Chapter-5

Image Restoration & Reconstruction

Noise and Images

- Images are normally degraded by stochastic variations known as noise:
  - Digital images are quantized: Typically 256 intensity levels are recorded
  - Lossy compression introduces errors
  - Imperfect sensors also introduce noise
- Noise can be additive, multiplicative or impulsive

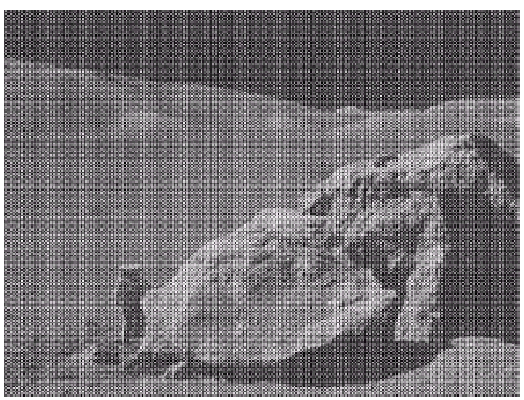
- Noise can be seen as a probability density function (PDF) e.g.,
  - Uniform
  - Gaussian
- Adding noise to image
  - A random number seed must be specified
  - Each pixel in the noisy image is the sum of the true pixel value and a random, uniform or Gaussian distributed noise value

Image Restoration

Attempts to restore images that have been degraded

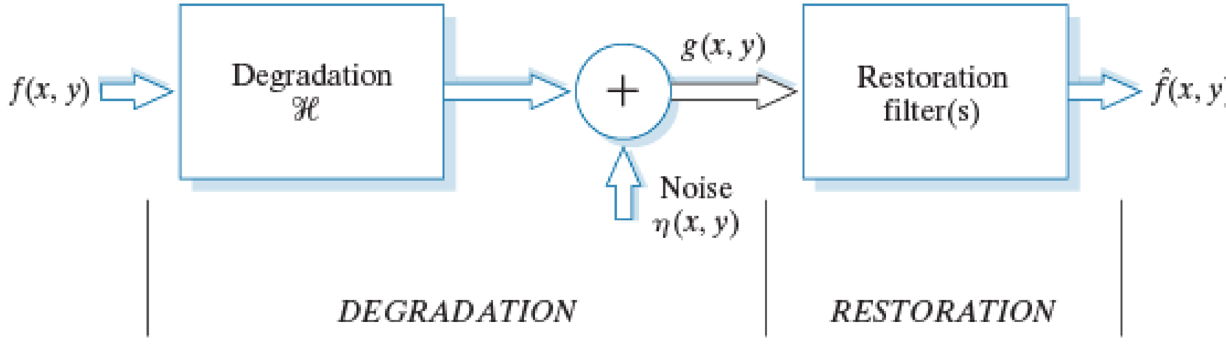
- Recover an image that has been degraded by using a prior knowledge of the degradation phenomenon
- Identify cause of degradation and attempt to reverse it
- Model the degradation and apply the inverse process in order to recover the original image.

Similar to image enhancement, but more objective



5.1 A Model of Image Degradation/Restoration Process

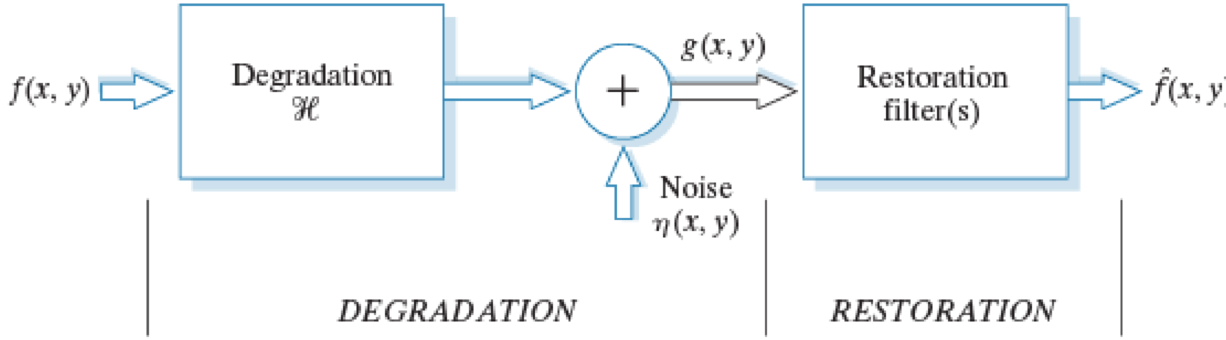
FIGURE 5.1
A model of the image degradation/restoration process.



- Degradation
  - Degradation function H
  - Additive noise $\eta(x, y)$

A Model of Image Degradation/Restoration Process

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



If H is a linear, position-invariant process, then the degraded image is given in the spatial domain by

$$g(x,y) = h(x,y) \star f(x,y) + \eta(x,y)$$

In Frequency Domain:

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

Noise Sources

- The principal sources of noise in digital images arise during **image acquisition and/or transmission**
  - Imaging sensors can be affected by ambient conditions
  - Interference can be added to an image during transmission
- ✓ Image acquisition
  - ➢ Light levels, sensor temperature, etc.
- ✓ Transmission
  - ➢ Lightning or other atmospheric disturbance

5.2 Noise Model

Noisy image is modelled as follows:

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

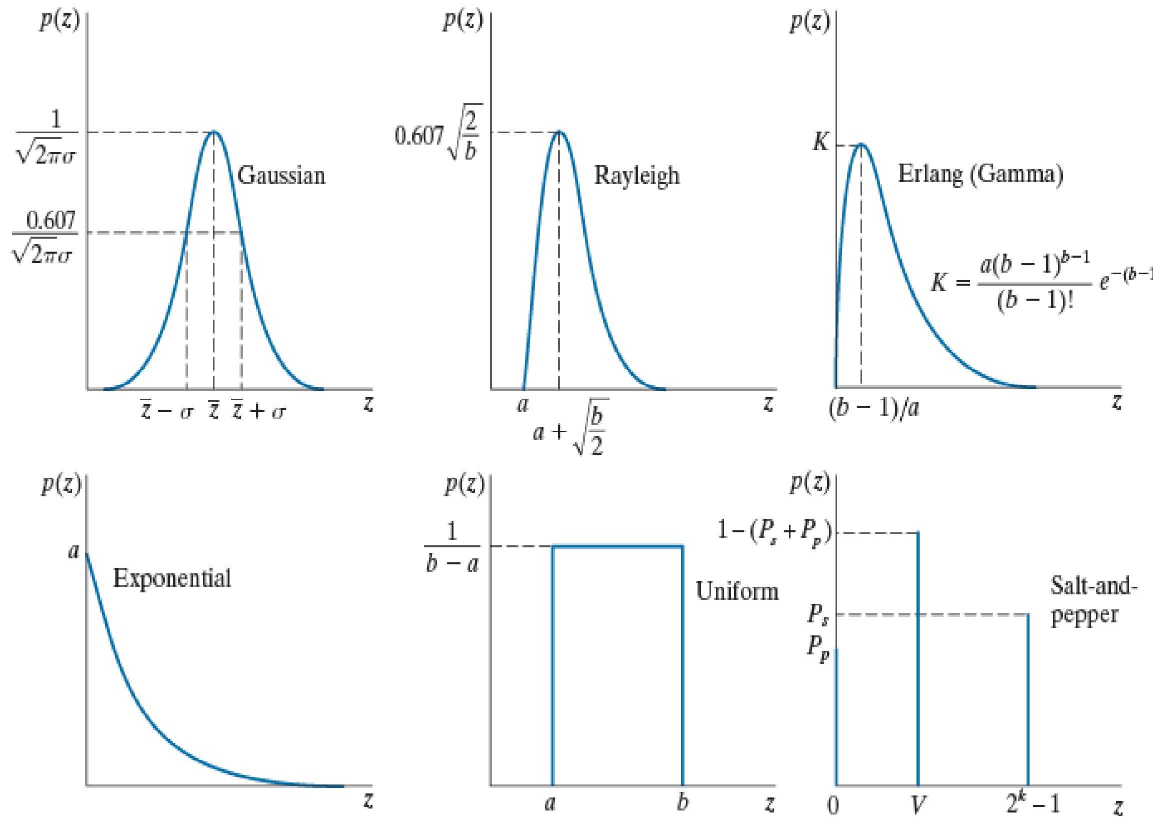
- $f(x, y)$: The original image pixel
- $\eta(x, y)$: Noise term
- $g(x, y)$: Resulting noisy pixel

If the noise model can be estimated then it is possible to restore the image

Noise Models (cont…)

Common models for the image noise term $\eta(x, y)$:

- Gaussian
  - Most common
- Rayleigh
- Erlang (Gamma)
- Exponential
- Uniform
- Impulse
  - *Salt and pepper* noise



a b c
d e f

FIGURE 5.2 Some important probability density functions.

5.2 Noise Models

- **White noise**
  - Power spectrum of white noise is constant (flat)

Assumptions

- With the exception of spatially periodic noise, we assume
  - Noise is independent of spatial coordinates
  - Noise is uncorrelated with respect to the image itself

Noise Models – Where it Occurs

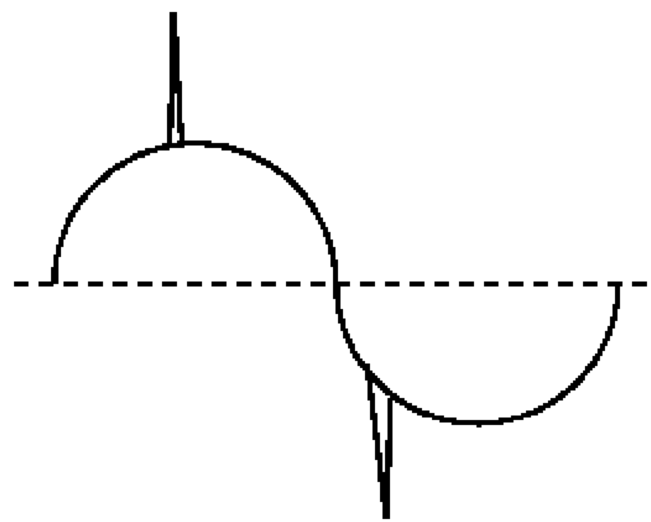
- **Gaussian noise**
  - Electronic circuit noise
  - Sensor noise due to poor illumination
  - High temperature
  - Central Limit Theorem (Sum of independent noise sources)

- **Rayleigh noise**

  Range imaging (Attenuation)

Noise Models

- **Erlang (gamma) noise:** Laser imaging
- **Exponential noise:** Laser imaging
- **Uniform noise:** Least descriptive; Basis for numerous random number generators
- **Impulse noise:** Quick transients such as faulty switching



Impulse Noise

Gaussian Noise

The PDF of Gaussian random variable, z, is given by

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

where, z represents intensity

$\bar{z}$ is the mean (average) value of z

σ is the standard deviation

Gaussian Noise

The PDF of Gaussian random variable, z, is given by

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

- 70% of its values will be in the range

$$\left[(\overline{z}-\sigma),(\overline{z}+\sigma)\right]$$

- 95% of its values will be in the range

$$\left[(\overline{z}-2\sigma),(\overline{z}+2\sigma)\right]$$

Rayleigh Noise

The PDF of Rayleigh noise is given by

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

The mean and variance of this density are given by

$$\bar{z} = a + \sqrt{\pi b / 4}$$

$$\sigma^2 = \frac{b(4-\pi)}{4}$$

Erlang (Gamma) Noise

The PDF of Erlang noise is given by

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

The mean and variance of this density are given by

$$\bar{z} = b / a$$

$$\sigma^2 = b / a^2$$

Exponential Noise

The PDF of exponential noise is given by

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

The mean and variance of this density are given by

$$\bar{z} = 1/a$$

$$\sigma^2 = 1/a^2$$

Uniform Noise

The PDF of uniform noise is given by

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

The mean and variance of this density are given by

$$\bar{z} = (a+b)/2$$

$$\sigma^2 = (b-a)^2/12$$

5.2.2 Impulse Noise Model: Bipolar & Salt-and-Pepper noise

- Bipolar impulse noise follows the following distribution

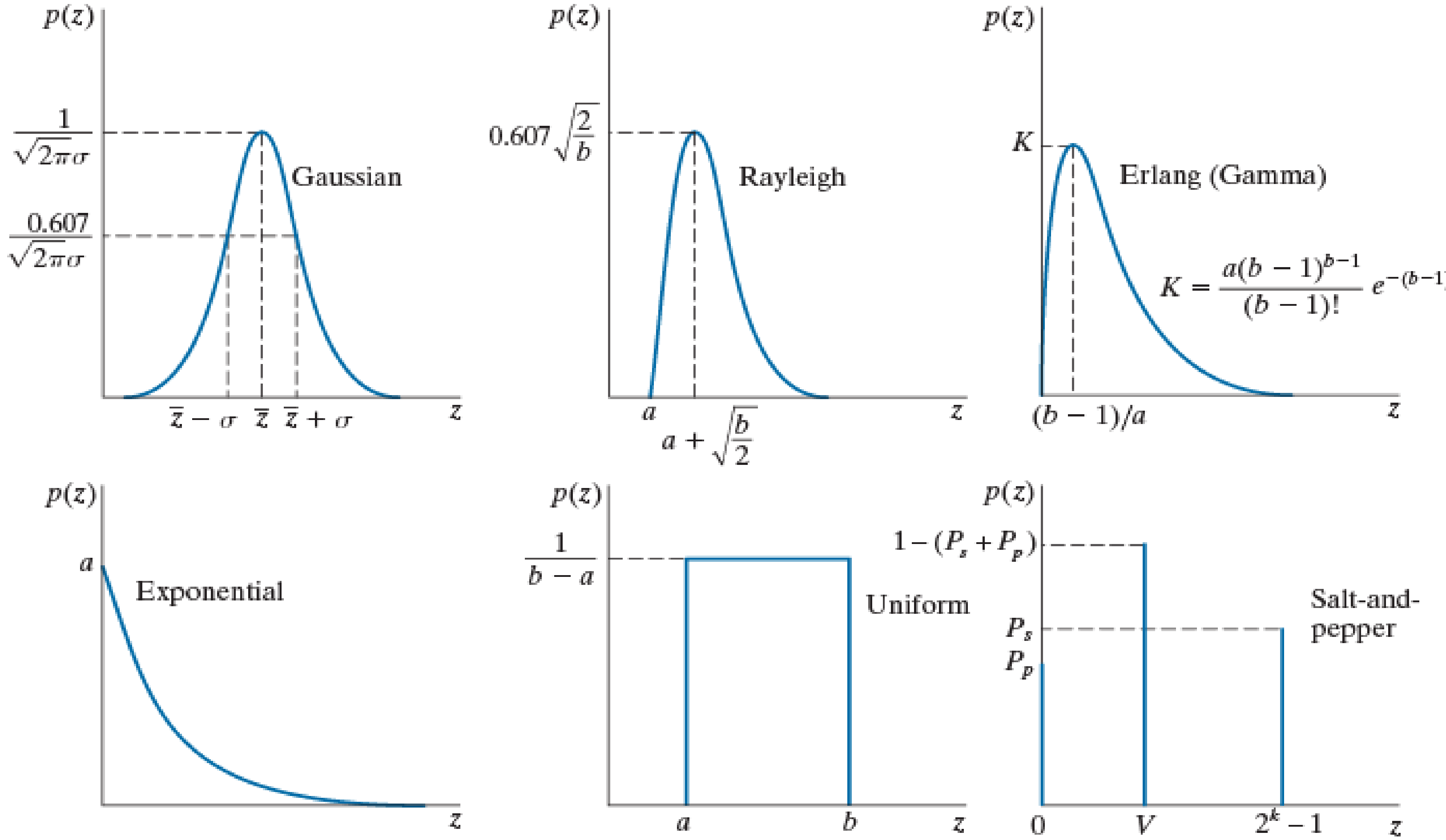
$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$

a and b are intensity levels

- If P_a or P_b is zero, we have unipolar impulse noise. If both are nonzero, and almost equal, this is also called salt-and-pepper noise

- If $b > a$, gray-level b will appear as a light dot, while level a will appear like a dark dot

Important Probability Density Functions (PDF)



a b c
d e f

FIGURE 5.2 Some important probability density functions.

Wright State University

Examples of Noise: Original Image

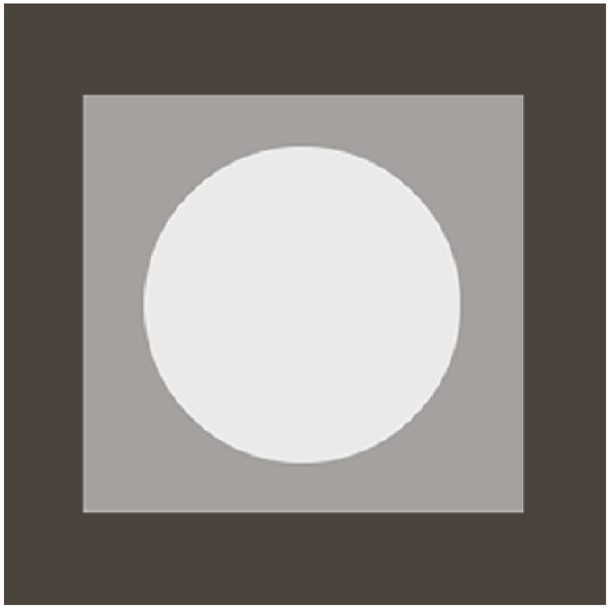
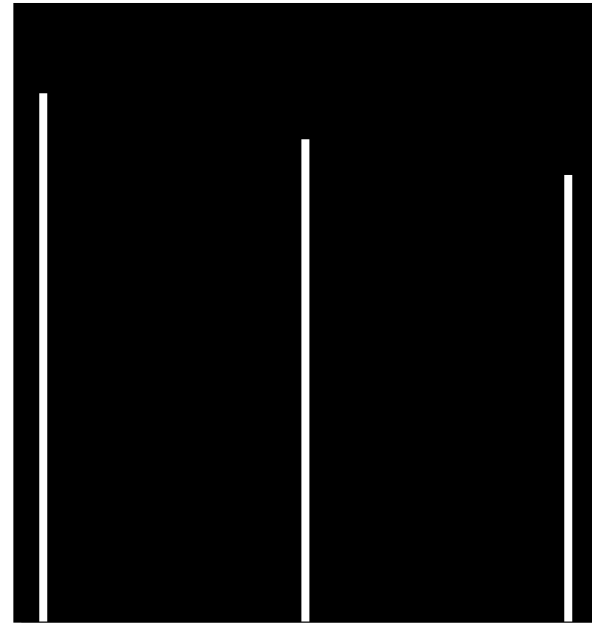


FIGURE 5.3 Test pattern used to illustrate the characteristics of the PDFs from Fig. 5.2.

Histogram



- The test pattern is ideal for demonstrating the effect of adding noise
- The following slides will show the result of adding noise based on various models to this image

Examples of Noise: Noisy Images

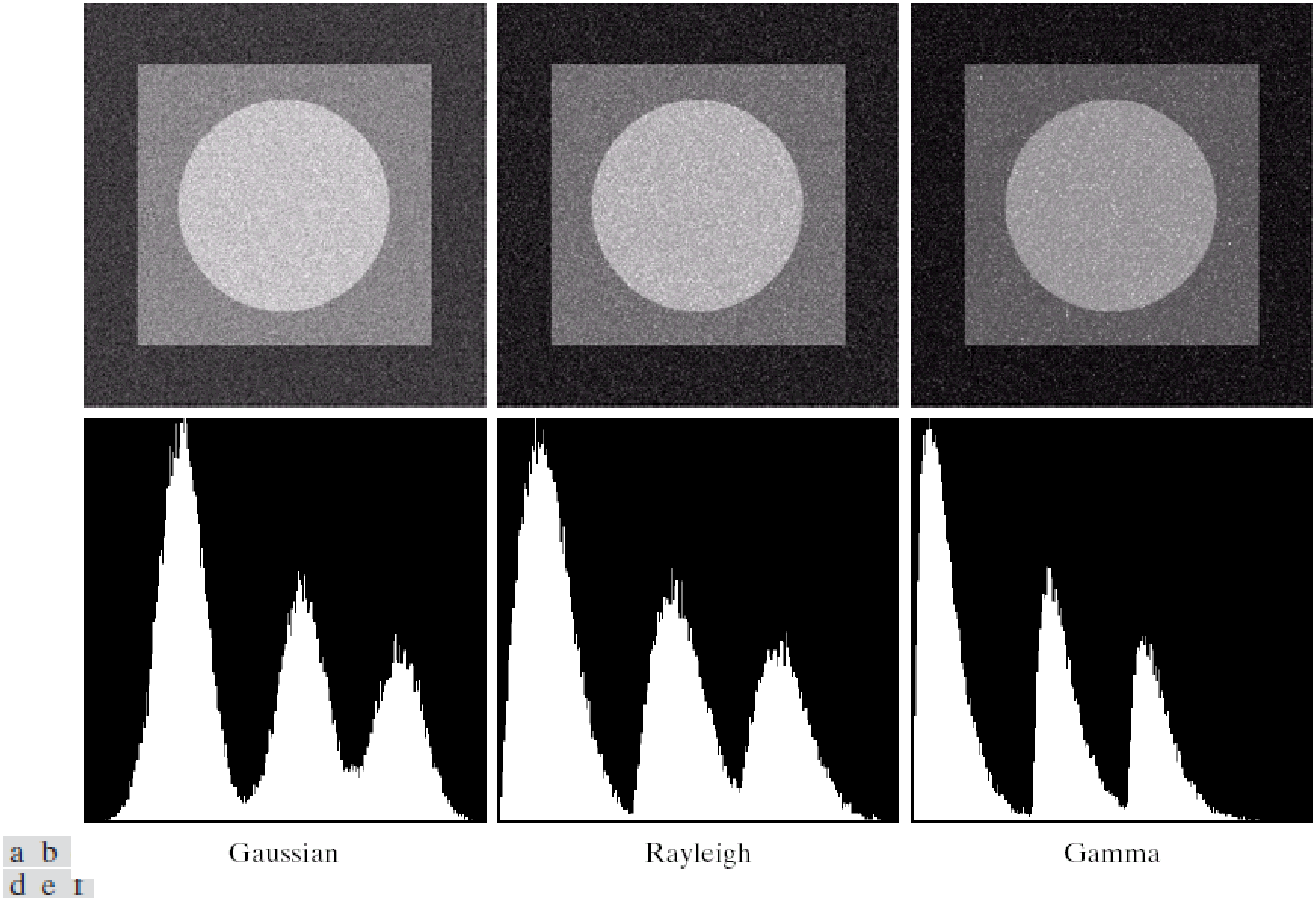


FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and Erlanga noise to the image in Fig. 5.3.

Examples of Noise: Noisy Images

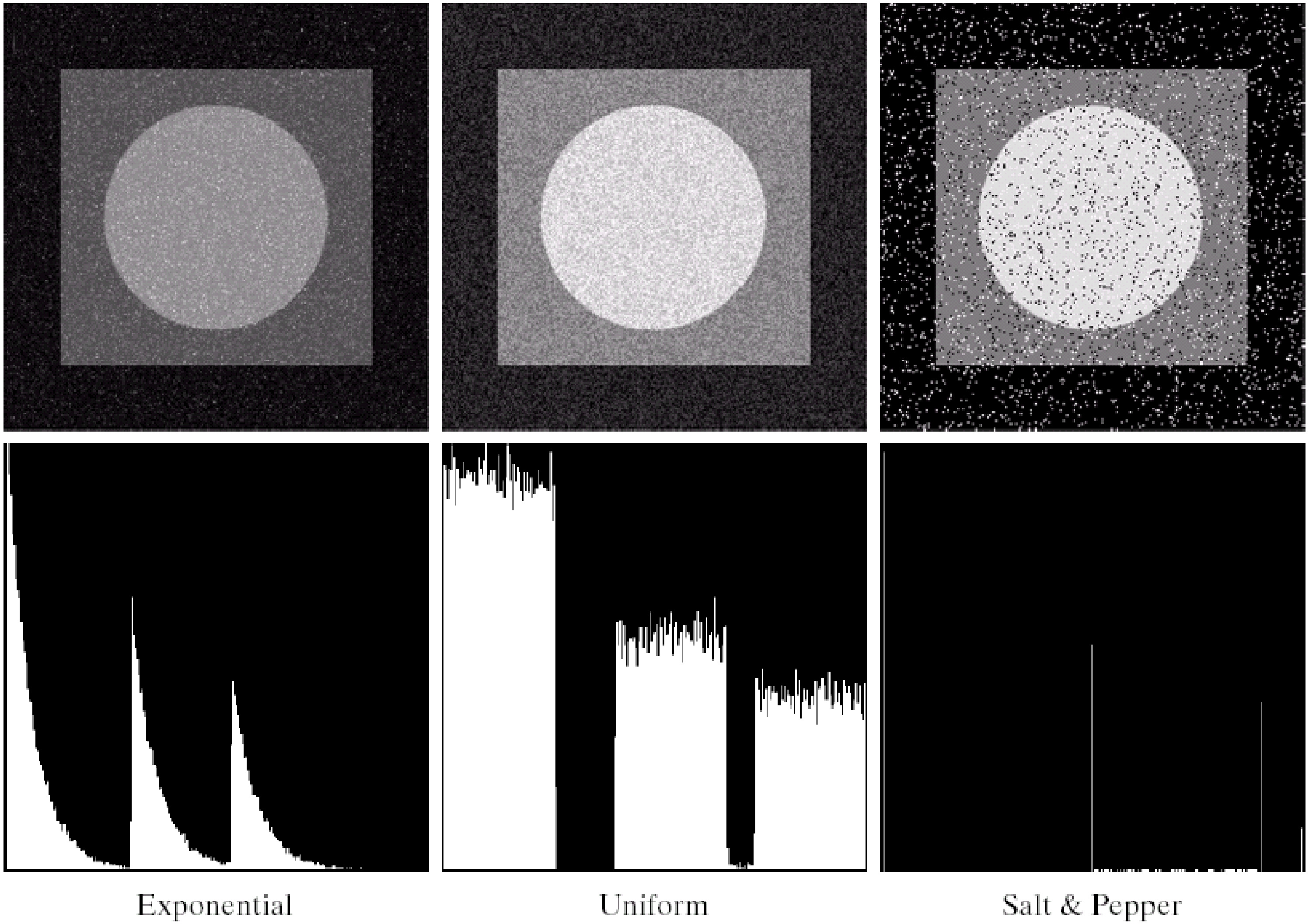


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt and pepper noise to the image in Fig. 5.3.

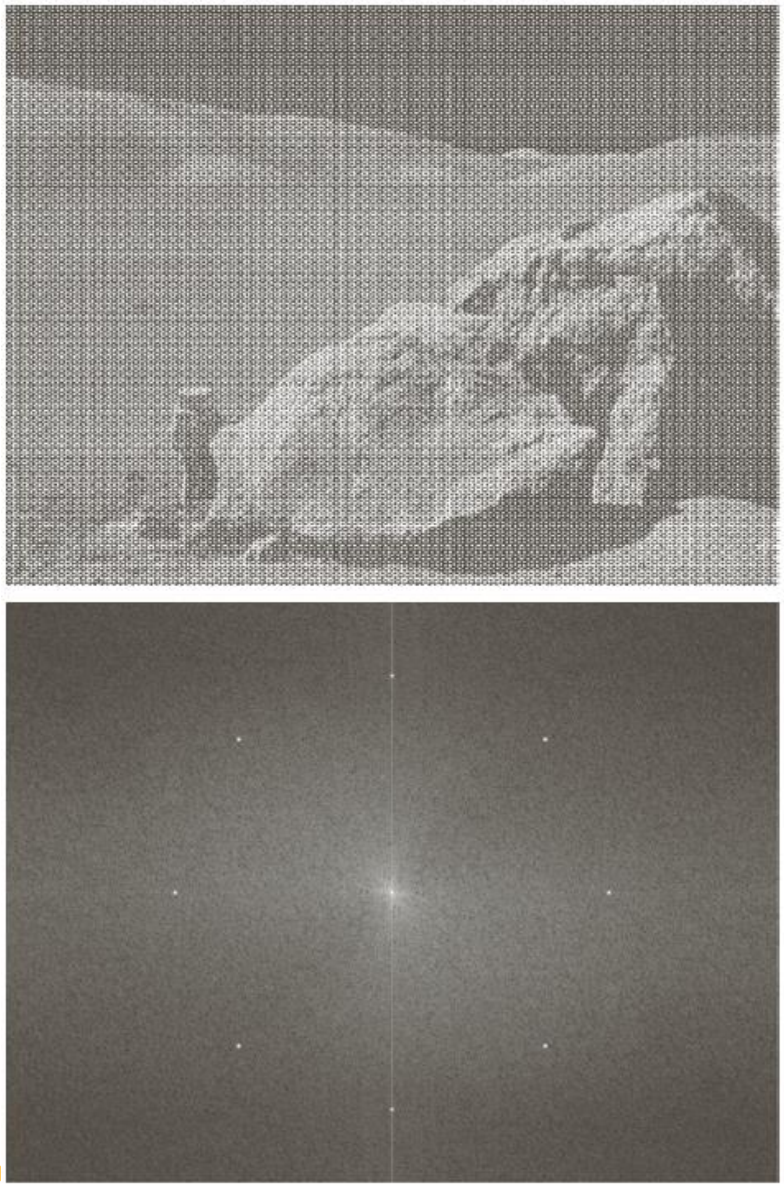
5.2.3 Periodic Noise

- Periodic noise arises typically during image acquisition due to electrical interference
- It is a type of spatially dependent noise
- Gives rise to regular noise patterns in an image
- Frequency domain techniques in the Fourier domain are most effective at removing 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.)

