Image Restoration

- Image restoration vs. image enhancement

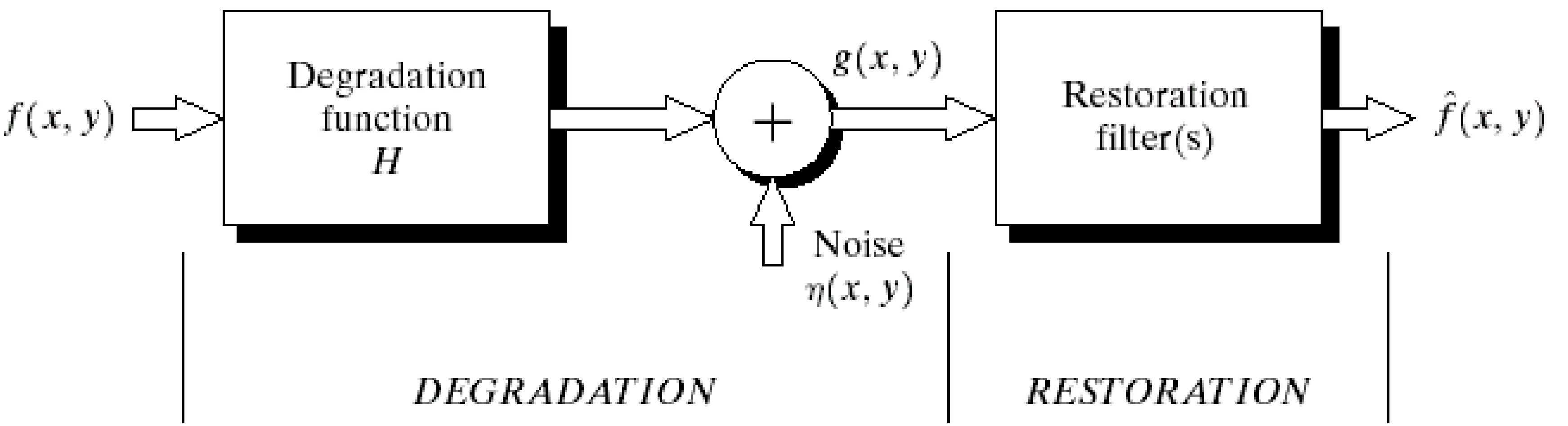
  - Enhancement:
    - largely a subjective process
    - Priori knowledge about the degradation is not a must
    - Procedures are heuristic and take advantage of the psychophysical aspects of human visual system

  - Restoration:
    - more an objective process
    - Images are degraded and inverse process is applied to recover the original image.
    - Tries to recover the images by using the knowledge about the degradation

An Image Degradation Model

- Two types of degradation
  - Additive noise
    - Spatial domain restoration (denoising) techniques are preferred
  - Image blur
    - Frequency domain methods are preferred

A Model of the Image Degradation/Restoration Process



An Image Degradation Model

- In spatial domain, we model the degradation process by a degradation function $h(x,y)$, an additive noise term, $\eta(x,y)$, as

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

$f(x,y)$ is the (input) image free from any degradation

$g(x,y)$ is the degraded image

$*$ is the convolution operator

The goal is to obtain an estimate of $f(x,y)$ according to the knowledge about the degradation function h and the additive noise η

What is Noise ?

- Image noise is random variation of brightness or color information in digital images, and usually an aspect of electronic noise.

Sources of Noise in Digital Images

- During Image acquisition and/or transmission

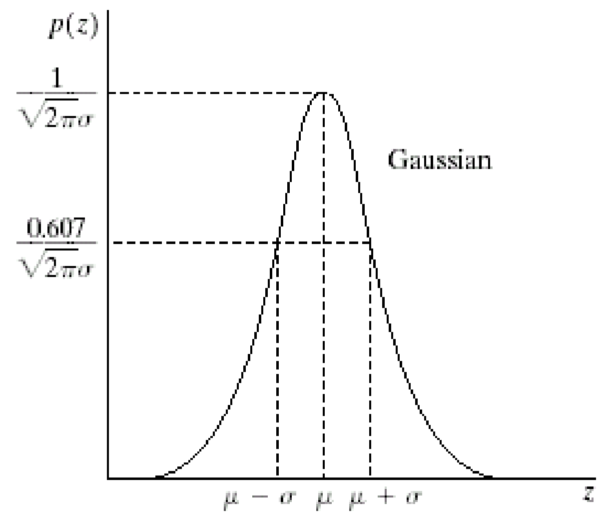
- Imaging Sensors are affected by environmental conditions and quality of sensing elements during Image acquisition.

- Images transmitted through wireless N/W might be corrupted as a result of lightning and other atmospheric disturbances.

Gaussian Noise (Normal Noise)

- Noise (image) can be classified according the distribution of the values of pixels (of the noise image) or its (normalized) histogram
- Gaussian noise is characterized by two parameters, μ (mean) and σ^2 (variance). The PDF of Gaussian Random variable z is given by

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



z is the intensity

μ is the mean value of z and

σ is the standard deviation

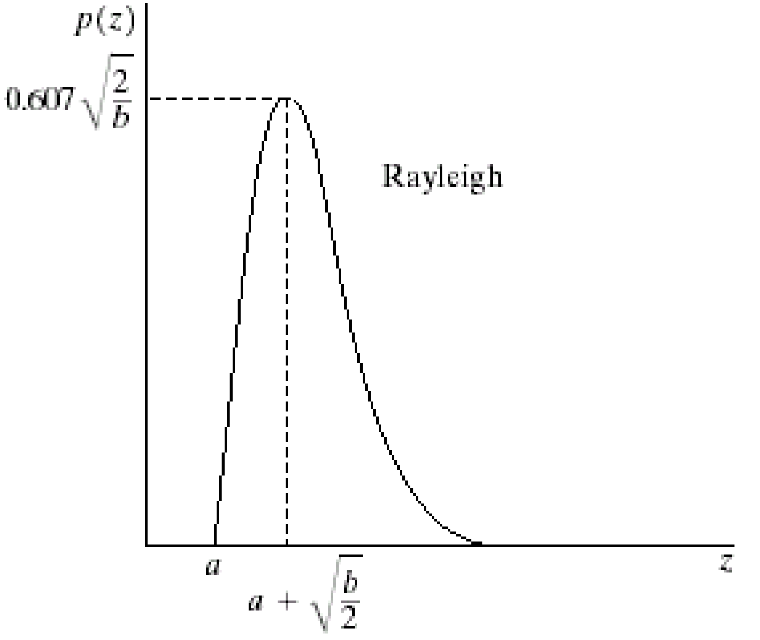
σ^2 is the variance

- 70% values of z fall in the range [(μ-σ),(μ+σ)]
- 95% values of z fall in the range [(μ-2σ),(μ+2σ)]

