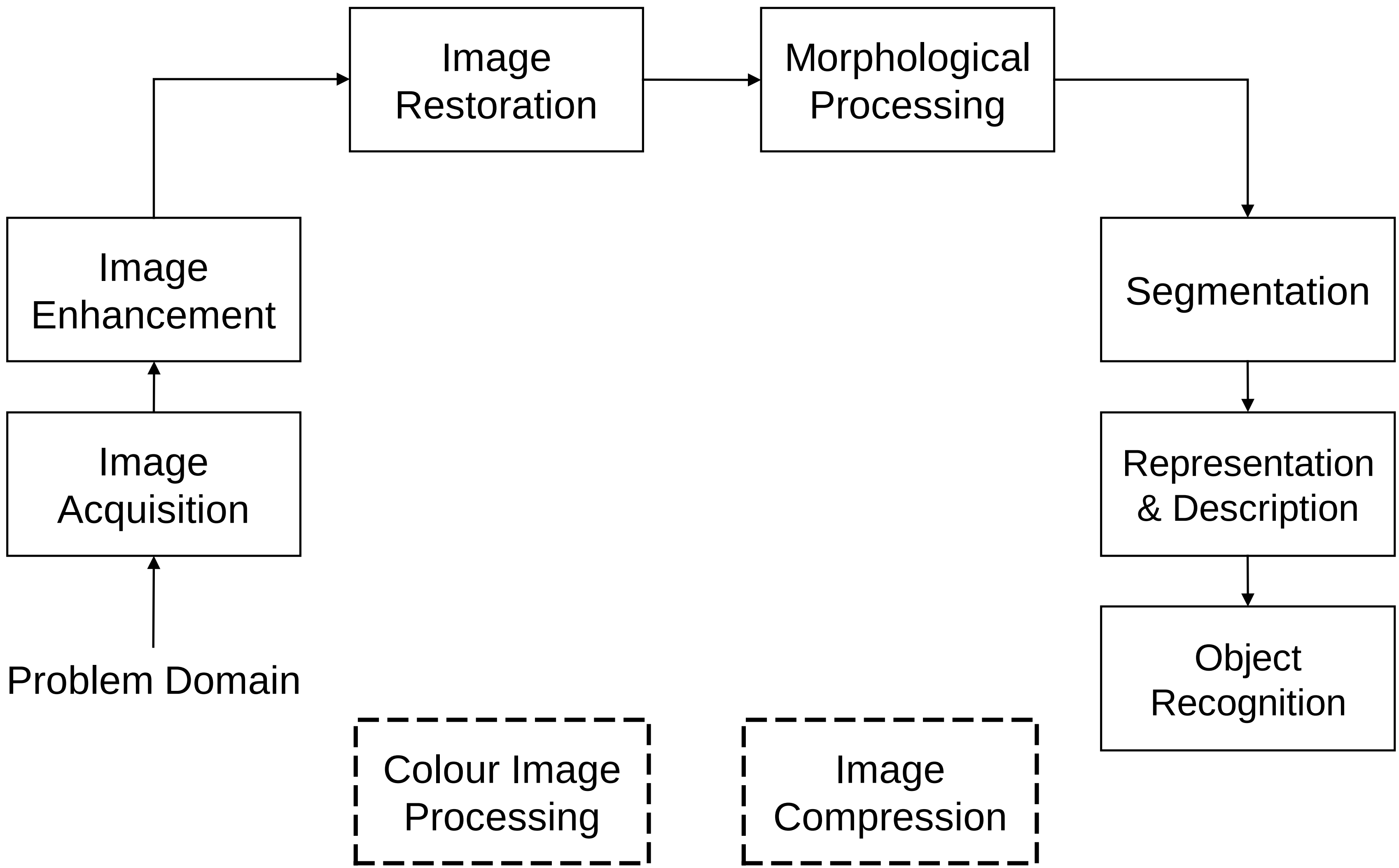
Image Enhancement

In Frequency Domain

Partially Adopted from Brian Mac Namee

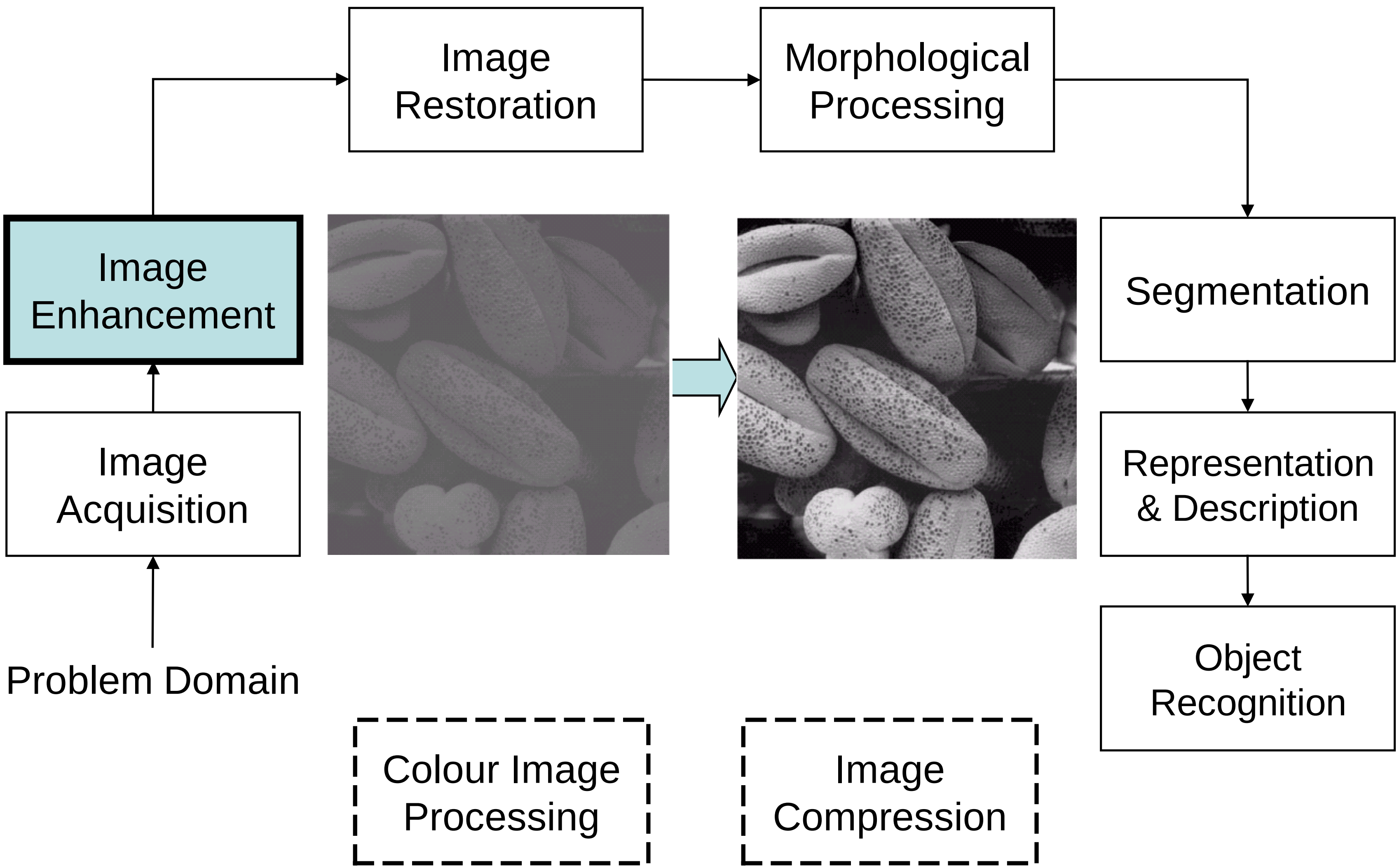
Contents

Phases of Digital Image Processing



Phases of Digital Image Processing: Image Enhancement

Images taken from Gonzalez & Woods, Digital Image Processing (2002)



Jean Baptiste Joseph Fourier

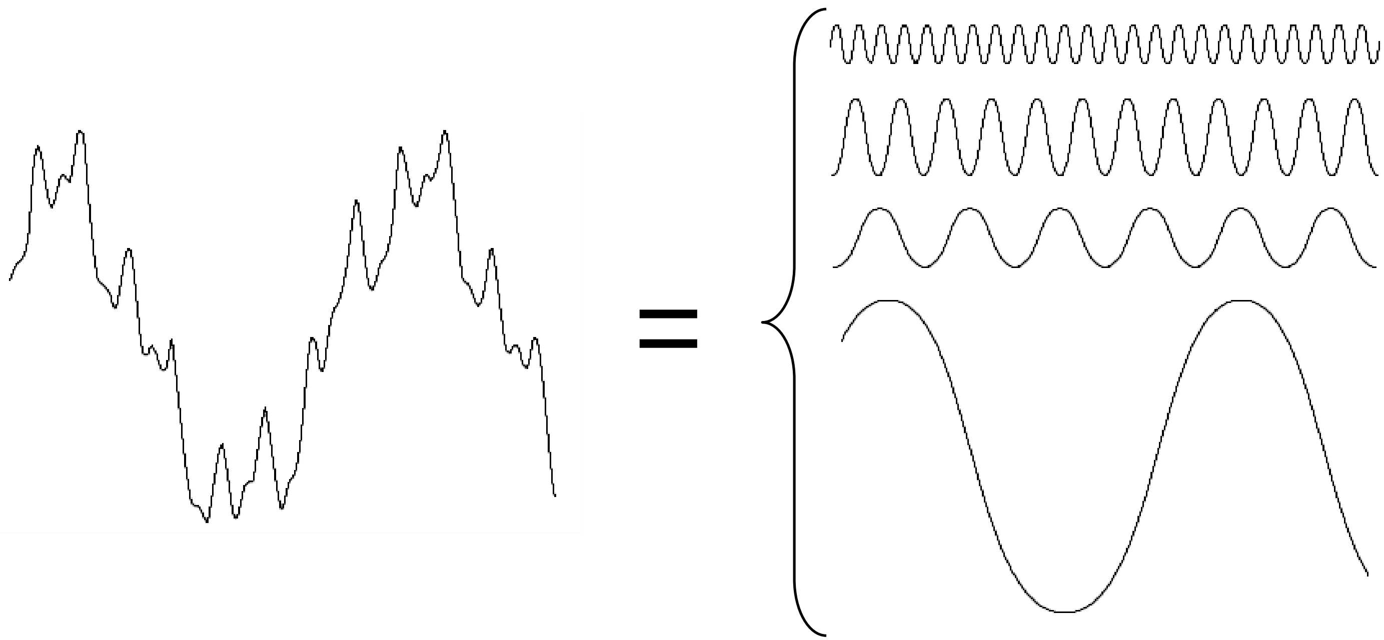
Fourier was born in Auxerre, France in 1768.

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

Nobody gave much attention when the work was first published.

One of the most important mathematical theories in Modern Engineering / Science.

Theoretical View



Any function that periodically repeats itself can be expressed as the sum of <u>sines</u> and/or <u>cosines</u> of different frequencies, each multiplied by a different coefficient – a <u>*Fourier series*</u>.

Theoretical View

Even function that are not periodic (but whose area under the curve is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighing function – a *Fourier transform*.

Its utility is even greater than the Fourier series in most practical problems.

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

The One-Dimensional Fourier Transform and its Inverse

The Fourier transform, $F(u)$, of a single variable, continuous function, $f(x)$ is defined by:

$$F(u) = \int_{-\propto}^{\propto} f(x) e^{-j2\pi ux}\, dx, \quad where\ j = \sqrt{-1}.$$

Conversely, given $F(u)$, we can obtain $f(x)$ by means of the *inverse* Fourier transform

$$f(x) = \int_{-\propto}^{\propto} F(u) e^{j2\pi ux}\, du, \qquad where\ j = \sqrt{-1}.$$

The two-Dimensional Fourier Transform and its Inverse

$$F(u,v) = \int_{-\propto}^{\propto}\int_{-\propto}^{\propto} f(x,y)e^{-j2\pi(ux+vy)}\,dxdy$$

$$f(x,y) = \int_{-\propto}^{\propto}\int_{-\propto}^{\propto} F(u,v)e^{j2\pi(ux+vy)}\,dudv$$

The Discrete Fourier Transform (DFT)

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

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

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

The Discrete Fourier Transform (DFT)

- Fourier Spectrum:

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

- Phase Angle / Phase Spectrum:

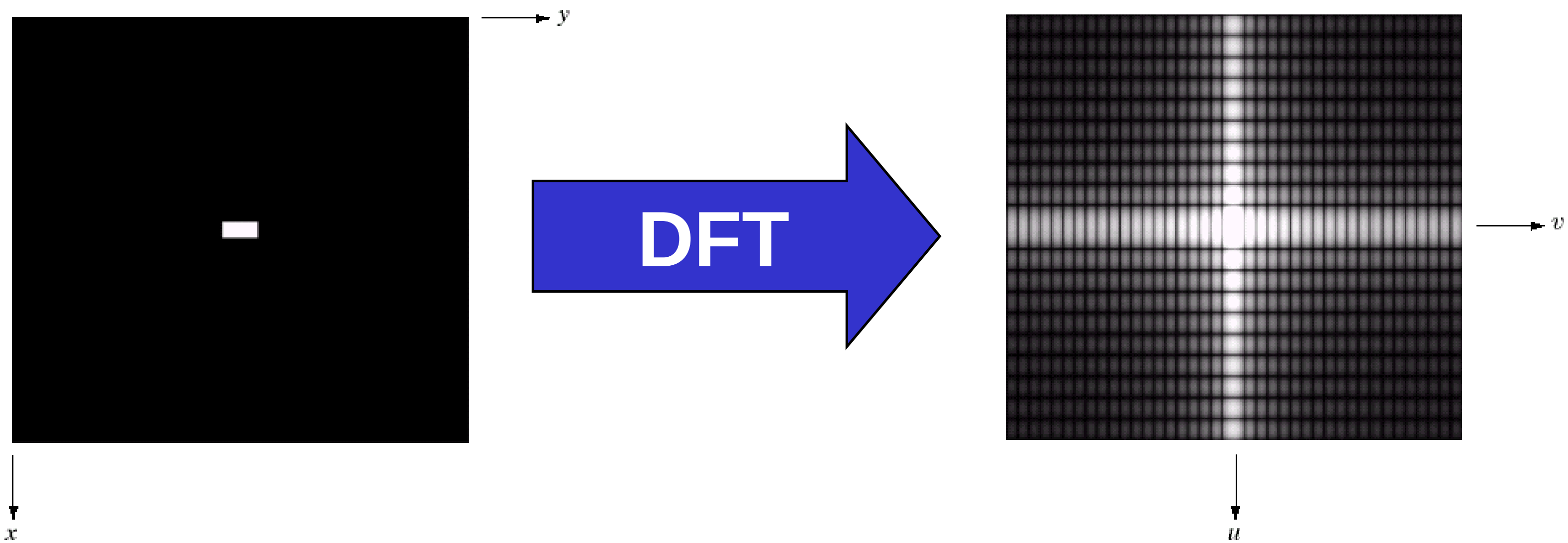
$$\emptyset(u,v) = \tan^{-1}[\frac{I(u,v)}{R(u,v)}]$$

- Power Spectrum:

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

DFT of an Image

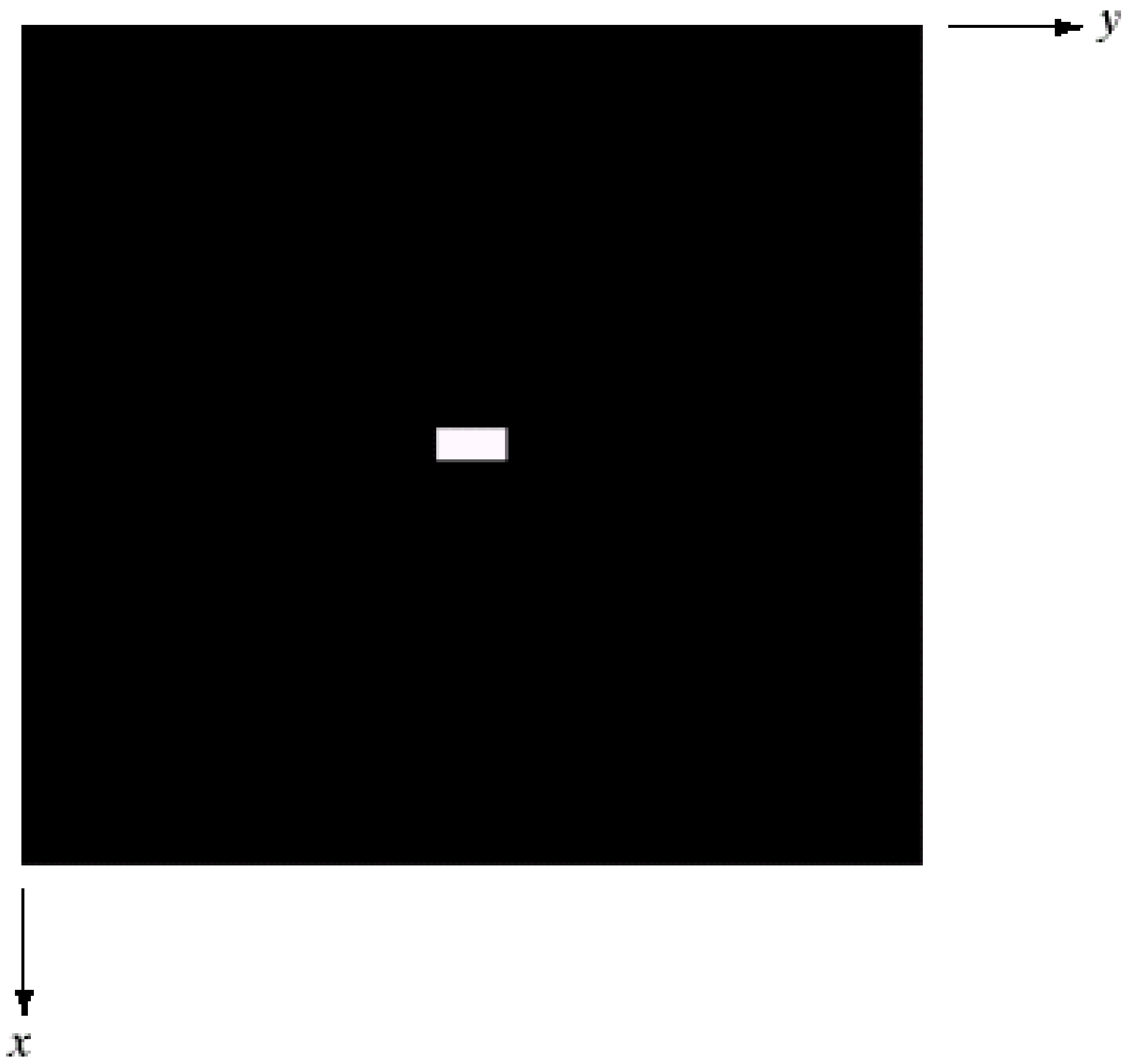
The DFT of a two dimensional image can be visualised by showing the spectrum of the image component frequencies.



