Image & Video Processing

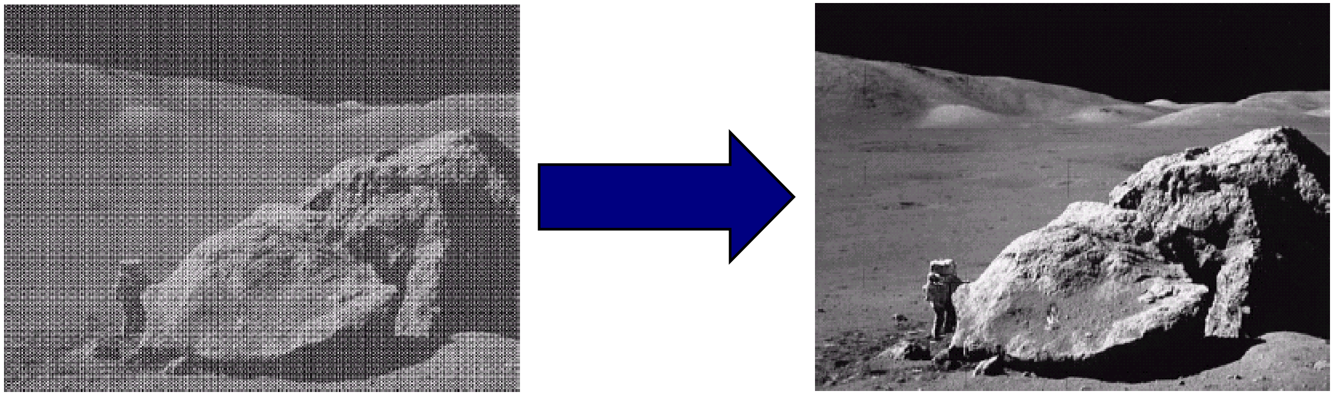
Image Restoration:

Noise Removal

What is Image Restoration?

Image restoration attempts to restore images that have been degraded

- Identify the degradation process and attempt to remove it in order to go back to the "original"
- Similar to image enhancement, but more objective

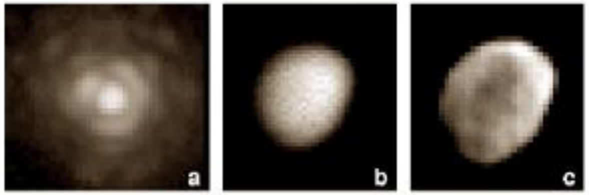


Applications

Started from the 1950s

- Scientific explorations
- Legal investigations
- Film making and archival
- Image and video (de-)coding
- …
- Consumer photography

Example of image restoration
Asteroid Vesta

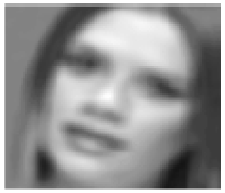


Property of Taurus Film and Joanneum Research

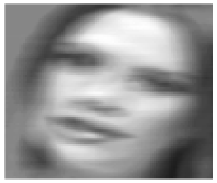
Degradation causes

Degradation examples:

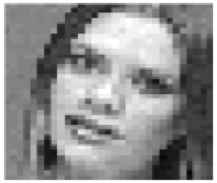
- original



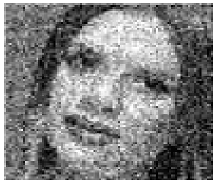
- optical blur



- motion blur



- spatial quantization (discrete pixels)



- additive intensity noise

Causes:

- Camera: translation, shake, out-of-focus …
- Environment: scattered and reflected light
- Device noise: CCD/CMOS sensor and circuitry
- Quantization noise
- Transmission error

A Model for Image Distortion/Restoration

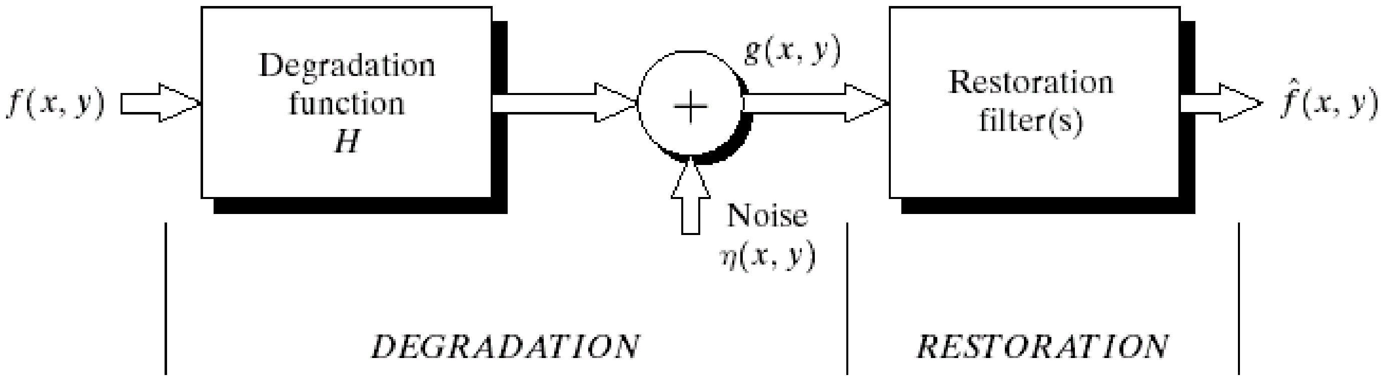


FIGURE 5.1 A model of the image degradation/restoration process.

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

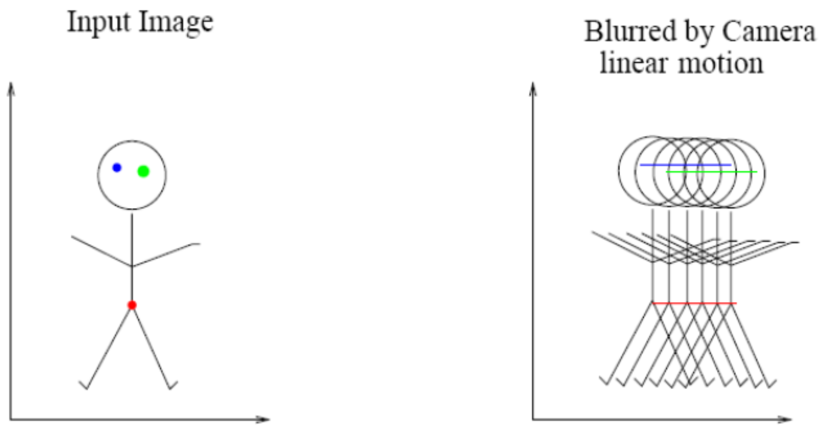
where *f(x, y)* is the original image pixel, *H* is the degradation function, *η(x, y)* is the noise term and *g(x, y)* is the resulting noisy pixel

Assumptions for the Distortion Model

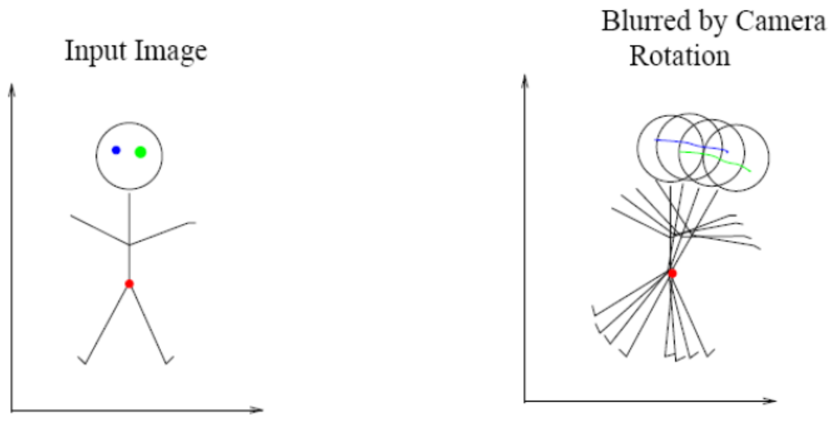
- Noise
  - Independent of spatial location
    - Exception: periodic noise ..
  - Uncorrelated with image
- Degradation function H
  - Linear
  - Position-invariant

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

Step #1: image degraded only by noise.



SPACE-INVARIENT RESPONSE - each point on image gives same response just shifted in position.

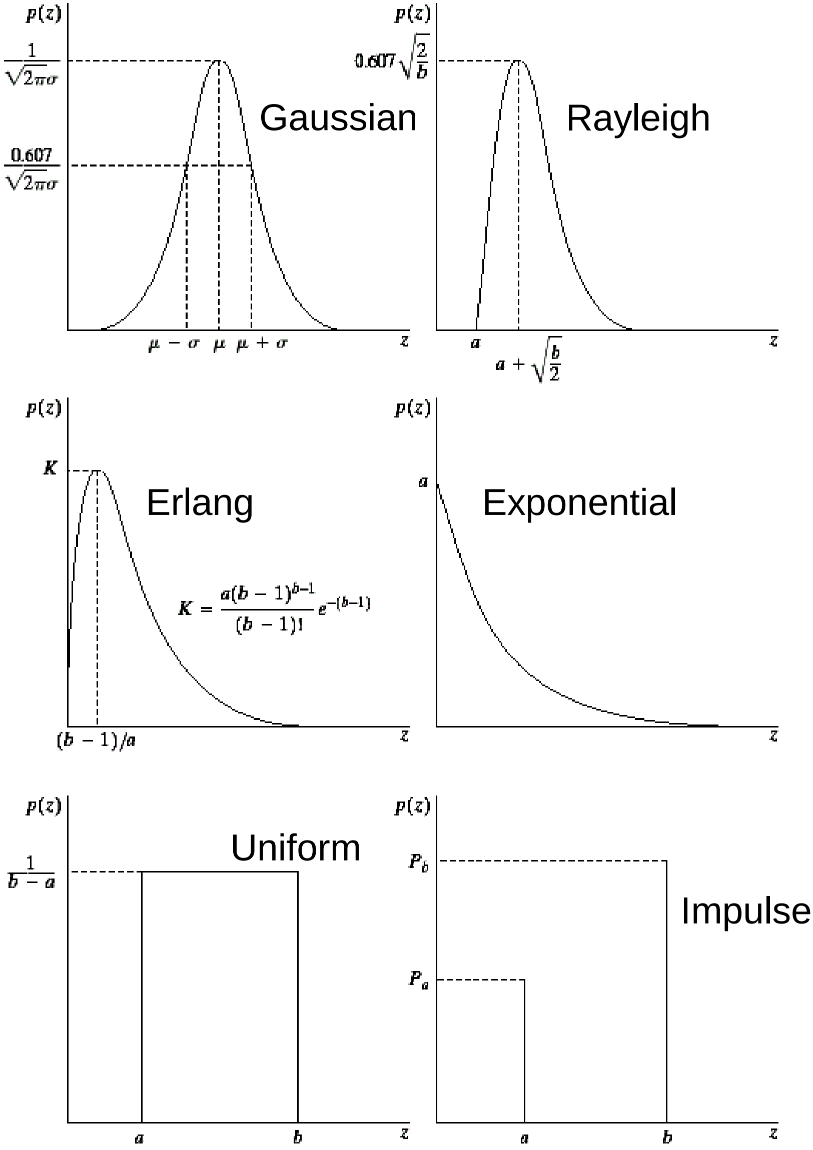


SPACE-VARIENT RESPONSE - each point on image gives a different response

Noise Models

There are many different models for the image noise term $\eta(x, y)$:

- Gaussian
  - Most common model
- Rayleigh
- Erlang
- Exponential
- Uniform
- Impulse
  - *Salt and pepper* noise

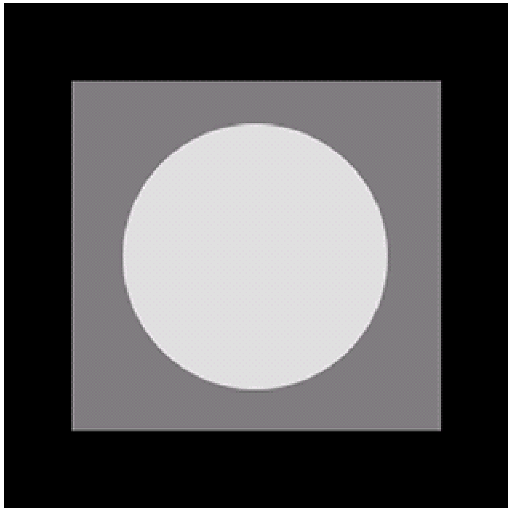


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

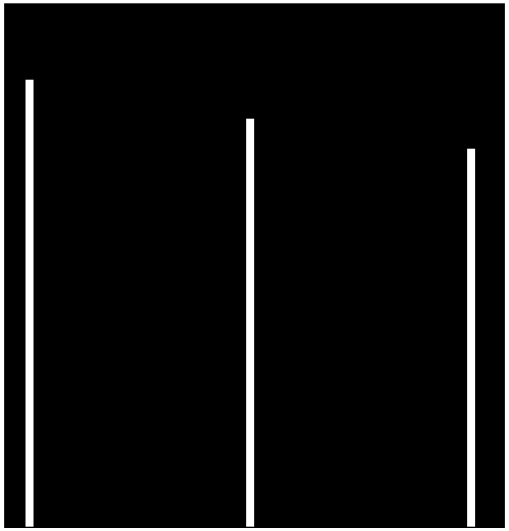
Noise Example

The test pattern to the right is ideal for demonstrating the addition of noise

The following slides will show the result of adding noise based on various models to this image



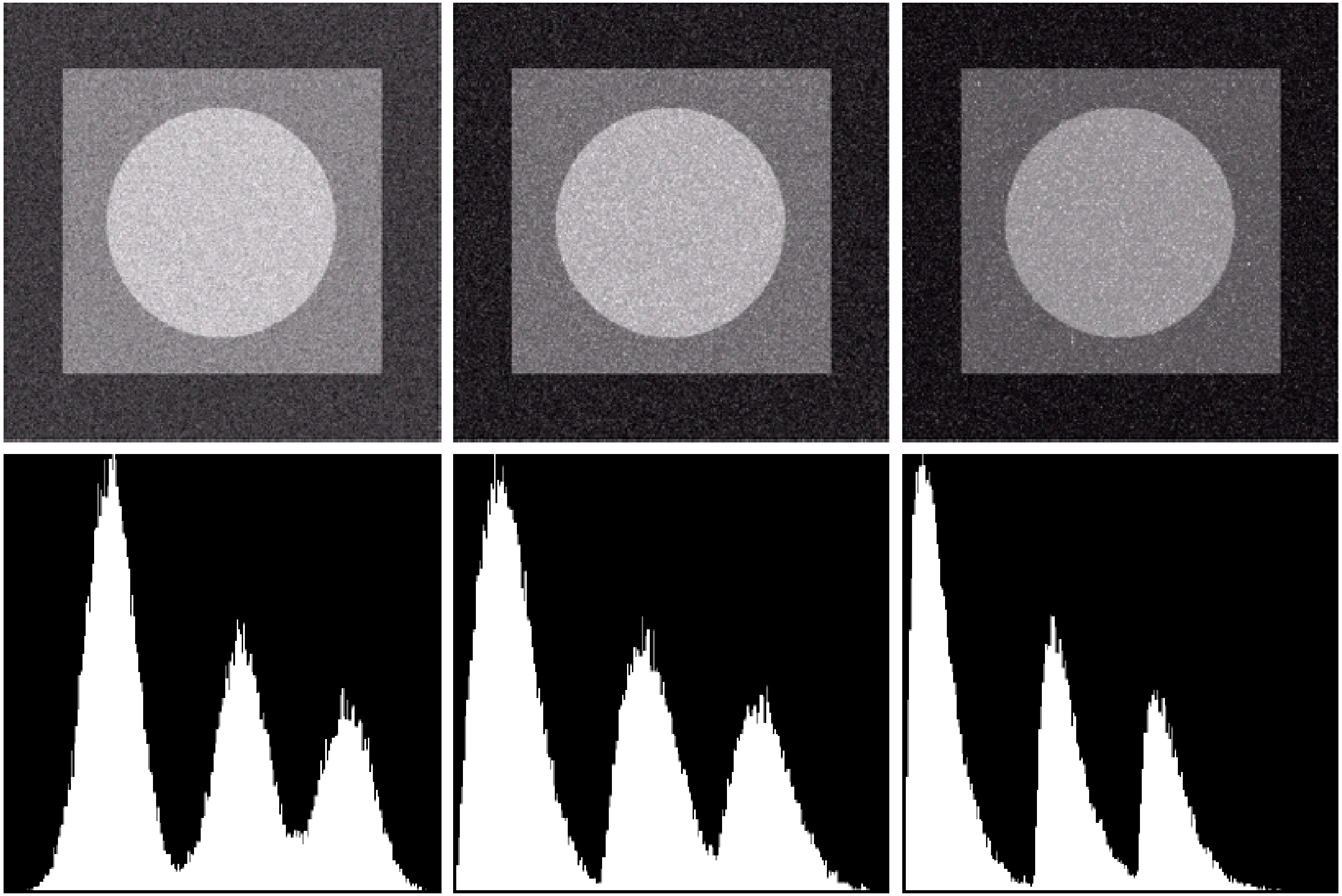
Image



Histogram

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Noise Example (cont…)



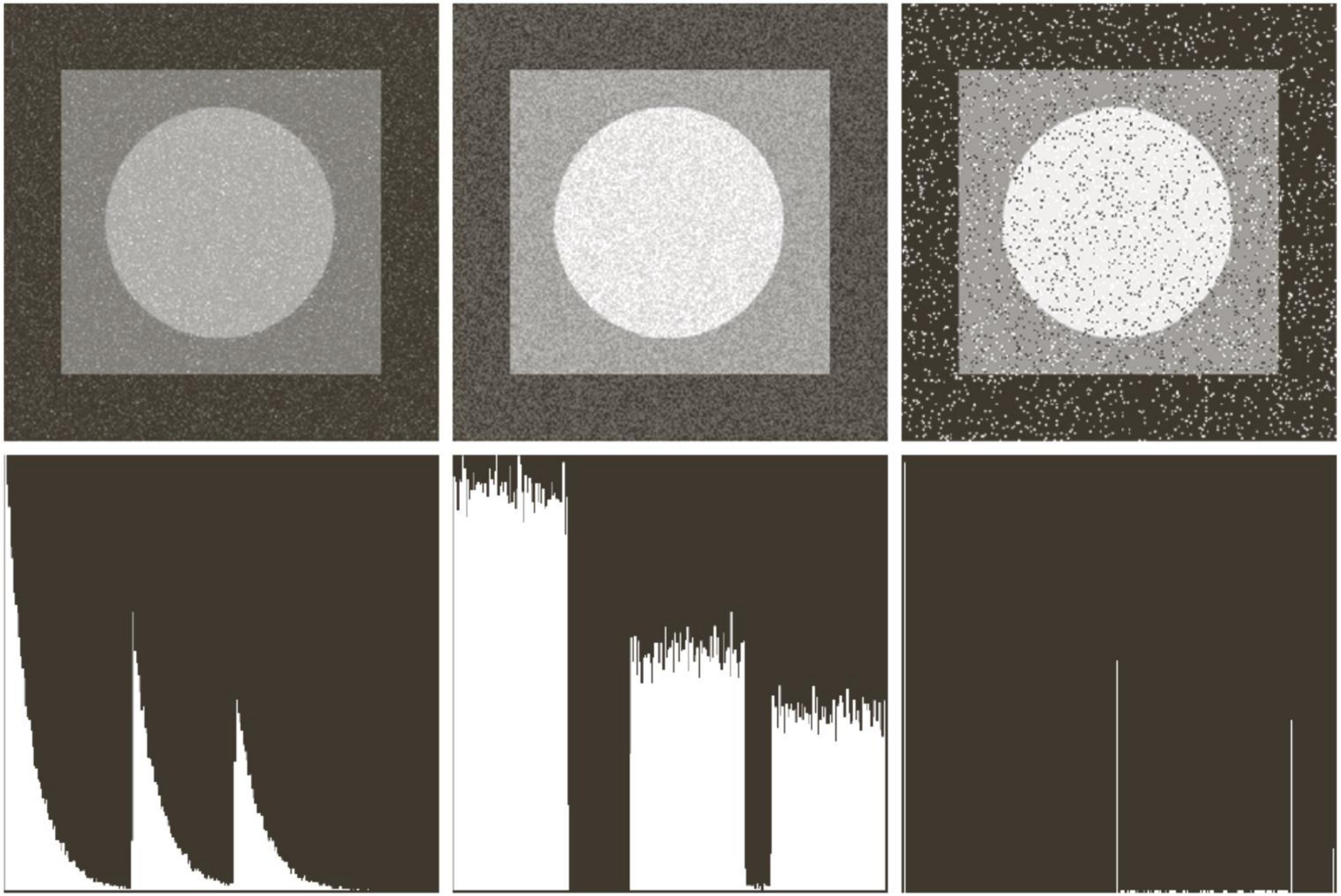
Gaussian

Rayleigh

Erlang

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Noise Example (cont…)



Exponential Uniform Impulse

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Estimation of noise parameters

