Computer Tomography: Computational theory and methods

University of Bergen

15 October 2004, Bergen.

Borislav V. Minchev

Borko.Minchev@ii.uib.no

http://www.ii.uib.no/∼borko

Department of computer science

University of Bergen, Norway

Outline

- Introduction
- Fields of applications
- Principles of X-ray CT
- Reconstruction techniques in 2D
  - Transform based methods (FBP)
  - Algebraic reconstruction techniques
- Con-beam reconstructions
  - Circular
  - Helical
- Conclusions

Definition

Tomography: technique for imaging cross-sections or slices from a set of external measurements of a spatially varying function.

Fields of applications

- X-ray CT
  - medical imaging
  - industrial nondestructive evaluation
  - airport screening
  - microtomography

Fields of applications

- X-ray CT
  - medical imaging
  - industrial nondestructive evaluation
  - airport screening
  - microtomography
- Magnetic Resonance Imaging (MRI)
  - electromagnetic waves

Fields of applications

- X-ray CT
  - medical imaging
  - industrial nondestructive evaluation
  - airport screening
  - microtomography
- Magnetic Resonance Imaging (MRI)
  - electromagnetic waves
- Transmission electron microscopy
  - an electron beem

Fields of applications

- X-ray CT
  - medical imaging
  - industrial nondestructive evaluation
  - airport screening
  - microtomography
- Magnetic Resonance Imaging (MRI)
  - electromagnetic waves
- Transmission electron microscopy
  - an electron beem
- CT principles are also applied in:
  - oceanography
  - astronomy
  - geophysics
  - optics

Some hisrory

- Radon 1917
  - reconstruction of a function from its projections

Some hisrory

- Radon 1917
  - reconstruction of a function from its projections
- Hounsfield 1972
  - patented the first CT scanner

Some hisrory

- Radon 1917
  - reconstruction of a function from its projections
- Hounsfield 1972
  - patented the first CT scanner
- Cormack 1963 - 1964
  - contributionto mathematics of X-ray CT

Some hisrory

- Radon 1917
  - reconstruction of a function from its projections
- Hounsfield 1972
  - patented the first CT scanner
- Cormack 1963 - 1964
  - contributionto mathematics of X-ray CT
- Hounsfield and Cormack shared the 1979 Nobel Prize for medicine

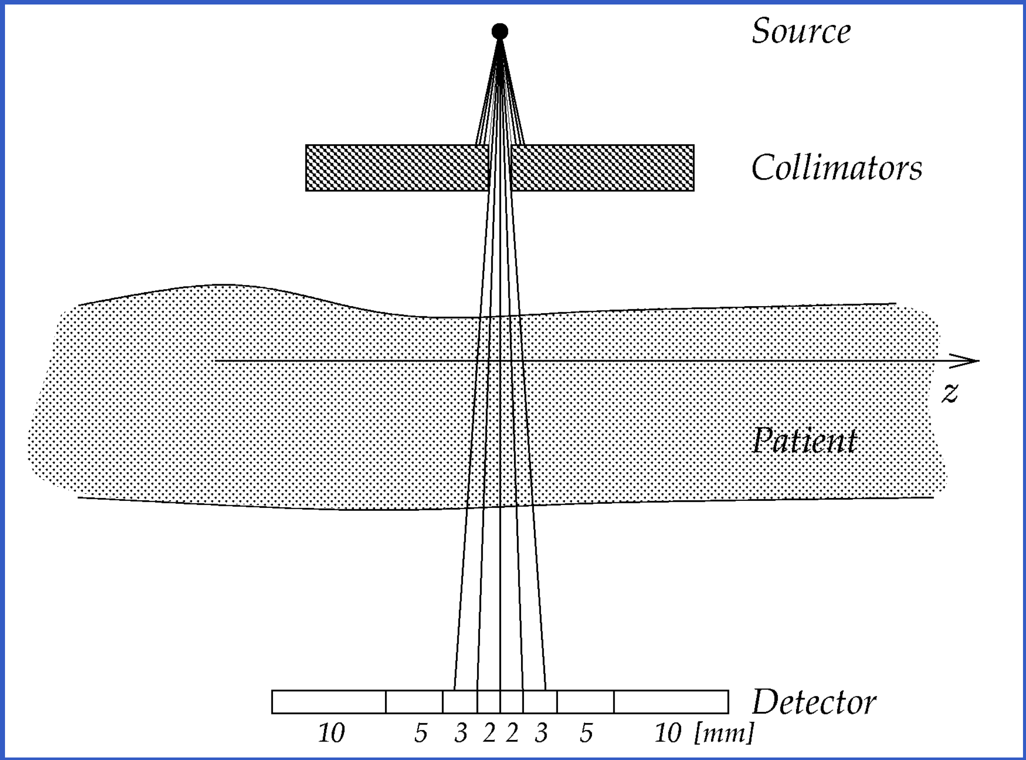
Some hisrory

- Radon 1917
  - reconstruction of a function from its projections
- Hounsfield 1972
  - patented the first CT scanner
- Cormack 1963 - 1964
  - contributionto mathematics of X-ray CT
- Hounsfield and Cormack shared the 1979 Nobel Prize for medicine
- since then the field is steadily evolving and givis rise to never ceasing flow of new algorithms

Principles of X-ray CT

Principles of X-ray CT

- X-ray traverses an object along stight line
- attenuated signals from various directions are recorded
- reconstruction algorithms are used to convert the projections



Projections

Let

$f(x, y)$	represents the density of a two-dimensional object
I_{in}	applied intensity form the X-ray source
I_{out}	the attenuated intensity of a ray as it propagates throught the object
	along the line $L = L(\theta, t)$

Projections

Let

$f(x,y)$	represents the density of a two-dimensional object
I_{in}	applied intensity form the X-ray source
I_{out}	the attenuated intensity of a ray as it propagates throught the object along the line $L = L(\theta, t)$

