Digital Image Processing

Chap 5: Image Restoration & Reconstruction

清華大學電機系林嘉文
cwlin@ee.nthu.edu.tw
Tel: 03-5731152

Model of Image Degradation & Restoration

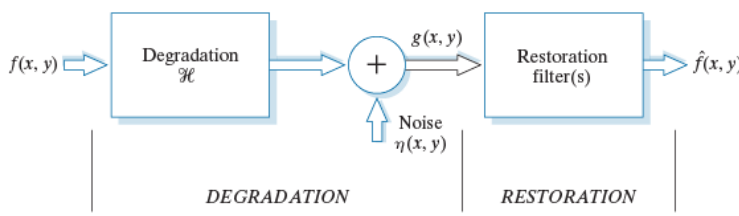


FIGURE 5.1 A model of the image degradation/ restoration process.

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y)$$
$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

Noise Models

- Spatial and frequency property of noise
  - **White noise (random noise)**
  - **Gaussian noise**

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2}$$

    - μ: mean ; σ^2 :variance
    - 70% $[(\mu-\sigma), (\mu+\sigma)]$
    - 95% $[(\mu-2\sigma), (\mu+2\sigma)]$
  - **Rayleigh noise**

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a \end{cases}$$

    - $\mu = a+(\pi b/4)^{1/2}$; $\sigma^2 = b(4-\pi)/4$

Noise Models

- Spatial and frequency property of noise
  - **Erlang (gamma) noise**

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases}$$

    - $\mu = b/a$; $\sigma^2 = b/a^2$
  - **Exponential noise**

$$p(z) = \begin{cases} ae^{-az} & for\ z \geq 0 \\ 0 & for\ z < 0 \end{cases}$$

    - $\mu = 1/a$; $\sigma^2 = 1/a^2$

Noise Models

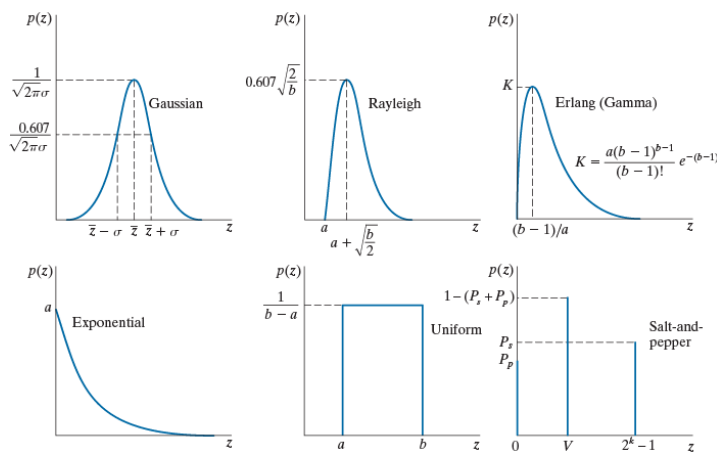
- Spatial and frequency property of noise
  - **Uniform noise**

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{for } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$

  - $\mu = (a+b)/2$; $\sigma^2 = (b-a)^2/12$
  - **Impulse noise (salt and pepper noise)**

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$

Noise Models



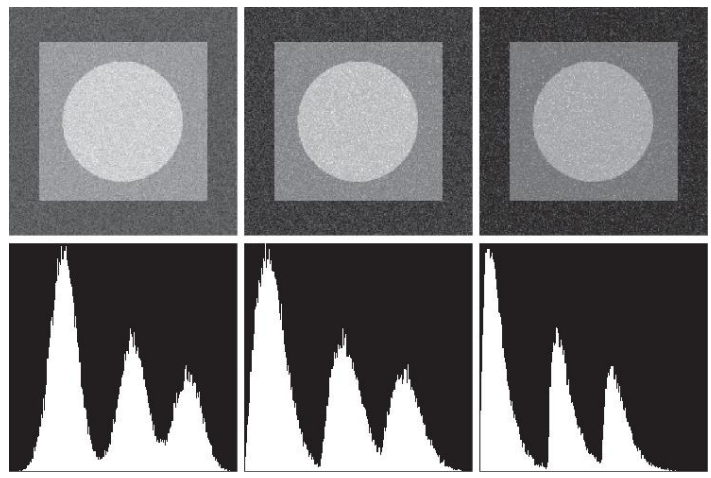
a b c
d e f

FIGURE 5.2 Some important probability density functions.

Noise Models

FIGURE 5.3
Test pattern used to illustrate the characteristics of the PDFs from Fig. 5.2.

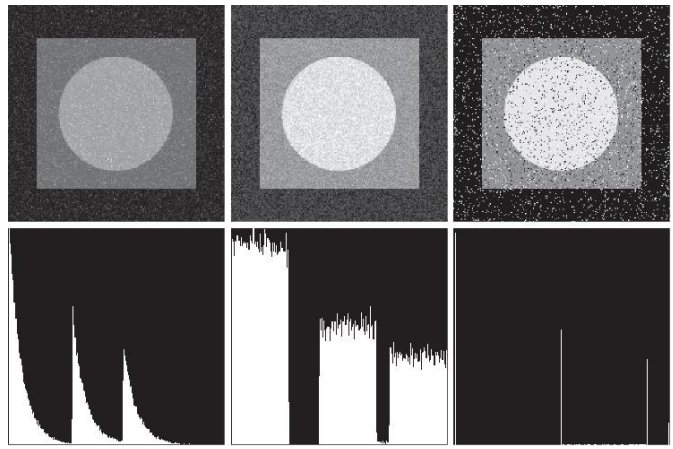
Noise Models



a b c
d e f

FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and Erlanga noise to the image in Fig. 5.3.

Noise Models



g h i
j k l

FIGURE 5.4 (*continued*) Images and histograms resulting from adding exponential, uniform, and salt-and-pepper noise to the image in Fig. 5.3. In the salt-and-pepper histogram, the peaks in the origin (zero intensity) and at the far end of the scale are shown displaced slightly so that they do not blend with the page background.

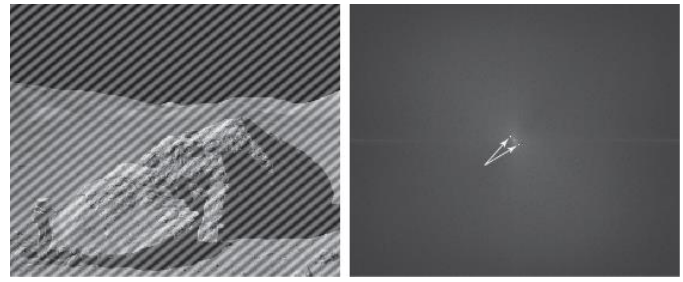
Periodic Noise

- Periodic noise is due to the electrical or electromechanical interference during image acquisition.
- It can be estimated through the inspection of the Fourier spectrum of the image.

Periodic Noise

a b

FIGURE 5.5
(a) Image corrupted by additive sinusoidal noise. (b) Spectrum showing two conjugate impulses caused by the sine wave. (Original image courtesy of NASA.)



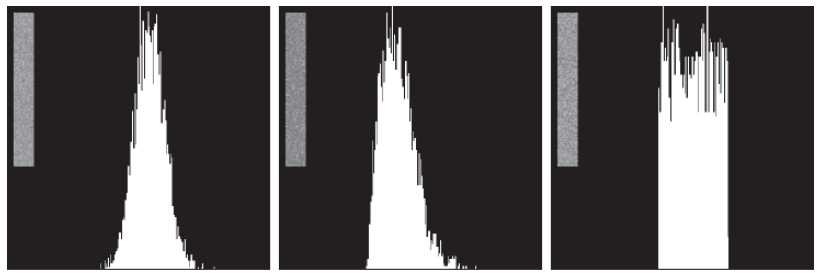
Noise Parameters Estimation

- Capture a set of images of "flat" environment.
- Use an optical sensor to capture the image of a solid gray board that is illuminated uniformly.
- The resulting image is a good indicator of system noise.
- Find the mean μ and standard deviation σ of the histogram $p(z_i)$ of the resulting image.

$$\mu = \sum_{z_i \in S} z_i p(z_i) \qquad \sigma^2 = \sum_{z_i \in S} (z_i - \mu)^2 p(z_i)$$

- From μ and $\sigma \Rightarrow$ calculate a and b (the parameter of that specific noise distribution).

Noise Parameters Estimation



a b c

FIGURE 5.6 Histograms computed using small strips (shown as inserts) from (a) the Gaussian, (b) the Rayleigh, and (c) the uniform noisy images in Fig. 5.4.

Restoration Using Spatial Filtering

- When the only degradation is the additive noise

$$g(x,y) = f(x,y) + \eta(x,y)$$
$$G(u,v) = F(u,v) + N(u,v)$$

- **Mean filters**
  - **Arithmetic mean filter**: the simplest of mean filters

$$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s,t)$$

  - **Geometric mean filter**: preserves more details

$$\hat{f}(x,y) = \left[\prod_{(s,t)\in S_{xy}} g(s,t) \right]^{1/mn}$$

Restoration Using Spatial Filtering

- **Harmonic mean filter**: good for salt noise and Gaussian noise, but fail for pepper noise.

$$\hat{f}(x,y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{g(s,t)}}$$

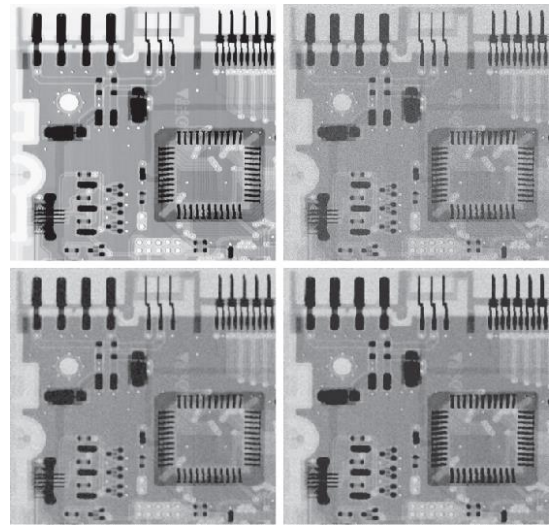
- **Contra-harmonic filter**: $Q > 0$ reduces pepper noise; $Q < 0$ reduces salt noise.

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$$

Restoration Using Spatial Filtering

a b
c d

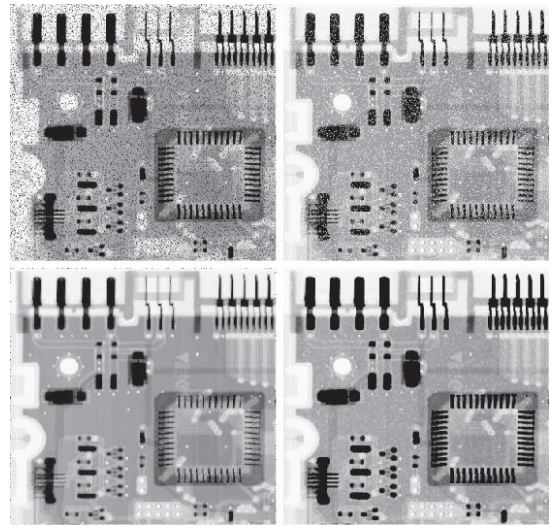
FIGURE 5.7
(a) X-ray image of circuit board. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3×3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)



Restoration Using Spatial Filtering

a b
c d

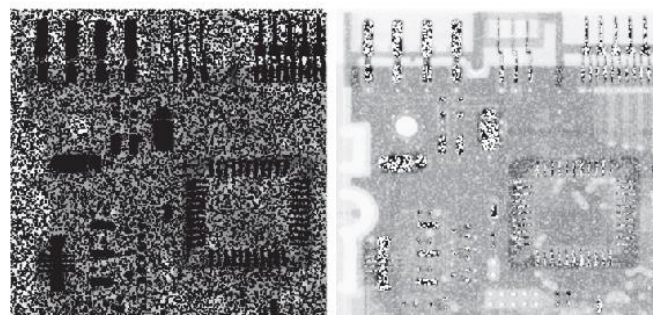
FIGURE 5.8
(a) Image corrupted by pepper noise with a probability of 0.1. (b) Image corrupted by salt noise with the same probability. (c) Result of filtering (a) with a 3×3 contra-harmonic filter $Q = 1.5$. (d) Result of filtering (b) with $Q = -1.5$.