Rayleigh Noise Model

Rayleigh noise is given by

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



a and b can be obtained through mean and variance

$$\mu = a + \sqrt{\pi b / 4} \text{ and}$$

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

Usage: Real-time systems

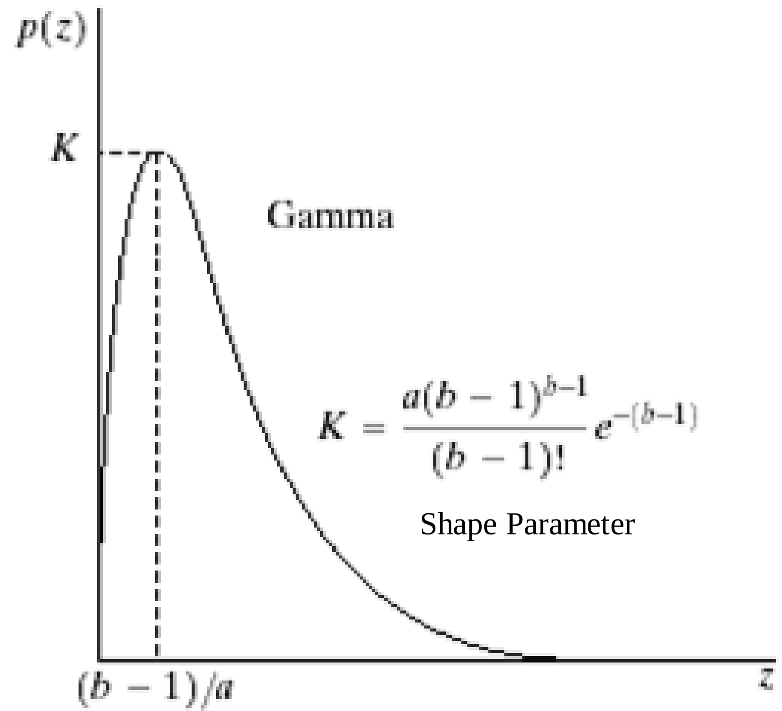
Erlang Noise Model (Gamma)

Erlang (Gamma) noise

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

Mean $\mu = b/a$ and

Variance $\sigma^2 = \dfrac{b}{a^2}$

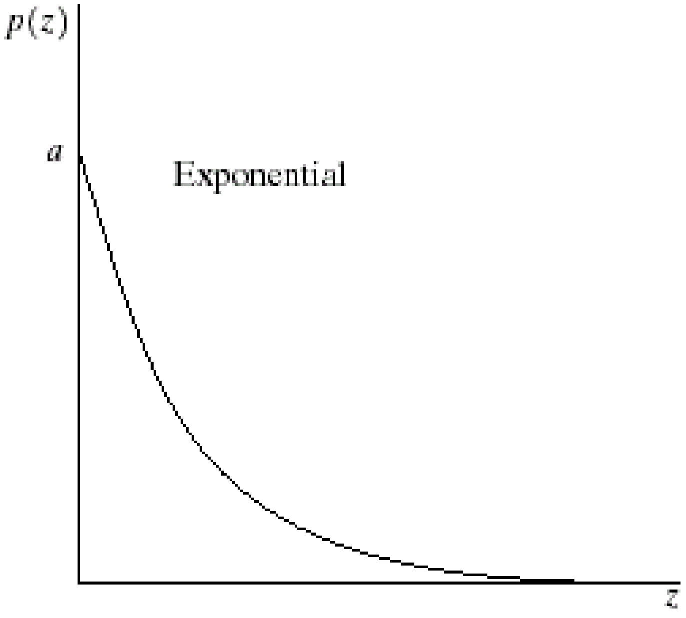


Shape Parameter

Exponential Noise Model (Predictive Noise)

- **Exponential** noise

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



$$\mu = 1/a \ \text{ and } \sigma^2 = \frac{1}{a^2}$$

Usage: Predictive Noise

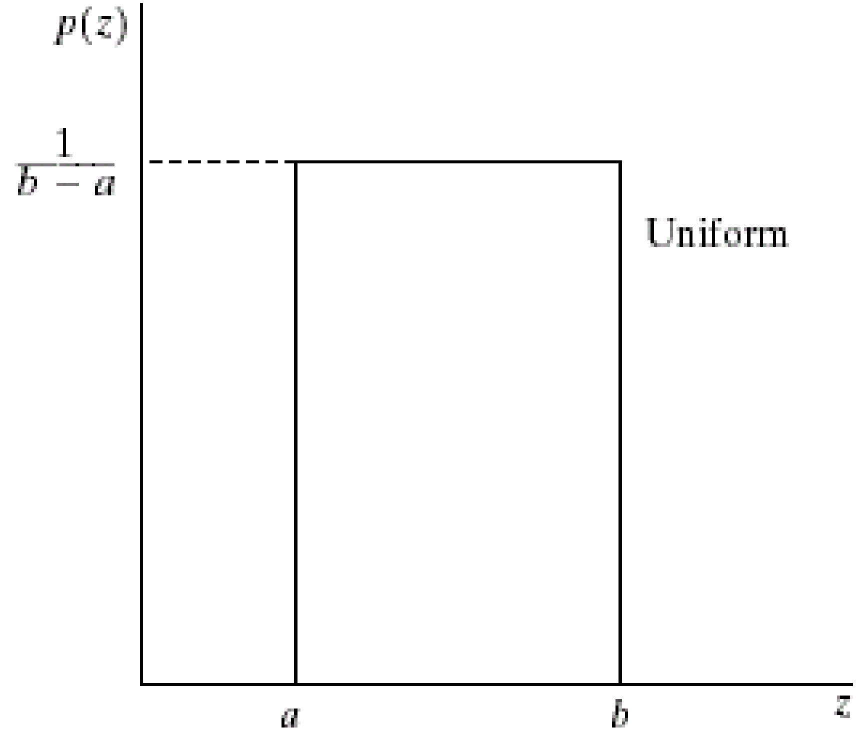
Uniform Noise Model (Quantization)