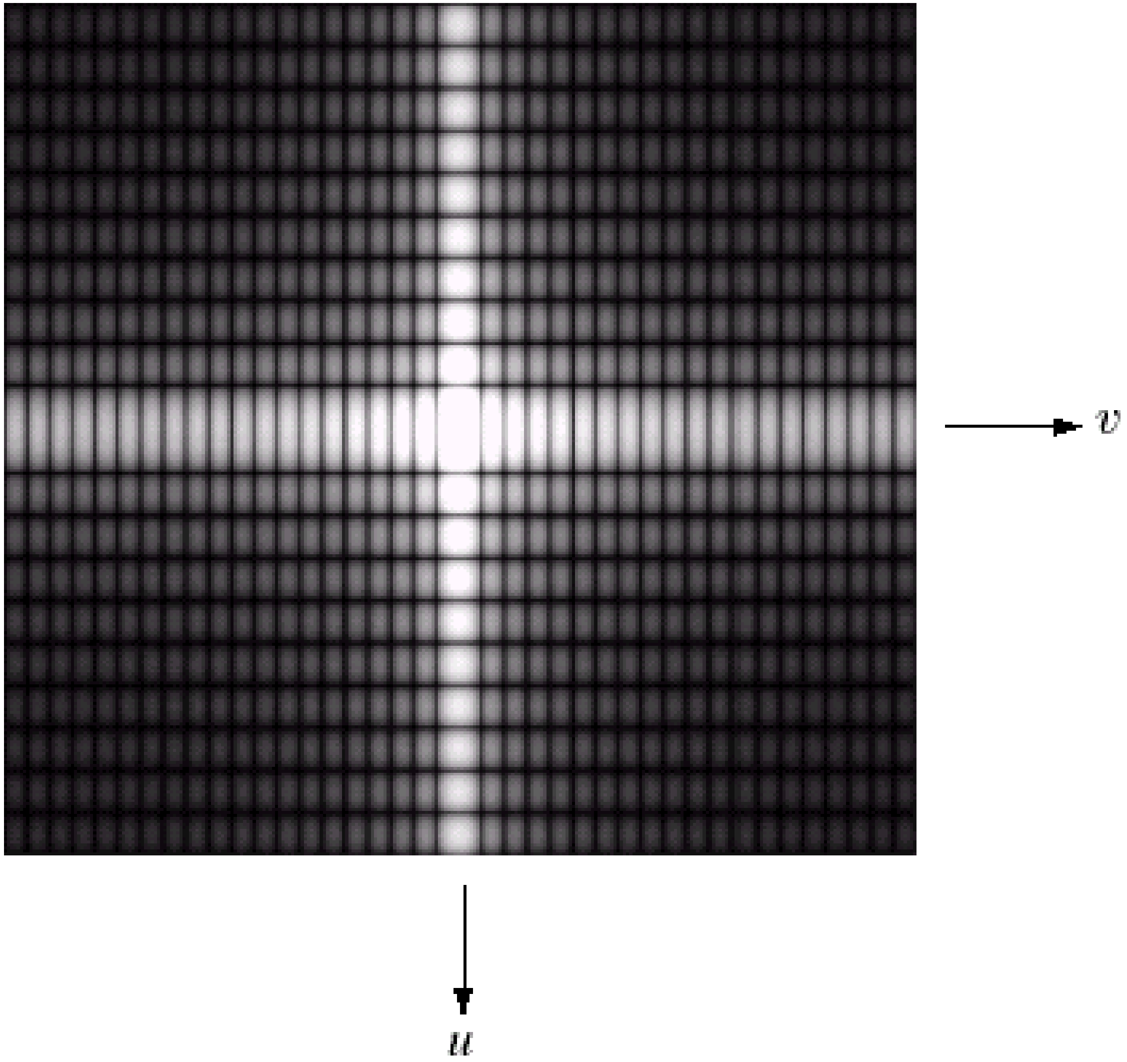
It is common practice to multiply the input image function by $(-1)^{x+y}$ prior to computing Fourier transform.

This operation shifts the origin of FT of $f(x,y)(-1)^{x+y}$ (i.e. $F(0,0)$) to [M/2, N/2].

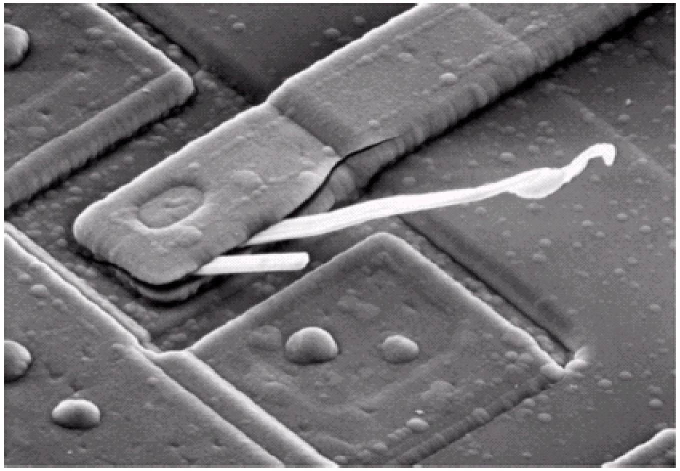
DFT of an Images



DFT of an Image

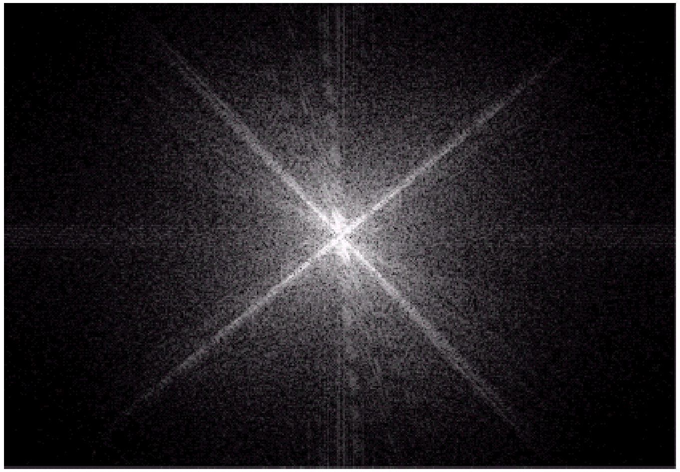


DFT of an Image



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

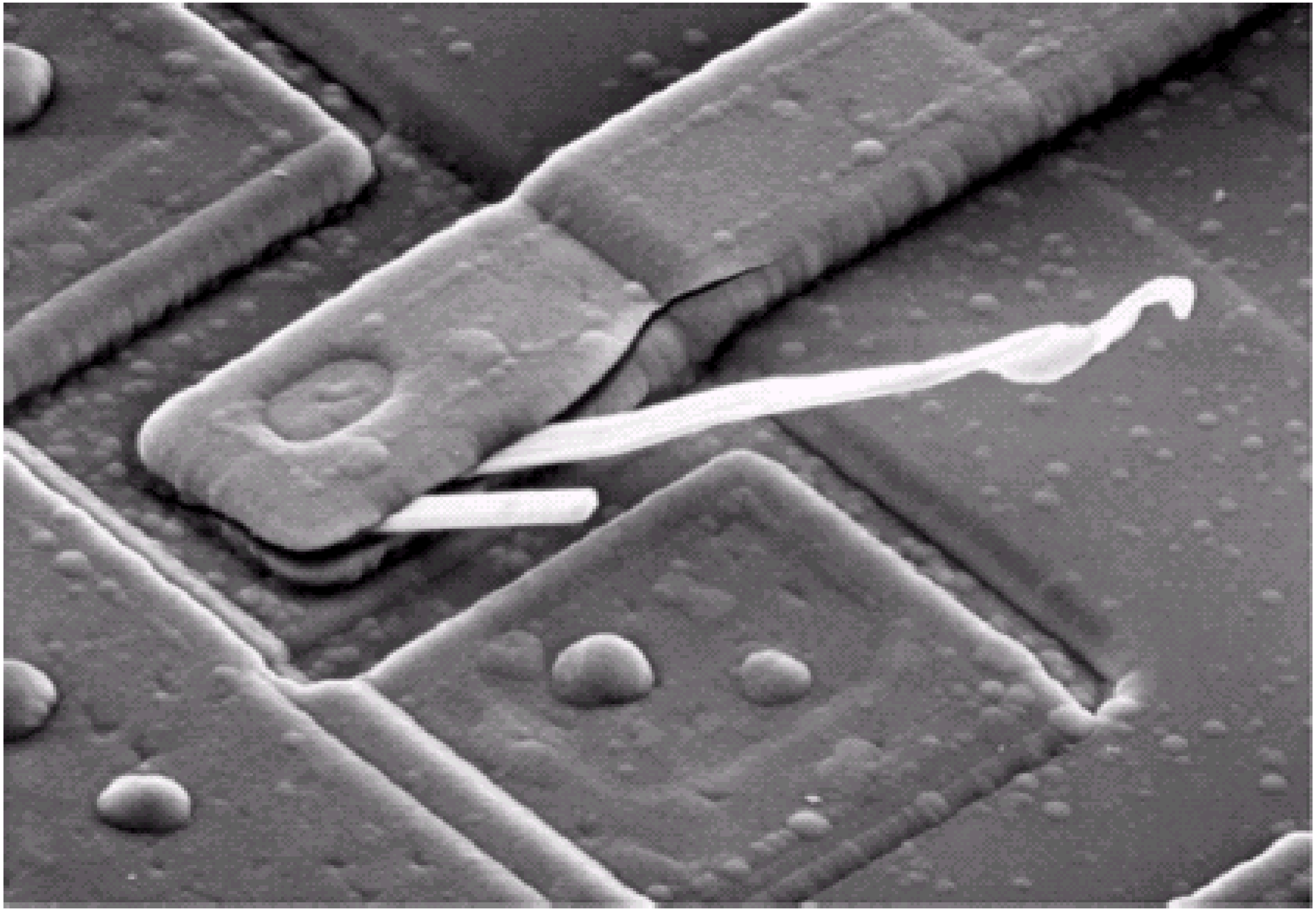
DFT



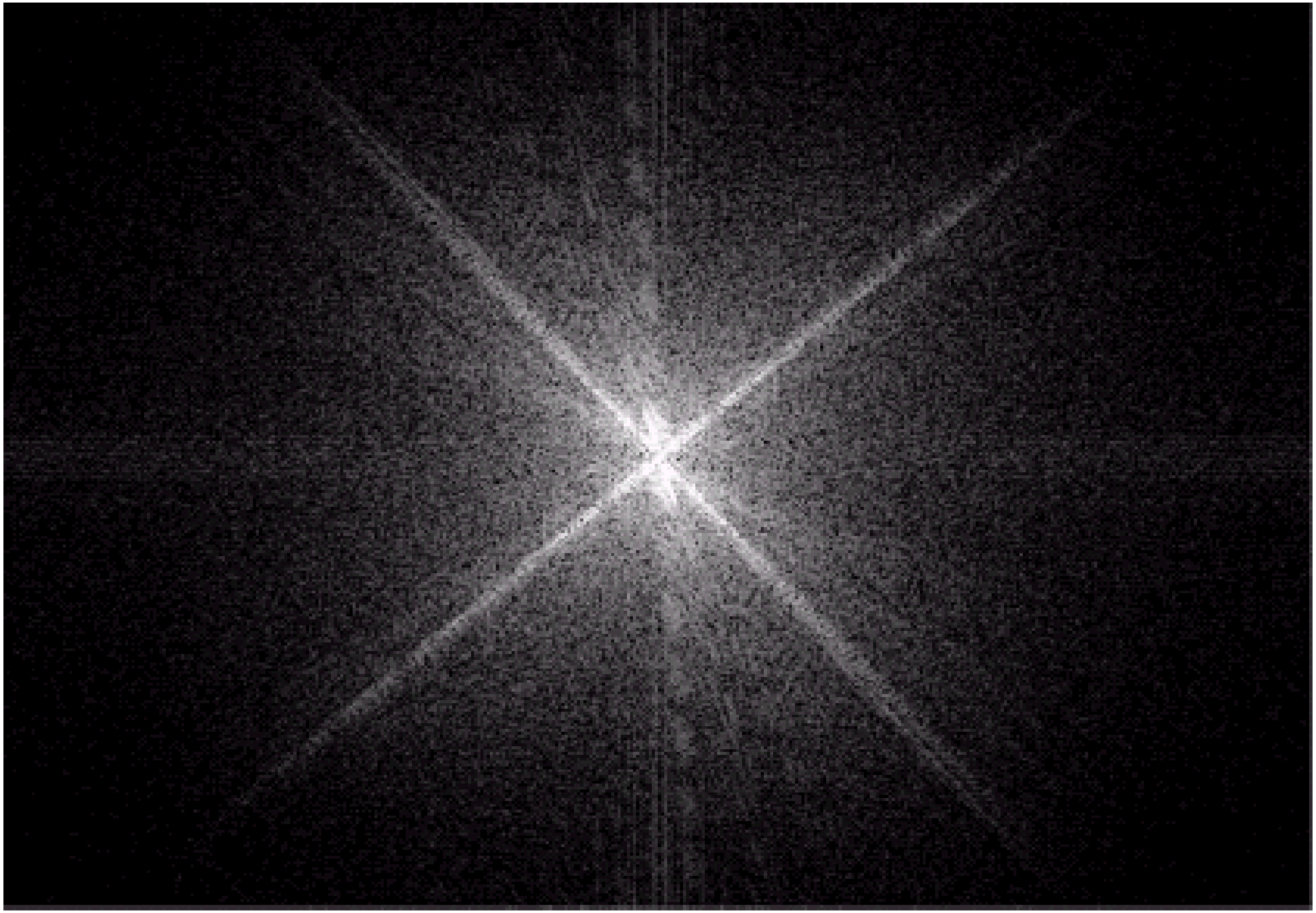
Fourier spectrum of the image

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

DFT of an Image



DFT of an Image



The Inverse DFT

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

The inverse DFT is given by:

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

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

Basics of Filtering in Frequency Domain

To filter an image in the frequency domain:

1. Multiply the input image by $(-1)^{x+y}$ to center the transform, as indicated in

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

It states that the origin of the Fourier transform of $f(x,y)(-1)^{x+y}$ [i.e., $F(0,0)$] is located at $u = M/2$ and $v = N/2$.

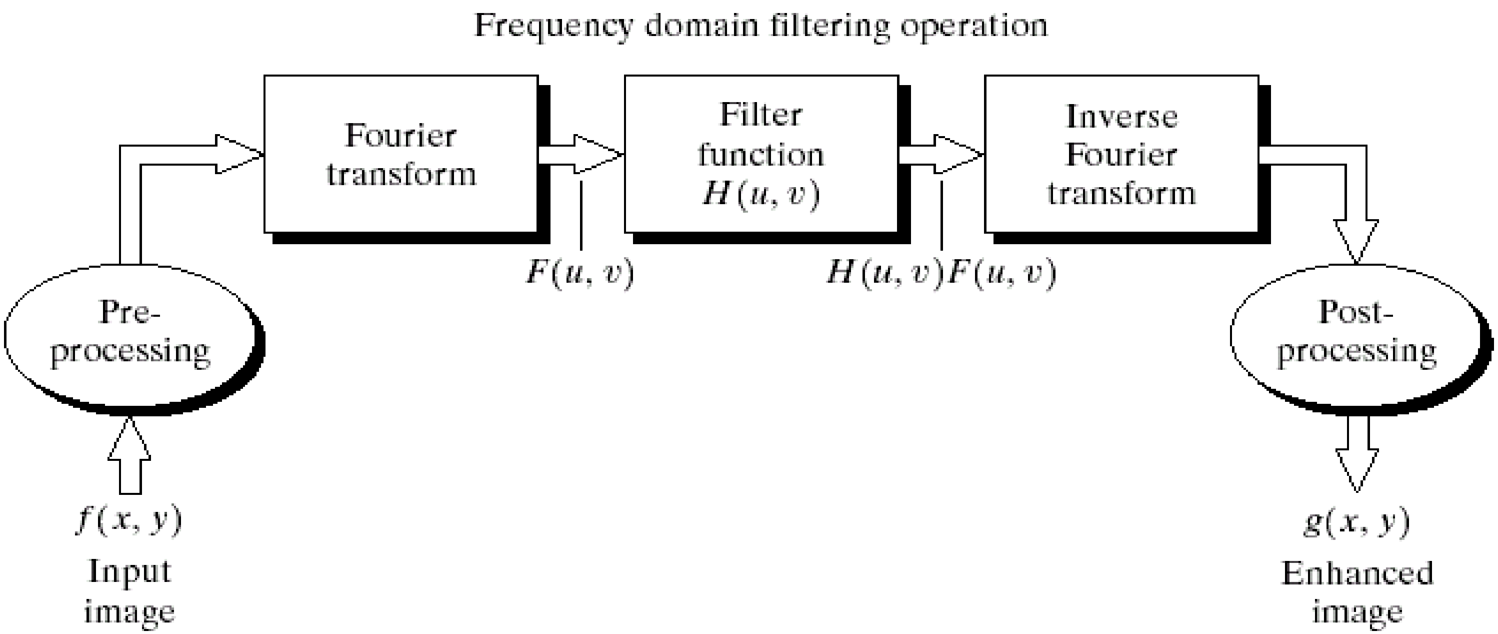
2. Compute $F(u,v)$, the DFT of the image from (1).

3. Multiply $F(u,v)$ by a filter function $H(u,v)$.

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Basics of Filtering in Frequency Domain

4. Compute the inverse DFT of the result in (3).
5. Obtain the real part of the result in (4).
6. Multiply the result in (5) by $(-1)^{x+y}$.



Basics of Filtering in Frequency Domain

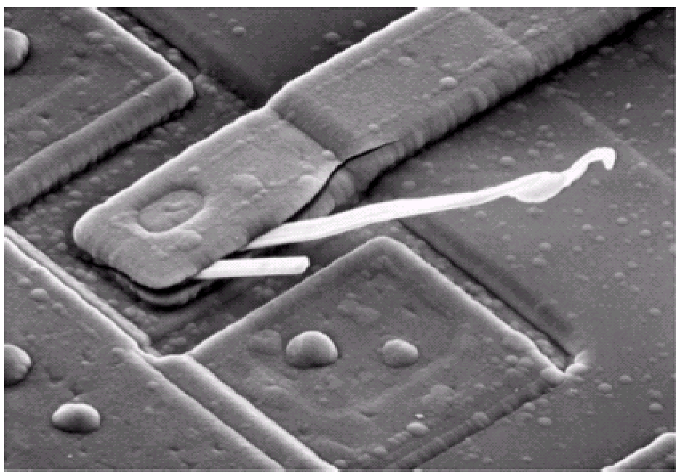
- Each component of H multiplies both the real and imaginary parts of the corresponding components in F.

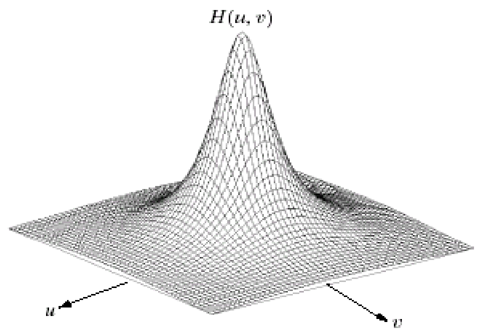
- This type of filters are called ***zero-phase-shift*** filters.