An Example of Periodic Noise



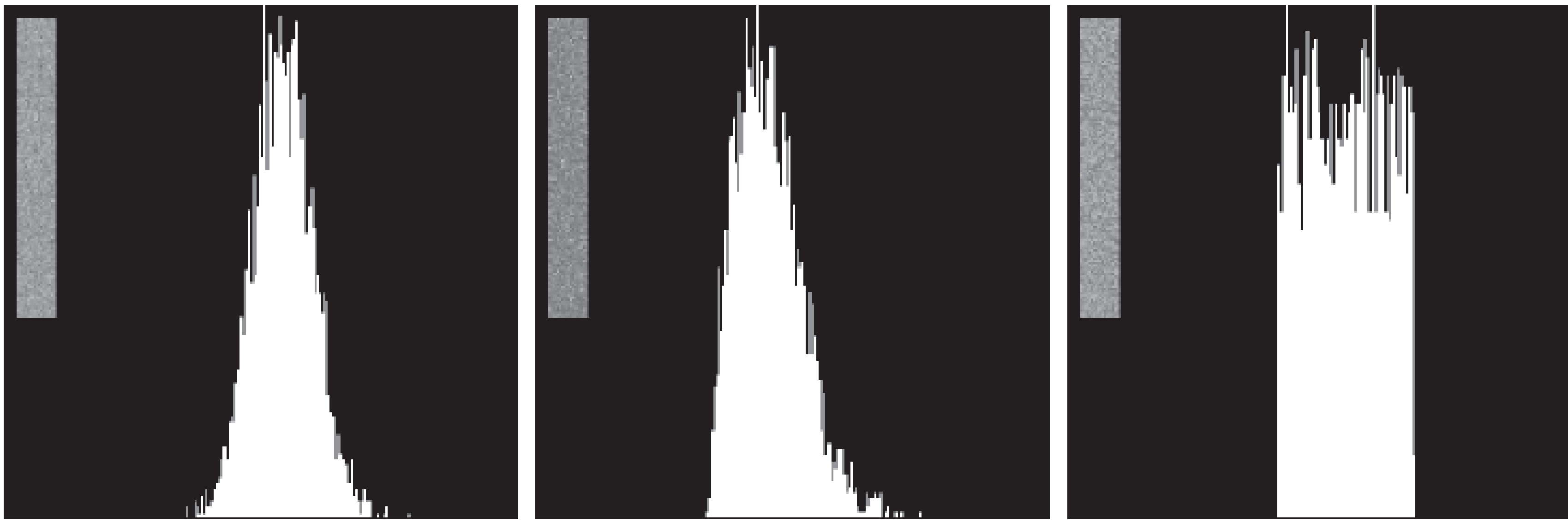
a
b

FIGURE 5.5 (a) Image corrupted by sinusoidal noise. (b) Spectrum (each pair of conjugate impulses corresponds to one sine wave). (Original image courtesy of NASA.)

3rd Edition

5.2.4 Estimation of Noise Parameters

The shape of the histogram identifies the closest PDF match



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.

Estimation of Noise Parameters

Consider a subimage denoted by S, and let $p_s(z_i)$, $i = 0,\ 1,\ \ldots,\ L-1$, denote the probability estimates of the intensities of the pixels in S. The mean and variance of the pixels in S:

$$\bar{z} = \sum_{i=0}^{L-1} z_i p_s(z_i)$$

and

$$\sigma^2 = \sum_{i=0}^{L-1} (z_i - \bar{z})^2 p_s(z_i)$$

5.3 Restoration in the Presence of Noise Only - Spatial Filtering

Only degradation is due to Additive Noise:

Spatial Domain: $g(x, y) = f(x, y) + \eta(x, y)$

Frequency Domain: $G(u,v) = F(u,v) + N(u,v)$

- Noise is unknown → Not possible to subtract out
- Need spatial or frequency-domain filtering

5.3.1 Spatial Filtering: Mean Filters

- We can use different spatial filters to remove different kinds of noise
- The *arithmetic mean* filter is a very simple one and is calculated as follows:

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

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

This is implemented as the simple smoothing filter

It blurs the image.

Spatial Filtering: Mean Filters

There are other kinds of mean filters exhibiting slightly different behavior:

- Geometric Mean
- Harmonic Mean
- Contra-harmonic Mean

WRIGHT STATE UNIVERSITY

Spatial Filtering: Mean Filters

Geometric mean filter

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

- Generally, a geometric mean filter achieves smoothing comparable to the arithmetic mean filter

- But it tends to lose less image detail in the process

Spatial Filtering: Mean Filters

Harmonic mean filter

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

- Works well for salt noise, but fails for pepper noise
- Also does well with other types of noise, such as, Gaussian noise

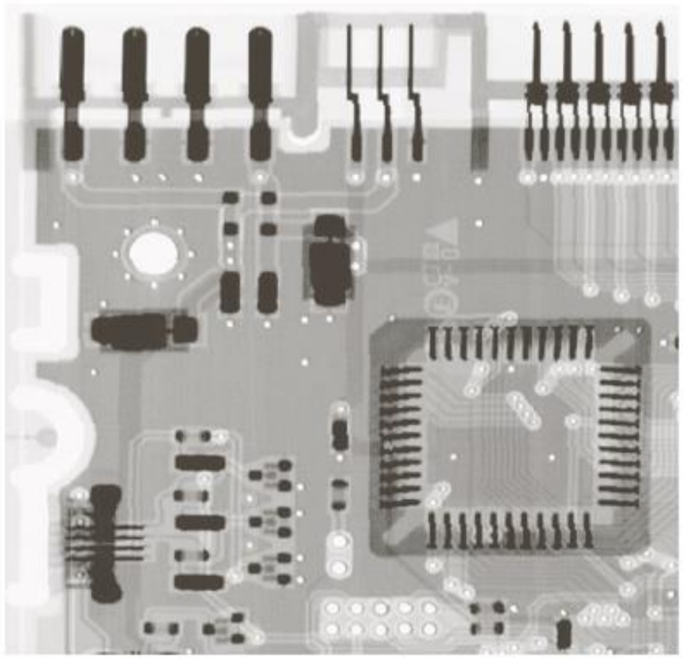
Spatial Filtering: Mean Filters

Contraharmonic mean filter

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

- Q is the order of the filter
- Well suited for reducing effects of salt-and-pepper noise
  - $Q>0$ eliminates pepper noise
  - $Q<0$ eliminates salt noise
  - But not both at the same time

Spatial Filtering: Example



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.)

Noise Removal Examples

Original image

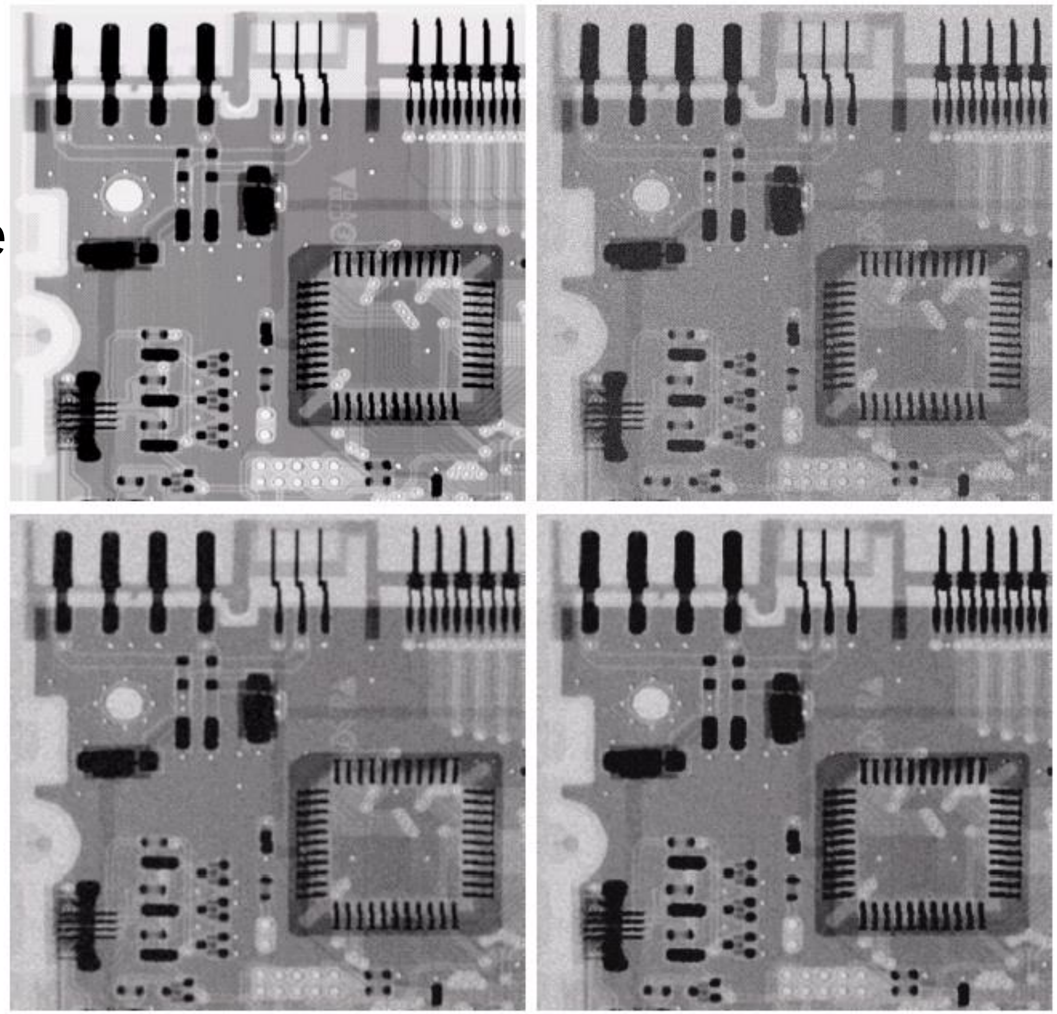
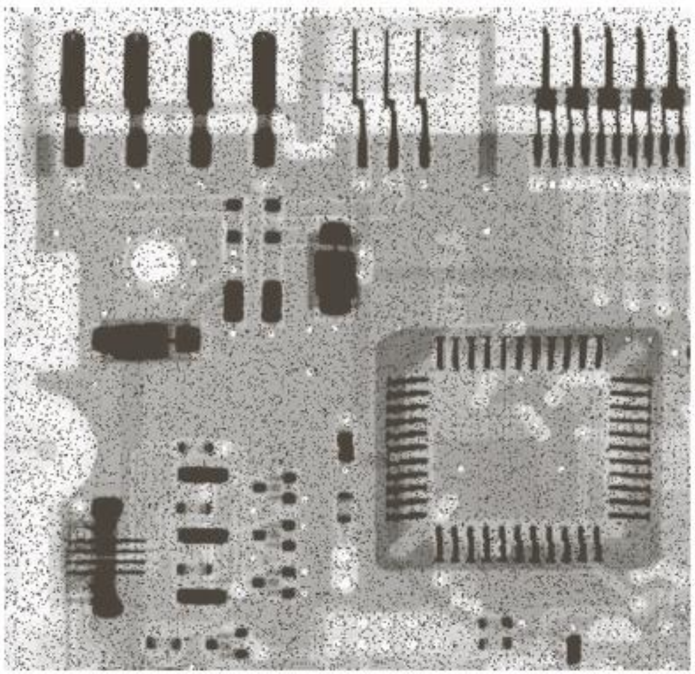


Image corrupted by Gaussian noise

3x3 Arithmetic Mean Filter

3x3 Geometric Mean Filter (less blurring than AMF, the image is sharper)

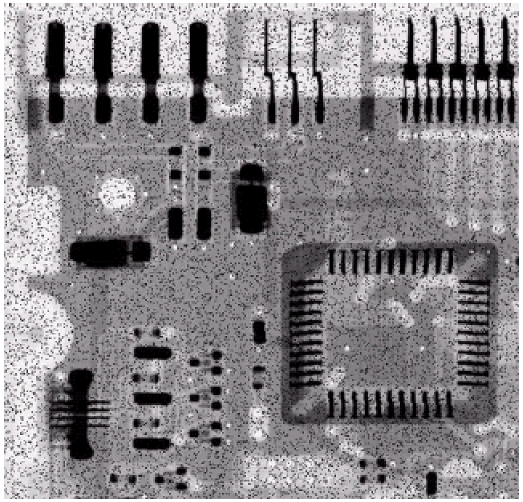


a b
c d

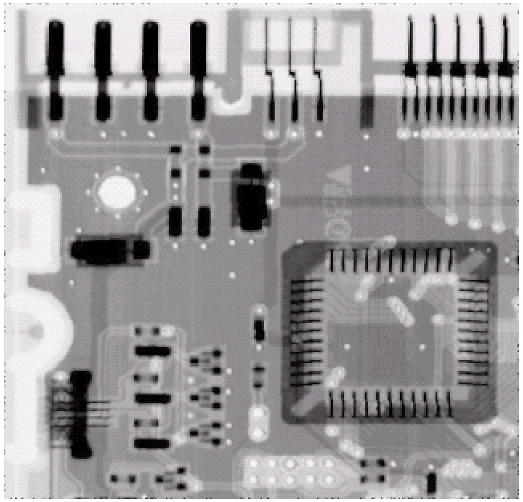
FIGURE 5.8
(a) Image corrupted by pepper noise with a probability of 0.1. (b) Image corrupted by salt noise with the same probability. (c) Result of filtering (a) with a 3×3 contra-harmonic filter $Q = 1.5$. (d) Result of filtering (b) with $Q = -1.5$.

Noise Removal Examples (cont…)

Image corrupted by pepper noise at 0.1

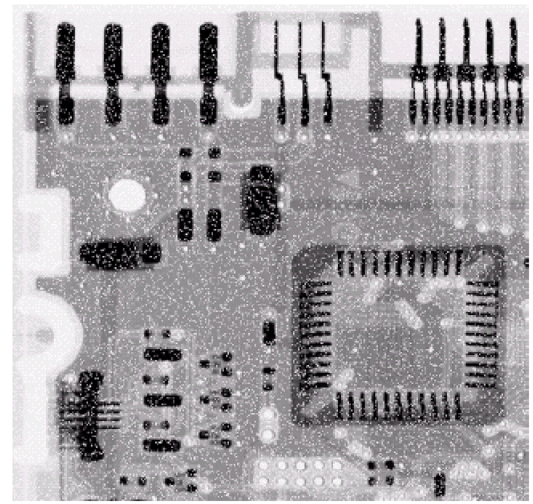


Filtering with a 3x3 Contraharmonic Filter with Q=1.5

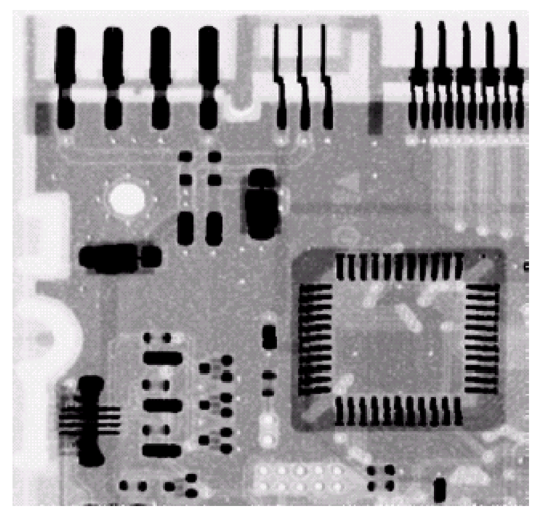


Noise Removal Examples (cont…)

Image corrupted by salt noise at 0.1



Filtering with a 3x3 Contraharmonic Filter with Q=-1.5

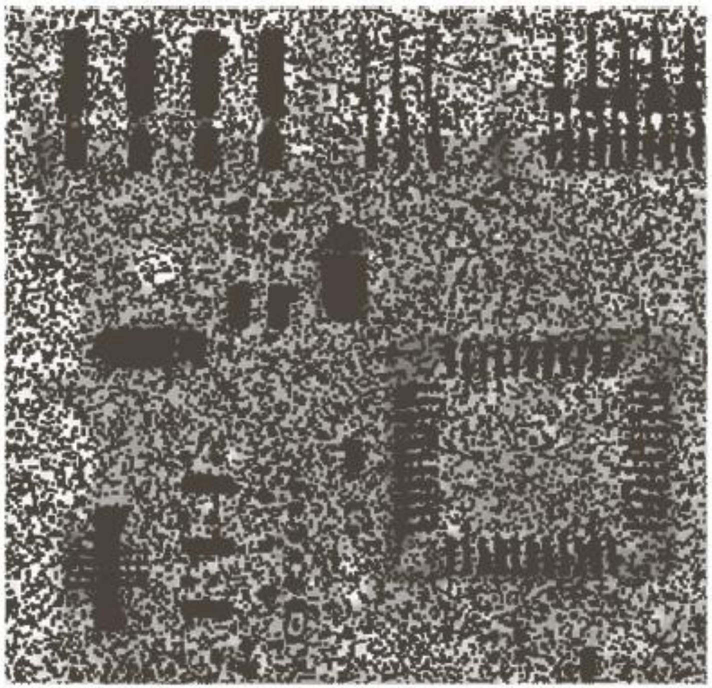


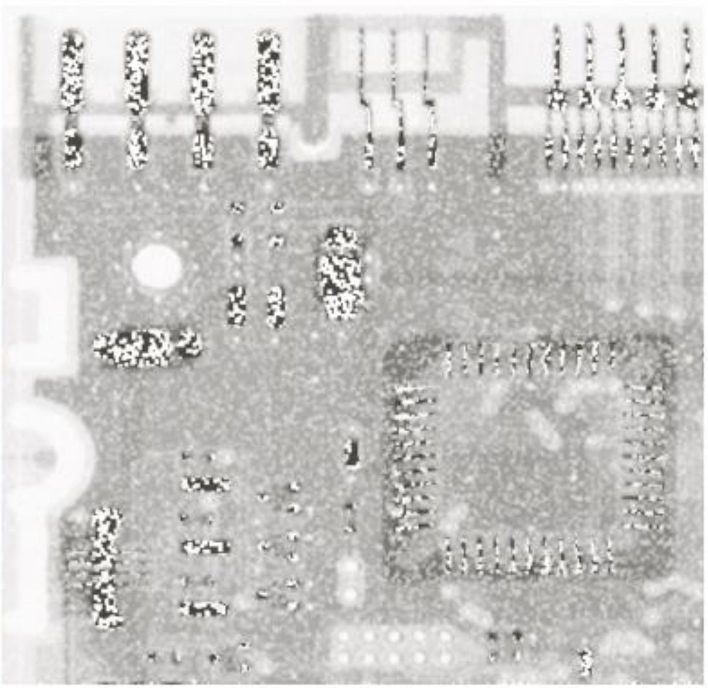
Contraharmonic Filter

Choosing the wrong value for Q when using the contraharmonic filter can have drastic results

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 $Q = 1.5$.





Pepper noise filtered by a 3x3 CF with Q=-1.5

Salt noise filtered by a 3x3 CF with Q=1.5

5.3.2 Order Statistics Filters

Spatial filters based on ordering the pixel values that make up the neighbourhood defined by the filter support.

Useful spatial filters include

- Median filter
- Max and Min filter
- Midpoint filter
- Alpha trimmed mean filter

Median Filter

Median Filter:

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

Excellent at noise removal, without the smoothing effects that can occur with other smoothing filters.

Particularly good when salt and pepper noise is present at the same time.

Max and Min Filters

Max Filter:

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

Min Filter:

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

Max filter is good for pepper noise and Min filter is good for salt noise.

Midpoint Filter

Midpoint Filter:

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

Good for random Gaussian and uniform noise.

Alpha-Trimmed Mean Filter

Alpha-Trimmed Mean Filter:

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

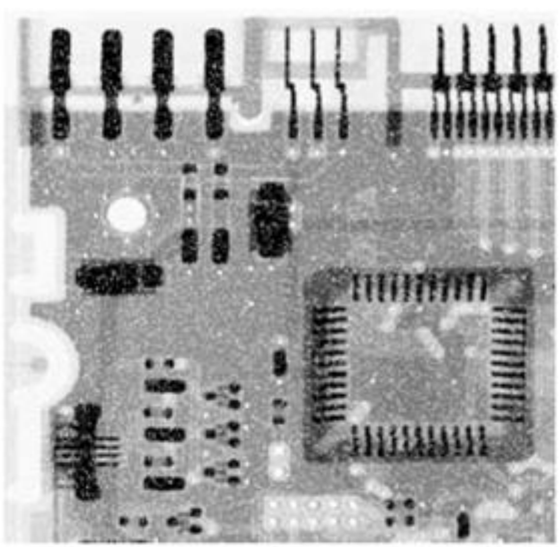
- Deletes $d/2$ lowest and $d/2$ highest intensity gray levels

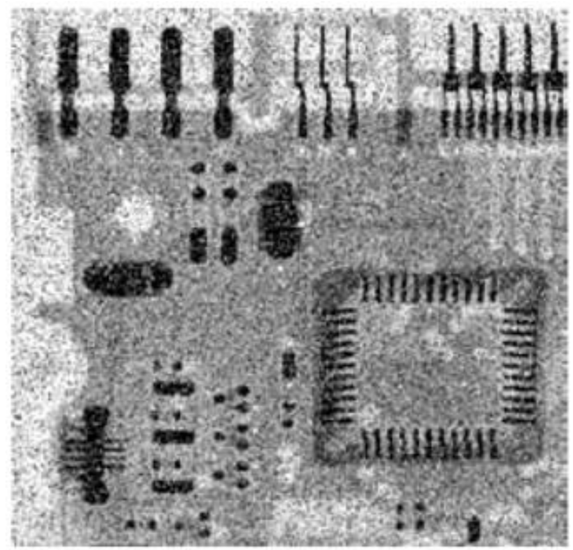
- $g_r(s, t)$ represents the remaining $mn - d$ pixels.

Impulse Noise Example - Salt-and-Pepper

Salt noise: with a certain probability P_a a pixel becomes white.

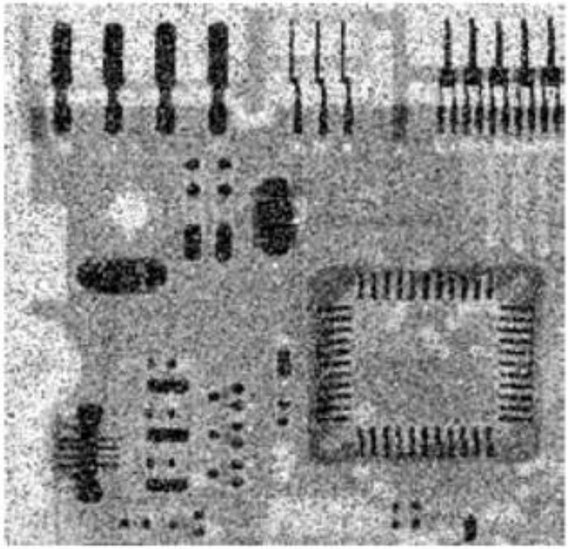
Pepper noise: with a certain probability P_b a pixel becomes black.





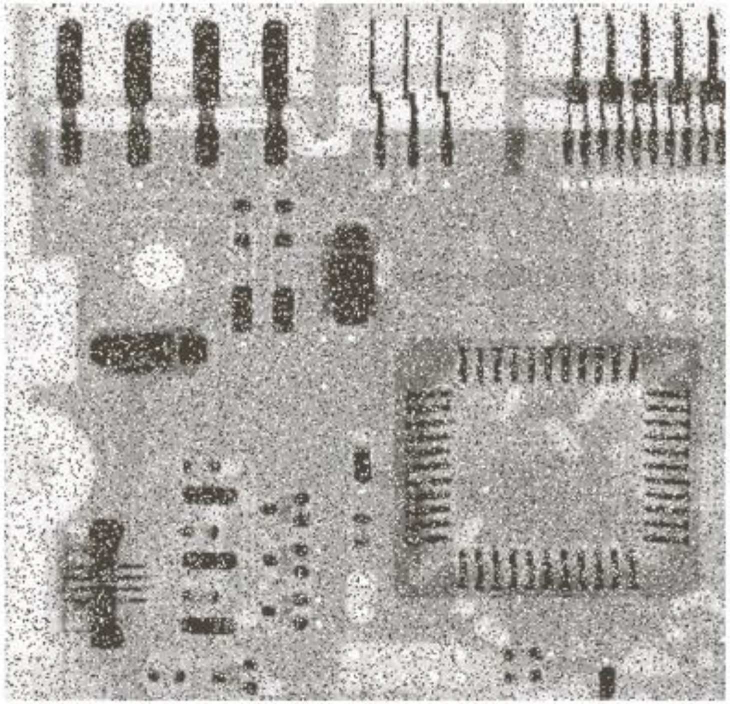
Example - Impulse Noise

Salt and pepper noise: with a certain probability p, a pixel becomes either white or black.



a b
c d

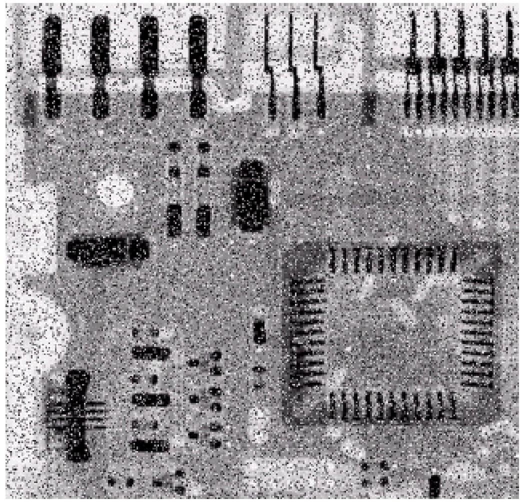
FIGURE 5.10 (a) Image corrupted by salt-and- pepper noise with probabilities $P_s = P_p = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.

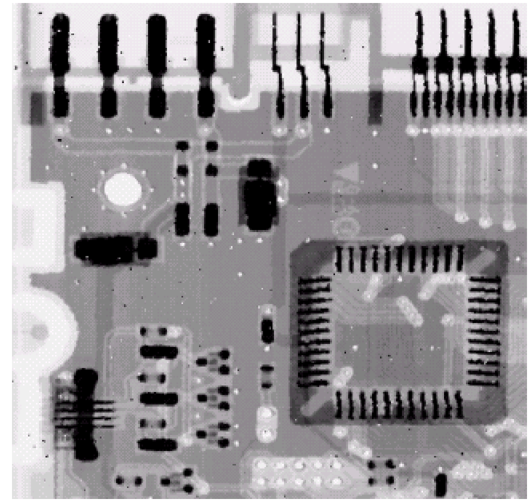


E

Fig. 5.10 Noise Removal Examples

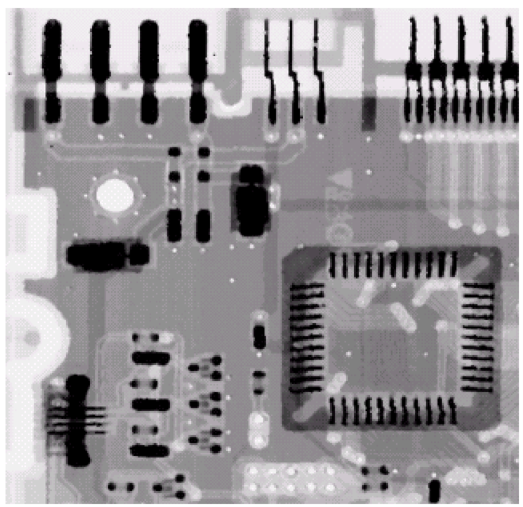
Image corrupted by Salt And Pepper noise at $P_a = P_b = 0.1$

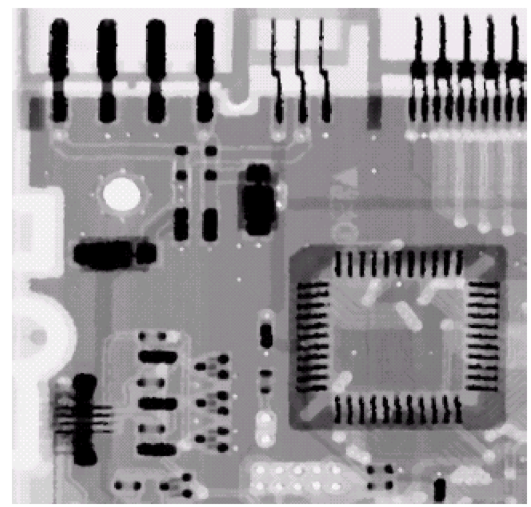




Result of 1 pass with a 3x3 Median Filter

Result of 2 passes with a 3x3 Median Filter





Result of 3 passes with a 3*3 Median Filter

Repeated passes remove the noise better but also blur the image

Results with Max and Min Filters

Originally had pepper noise

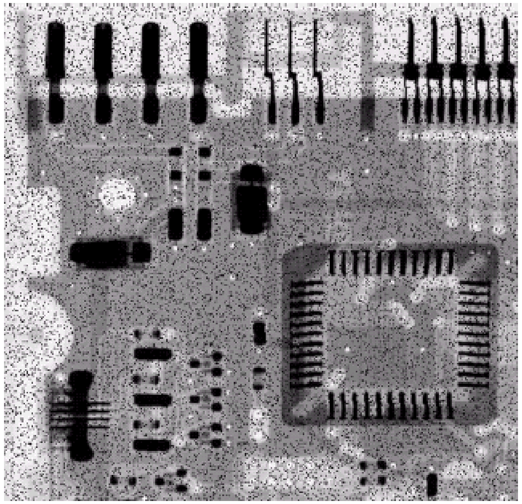
Originally had salt noise

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.

