Digital Image Processing

Image Enhancement:
Filtering in the Frequency Domain

The Discrete Fourier Transform (DFT)

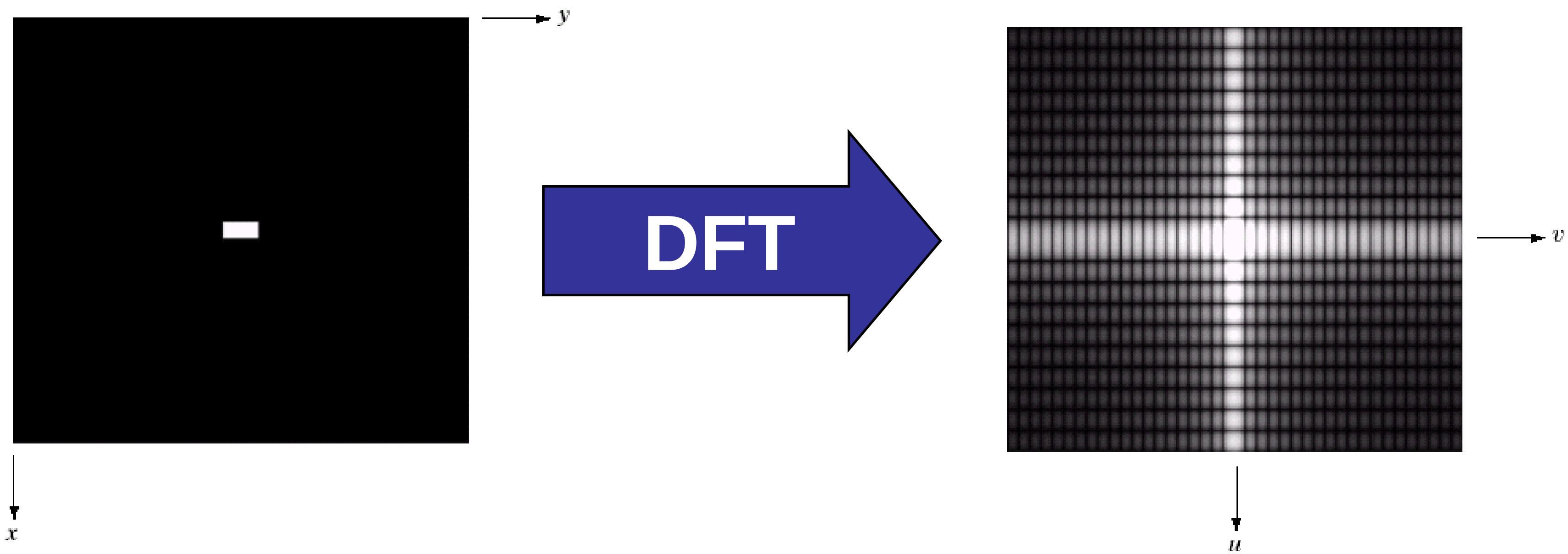
The *Discrete Fourier Transform* of $f(x, y)$, for x = 0, 1, 2…M-1 and y = 0,1,2…N-1, denoted by $F(u, v)$, is given by the equation:

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

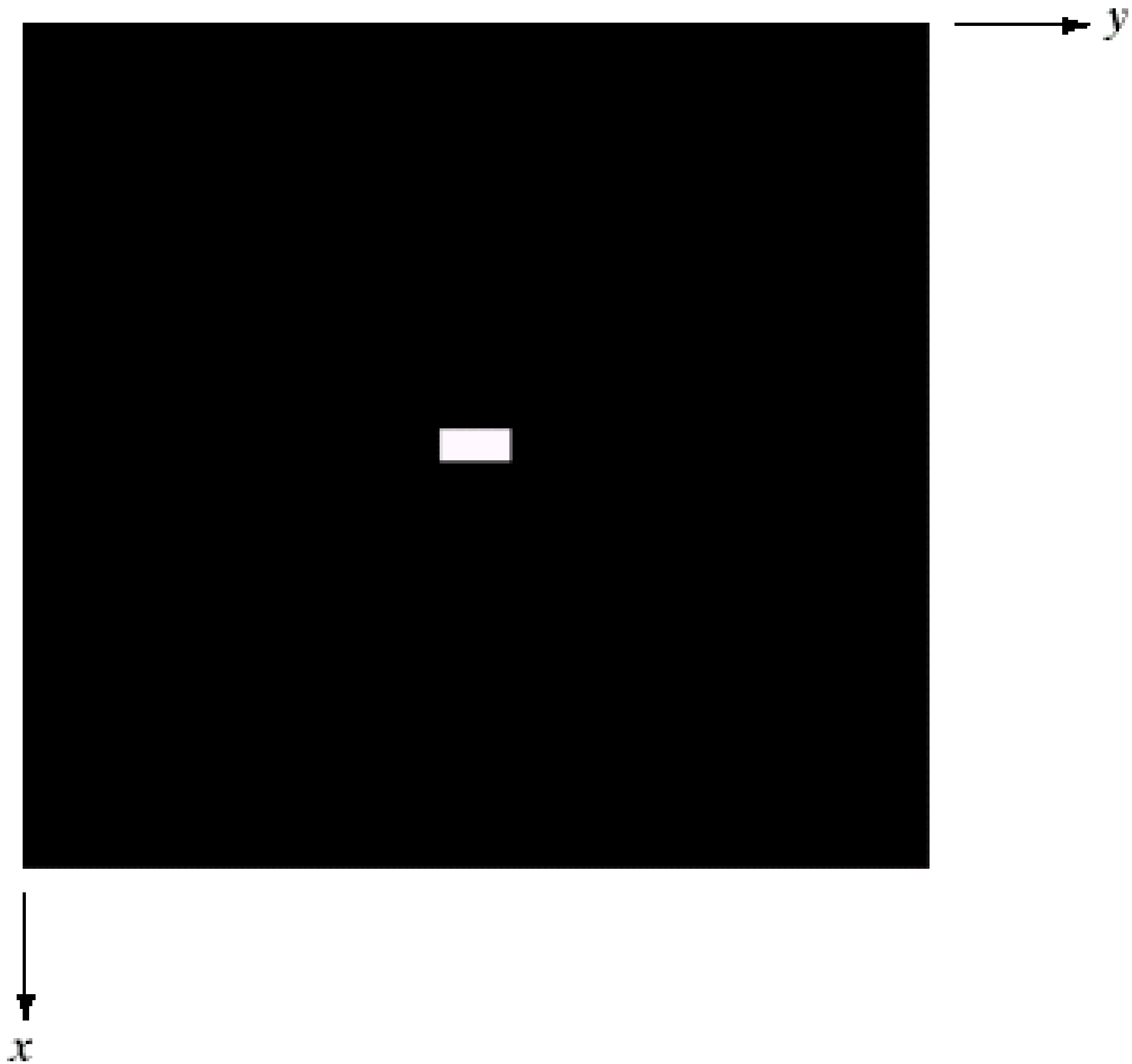
for u = 0, 1, 2…M-1 and v = 0, 1, 2…N-1.

DFT & Images

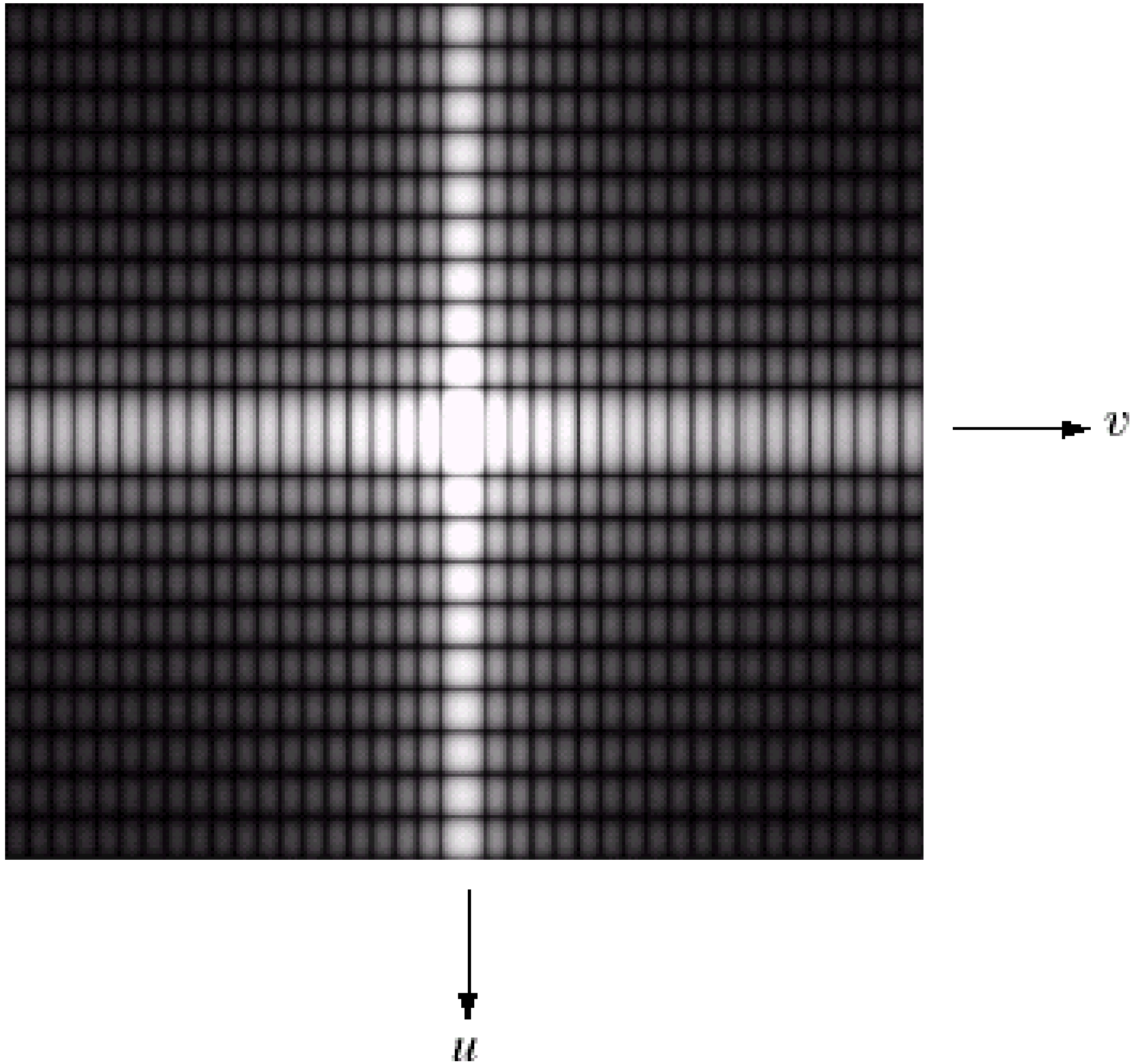
The DFT of a two dimensional image can be visualised by showing the spectrum of the images component frequencies



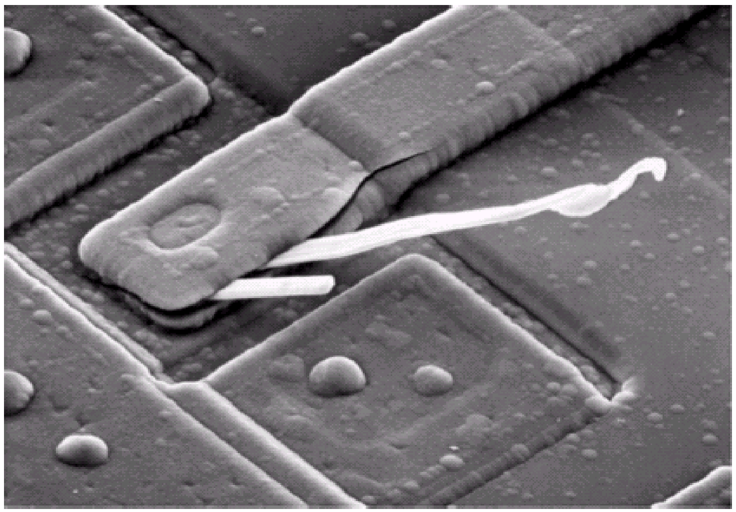
DFT & Images



DFT & Images

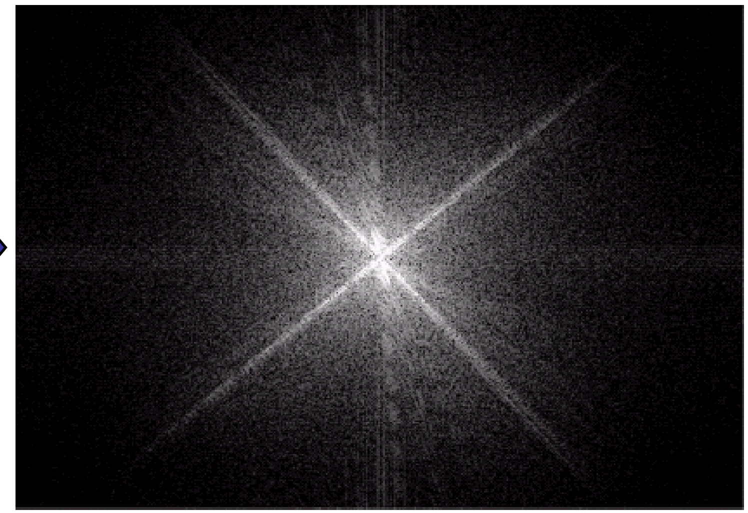


DFT & Images (cont…)



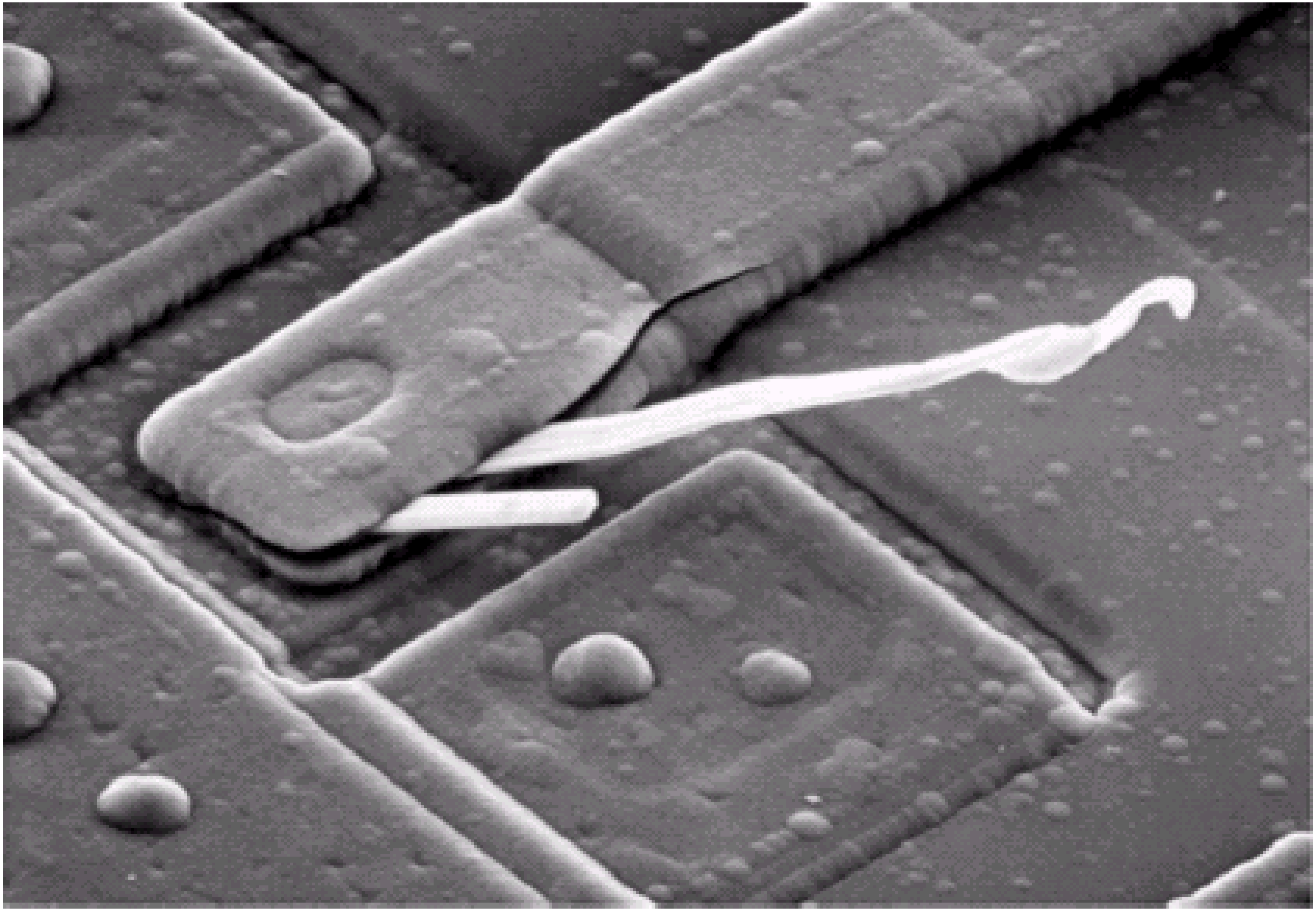
Scanning electron microscope image of an integrated circuit magnified ~2500 times

