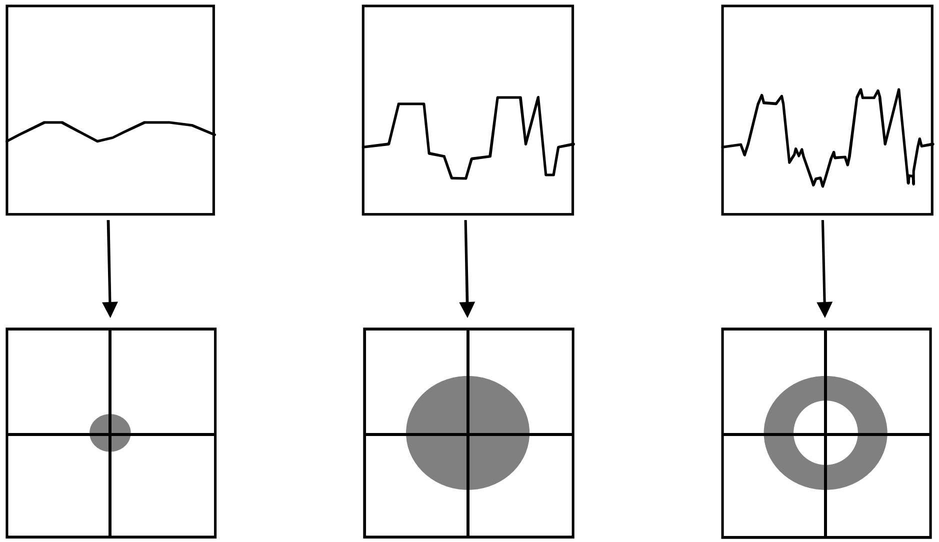
Image Processing in the Frequency Domain

- Filtering
- Restoration / Enhancement
- Inverse Filtering
- Tomography

Filtering in the Freq. Domain

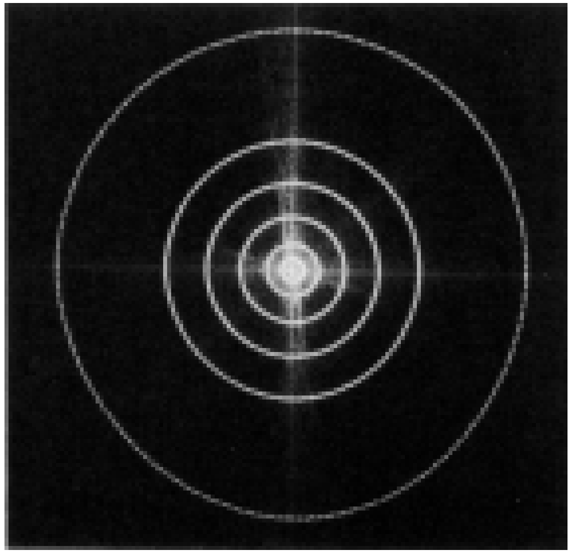
- Low Pass filters
- High Pass filters
- Band Pass filters
- Blurring
- Sharpening



Frequency Bands

Image

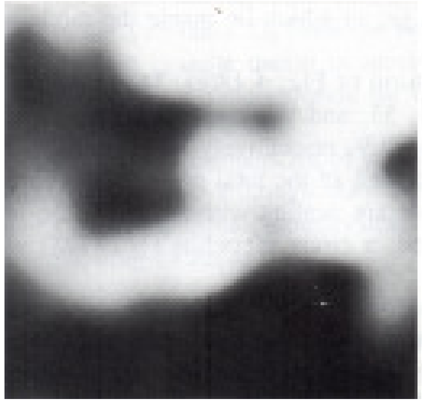
Fourier Spectrum



Percentage of image power enclosed in circles (small to large) :

90, 95, 98, 99, 99.5, 99.9

Blurring – Ideal Low Pass Filter

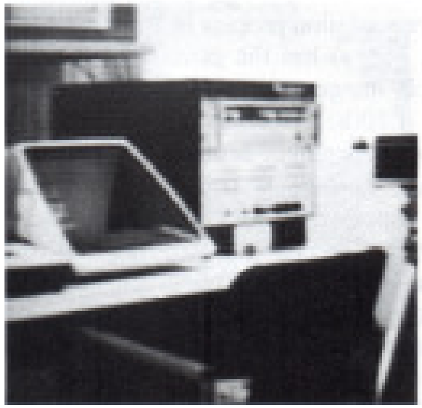


90%

95%

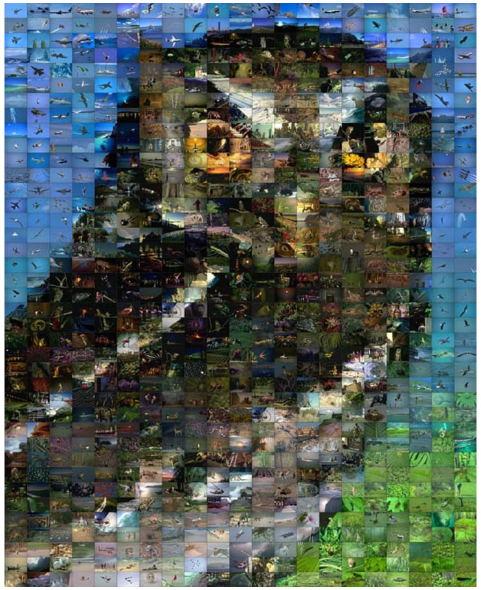
98%

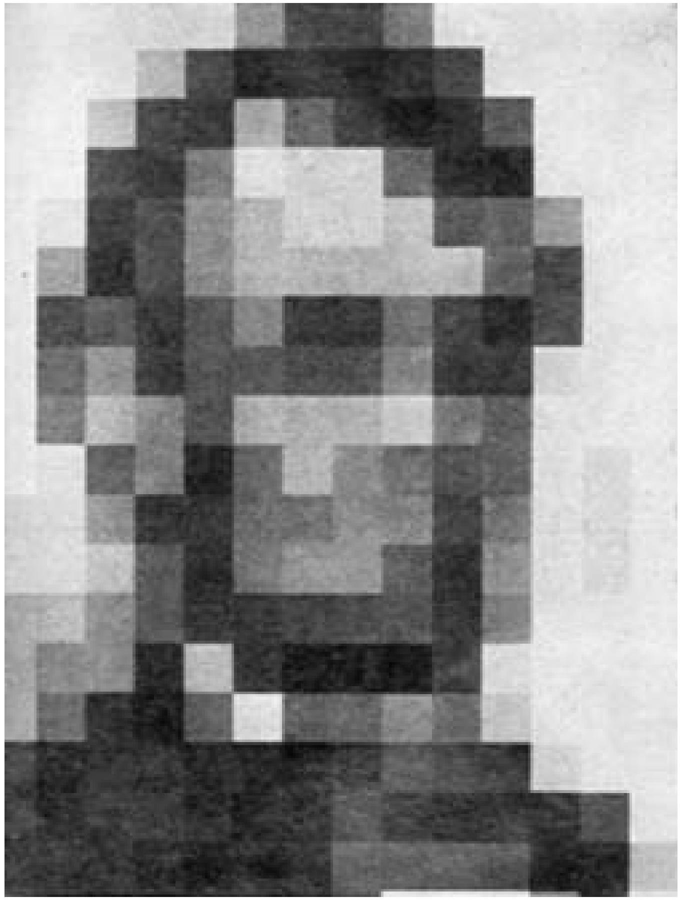
99%



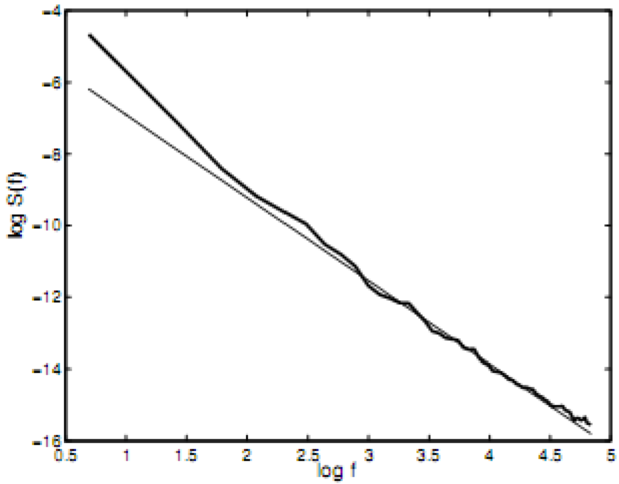
99.5%

99.9%





The Power Law of Natural Images



(b)

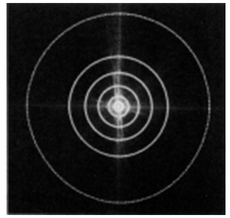


Figure 1: (a) A natural image (256 × 256 pixels), and (b) its circularly averaged power spectrum (thick line) and a linear fit to the high frequency portion (thin line). The slope in (b) is 2.3.

- The power in a disk of radii $r=\mathrm{sqrt}(u^2+v^2)$ follows: $P(r)=Ar^{-\alpha}$ where $\alpha \approx 2$

Recall: The Convolution Theorem

$g = f * h$	$g = f \cdot h$
implies	implies
$G = F \cdot H$	$G = F * H$

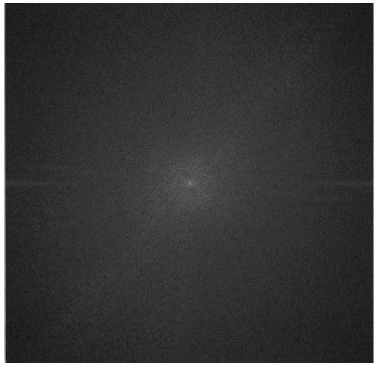
Low Pass Filter

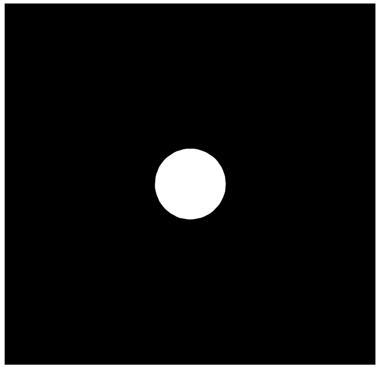
spatial domain

frequency domain

f(x,y)

F(u,v)





H(u,v)

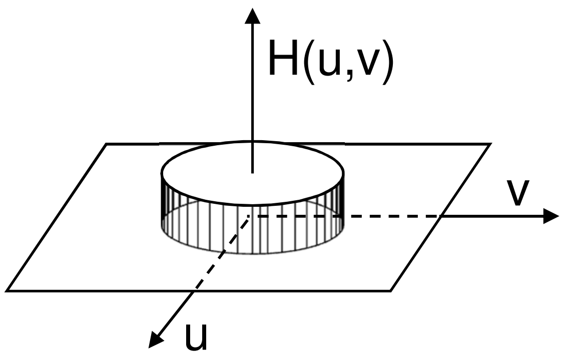
g(x,y)

