Chapter 1: Mathematical Preliminaries

Contents

- Signals and Systems.
- Fourier Transform Basics
- Analytical Image Reconstructing Techniques
- Iterative Reconstruction Methods
- Image Quality Assessment and System Optimization.

Reference book: Chapter 2 in <<Medical Imaging Signals and Systems>>, Prince and Links, Prentice Hall, 2006.

Part I: Signals and Systems

Reading Material:

Chapter 2 in

Medical Imaging Signal and Systems, 2nd Ed.

J. L. Prince et. al,

Prentice Hall, 2012.

Signals

Learning Objectives:

1. Mathematically, signals are typically described as functions of independent variables, typically time or space, with the signal itself being the dependent variable (amplitude).
2. Decomposing an arbitrary signal into a linear combination of a series of ***basis*** functions.

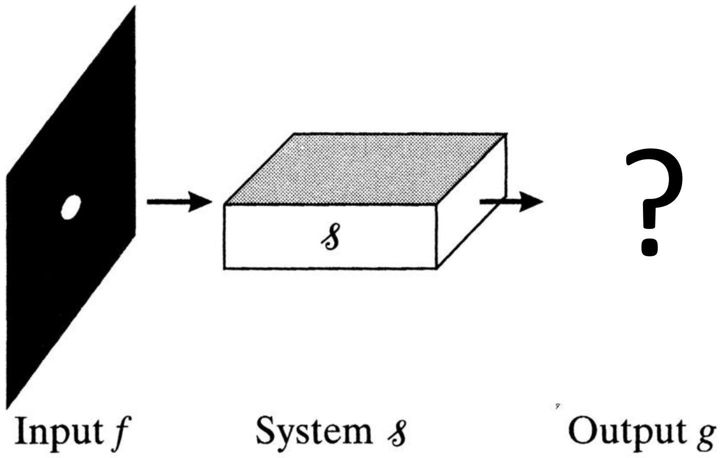
   2.1 Examples of the basis functions and their properties.

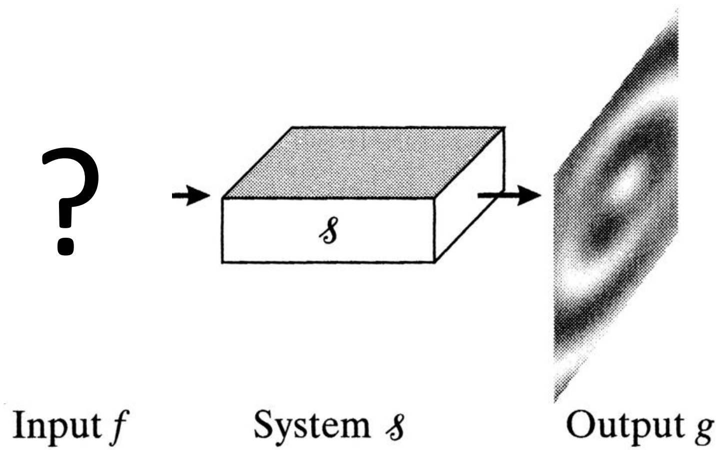
   2.2 How?

   2.3 Why?
3. Other useful signal functions, e.g., square function, triangle functions ...
4. Periodic signals and separable signal functions.

The Basic Problems in Imaging

The forward problem: Given an input signal and the known response of an imaging system, what is the output is going to be?





The inverse problem:

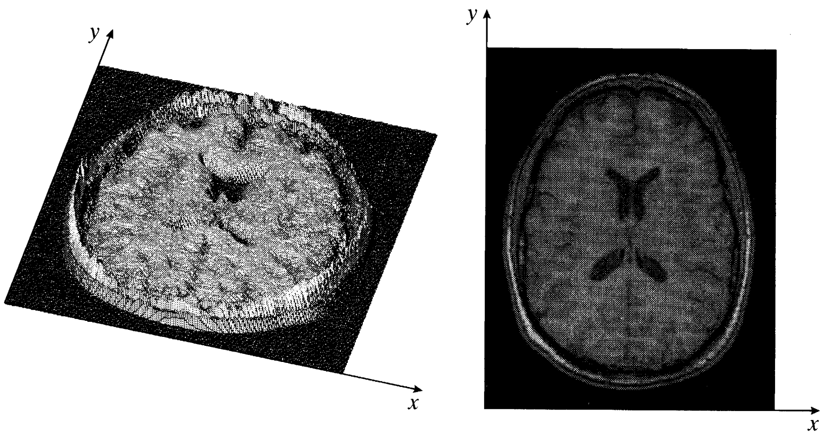
Given an output signal and the known system response, what should be the input signal that gave rise to the output data?

Introduction to Signals

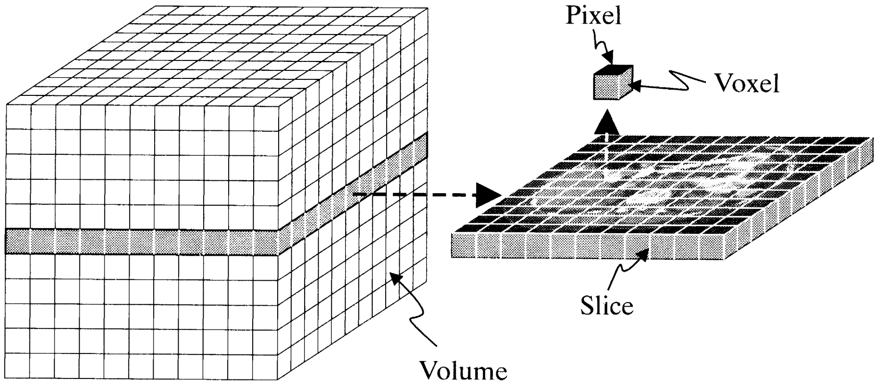
- Continuous signal:

A continuous 2-D signal

$$f(x, y), \quad -\infty \leq x, y \leq \infty$$



- Discrete signal: Pixel and voxel representations of a continuous signal



Continuous Fourier Transform

- For any square-integrable function f(x,y), a continuous Fourier transform is defined as

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$where\ j = \sqrt{-1}$$

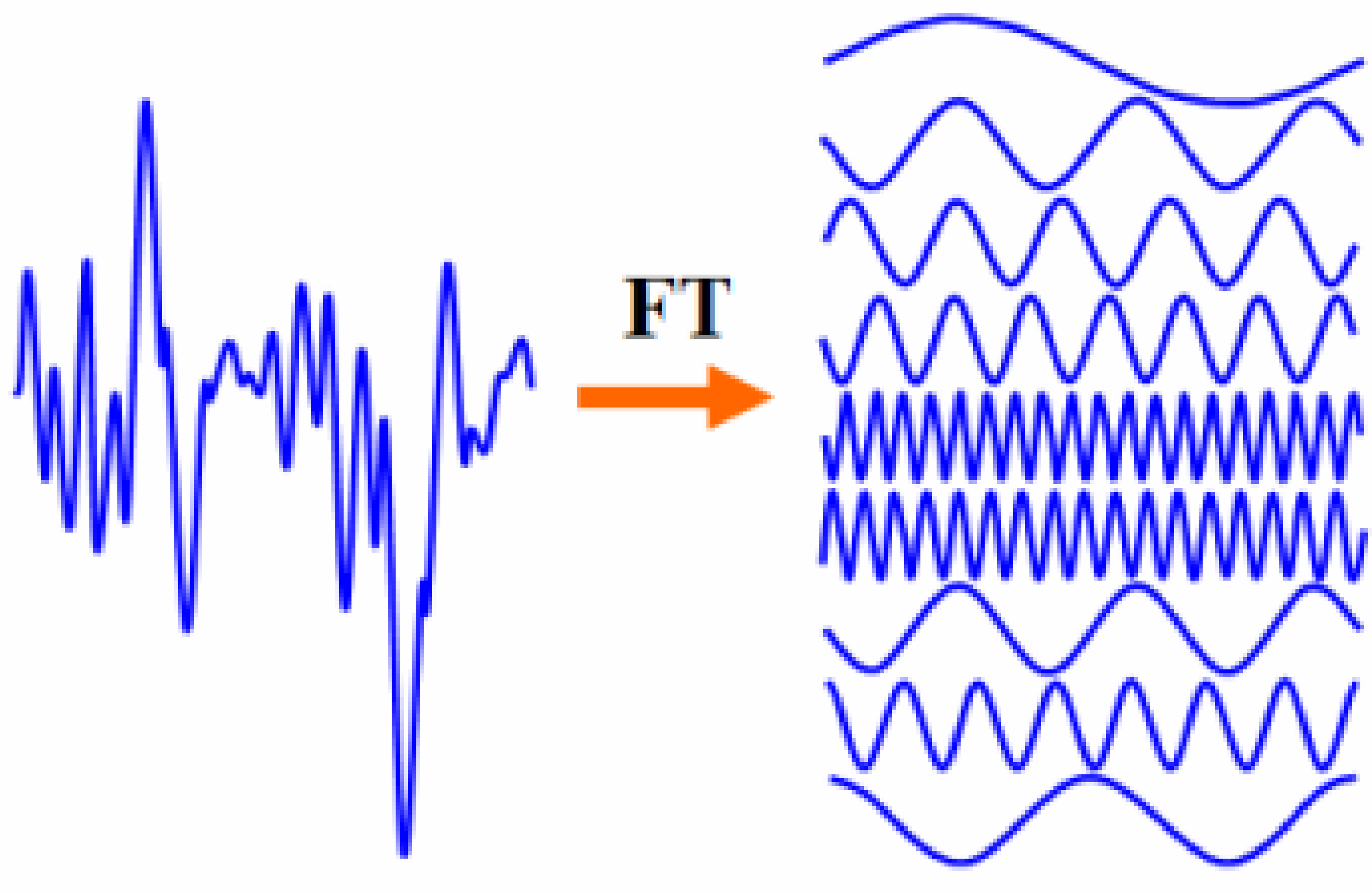
- We can also define an inverse Fourier transform as

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

- Both f(x,y) and F(u,v) have infinite support.
- Both f(x,y) and F(u,v) are defined on a continuum of values.
- f(x,y) and F(u,v) must contain the same information?

$$\boldsymbol{e^{-j\cdot 2\pi(ux+vy)} = cos[2\pi(ux+vy)] - j\cdot sin[2\pi(ux+vy)]}$$

Continuous Fourier Transform



www.revisemri.com

Discrete Fourier Transform in 1-D

The discrete Fourier transform (DFT) is defined as

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}, \mathrm{n} = 0,1,2,\ldots\mathrm{N}-1$$

$\mathrm{n} = 0$ corresponding to the DC component (spatial frenquency is zero)

$\mathrm{n} = 1,\ldots,\mathrm{N}/2-1$ are corresponding to the positive frenquencies $0 < \mathrm{u} < \mathrm{u_c}$

$\mathrm{n} = \mathrm{N}/2,\ldots,\mathrm{N}-1$ are corresponding to the negative frenquencies $-\mathrm{u_c} < \mathrm{u} < 0$

The inverse DFT is defined as

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{-\frac{j2\pi nk}{N}}, \mathrm{k} = 0,1,2,\ldots\mathrm{N}-1$$

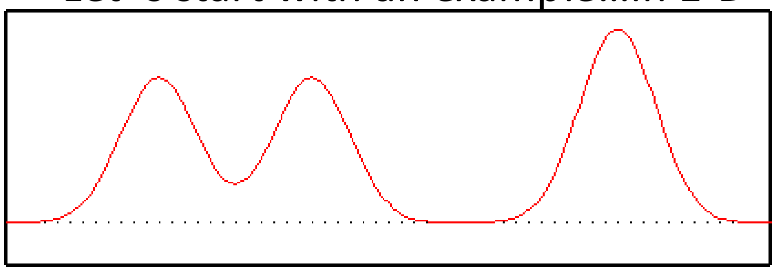
$$e^{-j\frac{2\pi nk}{N}} = cos\left[\frac{2\pi nk}{N}\right] - j \cdot sin[\frac{2\pi nk}{N}]$$

Continuous Fourier Transform

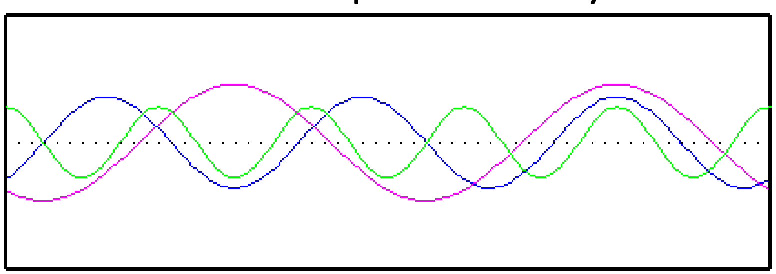
A Fourier Transform is an integral transform that re-expresses a function in terms of different sine waves of varying amplitudes, wavelengths, and phases.

So what does this mean exactly?

Let's start with an example…in 1-D

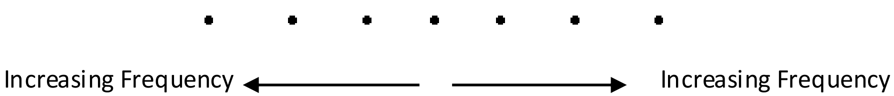


Can be represented by:



When you let these three waves interfere with each other you get your original wave function!

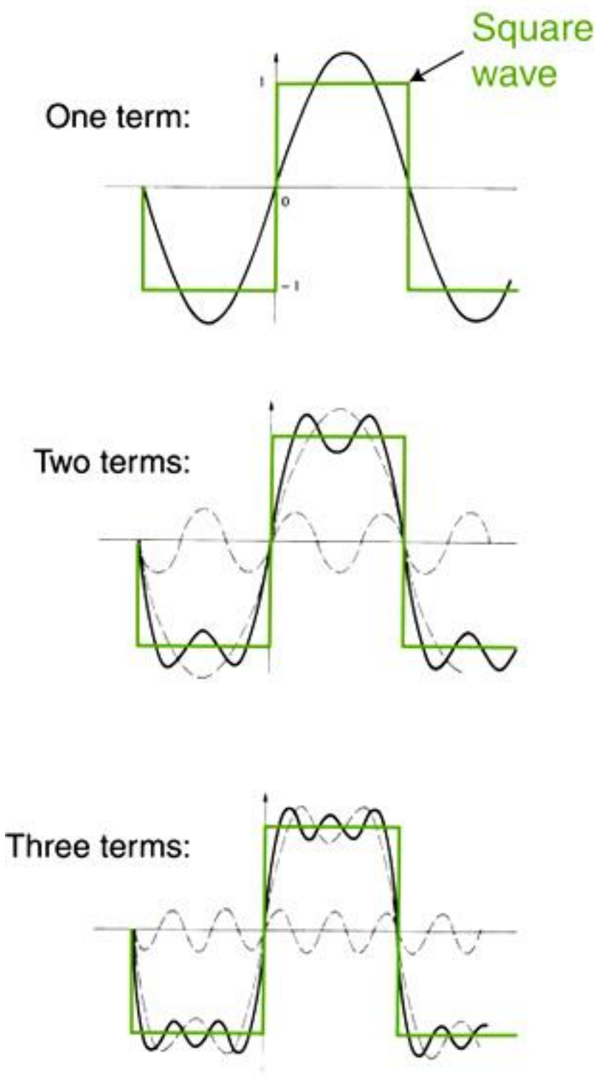
Since this object can be made up of 3 fundamental frequencies an ideal Fourier Transform would look something like this:



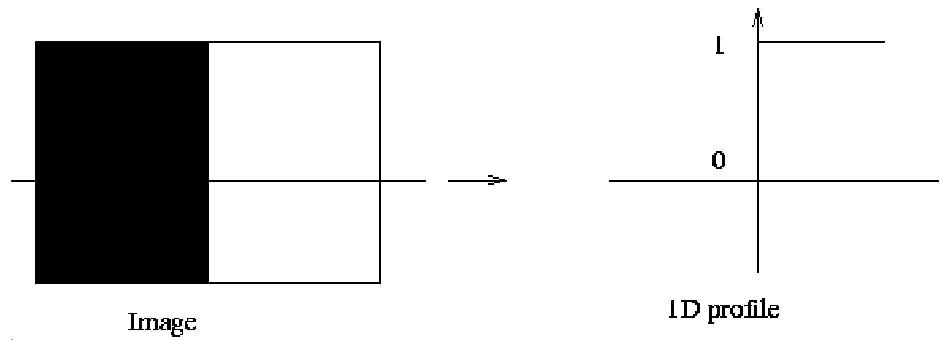
Notice that it is symmetric around the central point and that the number of points radiating outward corresponds to the distinct frequencies used in creating the image.

Fourier Transform and Spatial Frequency

Fourier transform provides information on the sinusoidal composition of a signal at different spatial frequencies.



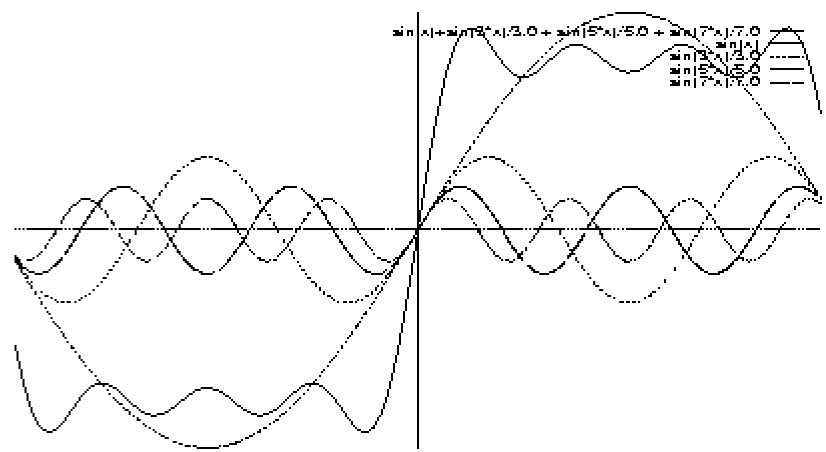
What is Spatial Frequency?



$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v) e^{-j2\pi(ux+vy)} du dv$$

$$e^{-j2\pi(ux+vy)}$$

$$= \cos[2\pi(ux+vy)] + j\sin[2\pi(ux+vy)]$$



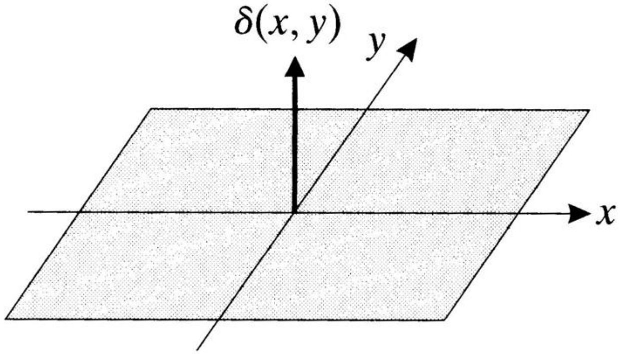
Point Impulse Signal

- A point source is mathematically represented by the delta function or Dirac function.

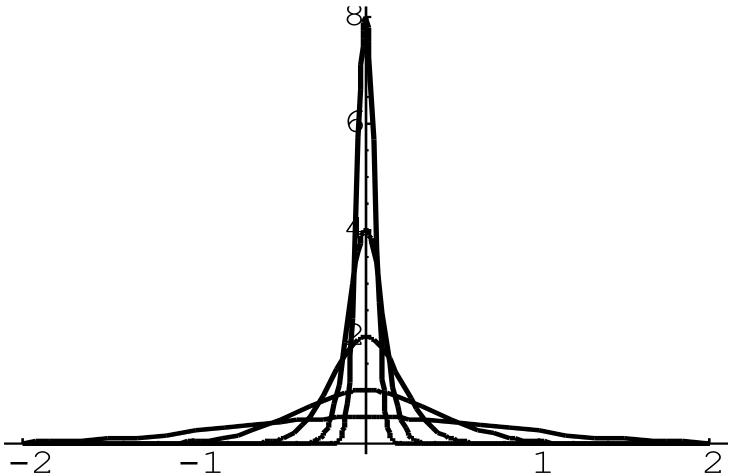
$$\delta(x, y)\begin{cases} \neq 0, & x = 0 \ and \ y = 0 \\ = 0, & otherwise \end{cases}$$

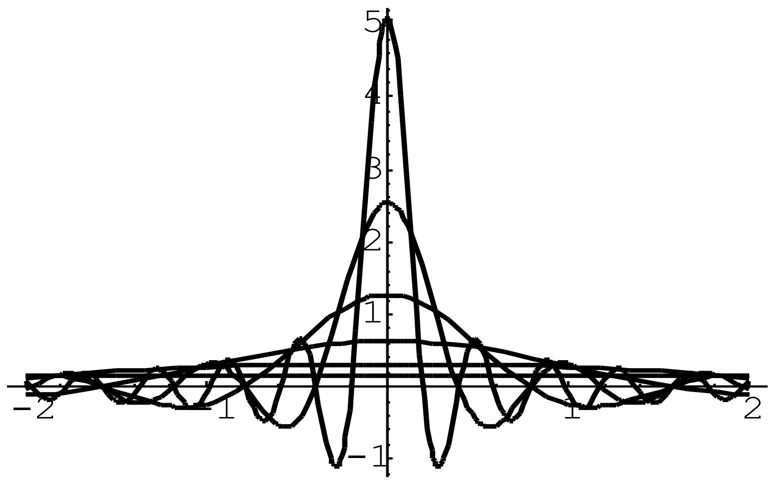
and

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(x, y)dxdy = 1$$



Point Impulse Signal





$$\delta(x) = \lim_{\alpha\to\infty} \alpha \; e^{-\pi\alpha^2 x^2}$$

$$\delta(x) = \lim_{\alpha\to\infty} \frac{\alpha}{\pi}\,\mathrm{sinc}(\alpha \mathrm{x});\; \mathrm{sinc}(\mathrm{x}) = \frac{\sin(x)}{x}$$

Point Impulse Signal

- The sampling property

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x-\xi, y-\eta)dxdy = f(\xi,\eta)$$

- The scaling property

$$\delta(ax,by) = \frac{1}{|ab|}\delta(x,y)$$

$$\delta(x,y)\begin{cases} \neq 0, & x=0 \; and \; y=0 \\ =0, & otherwise \end{cases}$$

and

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(x,y)dxdy = 1$$

Comb and Sampling Function

- The 2-D comb function

$$comb(x,y)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(x-m,y-n)$$

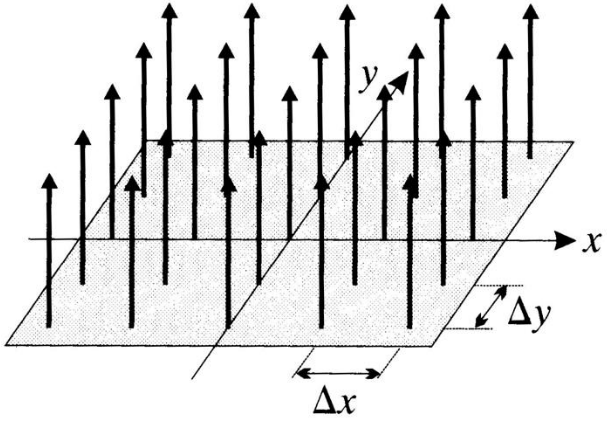
Comb and Sampling Function

- The 2-D sampling function

$$\delta_s(x, y, \Delta x, \Delta y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

where Δx and Δy are the sampling intervals

$$\delta_s(x, y, \Delta x, \Delta y) = \frac{1}{\Delta x \Delta y} comb\left(\frac{x}{\Delta x}, \frac{y}{\Delta y}\right)$$



$$comb(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m, y - n)$$

$$\delta(ax, by) = \frac{1}{|ab|} \delta(x, y)$$

Comb and Sampling Function

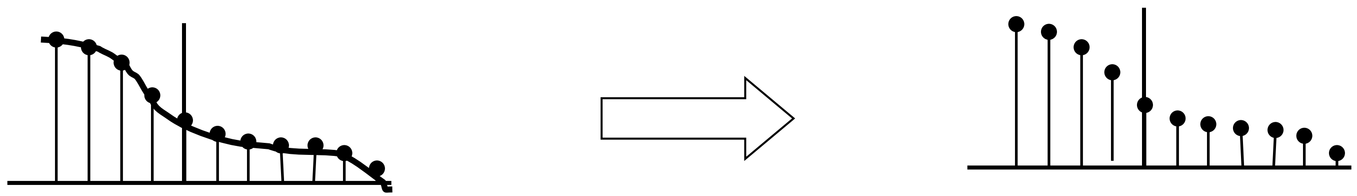
- The sampling function is critical for the discretization of continuous signals.
- The sampled signal function is then

$$\delta_s(x, y, \Delta x, \Delta y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

where Δx and Δy are the sampling intervals

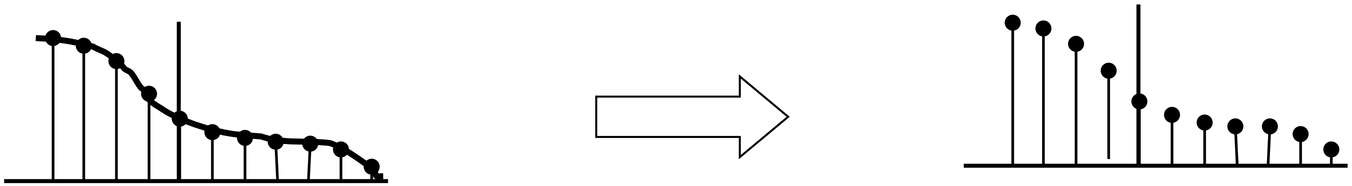
$$f_s(x, y) = f(x, y) \cdot \delta_s(x, y)$$

$$= f(x, y) \cdot \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$



A central question about sampling:

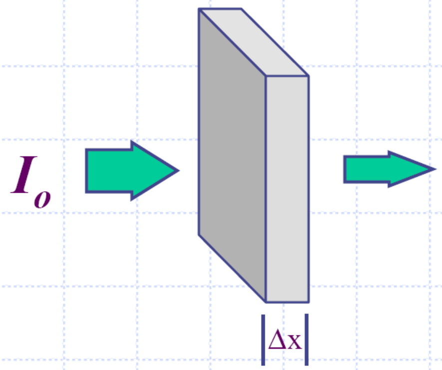
Will this continuous-to-discrete sampling process cause any loss in information?



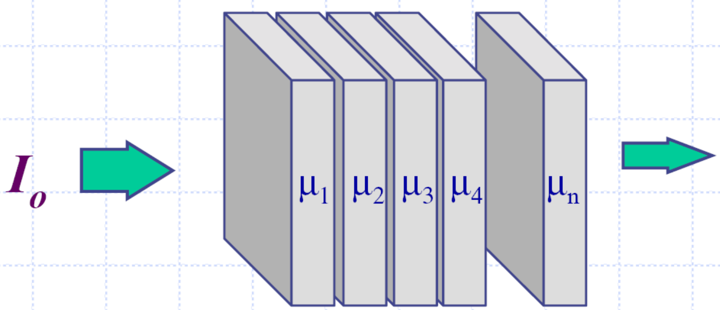
X-ray Planar Radiography (Revisited)

What are we measuring with planar X-ray radiography?

2 % attenuation change detectable in film



$$I = I_o e^{-\mu \Delta x}$$

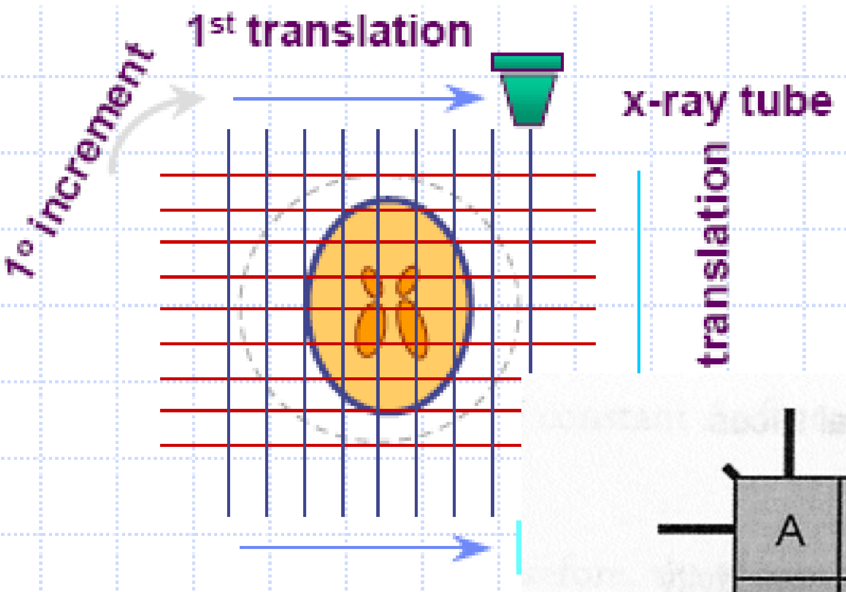


$$I = I_o e^{-\mu_1 \Delta x} e^{-\mu_2 \Delta x} \cdots e^{-}$$

$$= I_o e^{-(\mu_1 + \mu_2 + \cdots + \mu_n)\Delta x}$$

$$\Delta x \rightarrow 0, \qquad P = -\ln\left(\frac{I}{I_o}\right) = \int_{-\infty}^{\infty} \mu(x)\, dx$$

X-ray Computed Tomography (CT) (Revisited)



0.2 % attenuation change detectable in CT Images !!

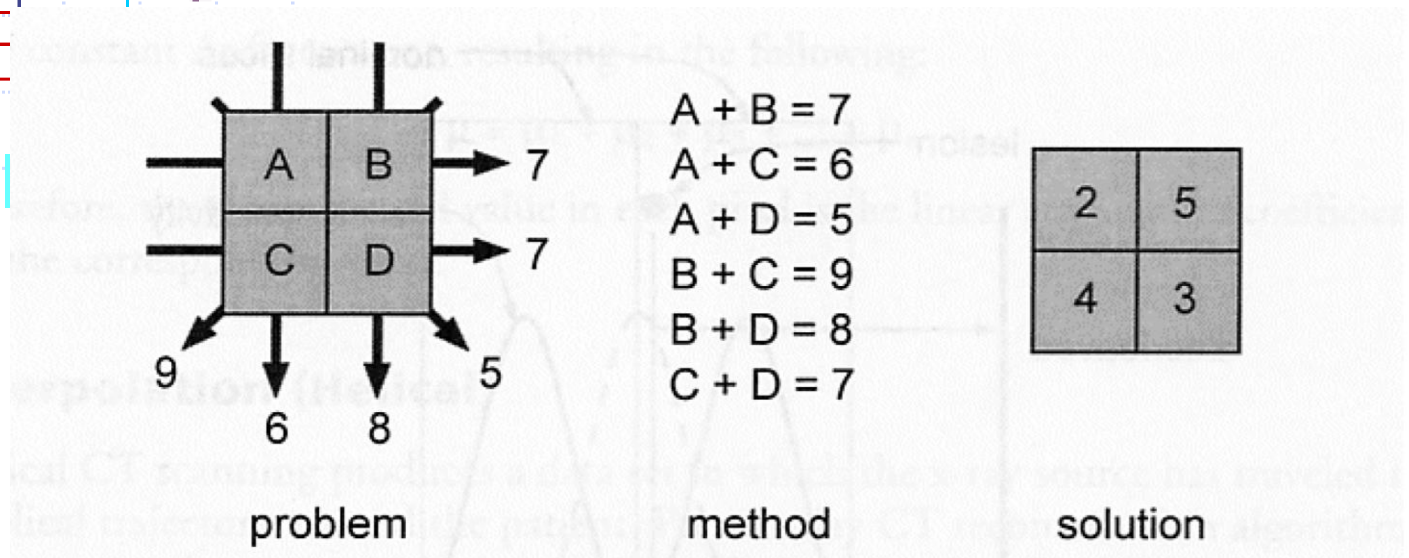
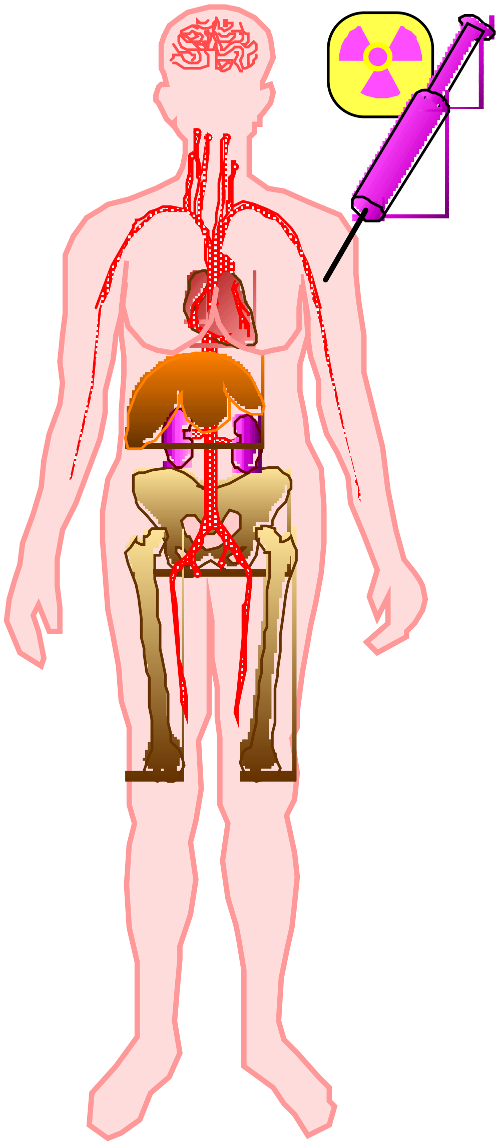


FIGURE 13-27. The mathematical problem posed by computed tomographic (CT) reconstruction is to calculate image data (the pixel values—A, B, C, and D) from the projection values *(arrows)*. For the simple image of four pixels shown here, algebra can be used to solve for the pixel values. With the six equations shown, using substitution of equations, the solution can be determined as illustrated. For the larger images of clinical CT, algebraic solutions become unfeasible, and filtered backprojection methods are used.

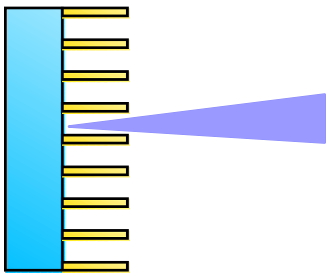
Emission Tomography (Revisited)



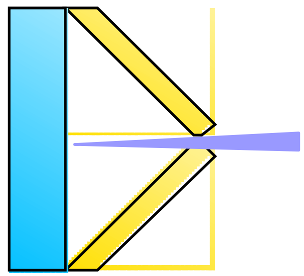
- Drug is labeled with radioisotopes that emit gamma rays.
- Drug localizes in patient according to metabolic properties of that drug.
- Trace (pico-molar) quantities of drug are sufficient.
- Radiation dose fairly small (<1 rem).

Drug Distributes in Body

Single Photon Emission Computed Tomography (SPECT)



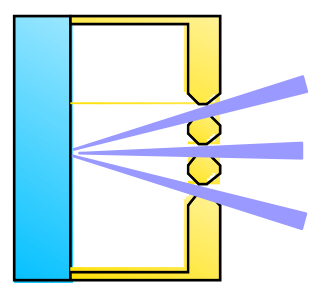
Collimator



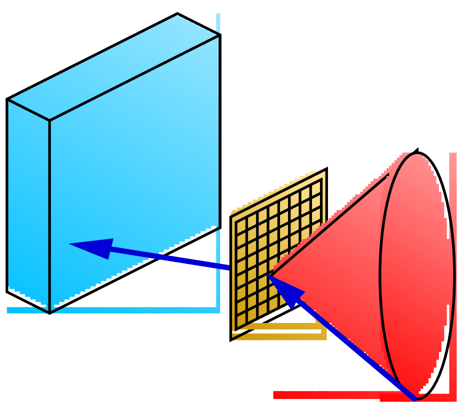
Pinhole

Collimator in front of the detector to select gamma rays from certain directions only …

Rotated around the object for collecting multiple projections …

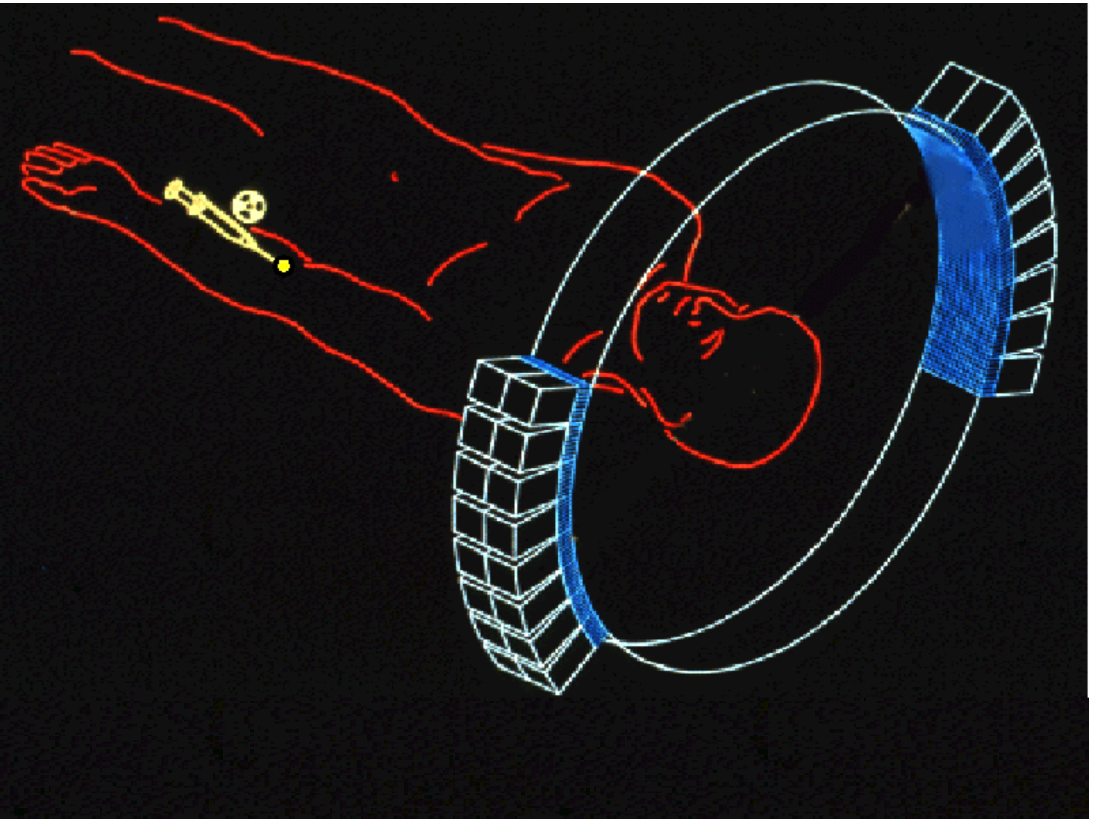


Coded Aperture



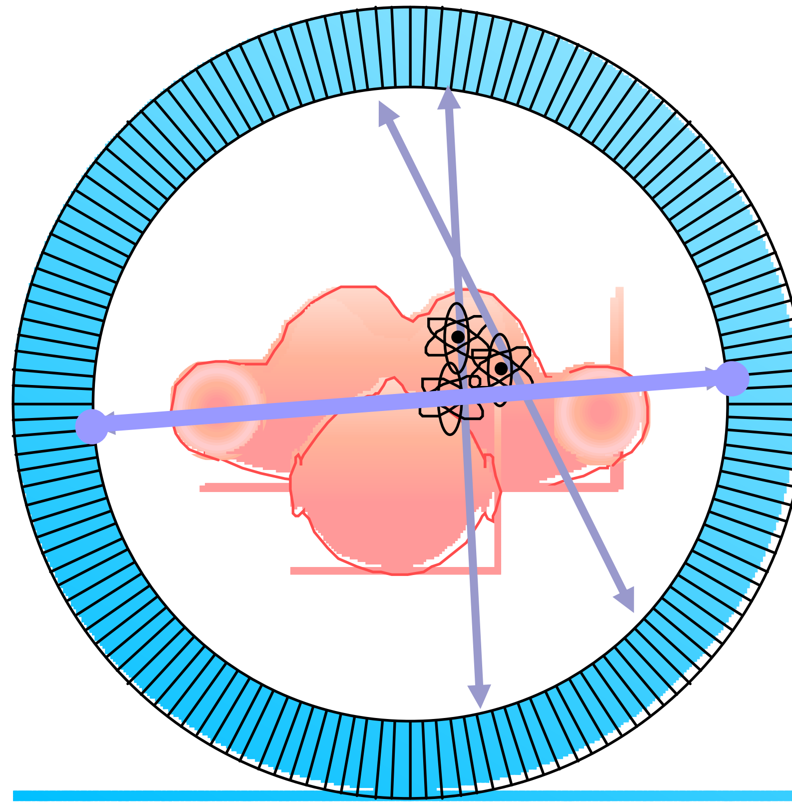
Compton

Positron Emission Tomography (Revisited)



Typical Detection Process

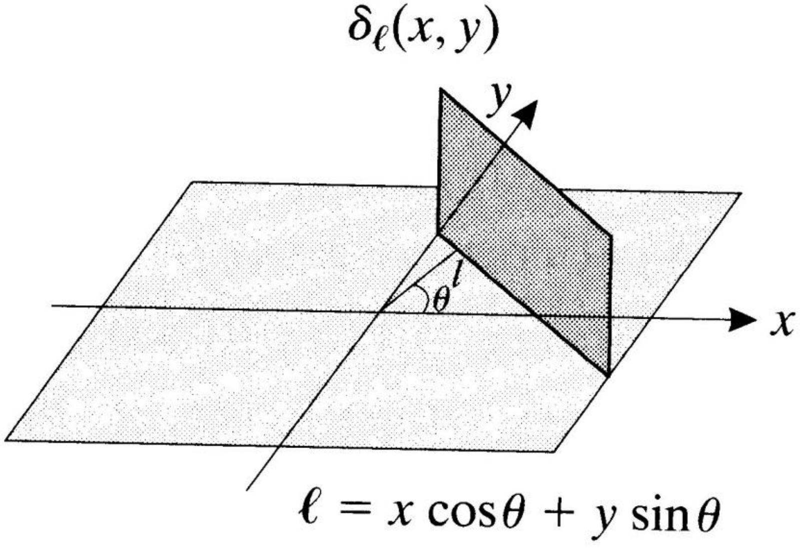
Collection of Line-integrals



Line Impulse Signal

$$\delta_L(x,y) = \delta(x\cos\theta + y\sin\theta - l)$$

$$where\ \delta(x) = \begin{cases} >0, & x\cos\theta + y\sin\theta = l \\ 0, & otherwise \end{cases}$$



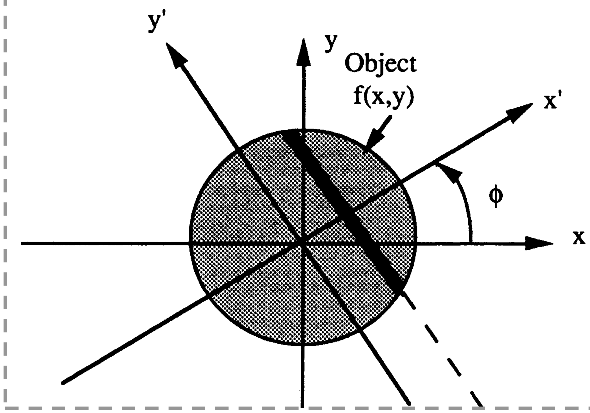
Line Impulse Signal

$$\delta_L(x, y) = \delta(x\cos\theta + y\sin\theta - l)$$

- It can be used to measure the spatial resolution of a given imaging system.

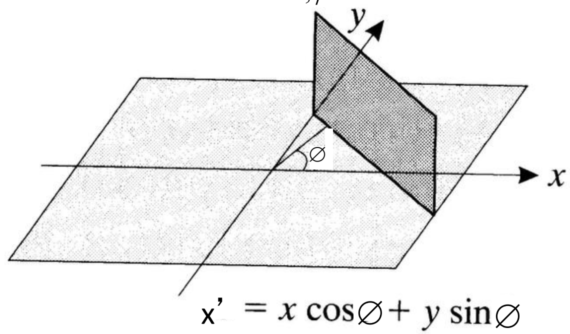
- It is used to calculate the line-integral projection data for a given 2-D object.

Line Impulse Signal



×

$$\delta_{x',\phi}(x,y) = \delta(x\cos\phi + y\sin\phi - x')$$



2-D integral

The value of the projection function $p_\phi(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The integral of the product of a line impulse function and a given 2-D signal gives the projection data from a given view …

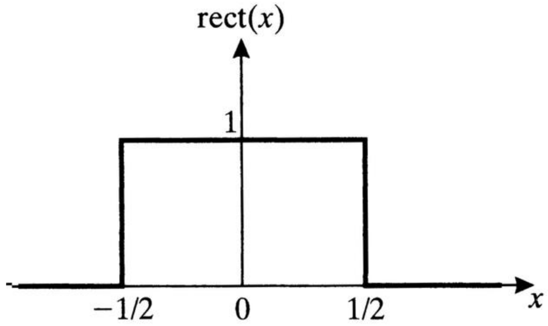
$$p_\phi(x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Line-impulse function is the key for modeling the projection process that underlying tomographic imaging process …

Rect Function

- Rect function:

$$rect(x,y)=\begin{cases}1, \text{ for } |\mathrm{x}|<\dfrac{1}{2} \text{ and } |\mathrm{y}|<\dfrac{1}{2}\\ 0, \text{ otherwise}\end{cases}$$



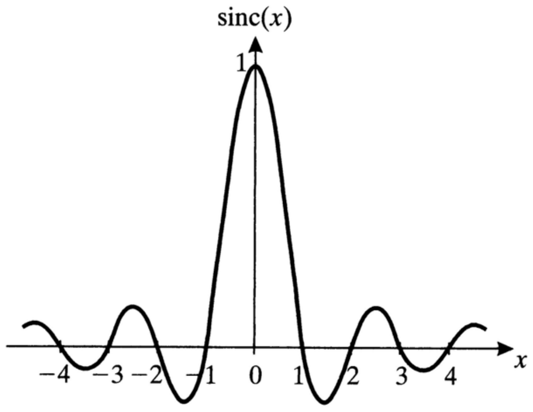
- It is normally used to pick up a particulate section of a given function:

$$f(x,y)\cdot rect\left(\frac{x-\xi}{w_X},\frac{y-\eta}{w_Y}\right)$$

Sinc Function

- The sinc function is defined as

$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$$



- The sinc function is normalized.

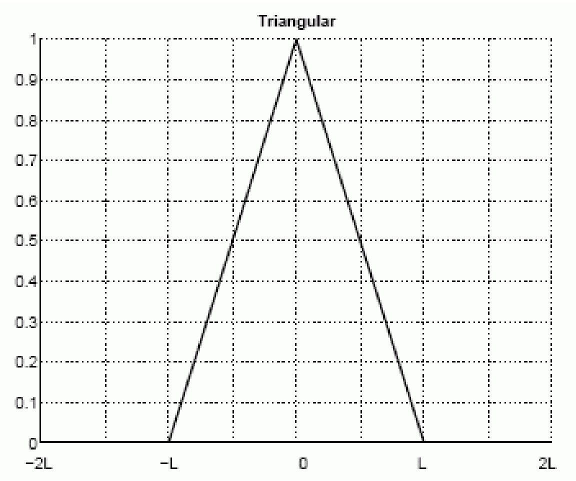
$$\int_{-\infty}^{\infty} \mathrm{sinc}(x)\,dx = 1$$

Any arbitrary band-limited signal can be written as a weighted sum of multiple sinc functions … (the Nyquist Sampling Theorem)

Triangular Signals and Gaussian Signals

- Triangular function:

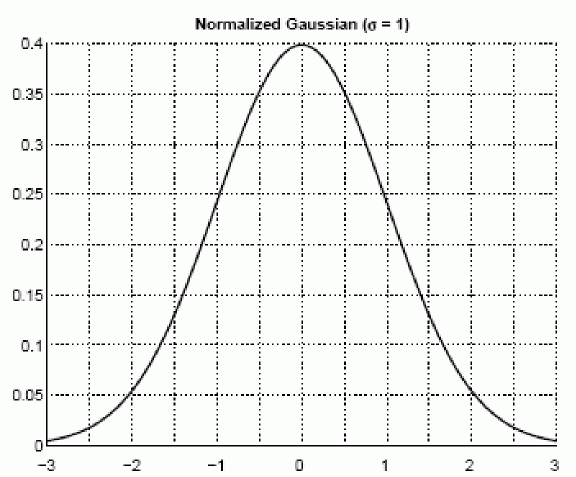
$$Tri\left(\frac{x}{2L}\right) = 1 - \frac{|x|}{L} \quad \text{for } |x| < L$$
$$= 0 \quad \text{for } |x| > L$$



- Normalized Gaussian function:

$$G_{1D}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

$$G_{2D}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



Separable Signals and Periodic Signals

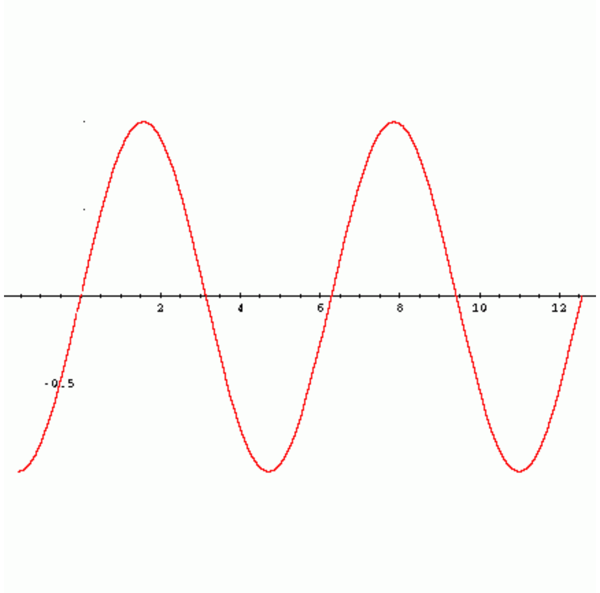
- The separable signals is a class of continuous signals that satisfy

$$f(x,y) = f_1(x) \cdot f_2(y)$$

- A 2-D signal is periodic if

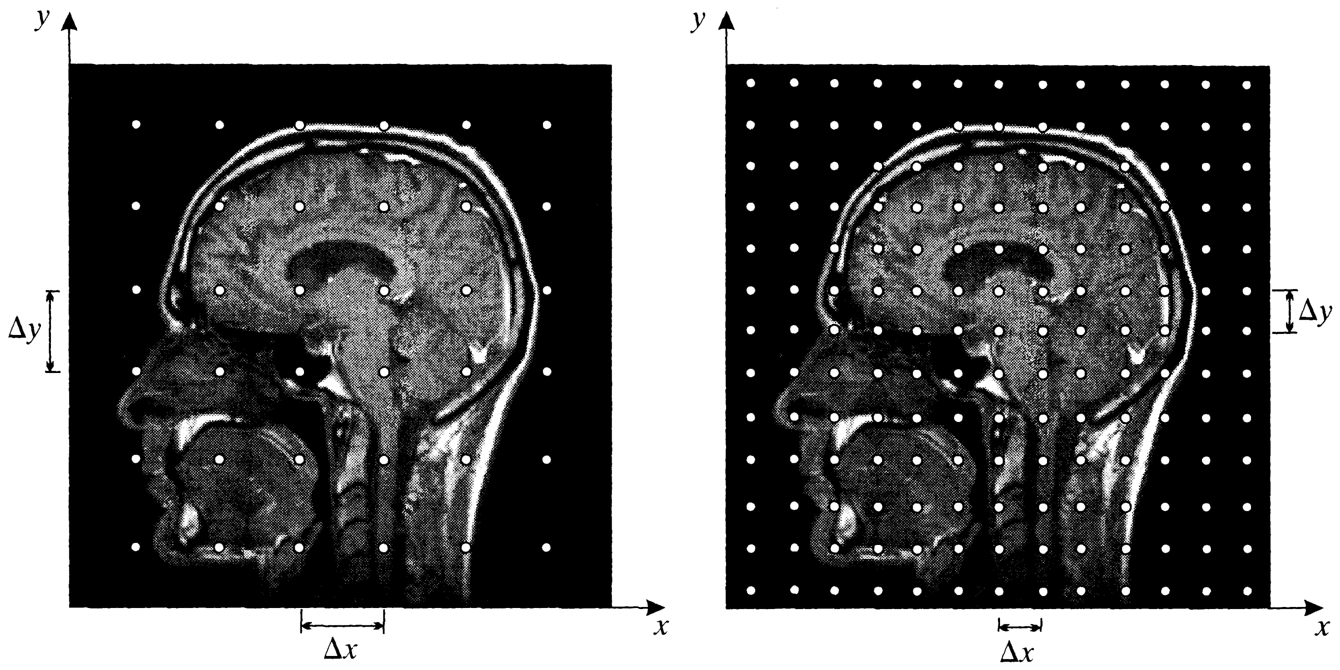
$$f(x,y) = f(x+X,y) = f(x,y+Y)$$

where X and Y are the signal periods



Two-Dimensional Sampling

$$
\begin{aligned}
f_s(x,y) &= f(x,y)\cdot\delta_s(x,y,\Delta x,\Delta y)\\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(x,y)\cdot\delta(x-n\Delta x,\, y-m\Delta y)\\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(n\Delta x, m\Delta y)\cdot\delta(x-n\Delta x,\, y-m\Delta y)
\end{aligned}
$$

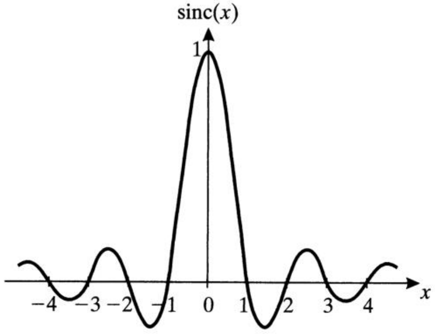


Restoration of the Original 2-D Function

Given that the Nyquist sampling condition is met, the original function could be recovered exactly as

$$
\begin{aligned}
f(x,y) &= f_s(x,y) * h(x,y) \\
&= f_s(x,y) * \left[\frac{1}{\Delta x}\cdot \mathrm{sinc}(\frac{x}{\Delta x})\right]\left[\frac{1}{\Delta y}\cdot \mathrm{sinc}(\frac{y}{\Delta y})\right] \\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(n\Delta x, m\Delta y)\cdot \delta(x-n\cdot\Delta x, y-m\cdot\Delta y) * \left\{\left[\frac{1}{\Delta x}\cdot \mathrm{sinc}(\frac{x}{\Delta x})\right]\left[\frac{1}{\Delta y}\cdot \mathrm{sinc}(\frac{y}{\Delta y})\right]\right\} \\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \frac{1}{\Delta x\Delta y} f(n\Delta x, m\Delta y)\cdot \mathrm{sinc}(\frac{x-n\cdot\Delta x}{\Delta x})\cdot \mathrm{sinc}(\frac{y-m\cdot\Delta y}{\Delta y})
\end{aligned}
$$

$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$$



System

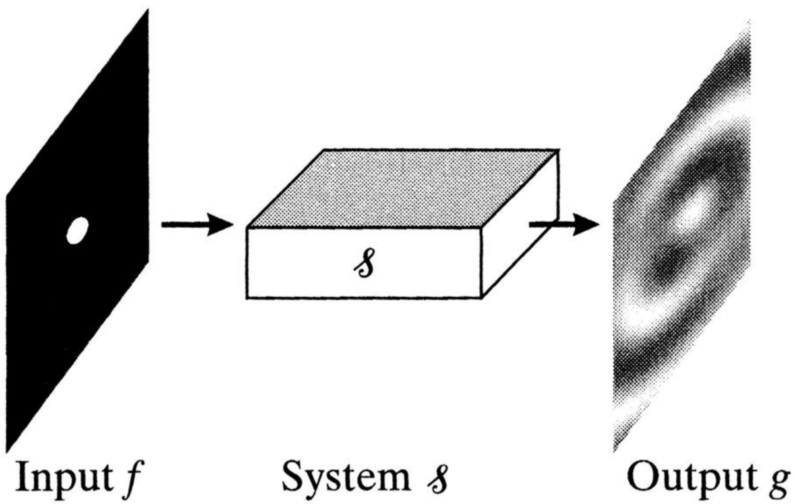
Learning Objectives:

1. Mathematical representation of an imaging system.
2. Linear systems:
   a. The superposition principle.
   b. Example of linear systems.
   c. Why does it matter?
3. Shift-invariant systems
4. The response of a linear and shift-invariant (LSI) system.

General Concept of a System

- A continuous-to-continuous system is defined as

$$g(x, y) = \mathcal{S}[f(x, y)]$$



- A system is a mapping process from an input signal to the output signal

Linear Systems

Linear Systems

A system is linear if it satisfies the **superposition principle**

$$S\left[f(x,y)=\sum_{i=1}^{I} w_i \cdot f_i(x,y)\right]=\sum_{i=1}^{I} w_i \cdot S[f_i(x,y)]$$

where

$f(x,y)$ is the input signal,

$S[\cdot]$ is an operator that represents the system,

$f(x,y)$ is the total input signal, and

w_is are weighting factors.

Linear Systems

A linear imaging system processes an input signal (such as light intensity from an object) to produce an output signal (the resulting image). To be considered "linear," the system must satisfy two core properties:

1. **Additivity:** If you have two different inputs, $f(x)$ and $g(x)$, the output of the system operating on their sum is equal to the sum of the outputs if the system had operated on each input individually:

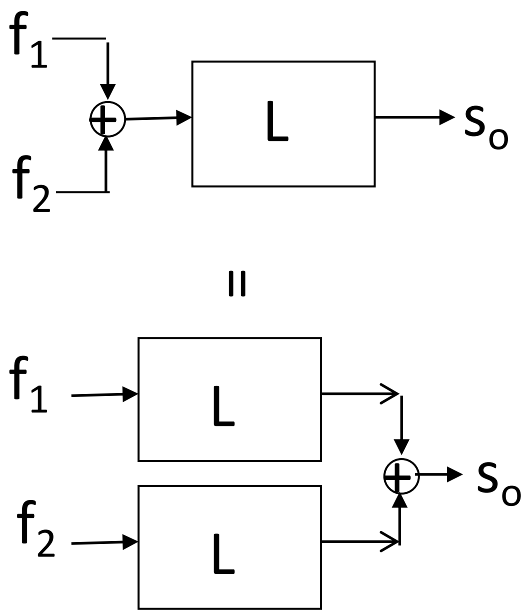
$$S[f(x) + g(x)] = S[f(x)] + S[g(x)].$$

1. **Homogeneity:** A change in the input signal's amplitude results in a corresponding, scaled change in the output signal's amplitude.

$$S[a \cdot f(x)] = a \cdot S[f(x)].$$

Linear Systems – An Example

For example, consider an amplifier with gain A:



$$S[w_1 f_1 + w_2 f_2] = A(w_1 f_1 + w_2 f_2)$$
$$= A w_1 f_1 + A w_2 f_2 = w_1 S[f_1] + w_2 S[f_2]$$

It satisfies the Superposition Principle

$$S\left[f(x,y) = \sum_{i=1}^{I} w_i \cdot f_i(x,y)\right] = \sum_{i=1}^{I} w_i \cdot S[f_i(x,y)]$$

Linear Systems – Why Important?

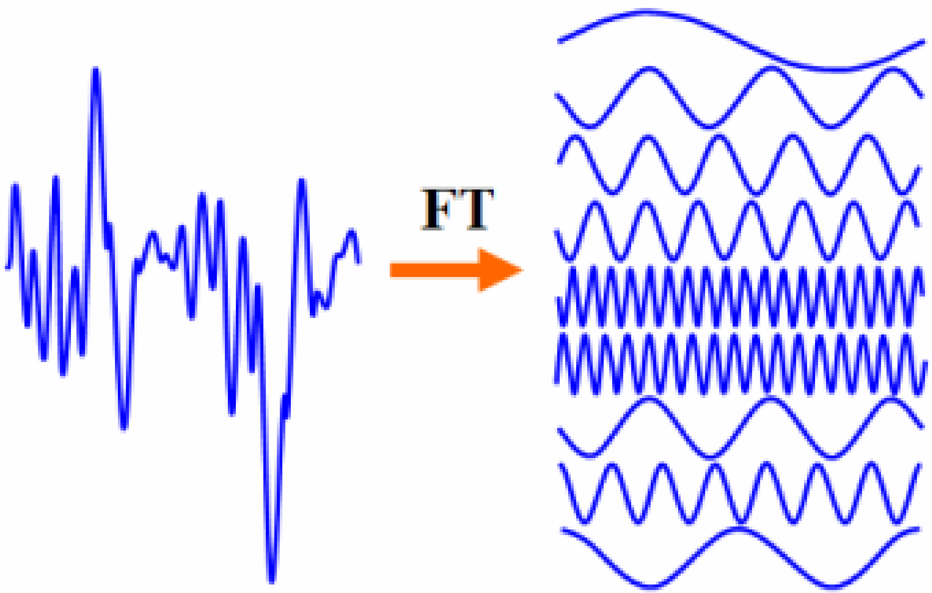
- Linear systems are mathematically easier to model.

$$g(x,y) = S\left[f(x,y)\right] = S\left[\sum_{i=1}^{I} w_i \cdot f_i(x,y)\right] = \sum_{i=1}^{I} w_i \cdot S\left[f_i(x,y)\right]$$

- Many imaging systems used in medical and other applications as designed to be linear systems.

Linear Systems – Why Important?

Consider that we can break up an arbitrary signal through Fourier transform



If we send the signal into a linear system, then the output is given as

$$g(x,y) = S\left[f(x,y)\right] = S\left[\sum_{i=1}^{I} w_i \cdot f_i(x,y)\right] = \sum_{i=1}^{I} w_i \cdot S\left[f_i(x,y)\right]$$

Impulse Response Function

Comb and Sampling Function (Revisited)

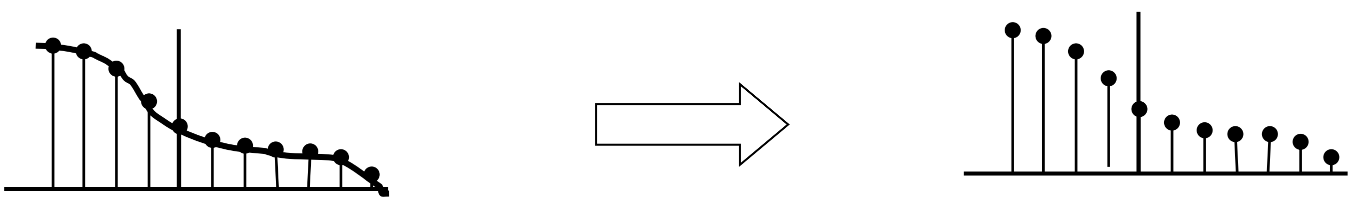
- The sampling function is critical for the discretization of continuous signals.
- The sampled signal function is then

$$\delta_s(x, y, \Delta x, \Delta y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

where Δx and Δy are the sampling intervals

$$f_s(x, y) = f(x, y) \cdot \delta_s(x, y)$$

$$= f(x, y) \cdot \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$



Linear Systems (Revisited)

Given a sampled input signal,

$$f_s(x,y) = f(x,y)\cdot s(x,y)$$

$$= \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\left[f(m\Delta x, n\Delta y)\cdot \delta(x-m\Delta x, y-n\Delta y)\right]$$

The response of the linear system is

$$g(x,y) = S\left[f_s(x,y)\right]$$

$$= S\left[\frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\left[f(m\Delta x, n\Delta y)\cdot \delta(x-m\Delta x, y-n\Delta y)\right]\right]$$

$$= \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\left[\frac{1}{\Delta x \Delta y} f(m\Delta x, n\Delta y)\cdot S\left[\delta(x-m\Delta x, y-n\Delta y)\right]\right]$$

Linear Systems (Revisited)

Alternatively, if we consider a continuous input signal, $f(x,y)$, the output from a system is given by,

The Sampling property of the Delta function

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x-\xi,y-\eta)dxdy = f(\xi,\eta)$$

or

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\delta(\xi-x,\eta-y)dxdy = f(x,y)$$

$$\Leftrightarrow$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\delta(x-\xi,y-\eta)dxdy = f(x,y)$$

$$g(x,y) = \mathrm{S}[f(x,y)]$$

$$= \mathrm{S}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,\delta(x-\xi,y-\eta)d\xi d\eta\right]$$

$$= \mathrm{S}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,\delta(\xi-x,\eta-y)d\xi d\eta\right]$$

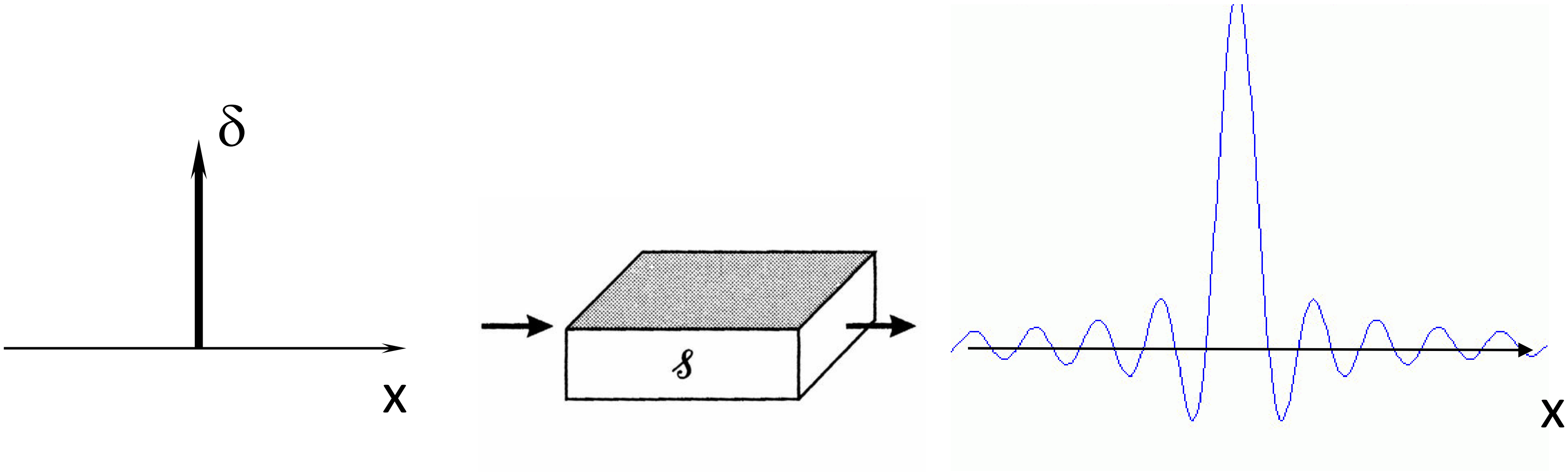
$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{S}[f(\xi,\eta)\delta(x-\xi,y-\eta)]d\xi d\eta$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,\mathrm{S}[\delta(x-\xi,y-\eta)]d\xi d\eta$$

Assuming the system is linear

Impulse Response Function

One of the most common shape for impulse responses used in imaging application



The **impulse response function** (in the 2-D case) is defined as

$$h(x, y, \xi, \eta) \equiv \mathrm{S}[\delta(x-\xi, y-\eta)].$$

Impulse Response Function (IRF)

For a linear system, knowing the IRF enables one to compute the output from any arbitrary input function.

$$
\begin{aligned}
g(x,y) &= \mathrm{S}[f(x,y)] \\
&= \mathrm{S}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,\delta(x-\xi,y-\eta)d\xi d\eta\right] \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{S}[f(\xi,\eta)\delta(x-\xi,y-\eta)]d\xi d\eta \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,\mathrm{S}[\delta(x-\xi,y-\eta)]d\xi d\eta.
\end{aligned}
$$

Remember that the **impulse response function** is defined as

$$h(x,y,\xi,\eta) \equiv \mathrm{S}[\delta(x-\xi,y-\eta)].$$

So the response of the above linear system to an arbitrarily defined input signal, $f(x,y)$, becomes

$$g(x,y) = \mathrm{S}[f(x,y)] \equiv \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\,h(x,y,\xi,\eta)d\xi d\eta.$$

Shift-Invariant Systems

Impulse Response Function

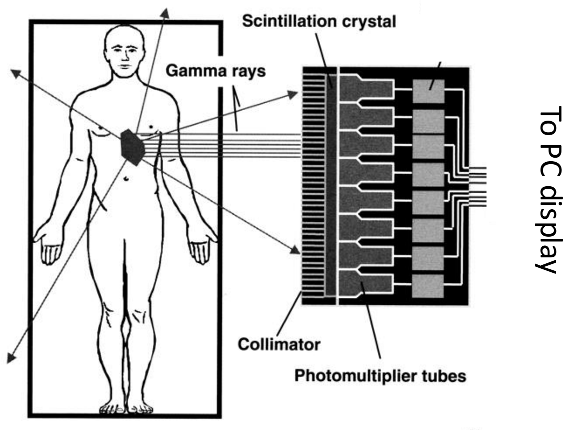
- For a 2-D problem, the impulse response is a 4-D function.

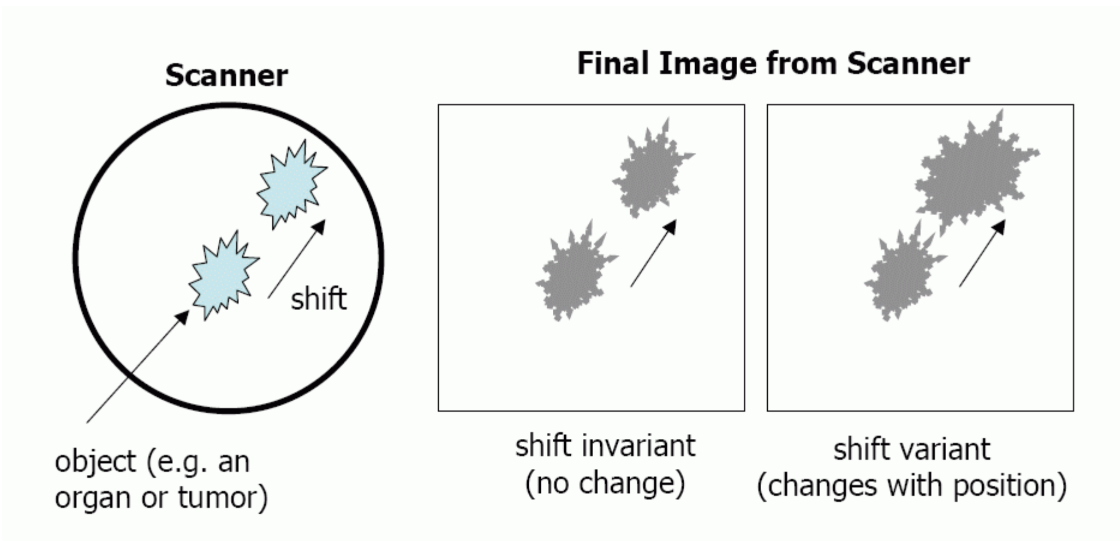
$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\, h(x,y,\xi,\eta)\, d\xi\, d\eta$$

- The computation can be greatly reduced with further simplifications …

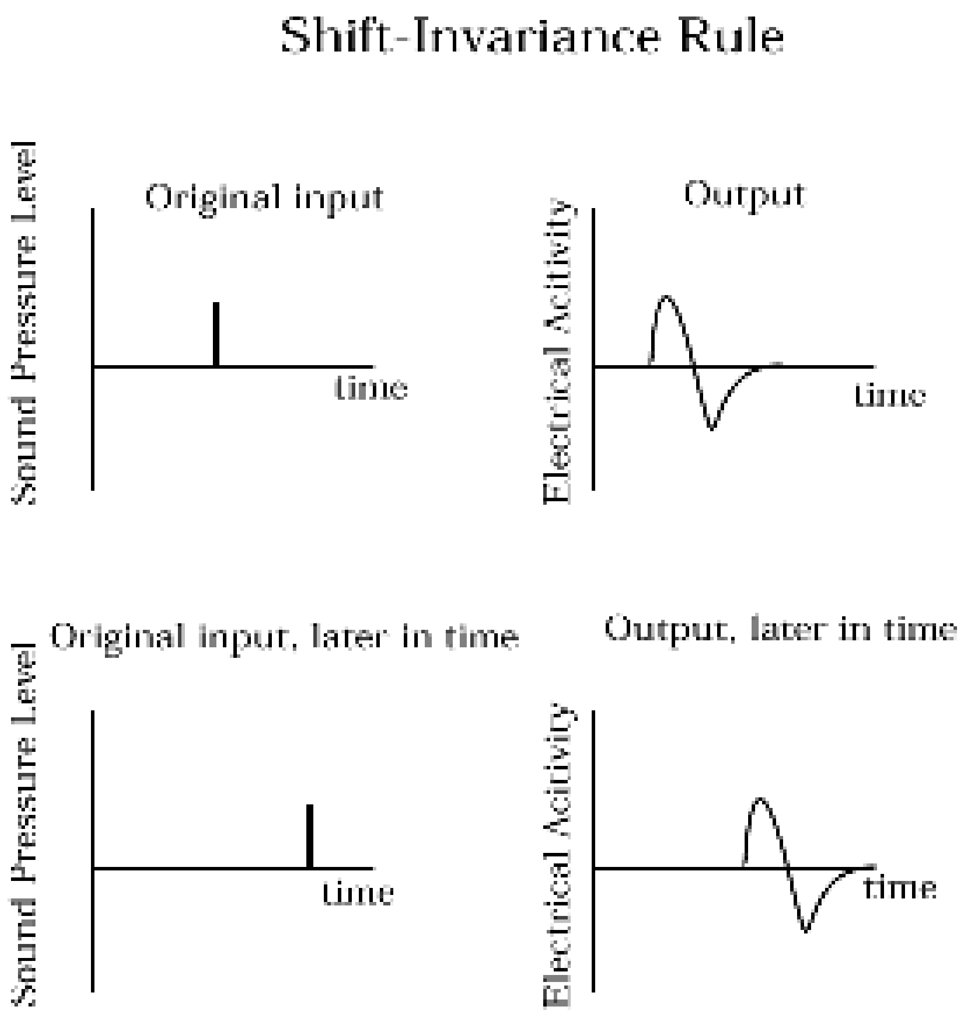
What exactly is this function again? What does it tell us about the system?

Shift Invariant Systems





Shift Invariant Systems



Shift Invariant Systems

Remember that a system is called shift-invariant if

$$g(x-\Delta x, y-\Delta y) = S\left[f(x-\Delta x, y-\Delta y)\right]$$

Note that shift-invariance does not require or imply linearity.

The impulse response function of a shift invariant system is

$$h(x, y, \xi, \eta) = S[\delta_{\xi,\eta}(x, y)] = h(x-\xi, y-\eta)$$

4-D → 2-D

Linear and Shift-Invariant (LSI) Systems

Linear and Shift-Invariant (LSI) Systems

- The response of a **linear and shift-invariant (LSI)** system to an arbitrarily given signal, $f(x,y)$, is

$$
\begin{aligned}
g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\, h(x,y,\xi,\eta)\, d\xi\, d\eta \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\, h(x-\xi,y-\eta)\, d\xi\, d\eta \\
&= f(x,y) * h(x,y),
\end{aligned}
$$

where * denotes the convolution operation, and

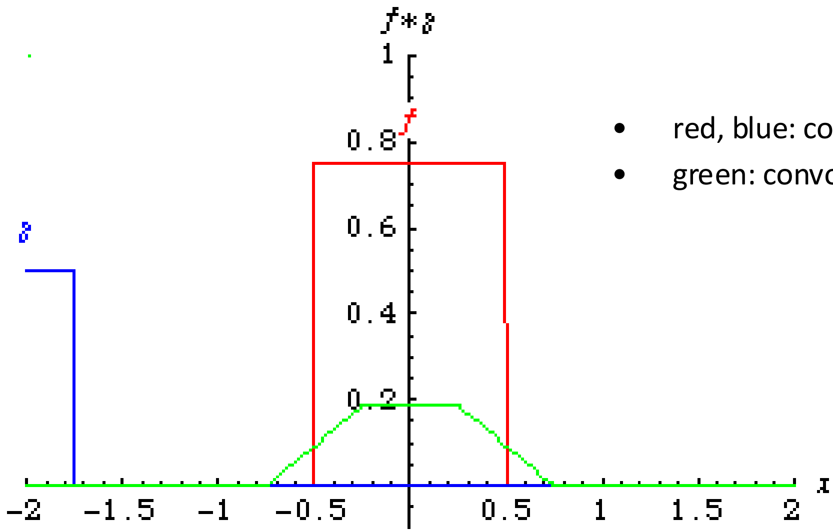
$$h(x,y) \equiv S[\delta(x,y)].$$

- The output of a linear and shift-invariant (LSI) system is the **input signal convolved with the impulse response function**.

- The impulse response function is also referred to as the **point-spread function** (PSF).

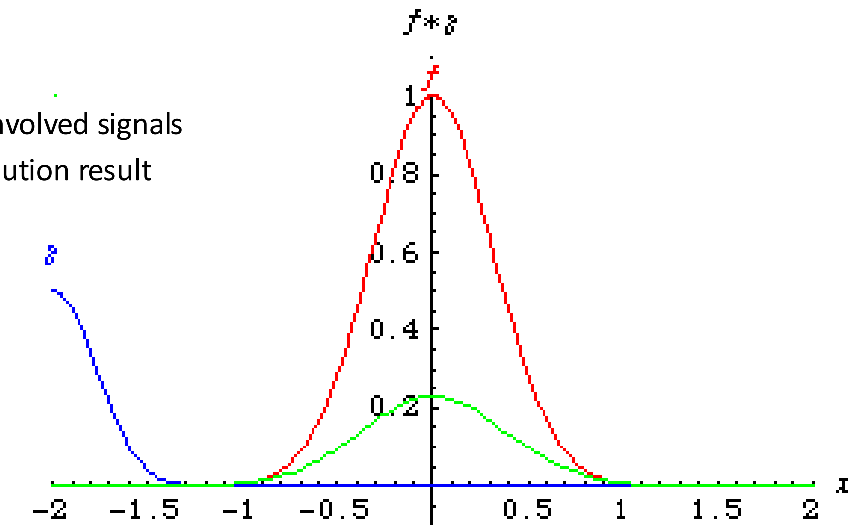
Convolution Operation in 1-D – Examples

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\xi)g(x-\xi)d\xi$$



- red, blue: convolved signals
- green: convolution result

two boxes



two Gaussians

Properties of Convolution Operation

$$g(x,y) = f(x,y) * h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)h(x-\xi,y-\eta)d\xi d\eta$$

- Commutativity

$$h_1(x,y) * h_2(x,y) = h_2(x,y) * h_1(x,y)$$

- Distributivity

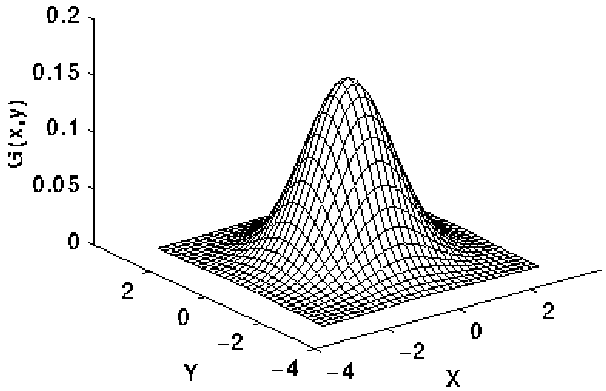
$$\left[h_1(x,y) + h_2(x,y)\right] * f(x,y) = h_1(x,y) * f(x,y) + h_2(x,y) * f(x,y)$$

Convolution Operation – Examples

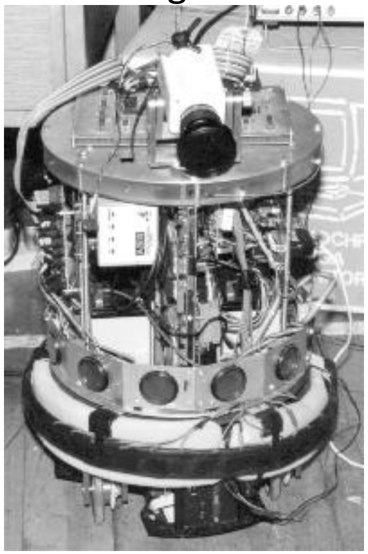
$$g(x,y) = f(x,y) * gaussian(x,y)$$

where

$$gaussian(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



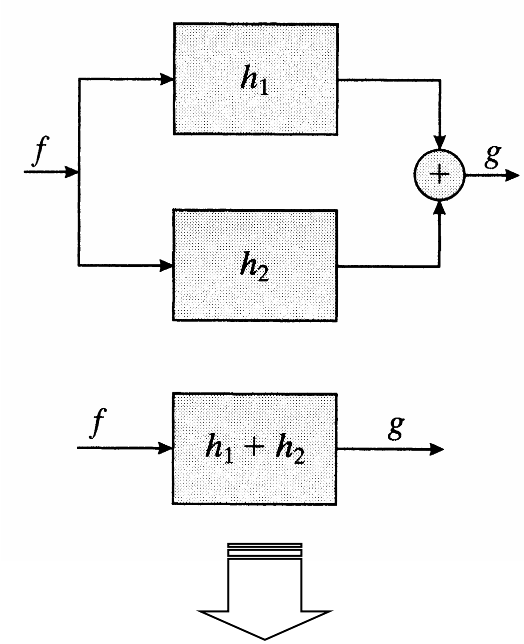
original

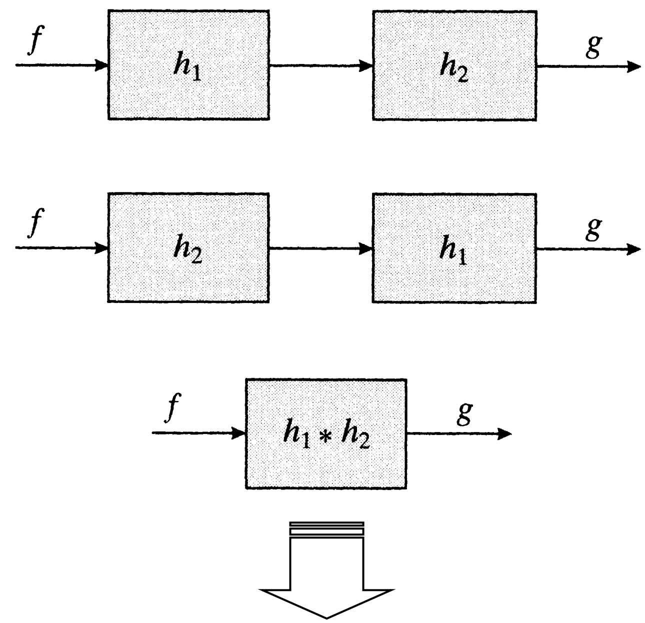


σ=1

σ=2

Connection of LSI Systems





$$g(x,y) = \left[h_1(x,y) + h_2(x,y)\right] * f(x,y)$$
$$= \left[h_2(x,y) + h_1(x,y)\right] * f(x,y)$$

$$g(x,y) = h_2(x,y) * \left[h_1(x,y) * f(x,y)\right]$$
$$= h_1(x,y) * \left[h_2(x,y) * f(x,y)\right]$$

LSI systems may be decomposed into the combination of multiple sub-systems. This may lead to a simplified mathematical representation of the complete system…

Separable Systems

A system is called separable if

$$h(x, y) = h_1(x)h_2(y)$$

in which case, the convolution between the input and the impulse response function is

Separable Systems

For a separable system, the 2-D convolution operation can be re-write as two 1-D convolution operations.

$$g(x,y) = h(x,y) * f(x,y) = h_1(x) * \left[h_2(y) * f(x,y)\right]$$

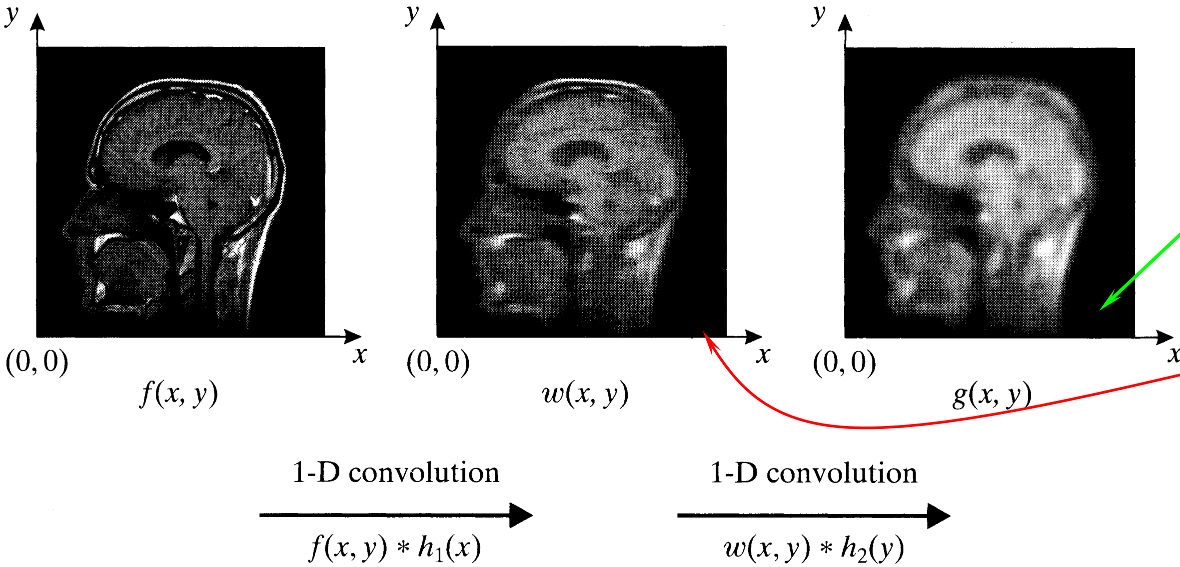
An example

$$gaussian(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}\right)\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}\right)$$

Separable Systems – An Example

An input image pass through a separable system having a impulse response function described by a 2-D Gaussian function

$$gaussian(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{x^2}{2\sigma^2}}\cdot\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{y^2}{2\sigma^2}}$$



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

$$=f(x,y)*\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$=\boxed{\left[f(x,y)*\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{x^2}{2\sigma^2}}\right]}*\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{y^2}{2\sigma^2}}$$

Fourier Transform and Sampling

Reading Material:

Chapter 2, Medical Imaging Signals and Systems, 2'nd Edition,

by Prince and Links, Prentice Hall, 2006.

Fourier Transform Basics

Fourier Transform

and

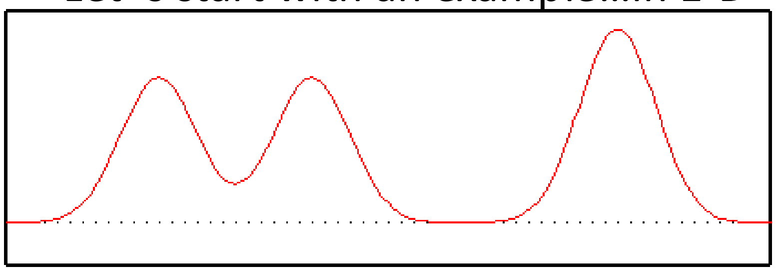
Decomposing a Periodic Signal into Sinusoidal Waves

Continuous Fourier Transform

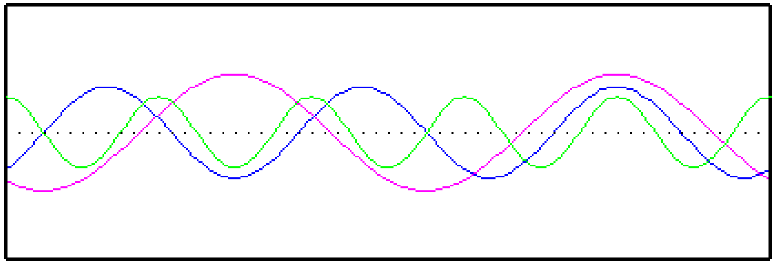
A Fourier Transform is an integral transform that re-expresses a function in terms of different sine waves of varying amplitudes, wavelengths, and phases.

So what does this mean exactly?

Let's start with an example…in 1-D

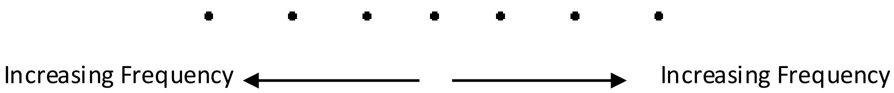


Can be represented by:



When you let these three waves interfere with each other you get your original wave function!

Since this object can be made up of 3 fundamental frequencies an ideal Fourier Transform would look something like this:



Notice that it is symmetric around the central point and that the amount of points radiating outward correspond to the distinct frequencies used in creating the image.

Fourier Transform – What and Why?

What is Fourier Transform?

- A function can be described by a summation of waves with different frequency, amplitudes and phases.

The importance of Fourier Transform in Imaging?

- Signal representations in the frequency domain provide **unique information.**
- Certain **computations** can be performed more efficiently in frequency domain.
- Certain **hardware** naturally measures signals in the frequency domain.

Continuous Fourier Transform

- For any square-integrable function f(x,y), a continuous Fourier transform is defined as

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$where\ j = \sqrt{-1}$$

- We can also define an inverse Fourier transform as

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

- Both f(x,y) and F(u,v) have infinite support.
- Both f(x,y) and F(u,v) are defined on a continuum of values.
- f(x,y) and F(u,v) must contain the same information.

Fourier Transform and Spatial Frequency

- Fourier transform can be viewed as a decomposition of the function f(x,y) into a linear combination of complex exponentials with strength F(u,v).

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

- Fourier transform provides information on the sinusoidal composition of a signal at different spatial frequencies.

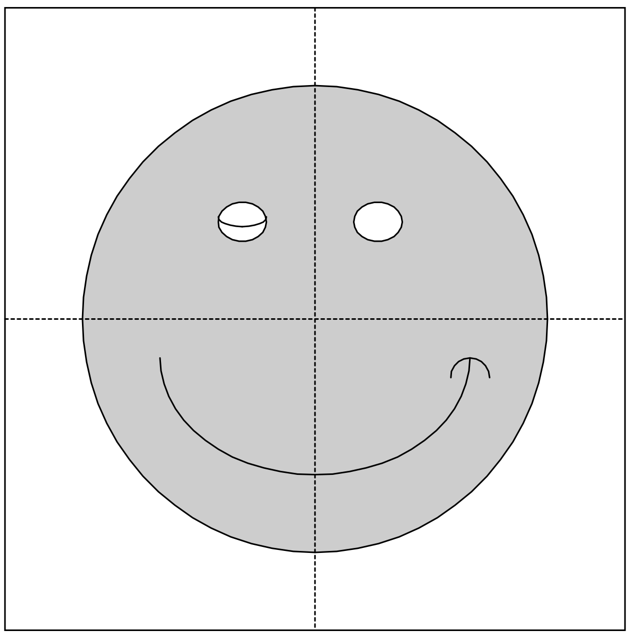
$$e^{j2\pi(ux+vy)} = \cos\left[2\pi(ux+vy)\right] + j\sin\left[2\pi(ux+vy)\right]$$

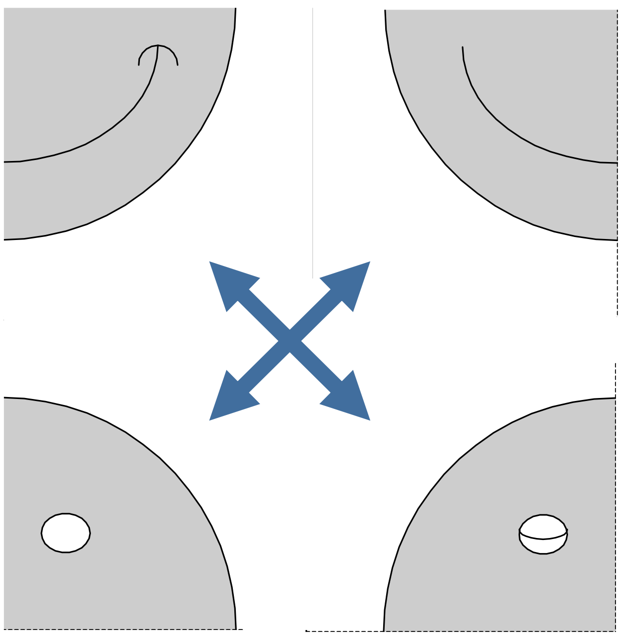
- F(u,v) is normally referred to as the spectrum of the function f(x,y).

fftshift in 2D

Use fftshift for 2D functions

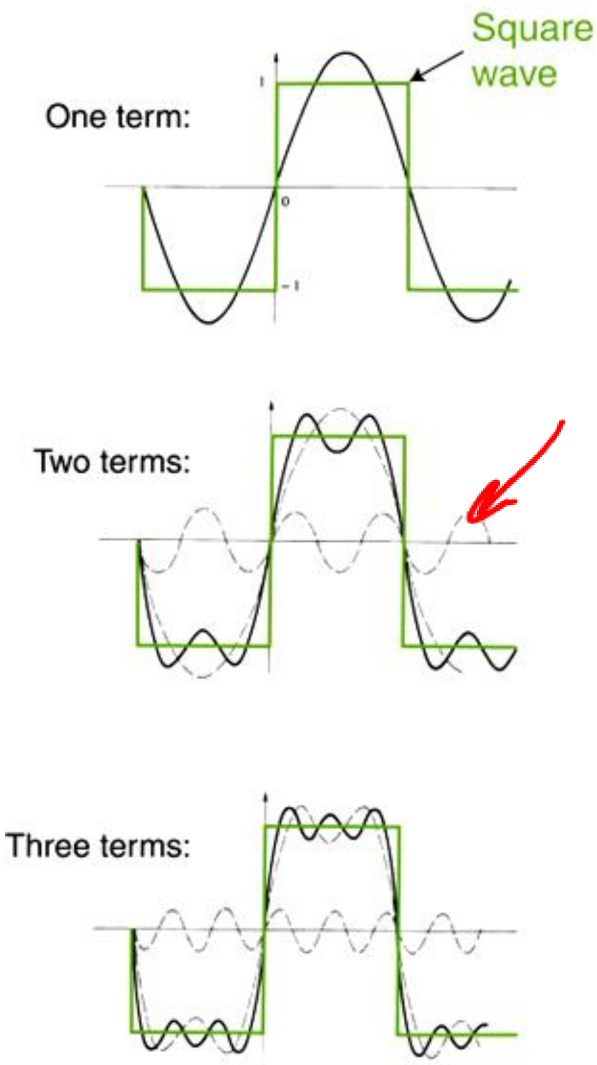
- **>>smiley2 = fftshift(smiley);**





Fourier Transform and Spatial Frequency

Fourier transform provides information on the sinusoidal composition of a signal at different spatial frequencies.

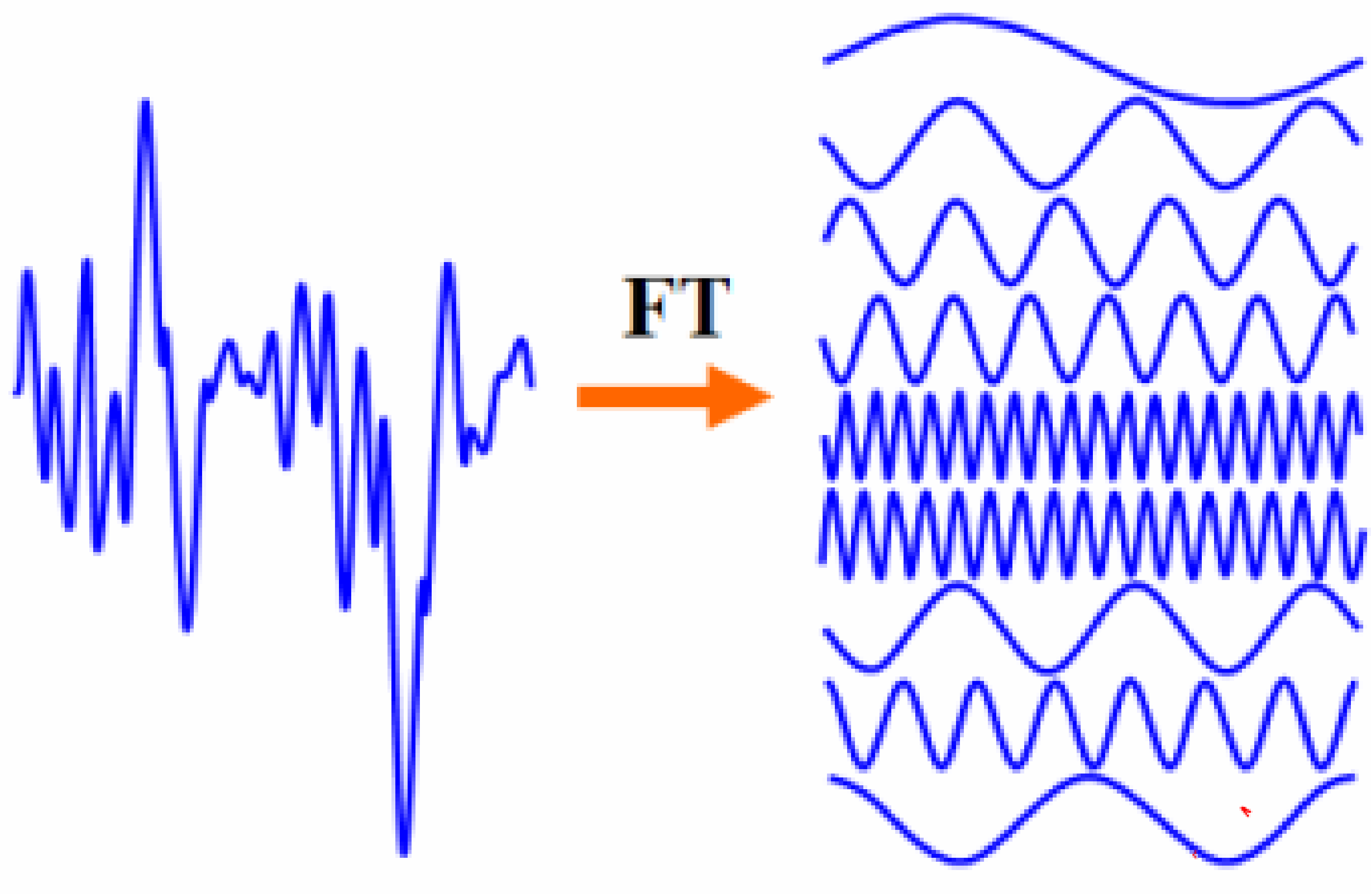


What is Spatial Frequency?

and

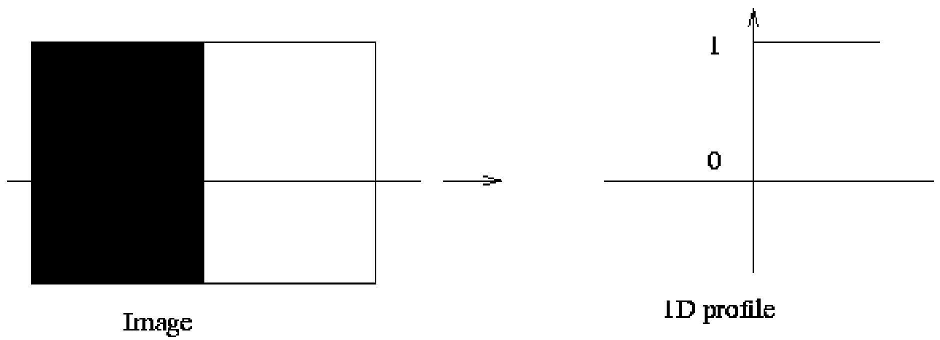
What are the so-called Higher or Lower Spatial Frequency Components?

Continuous Fourier Transform



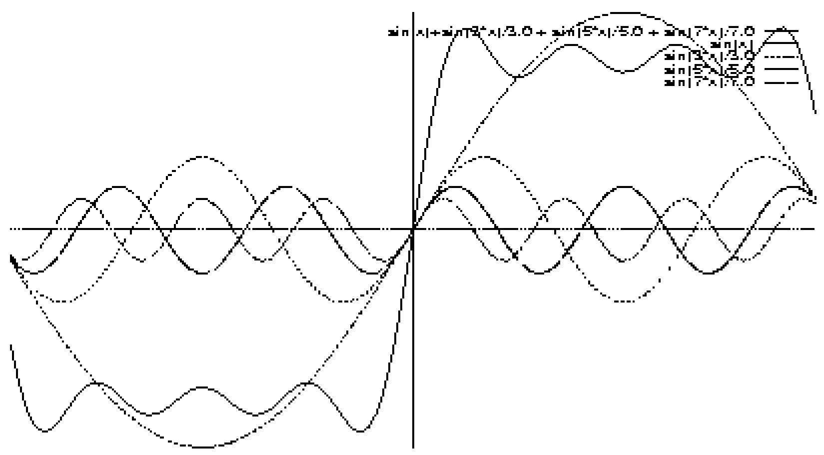
www.revisemri.com

What is Spatial Frequency?



$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{-j2\pi(ux+vy)}dudv$$

$$e^{-j2\pi(ux+vy)}$$
$$=\cos[2\pi(ux+vy)]+j\sin[2\pi(ux+vy)]$$



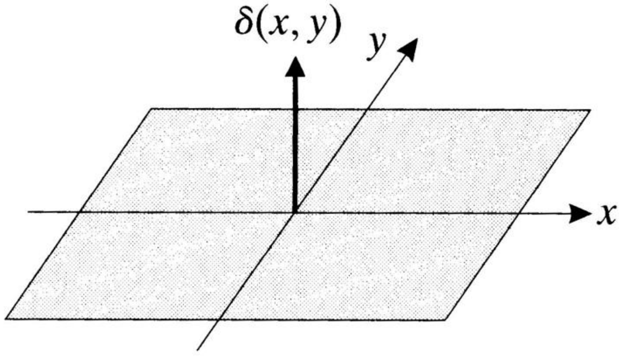
Point Impulse Signal

- A point source is mathematically represented by the delta function or Dirac function.

$$\delta(x,y)\begin{cases}\neq 0, & x=0\ and\ \ y=0\\ =0, & otherwise\end{cases}$$

and

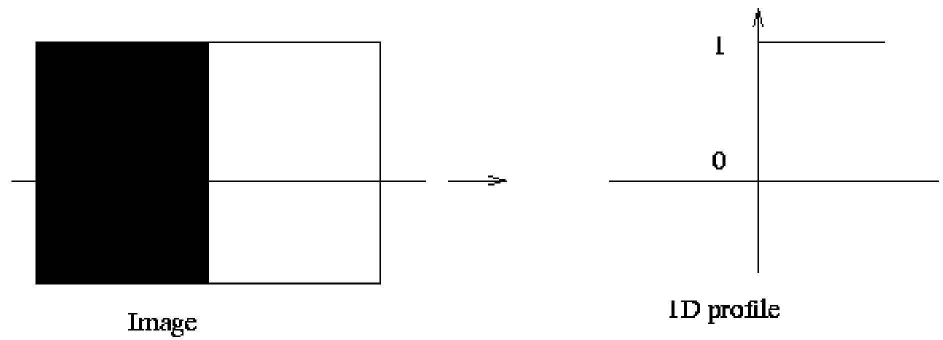
$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(x,y)dxdy=1$$



- The sampling property

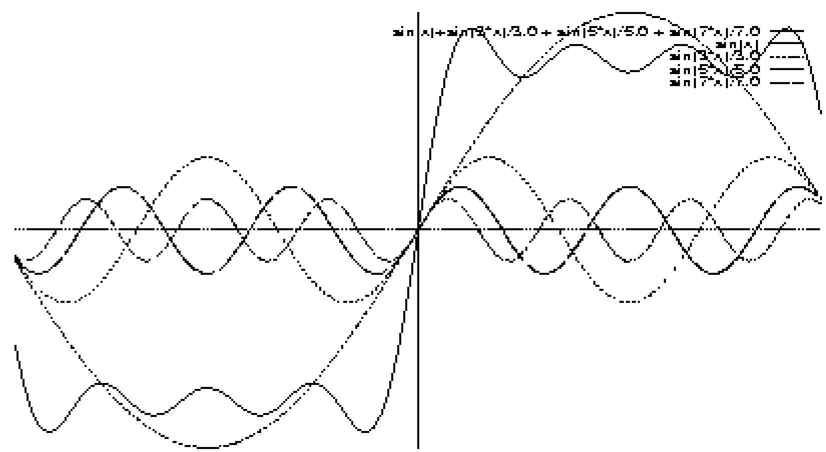
$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(\xi,\eta)\cdot\delta(x-\xi,y-\eta)d\xi d\eta$$

What is Spatial Frequency?

