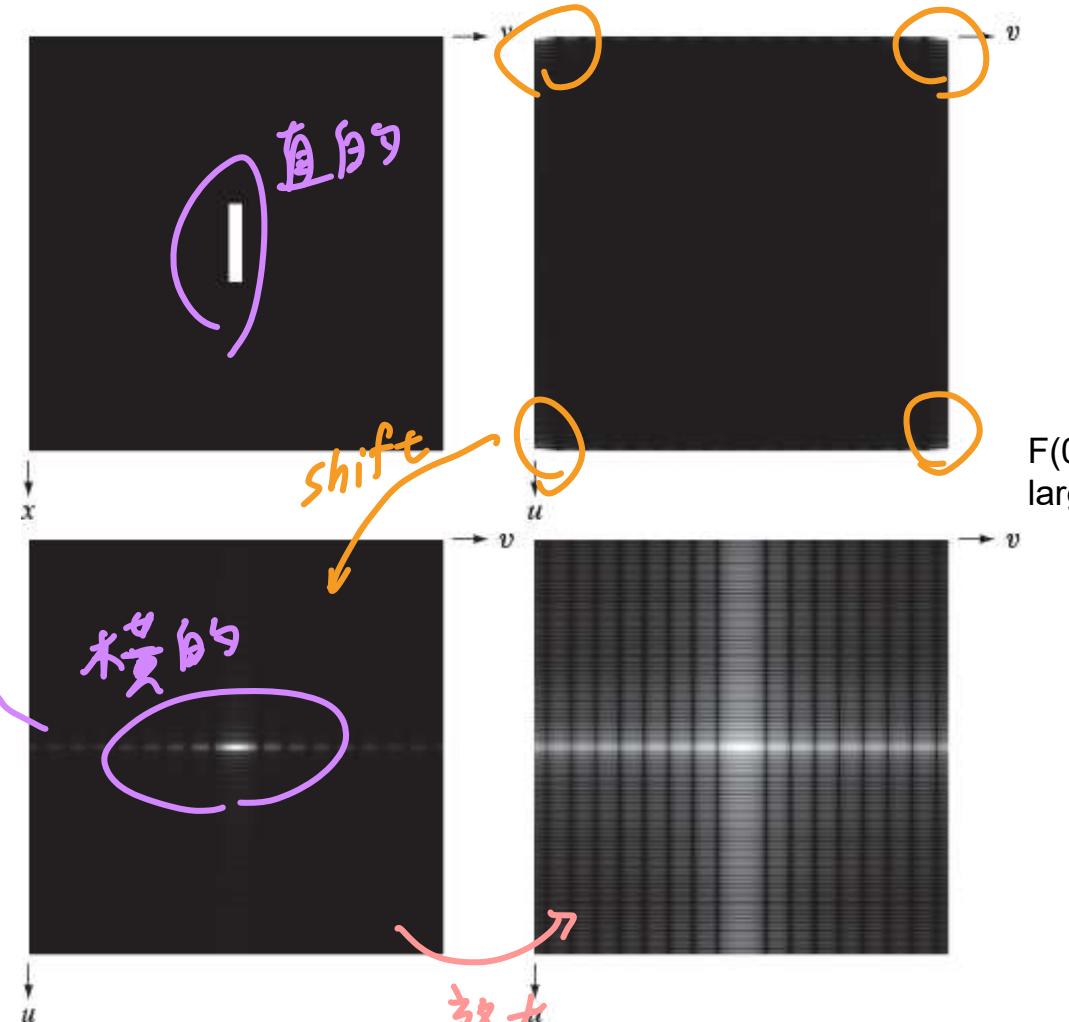
FIGURE 4.4 (a) A box function. (b) its Fourier transform, and (c) its spectrum. All functions extend to infinity in both directions. Note the inverse relationship between the width,  $W$ , of the function and the zeros of the transform.

頻率  
1/w會變窄

a  
b  
c  
d

**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.  
 (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).

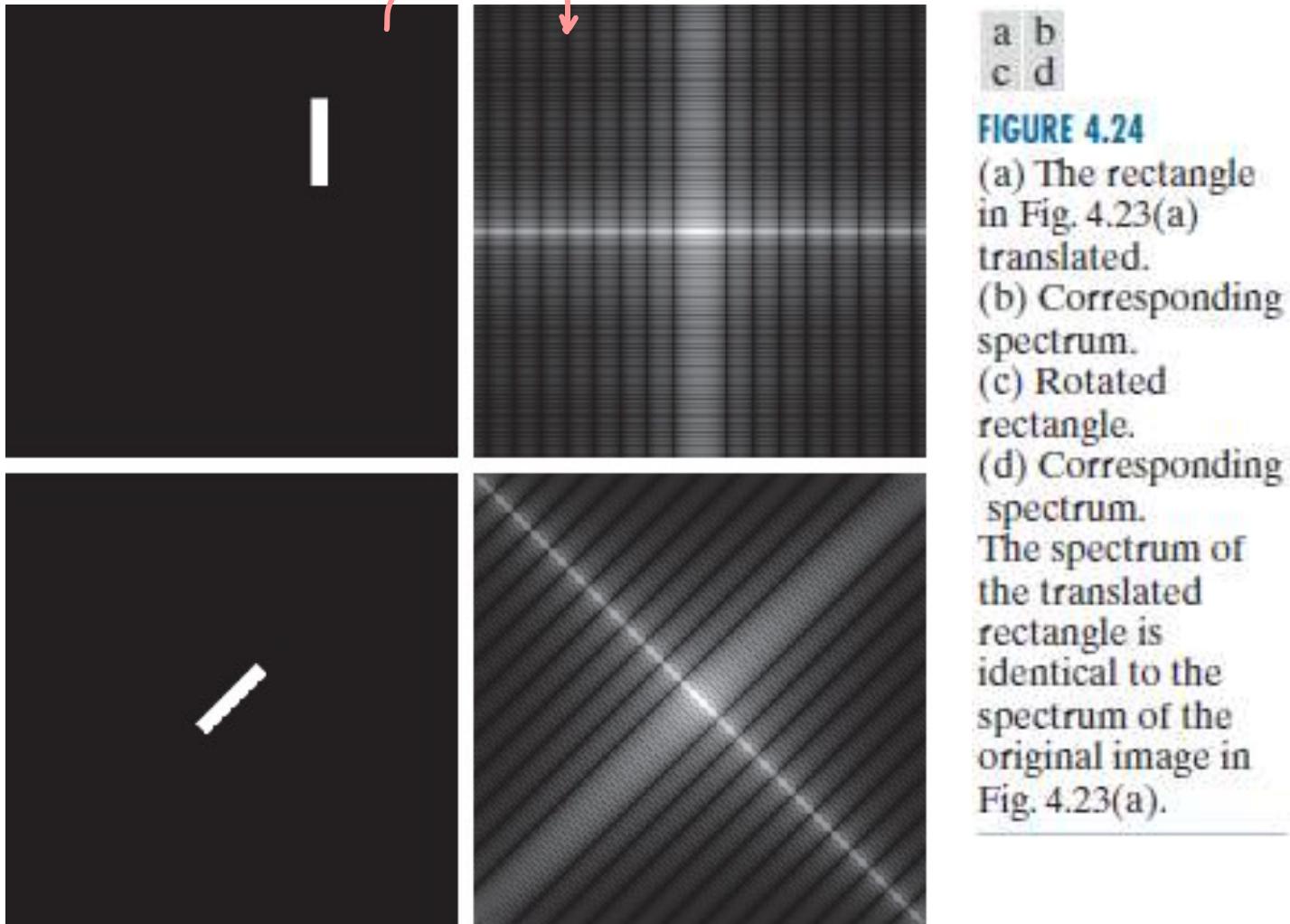
## Spectrum of a Rectangle Image



;(c)才會是模的

shift 不交叉

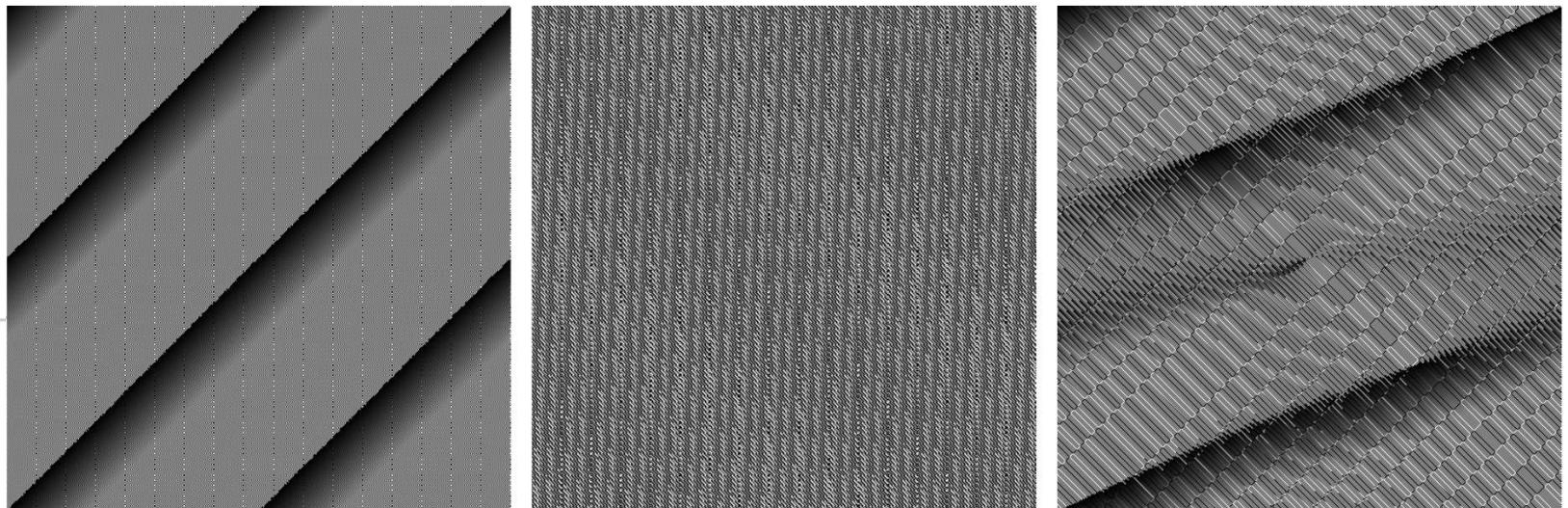
## Spectrum of a Rectangle Image



## Phase Angle of a Rectangle Image

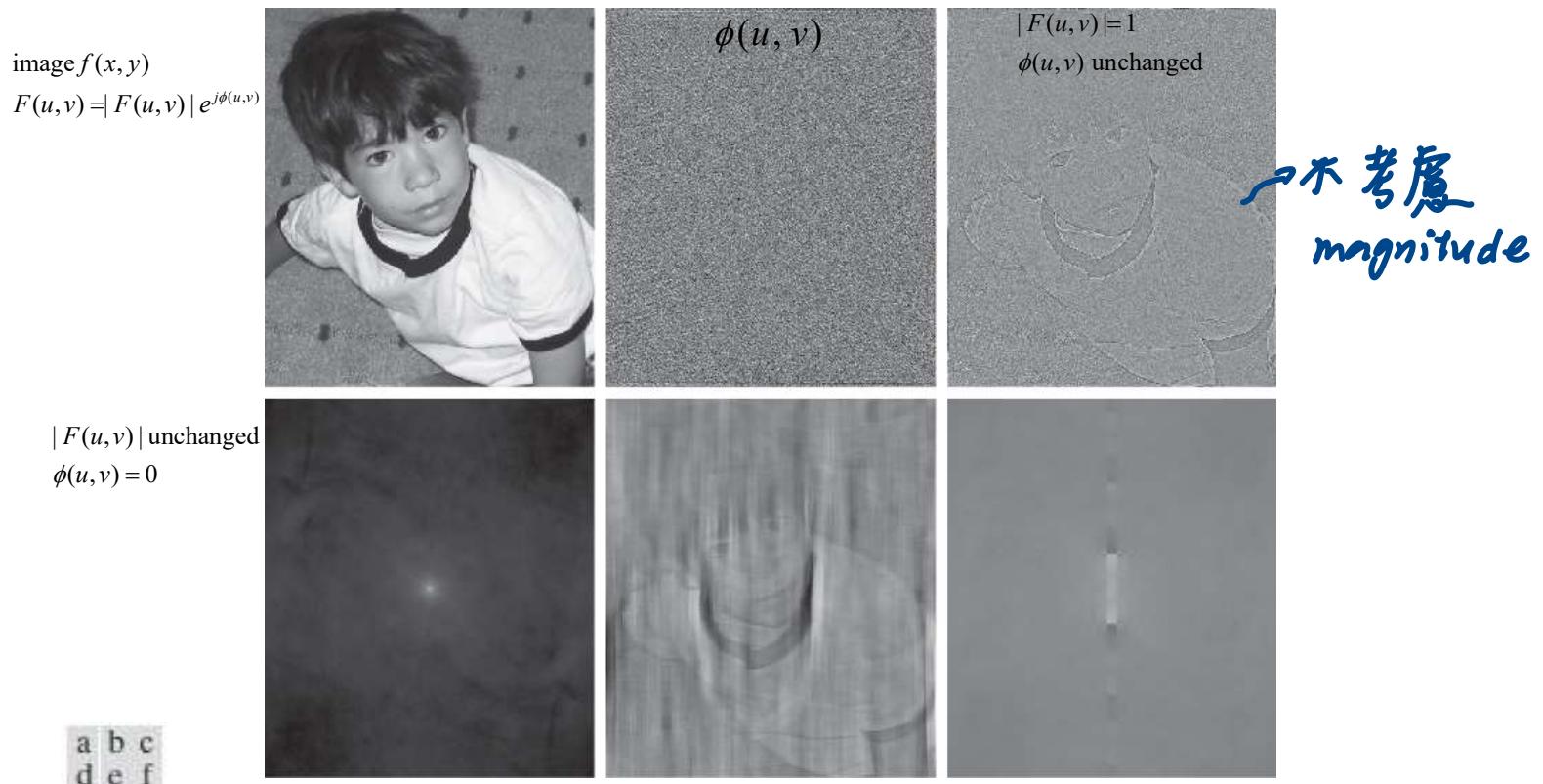
a b c

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



Note:  
phase info. 的重要性!

## Phase Angles and the Reconstructed Image



**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.

## 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

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

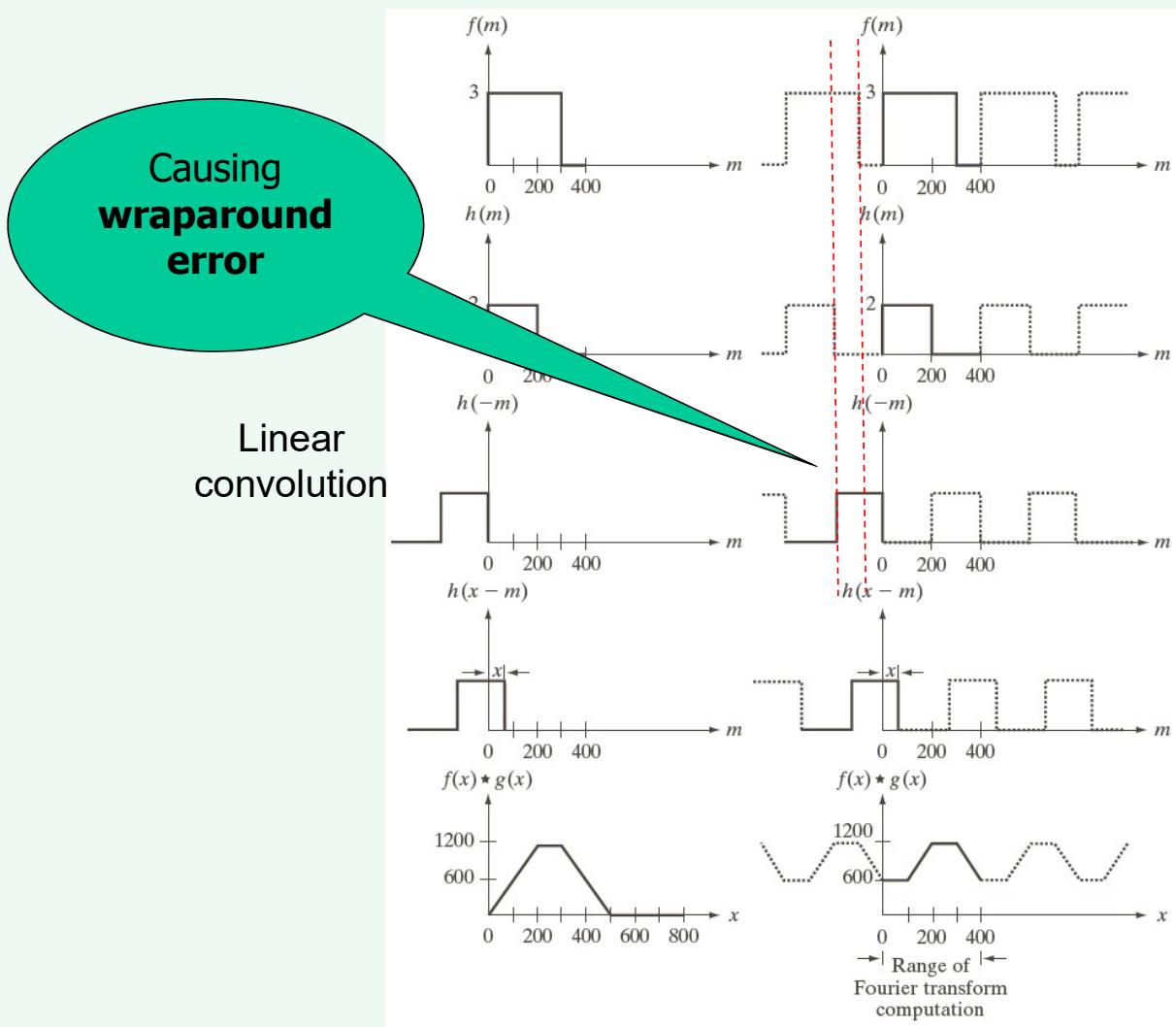
$$x = 0, 1, 2, \dots, M-1; y = 0, 1, 2, \dots, N-1.$$

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

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



## How to Make Spatial Domain Result = Frequency Domain Result



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.

