Digital Image Processing

COSC 6380/4393

Lecture – 25

Apr 18th, 2023

Pranav Mantini

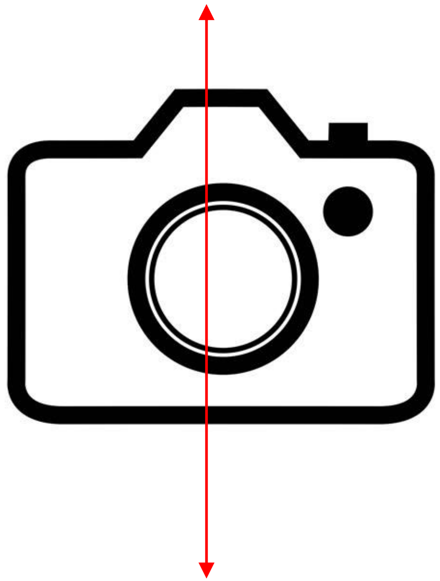
Review: Image Restoration

- Image restoration: recover an image that has been degraded by using a prior knowledge of the degradation phenomenon.

- Model the degradation and applying the inverse process in order to recover the original image.

A Model of Image Degradation/Restoration Process

$h(x,y) \otimes f(x,y)$



Motion
Turbulence

Additive noise

$\eta(x,y)$

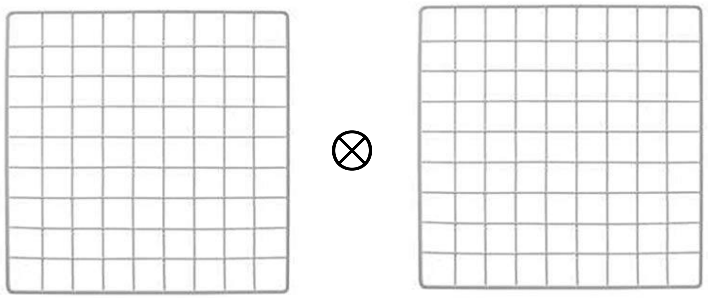
Noise:
Image acquisition: light, temperature, etc.

Transmission: atmospheric , electronic disturbance

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

A Model of Image Degradation/Restoration Process

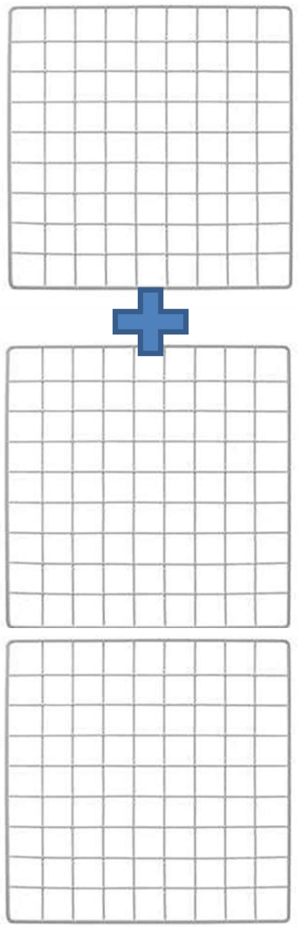
$h(x,y) \otimes f(x,y)$



Ideal image

Degradation function

$\eta(x,y)$

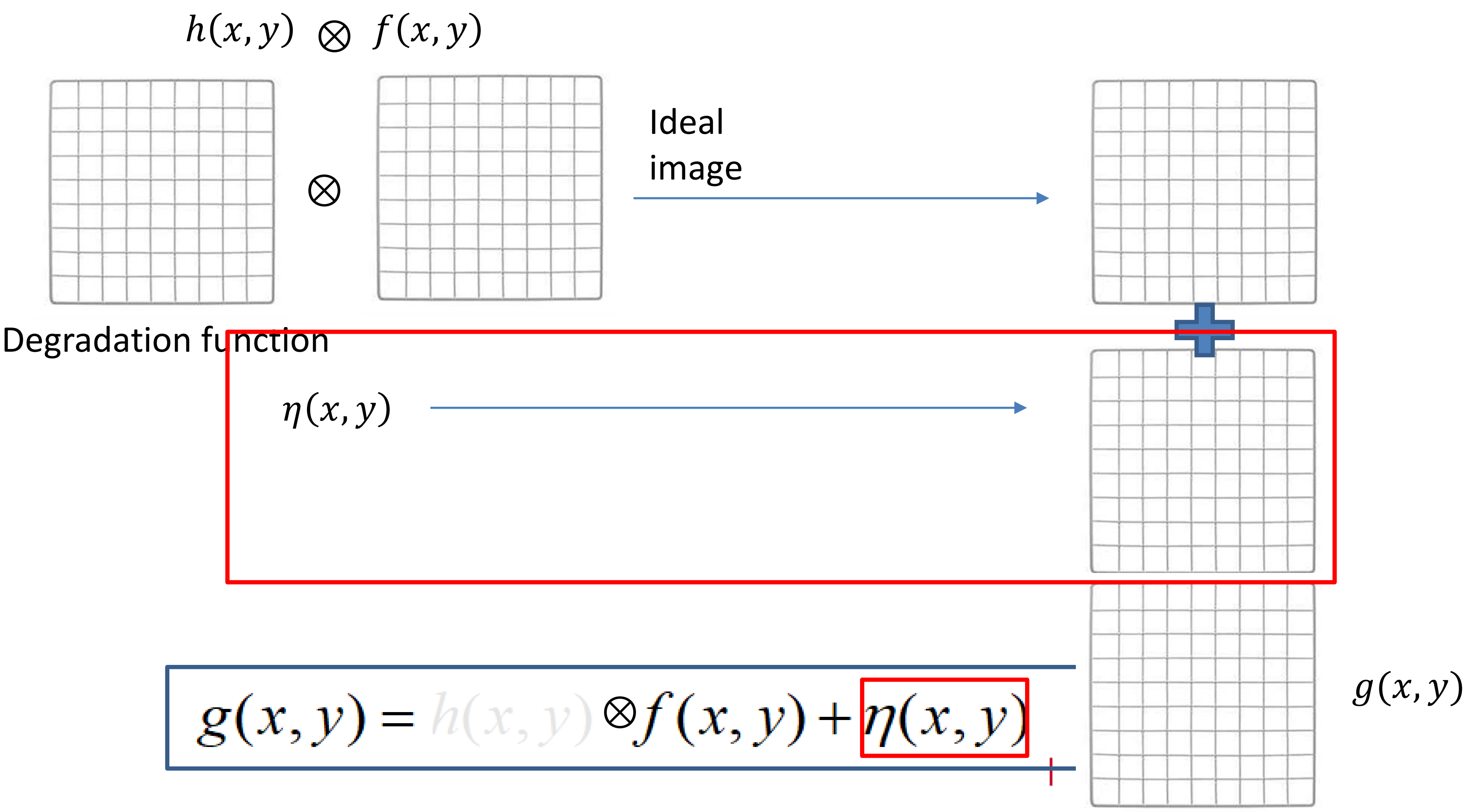


Goal of image restoration is to estimate $f'(x,y)$ that is as close as possible to $f(x,y)$

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

$g(x,y)$

A Model of Image Degradation/Restoration Process



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

Review: Noise Sources

- The principal sources of noise in digital images arise during **image acquisition and/or transmission**

✓ Image acquisition

e.g., light levels, sensor temperature, etc.

✓ Transmission

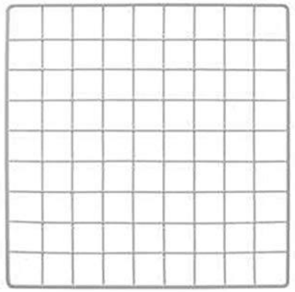
e.g., lightning or other atmospheric disturbance in wireless network

Types of Noise

- Spatially independent

The noise at location (x, y), $\eta(x, y)$ is defined by a function H that is not dependent on the (x, y)

$$\eta(x, y) \rightarrow H$$

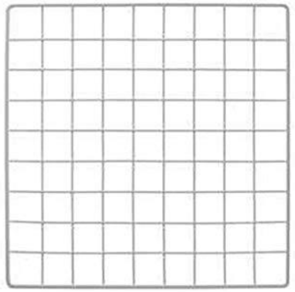


Statistical noise

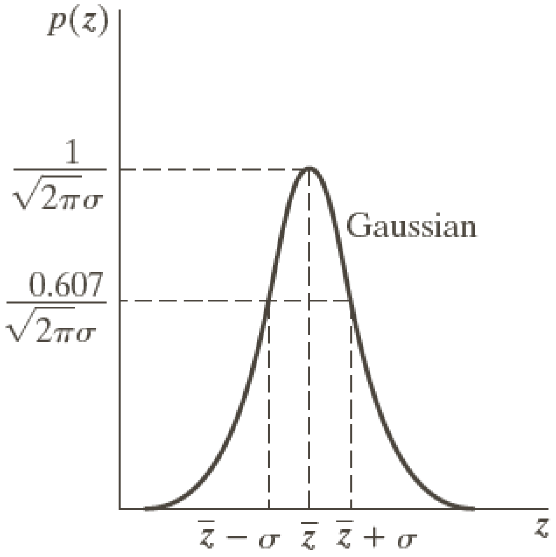
- Spatially dependent

The noise at location (x, y), $\eta(x, y)$ is defined by a function H that is dependent on the location (x, y)

