上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

IVM
Image, Video, and Multimedia Communications Laboratory

Digital Image Processing

Hongkai Xiong
熊红凯
http://ivm.sjtu.edu.cn

电子工程系
上海交通大学

2016

Topic

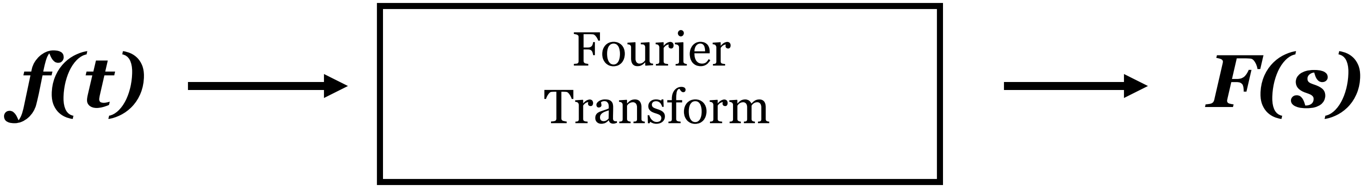
- Simple review of Fourier transform

- Filtering in frequency domain

Simple review of Fourier transform

The Fourier Transform

- The Fourier transform converts a signal from spatial domain to the frequency domain.



- The Continuous Fourier Transform of a one-dimensional continuous function *f(t)* is defined as

$$\mathbf{F}\{f(t)\} = F(s) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi st} dt$$

- The inverse Fourier transform

$$\mathbf{F}^{-1}\{F(s)\} = \int_{-\infty}^{\infty} F(s) e^{j2\pi st} ds$$

The Discrete Fourier Transform

- If $\{ f_i \}$ is a sequence of length N, then its *discrete Fourier transform* (DFT) is given by

$$F_n = \frac{1}{N} \sum_{i=0}^{N-1} f_i e^{-j2\pi \frac{n}{N} i}, \quad n = 0,1,\cdots N-1$$

- And the inverse DFT is given by

$$f_i = \sum_{n=0}^{N-1} F_n e^{-j2\pi \frac{i}{N} n}, \quad i = 0,1,\cdots, N-1$$

Where $0 \le i, n \le N-1$ are indices.

Properties of the Fourier Transform

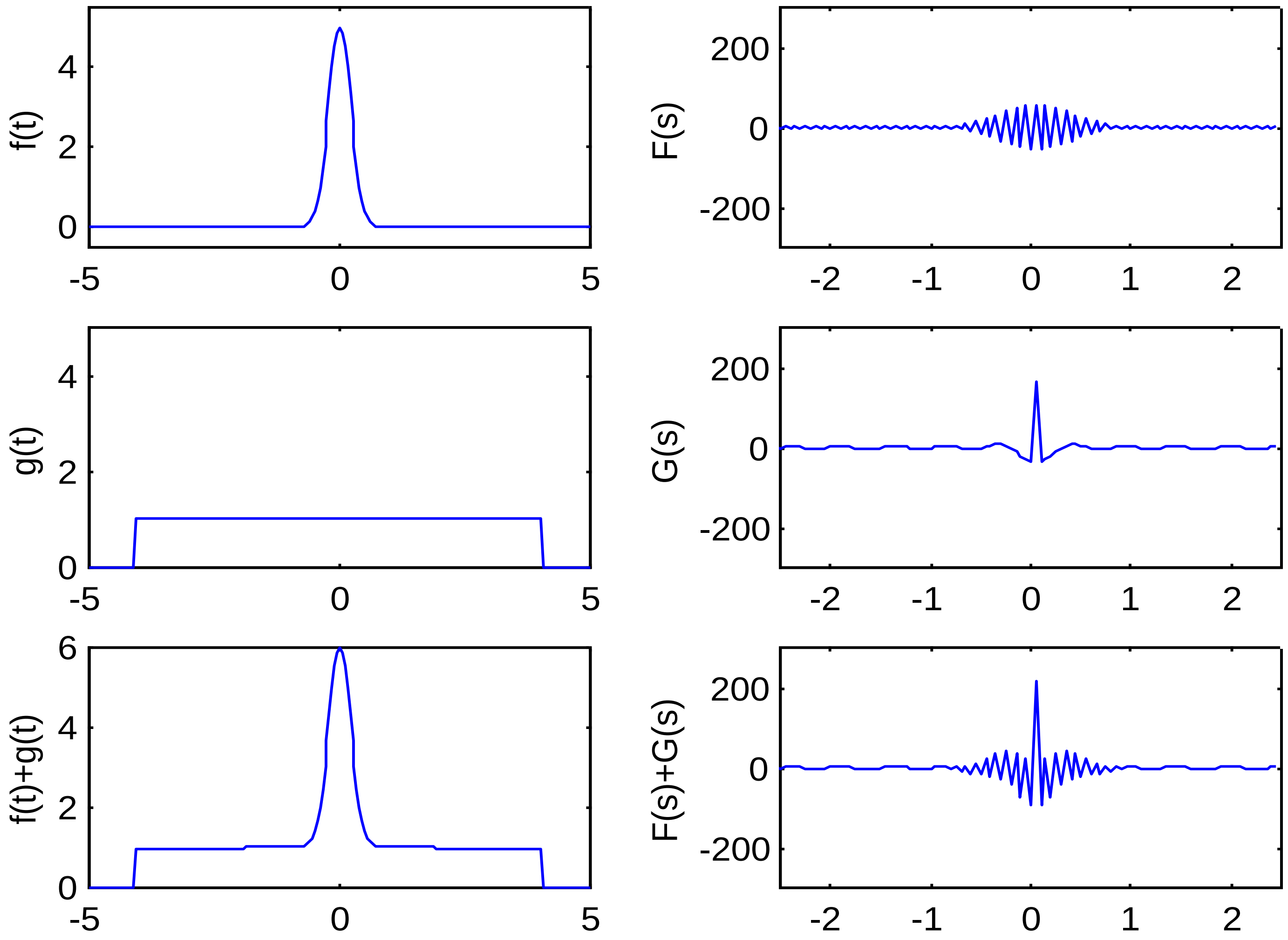
- The Addition Theorem
  - If $\mathbf{F}\{f(t)\}=F(s)$ and $\mathbf{F}\{g(t)\}=G(s)$, then

$$\mathbf{F}\{f(t)+g(t)\}=F(s)+G(s)$$

  - And take it as an axiom that for any real number c

$$\mathbf{F}\{cf(t)\}=cF(s)$$

  - This implies that Fourier transform is a linear transform.


f(t)
F(s)
g(t)
G(s)
f(t)+g(t)
F(s)+G(s)
-5
0
5
-2
-1
0
1
2
200
-200
4
2
6

The Shift Theorem

- Time shift:

$$\mathbf{F}\{f(t-a)\} = e^{-j2\pi as}F(s)$$

- Frequency shift:

$$\mathbf{F}\{f(t)e^{j2\pi s_0 t/N}\} = F(s-s_0)$$

When $s_0 = N/2$,

$$\mathbf{F}\{f(t)(-1)^t\} = F(s-N/2)$$

back

The Convolution Theorem

- The Fourier transform of the convolution of two functions is the product of the Fourier transforms of these two functions

$$\mathbf{F}\{f(t) * g(t)\} = F(s)G(s)$$

The Two-Dimensional DFT

- The discrete Fourier transform of an image of size M*N is given by

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

- Inverse Fourier transform

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

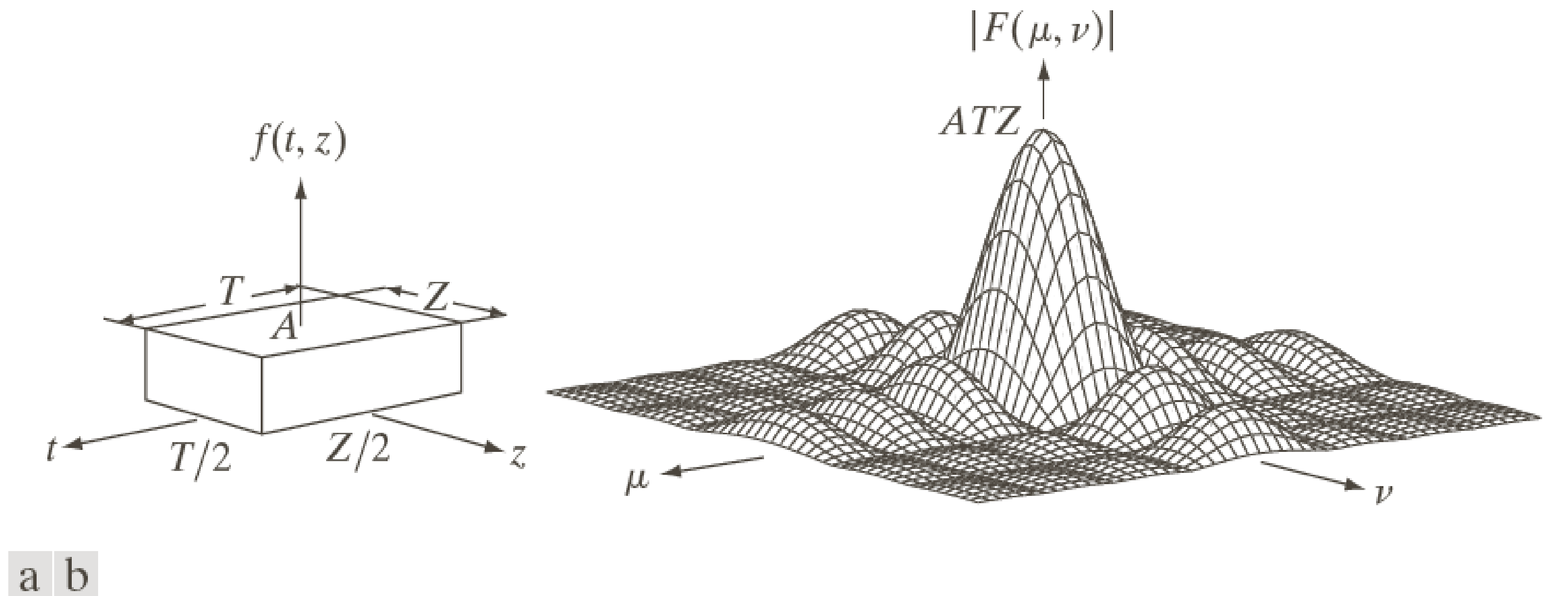


FIGURE 4.13 (a) A 2-D function, and (b) a section of its spectrum (not to scale). The block is longer along the *t*-axis, so the spectrum is more "contracted" along the μ-axis. Compare with Fig. 4.4.