then

$$\int_L f(x,y)ds = -\ln \frac{I_{out}}{I_{in}} = P_\theta(t)$$

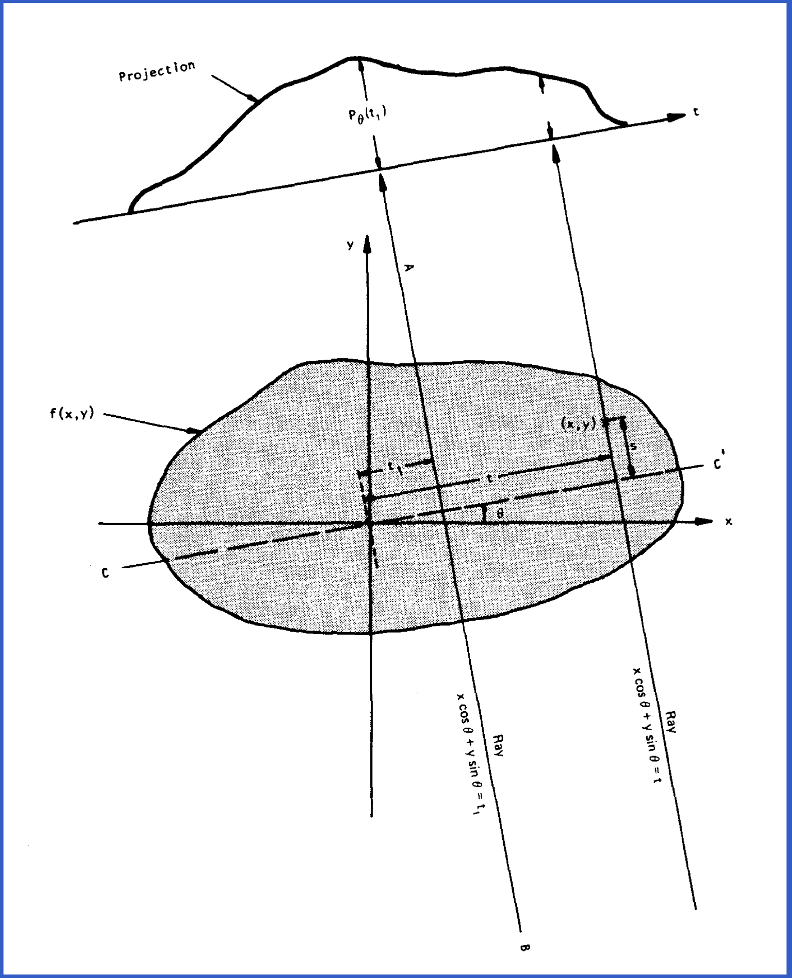
where

θ - projection angle

t - detector position

Projections

$$P_\theta(t) = \int_L f(x,y)ds$$



Projections

$$P_\theta(t) = \int_L f(x,y)ds$$

The line integral can be written as

Radon Transform

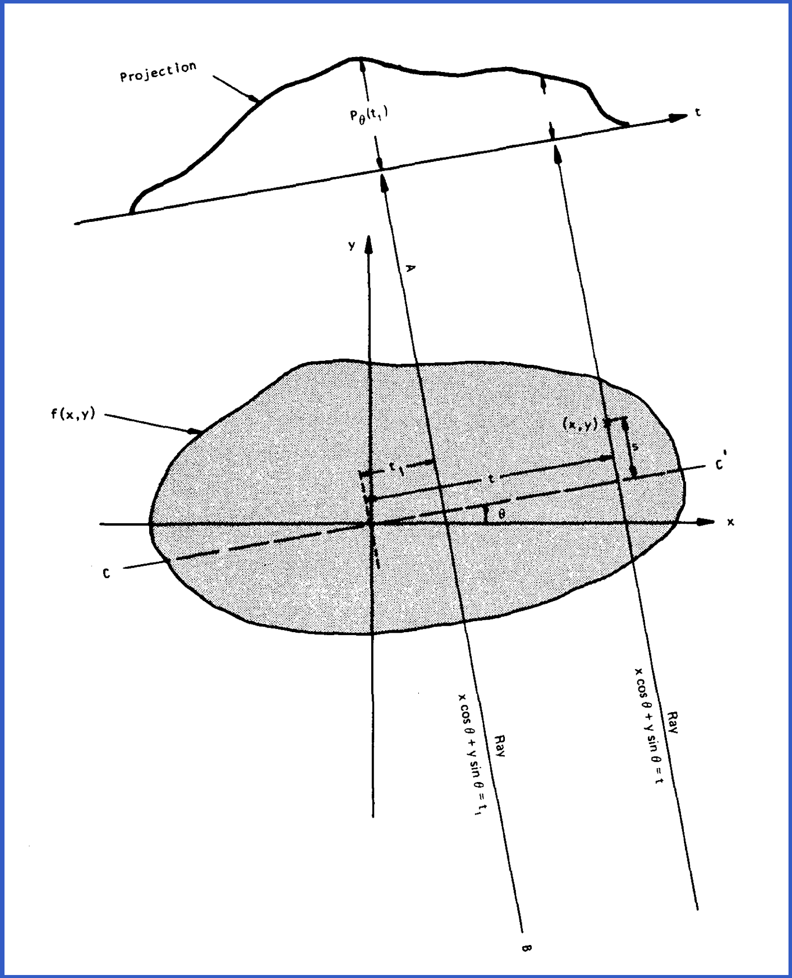
$$P_\theta(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(r(x,y,t))dxdy,$$

where

δ is the Dirac delta function and

$$r(x,y,t) = \cos\theta + y\sin\theta - t$$

(the equation of the X-ray)



Fourier Slice Theorem

Consider the two-dimensional FT of the object function

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

and the one-dimensional FT of the projection $P_\theta(t)$

$$S_\theta(w) = \int_{-\infty}^{\infty} P_\theta(t)e^{-j2\pi wt}dt$$

Fourier Slice Theorem

Consider the two-dimensional FT of the object function

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

and the one-dimensional FT of the projection $P_\theta(t)$

$$S_\theta(w) = \int_{-\infty}^{\infty} P_\theta(t)e^{-j2\pi wt}dt$$

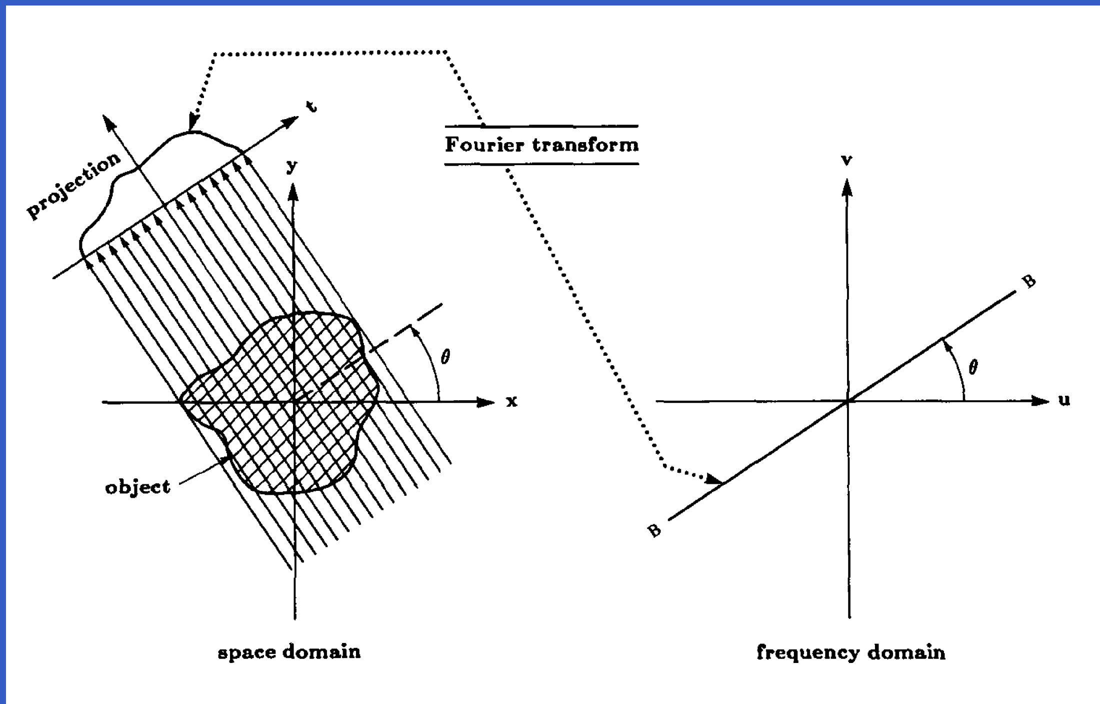
Simplest case, when $v = 0$ $(\theta = 0)$

$$\begin{aligned} F(u,0) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi ux}dxdy = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(x,y)dy\right]e^{-j2\pi ux}dx \\ &= \int_{-\infty}^{\infty}\left[\int_L f(x,y)ds\right]e^{-j2\pi ux}dx = \int_{-\infty}^{\infty} P_{\theta=0}(x)e^{-j2\pi ux}dx = S_{\theta=0}(u) \end{aligned}$$

Fourier Slice Theorem

In general for any anglr θ

$$S_\theta(w) = F(w, \theta) = F(w\cos\theta, w\sin\theta).$$



Fourier reconstruction algorithm

- Take projection of an object function at angles $\theta_1, \theta_2, \ldots, \theta_k$
- Fourier transform each projection to obtain the values of $F(u, v)$
- Recover the object function $f(x, y)$ by using the inverse FT

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

Fourier reconstruction algorithm

- Take projection of an object function at angles $\theta_1, \theta_2, \ldots, \theta_k$
- Fourier transform each projection to obtain the values of $F(u,v)$
- Recover the object function $f(x,y)$ by using the inverse FT

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

Note: infinite number of projections on an interval with lenght π allows exact reconstuction.

Fourier reconstruction algorithm

- Take projection of an object function at angles $\theta_1, \theta_2, \ldots, \theta_k$
- Fourier transform each projection to obtain the values of $F(u,v)$
- Recover the object function $f(x,y)$ by using the inverse FT

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

Discretising on the square $-A/2 < x < A/2; -A/2 < y < A/2$

$$f(x,y) \approx \frac{1}{A^2}\sum_{m=-N/2}^{N/2}\sum_{n=-N/2}^{N/2} F\left(\frac{m}{A},\frac{n}{A}\right)e^{j2\pi((m/A)x+(n/A)y)},$$

where$N$ is even integer.

The values of F(u,v) on a square grid are obtained from the available values along the radial lines by interpolation.

Reconstruction for Parallel Projection

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

$$= \int_{0}^{2\pi}\int_{0}^{\infty} F(w,\theta)e^{j2\pi w \overbrace{(x\cos\theta + y\sin\theta)}^{t}} wdwd\theta$$

Reconstruction for Parallel Projection

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

$$= \int_{0}^{2\pi}\int_{0}^{\infty} F(w,\theta)e^{j2\pi w \overbrace{(x\cos\theta + y\sin\theta)}^{t}} wdwd\theta$$

$$= \int_{0}^{\pi}\underbrace{\int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw}_{Q_\theta(t)}\,d\theta$$

Reconstruction for Parallel Projection

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

$$= \int_{0}^{2\pi}\int_{0}^{\infty} F(w,\theta)e^{j2\pi w \overbrace{(x\cos\theta + y\sin\theta)}^{t}} wdwd\theta$$

$$= \int_{0}^{\pi} \underbrace{\int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw}_{Q_\theta(t)}\, d\theta = \int_{0}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} P_\theta(\tau)e^{-j2\pi w\tau}d\tau|w|e^{j2\pi wt}dwd\theta$$

