

# Image Restoration

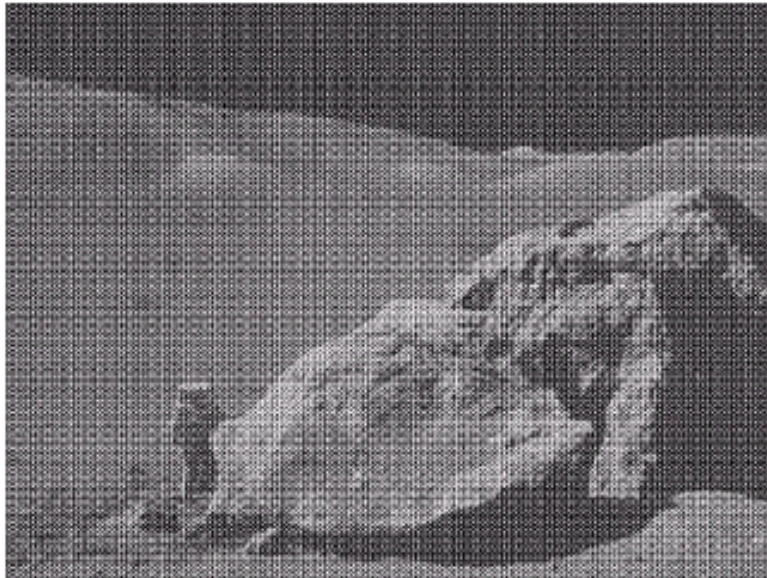
Lecturer: LeNga



# NỘI DUNG

- **Phục hồi ảnh**
- **Nhiều & Mô hình nhiều**
- **Lọc nhiễu**
  - **Dùng lọc không gian**
  - **Dùng lọc tần số**
- **Lọc tương thích**

- Nhằm khôi phục ảnh bị biến đổi
  - Xác định quá trình biến đổi và cố gắng làm quá trình ngược lại
  - Tương tự quá trình cải thiện ảnh nhưng nhiều mục đích



## Nhiều

- Nhiều xuất hiện trong quá trình thu nhận ảnh, số hóa và truyền
  - Cảm biến ảnh có thể bị ảnh hưởng bởi các điều kiện môi trường
  - Nhiều có thể can thiệp vào ảnh trong quá trình truyền
- Ảnh nhiễu được mô phỏng như sau:

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

$f(x, y)$  là ảnh ban đầu

$\eta(x, y)$  là nhiễu

$g(x, y)$  ảnh sau khi bị nhiễu tác động

⇒ Nếu xác định mô hình nhiễu, ta có thể tách nhiễu để phục hồi ảnh





## Common Types of Noise

**Gaussian noise:** variations in intensity drawn from a Gaussian normal distribution



Original



Gaussian noise

## Common Types of Noise

**Impulse noise:** random occurrences of white pixels



Original



Impulse noise

## Common Types of Noise

**Salt and pepper noise:** random occurrences of black and white pixels



Original

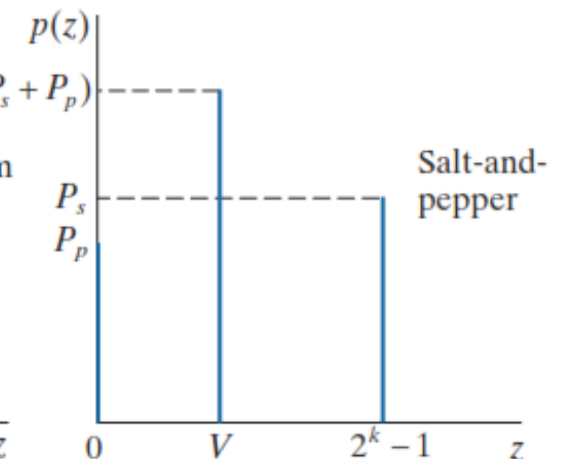
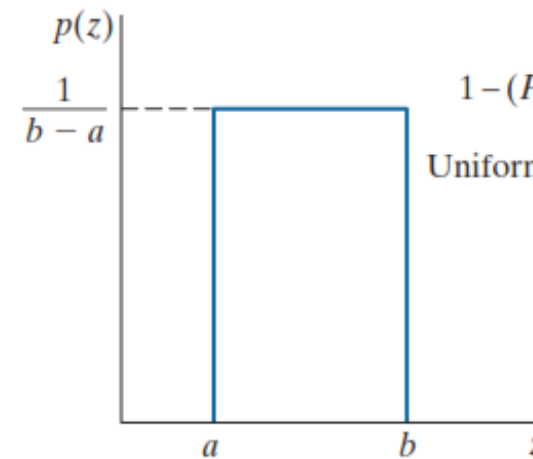
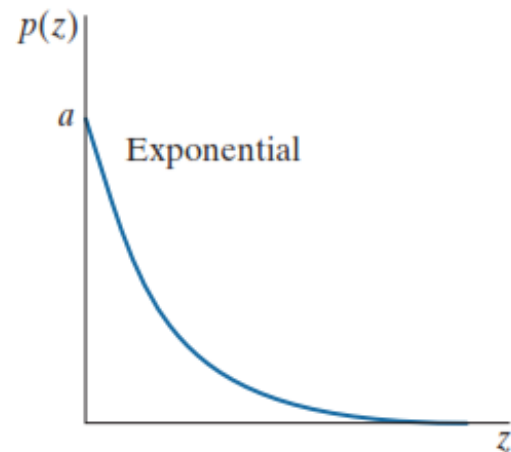
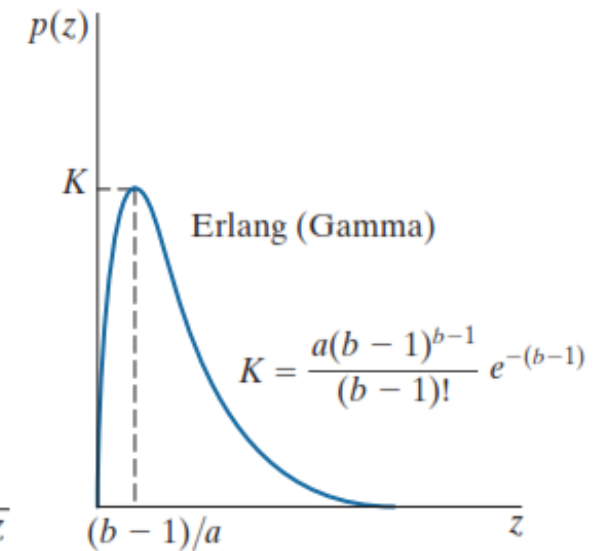
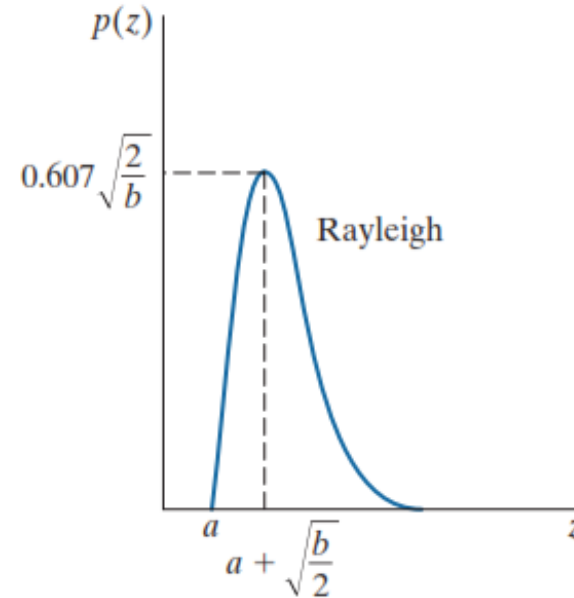
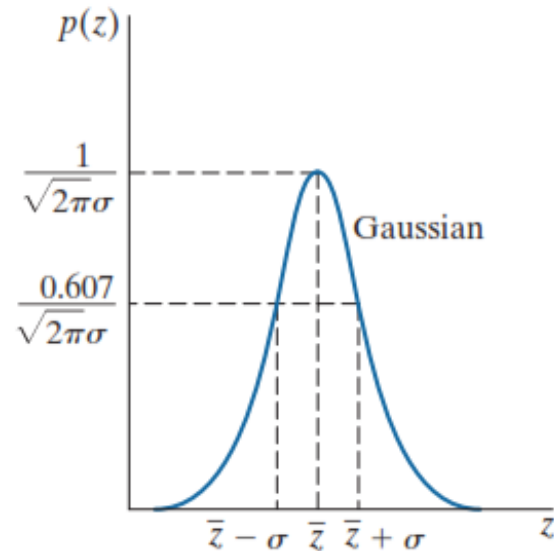


