



**Lecture No. #5**



# Digital Image Processing

**Prepared by:**

**MSc. Royida A. Ibrahim Alhayali**

**2015 / 2016**

***“Image Restoration & Reconstruction”***

## **Image Restoration**

Image restoration attempts to reconstruct or recover an image that has been degraded by a degradation phenomenon. Thus, restoration techniques are oriented toward modeling the degradation and applying the inverse process in order to recover the original image. As in image enhancement, the ultimate goal of restoration techniques is to improve an image in some predefined sense.

Like enhancement techniques, restoration techniques can be performed in the spatial domain and frequency domain. For example, noise removal is applicable using spatial domain filters whereas deblurring is performed using frequency domain filters because image blur are difficult to approach in the spatial domain using small masks.

# A Model of Image Degredation & Restoration

- ✓ *Image Degredation is a process that operates on an “input” image  $f(x,y)$  during the process of acquisition and/or transmission.*
- ✓ *It is assumed that it is the result of a **degradation function**  $H(u,v)$  plus an additive noise  $N(u,v)$  acting on the frequency domain:*

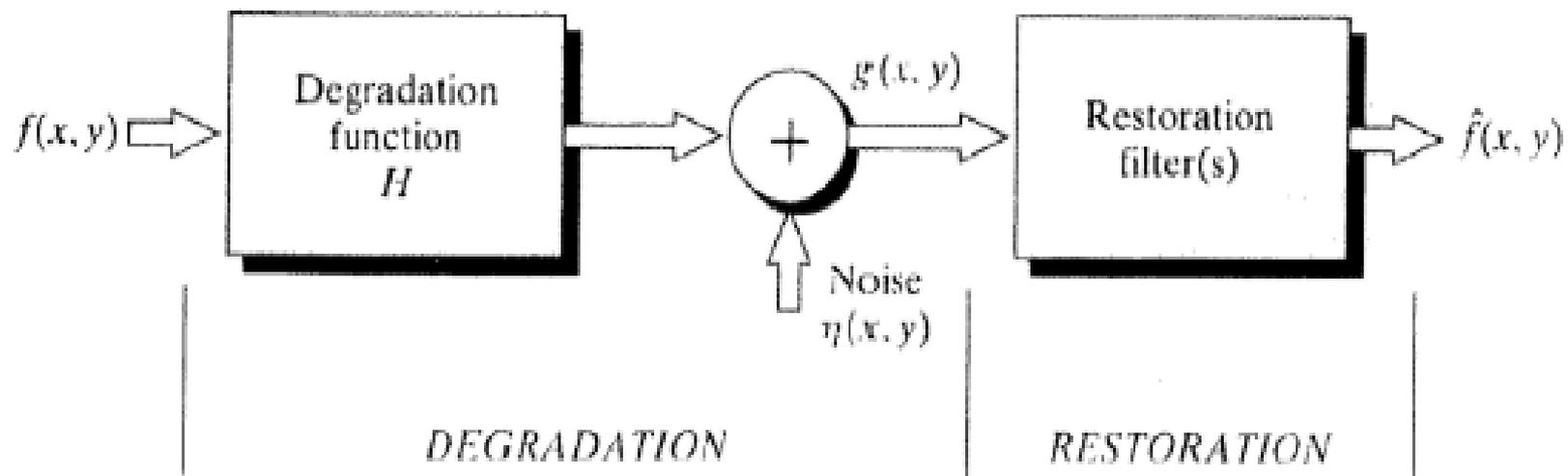
$$G(u,v) = H(u,v)F(u,v) + N(u,v).$$

- ✓ *The **observed** degraded image  $g(x,y)$ , can be modelled as*

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

*where  $h(x,y)$ ,  $\eta(x,y)$ , and  $g(x,y)$  are the inverse DFT of  $H(u,v)$ ,  $N(u,v)$ , and  $G(u,v)$ , respectively.*

- ✓ *The process of restoration aims to recover  $f(x,y)$  from the observed degraded image  $g(x,y)$ . Restoration requires a realistic model of noise to be removed first, and then filtering out  $H(u,v)$ .*



A model of the image degradation/restoration process

# *Noise Models*

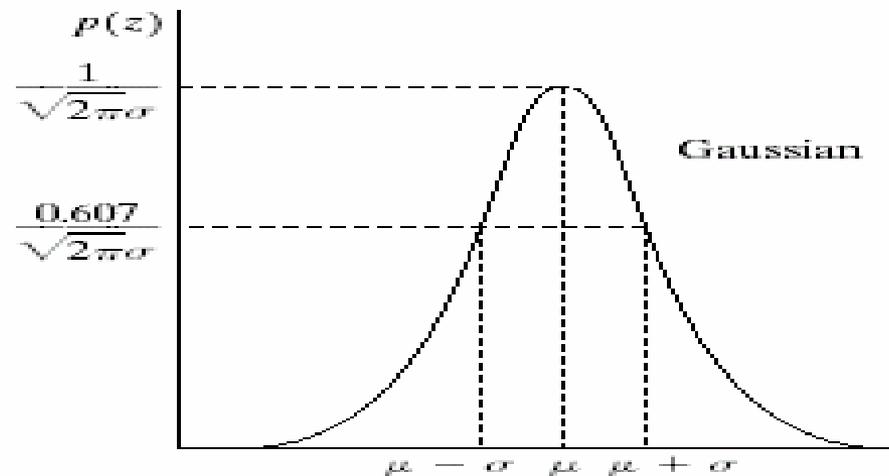
- *Noise in digital images arise during*
  - ❑ *Aquasition: environmental conditions (light level & sensor temperature)*
  - ❑ *and/or transmission – interference in the transmission channel.*
- *To remove noise we need to understand the spatial Characteristics of noise and its frequency characteristics (Fourier spectrum).*
- *Generally, noise level is assumed to be independent of position in image and uncorrelated to pixel values, i.e. we cannot say that noise effects some positions or pixels values more or less than other even if this is so visibly.*
- *White noise refers to noise function with constant Fourier spectrum*
- *Spatial noise is described by the statistical behavior of the gray-level values in the noise component of the degraded image.*

# *Statistical Models of Spatial Noise*

- ❖ *Spatial noise can be modelled in terms of the statistical behavior of the gray-level values in the noise component of the degraded image, i.e. a random variable with a specific probability distribution function (PDF).*
- ❖ *Important examples of noise models include:*
  - ❖ *Gaussian Noise.*
  - ❖ *Rayleigh Noise.*
  - ❖ *Gamma Noise.*
  - ❖ *Exponential Noise.*
  - ❖ *Uniform Noise.*
  - ❖ *Impulse Noise (Salt & Pepper).*