Reconstruction for Parallel Projection

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

$$= \int_{0}^{2\pi}\int_{0}^{\infty} F(w,\theta)e^{j2\pi w\overbrace{(x\cos\theta + y\sin\theta)}^{t}} wdwd\theta$$

$$= \int_{0}^{\pi}\underbrace{\int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw}_{Q_\theta(t)}\,d\theta = \int_{0}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} P_\theta(\tau)e^{-j2\pi w\tau}d\tau|w|e^{j2\pi wt}dwd\theta$$

$$= \int_{0}^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)\underbrace{\int_{-\infty}^{\infty}|w|e^{j2\pi w(t-\tau)}dw}_{h(t-\tau)}\,d\tau\right]d\theta = \int_{0}^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)h(t-\tau)d\tau\right]d\theta$$

Reconstruction for Parallel Projection

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

$$= \int_{0}^{2\pi}\int_{0}^{\infty} F(w,\theta)e^{j2\pi w\overbrace{(x\cos\theta + y\sin\theta)}^{t}} wdwd\theta$$

$$= \int_{0}^{\pi}\underbrace{\int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw}_{Q_\theta(t)}\,d\theta = \int_{0}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} P_\theta(\tau)e^{-j2\pi w\tau}d\tau|w|e^{j2\pi wt}dwd\theta$$

$$= \int_{0}^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)\underbrace{\int_{-\infty}^{\infty}|w|e^{j2\pi w(t-\tau)}dw}_{h(t-\tau)}\,d\tau\right]d\theta = \int_{0}^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)h(t-\tau)d\tau\right]d\theta$$

$$= \int_{0}^{\pi}(P_\theta * h)(x\cos\theta + y\sin\theta)d\theta = \int_{0}^{\pi} Q_\theta(x\cos\theta + y\sin\theta)d\theta$$

Reconstruction for Parallel Projection

$$
\begin{aligned}
f(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}\,dudv \\
&= \int_0^{2\pi}\int_0^{\infty} F(w,\theta)e^{j2\pi w\overbrace{(x\cos\theta+y\sin\theta)}^{t}}\,wdwd\theta \\
&= \int_0^{\pi}\underbrace{\int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw}_{Q_\theta(t)}\,d\theta = \int_0^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} P_\theta(\tau)e^{-j2\pi w\tau}d\tau|w|e^{j2\pi wt}dwd\theta \\
&= \int_0^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)\underbrace{\int_{-\infty}^{\infty}|w|e^{j2\pi w(t-\tau)}dw}_{h(t-\tau)}\,d\tau\right]d\theta = \int_0^{\pi}\left[\int_{-\infty}^{\infty} P_\theta(\tau)h(t-\tau)d\tau\right]d\theta \\
&= \underbrace{\int_0^{\pi}\underbrace{(P_\theta * h)}_{\text{Filtering}}(x\cos\theta+y\sin\theta)d\theta}_{\text{Backprojection}} = \int_0^{\pi} Q_\theta(x\cos\theta+y\sin\theta)d\theta
\end{aligned}
$$

The FBP algorithm

Step 1: Filtering

$$Q_\theta(t) = \int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw$$

The FBP algorithm

Step 1: Filtering

$$Q_\theta(t) = \int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw$$

Use badlimited FT to compute $S_\theta(w)$ i.e.

$$S_\theta(w) \approx S_\theta\left(m\frac{2W}{N}\right) = \frac{1}{2W}\sum_{k=-N/2}^{N/2-1} P_\theta\left(\frac{k}{2W}\right)e^{-j2\pi(mk/N)}, \quad m = -\frac{N}{2}, \ldots, \frac{N}{2},$$

where W is a frequency band limit and N is some (large) number corresponding to the number of the sampling points. (FFT)

The FBP algorithm

Step 1: Filtering

$$Q_\theta(t) = \int_{-\infty}^{\infty} S_\theta(w)|w|e^{j2\pi wt}dw$$

Use badlimited FT to compute $S_\theta(w)$ i.e.

$$S_\theta(w) \approx S_\theta\left(m\frac{2W}{N}\right) = \frac{1}{2W}\sum_{k=-N/2}^{N/2-1} P_\theta\left(\frac{k}{2W}\right)e^{-j2\pi(mk/N)}, \quad m = -\frac{N}{2},\ldots,\frac{N}{2},$$

where W is a frequency band limit and N is some (large) number corresponding to the number of the sampling points. (FFT)
Therefore

$$Q_\theta\left(\frac{k}{2W}\right) \approx \left(\frac{2W}{N}\right)\sum_{m=-N/2}^{N/2} S_\theta\left(m\frac{2W}{N}\right)\left|m\frac{2W}{N}\right|e^{j2\pi(mk/N)}, \quad k = -\frac{N}{2},\ldots,\frac{N}{2}.$$

Inverse DFT of the product $S_\theta(m(2W/N))$ and $|m(2W/N)|$.

The FBP algorithm

Step 1: Filtering

Superior results are obtained by deemphasizing the high frequencies with the Hamming window function H.

$$Q_\theta\left(\frac{k}{2W}\right) \approx \left(\frac{2W}{N}\right) \sum_{m=-N/2}^{N/2} S_\theta\left(m\frac{2W}{N}\right)\left|m\frac{2W}{N}\right| H\left(m\frac{2W}{N}\right) e^{j2\pi(mk/N)}$$

The FBP algorithm

Step 1: Filtering

Superior results are obtained by deemphasizing the high frequencies with the Hamming window function H.

$$Q_\theta\left(\frac{k}{2W}\right) \approx \left(\frac{2W}{N}\right) P_\theta\left(\frac{k}{2W}\right) * \phi\left(\frac{k}{2W}\right),$$

where $\phi(k/"W)$ is the inverse DFT of $|m(2W/N)|H(m(2W/N))$.

The FBP algorithm

Step 1: Filtering

Superior results are obtained by deemphasizing the high frequencies with the Hamming window function H.

$$Q_\theta\left(\frac{k}{2W}\right) \approx \left(\frac{2W}{N}\right) P_\theta\left(\frac{k}{2W}\right) * \phi\left(\frac{k}{2W}\right),$$

where $\phi(k/"W)$ is the inverse DFT of $|m(2W/N)|H(m(2W/N))$.

Step 2: Backprojection

Reconstruc $f(x, y)$ by the formula

$$f(x,y) \approx \frac{\pi}{K} \sum_{i=1}^{K} Q_{\theta_i}(x\cos\theta_i + y\sin\theta_i)$$

where K is the number of the projection angles θ_i.

Note The values of Q_θ in Step 2 can be approxiamted from the values obtained in Step 1 by interpolation (linear).

The FBP algorithm

Step 1: Filtering

Superior results are obtained by deemphasizing the high frequencies with the Hamming window function H.

$$Q_\theta\left(\frac{k}{2W}\right) \approx \left(\frac{2W}{N}\right) P_\theta\left(\frac{k}{2W}\right) * \phi\left(\frac{k}{2W}\right),$$

where $\phi(k/"W)$ is the inverse DFT of $|m(2W/N)|H(m(2W/N))$.

Step 2: Backprojection

Reconstruc $f(x,y)$ by the formula

$$f(x,y) \approx \frac{\pi}{K}\sum_{i=1}^{K} Q_{\theta_i}(x\cos\theta_i + y\sin\theta_i)$$

where K is the number of the projection angles θ_i.

Note The values of Q_θ in Step 2 can be approxiamted from the values obtained in Step 1 by interpolation (linear).

