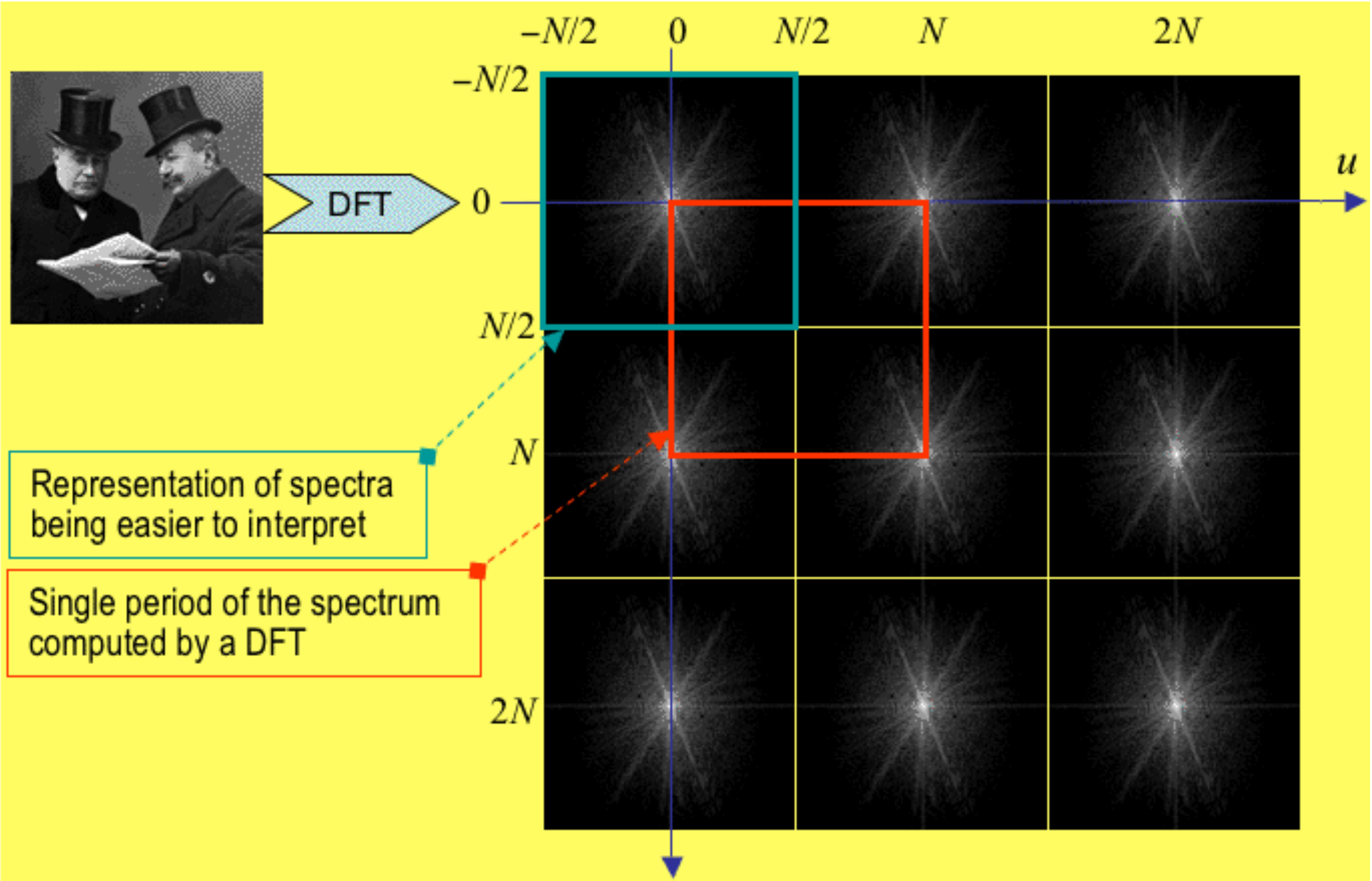
Math 3360: Mathematical Imaging

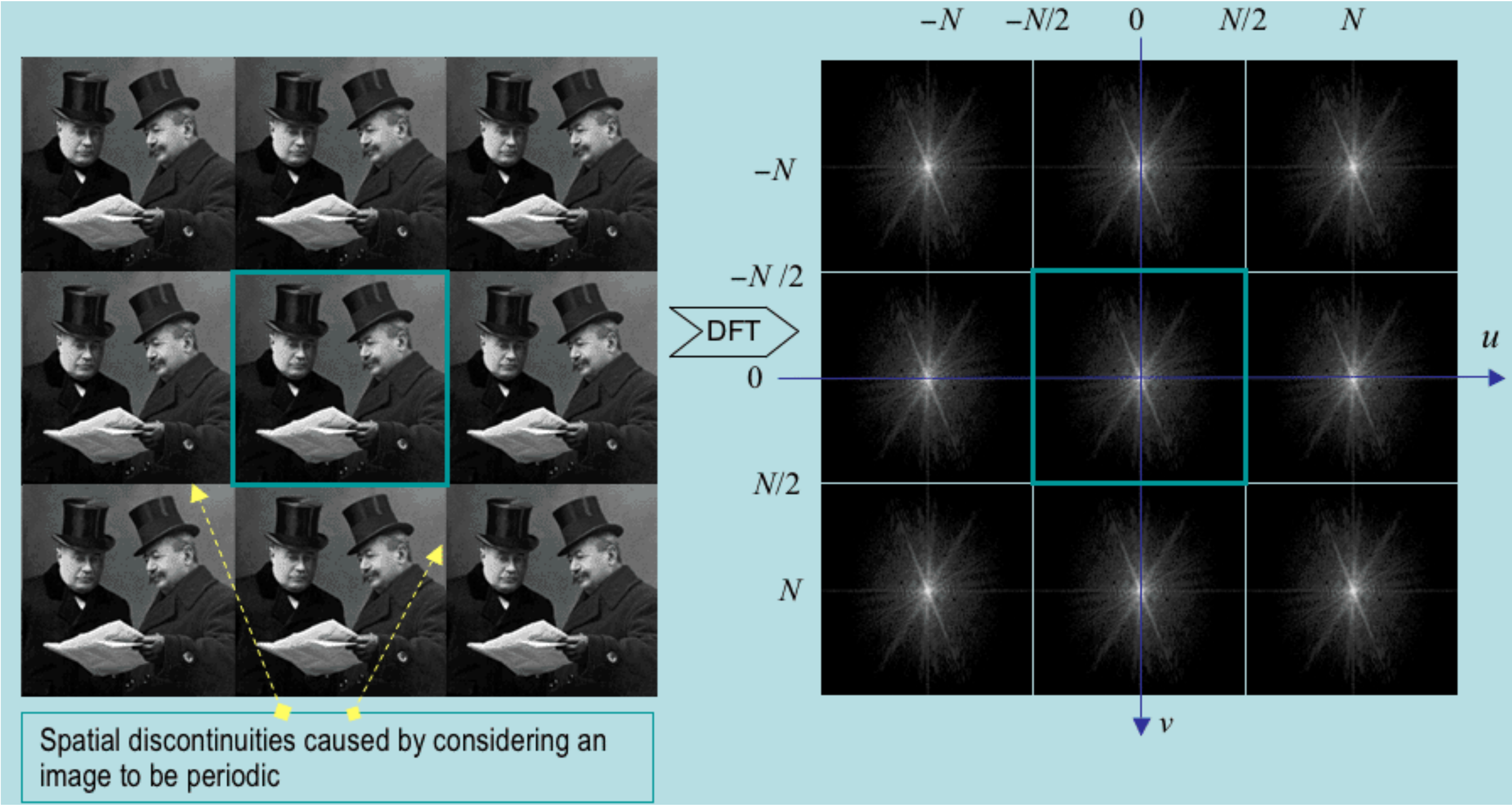
Lecture 12:
Image Denoising/Deblurring in Frequency Domain

Prof. Ronald Lok Ming Lui
Department of Mathematics,
The Chinese University of Hong Kong