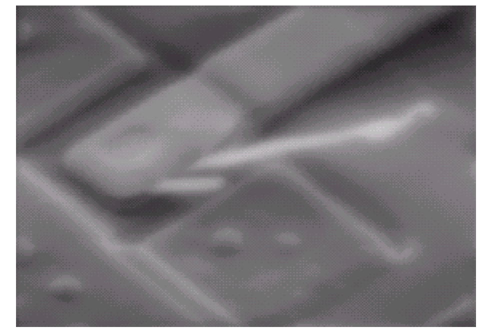
Some Basic Frequency Domain Filters

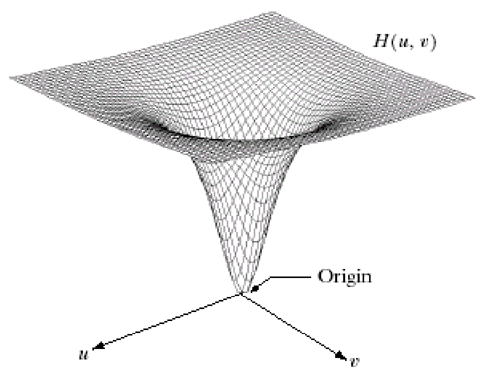
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

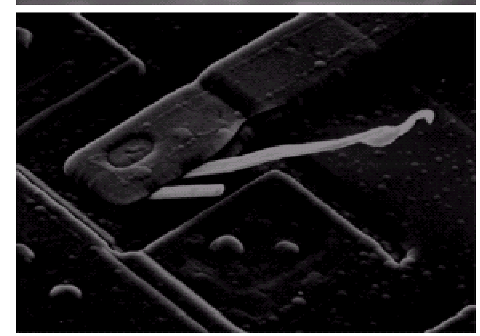
Low Pass Filter





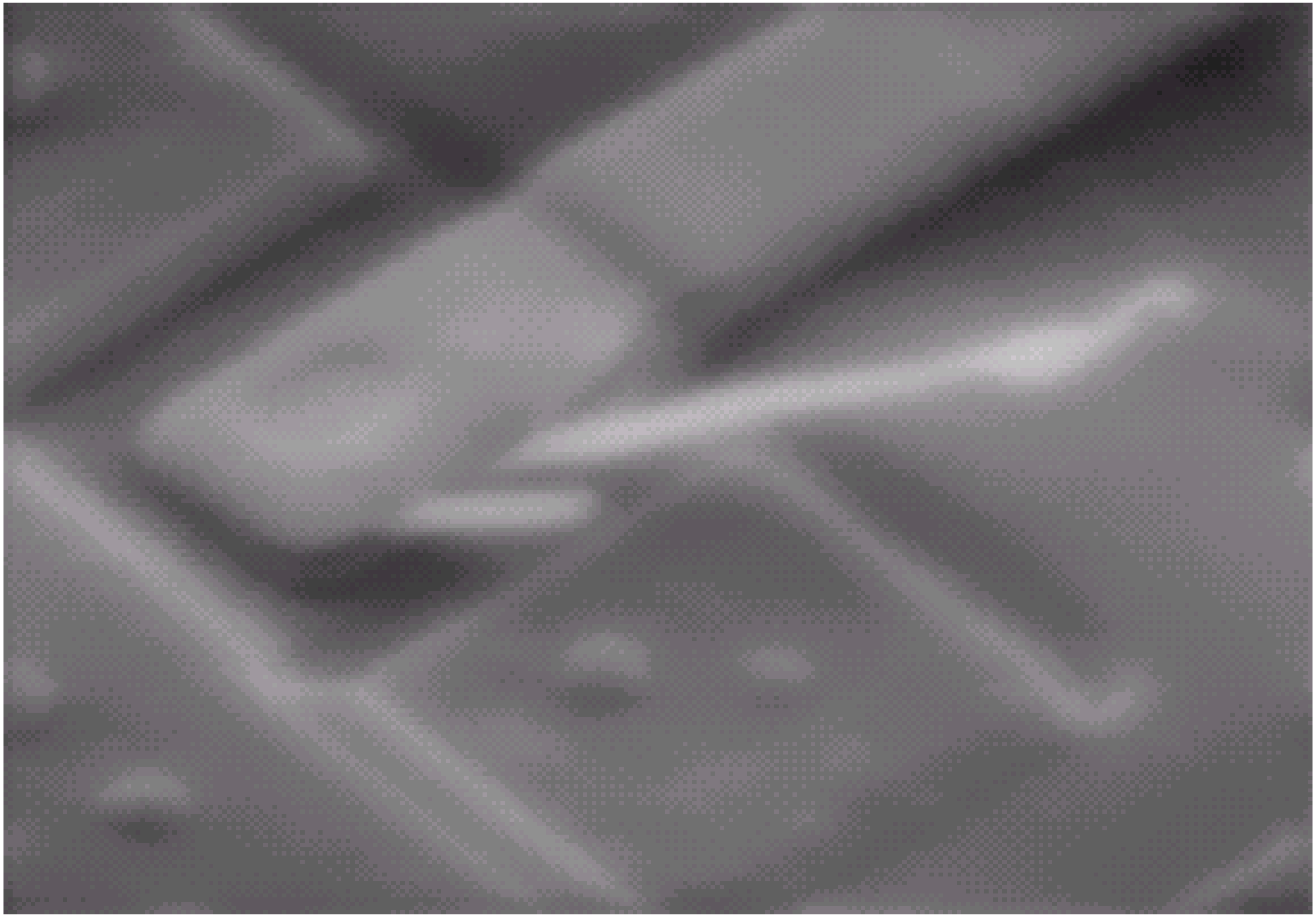






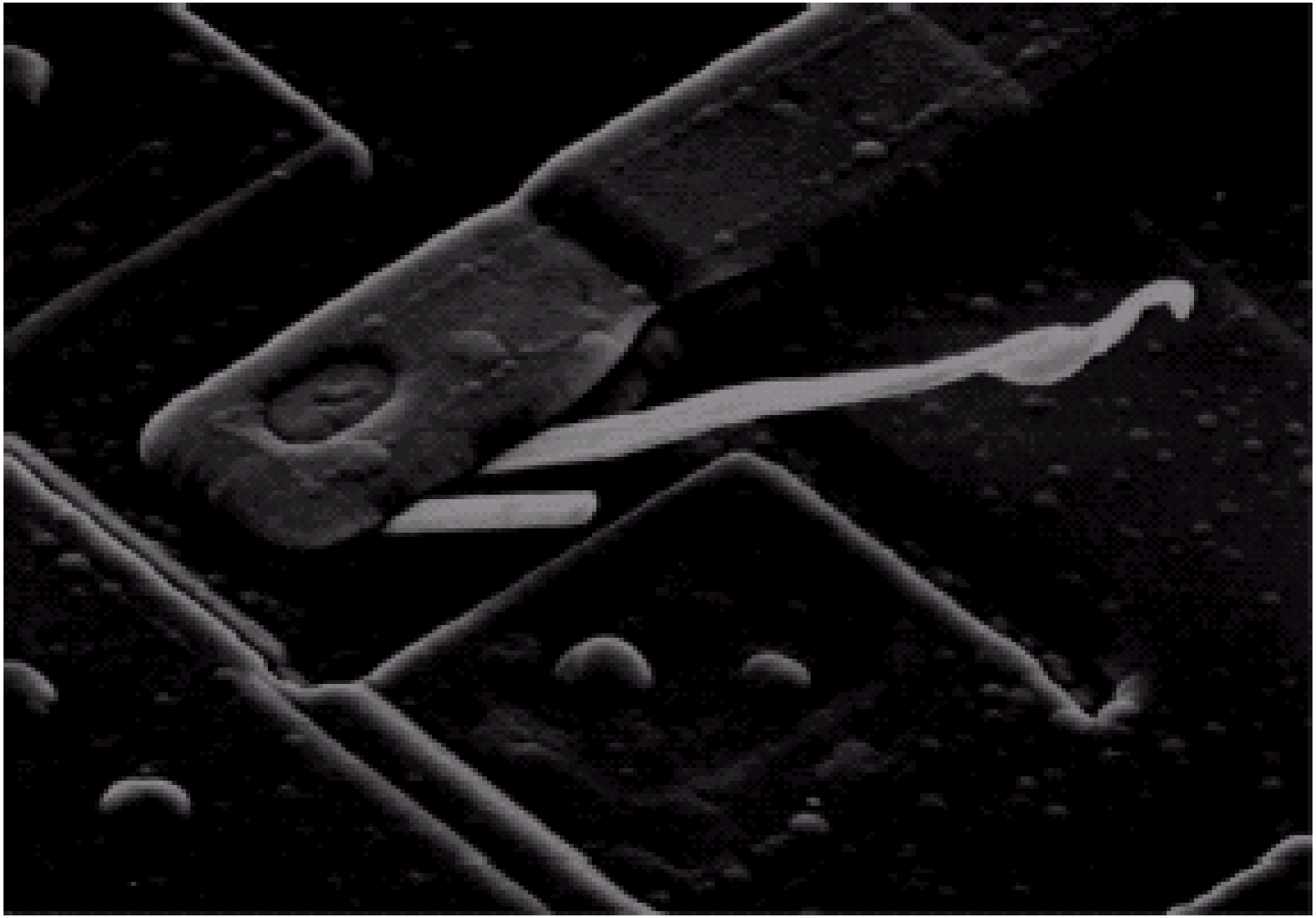
High Pass Filter

Some Basic Frequency Domain Filters



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Some Basic Frequency Domain Filters



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Filters in Frequency Domain

- ✓ Image Smoothing Filters
  - Ideal Low Pass Filter
  - Butterworth Low Pass Filter
  - Gaussian Low Pass Filter

- ✓ Image Sharpening Filters
  - Ideal High Pass Filter
  - Butterworth High Pass Filter
  - Gaussian High Pass Filter

Smoothing Frequency Domain Filters

Smoothing is achieved in the frequency domain by dropping out the high frequency components.

The basic model for filtering is:

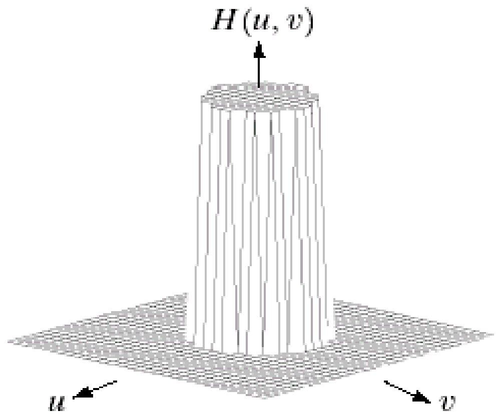
$$G(u,v) = H(u,v)F(u,v)$$

where $F(u,v)$ is the Fourier transform of the image being filtered and $H(u,v)$ is the filter transform function.

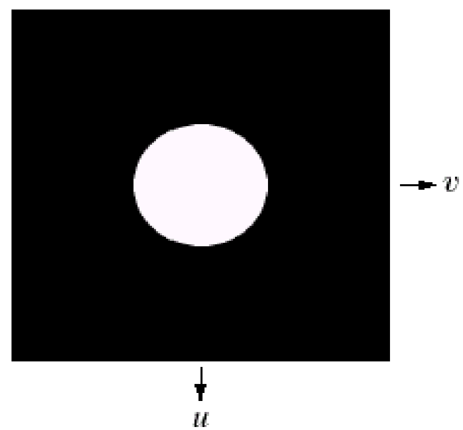
Low pass filters – <u>only pass the low frequencies</u>, drop the high ones.

Ideal Low Pass Filter

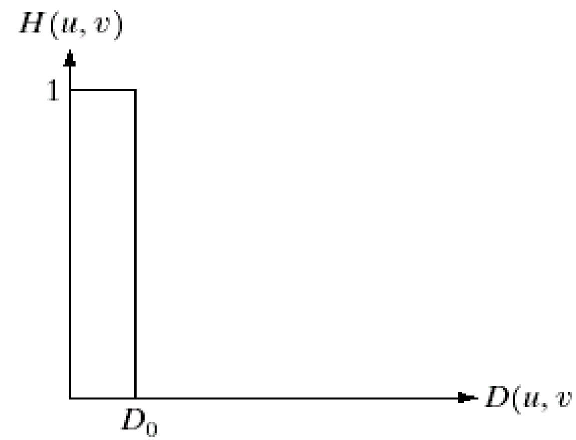
Simply cut off all high frequency components that are a specified distance D_0 from the origin of the transform.



Perspective plot
of the transfer function



Displayed as an
image



Radial cross section

Changing the distance changes the behaviour of the filter.

Ideal Low Pass Filter

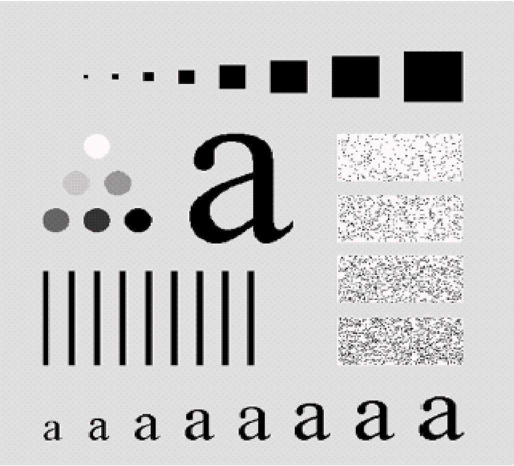
The transfer function for the ideal low pass filter can be given as:

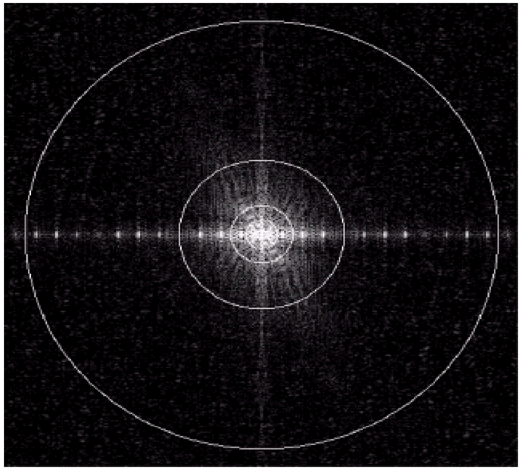
$$H(u,v) = \begin{cases} 1 \ \text{if} \ D(u,v) \le D_0 \\ 0 \ \text{if} \ D(u,v) > D_0 \end{cases}$$

where $D(u,v)$ is given as:

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

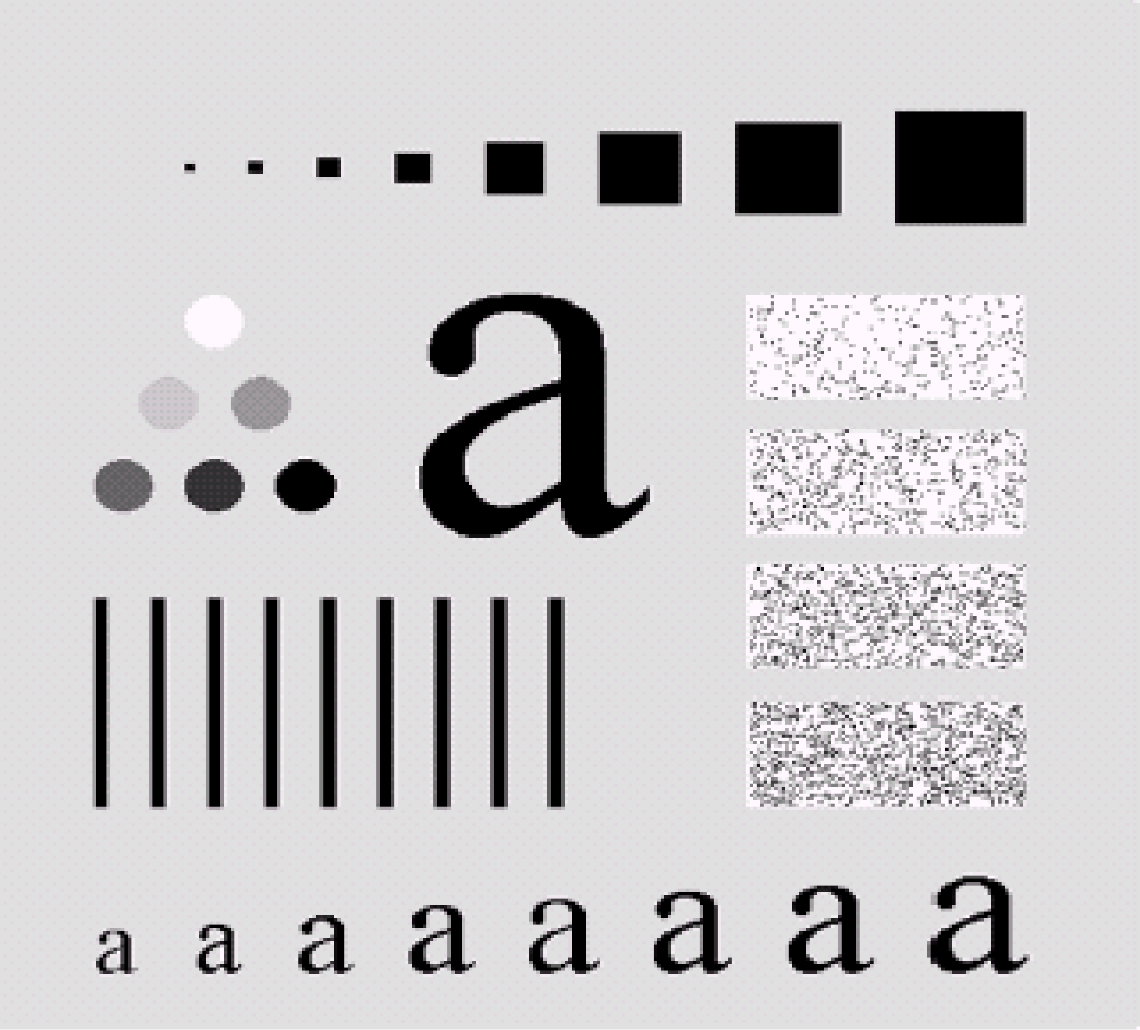
Ideal Low Pass Filter



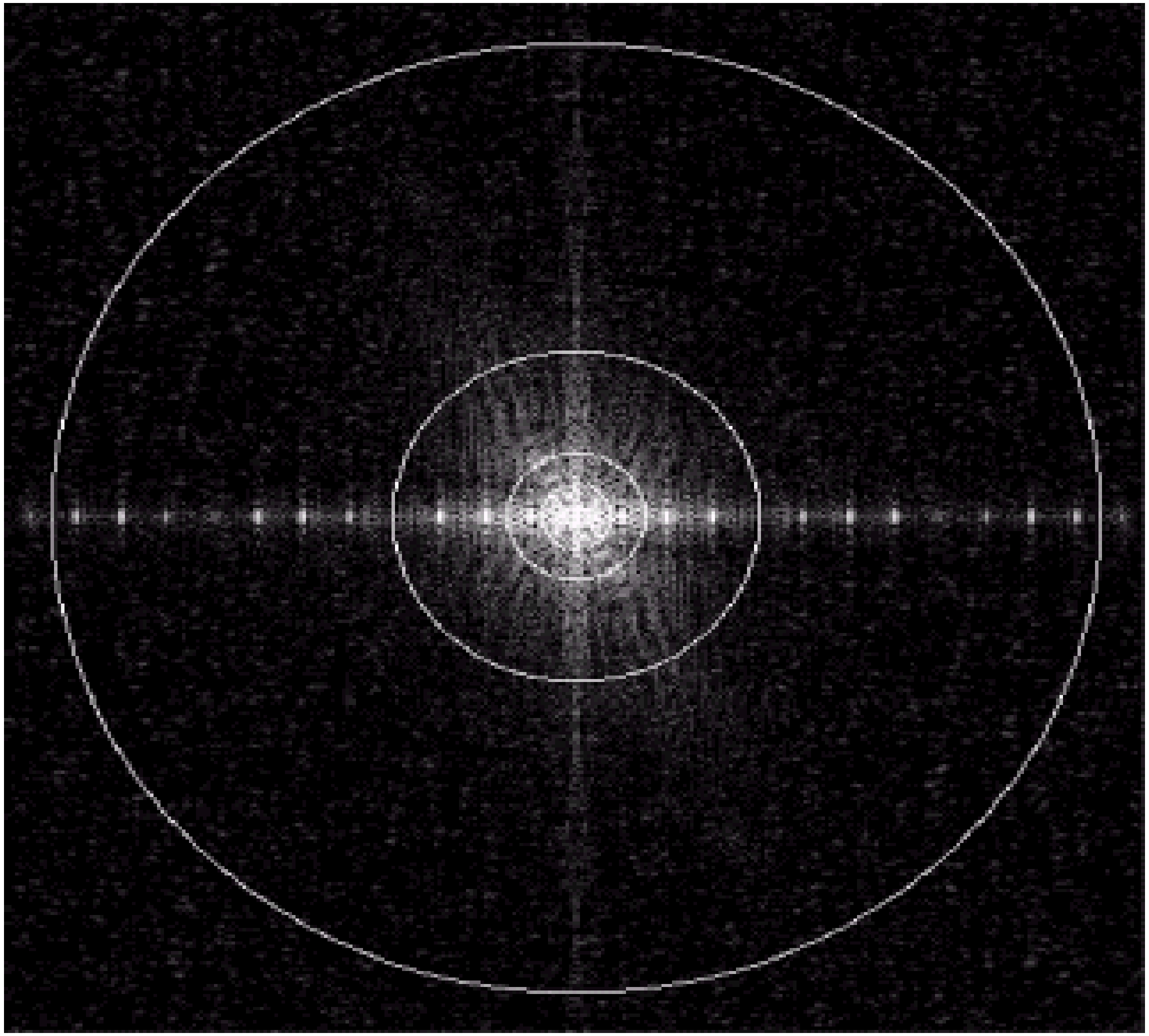


An image, it's Fourier spectrum and a series of ideal low pass filters of radius 5, 15, 30, 80, and 230 superimposed on top of it.

