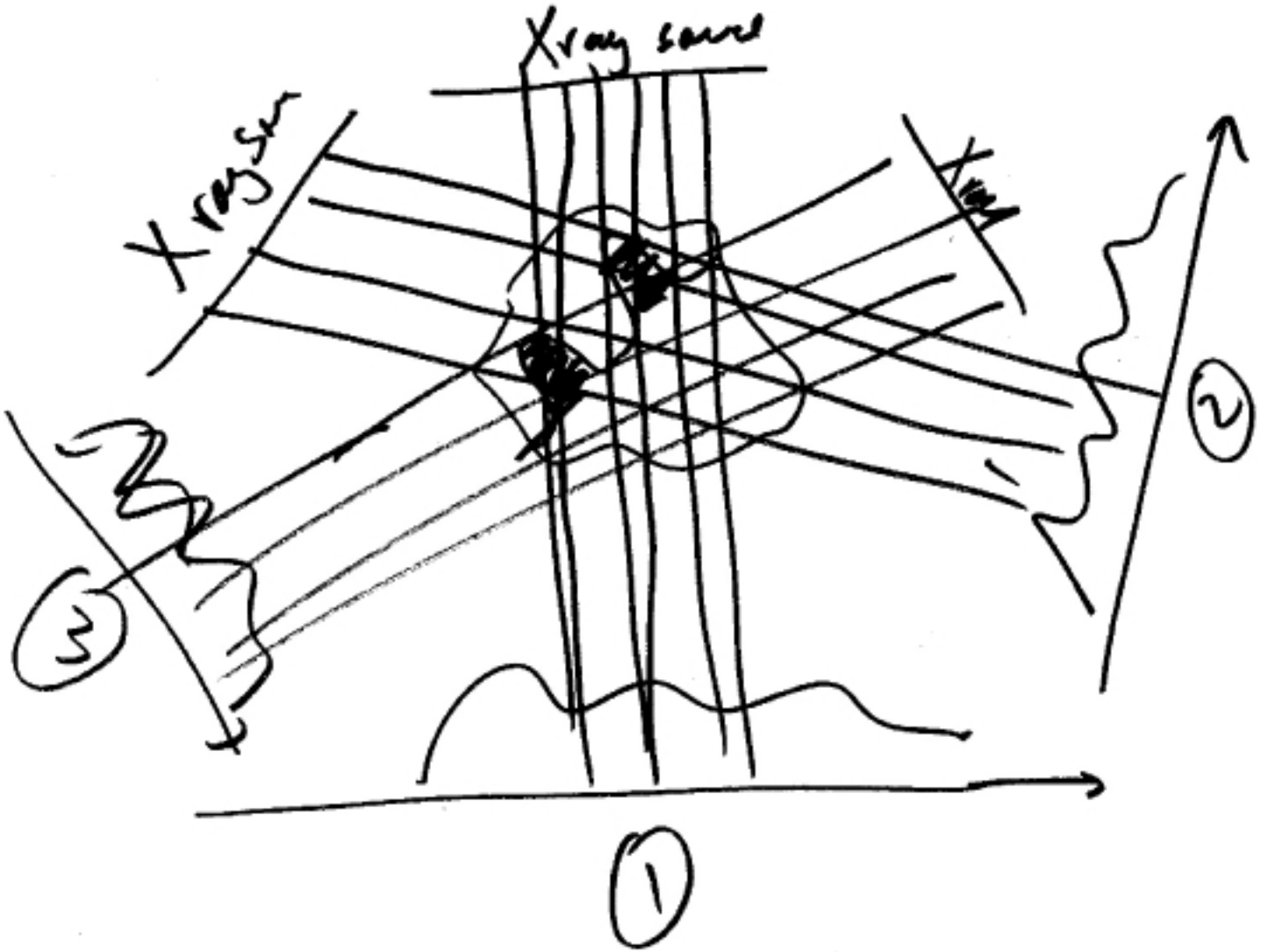
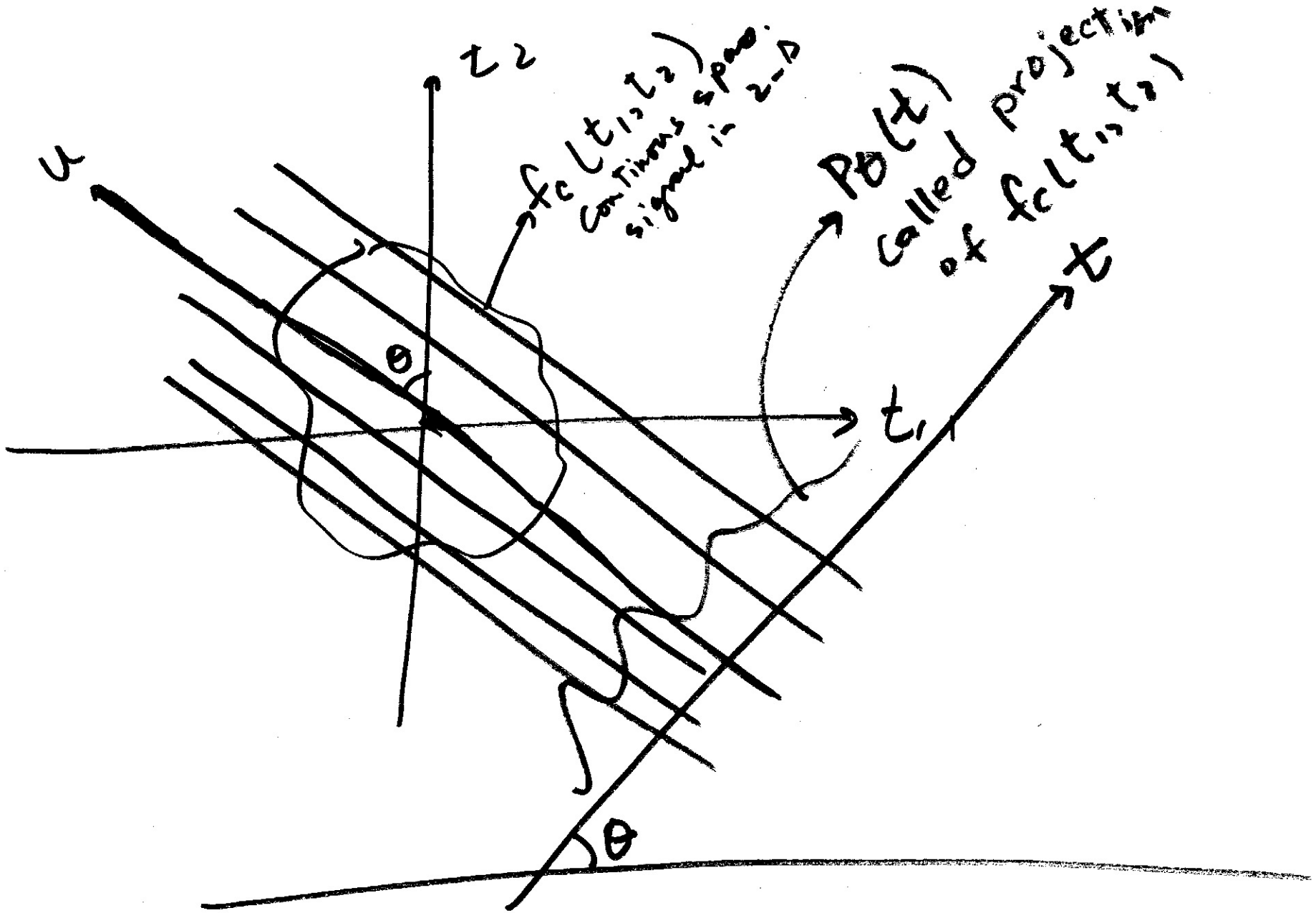
Jan. 30, 2008