Frequency spectrum of an image

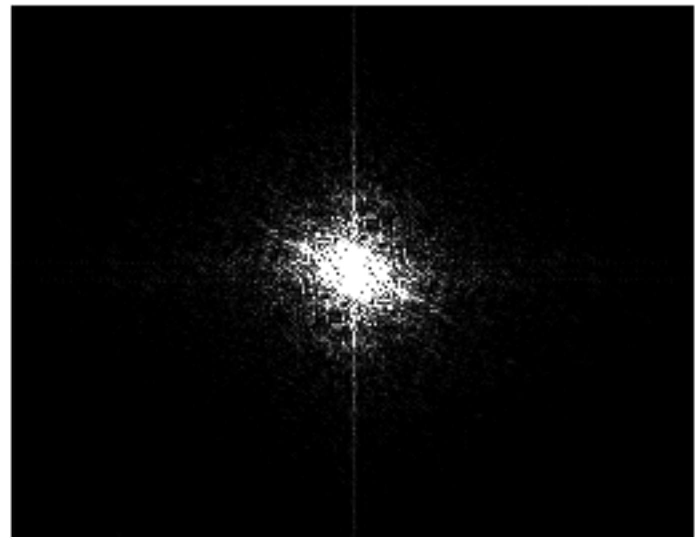


Frequency spectrum of an image



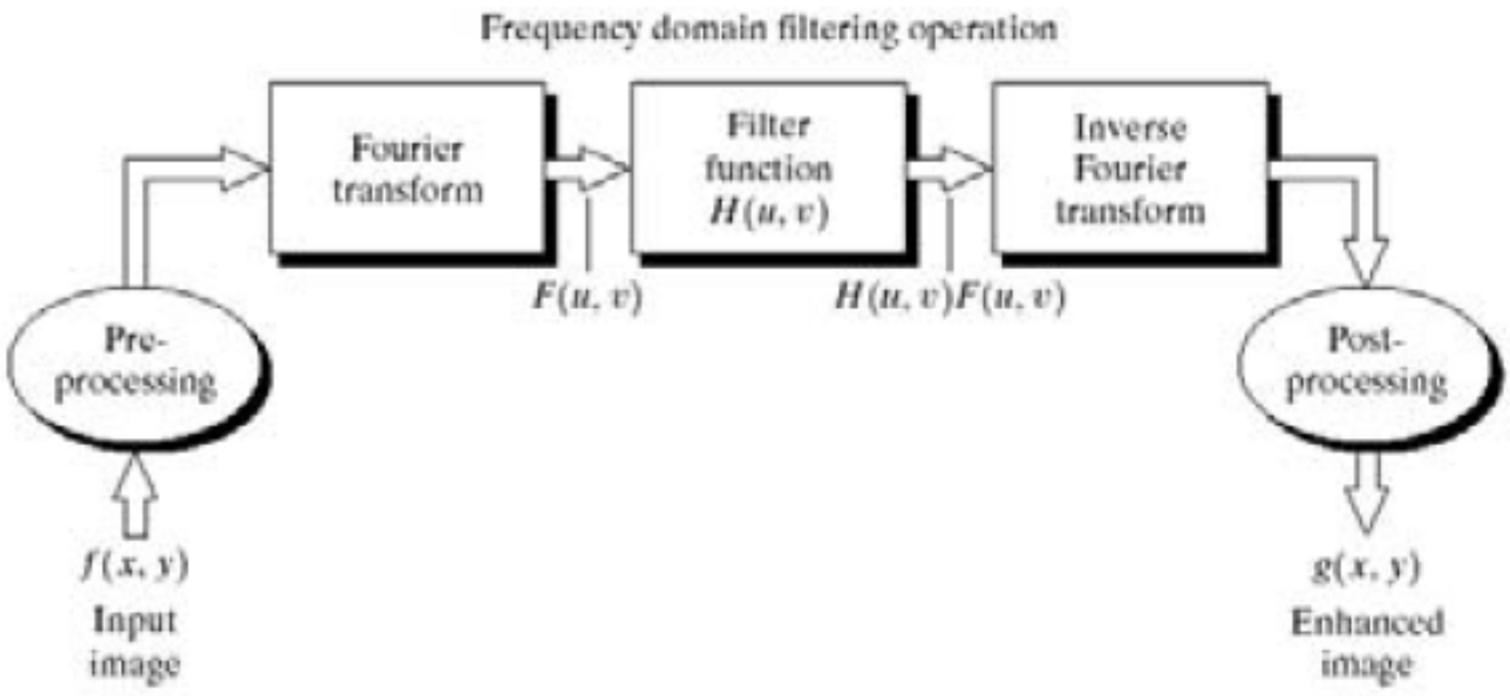
Frequency spectrum of an image

Original image

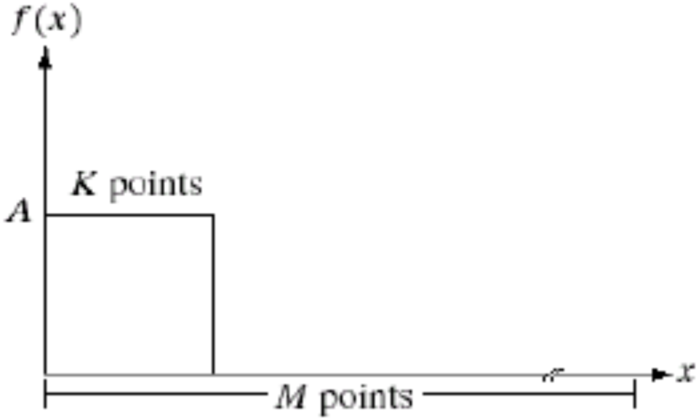


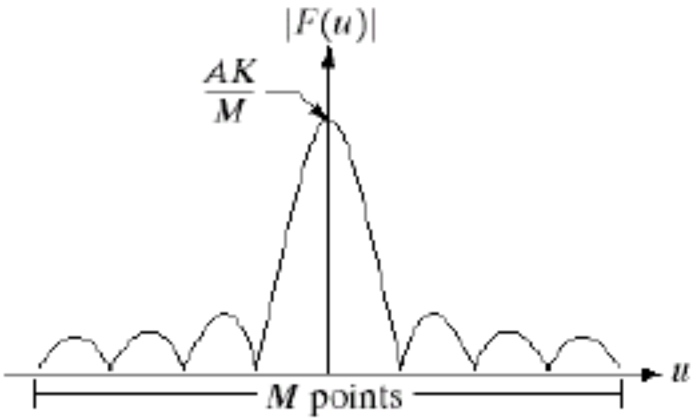
Spectrum

Key steps for image enhancement in the frequency domain

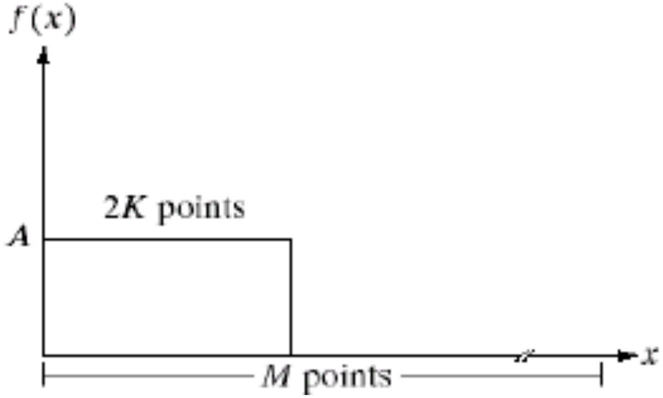


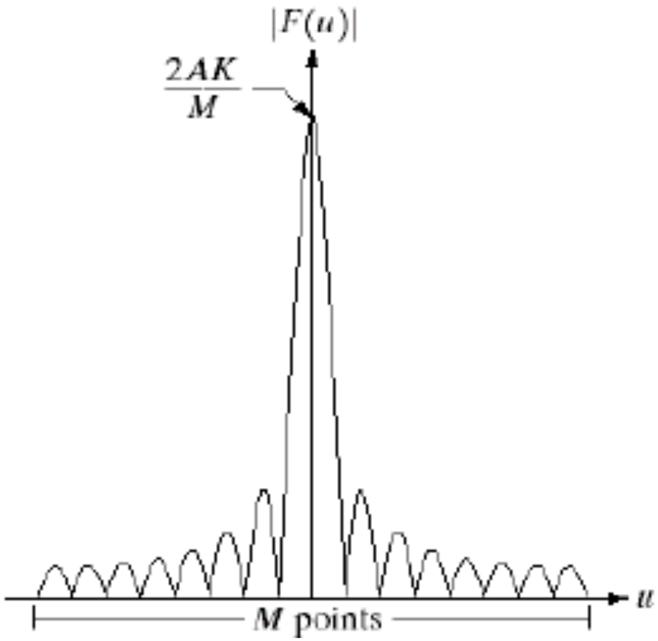
Relationship between spatial and frequency domain



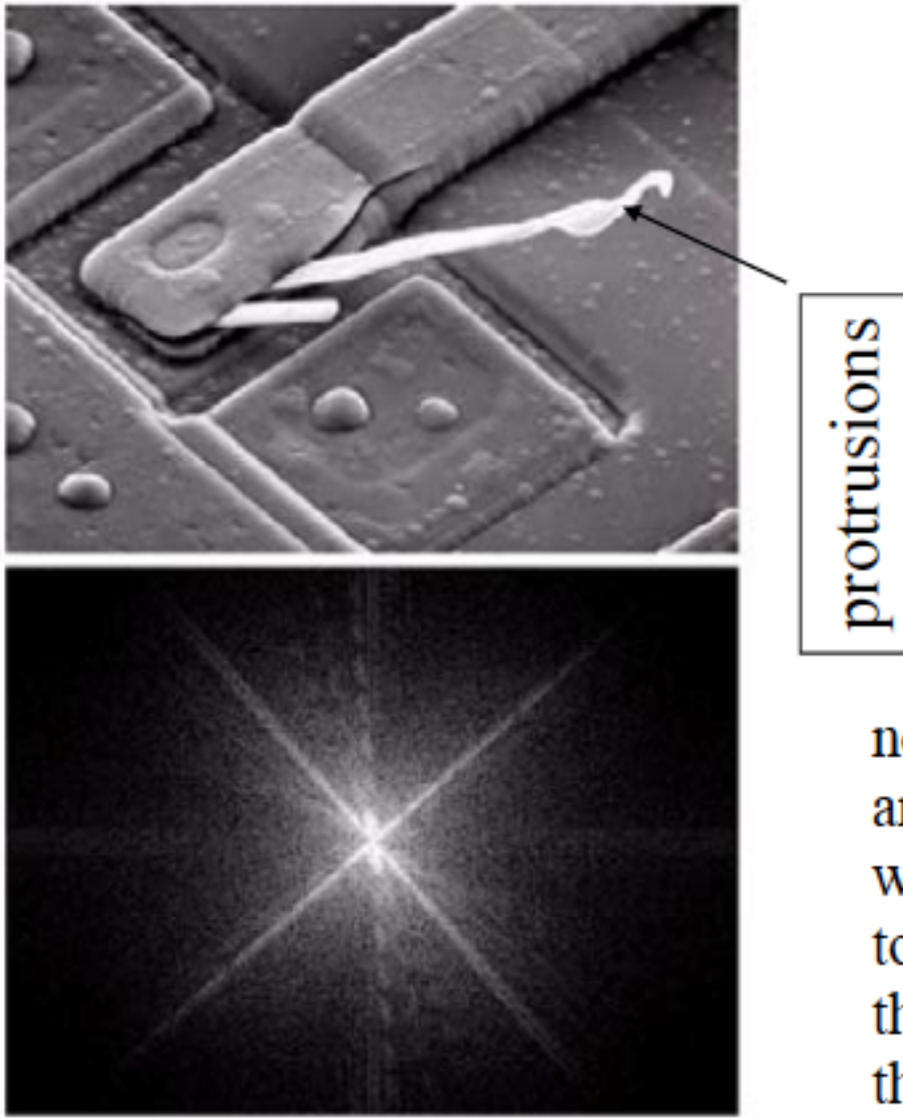


Flat filter **Low pass filtering**





Spatial and frequency domain



SEM: scanning electron Microscope

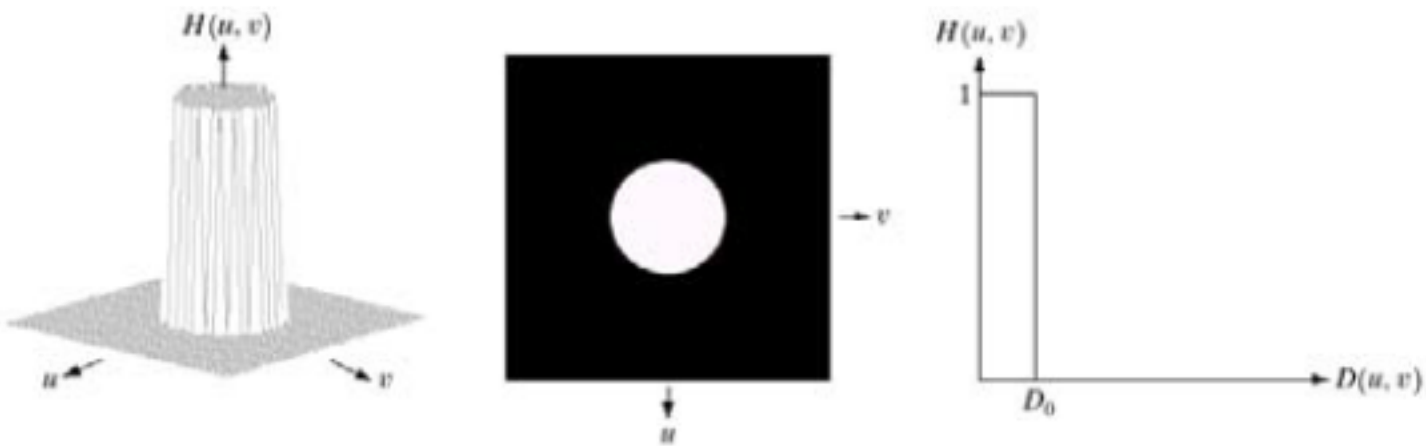
(a) SEM image of a damaged integrated circuit. (b) Fourier spectrum of (a). (Original image courtesy of Dr. J. M. Hudak, Brockhouse Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada.)

notice the ±45° components and the vertical component which is slightly off-axis to the left! It corresponds to the protrusion caused by thermal failure above.

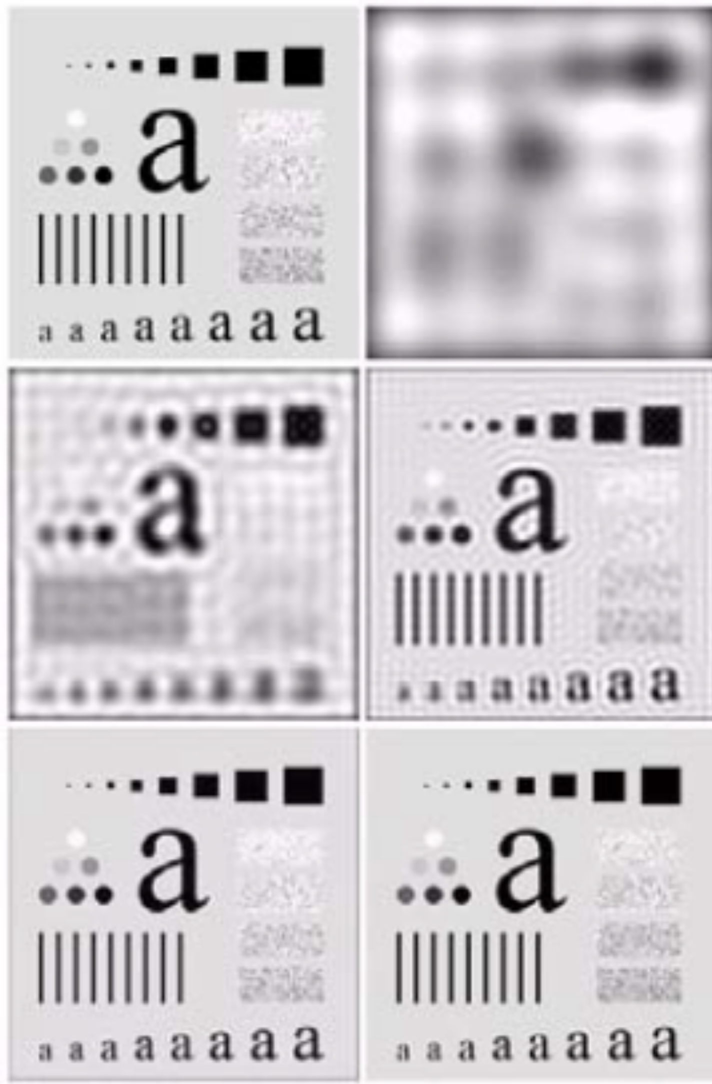
Ideal Low Pass Filter

Ideal low-pass filter $$H(u,v)=\begin{cases}1 & \text{if } D(u,v)\le D_0\\ 0 & \text{if } D(u,v)\succ D_0\end{cases}$$

D_0 is the cutoff frequency and D(u,v) is the distance between (u,v) and the frequency origin.



Ideal Low Pass Filter

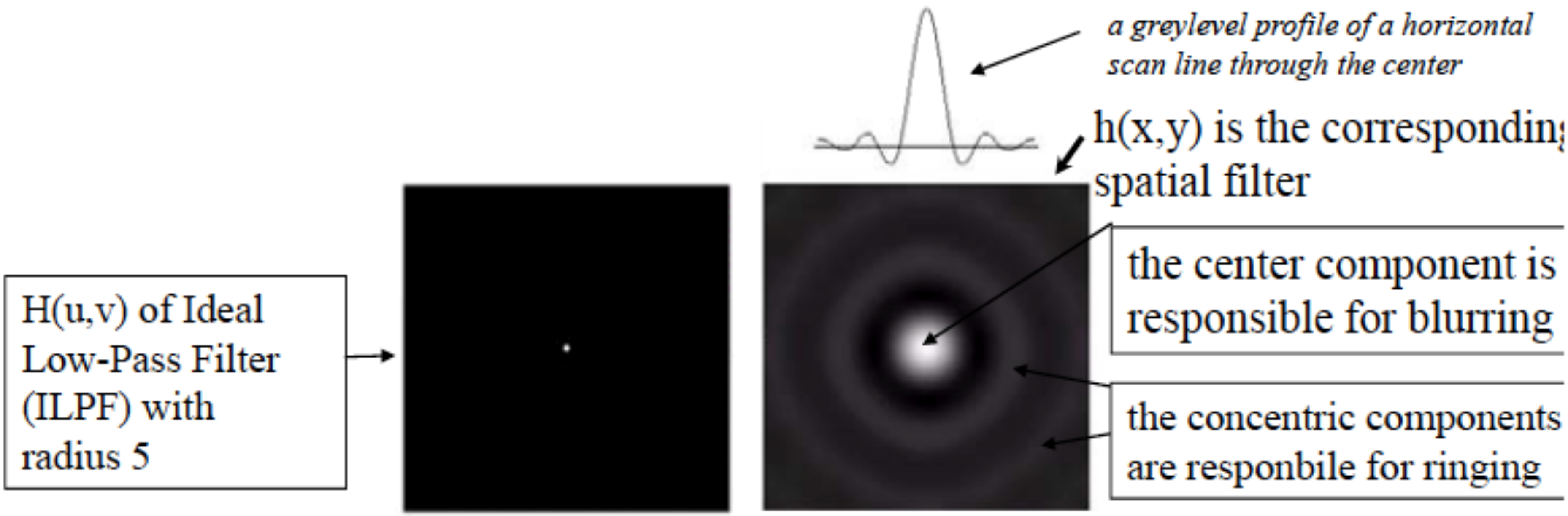


useless, even though only 8% of image power is lost!

