Digital Image Processing (CSE/ECE 478)

Lecture # 09: Filtering in Fourier Domain

Avinash Sharma

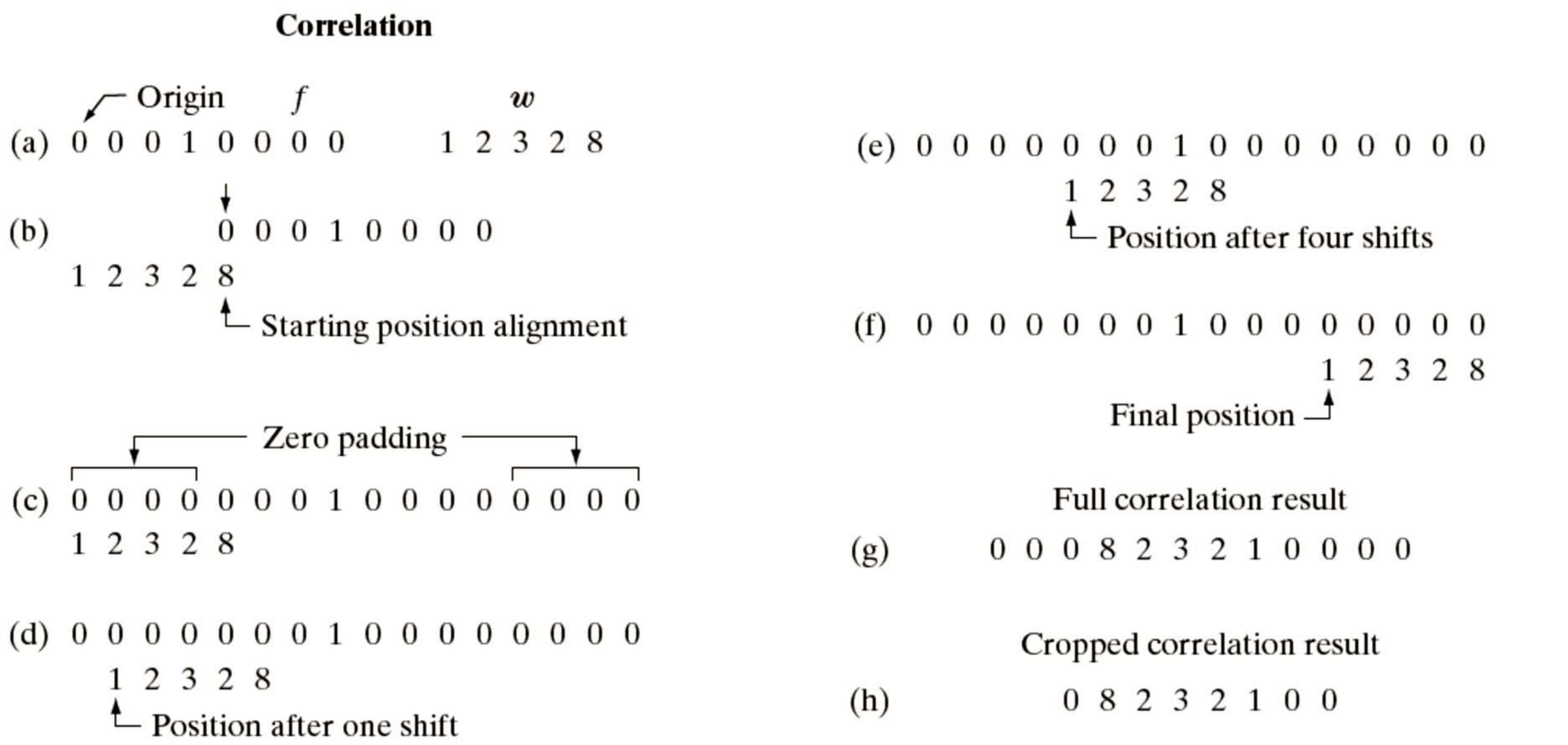
Center for Visual Information Technology (CVIT),

IIIT Hyderabad

Today's class

- Convolution Theorem
- Frequency domain filtering
  - Low pass
  - High Pass
  - Laplacian
- Next Class: FFT

Correlation



Convolution

Convolution

Origin $\quad f \quad$ *w* rotated 180°

0 0 0 1 0 0 0 0 $\quad$ 8 2 3 2 1 $\quad$ (i)

$\quad$ 0 0 0 1 0 0 0 0 $\quad$ (j)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (k)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (l)
$\quad$ 8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (m)
$\quad$ 8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (n)
$\quad$ 8 2 3 2 1

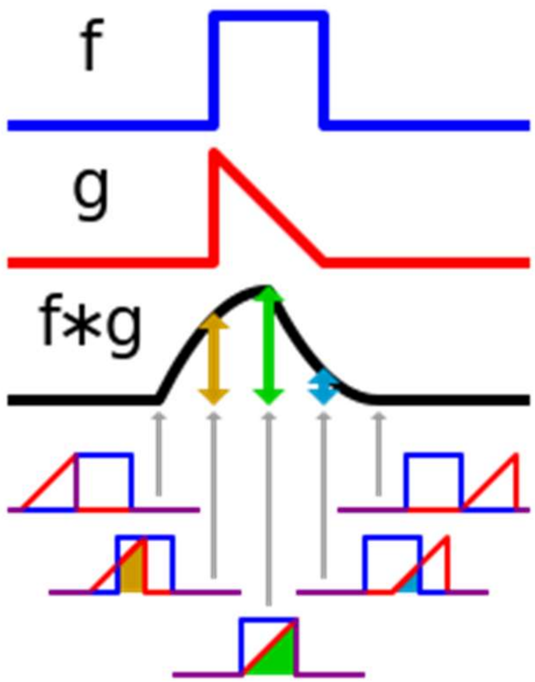
Full convolution result

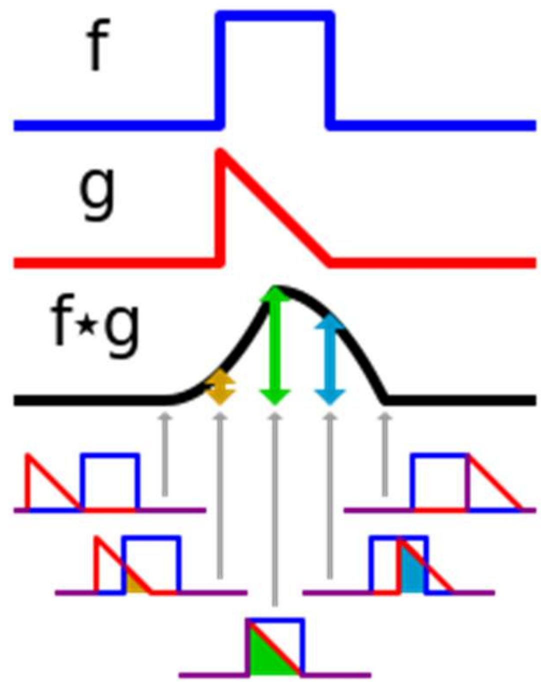
0 0 0 1 2 3 2 8 0 0 0 0 $\quad$ (o)

Cropped convolution result

0 1 2 3 2 8 0 0 $\quad$ (p)

