Digital Image Processing

Lecture # 5

Spatial & Frequency Domain Image Enhancement

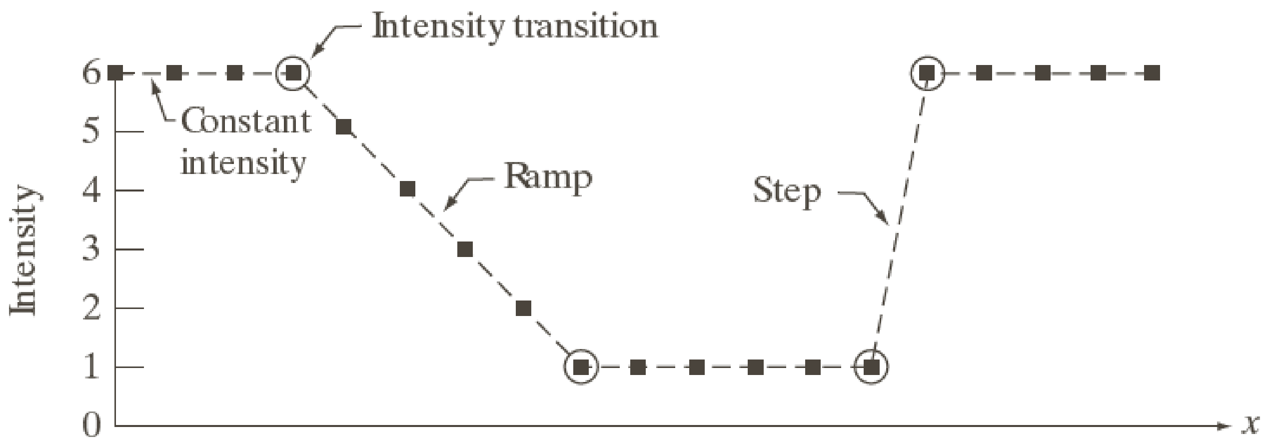
Sharpening Spatial Filters

Previously we have looked at smoothing filters which remove fine detail

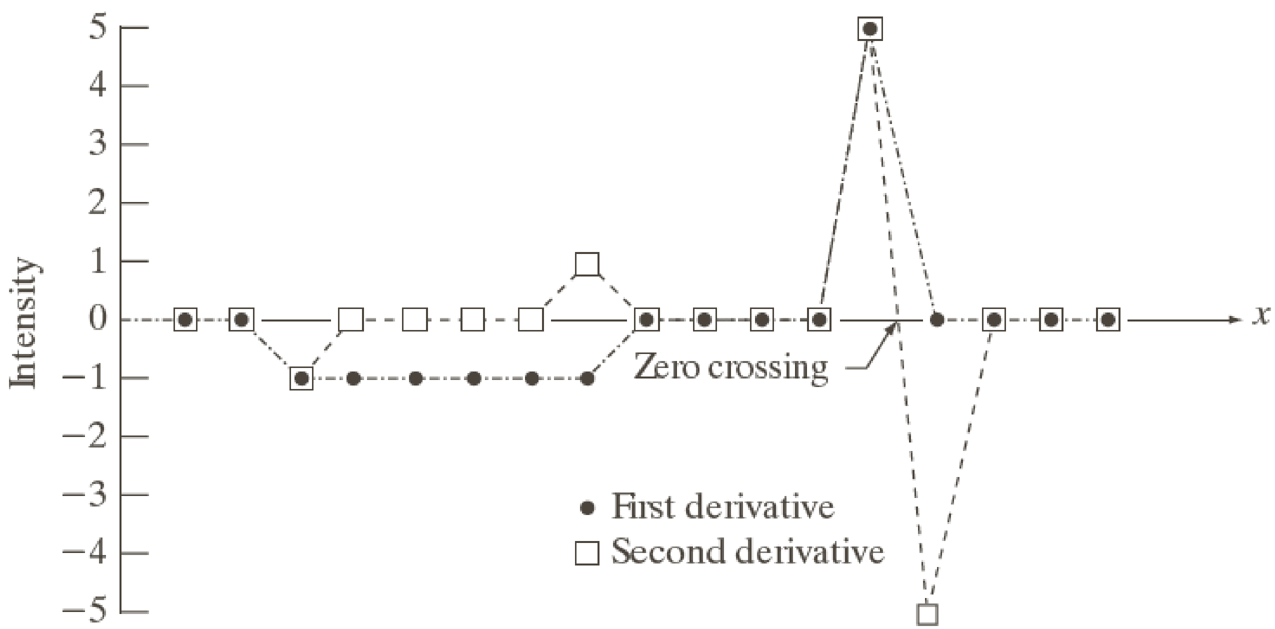
Sharpening spatial filters seek to highlight fine detail

- Remove blurring from images
- Highlight edges

Sharpening filters are based on *spatial differentiation*

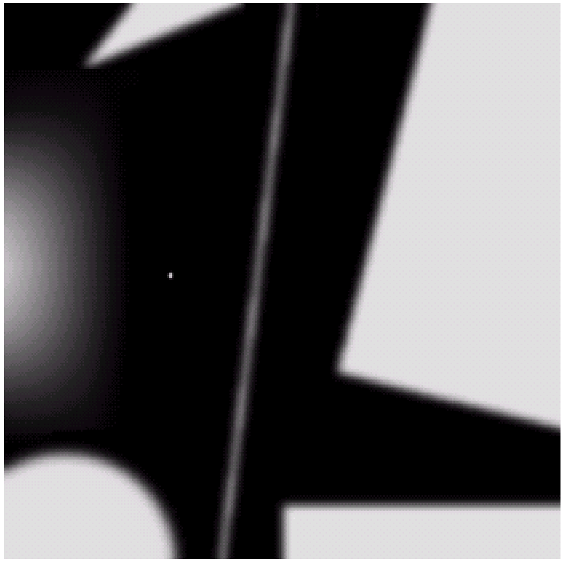


Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6 → x
1st derivative		0	0	−1	−1	−1	−1	−1	0	0	0	0	0	5	0	0	0	0	
2nd derivative		0	0	−1	0	0	0	0	1	0	0	0	0	5	−5	0	0	0	

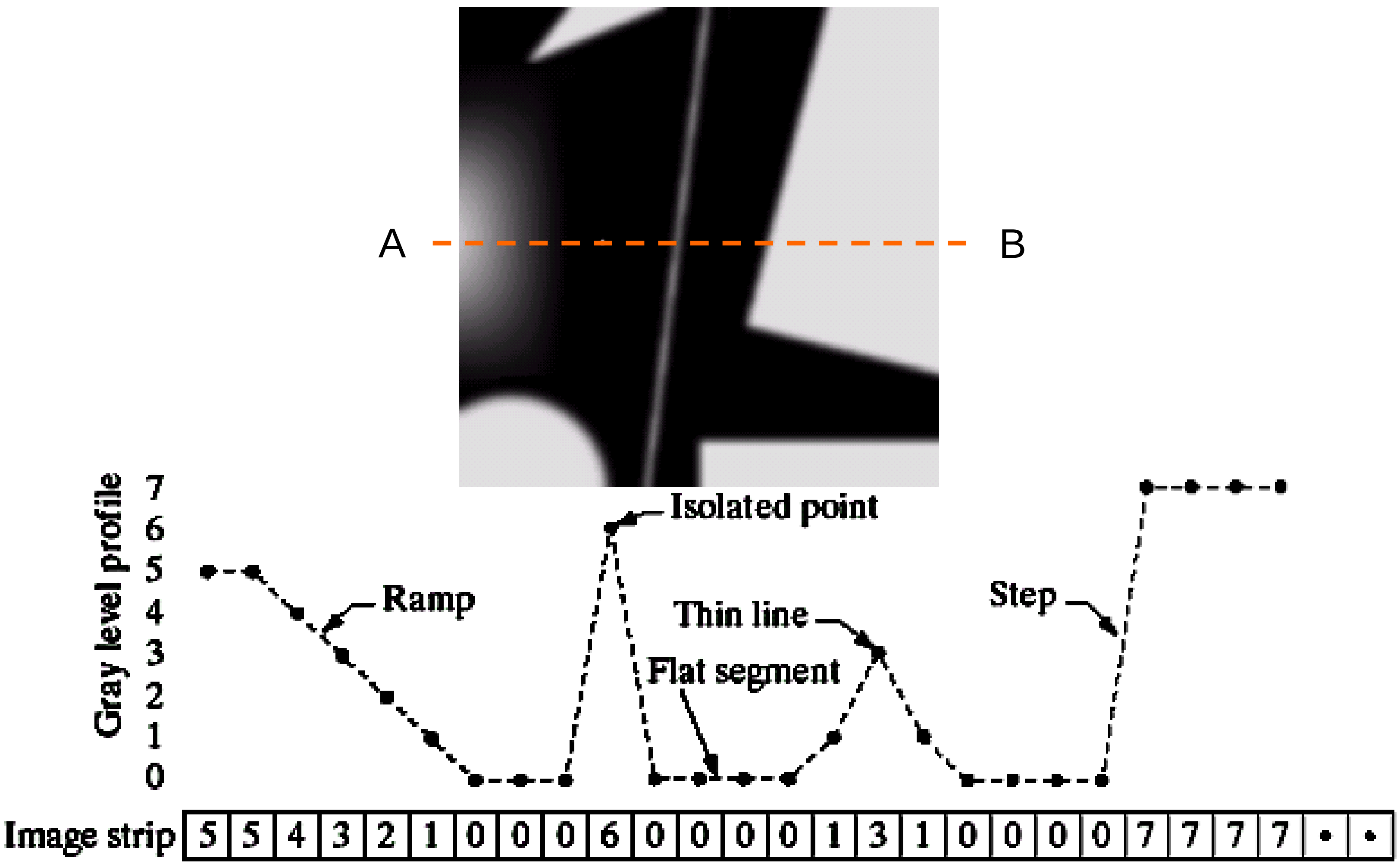


Spatial Differentiation

- Let's consider a simple 1 dimensional example



Spatial Differentiation



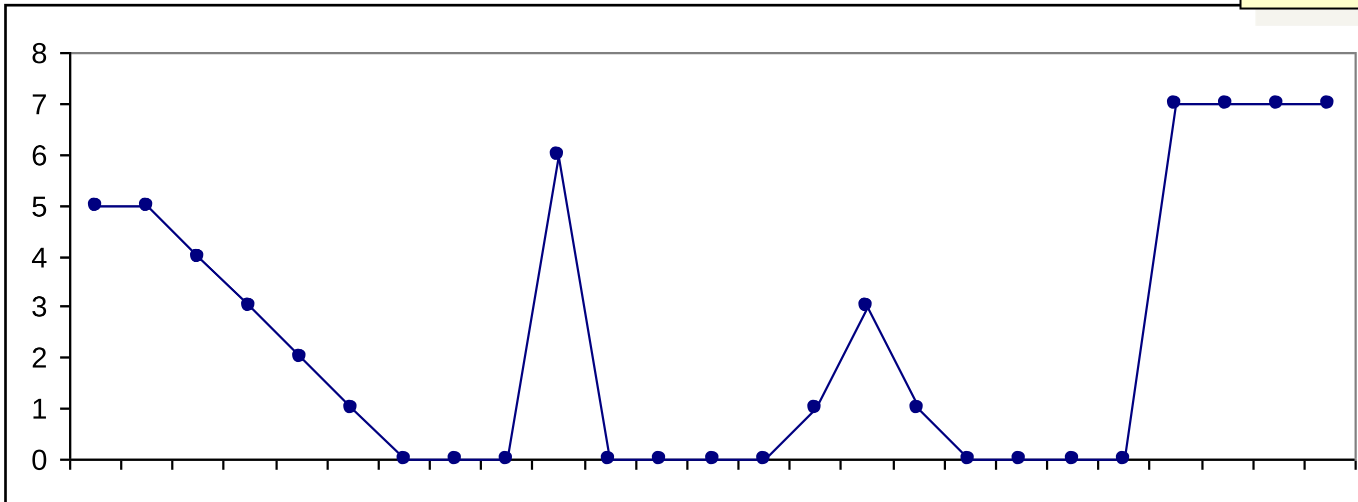
1st Derivative

The 1st derivative of a function is given by:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

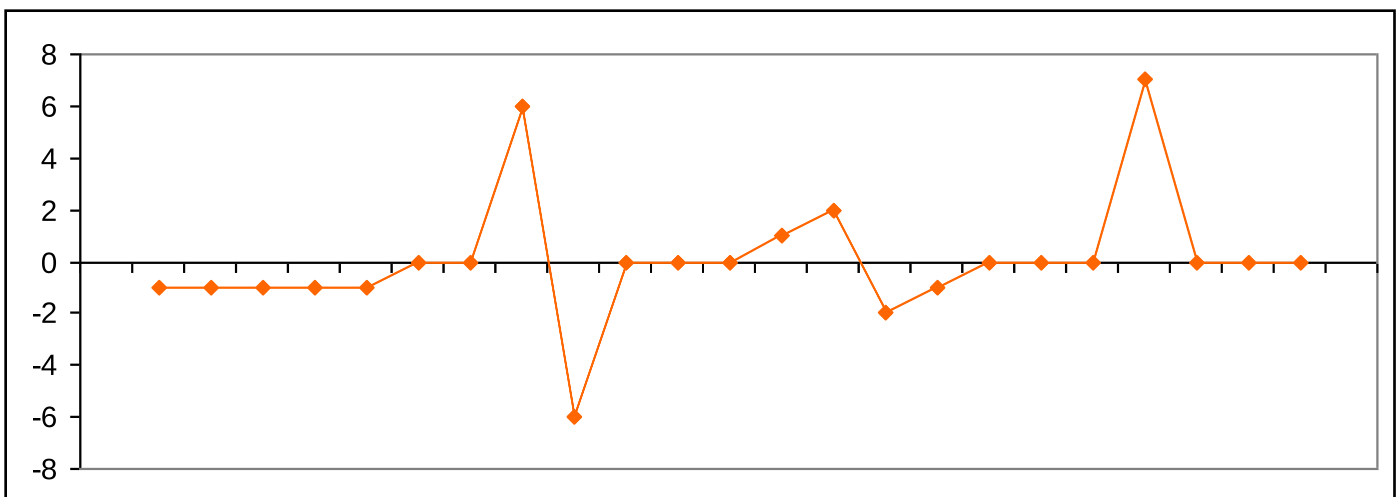
Its just the difference between subsequent values and measures the rate of change of the function