Uniform noise is given by

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

The mean and variance of this density are given by

$$\mu = (a+b)/2 \text{ and } \sigma^2 = \frac{(b-a)^2}{12}$$



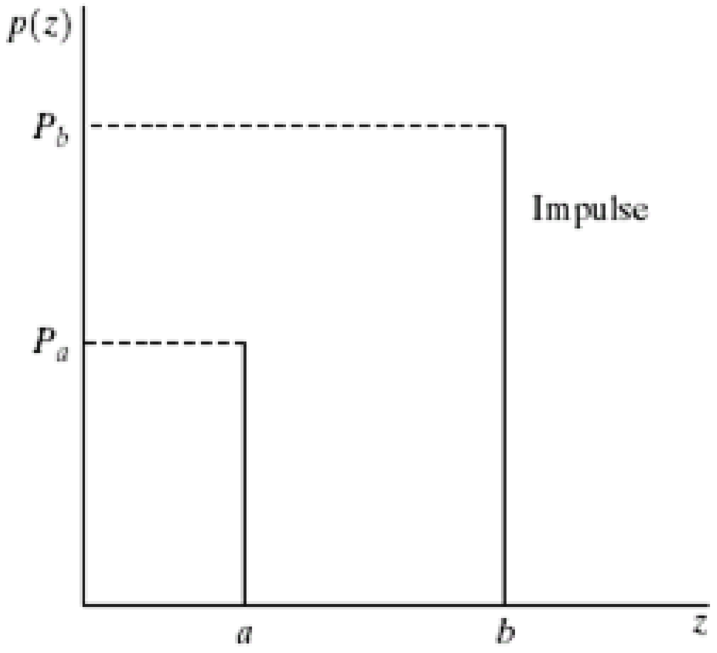
Usage: Quantization

Impulse Noise Model

- **Impulse** (salt-and-pepper) noise

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

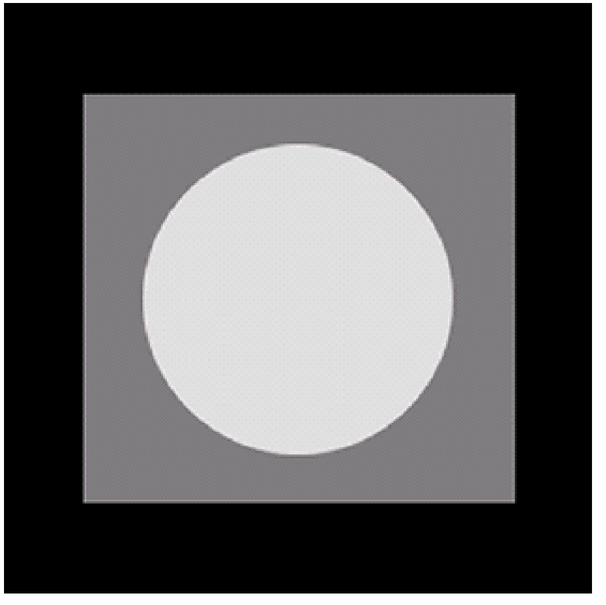
- Impulse noise can be positive or negative.
- Positive impulse appear white (salt).
- Negative impulse appear black (pepper).



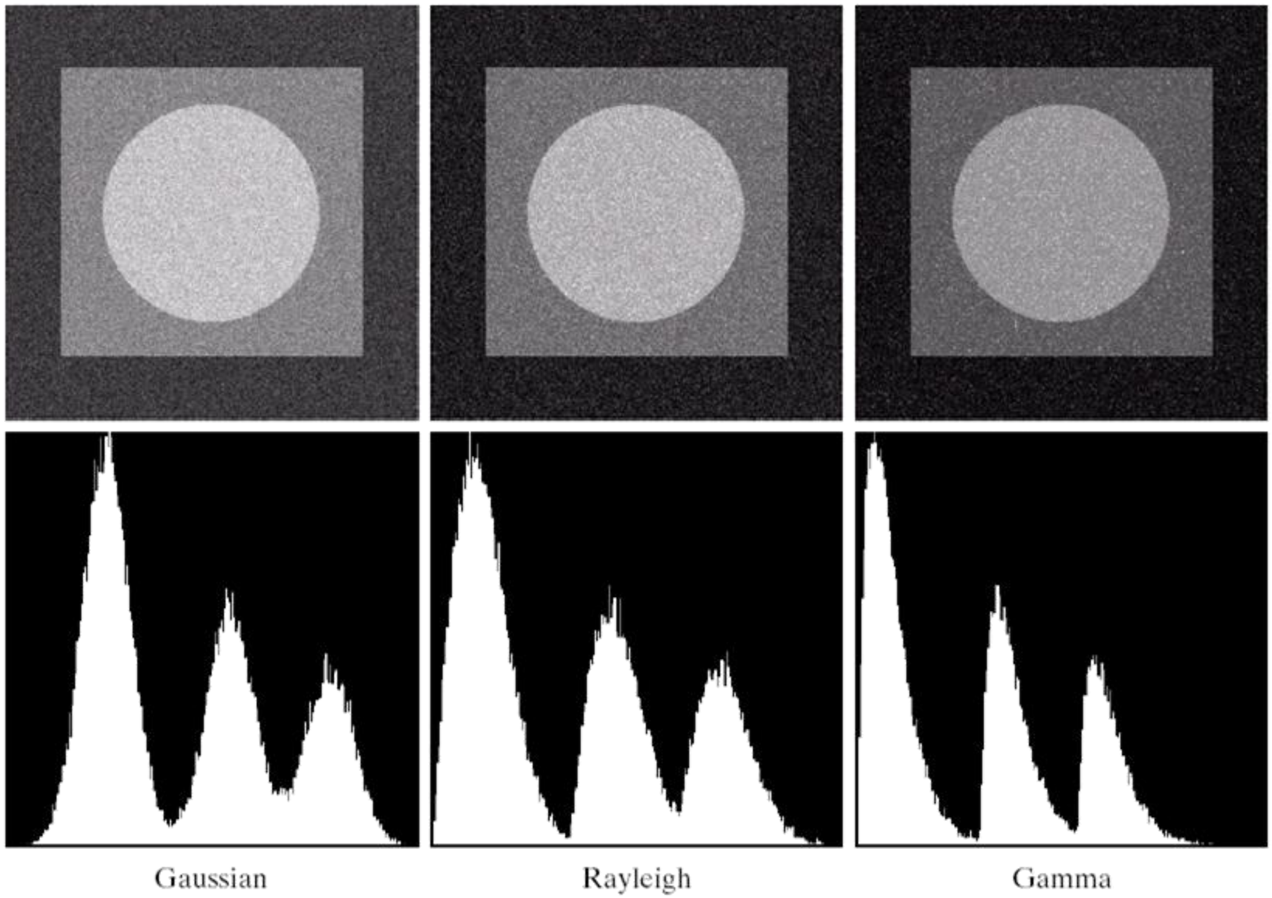
Impulse Noise Model

- If $b > a$, intensity b will appear as a light dot in the image
- If $a > b$, intensity a will appear as a dark dot in the image
- If $P_a = 0$ or $P_b = 0$, the impulse noise is called unipolar
- If neither probability is zero, and $P_a \approx P_b$, impulse noise will resemble randomly distributed salt and pepper granules
- a and b are expected to be saturated values implying that for an 8-bit image, $a = 0$ (black) and $b = 255$ (white)
- Found in situations with quick transitions, such as faulty switching during imaging.