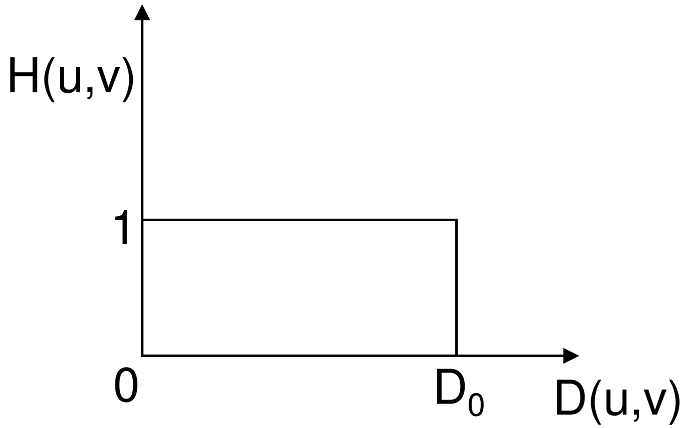
H(u,v) – Ideal Low Pass Filter

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

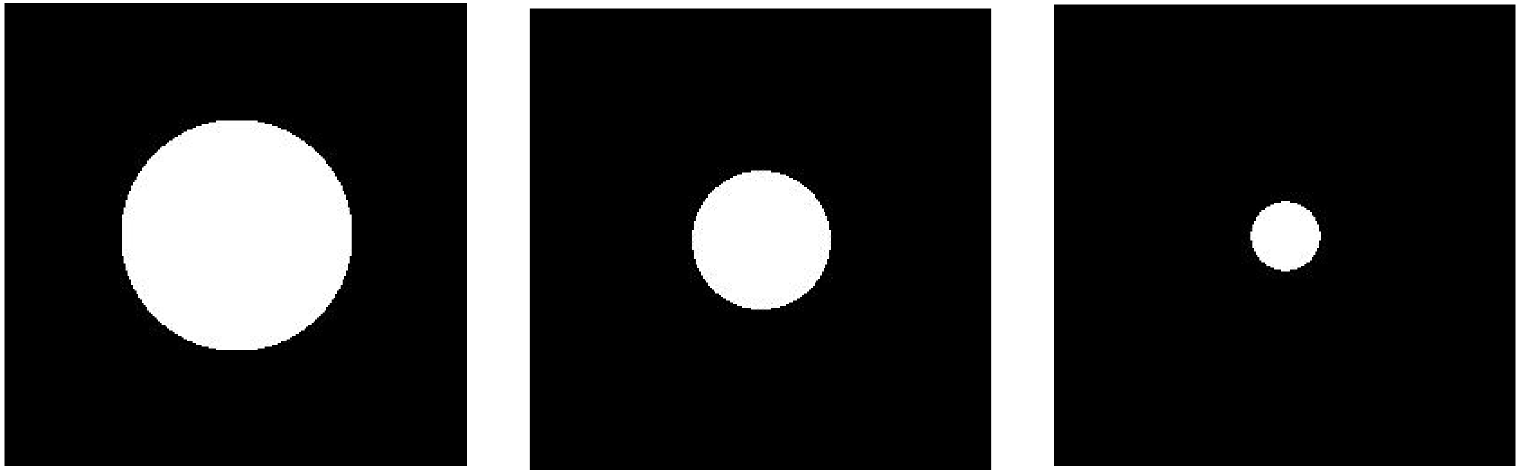
$$D(u,v) = \sqrt{u^2 + v^2}$$

D_0 = cut off frequency





Blurring – Ideal Low Pass



99.7% 99.37% 98.65%

Blurring – Ideal Low Pass

99.7%

99.6%

99.4%

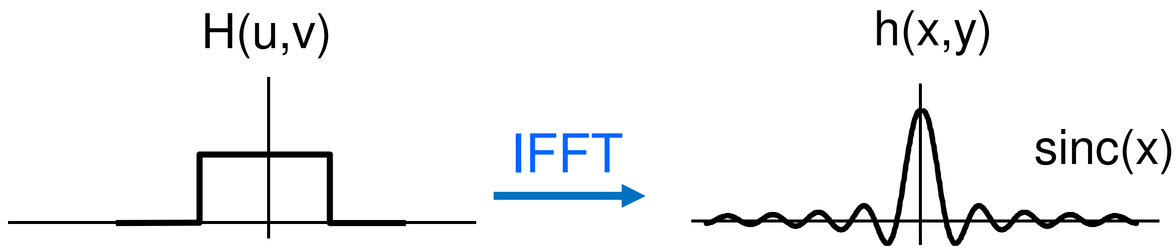
99.0%

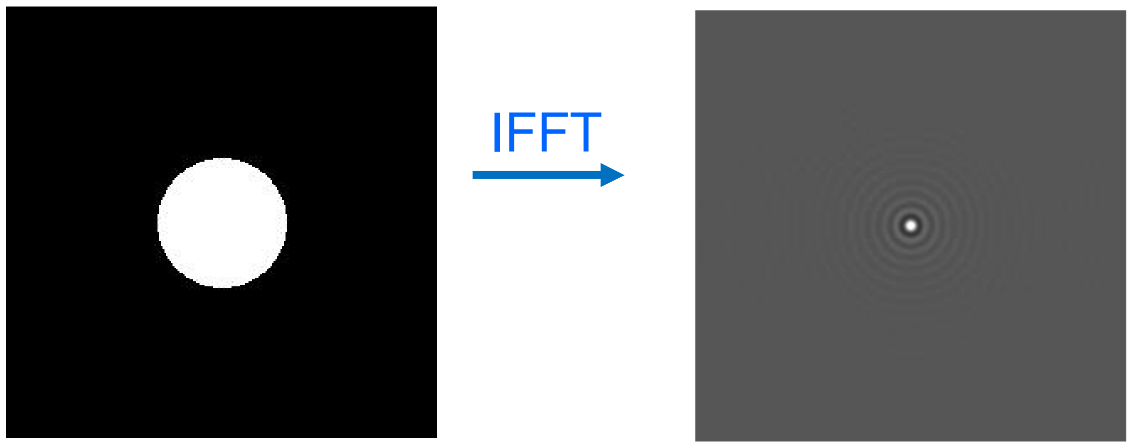
98.0%

96.6%

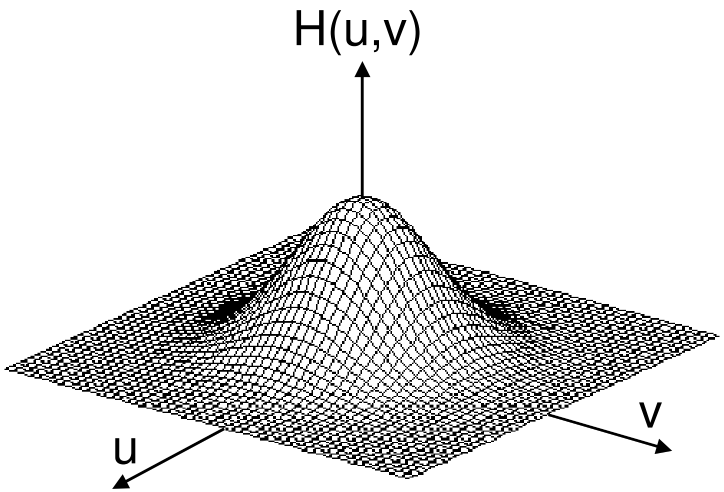
The Ringing Artifacts

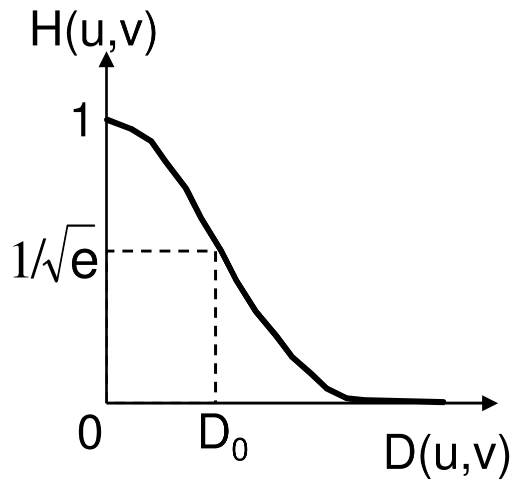
$$G(u,v) = F(u,v) \cdot H(u,v) \xrightarrow{\text{Convolution Theorem}} g(x,y) = f(x,y) * h(x,y)$$





Gaussian Blurring

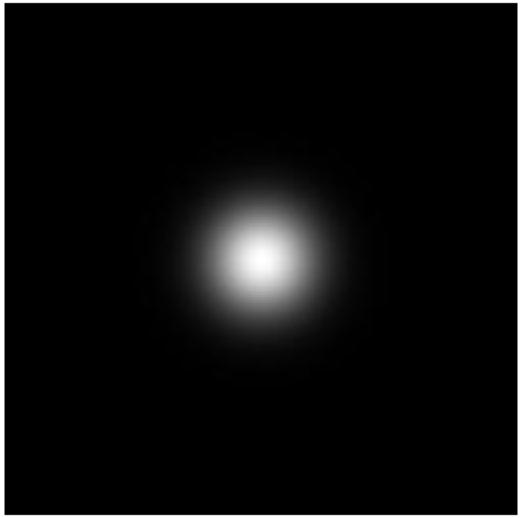




$$H(u,v) = e^{-D^2(u,v)/(2D^2{}_0)} \quad \text{where} \quad D(u,v) = \sqrt{u_2 + v_2}$$

Soften the Blurring + no Ringing

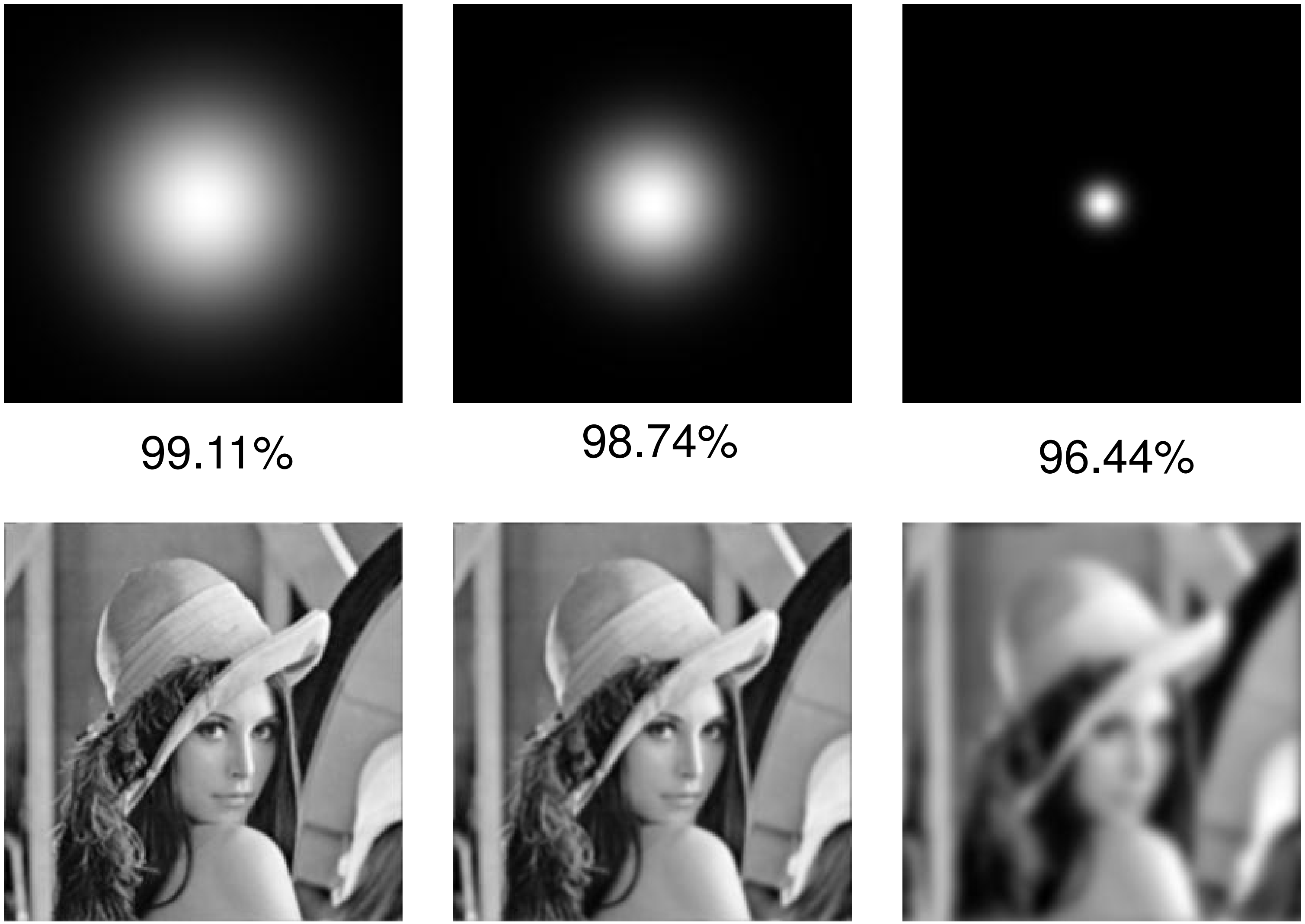
Gaussian Blurring



IFFT

Freq. domain

Spatial domain

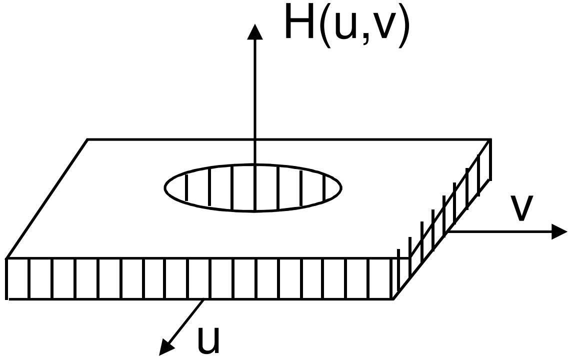


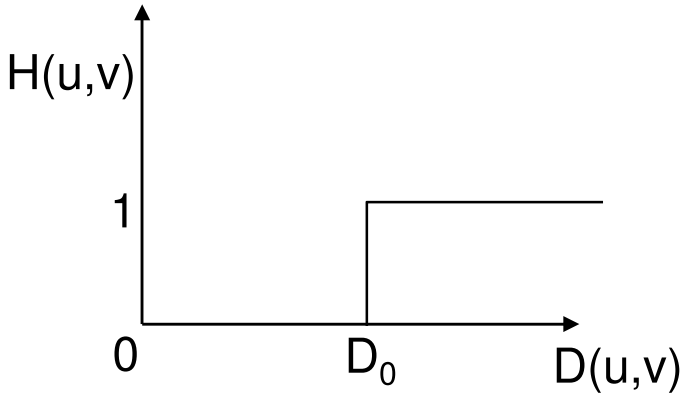
99.11%
98.74%
96.44%

High Pass – Ideal Filter

$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 \\ 1 & D(u,v) > D_0 \end{cases} \qquad \text{where} \qquad D(u,v) = \sqrt{u^2 + v^2}$$

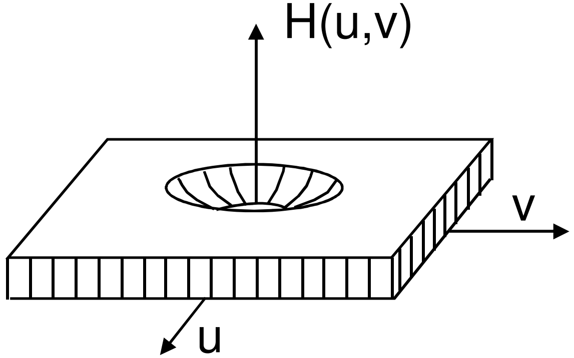
D_0 = cut off frequency

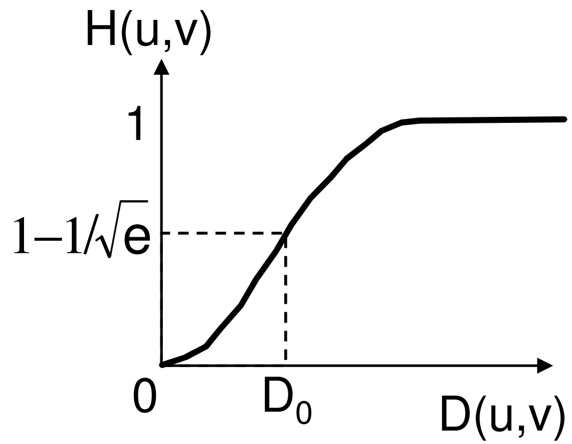




High Pass – Gaussian Filter

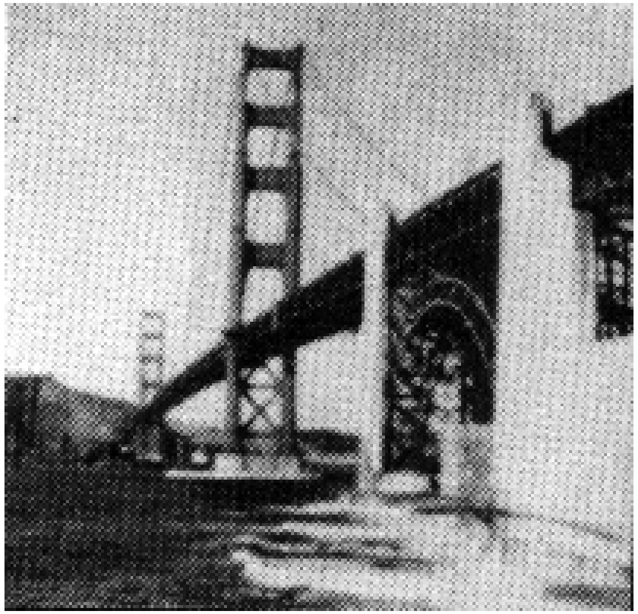
$$H(u,v) = 1 - e^{-D^2(u,v)/(2D^2_0)} \quad \text{where} \quad D(u,v) = \sqrt{u_2 + v_2}$$



