DIGITAL IMAGE PROCESSING

Image Enhancement:
Filtering in the Frequency Domain

Contents

In this lecture we will look at image enhancement in the frequency domain

- Jean Baptiste Joseph Fourier
- The Fourier series & the Fourier transform
- Image Processing in the frequency domain
  - Image smoothing
  - Image sharpening
- Fast Fourier Transform

Jean Baptiste Joseph Fourier

Fourier was born in Auxerre,

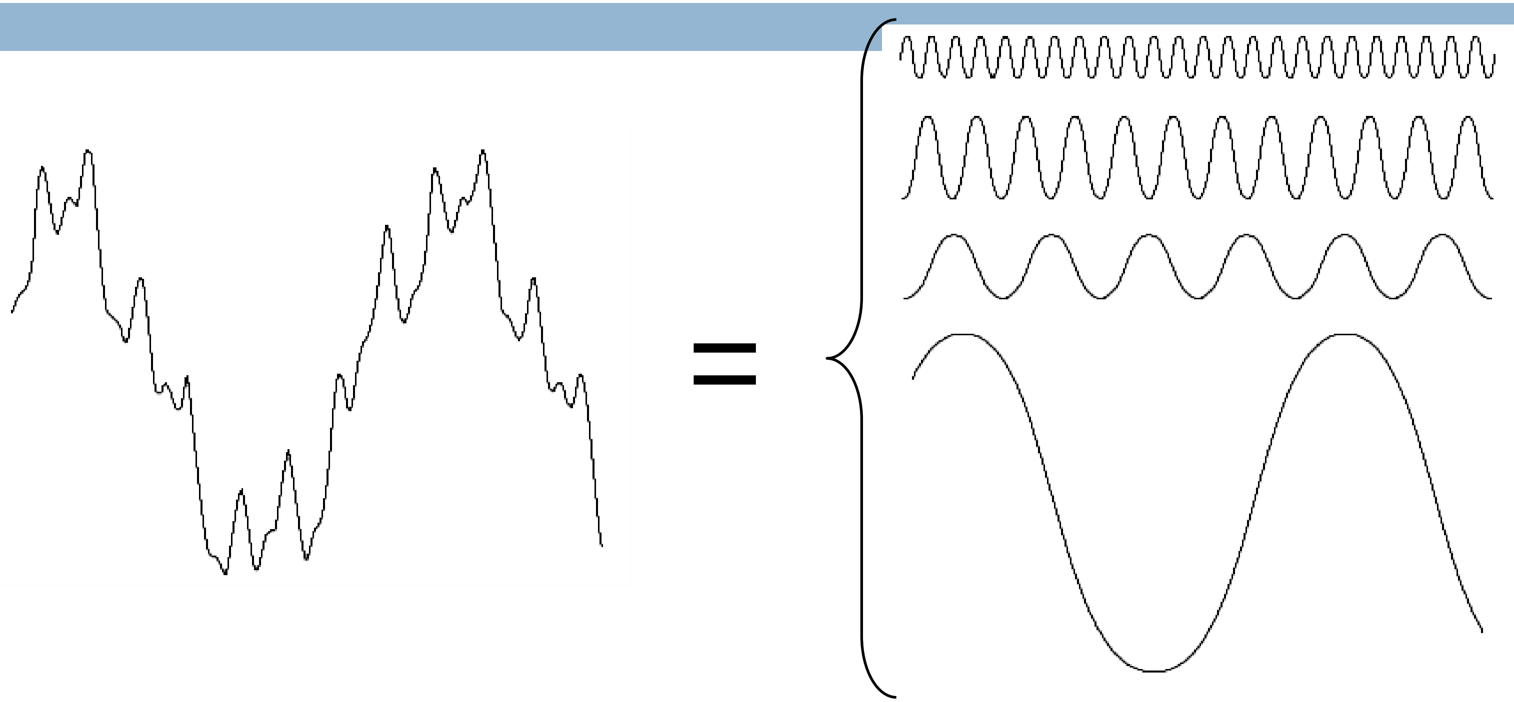
France in 1768

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

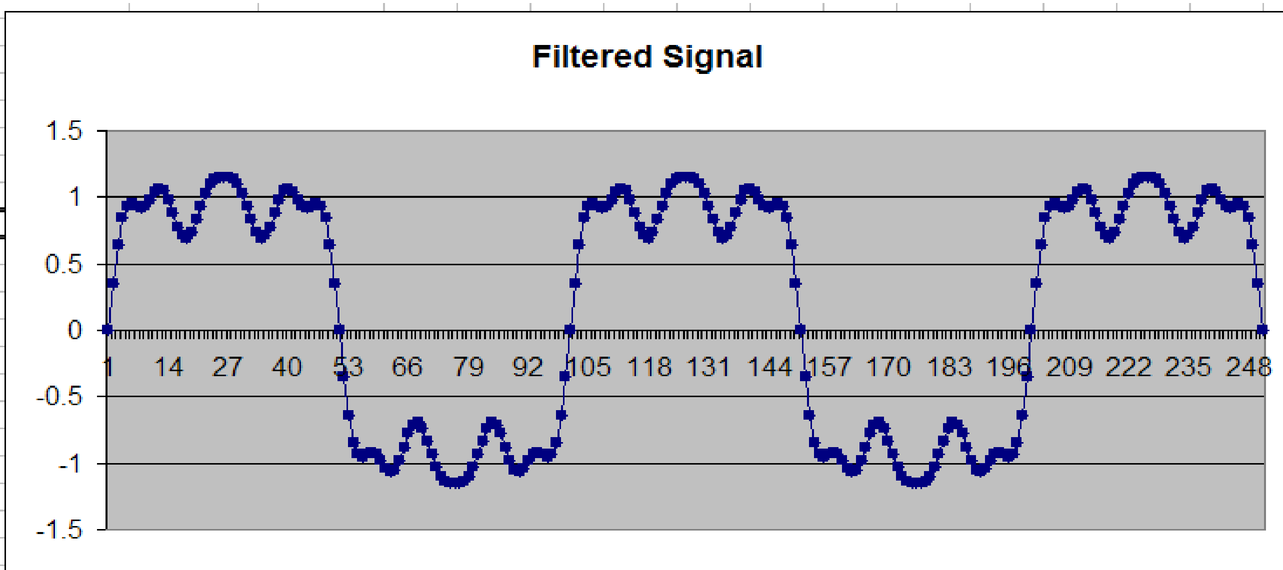
The Big Idea



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*

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

The Big Idea (cont…)



we get closer and closer to the original function as we add more and more frequencies

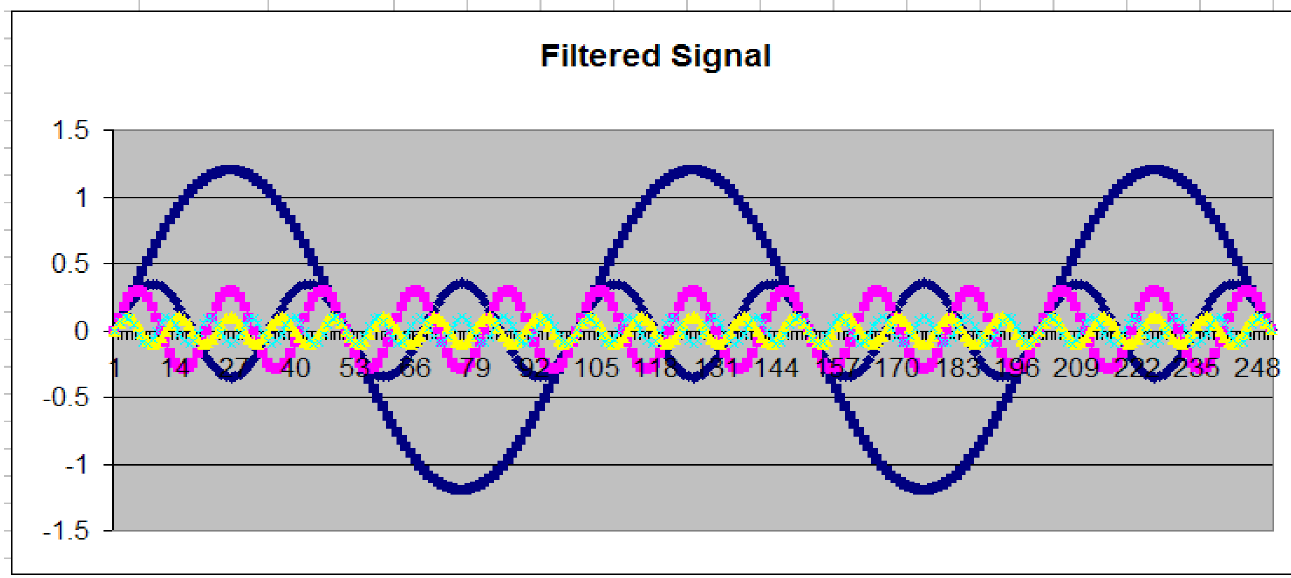


Image processing in the Frequency domain

- Why bother going into frequency domain?
- How this transformation is done?
- What does Fourier Transform tell us?
- What is the relationship between images in Spatial and Frequency domain?
- Analysis of frequency domain filters and their applications on images

Why bother going into frequency domain?

- Frequency domain representation makes it easy to visualize some characteristics of images
- It is easy to conceptualize filters in frequency domain
- Once a filter is selected in the frequency domain, it is usually implemented in the spatial domain
- Frequency domain steps:
  - Transformation from spatial to frequency domain
  - Image processing in the frequency domain
  - Inverse transformation back to the Spatial domain

What do frequencies mean in an image?

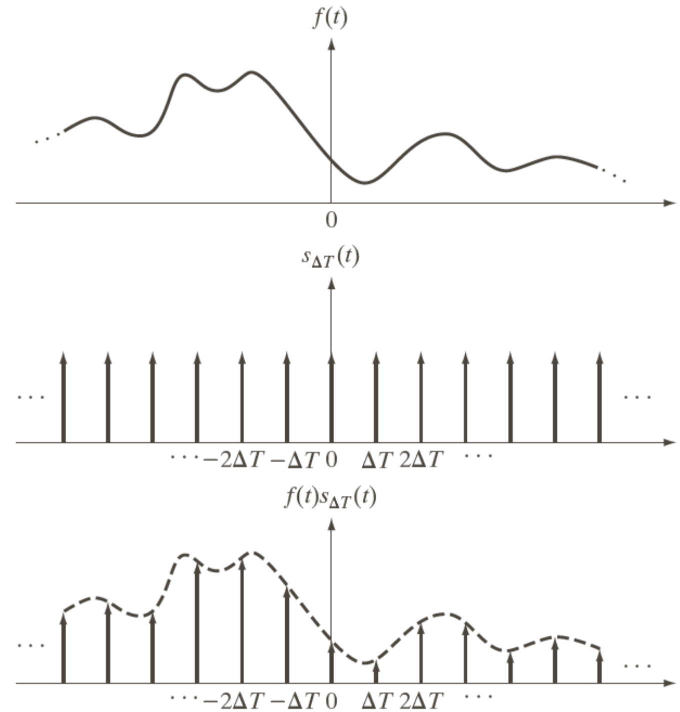
- Frequency in a digital image corresponds to the pace of variation of the pixels' gray level values
- Low frequencies indicate and correspond to slow varying pixel values
- High frequencies indicate high variation in the pixel values
- So frequency domain representation gives us a measure of pixels distribution in an image
- Fourier transform makes a connection between *spatial* and *frequency* domains

Discrete Fourier Transform of One Variable

Sampled Function

The Fourier Transform of a one variable function can be obtained using the relation:

$$F(u)=\int_{-\infty}^{\infty}\bar{f}(t)e^{-j2\pi ut}dt$$



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

Substituting…

$$F(u)=\int_{-\infty}^{\infty}\sum_{n=-\infty}^{\infty}f(t)\delta(t-n\Delta T)e^{-j2\pi ut}dt$$

$$=\sum_{n=-\infty}^{\infty}\int_{-\infty}^{\infty}f(t)\delta(t-\Delta T)e^{-j2\pi ut}dt$$

$$=\sum_{n=-\infty}^{\infty}f_n e^{-j2\pi un\Delta T}$$

The function *fn* is discrete, however its transform is continuous and infinitely periodic with period 1/delta T

Discrete Fourier Transform of One Variable

Sampled Function

Suppose we want to have M equally spaced samples of the transform taken over the period $u = 0$ to $u = \frac{1}{\Delta T}$

This can be done by: $u = \frac{m}{M\Delta T}$ $\quad m = 0,1,2,\ldots,M-1$

Substituting u …

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

Discrete Fourier Transform

- From DIP's perspective, the input is a two dimensional discrete signal (image)
- For a discrete one dimensional function f(x), with x=0, 2, ... M-1
- The DFT is:

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

- Or :

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)\left(\cos\frac{2\pi ux}{M} - j\sin\frac{2\pi ux}{M}\right)$$

- The **F(u)** is computed for **0 to M-1** frequencies
- Each **F(u)** represents a frequency component of the input image
- The inverse DFT is:

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

Discrete Fourier Transform

- The first value of F(u) starts from u=0 to u=M-1
- And for every value of 'u' all values of x which also range from 0 to M-1 are summed up and multiplied with the exponential term

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

- Or :

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)\left(\cos\frac{2\pi ux}{M} - j\sin\frac{2\pi ux}{M}\right)$$

- The **F(u)** is computed for **0 to M-1** frequencies

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

- At u=0, v=0, we get: $F(0,0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$ Average value

DFT- Translation

- The duration 0 to M-1 consists of two back to back half periods
- For display and filtering purposes it is often desired to have the complete period within this interval

$$f(x)e^{j2\pi(u_0x/M)} \Leftrightarrow F(u-u_0)$$

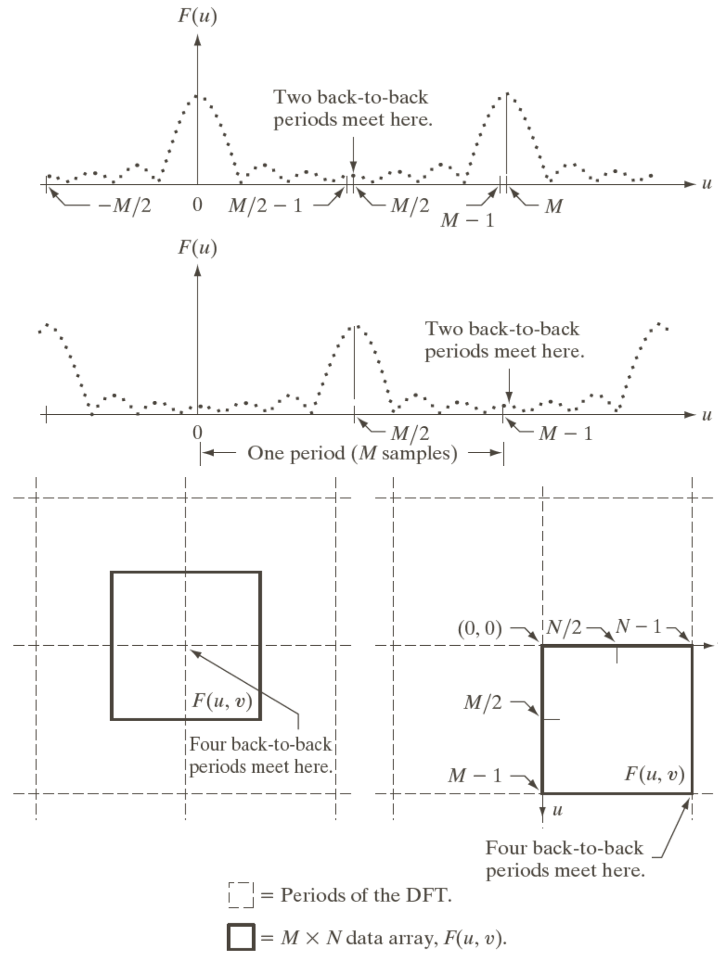
- In other words, multiplying f(x) by the exponential term shifts the transform so that the origin F(0), is located at

$$f(x,y)e^{j2\pi(u_0x/M+v_0x/M)} \Leftrightarrow F(u-u_0,v-v_0)$$

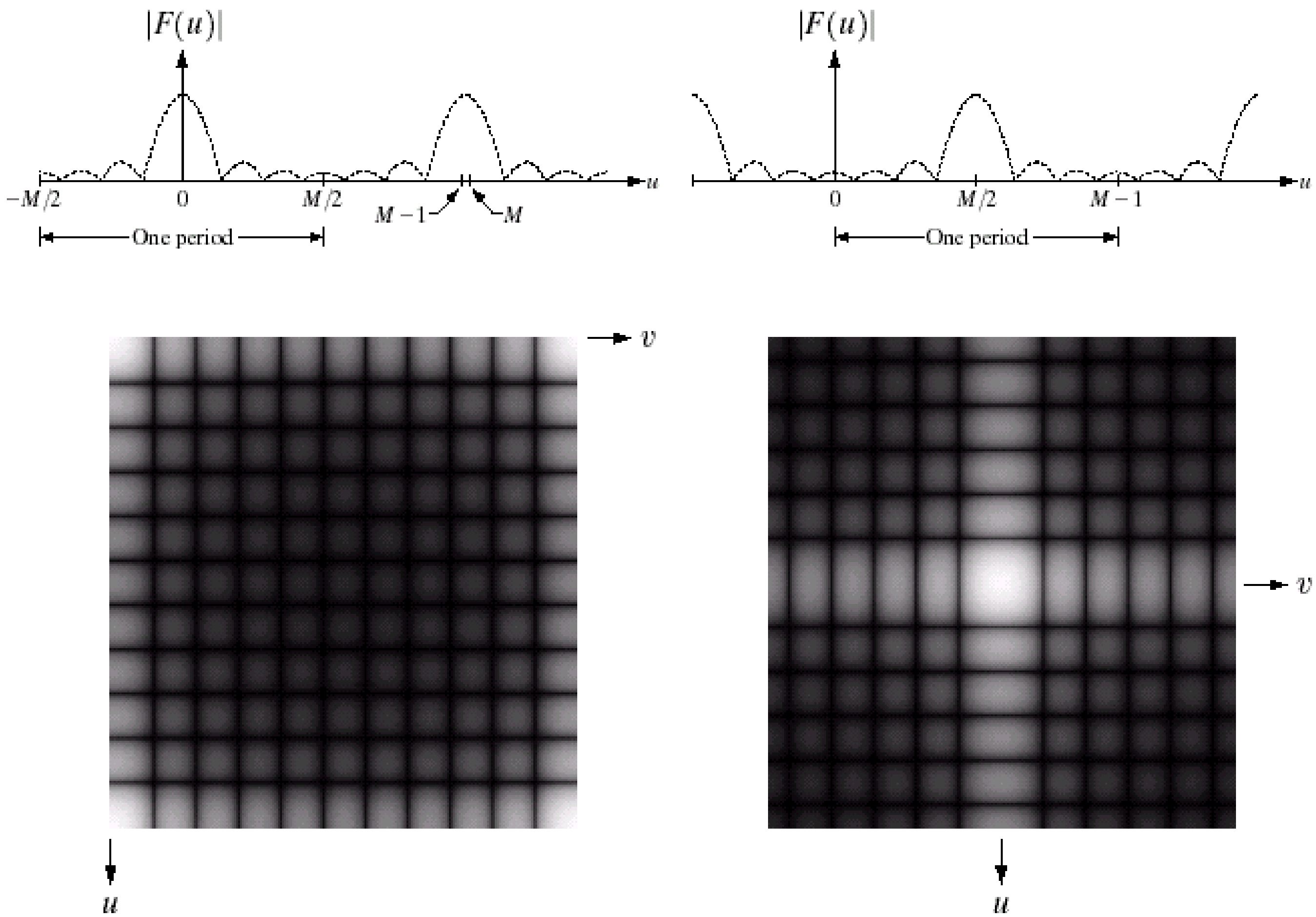
- If we let $u_0 = \dfrac{M}{2}$, then the exponential term becomes:

$$e^{j\pi x} \quad \text{or} \quad (-1)^x$$

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



PERIODICITY OF DFT



The Discrete Fourier Transform (DFT)

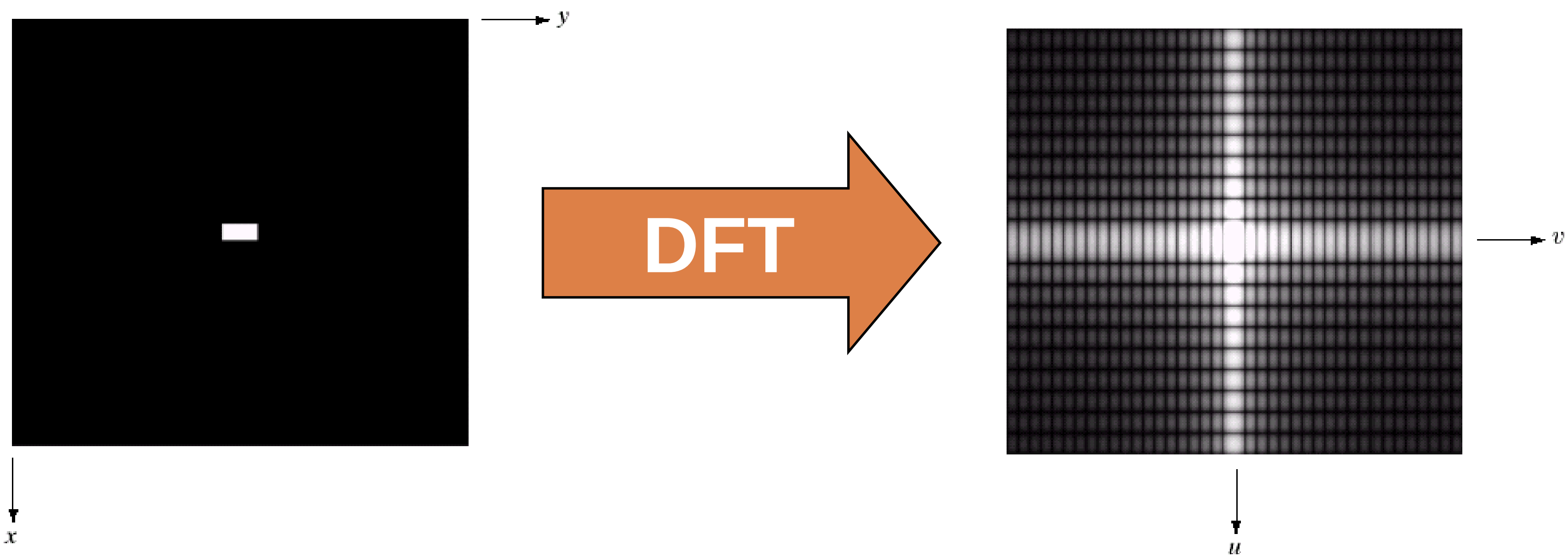
The *Discrete Fourier Transform* of $f(x, y)$, for $x = 0, 1, 2 \ldots M-1$ and $y = 0,1,2 \ldots N-1$, denoted by $F(u, v)$, is given by the equation:

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

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

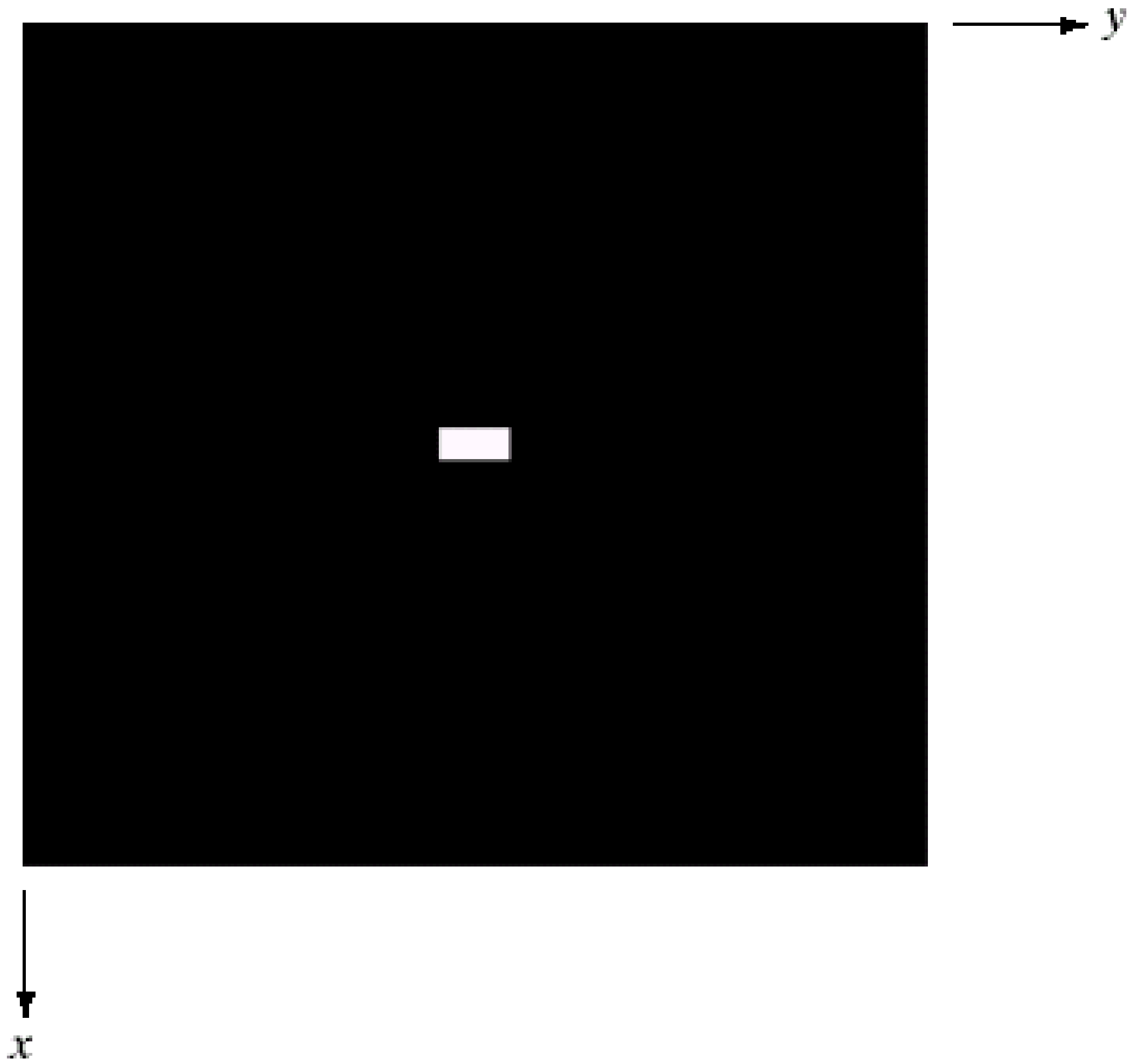
DFT & Images

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

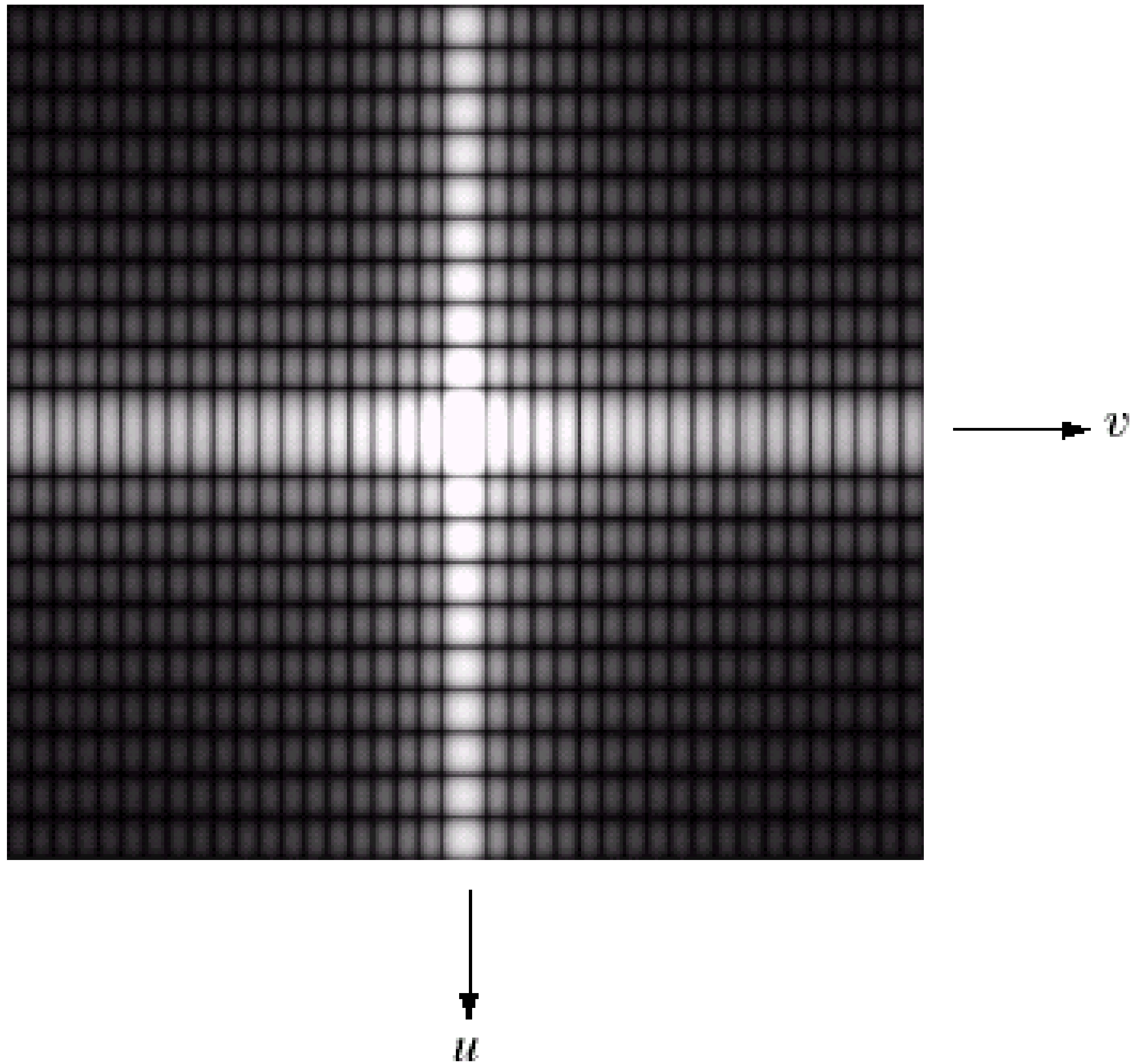


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

DFT & Images



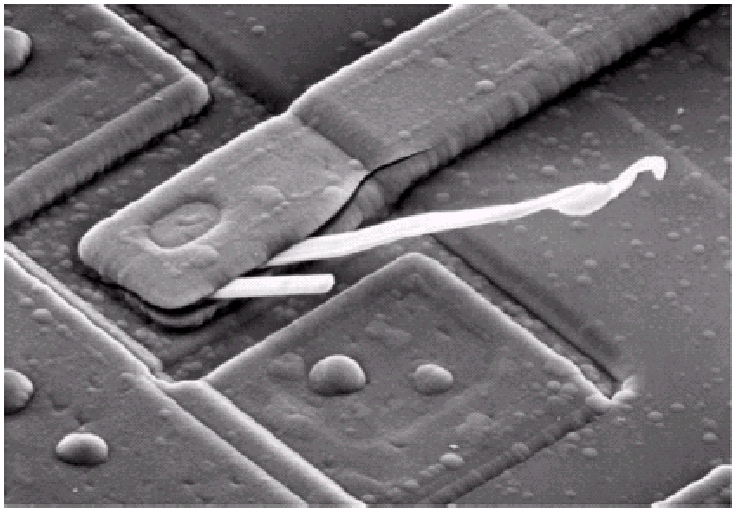
DFT & Images



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

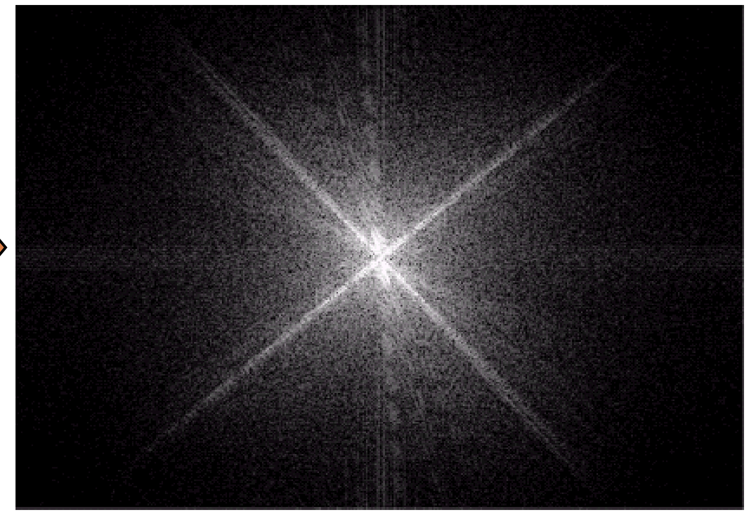
DFT & Images (cont…)

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



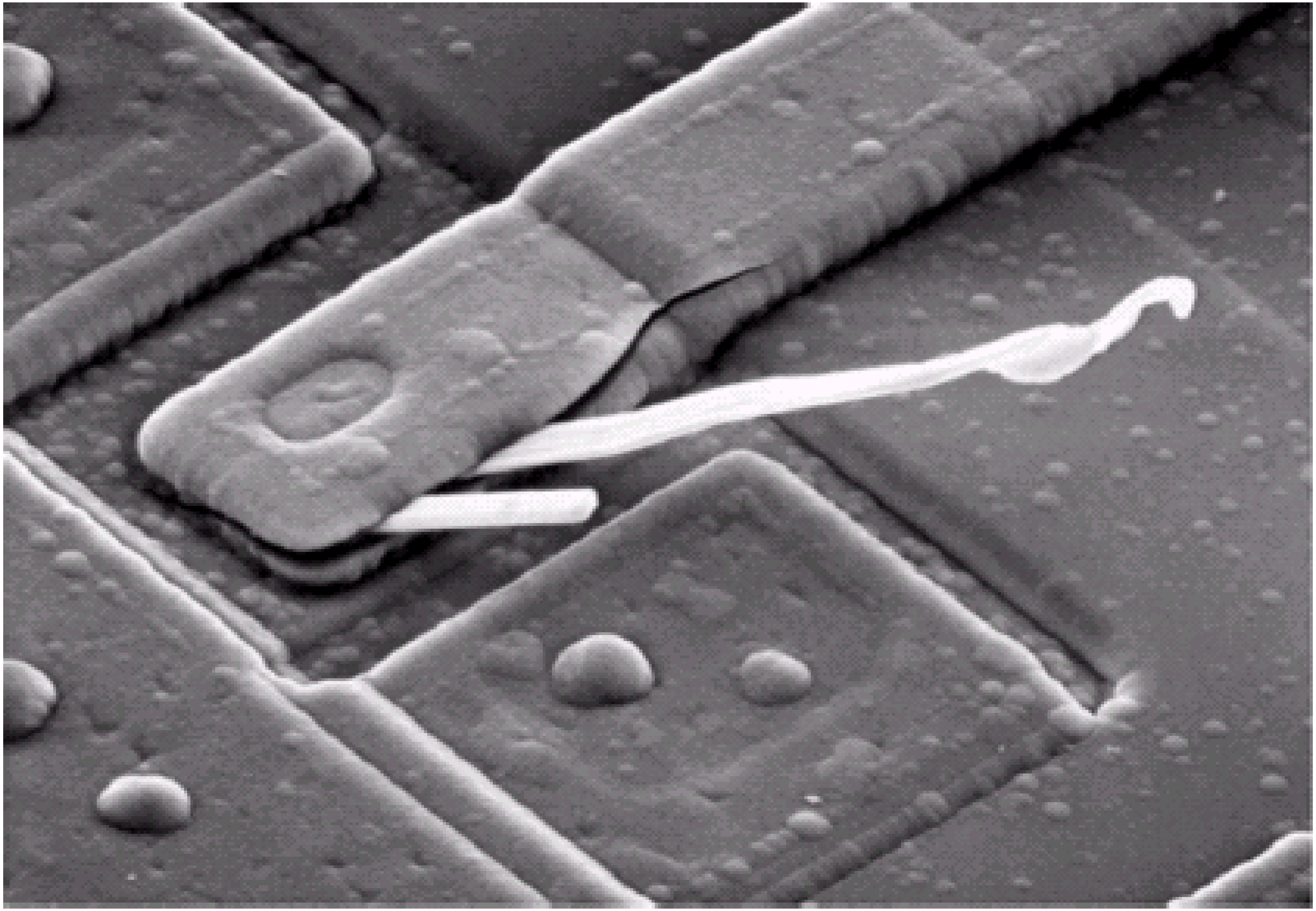
Scanning electron microscope image of an integrated circuit magnified ~2500 times

DFT



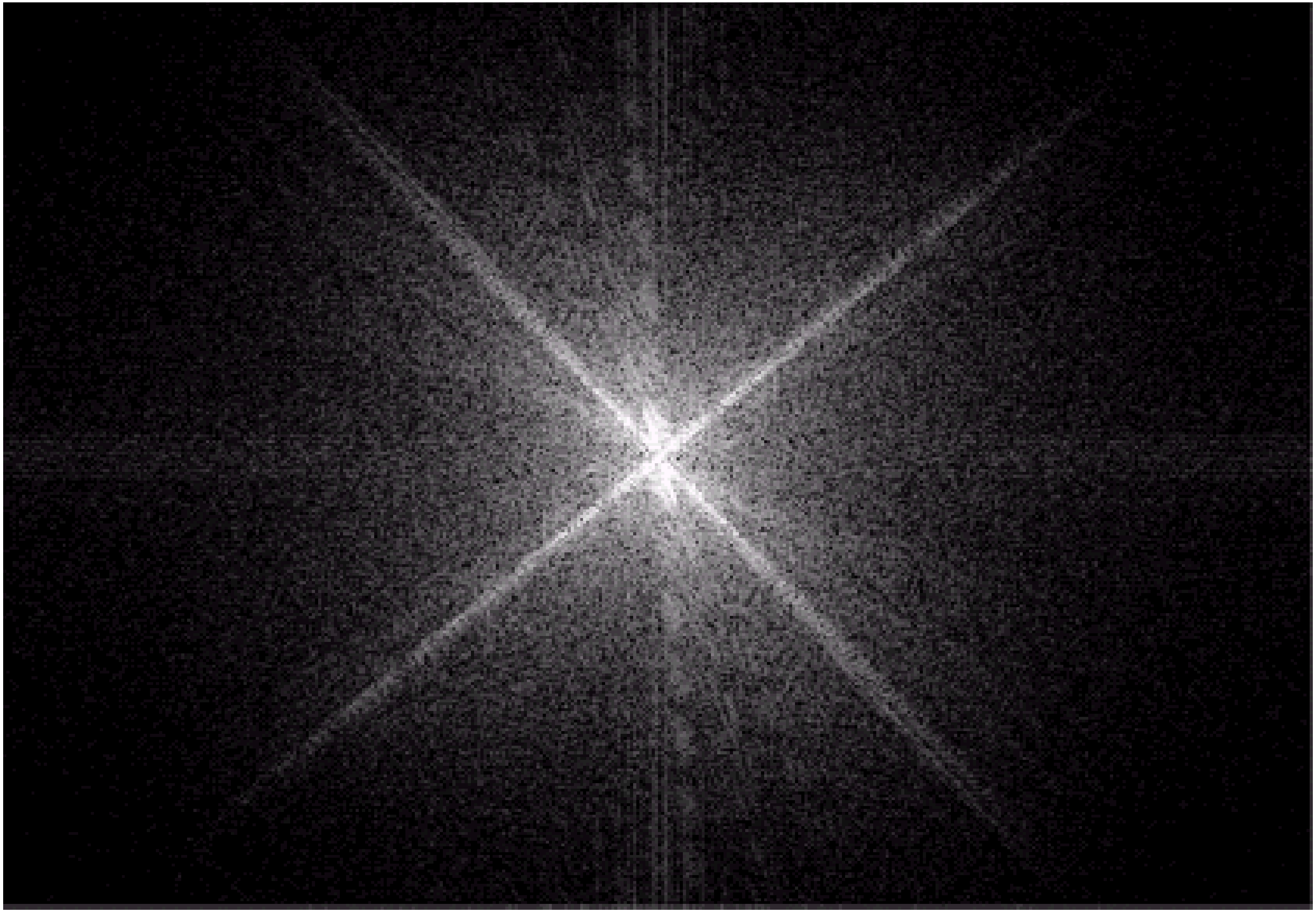
Fourier spectrum of the image

DFT & Images (cont…)



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

DFT & Images (cont…)



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

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

Filtering in the Frequency domain

- Apart from trivial cases, establishing a relation between spatial domain and frequency domain characteristics of images is difficult
- When u=v=0, this corresponds to average value
- Moving away from this point, the low frequencies correspond to slowing varying components in an image
- The higher frequencies correspond to faster gray level changes
- Such relationships (although gross) can help establishing enhancement techniques in the frequency domain

Basics of Filtering in the Frequency domain

Filtering in FD consists modifying the Fourier transform of an image with a filter function and then computing the inverse transform to get the output

With an image of size ***MxN***:

$$g(x, y) = \mathfrak{F}^{-1}\left[H(u,v)F(u,v)\right]$$

The functions F, H and g are arrays of size MxN

PROPERTIES OF DFT

CHOICE OF DFT INTERVAL

- If the interval lengths of $f(x)$ and $h(x)$ are M and N respectively interval length for $f(x)*h(x)$ will be M+N-1

Effect on frequency domain filtering!

