Image Restoration

FYS-2010-1 25V Image Analysis
by Elisabeth Wetzer

Overview

- **What is Image Restoration?**
- **Denoising**
- **Random Noise Models**
- **Denoising by Spatial Filtering**
  - Mean Filters
  - Order-statistic Filters (e.g. median filter)
  - Adaptive Filters
- **Denoising in Frequency Domain**
  - Band Reject/Pass
  - Notch
  - Optimum Notch Filtering

Image Restoration

- **Image Enhancement vs Restoration**
  - Enhancement: subjective
  - Restoration: objective w.r.t. some measure, uses a-priori knowledge about degradation
- **Principle**
  - Recover an image that has been degraded and/or is noisy
- **Methods**
  - Often involves optimization to reach optimum w.r.t. the chosen goodness criterion
  - Filtering (spatial and in frequency)

Image Restoration

- **Spatial Domain**
  - $g(x, y) = h(x, y) * f(x, y) + \eta(x, y)$
- **Frequency Domain**
  - $G(u, v) = H(u, v)F(u, v) + N(u, v)$
- **Assumption**
  - Position invariant system/filter
  - Linear system

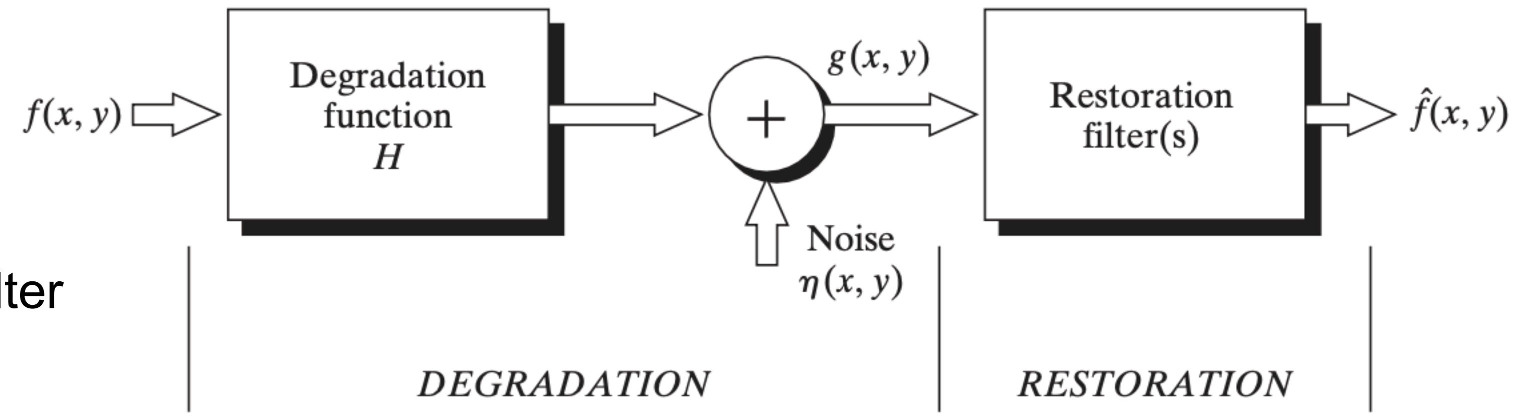


Image Restoration

- **Spatial Domain**
  - $g(x,y) = h(x,y) * f(x,y) + \boxed{\eta(x,y)}$

Additive Noise Terms

- **Frequency Domain**
  - $G(u,v) = H(u,v)F(u,v) + \boxed{N(u,v)}$

- **Assumption**
  - Position invariant system/filter
  - Linear system

Image Restoration

- **Spatial Domain**
  - $$g(x,y) = \boxed{h(x,y)} * f(x,y) + \eta(x,y)$$
- **Frequency Domain**
  - $$G(u,v) = \boxed{H(u,v)} F(u,v) + N(u,v)$$

Degradation Function

- **Assumption**
  - Position invariant system/filter
  - Linear system

Image Restoration: Denoising only today!

- **Spatial Domain**
  - $g(x, y) = \boxed{\cancel{h(x, y)}} * f(x, y) + \eta(x, y)$
- **Frequency Domain**
  - $G(u, v) = \boxed{\cancel{H(u, v)}} F(u, v) + N(u, v)$
- **Assumption**
  - Position invariant system/filter
  - Linear system

Image Restoration: Denoising

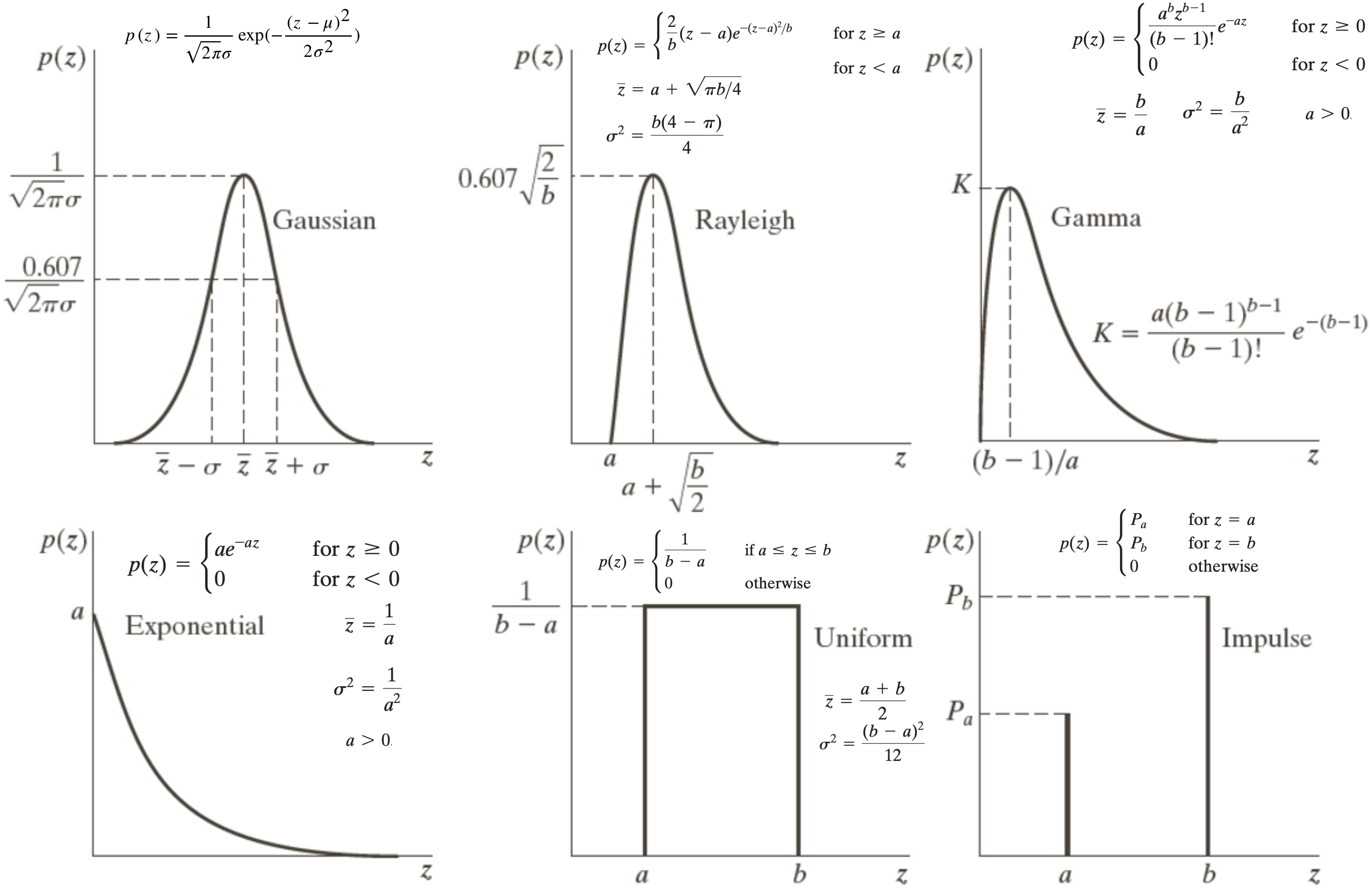
- **Spatial Domain**
  - $g(x, y) = h(x, y) + \eta(x, y)$
- **Frequency Domain**
  - $G(u, v) = H(u, v) + N(u, v)$

$$\boxed{\text{Goal: Find } \eta(x, y) \text{ or } N(u, v)}$$

Random Noise Models

- **Noise Sources**
  - Image acquisition: digitization and/or transmission
  - Influences: sensor temperature, light levels, CCD sensor type,…
- **Noise Probability Density Functions**
  - Spatial noise descriptors describe the statistical behavior of gray-level values in the noise component of the model as a random variable through its probability density function (pdf)
- **Additive Noise**
  - Drawing a random value from the noise distribution for each pixel and add it to intensity

Common Random Noise Models



Common Random Noise Models

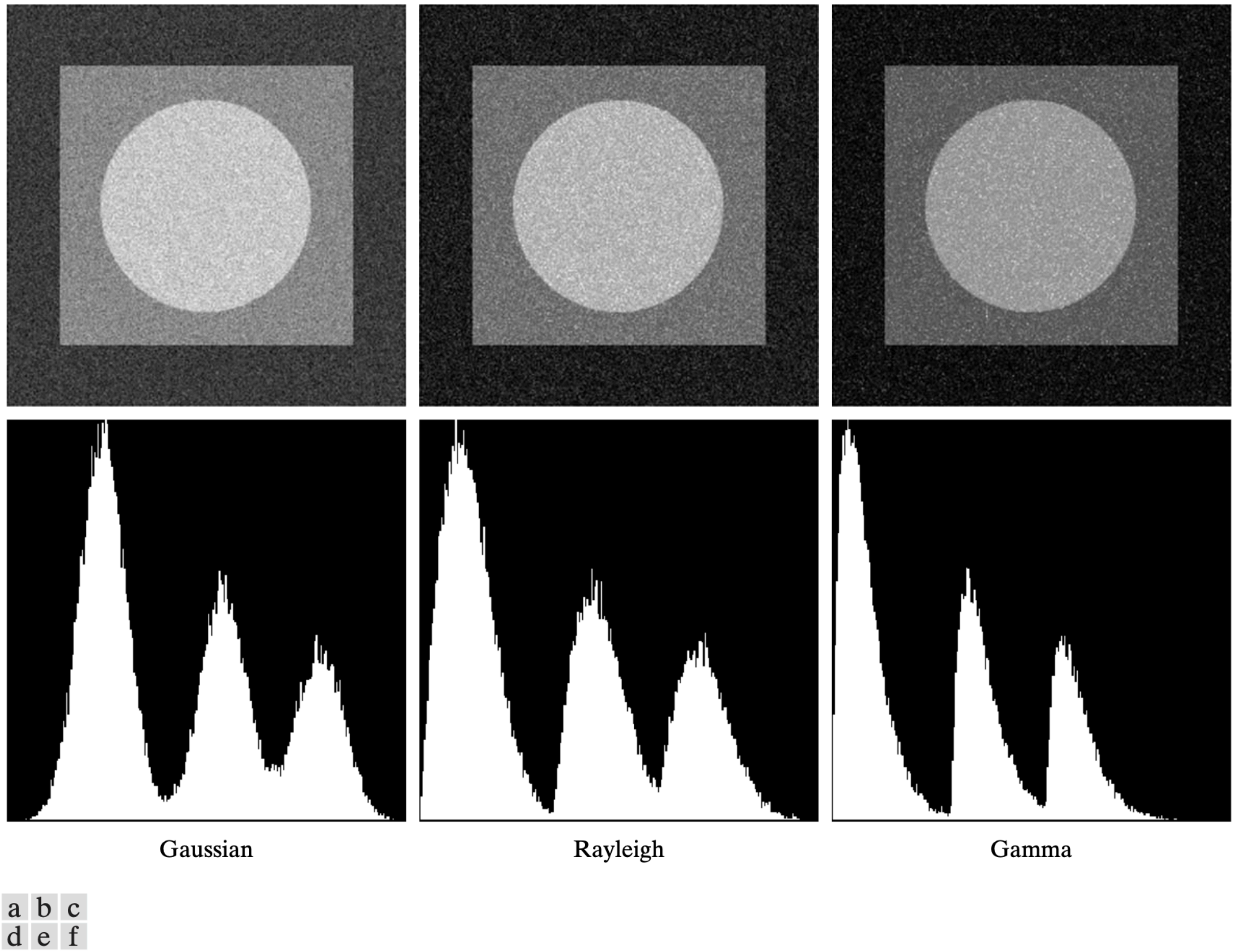


FIGURE 5.4 Images and histograms resulting from adding Gaussian, Rayleigh, and gamma noise to the image in Fig. 5.3.

How to find a suitable noise model?

- **Find a homogeneous area in an image**
- **Assess histogram**
- **Estimate the mean and variance**

Denoising by Spatial Filtering

- **Filter Types**
  - Mean Filters
    - Arithmetic (“normal”) Mean

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

    - Geometric Mean

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

    - Harmonic Mean

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

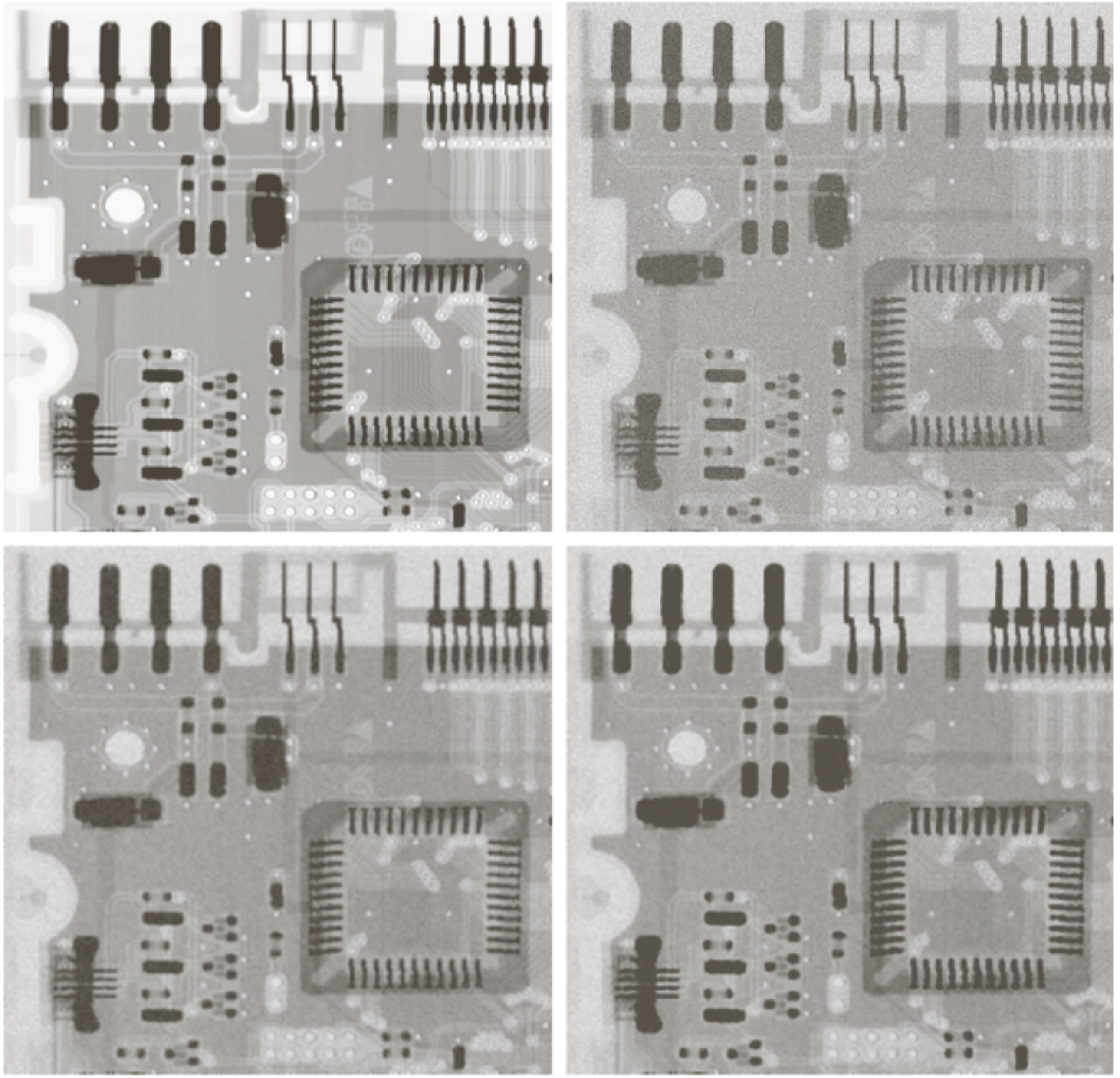
    - Contraharmonic Mean of Order Q

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

  - Order-statistic Filters (e.g. median filter)
  - Adaptive Filters

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**



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.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

