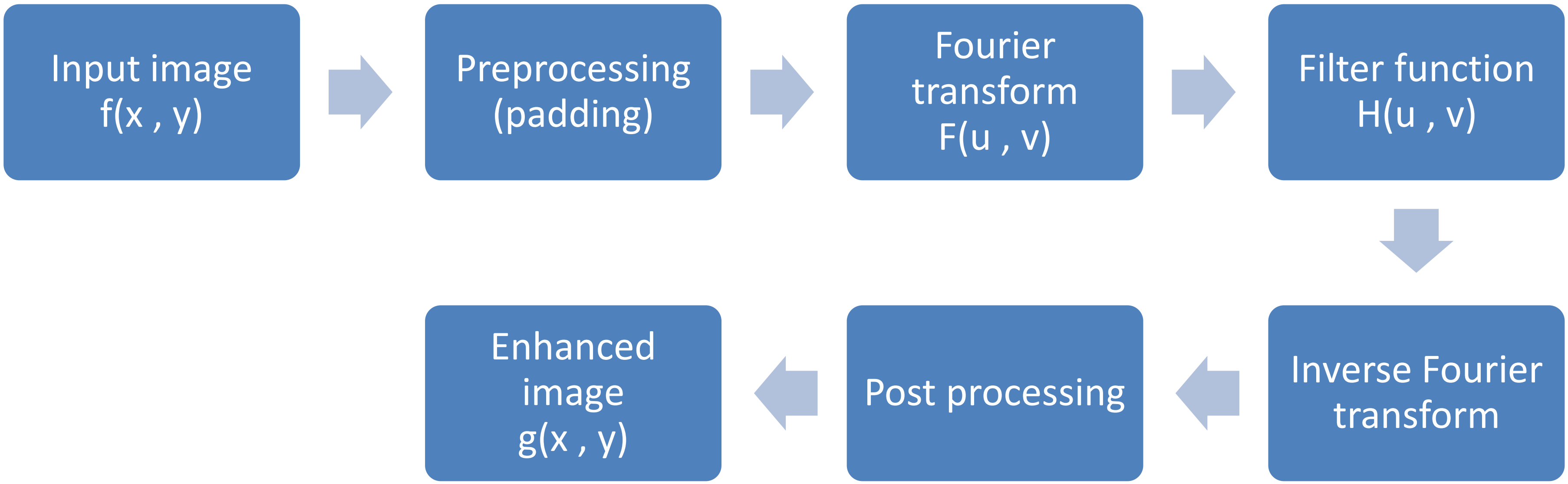
Digital Image Processing

Image Enhancement:
Filtering in the Frequency Domain

Mean of frequency filtering in image processing

The frequency domain is a space which is defined by Fourier transform. Fourier transform has a very wide application in image processing. Frequency domain analysis is used to indicate how signal energy can be distributed in a range of frequency.

Filtering in frequency domain:-



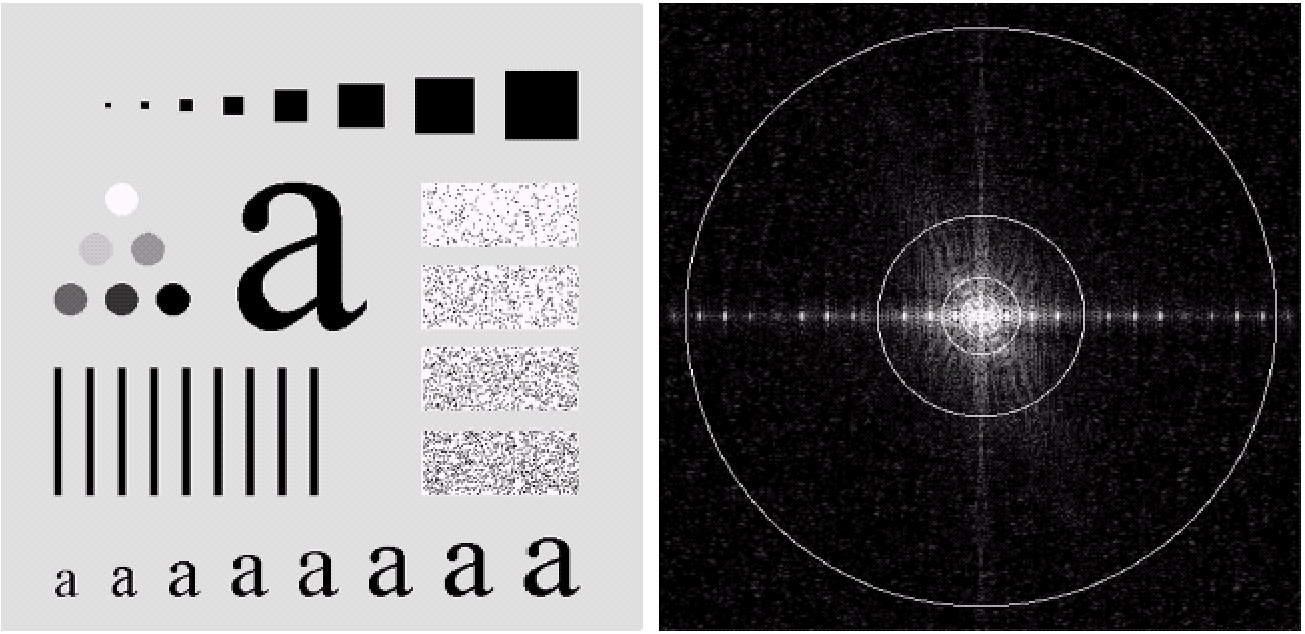
Smoothing frequency filters:-

- One of frequency filters called also LOW PASS FILTER.
- Used for smoothen the image as it allows only low frequency component to pass through.
- They are used to remove noise

