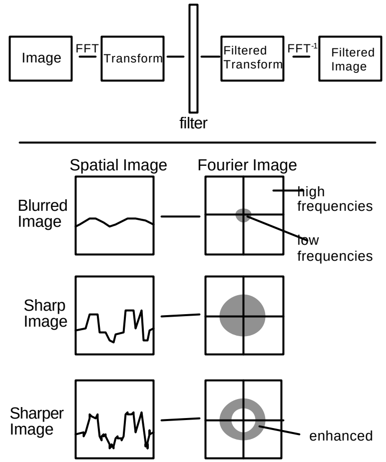
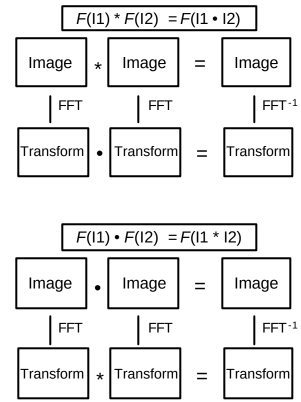
Image Enhancement - Frequency Domain

- Low Pass Filter
- High Pass Filter
- Band pass Filter
- Blurring
- Sharpening



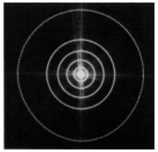
Convolution Theorem



Frequency Bands

Image

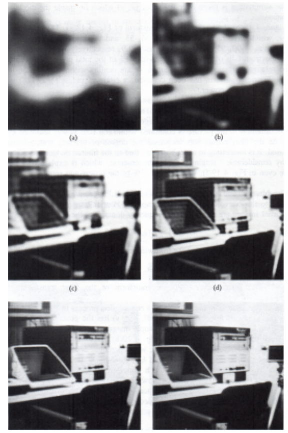
Fourier Spectrum



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

90, 95, 98, 99, 99.5, 99.9

Blurring - Ideal Low pass Filter



Low pass Filter

spatial domain — frequency domain

f(x,y) —— F(u,v)

$G(u,v) = F(u,v) \bullet H(u,v)$ (filter)

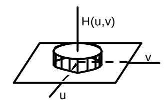
g(x,y) —— G(u,v)

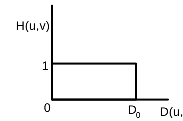
H(u,v) - Ideal Filter

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

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

D_0 = cut-off frequency



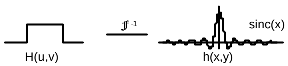


The Ringing Problem

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

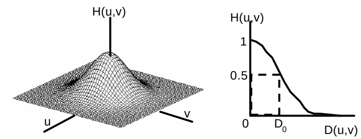
Convolution Theorm

$g(x,y) = f(x,y) * h(x,y)$



| D_0 ——— | Ringing radius + blur

H(u,v) - Butterworth Filter

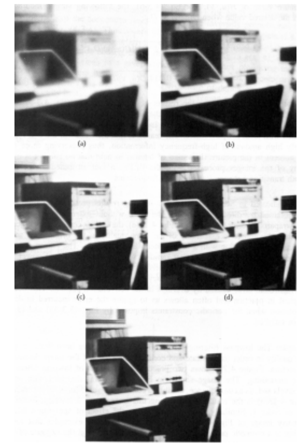


$$H(u,v) = \frac{1}{1 + (D(u,v)/D_0)^{2n}}$$

$$D(u,v) = \sqrt{u_2 + v_2}$$

Softer Blurring + no Ringing

Blurring - Butterworth Lowpass Filter



Low Pass Filtering - Image Smoothing

Original - 4 level Quantized Image

Smoothed Image

Original Noisy Image

Smoothed Image

Blurring in the Spatial Domain:

Averaging = convolution with $\frac{1}{4}\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}$

= point multiplication of the transform with sinc

Gaussian Averaging = convolution with $\frac{1}{16}\begin{bmatrix}1 & 2 & 1\\2 & 4 & 2\\1 & 2 & 1\end{bmatrix}$

= point multiplication of the transform with a gaussian.