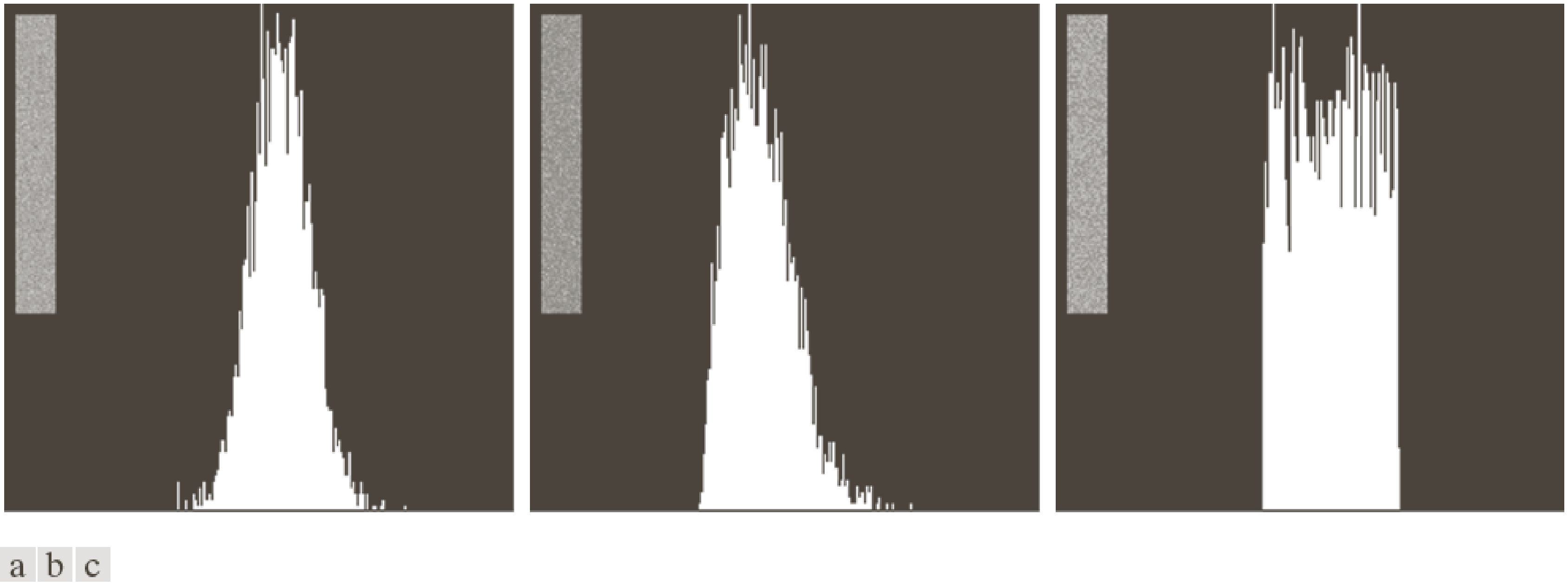


FIGURE 5.6 Histograms computed using small strips (shown as inserts) from (a) the Gaussian, (b) the Rayleigh, and (c) the uniform noisy images in Fig. 5.4.

Filtering to Remove Noise

We can use spatial filters of different kinds to remove different kinds of noise

The *arithmetic mean* filter is a very simple one and is calculated as follows:

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

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

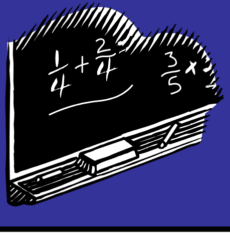
This is implemented as the simple smoothing filter

Blurs the image to remove noise

Other Means

There are different kinds of mean filters all of which exhibit slightly different behaviour:

- Geometric Mean
- Harmonic Mean
- Contraharmonic Mean



Other Means (cont…)

There are other variants on the mean which can give different performance

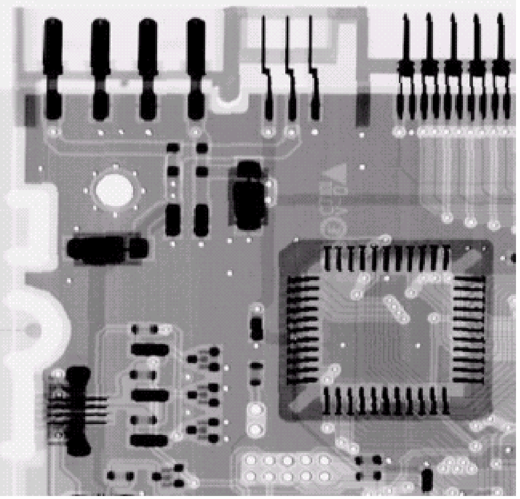
Geometric Mean:

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

Achieves similar smoothing to the arithmetic mean, but tends to lose less image detail

Noise Removal Examples

Original Image



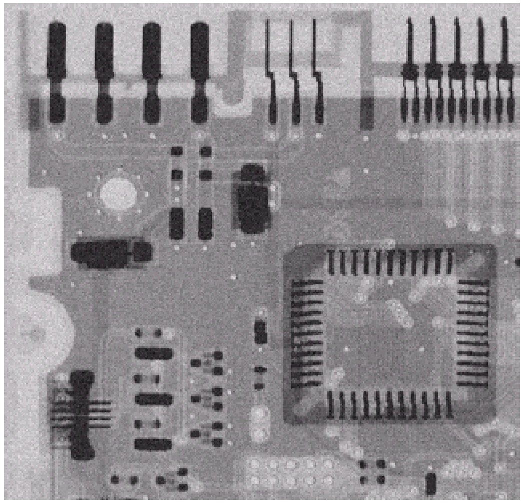
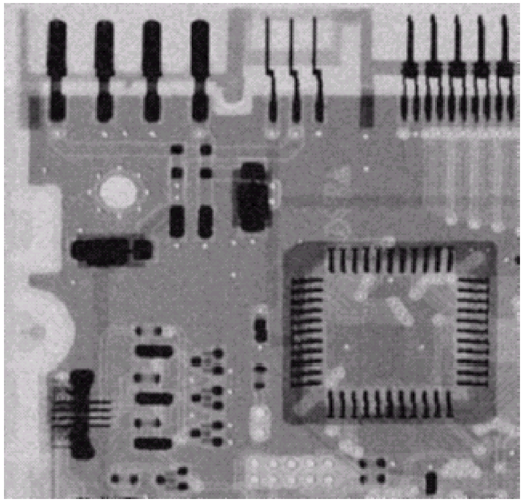
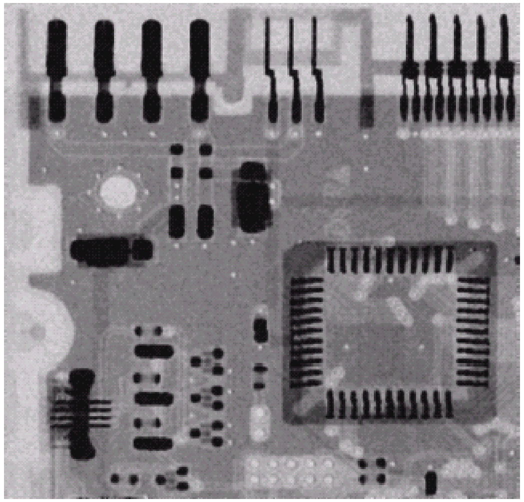


Image Corrupted By Gaussian Noise

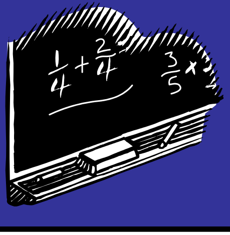
After A 3*3 Arithmetic Mean Filter





After A 3*3 Geometric Mean Filter

Images taken from Gonzalez & Woods, Digital Image Processing (2002)



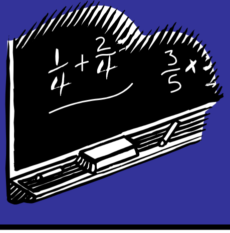
Other Means (cont…)

Harmonic Mean:

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

Works well for salt noise, but fails for pepper noise

Also does well for other kinds of noise such as Gaussian noise



Other Means (cont…)

Contraharmonic Mean:

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

Q is the *order* of the filter and adjusting its value changes the filter’s behaviour

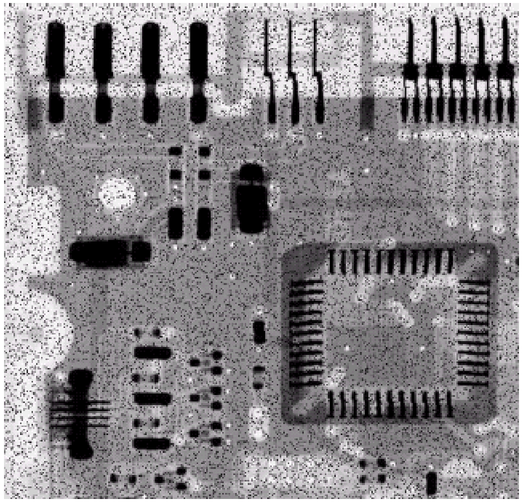
Positive values of *Q* eliminate pepper noise

Negative values of *Q* eliminate salt noise

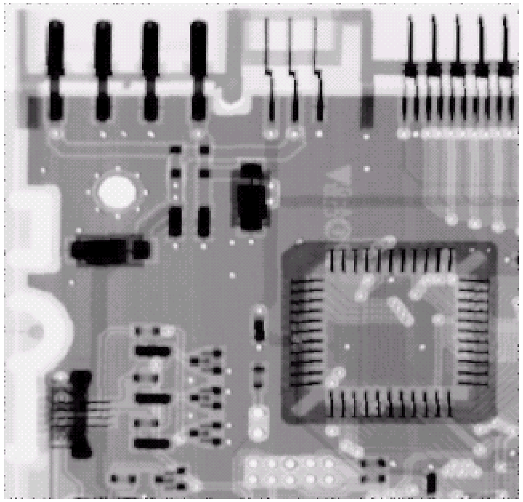
Noise Removal Examples (cont…)

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Corrupted By Pepper Noise



Result of Filtering Above With 3*3 Contraharmonic Q=1.5



Noise Removal Examples (cont…)

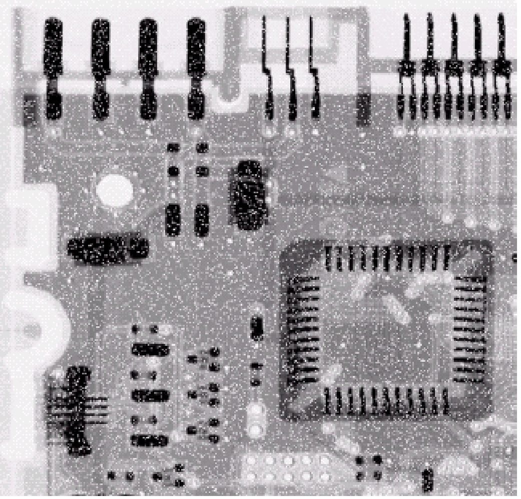
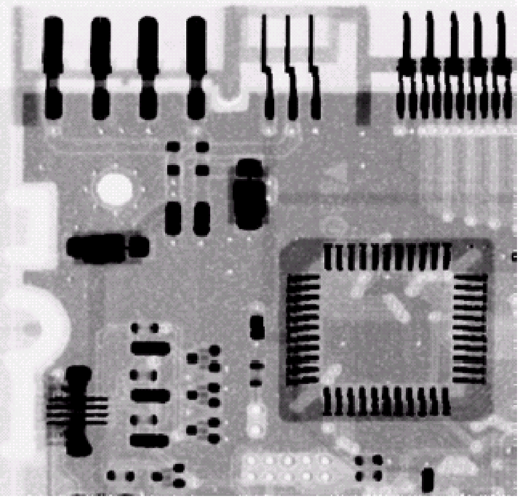