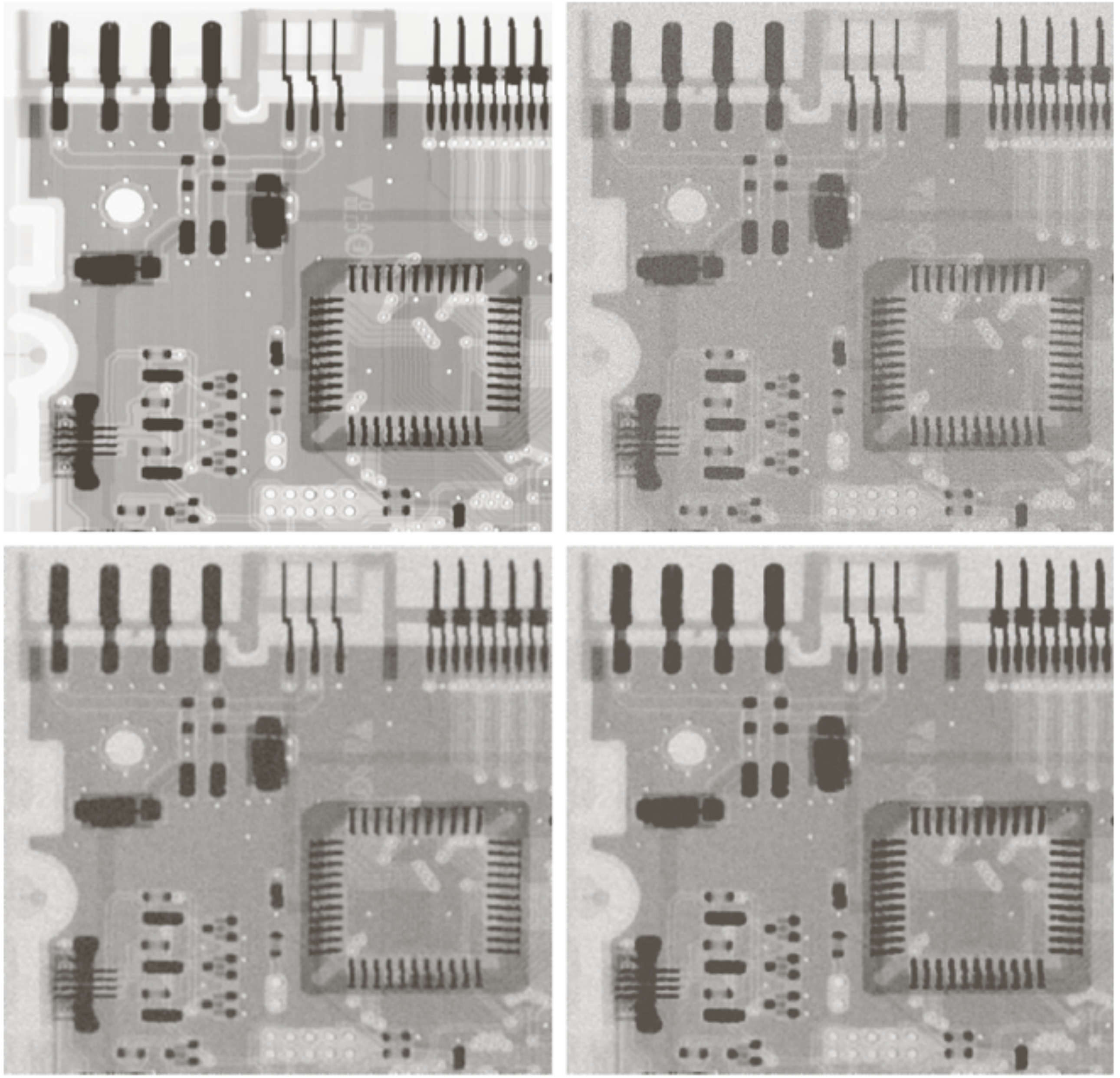


FIGURE 5.7 (a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3 × 3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$n \times n$ Neighborhood, here $n = 3$

$$\begin{bmatrix} \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \\ \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \\ \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \end{bmatrix}$$

$*$

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

	3.6		

(1+2+3+2+1+3+7+9+5)/9

a b
c d

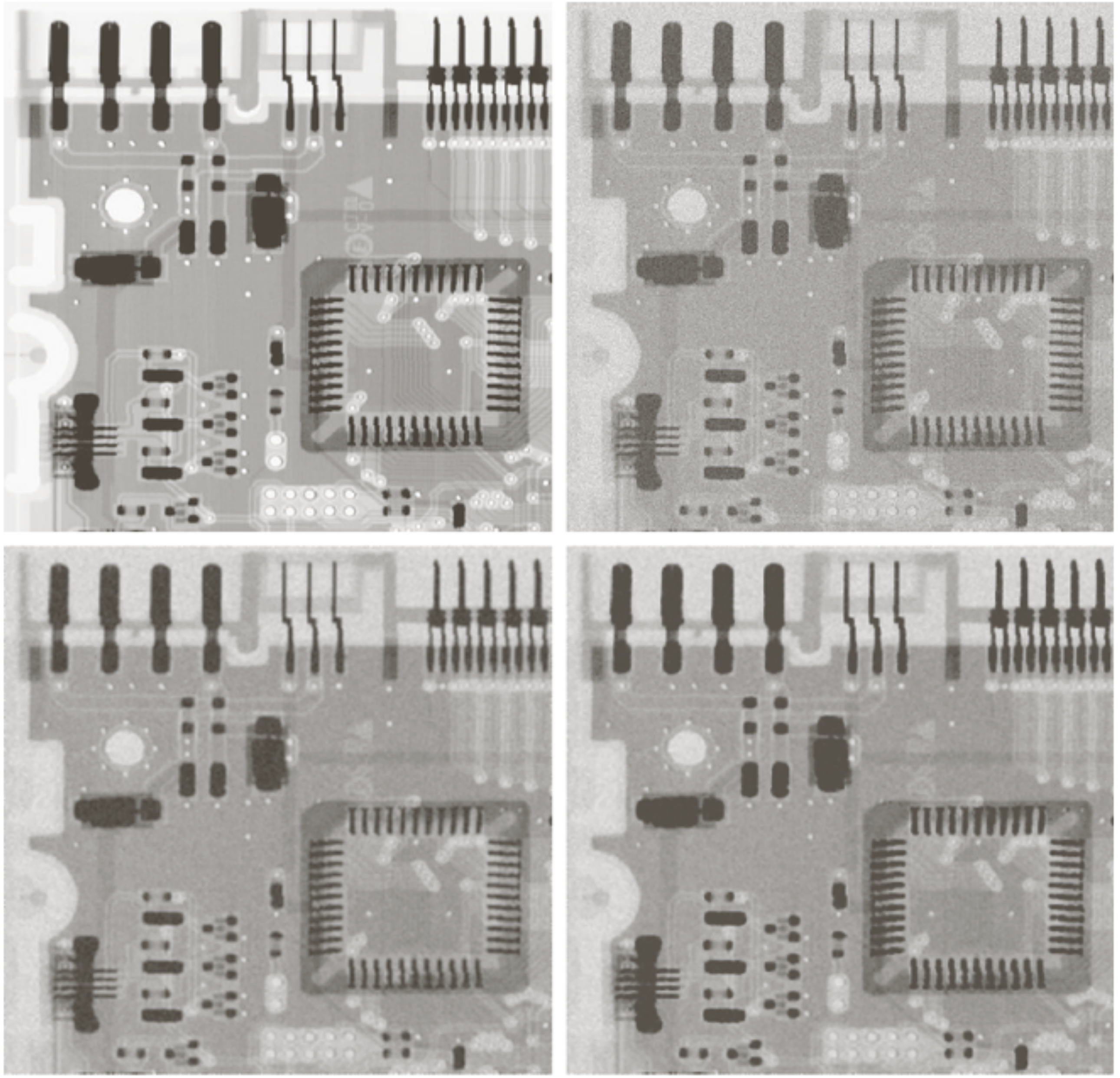


FIGURE 5.7 (a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3×3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

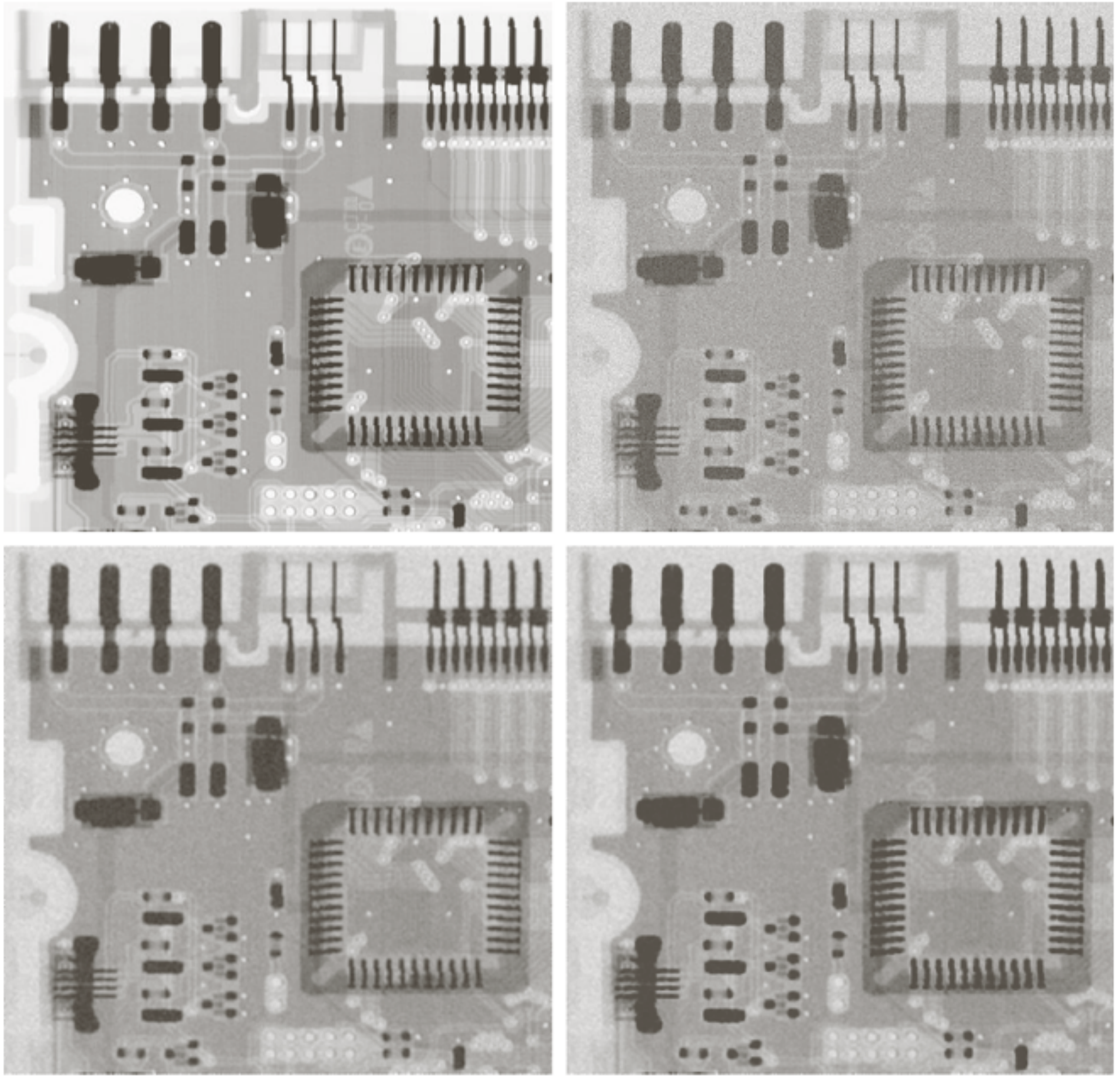
$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

	4		



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.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$n \times n$ Neighborhood, here $n = 3$

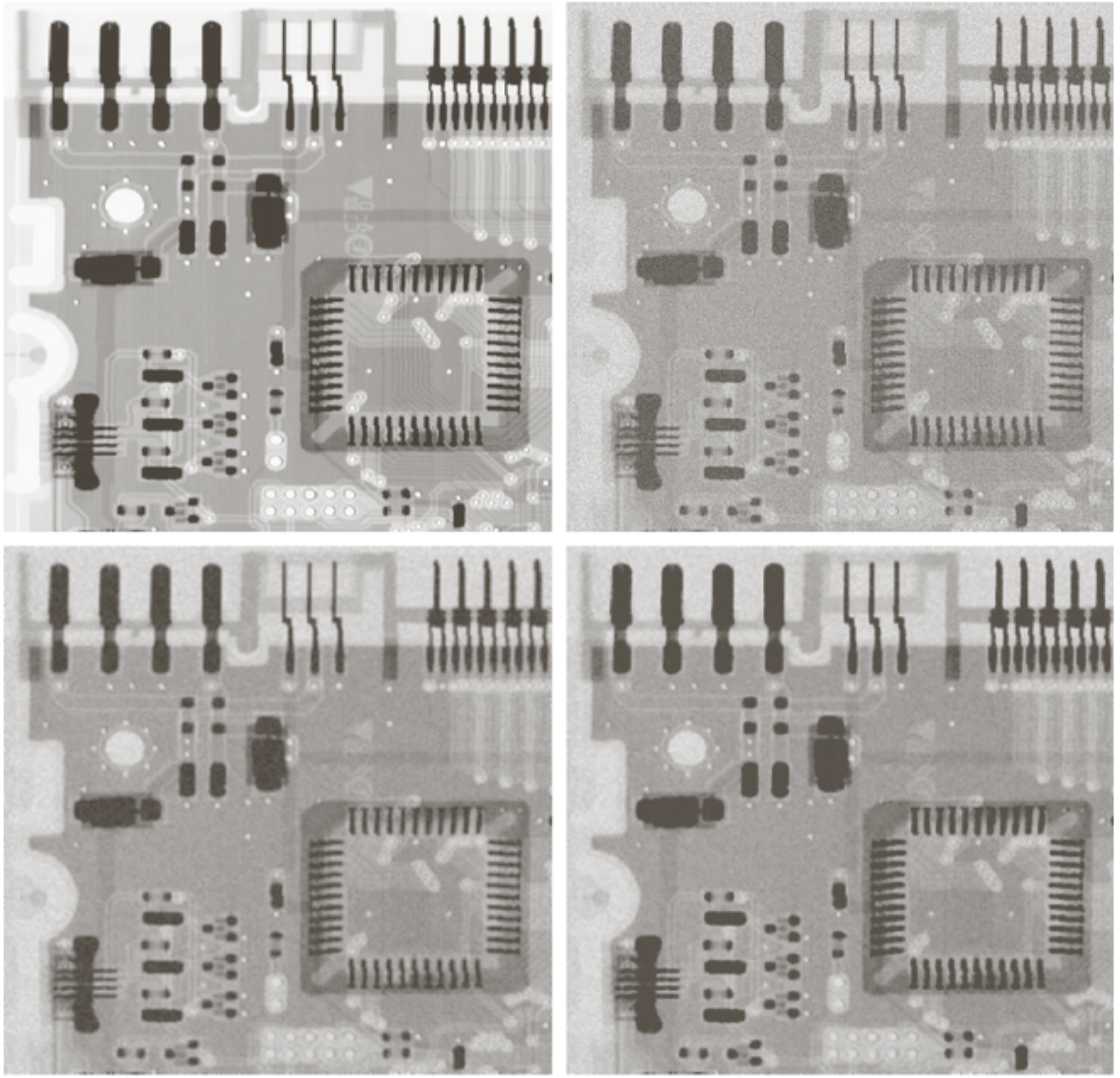
$*$

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

	4	?	

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.)



Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

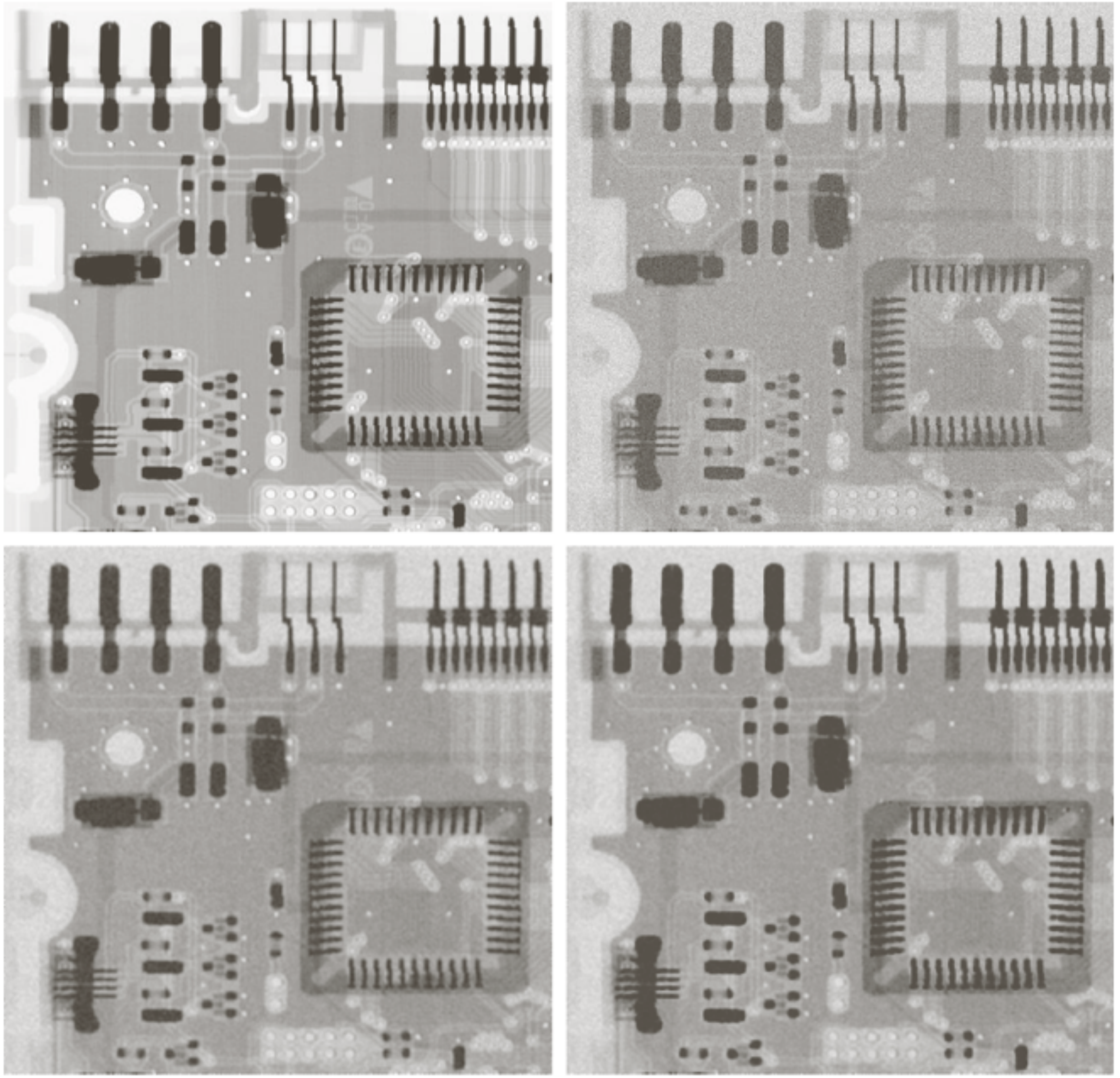
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$n \times n$ Neighborhood, here $n = 3$

$*$

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

	4	4	



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.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	5	1
8	1	2	1
0	3	3	2

?			
	4	4	

a b
c d

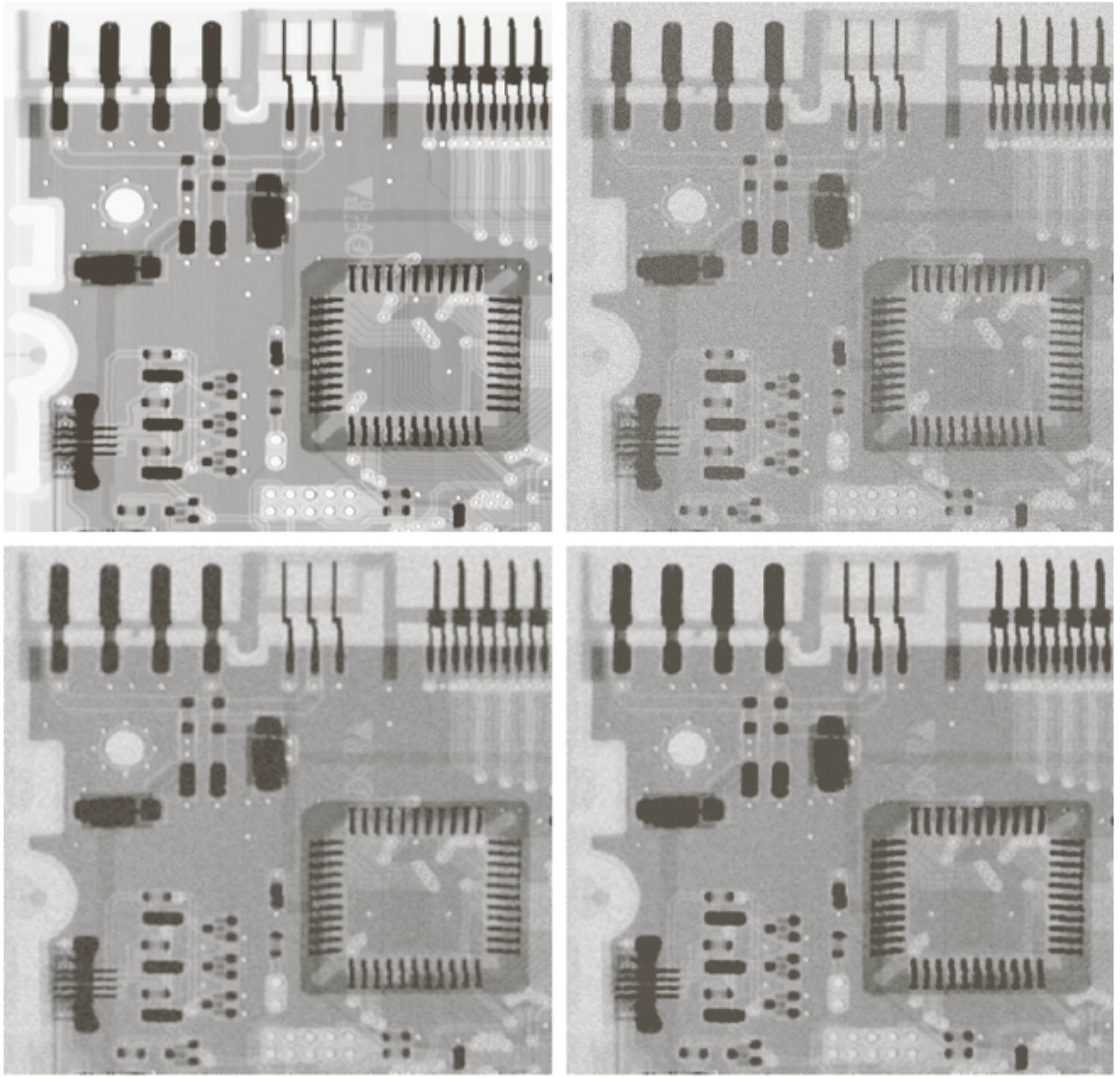


FIGURE 5.7 (a) X-ray image. (b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size 3×3. (d) Result of filtering with a geometric mean filter of the same size. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$n \times n$ Neighborhood, here $n = 3$

$$\begin{bmatrix} \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \\ \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \\ \frac{1}{n^2} & \frac{1}{n^2} & \frac{1}{n^2} \end{bmatrix}$$

$*$

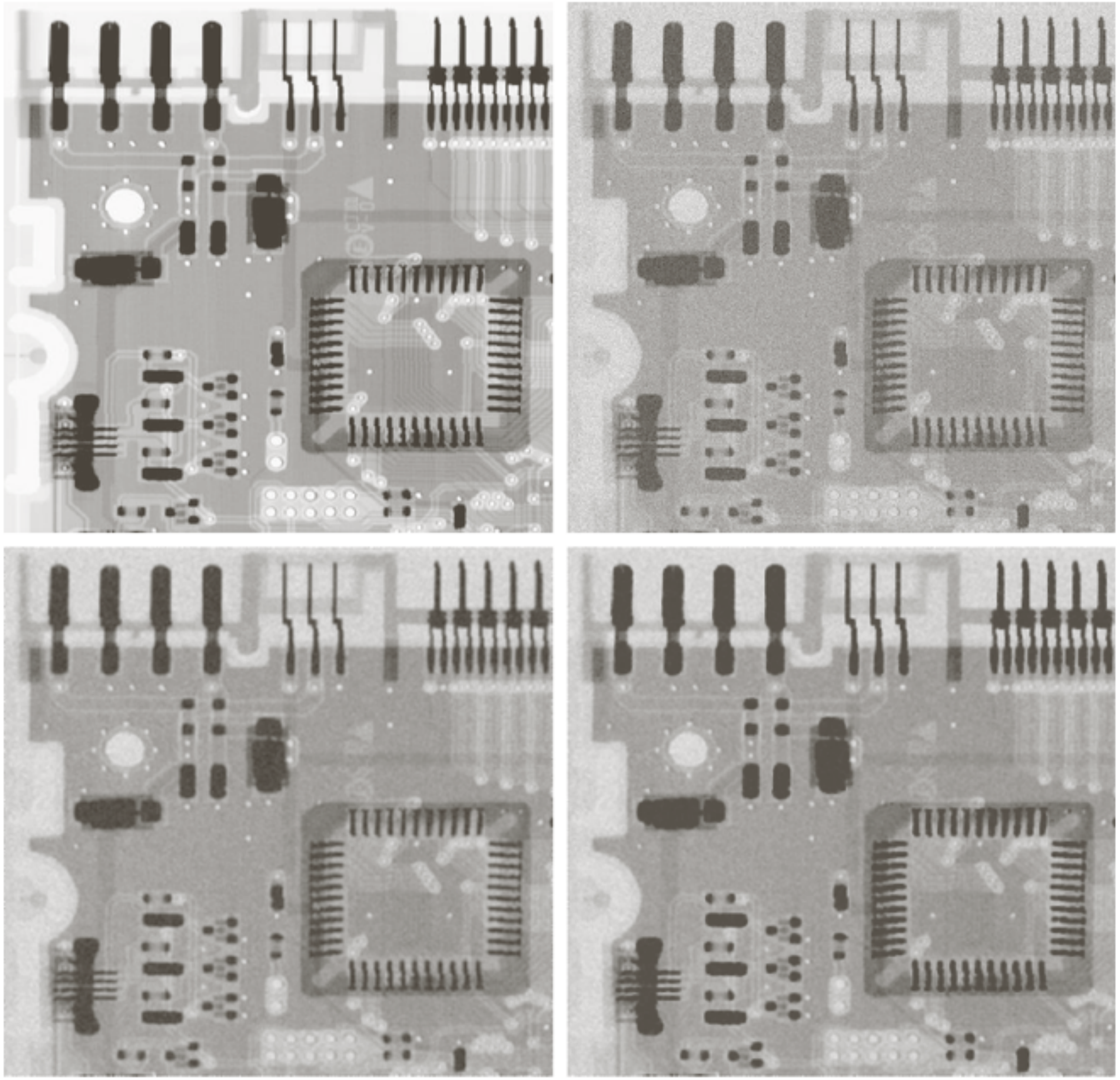
0	0	0		
0	1	2	3	5
0	2	1	3	4
	7	9	5	1
	8	1	2	1
	0	3	3	2

zero-padding

1			
	4	4	

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.)



Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

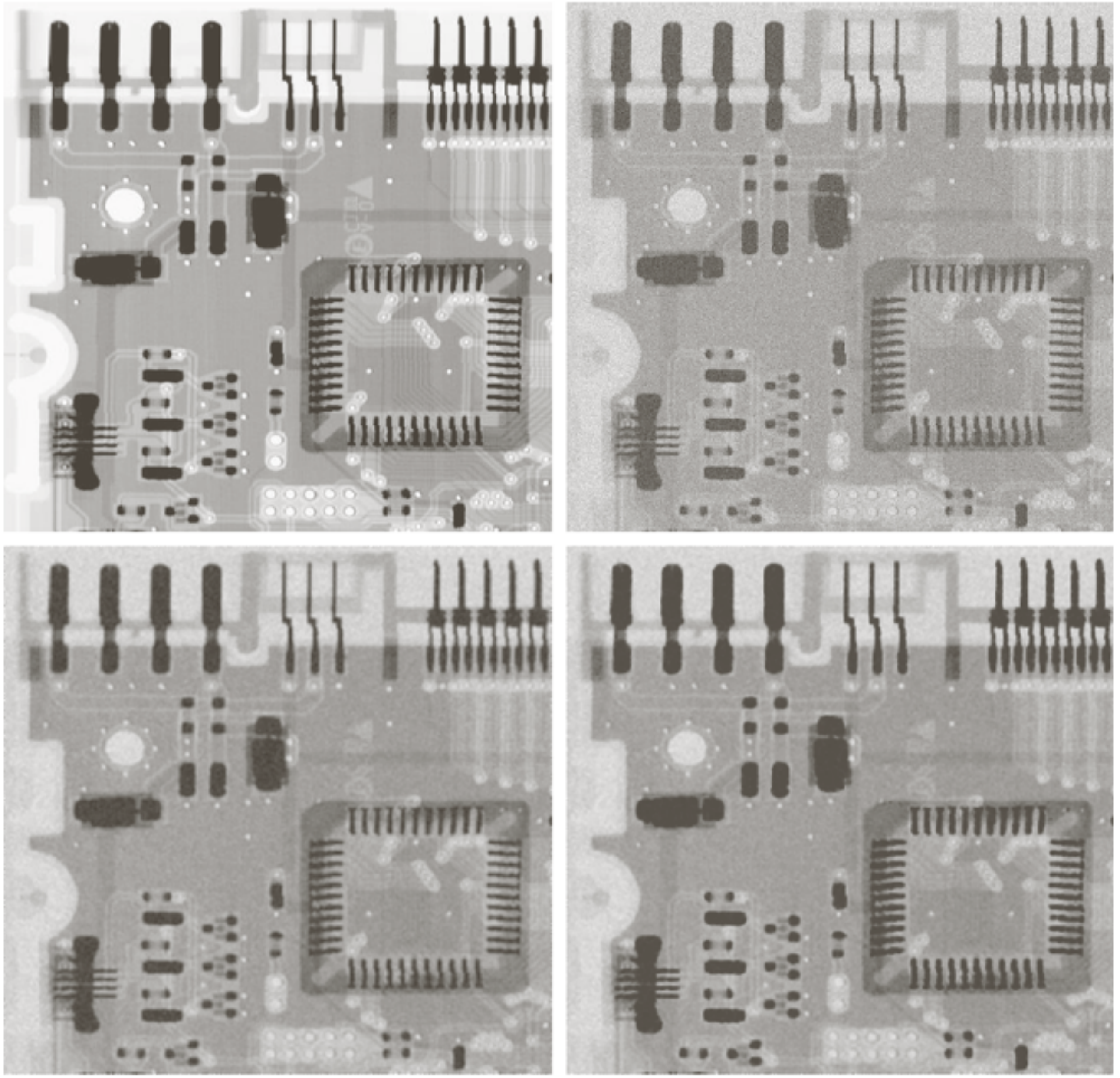
1	1	2		
1	1	2	3	5
2	2	1	3	4
	7	9	5	1
	8	1	2	1
	0	3	3	2

replicate

1			
	4	4	

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.)