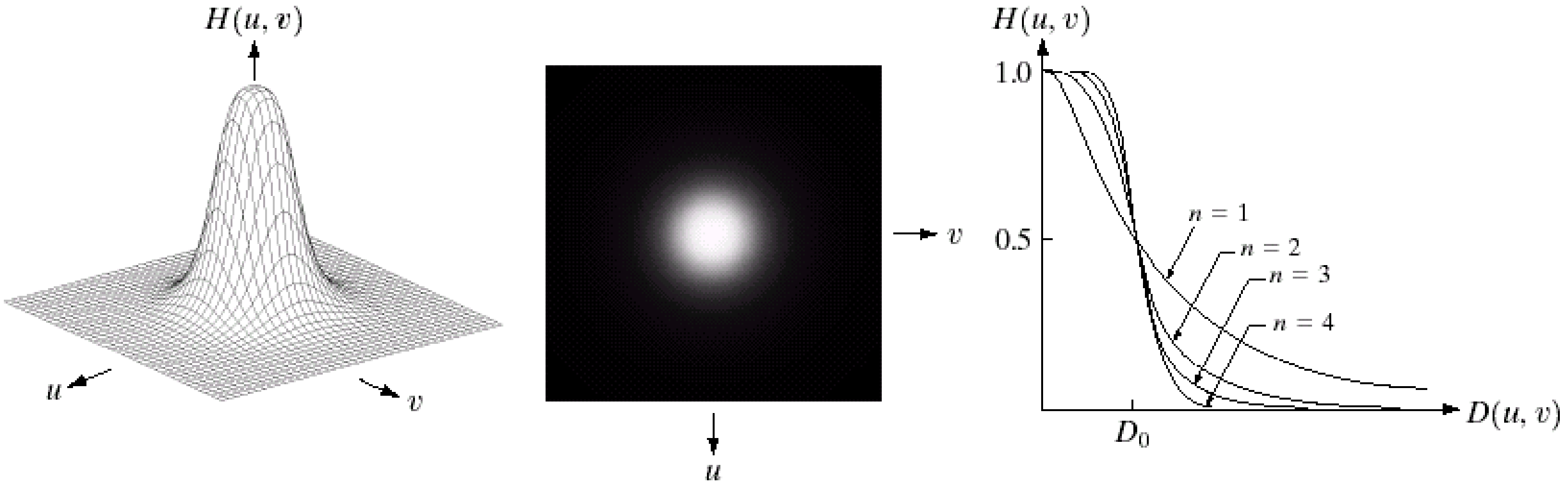
Types of low pass filter:-

- Ideal low pass filter
- Butterworth low pass filter
- Gaussian low pass filter

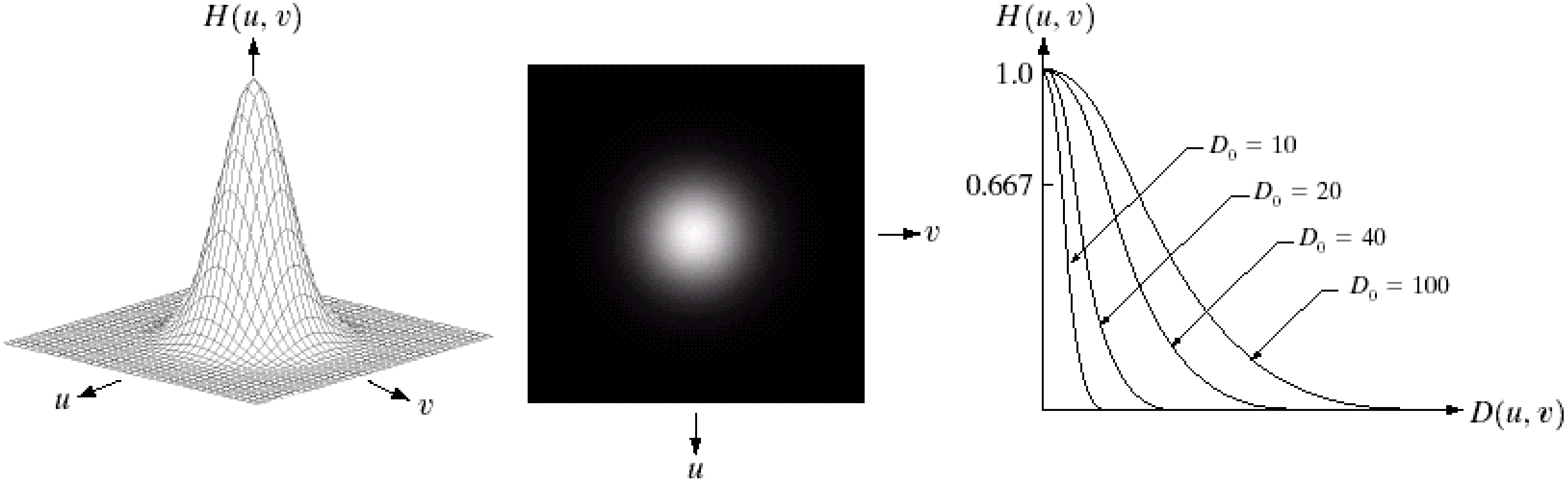
Ideal low pass filter



Butterworth Lowpass Filters



9 Gaussian Lowpass Filters

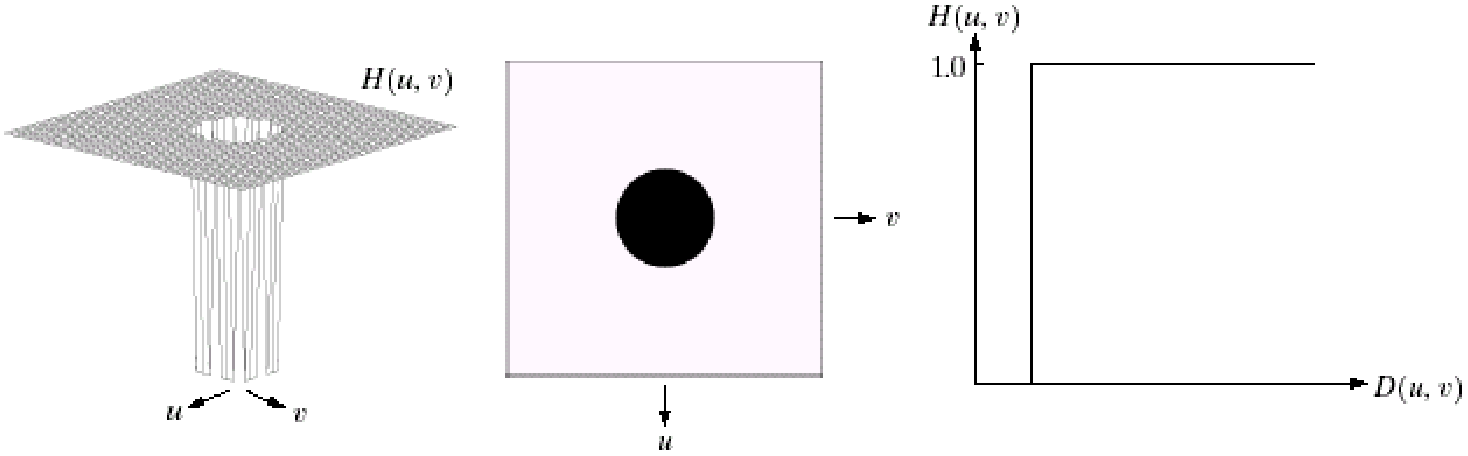