DFT

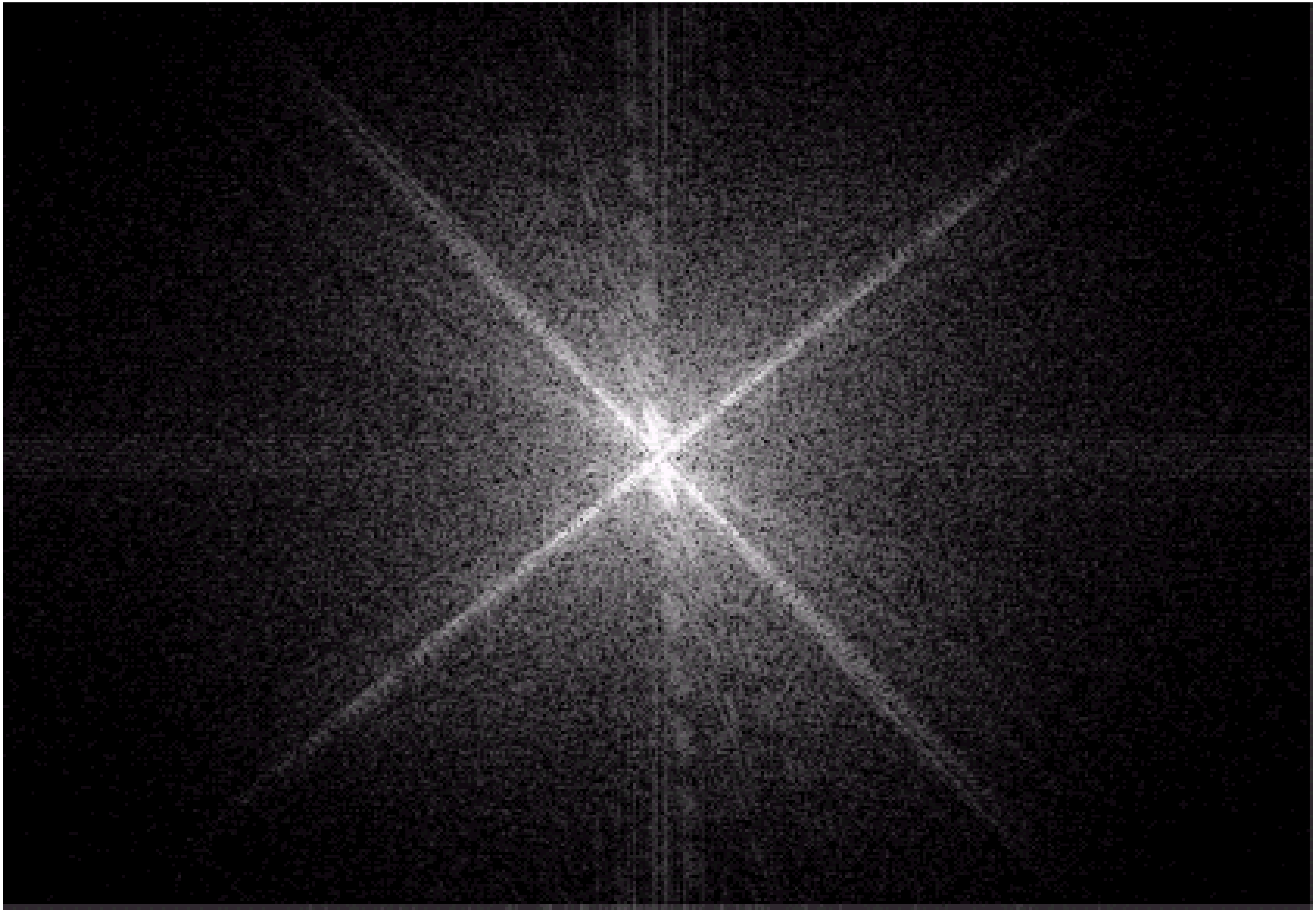


Fourier spectrum of the image

DFT & Images (cont…)



DFT & Images (cont…)



The Inverse DFT

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

The inverse DFT is given by:

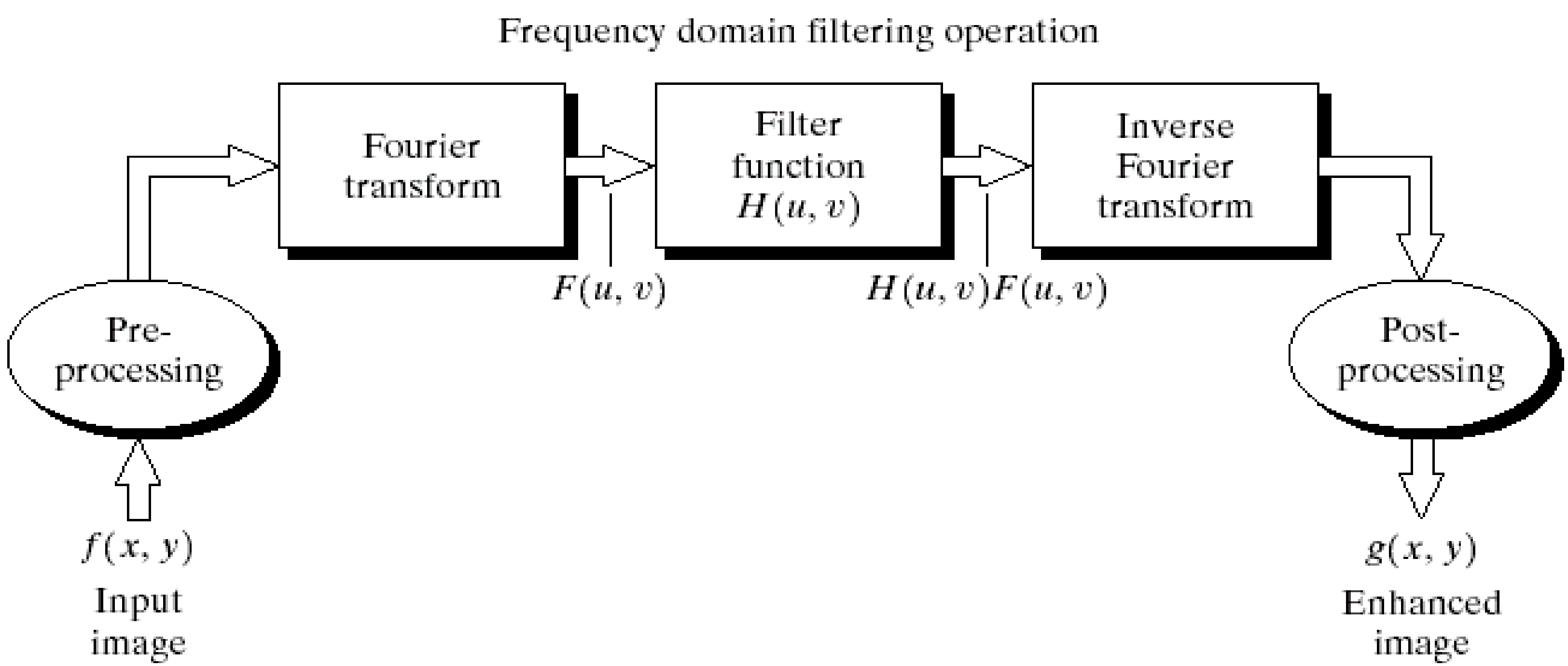
$$f(x,y)=\frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}F(u,v)e^{j2\pi(ux/M+vy/N)}$$

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

The DFT and Image Processing

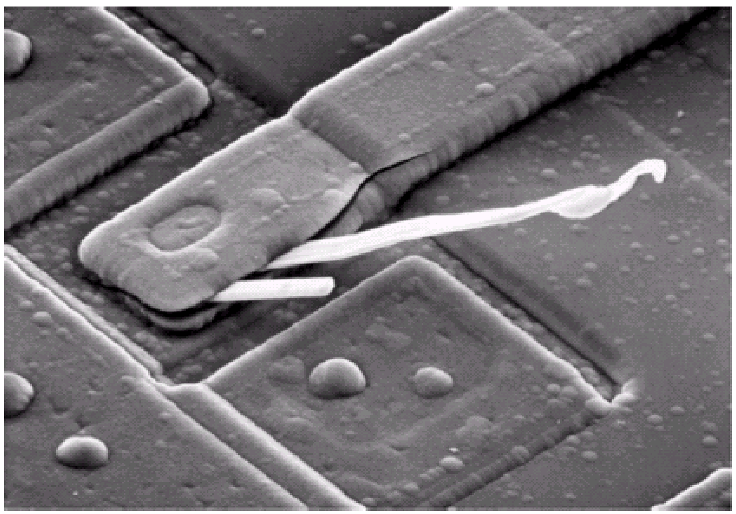
To filter an image in the frequency domain:

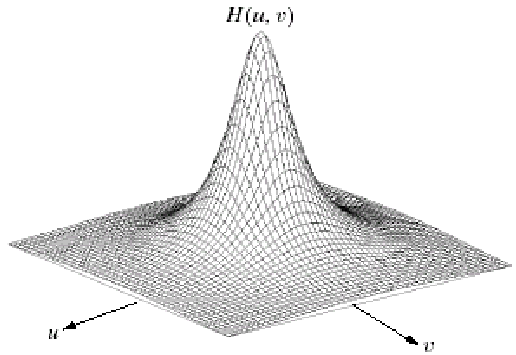
1. Compute *F(u,v)* the DFT of the image
2. Multiply *F(u,v)* by a filter function *H(u,v)*
3. Compute the inverse DFT of the result

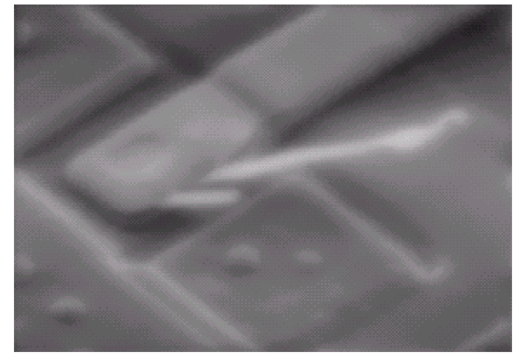


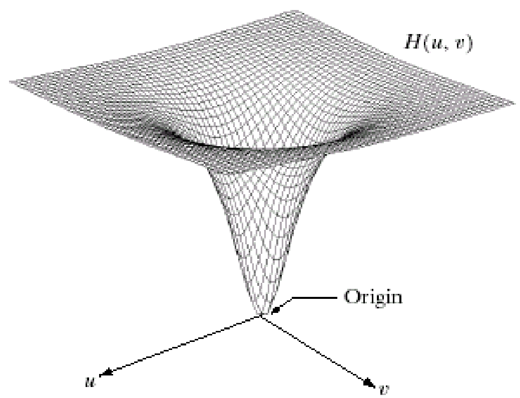
Some Basic Frequency Domain Filters

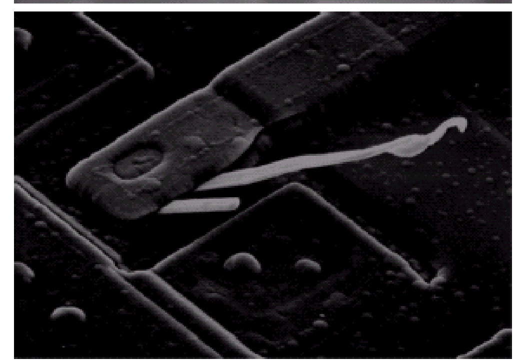
Low Pass Filter







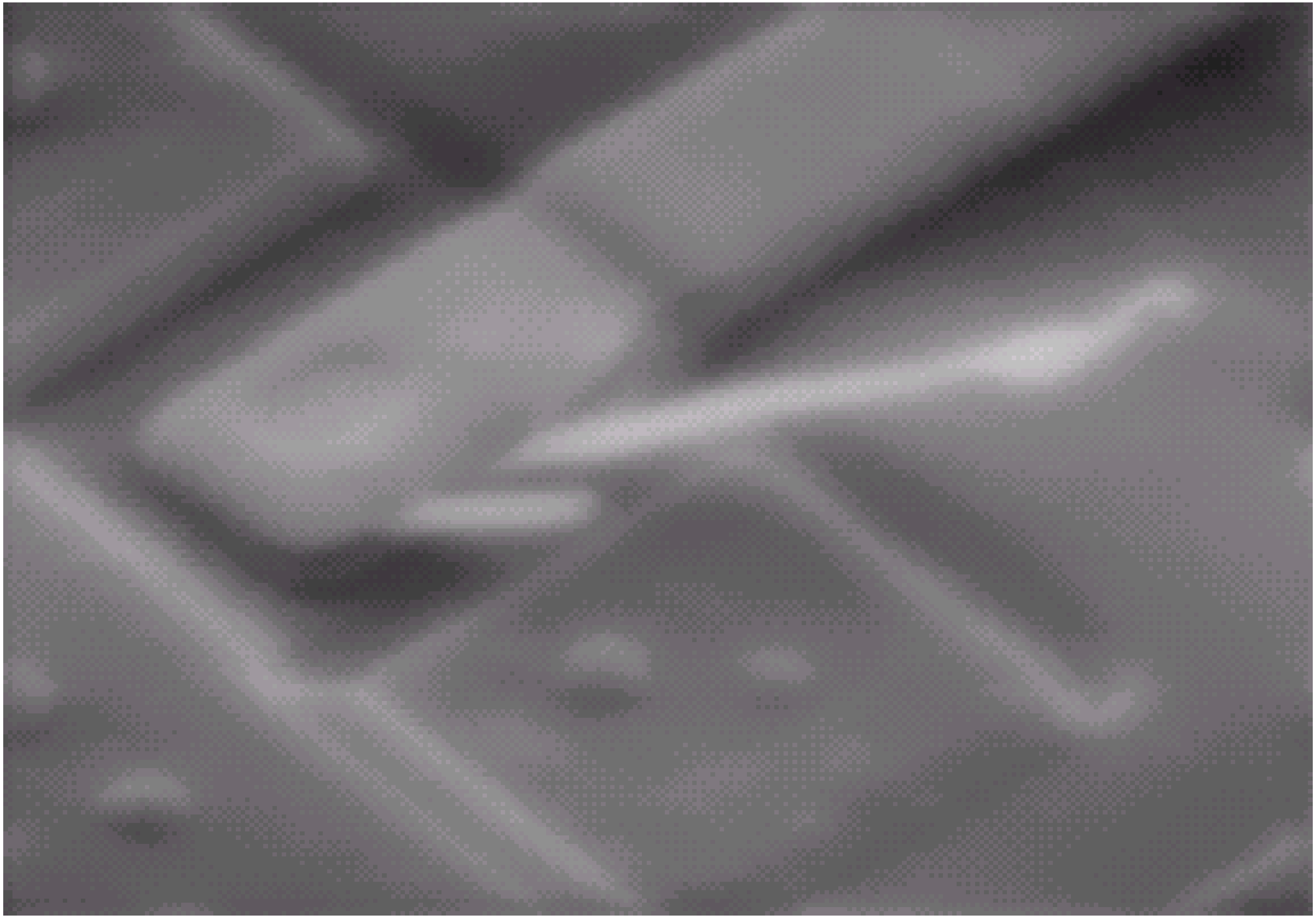




High Pass Filter

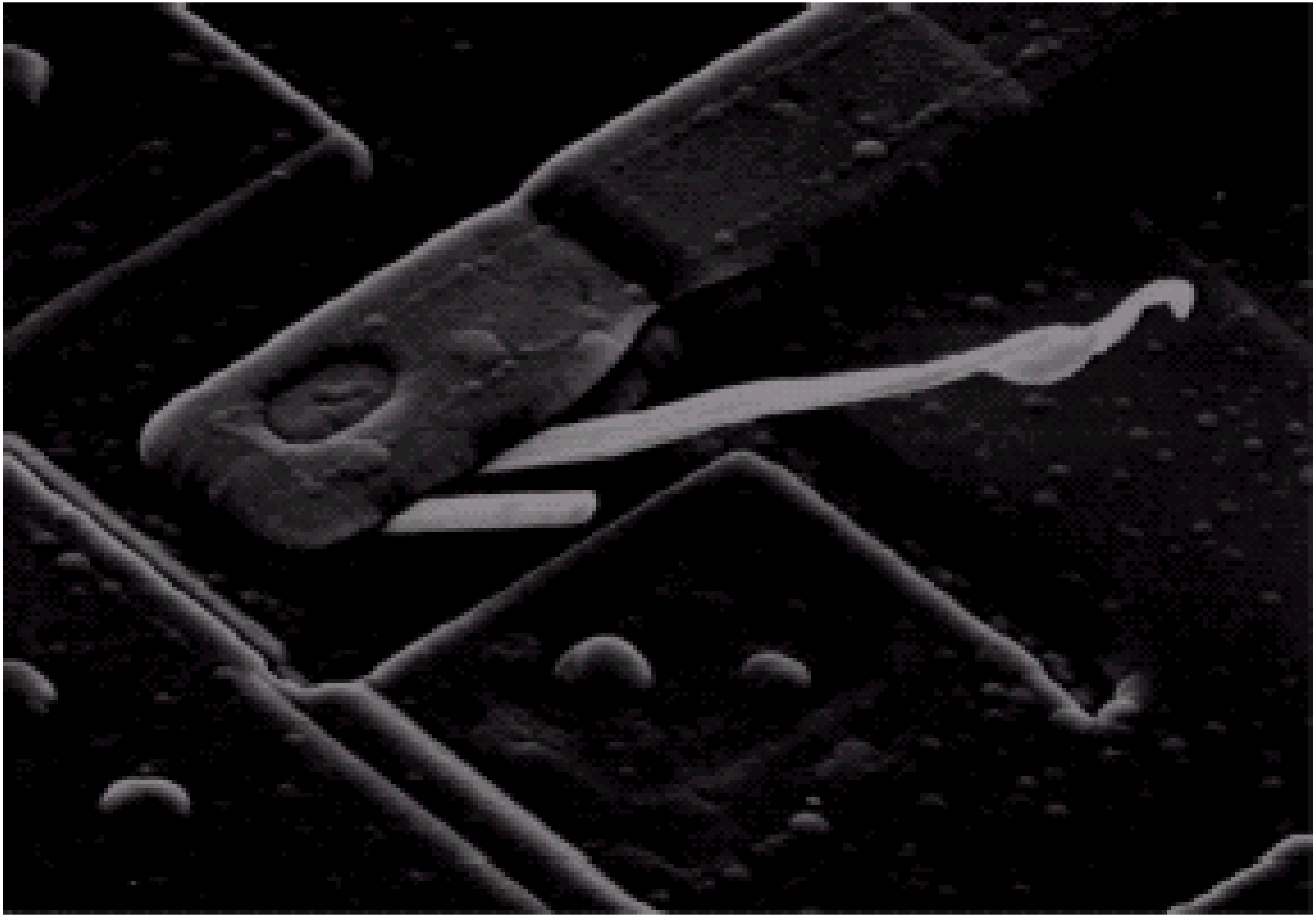
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