1st Derivative



5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7

	-1	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	0	0	



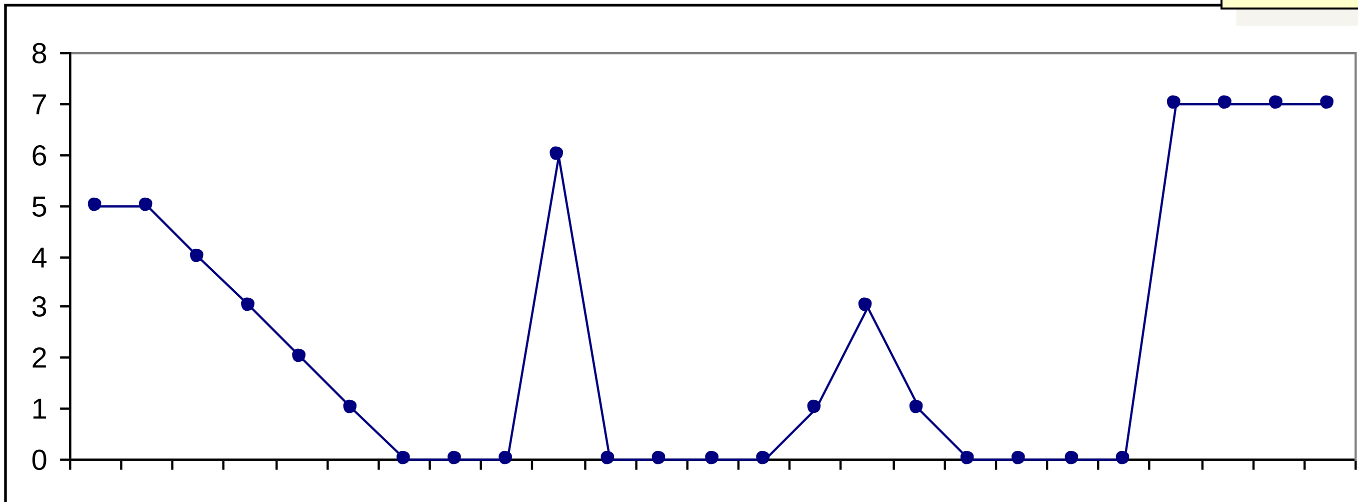
2^{nd} Derivative

The 2nd derivative of a function is given by:

Simply takes into account the values both before and after the current value

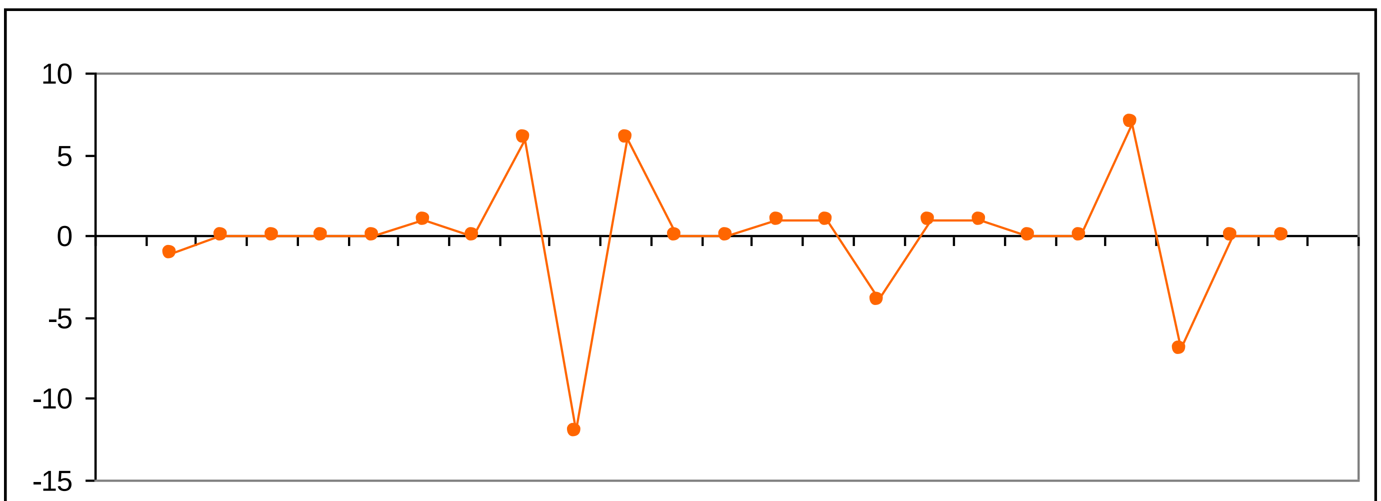
$$\frac{\partial^2 f}{\partial^2 x} = f(x+1) + f(x-1) - 2f(x)$$

2^{nd} Derivative



5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7

	-1	0	0	0	0	1	0	6	-12	6	0	0	1	1	-4	1	1	0	0	7	-7	0	0	



2^{nd} Derivative for Image Enhancement

The 2nd derivative is more useful for image enhancement than the 1st derivative - *Stronger response to fine detail*

We will come back to the 1st order derivative later on

The first sharpening filter we will look at is the *Laplacian*

Laplacian Filter

The Laplacian is defined as follows:

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

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

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

Laplacian Filter

So, the Laplacian can be given as follows:

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

Can we implement it using a filter/ mask?

0	1	0
1	-4	1
0	1	0

Laplacian Filter

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

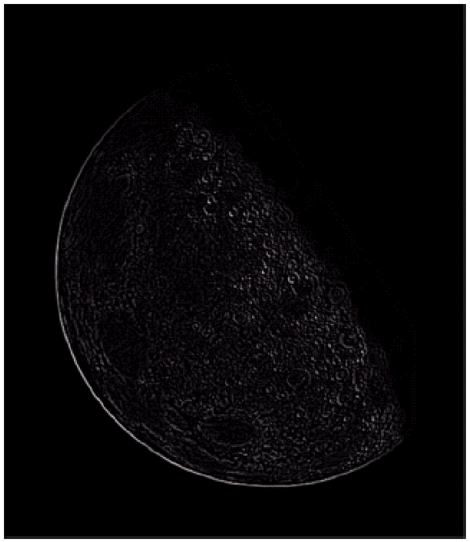
a	b
c	d

FIGURE 3.39 (a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

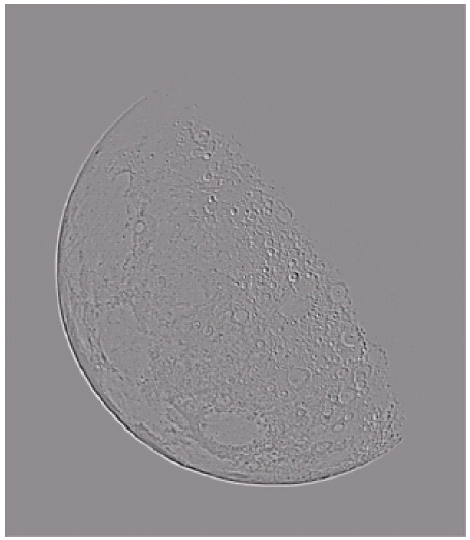
Laplacian Filter

Applying the Laplacian to an image we get a new image that highlights edges and other discontinuities

Original Image



Laplacian Filtered Image

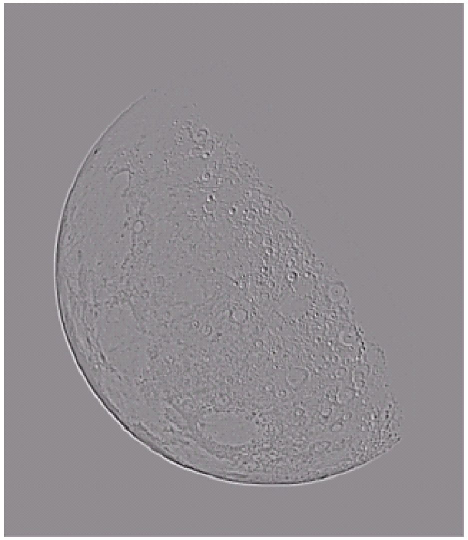


Laplacian Filtered Image Scaled for Display

Laplacian Image Enhancement

The result of a Laplacian filtering is not an enhanced image

To generate the final enhanced image



Laplacian
Filtered Image
Scaled for Display

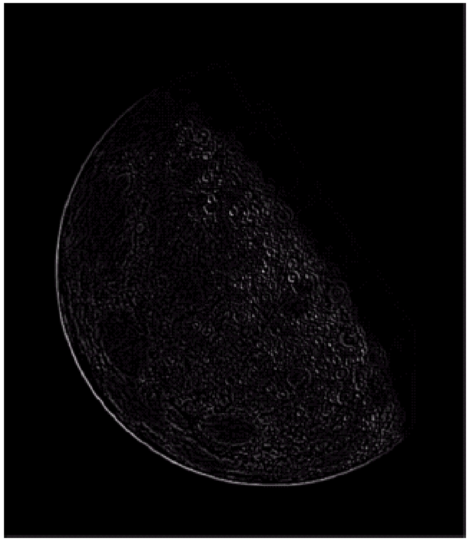
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f, & w_5 < 0 \\ f(x,y) + \nabla^2 f, & w_5 > 0 \end{cases}$$

Laplacian Image Enhancement

Original
Image

\-



Laplacian
Filtered Image

=

Sharpened
Image

In the final sharpened image edges and fine detail are much more obvious

Laplacian Image Enhancement

Simplified Image Enhancement

- The entire enhancement can be combined into a single filtering operation

$$
\begin{aligned}
g(x,y) &= f(x,y) - \nabla^2 f \\
&= f(x,y) - [f(x+1,y) + f(x-1,y) \\
&\qquad + f(x,y+1) + f(x,y-1) \\
&\qquad - 4f(x,y)]
\end{aligned}
$$

Simplified Image Enhancement

- The entire enhancement can be combined into a single filtering operation

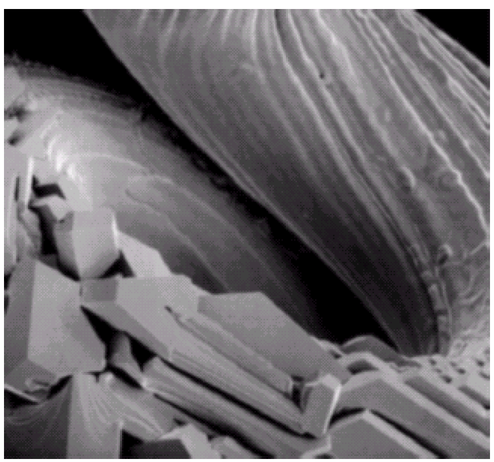
$$g(x,y) = f(x,y) - \nabla^2 f$$

$$= 5f(x,y) - f(x+1,y) - f(x-1,y) - f(x,y+1) - f(x,y-1)$$

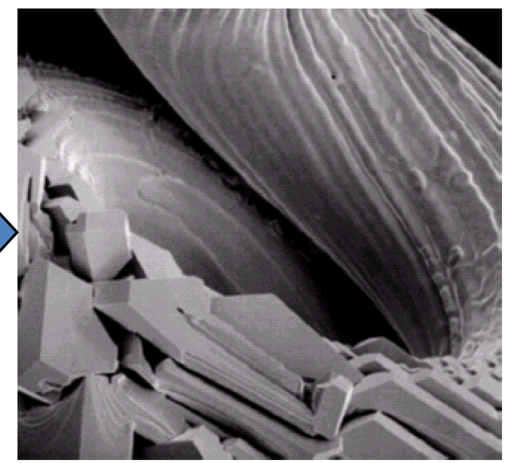
0	-1	0
-1	5	-1
0	-1	0

Simplified Image Enhancement

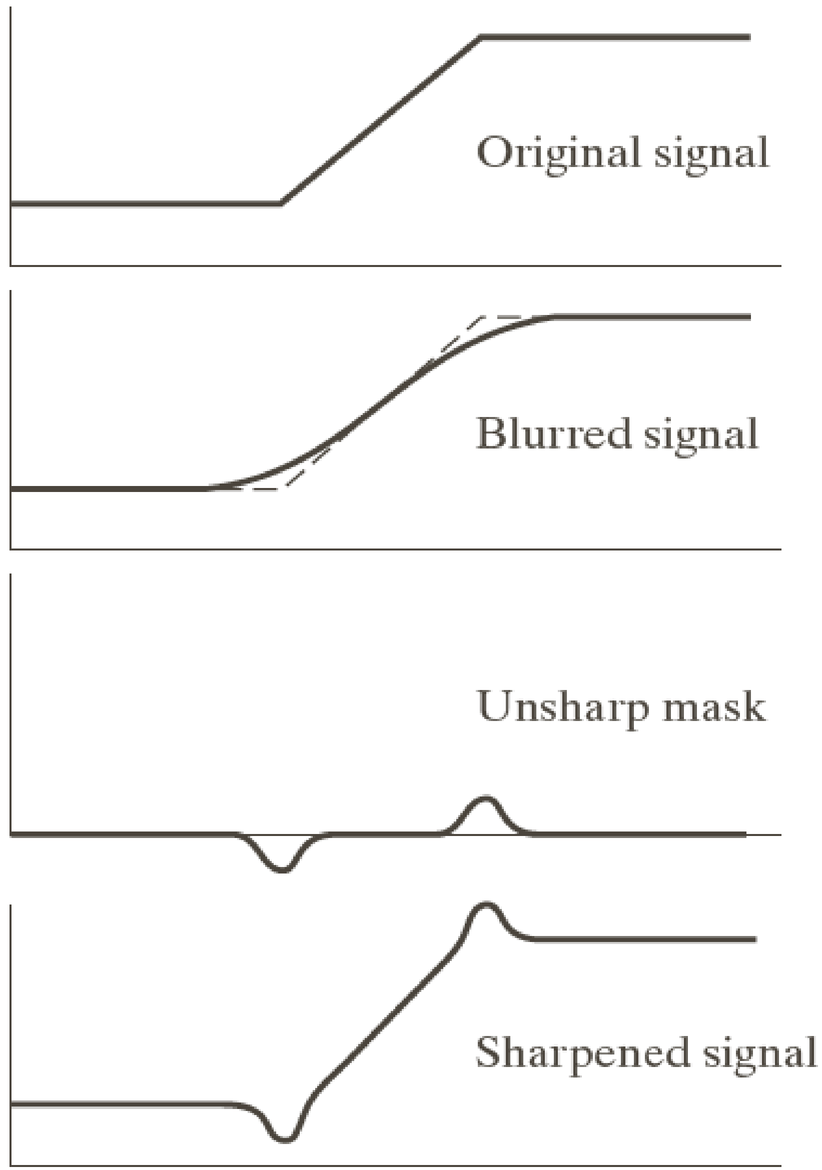
- This gives us a new filter which does the whole job for us in one step