$$\eta(x, y) \rightarrow H(x, y)$$



Periodic noise



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

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

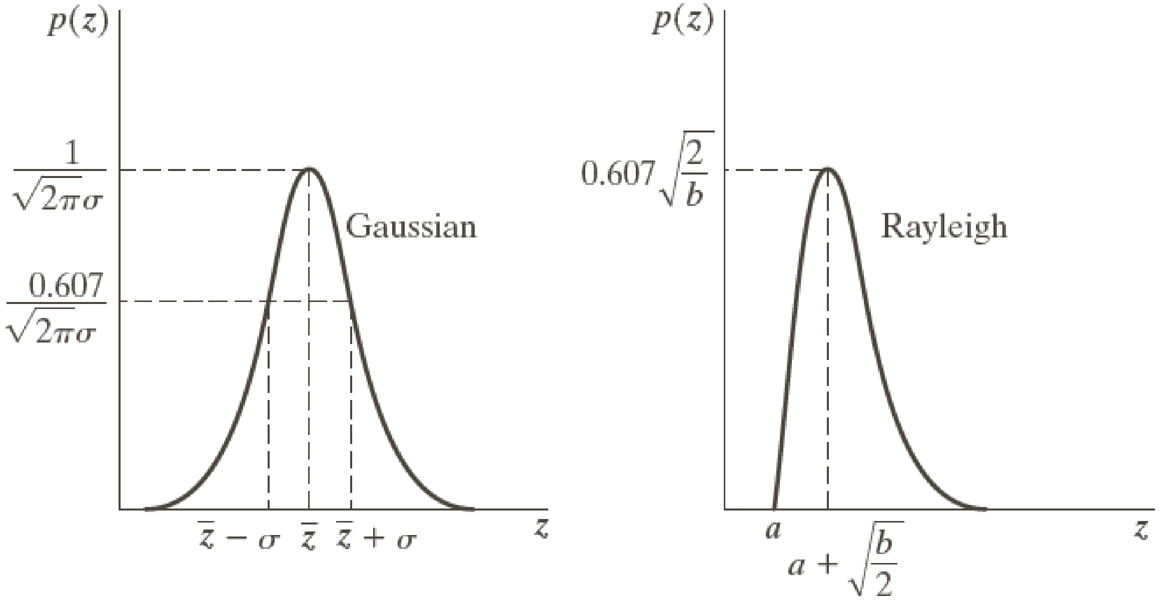
a	b	c
d	e	f

FIGURE 5.2 Some important probability density functions.

Review: Gaussian Noise

- **Gaussian noise**

  Electronic circuit noise, sensor noise due to poor illumination and/or high temperature



The PDF of Rayleigh noise is given by

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

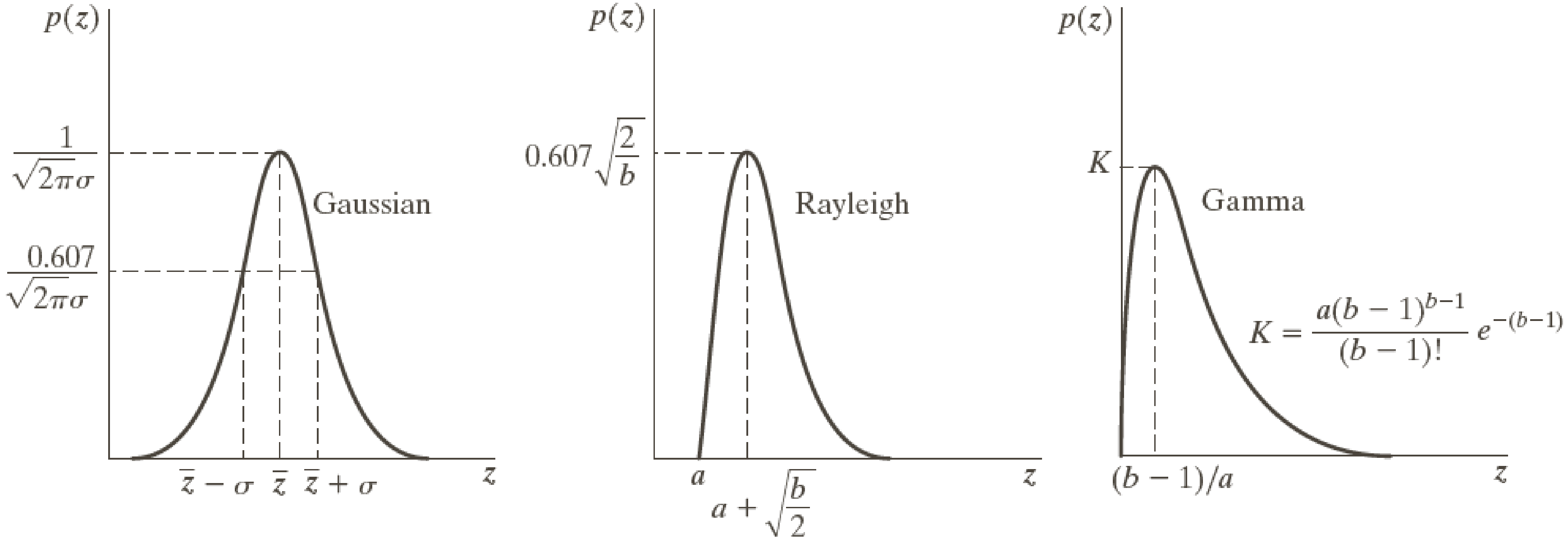
a	b	c
d	e	f

FIGURE 5.2 Some important probability density functions.

Review: Rayleigh Noise

- **Rayleigh noise**

  Range imaging

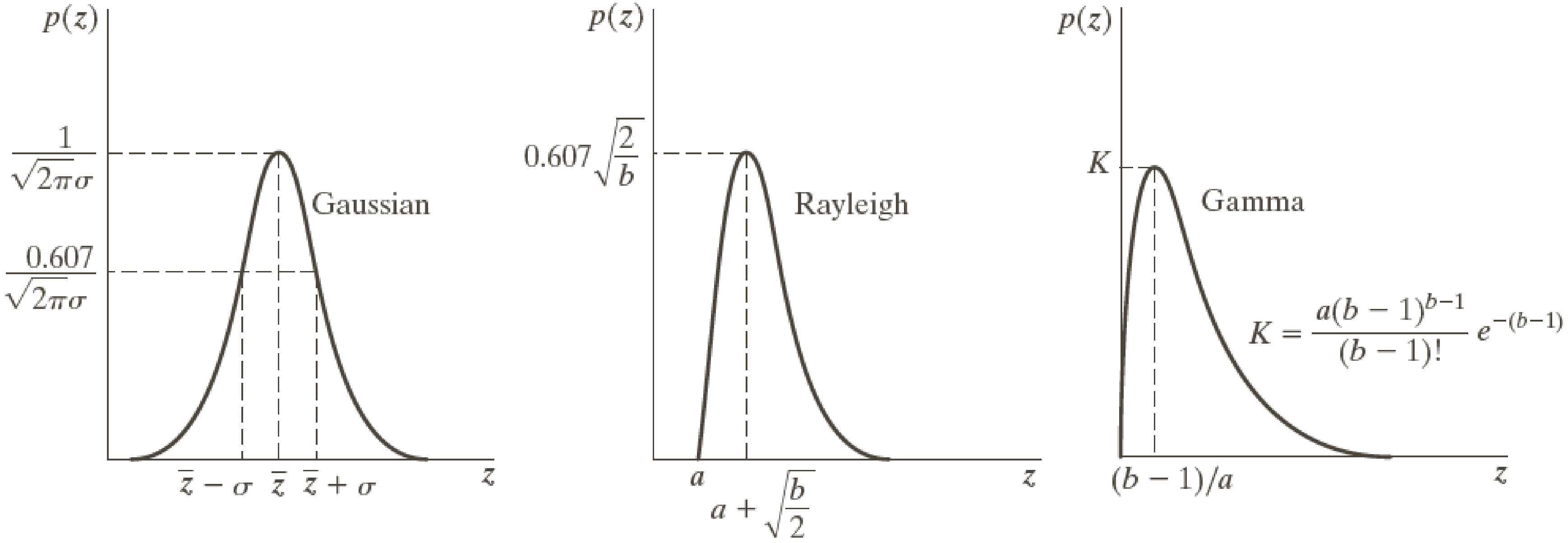


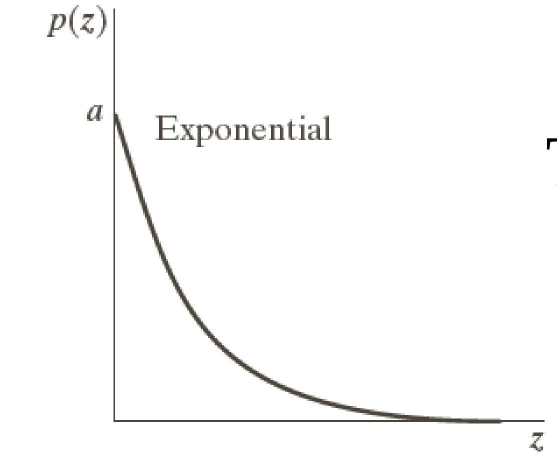
The PDF of Erlang noise is given by

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

a	b	c
d	e	f

FIGURE 5.2 Some important probability density functions.





