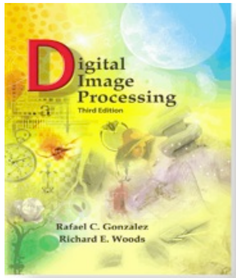


4. Filtering in the Frequency Domain

1. Background
2. Preliminary Concepts
3. Sampling and the Fourier Transform of Sampled Functions
4. The Discrete Fourier Transform of One Variable
5. Extension to Functions of Two Variables
6. Some Properties of the 2-D Discrete Fourier Transform
7. The Basics of Filtering in the Frequency Domain
8. Image Smoothing using Frequency Domain Filters
9. Image Sharpening using Frequency Domain Filters
10. Selective Filtering

4.1 Background

Fourier Series and Transform

Jean-Baptiste Fourier (1768)

• Any periodic function can be expressed as a weighted sum of sines and/or cosines (*Fourier series*)

• Other functions can be expressed as an integral of sines and/or cosines multiplied by a weighing function (*Fourier Transform*)

• Functions can be recovered by the inverse operation with *no loss of information*

• Applications to Image Enhancement

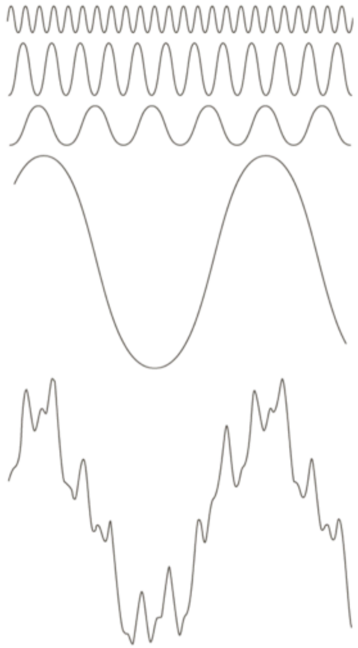
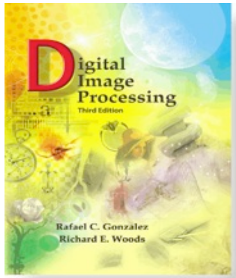


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.



4.2 Preliminary Concepts

4.2.1 Complex Numbers

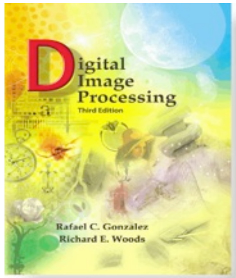
4.2.2 Fourier Series

If *f(t)* is a periodic function of a continuous variable *t*, with period *T*

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

Where:

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



4.2.3 Impulses and their Shifting Property

Unit impulse of a continuous variable *t* located at *t=0* :

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

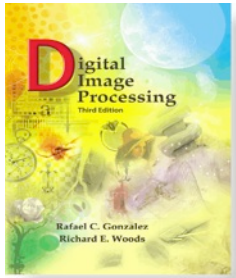
and

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

Sifting property :

$$\int_{-\infty}^{\infty} f(t)\delta(t)dt = f(0) \qquad \int_{-\infty}^{\infty} f(t)\delta(t-t_0)dt = f(t_0)$$

(if *f(t)* continuous at *t=0*)



Unit discrete impulse located at $x=0$:

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

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

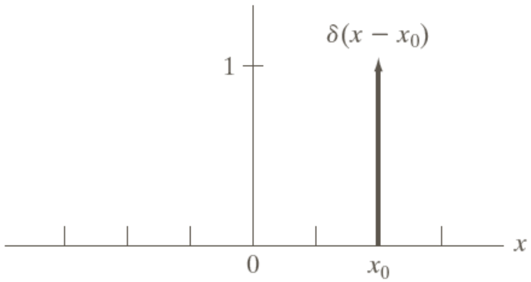
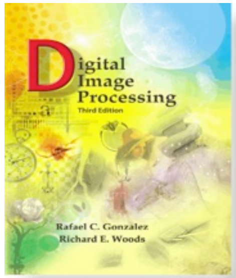


FIGURE 4.2 A unit discrete impulse located at $x = x_0$. Variable x is discrete, and δ is 0 everywhere except at $x = x_0$.

Sifting property :

$$\sum_{x=-\infty}^{\infty} f(x)\delta(x) = f(0) \qquad \sum_{x=-\infty}^{\infty} f(x)\delta(x - x_0) = f(x_0)$$



Impulse train : sum of infinitely many periodic impulses ΔT units apart:

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

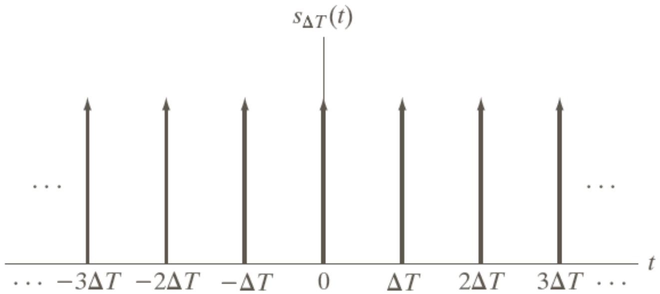


FIGURE 4.3 An impulse train.

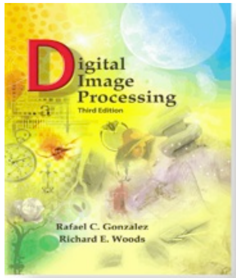
4.2.4 The Fourier Transform of Functions of One Continuous Variable

Fourier Transform of a continuous function *f(t)* of a continuous variable *t* :

$$FT\left[f(t)\right] = \quad F(\mu) = \int_{-\infty}^{\infty} f(t)e^{-j2\pi\mu t}dt$$

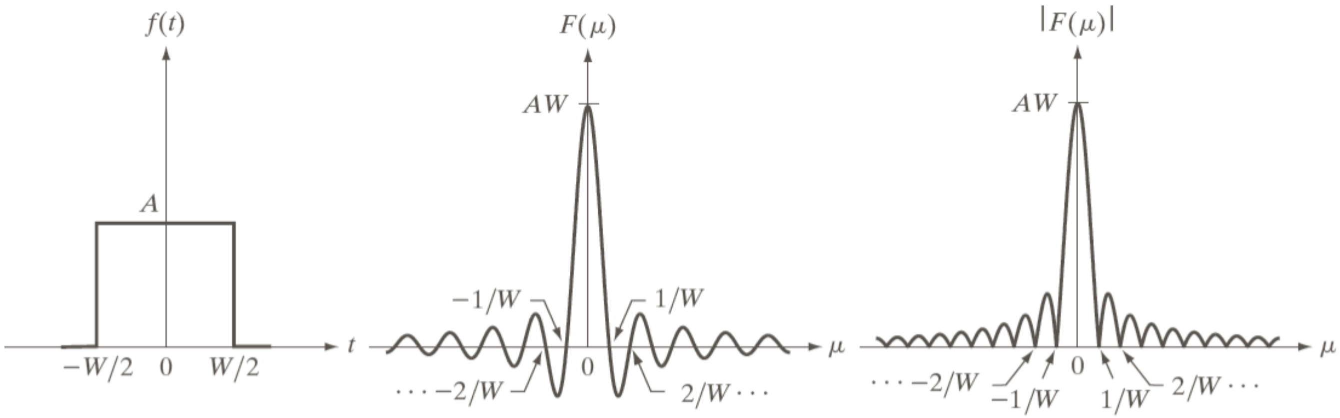
Inverse Fourier Transform:

$$FT^{-1}\left[F(\mu)\right] = \quad f(t) = \int_{-\infty}^{\infty} F(\mu)e^{j2\pi\mu t}d\mu$$



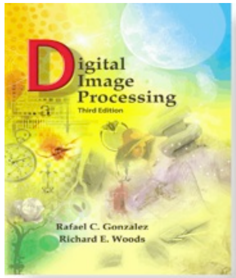
Example : $$F(\mu) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi\mu t} dt = \int_{-W/2}^{W/2} A e^{-j2\pi\mu t} dt$$

$$= AW \frac{\sin(\pi\mu W)}{(\pi\mu W)} \quad = AW\, sinc(\pi\mu W)$$



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.



Example 2 : Fourier Transform of a unit impulse located at the origin:

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

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

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

Example 2 : Fourier Transform *S(μ)* of an impulse train with period *ΔT* :

Periodic function with period *ΔT* =>

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

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

=> $$S(\mu) = FT\left[s_{\Delta t}(t)\right] = \frac{1}{\Delta T} FT\left[\sum_{n=-\infty}^{\infty} e^{j\frac{2\pi n}{T}}\right] = \frac{1}{\Delta T}\sum_{n=-\infty}^{\infty} \delta\left(\mu - \frac{n}{\Delta T}\right)$$

Also an *impulse train*, with period 1/*ΔT*

4.2.5 Convolution

Convolution of functions *f(t)* and *h(t)*, of one continuous variable *t* :

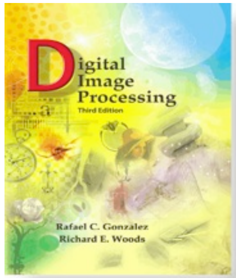
$$f(t)\bigstar h(t) = \int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau$$

$$FT\{f(t)\bigstar h(t)\} = H(\mu)F(\mu)$$

Fourier Transform pairs:

$$f(t)\bigstar h(t) \Longleftrightarrow H(\mu)F(\mu)$$

$$f(t)h(t) \Longleftrightarrow H(\mu)\bigstar F(\mu)$$

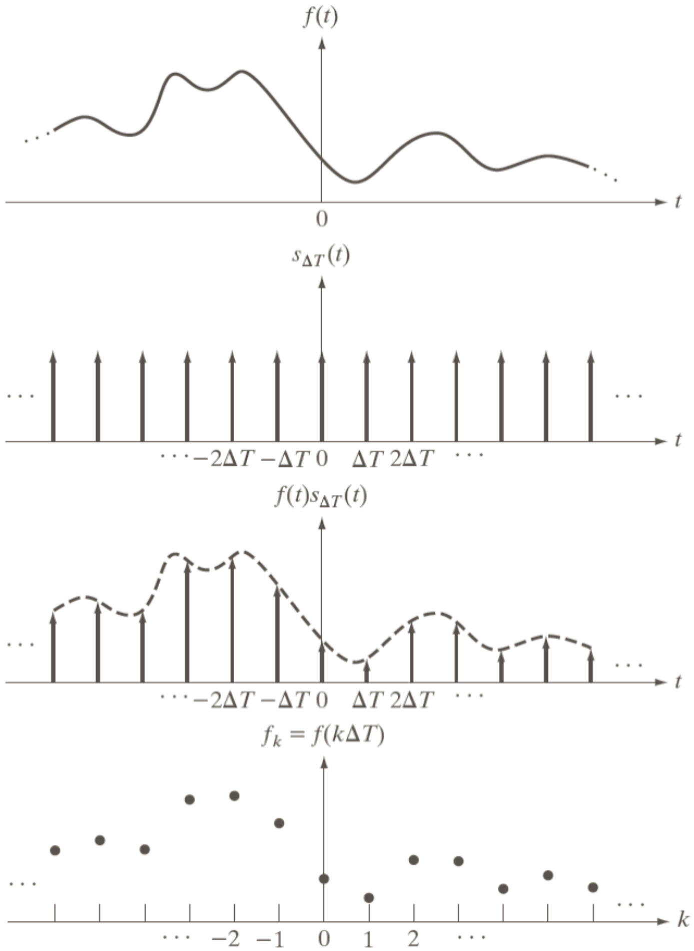


4.3 Sampling and the Fourier Transform of Sampled Functions

4.3.1 Sampling

Model of the sampled function: multiplication of *f(t)* by a sampling function equal to a train of impulses *ΔT* units apart

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



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.)

4.3.2 The Fourier Transform of Sampled Functions

$$\tilde{F}(\mu) = FT\left[\tilde{f}(t)\right] = F(\mu) \bigstar S(\mu) \quad = \frac{1}{\Delta T}\sum_{n=-\infty}^{\infty} F\left(\mu - \frac{n}{\Delta T}\right)$$

Infinite, *periodic* sequence of *copies* of *F(μ)*, *continuous*

Separation between copies determined by $1/\Delta T$

4.3.3 The Sampling Theorem

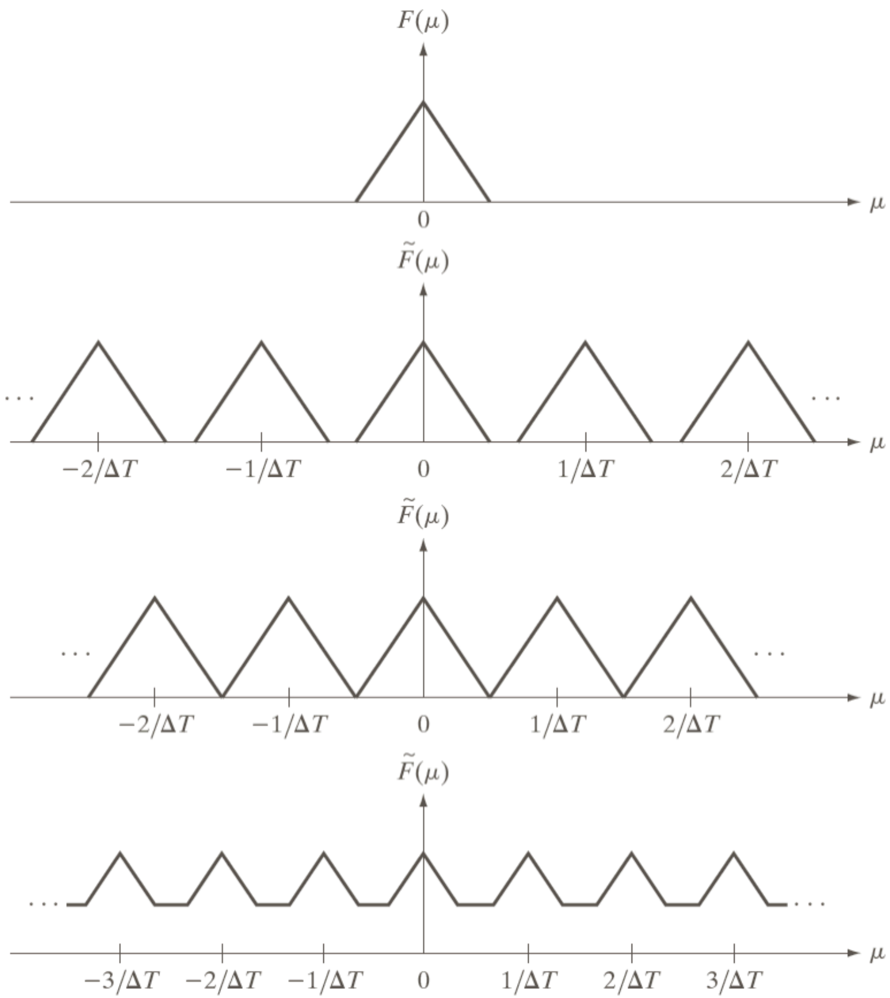
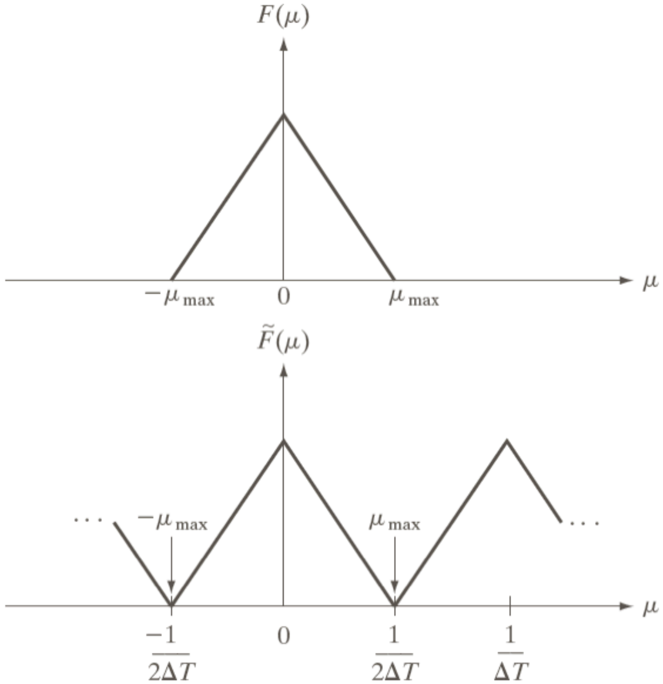


FIGURE 4.6 (a) Fourier transform of a band-limited function. (b)–(d) Transforms of the corresponding sampled function under the conditions of over-sampling, critically-sampling, and under-sampling, respectively.

A function *f(t)* whose Fourier transform is zero for values of frequencies outside a finite interval (band) [- $\mu_{\max}$, $\mu_{\max}$] 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.

