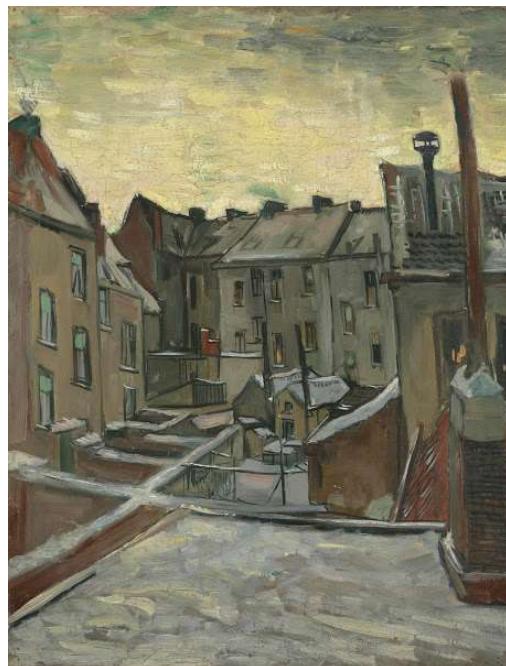


# **Introduction to Image Processing**



**Prof. Alexandre Zaghetto**  
<http://alexandre.zaghetto.com>  
zaghetto@unb.br

University of Brasília  
Department of Computer Science  
LISA: Laboratory of Imagens, Signals and Acoustics

---

# **Topic 04**

## **Filtering in the Frequency Domain**

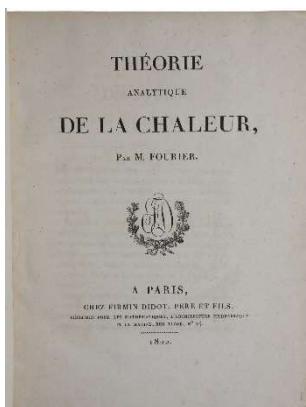
---

## 1. Background



- **Jean Baptiste Joseph Fourier** was a French mathematician born in 1768.
- The contribution for which he is most remembered was outlined in a memoir in 1807 and published in 1822 in his book, *La Théorie Analytique de la Chaleur* (**The Analytic Theory of Heat**).
- Basically, Fourier's contribution in this field states that **any periodic function** can be expressed as the **sum of sines and/or cosines** of different frequencies, each multiplied by a different coefficient (we now call this sum a **Fourier series**).
- Even functions that are **not periodic** (but whose area under the curve is finite) can be expressed as the **integral of sines and/or cosines** multiplied by a weighing function. The formulation in this case is the **Fourier transform**.

## 1. Background



- Fourier presented a paper in 1807 to the *Institut de France* on the use of sinusoids to represent temperature distributions.
- Among the reviewers were two of history's most famous mathematicians, Joseph Louis **Lagrange** (1736-1813), and Pierre Simon de **Laplace** (1749-1827).
- **Laplace** and the other reviewers **voted to publish the paper**.
- **Lagrange** adamantly **protested**.
- The *Institut de France* bowed to the prestige of Lagrange, and **rejected Fourier's work**.
- Only after Lagrange died that the paper was finally published, **some 15 years later**.

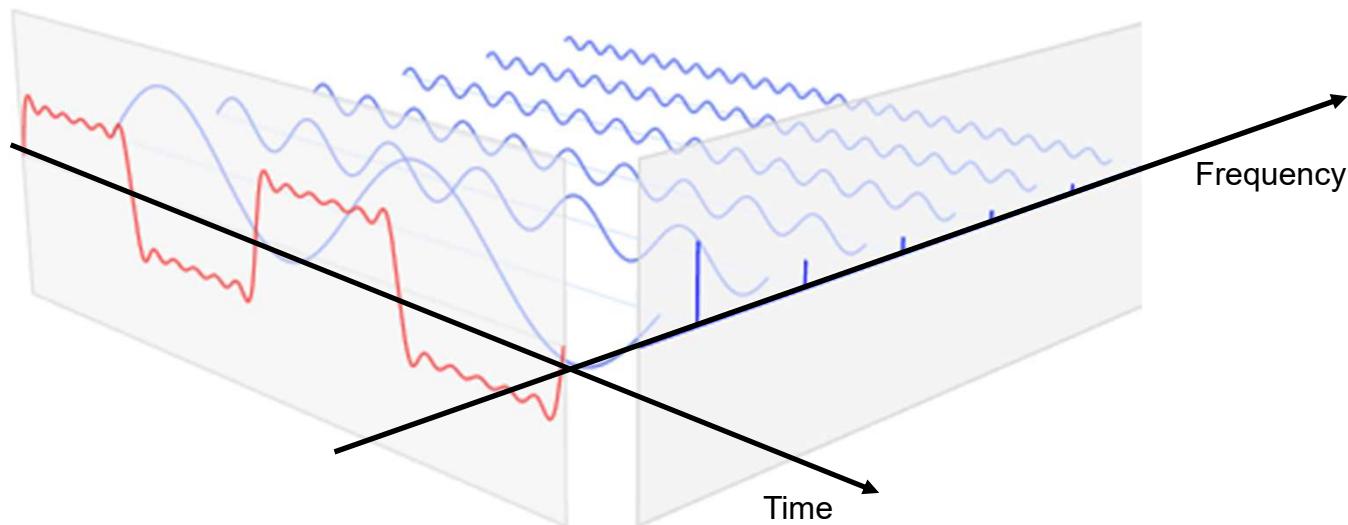
## 2. Preliminary Concepts

- A function, expressed in either a Fourier series or transform, **can be reconstructed completely** via with no loss of information.



## 2. Preliminary Concepts

- The advent of digital **computers** and the “discovery” of a **fast Fourier transform** (FFT) algorithm in the early 1960s (more about this later) revolutionized the field of signal processing.



## 2. Preliminary Concepts

- Complex numbers

$$C = R + jI$$

- Conjugate

$$C^* = R - jI$$

- Polar coordinates

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

$$|C| = \sqrt{R^2 + I^2}$$

$$\theta = \arctan(I/R)$$

- Exponential form

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

## 2. Preliminary Concepts

- The preceding equations are applicable also to complex functions.

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

$$|F(u)| = \sqrt{R(u)^2 + I(u)^2}$$

$$\theta(u) = \arctan[I(u)/R(u)]$$

- 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**.

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

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

## 2. Preliminary Concepts

- Fourier series (Trigonometric):

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ a_n \cos\left(\frac{2\pi n}{T}t\right) + b_n \sin\left(\frac{2\pi n}{T}t\right) \right]$$

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(nw_0 t) + b_n \sin(nw_0 t)] \quad w_0 = \frac{2\pi}{T}$$

$$a_0 = \frac{2}{T} \int_0^T f(t) dt$$

$$a_n = \frac{2}{T} \int_0^T f(t) \cos(nw_0 t) dt$$

$$b_n = \frac{2}{T} \int_0^T f(t) \sin(nw_0 t) dt$$

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(nw_0 t) + b_n \sin(nw_0 t)]$$

$$\begin{aligned} e^{jn w_0 t} &= \cos(n w_0 t) + j \sin(n w_0 t) & \cos(n w_0 t) &= \frac{1}{2} (e^{j n w_0 t} + e^{-j n w_0 t}) \\ e^{-j n w_0 t} &= \cos(n w_0 t) - j \sin(n w_0 t) & \xrightarrow{\hspace{1cm}} & \sin(n w_0 t) = \frac{1}{2j} (e^{j n w_0 t} - e^{-j n w_0 t}) \end{aligned}$$

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ \frac{a_n}{2} (e^{j n w_0 t} + e^{-j n w_0 t}) + \frac{b_n}{2j} (e^{j n w_0 t} - e^{-j n w_0 t}) \right]$$

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ \frac{a_n}{2} (e^{jn\omega_0 t} + e^{-jn\omega_0 t}) + \frac{b_n}{2j} (e^{jn\omega_0 t} - e^{-jn\omega_0 t}) \right]$$

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ \left( \frac{a_n - jb_n}{2} \right) e^{jn\omega_0 t} + \left( \frac{a_n + jb_n}{2} \right) e^{-jn\omega_0 t} \right]$$

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ \left( \frac{a_n - jb_n}{2} \right) e^{jn\omega_0 t} + \left( \frac{a_n + jb_n}{2} \right) e^{-jn\omega_0 t} \right]$$

- Taking

$$n \geq 1$$

$$c_0 = \frac{1}{2}a_0, \quad c_n = \frac{1}{2}(a_n - jb_n), \quad c_{-n} = \frac{1}{2}(a_n + jb_n)$$

- It follows that

$$f(t) = c_0 + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t} + \sum_{n=1}^{\infty} c_{-n} e^{-jn\omega_0 t}$$

---

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$f(t) = c_0 + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t} + \sum_{n=1}^{\infty} c_{-n} e^{-jn\omega_0 t}$$

$$f(t) = \sum_{n=1}^{\infty} c_{-n} e^{-jn\omega_0 t} + c_0 e^{j0\omega_0 t} + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t}$$

$$f(t) = \sum_{n=-\infty}^{-1} c_n e^{-jn\omega_0 t} + c_0 e^{j0\omega_0 t} + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t}$$

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t}$$

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$\begin{aligned} c_n &= \frac{1}{2} \overbrace{(a_n - jb_n)}^{\downarrow \quad \uparrow} & a_n &= \frac{2}{T} \int_0^T f(t) \cos(nw_0 t) dt \\ && b_n &= \frac{2}{T} \int_0^T f(t) \sin(nw_0 t) dt \\ c_n &= \frac{1}{2} \left[ \frac{2}{T} \int_0^T f(t) \cos(nw_0 t) dt - j \frac{2}{T} \int_0^T f(t) \sin(nw_0 t) dt \right] \\ &= \frac{1}{T} \int_0^T f(t) [\cos(nw_0 t) - j \sin(nw_0 t)] dt \end{aligned}$$

## 2. Preliminary Concepts

- Fourier series (Exponential):

$$c_n = \frac{1}{T} \int_0^T f(t) [\cos(nw_0 t) - j \sin(nw_0 t)] dt$$

$$= \frac{1}{T} \int_0^T f(t) e^{-jnw_0 t} dt$$

## 2. Preliminary Concepts

- Fourier series (Exponential):
- In summary

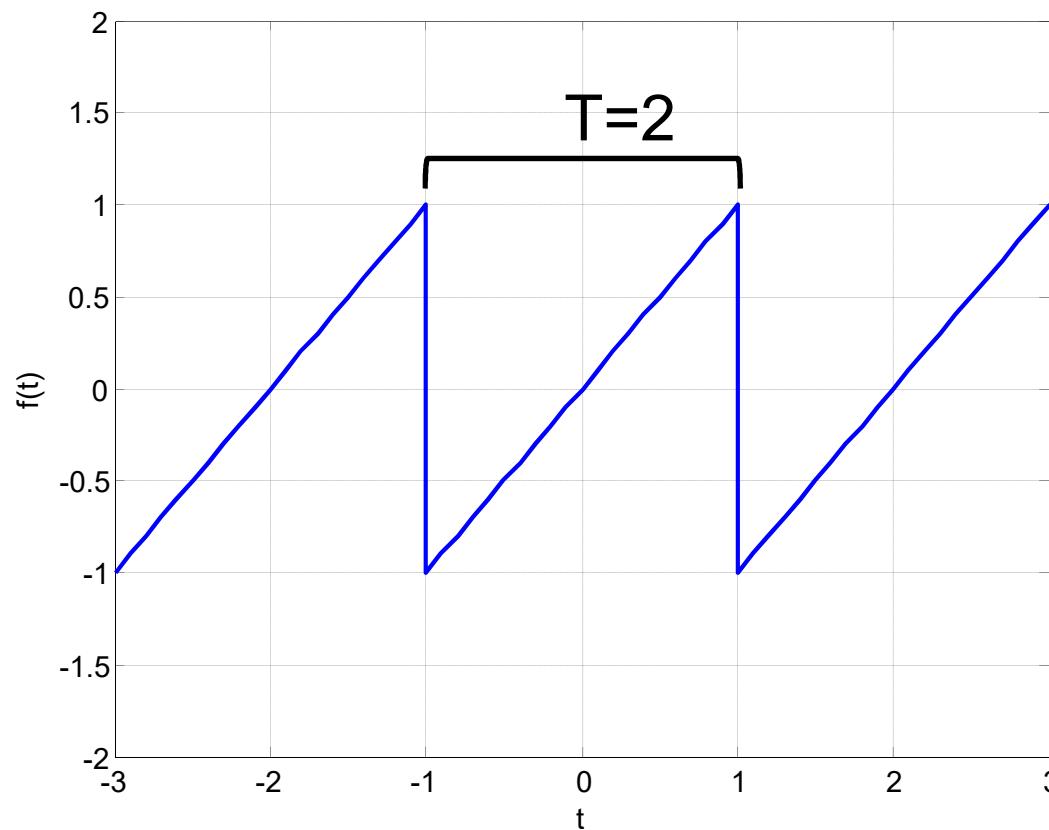
$$f(t) = \sum_{n=-\infty}^{\infty} F(n)e^{jw_0nt} \quad w_0 = \frac{2\pi}{T}$$
$$F(n) = \frac{1}{T} \int_0^T f(t)e^{-jw_0nt} dt, \quad n = 0, \pm 1, \pm 2, \dots$$

$f(t)$  is complex, continuous and periodic with period T

$F(n)$  is complex, discrete and non-periodic

## 2. Preliminary Concepts

- Example:



## 2. Preliminary Concepts

- Example:

$$\begin{aligned} F(n) &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-j\frac{2\pi n}{T}t} dt = \frac{1}{2} \int_{-1}^1 t e^{-j\frac{2\pi n}{2}t} dt = \\ &= \frac{1}{2} \int_{-1}^1 t [\cos(\pi nt) - j \sin(\pi nt)] dt \\ &= \frac{1}{2} \int_{-1}^1 t \cos(\pi nt) dt - \frac{j}{2} \int_{-1}^1 t \sin(\pi nt) dt \end{aligned}$$


## 2. Preliminary Concepts

- Example:

$$\begin{aligned} F(n) &= -\frac{j}{2} \int_{-1}^1 t \sin(\pi n t) dt = -j \int_0^1 t \sin(\pi n t) dt \\ &= -j \left[ \frac{\sin(\pi n t)}{(\pi n)^2} - t \frac{\cos(\pi n t)}{\pi n} \right]_0^1 \\ &= -j \left[ \frac{\sin(\pi n) - \pi n \cos(\pi n)}{(\pi n)^2} \right] \end{aligned}$$

$$\boxed{\int x \sin ax \, dx = \frac{\sin ax}{a^2} - \frac{x \cos ax}{a} + C}$$

## 2. Preliminary Concepts

- Example:

$$\begin{aligned} &= -j \left[ \frac{\sin(\pi n) - \pi n \cos(\pi n)}{(\pi n)^2} \right] \\ &= -j \left[ \frac{-\cos(\pi n)}{\pi n} \right] = -j \left[ \frac{-(-1)^n}{\pi n} \right] \end{aligned}$$

$$F(n) = j \frac{(-1)^n}{\pi n}, n \neq 0$$

$$F(0) = \frac{1}{T} \int_0^T f(t) e^{-j\omega_0 n t} dt = \int_{-1}^1 t dt = 0, n = 0$$

## 2. Preliminary Concepts

- Example:

$$\begin{aligned} f(t) &= \sum_{n=-\infty}^{\infty} F(n)e^{j\frac{2\pi n}{T}t} = \sum_{n=-\infty}^{\infty} j \frac{(-1)^n}{\pi n} e^{j\pi nt} \\ &= \sum_{n=-\infty}^{\infty} j \frac{(-1)^n}{\pi n} [\cos(\pi nt) + j \sin(\pi nt)], \quad n \neq 0 \\ &= \sum_{n=-\infty}^{\infty} j \frac{(-1)^n}{\pi n} \cos(\pi nt) - \sum_{n=-\infty}^{\infty} \frac{(-1)^n}{\pi n} \sin(\pi nt) \end{aligned}$$

0 

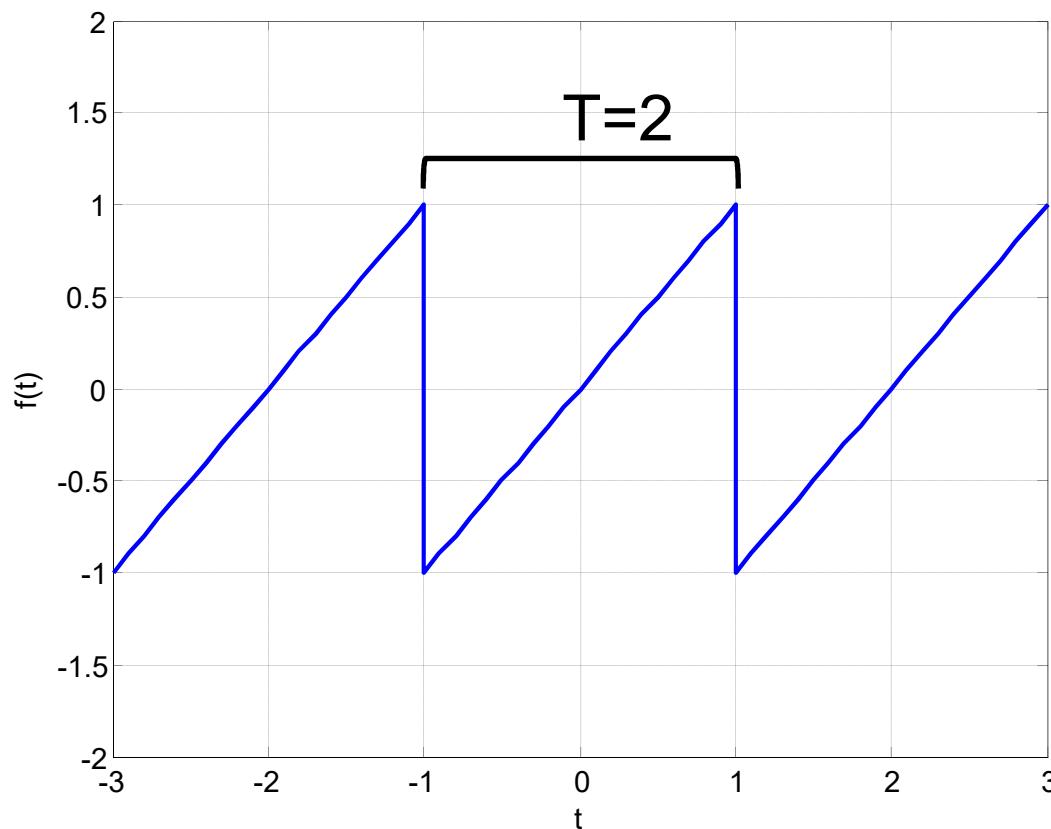
## 2. Preliminary Concepts

- Example:

$$\begin{aligned}f(t) &= -\sum_{n=-\infty}^{\infty} \frac{(-1)^n}{\pi n} \sin(\pi n t) \\&= -2 \sum_{n=1}^{\infty} \frac{(-1)^n}{\pi n} \sin(\pi n t)\end{aligned}$$

## 2. Preliminary Concepts

- MATLAB: s23FourierSeriesSawtooth.m



## 2. Preliminary Concepts

- Impulses and Their Sifting Property
  - A **unit impulse** of a continuous variable  $t$  is defined as

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

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

- An impulse may be viewed as a spike of infinity amplitude and zero duration, having unit area.

## 2. Preliminary Concepts

- Impulses and Their Sifting Property
  - An impulse has the so called **sifting property** with respect to integration

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

provided that  $f(t)$  is continuous at  $t = 0$ .

- A more general statement

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

## 2. Preliminary Concepts

- Impulses and Their Sifting Property

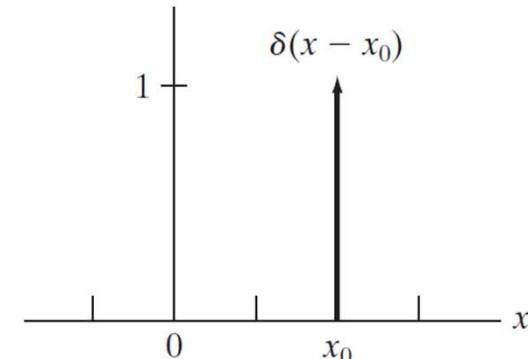
➤ The **unit discrete impulse**

$$\delta(x) = \begin{cases} 1 & x = 0 \\ 0 & x \neq 0 \end{cases} \quad \sum_{x=-\infty}^{\infty} \delta(x) = 1$$

➤ *The sifting property*

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

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

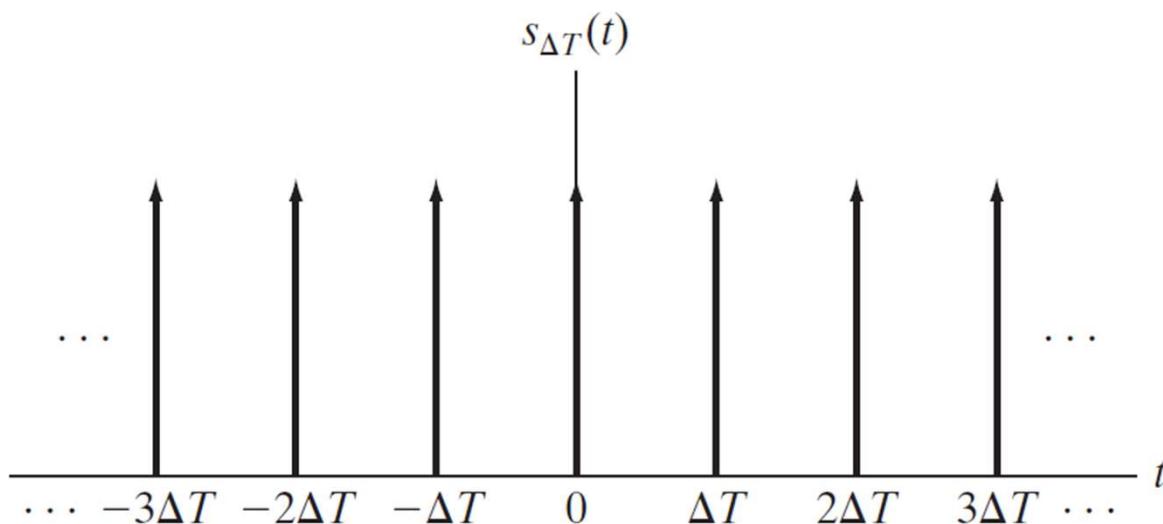


## 2. Preliminary Concepts

- Impulses and Their Sifting Property

- **Impulse train**

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



## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

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

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

## 2. Preliminary Concepts

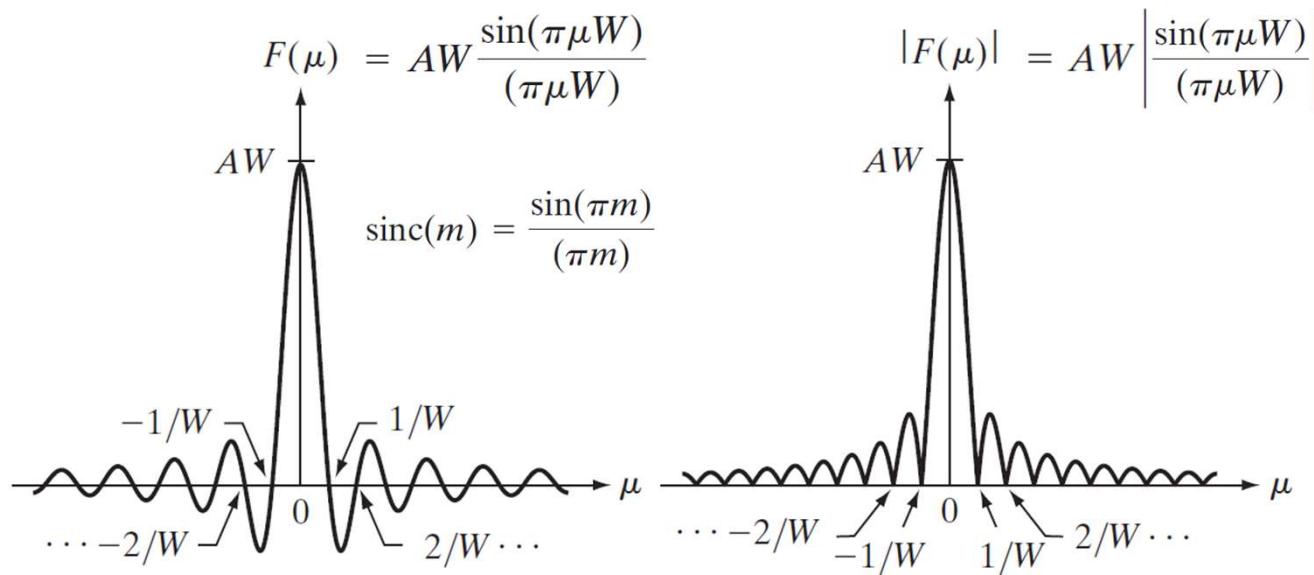
- The Fourier Transform of Functions of One Continuous Variable

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

$\int e^{cx} dx = \frac{1}{c} e^{cx}$

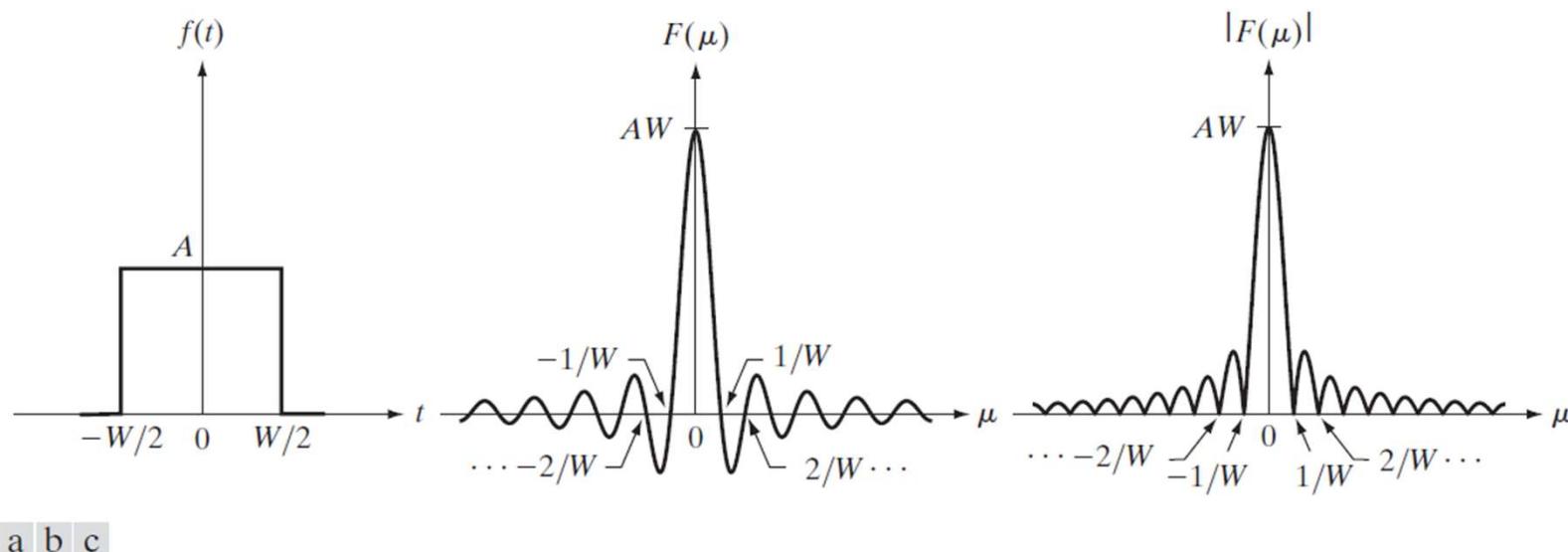
## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable
  - In general, the Fourier transform contains complex terms, and it is customary for display purposes to work with the magnitude of the transform, which is called the **frequency spectrum**.



## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable



a b c

**FIGURE 4.4** (a) A simple function; (b) its Fourier transform; and (c) the spectrum. All functions extend to infinity in both directions.

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable
  - The Fourier transform of a unit impulse located at the origin

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

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable
  - Similarly, the Fourier transform of an impulse located at  $t = t_0$

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

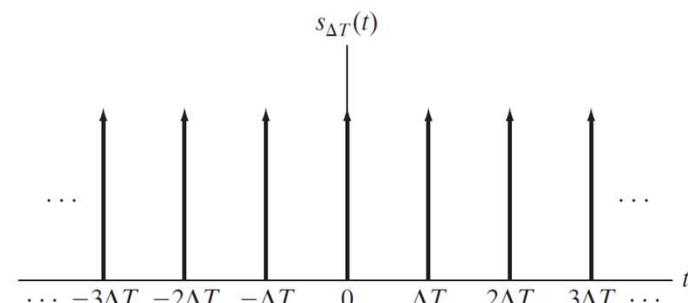
## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable
  - The **impulse train** below is periodic so that it can be expressed as a Fourier series

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

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

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



## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

➤ The preceding equation becomes

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

$$= \frac{1}{\Delta T} e^0$$

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

➤ The Fourier series expansion then becomes

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

---

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

- Our objective is to obtain the Fourier transform of this expression

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

- Because summation is a linear process, obtaining the **Fourier transform of a sum** is the same as obtaining the **sum of the transforms** of the individual components.

$$\Im \left\{ e^{j \frac{2\pi n}{\Delta T} t} \right\}$$

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable
  - Suppose we have  $f(t)$  and we know the Fourier Transform of a function  $g(t)$

$$f(t) = e^{j\frac{2\pi}{\Delta T}t} \quad G(\mu) = \delta(\mu - 1/\Delta T)$$

then

$$\begin{aligned} g(t) &= \int_{-\infty}^{+\infty} G(\mu) e^{j2\pi\mu t} d\mu \\ &= \int_{-\infty}^{+\infty} \delta(\mu - 1/\Delta T) e^{j2\pi\mu t} d\mu = e^{j\frac{2\pi}{\Delta T}t} \end{aligned}$$

It follows that  $f(t) = g(t)$

---

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

➤ If  $f(t) = g(t)$

then  $\mathcal{F}\{f(t)\} = \mathcal{F}\{g(t)\}$

and  $f(t) = e^{j\frac{2\pi}{\Delta T}t} \longleftrightarrow F(\mu) = \delta(\mu - 1/\Delta T)$

## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

➤ Therefore, if

$$f(t) = e^{j\frac{2\pi}{\Delta T}t} \longleftrightarrow F(\mu) = \delta(\mu - 1/\Delta T)$$

and

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

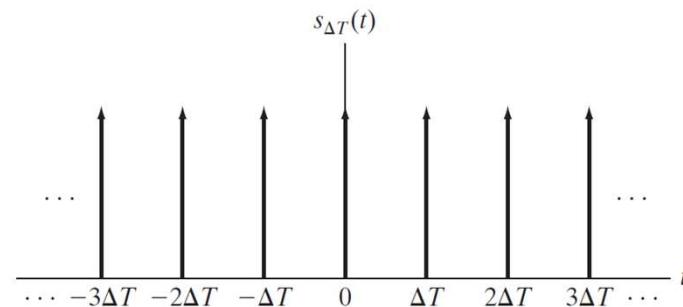
then

$$\begin{aligned} S(\mu) &= \Im\{s_{\Delta T}(t)\} = \Im\left\{\frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} e^{j\frac{2\pi n}{\Delta T}t}\right\} = \frac{1}{\Delta T} \Im\left\{\sum_{n=-\infty}^{\infty} e^{j\frac{2\pi n}{\Delta T}t}\right\} \\ &= \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \delta\left(\mu - \frac{n}{\Delta T}\right) \end{aligned}$$

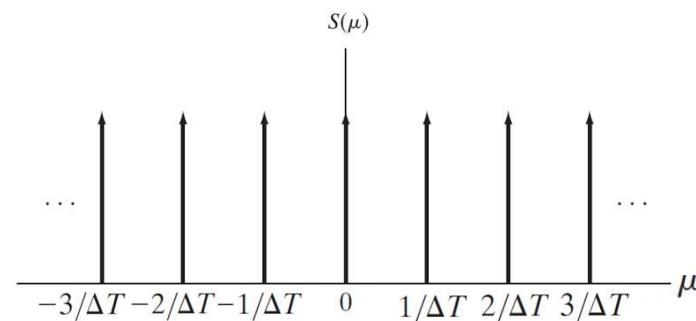
## 2. Preliminary Concepts

- The Fourier Transform of Functions of One Continuous Variable

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



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



## 2. Preliminary Concepts

- Convolution

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

- Fourier transform of the convolution

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

## 2. Preliminary Concepts

- Convolution

- It can be shown that (exercise),

$$\Im\{h(t - \tau)\} = H(\mu)e^{-j2\pi\mu\tau}$$

- From the previous equations

$$\Im\{f(t) \star h(t)\} = \int_{-\infty}^{\infty} f(\tau) [H(\mu)e^{-j2\pi\mu\tau}] d\tau$$

$$= H(\mu) \int_{-\infty}^{\infty} f(\tau) e^{-j2\pi\mu\tau} d\tau$$

$$= H(\mu) F(\mu)$$

## 2. Preliminary Concepts

- Convolution

- This result is one-half of the **convolution theorem** and is written as

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

- Following a similar development would result in the other half of the convolution theorem:

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

### 3. Sampling and Fourier Transform of Sampled Functions

- Sampling

- One way to model sampling is to multiply  $f(t)$  by a **sampling function** equal to a train of impulses  $\Delta T$  units apart

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

- The **value** of each sample is then given by the “strength” of the weighted impulse, which we obtain by integration. That is, the value,  $f_k$ , of an arbitrary sample in the sequence is given by

$$f_k = \int_{-\infty}^{\infty} f(t) \delta(t - k\Delta T) dt = f(k\Delta T)$$
$$k = \dots, -2, -1, 0, 1, 2, \dots$$

### 3. Sampling and Fourier Transform of Sampled Functions

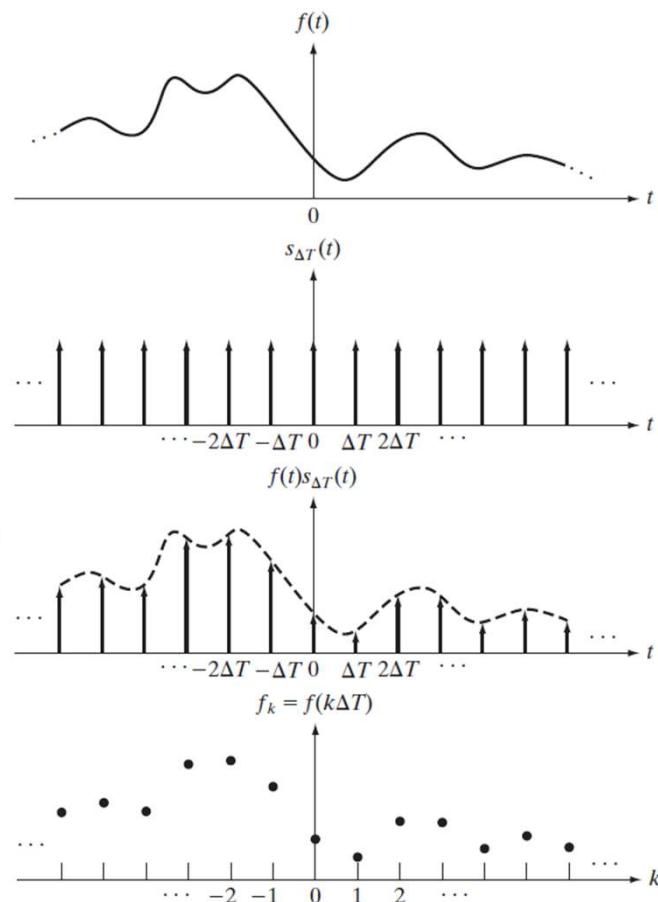
- Sampling

$$f(t)$$

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

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

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



a  
b  
c  
d

**FIGURE 4.5**  
(a) A continuous function.  
(b) Train of impulses used to model the sampling process.  
(c) Sampled function formed as the product of (a) and (b).  
(d) Sample values obtained by integration and using the sifting property of the impulse. (The dashed line in (c) is shown for reference. It is not part of the data.)