The PDF of exponential noise is given by

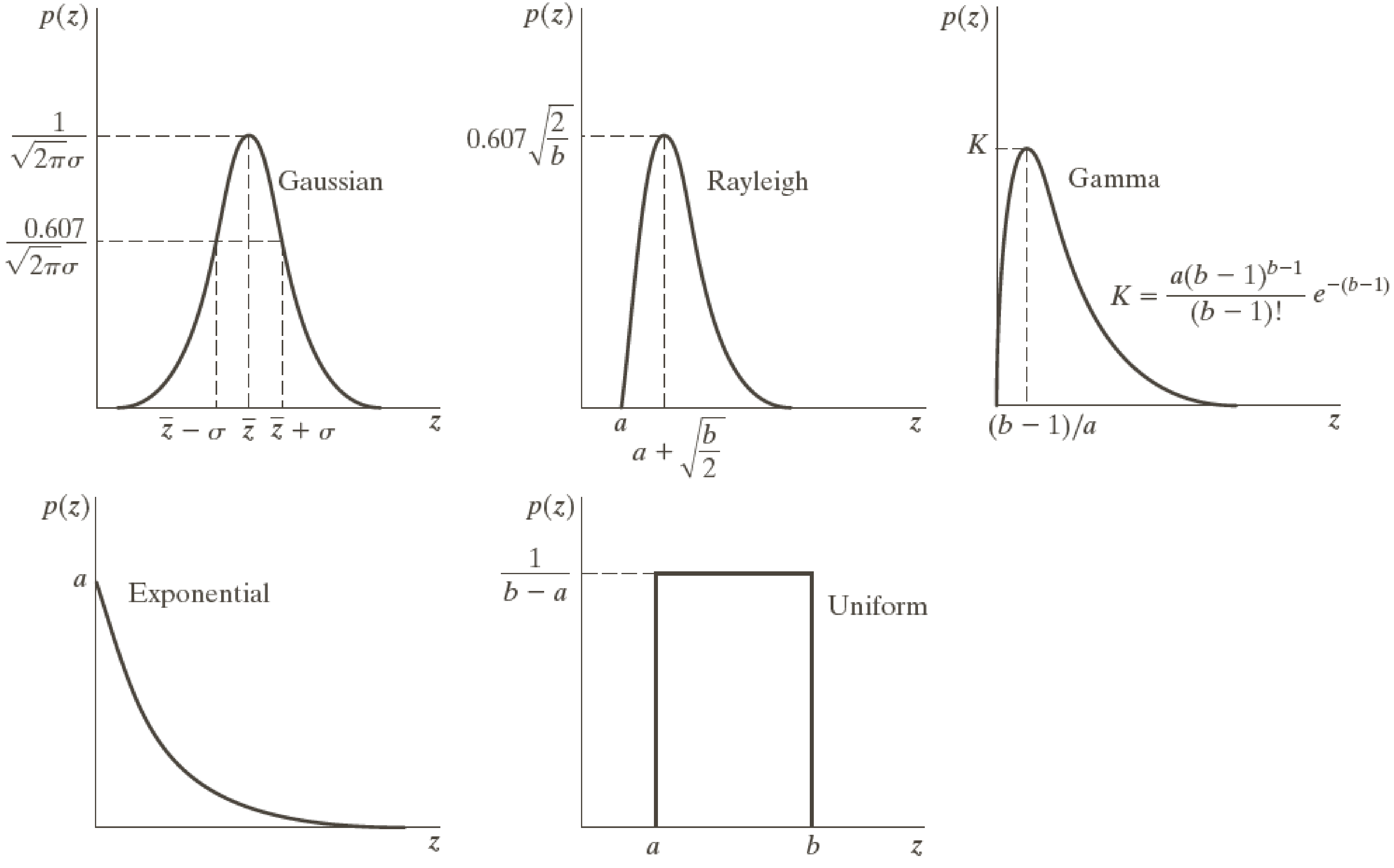
$$p(z) = \begin{cases} ae^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < a \end{cases}$$

a	b	c
d	e	f

FIGURE 5.2 Some important probability density functions.

Review:

- **Erlang (gamma) noise**: Laser imaging
- **Exponential noise**: Laser imaging



a b c
d e f

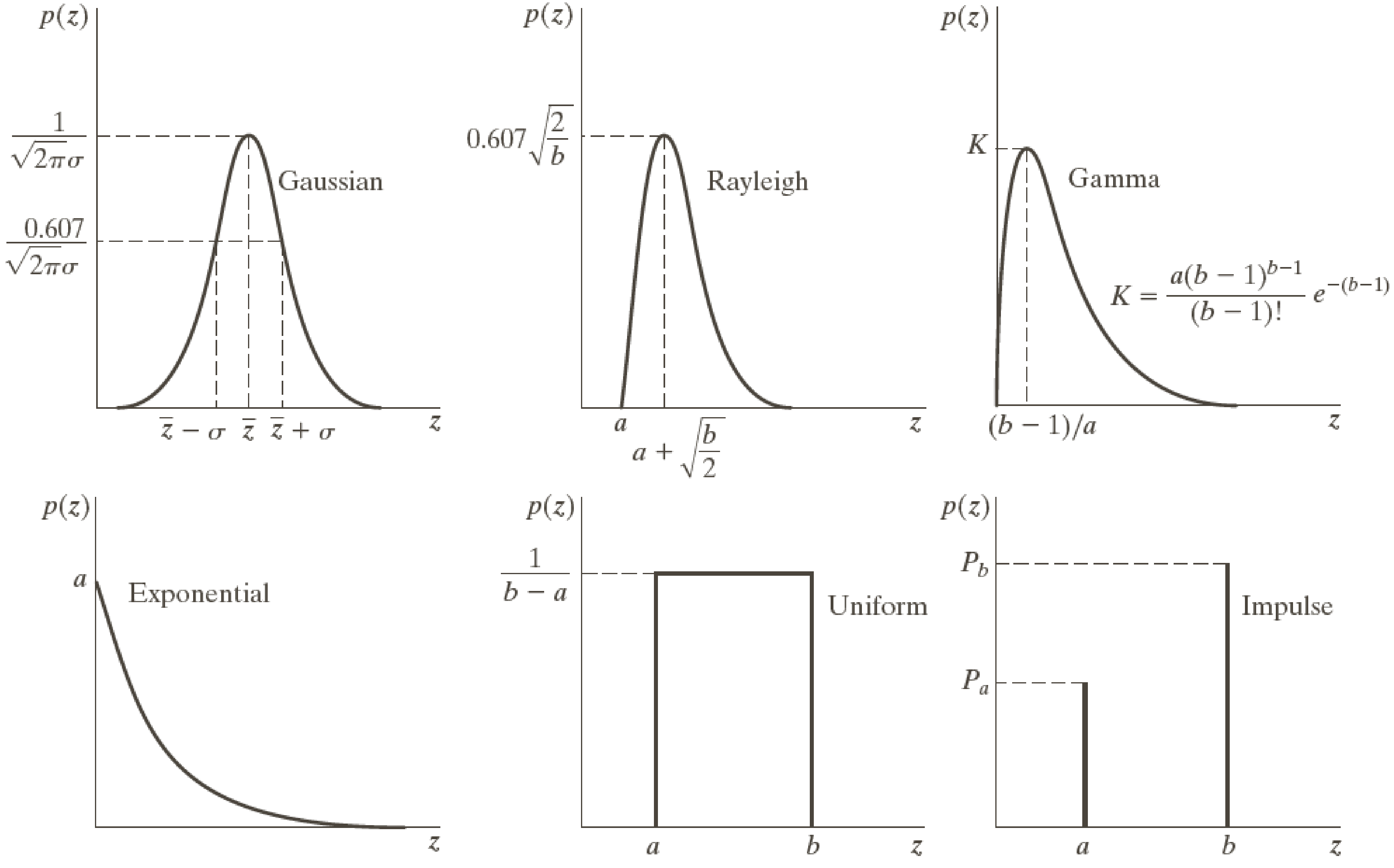
FIGURE 5.2 Some important probability density functions.

The PDF of uniform noise is given by

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

Review:

- **Uniform noise**: Least descriptive; Basis for numerous random number generators
- Not common in practical situations



a	b	c
d	e	f

FIGURE 5.2 Some important probability density functions.

The PDF of (bipolar) impulse noise is given by

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

Review:

- **Impulse noise**: quick transients, such as faulty switching

Examples of Noise: Original Image

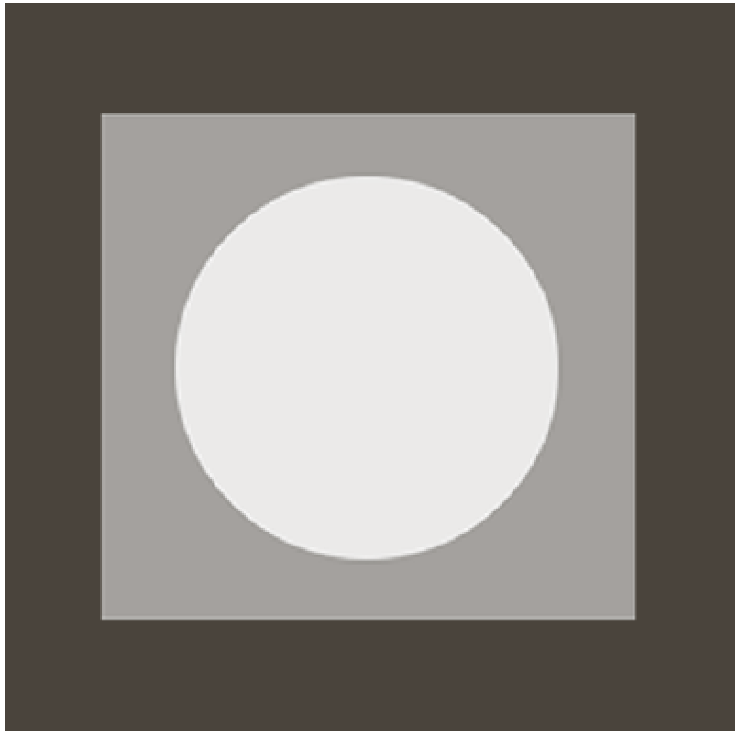


FIGURE 5.3 Test pattern used to illustrate the characteristics of the noise PDFs shown in Fig. 5.2.