Convolution vs Correlation

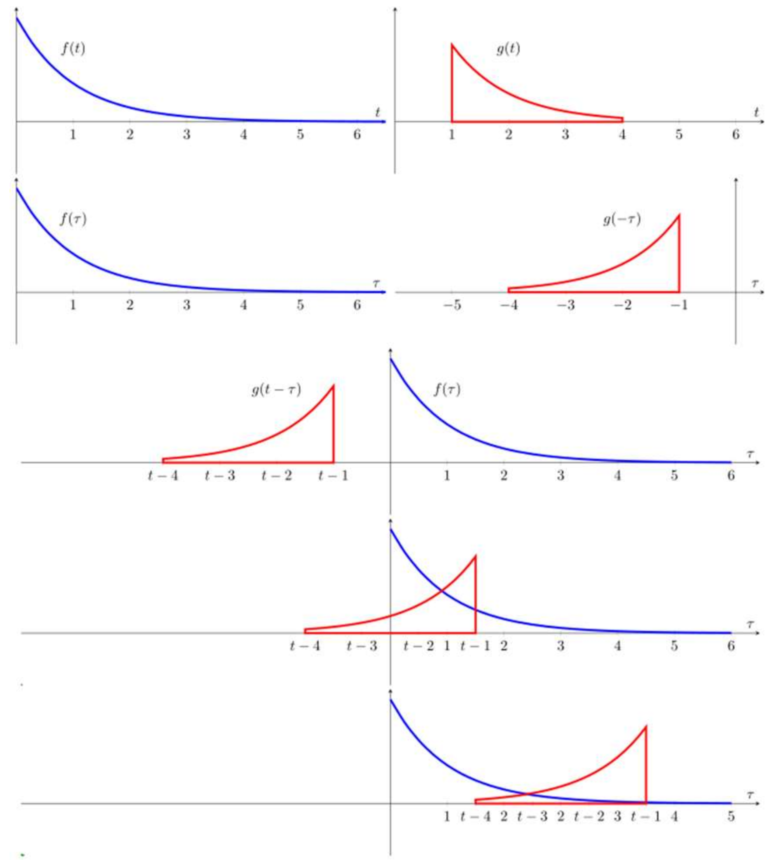




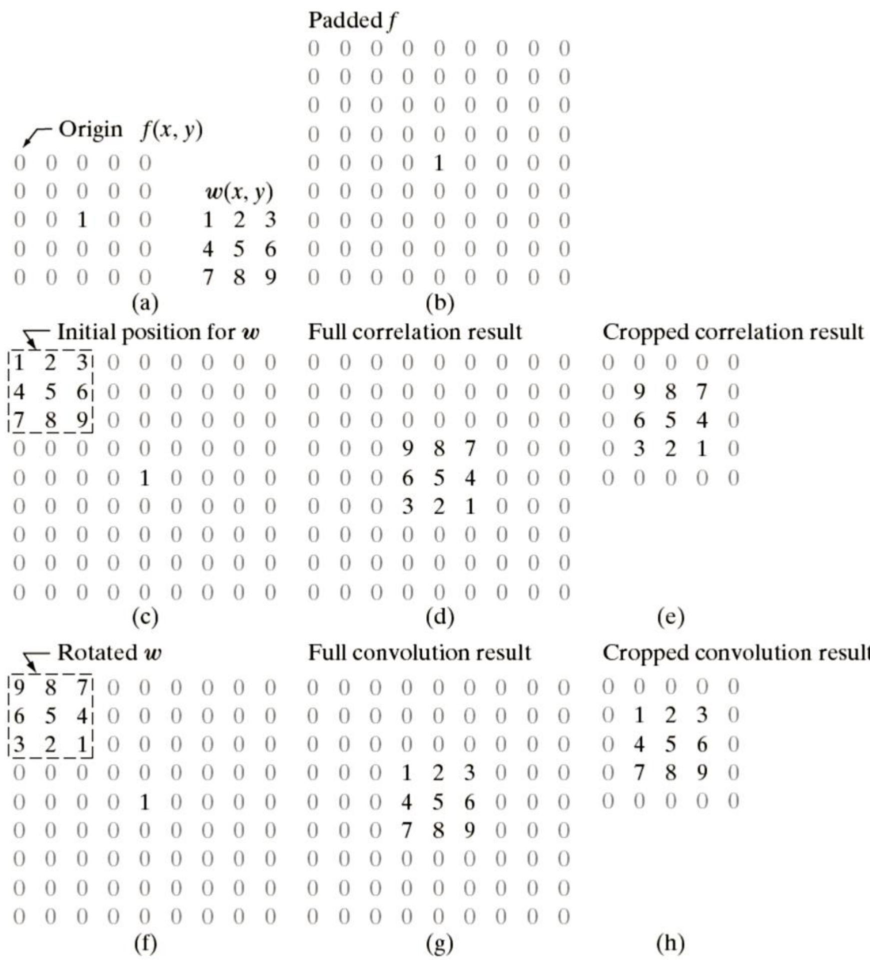
Both converge if signals involved are symmetric !

Convolution Operator

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$



Convolution vs Correlation (2D)



Convolution (2D)

$$w(x,y) \star f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)$$

- Evaluated for all values of displacement variables x and y
- Filter size $m \times n$ (notational convenience → m, n are assumed odd)
- $a = (m-1)/2$ and $b = (n-1)/2$

Convolution Theorem

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

In other words:

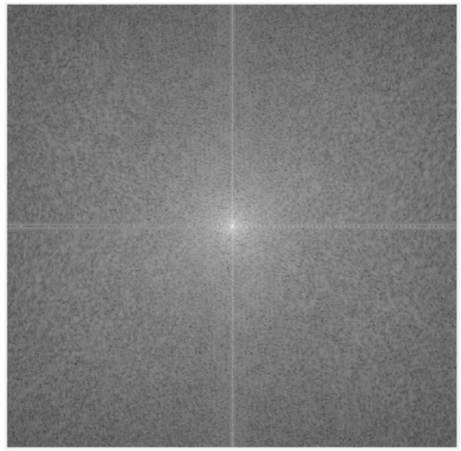
$$\Im(f(x,y) \bigstar h(x,y)) = F(u,v)H(u,v)$$

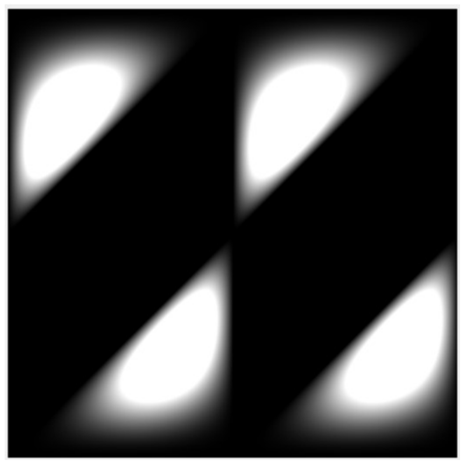
$$f(x,y) \bigstar h(x,y) = \Im^{-1}(F(u,v)H(u,v))$$

Correspondence to spatial filtering

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

Correspondence to spatial filtering





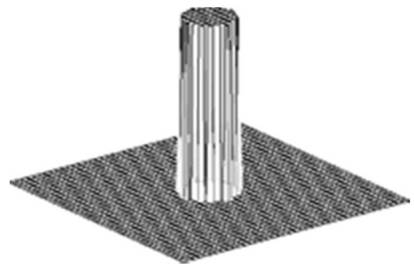
-1	0	1
-2	0	2
-1	0	1

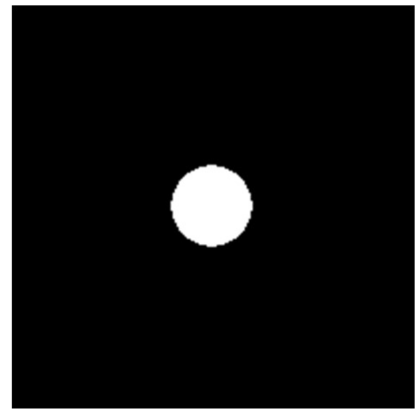
Correspondence to spatial filtering

```
%Sobel filter in frequency domain
f = rgb2gray(imread('boy.jpg'));
h = [-1 0 1; -2 0 2; -1 0 1];
F = fft2(double(f), 402, 402);
H = fft2(double(h), 402, 402);
F_fH = fftshift(H).*fftshift(F);
ffi = ifft2(ifftshift(F_fH));
```