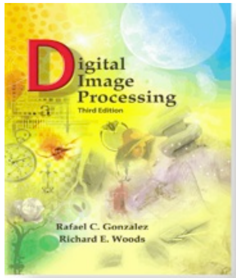
Sufficient separation guaranteed if:

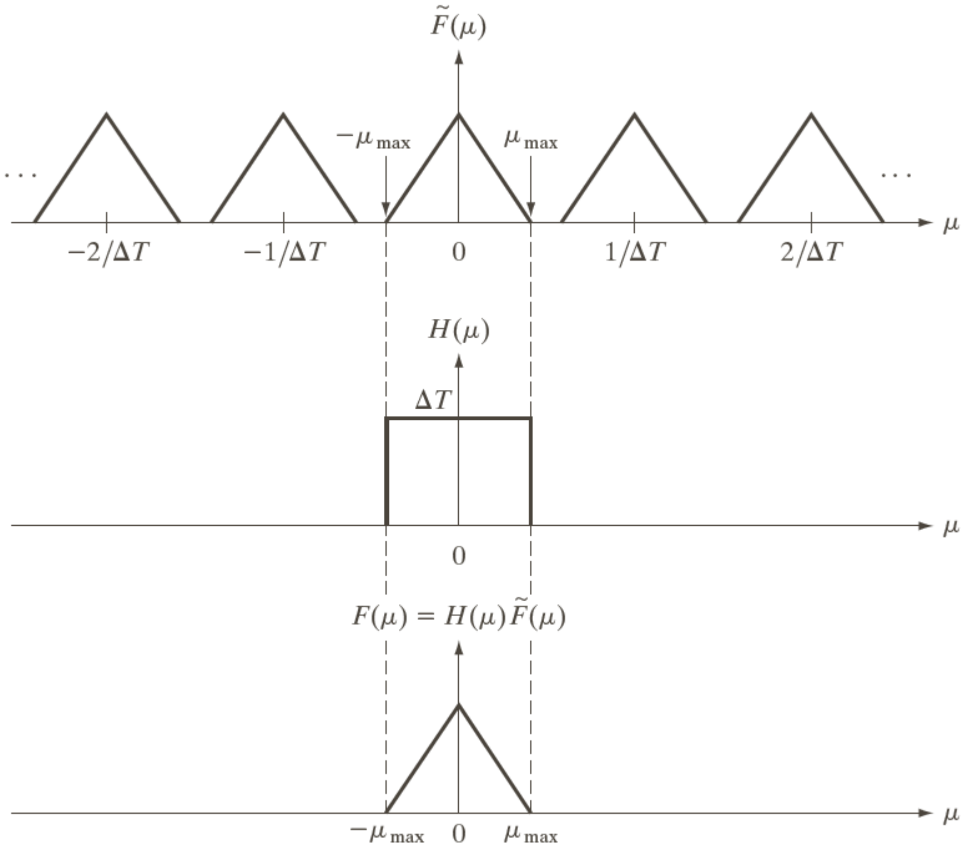
$$\frac{1}{\Delta T} > 2\mu_{max}$$

Sampling Theorem:

A continuous, band-limited function can be recovered completely from a set of its samples if the samples are acquired at a rate exceeding twice the highest frequency content of the function

NB: Sampling at: $\dfrac{1}{\Delta T} = 2\mu_{max}$ *Nyquist rate*





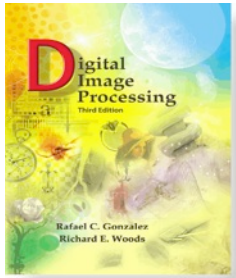
a
b
c

FIGURE 4.8 Extracting one period of the transform of a band-limited function using an ideal lowpass filter.

Example: ideal lowpass filter

=> Recover *f(t)* using the inverse FT:

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



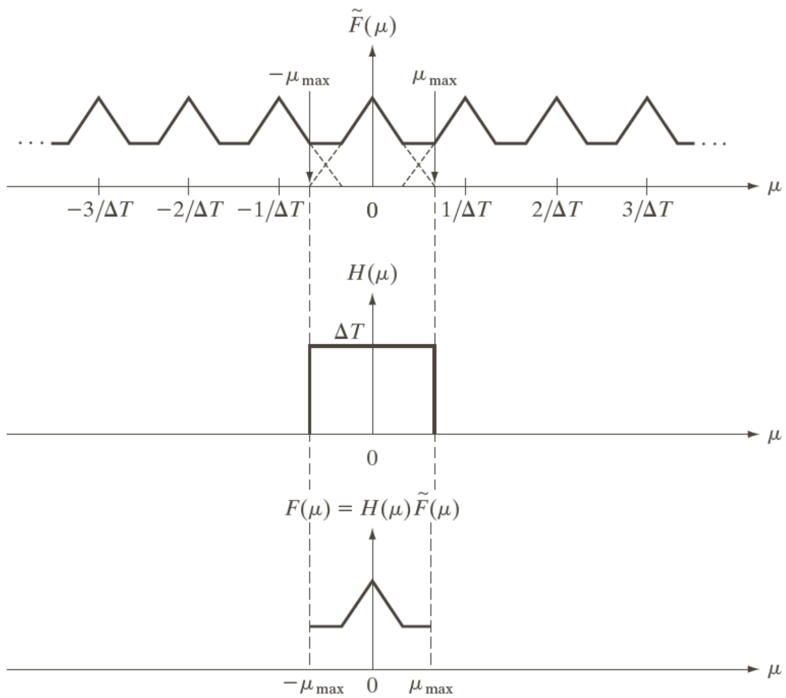
4.3.4 Aliasing

Effect of under-sampling a function

Transform corrupted by frequencies from adjacent periods

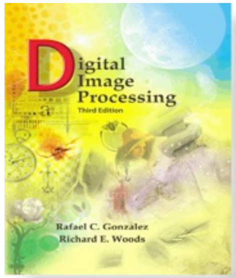
NB: the effect of aliasing can be reduced by smoothing the input function to attenuate its higher frequencies: *anti-aliasing*.

Has to be done *before* the sampling



a
b
c

FIGURE 4.9 (a) Fourier transform of an under-sampled, band-limited function. (Interference from adjacent periods is shown dashed in this figure). (b) The same ideal lowpass filter used in Fig. 4.8(b). (c) The product of (a) and (b). The interference from adjacent periods results in aliasing that prevents perfect recovery of $F(\mu)$ and, therefore, of the original, band-limited continuous function. Compare with Fig. 4.8.



4.3.4 Aliasing

Illustration: sampling a sine wave of period 2s

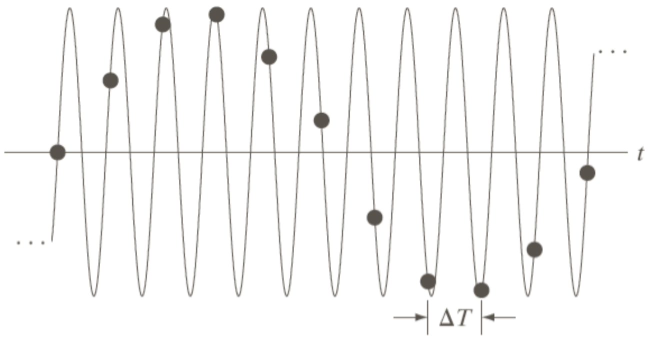


FIGURE 4.10 Illustration of aliasing. The under-sampled function (black dots) looks like a sine wave having a frequency much lower than the frequency of the continuous signal. The period of the sine wave is 2 s, so the zero crossings of the horizontal axis occur every second. ΔT is the separation between samples.

4.3.5 Function Reconstruction (Recovery) from Sampled Data

Recovery expressed in the spatial domain:

$$F(\mu) = H(\mu)\tilde{F}(\mu) \Rightarrow \quad f(t) = FT^{-1}\left[F(\mu)\right] \; = h(t) \bigstar \tilde{f}(t)$$

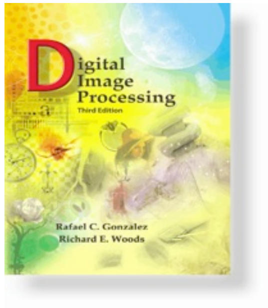
$$f(t) = \sum_{n=-\infty}^{\infty} f(n\Delta T)\, sinc\left[(t - n\Delta T)/\Delta T\right]$$

4.4 The Discrete Fourier Transform (DFT) of One Variable

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

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

f_n : discrete function



4.4 The Discrete Fourier Transform (DFT) of One Variable

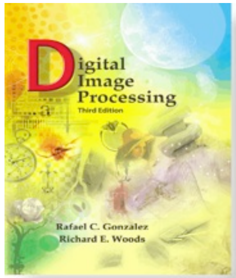
4.4.1 Obtaining the DFT from the Continuous Transform of a Sampled Function

Given a set $\{f(x)\}$ of M samples of $f(t)$ the DFT is of $f(x)$ is:

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

Inverse Discrete Fourier Transform (IDFT):

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



4.4 The Discrete Fourier Transform (DFT) of One Variable

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

Both forward and inverse discrete transforms are infinitely periodic, with period *M*

$$F(u) = F(u + kM)$$

$$f(x) = f(x + kM)$$

Where *k* is an integer

Discrete equivalent of the convolution:

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

4.4 The Discrete Fourier Transform (DFT) of One Variable

4.4.2 Relationship between the Sampling and Frequency Intervals

If $f(x)$ consists of M samples of a function $f(t)$ taken ΔT units apart, the duration of the record comprising the set $\{f(x)\}$ is:

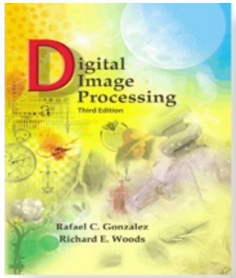
$$T = M\,\Delta T$$

The corresponding spacing in the discrete frequency domain is:

$$\Delta u = 1 / (M\,\Delta T) = 1/T$$

The entire frequency range spanned by the M components in the DFT is:

$$M\,\Delta u = 1/\,\Delta T$$



4.5 Extension to Function of Two Variables

4.5.1 The 2-D impulse and its Sifting Property

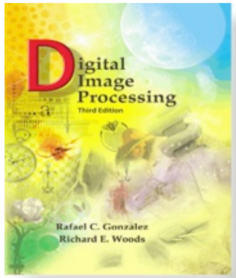
Impulse of two continuous variables *t* and *z* :

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

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

Impulse of two continuous variables *t* and *z* :

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



4.5 Extension to Function of Two Variables

4.5.1 The 2-D impulse and its Sifting Property

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

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

4.5 Extension to Function of Two Variables

4.5.1 The 2-D impulse and its Sifting Property

2-D discrete impulse:
$$\delta(x,y)=\begin{cases}1 & \text{if } x=y=0\\ 0 & \text{otherwise}\end{cases}$$

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

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

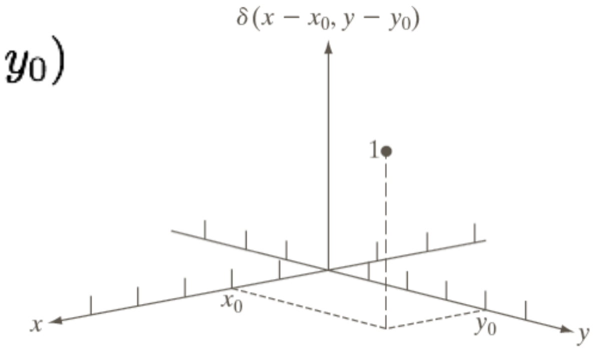


FIGURE 4.12 Two-dimensional unit discrete impulse. Variables x and y are discrete, and δ is zero everywhere except at coordinates (x_0, y_0).

4.5 Extension to Function of Two Variables

4.5.2 The 2-D Continuous Fourier Transform Pair

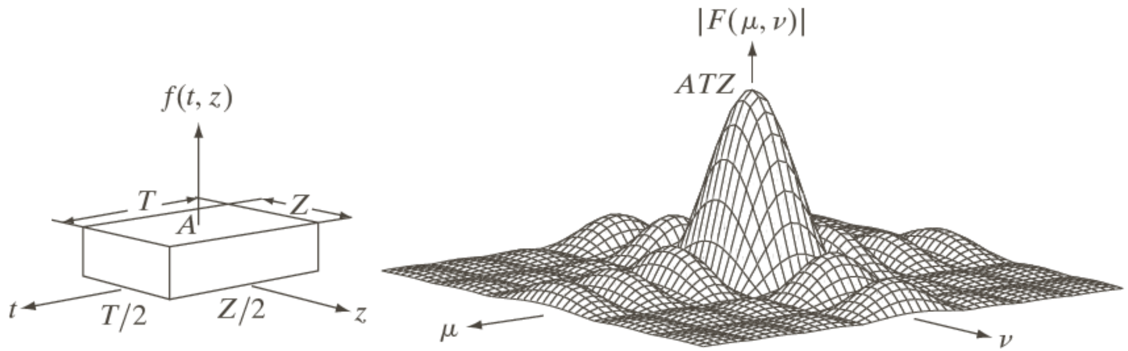
f(t,z) continuous function of two continuous variables *t* and *z*, the 2-D continuous Fourier transform pair is given by:

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

and

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

Example: Obtaining the 2-D Fourier transform of a simple function



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 μ-axis. Compare with Fig. 4.4.

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

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

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

Sampling in 2-D can be modeled using the sampling function (2-D impulse train):

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

Multiplying *f(t,z)* by $s_{\Delta T \Delta Z}(t,z)$ yields the sampled function

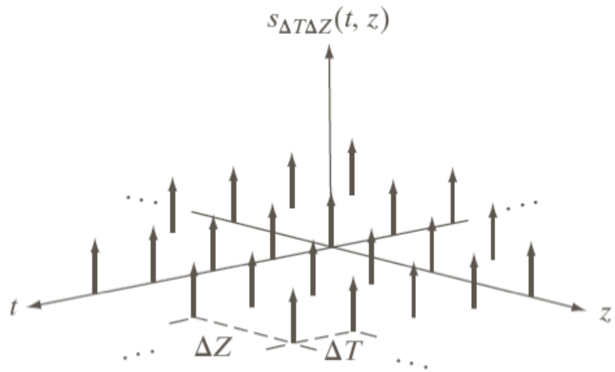
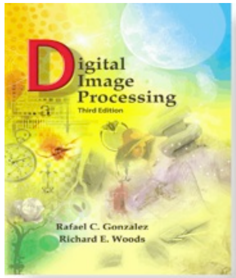


FIGURE 4.14 Two-dimensional impulse train.



f(t,z) is said to be *band-limited* if its Fourier transform is 0 outside of a rectangle $[-\mu_{max}, \mu_{max}]$ and $[-\nu_{max}, \nu_{max}]$

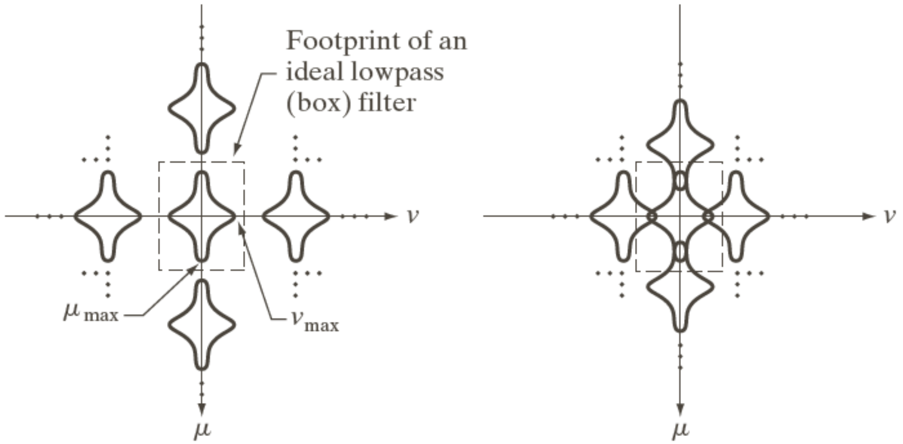
2-D sampling theorem:

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

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

and

$$\Delta Z < \frac{1}{2\nu_{max}}$$

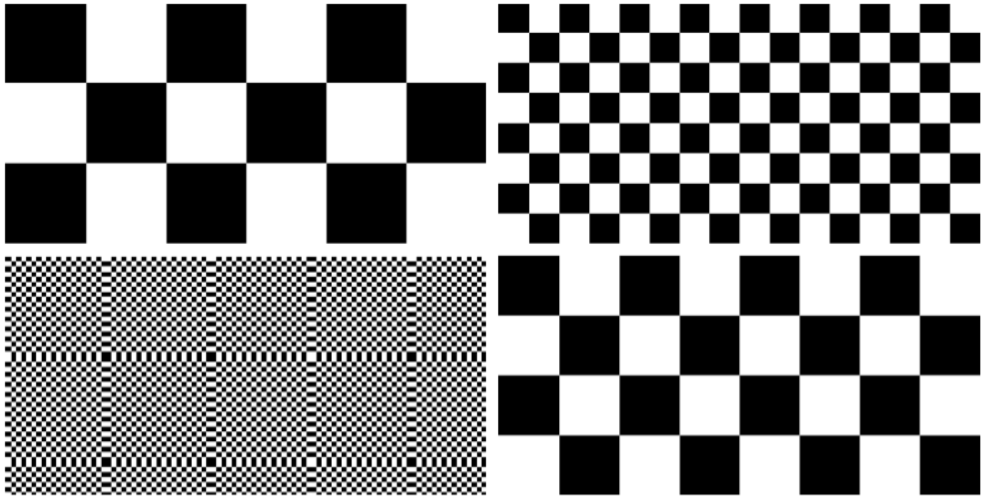


a b

FIGURE 4.15 Two-dimensional Fourier transforms of (a) an over-sampled, and (b) under-sampled band-limited function.

4.5.4 Aliasing in Images

Example: sampling checkerboards whose squares are of size 16x6 pixels



a b
c d

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.

Effects of aliasing can be reduced by slightly defocusing the scene to be digitized (attenuation of high frequencies), *before* the image is sampled : anti-aliasing

Effects of aliasing generally are worsened when the size of a digital image is reduced

Scaling: 50% Blurring prior to Scaling: 50%

a b c

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 3×3 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.)