0	-1	0
-1	5	-1
0	-1	0



Unsharp Masking



DIP-XE

DIP-XE

DIP-XE

DIP-XE

DIP-XE

Use of first derivatives for image enhancement: The Gradient

- The **gradient** of a function f(x,y) is defined as

$$\nabla \mathrm{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

Gradient Operators

Sobel Operator

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

Extract horizontal edges

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

Extract vertical edges

Emphasize more the current point (y direction)

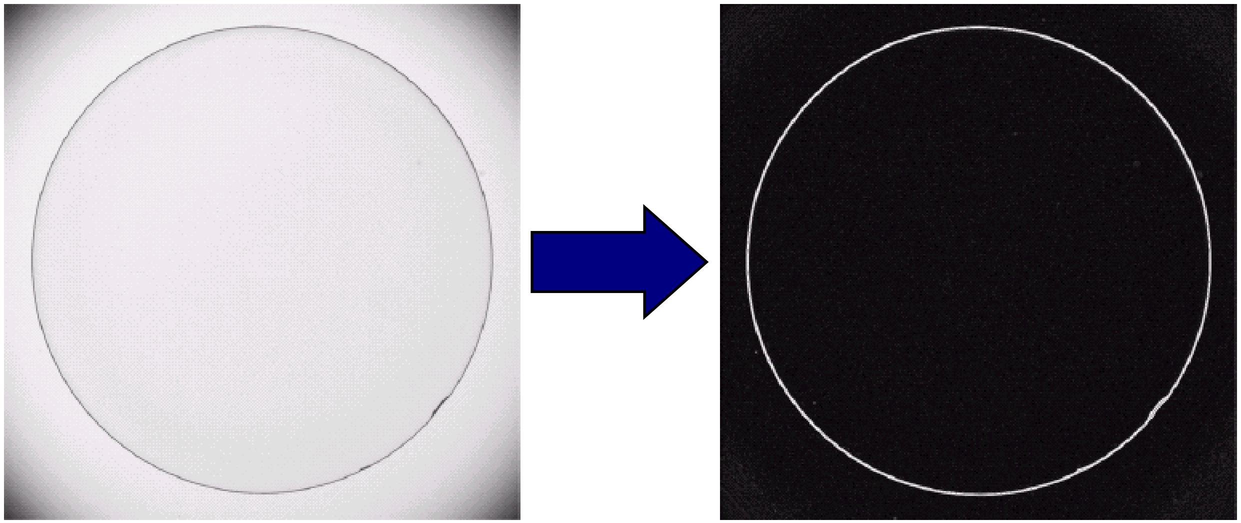
$$\nabla f \approx \left|(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)\right| + \left|(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)\right|$$

Emphasize more the current point (x direction)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Pixel Arrangement

Sobel Operator: Example



An image of a contact lens which is enhanced in order to make defects more obvious

Sobel filters are typically used for edge detection

Combining Spatial Enhancement Methods

Successful image enhancement is typically not achieved using a single operation

Rather we combine a range of techniques in order to achieve a final result

This example will focus on enhancing the bone scan

Combining Spatial Enhancement Methods

(a)

Laplacian filter of bone scan (a)

(b)

Sharpened version of bone scan achieved by subtracting (a) and (b)

(c)

Sobel filter of bone scan (a)

(d)

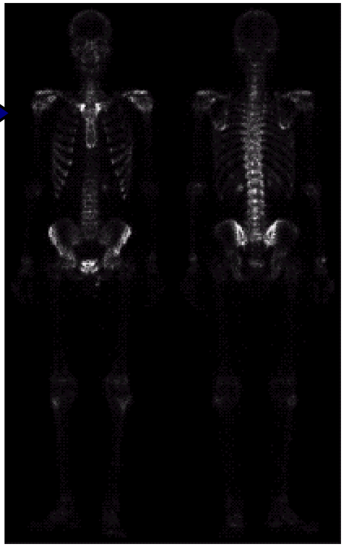
Combining Spatial Enhancement Methods

(e)

Image (d) smoothed with a 5*5 averaging filter

(f)

The product of (c) and (e) which will be used as a mask

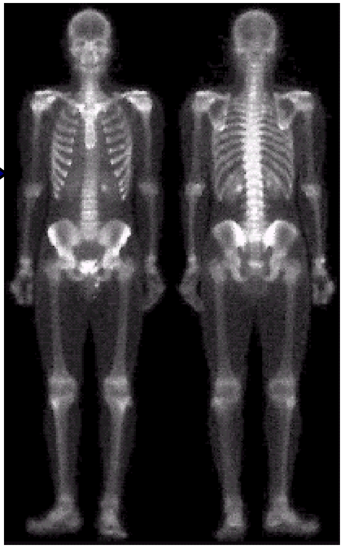


(g)

Sharpened image which is sum of (a) and (f)

(h)

Result of applying a power-law trans. to (g)



Compare the original and final images

Image Enhancement in Frequency Domain

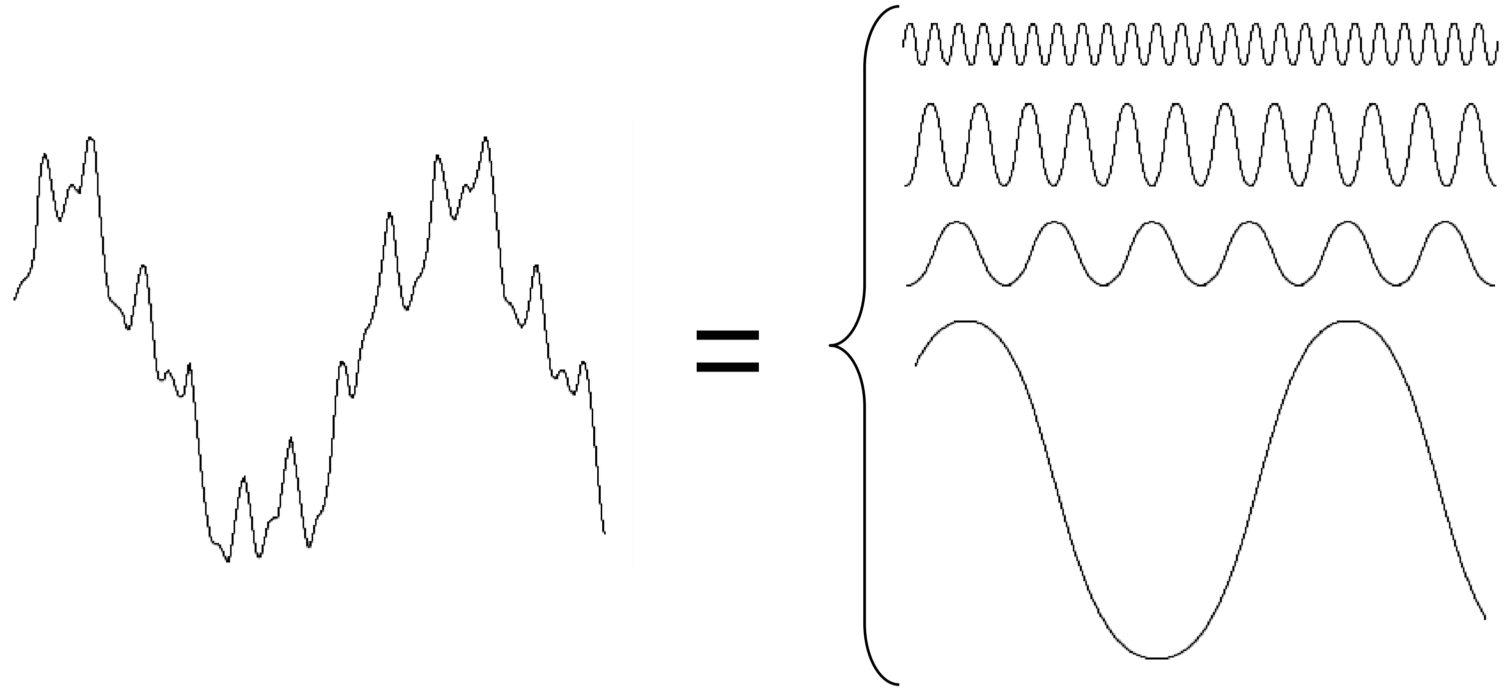
Joseph Fourier (1768 – 1830)

- Most famous for his work *"La Théorie Analitique de la Chaleur"* published in 1822
- Translated into English in 1878: *"The Analytic Theory of Heat"*

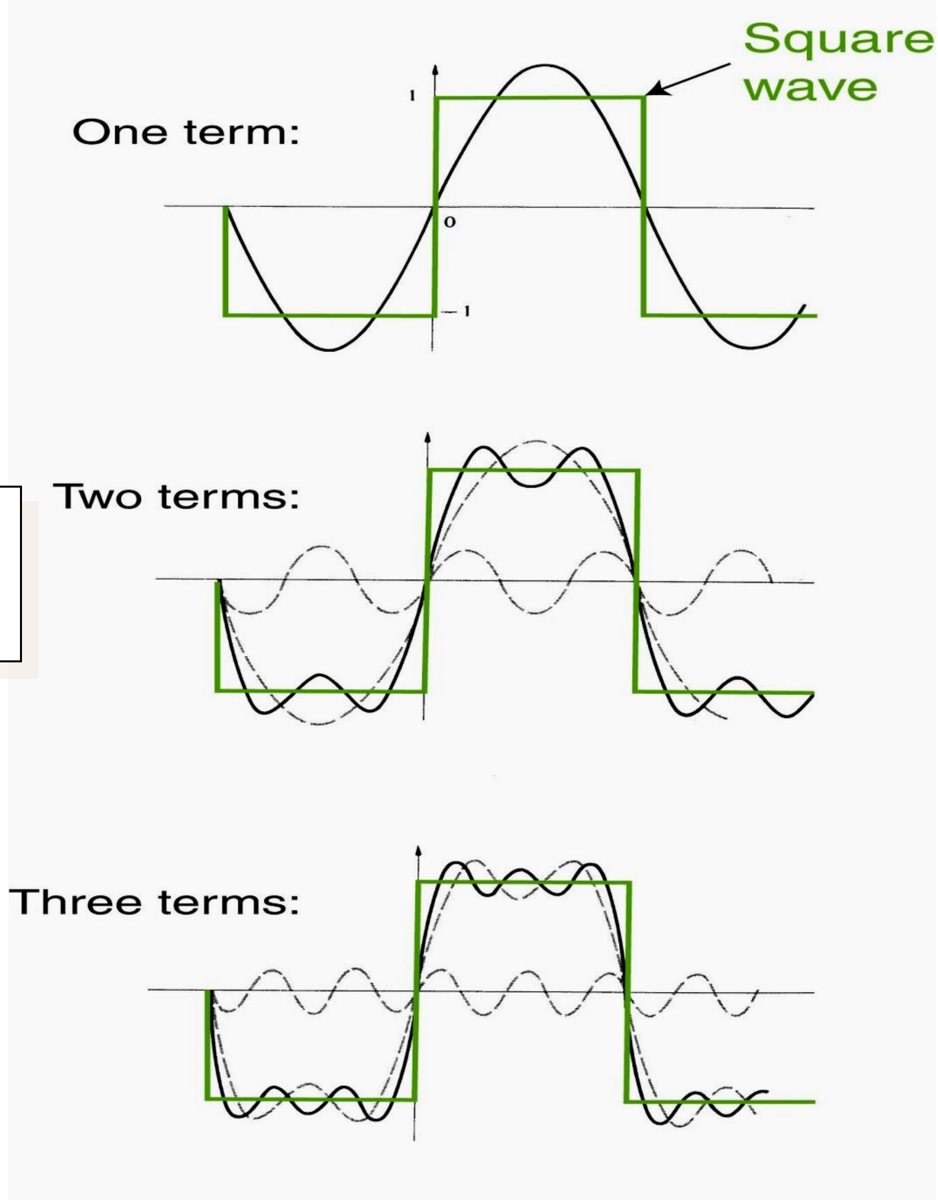
Nobody paid much attention when the work was first published
One of the most important mathematical theories in modern engineering