Salt and pepper noise

## Mô hình nhiều

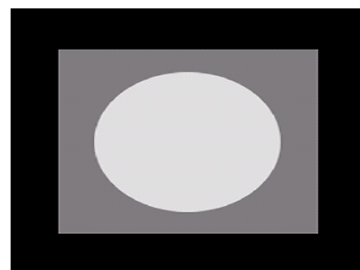
nhiều mô hình nhiều  $\eta(x, y)$  khác nhau:

- Gaussian
  - thông dụng
- Rayleigh
- Erlang
- Hàm mũ
- Đồng nhất
- Xung
  - Nhiều muối, nhiều hạt tiêu

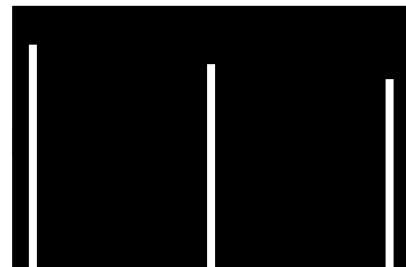




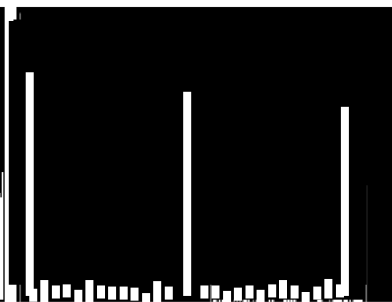
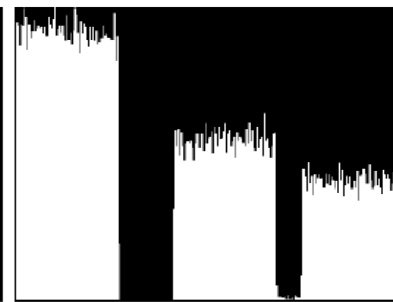
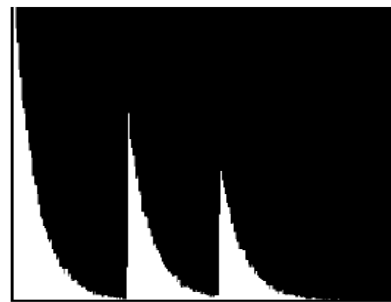
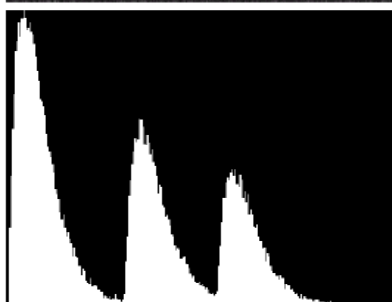
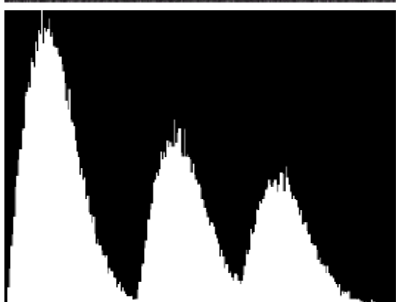
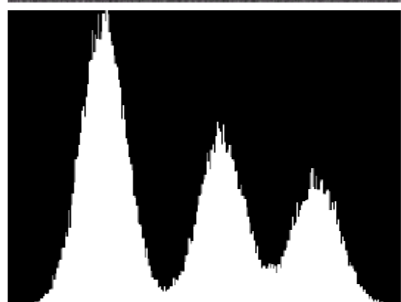
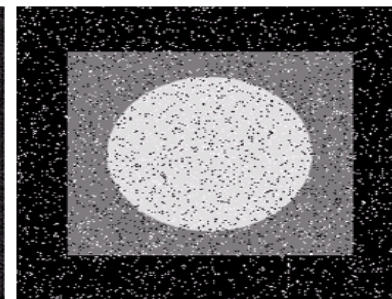
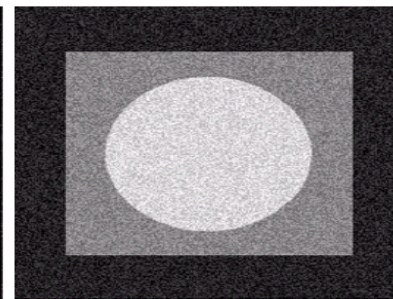
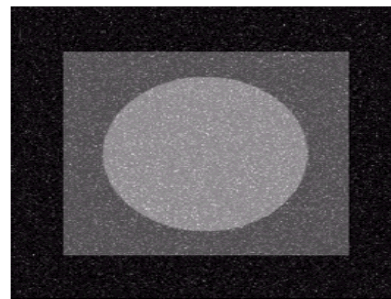
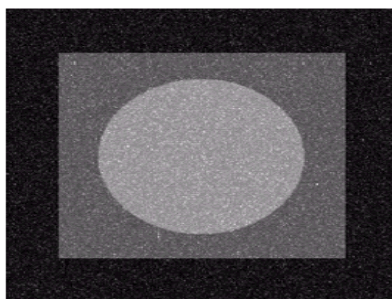
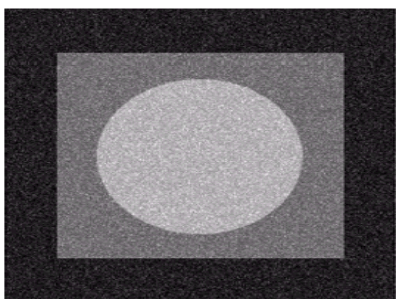
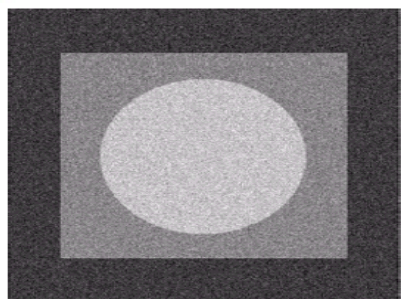
Ví dụ: biểu diễn nhiễu cộng