Tomography



Def Projection: is mathematical operation that is similar to the physical operation of taking an X-ray photograph with a collimated beam of radiation

---


z_2
$f_c(t_1, t_2)$ continuous space signal in 2-D
$P_\theta(t)$ called projection of $f_c(t_1, t_2)$
u
θ
t_1
t
θ

$$P_\theta(t) = \int_{u=-\infty}^{+\infty} f_c(t_1, t_2)\Big|_{\substack{t_1 = t\cos\theta - u\sin\theta \\ t_2 = t\sin\theta + u\cos\theta}} du$$

projection

Goal: Relate 2D F.T. of f_c To. 1D. F.T. of P_θ

Projection Slice theorem:

$$F_c(\Omega_1, \Omega_2) = 2.D.\ C.T.F.T.\ \{f_c(t_1, t_2)\}$$

$$F_c(\Omega_1, \Omega_2) = \int_{t_1=-\infty}^{+\infty} \int_{t_2=-\infty}^{+\infty} f_c(t_1, t_2)\, e^{-j\Omega_1 t_1}\, e^{-j\Omega_2 t_2}$$

$$f_c(t_1, t_2) = \frac{1}{4\pi^2} \iint F_c(\Omega_1, \Omega_2)\, e^{j\Omega_1 t_1}\, e^{j\Omega_2 t_2}\, d\Omega_1 d\Omega_2$$

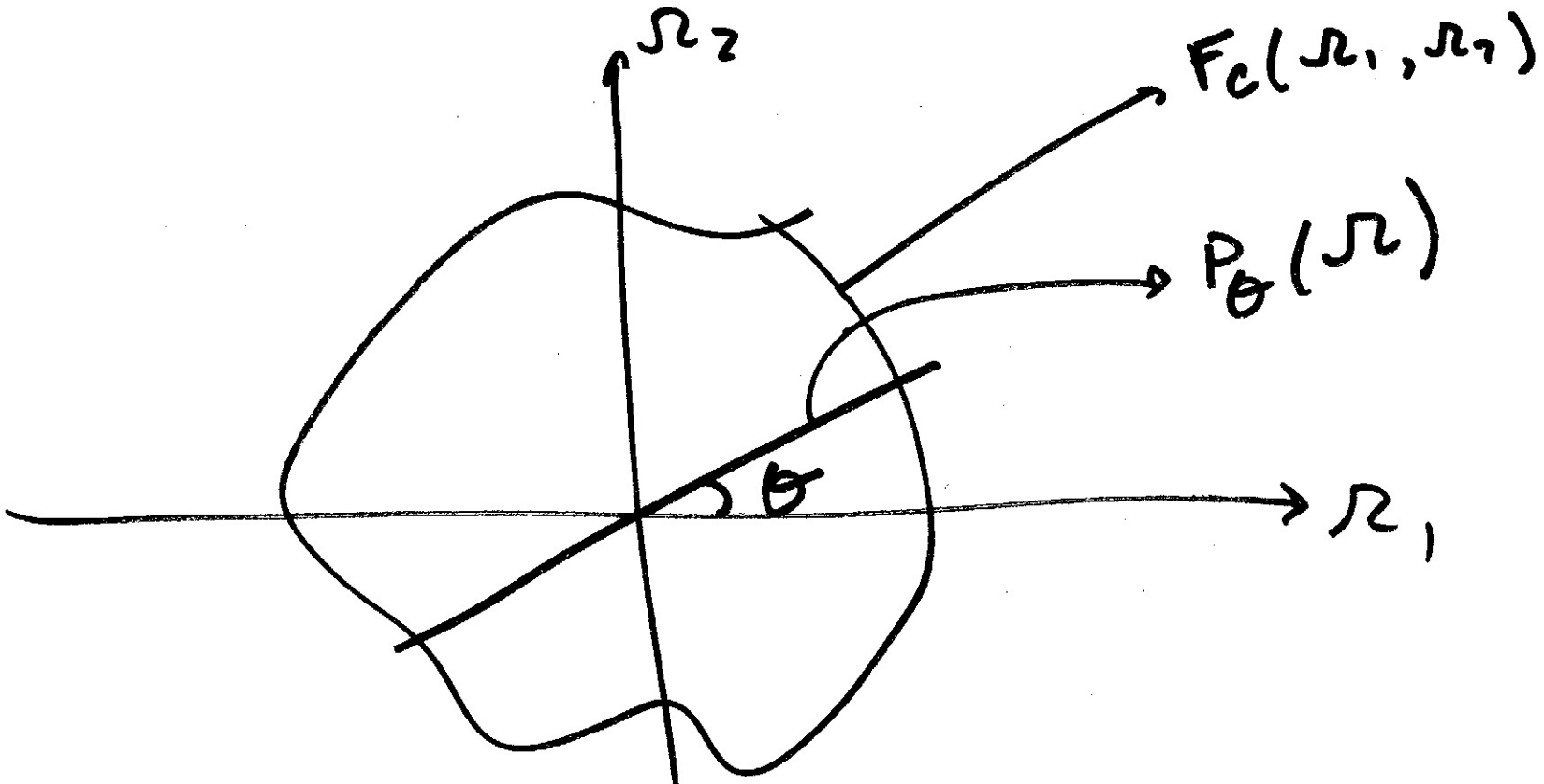
$$P_\theta(\Omega) = \text{1.D. C.T.F.T.} \left\{ P_\theta(t) \right\}$$