Moiré patterns

In optics: beat patterns produced between two gratings of approximately equal spacing

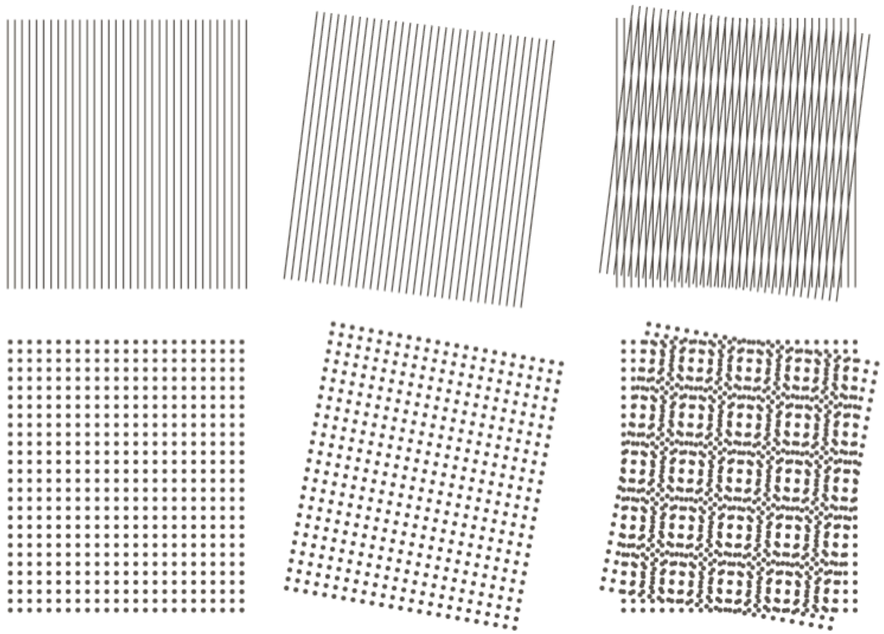
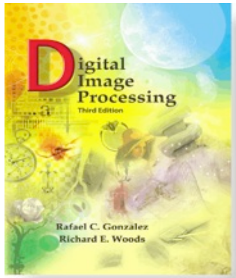


FIGURE 4.20 Examples of the moiré effect. These are ink drawings, not digitized patterns. Superimposing one pattern on the other is equivalent mathematically to multiplying the patterns.



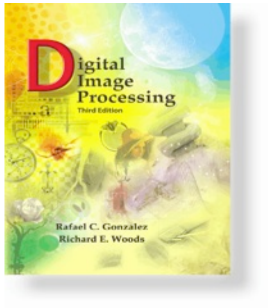
4.5.5 The 2-D Discrete Fourier Transform and its Inverse

2-D *Discrete Fourier Transform* (DFT) of a digital image *f(x,y)* of size *M*x*N*:

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

Inverse Discrete Fourier Transform (IDFT):

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

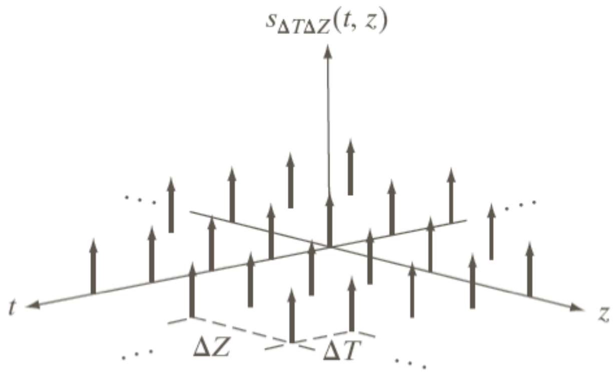


4.6 Some Properties of the 2-D Discrete Fourier Transform

4.6.1 Relationships between Spatial and Frequency Domains

If ΔT and ΔZ are the separations between samples, the separations between the corresponding discrete, frequency domain variables are:

$$\Delta u = \frac{1}{M\Delta T} \qquad \text{and} \qquad \Delta v = \frac{1}{N\Delta Z}$$

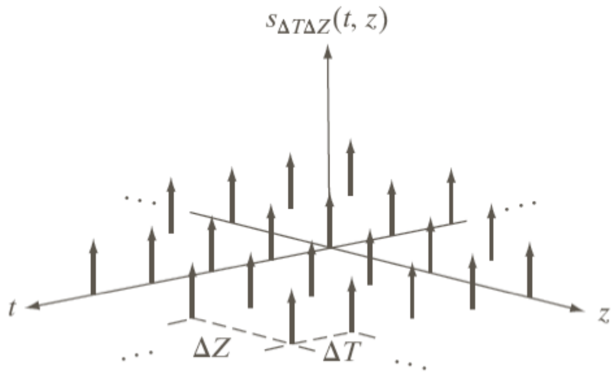


4.6.2 Translation and Rotation

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

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

Rotation (polar coordinates): $f(r, \theta+\theta_0) \Leftrightarrow F(\omega, \phi+\theta_0)$



4.6.3 Periodicity

The 2-D Fourier Transform and its inverse are infinitely periodic in both directions:

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

and

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

where k_1 and k_2 are integers

4.6.3 Periodicity

1-D Spectrum

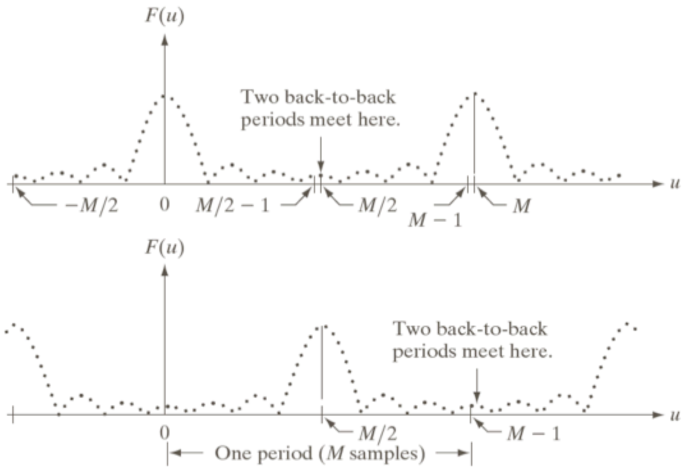
Shift the data so that the origin, $F(0)$, is located at u_0

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

If $u_0 = M/2$

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

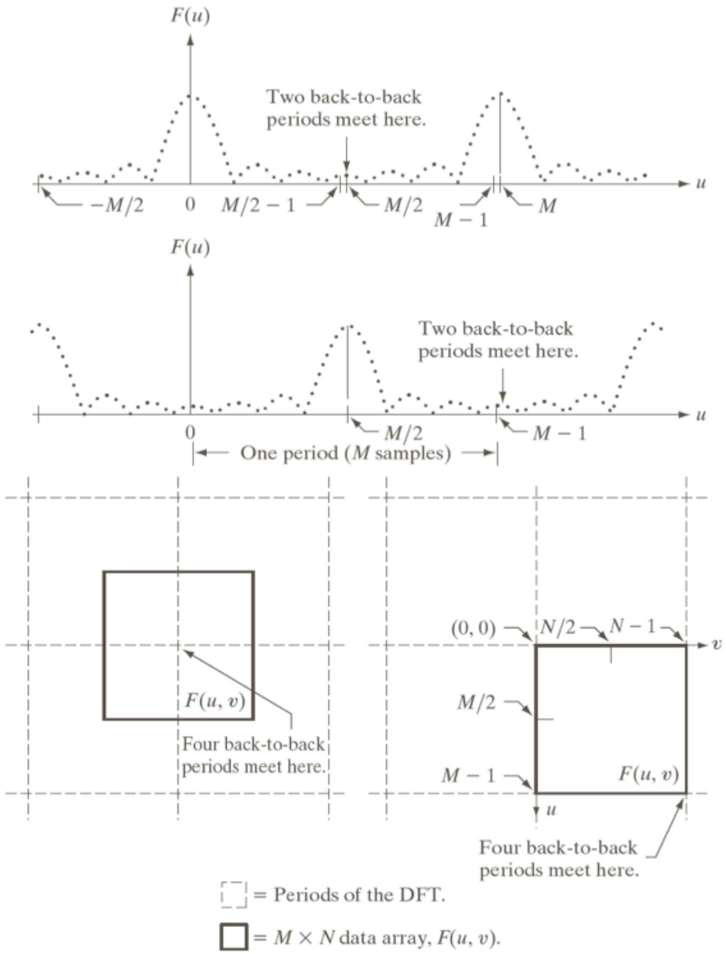
$F(0)$ at the centre of the interval $[0, M-1]$



a
b
c d

FIGURE 4.23 Centering the Fourier transform. (a) A 1-D DFT showing an infinite number of periods. (b) Shifted DFT obtained by multiplying $f(x)$ by $(-1)^x$ before computing $F(u)$.

4.6.3 Periodicity



a
b
c d

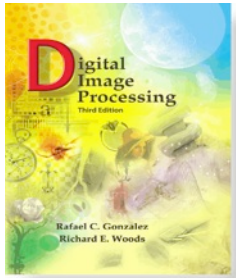
FIGURE 4.23 Centering the Fourier transform.

(c) A 2-D DFT showing an infinite number of periods. The solid area is the $M \times N$ data array, $F(u, v)$, obtained with Eq. (4.5-15). This array consists of four quarter periods. (d) A Shifted DFT obtained by multiplying $f(x, y)$ by $(-1)^{x+y}$ before computing $F(u, v)$. The data now contains one complete, centered period, as in (b).

2-D Spectrum

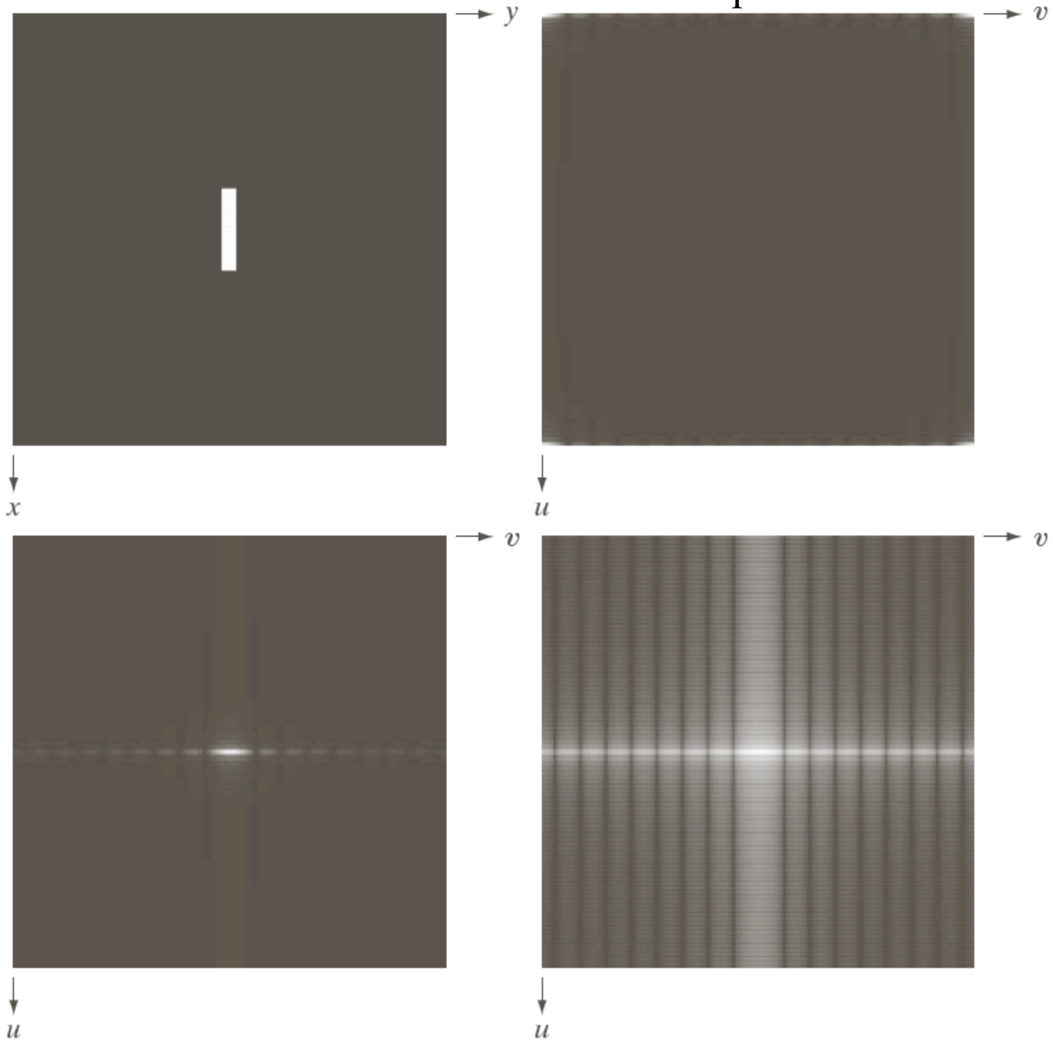
Shift the data so that the origin, *F(0,0)*, is at the centre of the *frequency rectangle* defined by [*0, M-1*] and [*0, N-1*]

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



FT Spectrum

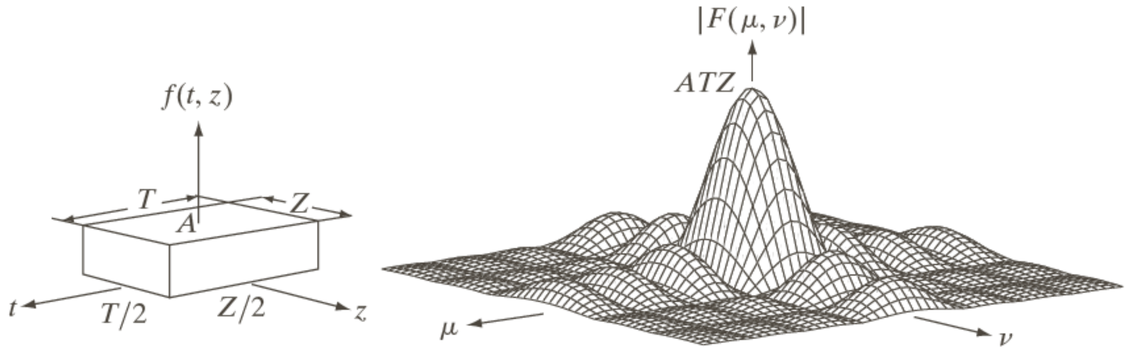
Centered
FT Spectrum



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.

(Reminder)

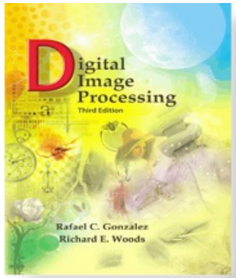


a b

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 μ-axis. Compare with Fig. 4.4.

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

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



4.6.4 Symmetry Properties

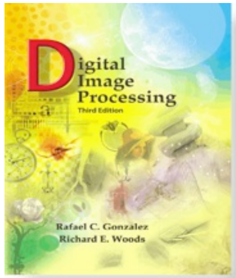
Example:

If *f(x,y)* is a *real* function, its Fourier transform is *conjugate symmetric*:

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

If *f(x,y)* is a *imaginary*, its Fourier transform is *conjugate antisymmetric*:

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



4.6.4 Symmetry Properties

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

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

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.