Spatial Filters

- **Arithmetic and geometric filters**
- **Generally good on Gaussian or uniform noise**

$n \times n$ Neighborhood, here $n = 3$

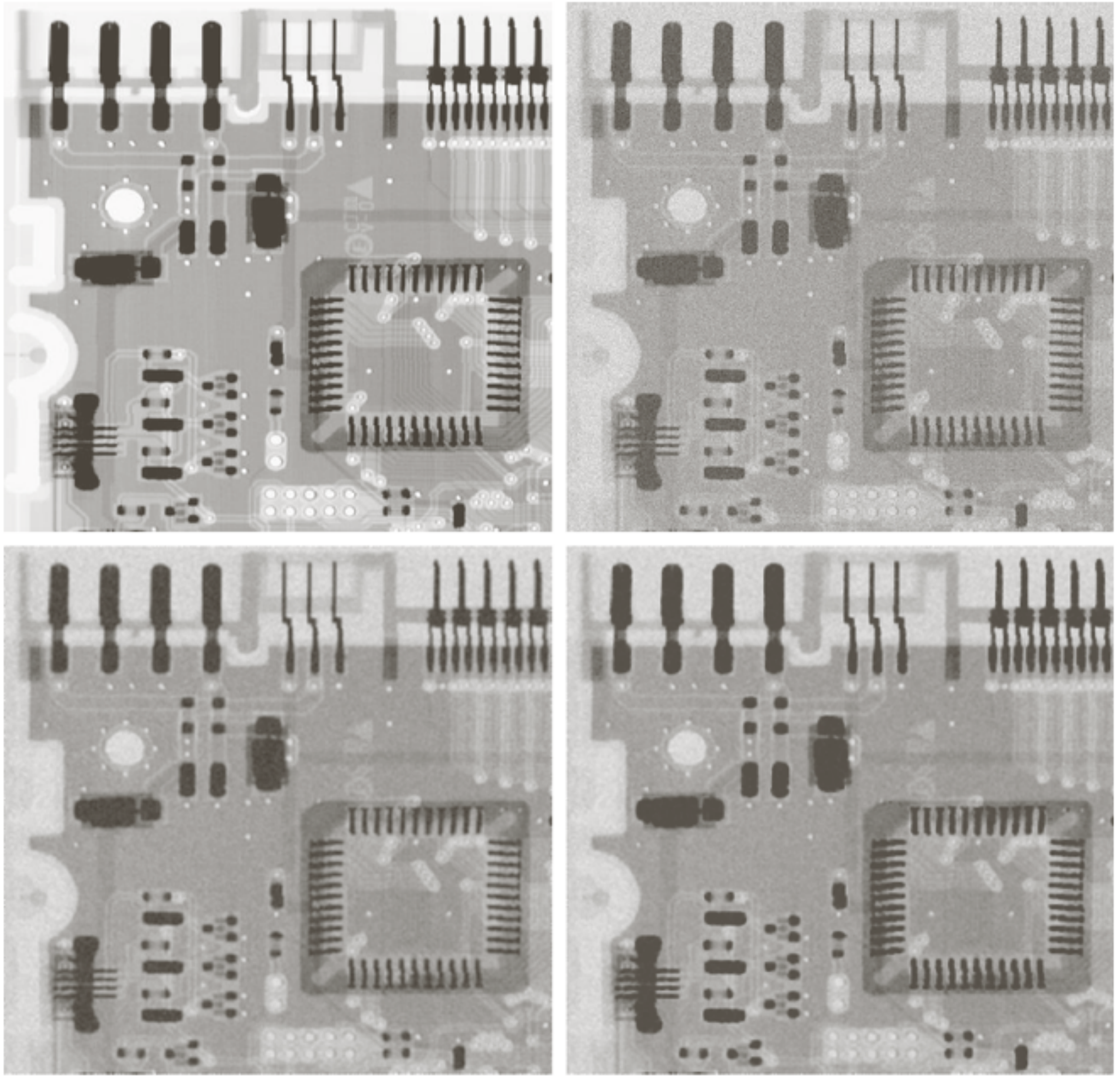
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	1		
2	1	2	3	5
1	2	1	3	4
	7	9	5	1
	8	1	2	1
	0	3	3	2

mirror

1			
	4	4	

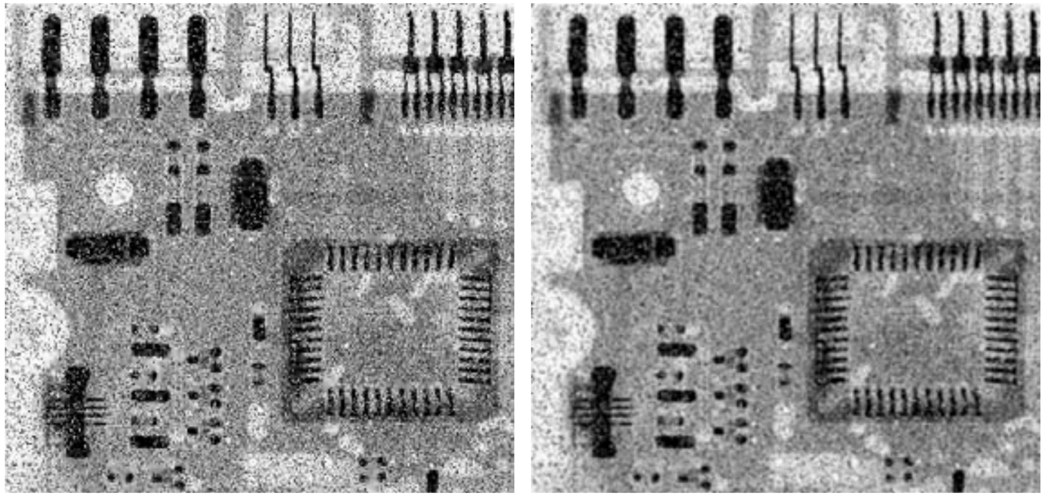


a b
c d

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.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

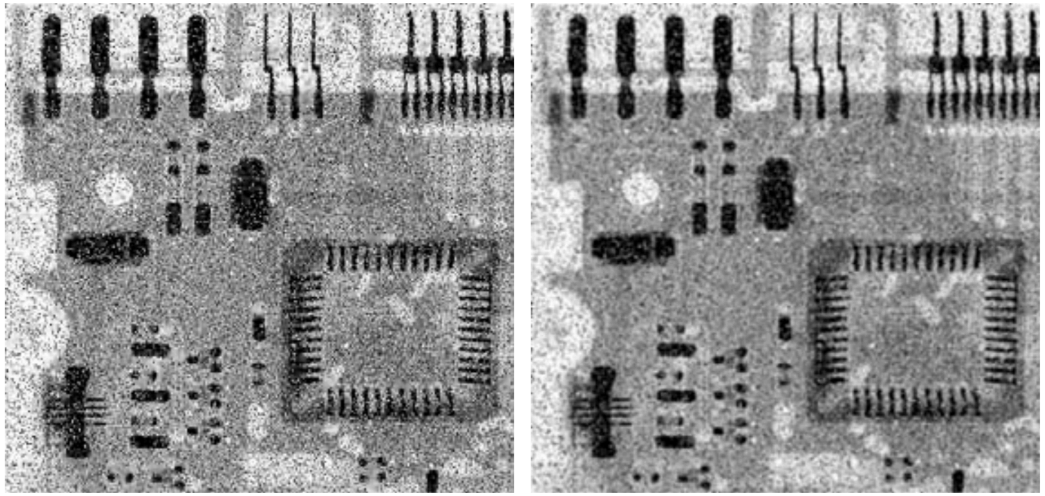
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

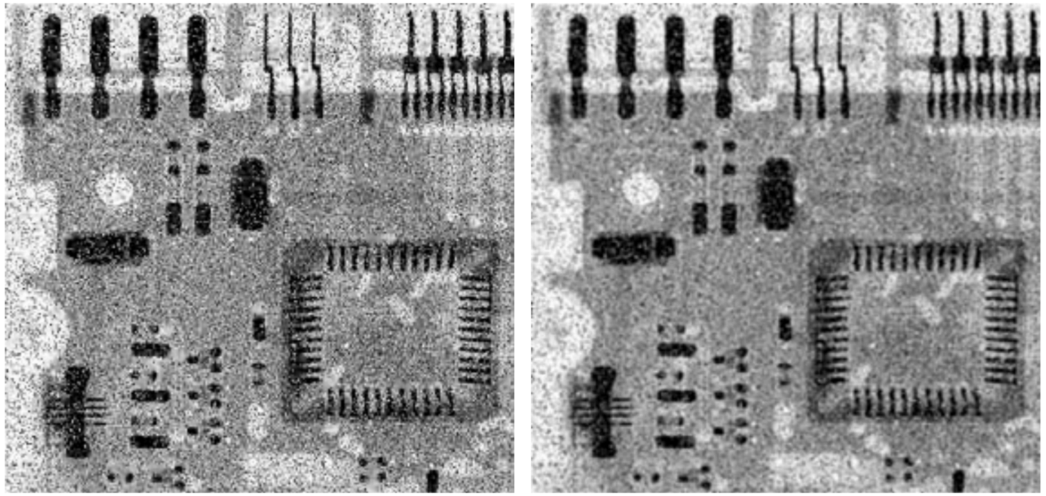
$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31		

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**

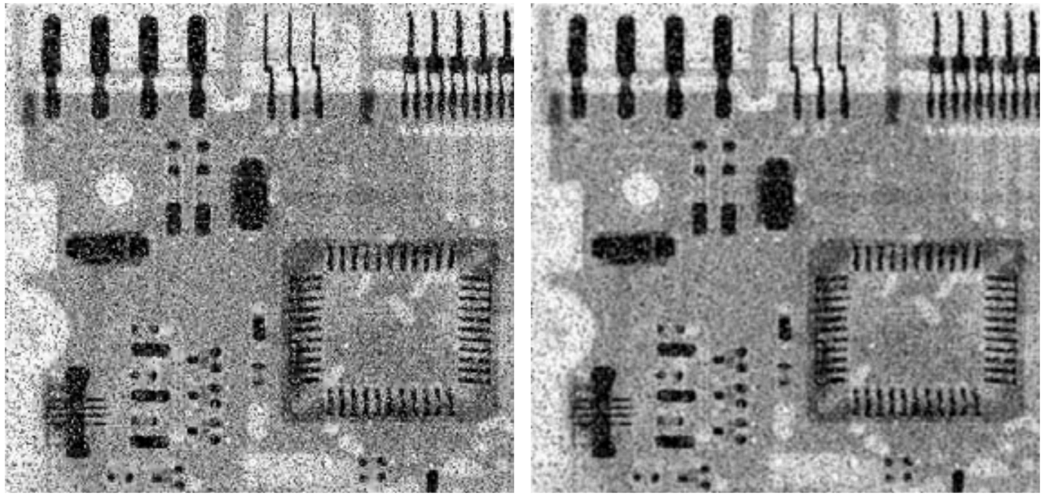
$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	
		31	

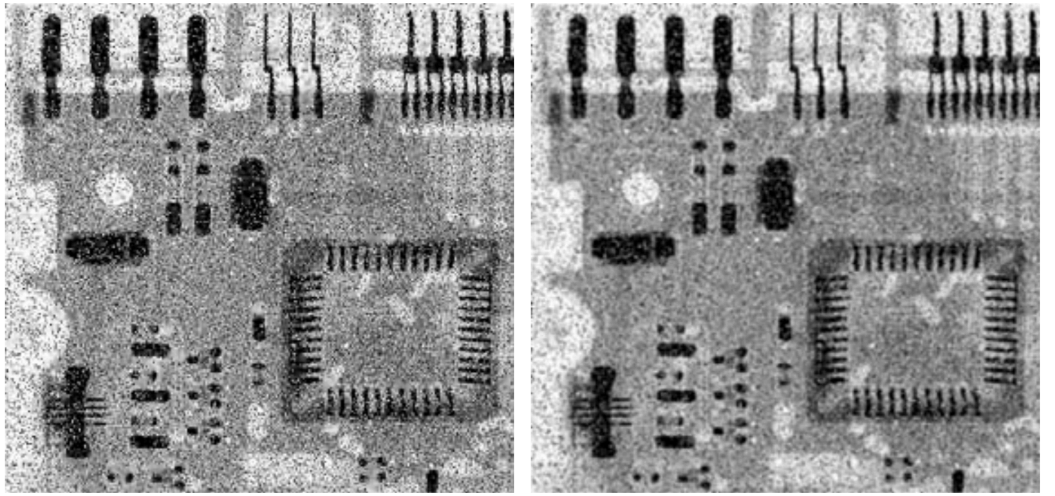


a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

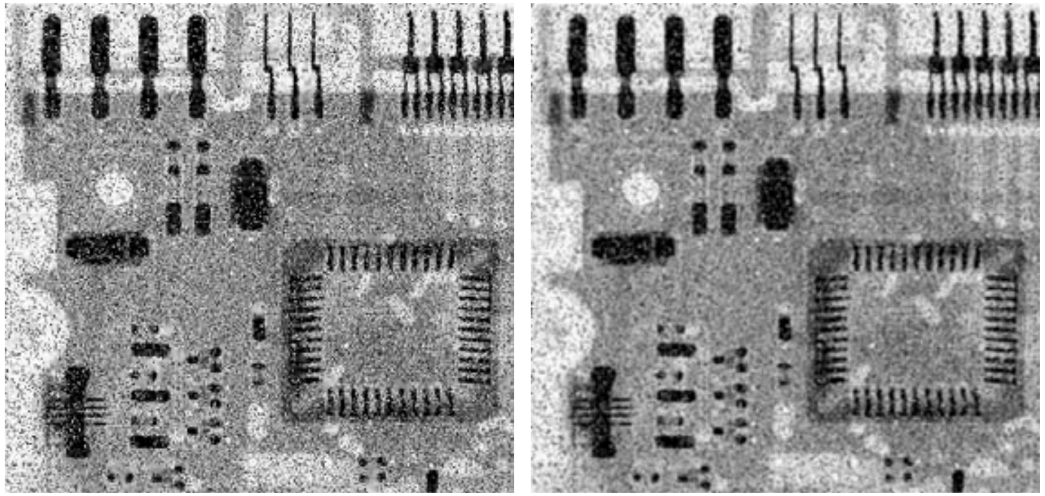
$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	
	32	31	

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

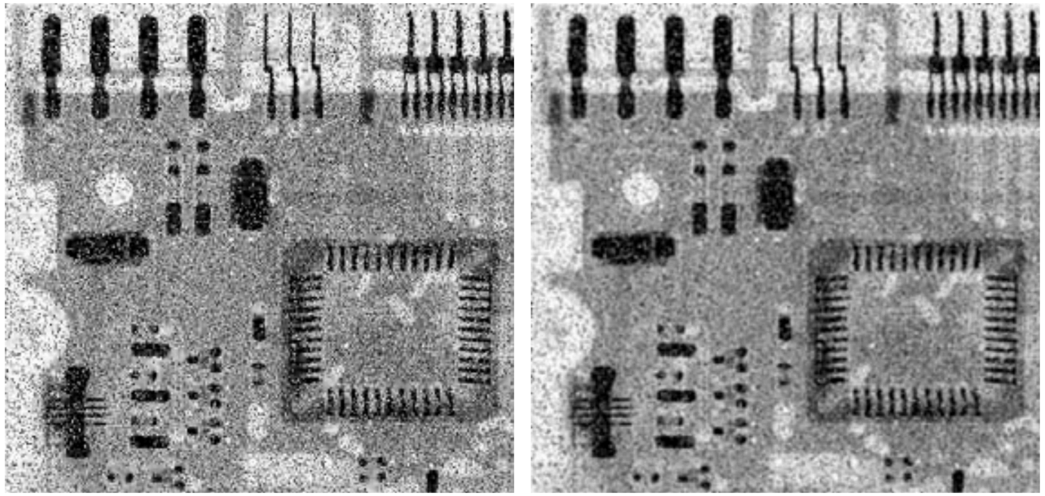
$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	
	32	31	
	32		

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

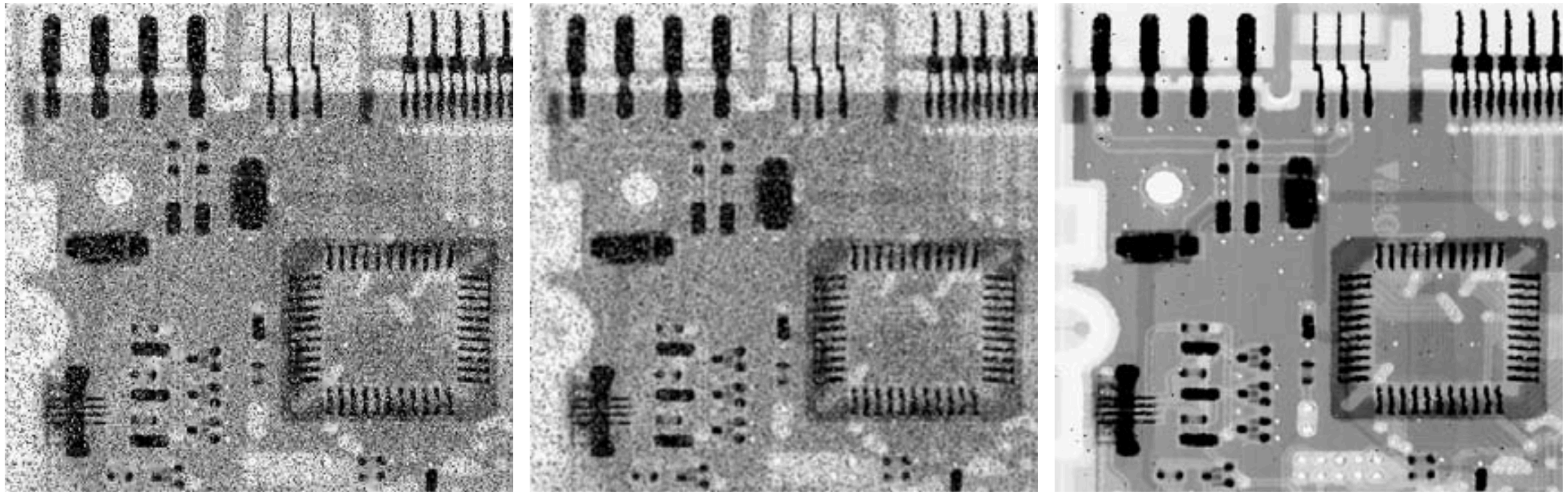
1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	
	32	31	
	32	31	

noise gets smooched out in the entire neighborhood

Spatial Filters

- **Salt & pepper noise**
- **Median filter much better**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