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:

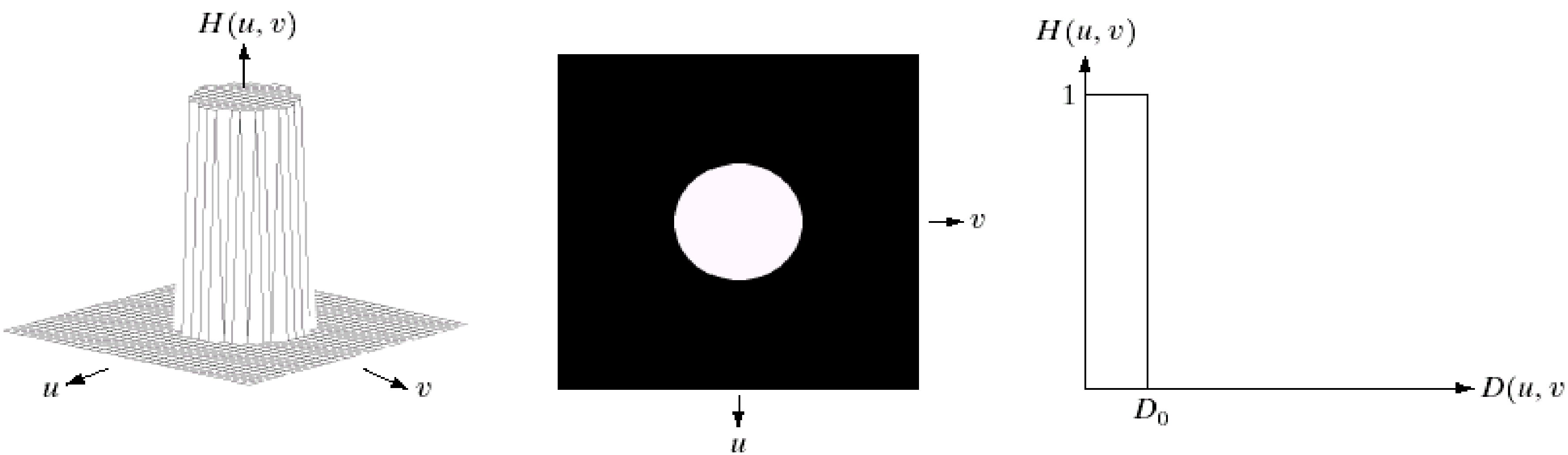
$$G(u,v) = H(u,v)F(u,v)$$

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



changing the distance changes the behaviour of the filter

Ideal Low Pass Filter (cont…)

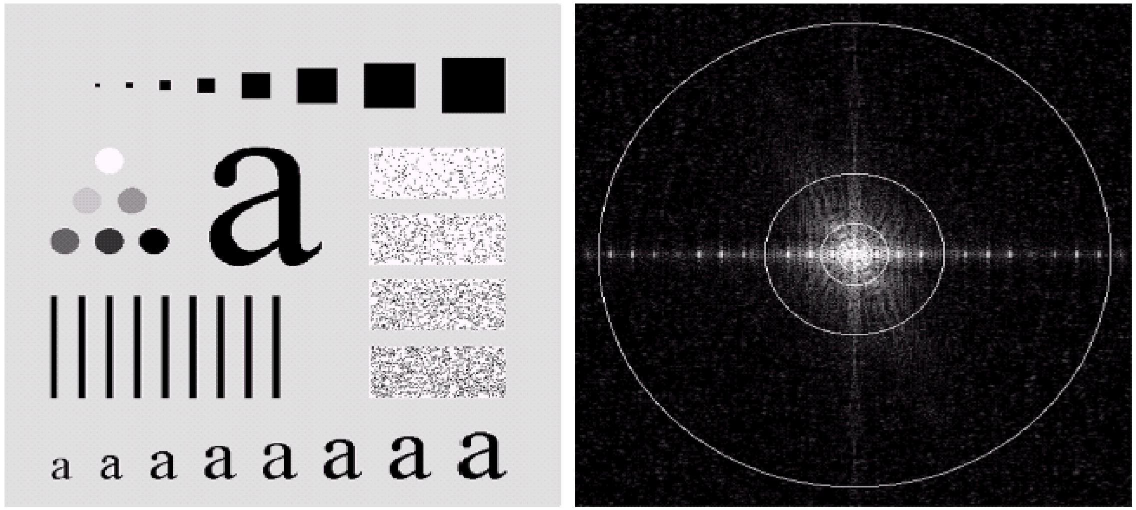
The transfer function for the ideal low pass filter can be given as:

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

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

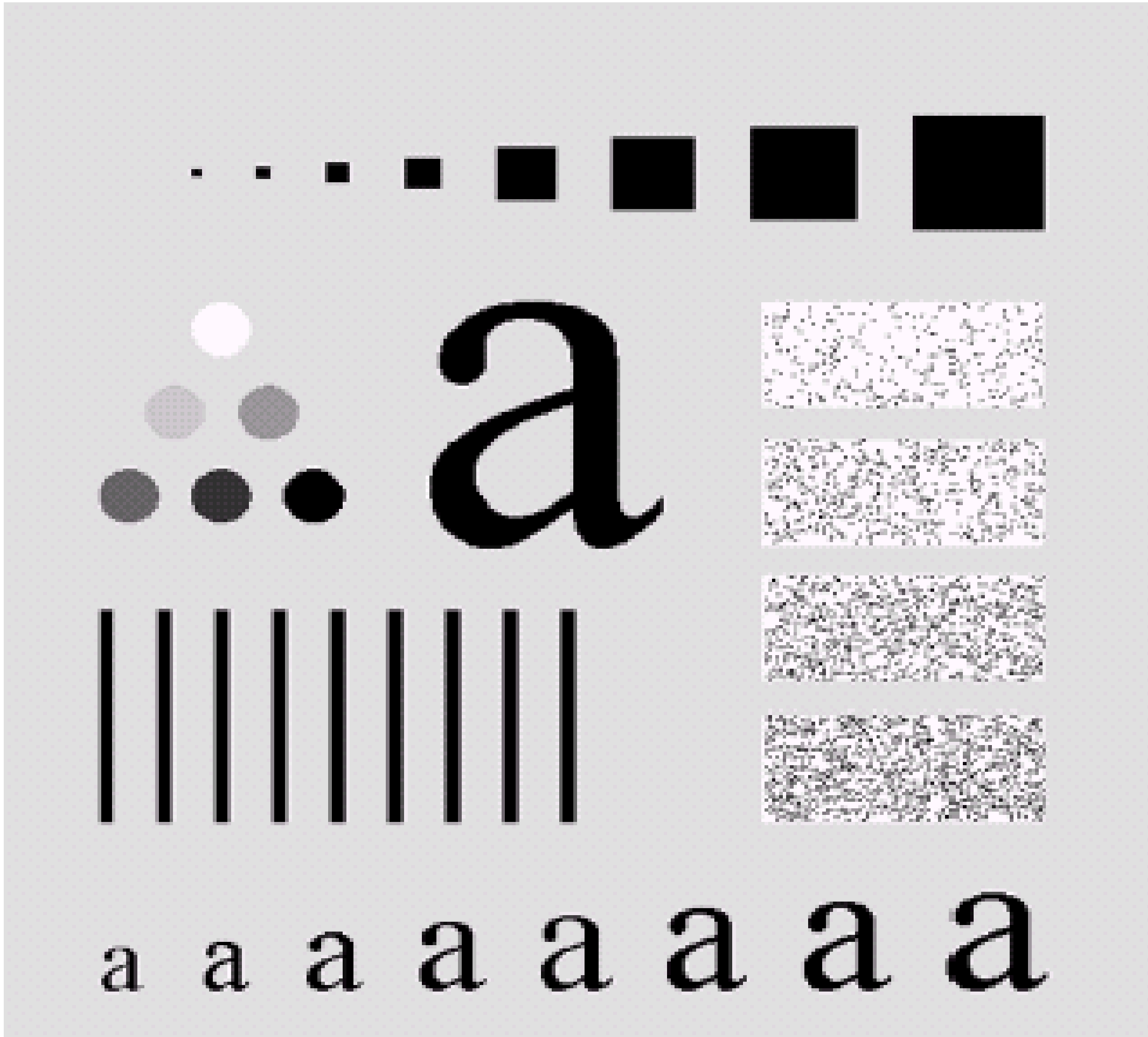
$$D(u,v)=[(u-M/2)^2+(v-N/2)^2]^{1/2}$$

Ideal Low Pass Filter (cont…)



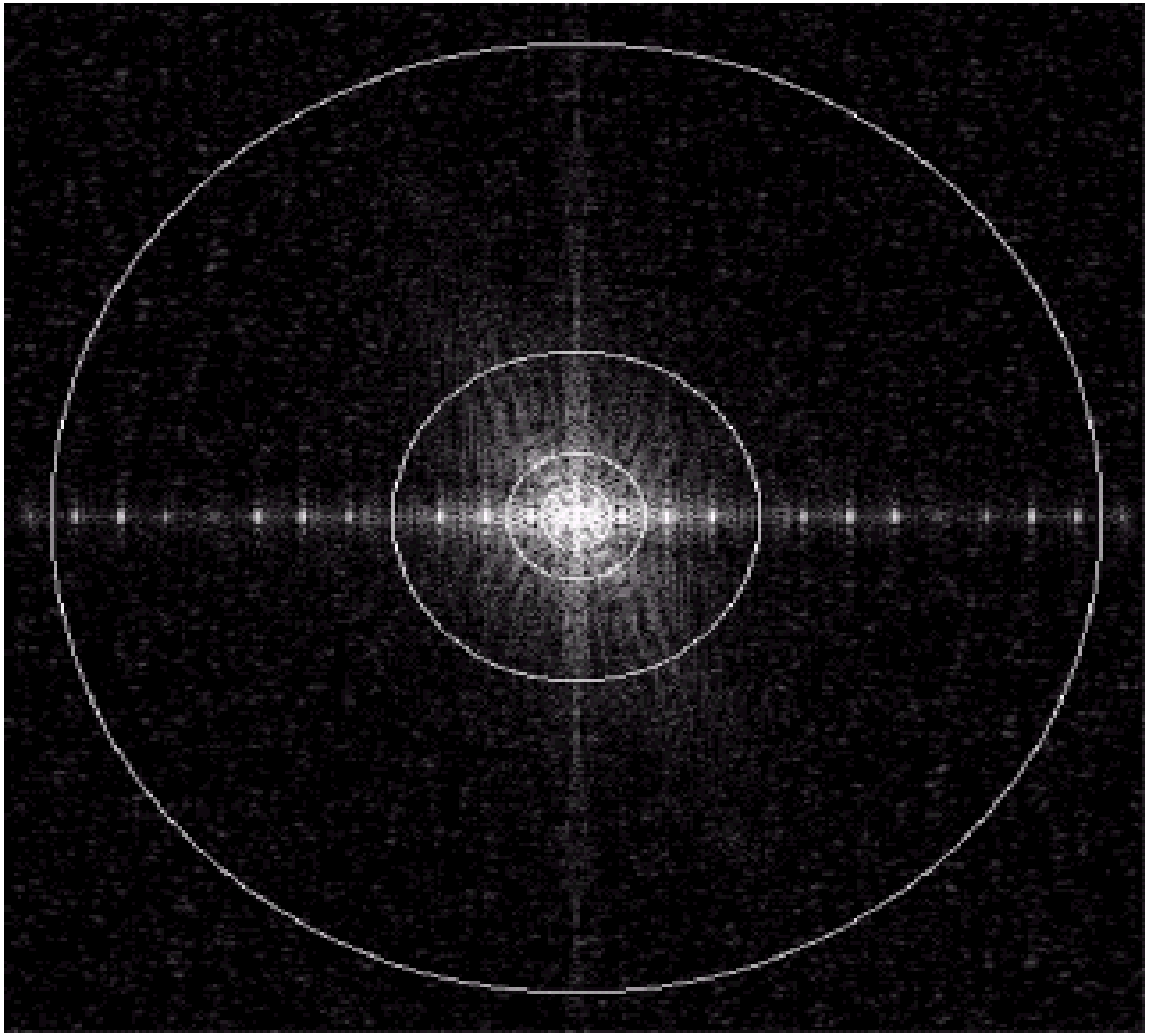
Above we show an image, it's Fourier spectrum and a series of ideal low pass filters of radius 5, 15, 30, 80 and 230 superimposed on top of it

Ideal Low Pass Filter (cont…)



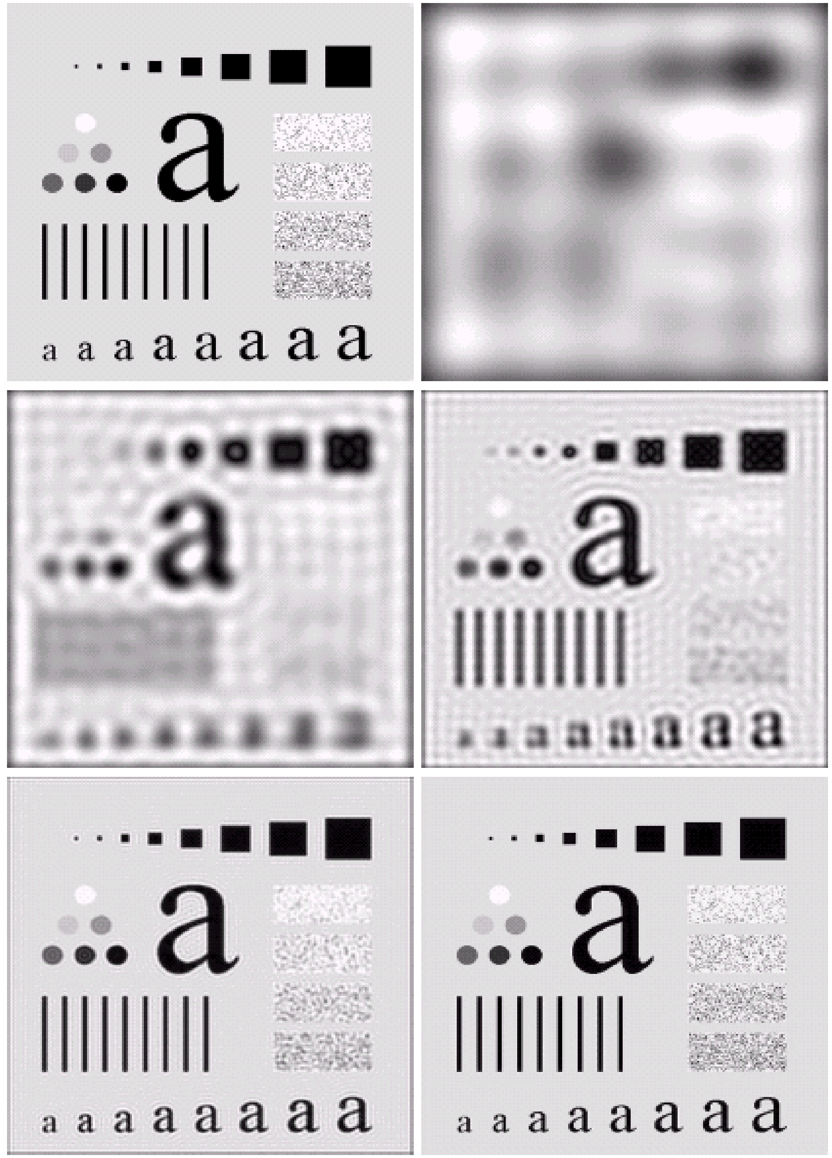
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

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

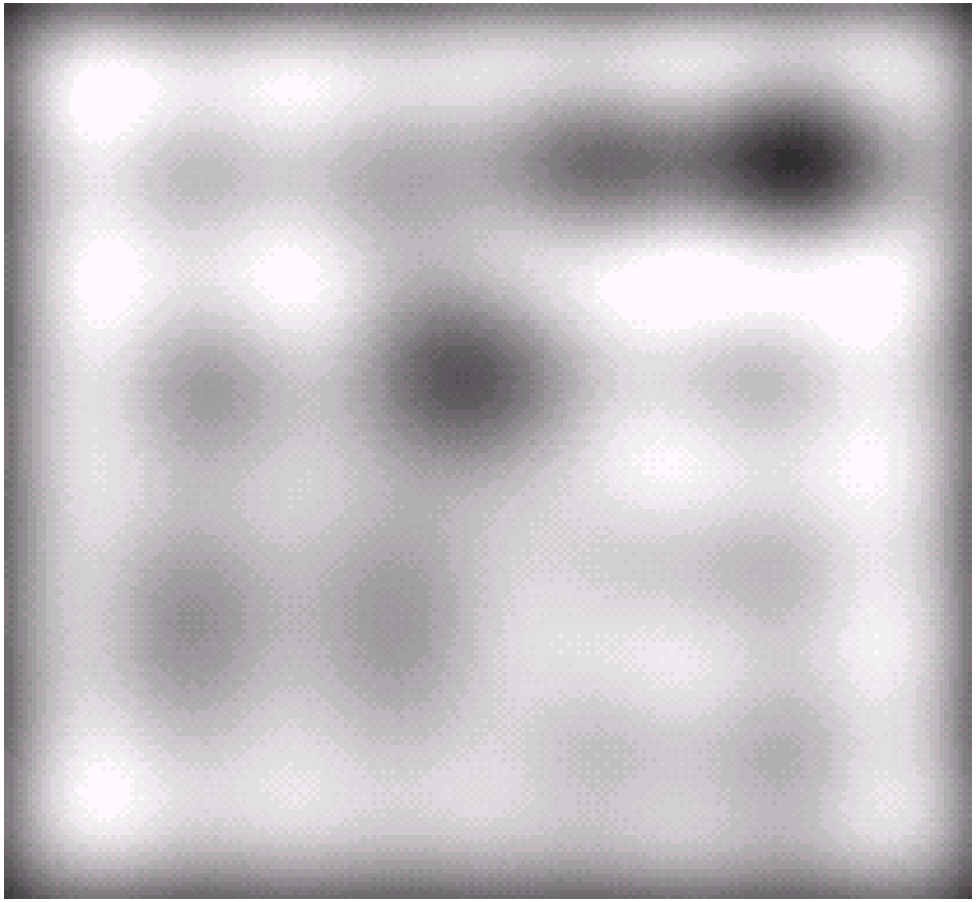
Result of filtering with ideal low pass filter of radius 15