The 2-D Convolution Theorem

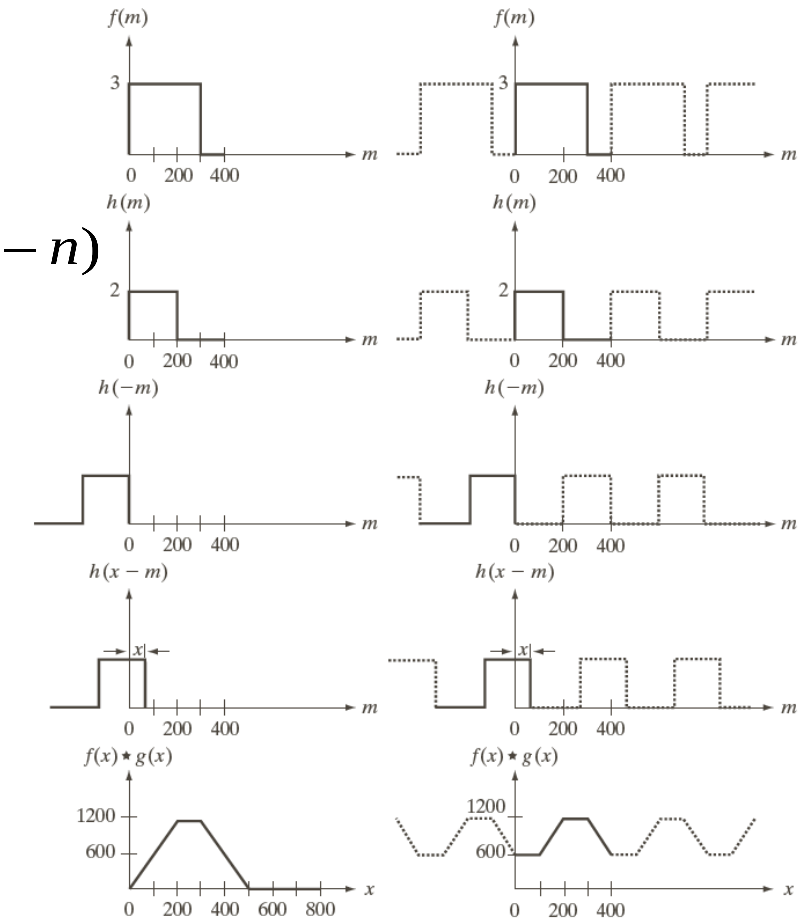
The 2-D convolution theorem is given by the expressions:

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

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

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

$$f(x) * h(x) = \sum_{m=0}^{399} f(x)h(x-m)$$



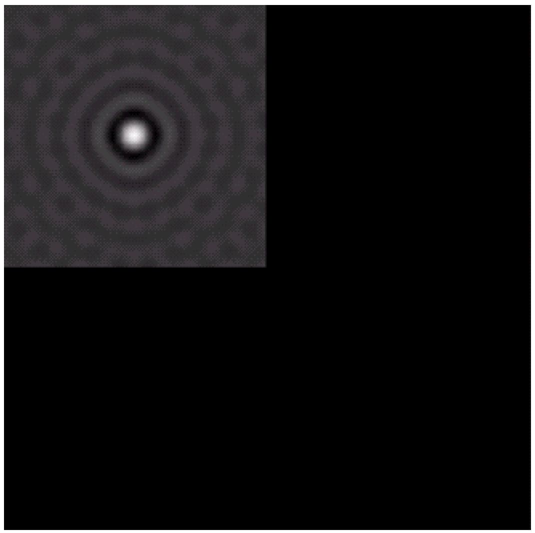
Taking into account the periodicity implied by the DFT yields incorrect convolution result

The solution is to use zero padding

If f(x) and h(x) consists of A and B samples, the padding should be:

P $\geq$ A+B-1

CHOICE OF INTERVAL SIZE FOR CONVOLUTION: PADDING EXAMPLE



Basics of Filtering in the Frequency domain

Filtering in FD consists of the following steps:

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

Basics of Filtering in the Frequency domain

Steps of filtering in FD

a	b	c
d	e	f
g	h	

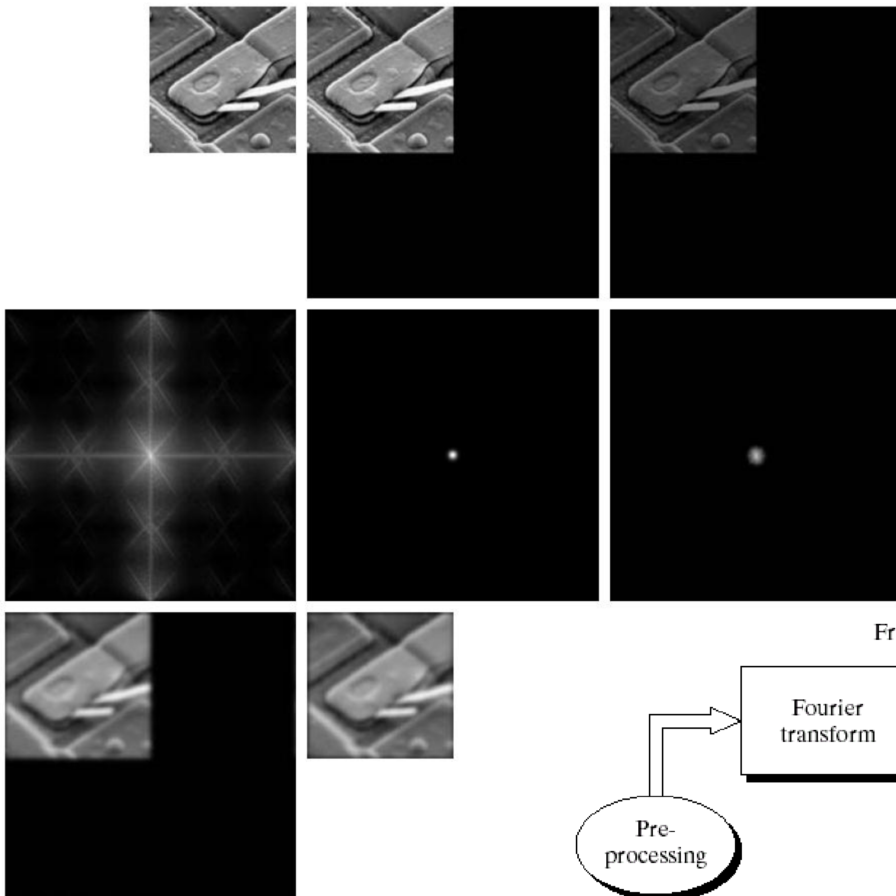
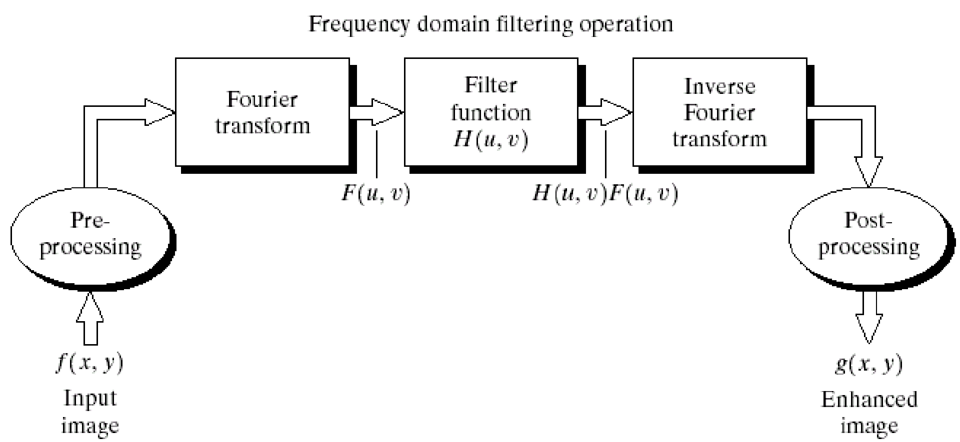


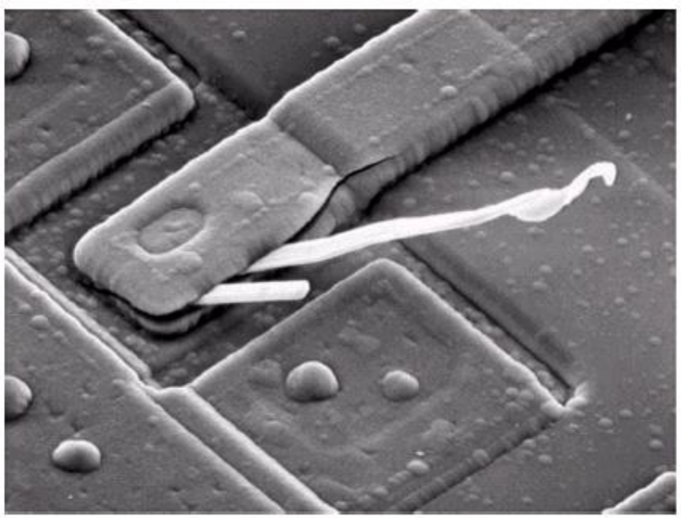
FIGURE 4.36
(a) An $M \times N$ image, f.
(b) Padded image, f_p of size $P \times Q$.
(c) Result of multiplying f_p by $(-1)^{x+y}$.
(d) Spectrum of F_p. (e) Centered Gaussian lowpass filter, H, of size $P \times Q$.
(f) Spectrum of the product HF_p.
(g) g_p, the product of $(-1)^{x+y}$ and the real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

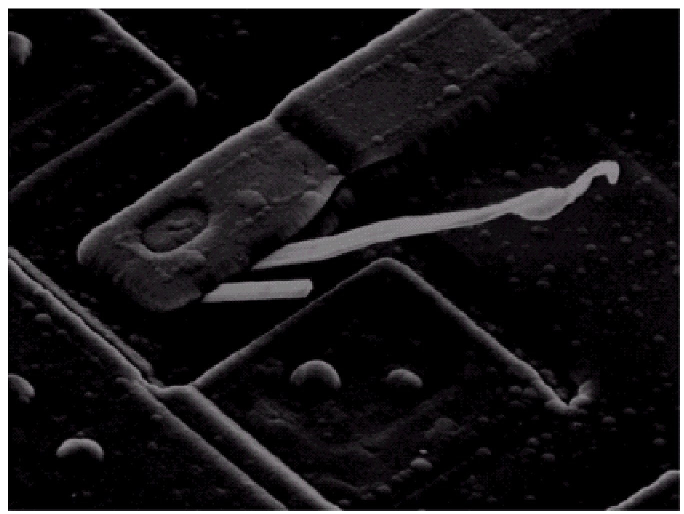


Some basic filters and their properties

- A filter function that forces the average value of the Transform to zero:

$$H(u,v) = \begin{cases} 0 & if\ (u,v) = (M/2, N/2) \\ 1 & otherwise \end{cases}$$



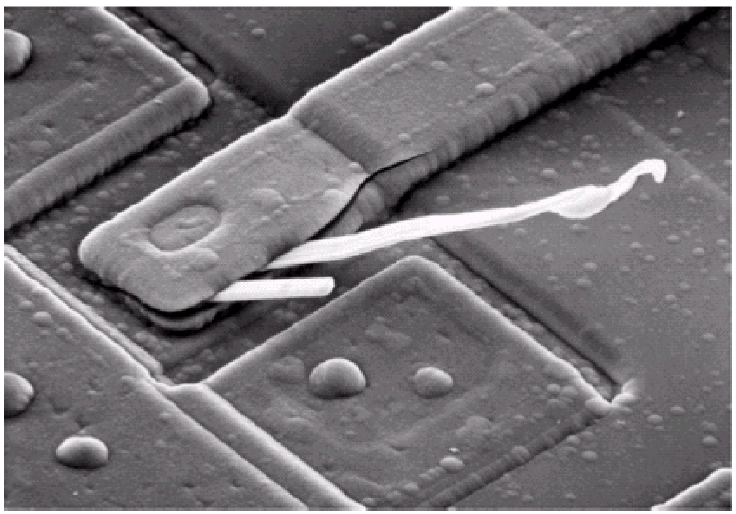


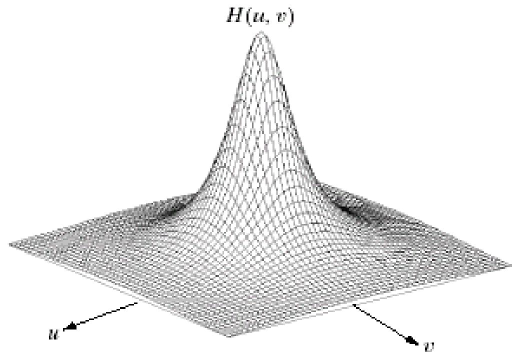
A zero average value means negative and positive values – the displayed image is scaled

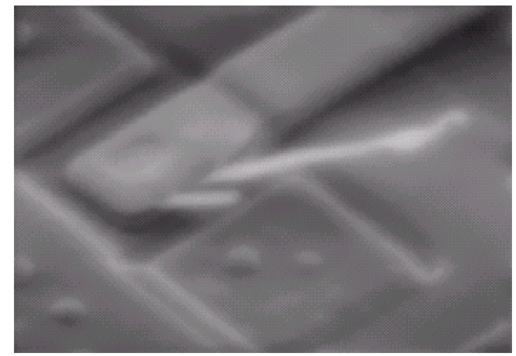
Some Basic Frequency Domain Filters

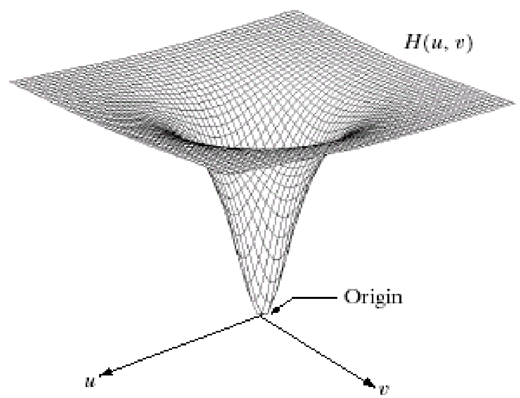
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

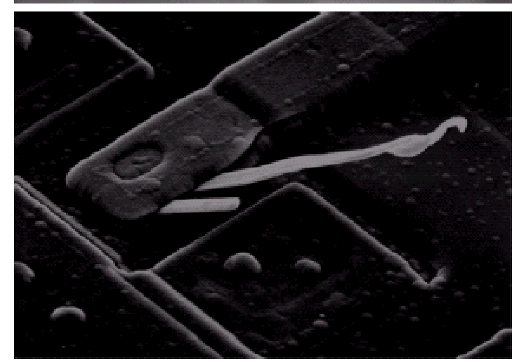
Low Pass Filter





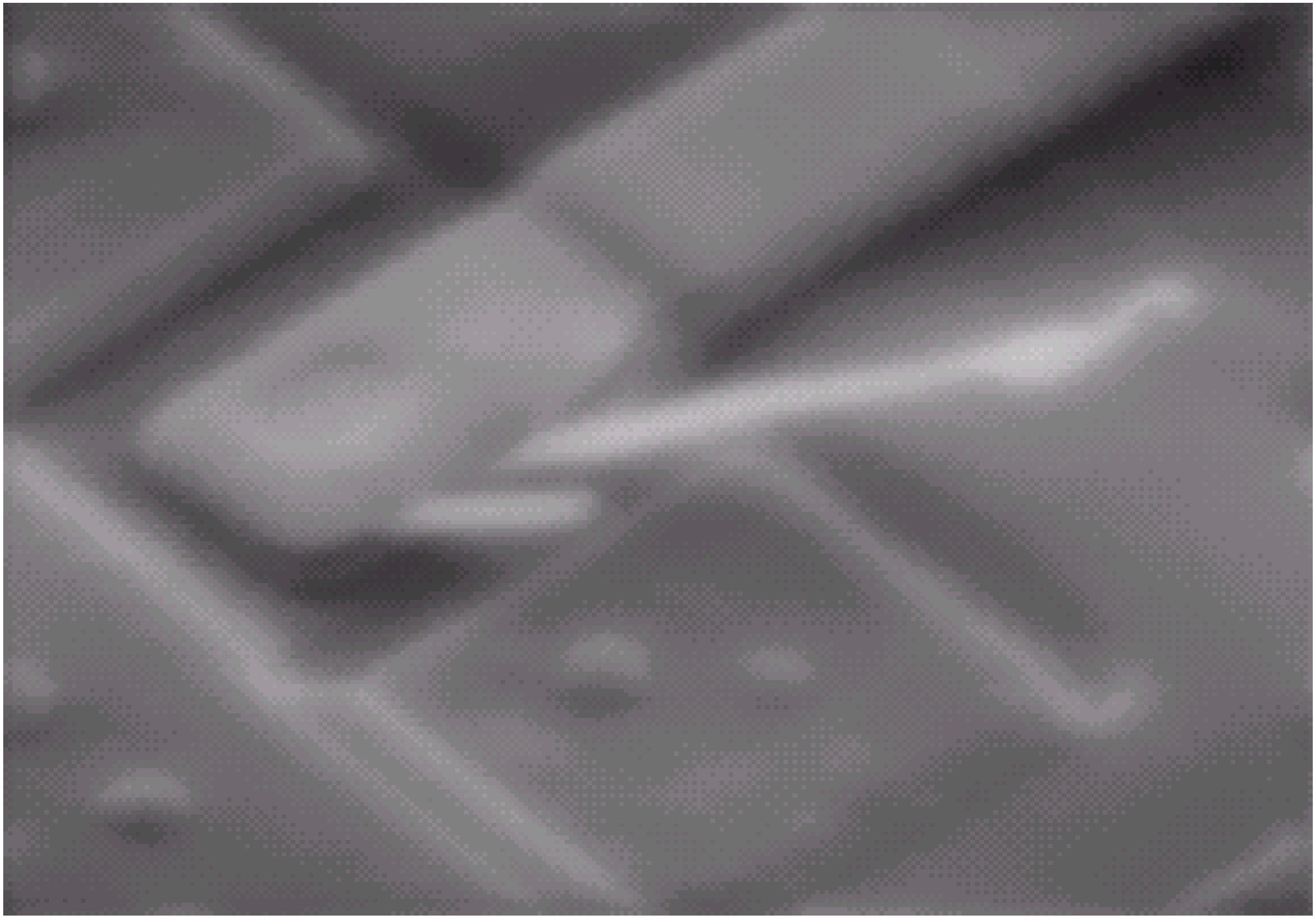






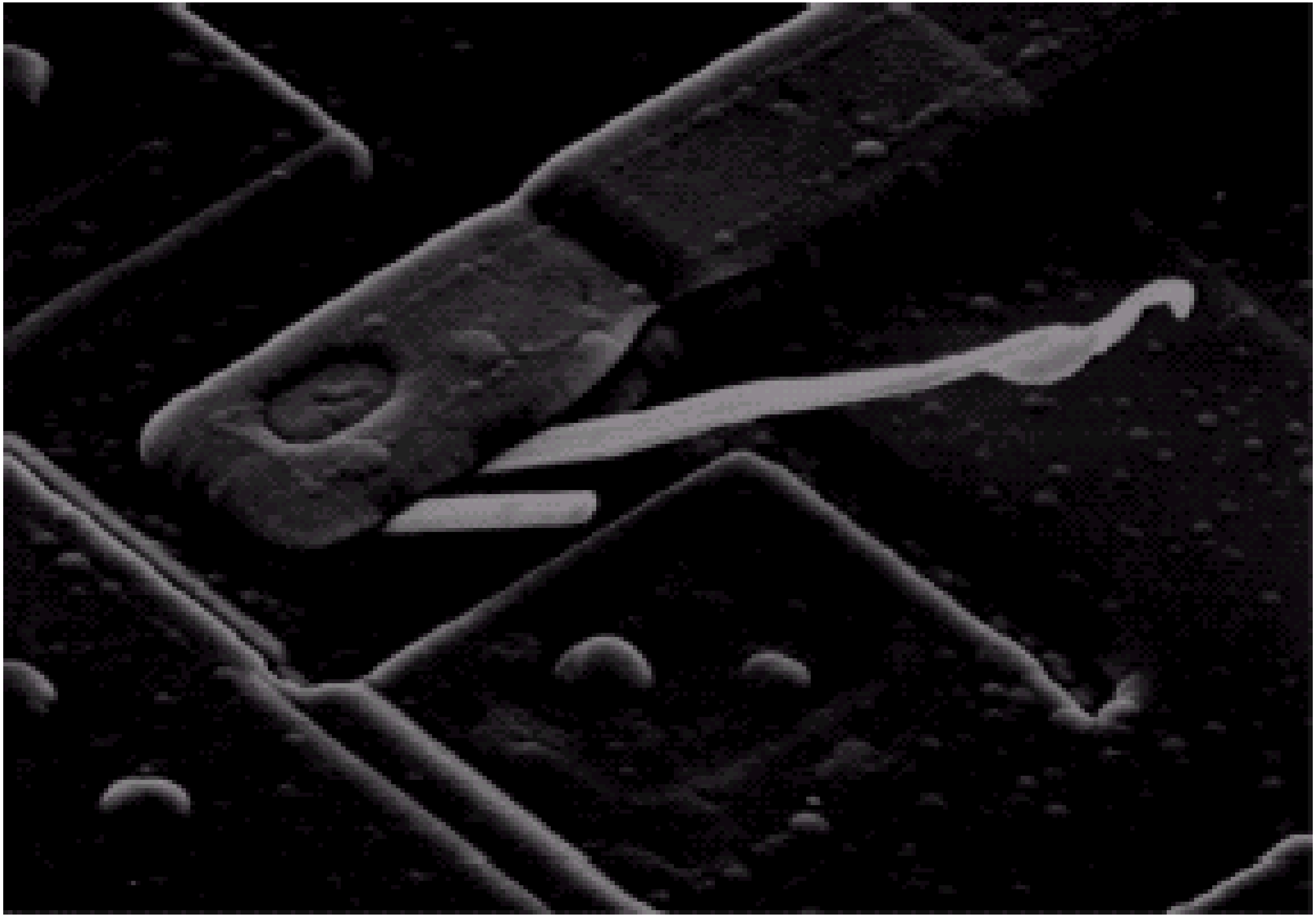
High Pass Filter

Some Basic Frequency Domain Filters



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

Some Basic Frequency Domain Filters



Smoothing Frequency Domain Filters

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

The basic model for filtering is:

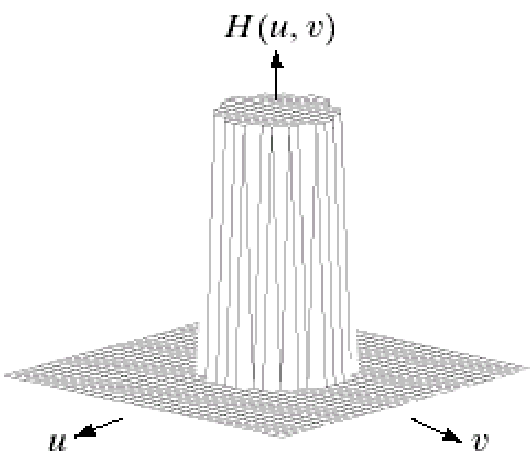
$$G(u,v) = H(u,v)F(u,v)$$

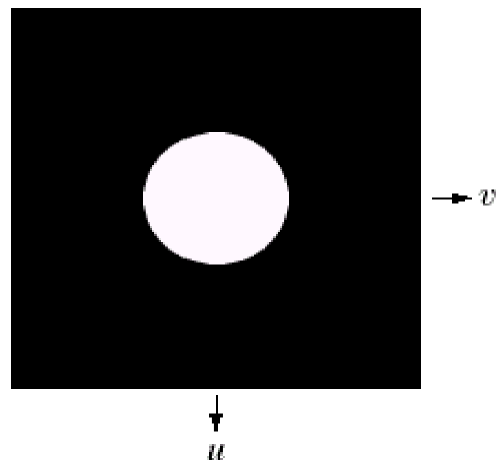
where $F(u,v)$ is the Fourier transform of the image being filtered and $H(u,v)$ is the filter transform function

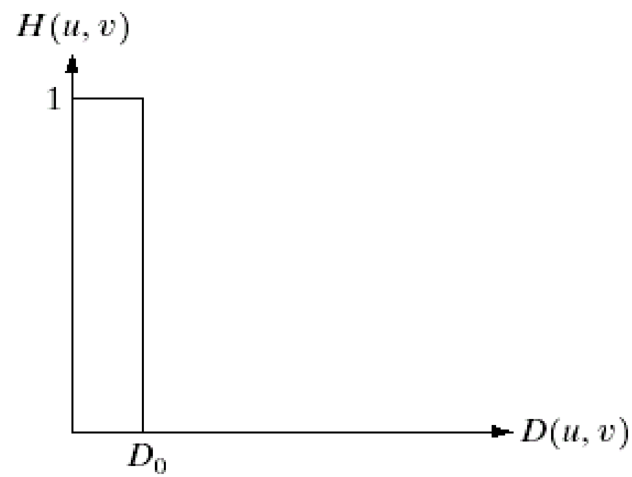
Low pass filters – only pass the low frequencies, drop the high ones

Ideal Low Pass Filter

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







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

Ideal Low Pass Filter (cont…)