4.6.5 Fourier Spectrum and Phase Angle

The 2-D DFT can be expressed in polar form: $F(u,v) = |F(u,v)|e^{j\phi(u,v)}$

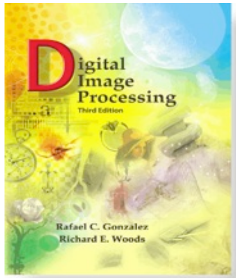
Where the magnitude $|F(u,v)| = \left[R^2(u,v) + I^2(u,v)\right]^{1/2}$

is called the *Fourier* (or *frequency*) *spectrum*, and

$$\phi(u,v) = \arctan\left[\frac{I(u,v)}{R(u,v)}\right]$$ is the *phase angle*

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

NB: *R* and *I* are the real and imaginary parts of *F(u,v)*



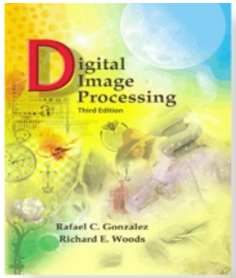
4.6.5 Fourier Spectrum and Phase Angle

$$|F(u,v)| = |F(-u,-v)| \qquad \text{and} \qquad \phi(u,v) = -\phi(-u,-v)$$

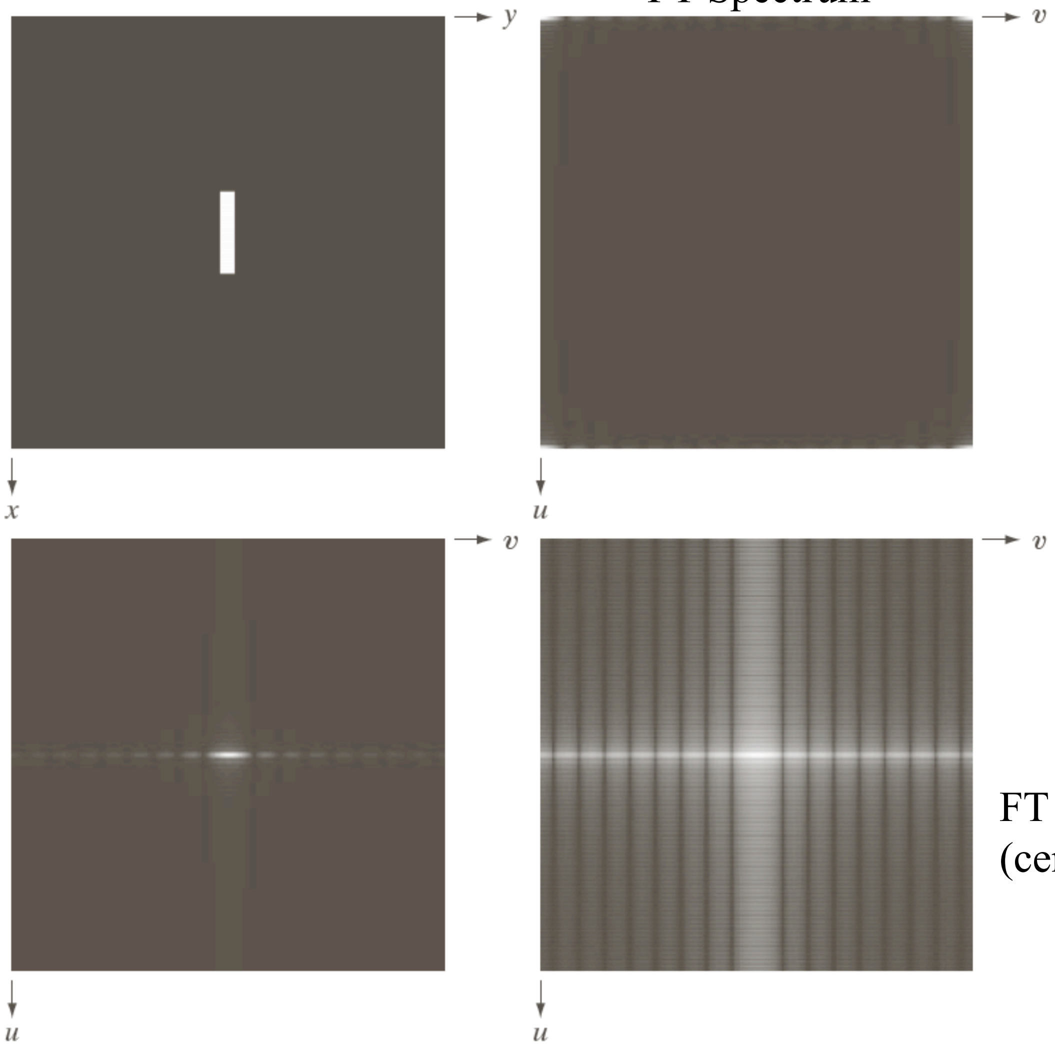
$$\Rightarrow \quad F(0,0) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y) \quad = MN\bar{f}(x,y)$$

Where $\bar{f}$ denotes the average value of f

MN usually large => |*F(0,0)*| typically is the largest component of the spectrum



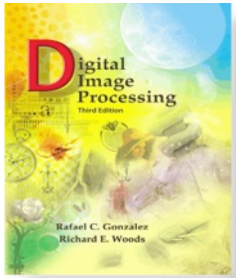
FT Spectrum



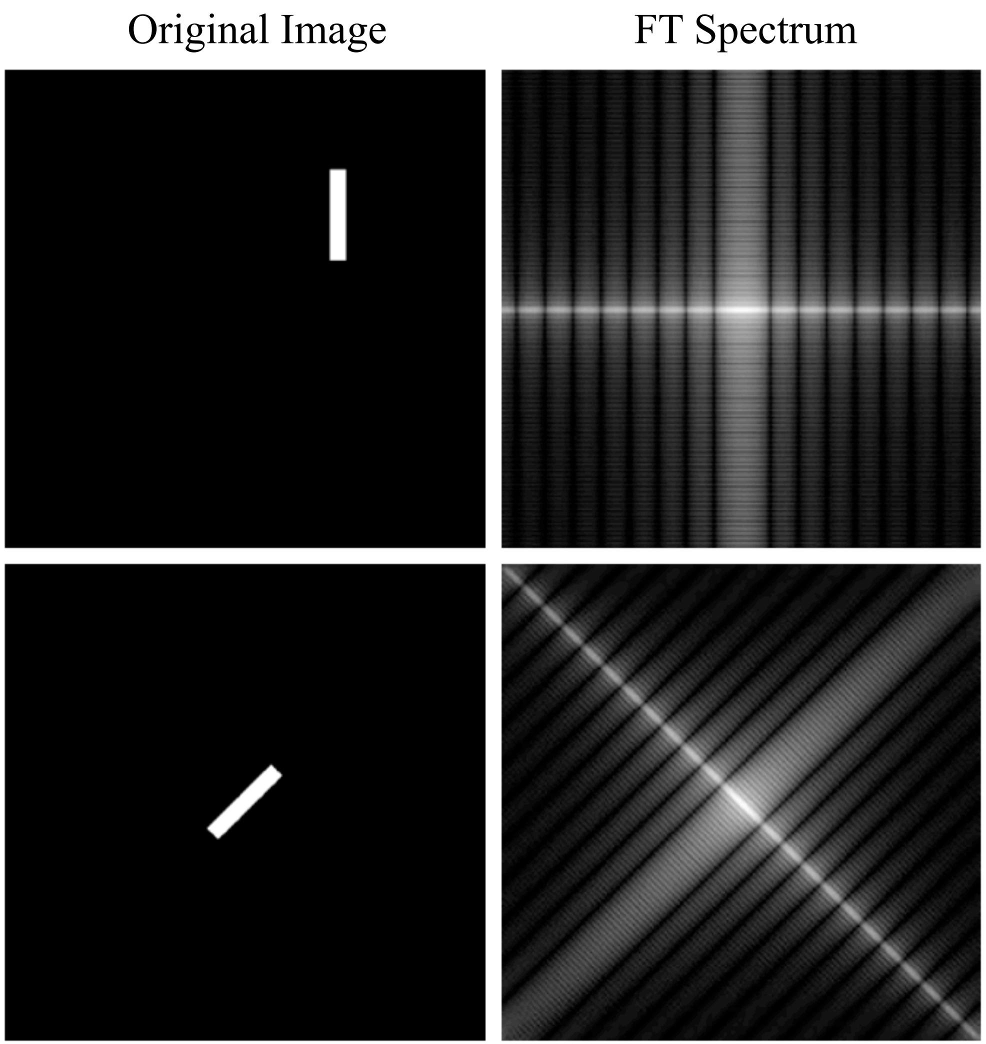
FT Spectrum (centered)

FT Spectrum (centered + log)

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.

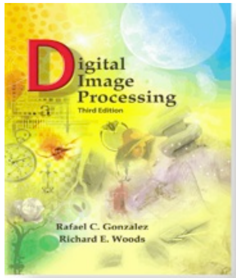


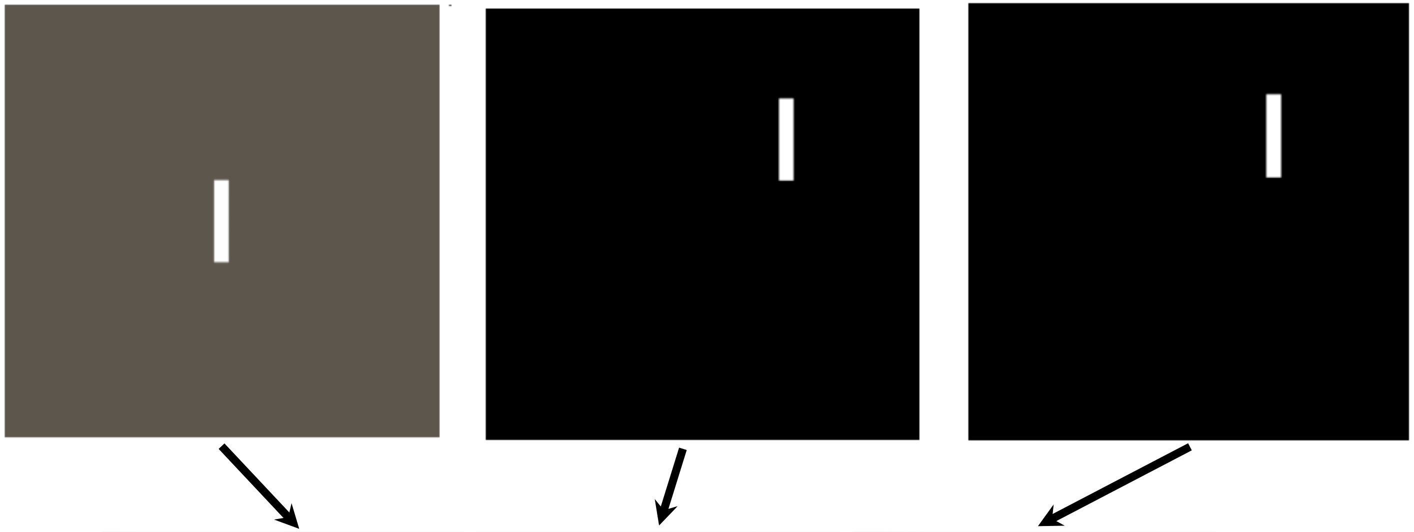
Spectrum is **insensitive to translation**
But **rotates by the same angle** as the rotated image



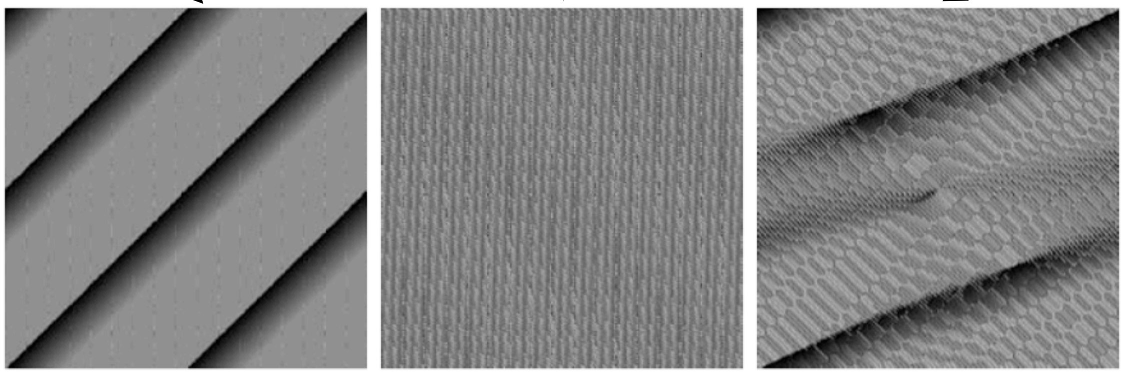
a b
c d

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).



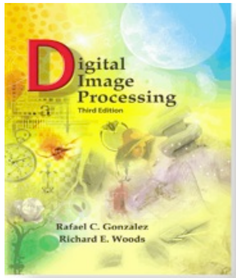


Phase angle images:

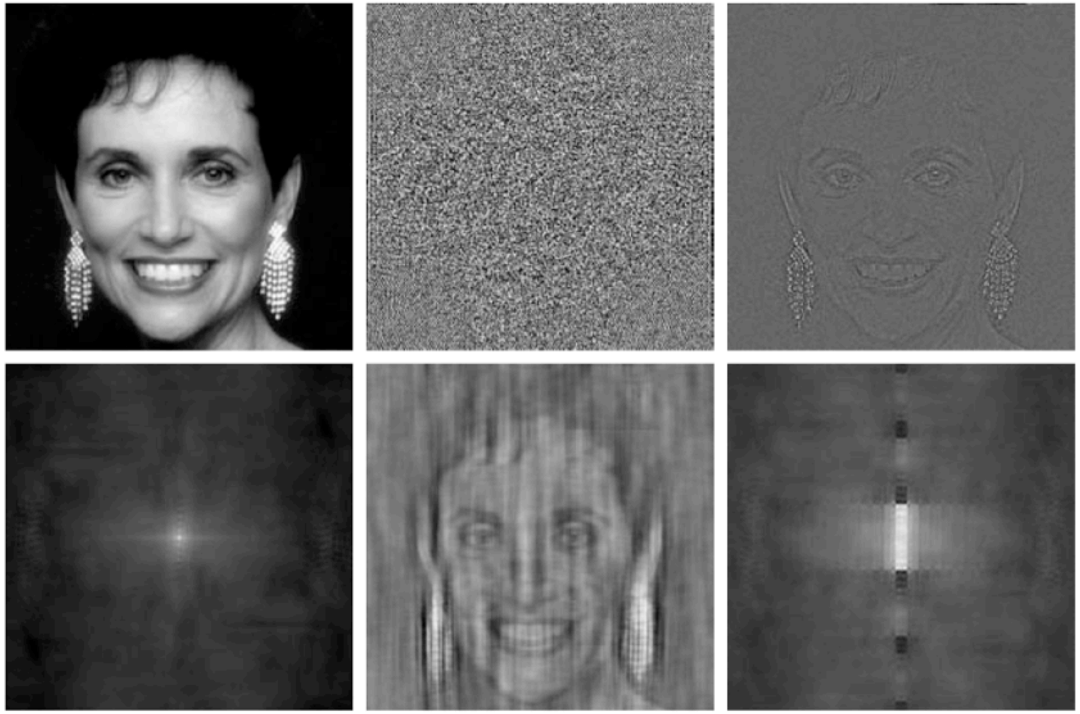


a b c

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).



Phase angle



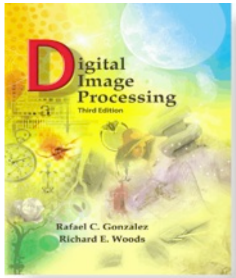
Reconstruction using phase angle only

Reconstruction using spectrum only

Reconstruction using:
- Right Spectrum
- Wrong Phase

a b c
d e f

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



4.6.6 The 2-D Convolution Theorem

2-D *circular convolution* :

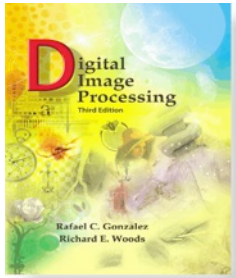
$$f(x,y) \bigstar h(x,y) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)$$

2-D convolution theorem :

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

and conversely:

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



Chapter 4
Filtering in the Frequency Domain

Data from adjacent periods produce wraparound error
=> Need to append zeros to both functions

In 2-D, padd the two images array of size

MxN by zeros

Padded images of size PxQ, with:

$$P \geq 2M\text{-}1$$

and

$$Q \geq 2N\text{-}1$$

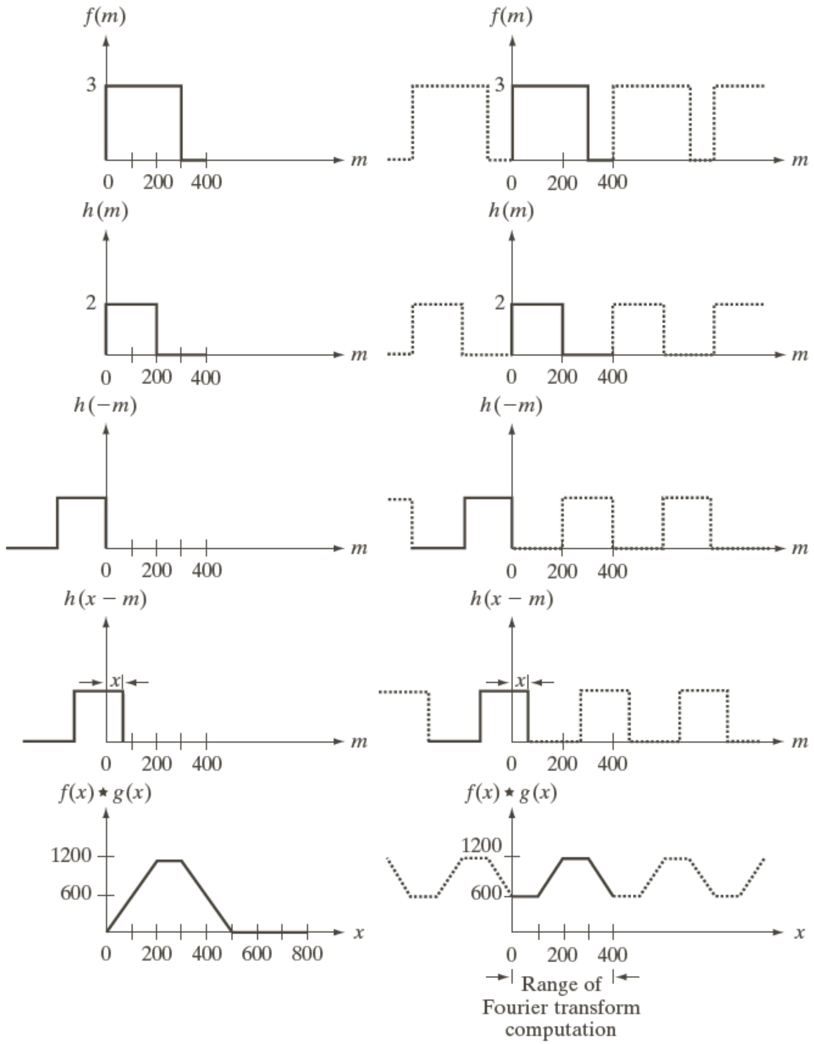
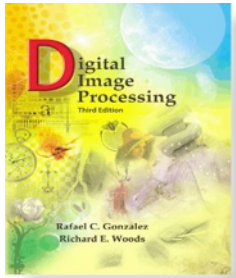


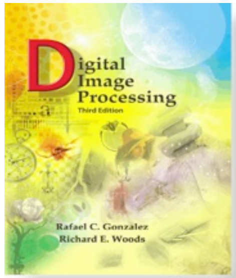
FIGURE 4.28 Left column: convolution of two discrete functions obtained using the approach discussed in Section 3.4.2. The result in (e) is correct. Right column: Convolution of the same functions, but taking into account the periodicity implied by the DFT. Note in (j) how data from adjacent periods produce wraparound error, yielding an incorrect convolution result. To obtain the correct result, function padding must be used.