Image



Histogram



Gaussian

Rayleigh

Erlang

Exponential

Uniform

Impulse

cộng nhiễu với các mô hình khác nhau

- Dùng lọc không gian đơn giản để loại nhiễu:
  - Trung bình số học
  - Trung bình hình học
  - Trung bình Harmonic
  - Trung bình Contraharmonic

- **Trung bình số học**

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

⇒ Lọc làm mịn đơn giản, làm mờ ảnh để loại nhiễu

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

- Trung bình hình học

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

⇒ Làm mịn như trung bình số học nhưng ít làm mất chi tiết ảnh

- Trung bình Harmonic

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

⇒ Làm việc tốt với nhiễu muối nhưng không tốt với nhiễu hạt tiêu

- Trung bình Contraharmonic

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

$Q$  là bậc của bộ lọc; thay đổi  $Q$  sẽ làm thay đổi hành vi của bộ lọc

$Q > 0 \Rightarrow$  loại nhiễu hạt tiêu

$Q < 0 \Rightarrow$  loại nhiễu muối



# Lọc nhiễu - dùng lọc không gian

Original  
Image

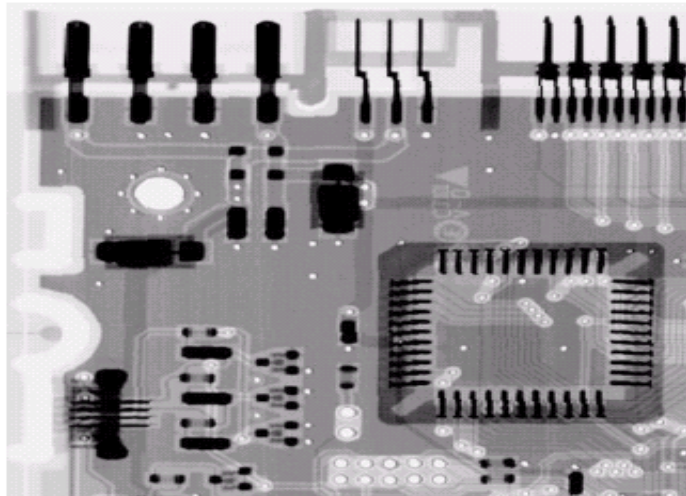
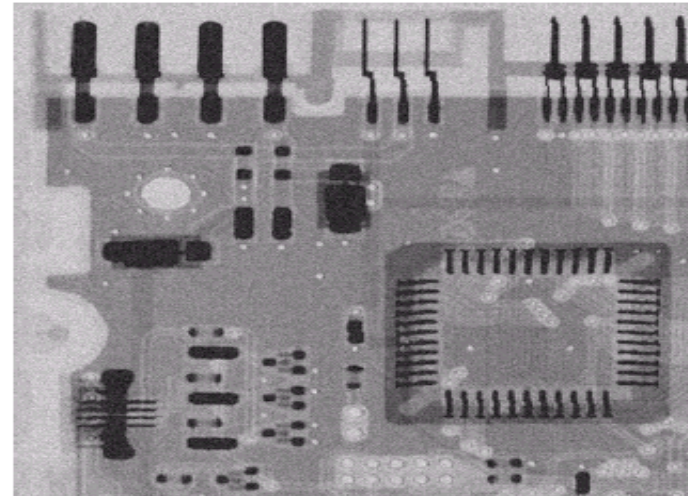
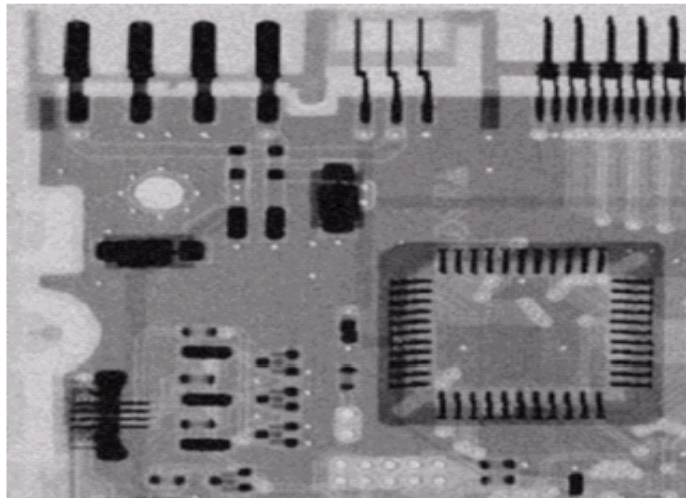