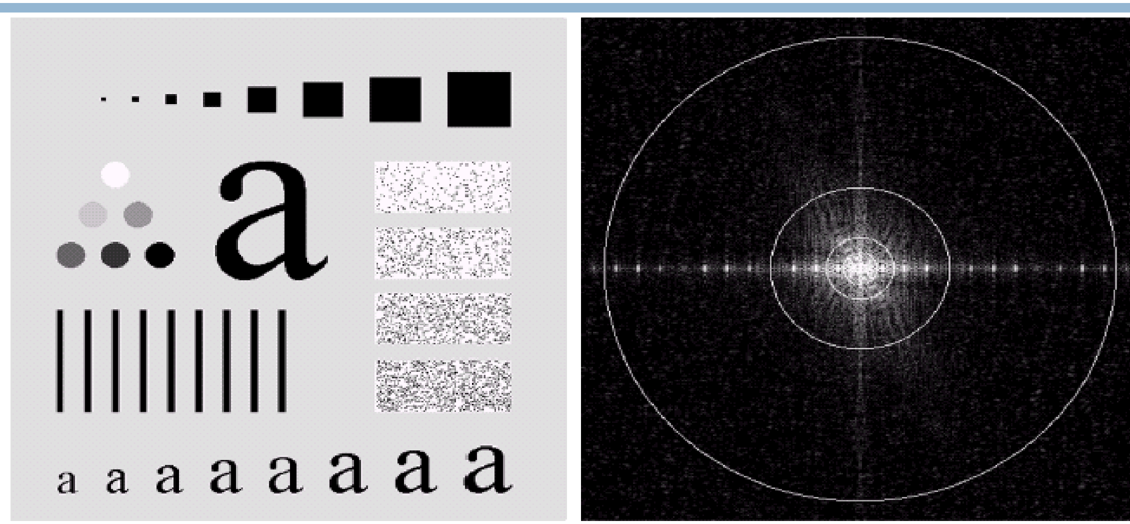
The transfer function for the ideal low pass filter can be given as:

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

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

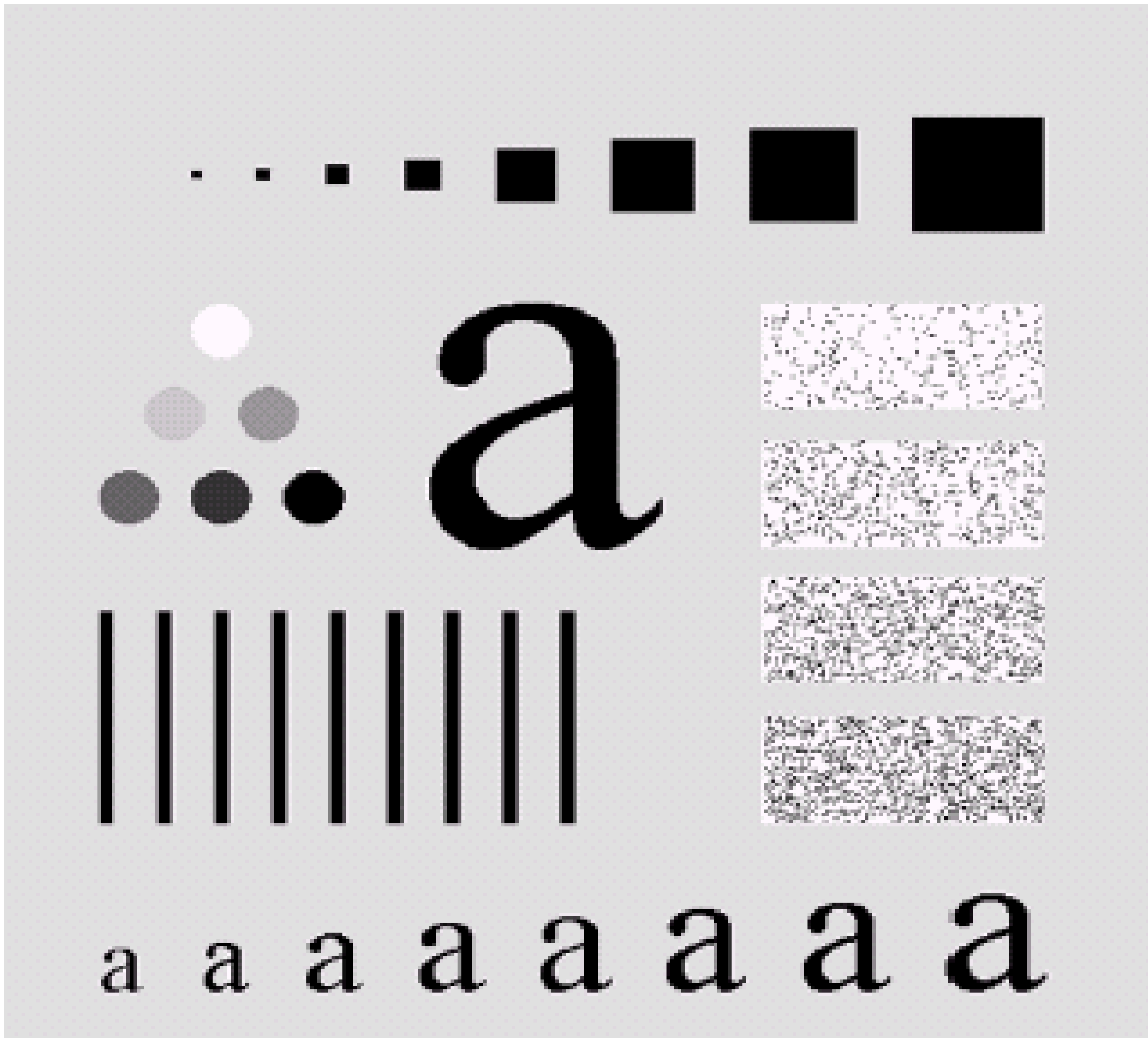
$$D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2}$$

Ideal Low Pass Filter (cont…)



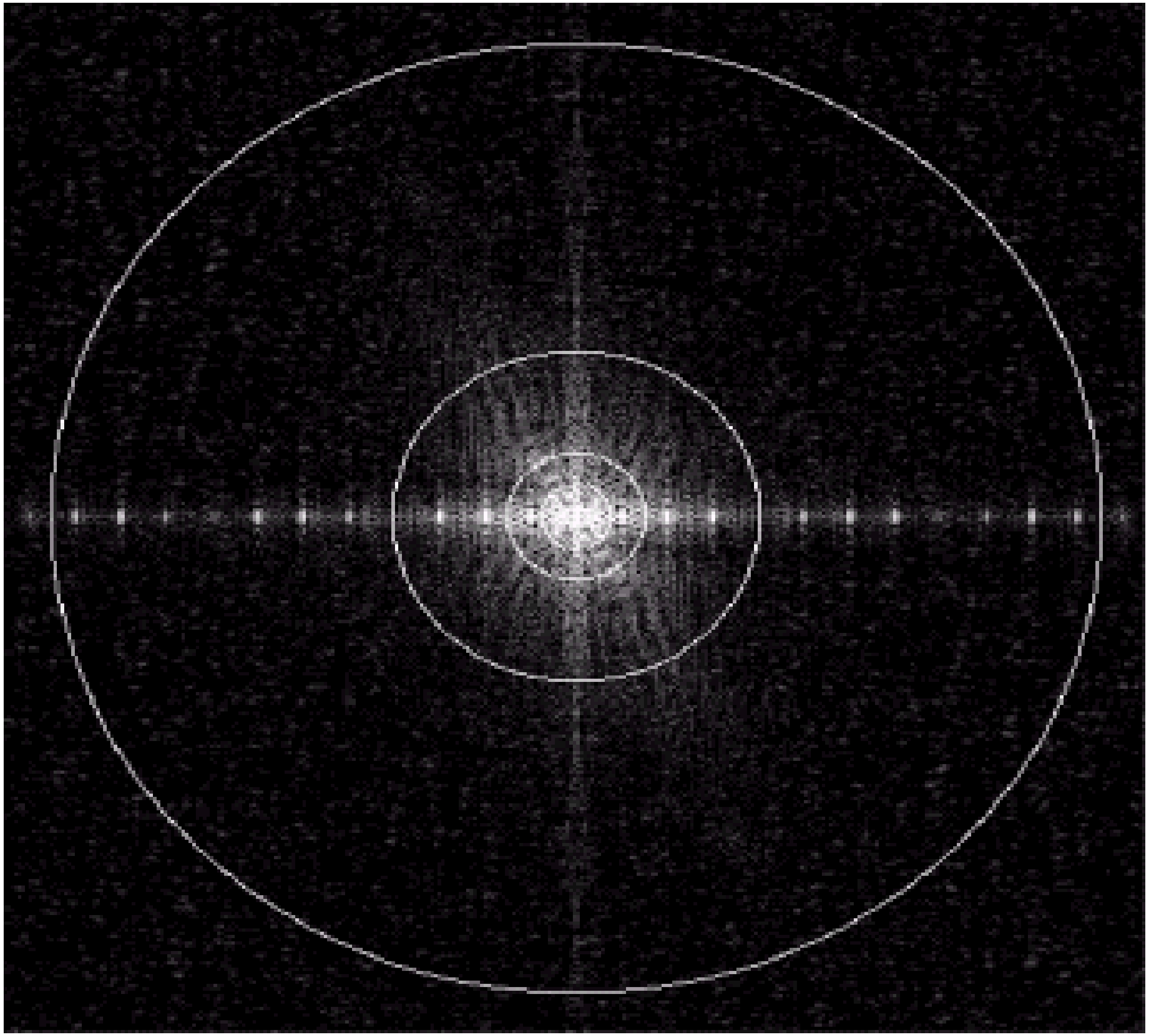
Above we show an image, it's Fourier spectrum and a series of ideal low pass filters of radius 5, 15, 30, 80 and 230 superimposed on top of it

Ideal Low Pass Filter (cont…)



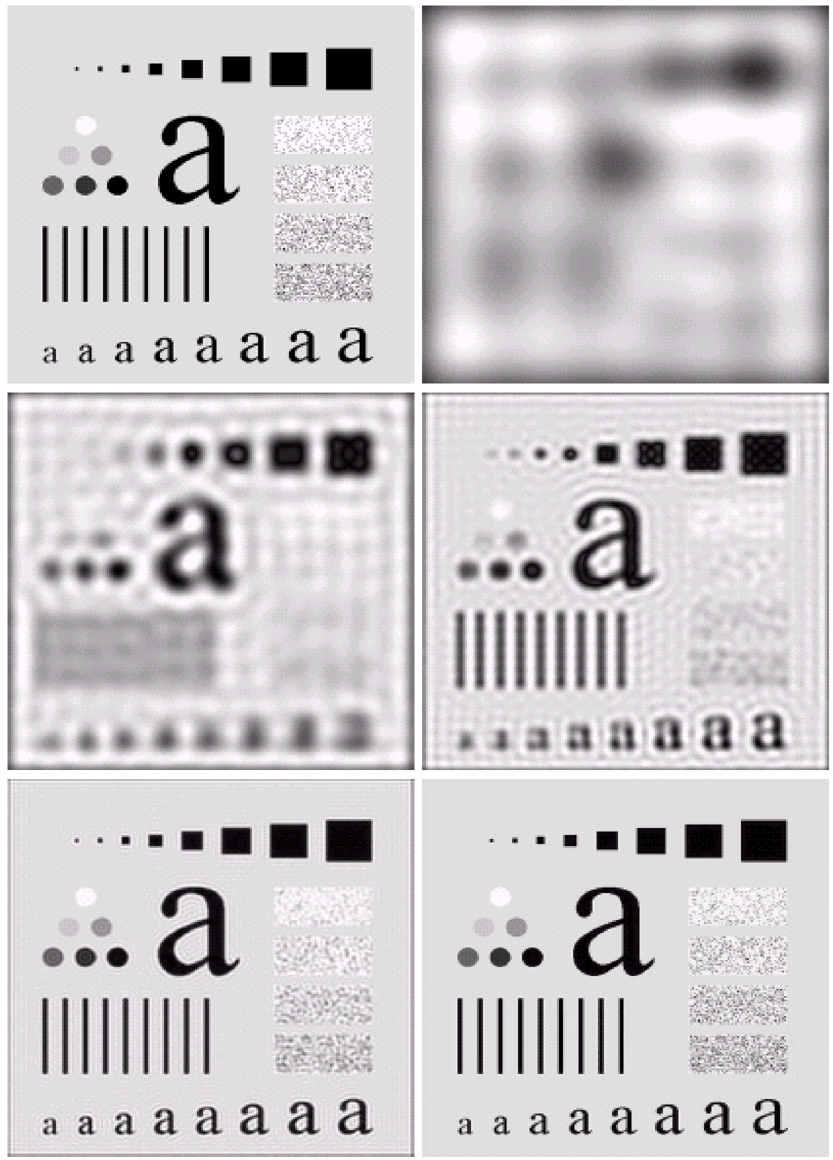
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Ideal Low Pass Filter (cont…)



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

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

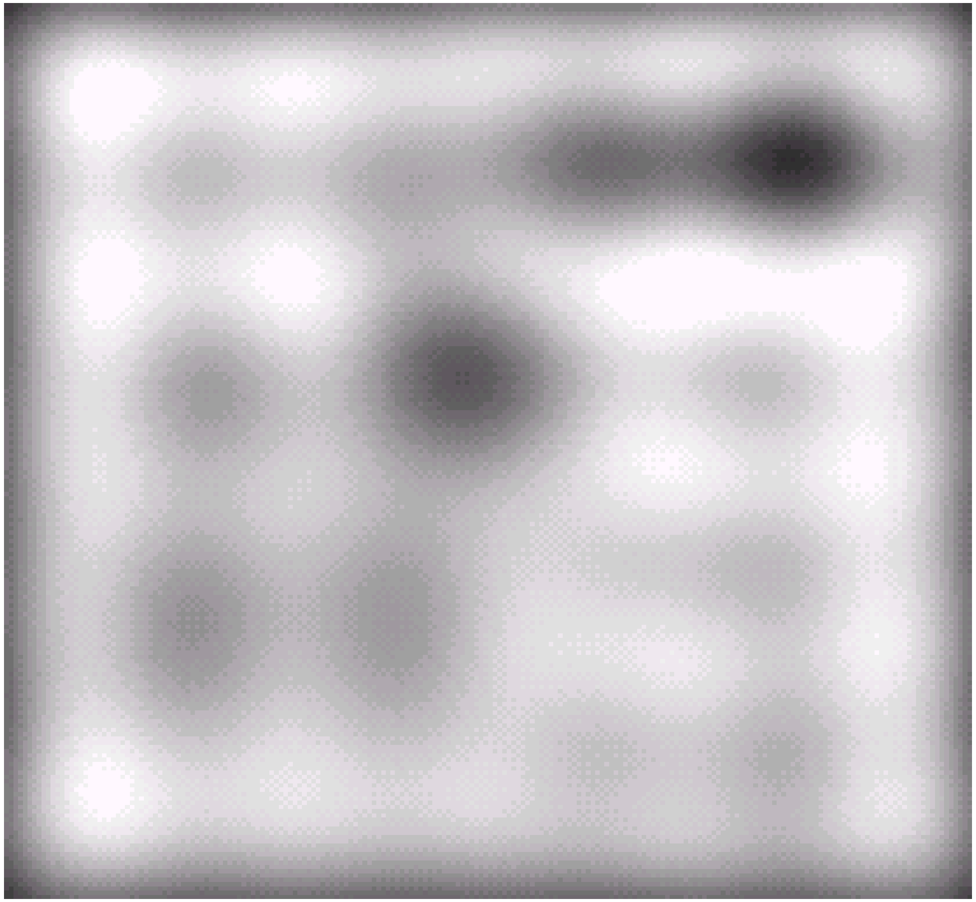
Result of filtering with ideal low pass filter of radius 30

Result of filtering with ideal low pass filter of radius 80

Result of filtering with ideal low pass filter of radius 230

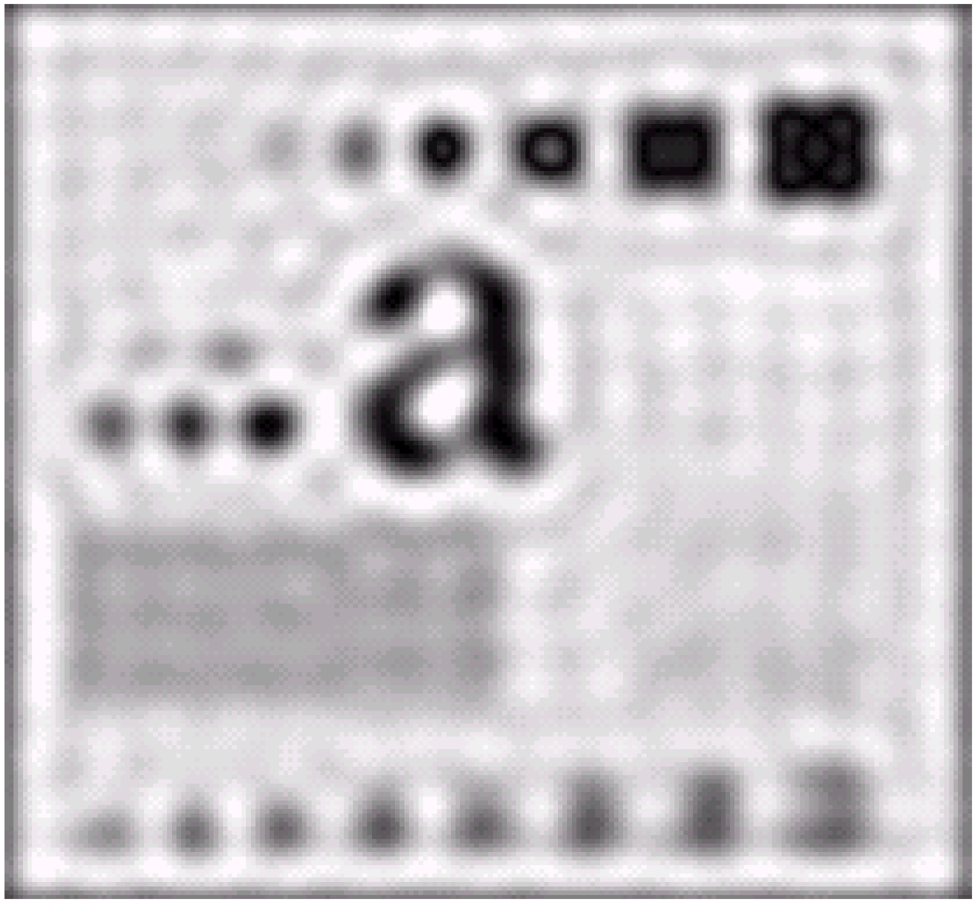
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Ideal Low Pass Filter (cont…)



Result of filtering with ideal low pass filter of radius 5

Ideal Low Pass Filter (cont…)



Result of filtering with ideal low pass filter of radius 15

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

Ideal Low pass filters – blurring & ringing

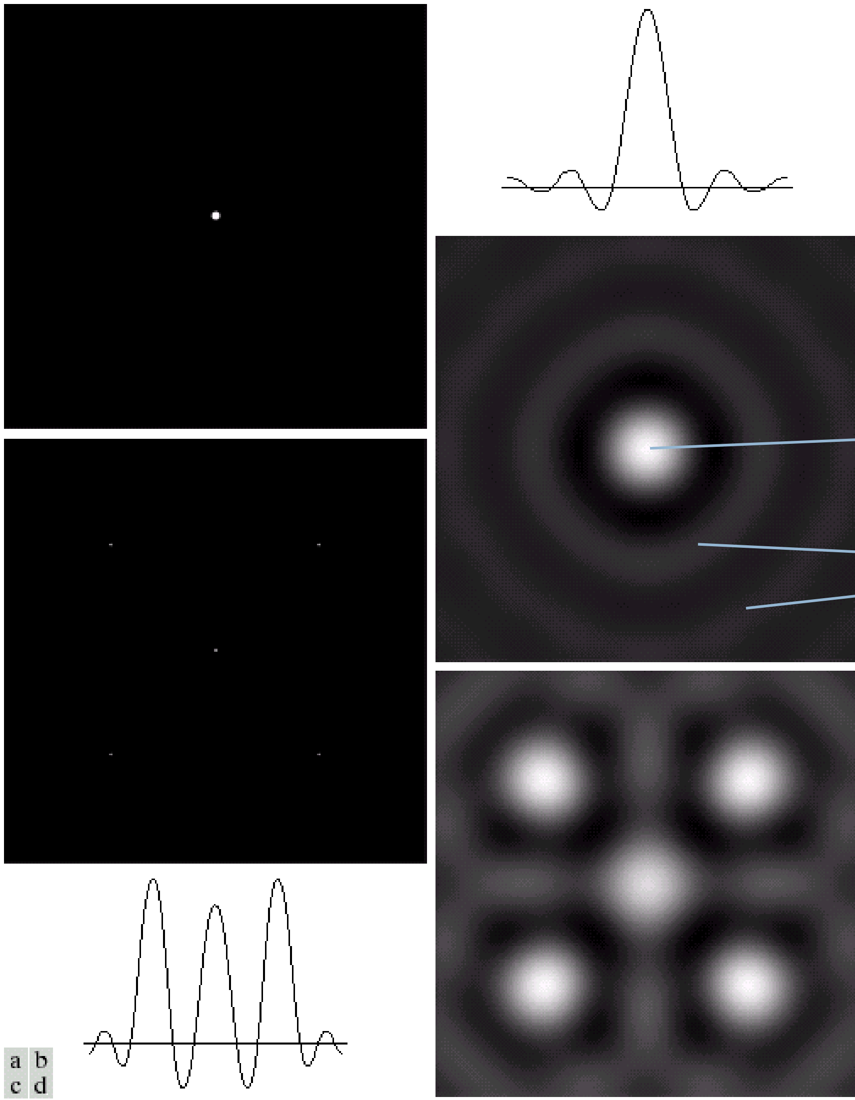


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

Because of the reciprocity of H(u,v) with h(x,y) – the blurring and ringing is more severe for narrower filters in the frequency domain