### 3. Sampling and Fourier Transform of Sampled Functions

- Fourier Transform of Sampled Functions

➤ From the convolution theorem

$$\begin{aligned}\tilde{F}(\mu) &= \Im\{\tilde{f}(t)\} \\ &= \Im\{f(t)s_{\Delta T}(t)\} \\ &= F(\mu) \star S(\mu)\end{aligned}$$

➤ And we know that

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

### 3. Sampling and Fourier Transform of Sampled Functions

- Fourier Transform of Sampled Functions

➤ But directly from the definition

$$\begin{aligned}\tilde{F}(\mu) &= F(\mu) \star S(\mu) \\ &= \int_{-\infty}^{\infty} F(\tau) S(\mu - \tau) d\tau \quad S(\mu) = \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \delta\left(\mu - \frac{n}{\Delta T}\right) \\ &= \frac{1}{\Delta T} \int_{-\infty}^{\infty} F(\tau) \sum_{n=-\infty}^{\infty} \delta\left(\mu - \tau - \frac{n}{\Delta T}\right) d\tau \\ &= \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} F(\tau) \delta\left(\mu - \tau - \frac{n}{\Delta T}\right) d\tau \\ &= \frac{1}{\Delta T} \sum_{n=-\infty}^{\infty} F\left(\mu - \frac{n}{\Delta T}\right)\end{aligned}$$

### 3. Sampling and Fourier Transform of Sampled Functions

- Fourier Transform of Sampled Functions

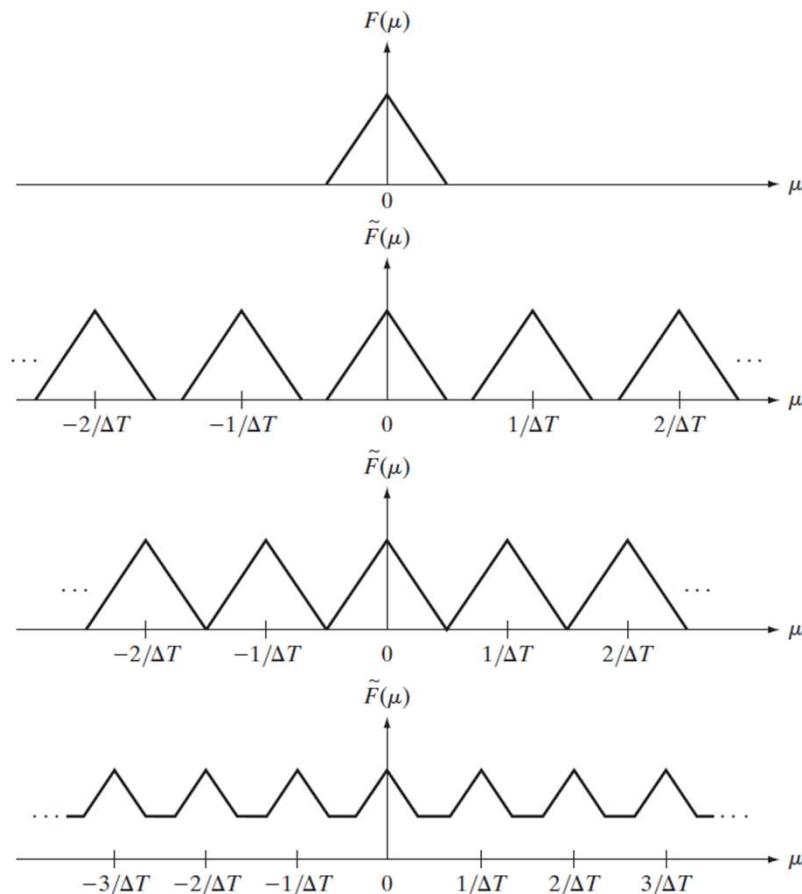
- This summation shows that the Fourier transform of a sampled function is an **infinite, periodic** sequence of **copies** of  $F(\mu)$ , the transform of the original continuous function  $f(t)$ .

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

- The separation between copies is determined by the value of  $1/\Delta T$ .
- Observe that although  $\tilde{f}(t)$  is a sampled function, its transform  $\tilde{F}(\mu)$  is continuous because it consists of copies of  $F(\mu)$  which is a continuous function.

### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem



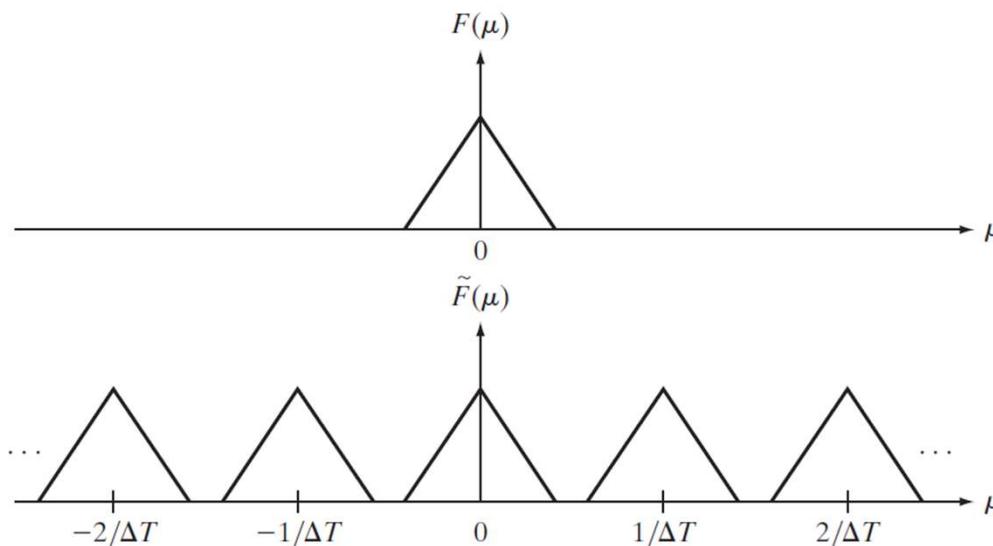
a  
b  
c  
d

**FIGURE 4.6**

(a) Fourier transform of a band-limited function.  
 (b)–(d)  
 Transforms of the corresponding sampled function under the conditions of over-sampling, critically-sampling, and under-sampling, respectively.

### 3. Sampling and Fourier Transform of Sampled Functions

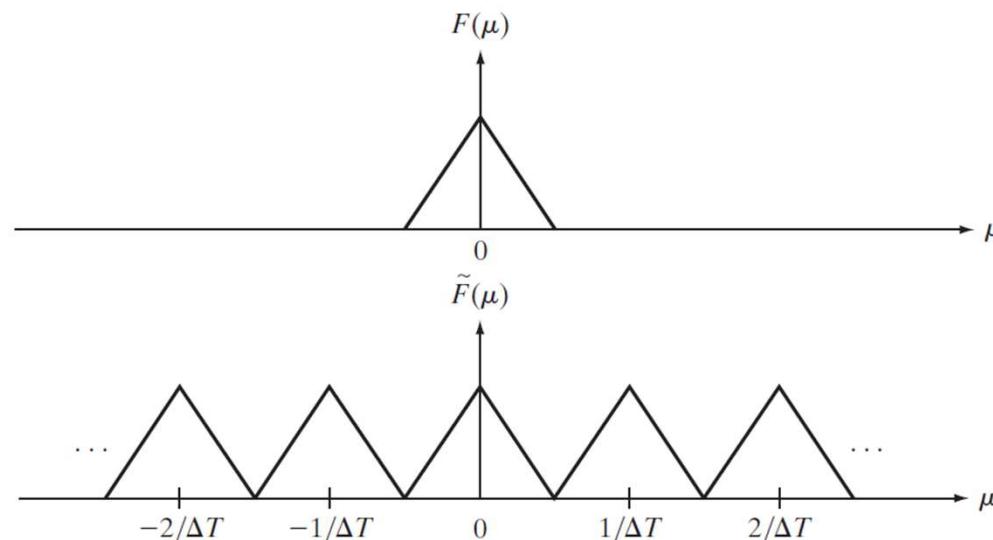
- The Sampling Theorem



- **Over-sampling:** the sampling rate was high enough to provide sufficient separation between the periods and thus preserve the integrity of  $F(\mu)$ .

### 3. Sampling and Fourier Transform of Sampled Functions

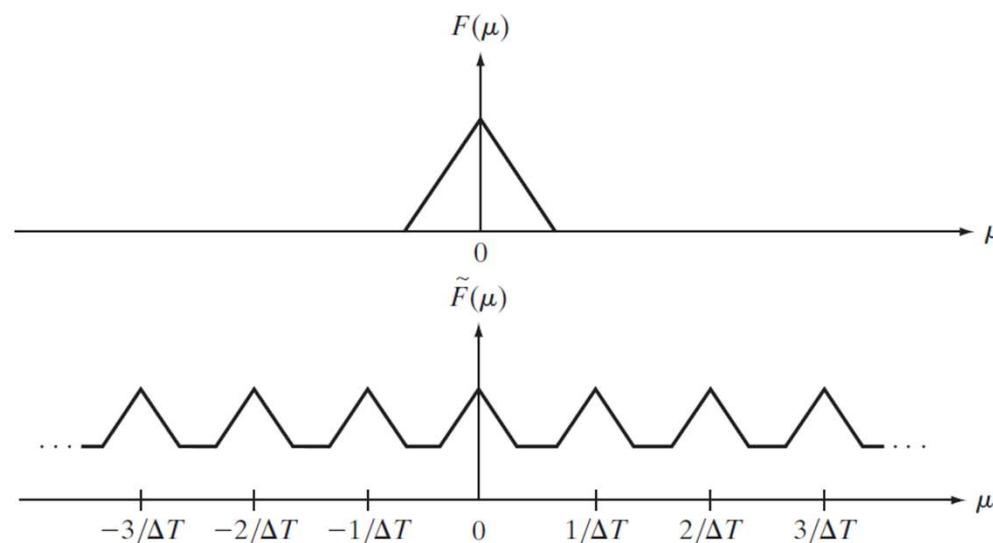
- The Sampling Theorem



- **Critically-sampling:** the sampling rate was just enough to preserve  $F(\mu)$ .

### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem

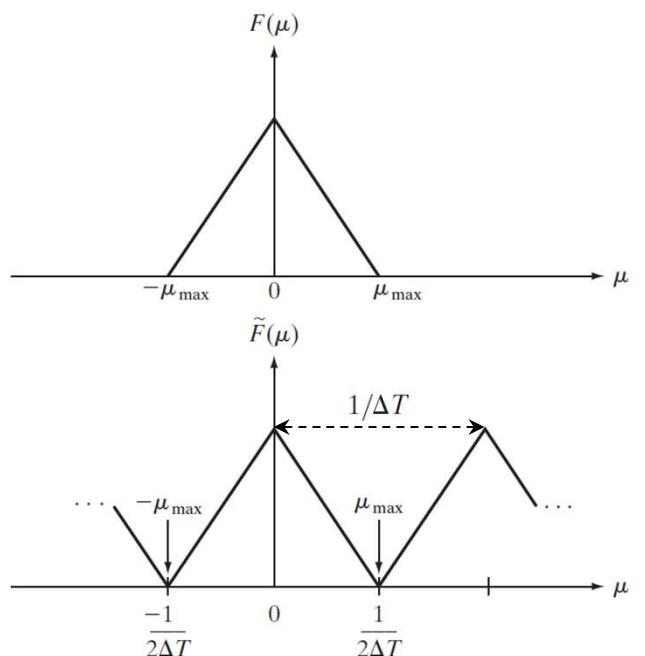


- **Under-sampling:** the sampling rate was below the minimum required to maintain distinct copies of  $F(\mu)$  and thus failed to preserve the original transform.

### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem

- A function  $f(t)$  whose Fourier transform is zero for values of frequencies outside a finite interval (band) about the origin is called a **band-limited** function.



a  
b

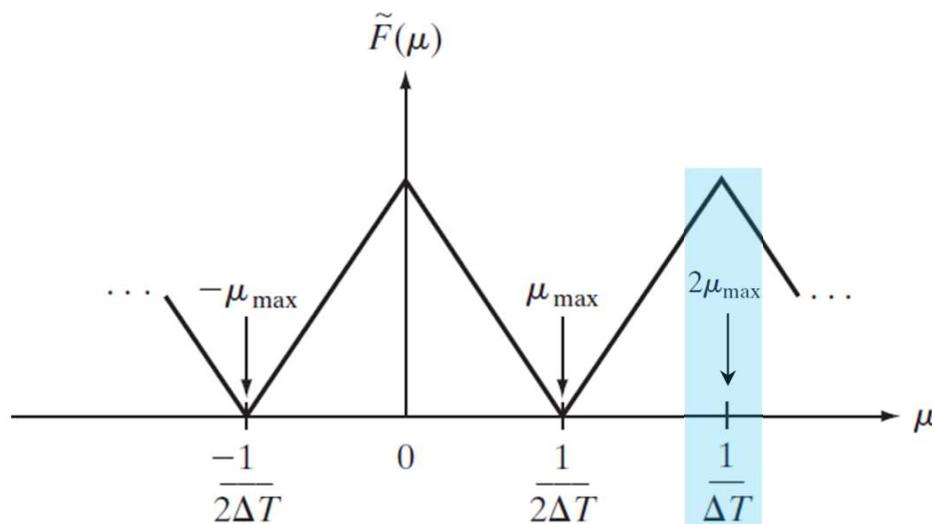
**FIGURE 4.7**  
 (a) Transform of a band-limited function.  
 (b) Transform resulting from critically sampling the same function.

### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem

- Extracting from  $\tilde{F}(\mu)$  a **single period** that is equal to  $F(\mu)$  is possible if the separation between copies is sufficient.
- Sufficient separation is guaranteed if

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



### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem

- This equation indicates that 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.**

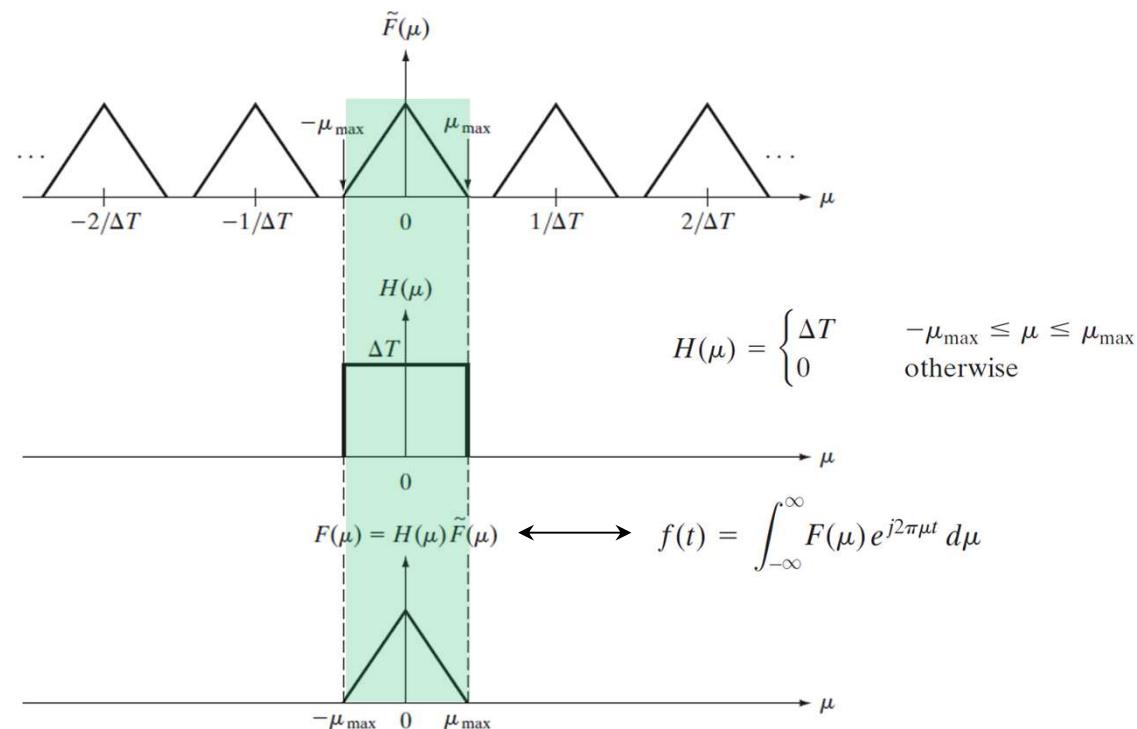
$$\mu_s > 2\mu_{\max}$$

- This result is known as the **sampling theorem**.
- A sampling rate equal to exactly twice the highest frequency is called the **Nyquist rate**.

### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem

- Extracting one period of the transform of a band-limited function using an **ideal lowpass filter**.

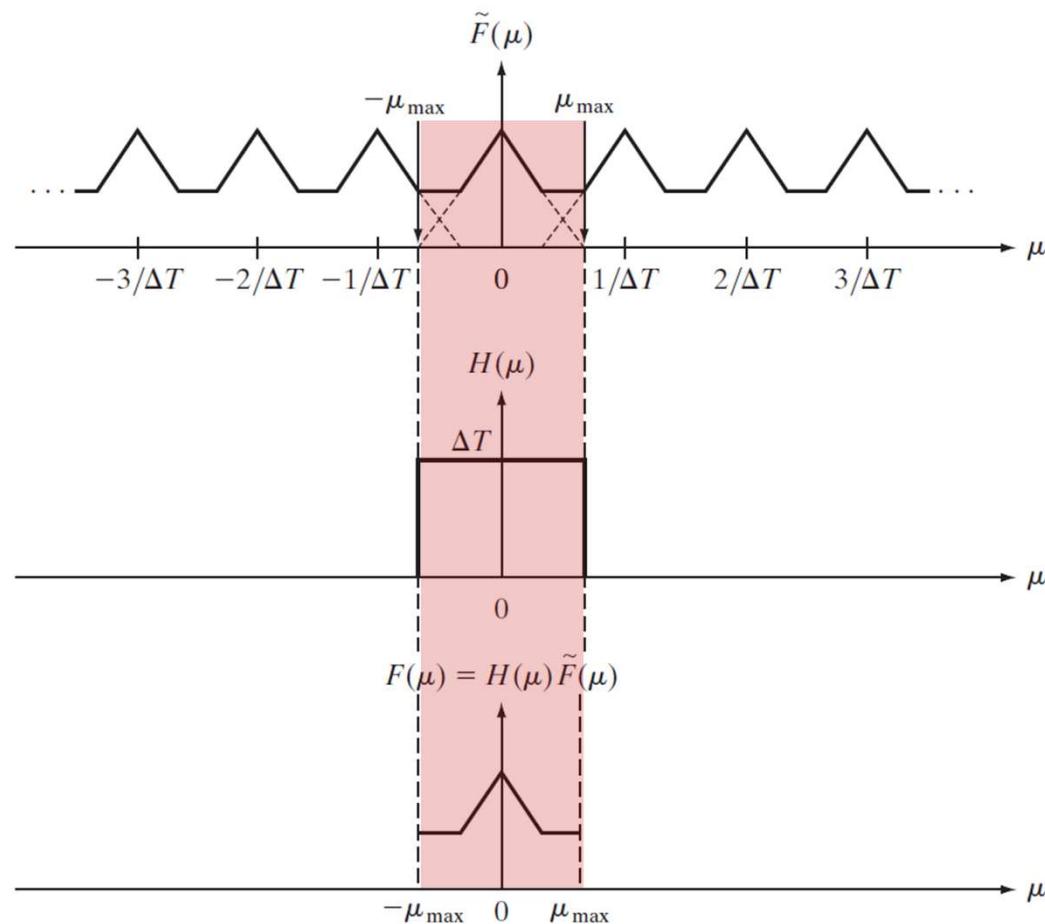


### 3. Sampling and Fourier Transform of Sampled Functions

- The Sampling Theorem
  - Function  $H(\mu)$  is called a **lowpass filter** because it passes frequencies at the low end of the frequency range but it eliminates all higher frequencies.
  - It is called also an **ideal lowpass filter** because of its infinitely rapid transitions in amplitude.
  - We will have much more to say about filtering later.

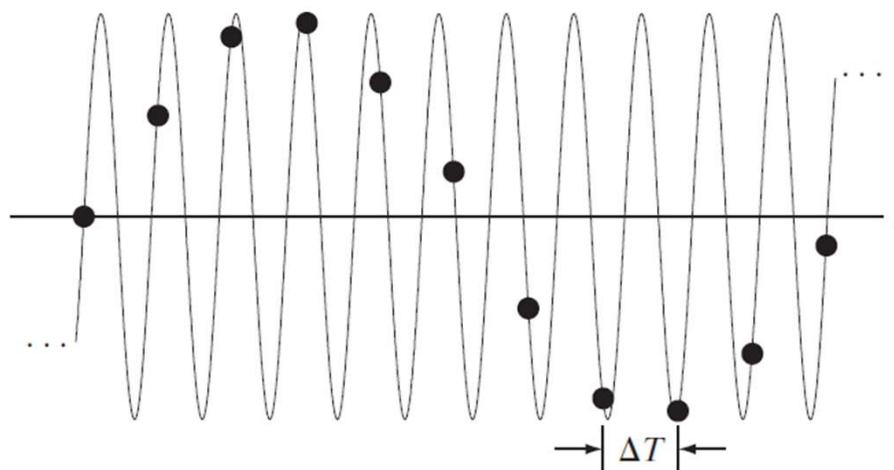
### 3. Sampling and Fourier Transform of Sampled Functions

- Aliasing



### 3. Sampling and Fourier Transform of Sampled Functions

- Aliasing



**FIGURE 4.10** Illustration of aliasing. The under-sampled function (black 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.

### 3. Sampling and Fourier Transform of Sampled Functions

- Function reconstruction
  - Using the convolution theorem

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

### 3. Sampling and Fourier Transform of Sampled Functions

- Function reconstruction

➤ Using the convolution theorem

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

$$h(t) = \mathfrak{I}^{-1}\{H(u)\} = \int_{-\mu_{\max}}^{\mu_{\max}} \Delta T e^{j2\pi u t} du = \frac{\Delta T}{\pi t} \sin(2\pi\mu_{\max} t)$$

$$= 2\Delta T \mu_{\max} \frac{\sin(\pi 2\mu_{\max} t)}{\pi 2\mu_{\max} t} = 2\Delta T \mu_{\max} \text{sinc}(2\mu_{\max} t)$$

$$= 2\Delta T \frac{1}{2\Delta T} \text{sinc}\left(\frac{t}{\Delta T}\right) = \text{sinc}\left(\frac{t}{\Delta T}\right)$$

$$\boxed{\int e^{cx} dx = \frac{1}{c} e^{cx}}$$

$$\boxed{\text{sinc}(m) = \frac{\sin(\pi m)}{(\pi m)}}$$

### 3. Sampling and Fourier Transform of Sampled Functions

- Function reconstruction

➤ Using the convolution theorem

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

$$f(t) = \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(\tau) \delta(\tau - n\Delta T) h(t - \tau) d\tau$$

$$= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} f(\tau) h(t - \tau) \delta(\tau - n\Delta T) d\tau$$

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

$$\boxed{h(t) = \text{sinc}\left(\frac{t}{\Delta T}\right) \rightarrow h(t - n\Delta T) = \text{sinc}\left(\frac{t - n\Delta T}{\Delta T}\right)}$$

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

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function.

- In practice, we work with a finite number of samples in time and in frequency domain.
- First we find  $\tilde{F}(u)$  in terms of the sampled function  $\tilde{f}(t)$  itself.

$$\begin{aligned}
 \tilde{F}(\mu) &= \int_{-\infty}^{\infty} \tilde{f}(t) e^{-j2\pi\mu t} dt \\
 &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(t) \delta(t - n\Delta T) e^{-j2\pi\mu t} dt \\
 &= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} f(t) \delta(t - n\Delta T) e^{-j2\pi\mu t} dt \\
 &= \sum_{n=-\infty}^{\infty} f_n e^{-j2\pi\mu n\Delta T}
 \end{aligned}$$

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

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function
  - Although  $f_n$  is a discrete function, its Fourier  $\tilde{F}(\mu)$  is continuous and infinitely periodic with period  $1/\Delta T$  (see slide 48).
  - All we need to characterize  $\tilde{F}(\mu)$  is one period, and sampling one period is the basis for the DFT.
  - Suppose that we want to obtain  $M$  equally spaced samples of  $\tilde{F}(\mu)$  taken from  $\mu = 0$  to  $\mu = 1/\Delta T$ .
  - This is accomplished by taking the samples at the following frequencies:

$$\mu = \frac{m}{M\Delta T} \quad m = 0, 1, 2, \dots, M - 1$$

---

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function

➤ Substituting  $\mu = \frac{m}{M\Delta T}$        $m = 0, 1, 2, \dots, M - 1$

into

$$\tilde{F}(\mu) = \sum_{n=-\infty}^{\infty} f_n e^{-j2\pi\mu n \Delta T}$$

and letting  $F_m$  denote the result, then

$$F_m = \sum_{n=0}^{M-1} f_n e^{-j2\pi m n / M} \quad m = 0, 1, 2, \dots, M - 1$$

- This is the **discrete Fourier transform** we are seeking. A set  $\{f_n\}$  consisting of  $M$  samples of  $f(t)$  yields a set  $\{F_m\}$  of  $M$  complex discrete values.

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function

- Conversely, given  $\{F_m\}$  we can recover the sample set  $\{f_n\}$  by using the IDFT.

$$f_n = \frac{1}{M} \sum_{m=0}^{M-1} F_m e^{j2\pi mn/M} \quad n = 0, 1, 2, \dots, M-1$$

- Demonstration

$$\begin{aligned} f_n &= \frac{1}{M} \sum_{m=0}^{M-1} \left\{ \sum_{k=0}^{M-1} f_k e^{-j2\pi mk/M} \right\} e^{j2\pi nm/M} \\ &= \sum_{k=0}^{M-1} f_k \left\{ \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m(k-n)}{M}} \right\} = \\ &= \sum_{k=0}^{M-1} f_k \delta(k-n) = f_n, \quad n = 0 \dots M-1 \end{aligned}$$

$$\begin{aligned} \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m(k-n)}{M}} &= \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m\alpha}{M}}, \quad \alpha = (k-n) \\ \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m\alpha}{M}} &= 0, \text{ for } \alpha \neq 0 \\ \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m\alpha}{M}} &= 1, \text{ for } \alpha = 0 \end{aligned} \left. \right\} \frac{1}{M} \sum_{m=0}^{M-1} e^{j2\pi \frac{m(k-n)}{M}} = \delta(k-n)$$

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function

- It is more intuitive, especially in two dimensions, to use the notation  $x$  and  $y$  for image coordinate variables and  $u$  and  $v$  for frequency variables.

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

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

- It can be shown that both the forward and inverse discrete transforms are **infinitely periodic**, with period  $M$ .

$$f(x) = f(x + kM) \quad F(u) = F(u + kM)$$

---

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

- Obtaining the DFT from the Continuous Transform of a Sampled Function

- The discrete equivalent of the convolution

$$f(x) \star h(x) = \sum_{m=0}^{M-1} f(m)h(x-m) \quad x = 0, 1, 2, \dots, M-1$$

- Because in the preceding formulations the functions are periodic, their convolution is also periodic.
  - For this reason, the process inherent in this equation often is referred to as **circular convolution**, and is a direct result of the periodicity of the DFT and its inverse.
  - Finally, the convolution theorem is applicable also to discrete variables.

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

- Relationship between the sampling and frequency intervals

- If  $f(x)$  consists of  $M$  samples of a function  $f(t)$  taken  $\Delta T$  units apart, the duration of the record comprising the set

$$\{f(x)\}, x = 0, 1, 2, \dots, M - 1$$

is

$$T = M\Delta T$$

- The corresponding spacing  $\Delta u$ , in the discrete frequency domain follows from

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

- The entire frequency range spanned by the  $M$  components of the DFT is

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

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

- Relationship between the sampling and frequency intervals

➤ Thus, we see that the resolution in frequency,  $\Delta u$ , of the DFT depends on the duration  $T$  over which the continuous function,  $f(t)$ , is sampled.

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

➤ The range of frequencies spanned by the DFT depends on the sampling interval.

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

➤ Observe that both expressions exhibit inverse relationships with respect to  $T$  and  $\Delta T$ .

## 5. Extension to Functions of Two Variables

- The 2-D Impulse and Its Sifting Property
  - The impulse of two continuous variables is given by

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

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

- The 2-D impulse exhibits the sifting property under integration

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

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

## 5. Extension to Functions of Two Variables

- The 2-D Impulse and Its Sifting Property

- For discrete variables

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

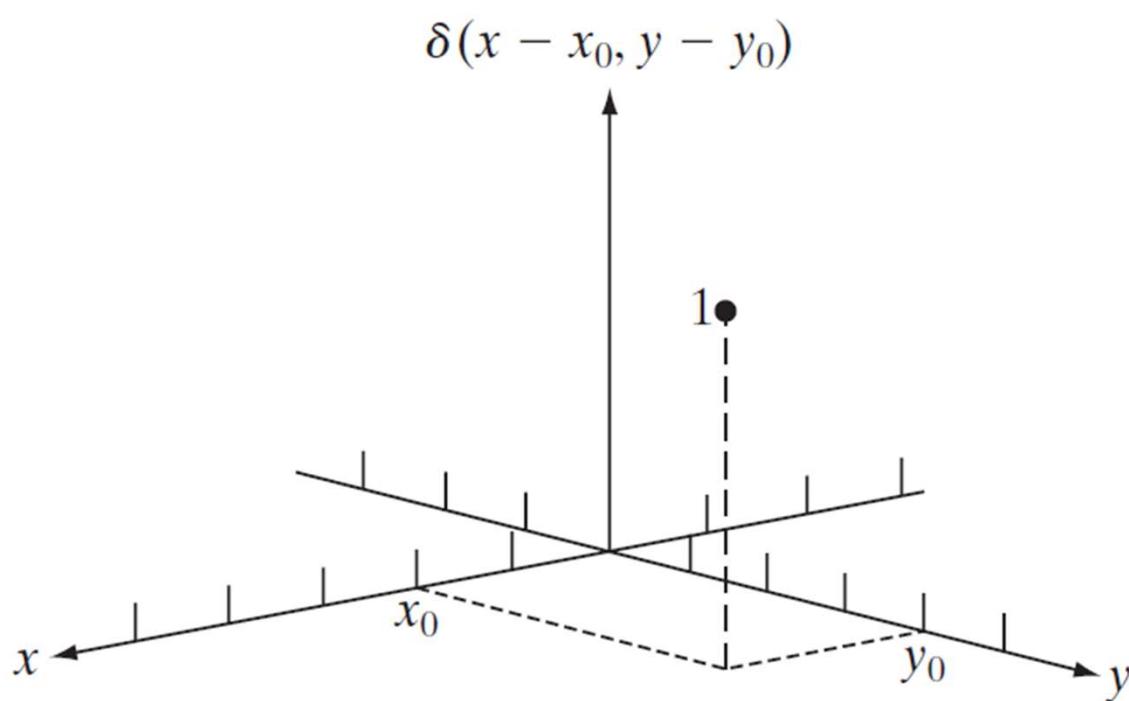
$$\sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} f(x, y) \delta(x, y) = f(0, 0)$$

- For an impulse located at coordinates  $(x_0, y_0)$

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

## 5. Extension to Functions of Two Variables

- The 2-D Impulse and Its Sifting Property
  - For an impulse located at coordinates  $(x_0, y_0)$



**FIGURE 4.12**  
Two-dimensional unit discrete impulse. Variables  $x$  and  $y$  are discrete, and  $\delta$  is zero everywhere except at coordinates  $(x_0, y_0)$ .