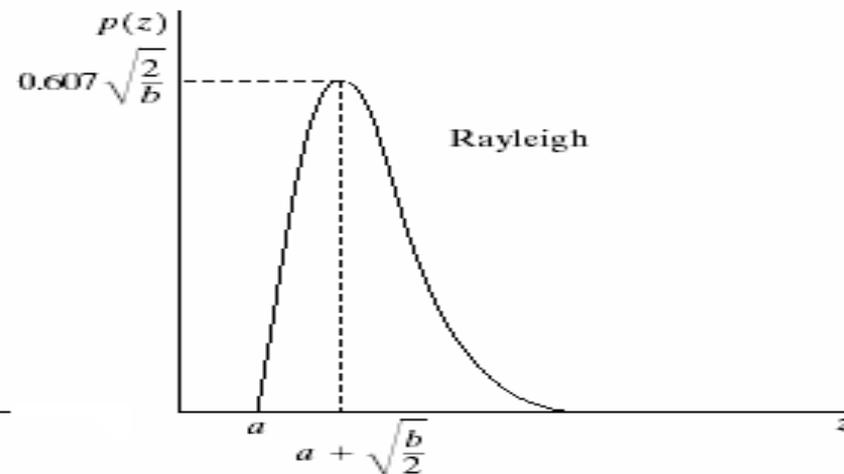
Gaussian : 
$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2},$$

where  $z$  is the gray - value,  $\mu$  is the mean and  $\sigma$  is the standard deviation.



Rayleigh :

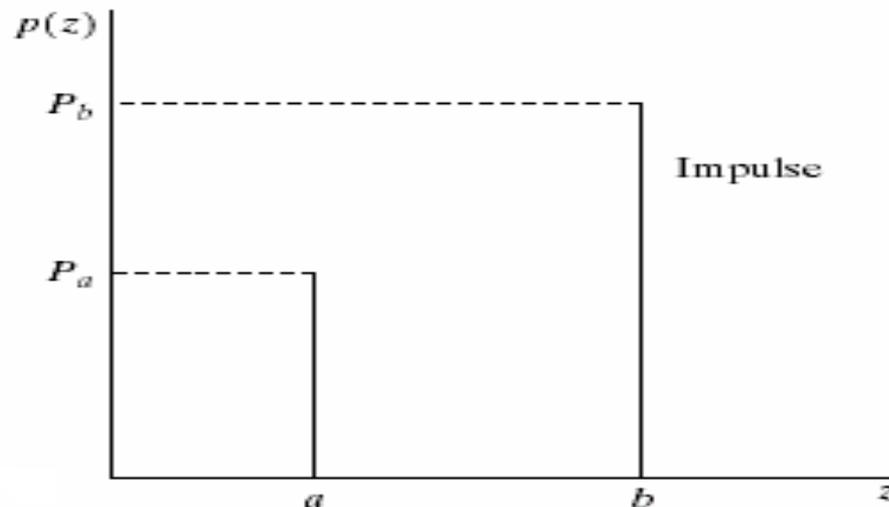
$$p(z) = \begin{cases} (2/b)(z-a)e^{-(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a. \end{cases}$$



Impulse (Salt and Pepper) :

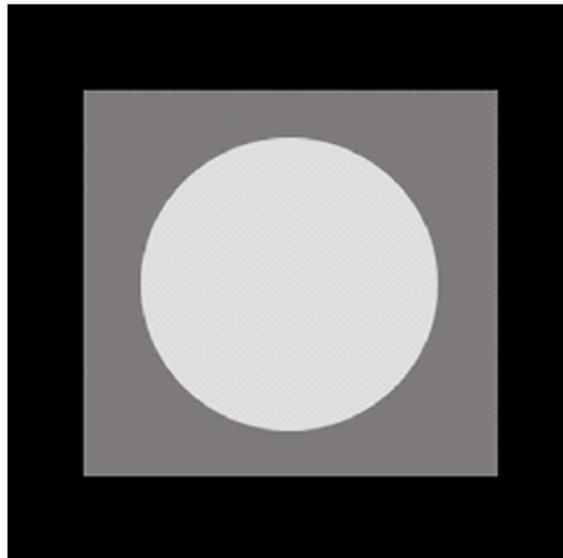
$$p(z) = \begin{cases} P_a & \text{if } z = a \\ P_b & \text{if } z = b \\ 0 & \text{otherwise} \end{cases} .$$

If  $b > a$ , then gray - level  $b$  appears as a light dot (salt),  
otherwise gray - level  $a$  appears as a dark dot (Pepper).