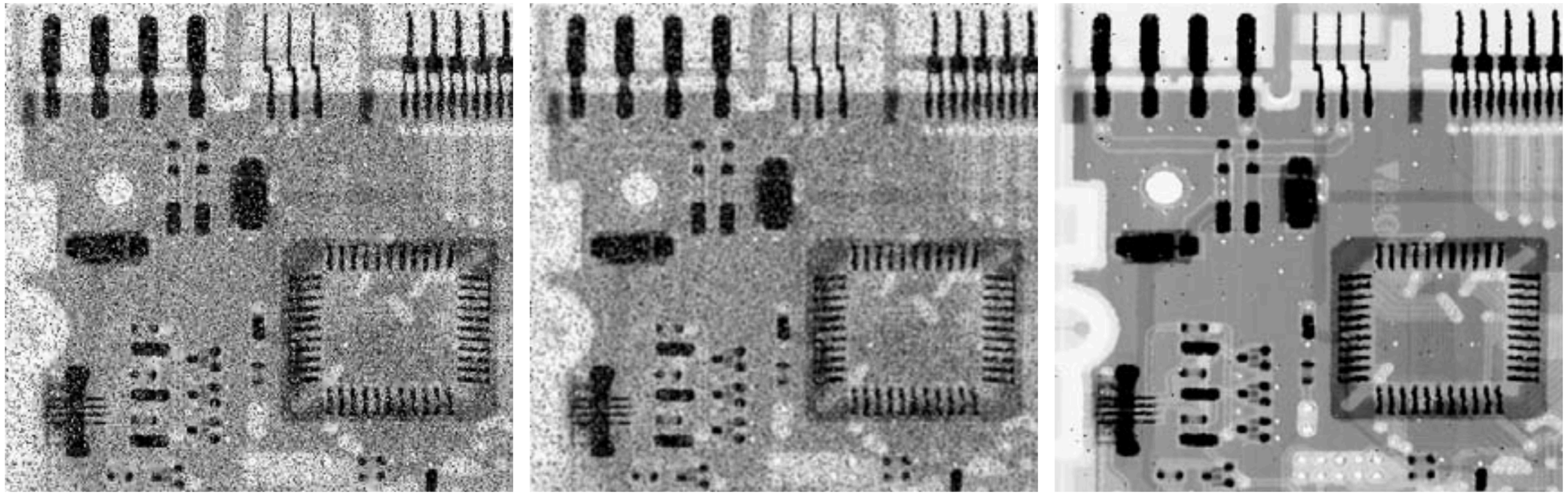
1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	2.5		

(1,1,2,2,3,3,9,255)

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

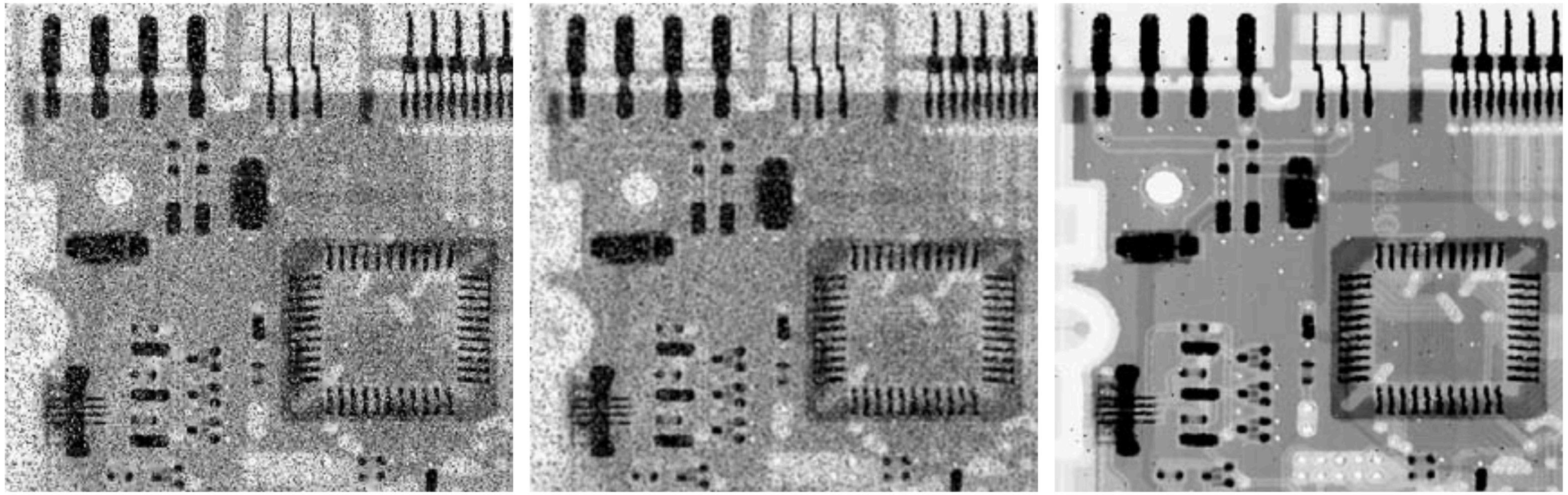
1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	3		

(1,1,2,2,3,3,9,255)

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

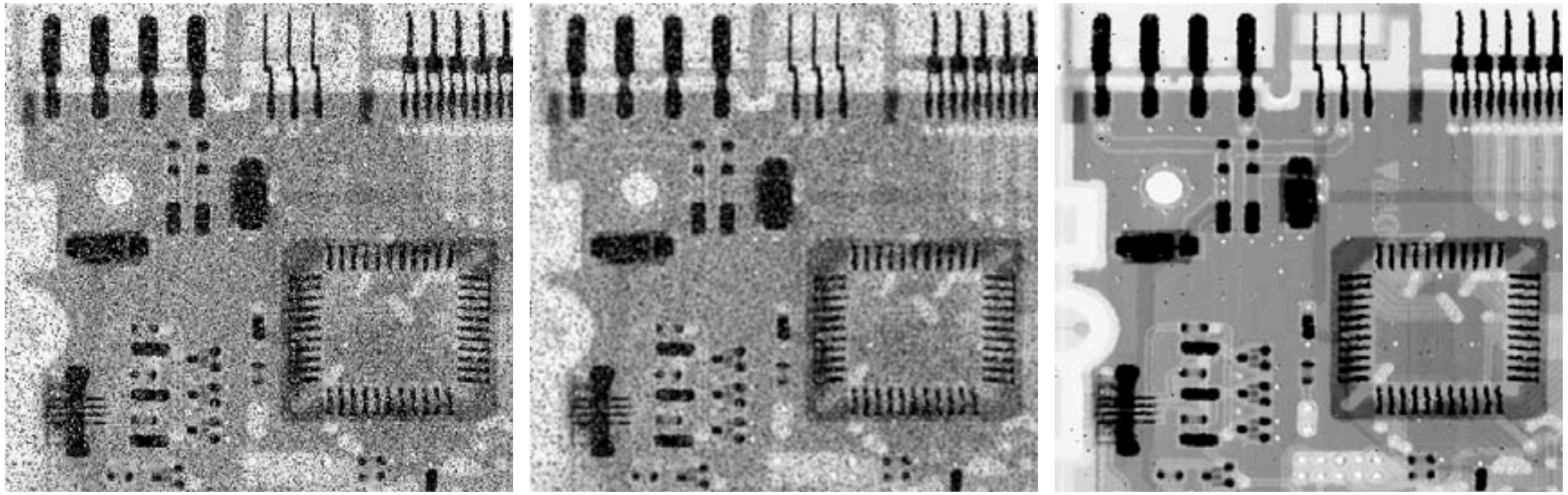
1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	3		

(1,1,2 $\boxed{2,3}$ 3,9,255)

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

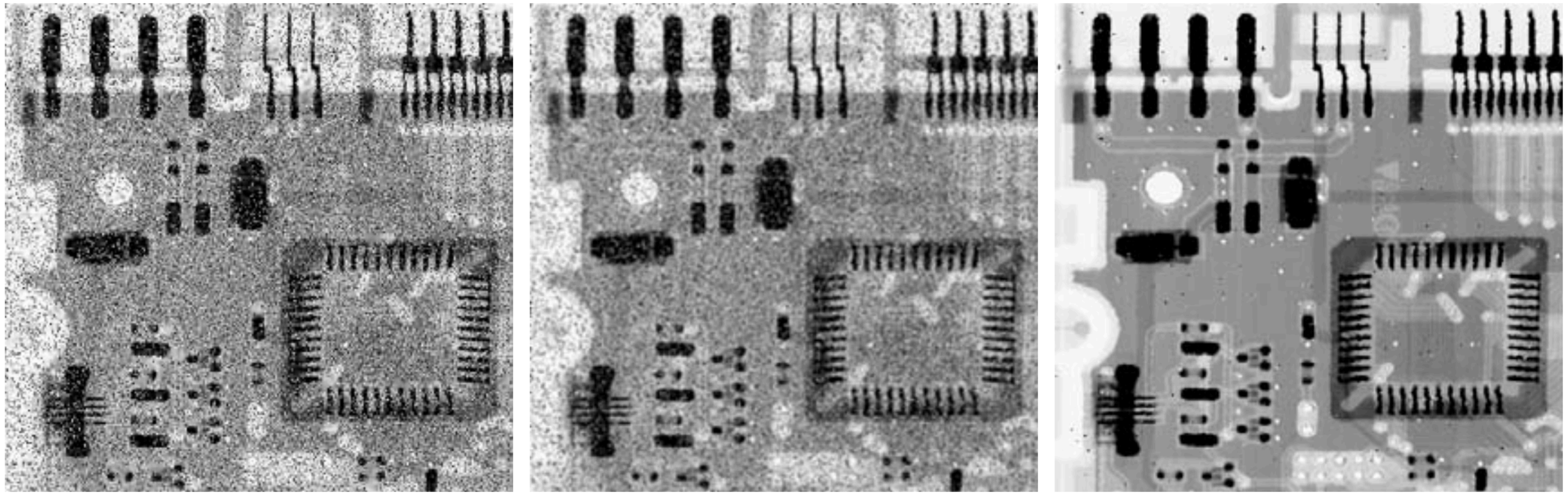
$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	3	?	

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

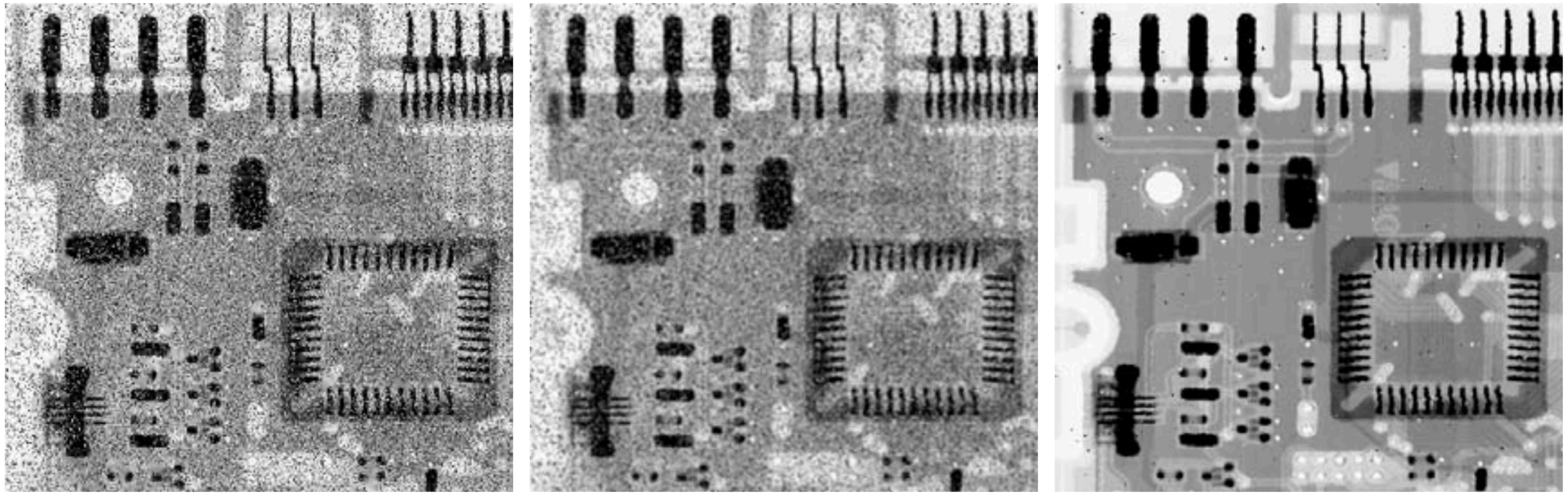
$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	3	3	

Spatial Filters

- **Arithmetic and geometric filters**
- **Very poor for salt & pepper noise**
- **Median Filter much better for this type of noise!**



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

$n \times n$ Neighborhood, here $n = 3$

$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$
$\frac{1}{n^2}$	$\frac{1}{n^2}$	$\frac{1}{n^2}$

$*$

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	3	3	
	3	2	
	3	2	

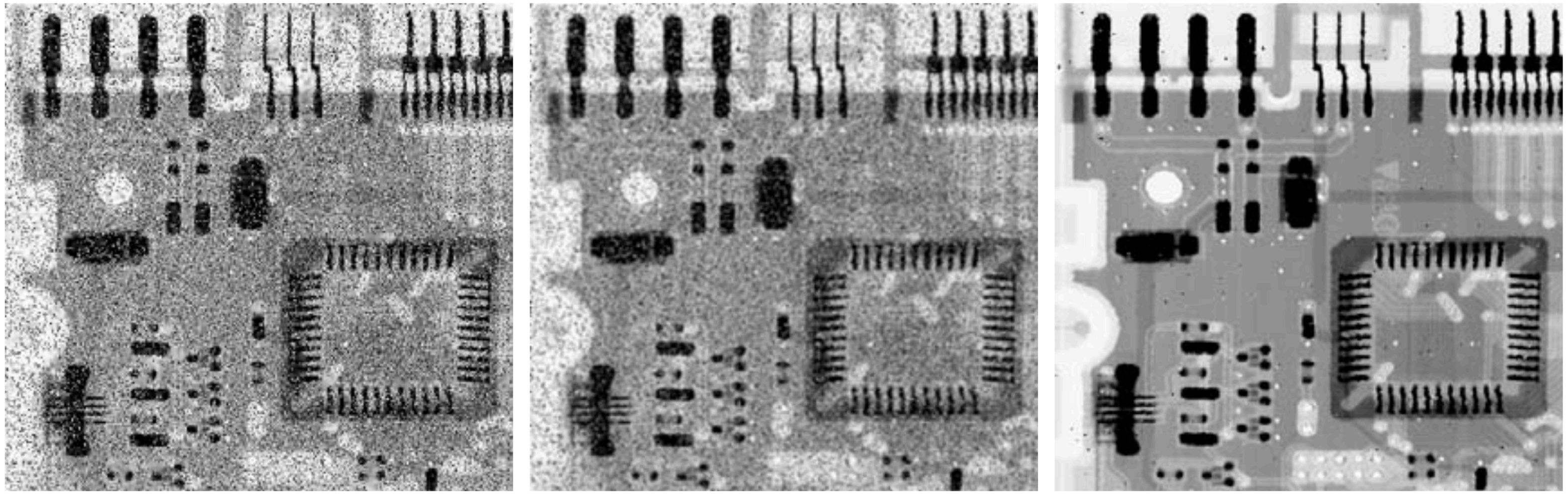
Spatial Filters

- **Mean vs Median Filter**

1	2	3	5
2	1	3	4
7	9	255	1
8	1	2	1
0	3	3	2

	31	31	
	32	31	
	32	31	

	3	3	
	3	2	
	3	2	

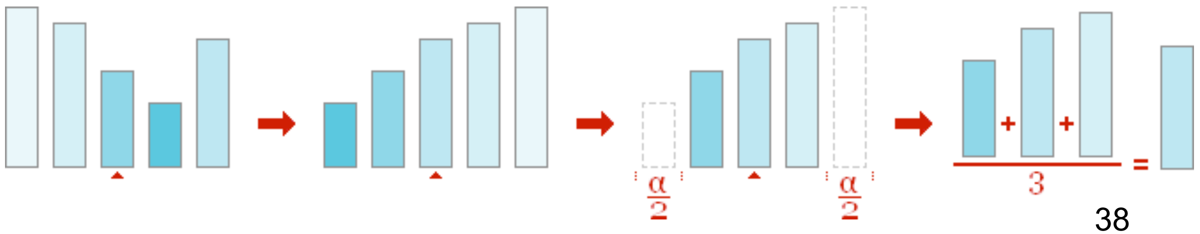


a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filters

- **Order-Statistic Filters**
- **Ranking**
  - Rank the intensities in a neighborhood
- **Pick Value**
  - Min
  - Max
  - Mean
  - α-trimmed
    - sort, find median, take mean of [median - $\frac{\alpha}{2}$, median + $\frac{\alpha}{2}$]