$$P_\theta(\Omega) = \int_{t=-\infty}^{+\infty} P\theta(t)\, e^{-j\Omega t}\, dt$$

Projection Slice Theorem:

$$P_\theta(\Omega) = F_c(\Omega_1, \Omega_2) \Big|_{\substack{\Omega_1 = \Omega\cos\theta \\ \Omega_2 = \Omega\sin\theta}}$$

$$P_\theta(\Omega) = F_c(\Omega\cos\theta, \Omega\sin\theta)$$

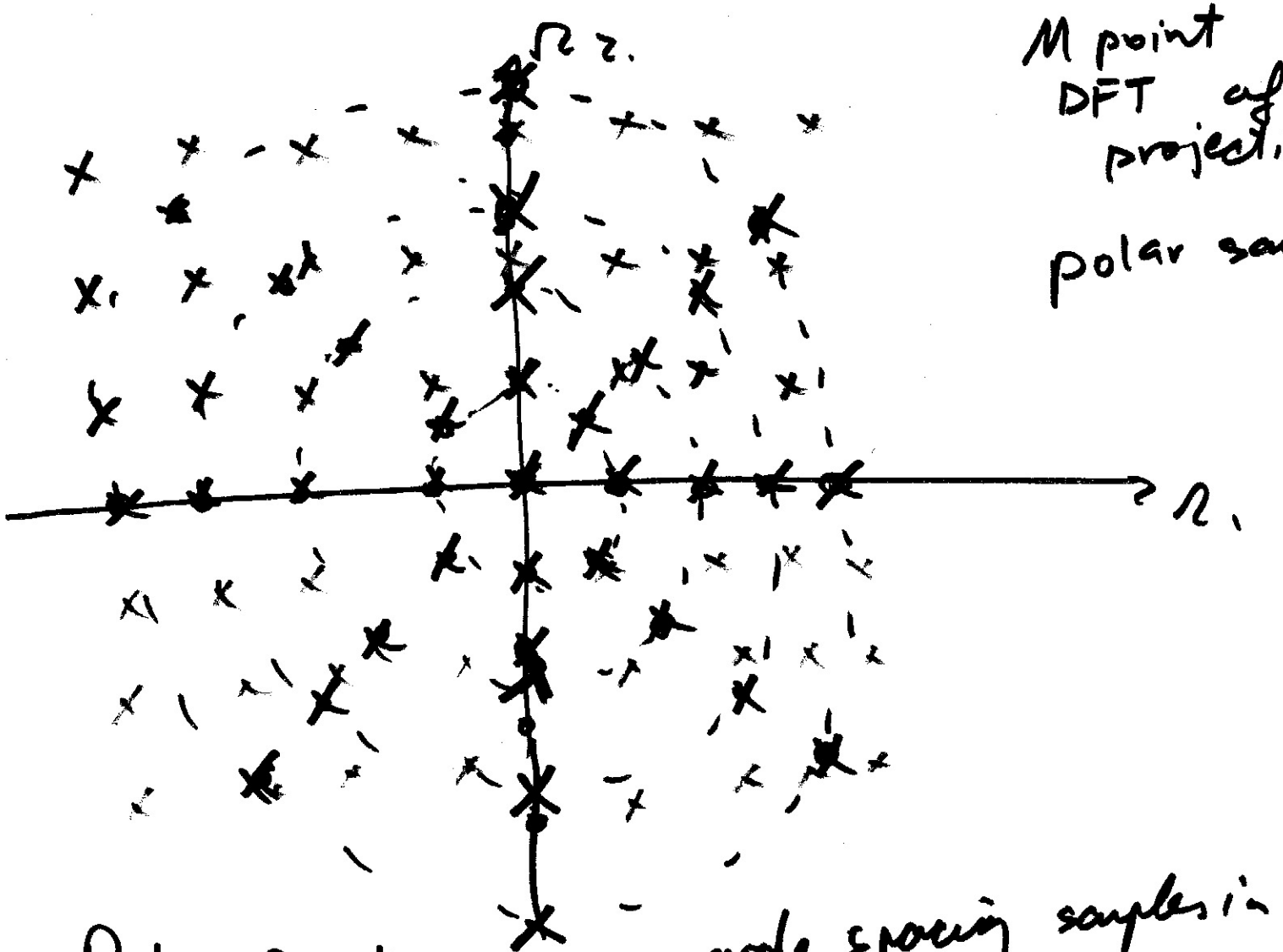


By Taking multiple projection -

you get 2D F.T. of the object

→ Inverse 2D. F.T

→ you get the object $f_c(t_1, t_2)$



M point DFT of projection.

polar sample

Polar Sampling: Every angle spacing samples in Fourier domain are equal.