Restoration Using Spatial Filtering

a b

FIGURE 5.9
Results of selecting the wrong sign in contraharmonic filtering. (a) Result of filtering Fig. 5.8(a) with a contraharmonic filter of size 3×3 and $Q = -1.5$. (b) Result of filtering Fig. 5.8(b) using $Q = 1.5$.



Order-Statistics Filters

- **Median filter** $\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{\text{median}}\{g(s,t)\}$
- **Max filter**: find the brightest points to reduce the pepper noise

$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}}\{g(s,t)\}$$

- **Min filter**: find the darkest point to reduce the salt noise

$$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}}\{g(s,t)\}$$

- **Midpoint filter**

$$\hat{f}(x,y) = \frac{1}{2}\left[\min_{(s,t)\in S_{xy}}\{g(s,t)\} + \max_{(s,t)\in S_{xy}}\{g(s,t)\}\right]$$

Order-Statistics Filters

- **Alpha-trimmed mean filter**
  - Suppose delete $d/2$ lowest and $d/2$ highest gray-level value in the neighborhood of (s,t) and average the rest

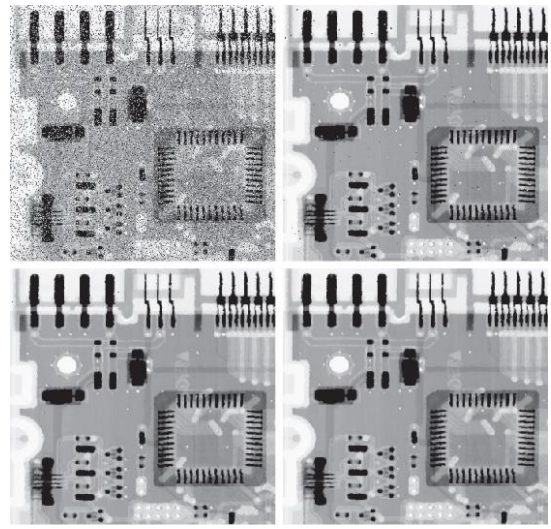
$$\hat{f}(x,y) = \frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} g_r(s,t)$$

where $d = 0 \sim mn-1$

Order-Statistics Filters

a b
c d

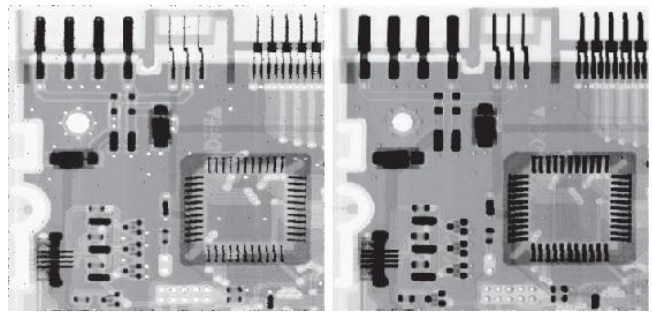
FIGURE 5.10
(a) Image corrupted by salt-and- pepper noise with probabilities $P_s = P_p = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.



Order-Statistics Filters

a b

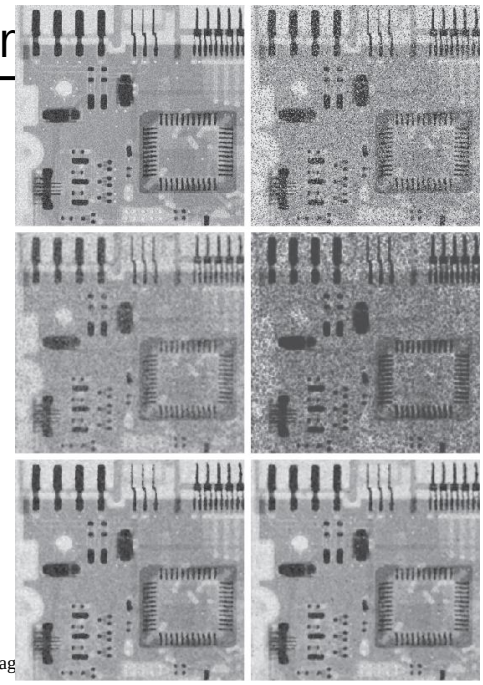
FIGURE 5.11
(a) Result of filtering Fig. 5.8(a) with a max filter of size 3×3. (b) Result of filtering Fig. 5.8(b) with a min filter of the same size.



Order-Statistics Filter

a b
c d
e f

FIGURE 5.12
(a) Image corrupted by additive uniform noise. (b) Image additionally corrupted by additive salt-and-pepper noise. (c)-(f) Image (b) filtered with a 5×5: (c) arithmetic mean filter; (d) geometric mean filter; (e) median filter; (f) alpha-trimmed mean filter, with $d = 6$.



Adaptive Local Noise Reduction Filter

- Measurements for local region S_{xy} centered at (x, y)
  - (1) noisy image at (x, y): $g(x, y)$
  - (2) The local mean m_L in S_{xy}
  - (3) The local variance σ_L^2 in S_{xy}
  - (4) The variance of noise σ_η^2
- Assuming **additive noise**, the filter should follow
  - (a) Zero-noise case: If $\sigma_\eta^2 = 0$ then $f(x, y) = g(x, y)$
  - (b) Edges: If $\sigma_L^2 \gg \sigma_\eta^2$ then $f(x, y) \approx g(x, y)$ (edge preserved)
  - (c) If $\sigma_L^2 = \sigma_\eta^2$ then $f(x, y) = m_L$,
  - (d) Adaptive filter (assume $\sigma_L^2 \geq \sigma_\eta^2$ because S_{xy} is a subset of $g(x, y)$)

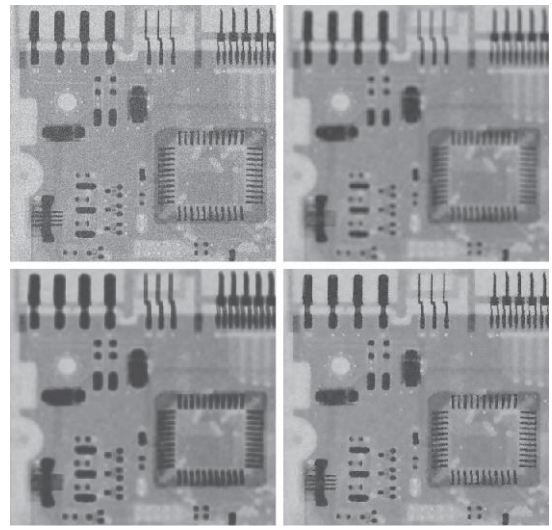
$$\hat{f}(x, y) = g(x, y) - \frac{\sigma_\eta^2}{\sigma_L^2}[g(x, y) - m_L]$$

In practice, σ_η^2 should be estimated before filtering.

Adaptive Local Noise Reduction Filter

a b
c d

FIGURE 5.13
(a) Image corrupted by additive Gaussian noise of zero mean and a variance of 1000. (b) Result of arithmetic mean filtering. (c) Result of geometric mean filtering. (d) Result of adaptive noise-reduction filtering. All filters used were of size 7×7.



Adaptive Median Filter

- **Adaptive median filter** can handle impulse noise with larger probabilities (> 0.2) P_a and P_b.

  Idea: changing (increasing) the window size during operation

  1. To remove impulse (salt-and-pepper) noise
  2. To provide smoothing of non-impulsive noise
  3. To reduce distortion (excessive thinning or thickening of object boundaries)

- Notations:

  $z_{\min}$ = minimum gray-level value in S_{xy}

  $z_{\max}$ = maximum gray-level value in S_{xy}

  z_{med} = median gray-level value in S_{xy}

  z_{xy} = gray-level at (x, y)

  $S_{\max}$ = maximum allowed size of S_{xy}

Adaptive Median Filter

- The **adaptive median filtering** algorithm works in two stages:
- Stage *A*: $A1 = z_{\text{med}} - z_{\text{min}}$

  $A2 = z_{\text{med}} - z_{\text{max}}$

  If $A1 > 0$ AND $A2 < 0$ go to stage B

  else increase the window size (z_{med} is an impulse)

  If window size $\leq S_{\text{max}}$ repeat stage *A*

  else output z_{xy}

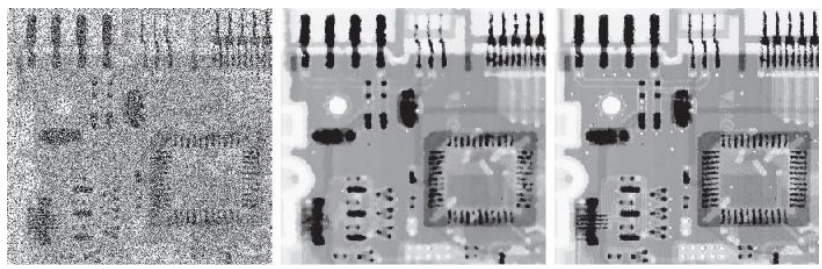
  Stage *B*: $B1 = z_{xy} - z_{\text{min}}$ (z_{med} is not an impulse)

  $B2 = z_{xy} - z_{\text{max}}$

  If $B1 > 0$ AND $B2 < 0$, output z_{xy}

  Else output z_{med}

Adaptive Median Filter



a b c

FIGURE 5.14 (a) Image corrupted by salt-and-pepper noise with probabilities $P_s = P_p = 0.25$. (b) Result of filtering with a 7×7 median filter. (c) Result of adaptive median filtering with $S_{\text{max}} = 7$.

Bandreject Filters

- **Ideal bandreject filter**

$$H(u,v)=\begin{cases}1 & \text{if } D(u,v)<D_0-W/2\\ 0 & \text{if } D_0-W/2\le D(u,v)\le D_0+W/2\\ 1 & \text{if } D(u,v)>D_0+W/2\end{cases}$$

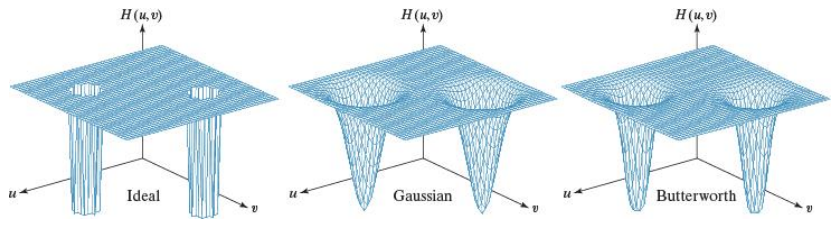
- **Butterworth bandreject filter**

$$H(u,v)=\frac{1}{1+\left[\dfrac{D(u,v)W}{D^2(u,v)-D_0^2}\right]^{2n}}$$

- **Gaussian bandreject filter**

$$H(u,v)=1-e^{-\frac{1}{2}\left[\frac{D^2(u,v)-D_0^2}{D(u,v)W}\right]^2}$$

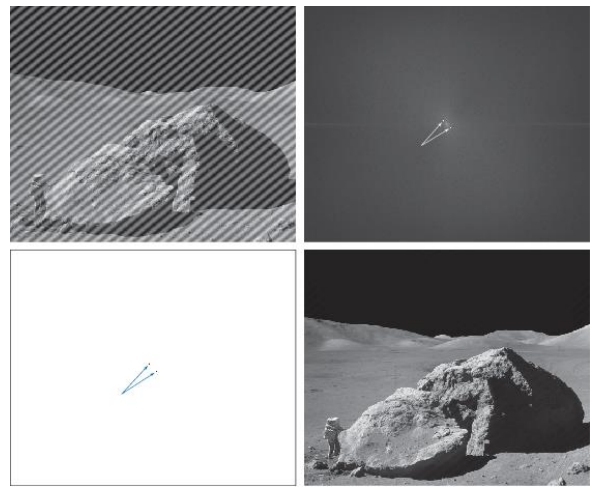
Bandreject Filters



a b c

FIGURE 5.15 Perspective plots of (a) ideal, (b) Gaussian, and (c) Butterworth notch reject filter transfer functions.

Bandreject Filters



a b
c d

FIGURE 5.16
(a) Image corrupted by sinusoidal interference. (b) Spectrum showing the bursts of energy caused by the interference. (The bursts were enlarged for display purposes.) (c) Notch filter (the radius of the circles is 2 pixels) used to eliminate the energy bursts. (The thin borders are not part of the data.) (d) Result of notch reject filtering. (Original image courtesy of NASA.)