Noise Removal Examples (cont…)

Image corrupted by Pepper noise



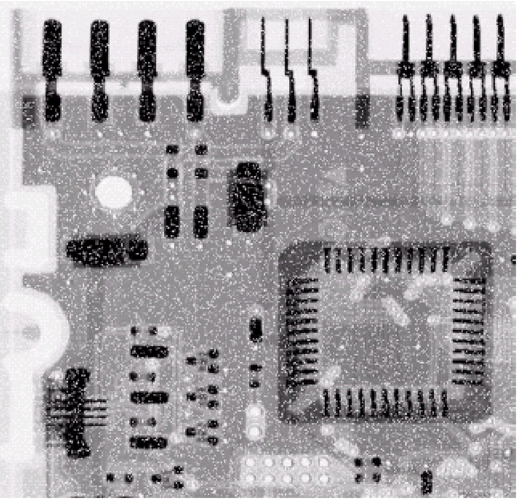
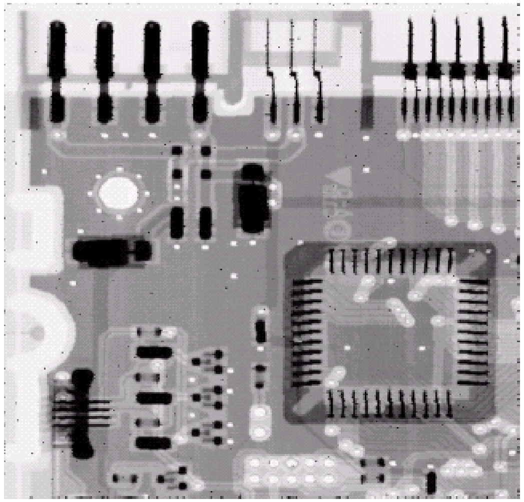
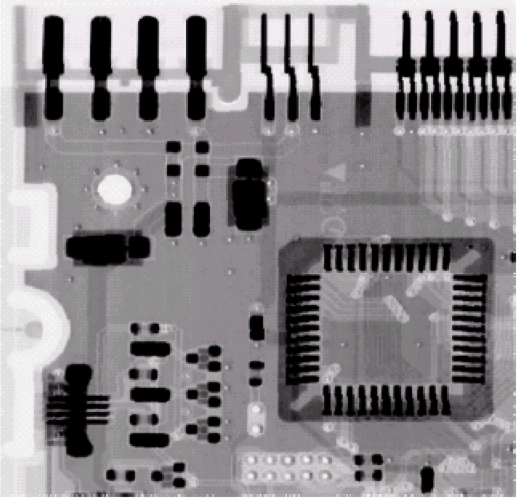


Image corrupted by Salt noise

Filtering above with a 3x3 Max Filter





Filtering above with a 3x3 Min Filter

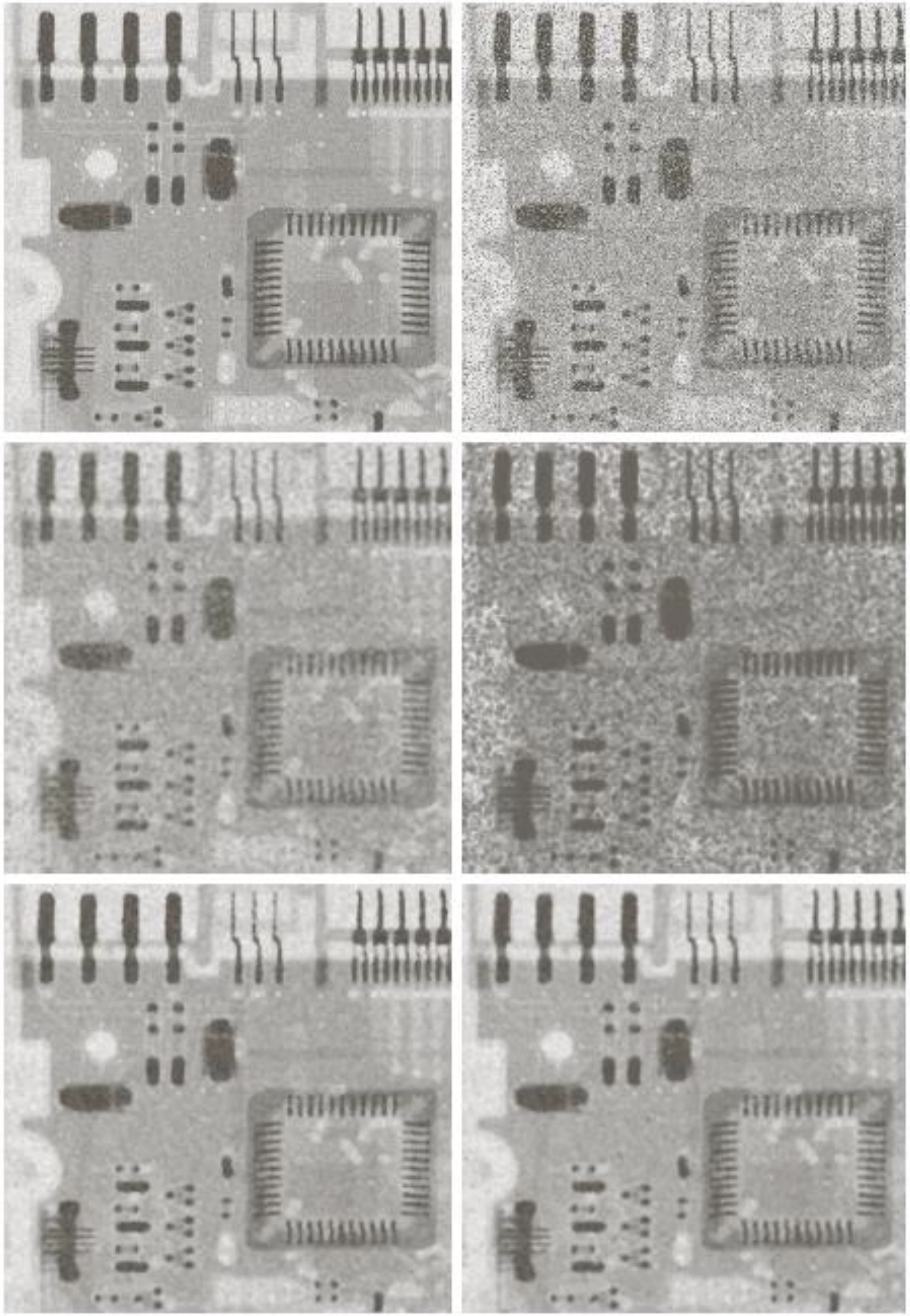
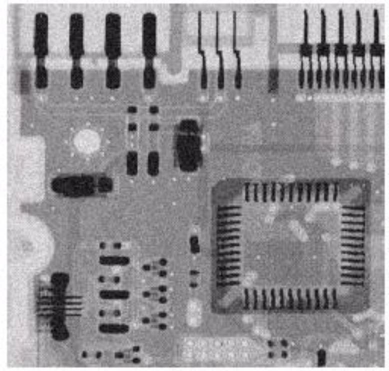


FIGURE 5.12 (a) Image corrupted by additive uniform noise. (b) Image additionally corrupted by additive salt-and-pepper noise. (c)-(f) Image (b) filtered with a 5×5: (c) arithmetic mean filter; (d) geometric mean filter; (e) median filter; (f) alpha-trimmed mean filter, with $d = 6$.

Noise Removal Examples (cont…)

Image corrupted by uniform noise



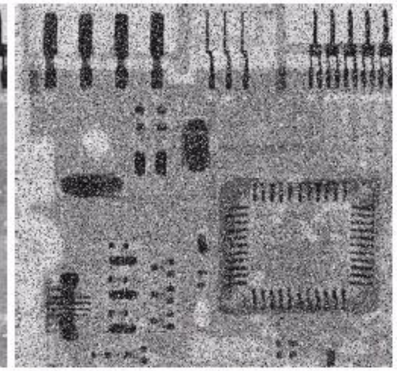
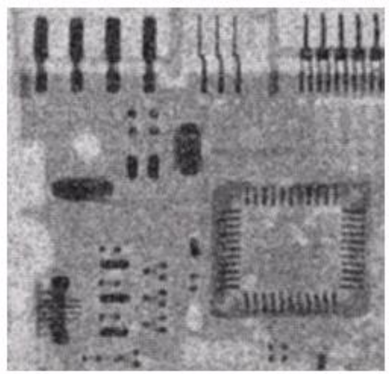
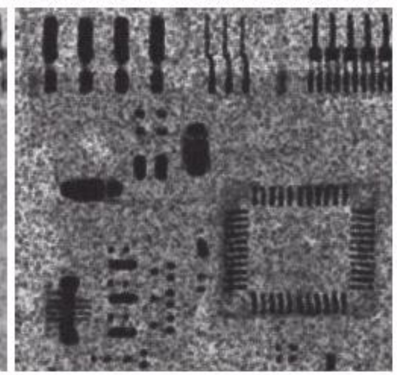


Image further corrupted by Salt and Pepper noise

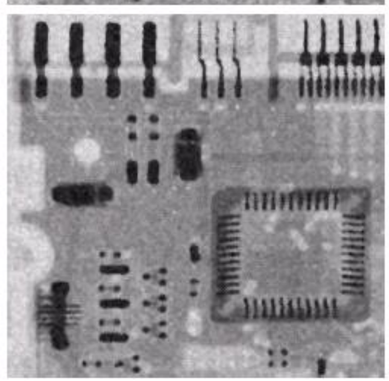
Filtering by a 5x5 Arithmetic Mean Filter

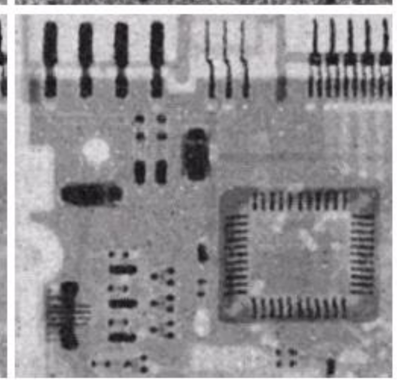




Filtering by a 5x5 Geometric Mean Filter

Filtering by a 5x5 Median Filter





Filtering by a 5x5 Alpha-Trimmed Mean Filter (d=5)

5.3.3 Spatial Filtering: Adaptive Filters

- The filters discussed so far are applied to an entire image without any regard for how image characteristics vary from one point to another
- The behavior of **adaptive filters** changes depending on the characteristics of the image inside the filter region defined by the $m \times n$ rectangular window
- In general, performance is superior to that of the filters discussed so far

Adaptive Filters: Adaptive, Local Noise Reduction Filters

S_{xy}: local region

The response of the filter at the center point (x,y) of S_{xy} is based on four quantities:

(a) $g(x, y)$, the value of the noisy image at (x, y);

(b) σ_η^2, the variance of the noise corrupting $f(x, y)$ to form $g(x, y)$;

(c) $\bar{z}_{S_{xy}}$, the local mean of the pixels in S_{xy};

(d) $\sigma_{S_{xy}}^2$, the local variance of the pixels in S_{xy}.

Adaptive Filters: Adaptive, Local Noise Reduction Filters

The behavior of the filter:

(a) if σ_η^2 is zero, the filter should return simply the value of $g(x, y)$.

(b) if the local variance is high relative to σ_η^2, the filter should return a value close to $g(x, y)$;

(c) if the two variances are equal, the filter returns the arithmetic mean value of the pixels in S_{xy}.

Adaptive Filters: Adaptive, Local Noise Reduction Filters

An adaptive expression for obtaining $f(x, y)$ based on these assumptions:

$$f(x,y) = g(x,y) - \frac{\sigma_{\eta}^{2}}{\sigma_{S_{xy}}^{2}}\left[g(x,y) - \bar{z}_{S_{xy}}\right]$$

a	b
c	d

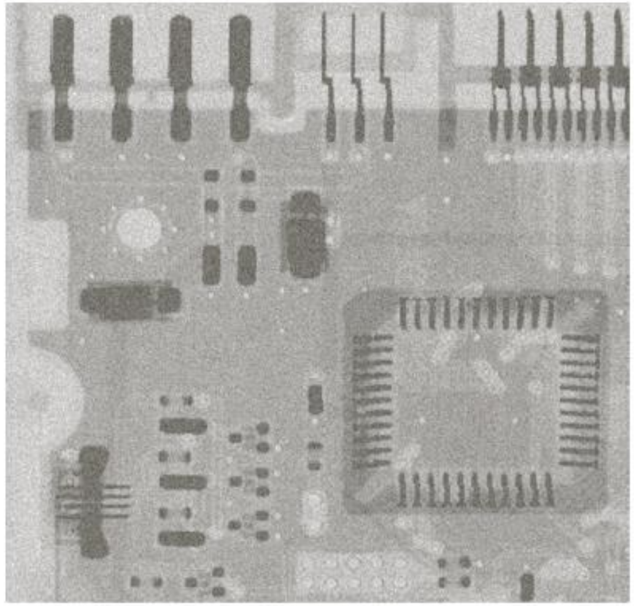


FIGURE 5.13
(a) Image corrupted by additive Gaussian noise of zero mean and a variance of 1000. (b) Result of arithmetic mean filtering. (c) Result of geometric mean filtering. (d) Result of adaptive noise-reduction filtering. All filters used were of size 7×7.

Adaptive Filters: Adaptive Median Filters

The notation:

$$z_{\min} = \text{minimum intensity value in } S_{xy}$$

$$z_{\max} = \text{maximum intensity value in } S_{xy}$$

$$z_{\text{med}} = \text{median intensity value in } S_{xy}$$

$$z_{xy} = \text{ intensity value at coordinates } (x, y)$$

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

Adaptive Filters: Adaptive Median Filters

The adaptive median-filtering works in two stages:

Stage A:

$$A1 = z_{\text{med}} - z_{\text{min}}; \qquad A2 = z_{\text{med}} - z_{\text{max}}$$

if A1>0 and A2<0, go to stage B

Else increase the window size

if window size $\le S_{\text{max}}$, repeat stage A; Else output z_{med}

Stage B:

$$B1 = z_{xy} - z_{\text{min}}; \qquad B2 = z_{xy} - z_{\text{max}}$$

if B1>0 and B2<0, output z_{xy}; Else output z_{med}

WRIGHT STATE UNIVERSITY

Adaptive Filters: Adaptive Median Filters

The adaptive median-filtering works in two stages:

Stage A:

$A1 = z_{\text{med}} - z_{\text{min}}$; $A2 = z_{\text{med}} - z$

if A1>0 and A2<0, go to stage B

Else increase the window size

if window size $\leq S_{\max}$, repeat stage A; Else output z_{med}

If true, z_med is in between max and min, i.e., not an impulse

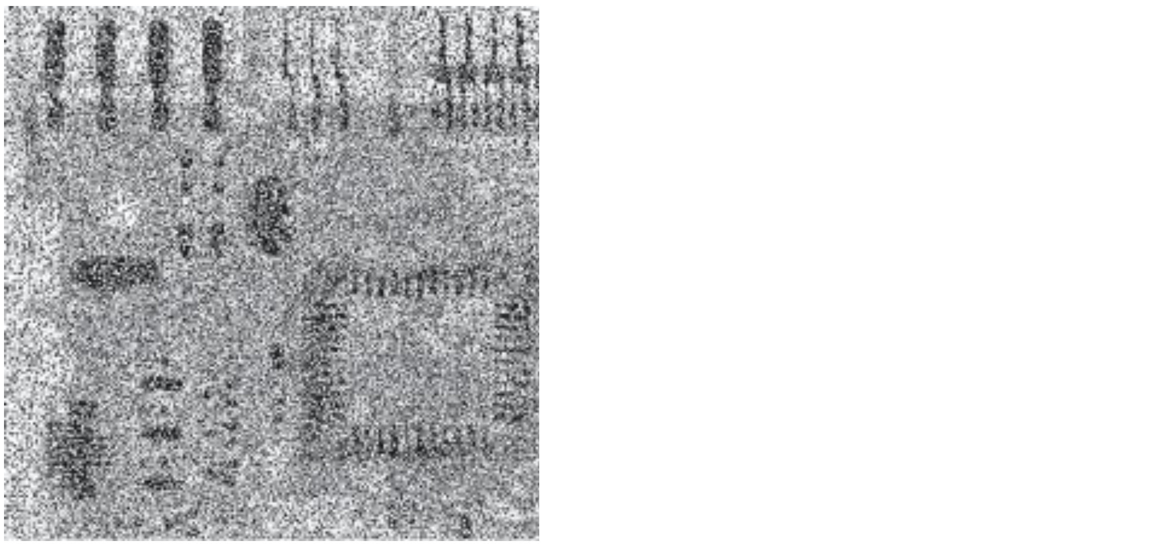
Stage B:

$B1 = z_{xy} - z_{\text{min}}$; $B2 = z_{xy} - z$

if B1>0 and B2<0, output z_{xy}; Else output z_{med}

The processed point is an impulse or not

Example: Adaptive Median Filters

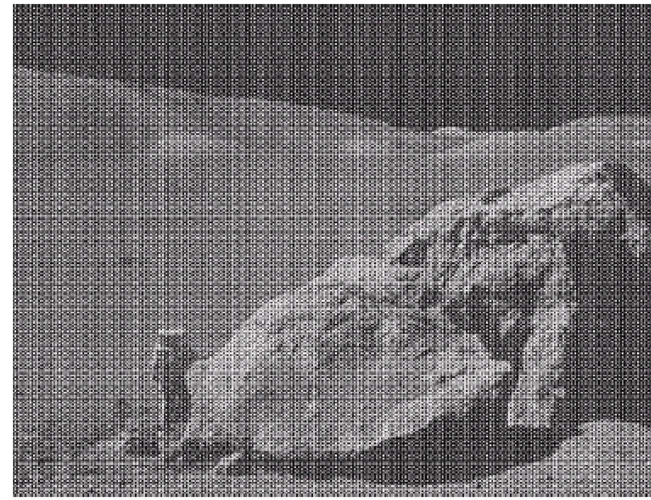


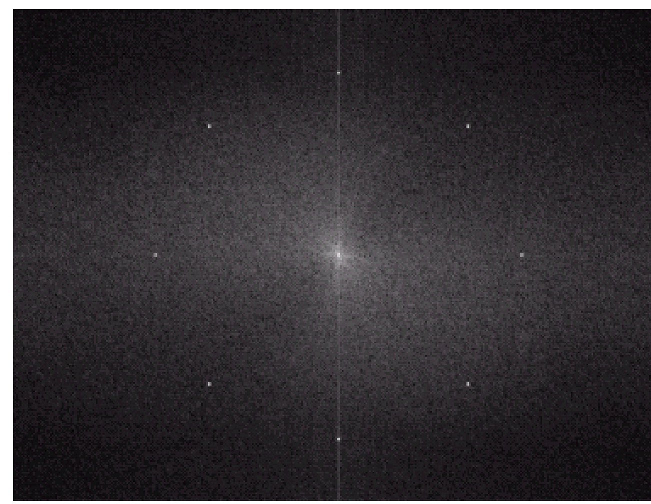
a b c

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 $S_{max} = 7$.

5.4 Periodic Noise Reduction by Frequency Domain Filtering

- Typically arises due to electrical or electromagnetic interference.
- Gives rise to regular noise patterns in an image.
- Frequency domain techniques in the Fourier domain are most effective at removing periodic noise.





Periodic Noise Reduction by Frequency Domain Filtering

The basic idea

- Periodic noise appears as concentrated bursts of energy in the Fourier transform, at locations corresponding to the frequencies of the periodic interference

Approach

- A selective filter is used to isolate the noise

Band Reject Filters

- Removing periodic noise form an image involves removing a particular range of frequencies from that image.
- *Band reject* filters can be used for this purpose
- An ideal band reject filter is given as follows:

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

Band Reject Filters (cont…)

The ideal band reject filter is shown below, along with Butterworth and Gaussian versions of the filter

Ideal Band Reject Filter

Butterworth Band Reject Filter (of order 1)

Gaussian Band Reject Filter

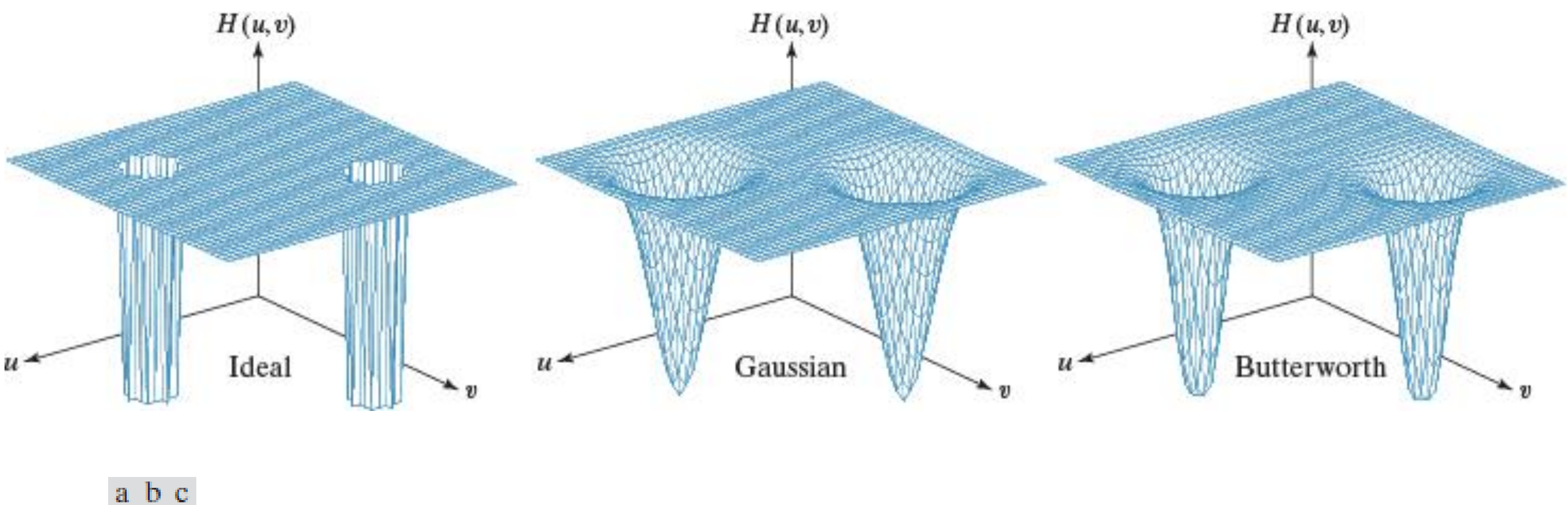
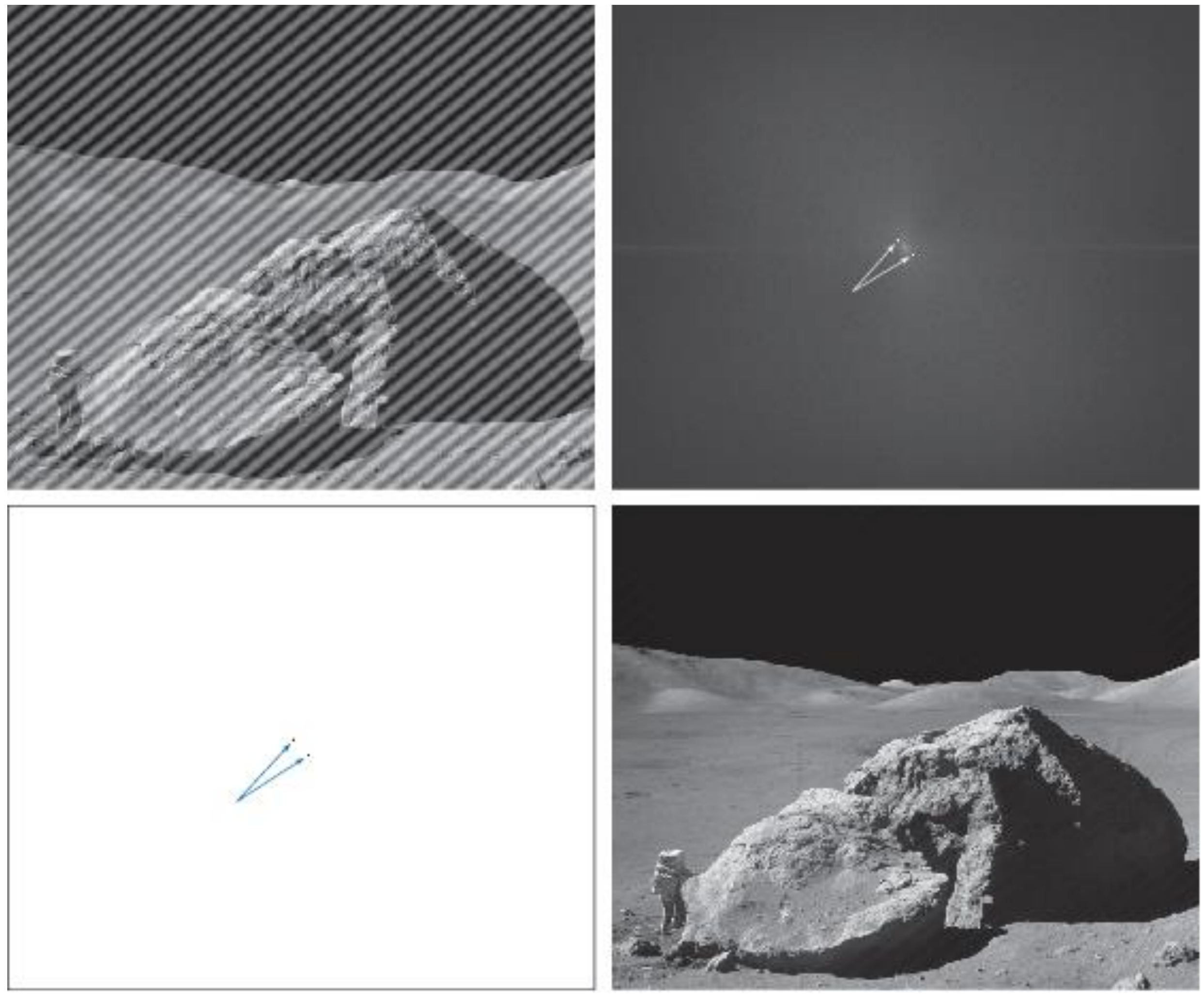


FIGURE 5.15 Perspective plots of (a) ideal, (b) Gaussian, and (c) Butterworth notch reject filter transfer functions.

Image corrupted by sinusoidal noise

Fourier spectrum of corrupted image



a b
c d

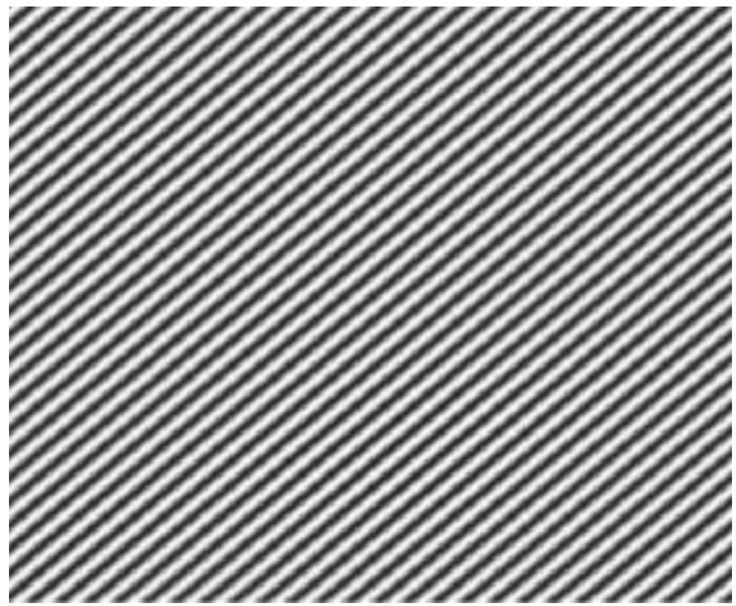
FIGURE 5.16 (a) Image corrupted by sinusoidal interference. (b) Spectrum showing the bursts of energy caused by the interference. (The bursts were enlarged for display purposes.) (c) Notch filter (the radius of the circles is 2 pixels) used to eliminate the energy bursts. (The thin borders are not part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

Notch filter

Filtered image

Extracted Sinusoidal Pattern

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

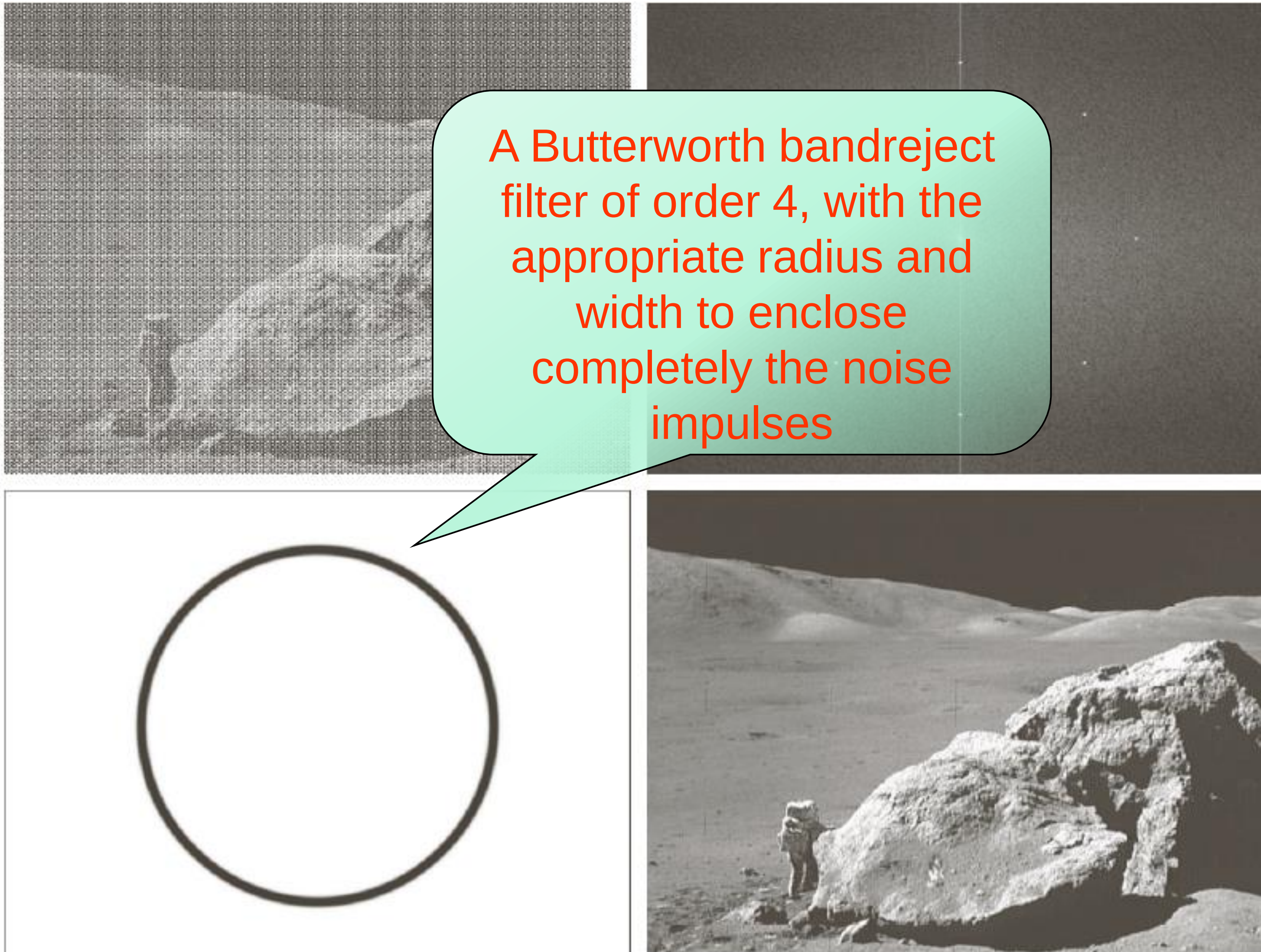


a b
c d

FIGURE 5.16 (a) Image corrupted by sinusoidal noise. (b) Spectrum of (a). (c) Butterworth bandreject filter (white represents 1). (d) Result of filtering. (Original image courtesy of NASA.)