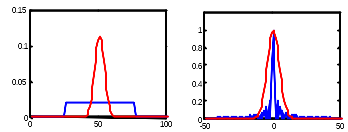


Image Domain   Frequency Domain

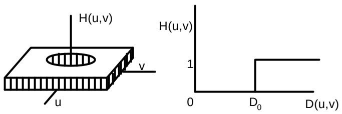
Image Sharpening - High Pass Filter

H(u,v) - Ideal Filter

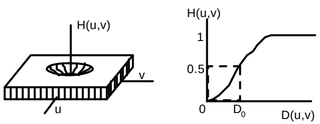
$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 \\ 1 & D(u,v) > D_0 \end{cases}$$

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

D_0 = cut-off frequency



H(u,v) - Butterworth Filter



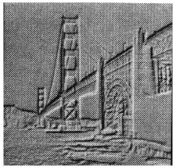
$$H(u,v) = \frac{1}{1 + (D_0/D(u,v))^{2n}}$$

$$D(u,v) = \sqrt{u_2 + v_2}$$

High Pass Filtering

Original

High Pass Filtered



High Frequency Emphasis

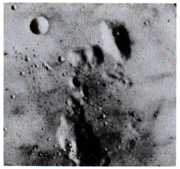
Emphasize High Frequency.
Maintain Low frequencies and Mean.

$$H'(u,v) = K_0 + H(u,v)$$

(Typically $K_0 = 1$)

High Frequency Emphasis - Example

Original



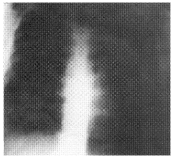
High Frequency Emphasis

Original

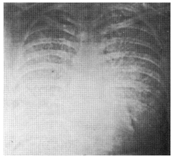
High Frequency Emphasis

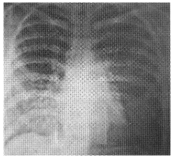
High Pass Filtering - Examples

Original

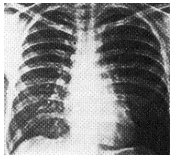


High pass Butterworth Filter





High Frequency Emphasis



High Frequency Emphasis + Histogram Equalization

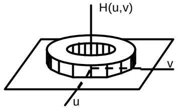
Band Pass Filtering

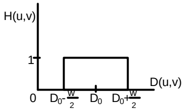
$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 - \frac{w}{2} \\ 1 & D_0 - \frac{w}{2} \le D(u,v) \le D_0 + \frac{w}{2} \\ 0 & D(u,v) > D_0 + \frac{w}{2} \end{cases}$$

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

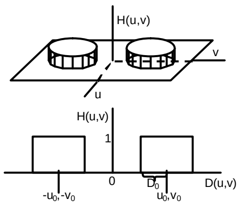
D_0 = cut-off frequency

w = band width





Local Frequency Filtering



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

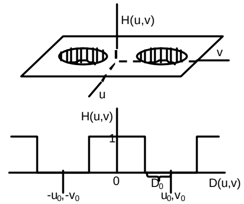
$D_1(u,v) = \sqrt{(u-u_0)^2 + (v-v_0)^2}$

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

D_0 = local frequency radius

u_0,v_0 = local frequency coordinates

Band Rejection Filtering



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

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

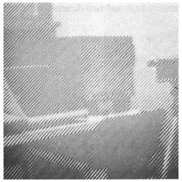
$D_2(u,v) = \sqrt{(u+u_0)^2 + (v+v_0)^2}$

D_0 = local frequency radius

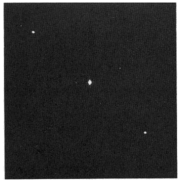
u_0,v_0 = local frequency coordinates

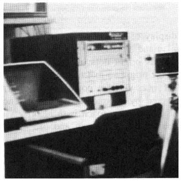
Band Reject Filter - Example

Original Noisy image



Fourier Spectrum



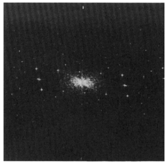


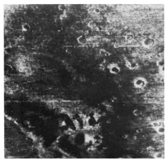
Band Reject Filter

Local Reject Filter - Example

Original Noisy image

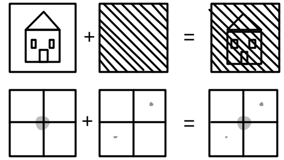
Fourier Spectrum





Local Reject Filter

Image Enhancement



Homomorphic Filtering

Reflectance Model:

Illumination	i(x,y)
Surface Reflectance	r(x,y)
Brightness	$f(x,y) = i(x,y) \cdot r(x,y)$

Assumptions:

Illumination changes "slowly" across scene
$\Rightarrow$ Illumination $\approx$ low frequencies.