Ideal High Pass Filters

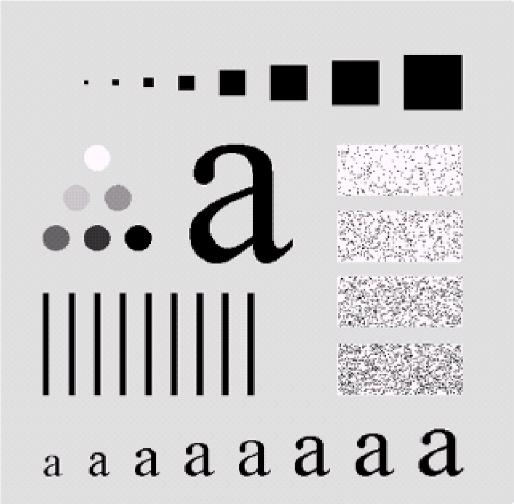
The ideal high pass filter is given as:

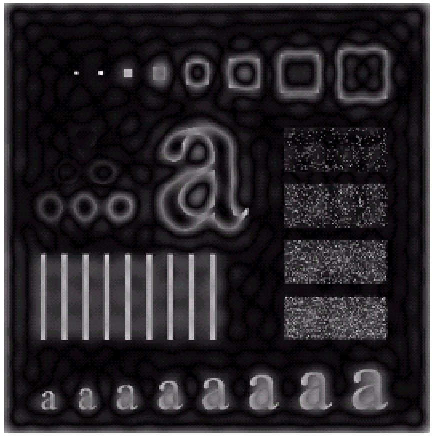
$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

where D_0 is the cut off distance as before

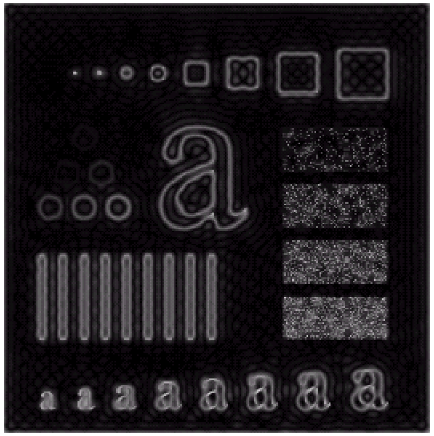


Ideal High Pass Filters (cont…)