Image corrupted by sinusoidal noise

Fourier spectrum of corrupted image



A Butterworth bandreject filter of order 4, with the appropriate radius and width to enclose completely the noise impulses

Butterworth band reject filter

Filtered image

3rd Edition

Performance of Notch Filters

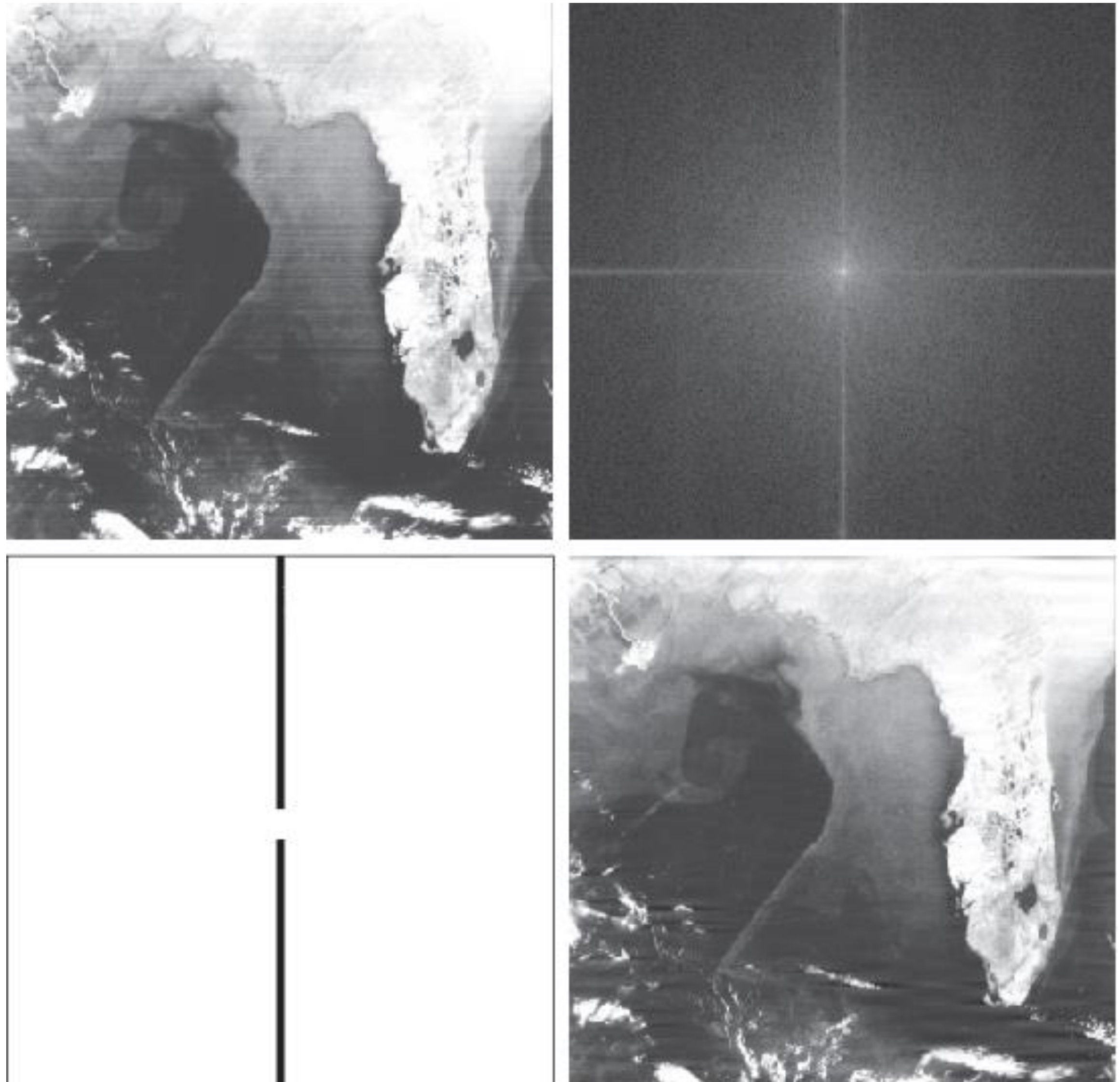
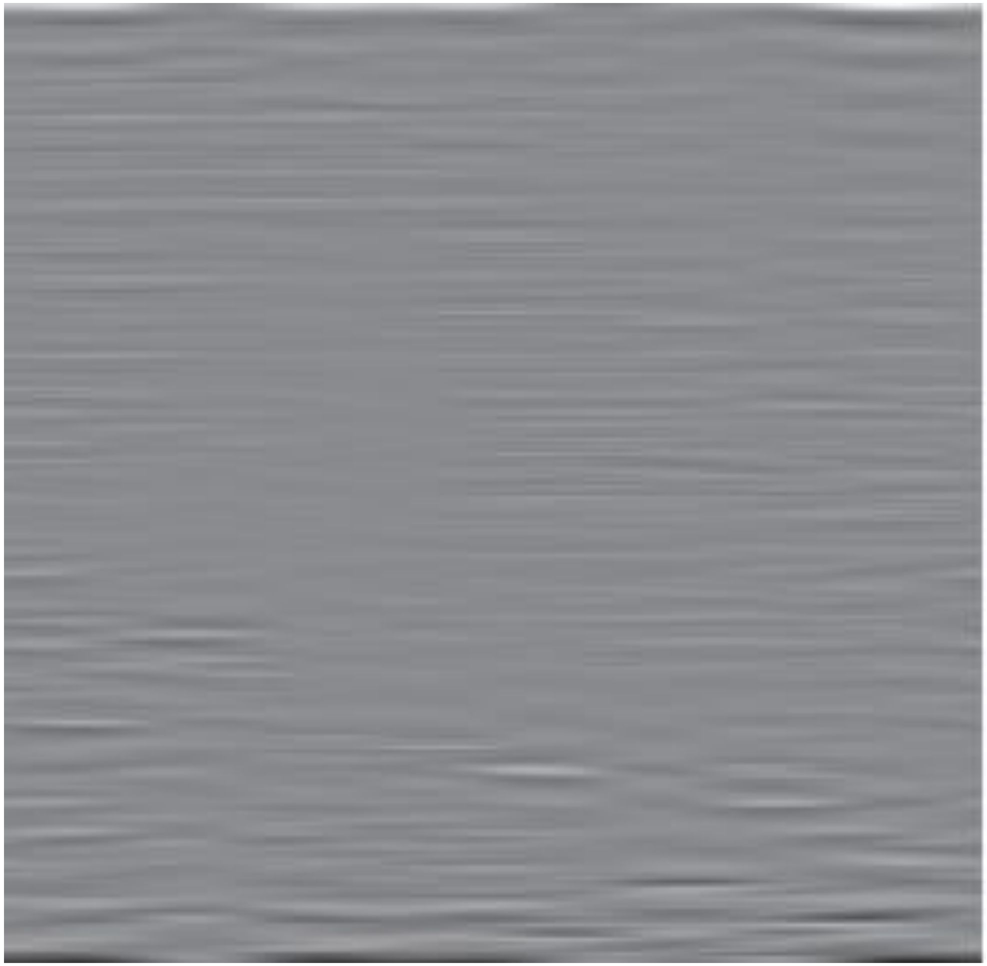


FIGURE 5.18 (a) Satellite image of Florida and the Gulf of Mexico. (Note horizontal sensor scan lines.) (b) Spectrum of (a). (c) Notch reject filter transfer function. (The thin black border is not part of the data.) (d) Filtered image. (Original image courtesy of NOAA.)

Performance of Notch Filters

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



5.4.4 Optimum Notch Filtering

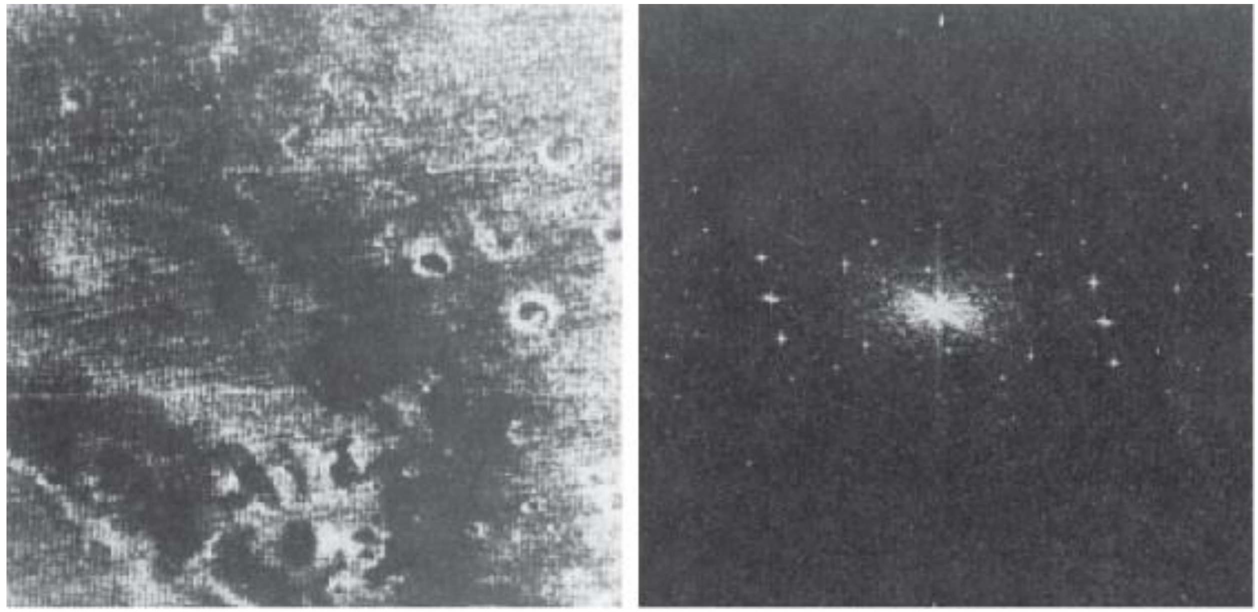


FIGURE 5.20 (a) Image of the Martian terrain taken by Mariner 6. (b) Fourier spectrum showing periodic interference. (Courtesy of NASA.)

- Several interference components are present
- The methods discussed in the preceding sections are not always acceptable because they remove much image information
- The frequency components tend to have broad skirts that carry information about the interference pattern and
- The skirts are not always easily detectable.

Optimum Notch Filtering

- Minimizes local variances of the restored estimated $f(x, y)$

Procedure for restoration tasks in multiple periodic interference

- Isolate the principal contributions of the interference pattern
- Subtract a variable, weighted portion of the interference pattern from the corrupted image

Optimum Notch Filtering: Step 1

- **Extract the principal frequency components of the interference pattern**

- Use Notch-Pass filters to isolate the interference at the locations of each spike.

$$N(u,v) = H_{NP}(u,v)G(u,v)$$

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

- Designing the Notch-Pass filter requires considerable judgment on a case by case basis → Done interactively

Optimum Notch Filtering: Step 2

Filtering procedure usually yields only an approximation of the true pattern. The effect of components not present in the estimate of $\eta(x, y)$ can be minimized instead by subtracting from $g(x, y)$ a weighted portion of $\eta(x, y)$ to obtain an estimate of $f(x, y)$:

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

One approach is to select $w(x, y)$ so that the variance of the estimate $f(x, y)$ is minimized over a specified neighborhood of every point (x, y).

Optimum Notch Filtering: Step 2

Consider a neighborhood S_{xy} of size $m \times n$

The local variance of $f(x, y)$:

$$\sigma^2(x, y) = \frac{1}{mn} \sum_{(r,c)\in S_{xy}} \left[f(r,c) - \overline{f} \right]^2$$

where,

$$\overline{f} = \frac{1}{mn} \sum_{(r,c)\in S_{xy}} f(r,c)$$

Optimum Notch Filtering: Step 3

The local variance of $f(x, y)$:

$$\sigma^2(x,y) = \frac{1}{mn} \sum_{(r,c)\in S_{xy}} \left\{ \left[g(r,c) - w(r,c)\eta(r,c) \right] - \left[\bar{g} - \overline{w\eta} \right] \right\}^2$$

where,

$\bar{g}$: Average of $g(r,c)$ in S_{xy}

$\overline{w\eta}$: Average of the product $w(r,c)\eta(r,c)$ in S_{xy}

Assuming $w(x, y)$ remains constant over S_{xy}

$w(r,c) = w(x,y)$ and $\overline{w\eta} = w(x,y)\bar{\eta}$

Finally, $\sigma^2(x,y) = \frac{1}{mn} \sum_{(r,c)\in S_{xy}} \left\{ \left[g(r,c) - w(x,y)\eta(r,c) \right] - \left[\bar{g} - w(x,y)\bar{\eta} \right] \right\}^2$

Optimum Notch Filtering: Step 4

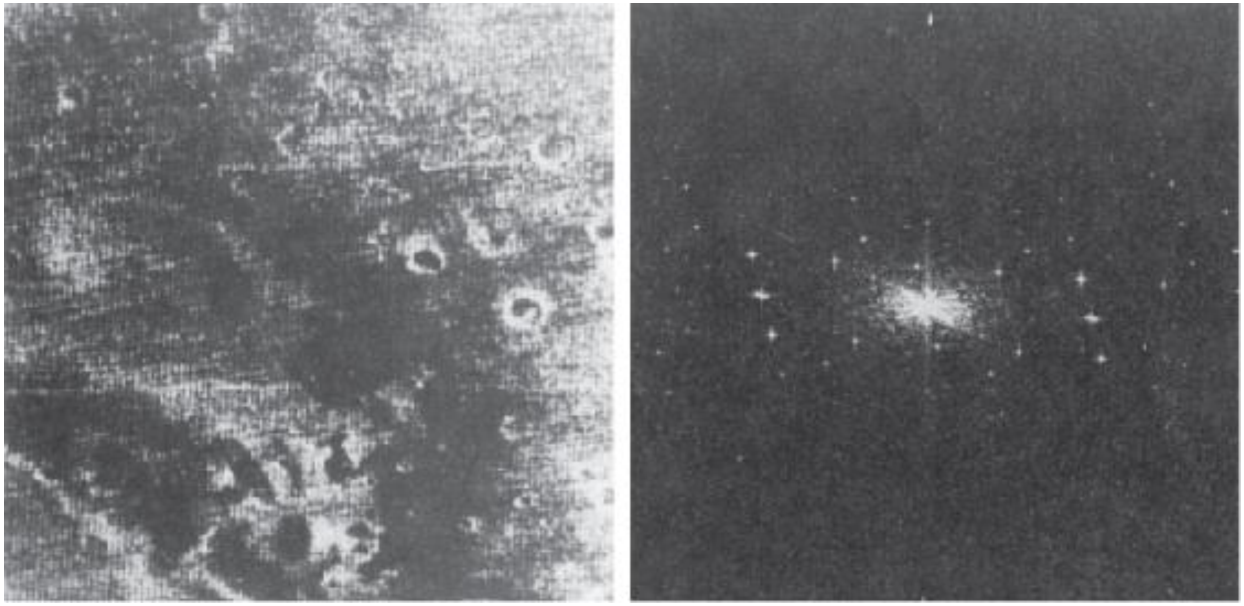
To minimize, $\sigma^2(x,y)$ w.r.t. $w(x,y)$,

solve: $$\frac{\partial\sigma^2(x,y)}{\partial w(x,y)} = 0$$

The result is,

$$w(x,y) = \frac{\overline{g\eta} - \bar{g}\bar{\eta}}{\overline{\eta^2} - \bar{\eta}^2}$$

Optimum Notch Filtering: Example



a b

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

Optimum Notch Filtering: Example

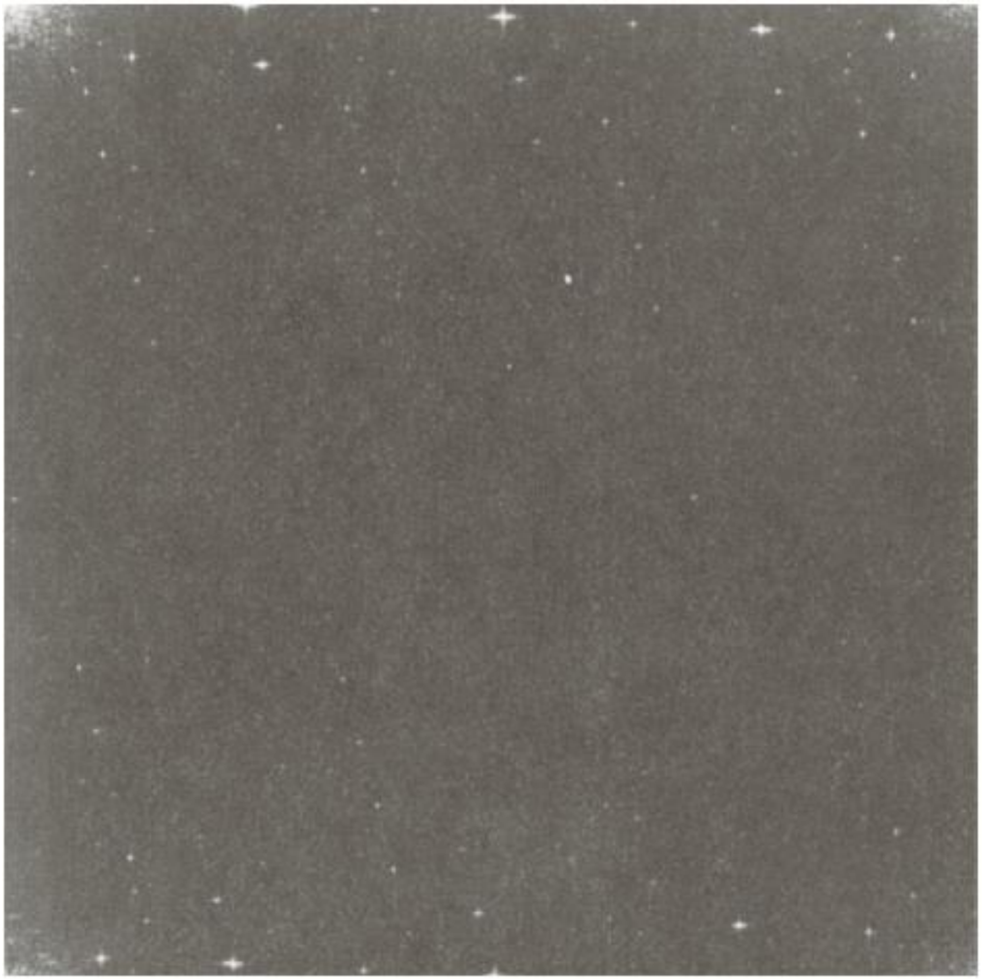


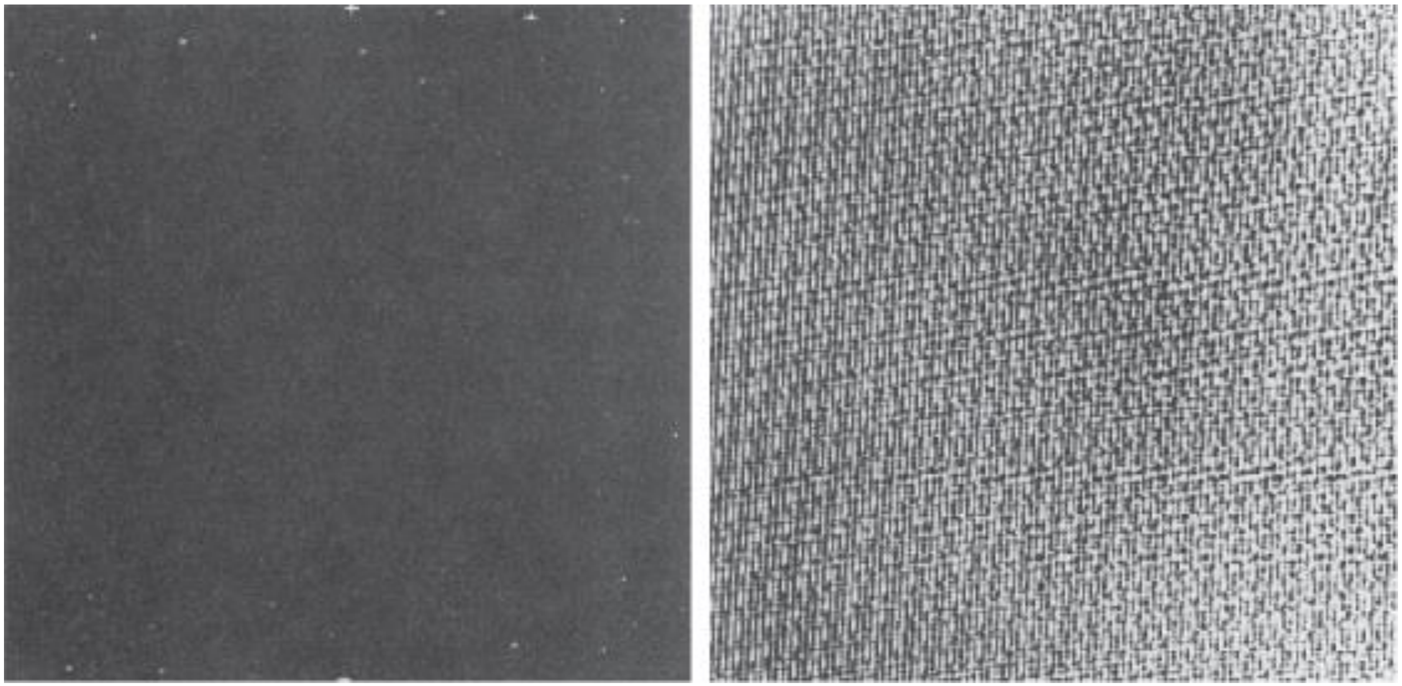
FIGURE 5.21 Uncentered Fourier spectrum of the image in Fig. 5.20(a). (Courtesy of NASA.)

- Note: The origin was not shifted in this case.
- The $u = v = 0$ is at the top-left corner.

Optimum Notch Filtering: Example

Noise Only Patterns:

$N(u, v)$ $n(x, y)$



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.)

Optimum Notch Filtering: Example

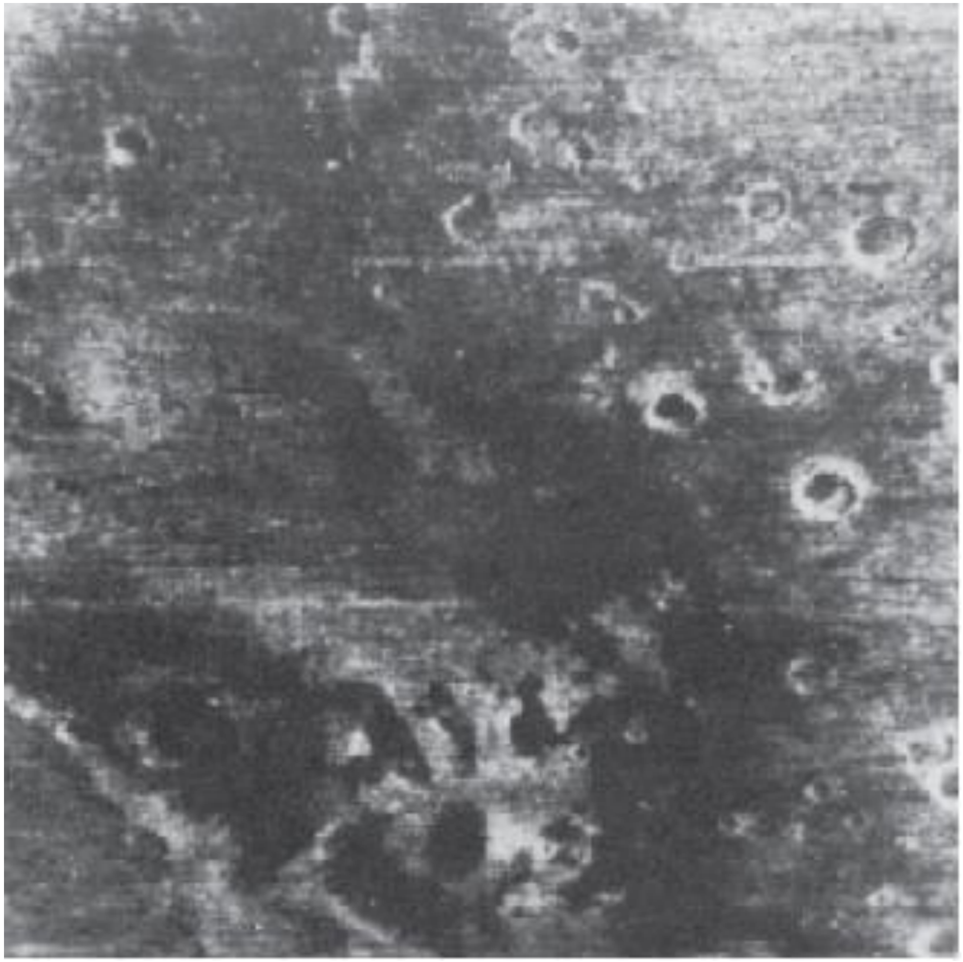


FIGURE 5.23 Restored image. (Courtesy of NASA.)

WRIGHT STATE UNIVERSITY

Chapter-5

Image Restoration & Reconstruction

Part-2

5.5 Linear, Position-Invariant Degradations

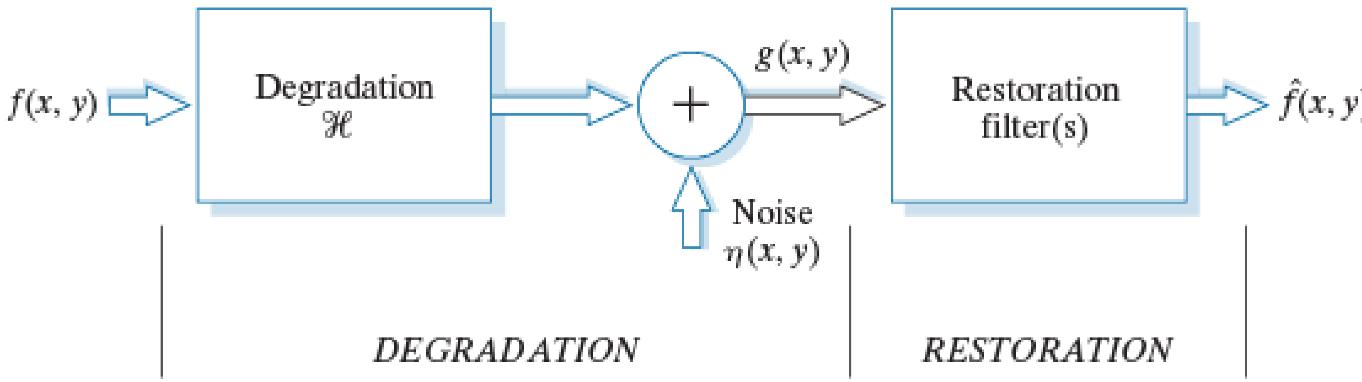


FIGURE 5.1 A model of the image degradation/restoration process.

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

Linear, Position-Invariant Degradations

$\mathcal{H}$ is linear

$$\mathcal{H}\left[af_1(x,y)+bf_2(x,y)\right]=a\mathcal{H}\left[f_1(x,y)\right]+b\mathcal{H}\left[f_2(x,y)\right]$$

f_1 and f_2 are any two input images.

An operator having the input-output relationship

$g(x,y)=\mathcal{H}\left[f(x,y)\right]$ is said to be position invariant

if

$$\mathcal{H}\left[f(x-\alpha,y-\beta)\right]=g(x-\alpha,y-\beta)$$

for any $f(x,y)$ and any α and β.

WRIGHT STATE UNIVERSITY

Linear, Position-Invariant Degradations

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\delta(x-\alpha,y-\beta)\,d\alpha\, d\beta$$

Assume for a moment that $\eta(x,y)=0$

if $\mathcal{H}$ is a linear operator,

ition (or
olm)
of the
ind

e
se

Linear, Position-Invariant Degradations

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\delta(x-\alpha,y-\beta)d\alpha d\beta$$

Assume for a moment that $\eta(x,y)=0$

if $\mathcal{H}$ is a linear operator,

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

$$=\mathcal{H}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\delta(x-\alpha,y-\beta)d\alpha d\beta\right]$$

$$=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\mathcal{H}\left[f(\alpha,\beta)\delta(x-\alpha,y-\beta)\right]d\alpha d\beta$$

$$=\underline{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\underline{\mathcal{H}\left[\delta(x-\alpha,y-\beta)\right]}d\alpha d\beta}$$

Superposition (or Fredholm) integral of the first kind

Impulse response

Linear, Position-Invariant Degradations

Assume for a moment that $\eta(x, y) = 0$

if $\mathcal{H}$ is a linear operator and position invariant,

$$\mathcal{H}\left[\delta(x-\alpha, y-\beta)\right] = h(x-\alpha, y-\beta)$$

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

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha, \beta)\mathcal{H}\left[\delta(x-\alpha, y-\beta)\right] d\alpha\, d\beta$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha, \beta) h(x-\alpha, y-\beta) d\alpha\, d\beta$$

Convolution integral in 2-D

Linear, Position-Invariant Degradations

In the presence of additive noise,

if $\mathcal{H}$ is a linear operator and position invariant,

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

$$=h(x,y) * f(x,y)+\eta(x,y)$$

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

5.6 Estimation of Degradation Model

Degradation model: $$g(x,y) = f(x,y) * h(x,y) + \eta(x,y)$$

or

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

Purpose: Estimate $h(x,y)$ or $H(u,v)$

Why?

If we know exactly $h(x,y)$, regardless of noise, we can perform deconvolution to get $f(x,y)$ back from $g(x,y)$.