Result of filtering with ideal low pass filter of radius 30

Result of filtering with ideal low pass filter of radius 80

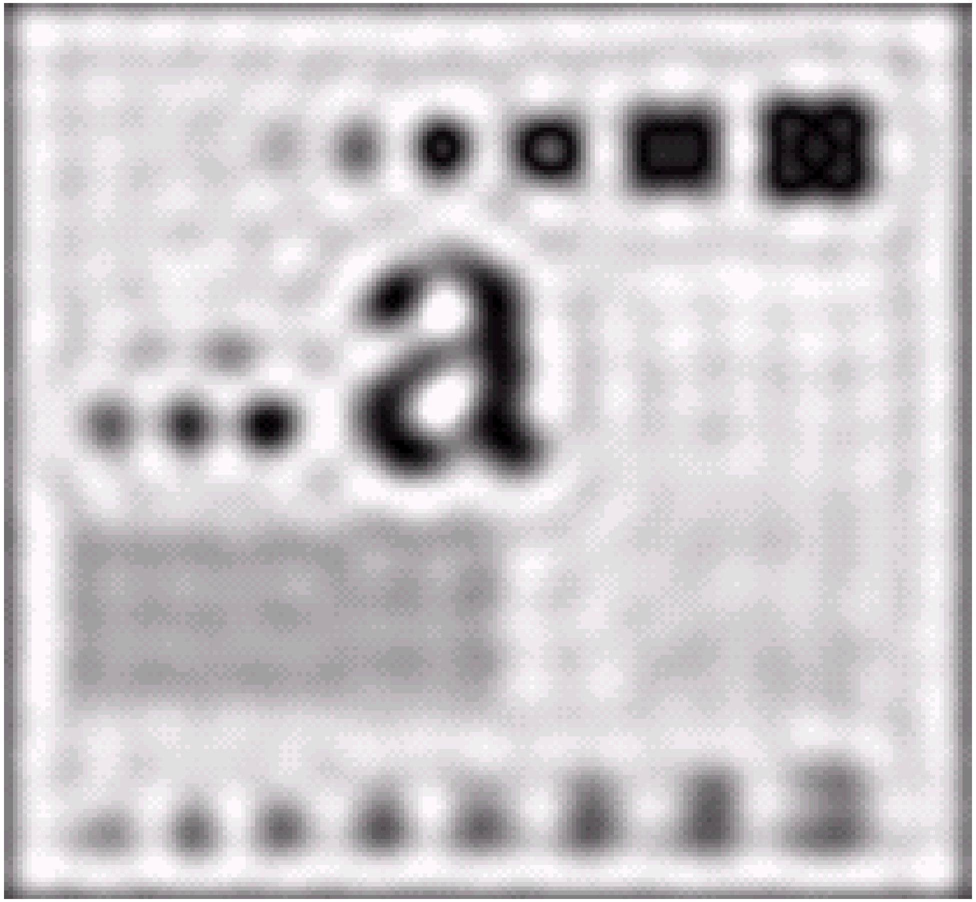
Result of filtering with ideal low pass filter of radius 230

Ideal Low Pass Filter (cont…)



Result of filtering with ideal low pass filter of radius 5

Ideal Low Pass Filter (cont…)

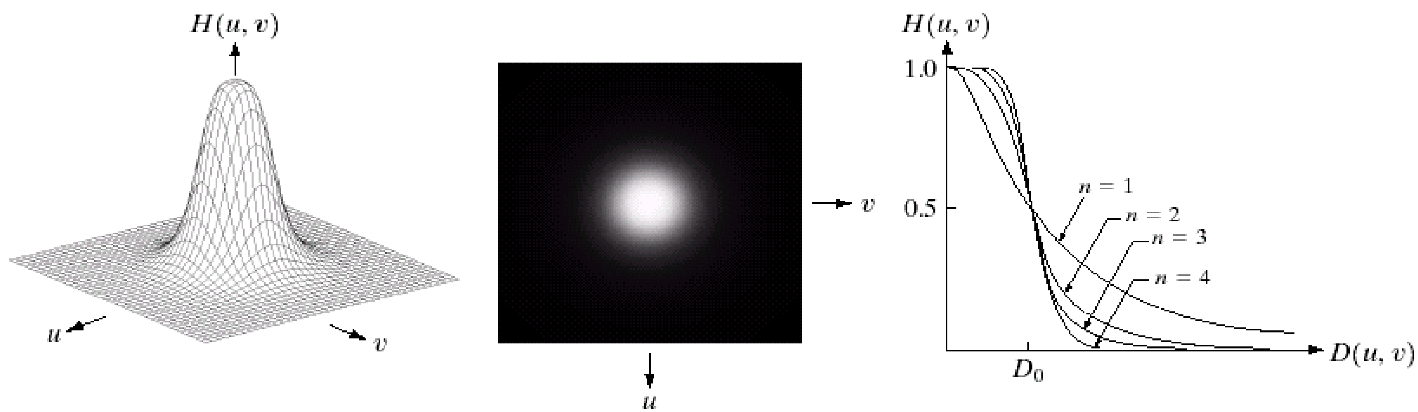


Result of filtering with ideal low pass filter of radius 15

Butterworth Lowpass Filters

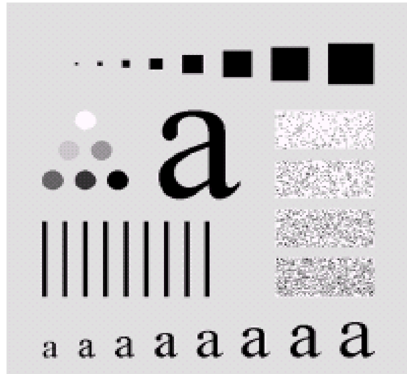
The transfer function of a Butterworth lowpass filter of order n with cutoff frequency at distance D_0 from the origin is defined as:

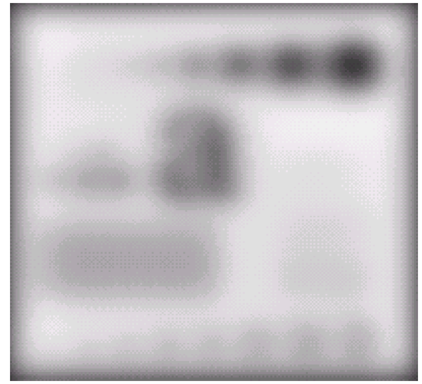
$$H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^{2n}}$$



Butterworth Lowpass Filter (cont…)

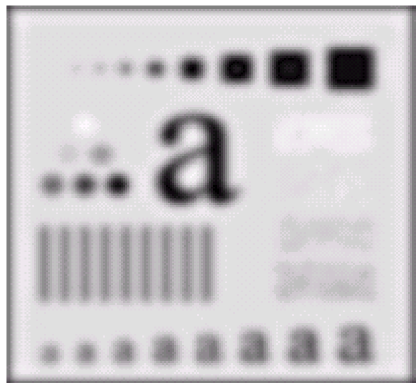
Original image

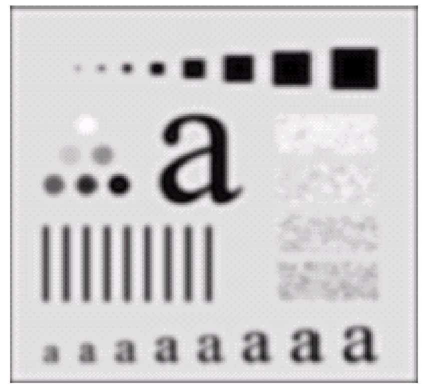




Result of filtering with Butterworth filter of order 2 and cutoff radius 5

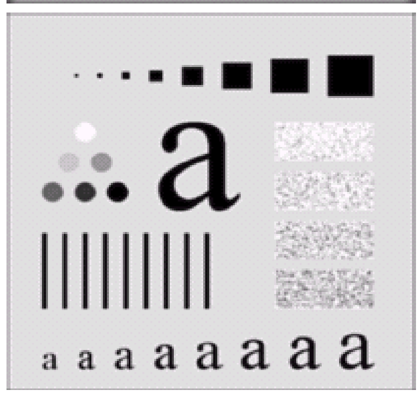
Result of filtering with Butterworth filter of order 2 and cutoff radius 15

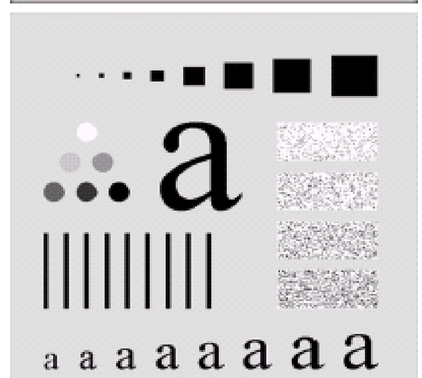




Result of filtering with Butterworth filter of order 2 and cutoff radius 30

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



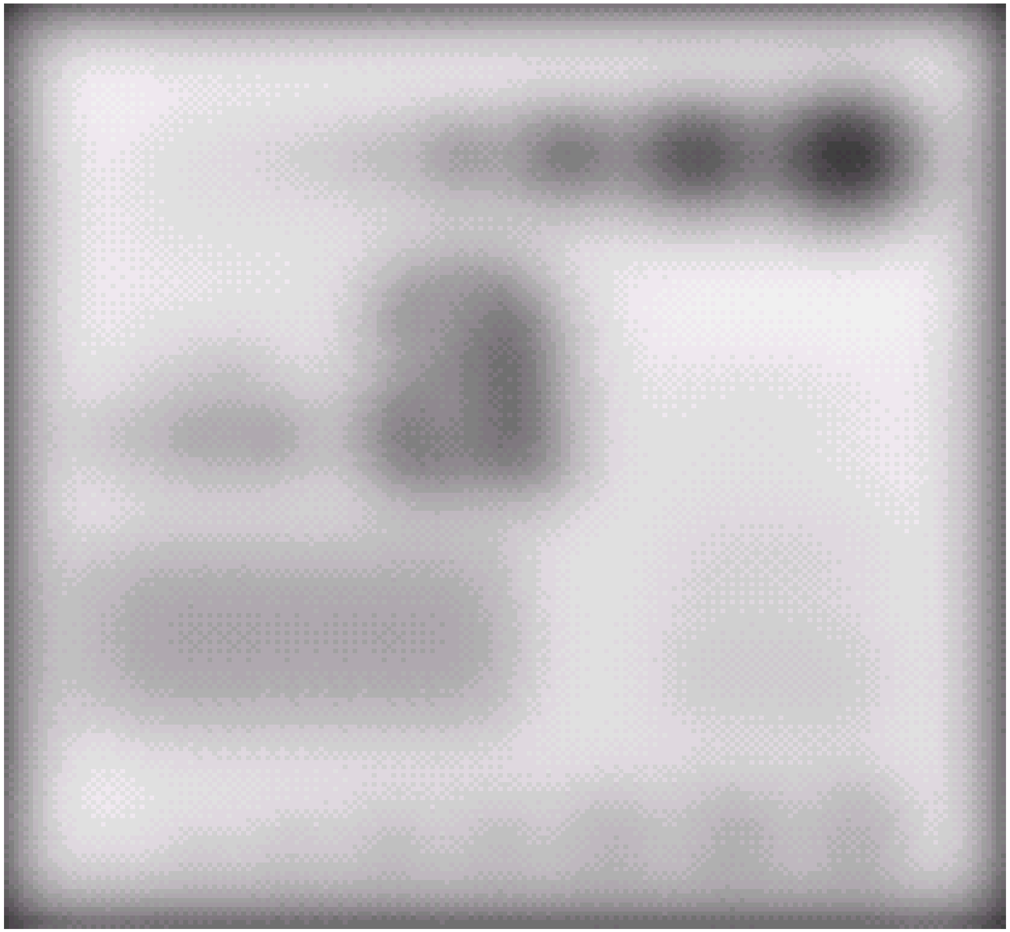


Result of filtering with Butterworth filter of order 2 and cutoff radius 230

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

Butterworth Lowpass Filter (cont…)

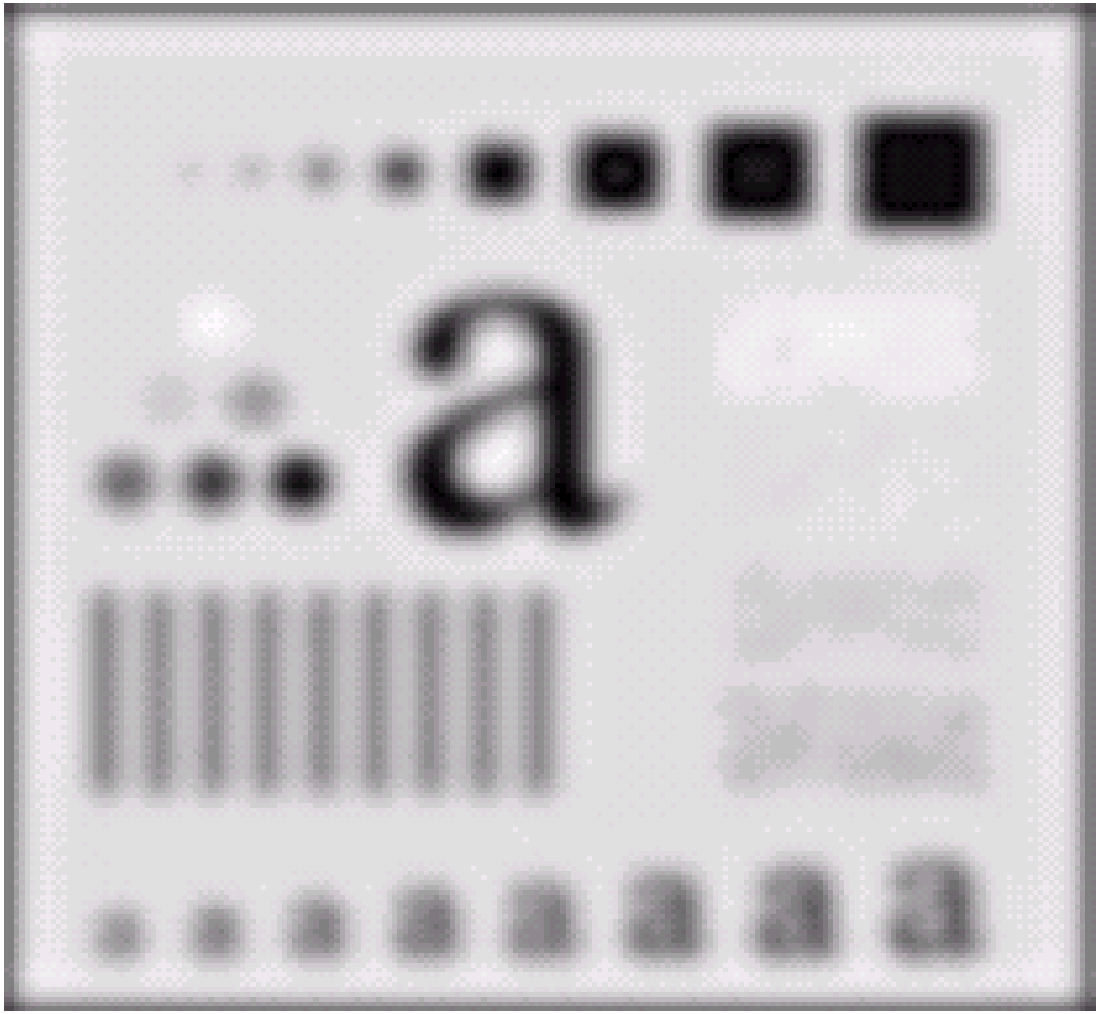
Images taken from Gonzalez & Woods, Digital Image Processing (2002)



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

Butterworth Lowpass Filter (cont…)

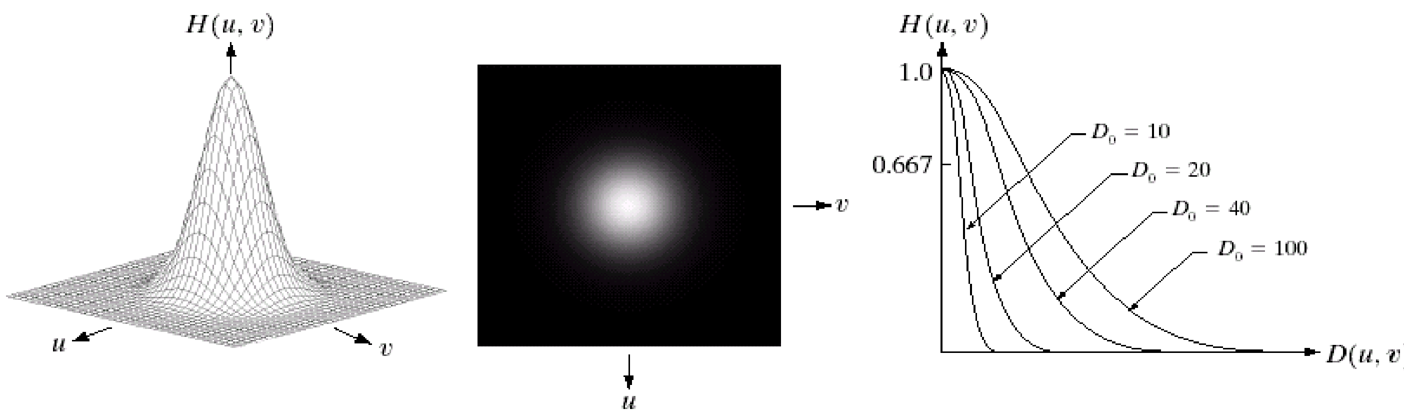
Result of filtering with Butterworth filter of order 2 and cutoff radius 15