Ideal Low Pass Filters

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





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

$D_0 \rightarrow$ cut off frequency

Image courtesy: Gonzalez and Woods

Ideal Low Pass Filters

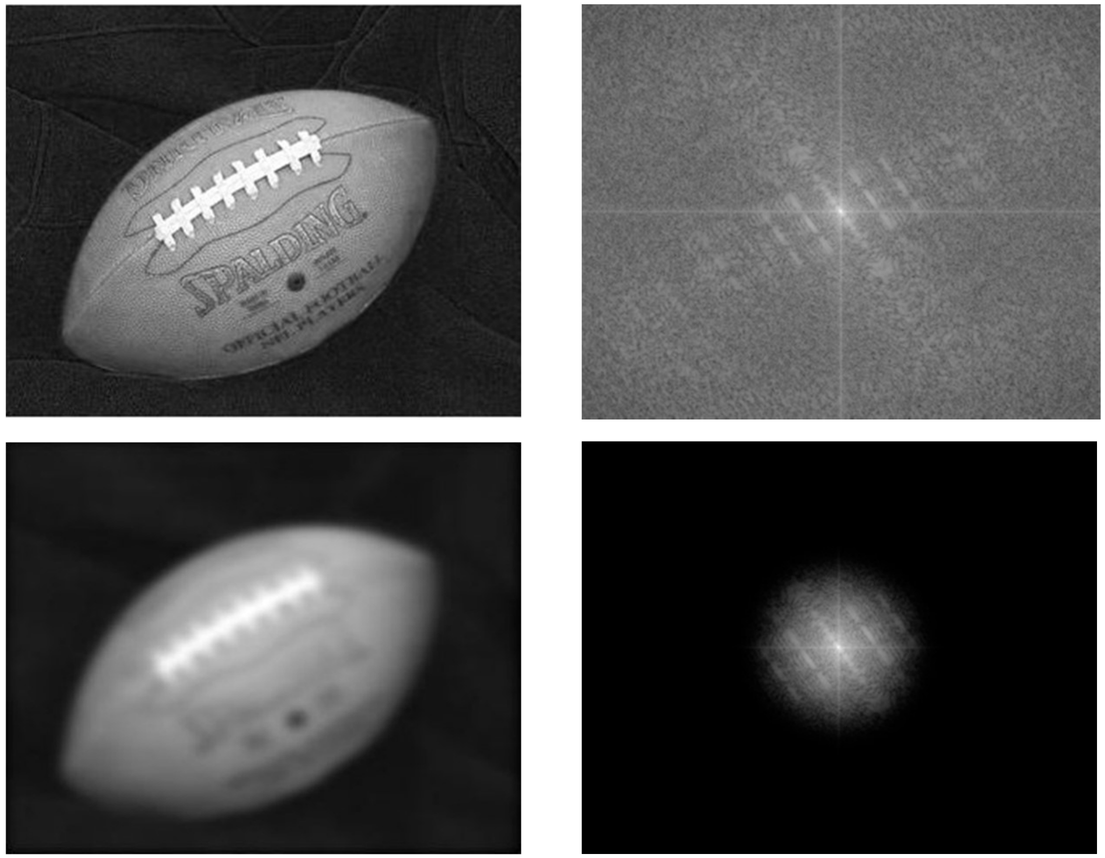
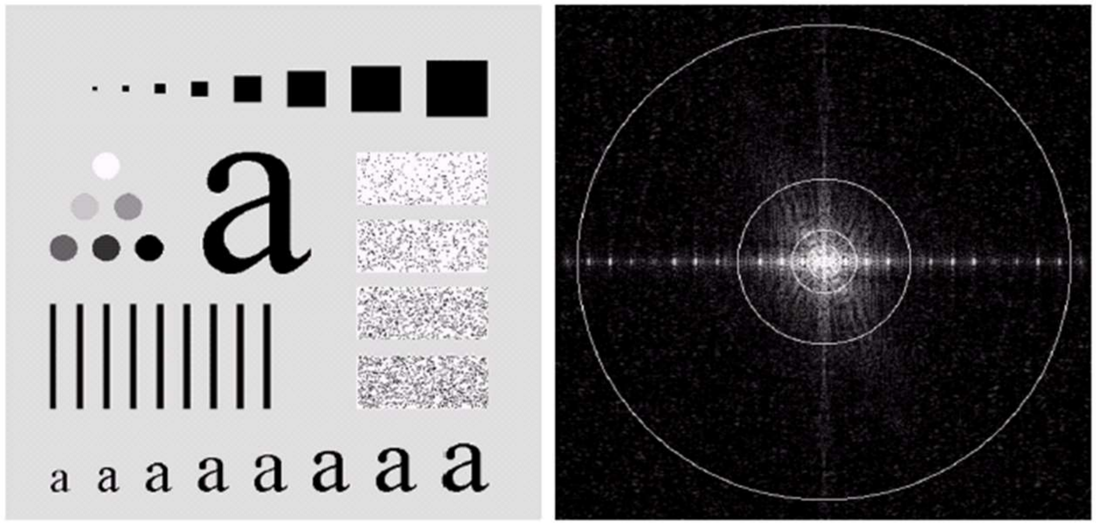