5.6 Estimating the Degradation Function

- Three principal ways to estimate the degradation function

1. Observation

2. Experimentation

3. Mathematical Modeling

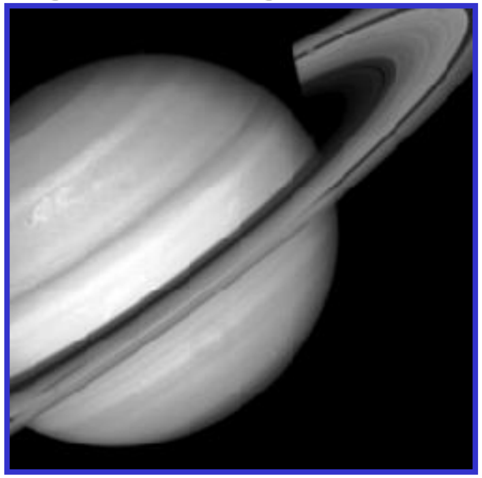
5.6.1 Estimation by Image Observation

- Take a subimage with simple structures from image
- Construct estimate of what the subimage should have been prior to degradation
- Determine subimage degradation function $h_s(x,y)$ based on the observed subimage $g_s(x,y)$ and constructed subimage $f_s(x,y)$
- Reconstruct complete degradation function $h(x,y)$ based on $h_s(x,y)$
- Works based on the assumption of position invariance
- Laborious Process – Used rarely for special cases

5.6.1 Estimation by Image Observation

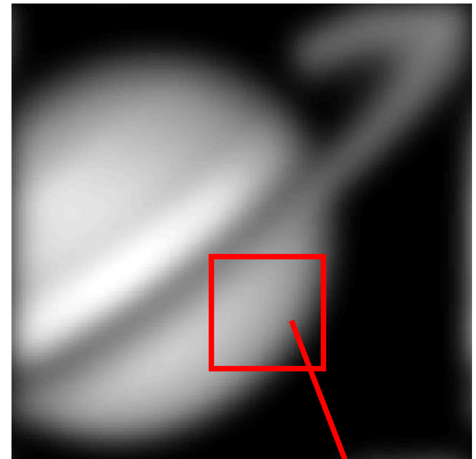
Original image (unknown)

$f(x,y)$



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

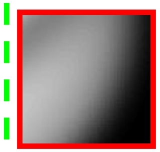
Degraded image



$g(x,y)$

Observation

Subimage



$g_s(x,y)$

DFT

$G_s(u,v)$

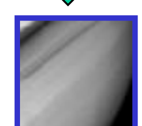
Estimated Transfer function

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

By sharpening or by processing by hand

Restoration process by estimation

Reconstructed Subimage



$\hat{f}_s(x,y)$

DFT

$\hat{F}_s(u,v)$

This case is used when we know only $g(x,y)$ and cannot repeat the experiment!

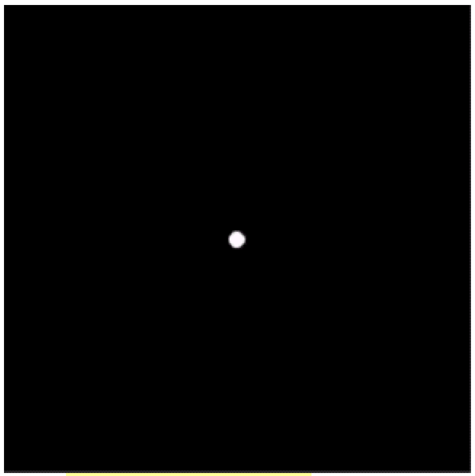
5.6.2 Estimation by Experiment

- Useful if equipment similar to equipment used to acquire degraded image is available
- Image an impulse (small dot of light) and adjust settings until the impulse response is close to that produced by the degradation
- Use the estimated degradation function to restore image
- Works based on the assumption of linearity and space invariance

5.6.2 Estimation by Experiment

- Used for same equipment set up & can repeat experiment

Input impulse image



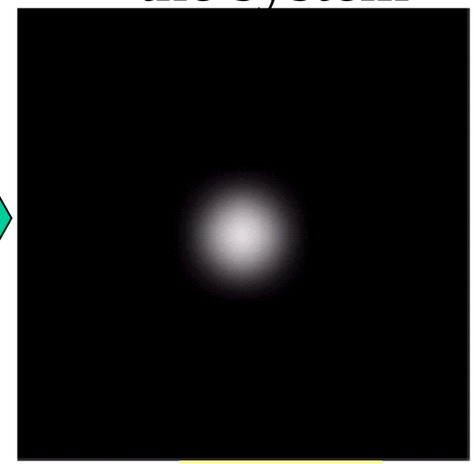
$A\delta(x,y)$

DFT

$DFT\{A\delta(x,y)\} = A$

System $H(\,)$

Response image from the system



$g(x,y)$

DFT

$G(u,v)$

$$H(u,v) = \frac{G(u,v)}{A}$$

5.6.3 Estimation by Degradation Modeling

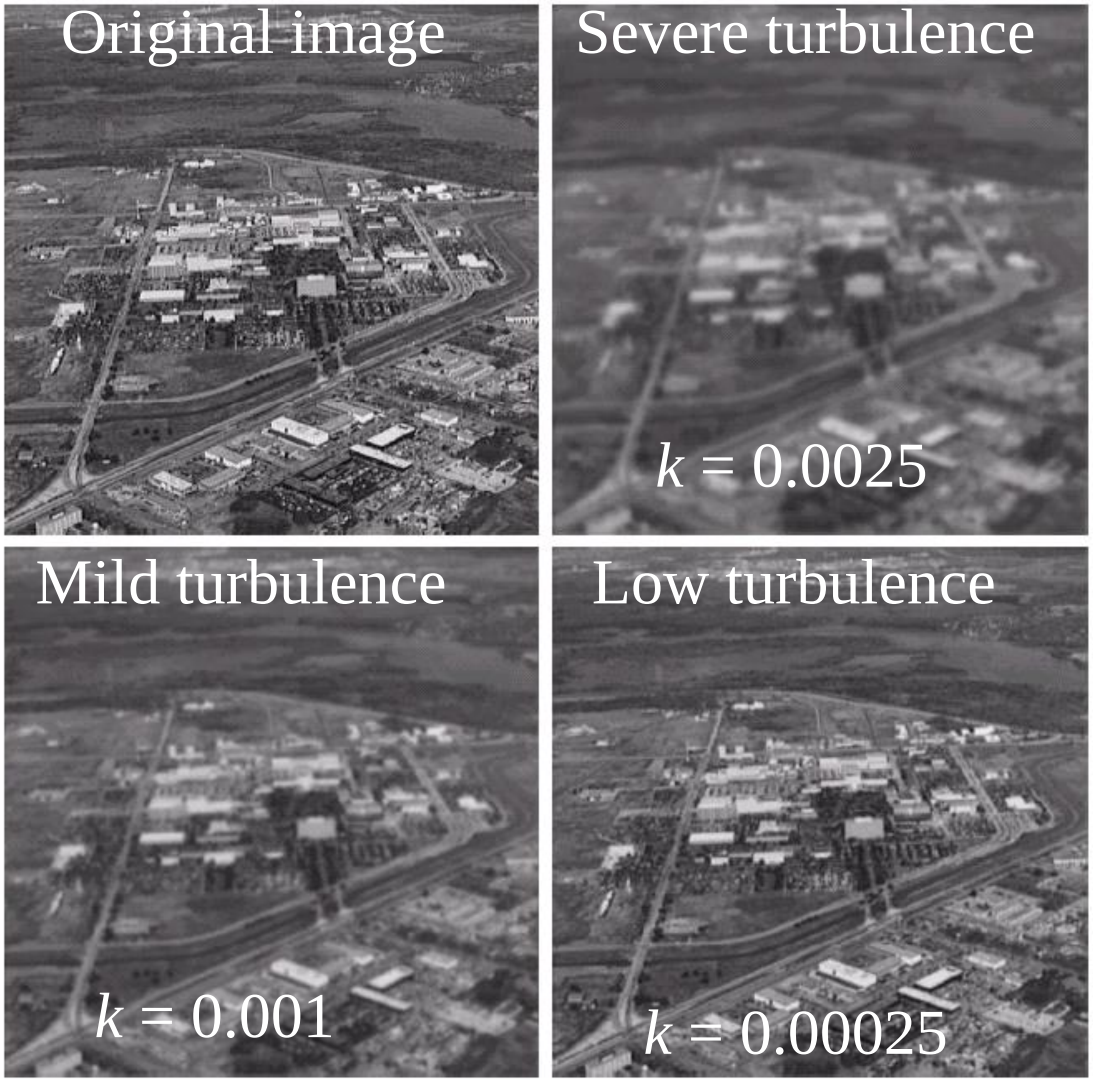
- Used when physical mechanism underlying the image formation process is known and can be expressed mathematically

- Potential physical conditions that can be modelled
  - Camera/Sensor conditions
  - Environment conditions

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.)

5.6.3 Estimation by Modeling



Example: Environmental conditions cause degradation

Atmospheric Turbulence model

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

k : constant - depends on the nature of the turbulence

Mathematical Modeling

- Derive a mathematical model from basic principles

E.g., An image blurred by uniform linear motion between the image and the sensor during image acquisition

Mathematical Modeling

- Suppose that an image $f(x, y)$ undergoes planar motion,
- $x_0(t)$ and $y_0(t)$: Camera velocities (time-varying components of motion in the x- and y-directions, respectively)
- The optical imaging process is perfect.
- T is the duration of the exposure (exposure time)
- The blurred image $g(x, y)$:

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

Mathematical Modeling

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

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

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

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

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

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

Mathematical Modeling

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

Suppose that the image undergoes uniform linear motion in the x-direction only, at a rate given by $x_0(t) = at/T$.

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

$$= \int_0^T e^{-j2\pi uat/T} dt$$

$$= \frac{T}{\pi ua} \sin(\pi ua) e^{-j\pi ua}$$

Mathematical Modeling

Suppose that the image undergoes uniform linear motion in the x-direction and y-direction, at a rate given by

$x_0(t) = at/T$ and $y_0(t) = bt/T$

For constant motion (no acceleration):

$$(x_0(t), y_0(t)) = (a\frac{t}{T}, b\frac{t}{T})$$

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

$$= \int_0^T e^{-j2\pi[ua+vb]t/T}dt$$

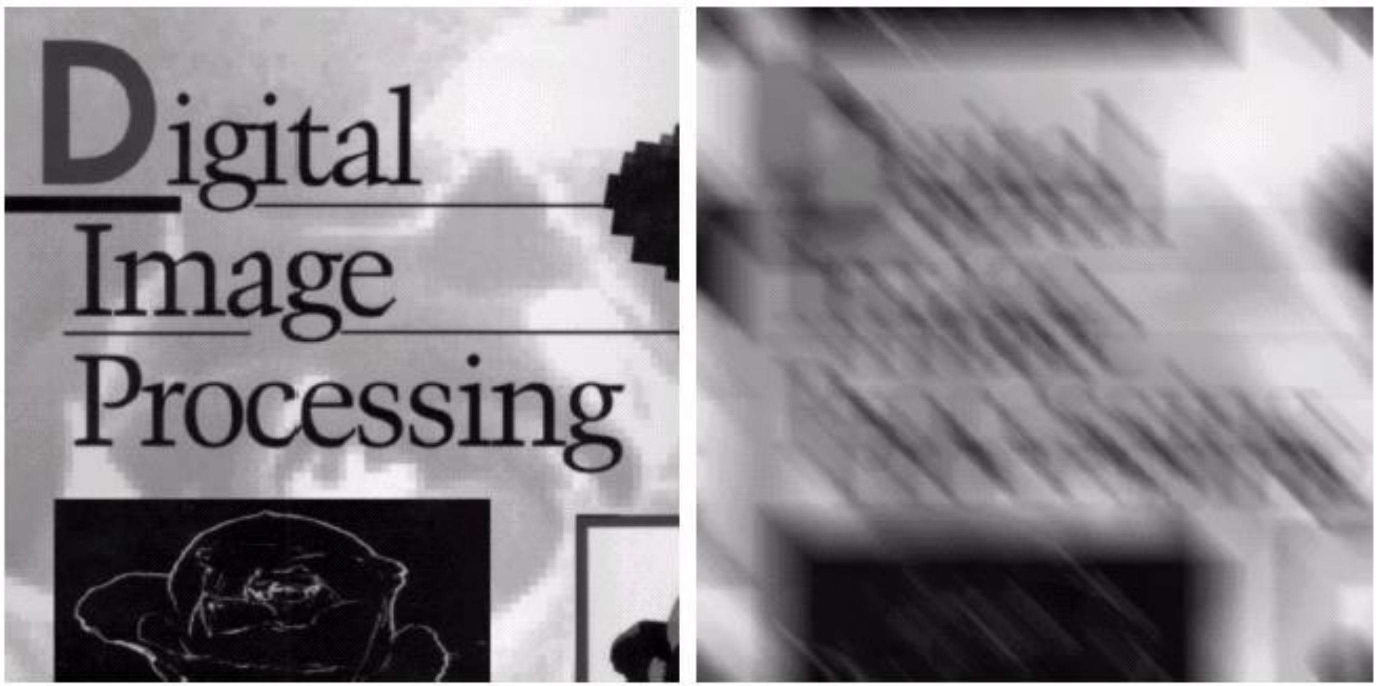
$$= \frac{T}{\pi(ua+vb)}\sin\left[\pi(ua+vb)\right]e^{-j\pi(ua+vb)}$$

WRIGHT STATE UNIVERSITY

Motion Blurring Example

- Motion Blurring Transfer Function for constant motion

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



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 $T = 1$.

Original image

Motion blurred image
$a = b = 0.1,\ T = 1$

5.7 Inverse Filtering

Degradation model: $G(u,v) = F(u,v)H(u,v) + N(u,v)$

With negligible Noise: $G(u,v) = F(u,v)H(u,v)$

Estimate of the transform of the original image:

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

$$F(u,v) = \frac{F(u,v)H(u,v) + N(u,v)}{H(u,v)}$$

$$= F(u,v) + \frac{N(u,v)}{H(u,v)}$$

Inverse Filtering

$$\hat{F}(u,v) = F(u,v) + \frac{N(u,v)}{H(u,v)}$$

1. We can't exactly recover the undegraded image because $N(u,v)$ is not known.
2. If the degradation function $H(u,v)$ has zero or very small values, then the ratio $N(u,v)/H(u,v)$ could easily dominate the estimate $\hat{F}(u,v)$.

5.7 Inverse Filter

From degradation model: $G(u,v) = F(u,v)H(u,v) + N(u,v)$

after we obtain $H(u,v)$, we can estimate $F(u,v)$ by the inverse filter:

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

Noise is enhanced when $H(u,v)$ is small.

- Usually, $H(0,0)$, *i.,e.,* at origin the degradation function H has the highest possible value

- To avoid the side effect of enhancing noise, we can apply this formulation to freq. component (u,v) only within a small radius D_0 from the center of $H(u,v)$.

- Limiting filter frequencies to near origin avoids small $H(u,v)$ values

In practice, the inverse filter is used rarely.

Inverse Filtering

EXAMPLE

The image in Fig. 5.25(b) was inverse filtered using the exact inverse of the degradation function that generated that image. That is, the degradation function is

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

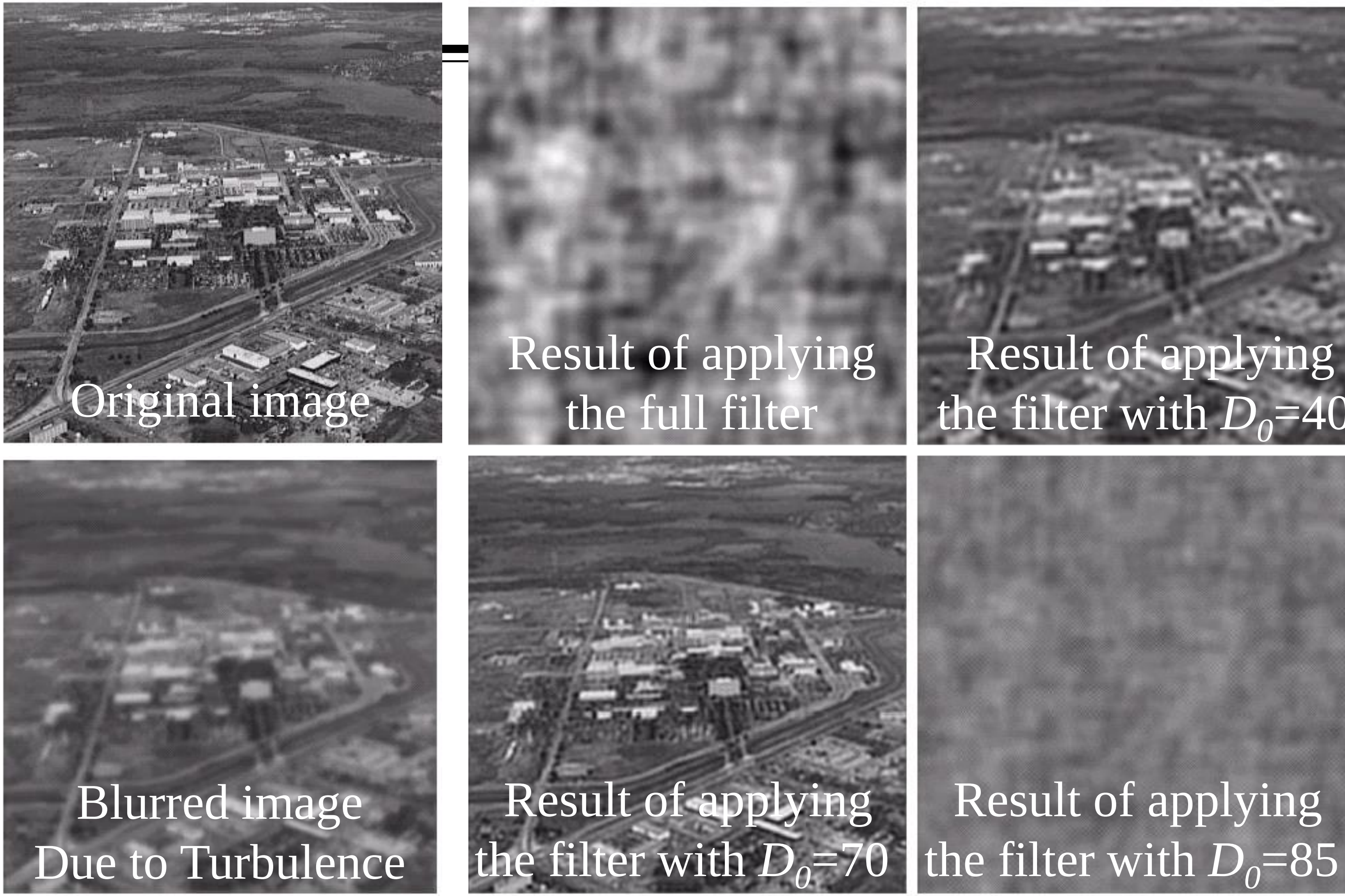
$$k = 0.0025,\ M = N = 480.$$

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.

A Butterworth lowpass function of order 10

Inverse Filter: Example



$$H(u,v) = e^{-0.0025(u^2+v^2)^{5/6}}$$

5.8 Minimum Mean Square Error (Wiener) Filtering

Norbert Wiener (1942)

- **Objective**

  Find an estimate of the uncorrupted image such that the mean square error between them is minimized

$$e^2 = E\left\{(f - f)^2\right\}$$

Minimum Mean Square Error (Wiener) Filtering

The minimum of the error function is given in the frequency domain by the expression

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

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

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

Minimum Mean Square Error (Wiener) Filtering

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

$H(u,v)$: degradation function

$H^*(u,v)$: complex conjugate of $H(u,v)$

$|H(u,v)|^2 = H^*(u,v)H(u,v)$

$S_\eta(u,v) = |N(u,v)|^2 =$ power spectrum of the noise

$S_f(u,v) = |F(u,v)|^2 =$ power spectrum of the undegraded image

Minimum Mean Square Error (Wiener) Filtering

Wiener Filter Formula:

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

Difficult to estimate

Approximated Formula:

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

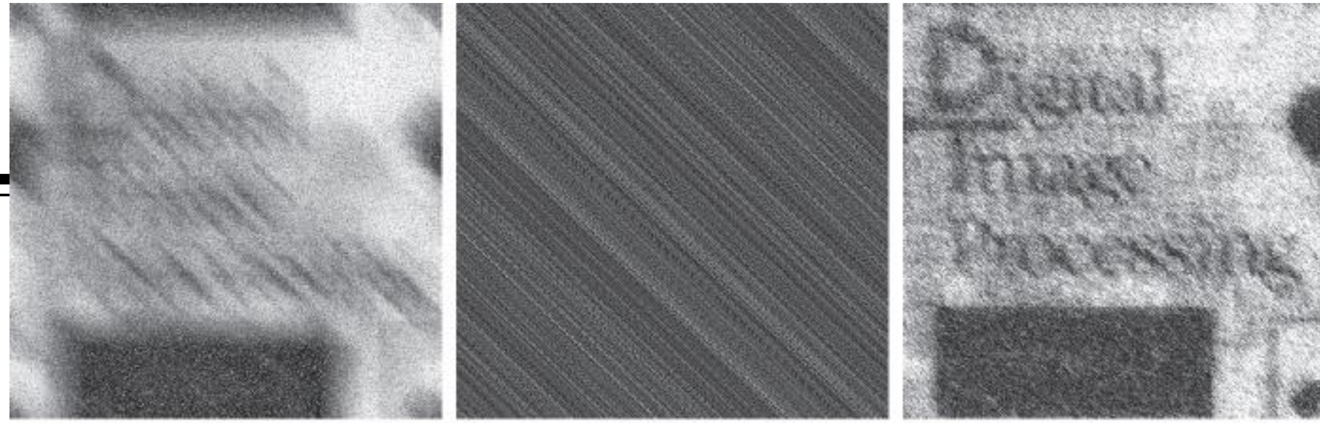
K is a specified constant.

In practice, K is chosen manually to obtained the best visual result!

Minimum Mean Square Error (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.



Left: degraded image

Middle: inverse filtering

Right: Wiener filtering

a	b	c
d	e	f
g	h	i

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.

Wiener 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.

Some Measures

Singal-to-Noise Ratio (SNR)

$$SNR = \frac{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} |F(u,v)|^2}{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} |N(u,v)|^2}$$

This ratio gives a measure of the level of information bearing singal power to the level of noise power.

Some Measures

Mean Square Error (MSE)

$$\text{MSE}=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[f(x,y)-f(x,y)\right]^2$$

Root-Mean-Sqaure-Error (RMSE)

$$\text{RMSE}=\frac{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} f(x,y)^2}{\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}| f(x,y)-f(x,y)|^2}$$

5.9 Constrained Least Squares Filtering

- In Wiener filter, the power spectra of the undegraded image and noise must be known. Although a constant estimate is sometimes useful, it is not always suitable.

- Constrained least squares filtering just requires the mean and variance of the noise.

5.9 Constrained Least Squares Filter

Degradation model: | Written in a matrix form

$$g(x,y) = f(x,y) * h(x,y) + \eta(x,y) \qquad \mathbf{g} = \mathbf{Hf} + \boldsymbol{\eta}$$

Objective: Find the minimum of a criterion function

$$C = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\nabla^2 f(x,y)\right]^2 \quad \text{(Sum of Laplacians)}$$

Subject to the constraint

$$\left\|\mathbf{g} - \mathbf{H}\hat{\mathbf{f}}\right\|^2 = \left\|\boldsymbol{\eta}\right\|^2 \qquad \text{where,} \quad \left\|\mathbf{w}\right\|^2 = \mathbf{w}^T\mathbf{w}$$

We get a constrained least square filter

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

where

$$P(u,v) = \text{Fourier transform of } p(x,y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Constrained Least Squares Filter: Example

Constrained least square filter

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

γ is adaptively adjusted to achieve the best result.

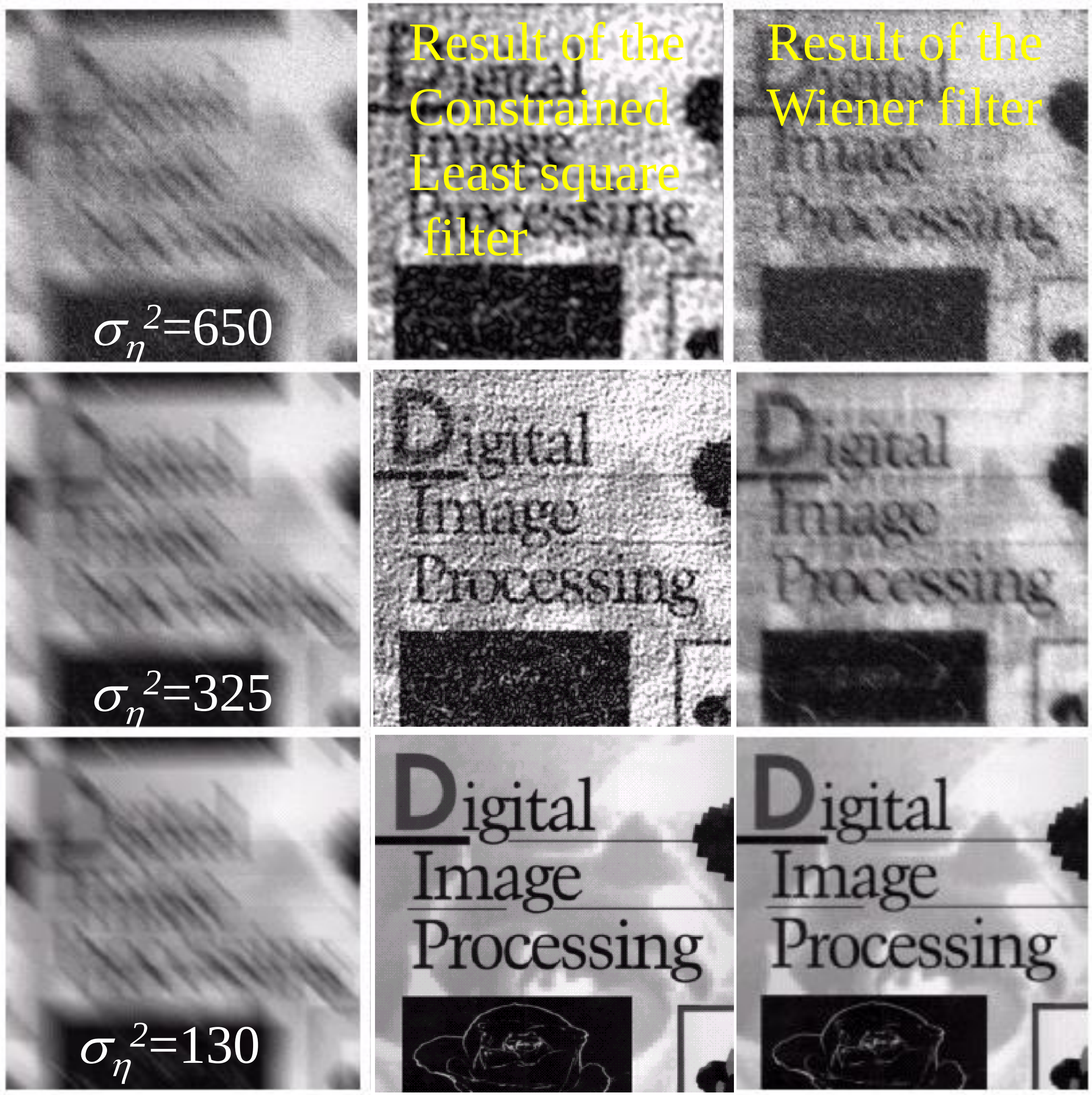
a b c

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.

- This result obtained with constrained least square filter
- Compare with Results in the previous Figure 5.29 (c)

Constrained Least Squares Filter: Example (cont.)

Image degraded by motion blur + AWGN



Constrained Least Squares Filter:Adjusting γ

Define $\mathbf{r} = \mathbf{g} - \mathbf{H}\hat{\mathbf{f}}$

It can be shown that $\phi(\gamma) = \mathbf{r}^T\mathbf{r} = \|\mathbf{r}\|^2$ is Monotonic increasing

We want to adjust gamma so that $\|\mathbf{r}\|^2 = \|\boldsymbol{\eta}\|^2 \pm a$ → 1

where, a = accuracy factor

1. Specify an initial value of γ

2. Compute $\|\mathbf{r}\|^2$

3. Stop if 1 is satisfied

   Otherwise return step 2 after increasing γ if $\|\mathbf{r}\|^2 < \|\boldsymbol{\eta}\|^2 - a$

   or decreasing γ if $\|\mathbf{r}\|^2 > \|\boldsymbol{\eta}\|^2 + a$

   Use the new value of γ to recompute

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

Constrained Least Squares Filter: Adjusting γ (cont.)