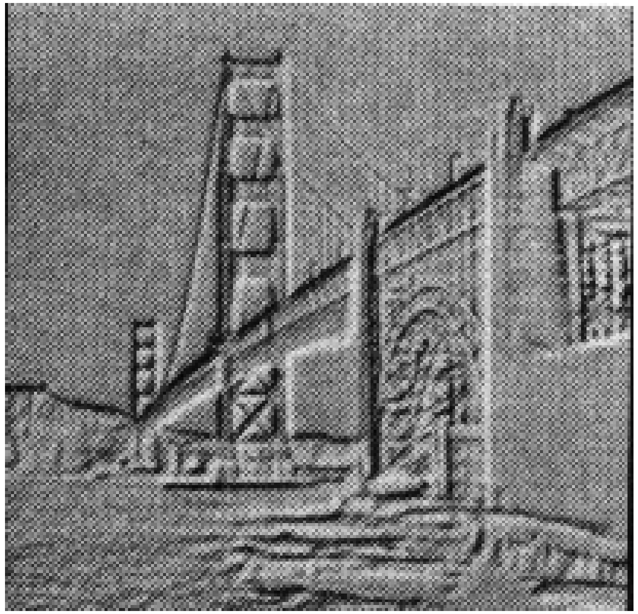


High Pass Filtering

Original

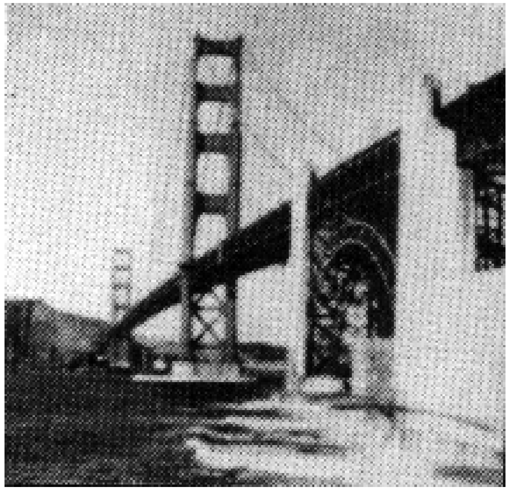


High Pass Filtered



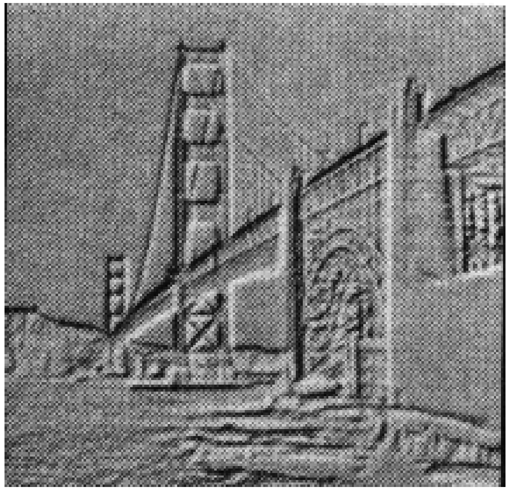
High Frequency Emphasis

Original



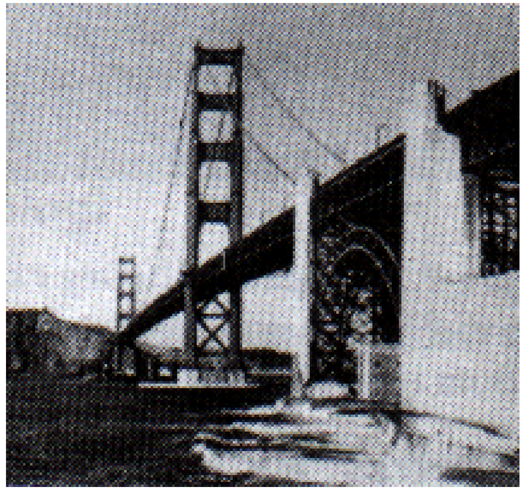
+

High Pass Filtered



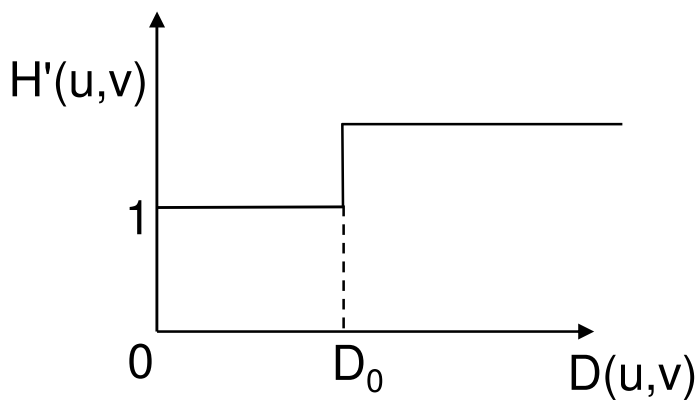
=

High Emphasize



High Frequency Emphasis

- Enhancing image f: $g = f + f * h$
- In the freq. domain: $G = F + F \cdot H = F \cdot (1+H)$



Emphasize high frequencies, maintain low frequencies and mean.

High Frequency - Examples

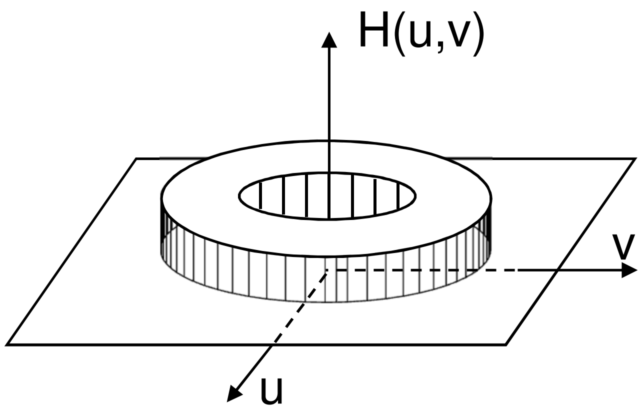
Original High Frequency Emphasis

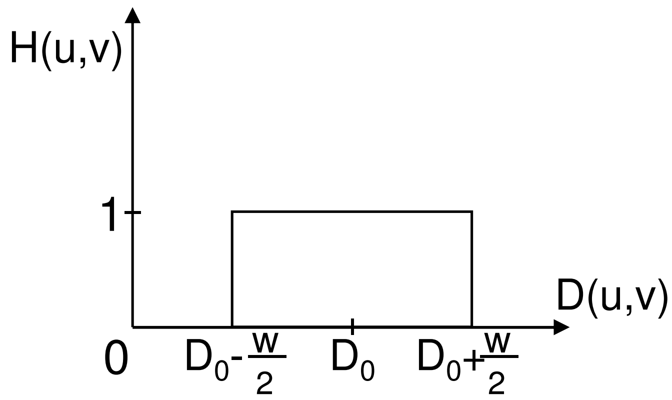
Band Pass Filter

$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 - \frac{w}{2} \\ 1 & D_0 - \frac{w}{2} \le D(u,v) \le D_0 + \frac{w}{2} \\ 0 & D(u,v) > D_0 + \frac{w}{2} \end{cases} \qquad \text{where} \quad D(u,v) = \sqrt{u^2 + v^2}$$

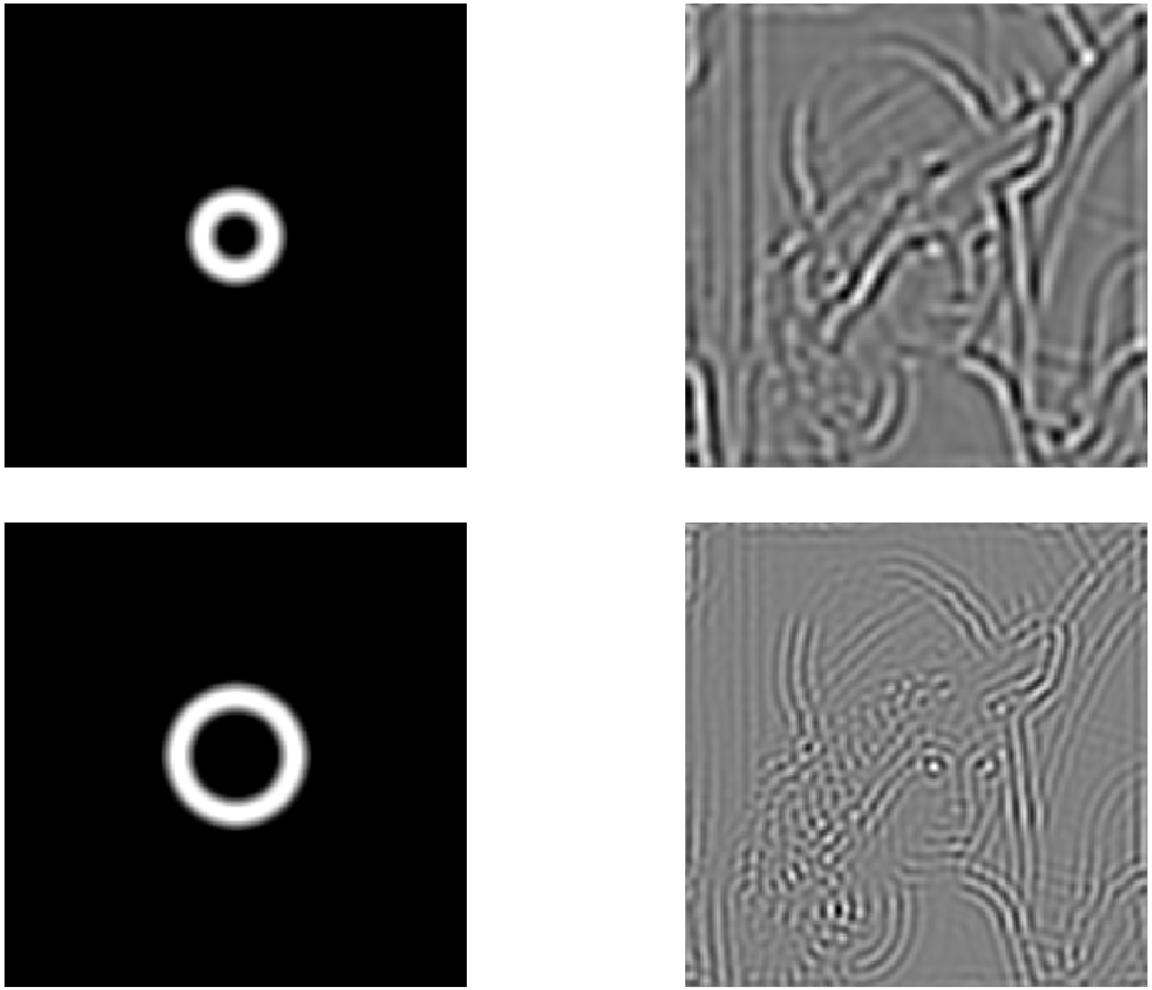
D_0 = cut off frequency

w = band width





Band Pass Filter



Local Frequency Filtering

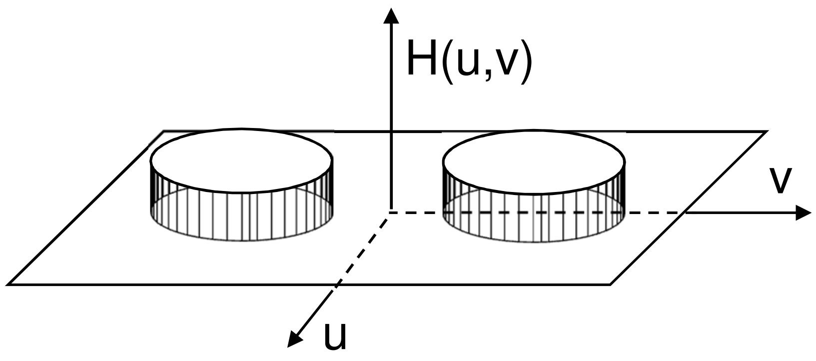
$$H(u,v) = \begin{cases} 1 & D_1(u,v) \le D_0 \ \textbf{ or } \ D_2(u,v) \le D_0 \\ 0 & \textbf{otherwise} \end{cases}$$

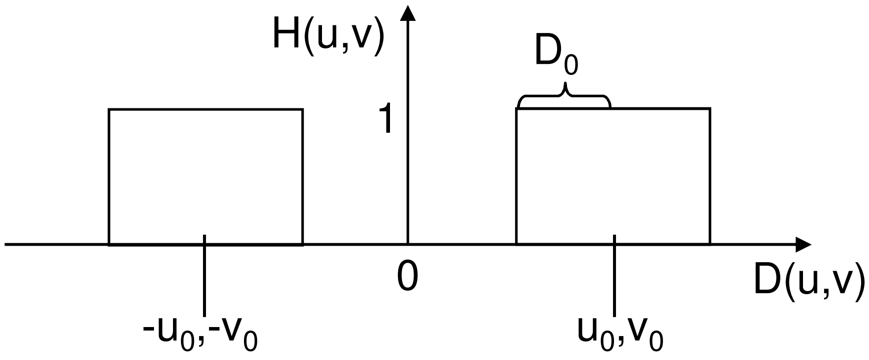
$$D_1(u,v) = \sqrt{(u-u_0)^2 + (v-v_0)^2}$$

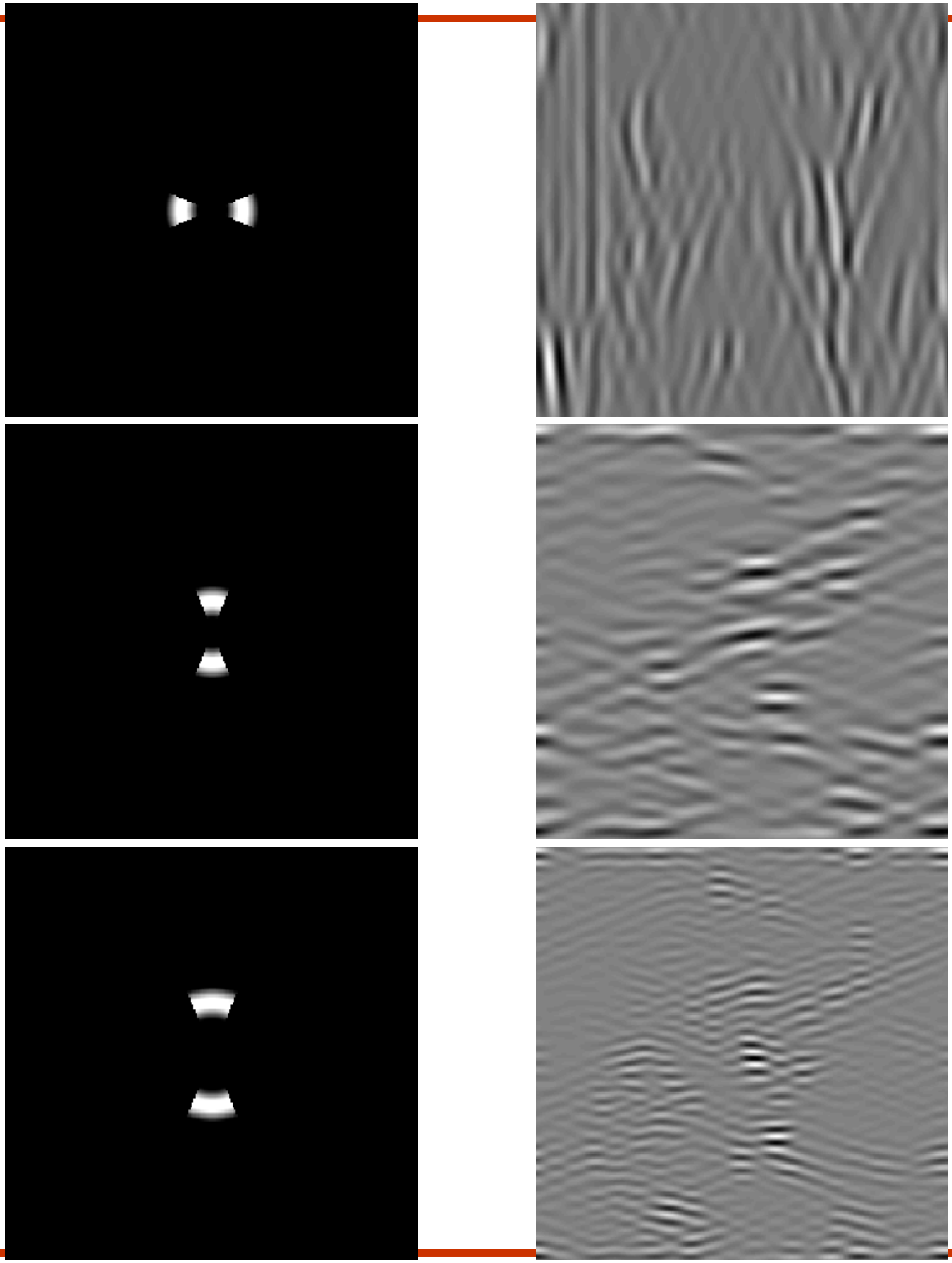
$$D_2(u,v) = \sqrt{(u+u_0)^2 + (v+v_0)^2}$$