4.6.7 Summary of 2-D Discrete Fourier Transform Properties

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

(Continued)



4.6.7 Summary of 2-D Discrete Fourier Transform Properties

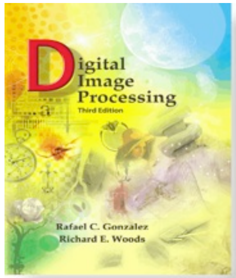
Name	Expression(s)
8) Periodicity (k_1 and k_2 are integers)	$F(u,v) = F(u+k_1M, v) = F(u, v+k_2N) = F(u+k_1M, v+k_2N)$ $f(x,y) = f(x+k_1M, y) = f(x, y+k_2N) = f(x+k_1M, y+k_2N)$
9) Convolution	$f(x,y)\bigstar h(x,y) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m, y-n)$
10) Correlation	$f(x,y)\star h(x,y)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

4.6.7 Summary of 2-D Discrete Fourier Transform Properties

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

(*Continued*)

TABLE 4.3
Summary of DFT pairs. The closed-form expressions in 12 and 13 are valid only for continuous variables. They can be used with discrete variables by sampling the closed-form, continuous expressions.

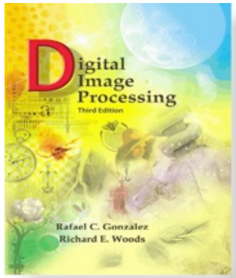


4.6.7 Summary of 2-D Discrete Fourier Transform Properties

TABLE 4.3
(Continued)

	Name	DFT Pairs
7)	Correlation theorem[†]	$f(x,y) \star h(x,y) \Leftrightarrow F^*(u,v)\,H(u,v)$ $f^*(x,y)h(x,y) \Leftrightarrow F(u,v) \star H(u,v)$
8)	Discrete unit impulse	$\delta(x,y) \Leftrightarrow 1$
9)	Rectangle	$\mathrm{rect}[a,b] \Leftrightarrow ab\dfrac{\sin(\pi ua)}{(\pi ua)}\dfrac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
10)	Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $j\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) - \delta(u - Mu_0, v - Nv_0)\right]$
11)	Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) + \delta(u - Mu_0, v - Nv_0)\right]$
	The following Fourier transform pairs are derivable only for continuous variables, denoted as before by t and z for spatial variables and by μ and ν for frequency variables. These results can be used for DFT work by sampling the continuous forms.	
12)	*Differentiation* (The expressions on the right assume that $f(\pm\infty, \pm\infty) = 0$.)	$\left(\dfrac{\partial}{\partial t}\right)^m\left(\dfrac{\partial}{\partial z}\right)^n f(t,z) \Leftrightarrow (j2\pi\mu)^m(j2\pi\nu)^n F(\mu,\nu)$ $\dfrac{\partial^m f(t,z)}{\partial t^m} \Leftrightarrow (j2\pi\mu)^m F(\mu,\nu);\ \dfrac{\partial^n f(t,z)}{\partial z^n} \Leftrightarrow (j2\pi\nu)^n F(\mu,\nu)$
13)	*Gaussian*	$A2\pi\sigma^2 e^{-2\pi^2\sigma^2(t^2+z^2)} \Leftrightarrow Ae^{-(\mu^2+\nu^2)/2\sigma^2}$ (A is a constant)

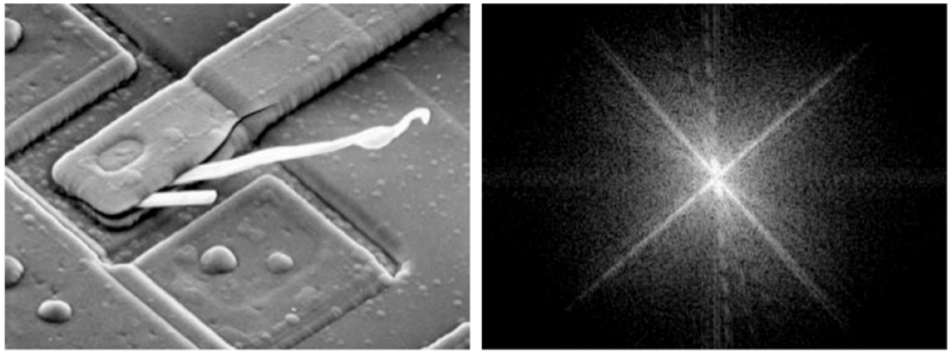
[†]Assumes that the functions have been extended by zero padding. Convolution and correlation are associative, commutative, and distributive.



4.7 The Basics of Filtering in the Frequency Domain

4.7.1 Additional Characteristics of the Frequency Domain

FT Spectrum



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.)

4.7.2 Frequency Domain Filtering Fundamentals

Given a digital image *f(x,y)* of size *M*x*N*, the basic filtering equation is:

$$g(x,y) = IDFT\left[H(u,v)F(u,v)\right]$$

NB: product = array multiplication

Filtered (output) image

Filter function (or filter transfer function)

DFT if the input image

NB: For simplification, use functions *H(u,v)* that are centered symmetric about their centre
+ centre *F(u,v)* multiplying *f(x,y)* by $(-1)^{x+y}$

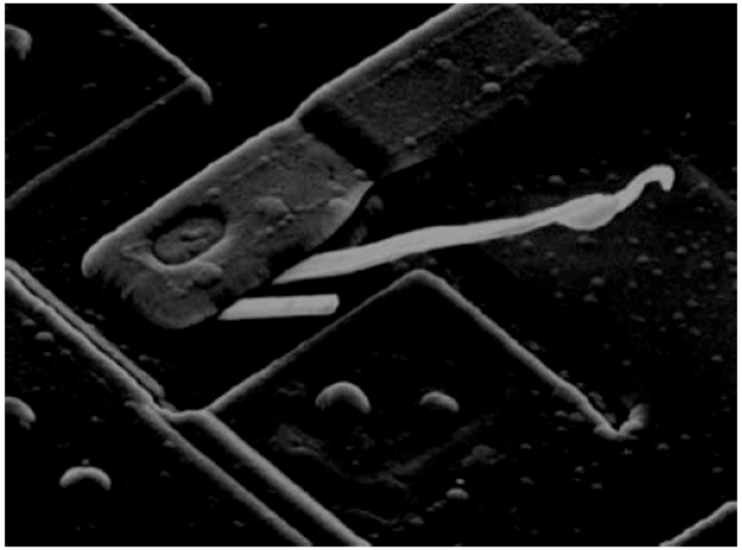
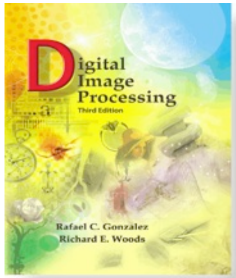


FIGURE 4.30 Result of filtering the image in Fig. 4.29(a) by setting to 0 the term $F(M/2, N/2)$ in the Fourier transform.



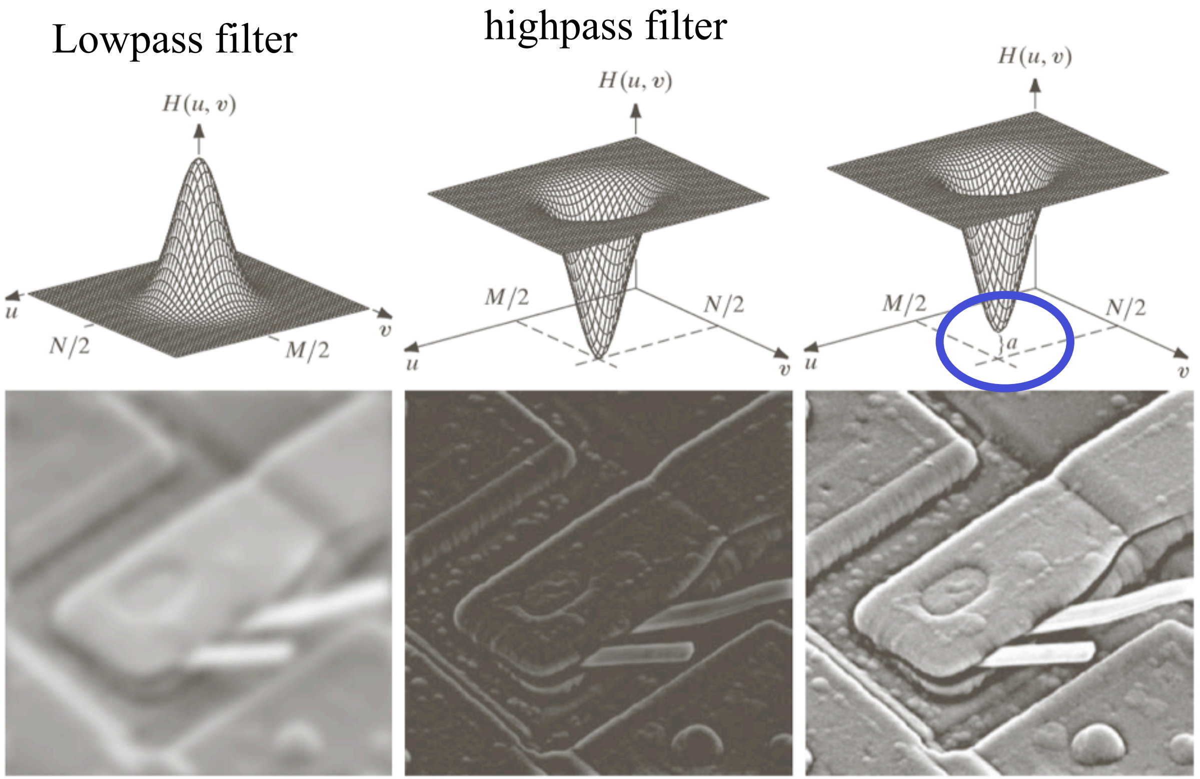
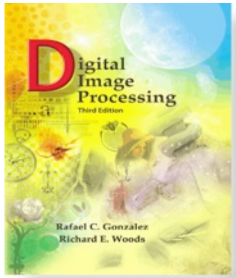
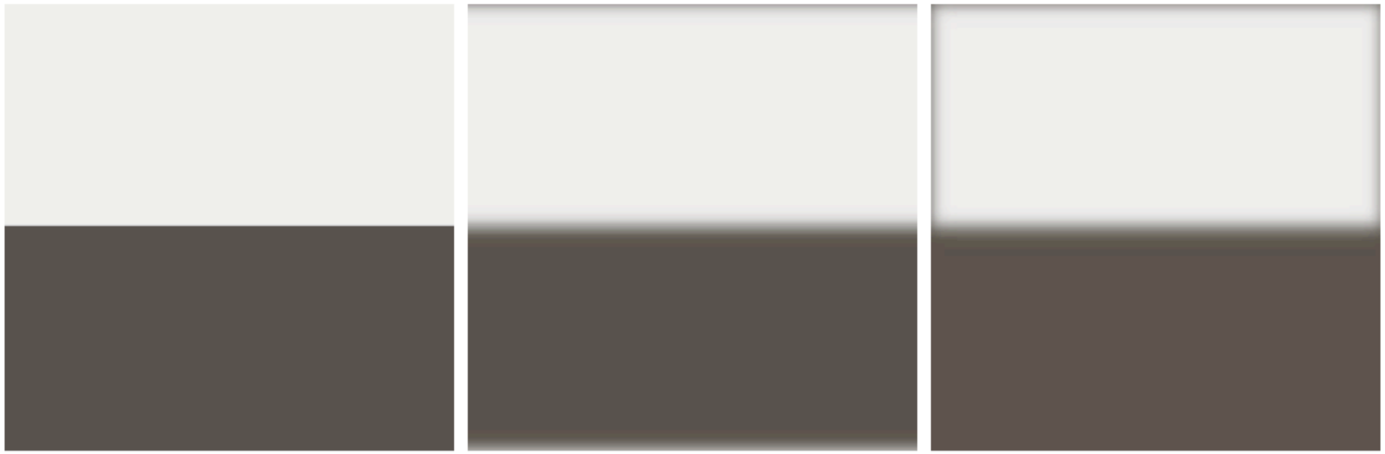


FIGURE 4.31 Top row: frequency domain filters. Bottom row: corresponding filtered images obtained using Eq. (4.7-1). We used $a = 0.85$ in (c) to obtain (f) (the height of the filter itself is 1). Compare (f) with Fig. 4.29(a).



Gaussian lowpass filter **Without** padding

Gaussian lowpass filter **With** padding

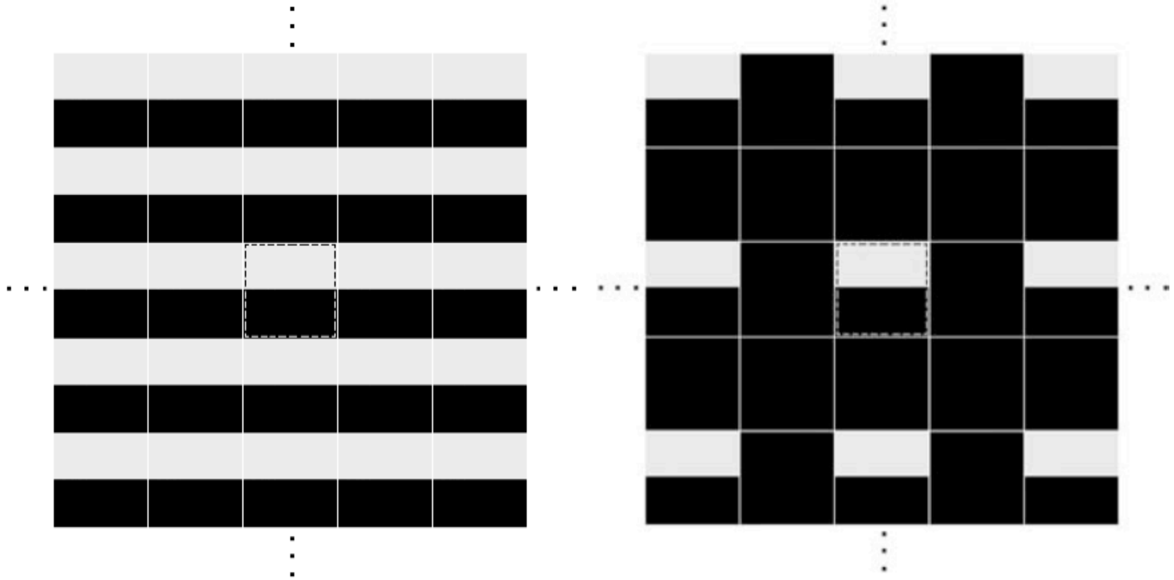


a b c

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).

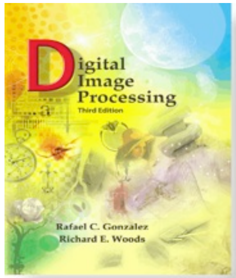
Periodicity **without** padding

Periodicity **without** padding



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.)



4.7.3 Steps for Filtering in the Frequency Domain

1. Given an input image $f(x,y)$ of size MxN, obtain padding parameters P and Q. Typically, $P=2M$ and $Q=2N$.
2. Form a padded image $f_p(x,y)$ of size PxQ by appending the necessary number of zeros to f(x,y).
3. Multiply $f_p(x,y)$ by $(-1)^{x+y}$ to centre its transform.
4. Compute the DFT, $F(u,v)$, of the image from step 3.
5. Generate a real, symmetric filter function, $H(u,v)$, of size PxQ with centre at coordinates $(P/2,Q/2)$. Form the product $G(u,v)=H(u,v)F(u,v)$ using array multiplication.
6. Obtain the processed image: $g_p(x,y) = \text{real}\left[IDFT\left[G(u,v)\right]\right](-1)^{x+y}$
7. Obtain the final processed result, $g(x,y)$, by extracting the MxN region from the top, left quadrant of $g_p(x,y)$

4.7.3 Steps for Filtering in the Frequency Domain

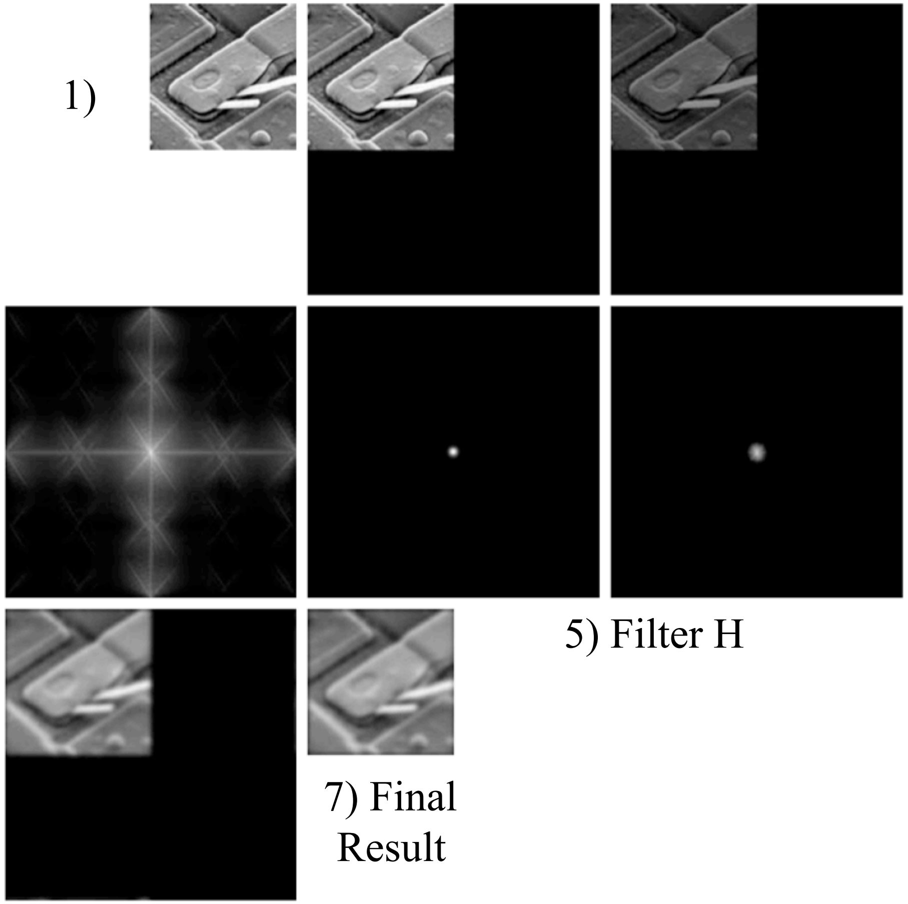
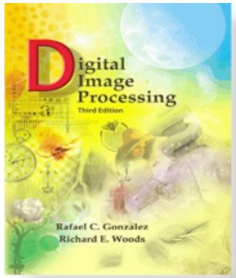