## 5. Extension to Functions of Two Variables

- The 2-D Continuous Fourier Transform Pair
  - Let  $f(t, z)$  be a continuous function of two continuous variables,  $t$  and  $z$ .
  - The two-dimensional, continuous Fourier transform pair is given by the expressions

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

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

## 5. Extension to Functions of Two Variables

- The 2-D Continuous Fourier Transform Pair

➤ Example

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

$$= \int_{-T/2}^{T/2} \int_{-Z/2}^{Z/2} A e^{-j2\pi(\mu t + \nu z)} dt dz$$

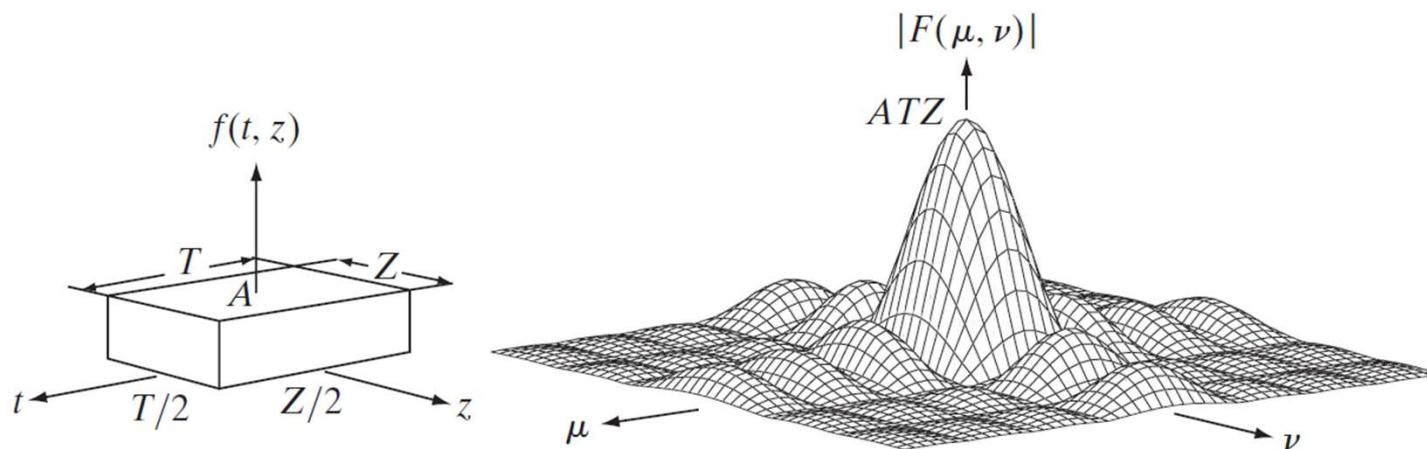
$$= ATZ \left[ \frac{\sin(\pi\mu T)}{(\pi\mu T)} \right] \left[ \frac{\sin(\pi\nu Z)}{(\pi\nu Z)} \right]$$

$$|F(\mu, \nu)| = ATZ \left| \frac{\sin(\pi\mu T)}{(\pi\mu T)} \right| \left| \frac{\sin(\pi\nu Z)}{(\pi\nu Z)} \right|$$

## 5. Extension to Functions of Two Variables

- The 2-D Continuous Fourier Transform Pair

➤ Example



a b

**FIGURE 4.13** (a) A 2-D function, and (b) a section of its spectrum (not to scale). The block is longer along the  $t$ -axis, so the spectrum is more “contracted” along the  $\mu$ -axis. Compare with Fig. 4.4.

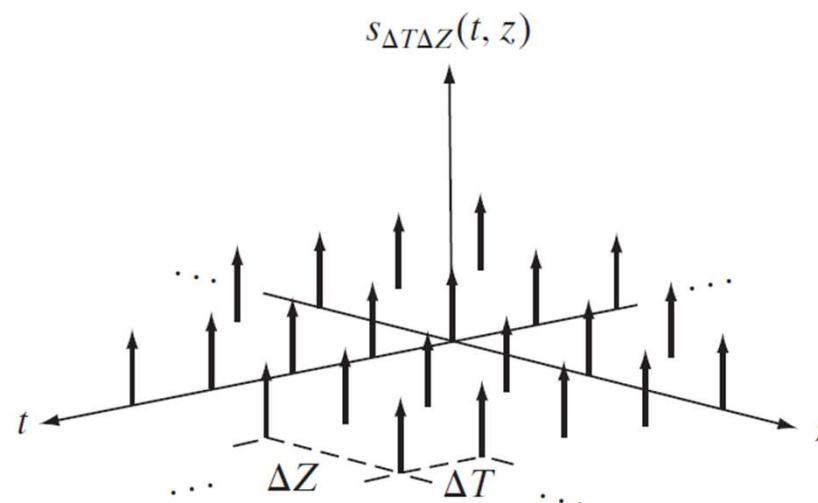
## 5. Extension to Functions of Two Variables

- Two-Dimensional Sampling and the 2-D Sampling Theorem

➤ 2-D impulse train

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

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



## 5. Extension to Functions of Two Variables

- Two-Dimensional Sampling and the 2-D Sampling Theorem
  - The 2-D sampling theorem states that a continuous, band-limited function can be recovered with no error from a set of its samples if the sampling intervals are

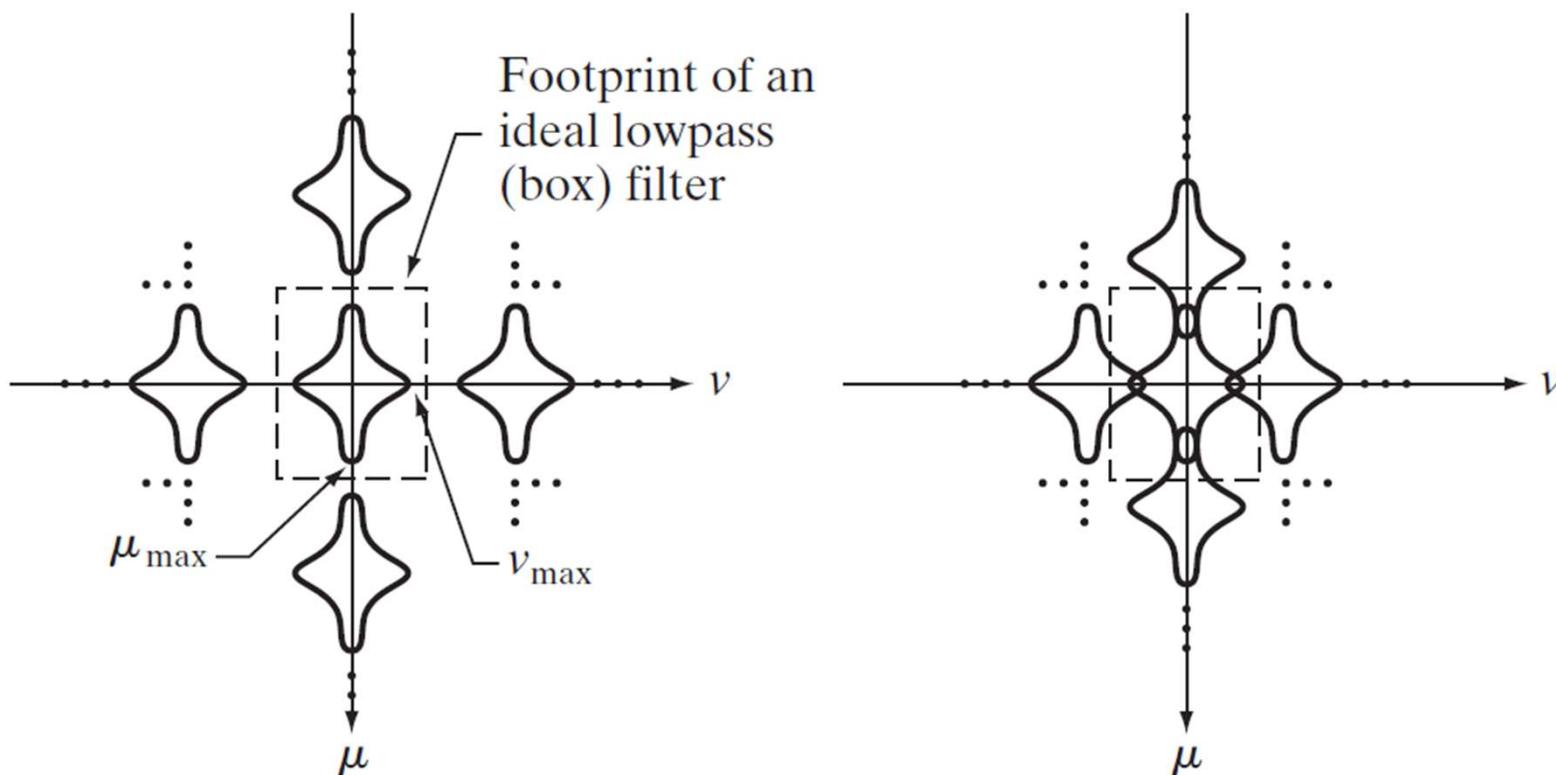
$$\frac{1}{\Delta T} > 2\mu_{\max} \quad \frac{1}{\Delta Z} > 2\nu_{\max}$$

- Stated another way, no information is lost if a 2-D, band-limited, continuous function is represented by samples acquired at rates greater than twice the highest frequency content of the function in both the  $\mu$ -  $\nu$ -directions.

$$\mu_s > 2\mu_{\max} \quad \nu_s > 2\nu_{\max}$$

## 5. Extension to Functions of Two Variables

- Aliasing in Images

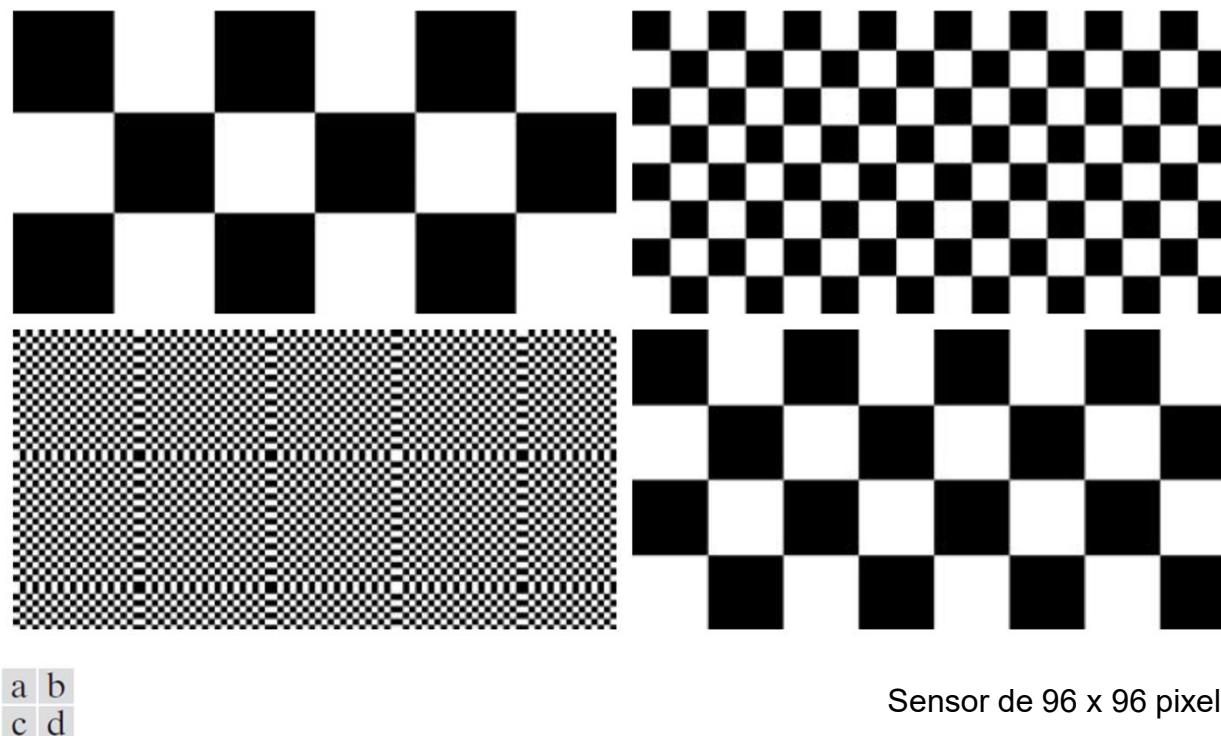


## 5. Extension to Functions of Two Variables

- Aliasing in Images (sampling)
  - Suppose an imaging system whose number of samples it can take is fixed at  $96 \times 96$  pixels.
  - If we use this system to digitize checkerboard patterns, it will be able to resolve patterns that are up to  $96 \times 96$  pixels in which the size of each square is  $1 \times 1$  pixels.
  - What happens when the detail (the size of the checkerboard squares) is less than one camera pixel?

## 5. Extension to Functions of Two Variables

- Aliasing in Images (sampling)



**FIGURE 4.16** Aliasing in images. In (a) and (b), the lengths of the sides of the squares are 16 and 6 pixels, respectively, and aliasing is visually negligible. In (c) and (d), the sides of the squares are 0.9174 and 0.4798 pixels, respectively, and the results show significant aliasing. Note that (d) masquerades as a “normal” image.

## 5. Extension to Functions of Two Variables

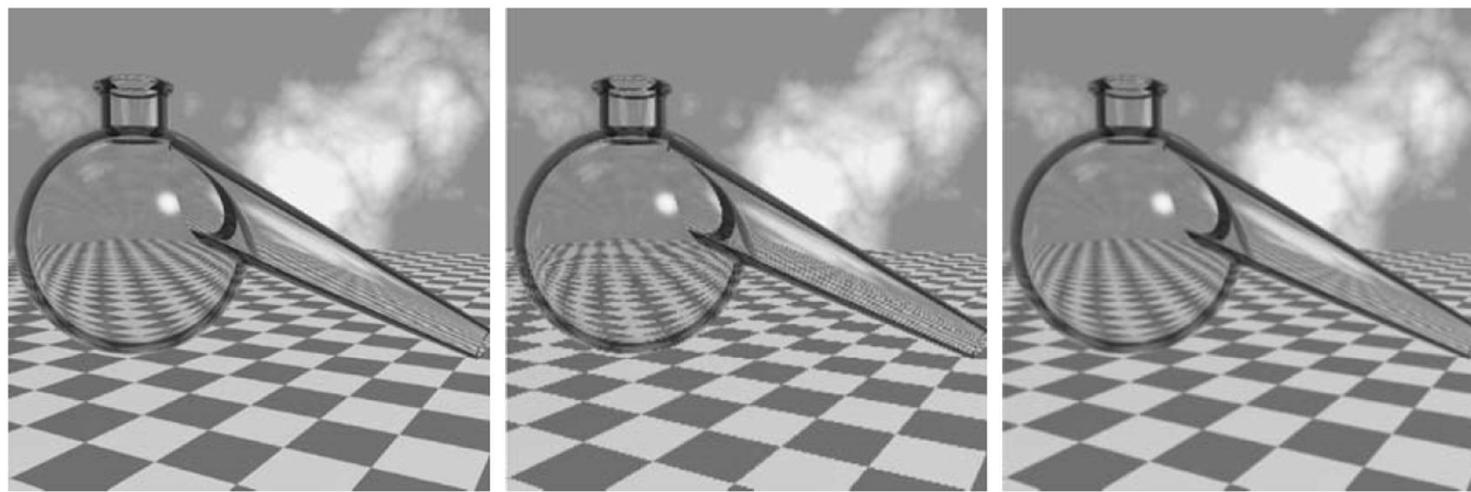
- Aliasing in Images (interpolation and resampling)



**FIGURE 4.17** Illustration of aliasing on resampled images. (a) A digital image with negligible visual aliasing. (b) Result of resizing the image to 50% of its original size by pixel deletion. Aliasing is clearly visible. (c) Result of blurring the image in (a) with a  $3 \times 3$  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.)

## 5. Extension to Functions of Two Variables

- Aliasing in Images (interpolation and resampling)
  - When you work with images that have strong edge content, the effects of aliasing are seen as block-like image components, called *jaggies*.

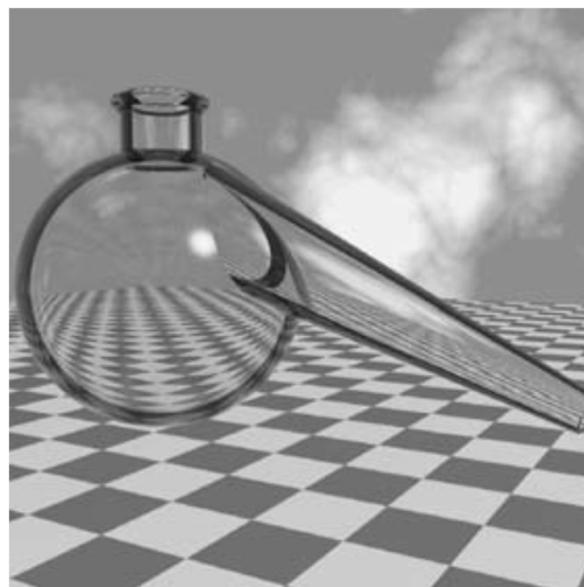


a b c

**FIGURE 4.18** Illustration of jaggies. (a) A  $1024 \times 1024$  digital image of a computer-generated scene with negligible visible aliasing. (b) Result of reducing (a) to 25% of its original size using bilinear interpolation. (c) Result of blurring the image in (a) with a  $5 \times 5$  averaging filter prior to resizing it to 25% using bilinear interpolation. (Original image courtesy of D. P. Mitchell, Mental Landscape, LLC.)

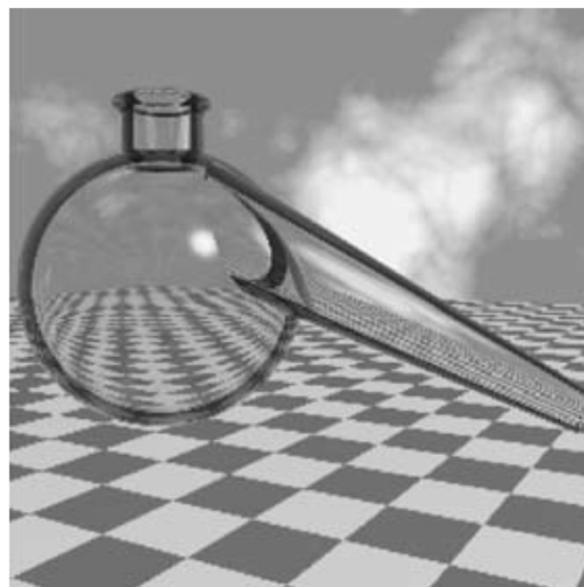
## 5. Extension to Functions of Two Variables

- Aliasing in Images (interpolation and resampling)
  - When you work with images that have strong edge content, the effects of aliasing are seen as block-like image components, called *jaggies*.



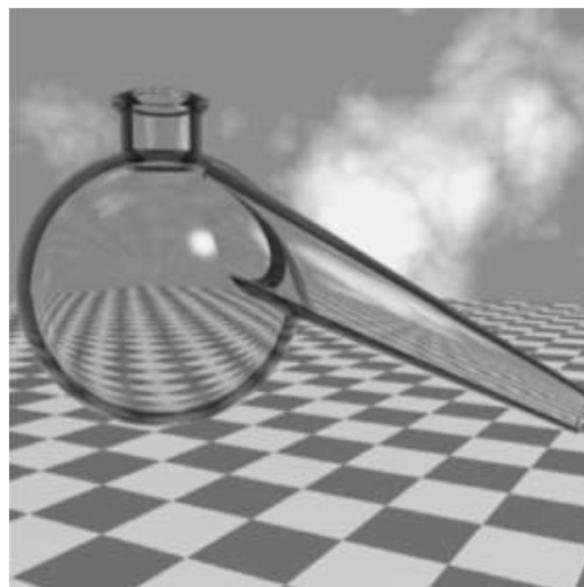
## 5. Extension to Functions of Two Variables

- Aliasing in Images (interpolation and resampling)
  - When you work with images that have strong edge content, the effects of aliasing are seen as block-like image components, called *jaggies*.



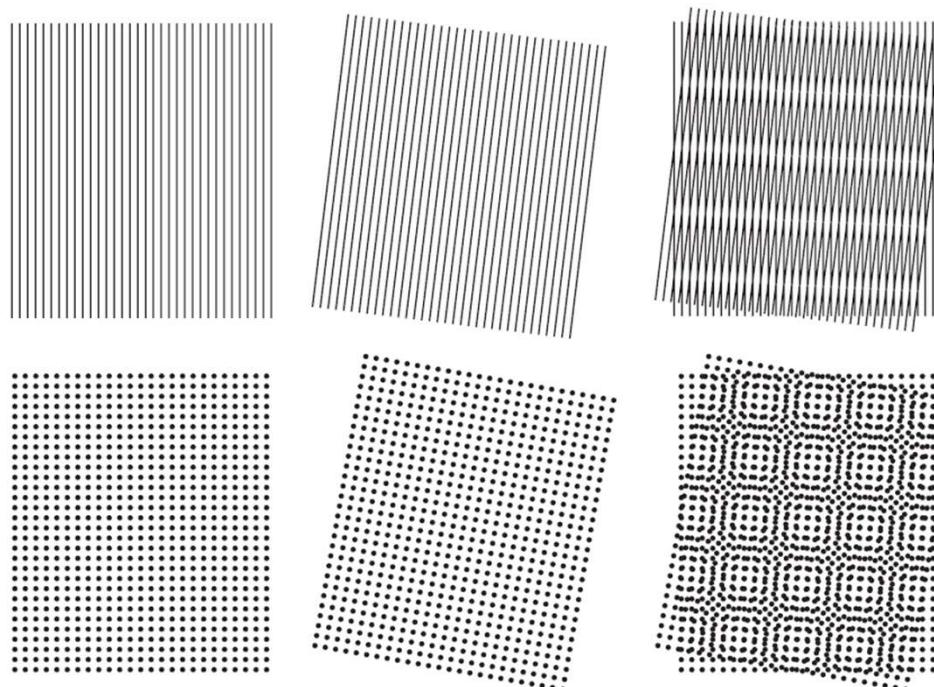
## 5. Extension to Functions of Two Variables

- Aliasing in Images (interpolation and resampling)
  - When you work with images that have strong edge content, the effects of aliasing are seen as block-like image components, called *jaggies*.



## 5. Extension to Functions of Two Variables

- Aliasing in Images (Moiré patterns)
  - This problem arises routinely when scanning media print, such as newspapers and magazines, or in images with periodic components.



a	b	c
d	e	f

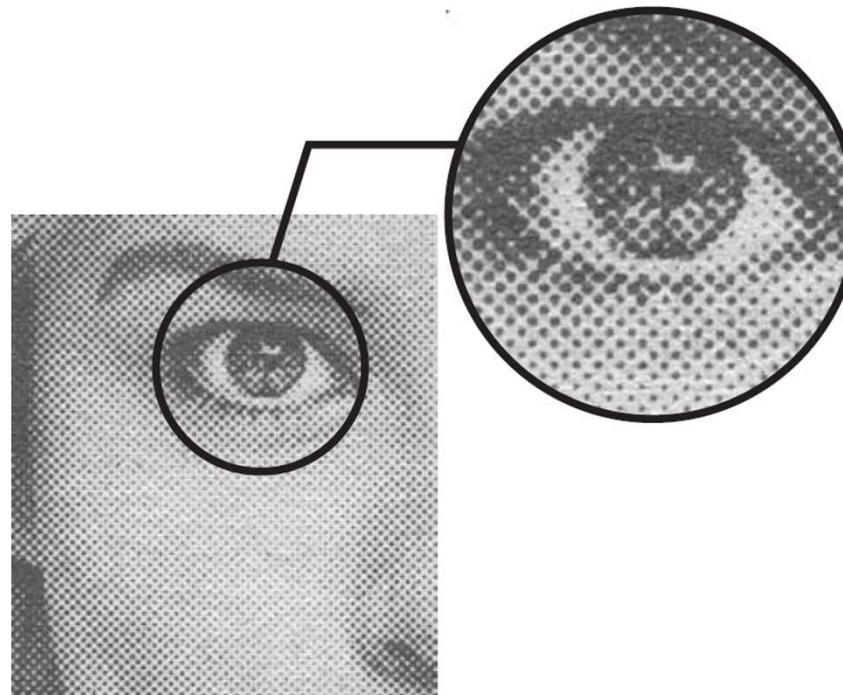
**FIGURE 4.20**

Examples of the moiré effect.  
These are ink drawings, not digitized patterns.  
Superimposing one pattern on the other is equivalent mathematically to multiplying the patterns.

## 5. Extension to Functions of Two Variables

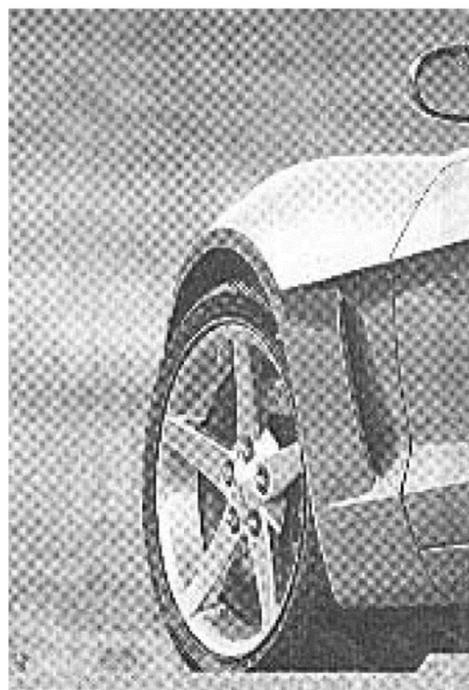
- Aliasing in Images (Moiré patterns)
  - Printed materials make use of halftone dots, which are black dots whose sizes and joining schemes are used to simulate gray tones.

**FIGURE 4.22**  
A newspaper image and an enlargement showing how halftone dots are arranged to render shades of gray.



## 5. Extension to Functions of Two Variables

- Aliasing in Images (Moiré patterns)
  - Printed materials make use of halftone dots, which are black dots whose sizes and joining schemes are used to simulate gray tones.



**FIGURE 4.21**  
A newspaper image of size  $246 \times 168$  pixels sampled at 75 dpi showing a moiré pattern. The moiré pattern in this image is the interference pattern created between the  $\pm 45^\circ$  orientation of the halftone dots and the north-south orientation of the sampling grid used to digitize the image.

## 5. Extension to Functions of Two Variables

- The 2-D Discrete Fourier Transform and Its Inverse

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

$u = 0, 1, 2, \dots, M - 1$  and  $v = 0, 1, 2, \dots, N - 1$ .

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

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

## 6. Some Properties of the 2-D DFT

- Relationships Between 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

$$\Delta u = \frac{1}{M\Delta T} \quad \Delta v = \frac{1}{N\Delta Z}$$

## 6. Some Properties of the 2-D DFT

- Translation

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

$$f(x - x_0, y - y_0) \Leftrightarrow F(u, v) e^{-j2\pi(x_0u/M + y_0v/N)}$$

- Rotation

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

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

## 6. Some Properties of the 2-D DFT

- Periodicity
  - The 2-D Fourier transform and its inverse are infinitely periodic in the  $u$  and  $v$  directions; that is,

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

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

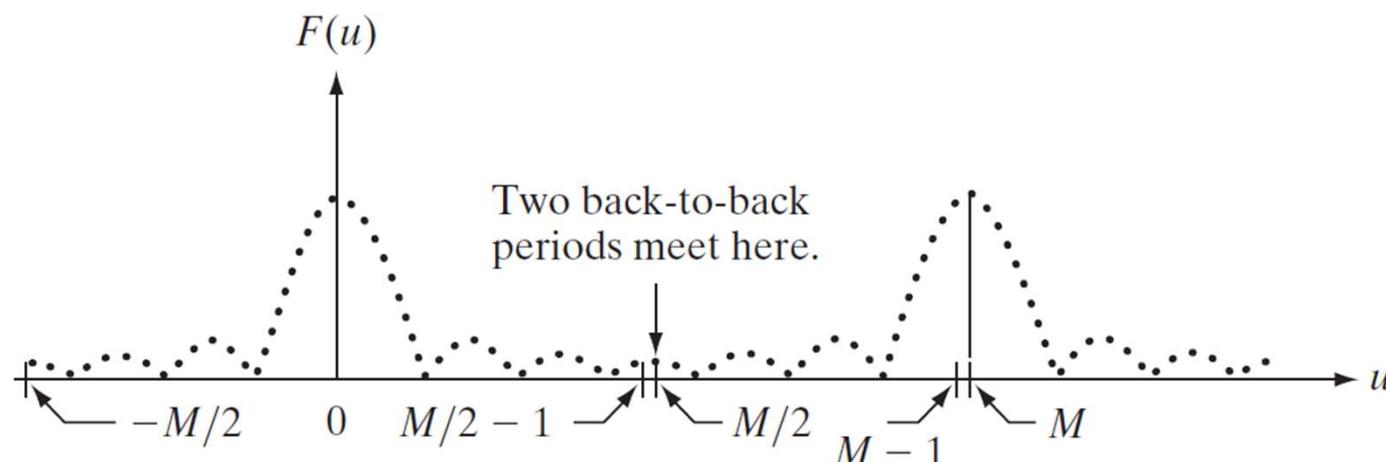
where  $M$  and  $N$  are the number of samples in  $u$  and  $v$  directions.

---

## 6. Some Properties of the 2-D DFT

- Periodicity

- Consider the 1-D spectrum. The transform data in the interval from 0 to  $M-1$  consists of two back-to-back half periods meeting at point  $M/2$ .



## 6. Some Properties of the 2-D DFT

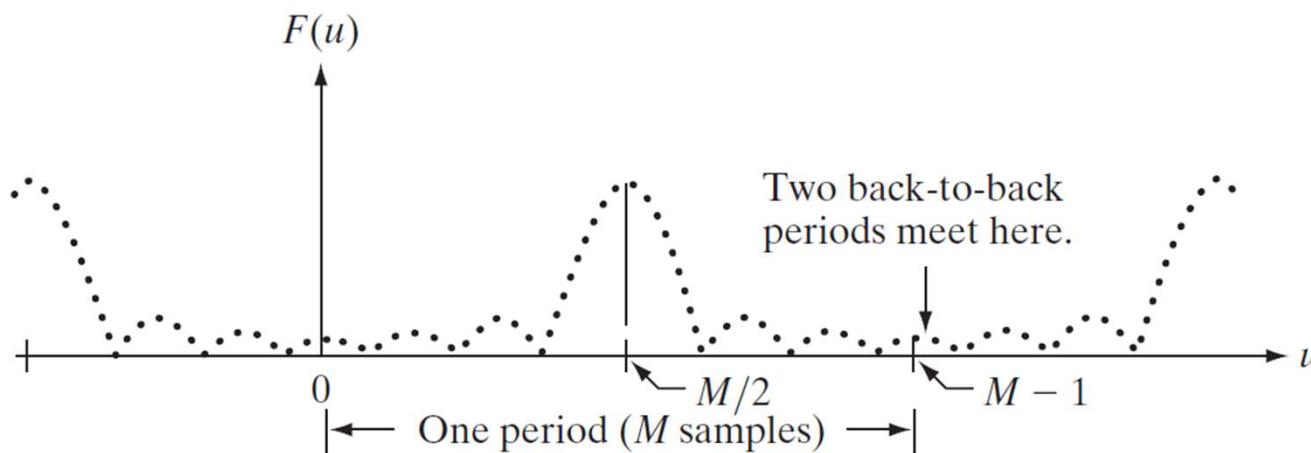
- Periodicity
  - For display and filtering purposes, it is more convenient to have in this interval a complete period of the transform in which the data are contiguous.
  - We know that  $f(x)e^{j2\pi(u_0x/M)} \Leftrightarrow F(u - u_0)$
  - In other words, multiplying  $f(x)$  by the exponential term shifts the data so that the origin,  $F(0)$ , is located at  $u_0$ .
  - If we let  $u_0 = M/2$ ,  $f(x)(-1)^x \Leftrightarrow F(u - M/2)$
  - That is, shifts the data so that  $F(0)$  is at the center of the interval  $[0, M-1]$ .