Denoising by Spatial Filtering

- **Filter Types**
  - Mean Filters
    - Arithmetic (“normal”) Mean
    - Geometric Mean
    - Harmonic Mean
    - Contraharmonic Mean of Order Q
  - Order-statistic Filters
    - Median
    - Min
    - Max
    - Midpoint
    - Alpha-Trimmed
  - Adaptive Filters

Spatial Filters: Mean Filters

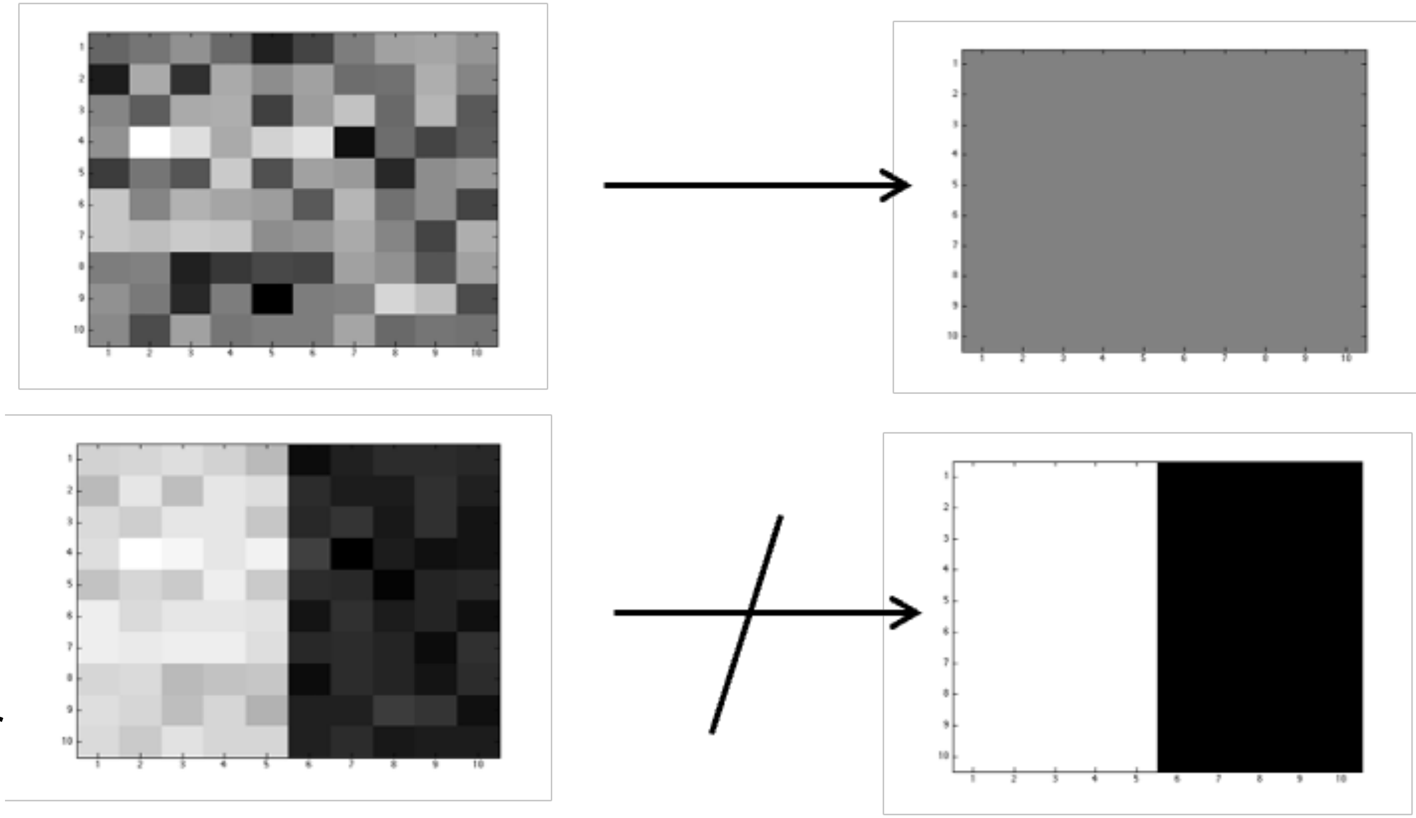
Type of Mean Filter	Mathmatical Expression	Usecases
Arithmetic Mean	$\hat{f}(x,y)=\frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t)$	Noise reduction by blurring **Good** for: Gaussian & uniform noise **Bad** for: Salt & Pepper
Geometric Mean	$\hat{f}(x,y)=\left(\prod_{(s,t)\in S_{xy} g(s,t)}\right)^{\frac{1}{mn}}$	Similar as arithmetic mean, but preserves more detail
Harmonic Mean	$\hat{f}(x,y)=\frac{mn}{\sum_{(s,t)\in S_{xy}}\frac{1}{g(s,t)}}$	Works for salt noise, not for pepper noise. Works for Gaussian noise.
Contraharmonic Mean of Order Q	$\hat{f}(x,y)=\frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$	Good for Salt & Pepper noise if Q>0 for pepper noise if Q<0 for salt noise

Spatial Filters: Order-Statistic Filters

Type of Mean Filter	Mathmatical Expression	Usecases
Median	$\hat{f}(x, y) = \text{median}_{(s,t)\in S_{xy}}\{g(s, t)\}$	Less blur than mean filtering Good for bipolar and unipolar impulse noise
Max	$\hat{f}(x, y) = \max_{(s,t)\in S_{xy}}\{g(s, t)\}$	Good for pepper noise
Min	$\hat{f}(x, y) = \min_{(s,t)\in S_{xy}}\{g(s, t)\}$	Good for salt noise
Midpoint	$\hat{f}(x, y) = \frac{1}{2}\left(\max_{(s,t)\in S_{xy}}\{g(s, t)\} + \min_{(s,t)\in S_{xy}}\{g(s, t)\}\right)$	Good for Gaussian or uniform noise
Alpha-trimmed mean filter	$\hat{f}(x, y) = \frac{1}{mn - \alpha}\sum_{(s,t)\in S_{xy}} g(s, t)$	Good for combination of salt-and-pepper and Gaussian noise

Spatial Filtering: Adaptive Filtering

- **Adaptive Filters**
  - Denoising by mean filtering may be useful in homogeneous areas, but less so close to edges
  - Adjust filter behavior based on statistical characteristics of the image inside the filter region
  - Often better performance, but computationally more expensive
  - Reasonable to consider mean and variance of a neighborhood for filter adaptation because they are important parameters also for filter design



Spatial Filtering: Adaptive Local Noise Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - σ_η^2 is the variance of the noise corrupting $f(x, y)$ to $g(x, y)$,
  - μ_L is the local mean and σ_L^2 the local variance of the pixels in S_{xy}

Spatial Filtering: Adaptive Local Noise Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - σ_η^2 is the variance of the noise corrupting $f(x, y)$ to $g(x, y)$,
  - μ_L is the local mean and σ_L^2 the local variance of the pixels in S_{xy}
- **Filter**
  - If $\sigma_\eta^2 = 0$, filter should return $g(x, y)$ as then $g(x, y) = f(x, y)$

Spatial Filtering: Adaptive Local Noise Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - σ_η^2 is the variance of the noise corrupting $f(x, y)$ to $g(x, y)$,
  - μ_L is the local mean and σ_L^2 the local variance of the pixels in S_{xy}
- **Filter**
  - If $\sigma_\eta^2 = 0$, filter should return $g(x, y)$ as then $g(x, y) = f(x, y)$
  - If $\sigma_L^2 \gg \sigma_\eta^2$, it implies an edge and the filter should return a value close to $g(x, y)$

Spatial Filtering: Adaptive Local Noise Filtering

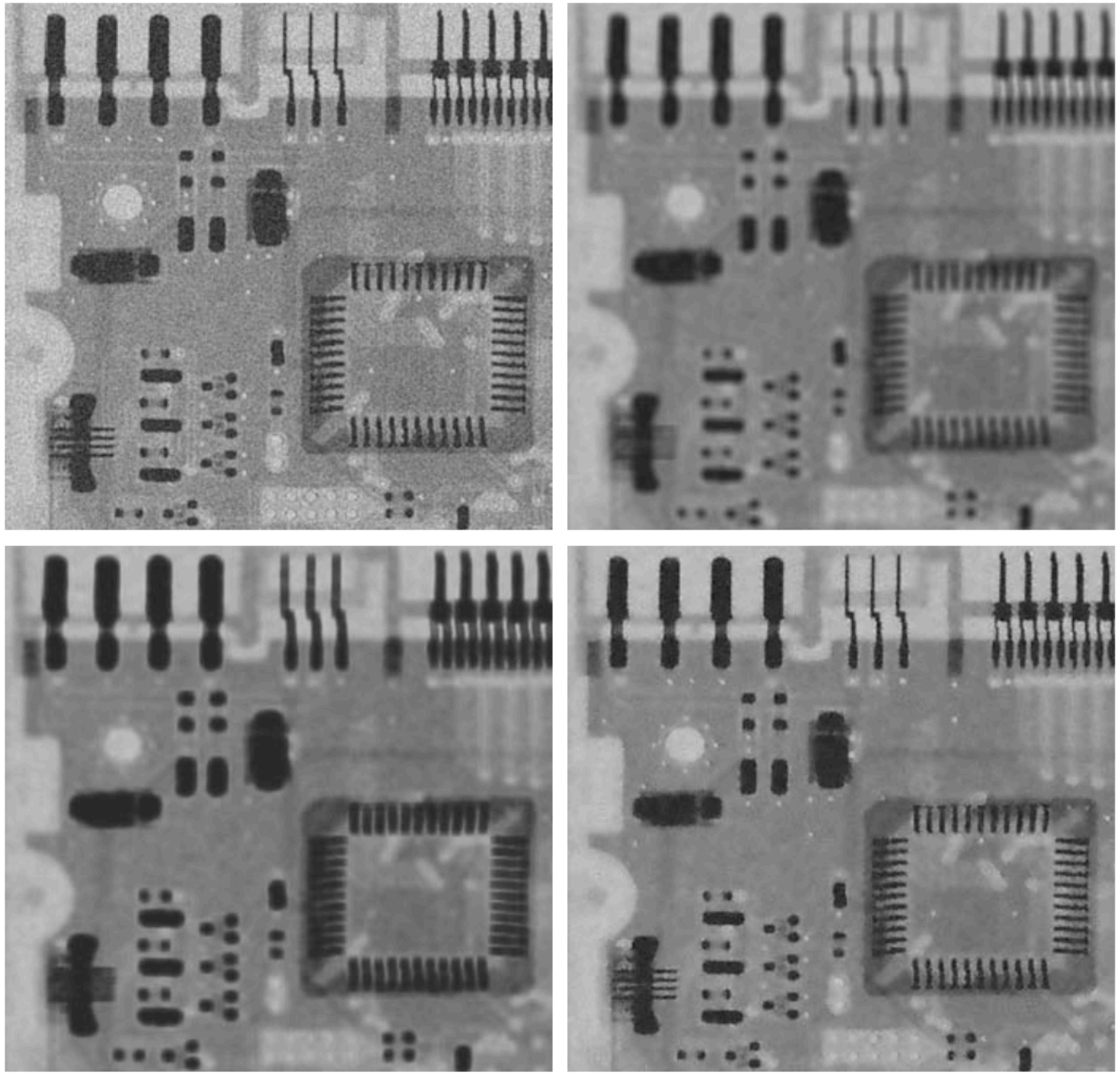
- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - σ_η^2 is the variance of the noise corrupting $f(x, y)$ to $g(x, y)$,
  - μ_L is the local mean and σ_L^2 the local variance of the pixels in S_{xy}
- **Filter**
  - If $\sigma_\eta^2 = 0$, filter should return $g(x, y)$ as then $g(x, y) = f(x, y)$
  - If $\sigma_L^2 \gg \sigma_\eta^2$, it implies an edge and the filter should return a value close to $g(x, y)$
  - If $\sigma_L^2 \approx \sigma_\eta^2$, the filter should act as a mean filter and return the arithmetic mean

Spatial Filtering: Adaptive Local Noise Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - σ_η^2 is the variance of the noise corrupting $f(x, y)$ to $g(x, y)$,
  - μ_L is the local mean and σ_L^2 the local variance of the pixels in S_{xy}
- **Filter**
  - If $\sigma_\eta^2 = 0$, filter should return $g(x, y)$ as then $g(x, y) = f(x, y)$
  - If $\sigma_L^2 \gg \sigma_\eta^2$, it implies an edge and the filter should return a value close to $g(x, y)$
  - If $\sigma_L^2 \approx \sigma_\eta^2$, the filter should act as a mean filter and return the arithmetic mean

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

Spatial Filtering: Adaptive Local Noise Filtering



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.

Spatial Filtering: Adaptive Local Noise Filtering

a b
c d

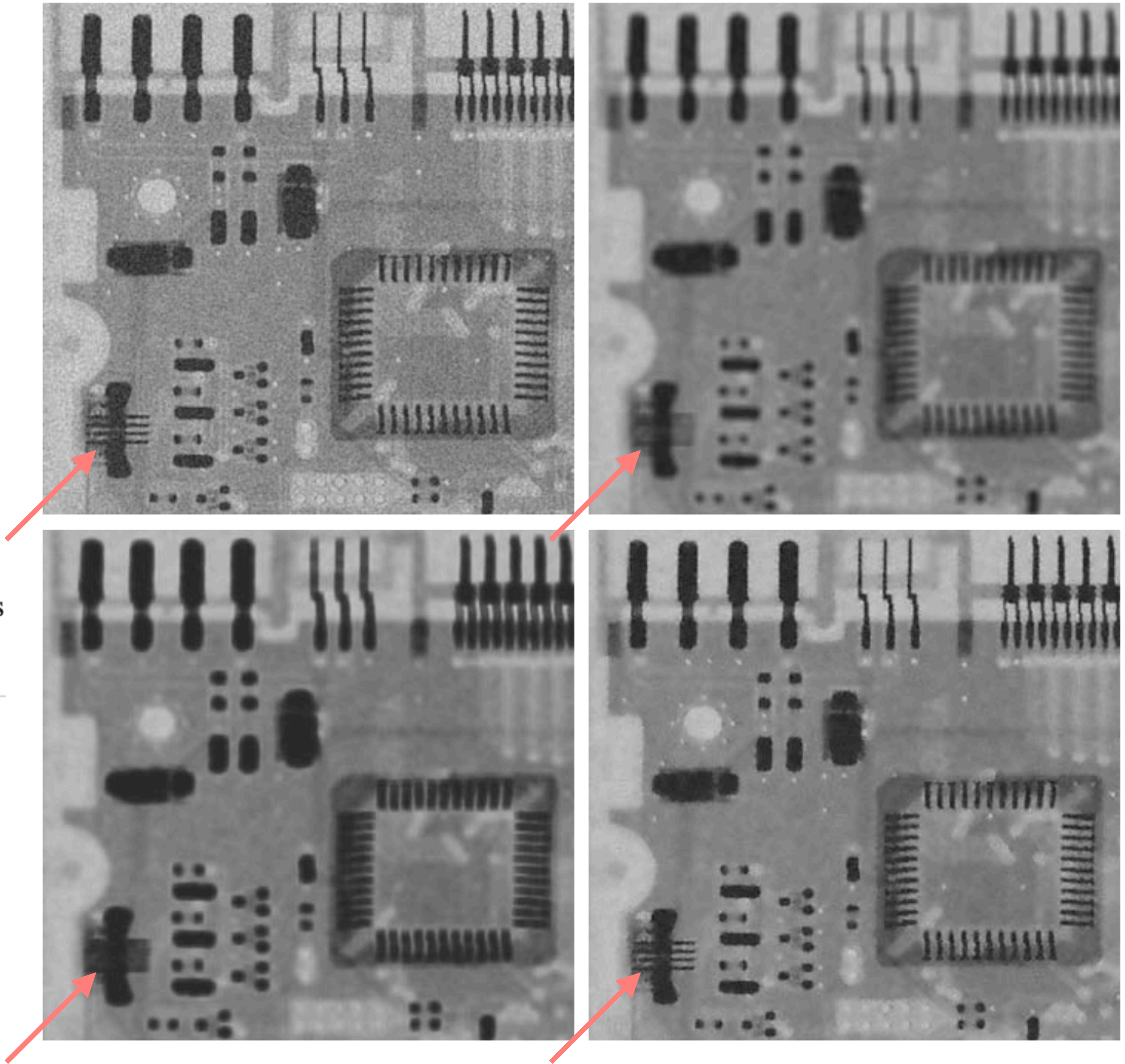


FIGURE 5.13 (a) Image corrupted by additive Gaussian noise of zero mean and variance 1000. (b) Result of arithmetic mean filtering. (c) Result of geometric mean filtering. (d) Result of adaptive noise reduction filtering. All filters were of size 7×7.