Image Corrupted By Salt Noise

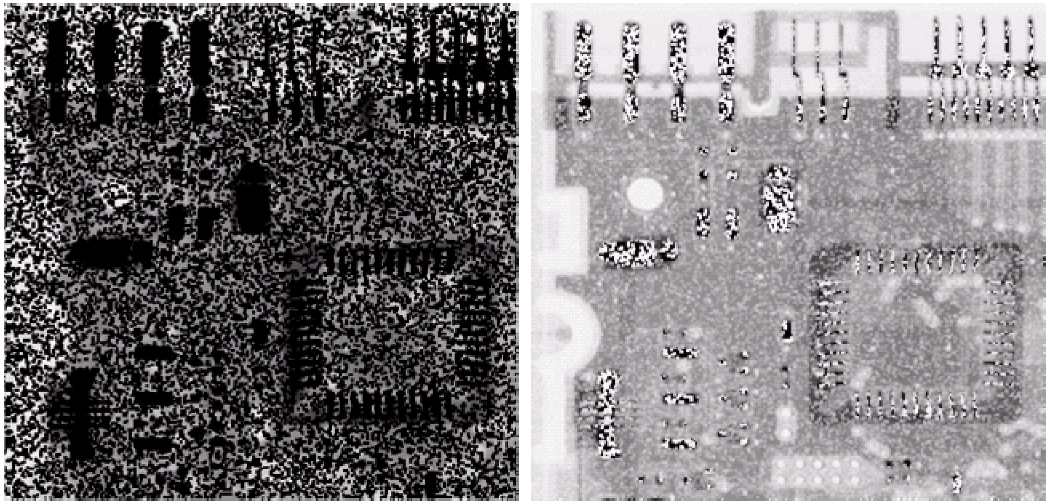


Result of Filtering Above With 3*3 Contraharmonic Q=-1.5

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

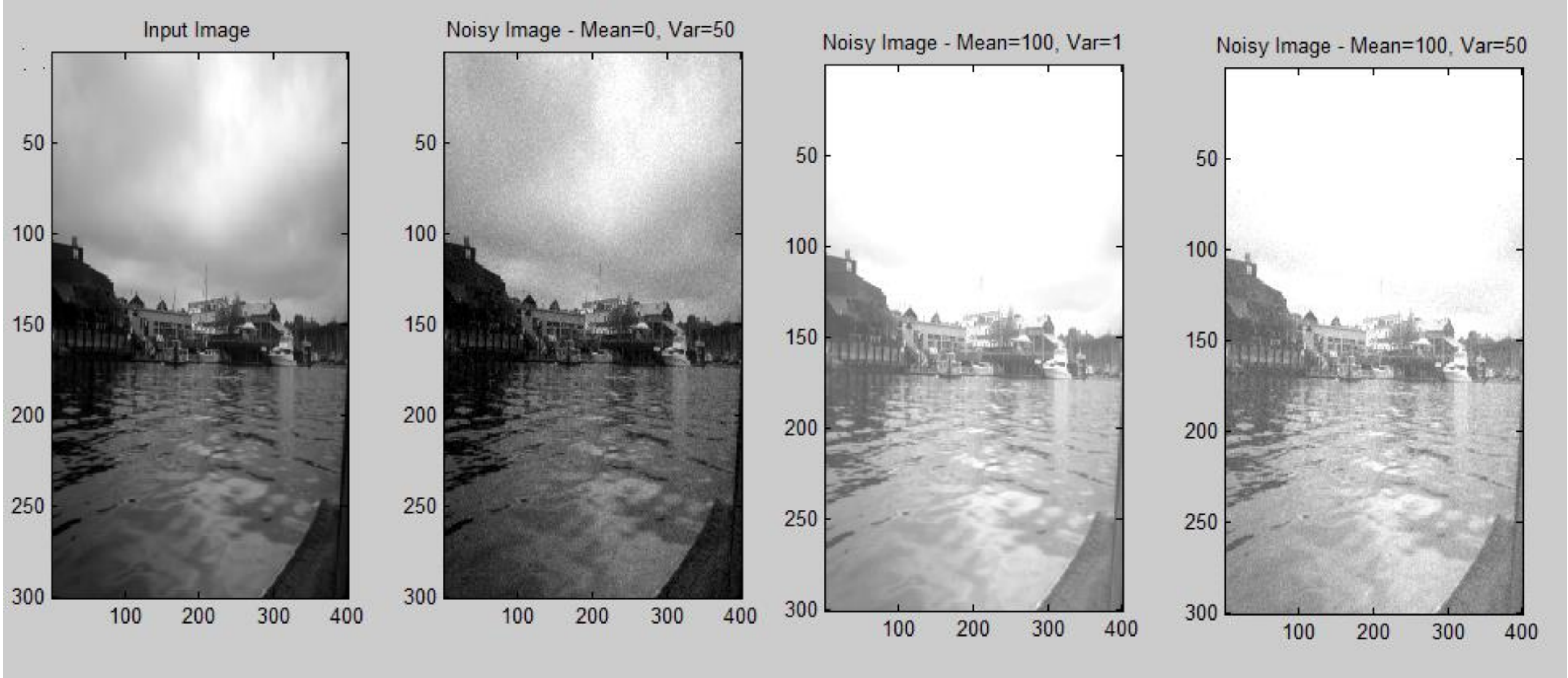
Contraharmonic Filter: Caution!

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

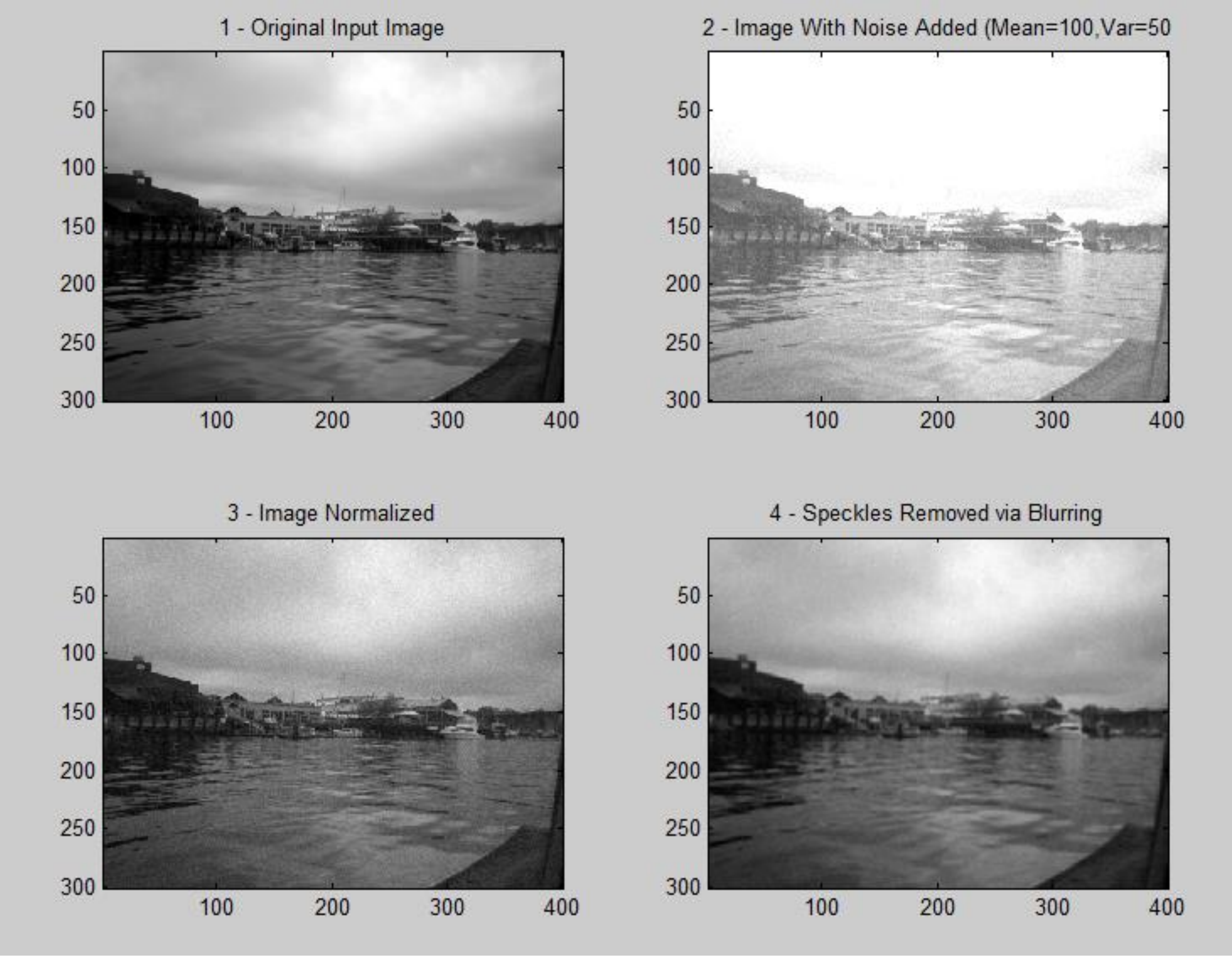


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Filtering of Gaussian noise


Input Image
Noisy Image - Mean=0, Var=50
Noisy Image - Mean=100, Var=1
Noisy Image - Mean=100, Var=50
50
100
150
200
250
300
100
200
300
400

Filtering of Gaussian noise (contd.)