Image courtesy: cs.uregina.ca

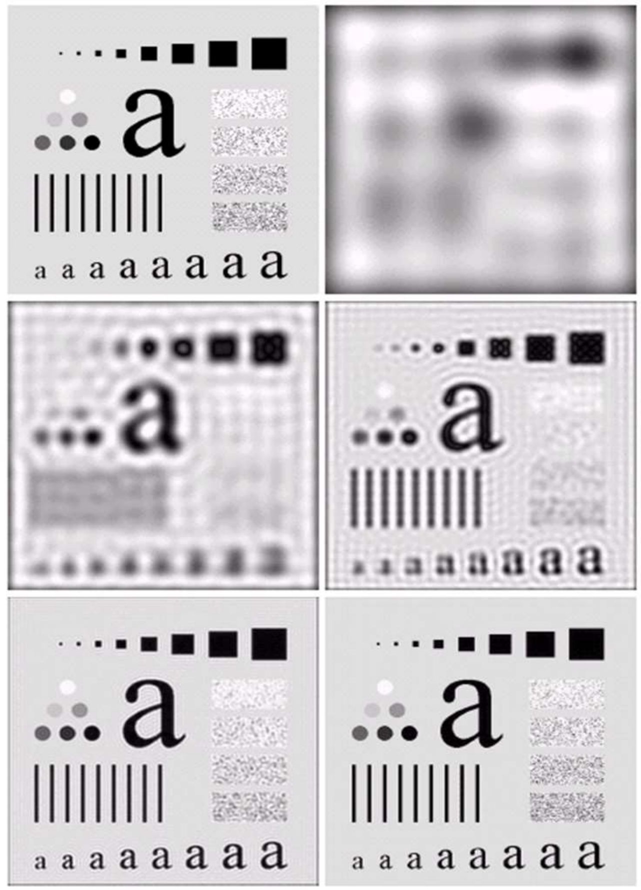
Ideal Low Pass Filters



Radii 10,30,60,160 and 460 → power 87, 93.1, 95.7, 97.8 and 99.2

Image courtesy: Gonzalez and Woods

Ideal Low Pass Filters



ILPF radius 10

ILPF radius 30

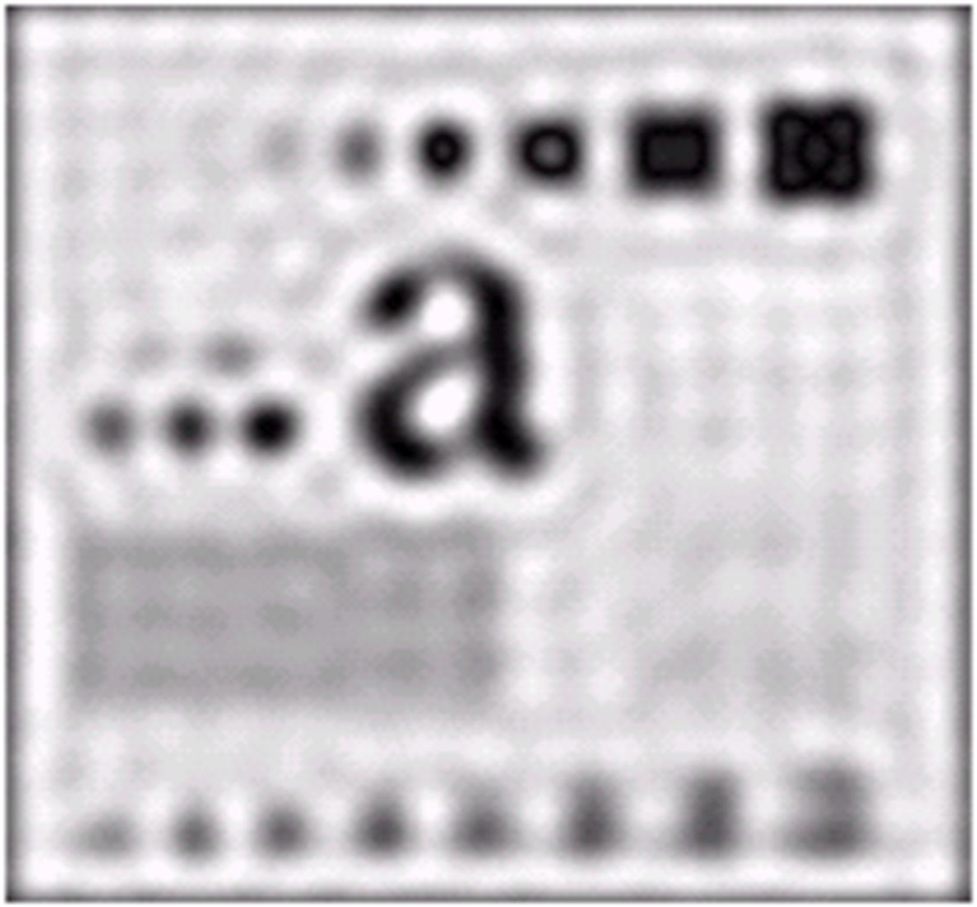
ILPF radius 60

ILPF radius 160

ILPF radius 460

Image courtesy: Gonzalez and Woods

Ideal Low Pass Filters



ILPF radius 30

Image courtesy: Gonzalez and Woods

Ideal Low Pass Filters

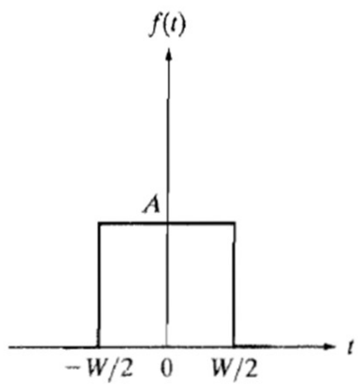


Image courtesy: Gonzalez and Woods

Relationship between u and x

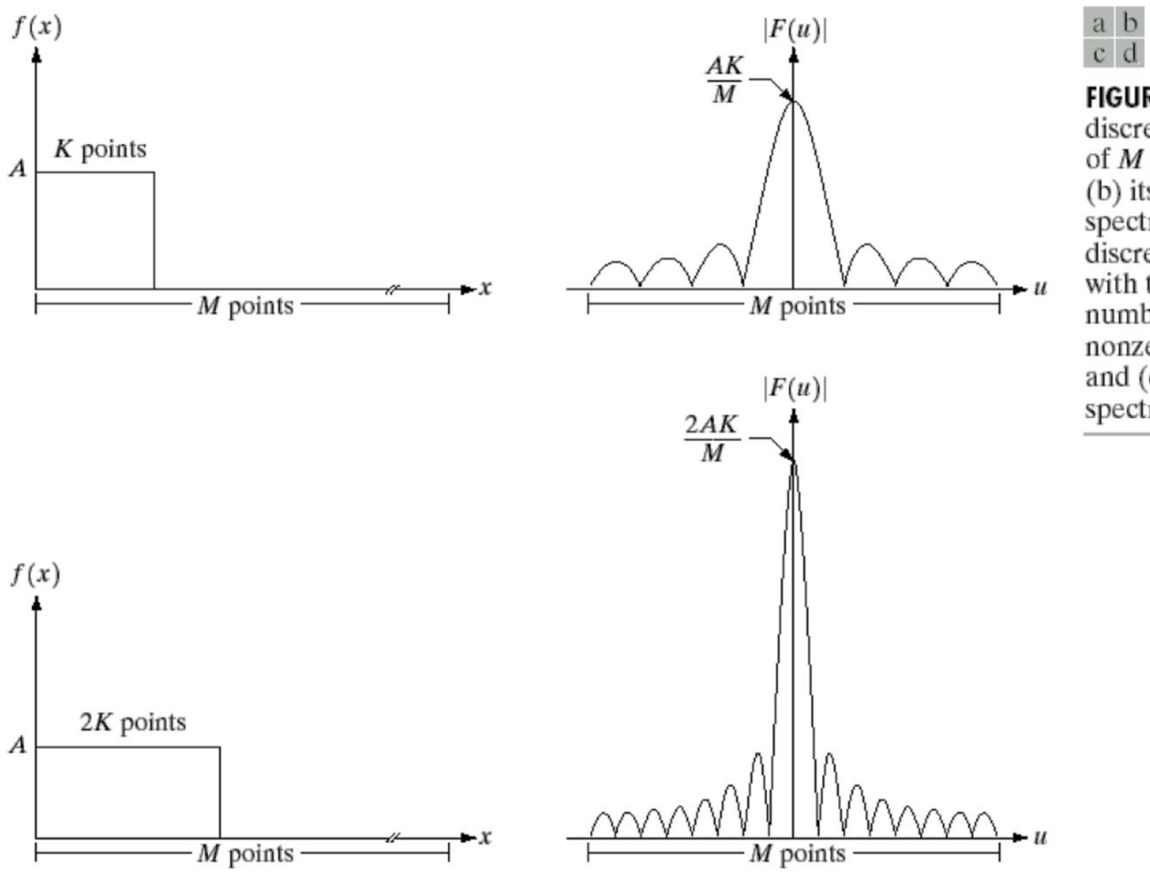
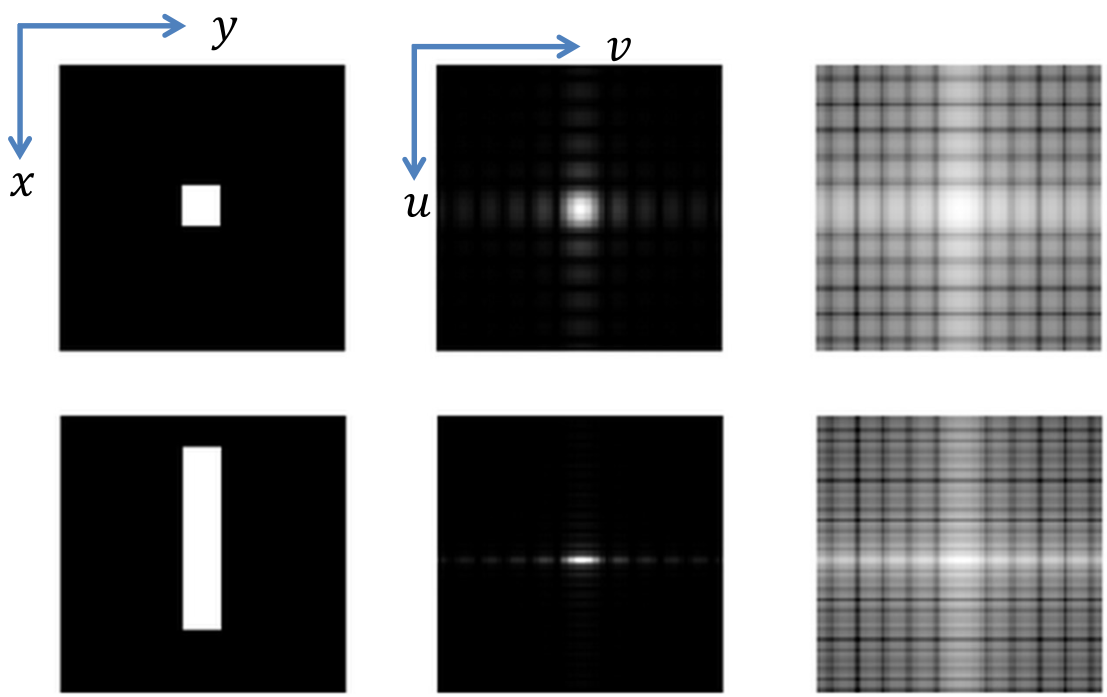
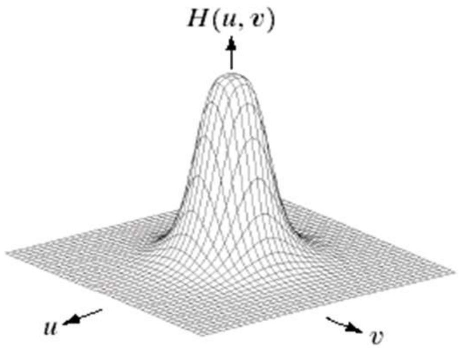


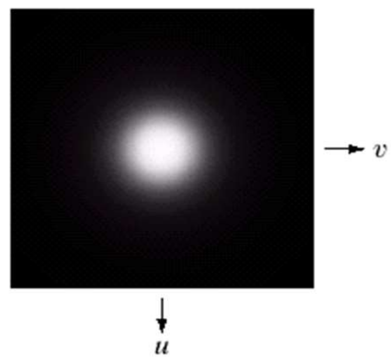
FIGURE 4.2 (a) A discrete function of M points, and (b) its Fourier spectrum. (c) A discrete function with twice the number of nonzero points, and (d) its Fourier spectrum.

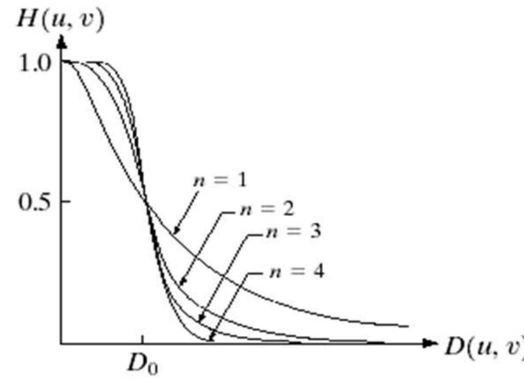
Relationship between u and x (or v and y)



Butterworth Low Pass Filters





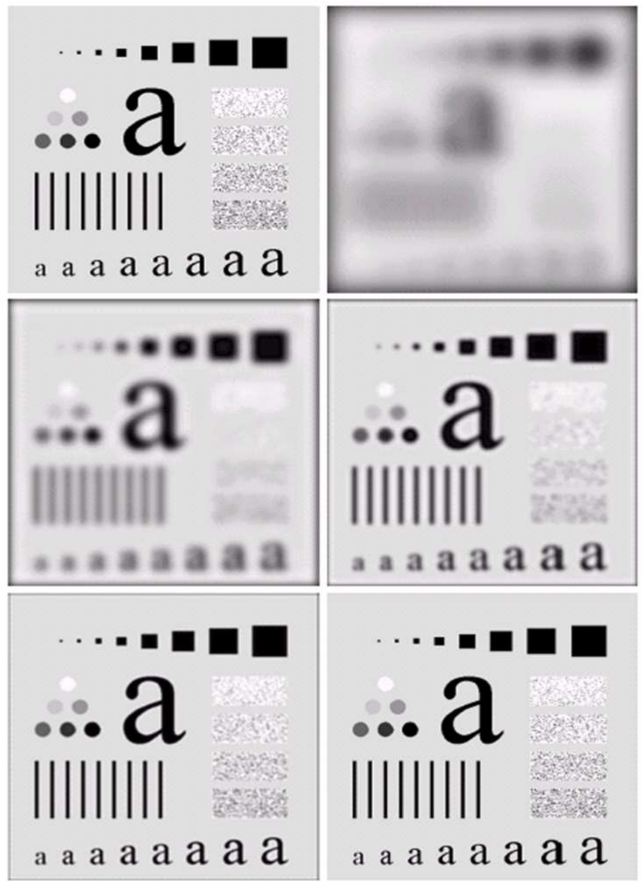


$$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}}$$ where $$D(u,v) = [(u-M/2)^2 + (v-N/2)^2]^{1/2}$$