Examples of Noise: Noisy Images(1)

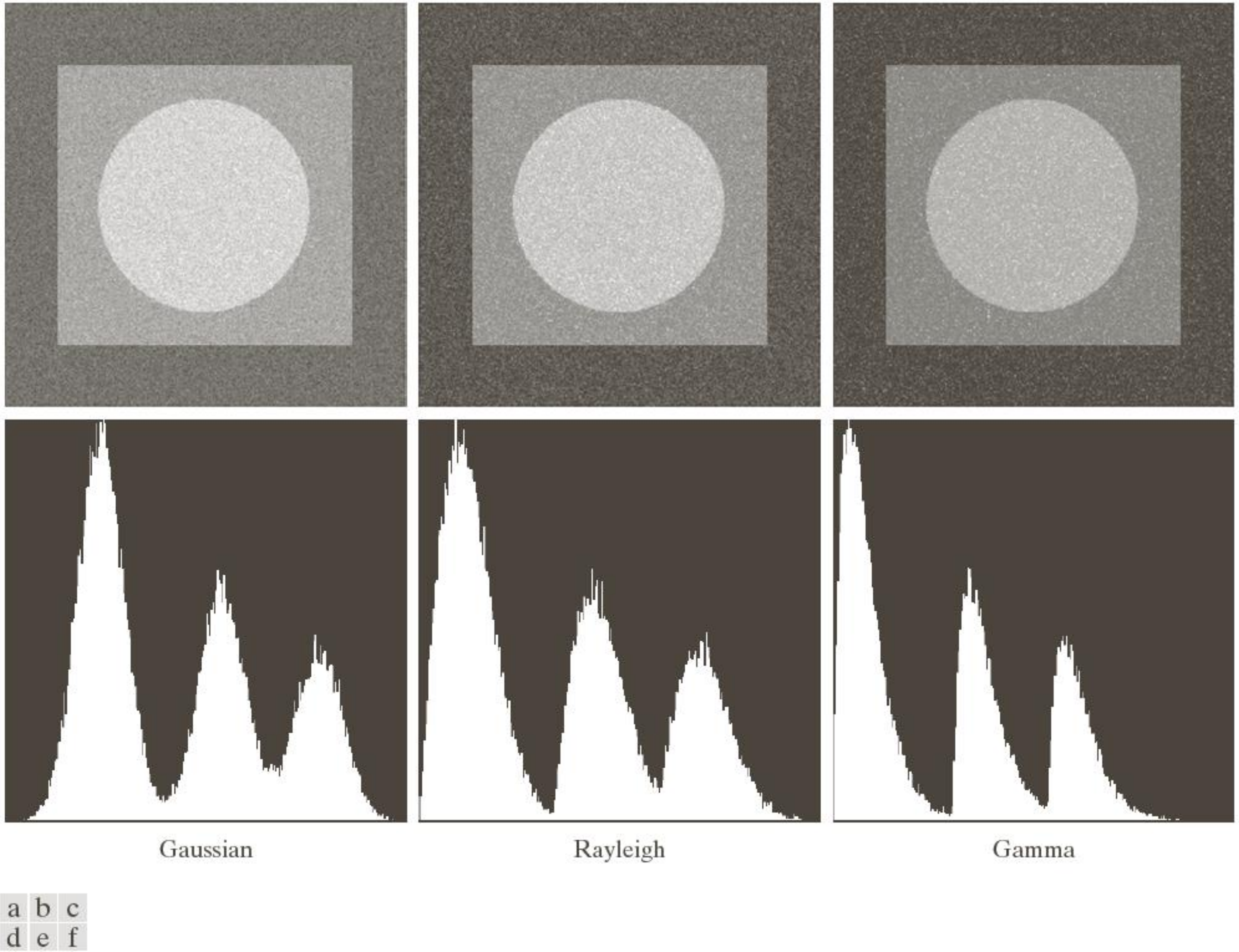


FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and gamma noise to the image in Fig. 5.3.

Examples of Noise: Noisy Images(2)

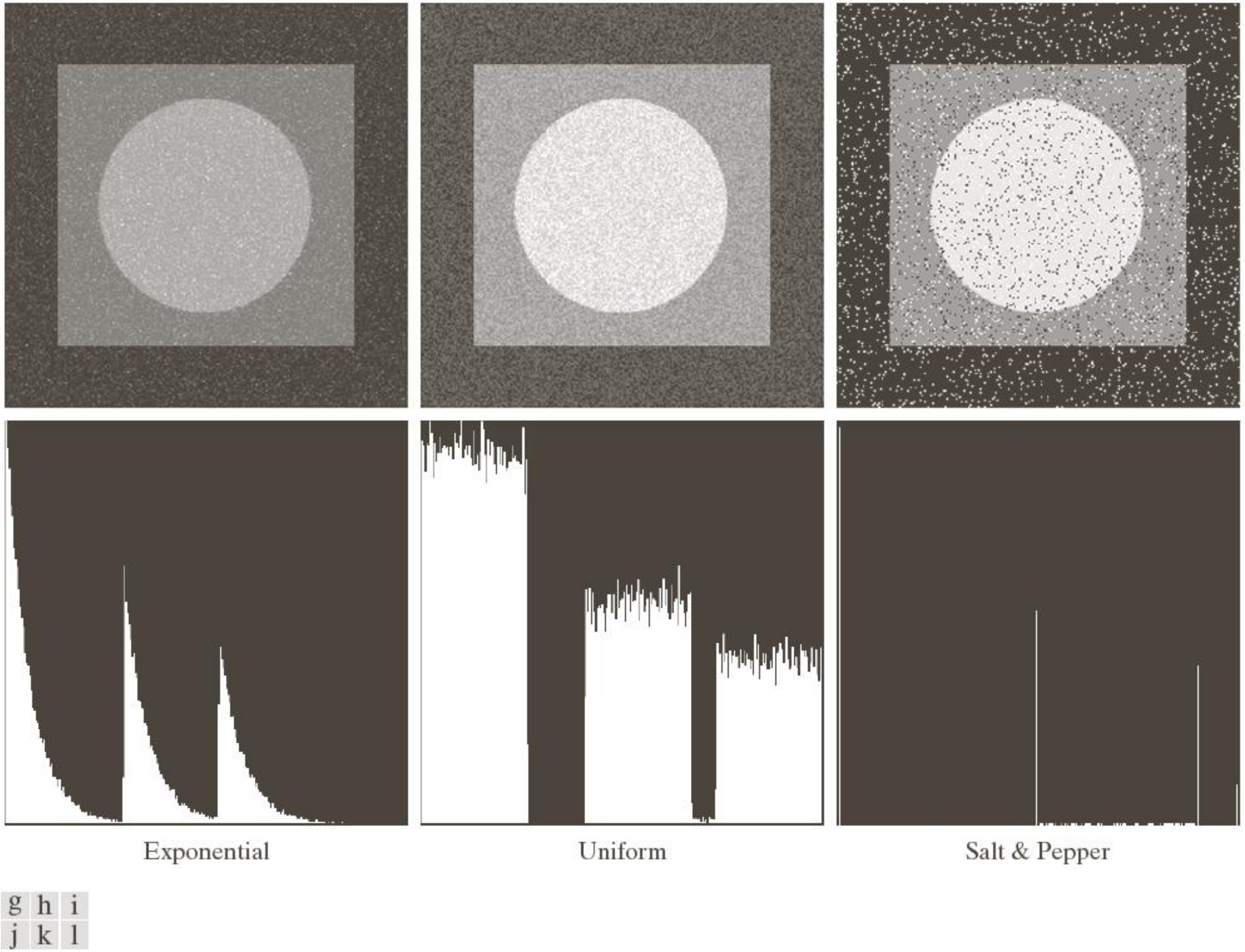
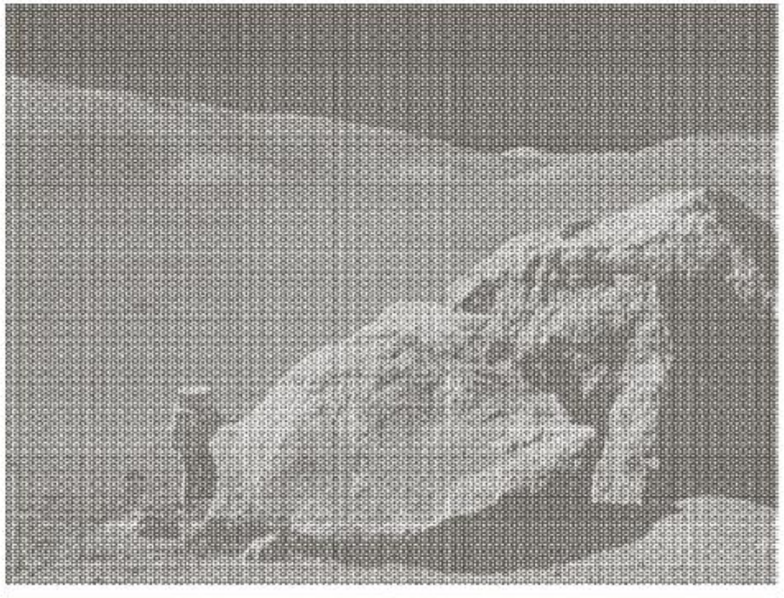


FIGURE 5.4 *(Continued)* Images and histograms resulting from adding exponential, uniform, and salt and pepper noise to the image in Fig. 5.3.

