EIE4512 - Digital Image Processing
Image Restoration & Reconstruction

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

Zhen Li
lizhen@cuhk.edu.cn
School of Science and Engineering
The Chinese University of Hong Kong, Shen Zhen
February 26-28, 2019

Restoration vs. Reconstruction

- *<u>Image restoration</u>* and *<u>image enhancement</u>* share a common goal: to improve image for human perception
- Image enhancement is mainly a subjective process in which individuals' opinions are involved in process design.
  - For instance: Image sharpening

a b

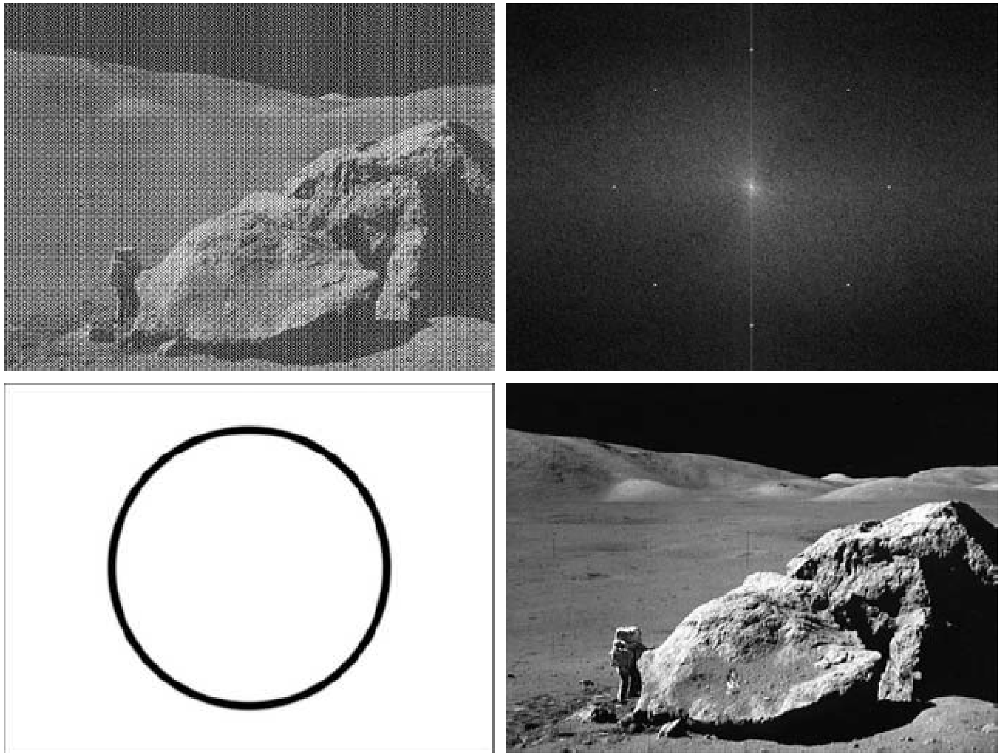
FIGURE 4.58 (a) Original, blurry image. (b) Image enhanced using the Laplacian in the frequency domain. Compare with Fig. 3.38(e).

Restoration vs. Reconstruction

- Image restoration is mostly an objective process which
  - utilizes a prior knowledge of degradation phenomenon to recover image.
  - models the degradation and then to recover the original image.
    - For instance: Image denoising

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.)



Model of Degradation/Restoration Process

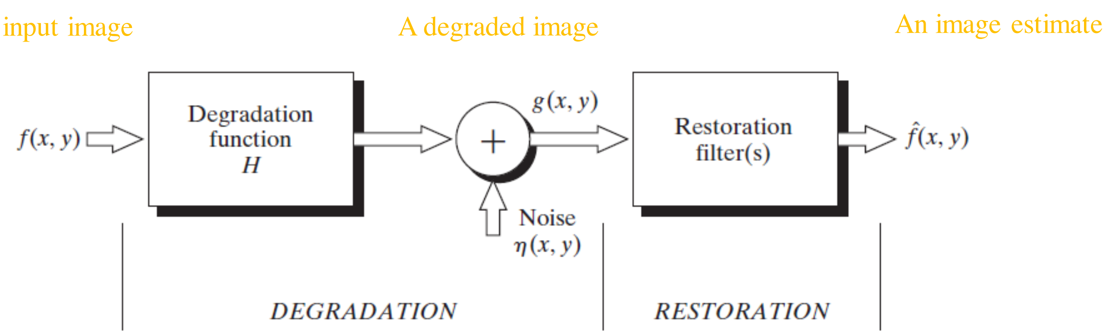


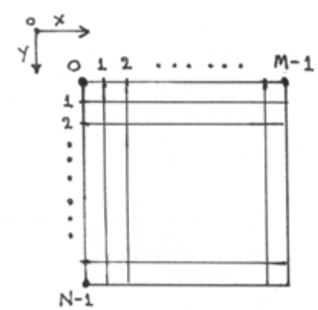
FIGURE 5.1 A model of the image degradation/restoration process.

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

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

- whose equivalent frequency domain representation is

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



The frame of reference

Objective of Restoration

- The objective of restoration is to obtain an image estimate which is as close as possible to the original input image.
- A typical difference measurement is the *mean square error (MSE)*:

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

- Generally, the more H and noise are known, the lower MSE will become.

The Source of Noises

The principal sources of noise in digital images arise during:

- Image acquisition
  - For instance, with a CCD camera, light levels and sensor temperature introduce noise to the resulting image.
- Image transmission
  - For instance, an image transmitted over a wireless network might be corrupted as a result of lighting or other atmospheric disturbance.

Spatial & Frequency Properties

- Spatial properties
  - We assume the noises are independent of spatial locations
  - We assume there is no correlation between pixel values and the noise components

- Frequency properties
  - The frequency content of noise in Fourier sense
  - If the Fourier spectrum is constant, the noise is usually called <u>**white noise**</u>.

Noise Distributions

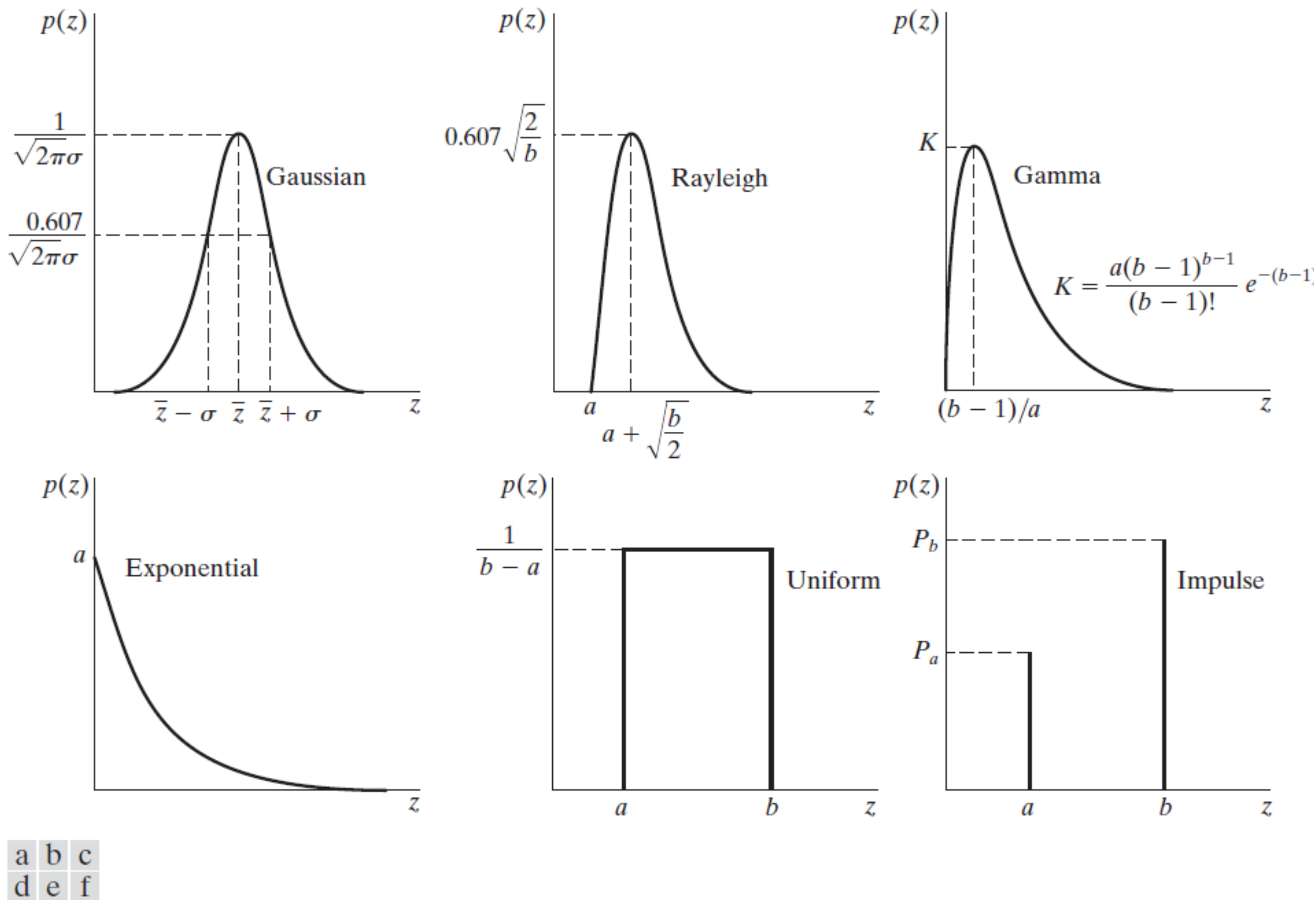
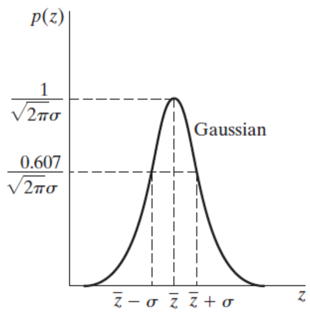


FIGURE 5.2 Some important probability density functions.

Gaussian Noise

Gaussian (normal) noise



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

z represents intensity

$\bar{z}$ is the mean of z

σ is the standard deviation of z

σ^2 is the variance of z

–frequently used in practice since it is mathematically tractable in both the spatial and frequency domains

–70% of z's values fall into the range $[(\bar{z}-\sigma),(\bar{z}+\sigma)]$

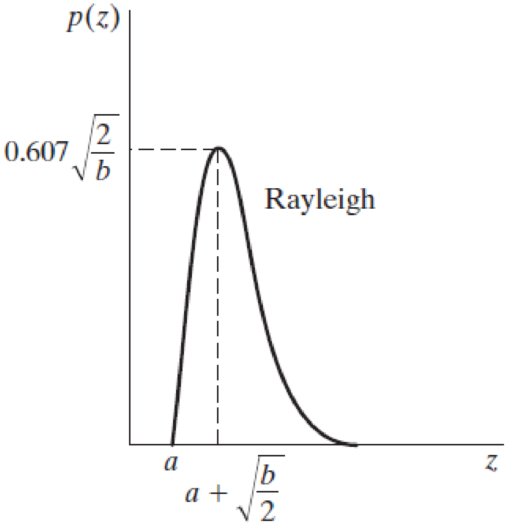
–95% of z's values fall into the range $[(\bar{z}-2\sigma),(\bar{z}+2\sigma)]$

–arising in an image due to factors such as electronic circuit noise and sensor noise due to poor illumination and/or high temperature

–Central limit theorem

Rayleigh Noise

Raleigh noise



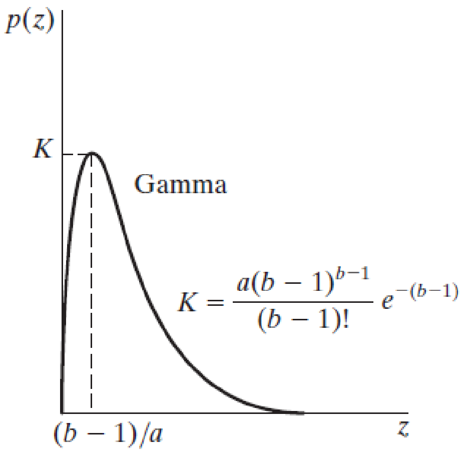
$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a \end{cases} \qquad \bar{z} = a + \sqrt{\pi b/4} \qquad \sigma^2 = \frac{b(4-\pi)}{4}$$