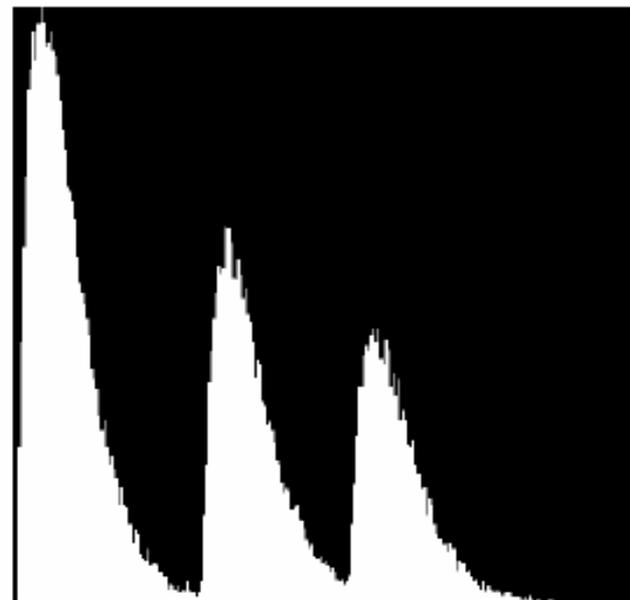
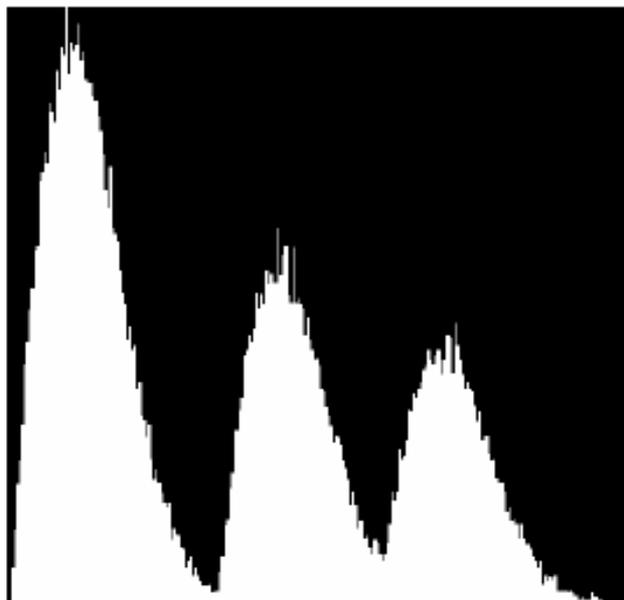
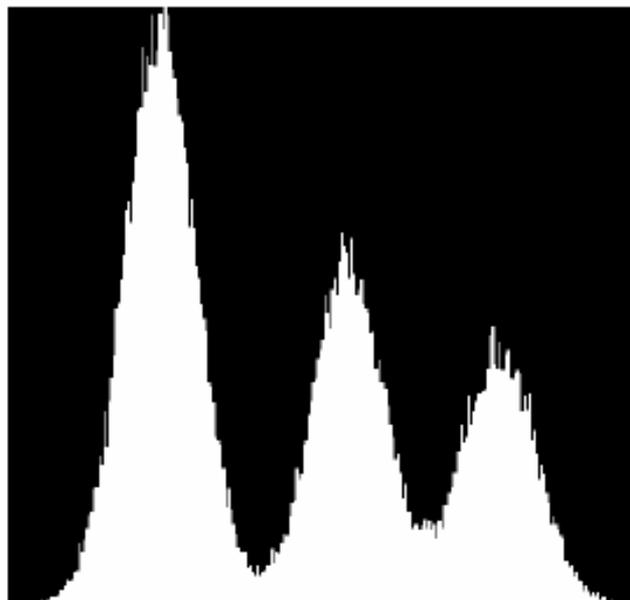
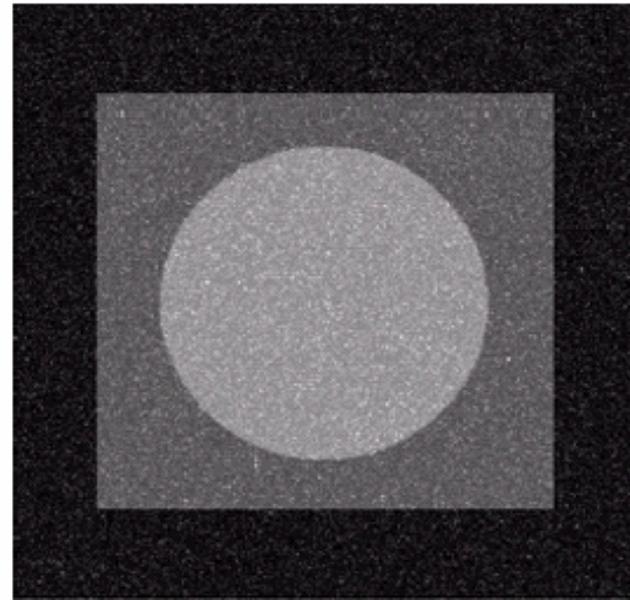
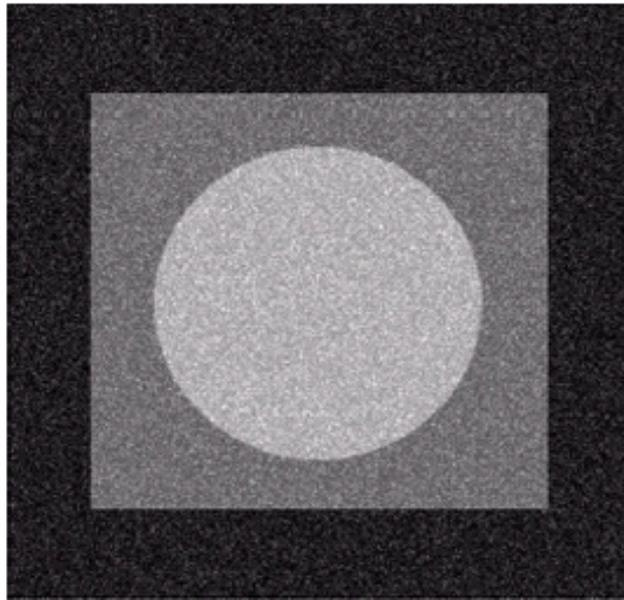
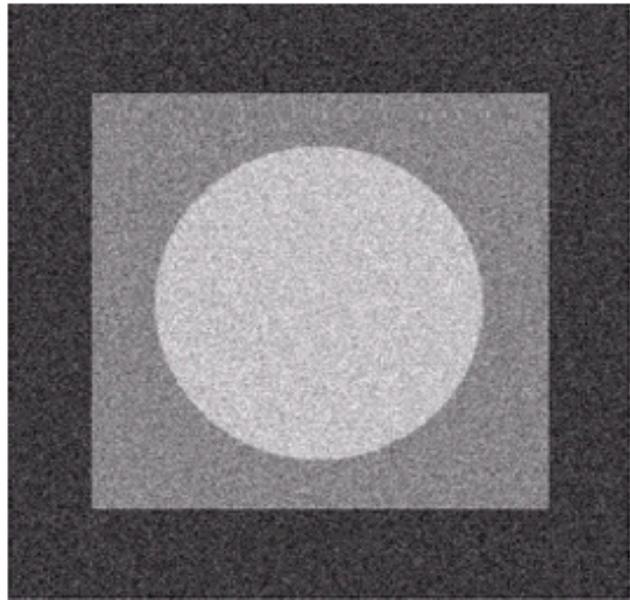


# *Determining noise models*

- ✓ *To determine the noise model in a noisy image, one may select a relatively small rectangular sub-image of relatively smooth region. The histogram of the sub-image approximates the density probability distribution of the corrupting model of noise.*
- ✓ *The simple image below is well-suited test pattern for illustrating the effect of adding noise of the various models.*