Computational cost: Backprojection summation —> $\mathcal{O}(N^3)$

Fast backprojections (Nilsson'96) —> $\mathcal{O}(N^2 \log N)$

Reconstruction for Fan Projections

Equiangular Rays

Let $R_\beta(\gamma)$ be a fan projection

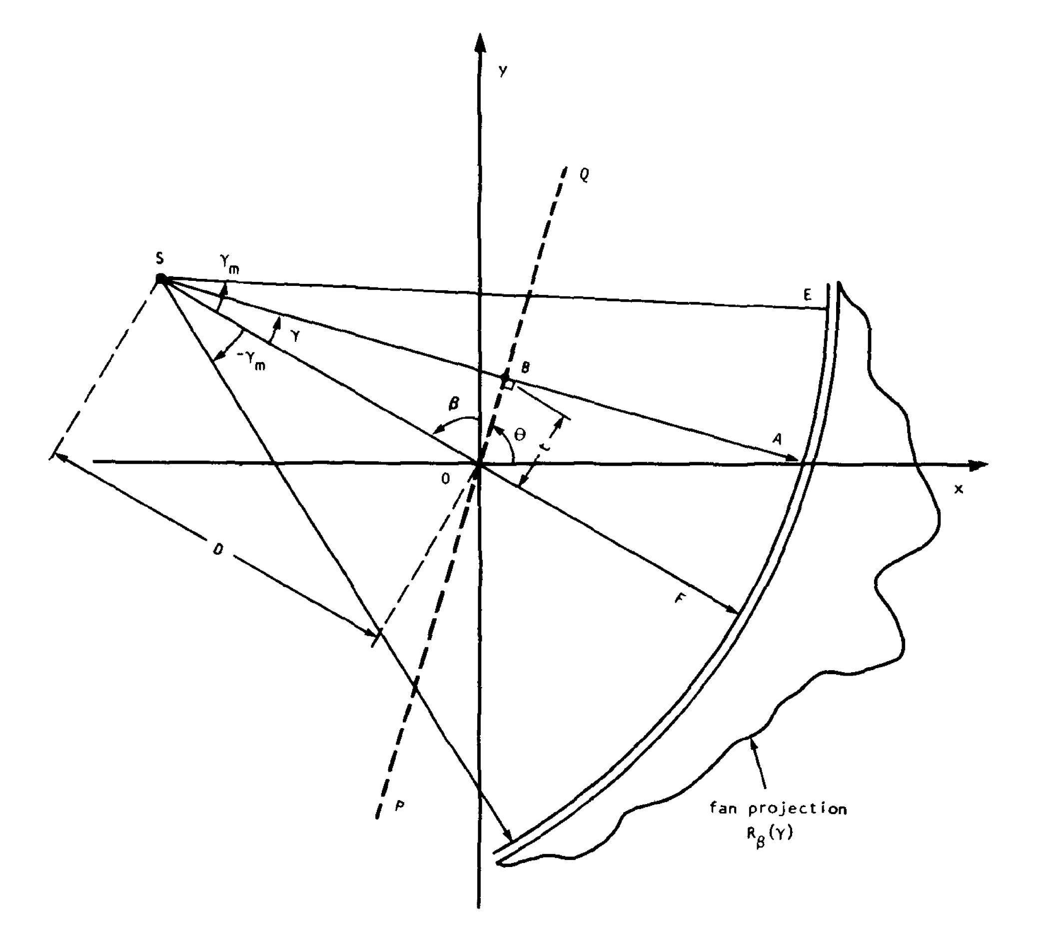
β is the angle between the source and a reference axis.

γ gives the location of a ray

$D =$|SO| source to origin dist.

L - source to point (x, y) dist.
In polar coordinates $[(r, \phi)]$

$L = \sqrt{D + (r\sin\eta)^2 + (r\cos\eta)^2}$,
where $\eta = \beta - \phi$.



Reconstruction for Fan Projections

Equiangular Rays

$$f(x,y) = \int_0^{\pi} \int_{-\infty}^{\infty} P_\theta(t) h(x\cos\theta + y\sin\theta - t) dt d\theta$$

$$f(r,\phi) = \frac{1}{2} \int_0^{2\pi} \int_{-\infty}^{\infty} P_\theta(t) h(r\cos(\theta - \phi) - t) dt d\theta$$

Reconstruction for Fan Projections

Equiangular Rays

$$
\begin{aligned}
f(x,y) &= \int_0^{\pi}\int_{-\infty}^{\infty} P_\theta(t)h(x\cos\theta + y\sin\theta - t)dtd\theta \\
f(r,\phi) &= \frac{1}{2}\int_0^{2\pi}\int_{-\infty}^{\infty} P_\theta(t)h(r\cos(\theta-\phi) - t)dtd\theta \\
&= \dots \\
&= \int_0^{2\pi} \frac{1}{L^2} Q_\beta(\gamma')d\beta,
\end{aligned}
$$

where

$$
\begin{aligned}
Q_\beta(\gamma) &= R'_\beta(\gamma) * g(\gamma) \\
R'_\beta(\gamma) &= R_\beta(\gamma)\cdot D\cdot\cos\gamma \\
g(\gamma) &= \frac{1}{2}\left(\frac{\gamma}{\sin\gamma}\right)^2 h(\gamma)
\end{aligned}
$$

Weighted Backprojection Algorithm

Equiangular Rays

Let
- α is the sampling interval of the projection
- β_i are the angles at which projections are taken

Step 1 For each $R_{\beta_i}(n\alpha),\ n \in \mathbb{N}$, generate the corresponding modified projection

$$R'_{\beta_i}(n\alpha) = R_{\beta}(n\alpha) \cdot D \cdot \cos n\alpha$$

Weighted Backprojection Algorithm

Equiangular Rays

Let
- α is the sampling interval of the projection
- β_i are the angles at which projections are taken

Step 1 For each $R_{\beta_i}(n\alpha),\ n \in \mathbb{N}$, generate the corresponding modified projection

$$R'_{\beta_i}(n\alpha) = R_{\beta}(n\alpha) \cdot D \cdot \cos n\alpha$$

Step 2 Convolve $R'_{\beta_i}(n\alpha)$ with $g(n\alpha) = \frac{1}{2}\left(\frac{n\alpha}{\sin n\alpha}\right)^2 h(n\alpha)$ to generate the corresponding filtered projection (FFT)

$$Q_{\beta_i}(n\alpha) = R'_{\beta_i}(n\alpha) * g(n\alpha)$$

Weighted Backprojection Algorithm

Equiangular Rays

Let
- α is the sampling interval of the projection
- β_i are the angles at which projections are taken

Step 1 For each $R_{\beta_i}(n\alpha),\ n \in \mathbb{N}$, generate the corresponding modified projection

$$R'_{\beta_i}(n\alpha) = R_\beta(n\alpha) \cdot D \cdot \cos n\alpha$$

Step 2 Convolve $R'_{\beta_i}(n\alpha)$ with $g(n\alpha) = \frac{1}{2}\left(\frac{n\alpha}{\sin n\alpha}\right)^2 h(n\alpha)$ to generate the corresponding filtered projection (FFT)

$$Q_{\beta_i}(n\alpha) = R'_{\beta_i}(n\alpha) * g(n\alpha) * k(n\alpha),$$

where $k(n\alpha)$ is a smoothing filter.

Weighted Backprojection Algorithm

Equiangular Rays

Let
- α is the sampling interval of the projection
- β_i are the angles at which projections are taken

Step 1 For each $R_{\beta_i}(n\alpha),\ n \in \mathbb{N}$, generate the corresponding modified projection

$$R'_{\beta_i}(n\alpha) = R_\beta(n\alpha) \cdot D \cdot \cos n\alpha$$

Step 2 Convolve $R'_{\beta_i}(n\alpha)$ with $g(n\alpha) = \frac{1}{2}\left(\frac{n\alpha}{\sin n\alpha}\right)^2 h(n\alpha)$ to generate the corresponding filtered projection (FFT)

$$Q_{\beta_i}(n\alpha) = R'_{\beta_i}(n\alpha) * g(n\alpha) * k(n\alpha),$$

where $k(n\alpha)$ is a smoothing filter.

Step 3 Perform a weighted backprojection

$$f(x,y) \approx \Delta\beta \sum_{i=1} K \frac{1}{L^2(x,y,\beta_1)} Q_{\beta_i}(\gamma'),$$

where γ' is the angle of a ray that passes through the point (x,y) and $\Delta\beta = 2\pi/K$.

Reconstruction for Fan Projections

Equally Spaced Detectors

Let $R_\beta(s)$ be a fan projection

β is the angle between the source and a reference axis.

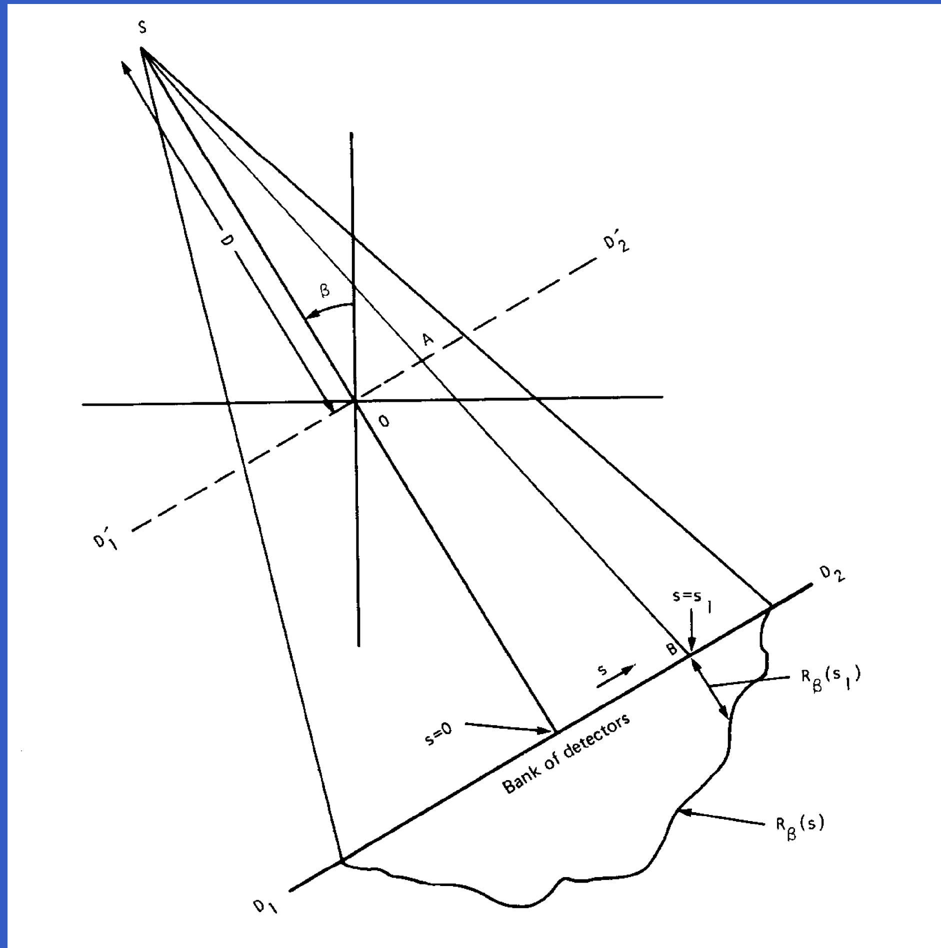
s is the distance along the detector

$D =$|SO| source to origin dist.