To get blue cartesian samples from red polar samples, interpolate.

$\Rightarrow$
- zeroth order $\longrightarrow$ nearest neighbor
- 1st order $\longrightarrow$ weighted sum of neighboring samples.

Concentric Squares. Distance between projection samples in F.D. varies a function of projection angle.

Reconstruction Strategies

1. Simple
   - nearest neighbor (zeroth order interp)
   - 1st order Interpolation
2. Radon Inversion Formula
3. Iterative Techniques

$$R_0 > \frac{\pi}{r} \tag{7.55}$$

which implies that

$$N > \frac{\pi d}{r} \tag{7.56}$$

Although this formula should be interpreted primarily as a rule of thumb, it is a reasonably reliable one. Note that the number of views increases as the desired resolution improves. This is not surprising.

7.3.4 Fourier-Domain Reconstruction Algorithms

Let us assume that N projections of $x(u_1, u_2)$ are available at the equispaced angles $\theta_i = \pi i/N$, $i = 0, 1, \ldots, N-1$, that each projection has been sampled at the same sampling rate, and that the M-point DFT of each sampled projection has been computed. These DFT values can be interpreted as samples of the Fourier transform of $x(u_1, u_2)$ on a regular polar raster, such as the one depicted in Figure 7.13(a). If we further assume that $x(u_1, u_2)$ has finite support and that it is approximately band-limited so that it can be represented adequately by its $(N \times N)$-point discrete Fourier transform, the reconstruction problem becomes one of interpolation of the Fourier transform. We can interpolate from the known transform values on the polar raster to the unknown ones on the DFT raster, perform an inverse DFT, and use the result as an estimate of the samples of x.

To perform the necessary interpolation we might consider the use of either zeroth-order or linear interpolation. In the Fourier domain, most points (ω_1, ω_2) on the DFT raster are surrounded by four polar samples, as depicted in Figure 7.14. With zeroth-order interpolation, each DFT sample is assigned the value of the nearest polar sample and with linear interpolation it is assigned a weighted average of the four nearest polar samples, the weighting varying inversely with the Euclidean distance between the points.

If we are free to vary the sampling rates of the individual projections, we can alter the form of the polar raster of DFT samples to facilitate the process of interpolation. The raster in Fig. 7.13(b), for example, results if the spacing between samples in the projection at angle θ is π/W_θ, where

$$W_\theta = \frac{R_0}{\max(|\cos\theta|, |\sin\theta|)} \tag{7.57}$$

With this raster, all the interpolation is confined to the rows and columns of a rectangular DFT lattice and is thus 1-D. This results not only in reduced computation but also in less error resulting from the interpolation process [18].

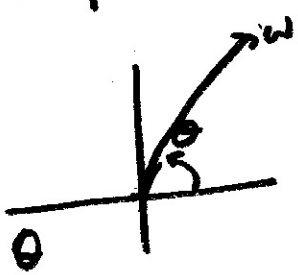
Some results of these algorithms are shown in Figures 7.15 and 7.16. It can be seen that the reconstructions made using linear interpolation are superior to those made with zeroth-order interpolation and that the modified polar raster gives superior reconstructions to the normal polar one. From Figure 7.16 one can discern how the quality of the reconstruction improves with the number of projections. Further details on these algorithms can be found in [18, 19].

Radon Inversion Formula

$$f_c(t_1,t_2) = \frac{1}{4\pi^2}\int_{-\infty}^{+\infty}\int F_c(\Omega_1,\Omega_2)\, e^{j\Omega_1 t_1}\, e^{j\Omega_2 t_2}\, d\Omega_1\, d\Omega_2$$

↳ convert to polar coordinates $\Omega_1, \Omega_2 \rightarrow \omega, \theta$

$$f_c(t_1,t_2) = \frac{1}{4\pi^2}\int_0^{\pi}\int_{-\infty}^{+\infty} \underbrace{F_c(\omega\cos\theta, \omega\sin\theta)}_{P_\theta(\omega)}\, e^{j\omega(t_1\cos\theta + t_2\sin\theta)}\, |\omega|\, d\omega\, d\theta$$



$$f_c(t_1,t_2) = \frac{1}{4\pi^2}\int_0^{\pi}\underbrace{\int_{-\infty}^{+\infty} P_\theta(\omega)\, e^{j\omega(t_1\cos\theta + t_2\sin\theta)}\, |\omega|\, d\omega}_{I}\, d\theta$$

$$I = \left\{ \mathcal{F}^{-1}\left[P_\theta(\omega)\,|\omega| \right] \right\}_{t = t_1\cos\theta + t_2\sin\theta}$$

Define $G_\theta(\omega) = P_\theta(\omega)\,|\omega|$

$$F^{-1}\{G_\theta(\omega)\} = g_\theta(t)$$

$$I = \int_{-\Omega}^{+\Omega} G_\theta(\omega)\, e^{j\omega(t_1\cos\theta + t_2\sin\theta)}\, d\omega$$

$$I = \left[g_\theta(t) \right]_{t = t_1\cos\theta + t_2\sin\theta}$$

$$I = g_\theta(t_1\cos\theta + t_2\sin\theta)$$

$$f_c(t_1,t_2) = \frac{1}{4\pi^2} \int_0^{\pi} g_\theta(t_1\cos\theta + t_2\sin\theta)\, d\theta \quad (*)$$