# *Addition of noise - Examples*

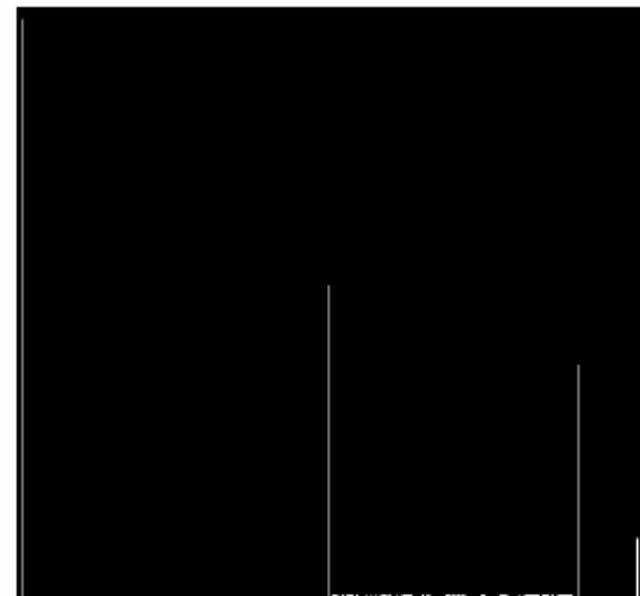
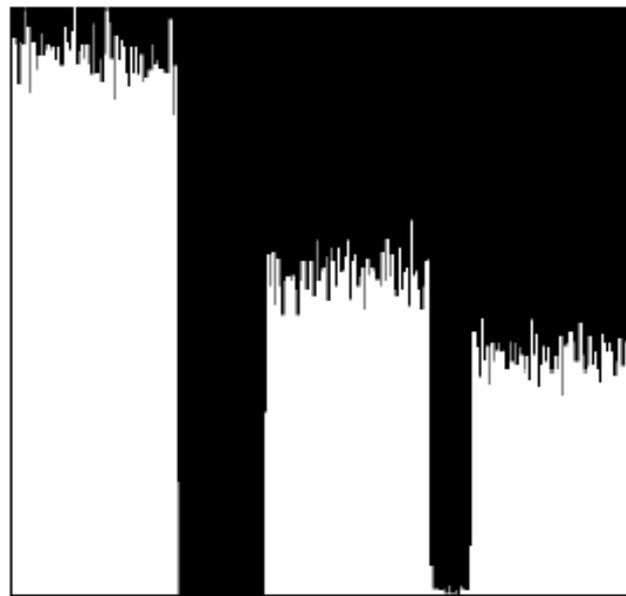
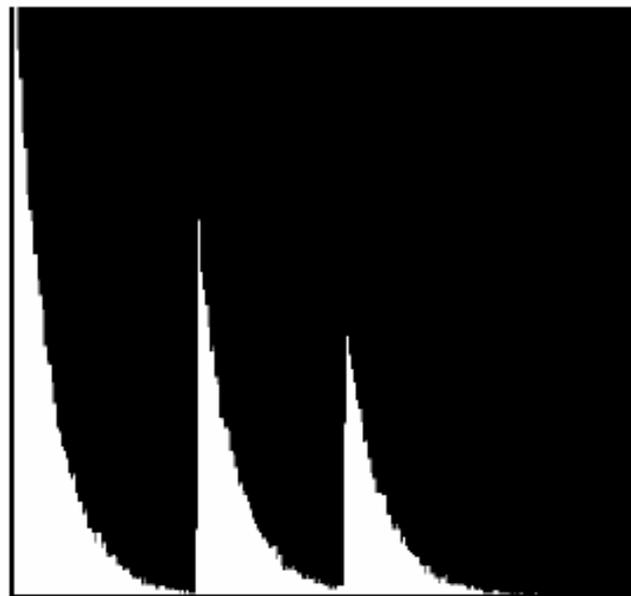
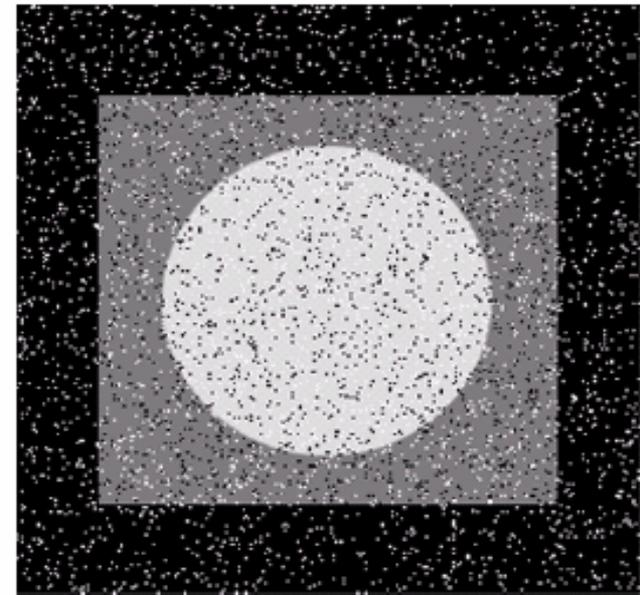
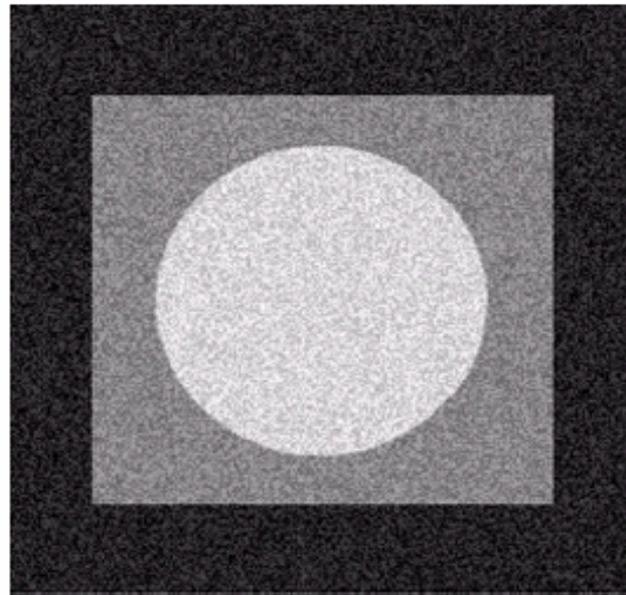
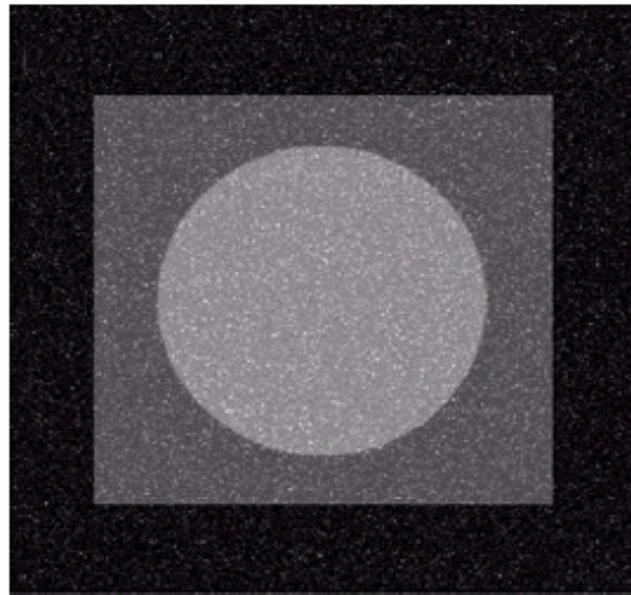


Gaussian

Rayleigh

Gamma

# *Addition of noise - Examples*



Exponential

Uniform

Salt & Pepper

# *Adding Noise with MATLAB*

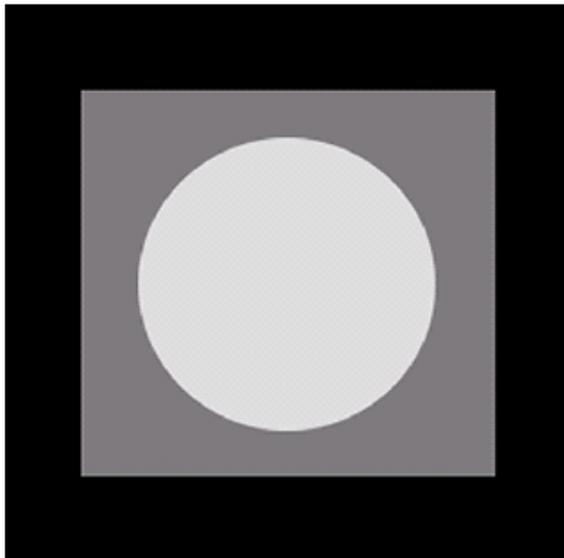
- ❖ *MATLAB provides special instructions to add noise to an image*