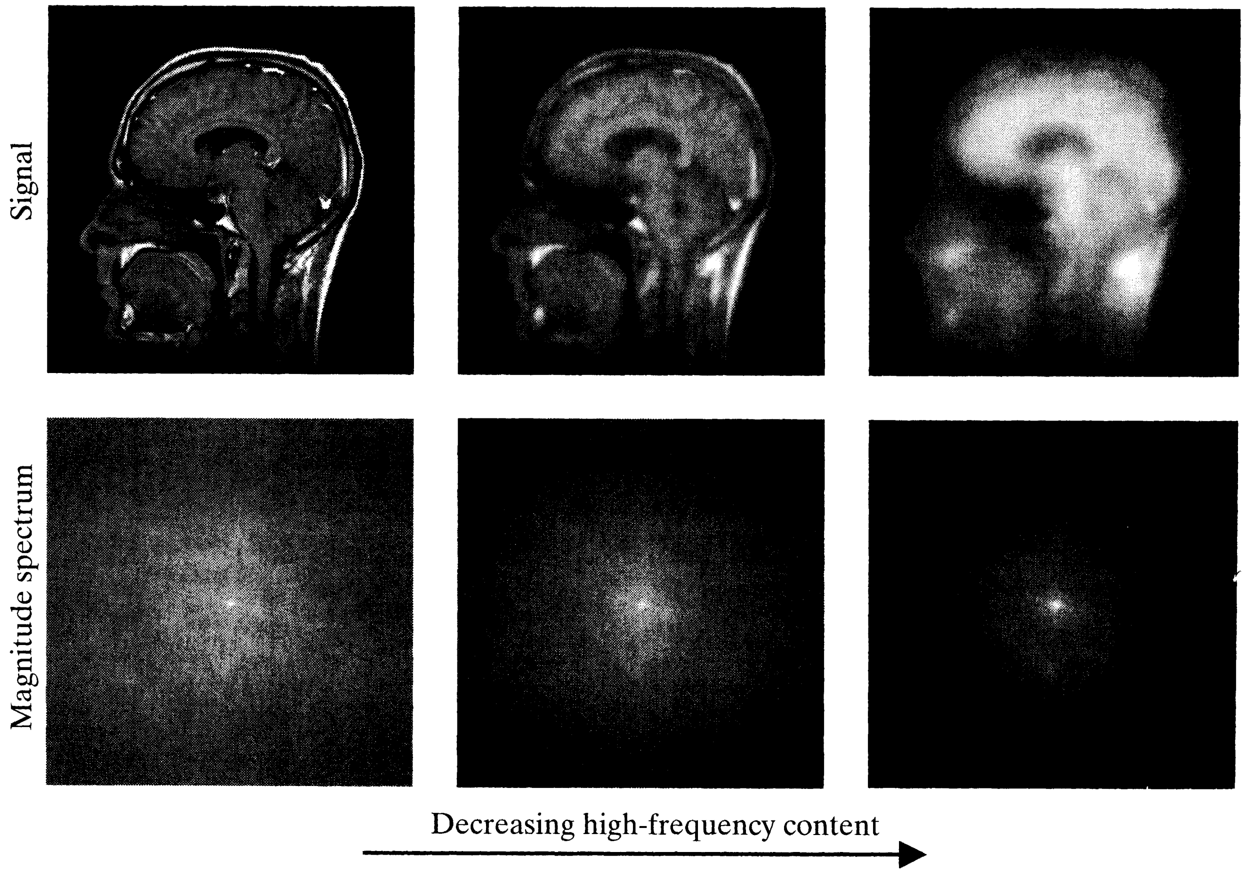


$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{-j2\pi(ux+vy)}dudv$$

$$e^{-j2\pi(ux+vy)}$$
$$=\cos[2\pi(ux+vy)]+j\sin[2\pi(ux+vy)]$$



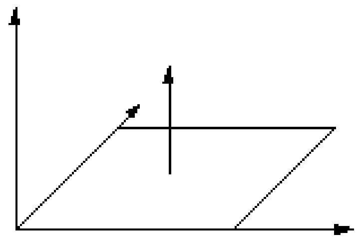
An Example



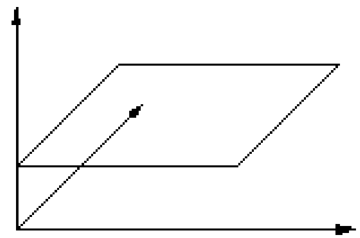
Signal
Magnitude spectrum
Decreasing high-frequency content

Examples

Delta function

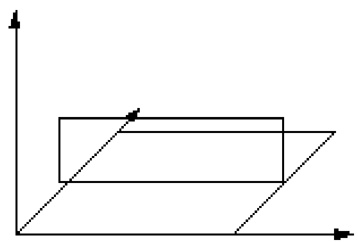


$\overset{\text{FFT}}{\rightleftharpoons}$

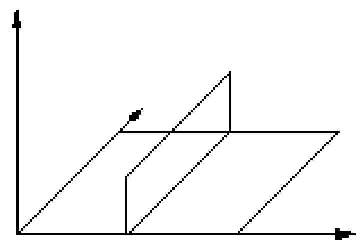


2-D DC plane

2-D line impulse

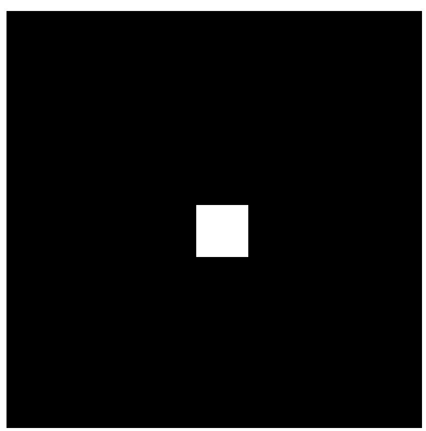


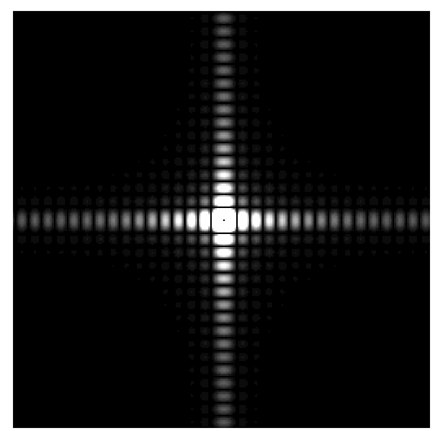
$\overset{\text{FFT}}{\rightleftharpoons}$



2-D line impulse

Square signal





2-D sinc function

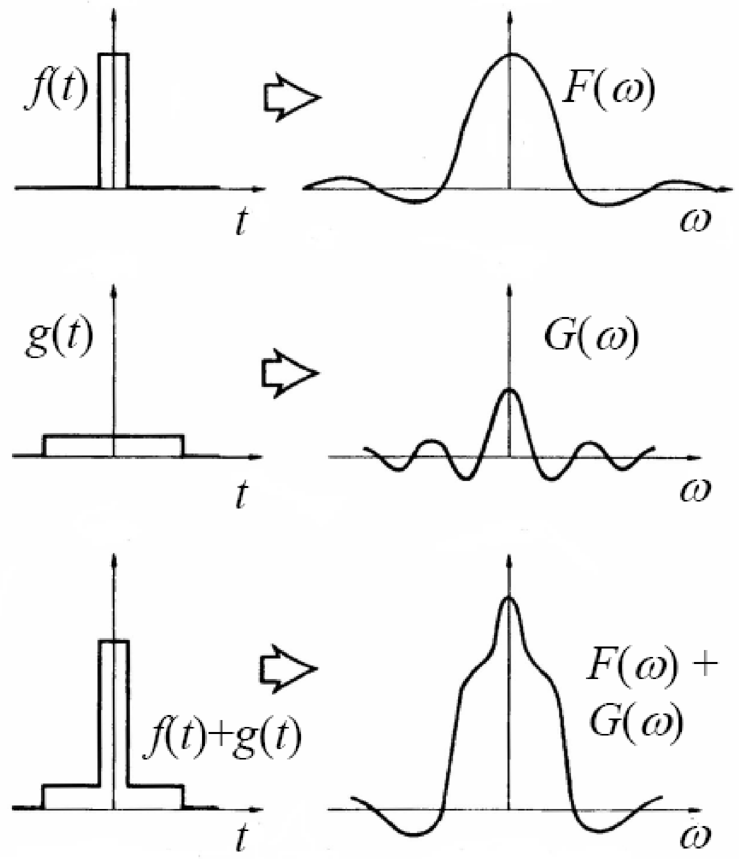
Basic Fourier Transform Pairs

Basic Fourier transform pairs

Signal	Fourier Transform
1	$\delta(u, v)$
$\delta(x, y)$	1
$\delta(x - x_0, y - y_0)$	$e^{-j2\pi(ux_0 + vy_0)}$
$\delta_s(x, y; \Delta x, \Delta y)$	$\mathrm{comb}(u\Delta x, v\Delta y)$
$e^{j2\pi(u_0 x + v_0 y)}$	$\delta(u - u_0, v - v_0)$
$\sin[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2j}[\delta(u - u_0, v - v_0) - \delta(u + u_0, v + v_0)]$
$\cos[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2}[\delta(u - u_0, v - v_0) + \delta(u + u_0, v + v_0)]$
$\mathrm{rect}(x, y)$	$\mathrm{sinc}(u, v)$
$\mathrm{sinc}(x, y)$	$\mathrm{rect}(u, v)$
$\mathrm{comb}(x, y)$	$\mathrm{comb}(u, v)$
$e^{-\pi(x^2 + y^2)}$	$e^{-\pi(u^2 + v^2)}$

Properties of Fourier Transform (1)

- Linearity $$F[a_1 f(x,y)+a_2 g(x,y)] = a_1 F[f(x,y)] + a_2 F[g(x,y)]$$



$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(ux+vy)} dxdy$$

$$where\ j = \sqrt{-1}$$

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v) e^{j2\pi(ux+vy)} dudv$$

Magnitude and Phase

- In general, Fourier transform is a complex valued signal, even if f(x,y) is real valued.
- It is sometimes useful to consider the magnitude and phase of the Fourier transform separately.

Fourier coefficients are complex: $$F(u,v) = F_R(u,v) + j \cdot F_I(u,v)$$

Magnitude: $$|F(u,v)| = \sqrt{F_R^2(u,v) + F_I^2(u,v)}$$

Phase: $$\angle F(u,v) = \tan^{-1} \frac{F_I(u,v)}{F_R(u,v)}$$

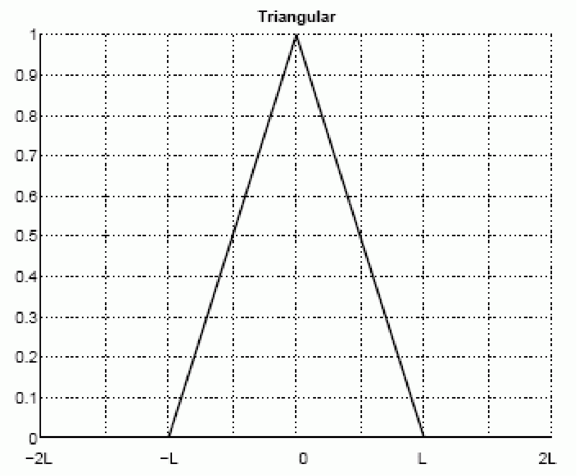
An alternative representation: $$F(u,v) = |F(u,v)| e^{j\angle F(u,v)}$$

- The square of the magnitude $|F(u,v)|^2$ is referred to as the power spectrum of the original function.

Triangular Signals and Gaussian Signals

- Triangular function:

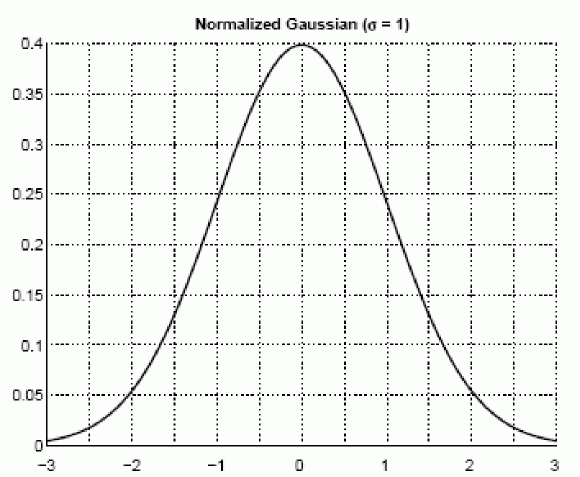
$$Tri\left(\frac{x}{2L}\right) = 1 - \frac{|x|}{L} \quad \text{for} \quad |x| < L$$
$$= 0 \quad \text{for} \quad |x| > L$$



- Normalized Gaussian function:

$$G_{1D}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

$$G_{2D}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

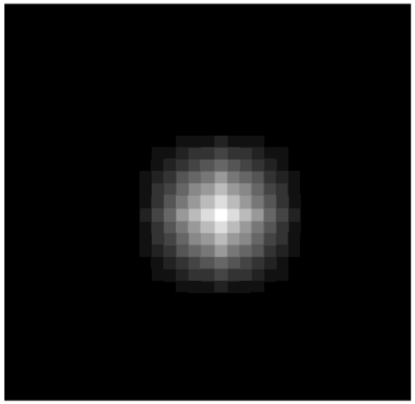


Properties of Fourier Transform (2)

Shifting Property – Shift in spatial domain is equivalent to phase change in spatial frequency domain.

$$\mathscr{F}\left[f(x-x_0, y-y_0)\right] = \mathscr{F}\left[f(u,v)\right] e^{-j2\pi(ux_0+vy_0)}$$

An example



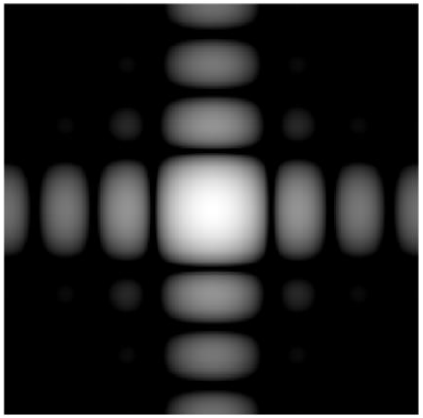
$\Lambda(x/16)\Lambda(y/16)$

real and even

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(ux+vy)} dx\,dy$$

$where\ j = \sqrt{-1}$

Properties of Fourier Transform



$$\text{Real}\{F(u,v)\} = 256\ \text{sinc}^2(16u)\text{sinc}^2(16v) \qquad Imag\{F(u,v)\} = 0$$

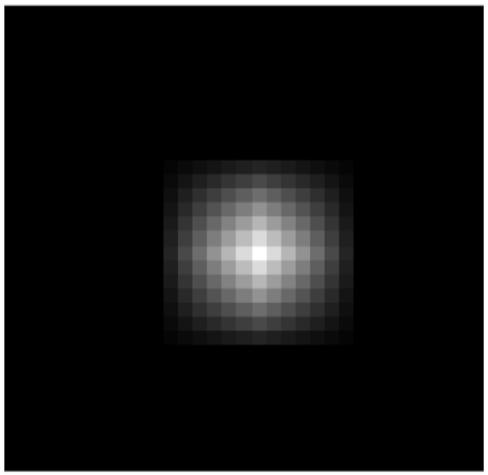
$$\left|F(u,v)\right| = \sqrt{\{Real[F(u,v)]\}^2 + \{Imag[F(u,v)]\}^2}$$

$$= 256\,\text{sinc}^2(16u)\text{sinc}^2(16v)$$

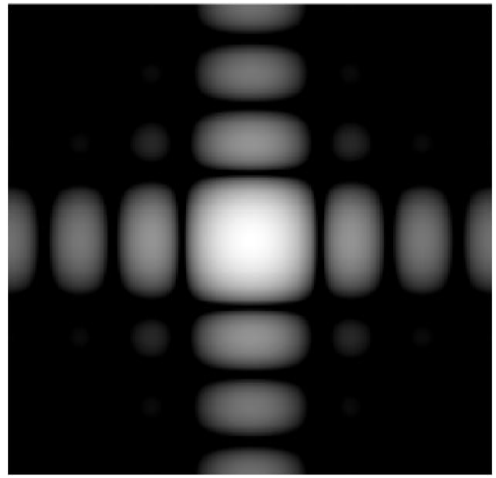
and

$$\angle F(u,v) = \tan^{-1}\frac{Imag[F(u,v)]}{Real[F(u,v)]} = 0$$

Properties of Fourier Transform



$\Lambda[(x-1)/16]\Lambda[y/16]$
shifted by 1



$|F(u,v)| =$

$256\,\mathrm{sinc}^2(16u)\mathrm{sinc}^2(16v)$

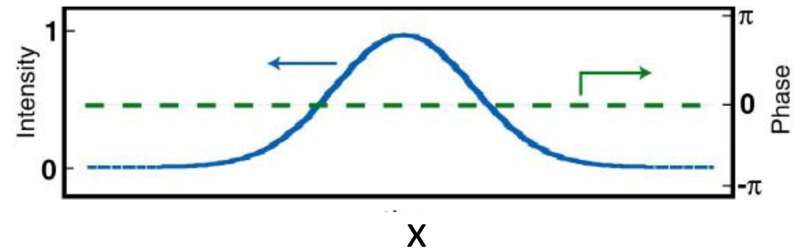
and

$\angle F(u,v) = -2\pi u$

Properties of Fourier Transform

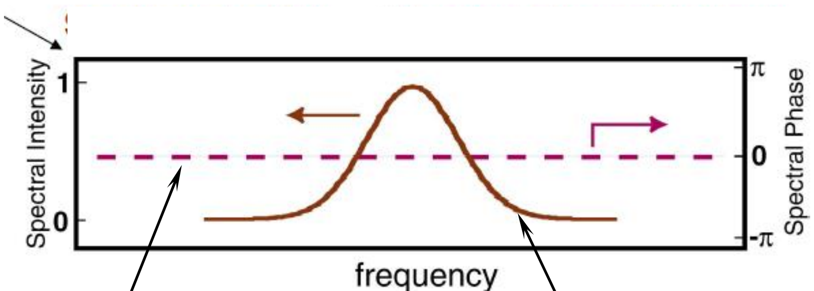
$$\text{1-D Gaussian function} : f(\text{x}) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2}$$

Spatial Domain



A Gaussian transforms to a Gaussian

Spatial Frequency Domain



$$\mathscr{F}\left[f(x-x_0, y-y_0)\right] = \mathscr{F}\left[f(u,v)\right]e^{-j2\pi(ux_0+vy_0)}$$

Spectral phase is zero

Magnitude is a Gaussian

Properties of Fourier Transform

$$F[f(x-a)] = F[f(x)]exp(-j2\pi ua)$$

Spatial Domain

$$F(u,v) = F_R(u,v) + j \cdot F_I(u,v)$$

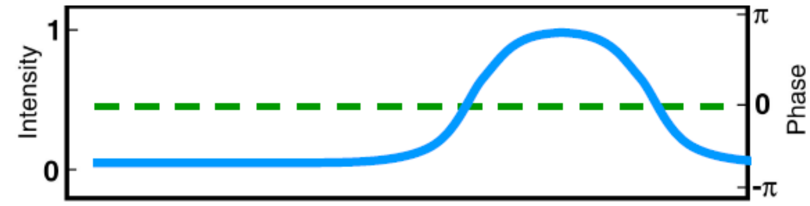
$$|F(u,v)| = \sqrt{F_R^2(u,v) + F_I^2(u,v)}$$

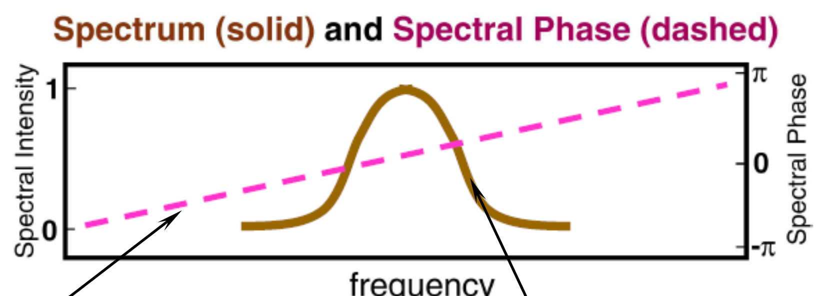
$$\angle F(u,v) = \tan^{-1}\frac{F_I(u,v)}{F_R(u,v)}$$

$$F(u,v) = |F(u,v)|e^{j\angle F(u,v)}$$

Spatial Frequency Domain

Properties





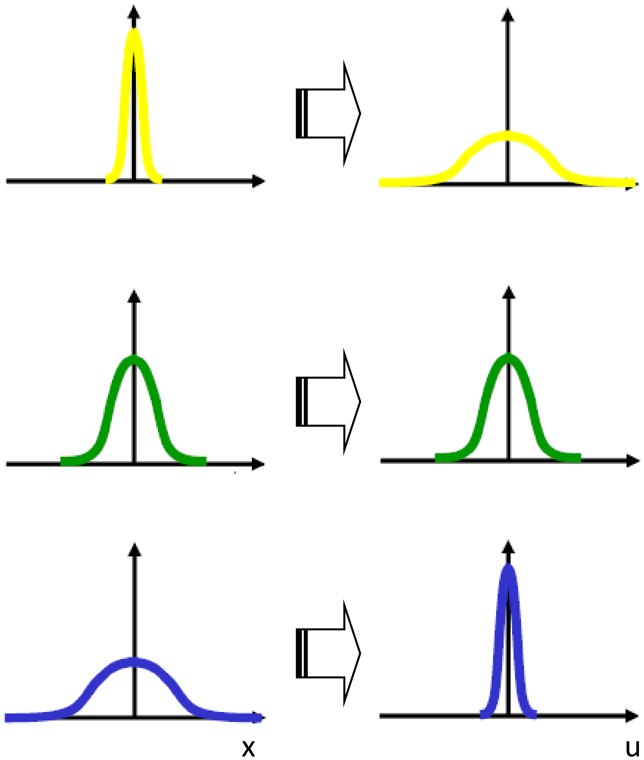
Linear shifting in spatial domain simply adds some linear phase to the pulse

Magnitude is unchanged

Properties of Fourier Transform

- Scaling $$F\left[f(ax,by)\right]=\frac{1}{ab}F\left(\frac{u}{a},\frac{v}{b}\right)$$

An 1-D example



Spatial domain, f(x)

Fourier Transform, F(u)

The shorter the pulse, the broader the spectrum !

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$where\ j=\sqrt{-1}$$

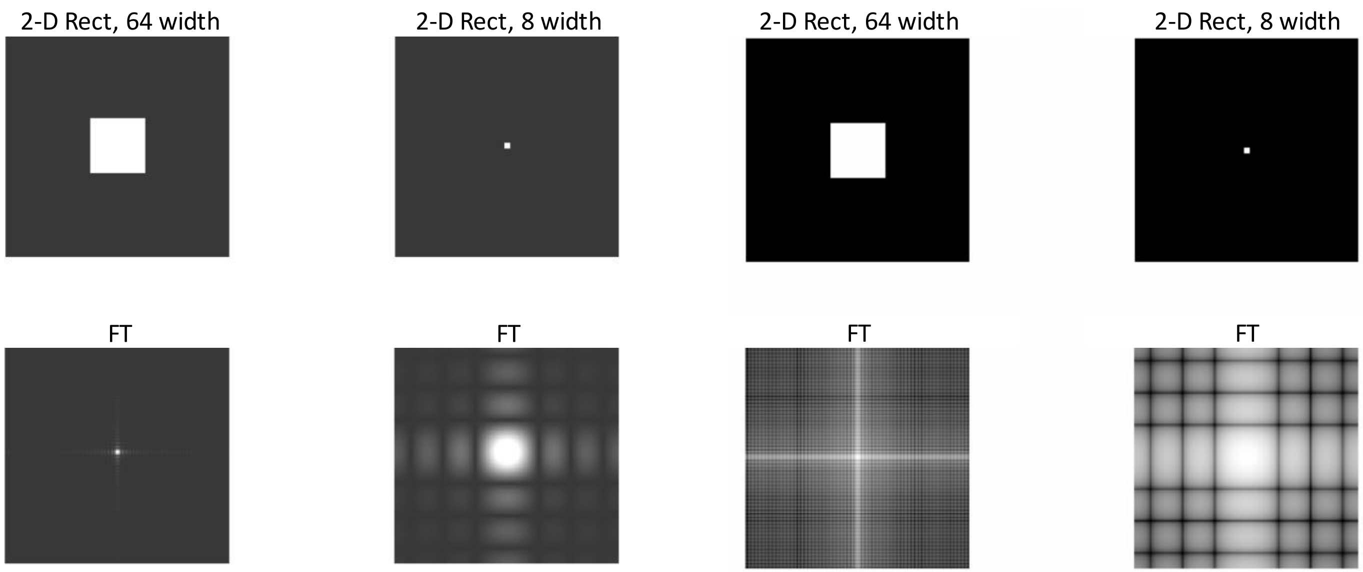
$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{j2\pi(ux+vy)}dudv$$

Properties of Fourier Transform

- Scaling

$$F\left[f(ax,by)\right]=\frac{1}{ab}F\left(\frac{u}{a},\frac{v}{b}\right)$$

A 2-D example



Linear scale Log scale

What is the Convolution Theorem of the Fourier Transform?

and

Its Implication to Modeling a System's Response to an Arbitrary Input Signal?

Linear and Shift Invariant Systems Revisited

For a shift-invariant system, the output is the input convolved with the impulse response function.

$$
\begin{aligned}
g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)h(x,y,\xi,\eta)\,d\xi d\eta \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)h(x-\xi,y-\eta)\,d\xi d\eta \\
&= f(x,y) * h(x,y)
\end{aligned}
$$

where $h(x,y,\xi,\eta)$ is the response of the system to an delta impulse signal

Convolution Theorem

- The convolution of two 2-D functions is defined as

$$f(x,y)*h(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\cdot h(x-\xi,y-\eta)\,d\xi d\eta$$

- The Fourier transform of the convolution is equal to the product of the individual Fourier transforms:

$$\mathscr{F}\left[f(x,y)*h(x,y)\right]=\mathscr{F}\left[f(x,y)\right]\cdot\mathscr{F}\left[h(x,y)\right]$$

where $\mathscr{F}\left[\cdot\right]$ is the Fourier transform operator.

The output from a **linear and shift-invariant** system is therefore

$$g(x,y)=f(x,y)*h(x,y)=\mathscr{F}^{-1}\left\{\mathscr{F}\left[f(x,y)\right]\cdot\mathscr{F}\left[h(x,y)\right]\right\}$$

Convolution Theorem

Proof:

$$\boldsymbol{F}\left[f(x,y) * g(x,y)\right]$$

Letting $x' = x - \xi$ ***and*** $y' = y - \eta$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left\{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\cdot g(x-\xi, y-\eta)\, d\xi\, d\eta\right\} e^{-j2\pi(ux+vy)}\, dx\, dy$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left\{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\cdot g(x',y')\cdot e^{-j2\pi[u(x'+\xi)+v(y'+\eta)]}\cdot dx'\, dy'\right\} d\xi\, d\eta$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left\{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\cdot e^{-j2\pi(u\xi+v\eta)}\cdot g(x',y')\cdot e^{-j2\pi(ux'+vy')}\, dx'\, dy'\right\} d\xi\, d\eta$$

$$= \left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi,\eta)\cdot e^{-j2\pi(u\xi+v\eta)}\, d\xi\, d\eta\right]\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x',y')\cdot e^{-j2\pi(ux'+vy')}\, dx'\, dy'\right]$$

$$= \boldsymbol{F}\left[f(x,y)\right]\cdot \boldsymbol{F}\left[g(x,y)\right]$$

Properties of Fourier Transform

- ***Product***

$$\mathcal{F}\left[f(x,y)\cdot g(x,y)\right]=\mathcal{F}\left[f(x,y)\right]*\mathcal{F}\left[g(x,y)\right]$$

Fourier transform of the product of two functions equals to the convolution of the Fourier transforms of individual function.

- ***The Convolution Theorem***

$$\mathcal{F}\left[f(x,y)*g(x,y)\right]=\mathcal{F}\left[f(x,y)\right]\cdot\mathcal{F}\left[g(x,y)\right]$$

$$\mathcal{F}\left[f(x,y)\cdot g(x,y)\right]=\mathcal{F}\left[f(x,y)\right]*\mathcal{F}\left[g(x,y)\right]$$

Proof:

Note that

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

The convolution theorem states:

$$\mathcal{F}\left[f(x,y)*g(x,y)\right]=\mathcal{F}\left[f(x,y)\right]\cdot\mathcal{F}\left[g(x,y)\right]$$

Similarly we can prove that:

$$\mathcal{F}^{-1}\left[f(x,y)*g(x,y)\right]=\mathcal{F}^{-1}\left[f(x,y)\right]\cdot\mathcal{F}^{-1}\left[g(x,y)\right]$$

If we define:

$$F=\mathcal{F}^{-1}\left[f(x,y)\right], G=\mathcal{F}^{-1}\left[g(x,y)\right], \quad then \quad f(x,y)=\mathcal{F}\left[F\right], g(x,y)=\mathcal{F}\left[G\right]$$

We can see that

$$\mathcal{F}^{-1}\left\{\mathcal{F}\left[F\right]*\mathcal{F}\left[G\right]\right\}=F\cdot G$$

Therefore

$$\mathcal{F}\left[F\cdot G\right]=\mathcal{F}\left[F\right]*\mathcal{F}\left[G\right]$$

Note that F and G are arbitrary function, so that

we can re-write the above equation as

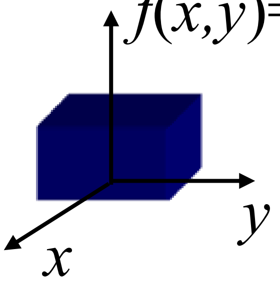
$$\mathcal{F}\left[f(x,y)\cdot g(x,y)\right]=\mathcal{F}\left[f(x,y)\right]*\mathcal{F}\left[g(x,y)\right]$$

Properties of Fourier Transform

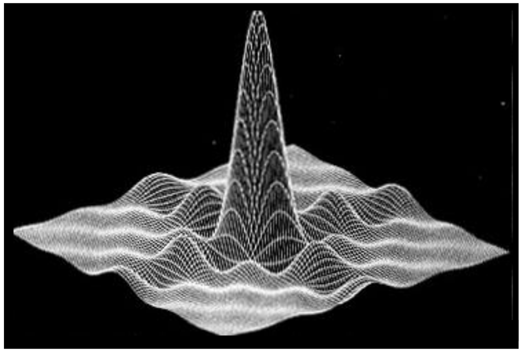
- Separable product

$$\begin{aligned} &if \\ &f(x,y) = f_1(x)\cdot f_2(y) \\ &then \\ &\boldsymbol{F}_{u,v}\left[f(x,y)\right] = \boldsymbol{F}_u\left[f_1(x)\right]\cdot \boldsymbol{F}_v\left[f_2(y)\right] \end{aligned}$$

$f(x,y)$=Rect(x)Rect(y)



$\mathscr{F}\{f(x,y)\}$=sinc(u)sinc(v)



Symmetry Properties

Expanding the Fourier transform of a function, f(t):

$$F(\omega) = \int_{-\infty}^{\infty} [\mathrm{Re}\{f(t)\} + i\,\mathrm{Im}\{f(t)\}]\,[\cos(\omega t) - i\sin(\omega t)]\,dt$$

Expanding more, noting that: $\int_{-\infty}^{\infty} O(t)\,dt = 0$ if O(t) is an odd function

= 0 if Re{f(t)} is odd   = 0 if Im{f(t)} is even

$$F(\omega) = \int_{-\infty}^{\infty} \mathrm{Re}\{f(t)\}\cos(\omega t)\,dt + \int_{-\infty}^{\infty} \mathrm{Im}\{f(t)\}\sin(\omega t)\,dt \quad \leftarrow \mathbf{Re}\{F(\omega)\}$$

= 0 if Im{f(t)} is odd   = 0 if Re{f(t)} is even

$$+ \; i\int_{-\infty}^{\infty} \mathrm{Im}\{f(t)\}\cos(\omega t)\,dt - i\int_{-\infty}^{\infty} \mathrm{Re}\{f(t)\}\sin(\omega t)\,dt \quad \leftarrow \mathbf{Im}\{F(\omega)\}$$

Even functions of w   Odd functions of w

Fourier transform of Circularly Symmetric Functions (1)

A function is circularly symmetric if

$$f_\theta(x,y) = f(x,y), \quad \text{for every } \theta,$$
$$\text{where } f_\theta(x,y) \text{ is a rotated version of } f(x,y).$$

In this case, the function

$$f(x,y) = f(r), \text{ where } r = \sqrt{x^2 + y^2}$$

The Fourier transform is also real and circularly symmetric.

$$\left|F(u,v)\right| = F(u,v) \text{ and } \angle F(u,v) = 0$$
$$\textit{and}$$
$$F(u,v) = F(q), \text{ where } q = \sqrt{u^2 + v^2}$$

Fourier Transform of Circularly Symmetric Functions (2)

The Fourier transform of a circularly symmetric function is called Hankel transform

$$F(q) = 2\pi \int_{-\infty}^{\infty} f(r) J_0(2\pi qr) r dr$$

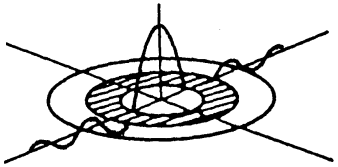
where $J_n(r)$ is the zero-order Bessel function,

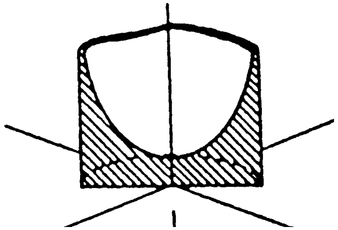
$$J_n(r) = \frac{1}{\pi} \int_0^{\pi} \cos(nr - r\sin\phi) d\phi, \quad n = 0,1,2,...,$$

The inverse Hankel transform is given by

$$f(r) = 2\pi \int_{-\infty}^{\infty} F(q) J_0(2\pi qr) q dq$$

$$\frac{\sin(2\pi ar)}{r}$$





$$\frac{\Pi(q/2a)}{\left(a^2 - q^2\right)^{\frac{1}{2}}}$$

Properties of Fourier Transform (4)

- Parseval's Theorem

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left|f(x,y)\right|^2 dudv = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left|F(u,v)\right|^2 dudv$$

Total energy of a signal of a signal f(x,y) in spatial domain equals its total energy in spatial frequency domain.

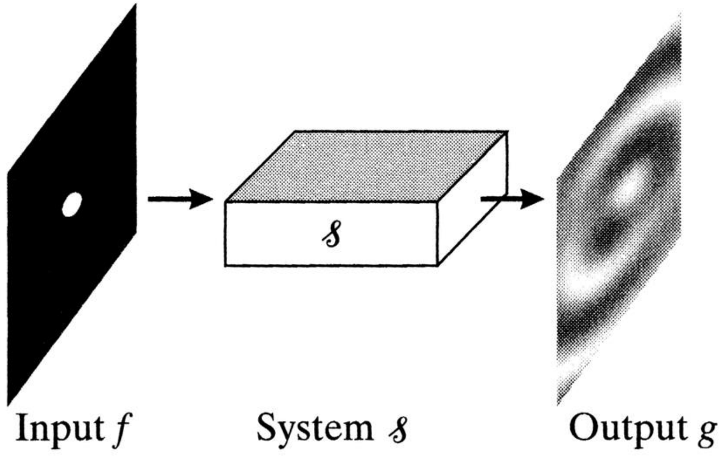
Fourier transform and its inverse is energy preserving!

Convolution Theorem Revisited

The convolution theorem enables one to perform convolution operation as multiplication process in spatial frequency domain.

$$f(x,y) * h(x,y) = \boldsymbol{F}^{-1}\left\{\boldsymbol{F}\left[f(x,y)\right]\cdot \boldsymbol{F}\left[h(x,y)\right]\right\}$$

By using the Fast Fourier Transform (FFT) algorithms, the convolution operation can be performed very efficiently !! This provide a practical way for modeling linear shift-invariant systems …



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

System Transfer Function

The Fourier transform of the impulse response function h is called the system transfer function.

$$H(u,v) = \boldsymbol{\mathcal{F}}\left[h(x,y)\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)e^{-2\pi j(ux+vy)}\,dx\,dy$$

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

System Transfer Function

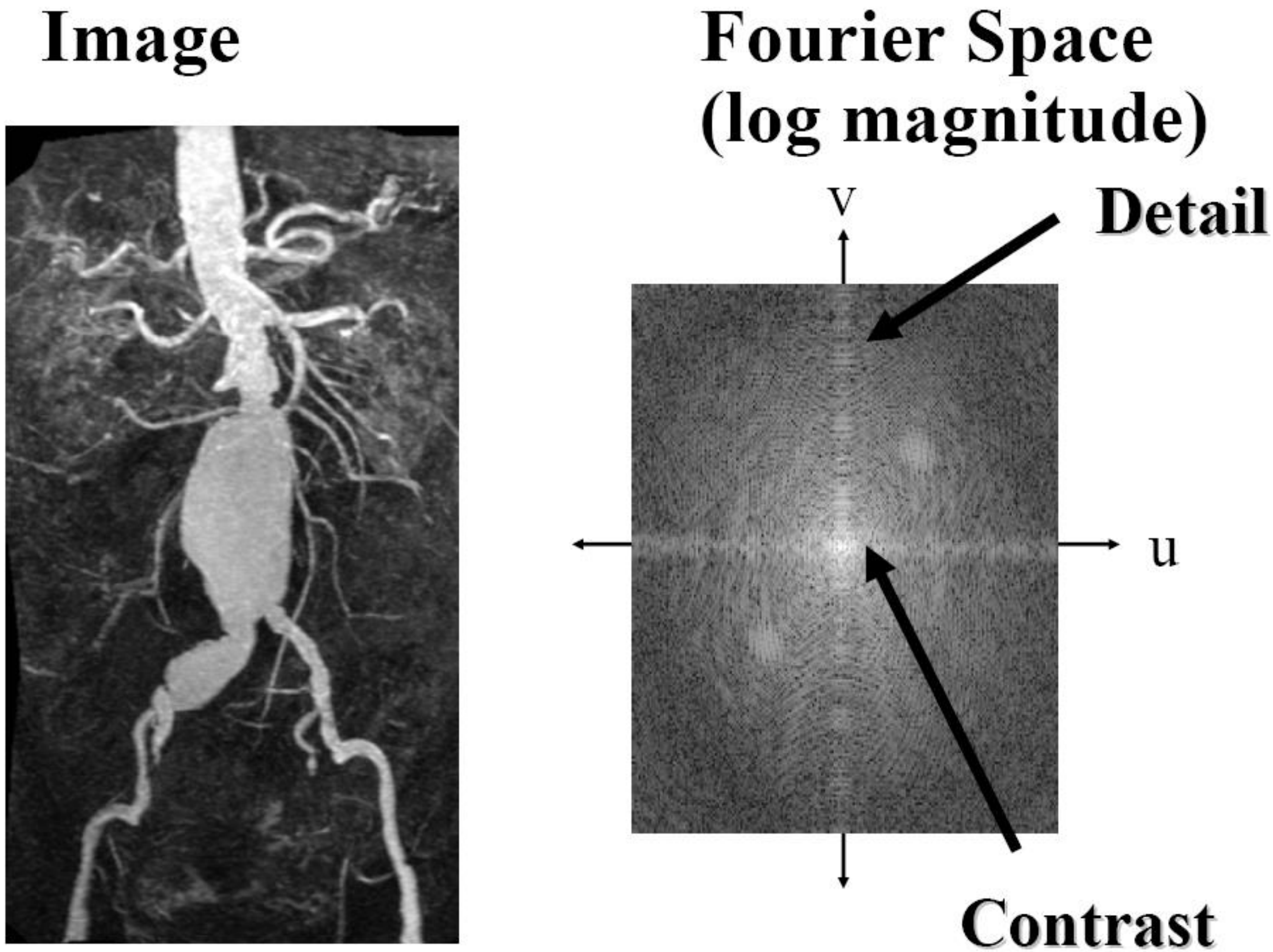
$$G(u,v) = F(u,v)H(u,v)$$

An ideal low-pass filter is defined as

$$H(u,v) = \begin{cases} 1 & \text{for } \sqrt{\mathrm{u}^2 + \mathrm{v}^2} \leq c \\ 0 & \text{for } \sqrt{\mathrm{u}^2 + \mathrm{v}^2} > c \end{cases}$$

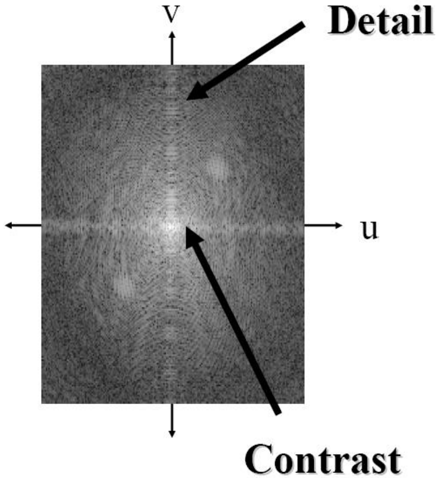
c is called the cut-off frequency.

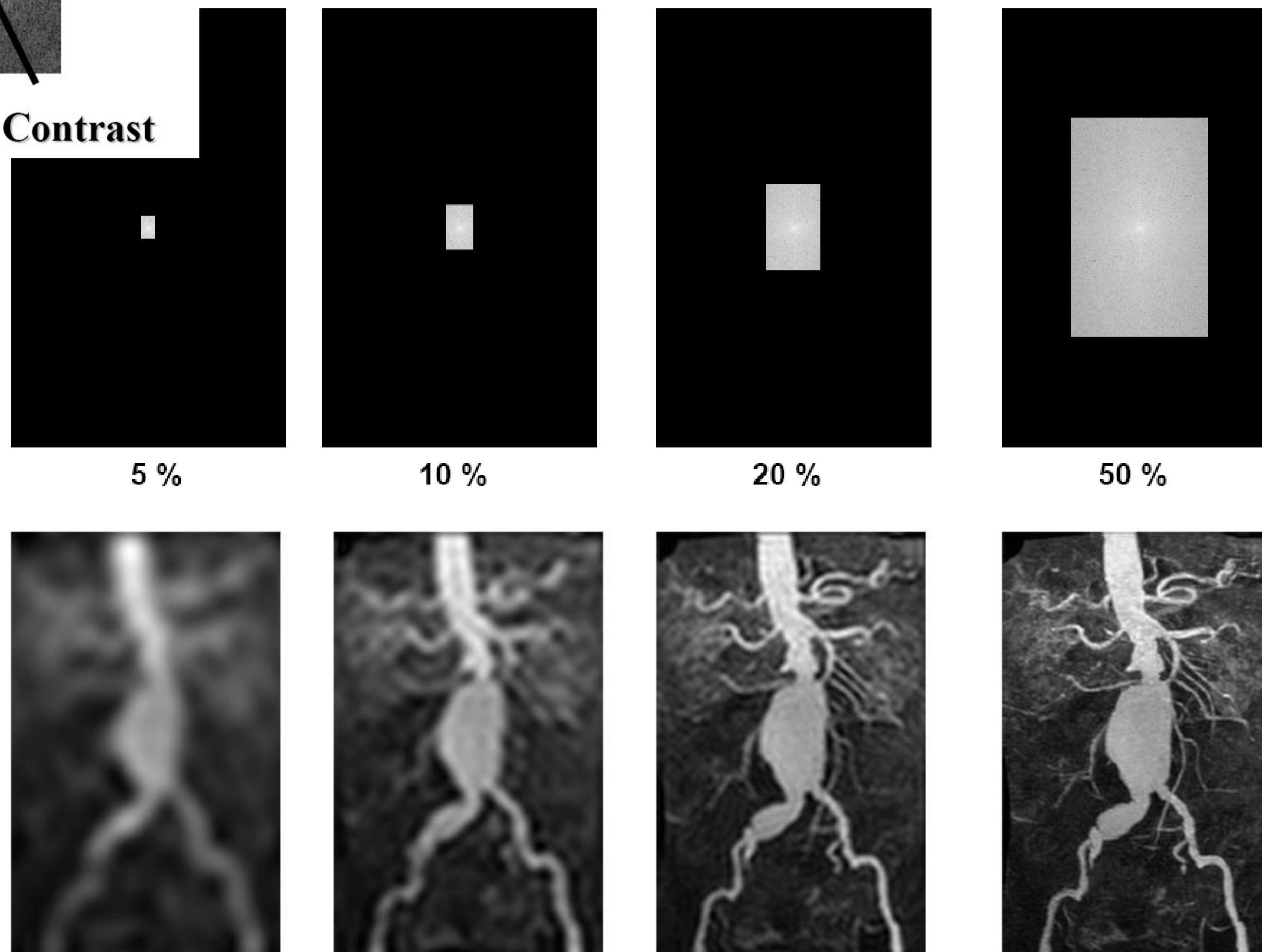
Spatial Frequency Revisited



Image

Fourier Space (log magnitude)





System Transfer Function

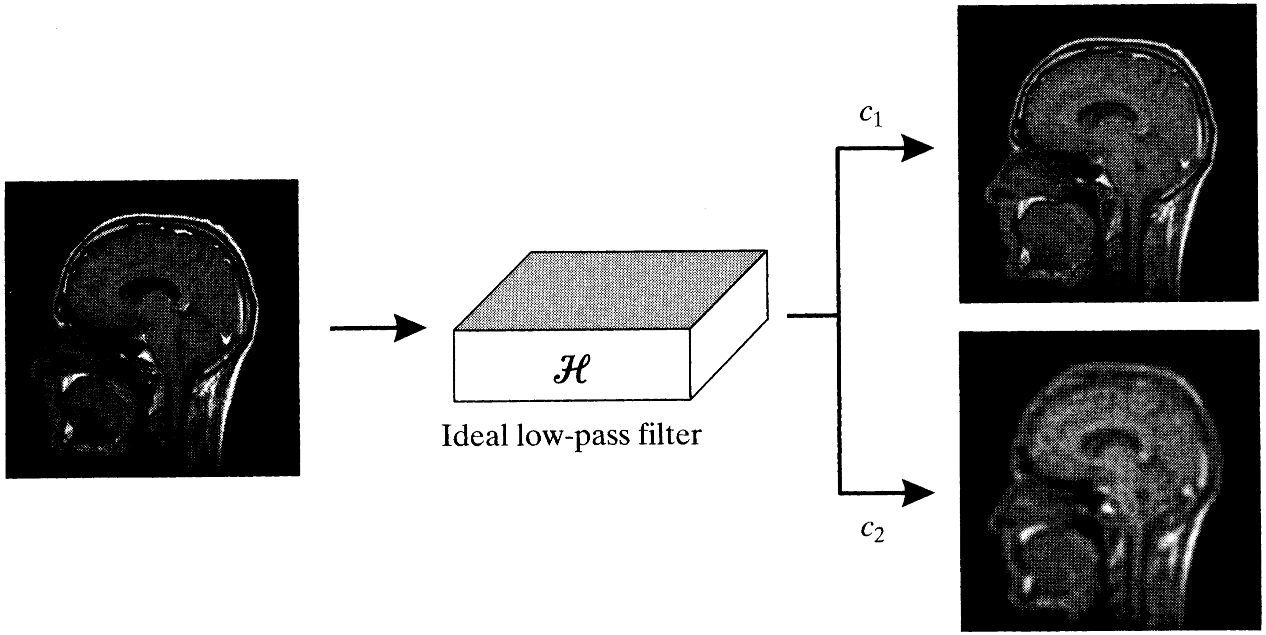


Figure 2.12
The response of an ideal low-pass filter for two values of the cutoff frequency c ($c_1 > c_2$).

Sampling

What is the Sampling?

and

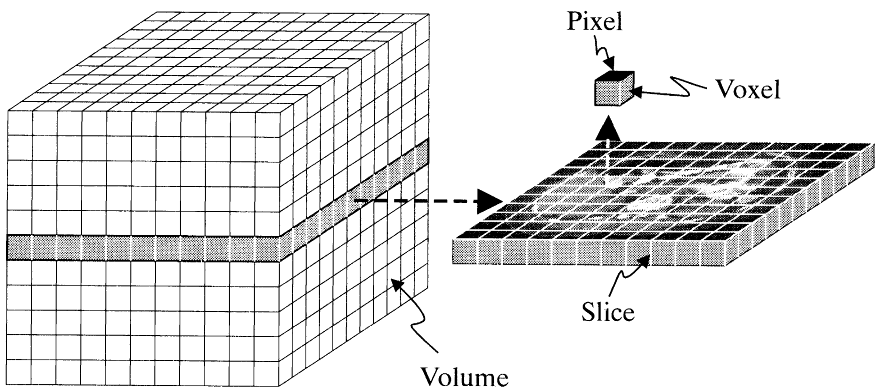
How to Sample a Continuous Signal into Discrete without Losing Information?

What and Why?

Transformation from continuous signals into discrete signals is called ***sampling***. The sampled data can then be processed by digital hardware.

$$f(x,y) \Rightarrow f(m\Delta x, n\Delta y), \text{ for m, n} = 0,1,\ldots$$

where $\Delta x, \Delta y$ are called sampling intervals or sampling periods.



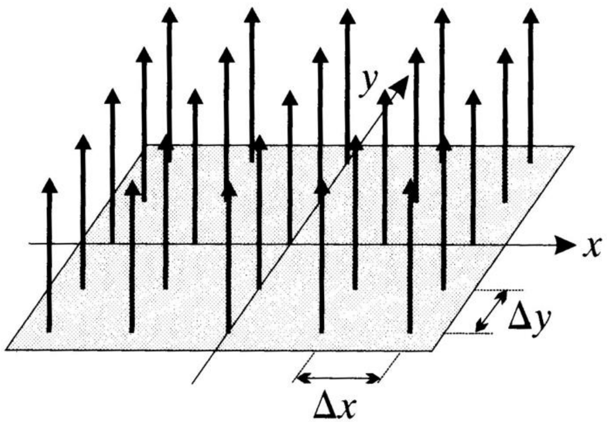
Comb and Sampling Function

- The 2-D sampling function

$$\delta_s(x,y,\Delta x,\Delta y)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(x-m\Delta x,y-n\Delta y)$$

where Δx and Δy are the sampling intervals

$$\delta_s(x,y,\Delta x,\Delta y)=\frac{1}{\Delta x\Delta y}comb\left(\frac{x}{\Delta x},\frac{y}{\Delta y}\right)$$

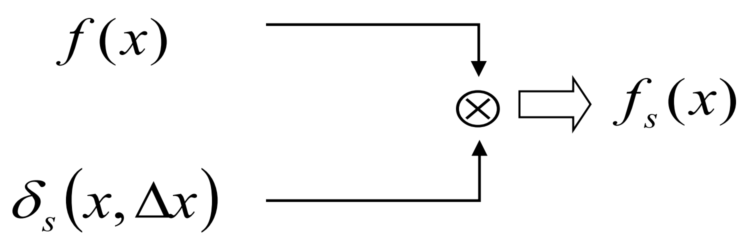


$$comb(x,y)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(x-m,y-n)$$

$$\delta(ax,by)=\frac{1}{|ab|}\delta(x,y)$$

Sampling in 1-D

Model

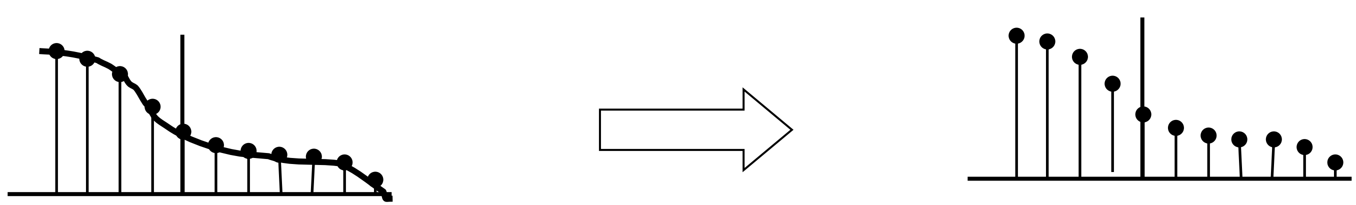


The 1-D sampling function

$$\delta_s(x, \Delta x) = \sum \delta(x - n \cdot \Delta x)$$

The sampled function

$$f_s(\mathrm{x}) = f(\mathrm{x}) \cdot \delta_s(x, \Delta x) = \sum f(n\Delta x)\delta(x - n\Delta x)$$



Basic Fourier Transform Pairs

Basic Fourier transform pairs

Signal	Fourier Transform
1	$\delta(u, v)$
$\delta(x, y)$	1
$\delta(x - x_0, y - y_0)$	$e^{-j2\pi(ux_0 + vy_0)}$
$\delta_s(x, y; \Delta x, \Delta y)$	$\mathrm{comb}(u\Delta x, v\Delta y)$
$e^{j2\pi(u_0x + v_0y)}$	$\delta(u - u_0, v - v_0)$
$\sin[2\pi(u_0x + v_0y)]$	$\frac{1}{2j}[\delta(u - u_0, v - v_0) - \delta(u + u_0, v + v_0)]$
$\cos[2\pi(u_0x + v_0y)]$	$\frac{1}{2}[\delta(u - u_0, v - v_0) + \delta(u + u_0, v + v_0)]$
$\mathrm{rect}(x, y)$	$\mathrm{sinc}(u, v)$
$\mathrm{sinc}(x, y)$	$\mathrm{rect}(u, v)$
$\mathrm{comb}(x, y)$	$\mathrm{comb}(u, v)$
$e^{-\pi(x^2 + y^2)}$	$e^{-\pi(u^2 + v^2)}$

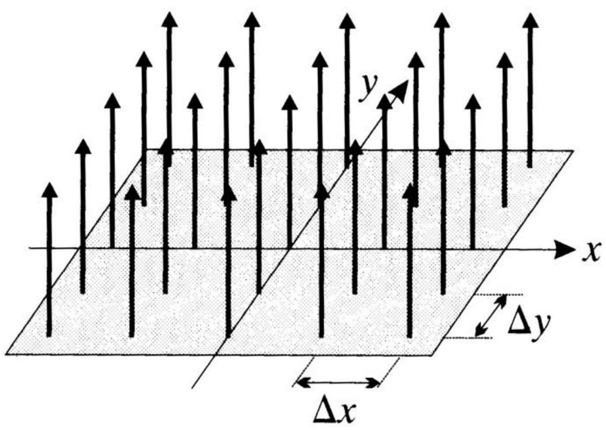
Comb and Sampling Function

- The 2-D sampling function

$$\delta_s(x, y, \Delta x, \Delta y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

where Δx and Δy are the sampling intervals

$$\delta_s(x, y, \Delta x, \Delta y) = \frac{1}{\Delta x \Delta y} comb\left(\frac{x}{\Delta x}, \frac{y}{\Delta y}\right)$$



$$comb(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m, y - n)$$

$$\delta(ax, by) = \frac{1}{|ab|} \delta(x, y)$$

Comb and Sampling Function

- The 2-D comb function

$$comb(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m, y - n)$$

Fourier Transform of Sampled Function

$$\mathrm{F}_{f_s}(u) = \mathsf{F}\left[f_s(x)\right] = \mathsf{F}\left[\delta_s(x,\Delta x)\cdot f(x)\right]$$

$$= comb(u\cdot \Delta x) * \mathsf{F}\left[f(x)\right]$$

$$= \left\{\sum_{n=-\infty}^{n=\infty} \delta\left[\Delta x\left(u - \frac{n}{\Delta x}\right)\right]\right\} * \mathrm{F}(u)$$

$$= \frac{1}{\Delta x}\sum_{n=-\infty}^{n=\infty}\left\{\delta\left(u - \frac{n}{\Delta x}\right) * \mathrm{F}(u)\right\}$$

$$= \frac{1}{\Delta x}\sum_{n=-\infty}^{n=\infty} \mathrm{F}\left(u - \frac{n}{\Delta x}\right)$$

$$comb(x) = \sum_{m=-\infty}^{\infty} \delta(x-m)$$

$$\delta_S(x,\Delta x) = \sum_{m=-\infty}^{\infty} \delta(x - m\Delta x)$$

$$\mathfrak{F}\left[\delta_s(x,\Delta x)\right] = comb(u\cdot \Delta x)$$

Multiplication in the spatial domain becomes convolution in the spatial frequency domain

Point Impulse Signal

- The sampling property

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x-\xi,y-\eta)dxdy = f(\xi,\eta)$$

- The scaling property

$$\delta(ax,by) = \frac{1}{|ab|}\delta(x,y)$$

$$\delta(x,y)\begin{cases} \neq 0, & x=0 \; and \; y=0 \\ =0, & otherwise \end{cases}$$

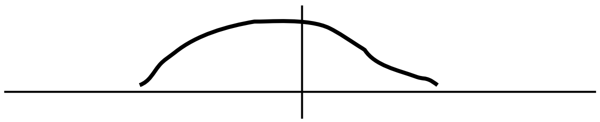
and

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(x,y)dxdy = 1$$

Fourier Transform of Sampled Function

$$\boldsymbol{F}\left[f_s(x)\right] = \frac{1}{\Delta x}\sum_{n=-\infty}^{n=\infty} \mathrm{F}\left(u - \frac{n}{\Delta x}\right)$$

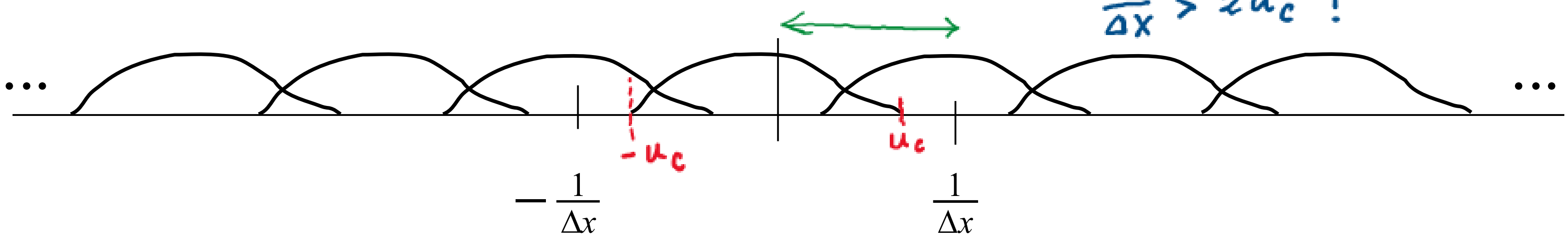
Prior to sampling, $\mathrm{F}(u) = \boldsymbol{F}\left[f(x)\right]$



After sampling, $F_s(u) = \boldsymbol{F}\left[f_s(x)\right]$

center-to-center : $\frac{1}{\Delta x}$

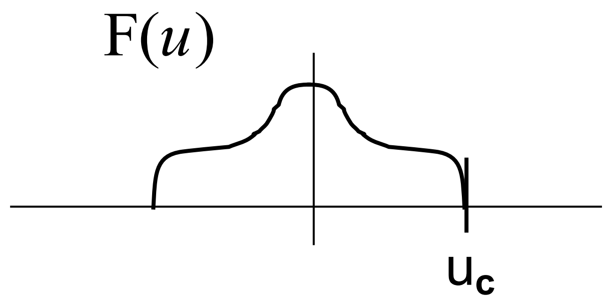
$\frac{1}{\Delta x} > 2u_c$?



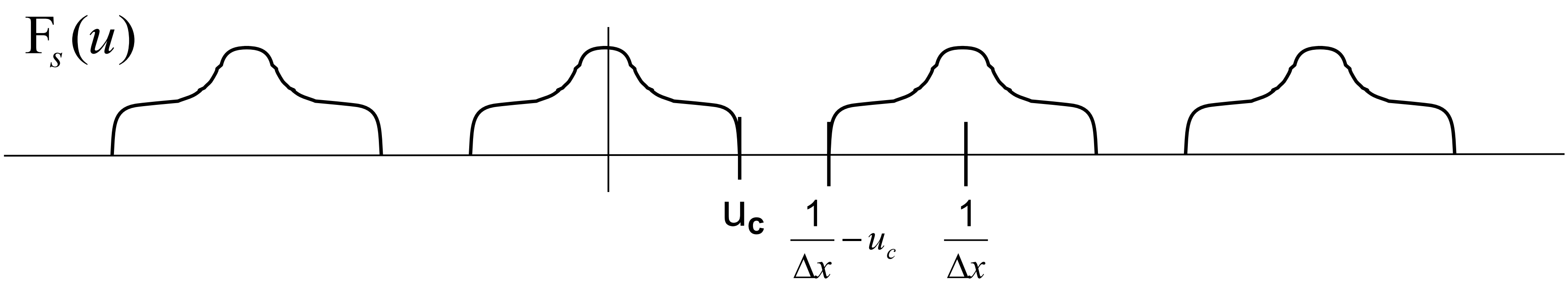
Fourier Transform of Sampled Function

If F(u) is band limited to u_c, (cutoff frequency)

F(u) = 0 for $|u| > u_c$.



To avoid overlap (aliasing),

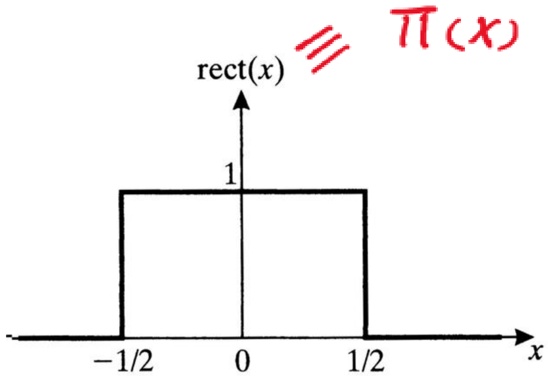


We would need $\frac{1}{\Delta x} - u_c > u_c$ **and therefore** $\frac{1}{\Delta x} > 2u_c$

Rect Function

- Rect function:

$$rect(x,y)=\begin{cases}1, \text{ for } |\mathrm{x}|<\dfrac{1}{2} \text{ and } |\mathrm{y}|<\dfrac{1}{2}\\ 0, \text{ otherwise}\end{cases}$$

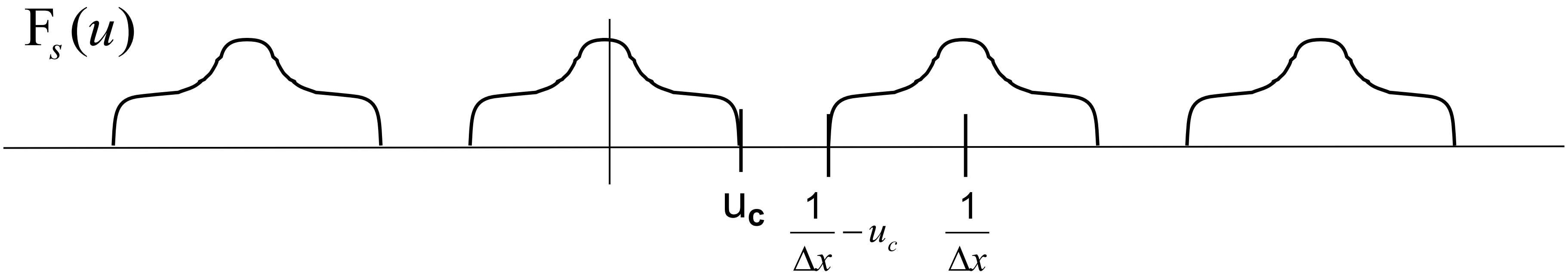


- It is normally used to pick up a particulate section of a given function:

$$f(x,y)\cdot rect(\frac{x-\xi}{w_X},\frac{y-\eta}{w_Y})$$

Fourier Transform of Sampled Function

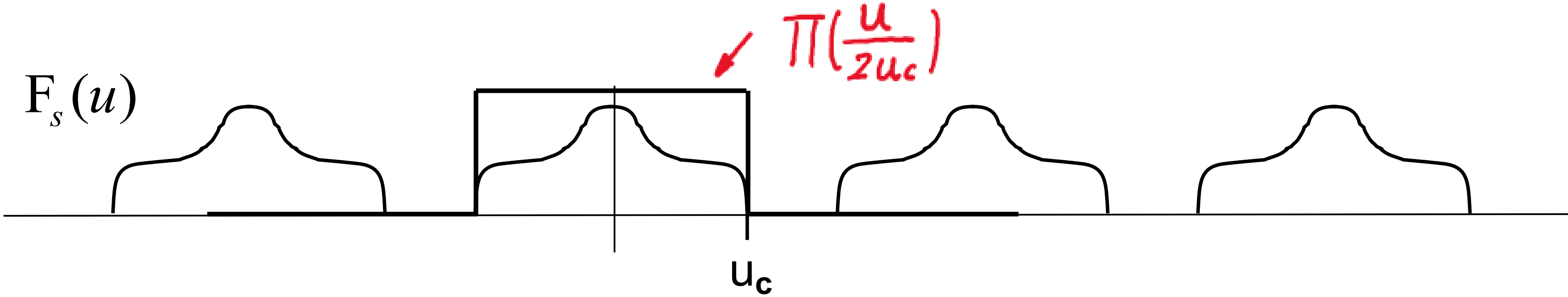
To avoid overlap (aliasing),



We would need $\frac{1}{\Delta x} - u_c > u_c$ **and therefore** $\frac{1}{\Delta x} > 2u_c$

$$f(x) = \mathcal{F}^{-1}\left[F_{f_s}(u) \cdot \Pi\left(\frac{u}{2u_c}\right)\right]$$

Restoration of Original Signal



Can we restore g(x) from the sampled frequency-domain signal? Yes, using the Interpolation Filter

$$H(u) = \Pi\left(\frac{u}{2u_c}\right)$$

$$\Updownarrow \quad \text{Fourier transform}$$

$$h(x) = 2u_c \cdot \mathrm{sinc}\,(2u_c x)$$

Basic Fourier Transform Pairs

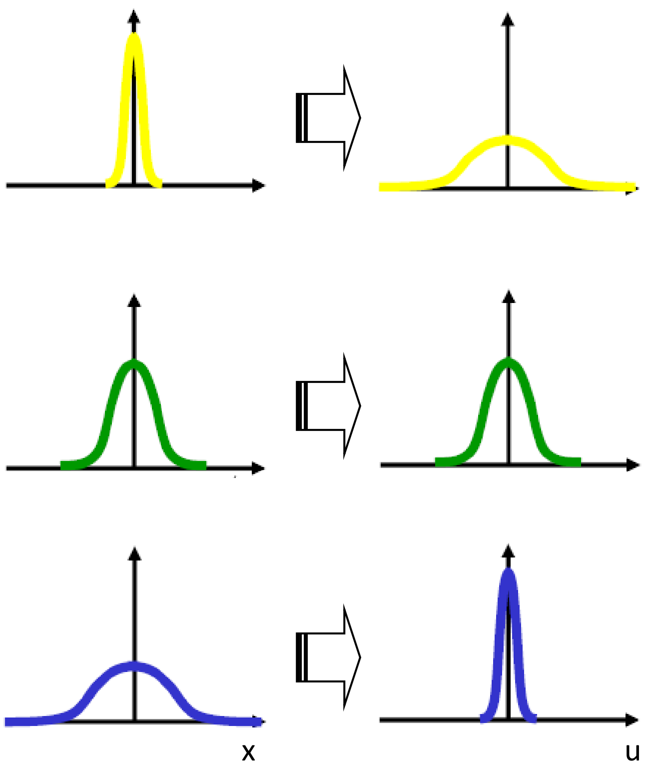
Basic Fourier transform pairs

Signal	Fourier Transform
1	$\delta(u, v)$
$\delta(x, y)$	1
$\delta(x - x_0, y - y_0)$	$e^{-j2\pi(ux_0 + vy_0)}$
$\delta_s(x, y; \Delta x, \Delta y)$	$\mathrm{comb}(u\Delta x, v\Delta y)$
$e^{j2\pi(u_0 x + v_0 y)}$	$\delta(u - u_0, v - v_0)$
$\sin[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2j}[\delta(u - u_0, v - v_0) - \delta(u + u_0, v + v_0)]$
$\cos[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2}[\delta(u - u_0, v - v_0) + \delta(u + u_0, v + v_0)]$
$\mathrm{rect}(x, y)$	$\mathrm{sinc}(u, v)$
$\mathrm{sinc}(x, y)$	$\mathrm{rect}(u, v)$
$\mathrm{comb}(x, y)$	$\mathrm{comb}(u, v)$
$e^{-\pi(x^2+y^2)}$	$e^{-\pi(u^2+v^2)}$

Properties of Fourier Transform

- Scaling $$F\left[f(ax,by)\right]=\frac{1}{ab}F\left(\frac{u}{a},\frac{v}{b}\right)$$

An 1-D example



Spatial domain, f(x)   Fourier Transform, F(u)

The shorter the pulse, the broader the spectrum !

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$where\ j=\sqrt{-1}$$

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}dudv$$

Restoration of Original Signal

The original function f can be recovered as

$f(x) = \mathcal{F}^{-1}\left[F_{f_s}(u) \cdot \Pi\left(\frac{u}{2u_c}\right)\right]$

$$f(x) = f_s(x) * h(x)$$

$$f_s(x) = \sum f(n\Delta x)\delta(x - n\Delta x)$$

$$= f_s(x) * [2u_c \cdot sinc(2u_c x)]$$

$$\mathrm{H}(u) = \Pi\left(\frac{u}{2u_c}\right)$$

$$\mathcal{F} \Updownarrow$$

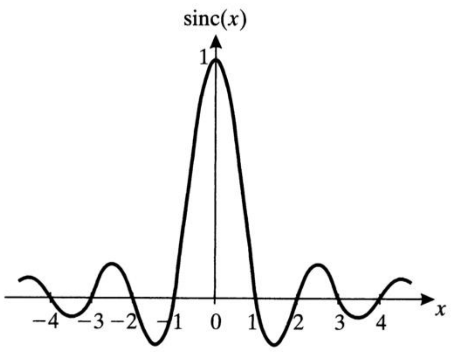
$$h(x) = 2u_c \cdot \mathrm{sinc}\,(2u_c x)$$

$$= \left[\sum_{n=-\infty}^{\infty} f(n \cdot \Delta x) \cdot \delta(x - n \cdot \Delta x)\right] * [2u_c \cdot \mathrm{sinc}(2u_c x)]$$

$$= \sum_{n=-\infty}^{\infty} 2 \cdot u_c \cdot f(n \cdot \Delta x) \cdot sinc(2u_c(x - n\Delta x))$$

$$\int_{-\infty}^{\infty} \delta(u - \xi) \cdot f(u) du = f(\xi)$$

f(x) is restored from a combination of sinc functions, each weighted and shifted according to its corresponding sampling point.



Nyquist Sampling Theorem

Nyquist Theorem:

To restore the original function, the sampling rate must be greater than twice the highest frequency component of the function.

Nyquist Sampling Interval:

The maximum sampling interval allowed without introduce aliasing is

$$\Delta x \leq \frac{1}{2u_c}$$

Harry Nyquist

Modulation

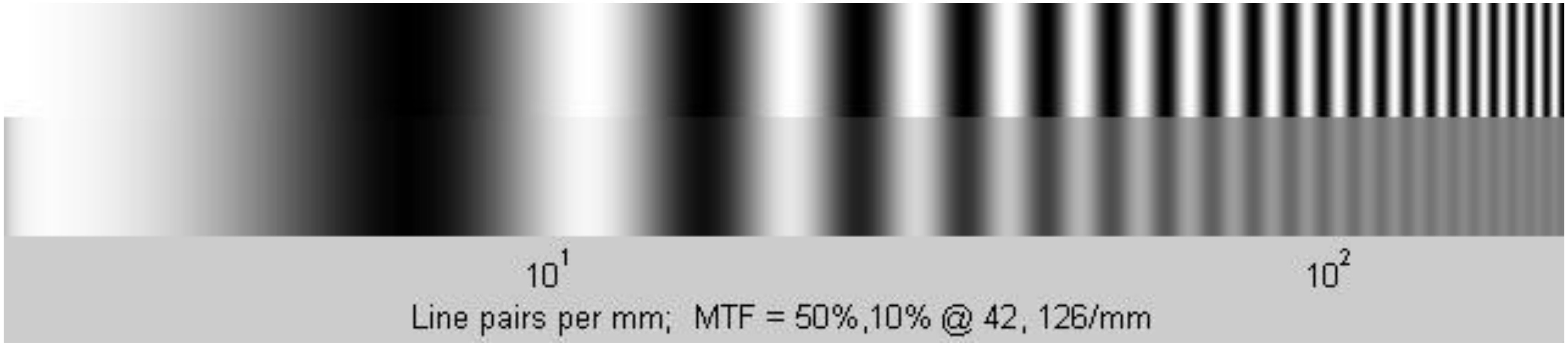
- **Now let this signal to pass through an LSI imaging system. Suppose an input signal function. Since**

$$f(x,y) = A + B\sin(2\pi u_0 x) = A + \frac{B}{2j}\left[e^{j2\pi u_0 x} - e^{-j2\pi u_0 x}\right],$$

- **Suppose the system impulse response function h(x,y) is circularly symmetric,**

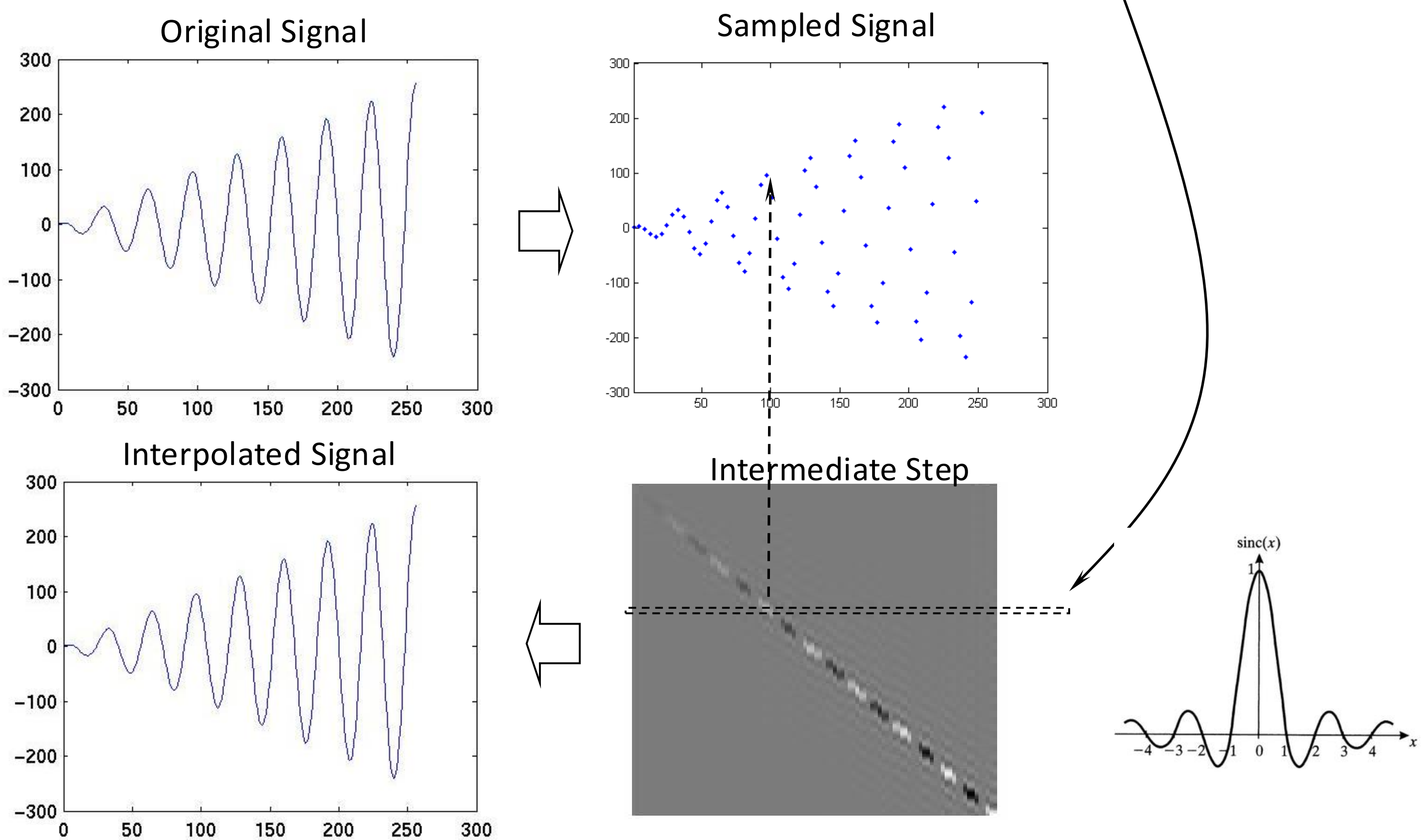
$H(u,v) = \mathcal{F}[h(x,y)]$

$$g(x,y) = AH(0,0) + B\,|H(u_0,0)|\sin(2\pi u_0 x).$$



An Example

$$f(x)=\sum_{n=-\infty}^{\infty} \boxed{2\cdot u_c \cdot f(n\cdot\Delta x)\cdot sinc(2u_c(x-n\Delta x))}$$



Adding all vertical values gives back the original function

Nyquist Condition

Nyquist Theorem:

To restore the original function, the sampling rate must be greater than twice the highest frequency component of the function.

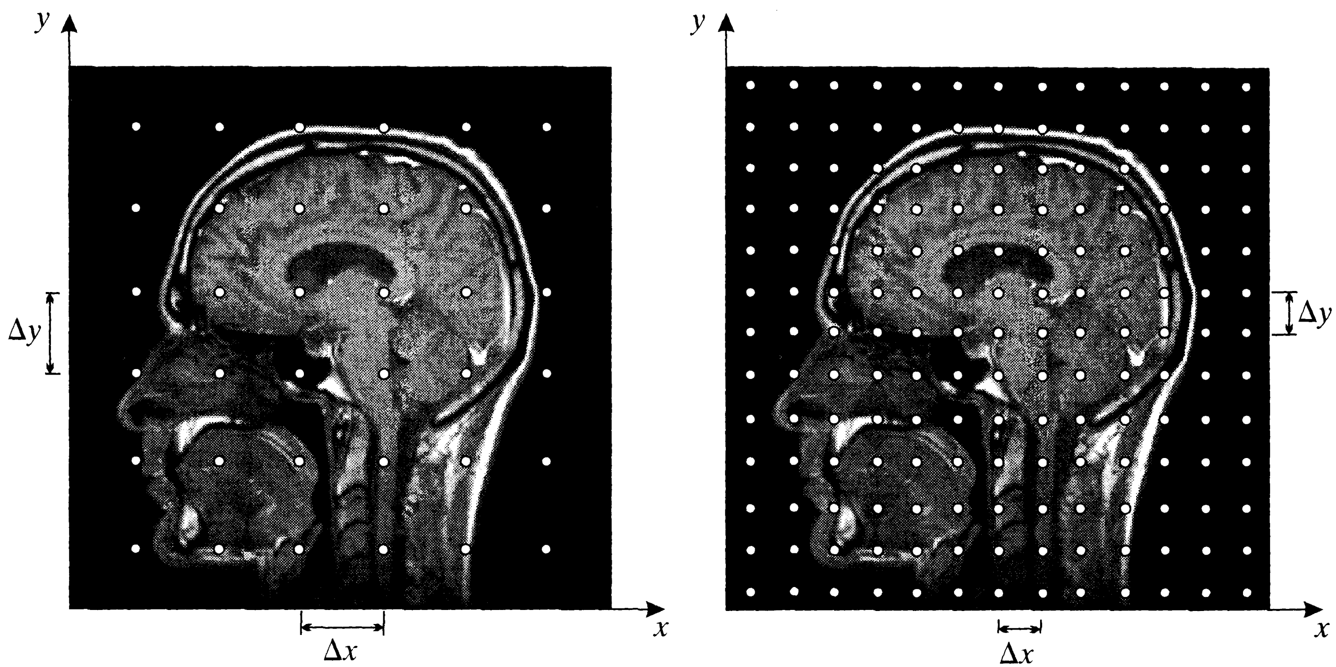
Nyquist Sampling Interval:

The maximum sampling interval allowed without introduce aliasing is

$$\Delta x \leq \frac{1}{2u_c}$$

Two-Dimensional Sampling

$$f_s(x,y) = f(x,y)\cdot\delta_s(x,y,\Delta x,\Delta y)$$
$$= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(x,y)\cdot\delta(x-n\Delta x, y-m\Delta y)$$
$$= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(n\Delta x, m\Delta y)\cdot\delta(x-n\Delta x, y-m\Delta y)$$



Fourier Transform of Sampled Image

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(u-\xi, v-\eta) f(u,v)\,du\,dv = f(\xi,\eta)$$

$$F_{f_s}(u) = \mathscr{F}\left[f_s(x,y)\right]$$

$$= \mathscr{F}\left[\delta_s\left(x,y,\Delta x,\Delta y\right)\cdot f(x,y)\right]$$

$$= comb\left(u\cdot\Delta x, v\cdot\Delta y\right) * \mathscr{F}\left[f(x,y)\right]$$

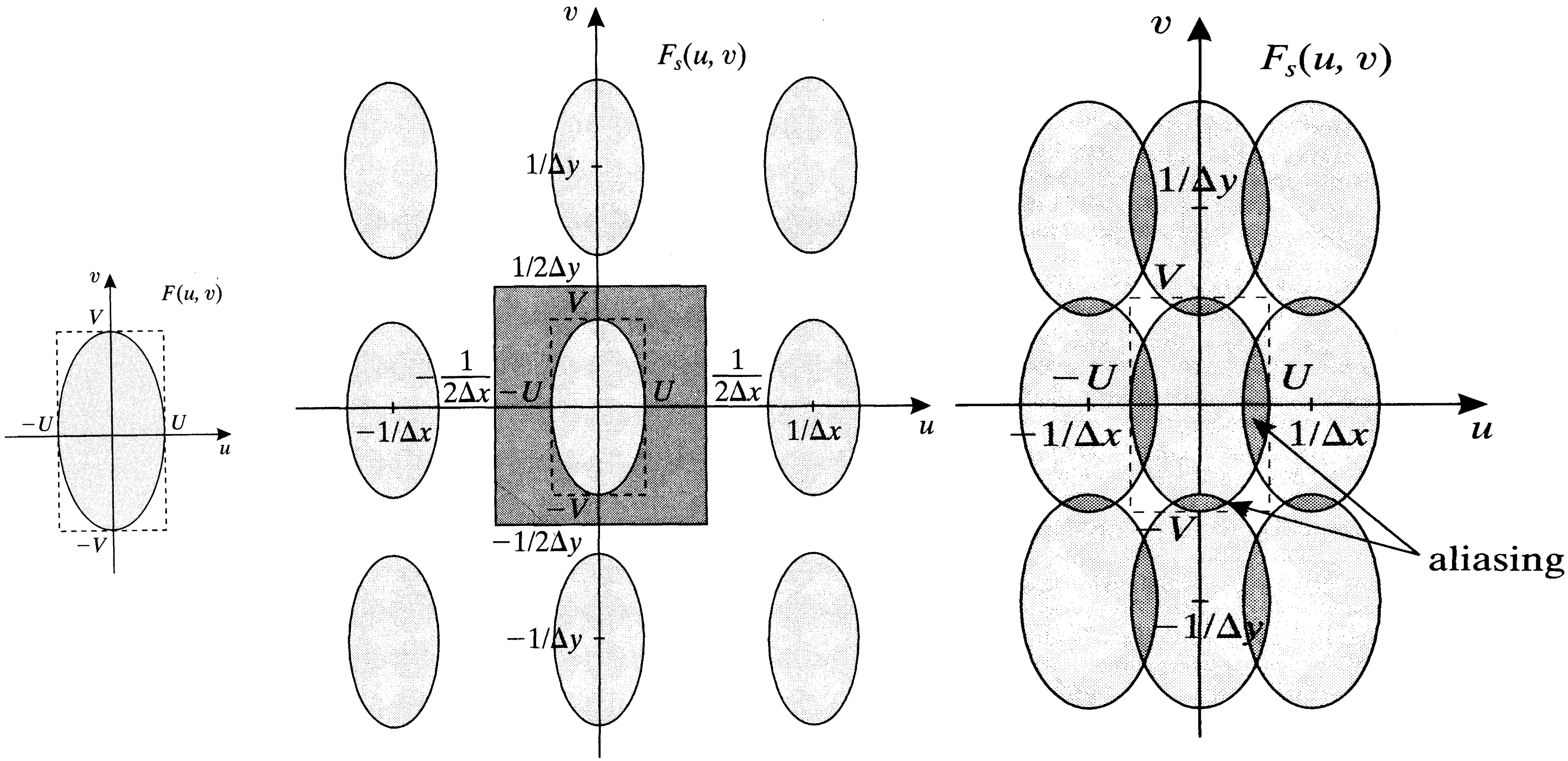
$$= \sum_{n=-\infty}^{n=\infty}\sum_{m=-\infty}^{m=\infty} \delta\left[\Delta x\left(u-\frac{n}{\Delta x}\right), \Delta y\left(v-\frac{m}{\Delta y}\right)\right] * F(u,v)$$

$$= \frac{1}{\Delta x\Delta y}\sum_{n=-\infty}^{n=\infty}\sum_{m=-\infty}^{m=\infty} \delta\left(u-\frac{n}{\Delta x}, v-\frac{m}{\Delta y}\right) * F(u,v)$$

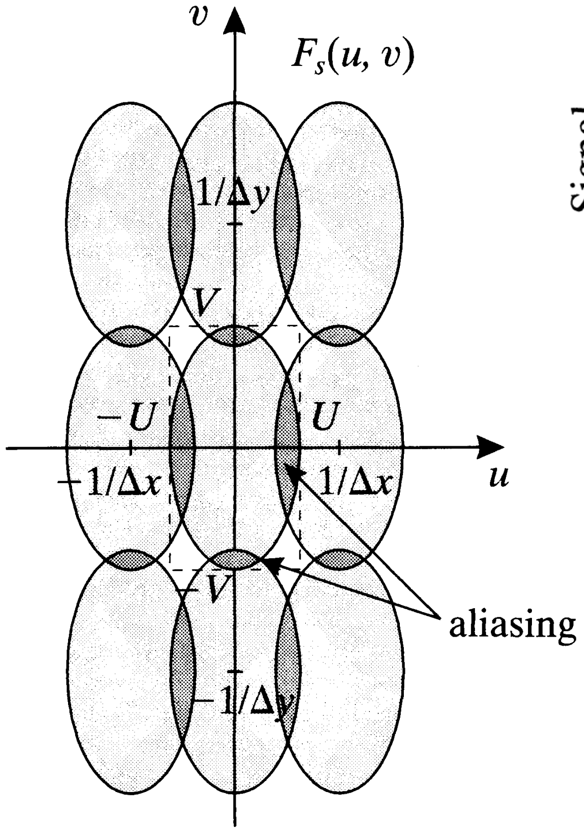
$$= \frac{1}{\Delta x\Delta y}\sum_{n=-\infty}^{n=\infty}\sum_{m=-\infty}^{m=\infty} F\left(u-\frac{n}{\Delta x}, v-\frac{m}{\Delta y}\right)$$

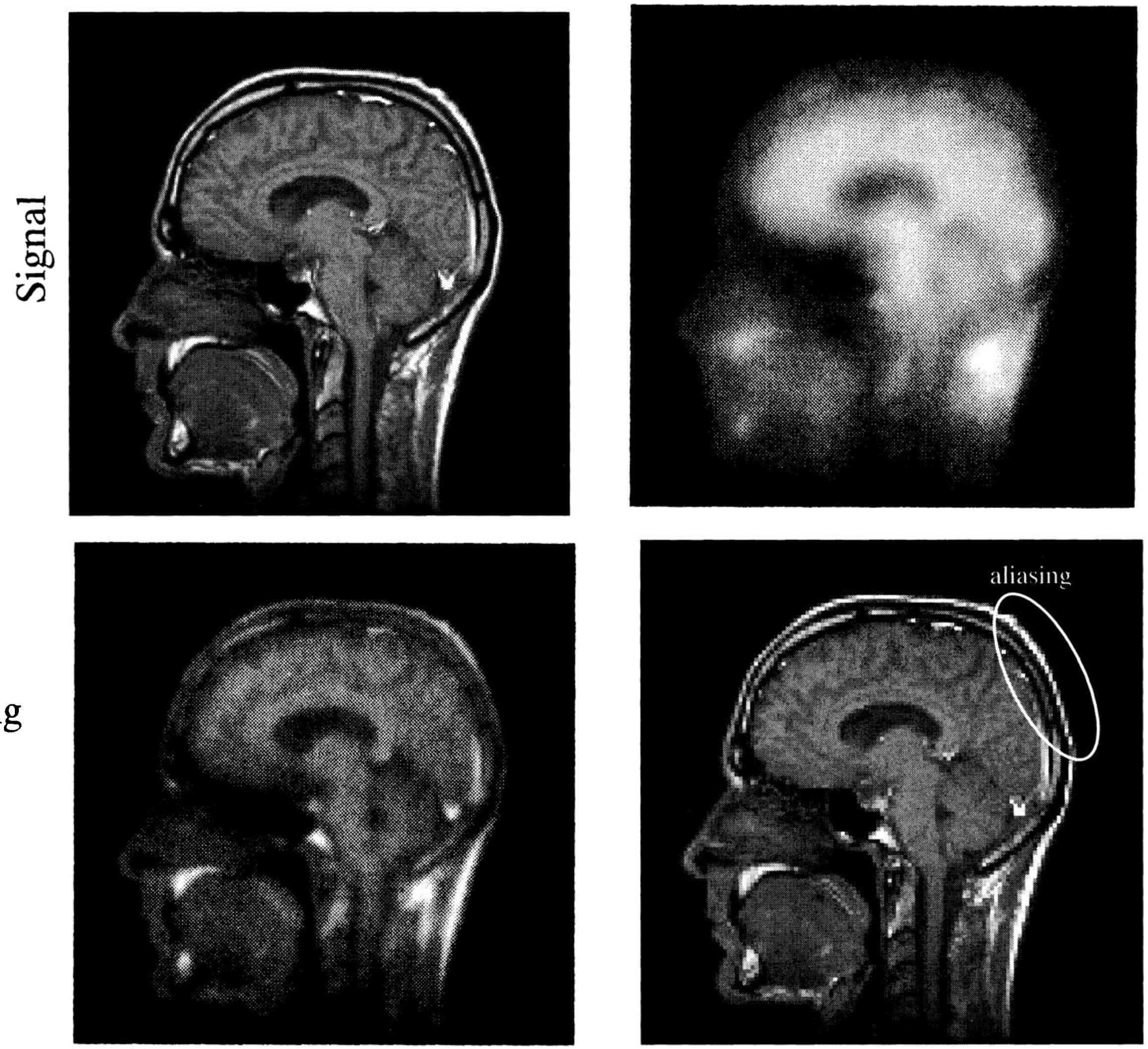
The result: Replicated F(u,v), or "islands" every 1/ Δx in u, and 1/Δy in v.

Fourier Transform of a Sampled 2-D Function



Consequence of Under-Sampling





Restoration of the Original 2-D Function

Interpolation Filter in 2-D

$$\mathrm{H}(u,v) = \prod(u\Delta x)\prod(v\Delta y)$$

$$\Updownarrow$$

$$h(x,y) = \left[\frac{1}{\Delta x}\cdot \mathrm{sinc}\,(\frac{x}{\Delta x})\right]\left[\frac{1}{\Delta y}\cdot \mathrm{sinc}\,(\frac{y}{\Delta y})\right]$$

Restoration of the Original 2-D Function

Given that the Nyquist sampling condition is met, the original function can be recovered exactly.

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(u-\xi, v-\eta)\, f(u,v)\, du\, dv = f(\xi,\eta)$$

$$
\begin{aligned}
f(x,y) &= f_s(x,y) * h(x,y) \\
&= f_s(x,y) * \left[\frac{1}{\Delta x}\cdot \operatorname{sinc}\left(\frac{x}{\Delta x}\right)\right]\left[\frac{1}{\Delta y}\cdot \operatorname{sinc}\left(\frac{y}{\Delta y}\right)\right] \\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} f(n\Delta x, m\Delta y)\cdot \delta(x-n\cdot\Delta x, y-m\cdot\Delta y) * \left\{\left[\frac{1}{\Delta x}\cdot \operatorname{sinc}\left(\frac{x}{\Delta x}\right)\right]\left[\frac{1}{\Delta y}\cdot \operatorname{sinc}\left(\frac{y}{\Delta y}\right)\right]\right\} \\
&= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \frac{1}{\Delta x \Delta y} f(n\Delta x, m\Delta y)\cdot \operatorname{sinc}\left(\frac{x-n\cdot\Delta x}{\Delta x}\right)\cdot \operatorname{sinc}\left(\frac{y-m\cdot\Delta y}{\Delta y}\right)
\end{aligned}
$$

Nyquist Sampling Theorem

Nyquist Theorem:

To restore the original function, the sampling rate must be greater than twice the highest frequency component of the function.

Nyquist Sampling Interval:

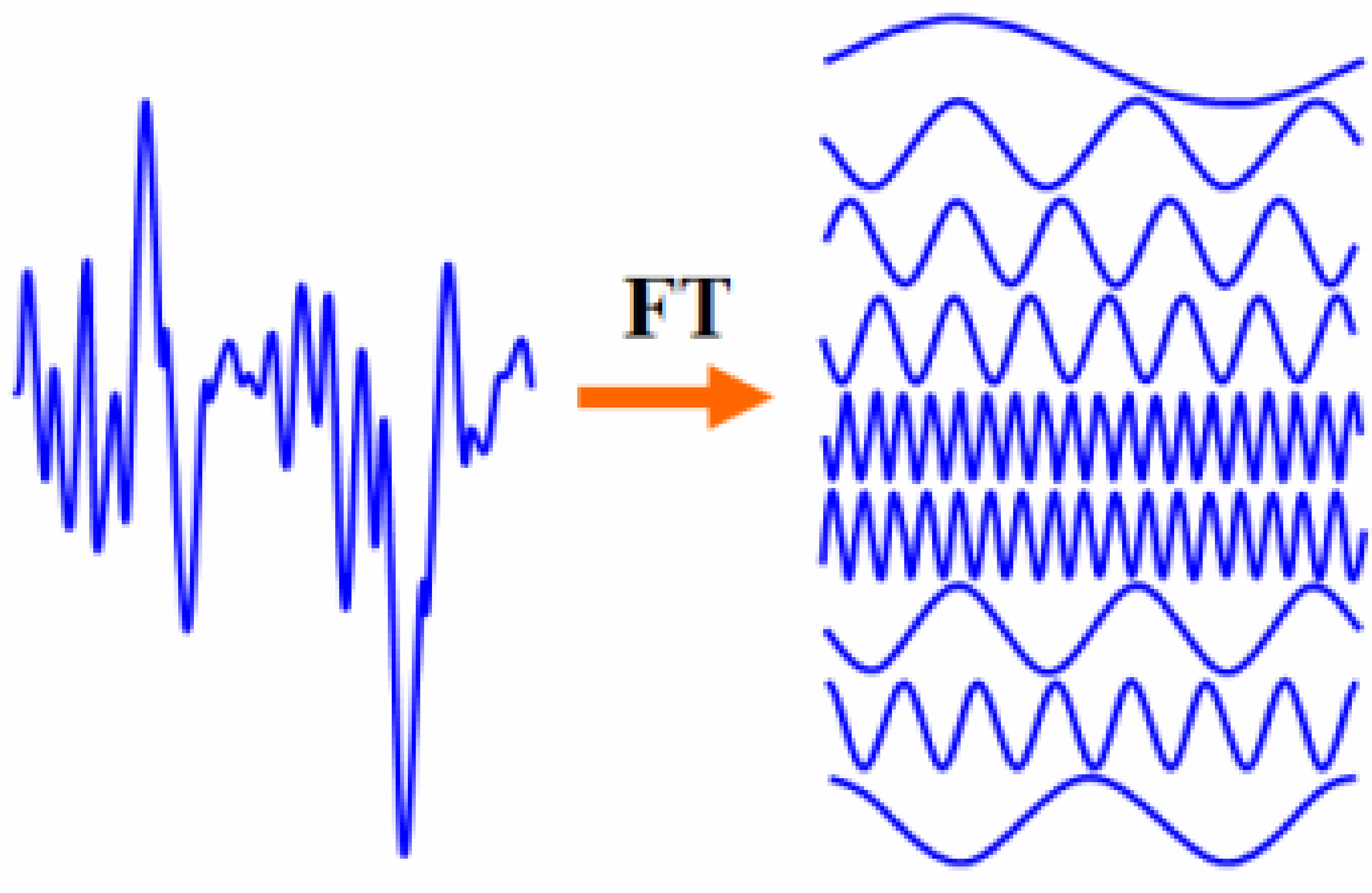
The maximum sampling interval allowed without introduce aliasing is

$$\Delta x \leq \frac{1}{2u_c}$$

Harry Nyquist

From continuous to discrete Fourier transform

Fourier Transform and Spatial Frequency



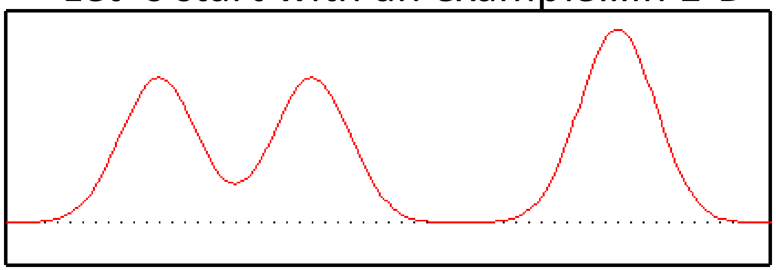
www.revisemri.com

Fourier Transform and Spatial Frequency

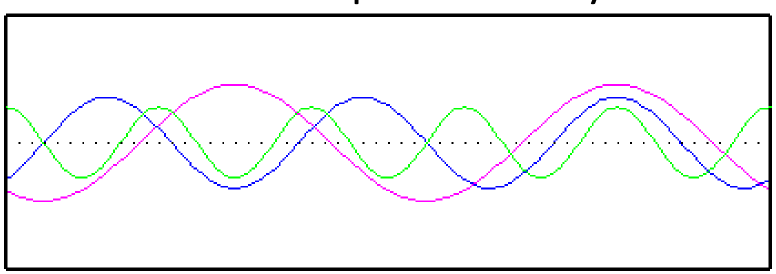
A Fourier Transform is an integral transform that re-expresses a function in terms of different sine waves of varying amplitudes, wavelengths, and phases.

So what does this mean exactly?

Let's start with an example…in 1-D

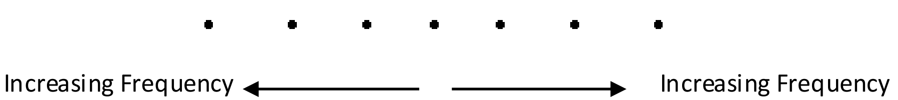


Can be represented by:



When you let these three waves interfere with each other you get your original wave function!

Since this object can be made up of 3 fundamental frequencies an ideal Fourier Transform would look something like this:



Notice that it is symmetric around the central point and that the amount of points radiating outward correspond to the distinct frequencies used in creating the image.

System Transfer Function

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

An ideal low-pass filter is defined as

$$H(u,v) = \begin{cases} 1 & \text{for } \sqrt{\mathrm{u}^2 + \mathrm{v}^2} \le c \\ 0 & \text{for } \sqrt{\mathrm{u}^2 + \mathrm{v}^2} > c \end{cases}$$

c is called the cut-off frequency.

System Transfer Function

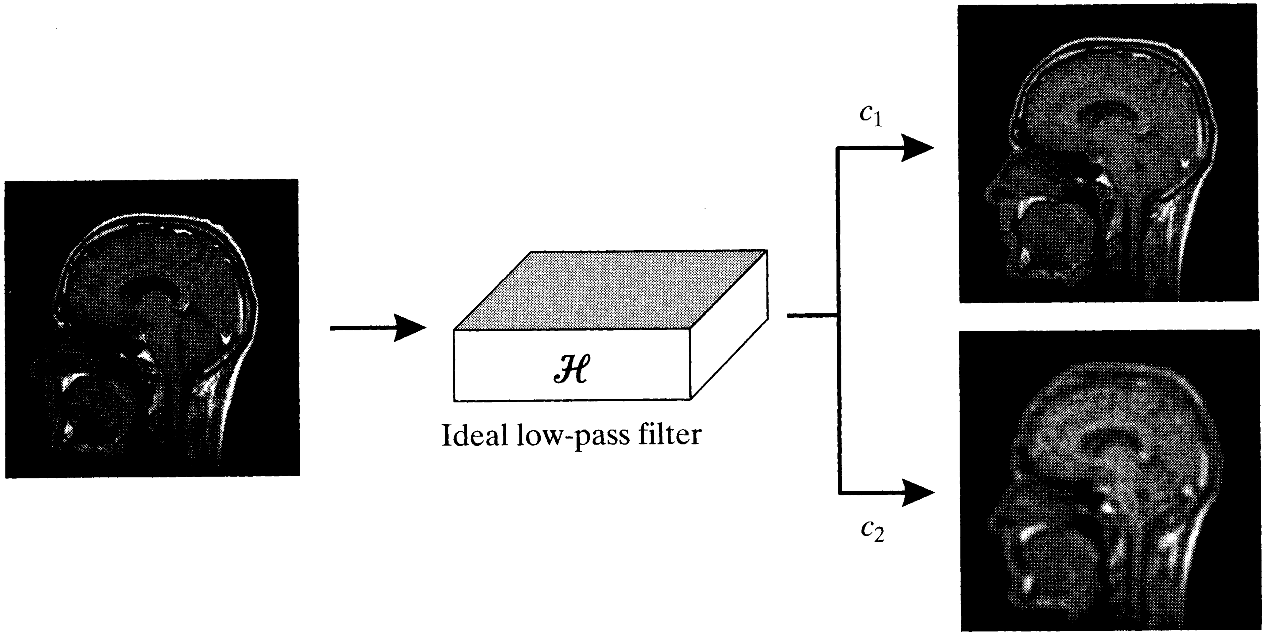
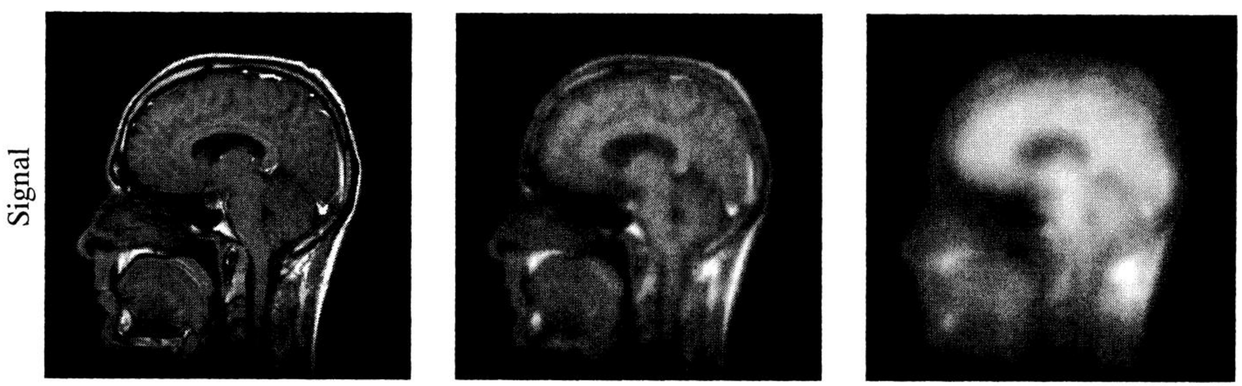


Figure 2.12
The response of an ideal low-pass filter for two values of the cutoff frequency c ($c_1 > c_2$).



Fourier Transform of Sampled Function

$$F_{f_s}(u) = \mathrm{F}\left[f_s(x)\right] = \mathrm{F}\left[\delta_s(x,\Delta x)\cdot f(x)\right]$$

$$= comb(u\cdot \Delta x) * \mathrm{F}\left[f(x)\right]$$

$$= \left\{\sum_{n=-\infty}^{n=\infty} \delta\left[\Delta x\left(u - \frac{n}{\Delta x}\right)\right]\right\} * F(u)$$

$$= \frac{1}{\Delta x}\sum_{n=-\infty}^{n=\infty}\left\{\delta\left(u - \frac{n}{\Delta x}\right) * F(u)\right\}$$

$$= \frac{1}{\Delta x}\sum_{n=-\infty}^{n=\infty} F\left(u - \frac{n}{\Delta x}\right)$$

$$comb(x) = \sum_{m=-\infty}^{\infty} \delta(x-m)$$

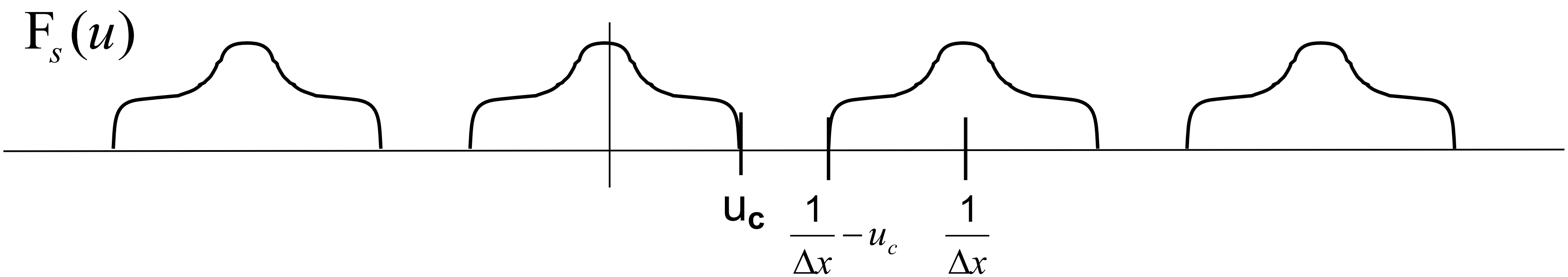
$$\delta_S(x,\Delta x) = \sum_{m=-\infty}^{\infty} \delta(x - m\Delta x)$$

$$\mathfrak{F}[\delta_s(x,\Delta x)] = comb(u\cdot\Delta x)$$

Multiplication in one domain becomes convolution in the other,

Fourier Transform of Sampled Function

To avoid overlap (aliasing),



We would need $\frac{1}{\Delta x} - u_c > u_c$ **and therefore** $\frac{1}{\Delta x} > 2u_c$

$$f(x) = \mathcal{F}^{-1}\left[F_{f_s}(u) \cdot \Pi\left(\frac{u}{2u_c}\right)\right]$$

Discrete Fourier Transform

Nyquist Condition - Revisited

Nyquist Theorem:

In order to restore the original function, the sampling rate must be greater than twice the highest frequency component of the function.

For a continuous but band-width limited function, all "information content" it contains can be preserved by a finite number of samples …

Continuous Fourier Transform of Sampled Function

$$F_{f_s}(u) = \mathscr{F}\left[f_s(x)\right] = \mathscr{F}\left[\delta_s\left(x, \Delta x\right) \cdot f(x)\right]$$

$$= \frac{1}{\Delta x} \sum_{n=-\infty}^{n=\infty} F\left(u - \frac{n}{\Delta x}\right)$$

Given that all "information content" of function f(x) is "carried" by a finite number (N) of samples in the spatial domain, can we equally use only N Fourier coefficients in spatial frequency domain to represent the sampled function?

Discrete Fourier Transform in 1-D

In reality, most experimental data comes in as sampled function defined on finite number of sampling points.

$$f_n = f(n\Delta x), \text{ for n} = 0,1,\ldots,\text{N - 1}$$
$$\text{where } \Delta x \text{ is the sampling interval}$$

The discrete Fourier transform (DFT) provides a de-composition of the sampled function with N spatial frequencies

$$U_n = \frac{n}{N\Delta x}, \text{ n} = -\frac{N}{2},\ldots,\frac{N}{2}$$
$$\text{where } \Delta x \text{ is the sampling interval}$$

Note that there are really N+1 spatial frequencies here, but the two extreme values will be identical.

Discrete Fourier Transform

In reality, most experimental data comes in as sampled function defined on finite number of sampling points.

$$\boldsymbol{F}\left[f_n\right] \equiv U_n = \int_{-\infty}^{\infty} f(x) e^{-j2\pi u_n x}\, dx \;\approx \sum_{k=0}^{N-1} f(k\Delta x) e^{-j2\pi u_n x} \Delta x$$

$$= \Delta x \sum_{k=0}^{N-1} f(k\Delta x) e^{-\frac{j2\pi nk}{N}} \;= \Delta x \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}$$

where Δx is the sampling interval and $u_n = \dfrac{n}{N\Delta x}$, is a given spatial frenquency.

Note that U_n is periodic in n with a period N,

$$U_{-n} = U_{N-n}, \mathrm{n} = 1,2,\ldots$$

So we normally let n in U_n varying from 0 to N-1.

Discrete Fourier Transform in 1-D

The discrete Fourier transform (DFT) is defined as

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}, \mathrm{n = 0,1,2,...N-1}$$

$$\mathrm{n = 0 \text{ corresponding to the DC component (spatial frenquency is zero)}}$$

$$\mathrm{n = 1,..., N/2-1 \text{ are corresponding to the positive frenquencies } 0 < u < u_c}$$

$$\mathrm{n = N/2, ..., N-1 \text{ are corresponding to the negative frenquencies } -u_c < u < 0}$$

The inverse DFT is defined as

$$f_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{-\frac{j2\pi nk}{N}}, \mathrm{k = 0,1,2,...N-1}$$

Discrete Fourier Transform

The 1-D Discrete Fourier Transform (DFT) is defined as

$$F_n = \sum_{k=0}^{N-1} f_k W^{nk}, \quad \text{n=0,1,2…,N-1}$$

$$\text{with} \quad W \equiv e^{\frac{j2\pi}{N}}$$

So calculating the DFT involves multiplication of a vector with a matrix

$$\mathbf{F} = \mathbf{Df} \quad \Rightarrow \quad \begin{pmatrix} F_0 \\ F_1 \\ F_2 \\ \vdots \\ F_{N-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & \cdots & W^{N-1} \\ 1 & W^2 & W^4 & \cdots & W^{2(N-1)} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{N-1} & W^{2(N-1)} & \cdots & W^{(N-1)^2} \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_{N-1} \end{pmatrix}$$

It requires $O(N^2)$ complex operations!