D_0 = local frequency radius

u_0,v_0 = local frequency coordinates







Local Frequency Reject

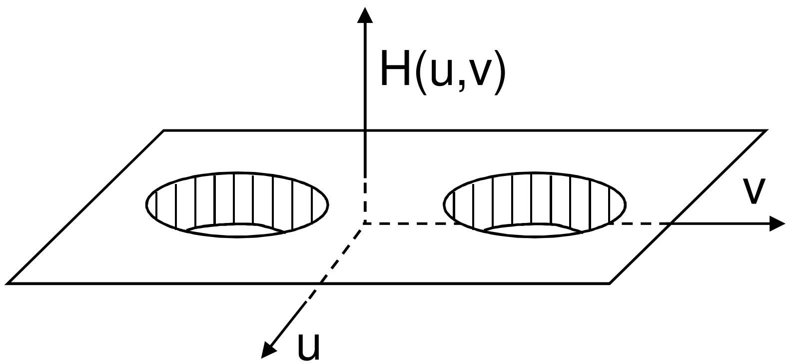
$$H(u,v) = \begin{cases} 0 & D_1(u,v) \le D_0 \ \textbf{ or } \ D_2(u,v) \le D_0 \\ 1 & \textbf{otherwise} \end{cases}$$

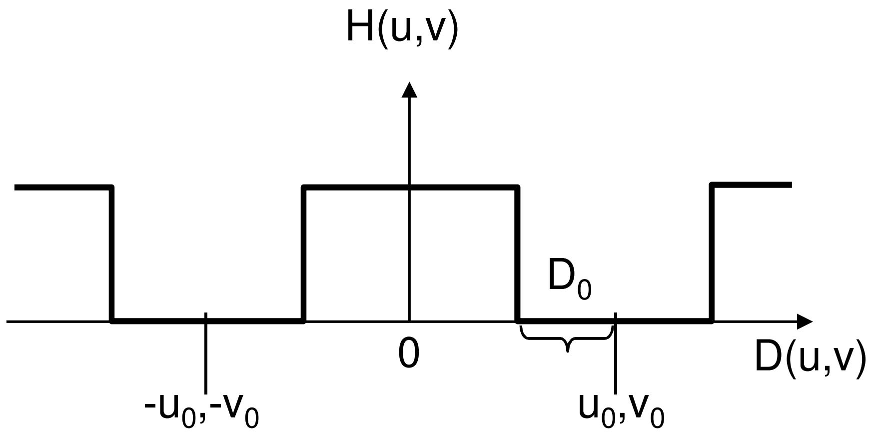
$$D_1(u,v) = \sqrt{(u-u_0)^2 + (v-v_0)^2}$$

$$D_2(u,v) = \sqrt{(u+u_0)^2 + (v+v_0)^2}$$

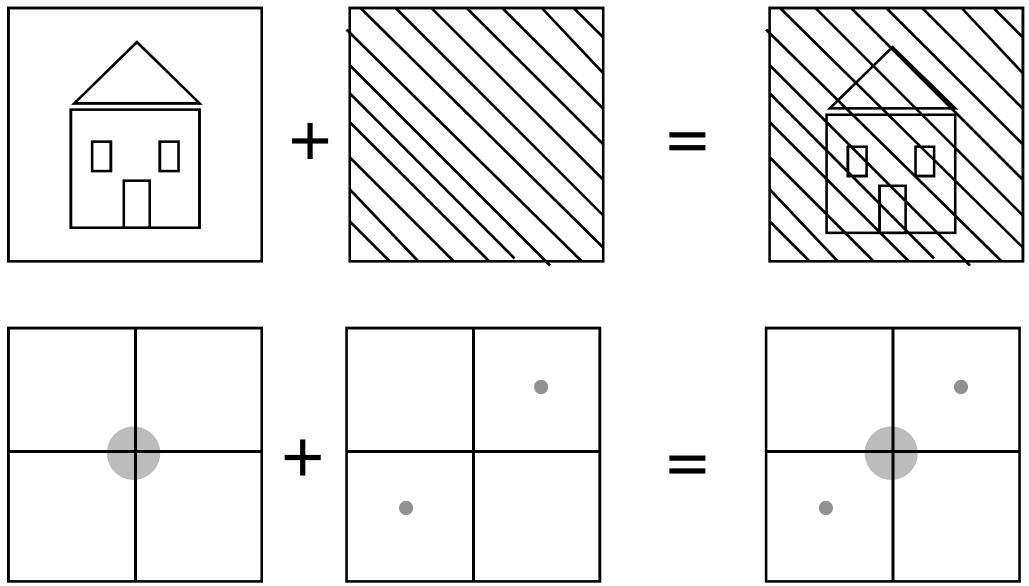
D_0 = local frequency radius

u_0,v_0 = local frequency coordinates



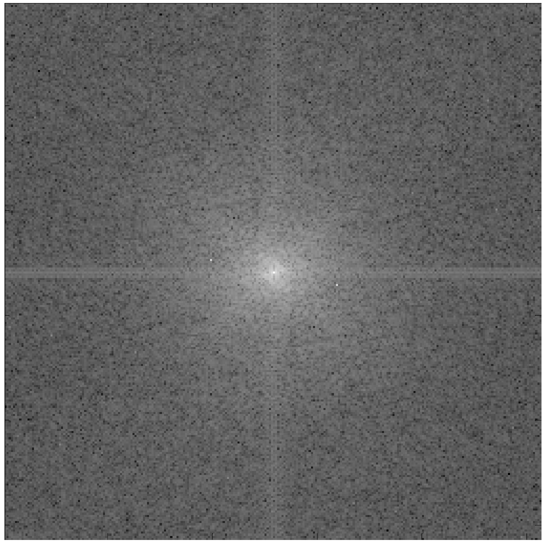


Local Frequency Reject for Noise Removal

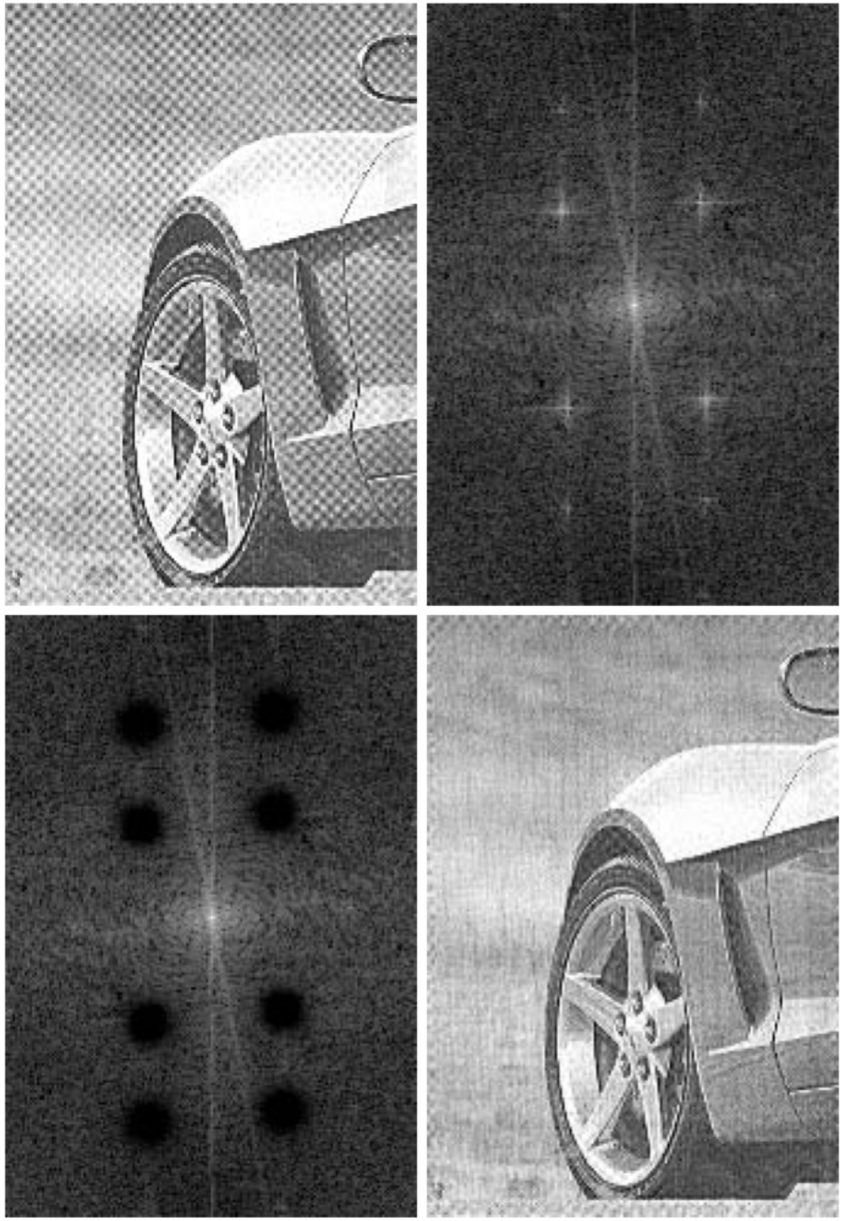


Original Noisy image

Fourier Spectrum



Local Reject Filter



a	b
c	d

FIGURE 4.64 (a) Sampled newspaper image showing a moiré pattern. (b) Spectrum. (c) Butterworth notch reject filter multiplied by the Fourier transform. (d) Filtered image.

Demo

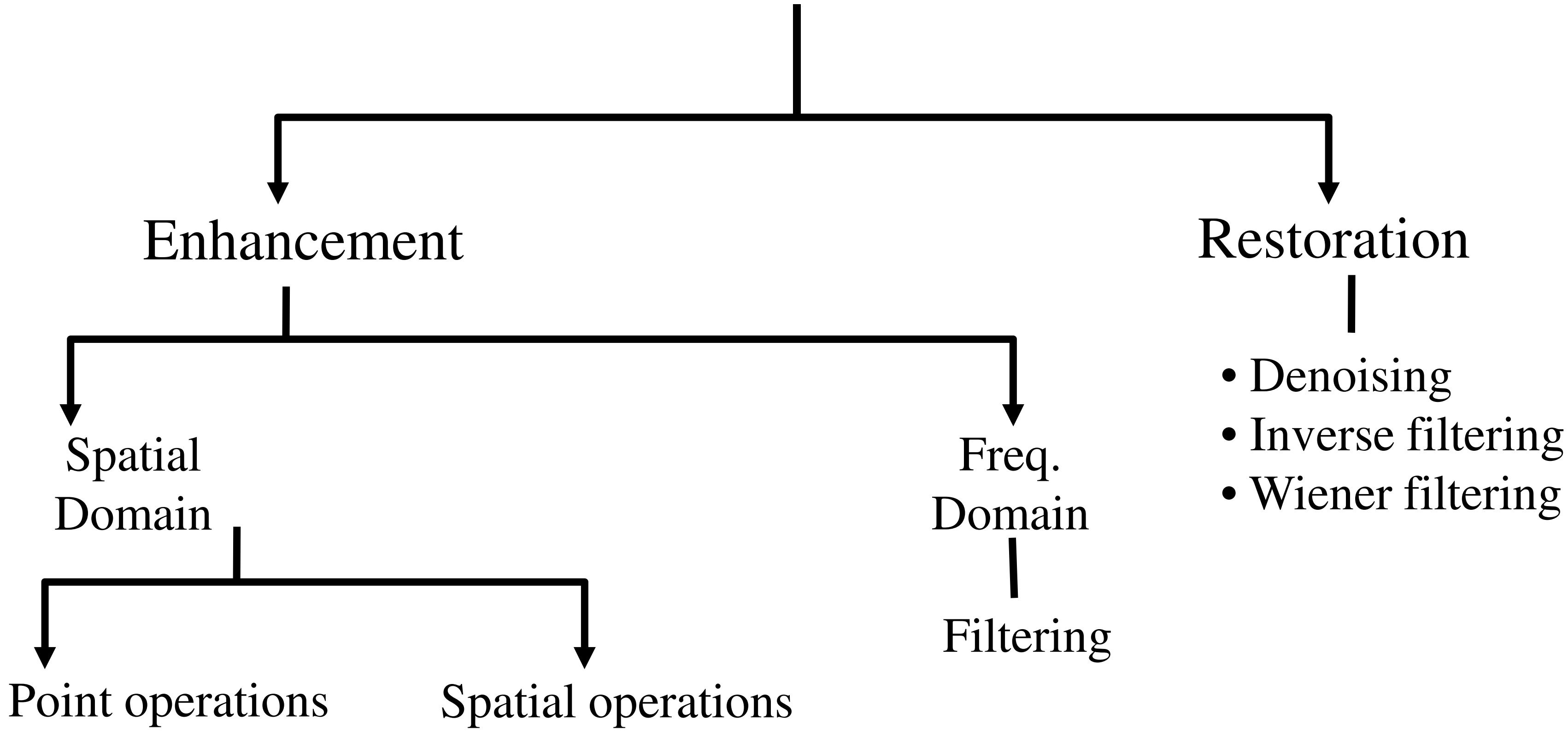
Enhancement and Restoration in the Freq. Domain

- Inverse Filtering + Prior
- Wiener Filter
- Image Tomography

Enhancement v.s. Restoration

- Image Enhancement:
  - A process which aims to improve bad images so they will “look” better.
- Image Restoration:
  - A process which aims to invert known degradation operations applied to images.

Image Preprocessing



Examples

Haze

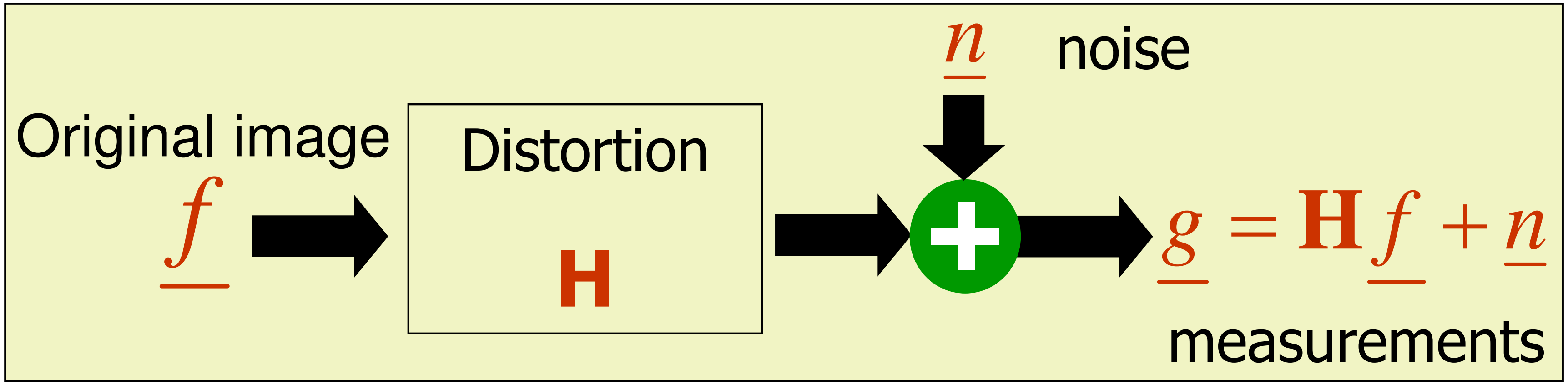
Echo image

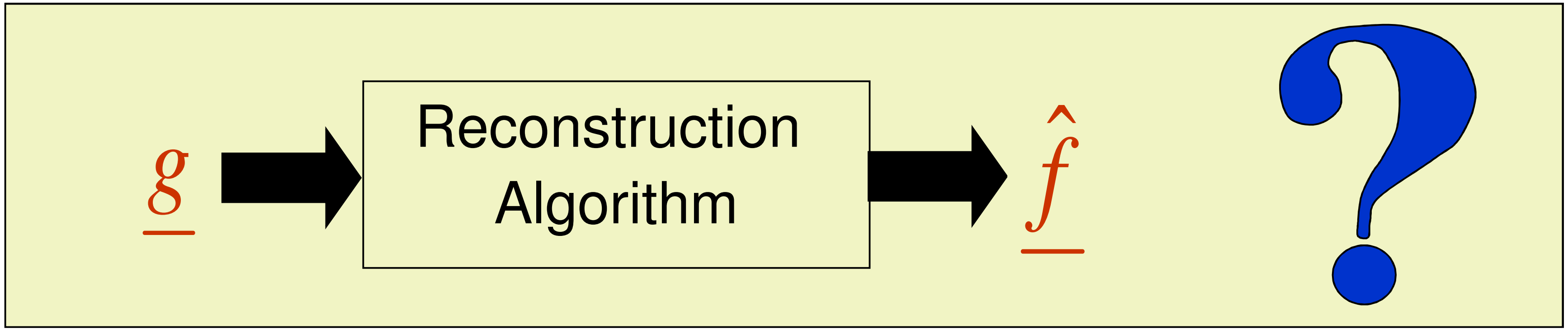
Motion Blur

Blurred image

Blurred image + additive white noise

Reconstruction as an Inverse Problem





Typically – H is a linear shift-invariant image operation $\rightarrow$ Convolution !!!