*imnoise (f, type, parameters)*

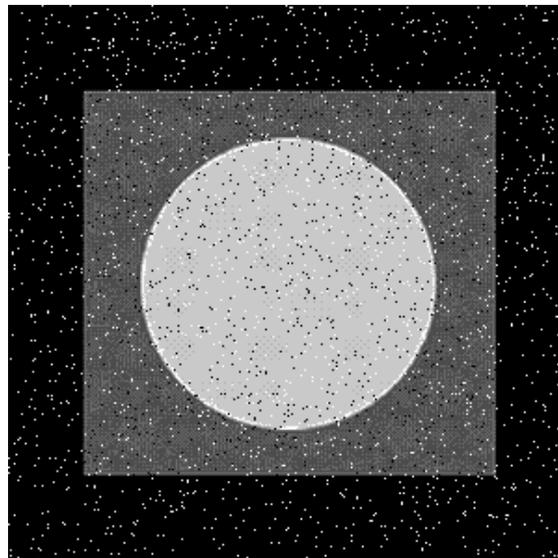
*e.g. imnoise (f, 'gaussian', m, var), and*

*imnoise (f, 'salt & pepper', d)*

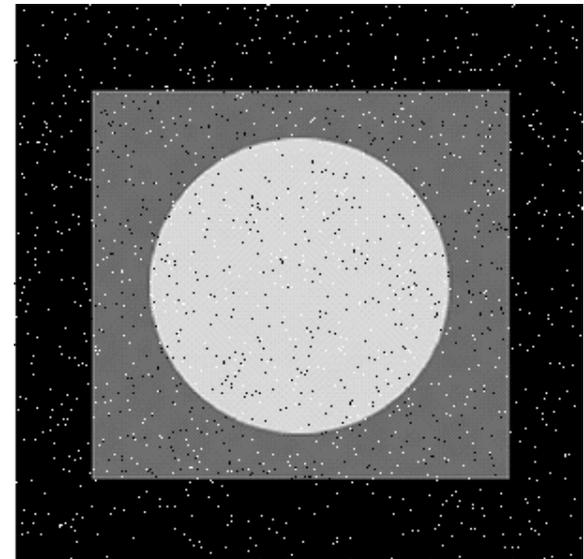
- ❖ *The left-most image below has been corrupted with “Salt & Pepper” noise with different densities.*



**Original Clean image**



**Salt&pepper, d=0.05**



**Salt&pepper, d=0.025**

# *Image restoration in the presesncce of Noise Only*

❖ *Here we assume that  $H(u,v)=1$ , i.e. the degradation is modelled as:*

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

❖ *Spatial filters have been designed to remove noise include:*

*1. Mean filters – Arithmetic, Geometric, Harmonic, & Contraharmonic*

*2. Order Statistics Filter – Median, Max, min, Midpoint*

*And*

*3. Adaptive Filters.*

❖ *These filters are applied in the same way as before using a square array neighbourhood of size  $3 \times 3$ ,  $5 \times 5$ , ...etc.*

❖ *The resulting image approximates the perceived clean image  $f(x,y)$ .*

# Types of Mean Filters

1. Arithmetic Mean :

$$\hat{f}(x, y) = \frac{1}{mn} \sum g(x, y).$$

The sum is taken over all pixels in the mask.

2. Geometric Mean :

$$\hat{f}(x, y) = \left[ \prod g(s, t) \right]^{1/mn}$$

The product is taken over all pixels in the mask.

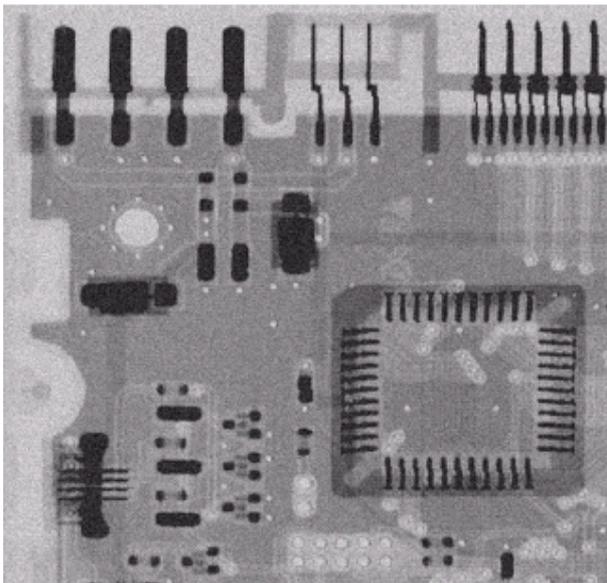


Image corrupted by adding Gaussian noise

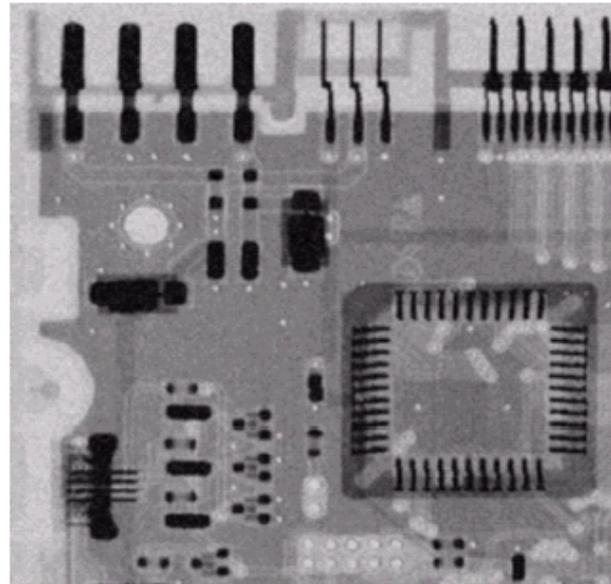


Image restored with arithmetic mean filter

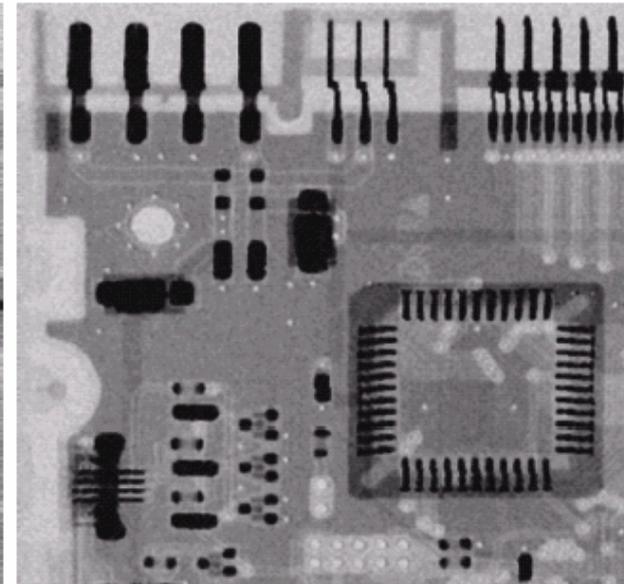


Image restored with Geometric mean filter

## *Types of Mean Filters - continued*

### 3. Harmonic Mean :

$$\hat{f}(x, y) = \frac{mn}{\sum \frac{1}{g(x, y)}}.$$

The sum is taken over all pixels in the mask.