U - the dist. source to projection of a pixel on the central ray

In polar coordinates $[(r,\phi)]$

$$U = \frac{D + r\sin(\beta-\phi)}{D}$$



Reconstruction for Fan Projections

Equally Spaced Detectors

$$
\begin{aligned}
f(x,y) &= \int_0^{\pi}\int_{-\infty}^{\infty} P_\theta(t)h(x\cos\theta + y\sin\theta - t)\,dt\,d\theta \\
f(r,\phi) &= \frac{1}{2}\int_0^{2\pi}\int_{-\infty}^{\infty} P_\theta(t)h(r\cos(\theta-\phi) - t)\,dt\,d\theta \\
&= \ldots \\
&= \int_0^{2\pi} \frac{1}{U^2} Q_\beta(s')\,d\beta,
\end{aligned}
$$

where

$$
\begin{aligned}
Q_\beta(s) &= R'_\beta(s) * g(s) \\
R'_\beta(s) &= R_\beta(s)\cdot\frac{D}{\sqrt{D^2+s^2}} \\
g(s) &= \frac{1}{2}h(s)
\end{aligned}
$$

Algebraic reconstruction techniques

Advantages

- easy to handle different rays geometry
- more adaptable to missing data - better image quality
- metal artifacts are reduced
- less radiation dose

Algebraic reconstruction techniques

Advantages

- easy to handle different rays geometry
- more adaptable to missing data - better image quality
- metal artifacts are reduced
- less radiation dose

Disadvantage

Higher computational cost!

Image and Projection Representation

The idea: Assume that the cross section consists of array of unknowns

Image and Projection Representation

The idea: Assume that the cross section consists of array of unknowns

N - total number of cells

$f_j = const$ is the value of $f(x, y)$
in the j-th cell

p_i - the i-th line integral *ray-sum*

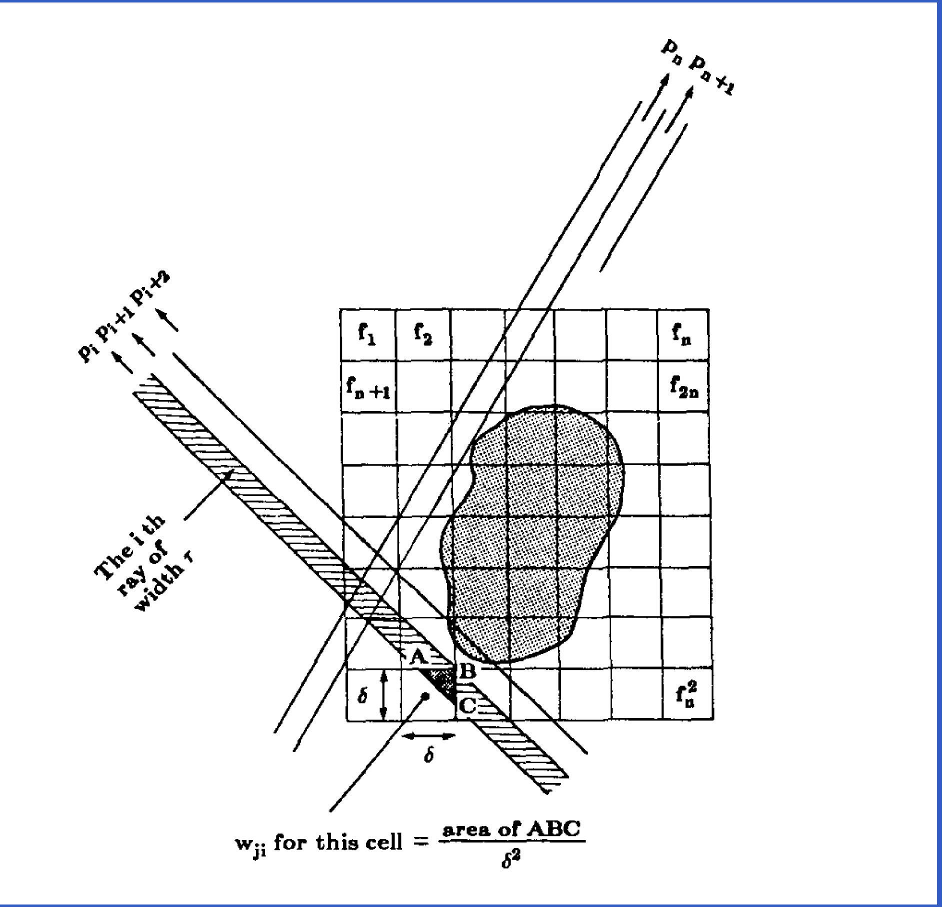


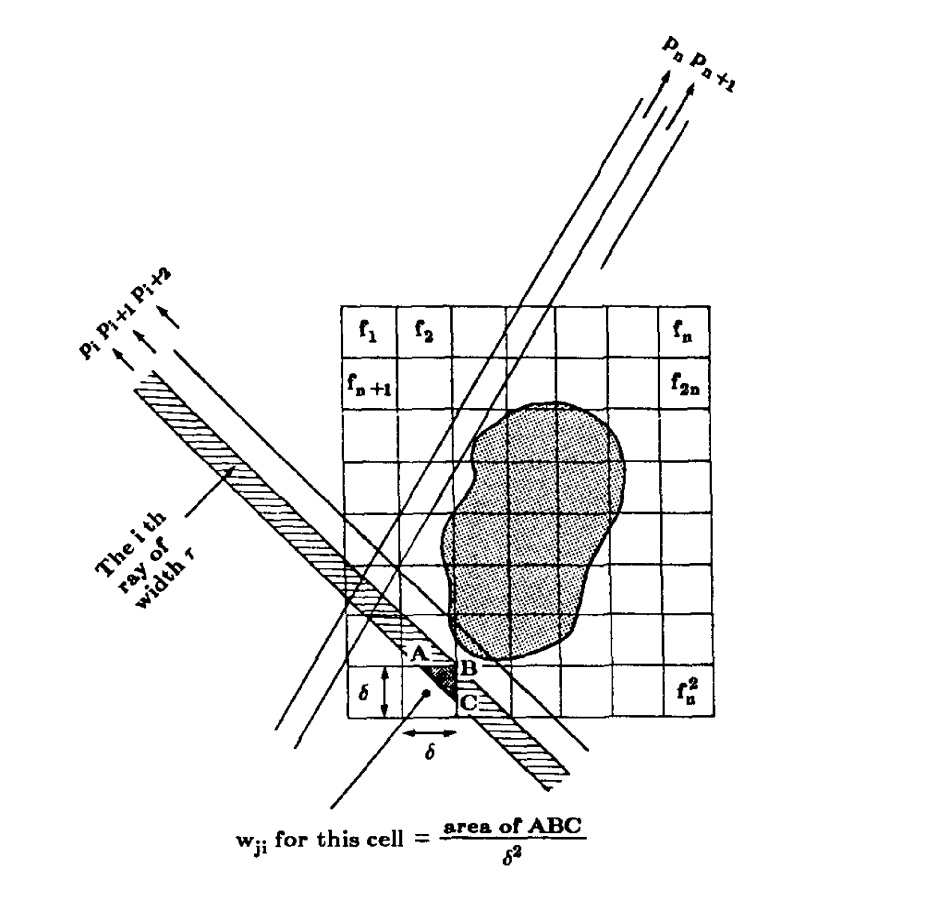
Image and Projection Representation

The idea: Assume that the cross section consists of array of unknowns

N - total number of cells

$f_j = const$ is the value of $f(x, y)$ in the j-th cell

p_i - the i-th line integral *ray-sum*



$$\sum_{j=1}^{N} w_{ij} f_j = p_i, \quad i = 1, 2, \ldots, M$$

where

M is the total number of rays

w_{ij} represents the contribution of the j-th cell to the i-th ray integral

Projection methods

Therefore we have to solve the following linear system of equations

$$
\begin{aligned}
w_{11} f_1 + w_{12} f_2 + \cdots + w_{1N} f_N &= p_1 \\
w_{21} f_1 + w_{22} f_2 + \cdots + w_{2N} f_N &= p_2 \\
\vdots \\
w_{M1} f_1 + w_{M2} f_2 + \cdots + w_{MN} f_N &= p_M
\end{aligned}
$$

for large values of M and N.