Surface reflections change "sharply" across scene
$\Rightarrow$ reflectance $\approx$ high frequencies.

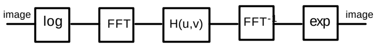
Goal: repress the low frequencies associated with I(x,y).
However:

$$\mathfrak{F}(i(x,y) \cdot r(x,y)) \neq \mathfrak{F}(i(x,y)) \cdot \mathfrak{F}(r(x,y))$$

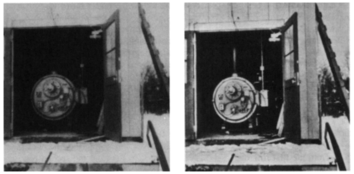
Perform:

$$z(x,y) = \mathbf{log}(f(x,y))$$
$$= \mathbf{log}(i(x,y) \cdot (r(x,y)) = \mathbf{log}(i(x,y)) + \mathbf{log}(r(x,y))$$

Homomorphic Filtering:



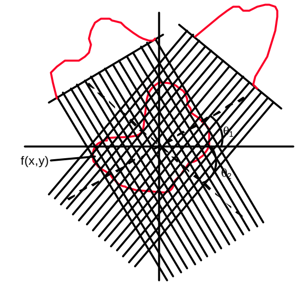
Homomorphic Filtering



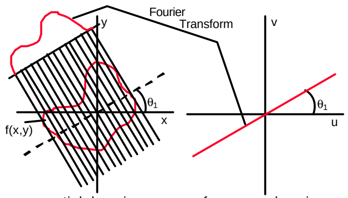
Original Filtered

Computerized Tomography

Reconstruction from projections



Reconstruction from Projections



spatial domain   frequency domain

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):

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

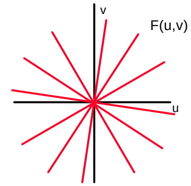
Fourier Transform at v=0 :

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

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

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

The 1D Fourier Transform of g(x)

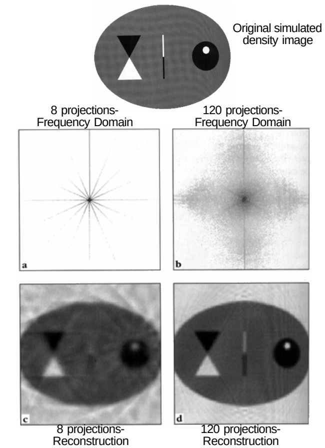


Interpolations Method:

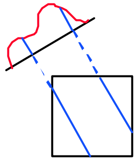
Interpolate (linear, quadratic etc) in the frequency space to obtain all values in F(u,v).

Perform **Inverse Fourier Transform** to obtain the image f(x,y).

Reconstruction from Projections - Example



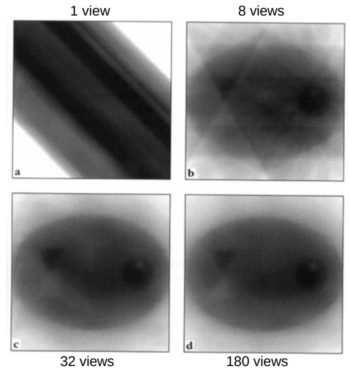
Back Projection Reconstruction



g(x) is Back Projected along the line of projection. The value of g(x) is added to the existing values at each point which were obtained from other back projections.

Note: a blurred version of the original is obtained. (for example consider a single point object is back projected into a blurred delta).

Back Projection Reconstruction - Example



Filtered Back Projection - Example