Inverse Discrete Fourier Transform

The original spatial domain function can be recovered

$$f_n = \sum_{k=0}^{N-1} F_k W^{-nk}, 0 \le \mathrm{n} \le \mathrm{N-1} \text{ and } W \equiv e^{\frac{j2\pi}{N}}$$

So calculating the DFT involves multiplication of a vector with a matrix

$$\mathbf{f} = \mathbf{D}^{-1}\mathbf{F} \quad \Longrightarrow \quad \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_{N-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & W^{-1} & W^{-2} & \cdots & W^{-(N-1)} \\ 1 & W^{-2} & W^{-4} & \cdots & W^{-2(N-1)} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{-(N-1)} & W^{-2(N-1)} & \cdots & W^{-(N-1)^2} \end{pmatrix} \begin{pmatrix} F_0 \\ F_1 \\ F_2 \\ \vdots \\ F_{N-1} \end{pmatrix}$$

Fast Fourier Transform

In fact, the DFT can be calculated using the Fast Fourier Transform (FFT) algorithm …

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}, \mathrm{n} = 0,1,2,\ldots,\mathrm{N-1}$$

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}$$

$$= \sum_{k=0}^{N/2-1} e^{-\frac{j2\pi n(2k)}{N}} f_{2k} + \sum_{k=0}^{N/2-1} e^{-\frac{j2\pi n(2k+1)}{N}} f_{2k+1}$$

$$= \sum_{k=0}^{N/2-1} e^{-\frac{j2\pi nk}{N/2}} f_{2k} + W^n \sum_{k=0}^{N/2-1} e^{-\frac{j2\pi nk}{N/2}} f_{2k+1} \qquad W \square e^{\frac{j2\pi}{N}}$$

$$= F_n^e + W^n F_n^o, \qquad \mathrm{n} = 0,1,2,\ldots,\mathrm{N-1}$$

Fast Fourier Transform

The wonderful thing about the previous results is that it can be used recursively … so that …

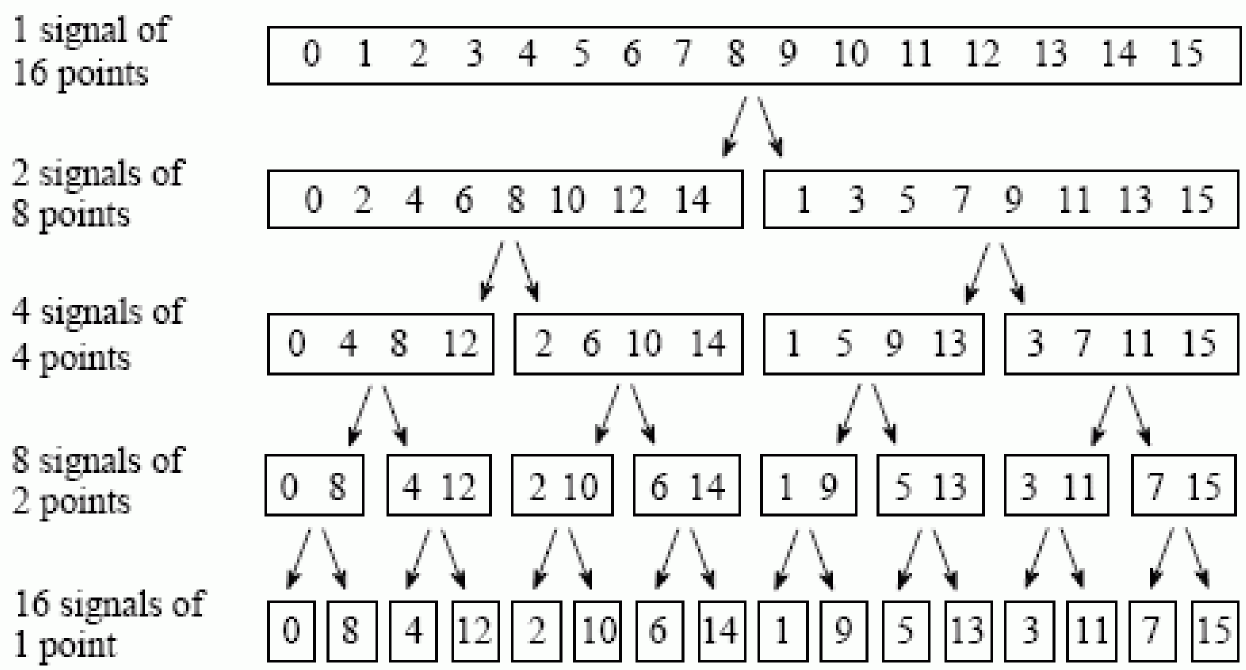


FIGURE 12-2
The FFT decomposition. An N point signal is decomposed into N signals each containing a single point. Each stage uses an *interlace decomposition*, separating the even and odd numbered samples.

http://www.dspguide.com/ch12/2.htm

Discrete Fourier Transform in 1-D

The discrete Fourier transform (DFT) is defined as

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}, \mathrm{n} = 0,1,2,\ldots\mathrm{N}-1$$

$\mathrm{n} = 0$ corresponding to the DC component (spatial frenquency is zero)

$\mathrm{n} = 1,\ldots,\mathrm{N}/2-1$ are corresponding to the positive frenquencies $0 < \mathrm{u} < \mathrm{u_c}$

$\mathrm{n} = \mathrm{N}/2,\ldots,\mathrm{N}-1$ are corresponding to the negative frenquencies $-\mathrm{u_c} < \mathrm{u} < 0$

The inverse DFT is defined as

$$f_k = \frac{1}{N}\sum_{n=0}^{N-1} F_n e^{-\frac{j2\pi nk}{N}}, \mathrm{k} = 0,1,2,\ldots\mathrm{N}-1$$

Fast Fourier Transform

Number of operations are needed for an 1-D Discrete Fourier Transform (DFT) of N points:

$$F_n = \sum_{k=0}^{N-1} f_k W^{nk}$$

N × N complex multiplications and N*(N-1) complex additions!

Number of operations are needed for an 1-D Discrete FFT of N points:

$N/2 \times (\log_2 N-1) \sim N/2 \times \log_2 N$ complex multiplications and $N \times \log_2 N$ complex additions!

$N \times N \rightarrow N/2 \cdot \log_2 N$

Fast Fourier Transform

For an 1-D Discrete Fourier Transform of 1024 points:

DFT: ~1048576 (×) and 1047552 (+)

Versus

FFT: 5120 (×) and 10240 (+)

For convolution operation:

$$g(x,y) = f(x,y) * h(x,y) = \mathscr{F}^{-1}\left\{\mathscr{F}\left[f(x,y)\right]\cdot\mathscr{F}\left[g(x,y)\right]\right\}$$

Direct: ~1048576 (×)

Versus

FFT: ~5120 × 3 (×)

A factor of ~600 difference!

Properties of Discrete Fourier Transform

Linearity

$$\text{if } f(n) \Leftrightarrow F(m) \text{ and } g(n) \Leftrightarrow G(m)$$
$$\text{then } af(\text{x}) + bg(\text{x}) \Leftrightarrow aF(u) + bG(u)$$

Shifting

$$DFT\left[f(n-k)\right] \Rightarrow DFT\left[\text{F(m)}\right] e^{-i \cdot 2\pi \cdot \frac{km}{N}}$$

Example : if k=1 there is a 2π shift as m varies from 0 to N-1

Discrete Fourier Transform in 2-D

2-D DFT is defined as

$$F(n,m) \equiv \sum_{l=0}^{M-1}\sum_{k=0}^{N-1} f_s(k,l)e^{-\frac{j2\pi nk}{N}}e^{-\frac{j2\pi ml}{M}},$$

where

$$f(k,l) \equiv f(k\Delta x, l\Delta y)$$

$$\text{n} = 0,1,2,\ldots\text{N - 1 and m} = 0,1,2,\ldots\text{M - 1}$$

Similar to the continuous case,

$$\begin{aligned} F(n,m) &= \text{FFT on Index 1 (FFT on Index 2}[f(k,l)]) \\ &= \text{FFT on Index 2 (FFT on Index 1}[f(k,l)]) \end{aligned}$$

Discrete Inverse Fourier Transform in 2-D

2-D inverse DFT is defined as

$$f_s(n,m) \equiv \frac{1}{N \cdot M} \sum_{l=0}^{M-1} \sum_{k=0}^{N-1} F(k,k) e^{-\frac{j2\pi nk}{N}} e^{-\frac{j2\pi ml}{M}},$$

where

$$f_s(n,m) \equiv f(n\Delta x, m\Delta y)$$

$$\mathrm{n} = 0,1,2,\ldots \mathrm{N} - 1 \text{ and } \mathrm{m} = 0,1,2,\ldots \mathrm{M} - 1$$

Chapter 2: Mathematical Preliminaries

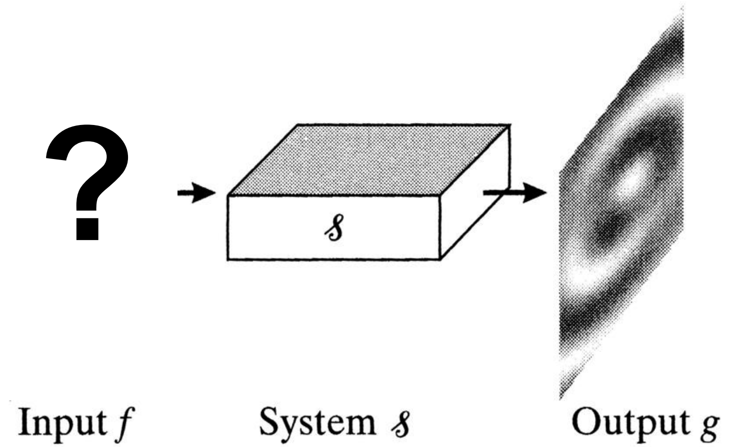
Mathematical Preliminaries for 2-D Image Reconstructions

Contents

- Projection and Radon transform
- Sinogram
- Central slice theorem
- Image reconstruction with simple backprojection
- Filtered-backprojection
- Other analytical reconstruction methods.
- Matlab examples

The Inverse Problem

General Inverse Problem and Image Reconstruction



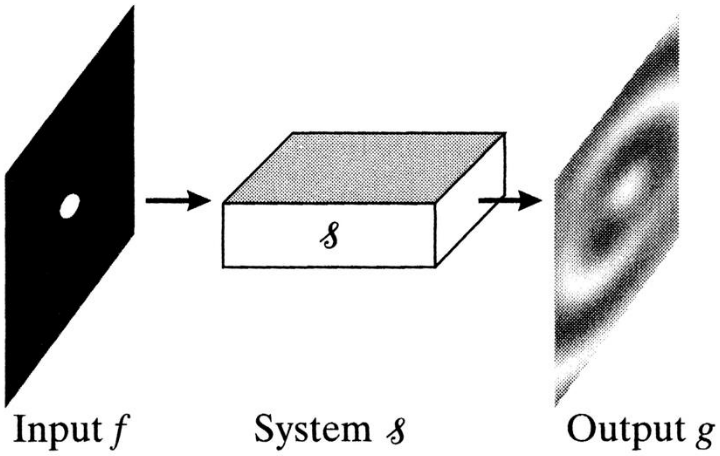
What should be the input signal that gave rise to the output data?

Convolution Theorem Revisited

The convolution theorem enables one to perform convolution operation as multiplication process in spatial frequency domain.

$$f(x,y) * g(x,y) = \mathscr{F}^{-1}\left\{\mathscr{F}\left[f(x,y)\right]\cdot \mathscr{F}\left[g(x,y)\right]\right\}$$

By using the *Fast Fourier Transform* (FFT) algorithms, the convolution operation can be performed very efficiently !! This provide a practical way for modeling linear shift-invariant systems …



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

Radon Transform and Projection

Projection Data from Early X-ray CT Systems

- Uses a collimator to keep exposure to a slice
- Builds image from multiple projections
- We will assume parallel rays for now.
    - Actually, how first-generation scanners worked.
    - Translate source and single detector across the body for one angle
    - Then rotate the source and detector to get the next angle

The principles of sectional imaging 103

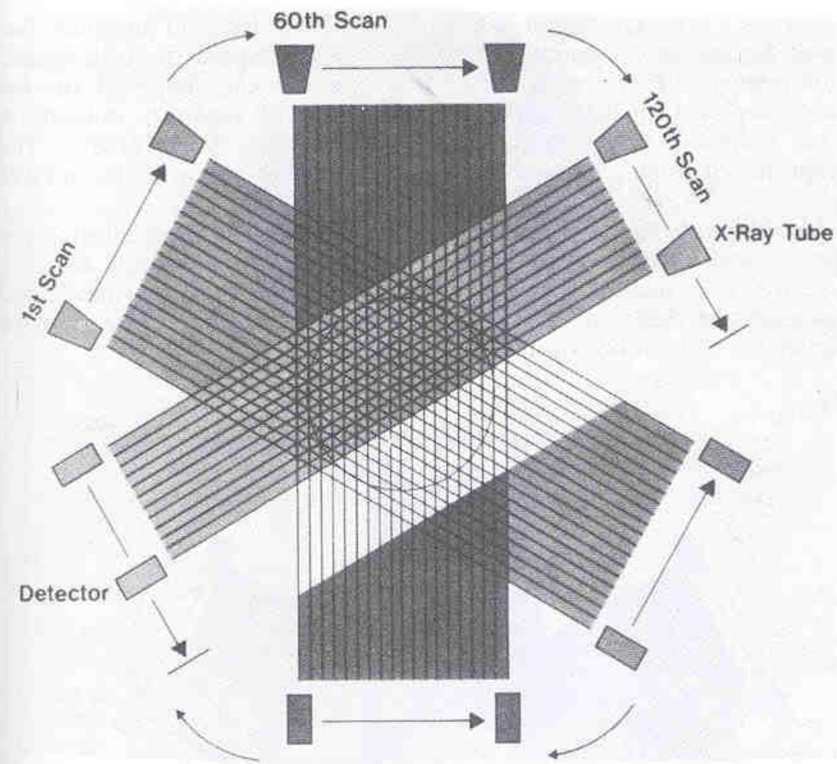
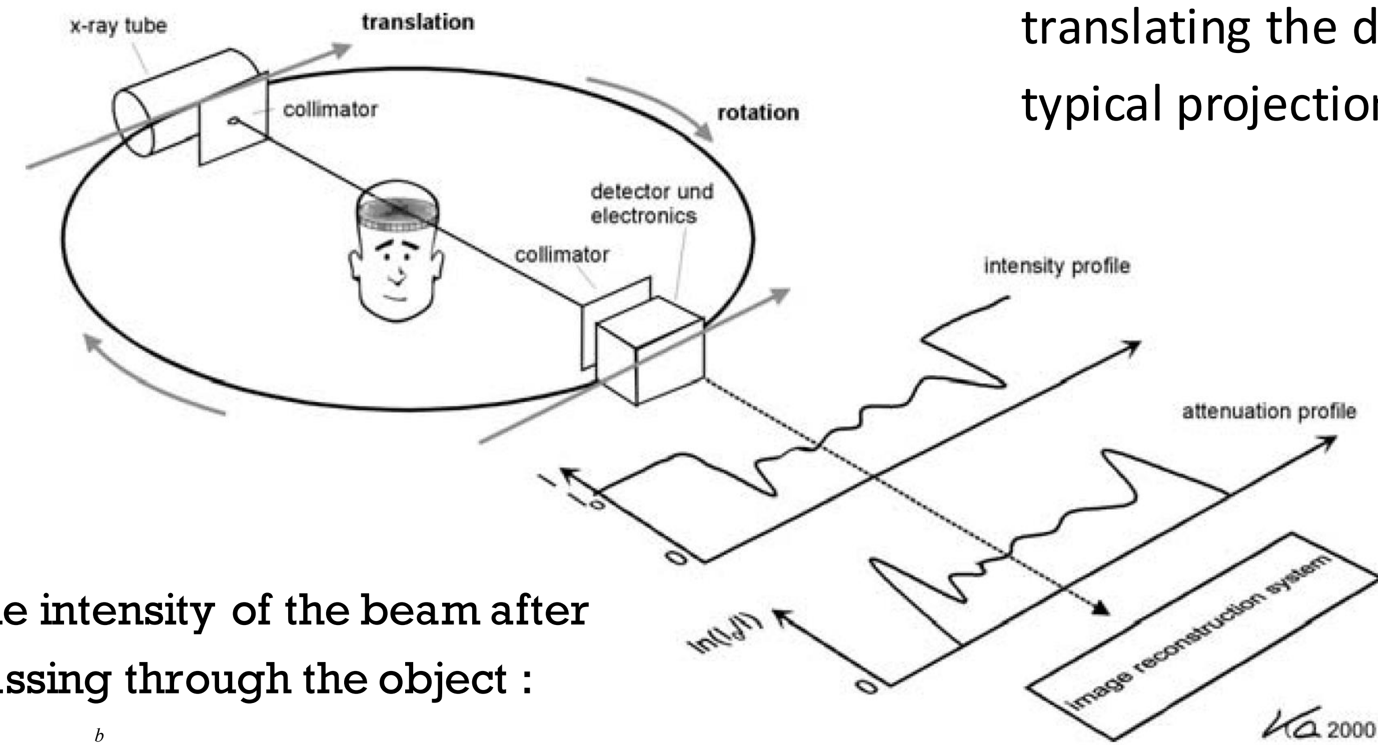


Figure 4.5 Schematic representation of a first-generation CT scanner. It utilises a single pencil beam and single detector for each scan slice. The x-ray source and detector are passed transversely across the object being scanned, with incremental rotations of the system at the end of each transverse motion. (Reproduced from Maravilla and Pastel (1978).)

Projection Data from Early X-ray CT Systems



Data measured by translating the detector is a typical projection data.

The intensity of the beam after passing through the object :

$$I = I_0 e^{-\int_a^b \mu(t)\cdot dt} \Leftrightarrow \ln(I_0 / I) = \int_a^b \mu(t)\cdot dt$$

From Computed Tomography, Kalender, 2000.

A Typical Gamma Camera

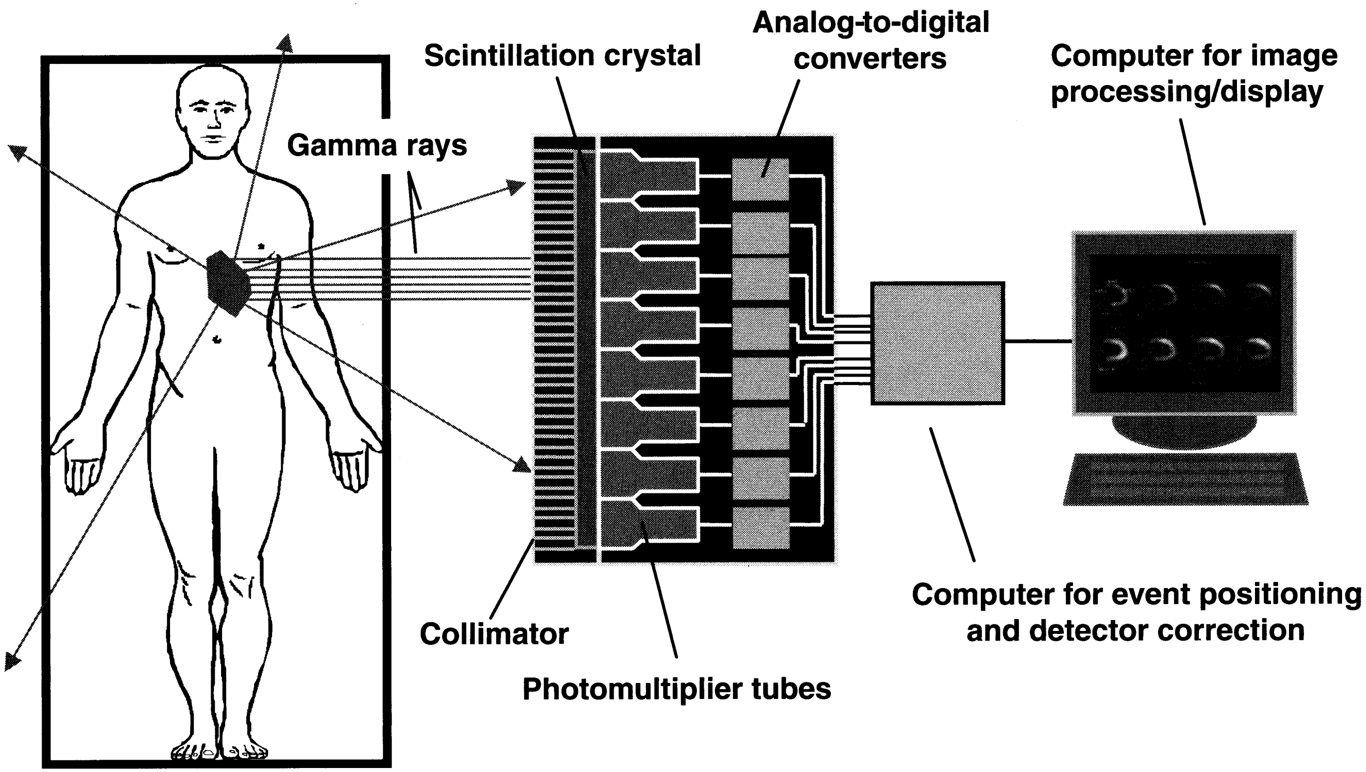
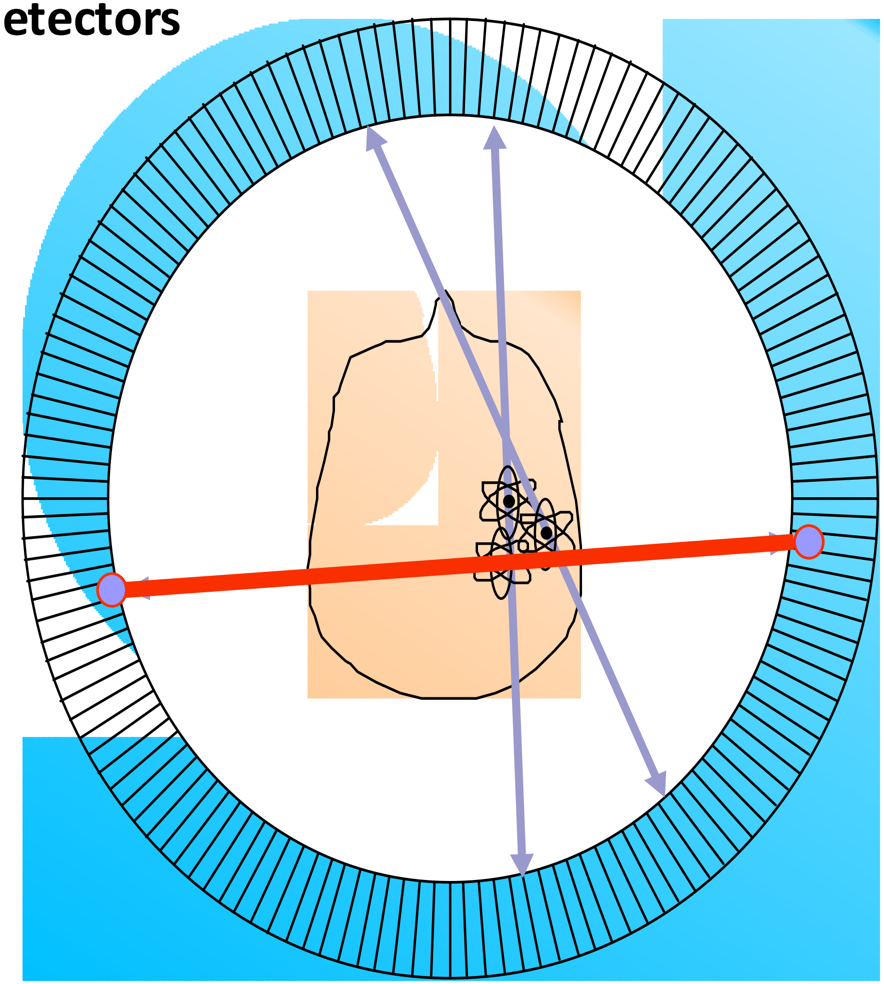


FIGURE 7 Schematic diagram of a conventional gamma camera used in SPECT. The collimator forms an image of the patient on the scintillation crystal, which converts gamma rays into light. The light is detected by photomultiplier tubes, the outputs of which are digitized and used to compute the spatial coordinates of each gamma-ray event (with respect to the camera face). A computer is used to process, store, and display the images.

Detect Radioactive Decays

Ring of Photon Detectors



- Radionuclide decays, emitting β^{+}.
- β^{+} annihilates with e^{-} from tissue, forming back-to-back 511 keV photon pair.
- 511 keV photon pairs detected via time coincidence.
- Positron lies on a line defined by a detector pair (known as a chord, a line of response, or a *LOR*).

Detect Pair of Back-to Back 511 keV Photons

Example -- MRI of the Brain

(a) (b) (c)

Figure 12.10 Three images of the same slice through the skull. Contrast between the tissue types are classified as (a) P_D-weighted, (b) T_2-weighted, and (c) T_1-weighted. (Used with permission of GE Healthcare.)

Proton Density
T_E = 17 ms, T_R = 6000 ms

T_2 Contrast
T_E = 102 ms, T_R = 6000 ms

- **Long T_R → minimizing T_1 effect**
- **GM, WM decayed, CSF partially decayed → large contrast between GM/WM and CSF**
- **Small contrast between GM and WM (WM is darker due to shorter T_2)**

T_1 Contrast
T_E = 17 ms, T_R = 600 ms

- **Short T_R → building up T_1 effect**
- **T_R falls between T_1's for GM, WM but much shorter than that for CSF → GM/WM should built up more (brighter in the image) that CSF.**

TABLE 12.2

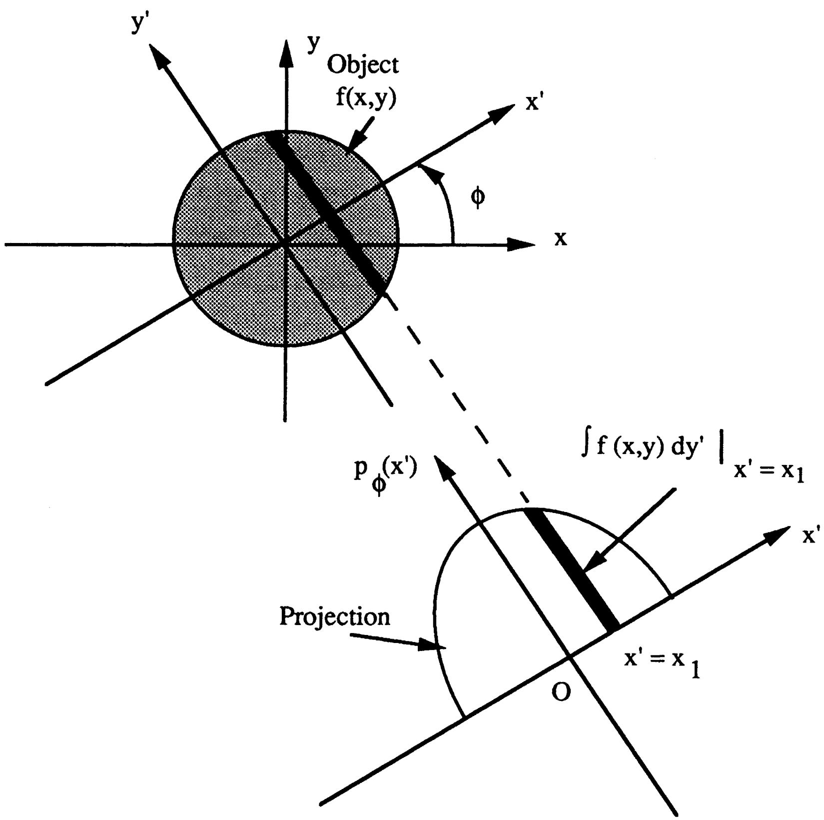
Typical Brain Tissue Parameters Measured at 1.5 T

Tissue Type	Relative P_D	T_2 (ms)	T_1 (ms)
White matter	0.61	67	510
Gray matter	0.69	77	760
Cerebrospinal fluid	1.00	280	2650

Source: Adapted from Liang and Lauterbur, 2000.

How do we mathematically describe projections?

The Line Integral Projection

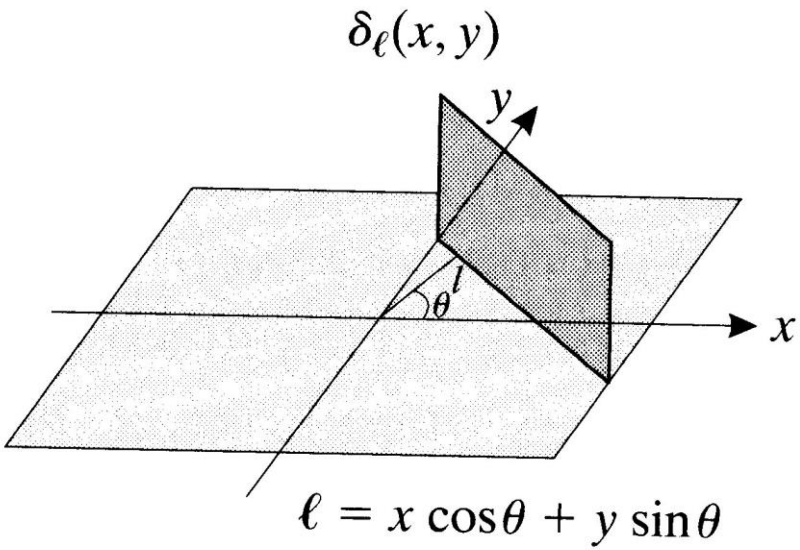


- Measures the integral property a physical object along a straight line.
- Projection data is naturally acquired by successive sampling the object at given angular intervals.

Line Impulse Signal (revisited)

$$\delta_L(x,y) = \delta(x\cos\theta + y\sin\theta - l)$$

$$where\ \delta(x) = \begin{cases} >0, & x\cos\theta + y\sin\theta = l \\ 0, & otherwise \end{cases}$$



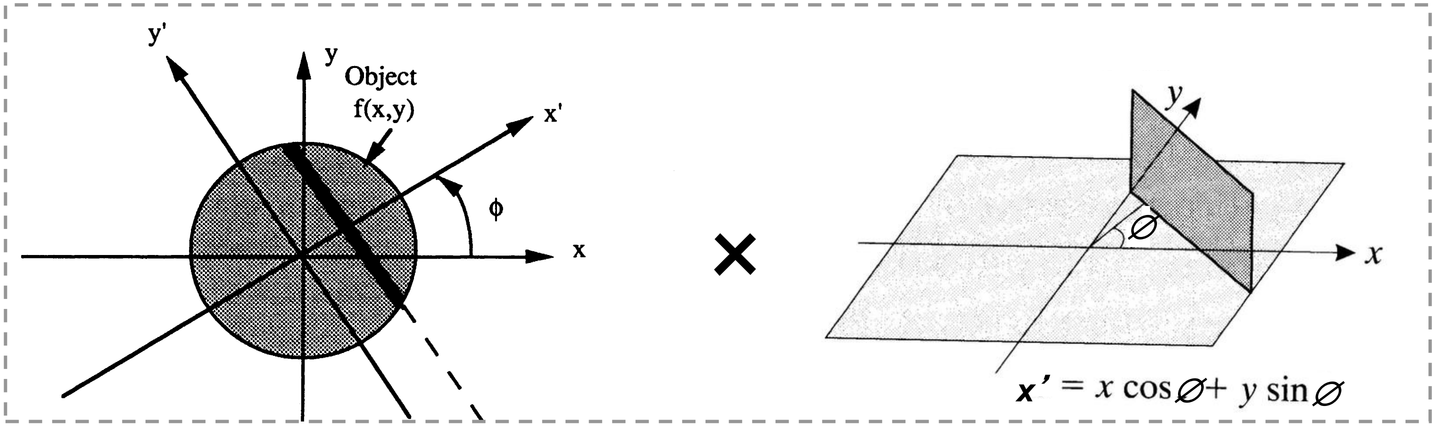
Line Impulse Signal (revisited)

$$\delta_L(x, y) = \delta(x\cos\theta + y\sin\theta - l)$$

- It can be used to measure the spatial resolution of a given imaging system.

- It is used to calculate the line-integral projection data for a given 2-D object.

Radon Transform



2-D integral

The value of the projection function $p_{\phi}(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The **Radon transform**, which produces the projection data, is defined as

$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Sinogram

Radon Transform and Sinogram

Radon transform maps a 2-D function f(x,y) into a sinogram.

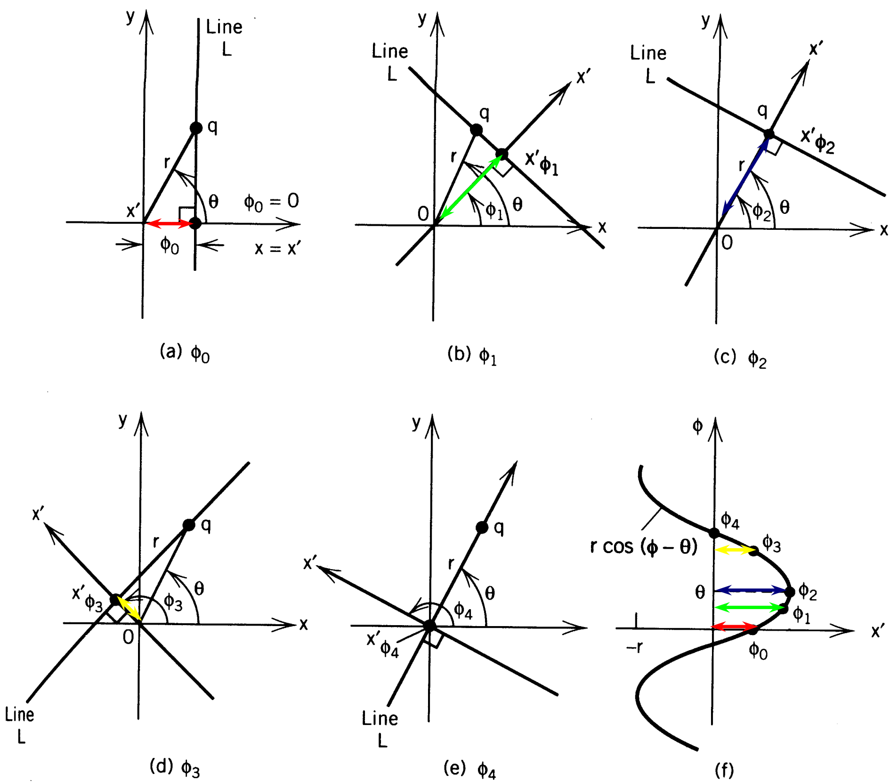


Figure 3-2 Sinogram of a point q at a various angles ϕ_i for a given θ: (a) $\phi_0 = 0$, (b) $\phi_1 = 45°$, (c) $\phi_2 = 60°$, (d) $\phi_3 = 135°$, and (e) $\phi_4 = 150°$. At $i = 0$ or $\phi_i = 0$, the rotated coordinates x' coincide with the reference x-axis and give $x' = r\cos(\phi - \theta) = r\cos\theta$.

$$\mathscr{P}(\phi, x') = \mathscr{R}[f(x,y)]$$

Any point represented by polar coordinates (r,θ) will simply follow the equation $x' = r\cos(\phi - \theta)$ in the sinogram space.

Radon Transform and Sinogram

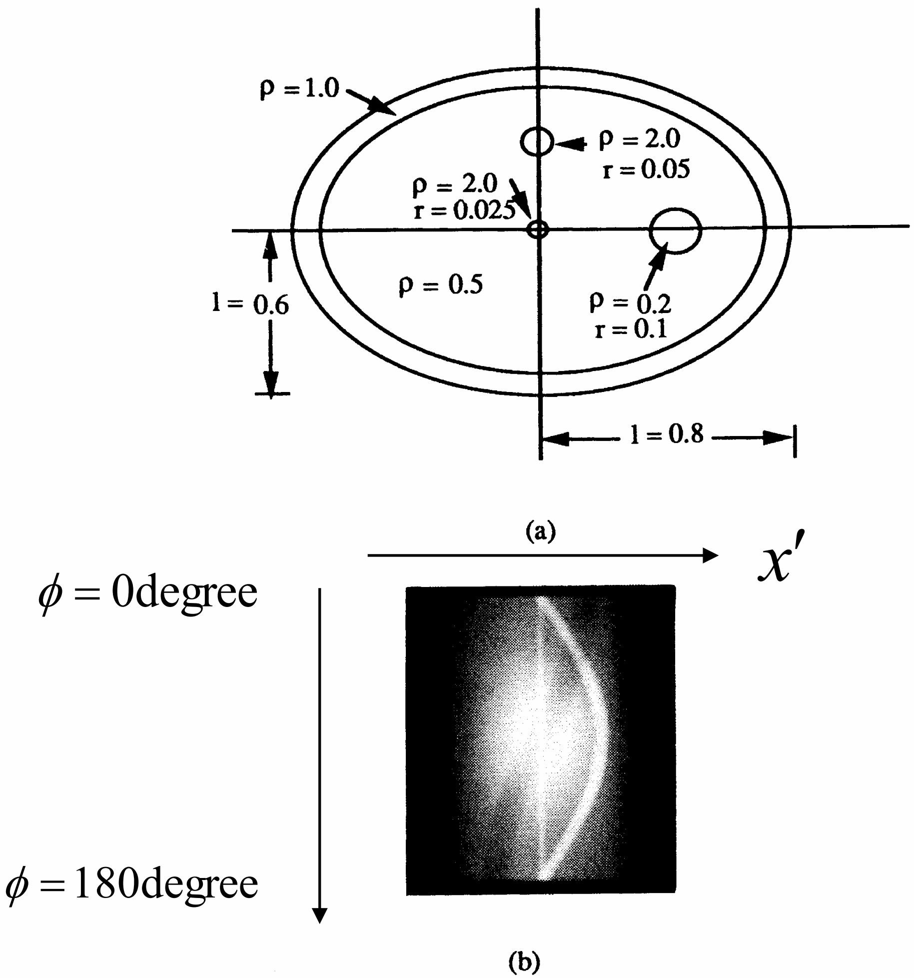
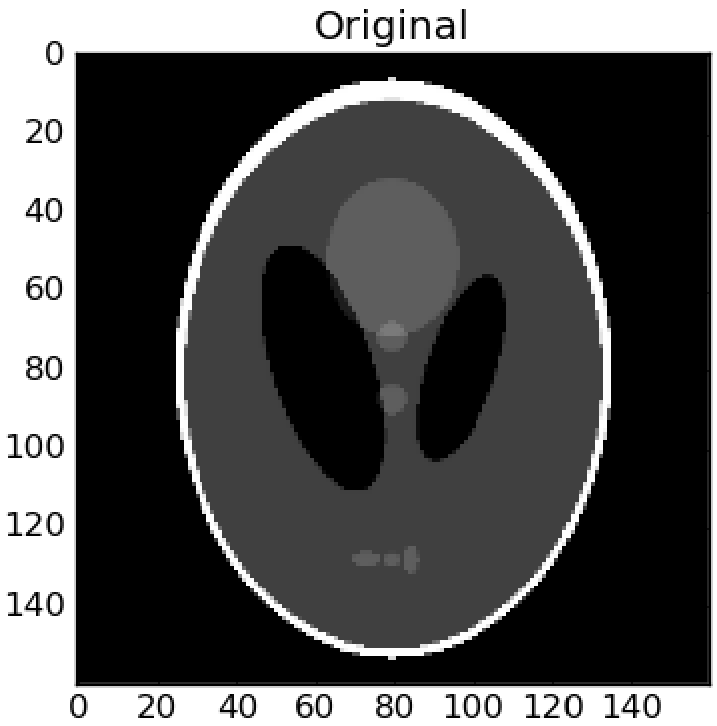


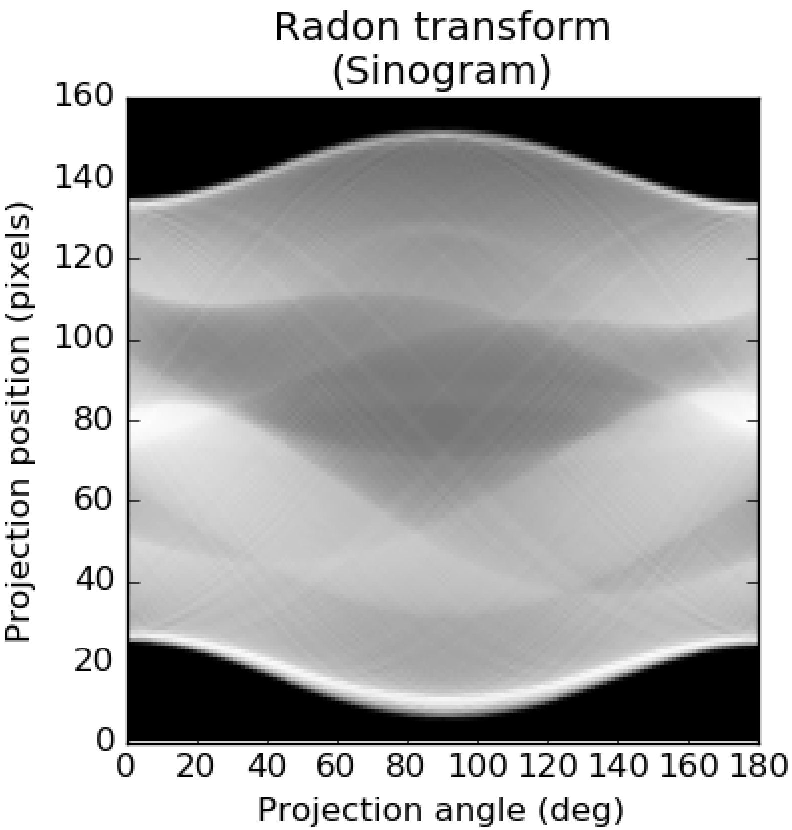
Figure 3-3 (a) Phantom with several distinct objects at various positions; (b) correspondi sinogram.

Sinogram is a 2-D function representing the original function f(x,y) into the projection data space.

Sinogram is the basic data format for reconstruction

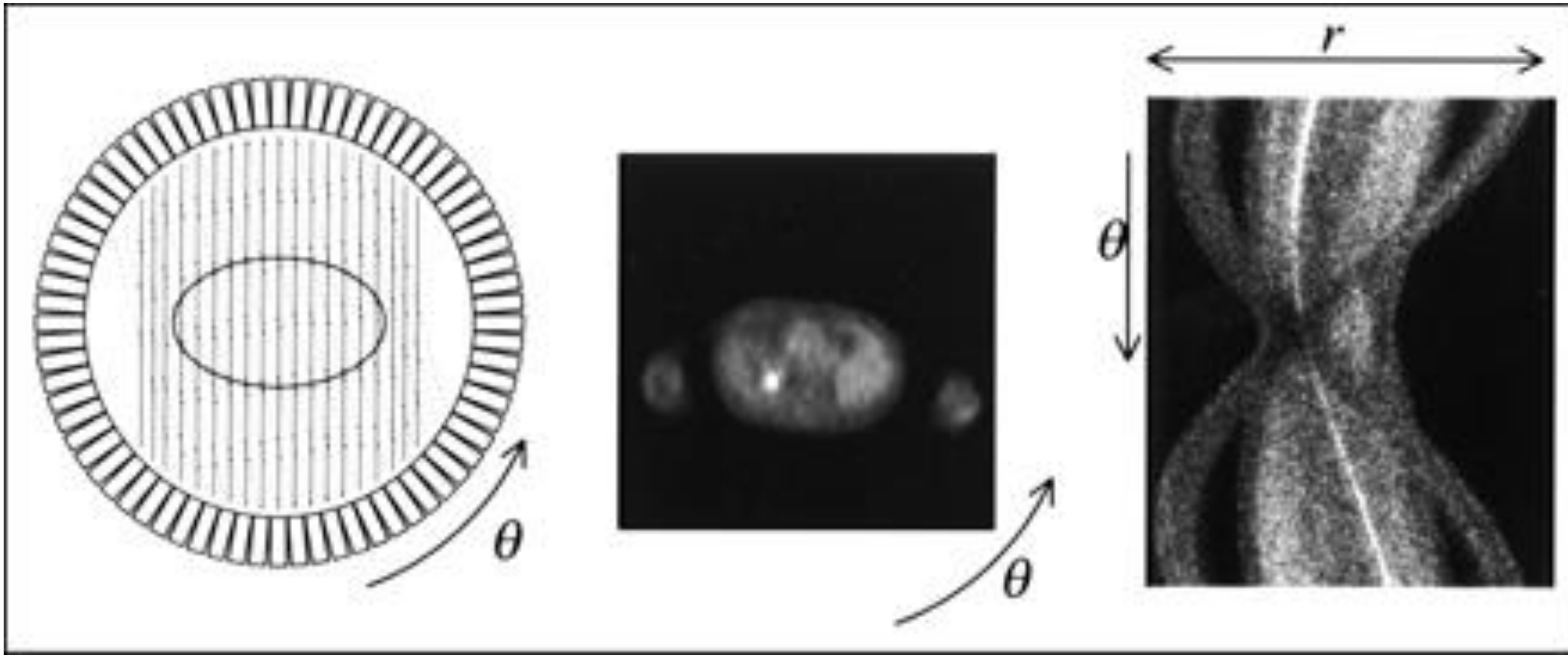
Radon Transform and Sinogram





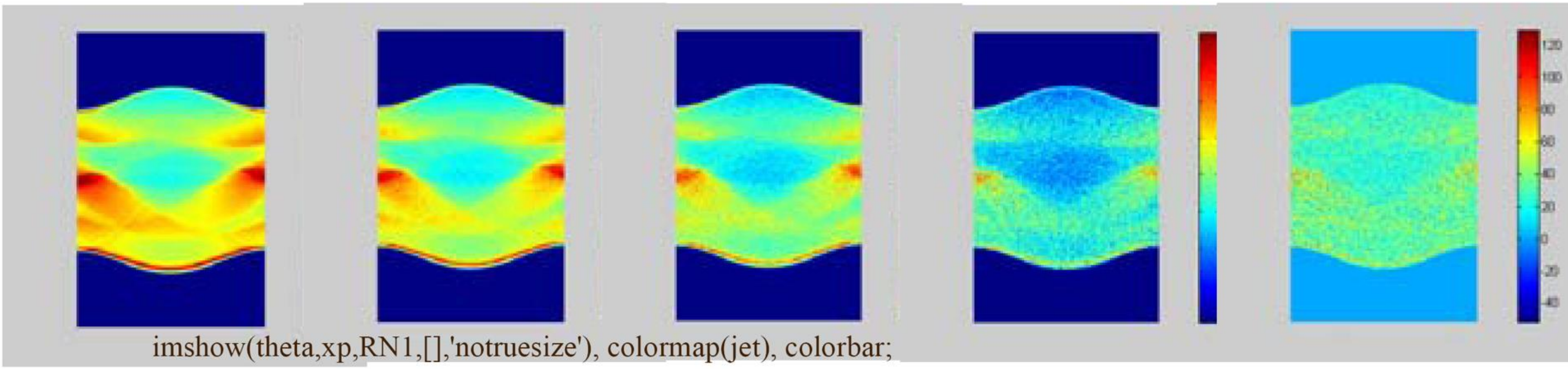
http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

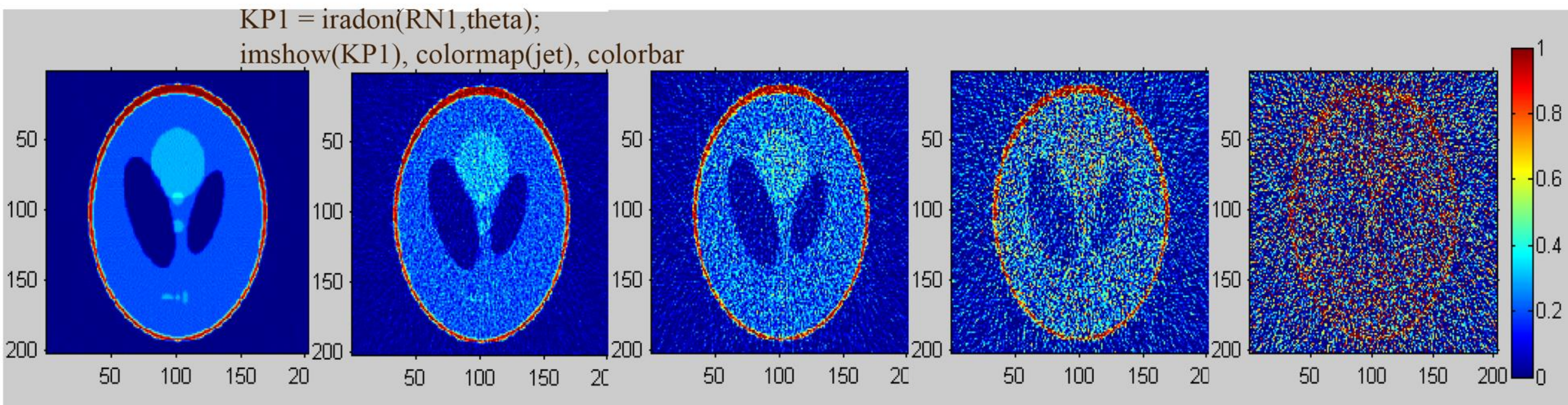
Radon Transform and Sinogram



http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

Matlab Examples





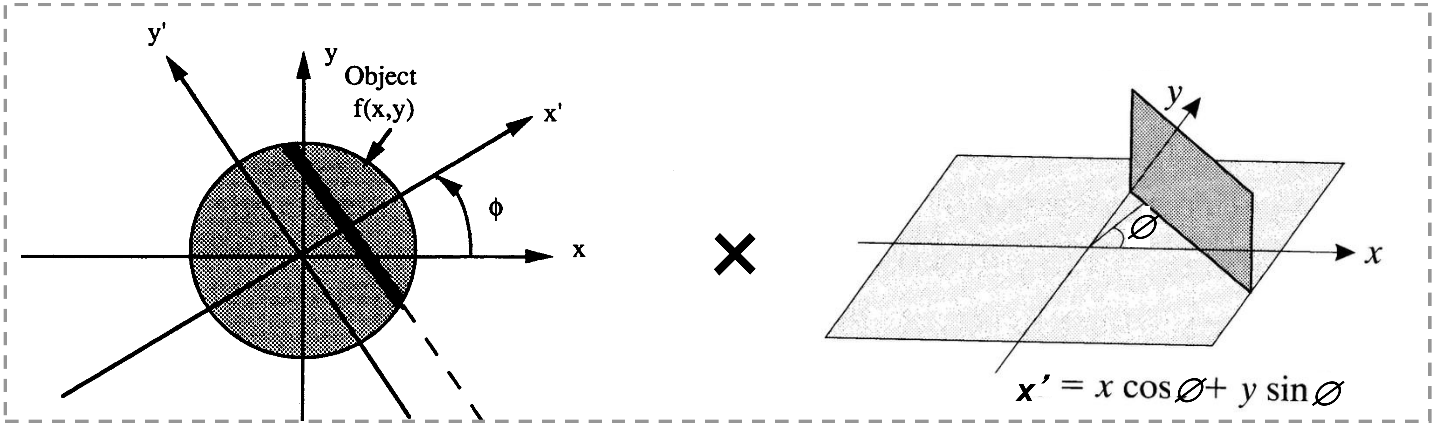
sd=0 sd=0.05 sd=0.10 sd=0.20 sd=0.50

What Exactly Does Projection Data Tell Us?

The Central-Slice Theorem

(or Projection-Slice Theorem)

Radon Transform



2-D integral

The value of the projection function $p_{\phi}(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The **Radon transform**, which produces the projection data, is defined as

$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Central Slice Theorem

Let's consider the 2D Fourier transform (FT) of an arbitrarily given function

$$F(u,v) = \int\int f(x,y) \cdot e^{-2\pi i (ux + vy)}\, dx\, dy\,. \qquad (1)$$

In polar coordinates and within the spatial-frequency domain, let

$$u = \rho \cos \phi \qquad \text{and} \qquad v = \rho \sin \phi,$$

Recall that
$\mathcal{F}[f(x-\tau)] = \mathcal{F}[f(x)]e^{-2\pi i\omega\tau}$,
and
$\mathcal{F}[f(x)] = 1$,
So
$\mathcal{F}[\delta(x-\tau)] = \mathrm{e}^{-2\pi i\omega\tau}$.

then (1) becomes

$$F(\rho, \phi) = \int\int f(x,y) \cdot e^{-2\pi i \rho (x \cos \phi + y \sin \phi)}\, dx\, dy\,. \qquad (2)$$

If we treat ($x \cos \phi + y \sin \phi$) as a constant, then $exp\,[\,-i\,2\pi\,\rho\,(x \cos \phi + y \sin \phi)]$ could be written as the Fourier transform of a shifted delta function. Let's write the complex exponential as the FT of a δ function,

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \mathcal{F}_{1d,x'}\{\delta[x' - (x \cos \phi + y \sin \phi]\}\; dx \cdot dy\,,$$

and

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \{\int_{x'} \delta[x' - (x \cos \phi + y \sin \phi] \cdot e^{-2\pi i \rho x'}\, dx'\} \cdot dx\, dy\,. \qquad (3)$$

Central Slice Theorem

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \{\int_{x'} \delta[x' - (x \cos \phi + y \sin \phi] \cdot e^{-2\pi i \rho x'} dx' \} \cdot dx\, dy\,. \qquad (3)$$

At this point, we can change the order of the integration operators as

$$F(\rho, \phi) = \int_{x'} [\int_y \int_x f(x,y)\, \delta(x \cos \phi + y \sin \phi - x')\, dx\, dy]\, e^{-2\pi i \rho x'} dx'\,. \qquad (4)$$

Recall that the Radon transform is defined as,

$$p(\phi, x') = \int_x \int_y [f(x,y) \cdot \delta(x \cos \phi + y \sin \phi - x')] dx\, dy\,. \qquad (5)$$

We take the 1-D Fourier transform of $p(\phi, x')$ as

$$F\{p(\phi, x')\} = \int_{x'} [\int_y \int_x f(x,y)\, \delta(x \cos \phi + y \sin \phi - x')\, dx\, dy]\, e^{-2\pi i \rho x'} dx'\,, \qquad (6)$$

which is exactly what we wrote for $F(\rho, \phi)$ above!

Therefore, we have proved that

$$\mathcal{F}_{1d,x'}[p(\phi, x')] = F(\rho, \phi).$$

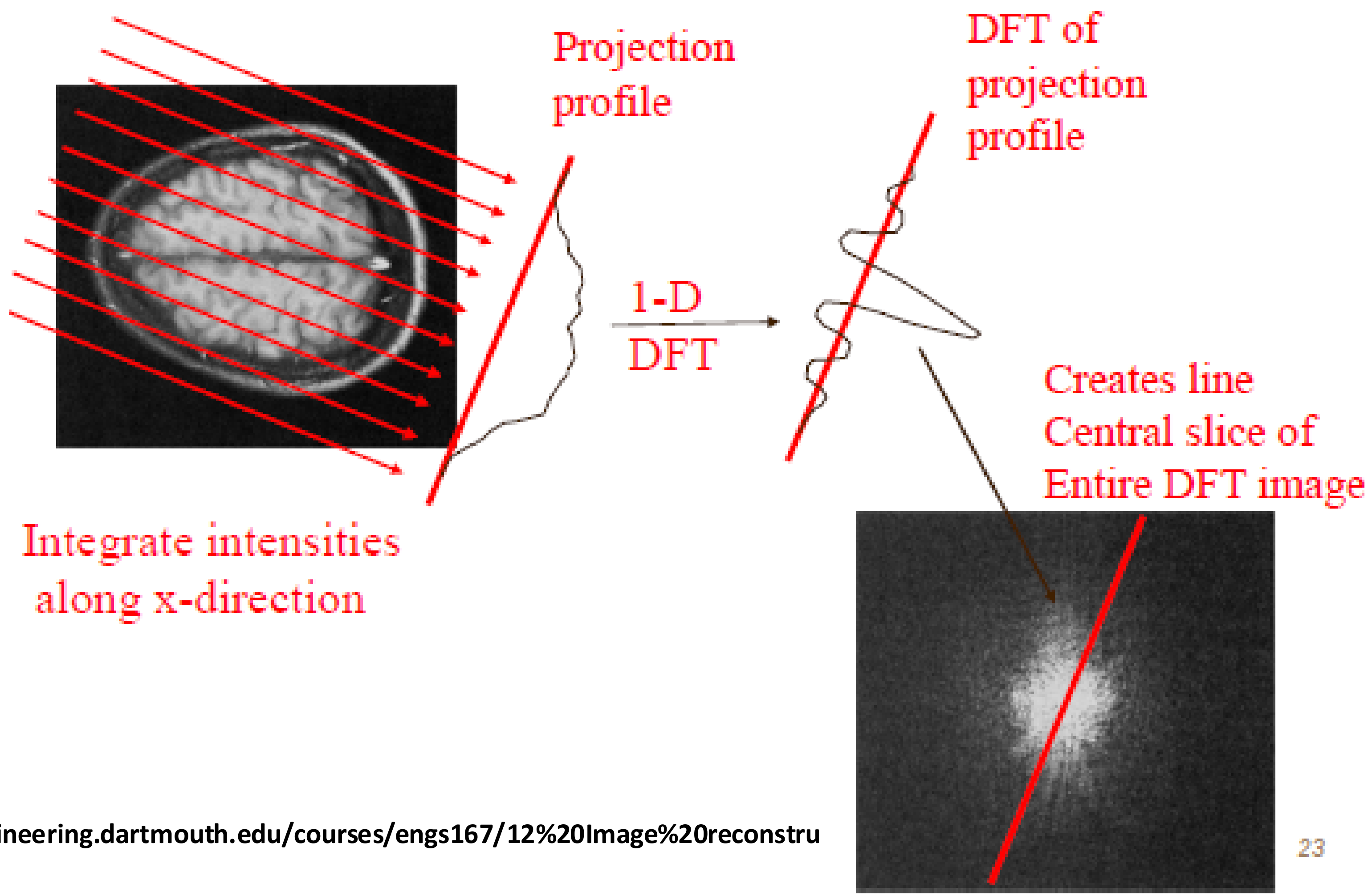
$F(\rho, \phi)$ is defined in Eq. (2).

Projection Slice Theorem or Central Slice Theorem

An alternative proof can be found on Page 77 in Foundations of Medical Imaging, by Z. H. Cho.

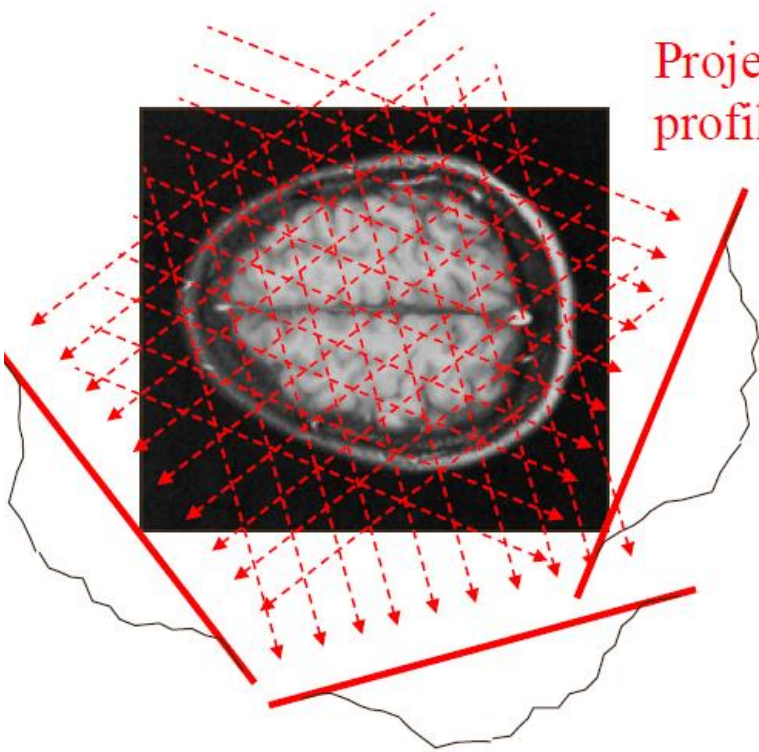
Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem

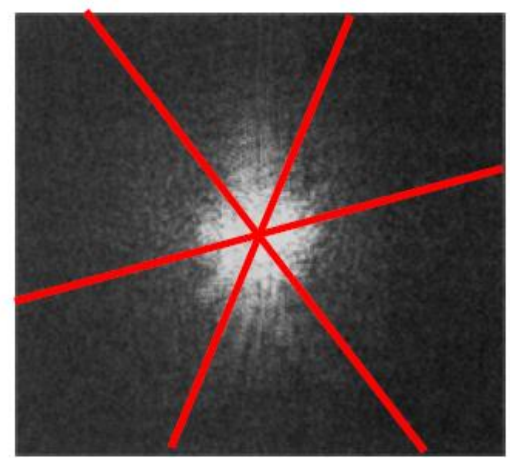


Projection profiles

Integrate intensities along x-direction

1-D DFTs

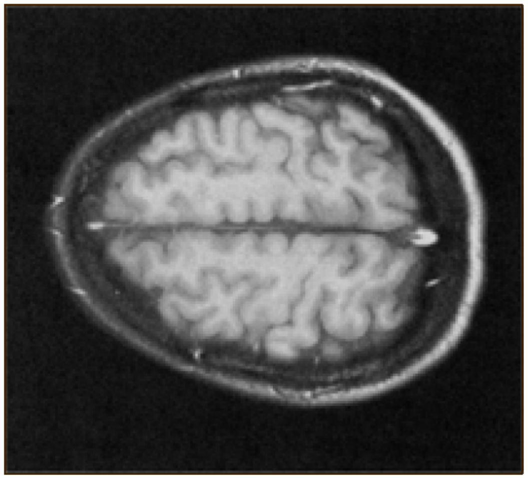
Create lines in central slice of entire DFT image



The more angles used, the better the Fourier space image is filled

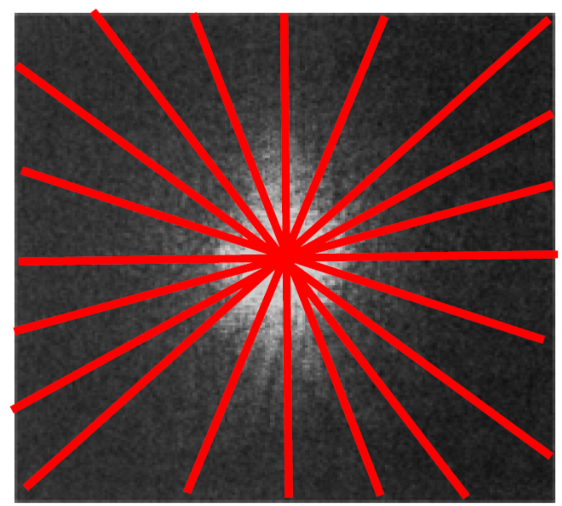
http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem



DFT image represents integration of original projections DFT transformed and summed together.

1-D DFTs at each projection

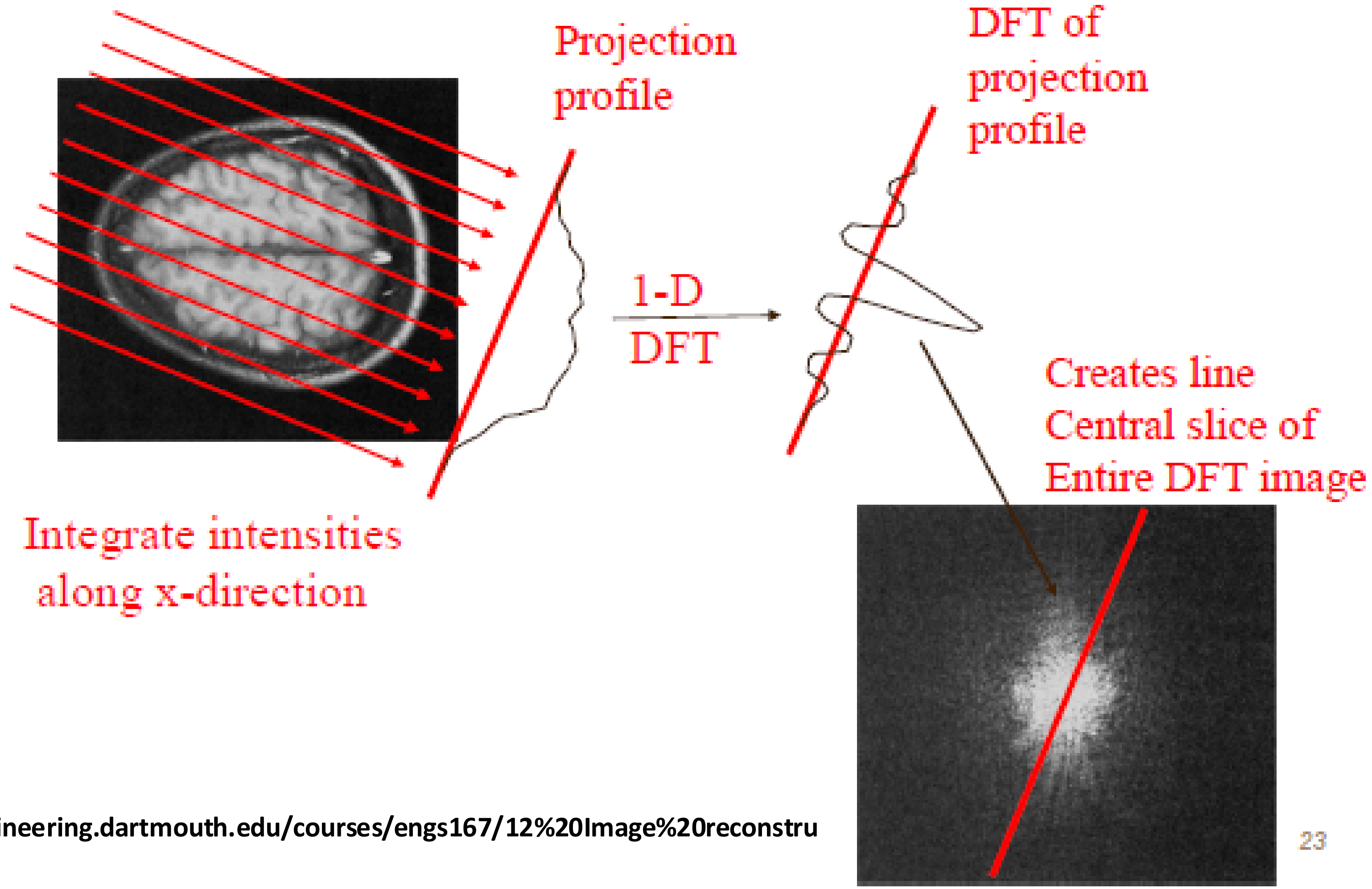


This is the fast way to create the DFT image from projection data. The more projections taken, the more complete the sampling.

http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

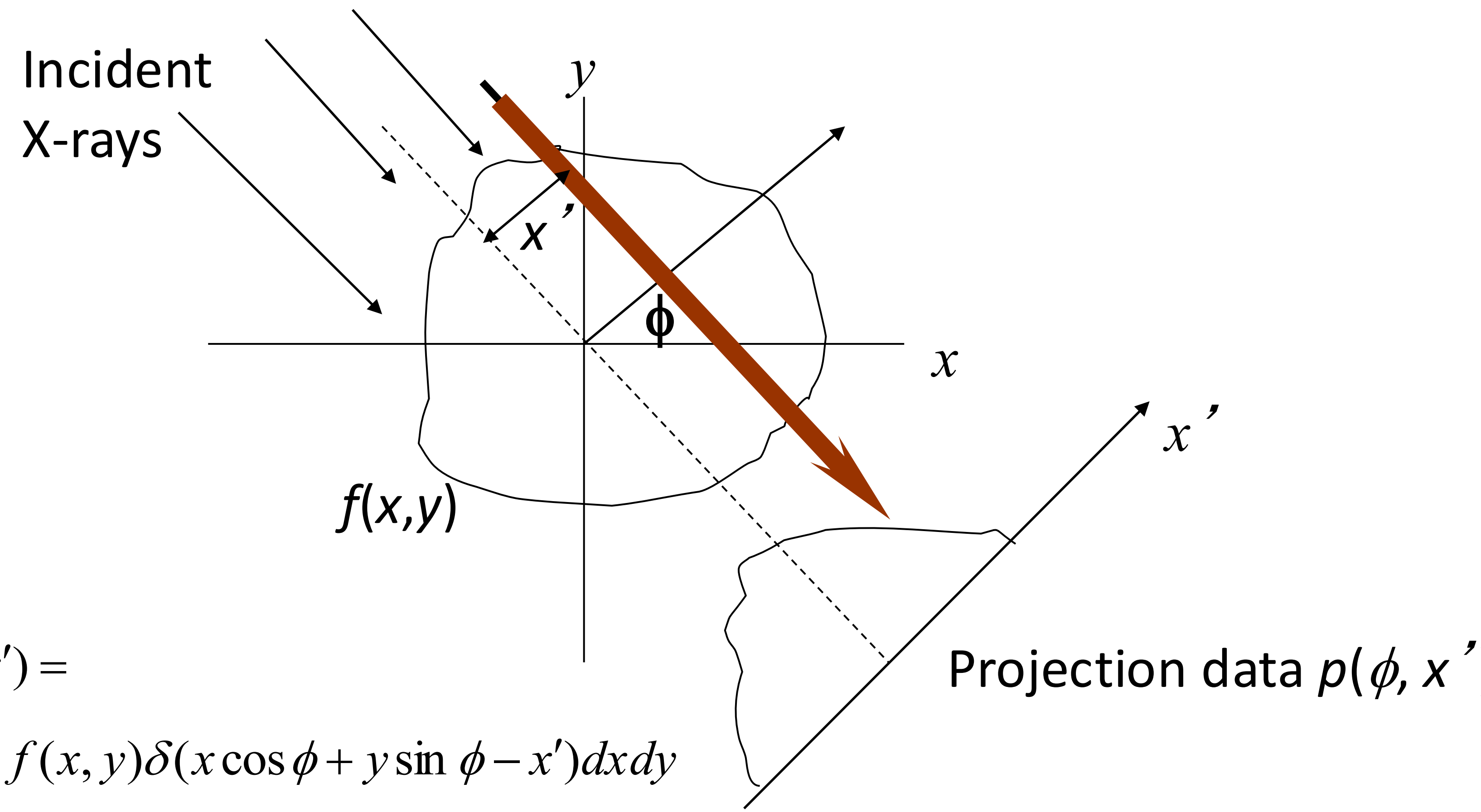
Central Section Or Projection Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,, x')] = F(\rho,\ \phi\,).$$

So, in words, the Fourier transform of a projection at an angle ϕ gives us a line in the polar Fourier space at the same angle ϕ.

The central slice theorem is the key to understanding reconstructions from projection data.

Radon Transform and Projection Data (Revisited)

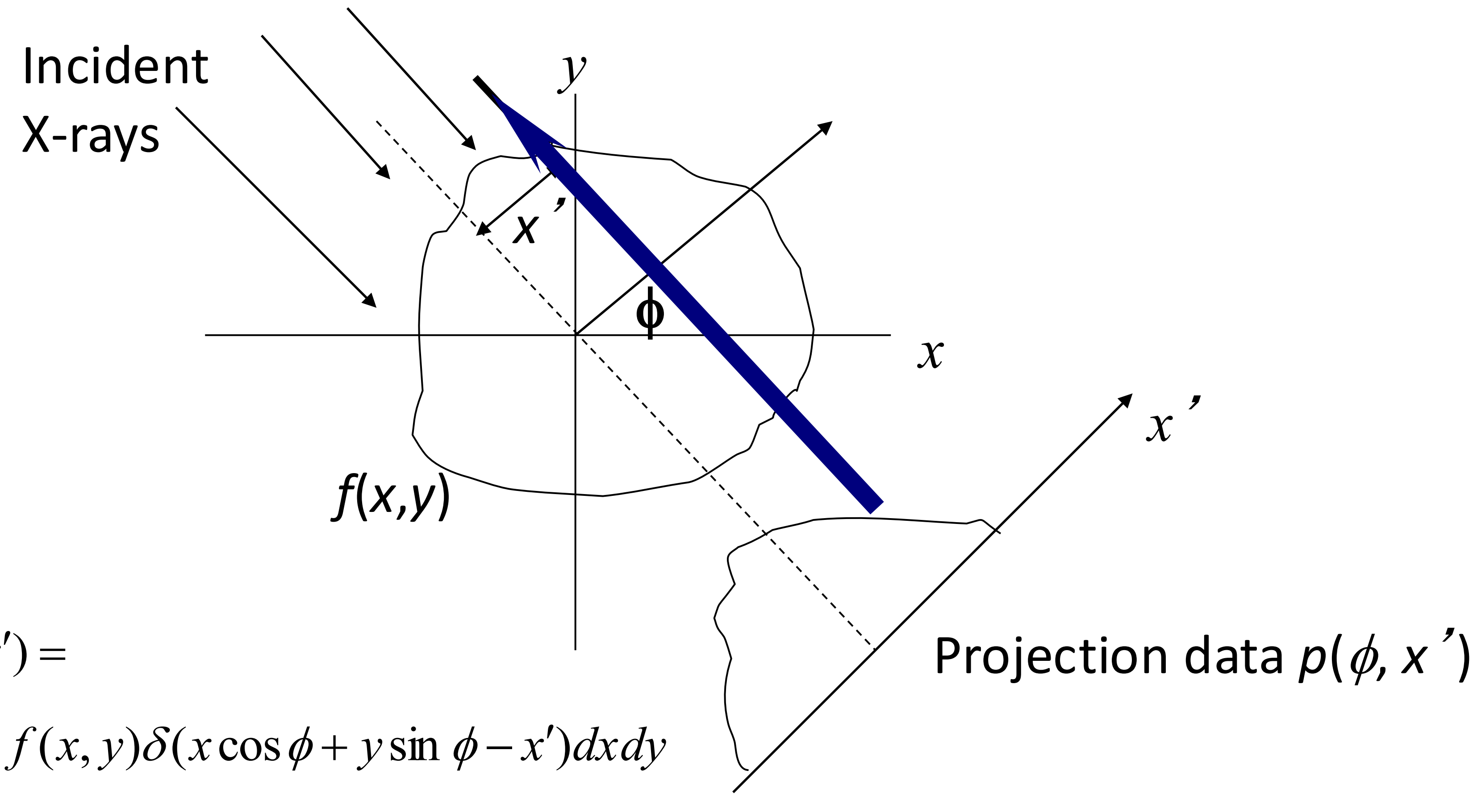


$$p(\phi, x') =$$
$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Simple Backprojection

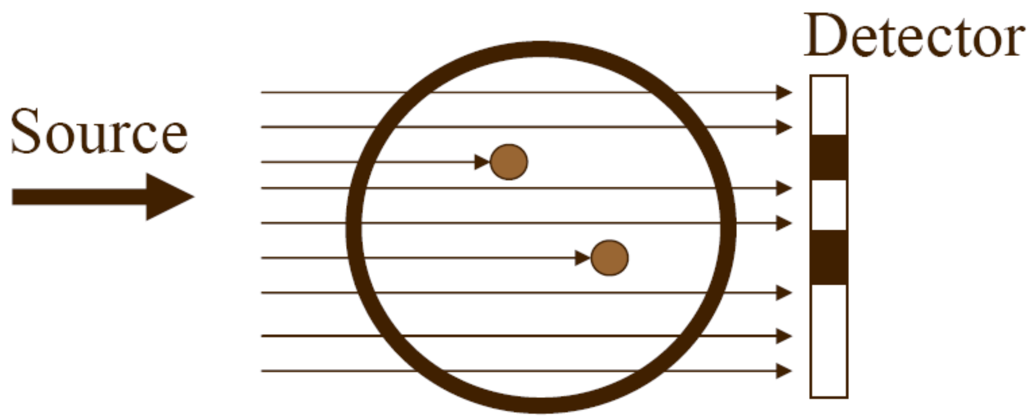
as a Way to Restore the Original Image?

Simple Backprojection



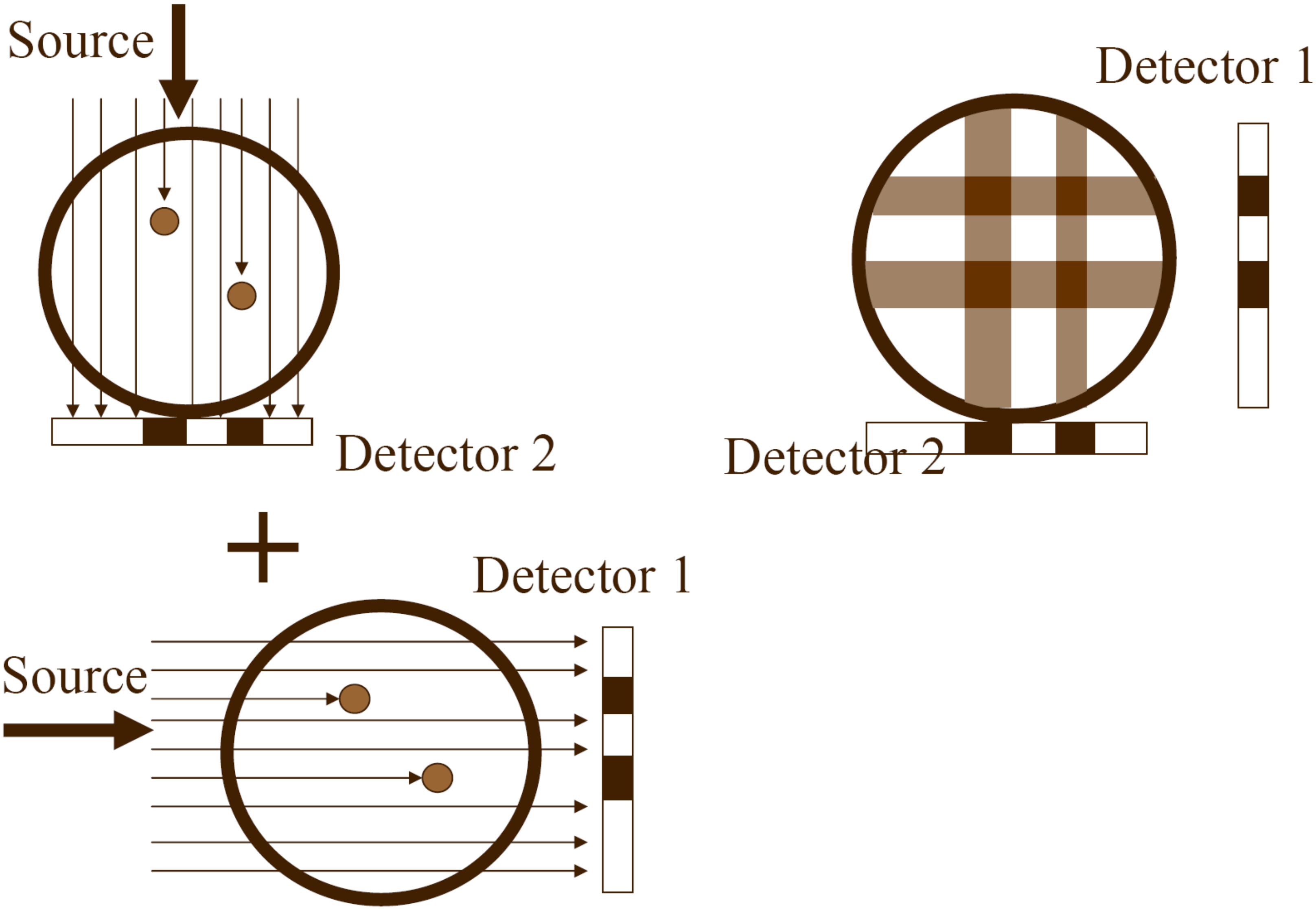
$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Simple Backprojection

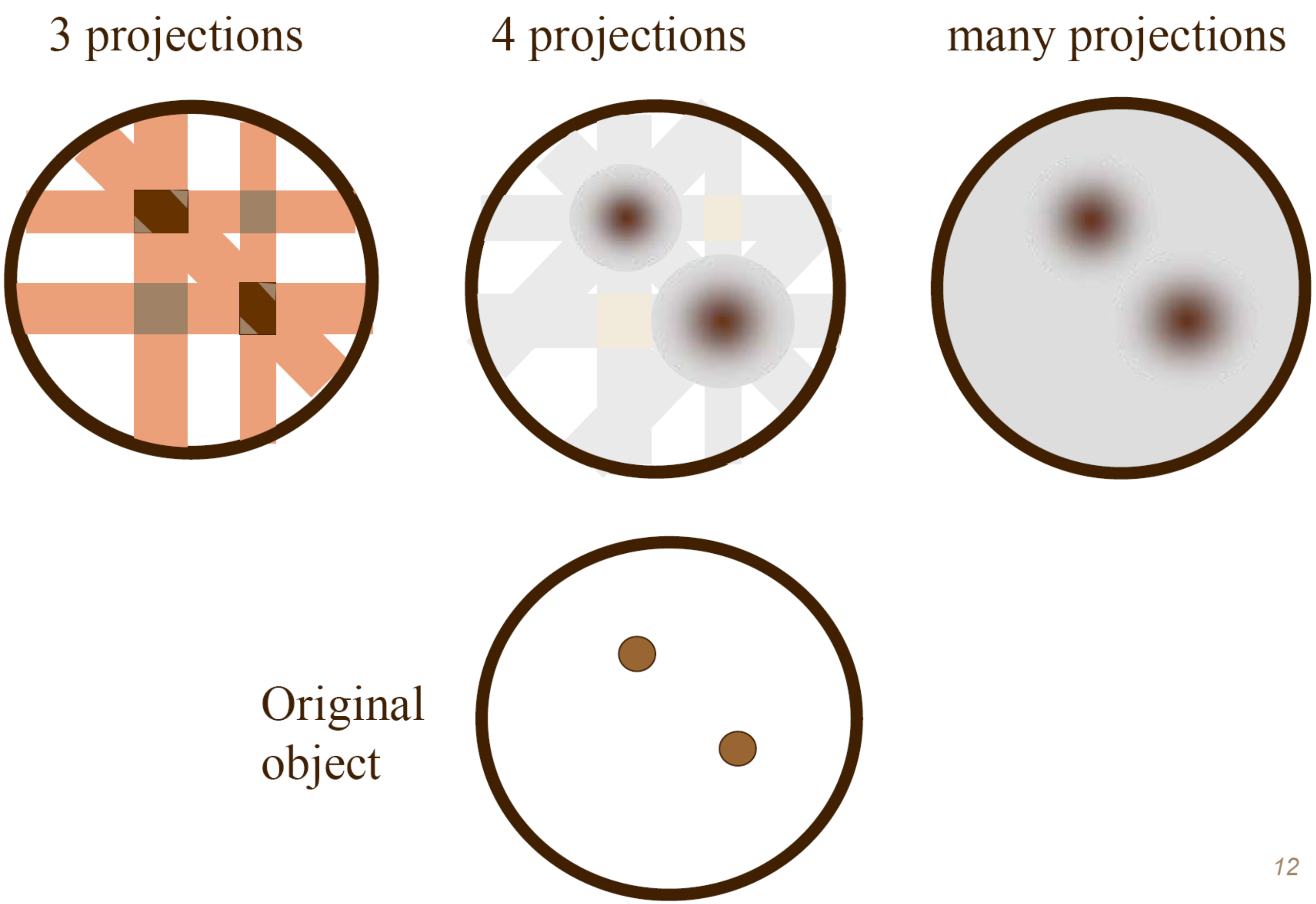


inverse = 1 back projection

Simple Backprojection



Simple Backprojection



Simple Backprojection

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{simple\ back\text{-}projection}(x,y) = \int b_\phi(x,y) \, d\phi$$

$$f_{\text{simple back-projection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$

Simple Back-projection and the 1/r Blurring

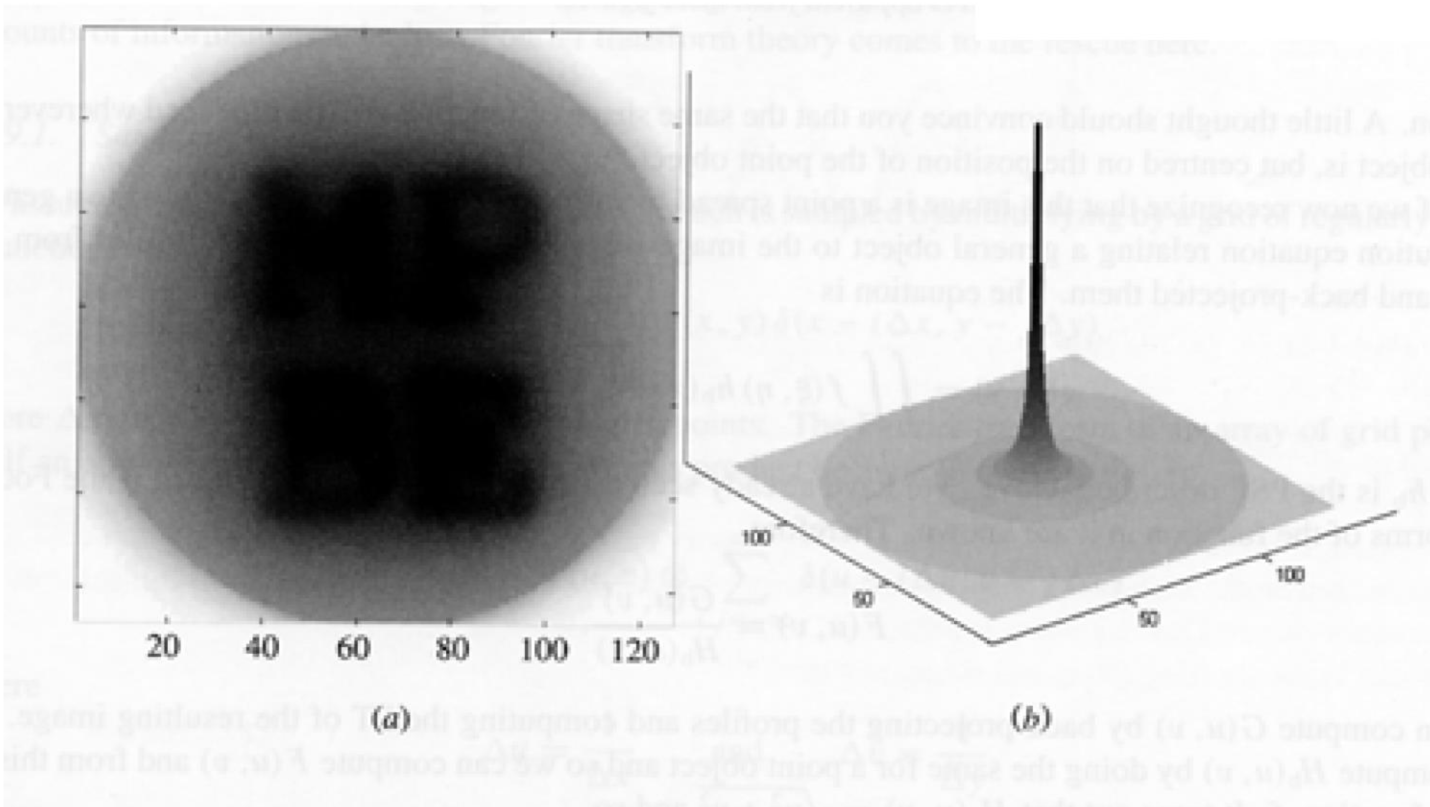


Figure 11.14. *(a) Image produced by back-projecting the sinogram given in figure 11.12(b); (b) the response to a point object in the middle of the field of view.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Impulse Response Function of the Simple Backprojection Operator

$$h_b(r) = 1/r$$

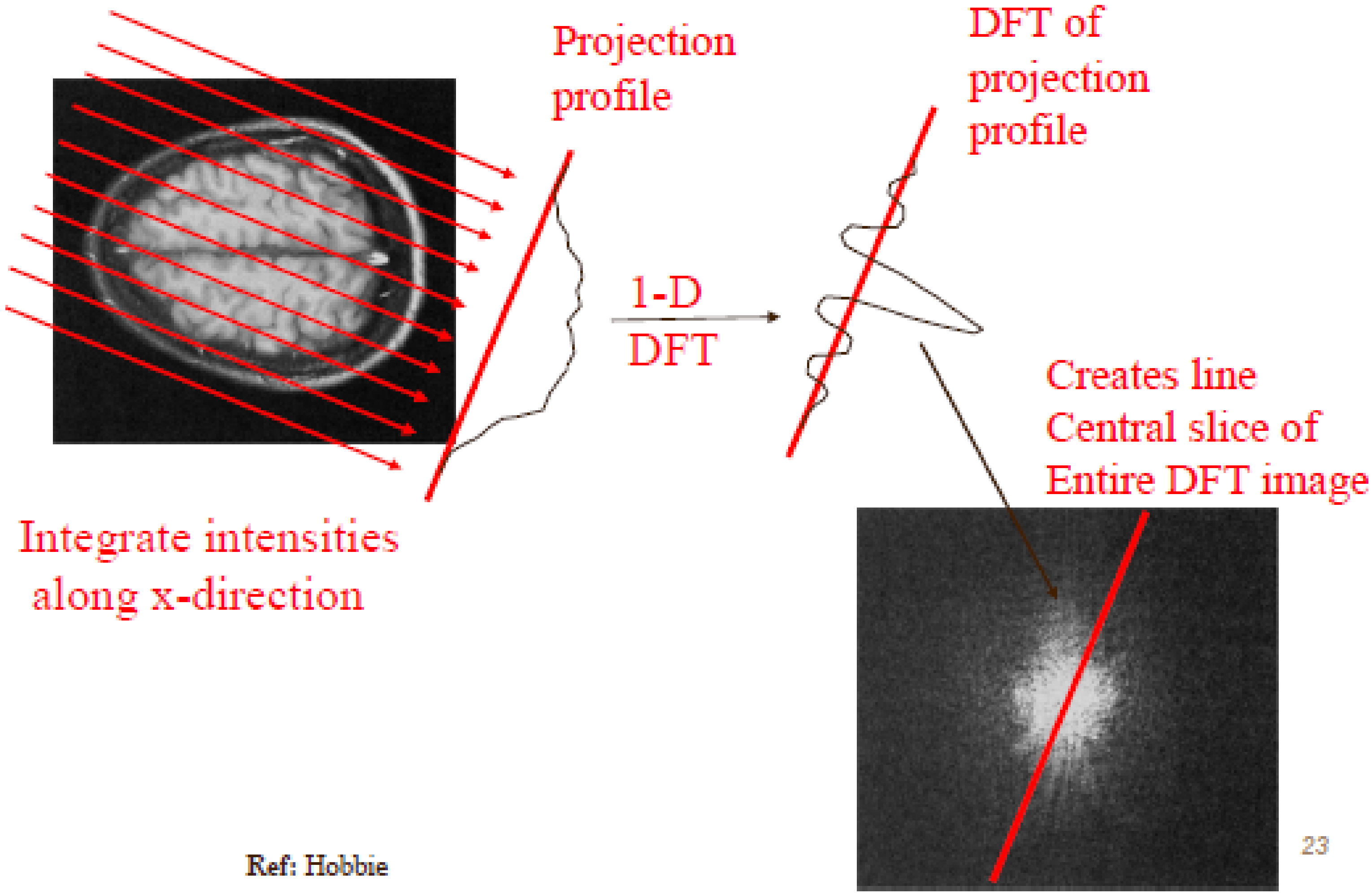
$$f_b(x,y) = f(x,y) * 1/r$$

$$F_b(\rho,\phi) = F(\rho,\phi)/\rho\,, \qquad \text{since } F\{1/r\} = 1/\rho$$

Backprojected image is blurred by convolution with $1/r$.

Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Simple Back-projection and the 1/r Blurring

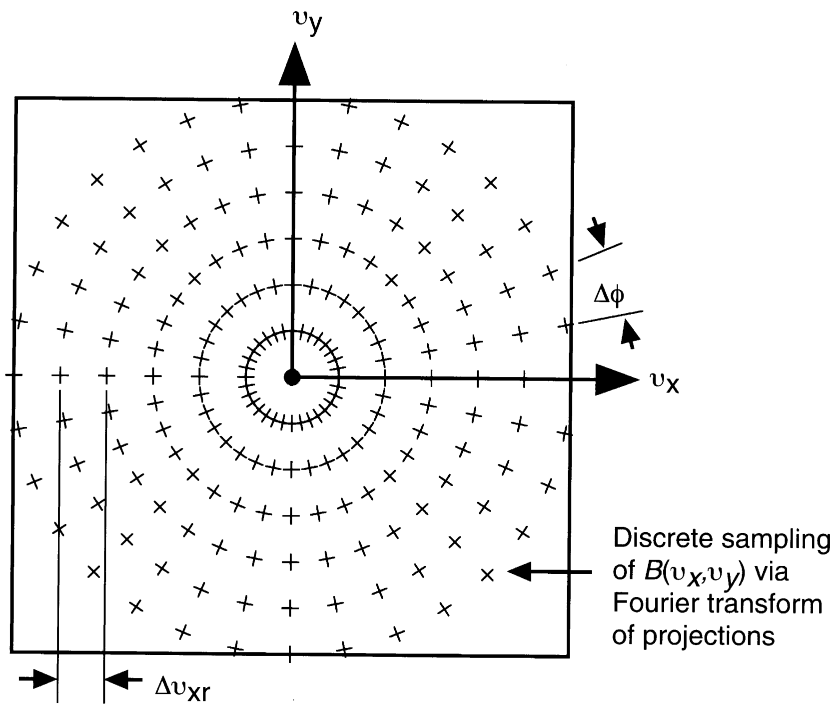


FIGURE 18 The discrete sampling pattern of $F(\upsilon_x,\upsilon_y)$ contained in $B(\upsilon_x,\upsilon_y)$, resulting from the use of discretely sampled projections.

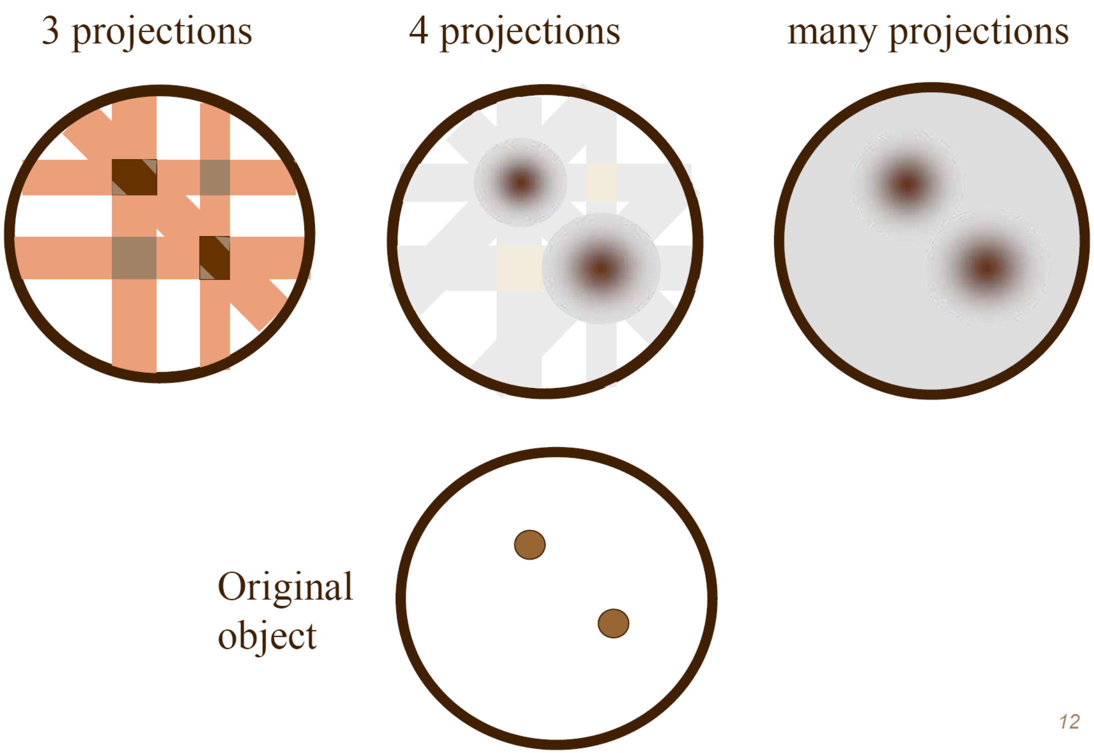
The nature of the 1/r blurring:
Radon transform produces equally spaced radial sampling in the Fourier domain.

2-D Inverse Radon Transform

Simple Backprojection (Revisited)

Reconstruction with simple backprojection

$$\hat{\boldsymbol{f}}_{\text{simple backprojection}}(x, y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$



From Simple Backprojection to Inverse Radon Transform

Simple backprojection

Result from a back projection from a single angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos \phi + y \sin \phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$\hat{f}_{\text{simple backprojection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Inverse Radon transform

$$\hat{f}_{\text{Inverse Radon transform}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} F^{-1}\{F_{1d,x'}[p_\phi(x')] \cdot |w|\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Inverse Radon Transform

Radon transform

$$\hat{\boldsymbol{f}}_{\text{simple backprojection}}(x,y)=\int_{0}^{\pi}d\phi\int_{-\infty}^{\infty}p_{\phi}(x')\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

Inverse Radon transform

$$\hat{\boldsymbol{f}}_{\text{inverse Radon transform}}(x,y)=\int_{0}^{\pi}d\phi\int_{-\infty}^{\infty}F^{-1}\{F[p_{\phi}(x')]\cdot|\omega|\}\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

$$\hat{f}(x,y)=\hat{f}(r,\theta)=\beta\mathscr{H}\{p_{\phi}(x')\}$$

The simple back-projection operator

$$=\beta\mathscr{H}\{\mathscr{R}[f(x,y)]\}$$

$$=\mathscr{R}^{-1}\{\mathscr{R}[f(x,y)]\}$$

where

$$\mathscr{H}\{p_{\phi}(x')\}=\mathscr{F}_1^{-1}[|\omega|]*p_{\phi}(x')$$

The inverse Radon transform can be represented as a filtering process followed by a back-projection operation.

Is there something missing in this discussion?, such as

Noise in data?

Possible artifacts?

Filtered Back-projection – The Optimum Filter

Have we forgot something? What about the noise in the projection data?

Where is the noise coming from?

In reality, the true projection is

$$\dot{p}_\phi(x') = p_\phi(x') + n_\phi(x')$$

Review of Fourier Transform and Filtering Spectral Filtering

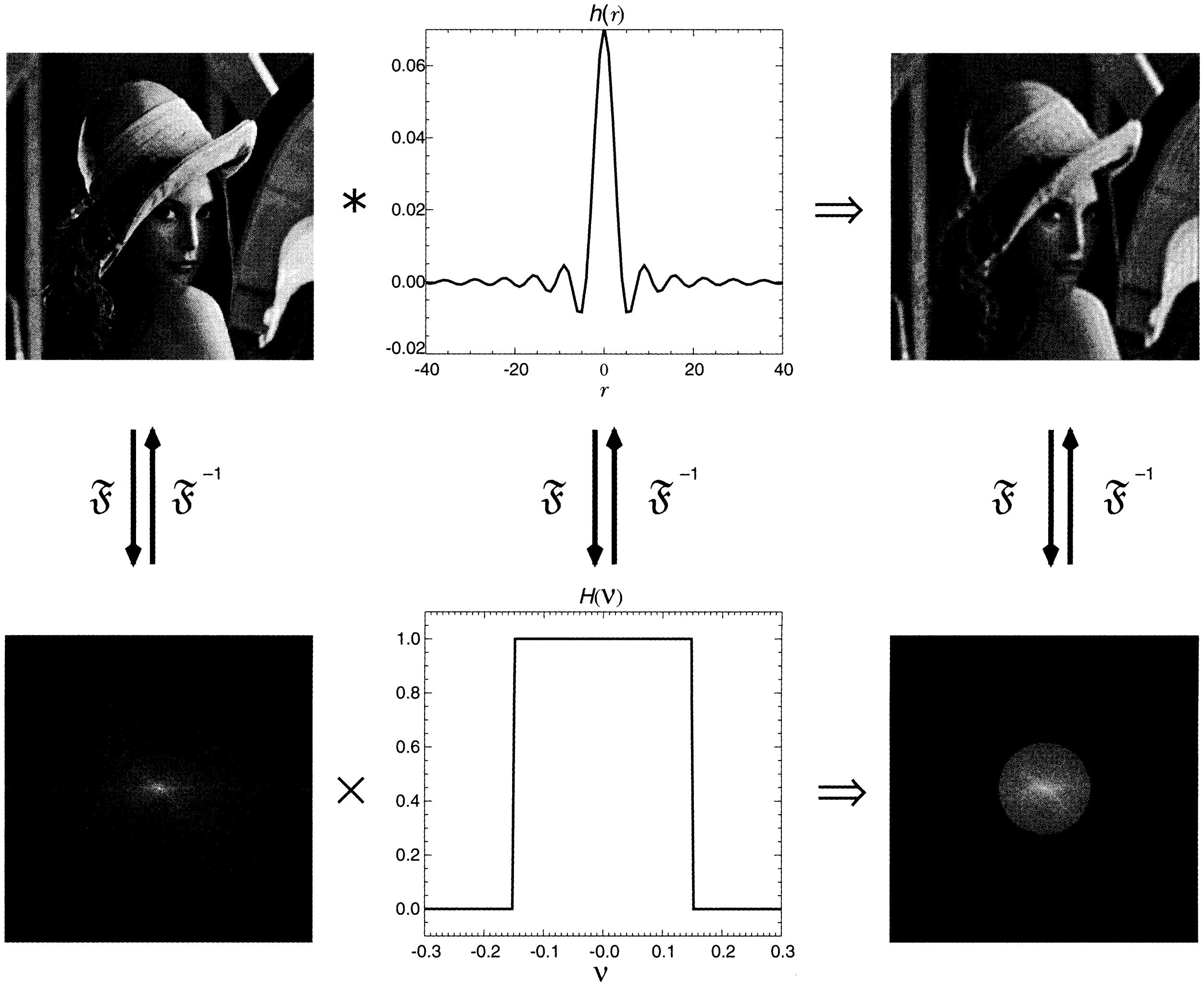


FIGURE 3 Ideal lowpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column are the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2 + y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Review of Fourier Transform and Filtering
Spectral Filtering

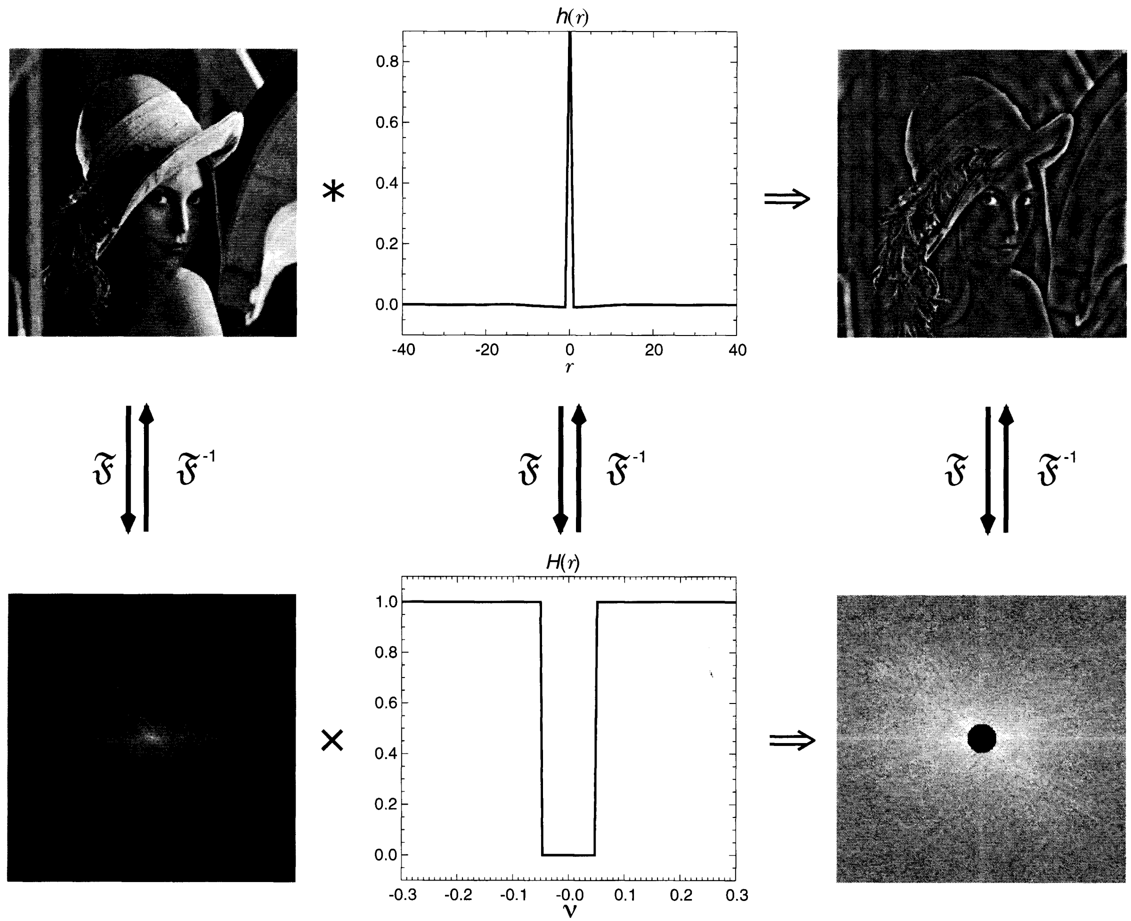
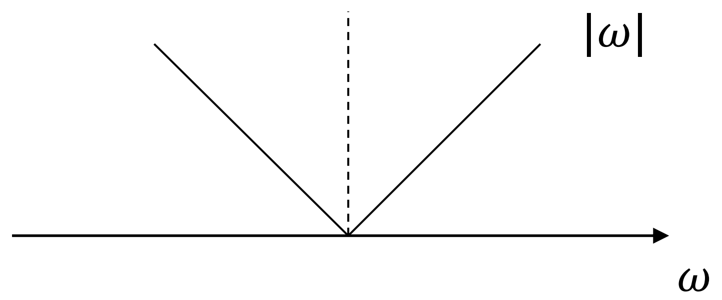


FIGURE 4 Ideal highpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column shows the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2 + y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Filtered Back-projection

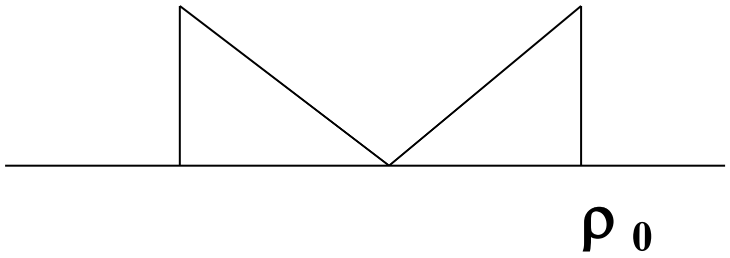
Ideal filter



The Ram-Lak filter

$$H_{\mathrm{RL}}(\omega) = \begin{cases} |\omega|, & (|\omega| \leq 2\pi B) \\ 0, & (\text{otherwise}) \end{cases}$$

Ram-Lak filter



From Inverse Radon Transform to Filtered Backprojection (FBP)

Simple backprojection

$$\hat{\boldsymbol{f}}_{\text{simple backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Inverse Radon transform

$$\hat{\boldsymbol{f}}_{\text{inverse Radon transform}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot |\omega|\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Filtered Backprojection (FBP)

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot |\omega| \cdot H'(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

or

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

From Inverse Radon Transform to Filtered Backprojection (FBP)

Filtered Backprojection (FBP)

$$\hat{\boldsymbol{f}}_{\mathrm{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot |\omega| \cdot H'(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx' ,$$

or

$$H(\omega) = |\omega| \cdot H'(\omega)$$

$$\hat{\boldsymbol{f}}_{\mathrm{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

$$= \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi^*(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx' ,$$

where the **filtered projection**, $p_\phi^*(x')$, is defined as

$$p_\phi^*(x') \equiv F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \equiv p_\phi(x') * h(x')$$

and $H(\omega) \equiv F[h(x')]$ is the filter defined in spatial frequency domain.

Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx',$$

where $H(\omega) = |\omega| \cdot H'(\omega)$.

The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot h(x\cos\phi + y\sin\phi - x')dx',$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

Simple and Filtered Back-projection

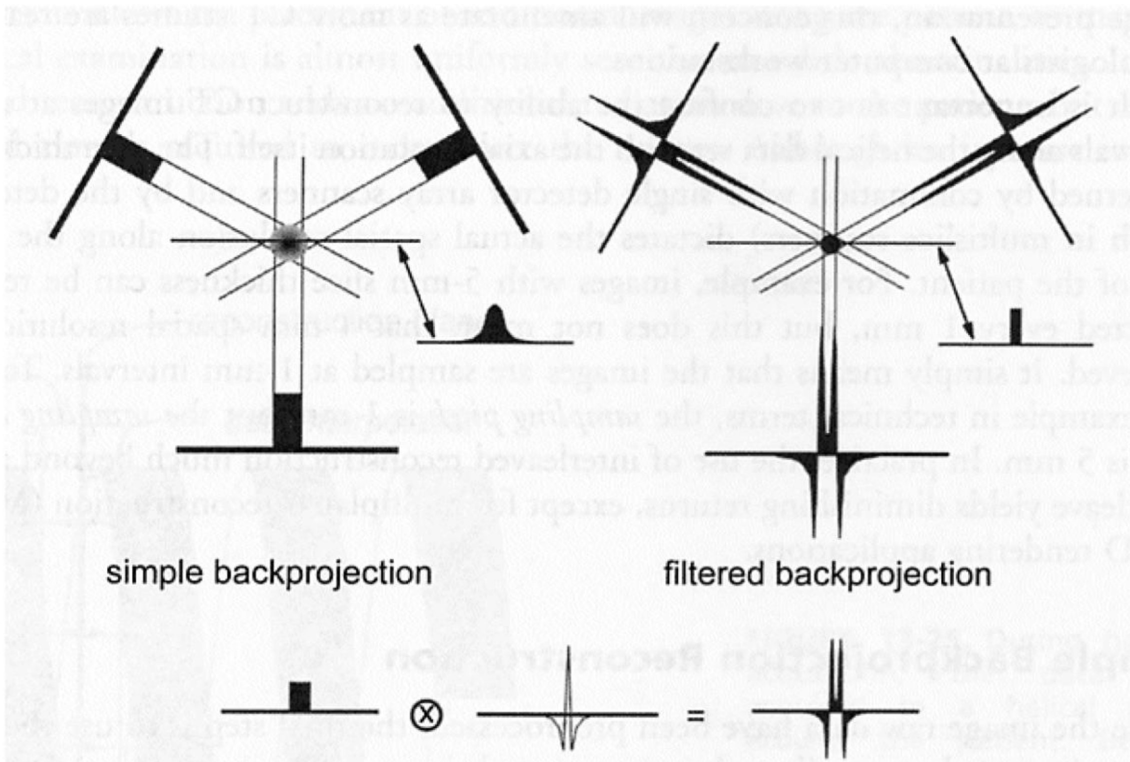
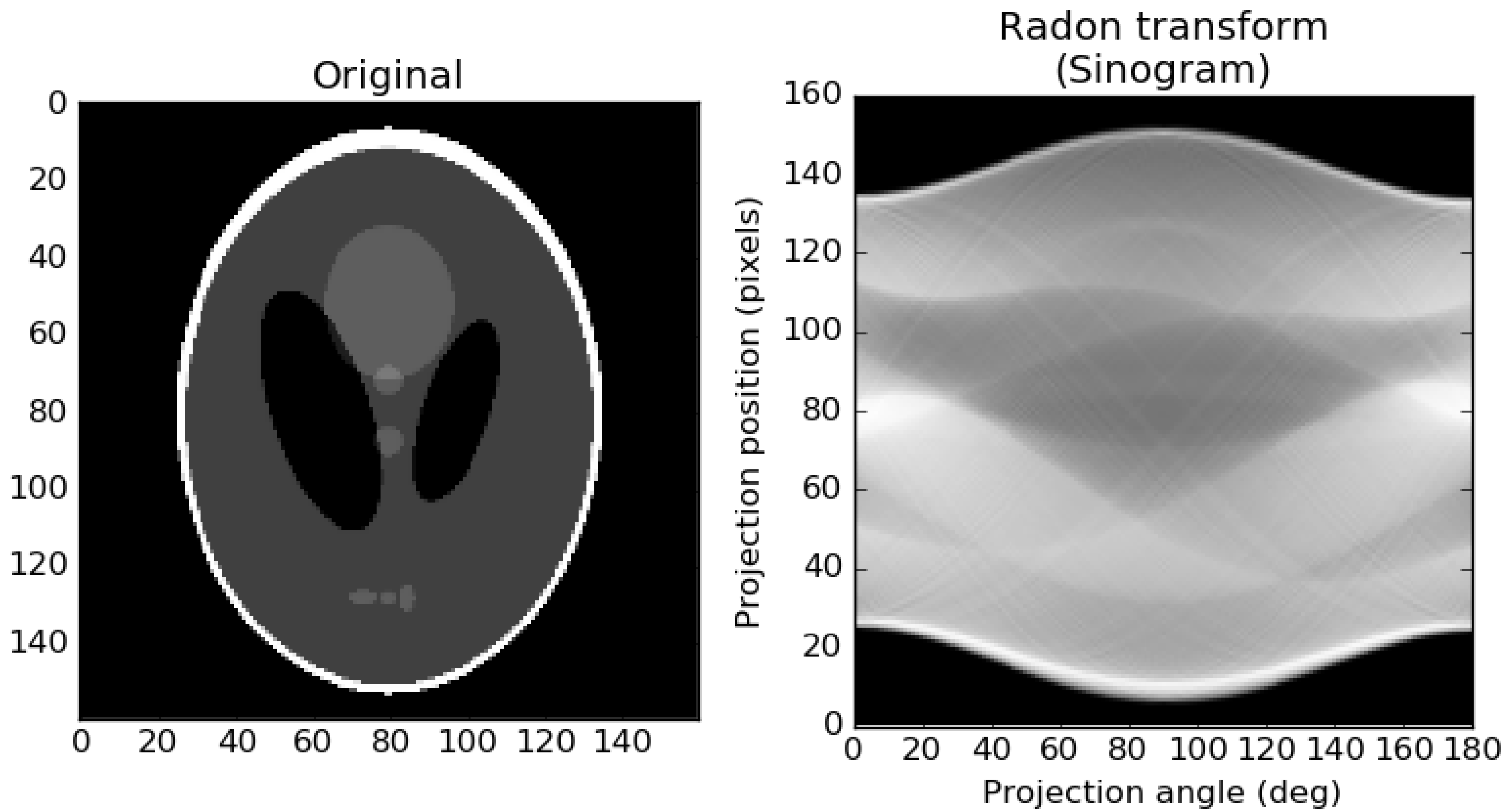


FIGURE 13-28. Simple backprojection is shown on the left; only three views are illustrated, but many views are actually used in computed tomography. A profile through the circular object, derived from simple backprojection, shows a characteristic 1/r blurring. With filtered backprojection, the raw projection data are convolved with a convolution kernel and the resulting projection data are used in the backprojection process. When this approach is used, the profile through the circular object demonstrates the crisp edges of the cylinder, which accurately reflects the object being scanned.

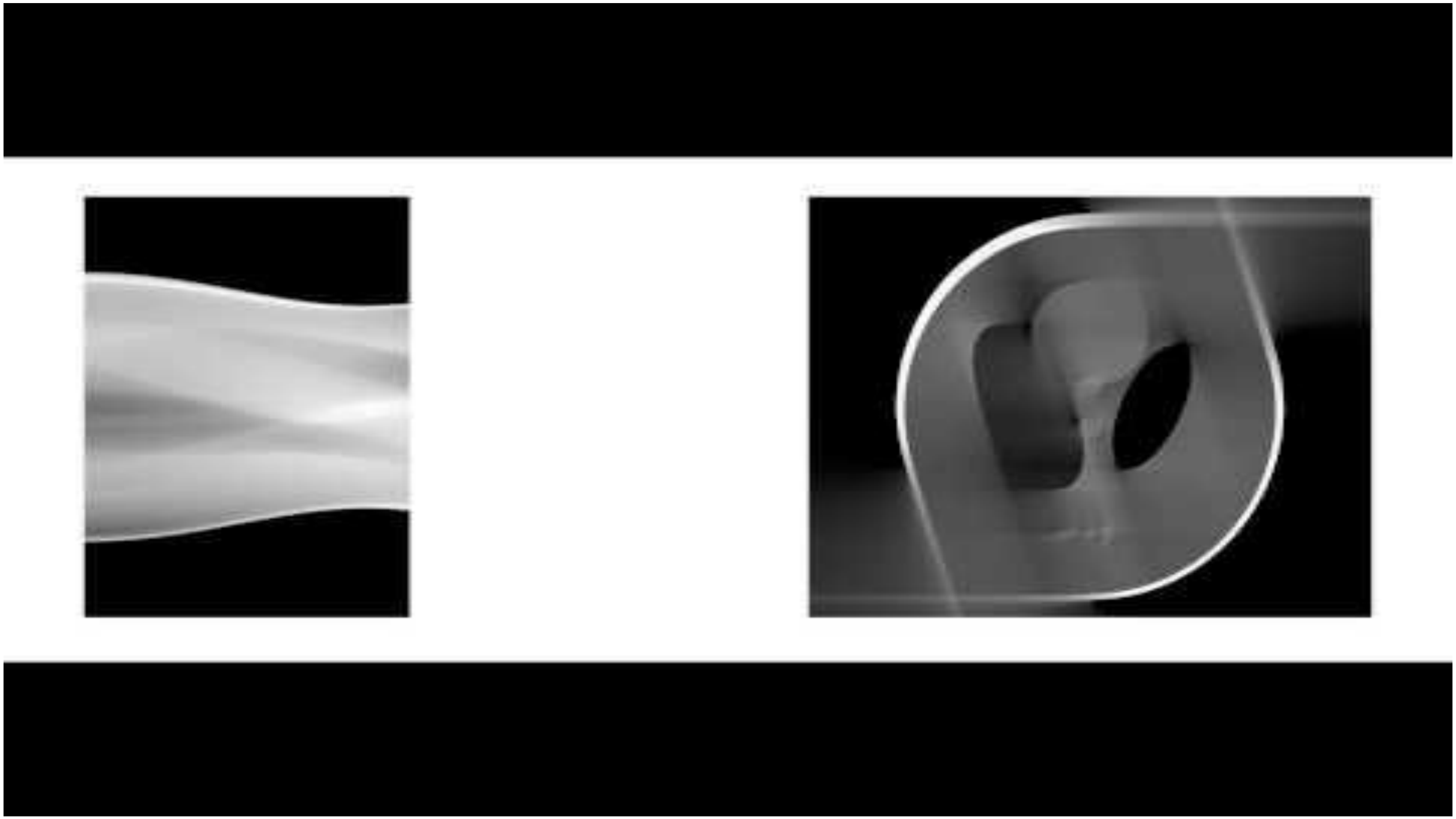
Chapters 12 & 13, The Essential Physics of Medical Imaging, Bushberg

Radon Transform and Sinogram



http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

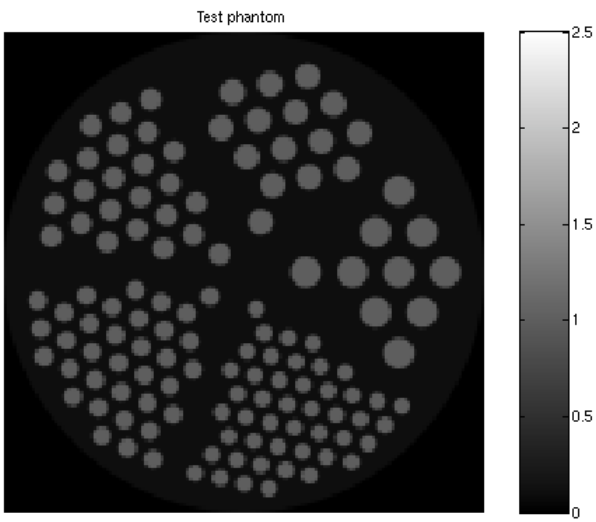
Filtered Back-projection

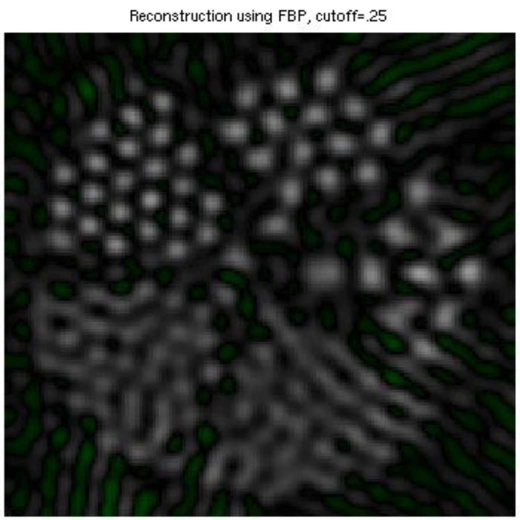


https://www.youtube.com/watch?v=ddZeLNh9aac

Filtered Back-projection

The “ringing” artifacts in reconstructed images.





In this context it is usually manifest itself as "streak artifacts". You may, for example, see this as lines radiating from the center and outwards. The term comes from electronics and is meant in the sense of a bell-ringing.

Filtered Back-projection

The sharp boundary of the Ram-Lak filter often makes the spatial domain filter kernel oscillatory.

It sometimes introduces the “ringing” artifacts in reconstructed images.

This can be effectively resolved by the Shepp and Logan filter

$$H_{SL}(\omega) = \begin{cases} |\omega| \sin\left(\dfrac{\omega}{4B}\right), & |\omega| < 2\pi B \\ 0, & \text{otherwise} \end{cases}$$

$$h_{SL}(x) = \frac{B}{\pi^2}\left\{\frac{1-\cos 2\pi B[(1/4B)+x]}{(1/4B)+x} + \frac{1-\cos 2\pi B[(1/4B)-x]}{(1/4B)-x}\right\}$$

An Alternative View of the Filters

In the spatial frequency domain

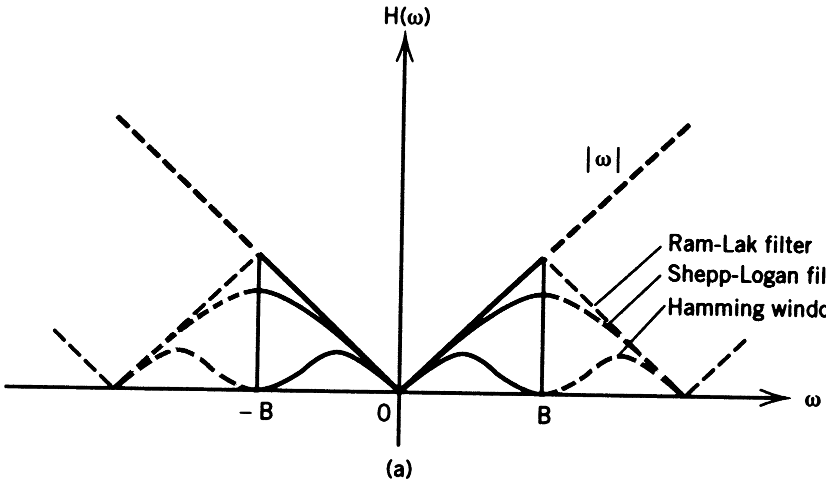


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

$$H(\omega)$$

In the spatial domain

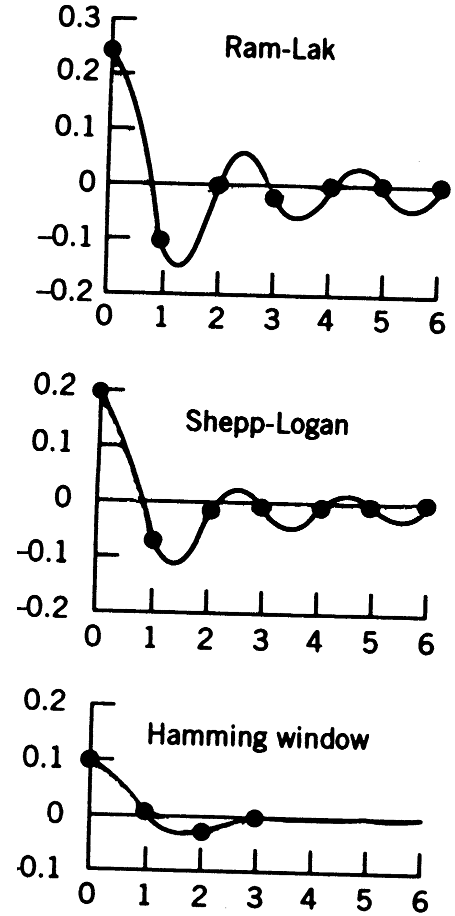


Figure 3-4 (b) Spatial domain filter kernals corresponding to the filter functions shown in the Ram-Lak filter is a high-pass filter with a sharp response but results in some noise enhancement, while the Shepp-Logan and the Hamming window filters are noise-smoothed filters and therefore have better SNR.

Filtered Back-projection

- Low spatial frequency data is overweighted. Filter to compensate for this. Weighted by $1/\omega$.

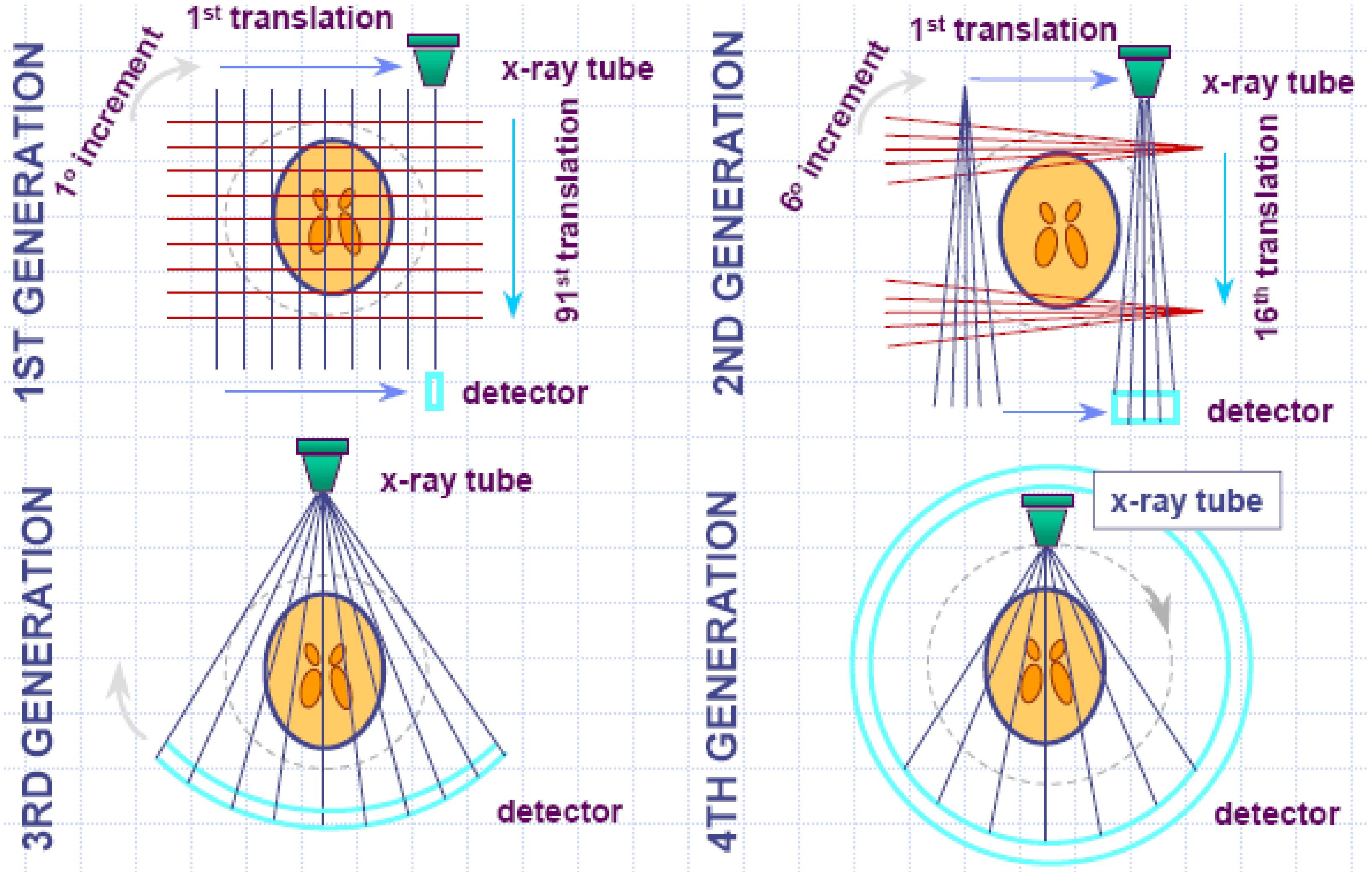
- Solution - filter each projection by $| \omega |$ to account for the uneven sampling density

Steps:

1) Fourier transform of the projection
2) Weight with $| \omega |$
3) Inverse transform
4) Back project
5) Add all angles

Most of current X-ray CT works in fan-beam mode …

X-ray Computed Tomography (CT)



Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\left\{F\left[p_\phi(x')\right] \cdot H(\omega)\right\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx',$$

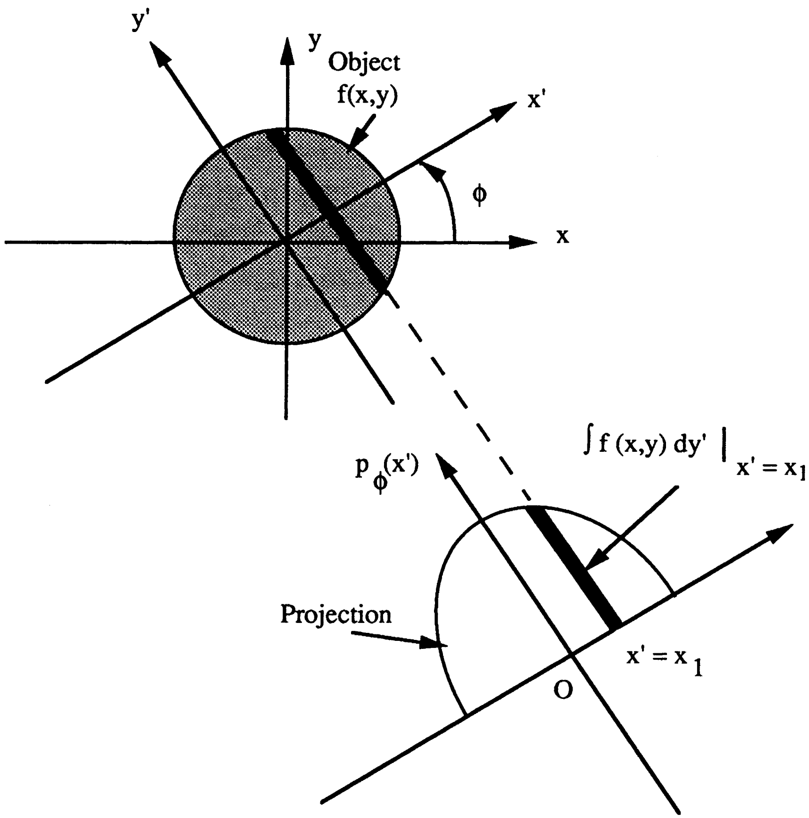
where $H(\omega) = |\omega| \cdot H'(\omega)$.

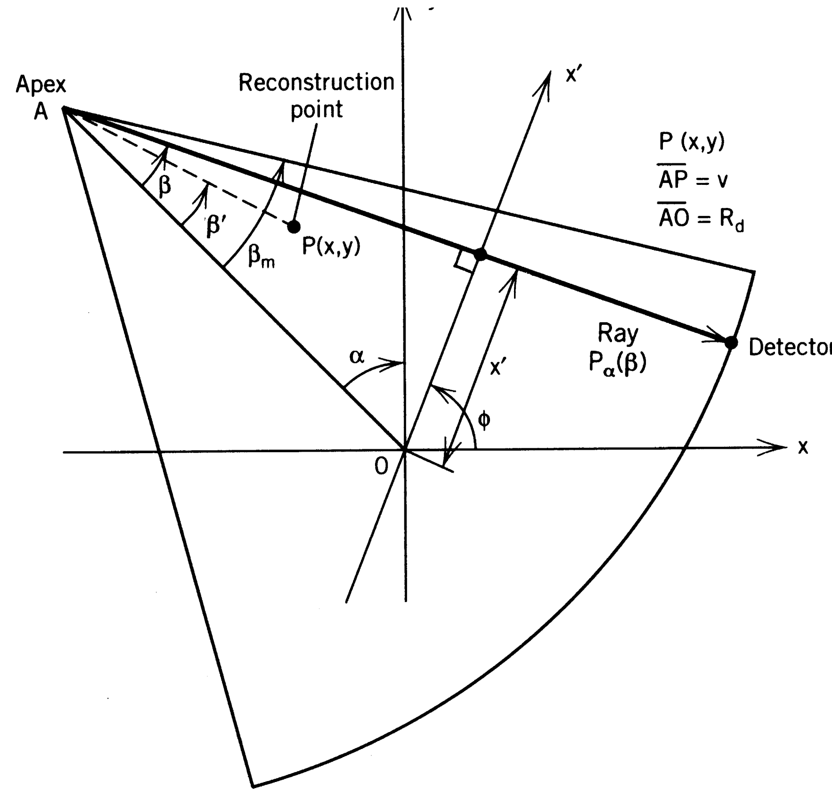
The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot h(x\cos\phi + y\sin\phi - x')dx' ,$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

Filtered Back-projection in Fan-Beam Mode





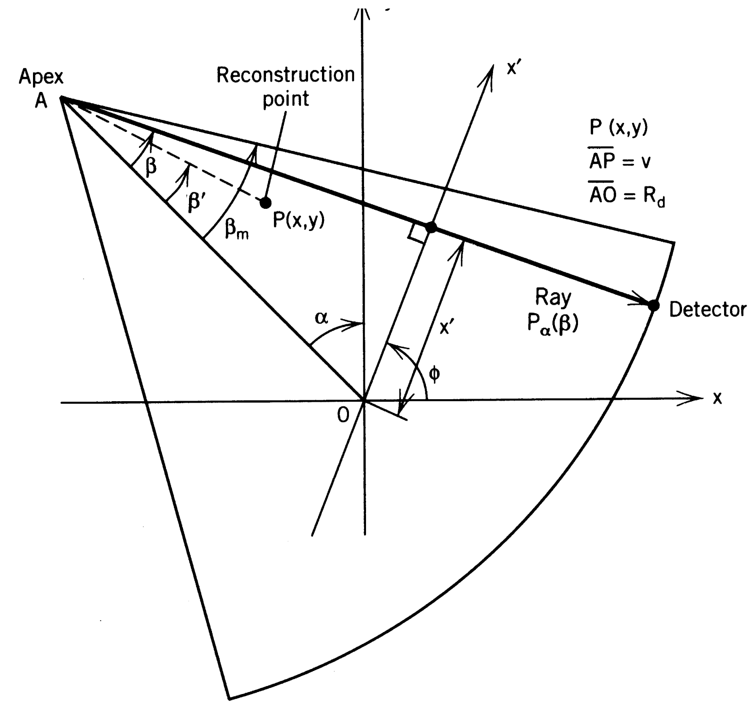
Comparing parallel beam and fan beam geometries

Filtered Back-projection in Fan-beam Mode

Since

$$x' = R_d \sin \beta$$

$$\phi = \alpha + \beta$$



The absolute value of the determinant of the Jacobian |J| is

$$|J| = \left|\frac{\partial(x'\phi)}{\partial(\alpha,\beta)}\right| = \begin{vmatrix} \dfrac{\partial x'}{\partial \alpha} & \dfrac{\partial x'}{\partial \beta} \\ \dfrac{\partial x'}{\partial \alpha} & \dfrac{\partial \phi}{\partial \beta} \end{vmatrix} = R_d \cos \beta$$

Filtered Back-projection in Fan-beam Mode

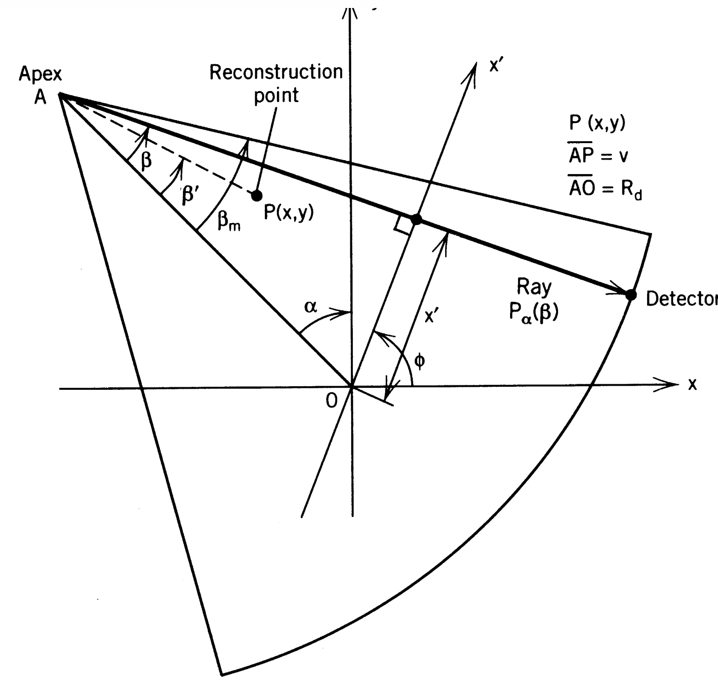
where β' is the angle between the central line and the line passing through the reconstructed point at (x,y). And v is the distance between the apex of the fan and the reconstruction point p.

$$v = \sqrt{(x\cos\alpha + y\sin\alpha)^2 + (x\sin\alpha - y\cos\alpha + R_d)^2}$$

$$\beta' = \tan^{-1}\left[\frac{x\cos\alpha + y\sin\alpha}{x\sin\alpha - y\cos\alpha + R_d}\right]$$

$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$



Filtered Back-projection in Fan-beam Mode (continued)

FBP in parallel beam case

$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\phi \int_{-\infty}^{\infty} dx' \, p_\phi(x') h(x \cos\phi + y \sin\phi - x')$$

$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

can be converted to FBP in fan beam mode by changing integration variables

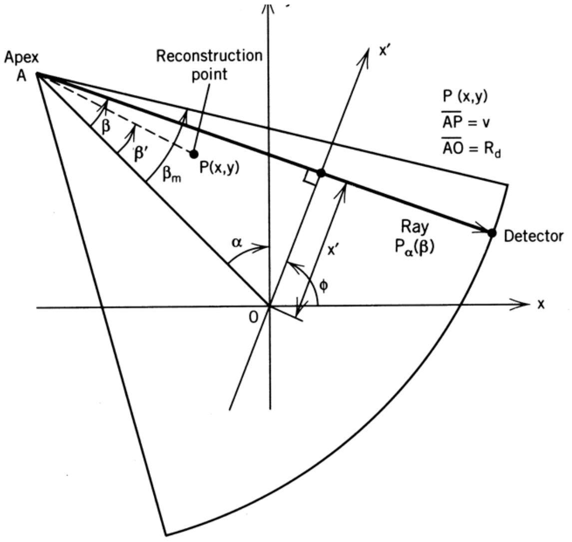
$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta$$

$$\times p_\alpha(\beta) h\{x \cos(\alpha + \beta) + y \sin(\alpha + \beta) - R_d \sin\beta\} |J|$$

Filtered Back-projection in Fan-beam Mode

Where the Jacobian |J| is

$$|J| = \left| \frac{\partial(x', \phi)}{\partial(\alpha, \beta)} \right| = R_d \cos\beta$$



and the FBP in fan beam mode becomes

$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta \, p_\alpha(\beta) h\{v \sin(\beta' - \beta)\} |J|$$

Filtered Back-projection in Fan-beam Mode (continued)

FBP in the parallel-beam case

$$\hat{f}(x, y) = \frac{1}{2\pi}\int_0^{2\pi} d\phi \int_{-\infty}^{\infty} dx'\, p_\phi(x') h(x\cos\phi + y\sin\phi - x')$$

$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

can be converted to **FBP in fan-beam mode** by changing integration variables

$$\hat{f}(x, y) = \frac{1}{2\pi}\int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta$$

$$|J| = \left|\frac{\partial(x', \phi)}{\partial(\alpha, \beta)}\right| = R_d \cos\beta$$

$$\times p_\alpha(\beta) h\{x\cos(\alpha+\beta) + y\sin(\alpha+\beta) - R_d\sin\beta\}|J|$$

Back-projection and Filtering Method for Reconstruction

Simple Back-projection and the 1/r Blurring

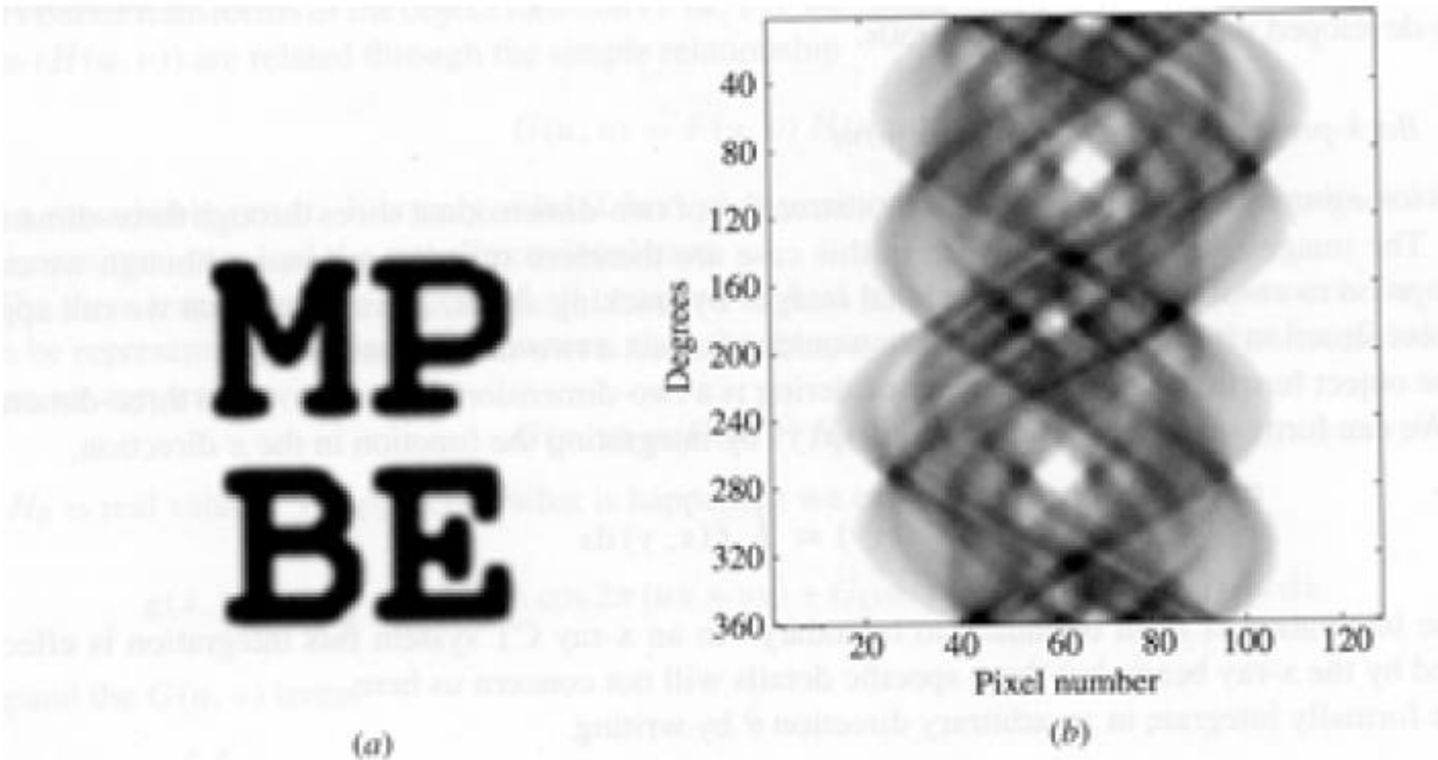
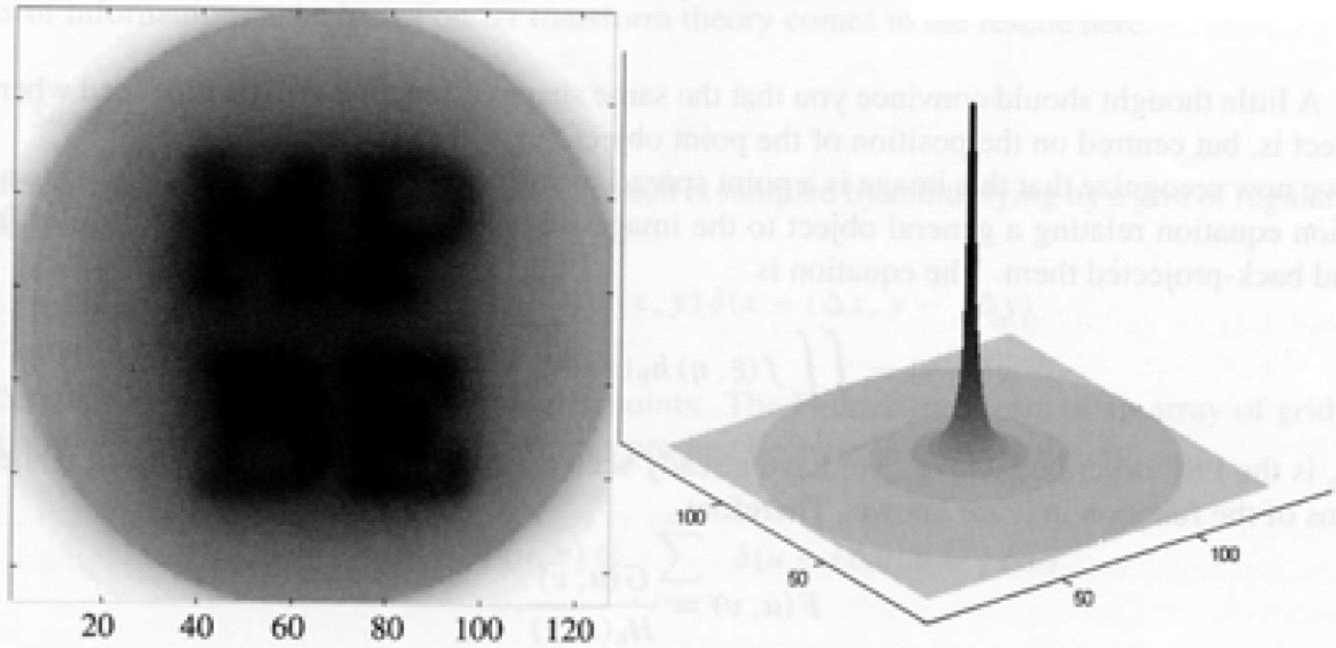


Figure 11.12. *(a) An image and (b) the Radon projection or sinogram of this image. The wavy sine li patterns explain the use of the term sinogram.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Simple Back-projection and the 1/r Blurring Revisited



(a) (b)

Figure 11.14. *(a) Image produced by back-projecting the sinogram given in figure 11.12(b); (b) the response to a point object in the middle of the field of view.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Impulse Response Function of Simple Backprojection Operator Revisited

$h_b(r) = 1/r$

$f_b(x,y) = f(x,y) * 1/r$

$F_b(\rho,\phi) = \mathrm{F}(\rho, \phi) / \rho$ since $F\{1/r\} = 1/\rho$

Back projected image is blurred by convolution with 1/*r*.

Reconstruction with Back-projection and Filtering

If we know that the consequence of simple back-proejction (under certain assumptions) is to apply a 1/r blurring on the input image …

$$b(x, y) = f(x, y) ** \left(\frac{1}{r}\right)$$

Can we envisage an alternative reconstruction method that

Back projection first

and then

***De-convolve the 1/r blurring*?**

Reconstruction with Back-projection and Filtering

The scheme leads to the back-projection and filtering reconstruction method

$$\hat{f}(x, y) = \mathscr{F}_2^{-1}[\rho B(\omega_x, \omega_y)]$$

where ρ is the radial spatial frequency and

$$B(\omega_x, \omega_y) = \mathscr{F}_2[b(x, y)]$$

Simple Backprojection --BB

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_\phi (x,y) = \int p(\phi, x') \, \delta (x \cos \phi + y \sin \phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{\text{back-projected}} (x,y) = \int b_\phi (x,y) \, d\phi$$

$$B (x,y) = f_b (x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p(\phi, x') \, \delta (x \cos \phi + y \sin \phi - x') \, dx'$$

Reconstruction with Back-projection and Filtering

BPF reconstruction provides relatively poor images compared with FBP results because of the following two issues:

The back projection step results in an image with infinite extend. Cutting it to NxN points for filtering leads to loss of information.

The 2-D filter function discussed above has a slope discontinuity and therefore leads to ringing artifact.

Reconstruction with Back-projection and Filtering

If high quality reconstructions were to be achieved the BPF method, the following aspects have to be considered

Although the final reconstruction is on NxN points, the back projection step should use an matrix that is at least 2Nx2N.

We would need to apply appropriate windowing to the filter function discussed above, just like in the parallel beam case …

Reconstruction with Back-projection and Filtering

The procedures of BPF algorithm is VERY SIMPLE!

1. Obtain the backprojected image at least up to $2N \times 2N$ if the size of the original image matrix is $N \times N$.
2. Obtain the 2-D FFT of the full $2N \times 2N$ data array.
3. Obtain the optimum filtering in 2-D in Fourier domain by multiplication of ρ filter.
4. Obtain the 2-D inverse FFT and select the $N \times N$ image in the central region.
5. Normalize the image.

From Page 88, Foundation of Medical Imaging, Z. H. Cho, John Wiley & Sons, 1993.

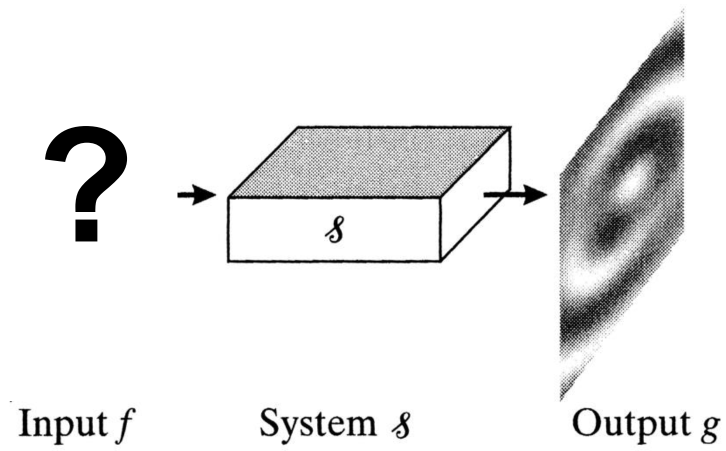
Chapter 2: Mathematical Preliminaries

Iterative Reconstruction Methods

Contents

- **Reconstruction and general linear inverse problems**
- **Statistical description of the data**
- **Image reconstruction criteria**
- **General structure of iterative reconstruction methods**
- **Maximum Likelihood Expectation Maximization (MLEM) algorithm**
- **Other related reconstruction methods**

General Inverse Problem and Image Reconstruction

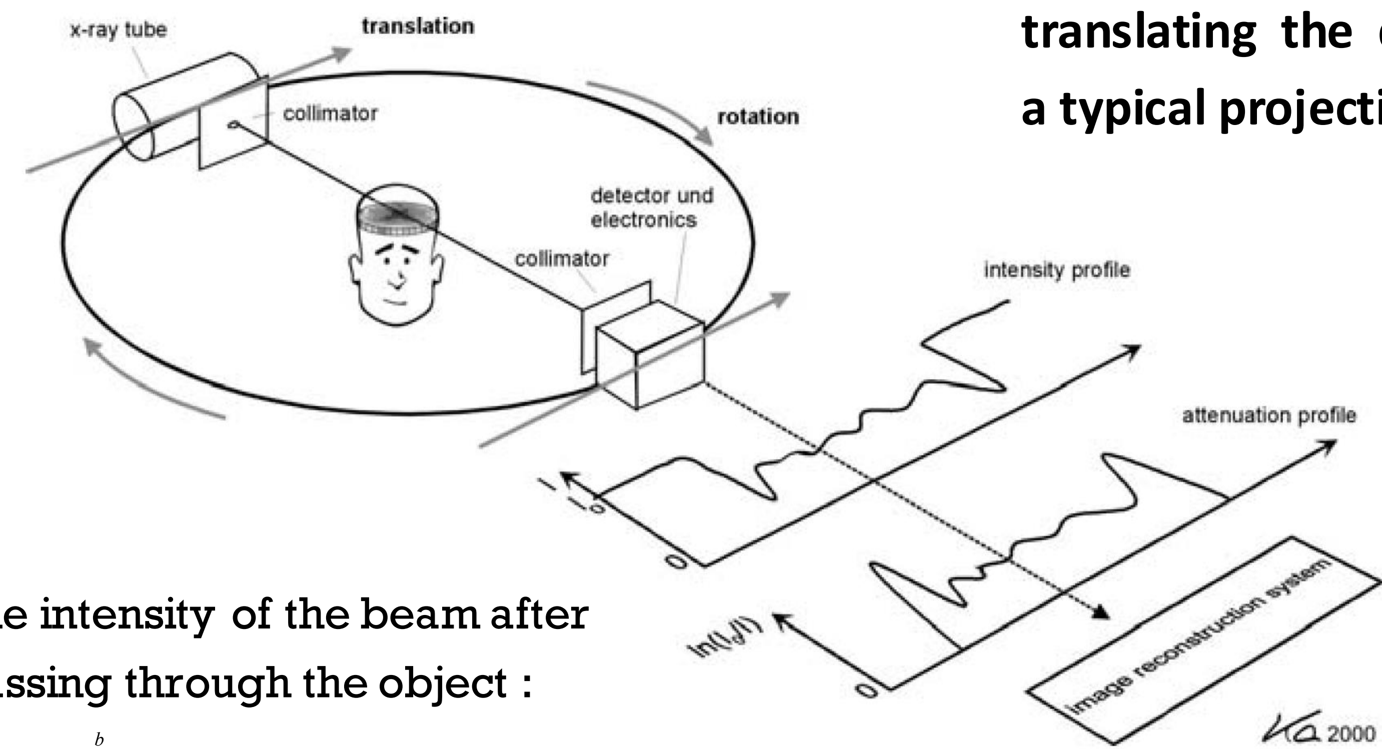


The inverse problem:

- Given a set of measured output signal
- Given the statistical description of the data
- Given a known system response function

what should be the input signal that gave rise to the output data?

Projection Data from Early X-ray CT Systems



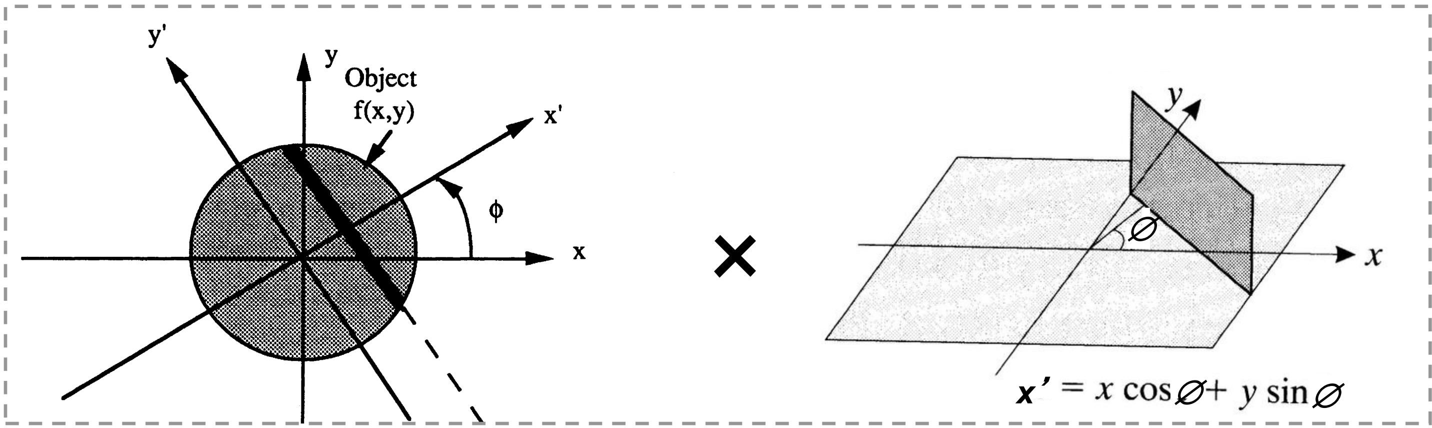
Data measured by translating the detector is a typical projection data.

The intensity of the beam after passing through the object :

$$I = I_0 e^{-\int_a^b \mu(t)\cdot dt} \Leftrightarrow \ln(I_0 / I) = \int_a^b \mu(t)\cdot dt$$

From Computed Tomography, Kalender, 2000.

Radon Transform



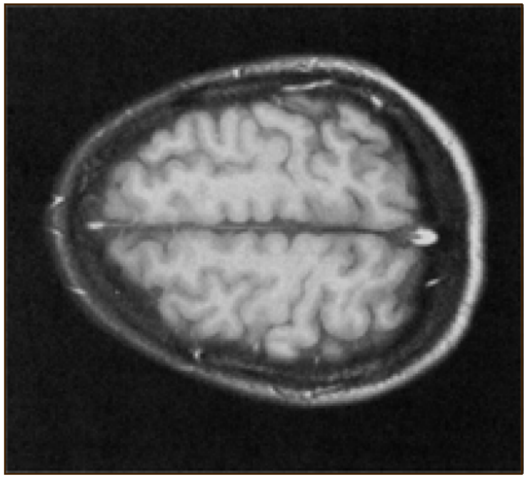
2-D integral

The value of the projection function $p_{\phi}(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The integral of a line impulse function and a given 2-D signal gives the *projection* data from a given view …

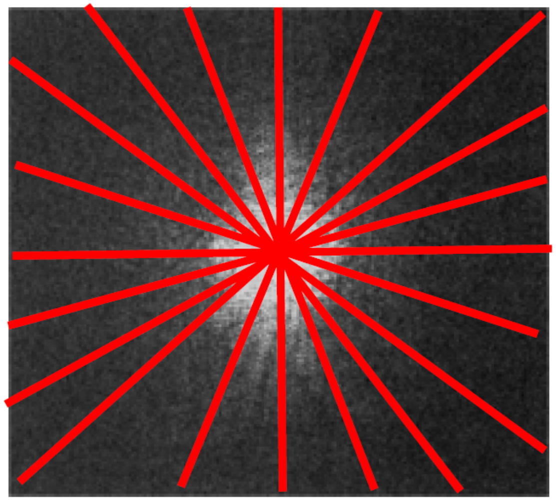
$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Central Slice Theorem



DFT image represents integration of original projections DFT transformed and summed together.

1-D DFTs at each projection



This is the fast way to create the DFT image from projection data. The more projections taken, the more complete the sampling.

http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Simple Back-projection and the 1/r Blurring

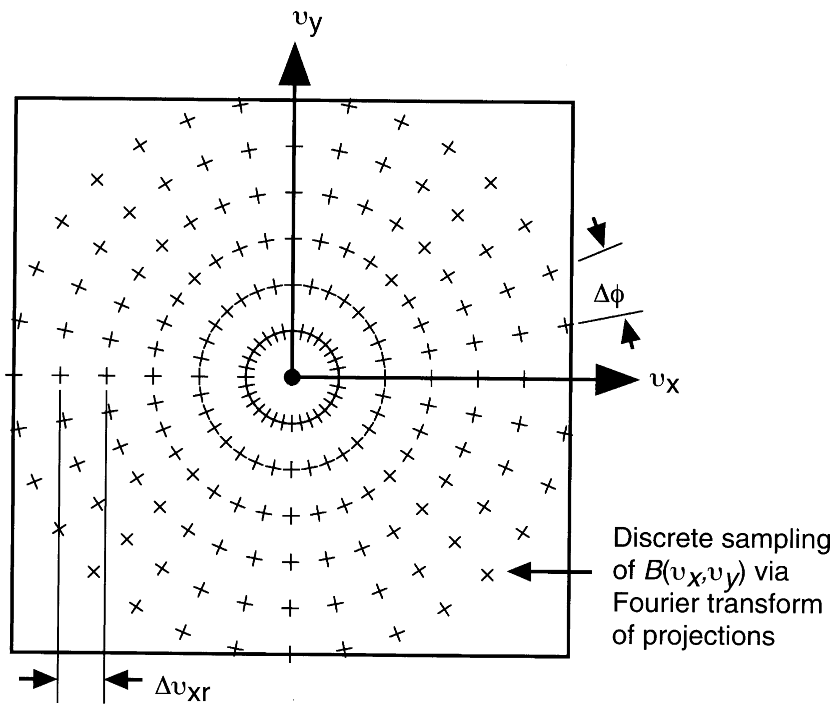
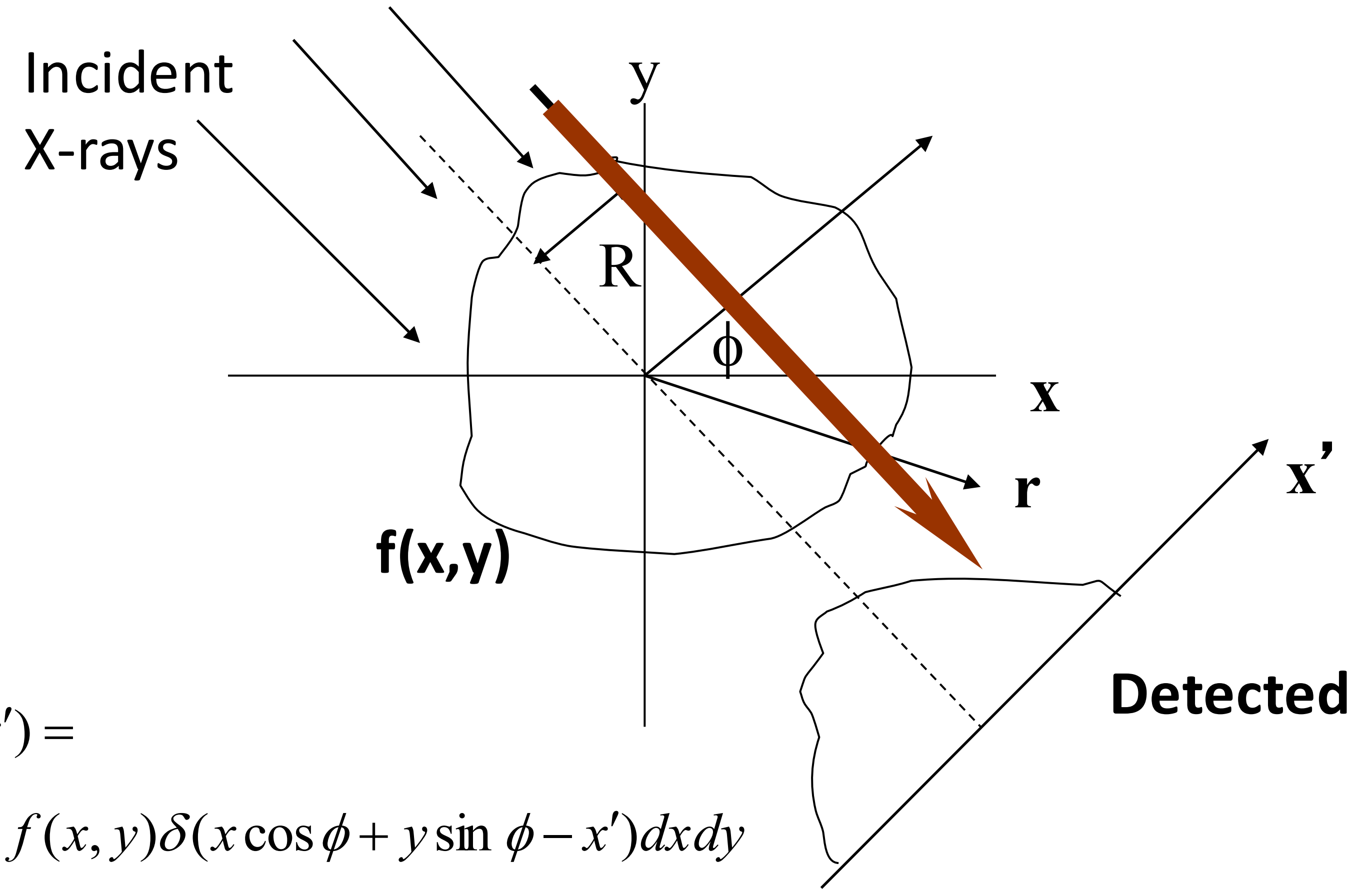


FIGURE 18 The discrete sampling pattern of $F(\upsilon_x,\upsilon_y)$ contained in $B(\upsilon_x,\upsilon_y)$, resulting from the use of discretely sampled projections.

The nature of the 1/r blurring:
Radon transform produced equally spaced radial sampling in Fourier domain.

Simple Back Projection from Projection Data

Projection data p(ϕ, x')



$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Simple Back Projection from Projection Data

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

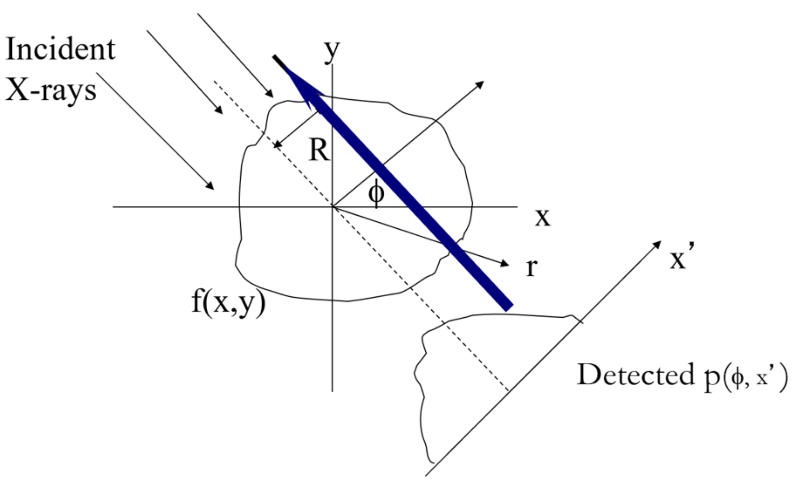
Result from a back projection from a single view angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{simple\ back\text{-}projection}(x,y) = \int b_\phi(x,y) \, d\phi$$

$$f_{\text{simple back-projection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$



Simple Back-projection and the 1/r Blurring

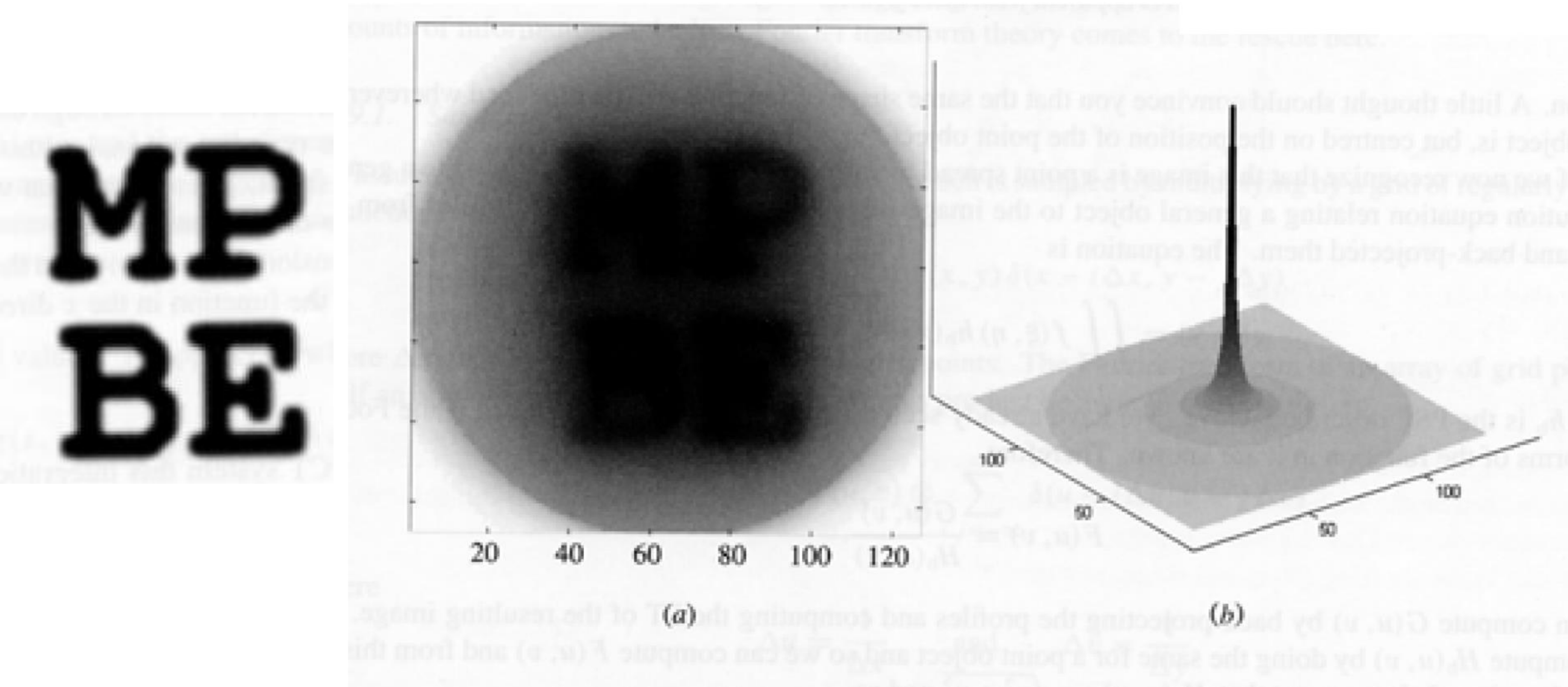


Figure 11.14. *(a) Image produced by back-projecting the sinogram given in figure 11.12(b); (b) the response to a point object in the middle of the field of view.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Simple Backprojection and Inverse Radon Transform

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_{\phi}(x,y) = \int_{-\infty}^{\infty} p_{\phi}(x') \cdot \delta(xcos\phi + ysin\phi - x')dx'$$

Adding up all the back projections from all the angles gives,

$$\hat{f}_{\text{simple backprojection}}(x,y) = \int_{0}^{\pi} d\phi \int_{-\infty}^{\infty} p_{\phi}(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

$$\hat{f}_{\text{Inverse Radon transform}}(x,y) = \int_{0}^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_{\phi}(x')] \cdot |w|\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Review of Fourier Transform and Filtering Spectral Filtering

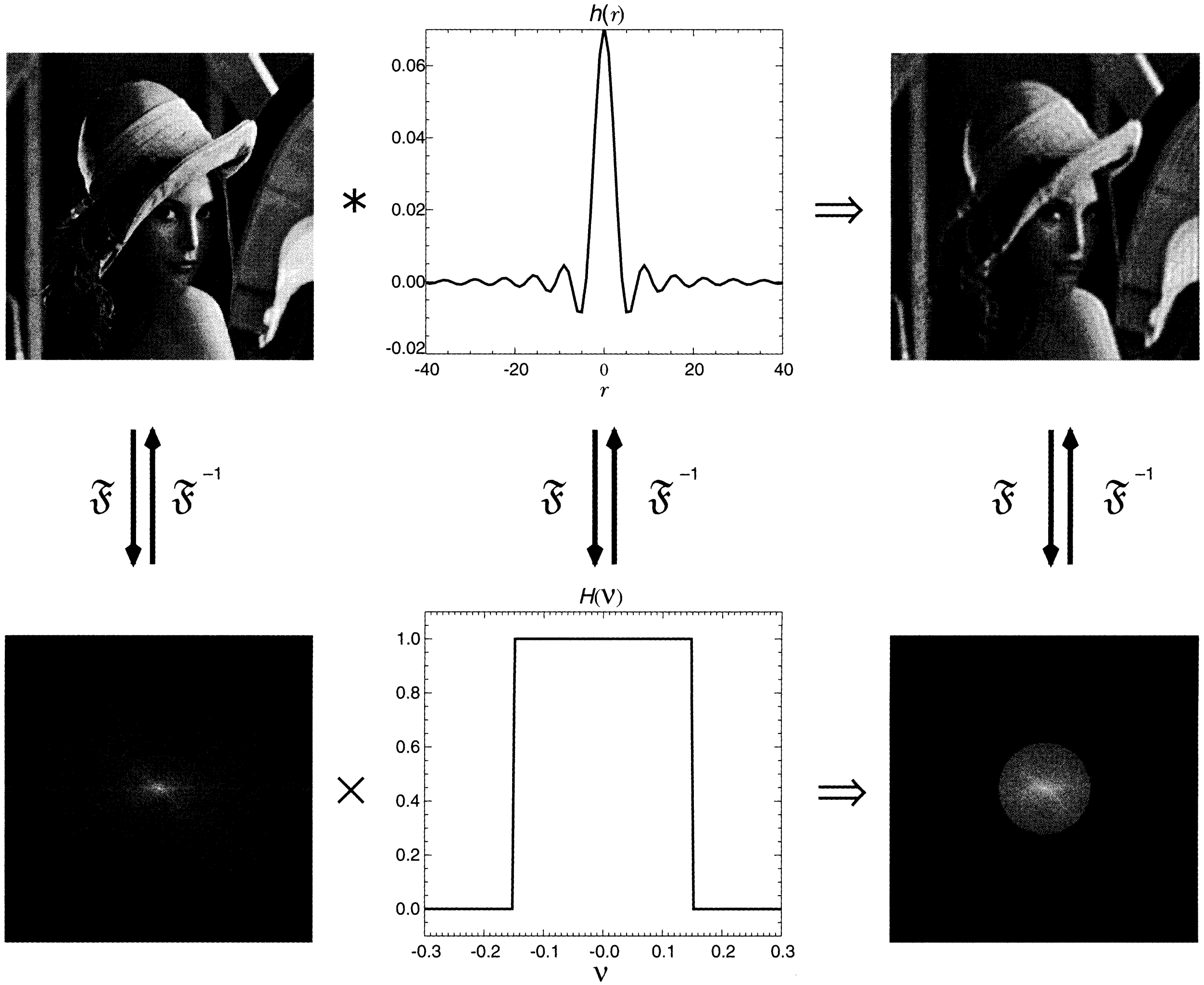


FIGURE 3 Ideal lowpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column are the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2+y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Review of Fourier Transform and Filtering
Spectral Filtering

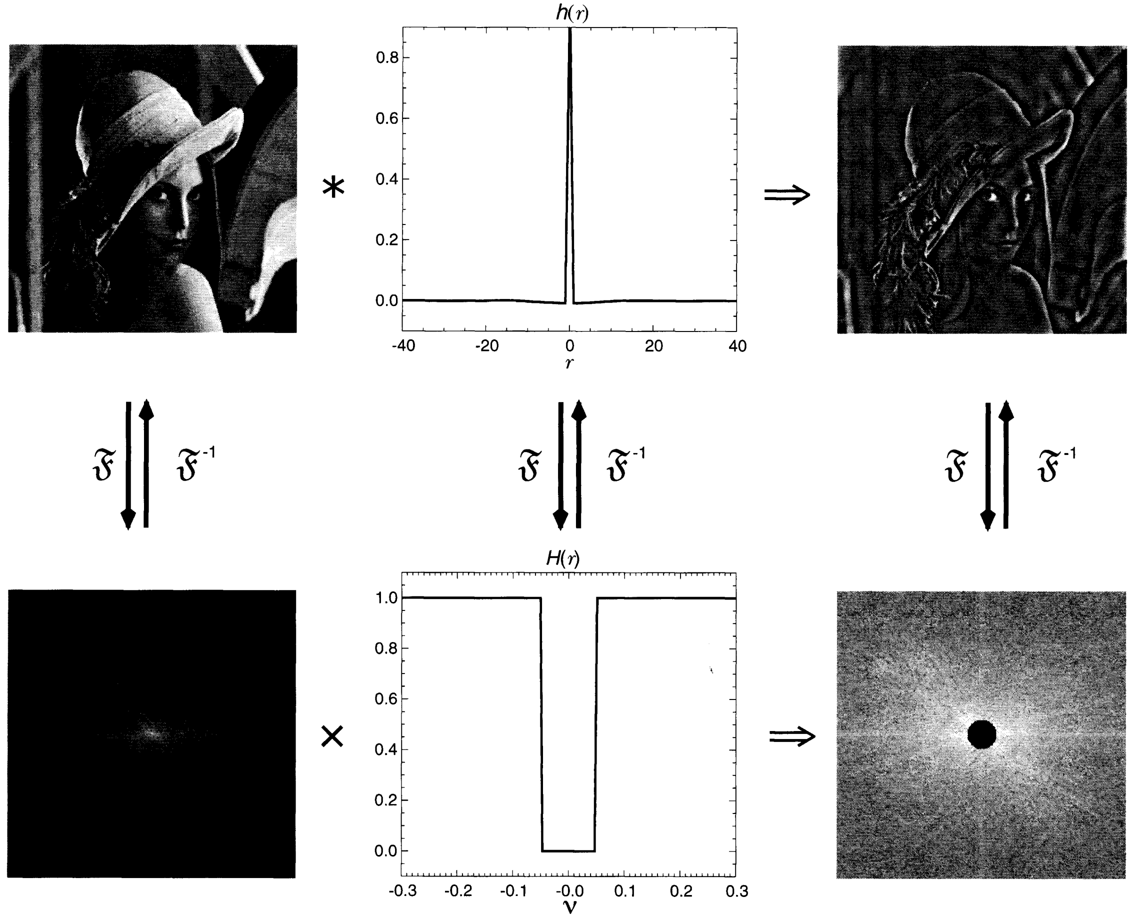


FIGURE 4 Ideal highpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column shows the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2+y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Filtered Back-projection

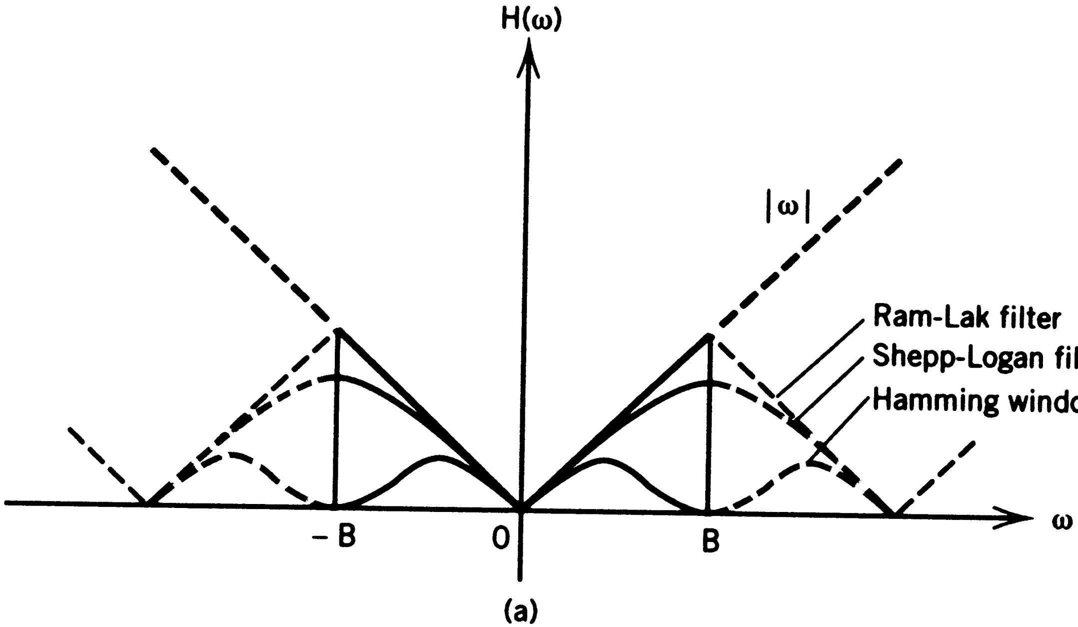
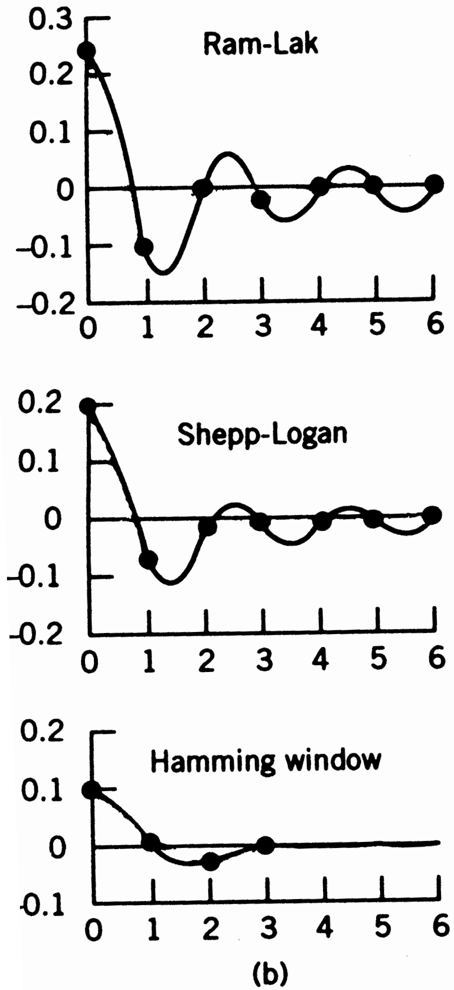


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

Filtered Back-Projection

$$\Delta x = 1/(2B),$$

Sampled version



$$h_{\mathrm{RL}}(0) = B^2 = \frac{1}{4\,\Delta x^2} \qquad (\text{if } k = 0)$$

$$h_{\mathrm{RL}}(k) = 0 \qquad (\text{if } k \text{ even})$$

$$h_{\mathrm{RL}}(k) = \frac{-4B^2}{\pi^2 k^2} = \frac{-1}{\pi^2 k^2\,\Delta x^2} \qquad (\text{if } k \text{ odd})$$

$$h_{SL}(k) = \frac{-2}{\pi^2\,\Delta x^2 (4k^2 - 1)} = \frac{-8B^2}{\pi^2 (4k^2 - 1)}$$

Figure 3-4 (b) Spatial domain filter kernals corresponding to the filter functions shown in the Ram-Lak filter is a high-pass filter with a sharp response but results in some noise enhancement, while the Shepp-Logan and the Hamming window filters are noise-smoothed filters and therefore have better SNR.

Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\left\{F\left[p_\phi(x')\right] \cdot H(\omega)\right\} \cdot \delta(x\cos\phi + y\sin\phi - x')\,dx',$$

where $H(\omega) = |\omega| \cdot H'(\omega)$.

The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

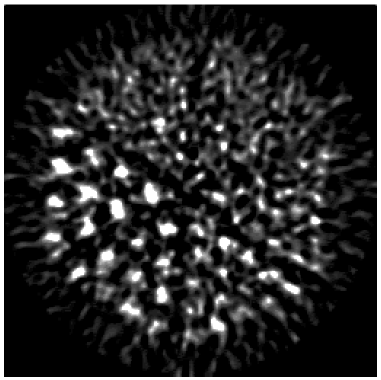
$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot h(x\cos\phi + y\sin\phi - x')\,dx',$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

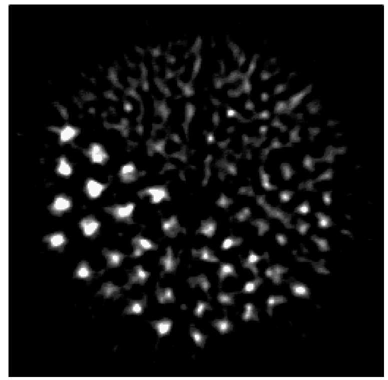
Chapter 2: Mathematical Preliminaries

Iterative Reconstruction Methods

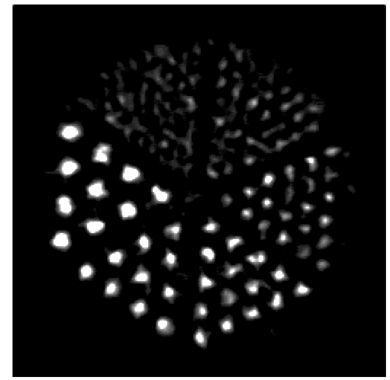
The Importance of Counts



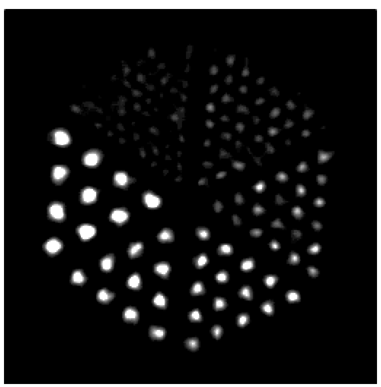
50 000 counts



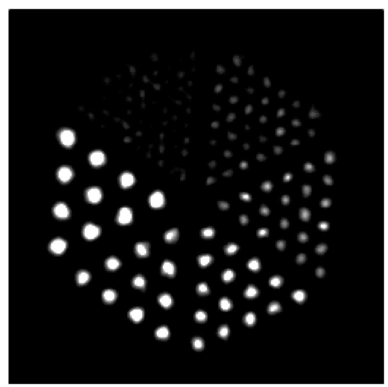
100 000 counts



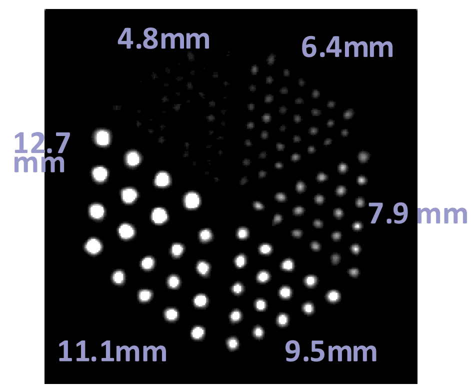
200 000 counts



500 000 counts



1 million counts

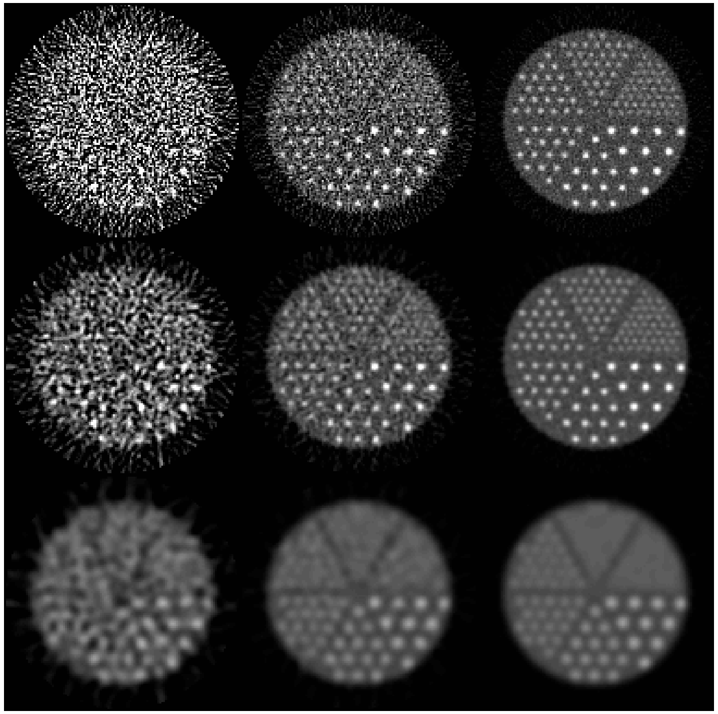


2 million counts

Noise In PET Images

- Noise in PET images is dominated by the counting statistics of the coincidence events detected.
- Noise can be reduced at the cost of image resolution by using an apodizing window on ramp filter in image reconstruction (FBP algorithm).

10^5 10^6 10^7 counts



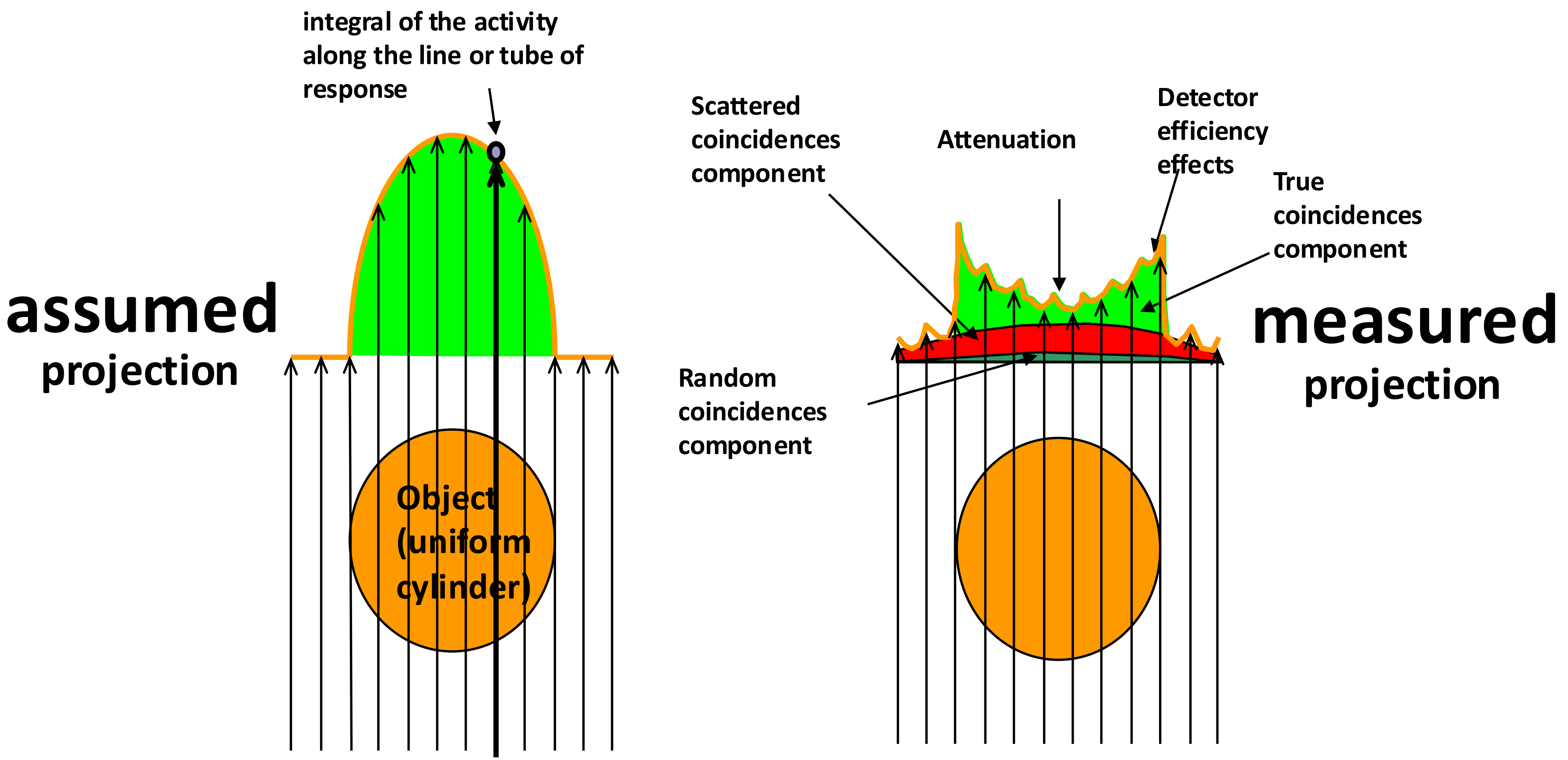
Unapodized ramp filter

Hanning window, 4mm

Hanning window, 8mm

An Example of PET Image Reconstruction

- Data corrections are necessary
  - the measured projections are not the same as the projections assumed during image reconstruction



Analytical Methods

- Advantages
  - Fast
  - Simple
  - Predictable, linear behaviour
- Disadvantages
  - Not very flexible
  - Image formation process is not modelled $\Rightarrow$ image properties are sub-optimal (noise, resolution)

Statistical Methods

- Advantages
  - Can accurately model the image formation process (use with non-standard geometries, e.g. not all angles measured, gaps)
  - Allow use of constraints and *a priori* information (non-negativity, boundaries)
  - Corrections can be included in the reconstruction process (attenuation, scatter, etc)

- Disadvantages
  - Slow
  - Less predictive behaviour (noise? convergence?)

Image Reconstruction and General Linear Inverse Problem

A basic problem for image reconstruction in emission tomography

Find the object distribution **f**, given (1) a set of projection measurements **g**, (2) information (in the form of a matrix **H**) about the imaging system that produced the measurements, and, possibly, (3) a statistical description of the data and (4) a statistical description of the object (Fig. 1).

Image Reconstruction and General Linear Inverse Problem

An Emission Tomography System

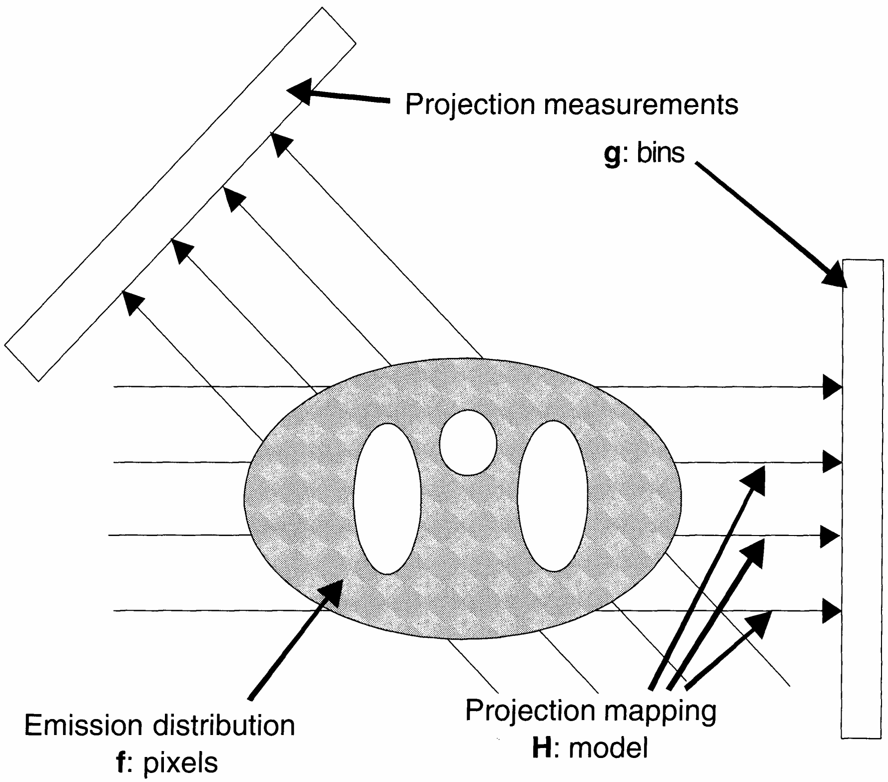
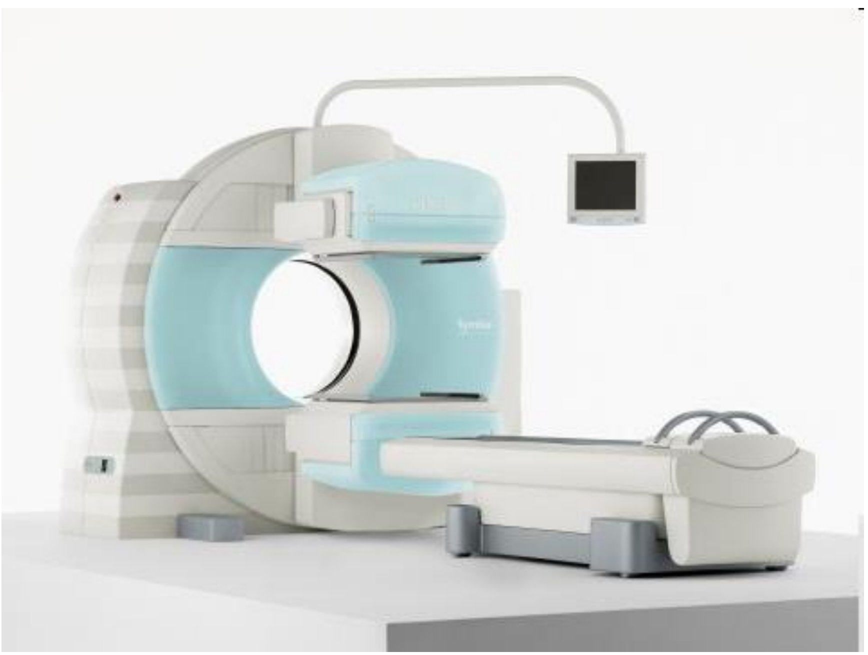


FIGURE 1 A general model of tomographic projection in which the measurements are given by weighted integrals of the emitting object distribution.

The Forward and Inverse Problem?

Single Photon Emission Computed Tomography (SPECT)

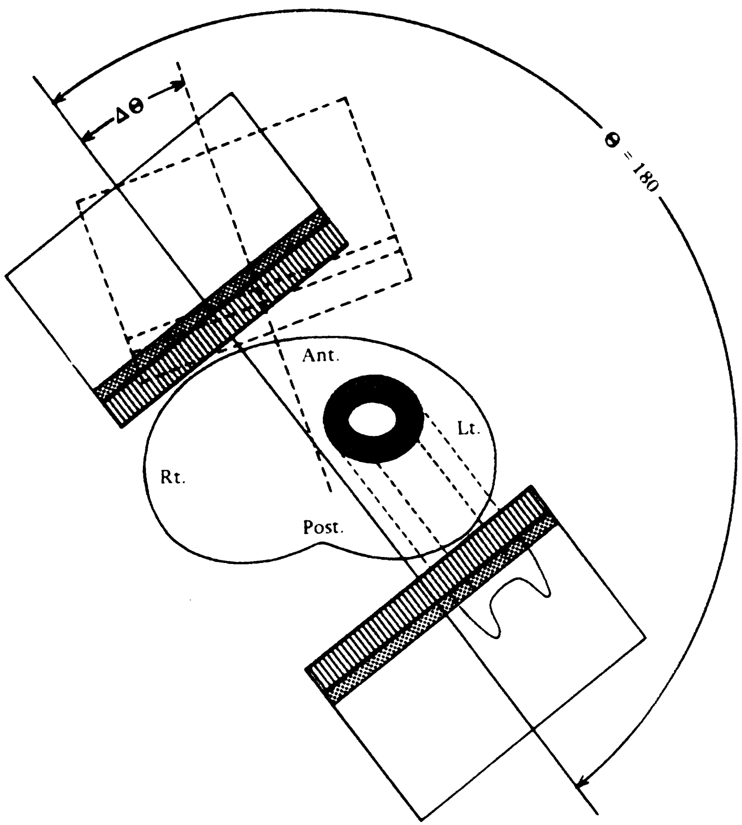


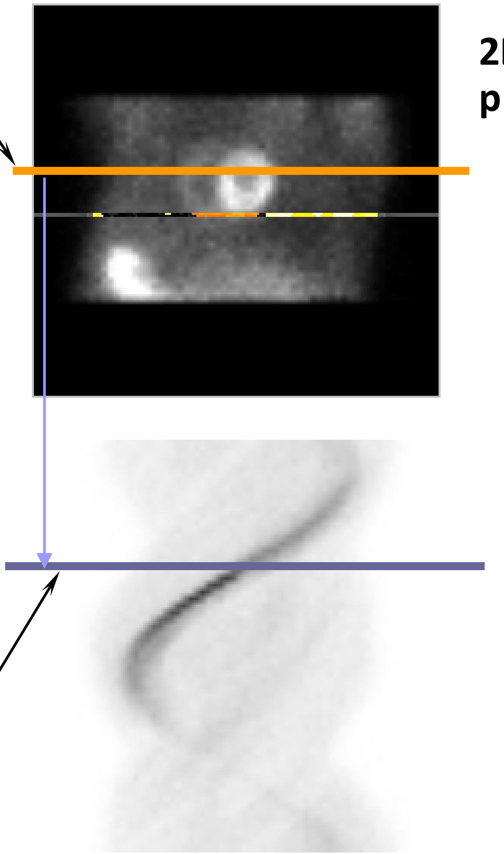
Siemens Symbia SPECT/CT

Philips Precedence SPECT/CT

Basic SPECT – Projection Data

Axial level of sinogram



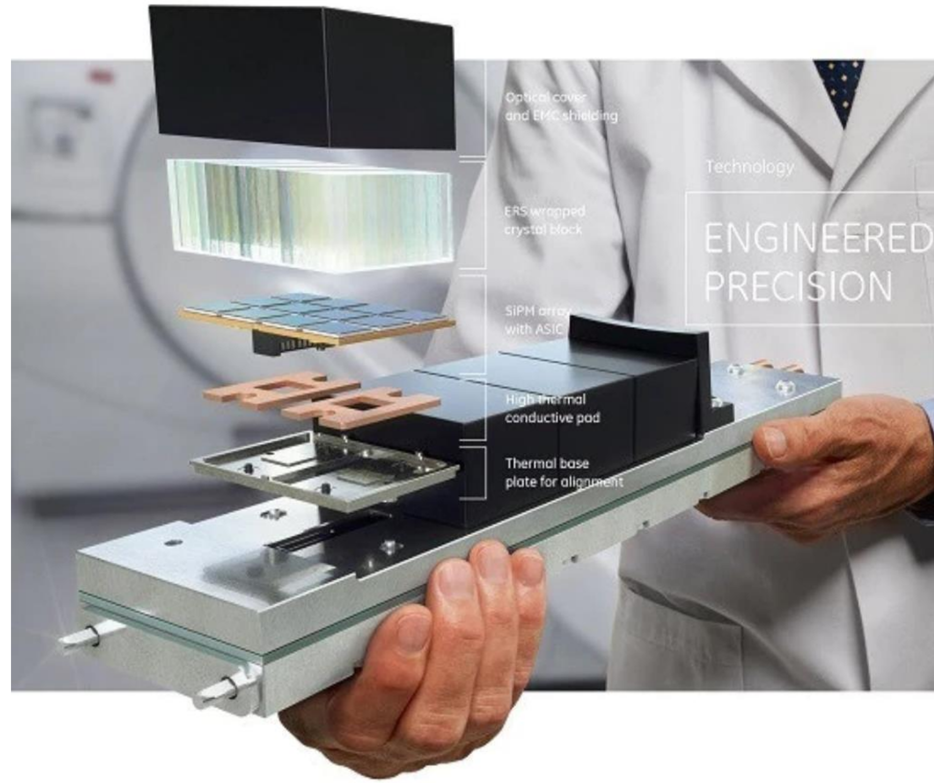


2D planar projection

Sinogram

Angle of above projection

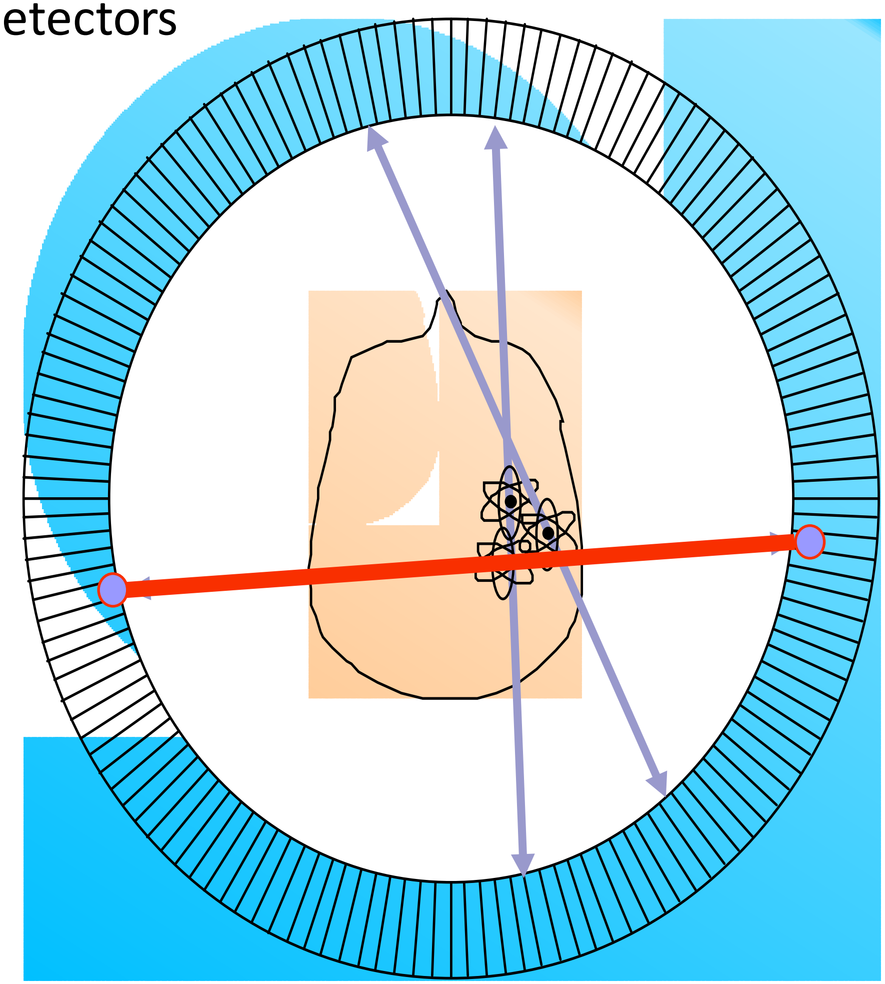
Modern PET Cameras



GE Discovery II PET/CT Scanner

Detect Radioactive Decays

Ring of Photon Detectors

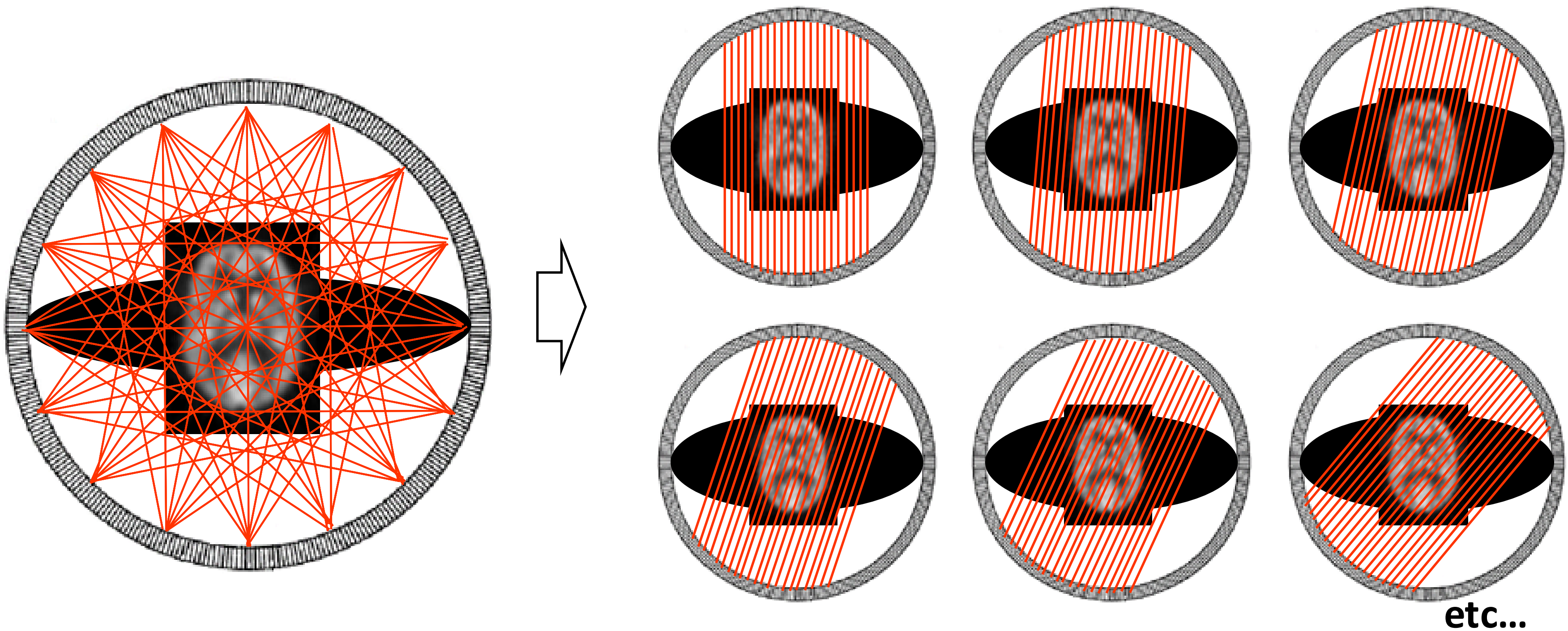


- Radionuclide decays, emitting β^+.
- β^+ annihilates with e^- from tissue, forming back-to-back 511 keV photon pair.
- 511 keV photon pairs detected via time coincidence.
- Positron lies on line defined by detector pair (known as a *chord* or a *line of response* or a *LOR*).

Detect Pair of Back-to Back 511 keV Photons

PET Data Acquisition

- Organization of data
  - True counts in LORs are accumulated
  - In some cases, groups of nearby LORs are grouped into one average LOR (“mashing”)
  - LORs are organized into projections



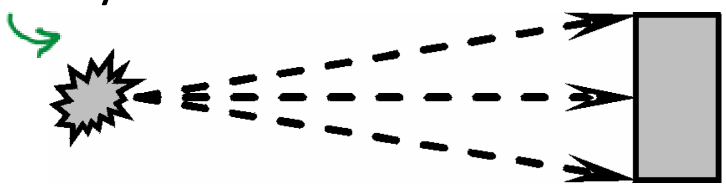
The Basic Idea of Statistical Reconstruction

Consider the simplest imaging problem as illustrated below:

N: The number of photons emitted by the source

k: The actual number of detected photons

μ: Mean or expected number of detected photons:

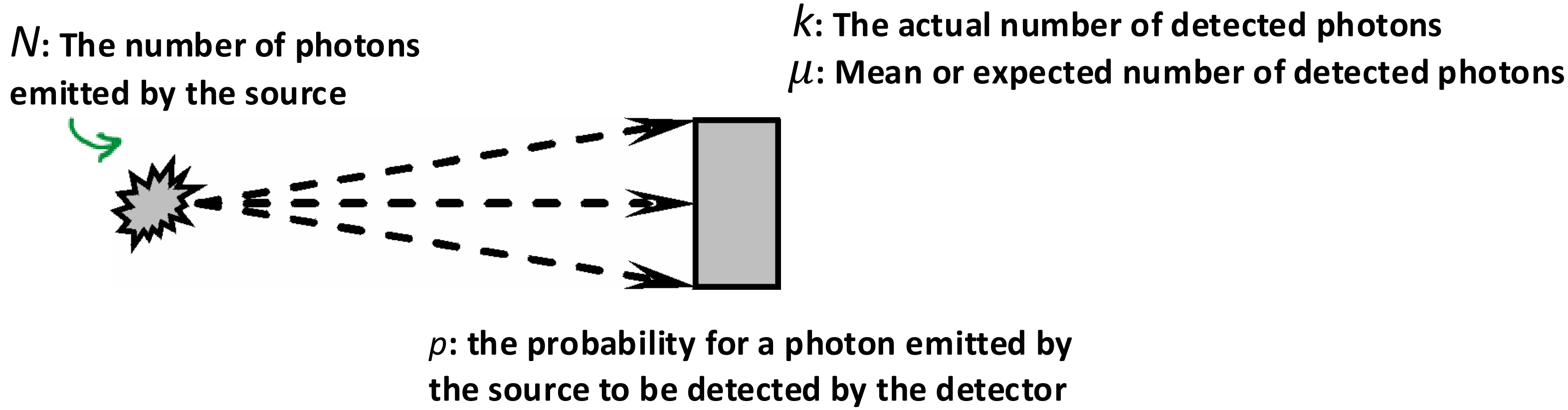


p: the probability for a photon emitted by the source to be detected by the detector

During an imaging study, we observed k counts, and we know that the probability for a photon emitted by the source to be detected by the detector is p, then

how do we estimate the number of photons being emitted by the source during the imaging period?

The Basic Idea of Statistical Reconstruction



We can estimate the underlying parameter *N* by the Maximum-Likelihood Estimation (MLE) strategy:

$$\hat{N}_{Maximum\ Likelihood\ estimation} = \arg\max_{N}\{P[k|N]\}$$

$P[k(N)]$ is call the Likelihood function: the probability of detecting *k* counts given there were a total of *N* gamma rays emitted by the source.

The Basic Idea of Statistical Reconstruction

The likelihood function:

In very non-technical terms, the conditional probability,

$$\boldsymbol{L}(\boldsymbol{f}, \boldsymbol{g}) \equiv p(\boldsymbol{g}|\boldsymbol{f}),$$

of the underlying source function ($\boldsymbol{f}$) leading to the measurement ($\boldsymbol{g}$).

The basic idea of Maximum-Likelihood reconstruction is simple:

We choose the solution (estimated source function) that maximizes the likelihood function, or the solution that is the most likely to produce the measured data.

$$\hat{\boldsymbol{f}}_{Maximum\ Likelihood} = \arg\max_{\boldsymbol{f}}\{P[\boldsymbol{g}|\boldsymbol{f}]\}$$

Source of Noise

Source of noise:

Measurement error, instrument malfunction …

Random fluctuation in the number of counts detected by detectors

Other additive noise

In emission tomography systems, *detector readout noise* is generally much smaller than the statistical fluctuation associated with the counting process.