So what is the problem?

$$\underline{g} \Rightarrow \boxed{\mathrm{H}^{-1}\left(\underline{g}-\underline{n}\right)} \Rightarrow \underline{\hat{f}}$$

- Typically:
  - The distortion H is singular or ill-posed.
  - The noise n is unknown, only its statistical properties can be learnt.

Key point: Stat. Prior of Natural Images

MAP (maximum a-posteriori) estimation:

$$\underline{\hat{f}} = \arg\max_{f} P(f|g)$$

Following Bayes Rule: $P(f|g)P(g) = P(g|f)P(f)$

likelihood prior

$$\underline{\hat{f}} = \arg\max_{f} \underbrace{P(g|f)}_{\text{likelihood}}\underbrace{P(f)}_{\text{prior}}$$

Bayesian Reconstruction (MAP)

- From amongst all possible solutions, choose the one that maximizes the a-posteriori probability:

$$P(f \mid g) \propto P(g \mid f)\, P(f)$$

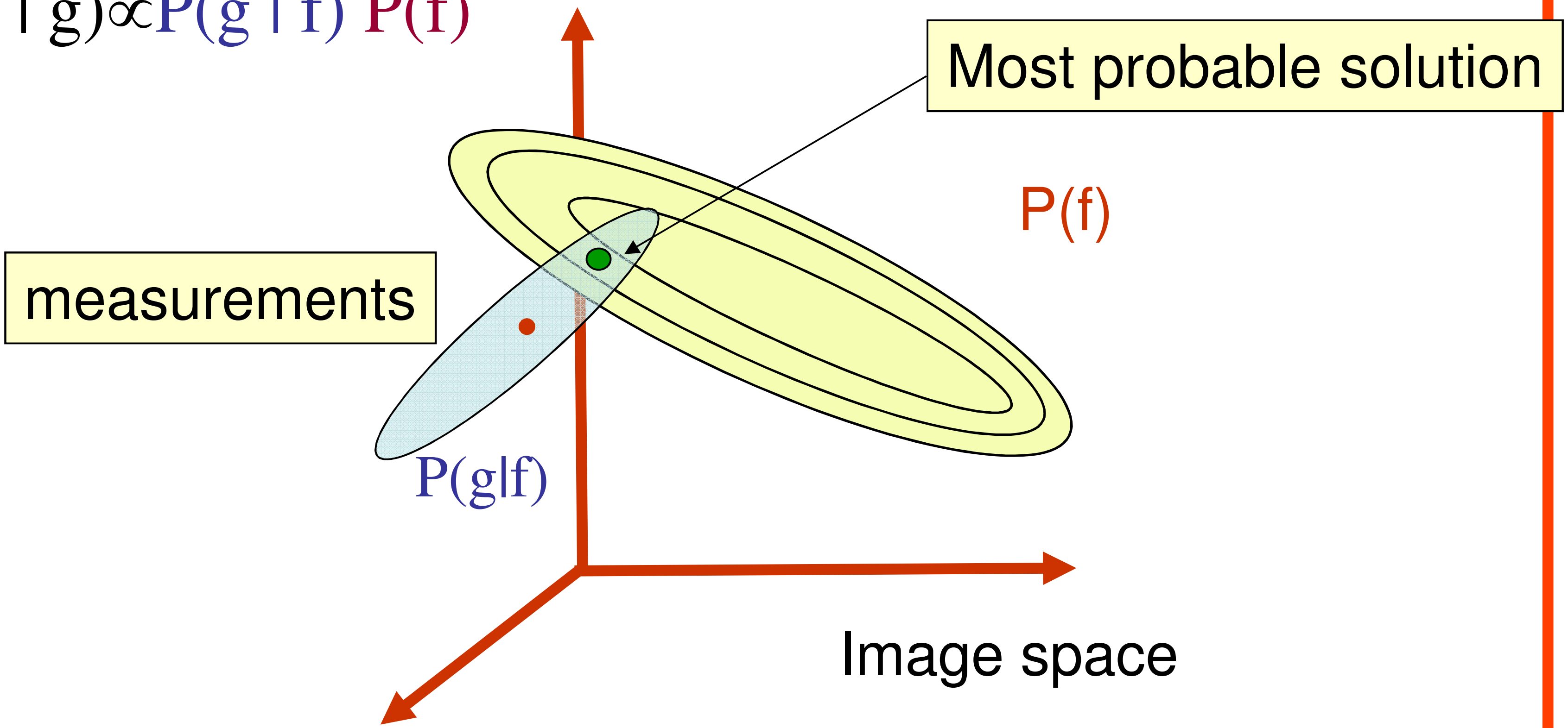


Image Denoising

Model: Additive noise

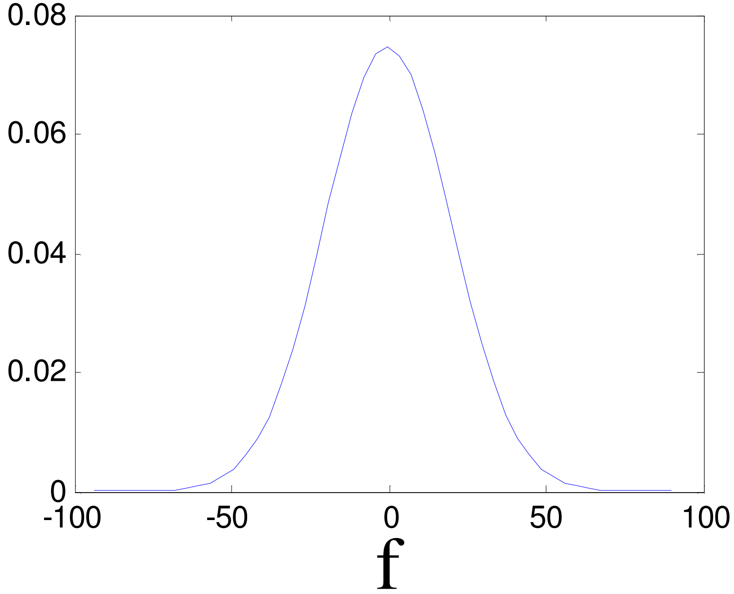
$$g = f + n$$

- The noise value is not known but its characteristics are known:
  - Parametric model
  - Parameters (mean, variance,...)

Examples of Independent and identically distributed (i.i.d) Noise

- Gaussian white noise (i.i.d.):

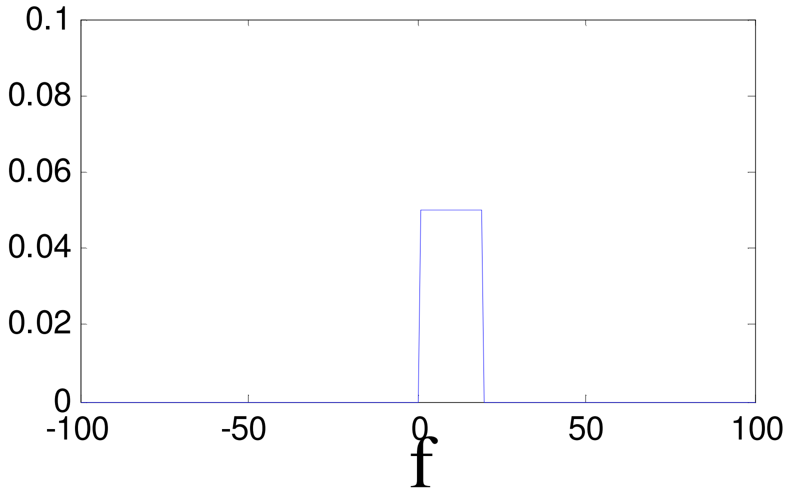
$$P(g \mid f) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(f-g)^2/2\sigma^2}$$



σ=20

- Uniform noise:

$$P(g \mid f) = \begin{cases} 1/(b-a) & a \le (g-f) \le b \\ 0 & otherwise \end{cases}$$

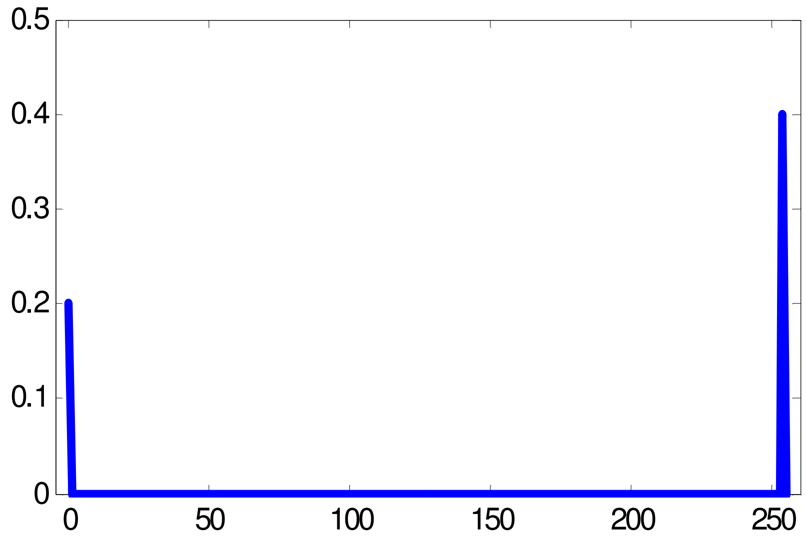


b-a=20

- Impulse noise (S & P):

$$P(g \mid f) = \begin{cases} P_a & \mathit{for} \;\; g = a \\ P_b & \mathit{for} \;\; g = b \\ 1 - P_a - P_b & \mathit{for} \;\; g = f \end{cases}$$

Note: this noise is not additive!



Bayesian Denoising

- Assume an additive noise model :

$$g = f + n$$

- A MAP estimate for the original f:

$$\hat{f} = \arg\max_{f} P(f \mid g)$$

- Using Bayes rule and taking the log likelihood :

$$\hat{f} = \arg\max_{f} \frac{P(g \mid f)P(f)}{P(g)} = \arg\max_{f} P(g \mid f)P(f)$$

Bayesian Denoising

If noise component is white Gaussian distributed:

g=f + n   where n is distributed ~N(0,σ)

$$P(g \mid f) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(g-f)^2/2\sigma^2}$$

$$\hat{f} = \arg\max_f P(g \mid f) P(f) = \arg\min_f \{-\log P(g \mid f) - \log P(f)\}$$

$$\hat{f} = \arg\min_f \{\underbrace{(g-f)^2}_{\text{data term}} + \underbrace{\lambda R(f)}_{\text{prior term}}\}$$

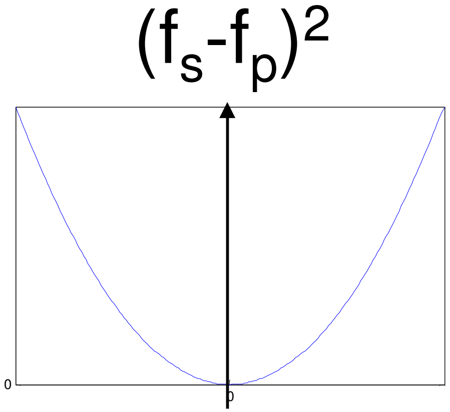
where R(f) is a penalty for non probable f

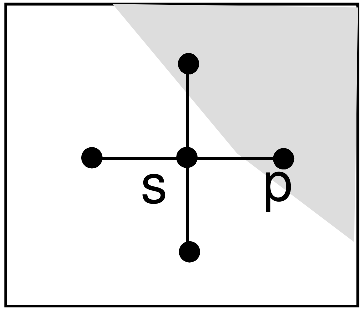
Example 1: Prior Term

- Similarity of neighboring pixels

$$R(f) = \sum_{s} \sum_{p \in N_s} W(s-p) \cdot (f_s - f_p)^2$$

– W(s-p) is a Gaussian profile giving less weight to distant pixels.





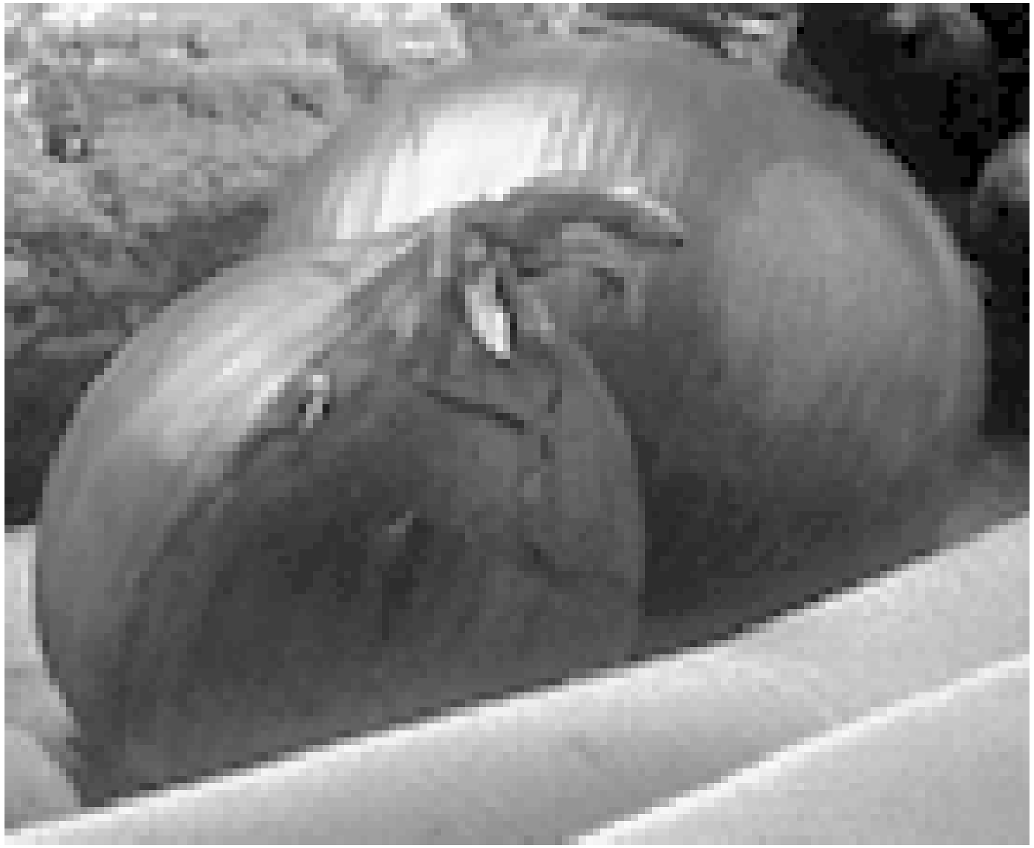
$$R(f_s) = \sum_{p \in N_s} W(s-p) \cdot (f_s - f_p)^2$$