## Zero Padding

- ▶ Can solve the wraparound error problem
- ▶ 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 for Images

- ▶  $f(x,y)$ : a  $A \times B$  image
- ▶  $h(x,y)$ : a  $C \times D$  image
- ▶ Wraparound error can be avoided by padding these images with zeros as follows:

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

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

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

## Basics of Filtering in the Frequency Domain

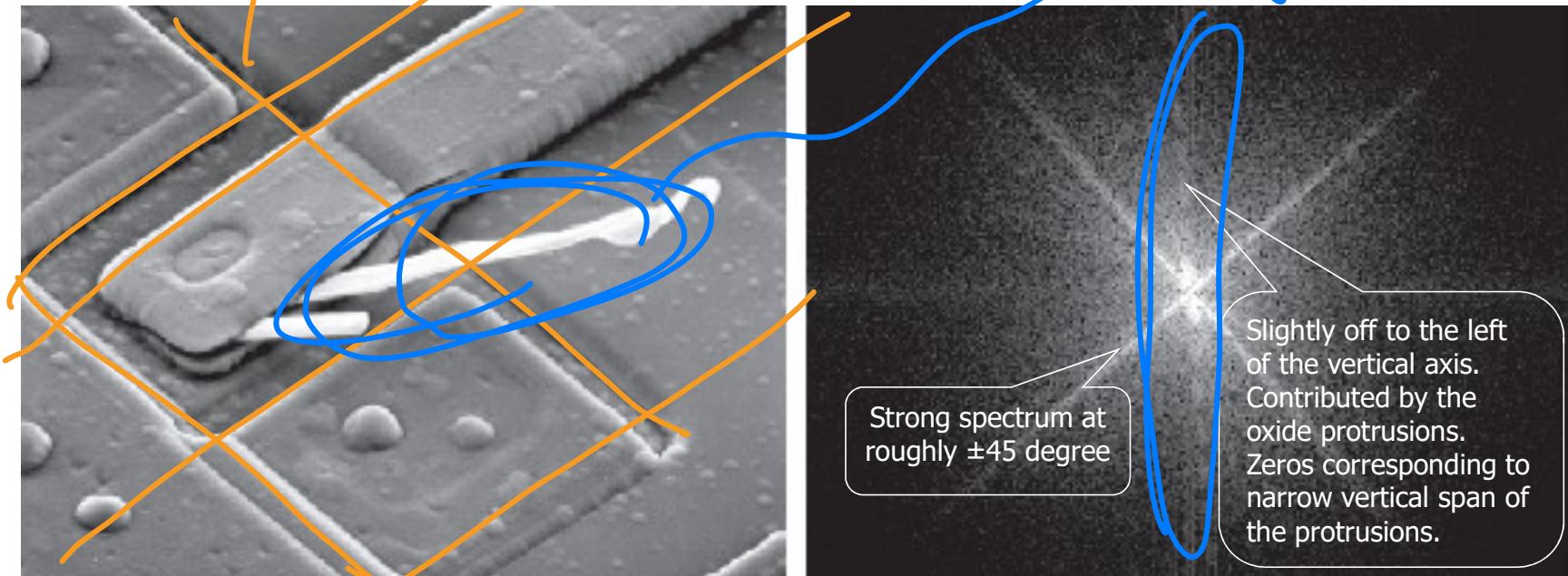
- ▶ Modify the DFT  $H(u,v)$  of an image by a filter  $F(u,v)$
- ▶ Compute the inverse transform to obtain the processed result

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

但現在社會出這個。  
它在 something is wrong

∴ edge 都是這兩  
方向，所以 dft 後，應該也是只有 2 方向

## Frequency Spectrum Revealing Intensity Characteristics



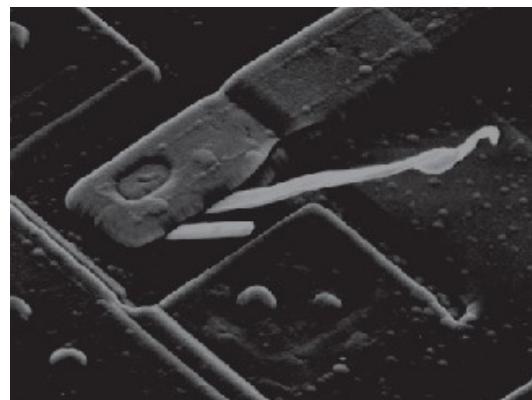
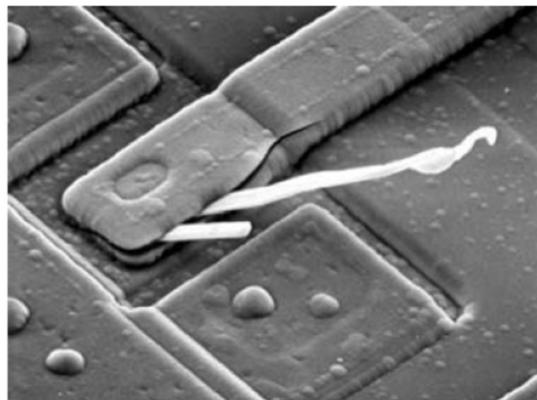
a b

Scanning electron microscope

**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.)

## Removal of DC Value

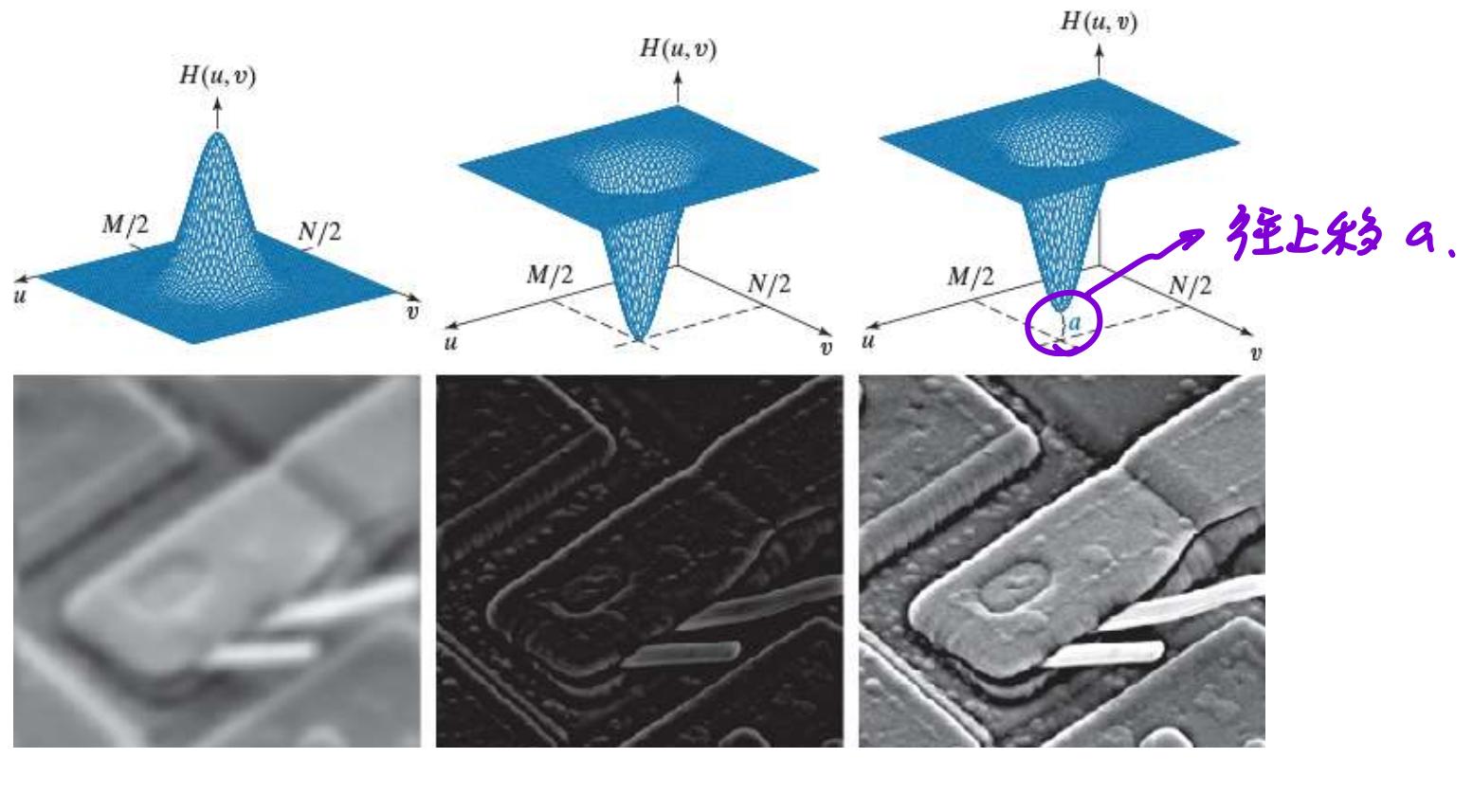
- ▶ Apply a filter  $H(u,v)$  that is 0 at the center of the transform and 1 elsewhere



All negative values  
are clipped to zero  
by the display.

**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.

## Lowpass, Highpass, and Offset Highpass Filtering



**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).

要會搞個3D圖!

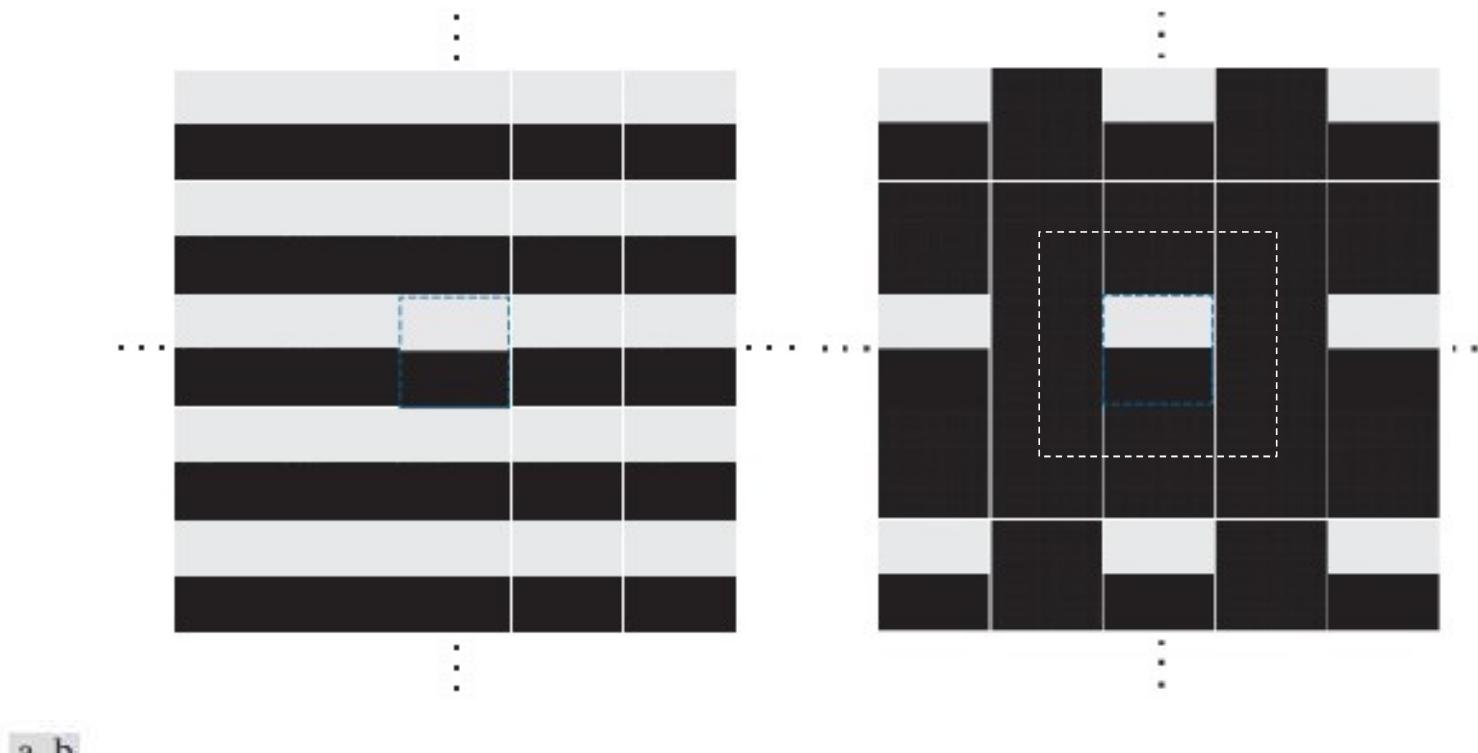
## Gaussian Lowpass Filtering



a | b | c

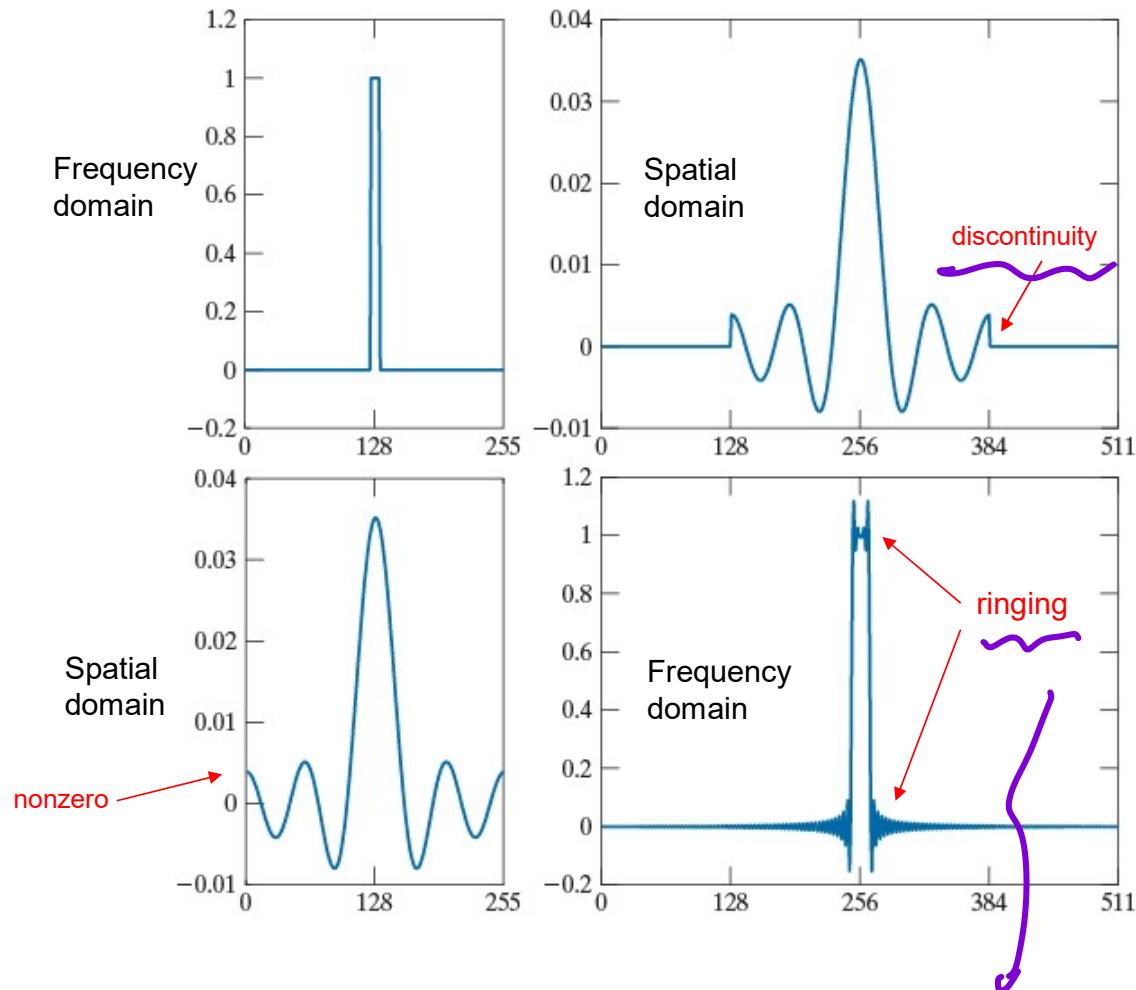
**FIGURE 4.31** (a) A simple image. (b) Result of blurring with a Gaussian lowpass filter without padding. (c) Result of lowpass filtering with zero padding. Compare the vertical edges in (b) and (c).