The big idea …

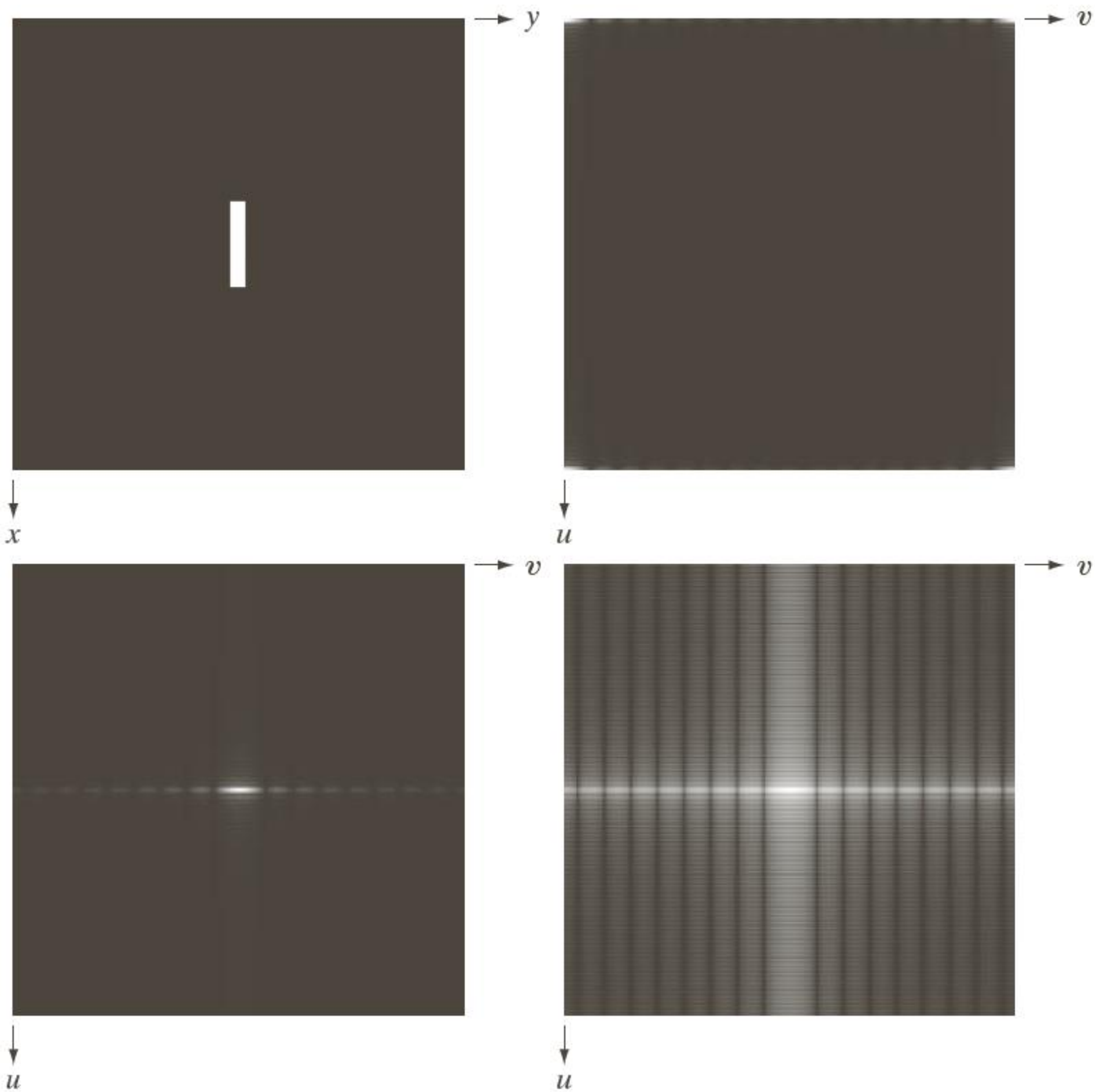
Any function that periodically repeats itself can be expressed as a sum of sines and cosines of different frequencies each multiplied by a different coefficient – a *Fourier series*



The big idea…

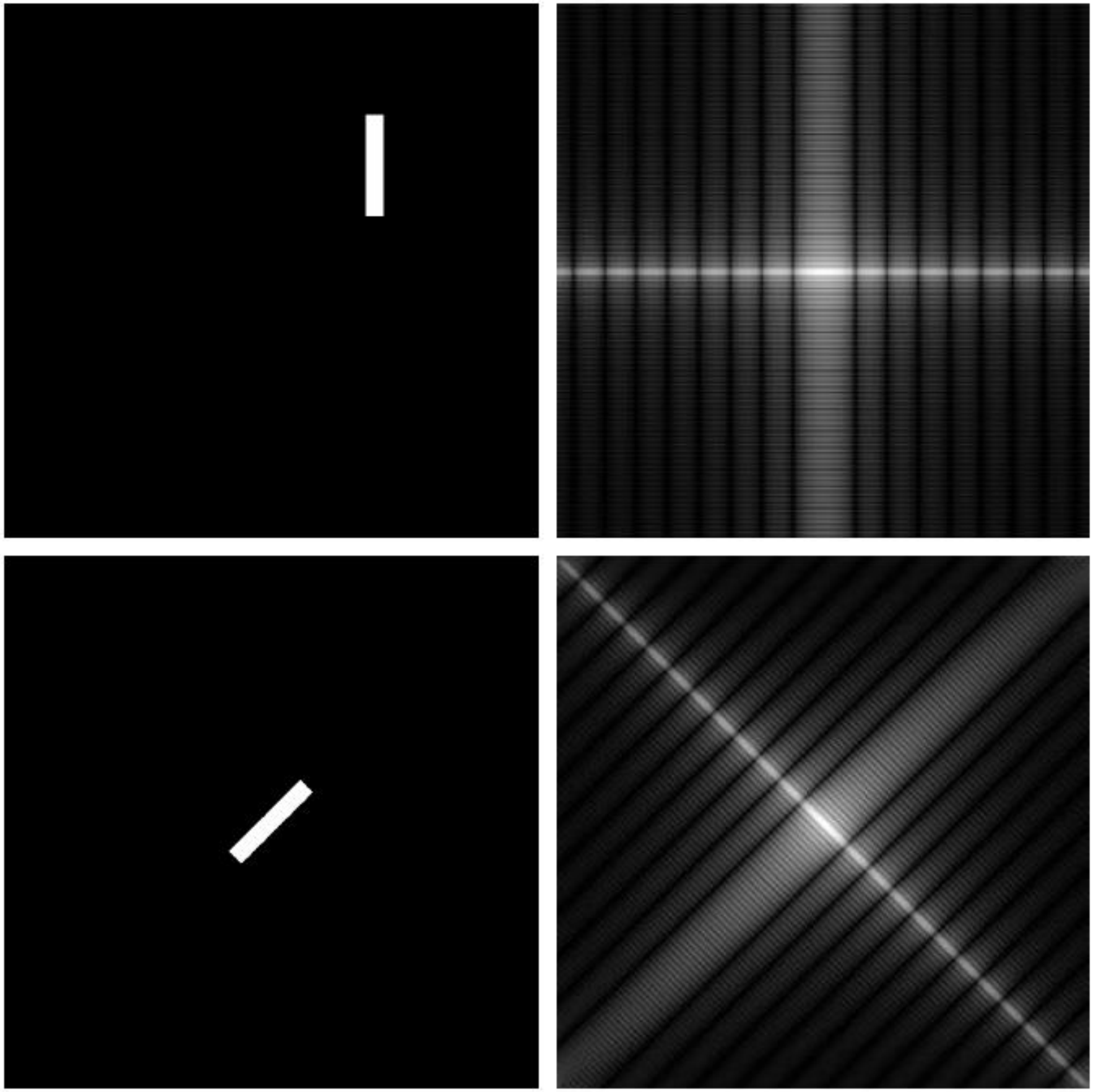


Approximating a square wave as the sum of sine waves



a b
c d

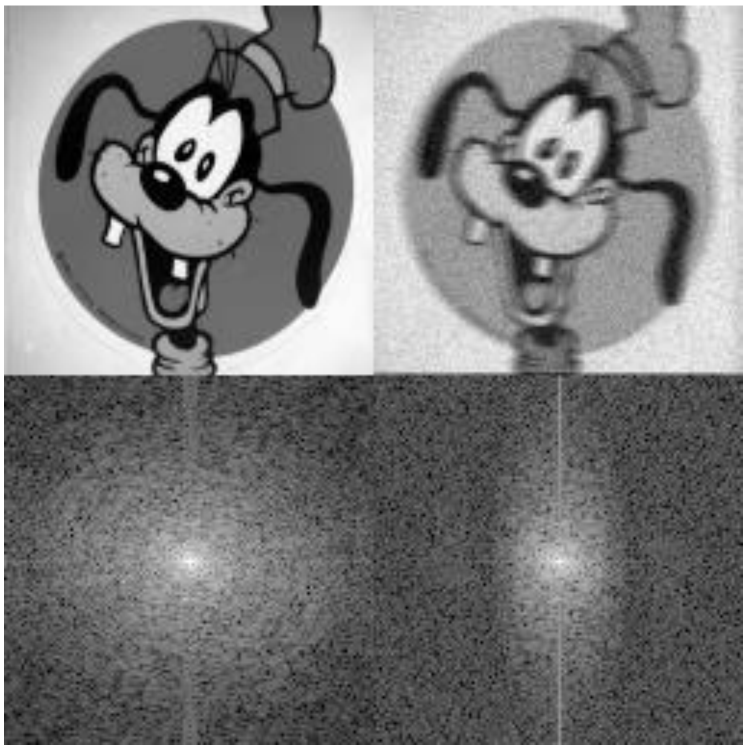
FIGURE 4.24 (a) Image. (b) Spectrum showing bright spots in the four corners. (c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate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

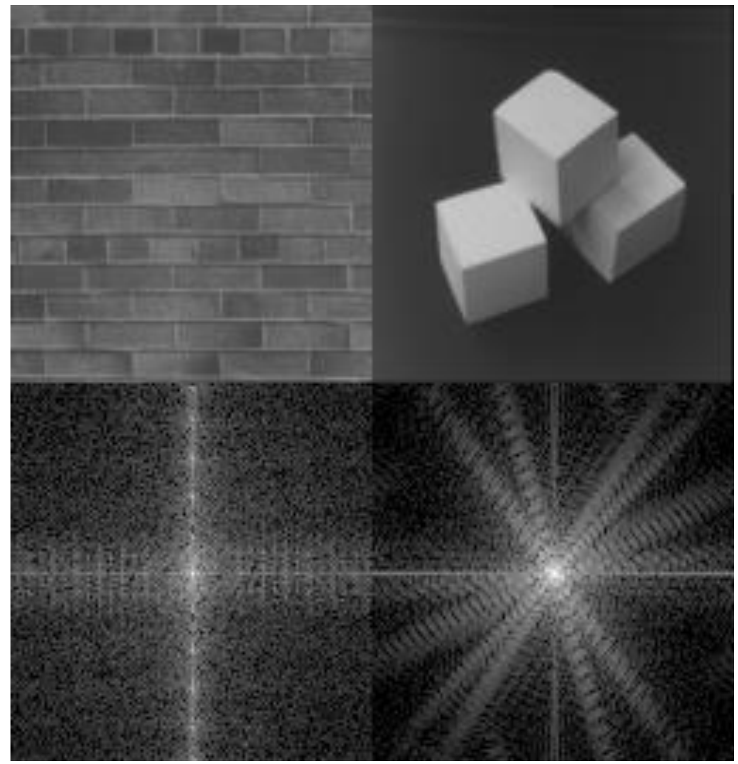


a	b
c	d

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).

Frequencies in Images





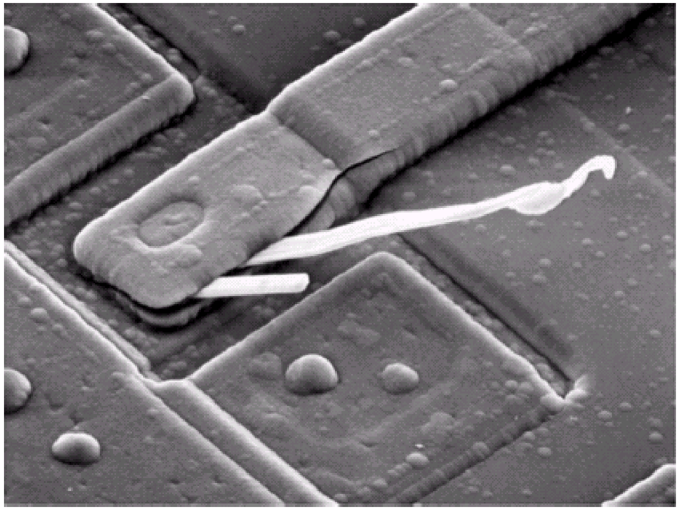
The Discrete Fourier Transform (DFT)

The *Discrete Fourier Transform* of $f(x, y)$, for x = 0, 1, 2…M-1 and y = 0,1,2…N-1, denoted by $F(u, v)$, is given by the equation:

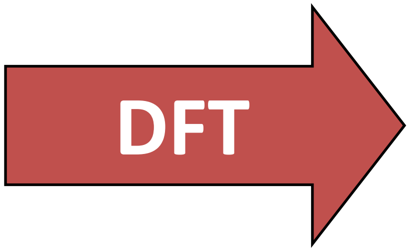
$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M + vy/N)}$$

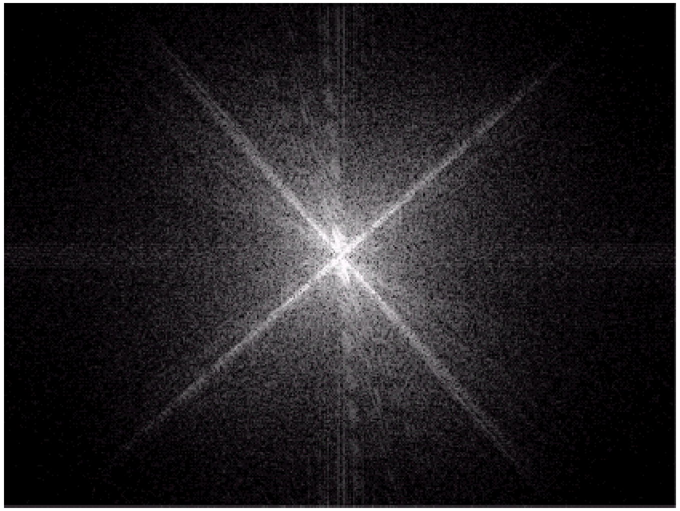
for u = 0, 1, 2…M-1 and v = 0, 1, 2…N-1.