## 6. Some Properties of the 2-D DFT

- Periodicity

- For display and filtering purposes, it is more convenient to have in this interval a complete period of the transform in which the data are contiguous.

$$f(x)(-1)^x \Leftrightarrow F(u - M/2)$$

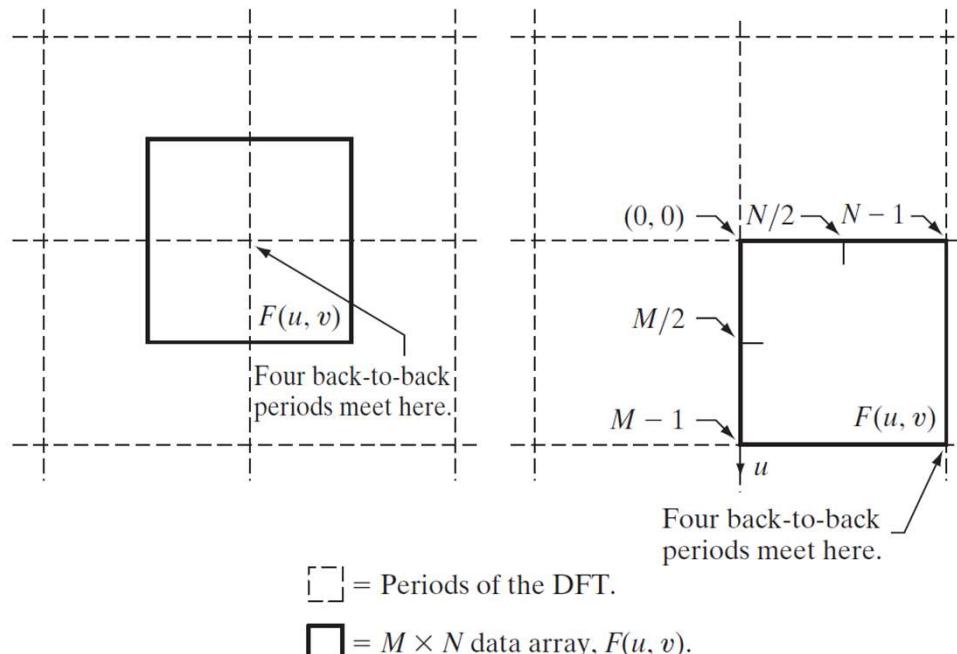


## 6. Some Properties of the 2-D DFT

- Periodicity

- In 2-D the situation is more difficult to graph, but the principle is the same.

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



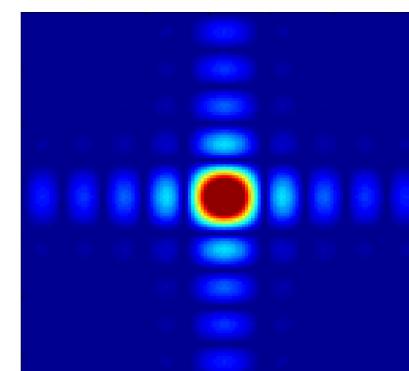
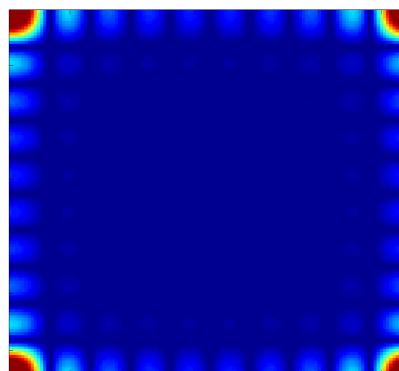
## 6. Some Properties of the 2-D DFT

- Periodicity

- In 2-D the situation is more difficult to graph, but the principle is the same.

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

- Using this equation shifts the data so that  $F(0,0)$  is at the center of the frequency rectangle defined by the intervals and  $[0 M-1]$ ,  $[0 N-1]$ .



## 6. Some Properties of the 2-D DFT

- Symmetry

- An important result from functional analysis is that any **real or complex function**,  $w(x, y)$ , can be expressed as the sum of an **even** and an **odd** part (each of which can be real or complex):

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

- Where,

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

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

- It follows that

$$w_e(x, y) = w_e(-x, -y) \quad w_o(x, y) = -w_o(-x, -y)$$

---

## 6. Some Properties of the 2-D DFT

- Symmetry

- **Even** functions are said to be **symmetric** and **odd** functions are **antisymmetric**.
- When we talk about symmetry we are referring to symmetry about the center point of a sequence
- It is more convenient to think only in terms of **nonnegative indices**, in which case the definitions of evenness and oddness become

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

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

where  $M$  and  $N$  are the number of rows and columns of a 2-D array.

---

## 6. Some Properties of the 2-D DFT

- Symmetry
  - Although **evenness** and **oddness** are visualized easily for continuous functions, **these concepts are not as intuitive** when dealing with **discrete** sequences.
    - ✓ Even sequences
      - $f(x) = f(M - x)$
      - $f(0)$  is immaterial in the test for evenness.

## 6. Some Properties of the 2-D DFT

- Symmetry

- Although **evenness** and **oddness** are visualized easily for continuous functions, **these concepts are not as intuitive** when dealing with **discrete** sequences.
  - ✓ Even sequences
    - Example

$$f = \{f(0) \quad f(1) \quad f(2) \quad f(3)\} = \{2 \quad 1 \quad 1 \quad 1\}$$

$$M = 4 \quad f(x) = f(4 - x)$$

$$f(0) = f(4), \quad f(2) = f(2),$$

$$f(1) = f(3), \quad f(3) = f(1)$$

## 6. Some Properties of the 2-D DFT

- Symmetry
  - Although **evenness** and **oddness** are visualized easily for continuous functions, **these concepts are not as intuitive** when dealing with **discrete** sequences.
    - ✓ Odd sequences
      - $g(x) = -g(M - x)$
      - When **M** is **even**, a 1-D **odd** sequence has the property that the points at position **0** and **M/2** are **always zero**.
      - When **M** is **odd**, the first term still has to be **0**, but the **remaining** terms form pairs with **equal value** but **opposite sign**.

## 6. Some Properties of the 2-D DFT

- Symmetry

- Although **evenness** and **oddness** are visualized easily for continuous functions, **these concepts are not as intuitive** when dealing with **discrete** sequences.
  - ✓ Odd sequences
    - Example

$$g = \{g(0) \ g(1) \ g(2) \ g(3)\} = \{0 \ -1 \ 0 \ 1\}$$

- However, the sequence below is neither odd nor even, although the “basic” structure appears to be odd.

$$g = \{0 \ -1 \ 0 \ 1 \ 0\}$$

## 6. Some Properties of the 2-D DFT

- Symmetry
  - Although **evenness** and **oddness** are visualized easily for continuous functions, **these concepts are not as intuitive** when dealing with **discrete** sequences.
    - ✓ The same basic considerations hold in 2-D.

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

- ✓ As an **exercise**, you use the previous equations to convince yourself that this 2-D sequence is **odd**.

## 6. Some Properties of the 2-D DFT

- Symmetry properties

- Now, a number of important symmetry properties of the DFT and its inverse can be establish.
- A property used frequently is that the Fourier transform of a **real** function,  $f(x, y)$ , is **conjugate symmetric**,

$$F^*(u, v) = F(-u, -v)$$

- If  $f(x, y)$  is **imaginary**, its Fourier transform is **conjugate antisymmetric**,

$$F^*(-u, -v) = -F(u, v)$$

## 6. Some Properties of the 2-D DFT

- Symmetry properties

- Now, a number of important symmetry properties of the DFT and its inverse can be establish.

	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

**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. The term *complex* indicates that a function has nonzero real and imaginary parts.

<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.

## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.
  - Because the 2-D DFT is complex in general, it can be expressed in polar form:

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

where

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

is the **frequency spectrum** and

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

is the **phase angle**.

---

## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.

➤ The **power spectrum** is defined as

$$\begin{aligned} P(u, v) &= |F(u, v)|^2 \\ &= R^2(u, v) + I^2(u, v) \end{aligned}$$

➤ The Fourier transform of a **real** function is **conjugate symmetric**, which implies that the spectrum has **even symmetry** about the origin:

$$|F(u, v)| = |F(-u, -v)|$$

➤ The phase angle exhibits the following **odd symmetry** about the origin:

$$\phi(u, v) = -\phi(-u, -v)$$

## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.

➤ Finally, it follows from

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

that

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

which indicates that the **zero-frequency term** is **proportional** to the **average value** of  $f(x, y)$ .

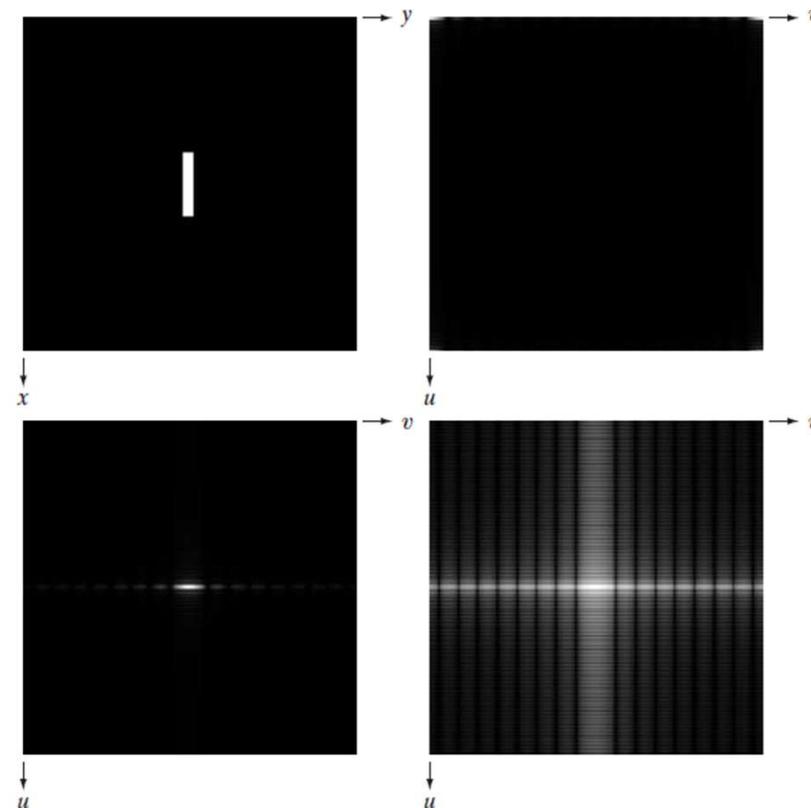
$$\begin{aligned} F(0, 0) &= MN \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \\ &= M\bar{f}(x, y) \end{aligned}$$

---

## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.

➤ Example (spectrum)



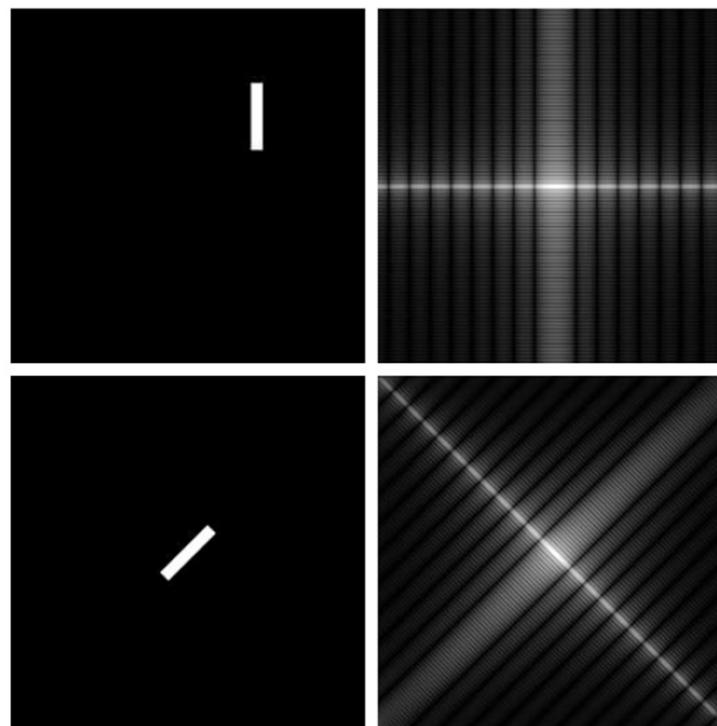
a  
b  
c  
d

**FIGURE 4.24**  
(a) Image.  
(b) Spectrum  
showing bright spots  
in the four corners.  
(c) Centered  
spectrum. (d) Result  
showing increased  
detail after a log  
transformation. The  
zero crossings of the  
spectrum are closer in  
the vertical direction  
because the rectangle  
in (a) is longer in that  
direction. The  
coordinate  
convention used  
throughout the book  
places the origin of  
the spatial and  
frequency domains at  
the top left.

## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.

- Example (spectrum)

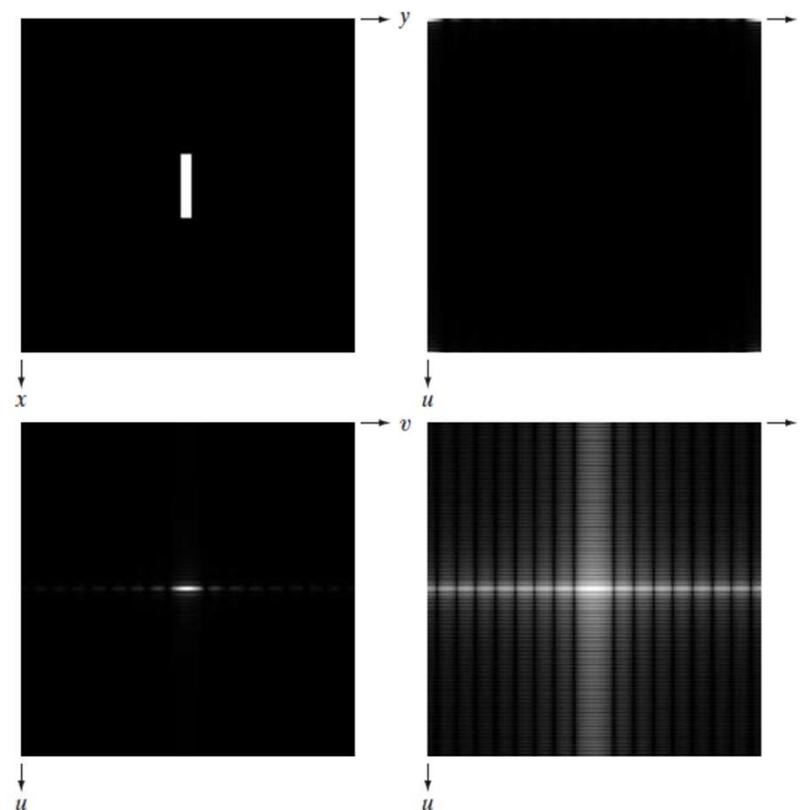


a	b
c	d

**FIGURE 4.25**  
(a) The rectangle in Fig. 4.24(a) translated, and (b) the corresponding spectrum. (c) Rotated rectangle, and (d) the corresponding spectrum. The spectrum corresponding to the translated rectangle is identical to the spectrum corresponding to the original image in Fig. 4.24(a).

## 6. Some Properties of the 2-D DFT

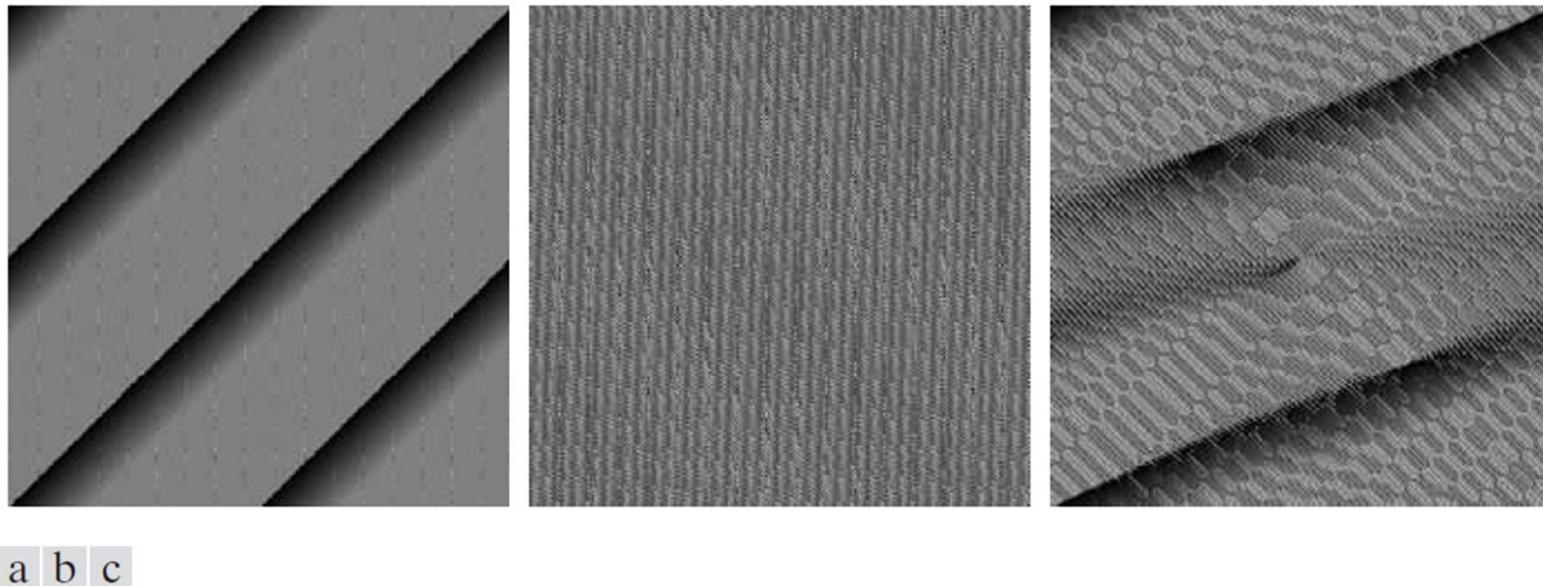
- MATLAB: s113Spectrum.m



## 6. Some Properties of the 2-D DFT

- Fourier Spectrum and Phase Angle.

- Example (phase angle)



a b c

**FIGURE 4.26** Phase angle array corresponding (a) to the image of the centered rectangle in Fig. 4.24(a), (b) to the translated image in Fig. 4.25(a), and (c) to the rotated image in Fig. 4.25(c).

## 6. Some Properties of the 2-D DFT

- The 2-D Convolution Theorem

$$f(x, y) \star h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)h(x - m, y - n)$$
$$x = 0, 1, 2, \dots, M - 1 \quad \text{and} \quad y = 0, 1, 2, \dots, N - 1$$

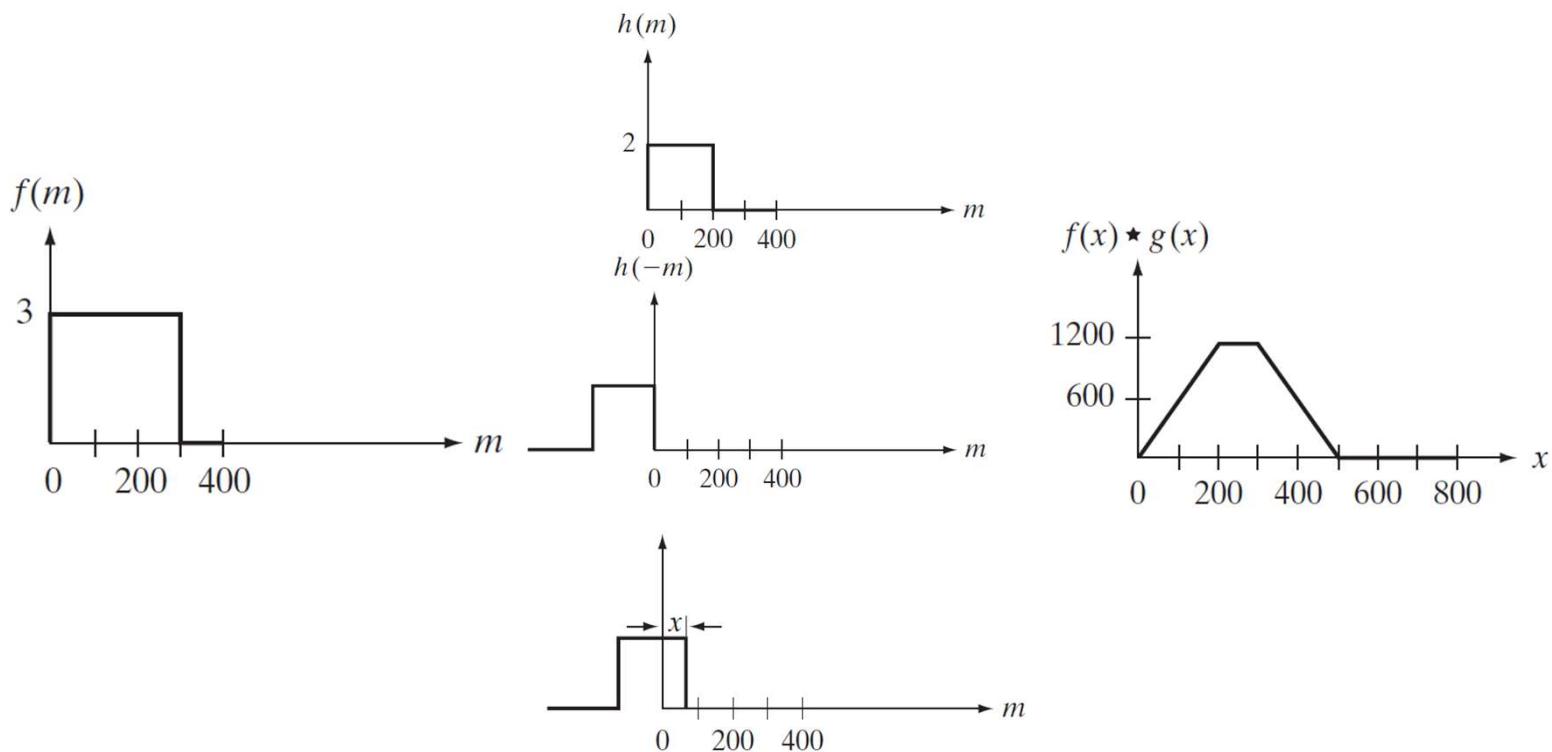
$$f(x, y) \star h(x, y) \Leftrightarrow F(u, v)H(u, v)$$

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

## 6. Some Properties of the 2-D DFT

- The 2-D Convolution Theorem

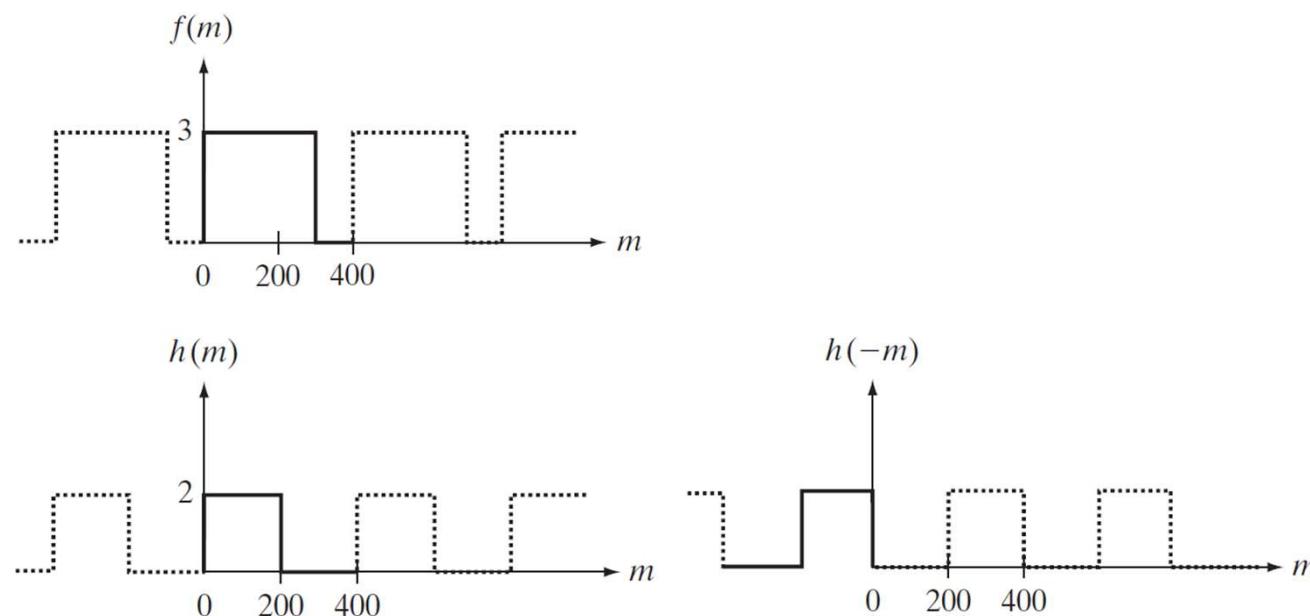
➤ 1-D example: convolution of two functions,  $f$  and  $h$ .



## 6. Some Properties of the 2-D DFT

- The 2-D Convolution Theorem

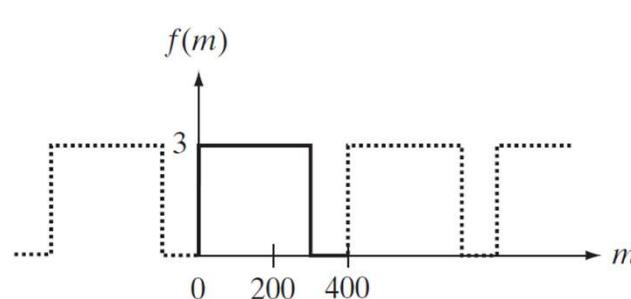
➤ 1-D example: if we use the DFT and the convolution theorem to obtain the same result, we must take into account the **periodicity inherent in the expression for the DFT**.



## 6. Some Properties of the 2-D DFT

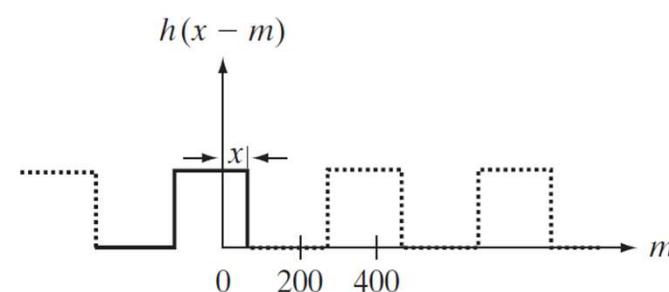
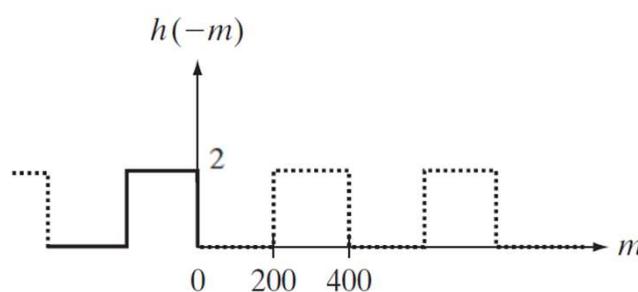
- The 2-D Convolution Theorem

➤ 1-D example: if we use the DFT and the convolution theorem to obtain the same result, we must take into account the periodicity inherent in the expression for the DFT.



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

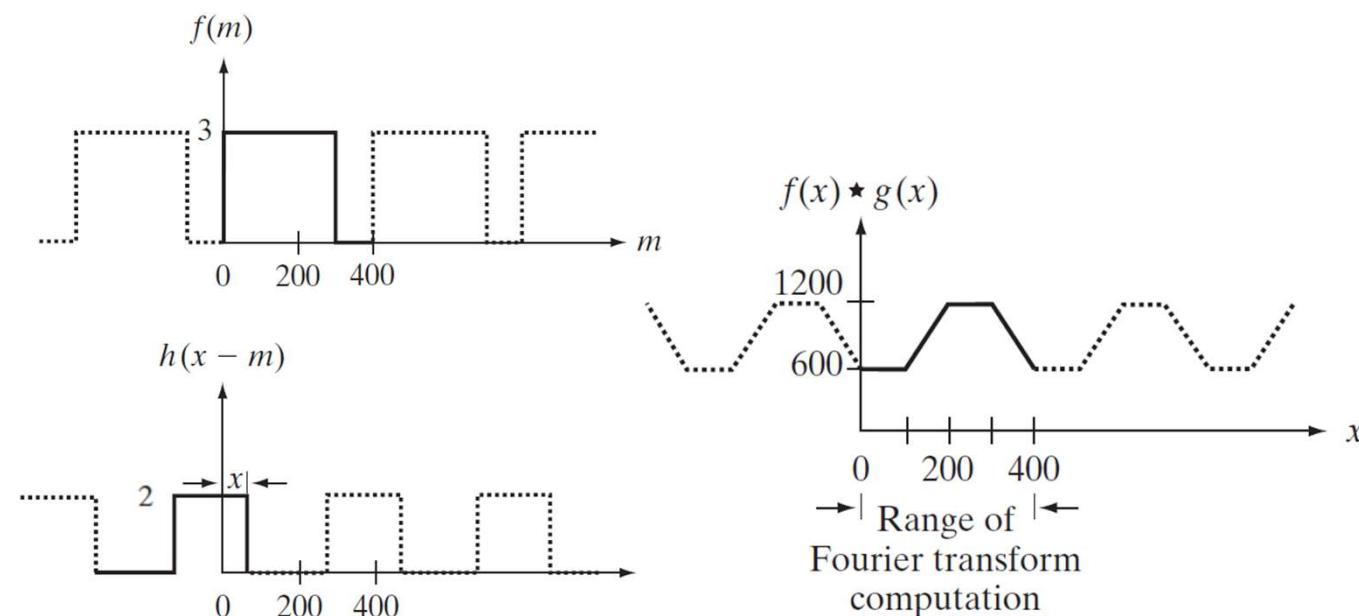
$$x = 0, 1, 2, \dots, M-1$$



## 6. Some Properties of the 2-D DFT

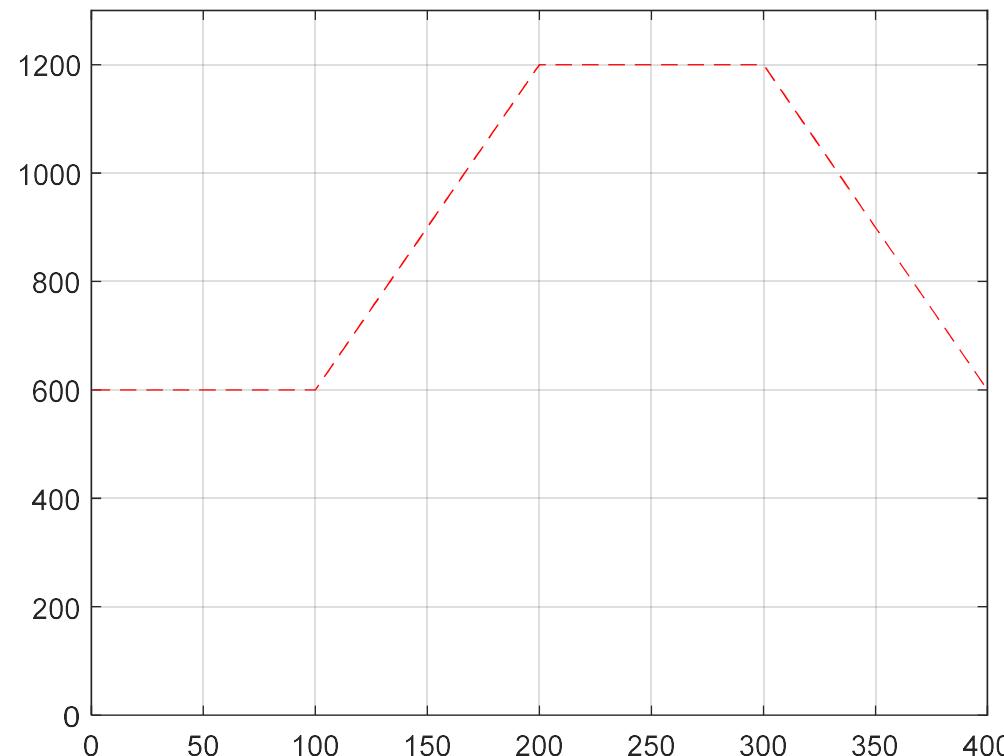
- The 2-D Convolution Theorem

➤ 1-D example: if we use the DFT and the convolution theorem to obtain the same result, we must take into account the periodicity inherent in the expression for the DFT.