Spatial Filtering: Adaptive Median Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - z_{min} minimum intensity, z_{max} maximum intensity, z_{med} median intensity value in S_{xy}
  - z_{xy} intensity value at (x, y)
  - S_{max} maximum allowed size of S_{xy}

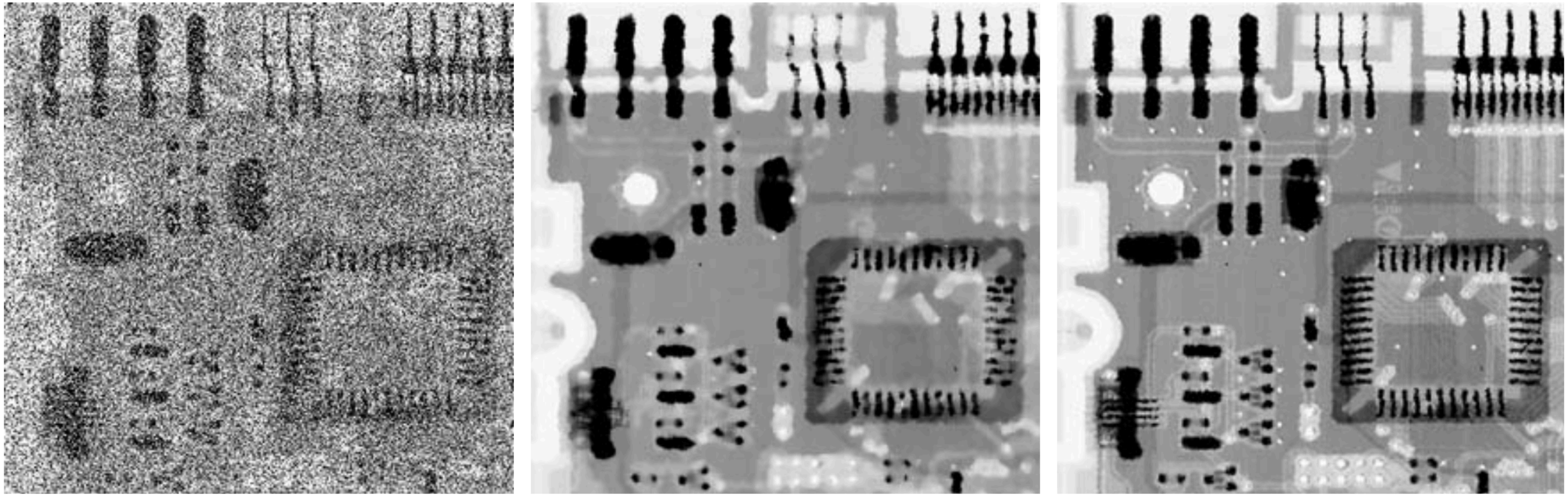
Spatial Filtering: Adaptive Median Filtering

- **Setting**
  - Local region S_{xy} and for a point (x, y) in S_{xy}, $g(x, y)$ is the value of the noisy image at (x, y),
  - z_{min} minimum intensity, z_{max} maximum intensity, z_{med} median intensity value in S_{xy}
  - z_{xy} intensity value at (x, y)
  - S_{max} maximum allowed size of S_{xy}
- **Filter in 2 stages:**

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

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

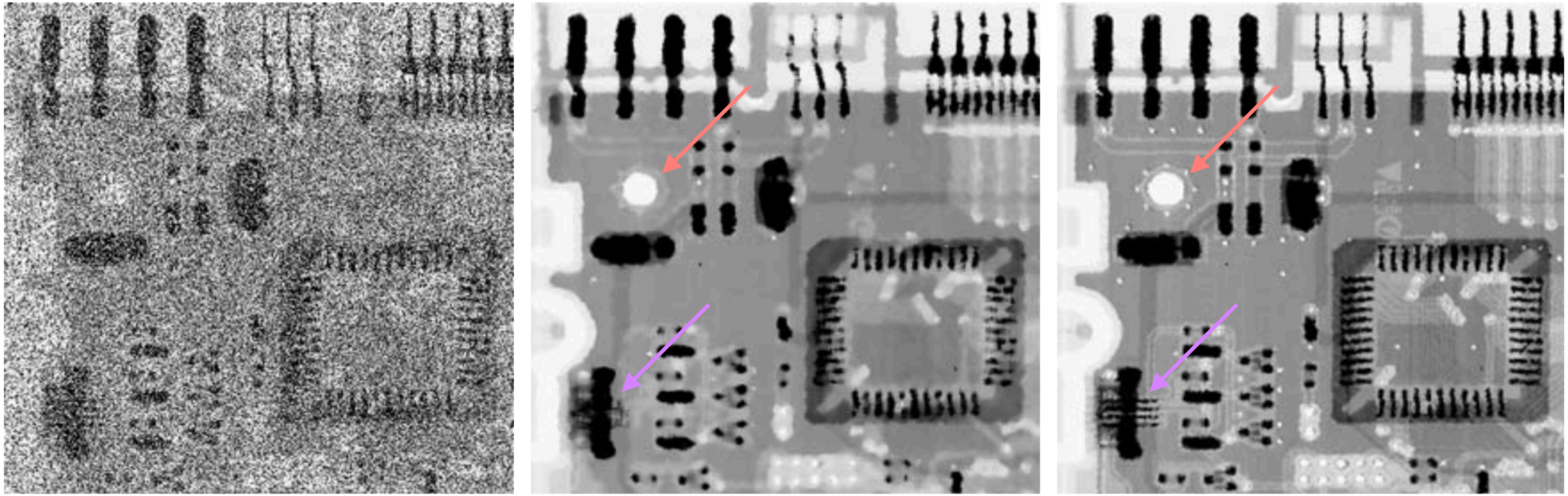
Spatial Filtering: Adaptive Median Filtering



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 $S_{\max} = 7$.

Spatial Filtering: Adaptive Median Filtering

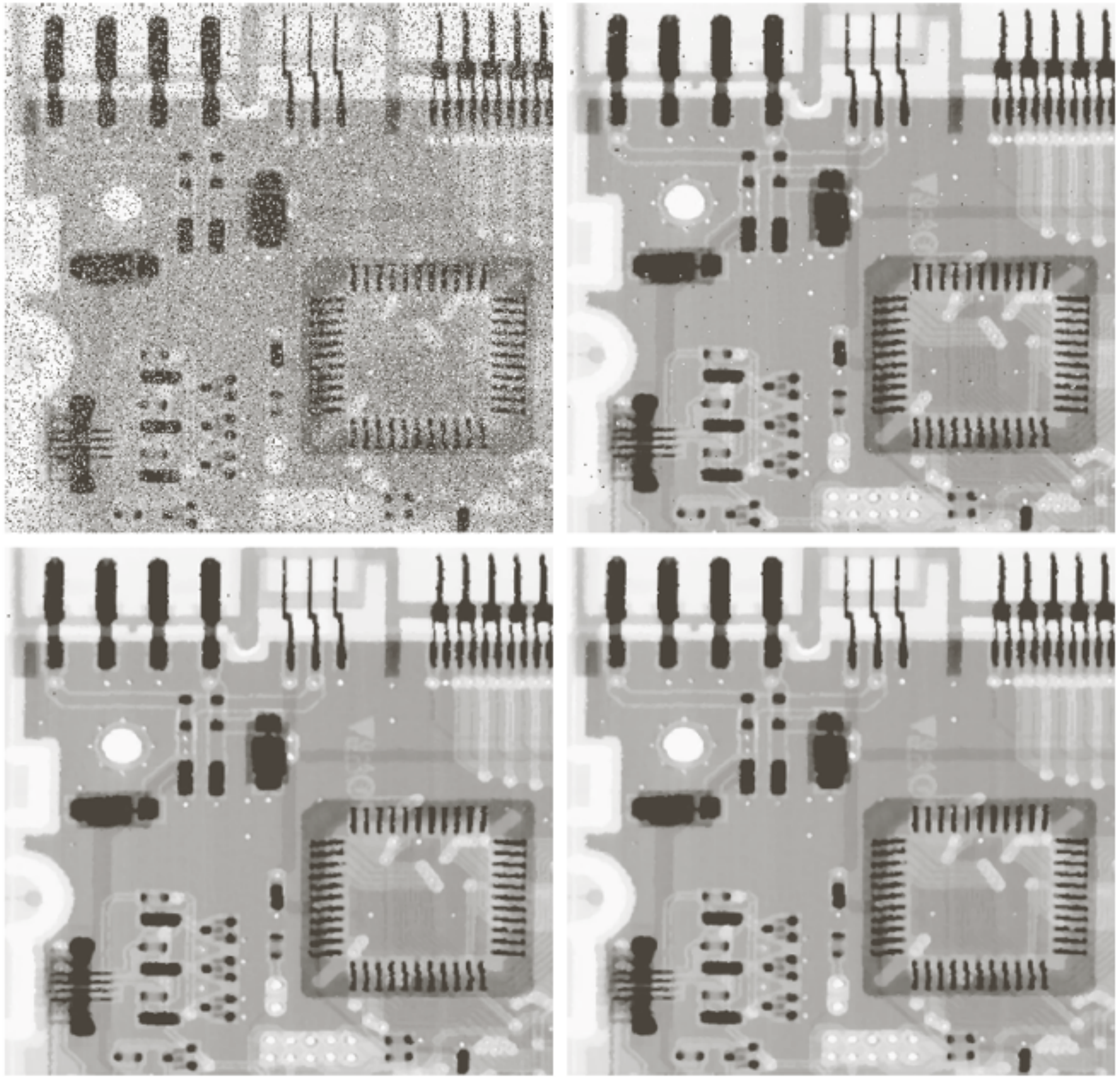


a b c

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 $S_{max} = 7$.

Repeated filtering

- Filters can be applied repeatedly
- Could be repeated until a criterion is met



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.

Frequency Domain Filtering: Periodic noise

- **Spatially Dependent**
  - (unlike previously discussed noise types)
- **Source**
  - Usually caused by electrical inference during acquisition
- **Best removed in frequency domain**
  - Periodic noise appears as concentrated bursts of energy in the Fourier transform
- **Filter**
  - Bandreject, Bandpass, Notch, Optimum-Notch

Periodic noise

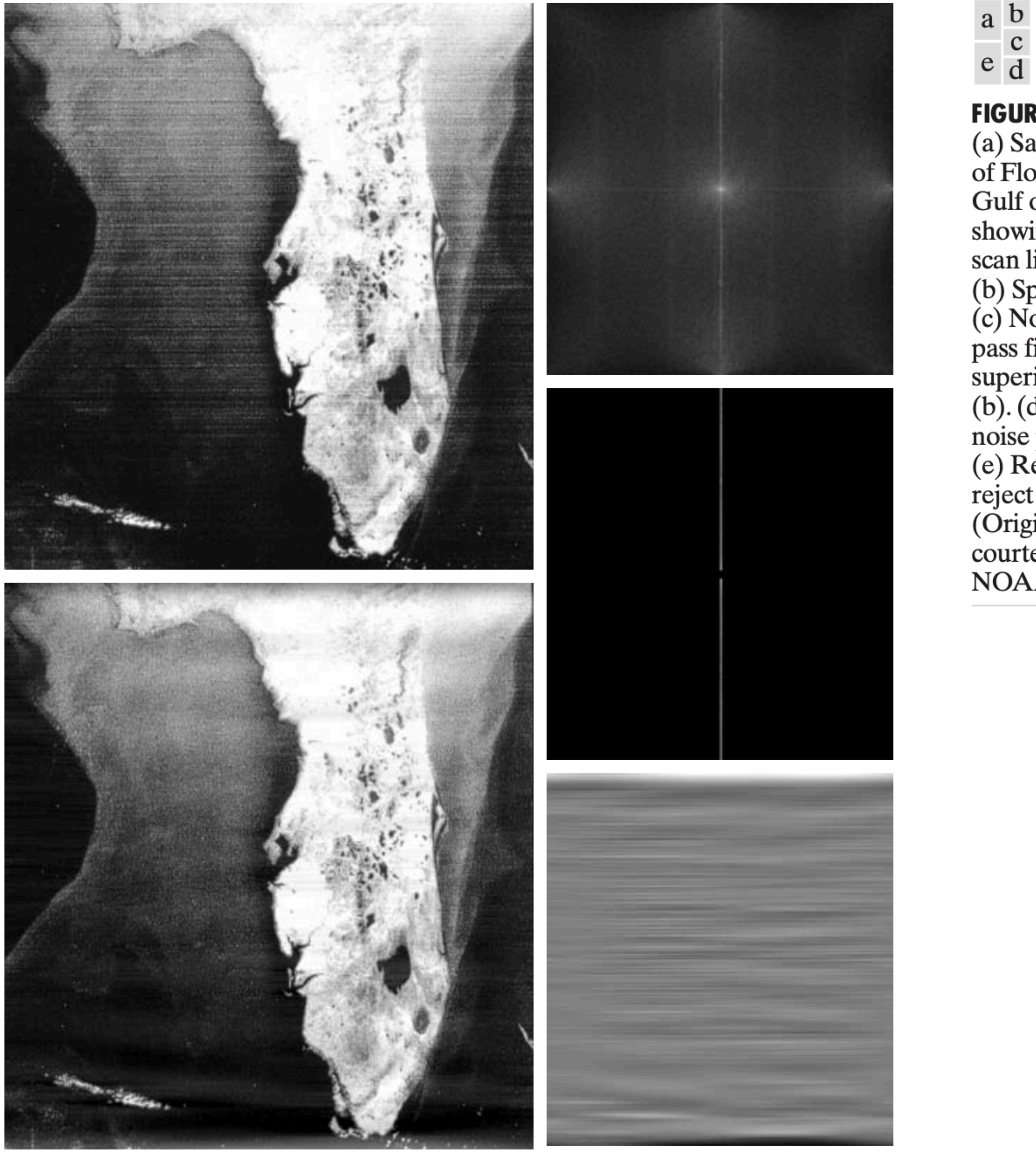
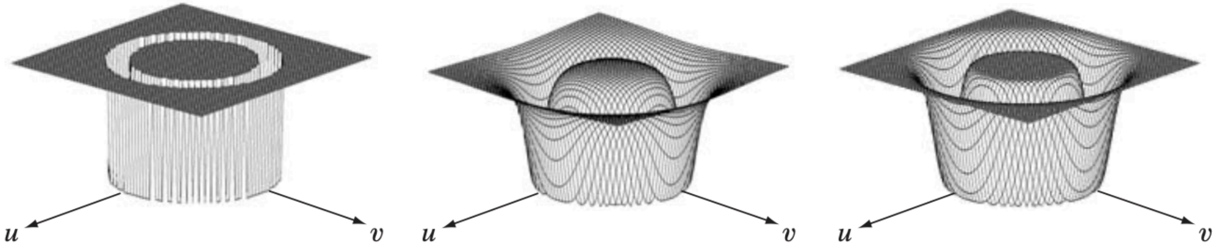


FIGURE 5.19 (a) Satellite image of Florida and the Gulf of Mexico showing horizontal scan lines. (b) Spectrum. (c) Notch pass filter superimposed on (b). (d) Spatial noise pattern. (e) Result of notch reject filtering. (Original image courtesy of NOAA.)