Recall $G_{\theta}(\omega) = P_{\theta}(\omega)\,|\omega|$

$g_{\theta}(t) = k(t) * P_{\theta}(t)$

where $k(t) = \mathcal{F}^{-1}\{|\omega|\}$

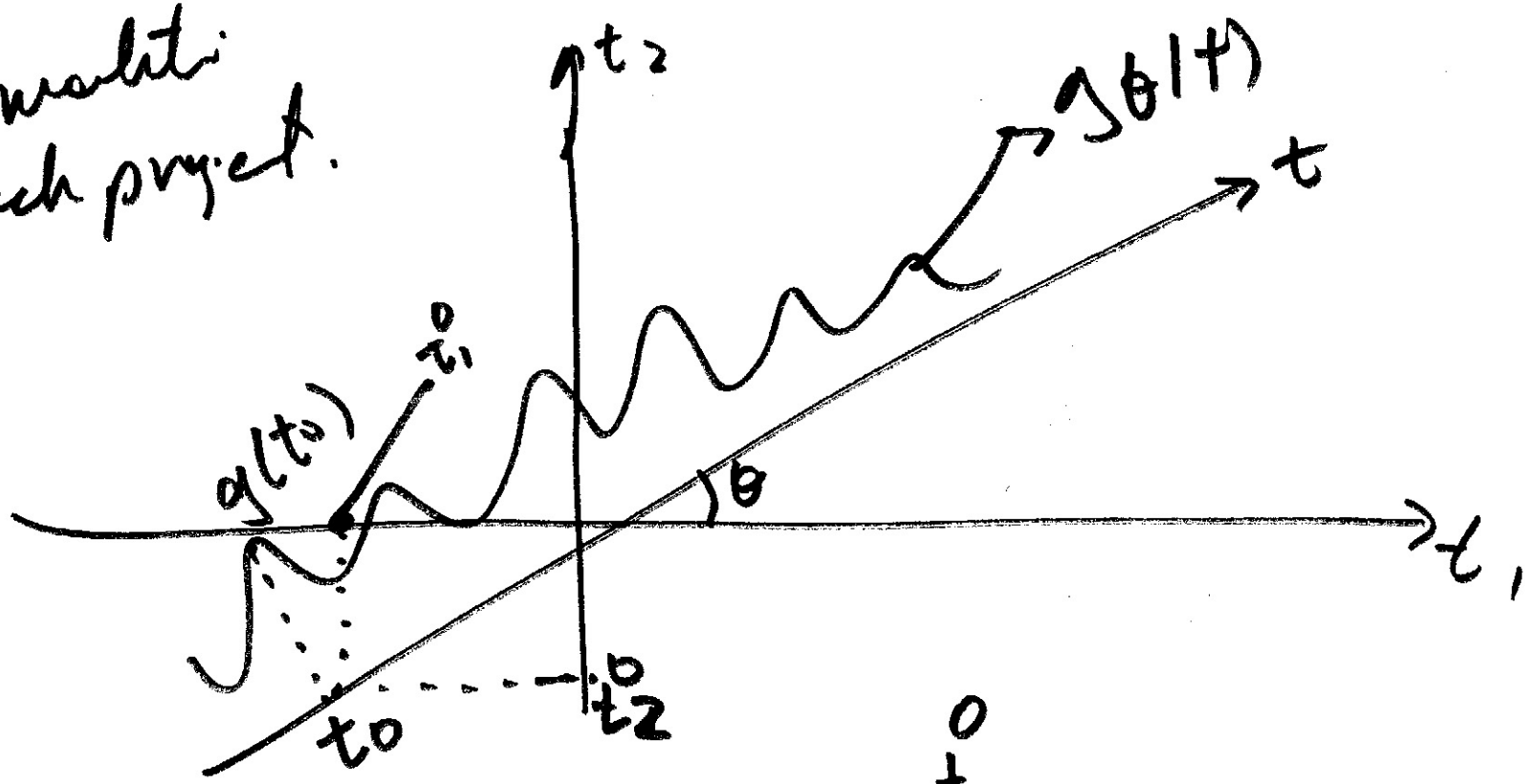
$$g_{\theta}(t) = \frac{d}{dt}\int_{-\infty}^{+\infty} \frac{P_{\theta}(\tau)}{t-\tau}\, d\tau$$

---

steps:

1. $P_{\theta}(t)$
2. convolve with $k(t)$ to get $g_{\theta}(t)$
3. plug $g_{\theta}(t)$ into (*)

Convolution
Back project.



at $t = t_0$

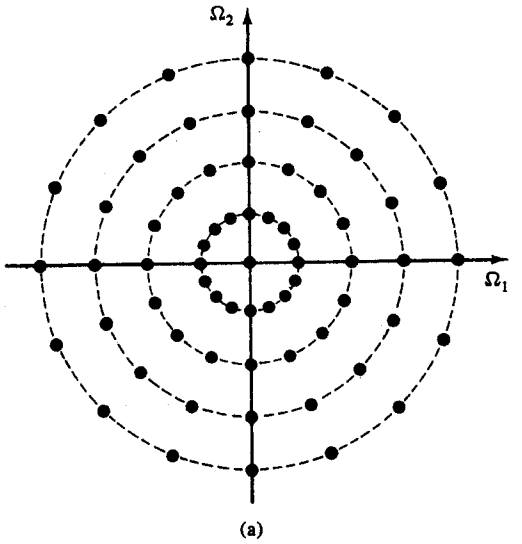
$t_0 \leftarrow t_1^0, t_2^0$

~~$f_t(t_1^0, t_2^0) = g(t_0)$~~

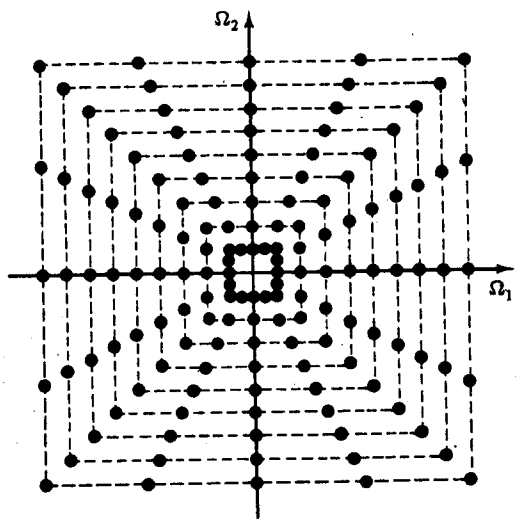
Iterative Recon

$$f_c^{K}(t_1,t_2) = f_c^{K-1}(t_1,t_2) + \sum_{i=1}^{N} \lambda_i \left[P_{\theta_i}(t_1 \cos\theta + t_2 \sin\theta) - D_i\{f_c^{K-1}(t_1,t_2)\} \right]$$