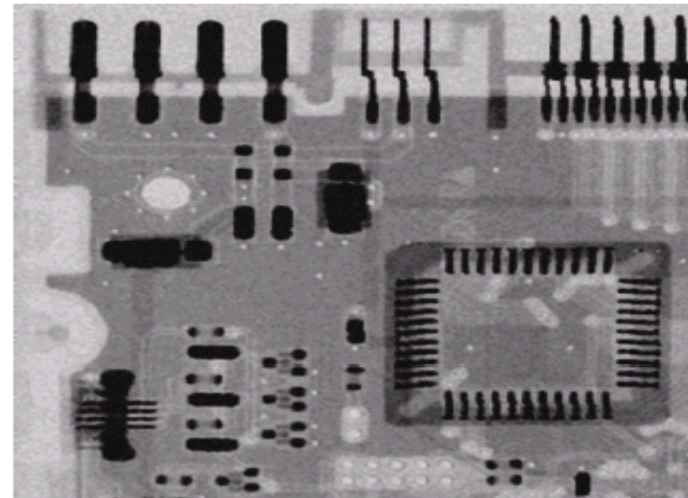
Image  
Corrupted  
By Gaussian  
Noise



After A 3\*3  
Arithmetic  
Mean Filter



After A 3\*3  
Geometric  
Mean Filter





# Lọc nhiễu - dùng lọc không gian

Image  
Corrupted  
By Pepper  
Noise

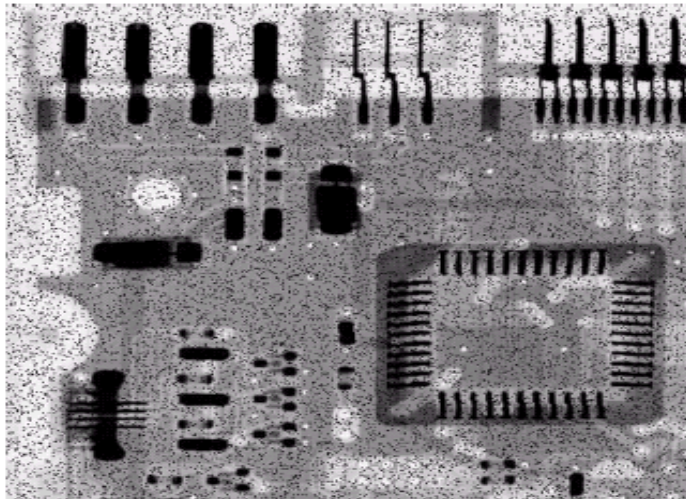
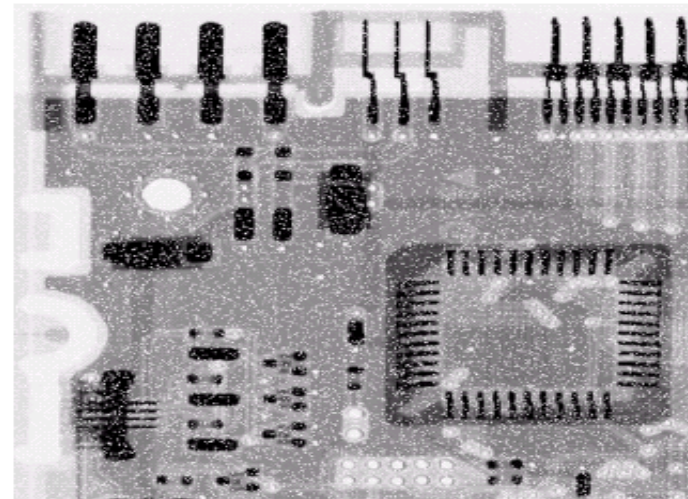
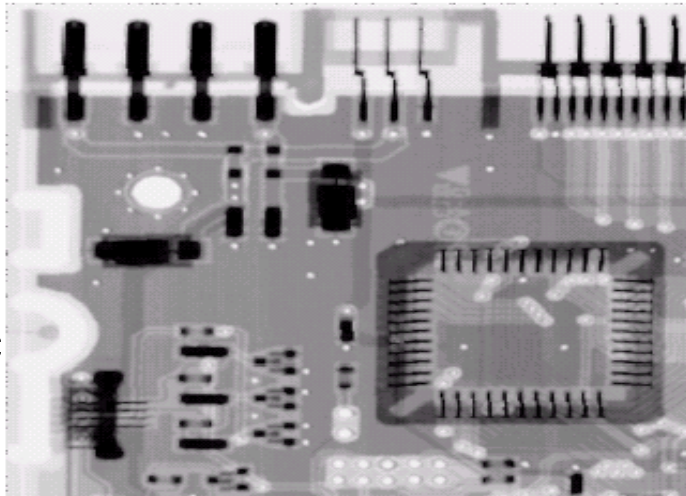


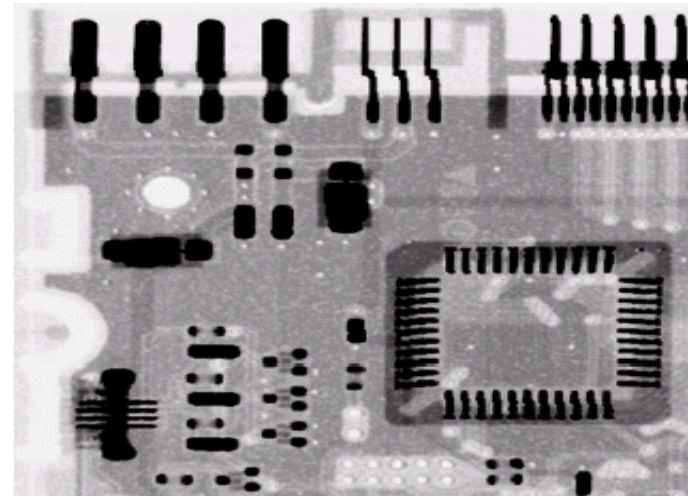
Image  
Corrupted  
By Salt  
Noise



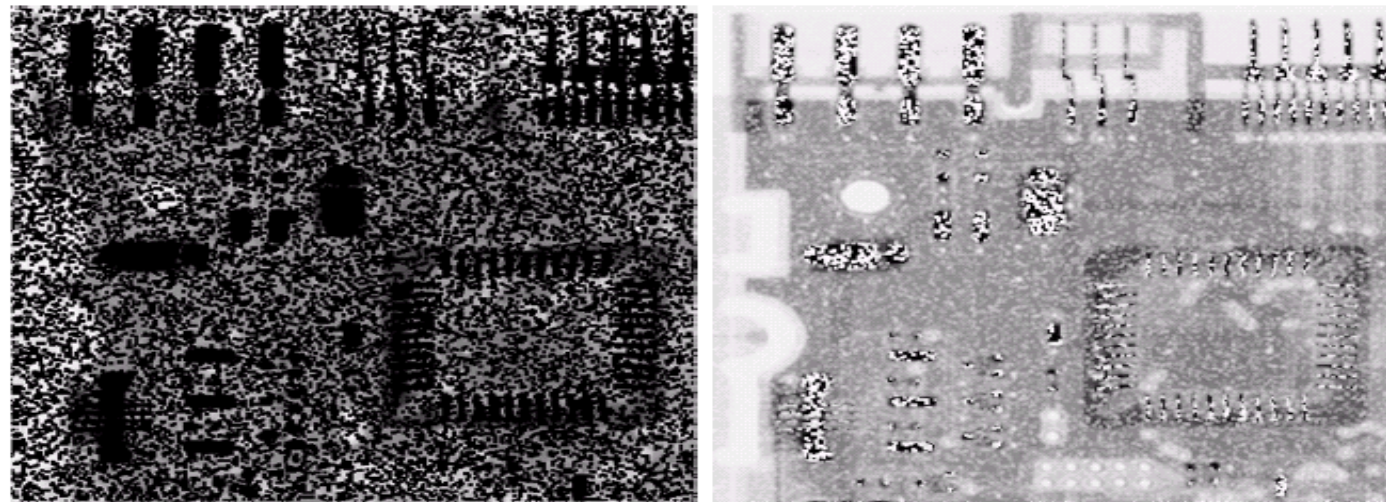
Result of  
Filtering Above  
With 3\*3  
Contra-harmonic  
 $Q=1.5$



Result of  
Filtering Above  
With 3\*3  
Contra-harmonic  
 $Q=-1.5$



## Lọc nhiễu - dùng lọc không gian



Chọn không đúng giá trị của  $Q$  khi sử dụng lọc **contraharmonic** có thể mạng lại những kết quả không mong muốn

- Dùng lọc không gian dựa trên giá trị pixel lân cận của bộ lọc (lọc thống kê)
  - Lọc trung vị
  - Lọc Max, Min
  - Lọc Midpoint
  - Lọc cắt Alpha