Image courtesy: Gonzalez and Woods

Butterworth Low Pass Filters (BLPF)

Order two, i.e.
n=2



BLPF cut off
frequency 10

BLPF cut off
frequency 30

BLPF cut off
frequency 60

BLPF cut off
frequency 160

BLPF cut off
frequency 460

Image courtesy: Gonzalez and Woods

Butterworth Low Pass Filters (BLPF)

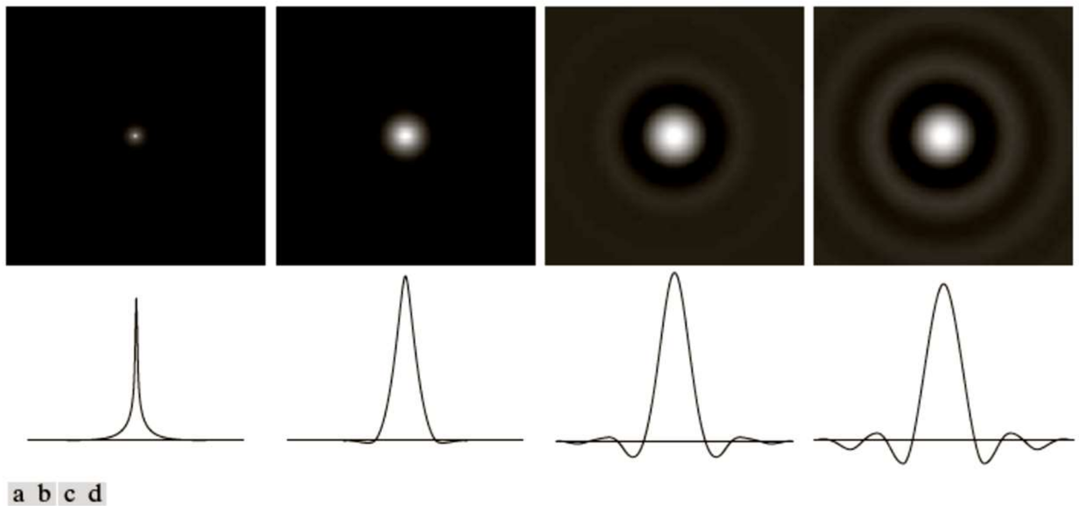
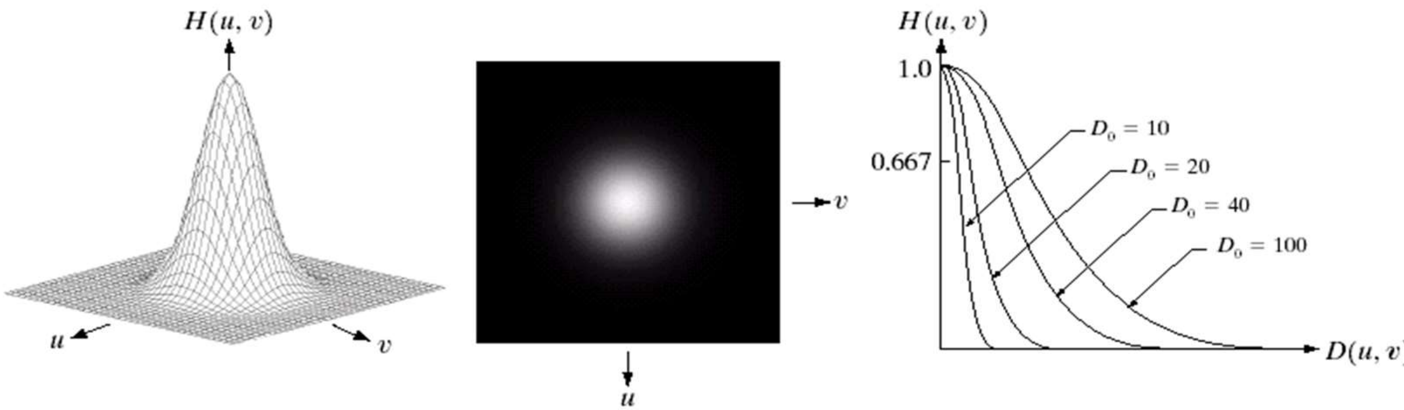


FIGURE 4.46 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is 1000×1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

Gaussian Low Pass Filters

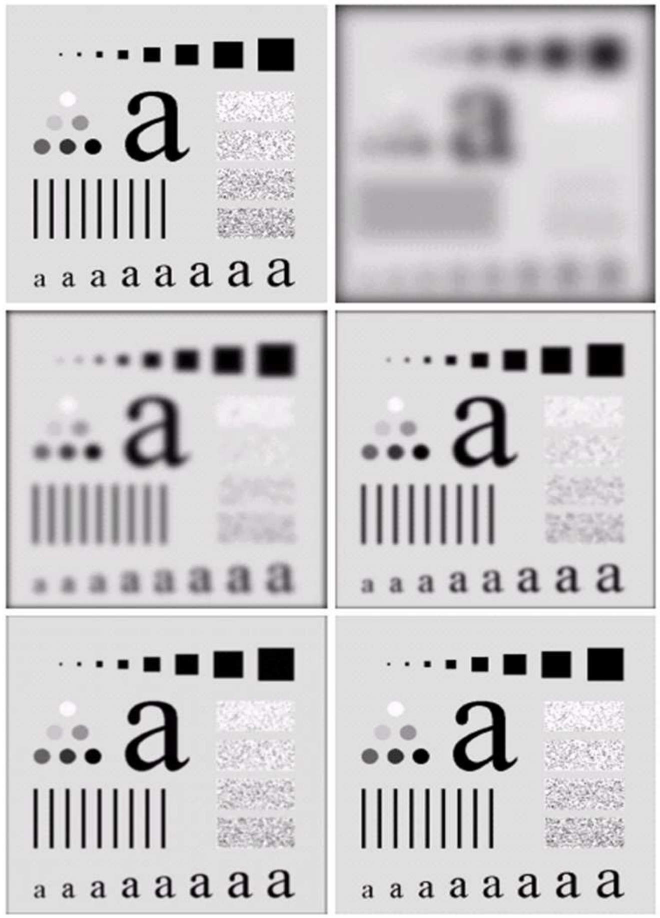


$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

Image courtesy: Gonzalez and Woods

Gaussian Low Pass Filters (GLPF)

No Ringing Phenomenon in Gaussian LPF



GLPF cut off frequency 10

GLPF cut off frequency 30

GLPF cut off frequency 60

GLPF cut off frequency 160

GLPF cut off frequency 460

Image courtesy: Gonzalez and Woods

Comparison (ILPF, BLPF, GLPF)