FIGURE 4.36 (a) An $M \times N$ image, f. (b) Padded image, f_p of size $P \times Q$. (c) Result of multiplying f_p by $(-1)^{x+y}$. (d) Spectrum of F_p. (e) Centered Gaussian lowpass filter, H, of size $P \times Q$. (f) Spectrum of the product HF_p. (g) g_p, the product of $(-1)^{x+y}$ and the real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



4.8 Image Smoothing using Frequency Domain Filters

NB: all filter functions are assumed to be discrete functions of size *P*x*Q*

4.8.1 Ideal Lowpass Filters

Ideal Lowpass Filter (ILPF) = 2-D lowpass filter that passes without attenuation all frequencies within a circle of radius D_0 from the origin and "cuts off" all frequencies outside this circle

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

Where: $D_0 \geq 0$

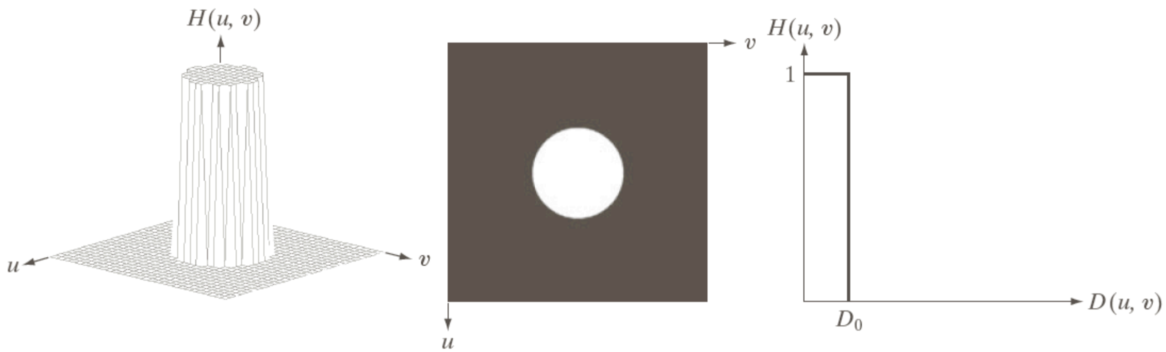
And *D(u,v)* is the distance between a point *(u,v)* and the centre of the frequency rectangle:

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

4.8.1 Ideal Lowpass Filters

The point of transition between $H(u,v) = 1$ and $H(u,v) = 0$ is called the *cutoff frequency*

NB: an ILPF cannot be realized with real electronic components

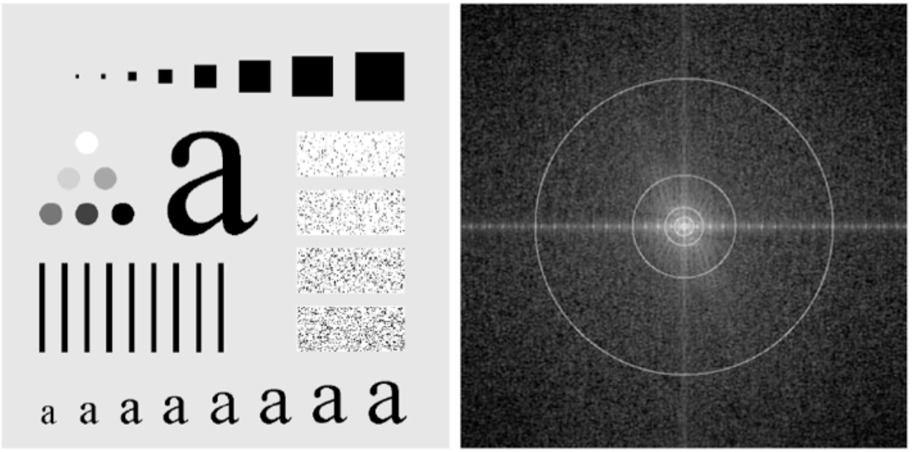


a b c

FIGURE 4.40 (a) Perspective plot of an ideal lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

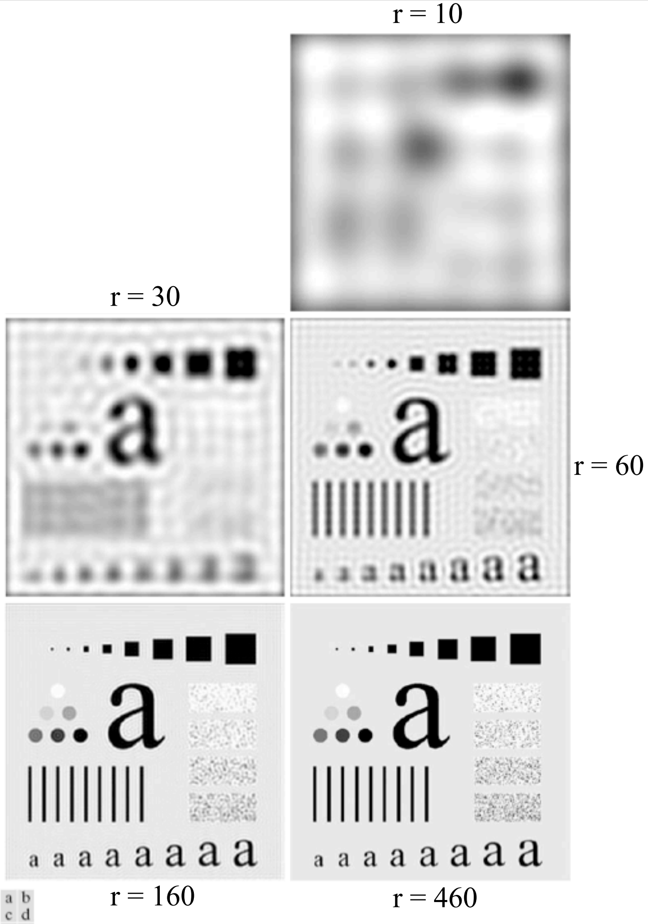
4.8.1 Ideal Lowpass Filters

Example:



a b

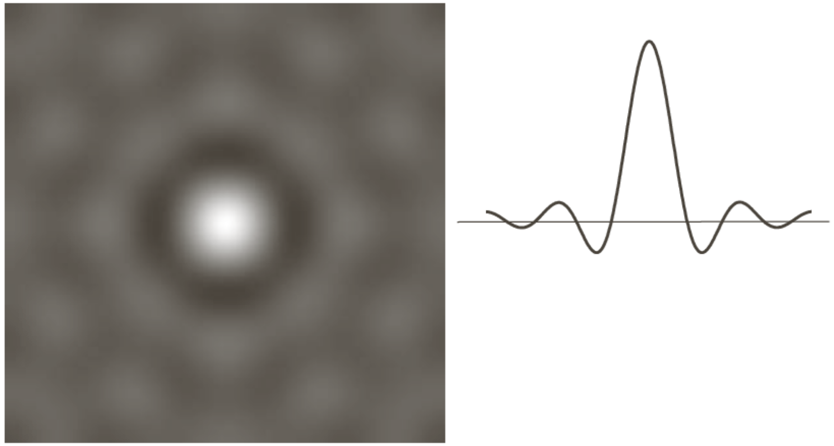
FIGURE 4.41 (a) Test pattern of size 688 × 688 pixels, and (b) its Fourier spectrum. The spectrum is double the image size due to padding but is shown in half size so that it fits in the page. The superimposed circles have radii equal to 10, 30, 60, 160, and 460 with respect to the full-size spectrum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



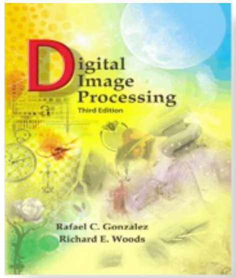
a b
c d
e f

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.

Explaining the blurring and "ringing" properties of ILPFs



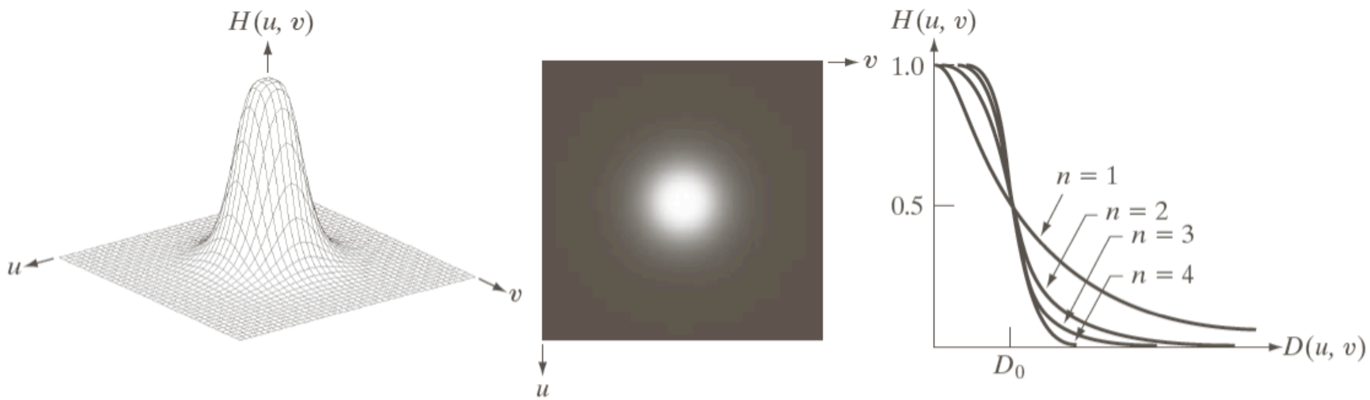
a b
FIGURE 4.43 (a) Representation in the spatial domain of an ILPF of radius 5 and size 1000×1000. (b) Intensity profile of a horizontal line passing through the center of the image.



4.8.2 Butterworth Lowpass Filters

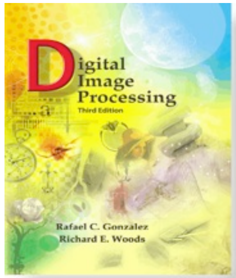
Transfer function of a Butterworth lowpass filter (BLPF) of order n and with cutoff frequency at a distance D_0 from the origin:

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

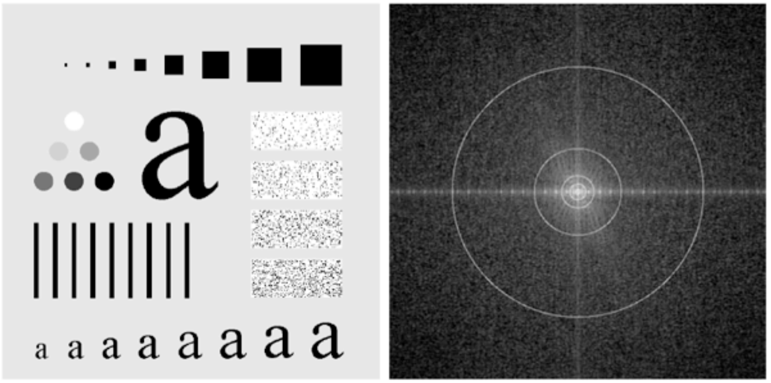


a b c

FIGURE 4.44 (a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.



4.8.2 Butterworth Lowpass Filters



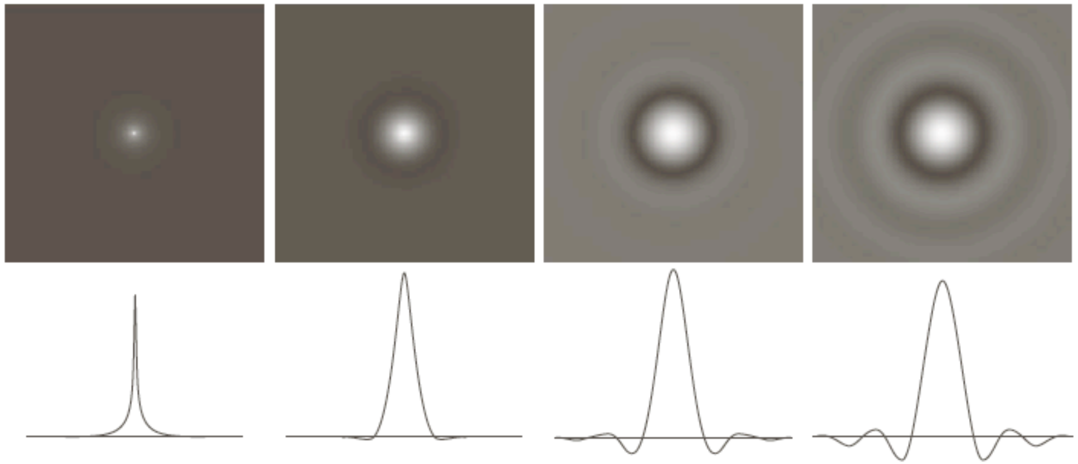
a b

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

a b
c d
e f

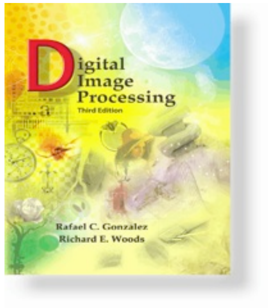
FIGURE 4.45 (a) Original image. (b)–(f) Results of filtering using BLPFs of order 2, with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Fig. 4.42.

4.8.2 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 × 1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.



4.8.3 Gaussian Lowpass Filters

Gaussian Lowpass Filters (GLPFs) in two-dimensions:

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2}$$ (σ = measure of spread about the centre)

$$\sigma = D_0 \;\Rightarrow\; H(u,v) = e^{-D^2(u,v)/2D_0^2}$$ (D_0 = cutoff frequency)

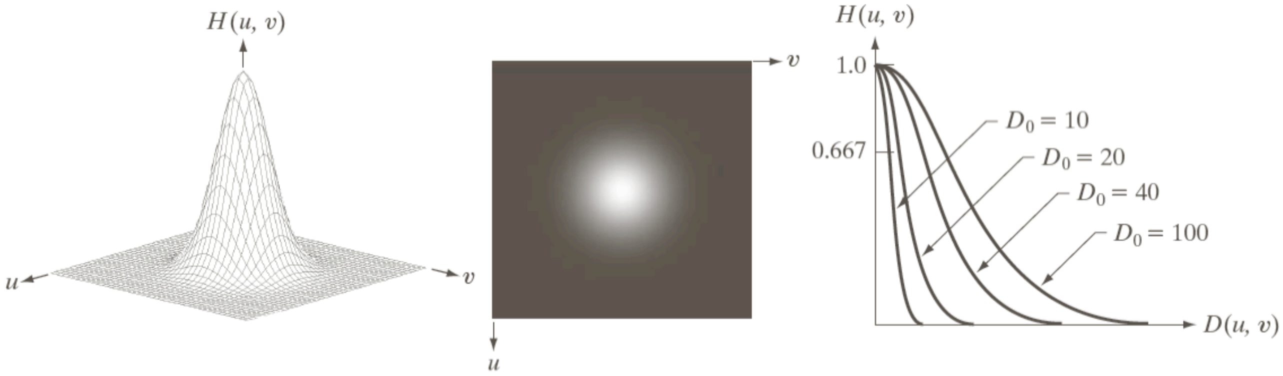
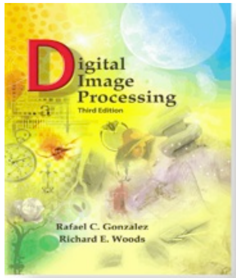
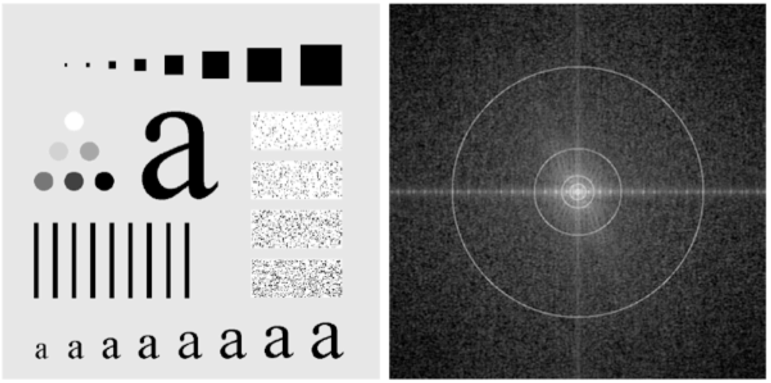