WRIGHT STATE

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

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

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

For computing $\|\mathbf{r}\|^2$

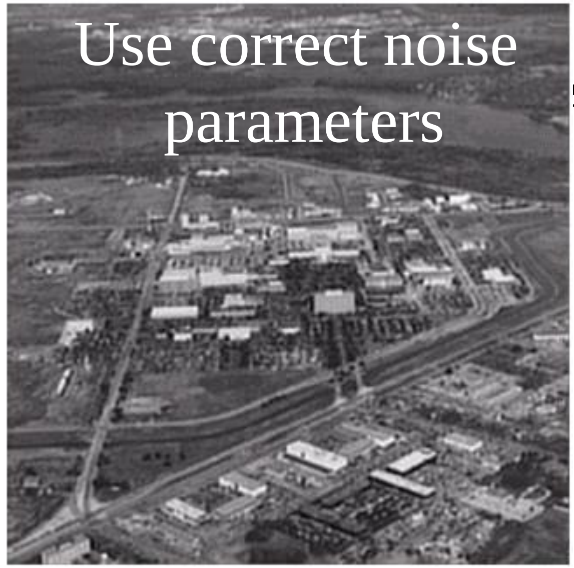
$$\bar{\eta} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \eta(x,y)$$

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

$$\|\boldsymbol{\eta}\|^2 = MN \left[\sigma_\eta^2 - \bar{\eta} \right]$$

For computing $\|\boldsymbol{\eta}\|^2$

Constrained Least Squares Filter: Example



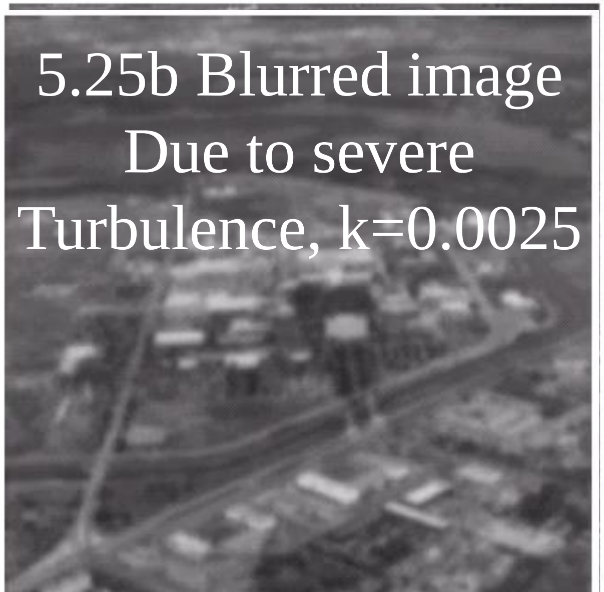
Correct parameters:

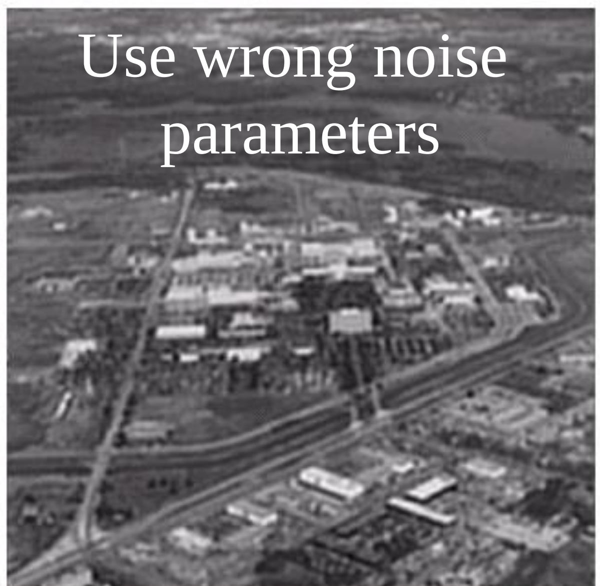
Initial $\gamma = 10^{-5}$

Correction factor $= 10^{-6}$

$a = 0.25$

$\sigma_\eta^2 = 10^{-5}$





Wrong noise parameter

$\sigma_\eta^2 = 10^{-2}$

Results obtained from constrained least square filters

5.10 Geometric Mean Filter

- Generalization of Wiener Filter

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

$\alpha=0,\beta=1:$	standard Wiener filter
$\alpha=1:$	inverse filter
$\alpha=0:$	parametric Wiener filter
$\alpha=1/2:$	geometric mean filter

5.11 Image Reconstruction from Projections

Reconstruct an image from a series of projections

X-ray Computed Tomography (CT)

"Computed tomography is a medical imaging method employing tomography where digital geometry processing is used to generate a three-dimensional image of the internals of an object from a large series of two-dimensional X-ray images taken around a single axis of rotation."

http://en.wikipedia.org/wiki/Computed_tomography

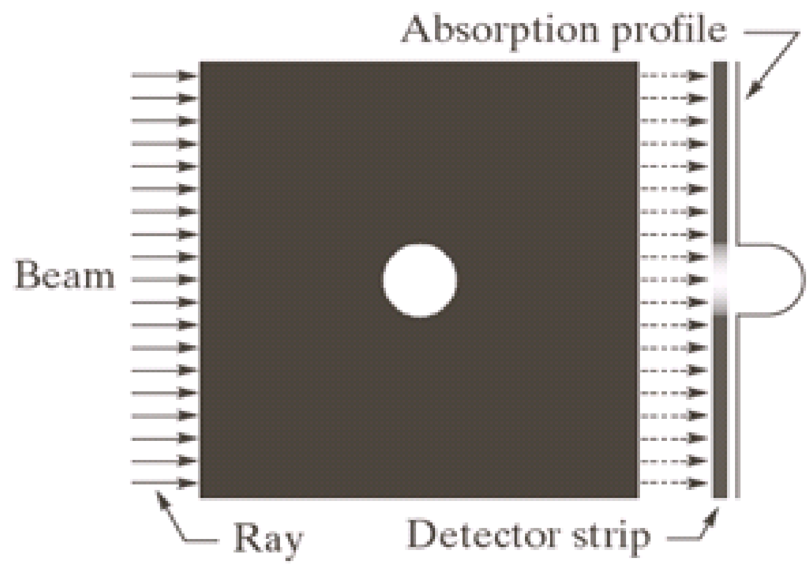
Backprojection

" In computed tomography or other imaging techniques requiring reconstruction from multiple projections, an algorithm for calculating the contribution of each voxel of the structure to the measured ray data, to generate an image; the oldest and simplest method of image reconstruction. "

http://www.medilexicon.com/medicaldictionary.php?t=9165

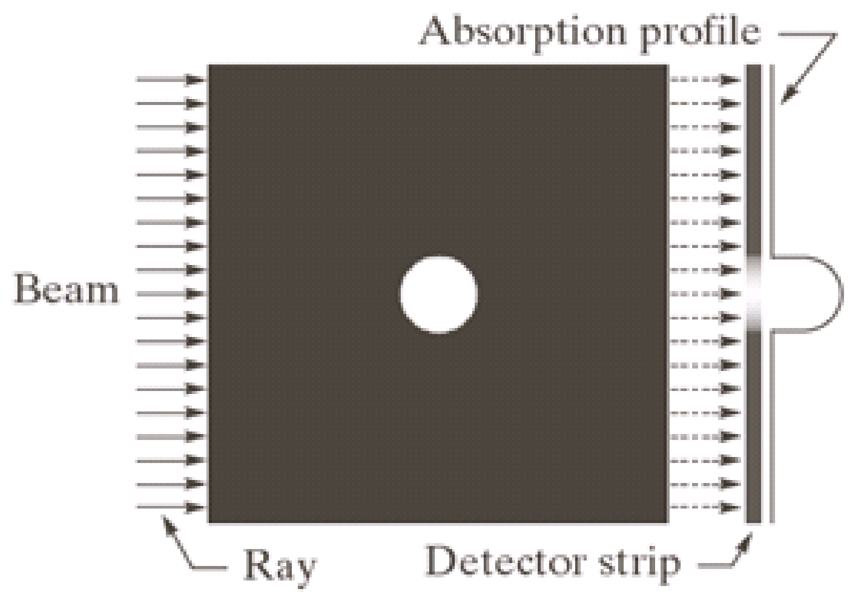
The Image Reconstruction Problem

- Consider a single object on a uniform background (suppose that this is a cross section of 3D region of a human body).
- Background represents soft, uniform tissue and the object is also uniform but with higher absorption characteristics.



The Image Reconstruction Problem (cont...)

- A beam of X-rays is emitted and part of it is absorbed by the object.
- The energy of absorption is detected by a set of detectors.
- The collected information is the absorption signal.



The Image Reconstruction Problem (cont...)

- A simple way to recover the object is to back-project the 1D signal across the direction the beam came from.
- This simply means to duplicate the signal across the 1D beam.

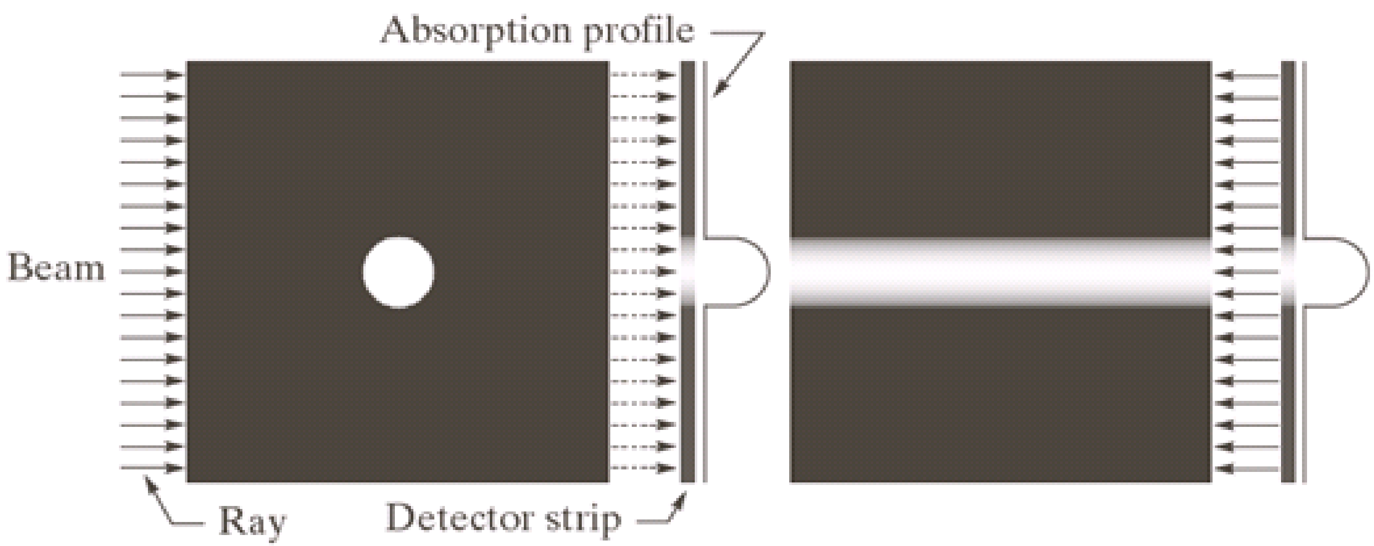


Image Reconstruction: Introduction

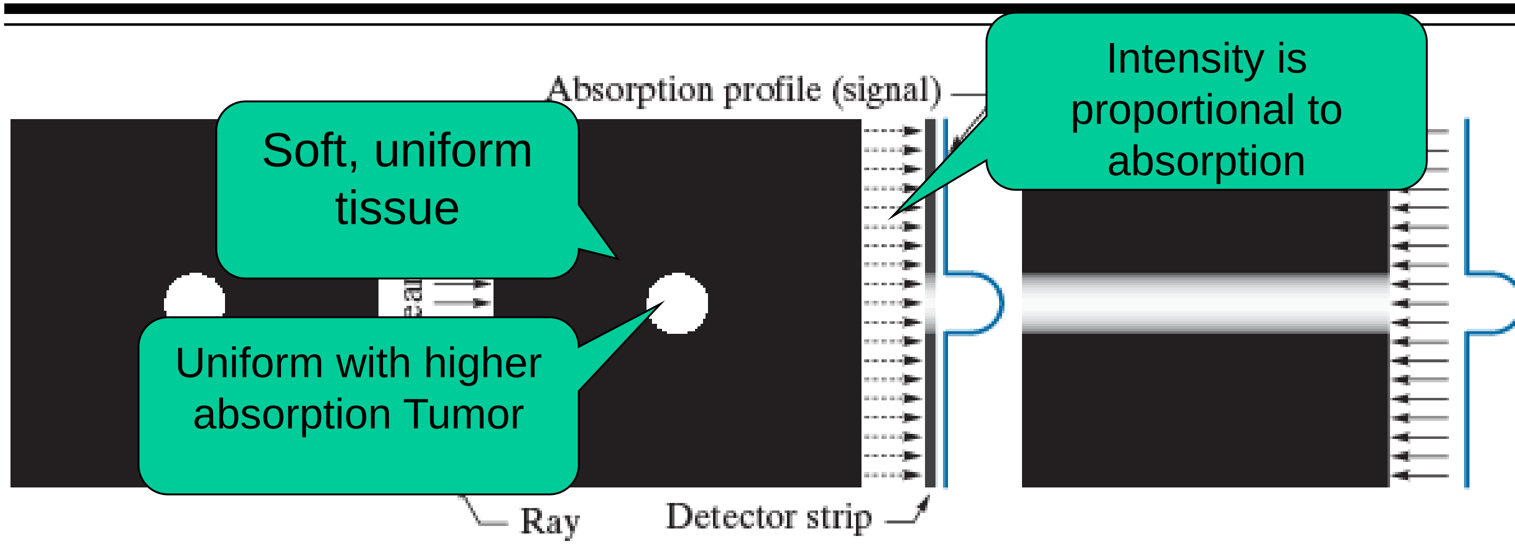


FIGURE 5.32 (a) Flat region with a single object. (b) Parallel beam, detector strip, and profile of sensed 1-D absorption signal. (c) Result of back-projecting the absorption profile. (d) Beam and detectors rotated by 90°. (e) Backprojection. (f) The sum of (c) and (e), intensity-scaled. The intensity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

The Image Reconstruction Problem (cont...)

- There is no means to determine the number of objects from a single projection.
- Rotate the position of the source-detector pair and obtain another 1D signal.
- Repeat the procedure and add the signals from the previous back-projections.
- Shows that the object of interest is located at the central square.

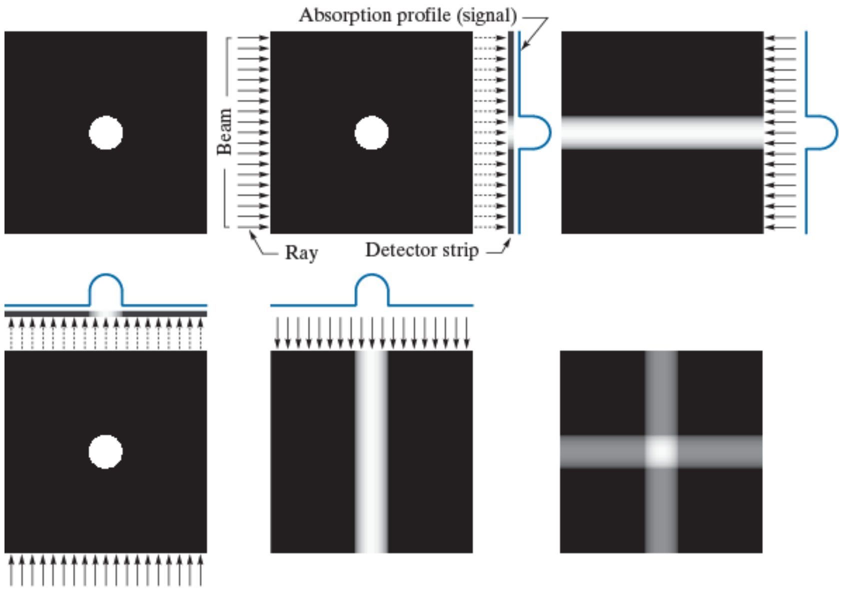


Image Reconstruction: Introduction

a b c
d e f

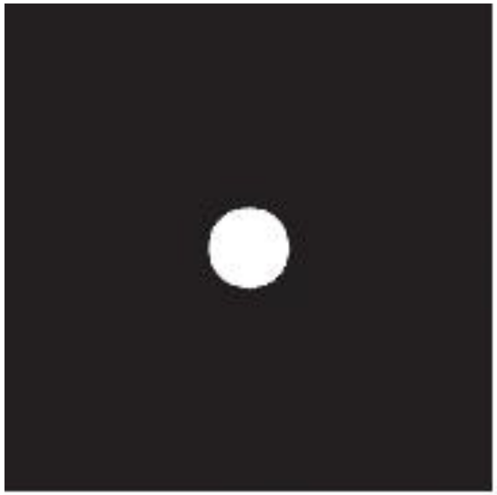
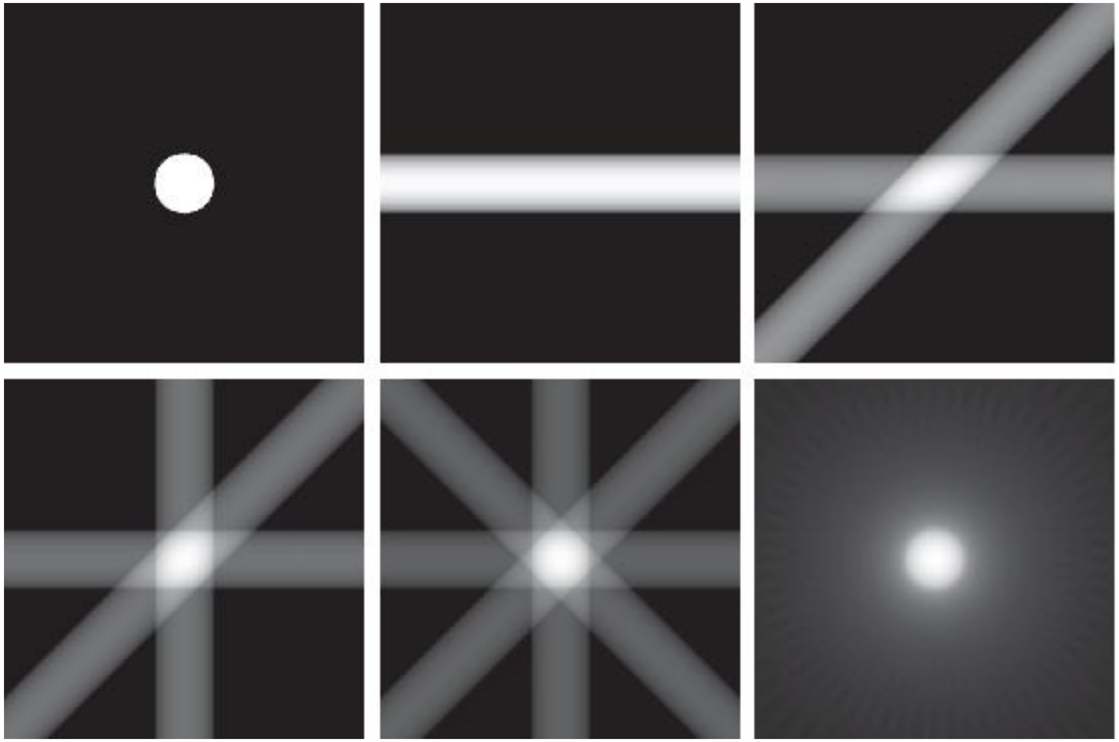


FIGURE 5.33 (a) Same as Fig. 5.32(a). (b)-(e) Reconstruction using 1, 2, 3, and 4 backprojections 45° apart. (f) Reconstruction with 32 backprojections 5.625° apart (note the blurring).

The Image Reconstruction Problem (cont…)



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).

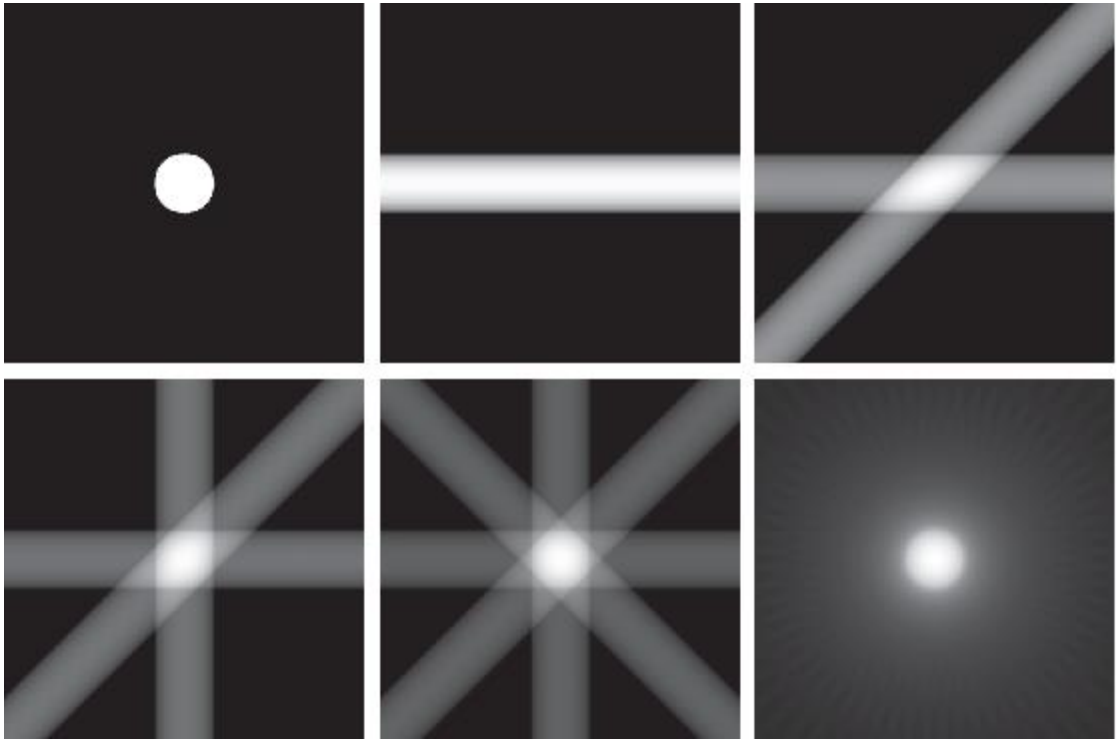
By taking more projections:

- The form of the object becomes clearer because brighter regions dominate the result
- Back-projections with few interactions with the object will fade into the background.

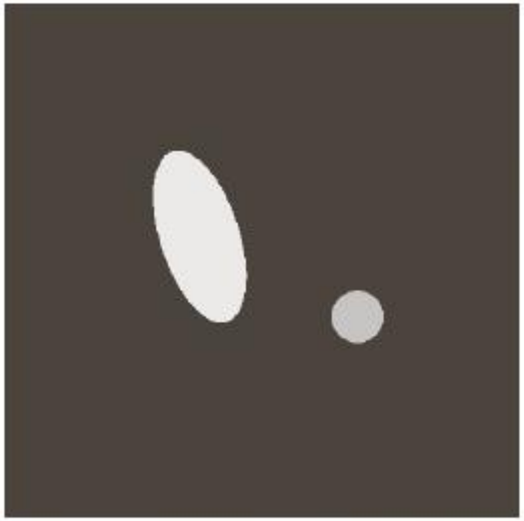
The Image Reconstruction Problem (cont...)

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).



- The image is blurred. Important problem!
- Only consider projections from 0 to 180 degrees as projections differing 180 degrees are mirror images of each other



a b c
d e f

FIGURE 5.34 (a) A region with two objects. (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.

The Image Reconstruction Problem (cont...)

a	b	c
d	e	f

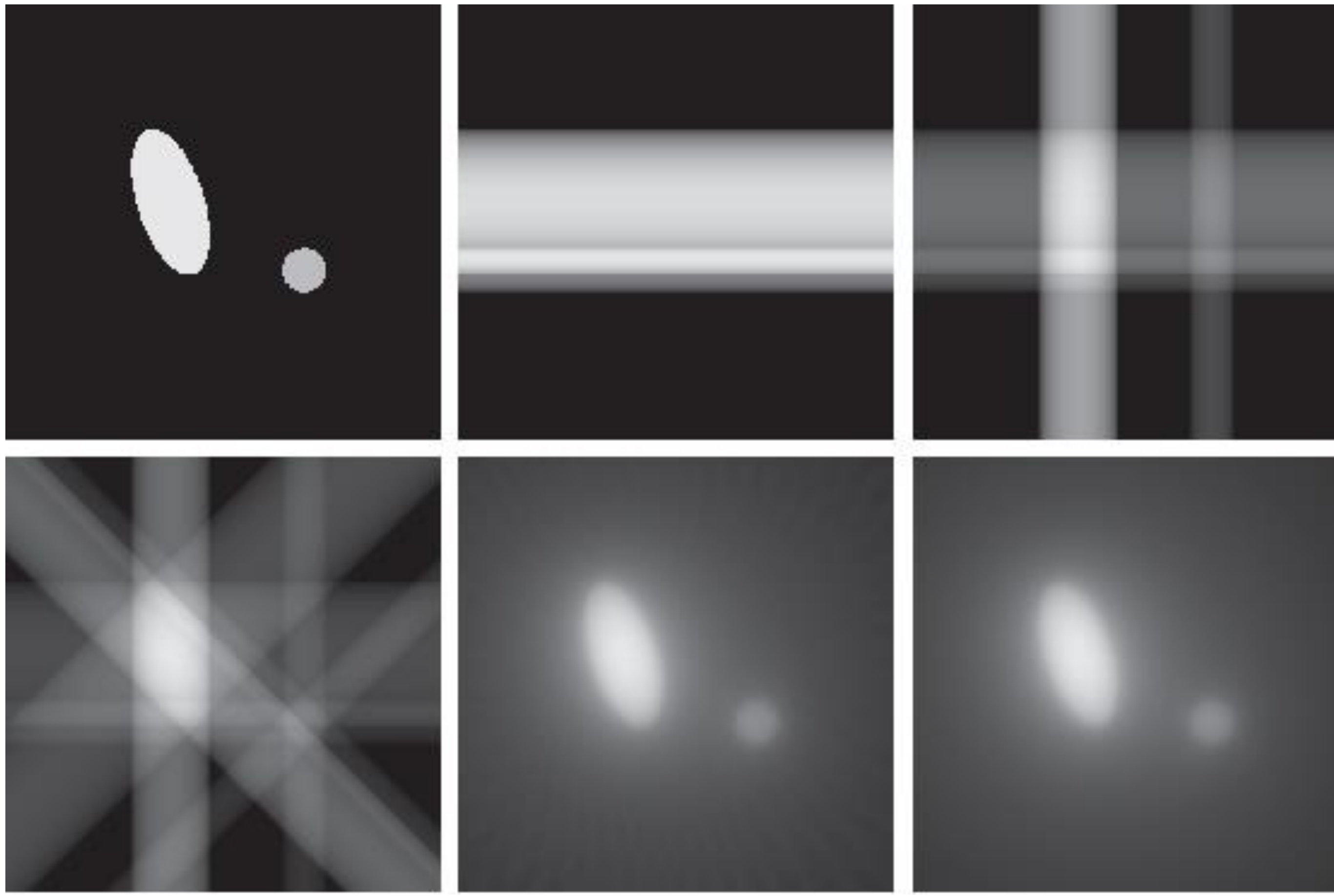


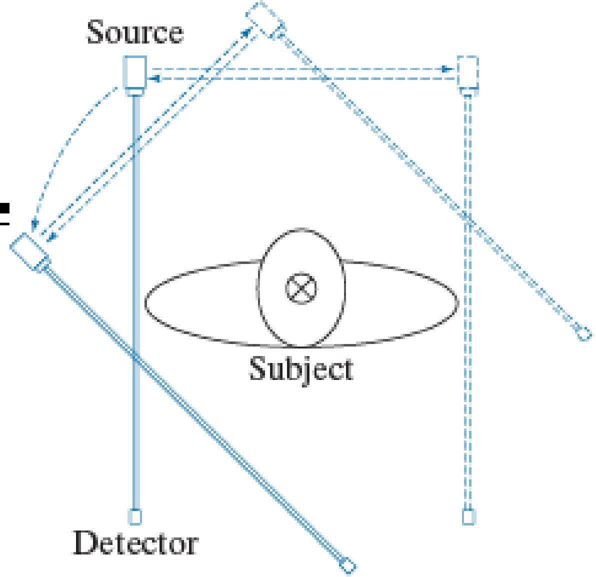
FIGURE 5.34
(a) Two objects with different absorption characteristics. (b)–(d) Reconstruction using 1, 2, and 4 backprojections, 45° apart. (e) Reconstruction with 32 backprojections, 5.625° apart. (f) Reconstruction with 64 backprojections, 2.8125° apart.

Principles of Computerized Tomography

- The goal of CT is to obtain a 3D representation of the internal structure of an object by X-raying it from many different directions.
- Imagine the traditional chest X-ray obtained by different directions. The image is the 2D equivalent of a line projections.
- Back-projecting the image would result in a 3D volume of the chest cavity.

Principles of Computerized Tomography (cont...)

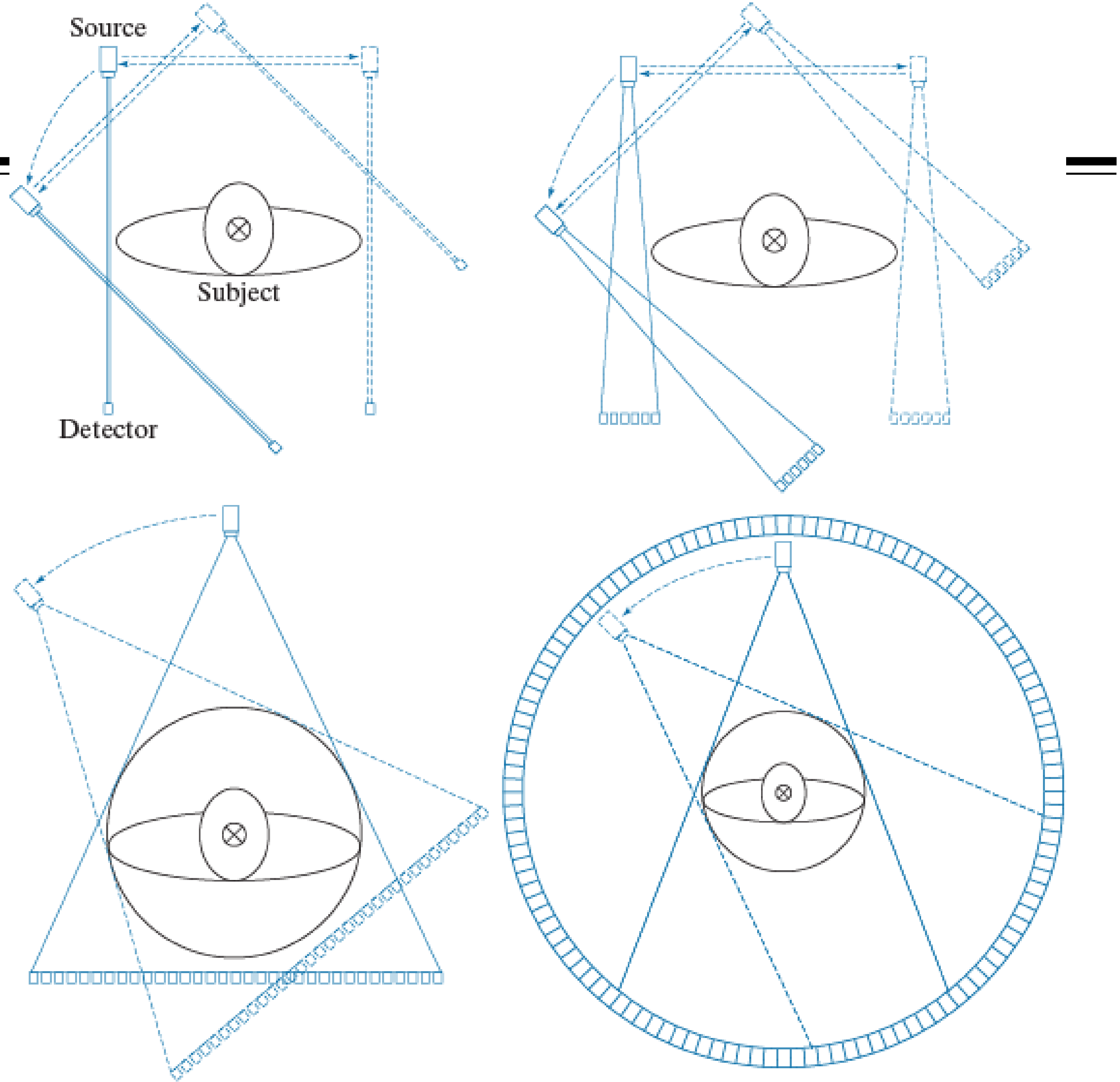
- CT gets the same information by generating slices through the body.
- A 3D representation is then obtained by stacking the slices.
- More economical due to fewer detectors.
- Computational burden and dosage is reduced.
- Theory developed in 1917 by J. Radon.
- Application developed in 1964 by A. M. Cormack and G. N. Hounsfield independently. They shared the Nobel prize in Medicine in 1979.