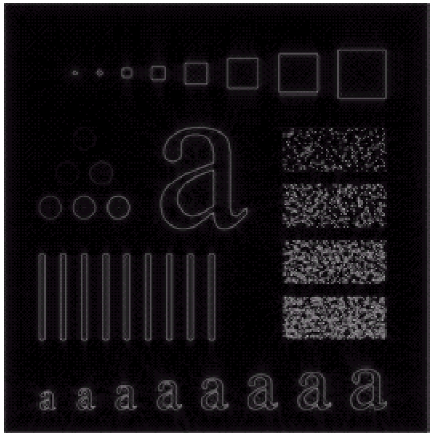




Results of ideal high pass filtering with $D_0 = 15$



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



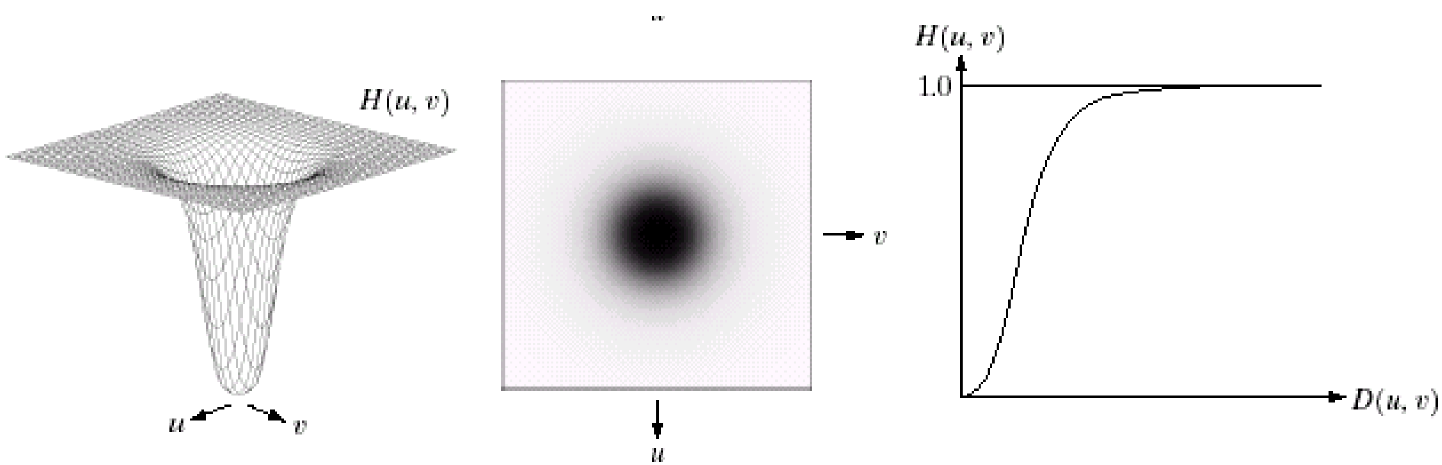
Results of ideal high pass filtering with $D_0 = 80$

Butterworth High Pass Filters

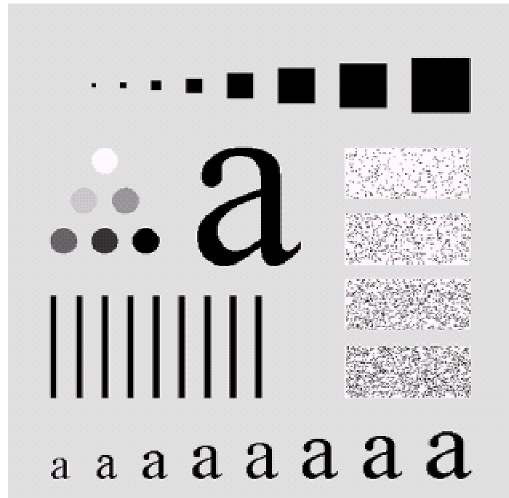
The Butterworth high pass filter is given as:

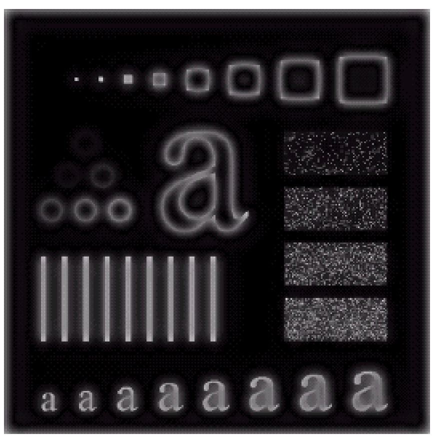
$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

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

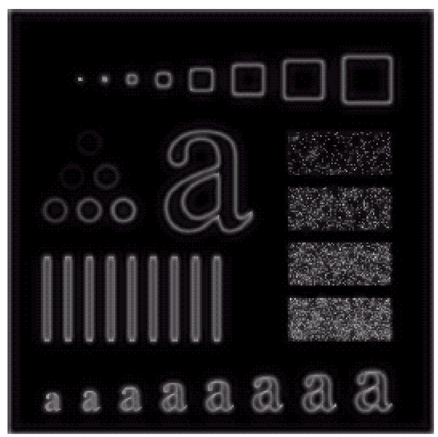


Butterworth High Pass Filters (cont…)