Bandpass Filters

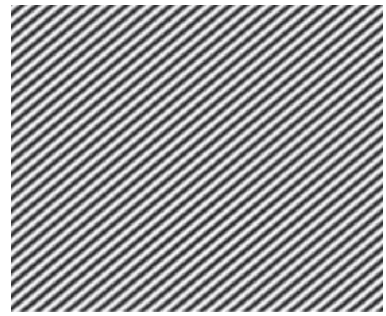
- Obtained from band reject filter

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

- To isolate an image of certain frequency band.

Bandpass Filters

FIGURE 5.17 Sinusoidal pattern extracted from the DFT of Fig. 5.16(a) using a notch pass filter.



Notch Filters

- A **notch filter** rejects (passes) frequencies in predefined neighborhoods about a center frequency. The filter function of a notch reject filter is

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

where

$$D_1(u,v)=\left[(u-M/2-u_0)^2+(v-N/2-v_0)^2\right]^{1/2}$$

$$D_2(u,v)=\left[(u-M/2+u_0)^2+(v-N/2+v_0)^2\right]^{1/2}$$

Notch Filters

- Butterworth notch reject filter

$$H(u,v) = \frac{1}{1+\left[\dfrac{D_0^2}{D_1(u,v)D_2(u,v)}\right]^n}$$

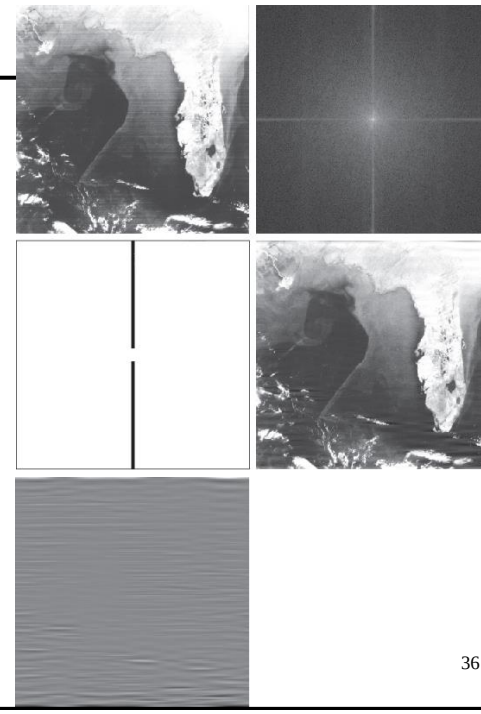
- Gaussian notch reject filter

$$H(u,v) = 1 - e^{-\frac{1}{2}\left[\frac{D_1(u,v)D_2(u,v)}{D_0^2}\right]}$$

$$H_{NP}(u,v) = 1 - H_{NR}(u,v)$$

- It becomes a low-pass filter when $u_0 = v_0 = 0$

Notch Filters



a b
c d

FIGURE 5.18
(a) Satellite image of Florida and the Gulf of Mexico. (Note horizontal sensor scan lines.) (b) Spectrum of (a). (c) Notch reject filter transfer function. (The thin black border is not part of the data.) (d) Filtered image. (Original image courtesy of NOAA.)

FIGURE 5.19
Noise pattern extracted from Fig. 5.18(a) by notch pass filtering.

Optimal Notch Filtering

- Clearly defined interference is not common.
- Images from electro-optical scanner are corrupted by periodic degradation.
- Several interference components are present.
- Place a **notch pass filter** $H(u,v)$ at the location of each spike, *i.e.*, $N(u,v) = H_{NP}(u,v)G(u,v)$, where $G(u,v)$is the corrupted image.

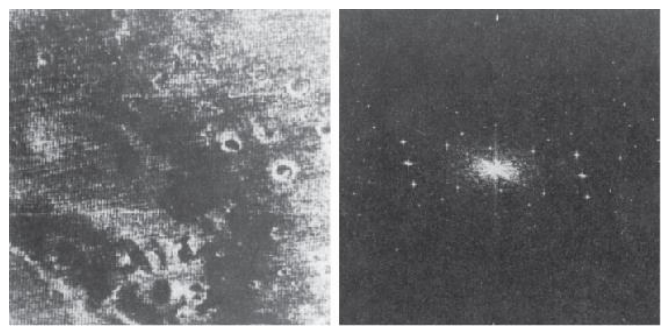
$$\eta(x,y) = \Im^{-1}\{H_{NP}(u,v)G(u,v)\}.$$

- The effect of components not present in the estimate of $\eta(x,y)$ can be minimized.

Optimal Notch Filtering

a b

FIGURE 5.20
(a) Image of the Martian terrain taken by Mariner 6. (b) Fourier spectrum showing periodic interference. (Courtesy of NASA.)



Optimal Notch Filtering

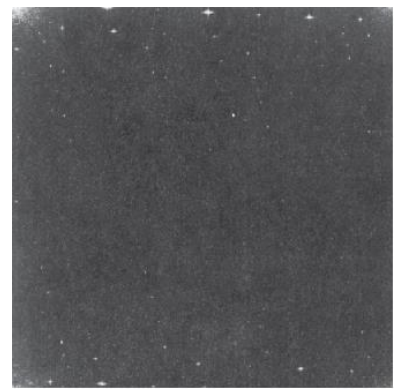
- The restored image is $\hat{f}(x,y) = g(x,y) - w(x,y)\eta(x,y)$
- The weighting function $w(x,y)$ is found so that the variance of $\hat{f}(x,y)$ is minimized for a selected neighborhood.
- One way is to select $w(x,y)$ so that the variance of the estimate $\hat{f}(x,y)$ is minimized over a small local neighborhood of size $(2a+1, 2b+1)$.

$$\sigma^2(x,y) = \frac{1}{(2a+1)(2b+1)} \sum_{t=-b}^{b} \sum_{s=-a}^{a} \left[\hat{f}(x+s, y+t) - \bar{\hat{f}}(x,y) \right]^2$$

- $\partial\sigma^2(x,y)/\partial w(x,y) = 0 \rightarrow$ to select $w(x,y)$

Optimal Notch Filtering

FIGURE 5.21 Uncentered Fourier spectrum of the image in Fig. 5.20(a). (Courtesy of NASA.)

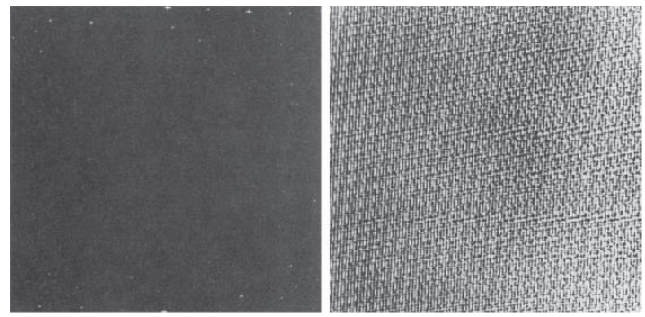


$a = b = 15$

Optimal Notch Filtering

a b

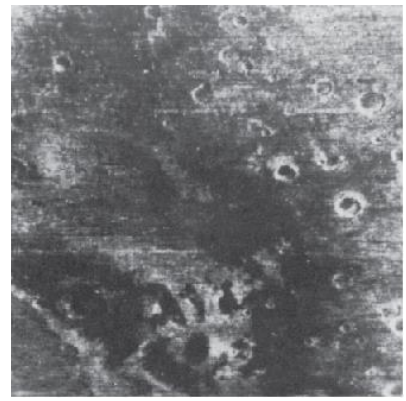
FIGURE 5.22
(a) Fourier spectrum of $N(u,v)$, and
(b) corresponding spatial noise interference pattern, $\eta(x,y)$. (Courtesy of NASA.)



Optimal Notch Filtering

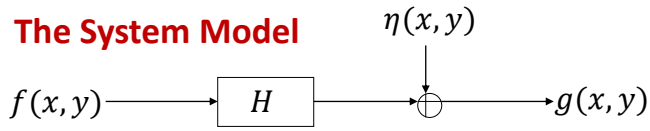
$$\hat{f}(x,y) = g(x,y) - w(x,y)\eta(x,y)$$

FIGURE 5.23
Restored image. (Courtesy of NASA.)



Linear Position-Invariant Degradations

- **The System Model**



$g(x,y)$: the degraded image

$f(x,y)$: the original image

$\eta(x,y)$: additive noise

H: System function which is a linear operator

Linear Position-Invariant Degradations

$g(x,y) = H[f(x,y)] + \eta(x,y)$

Assume H is linear and $\eta(x,y) = 0$

If $g(x,y) = H[f(x,y)]$ is position (space) invariant then

$$H[f(x-\alpha, y-\beta)] = g(x-\alpha, y-\beta)$$

$$\begin{aligned} g(x,y) &= H\left[f(x,y)\right] + \eta(x,y) \\ &= \iint f(\alpha,\beta) h(x-\alpha, y-\beta) d\alpha d\beta + \eta(x,y) \\ &= h(x,y) * f(x,y) + \eta(x,y) \end{aligned}$$

Estimating the Degradation Function

- To estimate the degradation function
  - by **Observation**
  - by **Experimentation**
  - by **Mathematical modeling**

  (Blind deconvolution)

Estimating the Degradation Function

- **By observation**: construct an unblurred image of some **strong signal content**.
- Let the **observed** image be $g_s(x, y)$
- The **processed** image is $\hat{f}_S(x, y)$
- Assume the noise is negligible
- Find the degradation function $H(u, v)$ which is similar to

$$H_s(u,v) = \frac{G_s(u,v)}{\hat{F}_s(u,v)}$$

Estimating the Degradation Function

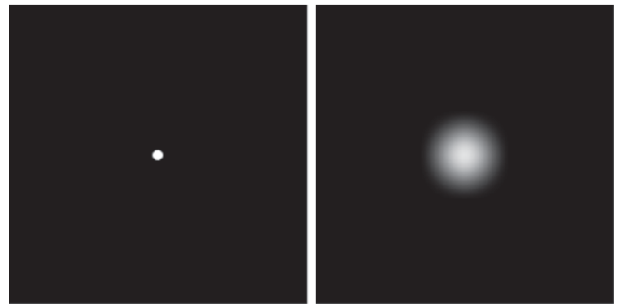
- **By experimentation**: to obtain the impulse response of the degradation by imaging an impulse (small dot of light) using the same system setting.
- The Fourier Transform of an impulse is a constant A
- Let $g(x, y)$ be the observed image.
- Find the degradation function as

$$H(u, v) = G(u, v)/A$$

Estimating the Degradation Function

a b

FIGURE 5.24 Estimating a degradation by impulse characterization. (a) An impulse of light (shown magnified). (b) Imaged (degraded) impulse.



Estimating the Degradation Function

- **By Modeling**: through experience
- The physical characteristic of turbulence:

$$H(u,v) = e^{-k(u^2+v^2)^{5/6}}$$

where k is a constant which depends on the nature of turbulence.

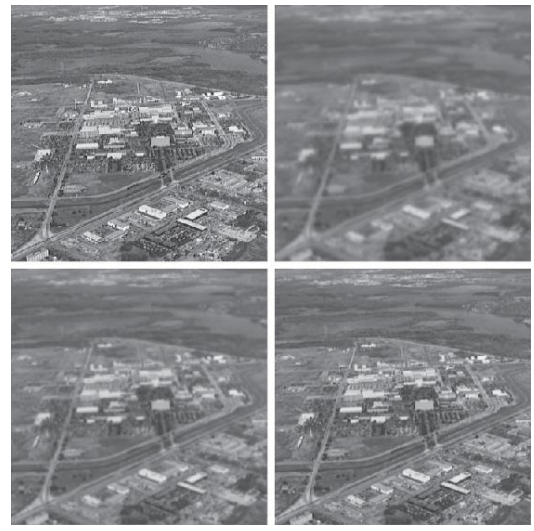
- It is similar to a Gaussian Low-pass function

k = 0.0025 (severe turbulence)

k = 0.001

k = 0.00025 (low turbulence)

Estimating the Degradation Function



a b
c d

FIGURE 5.25 Modeling turbulence. (a) No visible turbulence. (b) Severe turbulence, $k = 0.0025$. (c) Mild turbulence, $k = 0.001$. (d) Low turbulence, $k = 0.00025$. All images are of size 480×480 pixels. (Original image courtesy of NASA.)

Estimating the Degradation Function- by ~~Modeling~~

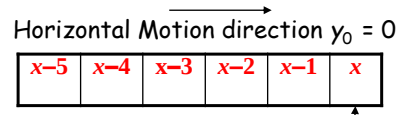
- **Example:**

  Image $f(x,y)$ is blurred by **uniform motion**.