## 6. Some Properties of the 2-D DFT

- MATLAB: s120ConvolutionPadding.m



## 6. Some Properties of the 2-D DFT

- The 2-D Convolution Theorem
  - The **closeness of the periods** is such that they interfere with each other to cause what is commonly referred to as **wraparound error**.
  - Fortunately, the solution to the wraparound error problem is simple.
  - Consider two functions,  $f(x)$  and  $h(x)$  composed of **A and B samples**, respectively.
  - We need to append  $P$  zeros to both functions.

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

## 6. Some Properties of the 2-D DFT

- The 2-D Convolution Theorem
  - Visualizing a similar example in 2-D would be more difficult, but the conclusions are similar.
  - 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 circular convolution can be avoided by **padding** these functions with zeros, as follows

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

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

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

---

## 7. The Basics of Filtering in the Frequency Domain

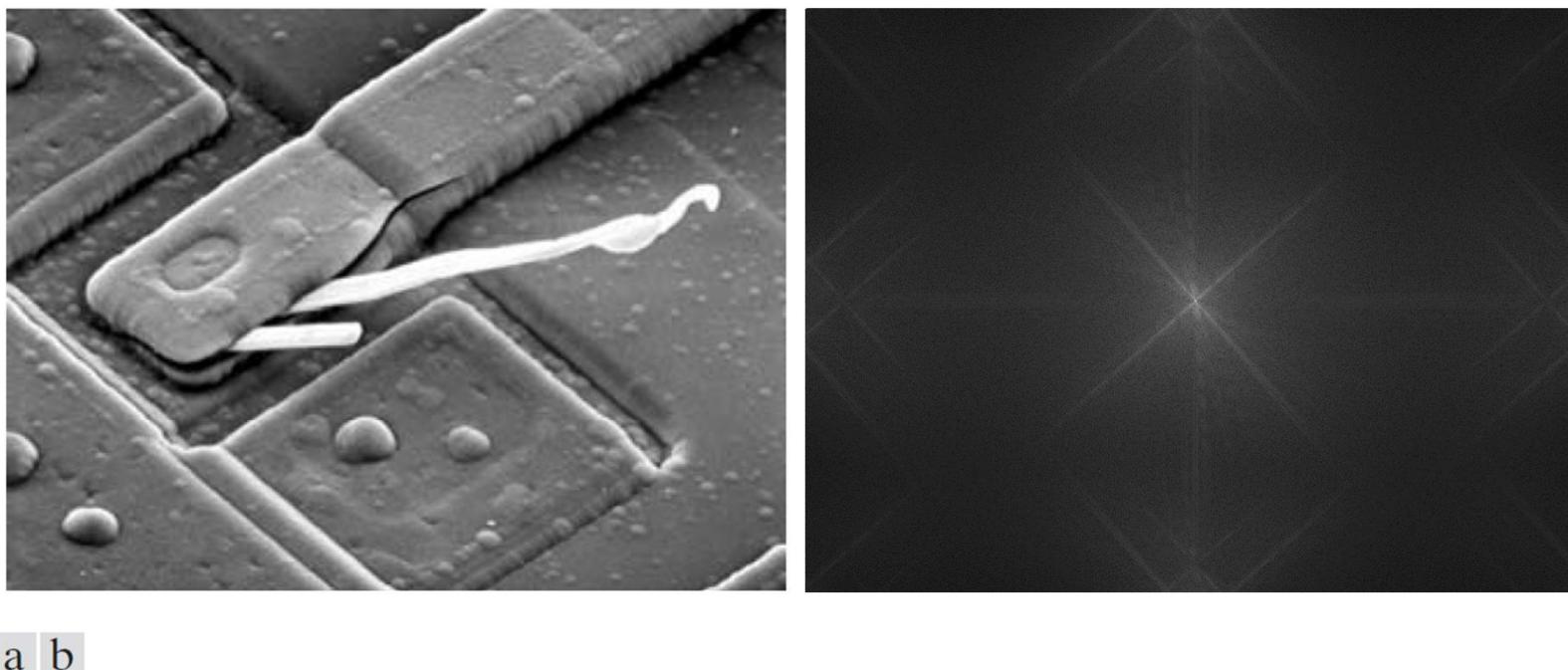
- Additional Characteristics of the Frequency Domain
  - Filtering techniques in the frequency domain are based on modifying the DFT of a signal to achieve a specific objective.
  - **Frequency** is directly related to **spatial rates of change**.
    - ✓ The **slowest** varying frequency component is proportional to the **average intensity** of an image.
    - ✓ As we move **away from the origin** of the shifted transform, the high frequencies correspond to the **fast varying intensity** components of an image.

## 7. The Basics of Filtering in the Frequency Domain

- Additional Characteristics of the Frequency Domain
  - The two components of the transform to which we have access are:
    - ✓ The transform **magnitude** (spectrum); and
    - ✓ The phase **angle**.
  - Visual analysis of the **phase** component generally is **not very useful**.
  - The **spectrum**, however, provides some **useful visual information**.

## 7. The Basics of Filtering in the Frequency Domain

- Additional Characteristics of the Frequency Domain

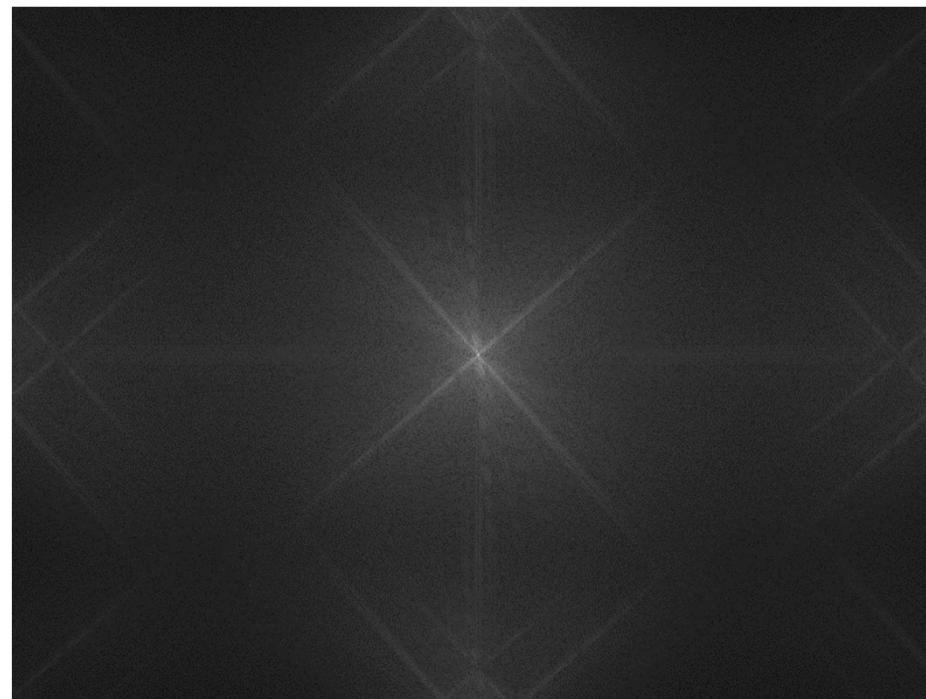


a | b

**FIGURE 4.29** (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.)

## 7. The Basics of Filtering in the Frequency Domain

- MATLAB: s125Spectrum.m

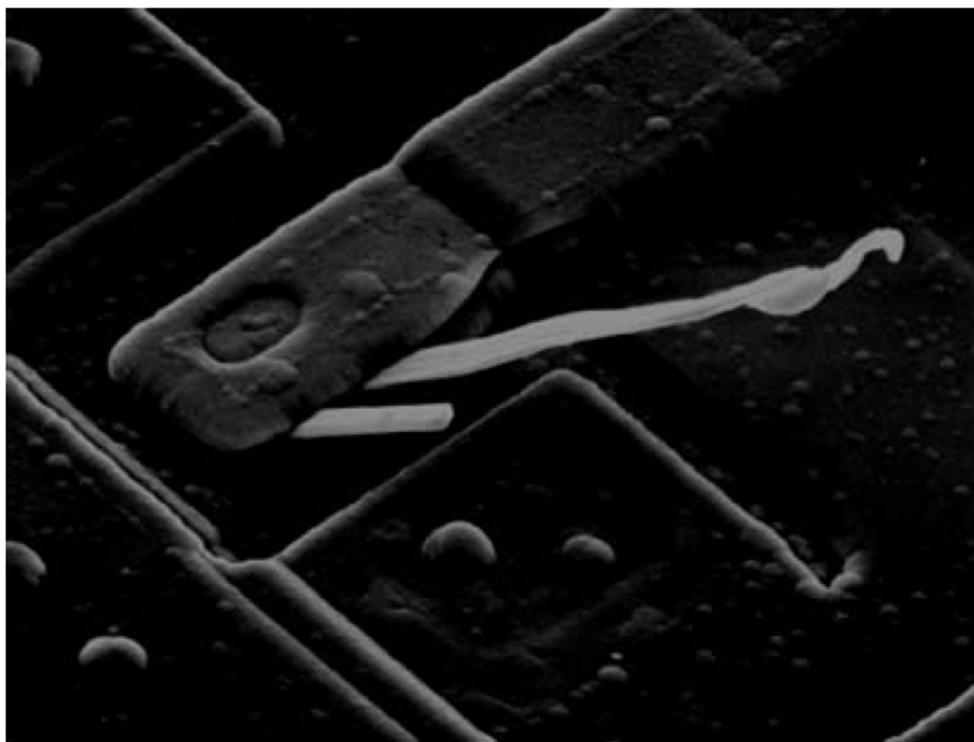


## 7. The Basics of Filtering in the Frequency Domain

- Frequency Domain Filtering Fundamentals
  - Given a digital image  $f(x,y)$  of  $M \times N$  pixels, the basic filtering equation in which we are interested has the form:
$$g(x, y) = \mathfrak{F}^{-1}[H(u, v)F(u, v)]$$
  - $F(u,v)$  is the DFT of the input image,  $f(x, y)$ , and  $H(u,v)$  is the filter;  $g(x, y)$  is the filtered image.
  - We are now in a position to consider the filtering process in some detail.
  - One of the simplest filters we can construct is a filter  $H(u, v)$  that is 0 at the center of the transform and 1 elsewhere.

## 7. The Basics of Filtering in the Frequency Domain

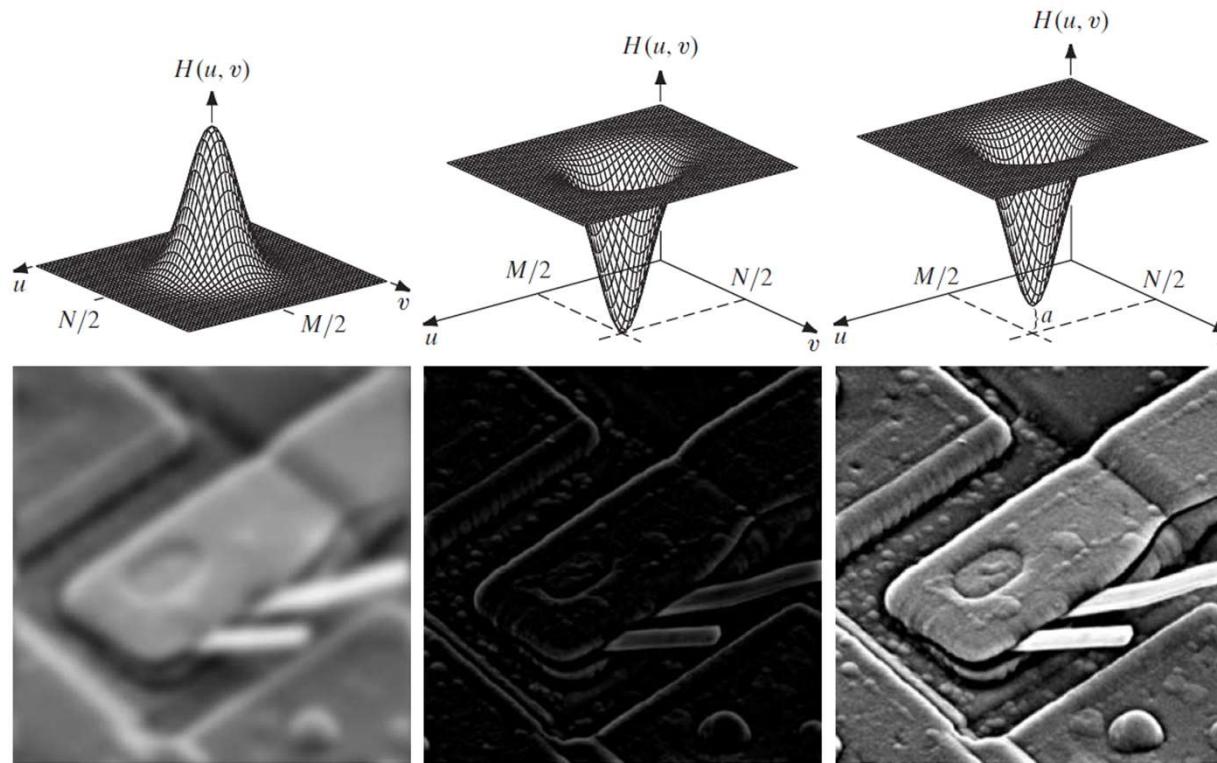
- Frequency Domain Filtering Fundamentals



**FIGURE 4.30**  
Result of filtering the image in Fig. 4.29(a) by setting to 0 the term  $F(M/2, N/2)$  in the Fourier transform.

## 7. The Basics of Filtering in the Frequency Domain

- Frequency Domain Filtering Fundamentals

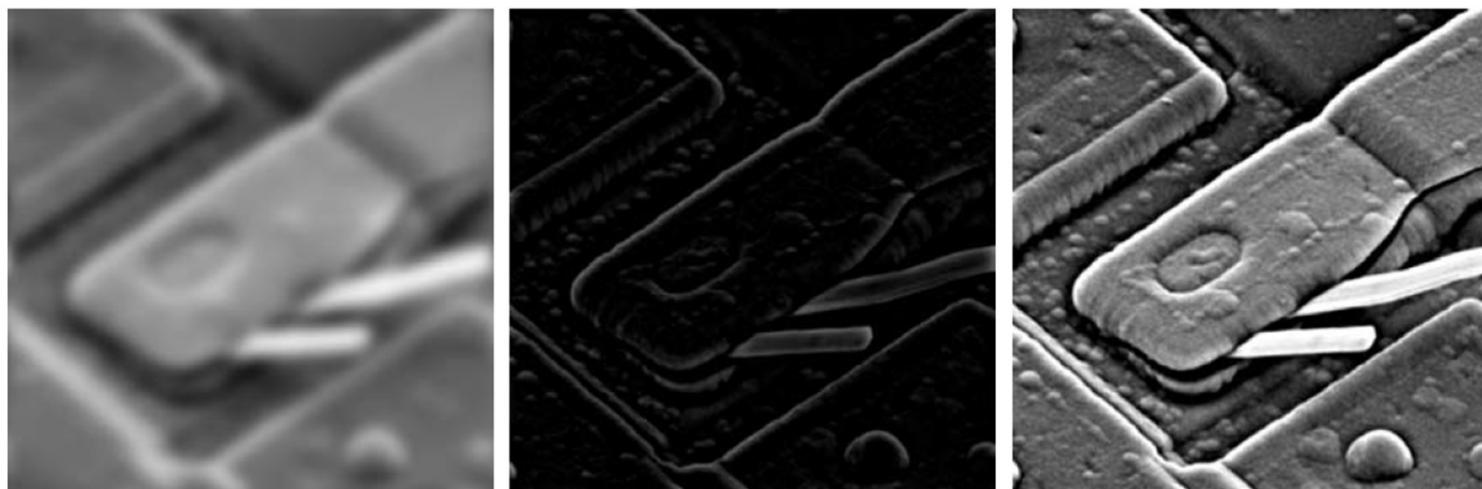


a	b	c
d	e	f

**FIGURE 4.31** Top row: frequency domain filters. Bottom row: corresponding filtered images obtained using Eq. (4.7-1). We used  $a = 0.85$  in (c) to obtain (f) (the height of the filter itself is 1). Compare (f) with Fig. 4.29(a).

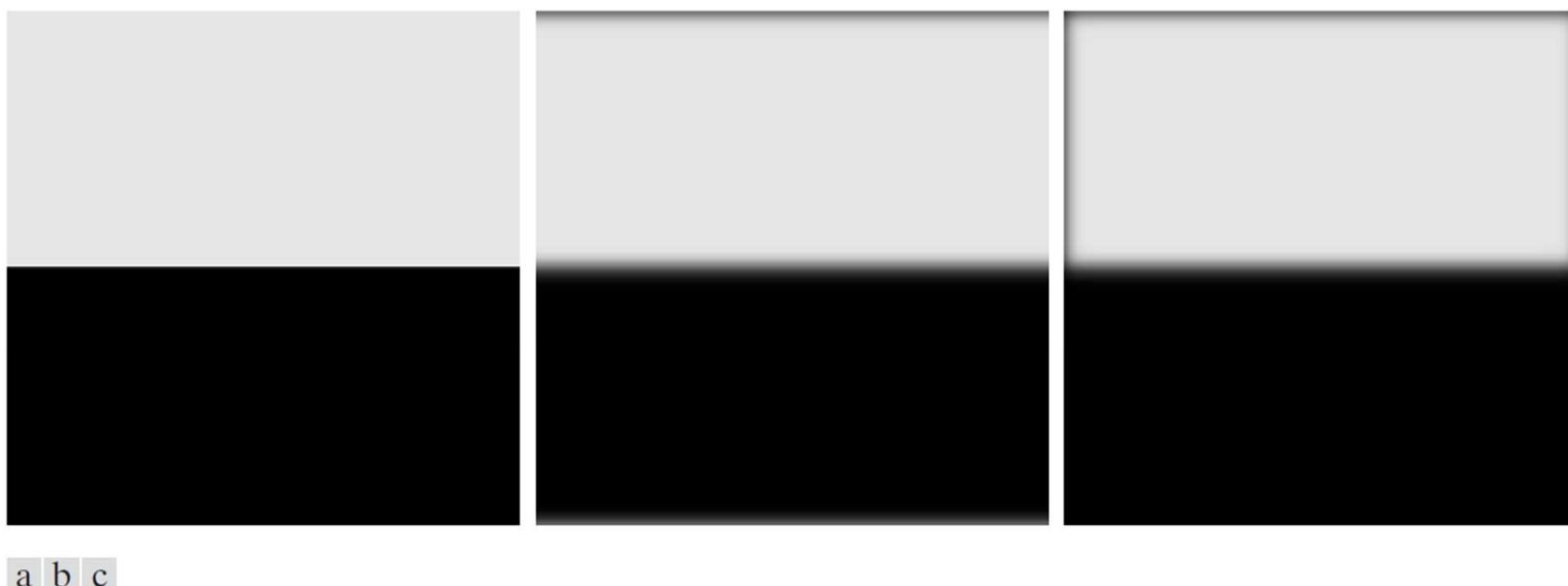
## 7. The Basics of Filtering in the Frequency Domain

- MATLAB: s130FilterExamples.m



## 7. The Basics of Filtering in the Frequency Domain

- Frequency Domain Filtering Fundamentals

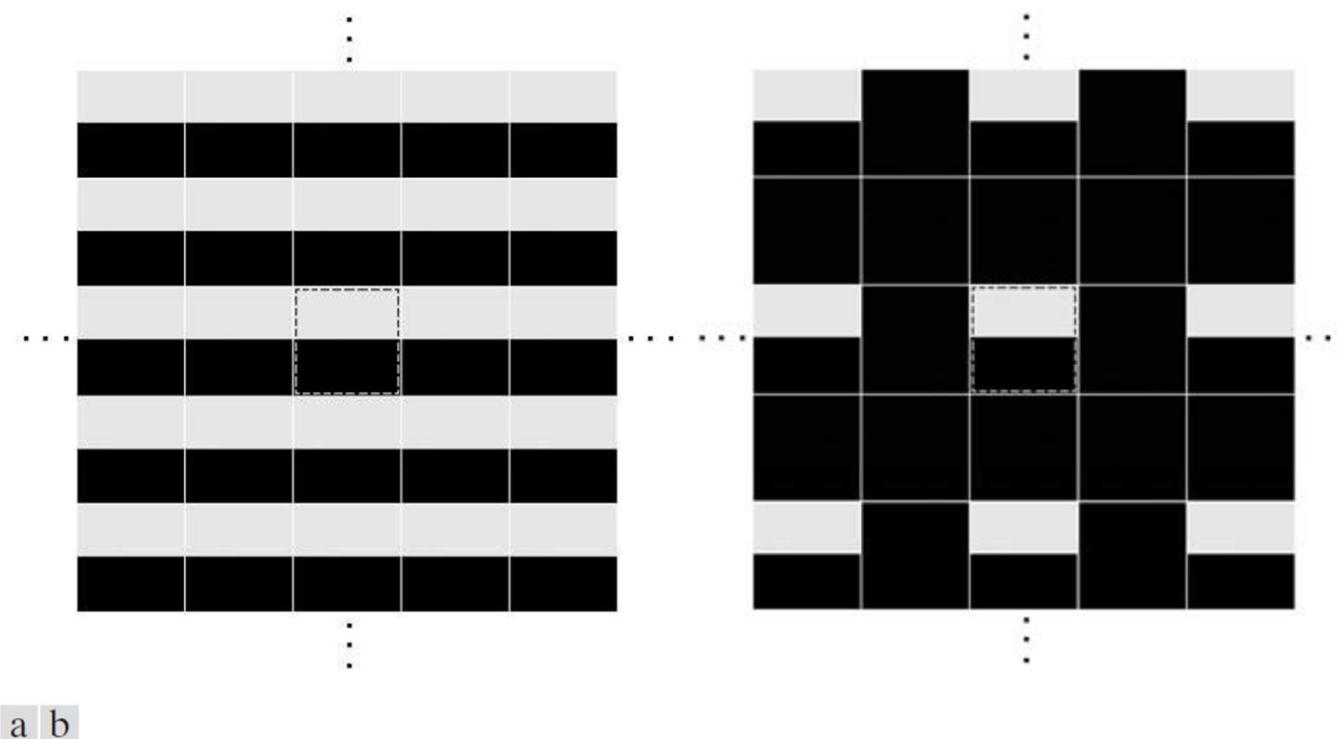


a b c

**FIGURE 4.32** (a) A simple image. (b) Result of blurring with a Gaussian lowpass filter without padding. (c) Result of lowpass filtering with padding. Compare the light area of the vertical edges in (b) and (c).

## 7. The Basics of Filtering in the Frequency Domain

- Frequency Domain Filtering Fundamentals



a

b

**FIGURE 4.33** 2-D image periodicity inherent in using the DFT. (a) Periodicity without image padding. (b) Periodicity after padding with 0s (black). The dashed areas in the center correspond to the image in Fig. 4.32(a). (The thin white lines in both images are superimposed for clarity; they are not part of the data.)

## 7. The Basics of Filtering in the Frequency Domain

- MATLAB: s129CircConv.m



## 7. The Basics of Filtering in the Frequency Domain

- Frequency Domain Filtering Fundamentals
  - **Low frequencies** in the transform are related to **slowly varying intensity** components in an image.
  - **High frequencies** are caused by **sharp transitions in intensity**
  - A filter that attenuates high frequencies while passing low frequencies (appropriately called a **lowpass filter**) would blur an image.
  - A filter with the opposite property (called a **highpass filter**) would enhance sharp detail.

## 7. The Basics of Filtering in the Frequency Domain

- Summary of Steps for Filtering in the Frequency Domain

- I. Given an input image  $f(x, y)$  of size  $M \times N$  obtain the padding parameters  $P$  and  $Q$  from  $P = 2M$  e  $Q = 2N$ .
- I. Form a padded image  $f_p(x, y)$  of size  $P \times Q$ .
- II. Multiply  $f_p(x, y)$  by  $(-1)^{x+y}$  to center its transform  
 $\rightarrow f_{ps}(x, y)$
- III. Compute DFT of  $f_{ps}(x, y) \rightarrow F_{ps}(u, v)$ .
- IV. Generate a real, symmetric filter function  $H(u, v)$  of size  $P \times Q$ , with center at coordinates  $P/2, Q/2$ .

## 7. The Basics of Filtering in the Frequency Domain

- Summary of Steps for Filtering in the Frequency Domain

V. Calculate  $G(u, v) = H(u, v).F_{ps}(u, v)$ .

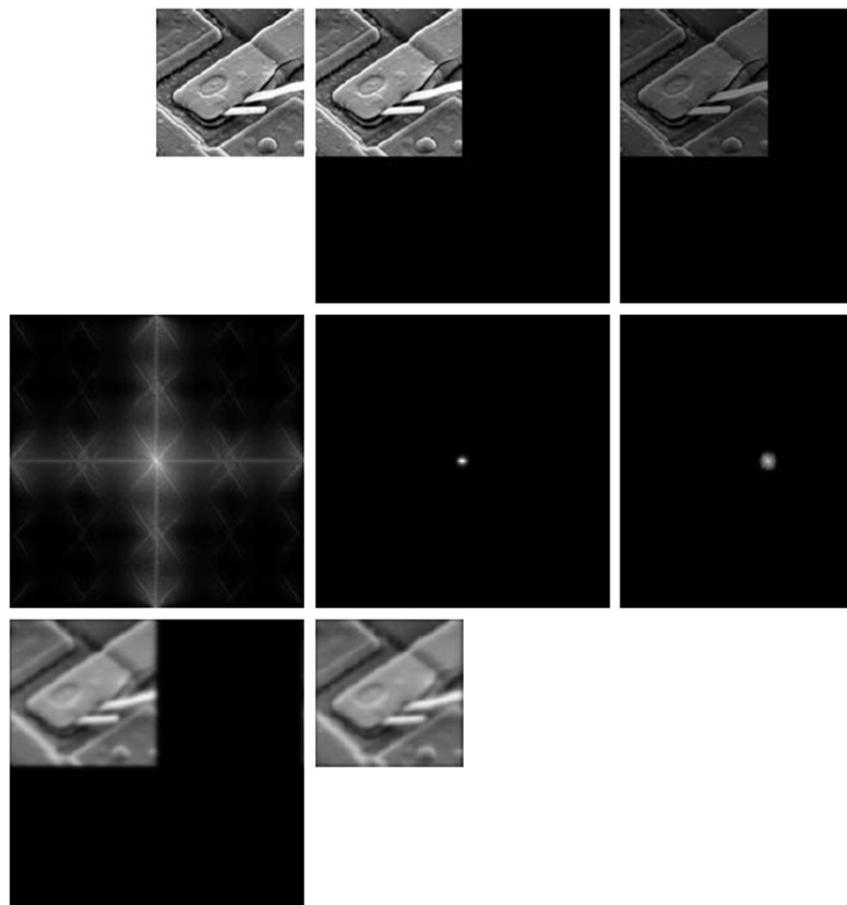
VI. Obtain the processed image:

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

VII. Obtain the final processed result  $g(x, y)$  by extracting the  $M \times N$  region from the top, left quadrant  $g_p(x, y)$ .

## 7. The Basics of Filtering in the Frequency Domain

- Summary of Steps for Filtering in the Frequency Domain

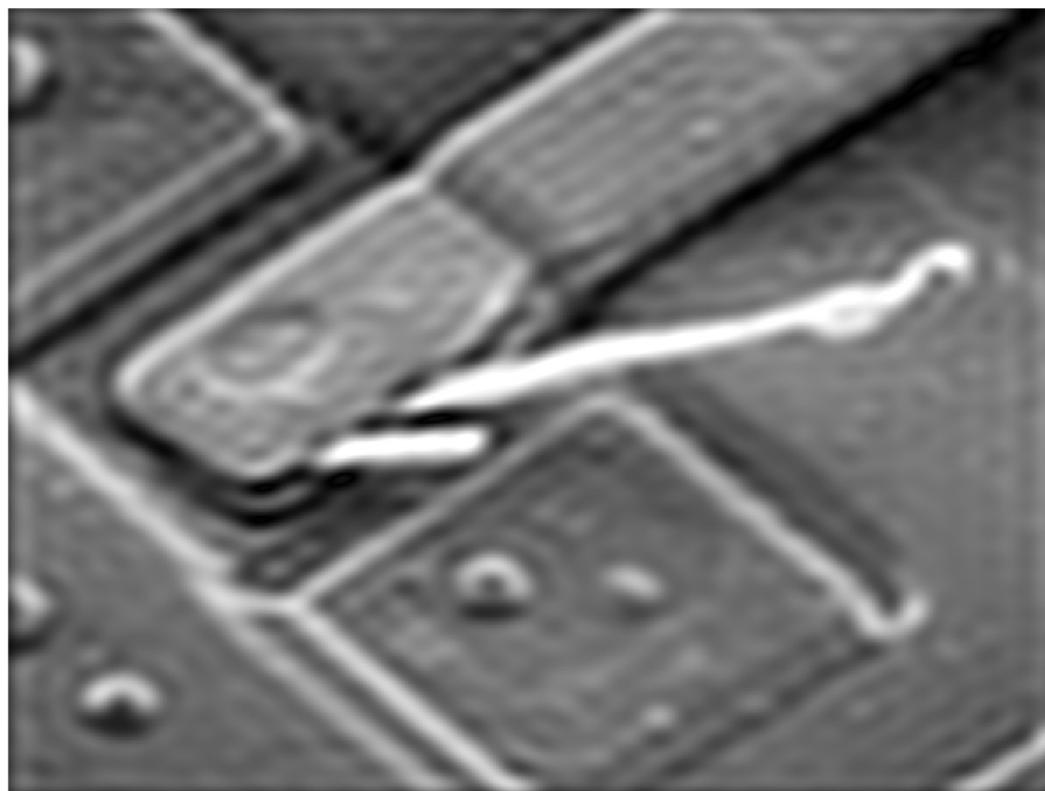


a b c  
d e f  
g h

**FIGURE 4.36**  
(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  $F_p$ . (e) Centered Gaussian lowpass filter,  $H$ , of size  $P \times Q$ .  
(f) Spectrum of the product  $HF_p$ .  
(g)  $g_p$ , the product of  $(-1)^{x+y}$  and the real part of the IDFT of  $HF_p$ .  
(h) Final result,  $g$ , obtained by cropping the first  $M$  rows and  $N$  columns of  $g_p$ .

## 7. The Basics of Filtering in the Frequency Domain

- MATLAB: s138FilterAlg.m



## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - The link between filtering in the spatial and frequency domains is the convolution theorem.
  - Given a filter  $H(u, v)$ , suppose that we want to find its equivalent representation in the spatial domain.
  - If we let  $f(x, y) = \delta(x, y)$ , then  $F(u, v) = 1$ .
  - The filtered output is

$$\mathcal{F}^{-1}\{F(u, v).H(u, v)\} = \mathcal{F}^{-1}\{H(u, v)\}$$

which is the corresponding filter in the spatial domain,  
 $h(x, y)$ .

---

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - Because this filter can be obtained from the response of a frequency domain filter to an impulse,  $h(x, y)$  sometimes is referred to as the **impulse response of  $H(u, v)$** .
  - And because the impulse response is of finite length, such filters are called **finite impulse response** (FIR) filters.
  - These are the **only types** of linear spatial filters **considered in this course**.

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - In practice, **we prefer** to implement **convolution** filtering using **small filter masks**.
  - However, filtering concepts are **more intuitive** in the **frequency domain**.
  - One way to take advantage of the properties of both domains is:
    1. To specify a filter in the frequency domain;
    2. Compute its IDFT; and
    3. Use the resulting, full-size spatial filter as a guide for constructing smaller spatial filter.

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

➤ Example (lowpass filter):

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

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

- ✓ The corresponding filter in the spatial domain is obtained by taking the inverse Fourier transform of  $H(u)$ :

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

- ✓ When  $H(u)$  has a broad profile (large value of  $s$ ),  $h(x)$  has a narrow profile, and vice versa.