–Displacement from origin; and skewed to the right; useful for approximating skewed histograms

–characterizing noise phenomena in range imaging

Erlang (Gamma) Noise

Erlang (Gamma) noise



$$p(z) = \begin{cases} \dfrac{a^b z^{b-1}}{(b-1)!} e^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases} \qquad \bar{z} = \frac{b}{a}, \qquad \sigma^2 = \frac{b}{a^2} \qquad a \in R^+, b \in Z^+$$

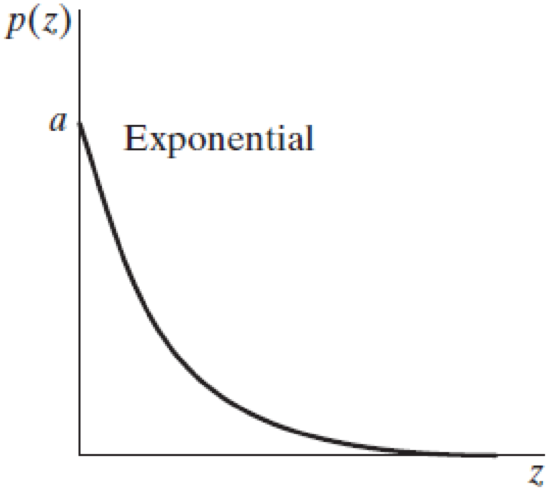
–developed by Erlang to model telephone traffics

–called Gamma noise if the denominator is the gamma function, $\Gamma(b)$

–useful in laser imaging

Exponential Noise

Exponential noise

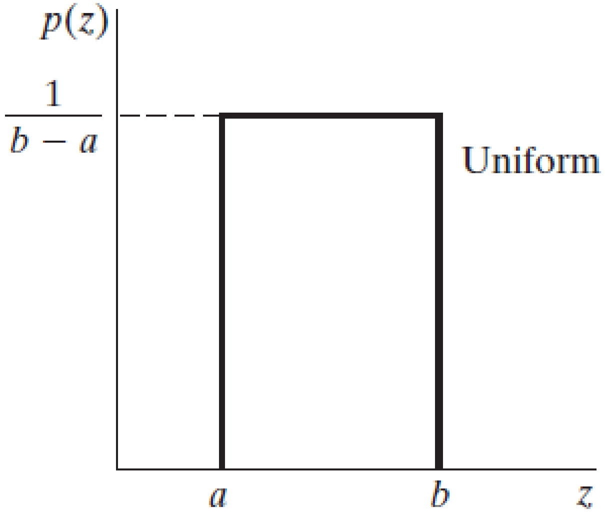


$$p(z) = \begin{cases} ae^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases} \qquad \bar{z} = \frac{1}{a} \qquad \sigma^2 = \frac{1}{a^2}$$

– a special case of the Erlang density, with b=1

Uniform Noise

Uniform noise



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

$$\bar{z} = \frac{a+b}{2}$$

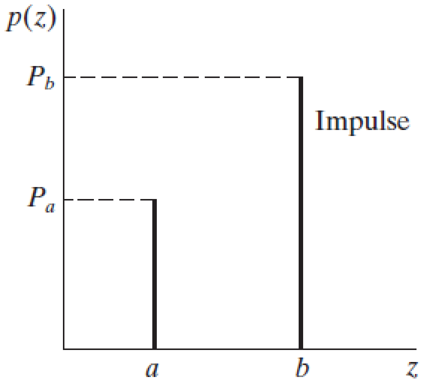
$$\sigma^2 = \frac{(b-a)^2}{12}$$

–each noise intensity being equally probable

Impulse Noise

<u>Impulse</u> (salt-and-pepper) noise

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



–bipolar if neither P_a or P_b is zero; in practice, for an 8-bit image, b=255 (white) and a = 0 (black)
–bipolar one, also known as salt-and-pepper, data-drop-out and spike noise
–called unipolar if either P_a or P_b is zero
–caused by either sensors' <u>failure to respond</u> (**pepper**, *black*) or sensors' <u>saturation</u> in color (**salt**, *white*)

Test Pattern

Figure 5.3 shows a test pattern well suited for illustrating the noise models just discussed,

- composed of simple constant areas that span the gray scale from black to near white in only three increments
- facilitating visual analysis of the characteristics of the various noise components added to the image

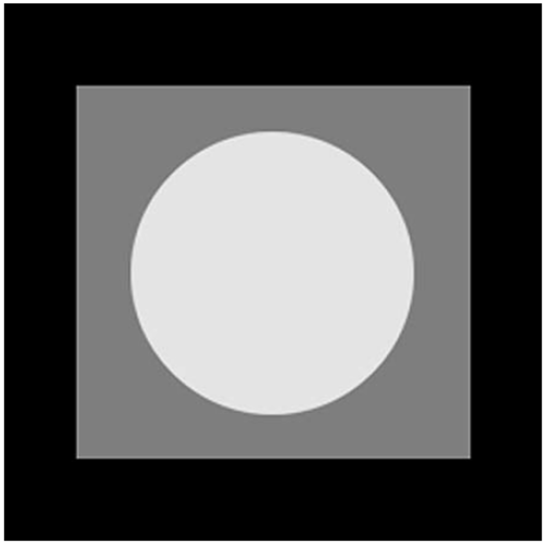


FIGURE 5.3 Test pattern used to illustrate the characteristics of the noise PDFs shown in Fig. 5.2.

Noise Examples

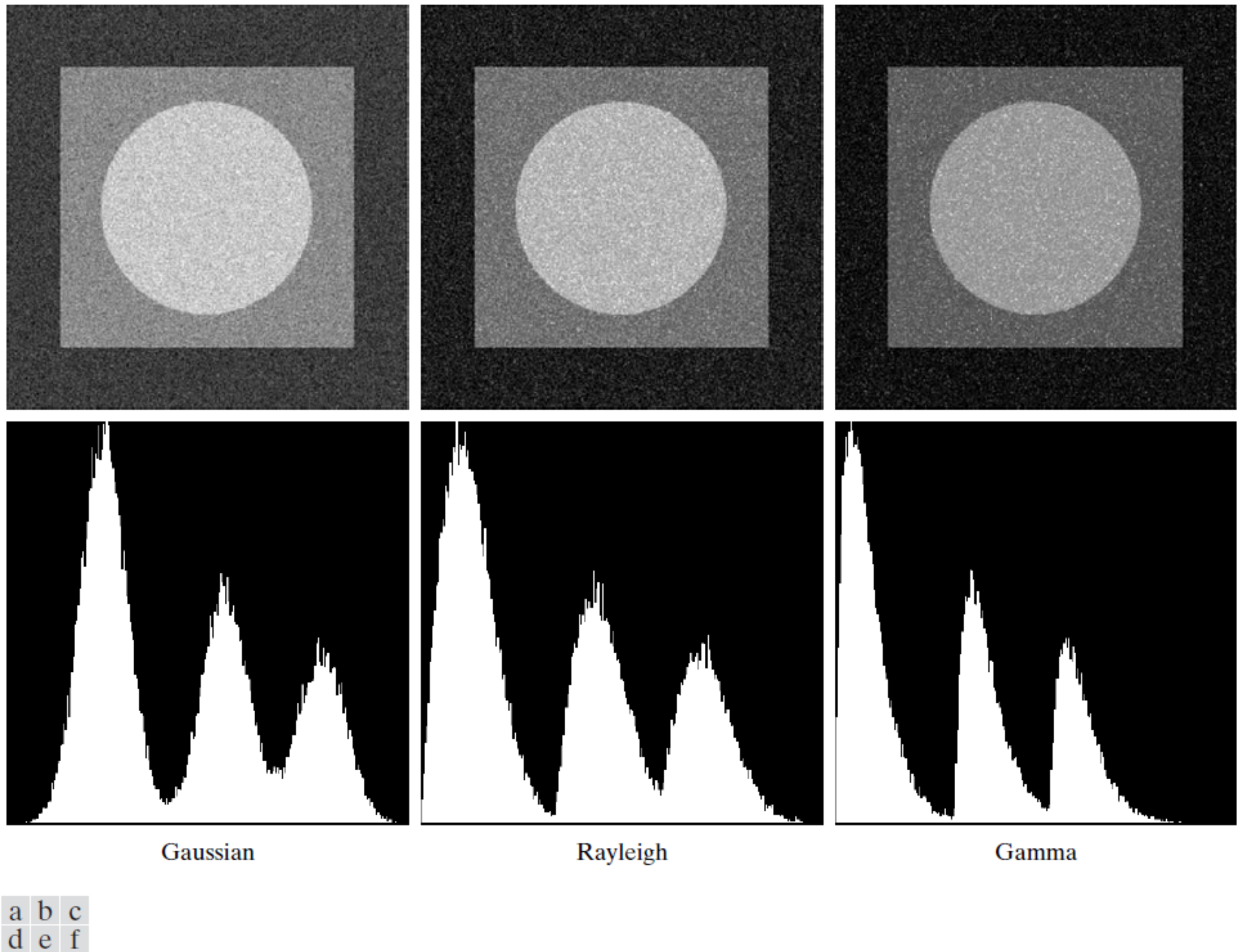


FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and gamma noise to the image in Fig. 5.3.

Noise Examples

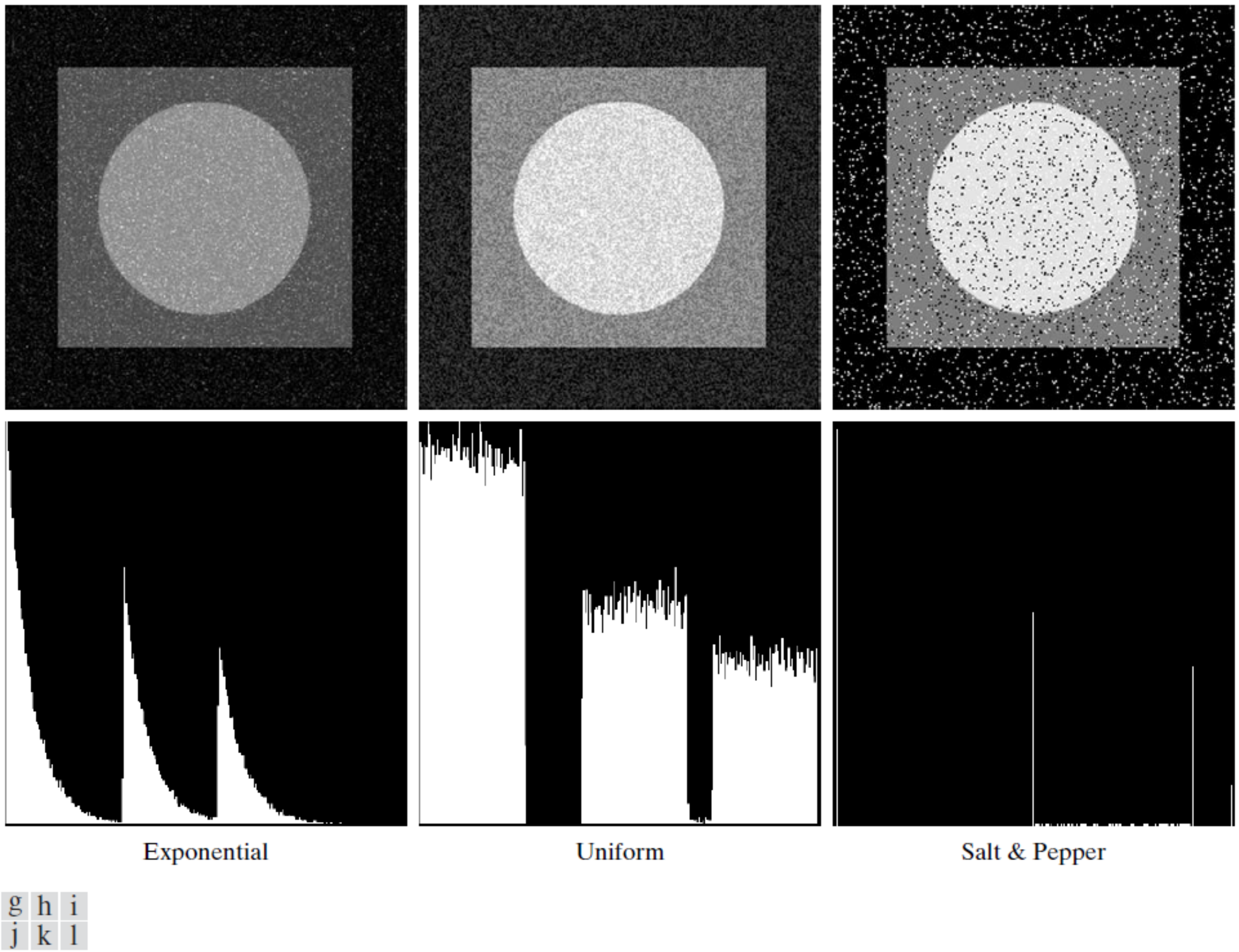


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt-and-pepper noise to the image in Fig. 5.3.