## Image Periodicity after Zero Padding



**FIGURE 4.32** (a) Image periodicity without image padding. (b) Periodicity after padding with 0's (black). The dashed areas in the center correspond to the image in Fig. 4.31(a). Periodicity is inherent when using the DFT. (The thin white lines in both images are superimposed for clarity; they are not part of the data.)

# Padding of Filter Transfer Function



a  
b  
c  
d

FIGURE 4.33

- (a) Filter transfer function specified in the (centered) frequency domain.  
 (b) Spatial representation (filter kernel) obtained by computing the IDFT of (a).  
 (c) Result of padding (b) to twice its length (note the discontinuities).  
 (d) Corresponding filter in the frequency domain obtained by computing the DFT of (c). Note the ringing caused by the discontinuities in (c). Part (b) of the figure is below (a), and (d) is below (c).

Correct way:

1. Pad the image to size  $P \times Q$
2. Construct the filter of the same size in the Frequency domain

∴ 有值 padding

## Zero-Phase-Shift Filters

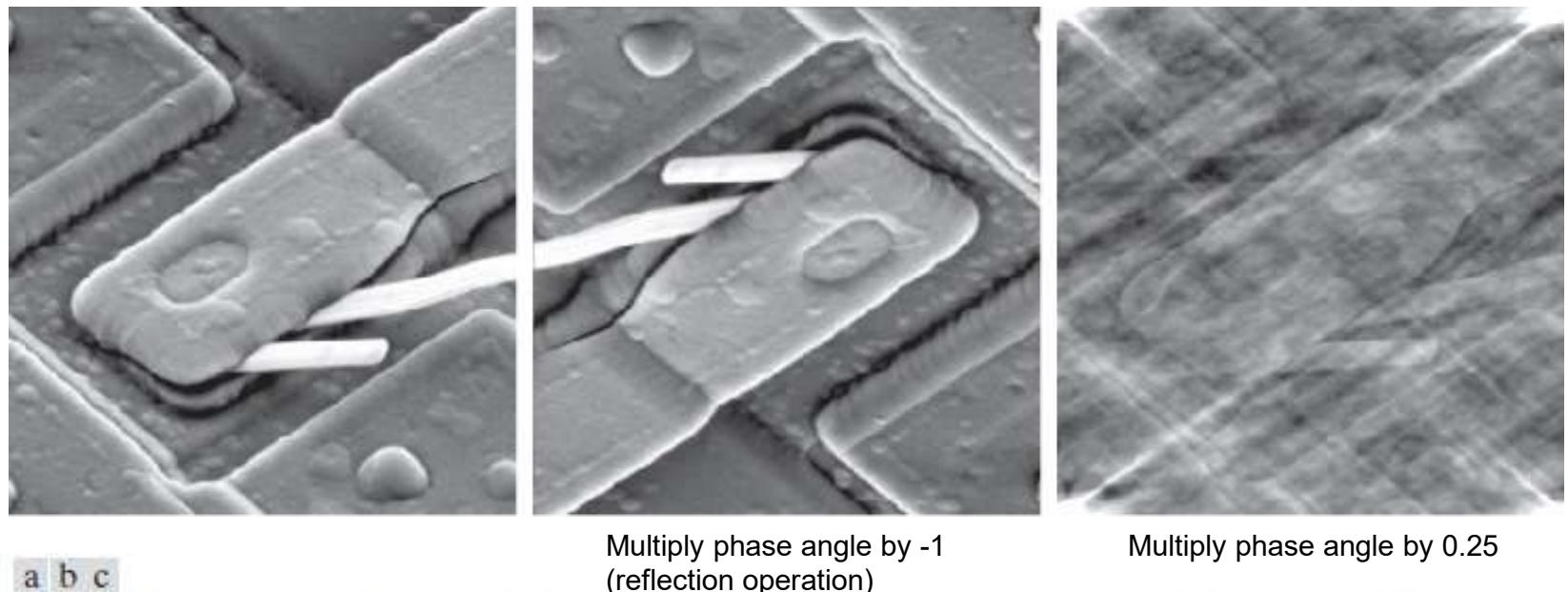
$$g(x, y) = \mathcal{I}^{-1}\{H(u, v)F(u, v)\}$$

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

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

**Zero-phase-shift** filters affect both real and imaginary parts equally and hence have no effect on the phase of image.

## Examples: Nonzero-Phase-Shift Filters



**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.

Even small changes in the phase angle can have dramatic (usually undesirable) effects on the filtered output

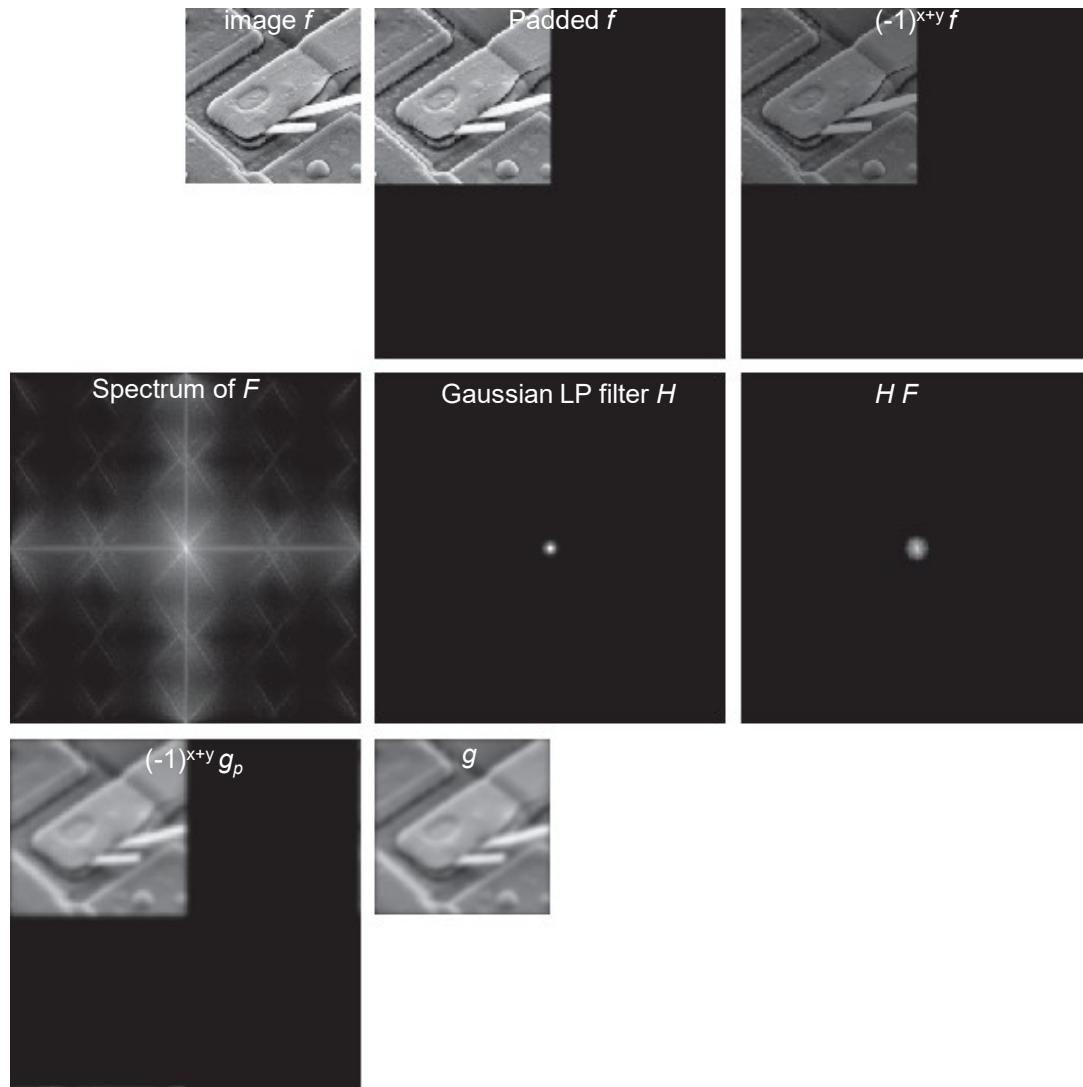
## Summary: Steps for Filtering in the Frequency Domain

1. Given an input image  $f(x,y)$  of size  $M \times N$ , obtain the padding parameters  $P$  and  $Q$ . Typically,  $P = 2M$  and  $Q = 2N$ . (DFT algorithms tend to run faster with even size)
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 obtained in Step 3
5. Construct a real, symmetric filter function  $H(u,v)$  of size  $P \times Q$  with center at  $(P/2, Q/2)$
6. Compute  $G(u,v) = H(u,v) F(u,v)$  by array multiplication
7. Apply the following operation to shift the image back

$$g_p(x,y) = \left\{ \text{real} \left[ \mathfrak{I}^{-1} [G(u,v)] \right] \right\} (-1)^{x+y}$$

8. Obtain  $g(x,y)$  by extracting the  $M \times N$  region from the top left quadrant of  $g_p(x,y)$

## Illustration of Filtering in the 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  $g_p$ .

## Correspondence Between Filtering in the Spatial and Frequency Domains (1)

- Convolution theorem is the link between them
- Corresponding filters form a transform pair

$$h(x) \Leftrightarrow H(u)$$

- Frequency-domain filtering can be 100x faster than spatial convolution
- Let  $H(u)$  denote the 1-D frequency domain Gaussian filter

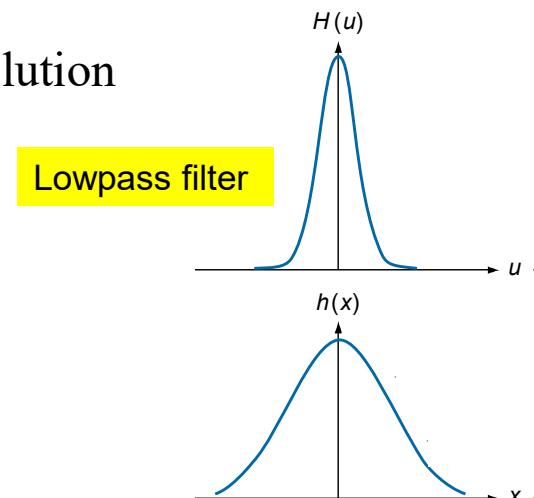
$$H(u) = Ae^{-u^2/2\sigma^2}$$

- The corresponding filter in the spatial domain is

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

Remarks:

1. Both functions of this transform pair are Gaussian and real → facilitates analysis
2. The functions behave reciprocally



## Correspondence Between Filtering in the Spatial and Frequency Domains (2)

- $f_h(x) = \delta(x) - h_l(x)$ ,  $F_h(u) = 1 - F_l(u)$
- Let  $H(u)$  denote the difference of Gaussian filters (DoG)

$$H(u) = Ae^{-u^2/2\sigma_1^2} - Be^{-u^2/2\sigma_2^2} \quad A \geq B \text{ and } \sigma_1 \geq \sigma_2$$

- The corresponding filter in the spatial domain is

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

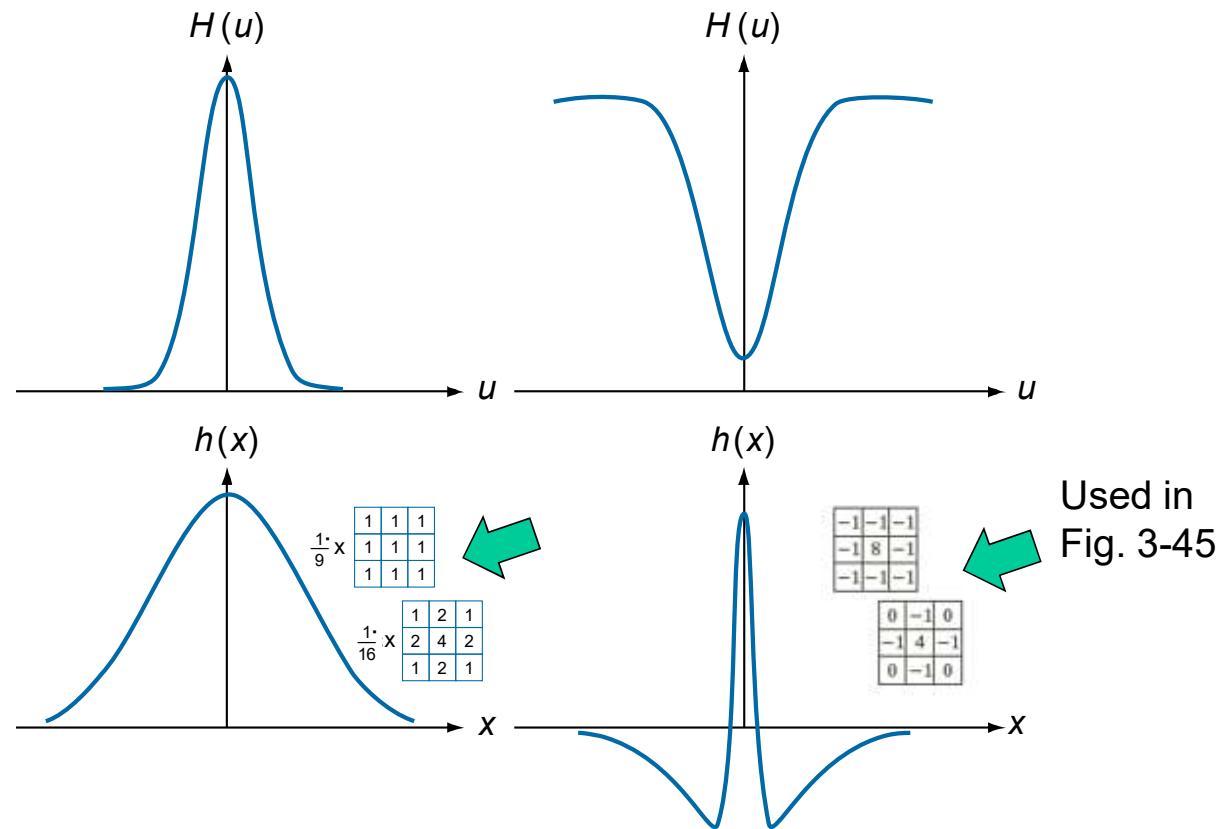
- Fig. 4.36(d) shows it has a positive center term and negative terms on either sides.