  - Suppose $x_0(t)$ and $y_0(t)$ are the time varying components of motion in x, y directions.
  - The total exposure at any point of the film is obtained by integrating the instantaneous exposure over the time interval during which the shutter is opened.

Estimating the Degradation Function- by ~~Modeling~~

- Assume T is **duration of exposure**, the blurred image $g(x,y)$ is



$$g(x,y) = \int_0^T f[x - x_0(t), y - y_0(t)]dt$$

$x_0(t) = at/T$, T = duration
a = displacement

- Fourier Transform of $g(x,y)$ is

$$G(u,v) = \int\int g(x,y)e^{-j2\pi(ux+vy)}dxdy$$

$$= \int\int \left[\int_0^T f[x - x_0(t), y - y_0(t)]dt\right] e^{-j2\pi(ux+vy)}dxdy$$

Estimating the Degradation Function- by ~~Modeling~~

- Reverse the order of integration

$$G(u,v)=\int_0^T\left[\int_{-\infty}^{\infty}\int f[x-x_0(t),y-y_0(t)]e^{-j2\pi(ux+vy)}dxdy\right]dt$$

- The term inside the outer brackets is Fourier transform of the displacement
- Therefore

$$G(u,v)=\int_0^T F(u,v)e^{-j2\pi(ux_0(t)+vy_0(t))}dt$$

$$=F(u,v)\int_0^T e^{-j2\pi(ux_0(t)+vy_0(t))}dt=F(u,v)H(u,v)$$

Estimating the Degradation Function- by ~~Modeling~~

- Assume uniform motion in x direction only, i.e., $x_0(t)=at/T,\ y_0(t)=0$, a is the displacement, T = exposure duration
- Simplify $H(u,v)=\int_0^T e^{-j2\pi(ux_0(t)+vy_0(t))}dt$ as

$$H(u,v)=\int_0^T e^{-j2\pi ux_0(t)}dt$$

$$=\int_0^T e^{-j2\pi uat/T}dt=\frac{T}{\pi ua}\sin(\pi ua)e^{-j\pi ua}$$

- **Problem**: when $u=n/a, H(u,v)=0$

Estimating the Degradation Function- by ~~Modeling~~

- If we allow y-component movement, with the motion given by $y_0 = bt/T$ then the degradation function is

$$H(u,v) = \frac{T}{\pi(ua+vb)}\sin(\pi(ua+vb))e^{-j\pi(ua+vb)}$$

Estimating the Degradation Function- by ~~Modeling~~

a b

FIGURE 5.26
(a) Original image. (b) Result of blurring using the function in Eq. (5-77) with $a = b = 0.1$ and $T = 1$.

Inverse Filtering

- With degraded image: $\mathbf{g} = \mathbf{H} \cdot \mathbf{f} + \boldsymbol{\eta}$
- Assuming the noise has been removed, our goal is to find $\hat{\mathbf{f}}$ such that $\mathbf{H}\hat{\mathbf{f}}$ approximates $\mathbf{g}$ in the least squares sense

$$\boldsymbol{\varepsilon} = \mathbf{g} - \mathbf{H} \cdot \hat{\mathbf{f}}$$

$$\|\boldsymbol{\varepsilon}\|^2 = \left\|\mathbf{g} - \mathbf{H} \cdot \hat{\mathbf{f}}\right\|^2 = \left(\mathbf{g} - \mathbf{H} \cdot \hat{\mathbf{f}}\right)^T \left(\mathbf{g} - \mathbf{H} \cdot \hat{\mathbf{f}}\right)$$

Inverse Filtering

- Minimizing $\|\boldsymbol{\varepsilon}\|^2 = J(\hat{\mathbf{f}})$ by using $\partial J(\hat{\mathbf{f}})/\partial \hat{\mathbf{f}}$

$$\partial J(\hat{\mathbf{f}})/\partial \hat{\mathbf{f}} = -2\mathbf{H}^T(\mathbf{g} - \mathbf{H}\hat{\mathbf{f}}) = 0$$

$$\hat{\mathbf{f}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{g}$$

$$= \mathbf{H}^{-1}(\mathbf{H}^T)^{-1}\mathbf{H}^T\mathbf{g} = \mathbf{H}^{-1}\mathbf{g} \text{ (if } \mathbf{H}^{-1} \text{ exists)}$$

- Or we may have

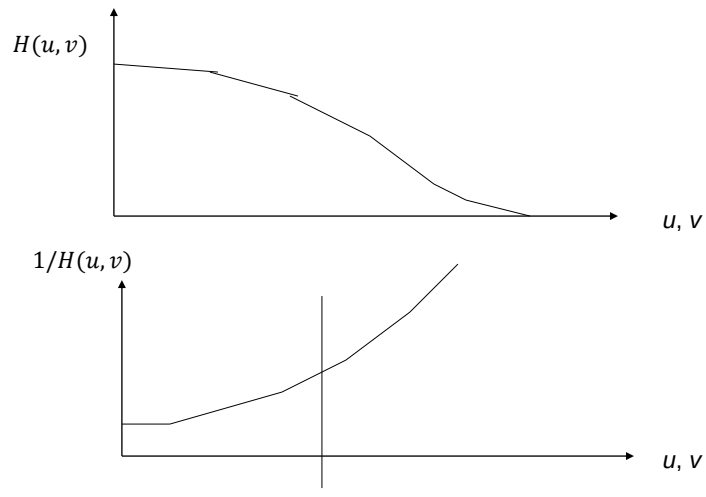
$$\hat{\mathbf{F}}(u,v) = \frac{\mathbf{G}(u,v)}{\mathbf{H}(u,v)}$$

Inverse Filtering

- **Without noise** $\mathbf{G}(u,v) = \mathbf{F}(u,v)\mathbf{H}(u,v)$
- We have $\hat{\mathbf{F}}(u,v) = \dfrac{\mathbf{G}(u,v)}{\mathbf{H}(u,v)} = \mathbf{F}(u,v)$
- **With noise** $\mathbf{G}(u,v) = \mathbf{F}(u,v)\mathbf{H}(u,v) + \mathbf{N}(u,v)$
- We have $\hat{\mathbf{F}}(u,v) = \mathbf{F}(u,v) + \dfrac{\mathbf{N}(u,v)}{\mathbf{H}(u,v)}$
- Restored image

$$\hat{f}(x,y) = \mathcal{F}^{-1}\{\hat{\mathbf{F}}(u,v)\}$$

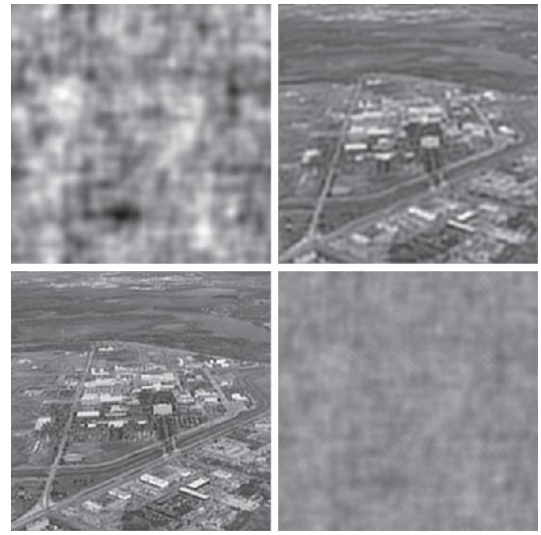
Inverse Filtering



Inverse Filtering

a b
c d

FIGURE 5.27 Restoring Fig. 5.25(b) using Eq. (5-78). (a) Result of using the full filter. (b) Result with H cut off outside a radius of 40. (c) Result with H cut off outside a radius of 70. (d) Result with H cut off outside a radius of 85.



Degradation function:

$$H(u,v) = e^{-k\left(u^2+v^2\right)^{5/6}}$$

$$K = 0.0025, M = N = 480$$

MMSE (Wiener) Filtering

- Incorporate both the degradation function and **statistical characteristics** of noise into restoration
- Considering both the image and noise as random processes, and find an estimate $\hat{f}$ of the uncorrupted image f such that the mean square error between them is minimized
- The error measure is $e^2 = E\left\{\left(f(x,y) - \hat{f}(x,y)\right)^2\right\}$

MMSE (Wiener) Filtering

- Minimizing the error $\partial e^2/\partial h_R(x,y) = 0$ we have $E\{[f(x,y) - \hat{f}(x,y)]g(x',y')\} = 0$ (**Principle of Orthogonality**). For image coordinate pair (x,y) and (x',y'), we assume the **restoration filter** is $h_R(x,y)$, and have

$$E\{f(x,y)g(x',y')\} = \int\int E\{g(\alpha,\beta)g(x',y')\}h_R(x-\alpha,y-\beta)d\alpha d\beta$$

- Assume the **ideal image** and **observed image** are jointly stationary, the expectation term can be expressed as **covariance function** as

$$K_{fg}(x-x',y-y') = \int\int_{-\infty}^{\infty} K_{gg}(\alpha-x',\beta-y')h_R(x-\alpha,y-\beta)d\alpha d\beta$$

MMSE (Wiener) Filtering

- Take Fourier Transform we have

$$H_R(u,v) = W_{fg}(u,v)W_{gg}^{-1}(u,v)$$

with additive noise

$$W_{fg}(u,v) = H^*(u,v)S_f(u,v)$$

$$W_{gg}(u,v) = |H(u,v)|^2 S_f(u,v) + S_\eta(u,v)$$

- $H(u,v)$: the degradation function
- $S_\eta(u,v) = |N(u,v)|^2$ = power spectrum of the noise
- $S_f(u,v) = |F(u,v)|^2$ = power spectrum of the signal

MMSE (Wiener) Filtering

Assume the image and noise are uncorrelated

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)S_f(u,v)}{S_f(u,v)\left|H(u,v)\right|^2 + S_\eta(u,v)} \right] G(u,v)$$

$$= \left[\frac{H^*(u,v)}{\left|H(u,v)\right|^2 + S_\eta(u,v)/S_f(u,v)} \right] G(u,v)$$

$$= \left[\frac{1}{H(u,v)} \frac{\left|H(u,v)\right|^2}{\left|H(u,v)\right|^2 + S_\eta(u,v)/S_f(u,v)} \right] G(u,v)$$

MMSE (Wiener) Filtering

- $H(u,v)$: the degradation function
- $S_\eta(u,v) = |N(u,v)|^2$ = power spectrum of the noise
- $S_f(u,v) = |F(u,v)|^2$ = power spectrum of the undegraded function
- If noise is zero, then it becomes an **inverse filter**.
- For a white noise $|N(u,v)|^2$ = constant then

$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)} \frac{\left|H(u,v)\right|^2}{\left|H(u,v)\right|^2 + K} \right] G(u,v)$$

MMSE (Wiener) Filtering

a b c

FIGURE 5.28 Comparison of inverse and Wiener filtering. (a) Result of full inverse filtering of Fig. 5.25(b). (b) Radially limited inverse filter result. (c) Wiener filter result.

MMSE Filtering

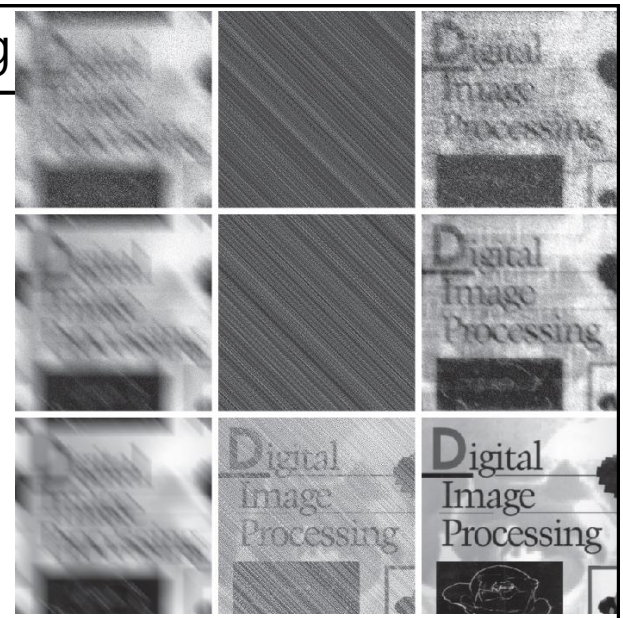


FIGURE 5.29 (a) 8-bit image corrupted by motion blur and additive noise. (b) Result of inverse filtering. (c) Result of Wiener filtering. (d)–(f) Same sequence, but with noise variance one order of magnitude less. (g)–(i) Same sequence, but noise variance reduced by five orders of magnitude from (a). Note in (h) how the deblurred image is quite visible through a "curtain" of noise.