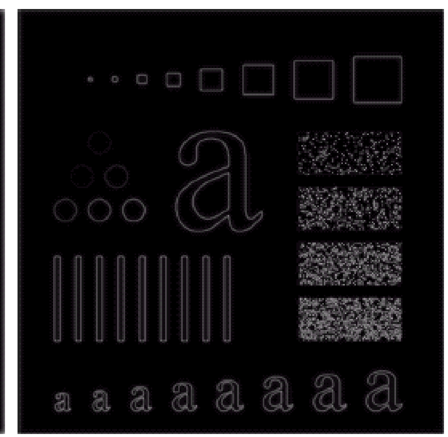




Results of Butterworth high pass filtering of order 2 with $D_0 = 15$



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



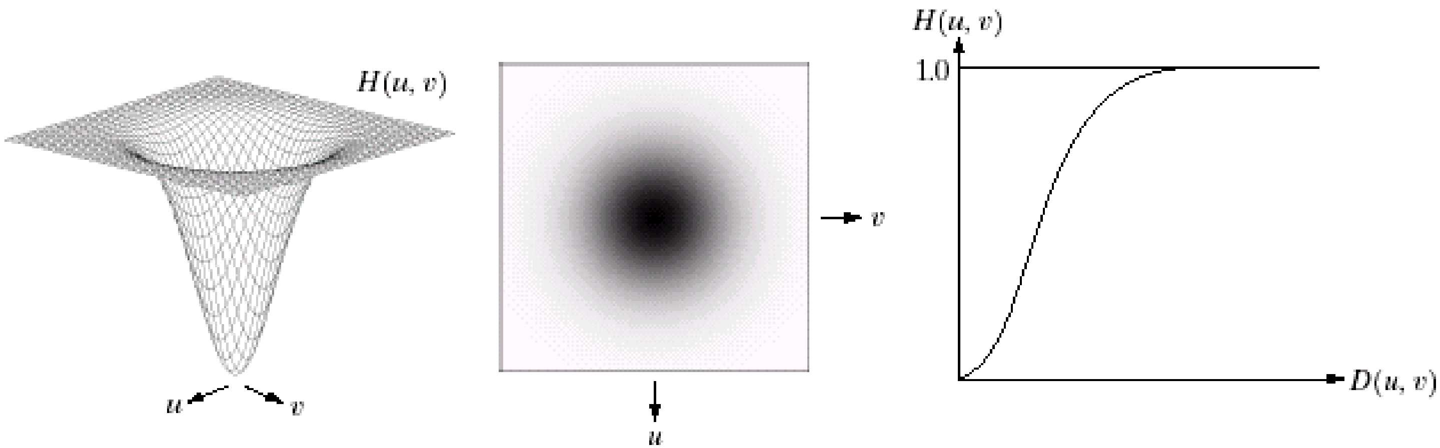
Results of Butterworth high pass filtering of order 2 with $D_0 = 80$

Gaussian High Pass Filters

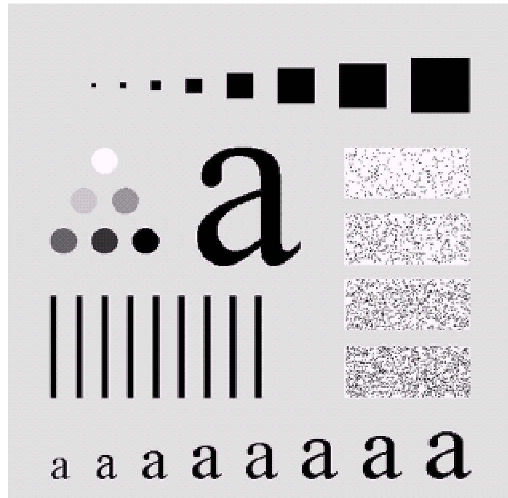
The Gaussian high pass filter is given as:

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

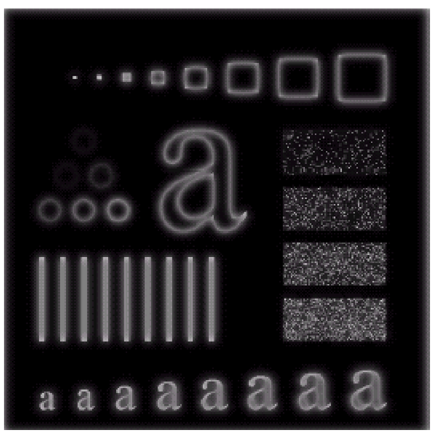
where D_0 is the cut off distance as before

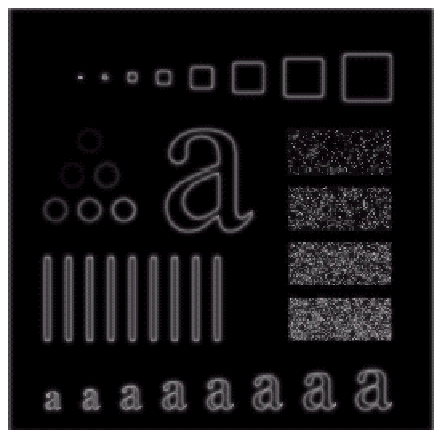


Gaussian High Pass Filters (cont…)