Gaussian Lowpass Filters

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

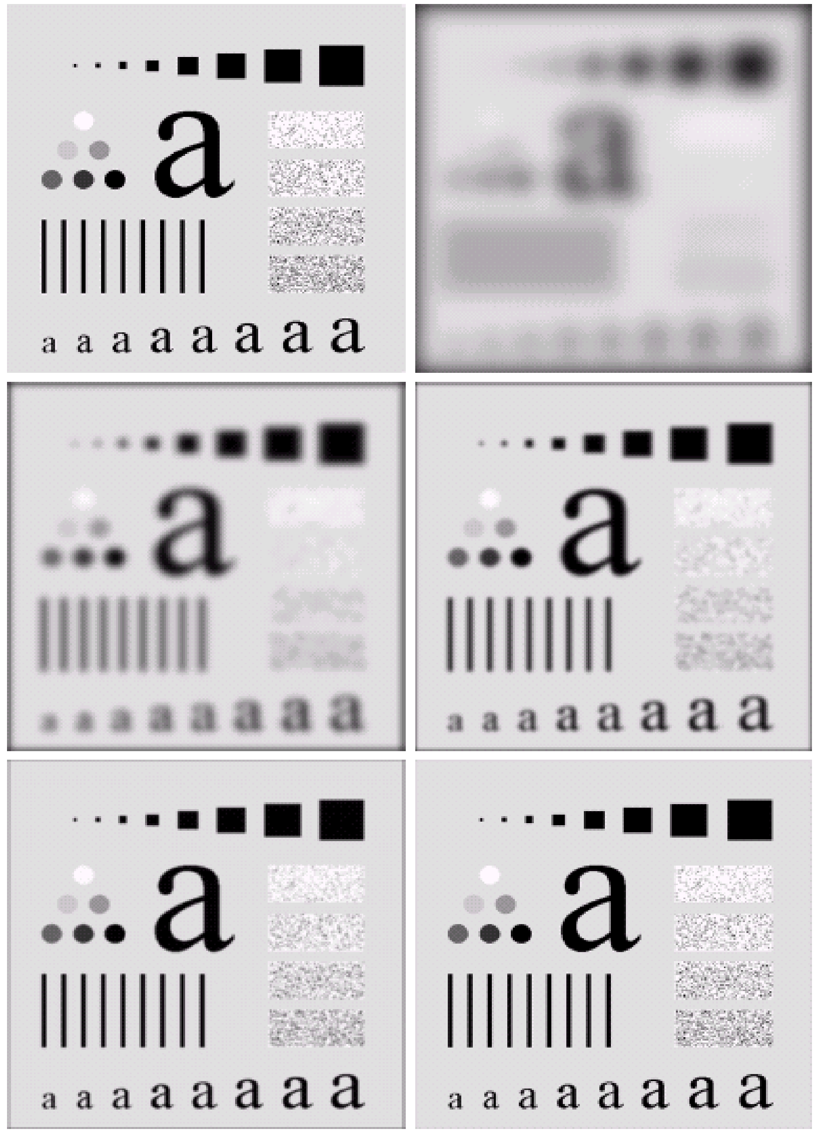
$$H(u,v) = e^{-D^2(u,v)/2D_0^{\,2}}$$



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

Gaussian Lowpass Filters (cont…)

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



Original image

Result of filtering with Gaussian filter with cutoff radius 5

Result of filtering with Gaussian filter with cutoff radius 15

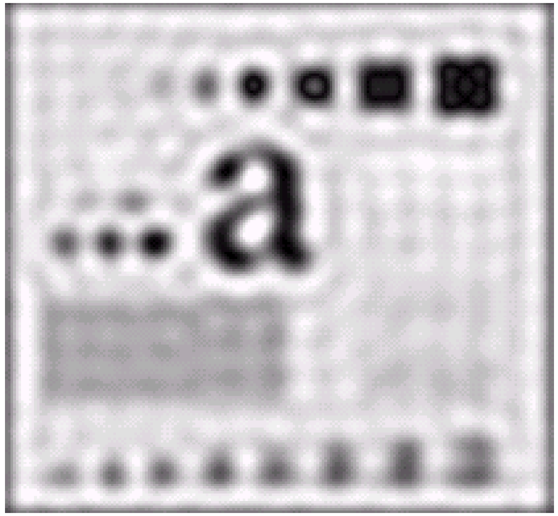
Result of filtering with Gaussian filter with cutoff radius 30

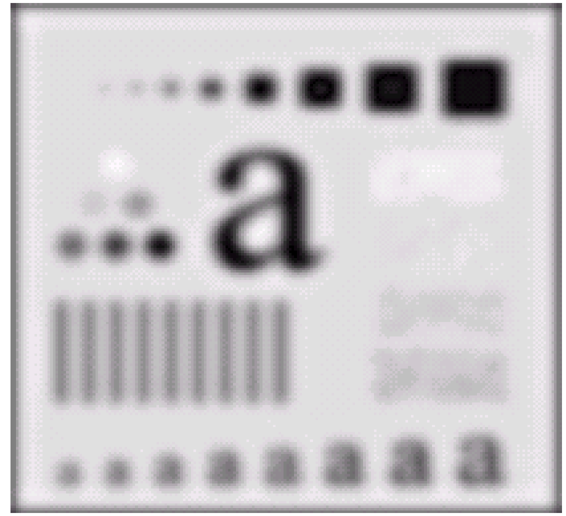
Result of filtering with Gaussian filter with cutoff radius 85

Result of filtering with Gaussian filter with cutoff radius 230

Lowpass Filters Compared

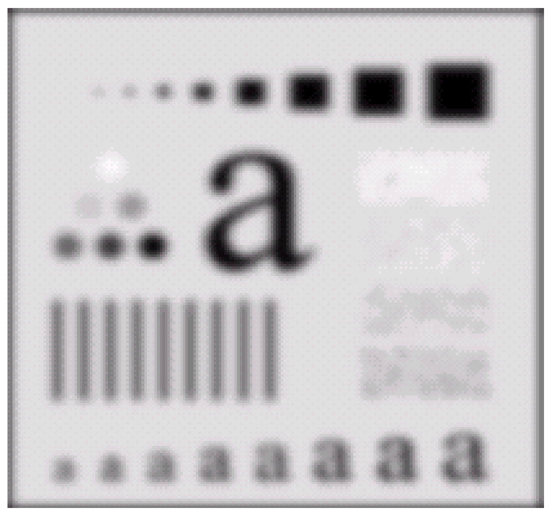
Result of filtering with ideal low pass filter of radius 15





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

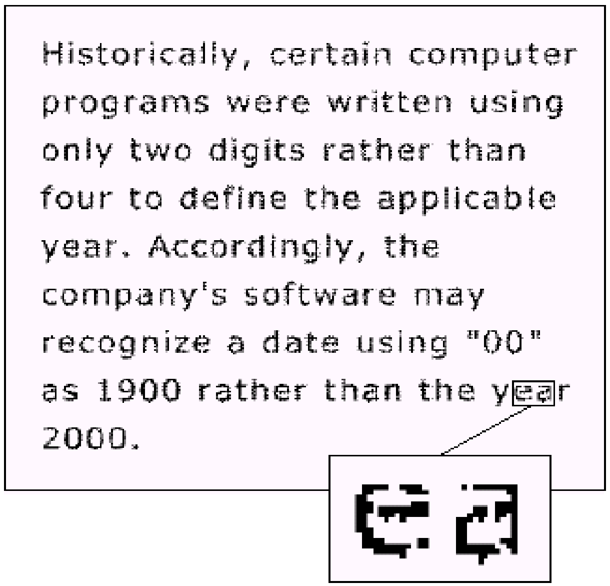
Result of filtering with Gaussian filter with cutoff radius 15

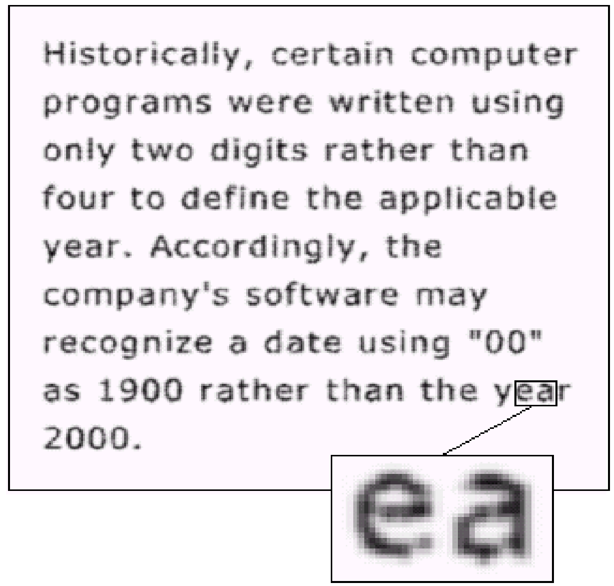


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Lowpass Filtering Examples

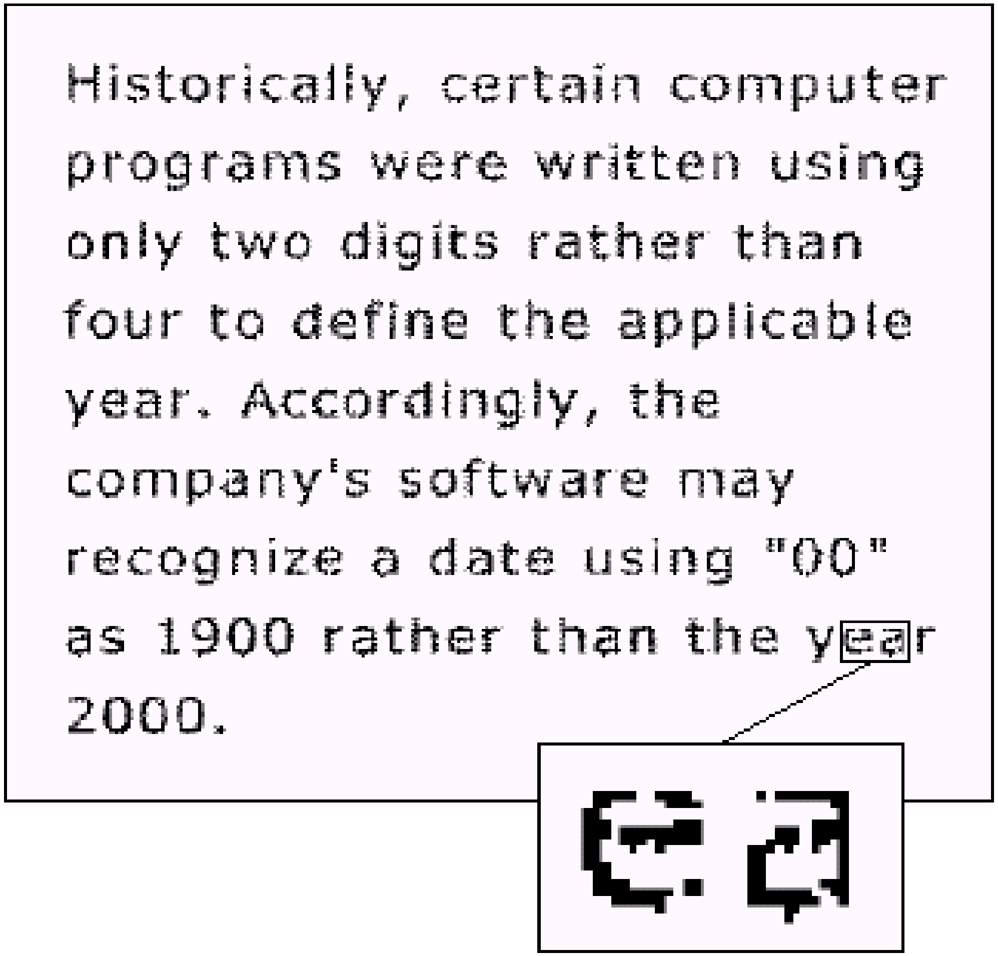
A low pass Gaussian filter is used to connect broken text



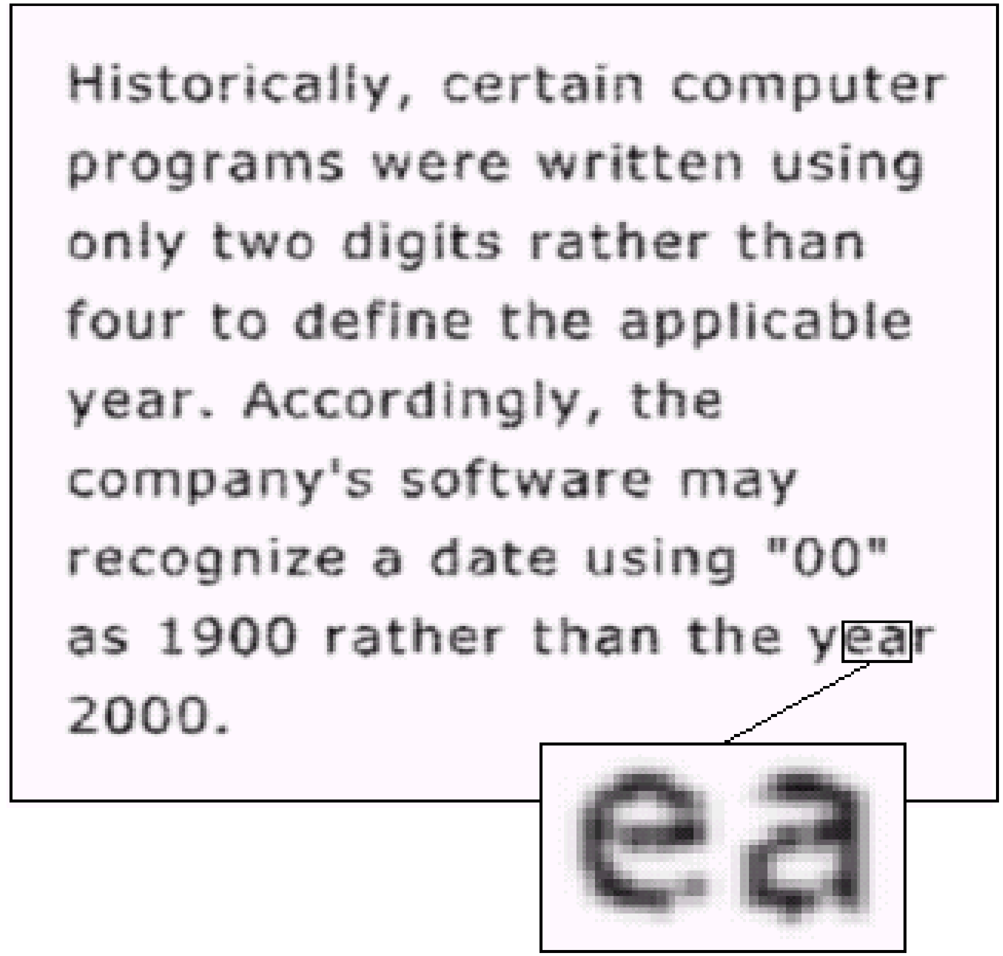


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

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)

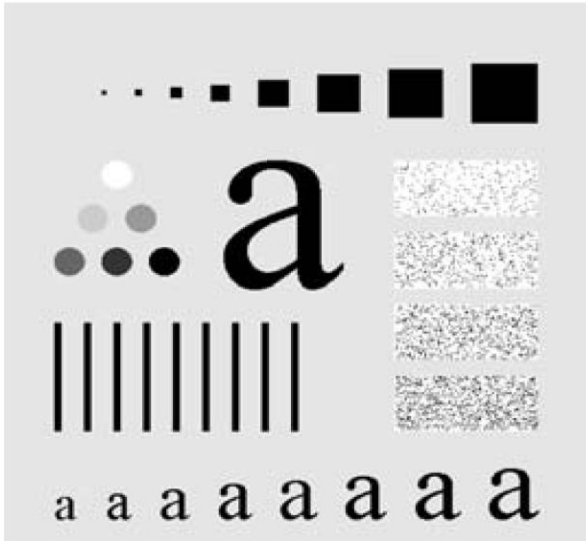
Lowpass Filtering Examples (cont…)

Different lowpass Gaussian filters used to remove blemishes in a photograph

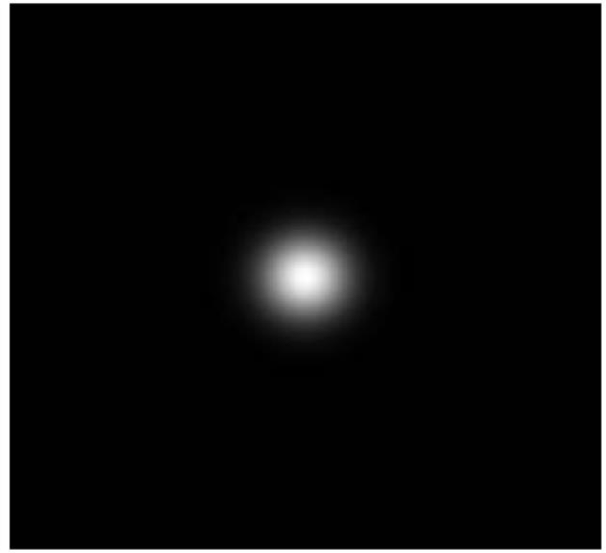
Lowpass Filtering Examples (cont…)