DoG preserves spatial information that lies between the range of frequencies that are preserved in two blurred images. Thus, the DoG is a spatial band-pass filter.  
--Wikipedia

## Correspondence Between Filtering in the Spatial and Frequency Domains (3)

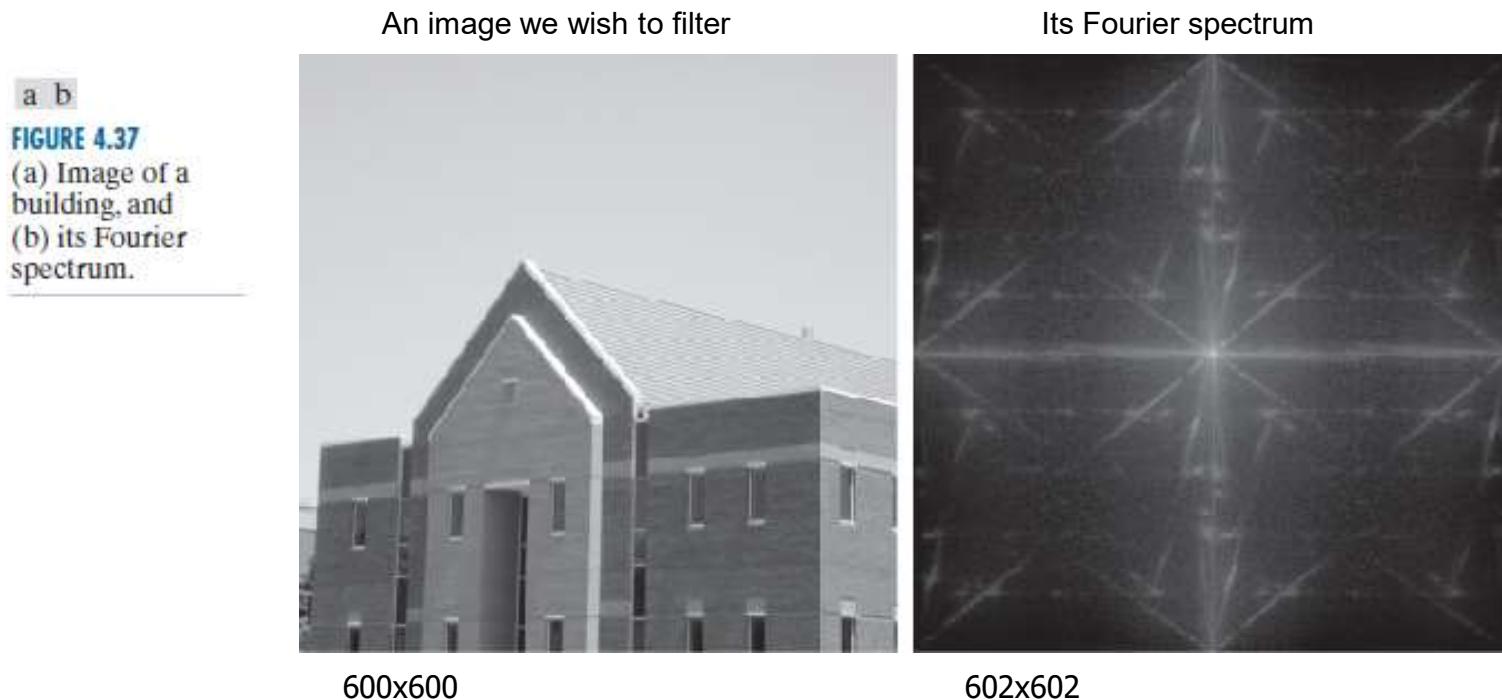
a c  
b d

**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 4.15

- Generate a “full” filter in the frequency domain from a small spatial (Sobel) kernel based on its mathematical property

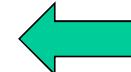


## Example 4.15 (cont'd)

Zero padding:

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

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



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

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

Procedure to generate spatial filter:

- Convert  $h$  to the smallest size that satisfies the odd symmetry requirement by adding a leading row and column of 0's, which makes it a 4x4 array.
- This will make the results identical for spatial filtering and frequency domain filtering.

## Example 4.15 (cont'd)

Procedure to generate  $H(u,v)$ :

1. Multiply  $h_p(x, y)$  by  $(-1)^{x+y}$  to center the frequency domain filter
2. Compute the forward DFT of the result in Step 1
3. Set the real part of the resulting DFT to 0 ( $H$  has to be pure imaginary because  $h_p$  is real and odd)
4. Multiply the result by  $(-1)^{u+v}$ , which is implicit when  $h(x, y)$  was moved to the center of  $h_p(x, y)$ .

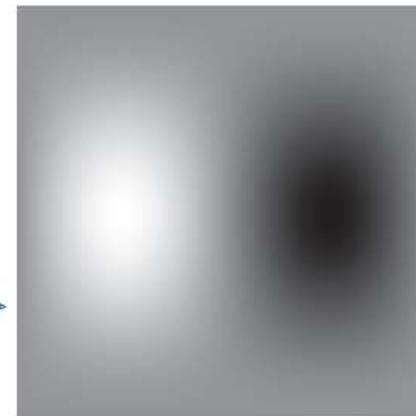
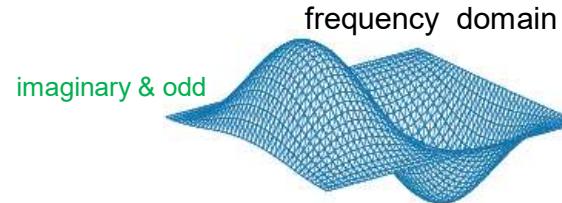
## Example 4.15 (cont'd)

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.

Sobel kernel in spatial domain  
**real & odd**

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



602x602

identical

## Image Smoothing Using Frequency-Domain Filters: ILPF

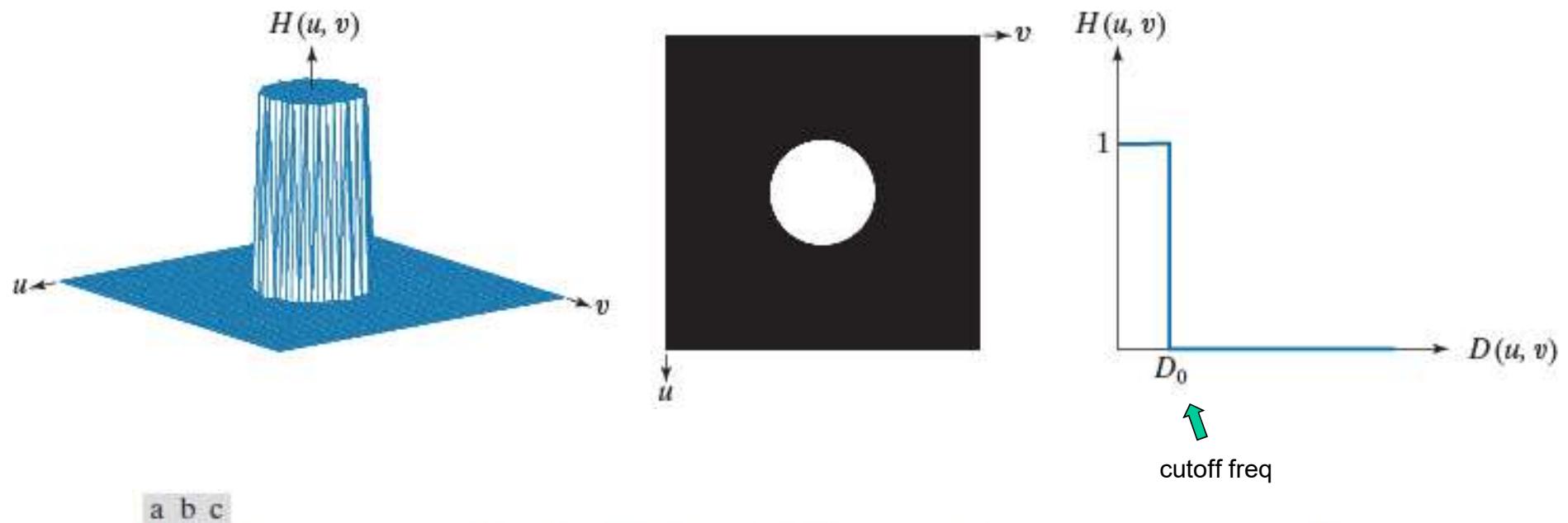
Ideal Lowpass Filters (ILPF)

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

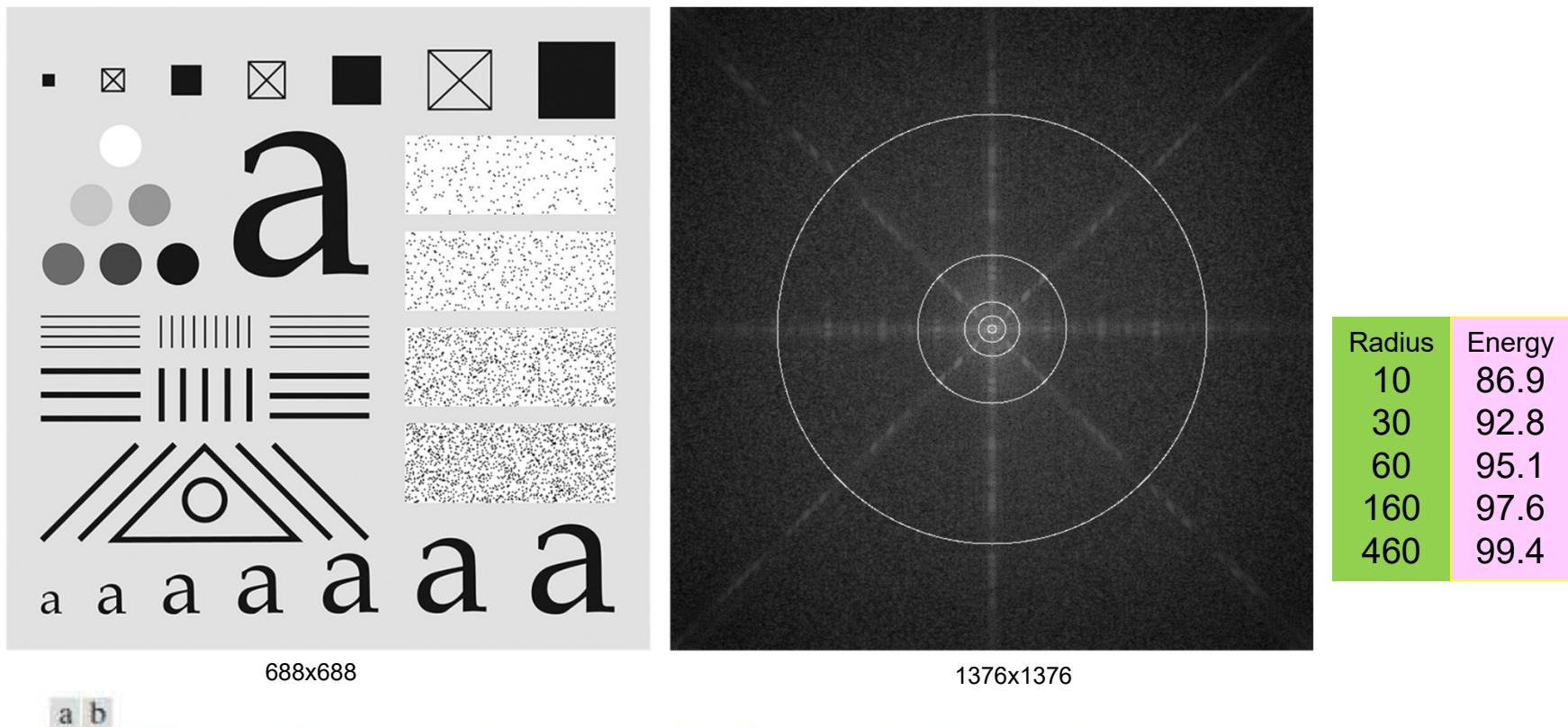
$D_0$  is a positive constant and  $D(u, v)$  is the distance between a point  $(u, v)$  in the frequency domain and the center of the frequency rectangle

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

## Image Smoothing Using Frequency-Domain Filters: ILPF



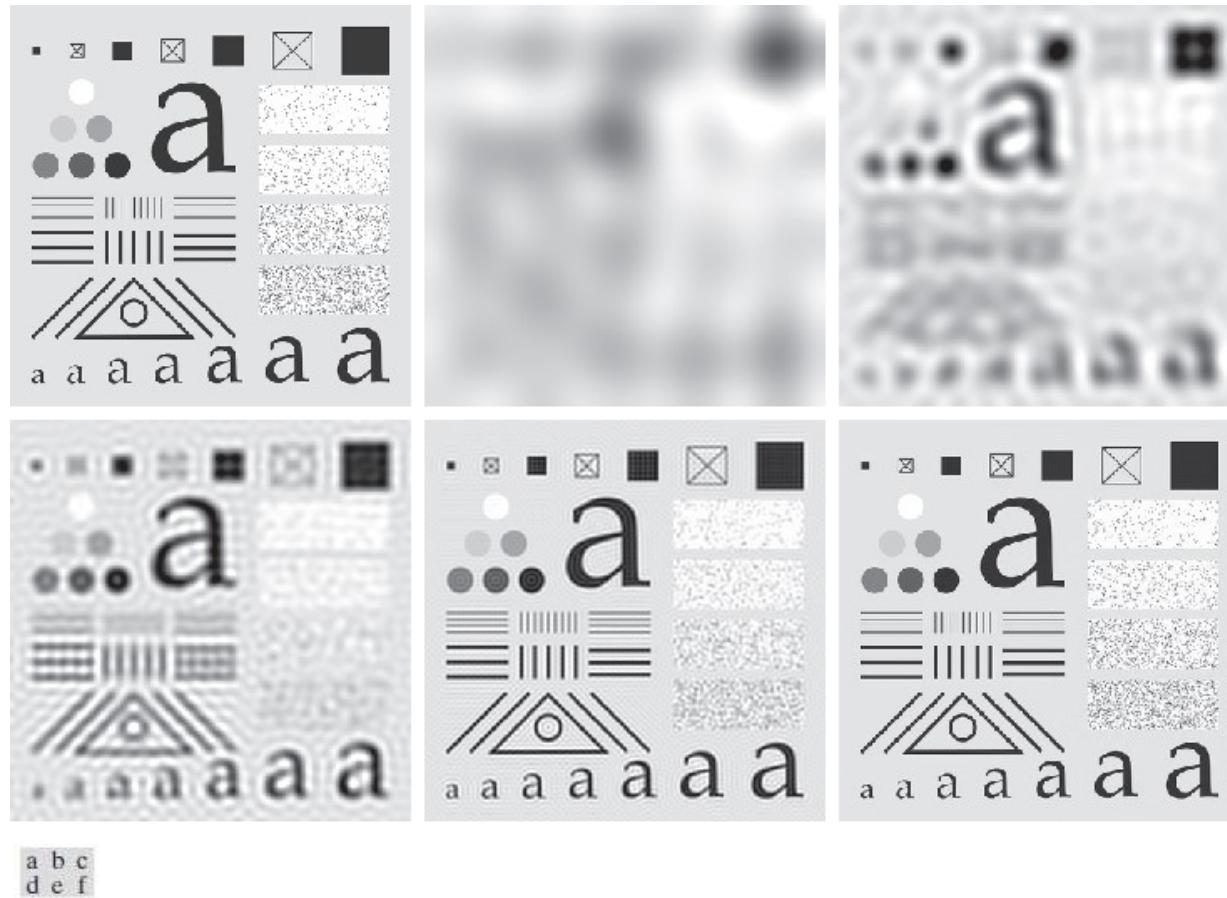
## ILPF Filtering Example



a b

**FIGURE 4.40** (a) Test pattern of size  $688 \times 688$  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.

## ILPF Filtering Example



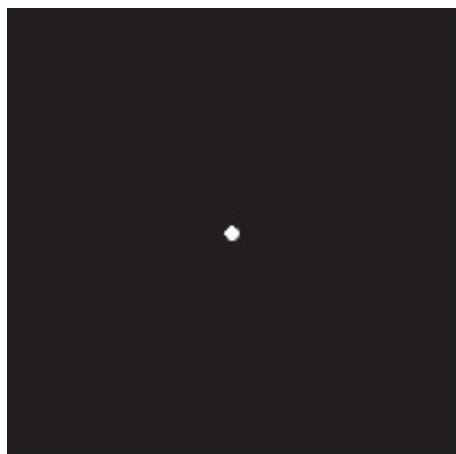
**FIGURE 4.41** (a) Original image of size  $688 \times 688$  pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively.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