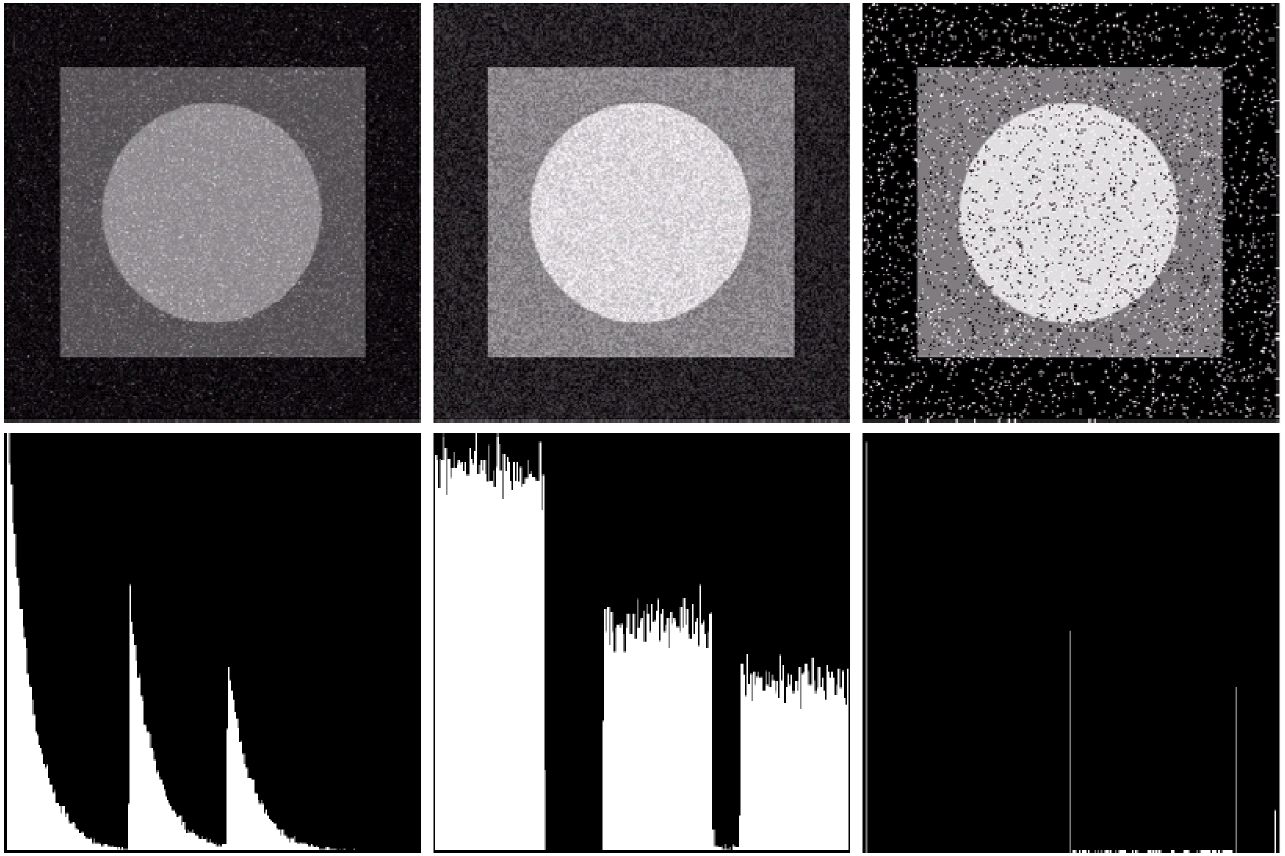
A Sample Image



Effect of Adding Noise to Sample Image



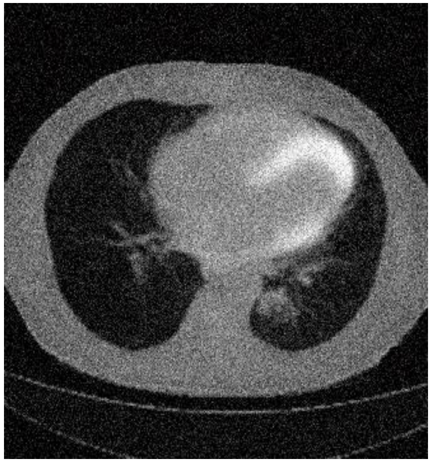
Effect of Adding Noise to Sample Image



Exponential — Uniform — Salt & Pepper

Gaussian Noise (Normal Noise)

```
>> i=imread('f:\pet.jpg');
>> a=rgb2gray(i);
>> b=imnoise(a,'gaussian');
>> figure, imshow(i)
>> figure, imshow(a)
>> figure, imshow(b)
```



Applicability of various Noise Models

- **Gaussian Noise:** Electrical circuit noise and Sensor noise due to poor illumination and/or high temperature.
- **Rayleigh Noise:** Characterize noise phenomena in range image.
- **Exponential and Gamma Noise:** Laser imaging.
- **Impulse Noise:** Occur when quick transients take place during imaging.
- **Uniform Noise:** the least descriptive of practical situations.

Restoration of Noise - Filters

- Mean Filters
  - Arithmetic Mean Filter
  - Geometric Mean Filter
  - Harmonic Mean Filter
  - Contraharmonic Filter
- Order-Statistics Filters
  - Median Filter
  - Max and Min Filter
  - Midpoint Filter
  - Alpha-trimmed Mean Filter

Restoration Filters

Mean filters: **Arithmetic Mean Filter**

- Smooth local variations in the image
- Noise is reduced as a result of blurring

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

- g(s,t) is the corrupted image
- $S_{x,y}$ is the mask [window size – (m x n)]

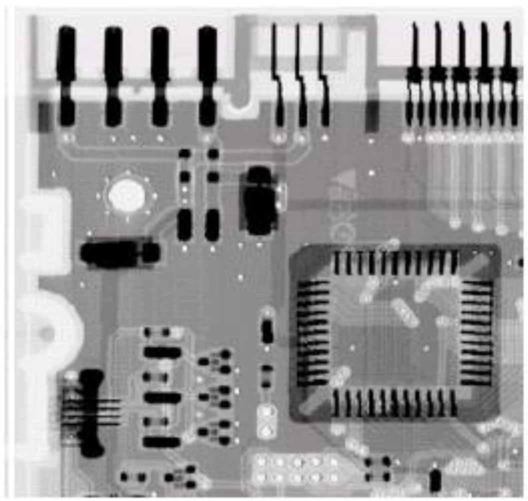
  – Causes a certain amount of blurring, (proportional to window size), thereby reducing the noise effect.