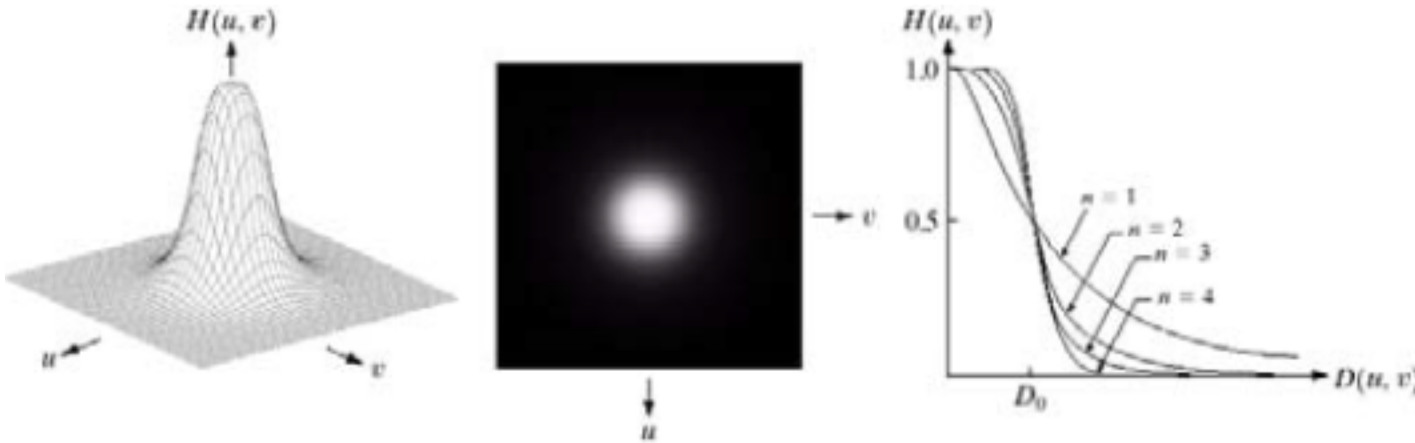
notice both blurring and ringing!

Ideal Low Pass Filter with larger and larger radii D0

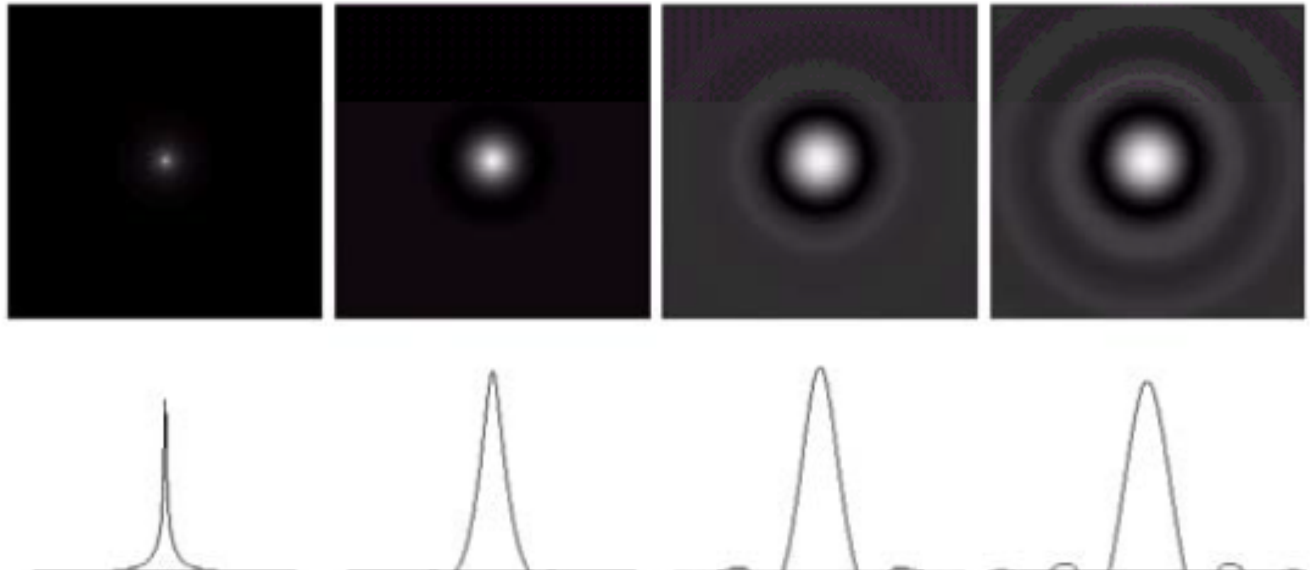
Explanation of ringing effect



Butterworth Low Pass Filter

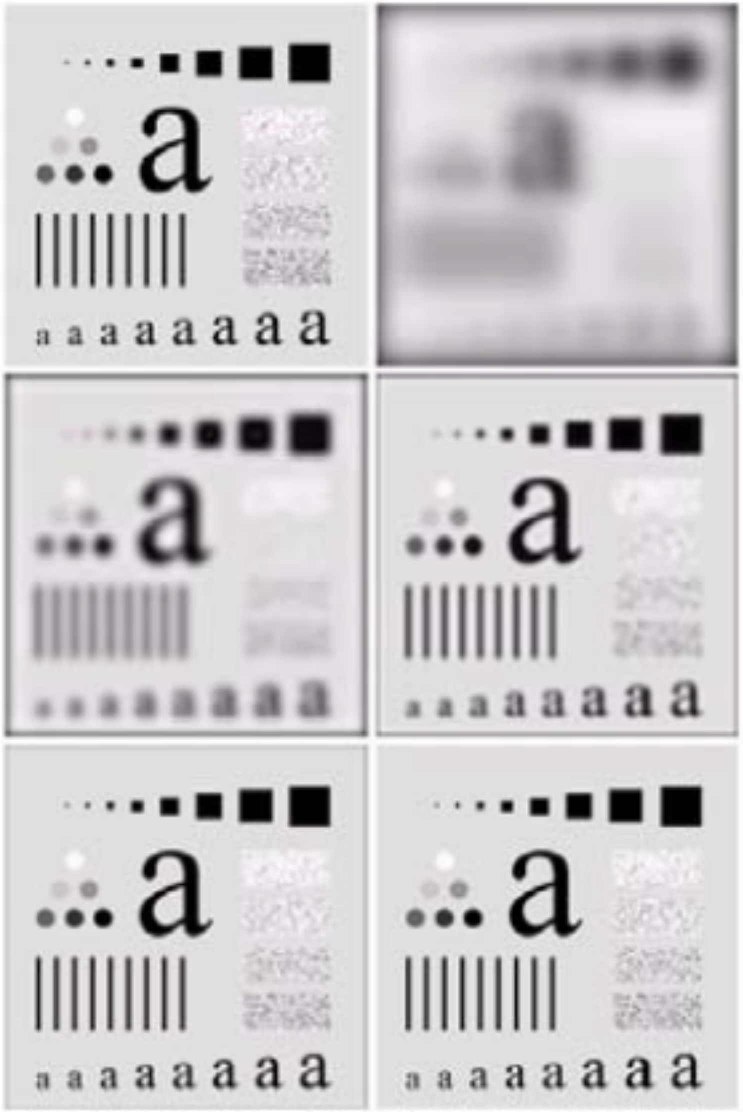


$$H(u,v)=\frac{1}{1+[D(u,v)/D_0]^{n}}$$



n=1 n=2 n=5 n=20

Butterworth Low Pass Filter

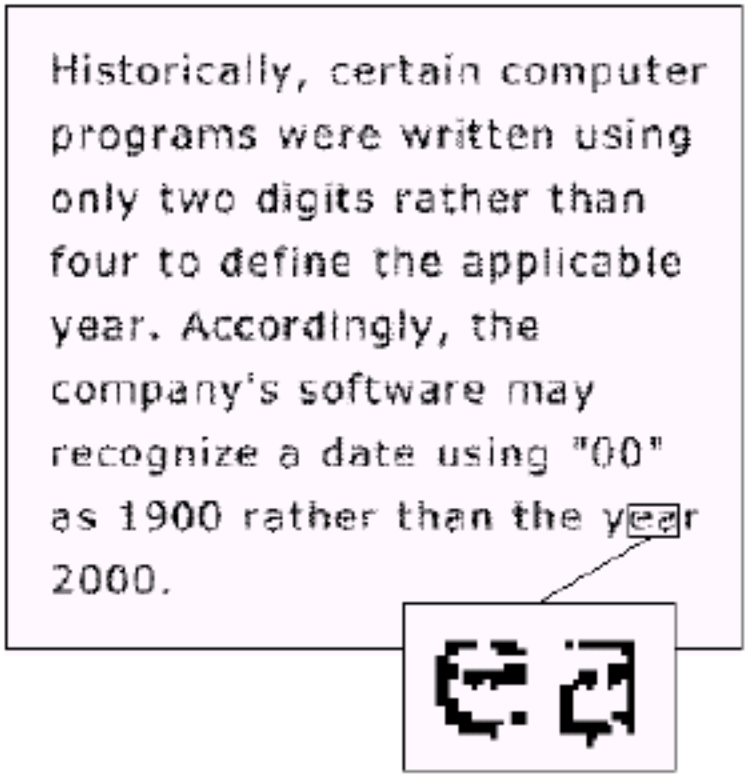


Butterworth Low Pass Filter with larger and larger radii D0

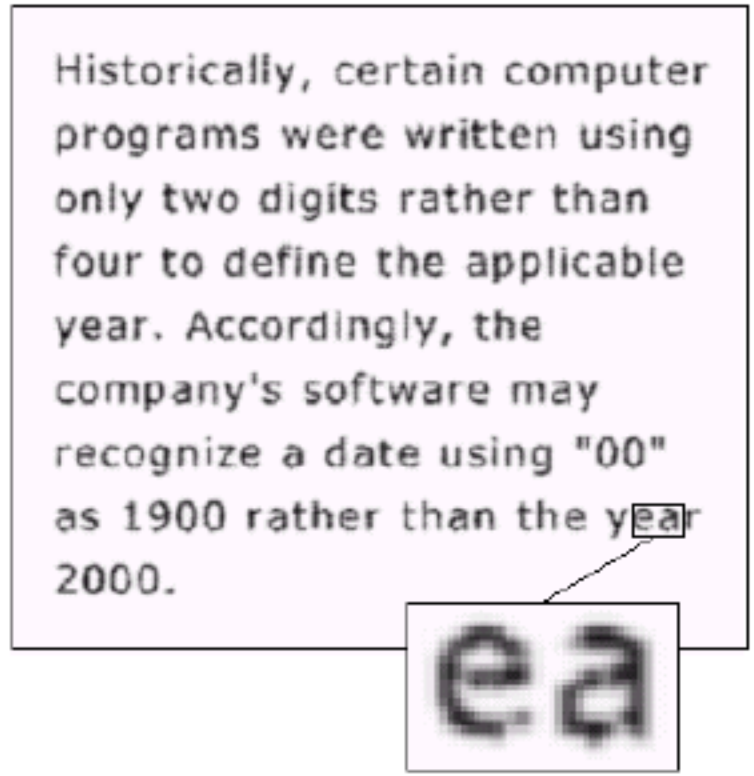
Gau … **)**

Gaussian Low Pass Filter

Applications: fax transmission, duplicated documents and old records.



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.

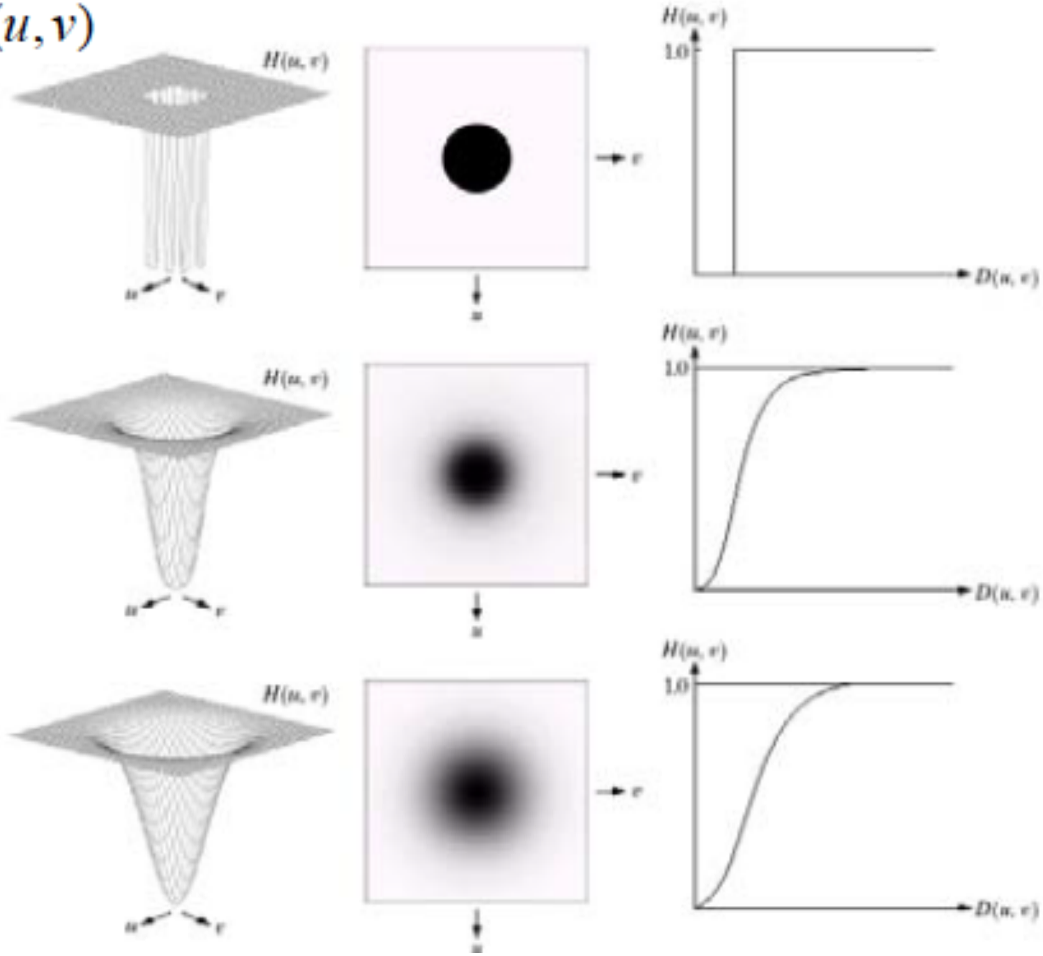
GLPF with D_0=80 is used.