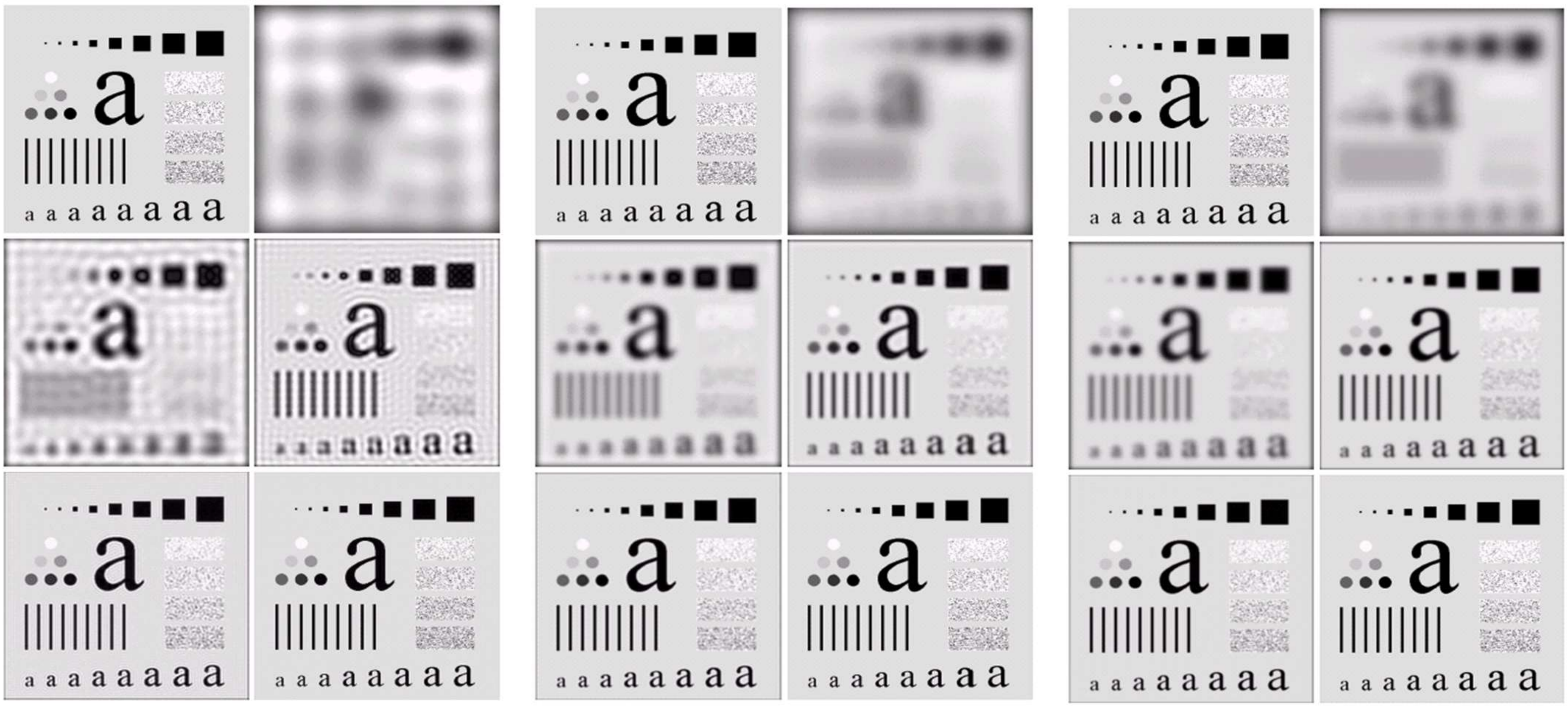


Image courtesy: Gonzalez and Woods

Low pass filtering application

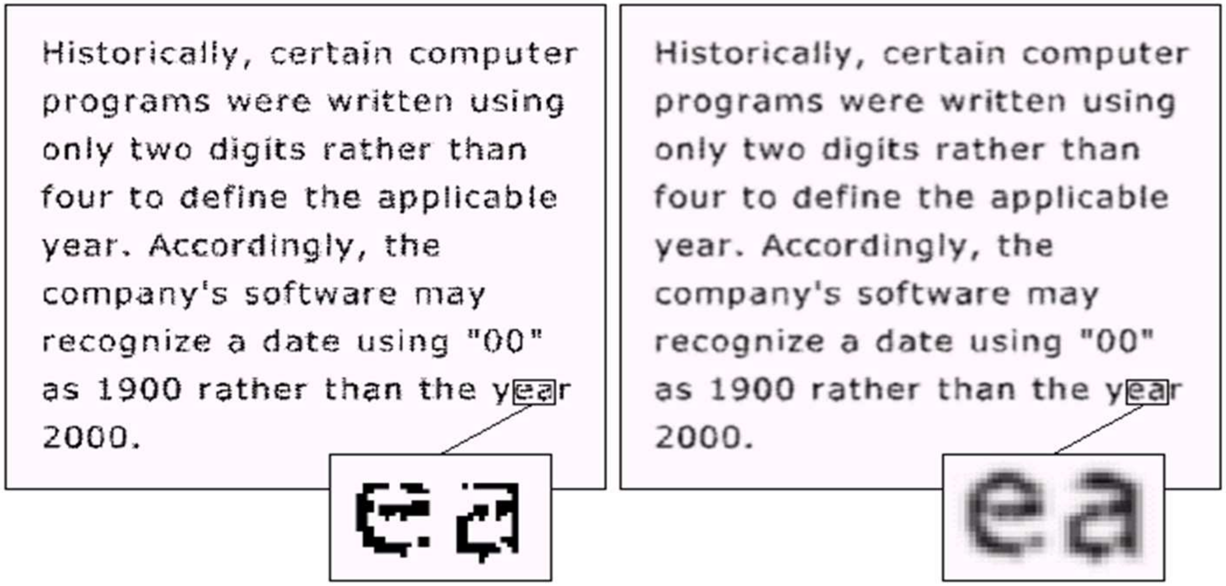


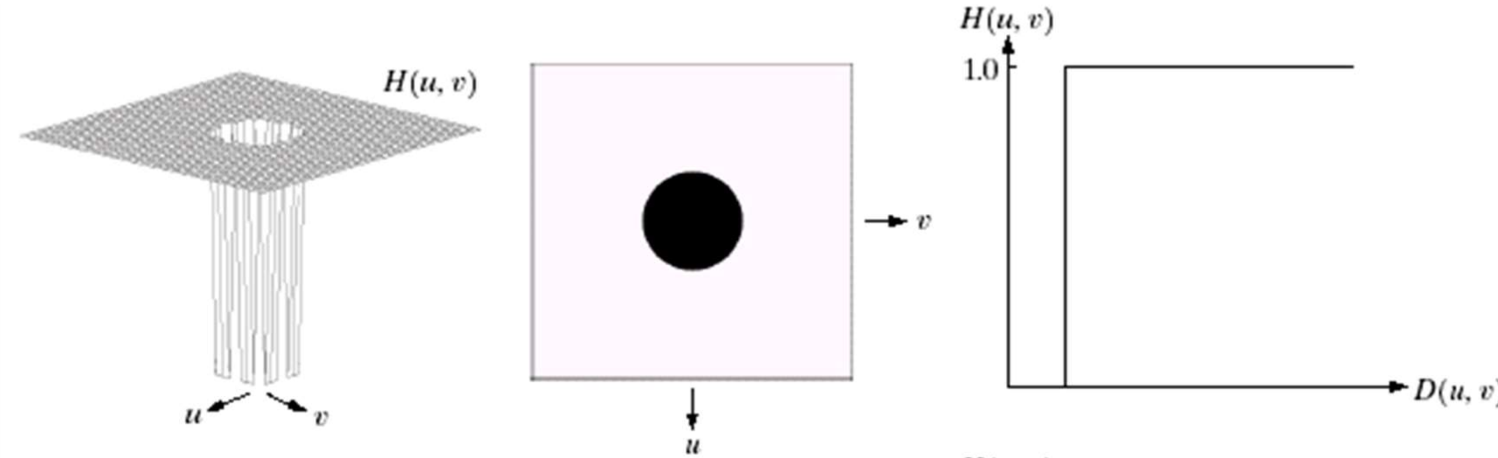
Image courtesy: Gonzalez and Woods

Image Sharpening in Frequency Domain

High Pass filter can be obtained from a given low pass filter:

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

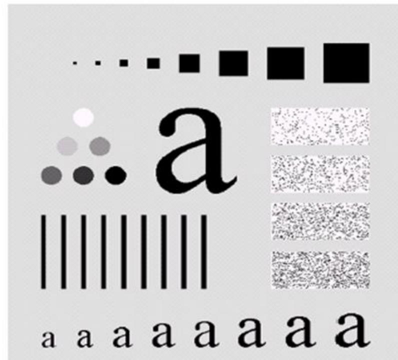
Ideal High Pass Filters

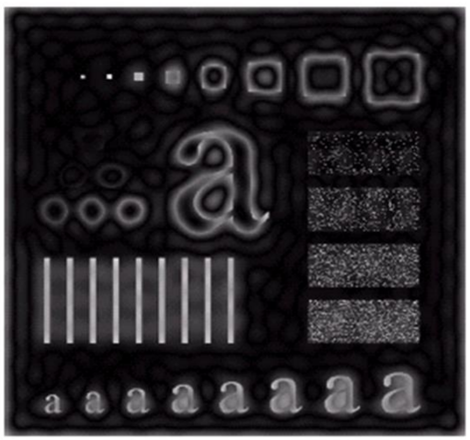


$$H(u,v)=\begin{cases}0 & \text{if } D(u,v)\le D_0\\ 1 & \text{if } D(u,v)>D_0\end{cases}$$