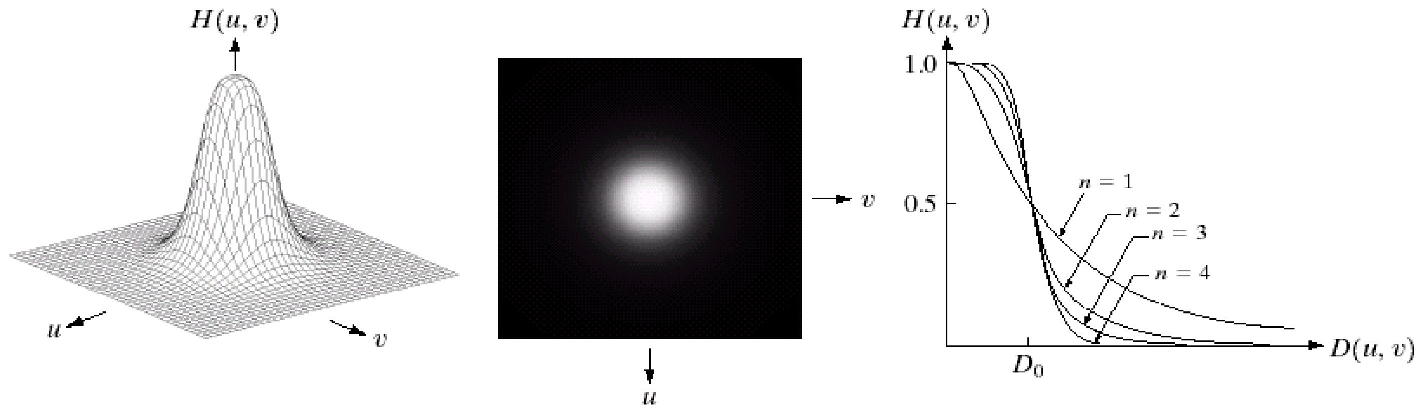
Causing blurring

Causing ringing

Butterworth Lowpass Filters

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

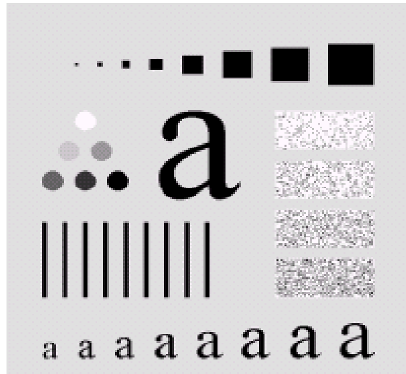
$$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}}$$

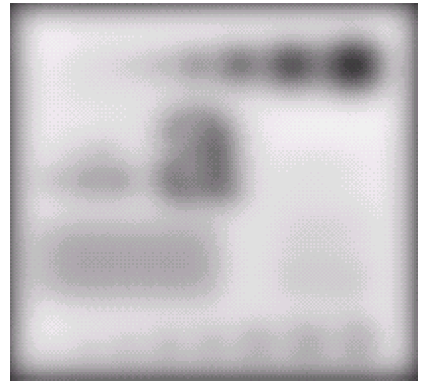


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Butterworth Lowpass Filter (cont…)

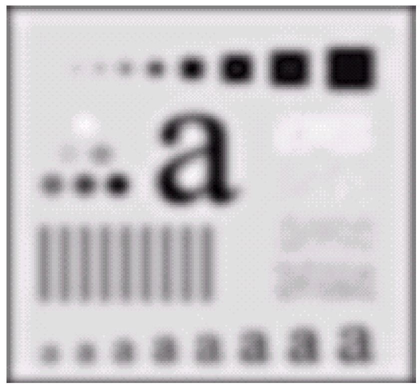
Original image

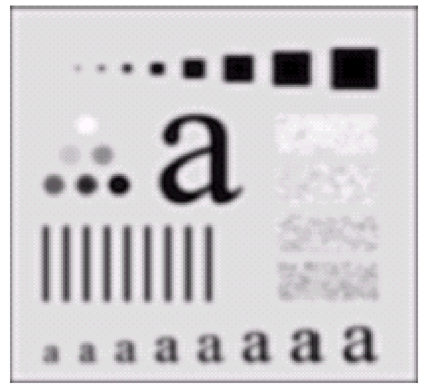




Result of filtering with Butterworth filter of order 2 and cutoff radius 5

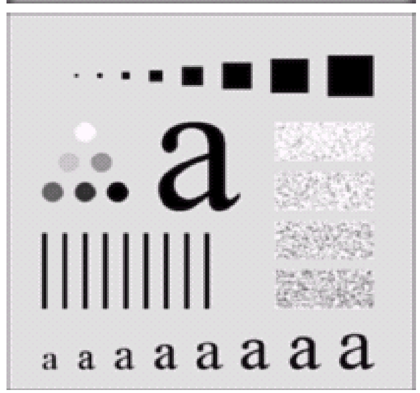
Result of filtering with Butterworth filter of order 2 and cutoff radius 15

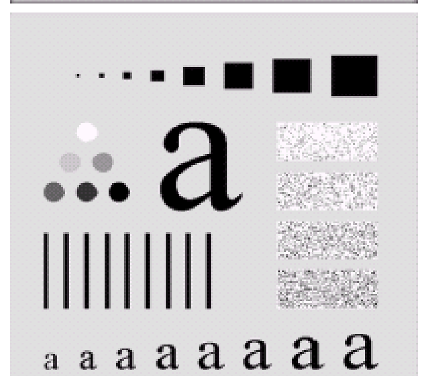




Result of filtering with Butterworth filter of order 2 and cutoff radius 30

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

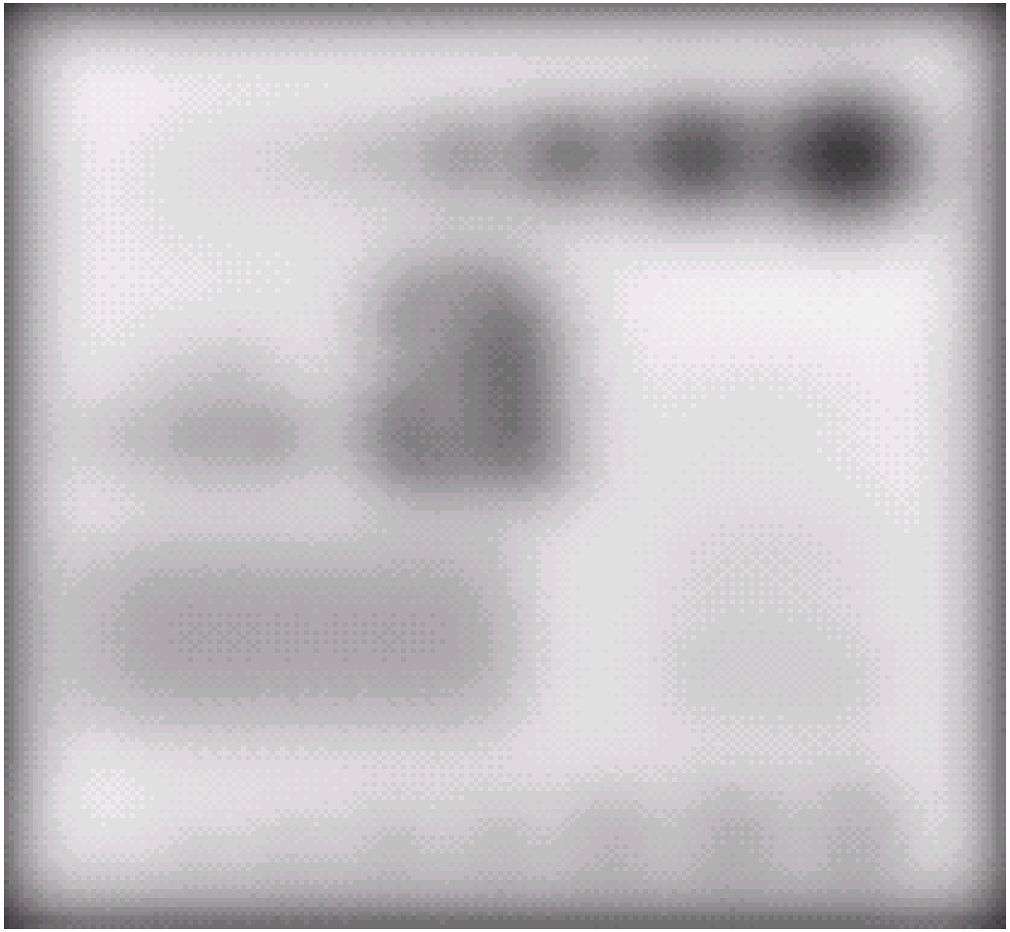




Result of filtering with Butterworth filter of order 2 and cutoff radius 230

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

Butterworth Lowpass Filter (cont…)

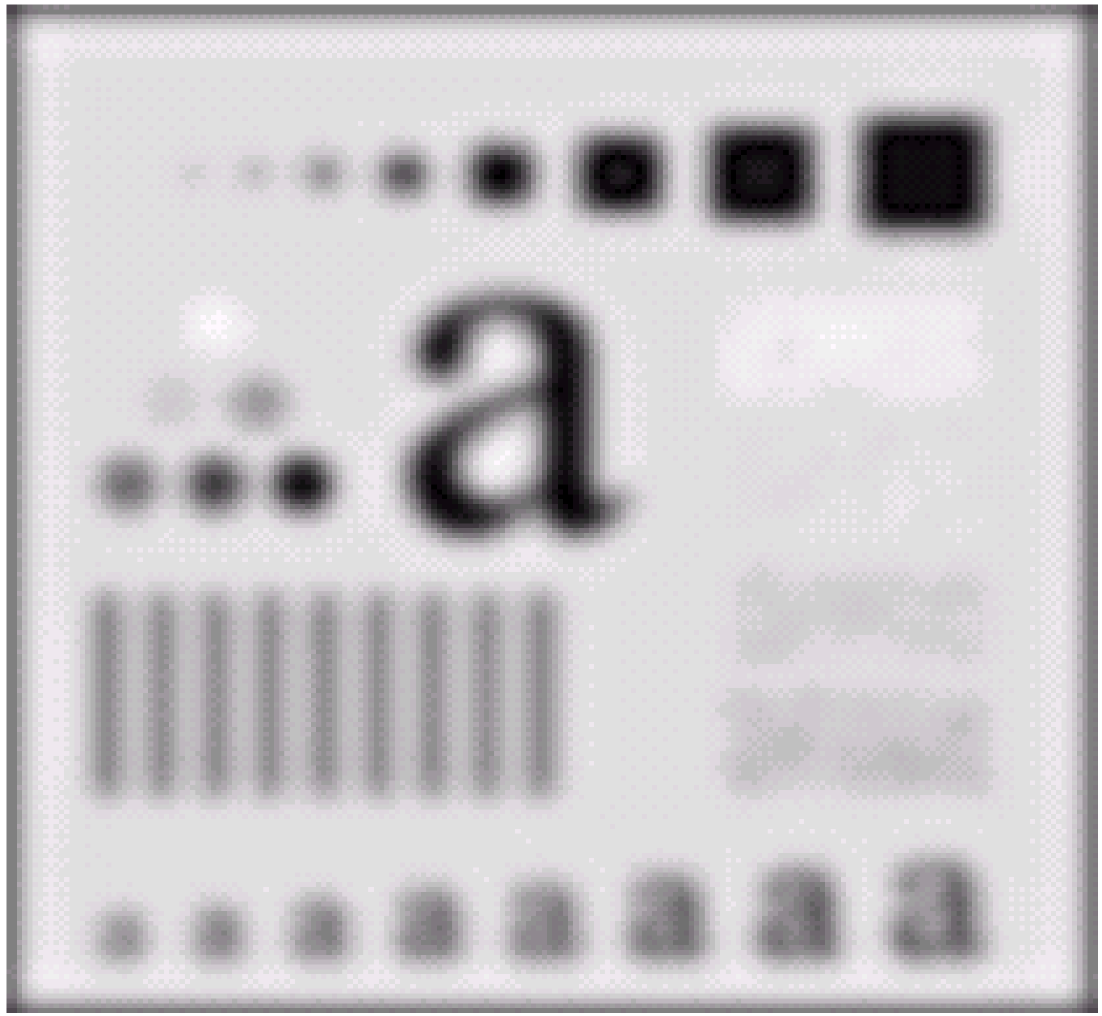


Result of filtering with Butterworth filter of order 2 and cutoff radius 5

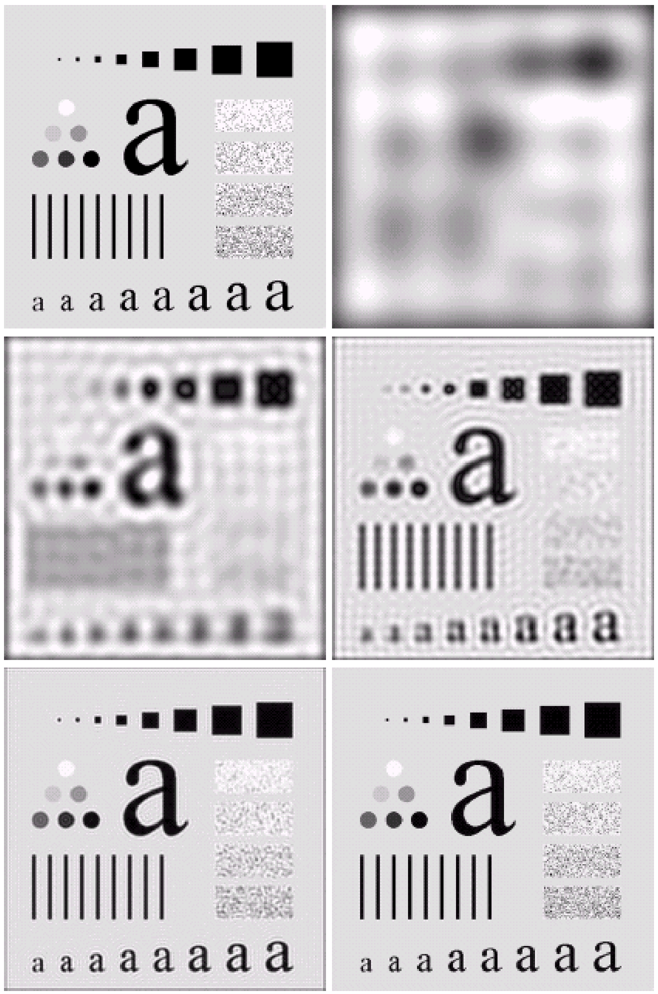
Butterworth Lowpass Filter (cont…)

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

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



Comparison of Ideal Lowpass and Butterworth Lowpass filters



ILPF

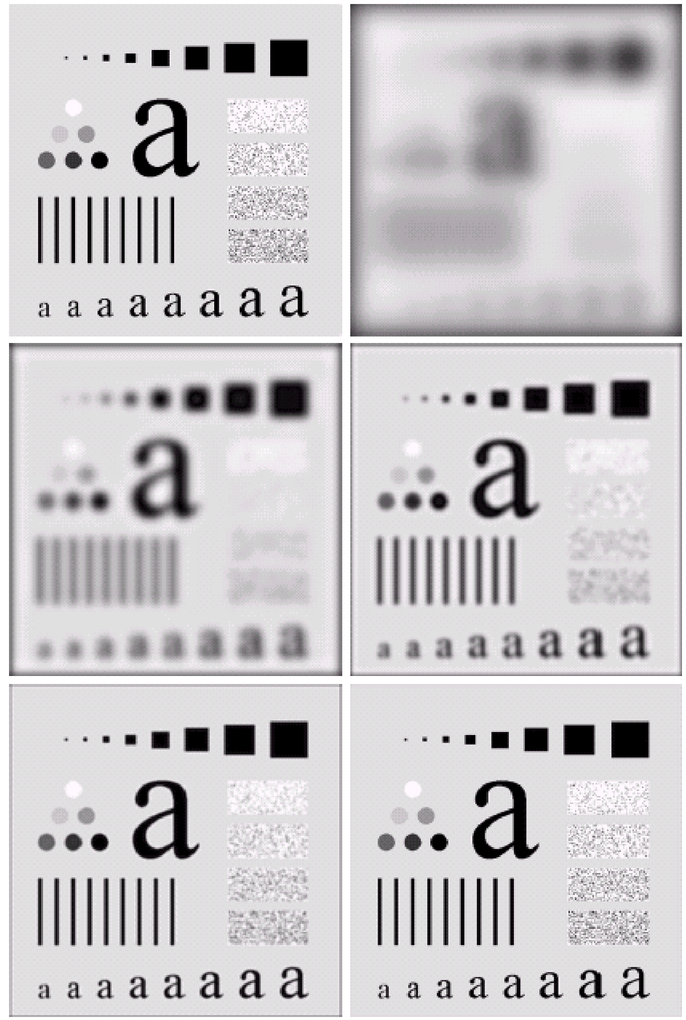


FIGURE 4.15 (a) Original image. (b)–(f) Results of filtering with BLPFs of order 2, with cutoff frequencies at radii of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Fig. 4.12.

Butterworth filter of n=2 and radii of 5, 15, 30, 80 and 230

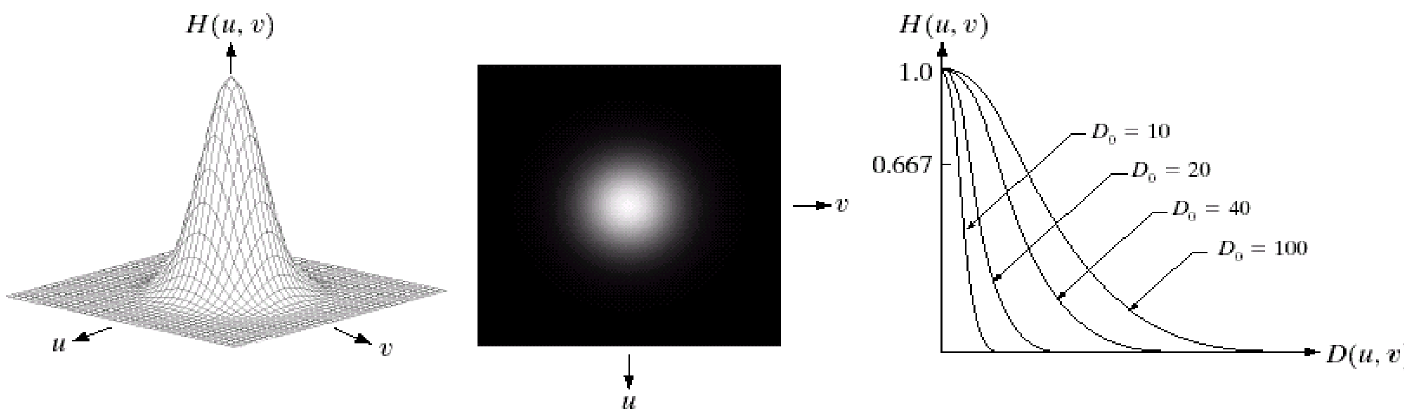
No ringing in this case (with n=2) ringing can be seen in filters of higher order

Gaussian Lowpass Filters

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

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

D_0 is the cutoff frequency



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

Gaussian Lowpass Filters (cont…)

- D_0 is set to 5 different radii of
- 5, 15, 30, 80 and 230
- Smooth transition in
- blurring as a function of
- increasing cutoff frequency

Original image

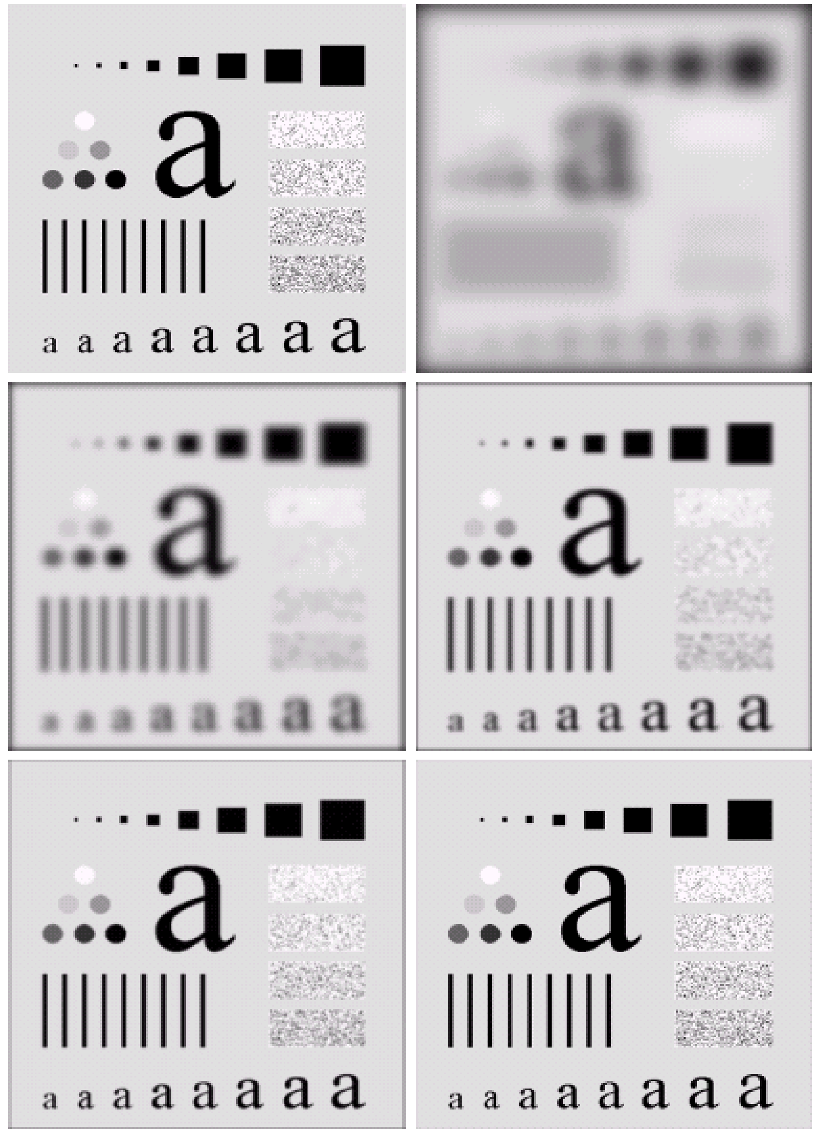
Result of filtering with Gaussian filter with cutoff radius 5

Result of filtering with Gaussian filter with cutoff radius 15

Result of filtering with Gaussian filter with cutoff radius 30

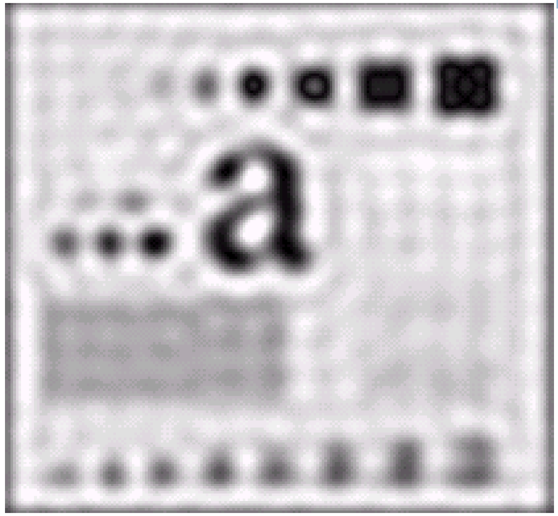
Result of filtering with Gaussian filter with cutoff radius 85