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

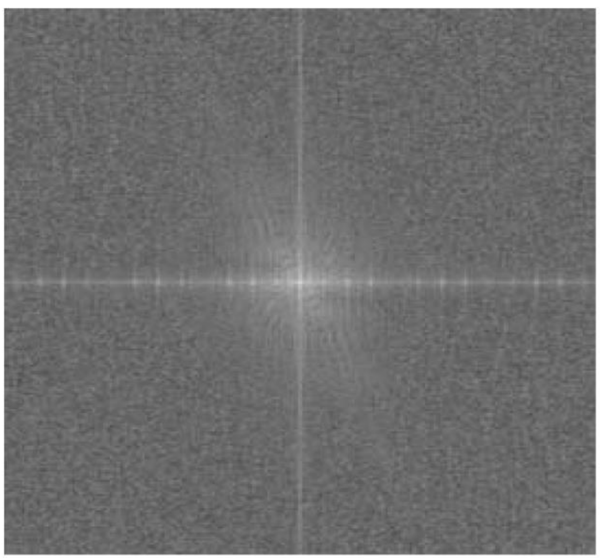
Lowpass Filtering Examples (cont…)



Original image



Gaussian lowpass filter



Spectrum of original image

Processed image

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:

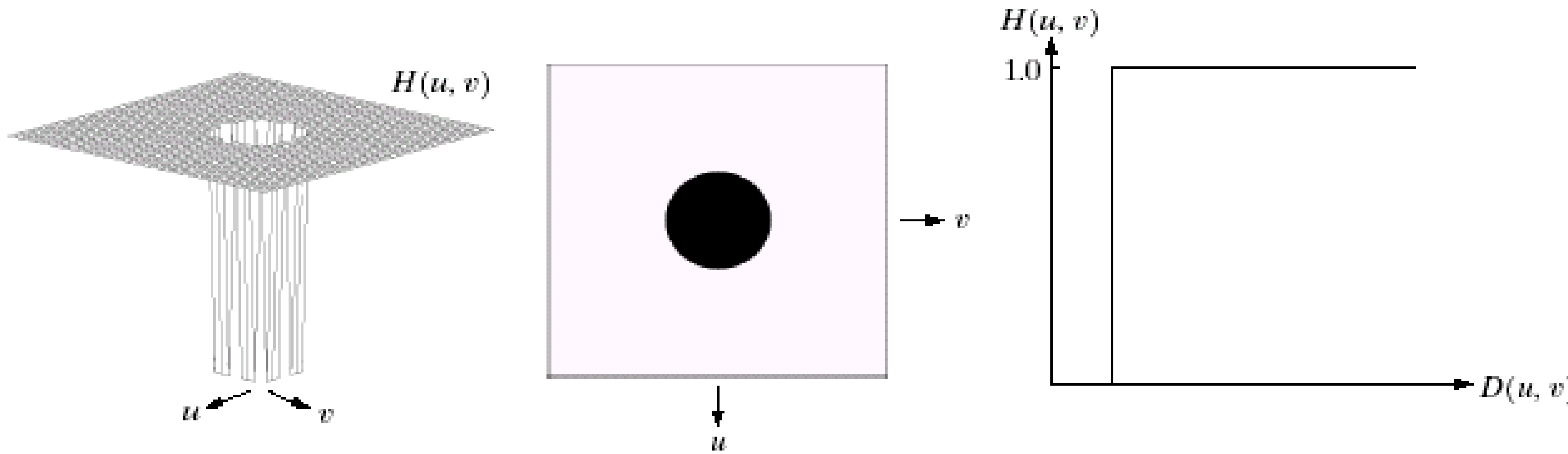
$$H_{hp}(u, v) = 1 - H_{lp}(u, v)$$

Ideal High Pass Filters

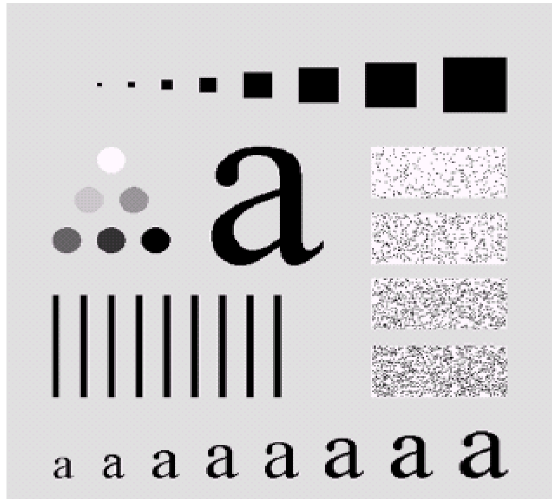
The ideal high pass filter is given as:

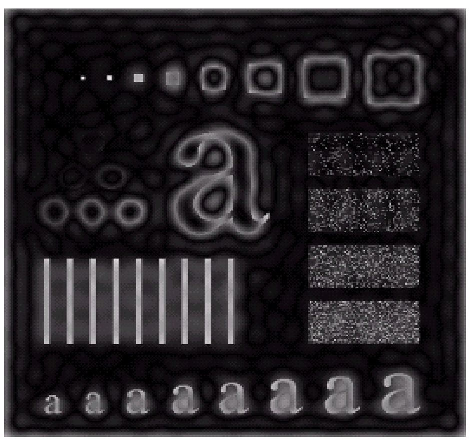
$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \leq D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

where D_0 is the cut off distance as before

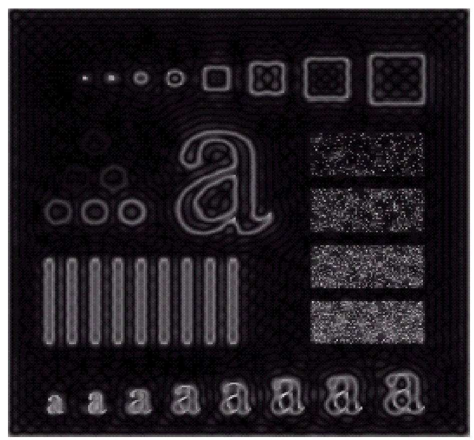


Ideal High Pass Filters (cont…)

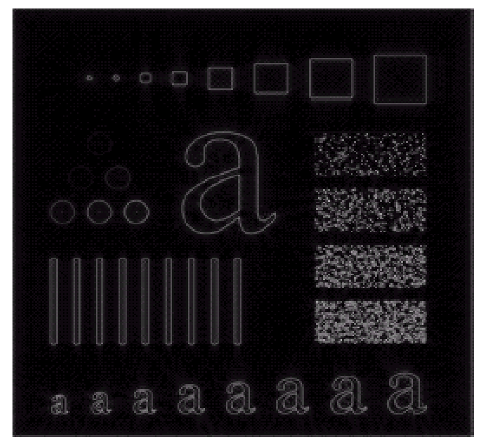




Results of ideal high pass filtering with $D_0 = 15$



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



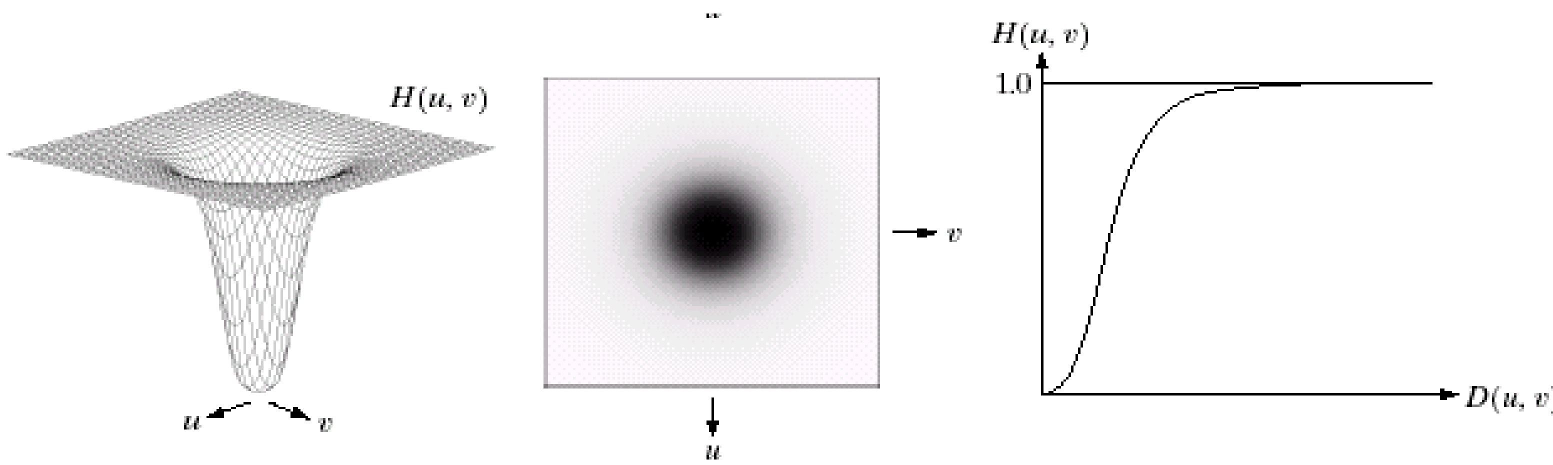
Results of ideal high pass filtering with $D_0 = 80$

Butterworth High Pass Filters

The Butterworth high pass filter is given as:

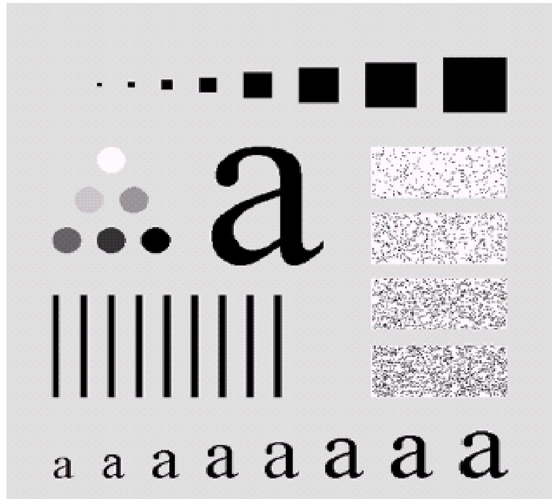
$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

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

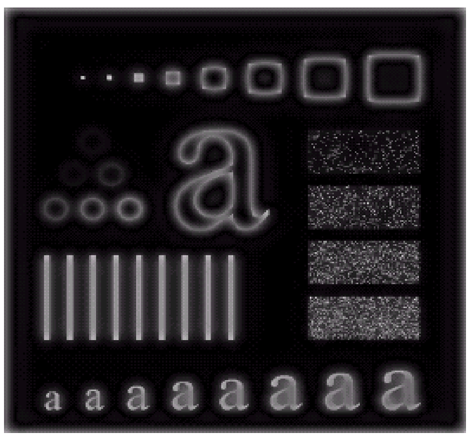


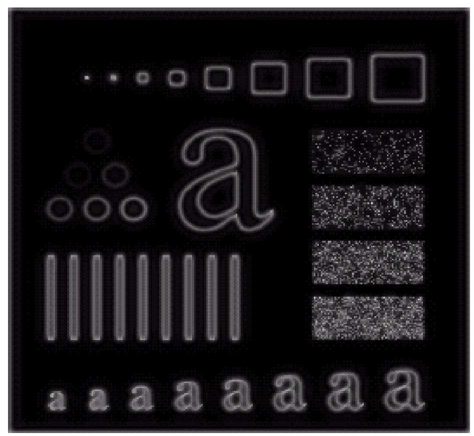
Butterworth High Pass Filters (cont…)

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

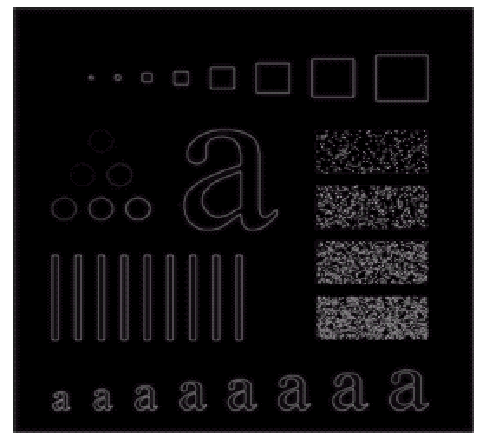


Results of Butterworth high pass filtering of order 2 with $D_0 = 15$





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



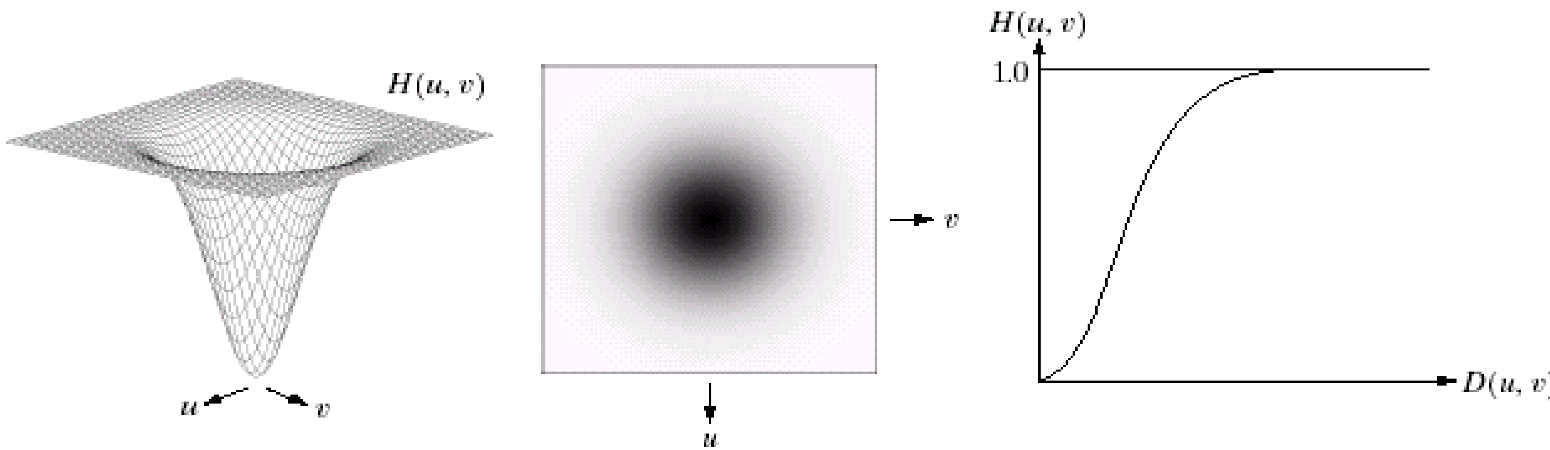
Results of Butterworth high pass filtering of order 2 with $D_0 = 80$

Gaussian High Pass Filters

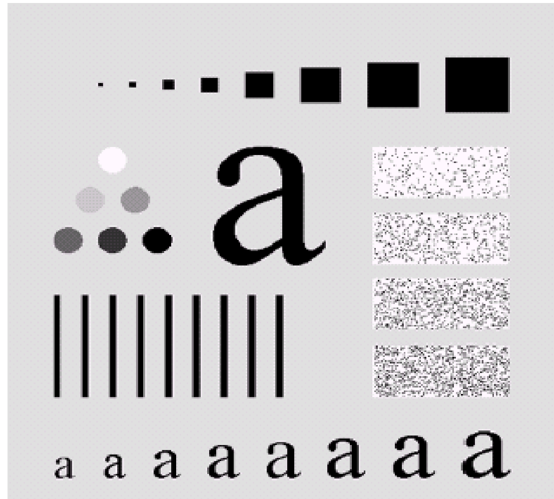
The Gaussian high pass filter is given as:

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

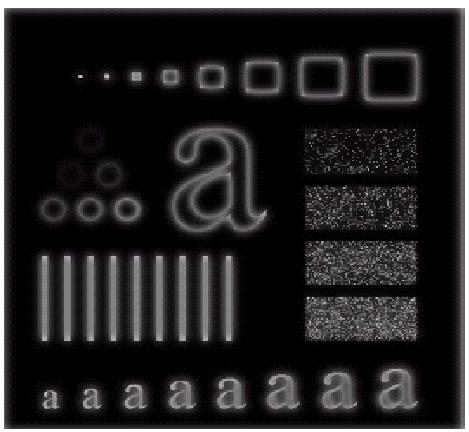
where D_0 is the cut off distance as before

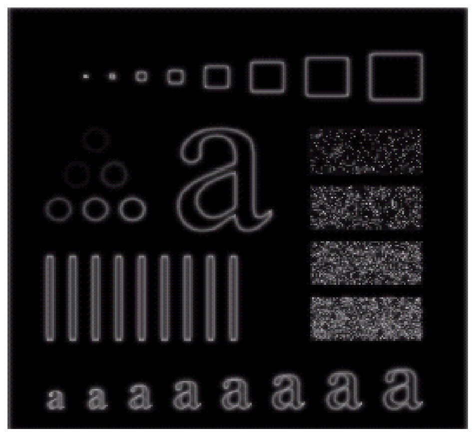


Gaussian High Pass Filters (cont…)