Constrained Least Squares Filtering

- **Difficulty for Wiener filter**: the power spectra of the undegraded image and the noise must be known.
- Only the mean and variance of the noise are required
- Given a noisy image in vector form $\mathbf{g} = \mathbf{H} \cdot \mathbf{f} + \boldsymbol{\eta}$ with $g(x, y)$ of size $M \times N$, and matrix **H** of dimension $MN \times MN$ and **H** is highly sensitive to noise.
- Base the optimality of restoration on a measure of smoothness, such as the 2nd derivative of an image, i.e.,

$$C = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\nabla^2 f(x, y)]^2$$

- Find the minimum of *C* subject to the constraint

$$\left\| \mathbf{g} - \mathbf{H}\hat{\mathbf{f}} \right\|^2 = \|\boldsymbol{\eta}\|^2$$

Constrained Least Squares Filtering

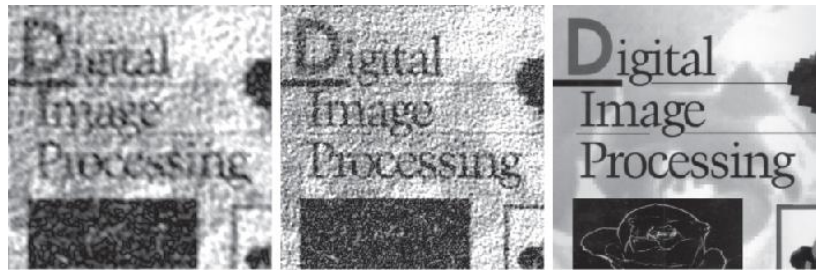
- The frequency domain solution to this optimization problem is

$$\hat{F}(u, v) = \left[\frac{H^*(u, v)}{|H(u, v)|^2 + \gamma |P(u, v)|^2} \right] G(u, v)$$

where $P(u, v)$ is Fourier transform of $p(x, y)$

$$p(x, y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Constrained Least Squares Filtering



a b c

FIGURE 5.30 Results of constrained least squares filtering. Compare (a), (b), and (c) with the Wiener filtering results in Figs. 5.29(c), (f), and (i), respectively.

Constrained Least Squares Filtering

- To compute γ by iteration as follows:

  Step 1. Define the residue vector $\mathbf{r}$: $\mathbf{r} = \mathbf{g} - \mathbf{H}\hat{\mathbf{f}}$

  $\hat{F}(u,v)$ is a function of γ, so is $\mathbf{r}$, $\phi(\gamma) = \mathbf{r}^T\mathbf{r} = \|\mathbf{r}\|^2$ is a monotonically increasing function of γ

  Step 2. Adjust γ so that $\|\mathbf{r}\|^2 = \|\boldsymbol{\eta}\|^2 \pm a$

  where a is an accuracy factor

  Step 3. If $\|\mathbf{r}\|^2 \approx \|\boldsymbol{\eta}\|^2$ then the best solution is found.

Constrained Least Squares Filtering

- Because $\phi(\gamma)$ is monotonic, finding γ is not difficult.

Step 1: Specify an initial γ

Step 2: Compute $\|\mathbf{r}\|^2$

Step 3: if $\|\mathbf{r}\|^2 = \|\boldsymbol{\eta}\|^2 \pm a$ is satisfied then **stop**

Step 4: if $\|\mathbf{r}\|^2 < \|\boldsymbol{\eta}\|^2 - a$ **increase** γ and **go to step 2**.

Step 5: if $\|\mathbf{r}\|^2 > \|\boldsymbol{\eta}\|^2 - a$ **decrease** γ and **go to step 2**.

Constrained Least Squares Filtering

- To compute the $\|\mathbf{r}\|^2$ and $\|\boldsymbol{\eta}\|^2$
- The vector form can also be rewritten as

$$R(u,v) = G(u,v) - H(u,v)\hat{F}(u,v)$$

- Compute the Inverse Fourier transform of $R(u,v)$

$$\|\mathbf{r}\|^2 = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} r^2(x,y)$$

- Consider the variance of the noise over the entire image, using the sample-average method

$$\sigma_\eta^2 = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\eta(x,y) - m_\eta\right] \qquad m_\eta = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\eta(x,y)$$

- So we have $\|\boldsymbol{\eta}\|^2 = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\eta^2(x,y) = MN\left[\sigma_\eta^2 + m_\eta^2\right]$

Constrained Least Squares Filtering

a b

FIGURE 5.31
(a) Iteratively determined constrained least squares restoration of Fig. 5.25(b), using correct noise parameters. (b) Result obtained with wrong noise parameters.

Geometric Mean Filter

Generalized form of Wiener filter:

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2}\right]^{\alpha} \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \beta\left[\frac{S_\eta(u,v)}{S_f(u,v)}\right]}\right]^{1-\alpha} G(u,v)$$

α, β are positive real constant

α = 1 reduces to inverse filter

α = 0 becomes parametric Wiener filter

α = 0, β = 1 standard Wiener filter

α = 1/2, β = 1 spectrum equalization filter

Geometric Mean Filter

- It is also called **Rubber sheet transform**, which consists of two operations:

1) **Spatial transformation**: **rearrangement** of the pixels (**locations**) on the image.
2) **Gray-level interpolation**: **assignment** of **gray levels** to pixels in the spatially transformed image.

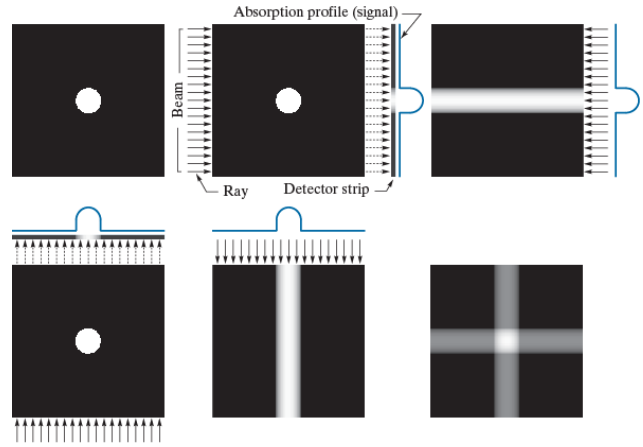
Image Reconstruction from Projections

- Reconstructing an image from a series of **projections**.
- Suppose image (2-D area) represents an **cross-section** of human body.
- Fig. 5.32 shows a single object on a uniform background, where the background in the image represents soft, uniform tissue, whereas the round object is a tumor with higher absorption characteristics.

  e.g., X-ray Computed Tomography (CT)
- Project the 1-D signal back across a **2-D area**: duplicate the 1-D signal in all columns of the reconstructed image.

→ **Back projection**

Image Reconstruction from Projections



a b c
d e f

FIGURE 5.32
(a) Flat region with a single object. (b) Parallel beam, detector strip, and profile of sensed 1-D absorption signal. (c) Result of back-projecting the absorption profile. (d) Beam and detectors rotated by 90°. (e) Backprojection. (f) The sum of (c) and (e), intensity-scaled. The intensity where the backprojections intersect is twice the intensity of the individual back-projections.

Image Reconstruction from Projections

- Rotate the source-detector pair by 90 degree
- A back-projection in vertical direction
- Assume the number of back-projections increases
- Adding all the back-projections

Image Reconstruction from Projections

a b c
d e f

FIGURE 5.33
(a) Same as Fig. 5.32(a). (b)-(e) Reconstruction using 1, 2, 3, and 4 backprojections 45° apart. (f) Reconstruction with 32 backprojections 5.625° apart (note the blurring).

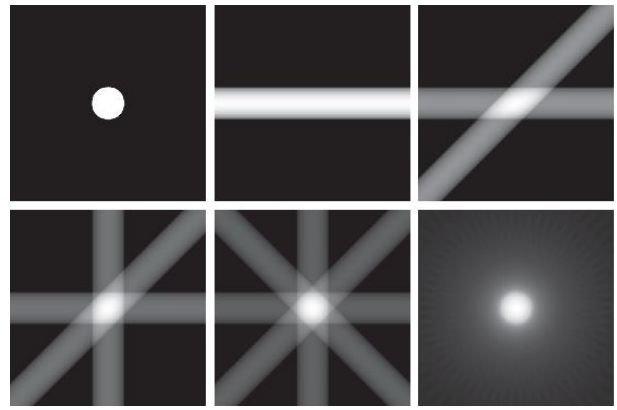
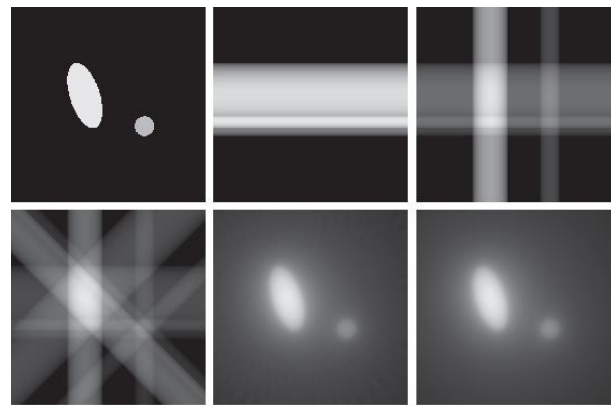


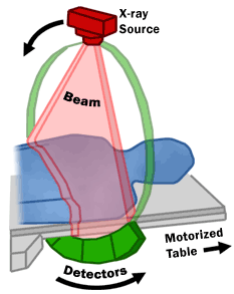
Image Reconstruction from Projections

a b c
d e f

FIGURE 5.34
(a) Two objects with different absorption characteristics. (b)–(d) Reconstruction using 1, 2, and 4 backprojections, 45° apart. (e) Reconstruction with 32 backprojections, 5.625° apart. (f) Reconstruction with 64 backprojections, 2.8125° apart.



Principal of Computed Tomography



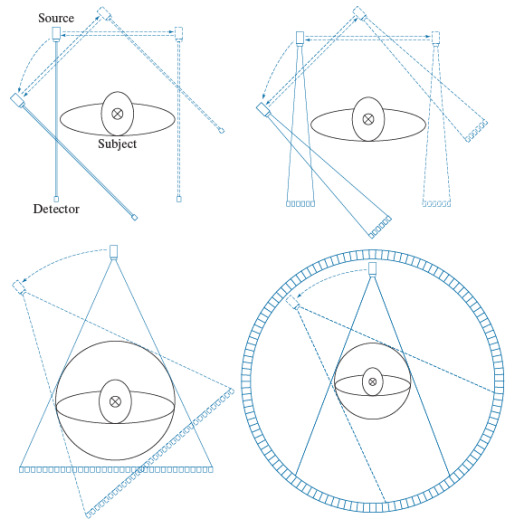
Principal of Computed Tomography

- X-ray Computed Tomography → obtain a 3-D representation of the internal structure of an object by X-raying the object from many directions.
- A set of 2-D projections can be used to reconstruct the 3-D structure.
- 1-D projection CT is more practical due to its reduced amount of detectors and computation.
- **Radon transform**: the mathematical concept of CT was invented by Johann Radon in 1917.
- Allan Cormack in 1963 applied the theory to build a CT prototype.
- Godfrey N. Hounsfield and his colleagues built the first medical CT machine in 1970
- Cormack and Hounsfield shared the 1979 Nobel Prize in Medicine

Principal of Computed Tomography

- 1st generation CT scanner: a “pencil” X-ray beam and a **single** detector. The **source-detector pair** translates and then rotates as shown in Fig. 5.35(a)
- 2nd generation CT scanner: **a fan-shape beam**, multiple detectors and fewer translations [see Fig. 5.35(b)].
- 3rd generation CT scanner: **a bank of detectors** to cover the entire field of view of a wider beam (no translation required [see Fig. 5.35(c)].
- 4th generation CT scanner: **a circular ring of detectors** to cover the entire field of view of a wider beam (no translation required) [see Fig. 5.35(d)].