- Fourier spectrum

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

- Phase angle

$$\phi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)} \right]$$

- Power spectrum

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

where R(u, v) and I(u, v) are the real and imaginary parts of F(u, v), respectively

- For display, it is common practice to shift the original point to the center (M/2, N/2). According to the shift theorem, we need to multiply the input image function by $(-1)^{x+y}$ prior to computing the Fourier transform.:

$$\mathcal{F}\left[f(x,y)(-1)^{x+y} \right] = F(u - M/2, v - N/2)$$

- What's the value of the transform at (u, v)=(0, 0)?

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

- The average of f(x, y).

A simple example

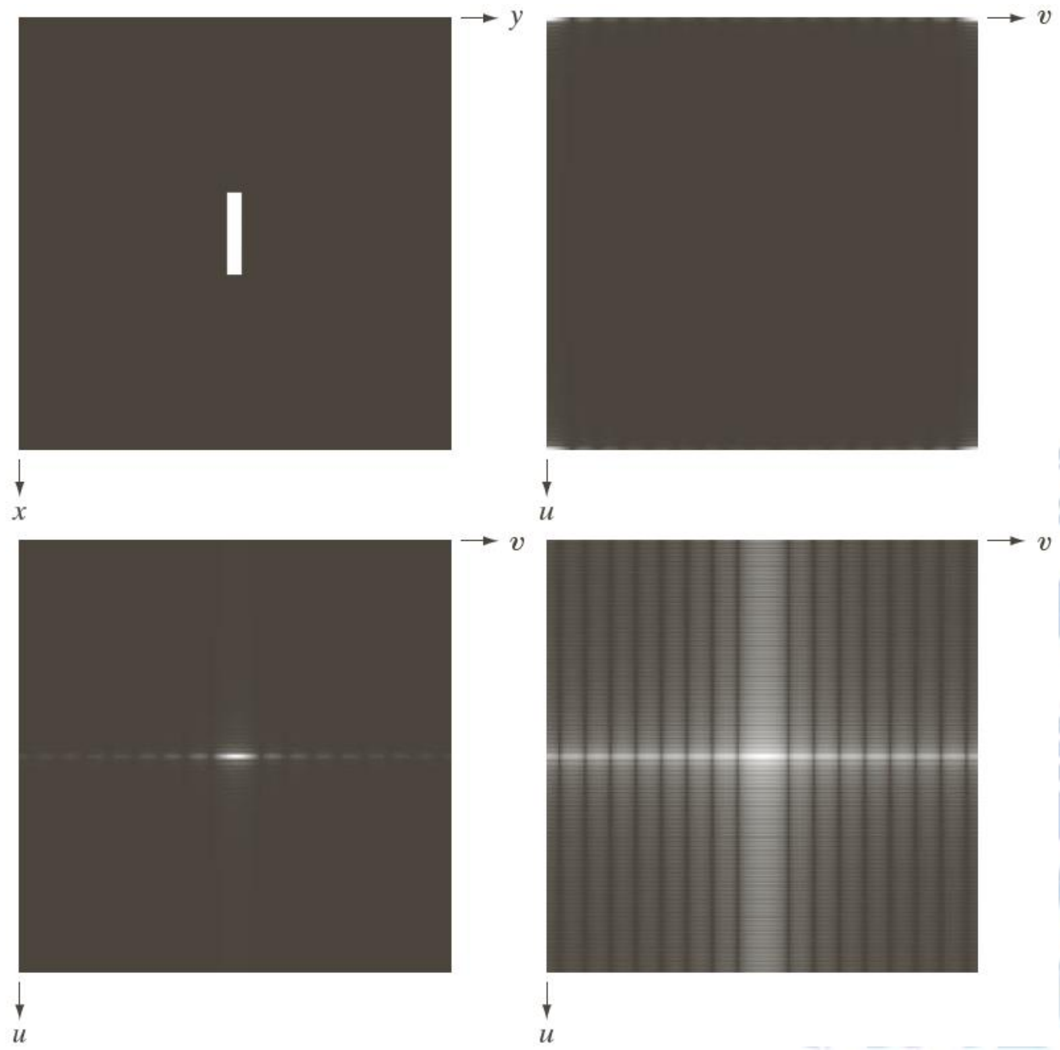
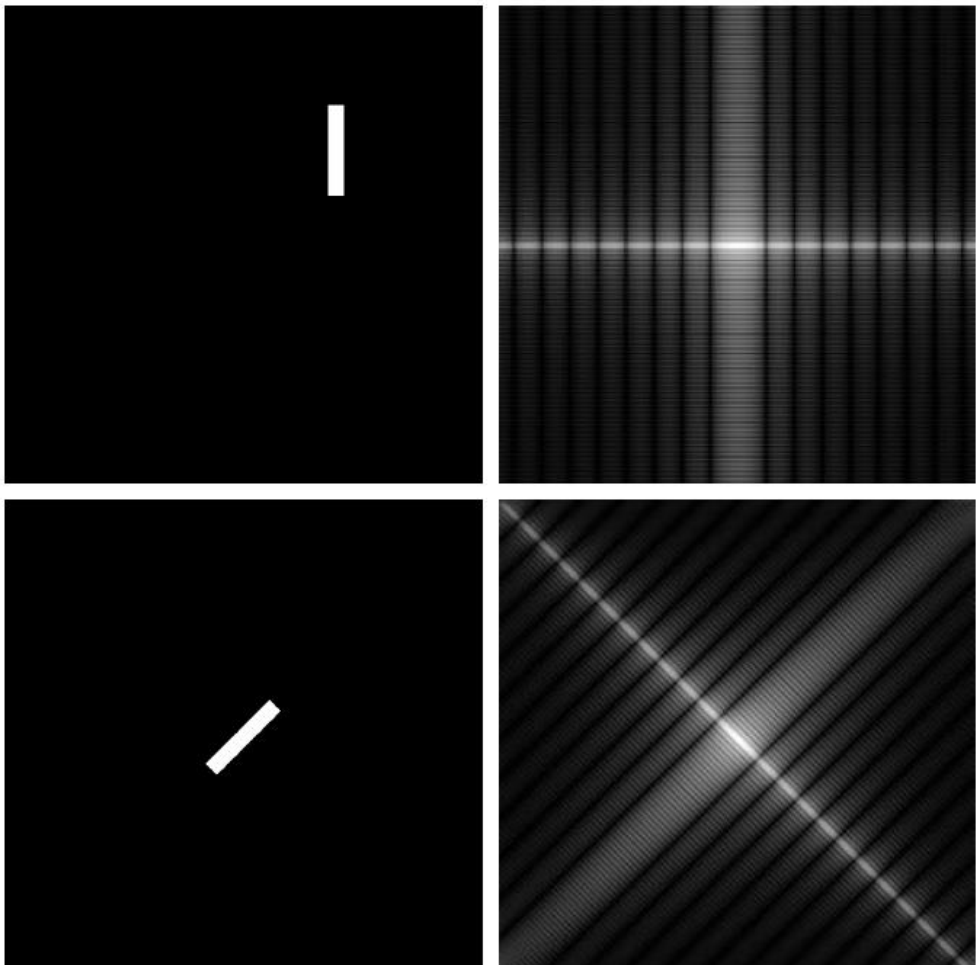


FIGURE 4.24 (a) Image. (b) Spectrum showing bright spots in the four corners. (c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate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

Translation and Rotation



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).