- $P_{\theta_i}(t_1 \cos\theta + t_2 \sin\theta)$ → actual projection observed
- λ_i → relaxation convergence rate
- $D_i\{f_c^{K-1}(t_1,t_2)\}$ → projection operator applied to K^{th} iteration reconstruction



(a)



(b)

Figure 7.13 (a) Polar raster of samples in the Fourier domain, obtained by sampling all projections at the same sampling rate. (b) Concentric squares raster, obtained by varying the sampling rate with the angle of the projection. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)

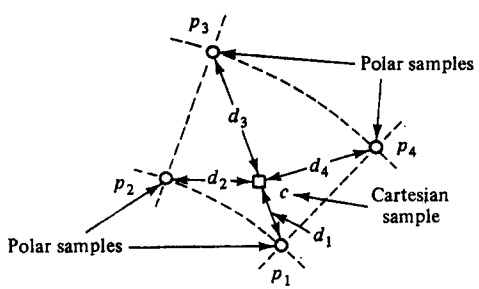


Figure 7.14 Parameters for the definition of zeroth-order and linear interpolation. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)

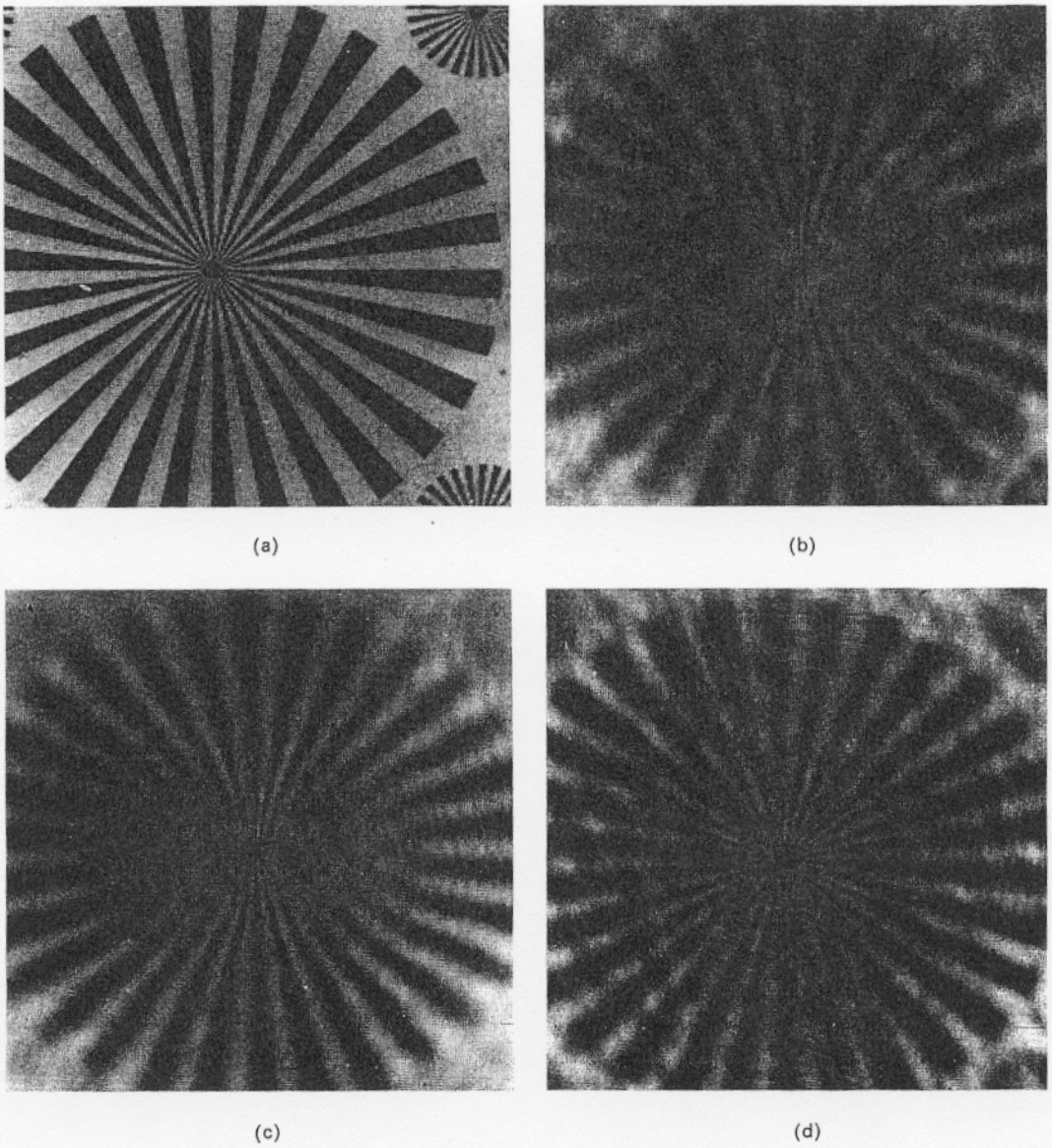


Figure 7.15 Reconstructions of the original image shown in (a) made from 64 equiangular projections using various interpolation algorithms. (b) Zeroth-order interpolation, polar raster. (c) Linear interpolation, polar raster. (d) Linear interpolation, concentric squares raster. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)