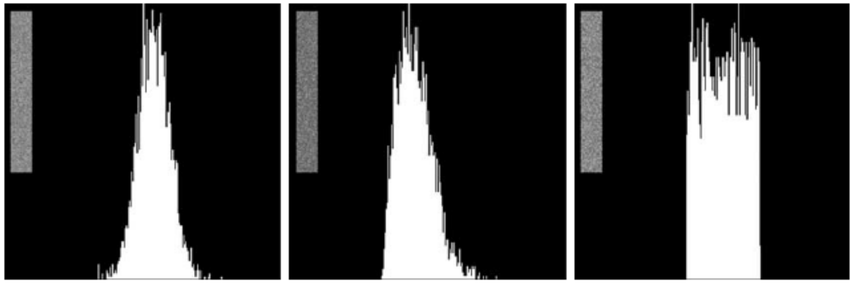
The salt-and pepper appearance of the image corrupted by impulse noise is the only one that is visually indicative of the type of noise causing the degradation

Estimation of Noise Parameters

- The parameters of noise PDFs
  maybe known partially from sensor specifications
  often necessary to be estimated for a particular imaging arrangement

  - capturing a set of images of "flat" environments

- Possible to be estimated from small patches of *reasonably constant background intensity*, when only images already generated by a sensor are available
  - e.g., the vertical strips of 150x20 pixels



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.

Noise-only Degradation

Spatial domain model

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

Frequency domain model

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

We often use <u>spatial filtering</u> for image restoration when <u>only additive noise is present</u>.

Filters for Noise Removal

- Mean Filters
  - Arithmetic mean filter
  - Geometric mean filter
  - Harmonic mean filter
  - Contraharmonic mean filter

- Order-Statistic Filters
  - Median filter
  - Max and min filter
  - Midpoint filter
  - Alpha-trimmed mean filter

- Adaptive Filters
  - Adaptive, local noise reduction filter
  - Adaptive median filter

Mean Filters

Arithmetic mean filter

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

S_{xy}, so-called a filter window, represents a rectangular sub-image of size $m \times n$, centered at (x,y)

- the simplest mean filters
- representing the restored pixel value at (x,y) by the arithmetic mean computed within the filter window
- smoothing local variations in an image → blurring
- noise-reducing as a by-product of blurring

Mean Filters (cont'd)

Geometric mean filter

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

• each restored pixel value given by the product of all the pixel values in the filter window, raised to the power 1/mn

• achieving smoothing comparable to the arithmetic mean filter, but tending to lose less image detail in the process

Harmonic mean filter

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

• working well for salt and Gaussian noises

• but failing for pepper noise

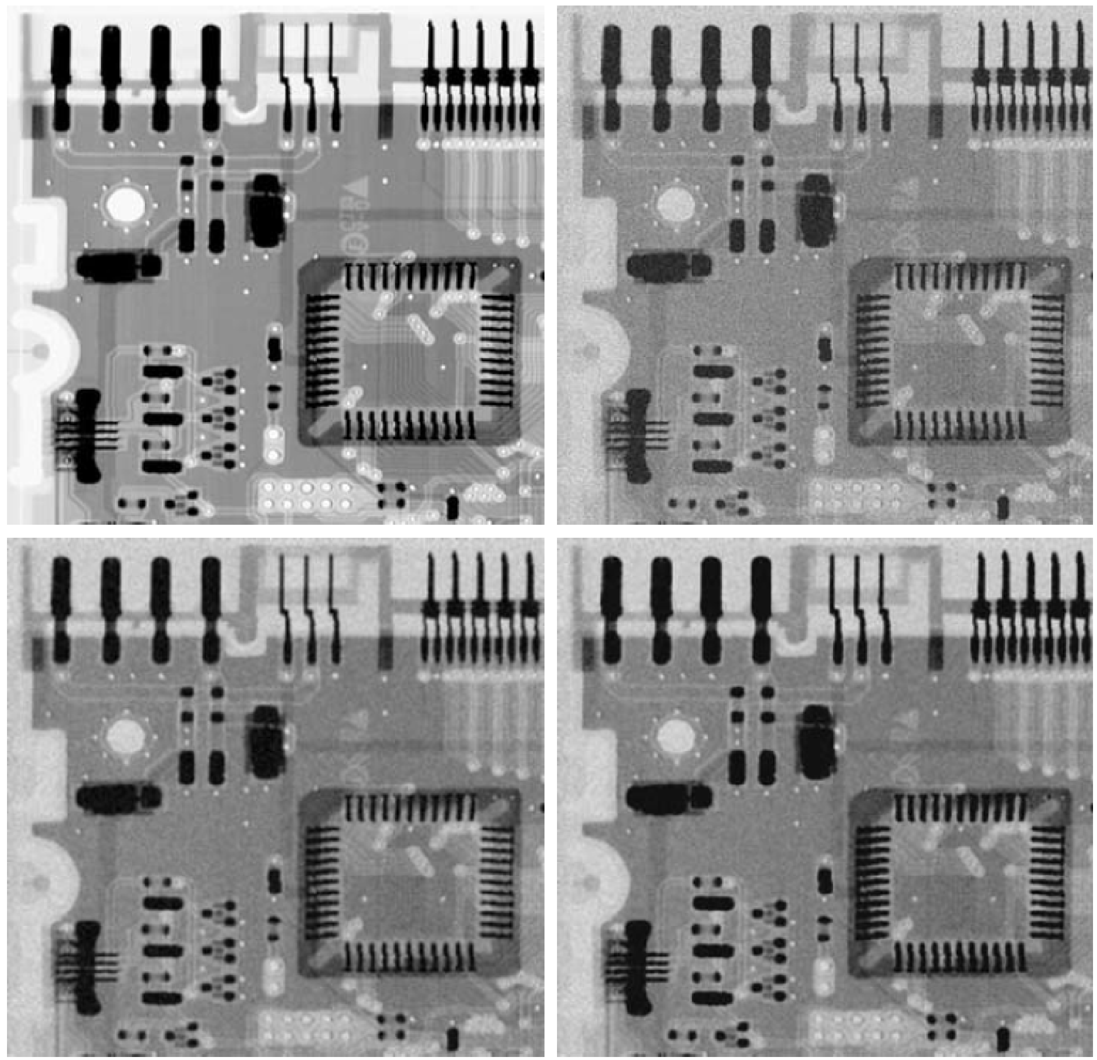
Mean Filters (cont'd)

<u>Contraharmonic mean filter</u>

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

- Q called the order of the filter and $Q \in R$
- well handling or virtually eliminating the effects of salt-and-pepper noise.
- however, unable to eliminate both salt and pepper noises simultaneously
    - eliminating <u>pepper noise</u> when $Q \in R^{+}$
    - eliminating <u>salt noise</u> when $Q \in R^{-}$

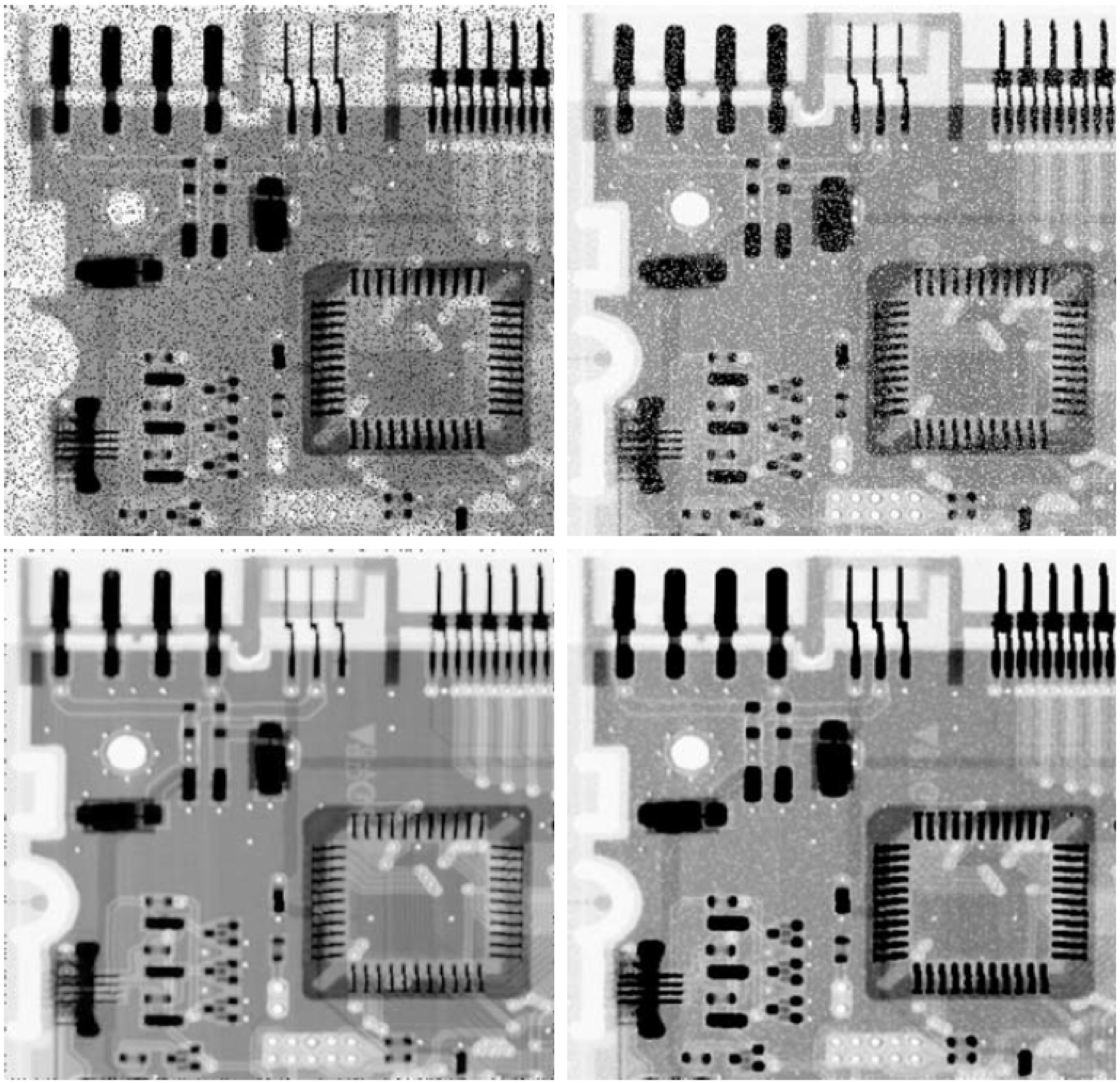
Mean Filters: Examples



a b
c d

FIGURE 5.7 (a) X-ray image. (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.)