The 2-D Convolution Theorem

- 2-D circular convolution

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

- The 2-D convolution theorem is given by

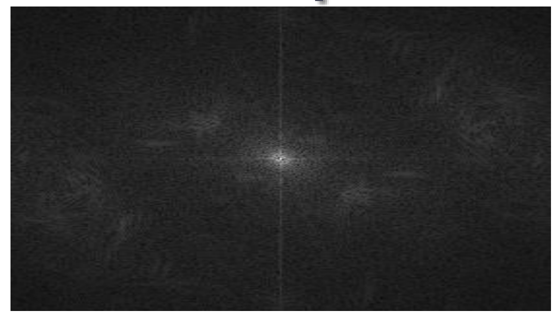
$$f(x,y) \star h(x,y) \Leftrightarrow F(u,v) H(u,v)$$

and, conversely,

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

2D convolution theorem example

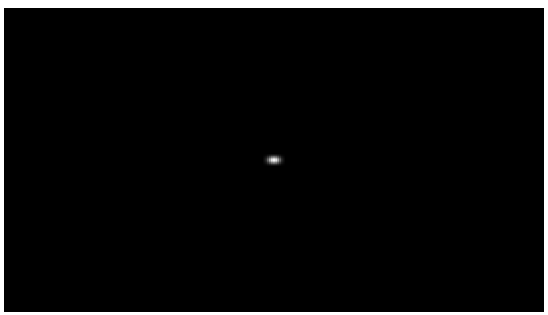
$f(x,y)$

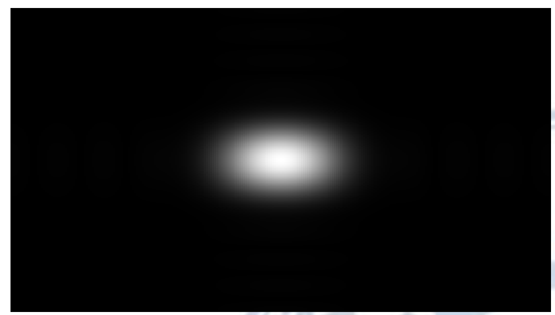


$|F(s_x,s_y)|$

$*$ $\times$

$h(x,y)$

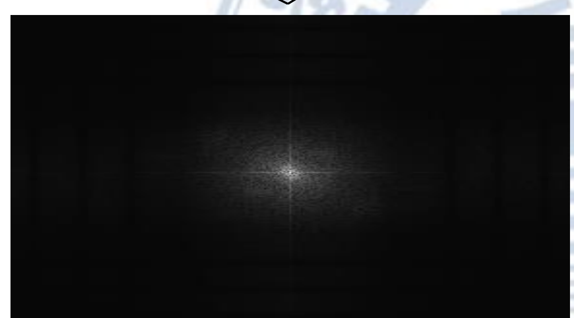




$|H(s_x,s_y)|$

$\Downarrow$ $\Downarrow$

$g(x,y)$



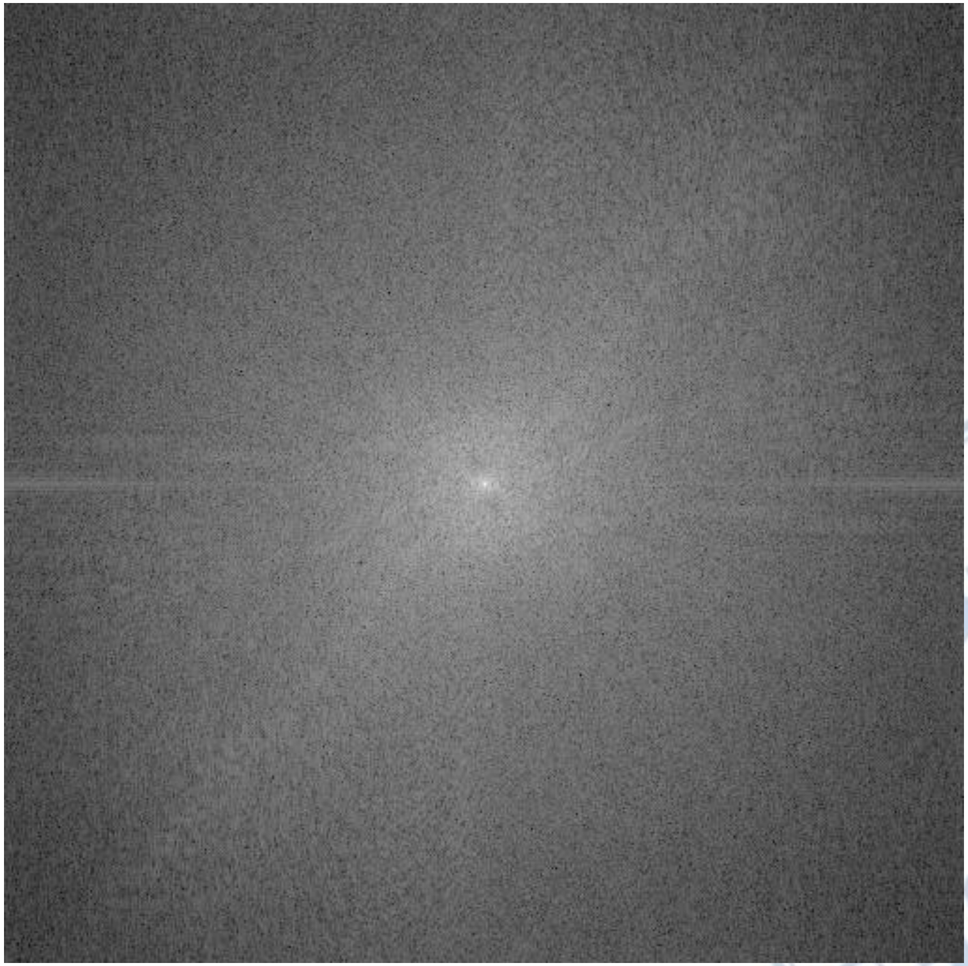
$|G(s_x,s_y)|$

Curious Things about FT on Images

Example:
cheetah pic

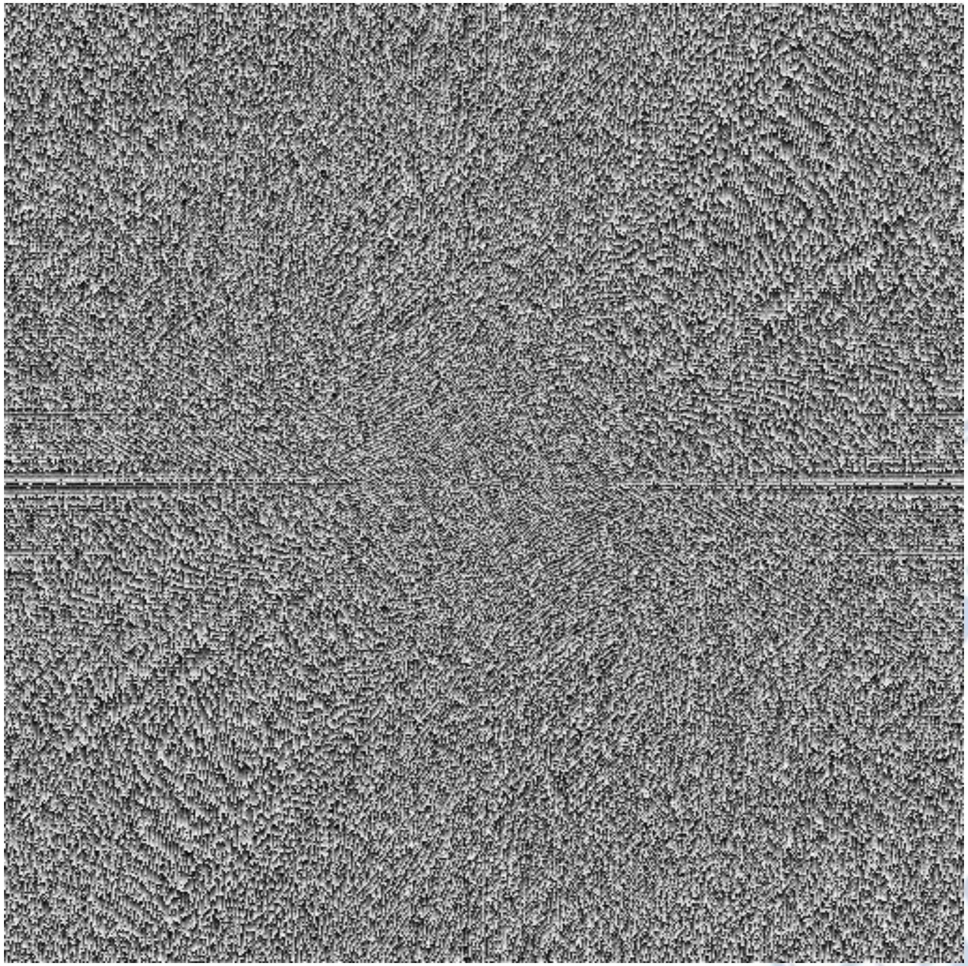
Curious Things about FT on Images

This is the magnitude transform of the cheetah pic



Curious Things about FT on Images

This is the phase transform of the cheetah pic

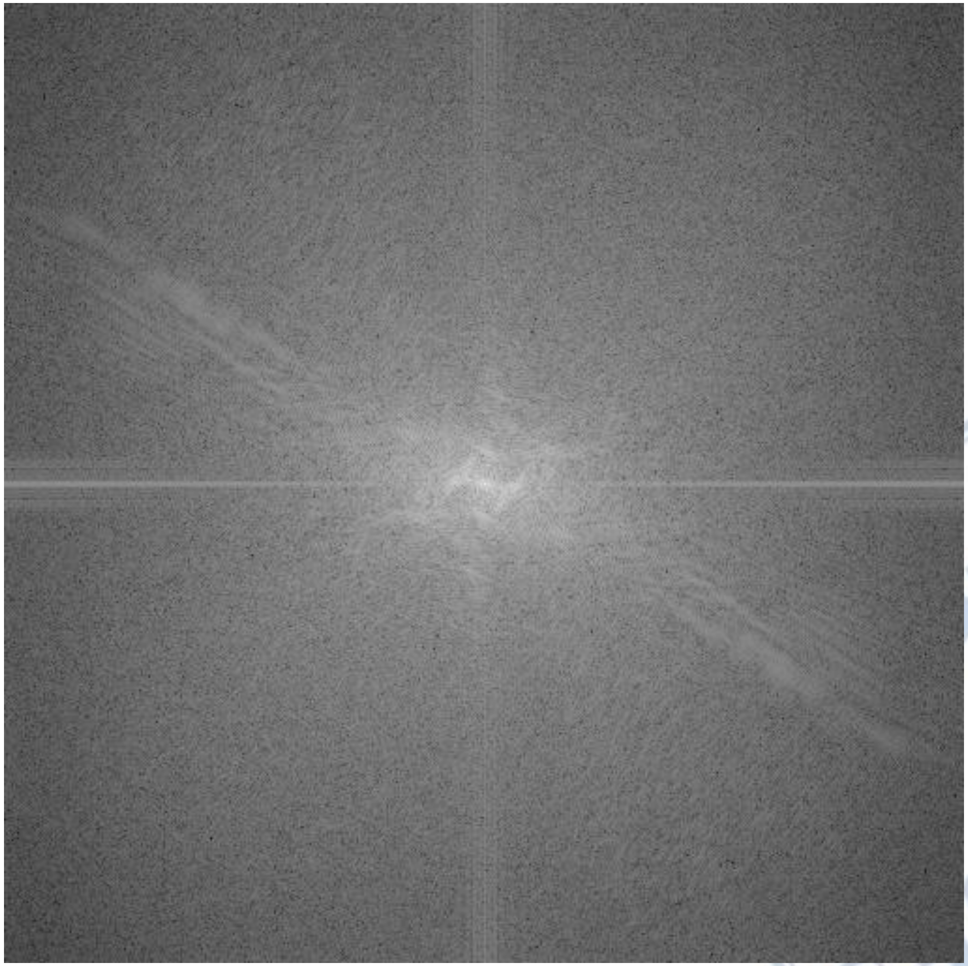


Curious Things about FT on Images

Example:
zebra pic

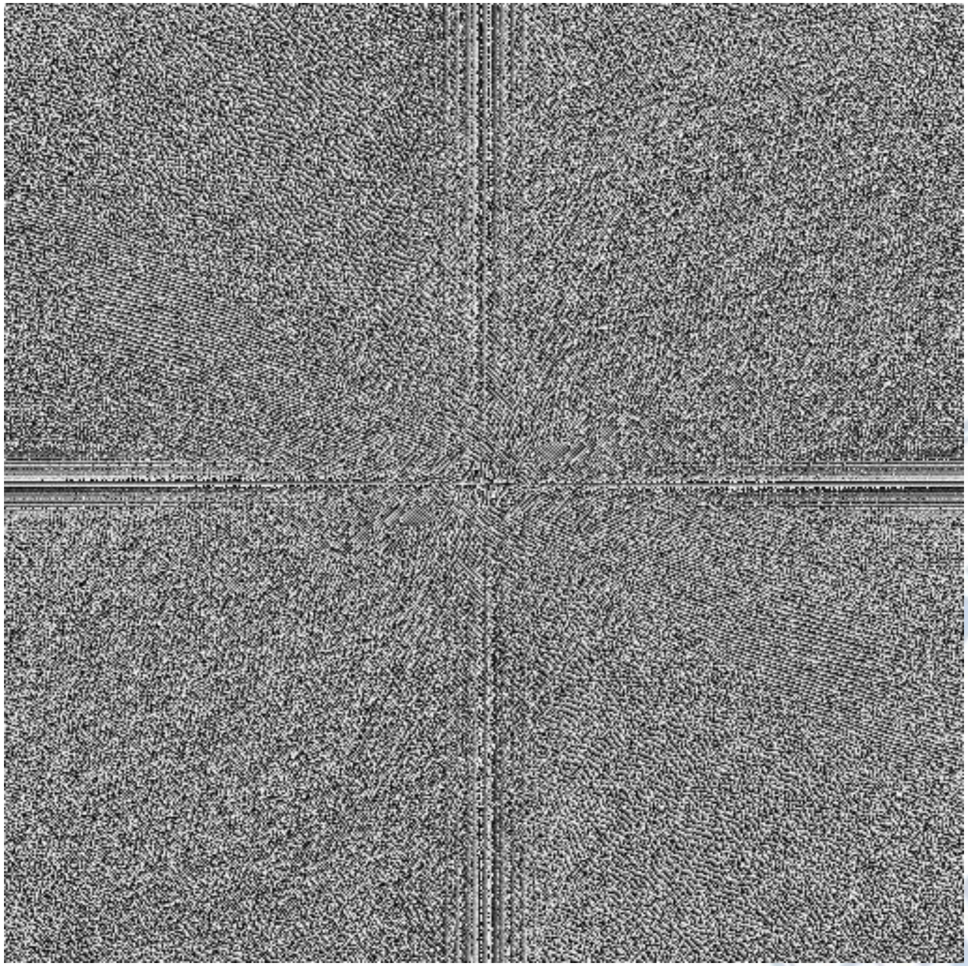
Curious Things about FT on Images

This is the magnitude transform of the zebra pic



Curious Things about FT on Images

This is the phase transform of the zebra pic

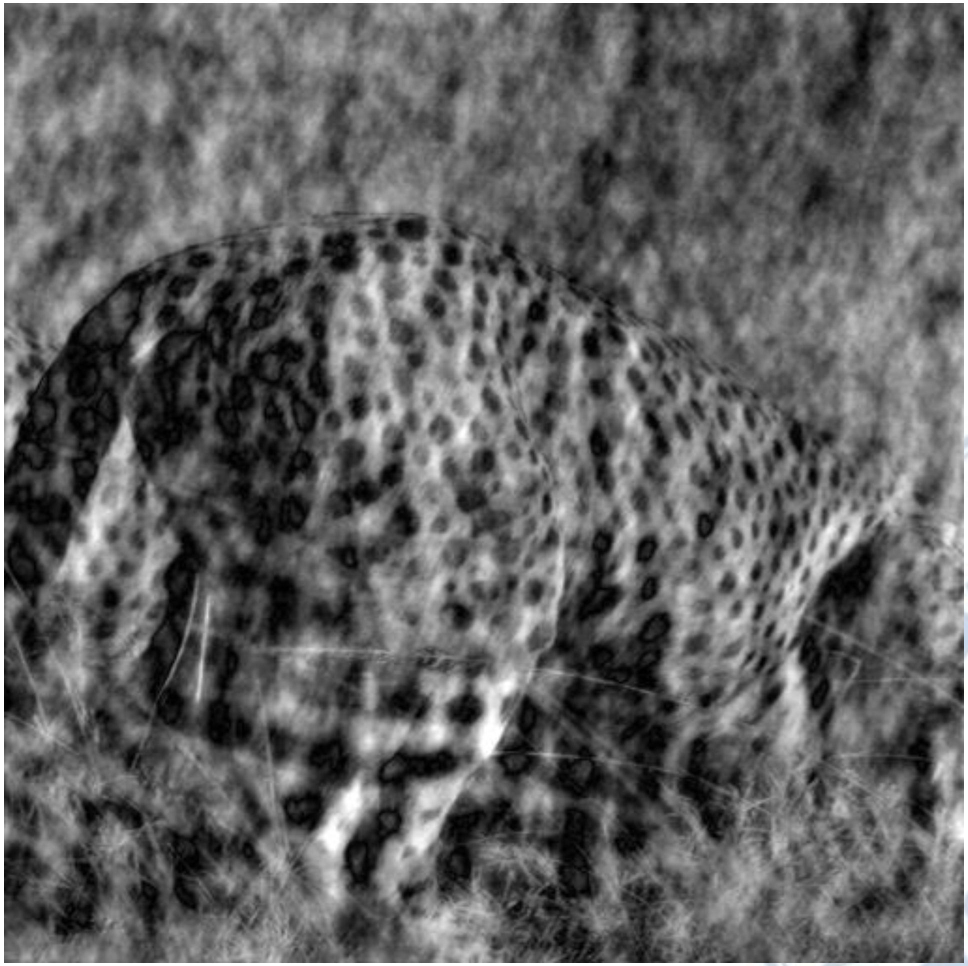


Curious Things about FT on Images

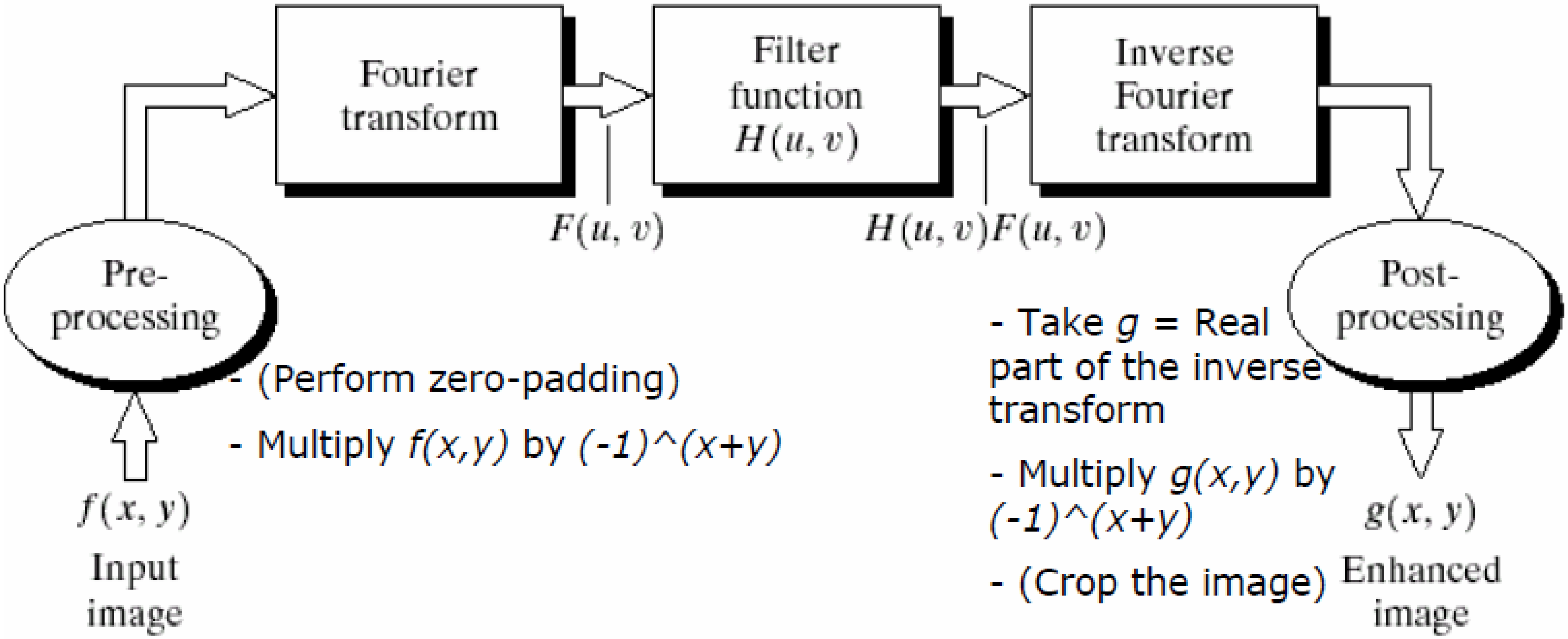
Reconstruction with zebra phase, cheetah magnitude