Projection methods

Therefore we have to solve the following linear system of equations

$$
\begin{aligned}
w_{11} f_1 + w_{12} f_2 + \cdots + w_{1N} f_N &= p_1 \\
w_{21} f_1 + w_{22} f_2 + \cdots + w_{2N} f_N &= p_2 \\
&\vdots \\
w_{M1} f_1 + w_{M2} f_2 + \cdots + w_{MN} f_N &= p_M
\end{aligned}
$$

for large values of M and N.

Use iterative methods - projestions (Kaczmarz'37, Tanabe'71)

Let $f^{(0)} = \left(f_1^{(0)}, f_2^{(0)}, \dots f_N^{(0)}\right)$ is the initial guess

$$f^{(i)} = f^{(i-1)} - \frac{\left(f^{(i-1)} \cdot w_i - p_i\right)}{w_i \cdot w_i} w_i \quad i = 1, 2, \dots,$$

where $w_i = (w_{i1}, w_{i,2}, \dots w_{iN})$

Projection methods

Theorem (Tanabe'71)
If there exists a unique solution $f^{(s)}$ to the system of equations, then

$$\lim_{k\to\infty} f^{(kM)} = f^{(s)}$$

Projection methods

Theorem (Tanabe'71)
If there exists a unique solution $f^{(s)}$ to the system of equations, then

$$\lim_{k\to\infty} f^{(kM)} = f^{(s)}$$

The rate of convergence depends of the angles between the hyperplanes

- Full orthogonalization is computationally not feasible
- Pairwise orthogonalization scheme (Ramakrishnan'79)
- Carefully choose the order of the hyperplanes (Hounsfield'72)

Projection methods

Theorem (Tanabe'71)
If there exists a unique solution $f^{(s)}$ to the system of equations, then

$$\lim_{k \to \infty} f^{(kM)} = f^{(s)}$$

The rate of convergence depends of the angles between the hyperplanes

- Full orthogonalization is computationally not feasible

- Pairwise orthogonalization scheme (Ramakrishnan'79)

- Carefully choose the order of the hyperplanes (Hounsfield'72)

• When $M > N$ the process oscillate in the neighborhood of the intersections of the hyperplanes

• When $M < N$ the process converges to a solution $f'^{(s)}$, such that $|f^{(0)} - f'^{(s)}|$ is minimized.

Projection methods

Consider again

$$f^{(i)} = f^{(i-1)} - \frac{\left(f^{(i-1)} \cdot w_i - p_i\right)}{w_i \cdot w_i} w_i \quad i = 1, 2, \ldots,$$

It can also be written as

$$f_j^{(i)} = f_j^{(i-1)} + \frac{p_i - q_i}{\sum_{k=1}^N w_{ik}^2} w_{ij} \quad j = 1, 2, \ldots, N, \quad i = 1, 2, \ldots,$$

where

$$q_i = f^{(i-1)} \cdot w_i = \sum_{k=1}^N f_k^{(i-1)} w_{ik}.$$

Therefore

$$f_j^{(i)} = f_j^{(i-1)} + \Delta f_j^{(i)},$$

where

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{\sum_{k=1}^N w_{ik}^2} w_{ij}.$$

Projection methods

Consider again

$$f^{(i)} = f^{(i-1)} - \frac{\left(f^{(i-1)} \cdot w_i - p_i\right)}{w_i \cdot w_i} w_i \quad i = 1, 2, \ldots,$$

It can also be written as

$$f_j^{(i)} = f_j^{(i-1)} + \frac{p_i - q_i}{\sum_{k=1}^N w_{ik}^2} w_{ij} \quad j = 1, 2, \ldots, N, \quad i = 1, 2, \ldots,$$

where

$$q_i = f^{(i-1)} \cdot w_i = \sum_{k=1}^N f_k^{(i-1)} w_{ik}.$$

Therefore

$$f_j^{(i)} = f_j^{(i-1)} + \Delta f_j^{(i)},$$

where

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{\sum_{k=1}^N w_{ik}^2} w_{ij}.$$

The main computational diffuculty is in the calculation, storage and retrieval of the weight coefficients w_{ij}.

Algebraic Reconstruction Tech.(ART)

- Replace w_{ij} by 0 or 1, depending wether the center of the j-th cell is within the i-th ray. Thus we can use

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{N_i}$$

for all cells whos centers are within the i-th ray.
$N_i = \sum_{k=1}^{N} w_{ik}^2$ is the number of cells whose centers are within the i-th ray.

Algebraic Reconstruction Tech.(ART)

- Replace w_{ij} by 0 or 1, depending wether the center of the j-th cell is within the i-th ray. Thus we can use

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{N_i}$$

for all cells whos centers are within the i-th ray.
$N_i = \sum_{k=1}^{N} w_{ik}^2$ is the number of cells whose centers are within the i-th ray.

- Superior approxiamtion is

$$\Delta f_j^{(i)} = \frac{p_i}{L_i} - \frac{q_i}{N_i},$$

where L_i is the lenght of the i-th ray trought the reconstruction region.

Algebraic Reconstruction Tech.(ART)

- Replace w_{ij} by 0 or 1, depending wether the center of the j-th cell is within the i-th ray. Thus we can use

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{N_i}$$

for all cells whos centers are within the i-th ray.
$N_i = \sum_{k=1}^{N} w_{ik}^2$ is the number of cells whose centers are within the i-th ray.

- Superior approxiamtion is

$$\Delta f_j^{(i)} = \frac{p_i}{L_i} - \frac{q_i}{N_i},$$

where L_i is the lenght of the i-th ray trought the reconstruction region.

- ART suffer from *salt and pepper noise*, caused by the approximations used for w_{ij}.

Algebraic Reconstruction Tech.(ART)

- Replace w_{ij} by 0 or 1, depending wether the center of the j-th cell is within the i-th ray. Thus we can use

$$\Delta f_j^{(i)} = \frac{p_i - q_i}{N_i}$$

for all cells whos centers are within the i-th ray.
$N_i = \sum_{k=1}^{N} w_{ik}^2$ is the number of cells whose centers are within the i-th ray.

- Superior approxiamtion is

$$\Delta f_j^{(i)} = \frac{p_i}{L_i} - \frac{q_i}{N_i},$$

where L_i is the lenght of the i-th ray trought the reconstruction region.

- ART suffer from *salt and pepper noise*, caused by the approximations used for w_{ij}.

- It is possible to reduce the noce by

relaxation (i.e. update a pixel by $\lambda_i \Delta f_j^{(i)},\ \lambda < 1$)

SIRT - change the cell values after going through all of the equations

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals

Insted of solving

$$p_i = \sum_{j=1}^{N} w_{ij} f_j, \quad i = 1, 2, \ldots, M.$$

We use again the Radon Transform

$$p_i = R_i f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)\delta(r_i(x,y))dxdy,$$

where $r_i(x,y) = 0$ is the equation of the i-th ray and R_i is the projection operator.

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals

Insted of solving

$$p_i = \sum_{j=1}^{N} w_{ij} f_j, \quad i = 1, 2, \ldots, M.$$

We use again the Radon Transform

$$p_i = R_i f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)\delta(r_i(x,y))dxdy,$$

where $r_i(x,y) = 0$ is the equation of the $i^{\text{-th}}$ ray and R_i is the projection operator.

Let $\{b_j(x,y)\}_{j=1}^{N}$ be basis functions and let

$$f(x,y) \approx \widetilde{f}(x,y) = \sum_{j=1}^{N} g_j b_j(x,y)$$

then

$$p_i = R_i f(x,y) \approx R_i \widetilde{f}(x,y) = \sum_{j=1}^{N} g_j a_{ij}$$

wher $a_{ij} = R_i b_j(x,y)$.

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals

Insted of solving

$$p_i = \sum_{j=1}^{N} w_{ij} f_j, \quad i = 1, 2, \ldots, M.$$

We solve

$$p_i = \sum_{j=1}^{N} g_j a_{ij}, \quad i = 1, 2, \ldots, M.$$

- The two systems are the same for the *pixel basis*

$$b_j(x,y) = \begin{cases} 1 & \text{inside the } j\text{-th pixel} \\ 0 & \text{everywhere else} \end{cases}$$

- Superior reconstructions are obtained by using bilinear elements

pyramid shaped basis

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals
  bilinear elements insted of the traditional pixel basis

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals
  bilinear elements insted of the traditional pixel basis
- Correction terms are simultaneously applied to all the rays
  as for the SIRT methods

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals
  bilinear elements insted of the traditional pixel basis
- Correction terms are simultaneously applied to all the rays
  as for the SIRT methods
- Hamming window
  emphasizes the corrections applied near the middle of a ray relative to those applied near the ends.