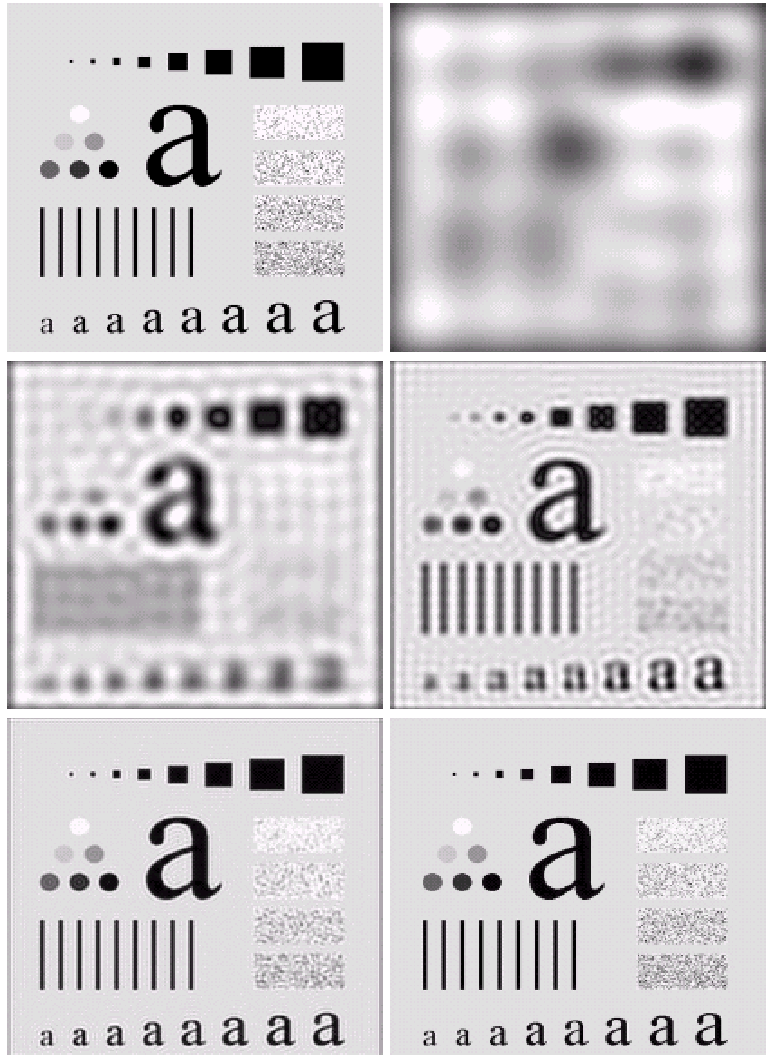
Ideal Low Pass Filter



Ideal Low Pass Filter



Ideal Low Pass Filter



Original image

Result of filtering with ideal low pass filter of radius 5

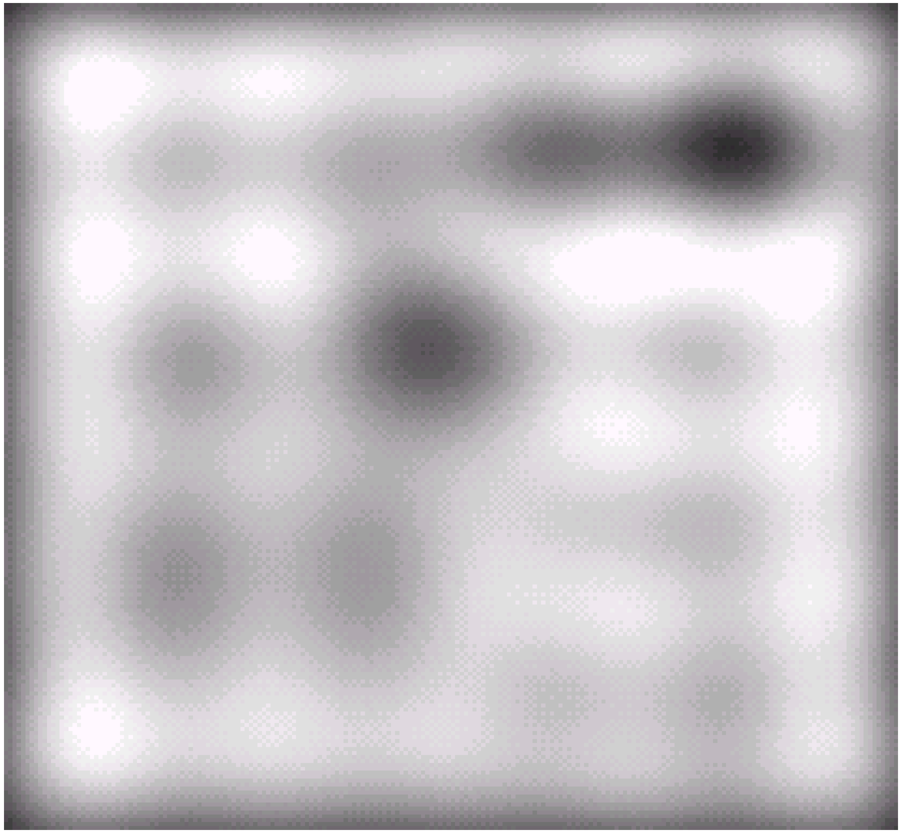
Result of filtering with ideal low pass filter of radius 15

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

Ideal Low Pass Filter

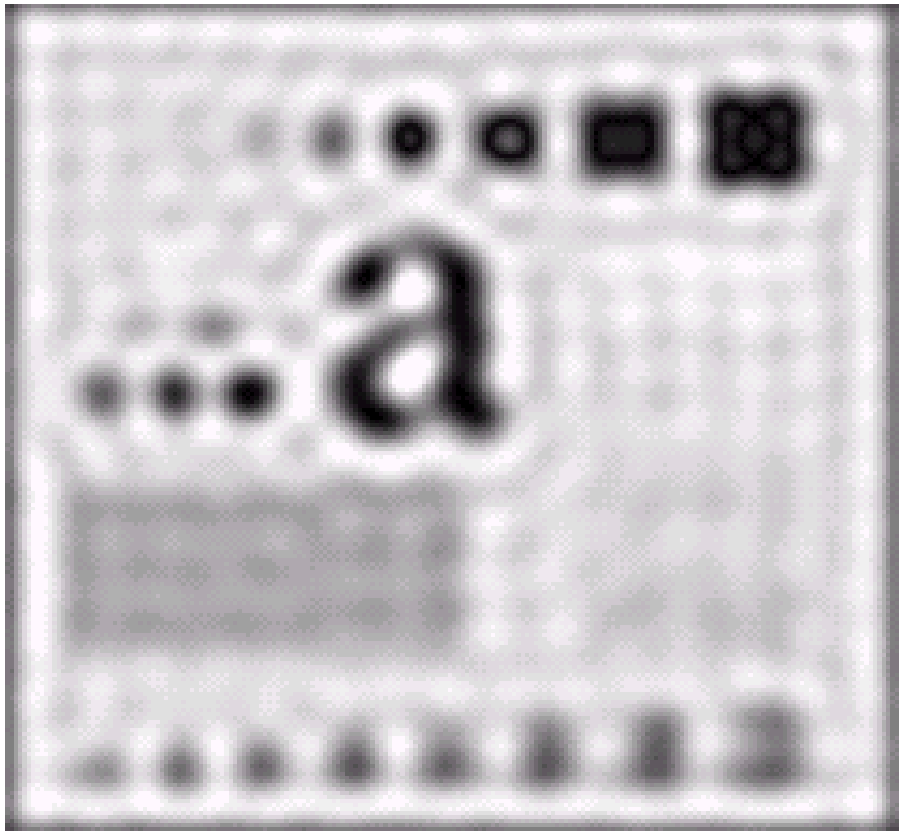


Result of filtering with ideal low pass filter of radius 5

Suffered by Ringing artifact.

Ideal Low Pass Filter

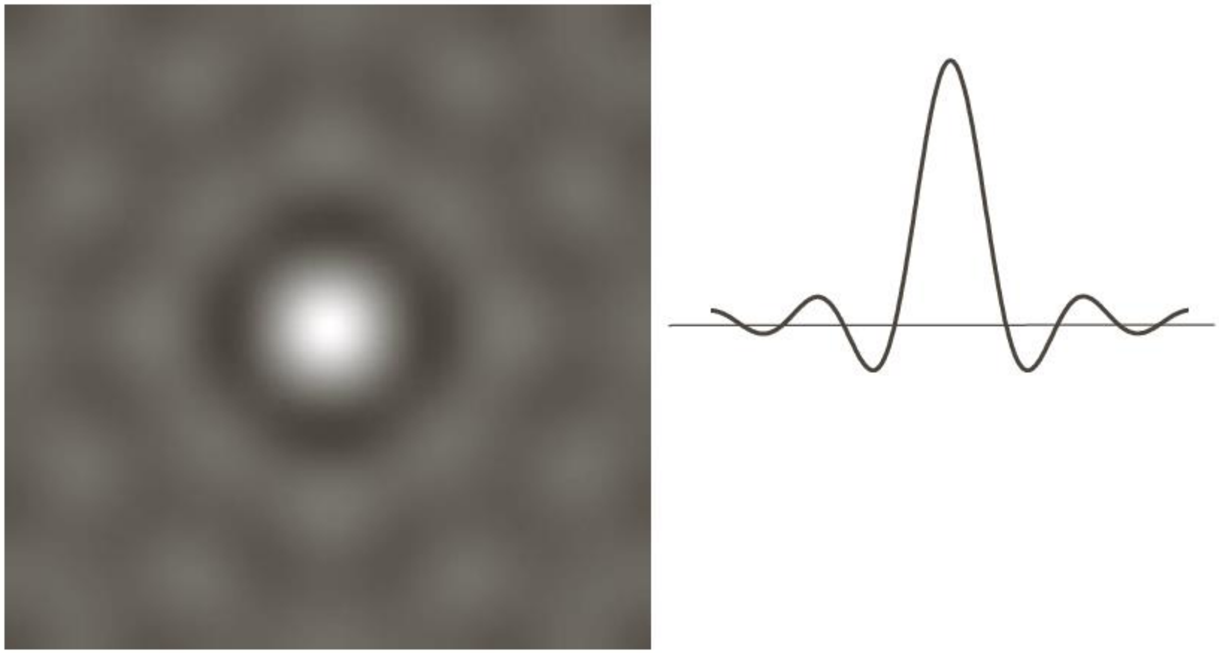
Images taken from Gonzalez & Woods, Digital Image Processing (2002)



Result of filtering with ideal low pass filter of radius 15

Suffered by Ringing artifact.

Ideal Low Pass Filter



a b

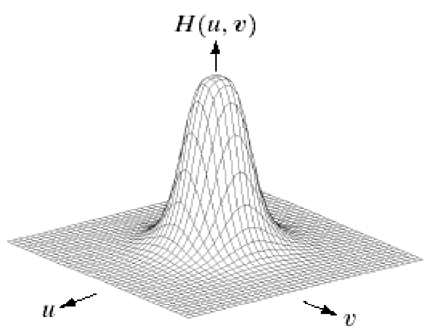
FIGURE 4.43 (a) Representation in the spatial domain of an ILPF of radius 5 and size 1000×1000. (b) Intensity profile of a horizontal line passing through the center of the image.

Butterworth Low Pass Filter

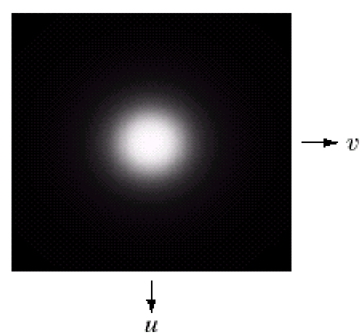
The transfer function of a Butterworth lowpass filter of order n with cutoff frequency at distance D_0 from the origin is defined as:

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

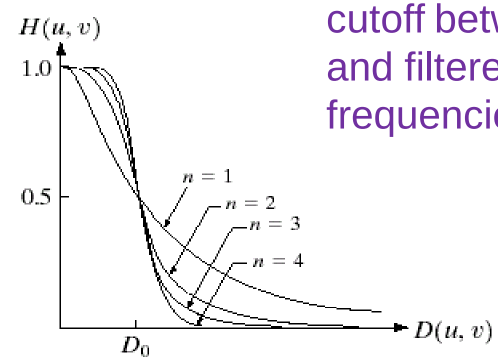
Unlike the ILPF, the BLPF transfer function does not have a sharp discontinuity that establishes a clear cutoff between passed and filtered frequencies.



Perspective plot of the transfer function



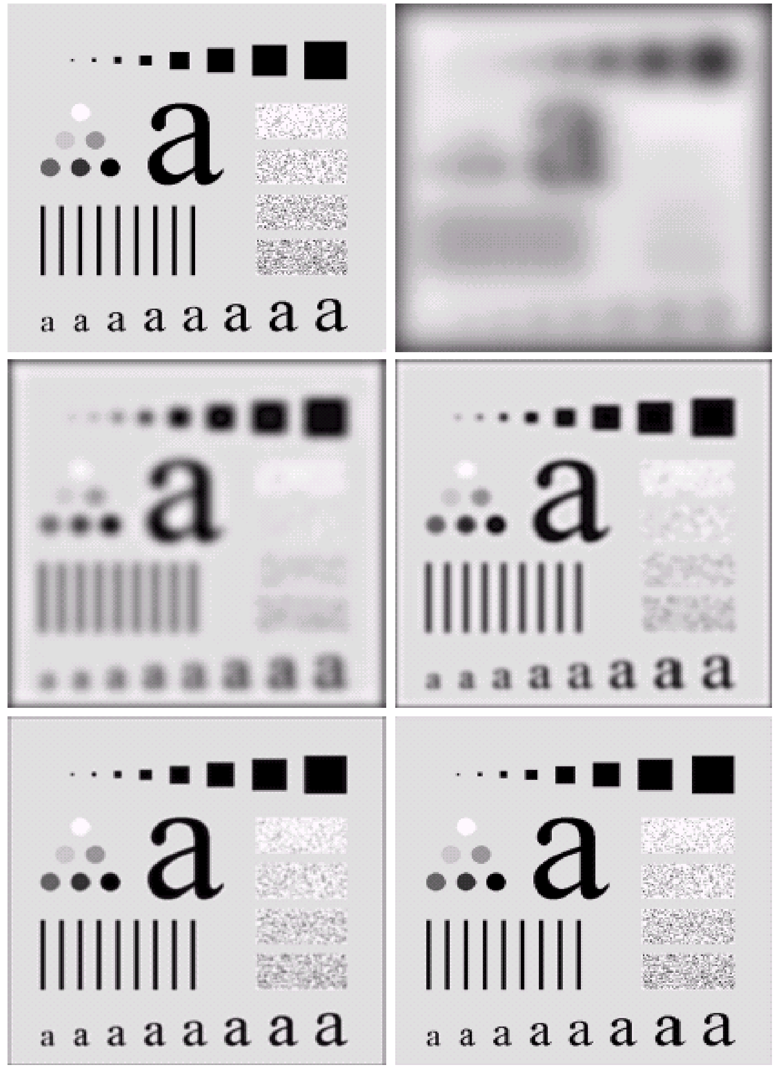
Displayed as an image



Radial cross section

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Butterworth Low Pass Filter



Original image

Result of filtering with Butterworth filter of order 2 and cutoff radius 5

Result of filtering with Butterworth filter of order 2 and cutoff radius 15

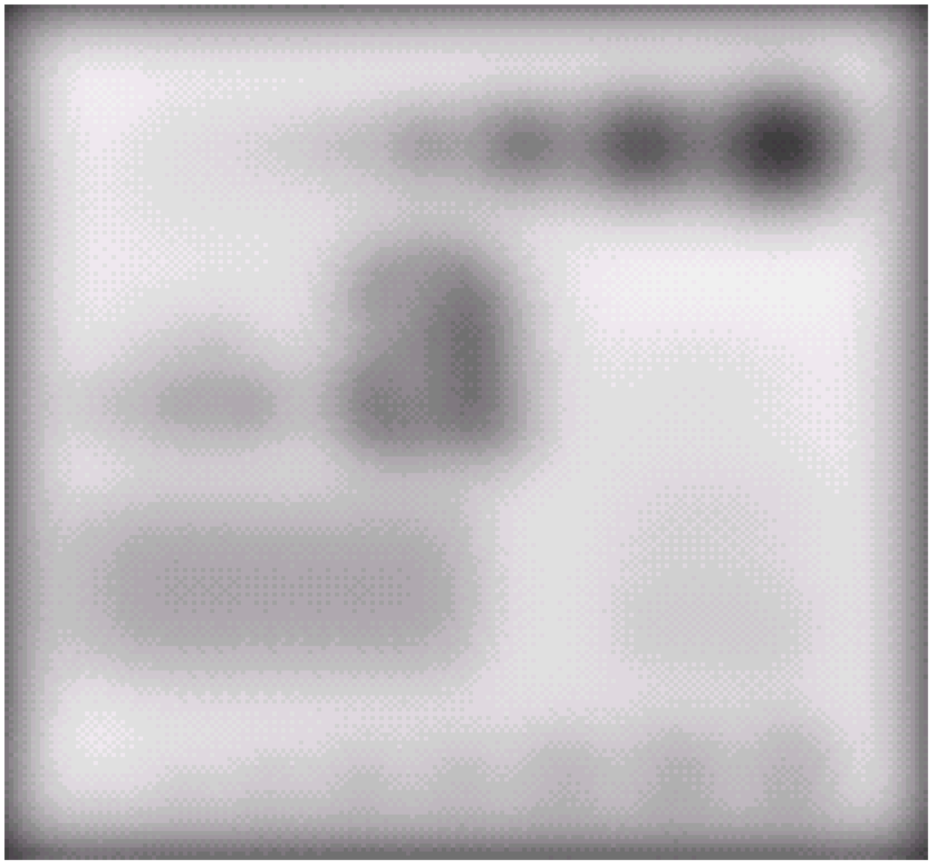
Result of filtering with Butterworth filter of order 2 and cutoff radius 30

Result of filtering with Butterworth filter of order 2 and cutoff radius 80

Result of filtering with Butterworth filter of order 2 and cutoff radius 230

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Butterworth Low Pass Filter



Result of filtering with Butterworth filter of order 2 and cutoff radius 5

No Ringing artifact.

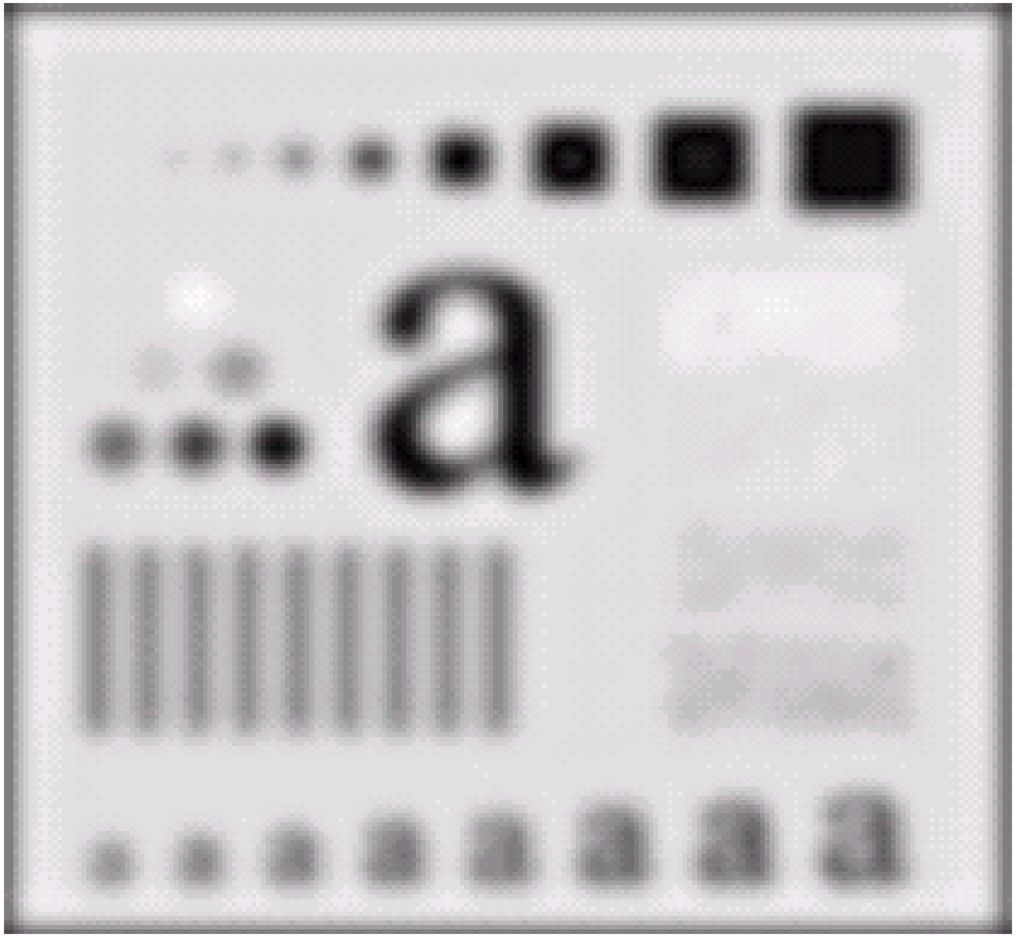
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Butterworth Low Pass Filter

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Result of filtering with Butterworth filter of order 2 and cutoff radius 15

No Ringing artifact.