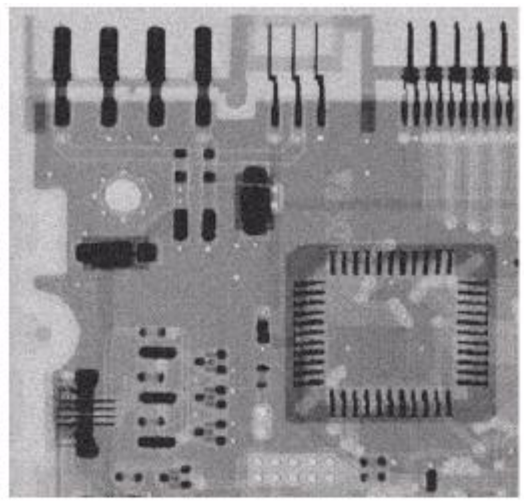
  – Works best for **Gaussian, Uniform or Erlang Noise**

Restoration Filters

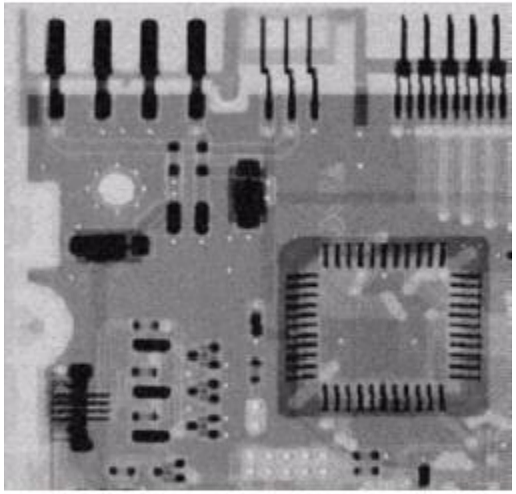
Mean filters: **Arithmetic Mean Filter**

Input Image





Corrupted by Gaussian Noise



Result of Arithmetic Mean Filter [3 x3]

Restoration Filters

Mean filters: **Geometric Mean Filter**

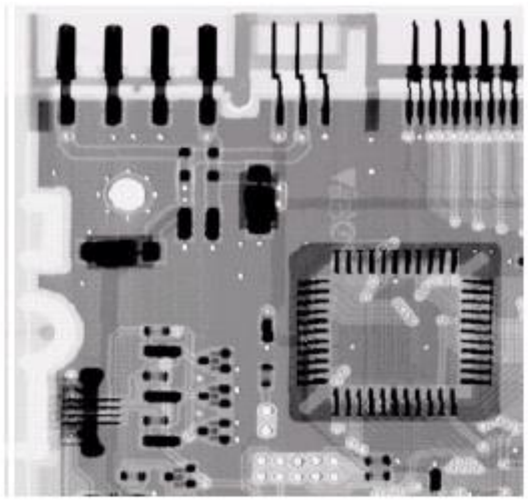
- Looses less details when compared to Arithmetic.
- Achieves smoothening.

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

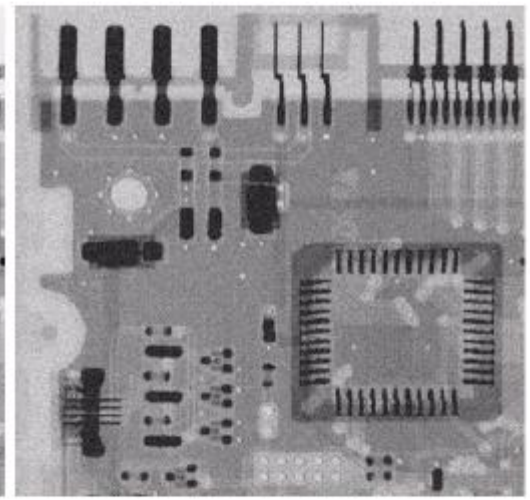
- g(s,t) is the corrupted image
- $S_{x,y}$ is the mask [window size – (m x n)]
  - A variation of arithmetic mean filter.
  - Primarily used on images with Gaussian noise.
  - Retains image details better than the arithmetic mean.

Restoration Filters

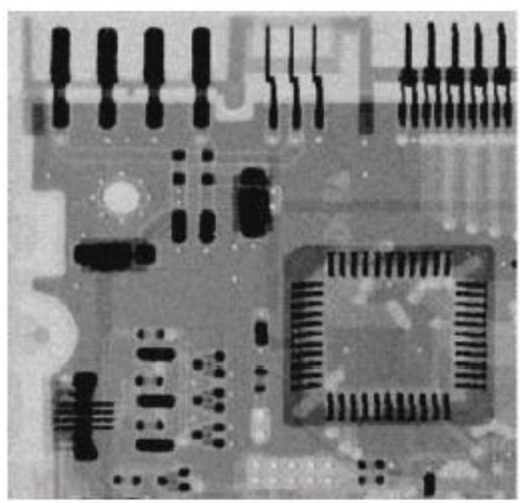
Mean filters: **Geometric Mean Filter**



Input Image



Corrupted by Gaussian Noise



Result of Geometric Mean Filter [3 x3]

Restoration in the Presence of Noise

Mean filters: **Harmonic mean filter**

- Works well for salt noise but fails for pepper noise.

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

- g(s,t) is the corrupted image
- $S_{x,y}$ is the mask [window size – (m x n)]

  - Another variation of the arithmetic mean filter.
  - Useful for images with Gaussian or Salt noise
  - Black pixels (pepper noise) are not filtered

Restoration in the Presence of Noise

Mean filters: **Harmonic mean filter**

Input Image

Image with Gaussian Noise

Result of Harmonic Filter [3 x3]

Restoration: Mean Filters Comparison

Input Image

Gaussian Noise

Arithmetic Mean

Arithmetic Mean[5 x5]

Geometric Filter

Harmonic Filter

Restoration in the Presence of Noise

Mean filters: **Contra-harmonic mean filter**

- Reduce the effects of salt-and-pepper noise

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

- Q is the order of the filter
- g(s,t) is the corrupted image
- $S_{x,y}$ is the mask [window size – (m x n)]

- If Q > 0, Eliminate pepper noise
- If Q < 0, Eliminate salt noise
- If Q = 0, Arithmetic mean filter
- If Q = -1, Harmonic mean filter