Application: Low Pass Filter

A LPF is also used in printing, e.g. to smooth fine skin lines in faces.

High Pass Filter

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

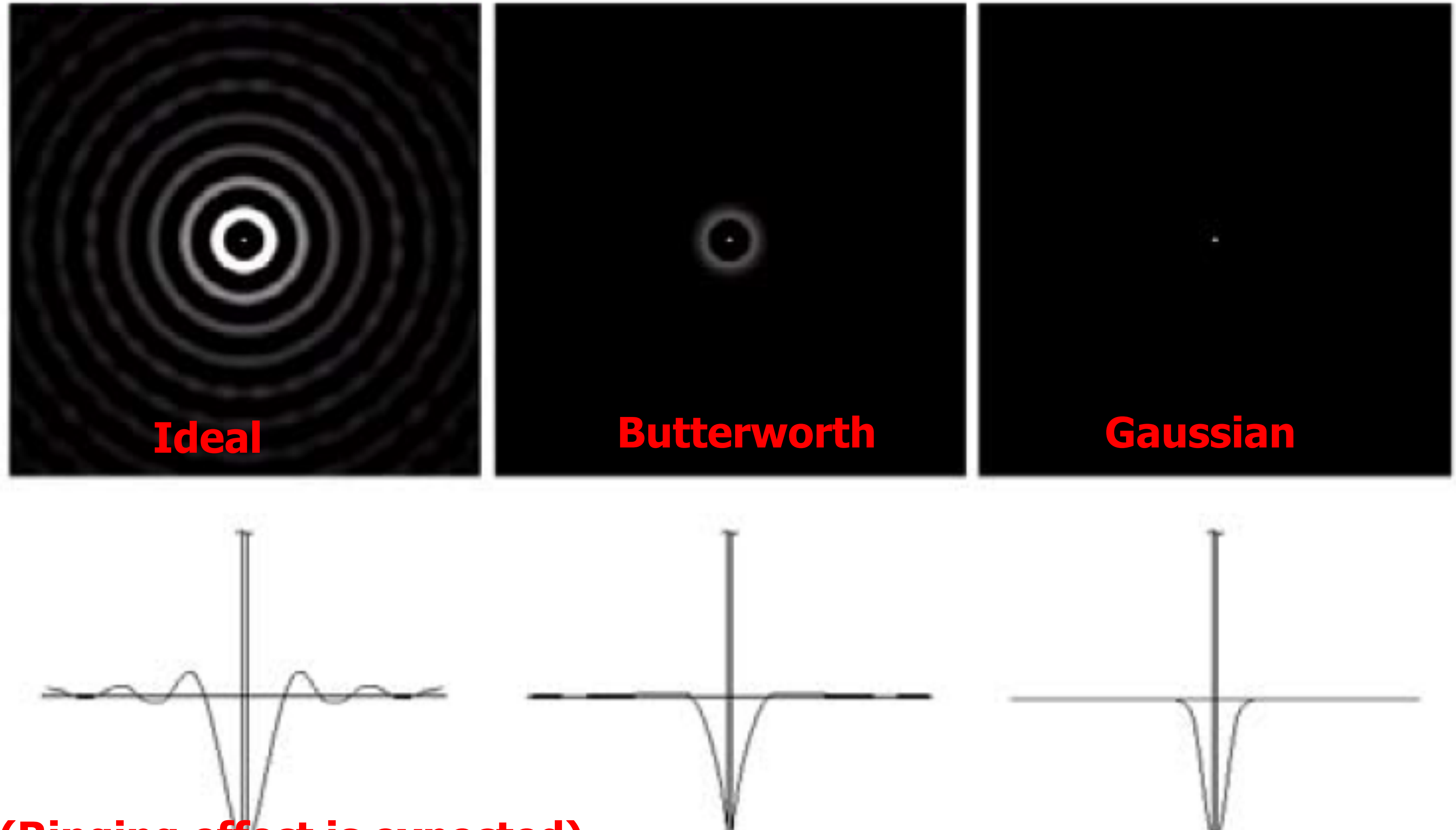


ideal high-pass filter

Butterworth high-pass

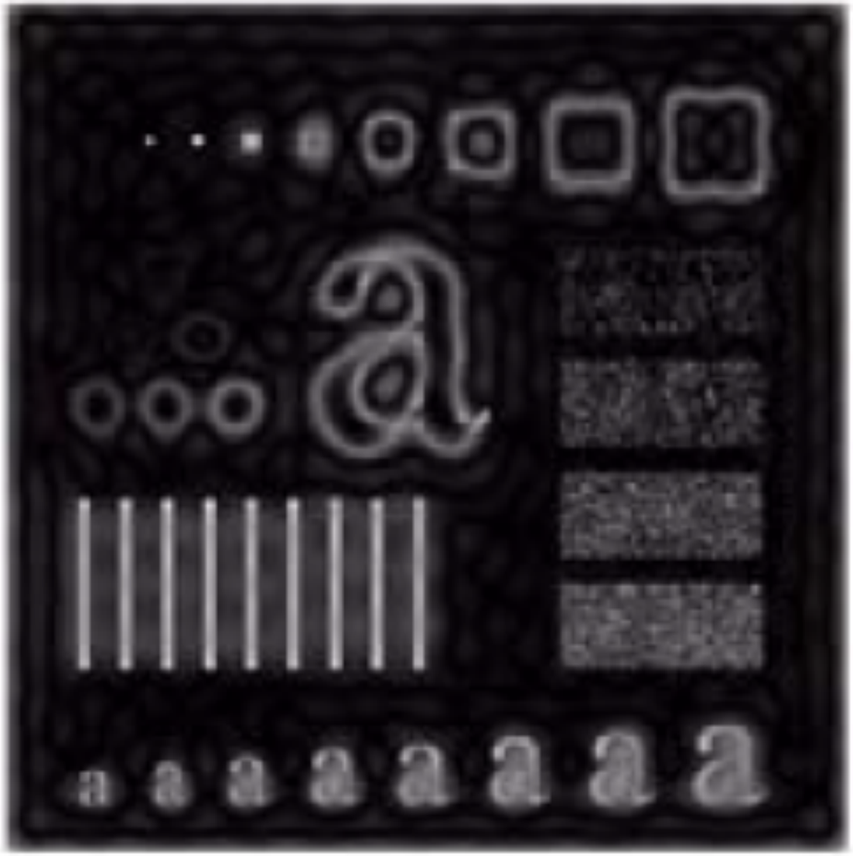
Gaussian high-pass

Spatial representation of High Pass Filter

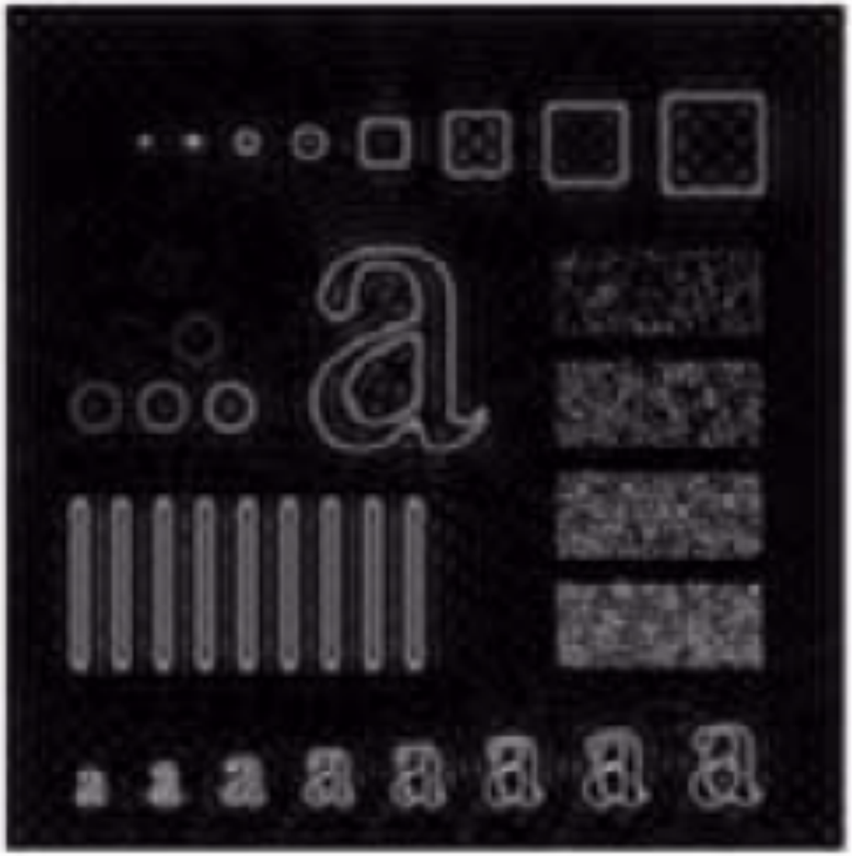


(Ringing effect is expected)

Ideal High Pass Filter



D0 = 15

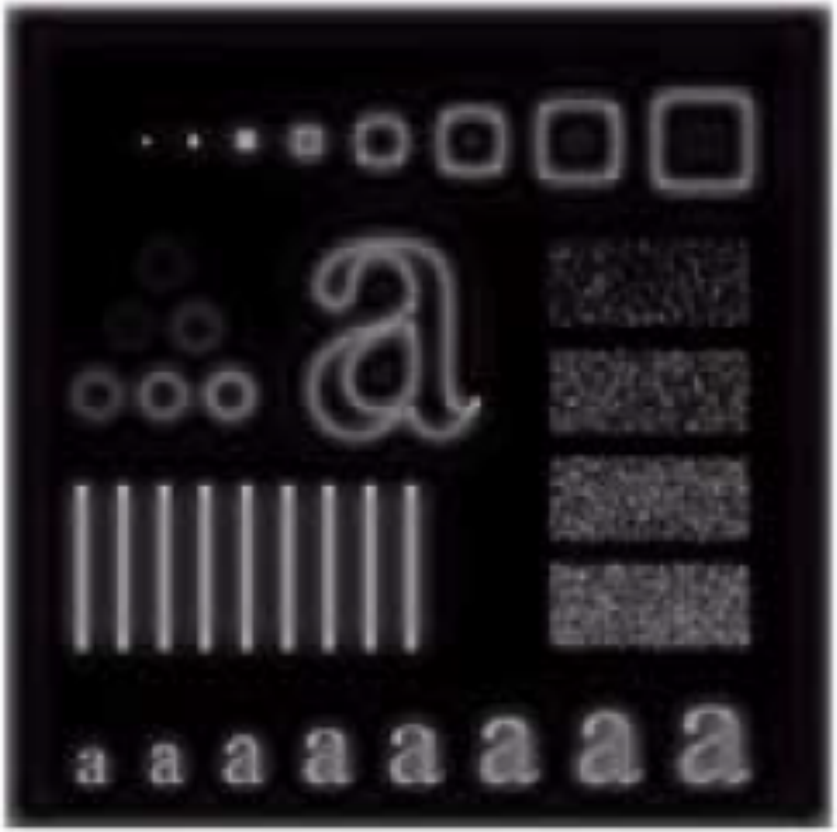


D0 = 30

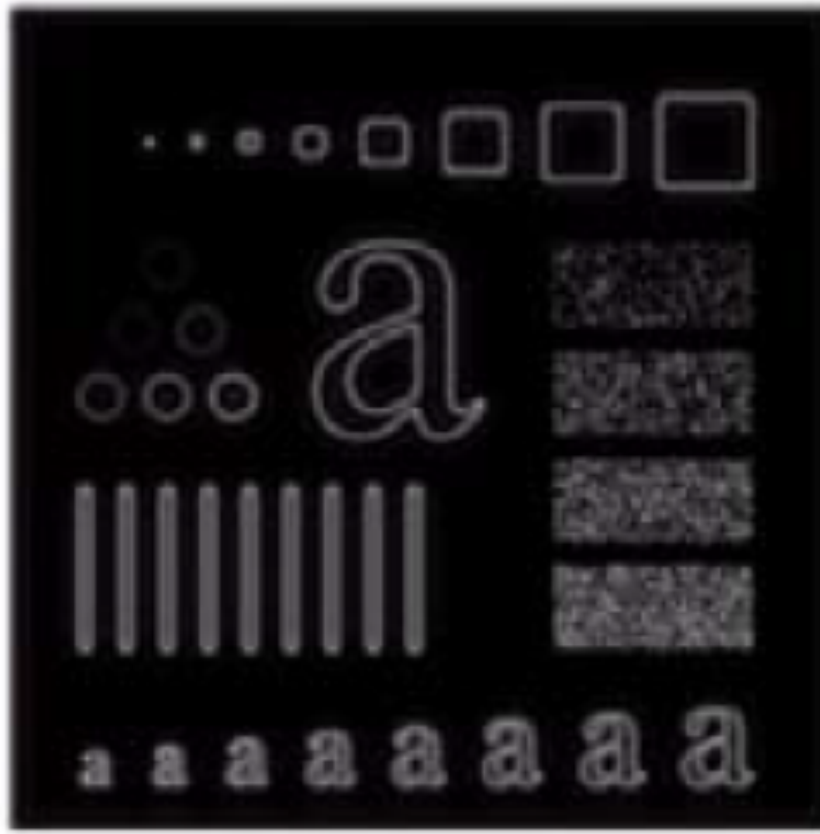
(Ringing effect is observed)