Result of filtering with Gaussian filter with cutoff radius 230

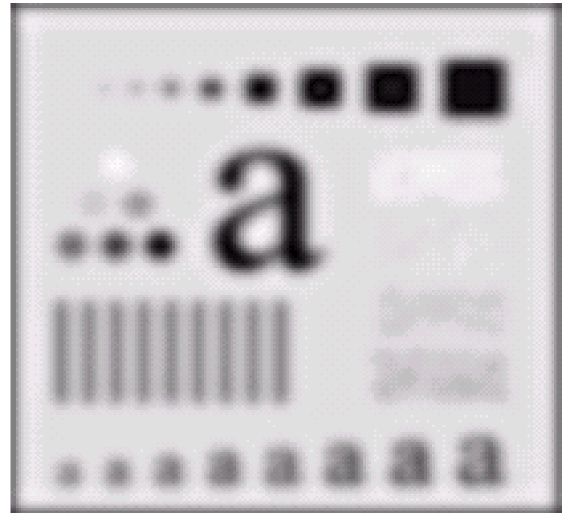


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

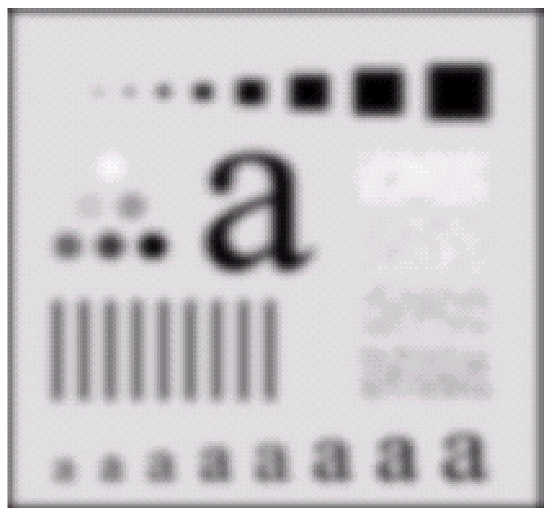
Lowpass Filters Compared



Result of filtering with ideal low pass filter of radius 15



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

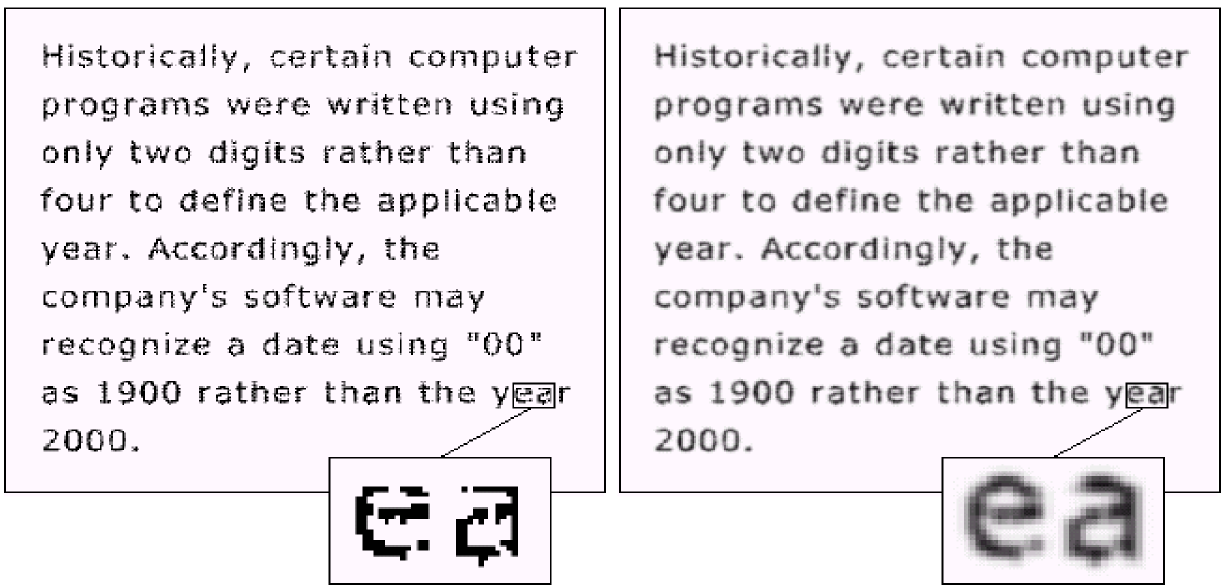


Result of filtering with Gaussian filter with cutoff radius 15

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

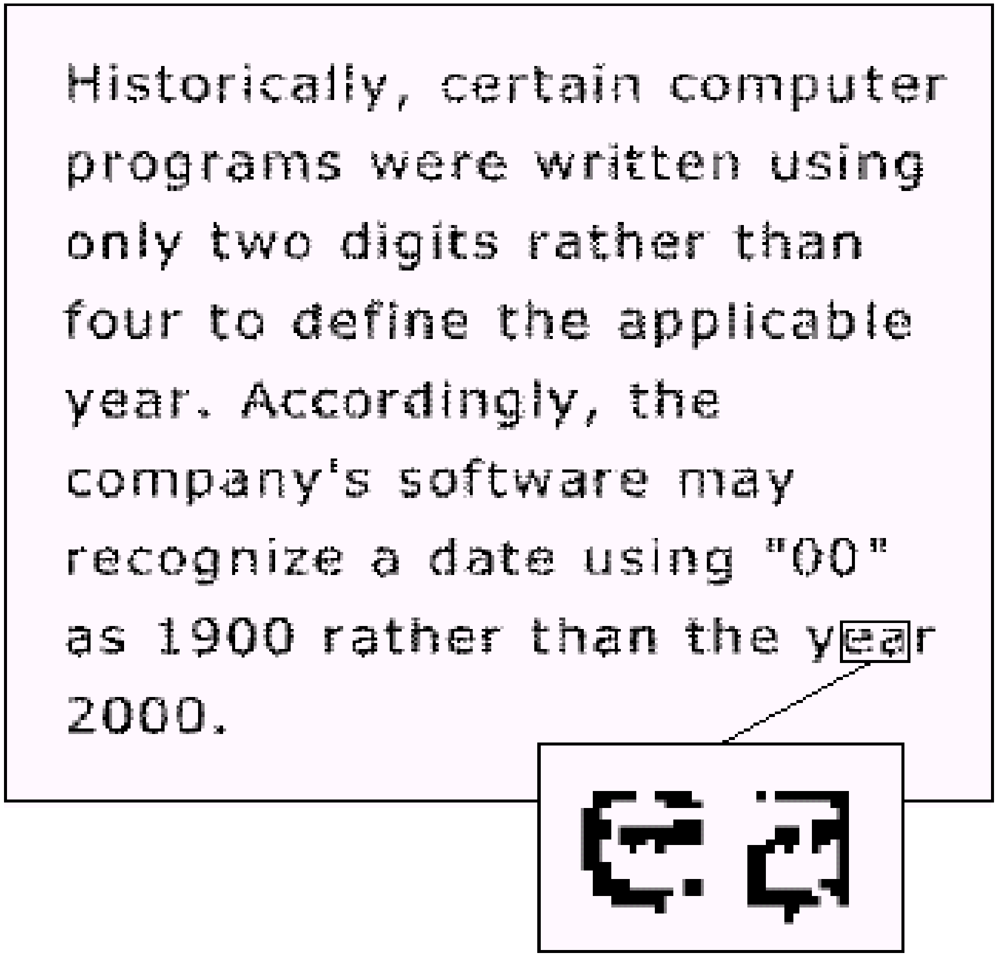
Lowpass Filtering Examples

- A low pass Gaussian filter is used to connect broken text
  - To improve readability (for instance in a character recognition system), typically the solution is to blur the broken corners

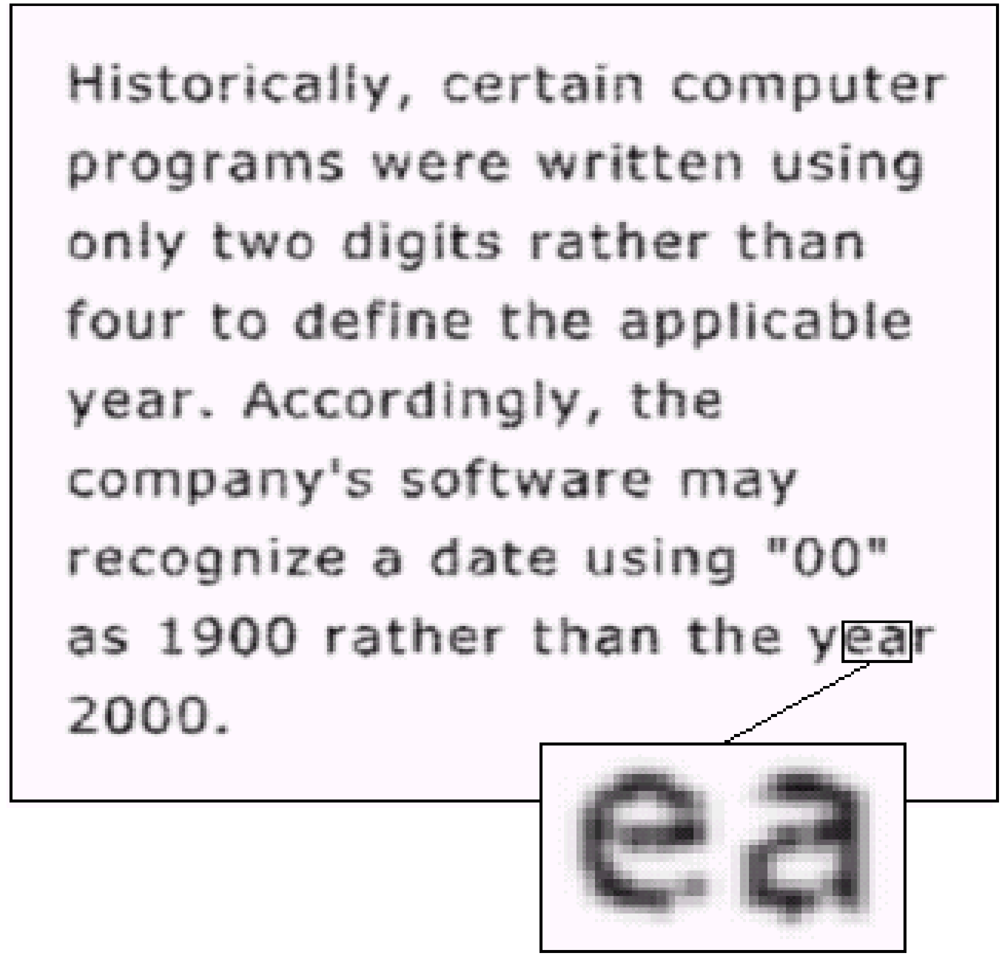


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Lowpass Filtering Examples



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

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

Lowpass Filtering Examples (cont…)

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

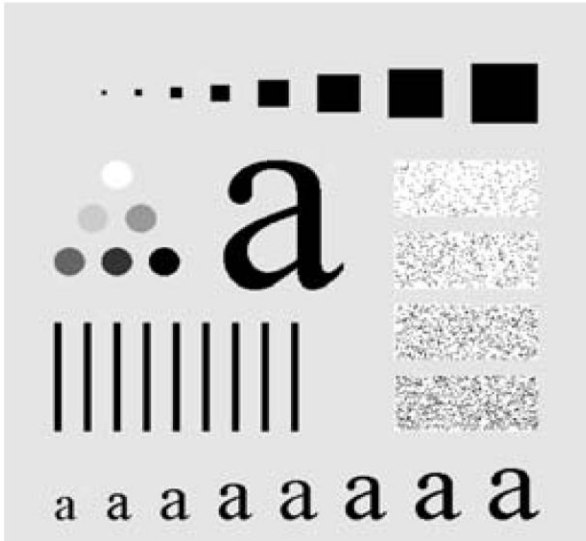
Different lowpass Gaussian filters used to remove blemishes in a photograph

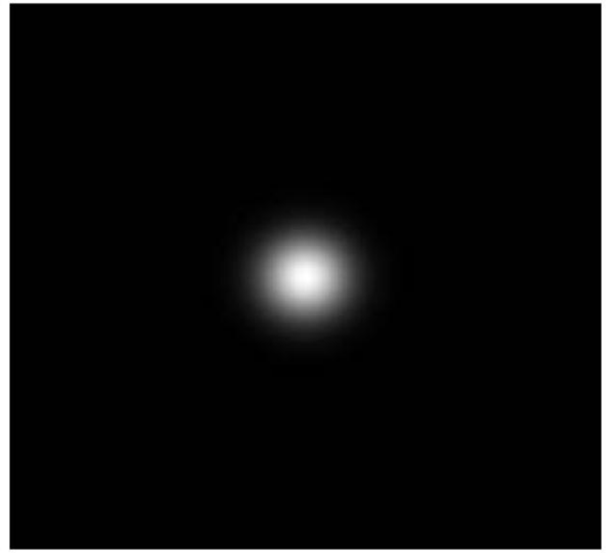
Lowpass Filtering Examples (cont…)

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

Lowpass Filtering Examples (cont…)

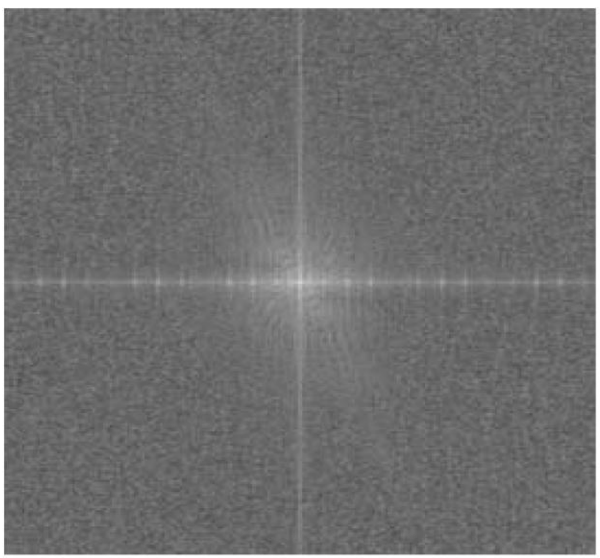
Original image





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, so:

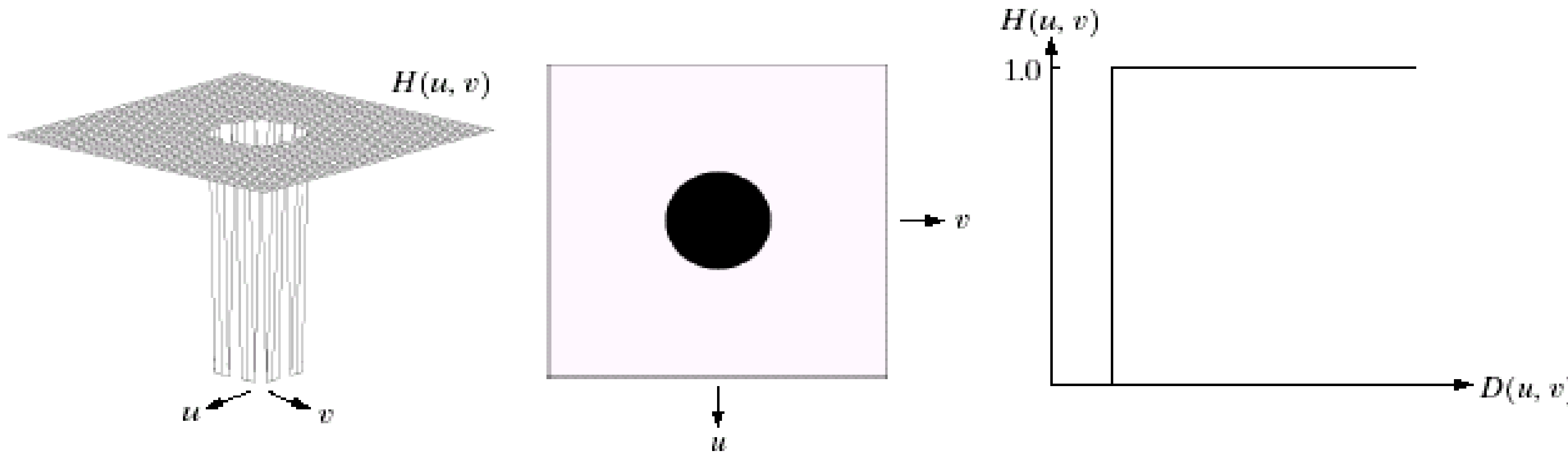
$$H_{hp}(u, v) = 1 - H_{lp}(u, v)$$

Ideal High Pass Filters

The ideal high pass filter is given as:

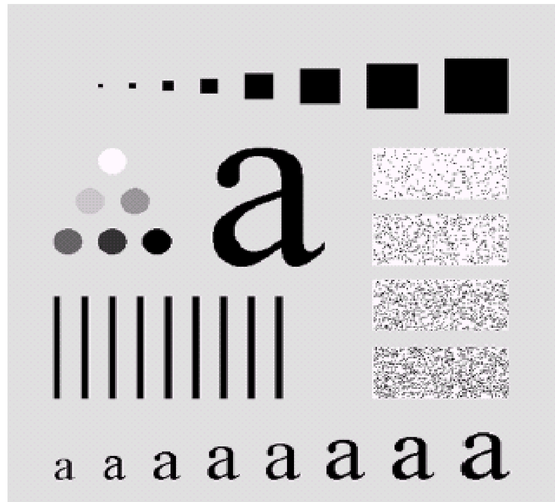
$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

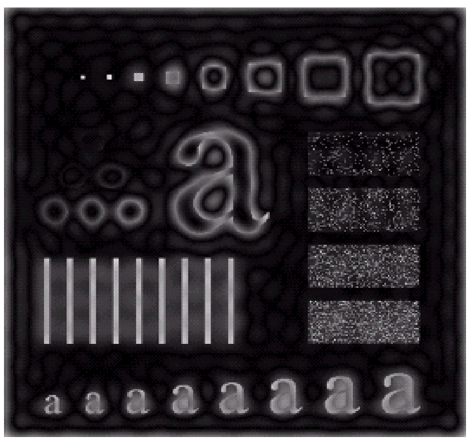
where D_0 is the cut off distance as before



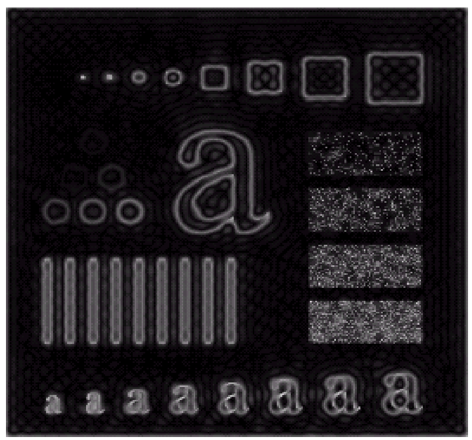
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Ideal High Pass Filters (cont…)

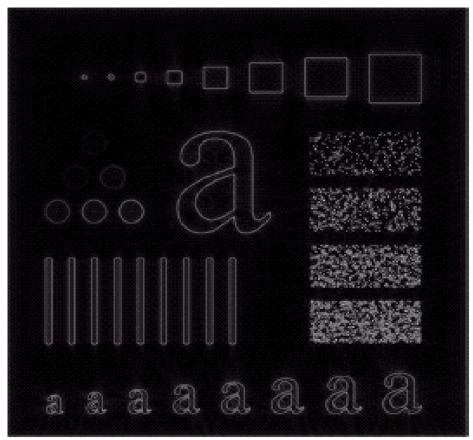




Results of ideal high pass filtering with $D_0 = 15$



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



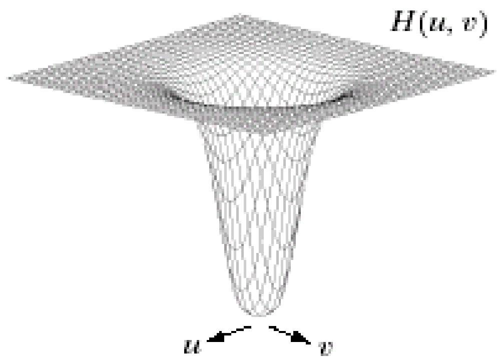
Results of ideal high pass filtering with $D_0 = 80$

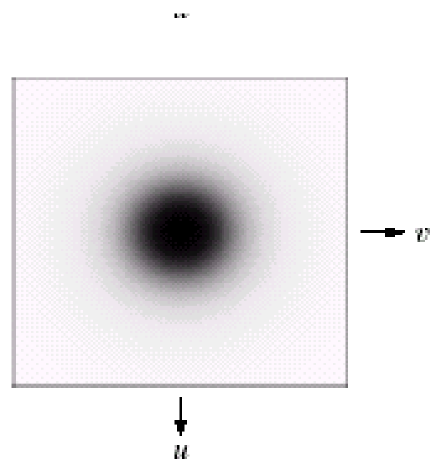
Butterworth High Pass Filters

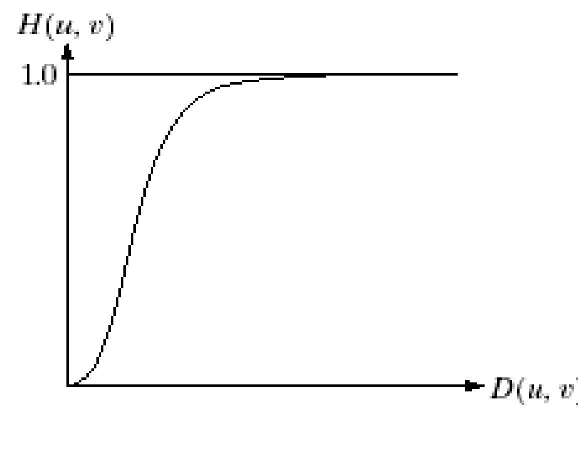
The Butterworth high pass filter is given as:

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

where n is the order and D_0 is the cut off distance as before



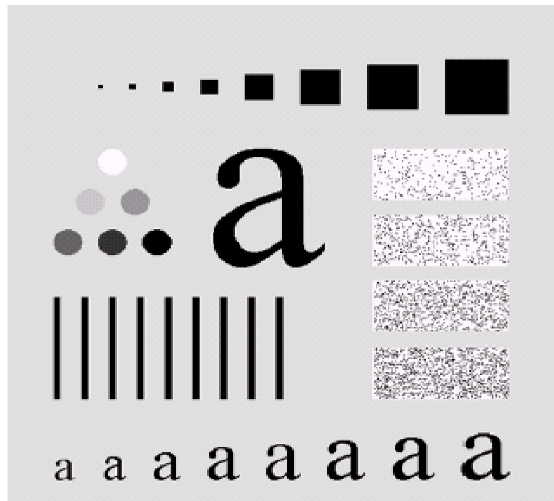


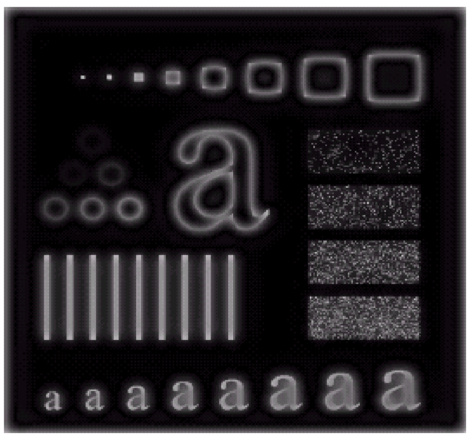


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

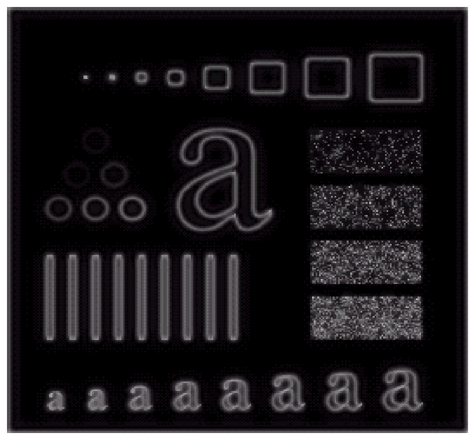
Butterworth High Pass Filters (cont…)

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

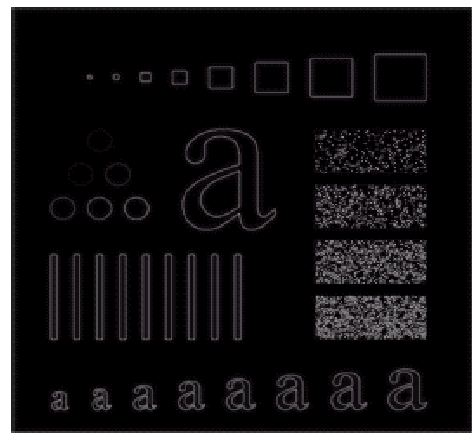




Results of Butterworth high pass filtering of order 2 with $D_0 = 15$



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



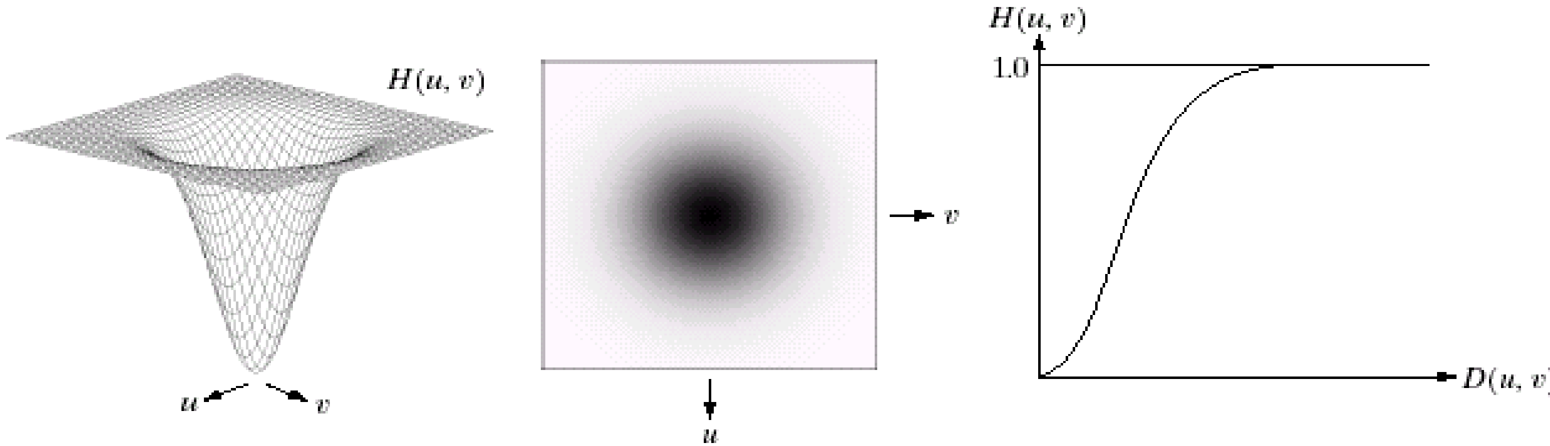
Results of Butterworth high pass filtering of order 2 with $D_0 = 80$

Gaussian High Pass Filters

The Gaussian high pass filter is given as:

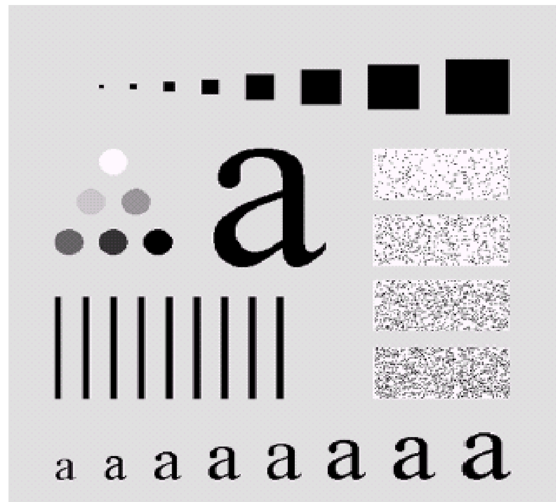
$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^{\,2}}$$

where D_0 is the cut off distance as before

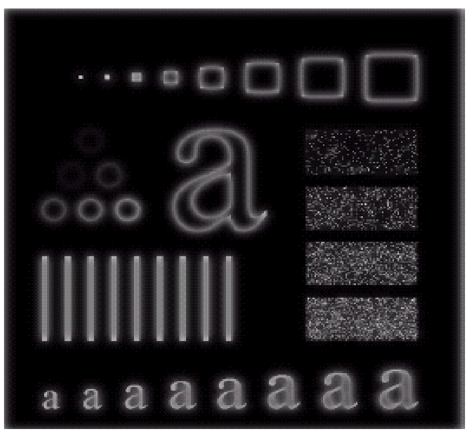


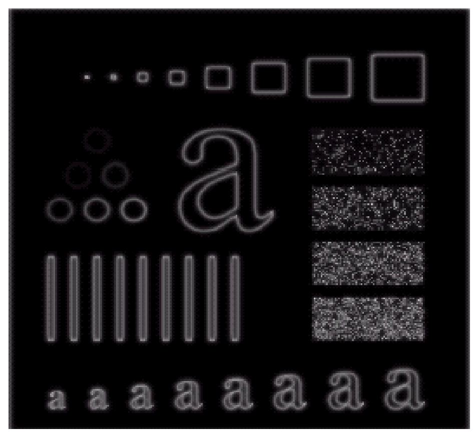
Gaussian High Pass Filters (cont…)

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



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



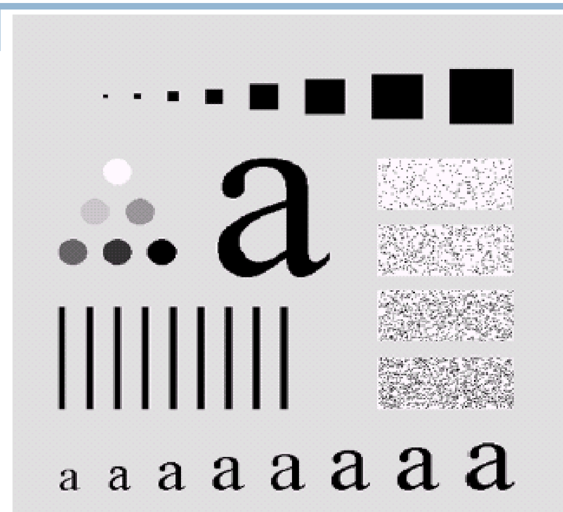


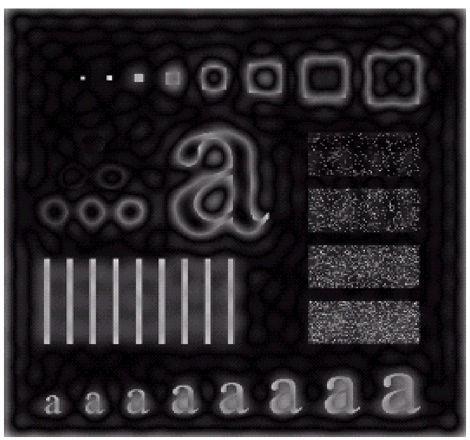
Results of Gaussian high pass filtering with $D_0 = 80$

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

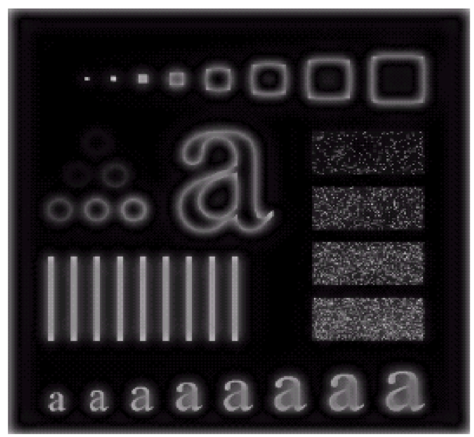
Highpass Filter Comparison

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

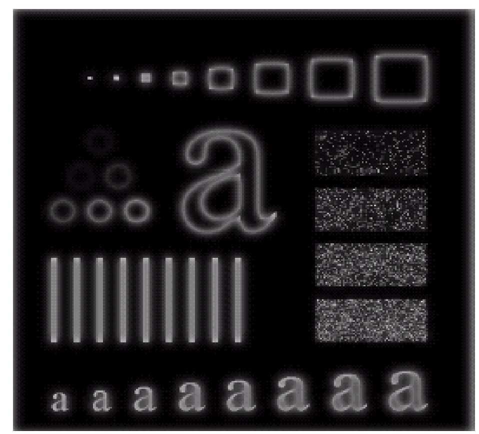




Results of ideal high pass filtering with $D_0 = 15$

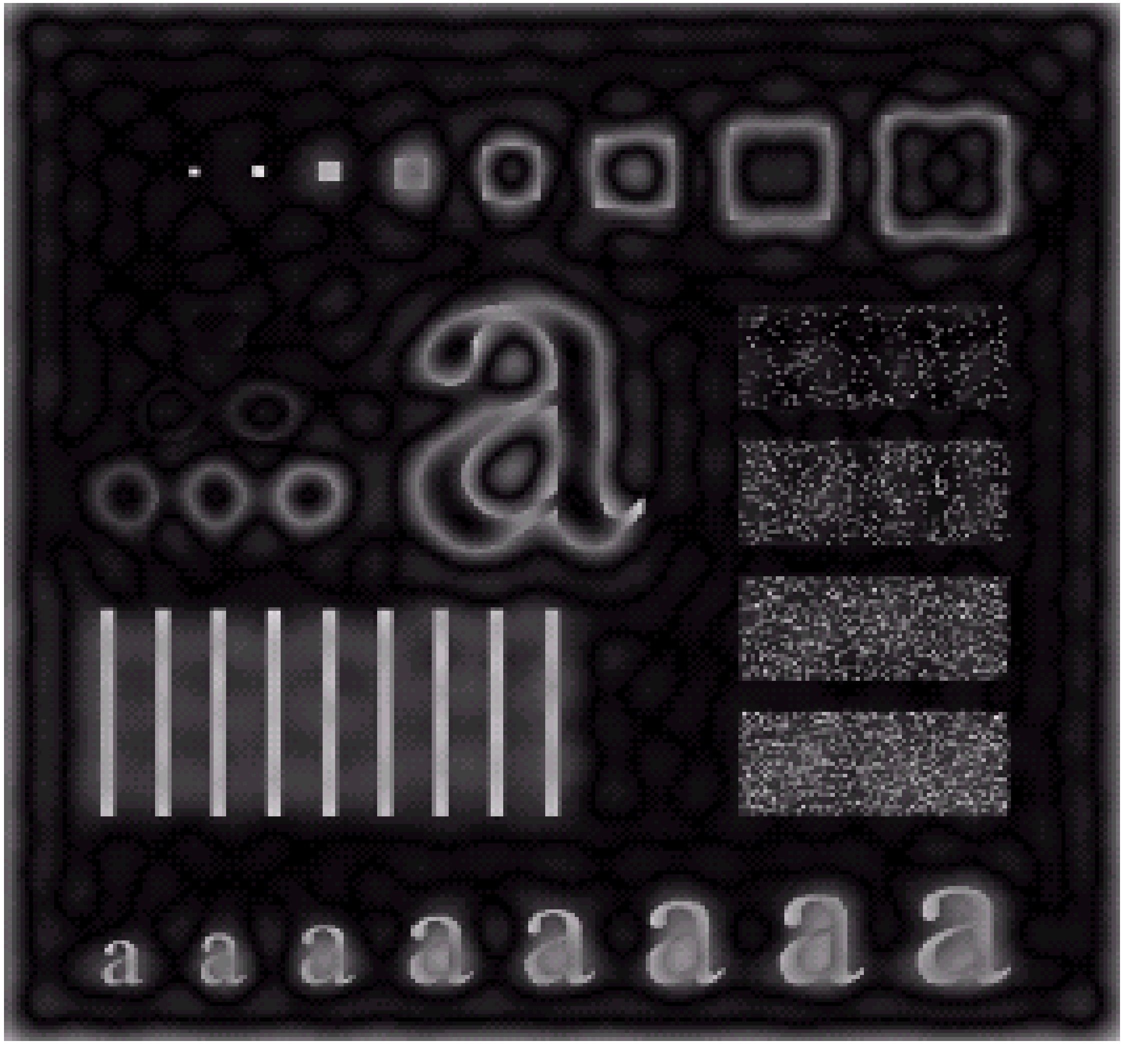


Results of Butterworth high pass filtering of order 2 with $D_0 = 15$



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

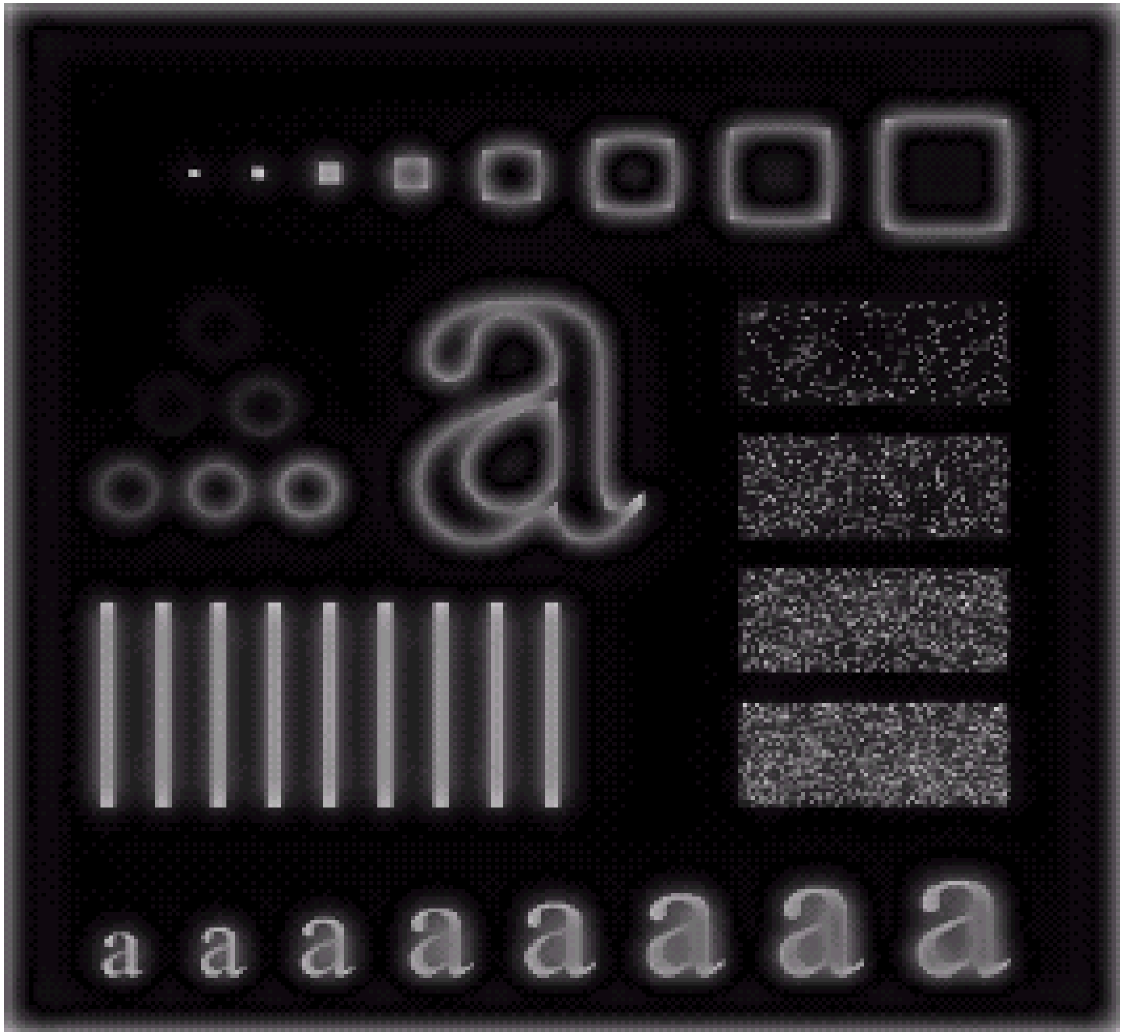
Highpass Filter Comparison



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

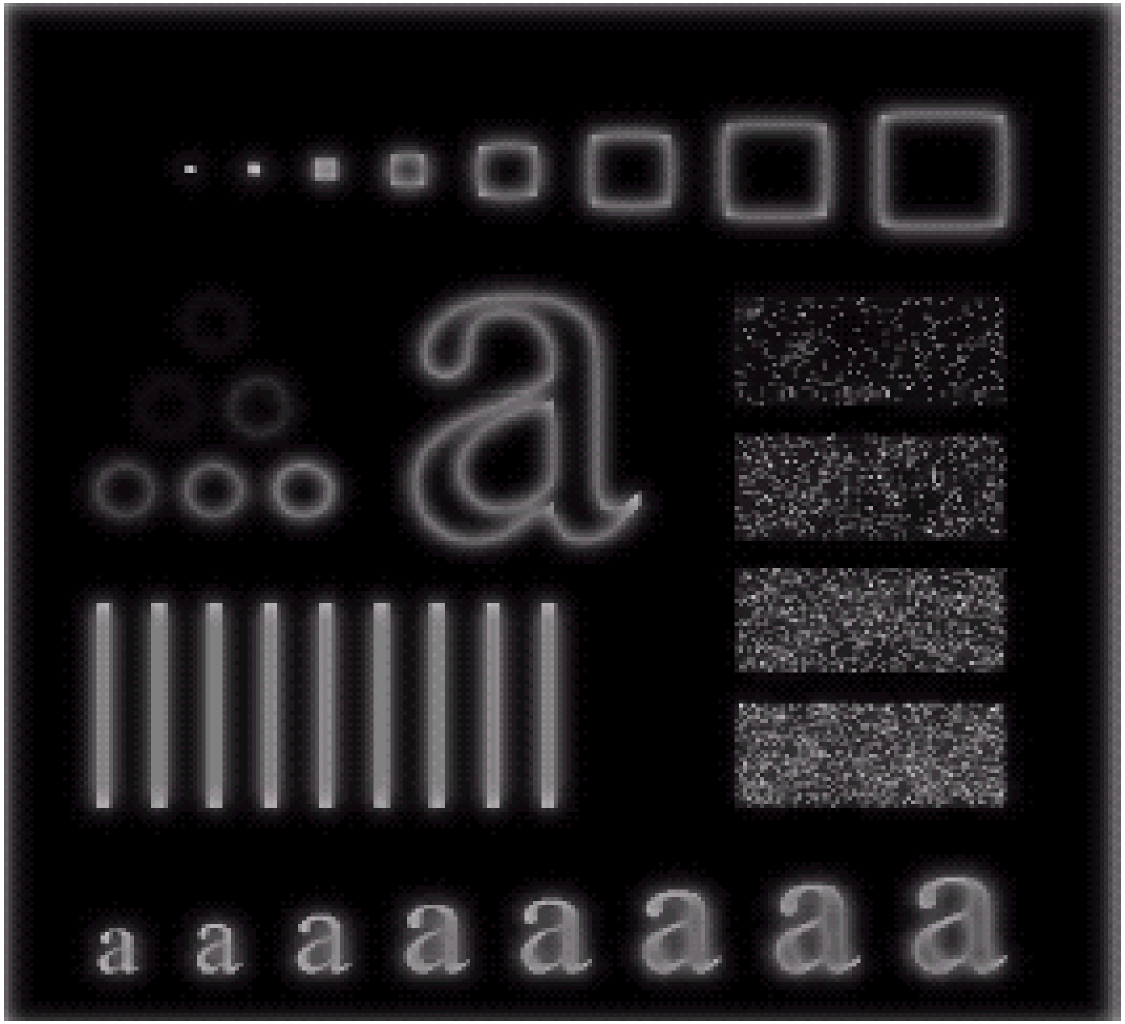
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