1 - Original Input Image
2 - Image With Noise Added (Mean=100,Var=50
3 - Image Normalized
4 - Speckles Removed via Blurring
50
100
150
200
250
300
100
200
300
400

Rank Filters (non linear filtering)

Spatial filters that are based on ordering the pixel values that make up the neighbourhood operated on by the filter

Useful spatial filters include

- – Median filter
- – Max and min filter
- – Midpoint filter
- – Alpha trimmed mean filter

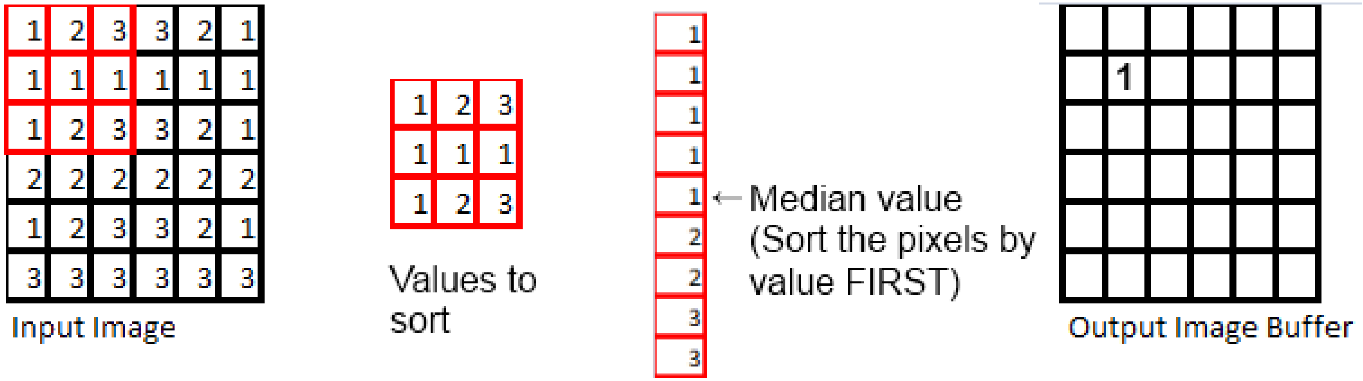
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Median Filter

Median Filter:

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

- Excellent at noise removal, without the smoothing effects that can occur with other smoothing filters
- Particularly good when salt and pepper noise is present



1	2	3	3	2	1
1	1	1	1	1	1
1	2	3	3	2	1
2	2	2	2	2	2
1	2	3	3	2	1
3	3	3	3	3	3

Input Image

1	2	3
1	1	1
1	2	3

Values to sort

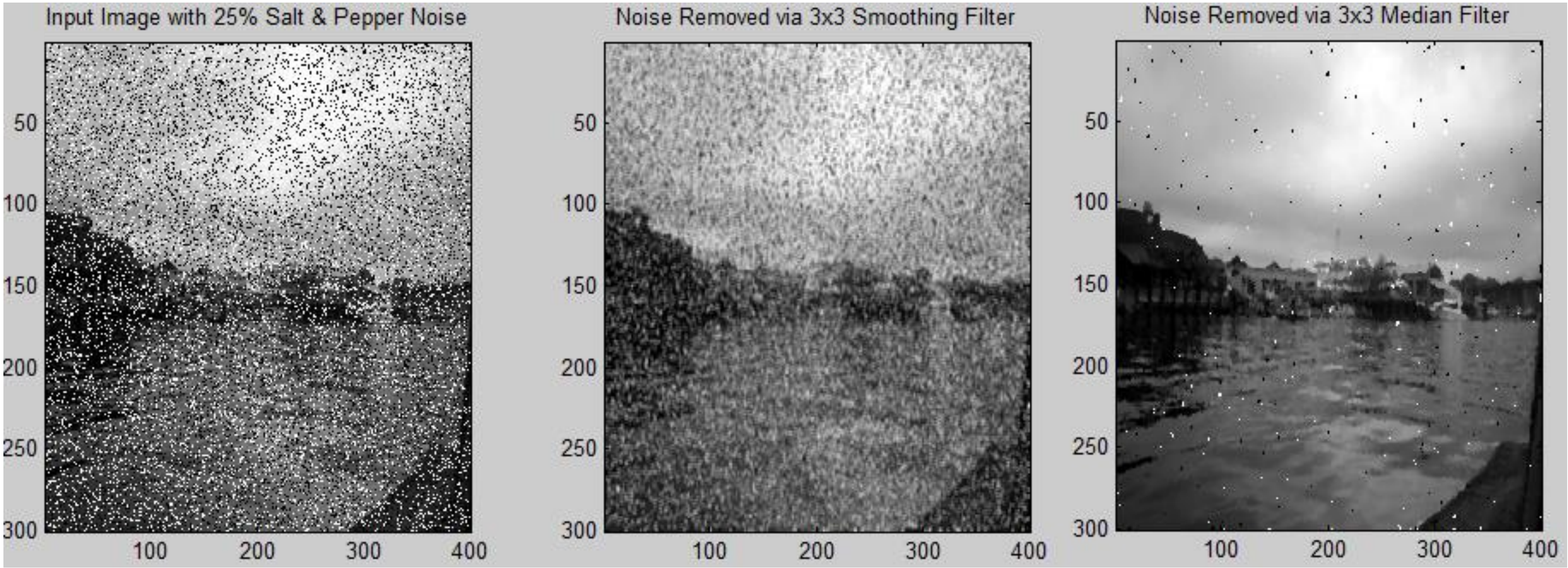
1
1
1
1
1
2
2
3
3

← Median value (Sort the pixels by value FIRST)

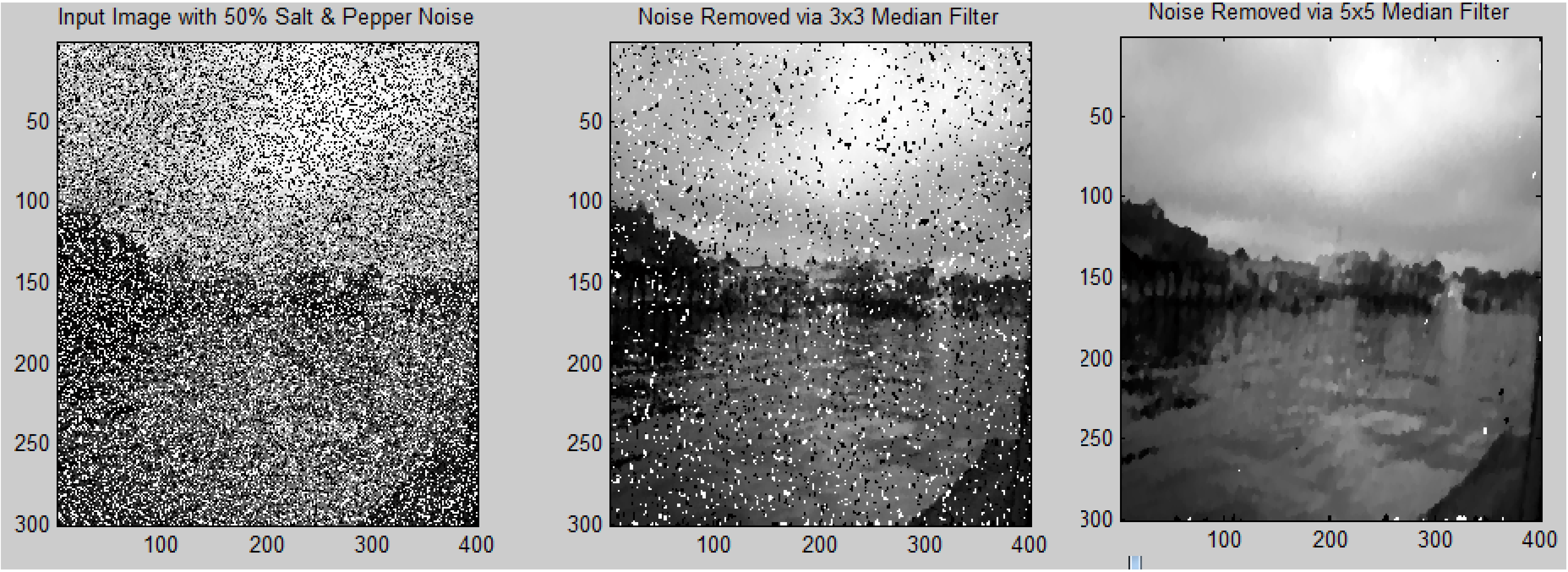
	1				

Output Image Buffer

Mean Vs. Median filter


Input Image with 25% Salt & Pepper Noise
Noise Removed via 3x3 Smoothing Filter
Noise Removed via 3x3 Median Filter
50
100
150
200
250
300
100
200
300
400

Noise Removal Examples



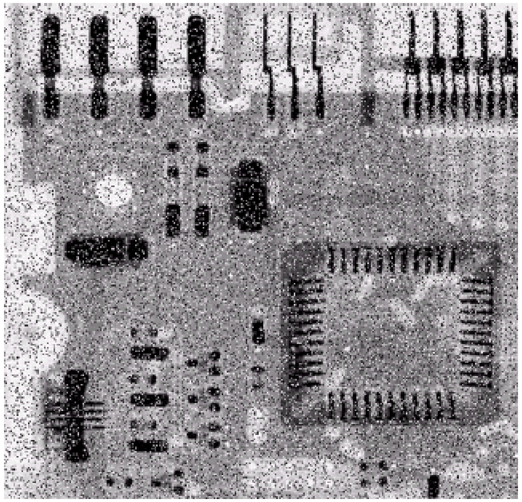
Input Image with 50% Salt & Pepper Noise

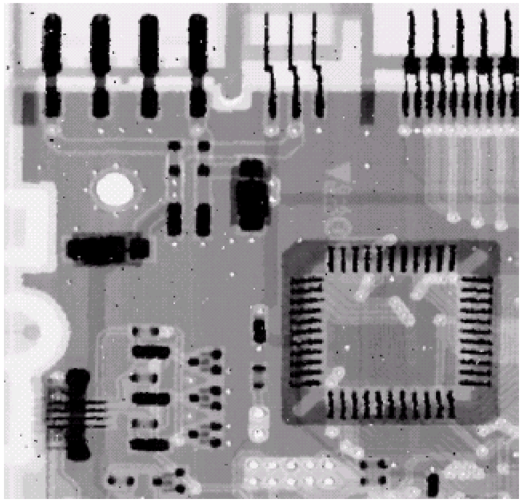
Noise Removed via 3x3 Median Filter

Noise Removed via 5x5 Median Filter

Noise Removal Examples (contd)

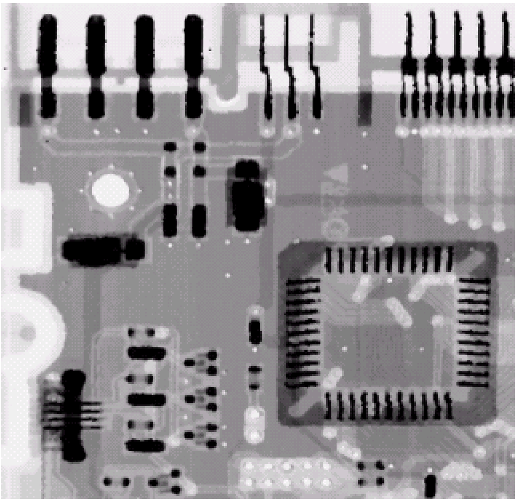
Image Corrupted By Salt And Pepper Noise

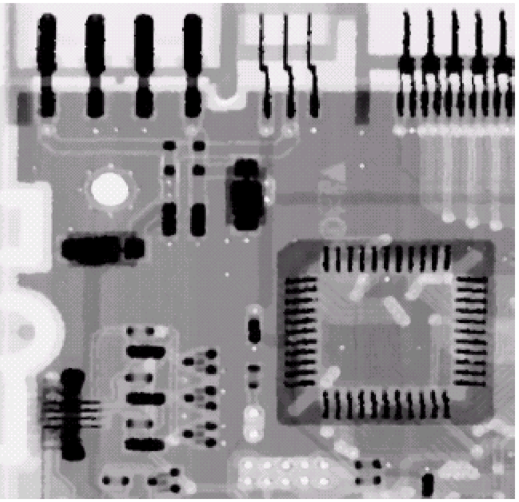




Result of 1 Pass With A 3*3 Median Filter

Result of 2 Passes With A 3*3 Median Filter





Result of 3 Passes With A 3*3 Median Filter

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Max and Min Filter

Max Filter:

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

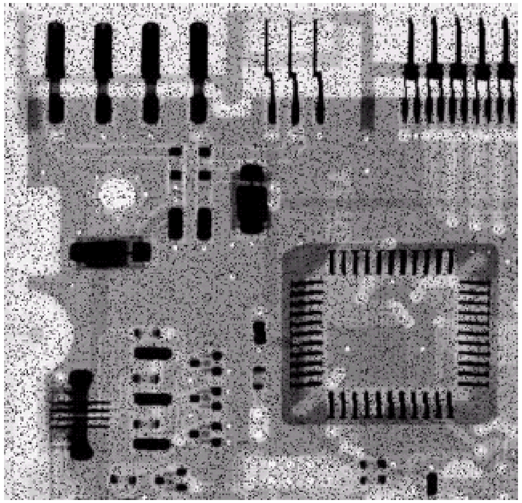
Min Filter:

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

Max filter is good for pepper noise and min is good for salt noise

Noise Removal Examples (cont…)

Image Corrupted By Pepper Noise



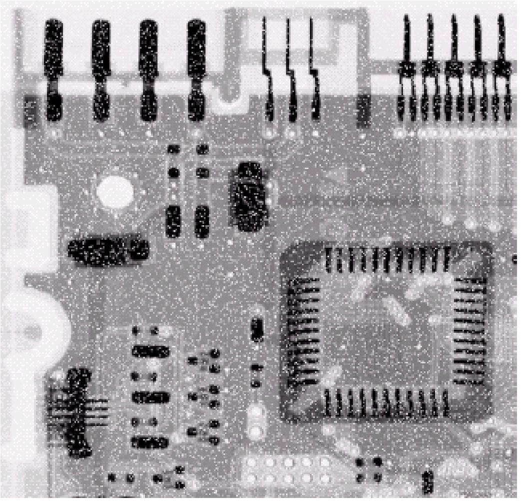
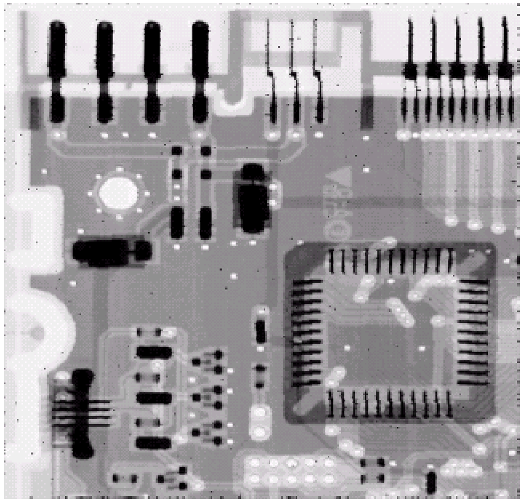
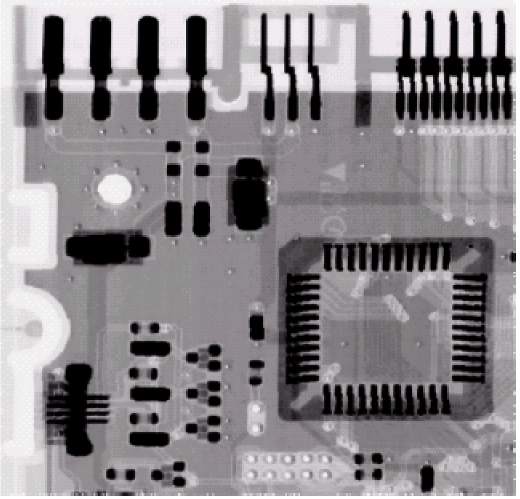


Image Corrupted By Salt Noise

Result Of Filtering Above With A 3*3 Max Filter





Result Of Filtering Above With A 3*3 Min Filter

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Alpha-Trimmed Mean Filter

Alpha-Trimmed Mean Filter:

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

We can delete the *d/2* lowest and *d/2* highest grey levels

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

Noise Removal Examples (cont…)

Image Corrupted By Uniform Noise

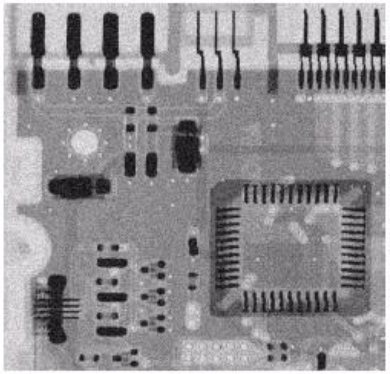
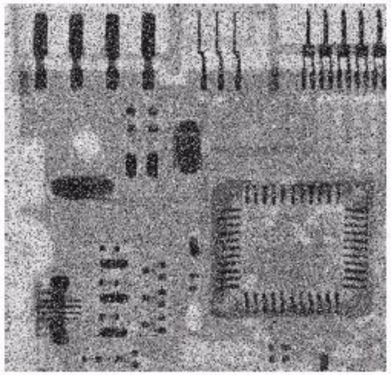
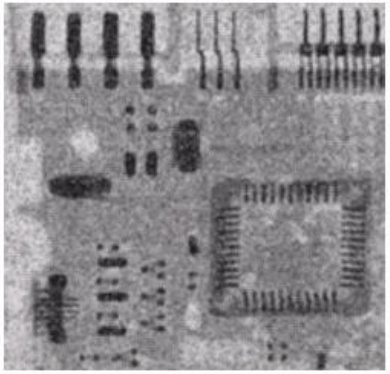


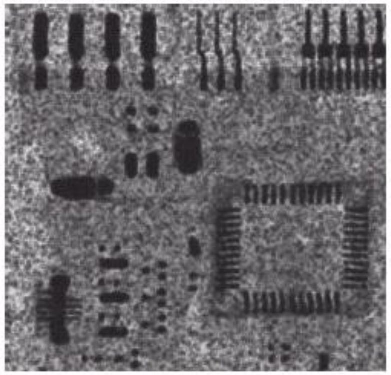
Image Further Corrupted By Salt and Pepper Noise



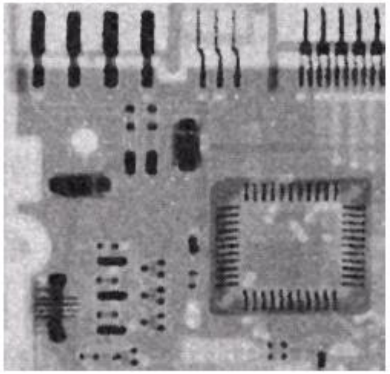
Filtered By 5*5 Arithmetic Mean Filter



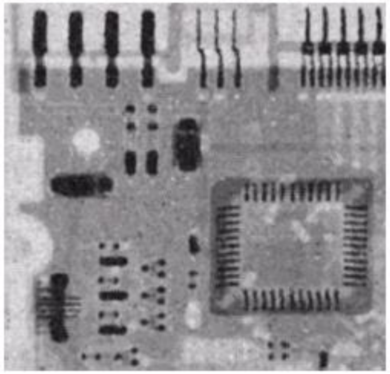
Filtered By 5*5 Geometric Mean Filter



Filtered By 5*5 Median Filter



Filtered By 5*5 Alpha-Trimmed Mean Filter



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

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 behaviour of **adaptive filters** changes depending on the characteristics of the image inside the filter region

Adaptive filter: Neighborhood-based

Adaptive local noise reduction filter

- Response based on 4 quanties
  - Local variance, variance of noise, g(x,y), and local mean
- Behavior of filter
  - If variance of noise is zero, return g(x,y)
  - If the local variance is high compared to the variance of noise, return a value close to g(x,y)
  - If the two variances are equal, return the mean value

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

Adaptive filter: Neighborhood-based

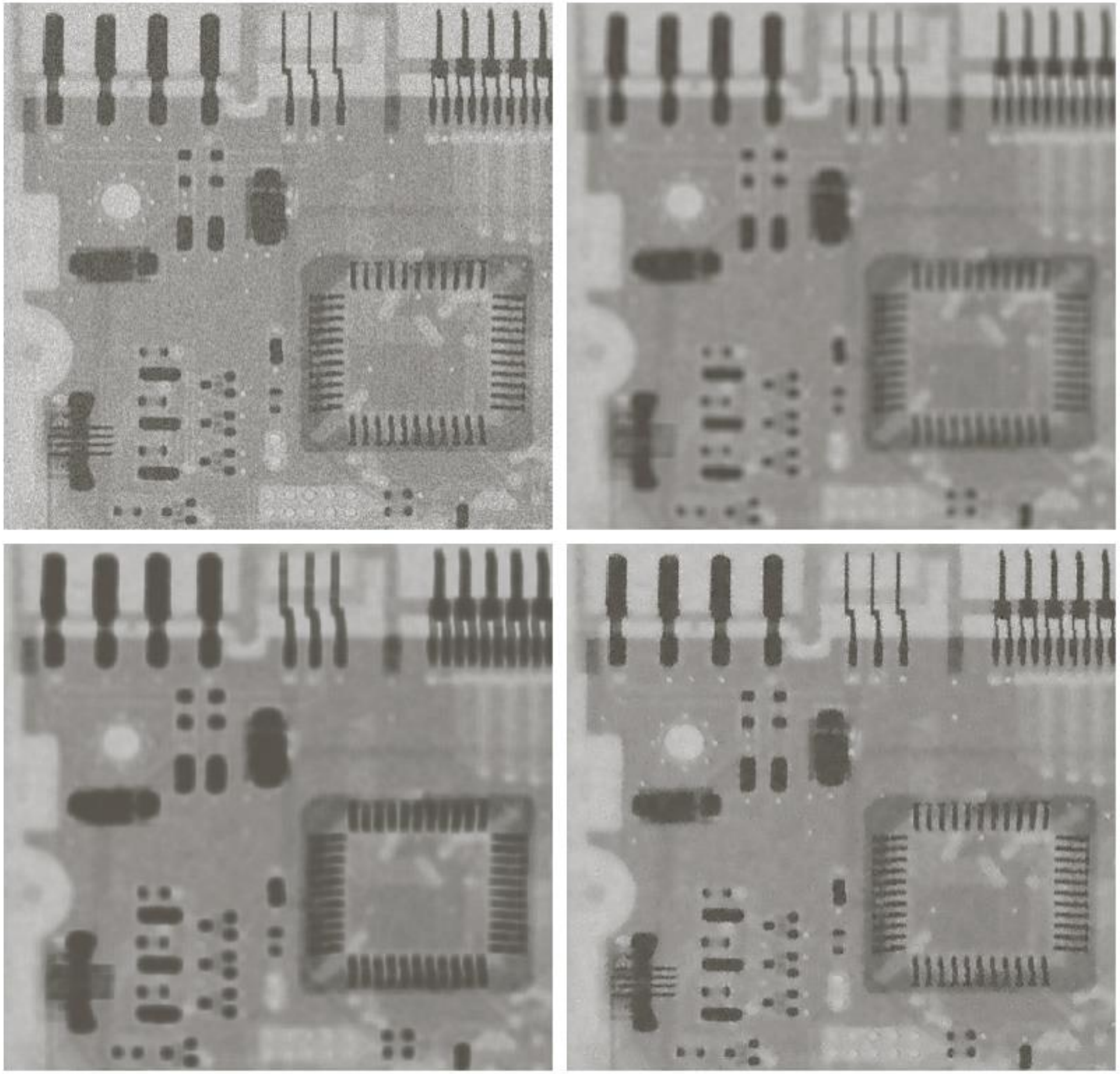


FIGURE 5.13 (a) Image corrupted by additive Gaussian noise of zero mean and variance 1000. (b) Result of arithmetic mean filtering. (c) Result of geometric mean filtering. (d) Result of adaptive noise reduction filtering. All filters were of size 7 × 7.