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.

WRIGHT STATE UNIVERSITY



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.

Other CTs

- **Electron beam CT** (Fifth-generation CT)

  Electron beam tomography (EBCT) was introduced in the early 1980s, by medical physicist Andrew Castagnini, as a method of improving the temporal resolution of CT scanners. Uses fan beam.

  High cost of EBCT equipment, and poor flexibility

- **Helical (or spiral) cone beam computed tomography** (Sixth-generation)

  A type of three dimensional computed tomography (CT) in which the source (usually of x-rays) describes a helical trajectory relative to the object while a two dimensional array of detectors measures the transmitted radiation on part of a cone of rays emitting from the source

  http://en.wikipedia.org/wiki/Computed_tomography

Other CTs

- Multislice CT (seventh-generation)

- The major benefit of multi-slice CT

  - Significant increase in detail
  - Utilizes X-ray tubes more economically
  - Reducing cost and potentially reducing dosage

The Radon Transform

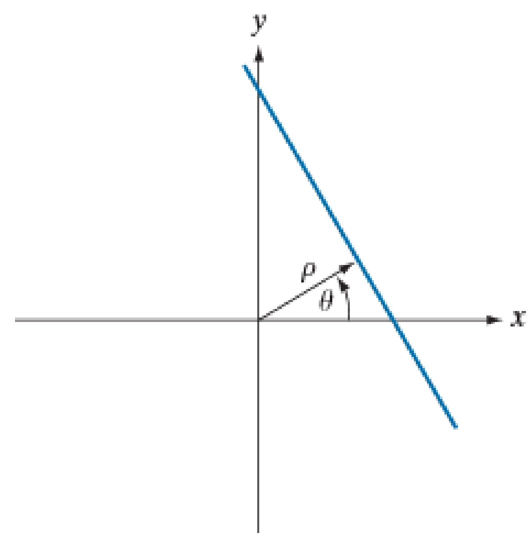
- A straight line in Cartesian coordinates may be described by its *slope-intercept* form:

$$y = ax + b$$

- Or, by its *normal representation*:

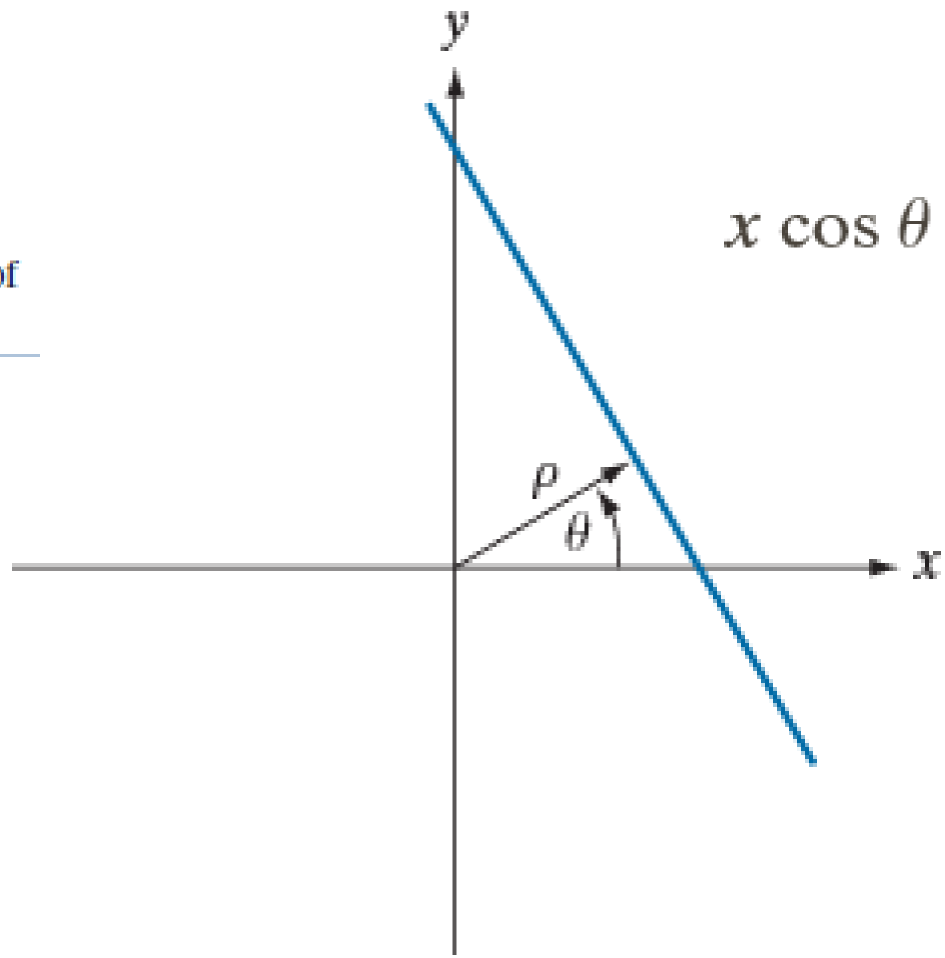
$$x\cos\theta + y\sin\theta = \rho$$

FIGURE 5.36 Normal representation of a line.



Projections and the Radon Transform

FIGURE 5.36 Normal representation of a line.



$$x \cos\theta + y \sin\theta = \rho$$

The Radon Transform (cont…)

- The projection of a parallel-ray beam may be modelled by a set of such lines.
- An arbitrary point (ρ_j, θ_k) in the projection signal is given by the ray-sum along the line,

$$x\cos\theta_k + y\sin\theta_k = \rho_j.$$

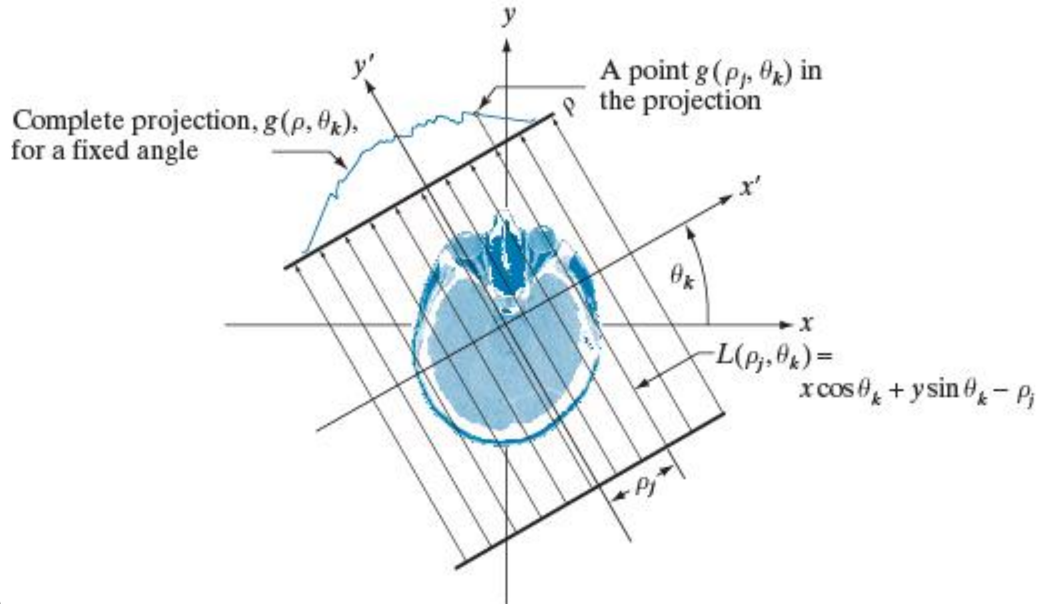


FIGURE 5.37 Geometry of a parallel-ray beam.

The Radon Transform (cont...)

The ray-sum is a line integral:

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

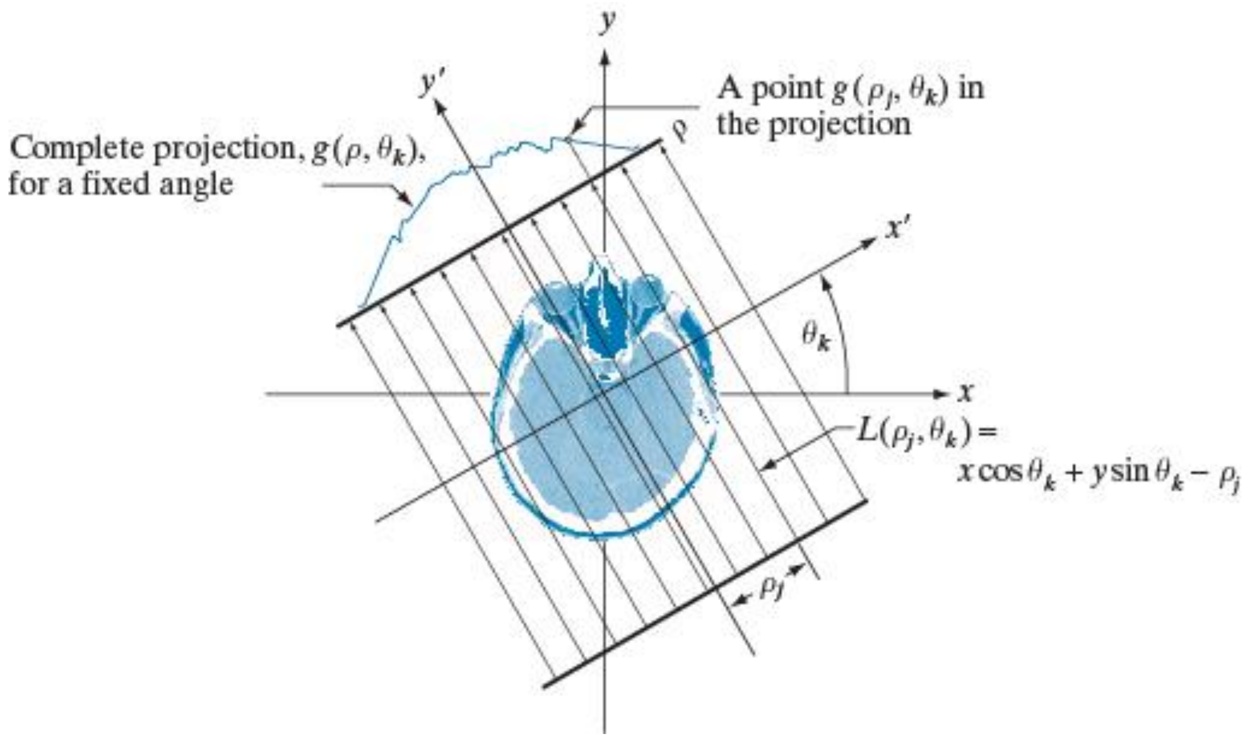


FIGURE 5.37 Geometry of a parallel-ray beam.

Projections and the Radon Transform

For all values of ρ and θ we obtain the Radon transform:

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

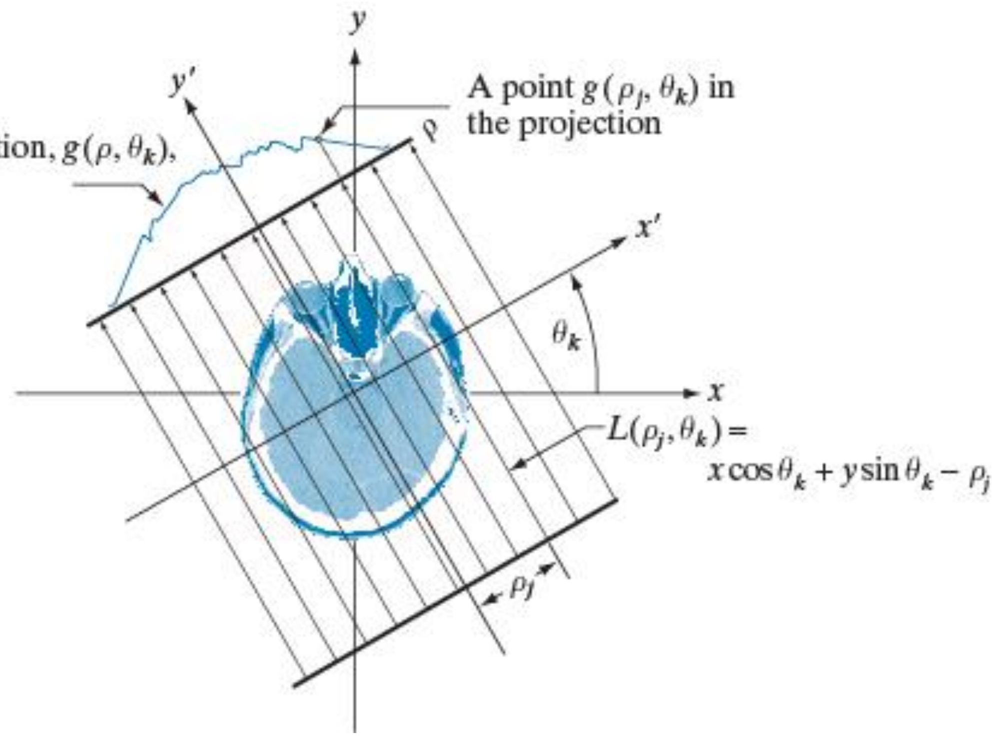


FIGURE 5.37 Geometry of a parallel-ray beam.

Projections and the Radon Transform

- **Radon transform** gives the projection (line integral) of $f(x, y)$ along an arbitrary line in the xy-plane

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

- Discrete Version

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

Example: Use Radon transform to obtain the projection of a Circular region

- Assume that the circle is centered on the origin of the *xy*-plane.
- Because the object is circularly symmetric, its projections are the same for all angles, so we just check the projection for $\theta = 0^{\circ}$

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

Example: Use Radon transform to obtain the projection of a circular region

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

Example: Using the Radon transform to obtain the projection of a circular region

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

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

$$=\int_{-\infty}^{\infty} f(\rho,y)dy$$

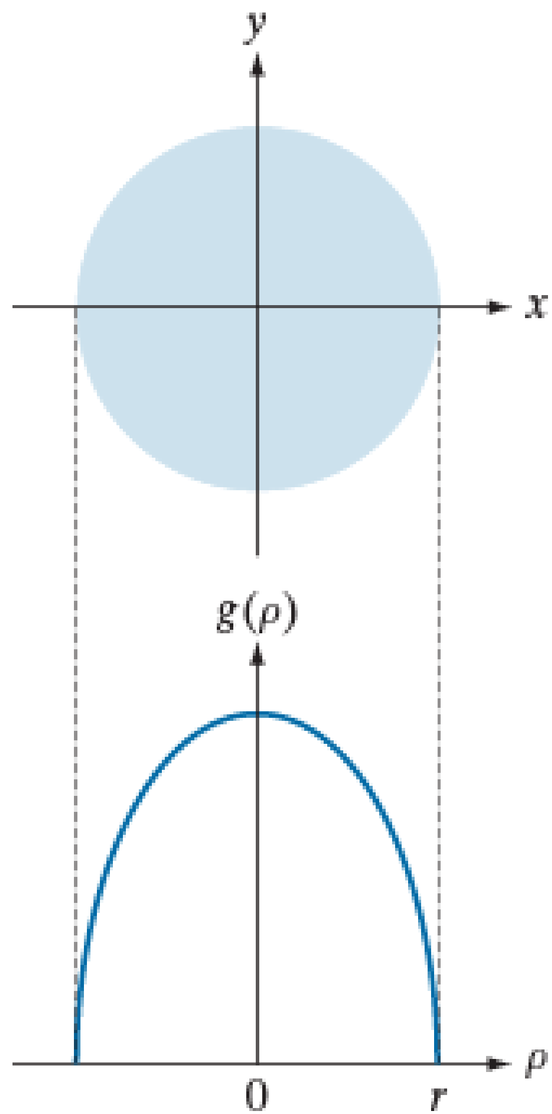
$$=\int_{-\sqrt{r^2-\rho^2}}^{\sqrt{r^2-\rho^2}} f(\rho,y)dy$$

$$=\int_{-\sqrt{r^2-\rho^2}}^{\sqrt{r^2-\rho^2}} A dy$$

$$=\begin{cases} 2A\sqrt{r^2-\rho^2} & |\rho|\le \mathrm{r} \\ 0 & \text{otherwise} \end{cases}$$

Radon Transform (contd..)

$$g(\rho) = \begin{cases} 2A\sqrt{r^2 - \rho^2} & |\rho| \leq \mathrm{r} \\ 0 & \text{otherwise} \end{cases}$$



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.

Sinogram: The Result of Radon Transform

- The representation of the Radon transform $g(\rho,\theta)$ as an image with ρ and θ as coordinates is called a *sinogram*.

- It is very difficult to interpret a sinogram.

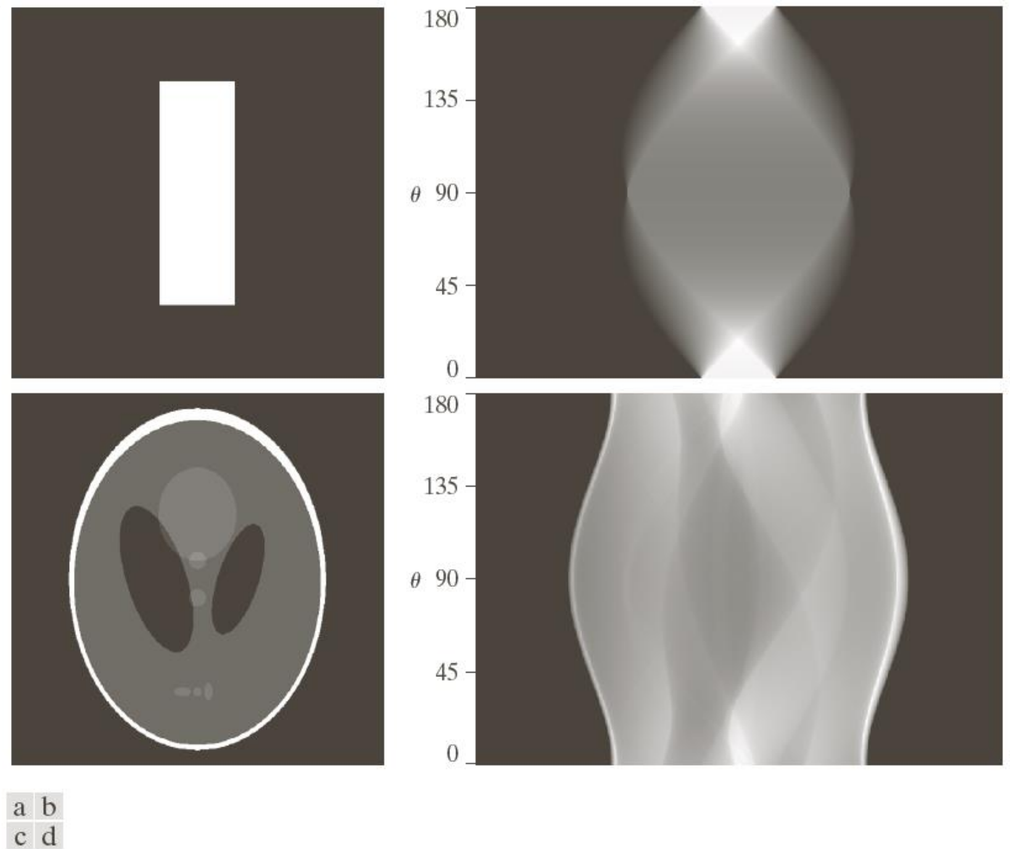


FIGURE 5.39 Two images and their sinograms (Radon transforms). Each row of a sinogram is a projection along the corresponding angle on the vertical axis. Image (c) is called the *Shepp-Logan phantom*. In its original form, the contrast of the phantom is quite low. It is shown enhanced here to facilitate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

The Radon Transform (cont…)

Why is this representation called a *sinogram*?

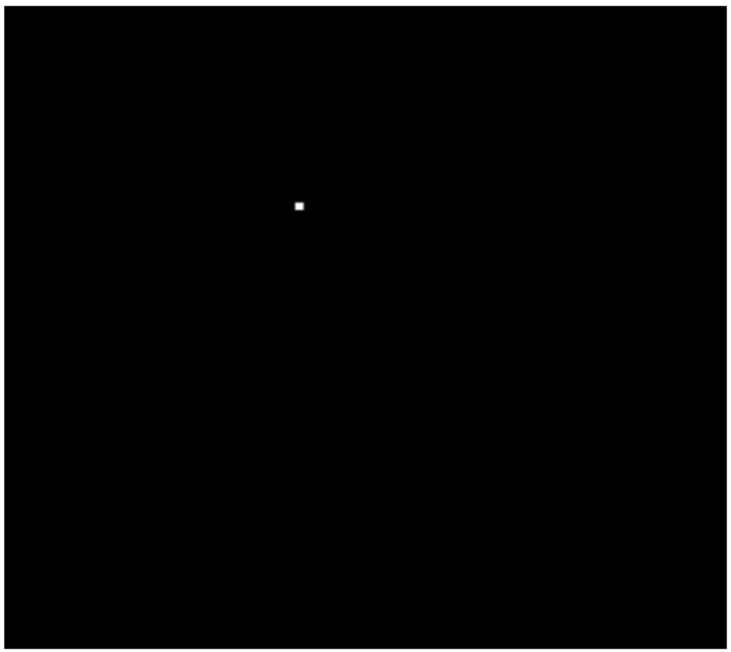
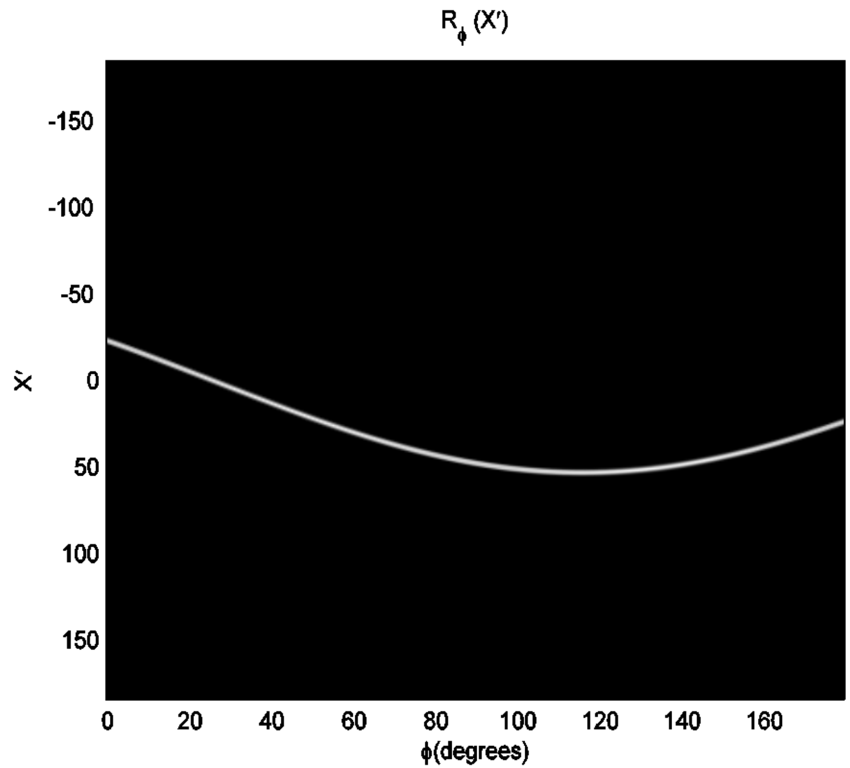


Image of a single point



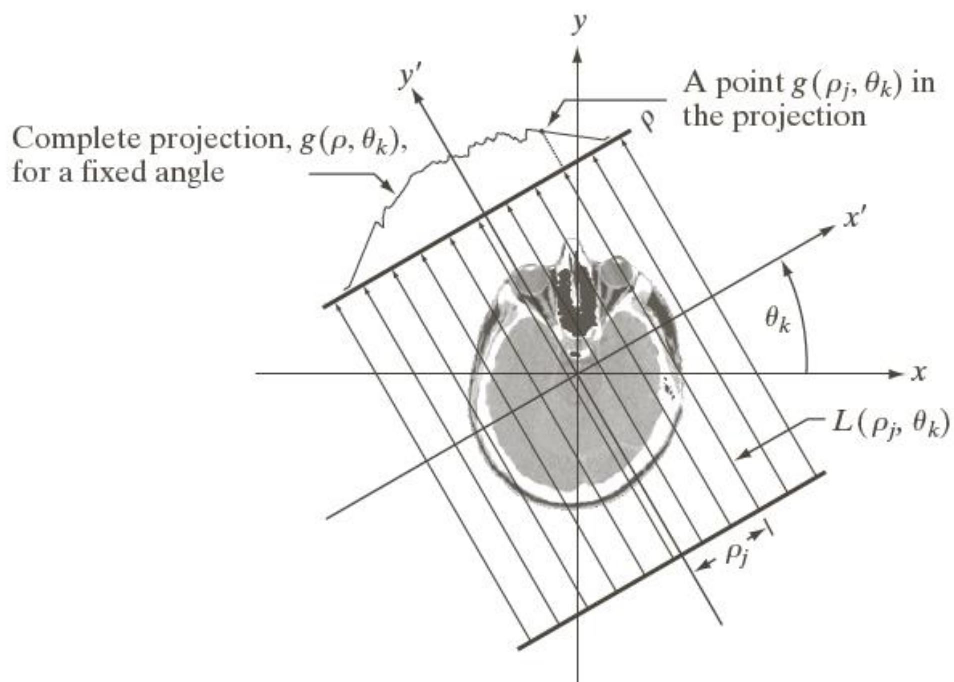
The Radon transform

The Radon Transform (cont...)

- The objective of CT is to obtain a 3D representation of a volume from its projections.
- The approach is to back-project each projection and sum all the back-projections to generate a slice.
- Stacking all the slices produces a 3D volume.
- We will now describe the back-projection operation mathematically.

The Radon Transform (cont...)

- For a fixed rotation angle θ_k, and a fixed distance ρ_j, back-projecting the value of the projection $g(\rho_j,\theta_k)$ is equivalent to copying the value $g(\rho_j,\theta_k)$ to the image pixels belonging to the line $x\cos\theta_k + y\sin\theta_k = \rho_j$.



The Radon Transform (cont...)

- Repeating the process for all values of ρ_j, having a fixed angle θ_k results in the following expression for the image values:

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

- This equation holds for every angle θ:

$$f_\theta(x,y) = g(\rho,\theta) = g(x\cos\theta + y\sin\theta,\theta)$$

Image Reconstruction

$$f_\theta(x,y) = g(x\cos\theta + y\sin\theta, \theta)$$

$$f(x,y) = \int_0^\pi f_\theta(x,y)\,d\theta$$

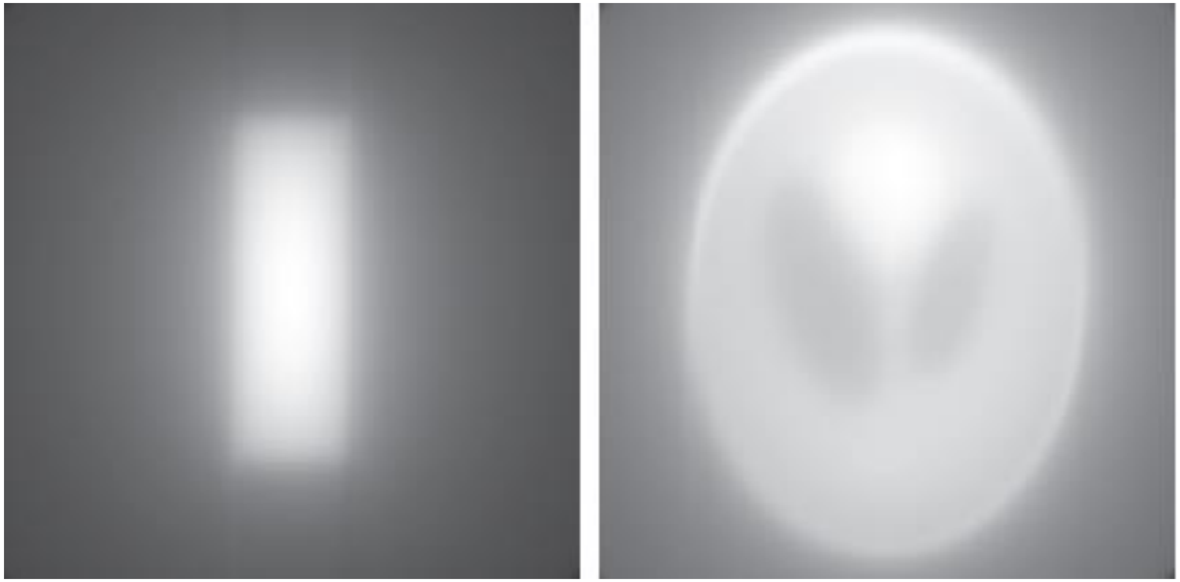
$$f(x,y) = \sum_{\theta=0}^{\pi} f_\theta(x,y)$$

A back-projected image formed is referred to as a laminogram

Examples: Laminogram

a b

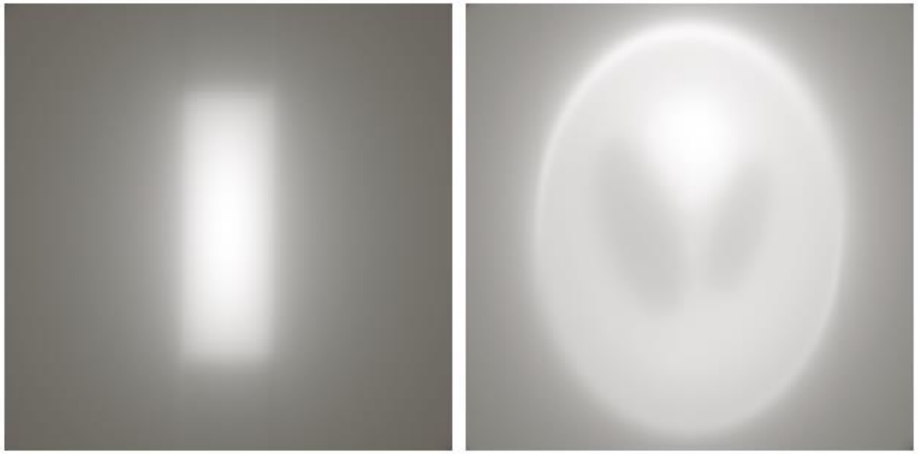
FIGURE 5.40 Backprojections of the sinograms in Fig. 5.39.