Bernoulli Process

Consider the radioactive disintegration process in a sample, it follows the following four conditions:

- ☞ **It consists of N trails.**
- ☞ **Each trail has a binary outcome: success or failure (decay or not).**
- ☞ **The probability of success (decay) is a constant from trail to trail – all atoms have equal probability to decay.**
- ☞ **The trails are independent.**

In statistics, these four conditions characterize a Bernoulli process.

Binomial Distribution

The binomial distribution gives the discrete probability distribution of obtaining exactly k successes out of N Bernoulli trials (where the result of each Bernoulli trial is true with probability *p* and false with probability *1-p*). The binomial distribution is therefore given by

$$p(k|N,p)=\binom{N}{k}p^{k}(1-p)^{N-k}$$

$$where$$

$$\binom{N}{k}=\frac{N!}{k!(N-k)!}$$

Binomial Distribution and Imaging Applications

Suppose we have *a point source* generated N gamma rays and we have a pixilated detector for detecting these photons

Each gamma ray photon has *a fixed probability p* of reaching a given detector element. *p* is defined by the relative distance between the source and the pixel and the collimation configuration used.

So the *probability of detecting k gamma rays* on the pixel is given as

$$p(k|N,p) = \binom{N}{k} p^k (1-p)^{N-k}$$

Furthermore, *the number of counts detected on different pixels are independent* …

Binomial Distribution

For a binomial distribution, the mean or the expectation of the number of disintegration in time t is given by

$$\mu \equiv \sum_{n=0}^{N} n \cdot P_n = \sum_{n=0}^{N} n \cdot \binom{N}{n} p^n q^{N-n} = Np$$

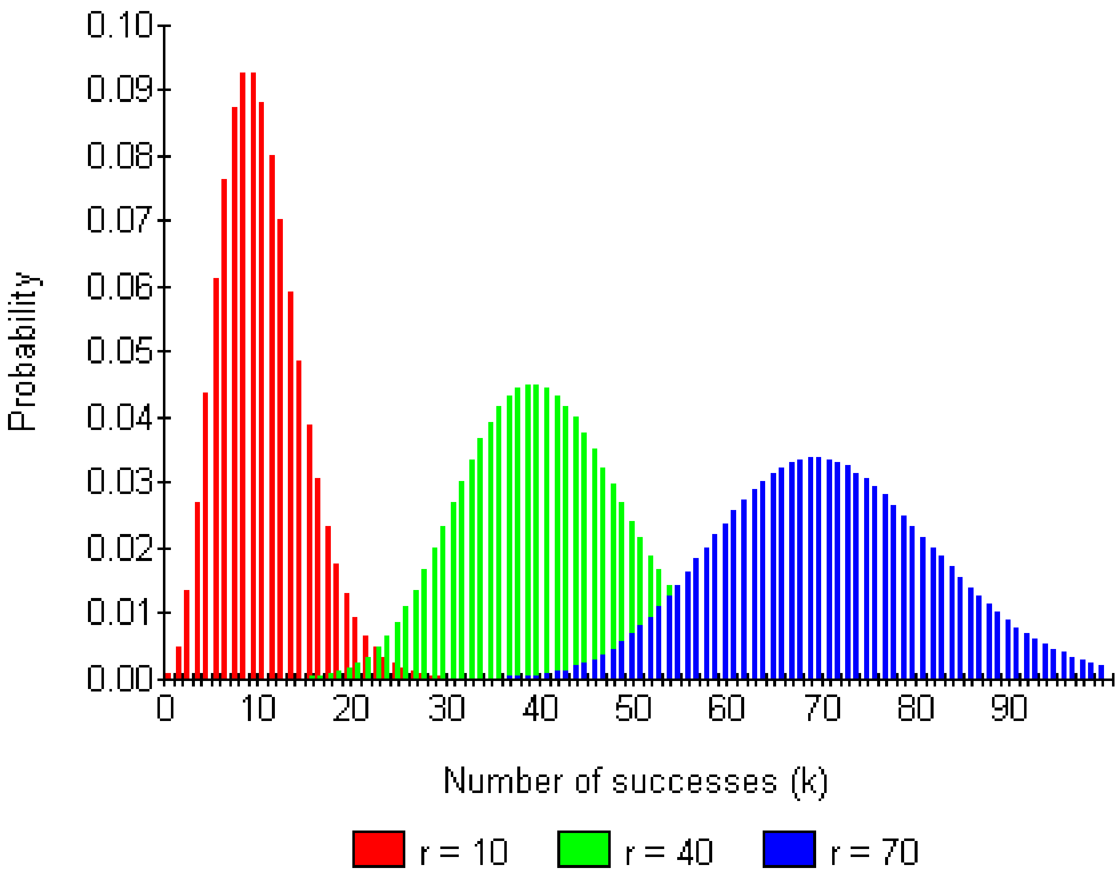
and the fluctuation on the number of disintegrations is given by the variance or the standard deviation of the

$$\sigma^2 \equiv \sum_{n=0}^{N} (n-\mu)^2 \cdot P_n = Npq$$

$$and$$

$$\sigma \equiv \sqrt{\sum_{n=0}^{N} (n-\mu)^2 \cdot P_n} = \sqrt{Npq}$$

Binomial Distribution



☞ **What are the mean and standard deviation of a Binomial distribution?**

Poisson Distribution

The counting statistics related to nuclear decay processes is often more conveniently described by the Poisson distribution, is related to situations that involves a collection of multiple trails that satisfy the following conditions:

1. The number of trials, N, is very large, e.g., $N > 1$.
2. Each trail is independent.
3. The probability that each single trial is successful is a constant and approaches zero, $p << 1$. Therefore, the number of successful trials fluctuates around a finite number.

Binomial and Poisson Random Variable

When $N\rightarrow\infty$ and $p\rightarrow 0$, k follows the Poisson distribution, whose probability function is

Binomial distribution

$$\boxed{p(k|N,p) = \binom{N}{k} p^k (1-p)^{N-k}} \Rightarrow \boxed{p(k) = \frac{\mu^k}{k!} e^{-\mu}}$$

Poisson distribution

where μ is the expected value of the Poisson variable defined

$$\mu = \lim_{N\rightarrow\infty} N\,p$$

Poisson Distribution

The probability of having *n* successful trails can be approximated with the Poisson distribution.

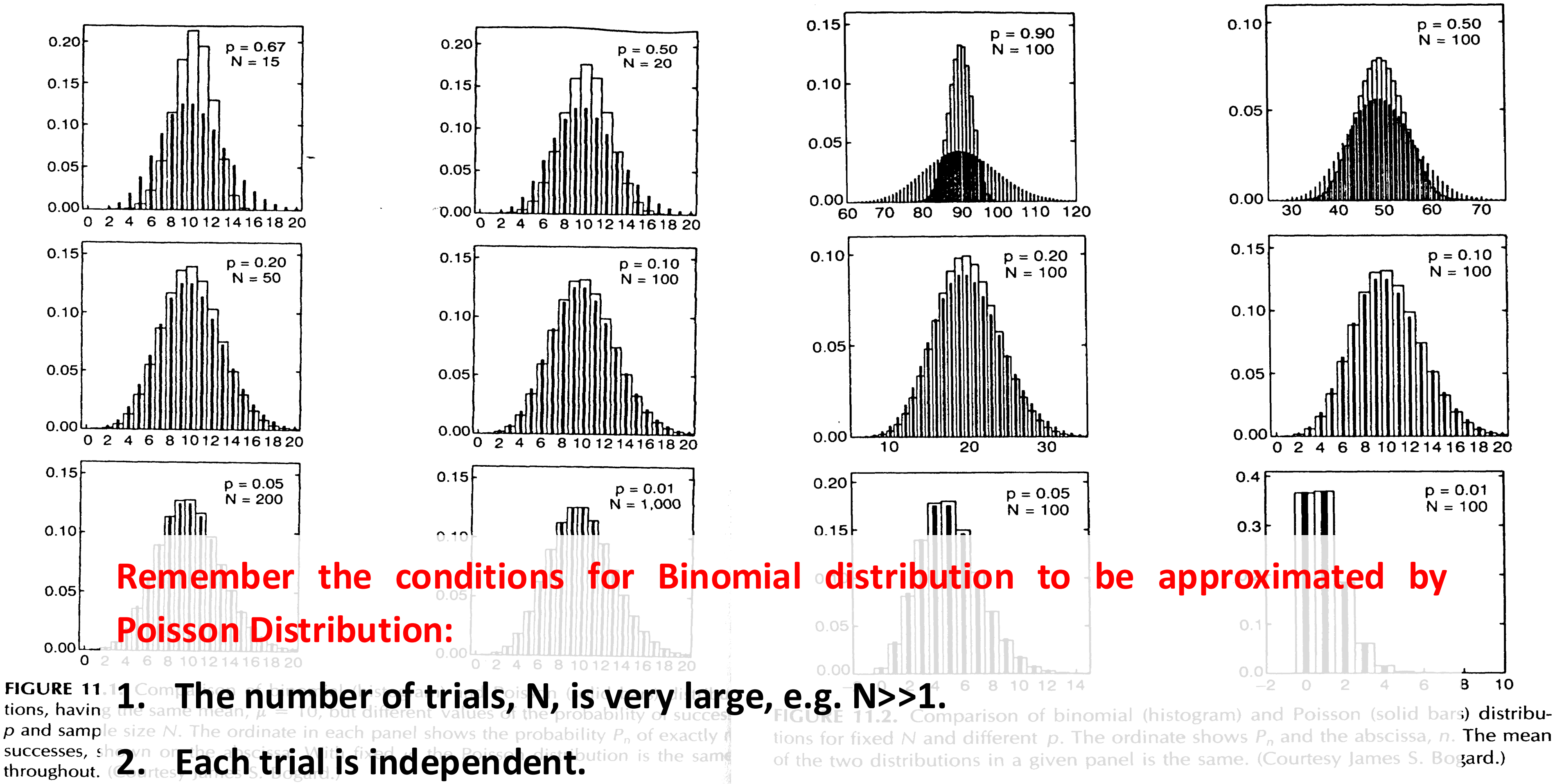
$$P(n \mid \mu) = \frac{\mu^n}{n!} e^{-\mu}$$

and the mean and the variance of number of successful trail are given by

$$Mean(n) = \mu = N \cdot p$$

$$Std(n) \equiv \sigma = \sqrt{\mu} = \sqrt{Np}$$

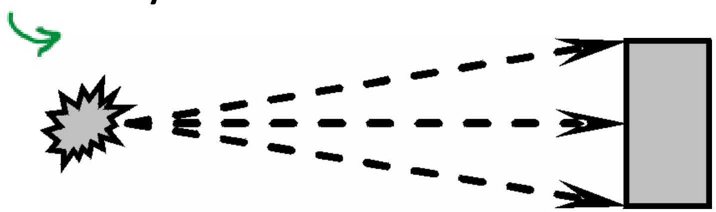
Poisson Distribution



Remember the conditions for Binomial distribution to be approximated by Poisson Distribution:

1. **The number of trials, N, is very large, e.g. N>>1.**
2. **Each trial is independent.**
3. **The probability that each single trial is successful is a constant and approaching zero, p<<1. So the number of successful trials is fluctuating around a finite number.**

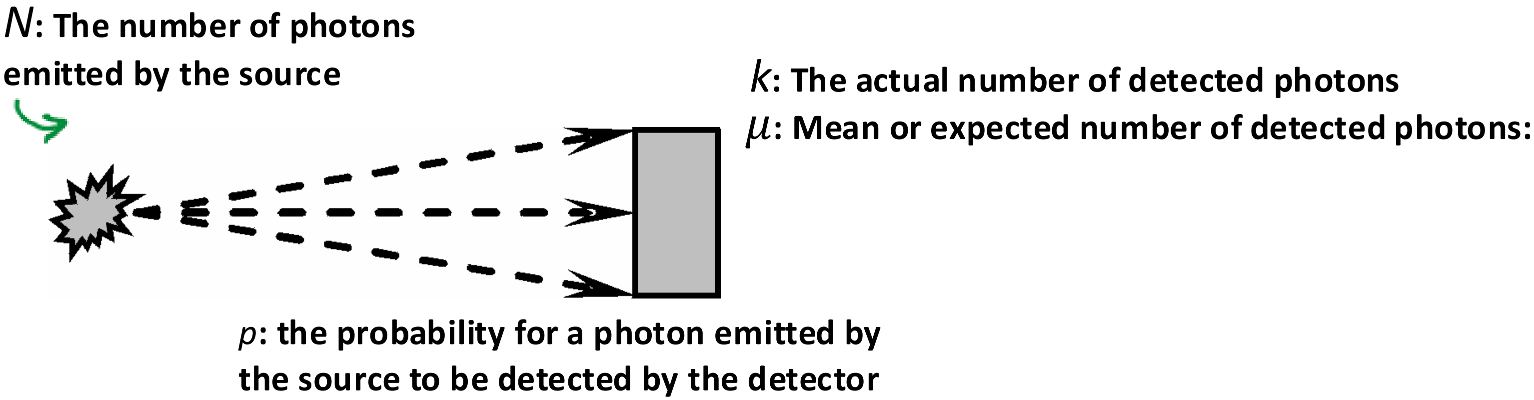
The Basic Idea of Statistical Reconstruction



We can estimate the underlying parameter N by the Maximum-Likelihood Estimation (MLE) strategy:

$$\hat{N}_{Maximum\ Likelihood\ estimation} = \arg\max_{N}\{P[k|N]\}$$

$P[k(N)]$ is call the Likelihood function: the probability of detecting k counts given there were a total of N gamma rays emitted by the source.



Since

$$P[k(N)]=\frac{(N\cdot p)^k}{k!}e^{-N\cdot p}$$

Then

$$\hat{N}_{Maximum\ likelihood} = \arg\max_N\{P[k|N]\}= \arg\max_N\left\{\frac{(N\cdot p)^k}{k!}e^{-N\cdot p}\right\},$$

alternatively

$$\hat{N}_{ML} = \arg\max_N\{\log[P[k|N]]\}$$

$$= \arg\max_N\{[k\cdot log(N\cdot p) - N\cdot p - log\, k!]\}$$

$$= \arg\max_N\{[k\cdot log(N\cdot p) - N\cdot p]\}$$

So the ML estimate of the number of gamma rays emitted by the source is

$$\hat{N}_{ML} = \arg\max_N\{\log[P[k|N]]\} = \arg\max_N\{[k\cdot \log(N\cdot p) - N\cdot p]\}$$

Image Reconstruction and General Linear Inverse Problem

An Emission Tomography System

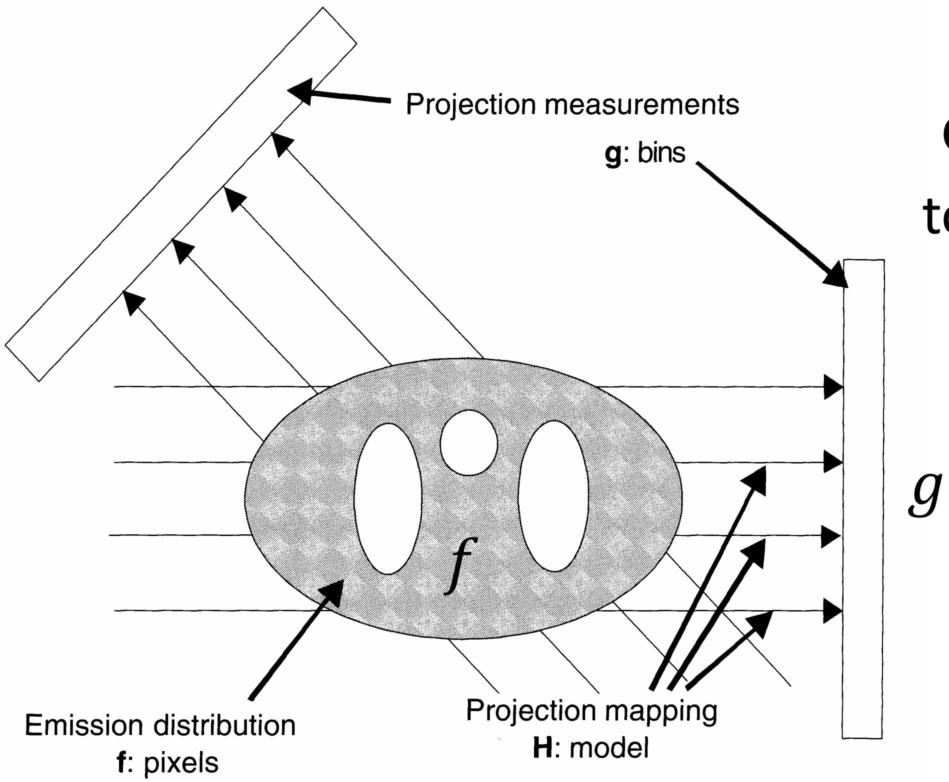


FIGURE 1 A general model of tomographic projection in which the measurements are given by weighted integrals of the emitting object distribution.

Given a measured data set g, how to formulate the ML estimator of f?

$$\hat{f}_{ML} = \arg\max P(\boldsymbol{g}|\boldsymbol{f})$$

Then how to evaluate $P(\boldsymbol{g}|\boldsymbol{f})$?

Image Reconstruction and General Linear Inverse Problem

Linear imaging systems – An alternative representation

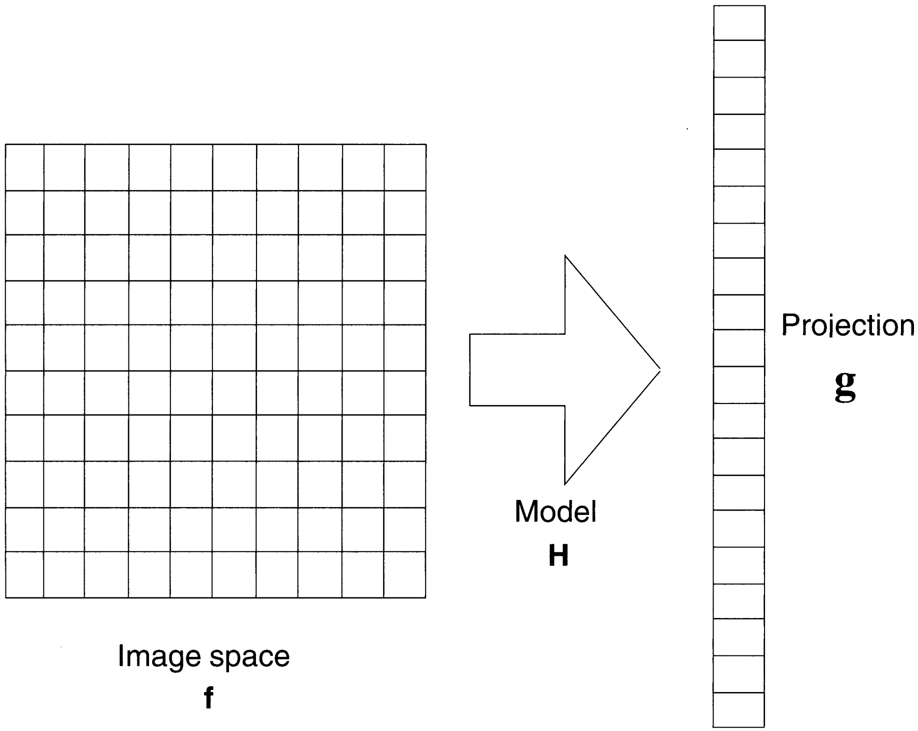


FIGURE 2 A discrete model of the projection process.

$$\bar{\mathbf{g}} = \mathbf{Hf}$$

Poisson Statistics of the Projection Data

Considering that the measured projection data, $\boldsymbol{g}=(g_1, g_2, g_3,\ldots, g_M)$, is a collection of independent Poisson random numbers, the probability of the measured projection data $\boldsymbol{g}$ may be written as

$$p(\boldsymbol{g}|\boldsymbol{f}) = p(\boldsymbol{g}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m}$$

where $\bar{g}_m$ is the expected value for the number of counts on detector pixel no. m.

$$\bar{g}_m = \sum_{n=1}^{N} f_n \boxed{p_{mn}}$$

In the context of emission tomography, p_{mn} is the probability that a gamma ray generated at a source pixel n is detected by detector element m.

A Typical Imaging System Described in Matrix Form

$$\begin{pmatrix} \bar{g}_1 \\ \bar{g}_2 \\ \bar{g}_3 \\ \vdots \\ \bar{g}_M \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1N} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2N} \\ p_{31} & p_{32} & p_{33} & \cdots & p_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{M1} & p_{M2} & p_{M3} & \cdots & p_{MN} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{pmatrix}$$

Expected number of photons detected on each detector element

Everything we know about the imaging system – The system response function.

The source object.

The number of photons being emitted from each source voxel

p_{mn} : the probability for a gamma ray emitted by the *n*'th source voxel to be detected by the *m*'th detector pixel.

The Maximum Likelihood Reconstruction

$$\hat{\boldsymbol{f}}_{ML} = \underset{f}{\operatorname{argmax}}\, L(\boldsymbol{f}, \boldsymbol{g}) = \underset{f}{\operatorname{argmax}} \log[L(\boldsymbol{f}, \boldsymbol{g})]$$

$$= \underset{f}{\operatorname{argmax}} \left\{ log \left[\prod_{m=1}^{M} \left(\frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m} \right) \right] \right\} = \underset{f}{\operatorname{argmax}} \left[\sum_{m=1}^{M} log \left(\frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m} \right) \right]$$

Consider all detector elements working independently, So the ML reconstruction for Poisson-distributed data is

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}} l(\boldsymbol{f}, \boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\sum_{m=1}^{M} (g_m \log \bar{g}_m - \bar{g}_m - \log g_m !) \right]$$

since g_m is not function of f, we get

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f}, \boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\sum_{m=1}^{M} (g_m \log \bar{g}_m - \bar{g}_m) \right]$$

$$\boxed{\bar{g}_m = \sum_{n=1}^{N} f_n p_{mn}}$$

$$\boxed{\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f}, \boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\sum_{m=1}^{M} \left(g_m \cdot \log \left(\sum_{n=1}^{N} f_n p_{mn} \right) - \left(\sum_{n=1}^{N} f_n p_{mn} \right) \right) \right]}$$

The Maximum Likelihood Reconstruction

Underlying/unknown image function to be estimated

The ML estimate

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\left(\boldsymbol{f}, \boldsymbol{g}\right) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\sum_{m=1}^{M} \left(g_m \cdot \log\left(\sum_{n=1}^{N} f_n p_{mn} \right) - \left(\sum_{n=1}^{N} f_n p_{mn} \right) \right) \right]$$

Measured data

The Maximum Likelihood Expectation Maximization (MLEM) Algorithm

The source (image) function $\boldsymbol{f}$ that has the maximum likelihood of giving rise to the observed data $\boldsymbol{g}$ can be found by the following iterative updating scheme

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{mn'} f_{n'}^{(old)}} p_{mn}, \qquad n = 1,2,\ldots,N$$

$where$

$m = 1,2,\ldots,M$ is the index of detector elements in the imaging system

$n = 1,2,\ldots,N$ is the index of source object pixels

$f_n^{(old)}$: the current estimate of the source function in pixel n

$f_n^{(new)}$: the updated estimate of the source pixel n

p_{nm}: the probability of a gamma ray generated in source pixel n and detected by the detector element m

g_m: the observed number of counts in detector pixel m

Shepp & Vardi, IEEE TMI, 1982.

The Maximum Likelihood Expectation Maximization (MLEM) Algorithm

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{mn'} f_{n'}^{(old)}} p_{mn}\,, \qquad n = 1,2,\ldots,N$$

Each iteration updates all elements in the source function *f* sequentially.

Each iteration is guaranteed to provide a new image function that has an *increased likelihood* compared to the old image function (unless the maximum likelihood solution has been achieved).

IEEE Trans Med Imaging, Maximum likelihood reconstruction for emission tomography L A Shepp, Y Vardi, 1982.

Structure of MLEM Algorithm

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{n'm} f_{n'}^{(old)}} p_{nm}, \quad n = 1,2,\ldots,N$$

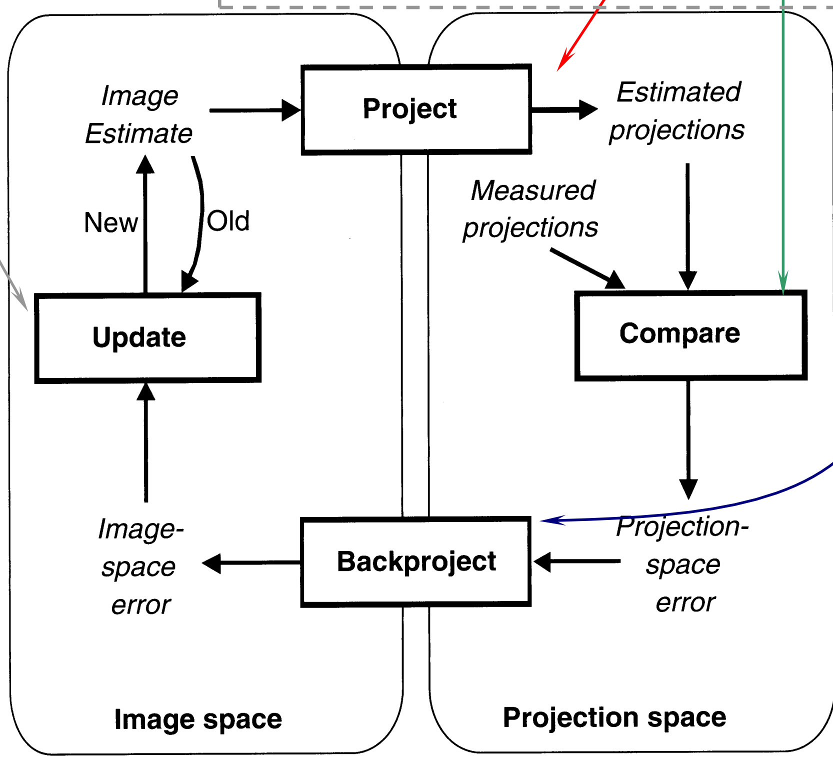


FIGURE 6 Flowchart of a generic iterative reconstruction algorithm.

An Example of PET Image Reconstruction

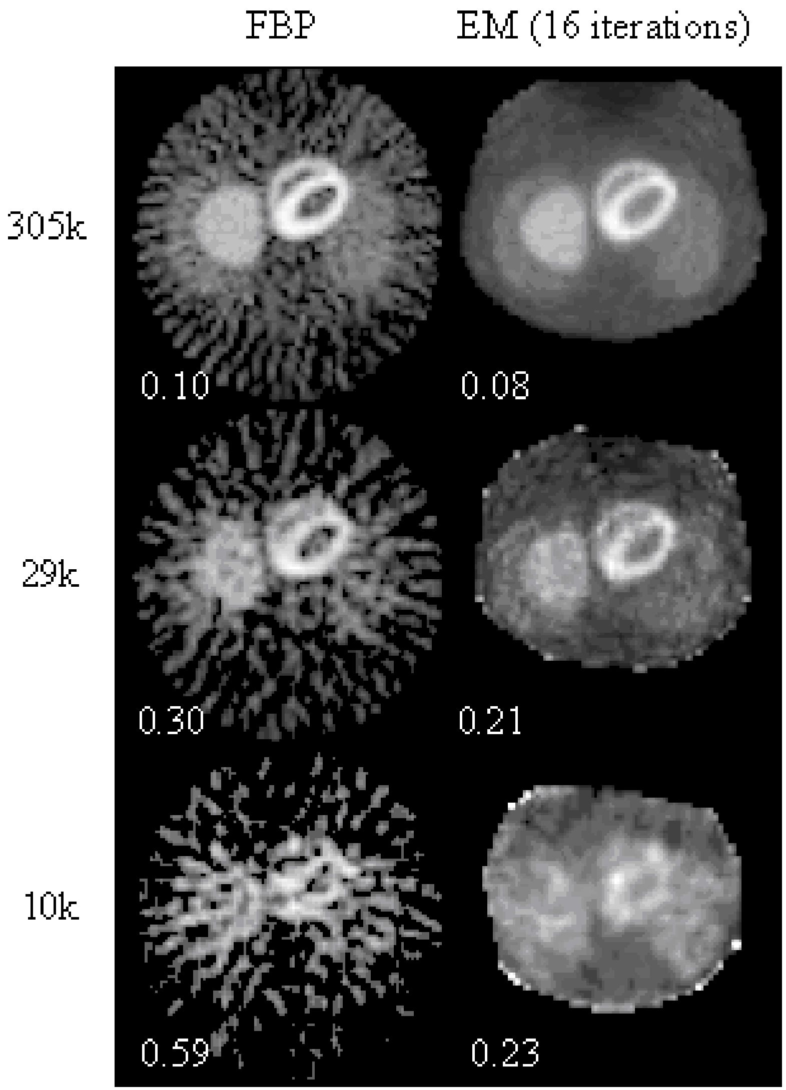


Figure 3: Comparison of EM reconstruction with FBP for different total counts. Note particular the streaking and noise appearance at low counts using FBP. Figures indicate the noise (standard deviation / mean) for a region in the liver.

http://web.uchile.cl/vignette/borrar3/alasbimn/CDA/sec_a/0,1205,SCID%253D470%2526PRT%253D455%2526LNID%253D10,00.html

Properties of the MLEM Algorithm

Advantages

- Relatively *simple* but *highly useful*.
- Provides room for incorporating a wide range of physical effects and detector system characteristics and therefore provide an *improved image quality* for low statistics data.
- *Consistent and predictable behavior*.
- Automatically incorporates *non-negativity constraint*. So the image values are always non-negative.
- Provides *theoretically optimum solution* (Be careful when saying this!!).

Properties of the MLEM Algorithm

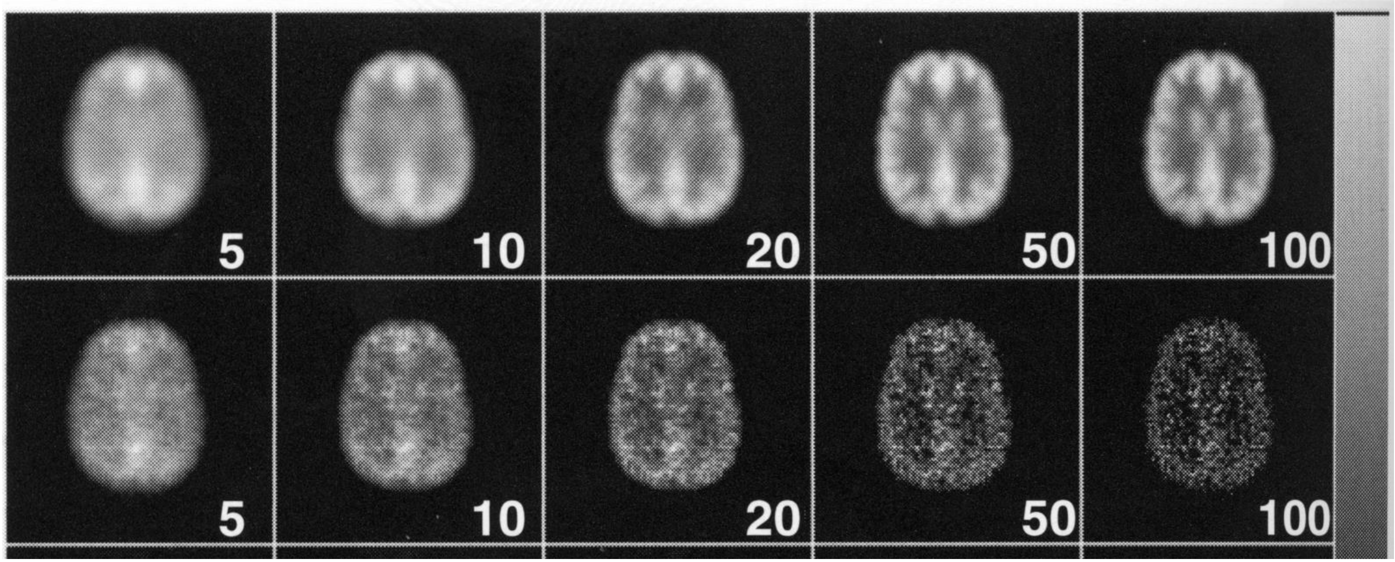
Shortcomings

- *Relatively slow* convergence. For typical 3-D reconstruction, it takes several tens of iterations to converge. In principle, each iteration takes similar amount of computation to that for a complete FBP reconstruction!
- Tends to yield *noisy reconstructions* when projection data has low statistics.

What exactly is the problem?

The algorithm tends to fit to the noise in the (observed) projection data!

Properties of the MLEM Algorithm



With noise free data

With noisy data

Properties of MLEM Algorithm

How to improve?

Adding extra (or sometime called *a priori*) information in the reconstruction process.

This can be done using two numerically equivalent approaches:

- This can be done by trying to discourage possible image solutions that have large fluctuations (noisy images). – Penalized Maximum Likelihood (PML) methods.
- Using the Bayes' law to incorporate the *a priori* information – Maximum a posteriori (MAP) methods.

Further Improvements – Penalized ML Algorithms

The basic idea:

Instead of finding the original image by maximizing the likelihood function:

Probability of a measurement *g* given the underlying image *f*

$$\hat{\boldsymbol{f}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, \boxed{p(\boldsymbol{g} \mid \boldsymbol{f})} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, \boxed{l(\boldsymbol{f}, \boldsymbol{g})}$$

The so-called log-likelihood function

We can use a different selection criterion

$$\hat{\boldsymbol{f}} = \underset{\boldsymbol{f}}{\operatorname{argmax}} \boxed{\left[l(\boldsymbol{f}, \boldsymbol{g}) - \boxed{\beta U(\boldsymbol{f})} \right]}$$

The so-called penalty function that has increased value when an undesired imaging feature presents in the solution.

β is a scaling constant that controls "how strong the penalty is for the presence of a given feature"

So by minimizing, we are selecting an image by dis-encouraging the undesired features to be presented in the image.

Penalized Maximum Likelihood (PML) reconstruction …

An Example Penalized ML Algorithms

An example is to use the so-called **roughness penalty** function that measures the differences between neighboring pixels in the image, as follows

$$\boxed{\hat{\boldsymbol{f}}_{PML} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}}\left[l(\boldsymbol{f},\boldsymbol{g}) - \beta \sum_{j=1}^{N} \sum_{k=\mathcal{N}_j} \phi(f_j - f_k)\right],}$$

where

$\phi(f_j - f_k)$ may be defined as $a_{jk}(f_j - f_k)^2$,

$\mathcal{N}_j$ refers to the collection of neighboring pixels of pixel *j*, and

a_{jk} is a user-defined weighting factor to reflect different preferences on how the user would like to penalize the differences between the target pixel j and its neighboring pixels.

So the above Penalized ML Algorithm becomes

$$\boxed{\hat{\boldsymbol{f}}_{PML} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}}\left[l(\boldsymbol{f},\boldsymbol{g}) - \beta \sum_{j=1}^{N} \sum_{k=\mathcal{N}_j} a_{jk}(f_j - f_k)^2\right],}$$

Further Improvements –
Maximum A Posteriori (MAP) Algorithms

Alternatively, we may consider the unknown image, ***f***, as a random variable that follows a statistical law $p(\boldsymbol{f})$.

The Bayis's law:

if the underlying image is *f*, the conditional probability of observing the measurement *g* is

a priori probability

$$\boxed{p(\boldsymbol{f} \mid \boldsymbol{g})} = \frac{\boxed{p(\boldsymbol{g} \mid \boldsymbol{f})}\boxed{p(\boldsymbol{f})}}{p(\boldsymbol{g})}$$

a posteriori probability

This leads to the maximum a posteriori (MAP) estimation …

Further Improvements –

Bayesian Estimation and Maximum A Posteriori (MAP) Algorithms

The maximum a posteriori (MAP) approach…

$$\hat{\boldsymbol{f}}_{MAP} = \underset{\boldsymbol{f}}{\operatorname{argmax}} \log p(\boldsymbol{f} \mid \boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \log \frac{p(\boldsymbol{g} \mid \boldsymbol{f})\, p(\boldsymbol{f})}{p(\boldsymbol{g})}$$

$$= \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\log \boxed{p(\boldsymbol{g} \mid \boldsymbol{f})} + \log \boxed{p(\boldsymbol{f})}\right]$$

The likelihood function

The *a priori* probability of the data. This is where you can fold in your prior knowledge about the underlying image … for example, the image “should” be relatively smooth …

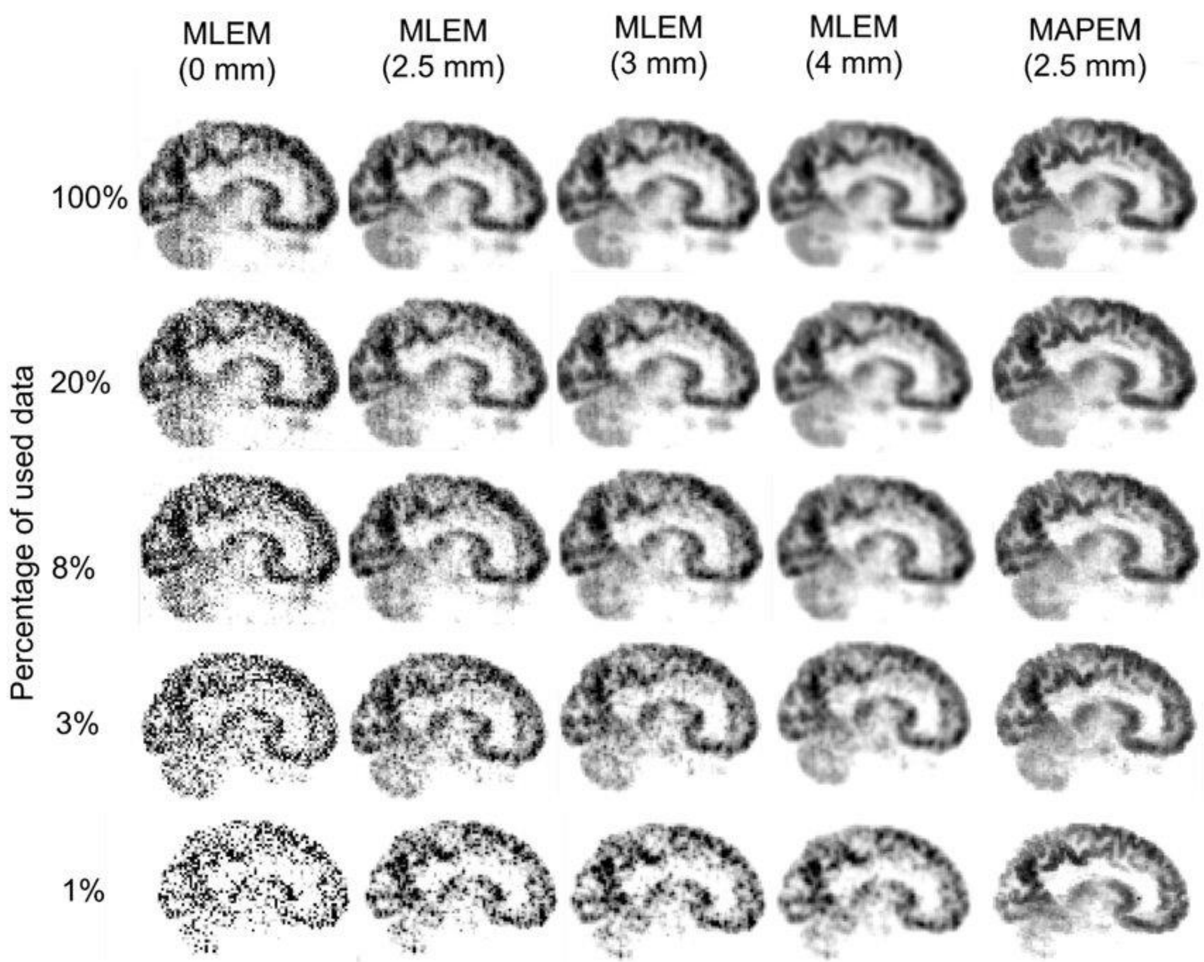
This is numerically equivalent to PML approach if we write

$$p(\boldsymbol{f}) = -\beta U(\boldsymbol{f}),$$

therefore

$$\hat{\boldsymbol{f}}_{\mathrm{PML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}} [\log p(\boldsymbol{g}|\boldsymbol{f}) - \beta U(\boldsymbol{f})] \Leftrightarrow \hat{\boldsymbol{f}}_{\mathrm{MAP}} = \underset{\boldsymbol{f}}{\operatorname{argmax}} [\log p(\boldsymbol{g}|\boldsymbol{f}) + \log p\,(\boldsymbol{f})].$$

Example PET Images Reconstructed with MLEM and PML Algorithms



The reconstruction of various percentages of the FDG PET data using the MLEM and MAP-EM algorithms. The MLEM reconstructions have each been post-smoothed using a Gaussian kernel with 0, 2.5, 3, and also 4 mm FWHM. All PET images are shown with the same grey scale intensity range.

"PET image reconstruction using multi-parametric anato-functional priors", *Mehranian, et al., PMB, 2017.*

Example PET Images Reconstructed with MLEM and PML Algorithms

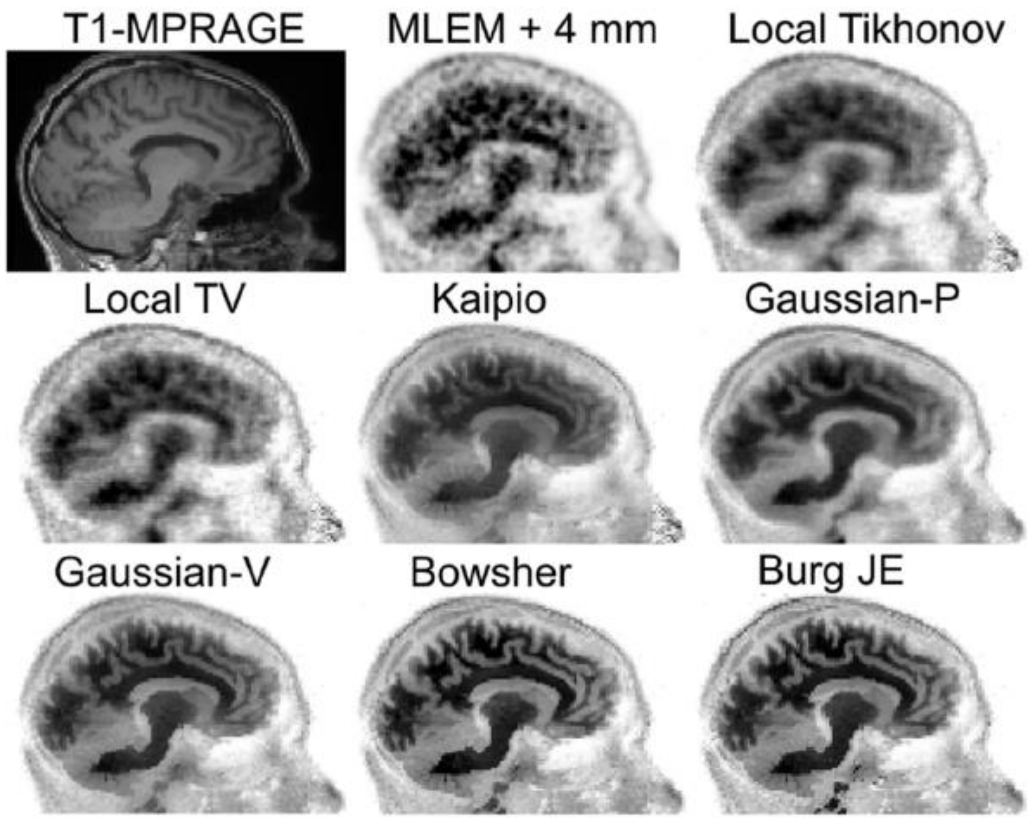


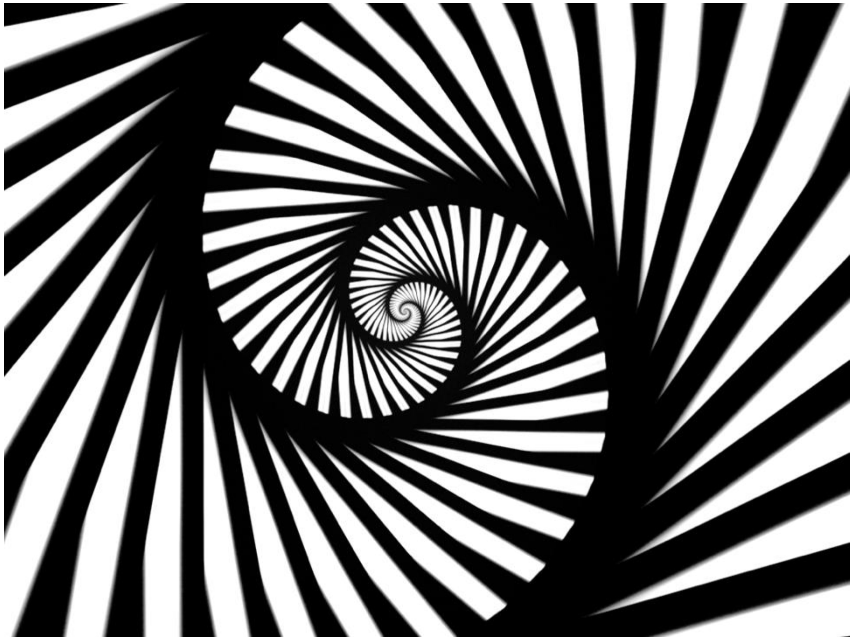
Figure 10. The reconstruction results of the 5 min of the [^{18}F]florbetaben dataset using different algorithms. All PET images are shown with the same displaying window.

"PET image reconstruction using multi-parametric anato-functional priors", *Mehranian, et al., PMB, 2017.*

Chapter 3: Mathematical Preliminaries

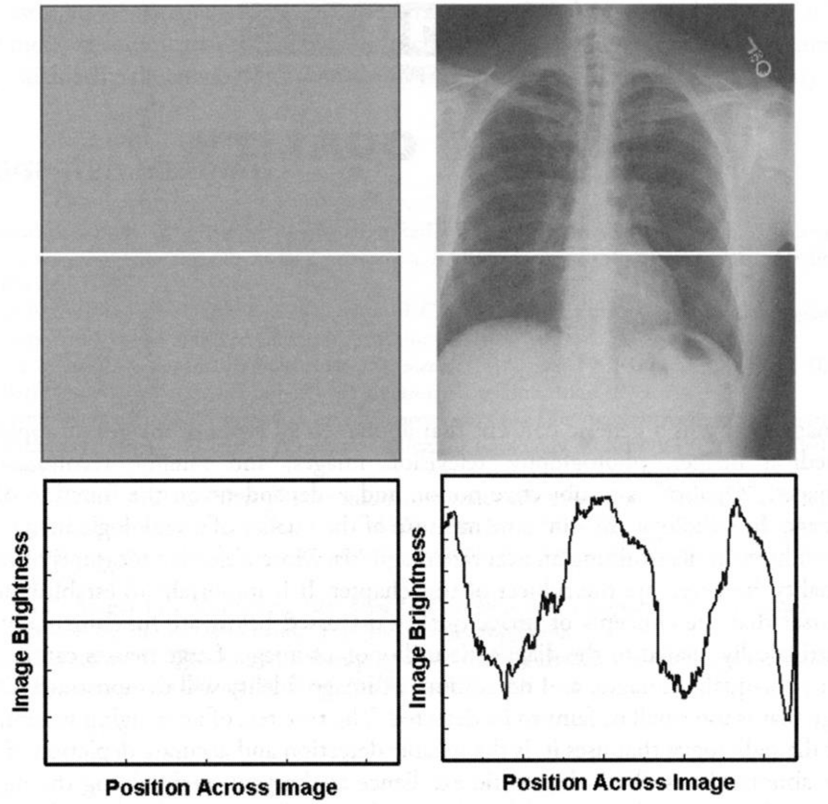
Image Quality

Contrast



Contrast

- What is contrast?
- The difference in the image gray scale between closely adjacent regions of the image.
- Medical image contrast is the result of many steps during image acquisition, processing and display

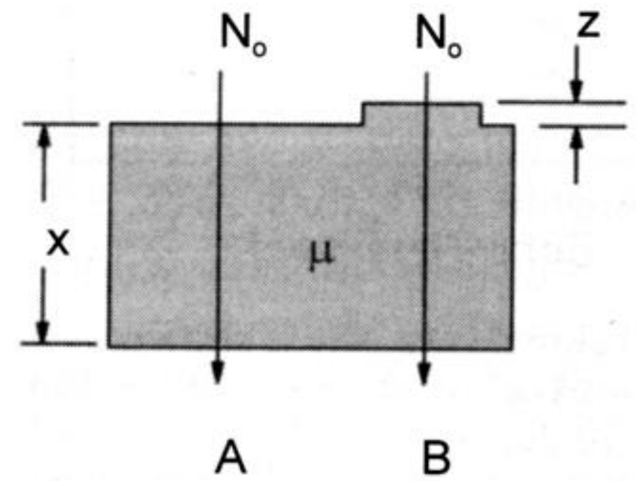


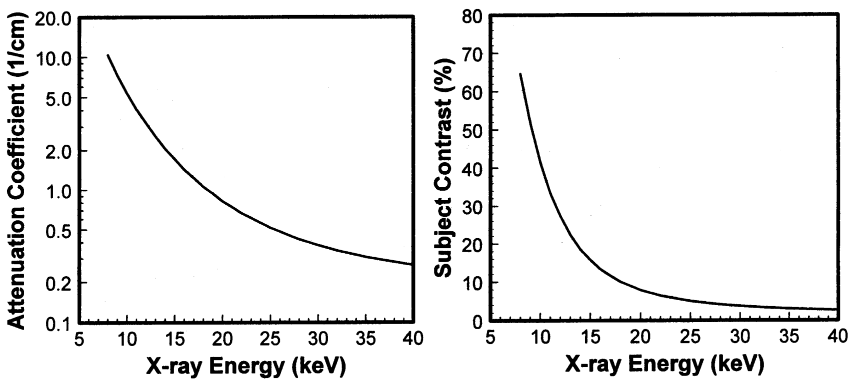
Subject (or Object) Contrast

- Difference in some aspects of the signal prior to it being recorded

- Consequence of fundamental differences in the object, e.g., in x-ray intensity based on attenuation

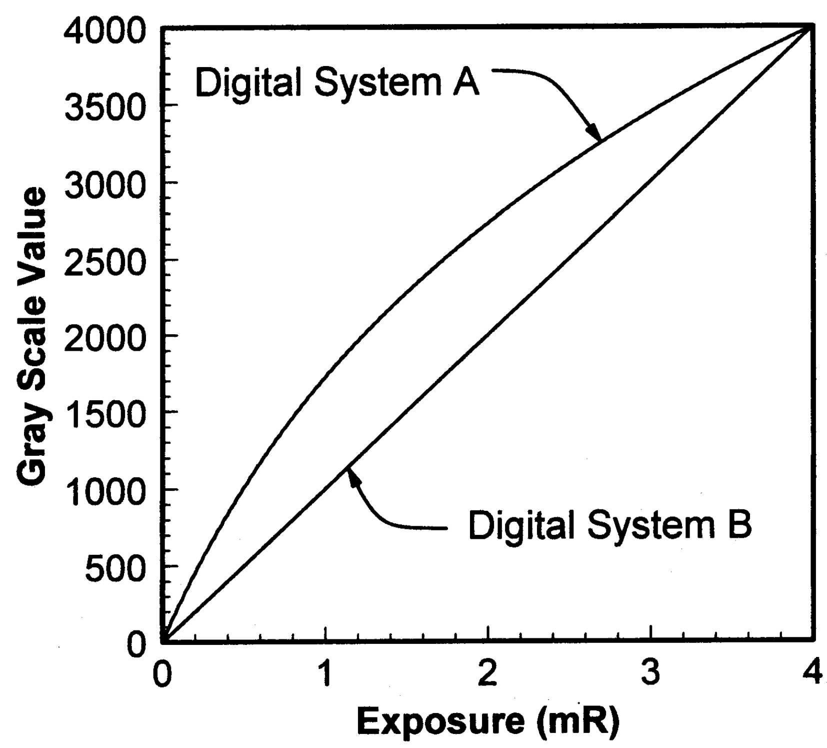
- C = (A-B)/A





Detector Contrast

- Detector Contrast (C_d)
- A detector's characteristics play an important role in producing contrast in the final image
- C_d determined principally by how the detector 'maps' detected energy into the output signal
- Characteristic curve: input radiation exposure to output value (analog or digital)



Modulation

Recall that an arbitrary signal can be decomposed into the weighted sum of periodic signals, such as sinusoidal waves, so study the response of an imaging system to such signals provide a effective approach for quantifying the image quality.

Modulation

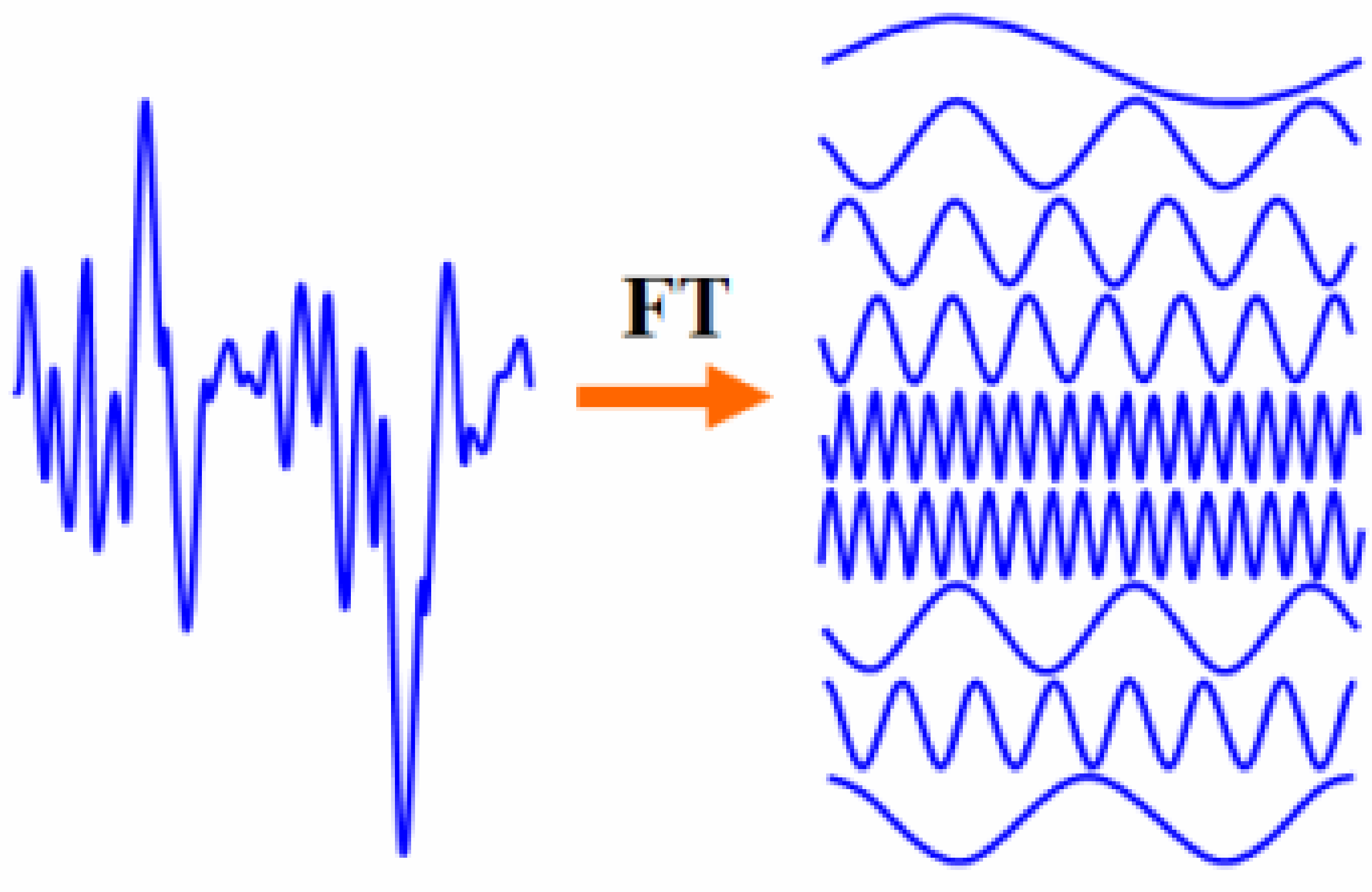
- The modulation m_f is an effective way to quantify the contrast of a periodic signal

$$m_f = \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}}.$$

- In general, m_f is referred to as the contrast of a periodic signal f(x,y) relative to its average value.

- So within two signals, f(x,y) and g(x,y), with the same average value , f(x,y) is said to have more contrast if $m_f > m_g$.

Continuous Fourier Transform



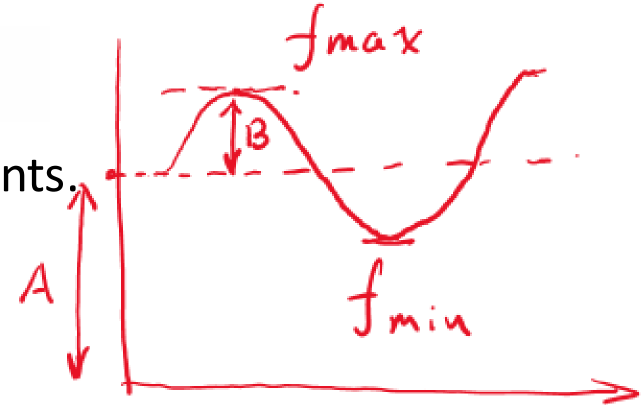
www.revisemri.com

Modulation

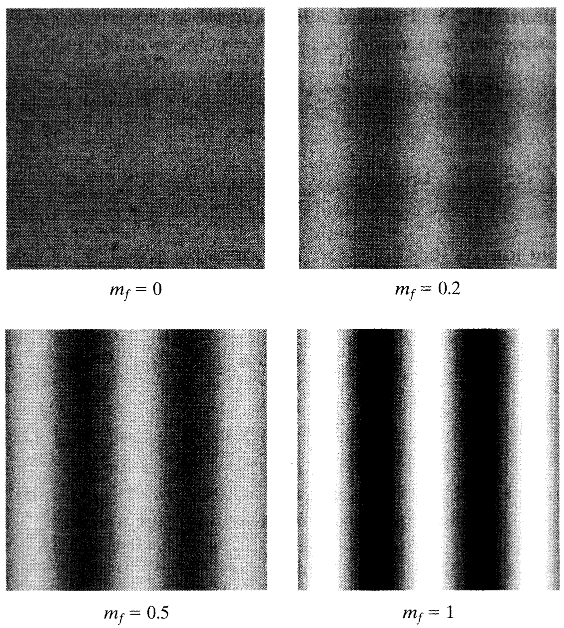
- Suppose an input signal function

$$f(x,y) = A + B\sin(2\pi u_0 x)\,,$$

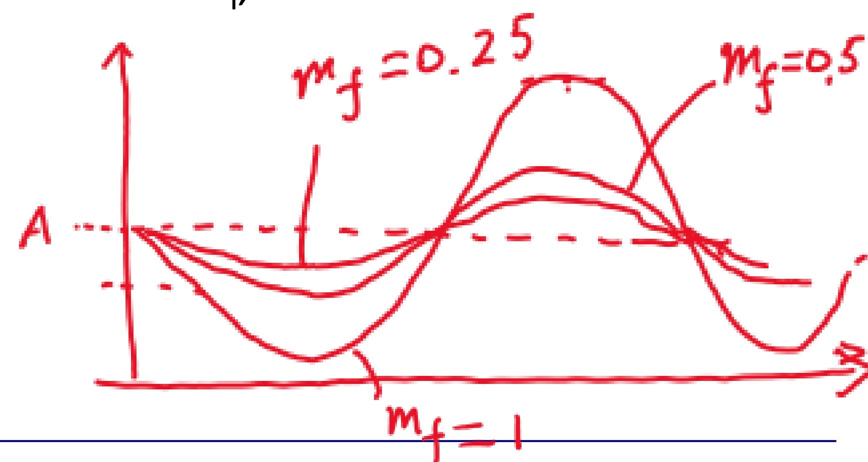
where A>B and both are non-negative constants.



$$m_f = \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}}\,. \Rightarrow m_f = \frac{B}{A}\,.$$



Greater m_f, more contrast



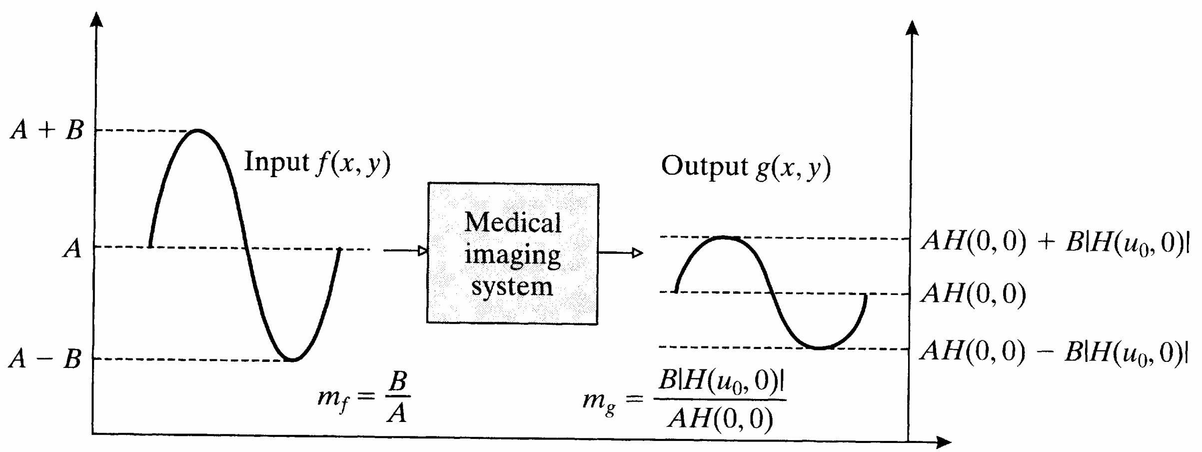
Modulation

Now let the following signal pass through an LSI imaging system,

$$f(x,y) = A + B\sin(2\pi u_0 x) = A + \frac{B}{2j}\left[e^{j2\pi u_0 x} - e^{-j2\pi u_0 x}\right],$$

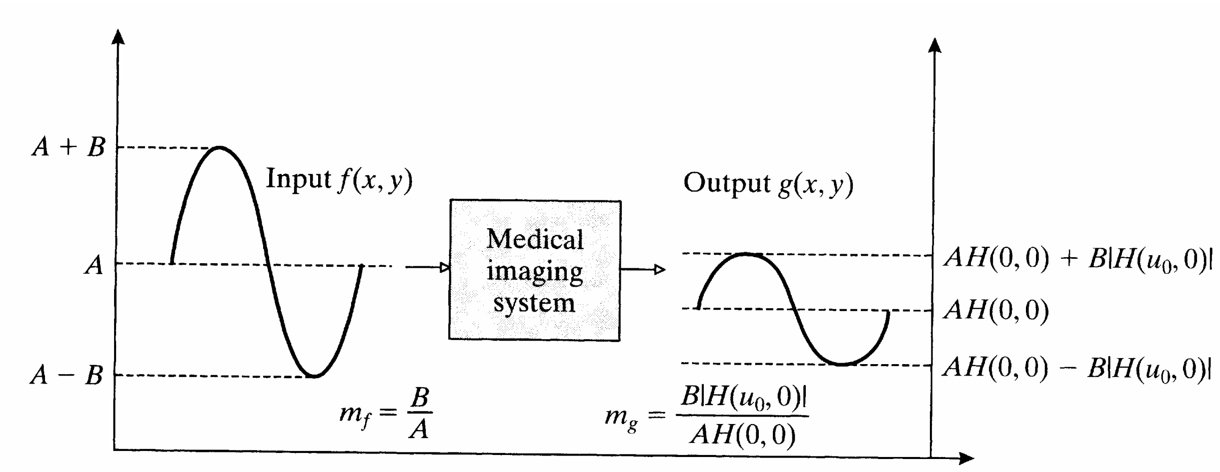
and consider the system impulse response function h(x,y) as circularly symmetric, then

$$g(x,y) = AH(0,0) + B\,|H(u_0,0)|\sin(2\pi u_0 x).$$



$$m_f = \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}}$$

Modulation



Since

$$g_{min} = AH(0,0) - B|H(u_0, 0)| \quad \text{and} \quad g_{\max} = AH(0,0) + B|H(u_0, 0)|,$$

then the modulation of the input and output signals are

$$m_f = \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}} = \frac{B}{A}, \quad \text{and} \quad m_g = \frac{g_{\max} - g_{\min}}{g_{\max} + g_{\min}} = \frac{B|H(u_0,0)|}{AH(0,0)} = m_f \frac{|H(u_0,0)|}{H(0,0)}$$

$$\frac{m_g}{m_f} = \frac{H(u_0,0)}{H(0,0)} \leq 1$$

Modulation Transfer Function (MTF)

- In case of non-isotropic impulse response function (h(x,y) is not circularly symmetric), the MTF can be defined as

$$\mathrm{MTF}(u,v) = \frac{m_g}{m_f} = \frac{|H(u,v)|}{H(0,0)},$$

- A typical MTF of an imaging system

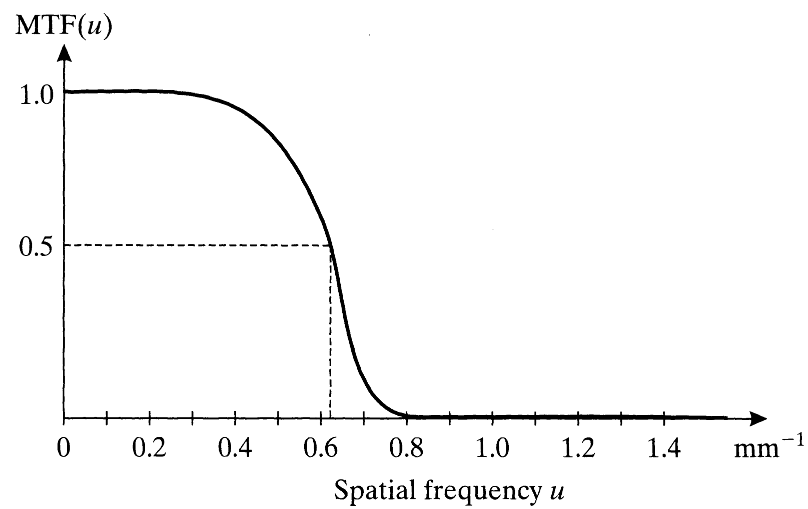
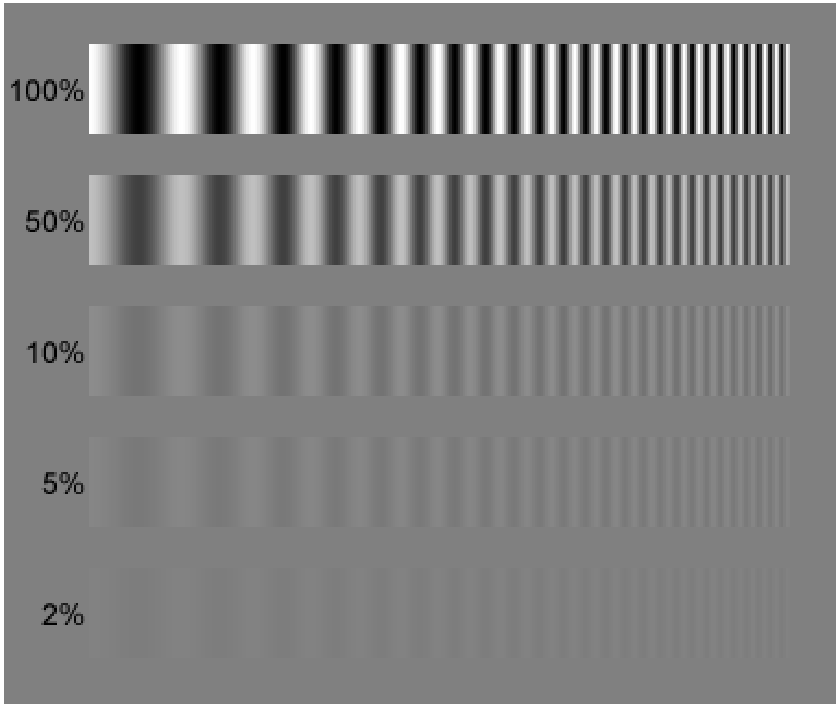


Figure 3.3
A typical MTF of a medical imaging system.

Modulation Transfer Function (MTF)

In case of non-isotropic impulse response function (h(x,y) is not circularly symmetric), the MTF can be defined as

$$\mathrm{MTF}(u,v) = \frac{m_g}{m_f} = \frac{|H(u,v)|}{H(0,0)},$$



Uniform MTF

Basic Fourier Transform Pairs

Basic Fourier transform pairs

Signal	Fourier Transform
1	$\delta(u, v)$
$\delta(x, y)$	1
$\delta(x - x_0, y - y_0)$	$e^{-j2\pi(ux_0 + vy_0)}$
$\delta_s(x, y; \Delta x, \Delta y)$	$\mathrm{comb}(u\Delta x, v\Delta y)$
$e^{j2\pi(u_0 x + v_0 y)}$	$\delta(u - u_0, v - v_0)$
$\sin[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2j}[\delta(u - u_0, v - v_0) - \delta(u + u_0, v + v_0)]$
$\cos[2\pi(u_0 x + v_0 y)]$	$\frac{1}{2}[\delta(u - u_0, v - v_0) + \delta(u + u_0, v + v_0)]$
$\mathrm{rect}(x, y)$	$\mathrm{sinc}(u, v)$
$\mathrm{sinc}(x, y)$	$\mathrm{rect}(u, v)$
$\mathrm{comb}(x, y)$	$\mathrm{comb}(u, v)$
$e^{-\pi(x^2 + y^2)}$	$e^{-\pi(u^2 + v^2)}$

Modulation Transfer Function (MTF)

- In order to fully quantify the response of an LSI for an arbitrary signal, f(x,y), we would need to know the response of the system to sinusoidal signals at different frequencies.

- Modulation Transfer Function (MTF).

$$\mathrm{MTF}(u) = \frac{m_g}{m_f} = \frac{|H(u,0)|}{H(0,0)}\,.$$

- MFT is, in effect, the "frequency response function" of a given imaging system. It is normally evaluated for positive frequencies only.

- Most imaging systems lead to decreased contrast, so that

$$0 \leq \mathrm{MTF}(u) \leq \mathrm{MTF}(0) = 1, \qquad \text{for every } u\,,$$

Modulation Transfer Function (MTF)

$$MTF(u) = \frac{m_g}{m_f} = \frac{|H(u,0)|}{H(0,0)}$$

If an imaging system is designed to faithfully reproduce a DC signal, then ((0,0)=1. So, the MTF becomes

$$MTF(u) = \frac{m_g}{m_f} = |H(u,0)| \ .$$

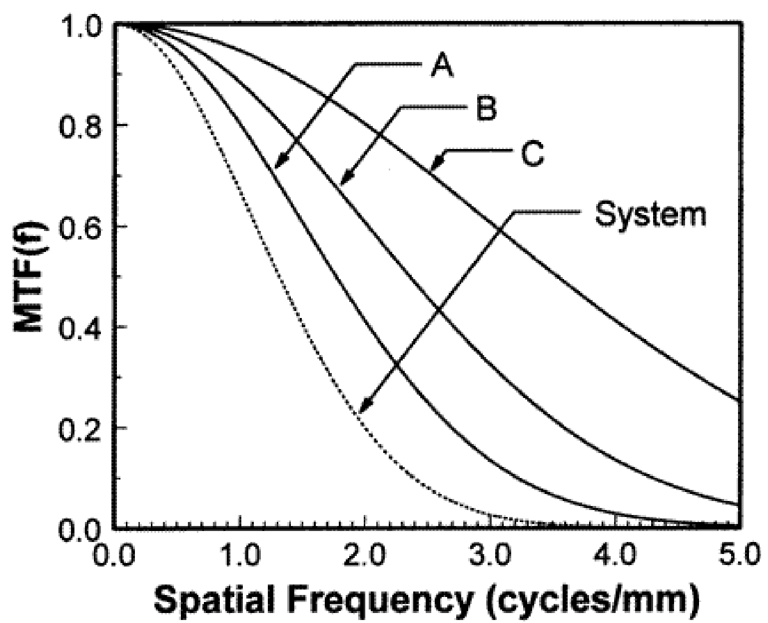
System Cascade

Given a cascade of LSI systems, the output of a signal through the overall system is given by

$$g(x, y) = h_K(x, y) * \cdots * (h_2(x, y) * (h_1(x, y) * f(x, y))) .$$

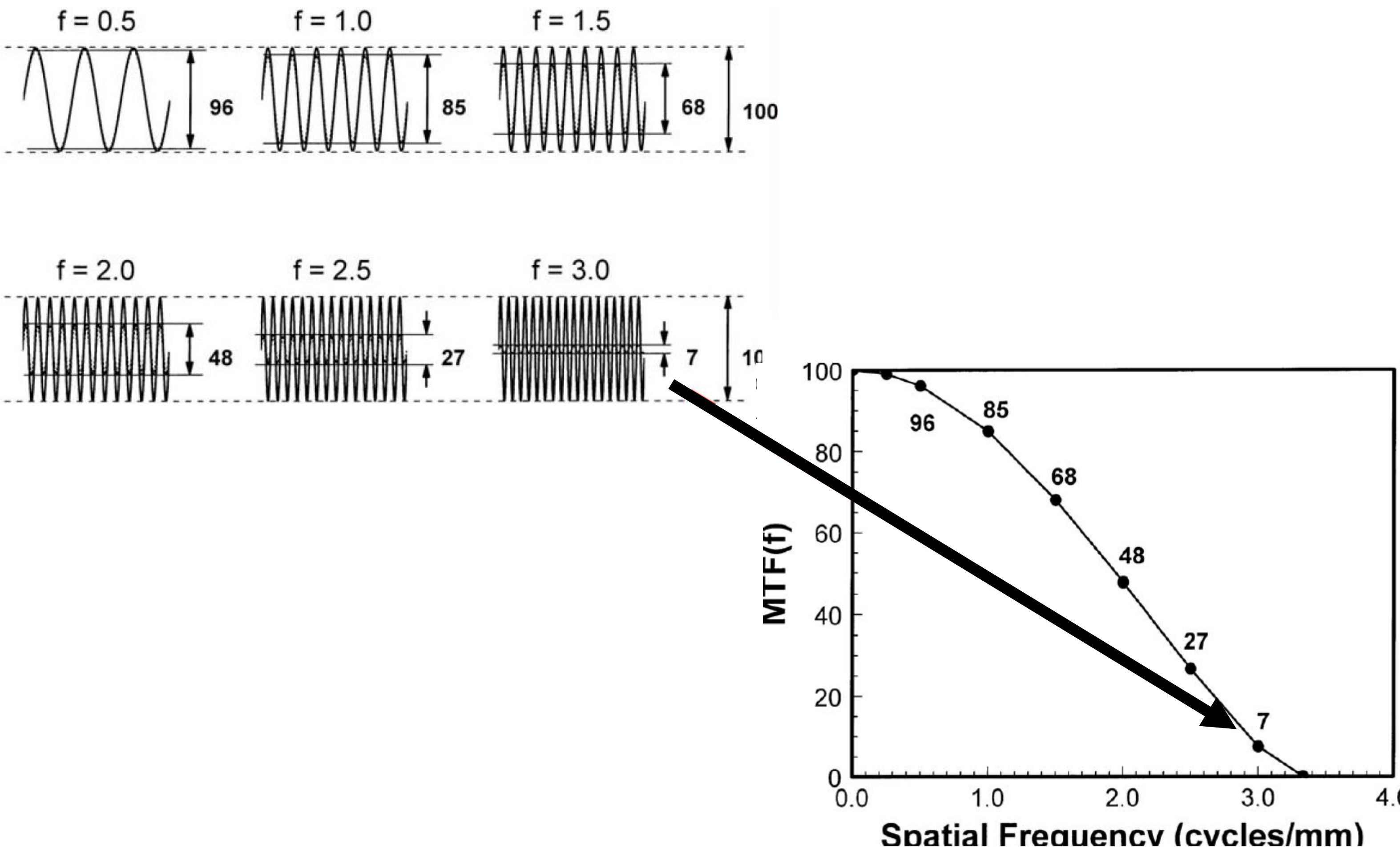
The overall MTF is the product of the MTF for sub-systems:

$$MTF_{total}(u) = \prod_{i=1}^{i=I} MTF_i(u)$$



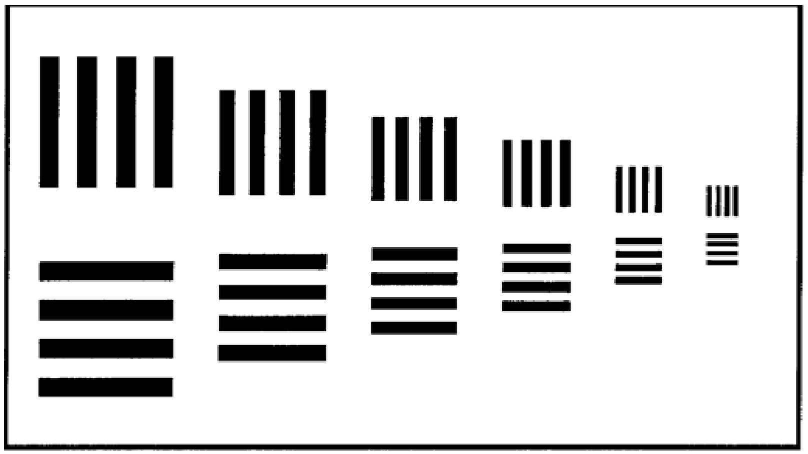
How is the Modulation Transfer Function Measured?

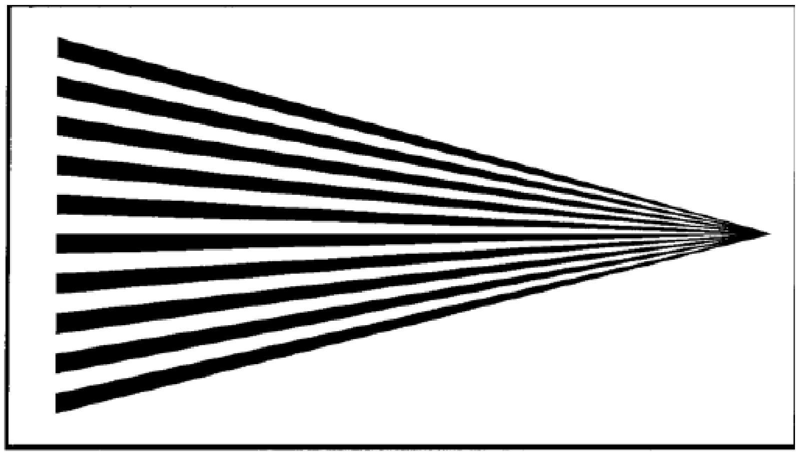
- Modulation Transfer Function (MTF).

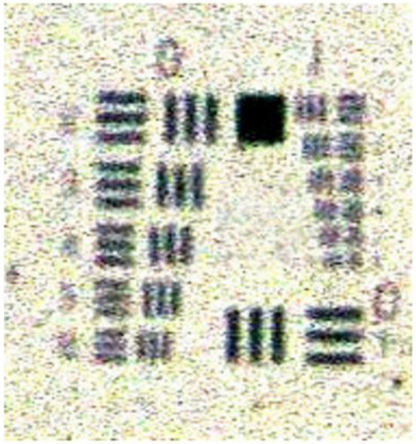


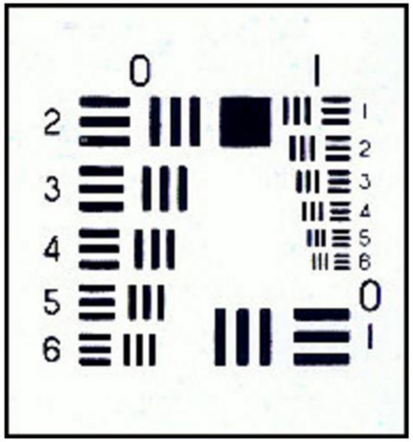
How is the Modulation Transfer Function Measured?

Standard Test Chart for Determining the Modulation Transfer Function (MTF) of an Imaging System.



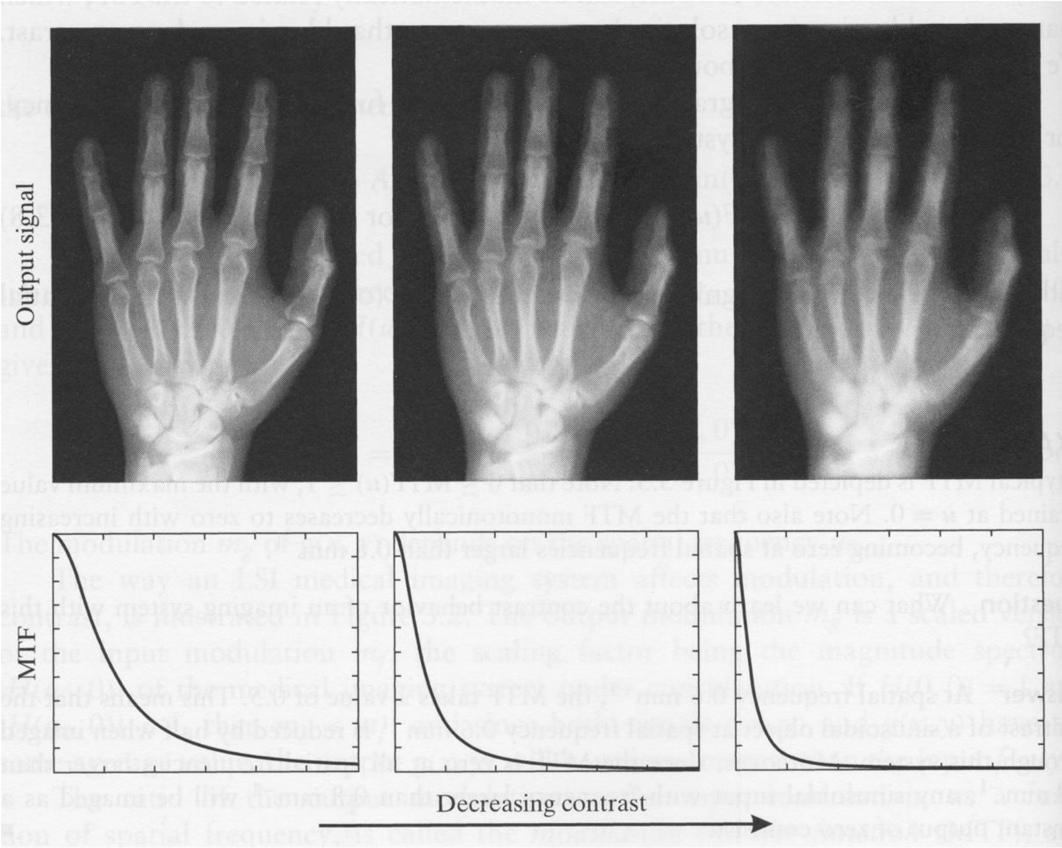






Example of the Modulation Transfer Function

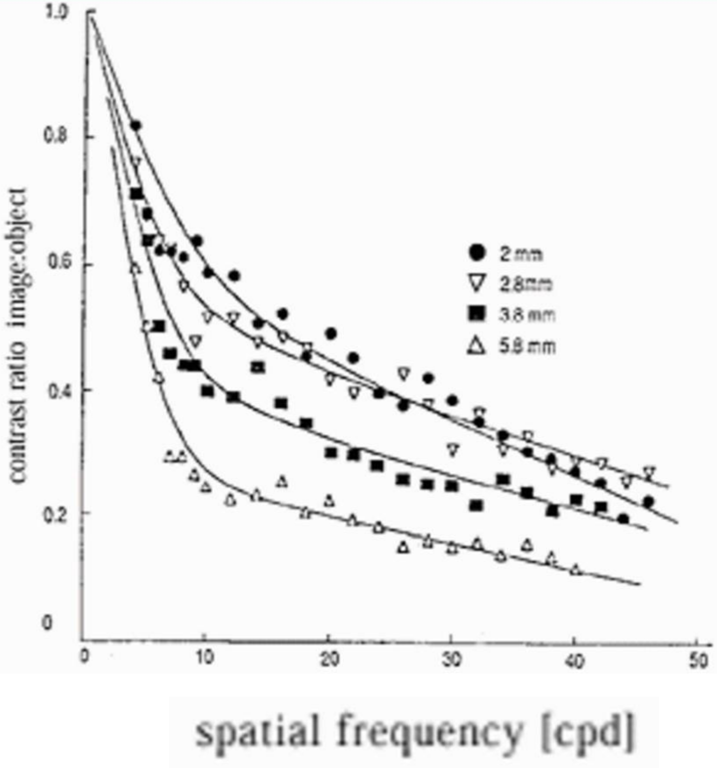
- An example of the same signal passing through three imaging systems with decreasingly poor MTF, which leads to decreasing contrast in images



Example of the Modulation Transfer Function

Optical modulation transfer function (MTF) of the human eye

- MTF is measured directly with sinewave gratings.
- The optical modulation transfer function (MTF) can be interpreted as Fourier transform of the optical LSF.



Filtered Back-projection (revisited)

The sharp boundary of the Ram-Lak filter often makes the spatial domain filter kernel oscillatory.

It sometimes introduces the "ringing" artifacts in reconstructed images.

This can be effectively resolved by the Shepp and Logan filter

$$H_{SL}(\omega) = \begin{cases} |\omega| \sin\left(\dfrac{\omega}{4B}\right), & |\omega| < 2\pi B \\ 0, & \text{otherwise} \end{cases}$$

$$h_{SL}(x) = \frac{B}{\pi^2}\left\{\frac{1 - \cos 2\pi B[(1/4B) + x]}{(1/4B) + x} + \frac{1 - \cos 2\pi B[(1/4B) - x]}{(1/4B) - x}\right\}$$

Common Filters used in FBP (revisited)

In the spatial frequency domain

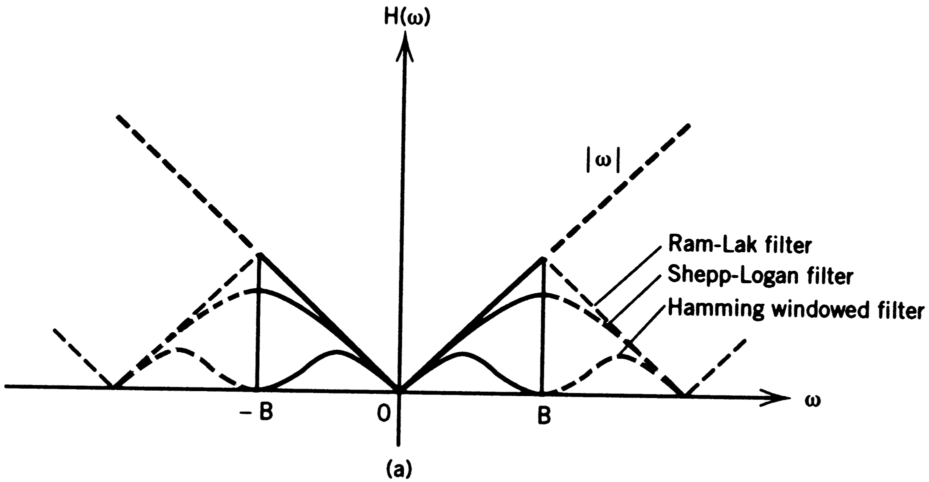


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

$$H(\omega)$$

In the spatial domain

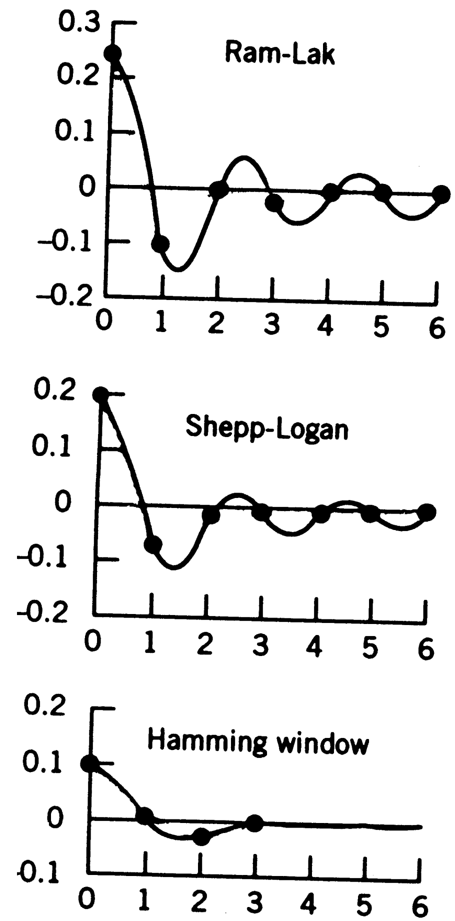


Figure 3-4 (b) Spatial domain filter kernals corresponding to the filter functions shown in the Ram-Lak filter is a high-pass filter with a sharp response but results in some noise enhancement, while the Shepp-Logan and the Hamming window filters are noise-smoothed filters and therefore have better SNR.

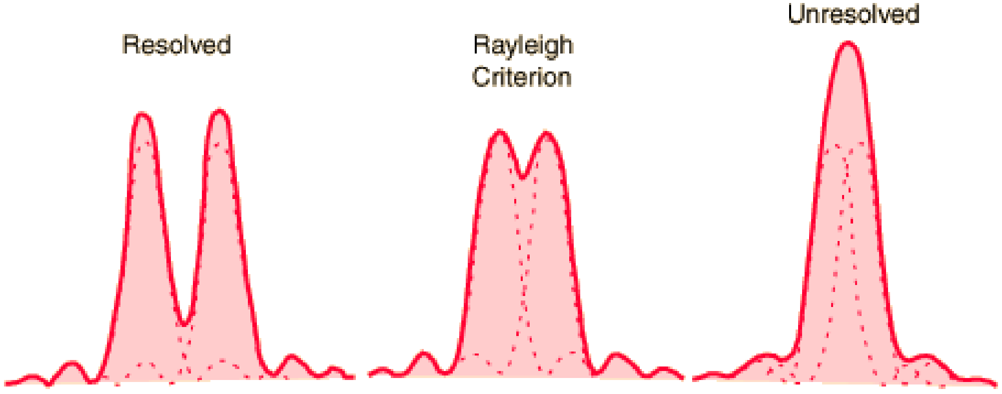
A Revisit to Key Image Quality Measures

Spatial Resolution

- How to quantify spatial resolution?
- How to measure spatial resolution?
- The relationship between spatial resolution and modulation transfer function (MTF)?

Spatial Resolution

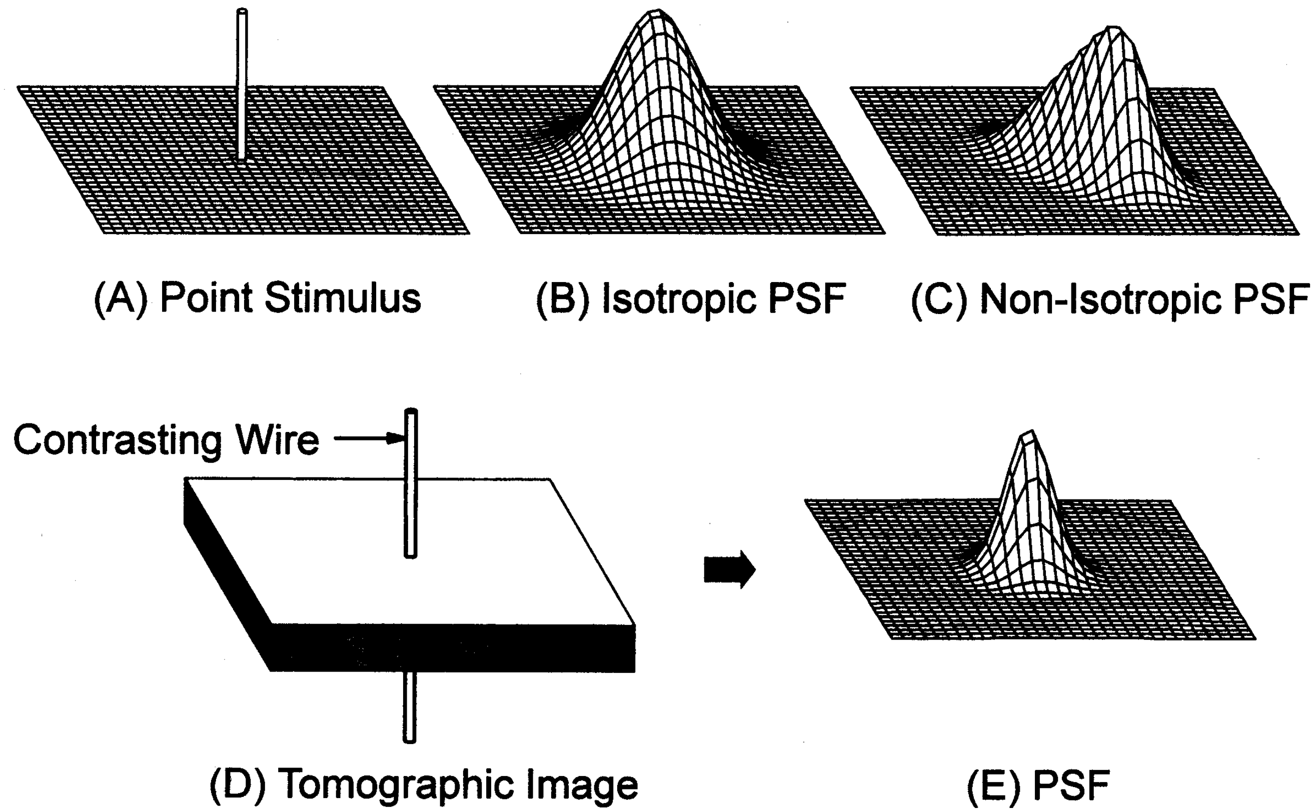
- Resolution: the ability of a given imaging system to accurately depict two distinct events in space, time, or frequency, respectively.
- Resolution can also be considered as the degree of blurring or smearing.



- Spatial resolution is fully described by the point-spread function or impulse response function, h(x,y).

The Point Spread Function (PSF)

One method is to stimulate the imaging system with a point impulse and observe the resulting point-response function.



Full-width at Half Maximum (FWHM) of The Point Spread Function (PSF)

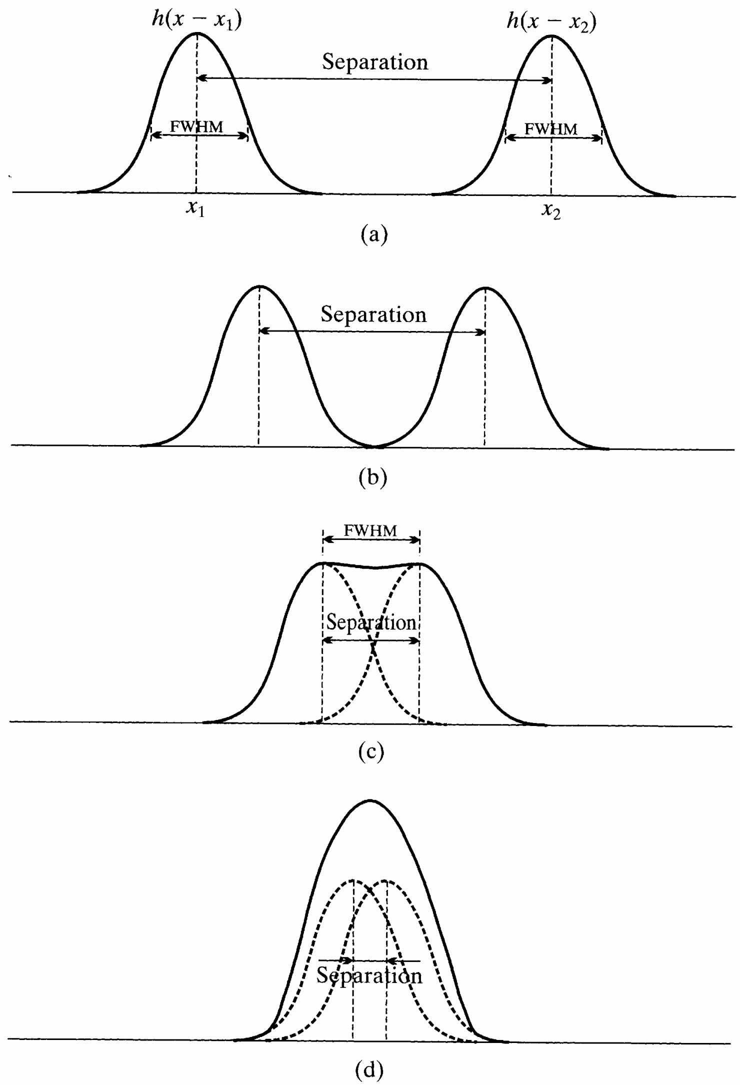
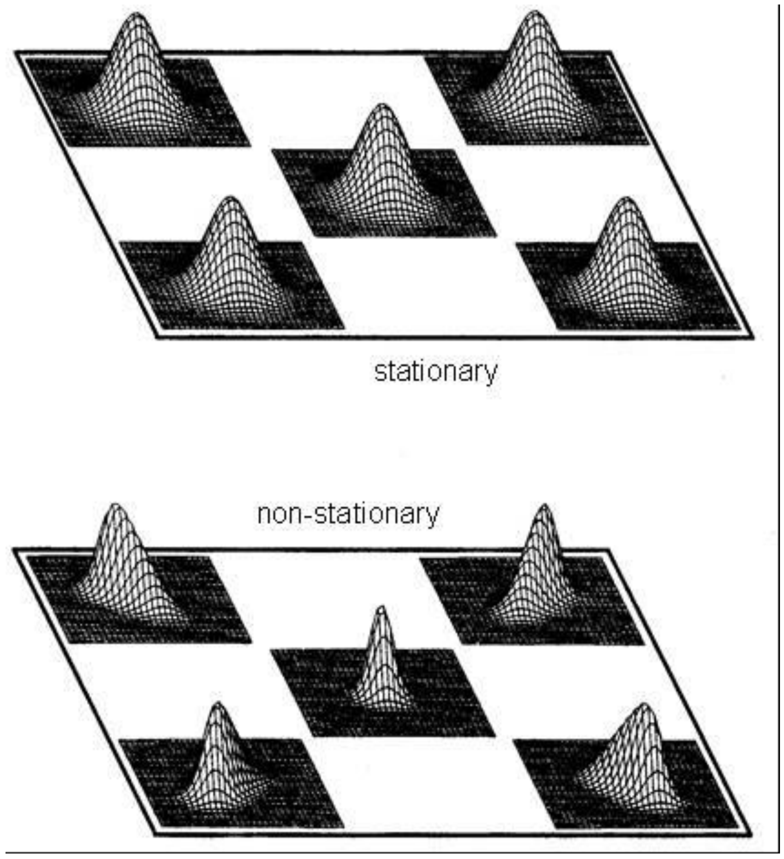


Figure 3.6
An example of the effect of system resolution on the ability to differentiate two points. The FWHM equals the minimum distance that the two points must be separated in order to be distinguishable.

Stationary and Non-stationary PSF

Spatial variation of the PSF is another important aspect of a given imaging system.



Other Ways to Measure the Spatial Resolution
Line Response Function

The resolution of an imaging system can also be estimated with the line-spread function

$$\text{line impulse } f(x,y) = \delta_\ell(x,y) = \delta(x\cos\theta + y\sin\theta - \ell)$$

If we assume the impulse response function of the imaging system *h*(*x*,*y*) is isotropic, it is sufficient to consider the response of the system to a vertical line through the origin,

$$f(x,y) = \delta(x).$$

For this case, line response function (the response of the system to the line impulse) is given by

$$\begin{aligned} g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\xi,\eta) f(x-\xi, y-\eta)\, d\xi\, d\eta\,, \\ &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} h(\xi,\eta)\delta(x-\xi)\, d\xi\right] d\eta\,, \\ &= \int_{-\infty}^{\infty} h(x,\eta)\, d\eta\,, \end{aligned}$$

Line Response Function (or Line-Spread Function, LSF)

$$Line\ Response\ Function:\ l(x) \equiv \int_{-\infty}^{\infty} h(x,\eta)d\eta$$

The 1-D Fourier transform of the line spread function is

$$\begin{aligned} L(u) &= \mathcal{F}_{1D}[l](u)\,, \\ &= \int_{-\infty}^{\infty} l(x)e^{-j2\pi ux}\,dx\,, \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,\eta)e^{-j2\pi ux}\,dx\,d\eta\,, \\ &= H(u,0). \end{aligned}$$

Remember that

$$MTF(u) = \frac{m_g}{m_f} = |H(u,0)|$$

So the values of the Fourier transform of the LSF crossing a horizontal line passing through the origin is numerically equivalent to the MTF,

$$L(u) = F_1[l(x)] = |H(u,0)| = \frac{m_g}{m_f} = MTF(u)$$

LSF and MTF

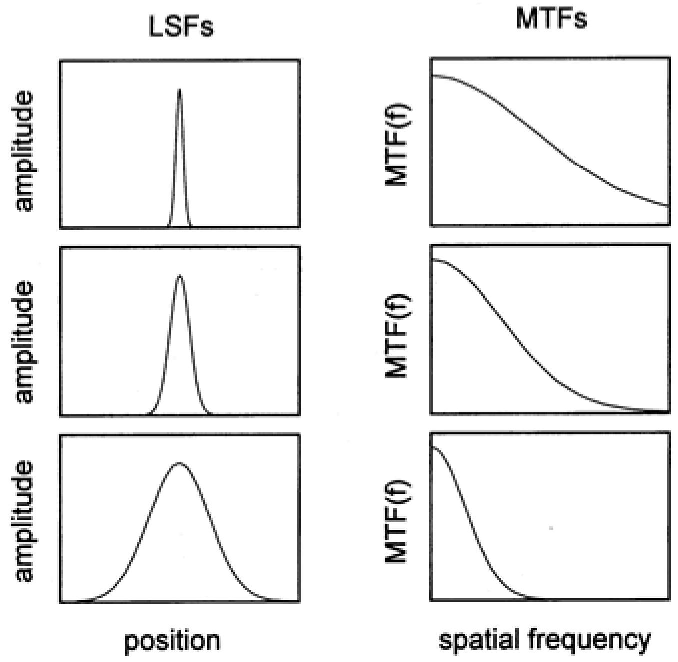
- Modulation Transfer Function (MTF).

$$\mathrm{MTF}(u) = \frac{m_g}{m_f} = \frac{|H(u,0)|}{H(0,0)} = \frac{|L(u)|}{L(0)}, \qquad \text{for every } u.$$

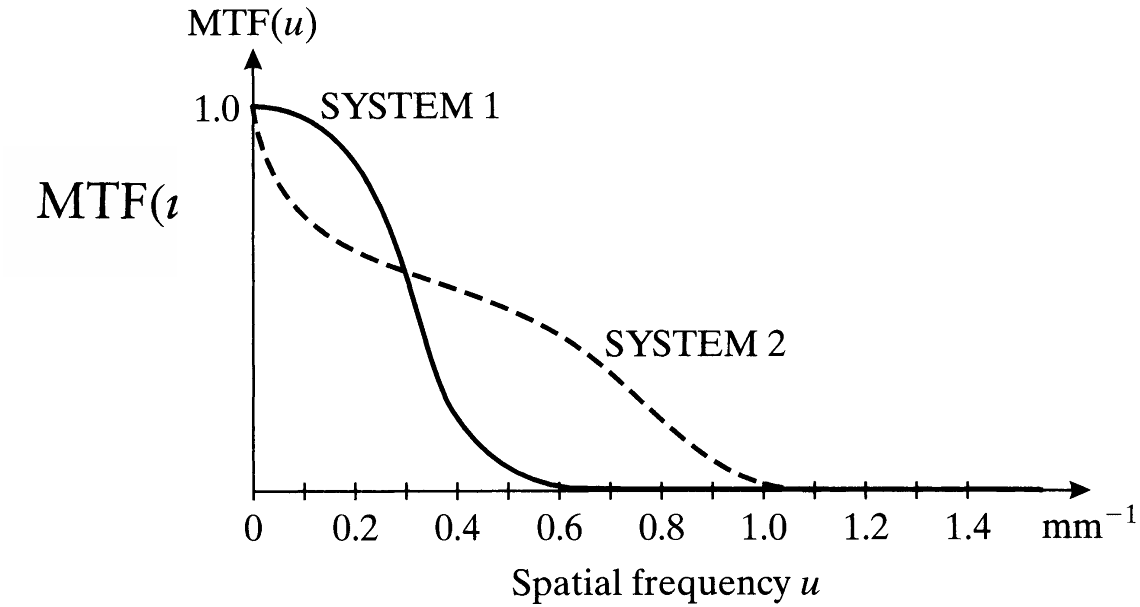
- For a "reasonable" imaging system, the L(0)=1, so that

$$\mathrm{MTF}(u) = L(u)$$

- MTF is an effective way to compare two imaging systems in terms of spatial resolution and contrast.



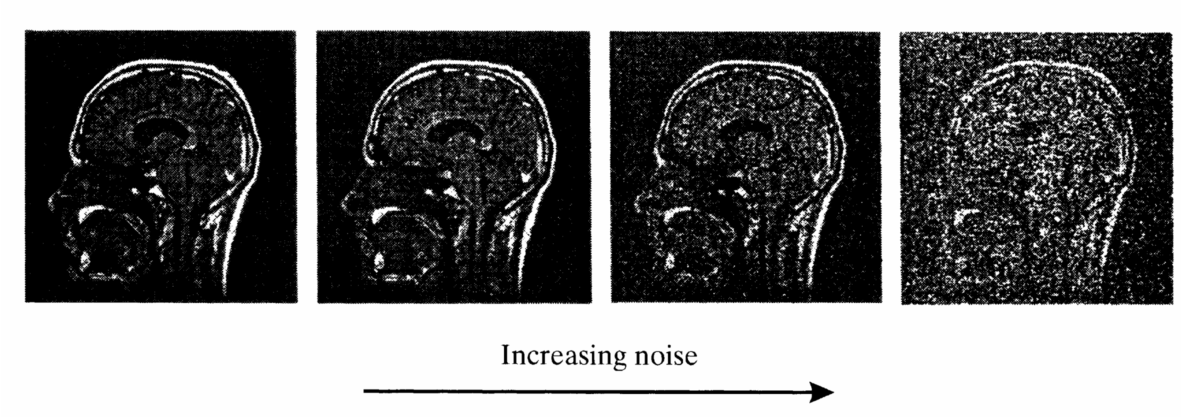
LSF and MTF

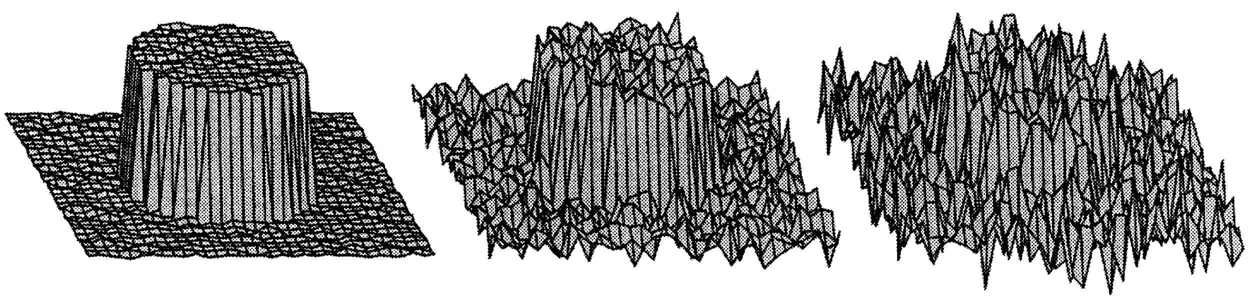


- System 1 has better low-frequency contrast, and it is better for imaging coarse features.
- System 2 has a better high-frequency contrast, and it is better for resolving fine details.
- The resolution of a system is related to the higher frequency response and the cut-off frequency of the MFT.