## Blurring and Ringing Properties of ILPF

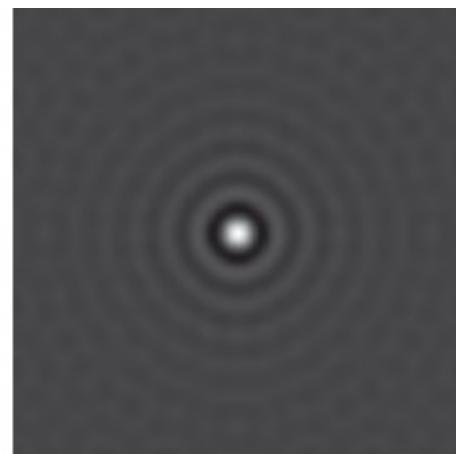
a b c

**FIGURE 4.42**

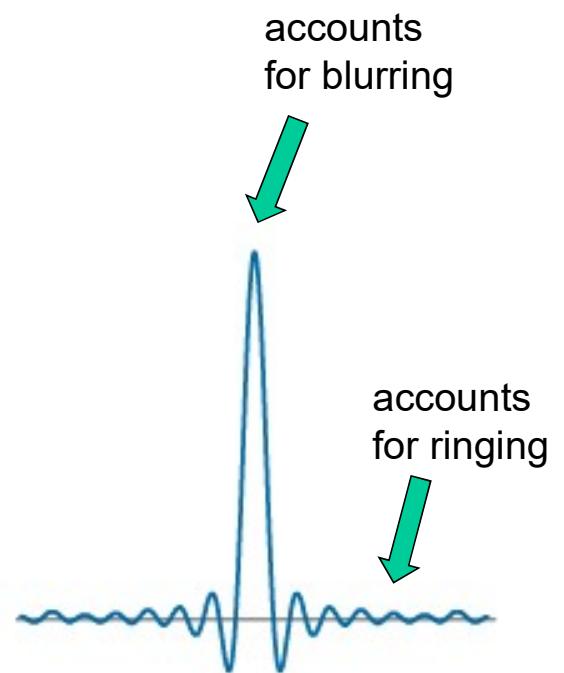
- (a) Frequency domain ILPF transfer function.
- (b) Corresponding spatial domain kernel function.
- (c) Intensity profile of a horizontal line through the center of (b).



frequency domain ILPF



spatial domain ILPF



## Image Smoothing Using Frequency-Domain Filters: GLPF

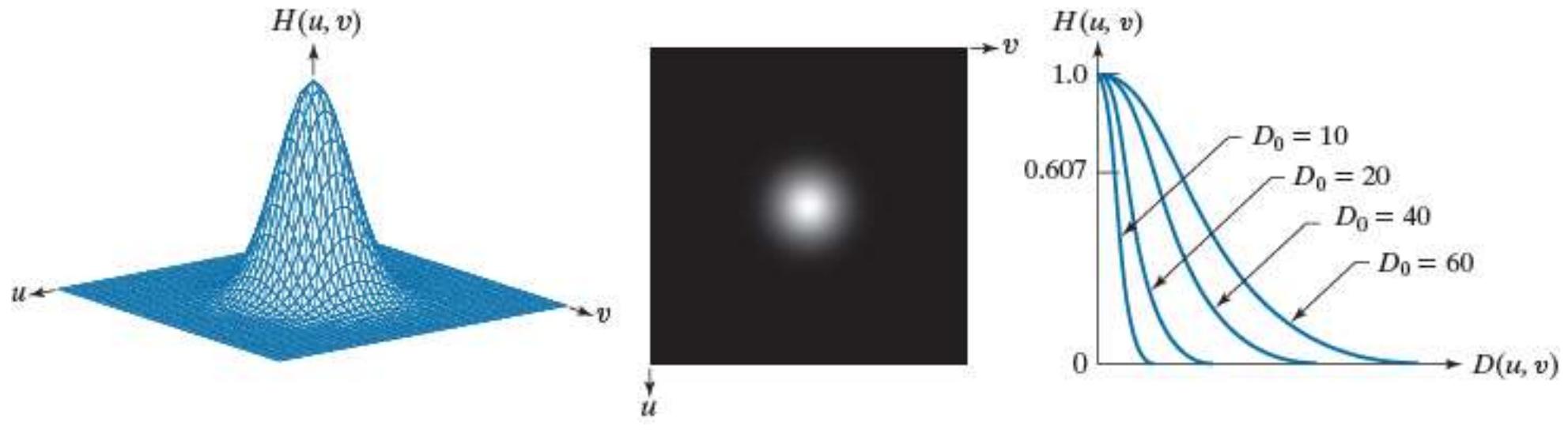
Gaussian Lowpass Filters (GLPF) in two dimensions is given

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

By letting  $\sigma = D_0$

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

## Image Smoothing Using Frequency-Domain Filters: GLPF

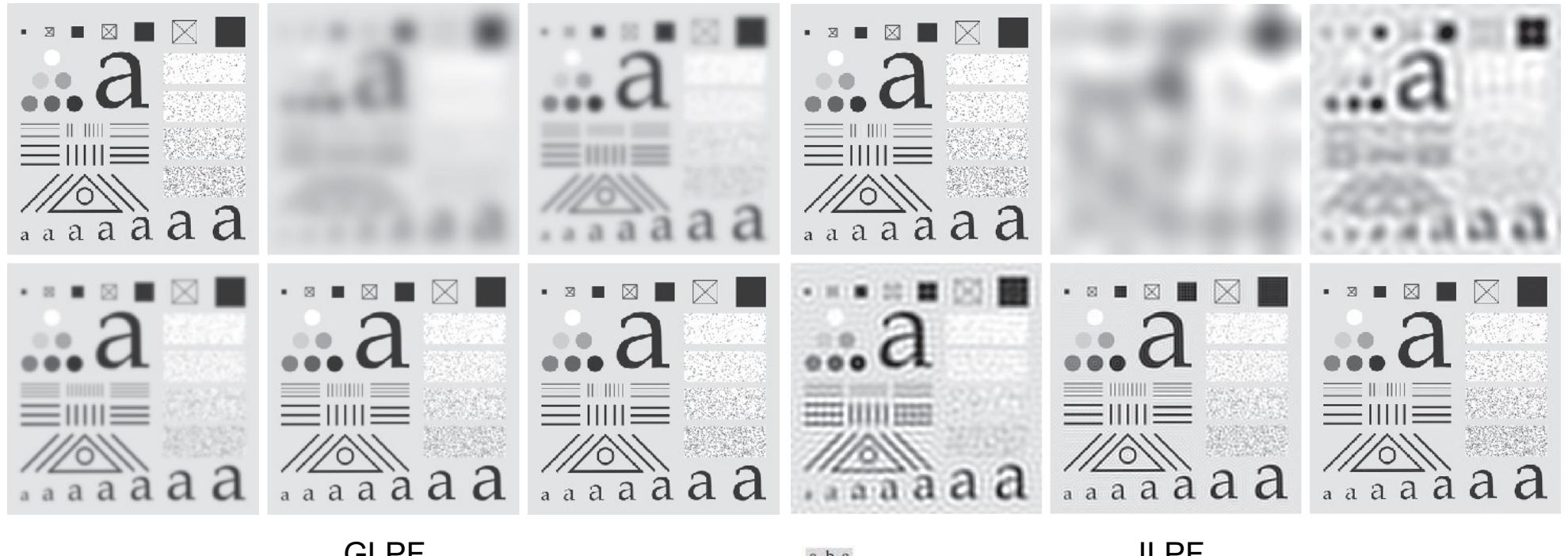


a b c

**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  $D_0$ .

$h(x,y)$  is also a Gaussian function  $\Rightarrow$  no ringing

## GLPF vs. ILPF



a b c  
d e f

**FIGURE 4.44** (a) Original image of size  $688 \times 688$  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.

Less smoothing, no ringing

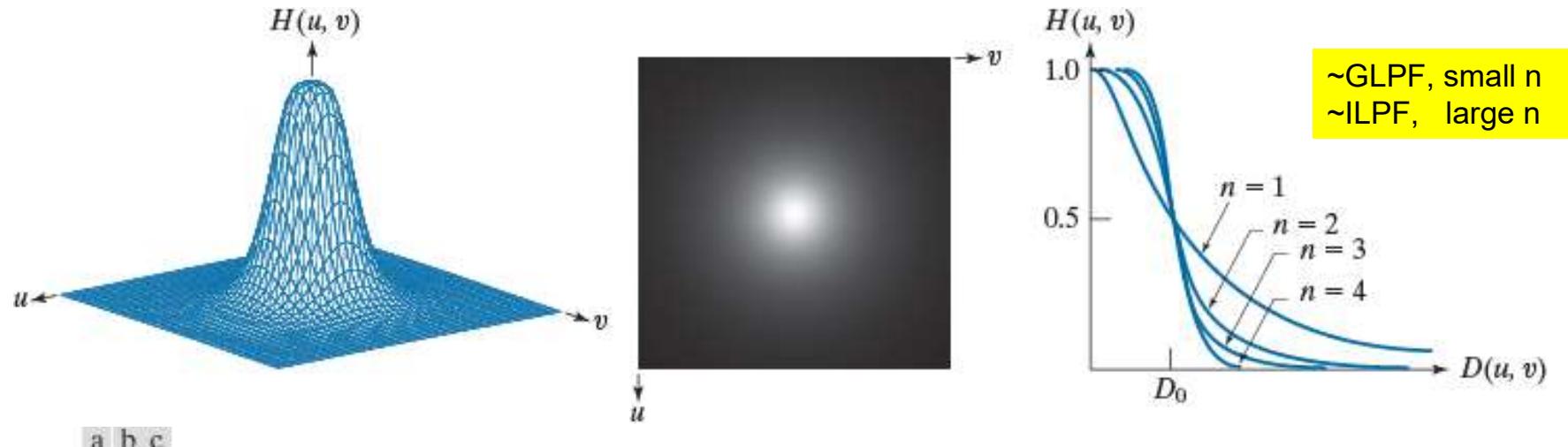
a b c  
d e f

**FIGURE 4.41** (a) Original image of size  $688 \times 688$  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).

## Image Smoothing Using Frequency-Domain Filters: BLPF

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

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



**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.

## Image Smoothing: BLPF vs. ILPF



a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

a b c

d e f

The extent of blurring is between  
GLPF and ILPF, see Fig. 4.46 (b)

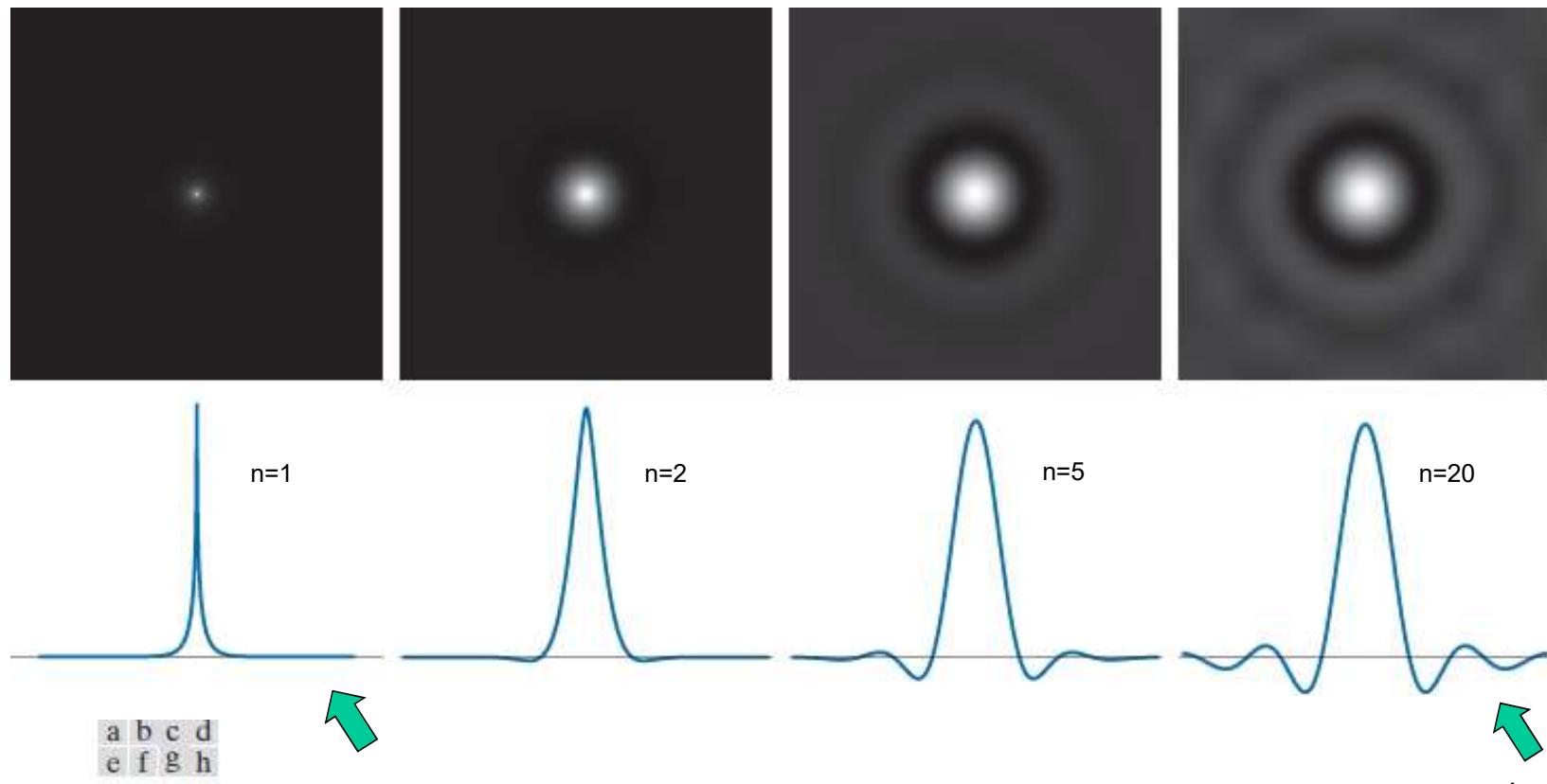
**FIGURE 4.46** (a) Original image of size  $688 \times 688$  pixels. (b)–(f) Results of filtering using BLPFs with cutoff frequencies set at the radii shown in Fig. 4.40 and  $n = 2.25$ . Compare with Figs. 4.41 and 4.44. We used mirror padding to avoid the black borders characteristic of zero padding.

a b c  
d e f

ILPF

**FIGURE 4.41** (a) Original image of size  $688 \times 688$  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).

## Spatial Representation of BLPF



**FIGURE 4.47** (a)–(d) Spatial representations (i.e., spatial kernels) corresponding to BLPF transfer functions of  $1000 \times 1000$  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.

## Using LPF to Repair Broken Text

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 GLPF, 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.



## Using LPF to Produce Soft-Looking Image



**FIGURE 4.49** (a) Original  $785 \times 732$  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).

## Using LPF to Soften Scanlines



a b c

**FIGURE 4.50** (a)  $808 \times 754$  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.)

## Image Sharpening Using Frequency-Domain Filters

A highpass filter is obtained from a given lowpass filter using

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

A 2-D ideal highpass filter (IHPL) is defined as

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

Attenuate low-frequency components without disturbing high-frequencies.