- **Lọc trung vị**

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

⇒ không gây hiệu ứng làm mịn ảnh như các lọc trung bình,

⇒ Tốt cho lọc nhiễu muối và hạt tiêu



- **Lọc Max**

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

- **Lọc Min**

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

⇒ Lọc Max tốt cho loại bỏ nhiễu hạt tiêu

⇒ Lọc Min tốt cho loại bỏ nhiễu muối

- **Lọc Midpoint**

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

⇒ Tốt loại bỏ nhiễu Gaussian và nhiễu đồng nhất

- **Alpha-Trimmed**

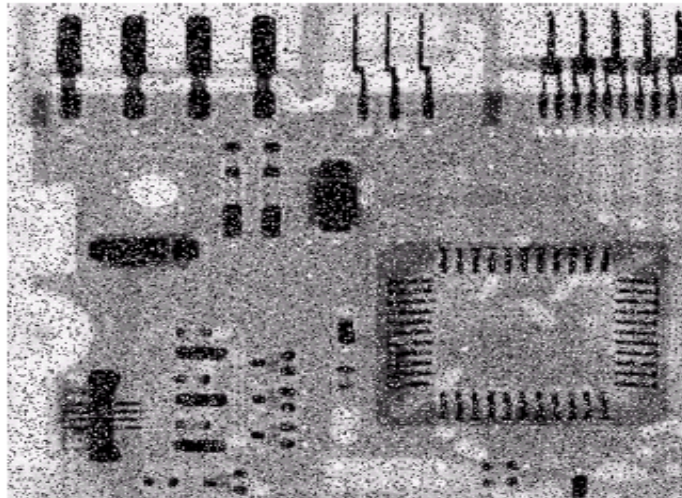
$$\hat{f}(x, y) = \frac{1}{mn - d} \sum_{(s,t) \in S_{xy}} g_r(s, t)$$

⇒ loại bỏ  $d/2$  mức xám lớn nhất và  $d/2$  mức xám bé nhất

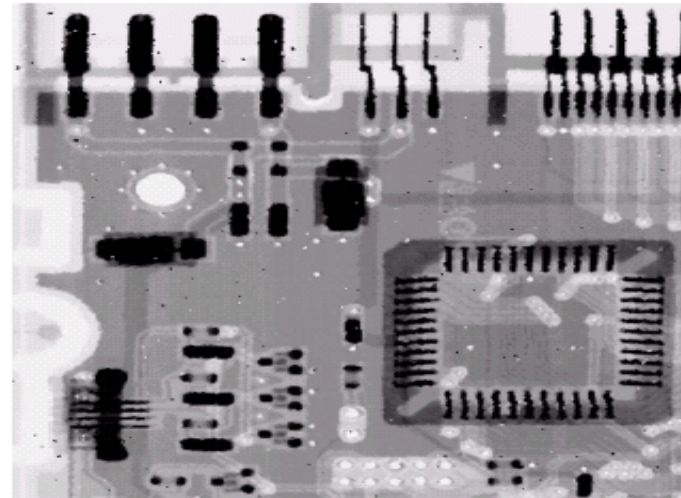
⇒  $g_r(s, t)$  biểu diễn phần còn lại của  $mn - d$  pixels



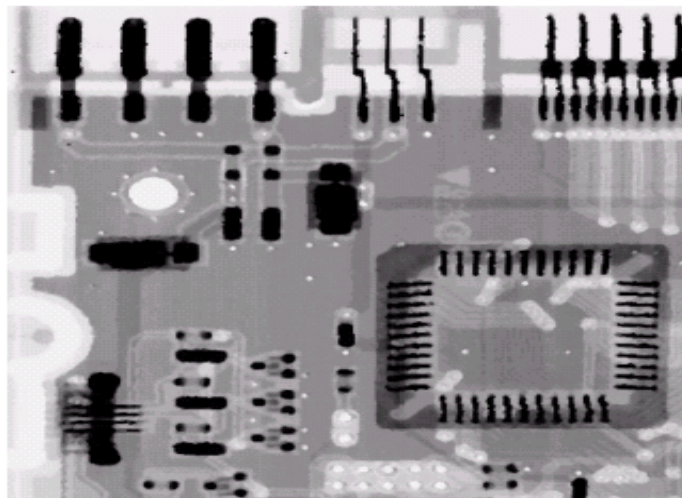
Image  
Corrupted  
By Salt And  
Pepper Noise



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

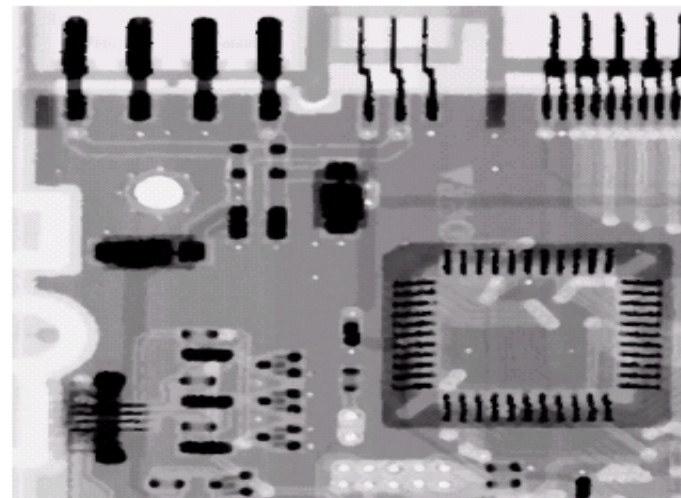




Image  
Corrupted  
By Pepper  
Noise

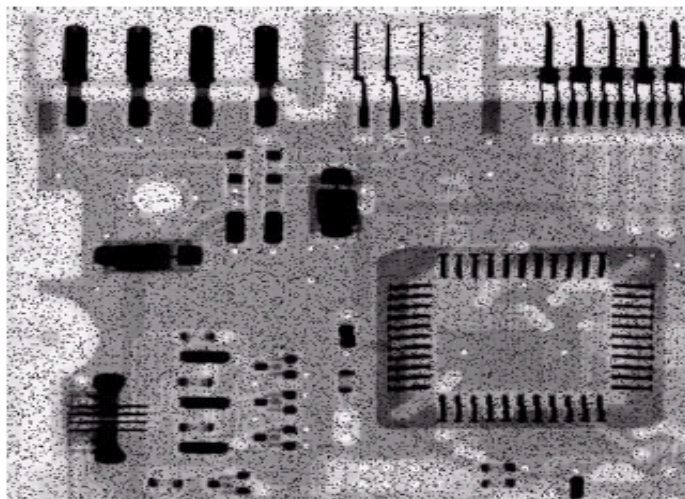
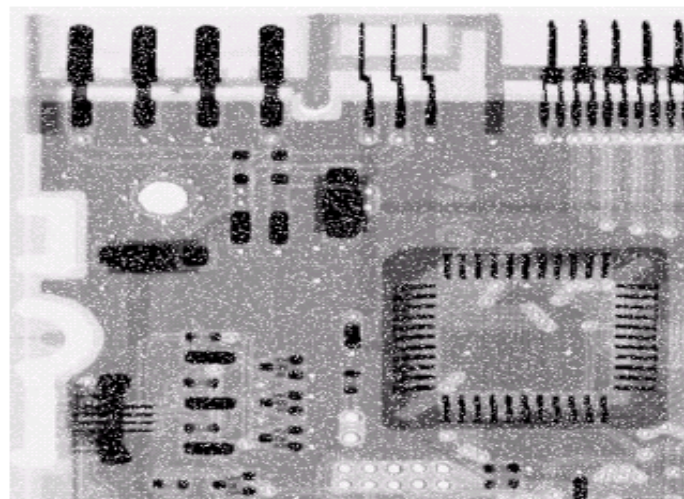
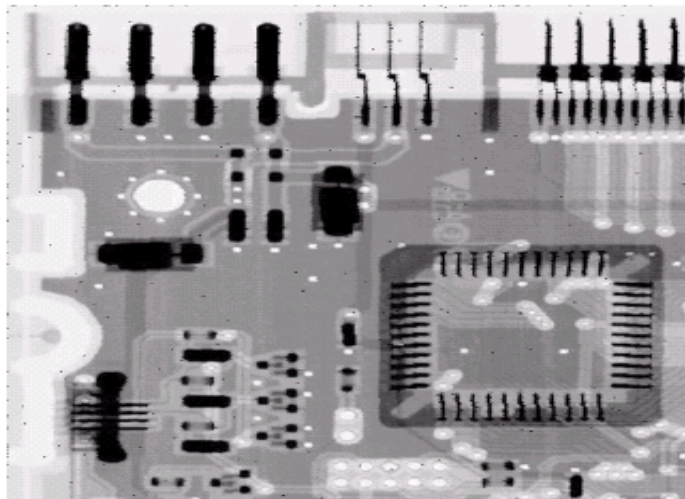