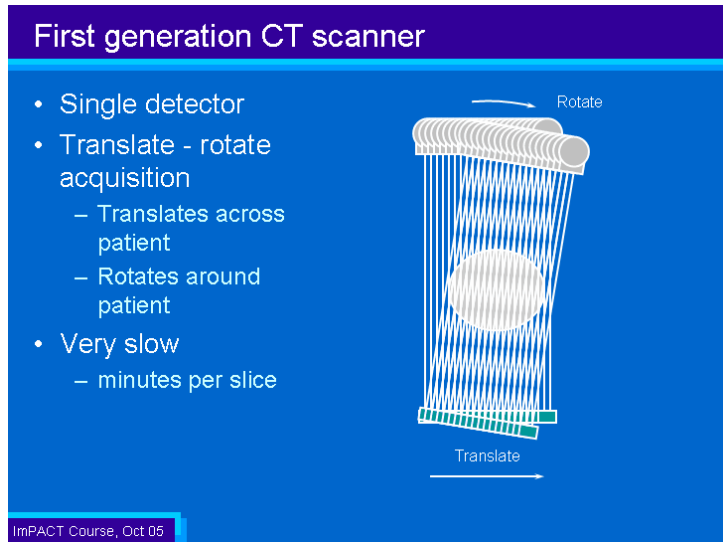
Principal of Computed Tomography



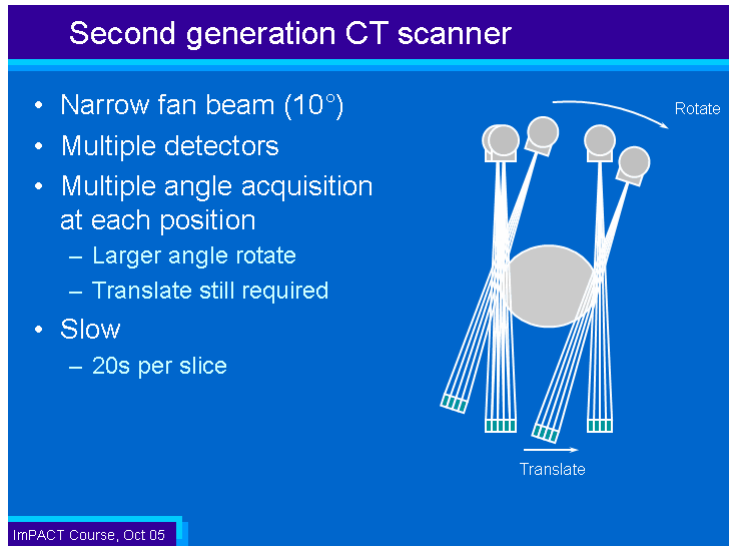
a b
c d

FIGURE 5.35
Four generations of CT scanners. The dotted arrow lines indicate incremental linear motion. The dotted arrow arcs indicate incremental rotation. The cross-mark on the subject's head indicates linear motion perpendicular to the plane of the paper. The double arrows in (a) and (b) indicate that the source/detector unit is translated and then brought back into its original position.

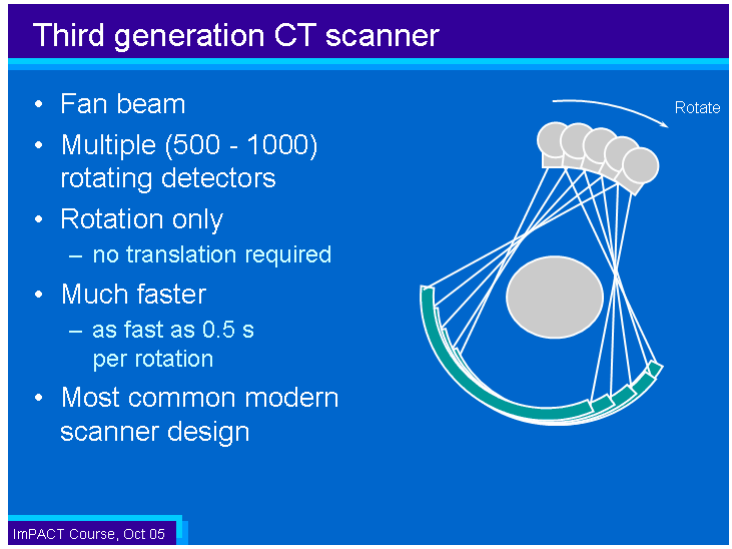
First-Generation CT Scanner



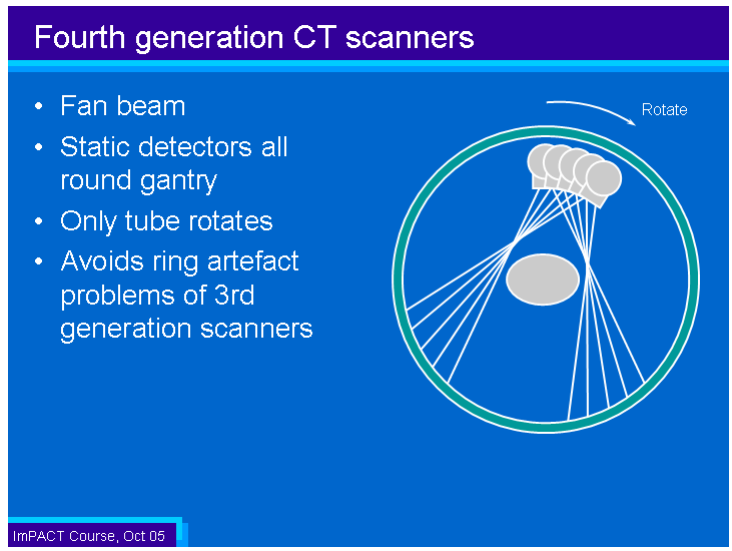
Second-Generation CT Scanner



Third-Generation CT Scanner



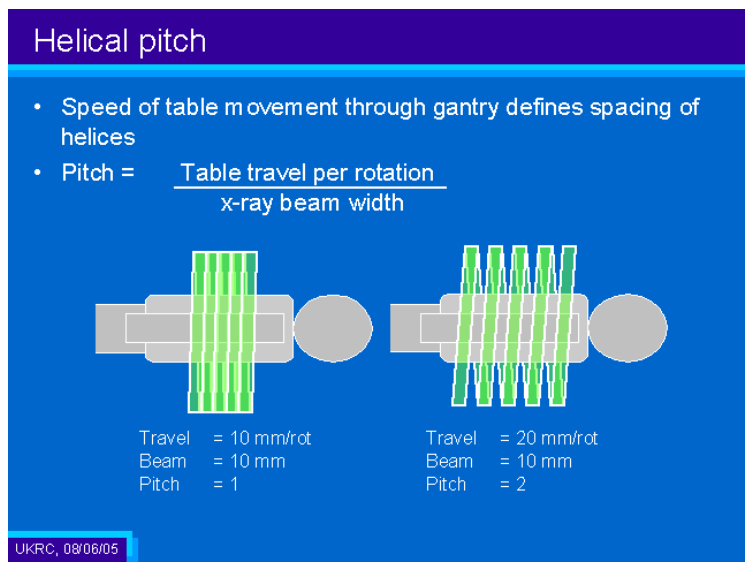
Fourth-Generation CT Scanner



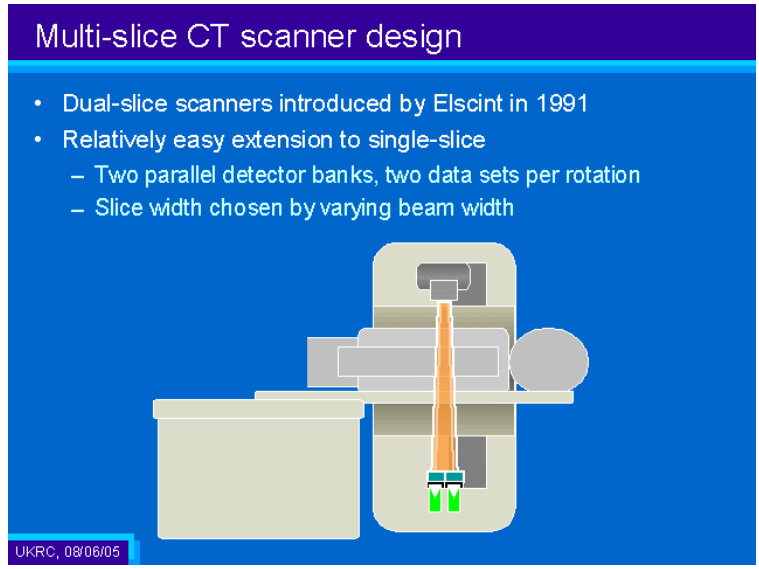
Principal of Computed Tomography

- 5th generation **Electron Beam** CT (EBCT): no mechanical motion required. Tungstan anodes circles the patient which generate a fan beam and excites a ring of detectors.
- The patient is kept stationary during the scanning. The scanning is halted while the position of patient is incremented.
- 6th generation **helical** CT scanner : the source/detector pair rotates continuously 360° while the patient is moved perpendicular to the scan.
- 7th generation **Multi-slice** CT scanner: "thick" fan beam are used in conjunction to the parallel banks of detectors to collect volumetric CT data simultaneously.
- It generates 3-D cross-sectional "slabs" rather than single cross-sectional images. It utilizes X-ray tube more efficiently and leads to less X-ray dosage.

Six-Generation CT Scanner



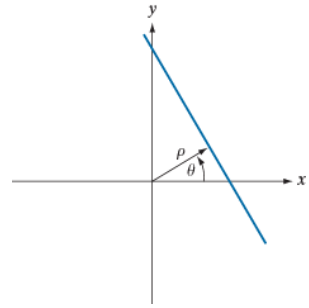
Seven-Generation CT Scanner



Projections and the Radon Transform

- CT imaging: X-ray, SPECT (Single Photon Emission CT), PET (Positron Emission Tomography), MRI
- A straight line in **Cartesian coordinate** $y = ax + b$ can be described in **polar coordinate** as $x\cos\theta + y\sin\theta = \rho$

FIGURE 5.36 Normal representation of a line.



Projections and the Radon Transform

- A projection of **a parallel beam** can be modeled as a set of lines as Fig. 5.37 shows
- An arbitrary point in the projection signal can be given by **a ray sum** along the line $x\cos\theta_k + y\sin\theta_k = \rho_j$ (i.e., $L(\rho_j, \theta_k)$)

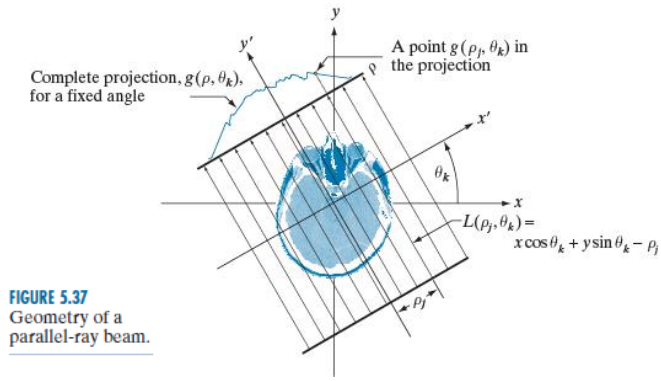


FIGURE 5.37 Geometry of a parallel-ray beam.

Projections and the Radon Transform

- The ray-sum is a line integral

$$g(\rho_j, \theta_k) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta_k + y\sin\theta_k - \rho_j)dxdy$$

- The δ function indicates that the integral is computed along the line $x\cos\theta_k + y\sin\theta_k = \rho_j$. We consider all θ and ρ as

$$g(\rho, \theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - \rho)dxdy$$

- In the discrete case, it can be described as

$$g(\rho, \theta) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\delta(x\cos\theta + y\sin\theta - \rho)$$

where x, y, θ, and ρ are discrete variable.

- **Radon transform**: It generates the projection of $f(x,y)$ along an arbitrary line in the *xy*-plane i.e., $R\{f(x,y)\} = g(\rho, \theta)$.

Projections and the Radon Transform

Example 5.17:

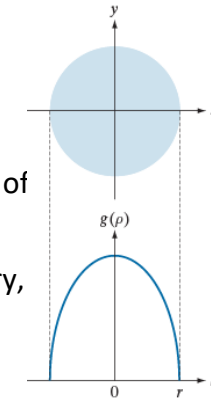
$$f(x,y)=\begin{cases}A & x^2+y^2\le r^2\\ 0 & \text{otherwise}\end{cases}$$

where A is a constant, r is the radius of the object.

Since the object is circularly symmetry, the projection is the same for all directions.