## 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

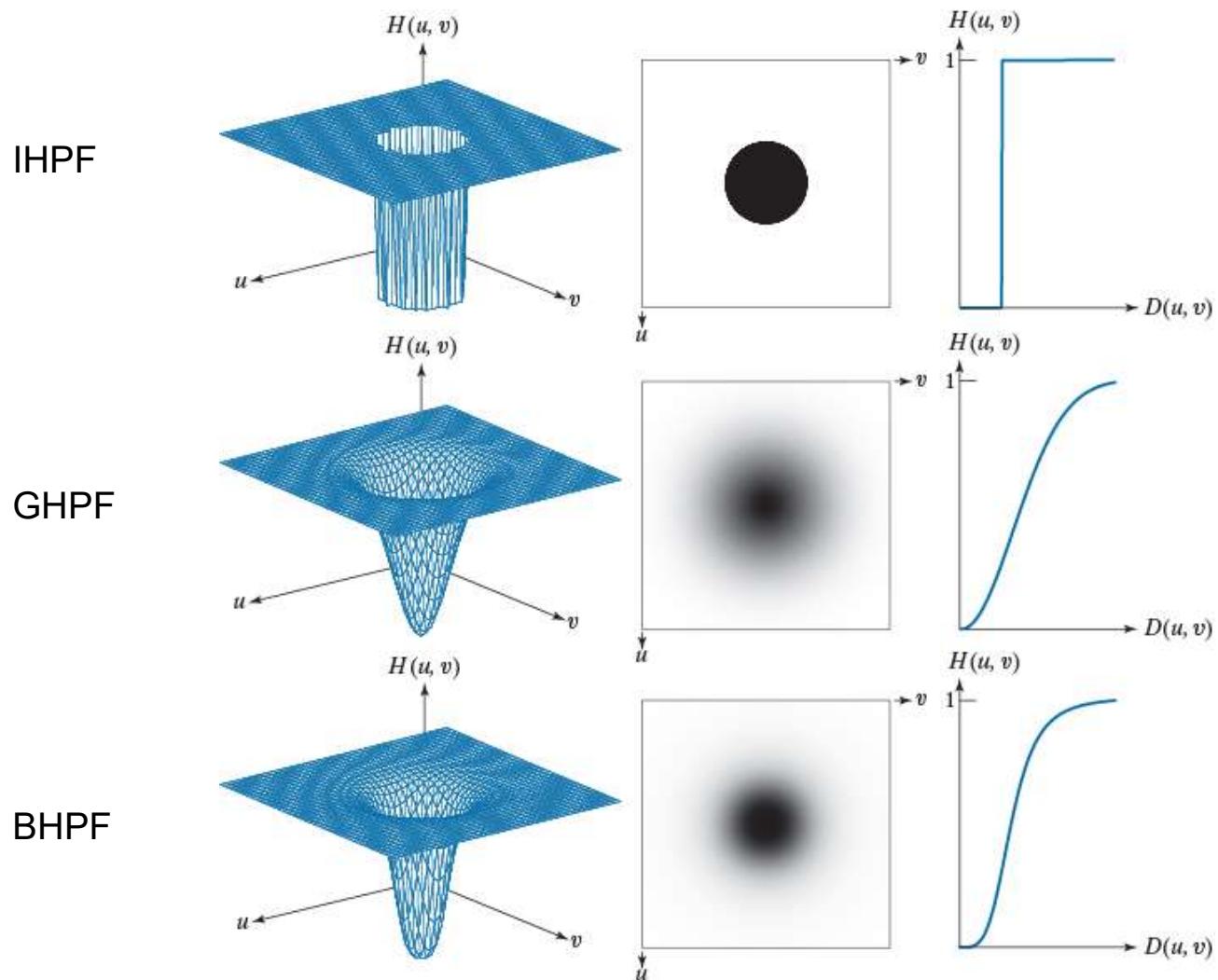
$$H(u, v) = 1 - e^{-D^2(u, v)/2D_0^2}$$

## IHPF, GHPF, and BHPF

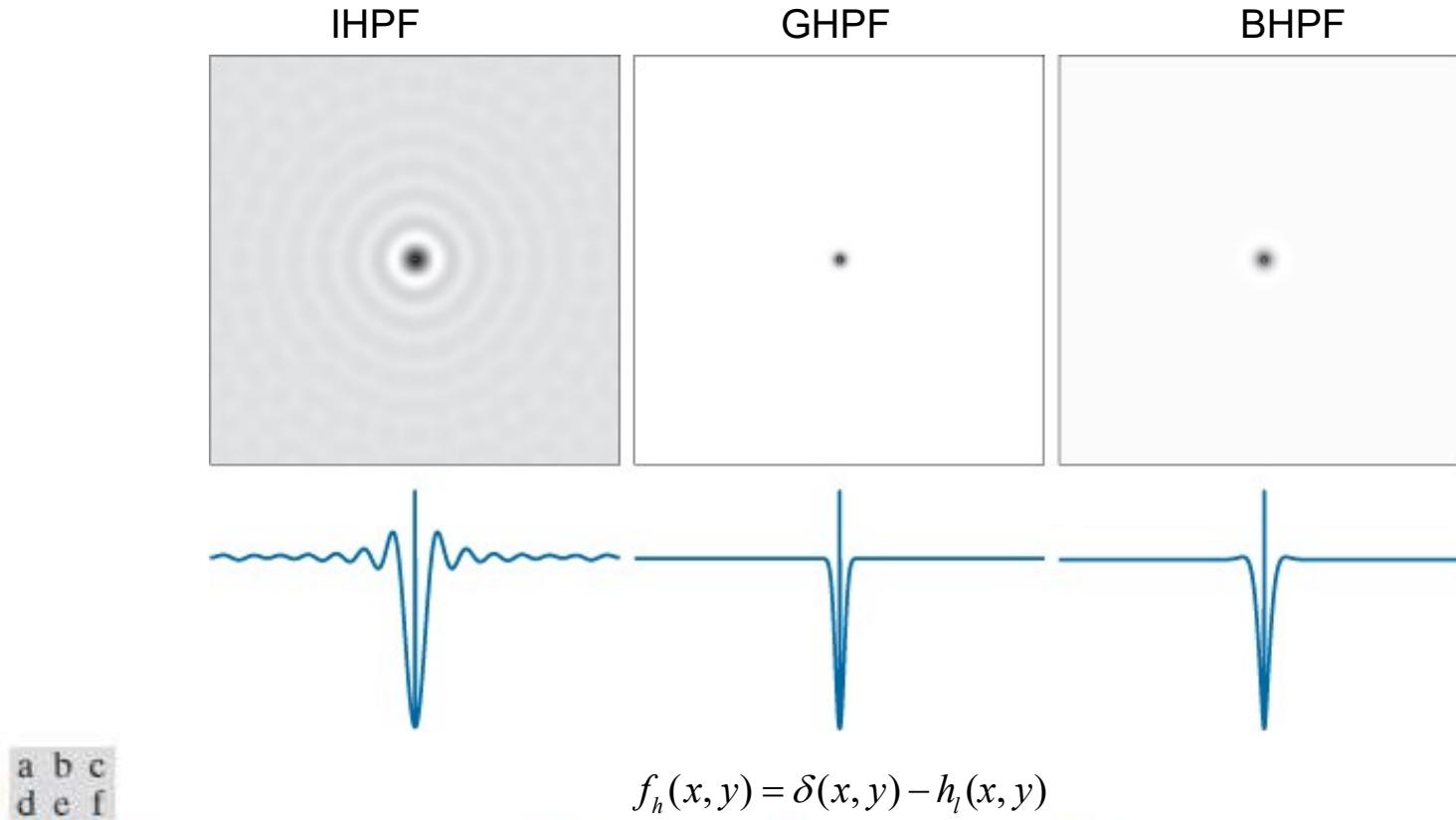
a b c  
d e f  
g h i

**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.)

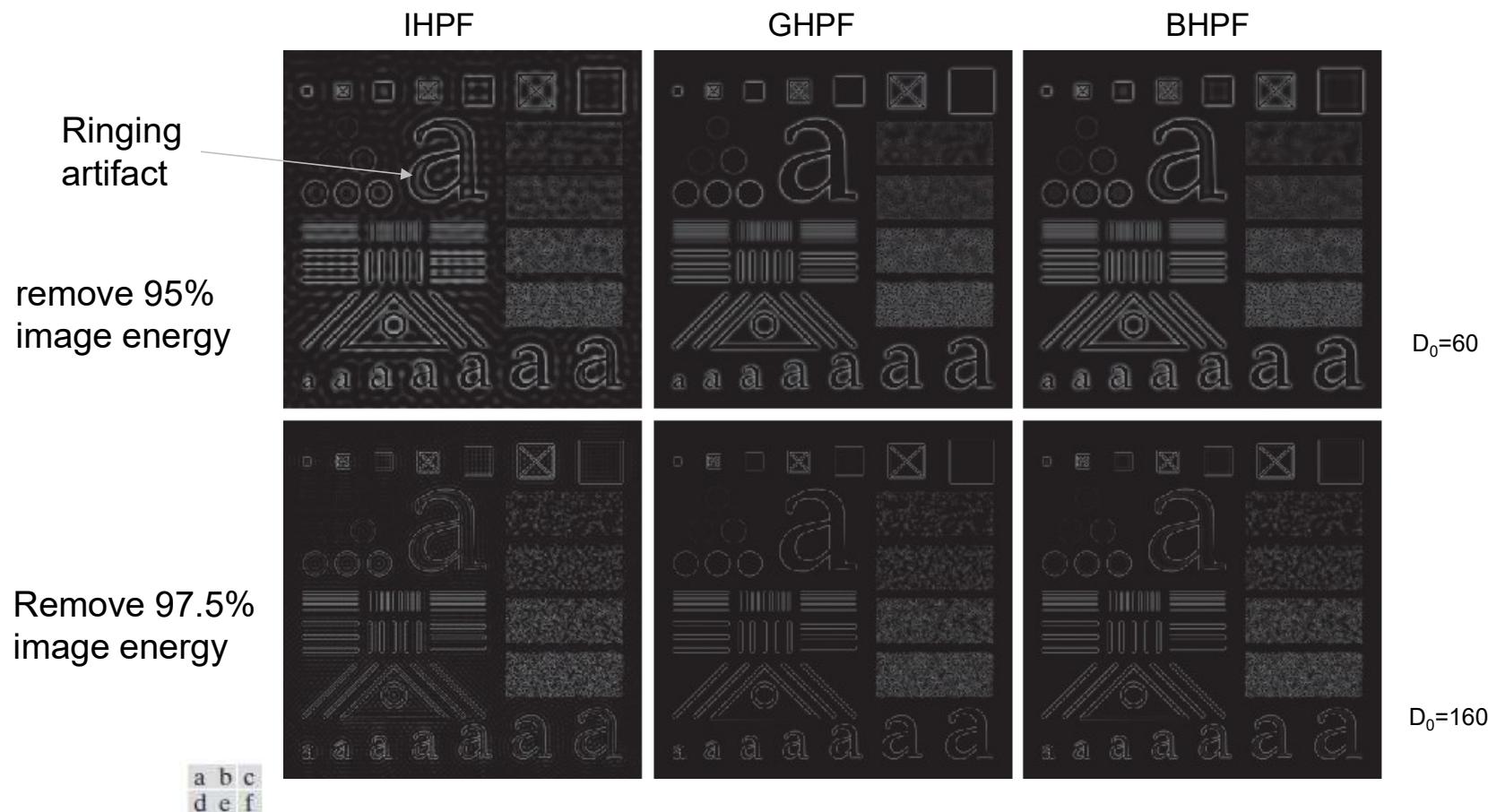


## Highpass Spatial Kernels



**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 Filtering by IHPF, GHPF, and BHPF



**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  $D_0 = 160$ .

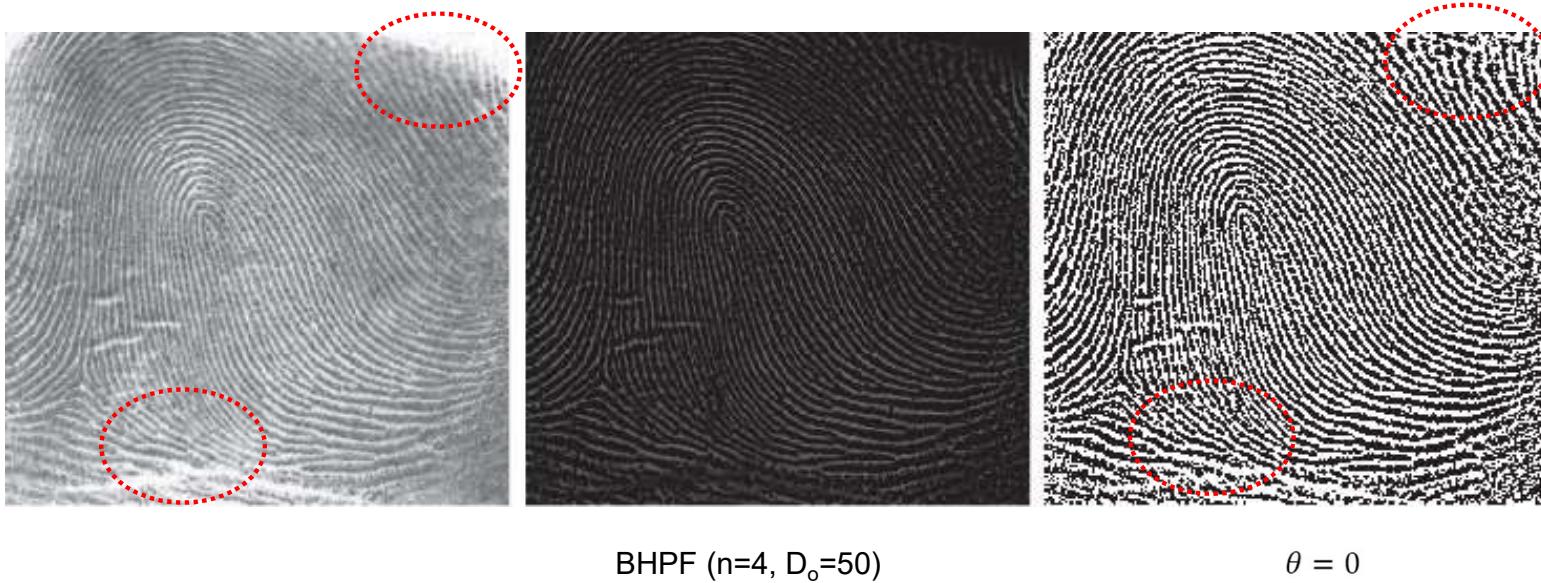


a b c

**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.

# Using Highpass Filtering and Thresholding for Image Enhancement

- Enhancement of print ridges and reduction of smudges



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.)

## The Laplacian in the Frequency Domain

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

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

The Laplacian image

$$\nabla^2 f(x, y) = \mathfrak{F}^{-1} \{ H(u, v) F(u, v) \} \quad (4-125)$$

Enhanced image is obtained by

$$g(x, y) = f(x, y) + c \nabla^2 f(x, y) \quad (4-126)$$

$c = -1$  because  $H(u, v)$  is negative

## The Laplacian in the Frequency Domain

The enhanced image can be obtained in the frequency domain by

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

Handle the scale difference:

- Normalize  $f(x, y)$  to  $[0, 1]$  before taking DFT
- Divide  $\nabla^2 f(x, y)$  by its maximum value to bring it to  $[-1, 1]$

## Image Sharpening Using The 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. 3.46(d).  
(Original image  
courtesy of  
NASA.)



Better than Fig. 3.46(d)  
Because the spatial  
Laplacian kernel  
encompasses a very  
small neighborhood,  
while the formulation in  
Eqs. (4-125) and (4-126)  
encompasses the entire  
image.

Fig. 3.46(d)

## Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Fitering

$$g_{mask}(x, y) = f(x, y) - f_{LP}(x, y) \quad (\text{Equivalent to Eq. (3-55)})$$

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

Unsharp masking and highboost filtering

$$g(x, y) = f(x, y) + k * g_{mask}(x, y) \quad \begin{cases} k=1 & \text{unsharp masking} \\ k>1 & \text{high-boost filtering} \end{cases}$$

$$\begin{aligned} g(x, y) &= \mathcal{J}^{-1}\left\{\left[1 + k * [1 - H_{LP}(u, v)]\right]F(u, v)\right\} \\ &= \mathcal{J}^{-1}\left\{\underbrace{[1 + k * H_{HP}(u, v)]}_{\text{High-frequency-emphasis filter transfer function}}F(u, v)\right\} \end{aligned}$$

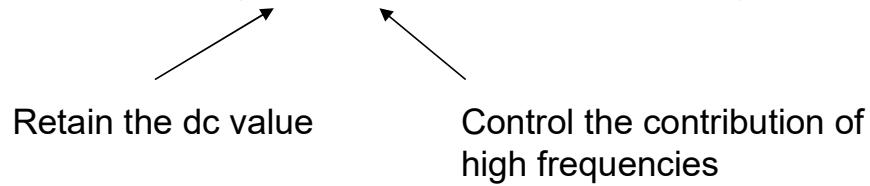
Unlike highpass filters,  
these filters retain the  
dc value of the image

High-frequency-emphasis filter transfer function

## Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Fitering

A slightly more general form:

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



# Image Enhancement by High-Frequency-Emphasis Filtering

a  
b  
c  
d

**FIGURE 4.57**

(a) A chest X-ray.  
(b) Result of filtering with a GHPF function.  
(c) Result of high-frequency-emphasis filtering using the same GHPF. (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.)