Butterworth Low Pass Filter

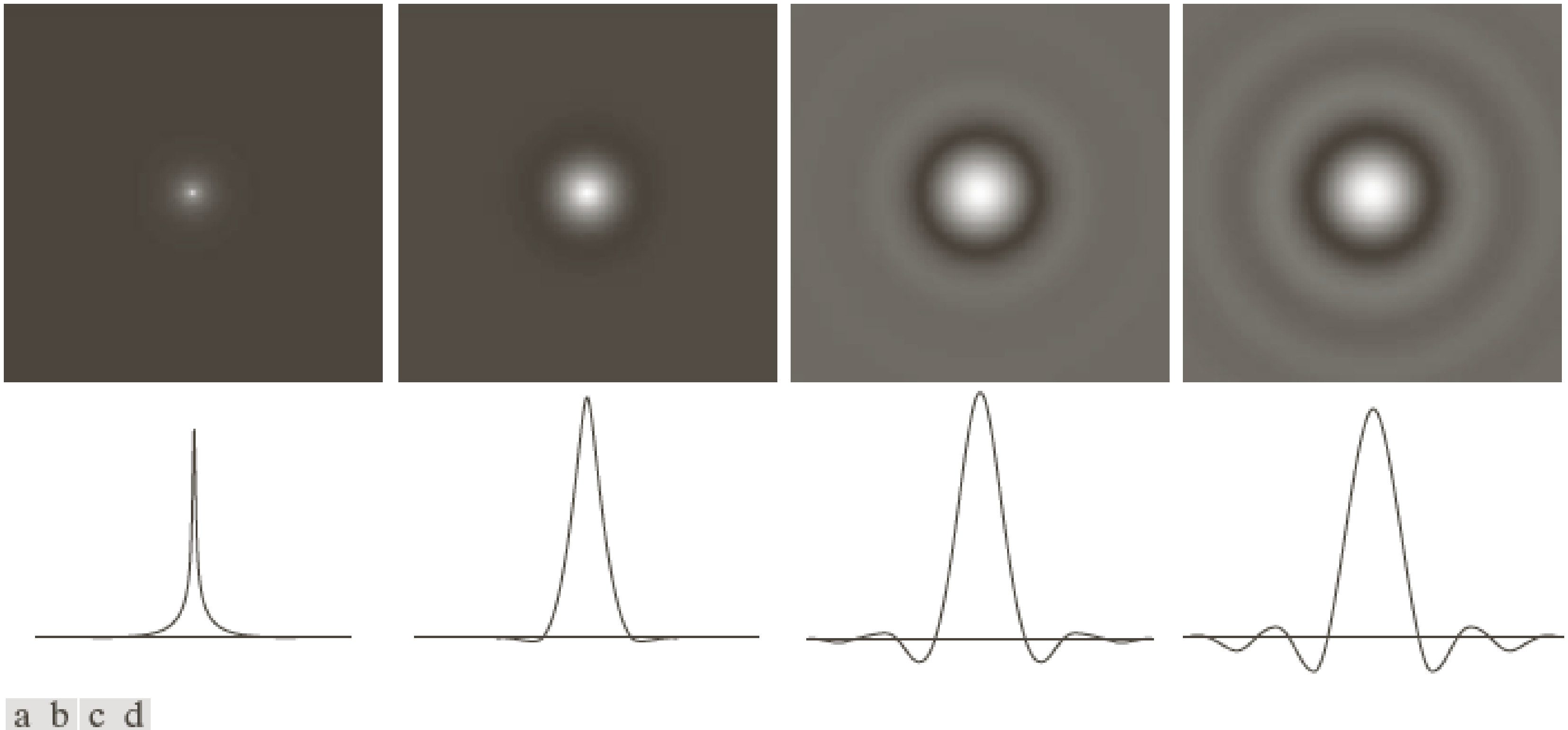
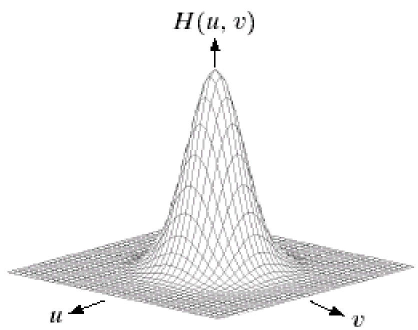


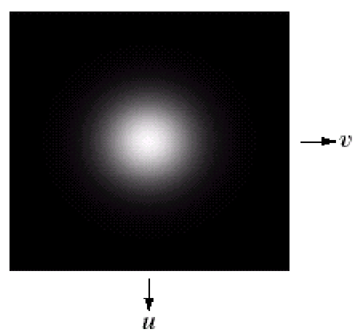
FIGURE 4.46 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is 1000 × 1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

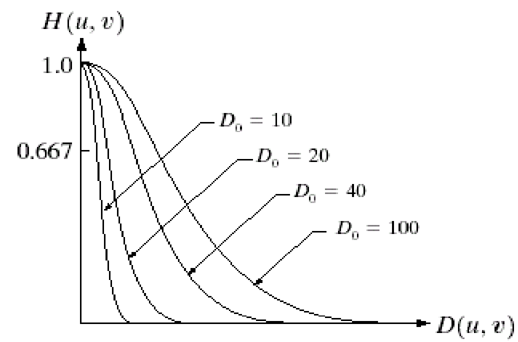
Gaussian Low Pass Filter

The transfer function of a Gaussian lowpass filter is defined as:

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





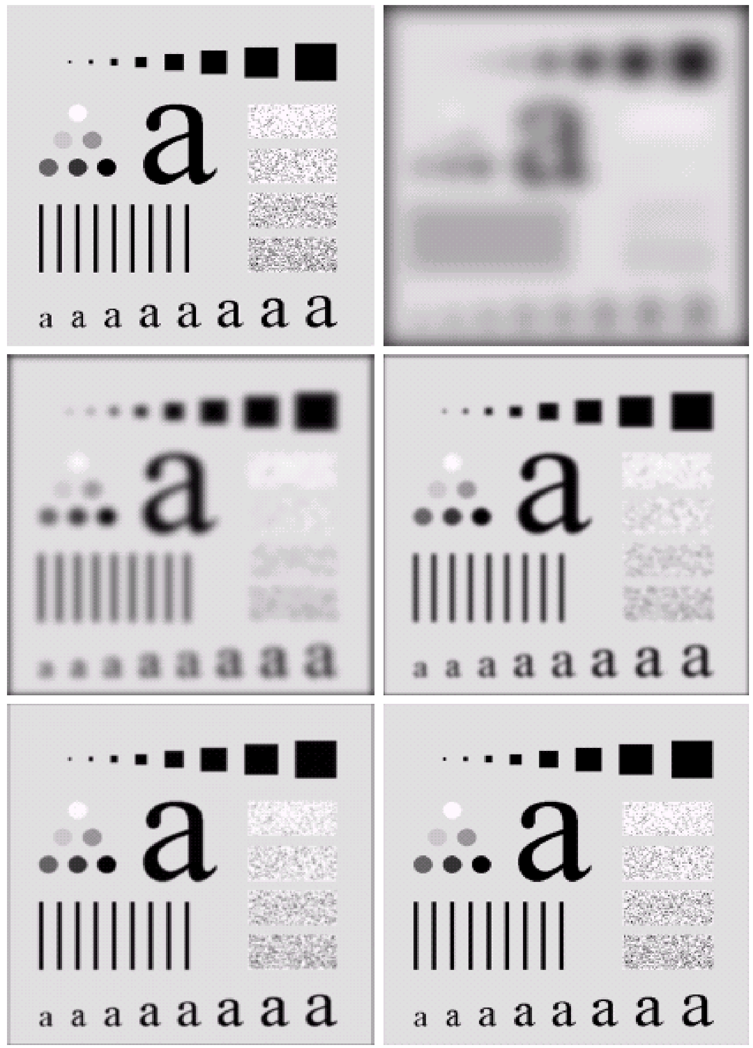


Perspective plot of the transfer function

Displayed as an image

Radial cross section

Gaussian Low Pass Filter



Original image

Result of filtering with Gaussian filter with cutoff radius 5

Result of filtering with Gaussian filter with cutoff radius 15

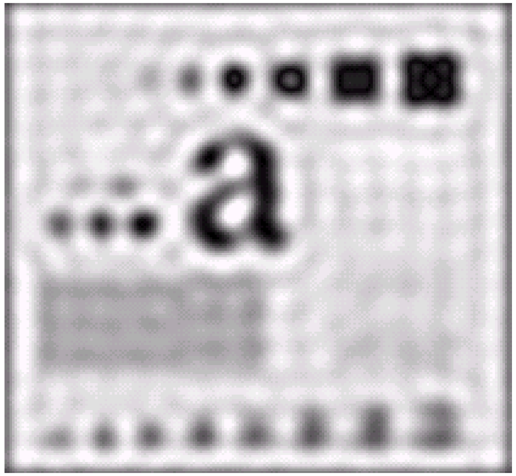
Result of filtering with Gaussian filter with cutoff radius 30

Result of filtering with Gaussian filter with cutoff radius 85

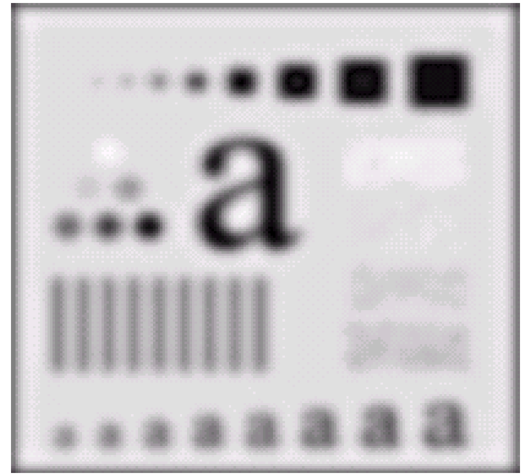
Result of filtering with Gaussian filter with cutoff radius 230

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

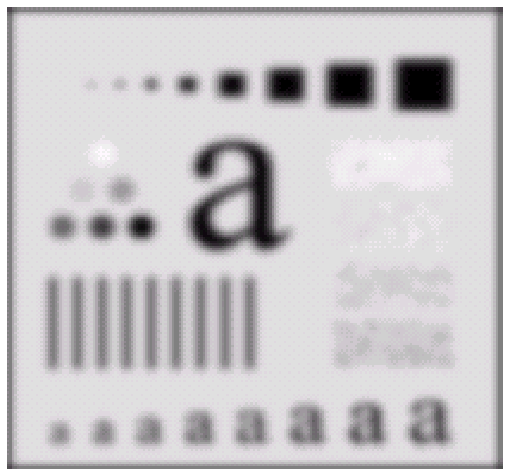
Low Pass Filters Comparison



Result of filtering with ideal low pass filter of radius 15



Result of filtering with Butterworth filter of order 2 and cutoff radius 15



Result of filtering with Gaussian filter with cutoff radius 15

Gaussian filter did not achieve as much smoothing as the Butterworth filter.

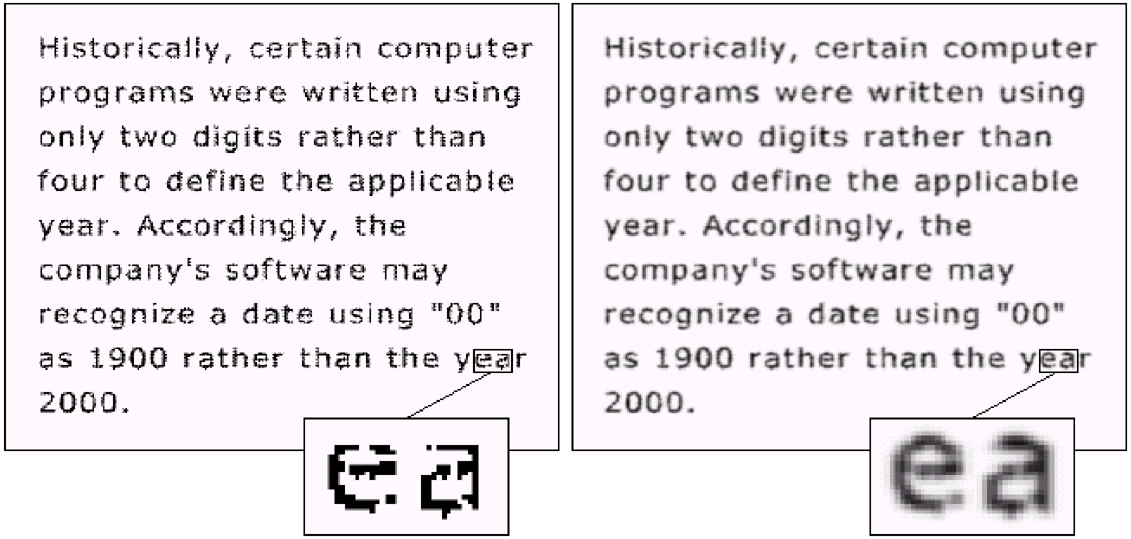
This is happened because the profile of Gaussian is not as “tight” as the profile of the Butterworth filter.

However, the results are quite comparable in general, as we are assured no ringing in the case of GLPF.

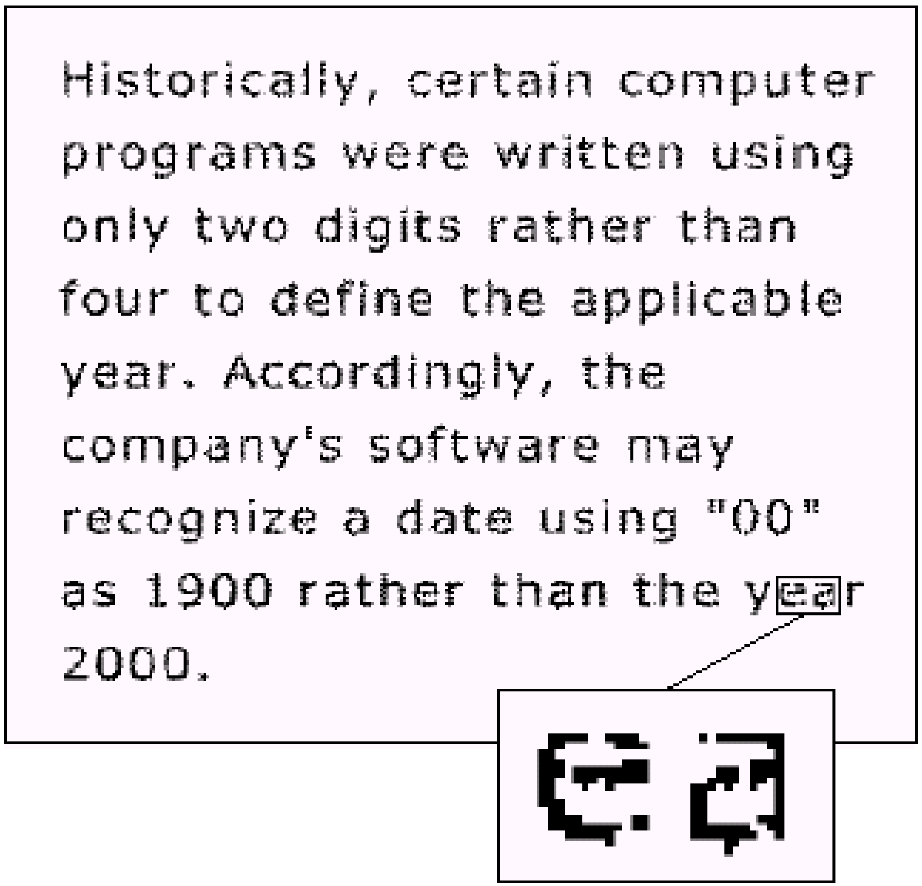
This is an important characteristics in practice, especially in situations where any type of artifact (e,g., medical imaging) is not acceptable.

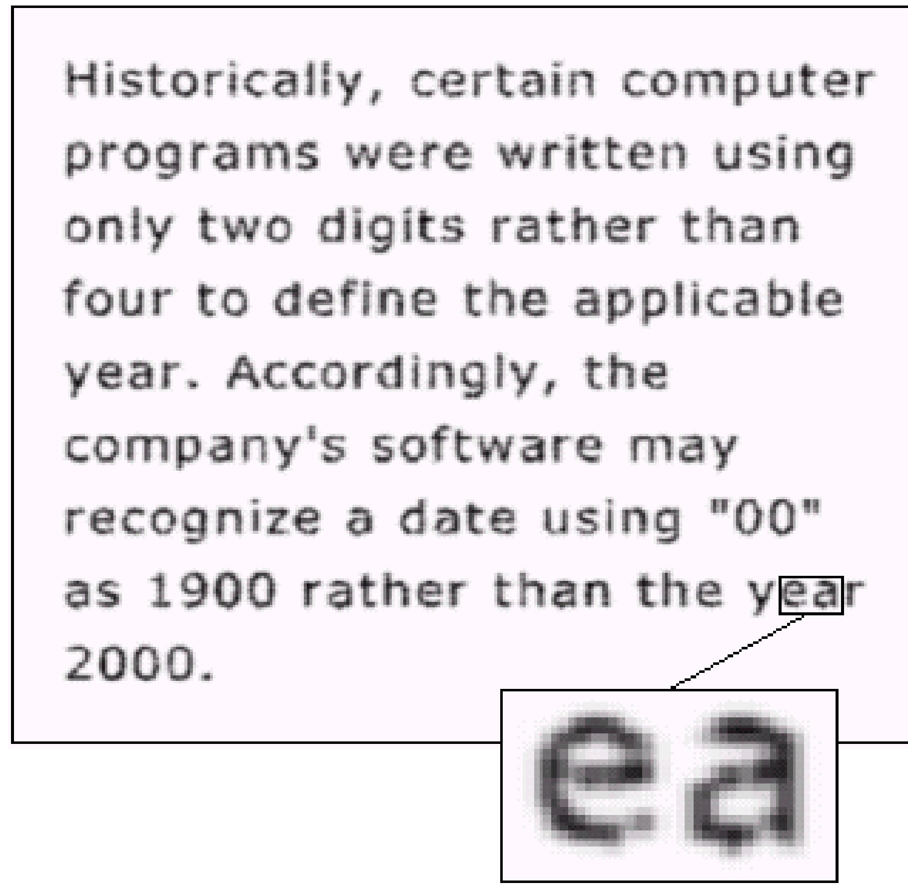
Low Pass Filtering Example

A low pass Gaussian filter is used to connect broken text.



Low Pass Filtering Example





Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Low Pass Filtering Example

Different lowpass Gaussian filters used to remove blemishes (small mark) in a photograph.

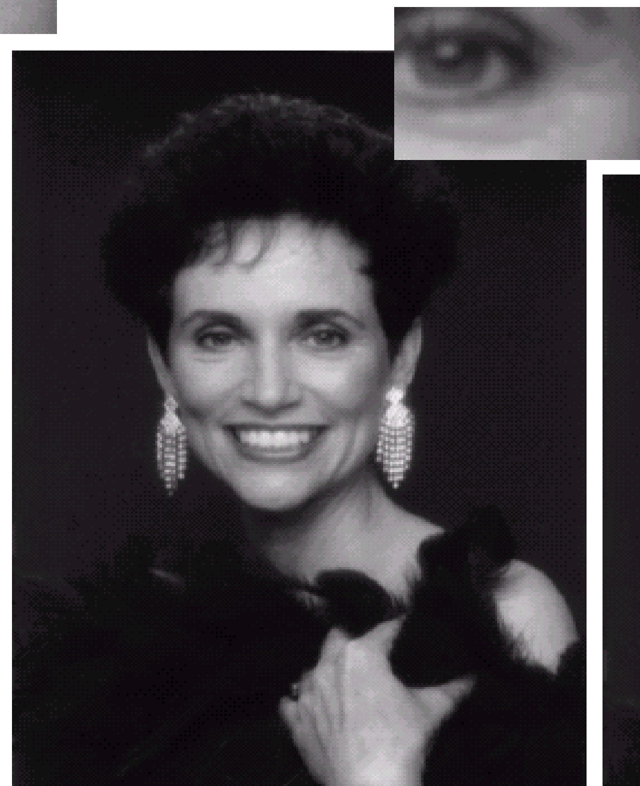
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Low Pass Filtering Example

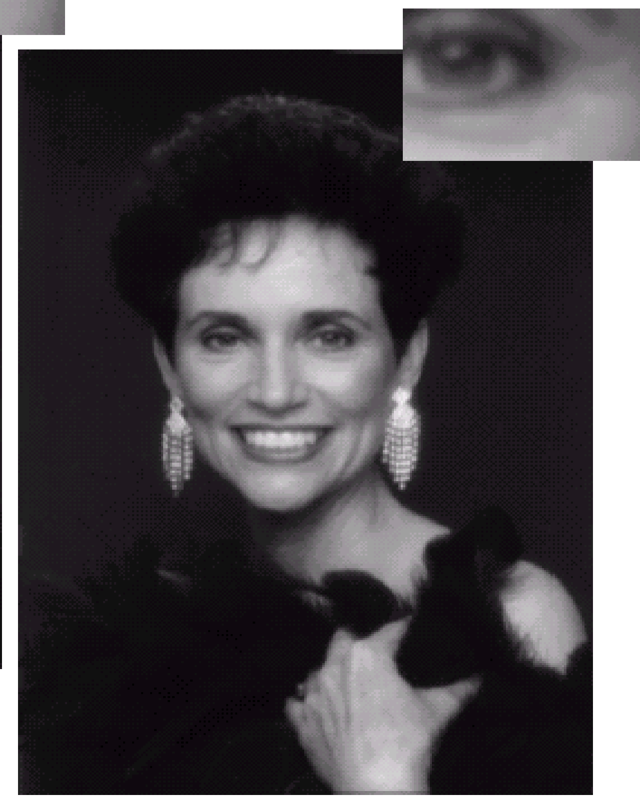
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Reduction in skin lines in the magnified sections of the filtered images.