- This leads to Gaussian smoothing:

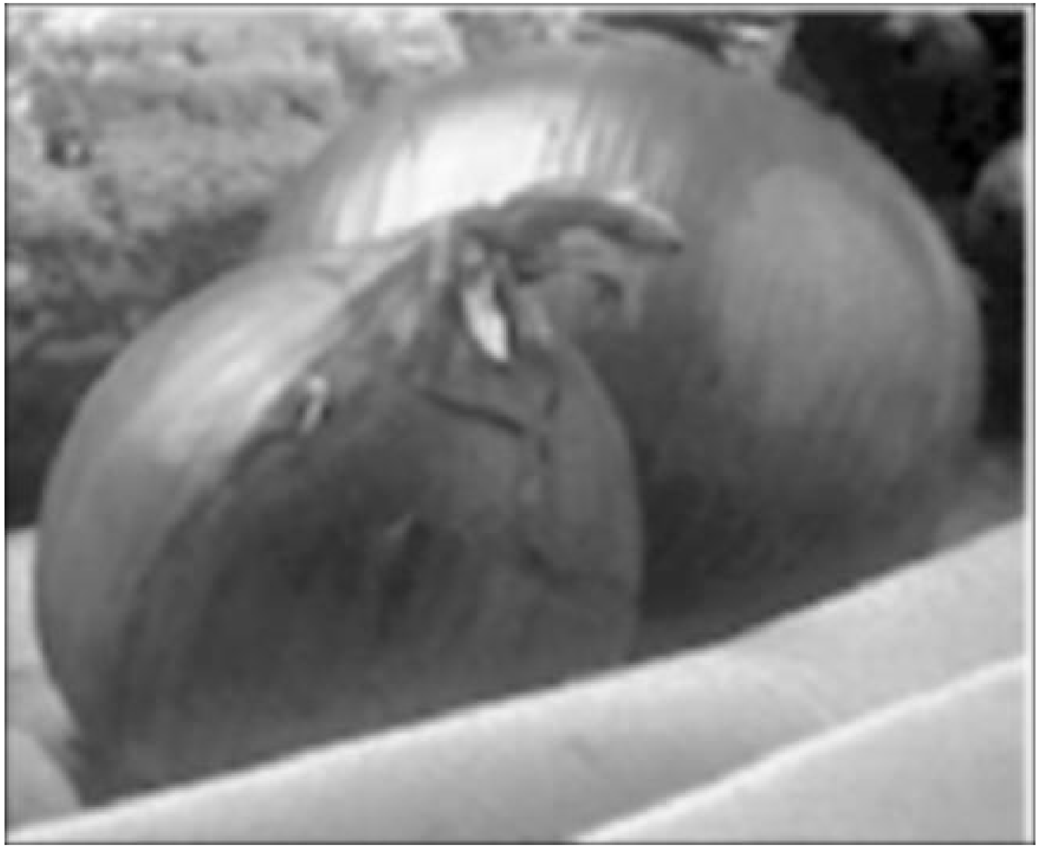
$$\hat{f}_s = (1-\alpha) g_s + \alpha \frac{\sum_{p \in N_p} W(s-p) g_p}{\sum_{p \in N_p} W(s-p)}$$

- Reduces noise but blurs out edges.
- The parameter α depends on the noise variance.

Noisy Image



Filtered Image

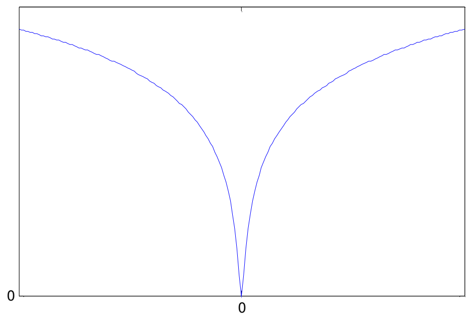


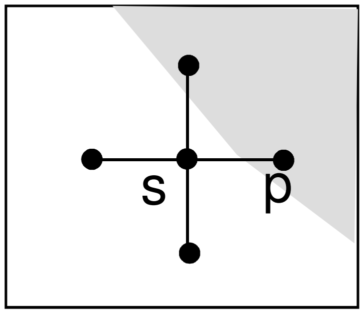
Example 2: Prior Term

- Edge sensitive similarity:

$$R(f) = \sum_{p \in N_s} W(s-p)\log\left(1+\left(f_s - f_p\right)^2\right)$$

$\log(1+(f_s-f_p)^2)$





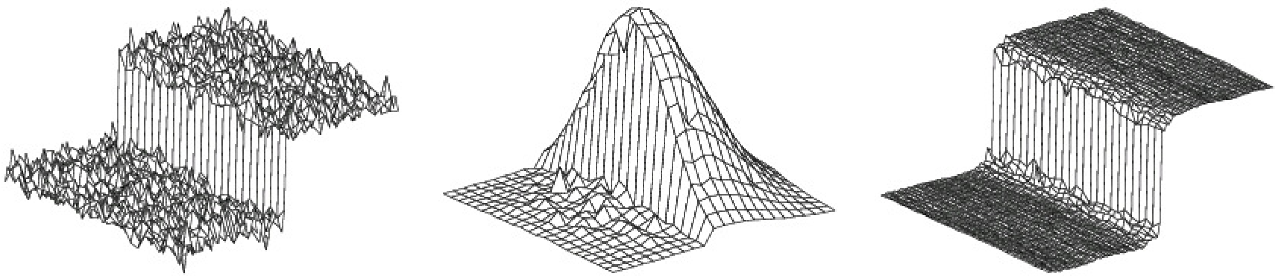
$$R(f) = \sum_{p \in N_s} W(s-p) \log\left(1 + \left(f_s - f_p\right)^2\right)$$

- This leads to edge-preserving smoothing:

$$\hat{f}_s = (1-\alpha) g_s + \alpha \frac{\sum_{p \in N_p} W_1(s-p) W_2\left(g_s - g_p\right) g_p}{\sum_{p \in N_p} W_1(s-p) W_2\left(g_s - g_p\right)}$$

  - W_1 is a monotonically descending spatial weight
  - W_2 is a monotonically descending photometric weight

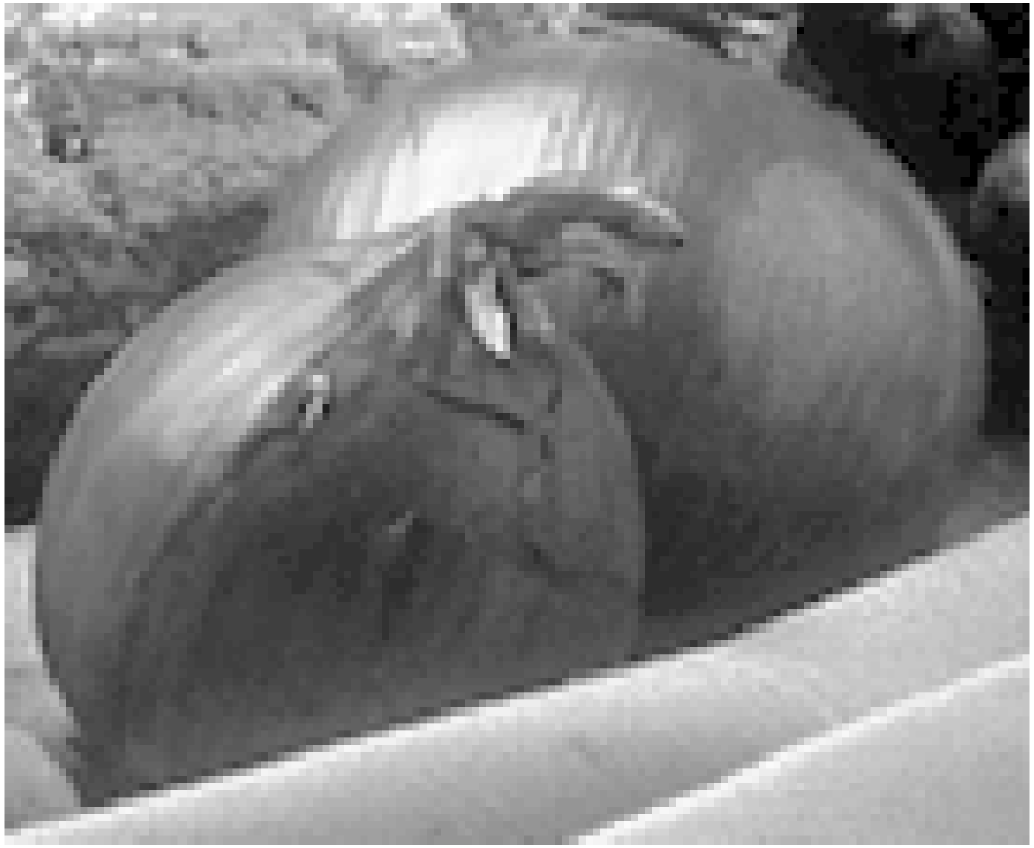
Bilateral Filtering



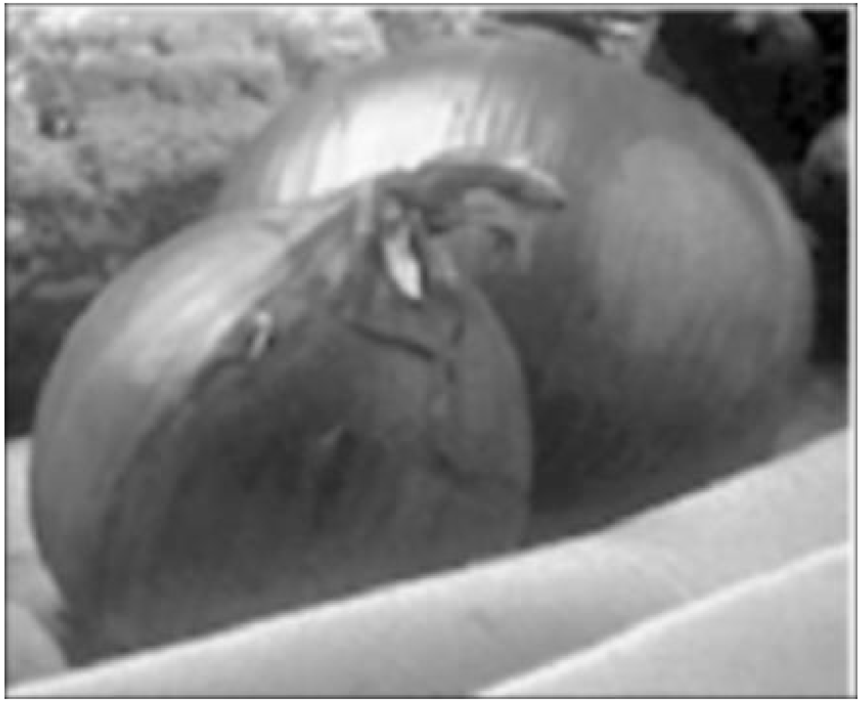
(a) (b) (c)

Left: noisy image **Middle**: weights **Right**: filtered image

Noisy Image



Filtered Image



Edge preserving smoothing

Gaussian smoothing

Inverse Filtering

$$g = Hf + n$$

Assume H is convolution with **known** mask h and no noise:

$$\boxed{g(x,y) = h(x,y) * f(x,y)}$$

$$\updownarrow$$

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

$$\Rightarrow \quad \hat{F}(u,v) = G(u,v) / H(u,v)$$

Inverse Filtering

$$g = Hf + n$$

Assume H is convolution with **known** mask h and **with** noise:

$$\boxed{g(x,y) = h(x,y) * f(x,y) + n(x,y)}$$

$$\updownarrow$$

$$G(u,v) = H(u,v) \cdot F(u,v) + N(u,v)$$

$$\Rightarrow \quad \hat{F}(u,v) = G(u,v)/H(u,v) + N(u,v)/H(u,v)$$

Inverse Filtering (Cont.)

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

Two problems with the above formulation:

1. H(u,v) might be zero for some (u,v).
2. In the presence of noise, the noise might be amplified:

Solution: Use prior information

$$\boxed{\hat{F} = \arg\min_F \underbrace{\left(HF - G\right)^2}_{\text{data term}} + \underbrace{\lambda R\left(F\right)}_{\text{prior term}}}$$

Option 1: Prior Term

- Use penalty term that restrains high F values:

$$\hat{F} = \arg\min_{F} E(F)$$

where $$E(F) = (HF - G)^2 + \lambda F^2$$

- Solution: $$\frac{\partial E(F)}{\partial F} = 2H^*(HF - G) + 2\lambda F = 0$$

$$\boxed{\hat{F} = \frac{H^*}{H^*H + \lambda} G}$$

$H(u,v) >> 1 \;\Rightarrow\; \hat{F} = G/H$

$H(u,v) << 1 \;\Rightarrow\; \hat{F} = 0$

Degraded Image (echo)

$$\hat{F} = \frac{H^*}{H^*H + \lambda} G$$

Degraded Image (echo+noise)

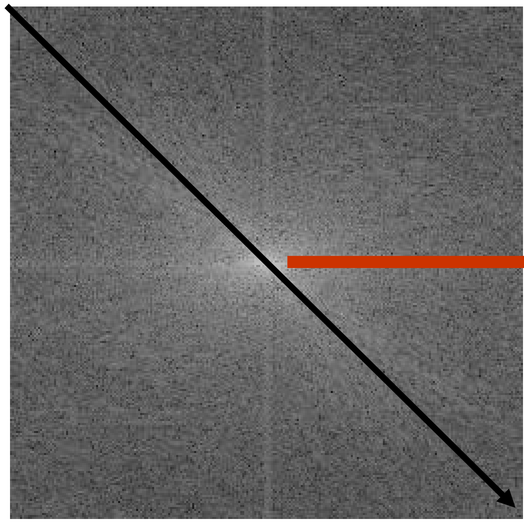
$$\hat{F} = \frac{H^*}{H^* H + \lambda} G$$

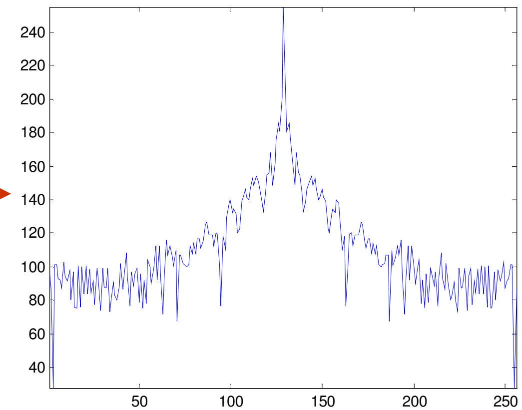
Option 2: Prior Term

1. Natural images tend to have low energy at high frequencies
2. White noise tend to have constant energy along freq.

where $$\hat{F} = \arg\min_F E(F)$$

$$E(F) = (HF - G)^2 + \lambda(u^2 + v^2)F^2$$





- Solution: $$\frac{\partial E(F)}{\partial F} = 2H^*(HF-G) + 2\lambda(u^2+v^2)F = 0$$

$$\boxed{\hat{F} = \frac{H^*}{H^*H + \lambda(u^2+v^2)} G}$$

- This solution is known as the *Wiener Filter*
- Here we assume N(u,v) is constant.
- If N(u,v) is not constant:

$$\hat{F} = \frac{H^*}{H^*H + \lambda(u^2+v^2) \cdot N(u,v)} G$$

Degraded Image (echo+noise)