DFT & Images

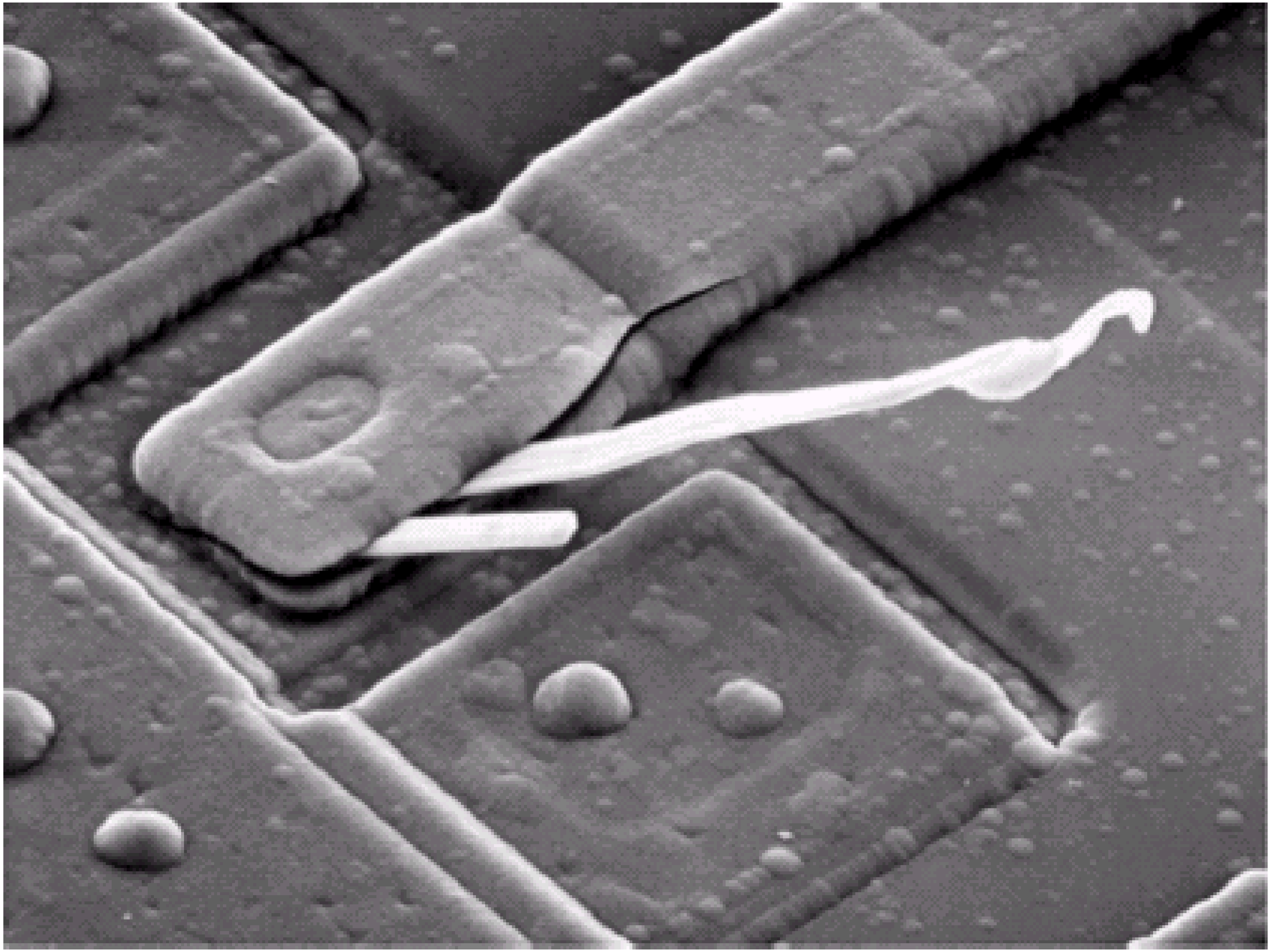


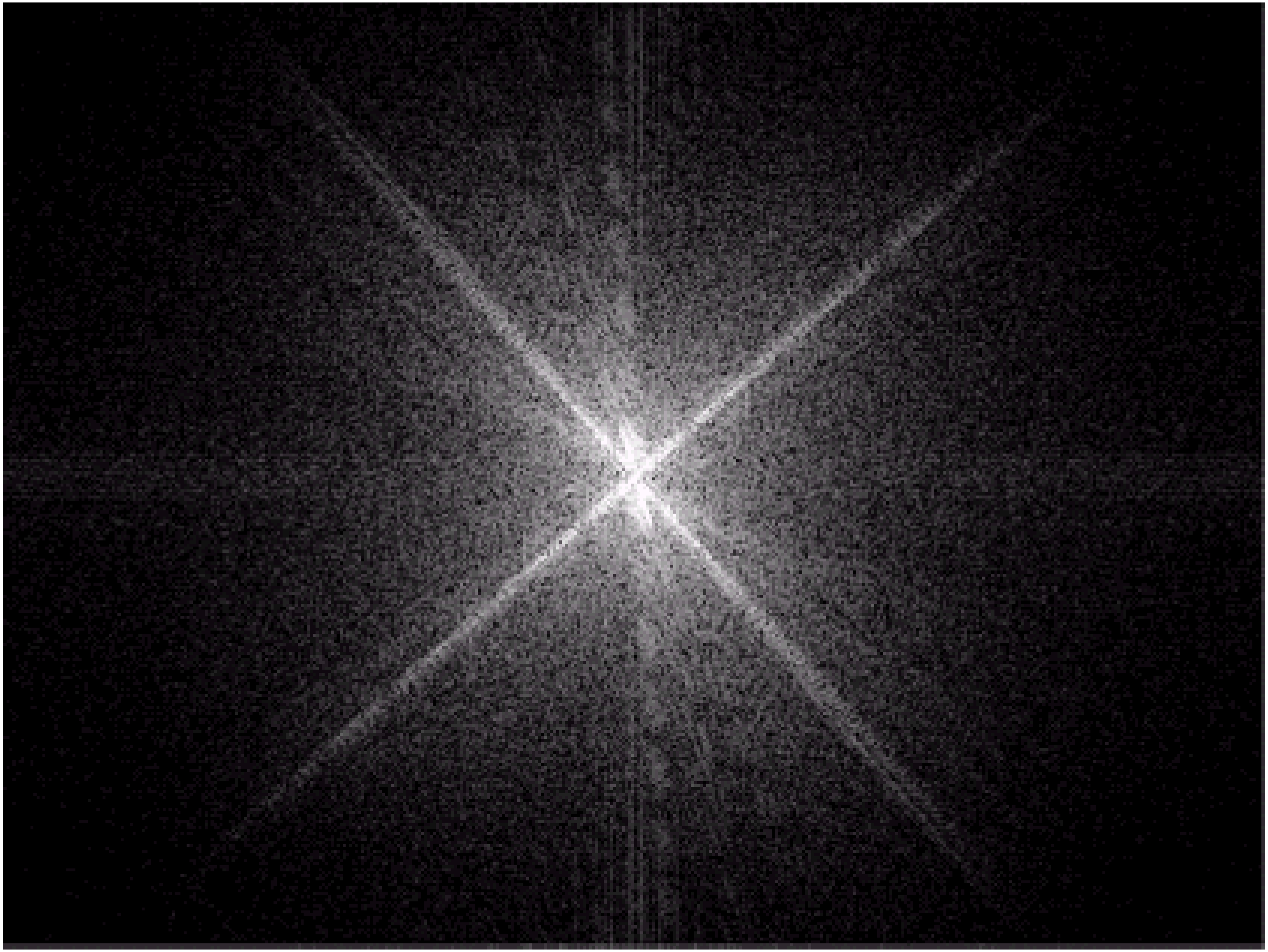
Scanning electron microscope image of an integrated circuit magnified ~2500 times





Fourier spectrum of the image





The Inverse DFT

It is really important to note that the Fourier transform is completely **reversible**

The inverse DFT is given by:

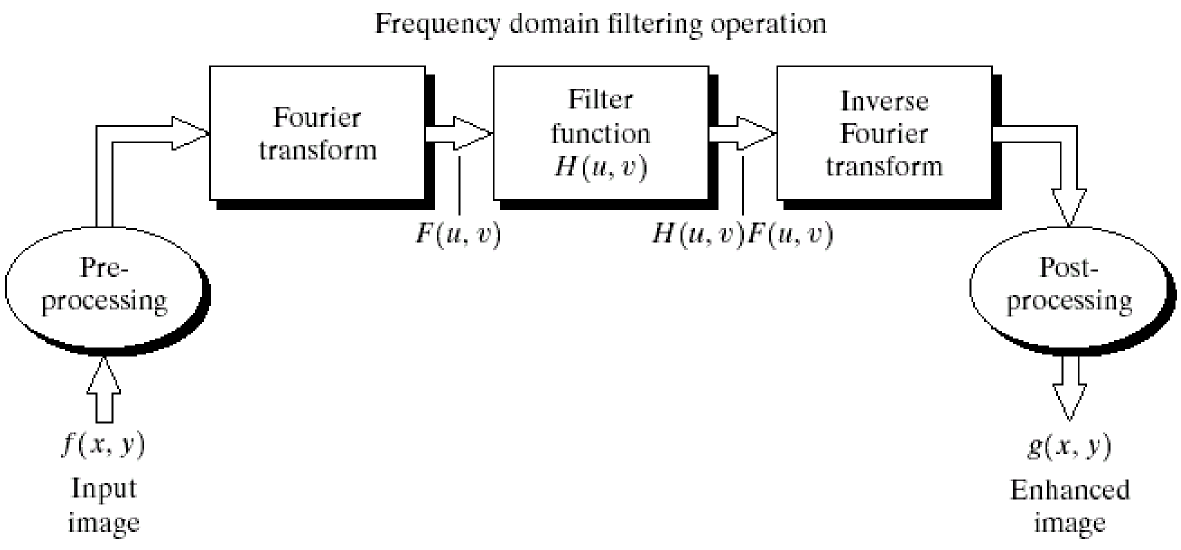
$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux/M + vy/N)}$$

for x = 0, 1, 2…M-1 and y = 0, 1, 2…N-1

The DFT and Image Processing

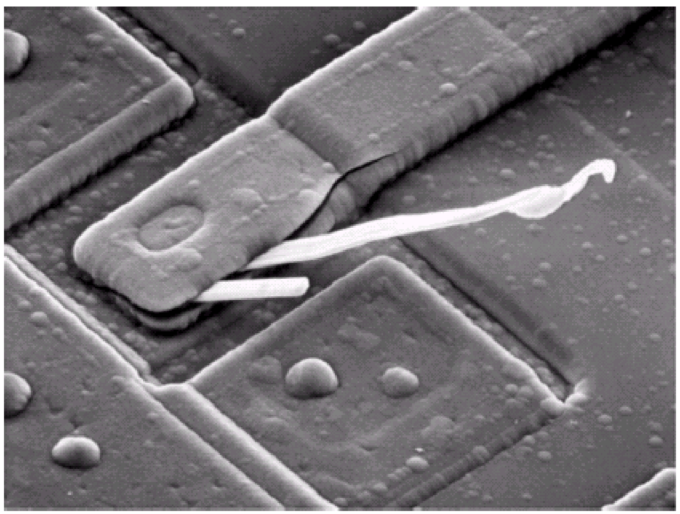
To filter an image in the frequency domain:

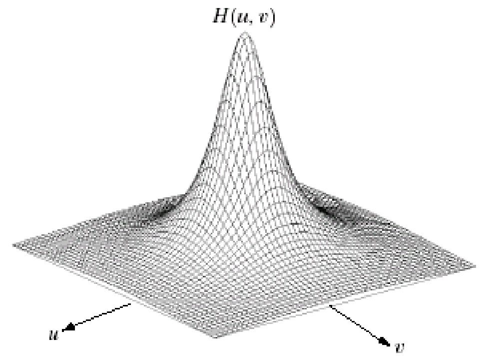
1. Compute $F(u,v)$ the DFT of the image
2. Multiply $F(u,v)$ by a filter function $H(u,v)$
3. Compute the inverse DFT of the result