Frequency Domain Filtering: Band Reject Filter



a b c

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

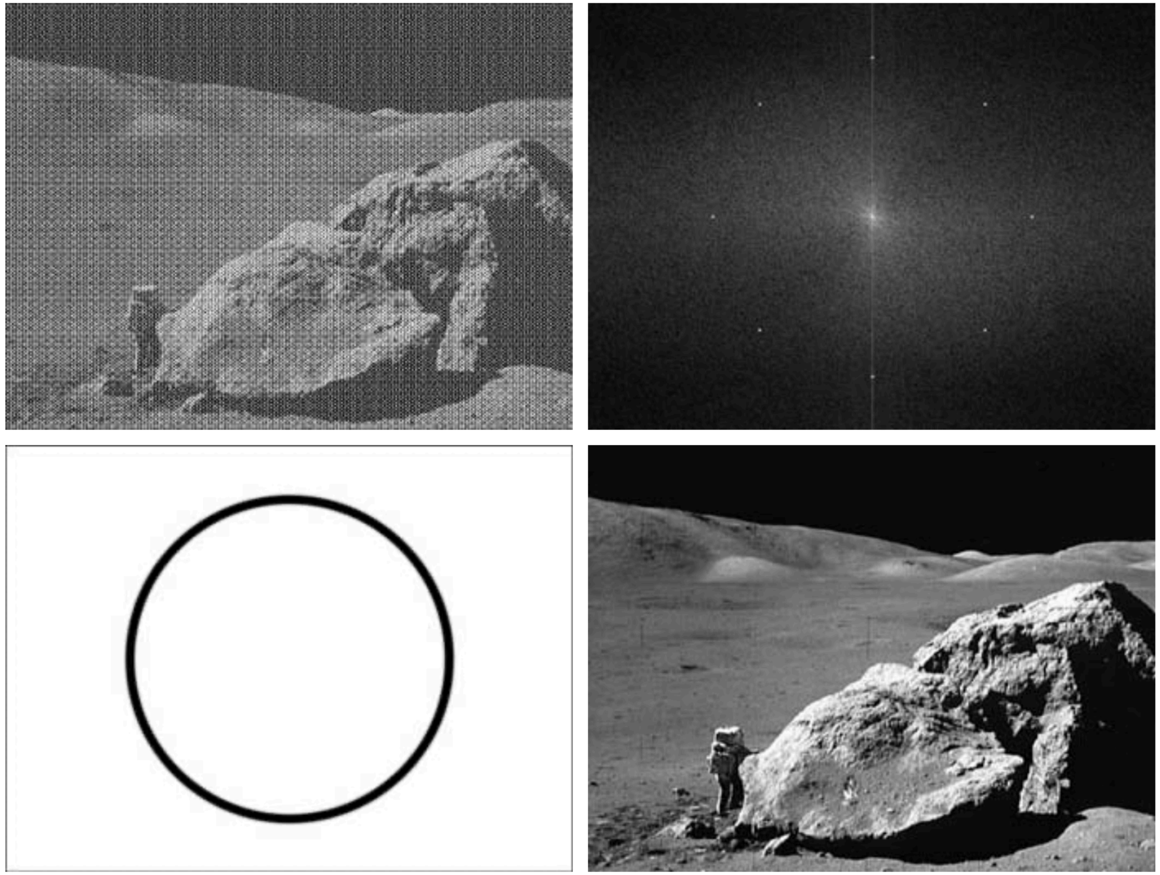


TABLE 4.6
Bandreject filters. W is the width of the band, D is the distance $D(u, v)$ from the center of the filter, D_0 is the cutoff frequency, and n is the order of the Butterworth filter. We show D instead of $D(u, v)$ to simplify the notation in the table.

Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \le D \le D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = \dfrac{1}{1 + \left[\dfrac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u,v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$

Frequency Domain Filtering: Notch Filter

- Rejects (or passes) frequencies in predefined neighborhoods about a center frequency
- Due to the symmetry of the Fourier transform, notch filters must appear in symmetric pairs about the origin
- The number of pairs of notch filters that can be implemented is arbitrary

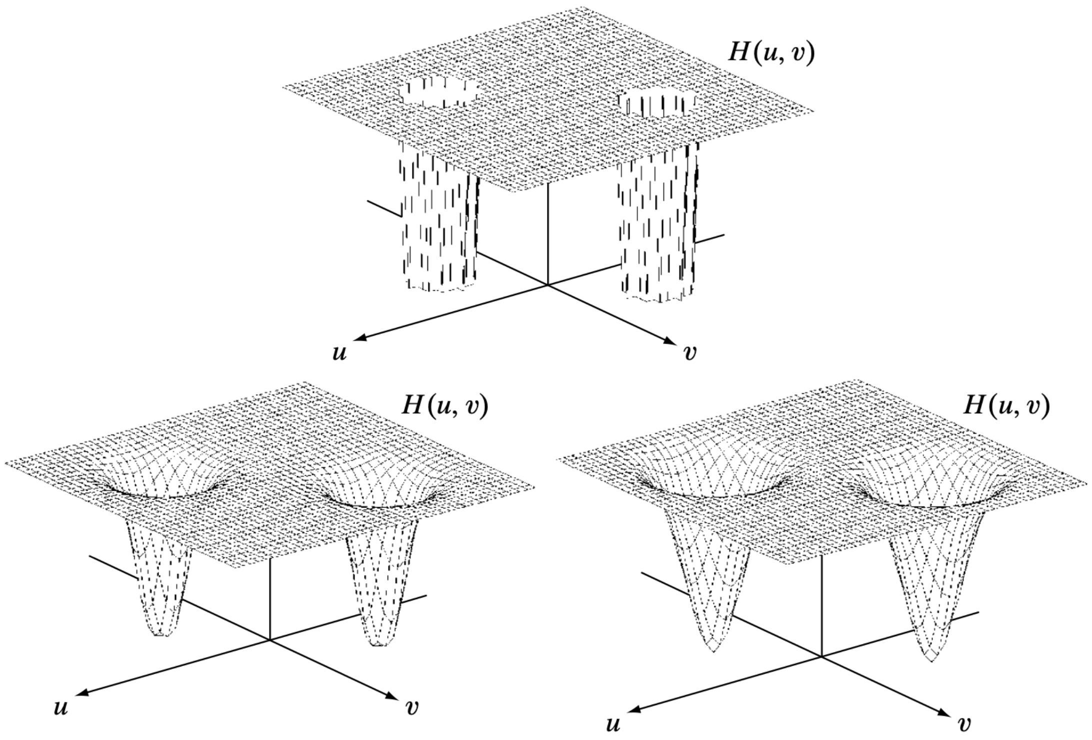
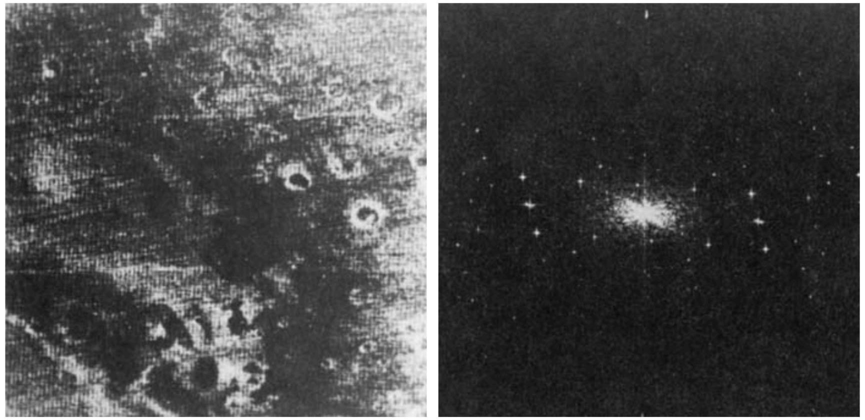


FIGURE 5.18 Perspective plots of (a) ideal, (b) Butterworth (of order 2), and (c) Gaussian notch (reject) filters.

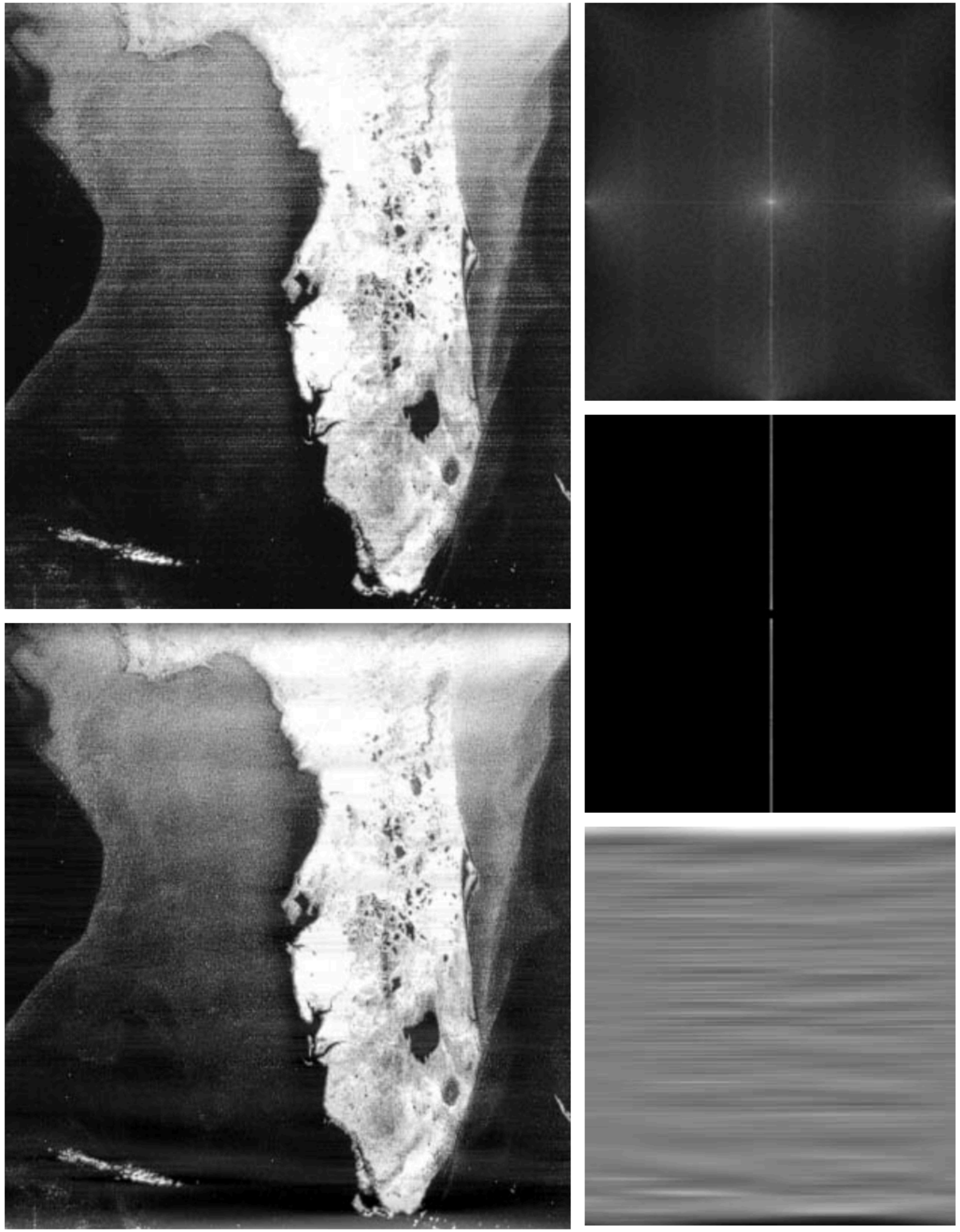
Notch Filter

- Band Pass/Reject would not be possible in this case
- Notch has to be symmetric around origin
- Can also be used to isolate noise pattern



a b

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



a b c e d

FIGURE 5.19 (a) Satellite image of Florida and the Gulf of Mexico showing horizontal scan lines. (b) Spectrum. (c) Notch pass filter superimposed on (b). (d) Spatial noise pattern. (e) Result of notch reject filtering. (Original image courtesy of NOAA.)

Frequency Domain Filtering: Optimum-Notch

- Isolate principal contributions of inference pattern by placing a notch filter at the location of each spike
- Subtract a variable, weighted portion of the pattern from the corrupted image, i.e.

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

- $w(x, y)$ is *weighting* or *modulation function* and is to be determined by optimization
- One approach for optimization: minimize the local variance of the estimate $\hat{f}(x, y)$

Frequency Domain Filtering: Optimum-Notch

Consider a neighborhood of size $(2a + 1)$ by $(2b + 1)$ about a point (x, y). The "local" variance of $\hat{f}(x, y)$ at coordinates (x, y) can be estimated from the samples, as follows:

$$\sigma^2(x, y) = \frac{1}{(2a + 1)(2b + 1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \left[\hat{f}(x + s, y + t) - \overline{\hat{f}}(x, y) \right]^2 \qquad (5.4\text{-}6)$$

where $\overline{\hat{f}}(x, y)$ is the average value of $\hat{f}$ in the neighborhood; that is,

$$\overline{\hat{f}}(x, y) = \frac{1}{(2a + 1)(2b + 1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \hat{f}(x + s, y + t) \qquad (5.4\text{-}7)$$

Points on or near the edge of the image can be treated by considering partial neighborhoods or by padding the border with 0s.

Frequency Domain Filtering: Optimum-Notch

Putting that into $\hat{f}(x, y) = g(x, y) - w(x, y)\eta(x, y)$, this yields

$$\sigma^2(x, y) = \frac{1}{(2a + 1)(2b + 1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \{[g(x + s, y + t) - w(x + s, y + t)\eta(x + s, y + t)] - [\bar{g}(x, y) - \overline{w(x, y)\eta(x, y)}]\}^2 \tag{5.4-8}$$

Assuming that $w(x, y)$ remains essentially constant over the neighborhood gives the approximation

$$w(x + s, y + t) = w(x, y) \tag{5.4-9}$$

for $-a \leq s \leq a$ and $-b \leq t \leq b$. This assumption also results in the expression

$$\overline{w(x, y)\eta(x, y)} = w(x, y)\bar{\eta}(x, y) \tag{5.4-10}$$

in the neighborhood. With these approximations, Eq. (5.4-8) becomes

Frequency Domain Filtering: Optimum-Notch

With these approximations, Eq. (5.4-8) becomes

$$\sigma^2(x, y) = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \Big\{ [g(x+s, y+t) - w(x, y)\eta(x+s, y+t)] - [\bar{g}(x, y) - w(x, y)\bar{\eta}(x, y)] \Big\}^2 \qquad (5.4\text{-}11)$$

To minimize $\sigma^2(x, y)$, we solve

$$\frac{\partial \sigma^2(x, y)}{\partial w(x, y)} = 0 \qquad (5.4\text{-}12)$$

for $w(x, y)$.

Frequency Domain Filtering: Optimum-Notch

The result is

$$w(x, y) = \frac{\overline{g(x, y)\eta(x, y)} - \overline{g}(x, y)\overline{\eta}(x, y)}{\overline{\eta^2}(x, y) - \overline{\eta}^2(x, y)} \tag{5.4-13}$$

To obtain the restored image $\hat{f}(x, y)$, we compute $w(x, y)$ from Eq. (5.4-13) and then use Eq. (5.4-5). As $w(x, y)$ is assumed to be constant in a neighborhood, computing this function for every value of x and y in the image is unnecessary. Instead, $w(x, y)$ is computed for *one* point in each nonoverlapping neighborhood (preferably the center point) and then used to process all the image points contained in that neighborhood.

Summary

- **Image Restoration**
  - Aim: Recover original signal after degradation, today only noise models were considered
- **Find out something about this noise**
  - If possible, find a homogeneous patch in the image and look at the histogram of intensities there, try to read out the type of noise in the histogram
  - Inspect the frequency domain
- **Apply best suiting filtering technique to the image based on that noise model**
  - Periodic Noise -> Frequency domain
  - Salt & Pepper -> Order-Statistic Filters
  - Adaptive Filters can improve outcome by preserving edges, but come at computational cost