Mean Filters: Examples



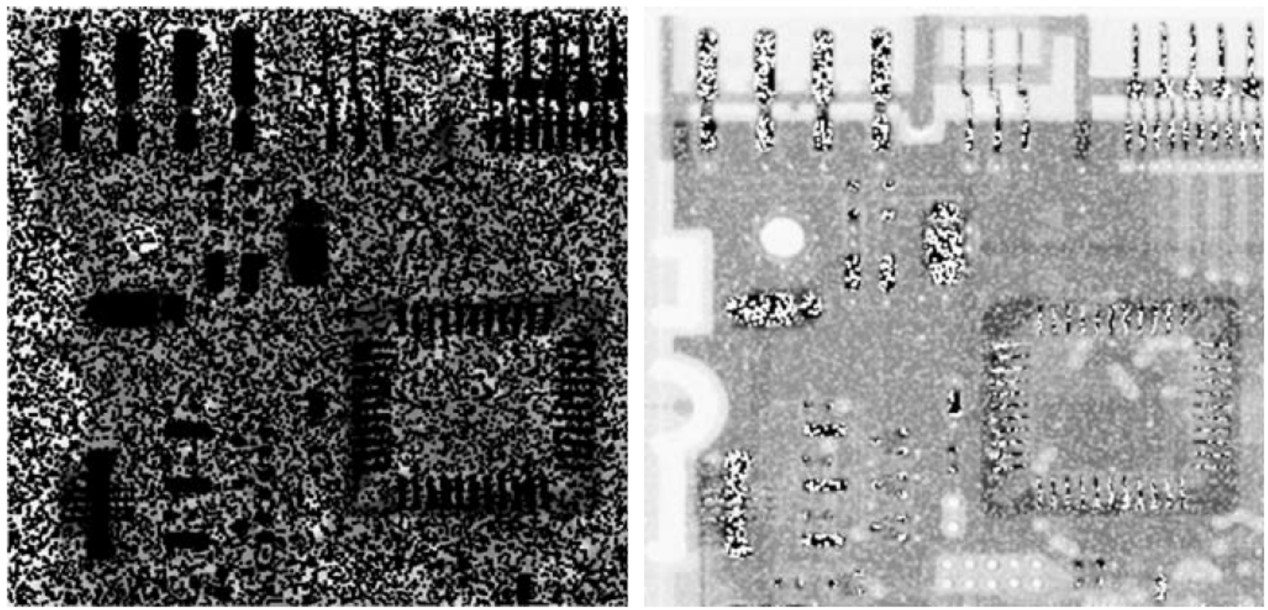
a	b
c	d

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 of order 1.5. (d) Result of filtering (b) with $Q = -1.5$.

Mean Filters: Examples

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 5.8(b) with $Q = 1.5$.



Wrong Contraharmonic filter effect

Order-Statistic Filters (OSF)

- Order-statistic filters are spatial filters whose response is based on ordering (ranking) the pixel values contained in certain neighborhoods.

Order-Statistic Filters

Median filter

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

• the best-known of order-statistic filters
• representing the restored pixel value at (x,y) by the median (ranked in the 50th percentile) of intensity levels in the filter window
• for certain types of noise, providing excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters on the same basis (of similar size)
• particularly effective in the presence of both bipolar and unipolar impulse noise

Order-Statistic Filters

Max and min filters

$$\hat{f}(x,y) = \begin{cases} \max\limits_{(s,t)\in S_{xy}} \{g(s,t)\} & \text{for the max filter} \\ \min\limits_{(s,t)\in S_{xy}} \{g(s,t)\} & \text{for the min filter} \end{cases}$$

• representing the restored pixel value at (x,y) by the maximum/minimum of intensity levels in the filter window
• the max filter greatly reducing pepper noise (black dots)
• the min filter greatly reducing salt noise (white dots)

Order-Statistic Filters

Midpoint filter

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

• representing the restored pixel value at (x,y) by the midpoint between the darkest and brightest points in the filter window
• working best for randomly distributed noise, e.g., Gaussian or uniform noise

Alpha-trimmed mean filter

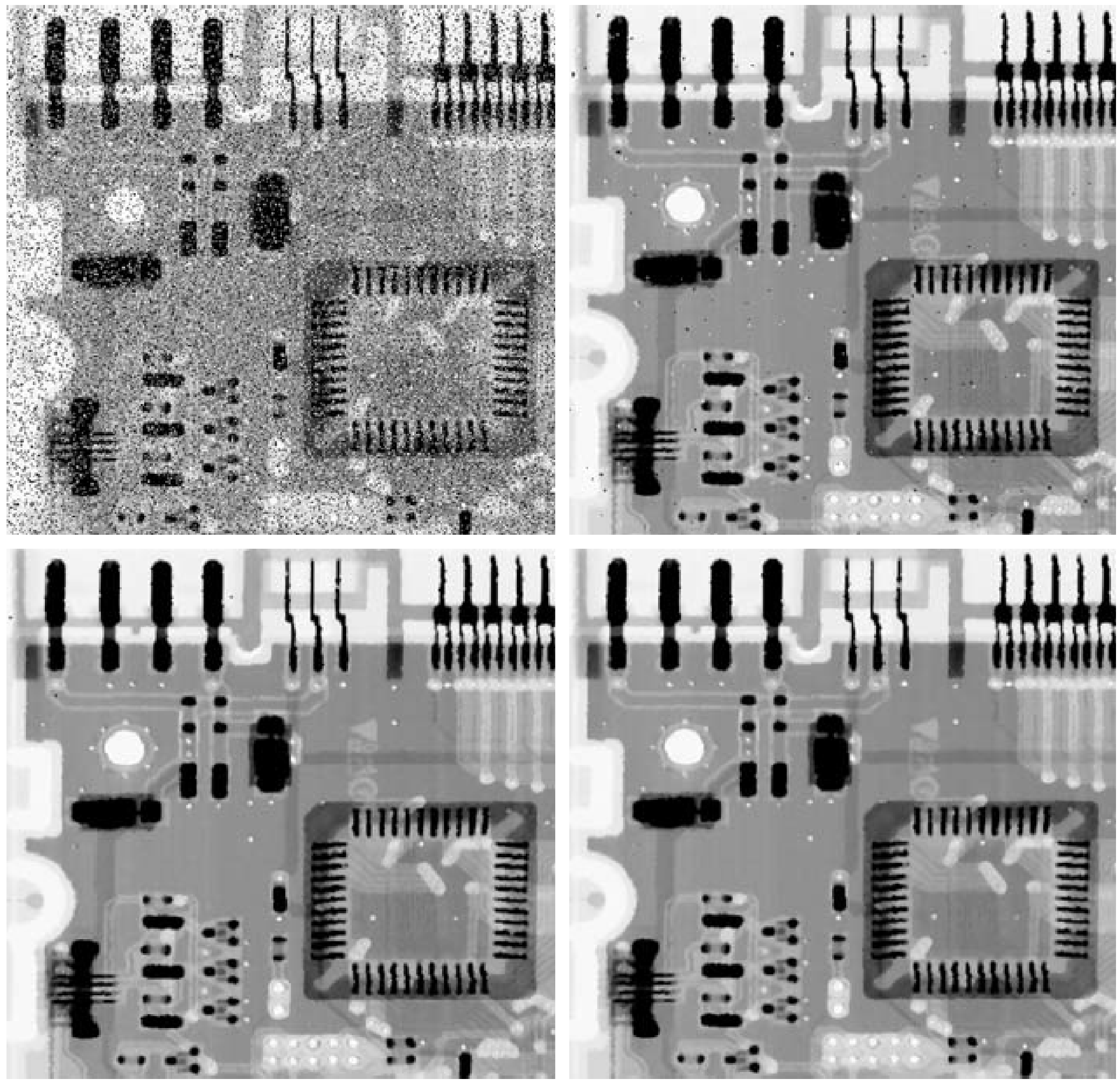
$$\hat{f}(x,y)=\frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} g_r(s,t)$$

• $g_r(x,y)$ representing the trimmed filter window of size $mn-d$ after deleting the d/2 lowest and the d/2 highest values out of the original filter window
• becoming a median filter when $d=mn-1$
• efficiently handling mixture noise, e.g., a combination of salt-and-pepper and Gaussian noise

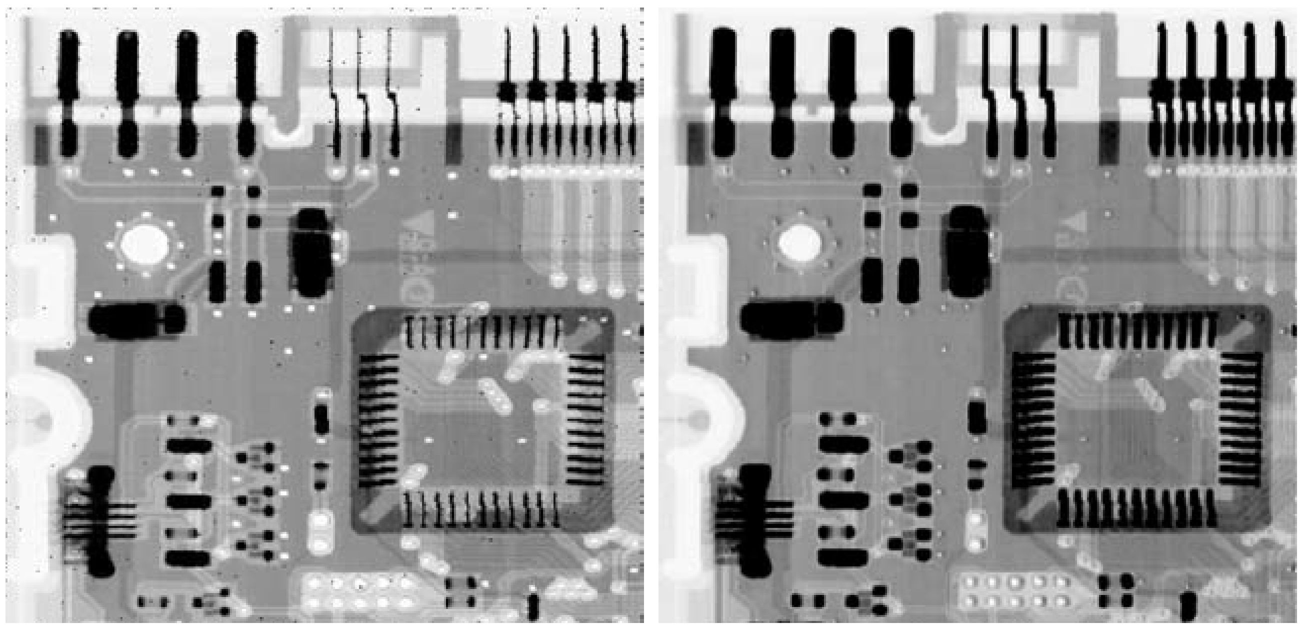
Order-Statistic Filters: Examples

a b
c d

FIGURE 5.10
(a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.1$.
(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.



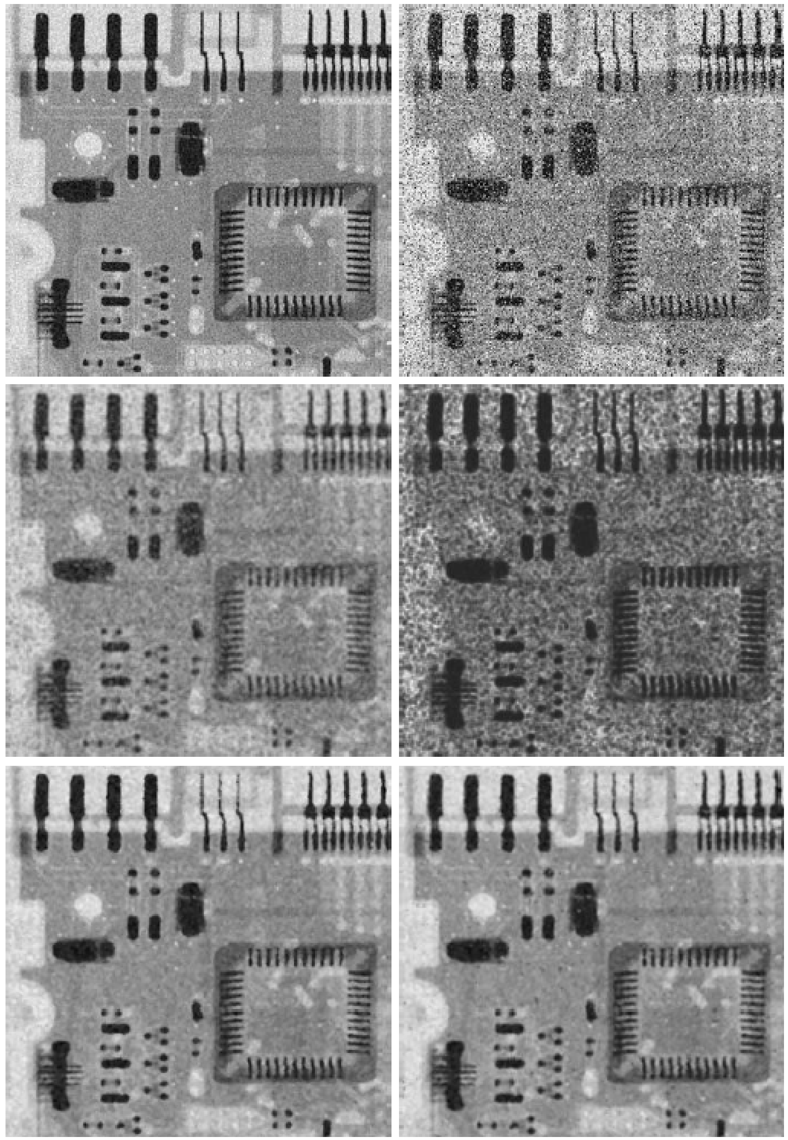
Order-Statistic Filters: Examples



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 5.8(b) with a min filter of the same size.