FIGURE 4.47 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

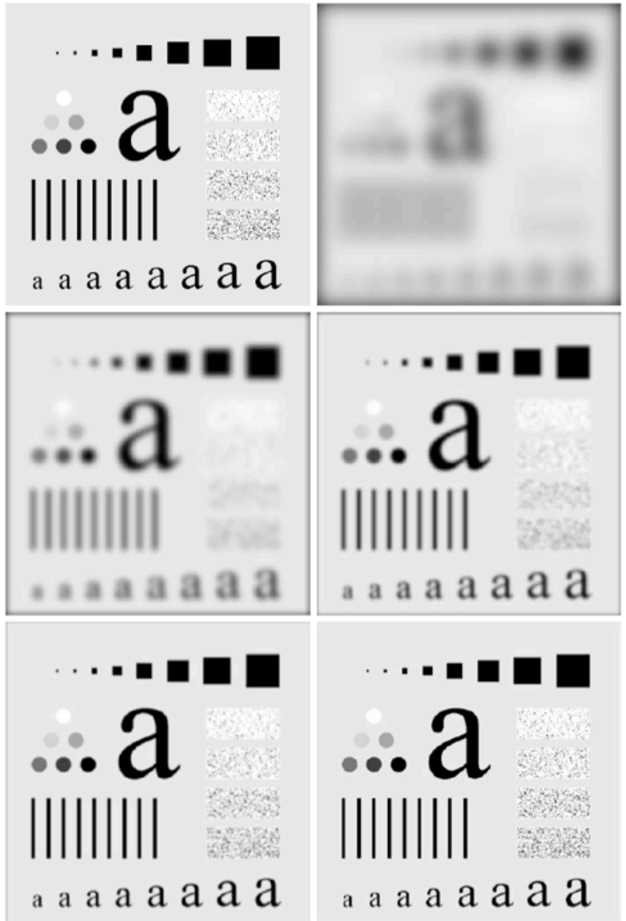


4.8.3 Gaussian Lowpass Filters



a b

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



a b
c d
e f

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

TABLE 4.4

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

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

4.8.4 Additional Examples of Lowpass Filtering

Character recognition (machine perception):

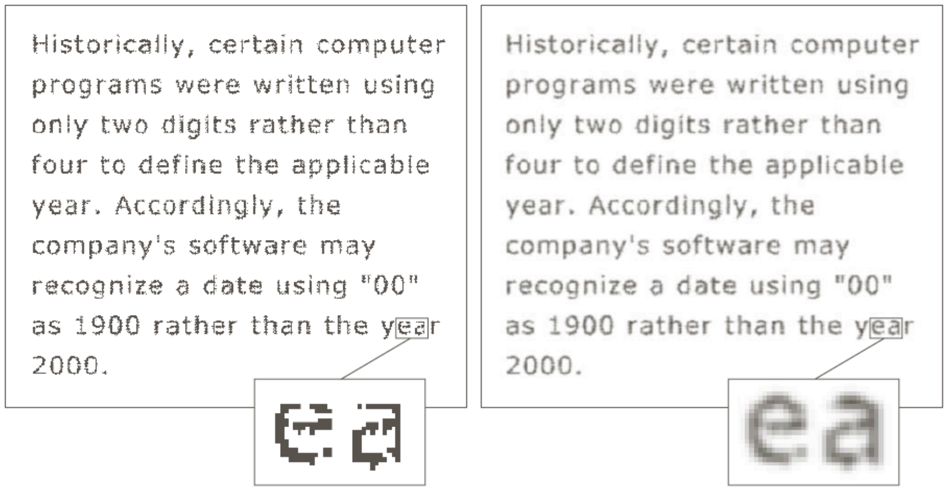


FIGURE 4.49 (a) Sample text of low resolution (note broken characters in magnified view). (b) Result of filtering with a GLPF (broken character segments were joined).

Blurring to fill "visual gaps" => help reading broken characters

Printing and publishing industry: “cosmetic” processing

Original Image

GLPF (Do=100)

GLPF (Do=80)

a b c

FIGURE 4.50 (a) Original image (784 × 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).

4.9 Image Sharpening using Frequency Domain Filters

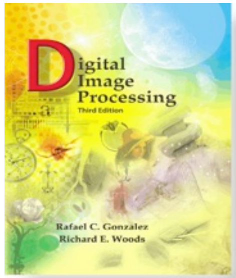
Highpass filtering: attenuation of the high-frequency components of the Fourier transform of the image

As before :

- Radially symmetric filters
- All filter functions are assumed to be discrete functions of size *P*x*Q*

A highpass H_{HP} filter can be obtained from a given lowpass H_{LP} filter by:

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



4.9.1 Ideal Highpass Filters

A 2-D *Ideal Highpass Filter* (IHPF) is defined as:

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

Chapter 4
Filtering in the Frequency Domain

Ideal Highpass Filter:

Butterworth Highpass Filter:

Gaussian Highpass Filter:

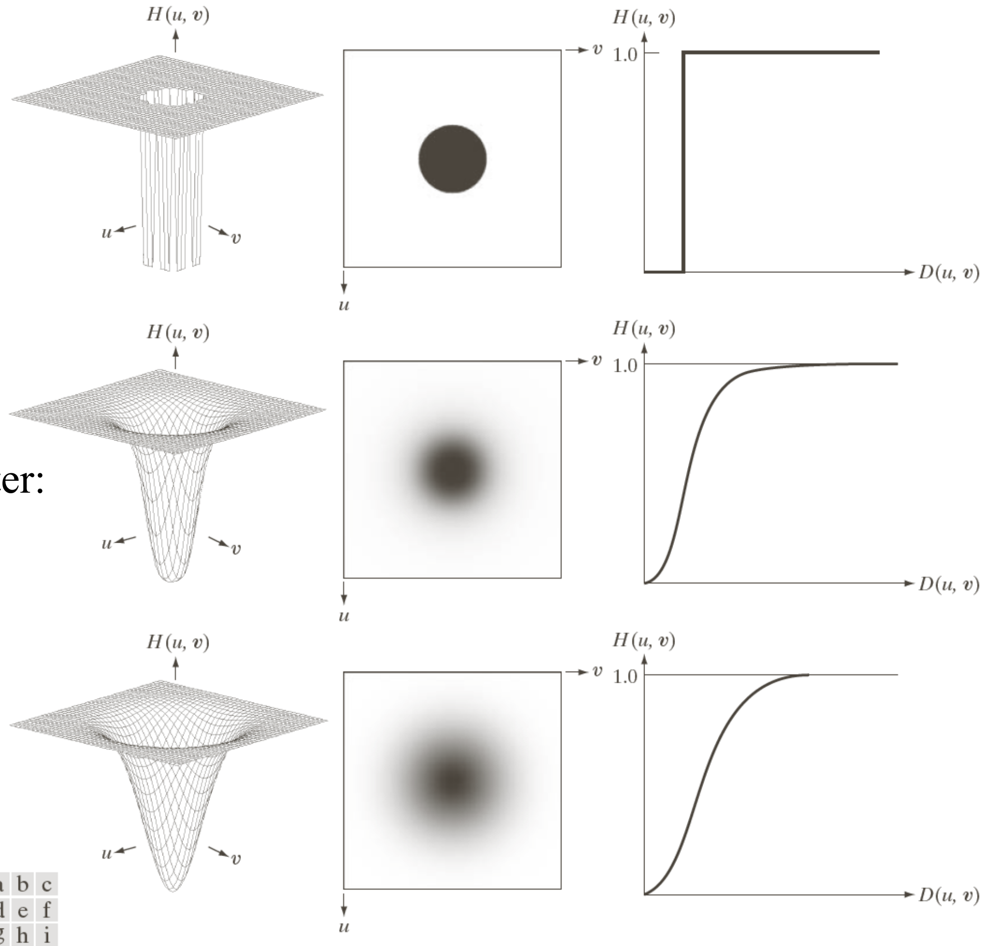


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

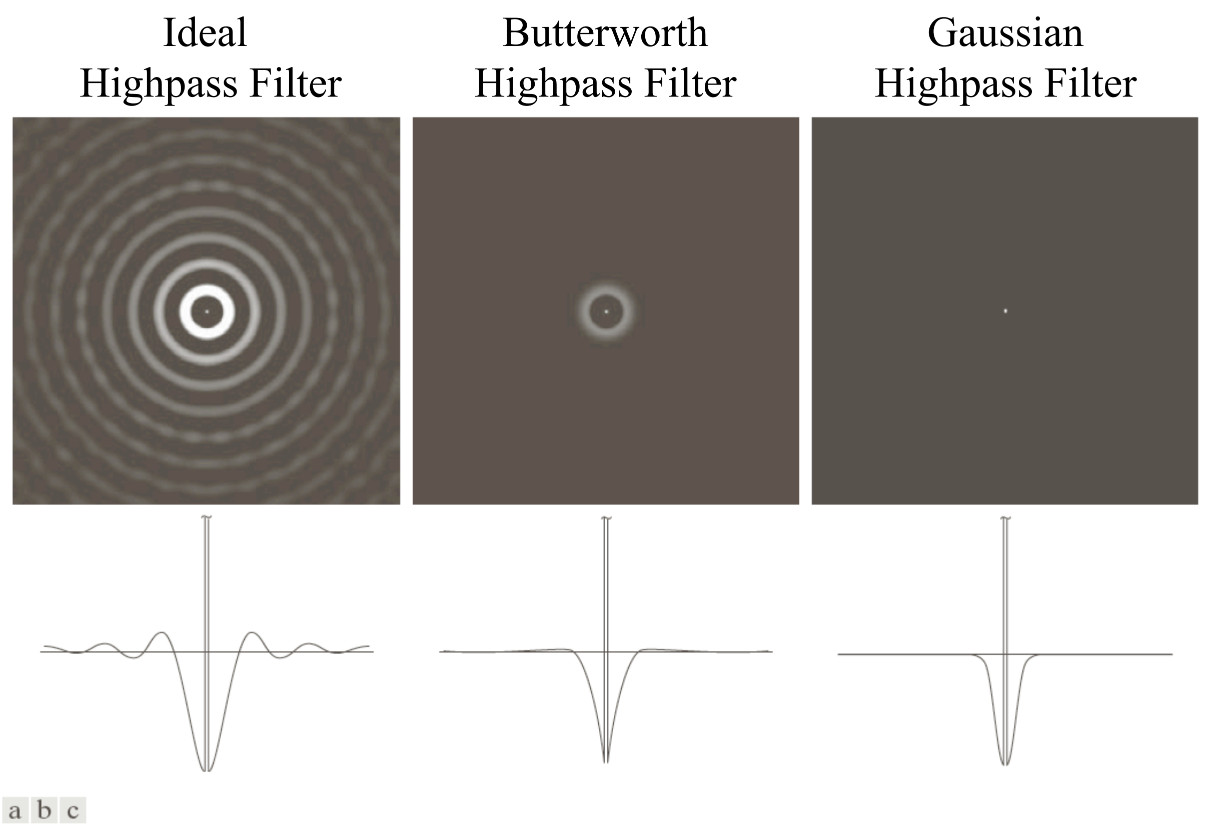


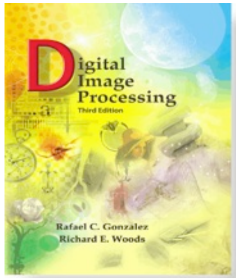
FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

4.9.1 Ideal Highpass Filters

Do = 30 Do = 60 Do = 160

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.



4.9.2 Butterworth Highpass Filters

A 2-D *Butterworth Highpass Filter* (BHPF) of order n and cutoff frequency D_0 is defined as:

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

4.9.2 Butterworth Highpass Filters

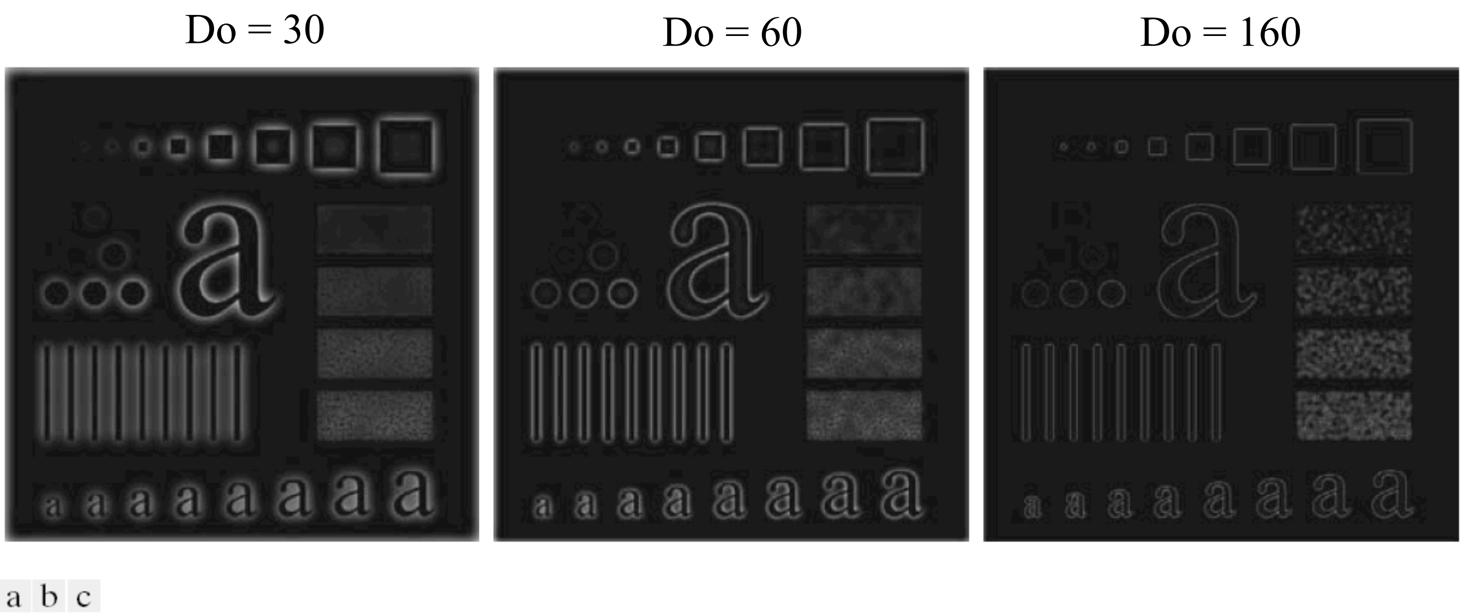


FIGURE 4.55 Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with $D_0 = 30, 60,$ and 160, corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPF.

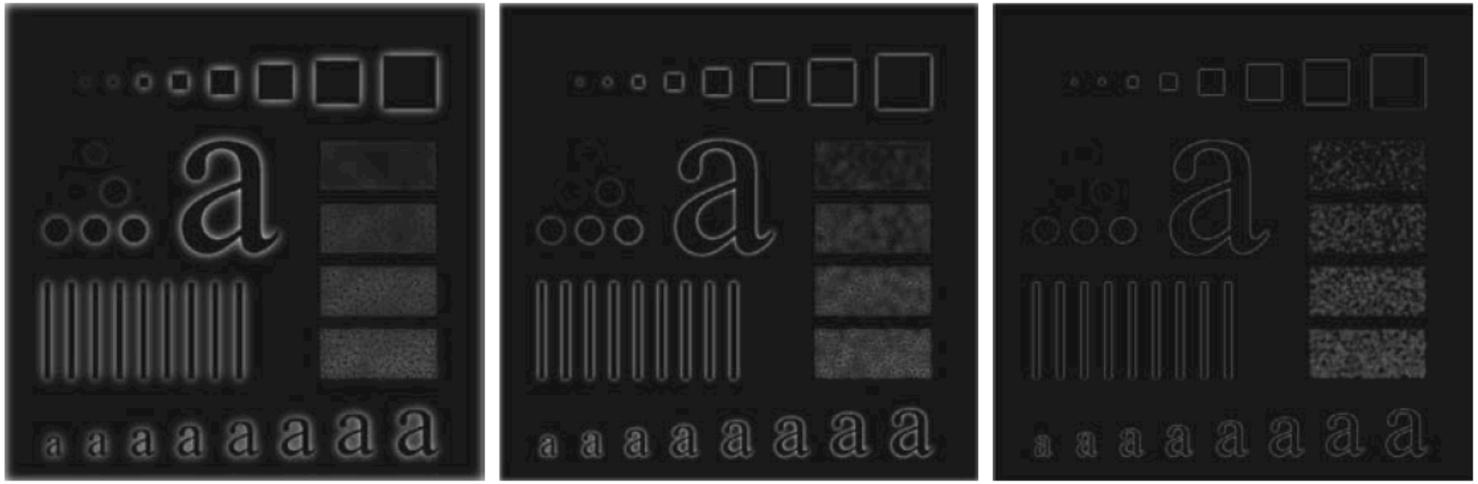
4.9.3 Gaussian Highpass Filters

The transfer function of the *Gaussian Highpass Filter* (GHPF) with cutoff frequency locus at a distance D_0 from the centre of the frequency rectangle is defined as:

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

4.9.3 Gaussian Highpass Filters

Do = 30 Do = 60 Do = 160



a b c

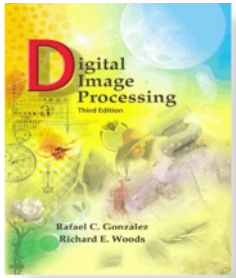
FIGURE 4.56 Results of highpass filtering the image in Fig. 4.41(a) using a GHPF with $D_0 = 30, 60$, and 160, corresponding to the circles in Fig. 4.41(b). Compare with Figs. 4.54 and 4.55.