Periodic Noise

- Periodic noise in an image arises typically from electrical or electromechanical interference during image acquisition.
- It is a type of spatially dependent noise
- Periodic noise can be reduced significantly via frequency domain filtering

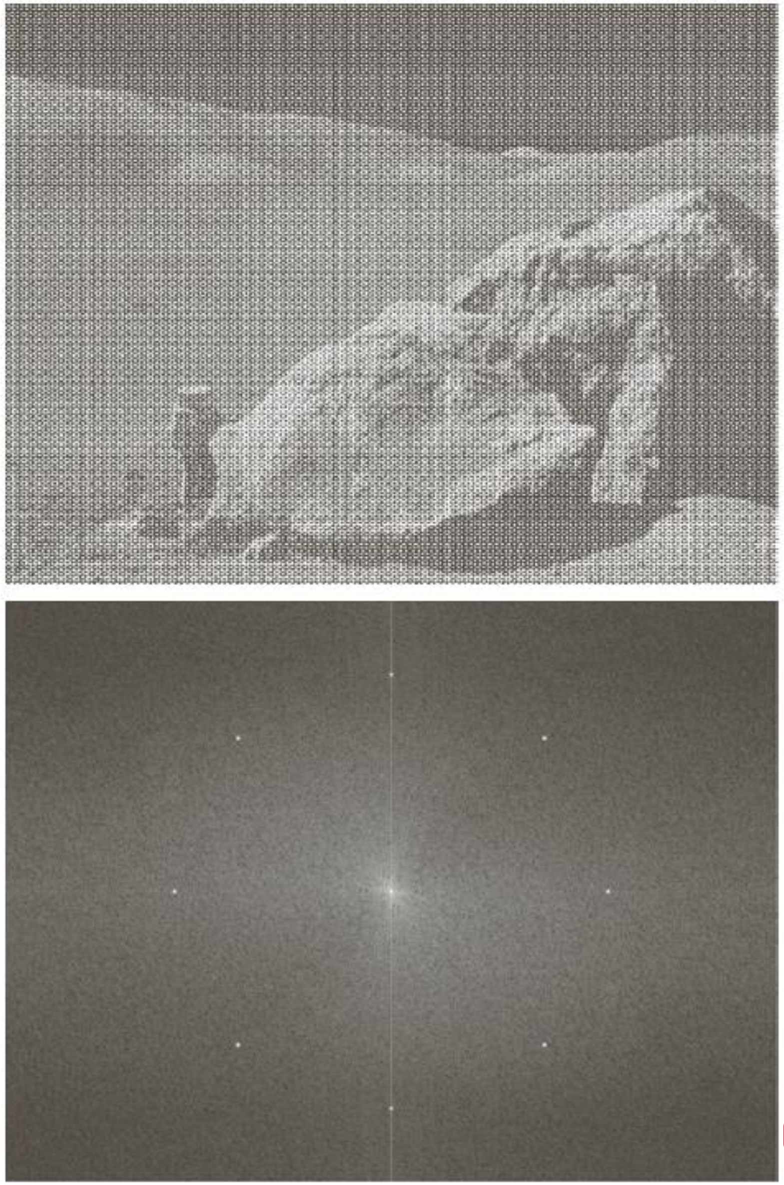
An Example of Periodic Noise



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.)

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.)

Estimation of Noise

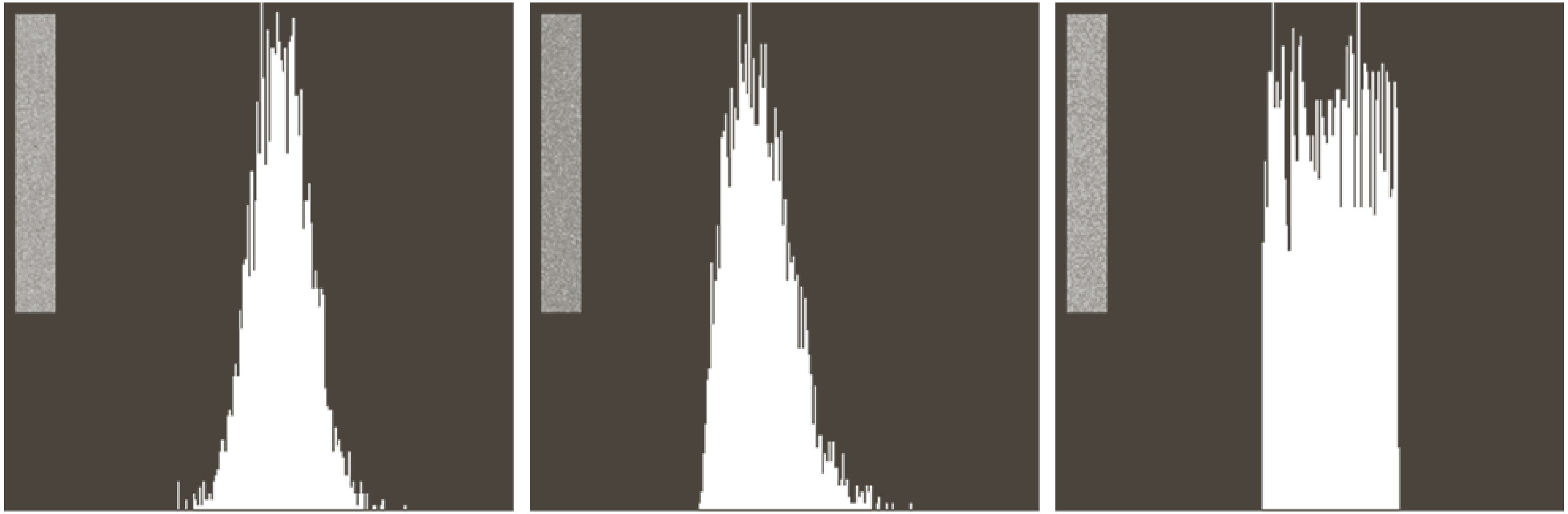
- Periodic noise:
  - Produces frequency spikes
  - Estimated by inspecting the Fourier spectrum.
  - Can be automated if the spikes are pronounced
  - Infer periodicity directly from images (only simplistic cases)

Estimation of Noise

- Statistical noise:
  - Known partially from the sensor specifications
  - If imaging system is available, capture a set of flat images (solid grey board) and study the histogram
  - If only images are available, select an area of image with reasonably constant background and study the histogram.

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

Rayleigh Noise

The PDF of Rayleigh noise is given by

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

The mean and variance of this density are given by

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

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

Restoration in the Presence of Noise Only – Spatial Filtering

Noise model without degradation

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

Restoration in the Presence of Noise Only – Spatial Filtering

Noise model without degradation

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

and

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

Restoration Filters

- Mean Filters
- Order statistic filters
- Adaptive filters

Spatial Filtering: Mean Filters (1)

Let S_{xy} represent the set of coordinates in a rectangle subimage window of size $m \times n$, centered at (x, y).

Arithmetic mean filter

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

Spatial Filtering: Mean Filters (2)

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: Example (1)