Example: Removing Mixed Noises



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. Image (b) filtered with a 5 × 5: (c) arithmetic mean filter; (d) geometric mean filter; (e) median filter; and (f) alpha-trimmed mean filter with $d = 5$.

Adaptive Filters

The filters discussed thus far are non-adaptive filters.

- whose coefficients are static, collectively forming the transfer function
- applied to an image regardless of how image characteristics vary from one point to another

In this section, two adaptive filters are discussed.

- whose behavior changes according to statistical characteristics of the image inside the filter window
- whose performance is superior to that of non-adaptive filters having discussed

Adaptive filters

- Adapted to the behavior based on the statistical characteristics of the image inside the filter region S_{xy}
- Improved performance v.s increased complexity
- Example: Adaptive local noise reduction filter

Adaptive local noise reduction filter

- Simplest statistical measurement
  - Mean and variance
- Known parameters on local region S_{xy}
  - g(x,y): noisy image pixel value
  - σ_{η}^{2}: noise variance (assume known a prior)
  - m_L : local mean
  - σ_L^{2} : local variance

Adaptive Local Noise Reduction

Adaptive, local noise reduction filter

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

• based on local mean (average intensity) m_L and local variance (contrast) σ_L^2
• if $\sigma_\eta^2 = 0$, no change
• if $\sigma_L^2 > \sigma_\eta^2$, edge, keep unchanged or less changed
• if $\sigma_L^2 \approx \sigma_\eta^2$, the m_L returns

• only the variance of corrupting noise $\hat{f}(x,y) \geq 0$ needed to be known or estimated
• assume $\sigma_\eta^2 \leq \sigma_L^2$, otherwise, set the ratio =1

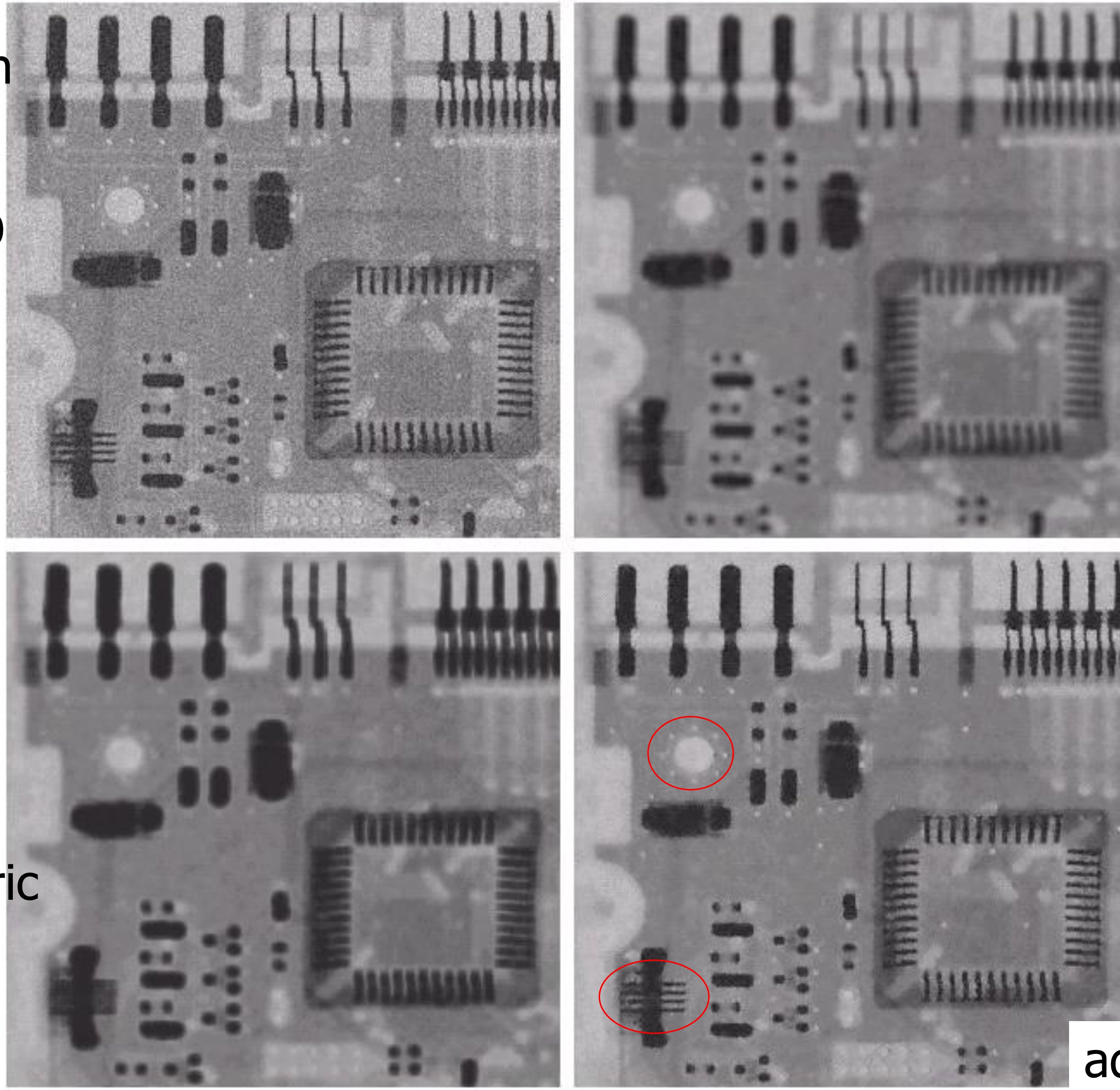
Adaptive local noise reduction filter (cont.)

- Analysis: we want to do
  - If σ_{η}^{2} is zero, return g(x,y)
  - If $\sigma_{L}^{2} > \sigma_{\eta}^{2}$, return value close to g(x,y)
  - If $\sigma_{L}^{2} = \sigma_{\eta}^{2}$, return the arithmetic mean m_L
- Formula

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

Gaussian noise
$\mu=0$
$\sigma^2=1000$

Arith. mean 7x7



Geometric mean 7x7

adaptive

Adaptive Median Filter

Whether z_{med} is NOT impulse

Adaptive median filter

z_{min} = minimum intensity value in S_{xy}
z_{max} = maximum intensity value in S_{xy}
z_{med} = median of intensity values in S_{xy}
z_{xy} = intensity value at coordinates (x, y)
S_{max} = maximum allowed size of S_{xy}

The adaptive median-filtering algorithm works in two stages, denoted stage A and stage B, as follows:

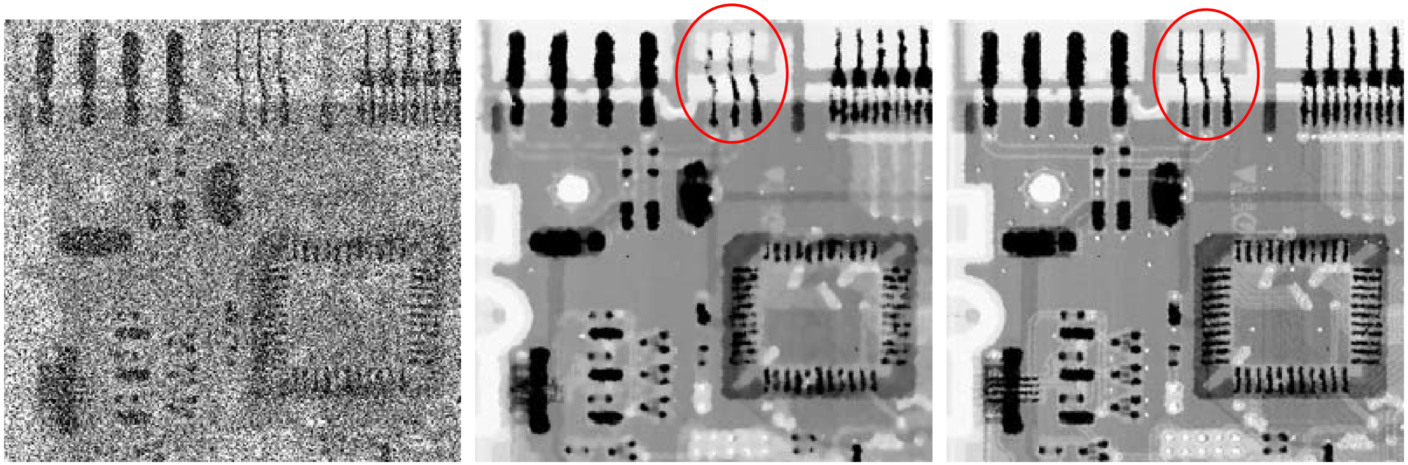
Stage A:
- $A1 = z_{med} - z_{min}$
- $A2 = z_{med} - z_{max}$
- If $A1 > 0$ AND $A2 < 0$, go to stage B
- Else increase the window size
- If window size $\leq S_{max}$ repeat stage A
- Else output z_{med}

Stage B:
- $B1 = z_{xy} - z_{min}$
- $B2 = z_{xy} - z_{max}$
- If $B1 > 0$ AND $B2 < 0$, output z_{xy}
- Else output z_{med}

Whether z_{xy} is NOT an extreme value

- representing the restored pixel value at (x,y) by executing pseudocode
- the size of filter window is adaptive
- three purposes: to remove salt-and-pepper noise (capable of handling large P_a and P_b), to smooth non-impulsive noise, and to reduce distortion, e.g., excessive thinning or thickening of object boundaries
- performance is better than un-adaptive median filter

Examples



a b c

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

Periodic noise reduction

- Pure sine wave
  - Appear as a pair of impulse (conjugate) in the frequency domain

$$\begin{cases} f(x,y) = A\sin(u_0 x + v_0 y) \\ F(u,v) = -j\frac{A}{2}\left[\delta(u - \frac{u_0}{2\pi}, v - \frac{v_0}{2\pi}) - \delta(u + \frac{u_0}{2\pi}, v + \frac{v_0}{2\pi})\right] \end{cases}$$

Periodic noise reduction (cont.)