TABLE 4.5
Highpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

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

Example: using highpass filtering and thresholding for image enhancement

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.)



4.9.4 The Laplacian in the Frequency Domain

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

Or, with respect to the centre of the frequency rectangle:

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

The Laplacian image is obtained by:

$$\nabla^2 f(x,y) = IDFT\left[H(u,v)F(u,v)\right]$$

Enchancement of the image is achieved using (H(u,v) negative):

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

4.9.4 The Laplacian in the Frequency Domain

$$\nabla^2 f(x,y) = IDFT\left[H(u,v)F(u,v)\right]$$

$\Rightarrow$ Introduction of large scaling factors

$\Rightarrow$ Practical solution: normalize *f(x,y)* to the range [0,1] before computing the DFT, and divide $\nabla^2 f(x,y)$ by its maximum value

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).

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

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

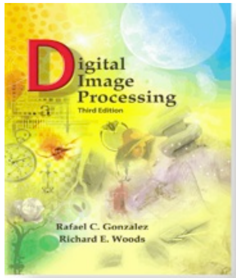
with

$$f_{LP}(x,y) = IDFT\left[H_{LP}(u,v)F(u,v)\right]$$

Smoothed image Lowpass Filter

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

- *k=1 => unsharp masking*
- *k>1 => highboost filtering*



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

Expressed in terms of frequency domain computations:

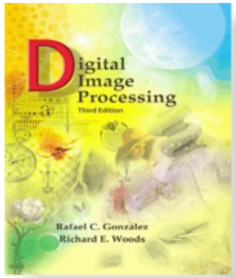
Lowpass filter: $$g(x,y) = IDFT\left\{ [1 + k * [1 - H_{LP}(u,v)]]\, F(u,v) \right\}$$

Highpass filter: $$g(x,y) = IDFT\left\{ \underbrace{[1 + k * H_{HP}(u,v)]}_{\textit{High-frequency-emphasis filter}}\, F(u,v) \right\}$$

More general formulation:

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

Where: $k_1 \geq 0$ gives controls of the offset from the origin
$k_2 \geq 0$ controls the contribution of high frequencies

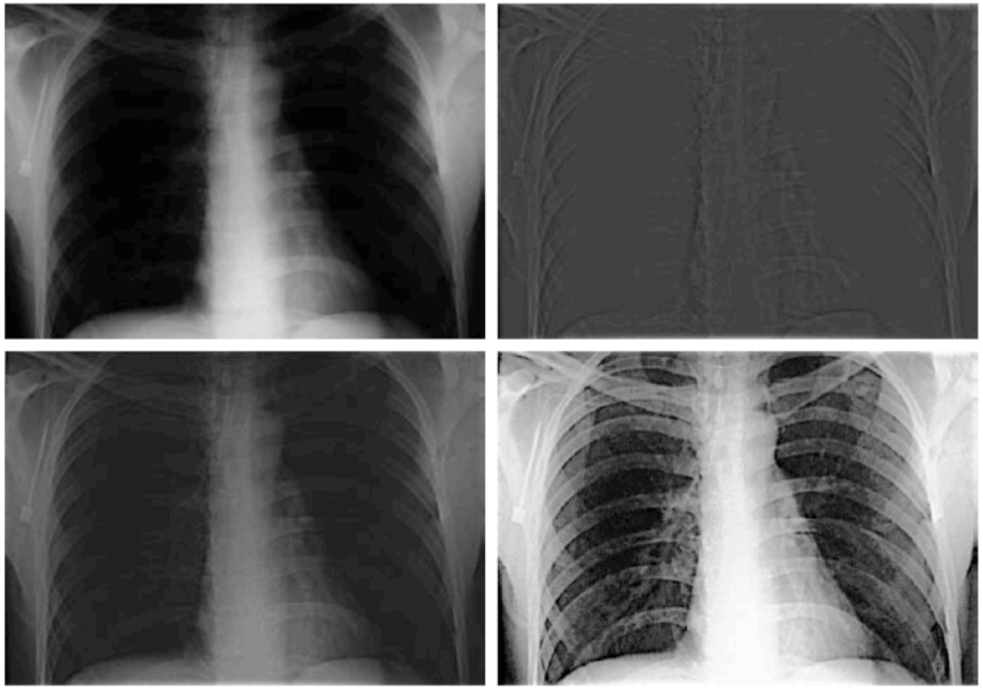


4.9.5 Unsharp Masking, Highboost Filtering and High-Frequency-Emphasis Filtering

Gaussian highpass filtering

High-frequency emphasis filtering

Histogram Equalisation



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.)

4.10 Selective Filtering

4.10.1 Bandreject and Bandpass Filters

TABLE 4.6
Bandreject filters. W is the width of the band, D is the distance $D(u, v)$ from the center of the filter, D_0 is the cutoff frequency, and n is the order of the Butterworth filter. We show D instead of $D(u, v)$ to simplify the notation in the table.

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

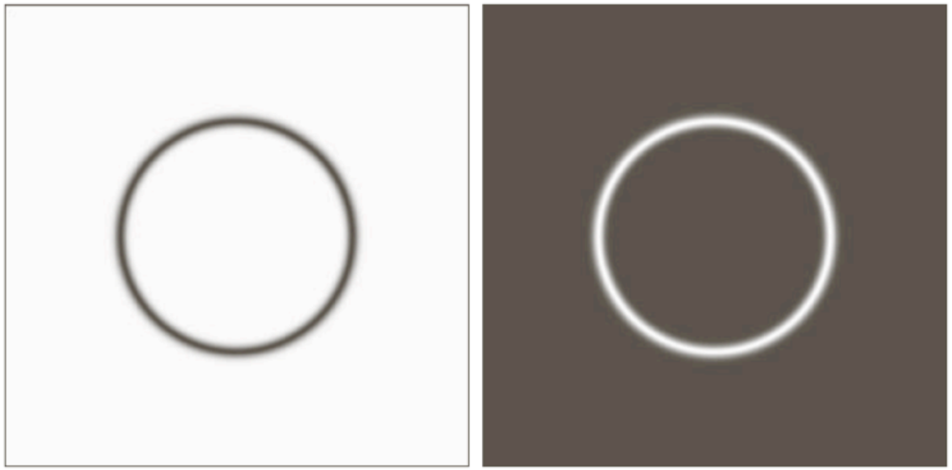
A *bandpass* filter is obtained from a *bandreject* filter as:

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

4.10.1 Bandreject and Bandpass Filters

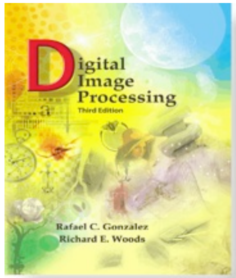
Example: Bandreject Gaussian filter

Bandreject Bandpass



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.



4.10.2 Notch Filters

Reject (or pass) frequencies in a predefined neighbourhood about the centre of the frequency rectangle

Constructed as products of highpass filters whose centres have been translated to the centres of the notches

$$H_{NR}(u,v) = \prod_{k=1}^{Q} \underset{\substack{\text{centre at} \\ (u_k,\, v_k)}}{H_k(u,v)}\, \underset{\substack{\text{centre at} \\ (-u_k,\, -v_k)}}{H_{-k}(u,v)}$$

=> Distances computations:

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

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

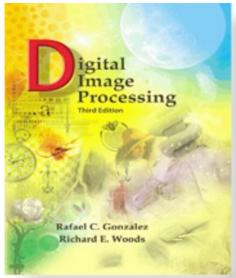
4.10.2 Notch Filters

Example: Butterworth notch reject filter of order n, containing 3 notch pairs:

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

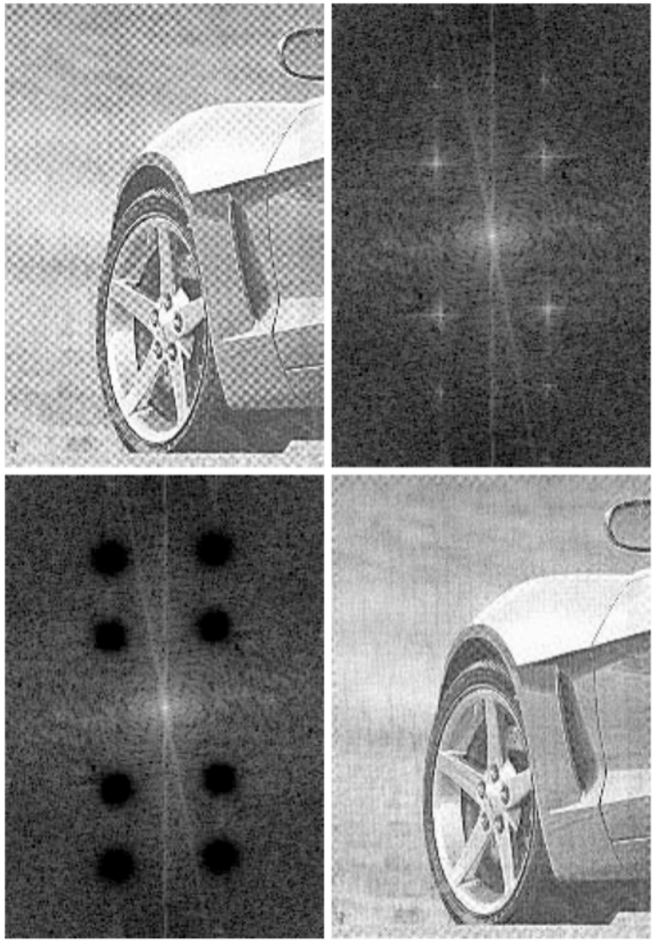
A *Notch Pass filter* (NP) is obtained from a *Notch Reject filter* (NR) using:

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



Chapter 4
Filtering in the Frequency Domain

Newspaper image FT Spectrum

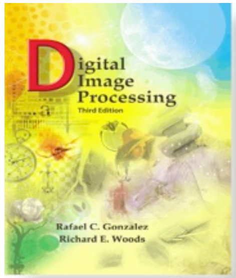


a b
c d

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

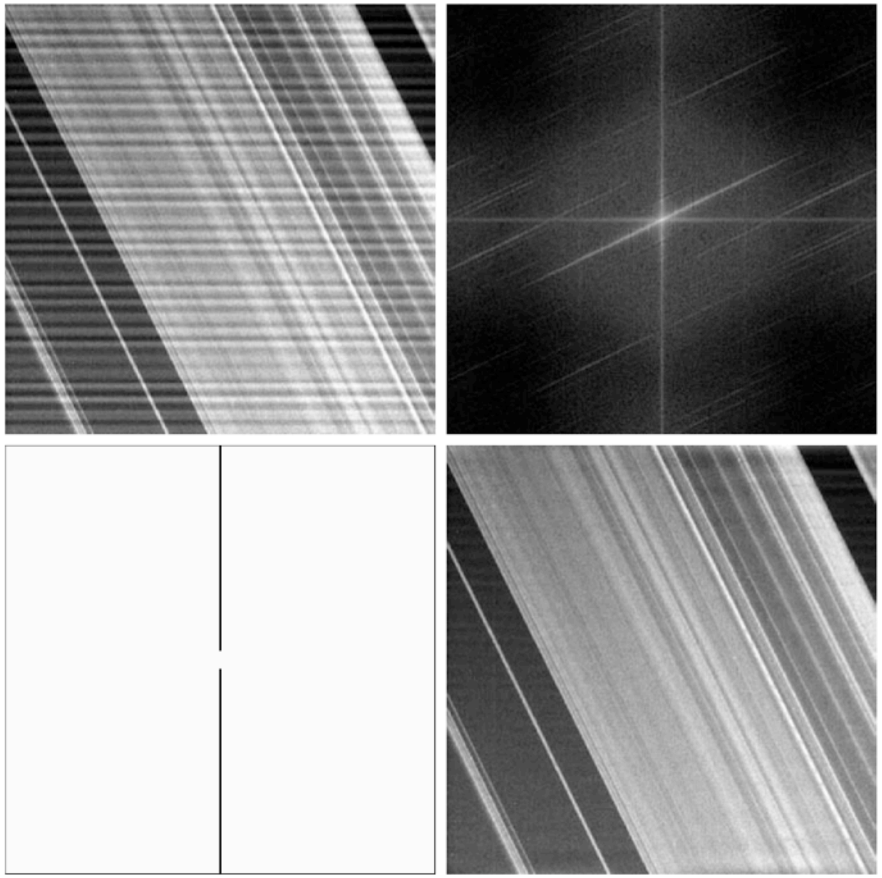
Butterworth notch reject filter multiplied by FT

Filtered image



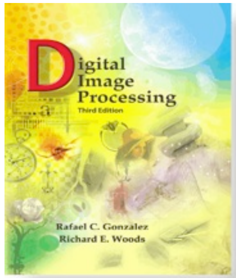
FT Spectrum

Vertical notch reject filter



Filtered image

FIGURE 4.65 (a) 674 × 674 image of the Saturn rings showing nearly periodic interference. (b) Spectrum: The bursts of energy in the vertical axis near the origin correspond to the interference pattern. (c) A vertical notch reject filter. (d) Result of filtering. The thin black border in (c) was added for clarity; it is not part of the data. (Original image courtesy of Dr. Robert A. West, NASA/JPL.)



4.11 Implementation

4.11.1 Separability of the 2-D DFT

The 2-D DFT is separable into 1-D transforms:

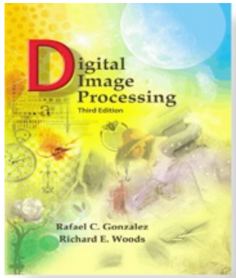
$$F(u,v) = \sum_{x=0}^{M-1} e^{-j2\pi ux/M} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi vy/N}$$

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

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

= 1-DFT of a row of f(x,y)

The 2-D DFT of f(x,y) can be obtained by computing the 1-D transform of each row of f(x,y) and then computig the 1-D transform along each column of the result



4.11 Implementation

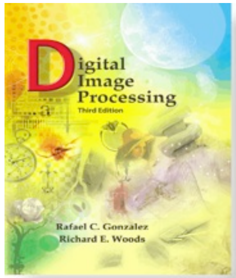
4.11.2 Computing the IDFT Using a DFT Algorithm

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

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

$\Rightarrow$ To compute the IDFT :
Compute the 2-D forward DFT of $F^*(u,v)$ $\Rightarrow MNf^*(x,y)$
Take the complex conjugate and multiply the result by $1/MN$

NB: when $f(x,y)$ is real : $f^*(x,y) = f(x,y)$



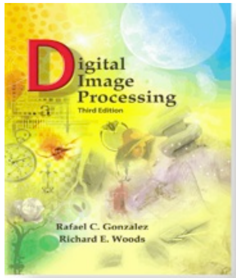
4.11.3 The Fast Fourier Transform (FFT)

$$F(u) = \sum_{x=0}^{M-1} f(x) W_M^{ux} \qquad \text{where:} \qquad W_M = e^{-j2\pi/M}$$

M assumed to be $M = 2^n$ => $M = 2K$

=>
$$F(u) = \underbrace{\sum_{x=0}^{K-1} f(2x) W_K^{ux}}_{F_{even}(u)} + \underbrace{\sum_{x=0}^{K-1} f(2x+1) W_K^{ux}}_{F_{odd}(u)} W_{2K}^{ux}$$

=> for $u = 0, 1, 2, \ldots, K-1$
$$F(u) = F_{even}(u) + F_{odd}(u) W_{2K}^{u}$$
$$F(u+K) = F_{even}(u) - F_{odd}(u) W_{2K}^{u}$$



4.11.3 The Fast Fourier Transform (FFT)

- Direct implementation: on the order of $(MN)^2$ summations and additions required
- FFT implementation: order of $(MN \log_2 MN)$

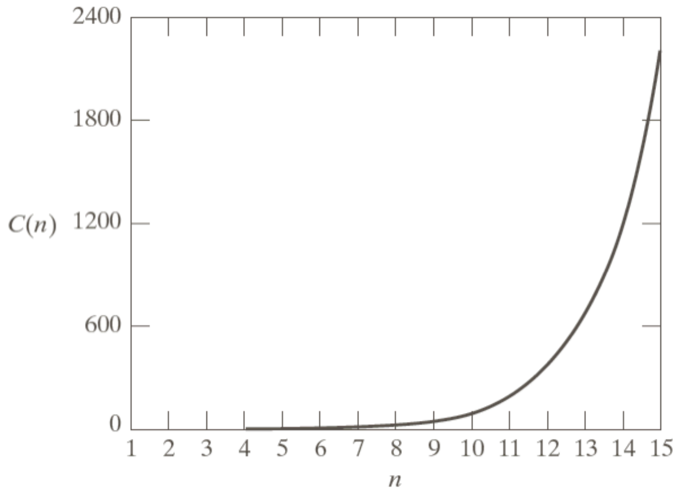
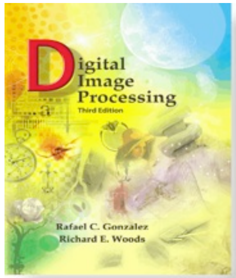


FIGURE 4.67 Computational advantage of the FFT over a direct implementation of the 1-D DFT. Note that the advantage increases rapidly as a function of n.



Glossary

- FT = Fourier Transform
- DFT = Discrete Fourier Transform
- IDFT = Inverse Discrete Fourier Transform
- FFT = Fast Fourier Transform
- ILPF (IHPF) = Ideal Lowpass (Highpass) Filter
- BLPF (BHPF) = Butterworth Lowpass (Highpass) Filter
- GLPF (GHPF) = Gaussian Lowpass (Highpass) Filter
- BP = Bandpass
- BR = Bandreject
- NP = Notch Pass
- NR = Notch Reject