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

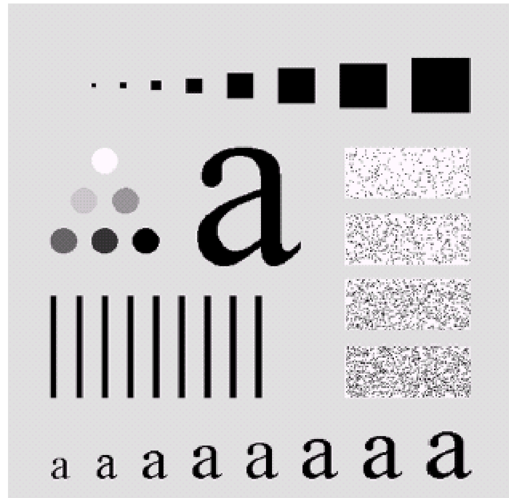


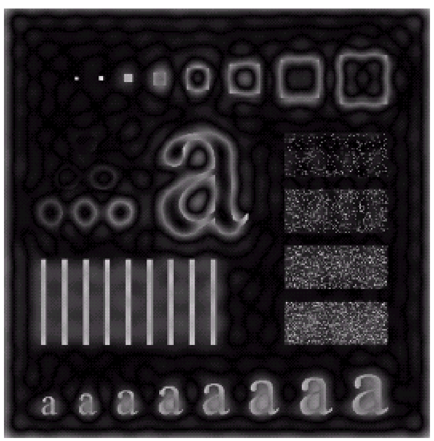


Results of Gaussian high pass filtering with $D_0 = 30$

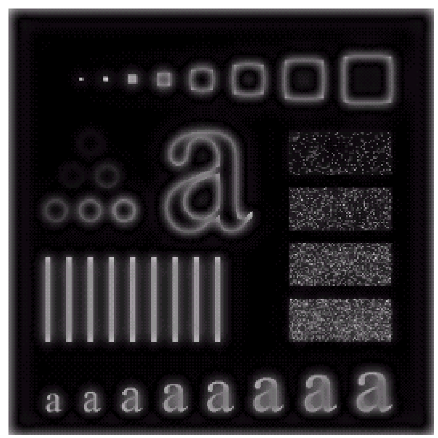
Results of Gaussian high pass filtering with $D_0 = 80$

Highpass Filter Comparison

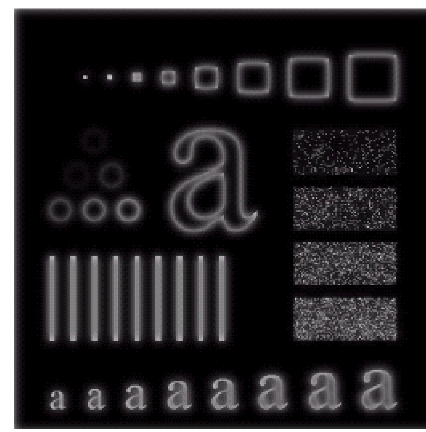




Results of ideal high pass filtering with D_0 = 15



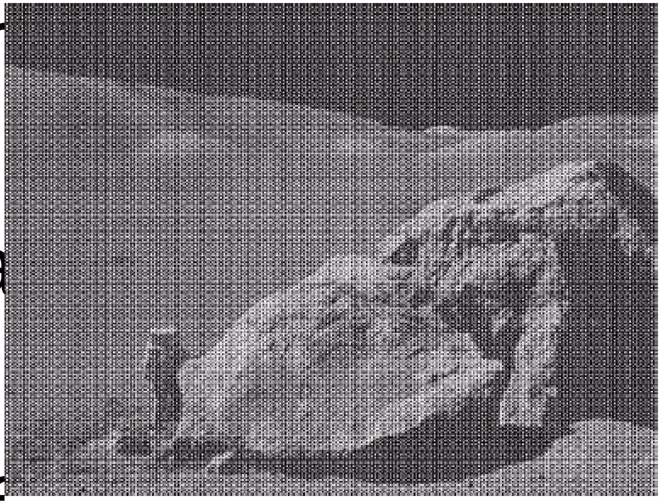
Results of Butterworth high pass filtering of order 2 with D_0 = 15

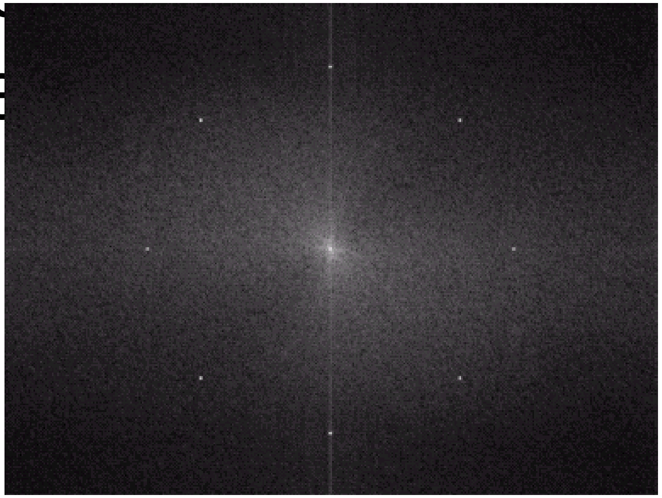


Results of Gaussian high pass filtering with D_0 = 15

Periodic Noise

- Typically arises due to electr
electromagnetic interference
- Gives rise to regular noise pa
image
- Frequency domain technique
Fourier domain are most effe
removing periodic noise