SART (Simultaneous ART)

First reported by Anderson and Kak '84

- Reduce the error in the approximation of the ray integrals
  bilinear elements insted of the traditional pixel basis
- Correction terms are simultaneously applied to all the rays
  as for the SIRT methods
- Hamming window
  emphasizes the corrections applied near the middle of a ray relative to those applied near the ends.

The overall SART iterative scheme is

$$g_j^{(k+1)} = g_j^{(k)} + \frac{\lambda_k}{\sum_{i=1}^{M} a_{ij}} \sum_{i=1}^{M} a_{ij} \frac{p_i - a_i \cdot g^{(k)}}{\sum_{j=1}^{N} a_{ij}}, \quad j = 1, 2, \ldots, N,$$

where a_i is the i-th row of A and $\lambda_k > 0$ are relaxation coefficients.

More iterative algorithms

Censor and Elfving'02

The idea: Use generalized projections and diagonal weighting matrices to reflect the sparsity of the A matrix.

Definition: A family $\{G_i\}_{i=1}^{M}$ of diagonal $N \times N$ matrices with diagonal elements $g_{ij} > 0$ is called *Sparsity Pattern Oriented* (SPO) with respect to $M \times N$ matix A if:

1) $\sum_{i=1}^{M} G_i = I$ 2) $g_{ij} = 0$ iff $a_{ij} = 0$ for every $i = 1, 2, \ldots, M$.

More iterative algorithms

Censor and Elfving'02

The idea: Use generalized projections and diagonal weighting matrices to reflect the sparsity of the A matrix.

Definition: A family $\{G_i\}_{i=1}^{M}$ of diagonal $N \times N$ matrices with diagonal elements $g_{ij} > 0$ is called *Sparsity Pattern Oriented* (SPO) with respect to $M \times N$ matix A if:

1) $\sum_{i=1}^{M} G_i = I$ 2) $g_{ij} = 0$ iff $a_{ij} = 0$ for every $i = 1, 2, \ldots, M$.

Diagonal Weighting (DWE)

$$g_j^{(k+1)} = g_j^{(k)} + \lambda_k \sum_{\substack{i=1 \\ g_{ij} \neq 0}}^{M} a_{ij} \frac{p_i - a_i \cdot g^{(k)}}{\sum_{\substack{l=1 \\ g_{il} \neq 0}}^{N} \frac{a_{il}^2}{g_{il}}}, \quad j = 1, 2, \ldots, N.$$

More iterative algorithms

Censor and Elfving'02

The idea: Use generalized projections and diagonal weighting matrices to reflect the sparsity of the A matrix.

Definition: A family $\{G_i\}_{i=1}^{M}$ of diagonal $N \times N$ matrices with diagonal elements $g_{ij} > 0$ is called *Sparsity Pattern Oriented* (SPO) with respect to $M \times N$ matix A if:

1) $\sum_{i=1}^{M} G_i = I$ 2) $g_{ij} = 0$ iff $a_{ij} = 0$ for every $i = 1, 2, \ldots, M$.

Special case

$$g_{ij} = \begin{cases} \frac{1}{s_j} & \text{if } a_{ij} \neq 0 \\ 0 & \text{if } a_{ij} = 0 \end{cases}$$

where s_j is the number of the nonzero elements in column j of A.

Component Averaging (CAV)

$$g_j^{(k+1)} = g_j^{(k)} + \lambda_k \sum_{i=1}^{M} a_{ij} \frac{p_i - a_i \cdot g^{(k)}}{\sum_{l=1}^{N} s_l a_{il}^2}, \quad j = 1, 2, \ldots, N.$$

More iterative algorithms

Censor and Elfving'02

The idea: Use generalized projections and diagonal weighting matrices to reflect the sparsity of the A matrix.

Definition: A family $\{G_i\}_{i=1}^{M}$ of diagonal $N \times N$ matrices with diagonal elements $g_{ij} > 0$ is called *Sparsity Pattern Oriented* (SPO) with respect to $M \times N$ matix A if:

1) $\sum_{i=1}^{M} G_i = I$ 2) $g_{ij} = 0$ iff $a_{ij} = 0$ for every $i = 1, 2, \ldots, M$.

Special case

$$g_{ij} = \begin{cases} \frac{1}{s_j} & \text{if } a_{ij} \neq 0 \\ 0 & \text{if } a_{ij} = 0 \end{cases}$$

where s_j is the number of the nonzero elements in column j of A.

Component Averaging (CAV)

$$g_j^{(k+1)} = g_j^{(k)} + \lambda_k \sum_{i=1}^{M} a_{ij} \frac{p_i - a_i \cdot g^{(k)}}{\sum_{l=1}^{N} s_l a_{il}^2}, \quad j = 1, 2, \ldots, N.$$

- Jiang and Wang'03 ART, SART, CAV, DWE are special cases of the Landweber method