Original image of size 1028×732



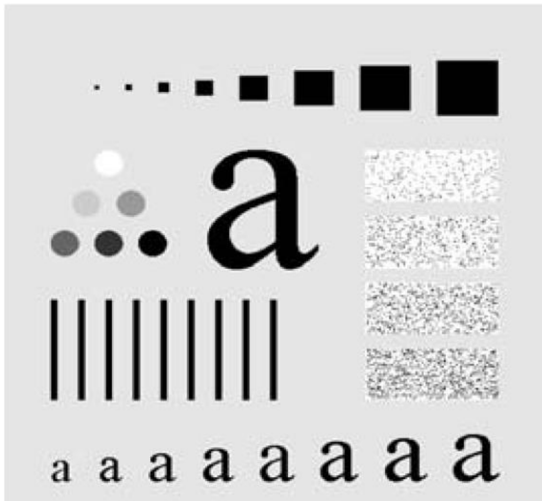
Filtered with a GLPF with $D_0 = 100$



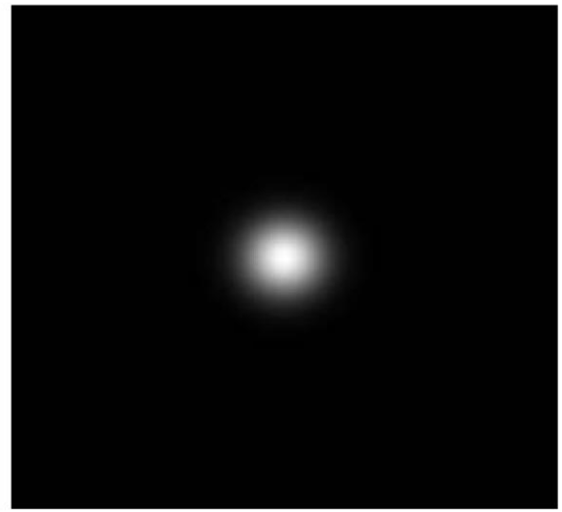
Filtered with a GLPF with $D_0 = 80$

Low Pass Filtering Example

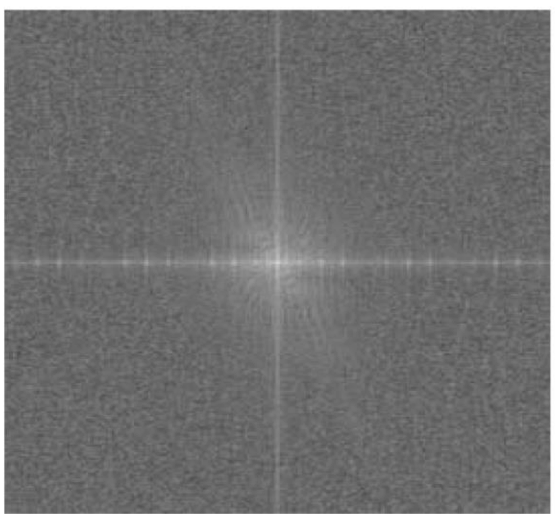
Original image



Gaussian lowpass filter



Spectrum of original image



Processed image

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Sharpening in the Frequency Domain

Edges and fine detail in images are associated with high frequency components.

High pass filters – only pass the high frequencies, drop the low ones.

High pass frequencies are precisely the reverse of low pass filters as:

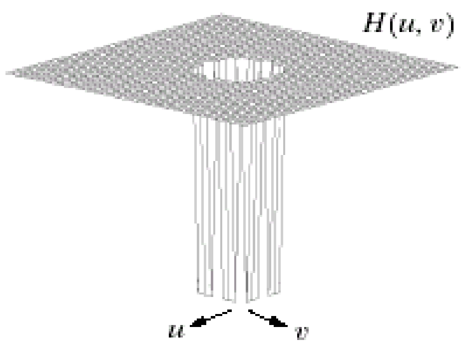
$$H_{hp}(u, v) = 1 - H_{lp}(u, v)$$

Ideal High Pass Filter

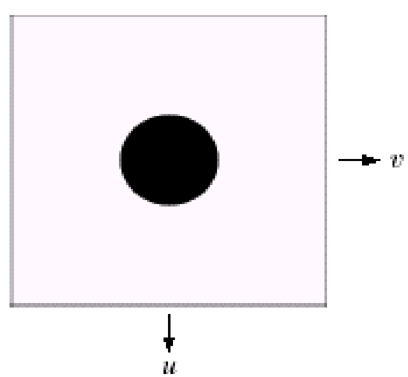
The ideal high pass filter is given as:

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

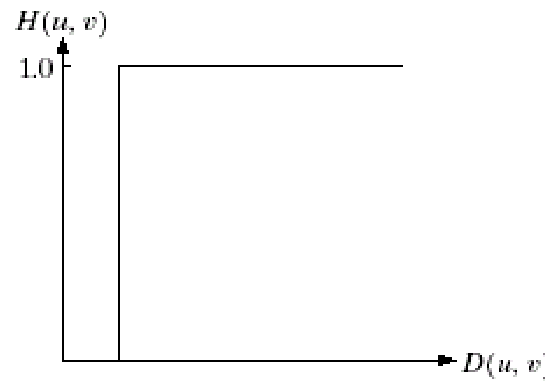
where D_0 is the cut off distance as low pass filter.



Perspective plot of the transfer function

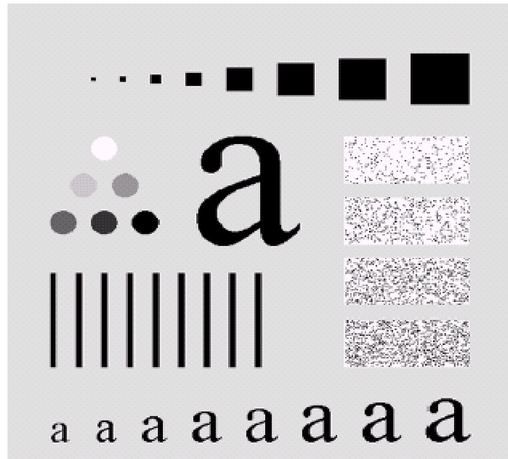


Displayed as an image

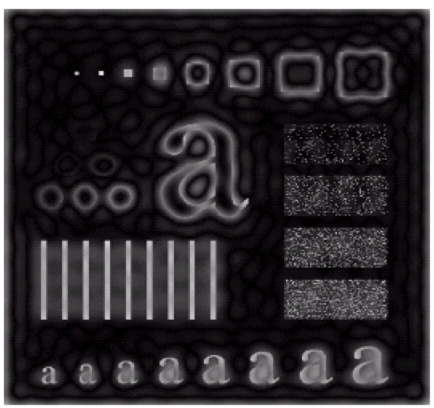


Radial cross section

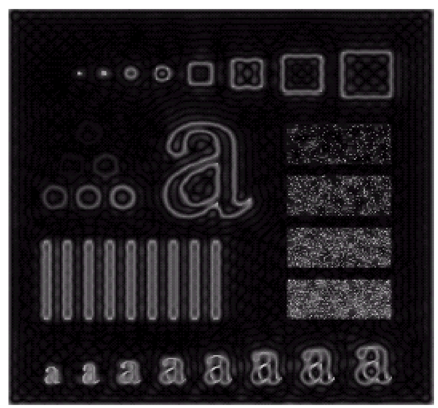
Ideal High Pass Filter



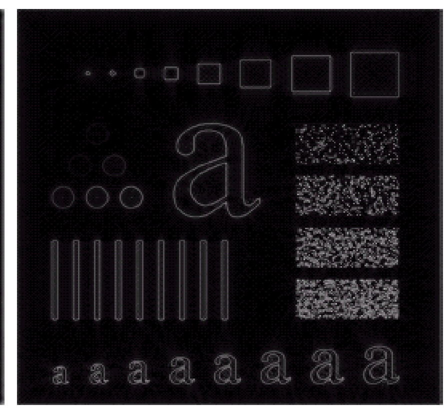
The ringing is so severe that it produced distorted, thickened object boundaries.



Results of ideal high pass filtering with D_0 = 15



Results of ideal high pass filtering with D_0 = 30



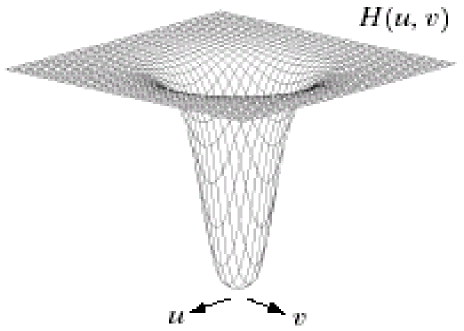
Results of ideal high pass filtering with D_0 = 80

Butterworth High Pass Filter

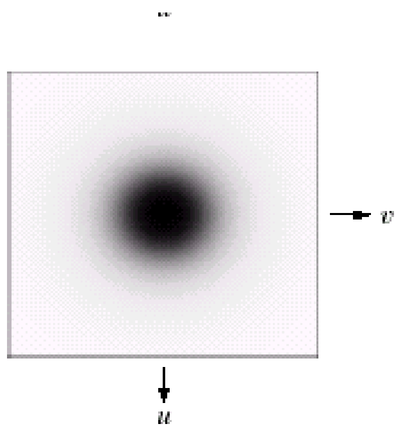
The Butterworth high pass filter is given as:

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

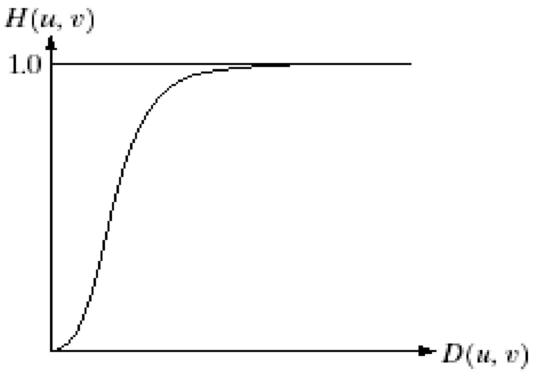
where n is the order and D_0 is the cut off distance as low pass filter.



Perspective plot
of the transfer function



Displayed as an
image

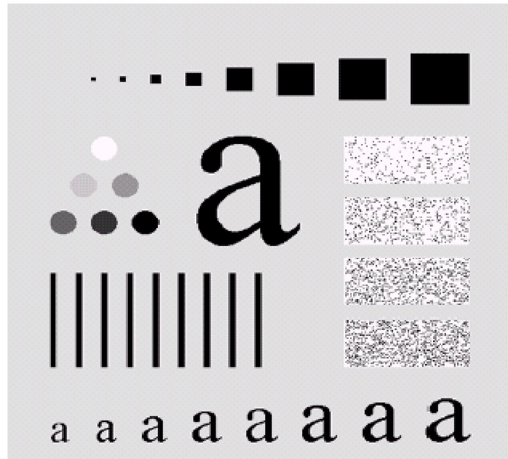


Radial cross section

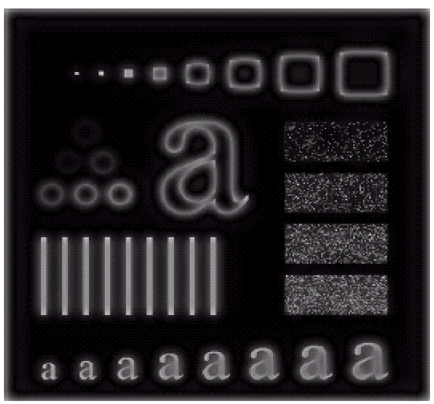
Butterworth High Pass Filter

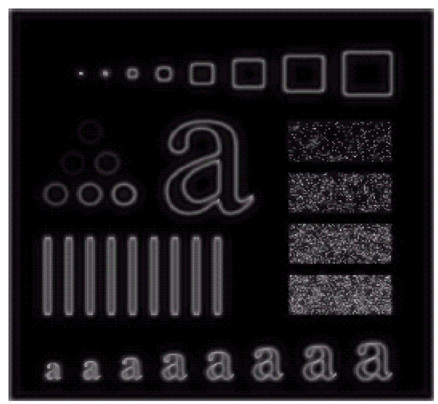
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

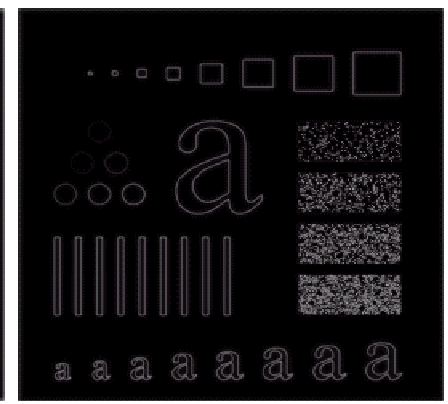
The boundaries are much less distorted than the outcome of IHPF.



Results of Butterworth high pass filtering of order 2 with $D_0 = 15$







Results of Butterworth high pass filtering of order 2 with $D_0 = 80$

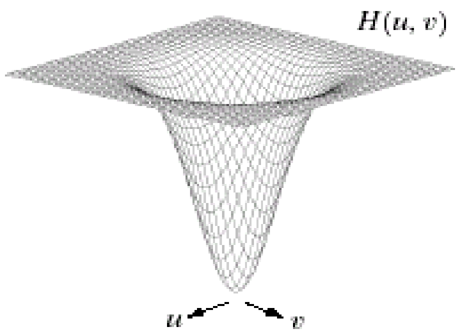
Results of Butterworth high pass filtering of order 2 with $D_0 = 30$

Gaussian High Pass Filter

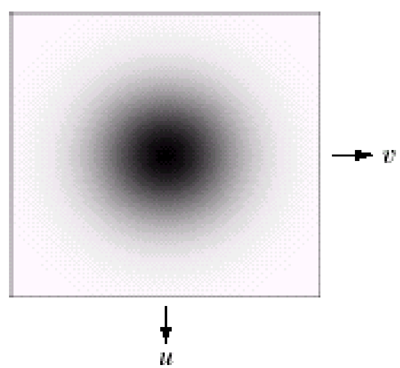
The Gaussian high pass filter is given as:

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

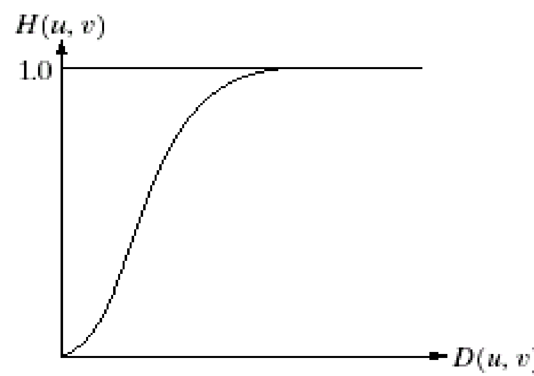
where D_0 is the cut off distance as before.