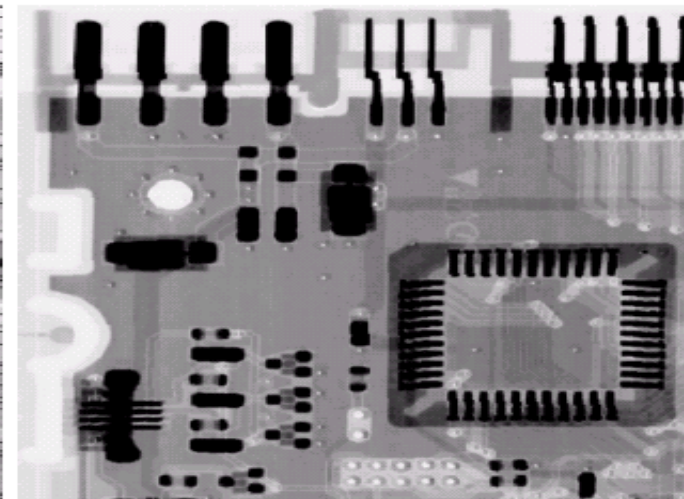
Image  
Corrupted  
By Salt  
Noise



Result Of  
Filtering  
Above  
With A 3\*3  
Max Filter



Result Of  
Filtering  
Above  
With A 3\*3  
Min Filter



# Lọc nhiễu - dùng lọc không gian

Image  
Corrupted  
By Uniform  
Noise

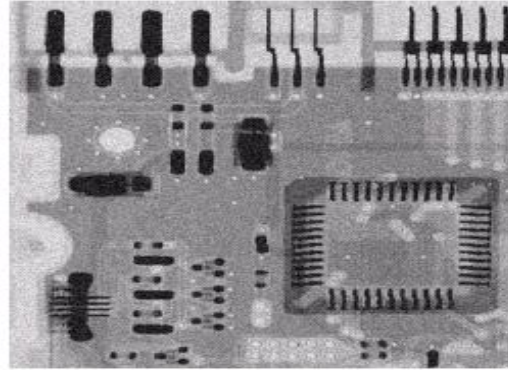
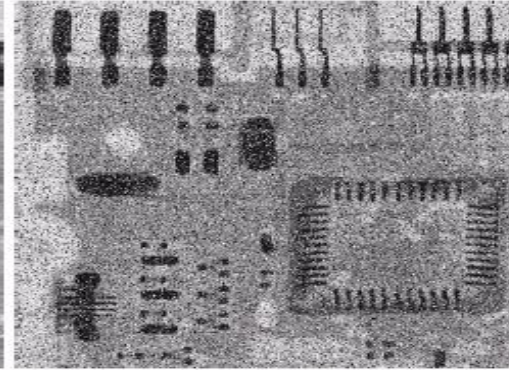
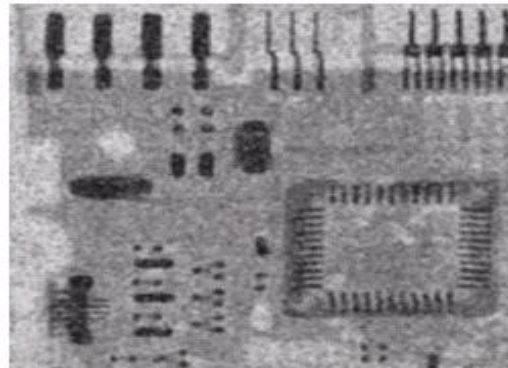


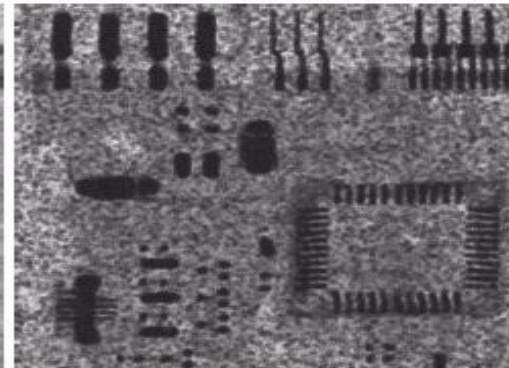
Image Further  
Corrupted  
By Salt and  
Pepper Noise



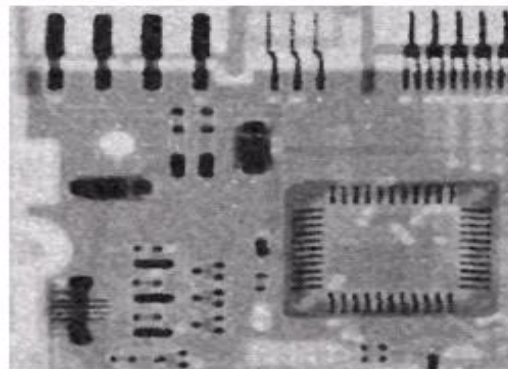
Filtered By  
5\*5 Arithmetic  
Mean Filter



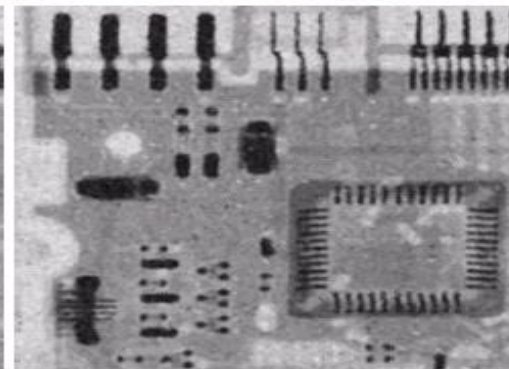
Filtered By  
5\*5 Geometric  
Mean Filter