$$g^{(k+1)} = g^{(k)} + \lambda_k V^{-1} A^* W (p - A g^{(k)})$$

3D Reconstructions

- Slice by Slice

Disadvantages

- low speed
- discrete nature
- higher radiation dose

3D Reconstructions

- Slice by Slice

  Disadvantages

  - low speed

  - discrete nature

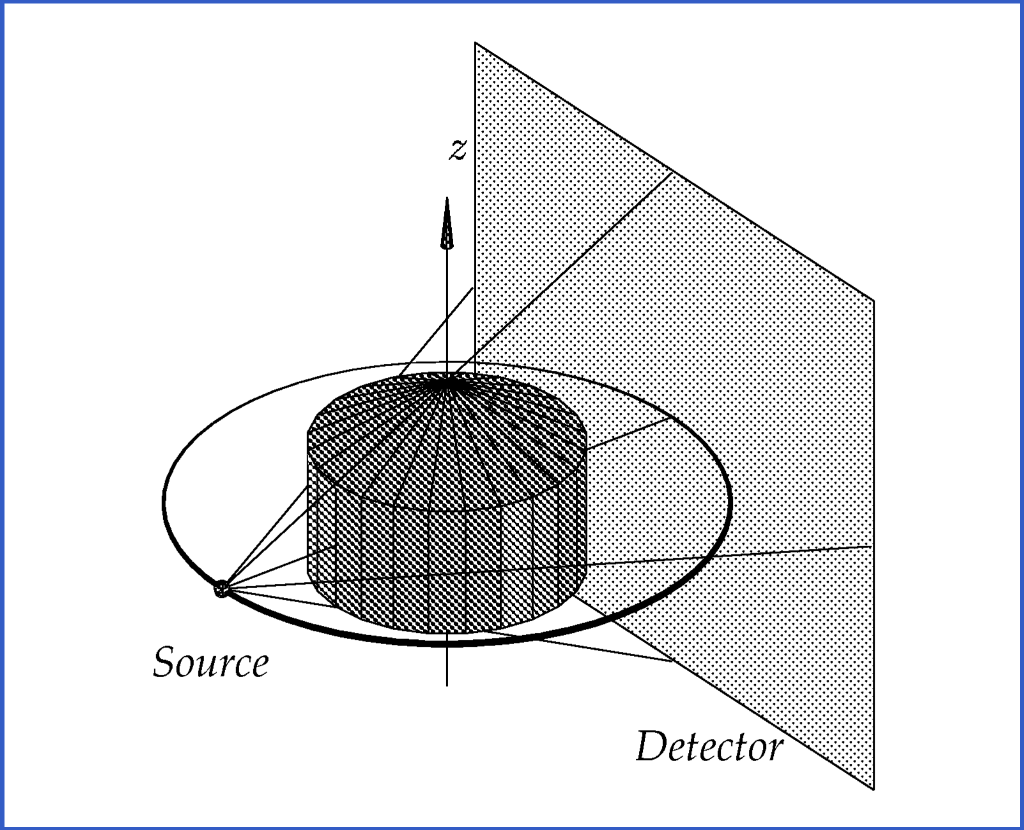
  - higher radiation dose

- Cone-Beam Reconstructions

  - Circular

  - Helical

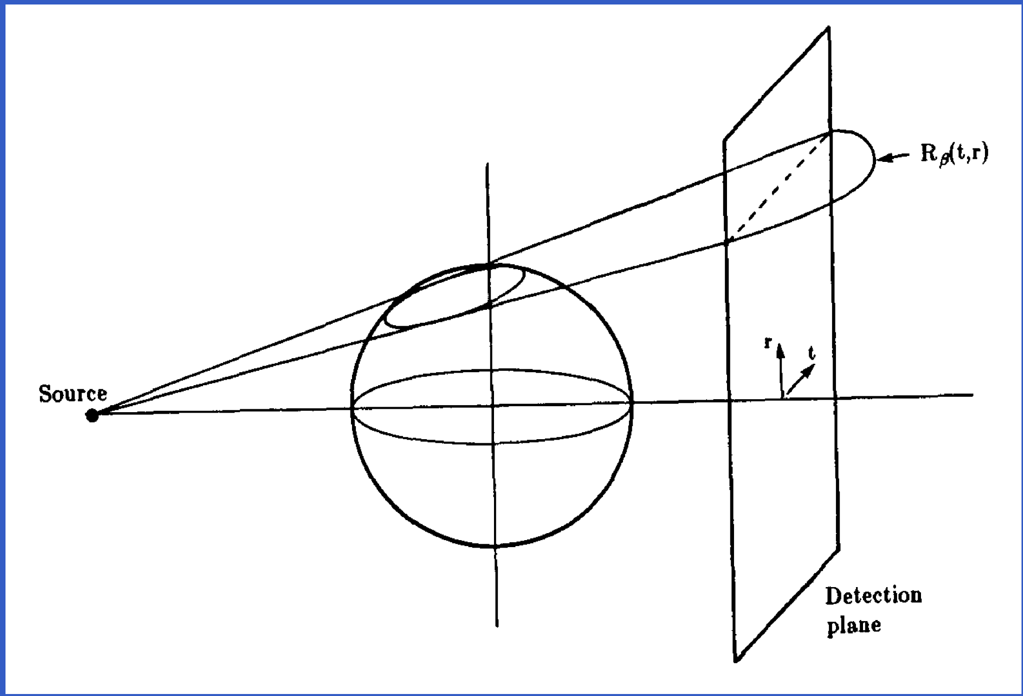
Circular Cone-Beam Reconstructions



Circular Cone-Beam Reconstructions

The Idea: Apply fan-beam FBP to each plane in the cone

Feldkamp, Davis, Kress ’84 (FDK)



Circular Cone-Beam Reconstructions

The Idea: Apply fan-beam FBP to
each plane in the cone

Feldkamp, Davis, Kress '84 (FDK)

A ray from the cone is described by

$$t = x\cos\theta + y\sin\theta$$
$$r = -(-x\sin\theta + y\cos\theta)\sin\gamma + z\cos\gamma$$

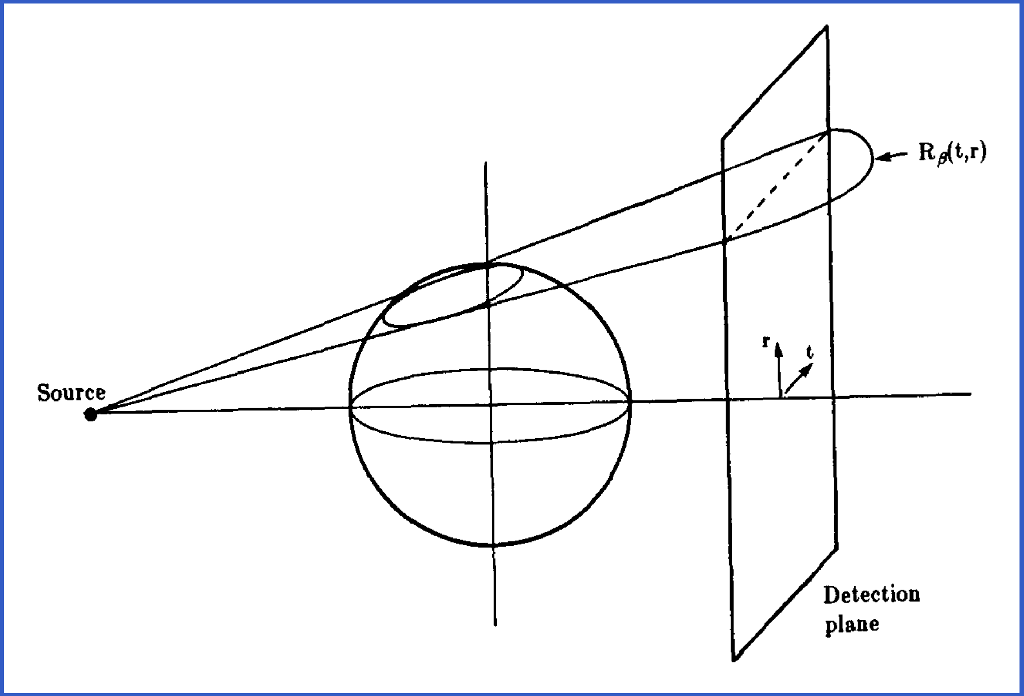
where

(t, θ) locate a ray in a fan

(r, γ) specify the location ot the fan

θ - projection angle

γ - fan angle



(x, y, z) —> (t, s, z) (rotation by θ degrees around the z-axes)

The FDK Algorithm

Based on the FBP Algorithm for equispatial rays

$$f(t,s,z) = \frac{1}{2}\int_0^{2\pi} \frac{D^2}{(D-s)^2} \int_{-\infty}^{\infty} R_\beta(p,\xi) \frac{D}{\sqrt{D^2+\xi^2+p^2}} h\left(\frac{D\,t}{D-s} - p\right) dp d\beta,$$

where $D = |SO|$ and $\xi = \frac{D}{D-s} z$.

The FDK Algorithm

Based on the FBP Algorithm for equispatial rays

$$f(t,s,z)=\frac{1}{2}\int_0^{2\pi}\frac{D^2}{(D-s)^2}\int_{-\infty}^{\infty}R_\beta(p,\xi)\frac{D}{\sqrt{D^2+\xi^2+p^2}}h\left(\frac{D\,t}{D-s}-p\right)dpd\beta,$$

where $D=|SO|$ and $\xi=\frac{D}{D-s}z$.

The Algorithm

Step 1 Compute the modified projection

$$R'_\beta(p,\xi)=\frac{D}{\sqrt{D^2+\xi^2+p^2}}R_\beta(p,\xi)$$

Step 2 Convolve $R'_\beta(p,\xi)$ with $h(p)/2$

$$Q_\beta(p,\xi)=R'_\beta(p,\xi)*\frac{1}{2}h(p)$$

Step 3 Backproject over the 3D reconstruction grid

$$f(t,s,z)=\int_0^{2\pi}\frac{D^2}{(D-s)^2}Q_\beta(\frac{D\,t}{D-s},\frac{D\,z}{D-s})d\beta$$

The Tuy–Smith sufficiency condition

Exact reconstruction is possible if all planes intersecting the object also intersect the source trajectory at least once.

The Tuy–Smith sufficiency condition

Exact reconstruction is possible if all planes intersecting the object also intersect the source trajectory at least once.

Circular trajectory does not satisfay this condition since a plane parallel to the trajectory may also intersect the object.

The Tuy–Smith sufficiency condition

Exact reconstruction is possible if all planes intersecting the object also intersect the source trajectory at least once.

Circular trajectory does not satisfay this condition since a plane parallel to the trajectory may also intersect the object.

Exact reconstruction form helical cone-beam data is possible. Tam'95 Kudo, Noo, Defrise '98

Conclusions

The challenges:

- Three dimentional images
- Better quality
- Faster algorithms
- Restricted in time

References

- K. Tanabe, *Projection Method for Solving a Singular Systems of Linear Equations and its Applications,* Numer. Math., 17, 203-214 (1971).
- H. Tuy, *An Inversion Formula for Cone-Beam Reconstruction,* SIAM J. Appl. Math. 42 546–552 (1983).
- L. Feldkamp, L. Davis and J. Kress, *Practical Cone-Beam Algorithm,* Journal of the Optical Society of America 1, 612-619 (1984).
- K. Tam, *Three-Dimensional Computerized Tomography Scanning Method and System for Large Objects with Smaller Area Detectors,* US Patent 5,390,112, Feb 14.(1995).
- S. Nilsson, *Fast Backprojection,* Technical report LiTH-ISY-R-1865, Linköping University (1996).
- A. Kak, M. Slaney, *Principles of Computerized Tomographic Imaging,* IEEE Press. (1998).
- H.Kudo, F. Noo and M. Defris, *Cone-Beam Filtered Backprojection Algorithm for Truncated Helical Data,* Physics in Medicine and Biology 43, 2885–2909 (1998).

References

- G.Wang and M. Vannier, *Computerized Tomography,* (1998) available at: `http://dolphin.radiology.uiowa.edu/ge/Teaching/ct/ct.html`
- Y. Saad, H. van der Vorst, *Iterative Solution of Linear Systems in the 20th Century,* J. Comput. Appl. Math. 123, 1–33 (2000).
- F. Natterer and F. Wübbeling, *Mathematical Methods in Image Reconstruction,* SIAM, (2001).
- H. Turbell, *Cone-Beam Reconstruction Using Filtered Backprojectio,* Dissertation No.672, Linköping Sudies in Science and Technology (2001).
- Y. Censor and T. Elfving, *Block-Iterative Algorithms with Diagonally Scaled Oblique Projections for the Linear Feasibility Problems,* Siam J. Matrix Anal. Appl., 24, 40-58 (2002).
- M. Jiang and G. Wang, *Convergence Studies on Iterative Algorithms for Image Reconstruction,* IEEE Transactions on Medical Imaging , 22, 569-579 (2003).