Perspective plot of the transfer function

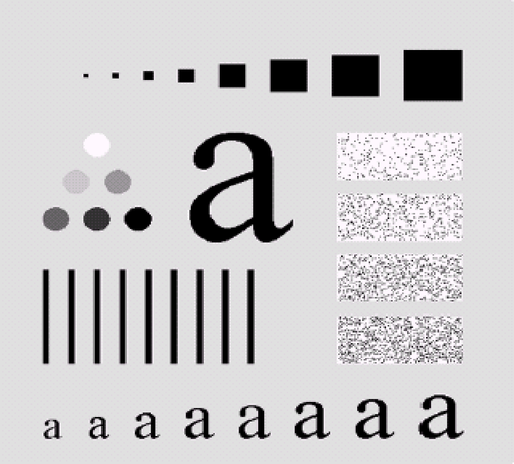


Displayed as an image

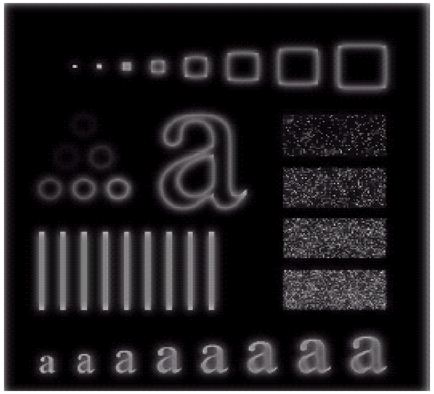


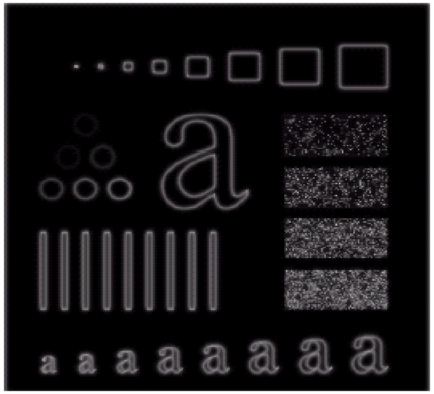
Radial cross section

Gaussian High Pass Filter

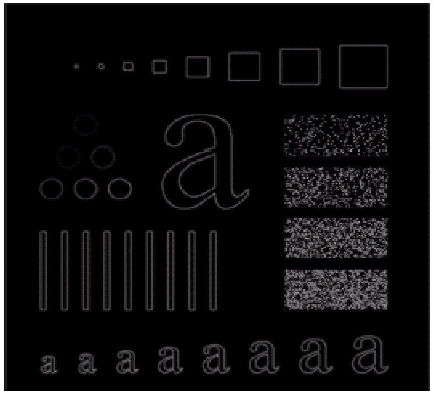


Results of Gaussian high pass filtering with $D_0 = 15$





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

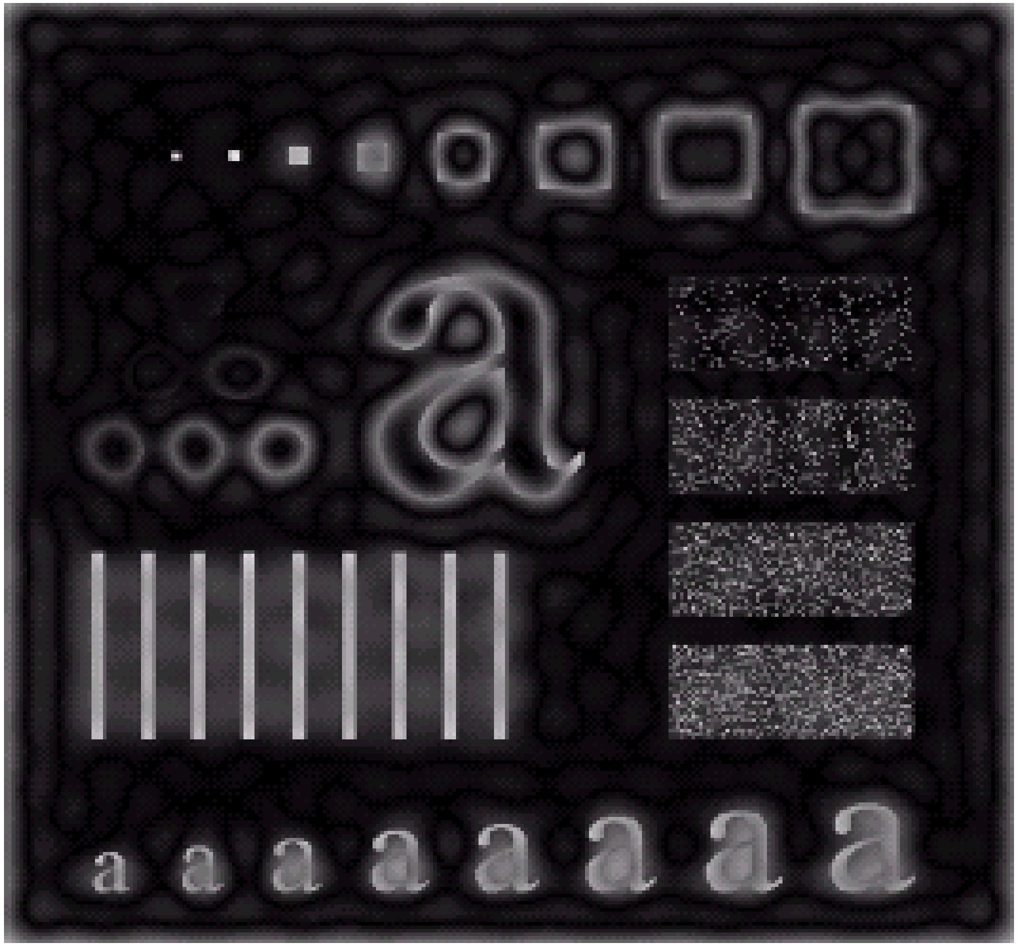


Results of Gaussian high pass filtering with $D_0 = 80$

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Highpass Filter Comparison

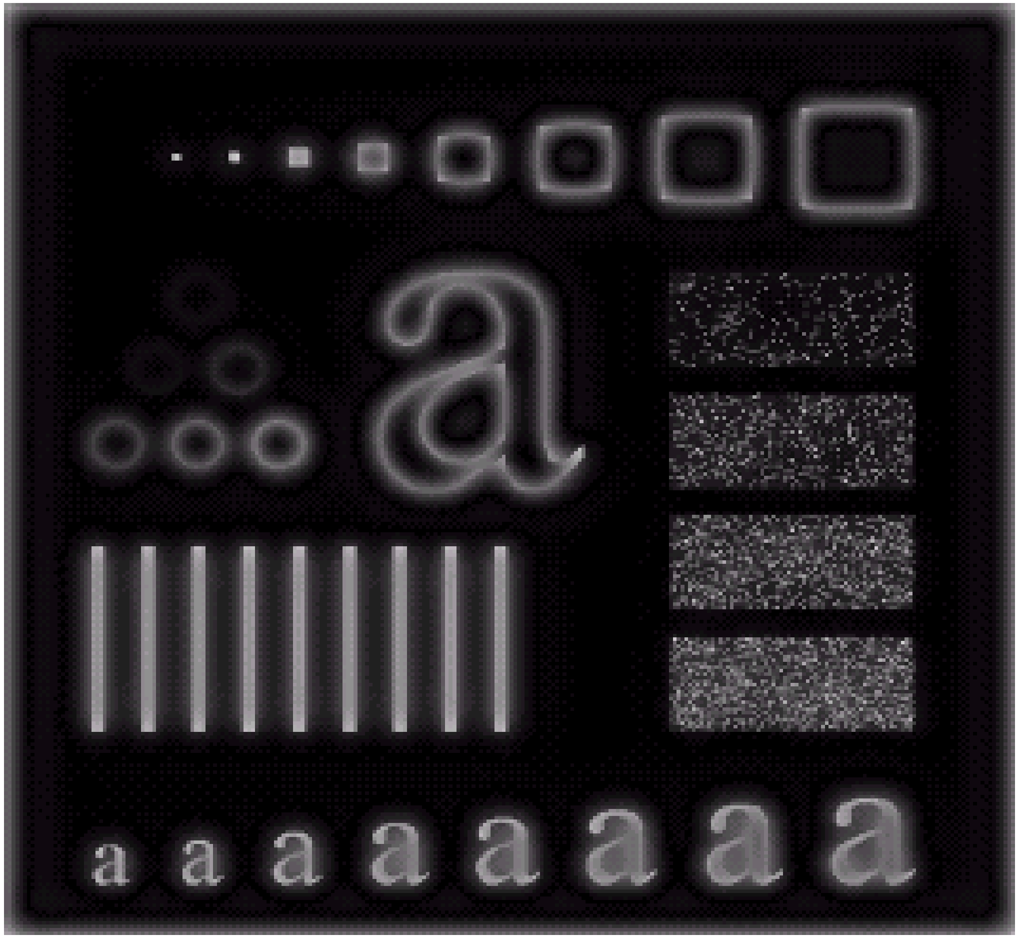
Images taken from Gonzalez & Woods, Digital Image Processing (2002)



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

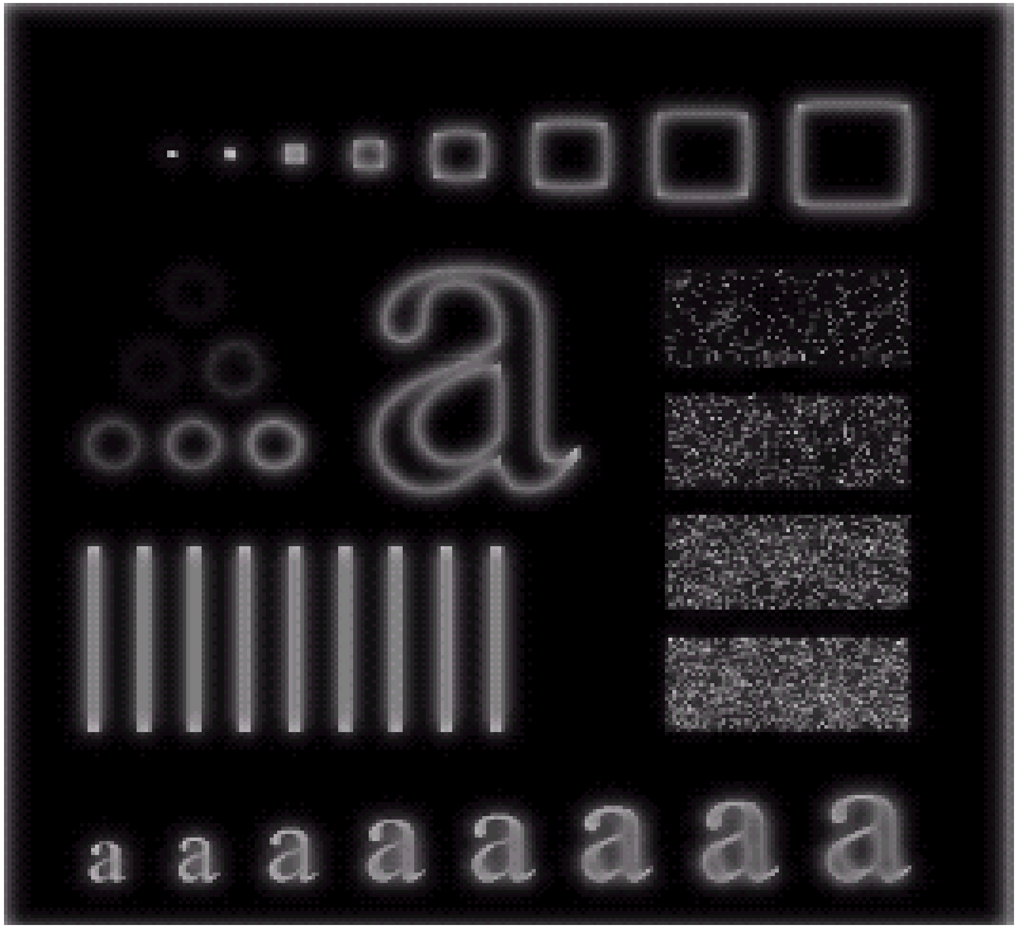
Distorted thickened object boundary.

Highpass Filter Comparison



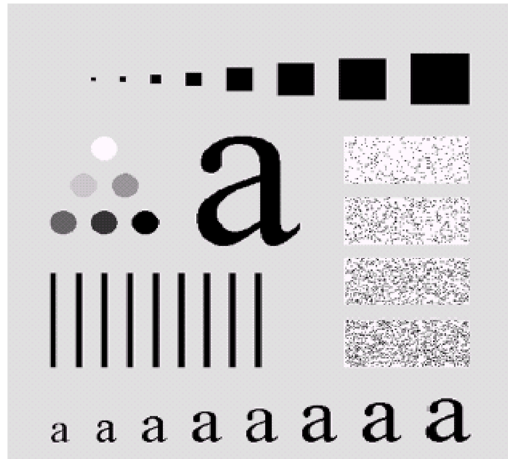
Results of Butterworth high pass filtering of order 2 with D_0 = 15

Highpass Filter Comparison



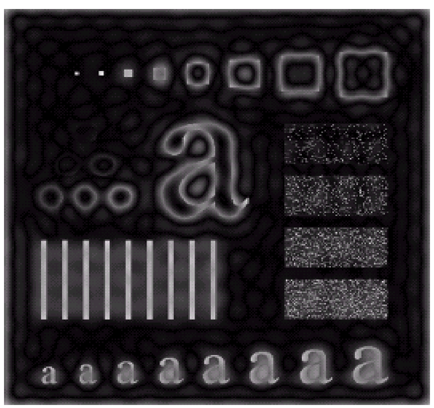
Results of Gaussian high pass filtering with D_0 = 15

Highpass Filter Comparison

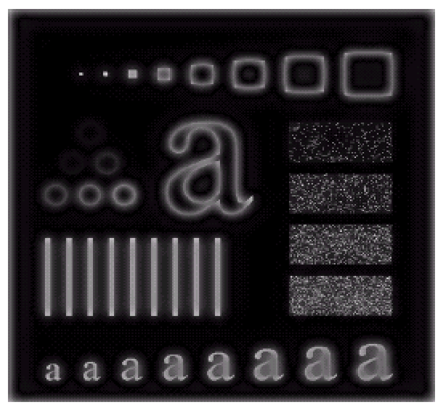


The results obtained by Gaussian filter are smoother than with the other two filters.

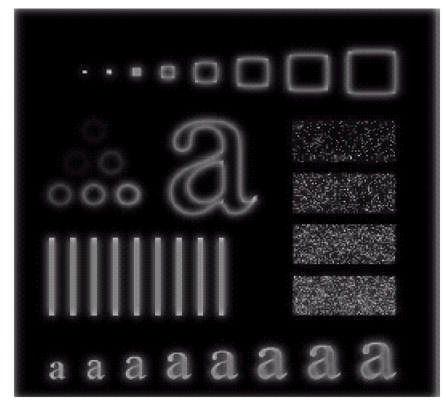
Even the filtering of the smaller objects and thin bars is clear with this filter.



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

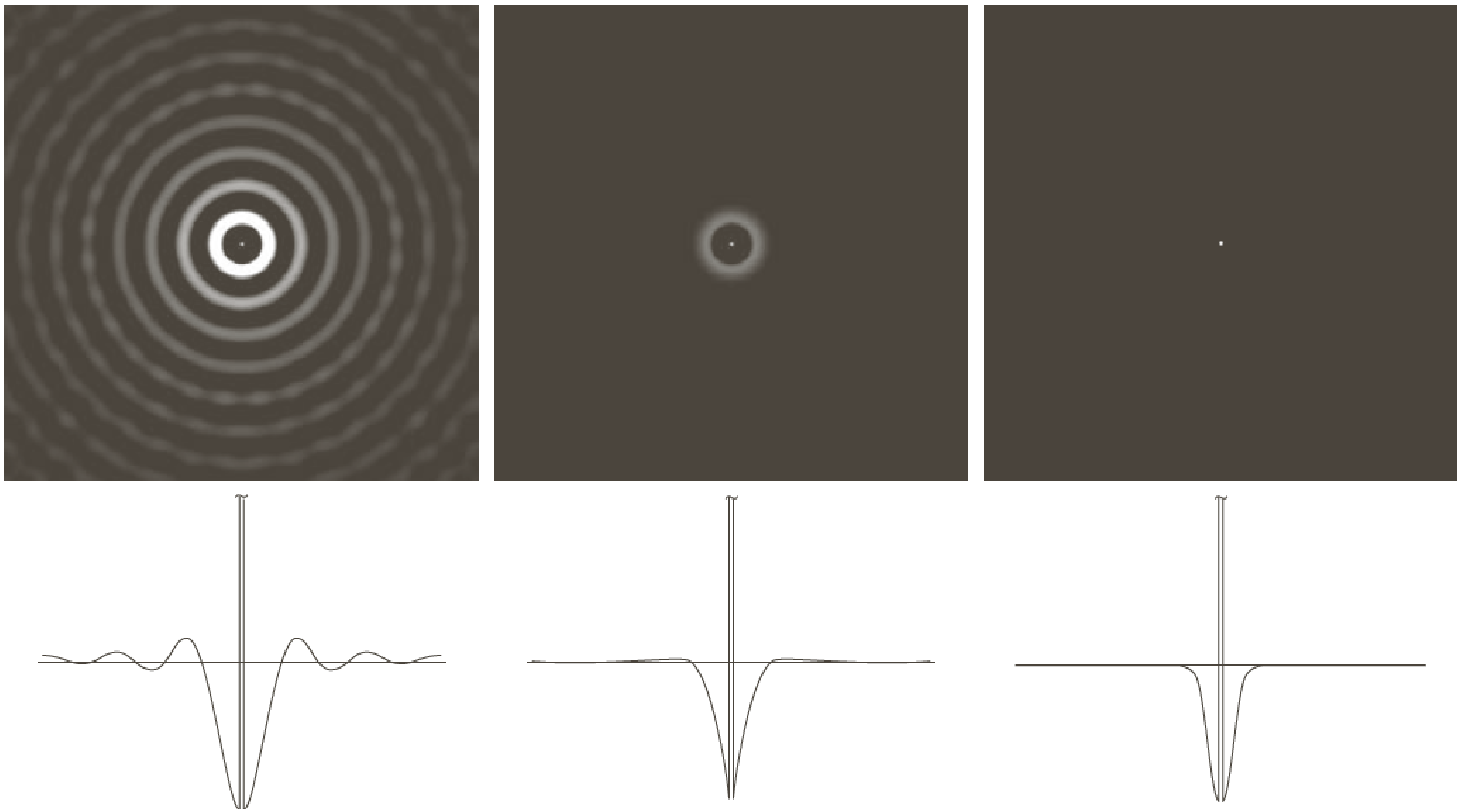


Results of Butterworth high pass filtering of order 2 with $D_0 = 15$



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

Highpass Filter Comparison



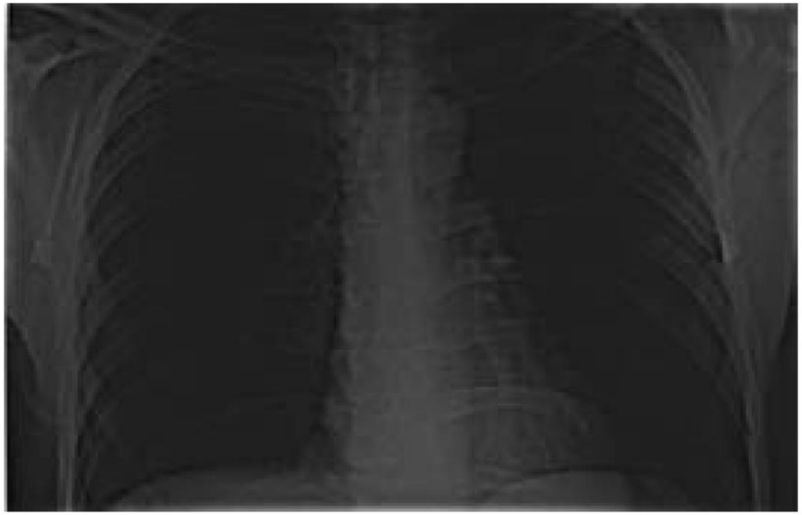
a b c

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.

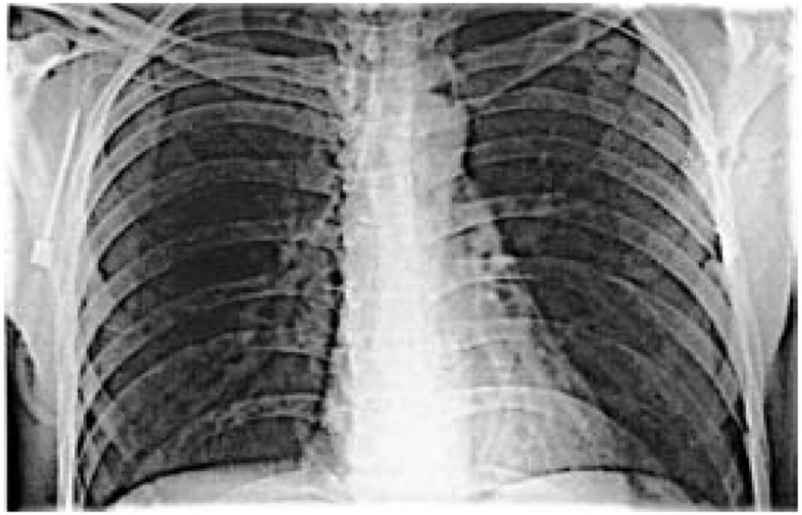
Highpass Filtering Example

Original image

Highpass filtering result



High frequency emphasis result



After histogram equalisation

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Convolution

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

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

The double arrow is used to indicate that the expression on the left (spatial convolution) can be obtained by taking the inverse Fourier transform of the expression on the right [the product $F(u,v)H(u,v)$ in the frequency domain].

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

Convolution theorem defines the correspondence between filtering in the spatial domain and frequency domain.

Convolution

Convolution equation is nothing more than an implementation for

(1) flipping one function about origin;

(2) shifting that function with respect to the other by changing the values of (x, y); and

(3) Computing a sum of products over all values of m and n, for each displacement (x, y).

The displacements (x, y) are integer increments that stop when the function no longer overlap.