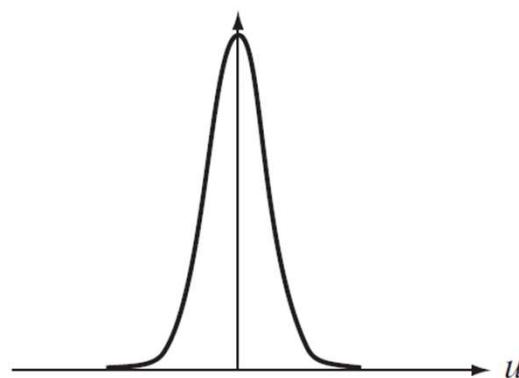
## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

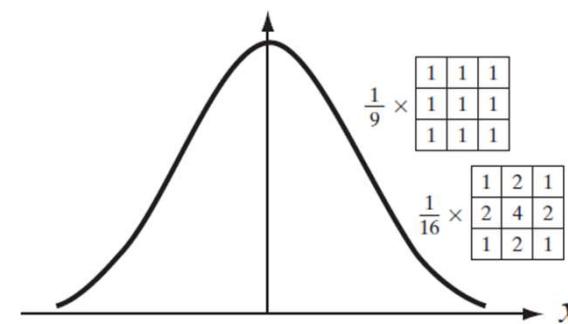
➤ Example (lowpass filter):

- ✓ Suppose that we want to use the shape of  $h(x)$  as a guide for specifying the coefficients of a small spatial mask

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



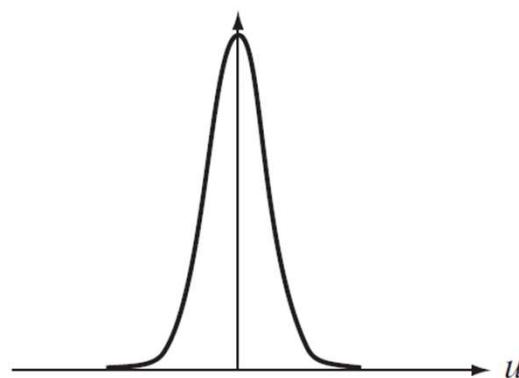
$$h(x) = \sqrt{2\pi}\sigma A e^{-2\pi^2\sigma^2 x^2}$$



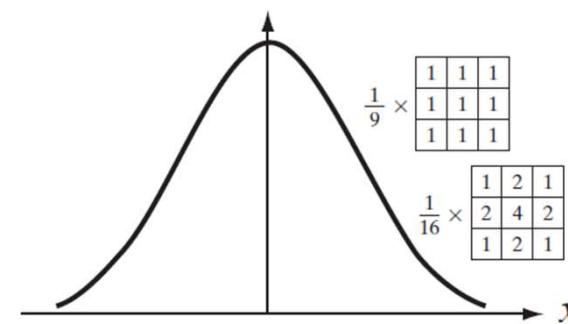
## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - Example (lowpass filter):
    - ✓ All coefficients are positive.

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



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



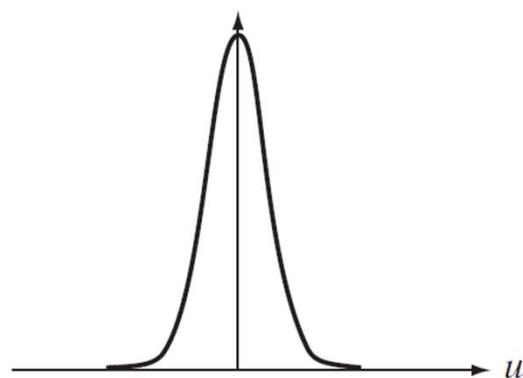
## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

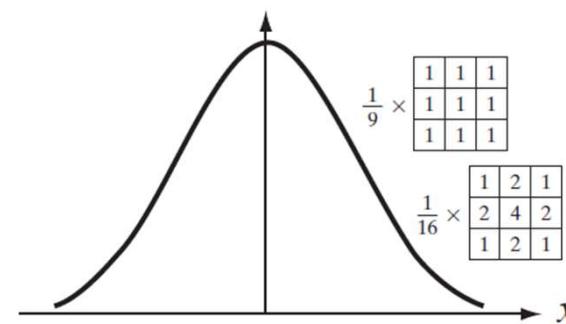
➤ Example (lowpass filter):

- ✓ The **narrower** the **frequency** domain **filter**, the more it will attenuate the low frequencies, resulting in increased blurring.

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



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



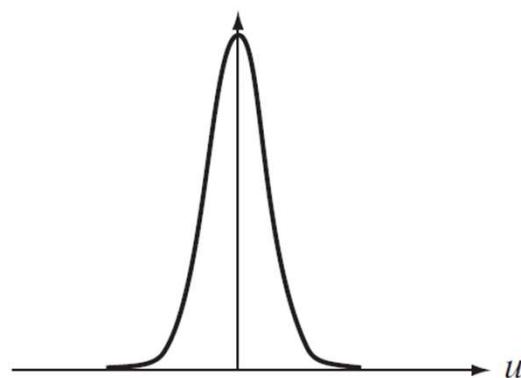
## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

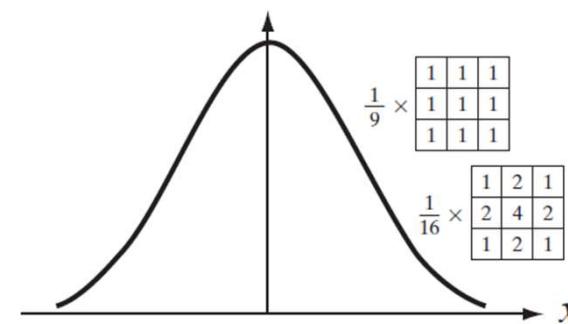
➤ Example (lowpass filter):

- ✓ In the **spatial** domain, this means that a **larger mask** must be used to **increase blurring**.

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



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



## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

➤ Example (highpass filter):

- ✓ Difference of Gaussians in the frequency domain:

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

$$A \geq B \text{ and } \sigma_1 > \sigma_2$$

- ✓ The corresponding filter in the spatial domain is

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

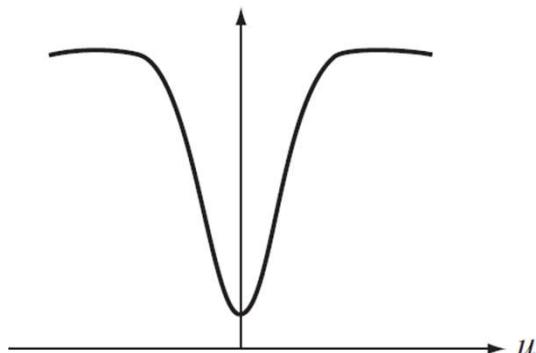
## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

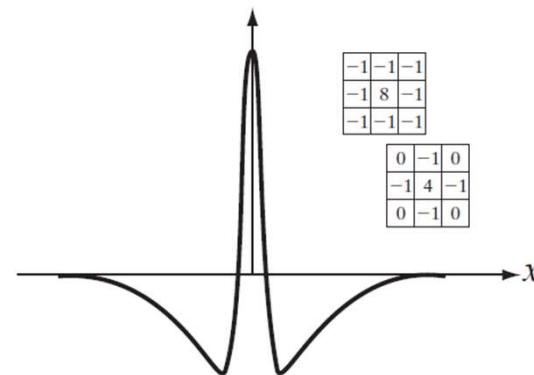
➤ Example (highpass filter):

- Has a positive center term with negative terms on either side.

$$H(u) = A e^{-u^2/2\sigma_1^2} - B e^{-u^2/2\sigma_2^2}$$



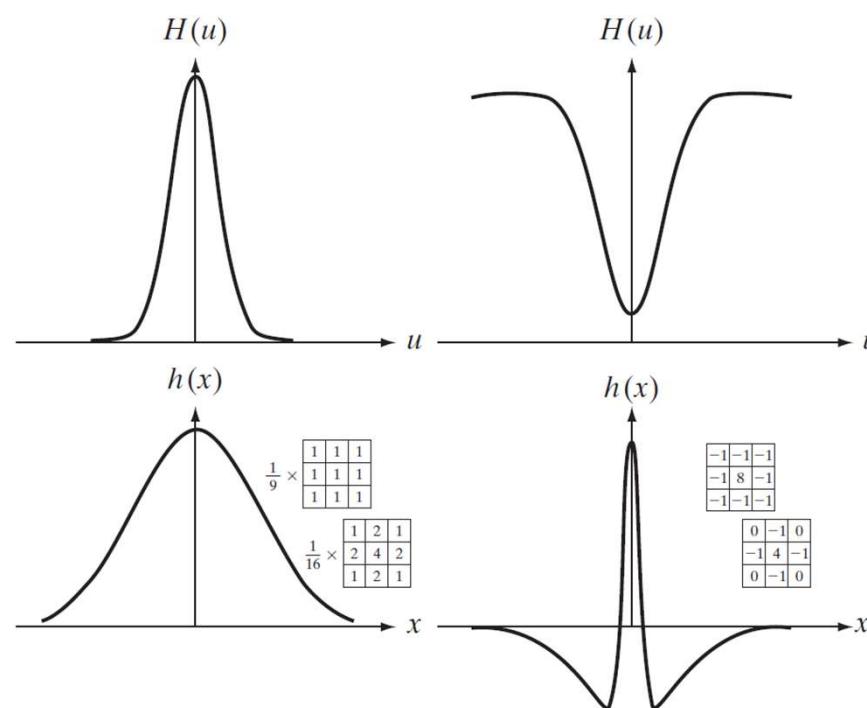
$$h(x) = \sqrt{2\pi}\sigma_1 A e^{-2\pi^2\sigma_1^2x^2} - \sqrt{2\pi}\sigma_2 B e^{-2\pi^2\sigma_2^2x^2}$$



## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains

➤ Example (lowpass and highpass filter):



**FIGURE 4.37**

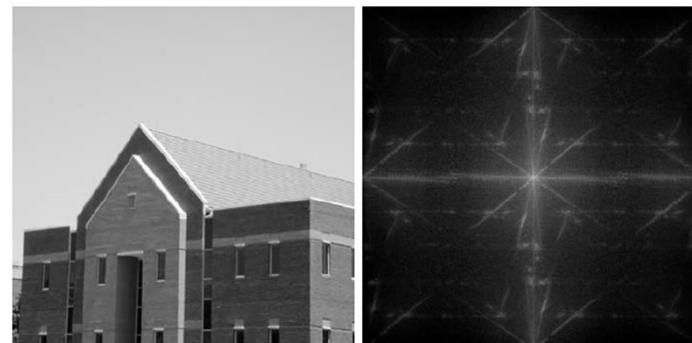
(a) A 1-D Gaussian lowpass filter in the frequency domain.  
 (b) Spatial lowpass filter corresponding to (a).  
 (c) Gaussian highpass filter in the frequency domain.  
 (d) Spatial highpass filter corresponding to (c). The small 2-D masks shown are spatial filters we used in Chapter 3.

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - Frequency domain can be viewed as a “laboratory”.
  - Some tasks that would be difficult to formulate in the spatial domain become easier in the frequency domain.
  - Once we have selected a filter in the frequency domain, the implementation is usually done in the spatial domain.
  - One approach is to specify small spatial masks that attempt to capture the “essence” of the full filter function in the spatial domain.
  - There are more formal approaches to design 2-D digital filters.

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - In this example, we start with a spatial mask and show how to generate the filter in the frequency domain.
  - To keep matters simple, we use the 3x3 Sobel vertical edge detector.
  - The figure below shows a pixel image that we wish to filter and its spectrum.



a b

**FIGURE 4.38**  
(a) Image of a  
building, and  
(b) its spectrum.

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains
  - Figure 4.39(a) shows the Sobel mask,  $h(x, y)$ .
  - Because the input image is of size  $600 \times 600$  pixels and the filter is of size  $3 \times 3$  we avoid wraparound error by padding  $f$  and  $h$  to size  $602 \times 602$  pixels.
  - The Sobel mask exhibits odd symmetry. To maintain this symmetry, we place  $h(x, y)$  so that its center is at the center of the padded array forming  $h_p(x, y)$ .

## 7. The Basics of Filtering in the Frequency Domain

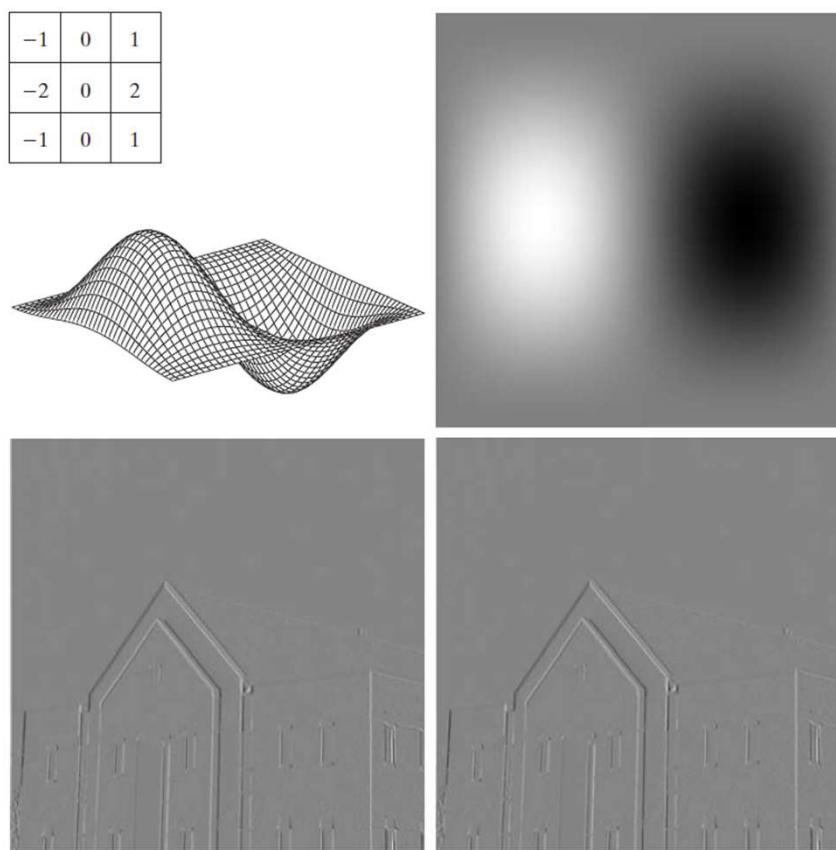
- Correspondence Between Filtering in the Spatial and Frequency Domains

➤ The procedure used to generate  $H(u, v)$  is:

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 1;
3. Set the real part of the resulting DFT to 0 to account for parasitic real parts -  $H(x, y)$  has to be purely imaginary; and
4. Multiply the result by  $(-1)^{u+v}$  (reverses the multiplication of  $H(u, v)$  by  $(-1)^{u+v}$  which is implicit when  $h(x, y)$  was moved to the center of  $h_p(x, y)$ ).

## 7. The Basics of Filtering in the Frequency Domain

- Correspondence Between Filtering in the Spatial and Frequency Domains



a b  
c d

**FIGURE 4.39**  
(a) A spatial mask and perspective plot of its corresponding frequency domain filter. (b) Filter shown as an image. (c) Result of filtering Fig. 4.38(a) in the frequency domain with the filter in (b). (d) Result of filtering the same image with the spatial filter in (a). The results are identical.

## 7. The Basics of Filtering in the Frequency Domain

- MATLAB: s153FilterGen.m

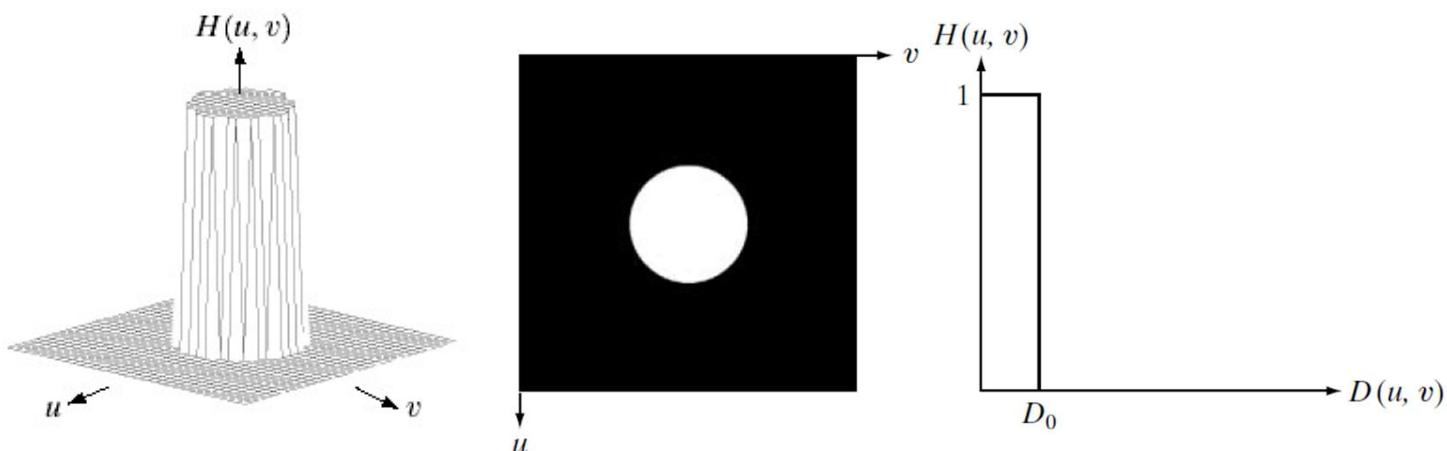


## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

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

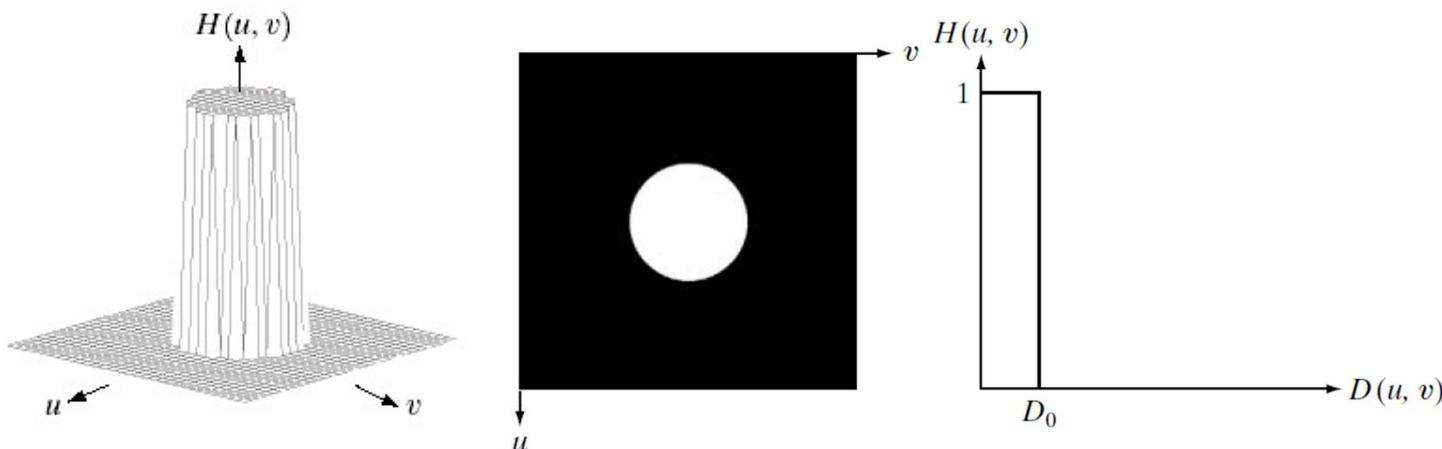
$$D(u, v) = [(u - P/2)^2 + (v - Q/2)^2]^{1/2}$$



## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

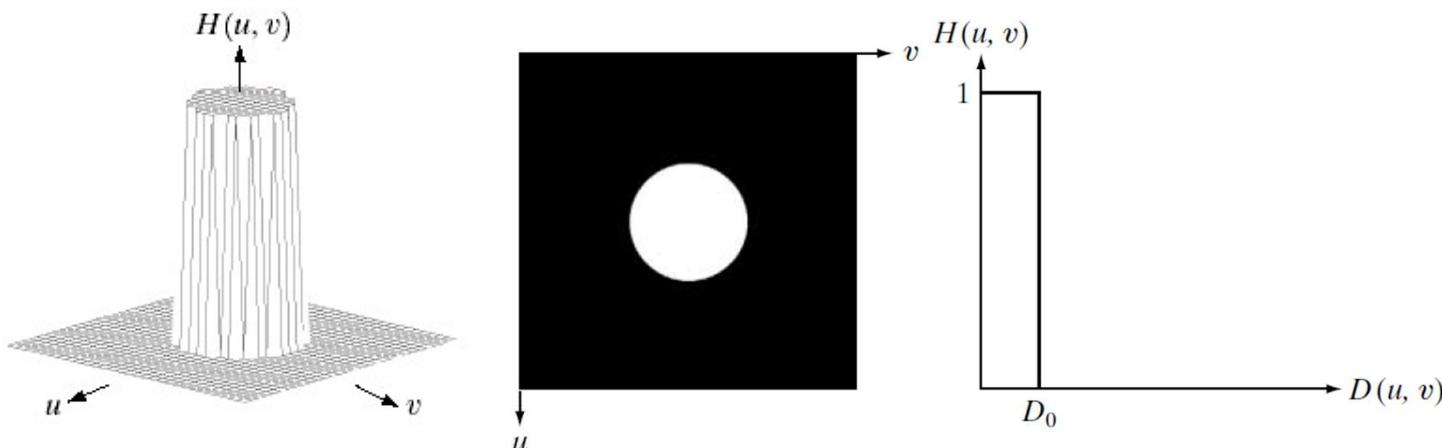
- The name **ideal** indicates that all frequencies on or inside a circle of radius  $D_0$  are passed, without attenuation.
- For an ILPF cross section, the point  $D_0$  of transition between 1 and 0 is called the *cutoff frequency*.



## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

- The sharp cutoff frequencies of an ILPF cannot be realized with electronic components, although they certainly can be simulated in a computer.
- The lowpass filters introduced in this chapter are compared by studying their behavior as a function of the same cutoff frequencies.



## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

- One way to establish a set of standard cutoff frequency loci is to compute circles that enclose specified amounts of **total image power**  $P_T$ .
  - This quantity is obtained by summing the components of the power spectrum of the padded images at each point

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

$$\begin{aligned} P(u, v) &= |F(u, v)|^2 \\ &= R^2(u, v) + I^2(u, v) \end{aligned}$$

## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

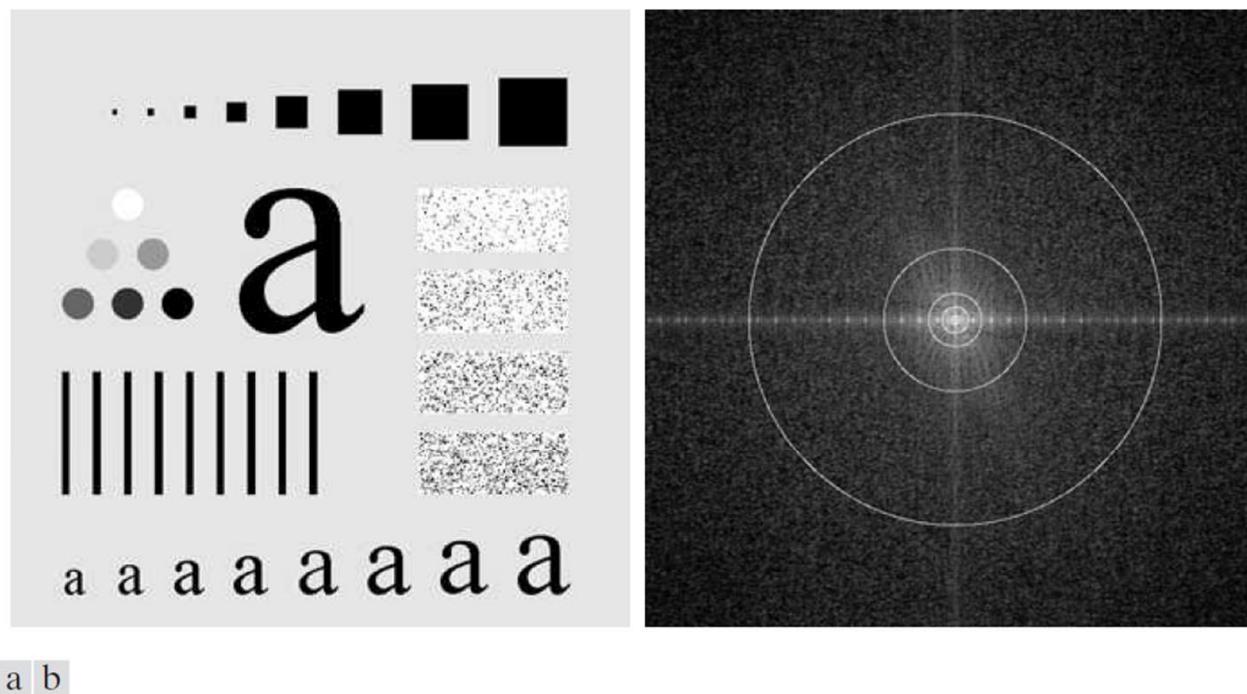
- If the DFT has been centered, a circle of radius  $D_0$  with origin at the center of the frequency rectangle encloses  $\alpha$  percent of the power, where

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

and the summation is taken over values of  $(u, v)$  that lie inside the circle or on its boundary.

## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters



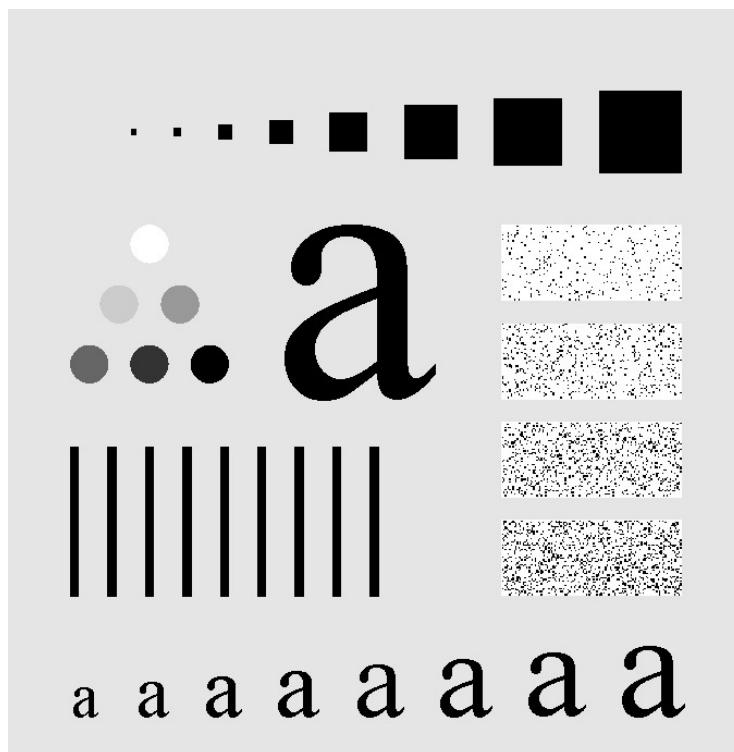
a

b

**FIGURE 4.41** (a) Test pattern of size  $688 \times 688$  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.

## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

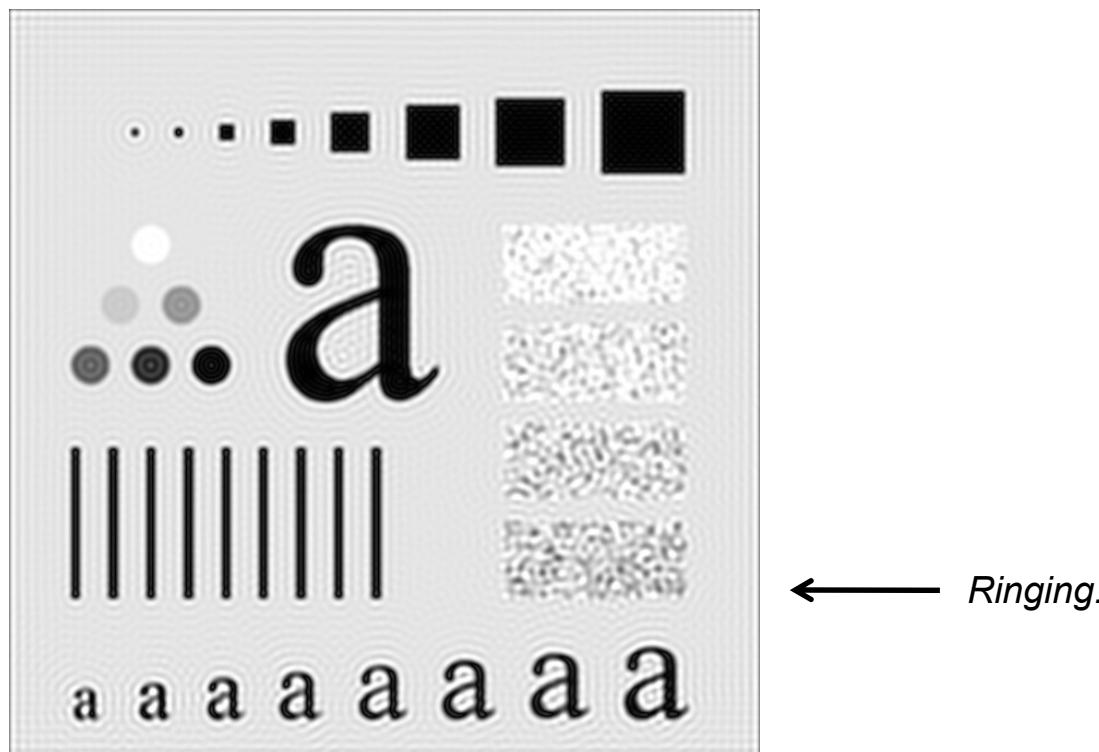
- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

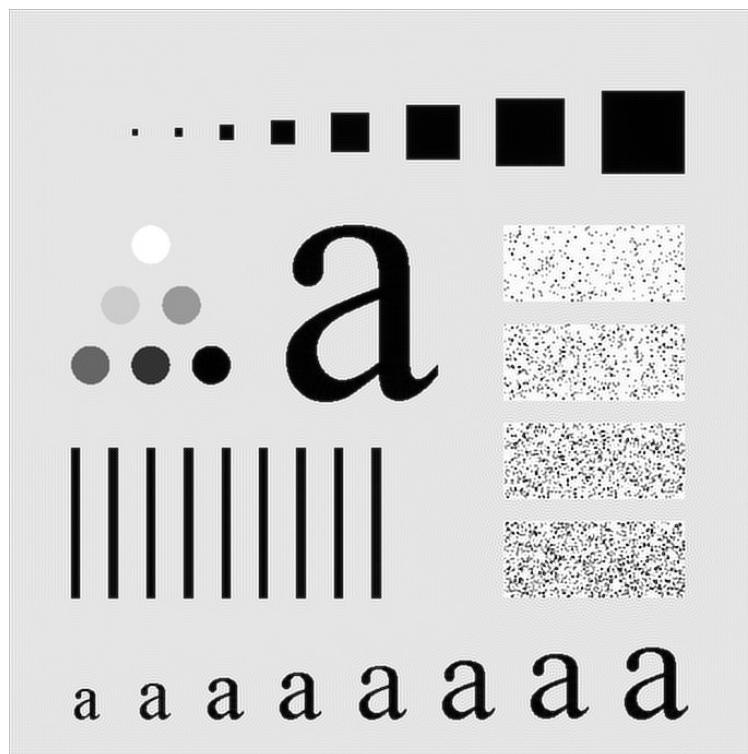
- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

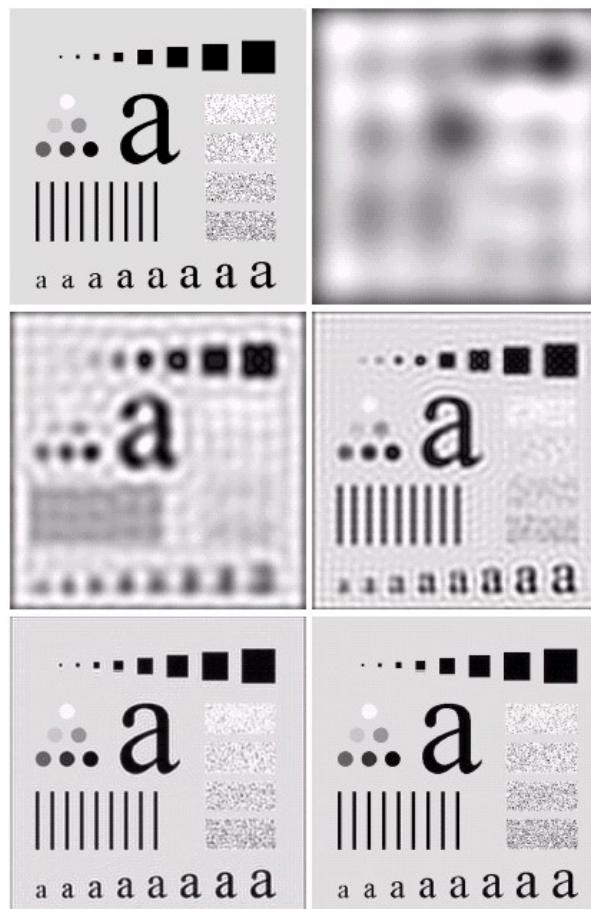
- Ideal Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

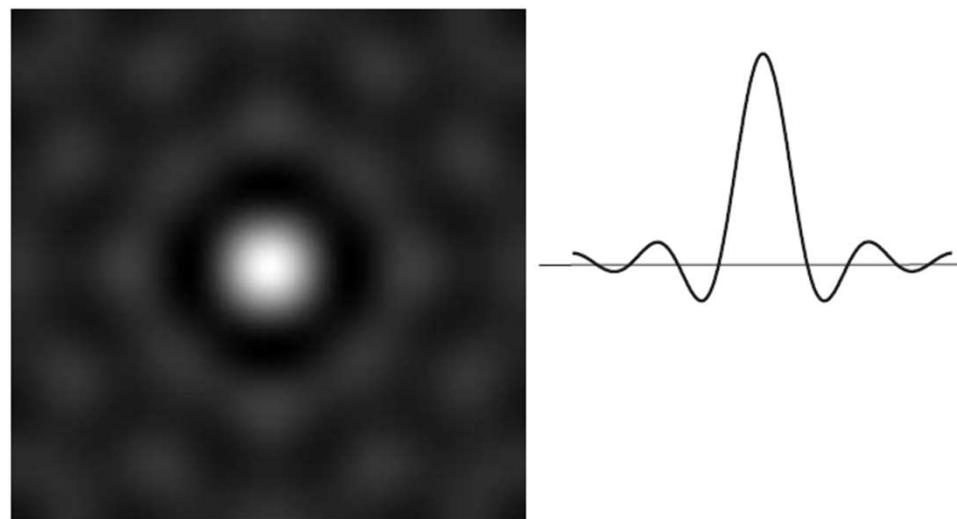
- Ideal Lowpass Filters



## 8. Image Smoothing Using Frequency Domain Filters

- Ideal Lowpass Filters

- The blurring and ringing properties of ILPFs can be explained using the convolution theorem.



a b

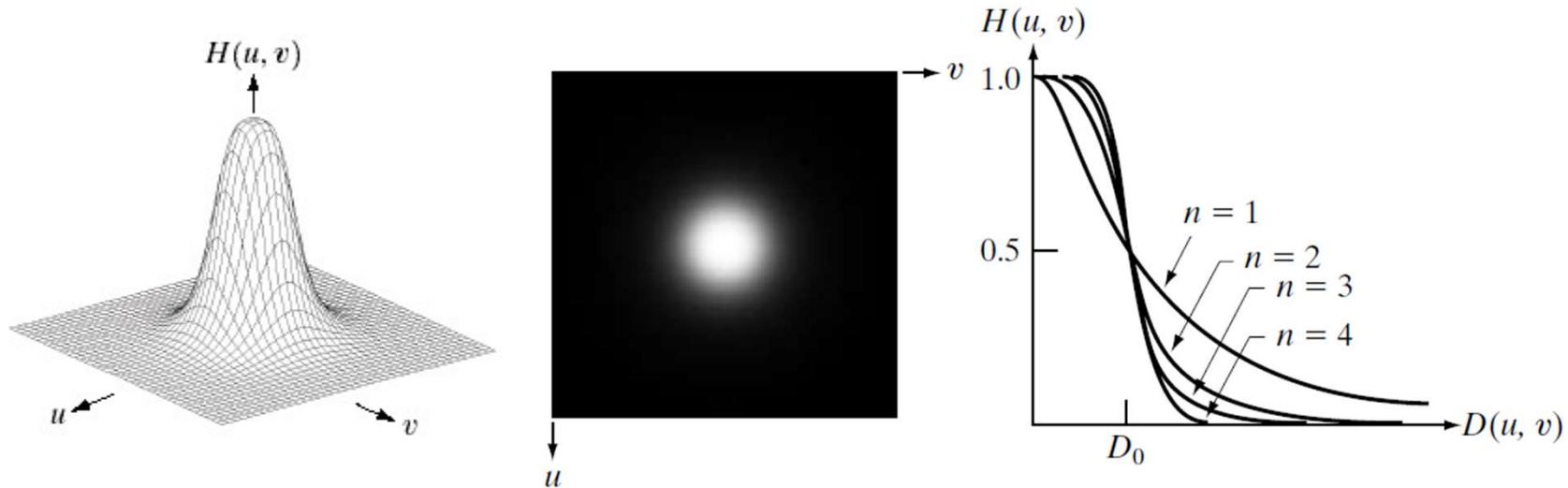
**FIGURE 4.43**  
(a) Representation in the spatial domain of an ILPF of radius 5 and size  $1000 \times 1000$ .  
(b) Intensity profile of a horizontal line passing through the center of the image.

## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters

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

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

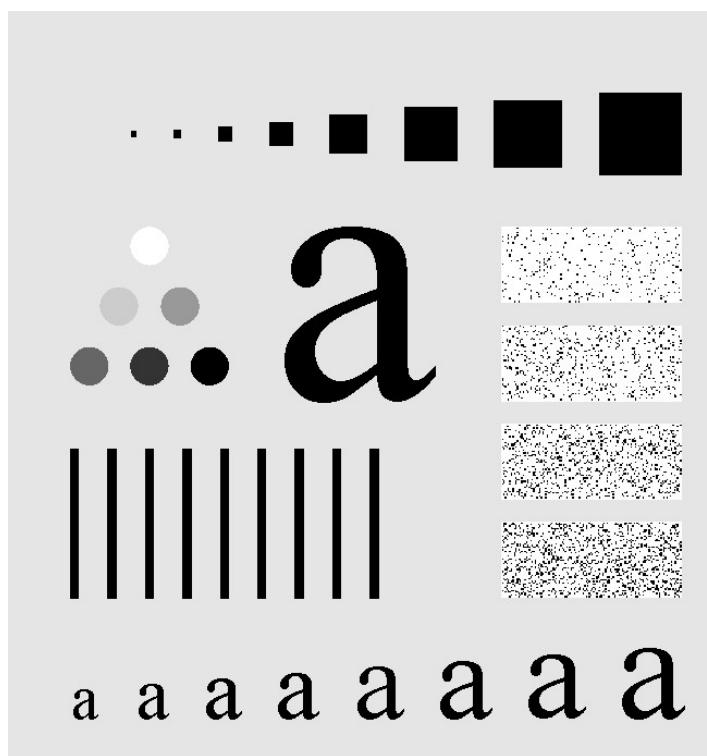


## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters
  - Unlike the ILPF, the **BLPF** transfer function **does not have a sharp discontinuity** that gives a clear cutoff between passed and filtered frequencies.
  - For filters with smooth transfer functions, cutoff frequency defined as those for which  $H(u, v)$  is **down to a certain fraction of its maximum** value (eg. 50%).
  - A BLPF of **order 1** has **no ringing** in the spatial domain. Ringing generally is **imperceptible** in filters of **order 2**, but can become **significant** in filters of **higher order**.

## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

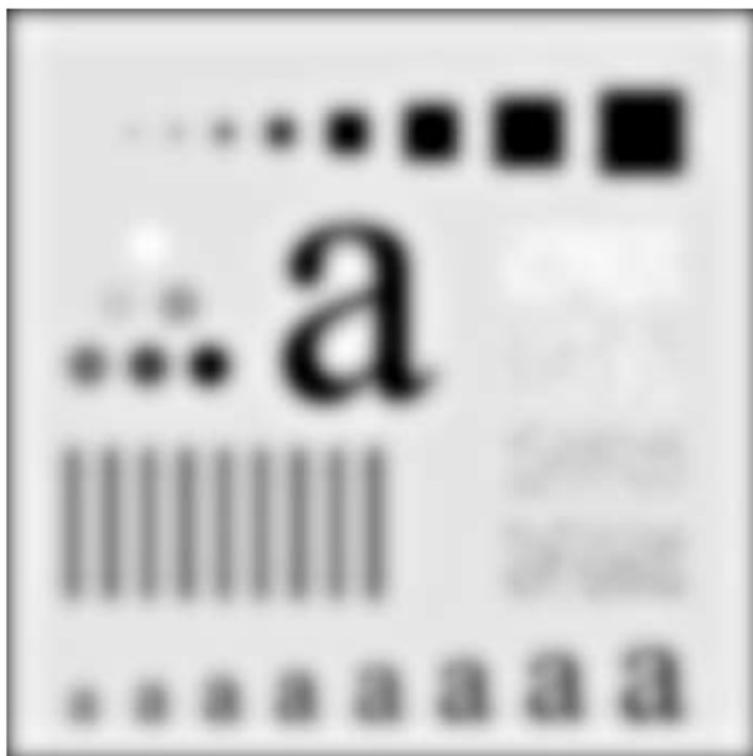
- Butterworth Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

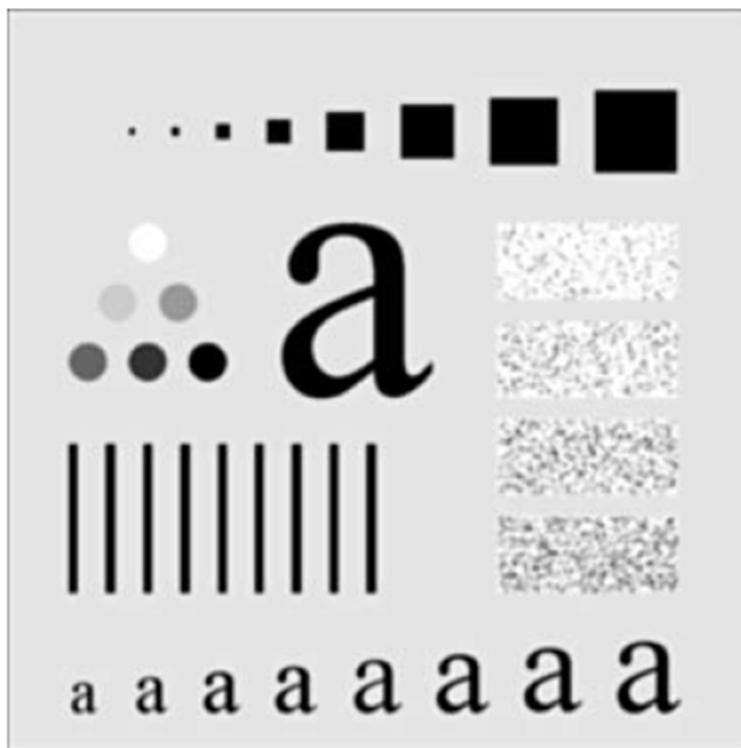
- Butterworth Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

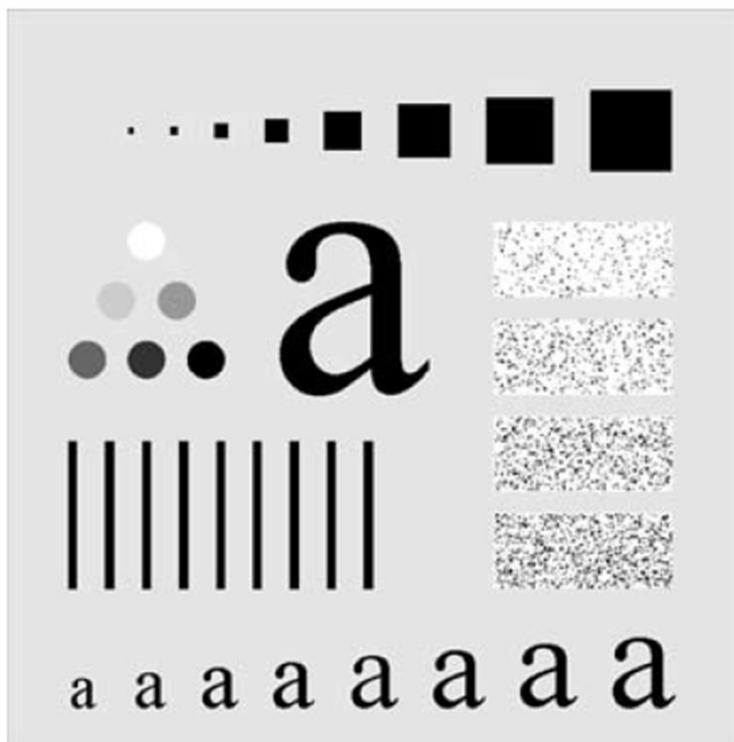
- Butterworth Lowpass Filters



**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.

## 8. Image Smoothing Using Frequency Domain Filters

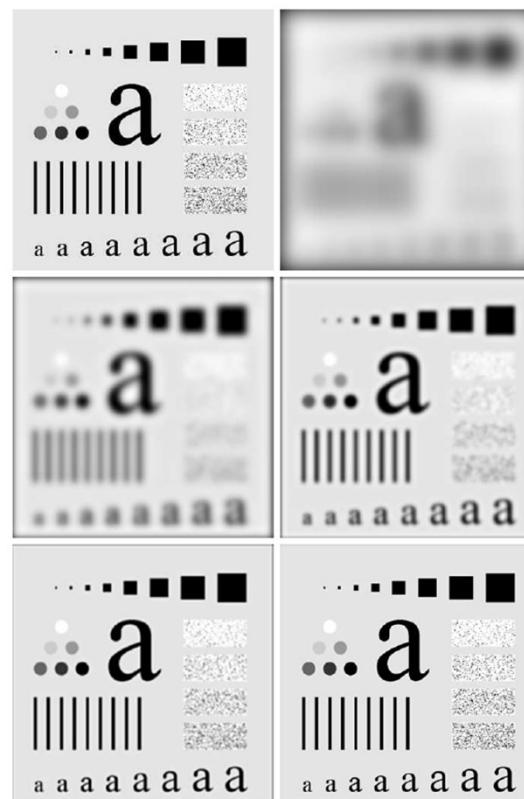
- Butterworth Lowpass Filters



**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.

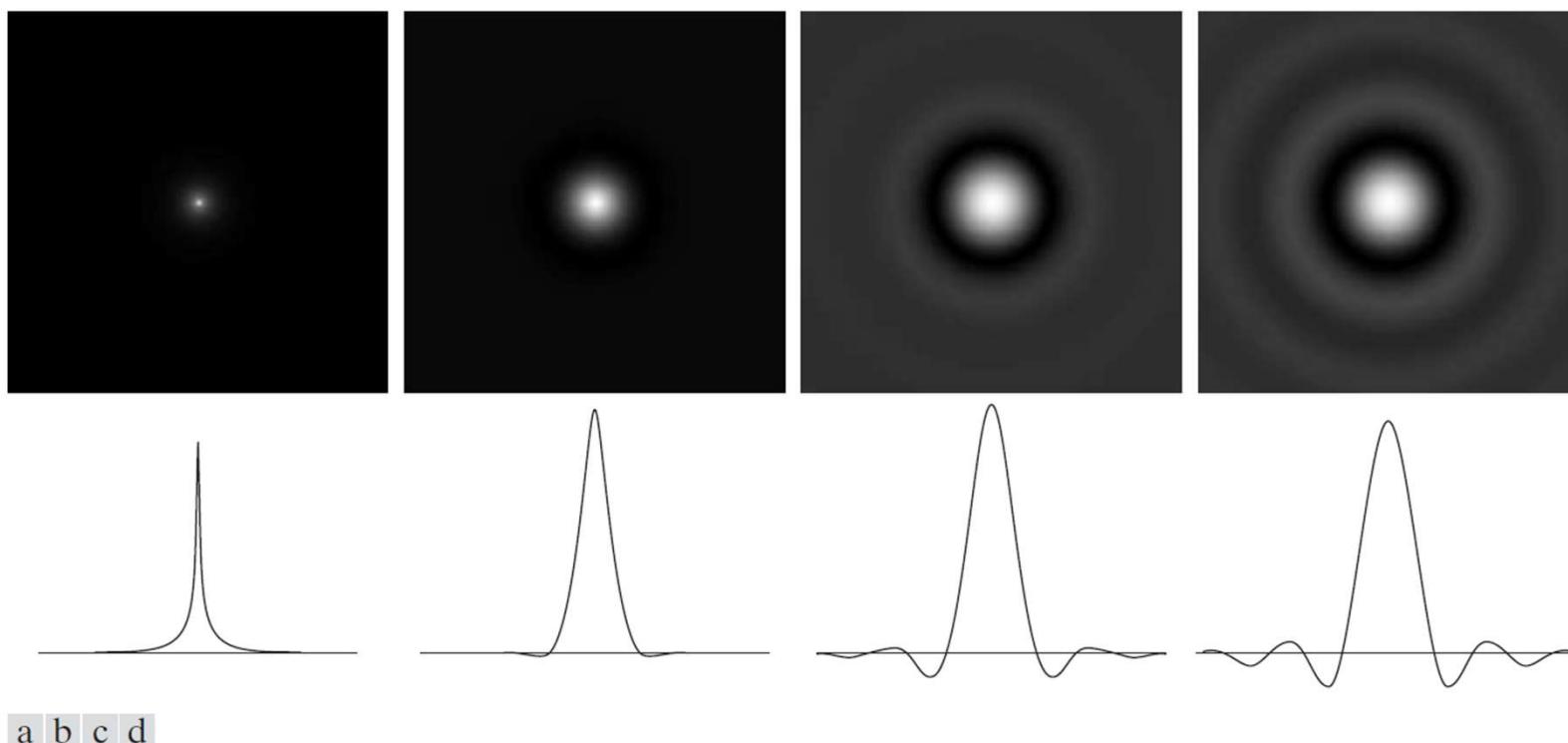
## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters
  - BLPFs of order 2. Cutoff frequencies same as before.



## 8. Image Smoothing Using Frequency Domain Filters

- Butterworth Lowpass Filters



a b c d

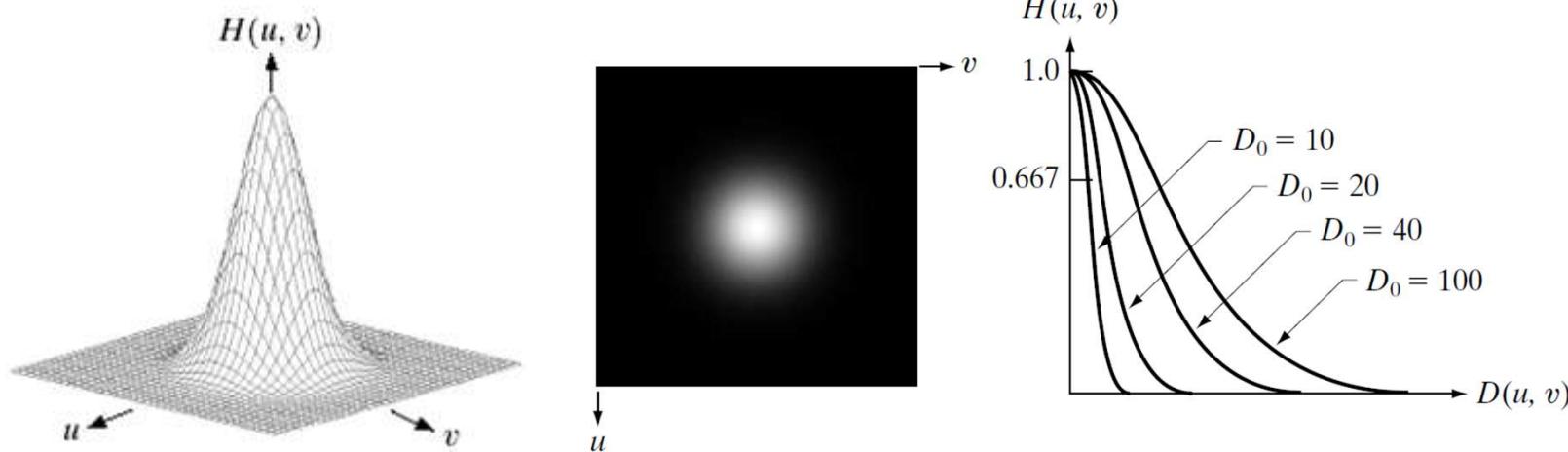
**FIGURE 4.46** (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is  $1000 \times 1000$  and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

## 8. Image Smoothing Using Frequency Domain Filters

- Gaussian Lowpass Filters

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

- $D_0$  is the cutoff frequency. When  $D(u, v) = D_0$ , the GLPF is down to 0.667 of its maximum value.



## 8. Image Smoothing Using Frequency Domain Filters

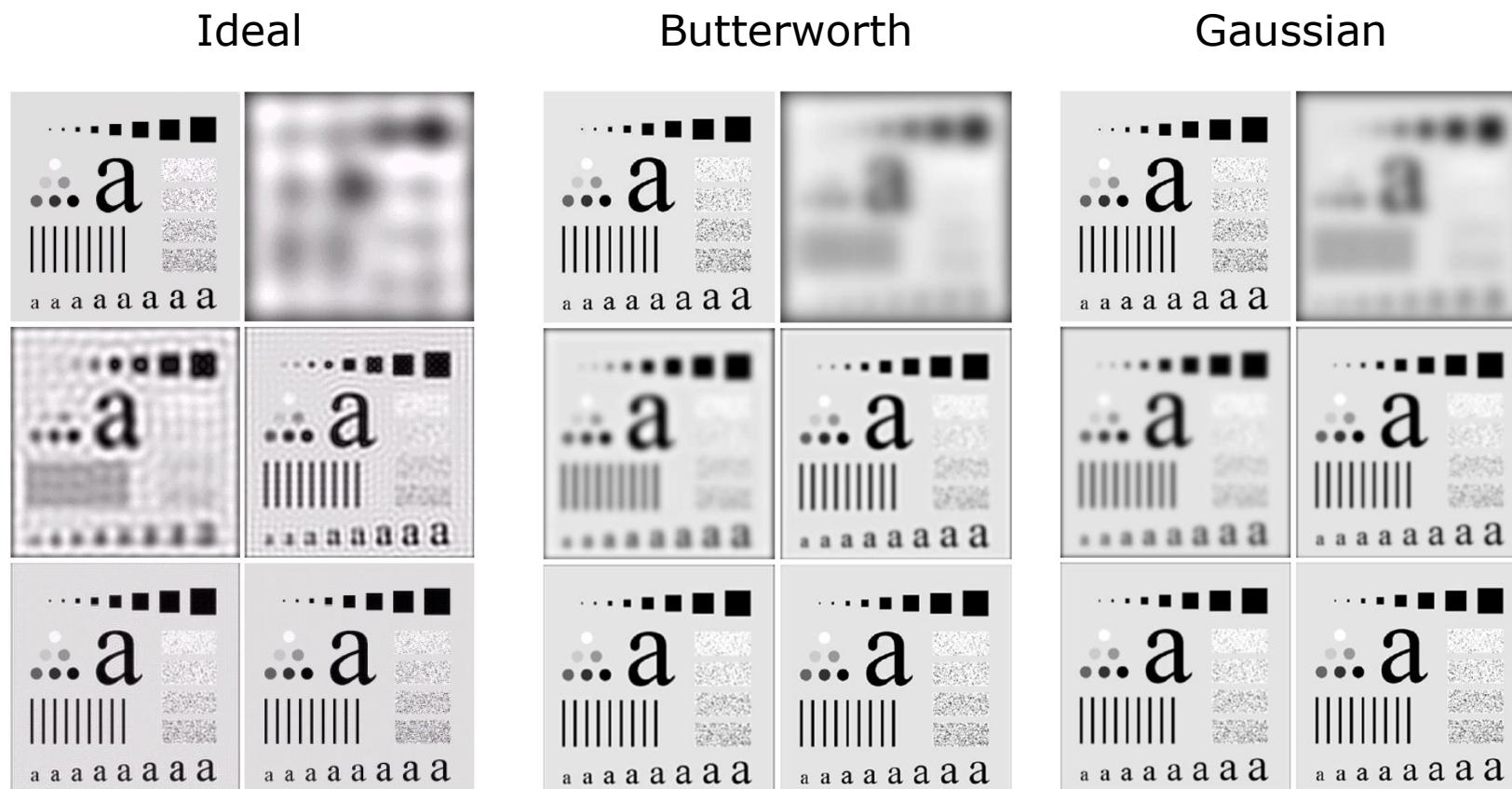
- Gaussian Lowpass Filters

➤ Cutoff frequencies same as before. No ringing effect.



## 8. Image Smoothing Using Frequency Domain Filters

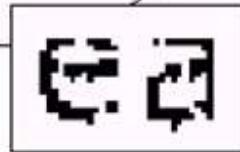
- Comparison



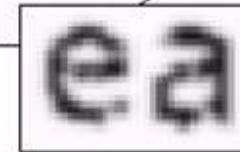
## 8. Image Smoothing Using Frequency Domain Filters

- Additional Examples of Lowpass Filtering

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.



a b

**FIGURE 4.49**  
(a) Sample text of low resolution (note broken characters in magnified view).  
(b) Result of filtering with a GLPF (broken character segments were joined).

## 8. Image Smoothing Using Frequency Domain Filters

- Additional Examples of Lowpass Filtering



a b c

**FIGURE 4.50** (a) Original image ( $784 \times 732$  pixels). (b) Result of filtering using a GLPF with  $D_0 = 100$ . (c) Result of filtering using a GLPF with  $D_0 = 80$ . Note the reduction in fine skin lines in the magnified sections in (b) and (c).

## 8. Image Smoothing Using Frequency Domain Filters

- Additional Examples of Lowpass Filtering



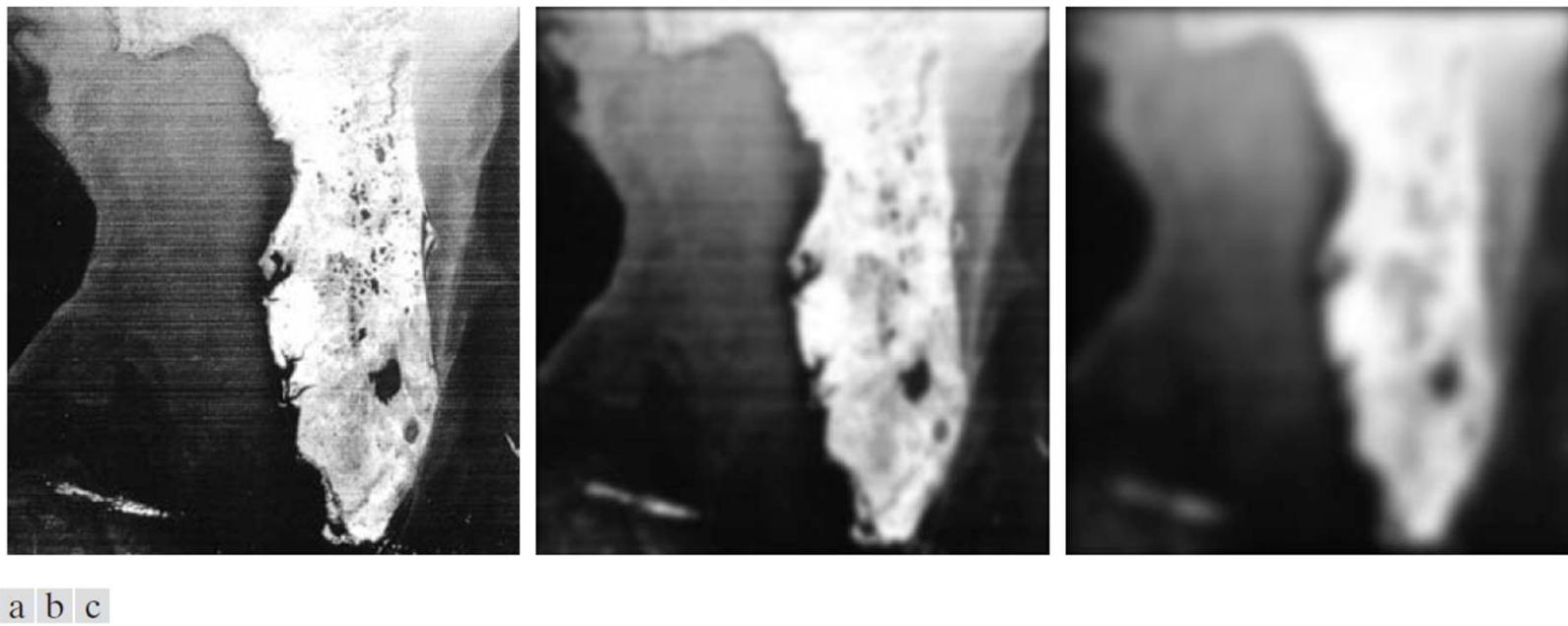
## 8. Image Smoothing Using Frequency Domain Filters

- Additional Examples of Lowpass Filtering



## 8. Image Smoothing Using Frequency Domain Filters

- Additional Examples of Lowpass Filtering
  - Here, the objective is to blur out as much detail as possible while leaving large features recognizable.



a b c

**FIGURE 4.51** (a) 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.)

## 9. Image Sharpening Using Frequency Domain Filters

- Highpass Filtering

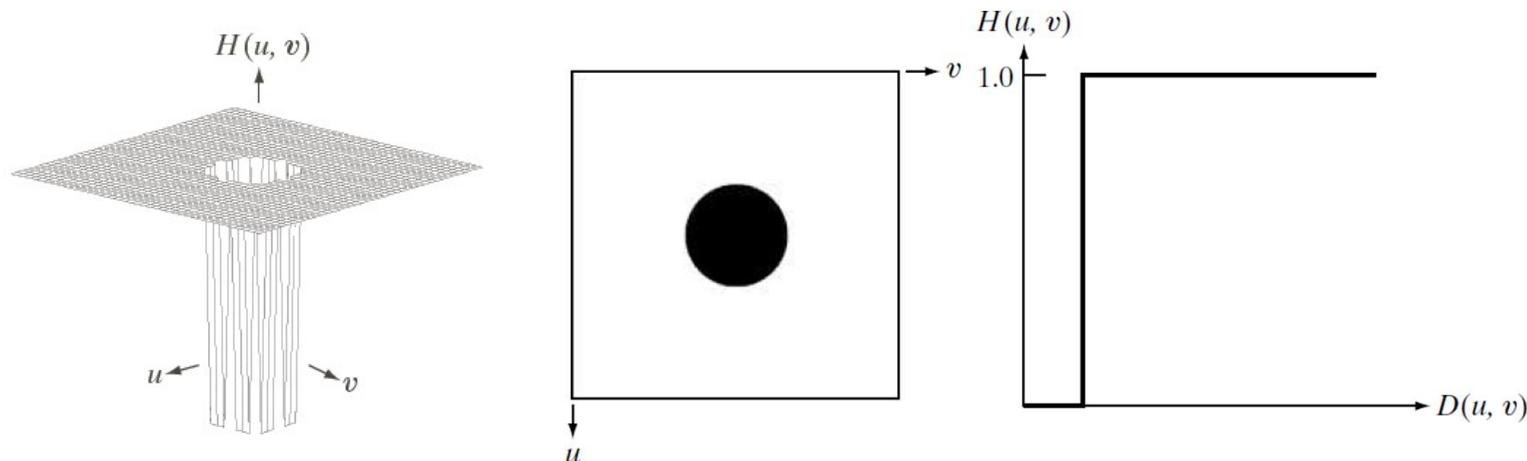
- A highpass filter is obtained from a given lowpass filter using the equation