Adaptive Median Filtering

The median filter performs relatively well on impulse noise as long as the spatial density of the impulse noise is not large

The adaptive median filter can handle much more spatially dense impulse noise, and does less distortion

The key insight in the adaptive median filter is that the filter size changes depending on the characteristics of the image

Adaptive Median Filtering (cont…)

The adaptive median filter has following purposes:

- Remove spatially dense impulse noise
- Reduce distortion

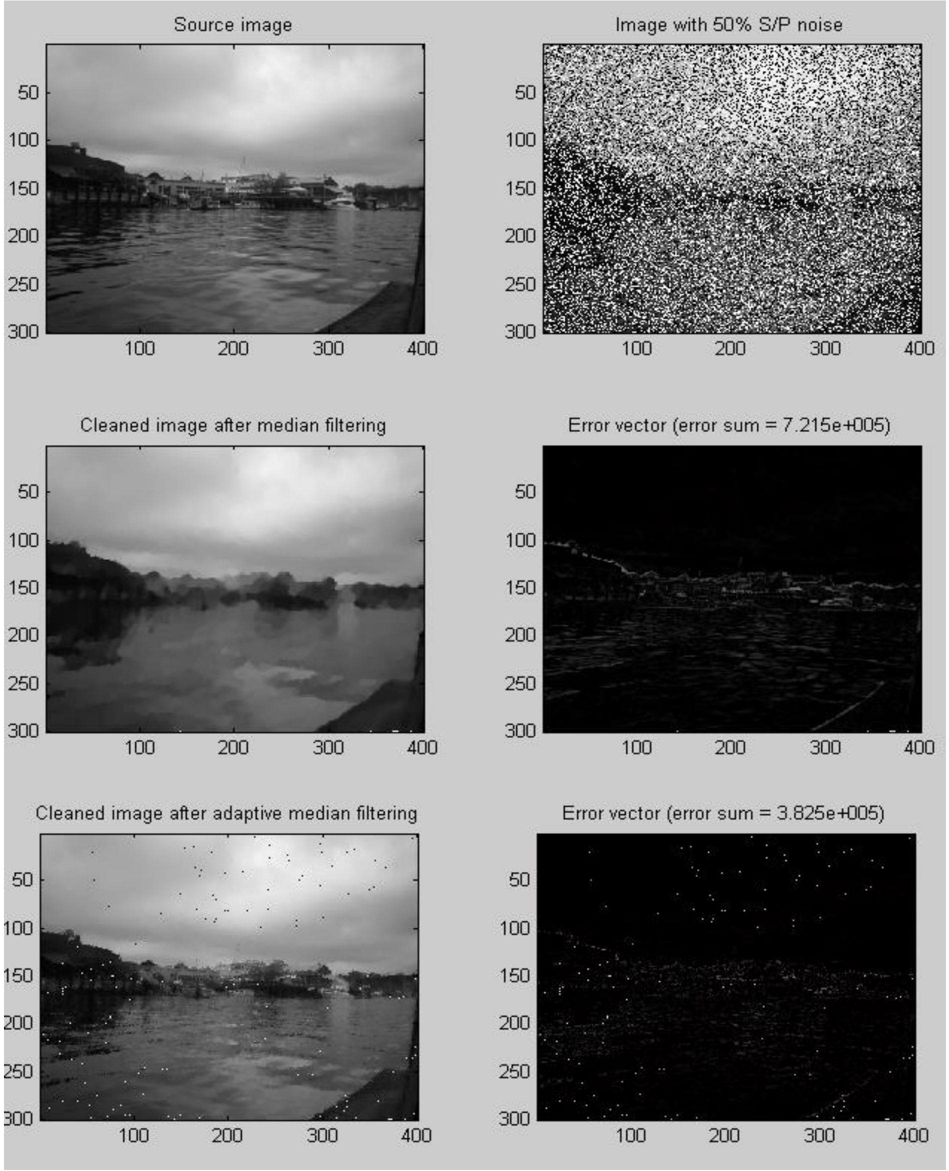
First examine the following notation:

- z_{min} = minimum grey level in S_{xy}
- z_{max} = maximum grey level in S_{xy}
- z_{med} = median of grey levels in S_{xy}
- z_{xy} = grey level at coordinates (x, y)
- S_{max} =maximum allowed size of S_{xy}

Adaptive Median Filtering (cont…)

```
Level A:  $A1 = z_{med} - z_{min}$
          $A2 = z_{med} - z_{max}$
          If $A1 > 0$ and $A2 < 0$, Go to level B
          Else increase the window size
          If window size $\leq S_{max}$ repeat level A
          Else output $z_{med}$
Level B:  $B1 = z_{xy} - z_{min}$
          $B2 = z_{xy} - z_{max}$
          If $B1 > 0$ and $B2 < 0$, output $z_{xy}$
          Else output $z_{med}$
```

Simple Adaptive Median Filtering Example


Source image
Image with 50% S/P noise
Cleaned image after median filtering
Error vector (error sum = 7.215e+005)
Cleaned image after adaptive median filtering
Error vector (error sum = 3.825e+005)
50
100
150
200
250
300
100
200
300
400

Using only level B and filter 7 x 7

Adaptive Median Filtering Examples

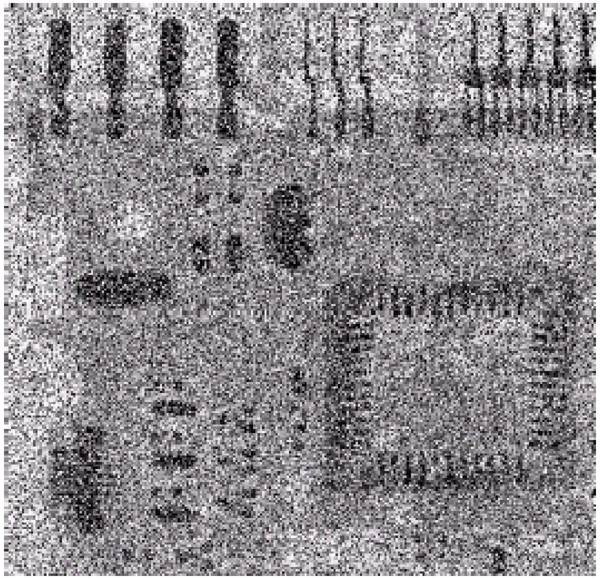
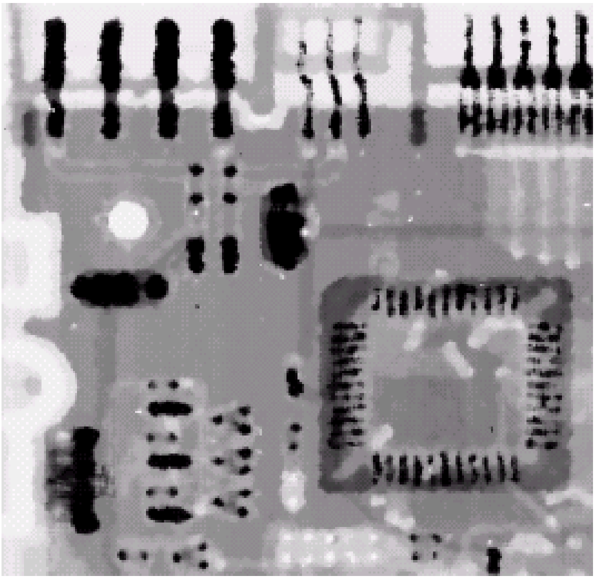
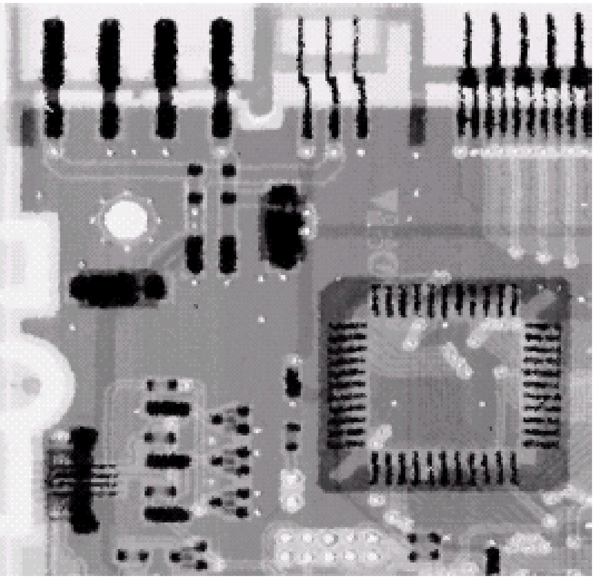


Image corrupted by salt and pepper noise with probabilities $P_a = P_b$=0.25



Result of filtering with a 7 * 7 median filter



Result of adaptive median filtering with S_{max} = 7

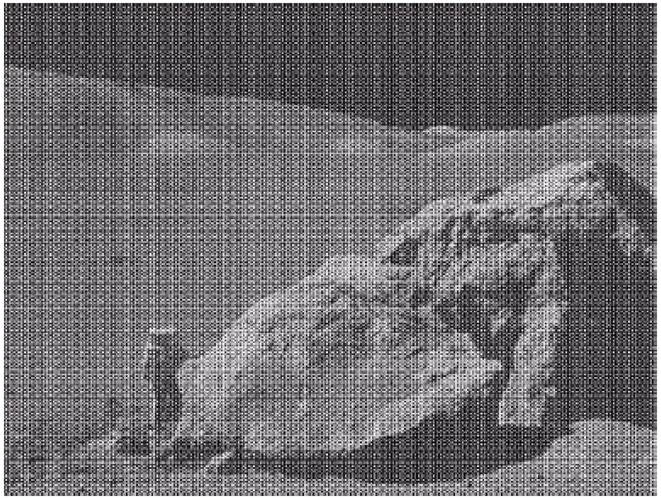
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