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 $D_0 = 15$

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

HIGHPASS FILTERS: ILLUSTRATION

Ideal

Butterworth

Gaussian

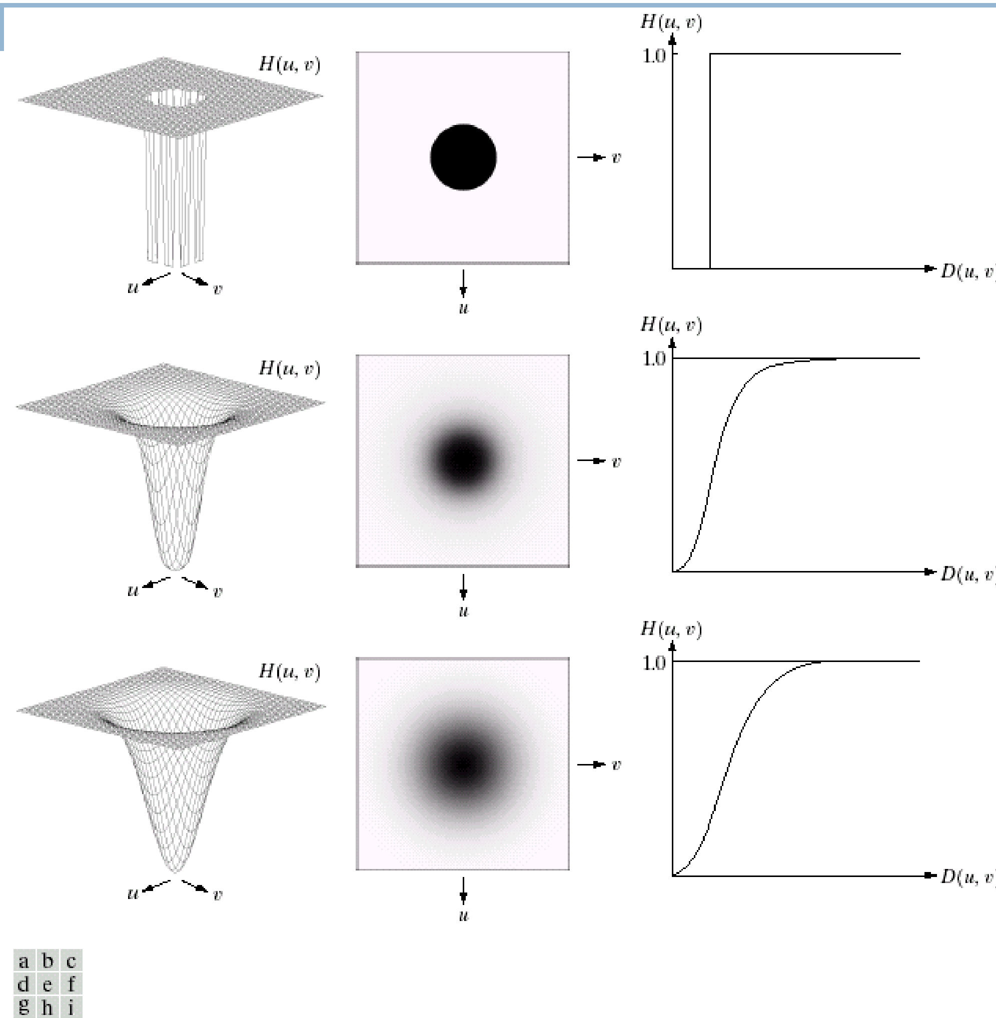
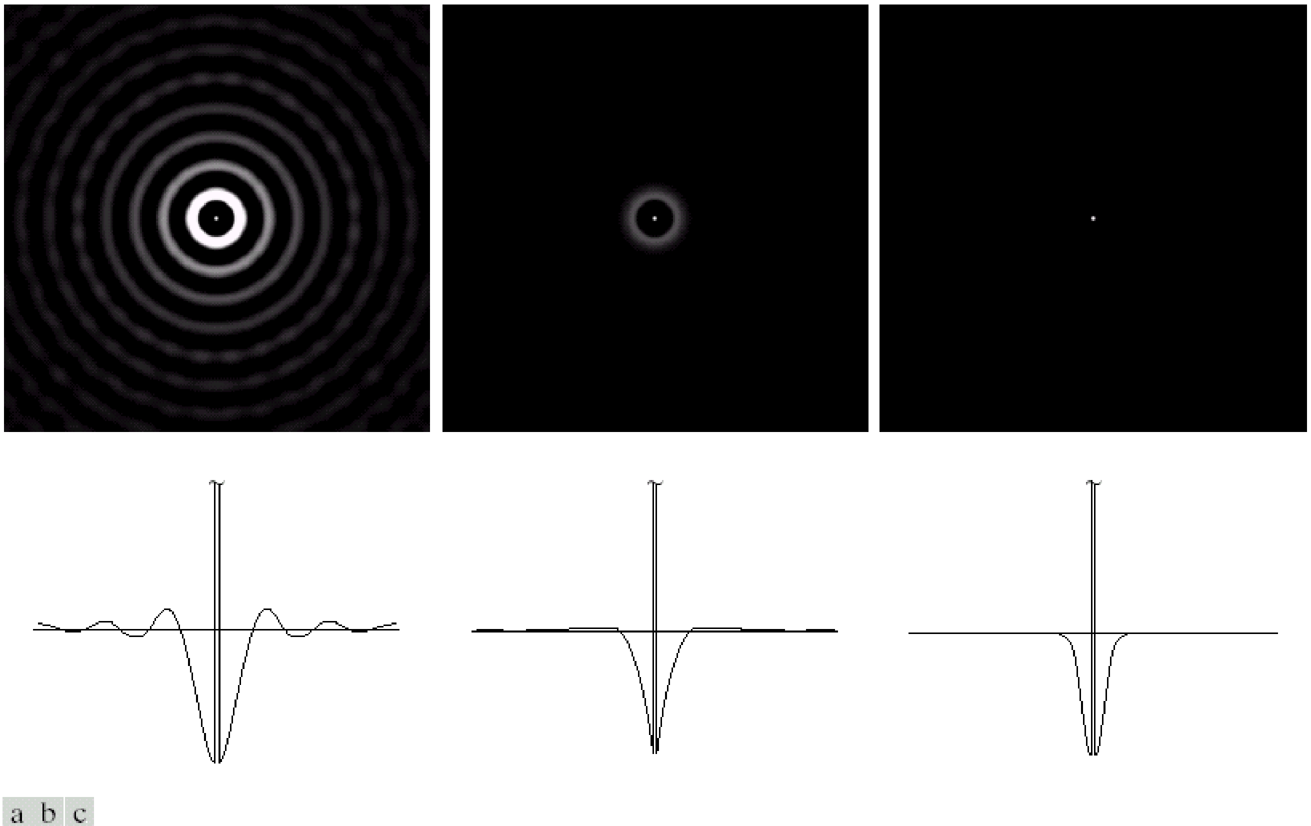


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

HIGHPASS FILTERS: ILLUSTRATION



a b c

FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

HIGH PASS FILTER

- Ideal High Pass

$$H(u,v) = 0 \quad \text{if } D(u,v) \leq D_0$$
$$= 1 \quad \text{if } D(u,v) > D_0$$

where $D(u,v)$ is the distance from $(M/2,N/2)$

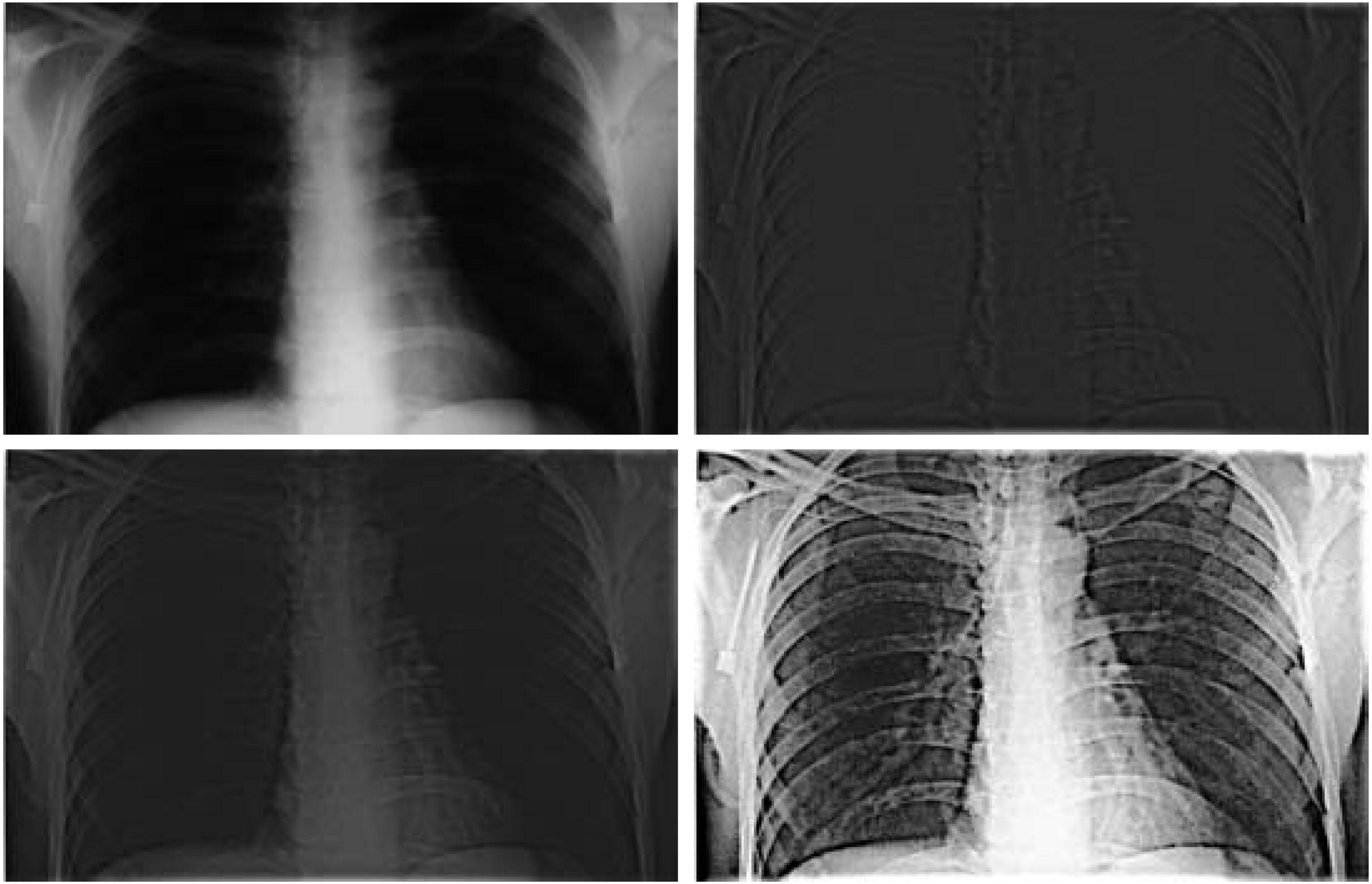
- Butterworth Highpass

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

- Gaussian Highpass

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

Highpass Filtering Example



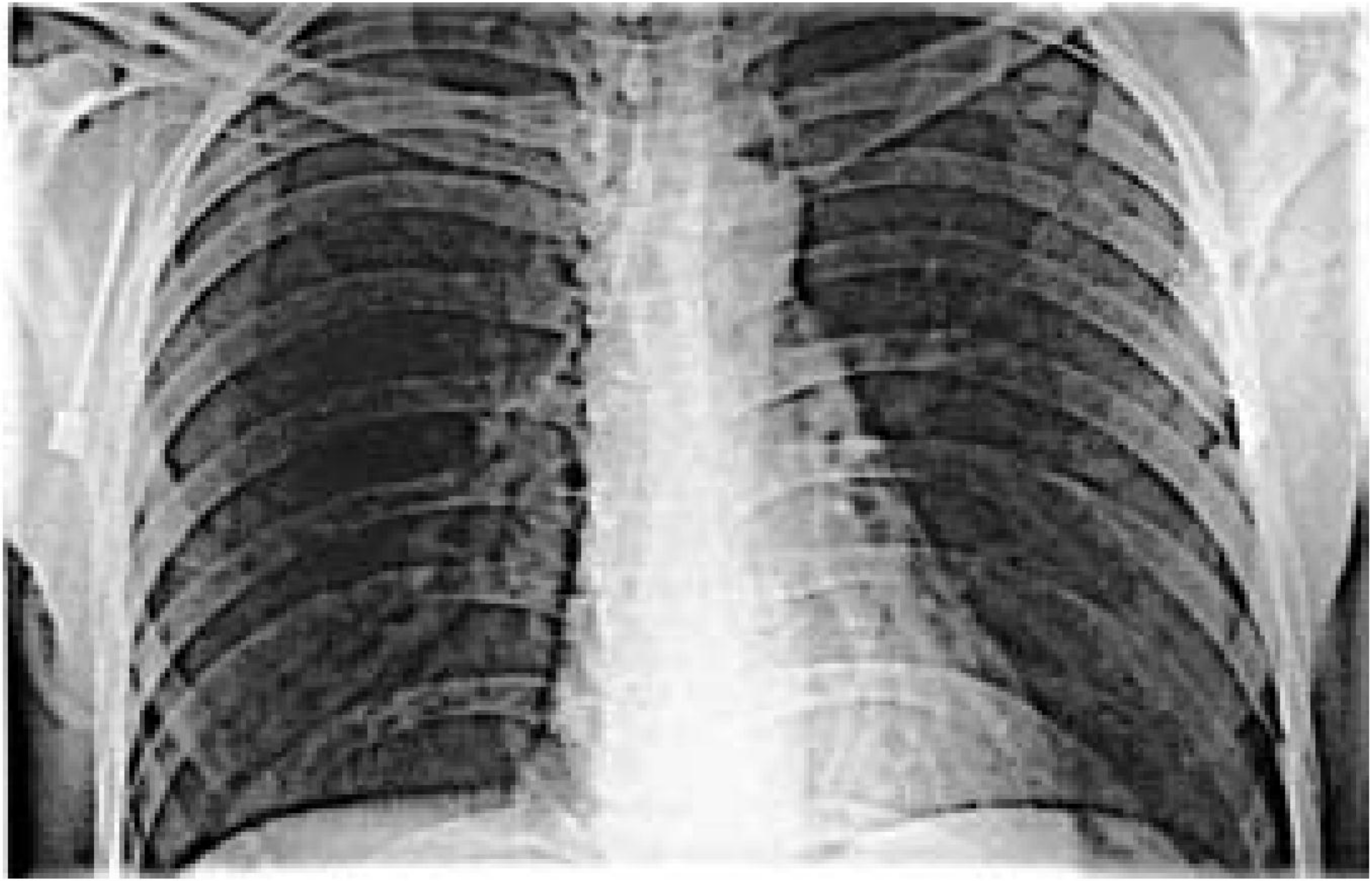
Original image

Highpass filtering result

High frequency emphasis result

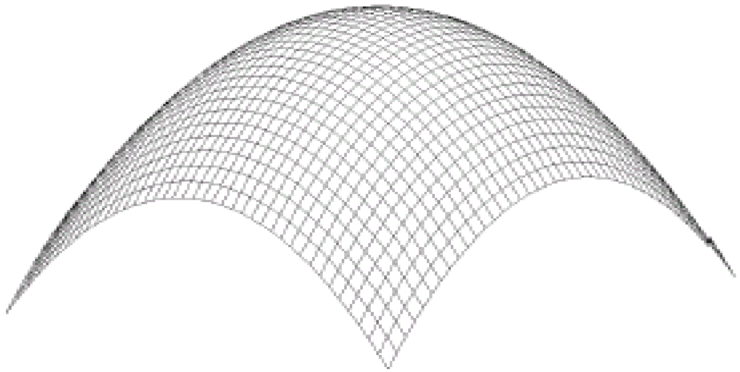
After histogram equalisation

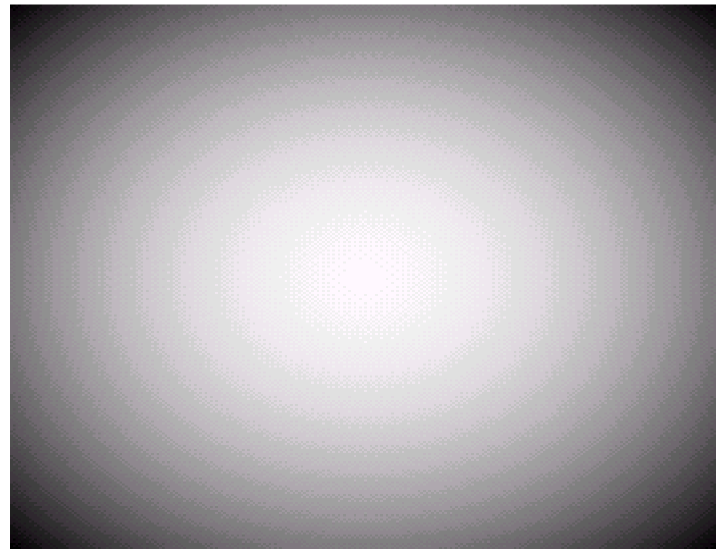
Images taken from Gonzalez & Woods, Digital Image Processing (2002)



Laplacian In The Frequency Domain

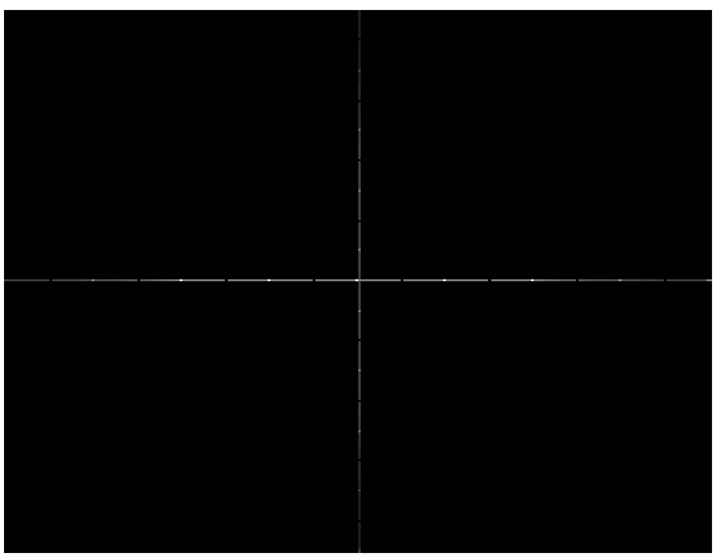
Laplacian in the frequency domain

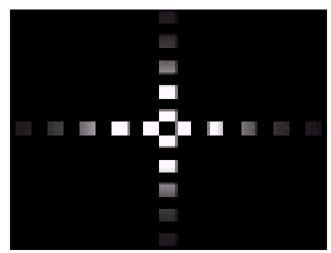


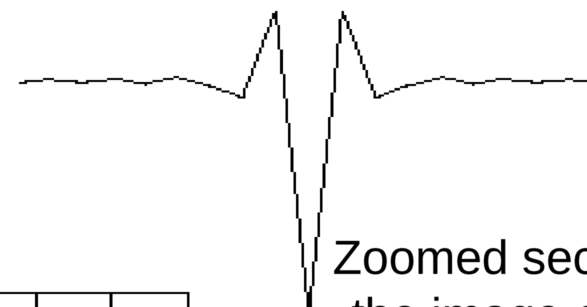


2-D image of Laplacian in the frequency domain

Inverse DFT of Laplacian in the frequency domain





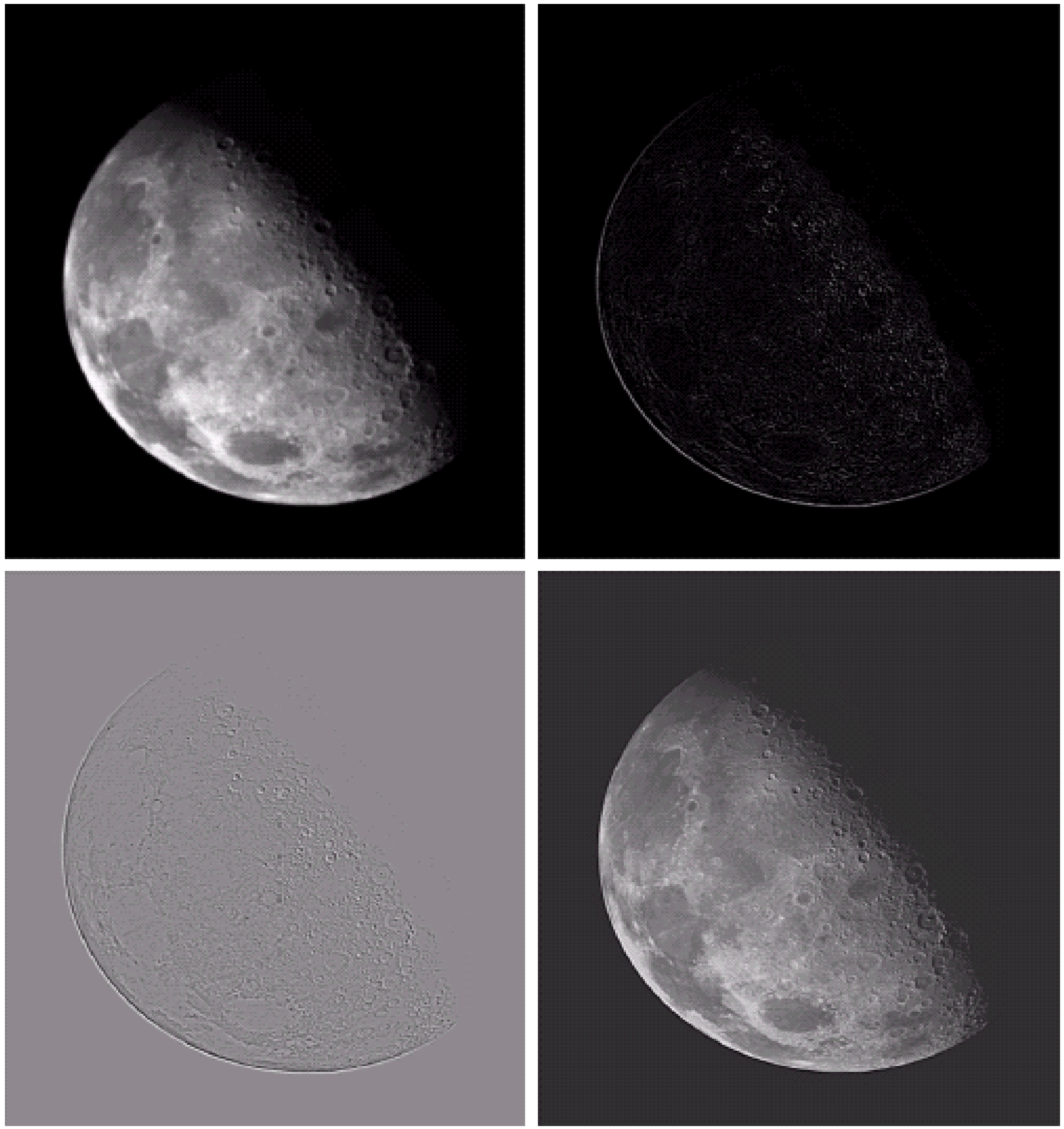


0	1	0
1	−4	1
0	1	0

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

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

Frequency Domain Laplacian Example



Original image

Laplacian filtered image

Laplacian image scaled

Enhanced image

Fast Fourier Transform

- 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 much faster – especially for large images

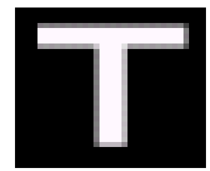
Summary

In this lecture we examined image enhancement in the frequency domain

- The Fourier series & the Fourier transform
- Image Processing in the frequency domain
  - Image smoothing
  - Image sharpening
- Fast Fourier Transform

Next time we will begin to examine image restoration using the spatial and frequency based techniques we have been looking at

Interesting Application Of Frequency Domain Filtering



Interesting Application Of Frequency Domain Filtering

Questions?