The Radon Transform (cont...)

- The final image is formed by integrating over all the back-projected images:

$$f(x,y) = \int_0^{\pi} f_\theta(x,y)\,d\theta$$



- Back-projection provides blurred images.
- We will reformulate the process to eliminate blurring.

The Fourier-Slice Theorem

- The *Fourier-slice theorem* or the *central slice theorem* relates the 1D Fourier transform of a projection with the 2D Fourier transform of the region of the image from which the projection was obtained.

- It is the basis of image reconstruction methods.

The Fourier-Slice Theorem

For a given value of θ, the 1-D Fourier transform of a projection with respect to ρ is

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

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

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

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

The Fourier-Slice Theorem

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

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

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

Illustration of the Fourier-slice theorem

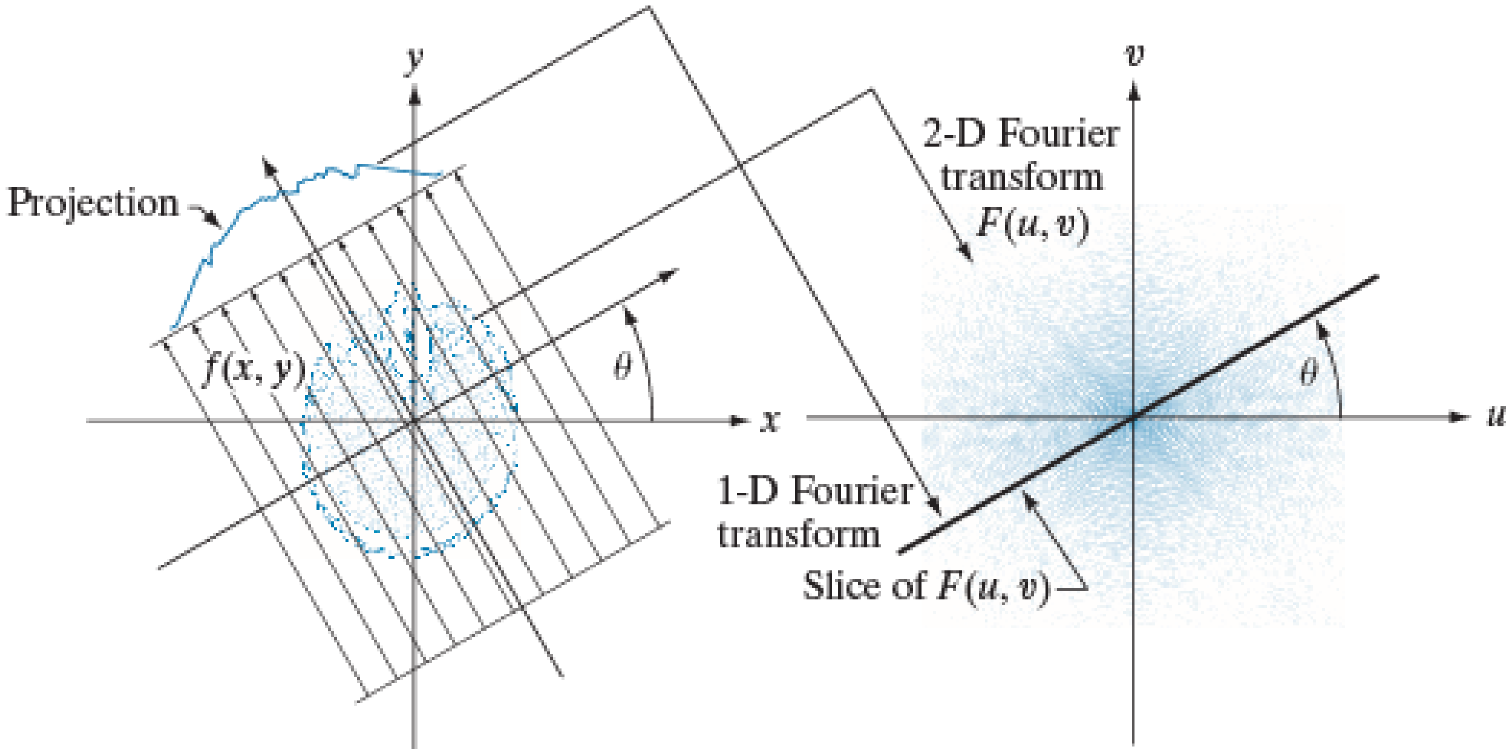


FIGURE 5.41 Illustration of the Fourier-slice theorem. The 1-D Fourier transform of a projection is a slice of the 2-D Fourier transform of the region from which the projection was obtained. Note the correspondence of the angle θ in the two figures.

Reconstruction Using Parallel-Beam Filtered Backprojections

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

Let $u=w\cos\theta, v=w\sin\theta$, then $dudv=wdwd\theta$,

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

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

$$G(w,\theta+180^\circ)=G(-w,\theta)$$

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

Reconstruction Using Parallel-Beam Filtered Backprojections

$$f(x,y)=\int_0^{\pi}\int_{-\infty}^{\infty}|w|\,G(w,\theta)e^{j2\pi w(x\cos\theta+y\sin\theta)}dw\,d\theta$$

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

It's not integrable

Approach:

- Window the ramp so it becomes zero outside of a defined frequency interval
- The window band-limits the ramp filter

Hamming / Hann Widow

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

$c=0.54,$ the function is called the <u>Hamming window</u>

$c=0.5,$ the function is called the <u>Han window</u>

The Plot of Hamming Widow

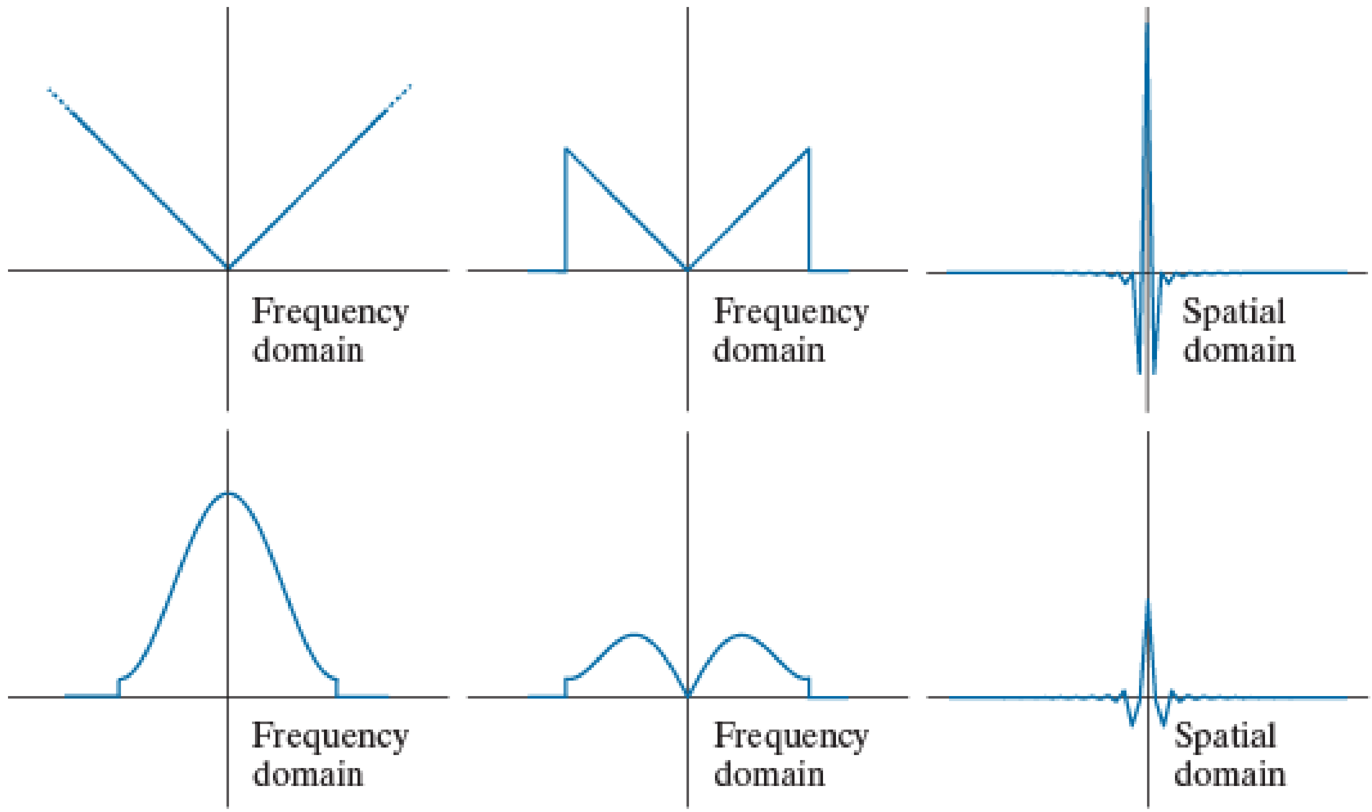


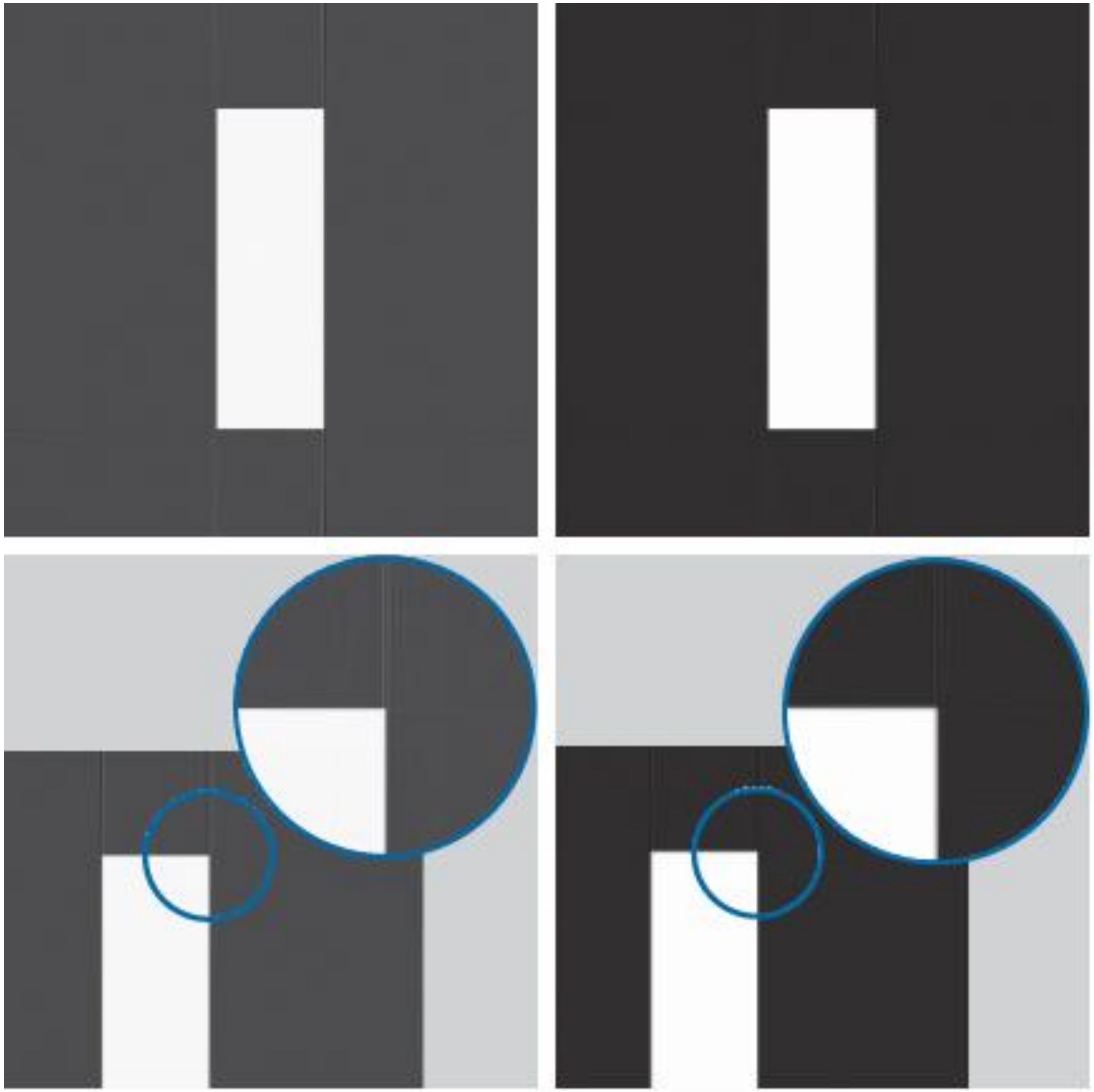
FIGURE 5.42
(a) Frequency domain ramp filter transfer function. (b) Function after band-limiting it with a box filter. (c) Spatial domain representation. (d) Hamming windowing function. (e) Windowed ramp filter, formed as the product of (b) and (d). (f) Spatial representation of the product. (Note the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

Filtered Backprojection

The complete, filtered backprojection (to obtain the reconstructed image $f(x,y)$) is described as follows:

1. Compute the 1-D Fourier transform of each projection
2. Multiply each Fourier transform by the filter function $|w|$ which has been multiplied by a suitable window (e.g., Hamming)
3. Obtain the inverse 1-D Fourier transform of each resulting filtered transform
4. Integrate (sum) all the 1-D inverse transforms from step 3

Examples: Filtered Backprojection



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).

Examples: Filtered Backprojection

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)

Implementation of Filtered Backprojection in Spatial Domain

- Fourier transform of the product of two frequency domain functions is equal to the convolution of the spatial representation
- Let s(p) denote the inverse Fourier transform of |w|

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

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

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

Reconstruction Using Fan-Beam Filtered Backprojections

$$\theta = \alpha + \beta$$

$$\rho = D \sin \alpha$$

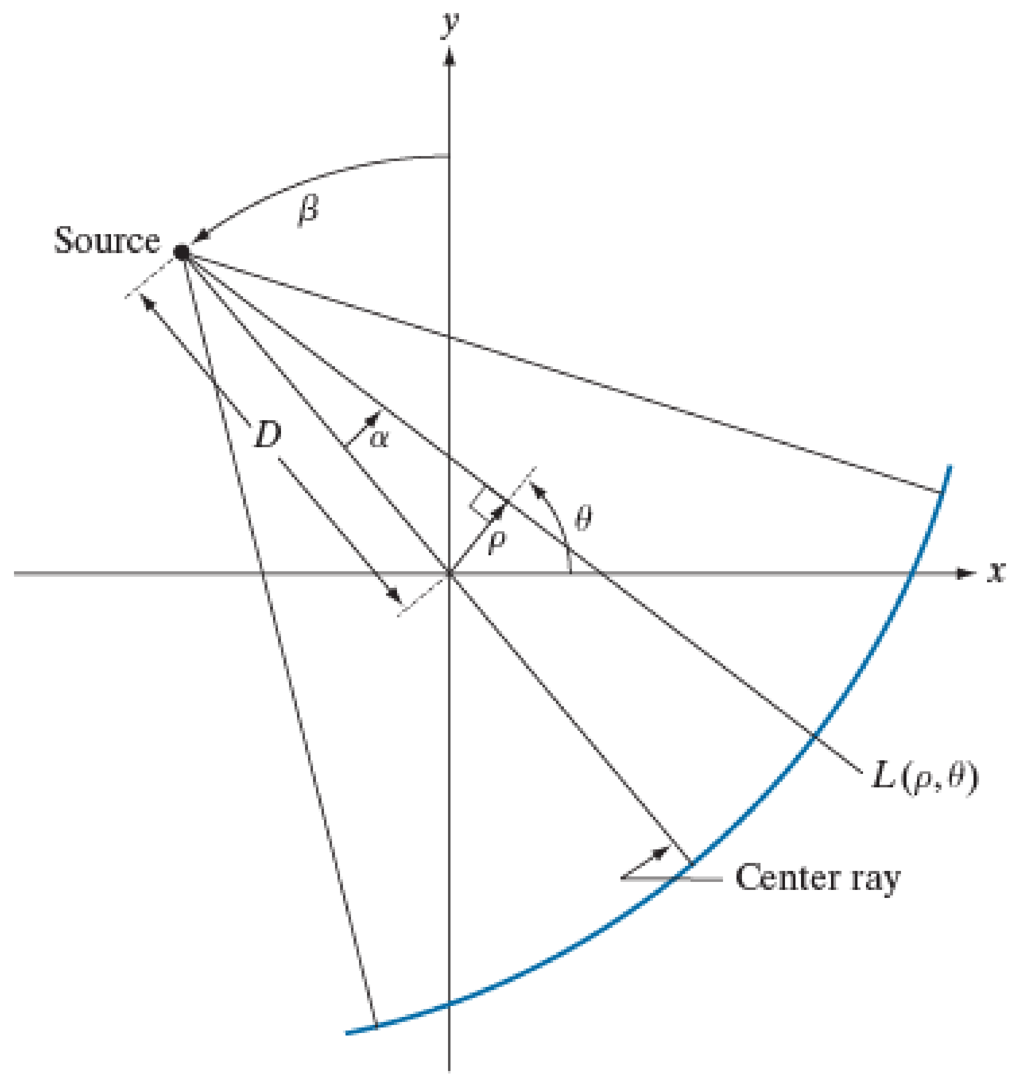


FIGURE 5.45 Basic fan-beam geometry. The line passing through the center of the source and the origin (assumed here to be the center of rotation of the source) is called the *center ray*.

Reconstruction Using Fan-Beam Filtered Backprojections

Objects are encompassed within a circular area of radius T about the origin of the plane, or g(ρ,θ)=0 for |ρ|>T

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

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

$$x=r\cos\varphi;\, y=r\sin\varphi$$

$$x\cos\theta+y\sin\theta=r\cos\varphi\cos\theta+r\sin\varphi\sin\theta$$

$$=r\cos(\varphi-\theta)$$

Reconstruction Using Fan-Beam Filtered Backprojections

$$x = r\cos\varphi; y = r\sin\varphi$$

$$x\cos\theta + y\sin\theta = r\cos\varphi\cos\theta + r\sin\varphi\sin\theta$$

$$= r\cos(\varphi - \theta)$$

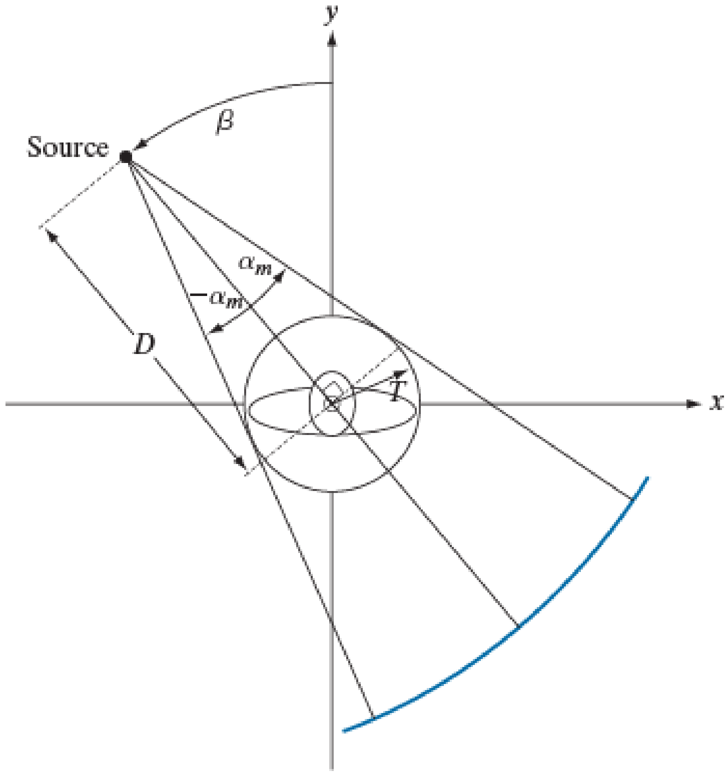
$$f(x,y) = \frac{1}{2}\int_0^{2\pi}\int_{-T}^{T} g(\rho,\theta)s\left[r\cos(\varphi-\theta)-\rho\right]d\rho d\theta$$

$$\theta = \alpha + \beta \qquad\qquad \rho = D\sin\alpha$$

$$d\rho d\theta = D\cos\alpha d\alpha d\beta$$

Reconstruction Using Fan-Beam Filtered Backprojections

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



$$d\rho d\theta = D\cos\alpha\, d\alpha\, d\beta$$

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

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

Reconstruction Using Fan-Beam Filtered Backprojections

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

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

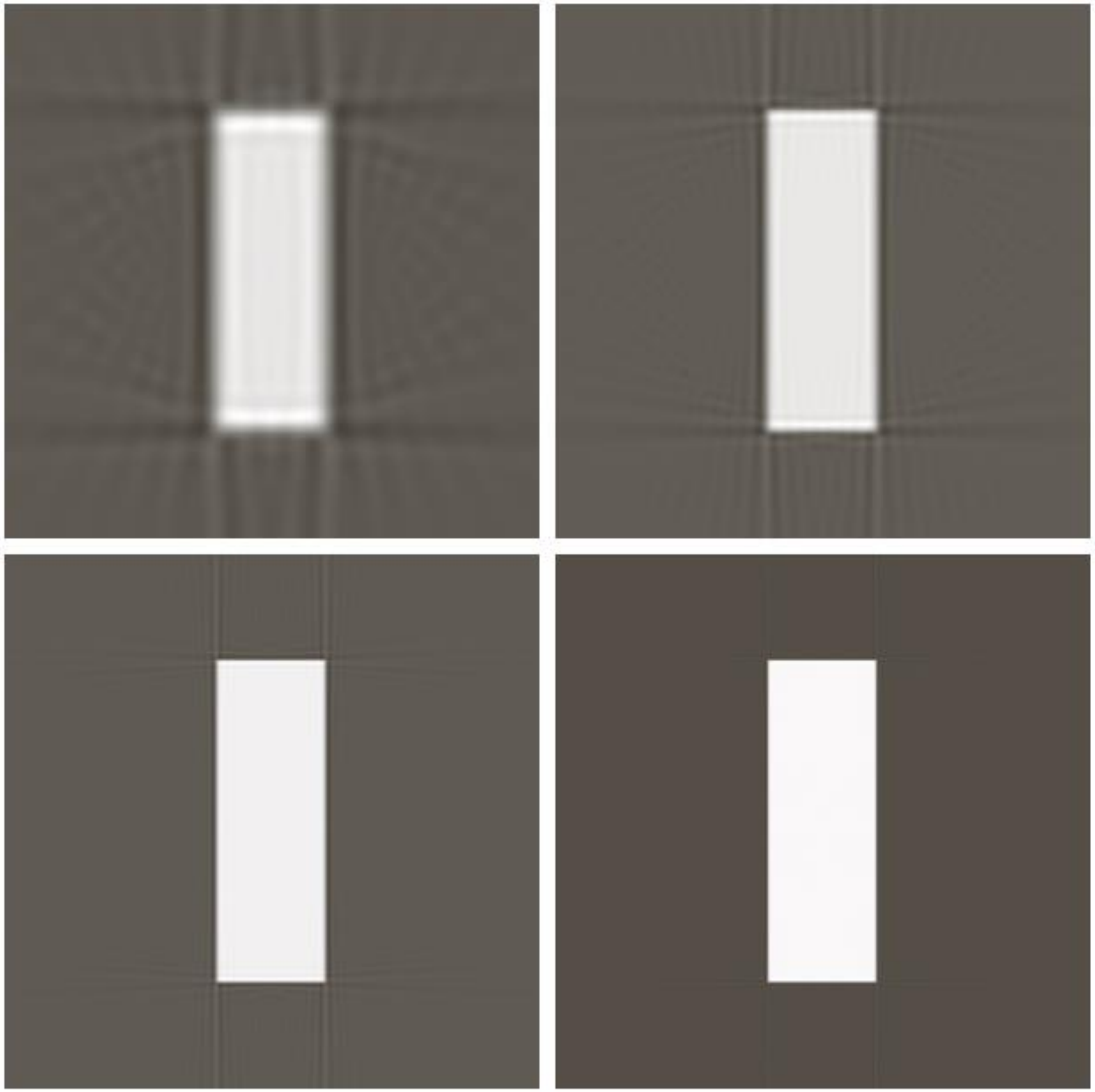
$$f(r,\varphi)=\frac{1}{2}\int_0^{2\pi}\int_{-\alpha_m}^{\alpha_m} p(\alpha,\beta)s\left[R\sin\left(\alpha'-\alpha\right)\right]D\cos\alpha\, d\alpha\, d\beta$$

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

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

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

Reconstruction Using Fan-Beam Filtered Backprojections (Contd..)

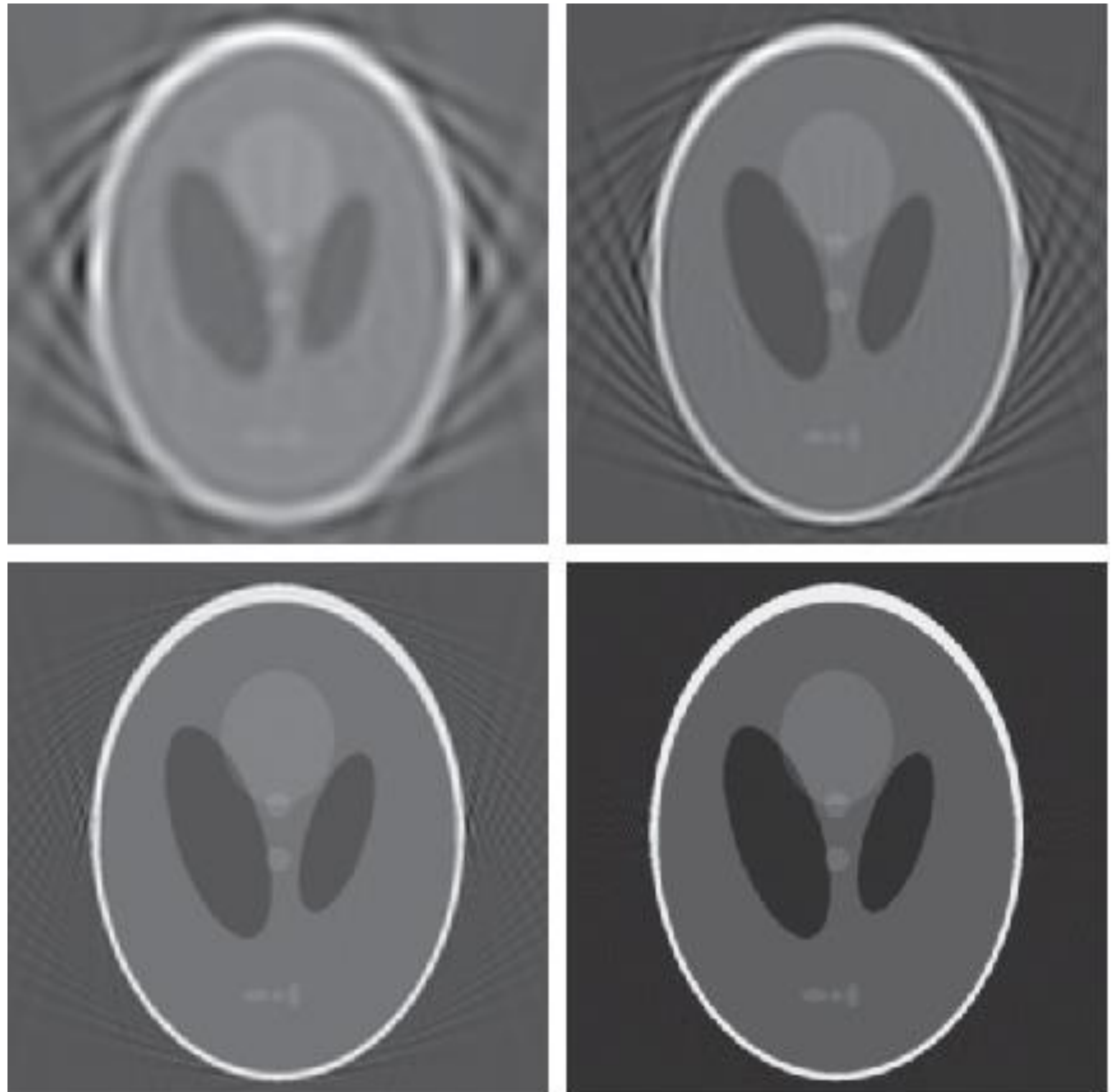


a b
c d

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).

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).



Acknowlegements

The slides are primarily based on the figures and images in the Digital Image Processing textbook by Gonzalez and Woods:

- http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

In addition, slides have been adopted and modified from the following sources:

- http://www.cs.uoi.gr/~cnikou/Courses/Digital_Image_Processing
- http://www.comp.dit.ie/bmacnamee/gaip.htm
- http://baggins.nottingham.edu.my/~hsooihock/G52IIP/
- http://gear.kku.ac.th/~nawapak/178353.html
- https://cs.nmt.edu/~ip/index.html