Band Reject Filters

Removing periodic noise form an image involves removing a particular range of frequencies from that image

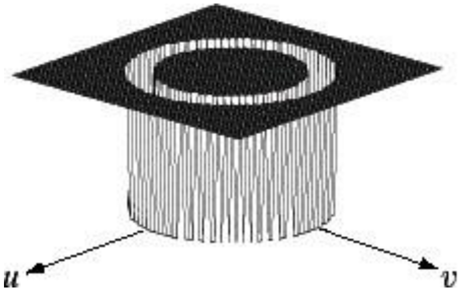
Band reject filters can be used for this purpose

An ideal band reject filter is given as follows:

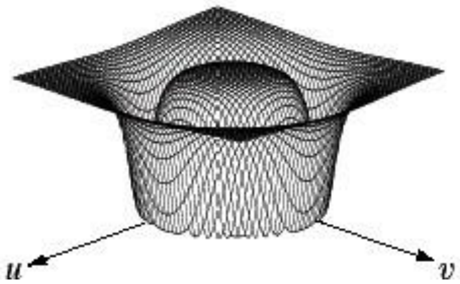
$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) < D_0 - \frac{W}{2} \\ 0 & \text{if } D_0 - \frac{W}{2} \le D(u,v) \le D_0 + \frac{W}{2} \\ 1 & \text{if } D(u,v) > D_0 + \frac{W}{2} \end{cases}$$

Band Reject Filters (cont…)

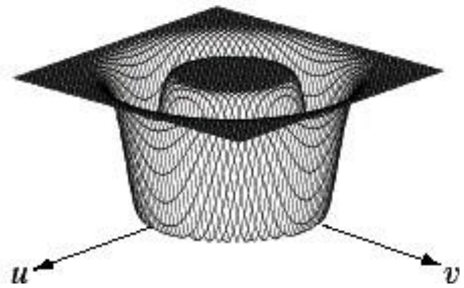
The ideal band reject filter is shown below, along with Butterworth and Gaussian versions of the filter



Ideal Band Reject Filter



Butterworth Band Reject Filter (of order 1)

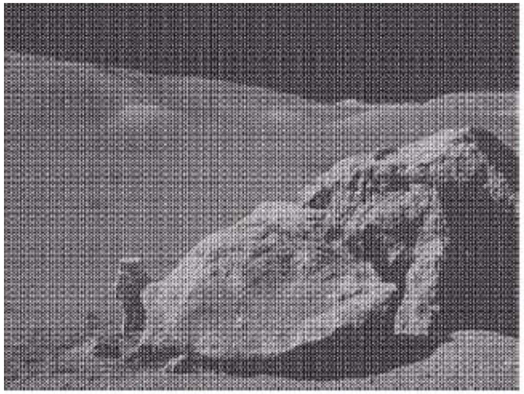


Gaussian Band Reject Filter

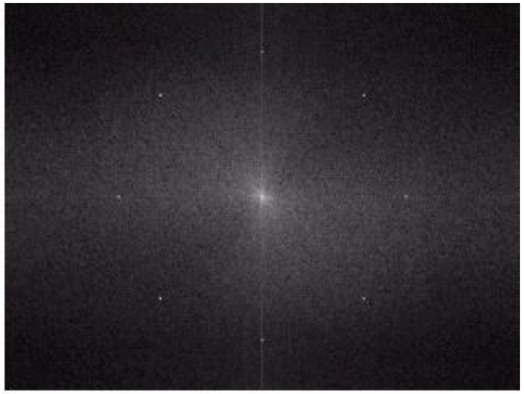
Band Reject Filter Example

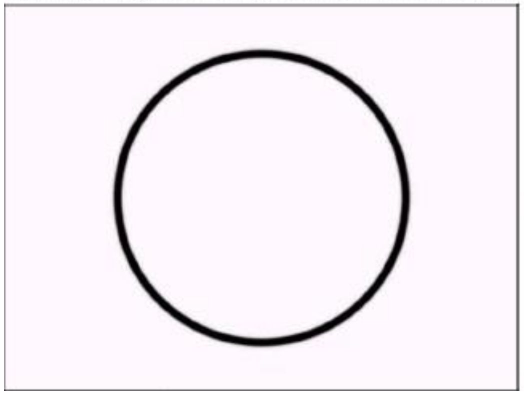
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image corrupted by sinusoidal noise