Image Noise

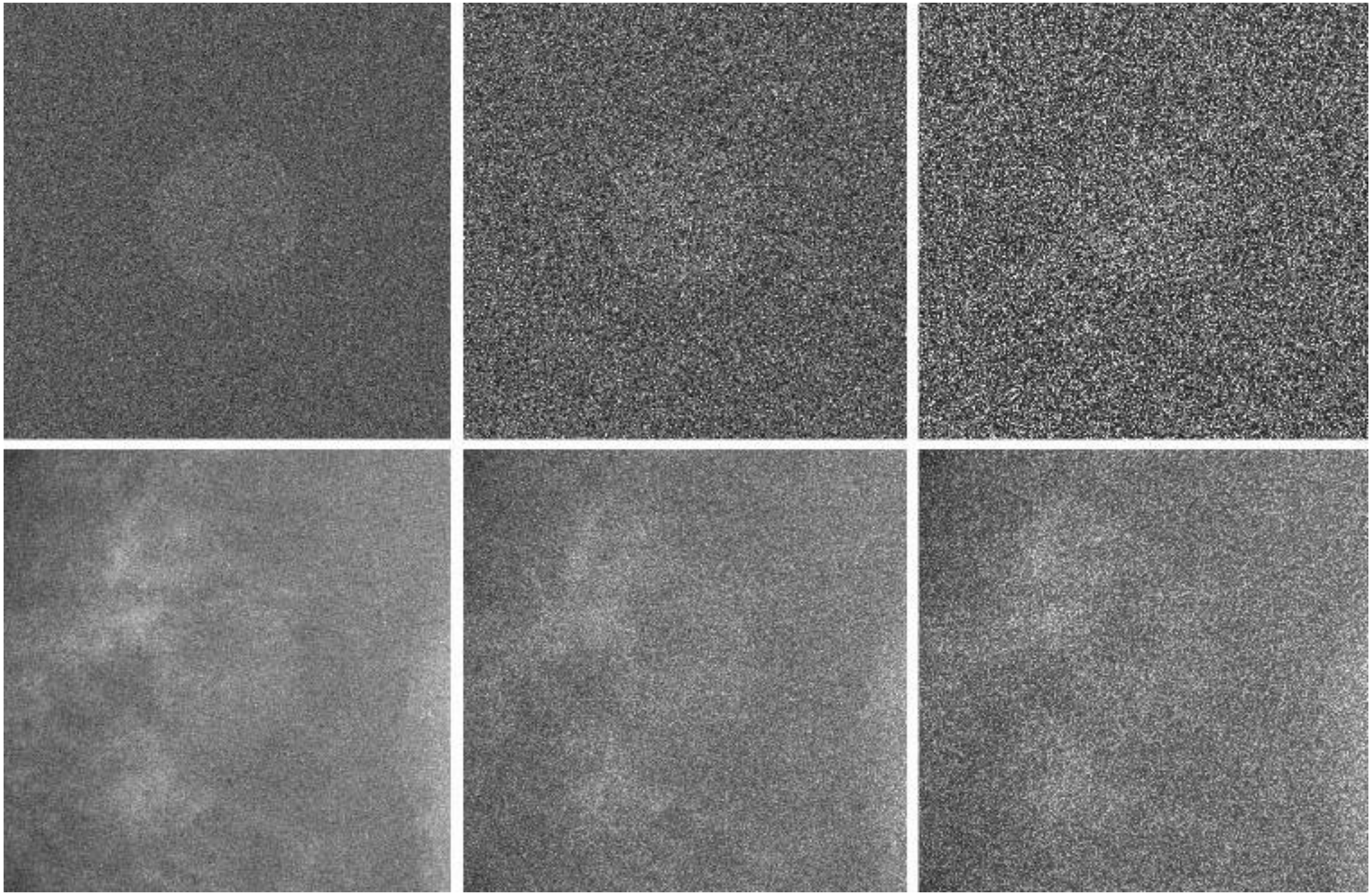
- The random fluctuation is referred to as the image noise. That has dramatic effects on the subsequent analysis, for example, for signal detection and quantification tasks …





Low Noise Medium Noise High Noise

Image Noise Reduces Contrast



Where is the noise coming from?

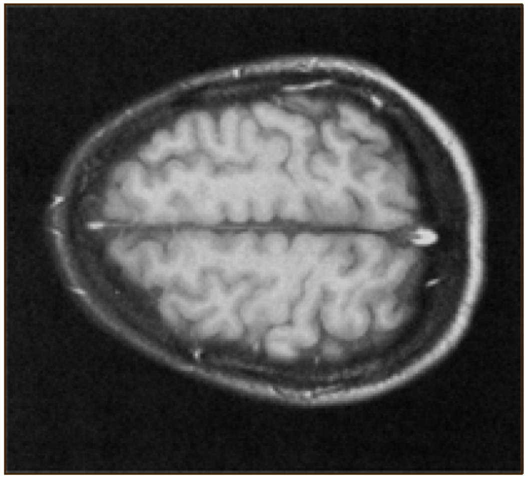
Image Noise

- Due to the random nature of the data acquired by any imaging system, the output images are normally multivariate random variables.

Input signal ⇨ Data acquisition hardware ⇨ $\boxed{\text{Noisy Data}}$ ⇨ Data processing ⇨ Final image

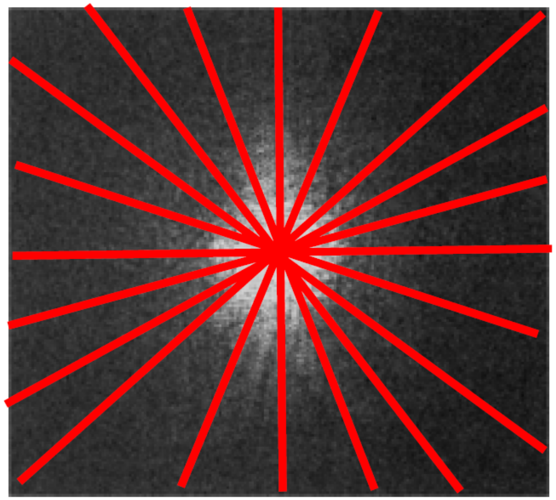
noise propagation

Central Slice Theorem



DFT image represents integration of original projections DFT transformed and summed together.

1-D DFTs at each projection



This is the fast way to create the DFT image from projection data. The more projections taken, the more complete the sampling.

http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Simple Back-projection and the 1/r Blurring

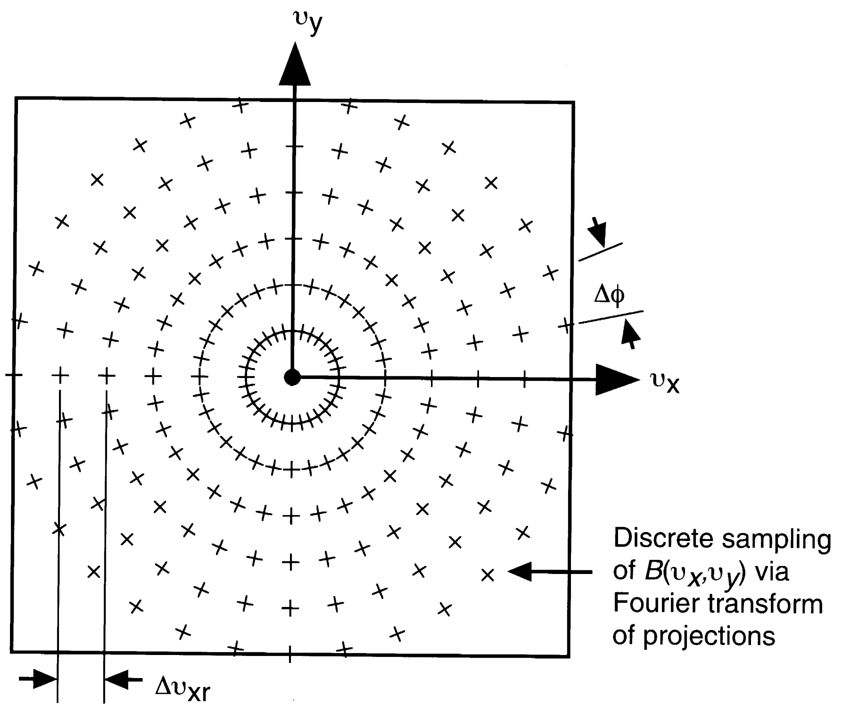


FIGURE 18 The discrete sampling pattern of $F(\upsilon_x, \upsilon_y)$ contained in $B(\upsilon_x, \upsilon_y)$, resulting from the use of discretely sampled projections.

The nature of the 1/r blurring:
Radon transform produced equally spaced radial sampling in Fourier domain.

Simple Backprojection and Inverse Radon Transform

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos \phi + y \sin \phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{back\text{-}projection}(x,y) = \int b_\phi(x,y) \, d\phi$$

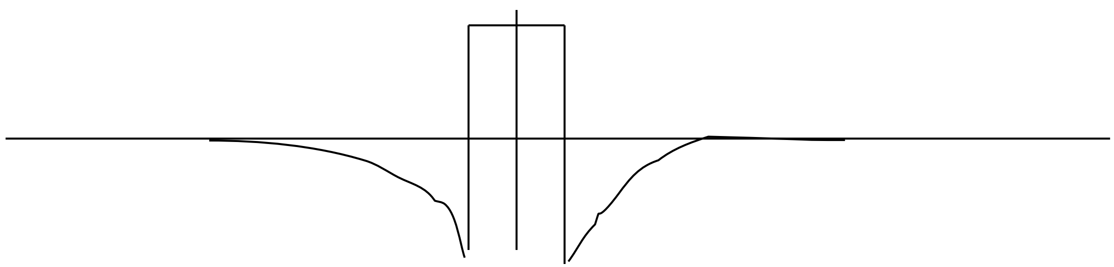
$$\hat{f}_{\text{simple backprojection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x') dx'$$

$$\hat{f}_{\text{Inverse Radon transform}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot |w|\} \cdot \delta(x\cos\phi + y\sin\phi - x') dx'$$

An Example – Noise on Images Acquired with FBP

The Ram-Lak filter in spatial domain

$$
\begin{aligned}
h_{\mathrm{RL}}(x) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} H_{\mathrm{RL}}(\omega)\exp(ix\omega)\, d\omega \\
&= \frac{1}{2\pi}\int_{-2\pi B}^{2\pi B} |\omega| \exp(ix\omega)\, d\omega \\
&= 2B^2 \operatorname{sinc}(2\pi Bx) - B^2 \operatorname{sinc}^2(\pi Bx)
\end{aligned}
$$



$$
\hat{f}(x, y) = \frac{1}{\pi}\int_0^{\pi} d\phi \int_{-\infty}^{\infty} dx'\, p_\phi(x')h(x\cos\phi + y\sin\phi - x')
$$

Or written in discrete form: $\hat{f}_i = \sum_{j=1}^{m} S_{ij} \cdot g_j$

Filtered Back-projection

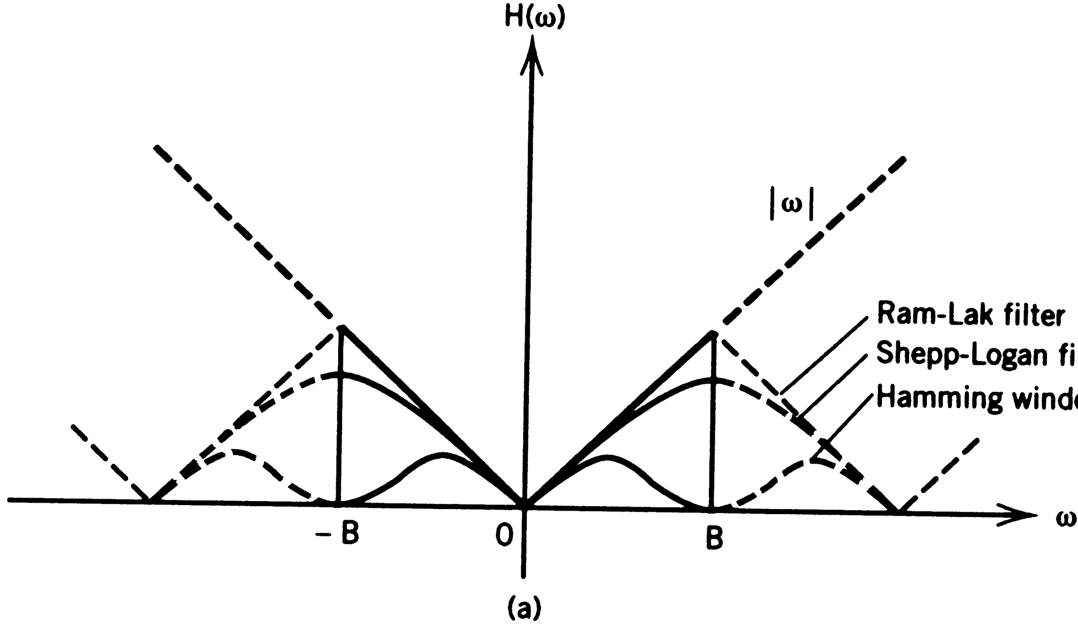


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

Covariance of Random Variables

Covariance provides a measure of the strength of the correlation between two or more random variables. The covariance for two random variables and, each with sample size, is defined by the expectation value

$$Variance(N_1) \equiv \sigma^2 = \int_{N_1} p(N_1) \cdot \left(N, -\bar{N}_1\right)^2 \cdot dN_1$$

$$Covariance(N_1, N_2) = \iint_{N_1 N_2} p(N_1) \cdot p(N_2) \cdot \left(N, -\bar{N}_1\right) \cdot \left(N_2 - \bar{N}_2\right) \cdot dN_2 dN_2$$

Why Gaussian Random Variable is Important?

- When a quantity is derived as the result of a large number of accumulative effects, and each effect has a small contribution to the final outcome, then the distribution of the quantity tends to follow Gaussian distribution.

- The measured value on a detector is the result of accumulated interactions during a measurements session…
- The reconstructed image at a given pixel is the sum of the contributions from the data acquired with a large number of detector elements in the system…

$$\hat{f}_i = \sum_{j=1}^{m} s_{ij} \cdot g_j \qquad \text{or} \qquad \hat{f} = S \cdot g$$

In this simple case, we know exactly what is the probability of a given reconstructed image!!

- It should follow Gaussian distribution

Mean of the reconstructed image: $\bar{\hat{f}} = S \cdot \bar{g}$

Covariance: $Cov\left(\hat{f}\right) = S \cdot Cov(g) \cdot S^T$

The Maximum Likelihood Reconstruction

Recall that the *likelihood function, L*(*f*,*g*), of a possible source function *f* is

$$L(\boldsymbol{f},\boldsymbol{g}) = p(\boldsymbol{g} \mid \boldsymbol{f})$$

So that the maximum likelihood solution (the image that maximizing the likelihood function) can be found as

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, L(\boldsymbol{f},\boldsymbol{g})$$

or equivalently

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \arg\max_{\boldsymbol{f}} \log\left[L(\boldsymbol{f},\boldsymbol{g})\right] \equiv \underset{\boldsymbol{f}}{\operatorname{argmax}}\, l(\boldsymbol{f},\boldsymbol{g})$$

where $l(\boldsymbol{f},\boldsymbol{g})$ is the log - likelihood function

$$l(\boldsymbol{f},\boldsymbol{g}) = \log\left[L(\boldsymbol{f},\boldsymbol{g})\right]$$

Poisson Statistics of the Projection Data

The probability of a given projection data g=(g_1, g_2, g_3,…, g_M) is

$$p(\mathbf{g}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{\bar{g}_m^{\boxed{g_m}}}{g_m!} e^{-\bar{g}_m}$$

Measured no. of counts on detector pixel m.

where $\bar{g}_m$ is the expected value for the number of counts on detector pixel # m

$$\bar{g}_m = \sum_{n=1}^{N} f_n \boxed{p_{nm}}$$

In the context of emission tomography, p_{nm} is the probability of a gamma ray generated at a source pixel n is detected by detector element m.

Remember that

$$\begin{pmatrix} \bar{g}_1 \\ \bar{g}_2 \\ \bar{g}_3 \\ \vdots \\ \bar{g}_M \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1N} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2N} \\ p_{31} & p_{32} & p_{33} & \cdots & p_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{M1} & p_{M2} & p_{M3} & \cdots & p_{MN} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{pmatrix}$$

The Maximum Likelihood Reconstruction

For data that follows Poisson distribution, the *likelihood function, L*(*f*,*g*), of a possible source function *f* is

$$L(\boldsymbol{f},\boldsymbol{g}) = p(\boldsymbol{g}\,|\,\boldsymbol{f}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{\bar{g}_m^{\;g_m}}{g_m!} e^{-\bar{g}_m}$$

$\bar{g}_m = \sum_{n=1}^{N} f_n p_{nm}$, is the expected number of counts on detector pixel m.

g_m is the measured number of counts on detector pixel m.

The log-likelihood function is

$$l(\boldsymbol{f},\boldsymbol{g}) = \log\left[L(\boldsymbol{f},\boldsymbol{g})\right] = \log\left[p(\boldsymbol{g}\,|\,\boldsymbol{f})\right]$$

$$= \prod_{m=1}^{M} p(g_m) = \log\left[\prod_{m=1}^{M} \frac{\bar{g}_m^{\;g_m}}{g_m!} e^{-\bar{g}_m}\right] = \sum_{m=1}^{M}\left(g_m \log \bar{g}_m - \bar{g}_m - \log g_m!\right)$$

The Maximum Likelihood Reconstruction

So the ML reconstruction for Poisson distributed data is

$$\hat{\boldsymbol{f}}_{ML} = \arg\max_{\boldsymbol{f}} l(\boldsymbol{f},\boldsymbol{g}) = \arg\max_{\boldsymbol{f}} \left[\sum_{m=1}^{M} \left(g_m \log \bar{g}_m - \bar{g}_m - \log g_m! \right) \right]$$

since g_m is not function of $\boldsymbol{f}$, we get

$$\boxed{\hat{\boldsymbol{f}}_{ML} = \arg\max_{\boldsymbol{f}} l(\boldsymbol{f},\boldsymbol{g}) = \arg\max_{\boldsymbol{f}} \left[\sum_{m=1}^{M} \left(g_m \log \bar{g}_m - \bar{g}_m \right) \right]}$$

The Maximum Likelihood Expectation Maximization (MLEM) Algorithm

The source (image) function *f* that has the maximum likelihood of giving rise to the observed data *g* can be found by the following iterative updating scheme

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{n'm} f_{n'}^{(old)}} p_{nm}, \quad n = 1,2,\ldots,N$$

where

$m = 1,2,\ldots,M$ is the index of detector elements in the imaging system

$n = 1,2,\ldots,N$ is the index of source object pixels

$f_n^{(old)}$: the current estimate of the source function in pixel n

$f_n^{(new)}$: the updated estimate of the source pixel n

p_{nm} : the probability of a gamma ray generated in source pixel n and detected by the detector element m

g_m : the observed number of counts in detector pixel m

Chapter 2: Mathematical Preliminaries

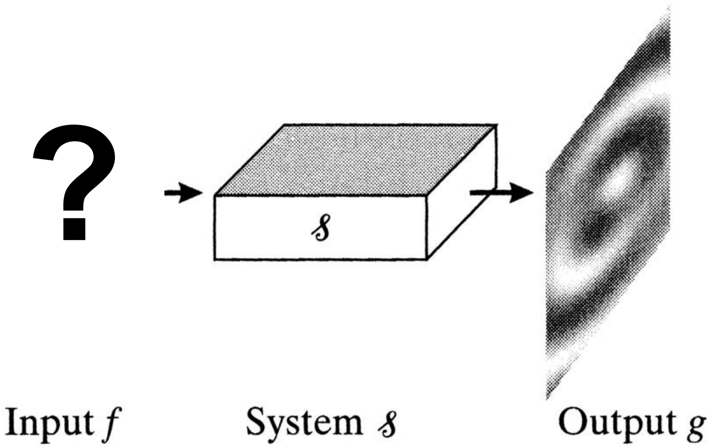
Mathematical Preliminaries for 2-D Image Reconstructions

Contents

- Projection and Radon transform
- Sinogram
- Central slice theorem
- Image reconstruction with simple backprojection
- Filtered-backprojection
- Other analytical reconstruction methods.
- Matlab examples

The Inverse Problem

General Inverse Problem and Image Reconstruction



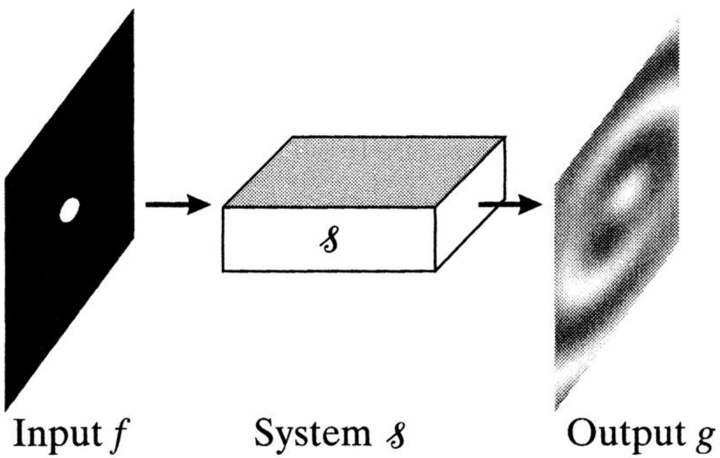
What should be the input signal that gave rise to the output data?

Convolution Theorem Revisited

The convolution theorem enables one to perform convolution operation as multiplication process in spatial frequency domain.

$$f(x,y) * g(x,y) = \mathscr{F}^{-1}\{\mathscr{F}[f(x,y)] \cdot \mathscr{F}[g(x,y)]\}$$

By using the *Fast Fourier Transform* (FFT) algorithms, the convolution operation can be performed very efficiently !! This provide a practical way for modeling linear shift-invariant systems …



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

Understanding the Nature of the Data

Radon Transform and Projection

Projection Data from Early X-ray CT Systems

- Uses a collimator to keep exposure to a slice
- Builds image from multiple projections
- We will assume parallel rays for now.
    - Actually, how first-generation scanners worked.
    - Translate source and single detector across the body for one angle
    - Then rotate the source and detector to get the next angle

The principles of sectional imaging 103

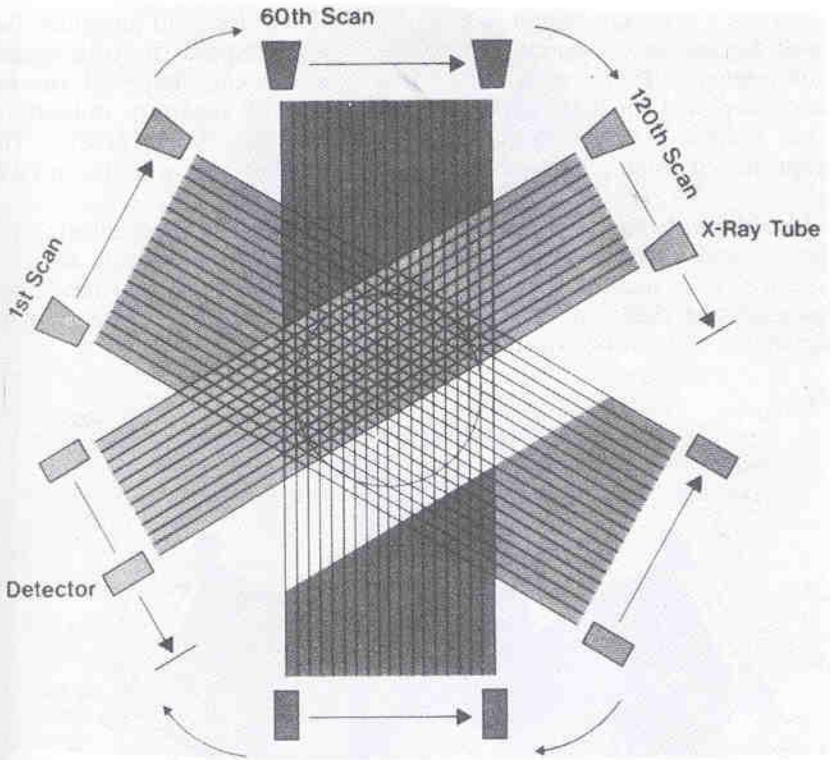
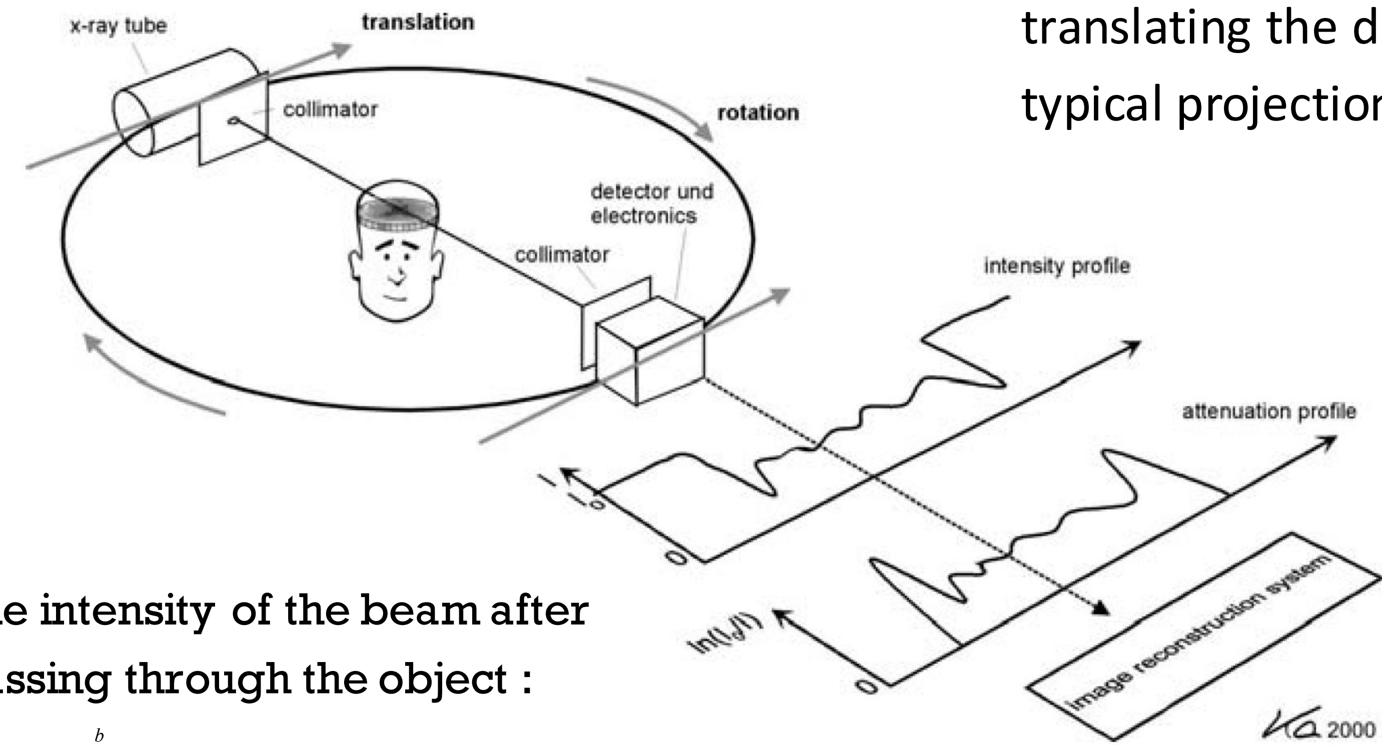


Figure 4.5 Schematic representation of a first-generation CT scanner. It utilises a single pencil beam and single detector for each scan slice. The x-ray source and detector are passed transversely across the object being scanned, with incremental rotations of the system at the end of each transverse motion. (Reproduced from Maravilla and Pastel (1978).)

Projection Data from Early X-ray CT Systems



Data measured by translating the detector is a typical projection data.

The intensity of the beam after passing through the object :

$$I = I_0 e^{-\int_a^b \mu(t)\cdot dt} \Leftrightarrow \ln(I_0 / I) = \int_a^b \mu(t)\cdot dt$$

From Computed Tomography, Kalender, 2000.

A Typical Gamma Camera

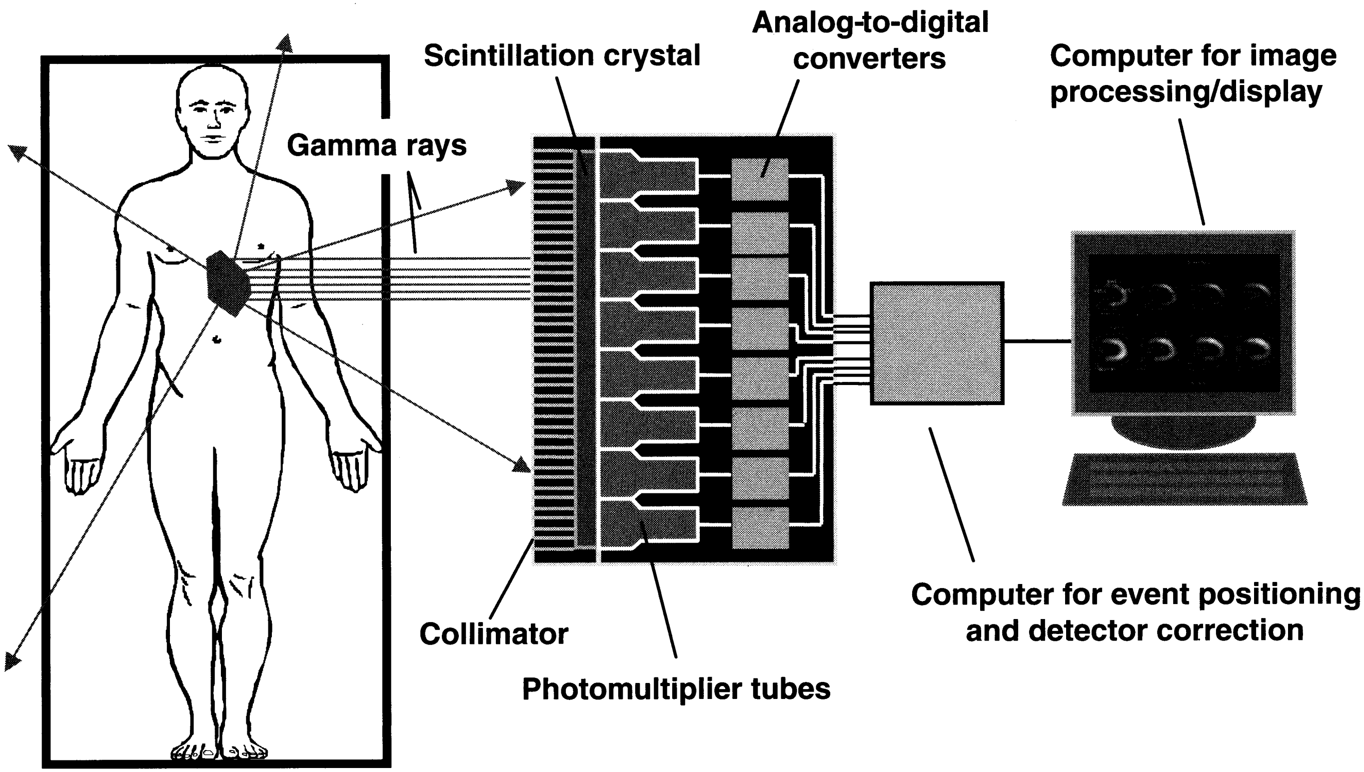
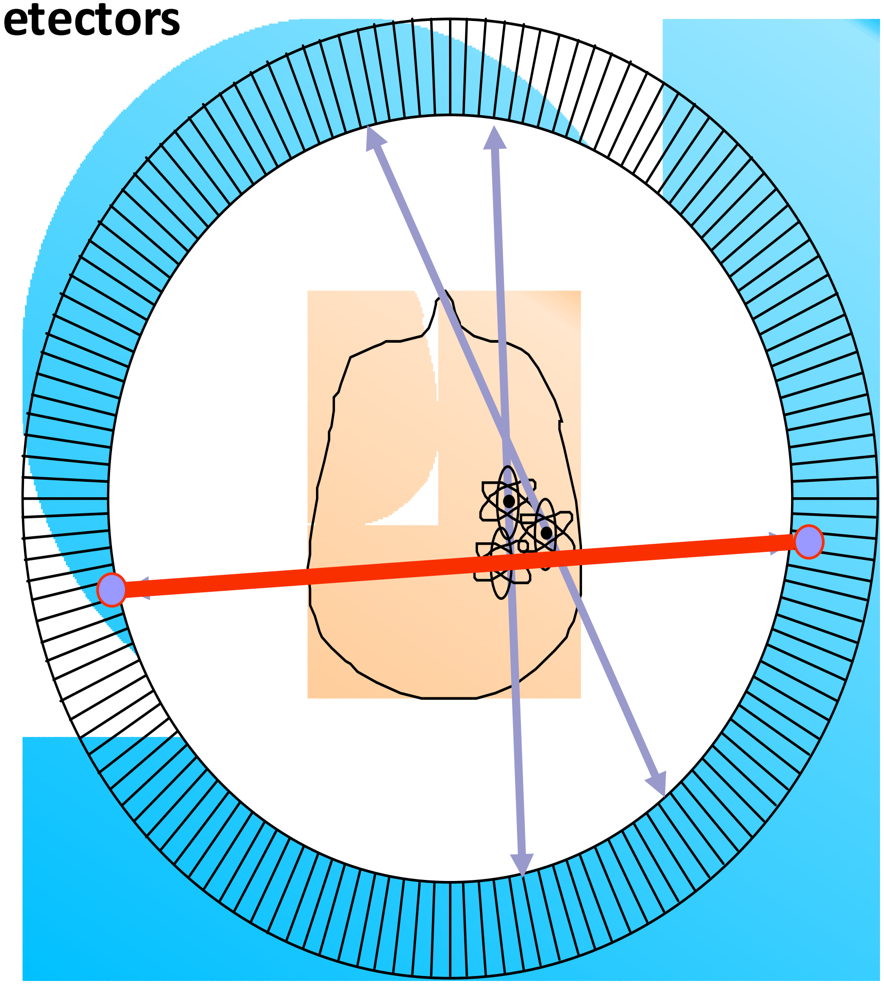


FIGURE 7 Schematic diagram of a conventional gamma camera used in SPECT. The collimator forms an image of the patient on the scintillation crystal, which converts gamma rays into light. The light is detected by photomultiplier tubes, the outputs of which are digitized and used to compute the spatial coordinates of each gamma-ray event (with respect to the camera face). A computer is used to process, store, and display the images.

Detect Radioactive Decays

Ring of Photon Detectors



- Radionuclide decays, emitting β^{+}.
- β^{+} annihilates with e^{-} from tissue, forming back-to-back 511 keV photon pair.
- 511 keV photon pairs detected via time coincidence.
- Positron lies on a line defined by a detector pair (known as a chord, a line of response, or a *LOR*).

Detect Pair of Back-to Back 511 keV Photons

Example -- MRI of the Brain

(a) (b) (c)

Figure 12.10 Three images of the same slice through the skull. Contrast between the tissue types are classified as (a) P_D-weighted, (b) T_2-weighted, and (c) T_1-weighted. (Used with permission of GE Healthcare.)

Proton Density
T_E = 17 ms, T_R = 6000 ms

T_2 Contrast
T_E = 102 ms, T_R = 6000 ms

- **Long T_R → minimizing T_1 effect**
- **GM, WM decayed, CSF partially decayed → large contrast between GM/WM and CSF**
- **Small contrast between GM and WM (WM is darker due to shorter T_2)**

T_1 Contrast
T_E = 17 ms, T_R = 600 ms

- **Short T_R → building up T_1 effect**
- **T_R falls between T_1's for GM, WM but much shorter than that for CSF → GM/WM should built up more (brighter in the image) that CSF.**

TABLE 12.2

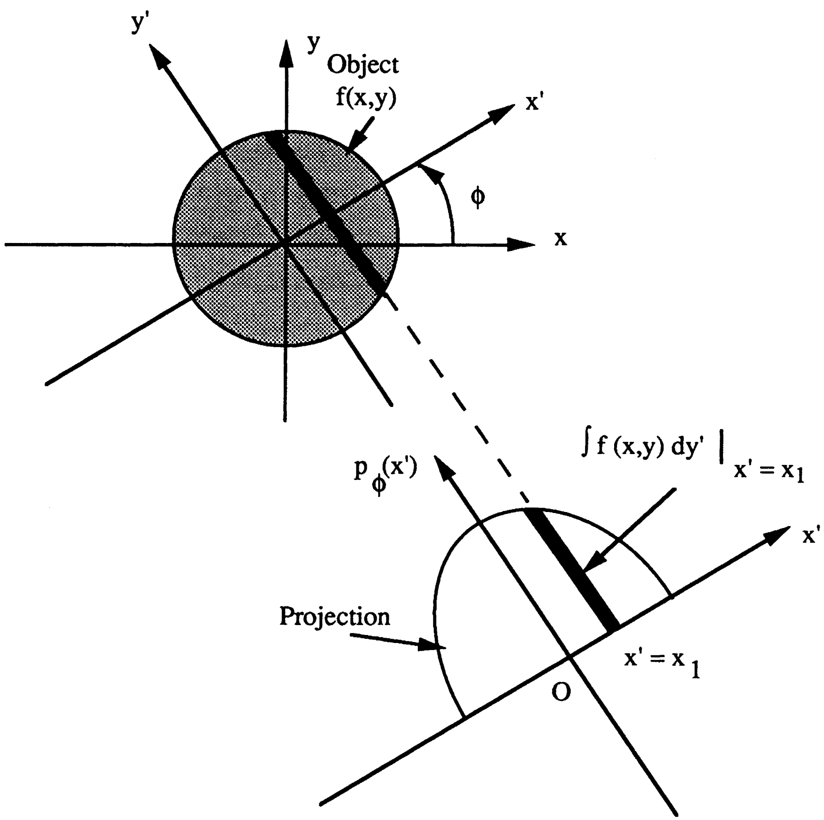
Typical Brain Tissue Parameters Measured at 1.5 T

Tissue Type	Relative P_D	T_2 (ms)	T_1 (ms)
White matter	0.61	67	510
Gray matter	0.69	77	760
Cerebrospinal fluid	1.00	280	2650

Source: Adapted from Liang and Lauterbur, 2000.

How do we mathematically describe projections?

The Line Integral Projection

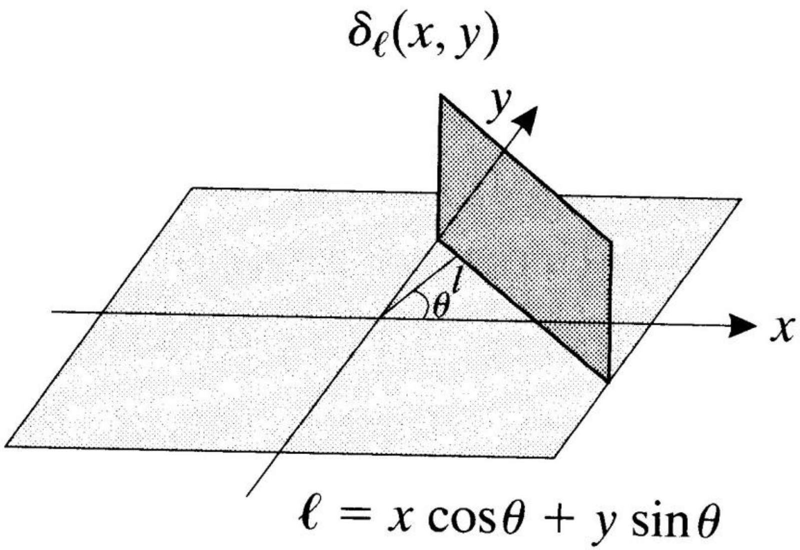


- Measures the integral property a physical object along a straight line.
- Projection data is naturally acquired by successive sampling the object at given angular intervals.

Line Impulse Signal (revisited)

$$\delta_L(x, y) = \delta(x\cos\theta + y\sin\theta - l)$$

$$where\ \delta(x) = \begin{cases} > 0, & x\cos\theta + y\sin\theta = l \\ 0, & otherwise \end{cases}$$



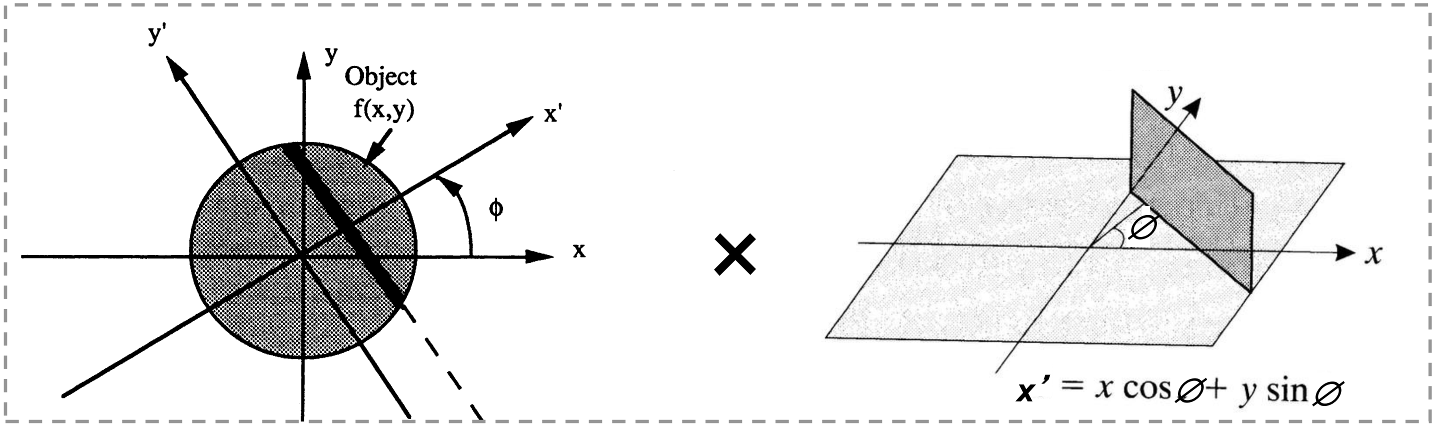
Line Impulse Signal (revisited)

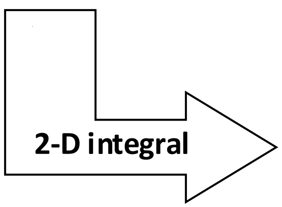
$$\delta_L(x, y) = \delta(x\cos\theta + y\sin\theta - l)$$

- It can be used to measure the spatial resolution of a given imaging system.

- It is used to calculate the line-integral projection data for a given 2-D object.

Radon Transform





The value of the projection function $p_\phi(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The **Radon transform**, which produces the projection data, is defined as

$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Sinogram

Radon Transform and Sinogram

Radon transform maps a 2-D function f(x,y) into a sinogram.

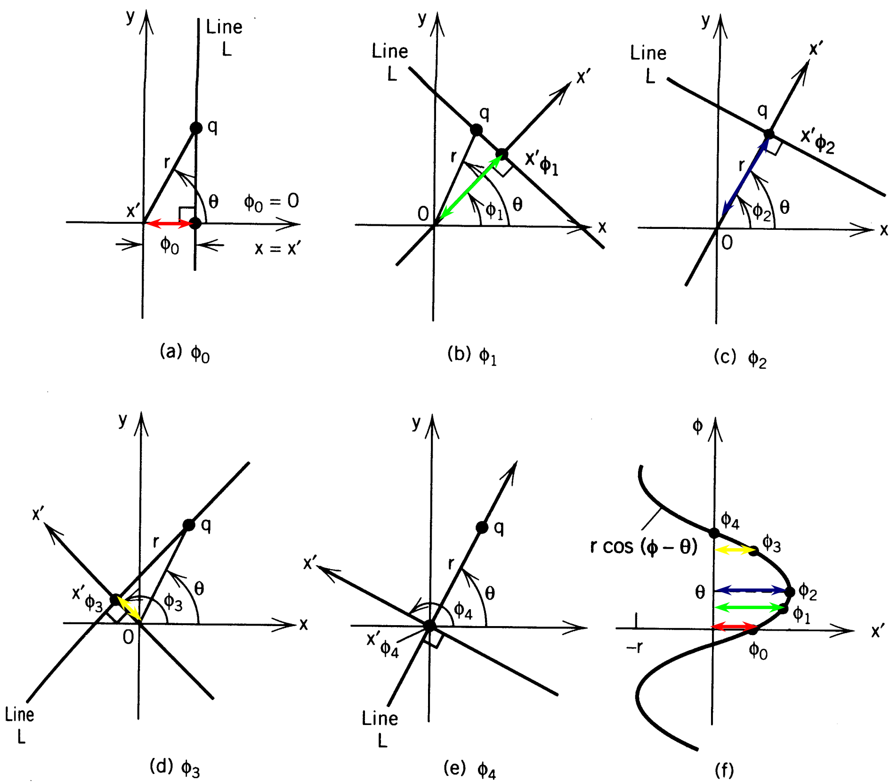


Figure 3-2 Sinogram of a point q at a various angles ϕ_i for a given θ: (a) $\phi_0 = 0$, (b) $\phi_1 = 45°$, (c) $\phi_2 = 60°$, (d) $\phi_3 = 135°$, and (e) $\phi_4 = 150°$. At $i = 0$ or $\phi_i = 0$, the rotated coordinates x' coincide with the reference x-axis and give $x' = r\cos(\phi - \theta) = r\cos\theta$.

$$\mathscr{P}(\phi, x') = \mathscr{R}[f(x, y)]$$

Any point represented by polar coordinates (r, θ) will simply follow the equation $x' = r\cos(\phi - \theta)$ in the sinogram space.

Radon Transform and Sinogram

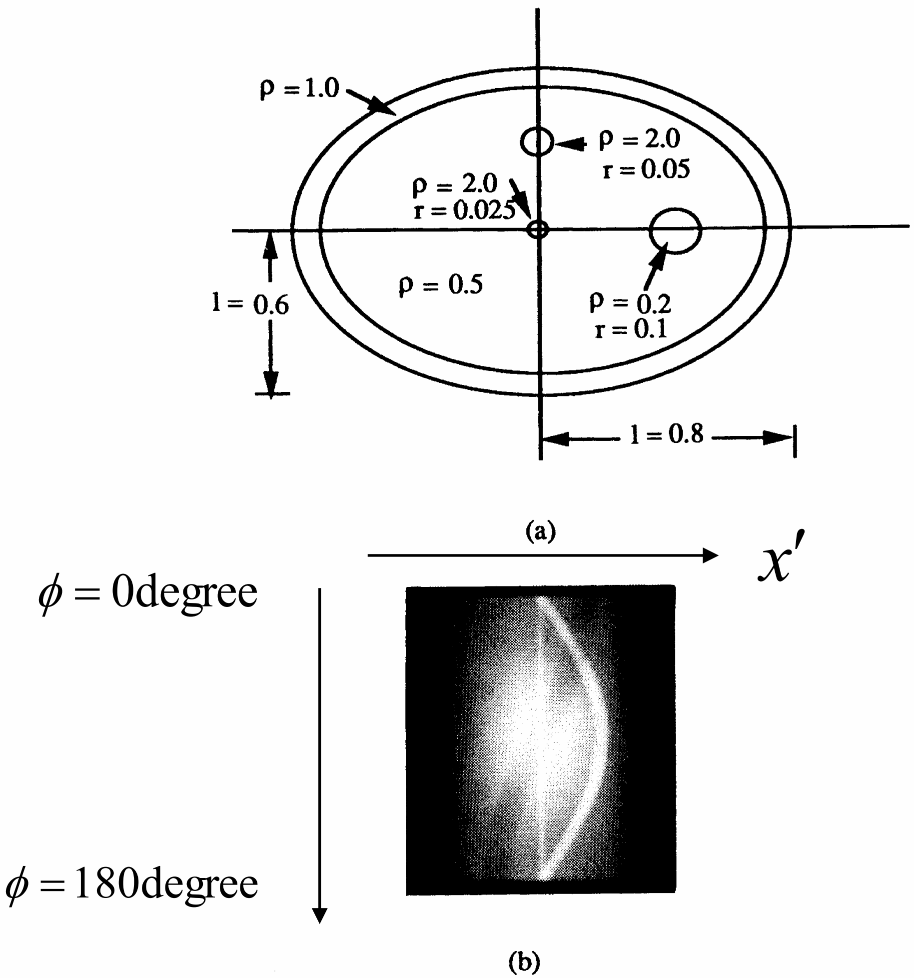
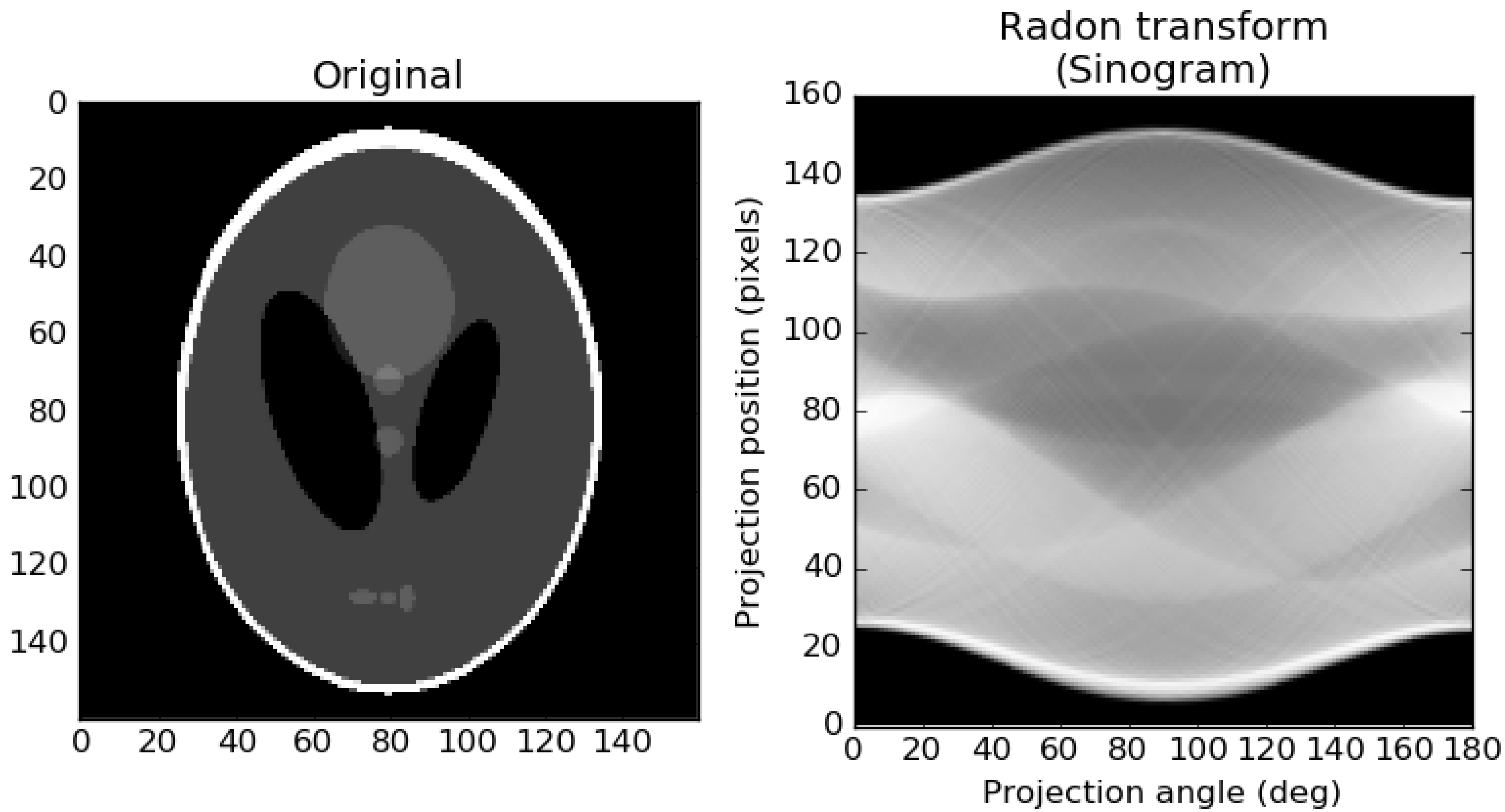


Figure 3-3 (a) Phantom with several distinct objects at various positions; (b) correspondi sinogram.

Sinogram is a 2-D function representing the original function f(x,y) into the projection data space.

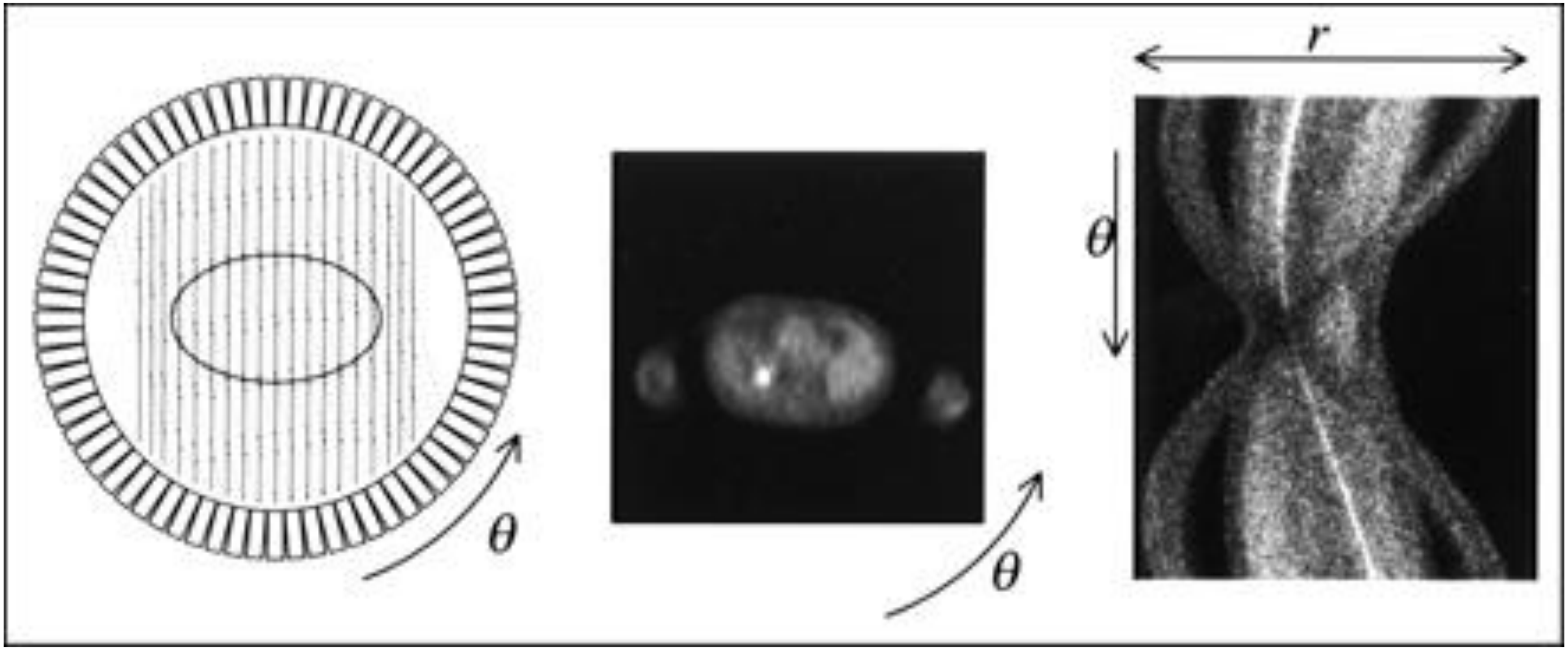
Sinogram is the basic data format for reconstruction

Radon Transform and Sinogram



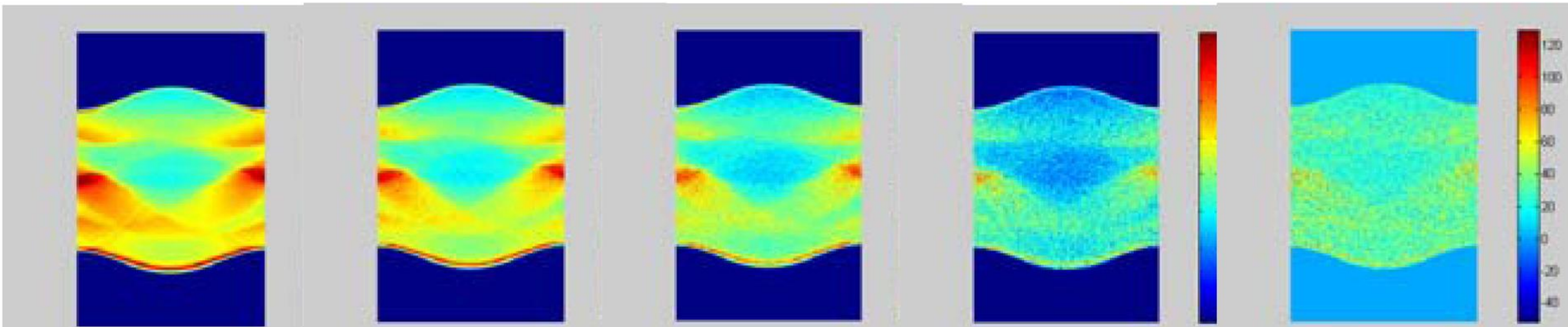
http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

Radon Transform and Sinogram



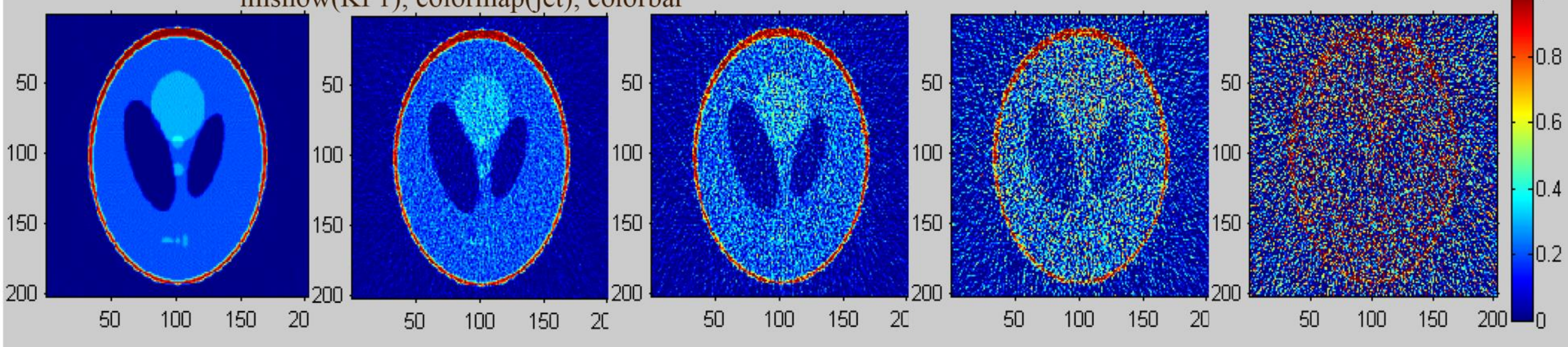
http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

Matlab Examples



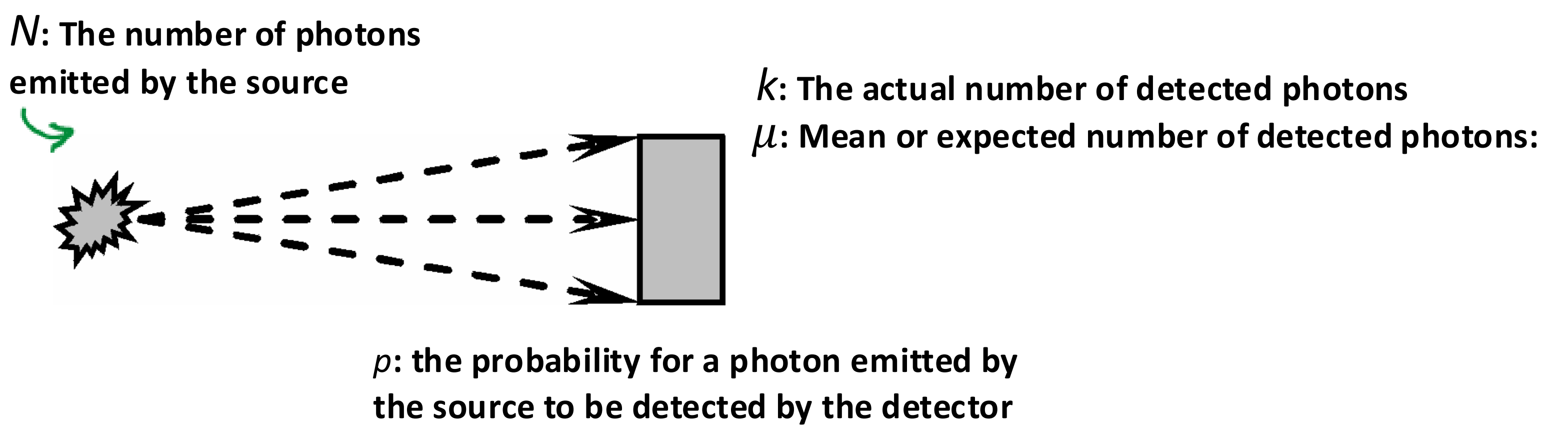
imshow(theta,xp,RN1,[],'notruesize'), colormap(jet), colorbar;

KP1 = iradon(RN1,theta);
imshow(KP1), colormap(jet), colorbar



sd=0 sd=0.05 sd=0.10 sd=0.20 sd=0.50

The Basic Idea of Statistical Reconstruction



We can estimate the underlying parameter *N* by the Maximum-Likelihood Estimation (MLE) strategy:

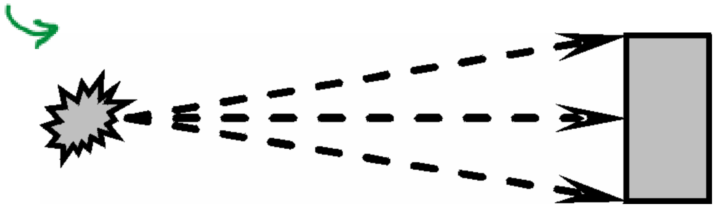
$$\hat{N}_{Maximum\ Likelihood\ estimation} = \arg\max_{N}\{P[k|N]\}$$

$P[k(N)]$ is call the Likelihood function: the probability of detecting *k* counts given there were a total of *N* gamma rays emitted by the source.

N: The number of photons emitted by the source

k: The actual number of detected photons

μ: Mean or expected number of detected photons:



p: the probability for a photon emitted by the source to be detected by the detector

Since

$$P[k(N)]=\frac{(N\cdot p)^k}{k!}e^{-N\cdot p}$$

Then

$$\hat{N}_{Maximum\ likelihood} = \arg\max_N\{P[k|N]\} = \arg\max_N\left\{\frac{(N\cdot p)^k}{k!}e^{-N\cdot p}\right\},$$

alternatively

$$\hat{N}_{ML} = \arg\max_N\{\log[P[k|N]]\}$$

$$= \arg\max_N\{[k\cdot log(N\cdot p) - N\cdot p - log\, k!]\}$$

$$= \arg\max_N\{[k\cdot log(N\cdot p) - N\cdot p]\}$$

So the ML estimate of the number of gamma rays emitted by the source is

$$\hat{N}_{ML} = \arg\max_N\{\log[P[k|N]]\} = \arg\max_N\{[k\cdot \log(N\cdot p) - N\cdot p]\}$$

Image Reconstruction and General Linear Inverse Problem

An Emission Tomography System

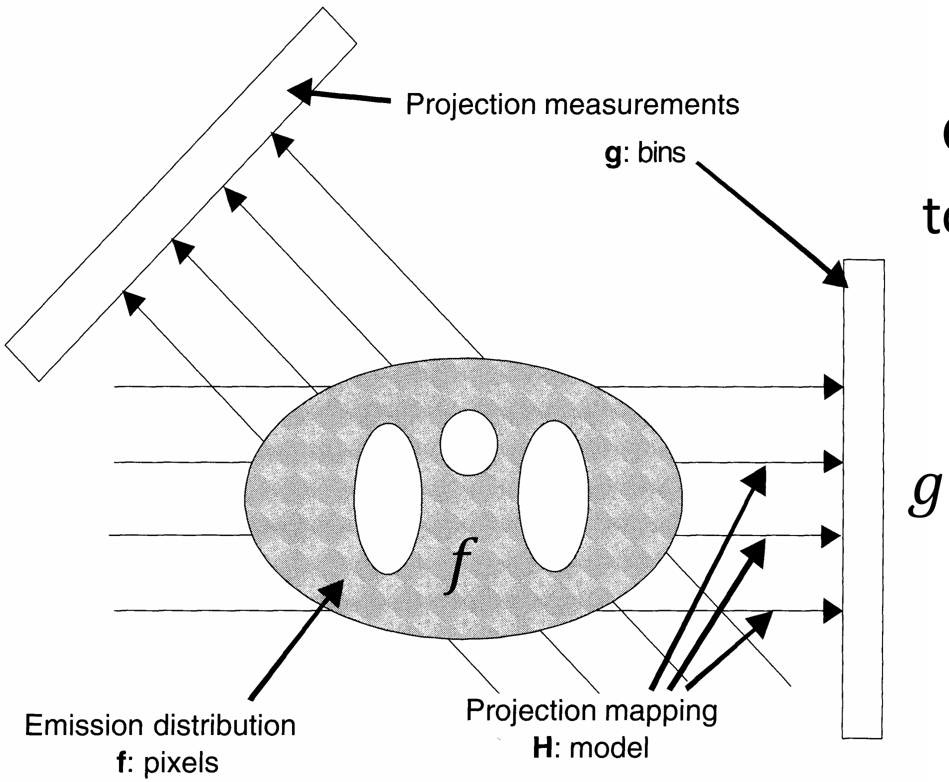


FIGURE 1 A general model of tomographic projection in which the measurements are given by weighted integrals of the emitting object distribution.

Given a measured data set g, how to formulate the ML estimator of f?

$$\hat{f}_{ML} = \arg\max P(\boldsymbol{g}|\boldsymbol{f})$$

Then how to evaluate $P(\boldsymbol{g}|\boldsymbol{f})$?

Image Reconstruction and General Linear Inverse Problem

Linear imaging systems – An alternative representation

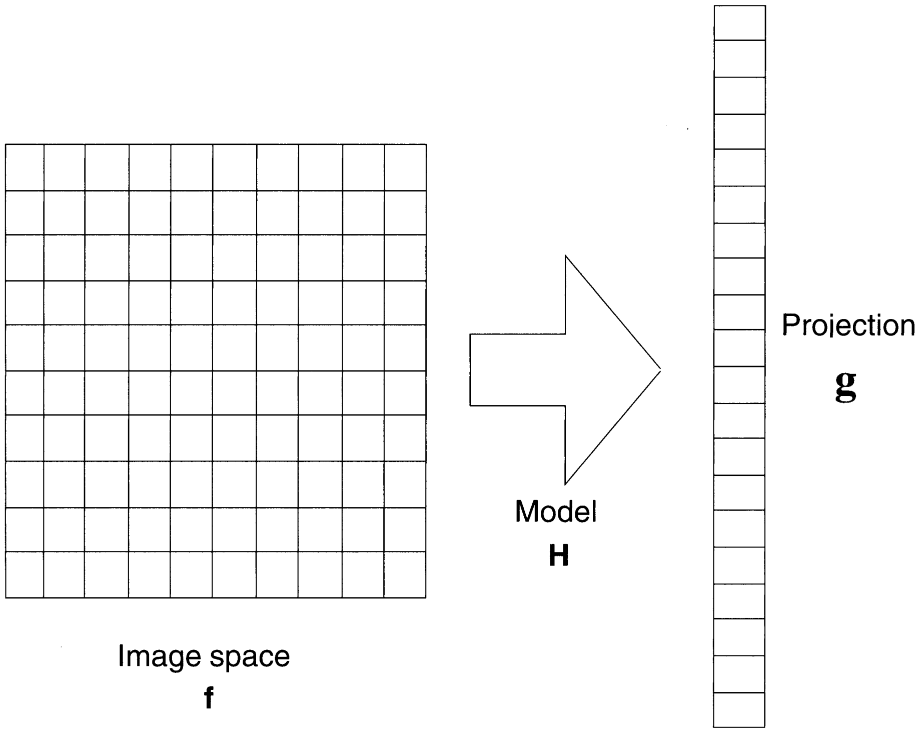


FIGURE 2 A discrete model of the projection process.

$$\bar{\mathbf{g}} = \mathbf{H}\mathbf{f}$$

Poisson Statistics of the Projection Data

In an Emission Tomography System

Considering that the measured projection data, $\boldsymbol{g}=(g_1, g_2, g_3,\ldots, g_M)$, is a collection of independent Poisson random numbers, the probability of the measured projection data $\boldsymbol{g}$ may be written as

$$p(\boldsymbol{g}|\boldsymbol{f}) = p(\boldsymbol{g}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m}$$

where $\bar{g}_m$ is the expected value for the number of counts on detector pixel no. m.

$$\bar{g}_m = \sum_{n=1}^{N} f_n \boxed{p_{mn}}$$

In the context of emission tomography, p_{mn} is the probability that a gamma ray generated at a source pixel n is detected by detector element m.

Poisson Statistics of the Projection Data

In a Transmission Tomography System

Considering that the measured projection data, $\boldsymbol{g}=(g_1, g_2, g_3, \ldots, g_M)$, is a collection of independent Poisson random numbers, the probability of the measured projection data $\boldsymbol{g}$ may be written as

$$p(\boldsymbol{g}|\boldsymbol{f}) = p(\boldsymbol{g}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m} \ ,$$

where $\bar{g}_m$ is the expected value for the number of counts on detector pixel no. m.

If the measured projection data is acquired with a transmission X-ray CT,

$$\bar{g}_m = g_{m,0} e^{-\sum_n a_{mn} \cdot f_n} \ ,$$

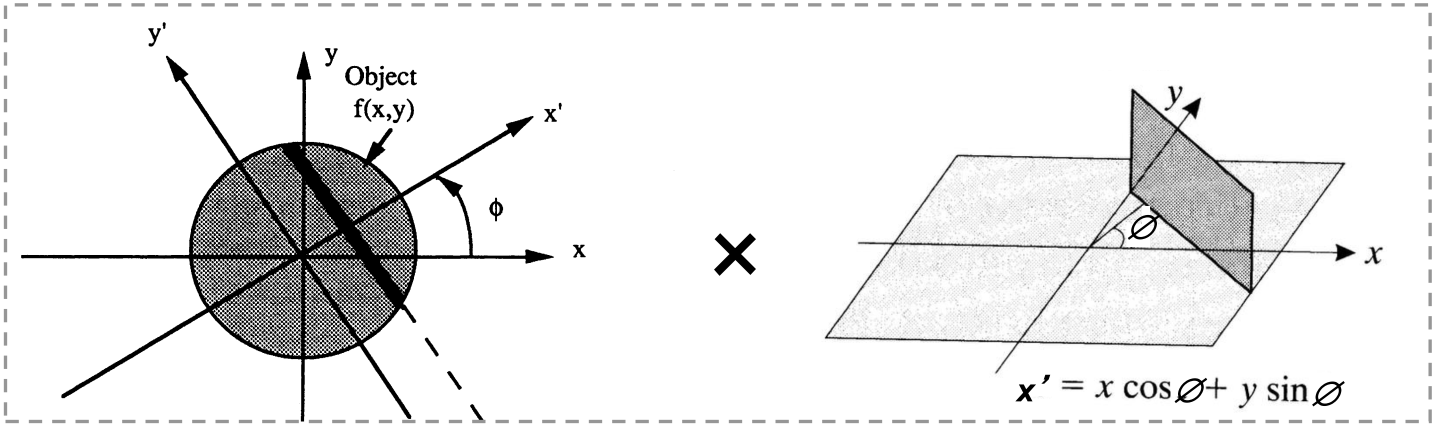
where $g_{m,0}$ is the number of photons that would have reached detector element *m* if there is no attenuation. $g_{m,0}$ can be regarded as a known constant that can be derived from the known physical design of the CT system.

What Exactly Does Projection Data Tell Us?

The Central-Slice Theorem

(or Projection-Slice Theorem)

Radon Transform



2-D integral

The value of the projection function $p_\phi(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The **Radon transform**, which produces the projection data, is defined as

$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Central Slice Theorem

Let's consider the 2D Fourier transform (FT) of an arbitrarily given function

$$F(u,v) = \int\int f(x,y) \cdot e^{-2\pi i (ux + vy)} dx\, dy\,. \qquad (1)$$

In polar coordinates and within the spatial-frequency domain, let

$$u = \rho \cos \phi \qquad \text{and} \qquad v = \rho \sin \phi,$$

Recall that
$\mathcal{F}[f(x-\tau)] = \mathcal{F}[f(x)]e^{-2\pi i\omega\tau}$,
and
$\mathcal{F}[f(x)] = 1$,
So
$\mathcal{F}[\delta(x-\tau)] = \mathrm{e}^{-2\pi i\omega\tau}$.

then (1) becomes

$$F(\rho, \phi) = \int\int f(x,y) \cdot e^{-2\pi i \rho (x \cos \phi + y \sin \phi)} dx\, dy\,. \qquad (2)$$

If we treat $(x \cos \phi + y \sin \phi)$ as a constant, then $exp\,[\,-i\, 2\pi\, \rho\, (x \cos \phi + y \sin \phi)]$ could be written as the Fourier transform of a shifted delta function. Let's write the complex exponential as the FT of a δ function,

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \mathcal{F}_{1d,x'}\{\delta[x' - (x \cos \phi + y \sin \phi]\}\ dx \cdot dy\,,$$

and

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \{\int_{x'} \delta[x' - (x \cos \phi + y \sin \phi] \cdot e^{-2\pi i \rho x'} dx'\} \cdot dx\, dy\,. \qquad (3)$$

Central Slice Theorem

$$F(\rho, \phi) = \int_y \int_x f(x,y) \cdot \{\int_{x'} \delta[x' - (x \cos\phi + y \sin\phi] \cdot e^{-2\pi i \rho x'} dx' \} \cdot dx\, dy \,. \qquad (3)$$

At this point, we can change the order of the integration operators as

$$F(\rho, \phi) = \int_{x'} [\int_y \int_x f(x,y)\, \delta(x \cos\phi + y \sin\phi - x') \,dx\, dy]\, e^{-2\pi i \rho x'} dx' \,. \qquad (4)$$

Recall that the Radon transform is defined as,

$$p(\phi, x') = \int_x \int_y [f(x,y) \cdot \delta(x \cos\phi + y \sin\phi - x')] dx\, dy \,. \qquad (5)$$

We take the 1-D Fourier transform of $p(\phi, x')$ as

$$F\{p(\phi, x')\} = \int_{x'} [\int_y \int_x f(x,y)\, \delta(x \cos\phi + y \sin\phi - x') \,dx\, dy]\, e^{-2\pi i \rho x'} dx' \,, \qquad (6)$$

which is exactly what we wrote for $F(\rho, \phi)$ above!

Therefore, we have proved that

$$\mathcal{F}_{1d,x'}[p(\phi, x')] = F(\rho, \phi).$$

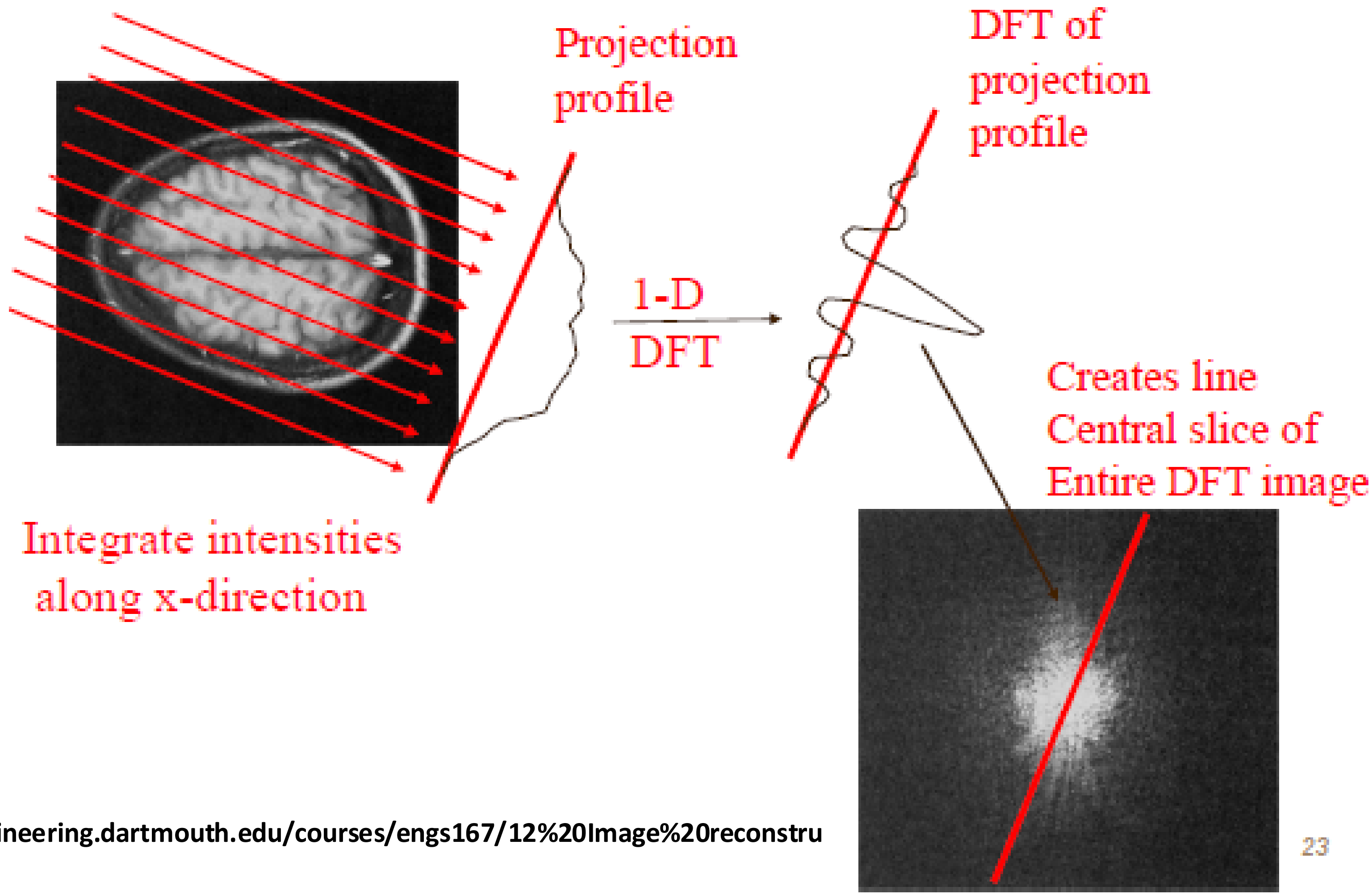
$F(\rho, \phi)$ is defined in Eq. (2).

Projection Slice Theorem or Central Slice Theorem

An alternative proof can be found on Page 77 in Foundations of Medical Imaging, by Z. H. Cho.

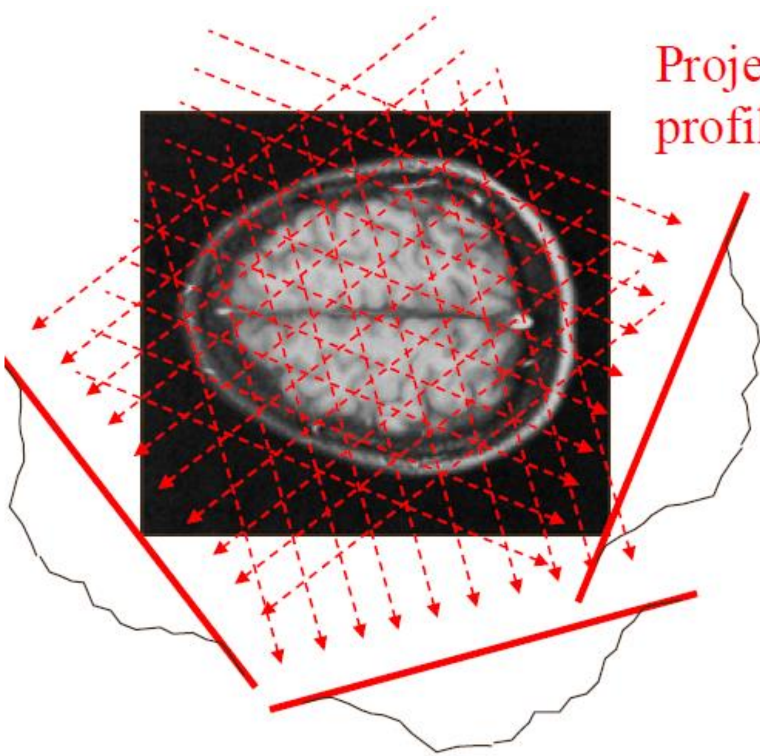
Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem

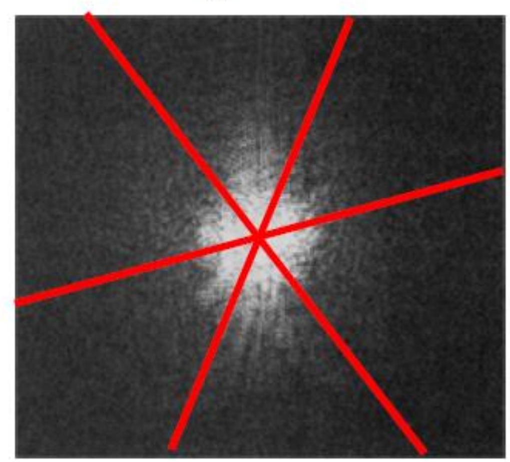


Projection profiles

Integrate intensities along x-direction

1-D DFTs

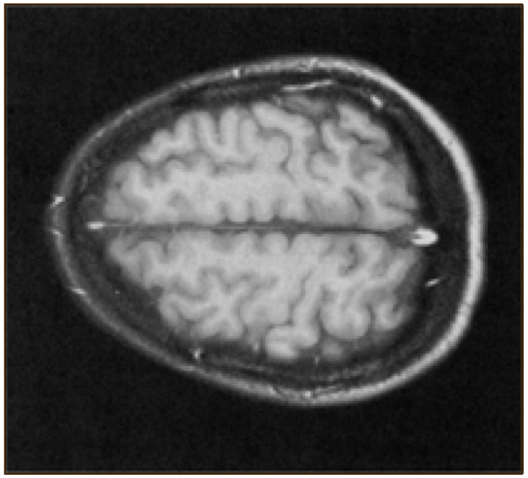
Create lines in central slice of entire DFT image



The more angles used, the better the Fourier space image is filled

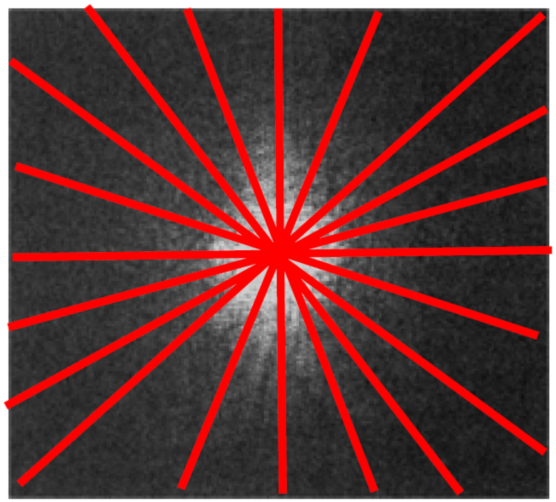
http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem



DFT image represents integration of original projections DFT transformed and summed together.

1-D DFTs at each projection

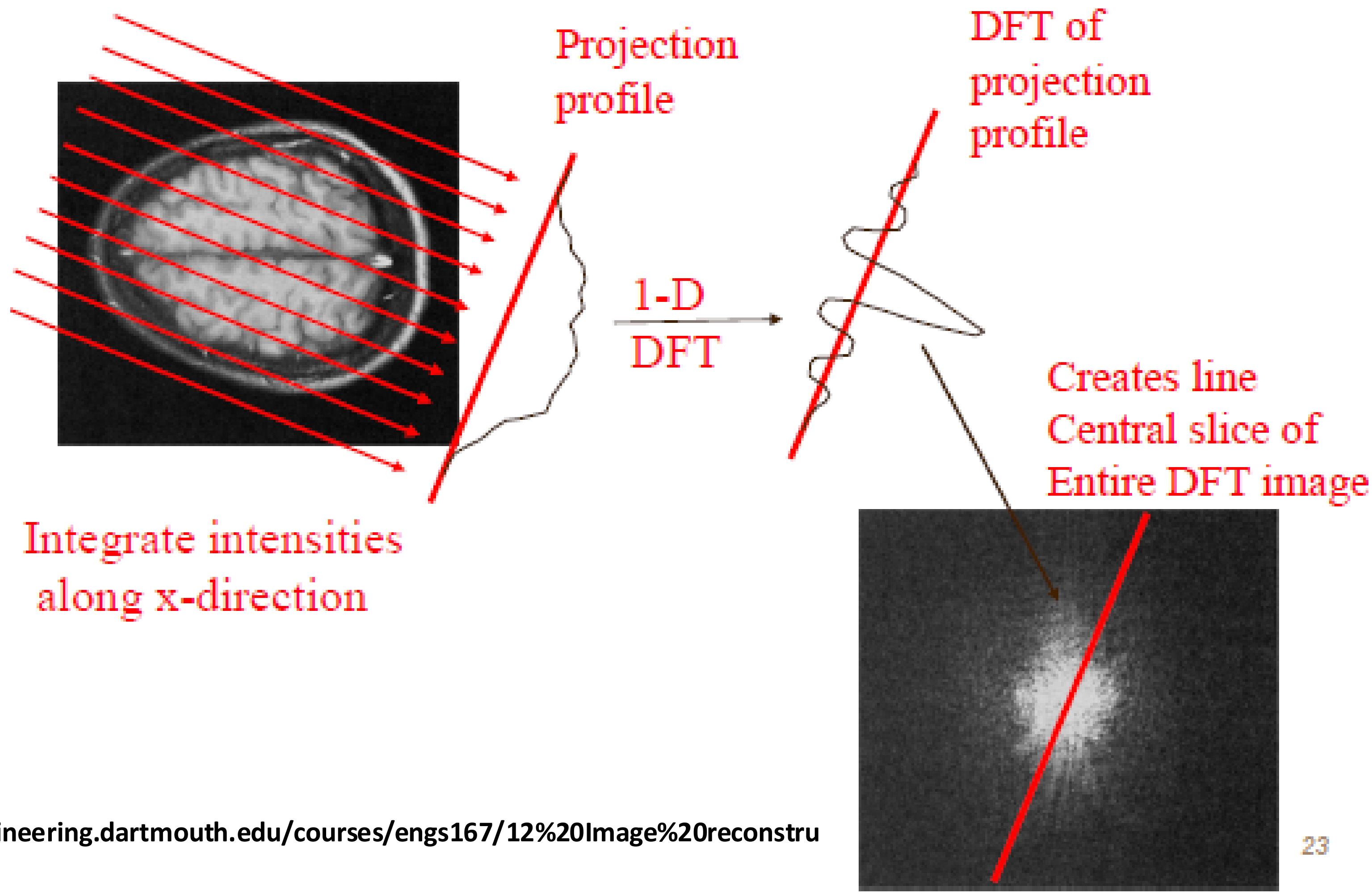


This is the fast way to create the DFT image from projection data. The more projections taken, the more complete the sampling.

http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,\,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

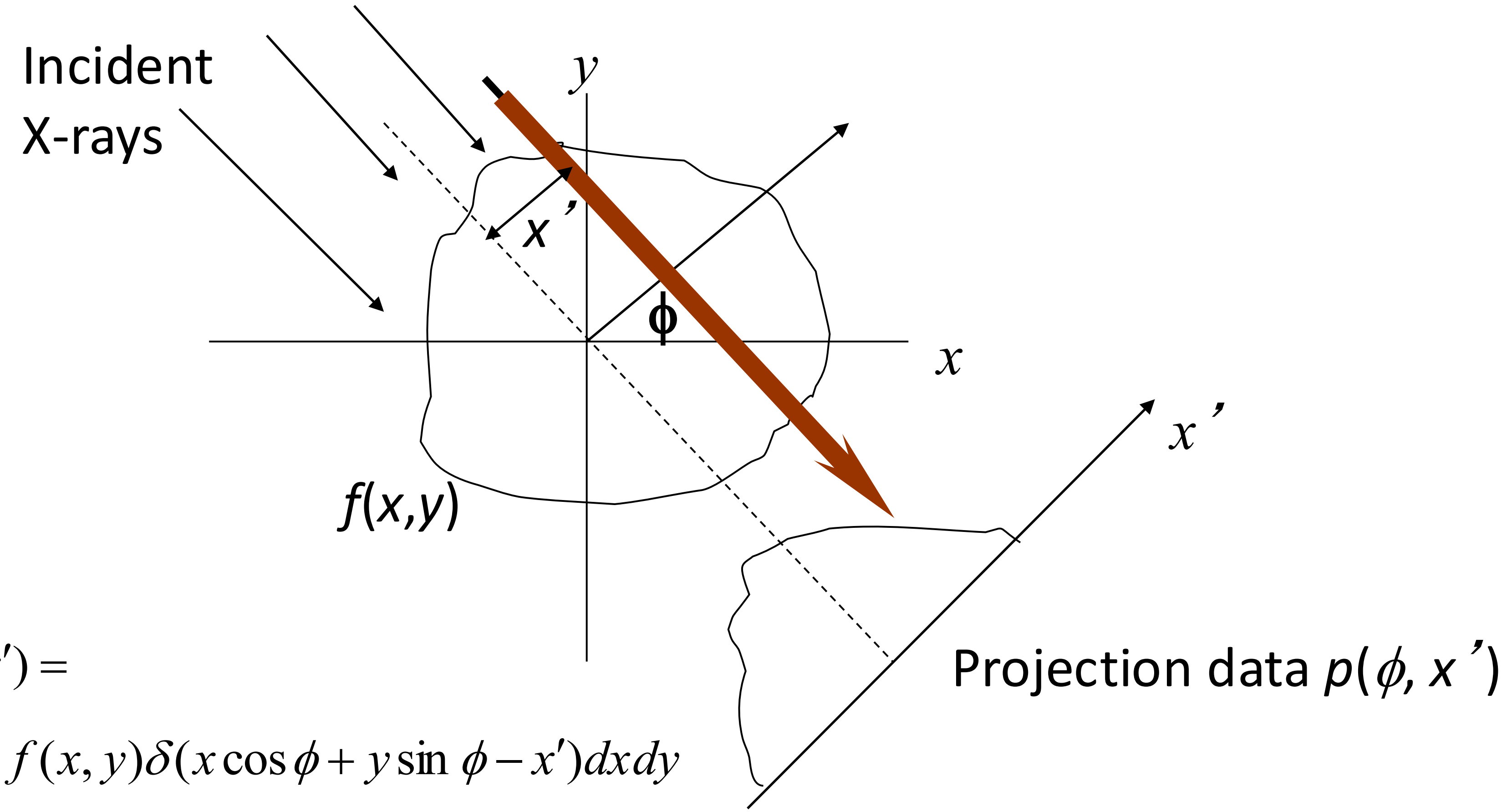
Central Section Or Projection Slice Theorem

$$\mathcal{F}_{1d,x'}[p(\phi, x')] = F(\rho, \phi).$$

So, in words, the Fourier transform of a projection at an angle ϕ gives us a line in the polar Fourier space at the same angle ϕ.

The central slice theorem is the key to understanding reconstructions from projection data.

Radon Transform and Projection Data (Revisited)

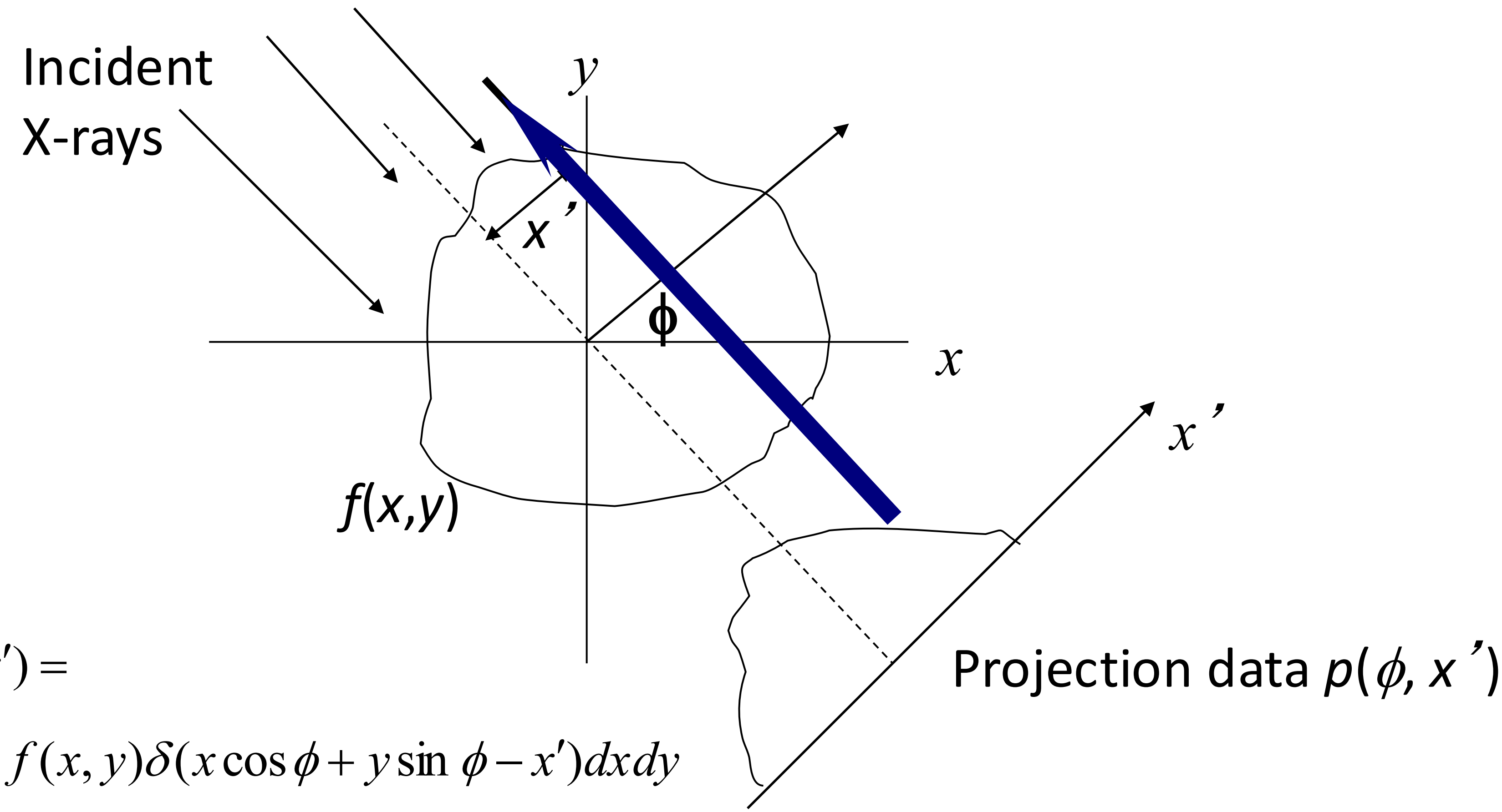


$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

If we know what the projection data is telling us about the underlying image, how can we reconstruct the unknown image function?

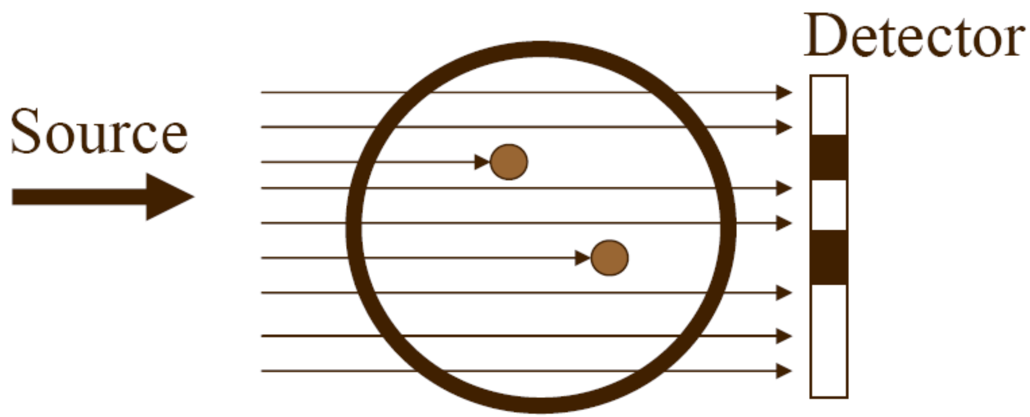
Simple Backprojection

Simple Backprojection



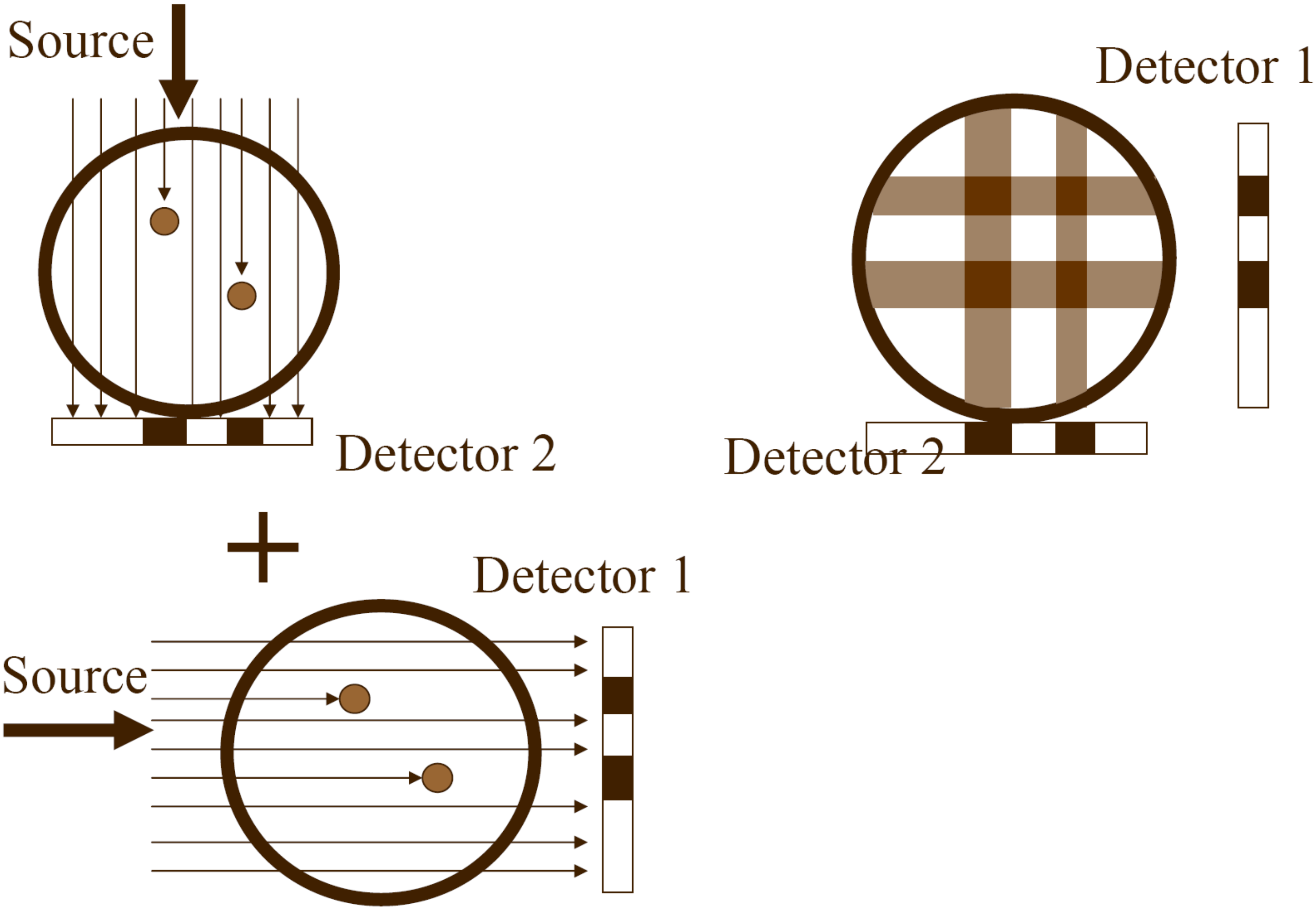
$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Simple Backprojection

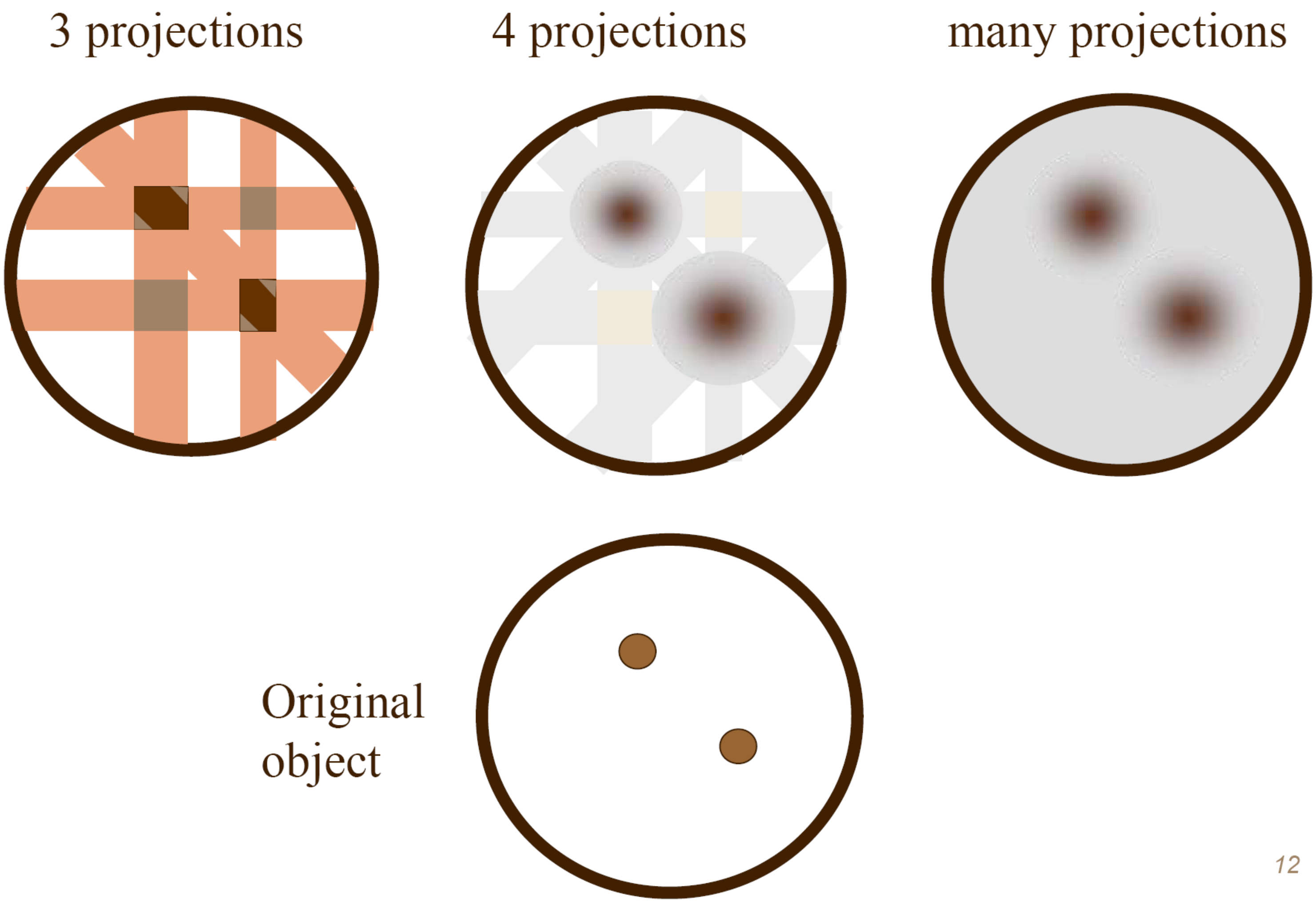


inverse = 1 back projection

Simple Backprojection



Simple Backprojection



Simple Backprojection

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos \phi + y \sin \phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{\text{simple back-projection}}(x,y) = \int b_\phi(x,y) \, d\phi$$

$$f_{\text{simple back-projection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \, \delta(x \cos \phi + y \sin \phi - x') \, dx'$$

Simple Back-projection and the 1/r Blurring

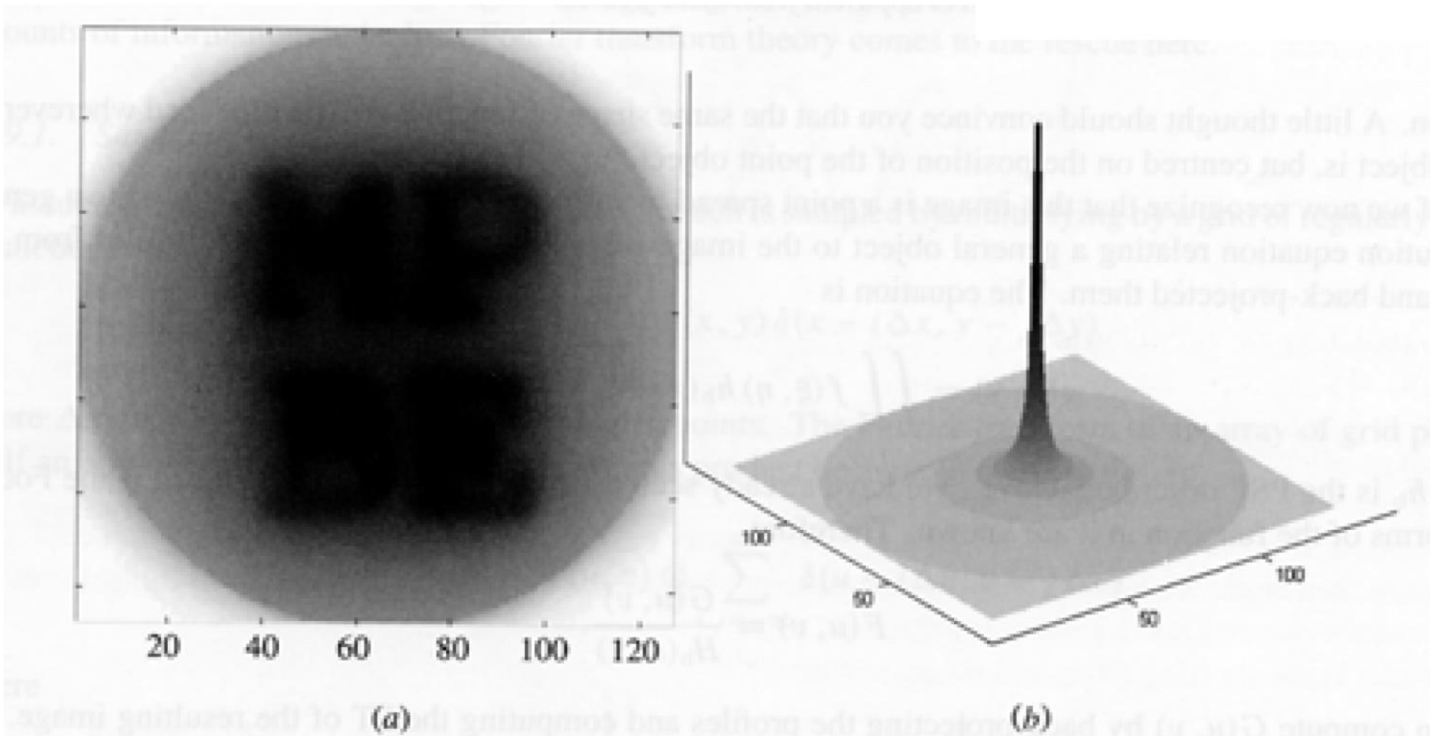


Figure 11.14. *(a) Image produced by back-projecting the sinogram given in figure 11.12(b); (b) the response to a point object in the middle of the field of view.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Impulse Response Function of the Simple Backprojection Operator

$h_b(r) = 1/r$

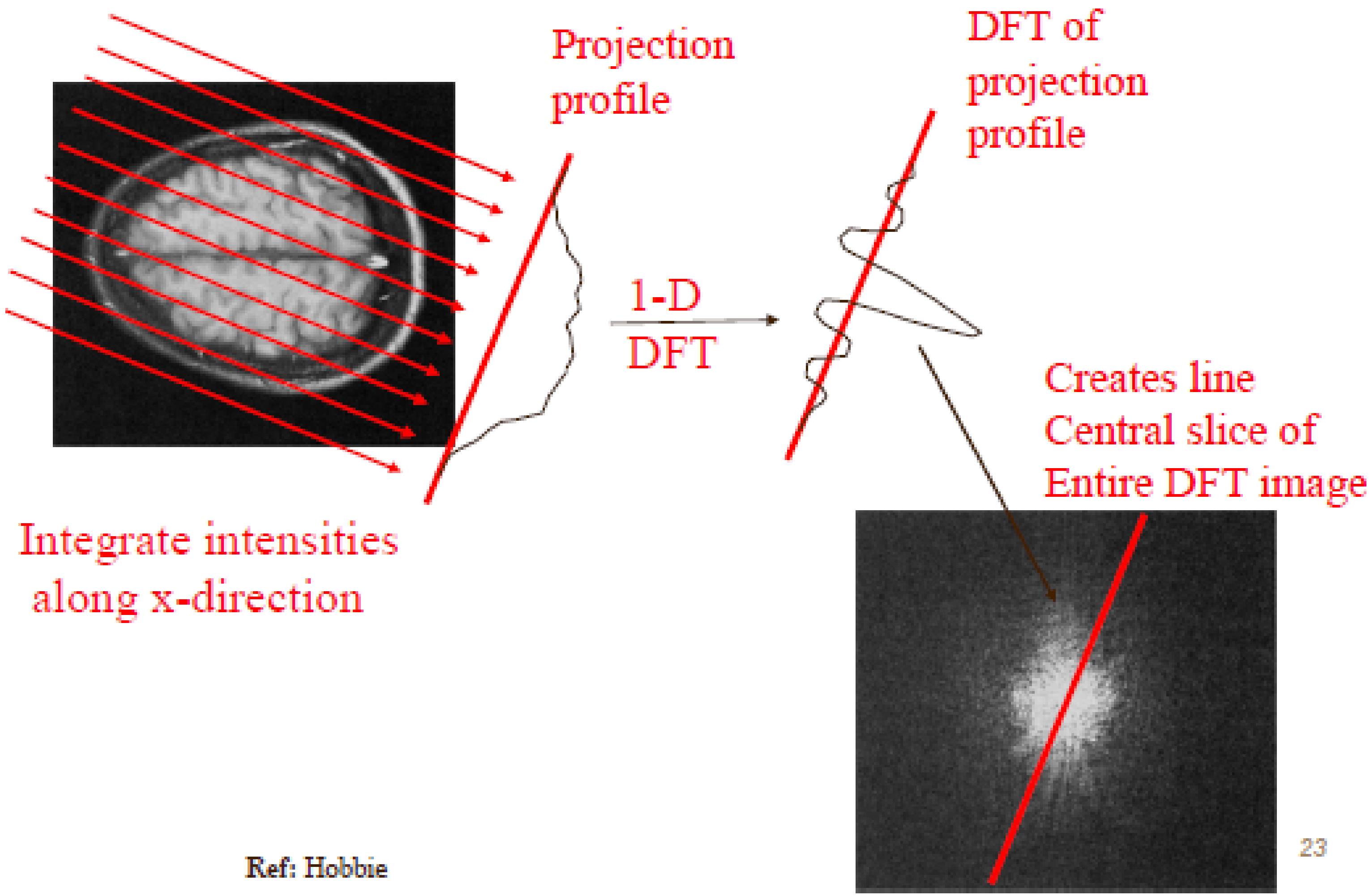
$f_b(x,y) = f(x,y) * 1/r$

$F_b(\rho,\phi) = F(\rho, \phi) / \rho$, since $F\{1/r\} = 1/\rho$

Backprojected image is blurred by convolution with $1/r$.