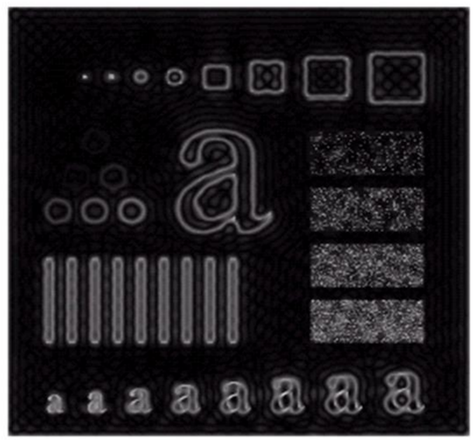
Image courtesy: Gonzalez and Woods

Ideal High Pass Filters

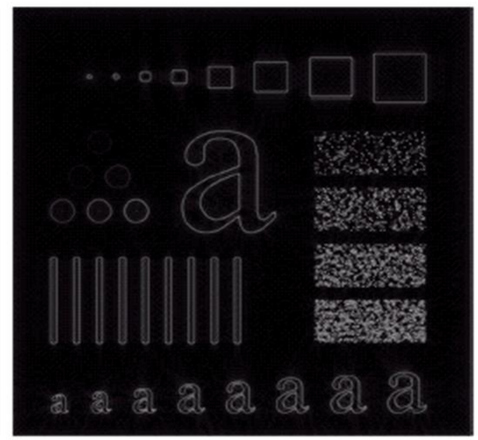




IHPL with D_0 = 30



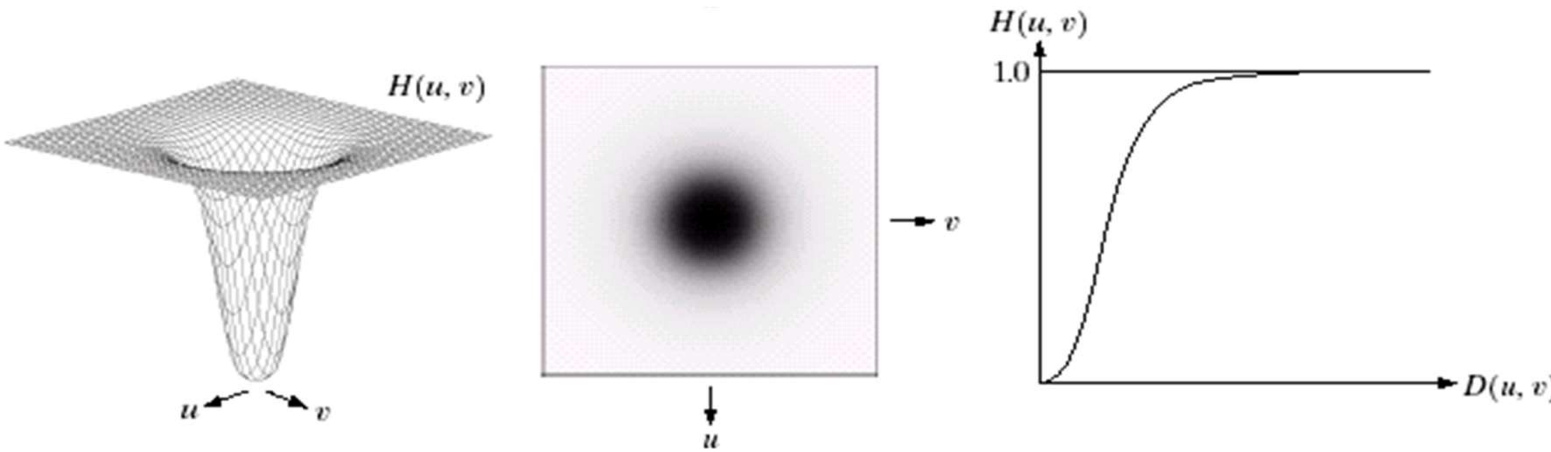
IHPF with D_0 = 60



IHPF with D_0 = 160

Image courtesy: Gonzalez and Woods

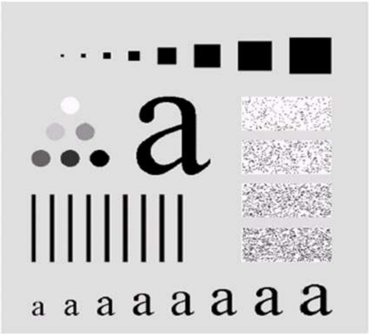
Butterworth High Pass Filters

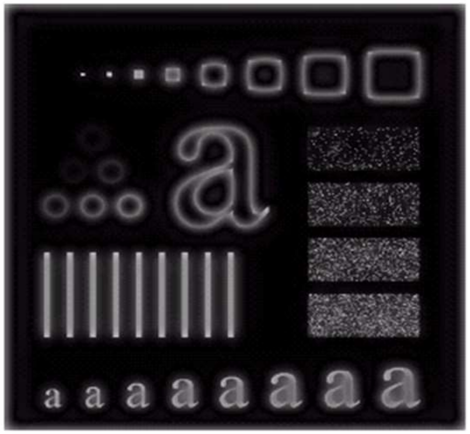


$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

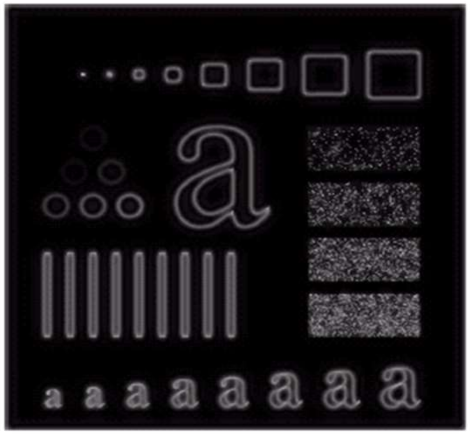
Image courtesy: Gonzalez and Woods

Butterworth High Pass Filters

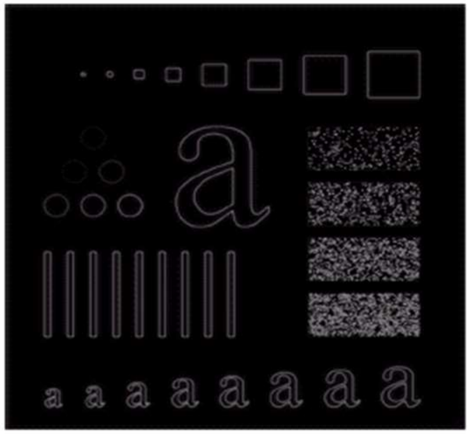




BHPL with D_0 = 30



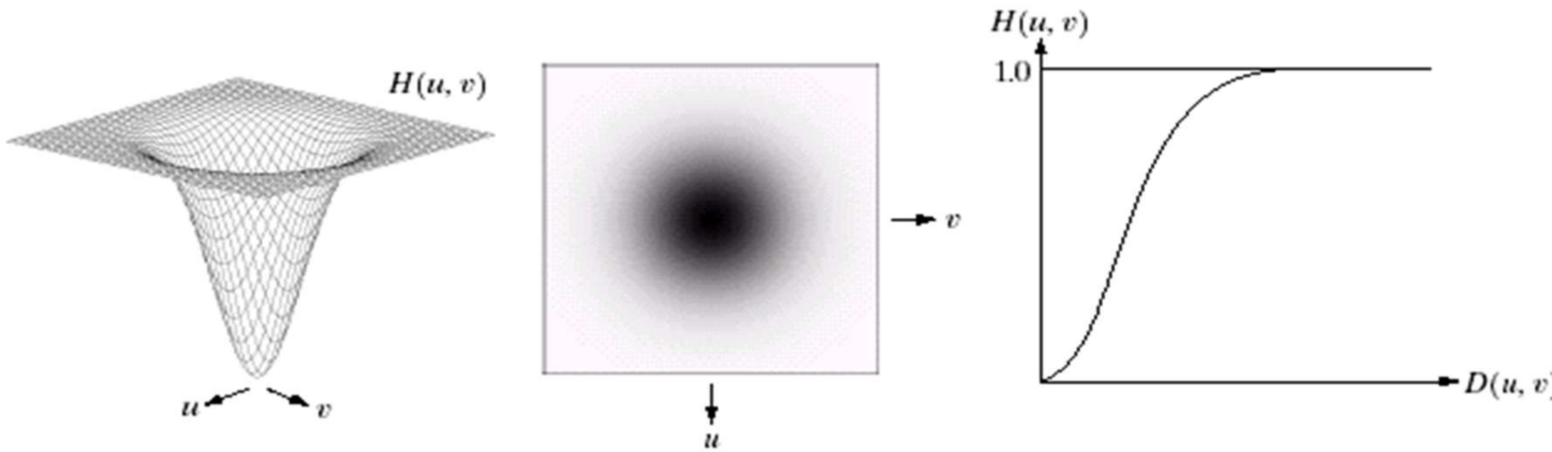
BHPF with D_0 = 60



BHPF with D_0 = 160

Image courtesy: Gonzalez and Woods

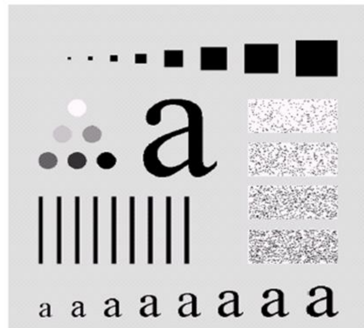
Gaussian High Pass Filters

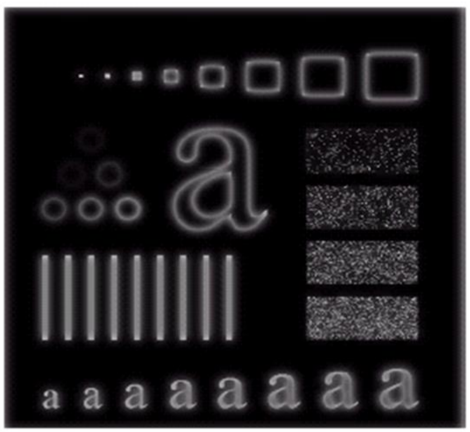


$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

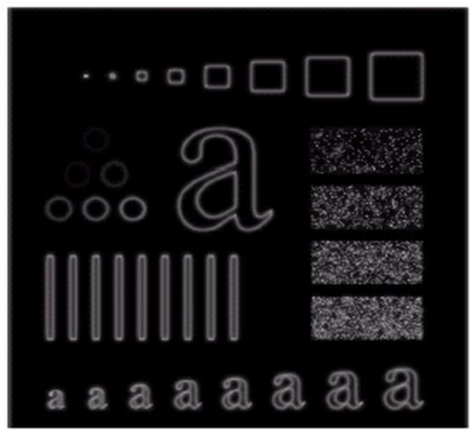
Image courtesy: Gonzalez and Woods

Gaussian High Pass Filters

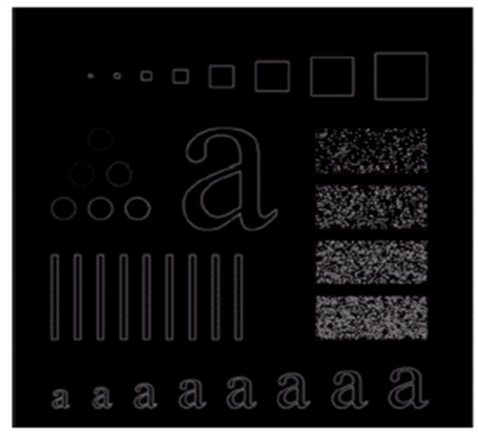




GHPL with D_0 = 30



GHPF with D_0 = 60



GHPF with D_0 = 160