Use a high-frequency-emphasis Gaussian filter with  $K_1=0.5$ ,  $k_2=0.75$

Use GHPF to avoid ringing

- A 503x720 slightly blurred image
- Intensity biased toward dark
- To show how spatial filtering complements spectral filtering
- Use a GHPF with  $D_0=70$ ; guarantee no ringing

Note that the intensity levels of (c) are in a narrow range of the gray scale. So we apply histogram equalization to (c).

## Homomorphic Filtering

Illumination-reflection model:

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

$$\Im[f(x, y)] \neq \Im[i(x, y)]\Im[r(x, y)]$$

Alternative formulation:

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

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

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

## Homomorphic Filtering

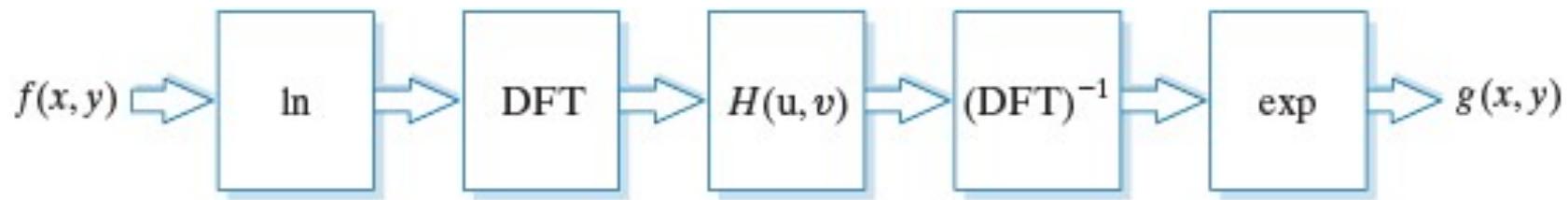
$$\begin{aligned} S(u, v) &= H(u, v)Z(u, v) \\ &= H(u, v)F_i(u, v) + H(u, v)F_r(u, v) \end{aligned}$$

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

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

Simultaneous intensity range  
compression and contrast enhancement

## Homomorphic Filtering



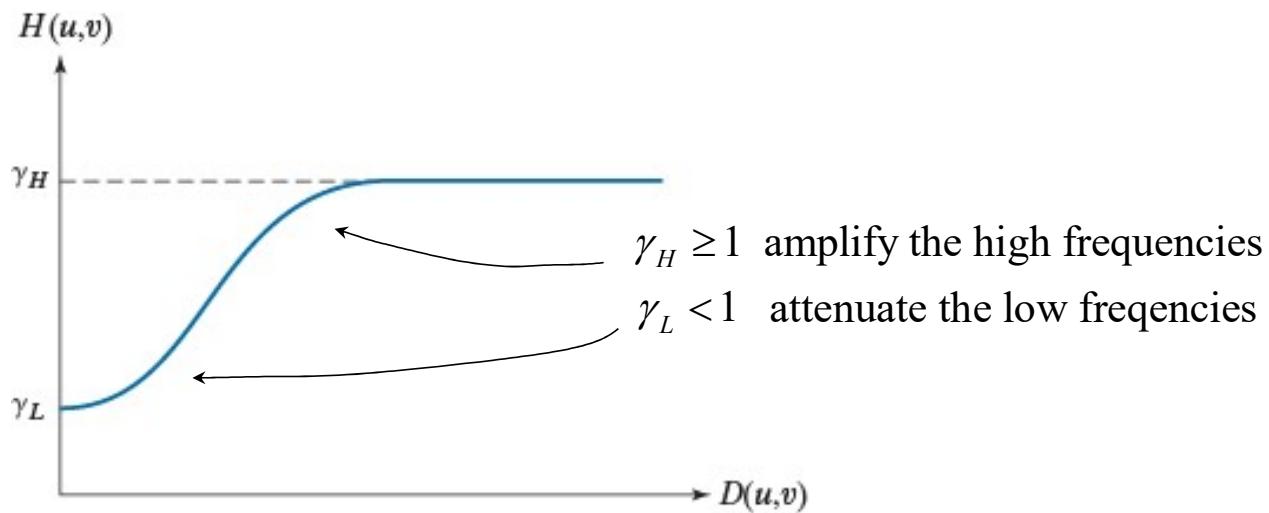
Illumination has slow spatial variation  $\rightarrow$  low frequency components  
Reflectance tends to vary abruptly  $\rightarrow$  high frequency components

The filtering approach allows the separation of illumination from reflectance components of a homomorphic system, on which the homomorphic filter can operate on these components differently.

We may control the processing of these two components through the specification of  $H(u, v)$ .

## Homomorphic Filtering

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

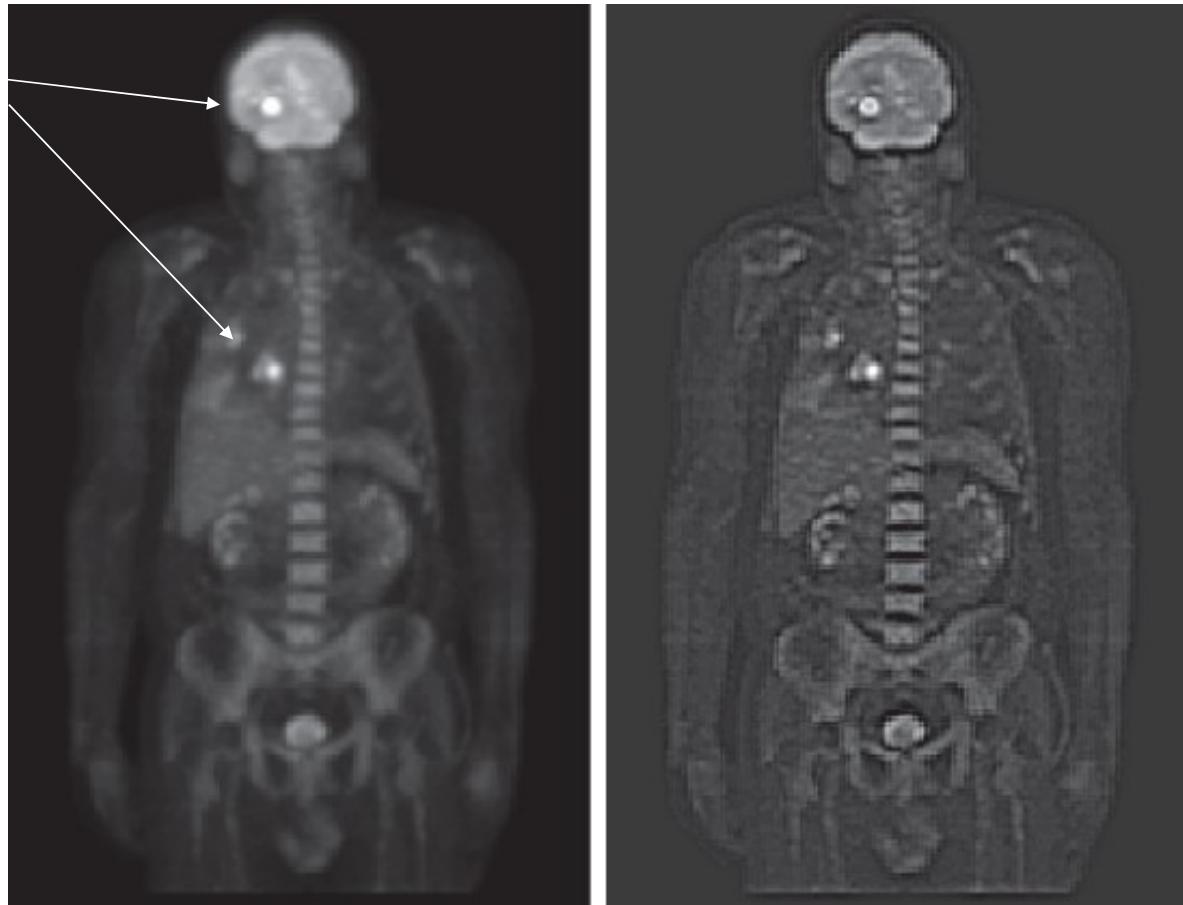


The shape of  $H(u,v)$  can be approximated by a GHPF,

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

## Homomorphic Filtering

- One tumor in the brain and one in the lung
- Blur image
- Low-intensity features obscured by high intensity of hot spots



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.)

$$\gamma_L = 0.25$$

$$\gamma_H = 2$$

$$c = 1$$

$$D_0 = 80$$

Reflectance components (edges) are sharpened considerably

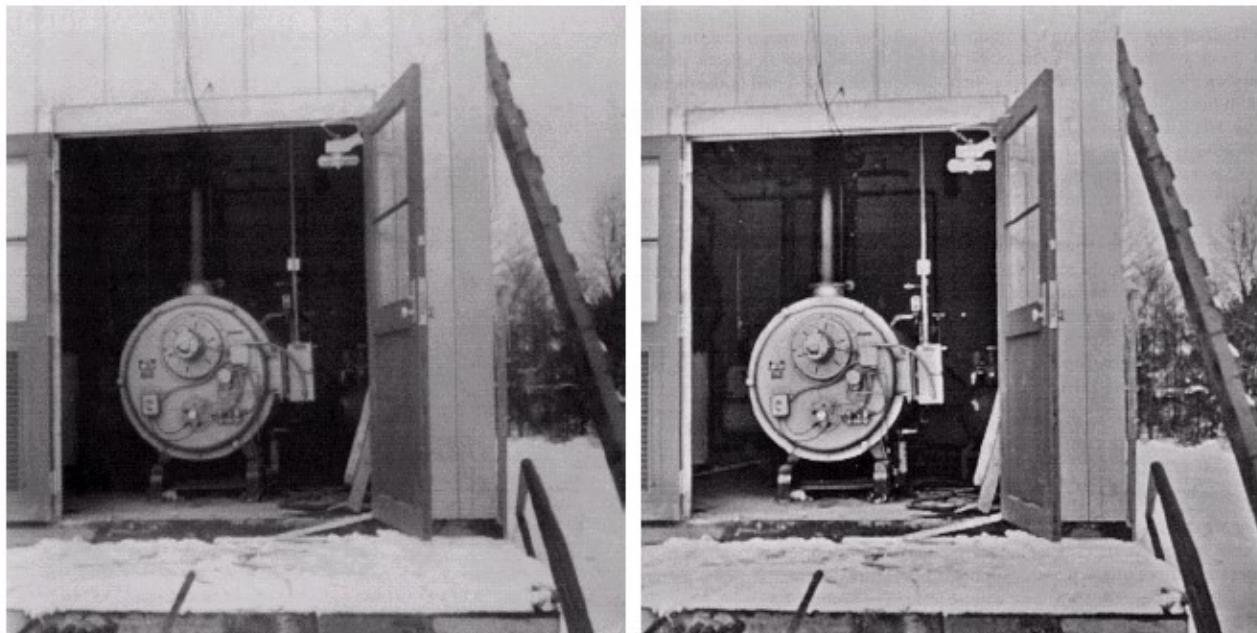
## Homomorphic Filtering

a b

**FIGURE**

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

---



## Selective Filtering

### **Non-Selective Filters:**

operate over the entire frequency rectangle

### **Selective Filters**

operate over some part, not entire frequency rectangle

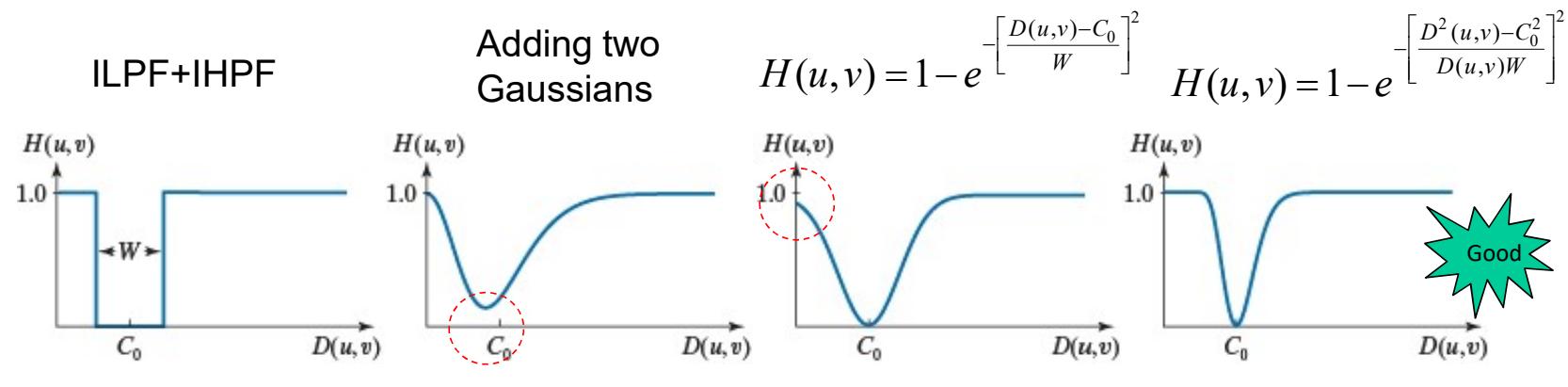
- **bandreject or bandpass:** process specific bands
- **notch filters:** process small regions of the frequency rectangle

# Bandrejected Filters

Requirements:

1.  $H(u,v)$  must be in the range  $[0,1]$
2.  $H(u,v)$  must be zero at  $C_0$  from the origin
3. Must be able to specify  $W$

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



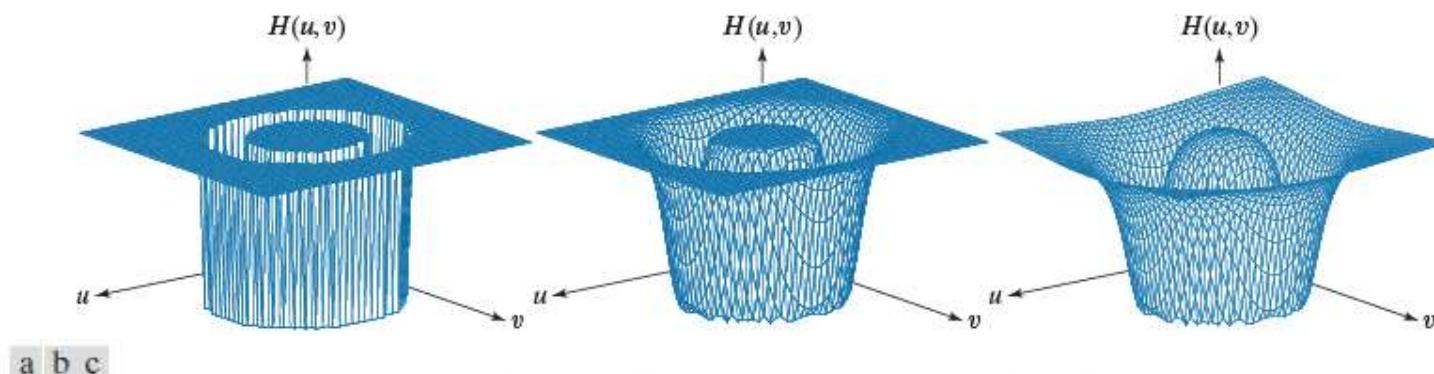
**FIGURE 4.61** Radial cross sections. (a) Ideal bandreject filter transfer function. (b) Bandreject transfer function formed by the sum of Gaussian lowpass and highpass filter functions. (The minimum is not 0 and does not align with  $C_0$ .) (c) Radial plot of Eq. (4-149). (The minimum is 0 and is 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.