Let θ = 0 we have

$$g(\rho,\theta=0)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x,y)\delta(x-\rho)dxdy=\int_{-\infty}^{\infty}f(\rho,y)dy$$



a
b

FIGURE 5.38 (a) A disk and, (b) a plot of its Radon transform, derived analytically. Here we were able to plot the transform because it depends only on one variable. When g depends on both ρ and θ, the Radon transform becomes an image whose axes are ρ and θ, and the intensity of a pixel is proportional to the value of g at the location of that pixel.

Projections and the Radon Transform

Example 5.17 (cont.)

- Since $g(\rho,\theta)=0$ when $|\rho|>r$
- The integral can be evaluated between $-r\le\rho\le r$ or from $y=-\sqrt{r^2-\rho^2}$ to $y=\sqrt{r^2-\rho^2}$
- So we have

$$g(\rho,\theta)=\int_{-\sqrt{r^2-\rho^2}}^{\sqrt{r^2-\rho^2}}f(\rho,\theta)dy=\int_{-\sqrt{r^2-\rho^2}}^{\sqrt{r^2-\rho^2}}Ady$$

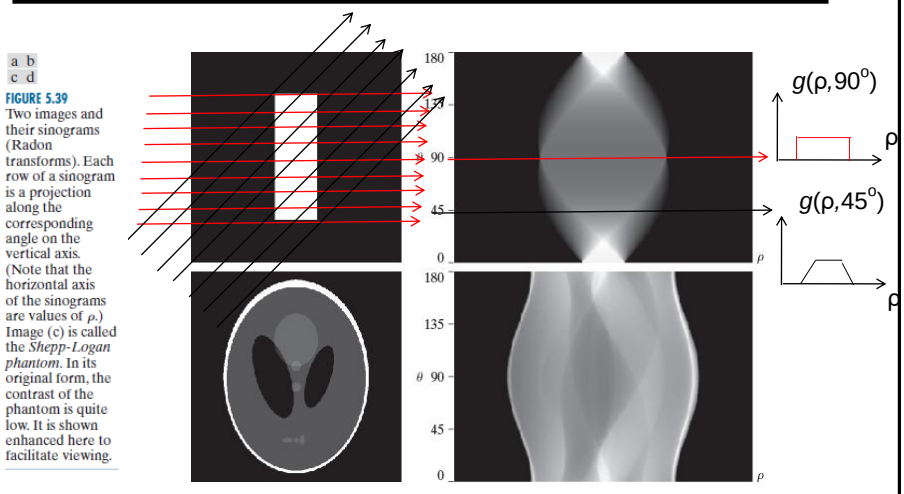
or

- $g(\rho,\theta)=g(\rho)=\begin{cases}2A\sqrt{r^2-\rho^2} & |\rho|\le r\\ 0 & \text{otherwise}\end{cases}$
- g is independent of θ because the object is symmetry about the origin.

Projections and the Radon Transform

- The **Radon transform** $g(\rho,\theta)$ is displayed as an image with ρ and θ as **rectilinear coordinate**, it is called a **sinogram** as shown in Fig. 5.39. (similar to Fourier Transform).
- Sinogram is **symmetric** in both directions about the center of image → object is symmetric and parallel to *x* and *y* axes.
- Sinogram is **smooth** → object has a uniform intensity.
- CT→ Obtain a 3-D representation of a volume from its projections.
- Back-project each projection and then sum all the projections to generate the image (slice).
- Stacking all resulting images (slices) → 3-D rendering of the volume.

Projections and the Radon Transform



a b
c d

FIGURE 5.39 Two images and their sinograms (Radon transforms). Each row of a sinogram is a projection along the corresponding angle on the vertical axis. (Note that the horizontal axis of the sinograms are values of ρ.) Image (c) is called the *Shepp-Logan phantom*. In its original form, the contrast of the phantom is quite low. It is shown enhanced here to facilitate viewing.

Projection:

$f(x,y) \longrightarrow g(\rho,\theta_k)$

2-D 1-D

Back-projection:

$f_\theta(x,y) \longleftarrow g(\rho,\theta_k)$

2-D 1-D

Projections and the Radon Transform

- *A* **single point** $g(\rho_j, \theta_k)$ of the complete projection $g(\rho, \theta_k)$ with fixed θ_k as shown in Fig. 5.37.
- Forming part of an image by **back-projecting** this single point $g(\rho_j, \theta_k) \rightarrow$ Copying the line $L(\rho_j, \theta_k)$ onto the image where the value of each point in that line is $g(\rho_j, \theta_k)$.
- Repeat the process for all ρ_j (fixed θ_k) we have

$$f_{\theta_k}(x, y) = g(\rho, \theta_k) = g(x\cos\theta_k + y\sin\theta_k, \theta_k)$$

- The image formed from a single **back-projection** obtained at an angle θ is $f_\theta(x, y) = g(x\cos\theta + y\sin\theta, \theta)$
- We form the final image by integrating over all the back-projected images $f(x, y) = \int_0^\pi f_\theta(x, y) d\theta$
- Or in the discrete case $f(x, y) = \sum_{\theta=0}^{\pi} f_\theta(x, y)$

101

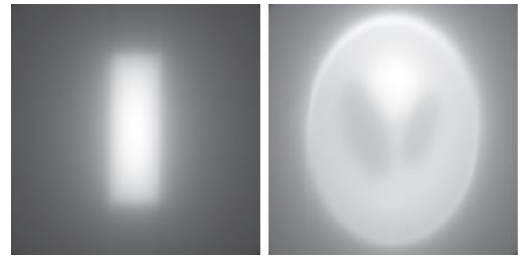
Projections and the Radon Transform

Example 5.18

$\sum_{\theta=0}^{\pi} f_\theta(x, y)$ is used to generate the back-projected image from projections $g(\rho, \theta)$ as shown in Figs. 5.32~5.34 and Fig. 5.40. It is called a **laminogram** which is only an approximation of original $f(x, y)$.

a b

FIGURE 5.40 Backprojections of the sinograms in Fig. 5.39.



The Fourier-Slice Theorem

- 1-D transform of a projection with respect to ρ is

$$G(\omega,\theta)=\int_{-\infty}^{\infty} g(\rho,\theta)e^{-j2\pi\omega\rho}d\rho$$

$$G(\omega,\theta)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta+y\sin\theta-\rho)e^{-j2\pi\omega\rho}dxdyd\rho$$

$$G(\omega,\theta)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\left[\int_{-\infty}^{\infty}\delta(x\cos\theta+y\sin\theta-\rho)e^{-j2\pi\omega\rho}d\rho\right]dxdy$$

$$G(\omega,\theta)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi\omega(x\cos\theta+y\sin\theta)}dxdy$$

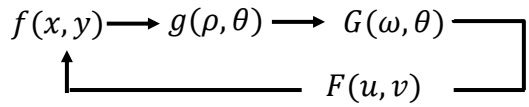
The Fourier-Slice Theorem

- By letting $u=\omega\cos\theta$ and $v=\omega\sin\theta$, we have

$$G(\omega,\theta)=\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy\right]_{u=\omega\cos\theta,v=\omega\sin\theta}$$

- Or

$$G(\omega,\theta)=\left[F(u,v)\right]_{u=\omega\cos\theta,v=\omega\sin\theta}=F(\omega\cos\theta,\omega\sin\theta)$$



- **Fourier-slice transform**: The Fourier transform of a projection is a slice of the 2-D Fourier transform of the region from which the projection was obtained

The Fourier-Slice Theorem

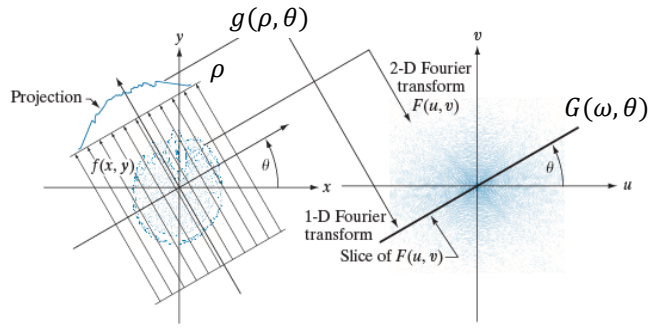


FIGURE 5.41 Illustration of the Fourier-slice theorem. The 1-D Fourier transform of a projection is a slice of the 2-D Fourier transform of the region from which the projection was obtained. Note the correspondence of the angle θ.

$$G(\omega,\theta)=\int_{-\infty}^{\infty} g(\rho,\theta)e^{-j2\pi\omega\rho}d\rho$$

$$G(\omega,\theta)=\left[F(u,v)\right]_{u=\omega\cos\theta,v=\omega\sin\theta}=F(\omega\cos\theta,\omega\sin\theta)$$

Reconstruction Using Parallel-Beam Filtered BP

- 2-D inverse Fourier transform

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

- By letting $u=\omega\cos\theta$ and $v=\omega\sin\theta$ and $dudv=\omega d\omega d\theta$

$$f(x,y)=\int_{0}^{2\pi}\int_{-\infty}^{\infty}F(\omega\cos\theta,\omega\sin\theta)e^{j2\pi\omega(x\cos\theta+y\sin\theta)}\omega d\omega d\theta$$

$$f(x,y)=\int_{0}^{2\pi}\int_{-\infty}^{\infty}G(\omega,\theta)e^{j2\pi\omega(x\cos\theta+y\sin\theta)}\omega d\omega d\theta$$

- Since $G(\omega,\theta+\pi)=G(-\omega,\theta)$, we have

$$f(x,y)=\int_{0}^{\pi}\int_{-\infty}^{\infty}|\omega|G(\omega,\theta)e^{j2\pi\omega(x\cos\theta+y\sin\theta)}d\omega d\theta$$

$G(-\omega,\theta)=G^*(\omega,\theta)$
$G(\omega,\theta+\pi)=G^*(\omega,\theta)$ ➡ $G(\omega,\theta+\pi)=G(-\omega,\theta)$

Reconstruction Using Parallel-Beam Filtered BP

- Integral with respect to ω, the term $x\cos\theta + y\sin\theta$ is a constant, treated as $x\cos\theta + y\sin\theta = \rho$

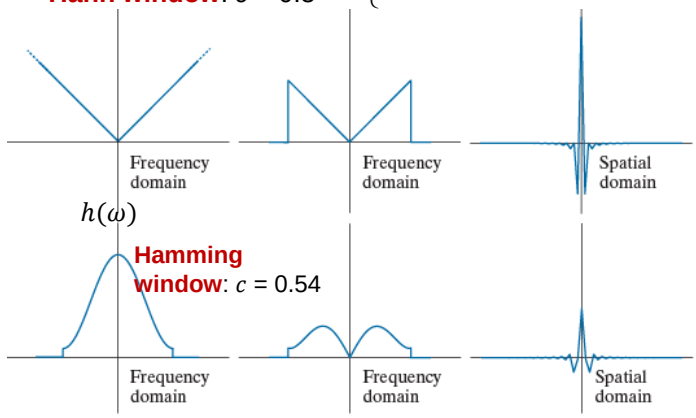
$$f(x,y) = \int_0^{\pi} \left[\int_{-\infty}^{\infty} |\omega| G(\omega,\theta) e^{j2\pi\omega\rho} d\omega \right]_{\rho = x\cos\theta + y\sin\theta} d\theta$$

- The added term $|\omega|$ is recognized as a 1-D filter function → **a ramp filter** which is not integrable (as shown in Fig. 5.42) → Inverse FT of $|\omega|$ is undefined
- Use a **box** (or **window**) to band-limit the ramp filter.
- **Windowing** the ramp so that it becomes zero outside a defined frequency interval (as shown in Fig. 5.42(d)).
- → **Filtered back-projection**

Reconstruction Using Parallel-Beam Filtered BP

$$\text{window function: } h(\omega) = \begin{cases} c + (c-1)\cos\dfrac{2\pi\omega}{M-1} & 0 \le \omega \le (M-1) \\ 0 & \text{otherwise} \end{cases}$$

Hann window: $c = 0.5$



a b c
d e f
FIGURE 5.42
(a) Frequency domain ramp filter transfer function. (b) Function after band-limiting it with a box filter. (c) Spatial domain representation. (d) Hamming windowing function. (e) Windowed ramp filter, formed as the product of (b) and (d). (f) Spatial representation of the product. (Note the decrease in ringing.)