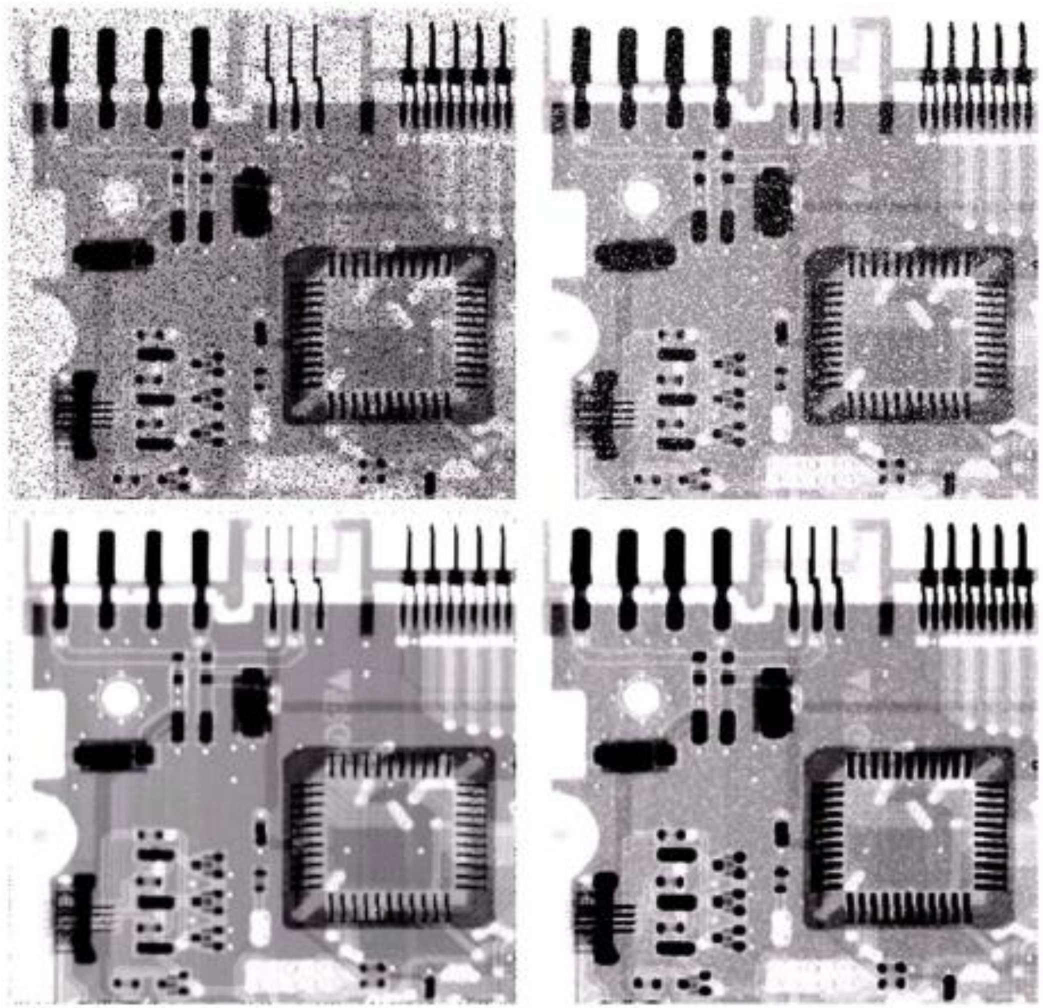
Contra-harmonic filter applications

Mean filters: **Contra-harmonic mean filter**

- **The positive-order filter:** Effectively reduces the pepper noise, at the expense of blurring the dark areas.
- **The negative-order filter:** Effectively reduces the salt noise, at the expenses of blurring the bright areas.
- **Arithmetic and Geometric Filters:** Suit the Gaussian or Uniform noise.
- **Contra-harmonic Filter:** Suit the impulse noise.

Contra-harmonic mean filter - Example

Pepper Noise with probability of 0.1



Salt Noise with probability of 0.1

Contra-harmonic filter [3 x 3]

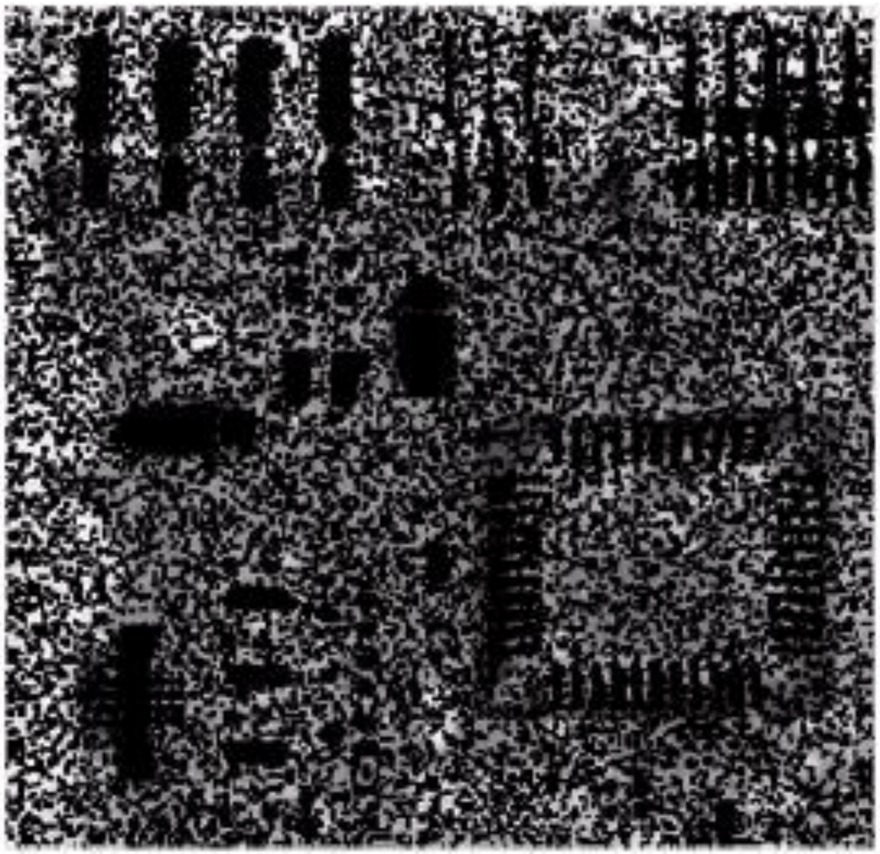
Q = 1.5

Contra-harmonic filter [3 x 3]