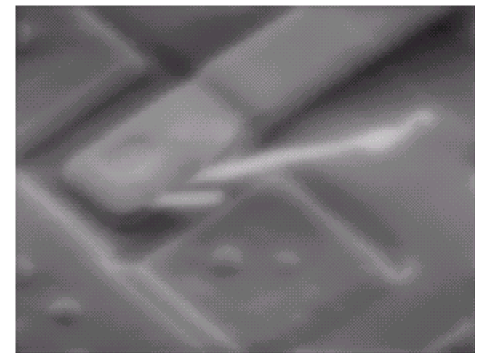


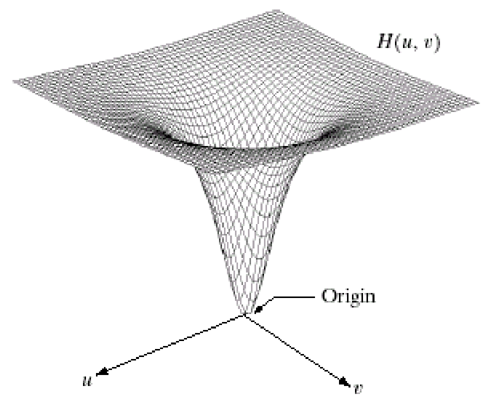
Some Basic Frequency Domain Filters

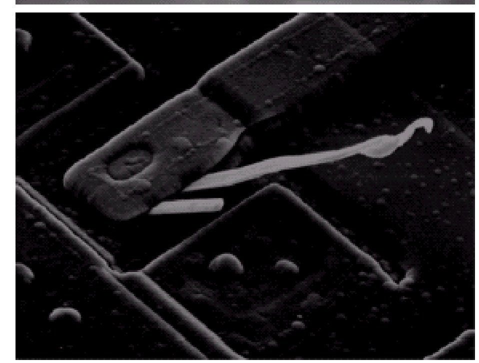
Low Pass Filter







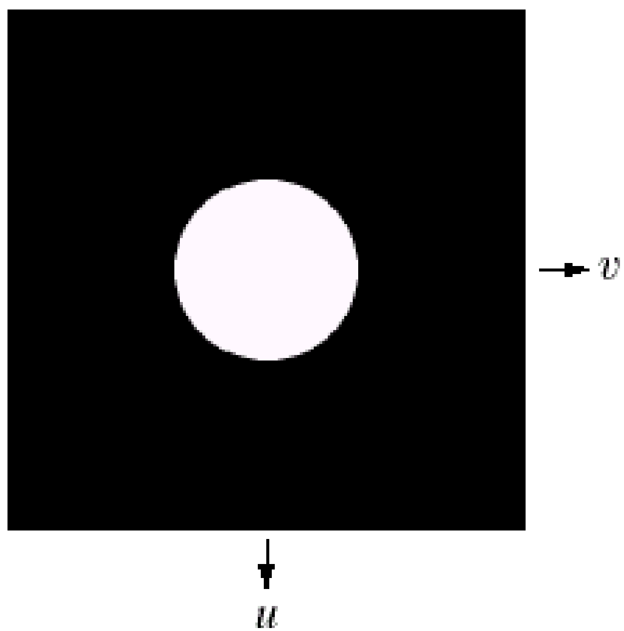


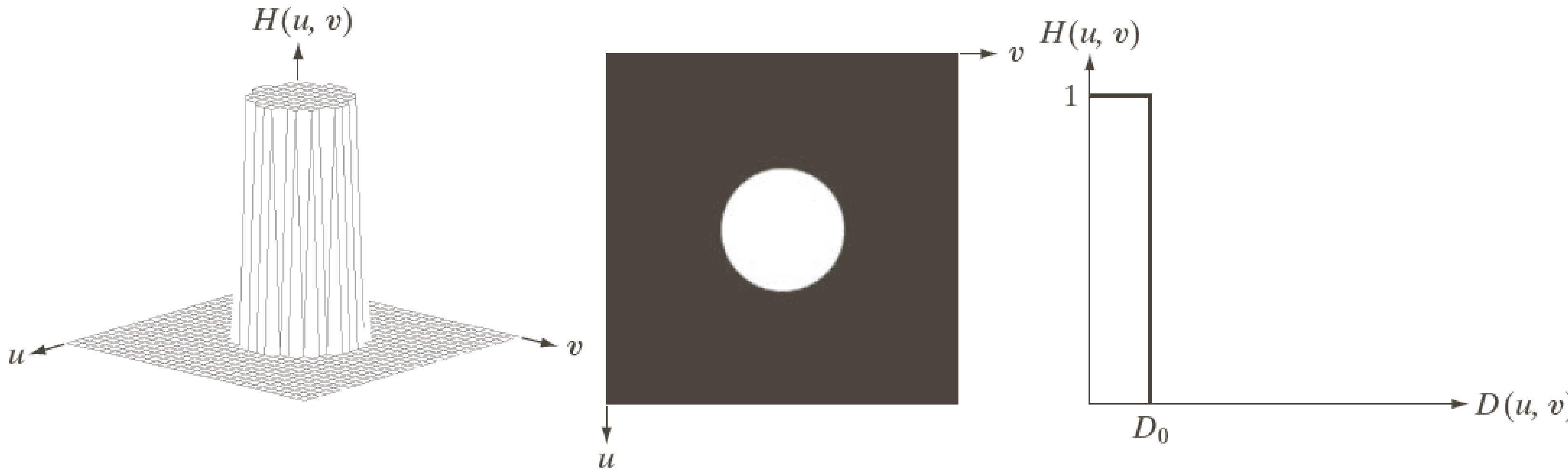


High Pass Filter

Ideal Low Pass Filter

Simply cut off all high frequency components that are a specified distance D_0 from the origin

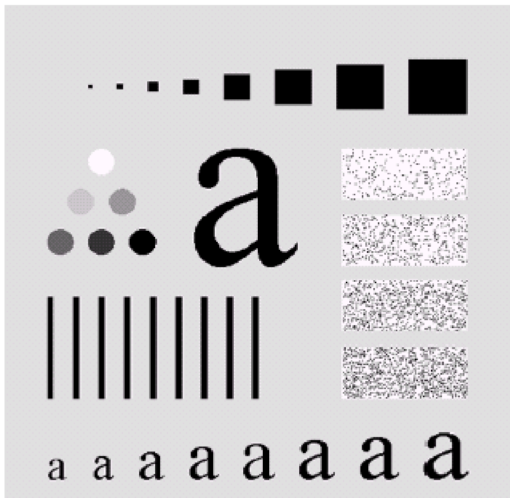


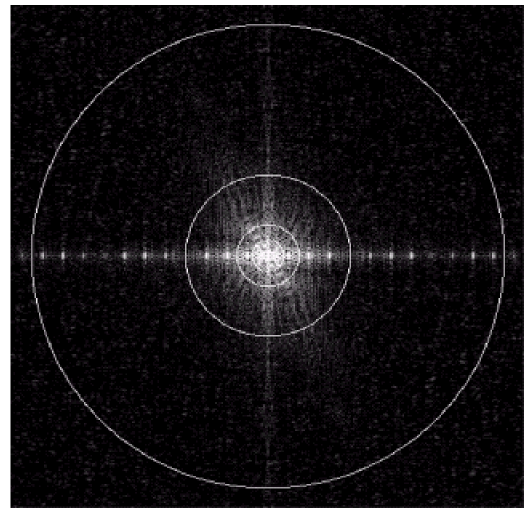


a b c

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

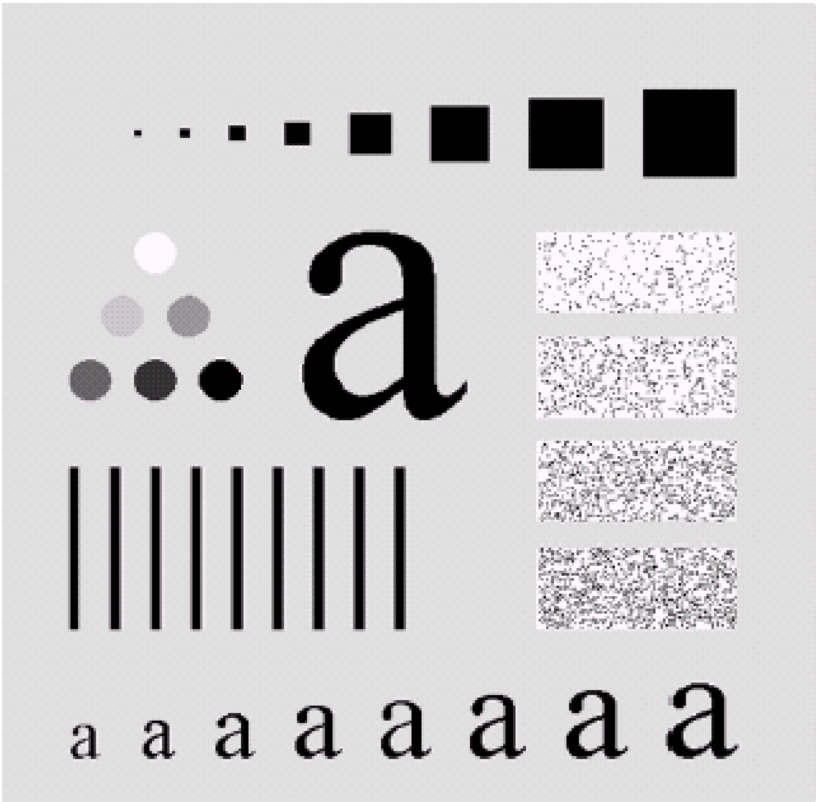
Ideal Low Pass Filter (cont…)



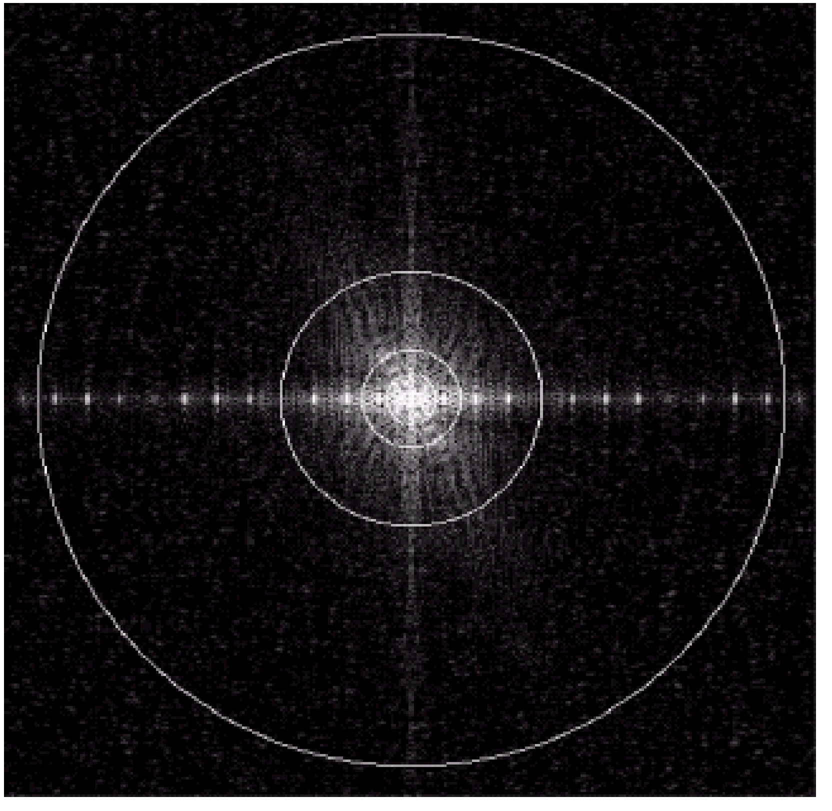


Above we show an image, it's Fourier spectrum and a series of ideal low pass filters of radius 5, 15, 30, 80 and 230 superimposed on top of it

Ideal Low Pass Filter (cont…)



Ideal Low Pass Filter (cont…)



Original image

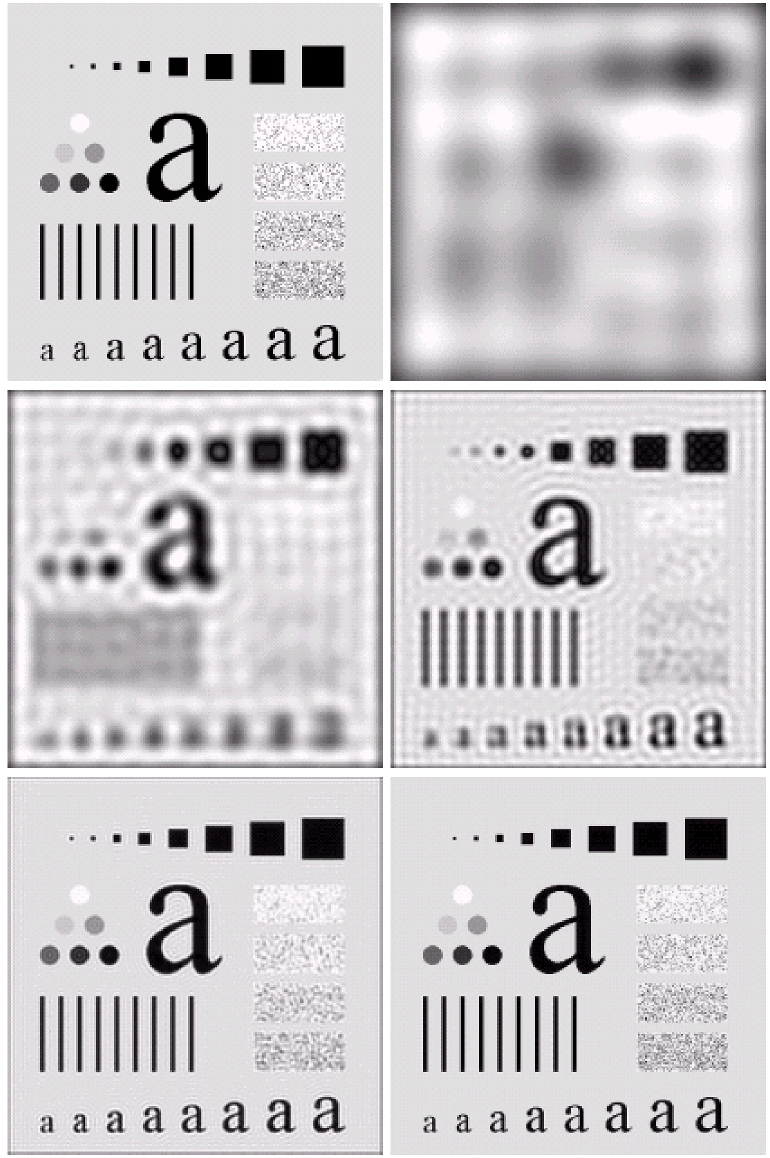
Result of filtering with ideal low pass filter of radius 5

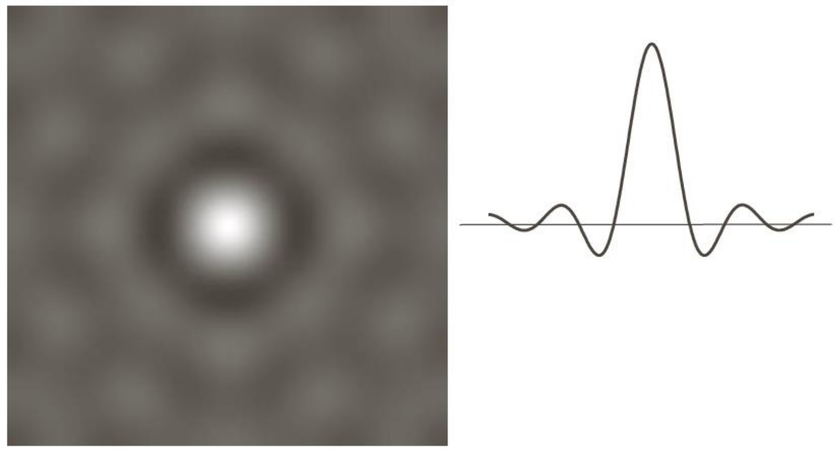
Result of filtering with ideal low pass filter of radius 15

Result of filtering with ideal low pass filter of radius 30

Result of filtering with ideal low pass filter of radius 80

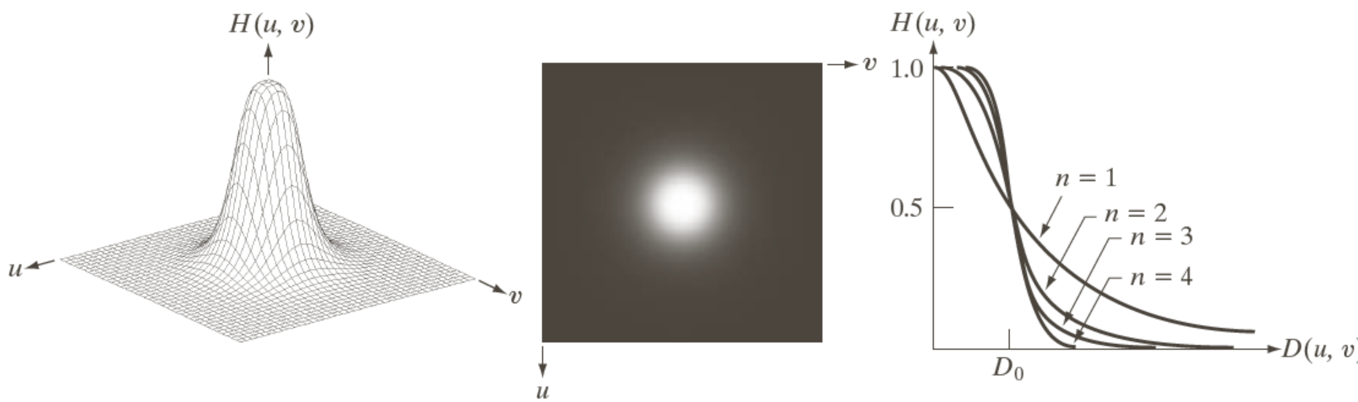
Result of filtering with ideal low pass filter of radius 230