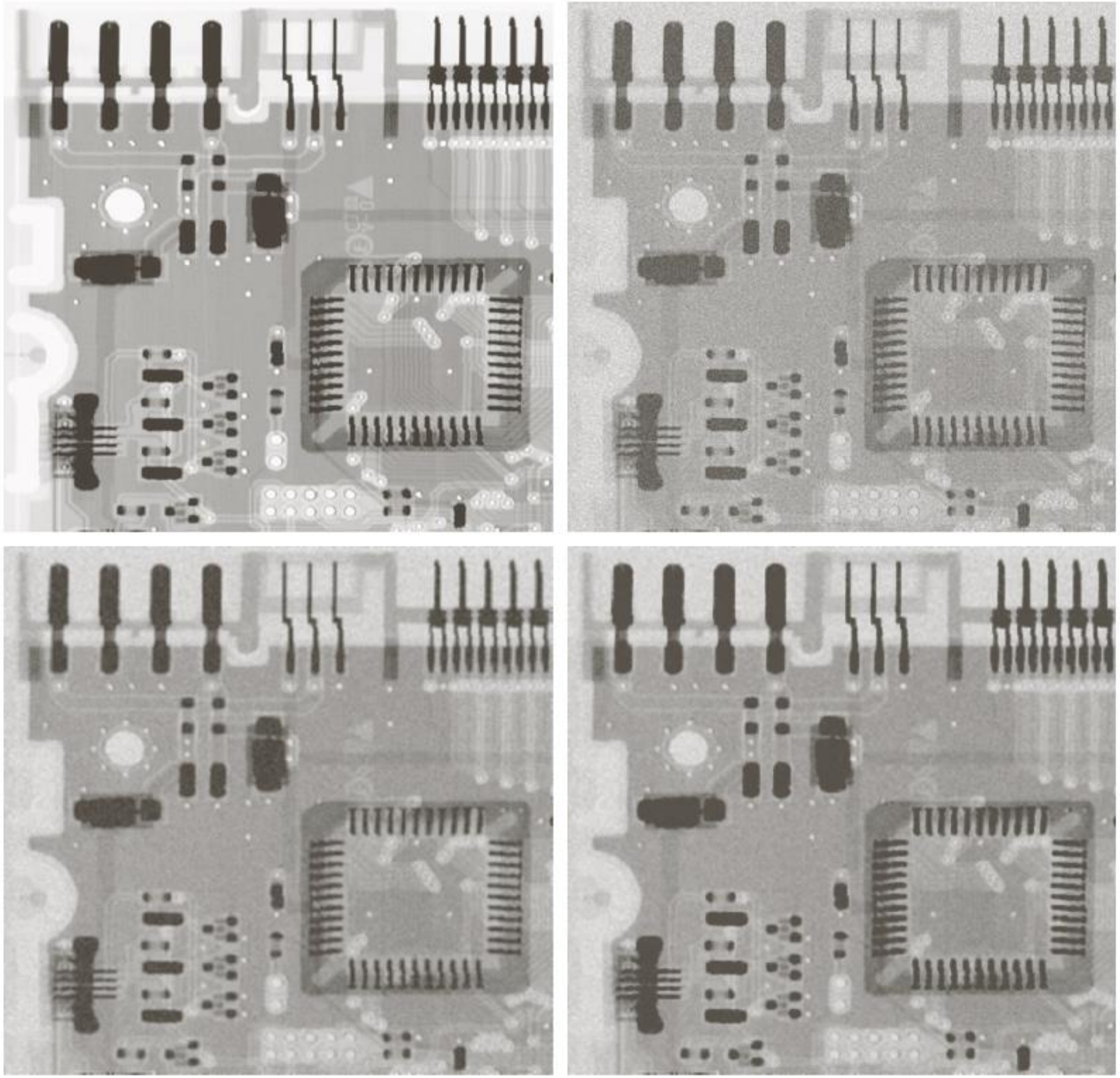


FIGURE 5.7
(a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3×3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filtering: Mean Filters (4)

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.

It is well suited for reducing the effects of **salt-and-pepper noise**. Q>0 for pepper noise and Q<0 for salt noise.

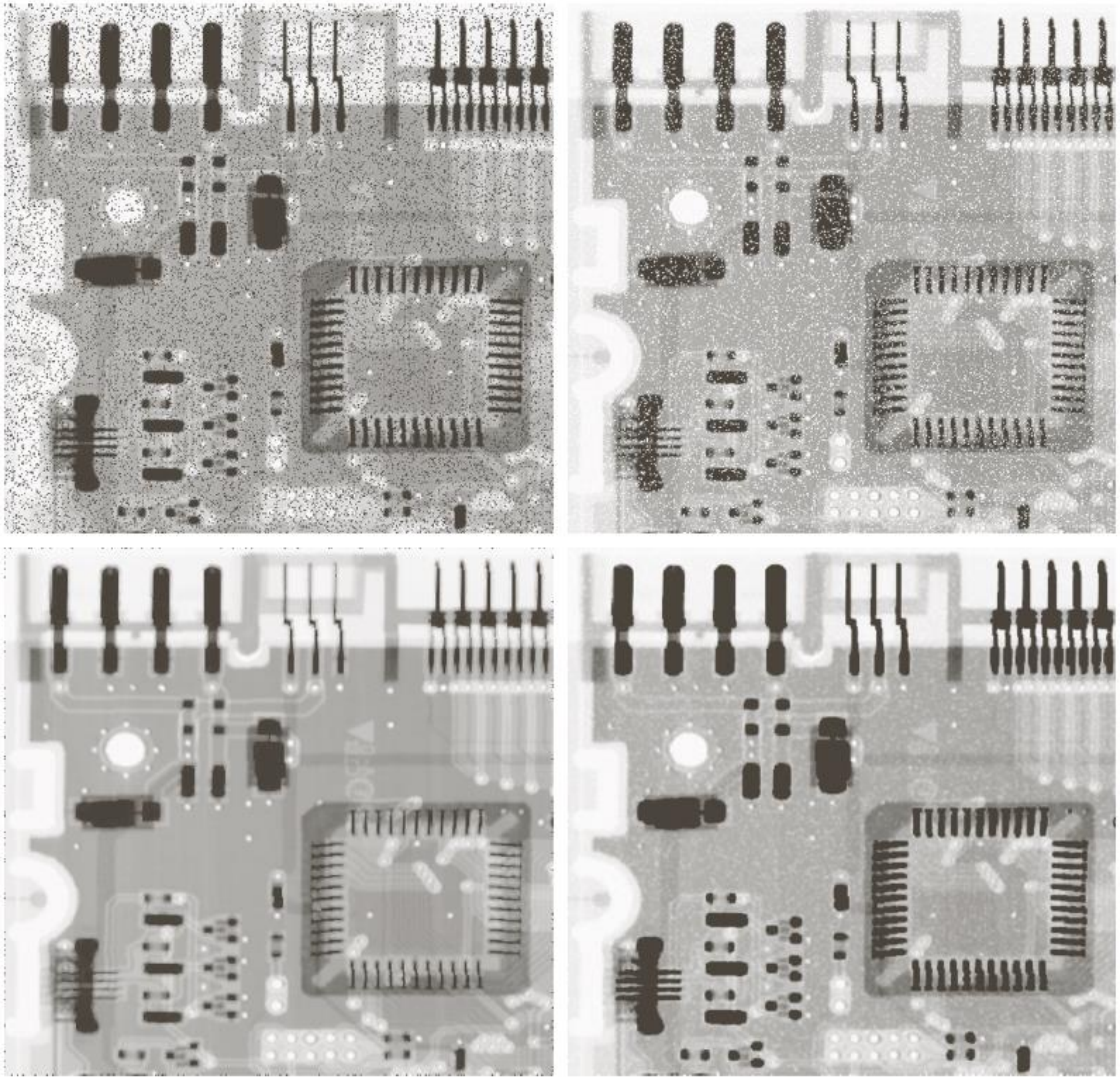
Spatial Filtering: Mean Filters (3)

Harmonic mean filter

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

It works well for **salt noise**, but **fails for pepper noise**.
It does well also with other types of noise like **Gaussian noise**.

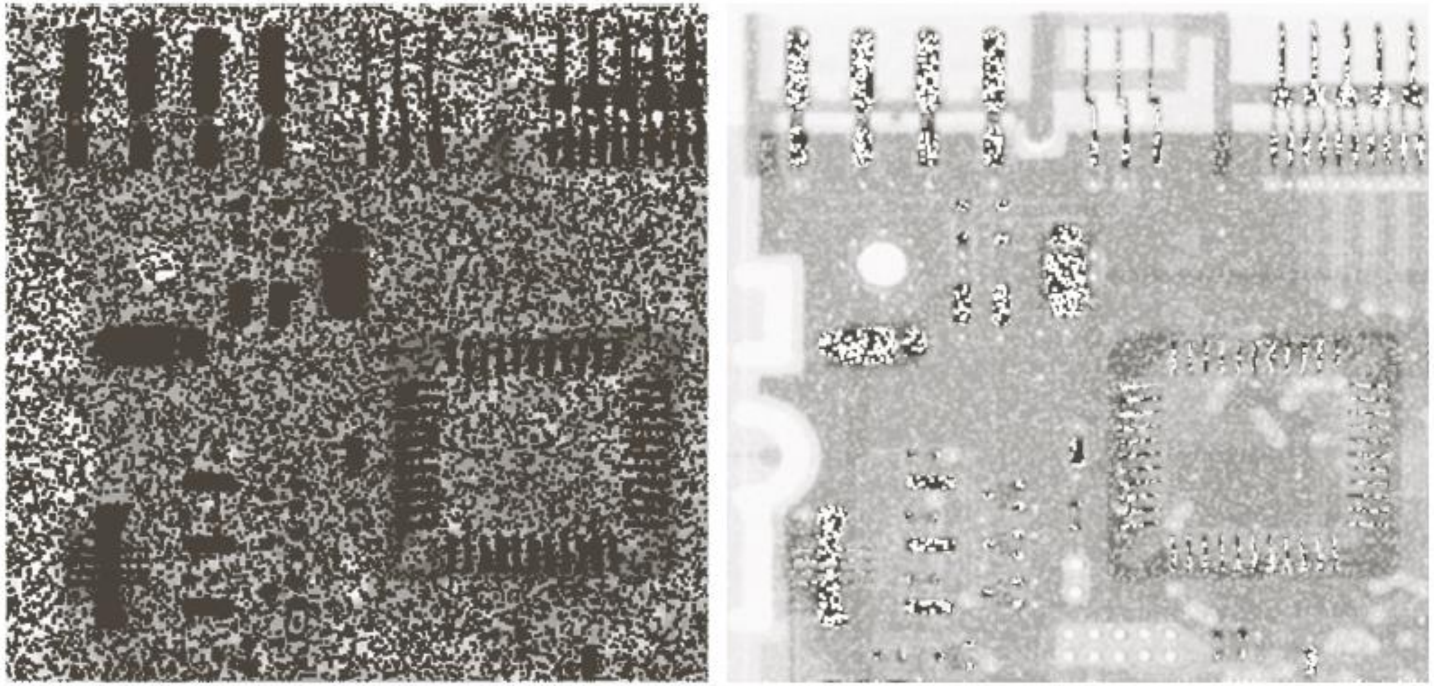
Spatial Filtering: Example (2)



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

Spatial Filtering: Example (3)



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 $Q = 1.5$.