Works well for salt noise removal

### 4. Contraharmonic Mean :

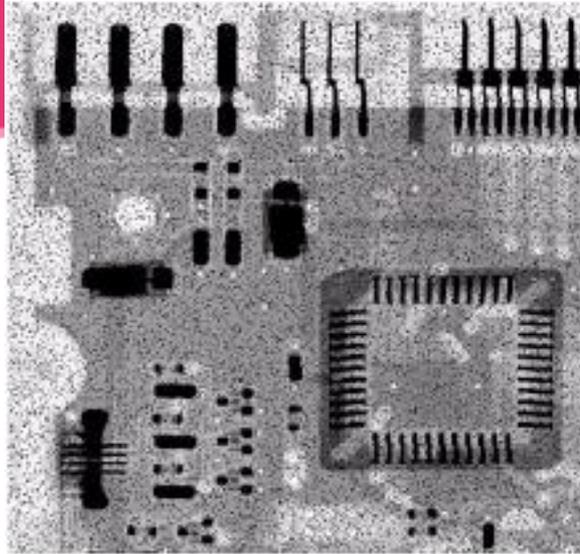
$$\hat{f}(x, y) = \frac{\sum g(x, y)^{Q+1}}{\sum g(x, y)^Q}.$$

Q is the order. Positive Q eliminates pepper, while negative Q eliminate salt.

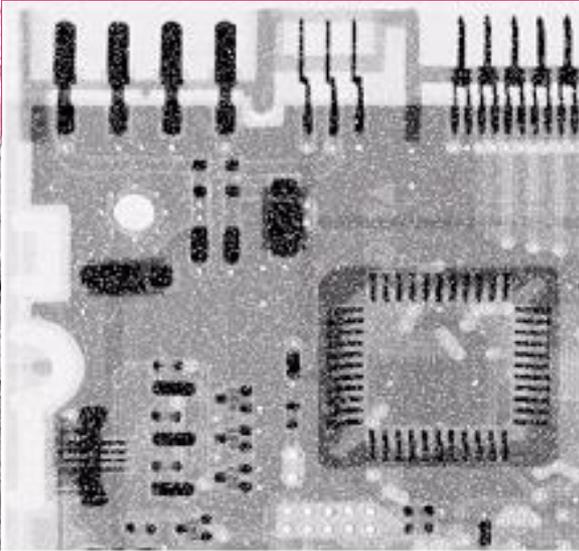
**Note that the harmonic filter is a special case of contraharmonic with Q=-1, and contraharmonic with Q=0 is the arithmetic mean filter.**

# *Effect of Mean Filters*

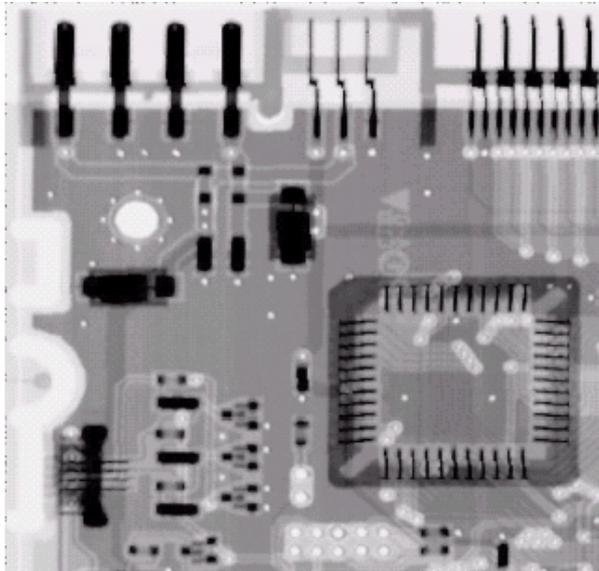
**Image (a)**  
corrupted by  
pepper with  
probability 0.1



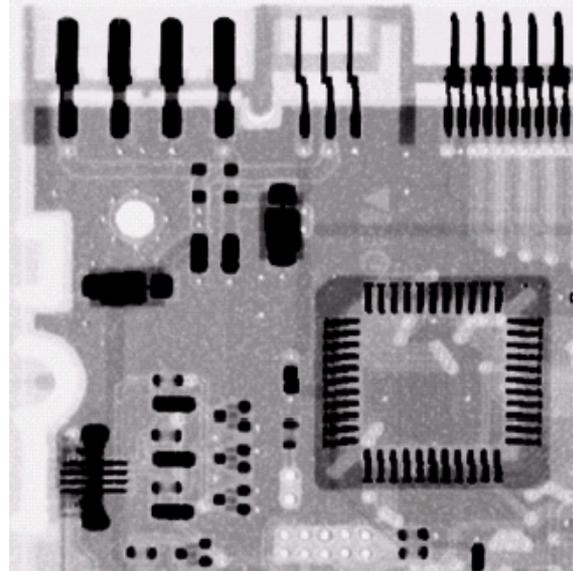
**Image (b)**  
corrupted by  
Salt with  
probability 0.1