Central Slice Theorem

$$\mathcal{F}_{1d,x'}[p\,(\phi\,,x')\,] = F(\rho,\,\phi\,)$$



http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Simple Back-projection and the 1/r Blurring

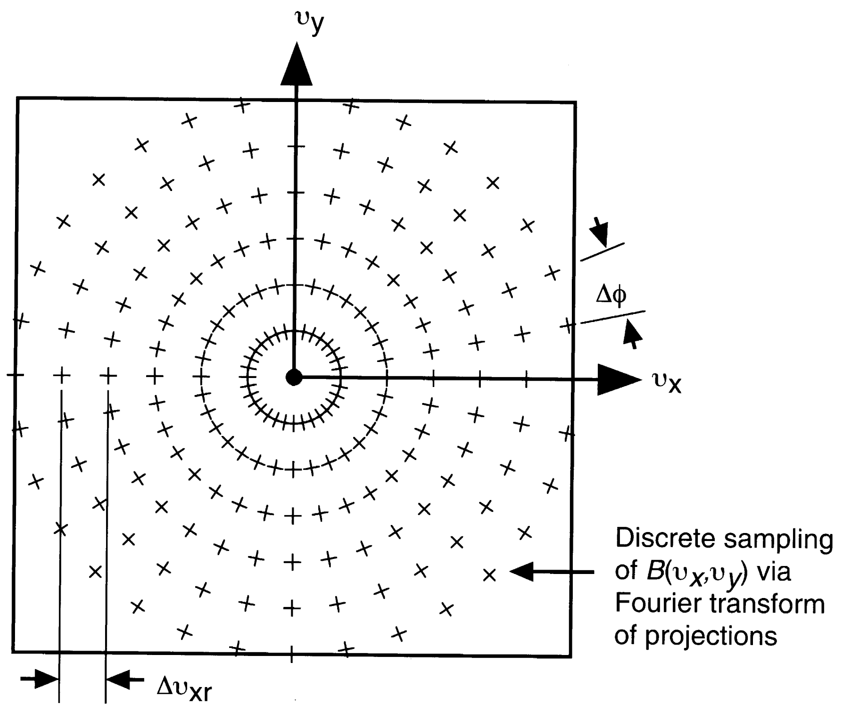


FIGURE 18 The discrete sampling pattern of $F(\upsilon_x,\upsilon_y)$ contained in $B(\upsilon_x,\upsilon_y)$, resulting from the use of discretely sampled projections.

The nature of the 1/r blurring:

Radon transform produces equally spaced radial sampling in the Fourier domain.

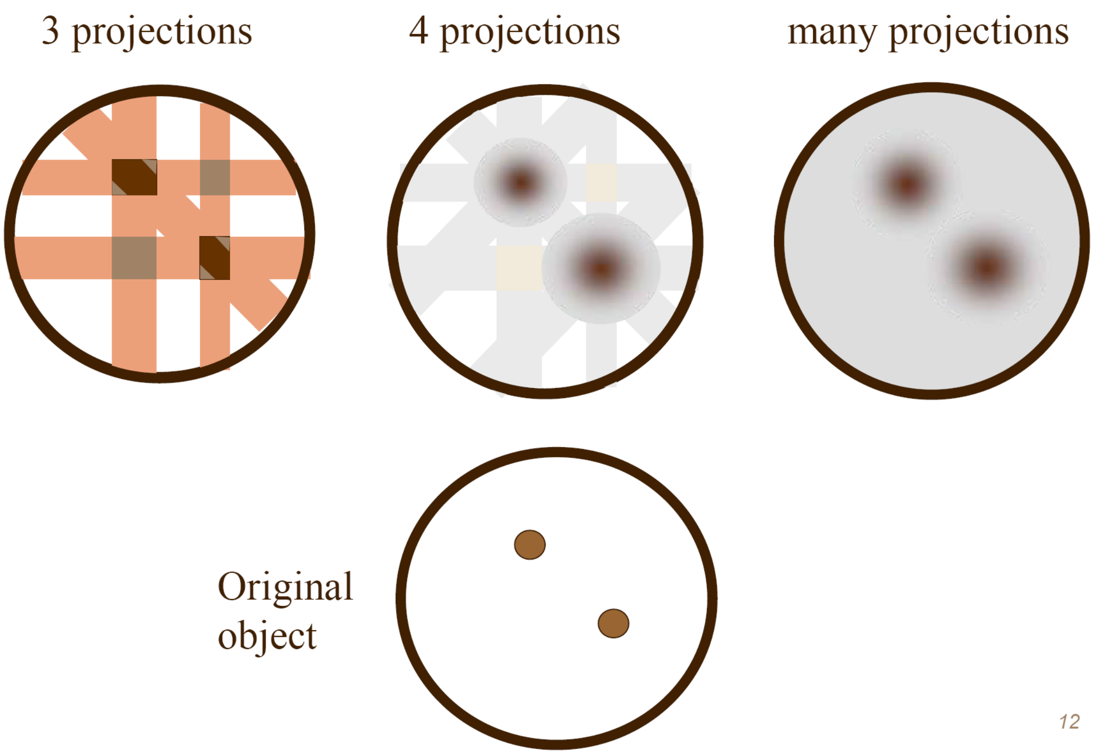
2-D Inverse Radon Transform

and Filtered Backprojection

Simple Backprojection (Revisited)

Reconstruction with simple backprojection

$$\hat{\boldsymbol{f}}_{\text{simple backprojection}}(x,y) = \int_{0}^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x')\,dx'$$



From Simple Backprojection to Inverse Radon Transform

Simple backprojection

Result from a back projection from a single angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos \phi + y \sin \phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$\hat{f}_{\text{simple backprojection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Inverse Radon transform

$$\hat{f}_{\text{Inverse Radon transform}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} F^{-1}\{F_{1d,x'}[p_\phi(x')] \cdot |w|\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Review of Fourier Transform and Filtering Spectral Filtering

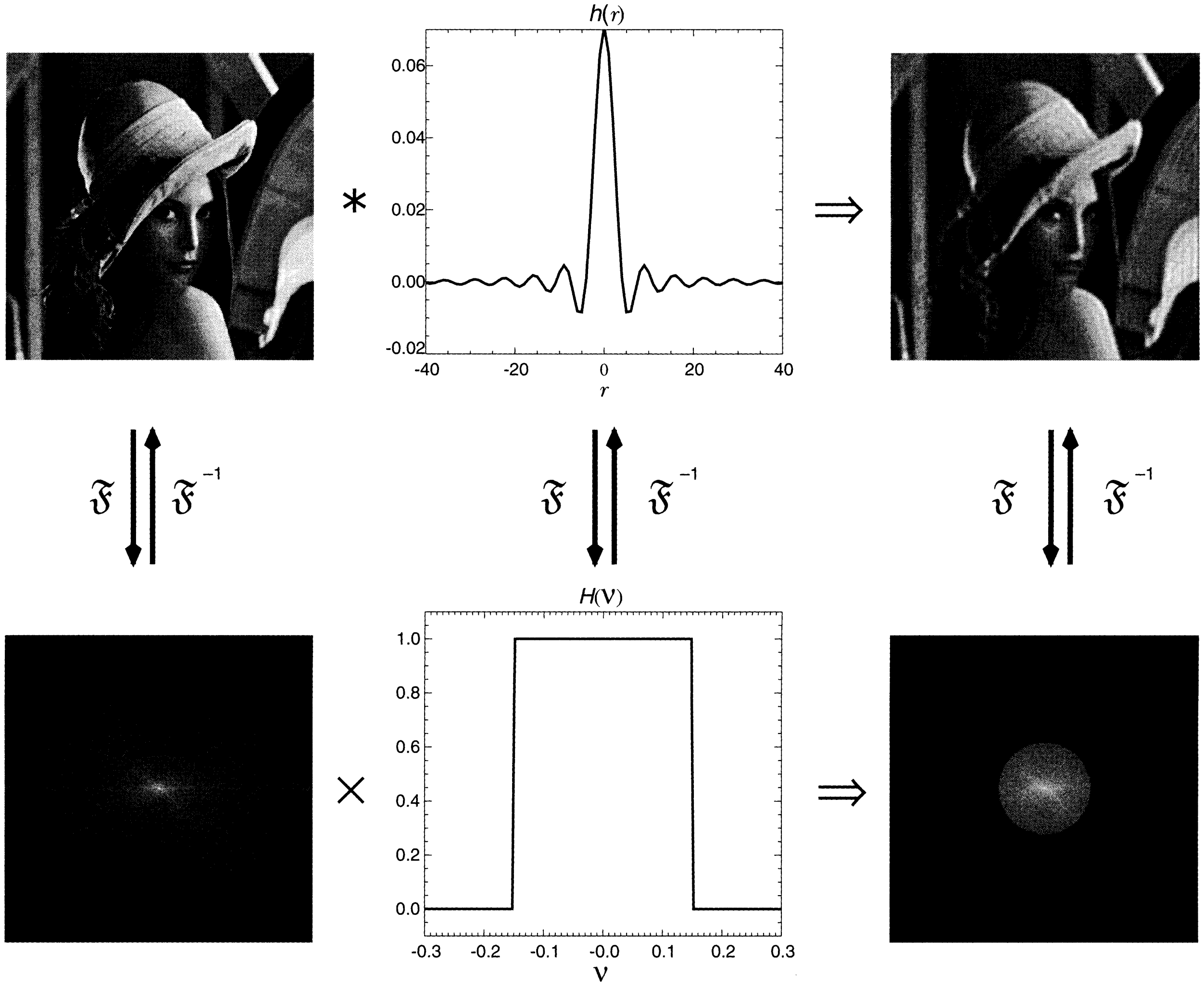


FIGURE 3 Ideal lowpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column are the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2+y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Review of Fourier Transform and Filtering
Spectral Filtering

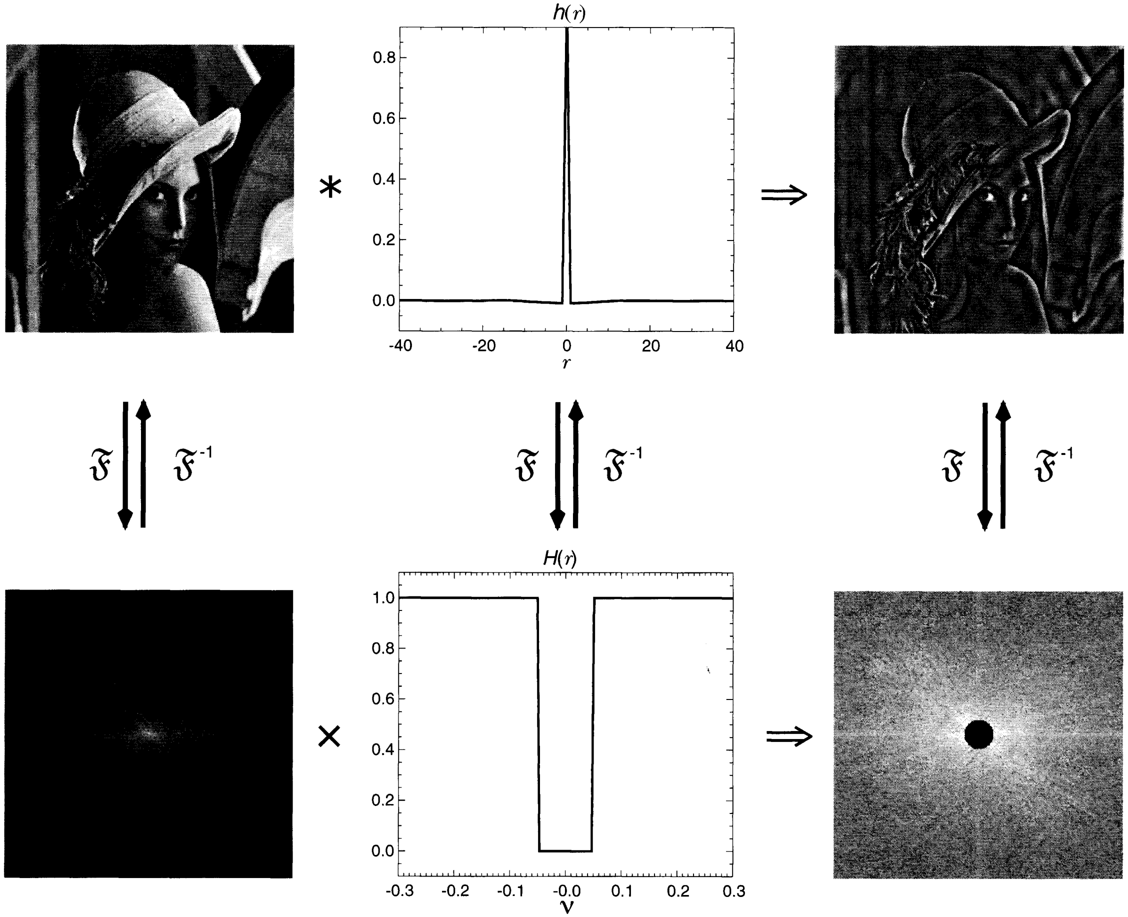
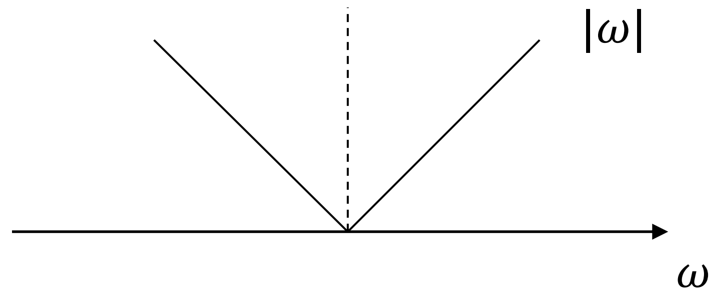


FIGURE 4 Ideal highpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column shows the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2 + y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Filtered Back-projection

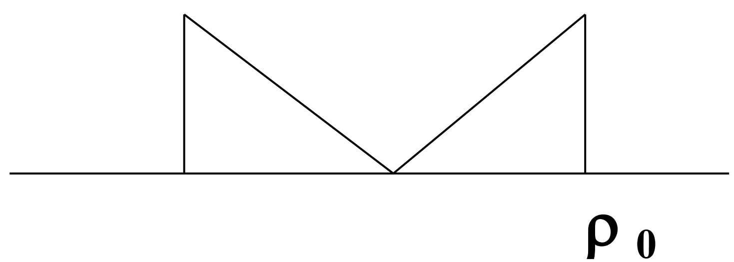
Ideal filter



The Ram-Lak filter

$$H_{\mathrm{RL}}(\omega) = \begin{cases} |\omega|, & (|\omega| \leq 2\pi B) \\ 0, & (\text{otherwise}) \end{cases}$$

Ram-Lak filter



From Inverse Radon Transform to Filtered Backprojection (FBP)

Simple backprojection

$$\hat{\boldsymbol{f}}_{\text{simple backprojection}}(x,y)=\int_0^\pi d\phi\int_{-\infty}^{\infty}p_\phi(x')\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

Inverse Radon transform

$$\hat{\boldsymbol{f}}_{\text{inverse Radon transform}}(x,y)=\int_0^\pi d\phi\int_{-\infty}^{\infty}F^{-1}\{F[p_\phi(x')]\cdot|\omega|\}\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

Filtered Backprojection (FBP)

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y)=\int_0^\pi d\phi\int_{-\infty}^{\infty}F^{-1}\{F[p_\phi(x')]\cdot|\omega|\cdot H'(\omega)\}\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

or

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y)=\int_0^\pi d\phi\int_{-\infty}^{\infty}F^{-1}\{F[p_\phi(x')]\cdot H(\omega)\}\cdot\delta(x\cos\phi+y\sin\phi-x')dx'$$

From Inverse Radon Transform to Filtered Backprojection (FBP)

Filtered Backprojection (FBP)

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot |\omega| \cdot H'(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx' ,$$

or

$$H(\omega) = |\omega| \cdot H'(\omega)$$

$$\hat{\boldsymbol{f}}_{\text{FBP}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

$$= \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi^*(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx' ,$$

where the **filtered projection**, $p_\phi^*(x')$, is defined as

$$p_\phi^*(x') \equiv F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \equiv p_\phi(x') * h(x')$$

and $H(\omega) \equiv F[h(x')]$ is the filter defined in spatial frequency domain.

Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx',$$

where $H(\omega) = |\omega| \cdot H'(\omega)$.

The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot h(x\cos\phi + y\sin\phi - x')dx' ,$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

Simple and Filtered Back-projection

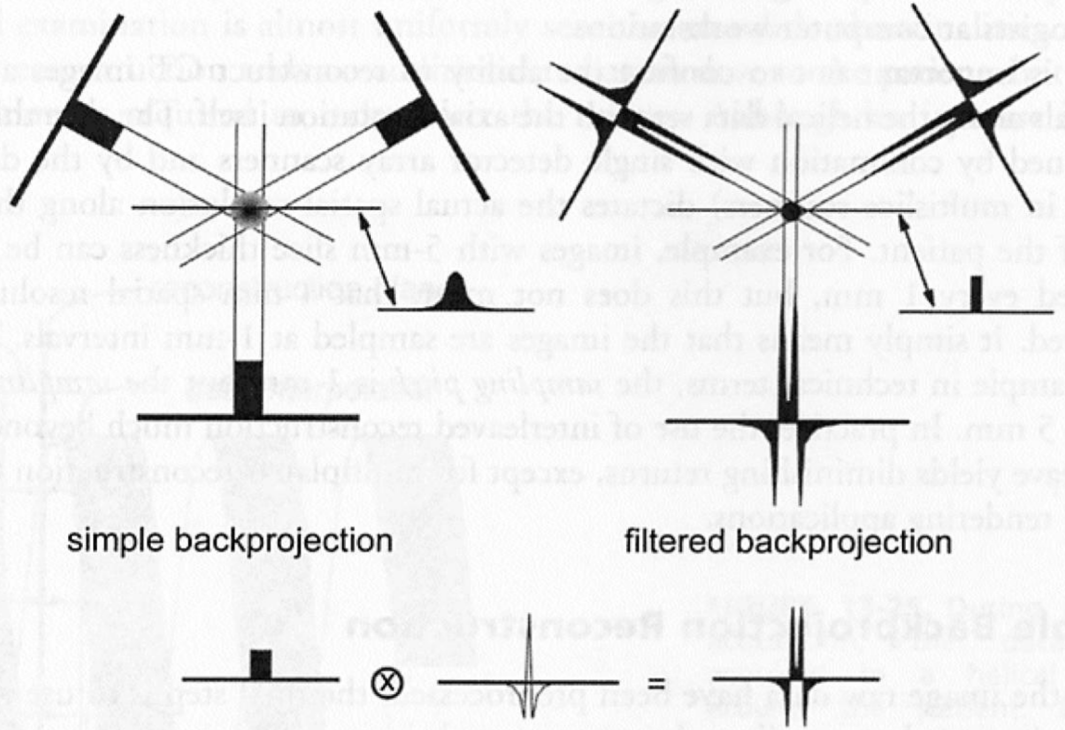
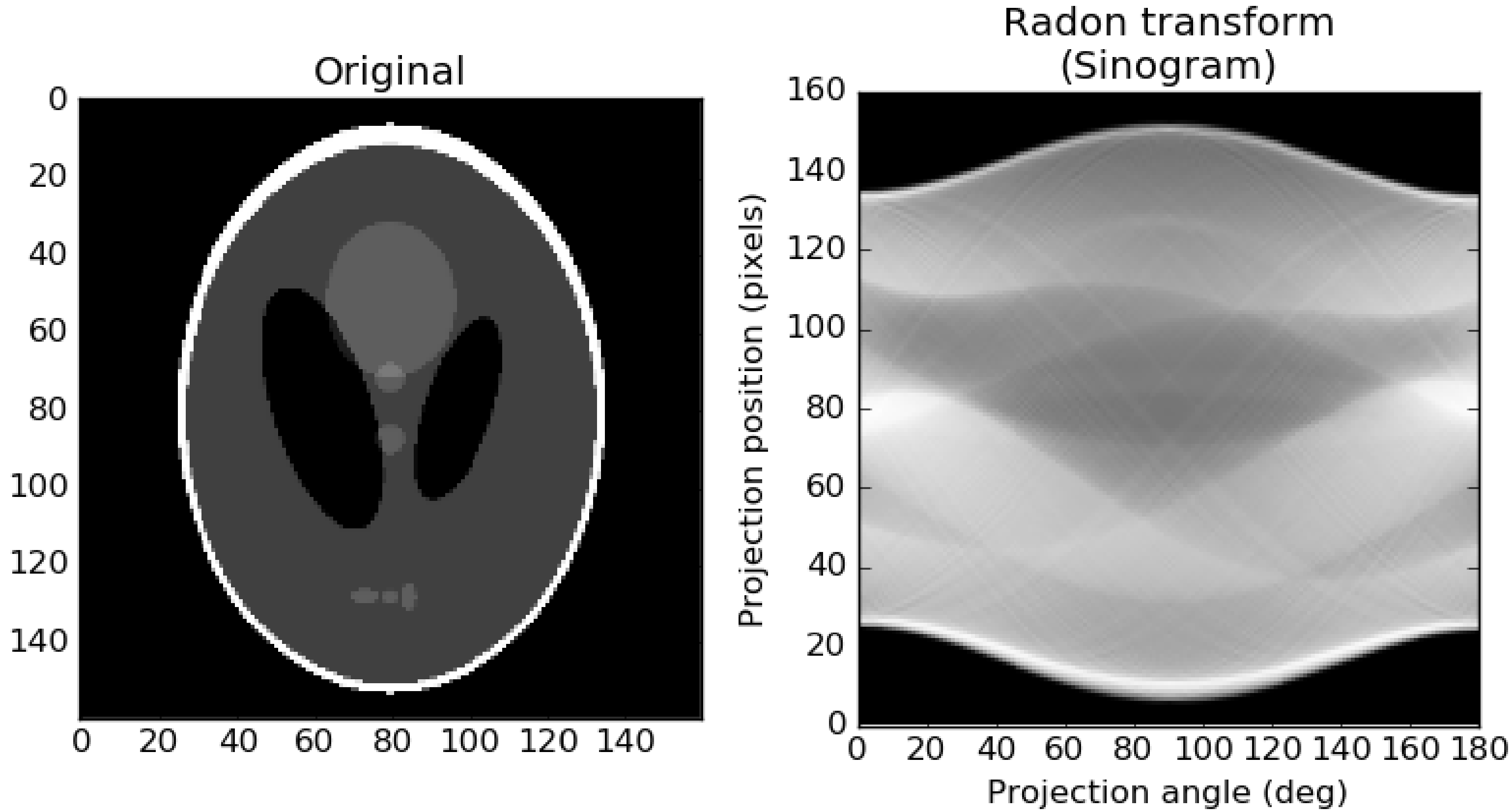


FIGURE 13-28. Simple backprojection is shown on the left; only three views are illustrated, but many views are actually used in computed tomography. A profile through the circular object, derived from simple backprojection, shows a characteristic 1/r blurring. With filtered backprojection, the raw projection data are convolved with a convolution kernel and the resulting projection data are used in the backprojection process. When this approach is used, the profile through the circular object demonstrates the crisp edges of the cylinder, which accurately reflects the object being scanned.

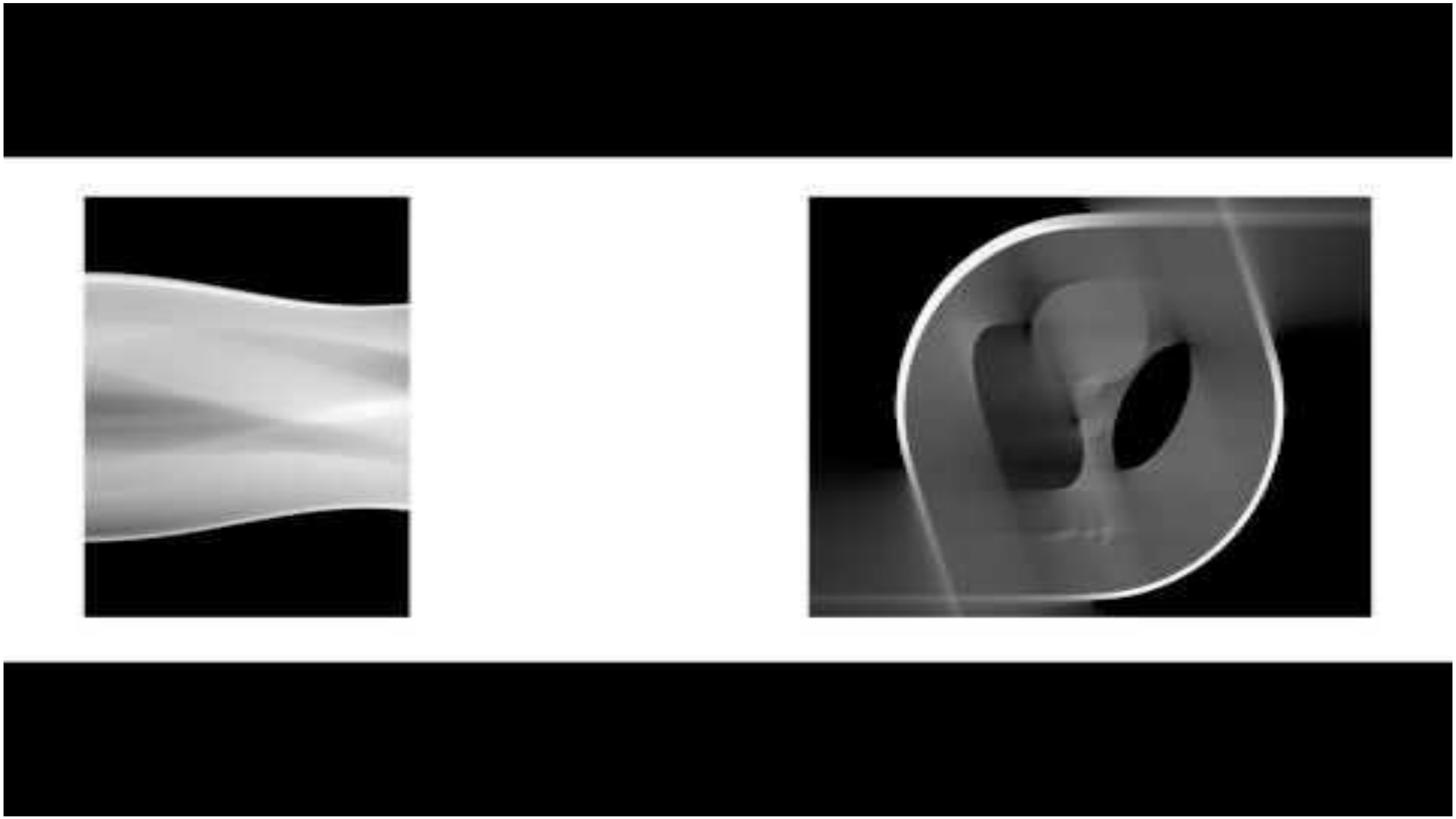
Chapters 12 & 13, The Essential Physics of Medical Imaging, Bushberg

Radon Transform and Sinogram



http://tech.snmjournals.org/cgi/content-nw/full/29/1/4/F3

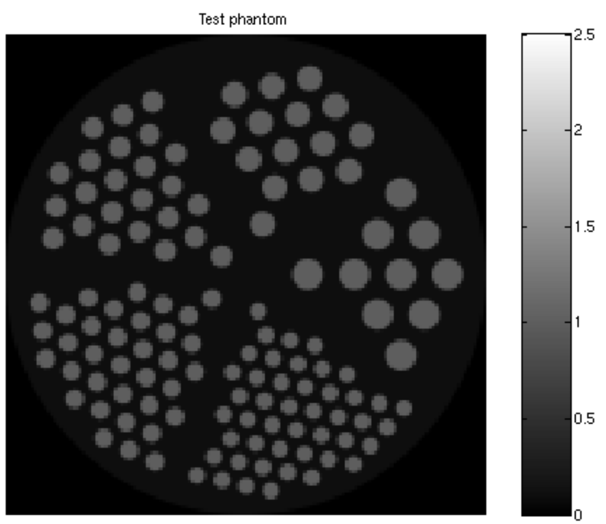
Filtered Back-projection

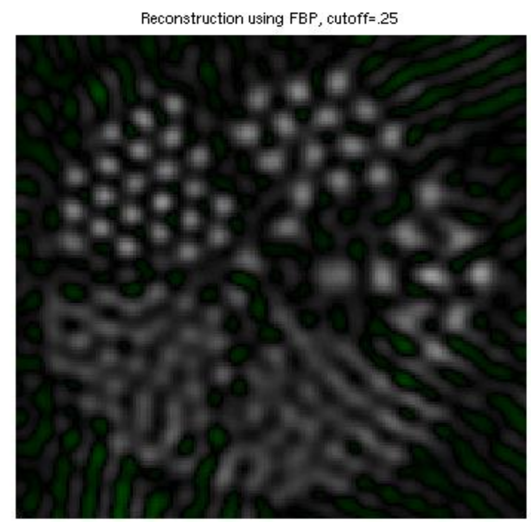


https://www.youtube.com/watch?v=ddZeLNh9aac

Filtered Back-projection

The “ringing” artifacts in reconstructed images.





In this context it is usually manifest itself as "streak artifacts". You may, for example, see this as lines radiating from the center and outwards. The term comes from electronics and is meant in the sense of a bell-ringing.

Filtered Back-projection

The sharp boundary of the Ram-Lak filter often makes the spatial domain filter kernel oscillatory.

It sometimes introduces the "ringing" artifacts in reconstructed images.

This can be effectively resolved by the Shepp and Logan filter

$$H_{SL}(\omega) = \begin{cases} |\omega| \sin\left(\dfrac{\omega}{4B}\right), & |\omega| < 2\pi B \\ 0, & \text{otherwise} \end{cases}$$

$$h_{SL}(x) = \frac{B}{\pi^2}\left\{\frac{1-\cos 2\pi B[(1/4B)+x]}{(1/4B)+x} + \frac{1-\cos 2\pi B[(1/4B)-x]}{(1/4B)-x}\right\}$$

An Alternative View of the Filters

In the spatial frequency domain

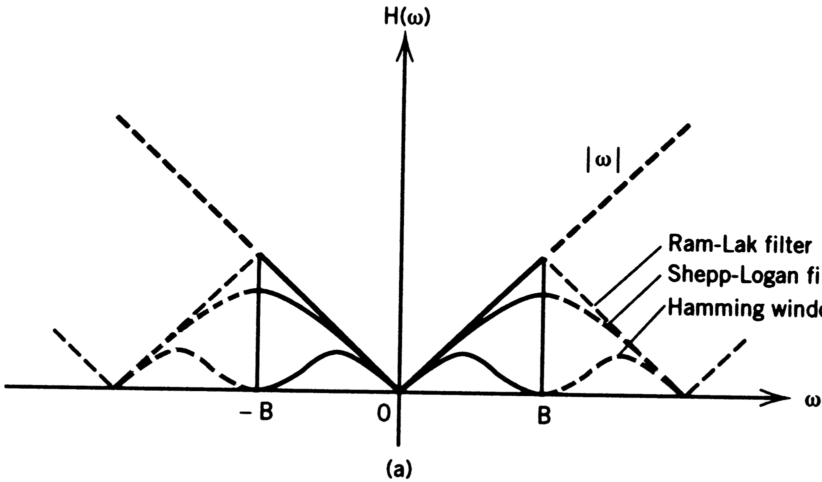


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

$$H(\omega)$$

In the spatial domain

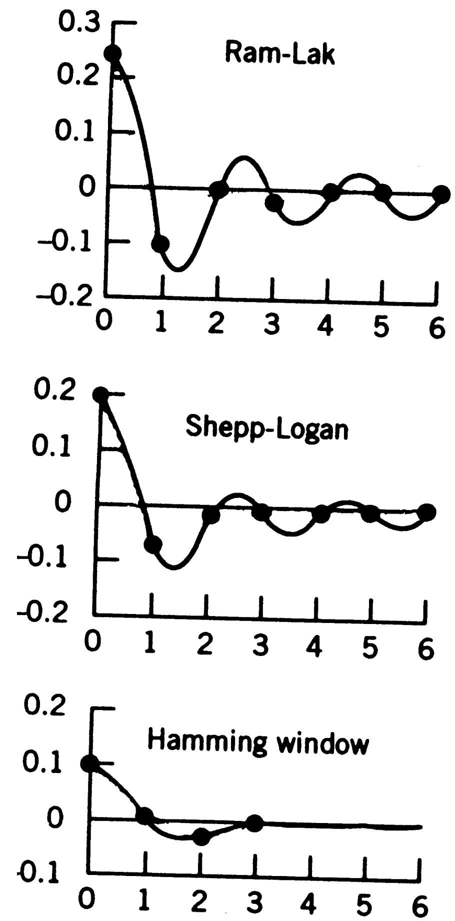


Figure 3-4 (b) Spatial domain filter kernals corresponding to the filter functions shown in the Ram-Lak filter is a high-pass filter with a sharp response but results in some noise enhancement, while the Shepp-Logan and the Hamming window filters are noise-smoothed filters and therefore have better SNR.

Filtered Back-projection

- Low spatial frequency data is overweighted. Filter to compensate for this. Weighted by $1/\omega$.

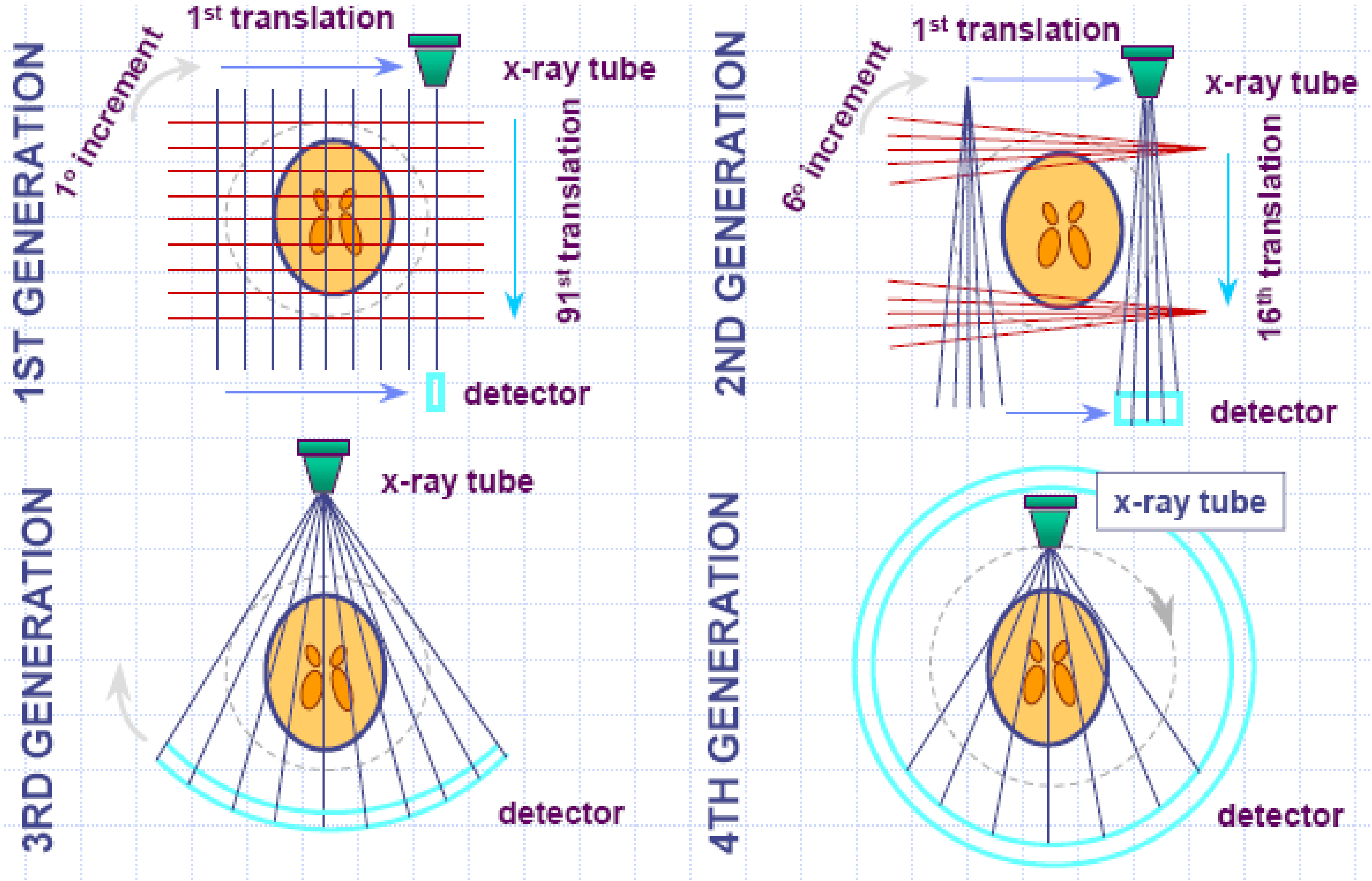
- Solution - filter each projection by $|\omega|$ to account for the uneven sampling density

Steps:

1) Fourier transform of the projection
2) Weight with $|\omega|$
3) Inverse transform
4) Back project
5) Add all angles

FBP in Fan-Beam Mode

X-ray Computed Tomography (CT)



Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_\phi(x')] \cdot H(\omega)\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx',$$

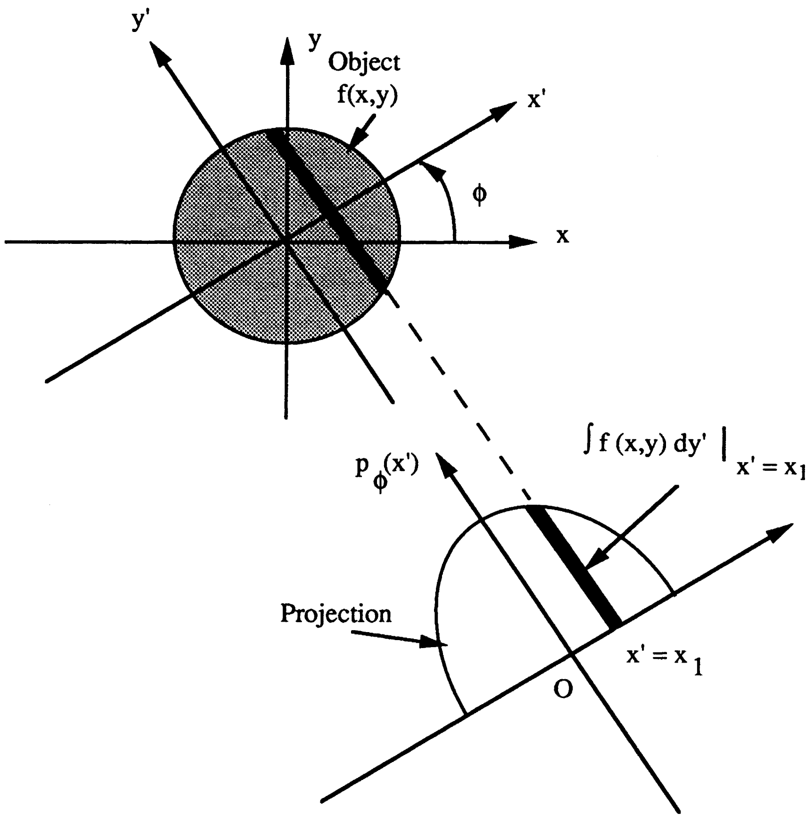
where $H(\omega) = |\omega| \cdot H'(\omega)$.

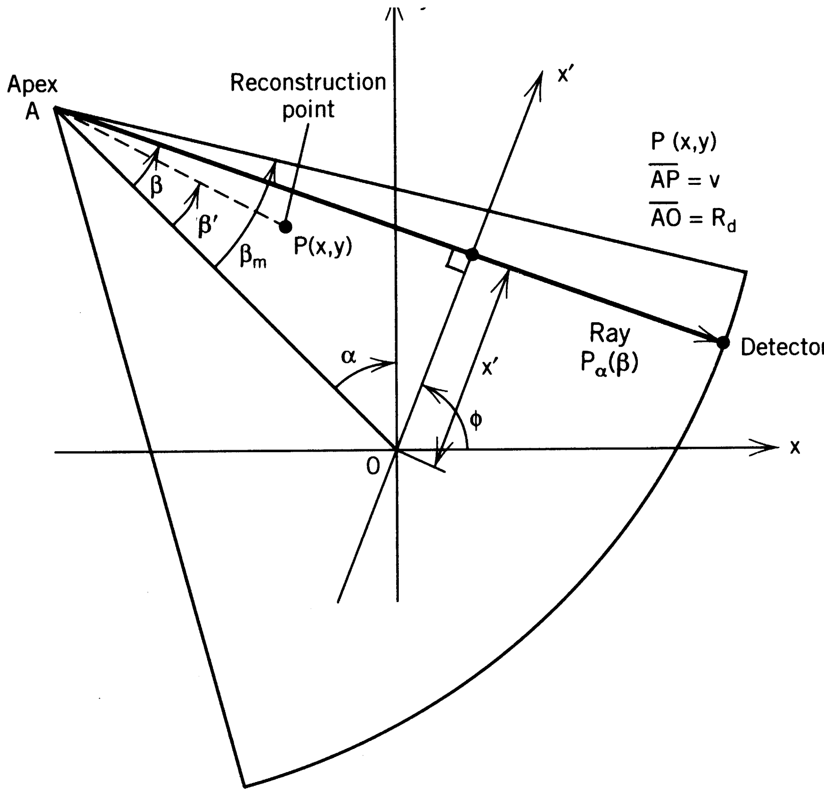
The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x,y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_\phi(x') \cdot h(x\cos\phi + y\sin\phi - x')dx',$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

Filtered Back-projection in Fan-Beam Mode

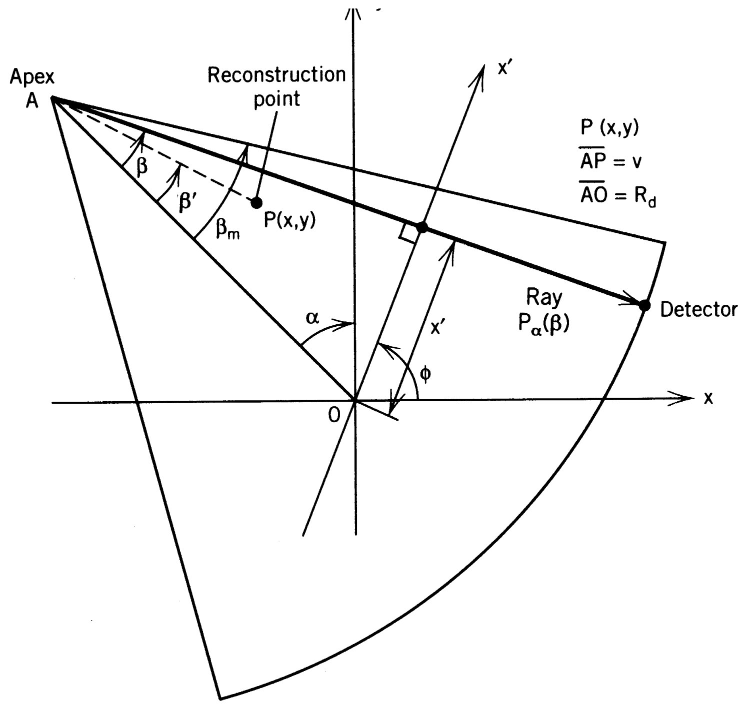




Comparing parallel beam and fan beam geometries

Filtered Back-projection in Fan-beam Mode

Since



$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

The absolute value of the determinant of the Jacobian |J| is

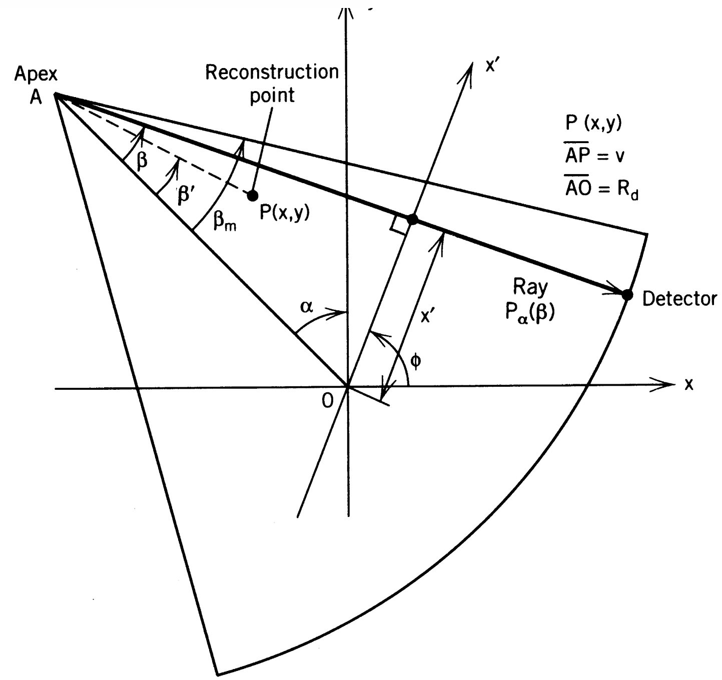
$$|J| = \left|\frac{\partial(x'\phi)}{\partial(\alpha,\beta)}\right| = \begin{vmatrix} \dfrac{\partial x'}{\partial\alpha} & \dfrac{\partial x'}{\partial\beta} \\ \dfrac{\partial x'}{\partial\alpha} & \dfrac{\partial\phi}{\partial\beta} \end{vmatrix} = R_d \cos\beta$$

Filtered Back-projection in Fan-beam Mode

where β' is the angle between the central line and the line passing through the reconstructed point at (x,y). And v is the distance between the apex of the fan and the reconstruction point p.

$$v = \sqrt{(x\cos\alpha + y\sin\alpha)^2 + (x\sin\alpha - y\cos\alpha + R_d)^2}$$

$$\beta' = \tan^{-1}\left[\frac{x\cos\alpha + y\sin\alpha}{x\sin\alpha - y\cos\alpha + R_d}\right]$$



$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

Filtered Back-projection in Fan-beam Mode (continued)

FBP in parallel beam case

$$\hat{f}(x, y) = \frac{1}{2\pi}\int_0^{2\pi} d\phi \int_{-\infty}^{\infty} dx'\, p_\phi(x') h(x\cos\phi + y\sin\phi - x')$$

$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

can be converted to FBP in fan beam mode by changing integration variables

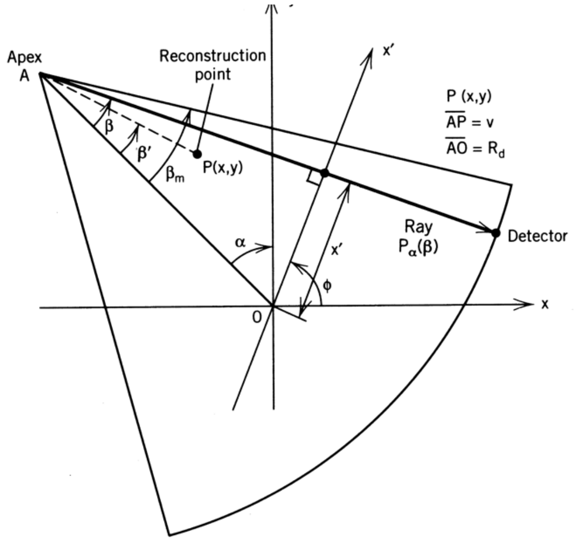
$$\hat{f}(x, y) = \frac{1}{2\pi}\int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta$$

$$\times p_\alpha(\beta) h\{x\cos(\alpha + \beta) + y\sin(\alpha + \beta) - R_d \sin\beta\}|J|$$

Filtered Back-projection in Fan-beam Mode

Where the Jacobian |J| is

$$|J| = \left| \frac{\partial(x', \phi)}{\partial(\alpha, \beta)} \right| = R_d \cos\beta$$



and the FBP in fan beam mode becomes

$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta\, p_\alpha(\beta) h\{v \sin(\beta' - \beta)\} |J|$$

Filtered Back-projection in Fan-beam Mode (continued)

FBP in the parallel-beam case

$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\phi \int_{-\infty}^{\infty} dx' \, p_\phi(x') h(x \cos\phi + y \sin\phi - x')$$

$$x' = R_d \sin\beta$$

$$\phi = \alpha + \beta$$

can be converted to **FBP in fan-beam mode** by changing integration variables

$$\hat{f}(x, y) = \frac{1}{2\pi} \int_0^{2\pi} d\alpha \int_{-\beta_m}^{\beta_m} d\beta \times p_\alpha(\beta) h\{x \cos(\alpha + \beta) + y \sin(\alpha + \beta) - R_d \sin\beta\} |J|$$

$$|J| = \left| \frac{\partial(x', \phi)}{\partial(\alpha, \beta)} \right| = R_d \cos\beta$$

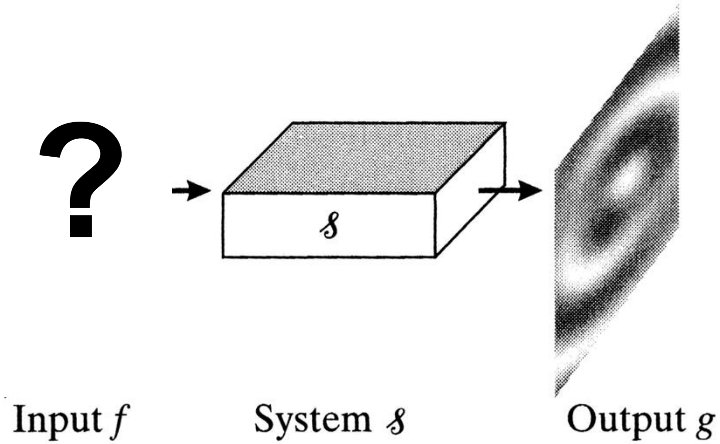
Chapter 2: Mathematical Preliminaries

Iterative Reconstruction Methods

Contents

- **Reconstruction and general linear inverse problems**
- **Statistical description of the data**
- **Image reconstruction criteria**
- **General structure of iterative reconstruction methods**
- **Maximum Likelihood Expectation Maximization (MLEM) algorithm**
- **Other related reconstruction methods**

General Inverse Problem and Image Reconstruction



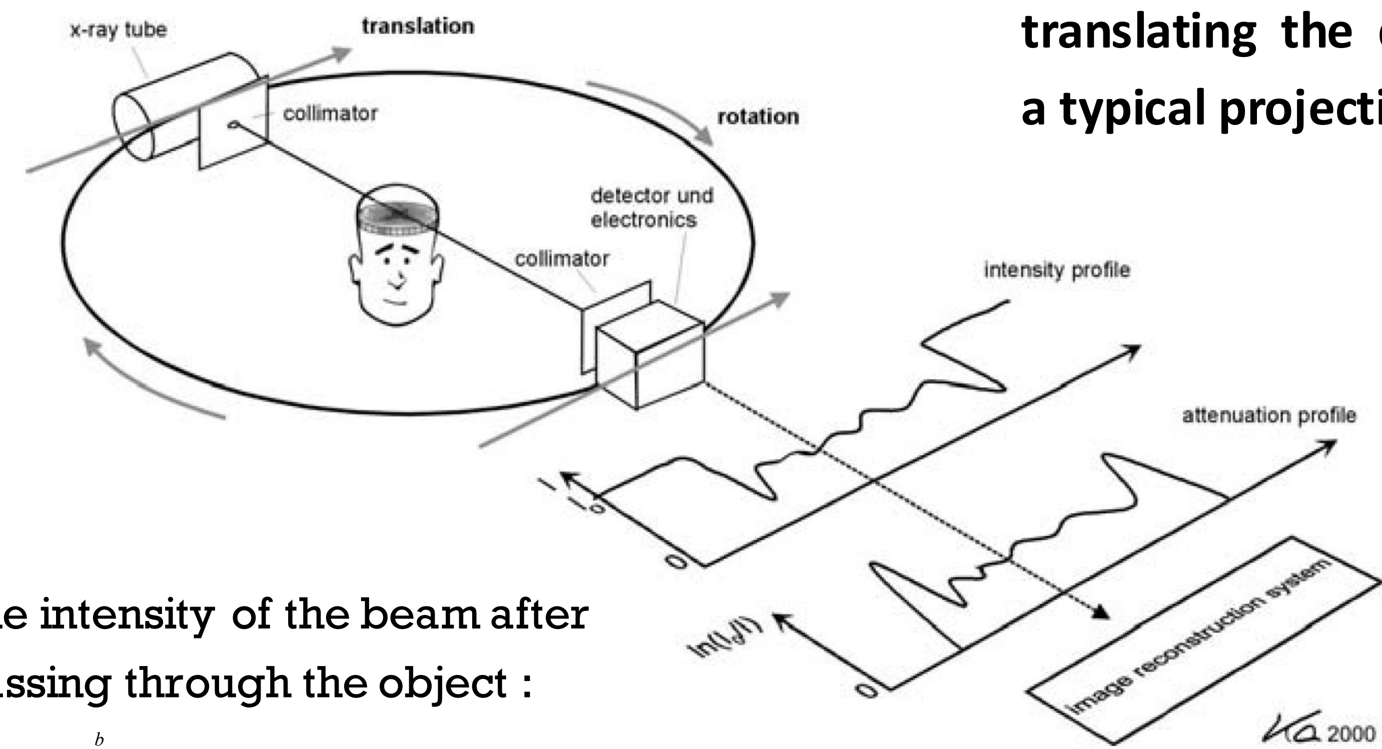
The inverse problem:

- Given a set of measured output signal
- Given the statistical description of the data
- Given a known system response function

what should be the input signal that gave rise to the output data?

Projection Data from Early X-ray CT Systems

Data measured by translating the detector is a typical projection data.

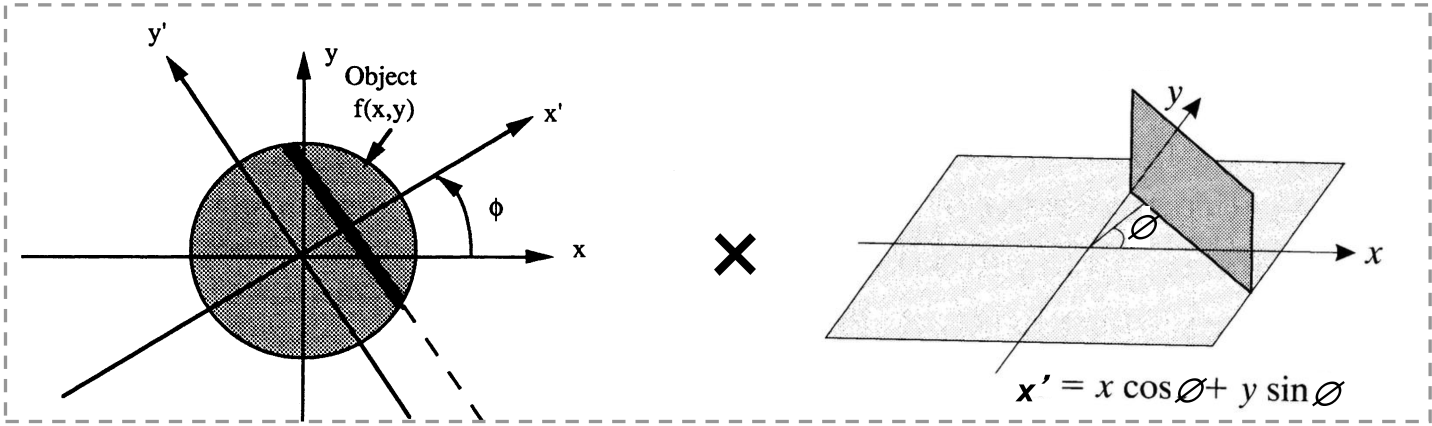


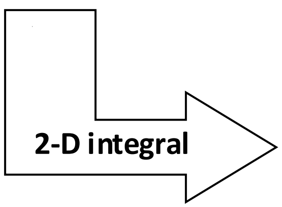
The intensity of the beam after passing through the object :

$$I = I_0 e^{-\int_a^b \mu(t) \cdot dt} \Leftrightarrow \ln(I_0 / I) = \int_a^b \mu(t) \cdot dt$$

From Computed Tomography, Kalender, 2000.

Radon Transform



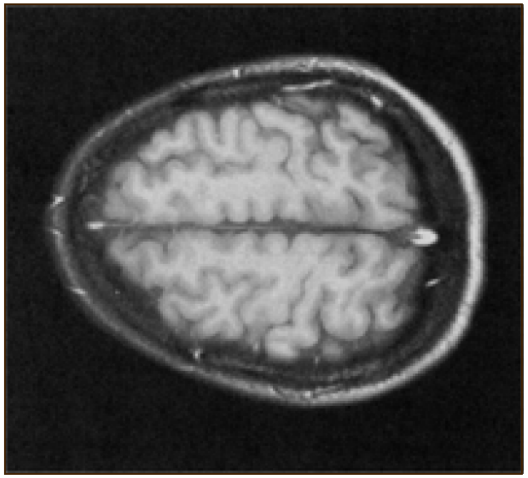


The value of the projection function $p_{\phi}(x')$ at this point is the integral of the function of f(x,y) along the straight line: $x' = x\cos\phi + y\sin\phi$

The integral of a line impulse function and a given 2-D signal gives the *projection* data from a given view …

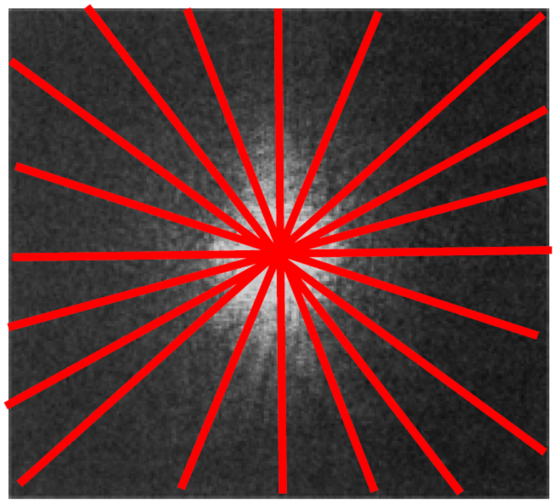
$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Central Slice Theorem



DFT image represents integration of original projections DFT transformed and summed together.

1-D DFTs at each projection



This is the fast way to create the DFT image from projection data. The more projections taken, the more complete the sampling.

http://engineering.dartmouth.edu/courses/engs167/12%20Image%20reconstruction.pdf

Simple Back-projection and the 1/r Blurring

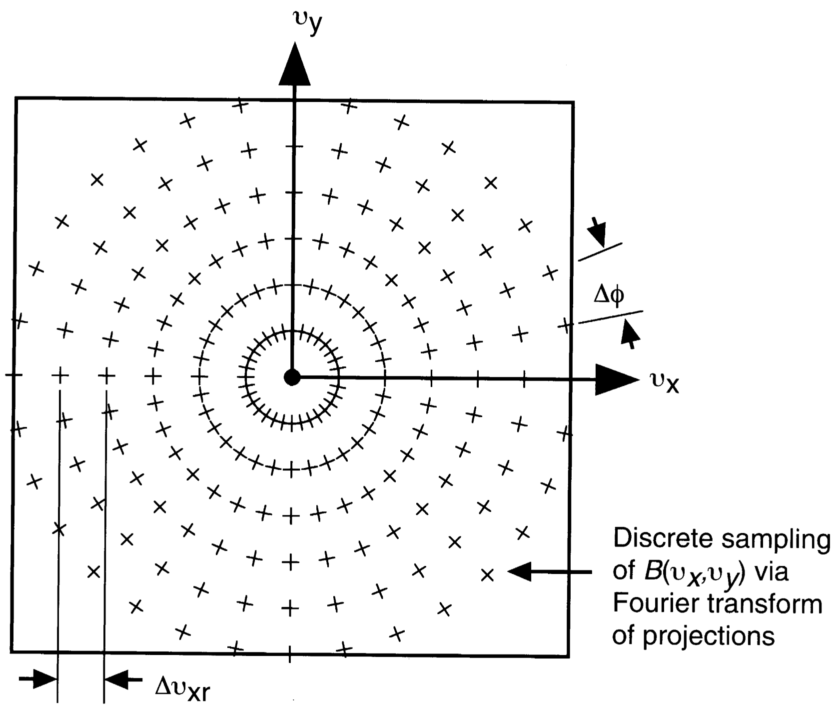
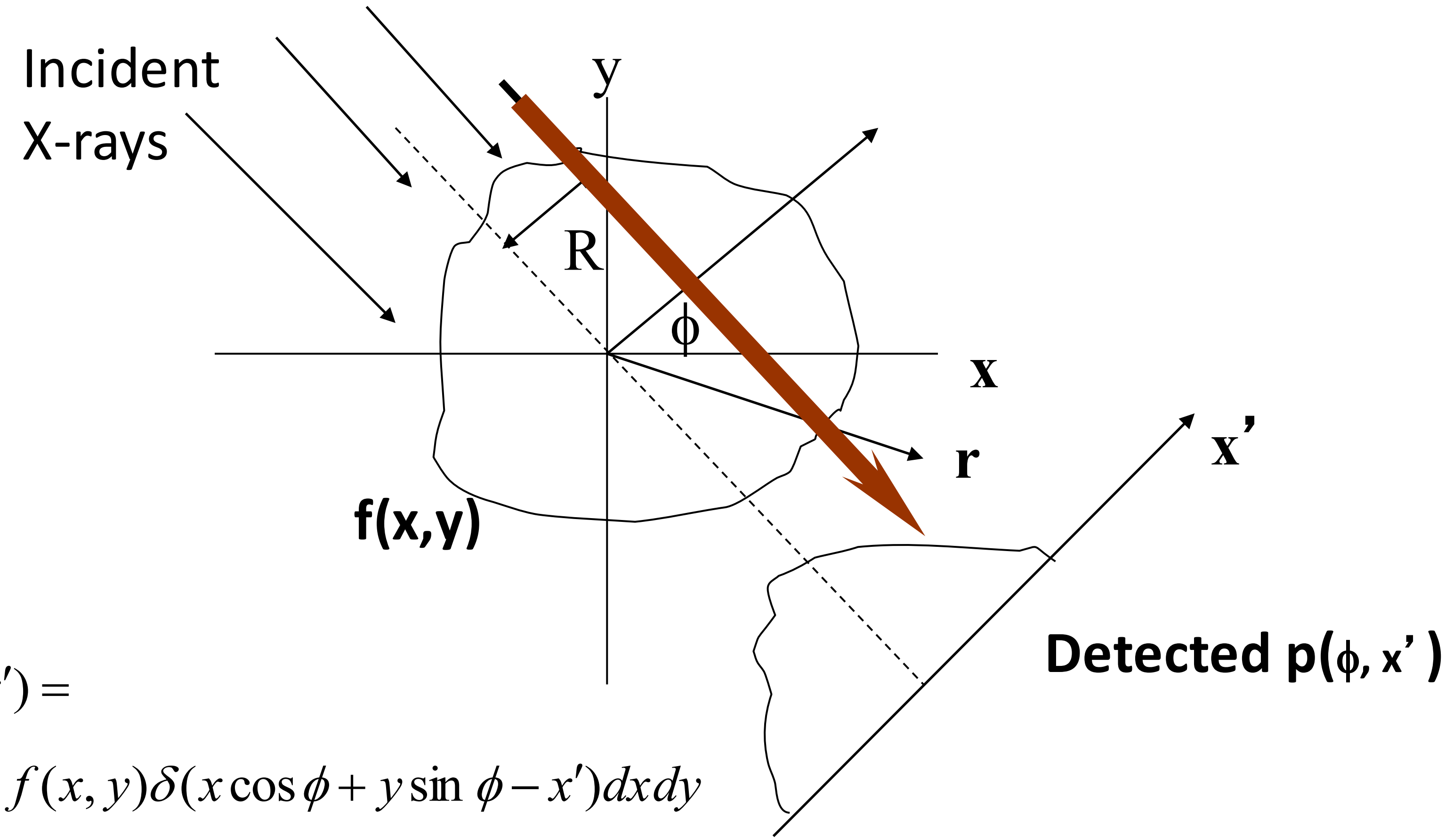


FIGURE 18 The discrete sampling pattern of $F(\upsilon_x,\upsilon_y)$ contained in $B(\upsilon_x,\upsilon_y)$, resulting from the use of discretely sampled projections.

The nature of the 1/r blurring:
Radon transform produced equally spaced radial sampling in Fourier domain.

Simple Back Projection from Projection Data

Projection data p(ϕ, x')



$$p(\phi, x') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\phi + y\sin\phi - x')dxdy$$

Simple Back Projection from Projection Data

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

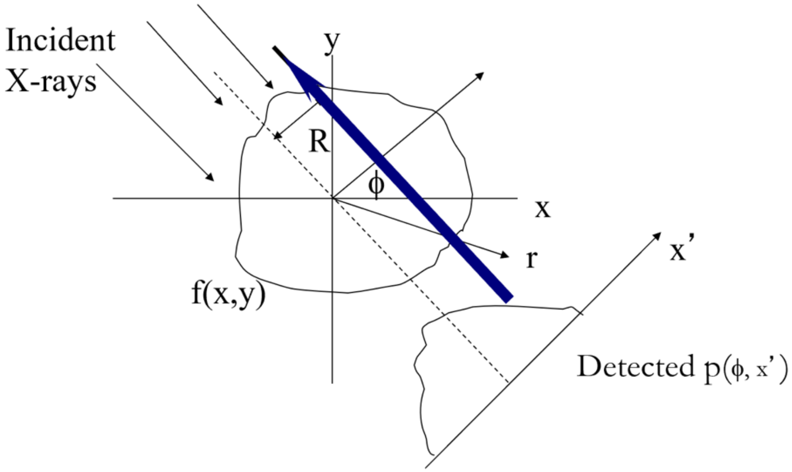
Result from a back projection from a single view angle:

$$b_\phi(x,y) = \int p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$

Adding up all the back projections from all the angles gives,

$$f_{simple\ back\text{-}projection}(x,y) = \int b_\phi(x,y) \, d\phi$$

$$f_{\text{simple back-projection}}(x,y) = \int_0^\pi d\phi \int_{-\infty}^{\infty} p_\phi(x') \, \delta(x \cos\phi + y \sin\phi - x') \, dx'$$



Simple Back-projection and the 1/r Blurring

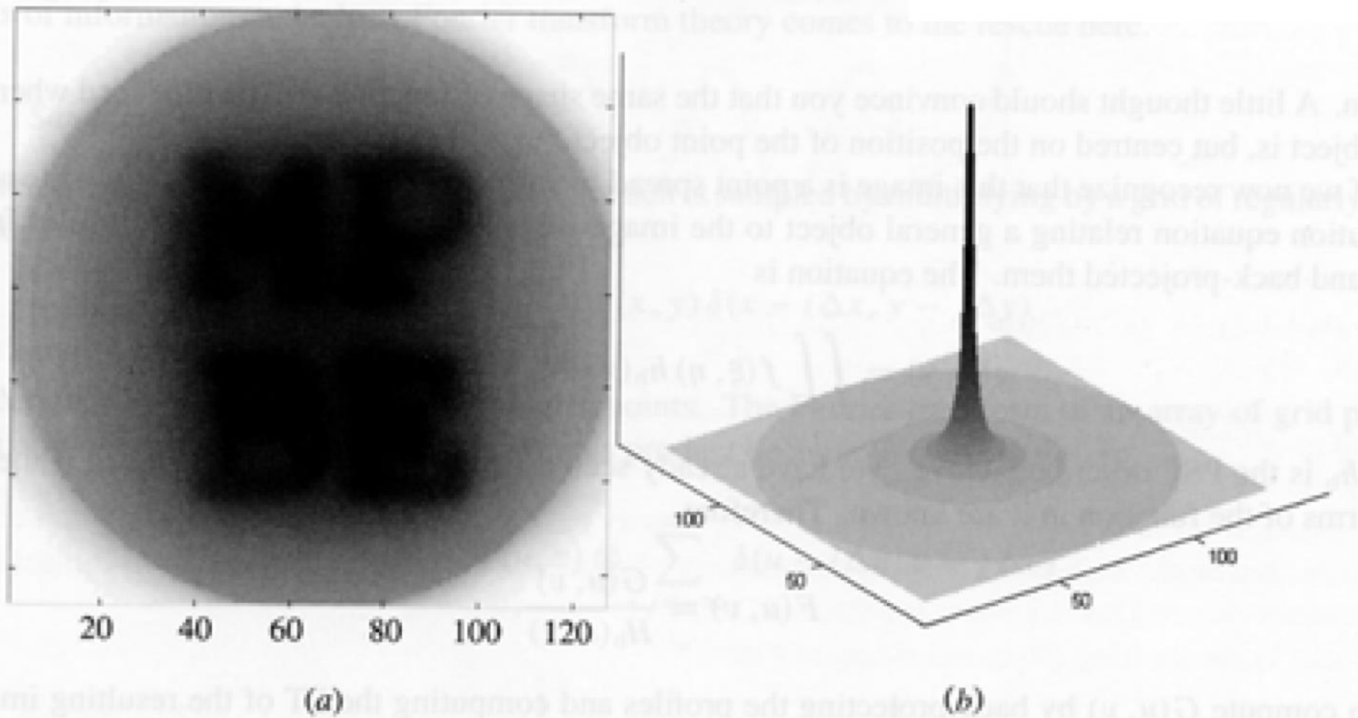


Figure 11.14. *(a) Image produced by back-projecting the sinogram given in figure 11.12(b); (b) the response to a point object in the middle of the field of view.*

From Medical Physics and Biomedical Engineering, Brown, IoP Publishing

Simple Backprojection and Inverse Radon Transform

Crude Idea 1: Take each projection and smear it back along the lines of integration it was calculated over.

Result from a back projection from a single view angle:

$$b_{\phi}(x,y) = \int_{-\infty}^{\infty} p_{\phi}(x') \cdot \delta(xcos\phi + ysin\phi - x')dx'$$

Adding up all the back projections from all the angles gives,

$$\hat{f}_{\text{simple backprojection}}(x,y) = \int_{0}^{\pi} d\phi \int_{-\infty}^{\infty} p_{\phi}(x') \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

$$\hat{f}_{\text{Inverse Radon transform}}(x,y) = \int_{0}^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\{F[p_{\phi}(x')] \cdot |w|\} \cdot \delta(x\cos\phi + y\sin\phi - x')dx'$$

Review of Fourier Transform and Filtering Spectral Filtering

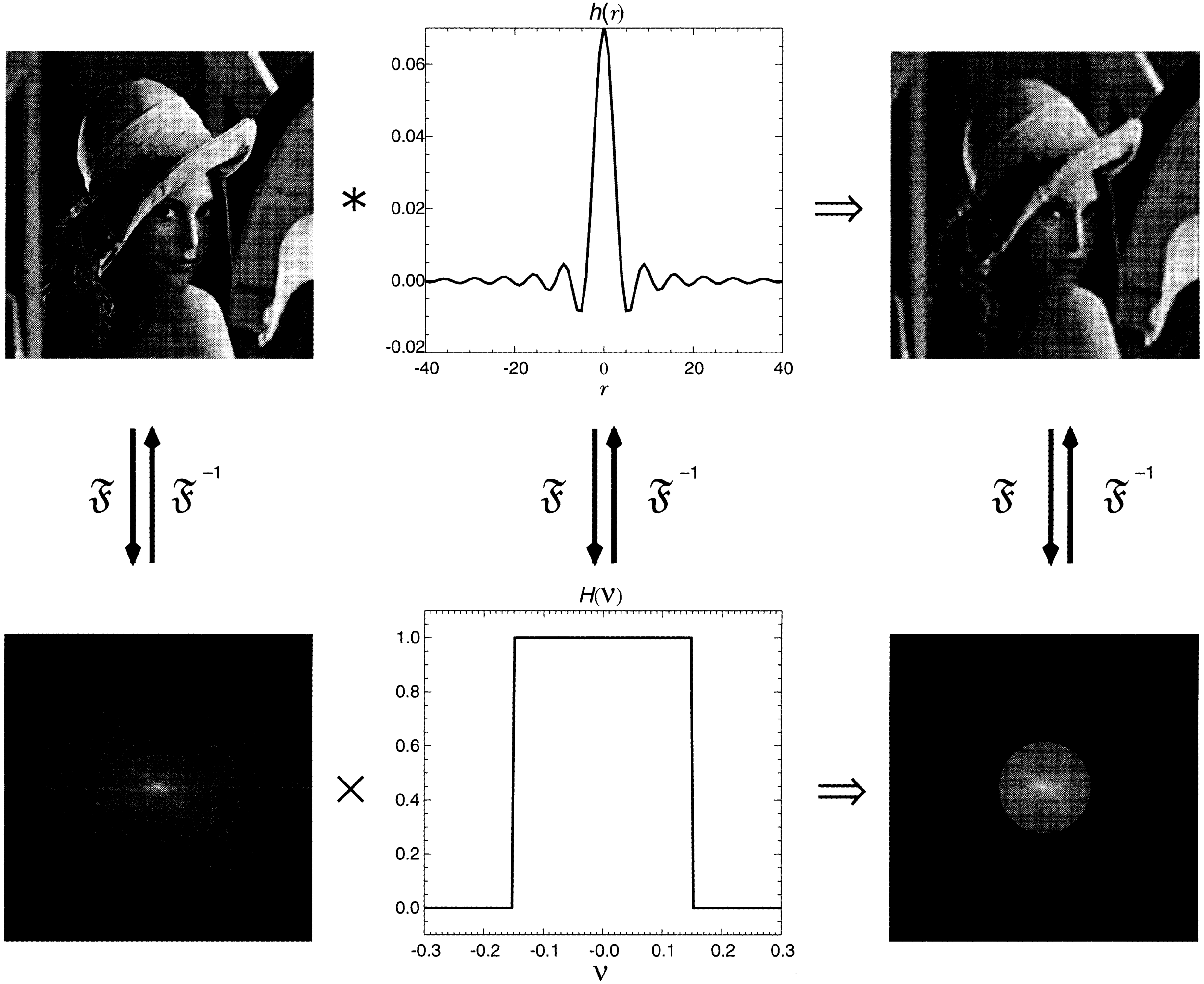


FIGURE 3 Ideal lowpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column are the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2 + y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Review of Fourier Transform and Filtering
Spectral Filtering

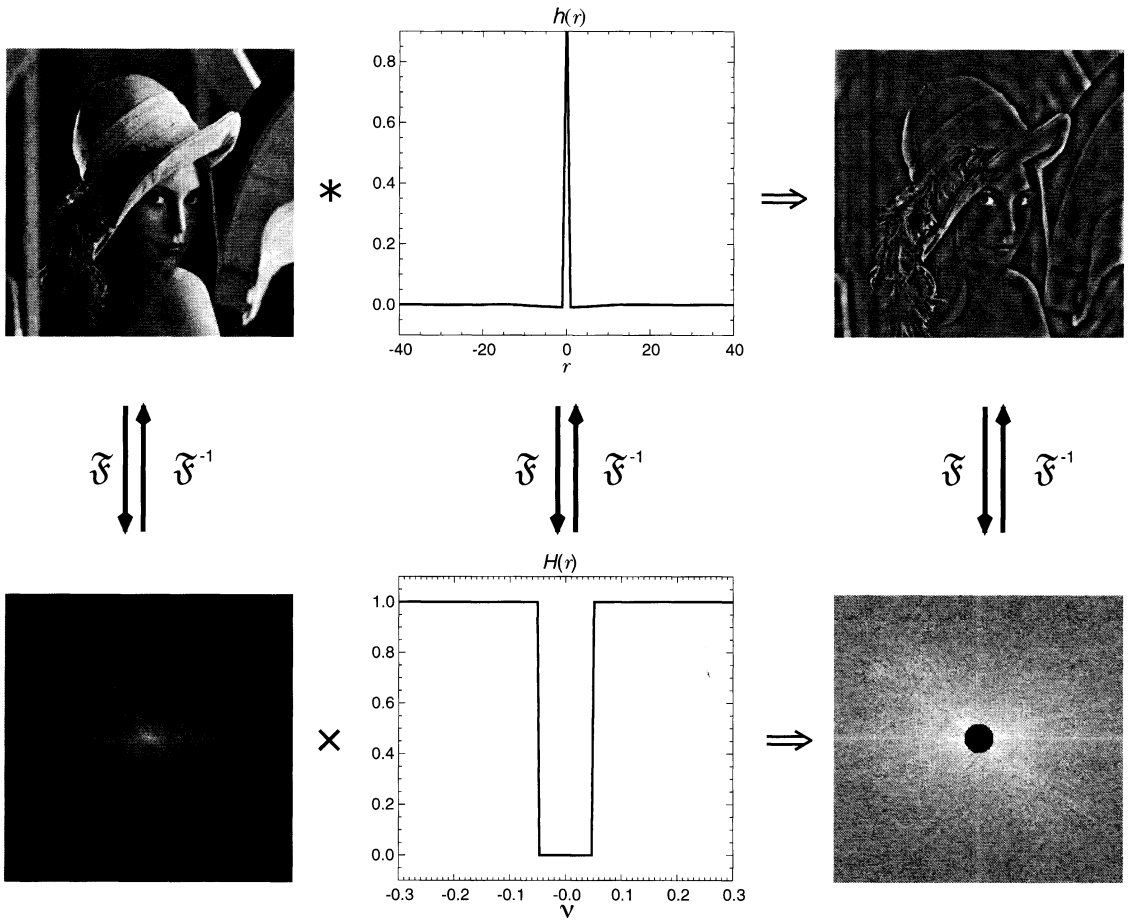


FIGURE 4 Ideal highpass filtering of an image. Figures in the top and bottom rows demonstrate the filtering operation in the spatial and frequency domains, respectively. The left column shows the input image and the magnitude of its Fourier transform. The middle column shows the circular symmetric PSF and its transfer function, where $\nu = \sqrt{\nu_x^2 + \nu_y^2}$ and $r = \sqrt{x^2+y^2}$. Images in the right column are the output image and the magnitude of its Fourier transform.

Filtered Back-projection

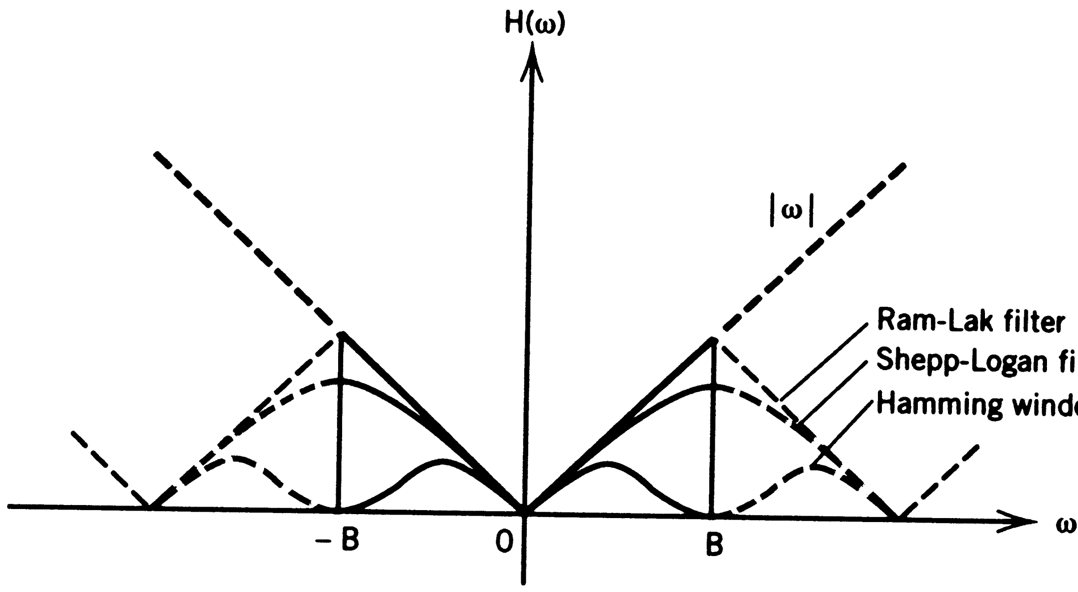


Figure 3-4 (a) Examples of the band-limited filter function of sampled data. Note the cyclic repetitiveness of the digital filter.

Filtered Back-Projection

$$\Delta x = 1/(2B),$$

Sampled version

$$h_{RL}(0) = B^2 = \frac{1}{4\,\Delta x^2} \qquad (\text{if } k = 0)$$

$$h_{RL}(k) = 0 \qquad (\text{if } k \text{ even})$$

$$h_{RL}(k) = \frac{-4B^2}{\pi^2 k^2} = \frac{-1}{\pi^2 k^2\,\Delta x^2} \qquad (\text{if } k \text{ odd})$$

$$h_{SL}(k) = \frac{-2}{\pi^2\,\Delta x^2 (4k^2 - 1)} = \frac{-8B^2}{\pi^2(4k^2 - 1)}$$

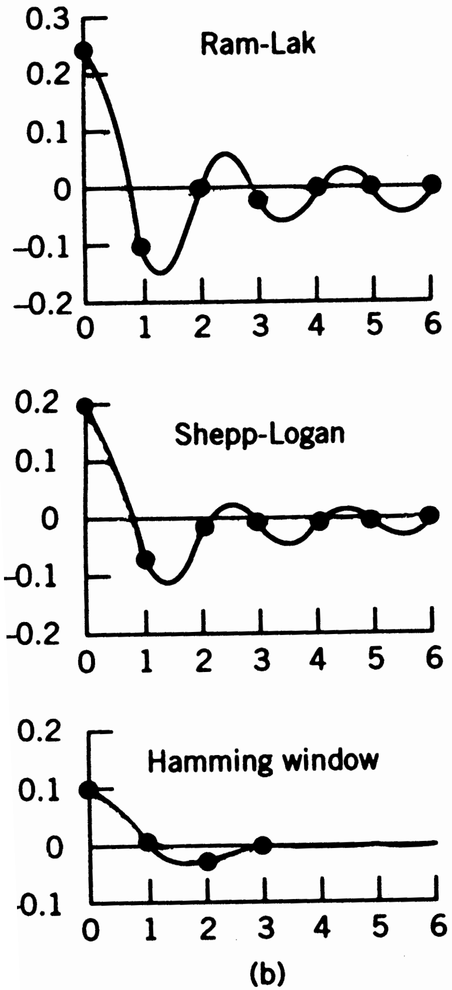


Figure 3-4 (b) Spatial domain filter kernals corresponding to the filter functions shown in the Ram-Lak filter is a high-pass filter with a sharp response but results in some noise enhancement, while the Shepp-Logan and the Hamming window filters are noise-smoothed filters and therefore have better SNR.

Filtered Backprojection (FBP) Algorithm (revisited)

The Filtered-backprojection (FBP) operator:

$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x, y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} F^{-1}\left\{F\left[p_{\phi}(x')\right] \cdot H(\omega)\right\} \cdot \delta(x\cos\phi + y\sin\phi - x')\,dx',$$

where $H(\omega) = |\omega| \cdot H'(\omega)$.

The Filtered-backprojection (FBP) operator can also be expressed within the spatial domain as

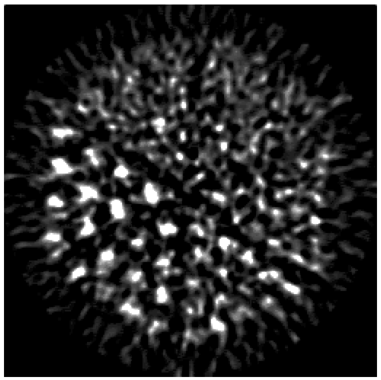
$$\hat{\boldsymbol{f}}_{\text{Filtered backprojection}}(x, y) = \int_0^{\pi} d\phi \int_{-\infty}^{\infty} p_{\phi}(x') \cdot h(x\cos\phi + y\sin\phi - x')\,dx',$$

where $h(x\cos\phi + y\sin\phi - x') = F^{-1}[H(\omega)]$.

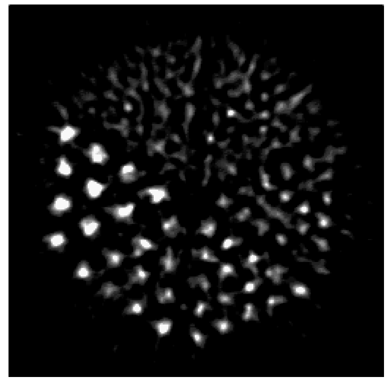
Chapter 2: Mathematical Preliminaries

Iterative Reconstruction Methods

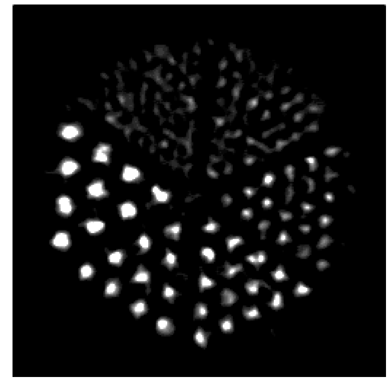
The Importance of Counts



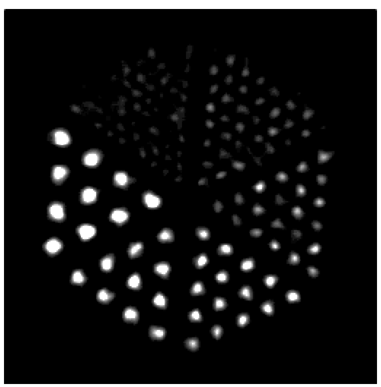
50 000 counts



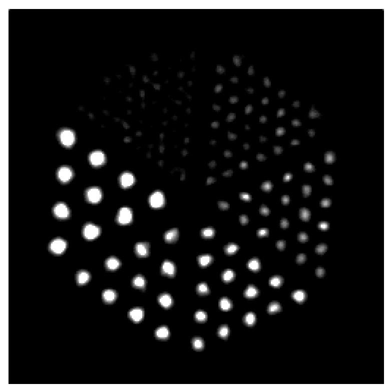
100 000 counts



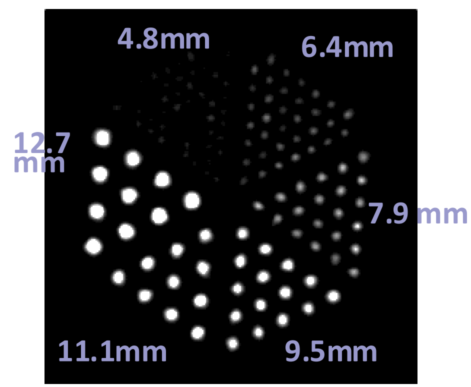
200 000 counts



500 000 counts



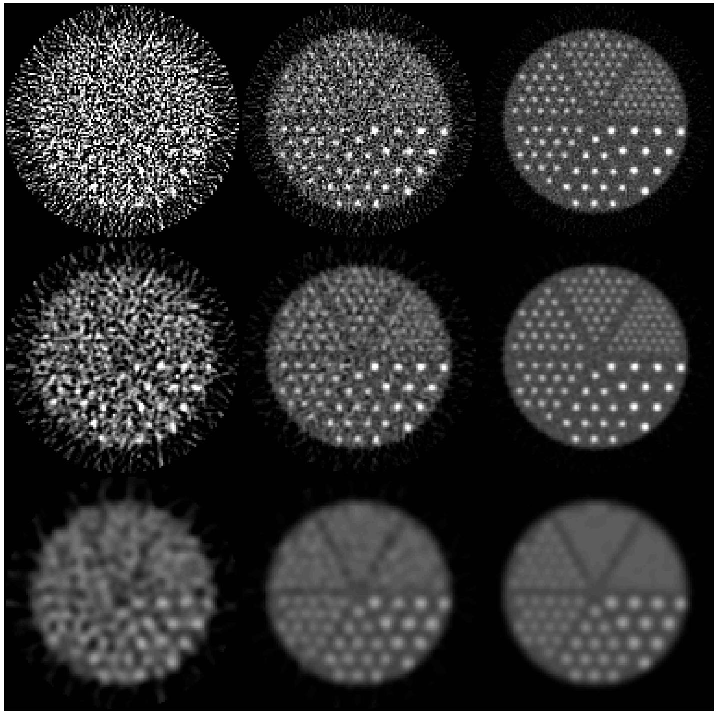
1 million counts



2 million counts

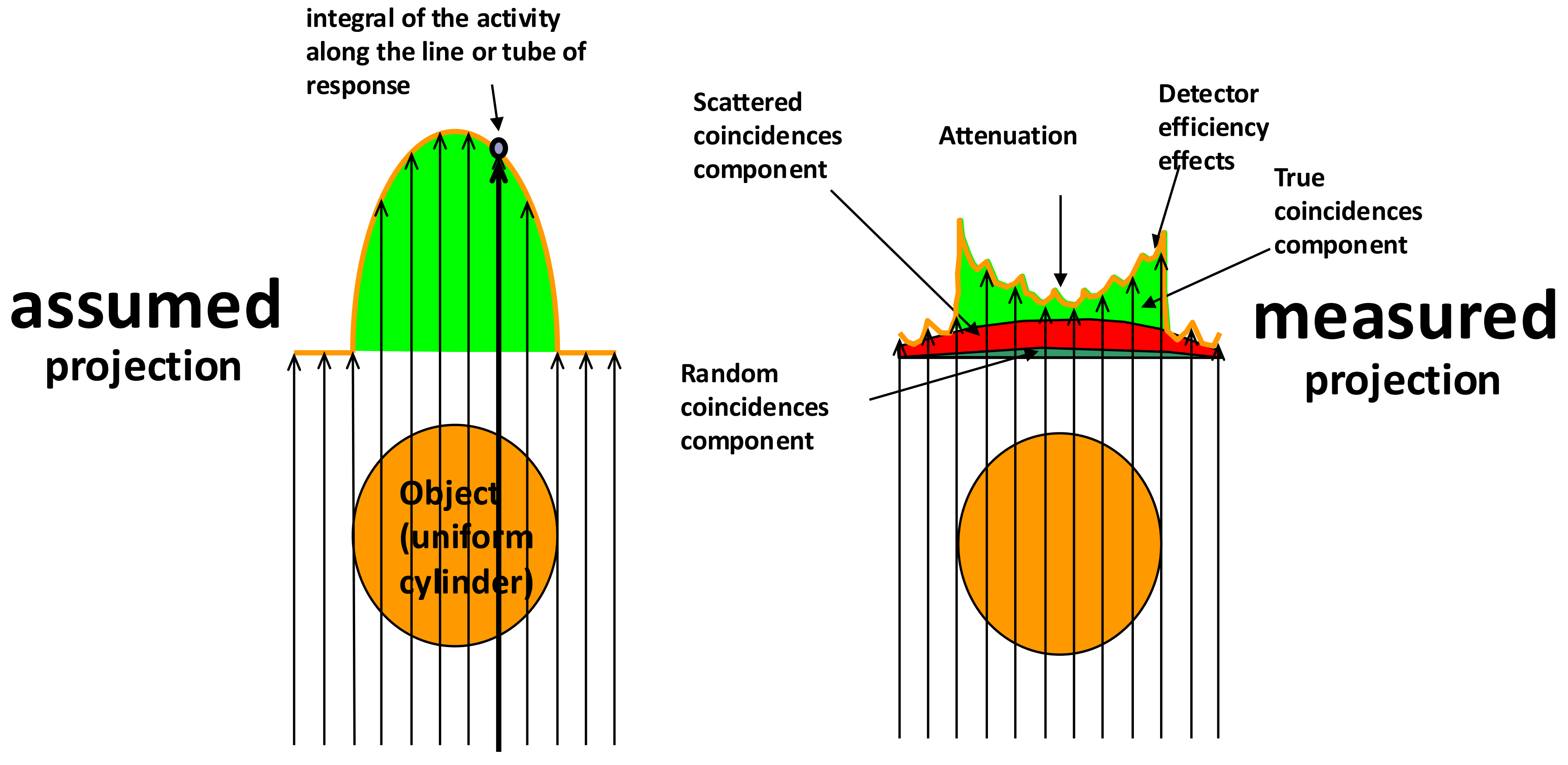
Noise In PET Images

- Noise in PET images is dominated by the counting statistics of the coincidence events detected.
- Noise can be reduced at the cost of image resolution by using an apodizing window on ramp filter in image reconstruction (FBP algorithm).



An Example of PET Image Reconstruction

- Data corrections are necessary
  - the measured projections are not the same as the projections assumed during image reconstruction



Analytical Methods

- Advantages
  - Fast
  - Simple
  - Predictable, linear behaviour
- Disadvantages
  - Not very flexible
  - Image formation process is not modelled $\Rightarrow$ image properties are sub-optimal (noise, resolution)

Statistical Methods

- Advantages
  - Can accurately model the image formation process (use with non-standard geometries, e.g. not all angles measured, gaps)
  - Allow use of constraints and *a priori* information (non-negativity, boundaries)
  - Corrections can be included in the reconstruction process (attenuation, scatter, etc)

- Disadvantages
  - Slow
  - Less predictive behaviour (noise? convergence?)

Image Reconstruction and General Linear Inverse Problem

A basic problem for image reconstruction in emission tomography

Find the object distribution **f**, given (1) a set of projection measurements **g**, (2) information (in the form of a matrix **H**) about the imaging system that produced the measurements, and, possibly, (3) a statistical description of the data and (4) a statistical description of the object (Fig. 1).

Image Reconstruction and General Linear Inverse Problem

An Emission Tomography System

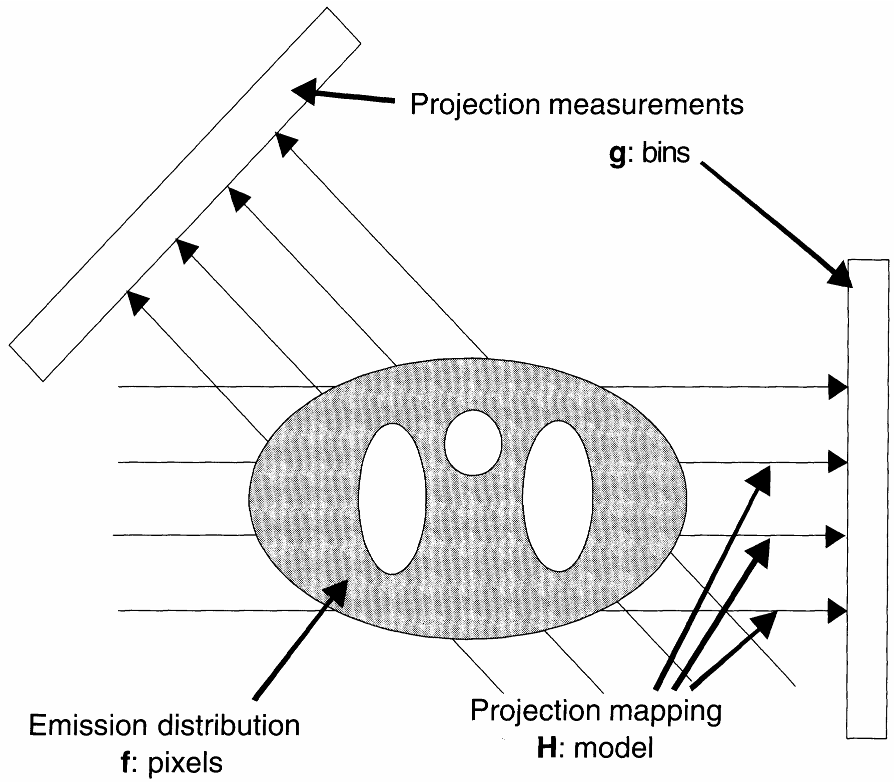
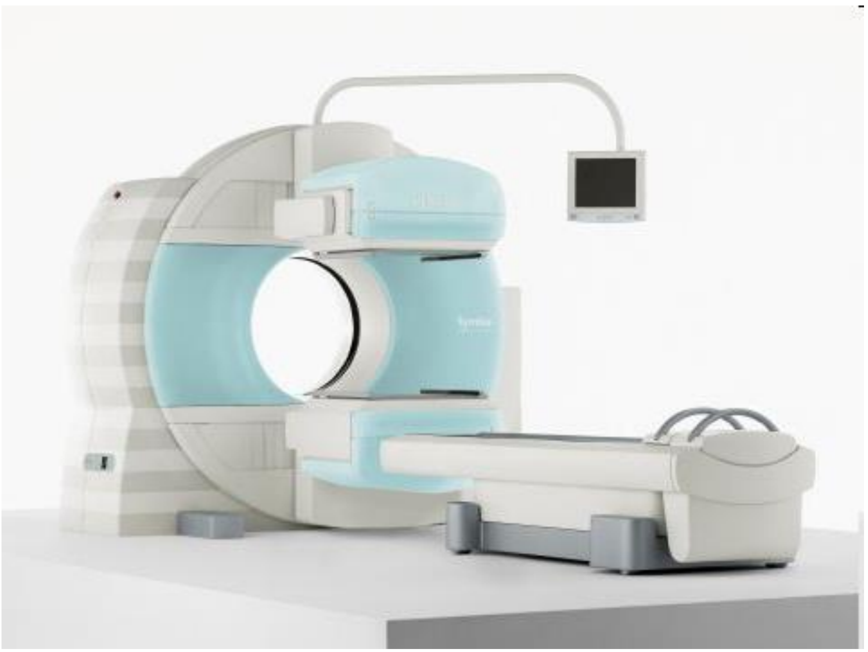


FIGURE 1 A general model of tomographic projection in which the measurements are given by weighted integrals of the emitting object distribution.

The Forward and Inverse Problem?