Ideal High Pass Filter

Butterworth High Pass Filter



D0 = 15



D0 = 30

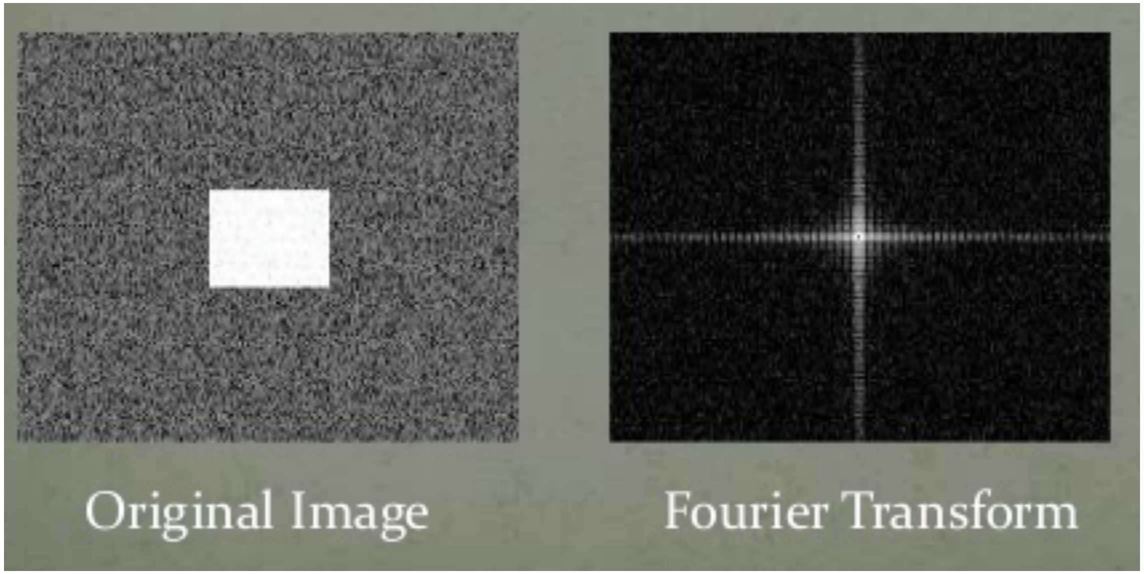
Comparison: High Pass Filter

High-pass filtering

Low-pass filtering

Image denoising examples

Ideal filtering



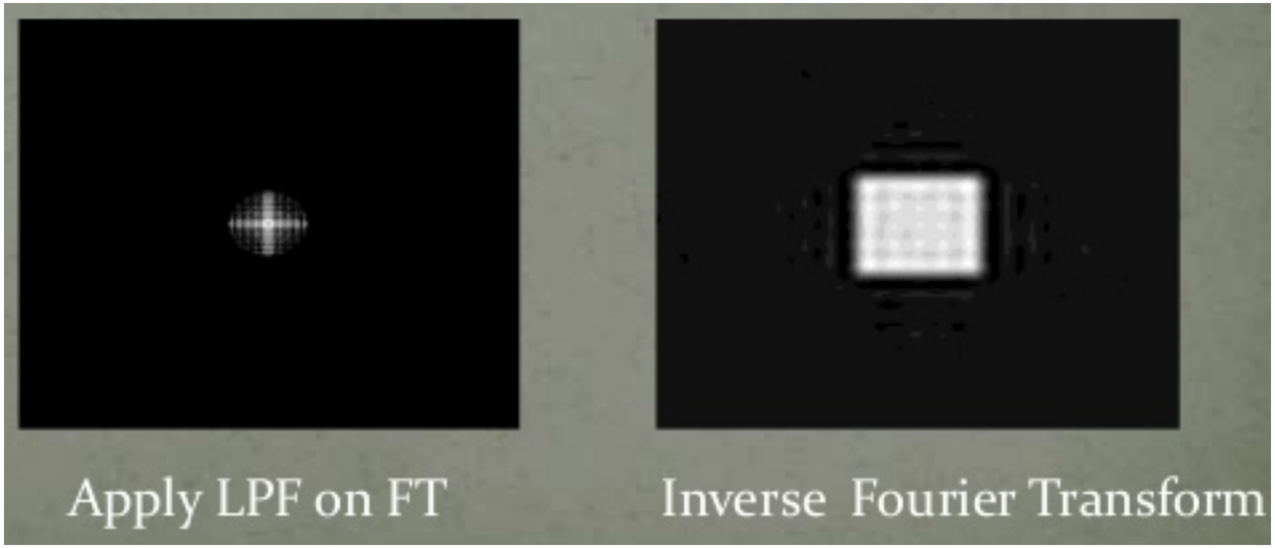
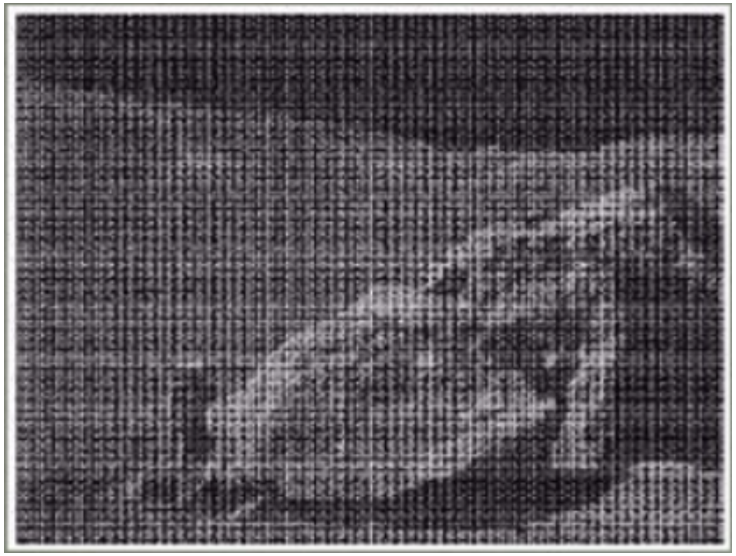
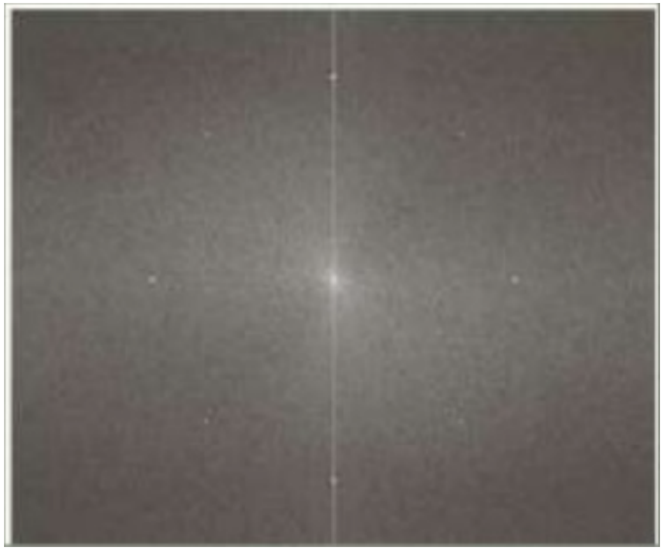


Image denoising examples

Ideal filtering



Noisy image



Frequency domain

Denoised

Image denoising examples

Butterworth filtering

Noisy

Image denoising examples

Butterworth filtering

Denoised

Image denoising examples

Gaussian filtering

noisy

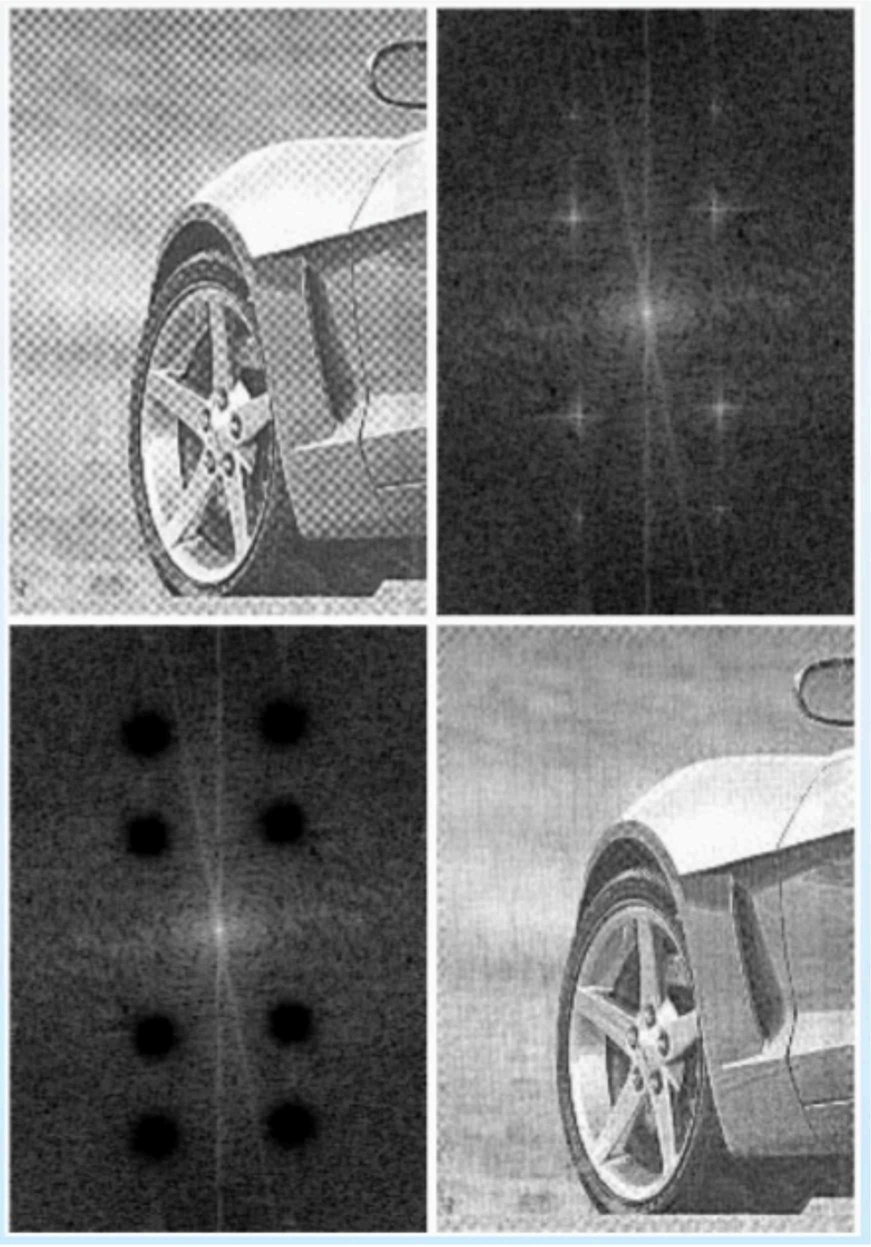
denoised

(σ=1)

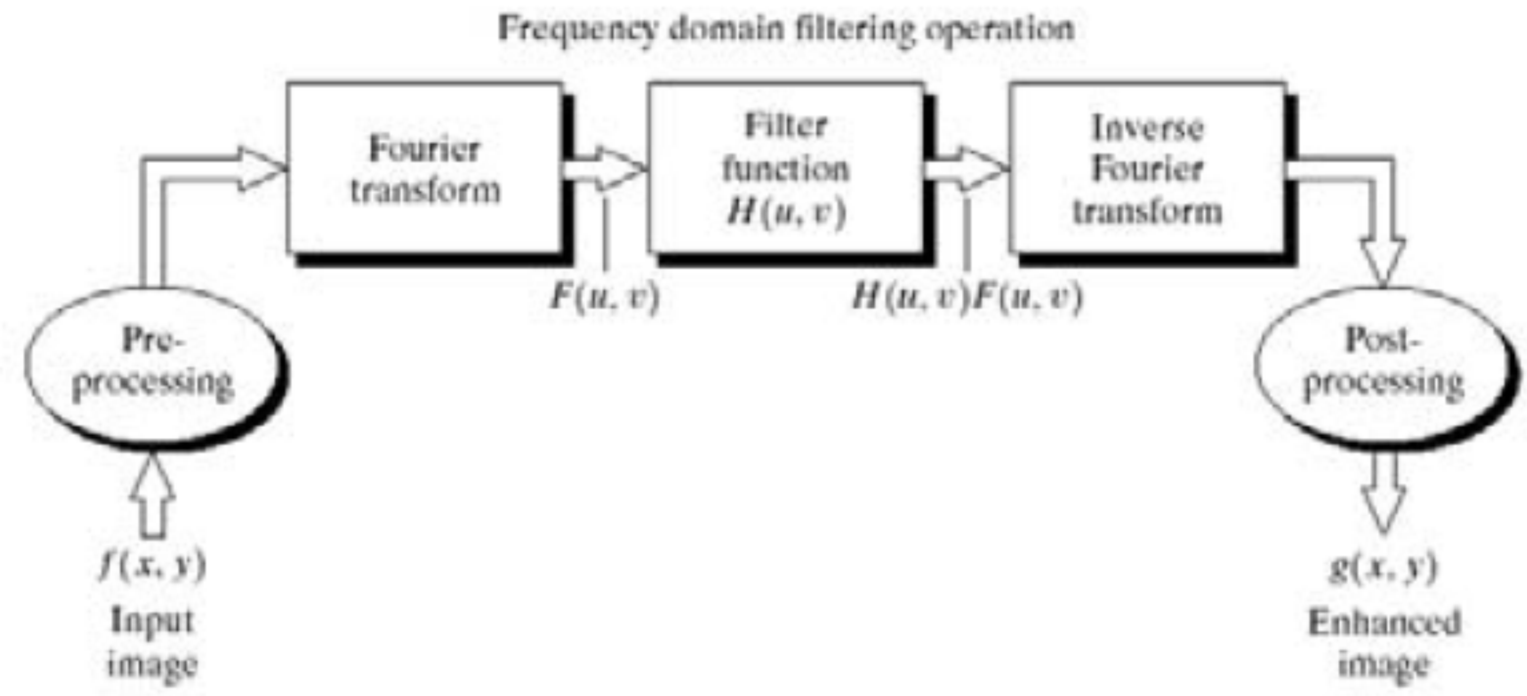
denoised

(σ=1.5)