Filtered By  
5\*5 Median  
Filter



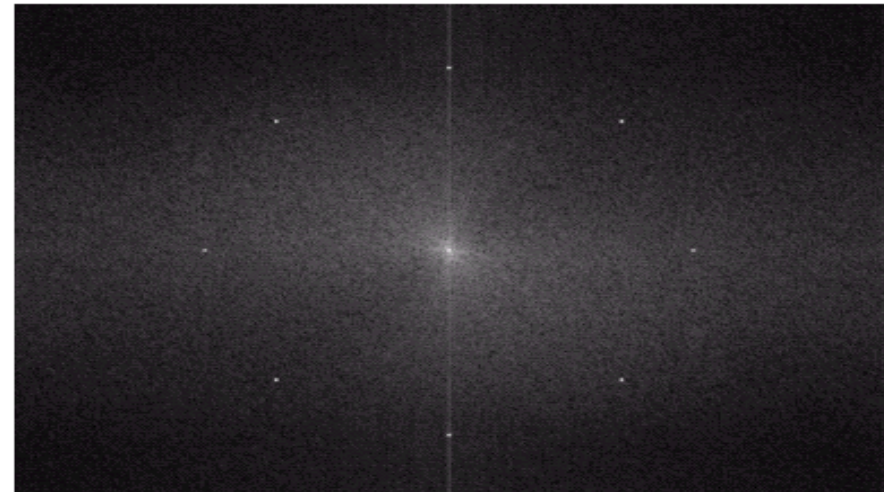
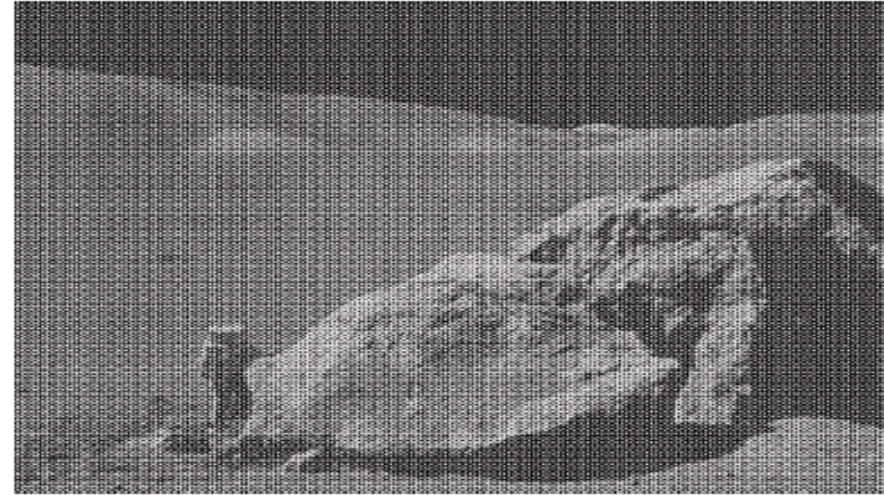
Filtered By  
5\*5 Alpha-Trimmed  
Mean Filter





## Nhiều lặp lại:

- Xuất hiện do can thiệp của điện hoặc điện từ
- Các kỹ thuật trong miền tần số nhằm loại bỏ hiệu quả các nhiễu định kỳ



## Lọc loại bỏ Band

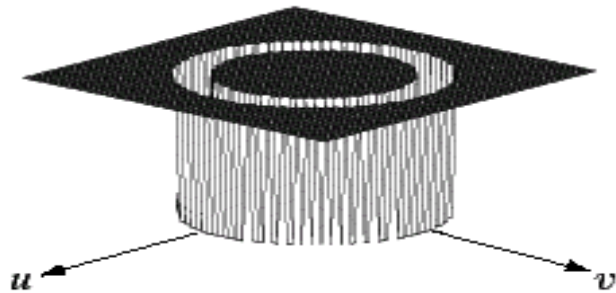
- Loại bỏ nhiễu định kỳ liên quan đến loại bỏ một phạm vi tần số trong ảnh  
 $\Rightarrow$  sử dụng lọc loại bỏ Band
- Lọc loại bỏ Band lý tưởng như sau:

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

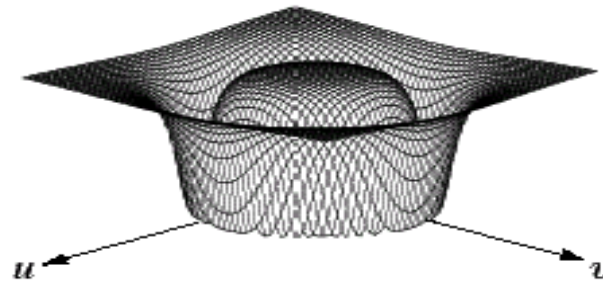


## Lọc loại bỏ Band

- Lọc loại bỏ Band lý tưởng, Butterworth và Gaussian



Ideal Band  
Reject Filter



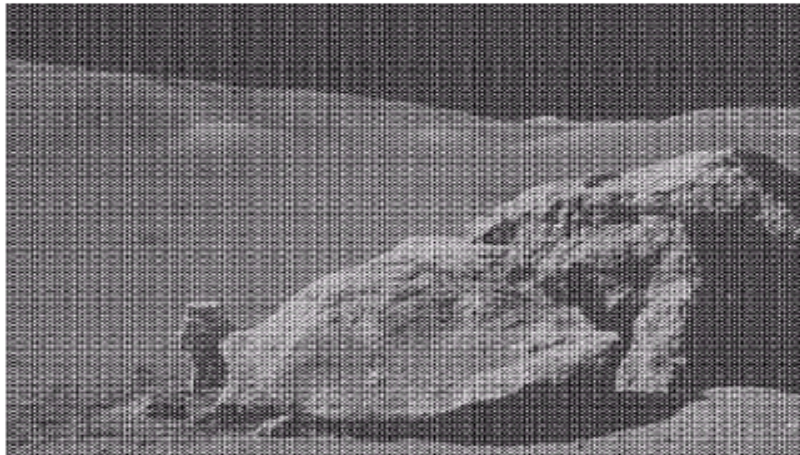
Butterworth  
Band Reject Filter  
(of order 1)



Gaussian  
Band Reject Filter

## Ví dụ lọc loại bỏ Band

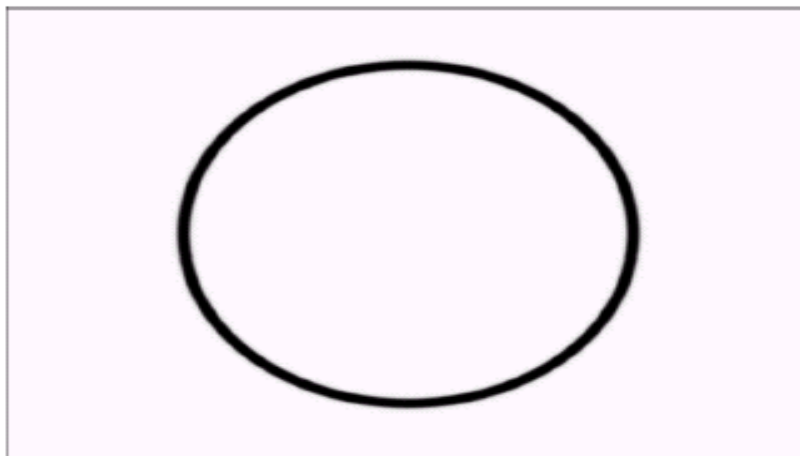
Image corrupted by sinusoidal noise



Fourier spectrum of corrupted image



Butterworth band reject filter



Filtered image



⇒ **Phục hồi ảnh bằng cách loại bỏ nhiễu:**

- Phục vụ nhiều mục đích hơn cải thiện ảnh
- Các kỹ thuật trong miền không gian rất hữu ích cho loại bỏ nhiễu
- Các kỹ thuật trong miền tần số hữu ích cho loại bỏ nhiễu định kỳ