- Bandreject filters
- Bandpass filters
- Notch filters
- Optimum notch filtering

Bandreject filters

* Reject an isotropic frequency

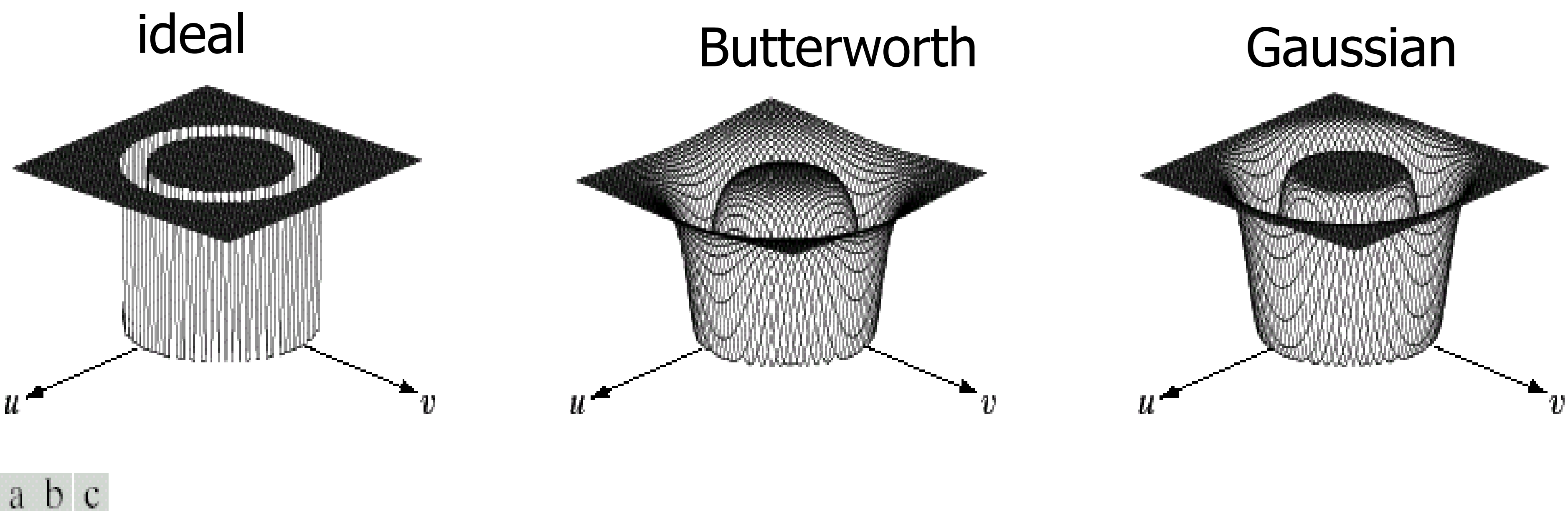
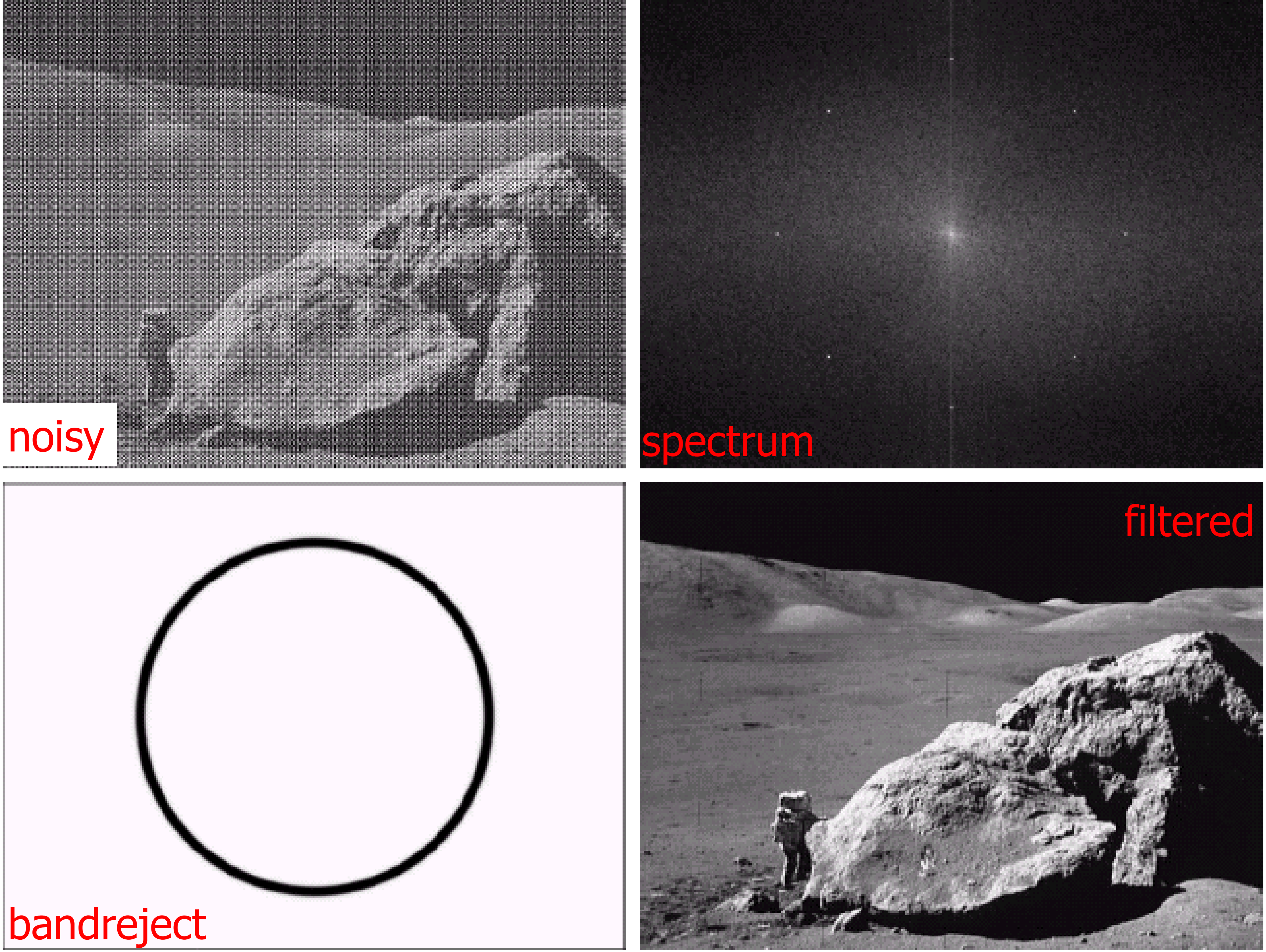
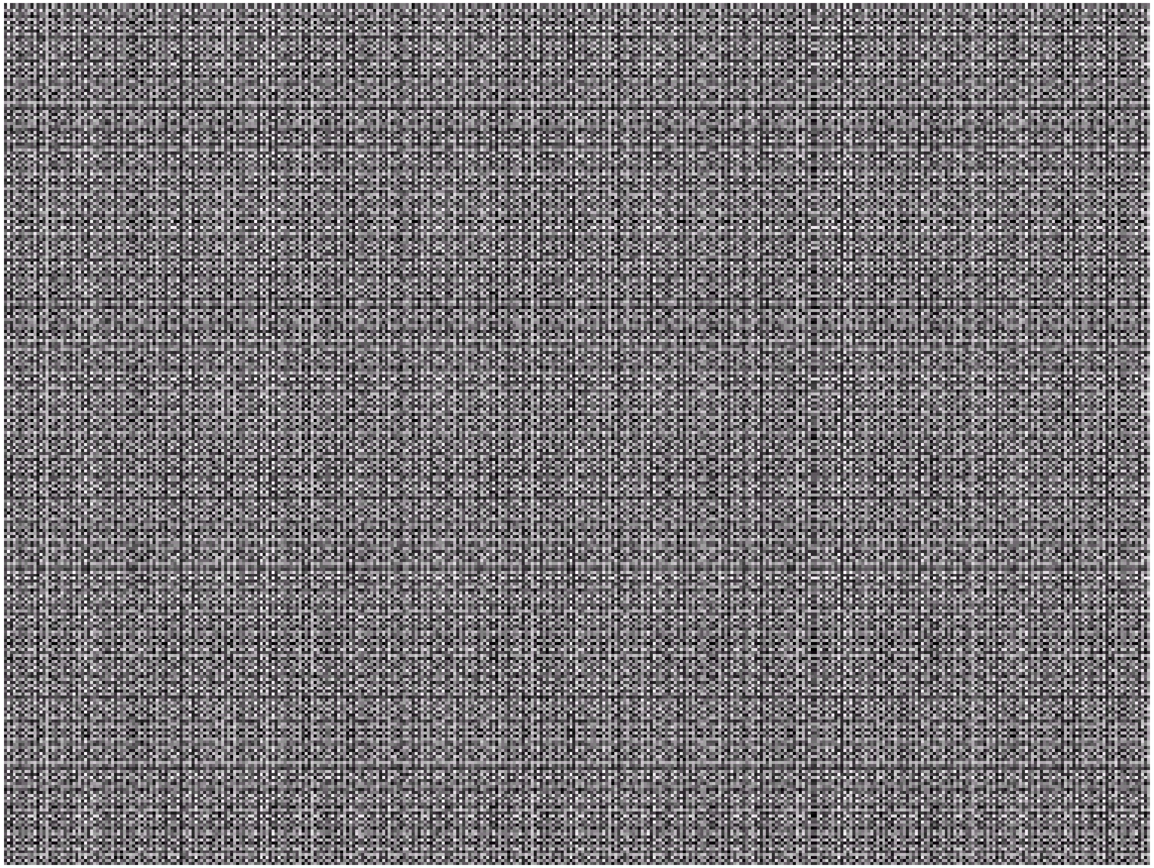


FIGURE 5.15 From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.


noisy
spectrum
bandreject
filtered

Bandpass filters

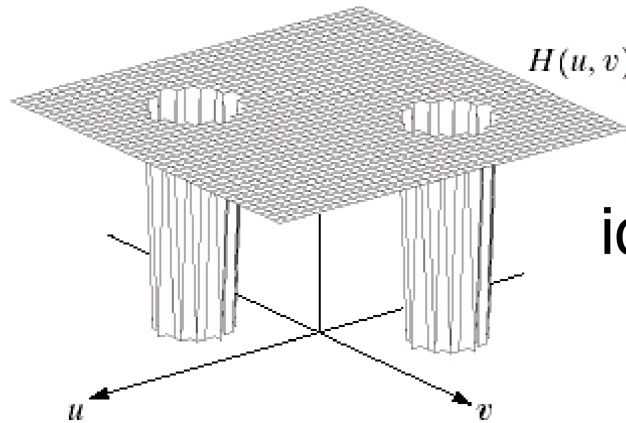
- $H_{bp}(u,v) = 1 - H_{br}(u,v)$



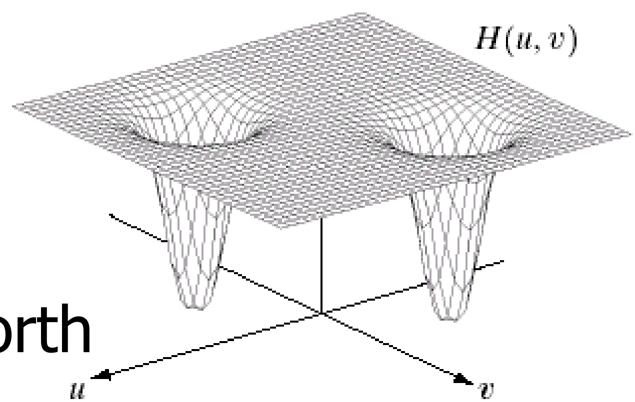
$$\Im^{-1}\left\{G(u,v)H_{bp}(u,v)\right\}$$

Notch filters

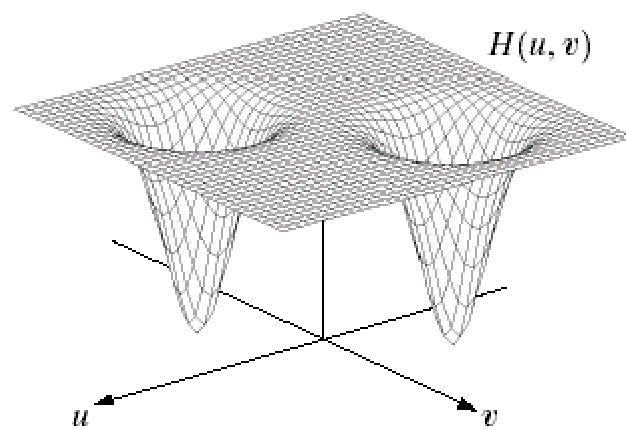
- Reject(or pass) frequencies in predefined neighborhoods about a center frequency