# Transfer Functions of IBRF, GBRF, and BBRF

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



**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  $512 \times 512$  elements, with  $C_0 = 128$  and  $W = 60$ .

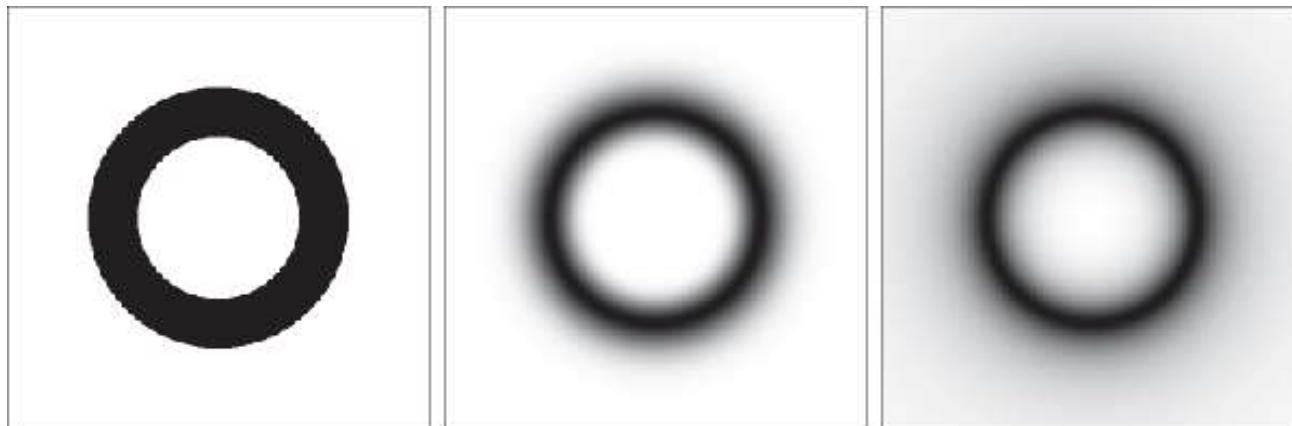
## IBRF, GBRF, and BBRF

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

a b c

**FIGURE 4.63**

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



## Notch Filters

- Most useful selective filters
- A notch filter is a BRF with a narrow stop band
- To be a zero-phase-shift filter, a notch filter centered at  $(u_0, v_0)$  must have a corresponding notch at  $(-u_0, -v_0)$ .
- A notch reject filter is constructed as the 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.

## Notch Filters

A Butterworth notch reject filter of order n

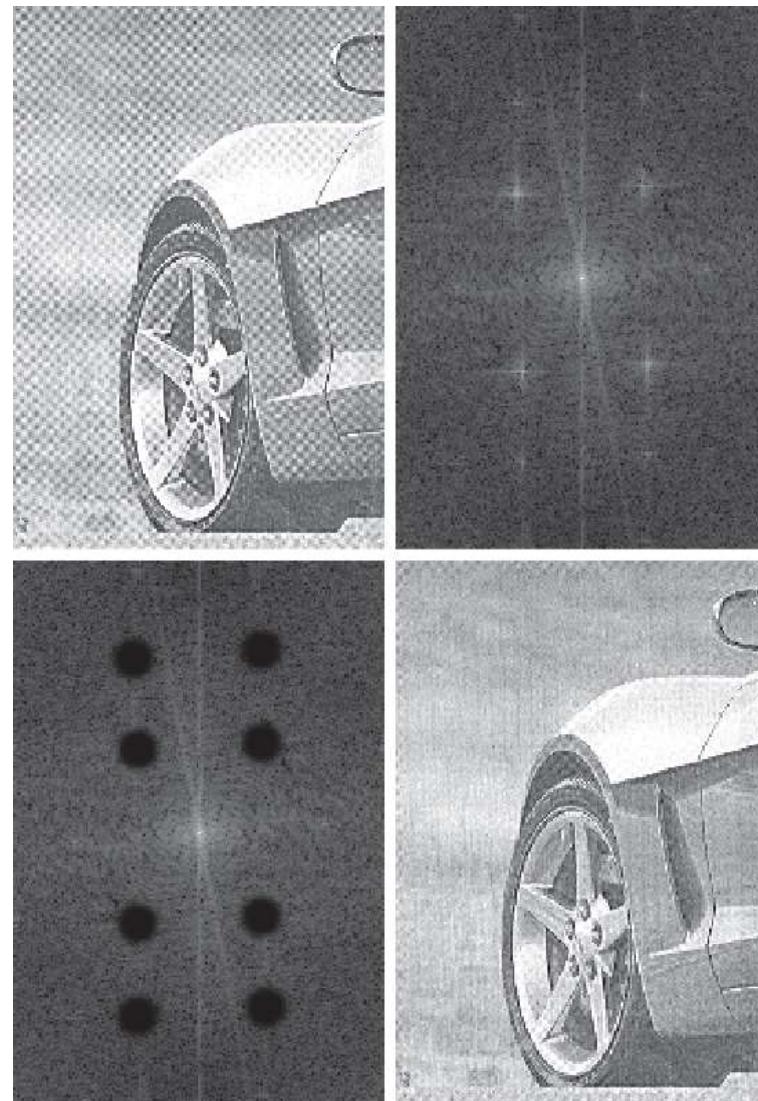
$$H_{NR}(u, v) = \prod_{k=1}^3 \left[ \frac{1}{1 + [D_{0k} / D_k(u, v)]^{2n}} \right] \left[ \frac{1}{1 + [D_{0k} / D_{-k}(u, v)]^{2n}} \right]$$

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

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

$D_{0k}$ : a constant

## Example 4.24: Notch Filter



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.

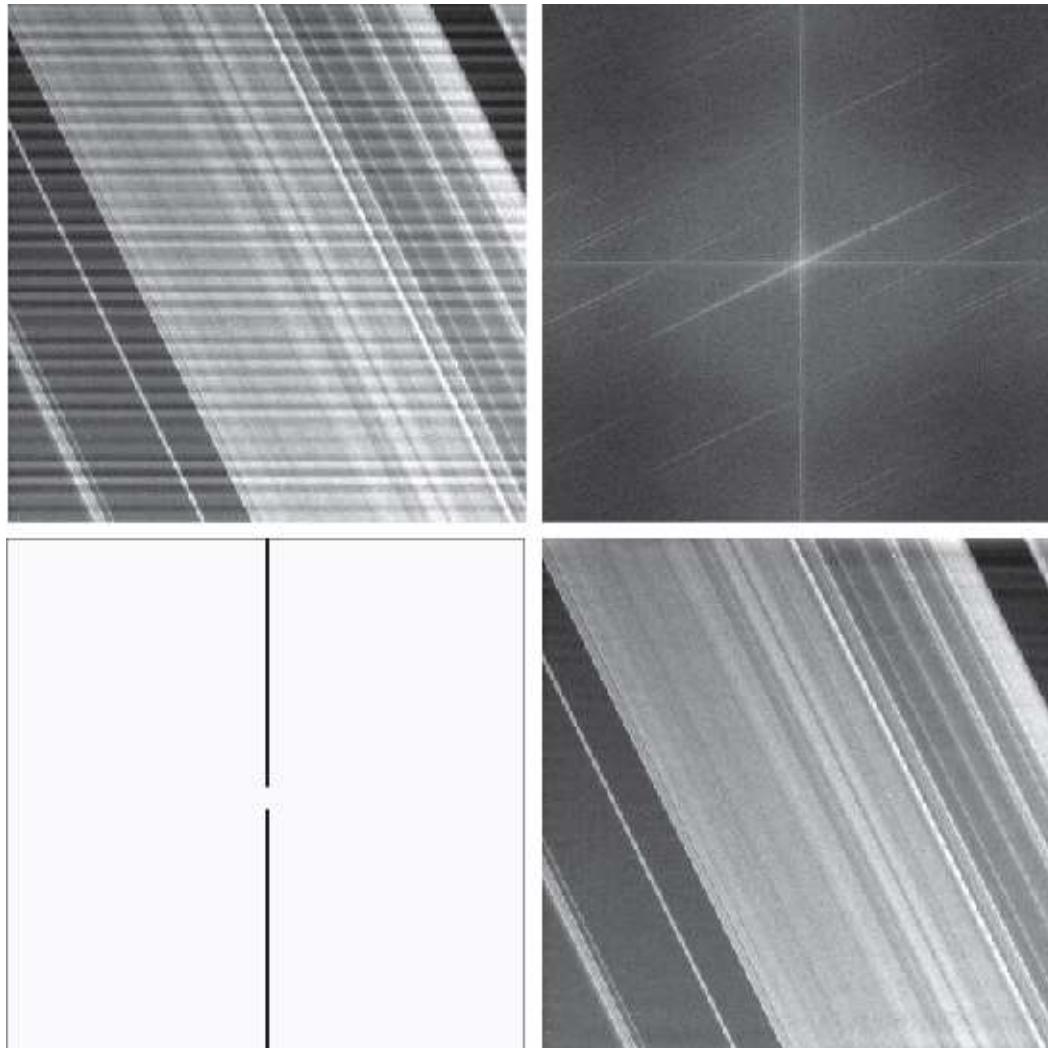
A Butterworth notch reject filter  $D_0=9$  and  $n=4$  for all notch pairs

## Example 4.25: Notch Filter

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.)

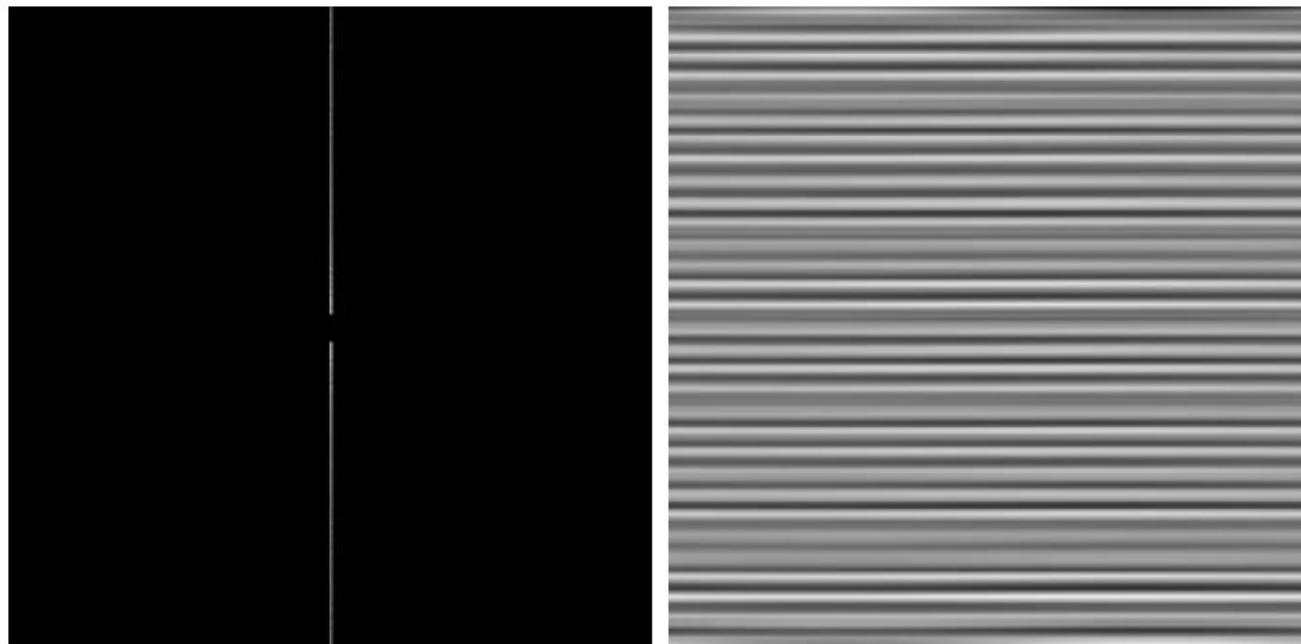


## Example 4.25: Notch Filter

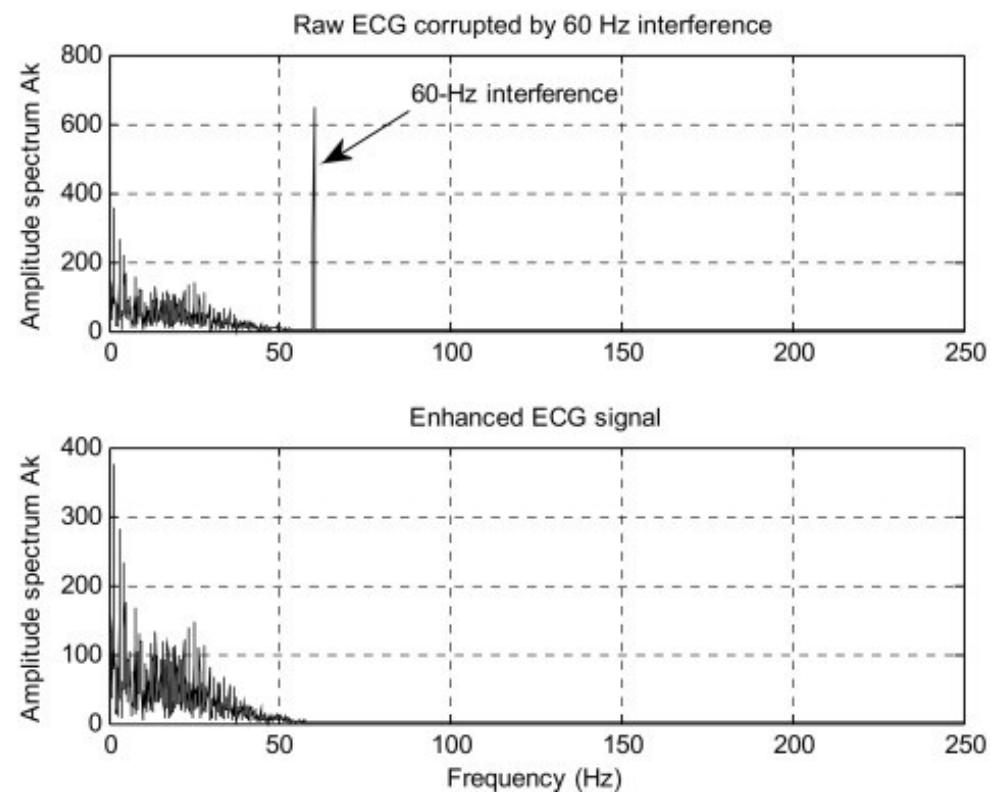
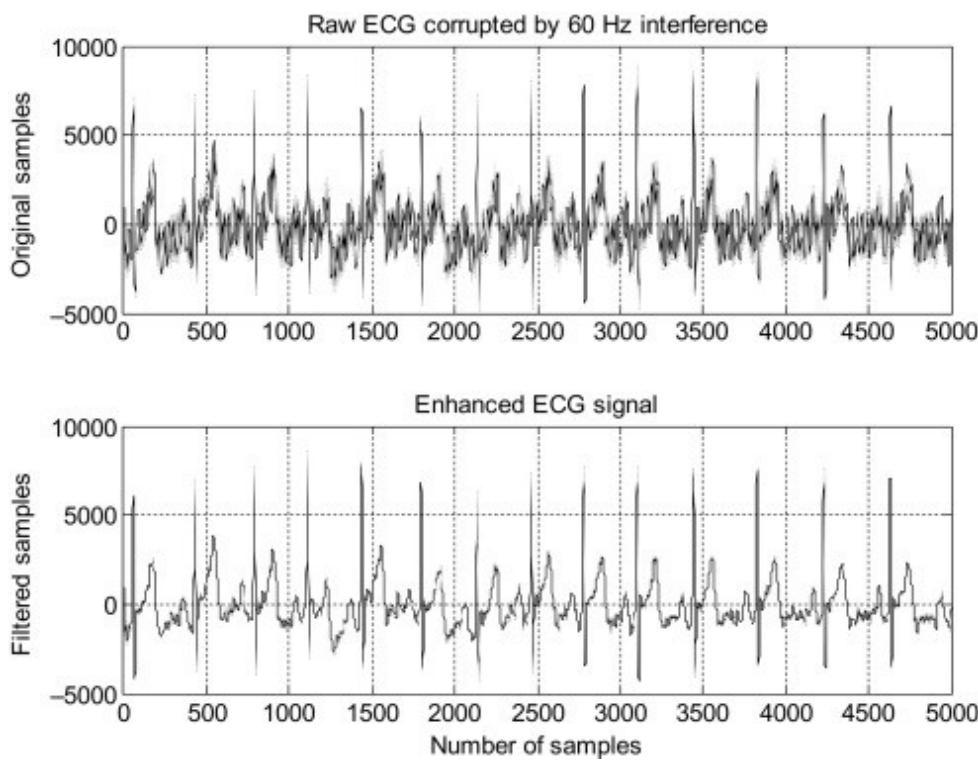
a b

**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).



# NRF



**FIGURE 4.67**  
Computational advantage of the FFT over a direct implementation of the 1-D DFT. The number of samples is  $M = 2^p$ . The computational advantage increases rapidly as a function of  $p$ .