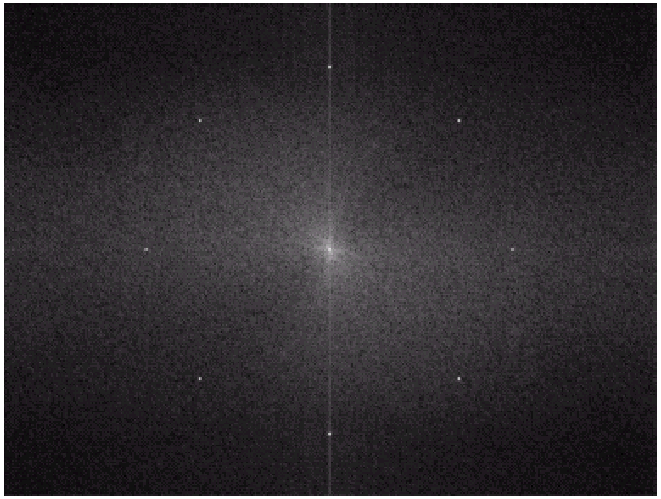
Periodic Noise removal

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





Band Reject Filters

Removing periodic noise from an image involves removing a particular range of frequencies from that image

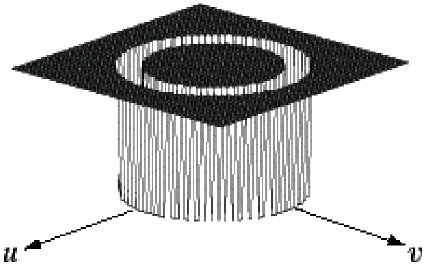
Band reject filters can be used for this purpose

An ideal band reject filter is given as follows:

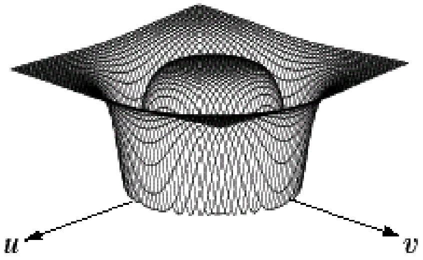
$$H(u,v) = \begin{cases} 1 & if \; D(u,v) < D_0 - \frac{W}{2} \\ 0 & if \; D_0 - \frac{W}{2} \le D(u,v) \le D_0 + \frac{W}{2} \\ 1 & if \; D(u,v) > D_0 + \frac{W}{2} \end{cases}$$

Band Reject Filters (cont…)

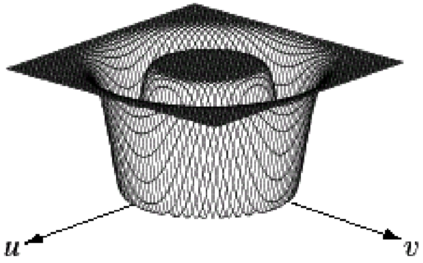
The ideal band reject filter is shown below, along with Butterworth and Gaussian versions



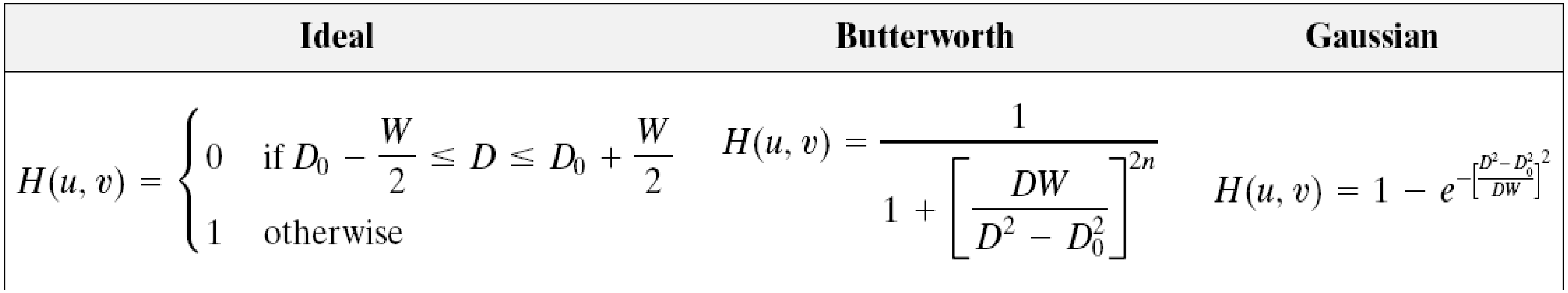
Ideal Band Reject Filter



Butterworth Band Reject Filter (of order 1)



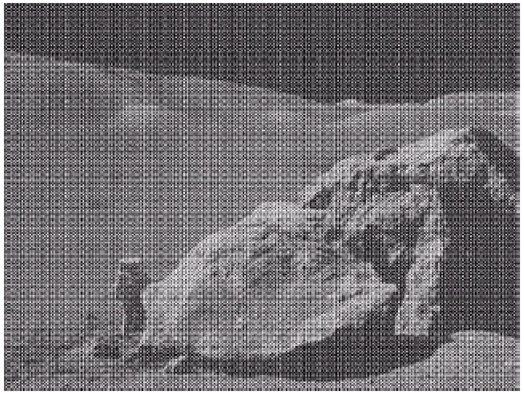
Gaussian Band Reject Filter



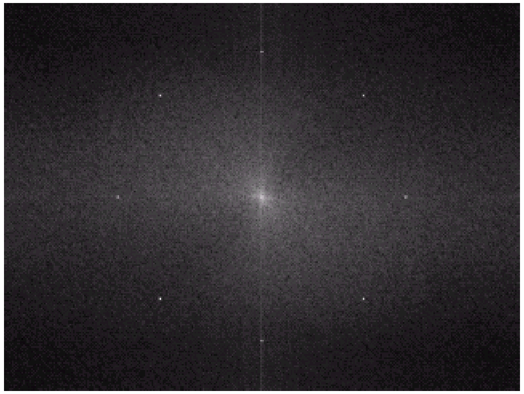
Ideal	Butterworth	Gaussian
$H(u, v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \leq D \leq D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u, v) = \dfrac{1}{1 + \left[\dfrac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u, v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$

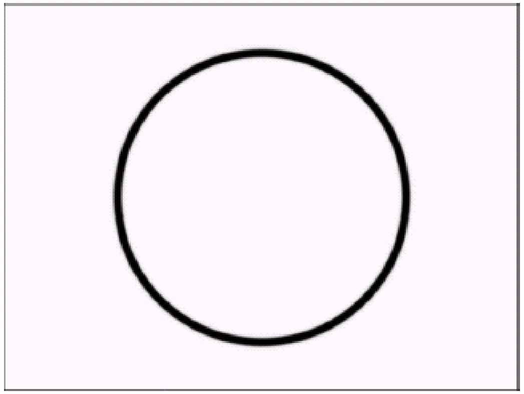
Band Reject Filter Example

Image corrupted by sinusoidal noise

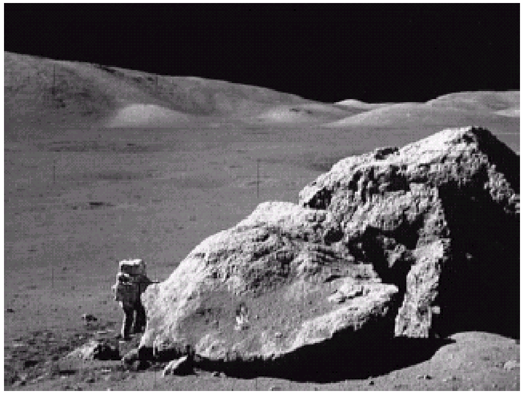


Fourier spectrum of corrupted image

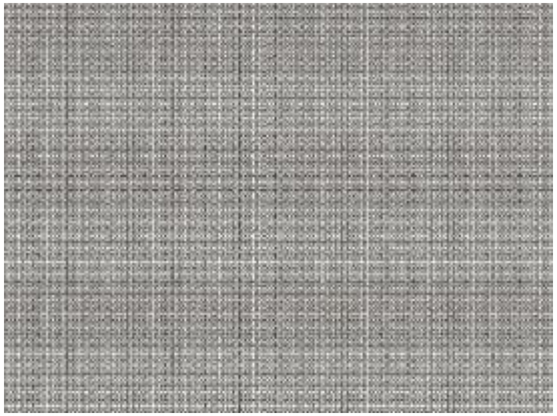




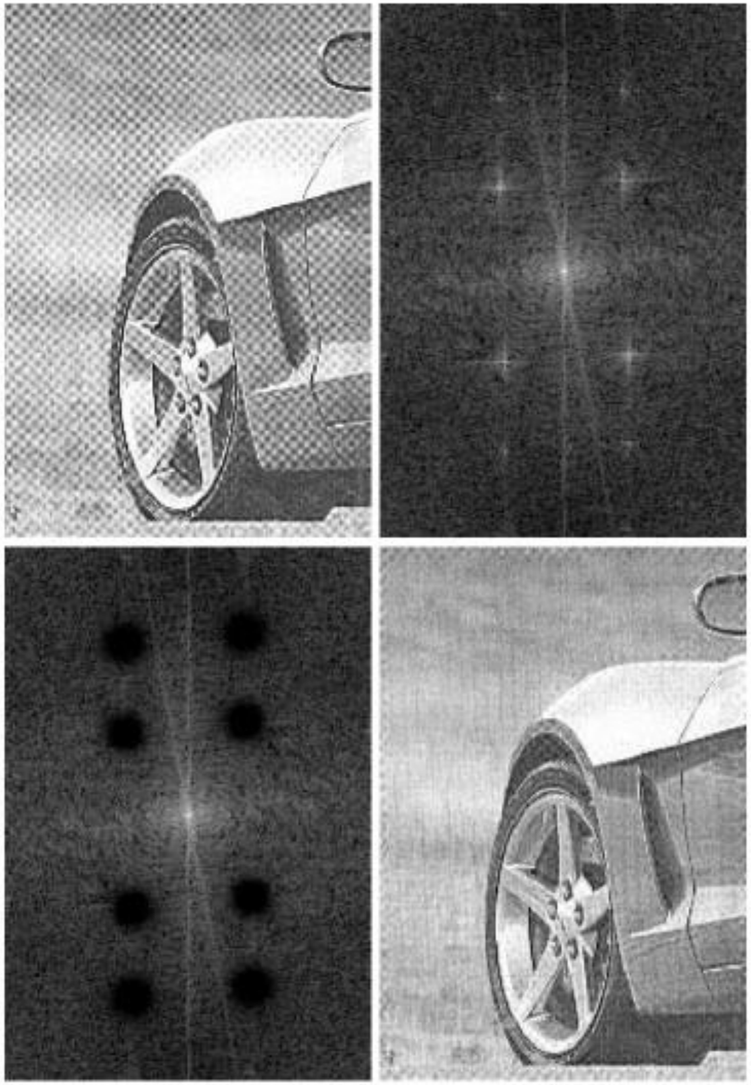
Butterworth band reject filter



Filtered image



Notch Filter



a b
c d

FIGURE 4.64 (a) Sampled newspaper image showing a moiré pattern. (b) Spectrum. (c) Butterworth notch reject filter multiplied by the Fourier transform. (d) Filtered image.