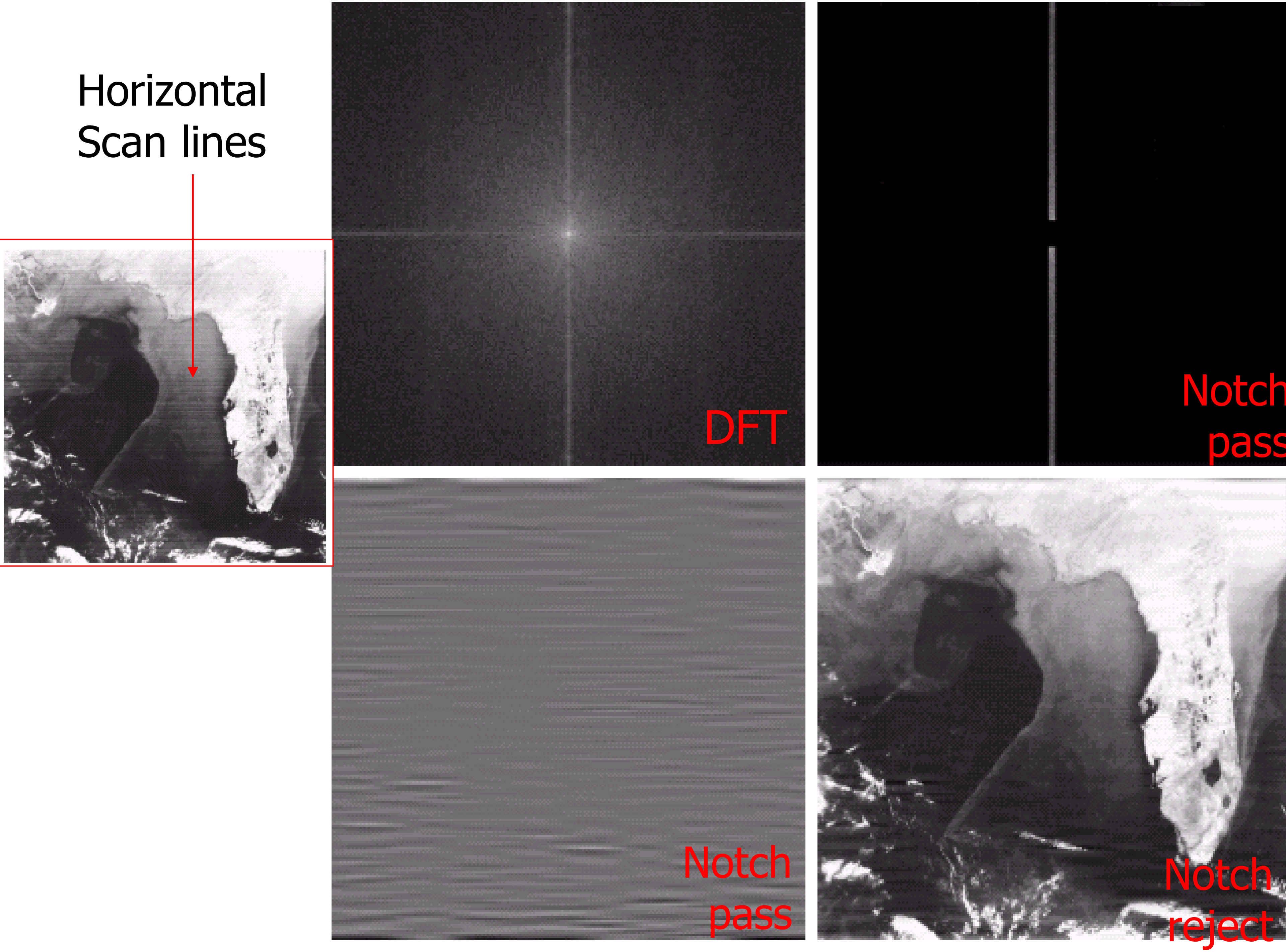
ideal



Butterworth



Gaussian


Horizontal
Scan lines
DFT
Notch
Notch
pass
Notch
reject

Estimating the degradation function

- Estimation by Image observation
- Estimation by experimentation
- Estimation by modeling

Estimation by image observation

- Take a window in the image
  - Simple structure
  - Strong signal content
- Estimate the original image in the window

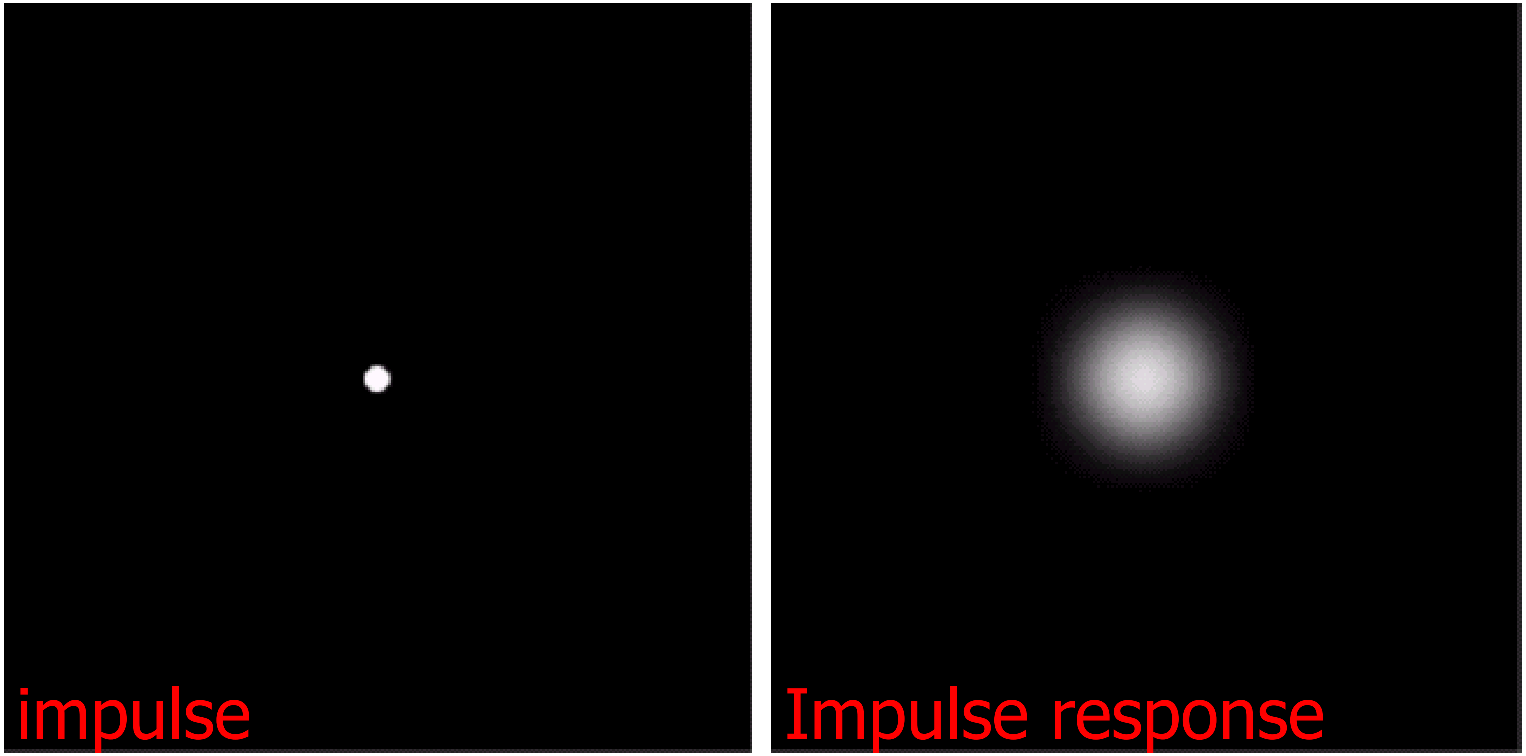
$$H_s(u,v) = \frac{G_s(u,v)}{\hat{F}_s(u,v)}$$

known

estimate

Estimation by experimentation

- If the image acquisition system is ready
- Obtain the impulse response



Estimation by modeling (1)

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

original

k=0.0025

k=0.001

k=0.00025

Estimation by modeling (2)

- Derive a mathematical model
- Ex. Motion of image

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

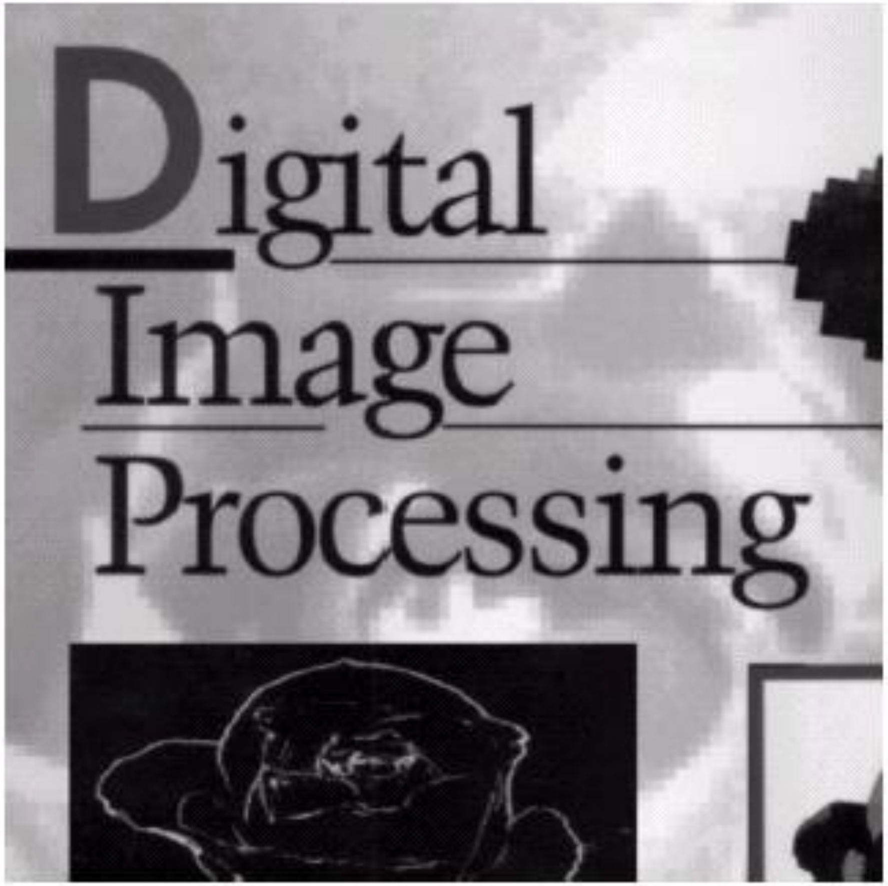
Planar motion

Fourier transform

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

Estimation by modeling: example

original



Apply motion model

Inverse filtering

- With the estimated degradation function H(u,v)

Unknown noise

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

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

Estimate of original image

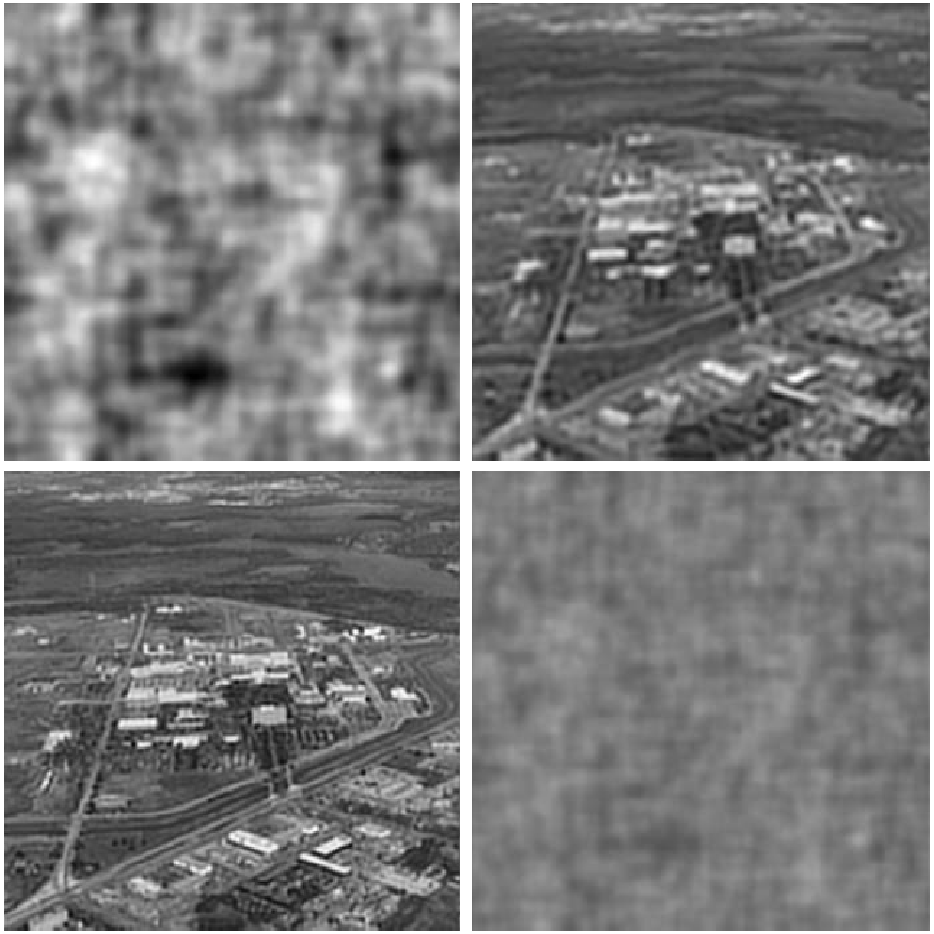
Problem: 0 or small values

Sol: limit the frequency around the origin

Effect of Inverse Filtering

a b
c d

FIGURE 5.27 Restoring Fig. 5.25(b) with Eq. (5.7-1). (a) Result of using the full filter. (b) Result with H cut off outside a radius of 40; (c) outside a radius of 70; and (d) outside a radius of 85.