Single Photon Emission Computed Tomography (SPECT)

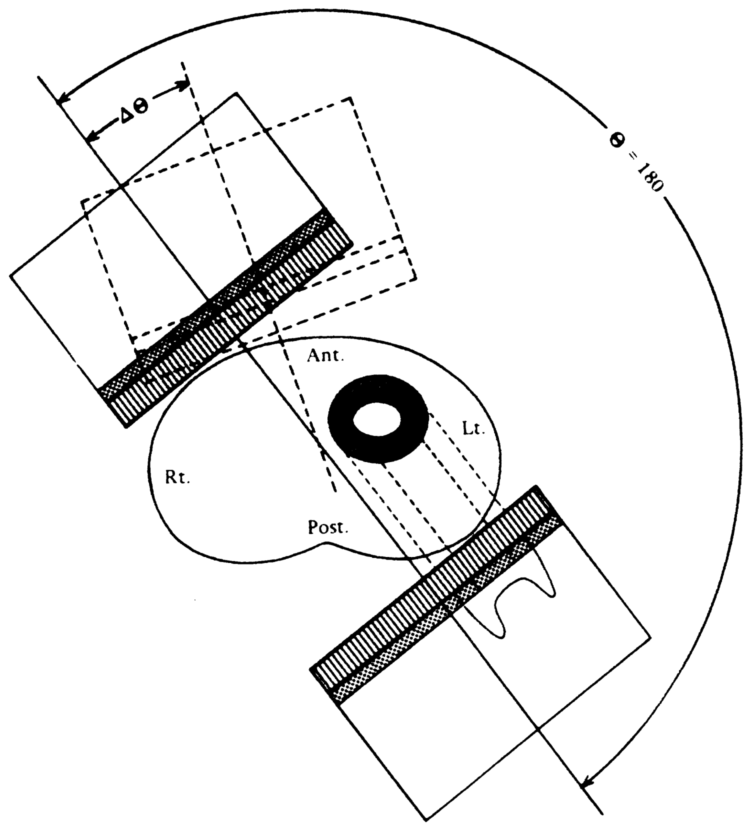


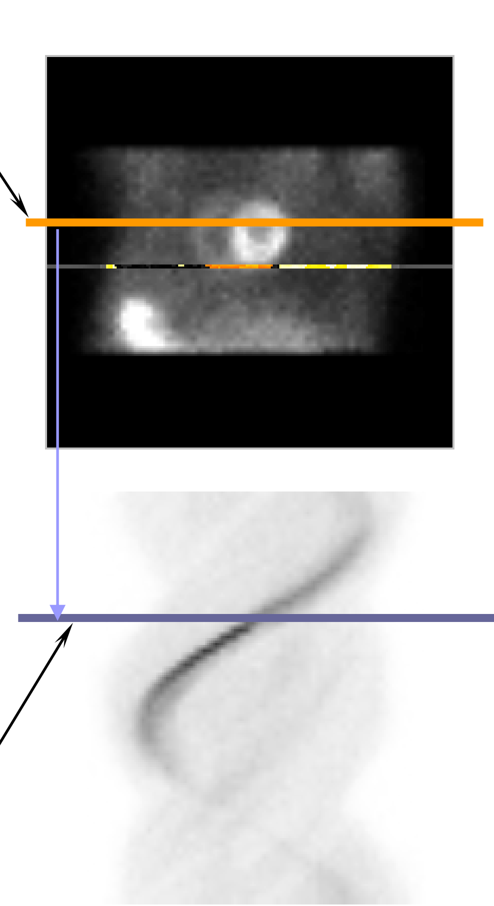
Siemens Symbia SPECT/CT

Philips Precedence SPECT/CT

Basic SPECT – Projection Data

Axial level of sinogram



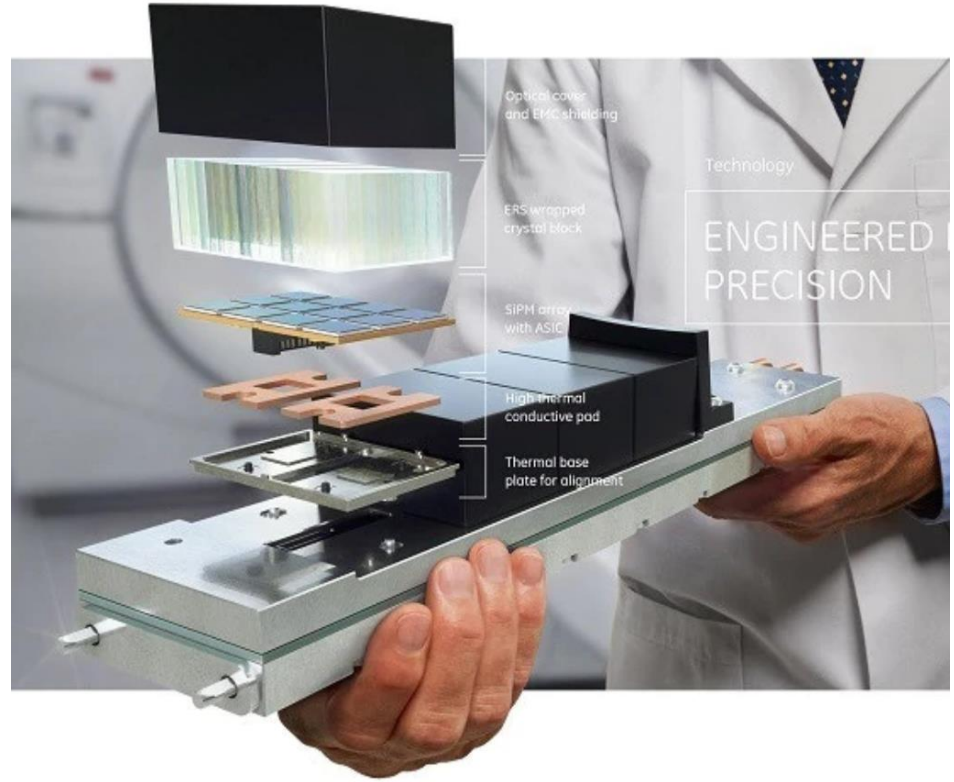


2D planar projection

Sinogram

Angle of above projection

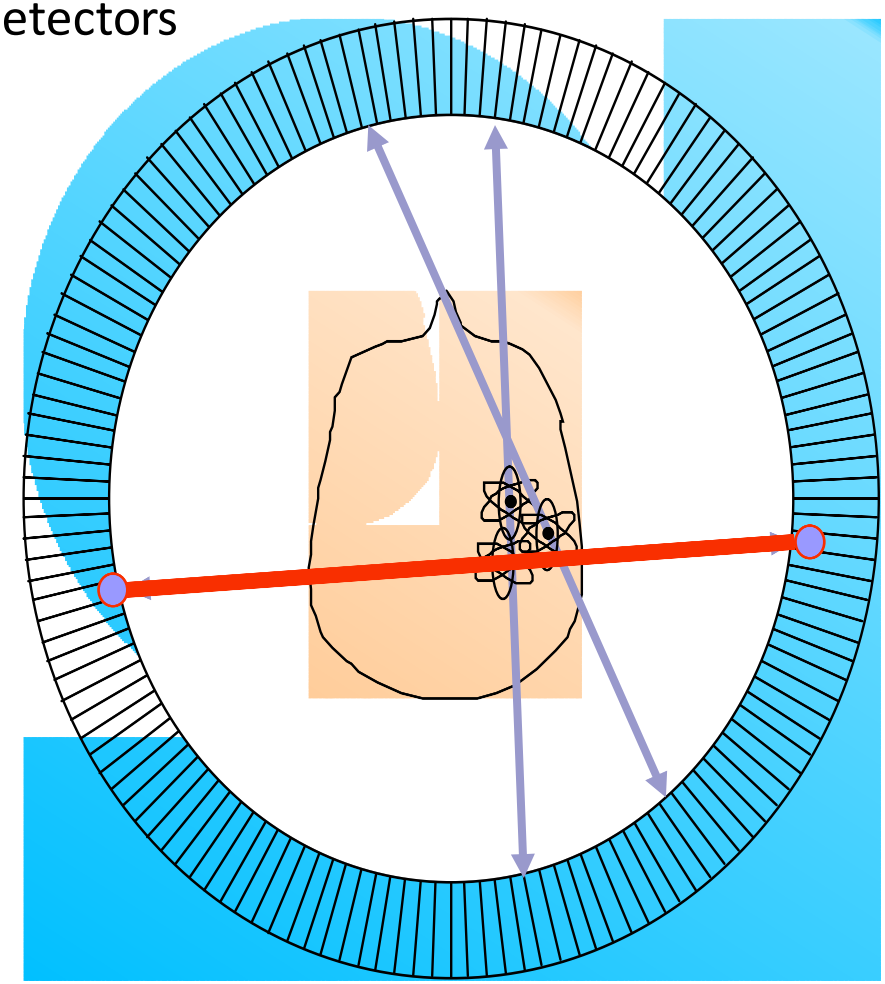
Modern PET Cameras



GE Discovery II PET/CT Scanner

Detect Radioactive Decays

Ring of Photon Detectors

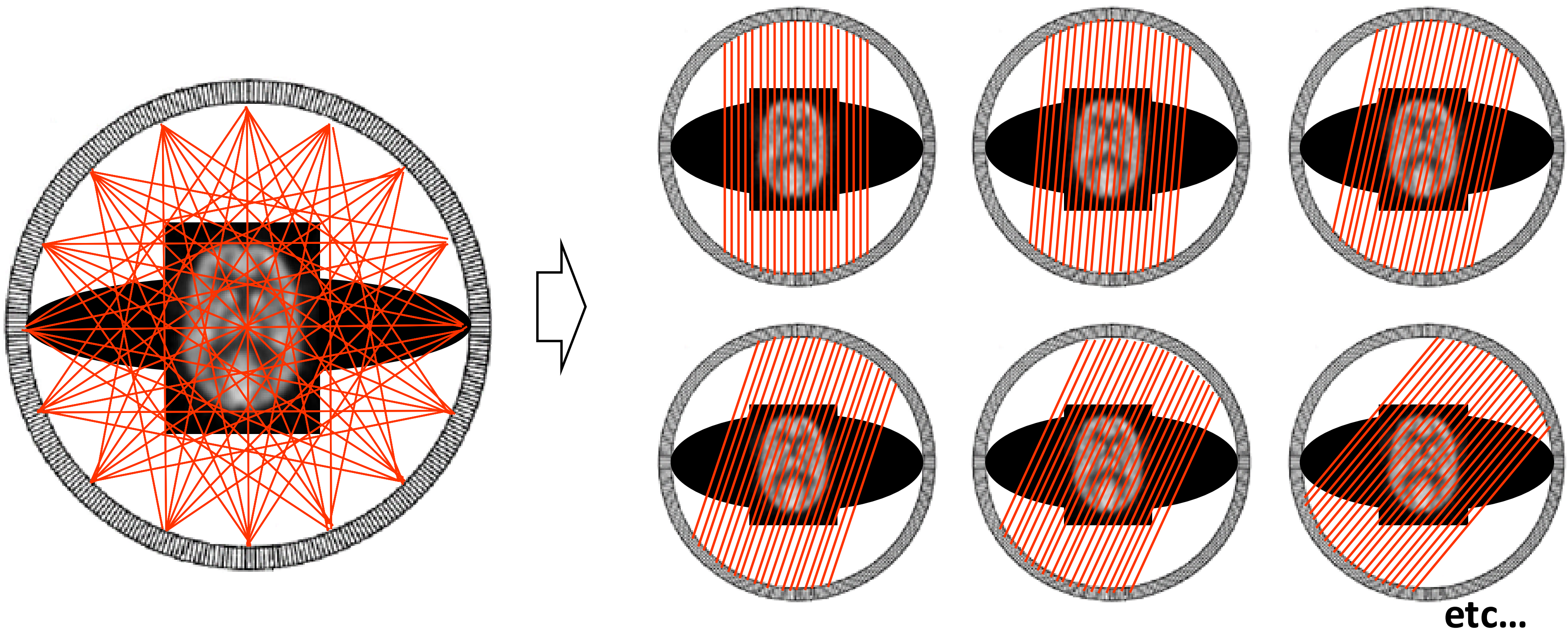


- Radionuclide decays, emitting β^{+}.
- β^{+} annihilates with e^{-} from tissue, forming back-to-back 511 keV photon pair.
- 511 keV photon pairs detected via time coincidence.
- Positron lies on line defined by detector pair (known as a *chord* or a *line of response* or a *LOR*).

Detect Pair of Back-to Back 511 keV Photons

PET Data Acquisition

- Organization of data
  - True counts in LORs are accumulated
  - In some cases, groups of nearby LORs are grouped into one average LOR ("mashing")
  - LORs are organized into projections



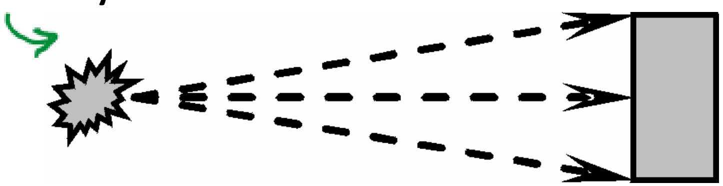
The Basic Idea of Statistical Reconstruction

Consider the simplest imaging problem as illustrated below:

N: The number of photons emitted by the source

k: The actual number of detected photons

μ: Mean or expected number of detected photons:



p: the probability for a photon emitted by the source to be detected by the detector

During an imaging study, we observed k counts, and we know that the probability for a photon emitted by the source to be detected by the detector is p, then

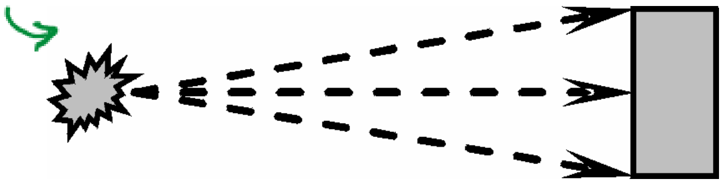
how do we estimate the number of photons being emitted by the source during the imaging period?

The Basic Idea of Statistical Reconstruction

N: **The number of photons emitted by the source**

k: **The actual number of detected photons**

μ: **Mean or expected number of detected photons:**



p: **the probability for a photon emitted by the source to be detected by the detector**

We can estimate the underlying parameter N by the Maximum-Likelihood Estimation (MLE) strategy:

$$\hat{N}_{Maximum\ Likelihood\ estimation} = \arg\max_{N}\{P[k|N]\}$$

$P[k(N)]$ is call the Likelihood function: the probability of detecting k counts given there were a total of N gamma rays emitted by the source.

The Basic Idea of Statistical Reconstruction

The likelihood function:

In very non-technical terms, the conditional probability,

$$\boldsymbol{L}(\boldsymbol{f}, \boldsymbol{g}) \equiv p(\boldsymbol{g}|\boldsymbol{f}),$$

of the underlying source function ($\boldsymbol{f}$) leading to the measurement ($\boldsymbol{g}$).

The basic idea of Maximum-Likelihood reconstruction is simple:
We choose the solution (estimated source function) that maximizes the likelihood function, or the solution that is the most likely to produce the measured data.

$$\hat{\boldsymbol{f}}_{Maximum\ Likelihood} = \arg\max_{\boldsymbol{f}} \{P[\boldsymbol{g}|\boldsymbol{f}]\}$$

Source of Noise

Source of noise:

Measurement error, instrument malfunction …

Random fluctuation in the number of counts detected by detectors

Other additive noise

In emission tomography systems, *detector readout noise* is generally much smaller than the statistical fluctuation associated with the counting process.

Bernoulli Process

Consider the radioactive disintegration process in a sample, it follows the following four conditions:

- It consists of N trails.
- Each trail has a binary outcome: success or failure (decay or not).
- The probability of success (decay) is a constant from trail to trail – all atoms have equal probability to decay.
- The trails are independent.

In statistics, these four conditions characterize a Bernoulli process.

Binomial Distribution

The binomial distribution gives the discrete probability distribution of obtaining exactly k successes out of N Bernoulli trials (where the result of each Bernoulli trial is true with probability *p* and false with probability *1-p*). The binomial distribution is therefore given by

$$p(k|N,p)=\binom{N}{k}p^k(1-p)^{N-k}$$

$$where$$

$$\binom{N}{k}=\frac{N!}{k!(N-k)!}$$

Binomial Distribution and Imaging Applications

Suppose we have *a point source* generated N gamma rays and we have a pixilated detector for detecting these photons

Each gamma ray photon has *a fixed probability p* of reaching a given detector element. *p* is defined by the relative distance between the source and the pixel and the collimation configuration used.

So the *probability of detecting k gamma rays* on the pixel is given as

$$p(k|N,p) = \binom{N}{k} p^k (1-p)^{N-k}$$

Furthermore, *the number of counts detected on different pixels are independent* …

Binomial Distribution

For a binomial distribution, the mean or the expectation of the number of disintegration in time t is given by

$$\mu \equiv \sum_{n=0}^{N} n \cdot P_n = \sum_{n=0}^{N} n \cdot \binom{N}{n} p^n q^{N-n} = Np$$

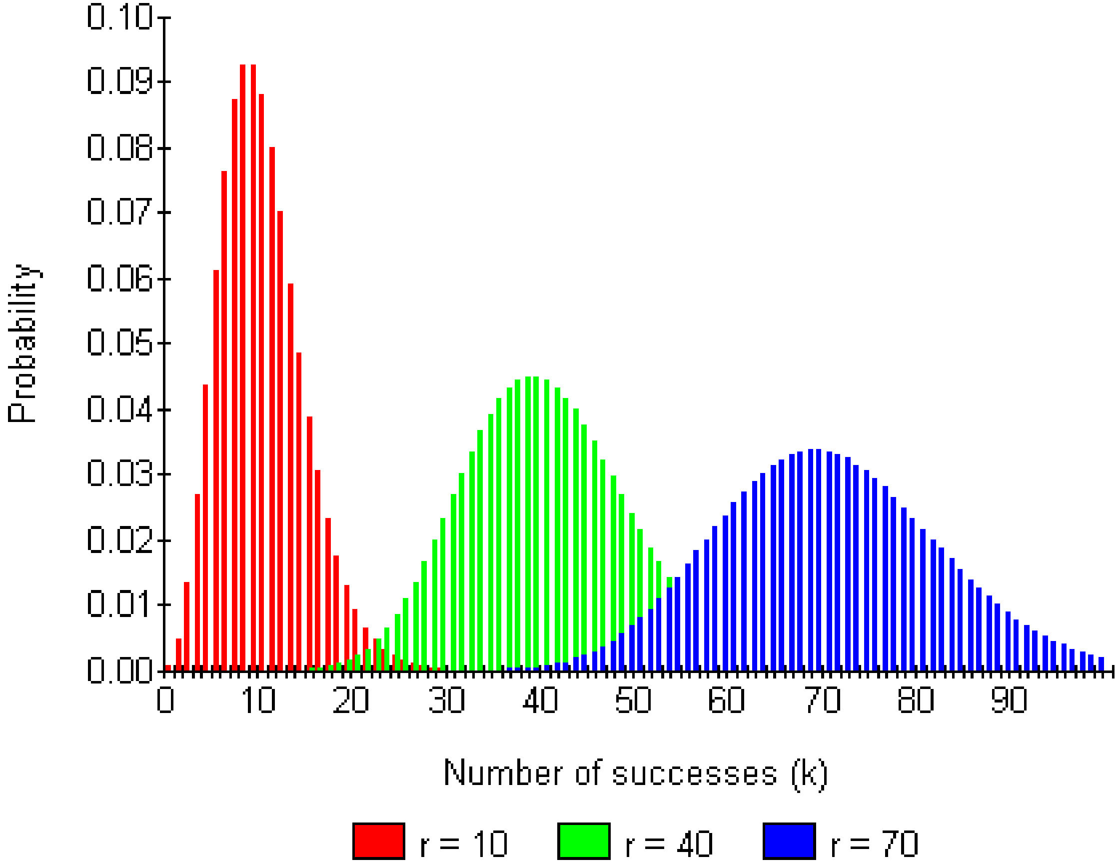
and the fluctuation on the number of disintegrations is given by the variance or the standard deviation of the

$$\sigma^2 \equiv \sum_{n=0}^{N} (n-\mu)^2 \cdot P_n = Npq$$

$$and$$

$$\sigma \equiv \sqrt{\sum_{n=0}^{N} (n-\mu)^2 \cdot P_n} = \sqrt{Npq}$$

Binomial Distribution



☞ **What are the mean and standard deviation of a Binomial distribution?**

Poisson Distribution

The counting statistics related to nuclear decay processes is often more conveniently described by the Poisson distribution, is related to situations that involves a collection of multiple trails that satisfy the following conditions:

1. The number of trials, N, is very large, e.g., $N > 1$.
2. Each trail is independent.
3. The probability that each single trial is successful is a constant and approaches zero, $p << 1$. Therefore, the number of successful trials fluctuates around a finite number.

Binomial and Poisson Random Variable

When $N \to \infty$ and $p \to 0$, k follows the Poisson distribution, whose probability function is

Binomial distribution

$$\boxed{p(k|N,p) = \binom{N}{k} p^k (1-p)^{N-k}} \Rightarrow \boxed{p(k) = \frac{\mu^k}{k!} e^{-\mu}}$$

Poisson distribution

where μ is the expected value of the Poisson variable defined

$$\mu = \lim_{N \to \infty} N\,p$$

Poisson Distribution

The probability of having *n* successful trails can be approximated with the Poisson distribution.

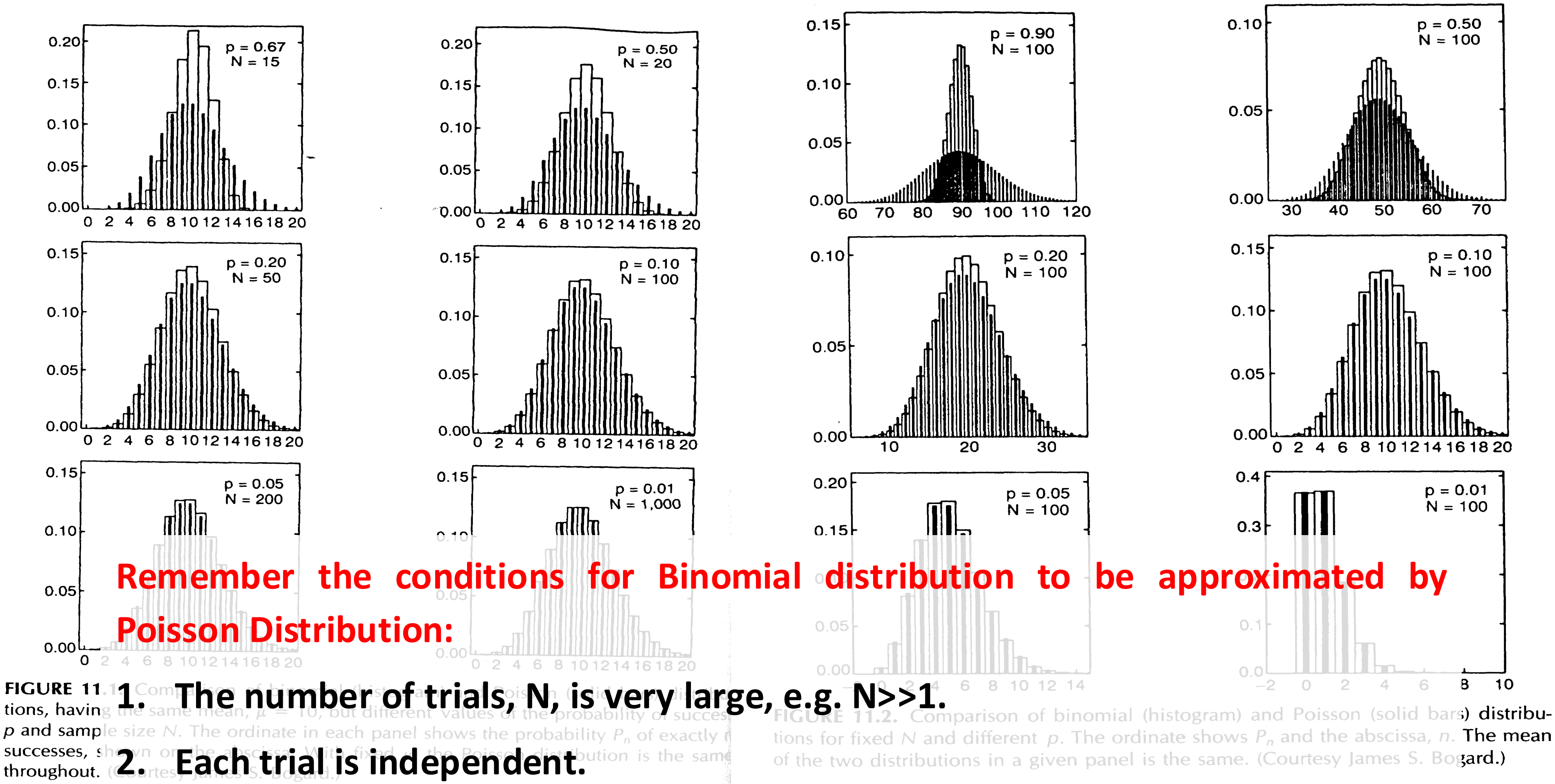
$$P(n \mid \mu) = \frac{\mu^n}{n!} e^{-\mu}$$

and the mean and the variance of number of successful trail are given by

$$Mean(n) = \mu = N \cdot p$$

$$Std(n) \equiv \sigma = \sqrt{\mu} = \sqrt{Np}$$

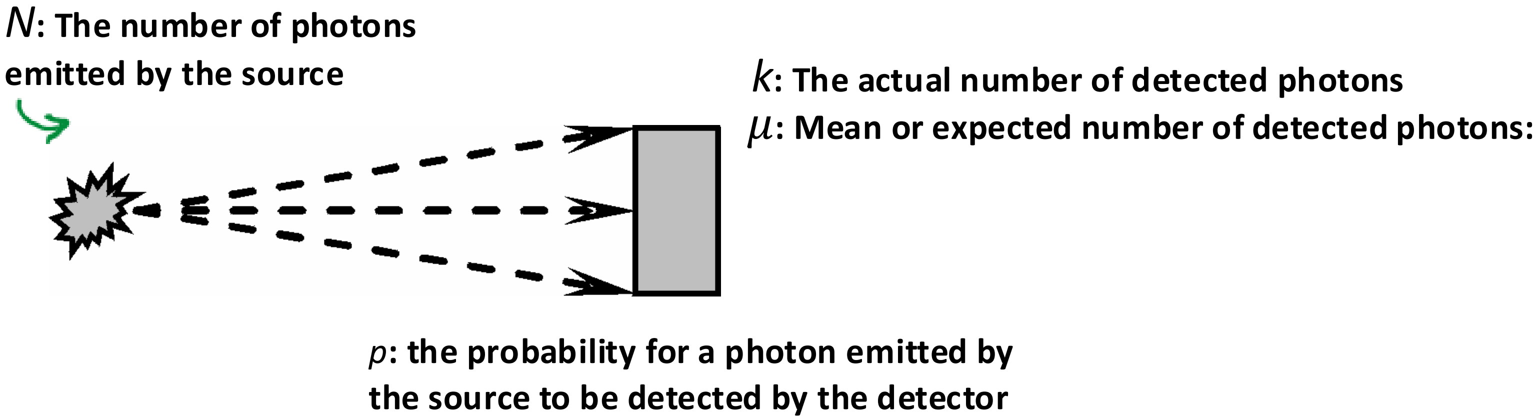
Poisson Distribution



Remember the conditions for Binomial distribution to be approximated by Poisson Distribution:

1. **The number of trials, N, is very large, e.g. N>>1.**
2. **Each trial is independent.**
3. **The probability that each single trial is successful is a constant and approaching zero, p<<1. So the number of successful trials is fluctuating around a finite number.**

The Basic Idea of Statistical Reconstruction



We can estimate the underlying parameter N by the Maximum-Likelihood Estimation (MLE) strategy:

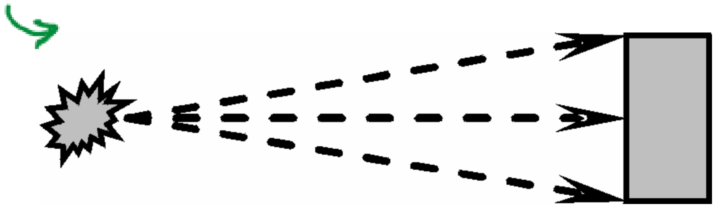
$$\hat{N}_{Maximum\ Likelihood\ estimation} = \arg\max_N \{P[k|N]\}$$

$P[k(N)]$ is call the Likelihood function: the probability of detecting k counts given there were a total of N gamma rays emitted by the source.

N: The number of photons emitted by the source

k: The actual number of detected photons

μ: Mean or expected number of detected photons:



p: the probability for a photon emitted by the source to be detected by the detector

Since

$$P[k(N)] = \frac{(N \cdot p)^k}{k!} e^{-N \cdot p}$$

Then

$$\widehat{N}_{Maximum\ likelihood} = \arg\max_N \{P[k|N]\} = \arg\max_N \left\{ \frac{(N \cdot p)^k}{k!} e^{-N \cdot p} \right\},$$

alternatively

$$\widehat{N}_{ML} = \arg\max_N \{\log[P[k|N]]\}$$

$$= \arg\max_N \{[k \cdot log(N \cdot p) - N \cdot p - log\, k!]\}$$

$$= \arg\max_N \{[k \cdot log(N \cdot p) - N \cdot p]\}$$

So the ML estimate of the number of gamma rays emitted by the source is

$$\widehat{N}_{ML} = \arg\max_N \{\log[P[k|N]]\} = \arg\max_N \{[k \cdot \log(N \cdot p) - N \cdot p]\}$$

Image Reconstruction and General Linear Inverse Problem

An Emission Tomography System

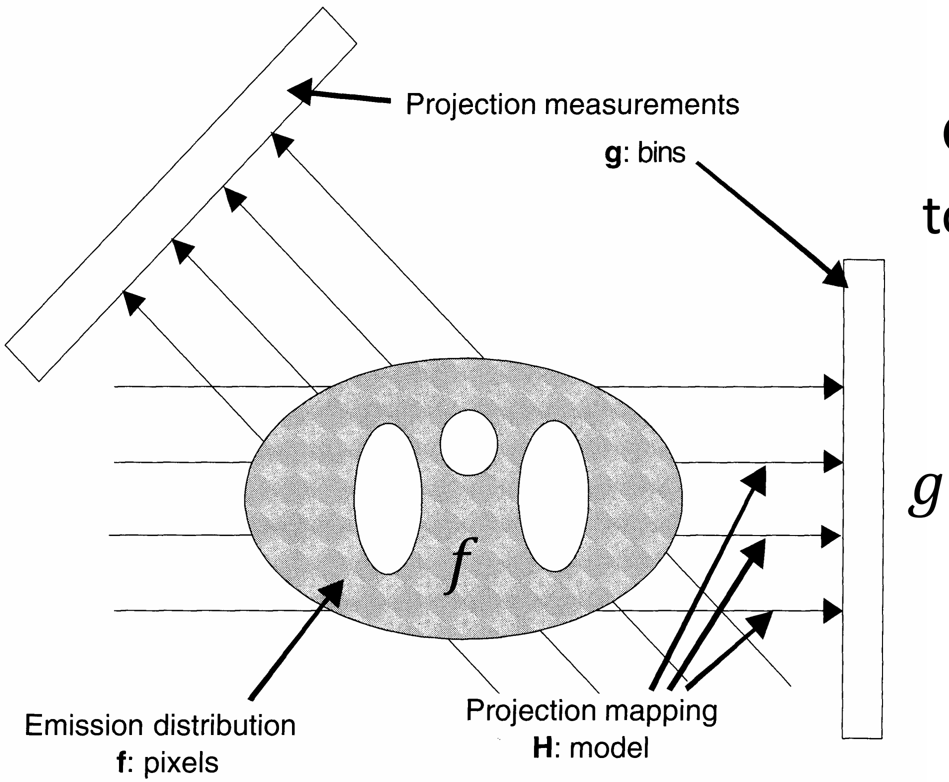


FIGURE 1 A general model of tomographic projection in which the measurements are given by weighted integrals of the emitting object distribution.

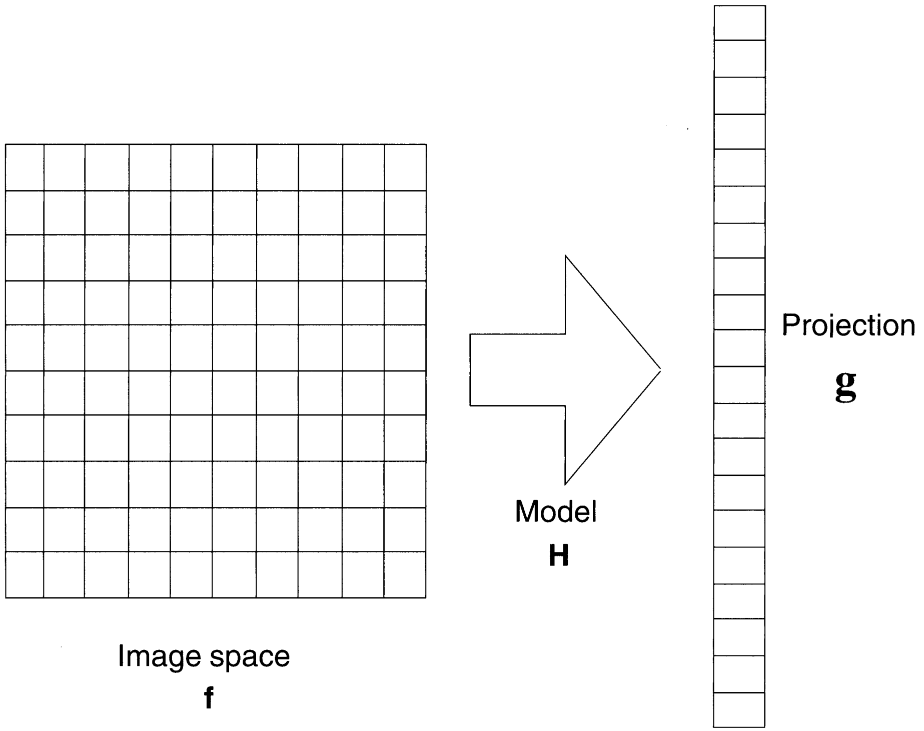
Given a measured data set g, how to formulate the ML estimator of f?

$$\hat{f}_{ML} = \arg\max P(\boldsymbol{g}|\boldsymbol{f})$$

Then how to evaluate $P(\boldsymbol{g}|\boldsymbol{f})$?

Image Reconstruction and General Linear Inverse Problem

Linear imaging systems – An alternative representation



$$\bar{\mathbf{g}} = \mathbf{Hf}$$

FIGURE 2 A discrete model of the projection process.

Poisson Statistics of the Projection Data

Considering that the measured projection data, $\boldsymbol{g}=(g_1, g_2, g_3,\dots, g_M)$, is a collection of independent Poisson random numbers, the probability of the measured projection data $\boldsymbol{g}$ may be written as

$$p(\boldsymbol{g}|\boldsymbol{f}) = p(\boldsymbol{g}) = \prod_{m=1}^{M} p(g_m) = \prod_{m=1}^{M} \frac{{\bar{g}_m}^{g_m}}{g_m!} e^{-\bar{g}_m}$$

where $\bar{g}_m$ is the expected value for the number of counts on detector pixel no. m.

$$\bar{g}_m = \sum_{n=1}^{N} f_n \boxed{p_{mn}}$$

In the context of emission tomography, p_{mn} is the probability that a gamma ray generated at a source pixel n is detected by detector element m.

A Typical Imaging System Described in Matrix Form

$$\begin{pmatrix} \bar{g}_1 \\ \bar{g}_2 \\ \bar{g}_3 \\ \vdots \\ \bar{g}_M \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1N} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2N} \\ p_{31} & p_{32} & p_{33} & \cdots & p_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{M1} & p_{M2} & p_{M3} & \cdots & p_{MN} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{pmatrix}$$

Expected number of photons detected on each detector element

Everything we know about the imaging system – The system response function.

The source object.

The number of photons being emitted from each source voxel

p_{mn} : the probability for a gamma ray emitted by the n'th source voxel to be detected by the m'th detector pixel.

The Maximum Likelihood Reconstruction

$$\hat{\boldsymbol{f}}_{ML} = \underset{f}{\operatorname{argmax}}\, L(\boldsymbol{f},\boldsymbol{g}) = \underset{f}{\operatorname{argmax}}\, \log[L(\boldsymbol{f},\boldsymbol{g})]$$

$$= \underset{f}{\operatorname{argmax}} \left\{ log \left[\prod_{m=1}^{M} \left(\frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m} \right) \right] \right\} = \underset{f}{\operatorname{argmax}} \left[\sum_{m=1}^{M} log \left(\frac{\bar{g}_m{}^{g_m}}{g_m!} e^{-\bar{g}_m} \right) \right]$$

Consider all detector elements working independently, So the ML reconstruction for Poisson-distributed data is

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{f}{\operatorname{argmax}} l(\boldsymbol{f},\boldsymbol{g}) = \underset{f}{\operatorname{argmax}} \left[\sum_{m=1}^{M} (g_m \log \bar{g}_m - \bar{g}_m - \log g_m\,!) \right]$$

since g_m is not function of f, we get

$$\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{f}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{f}{\operatorname{argmax}} \left[\sum_{m=1}^{M} (g_m \log \bar{g}_m - \bar{g}_m) \right]$$

$$\boxed{\bar{g}_m = \sum_{n=1}^{N} f_n p_{mn}}$$

$$\boxed{\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{f}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{f}{\operatorname{argmax}} \left[\sum_{m=1}^{M} \left(g_m \cdot \log \left(\sum_{n=1}^{N} f_n p_{mn} \right) - \left(\sum_{n=1}^{N} f_n p_{mn} \right) \right) \right]}$$

The Maximum Likelihood Reconstruction

Underlying/unknown image function to be estimated

The ML estimate

$$\boxed{\hat{\boldsymbol{f}}_{\boldsymbol{ML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\left(\boldsymbol{f},\boldsymbol{g}\right) = \underset{\boldsymbol{f}}{\operatorname{argmax}}\left[\sum_{m=1}^{M}\left(g_m \cdot \log\left(\sum_{n=1}^{N} f_n p_{mn}\right) - \left(\sum_{n=1}^{N} f_n p_{mn}\right)\right)\right]}$$

Measured data

The Maximum Likelihood Expectation Maximization (MLEM) Algorithm

The source (image) function $\boldsymbol{f}$ that has the maximum likelihood of giving rise to the observed data $\boldsymbol{g}$ can be found by the following iterative updating scheme

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{mn'} f_{n'}^{(old)}} p_{mn}\,, \qquad n = 1,2,\ldots,N$$

$where$

$m = 1,2,\ldots,M$ is the index of detector elements in the imaging system

$n = 1,2,\ldots,N$ is the index of source object pixels

$f_n^{(old)}$: the current estimate of the source function in pixel n

$f_n^{(new)}$: the updated estimate of the source pixel n

p_{nm}: the probability of a gamma ray generated in source pixel n and detected by the detector element m

g_m: the observed number of counts in detector pixel m

Shepp & Vardi, IEEE TMI, 1982.

The Maximum Likelihood Expectation Maximization (MLEM) Algorithm

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{mn'} f_{n'}^{(old)}} p_{mn}\,, \qquad n = 1,2,\ldots,N$$

Each iteration updates all elements in the source function *f* sequentially.

Each iteration is guaranteed to provide a new image function that has an *increased likelihood* compared to the old image function (unless the maximum likelihood solution has been achieved).

IEEE Trans Med Imaging, Maximum likelihood reconstruction for emission tomography L A Shepp, Y Vardi, 1982.

Structure of MLEM Algorithm

$$f_n^{(new)} = \frac{f_n^{(old)}}{\sum_{m=1}^{M} p_{nm}} \sum_{m=1}^{M} \frac{g_m}{\sum_{n'=1}^{N} p_{n'm} f_{n'}^{(old)}} p_{nm}, \quad n = 1,2,\ldots,N$$

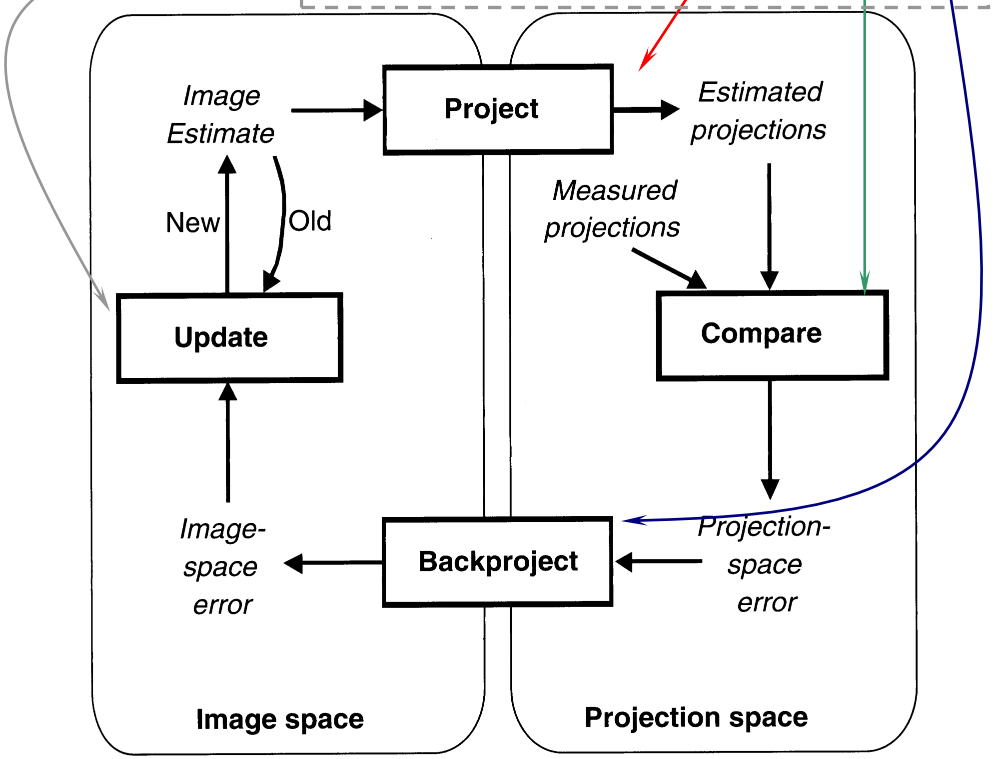


FIGURE 6 Flowchart of a generic iterative reconstruction algorithm.

An Example of PET Image Reconstruction

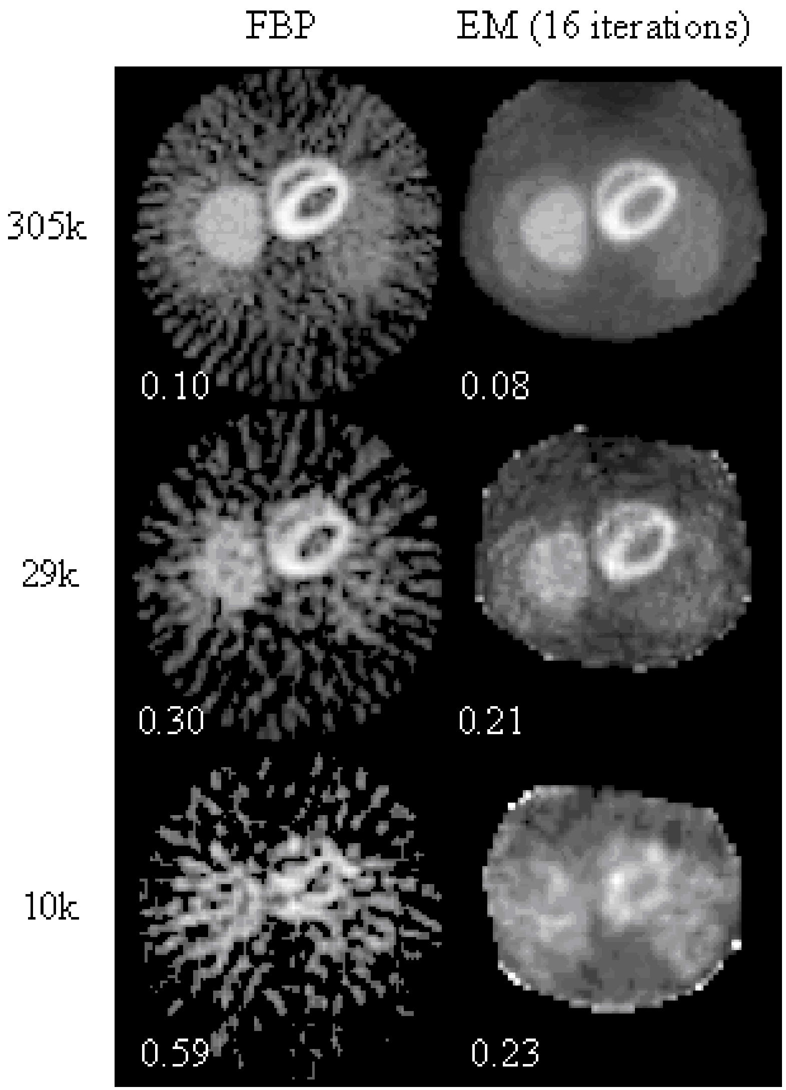


Figure 3: Comparison of EM reconstruction with FBP for different total counts. Note particular the streaking and noise appearance at low counts using FBP. Figures indicate the noise (standard deviation / mean) for a region in the liver.

http://web.uchile.cl/vignette/borrar3/alasbimn/CDA/sec_a/0,1205,SCID%253D470%2526PRT%253D455%2526LNID%253D10,00.html

Properties of the MLEM Algorithm

Advantages

- Relatively *simple* but *highly useful*.
- Provides room for incorporating a wide range of physical effects and detector system characteristics and therefore provide an *improved image quality* for low statistics data.
- *Consistent and predictable behavior*.
- Automatically incorporates *non-negativity constraint*. So the image values are always non-negative.
- Provides *theoretically optimum solution* (Be careful when saying this!!).

Properties of the MLEM Algorithm

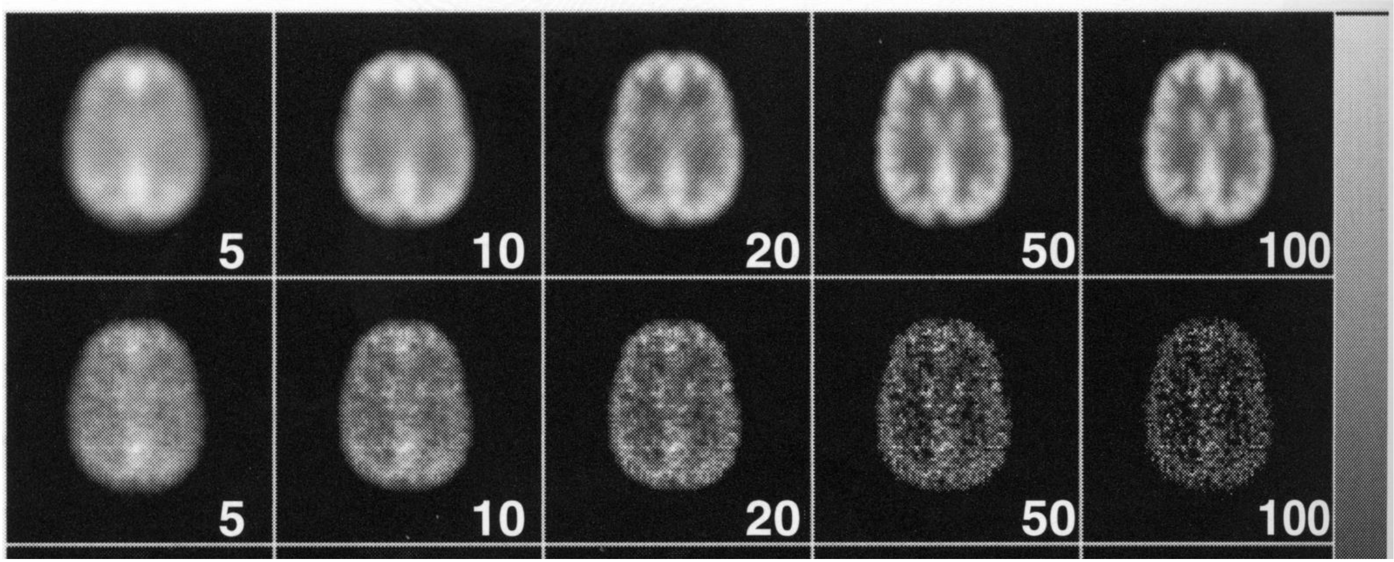
Shortcomings

- *Relatively slow* convergence. For typical 3-D reconstruction, it takes several tens of iterations to converge. In principle, each iteration takes similar amount of computation to that for a complete FBP reconstruction!
- Tends to yield *noisy reconstructions* when projection data has low statistics.

What exactly is the problem?

The algorithm tends to fit to the noise in the (observed) projection data!

Properties of the MLEM Algorithm



With noise free data

With noisy data

Properties of MLEM Algorithm

How to improve?

Adding extra (or sometime called *a priori*) information in the reconstruction process.

This can be done using two numerically equivalent approaches:

- This can be done by trying to discourage possible image solutions that have large fluctuations (noisy images). – Penalized Maximum Likelihood (PML) methods.
- Using the Bayes' law to incorporate the *a priori* information – Maximum a posteriori (MAP) methods.

Further Improvements – Penalized ML Algorithms

The basic idea:

Instead of finding the original image by maximizing the likelihood function:

$$\hat{\boldsymbol{f}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, \boxed{p(\boldsymbol{g} \mid \boldsymbol{f})} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, \boxed{l(\boldsymbol{f}, \boldsymbol{g})}$$

Probability of a measurement *g* given the underlying image *f*

The so-called log-likelihood function

We can use a different selection criterion

$$\hat{\boldsymbol{f}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, \boxed{\left[l(\boldsymbol{f}, \boldsymbol{g}) - \boxed{\beta U(\boldsymbol{f})}\right]}$$

The so-called penalty function that has increased value when an undesired imaging feature presents in the solution.

β is a scaling constant that controls "how strong the penalty is for the presence of a given feature"

So by minimizing, we are selecting an image by dis-encouraging the undesired features to be presented in the image.

Penalized Maximum Likelihood (PML) reconstruction …

An Example Penalized ML Algorithms

An example is to use the so-called **roughness penalty** function that measures the differences between neighboring pixels in the image, as follows

$$\boxed{\hat{\boldsymbol{f}}_{\boldsymbol{PML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}}\left[l(\boldsymbol{f},\boldsymbol{g}) - \beta \sum_{j=1}^{N} \sum_{k=\mathcal{N}_j} \phi(f_j - f_k)\right],}$$

where

$\phi(f_j - f_k)$ may be defined as $a_{jk}(f_j - f_k)^2$,

$\mathcal{N}_j$ refers to the collection of neighboring pixels of pixel *j*, and

a_{jk} is a user-defined weighting factor to reflect different preferences on how the user would like to penalize the differences between the target pixel j and its neighboring pixels.

So the above Penalized ML Algorithm becomes

$$\boxed{\hat{\boldsymbol{f}}_{\boldsymbol{PML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\ l\,(\boldsymbol{f},\boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}}\left[l(\boldsymbol{f},\boldsymbol{g}) - \beta \sum_{j=1}^{N} \sum_{k=\mathcal{N}_j} a_{jk}(f_j - f_k)^2\right],}$$

Further Improvements – Maximum A Posteriori (MAP) Algorithms

Alternatively, we may consider the unknown image, ***f***, as a random variable that follows a statistical law *p*(***f***).

The Bayis's law:

if the underlying image is *f*, the conditional probability of observing the measurement *g* is

a priori probability

$$\boxed{p(\boldsymbol{f}\,|\,\boldsymbol{g})} = \frac{\boxed{p(\boldsymbol{g}\,|\,\boldsymbol{f})}\,\boxed{p(\boldsymbol{f})}}{p(\boldsymbol{g})}$$

a posteriori probability

This leads to the maximum a posteriori (MAP) estimation …

Further Improvements –

Bayesian Estimation and Maximum A Posteriori (MAP) Algorithms

The maximum a posteriori (MAP) approach…

$$\hat{\boldsymbol{f}}_{MAP} = \underset{\boldsymbol{f}}{\operatorname{argmax}} \log p(\boldsymbol{f} \mid \boldsymbol{g}) = \underset{\boldsymbol{f}}{\operatorname{argmax}} \log \frac{p(\boldsymbol{g} \mid \boldsymbol{f})\, p(\boldsymbol{f})}{p(\boldsymbol{g})}$$

$$= \underset{\boldsymbol{f}}{\operatorname{argmax}} \left[\log \boxed{p(\boldsymbol{g} \mid \boldsymbol{f})} + \log \boxed{p(\boldsymbol{f})}\right]$$

The likelihood function

The *a priori* probability of the data. This is where you can fold in your prior knowledge about the underlying image … for example, the image "should" be relatively smooth …

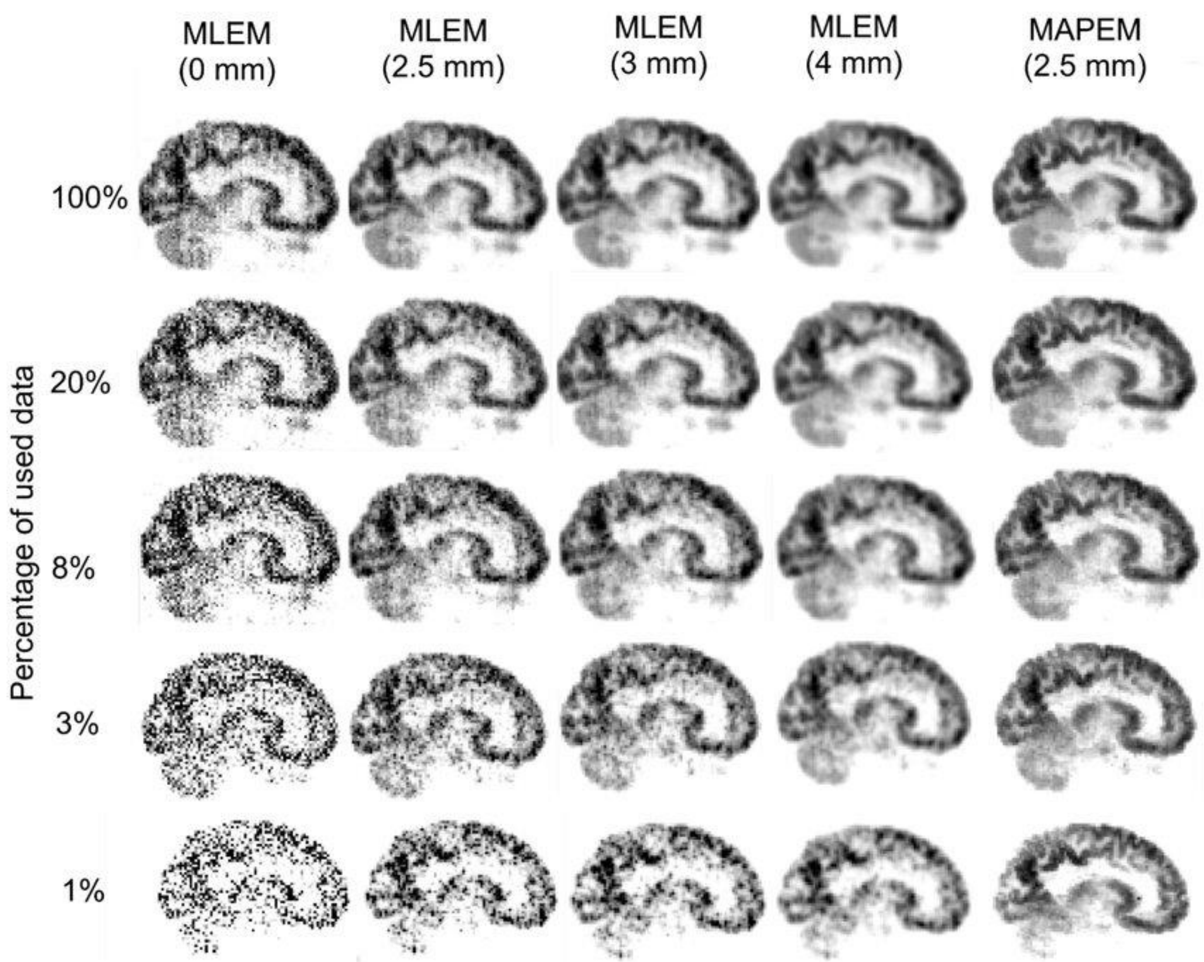
This is numerically equivalent to PML approach if we write

$$p(\boldsymbol{f}) = -\beta U(\boldsymbol{f}),$$

therefore

$$\hat{\boldsymbol{f}}_{\mathrm{PML}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, [\log p(\boldsymbol{g}|\boldsymbol{f}) - \beta U(\boldsymbol{f})] \Leftrightarrow \hat{\boldsymbol{f}}_{\mathrm{MAP}} = \underset{\boldsymbol{f}}{\operatorname{argmax}}\, [\log p(\boldsymbol{g}|\boldsymbol{f}) + \log p\,(\boldsymbol{f})].$$

Example PET Images Reconstructed with MLEM and PML Algorithms



The reconstruction of various percentages of the FDG PET data using the MLEM and MAP-EM algorithms. The MLEM reconstructions have each been post-smoothed using a Gaussian kernel with 0, 2.5, 3, and also 4 mm FWHM. All PET images are shown with the same grey scale intensity range.

"PET image reconstruction using multi-parametric anato-functional priors", *Mehranian, et al., PMB, 2017.*

Example PET Images Reconstructed with MLEM and PML Algorithms

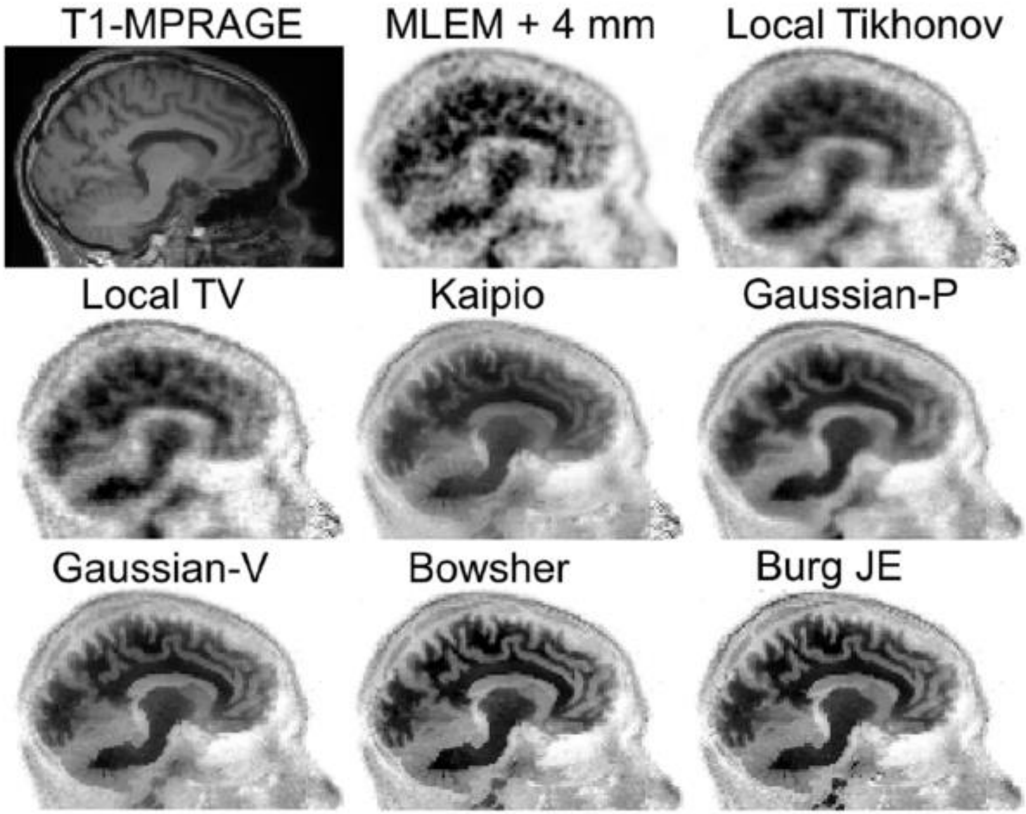


Figure 10. The reconstruction results of the 5 min of the [^{18}F]florbetaben dataset using different algorithms. All PET images are shown with the same displaying window.

"PET image reconstruction using multi-parametric anato-functional priors", *Mehranian, et al., PMB, 2017.*

A Brief Introduction on Matlab Functions Related to Image Processing

Overview

Getting Help

2D Matrix / Image

- **Coordinate system**
- **Display**
- **Storing images**

2D Functions

Discrete Fourier Transform

- **1D DFT**
- **fftshift**
- **2D DFT: zero-filling and shifting the image**

Getting Help in Matlab

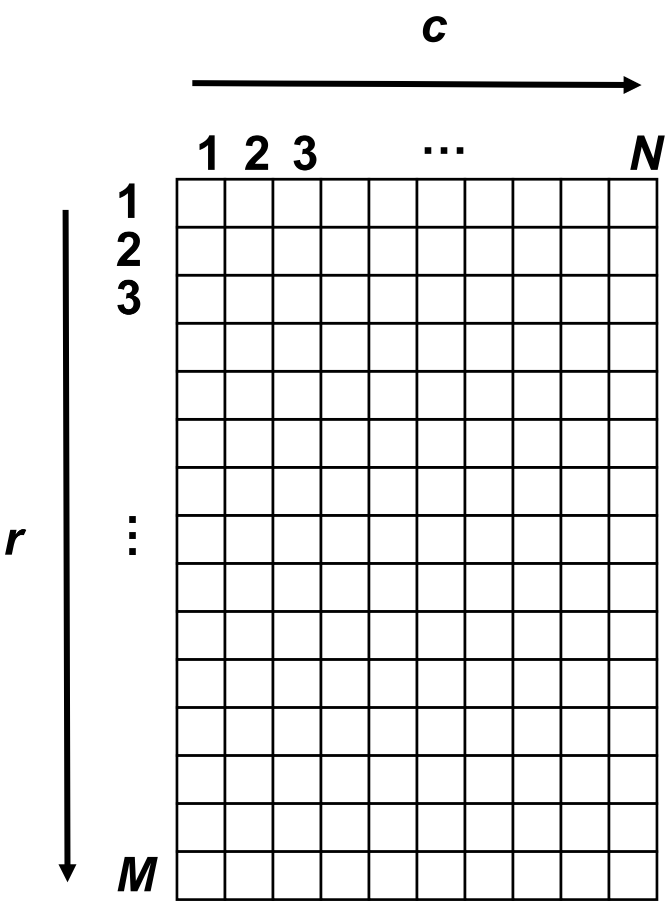
Online Help

- **Help Browser (? In toolbar or >>helpbrowser)**
- **Help Functions (>>help *functionname*)**
- **Matlab website: www.mathworks.com**
- **Demos**

Documentation

- **Online as pdf files**

2D Matrix



2D matrix has *M* rows and *N* columns

Example

- **>>m=zeros(400,200);**
- **>>m(1:50,1:100) = 1;**

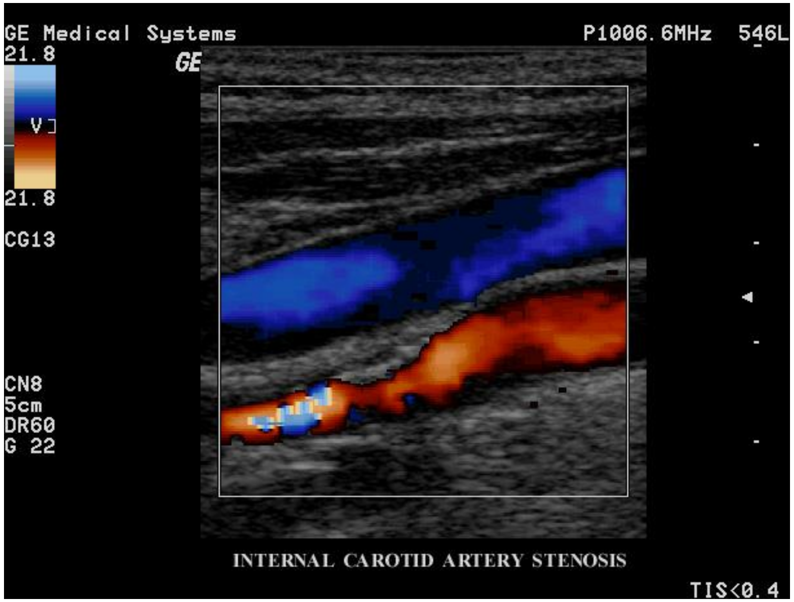
Note:

- **Index 1 to M, not 0 to M-1**
- **Origin of coordinate system is in upper left corner**
- **Row index increases from top to bottom**
- **Coordinate system is rotated in respect to 'standard' x-y coordinate system**

Image Types

Medical images are mostly represented as grayscale images

- **Human visual system resolves highest spatial resolution for grayscale contrast**
- **Color is sometimes used to superimpose information on anatomical data**



Save Image Matrix

MATLAB binary format

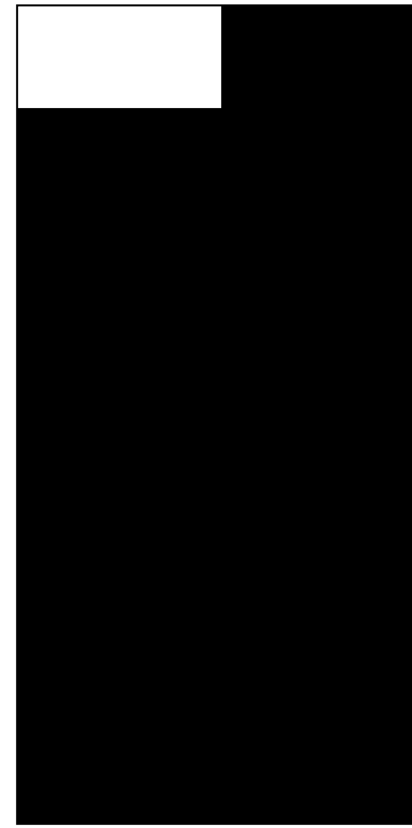
- **>>save example**

  -> writes out example.mat

Standard image format (bmp,tiff,jpeg,png,hdf,pcx,xwd)

- **>>imwrite(m,’example.tif’,’tif’)**

  -> writes out example.tif
- **Warning: imwrite expects data in range [0…1]**
- **If data range outside this [0…1], use mat2gray**
- **>>m_2 = mat2gray(m);**
- **>>imwrite(m_2,’example.tif’,’tif’)**



Save Figure (incl. labels etc)

Save in Matlab Figure Format:

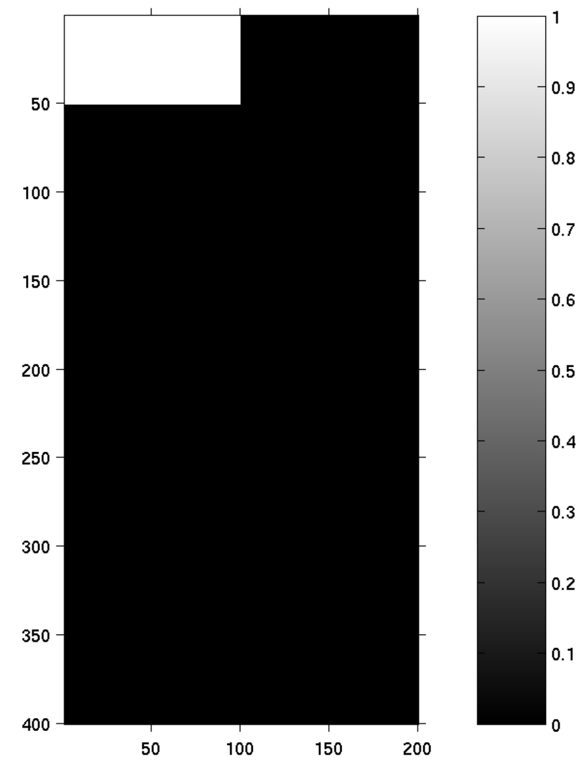
- **>File>Save As> in Figure Window**

  -> writes out example.fig

Standard image formats

- **>File>Export in Figure Window**
  - **E.g. jpg, tif, png, eps, …**
- **Alternatively, use print, e.g.**

  >>print –deps2 example.eps



Loading 2D Data and Images

Load matrix from MATLAB binary format

>>load example (loads example.mat)

Load matrix from standard image format

>>m_in = imread('example.tif','tif');

To check on an image: >>imfinfo('example.tif','tif')

Load figure from Matlab Figure Format (*.fig):

- **>File>Open> 'example.fig' in Figure Window**

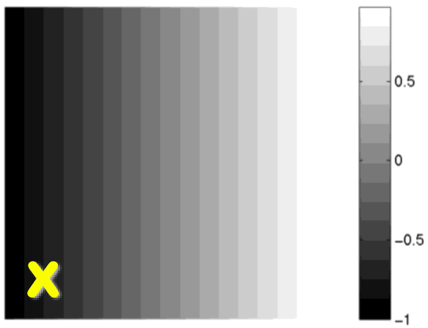
->Check loaded matrices

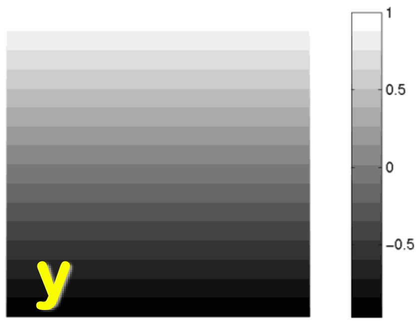
- **>>whos**

Some Common Image Formats

- TIF (TIFF)
  - Very common, compatible, all purpose image format
  - Allows for lossless compression
- JPEG
  - Allows for lossy compression (small file size)
  - Very common for internet
- PNG (portable network graphics)
  - Good for saving MATLAB images for importing into Microsoft documents such as Word
- Dicom
  - THE medical imaging format
  - Lots of header information (patient name & ID, imaging parameters, exam date, …) can be stored
  - Allows for lossy and lossless compression
  - Matlab function ‘dicomread’, ‘dicomwrite’
- EPS (Encapsulated Postscript)
  - Graphics format rather than an image format
  - Great for best quality print results
  - Large file size

2D Functions

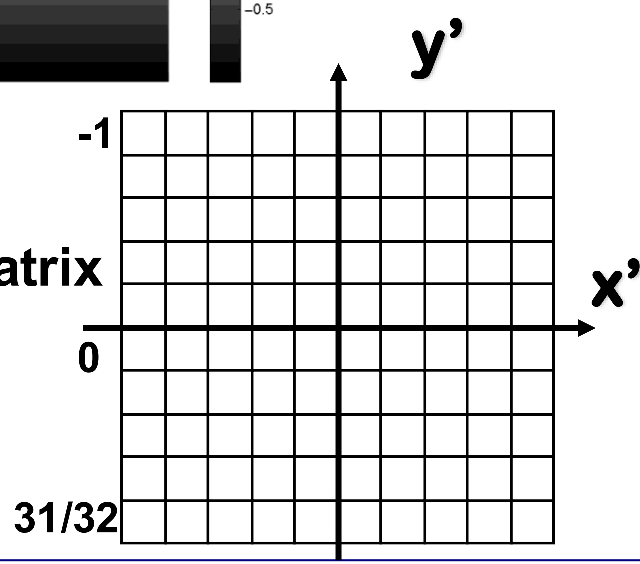


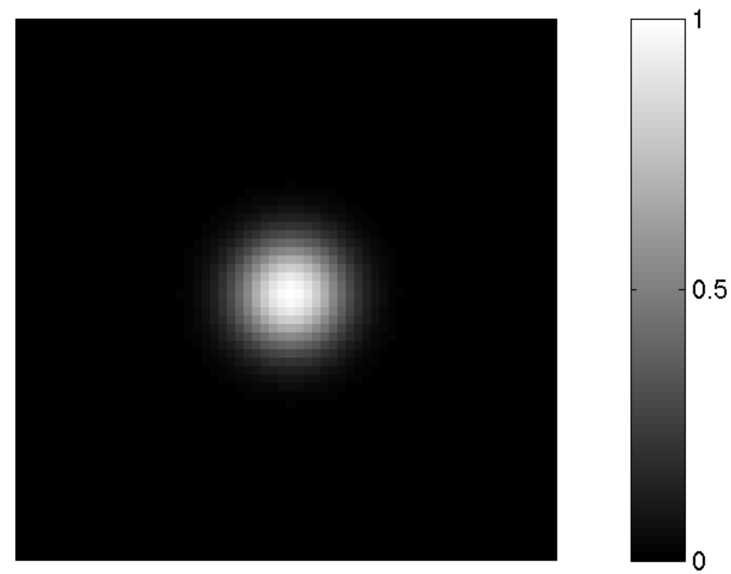


- **Create a matrix that evaluates 2D Gaussian: exp(-p/2(x^2+y^2)/s^2)**

```
>>ind = [-32:1:31] / 32;
>>[x,y] = meshgrid(ind,-1*ind);
>>z = exp(-pi/2*(x.^2+y.^2)/(.25.^2));
>>imshow(z)
>>colorbar
```

64x64 matrix





Discrete Fourier Transform in 1-D Revisited

The *discrete Fourier transform* (DFT) is defined as

$$F_n = \sum_{k=0}^{N-1} f_k e^{-\frac{j2\pi nk}{N}}, \mathrm{n = 0,1,2,...N - 1}$$

$\mathrm{n = 0}$ corresponding to the DC component (spatial frenquency is zero)

$\mathrm{n = 1,..., N/2 - 1}$ are corresponding to the positive frenquencies $0 < \mathrm{u} < \mathrm{u_c}$

$\mathrm{n = N/2, ..., N - 1}$ are corresponding to the negative frenquencies $-\mathrm{u_c} < \mathrm{u} < 0$

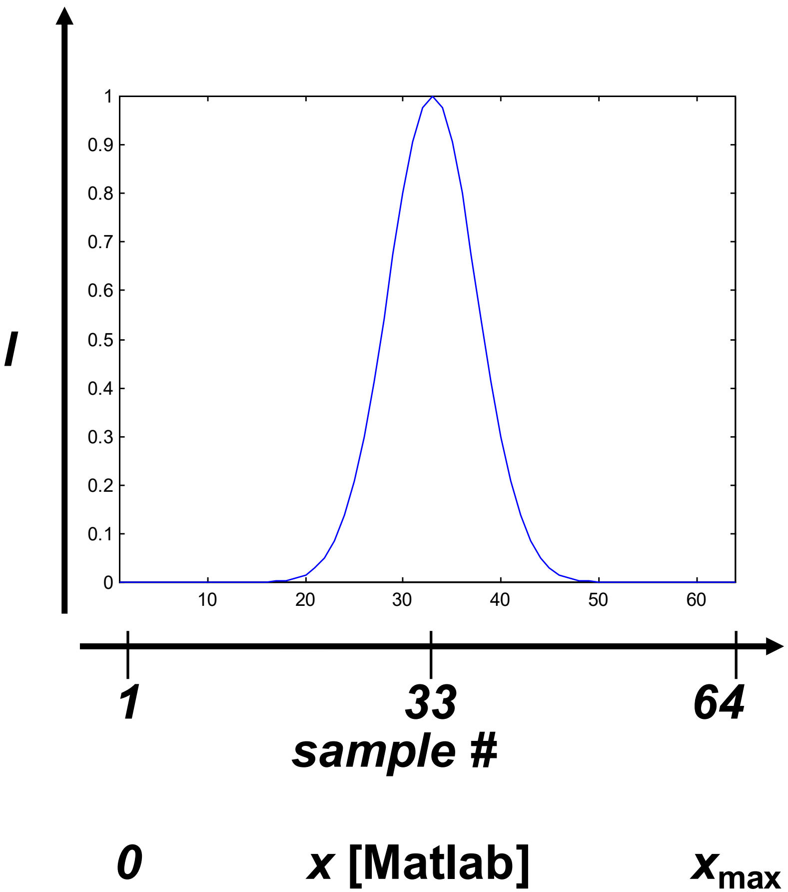
And the *inverse DFT* is defined as

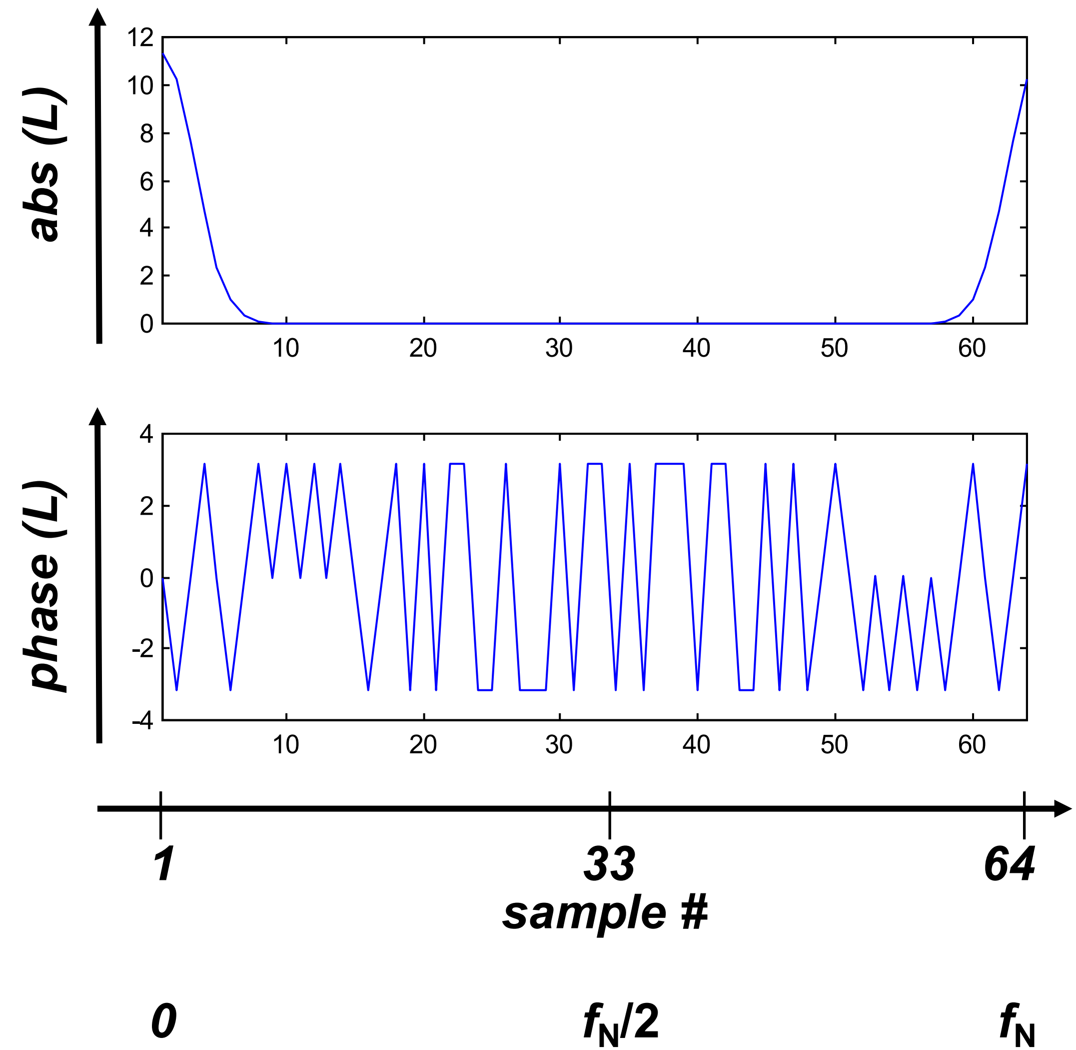
$$f_k = \frac{1}{N}\sum_{n=0}^{N-1} F_n e^{-\frac{j2\pi nk}{N}}, \mathrm{k = 0,1,2,...N - 1}$$

FT Conventions - 1D

1D Fourier Transform

- >>I = z(33,:);
- >>L = fft(I);

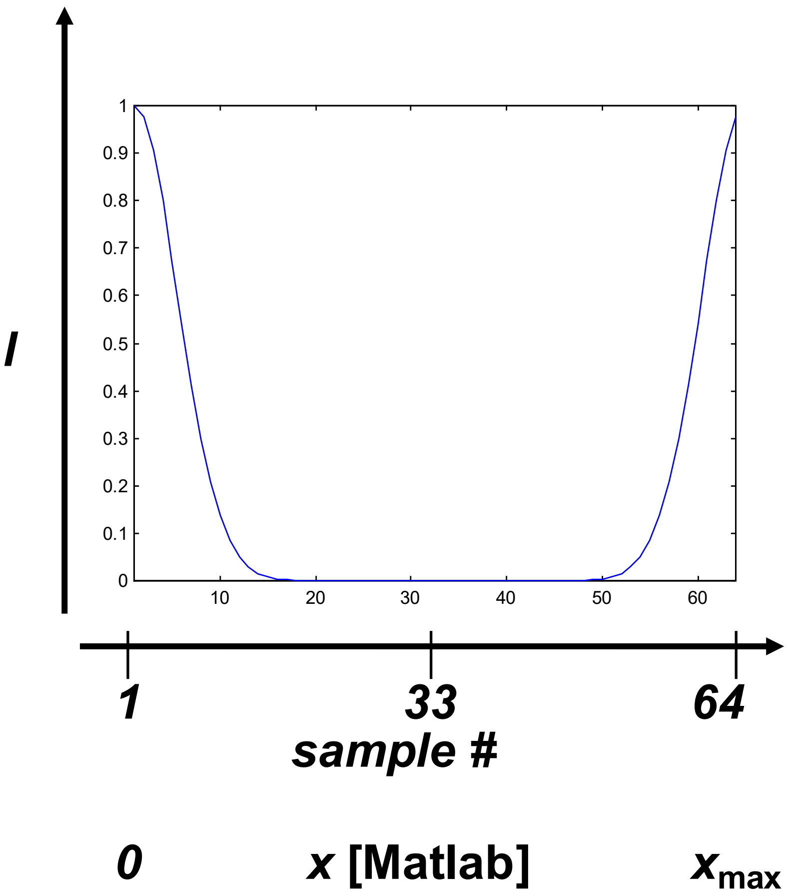


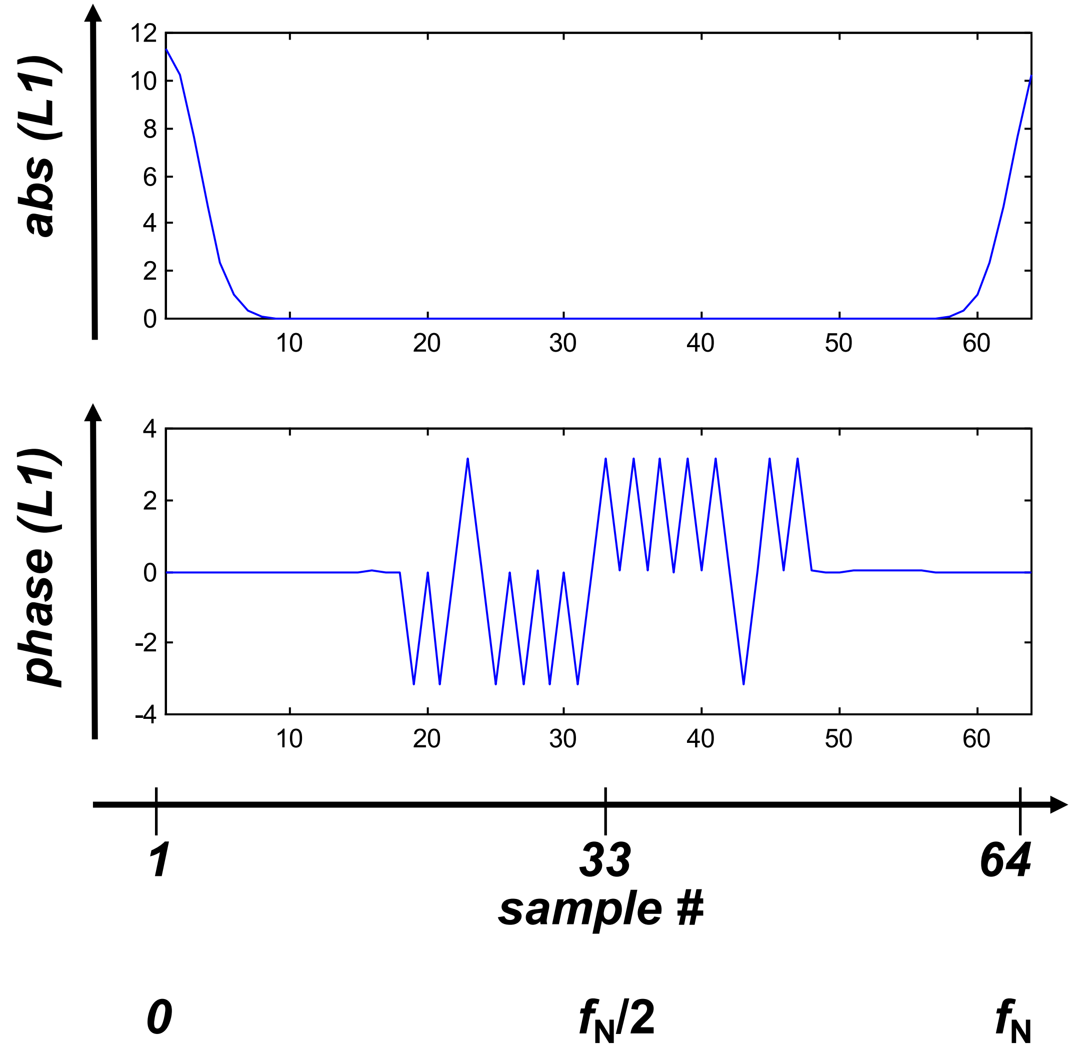


1D FT - fftshift

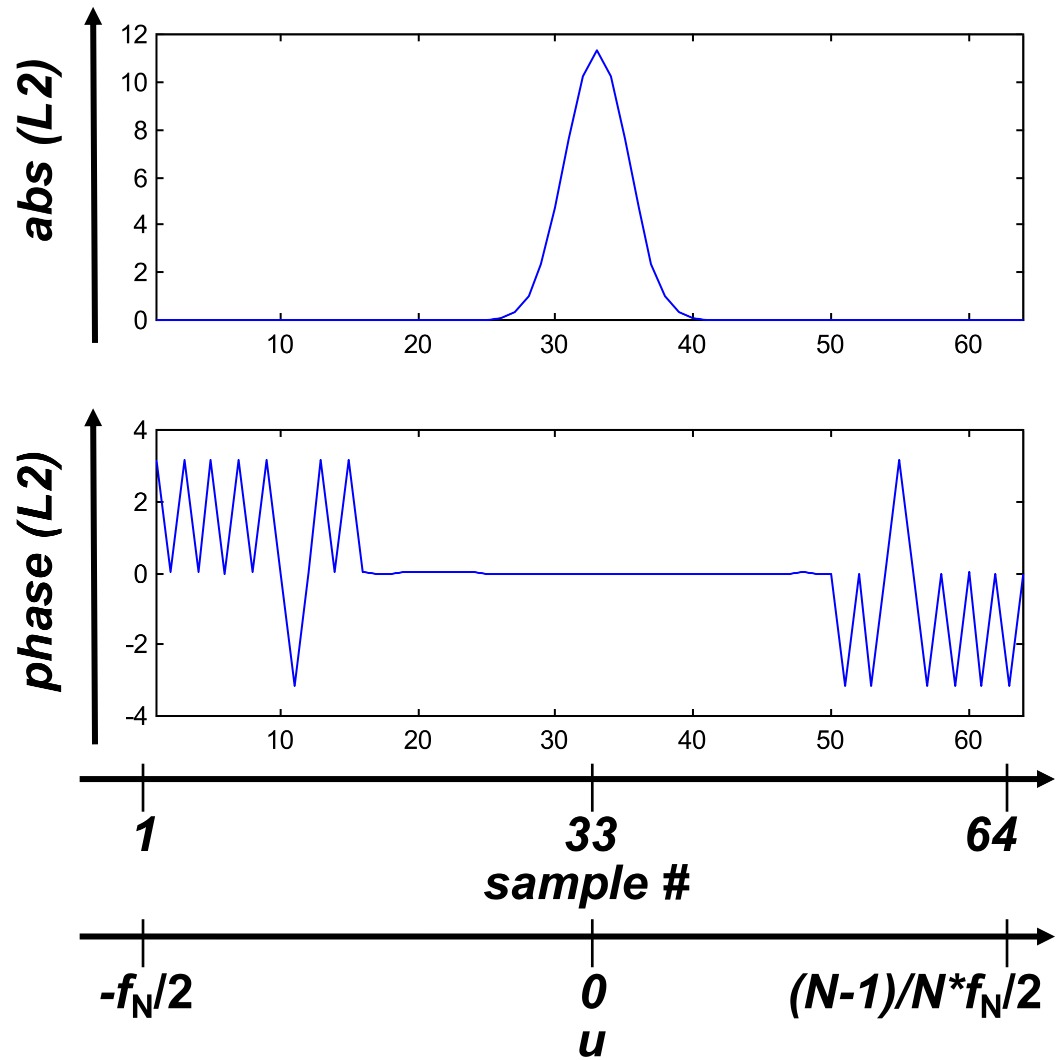
Use fftshift

>>L1 = fft(fftshift(I));





1D FT - 2xfftshift

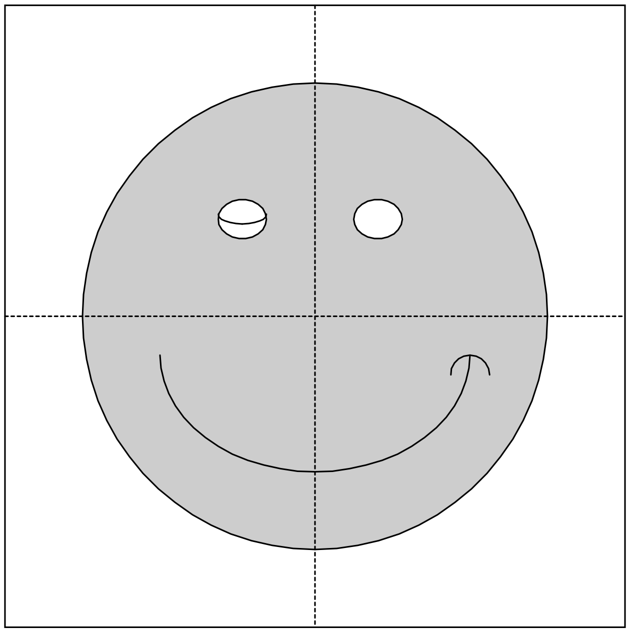


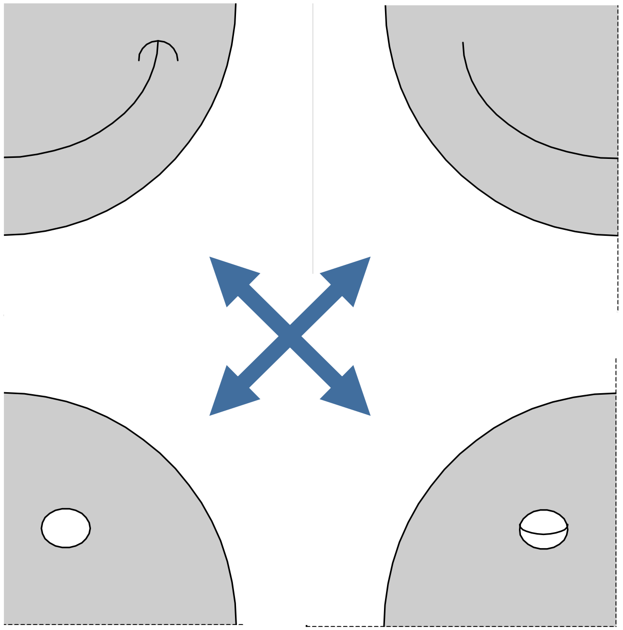
Center Fourier Domain: >>L2 = fftshift(fft(fftshift(l)));

fftshift in 2D

Use fftshift for 2D functions

- **>>smiley2 = fftshift(smiley);**

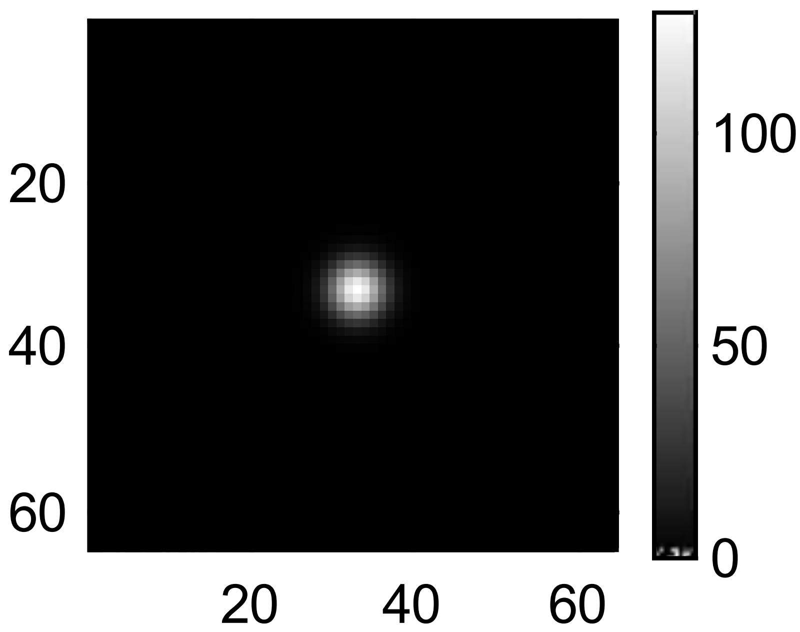




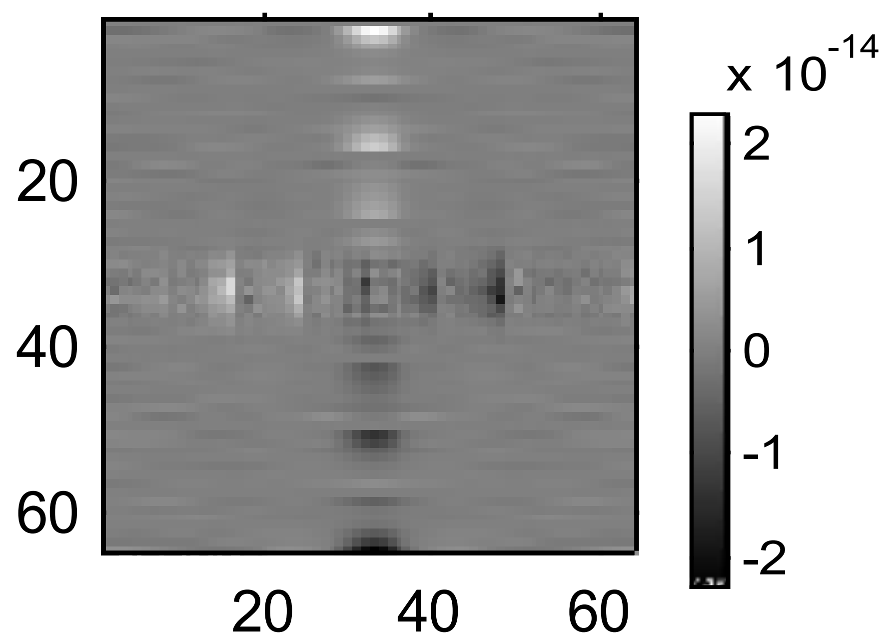
2D Discrete Fourier Transform

- >>Z=fftshift(fft2(fftshift(z)));
- >>whos
- >>figure
- >>imshow(real(Z))
- >>imshow(real(Z),[])
- >>colorbar

- >>figure
- >>imshow(imag(Z),[])
- >>colorbar



Real (Z)

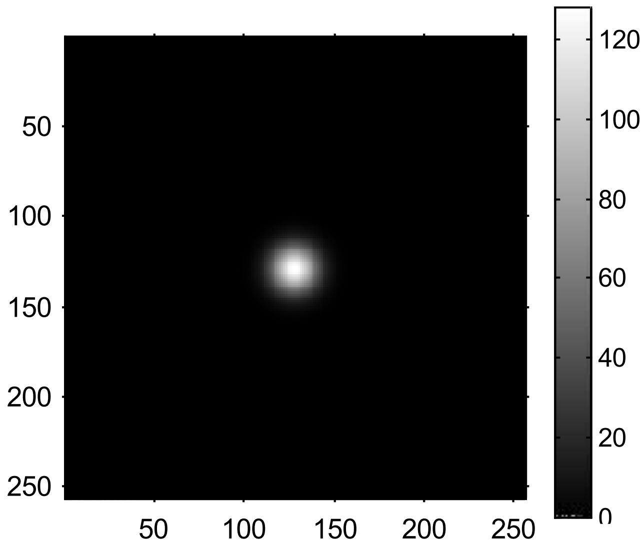


Imag (Z)

2D DFT - Zero Filling

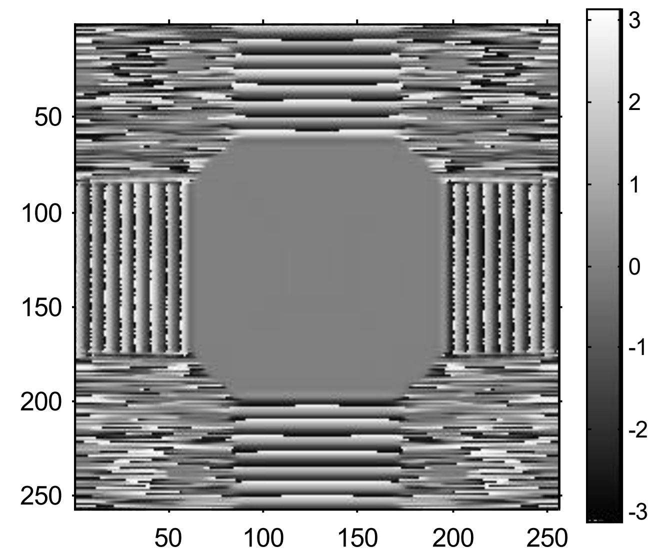
Interpolating Fourier space by zerofilling in image space

- **>>z_zf = zeros(256);**
- **>>z_zf(97:160,97:160) = z;**
- **>>Z_ZF = fftshift(fft2(fftshift(z_zf)));**
- **>>figure**
- **>>imshow(abs(Z_ZF),[])**
- **>>colorbar**



Magnitude (Z_ZF)

- **>>figure**
- **>>imshow(angle(Z_ZF),[])**
- **>>colorbar**



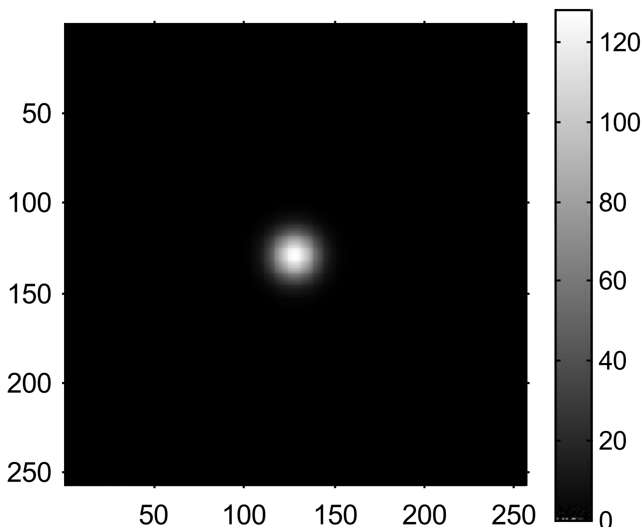
Phase (Z_ZF)

2D DFT - Voxel Shifts

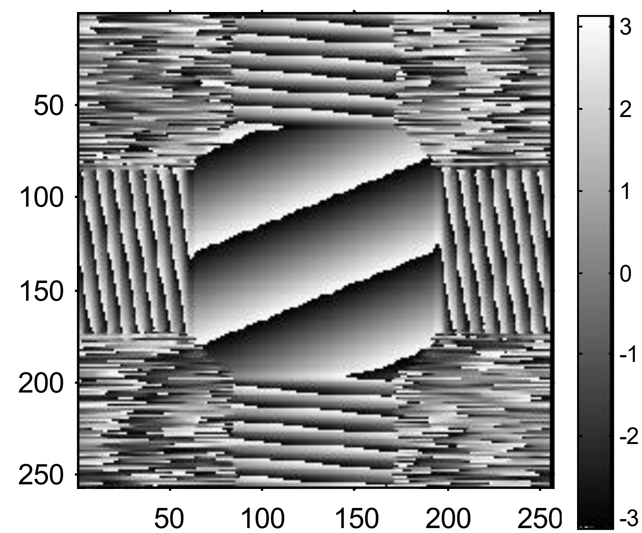
Shift image: 5 voxels up and 2 voxels left

- **>>z_s = zeros(256);**
- **>>z_s(92:155,95:158) = z;**
- **>>Z_S = fftshift(fft2(fftshift(z_s)));**
- **>>figure**
- **>>imshow(abs(Z_S),[])**
- **>>colorbar**

- **>>figure**
- **>>imshow(angle(Z_S),[])**
- **>>colorbar**



Magnitude (Z_S)



Phase (Z_S)

Review of functions …

Input / Output

- **>>imread**
- **>>imwrite**
- **>>imfinfo**

Image Display

- **>>imshow**

Others

- **>>axis**
- **>>colorbar**
- **>>colormap**
- **>>meshgrid**
- **>>fft, fft2**
- **>>fftshift**

More useful functions …

Image Display

- **>>title**
- **>>xlabel, ylabel**
- **>>subimage**
- **>>zoom**
- **>>mesh**

Others

- **>>imresize**
- **>>imrotate**
- **>>imhist**
- **>>conv2**
- **>>radon**
- **>>roipoly**

Demos in Image Processing Toolbox

- **>>help imdemos**

Matlab Examples

P = phantom('Modified Shepp-Logan',200); imshow(P)

Shepp-Logan phantom is a standard head CT simulation.

Ref: Matlab Help Guide

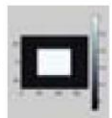
Matlab Examples

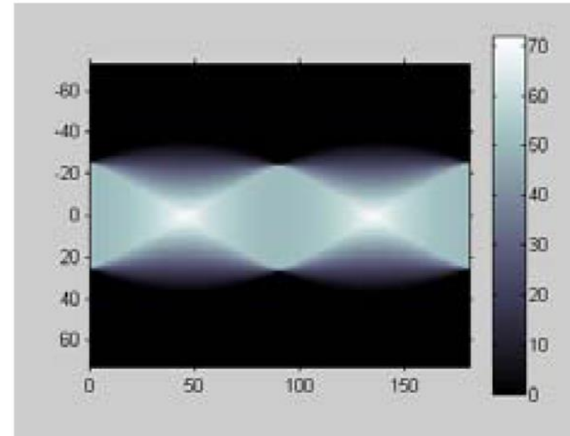
R = radon(I,theta)

If you omit theta, it defaults to 0:179.

Imaging a square

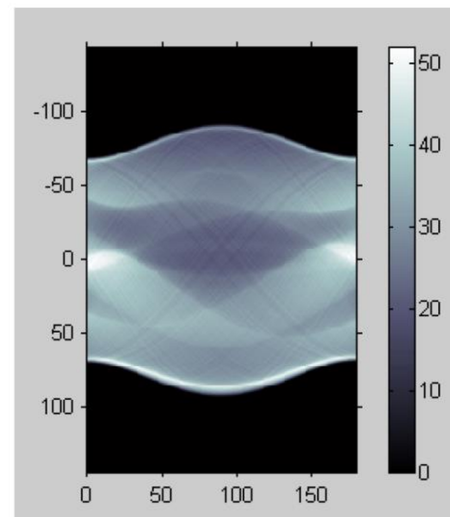
```
iptsetpref('ImshowAxesVisible','on')
I = zeros(100,100);
I(25:75,25:75) = 1;
theta = 0:180;
[RI,xp] = radon(I,theta);
imshow(theta,xp,RI,[],'notruesize'), colormap(bone), colorbar
```





Imaging the Modified Shepp-Logan phantom

```
P = phantom('Modified Shepp-Logan',200);  imshow(P)
[RP,xp] = radon(P,theta);
imshow(theta,xp,RP,[],'notruesize'), colormap(bone), colorbar
```



Ref: Matlab Help Guide

Matlab Examples

I = iradon(P,theta)
I = iradon(P,theta,interp,filter,d,n)

interp specifies the type of interpolation to use in the backprojection. The available options are listed in order of increasing accuracy and computational complexity:
'nearest' - nearest neighbor interpolation
'linear' - linear interpolation (default)
'spline' - spline interpolation

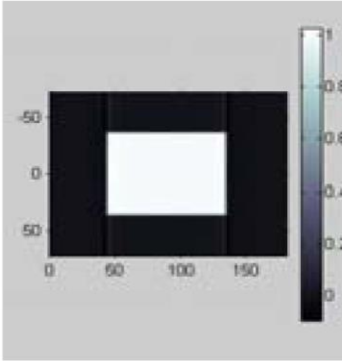
filter specifies the filter to use for frequency domain filtering. filter is a string that specifies any of the following standard filters:
'Ram-Lak' - The cropped Ram-Lak or ramp filter (default). The frequency response of this filter is | f |. Because this filter is sensitive to noise in the projections, one of the filters listed below may be preferable. These filters multiply the Ram-Lak filter by a window that de-emphasizes high frequencies.
'Shepp-Logan' - The Shepp-Logan filter multiplies the Ram-Lak filter by a sinc function.
'Cosine' - The cosine filter multiplies the Ram-Lak filter by a cosine function.
'Hamming' - The Hamming filter multiplies the Ram-Lak filter by a Hamming window.
'Hann' - The Hann filter multiplies the Ram-Lak filter by a Hann window.

d is a scalar in the range (0,1] that modifies the filter by rescaling its frequency axis. The default is 1. If d is less than 1, the filter is compressed to fit into the frequency range [0,d], in normalized frequencies; all frequencies above d are set to 0.

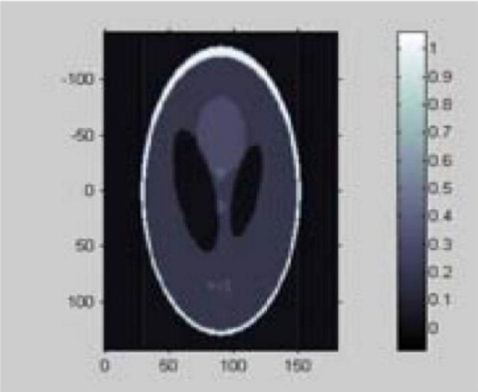
n is a scalar that specifies the number of rows and columns in the reconstructed image. If n is not specified, the size is determined from the length of the projections.
n = 2*floor(size(P,1)/(2*sqrt(2)))

Matlab Examples

I = iradon(P,theta)
I = iradon(P,theta,interp,filter,d,n)

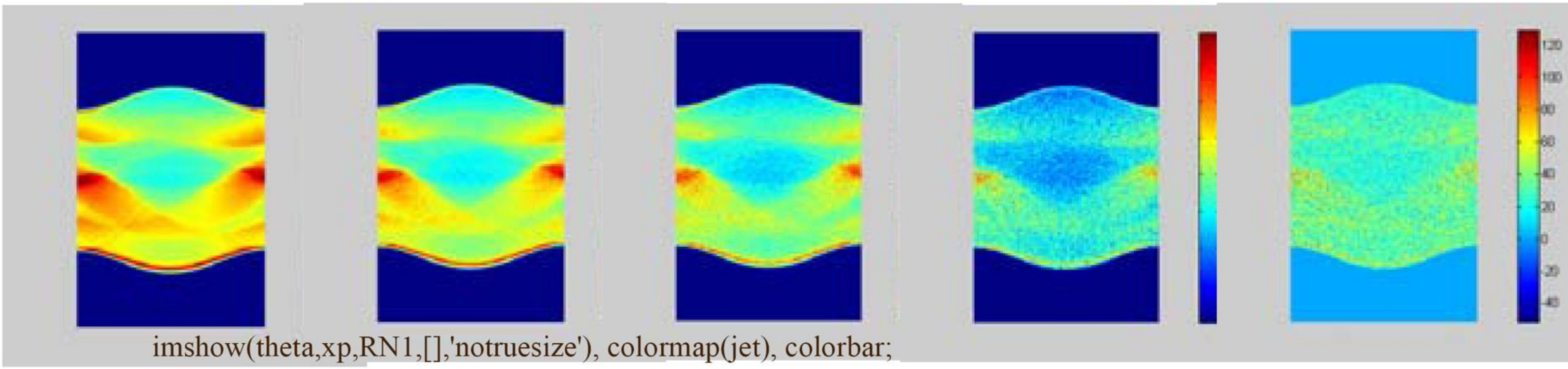


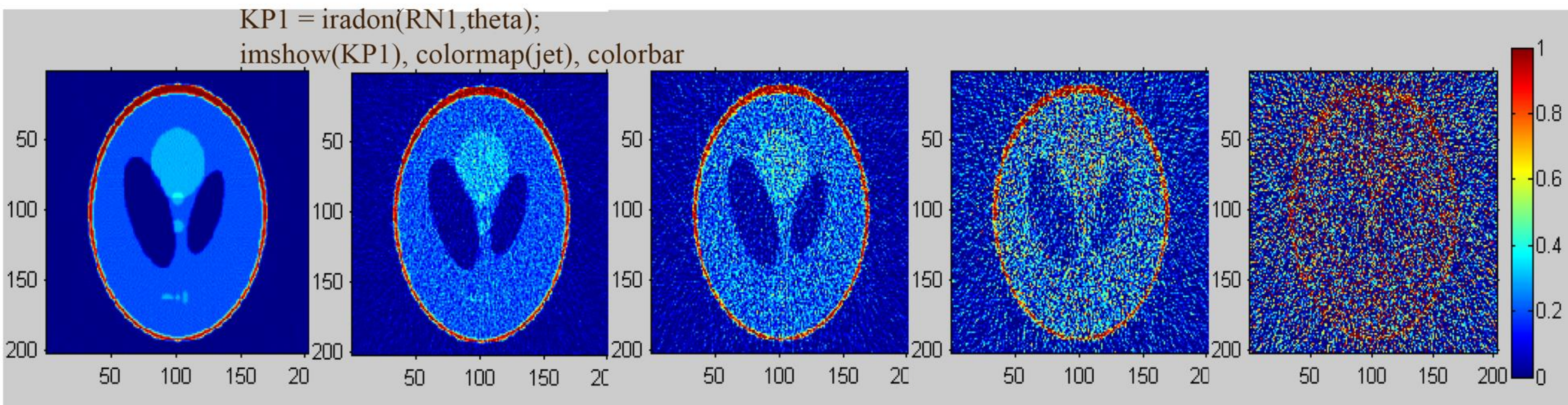
KI = iradon(PI,theta)
imshow(theta,xp,KI,[],'notruesize'), colormap(bone), colorbar



KP = iradon(P,theta)
imshow(theta,xp,KP,[],'notruesize'), colormap(bone), colorbar

Matlab Examples





sd=0 sd=0.05 sd=0.10 sd=0.20 sd=0.50