Spatial Filtering: Order-Statistic Filters

- Response is based on ranking the values of pixels in the image (within the filter).

Spatial Filtering: Order-Statistic Filters (1)

Median filter

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

- Best known
- Good for statistical noise
- Reduce noise with considerably less blurring effect
- Effective for both bi and unipolar impulse noise

Max filter

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

- 100^{th} percentile
- Ex. Pepper has low value, and hence works well

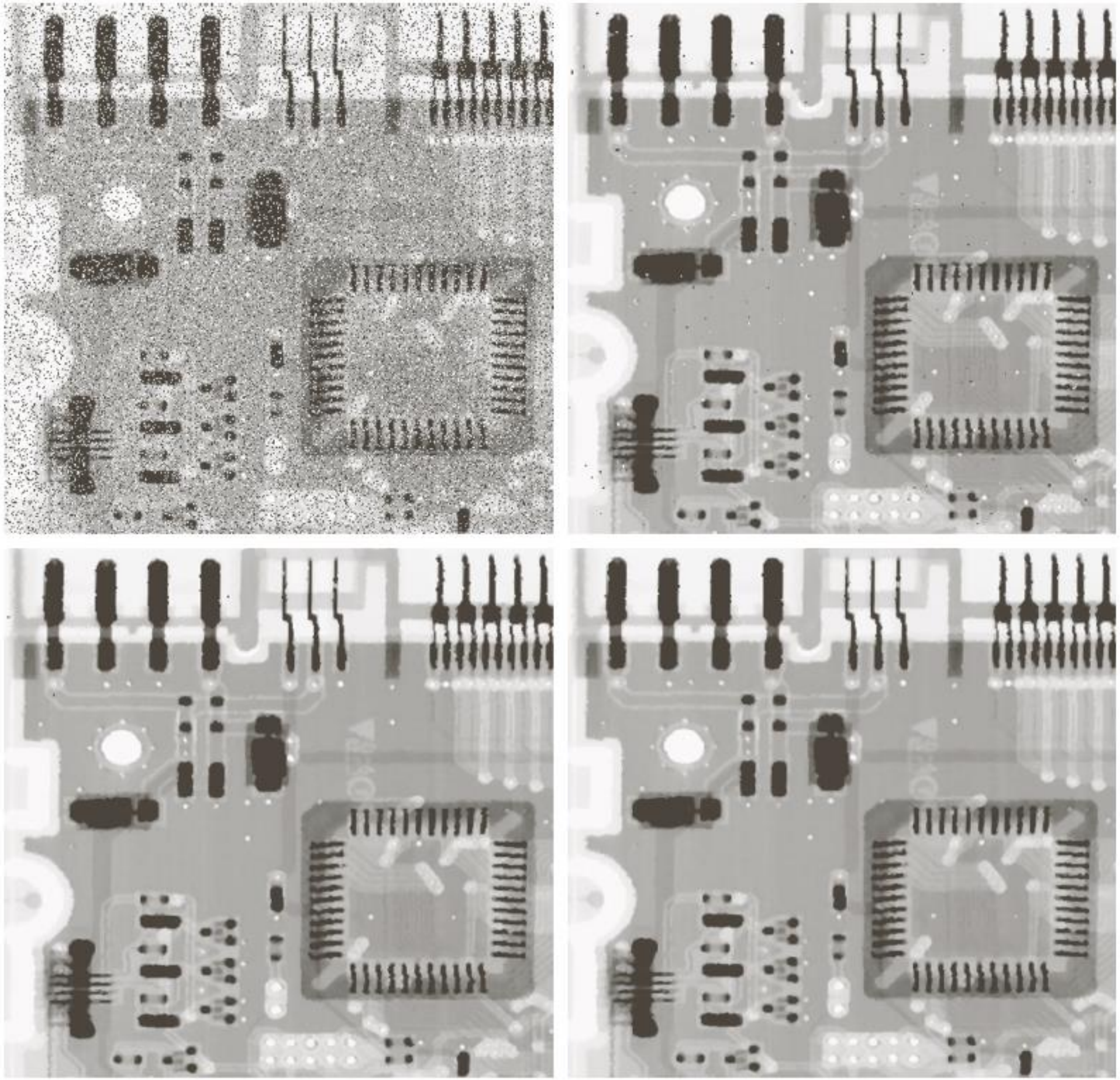
Min filter

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

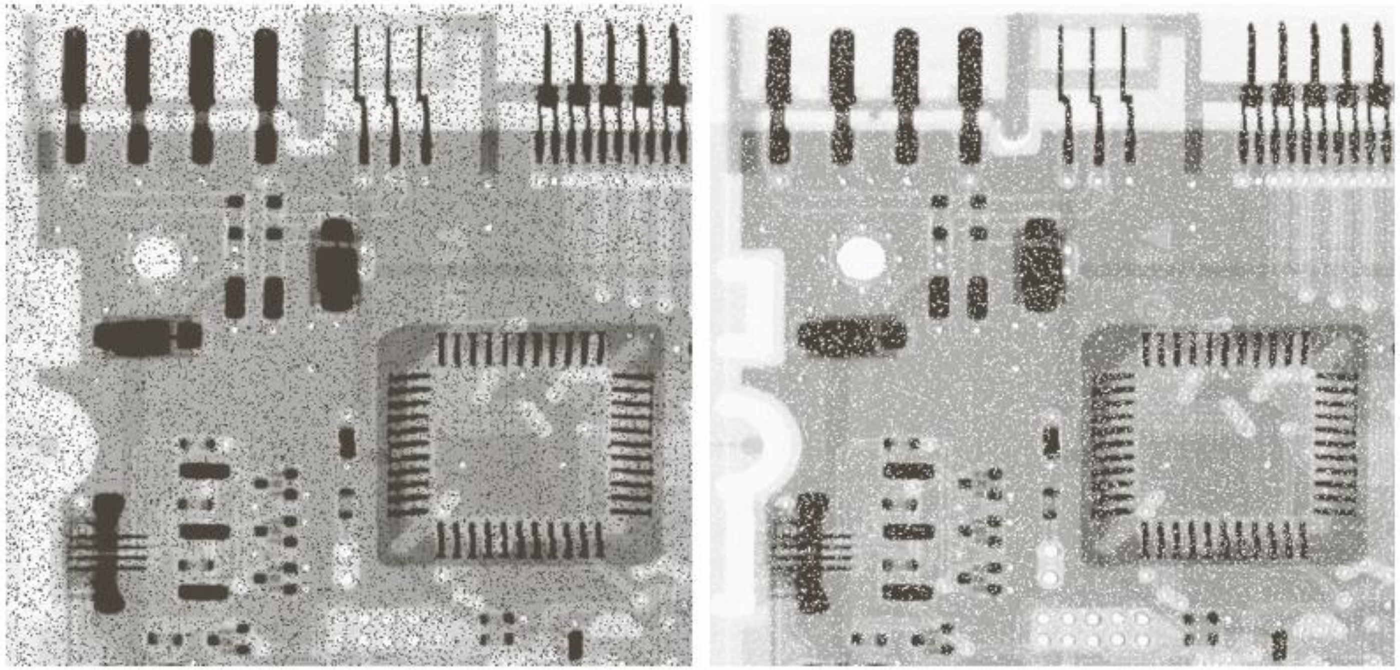
- 0^{th} percentile
- Works well for salt noise

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.

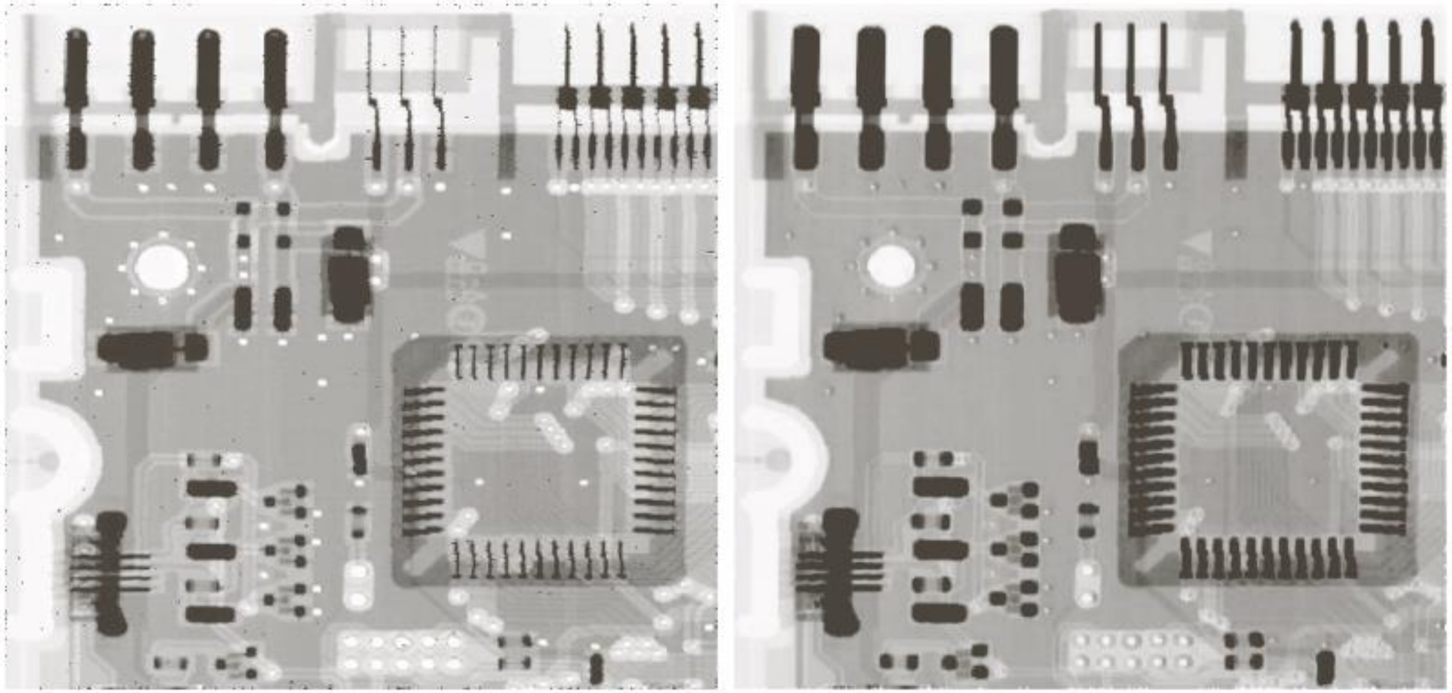


Spatial Filtering: Example (2)



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.