**Image (a)**  
restored  
with  
Contrahar  
monic  
filter,  $Q=1.5$

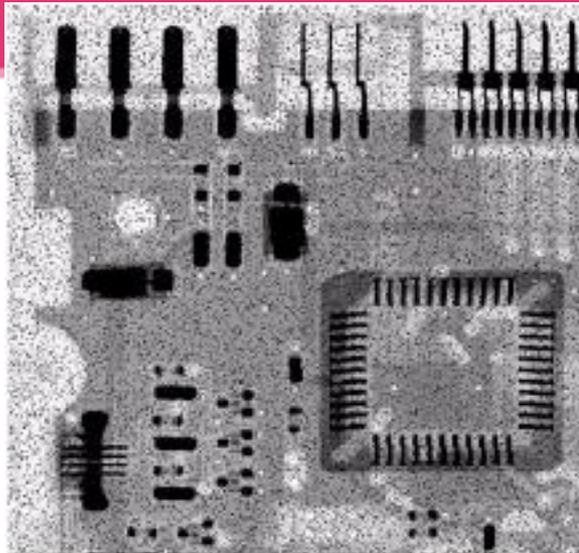


**Image (b)**  
restored with  
Contrahar  
monic filter,  
 $Q=-1.5$

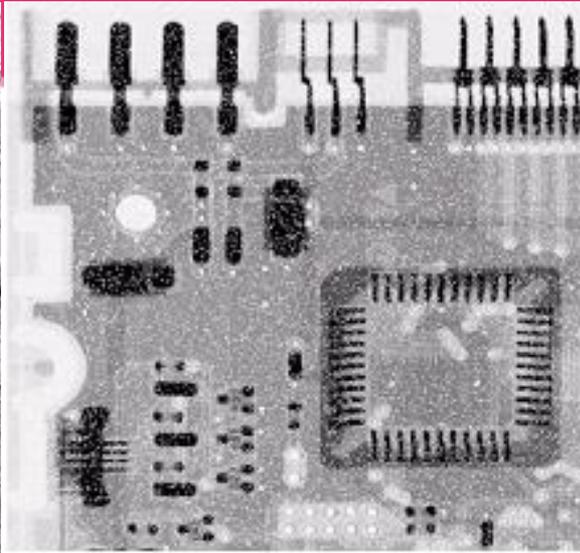


# *Effect of Contraharmonic Filters with wrong sign*

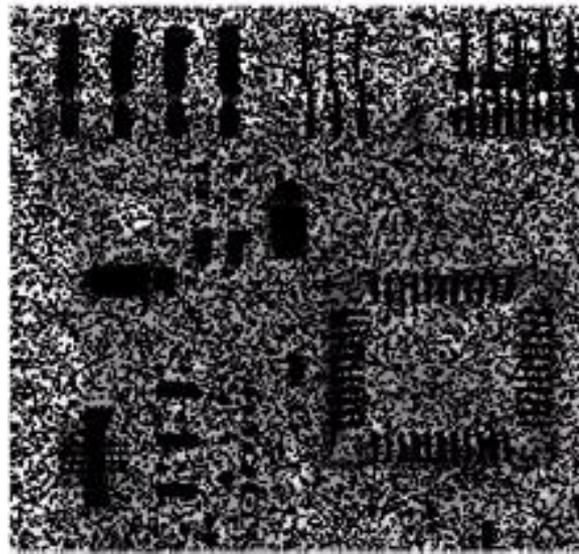
**Image (a)  
corrupted  
by pepper  
with  
probability  
0.1**



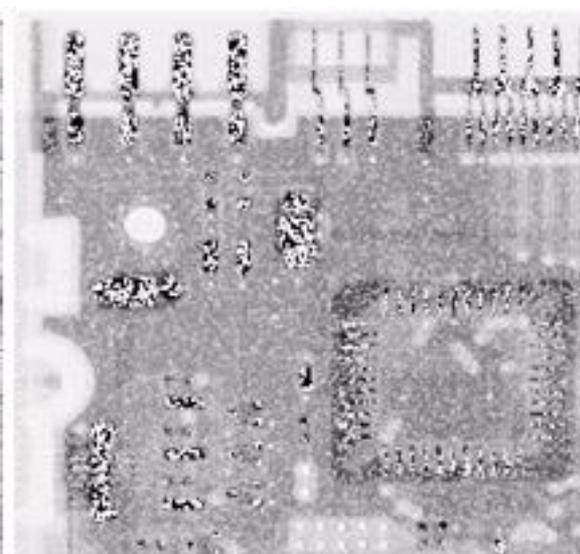
**Image (b)  
corrupted  
by Salt  
with  
probability  
0.1**



**Image (a)  
with  
application  
of  
Contraharm  
onic filter,  
Q=-1.5**



**Image (b)  
with  
application  
of  
Contraharm  
onic filter,  
Q=1.5**



# Salt noise and Pepper noise Removal

Image corrupted by pepper noise with probability 0.1

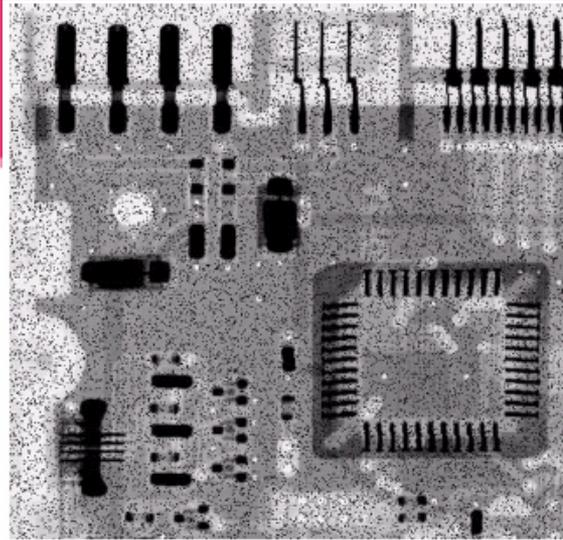
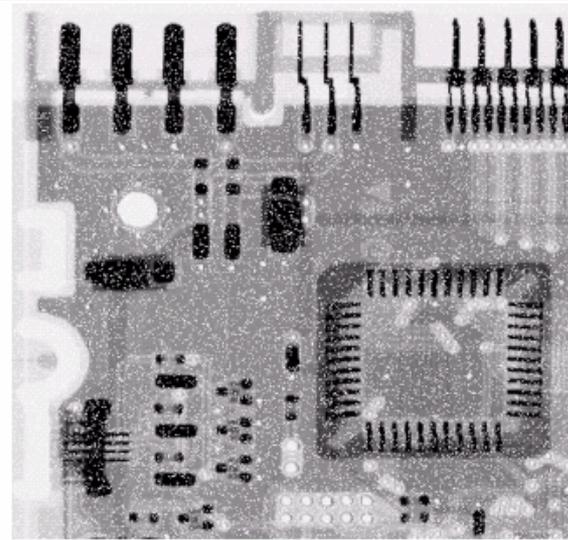
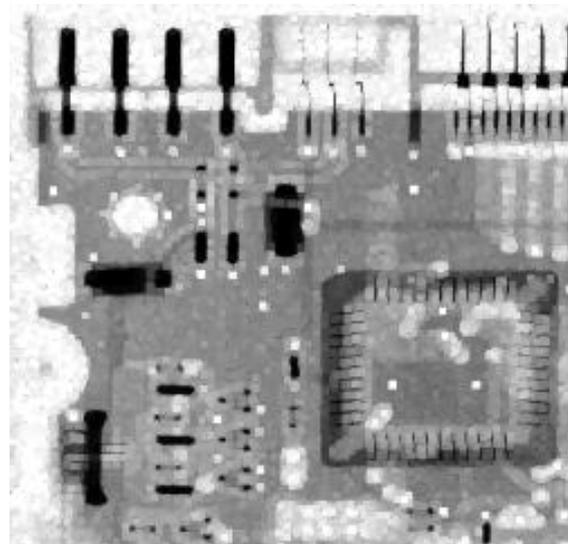


Image corrupted by salt noise with probability 0.1

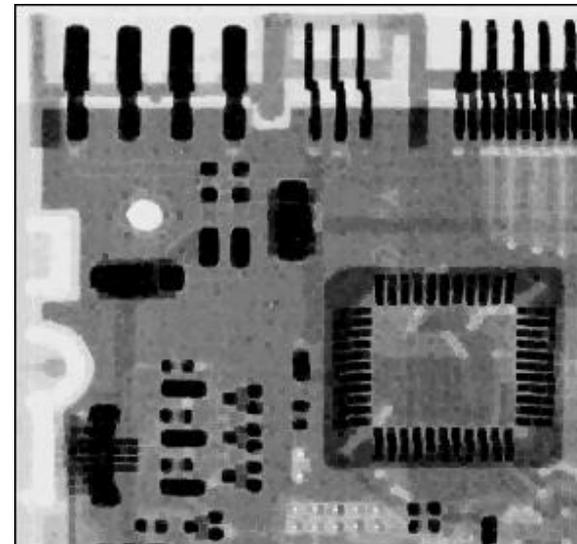


Min and Max order statistics filters may be useful in removing salt noise and pepper noise, respectively, but the result depends on the noise density.