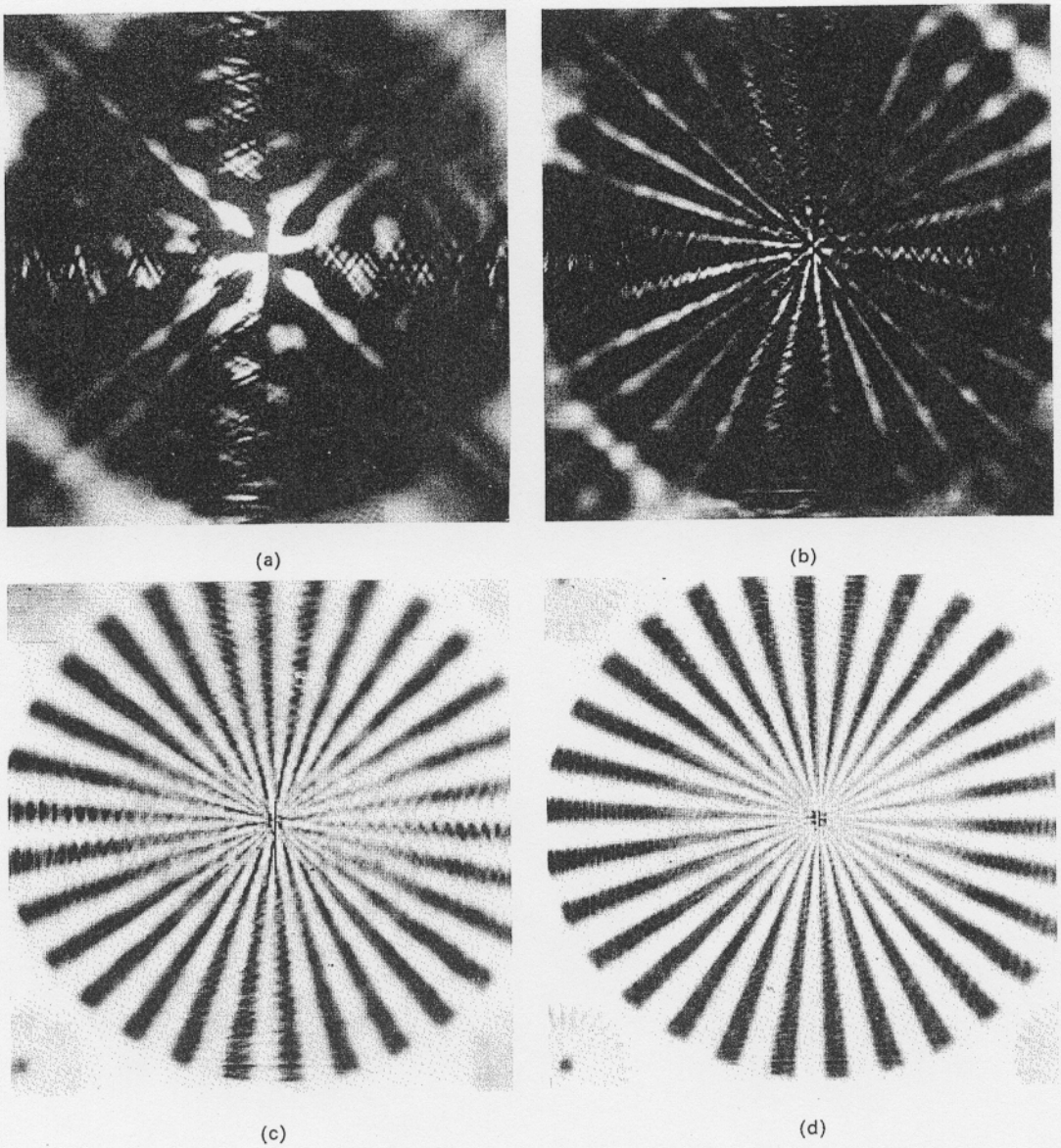
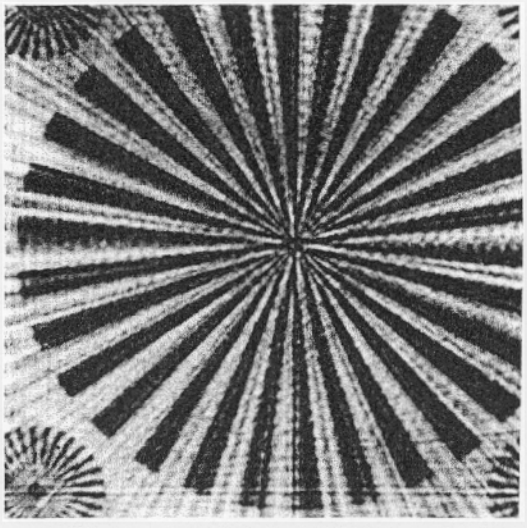


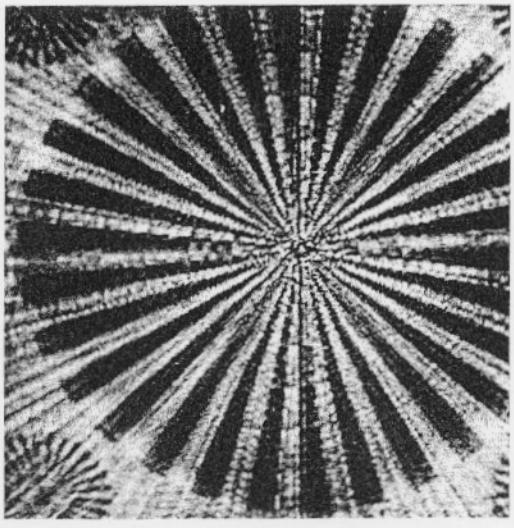
Figure 7.16 Reconstructions made using linear interpolation from a concentric squares raster using: (a) 16 projections; (b) 32 projections; (c) 64 projections; (d) 128 projections. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)

filter gain increases with increasing frequency, high-frequency noise will be amplified. Thus to minimize the deterioration that can result from such noise, the filter $k(t)$ is typically chosen to have an approximately linear response out to some cutoff frequency beyond which the response goes to zero. The exact shape of the frequency response is also governed by computational convenience [20, 21].