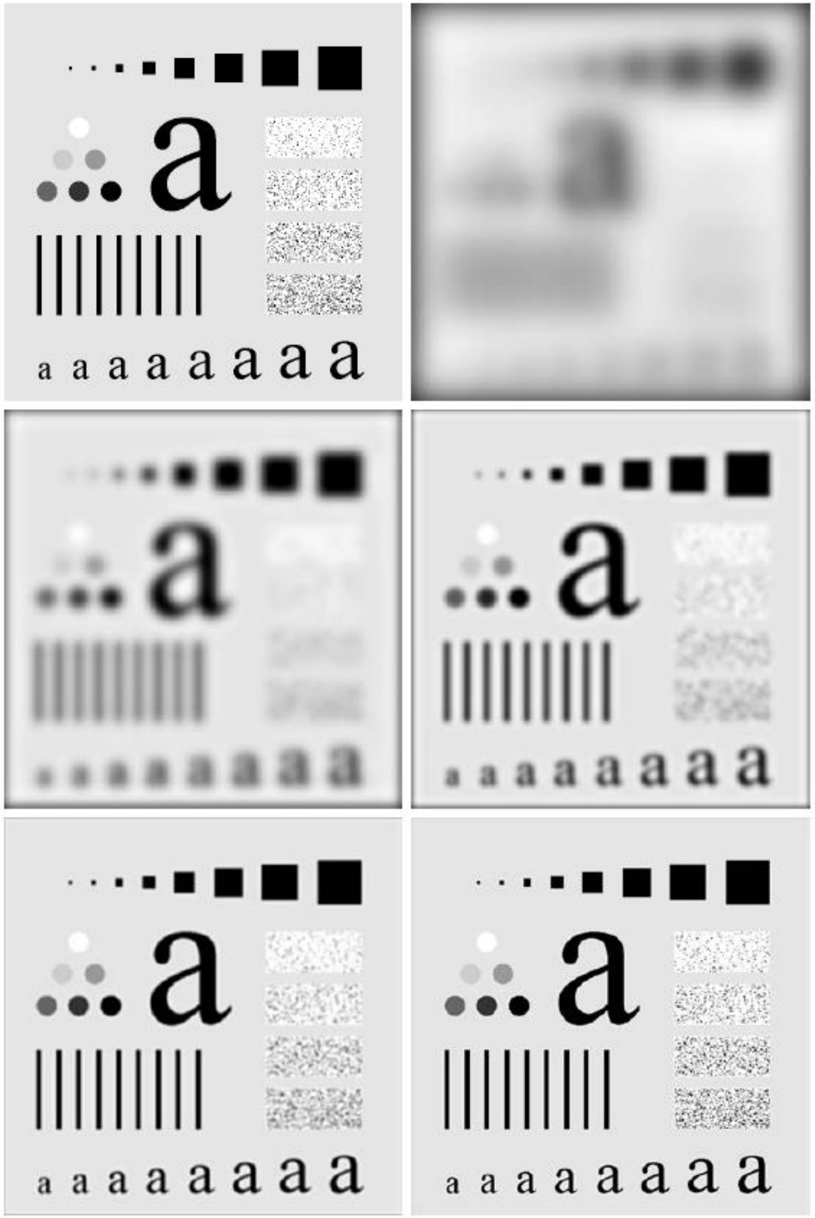
a b

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.



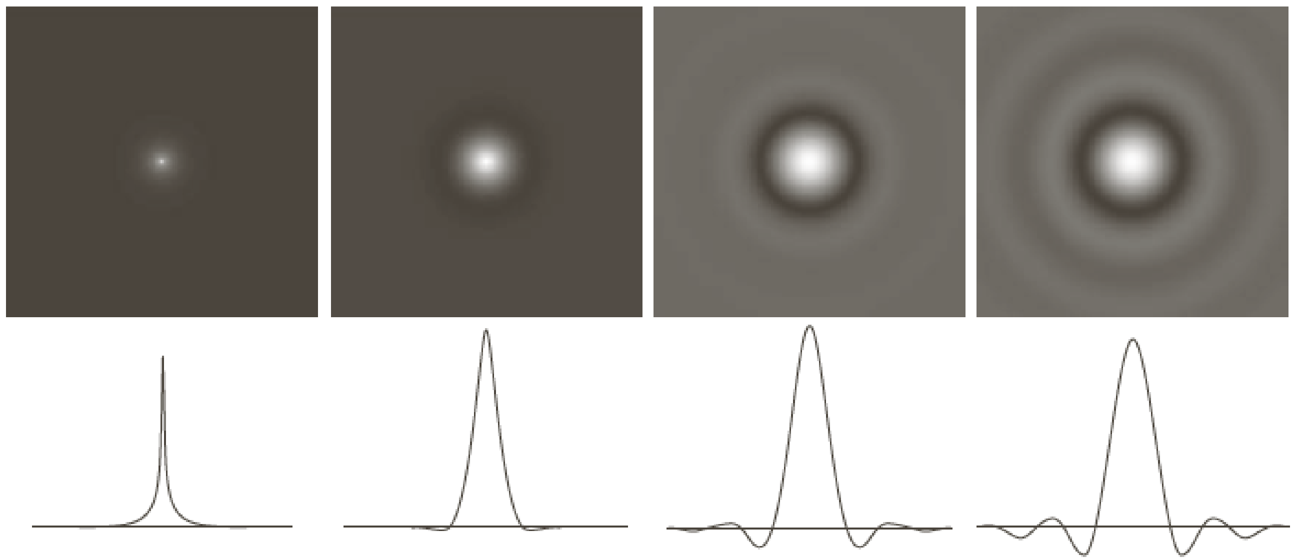
a b c

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.



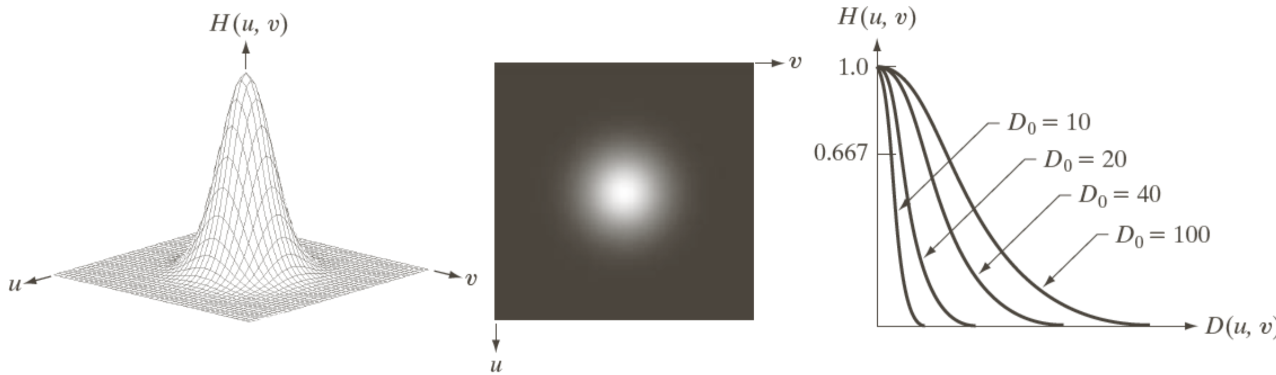
a b
c d
e f

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.



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.

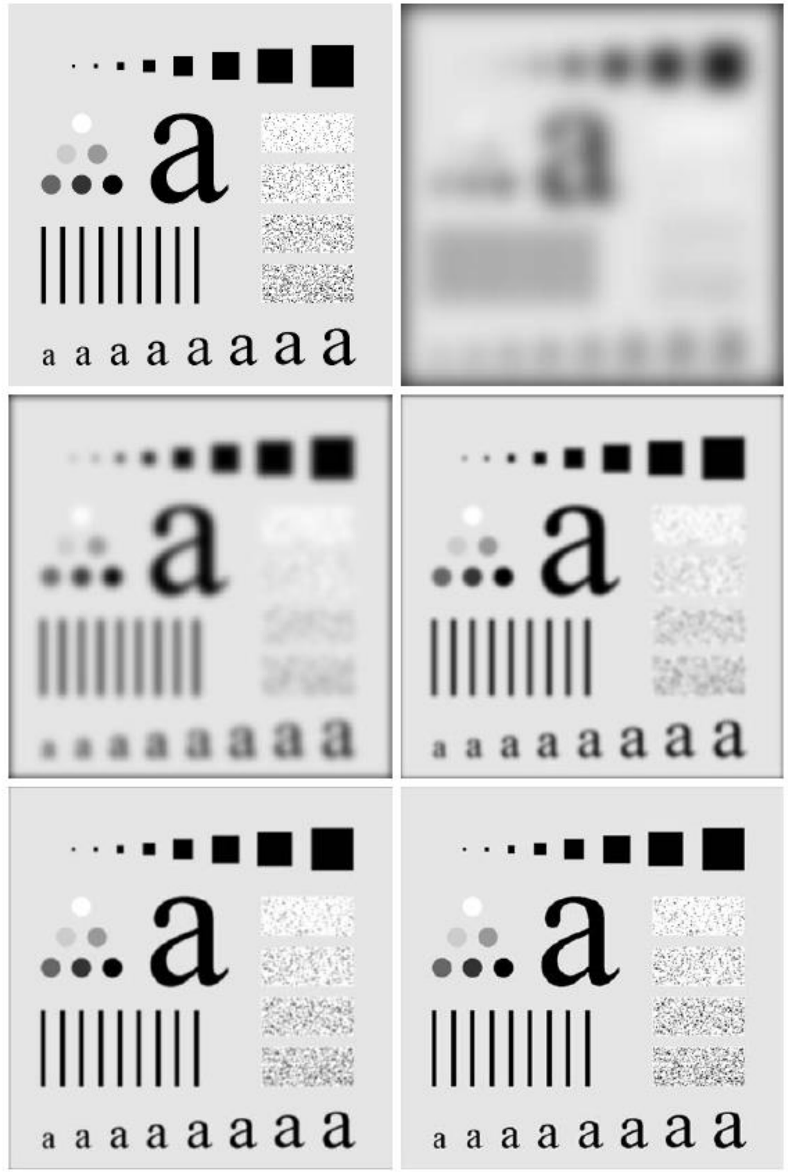


a b c

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.

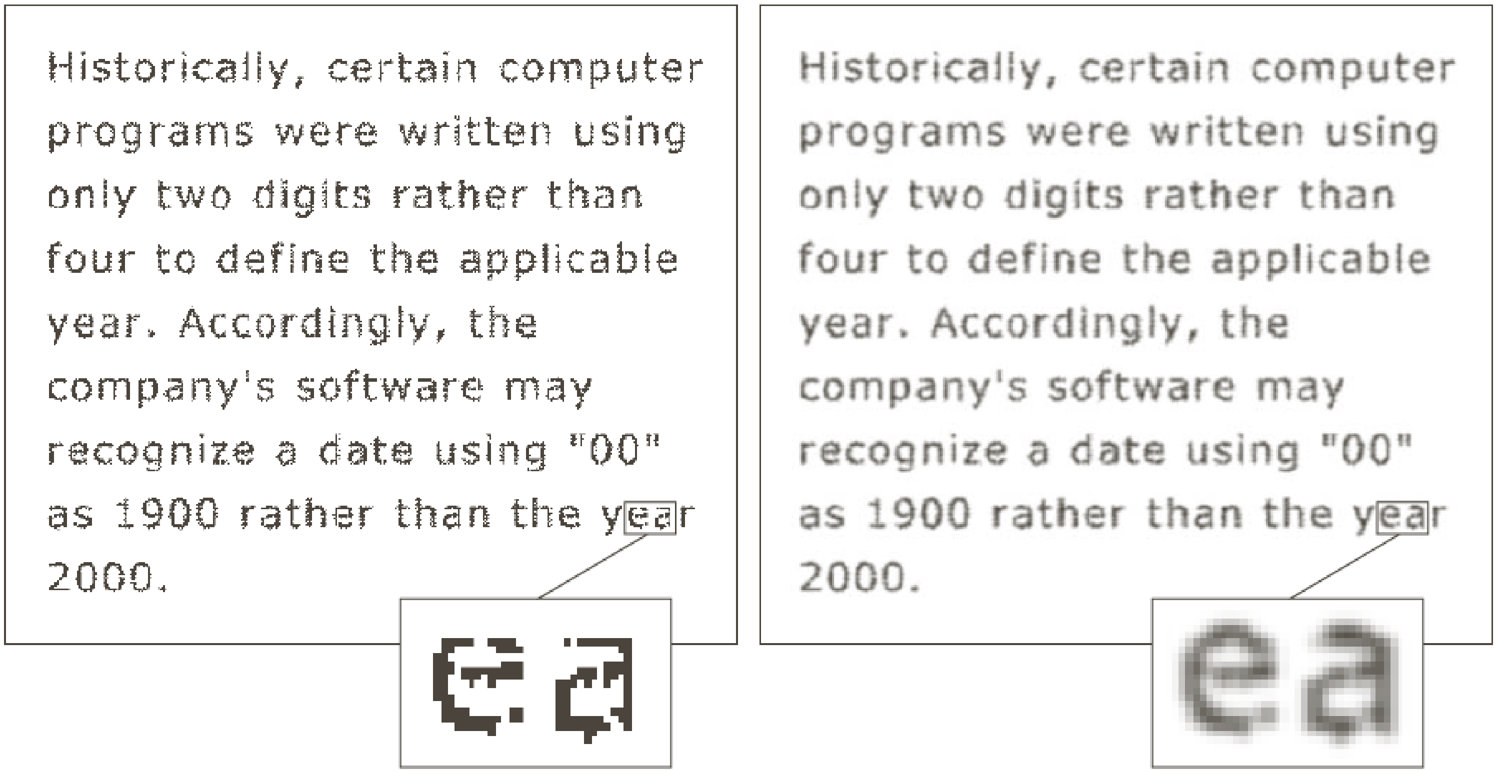
TABLE 4.4
Lowpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