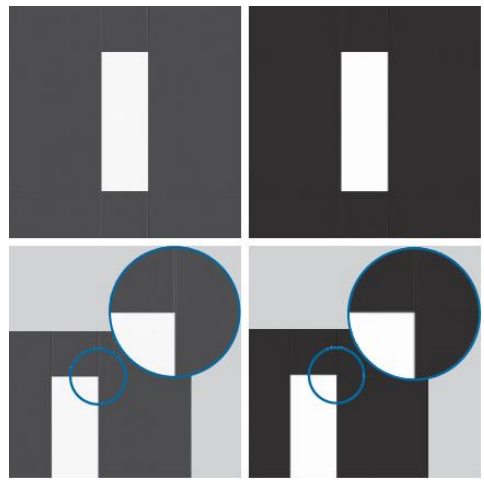
Reconstruction Using Parallel-Beam Filtered BP

- $G(\omega, \theta)$ is the 1-D transform of $g(\rho, \theta)$ which is a **single** projection obtained at a fixed angle θ.
- The complete **back-projected image** $f(x, y)$ is obtained by the following steps:

1. Compute the 1-D Fourier Transform of each single projection at a fixed angle θ, i.e., $g(\rho, \theta)$.
2. Multiply each Fourier Transform $G(\omega, \theta)$ by the filter function $|\omega|$ which is multiplied by a suitable windowing function.
3. Obtain the inverse 1-D Fourier Transform of each filtered transform.
4. Integral (sum) all the 1-D inverse transform from step 3.

Reconstruction Using Parallel-Beam Filtered BP

Example



a b
c d

FIGURE 5.43 Filtered backprojections of the rectangle using (a) a ramp filter, and (b) a Hamming windowed ramp filter. The second row shows zoomed details of the images in the first row. Compare with Fig. 5.40(a).

Reconstruction Using Parallel-Beam Filtered BP

Example

a b

FIGURE 5.44 Filtered backprojections of the head phantom using (a) a ramp filter, and (b) a Hamming windowed ramp filter. Compare with Fig. 5.40(b)

Reconstruction Using Parallel-Beam Filtered BP

Reconstruction in the spatial domain:

Let $s(\rho) = \Im^{-1}\{|\omega|\}$ and $*$ denote the spatial convolution

$$f(x,y) = \int_0^{\pi}\left[\int_{-\infty}^{\infty}|\omega|G(\omega,\theta)e^{2\pi\omega\rho}d\omega\right]_{\rho=x\cos\theta+y\sin\theta}d\theta$$

$$= \int_0^{\pi}\left[s(\rho)*g(\rho,\theta)\right]_{\rho=x\cos\theta+y\sin\theta}d\theta$$

$$= \int_0^{\pi}\left[\int_{-\infty}^{\infty}g(\rho',\theta)s(x\cos\theta+y\sin\theta-\rho')d\rho'\right]d\theta$$

- Individual back-projection at angle θ can be obtained by convolving the corresponding projection of $g(\rho,\theta)$ and $s(\rho)$
- $s(\rho)$: Inverse Fourier Transform of the **ramp filter**.

Reconstruction Using Fan-Beam Filtered BP

Let $p(\alpha, \beta)$ denote the **fan beam projection**.

α is the **angular position** of a particular detector measured with respect to the center ray. β is the **angular displacement** of the source measured with respect to the *y*-axis

$L(\rho, \theta)$ is a normal form **projection ray** of $p(\alpha, \beta)$.

with $\theta = \alpha + \beta$

and $\rho = D \cdot \sin\alpha$

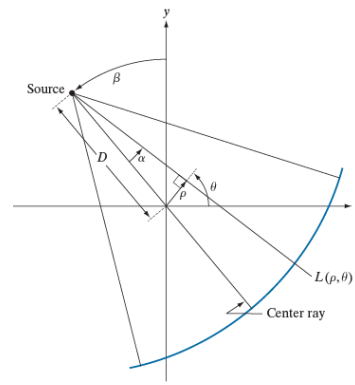


FIGURE 5.45 Basic fan-beam geometry. The line passing through the center of the source and the origin (assumed here to be the center of rotation of the source) is called the *center ray*.

Reconstruction Using Fan-Beam Filtered BP

- We assume that object encompass within a circular area of radius T about the origin of plane, $g(\rho, \theta) = 0$ for $|\rho| > T$
- The convolution back-projection formula for the parallel-imaging becomes

$$f(x,y) = \frac{1}{2}\int_0^{2\pi}\int_{-T}^{T} g(\rho,\theta)s(x\cos\theta + y\sin\theta - \rho)d\rho d\theta$$

Ray of parallel-beam (ρ, θ)

  - Integrating with respect to α and β
- Let $x = r\cos\varphi$, $y = r\sin\varphi$, and $x\cos\theta + y\sin\theta = r\cos(\theta - \varphi)$,
- We have

$$f(x,y) = \frac{1}{2}\int_0^{2\pi}\int_{-T}^{T} g(\rho,\theta)s(r\cos(\theta-\varphi) - \rho))d\rho d\theta$$

Reconstruction Using Fan-Beam Filtered BP

- Using $\theta = \alpha + \beta$ and $\rho = D\sin\alpha$, we change the integral from $f(x,y)$ to $f(r,\varphi)$ based on $x = r\cos\varphi$, $y = r\sin\varphi$,

$$f(r,\varphi) = \frac{1}{2}\int_{-\alpha}^{2\pi-\alpha}\int_{\sin^{-1}(-T/D)}^{\sin^{-1}(T/D)} g(D\sin\alpha, \alpha+\beta)\cdot s\left[r\cos(\beta+\alpha-\varphi) - D\sin\alpha\right] D\cos\alpha\, d\alpha\, d\beta$$

Ray of Fan-beam (α, β)

- Fig. 5.46 shows that $\sin^{-1}(T/D)$ has maximum value α_m
- $p(\alpha,\beta) = g(\rho,\theta)$ or $p(\alpha,\beta) = g(D\sin\alpha, \alpha+\beta)$

$$f(r,\varphi) = \frac{1}{2}\int_{0}^{2\pi}\int_{-\alpha_m}^{\alpha_m} p(\alpha,\beta)\cdot s\left[r\cos(\beta+\alpha-\varphi) - D\sin\alpha\right] D\cos\alpha\, d\alpha\, d\beta$$

Reconstruction Using Fan-Beam Filtered BP

$\sin^{-1}(T/D)$ has maximum value of α_m corresponding to $|\rho| > T$

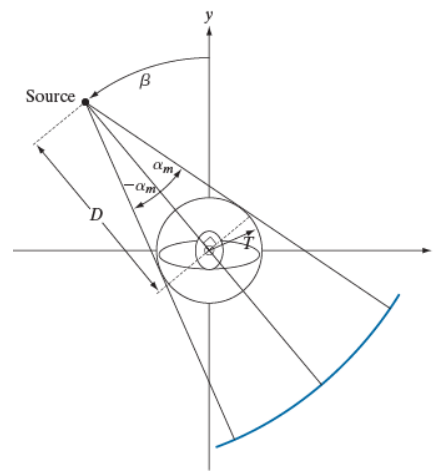


FIGURE 5.46 Maximum value of α needed to encompass a region of interest.

Reconstruction Using Fan-Beam Filtered BP

$r\cdot\cos(\beta+\alpha-\varphi)-D\cdot\sin\alpha=R\cdot\sin(\alpha'-\alpha)$

R and α' are determined by r, φ, β

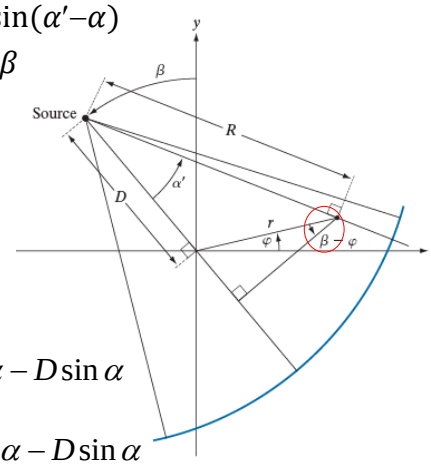


FIGURE 5.47 Polar representation of an arbitrary point on a ray of a fan beam.

$$
\begin{aligned}
&r\cos(\beta-\varphi+\alpha)-D\sin\alpha\\
&=\underbrace{r\cos(\beta-\varphi)}_{=R\sin\alpha'}\cos\alpha-\underbrace{r\sin(\beta-\varphi)}_{=R\cos\alpha'-D}\sin\alpha-D\sin\alpha\\
&=R\sin\alpha'\cos\alpha-R\cos\alpha'\sin\alpha+D\sin\alpha-D\sin\alpha\\
&=R\sin(\alpha'-\alpha)
\end{aligned}
$$

Reconstruction Using Fan-Beam Filtered BP

- By substituting

$$f(r,\varphi)=\frac{1}{2}\int_0^{2\pi}\int_{-\alpha_m}^{\alpha_m}p(\alpha,\beta)\cdot s\left[R\sin(\alpha'-\alpha)\right]D\cos\alpha\,d\alpha\,d\beta$$

$$s(R\sin\alpha)=\left(\frac{\alpha}{R\sin\alpha}\right)^2 s(\alpha)$$

$$f(r,\varphi)=\frac{1}{2}\int_0^{2\pi}\frac{1}{R^2}\left[\int_{-\alpha_m}^{\alpha_m}q(\alpha,\beta)h(\alpha'-\alpha)\,d\alpha\right]d\beta$$

where $h(\alpha)=\frac{1}{2}\left(\frac{\alpha}{\sin\alpha}\right)^2 s(\alpha)$ and $q(\alpha,\beta)=p(\alpha,\beta)\cdot D\cos\alpha$

- A **fan-beam projection** p taken at angle β corresponding to **parallel beam projection** g taken at corresponding angle θ

$$p(\alpha,\beta)=g(\rho,\theta)=g(D\cdot\sin\alpha,\alpha+\beta)$$

Reconstruction Using Fan-Beam Filtered BP

$$p(\alpha,\beta) = g(\rho,\theta) = g(D \cdot \sin\alpha\,, \alpha+\beta)$$

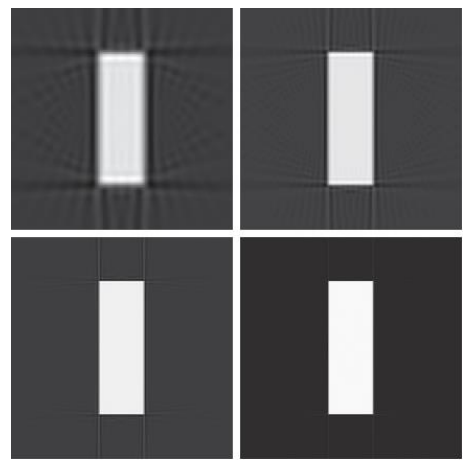
- $\Delta\beta$ is **angular increment** between two successive **fan-beam projections**.
- $\Delta\alpha$is **angular increment** between two rays $\rightarrow$ it determines the number of samples in each projection.
- Let $\Delta\alpha = \Delta\beta = \gamma,\ \alpha = n\gamma\,,\ \beta = m\gamma$

$$p(n\gamma, m\gamma) = g[D \cdot \sin n\gamma\,, (m+n)\gamma]$$

- The nth ray in the mth radial projection

  = the nth ray in the $(m+n)$th parallel projection

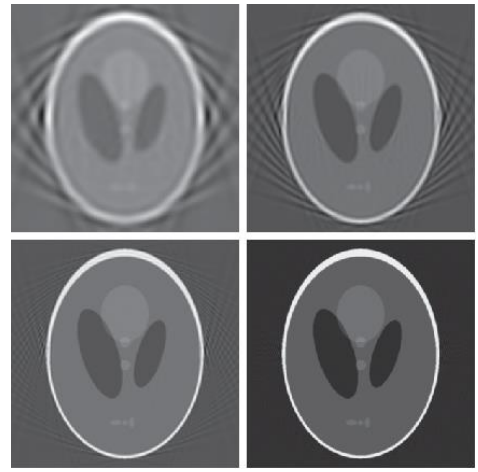
Reconstruction Using Fan-Beam Filtered BP



a b
c d

FIGURE 5.48
Reconstruction of the rectangle image from filtered fan backprojections. (a) 1° increments of α and β. (b) 0.5° increments. (c) 0.25° increments. (d) 0.125° increments. Compare (d) with Fig. 5.43(b).

Reconstruction Using Fan-Beam Filtered BP



a b
c d

FIGURE 5.49
Reconstruction of the head phantom image from filtered fan backprojections. (a) 1° increments of α and β. (b) 0.5° increments. (c) 0.25° increments. (d) 0.125° increments. Compare (d) with Fig. 5.44(b).