Removing Pepper noise with Max filter



Removing Salt noise with Min filter



# *Spatial Noise removing filters in MATLAB*

- ✓ *The mean as well as order statistics spatial filters can be applied using the following MATLAB instruction:*

***F=spfilt(c, Type, m, n, parameters),***

*Where Type refer to one of the filters defined earlier, and m and n represent the filter size.*

## ***Examples***

***F=spfilt(c, 'amean', m, n) - for arithmetic's mean***

***F=spfilt(c, 'gmean', m, n) - for geometric's mean***

***F=spfilt(c, 'hmean', m, n) - for Harmonic mean***

***F=spfilt(c, 'Chmean', m, n, Q) - for contraharmonic mean***

***F=spfilt(c, 'max', m, n) - for max mxn mean filter***

***F=spfilt(c, 'min', m, n) - for the min mxn filter***

***F=spfilt(c, 'midpoint', m, n) - for the midpoint mxn filter***

***F=spfilt(c, 'median', m, n) - for the median mxn filter***

# Adaptive filters

- ✓ The previous filters are applied regardless of local image variation.
- ✓ Adapted filters change their behaviour using local statistical parameters in the mask region. Consequently, adaptive filters outperform the non-adaptive ones.

- ✓ The following formula defines a simple adaptive filter:

$$f(x,y) = c(x,y) - \sigma_i^2 \times (c(x,y) - m_L) / \sigma_L^2,$$

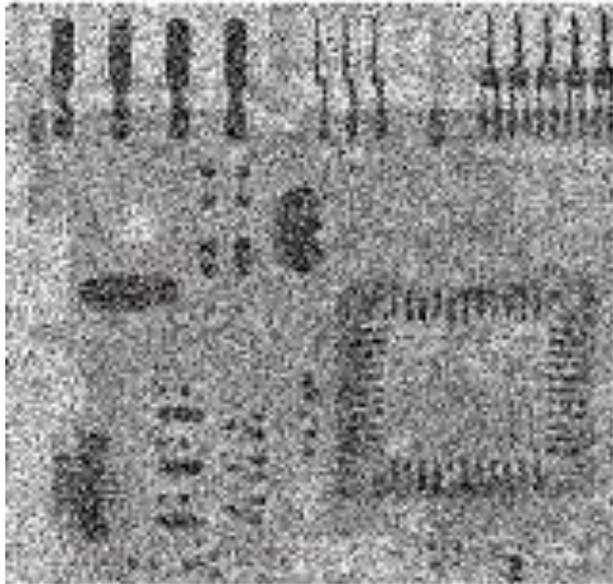
where  $m_L$  and  $\sigma_L$  stand for the mean and standard deviation of pixel values in the mask region, and  $\sigma_i$  is the image standard deviation.

- ✓ An adaptive median filter can be defined which aims to replace  $f(x,y)$  with the median of a neighbourhood up to a specified size as long as the **median** is different from the **max** and **min** values but  $f(x,y) = \min$  or  $f(x,y) = \max$ . Otherwise,  $f(x,y)$  is not changed. This filter is implemented using the MATLAB instruction:

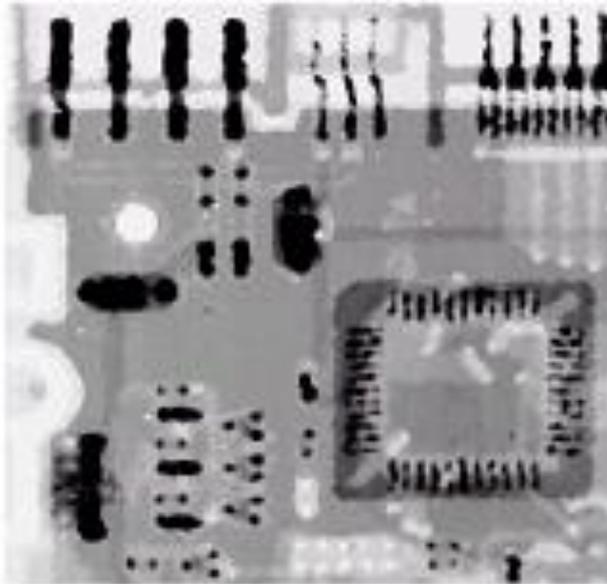
***adpmedian(f, Smax)***

where  $Smax$  is the maximum allowed filter size.

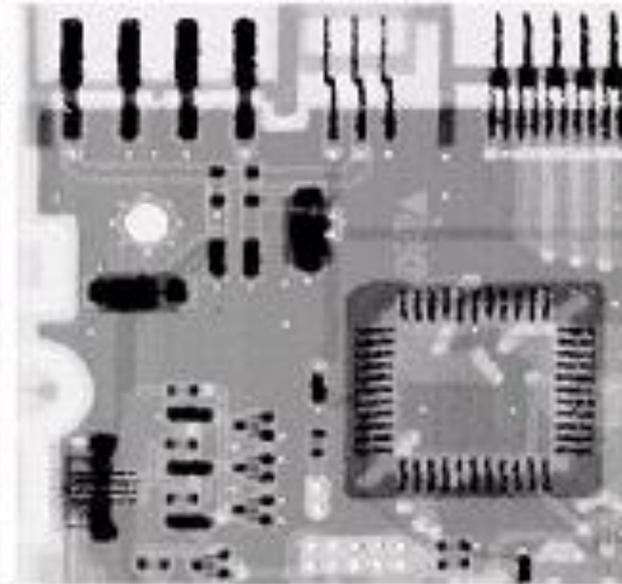
# Example



**Image heavily Corrupted with Salt & Pepper**



**Result of filtering with 7x7 median filter.**



**Result of filtering with adaptive 7x7 median filter.**

- *There are different noise models.*
- *For good strategy in removing noise and restoring image quality one needs to determine noise distribution. (Check the histogram in a reasonable size smooth region with visibly small variation in values)*
- *Once the noise model is estimated use an appropriate filter. Histogram in the same region indicates level of success.*
- *Denoising (i.e. removing noise) often introduce other side effects.*
- *Advanced de-noising filters are based on adaptive strategy, i.e. the procedure tries to adapt the application of the filter as it progresses (see the main reference book on these types of filters).*
- *Frequency domain filters provide powerful de-noising methods.*
- *Noise in Colour images may have different characteristics in different colour channels, but removing noise uses the same strategy.*