Curious Things about FT on Images

Reconstruction with cheetah phase, zebra magnitude



Filtering in frequency domain



- (Perform zero-padding)
- Multiply *f(x,y)* by *(-1)^(x+y)*

- Take *g* = Real part of the inverse transform
- Multiply *g(x,y)* by *(-1)^(x+y)*
- (Crop the image)

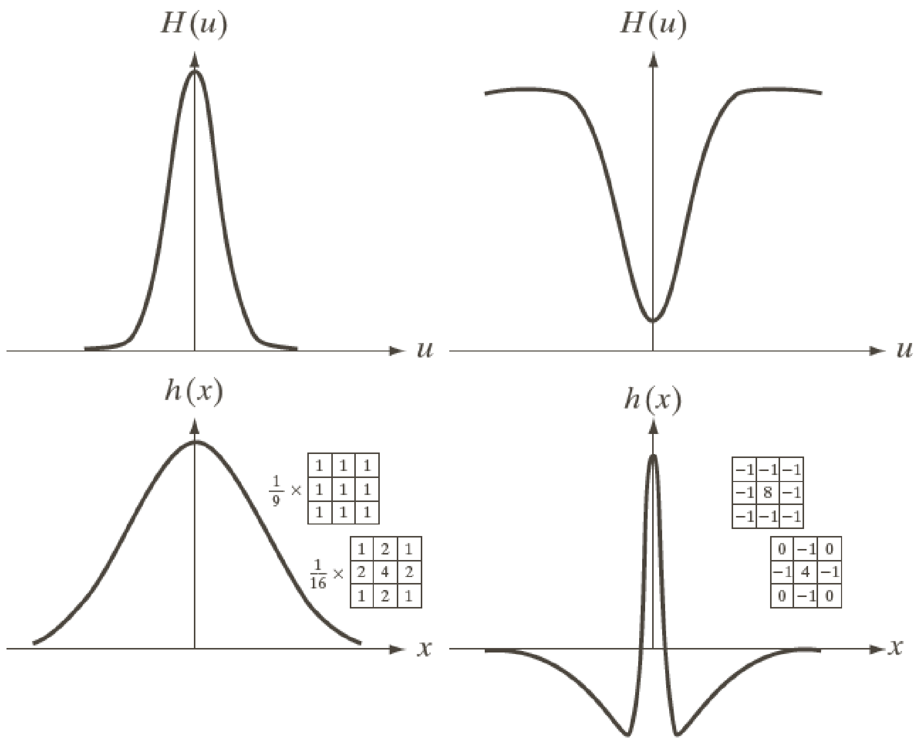
Steps for filtering in frequency domain

1) Multiply the input image by $(-1)^{x+y}$ to center the transform
2) Zero padding in case of aliasing
3) Compute F(u, v), the DFT of the image from (2)
4) Multiply F(u, v) by a filter function H(u, v)
5) Compute the inverse DFT of the result in (4)
6) Obtain the real part of the result in (5)
7) Multiply the result in (6) by $(-1)^{x+y}$
8) Crop the image

Correspondence Between Filtering in the Spatial and Frequency Domains

- h(x, y) is a spatial filter, and 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)

Gaussian low-pass filter and high-pass filter in the frequency and spatial domain



a c
b d

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.

Correspondence Between Filtering in the Spatial and Frequency Domains

A Gaussian kernel gives less weight to pixels further from the center of the window

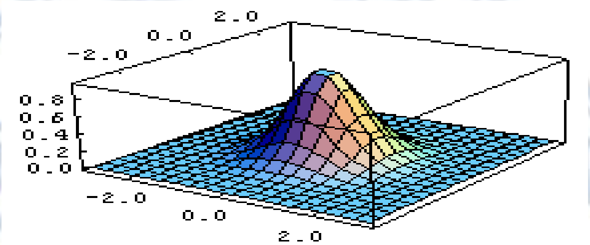
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$F[x, y]$

$\frac{1}{16}$

1	2	1
2	4	2
1	2	1

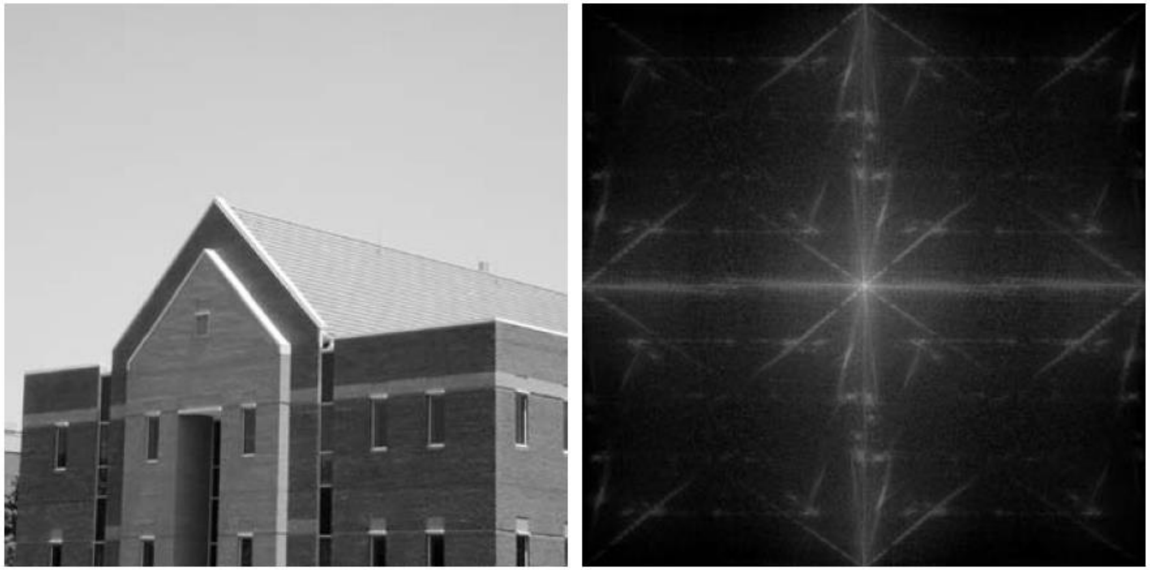
$H[u, v]$



This kernel is an approximation of a Gaussian function:

$$h(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{\sigma^2}}$$

Correspondence Between Filtering in the Spatial and Frequency Domains

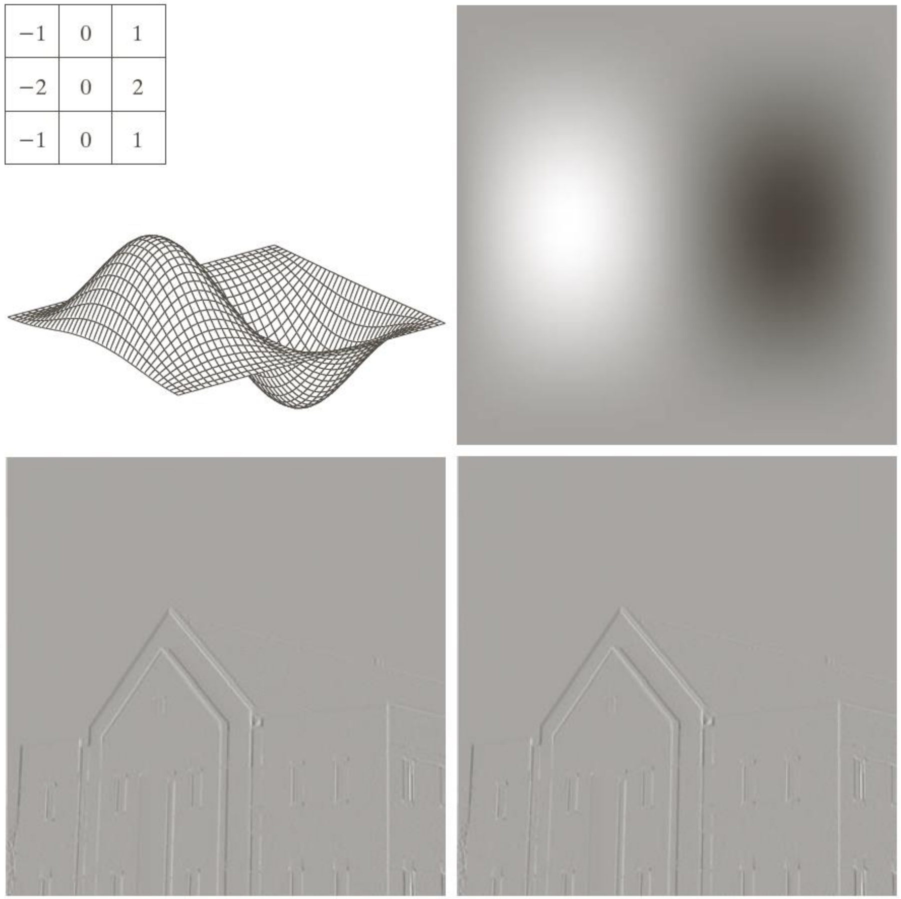


a b

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

Correspondence Between Filtering in the Spatial and Frequency Domains

−1	0	1
−2	0	2
−1	0	1



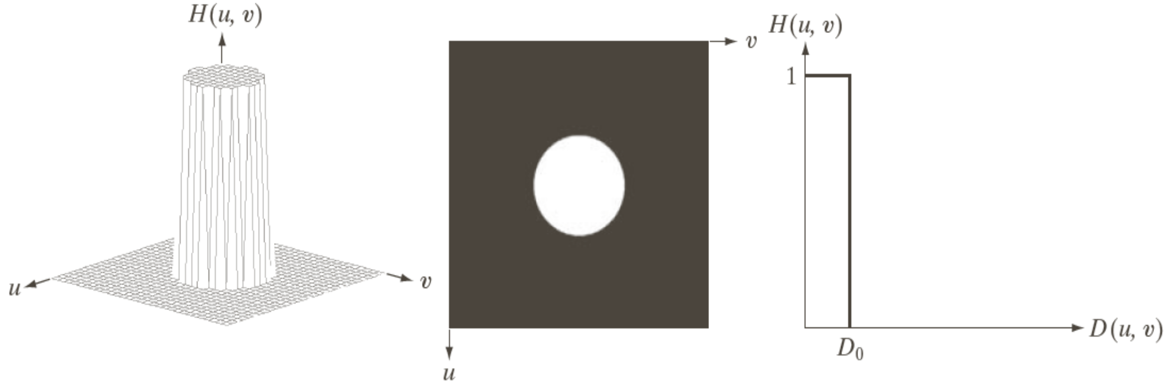
a b
c d

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.

Image Smoothing

- Ideal low-pass filters

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



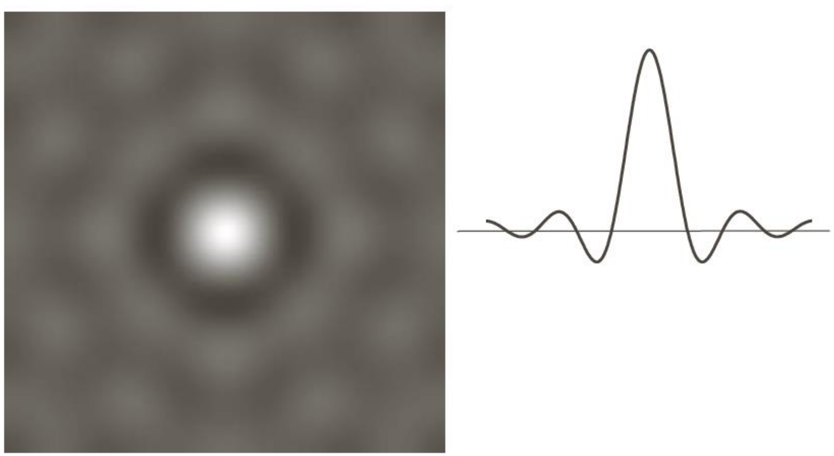
a b c

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

Image Smoothing

- Ideal low-pass filters

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

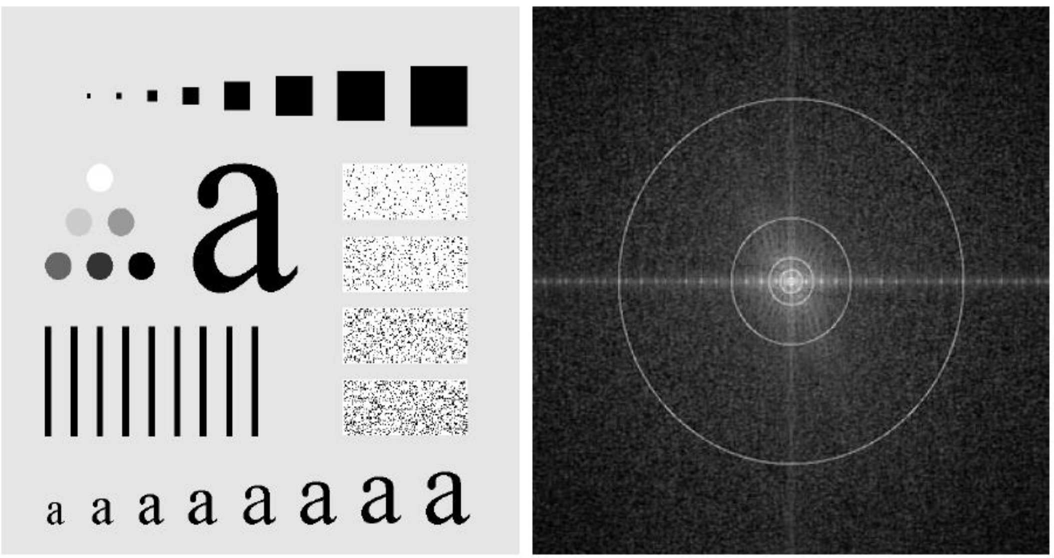


a b

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.

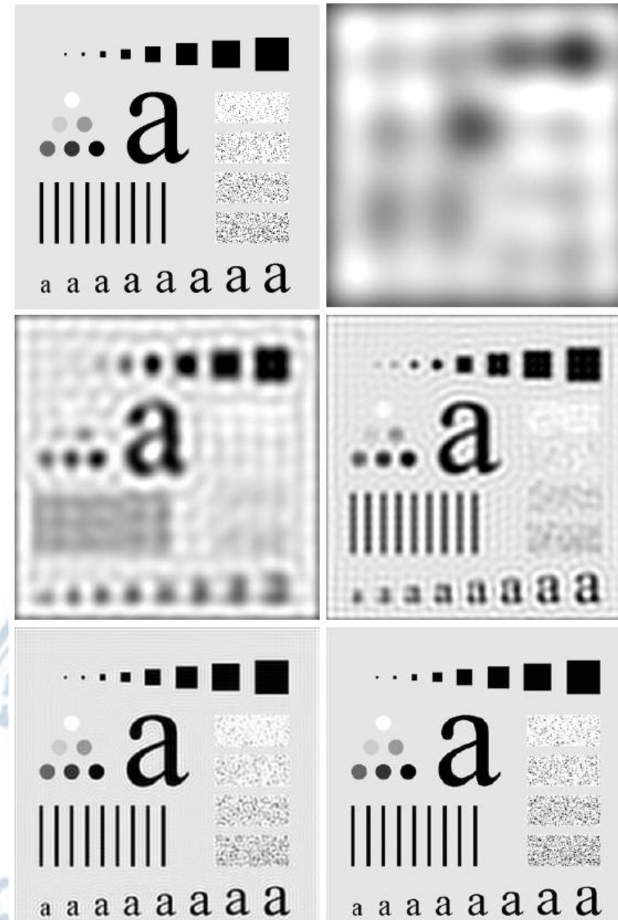
Image Smoothing

- Ideal low-pass 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.



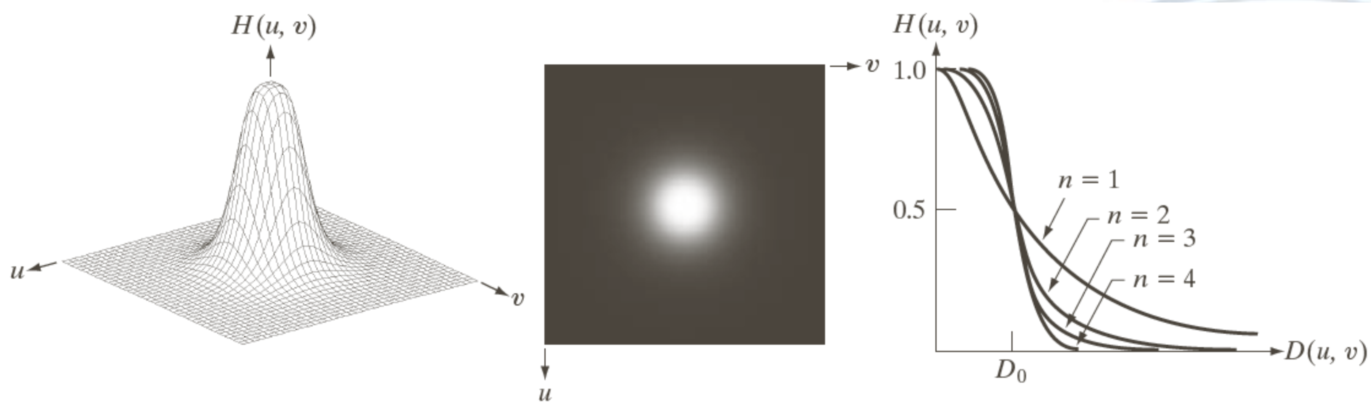
a b
c d
e f

FIGURE 4.42 (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.