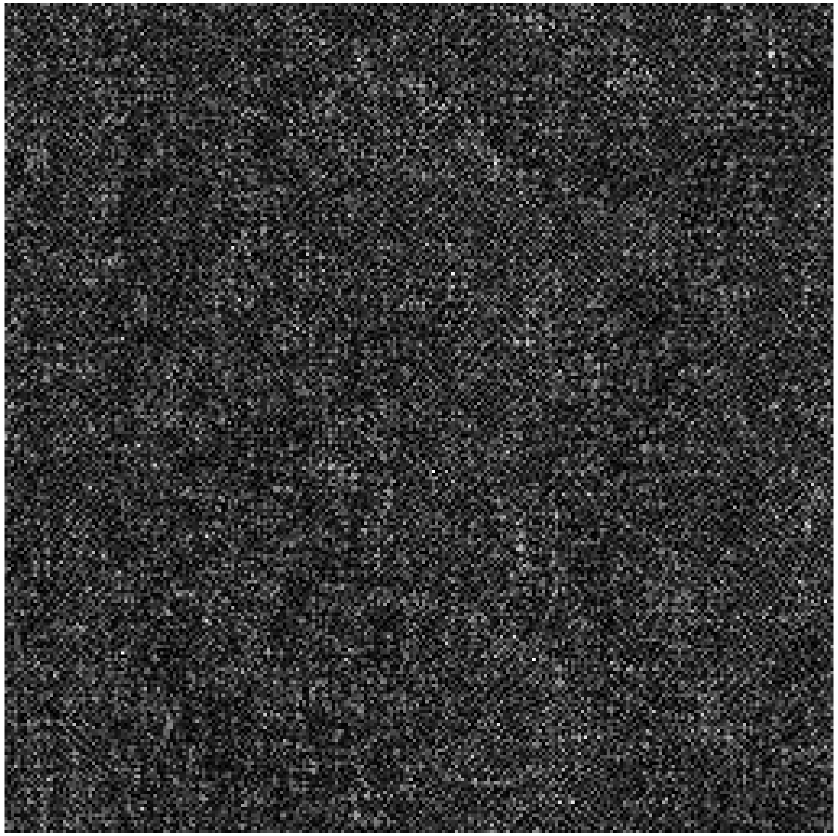
Wienner Filtering

Wienner

Previous

Degraded Image (blurred+noise)

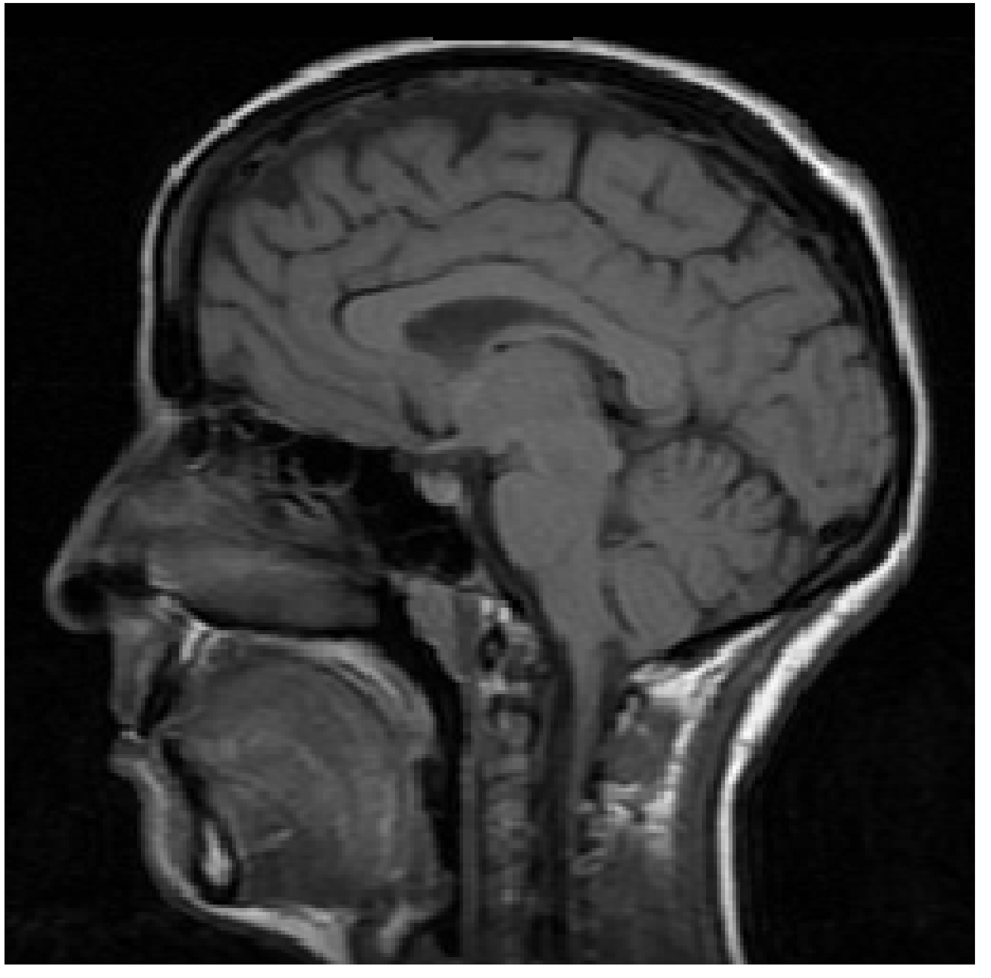
Inverse Filtering



Using Prior (Option 1)

Wienner Filtering

Computer Tomography using FFT



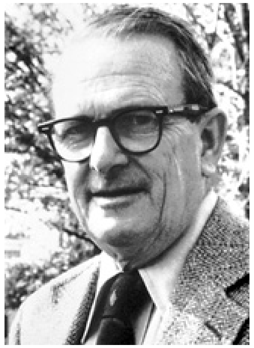
CT Scanners

- In 1901 Wilhelm Conrad Roentgen won the Nobel Prize (1st in physics) for his discovery of X-rays.

Wilhelm Conrad Röntgen

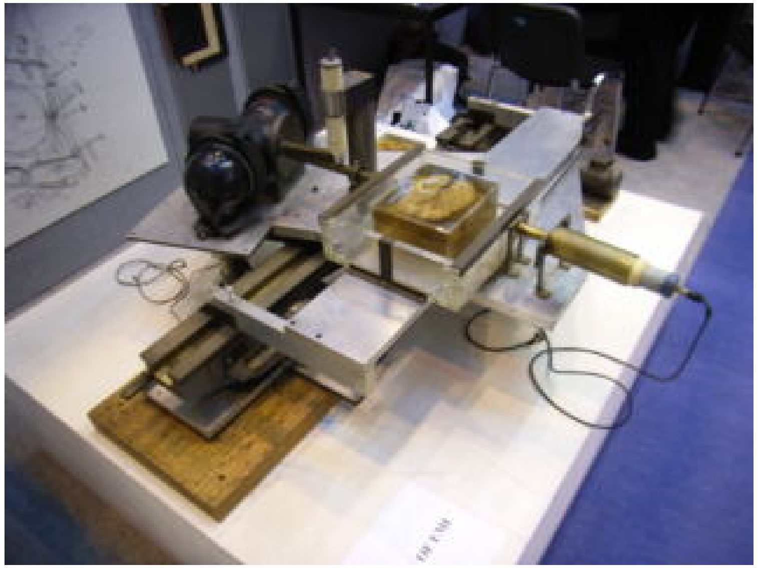
CT Scanners

- In 1979 G. Hounsfield & A. Cormack, won the Nobel Prize for developing the computer tomography.
- The invention revolutionized medical imaging.

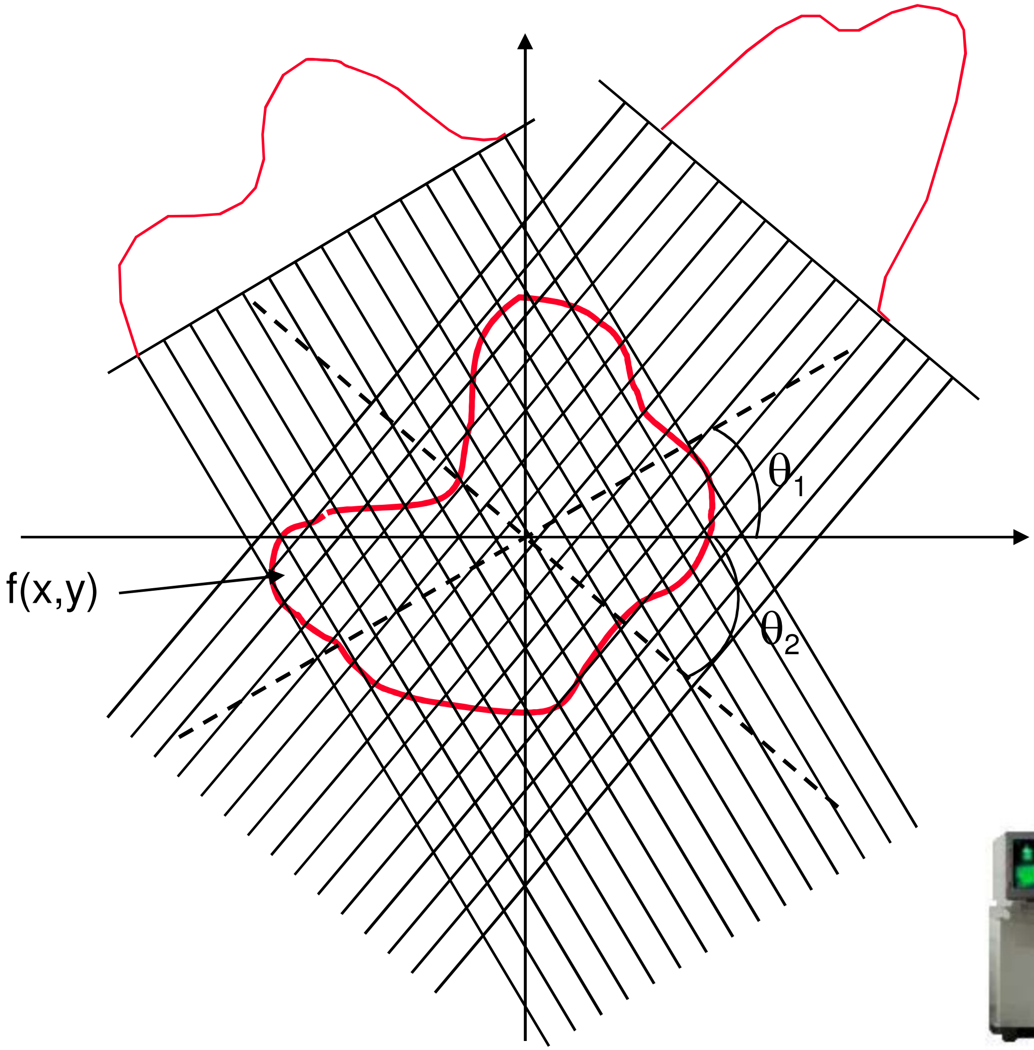


Allan M. Cormack

Godfrey N. Hounsfield

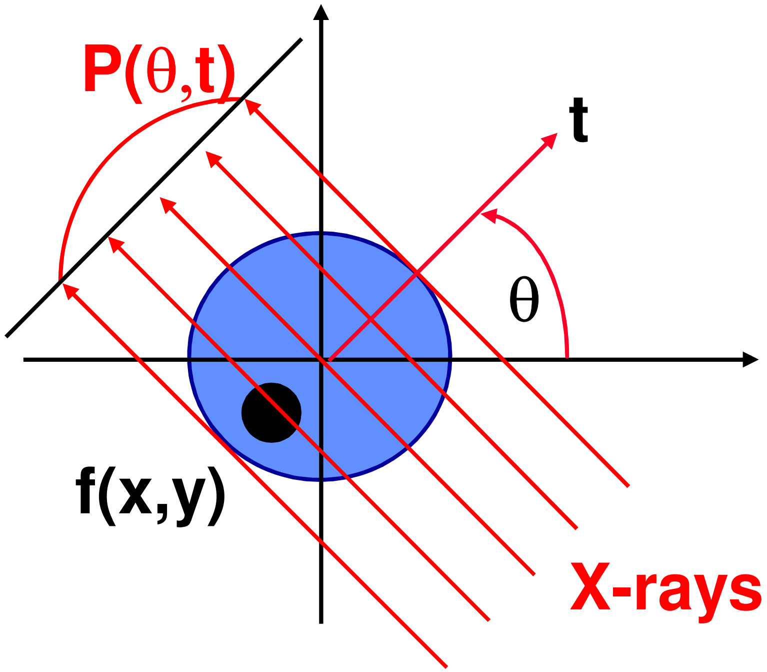


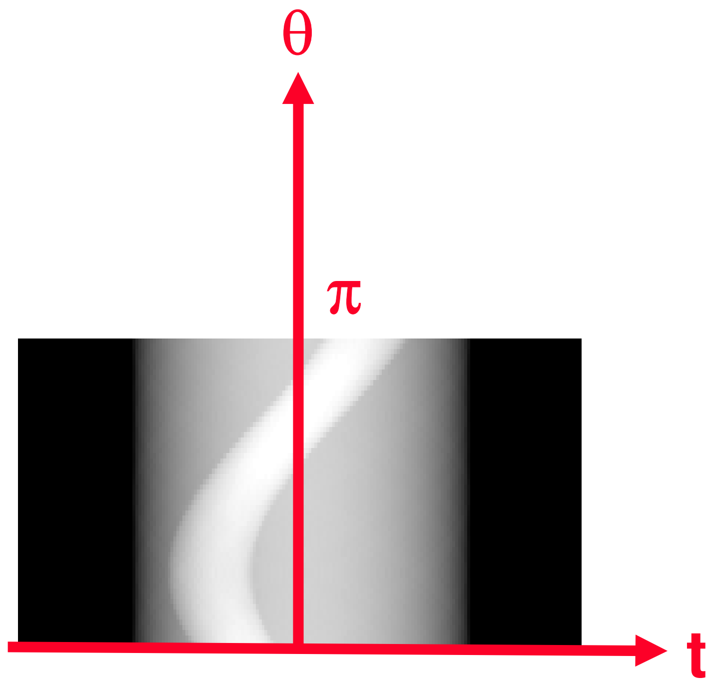
Tomography: Reconstruction from Projection



Projection & Sinogram

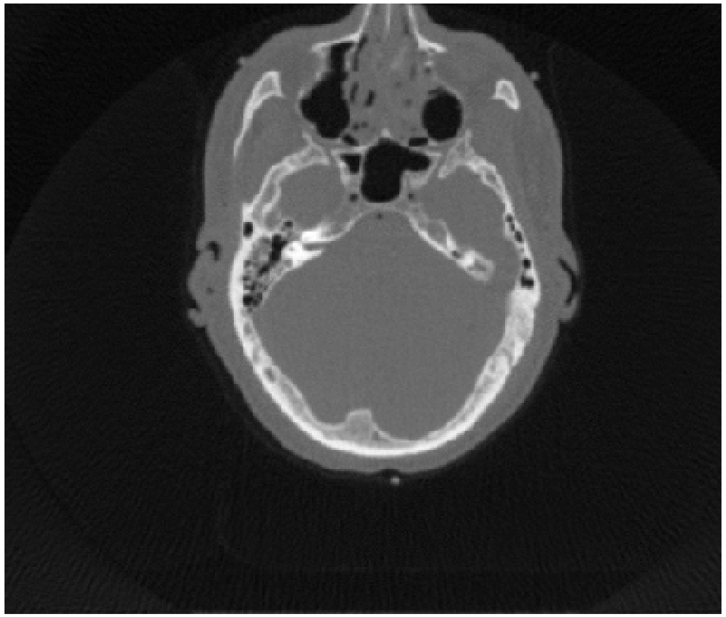
- Projection: All ray-sums in a direction
- Sinogram: collects all projections