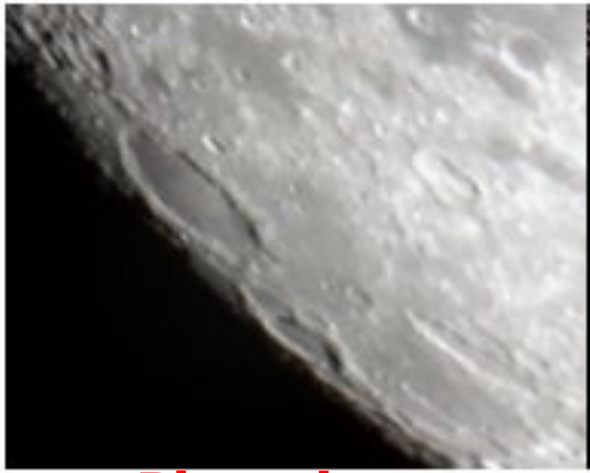
Image denoising examples



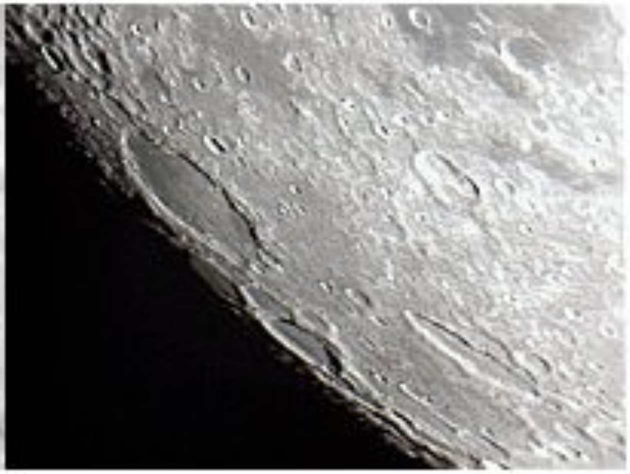
Key steps for image enhancement in the frequency domain



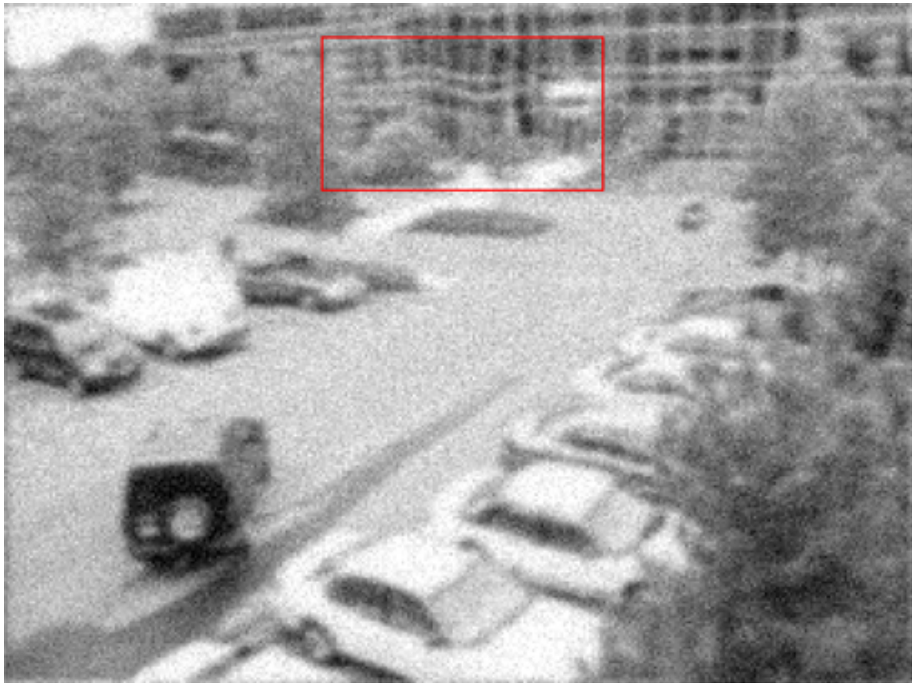
Example of turbulence blur



Blurry image

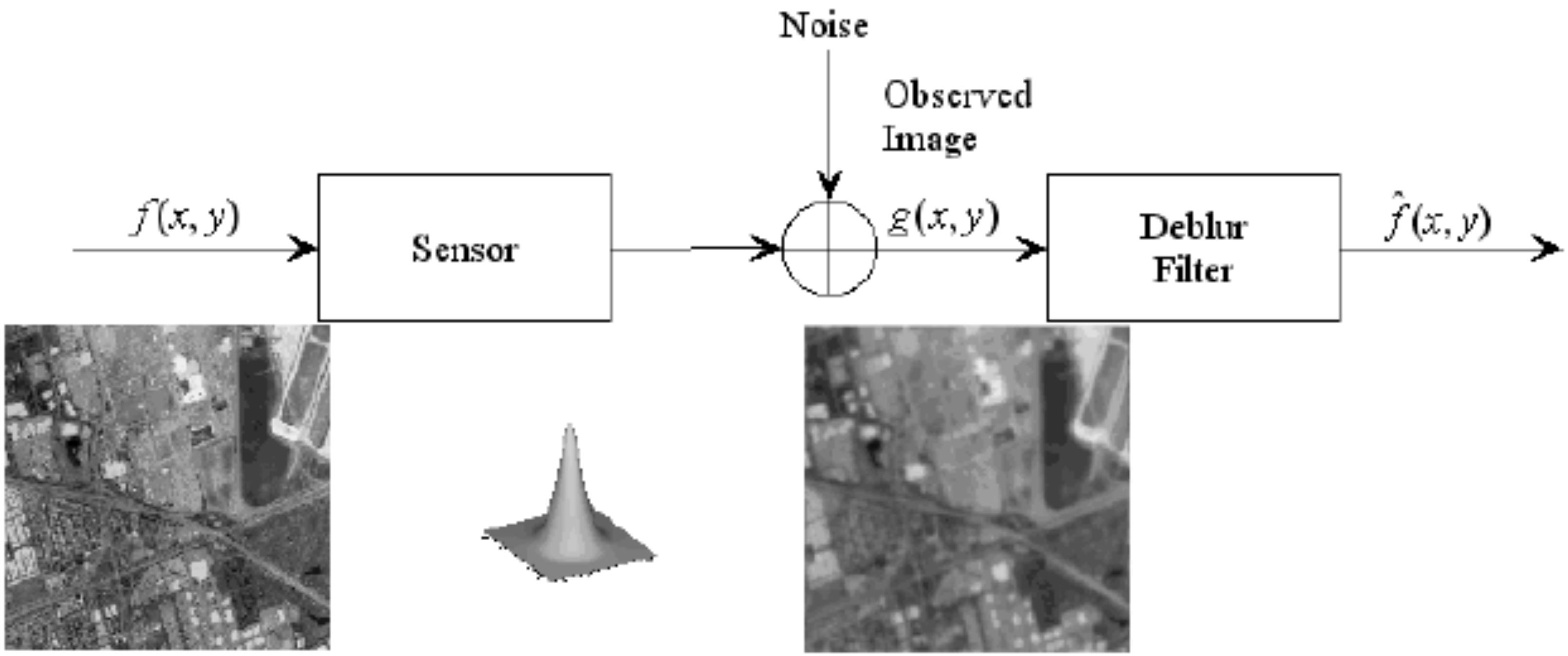


Deblurred image



Example of motion blur

Image deblur model



Linear model of observation system

$$g(x,y) = f(x,y) \star h(x,y) + \eta(x,y)$$

Image filtering in the frequency domain

The observation equation can also be expressed in the frequency domain as

$$G(u,v) = F(u,v)H(u,v) + \mathcal{N}(u,v)$$

We can construct an estimate of $F(u,v)$ by filtering the observation $G(u,v)$. Let $T(u,v)$ be a linear shift-invariant reconstruction filter.

$$\hat{F}(u,v) = G(u,v)T(u,v)$$

Our task is to find a filter $T(u,v)$ that provides a good estimate of the original image.

The solution must balance noise reduction and sharpening of the image. These are conflicting goals.

Direct inverse filter

$$T(u,v) = H^{-1}(u,v)$$

$$\hat{F}(u,v) = G(u,v)H^{-1}(u,v) = F(u,v) + \mathcal{N}(u,v)H^{-1}(u,v)$$

The result will be filtered noise added to the desired image.

The problem is that the inverse filter typically has very high gain at certain frequencies so that the noise term completely dominates the result.

Direct inverse filter

Original Image

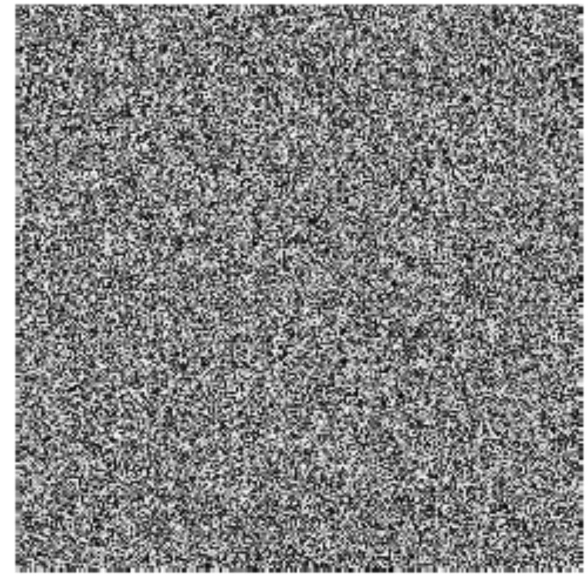
Blurred Image

A small amount of noise saturates the inverse filter.

Direct inverse filter

Original Image

Blurred Image



Restored with $H^{-1}(u,v)$

A small amount of noise saturates the inverse filter.

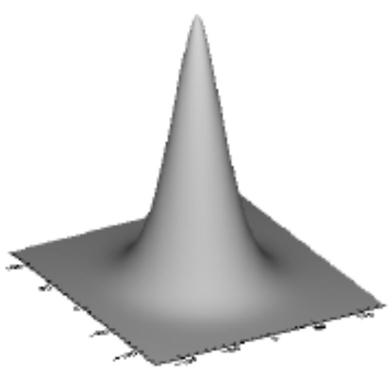
Modified inverse filter

$$B(u,v) = \frac{1}{1 + \left(\frac{u^2+v^2}{D^2}\right)^n}$$

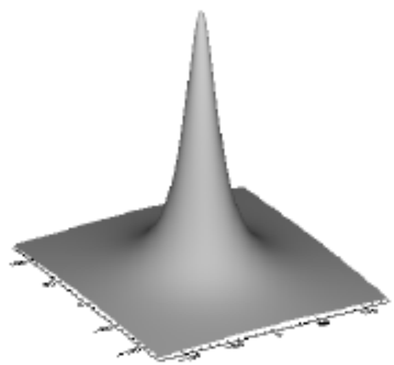
$$T(u,v) = \frac{B(u,v)}{H(u,v)}$$

$$\begin{aligned}\hat{F}(u,v) &= (F(u,v)H(u,v) + \mathcal{N}(u,v))T(u,v) \\ &= F(u,v)B(u,v) + \frac{\mathcal{N}(u,v)B(u,v)}{H(u,v)}\end{aligned}$$