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

## 9. Image Sharpening Using Frequency Domain Filters

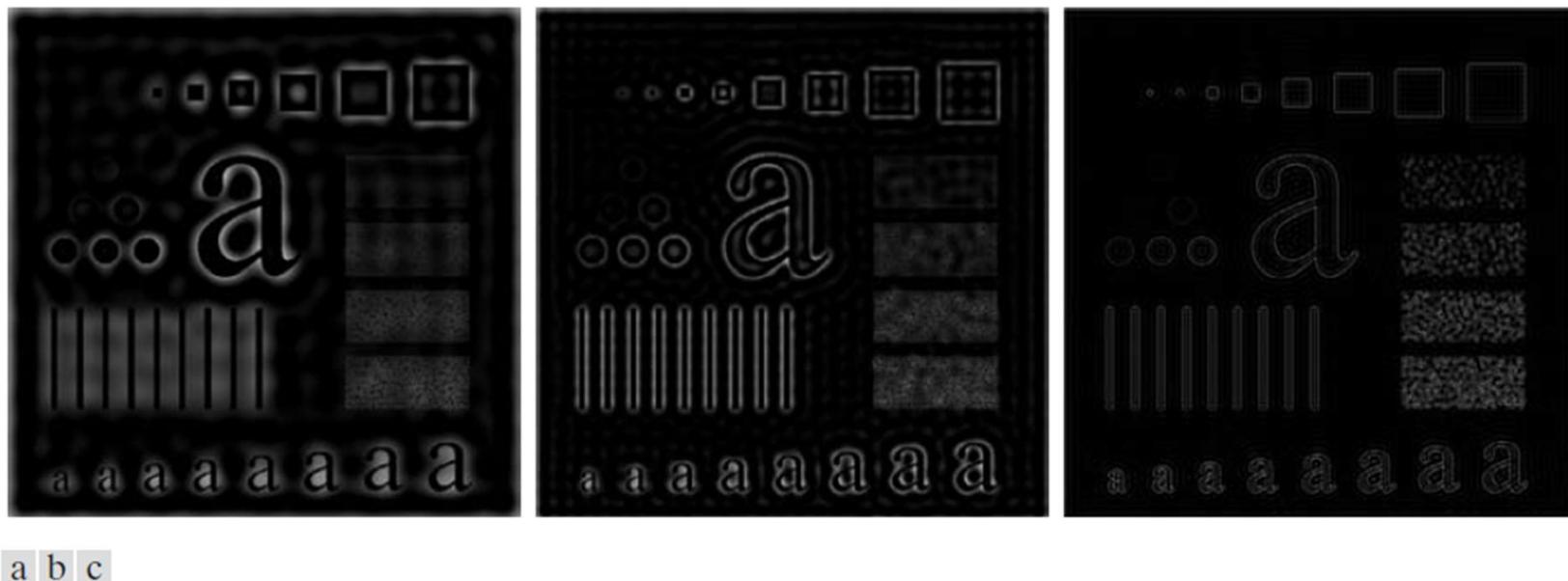
- Ideal Highpass Filters

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



## 9. Image Sharpening Using Frequency Domain Filters

- Ideal Highpass Filters



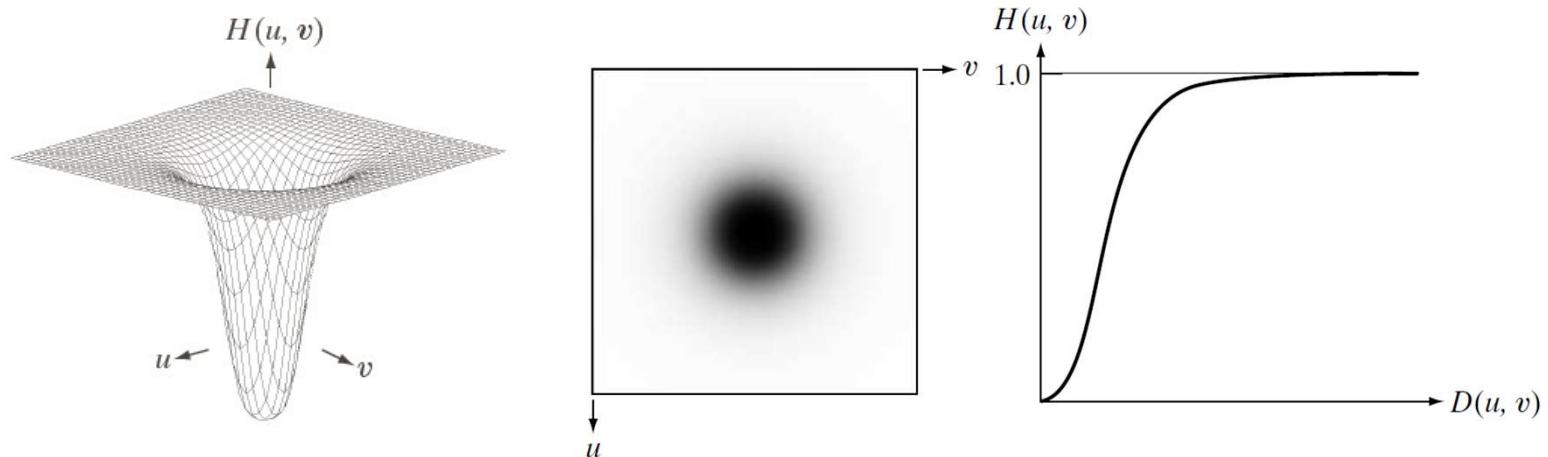
a b c

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

## 9. Image Sharpening Using Frequency Domain Filters

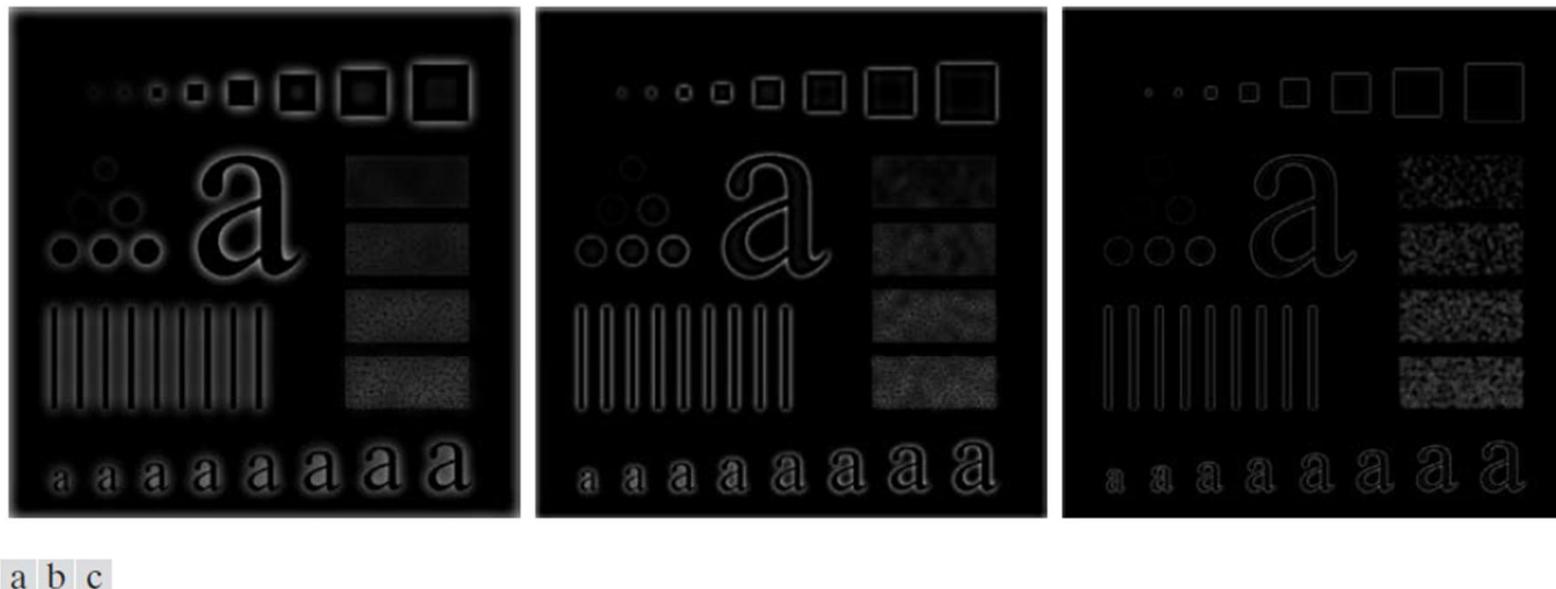
- Butterworth Highpass Filters

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



## 9. Image Sharpening Using Frequency Domain Filters

- Butterworth Highpass Filters

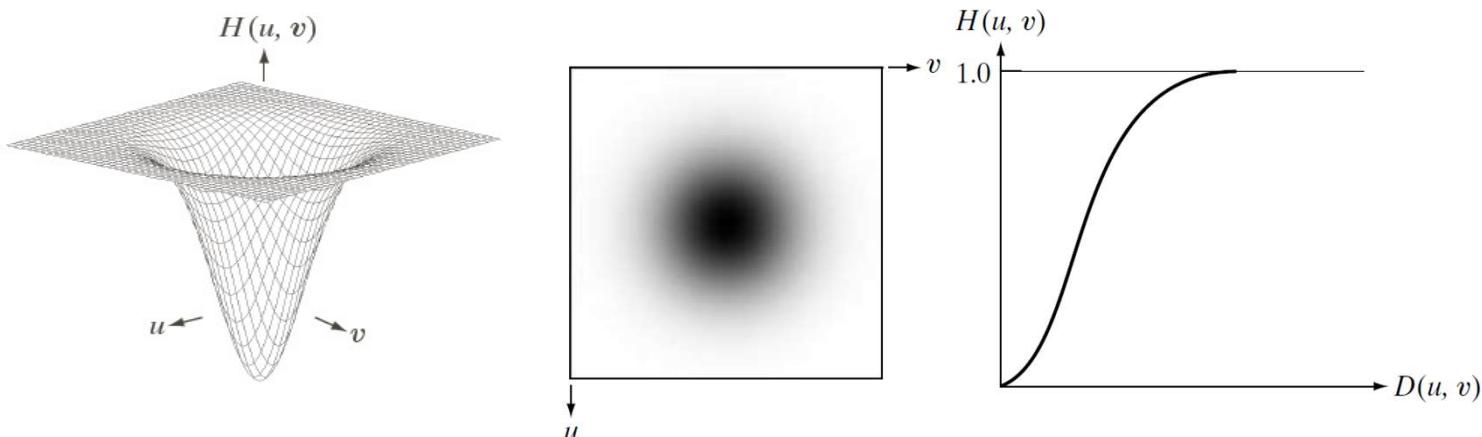


**FIGURE 4.55** Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with  $D_0 = 30, 60$ , and  $160$ , corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPF.

## 9. Image Sharpening Using Frequency Domain Filters

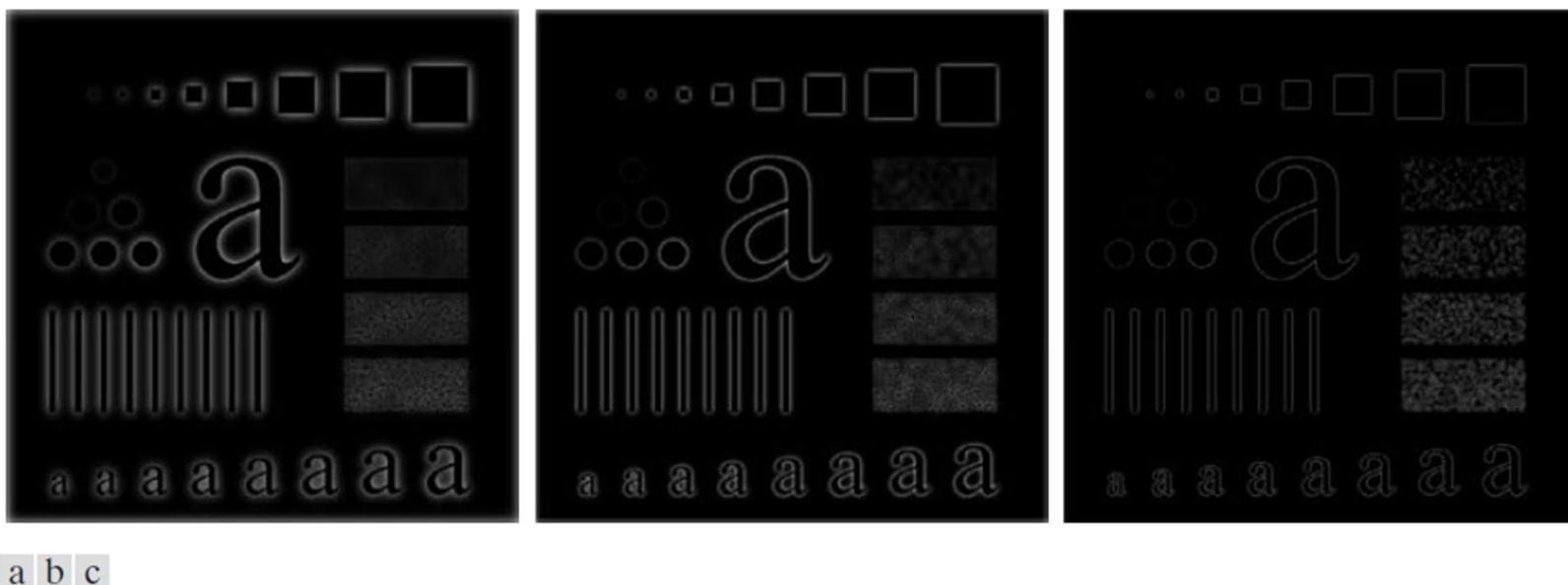
- Gaussian Highpass Filters

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



## 9. Image Sharpening Using Frequency Domain Filters

- Gaussian Highpass Filters



**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.

## 9. Image Sharpening Using Frequency Domain Filters

- Comparison

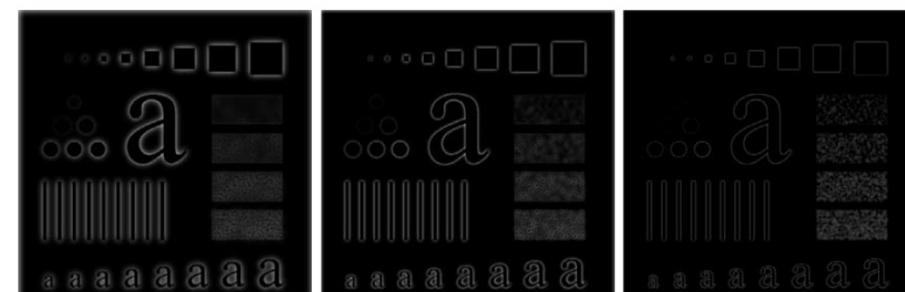
➤ Ideal



➤ Butterworth

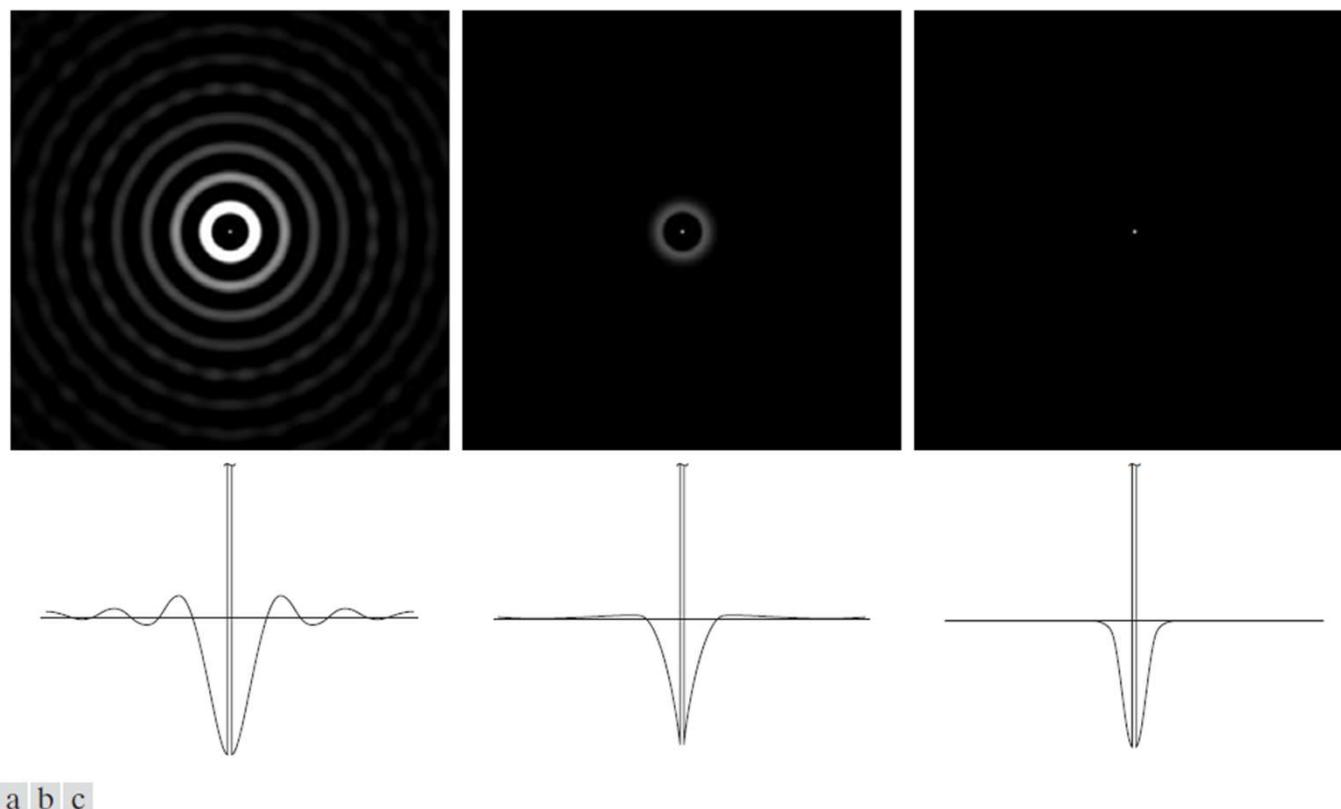


➤ Gaussian



## 9. Image Sharpening Using Frequency Domain Filters

- Spatial representation



a b c

**FIGURE 4.53** Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

## 9. Image Sharpening Using Frequency Domain Filters

- Example



a b c

**FIGURE 4.57** (a) Thumb print. (b) Result of highpass filtering (a). (c) Result of thresholding (b). (Original image courtesy of the U.S. National Institute of Standards and Technology.)

## 9. Image Sharpening Using Frequency Domain Filters

- Low/highpass filters equations

Lowpass filters.  $D_0$  is the cutoff frequency and  $n$  is the order of the Butterworth filter.

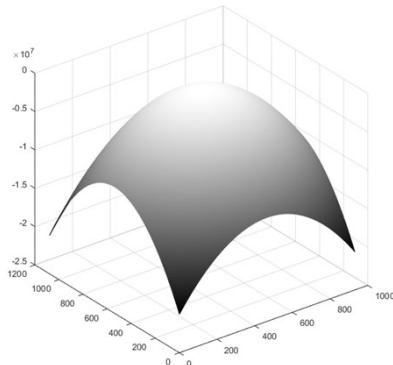
Ideal	Butterworth	Gaussian
$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$	$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$	$H(u, v) = e^{-D^2(u,v)/2D_0^2}$

Highpass filters.  $D_0$  is the cutoff frequency and  $n$  is the order of the Butterworth filter.

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

## 9. Image Sharpening Using Frequency Domain Filters

- The **Laplacian** in the Frequency Domain
  - It can be shown that the Laplacian can be implemented in the frequency domain using the filter



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

$$\begin{aligned} \left(\frac{\partial}{\partial t}\right)^m \left(\frac{\partial}{\partial z}\right)^n f(t, z) &\Leftrightarrow (j2\pi\mu)^m (j2\pi\nu)^n F(\mu, \nu) \\ \frac{\partial^m f(t, z)}{\partial t^m} &\Leftrightarrow (j2\pi\mu)^m F(\mu, \nu); \frac{\partial^n f(t, z)}{\partial z^n} \Leftrightarrow (j2\pi\nu)^n F(\mu, \nu) \end{aligned}$$

or, with respect to the center of the frequency rectangle, using the filter

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

## 9. Image Sharpening Using Frequency Domain Filters

- The Laplacian in the Frequency Domain

➤ Then, the Laplacian image is obtained as:

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

➤ As explained before, enhancement is achieved using the equation:

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

➤ Here,  $c = -1$  because  $H(u, v)$  is negative. In the frequency domain, is written as

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

## 9. Image Sharpening Using Frequency Domain Filters

- MATLAB: s193LaplacianFreq.m



a b

**FIGURE 4.58**  
(a) Original,  
blurry image.  
(b) Image  
enhanced using  
the Laplacian in  
the frequency  
domain. Compare  
with Fig. 3.38(e).

## 9. Image Sharpening Using Frequency Domain Filters

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

✓ First a mask is defined  $g_{\text{mask}}(x, y) = f(x, y) - f_{\text{LP}}(x, y)$

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

where  $H_{\text{LP}}(u, v)$  is a lowpass filter and  $F(u, v)$  is the Fourier transform of  $f(x, y)$ .

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

✓ This expression defines **unsharp** masking when  $k = 1$  and **highboost** filtering when  $k > 1$ .

## 9. Image Sharpening Using Frequency Domain Filters

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

- ✓ We can express the preceding result entirely in terms of frequency domain computations involving a lowpass filter:

$$g(x, y) = \mathfrak{F}^{-1}\left\{\left[1 + k * [1 - H_{LP}(u, v)]\right]F(u, v)\right\}$$

- ✓ We can express this result in terms of a highpass filter:

$$g(x, y) = \mathfrak{F}^{-1}\left\{\left[1 + k * H_{HP}(u, v)\right]F(u, v)\right\}$$

- ✓ The expression contained within the square brackets is called a **high-frequency emphasis** filter.
  - ✓ The constant,  $k$ , gives control over the proportion of high frequencies that influence the final result.

## 9. Image Sharpening Using Frequency Domain Filters

- Unsharp Masking, Highboost Filtering, and High-Frequency-Emphasis Filtering
  - ✓ A slightly more general formulation of high-frequency-emphasis filtering is the expression

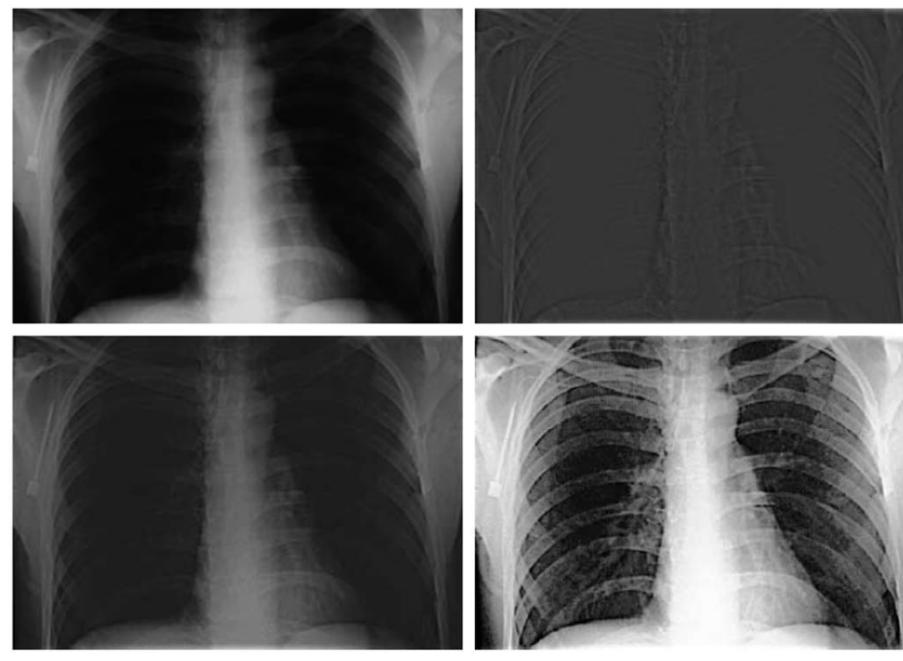
$$g(x, y) = \mathfrak{F}^{-1}\left\{ [k_1 + k_2 * H_{HP}(u, v)]F(u, v)\right\}$$

where  $k_1 \geq 0$  gives controls of the offset from the origin and  $k_2 \geq 0$  controls the contribution of high frequencies.

## 9. Image Sharpening Using Frequency Domain Filters

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

$$\begin{aligned} k_1 &= 0.5 \\ k_2 &= 0.75 \end{aligned}$$



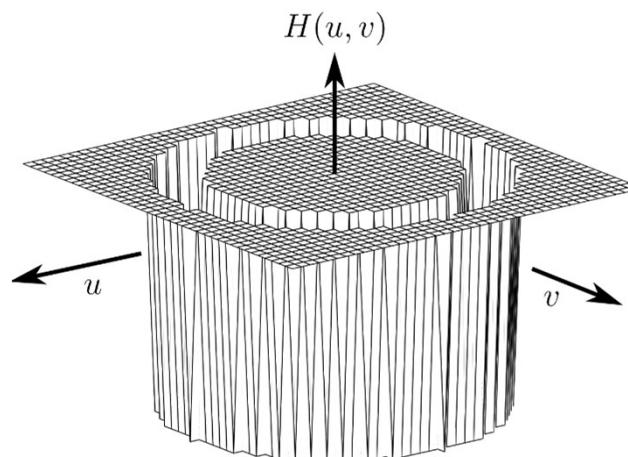
a b  
c d

**FIGURE 4.59** (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.)

## 10. Selective Filtering

- Bandreject and Bandpass Filters
  - Ideal Bandreject Filters

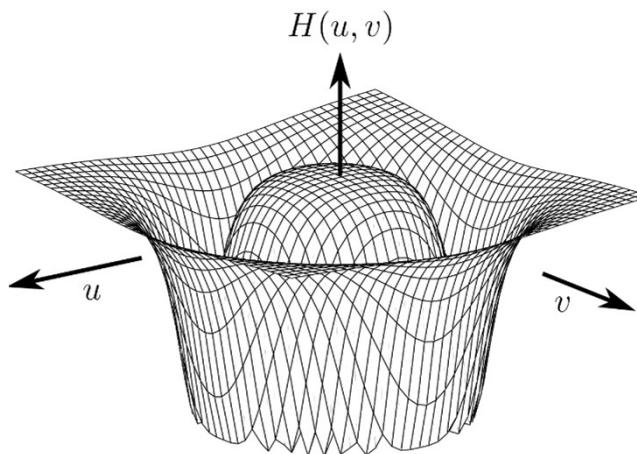
$$H(u, v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \leq D \leq D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$$



## 10. Selective Filtering

- Bandreject and Bandpass Filters
  - Butterworth Bandreject Filters

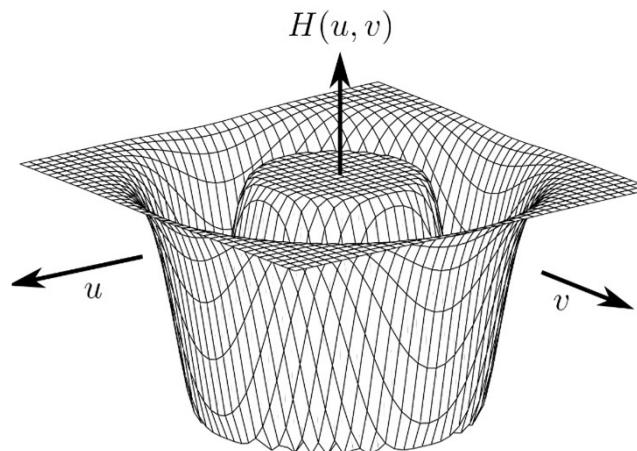
$$H(u, v) = \frac{1}{1 + \left[ \frac{DW}{D^2 - D_0^2} \right]^{2n}}$$



## 10. Selective Filtering

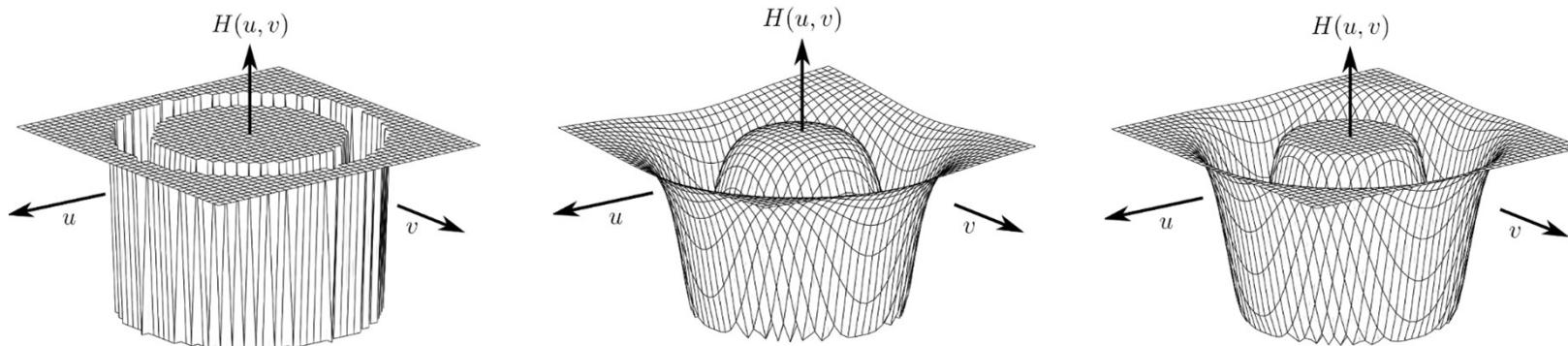
- Bandreject and Bandpass Filters
  - Gaussian Bandreject Filters

$$H(u, v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$$



## 10. Selective Filtering

- Bandreject and Bandpass Filters



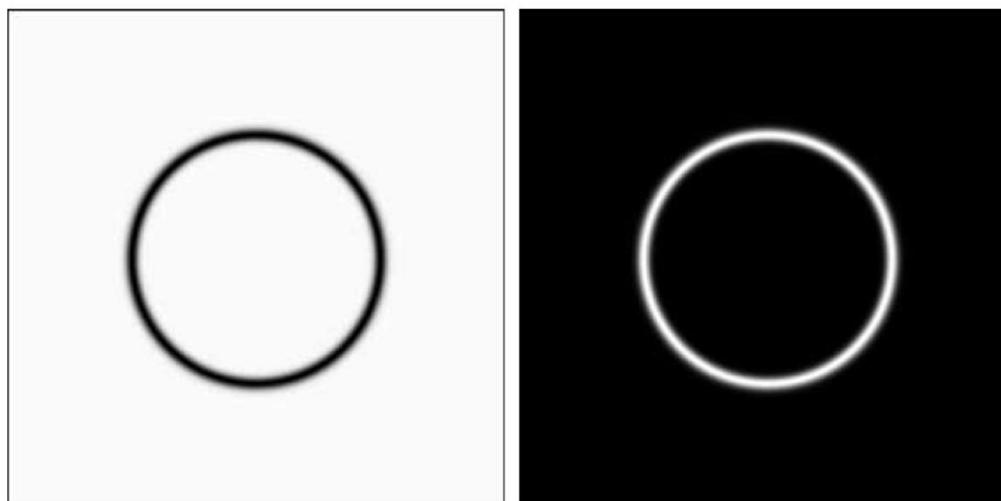
Ideal	Butterworth	Gaussian
$H(u, v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \leq D \leq D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u, v) = \frac{1}{1 + \left[ \frac{DW}{D^2 - D_0^2} \right]^{2n}}$	$H(u, v) = 1 - e^{-\left[ \frac{D^2 - D_0^2}{DW} \right]^2}$

- $D$  is the distance  $D(u,v)$  from the center of the filter,  $D_0$  is the radial center of the band,  $W$  is the width of the band (upper-lower cutoff frequency) and  $n$  is the order of the Butterworth filter.

## 10. Selective Filtering

- Bandreject and Bandpass Filters
  - A bandpass filter is obtained from a bandreject filter in the same manner that we obtained a highpass filter from a lowpass filter:

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



a b

**FIGURE 4.63**  
(a) Bandreject Gaussian filter.  
(b) Corresponding bandpass filter.  
The thin black border in (a) was added for clarity; it is not part of the data.

## 10. Selective Filtering

- MATLAB: s204FilterImplem.m

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

## 10. Selective Filtering

- Notch Filters

- Notch filters rejects (or passes) frequencies in a predefined neighborhood about the center of the frequency rectangle.
- They are **symmetric about the origin**, so 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 products of **highpass filters** whose centers have been **translated to the centers of the notches**. The general form is:

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

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

## 10. Selective Filtering

- Notch Filters

- These centers are specified with respect to the center of the frequency rectangle,  $(M/2, N/2)$ .
- The distance computations for each filter are thus carried out using the expressions

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

and

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

- For example, the following is a Butterworth notch reject filter of order  $n$ , containing **three notch pairs**

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

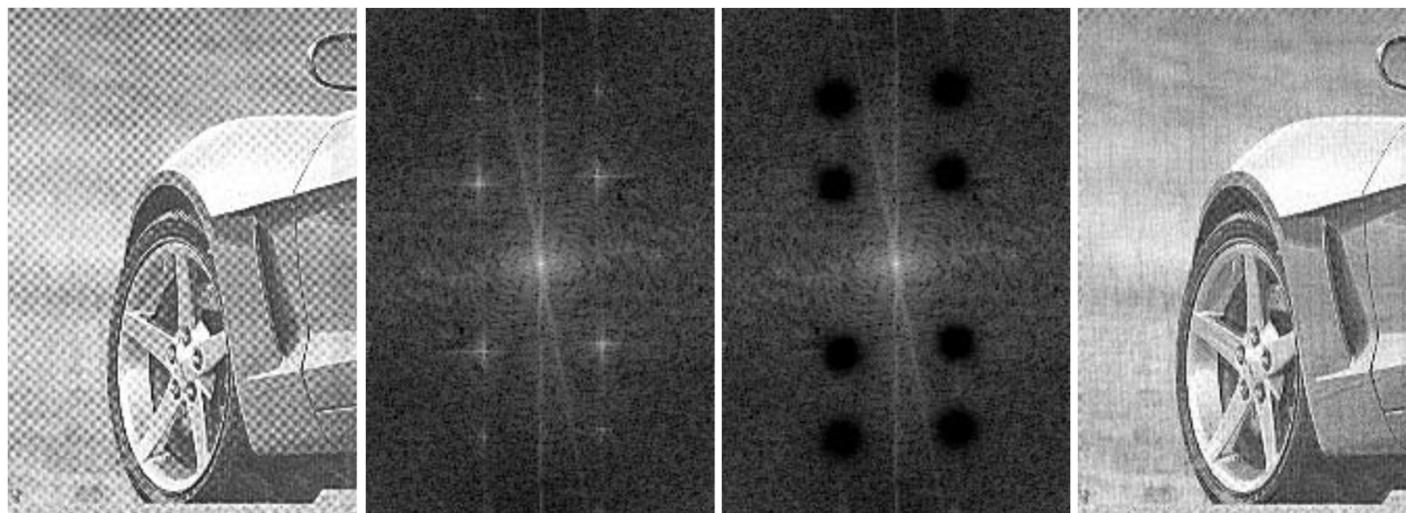
## 10. Selective Filtering

- Notch Filters

- A notch pass filter is obtained from a notch reject filter using the expression

$$H_{NP}(u, v) = 1 - H_{NR}(u, v)$$

- Example



**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.

## 10. Selective Filtering

- MATLAB: s206Notch.m