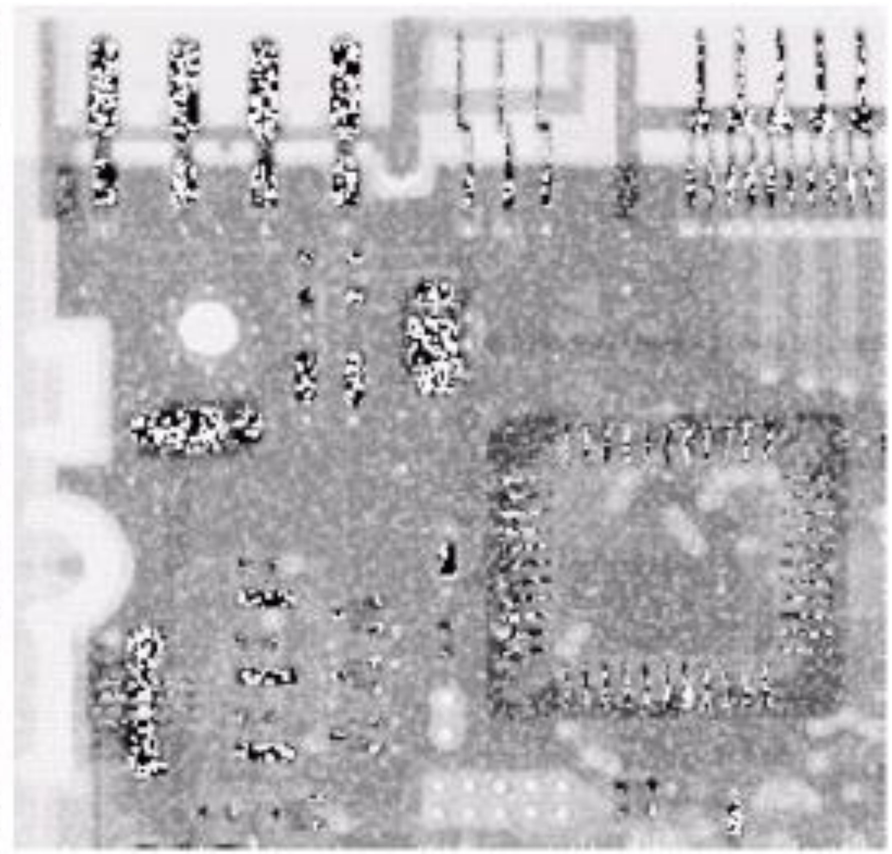
Q = - 1.5

Contra-harmonic Filter – Example

(Results of selecting Wrong Sign)



3x3 Contra-harmonic
Q= - 1.5



3x3 Contra-harmonic
Q= 1.5

Rank / Order / Order Statistics Filters

– Known as Rank filters, Order filters OR Order Statistics filters

– Operate on a neighborhood around a reference pixel by ordering (ranking) the pixel values and then performing an operation on those ordered values to obtain the new value for the reference pixel

– They perform very well in the presence of salt and pepper noise but are more computationally expensive as compared to mean filters

Rank / Order Statistics Filters: Median filter

- Output is based on ordering (ranking) the pixels in a subimage
- Replace the value of a pixel by the median of the gray levels in the neighborhood of that pixel (a specified mask)
- Excellent for removing both bipolar and unipolar impulse noise

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

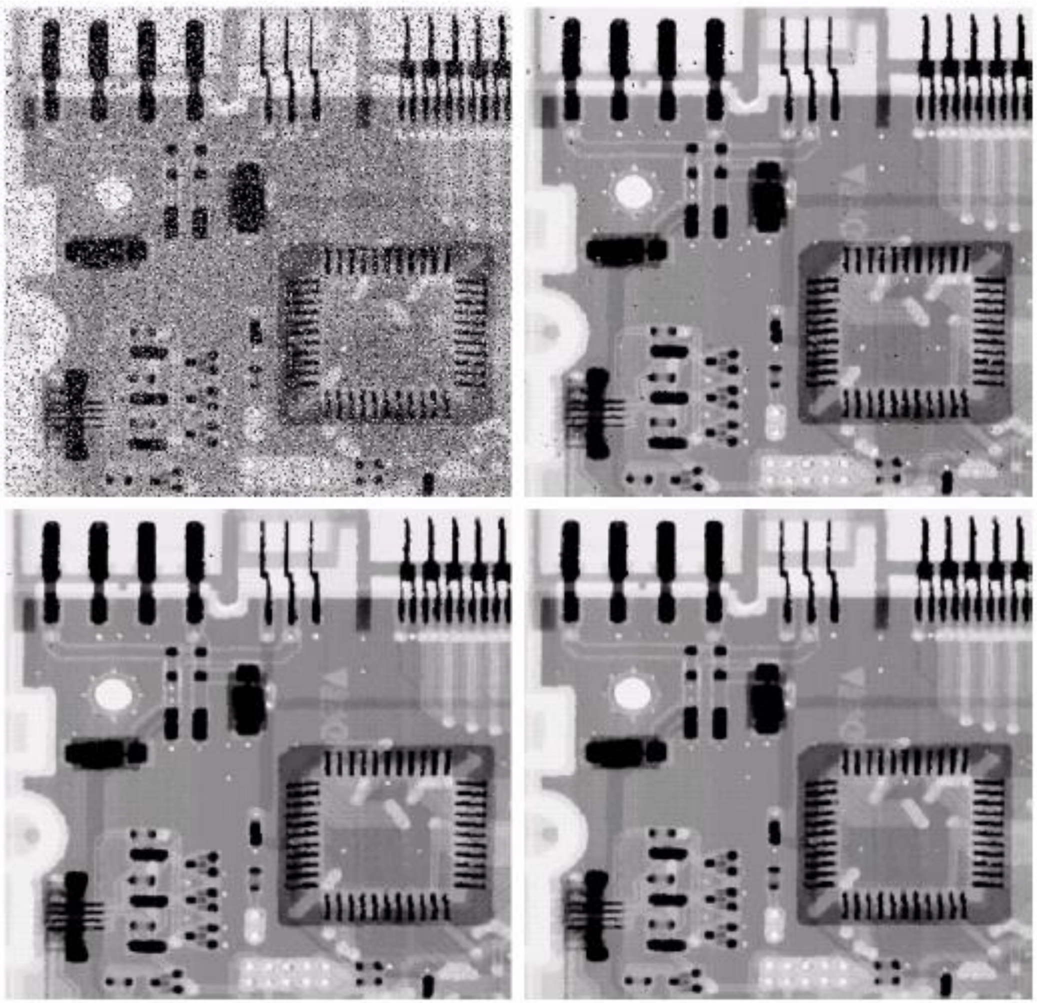
Rank / Order Statistics Filters: Median filter

- *Most popular and useful of the rank filters.*
- *It works by selecting the middle pixel value from the ordered set of values within the $m \times n$ neighborhood (W) around the reference pixel.*
  - If *mn* is an even number, the arithmetic average of the two values closest to the middle of the ordered set is used instead.

Median filter (Example)

Salt & Pepper noise
Pa = Pb = 0.1

One Pass
Media Filter
[3 x 3]



Two Pass
Media Filter
[3 x 3]

Three Pass
Media Filter
[3 x 3]

Rank / Order Statistics Filters: Max and Min filter

- Max filter$\rightarrow$ Replace the value of a pixel by the maximum of the gray levels (the brightest point) in the neighborhood of that pixel
- Min filter$\rightarrow$ Replace the value of a pixel by the minimum of the gray levels (the darkest point) in the neighborhood of that pixel