Correspondence between Filtering in the Spatial domain and Frequency Domain

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

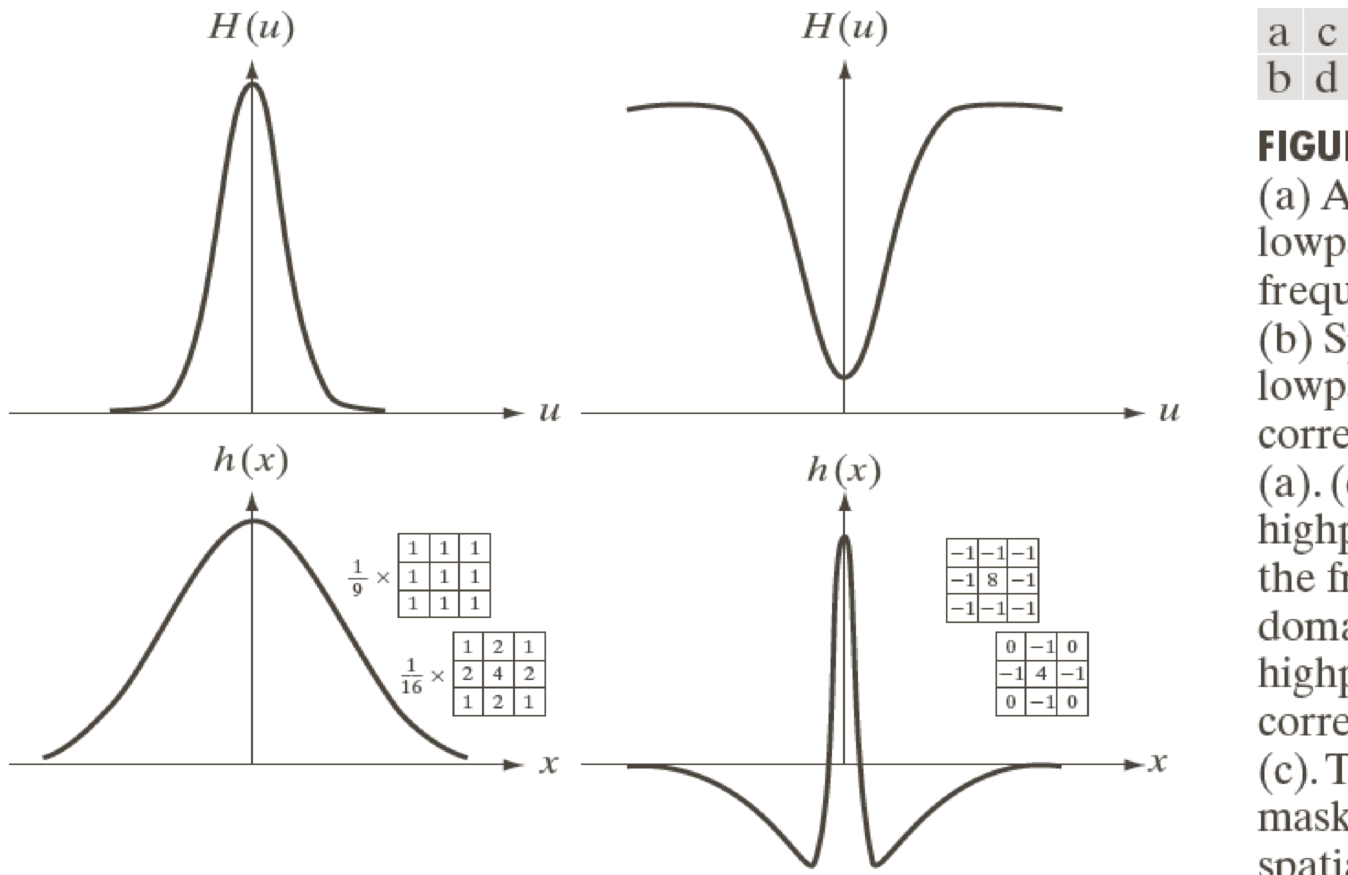
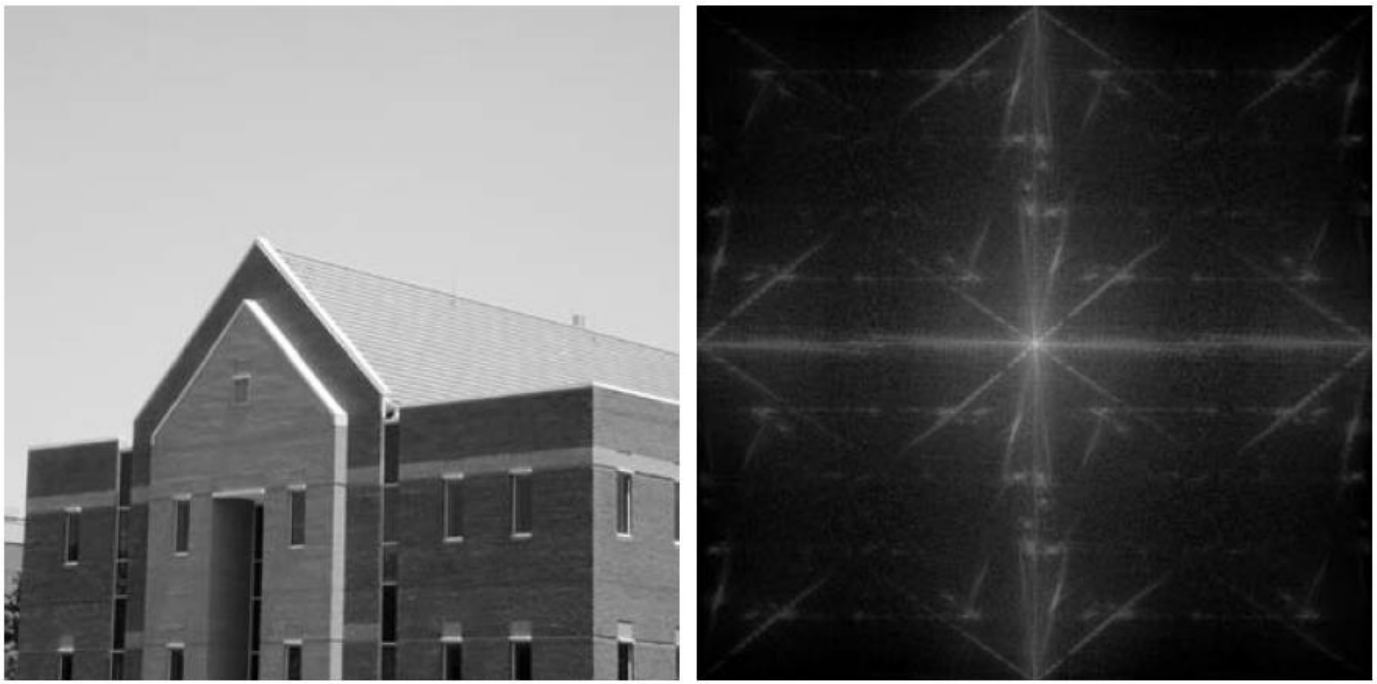


FIGURE 4.37 (a) A 1-D Gaussian lowpass filter in the frequency domain. (b) Spatial lowpass filter corresponding to (a). (c) Gaussian highpass filter in the frequency domain. (d) Spatial highpass filter corresponding to (c). The small 2-D masks shown are spatial filters we used in Chapter 3.

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

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

Correspondence between Filtering in the Spatial domain and Frequency Domain

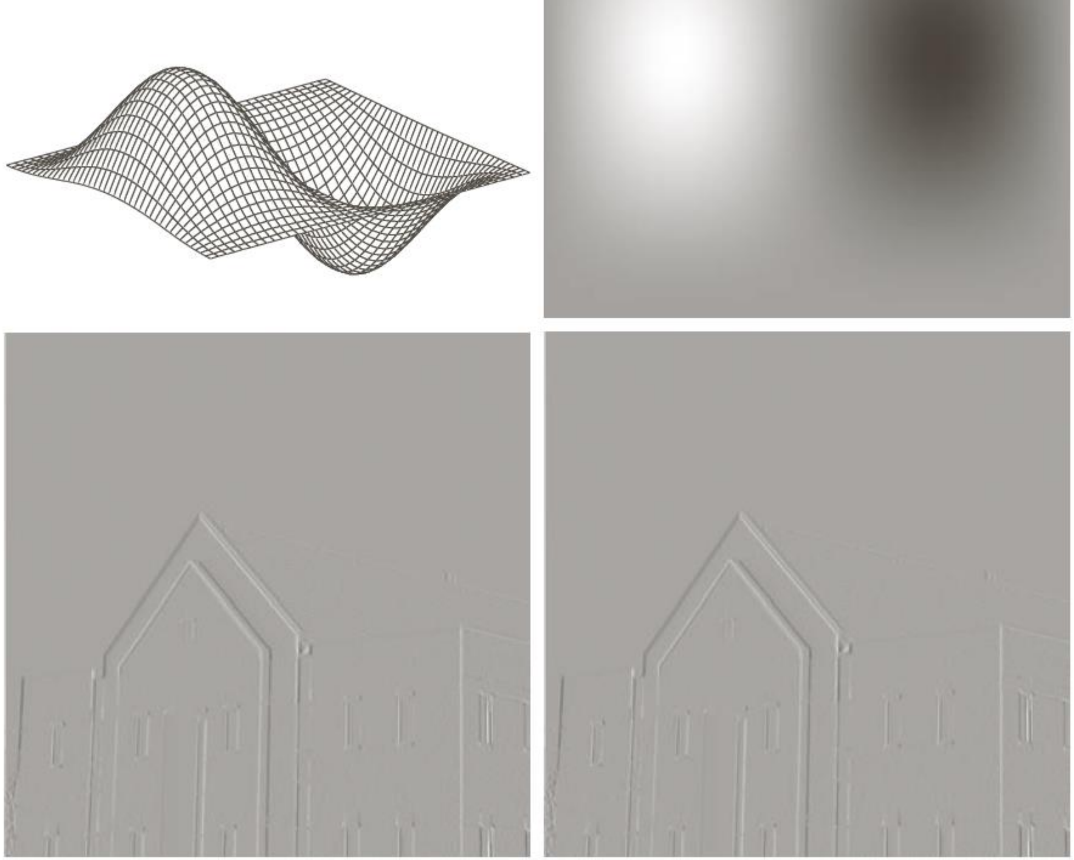


a b

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

Correspondence between Filtering in the Spatial domain and Frequency Domain

−1	0	1
−2	0	2
−1	0	1



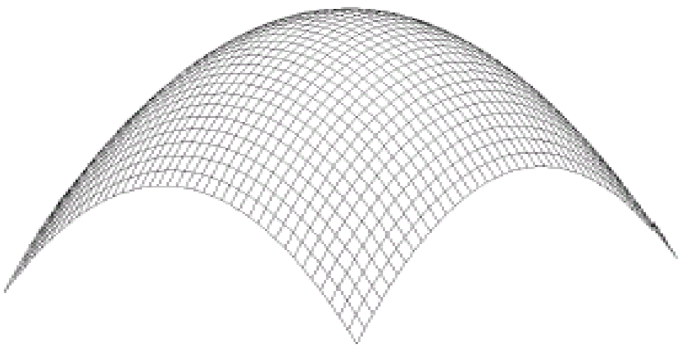
a	b
c	d

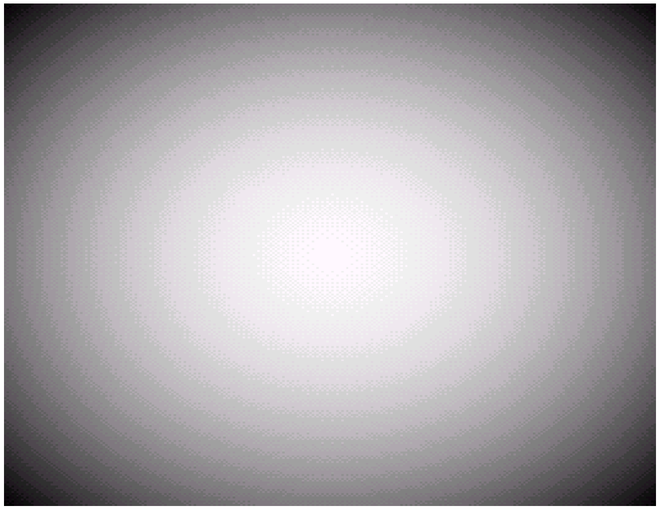
FIGURE 4.39 (a) A spatial mask and perspective plot of its corresponding frequency domain filter. (b) Filter shown as an image. (c) Result of filtering Fig. 4.38(a) in the frequency domain with the filter in (b). (d) Result of filtering the same image with the spatial filter in (a). The results are identical.

Laplacian In The Frequency Domain

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

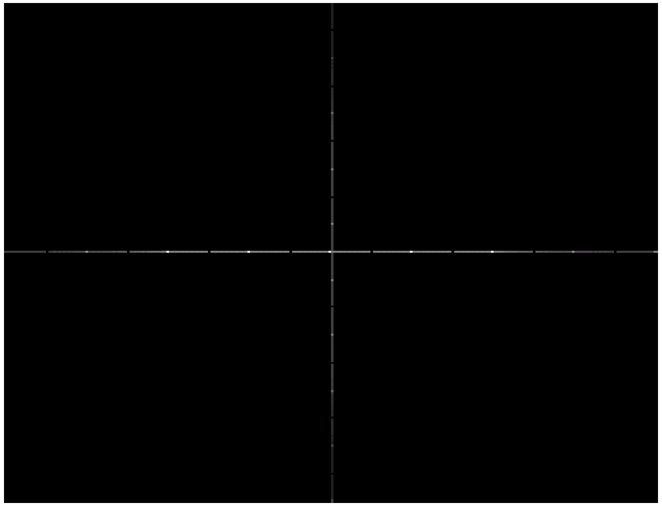
Laplacian in the frequency domain

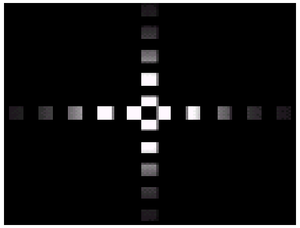


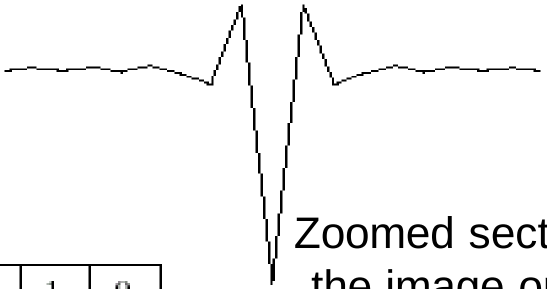


2-D image of Laplacian in the frequency domain

Inverse DFT of Laplacian in the frequency domain







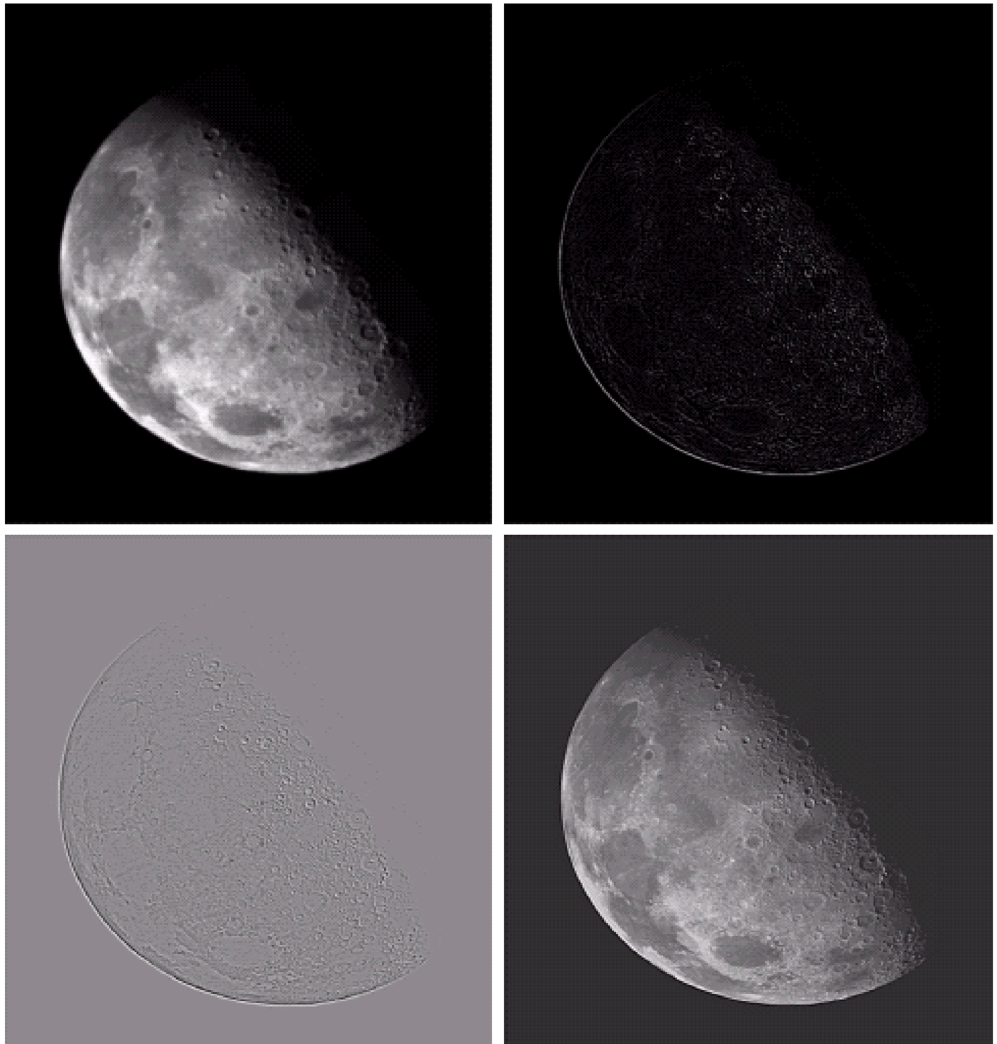
0	1	0
1	−4	1
0	1	0

Zoomed section of the image on the left compared to spatial filter

Frequency Domain Laplacian Example

Original image

Laplacian filtered image



Laplacian image scaled

Enhanced image

Correlation

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

$$f(x,y) \circ h(x,y) \Leftrightarrow F^*(u,v) H(u,v)$$

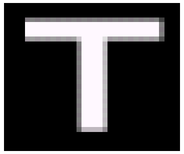
$$f^*(x,y) h(x,y) \Leftrightarrow F(u,v) \circ H(u,v)$$

If $f = x + iy \; then \; f^* = x - iy$

f^* denotes the complex conjugate of f. We normally deal with real functions (images), in which case $f^* = f$.

Application of Correlation

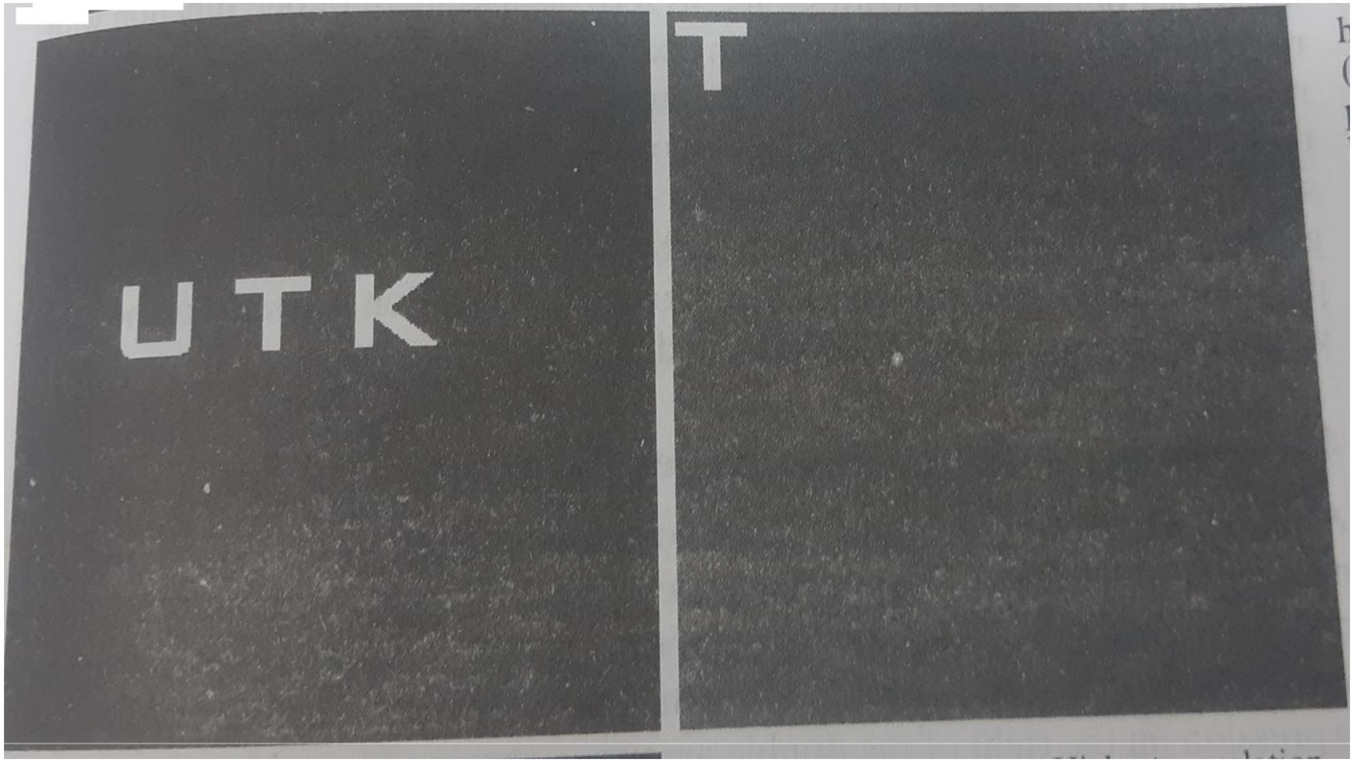
The principal use of correlation is for matching.



Template of size 38×42

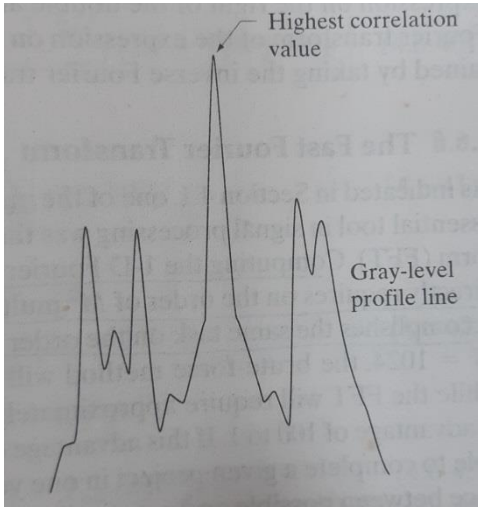
Input Image of size 256×256

Application of Correlation



Size of both the images is 298×298

Application of Correlation



Correlation function displayed as an image

Fast Fourier Transform (FFT)

The reason that Fourier based techniques have become so popular is the development of the *Fast Fourier Transform (FFT)* algorithm.

Allows the Fourier transform to be carried out in a reasonable amount of time.

Reduces the amount of time required to perform a Fourier transform by a factor of 100 – 600 times!

Frequency Domain Filtering & Spatial Domain Filtering

- Similar jobs can be done in the spatial and frequency domains.
- Filtering in the spatial domain can be easier to understand.
- Filtering in the frequency domain can be <u>much faster</u> – particularly for large images.