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

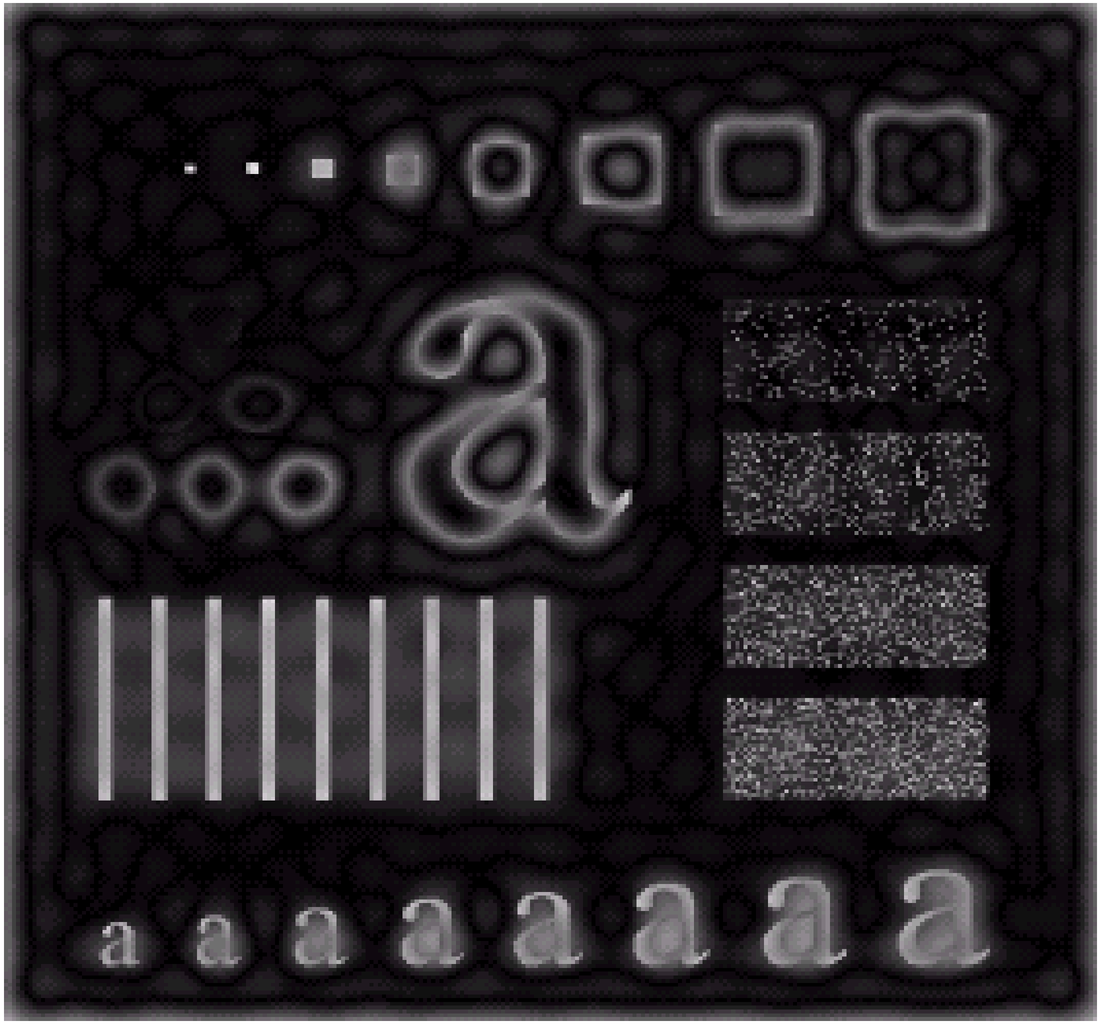




Results of Gaussian high pass filtering with $D_0 = 30$

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

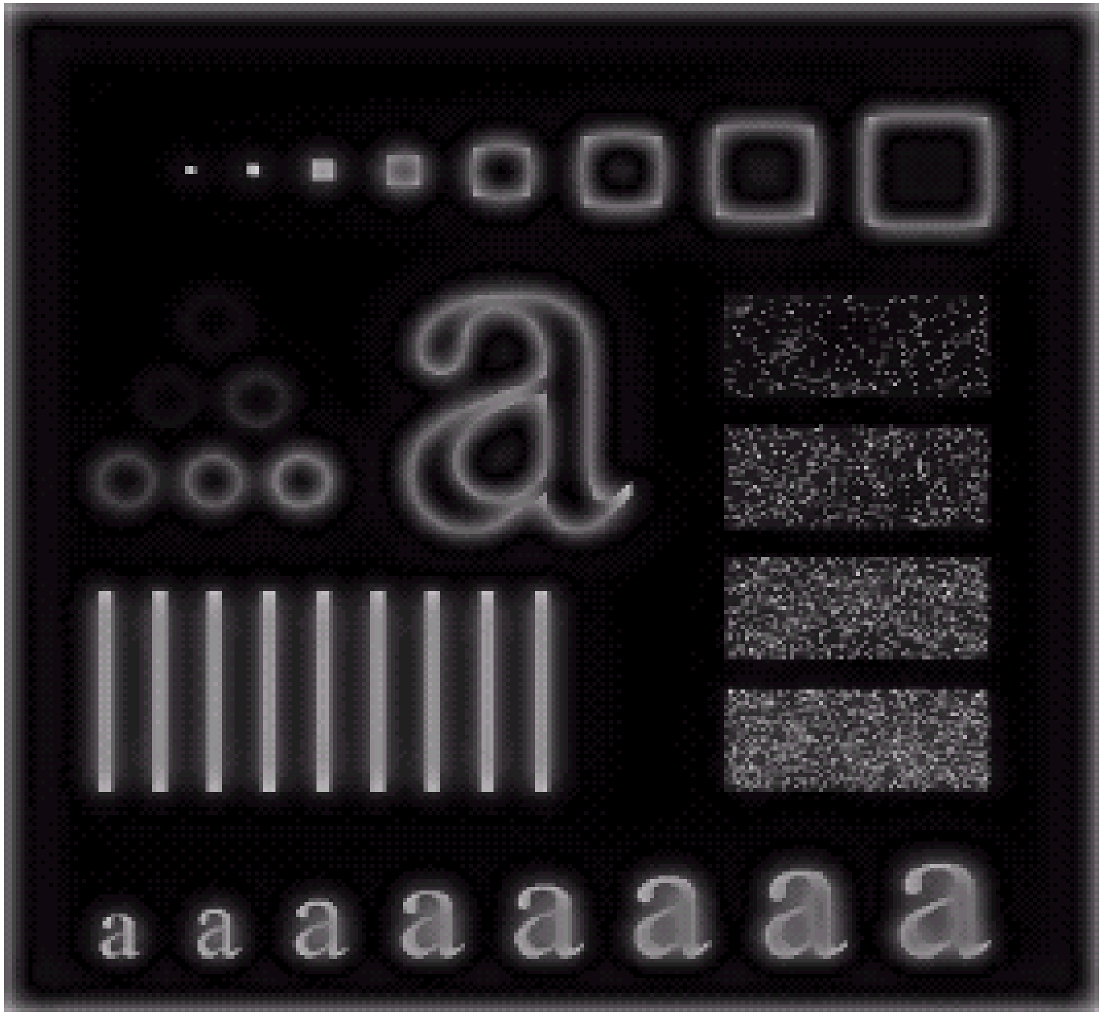
Highpass Filter Comparison



Results of ideal high pass filtering with D_0 = 15

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

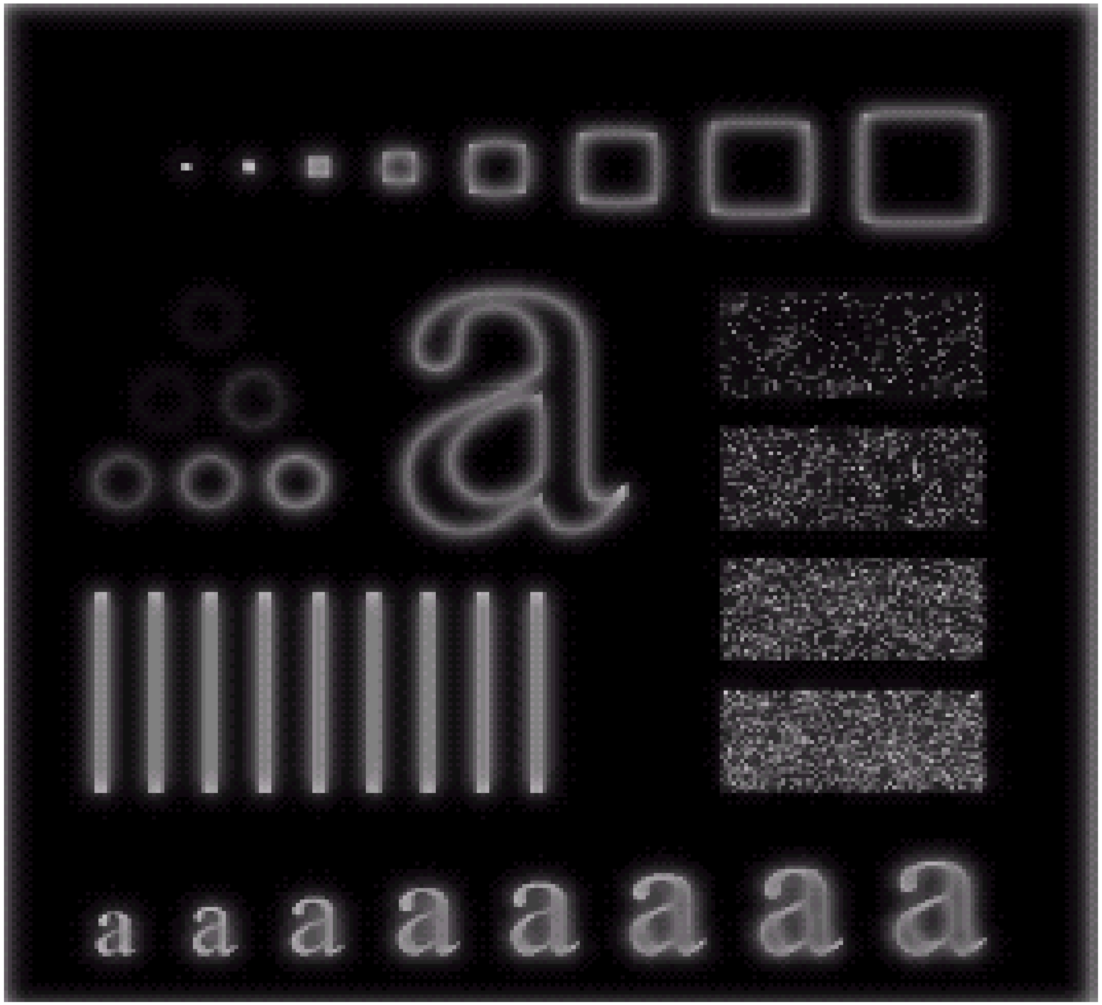
Highpass Filter Comparison



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

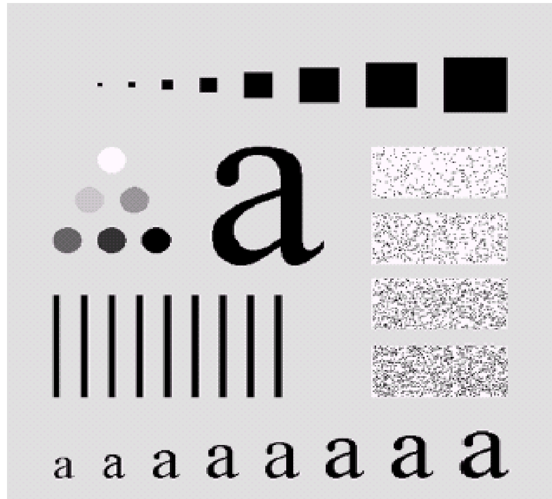
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

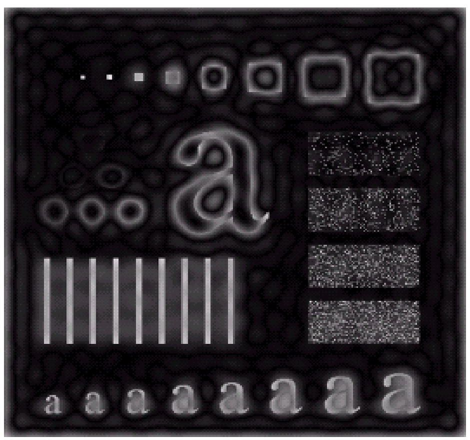
Highpass Filter Comparison



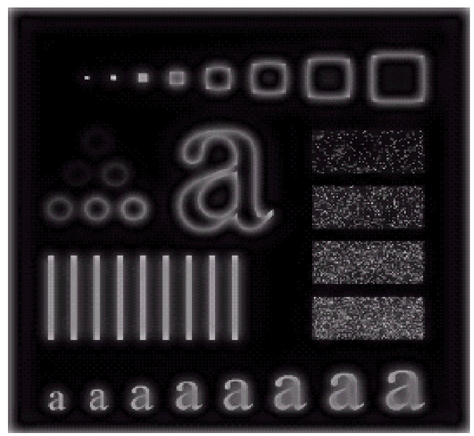
Results of Gaussian high pass filtering with $D_0 = 15$

Highpass Filter Comparison

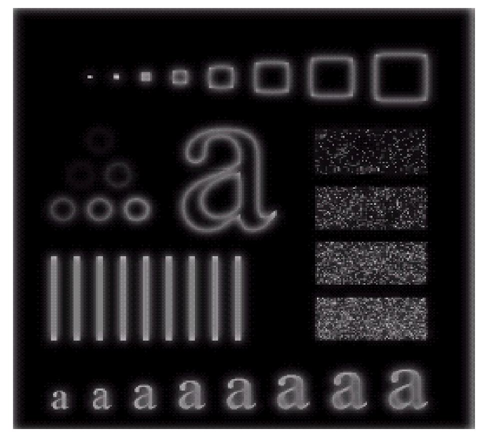




Results of ideal high pass filtering with $D_0 = 15$

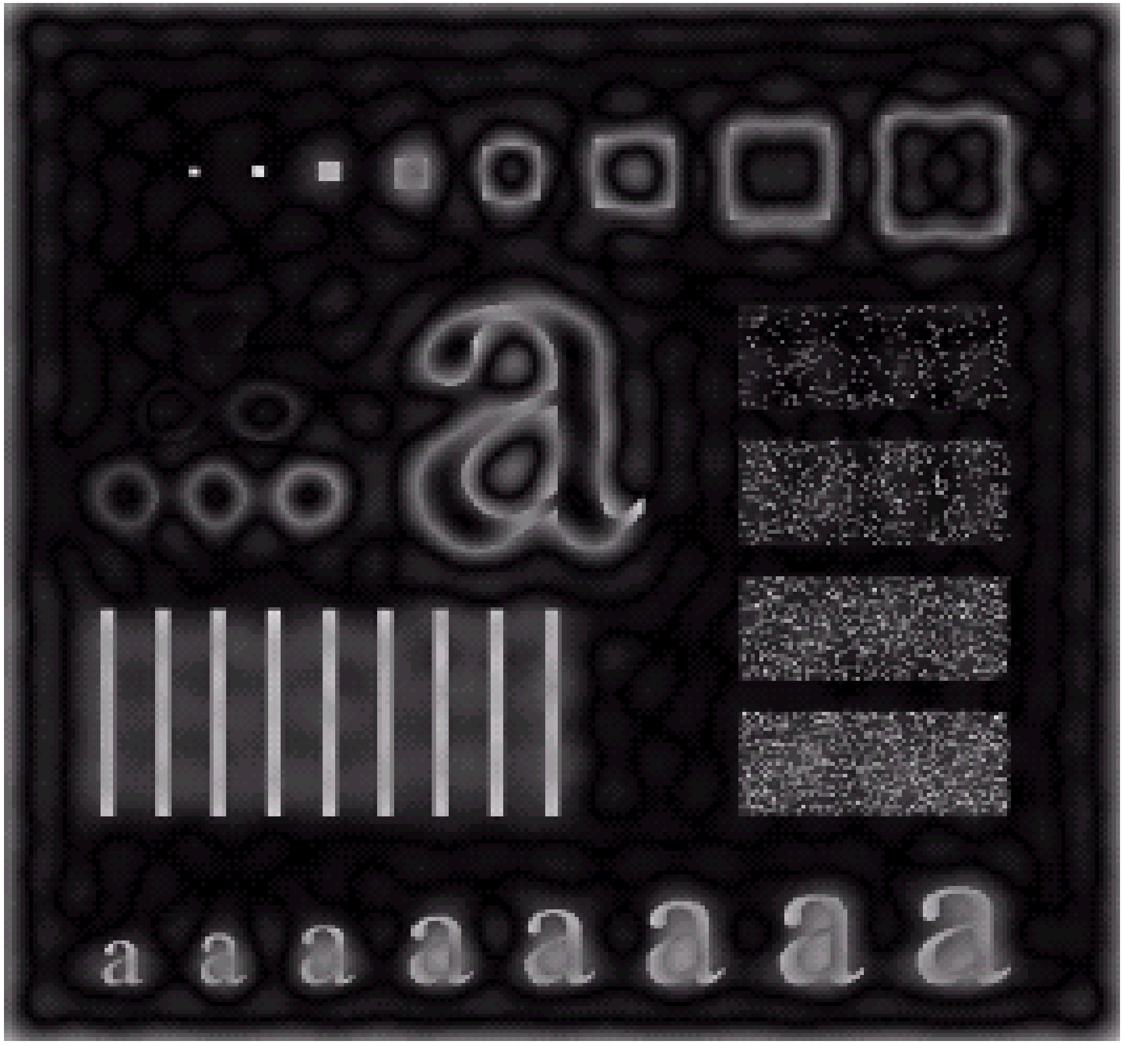


Results of Butterworth high pass filtering of order 2 with $D_0 = 15$



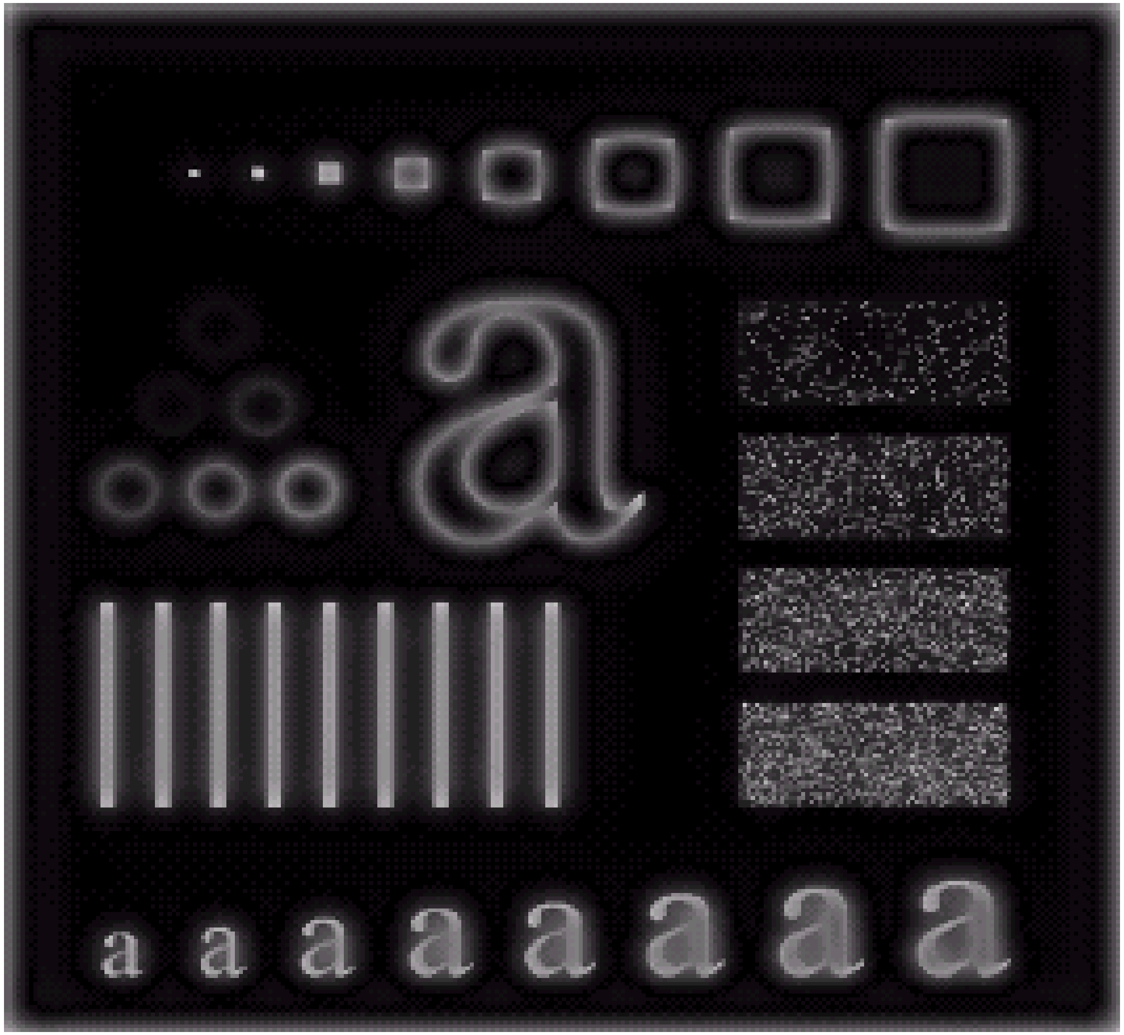
Results of Gaussian high pass filtering with $D_0 = 15$