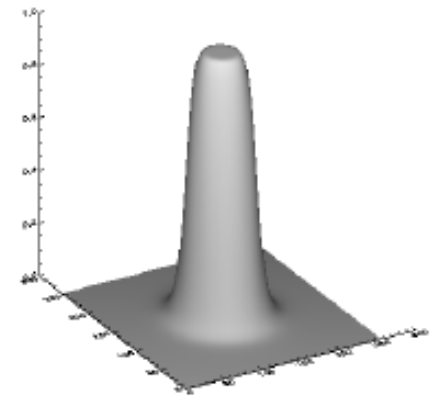
Modified inverse filter



$H(u, v)$



$B(u, v)$: $R = 20, n = 1$



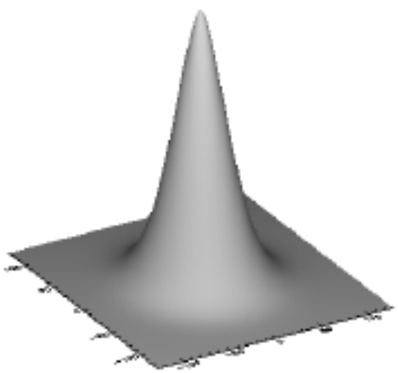
$R = 40, n = 1$

Blurred Image $G(u, v)$

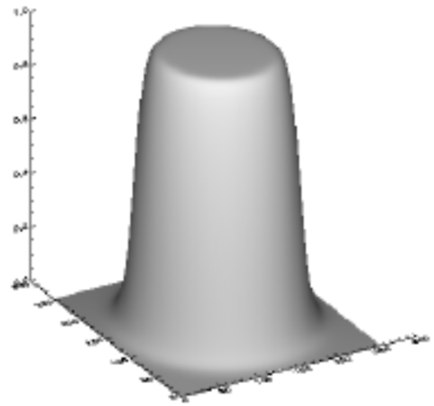
Restored using $R = 20$

Restored using $R = 40$

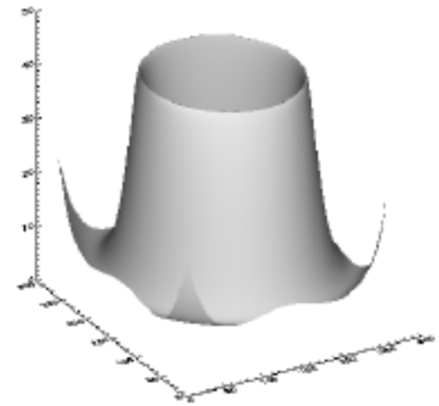
Modified inverse filter



$H(u, v)$



$B(u, v)$: $R = 90, n = 8$



Inverse B/H

Original Image $G(u, v)$

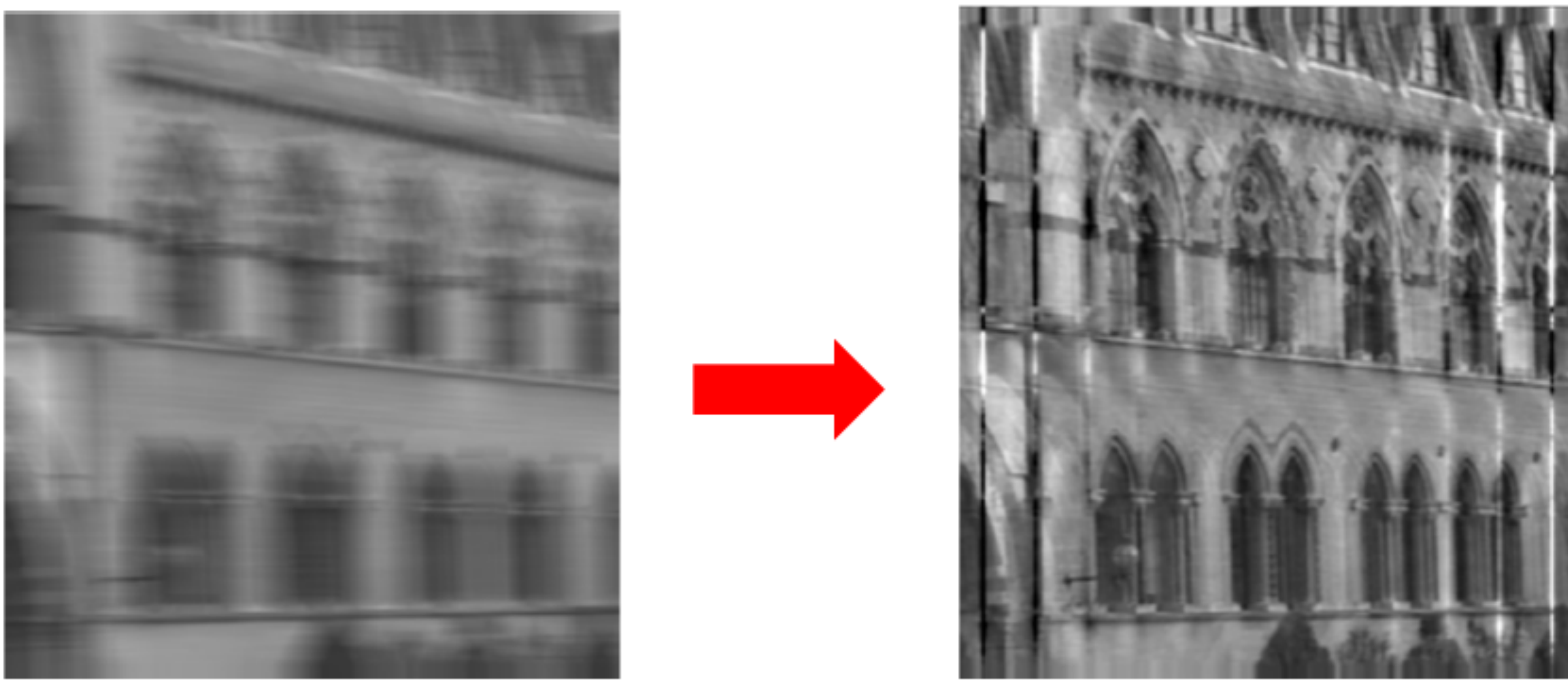
Blurred using $R = 20$

Restored

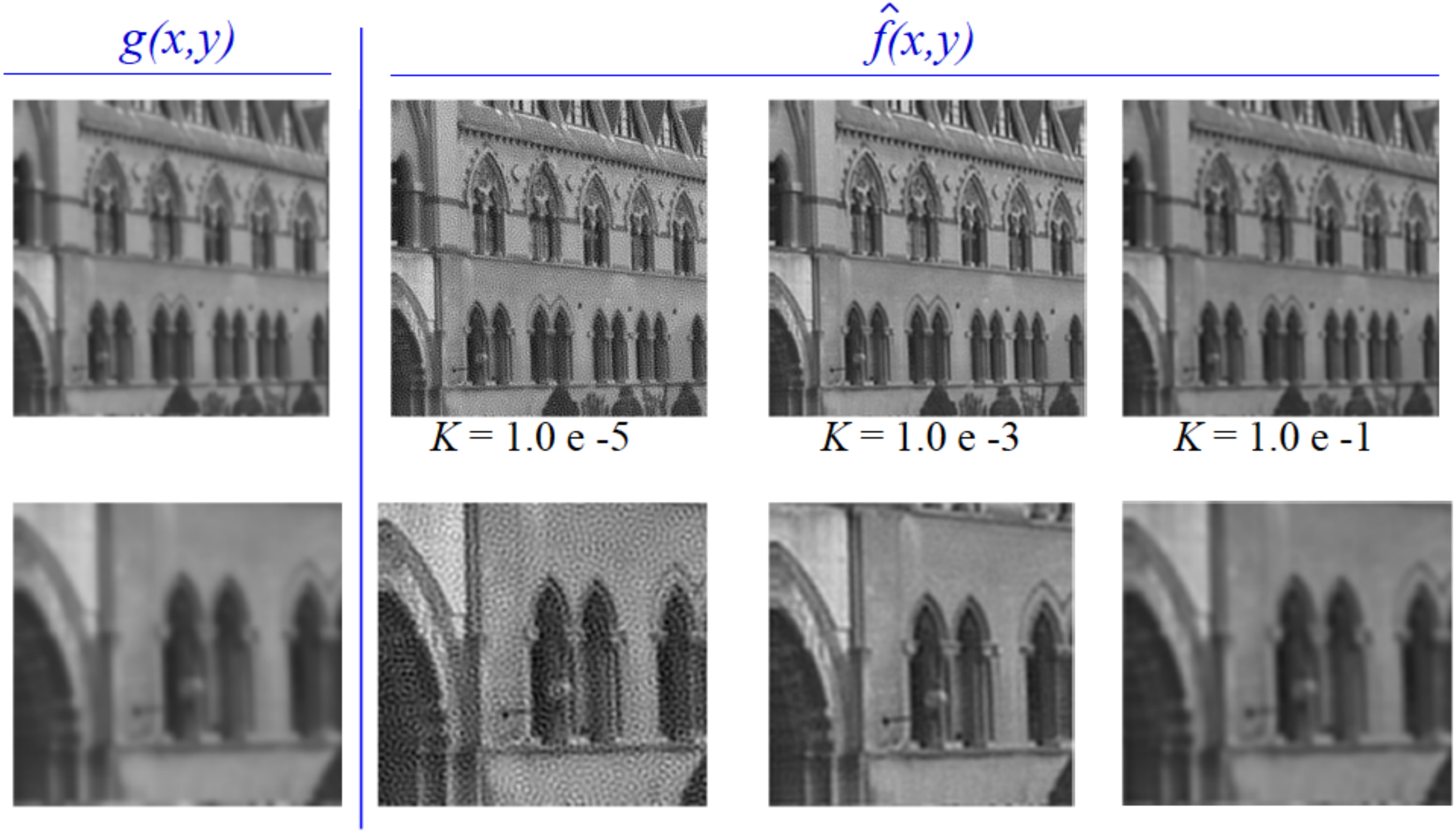
Wiener filter

$$\widehat{F}(u,v) = W(u,v)\, G(u,v)$$

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + K(u,v)}$$



Wiener filter



Wiener filter

$f(x,y)$

$g(x,y)$

$\hat{f}(x,y)$

$K = 5.0$ e -4

Wiener filter

Deblurred

High Noise | Medium Noise | Low Noise

Wiener filter

Comparison

Original

Wiener $K = 0.0001$

Inverse Butterworth $[90, 8]$

Wiener filter

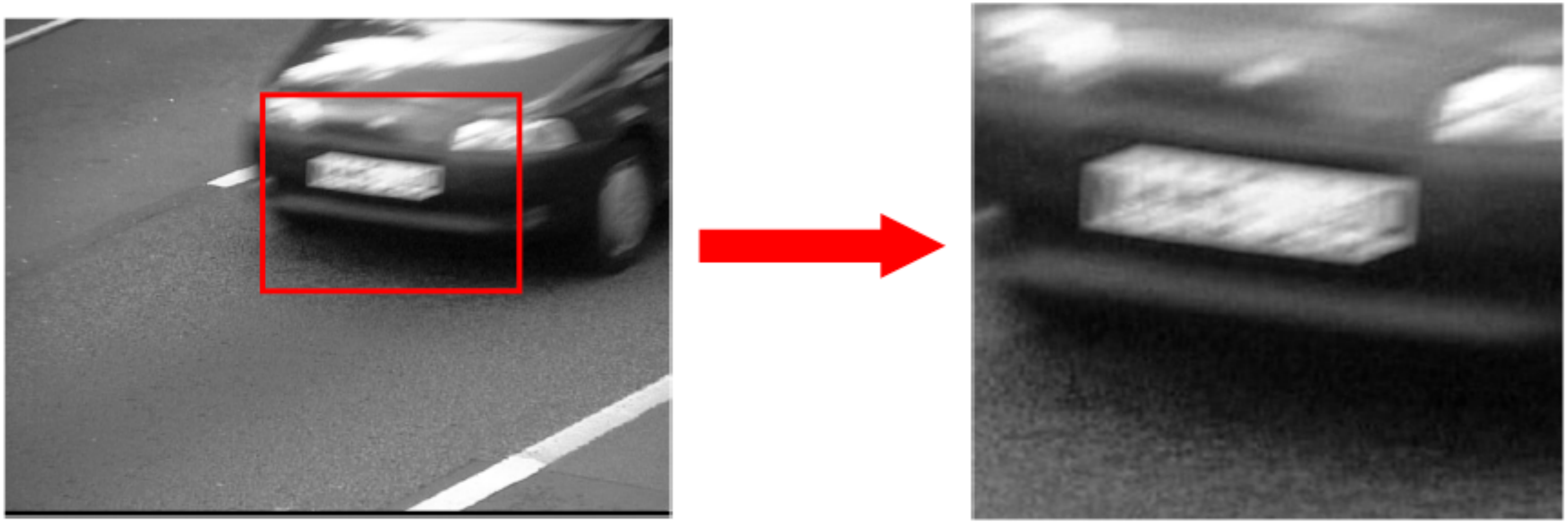
Application

Algorithm

1. Rotate image so that blur is horizontal
2. Estimate length of blur
3. Construct a bar modelling the convolution
4. Compute and apply a Wiener filter
5. Optimize over values of K

Wiener filter

Application



Zoom-in

Algorithm

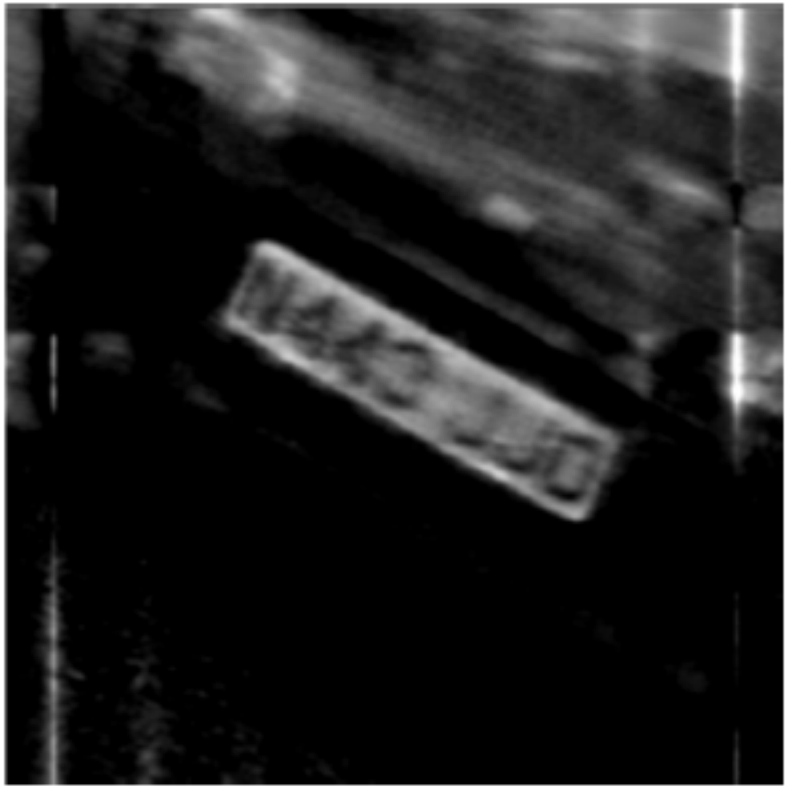
1. Rotate image so that blur is horizontal
2. Estimate length of blur
3. Construct a bar modelling the convolution
4. Compute and apply a Wiener filter
5. Optimize over values of K