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



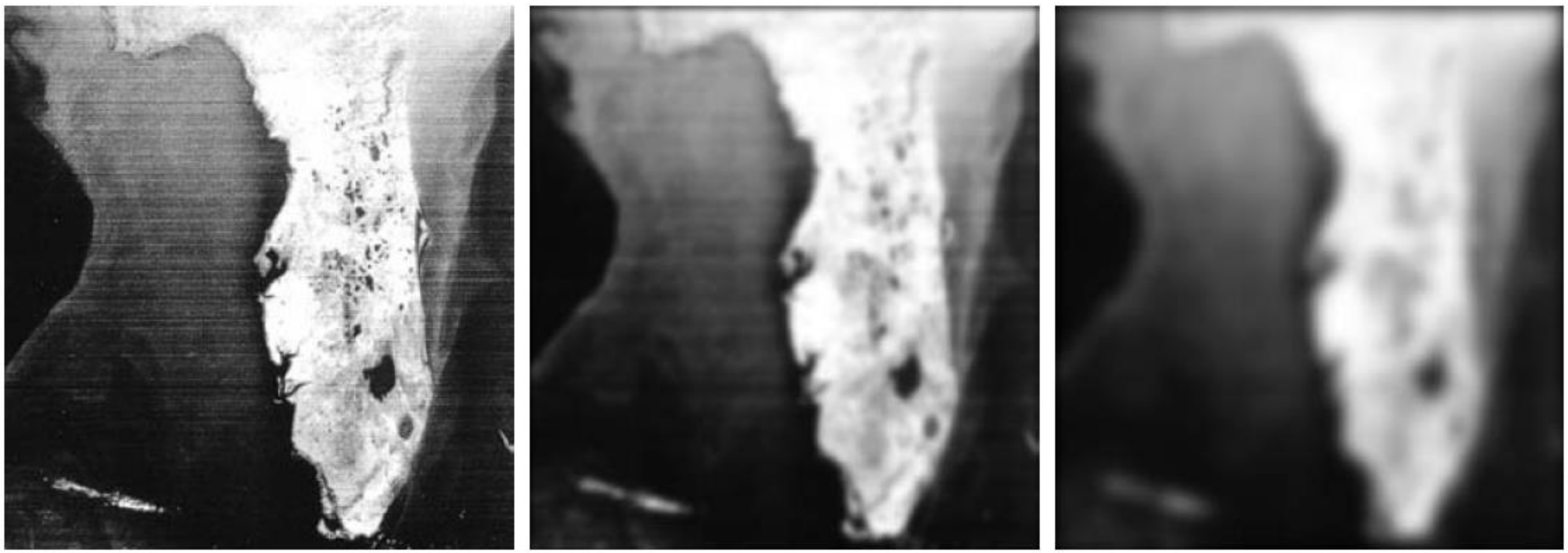
a b
c d
e f

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.


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.
ea
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.
ea

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).

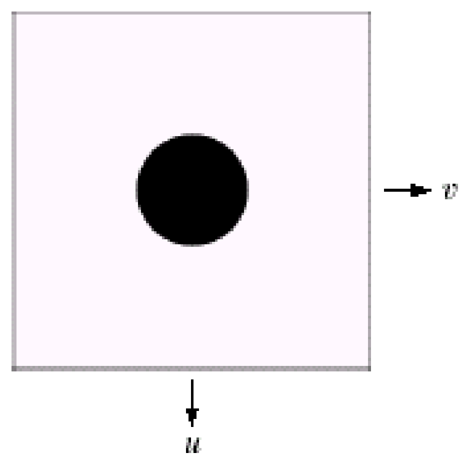


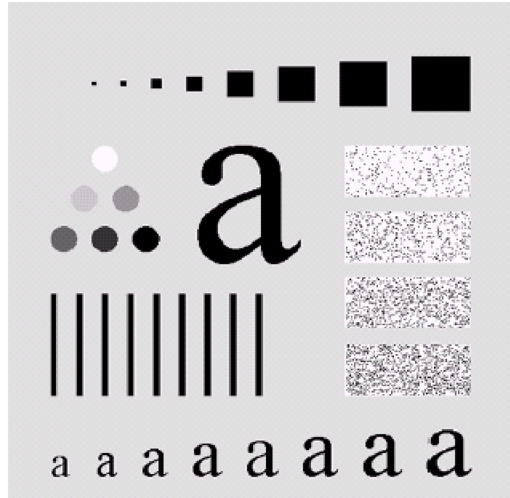
a b c

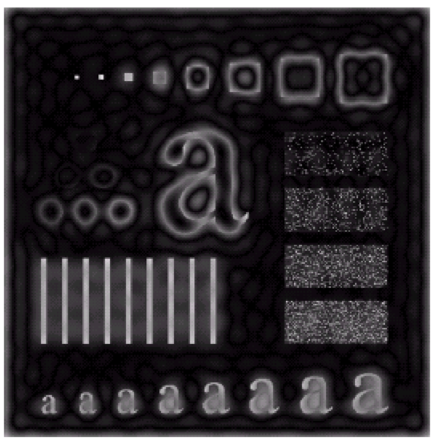
FIGURE 4.51 (a) Image showing prominent horizontal scan lines. (b) Result of filtering using a GLPF with $D_0 = 50$. (c) Result of using a GLPF with $D_0 = 20$. (Original image courtesy of NOAA.)

Ideal High Pass Filters

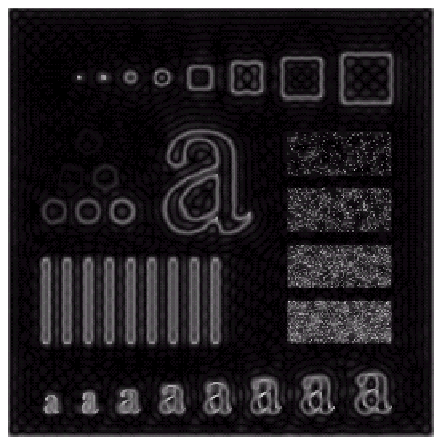
The ideal high pass filter is given as:



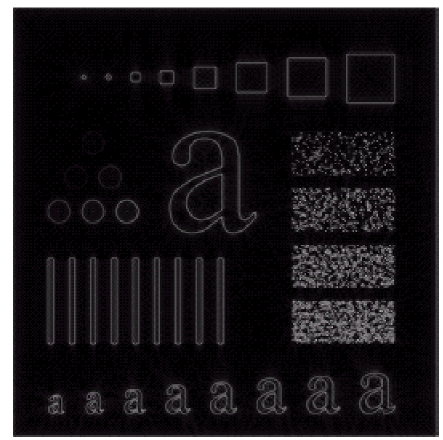




Results of ideal high pass filtering with $D_0 = 15$



Results of ideal high pass filtering with $D_0 = 30$



Results of ideal high pass filtering with $D_0 = 80$

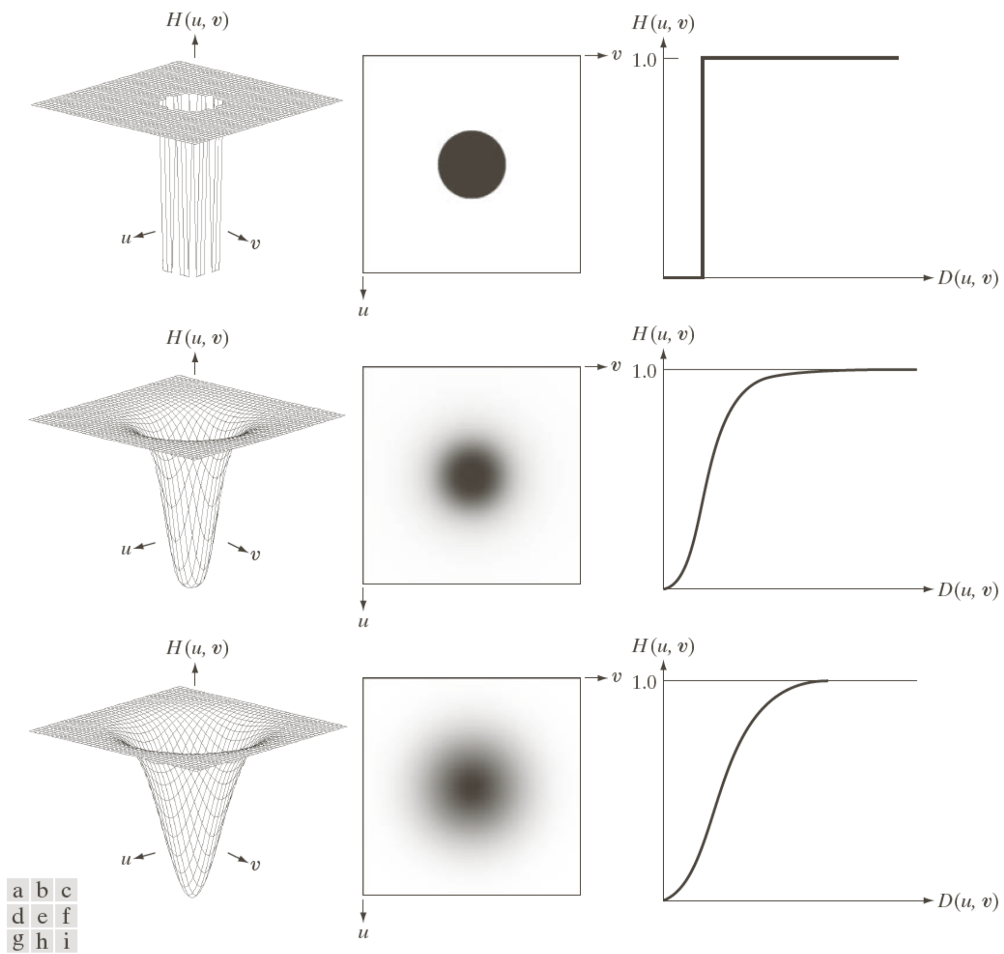


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

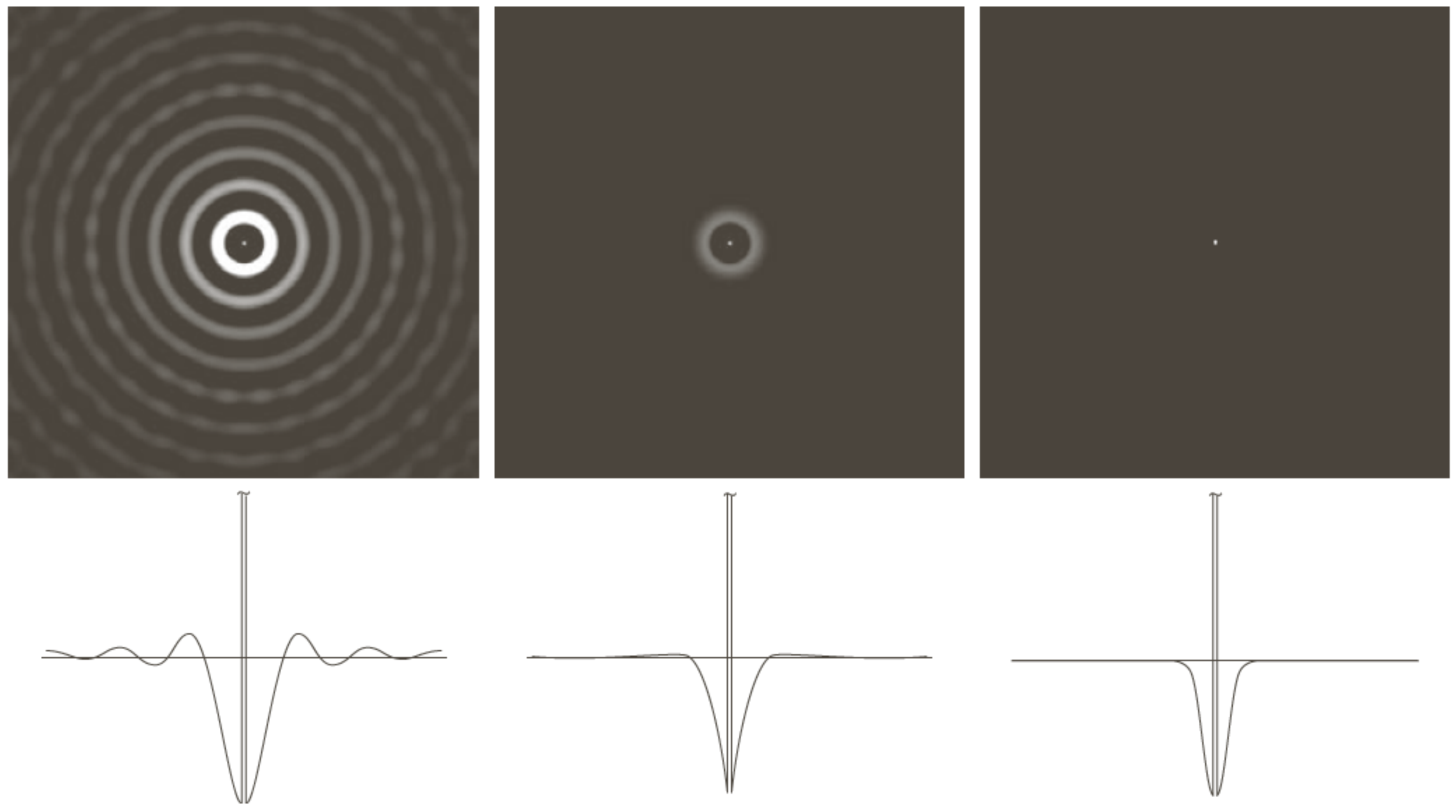


FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

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.

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.

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 4.6

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

Readings from Book (3rd Edn.)

- Sharpening Filters
- Frequency Analysis
- Filters in Frequency Domain

Acknowledgements

- Digital Image Processing", Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Peters, Richard Alan, II, Lectures on Image Processing, Vanderbilt University, Nashville, TN, April 2008
- Brian Mac Namee, Digitial Image Processing, School of Computing, Dublin Institute of Technology
- Computer Vision for Computer Graphics, Mark Borg

Material in these slides has been taken from, the following resources