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

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

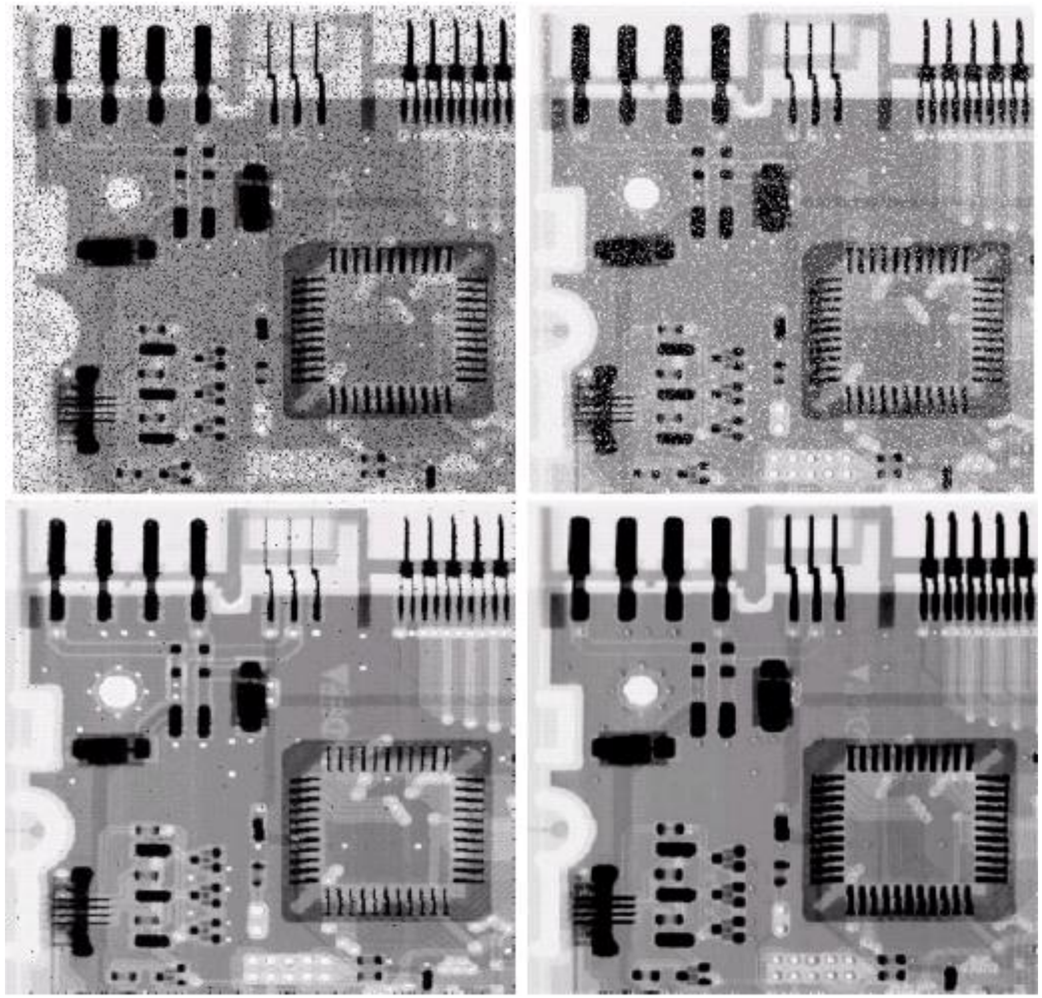
Rank / Order Statistics Filters: Max and Min filter

- *Max filter also known as 100th percentile filter*
- *Min filter also known as zeroth percentile filter*
- *Max filter helps in removing pepper noise*
- *Min filter helps in removing salt noise*

Max and Min filter (Example)

Pepper Noise

Salt Noise



Max Filter

Min Filter

Rank / Order Statistics Filters: Midpoint filter

- Filter's output→ the midpoint between the maximum and minimum values of the gray levels in the mask
- Combine order statistics and averaging
- Midpoint filter works best for randomly distributed noise (Gaussian or uniform)

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

– *Calculates the average of the highest and lowest pixel values within a window*

Alpha-Trimmed Mean Filter

- Alpha-trimmed mean filter takes the mean value of the pixels enclosed by an m×n mask after deleting the pixels with the d/2 lowest and the d/2 highest gray-level values

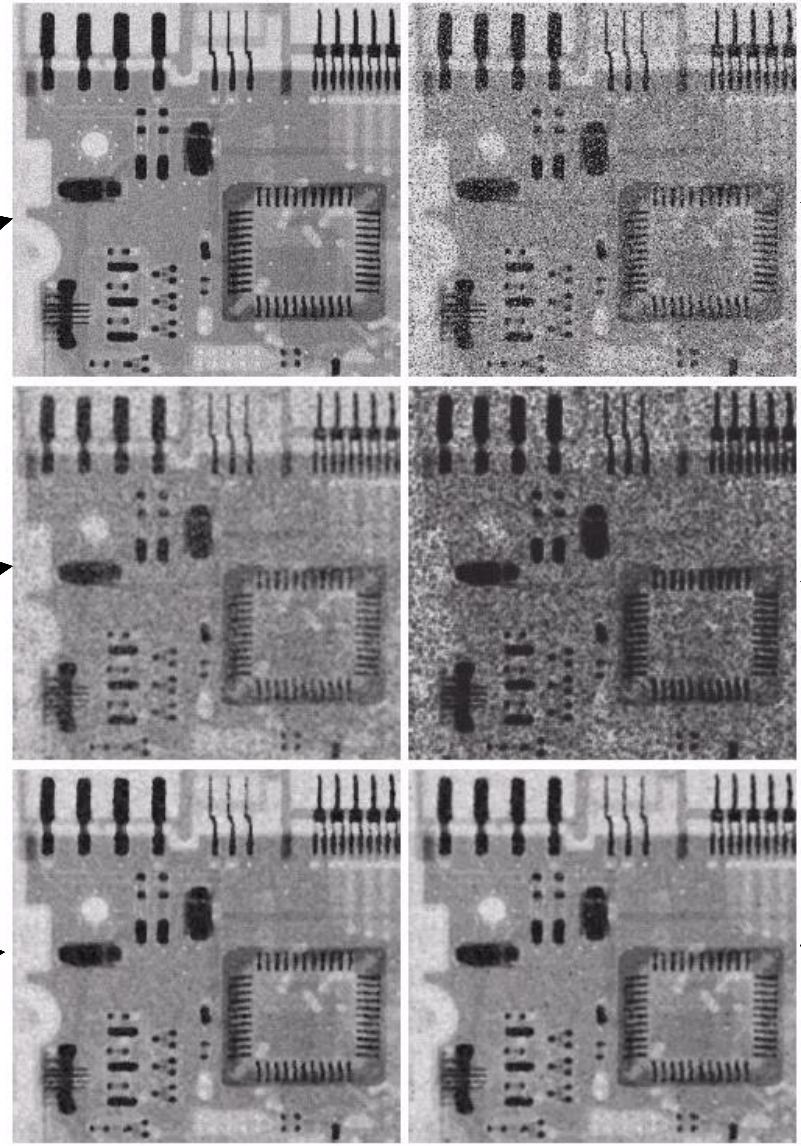
$$\hat{f}(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} g_r(s,t)$$

  - $g_r(s,t)$ represent the remaining mn-d pixels
  - It is useful in situations involving multiple types of noise like a combination of salt-and-pepper and Gaussian

De-Noising

Corrupted by additive Uniform noise

Added salt & pepper noise



5x5 Mean Filtering

5x5 Geo-Mean Filtering

5x5 Median Filtering

5x5 Alpha-trimmed Mean Filtering