Highpass Filter Comparison



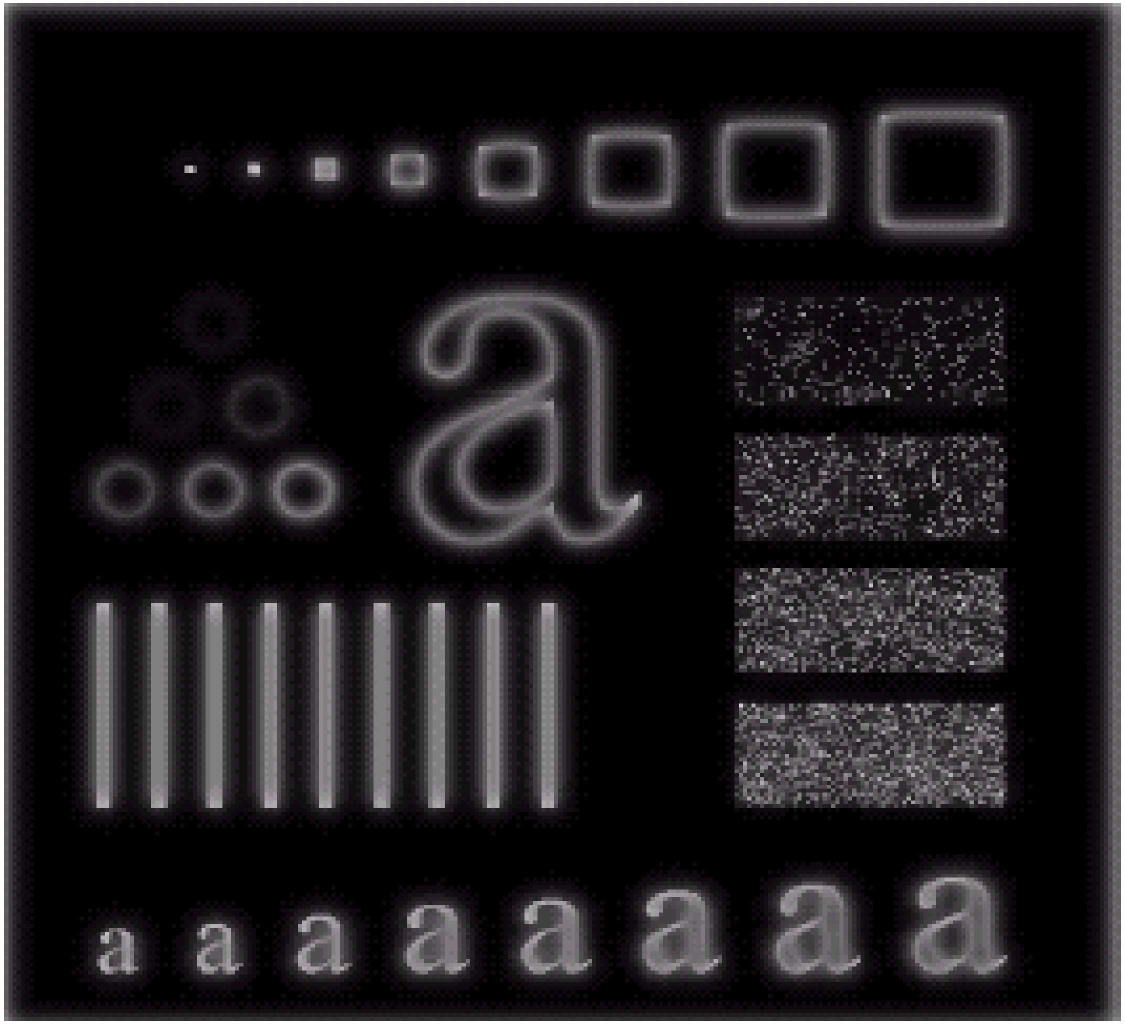
Results of ideal high pass filtering with D_0 = 15

Highpass Filter Comparison



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

Highpass Filter Comparison

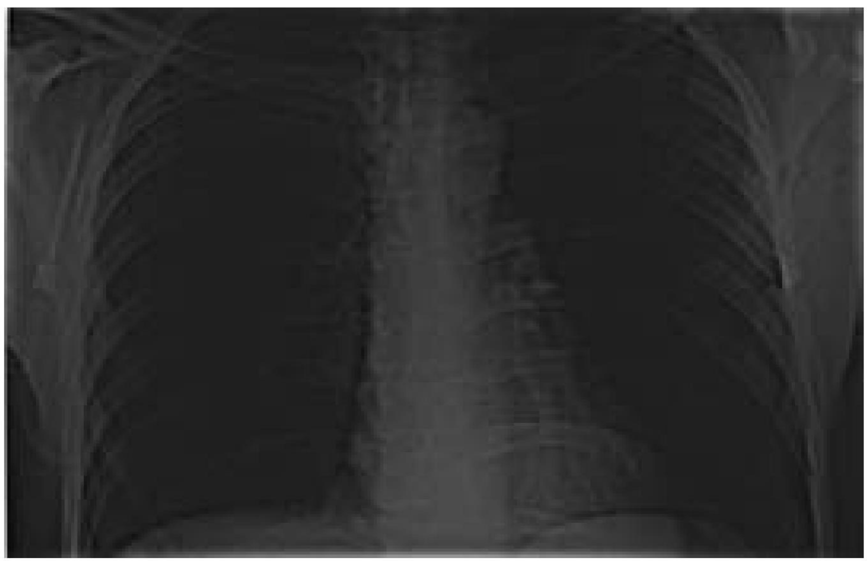


Results of Gaussian high pass filtering with D_0 = 15

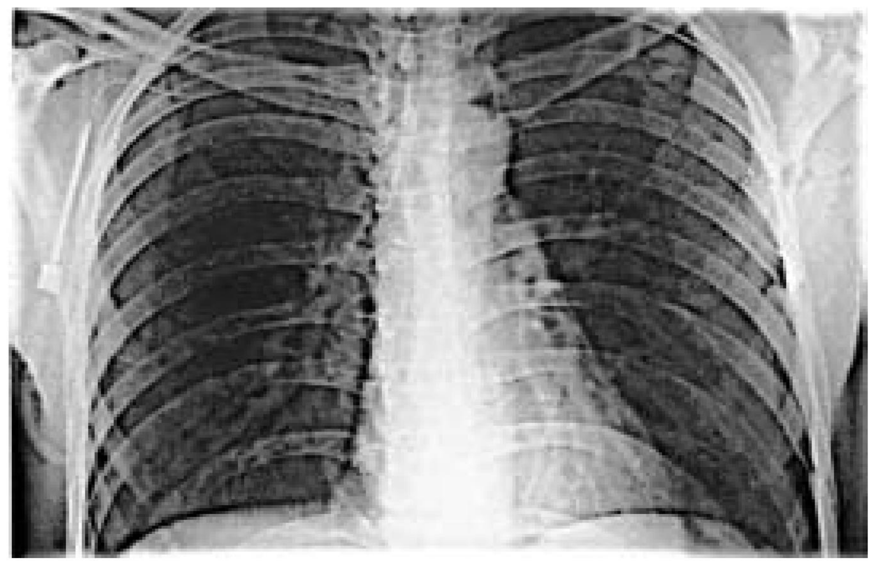
Highpass Filtering Example

Original image

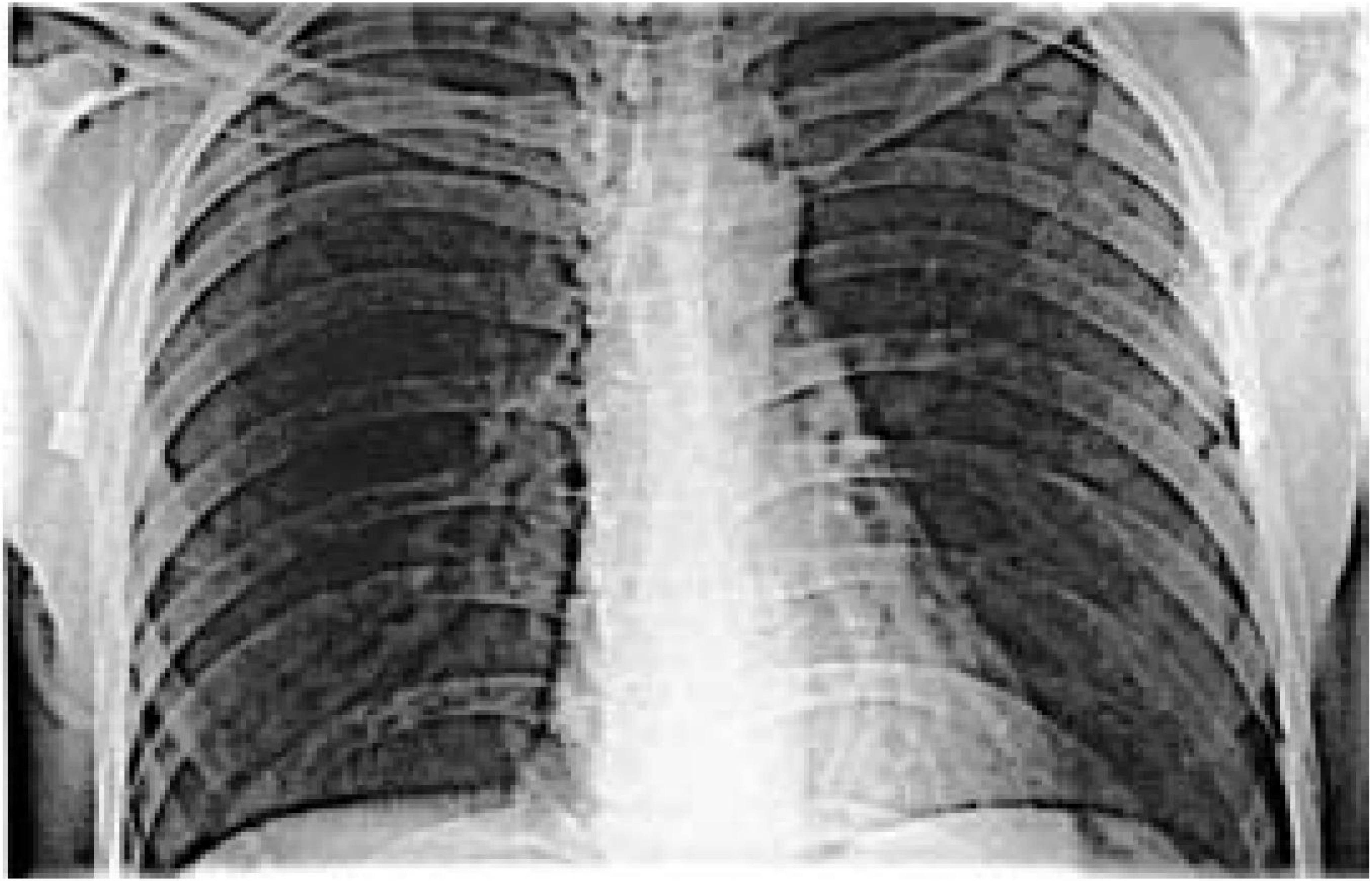
Highpass filtering result



High frequency emphasis result



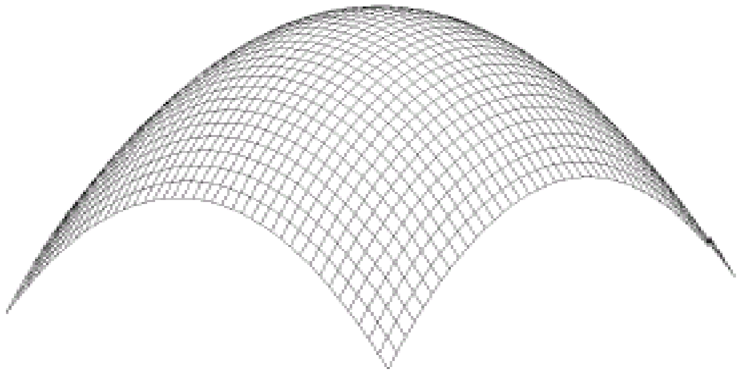
After histogram equalisation

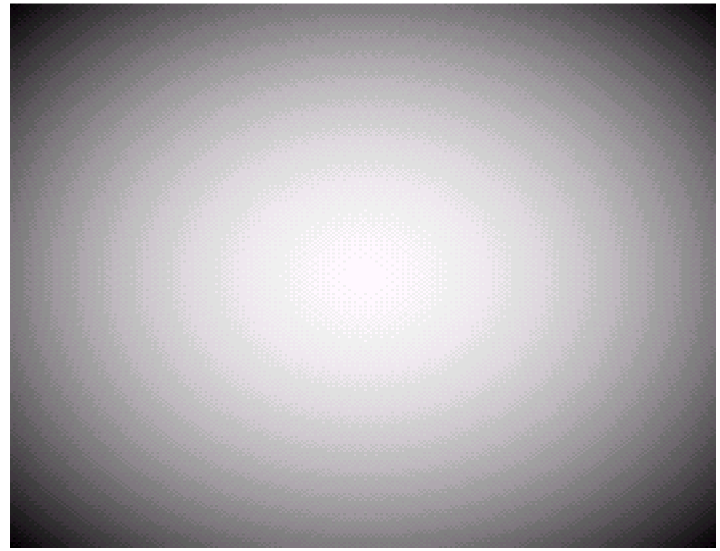


Laplacian In The Frequency Domain

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

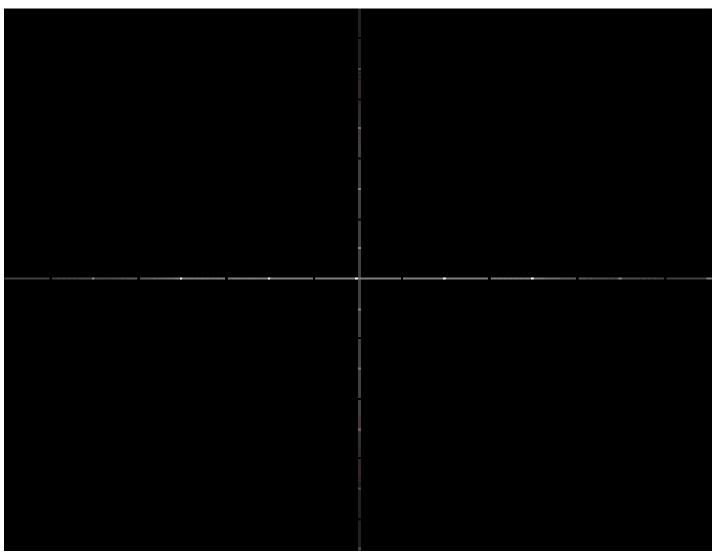
Laplacian in the frequency domain

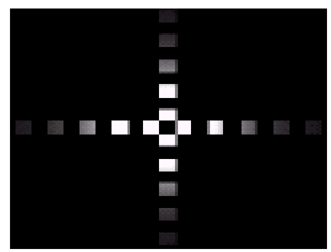




2-D image of Laplacian in the frequency domain

Inverse DFT of Laplacian in the frequency domain

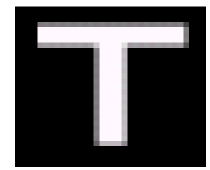




0	1	0
1	−4	1
0	1	0

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

Imteresting Application Of Frequency Domain Filtering



Interesting Application Of Frequency Domain Filtering