Wiener filter

Application

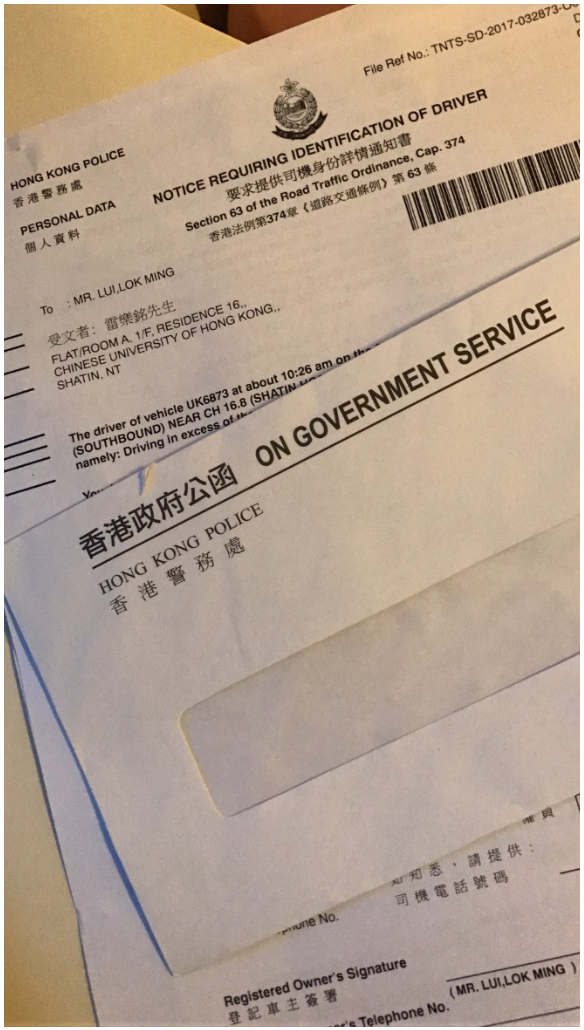
Blurry image

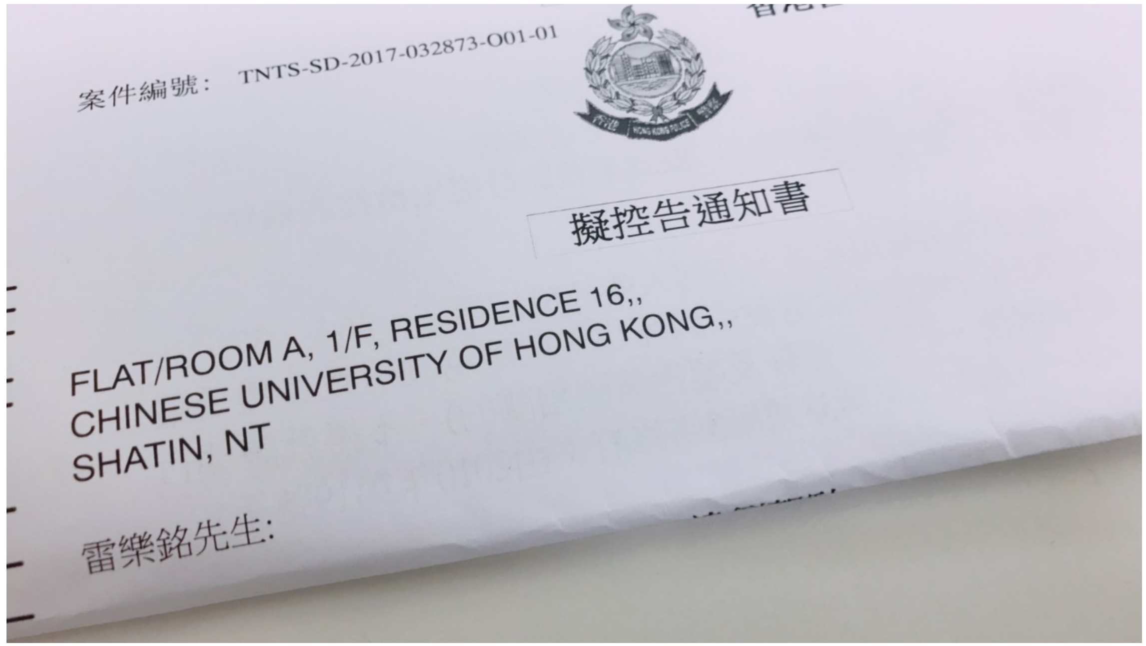
Deblurred image



Wiener filter

BUT sometimes it doesn't work!





Different image deblurring algorithms

Method 1: Direct inverse filtering

Let $T(u,v) = \dfrac{1}{H(u,v) + \varepsilon sgn(H(u,v))}$. Compute $\hat{F}(u,v) = G(u,v)T(u,v)$. Find inverse DFT of $\hat{F}(u,v)$ to get an image $\hat{f}(x,y)$.

(Here, $sgn(z) = 1$ if $Re(z) \geq 0$ and $sgn(z) = -1$ otherwise.)

Method 2: Modified inverse filtering

Let $B(u,v) = \dfrac{1}{1 + (\frac{u^2+v^2}{D^2})^n}$, and $T(u,v) = \dfrac{B(u,v)}{H(u,v) + \varepsilon sgn(H(u,v))}$, then

$$\hat{F}(u,v) = T(u,v)G(u,v) \approx F(u,v)B(u,v) + \frac{N(u,v)B(u,v)}{H(u,v) + \varepsilon sgn(H(u,v))}$$

$\dfrac{B(u,v)}{H(u,v) + \varepsilon sgn(H(u,v))}$ suppresses the high-frequency gain.

Different image deblurring algorithms

Method 3: Wiener Filter

The Wiener Filter is defined (in the frequency domain) as:

$$T(u,v) = \frac{\overline{H(u,v)}}{|H(u,v)|^2 + S_n(u,v)/S_f(u,v)}$$

where $S_n(u,v) = |N(u,v)|^2, S_f(u,v) = |F(u,v)|^2$ (Add parameters to avoid singularities)

If $S_n(u,v)$ and $S_f(u,v)$ are not known, then we let $K = S_n(u,v)/S_f(u,v)$ to get

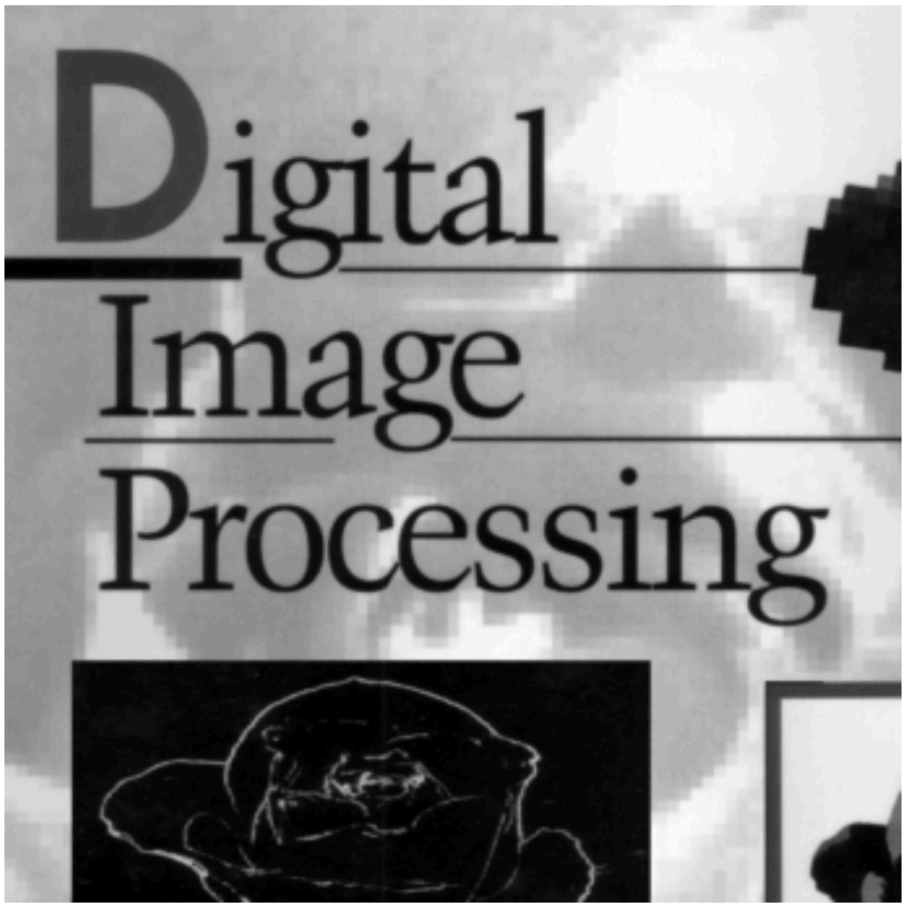
$$T(u,v) = \frac{\overline{H(u,v)}}{|H(u,v)|^2 + K}$$

Hence, Wiener Filter can be described as the inverse filtering as follows:

$$\hat{F}(u,v) = \left[\underbrace{\left(\frac{1}{H(u,v)} \right)}_{\text{direct inverse filter}} \underbrace{\left(\frac{|H(u,v)|^2}{|H(u,v)|^2 + K} \right)}_{\text{modifier}} \right] G(u,v)$$

Image deblur model

Original image



Blurred image

Wiener filter

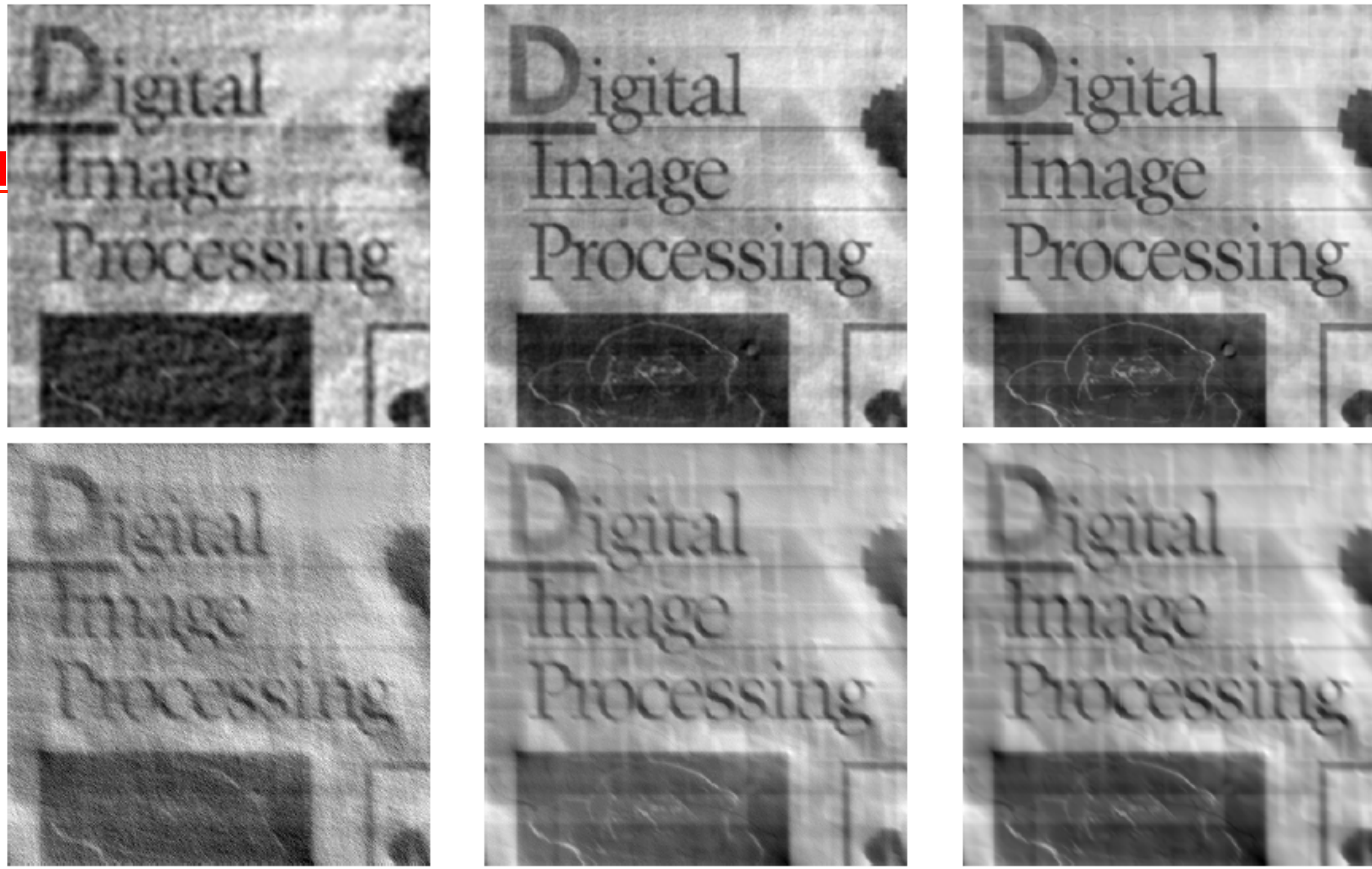
High Noise | Medium Noise | Low Noise

Constrained least square filtering

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma |P(u,v)|^2} \right] G(u,v)$$

Constrained Least Square

Wiener filter



High Noise | Medium Noise | Low Noise

Constrained least square filtering

Blurry image without noise

Constrained least square filtering

Blurry image without noise

Constrained least square filtering

Blurry images

Deblurred images