Image courtesy: Gonzalez and Woods

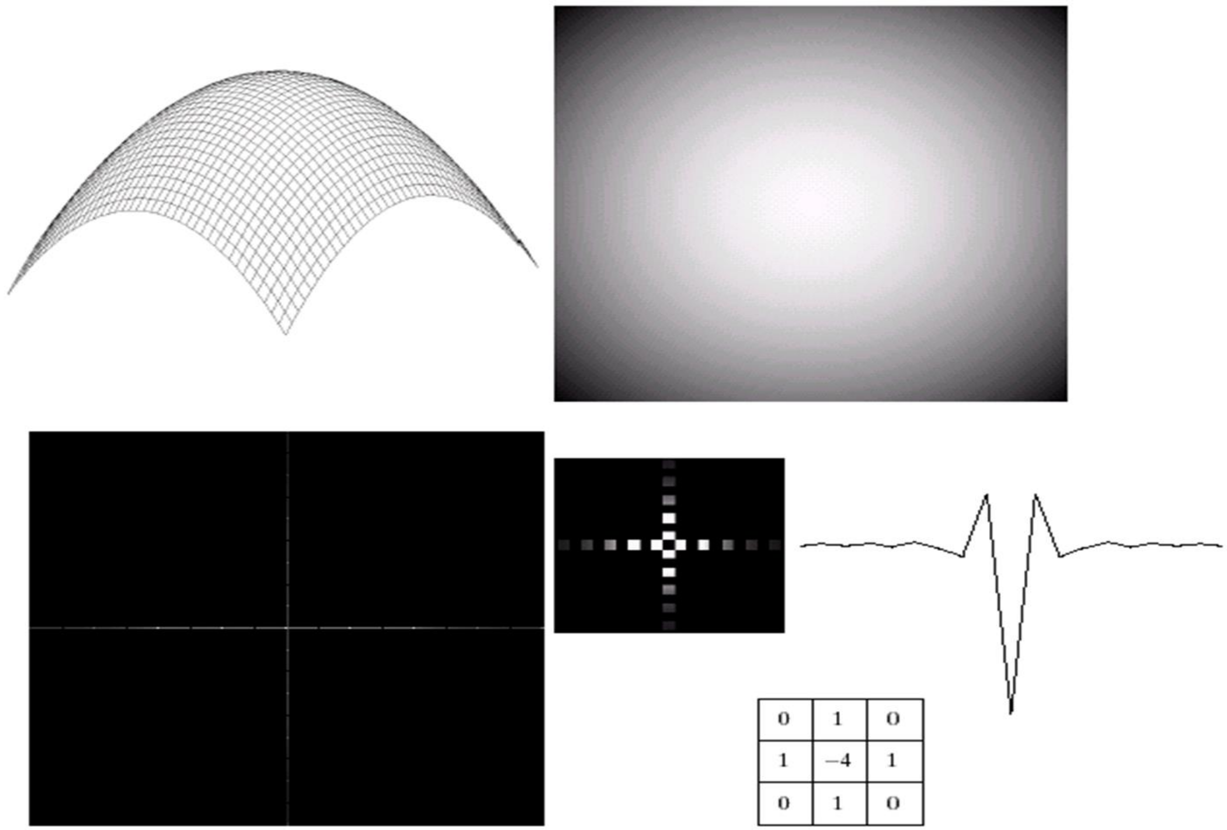
Laplacian in frequency domain

$$\Im[\frac{d^n f(x)}{dx^n}] = (ju)^n F(u)$$

$$\Im[\frac{\partial^2(f(x,y))}{\partial x^2} + \frac{\partial^2(f(x,y))}{\partial y^2}] = (ju)^2 F(u,v) + (jv)^2 F(u,v)$$

$$= -(u^2 + v^2) F(u,v)$$

Laplacian in frequency domain



0	1	0
1	−4	1
0	1	0

Image courtesy: Gonzalez and Woods

Laplacian in frequency domain

a b

FIGURE 4.58
(a) Original, blurry image.
(b) Image enhanced using the Laplacian in the frequency domain. Compare with Fig. 3.38(e).

Image courtesy: Gonzalez and Woods

Notch Reject filter (Notch pass filter)

Notch filters :

- are used to remove repetitive "Spectral" noise from an image
- are like a narrow highpass filter, but they "notch" out frequencies other than the dc component
- attenuate a selected frequency (and some of its neighbors) and leave other frequencies of the Fourier transform relatively unchanged

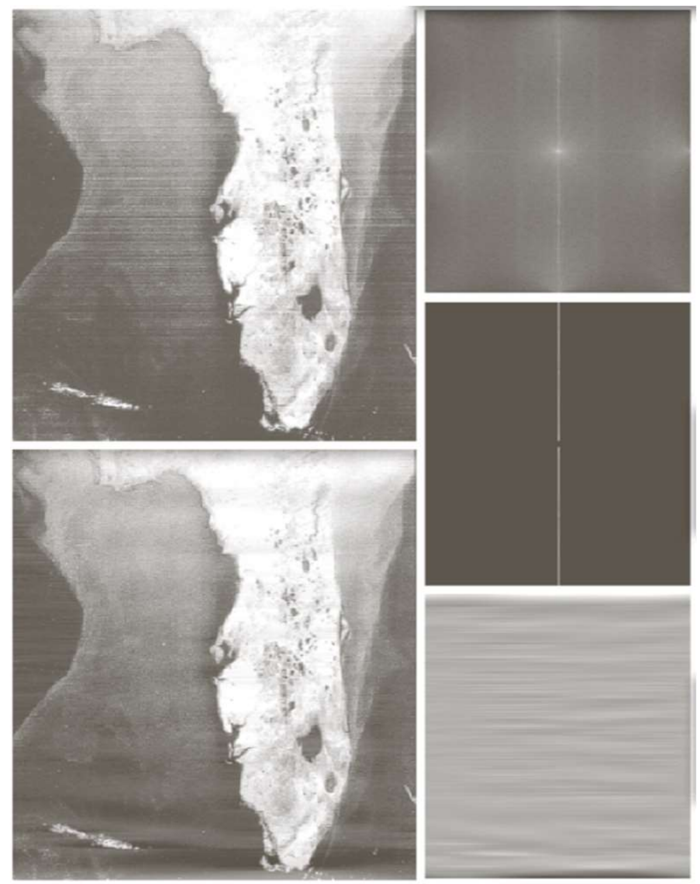


Image courtesy: Gonzalez and Woods

Filtering in frequency domain

- Band reject (Band pass filters)
- Unsharp Masking and High boost filtering
- Homomorphic filtering