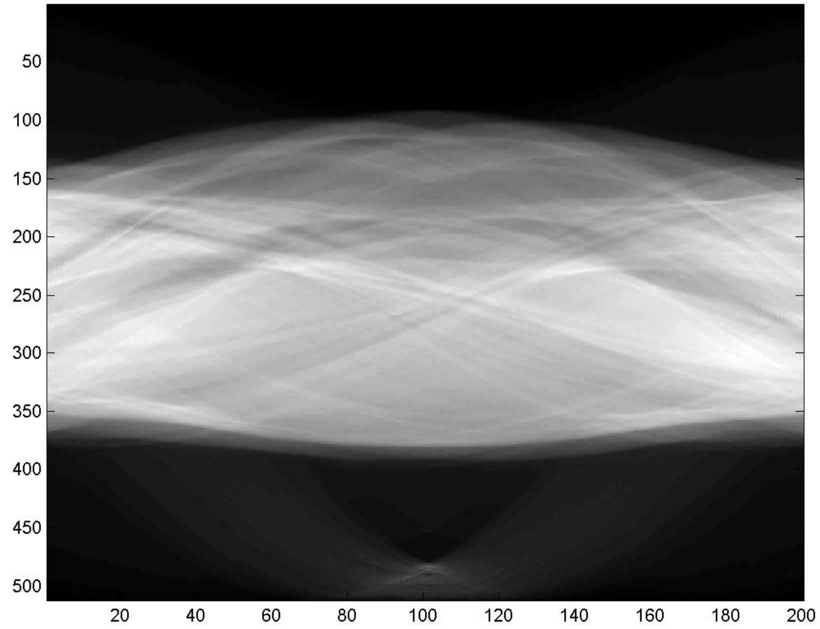




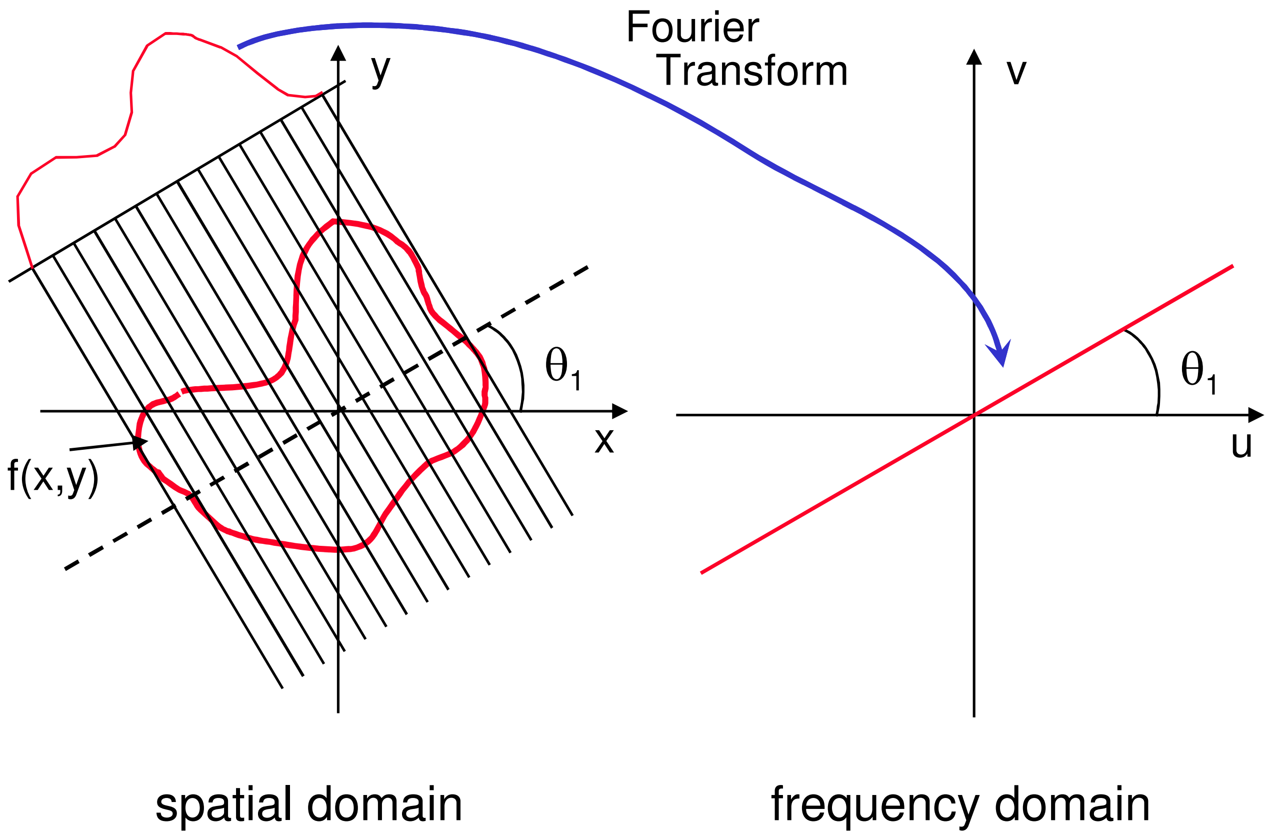
Sinogram

CT Image & Its Sinogram





The Slice Theorem



The Slice Theorem

f(x,y) = object

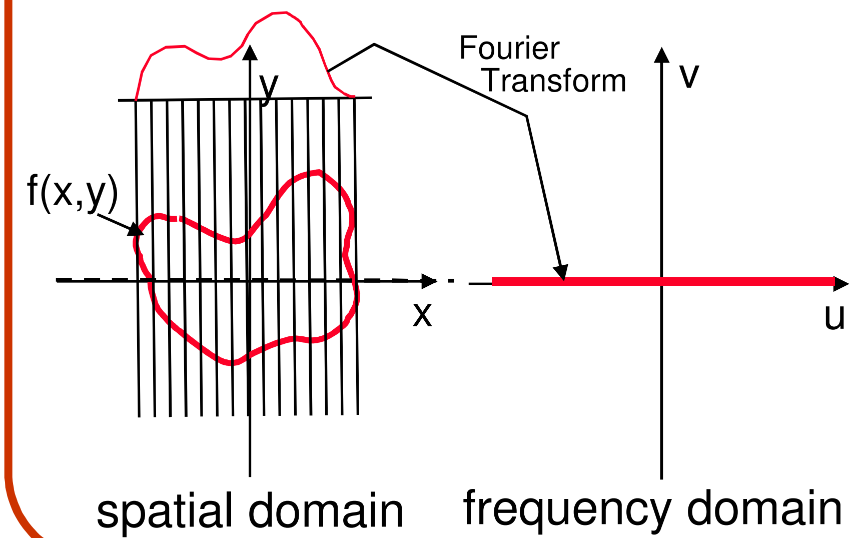
g(x) = projection of f(x,y) parallel to the y-axis: $g(x) = \int f(x,y)dy$

Fourier Transform of f(x,y): $F(u,v) = \int\int f(x,y)\, e^{-2\pi i(ux+vy)}\, dxdy$

Fourier Transform at v=0 : $F(u,0) = \int\int f(x,y)\, e^{-2\pi iux} dxdy$

$= \int \left[\int f(x,y)dy \right] e^{-2\pi iux} dx$

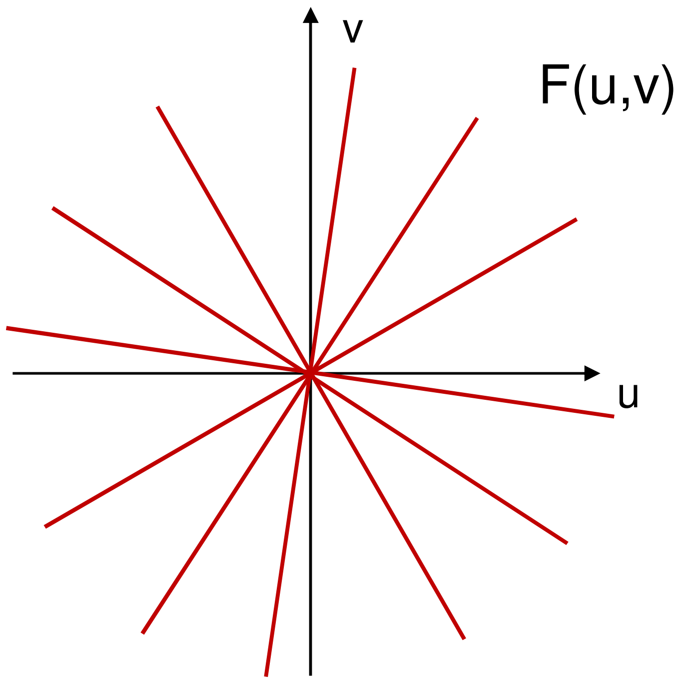
$= \int g(x)\, e^{-2\pi iux}\, dx = G(u)$



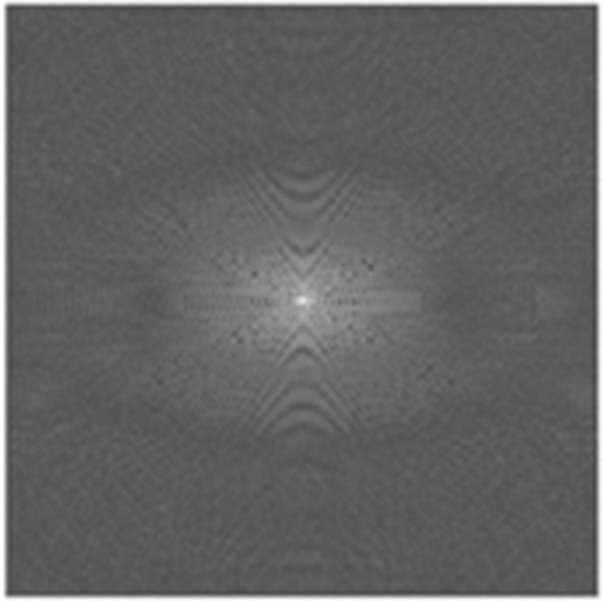
The 1D Fourier Transform of g(x)

Interpolation Method

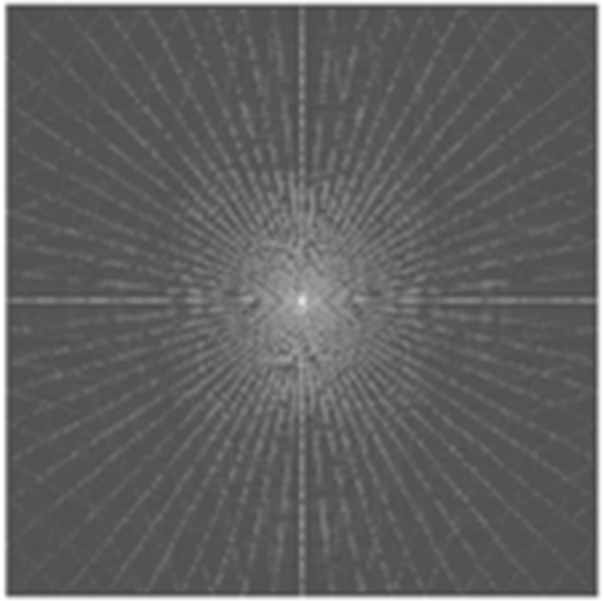
- **Interpolate** (linear, quadratic etc) in the frequency space to obtain all values in F(u,v).
- Perform **Inverse Fourier Transform** to obtain the image f(x,y).

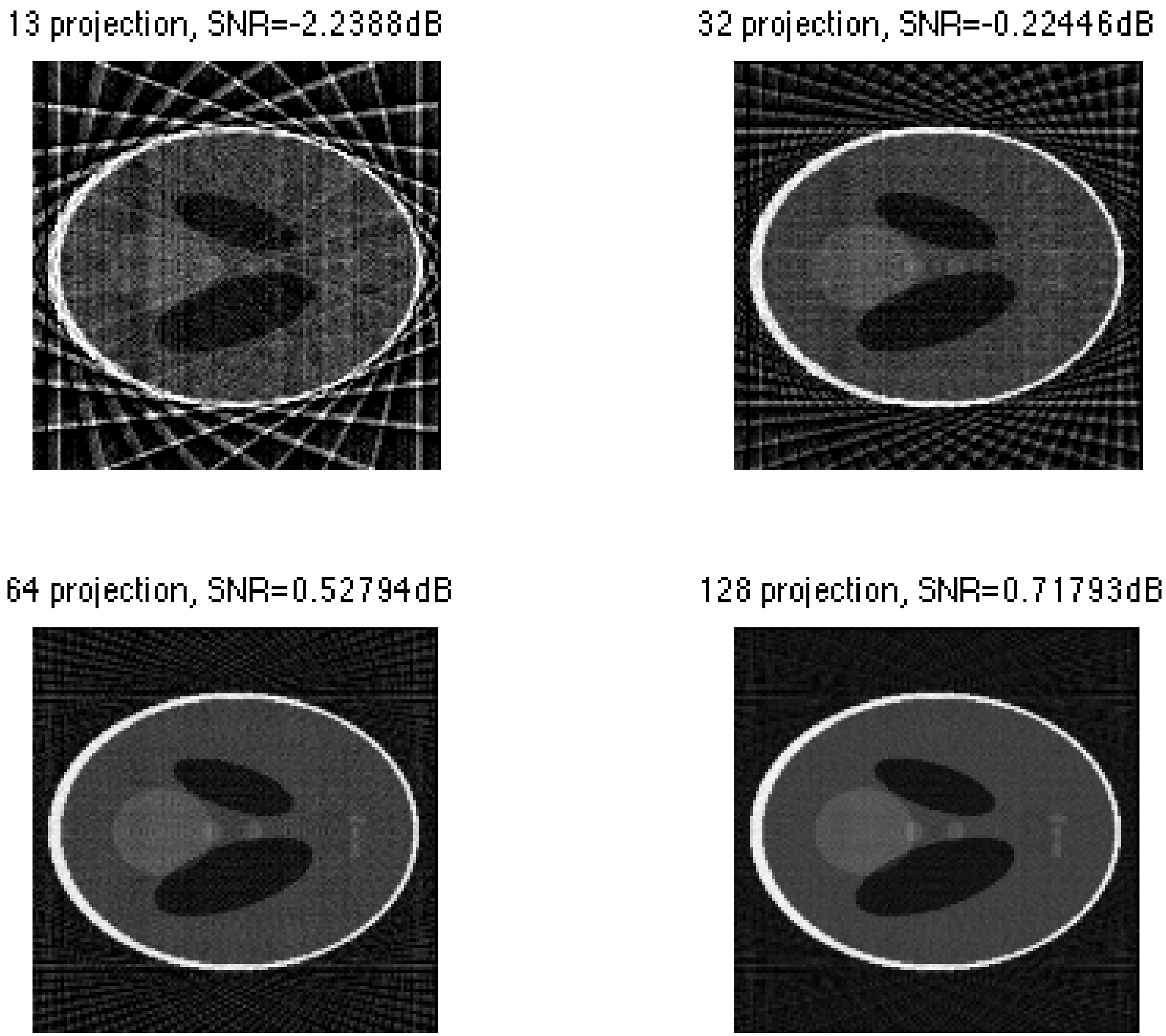


Fourier transform of the image



Fourier transform of the reconstruction




13 projection, SNR=-2.2388dB
32 projection, SNR=-0.22446dB
64 projection, SNR=0.52794dB
128 projection, SNR=0.71793dB

THE END