Các bộ lọc trên (không gian, tần số) áp dụng cho toàn bộ ảnh mà không để ý đến các đặc điểm của ảnh, chúng thay đổi theo pixel

⇒ **Lọc tương thích** sẽ thay đổi theo vùng lọc bên trong ảnh

- **Lọc trung vị tương thích**

- Làm việc khá tốt với các nhiễu xung khi mật độ nhiễu xung không quá lớn
- Có thể xử lý nhiễu xung cường độ cao và làm mịn đối với các nhiễu không xung
- Đặc trưng chính của lọc trung vị tương thích là kích thước bộ lọc thay đổi phụ thuộc vào đặc tính của ảnh

⇒ Quá trình lọc sẽ thao tác lên mỗi pixel của ảnh ban đầu và tạo ra pixel lọc

Các khái niệm:

- $z_{min}$  mức xám tối thiểu trong  $S_{xy}$
- $z_{max}$  mức xám tối đa trong  $S_{xy}$
- $z_{med}$  trung vị của mức xám trong  $S_{xy}$
- $z_{xy}$  mức xám tại tọa độ  $(x, y)$
- $S_{max}$  kích thước tối đa của  $S_{xy}$



Mức 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  $\leq$  repeat  $S_{max}$  level A

Else output  $z_{med}$

Mức B:

$$B1 = z_{xy} - z_{min}$$

$$B2 = z_{xy} - z_{max}$$

If  $B1 > 0$  and  $B2 < 0$ , output  $z_{xy}$

Else output  $z_{med}$

Đặc điểm chính của lọc trung vị tương thích:

- Loại bỏ nhiễu xung
- Làm mịn đối với các nhiễu khác
- Giảm méo

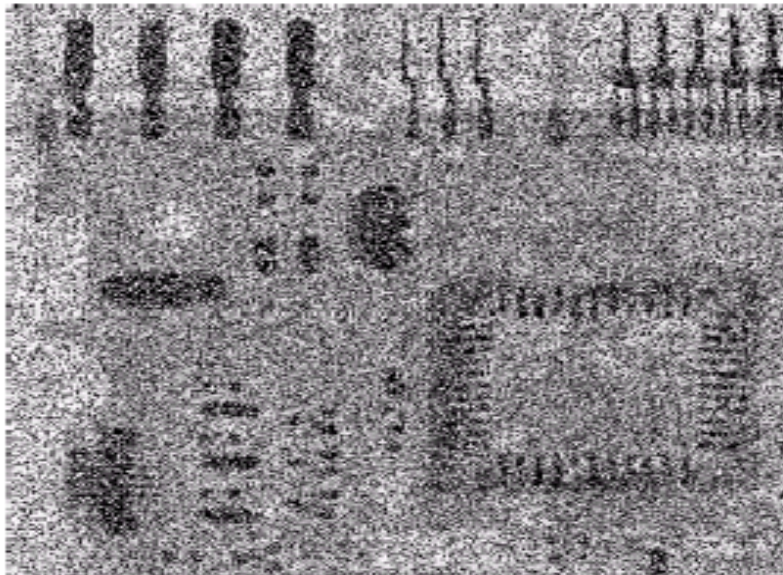
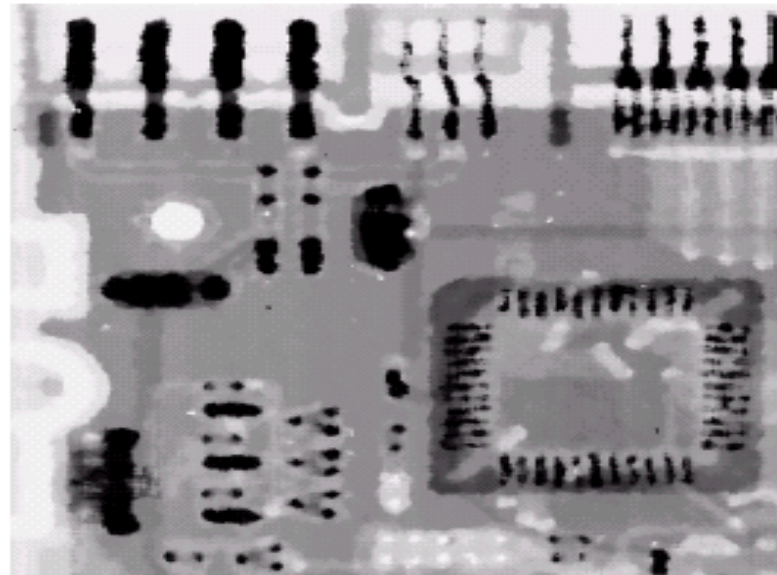
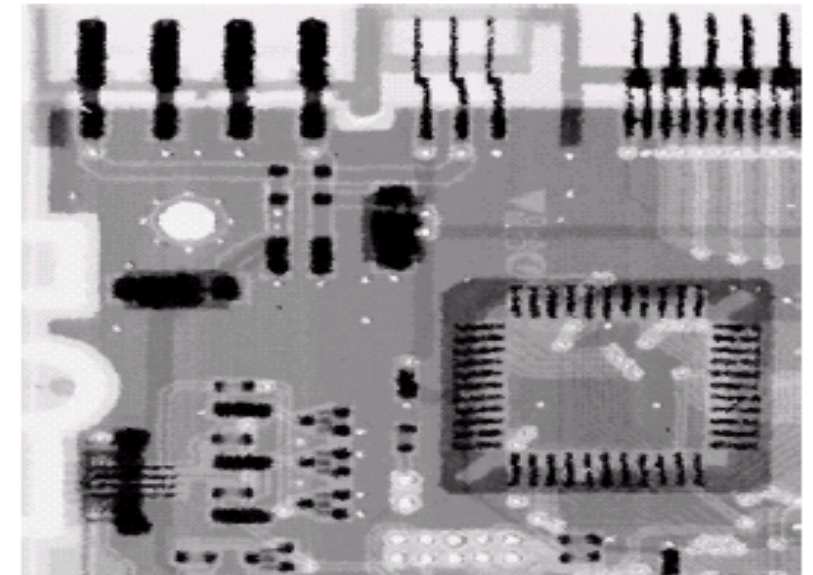


Image corrupted by salt & pepper noise with probabilities  $P_a = P_b = 0.25$



Result of filtering with a 7\*7 median filter



Result of adaptive median filtering with  $i = 7$





**Thank You...!**