Spatial Filtering: Order-Statistic Filters (2)

Midpoint filter

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

- Combines order statistics and averaging
- Works well for random noise, like Gaussian, and uniform

Spatial Filtering: Order-Statistic Filters (3)

Alpha-trimmed mean filter

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

We delete the $d/2$ lowest and the $d/2$ highest intensity values of $g(s,t)$ in the neighborhood S_{xy}. Let $g_r(s,t)$ represent the remaining $mn-d$ pixels.

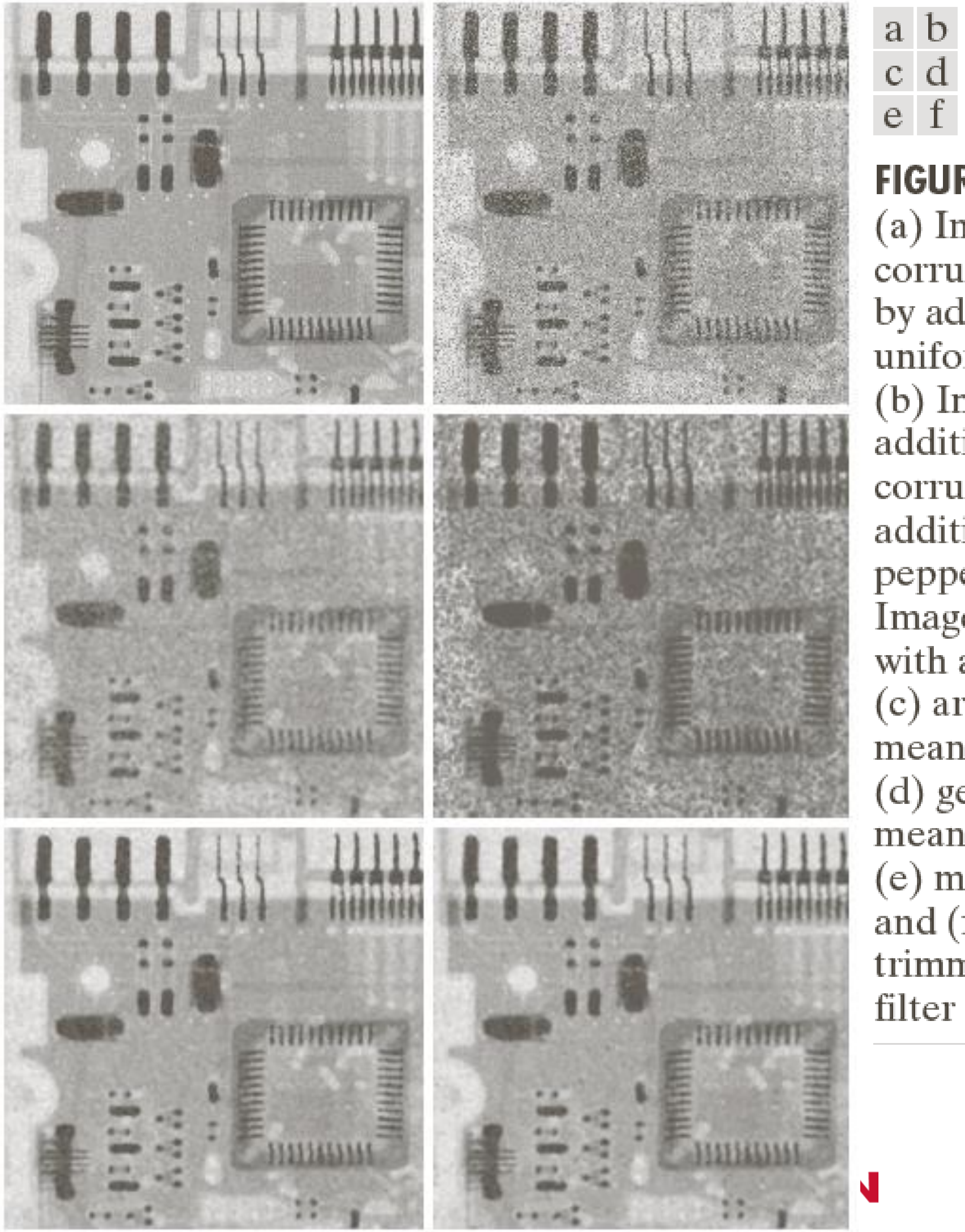


FIGURE 5.12 (a) Image corrupted by additive uniform noise. (b) Image additionally corrupted by additive salt-and-pepper noise. Image (b) filtered with a 5×5; (c) arithmetic mean filter; (d) geometric mean filter; (e) median filter; and (f) alpha-trimmed mean filter with $d = 5$.

Spatial Filtering: Adaptive Filters (1)

Adaptive filters

The behavior changes based on statistical characteristics of the image inside the filter region defined by the mxn rectangular window.

The performance is superior to that of the filters discussed

Trade-off: increased complexity

Adaptive Filters:

Adaptive, Local Noise Reduction Filters (1)

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) m_L, the local mean of the pixels in S_{xy};

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

Adaptive Filters:

Adaptive, Local Noise Reduction Filters (2)

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:

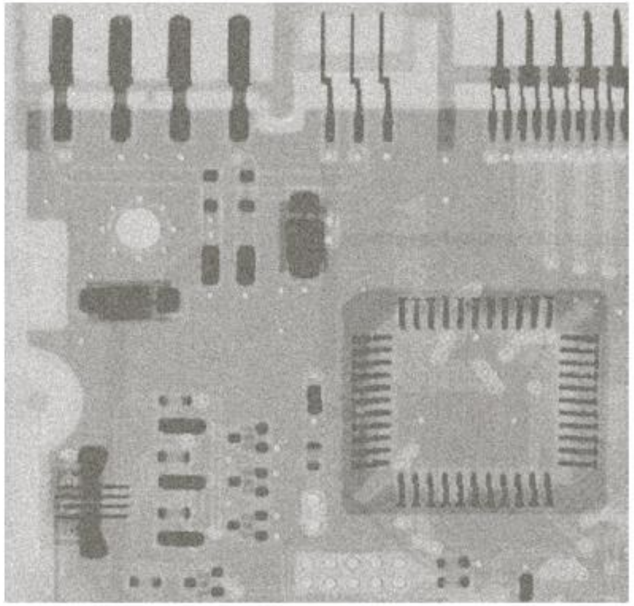
Adaptive, Local Noise Reduction Filters (3)

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

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

a	b
c	d

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



Adaptive Filters:

Adaptive Median Filters (1)

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 (2)

The adaptive median-filtering works in two stages:

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}

Adaptive Filters:

Adaptive Median Filters (2)

The adaptive median-filtering works in two stages:

Stage A:

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

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}

The median filter output is an impulse or not

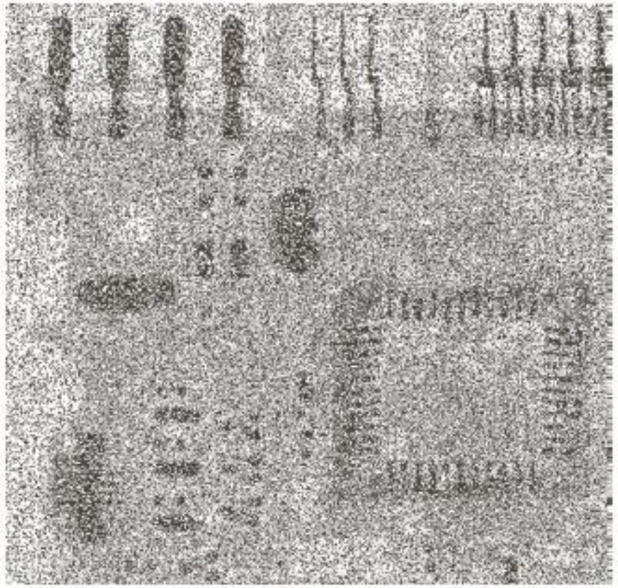
Stage B:

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

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 $P_a = P_b = 0.25$. (b) Result of filtering with a 7×7 median filter. (c) Result of adaptive median filtering with $S_{max} = 7$.