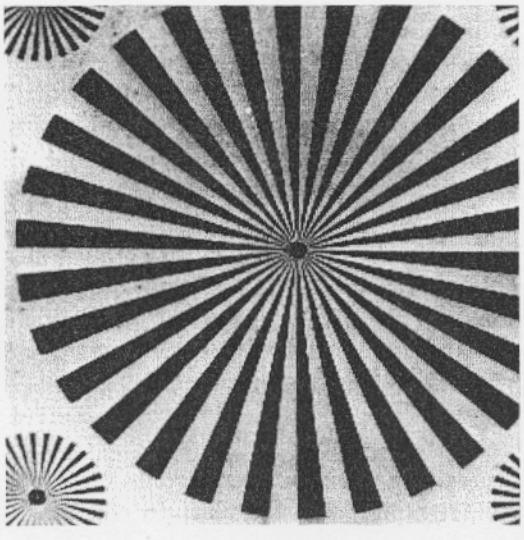
Some reconstructions obtained using this algorithm are shown in Figures 7.17 and 7.18. The resolution here is noticeably better than for the reconstructions



(a)



(b)



(c)

Figure 7.17 Reconstruction of the original image shown below from 64 equiangular projections using the convolution/back-projection method. (a) Concentric squares projections. (b) Polar projections. (c) Original. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)

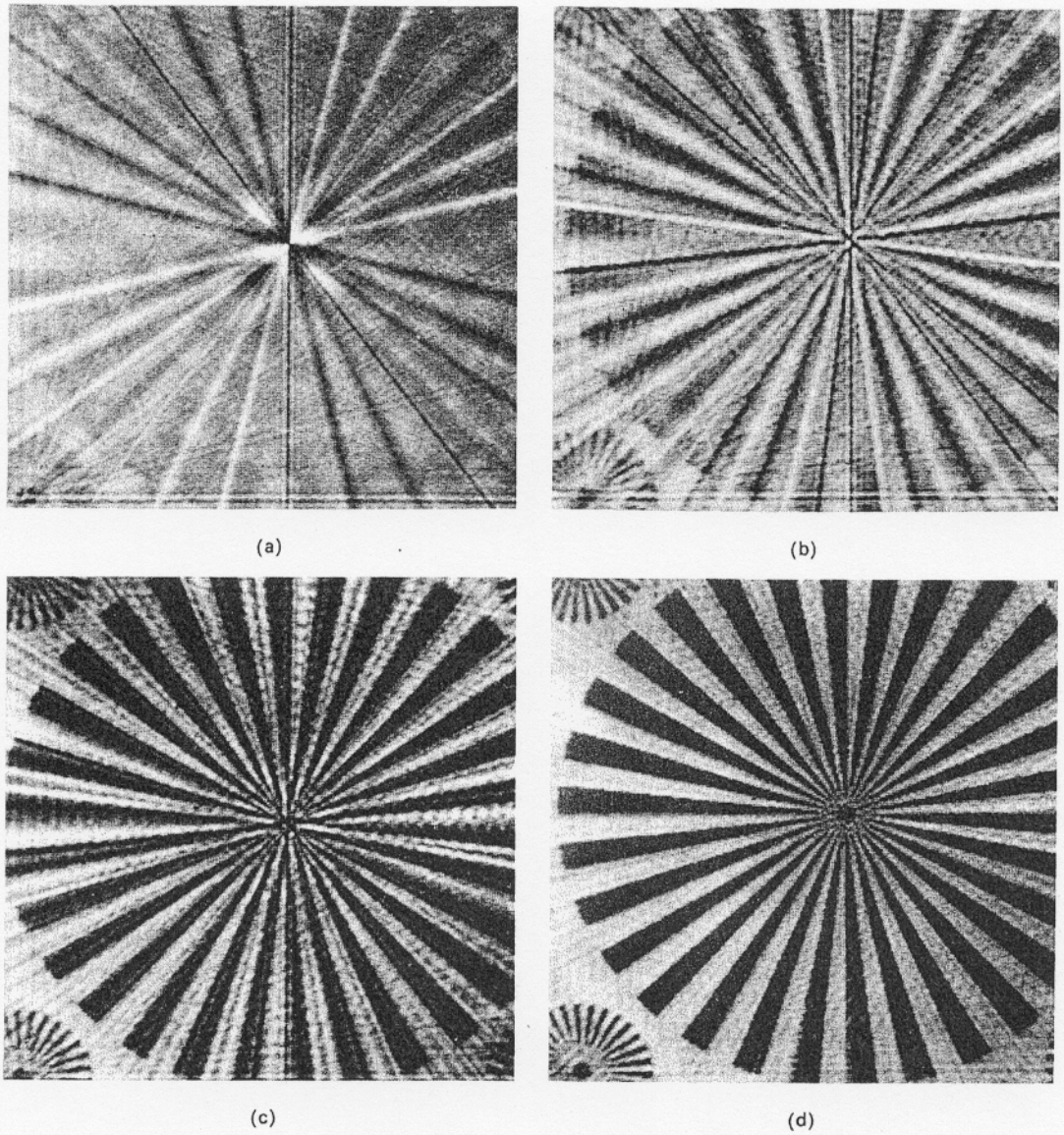


Figure 7.18 Reconstructions made using the convolution/back-projection method applied to concentric squares projections. (a) 16 projections. (b) 32 projections. (c) 64 projections. (d) 128 projections. (Courtesy of Russell M. Mersereau, *Proc. IEEE*, © 1974 IEEE.)