Image Smoothing

- Butterworth low-pass filters

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



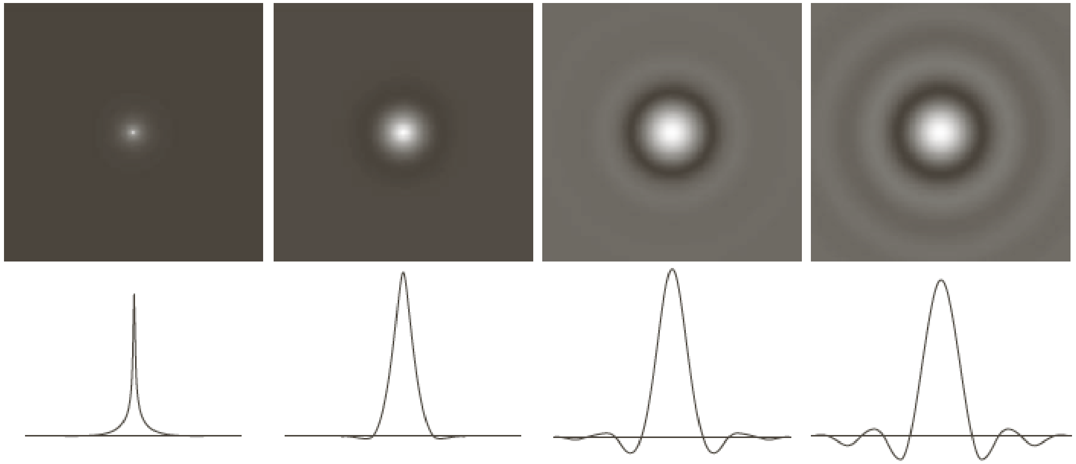
a b c

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.

Image Smoothing

- Butterworth low-pass filters

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



a b c d

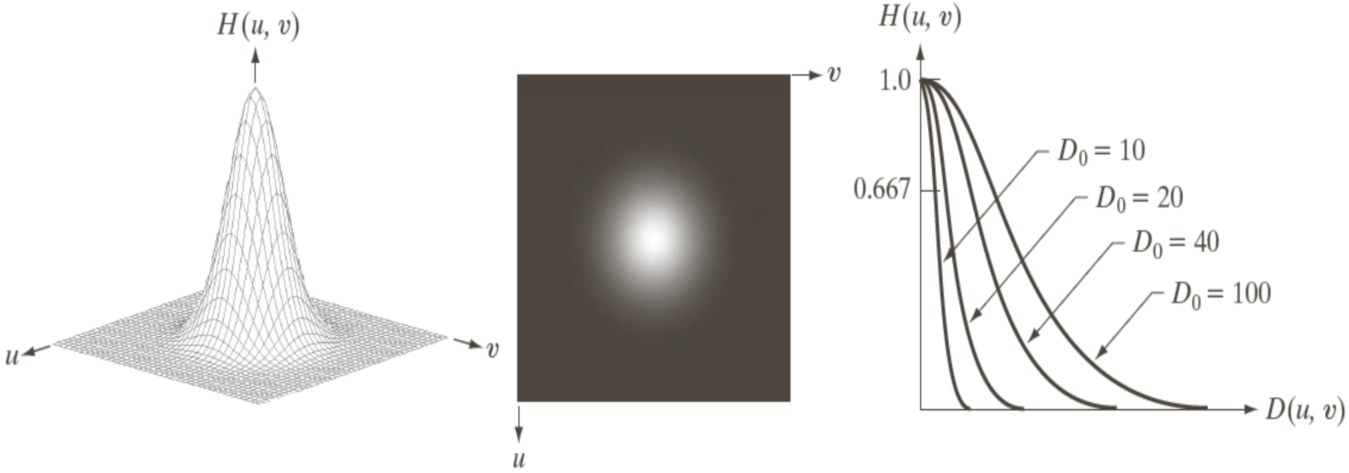
FIGURE 4.46 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is 1000 × 1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

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.

Image Smoothing

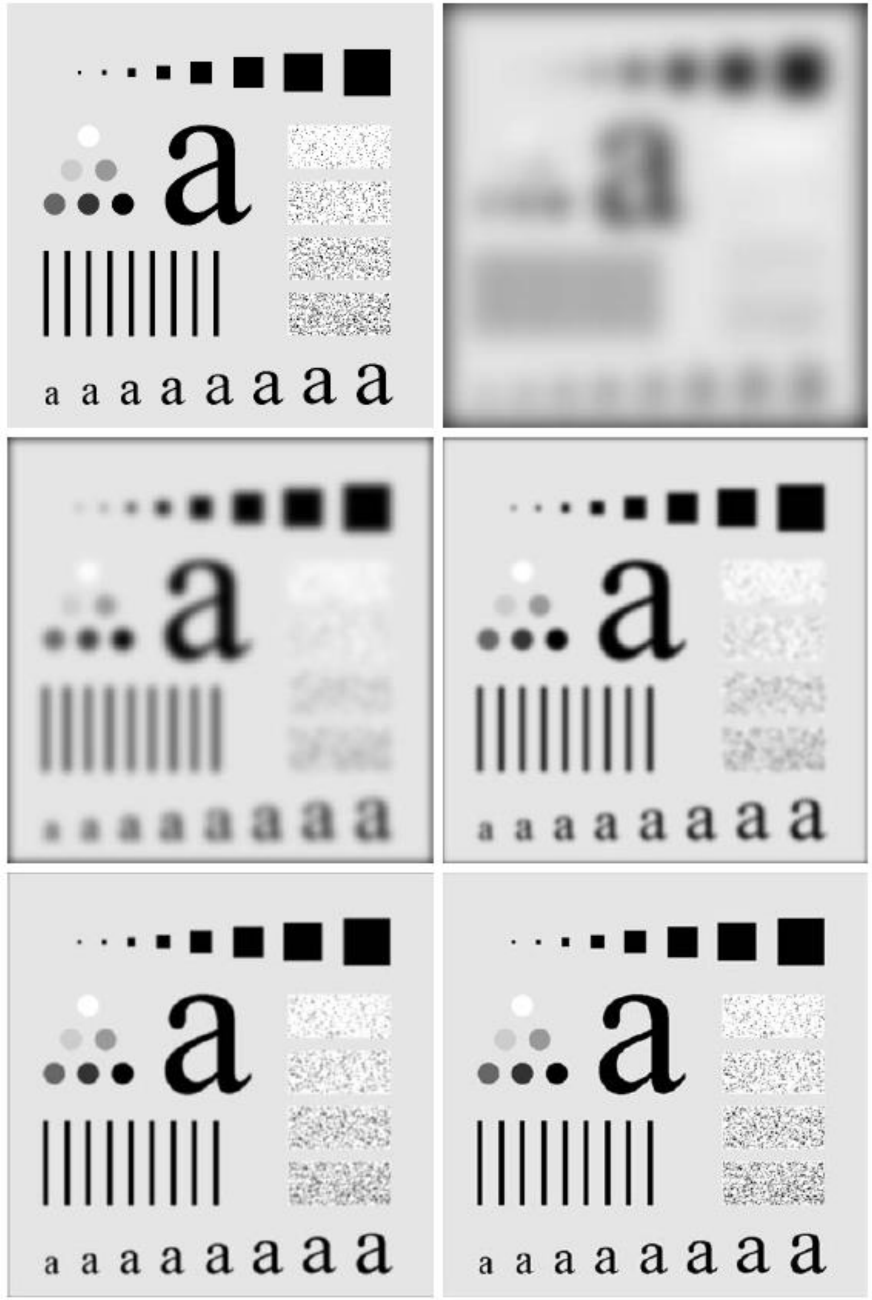
- Gaussian low-pass filters

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



a b c

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.

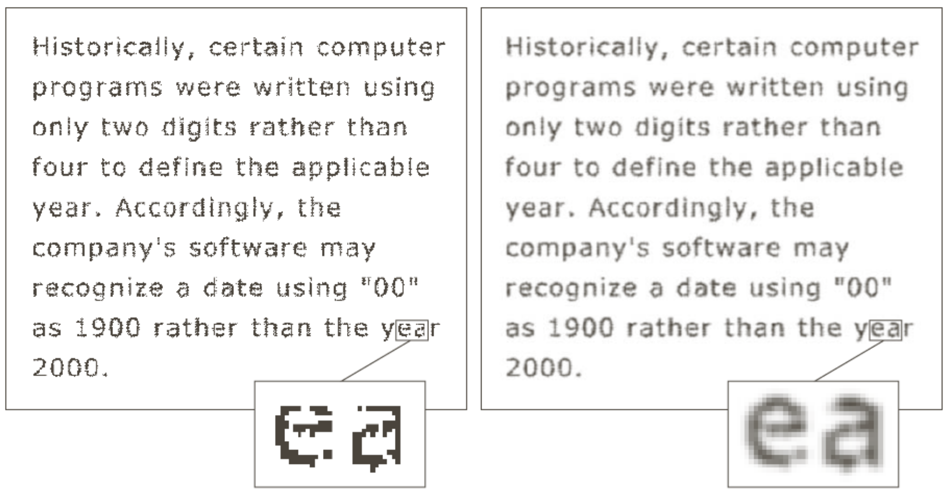


a b
c d
e f

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.

Image Smoothing

- Gaussian low-pass filters



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).