Fourier spectrum of corrupted image





Butterworth band reject filter

Filtered image

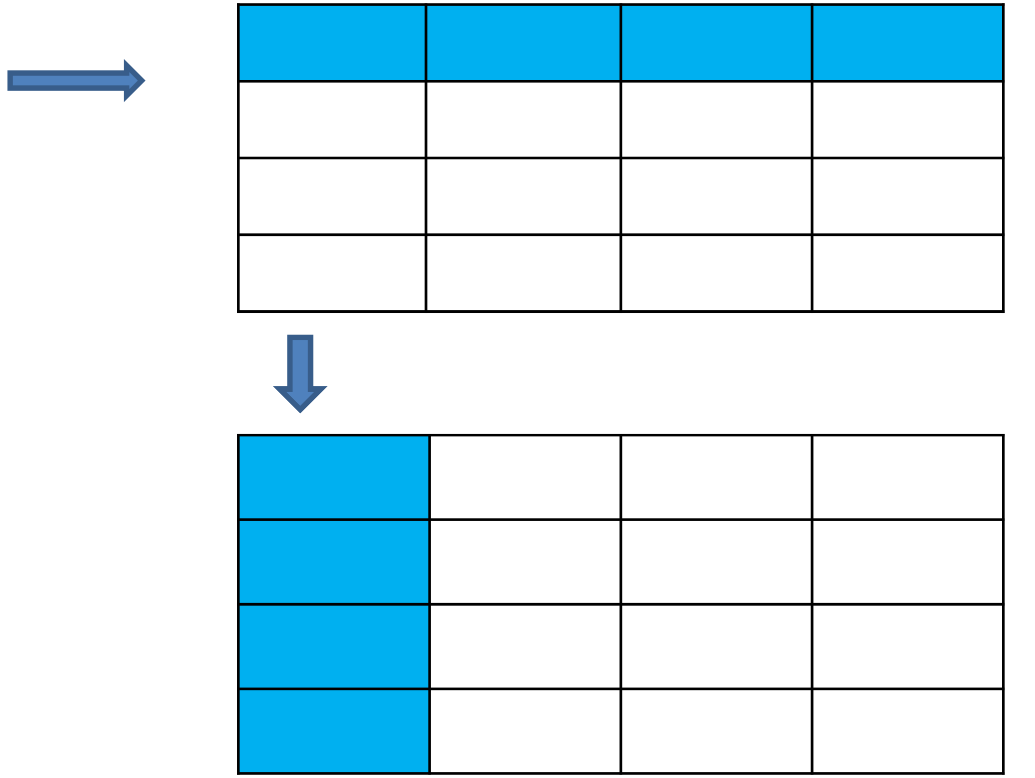
2D Discrete Fourier Transform

A [M,N] point DFT is periodic with period [M,N]

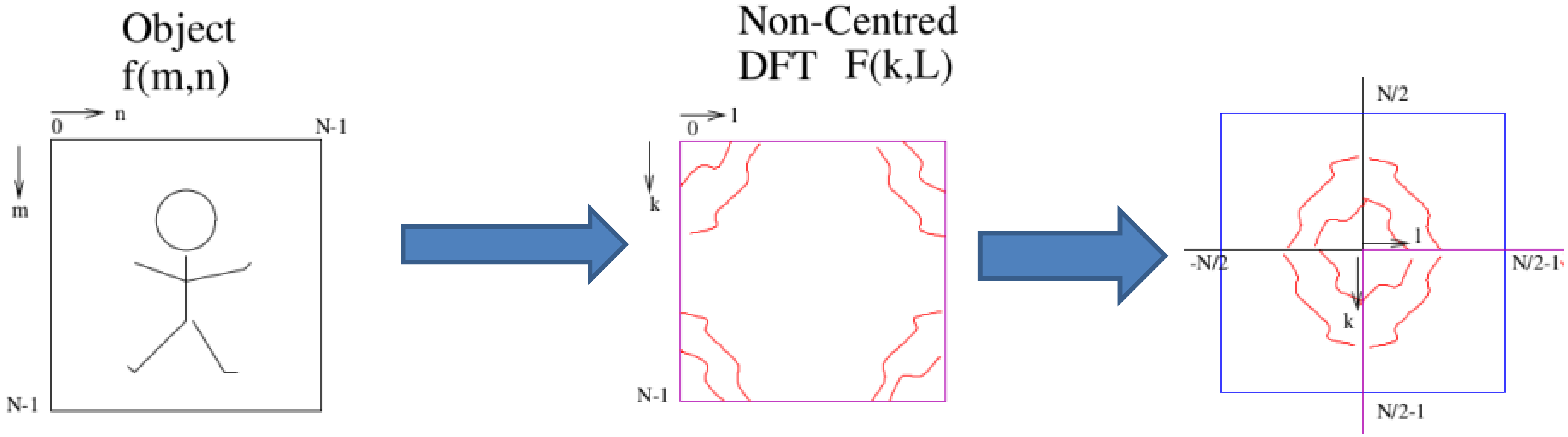
$$F[k,l] = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]e^{-j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)}$$

2D DFT complexity

A DFT performe as $O((N*M)^2)$ in time complexity



2D DFT complexity



0

Fast Fourier transform

Fast Fourier Transform, or FFT, is a computational algorithm that reduces the computing time and complexity of large transforms. FFT is just an algorithm used for fast computation of the DFT.

FFT complexity

- FFT breaks the DFT into smaller DFTs.
- FFT reduces the time complexity in the order of O (NlogN).