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

Wiener Filtering

- Traditional inverse filtering does not explicitly deal with the noise

- Wiener Filtering considers both the degradation and the statistical characteristics of the noise

- The objective is to find an image estimate such that the mean square error (MSE) between the uncorrupted image f and the estimate $\hat{f}$ is minimized.

$$\text{minimize} \quad E\left\{(f-\hat{f})^2\right\}$$

Wiener Filter: Solution

Based on aforementioned conditions, the image estimate in the frequency domain is given by

$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)} \frac{\left|H(u,v)\right|^2}{\left|H(u,v)\right|^2 + S_\eta(u,v)/S_f(u,v)} \right] G(u,v) \qquad \text{(Eq. 5.8-2)}$$

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

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

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

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

Case Study:

The Wiener Filter

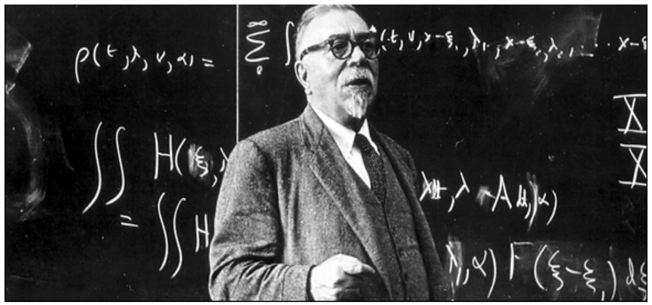
Let s_i be a digital signal consisting of N real numbers $i = 0, 1, 2, \ldots, N\text{-}1$ that has been generated via the time invariant linear process (where p_i – the IRF – is known)

$$s_i = \sum_j p_{i-j} f_j + n_i$$

$$\sum_j \equiv \sum_{j=0}^{N-1}$$

find an estimate for f_i of the form

$$\hat{f}_i = \sum_j q_j s_{i-j}$$



Solution

- Consider the least squares error

$$e = \|f_i - \hat{f}_i\|_2^2 \equiv \sum_{i=0}^{N-1} (f_i - \hat{f}_i)^2$$

- e is a minimum when

$$\frac{\partial}{\partial q_k} e(q_j) = 0 \quad \forall k$$

i.e. when

$$\sum_{i=0}^{N-1} \left(f_i - \sum_j q_j s_{i-j} \right) s_{i-k} = 0$$

Solution (continued)

- Using the convolution and correlation theorems

$$F_i S_i^* = Q_i S_i S_i^* \qquad Q_i = \frac{S_i^* F_i}{\mid S_i \mid^2}$$

- Since $S_i = P_i F_i + N_i$

$$Q_i = \frac{P_i^* \mid F_i \mid^2 + N_i^* F_i}{\mid P_i \mid^2 \mid F_i \mid^2 + \mid N_i \mid^2 + P_i F_i N_i^* + N_i P_i^* F_i^*}$$

Signal Independent Noise

- We can not compute Q_i because we do not know F_i or N_i

- However, we can expect that the information content of the signal f_i will not correlate with the noise n_i which means that

$$\sum_j n_{j-i} f_j = 0 \quad \text{and} \quad \sum_j f_{j-i} n_j = 0$$

$$N_i^* F_i = 0 \quad \text{and} \quad F_i^* N_i = 0$$

Signal Independent Noise Solution

Given that the noise is signal independent

$$Q_i = \frac{P_i^* \mid F_i \mid^2}{\mid P_i \mid^2 \mid F_i \mid^2 + \mid N_i \mid^2}$$

and

$$Q_i = \frac{P_i^*}{\mid P_i \mid^2 + \frac{|N_i|^2}{|F_i|^2}}$$

Computing the Signal-to-Noise-Ratio

- **Problem:** How can we find $\mid F_i \mid^2 / \mid N_i \mid^2$?
- Suppose we have a linear stationary process whereby we can record a signal twice at different times. Then

$$s_i = p_i \otimes f_i + n_i$$

$$s'_i = p_i \otimes f_i + n'_i$$

$$n_i \odot n'_i = 0, \quad f_i \odot n_i = 0, \quad n_i \odot f_i = 0,$$

$$f_i \odot n'_i = 0, \quad n'_i \odot f_i = 0.$$

Autocorrelation and Cross-correlation Functions

- ***Auto-correlating:*** $c_i = s_i \odot s_i$

$$C_i = S_i S_i^* = |P_i|^2 |F_i|^2 + |N_i|^2$$

- ***Cross-correlating:*** $c_i' = s_i \odot s_i'$

$$C_i' = |P_i|^2 |F_i|^2$$

$$\frac{|N_i|^2}{|F_i|^2} = \left(\frac{C_i}{C_i'} - 1\right) |P_i|^2$$

Practical Implementation

- Given that the signal-to-noise power ratio is not usually known, i.e. $|F_i|^2 / |N_i|^2$ we approximate the filter as

$$Q_i \sim \frac{P_i^*}{|P_i|^2 + \Gamma} \qquad \Gamma \sim \frac{1}{(\mathrm{SNR})^2}$$

- The value of the SNR (Signal-to-Noise-Ratio) becomes a user defined constant

Wiener Filter (cont'd)

Measures based on the power spectra of noise and of the undegraded image characterize the performance of restoration algorithms.

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

$$SNR = \frac{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u,v)|^2}{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |N(u,v)|^2} \text{ or } \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \hat{f}(x,y)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[f(x,y) - \hat{f}(x,y)\right]^2}$$

Wiener Filter (cont'd)

When noise spectrum is constant (white noise), things can be considerably simplified; however, the spectrum of undegraded image seldom is known.

Under this circumstance, the image estimate, Eq. 5.8-2, may be rewritten as

$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K} \right] G(u,v) \qquad \text{(Eq. 5.8-6)}$$

where K is a specified constant.

This equation is often actually utilized for Wiener filtering.

FFT Algorithm for the Wiener Filter

```
snr=snr*snr
constant=1/snr

for i=1, 2, ..., n; do:
    sr(i)=signal(i)
    si(i)=0.
    pr(i)=IRF(i)
    pi(i)=0.
enddo
        forward_fft(sr,si)
        forward_fft(pr,pi)
```

FFT Algorithm for the Wiener Filter (continued)

```
for i=1, 2, ..., n; do:
    denominator=pr(i)*pr(i)+pi(i)*pi(i)+constant

                fr(i)=pr(i)*sr(i)+pi(i)*si(i)
                fi(i)=pr(i)*si(i)-pi(i)*sr(i)
                fr(i)=fr(i)/denominator
                fi(i)=fi(i)/denominator
enddo
inverse_fft(fr,fi)

 for i=1, 2, ..., n; do:
     hatf(i)=fr(i)
 enddo
```

Signal Restoration using the *Wiener Filter*

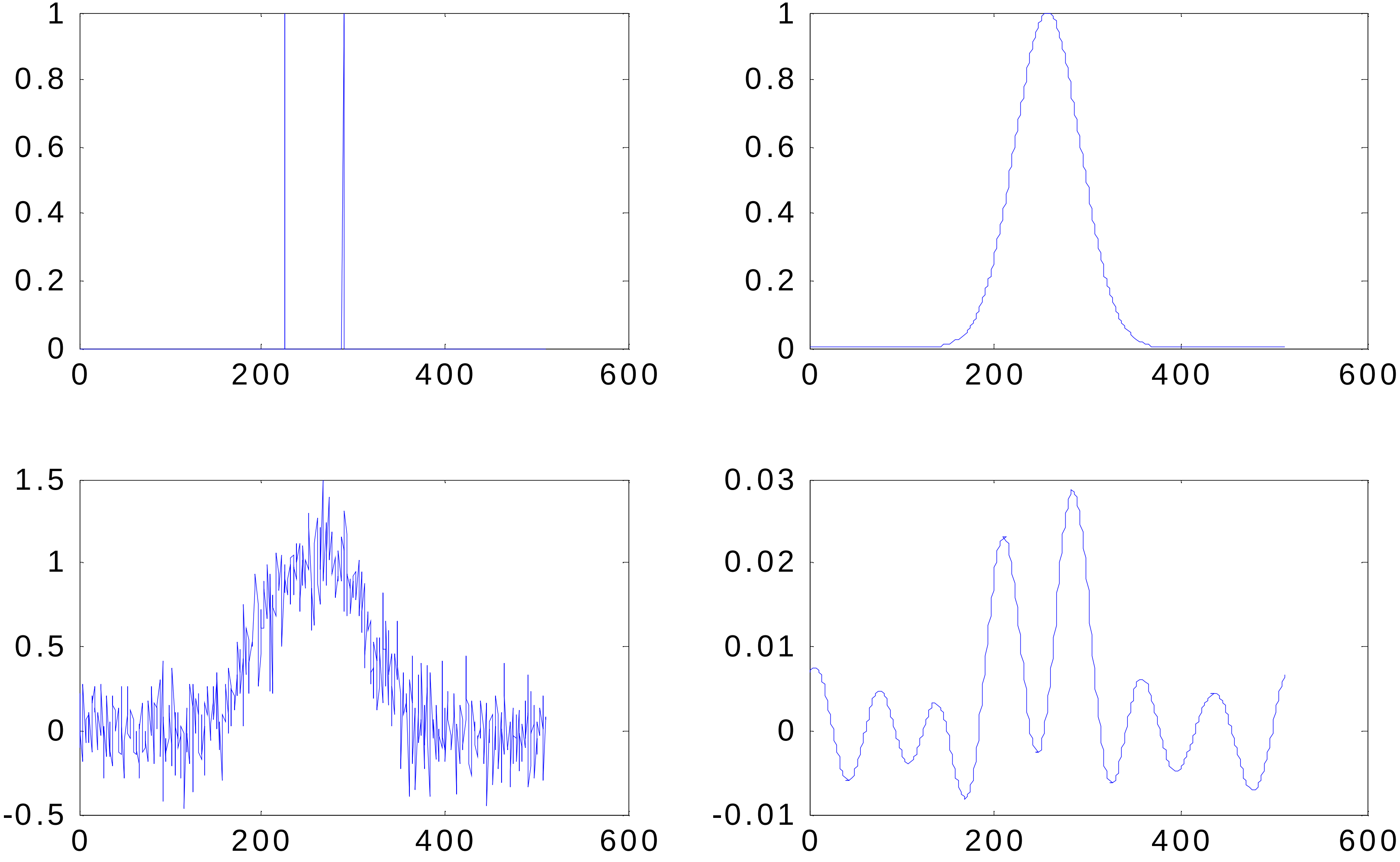


Image Restoration using the Wiener Filter

Image restoration using the Wiener filter.

Original image (left), Gaussian PSF (center) and restoration after application of the Wiener filter (right) using a standard deviation of 3 pixels (for the Gaussian PSF) and an SNR=1.

Wiener Filtering: Results

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.

K manually adjusted to yield the best visual results

Wiener filtering yielded a result very closer to the original image.

A Further Example