Image Sharpening

- A high-pass filter is obtained from a given low-pass filter

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

Image Sharpening

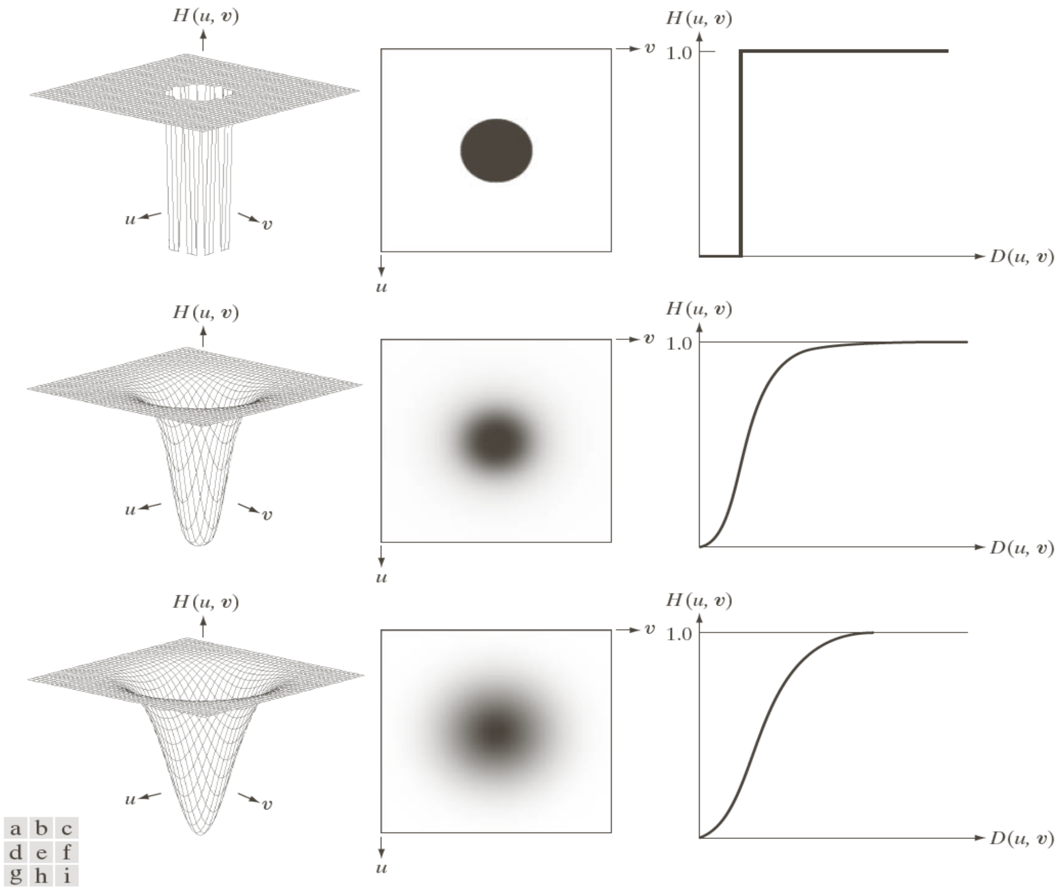
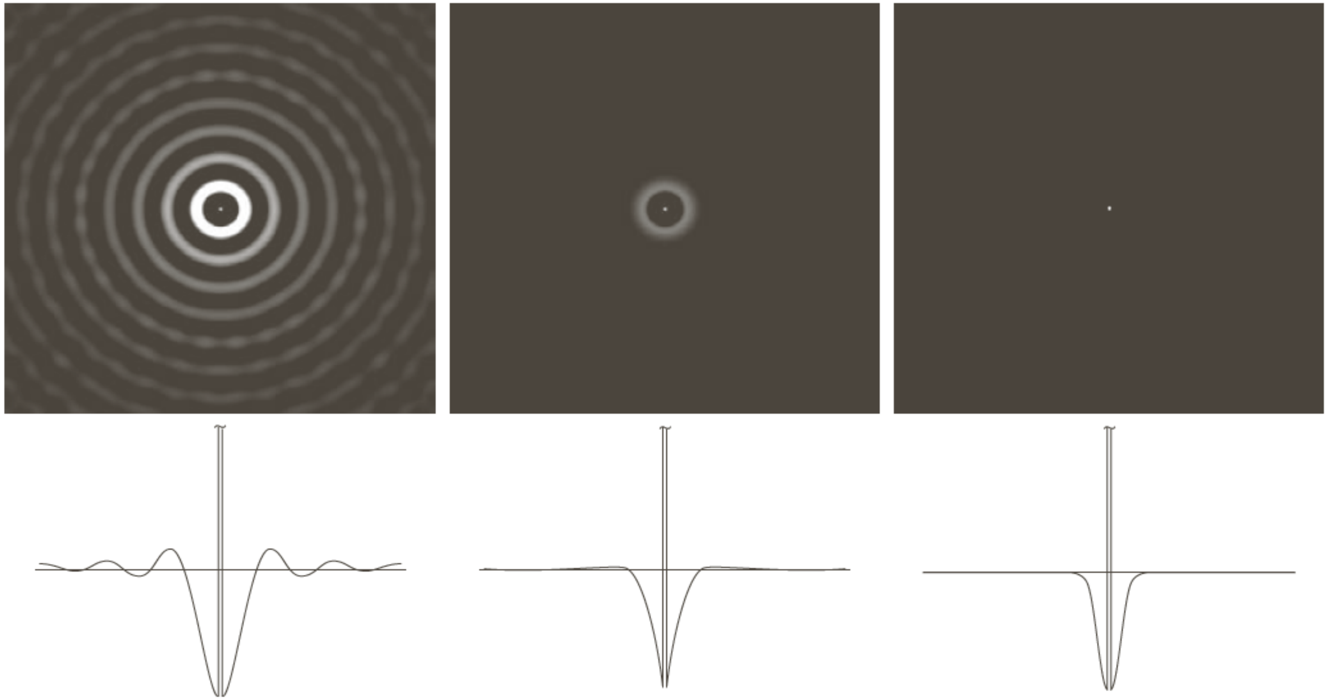


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

Image Sharpening



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.

Image Sharpening

- Ideal high-pass filters

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

a b c

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

Image Sharpening

- Butterworth high-pass filters

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

a b c

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

Image Sharpening

- Gaussian high-pass filters

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

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.

Image Sharpening

- The Laplacian in the frequency domain

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

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

Sharpening Spatial Filters

- The Laplacian
  - Laplacian operator

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

  - X-direction

$$\frac{\partial^2 f}{\partial x^2} = f\left(x+1, y\right) + f\left(x-1, y\right) - 2f\left(x, y\right)$$

  - Y-direction

$$\frac{\partial^2 f}{\partial y^2} = f\left(x, y+1\right) + f\left(x, y-1\right) - 2f\left(x, y\right)$$

Sharpening Spatial Filters

- The discrete Laplacian of two variables

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37
(a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

Image Sharpening

- Enhancement is achieved using the equation

$$g(x,y) = f(x,y) + c\left[\nabla^2 f(x,y)\right]$$

- In frequency domain

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

Image Sharpening

- High-boost filtering
  - Spatial domain:

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

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

  - Frequency domain

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

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

Image Sharpening

- High-frequency-emphasis filtering:

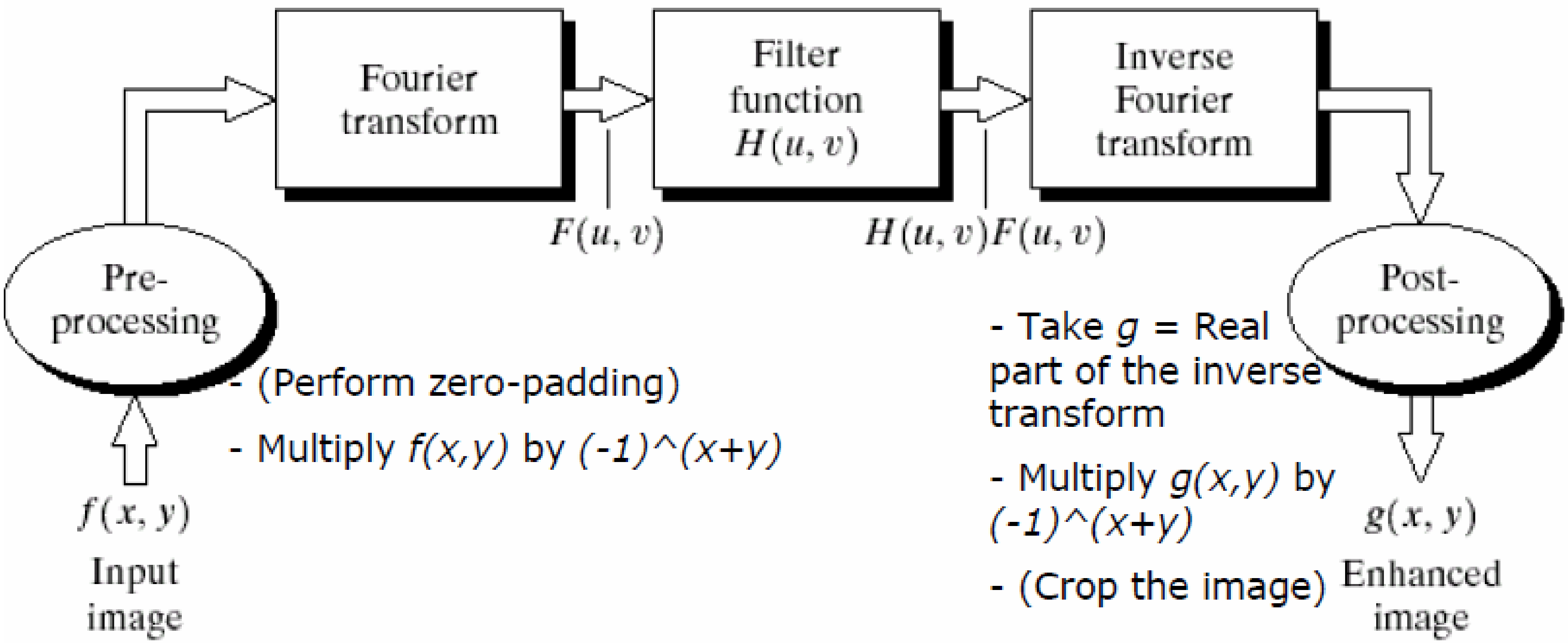
$$g(x,y) = \mathcal{F}^{-1}\left\{1 + k * H_{HP}(u,v)F(u,v)\right\}$$

Image Sharpening

- A slightly more general formulation of high-frequency-emphasis filtering

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

Take another look



Important things are to be repeated for 3 times !

Homework One

Page327: problem 4.8(b), 4.22, 4.28, 4.33, 4.35

Thank You!