IE 404

Image Restoration

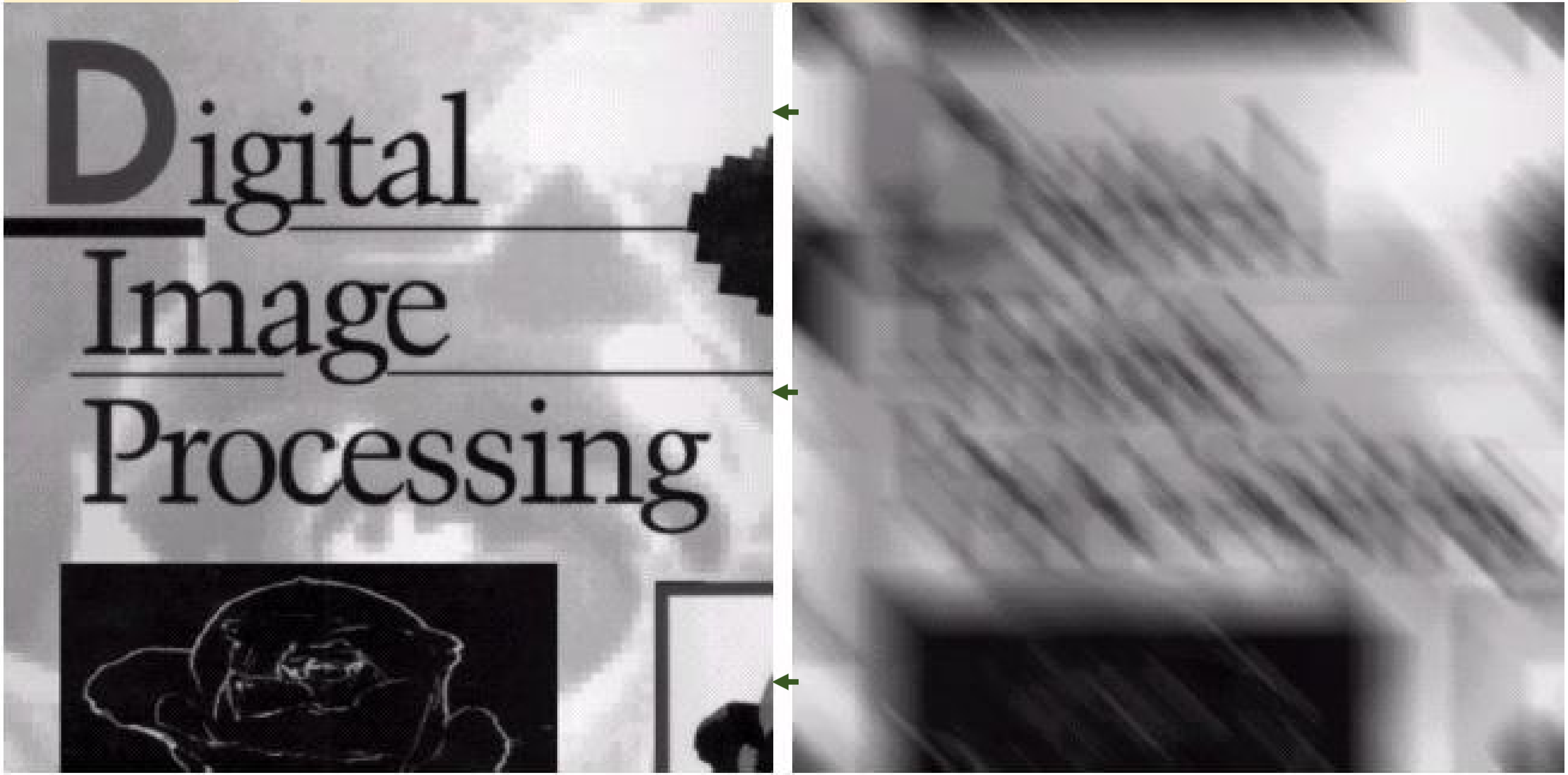
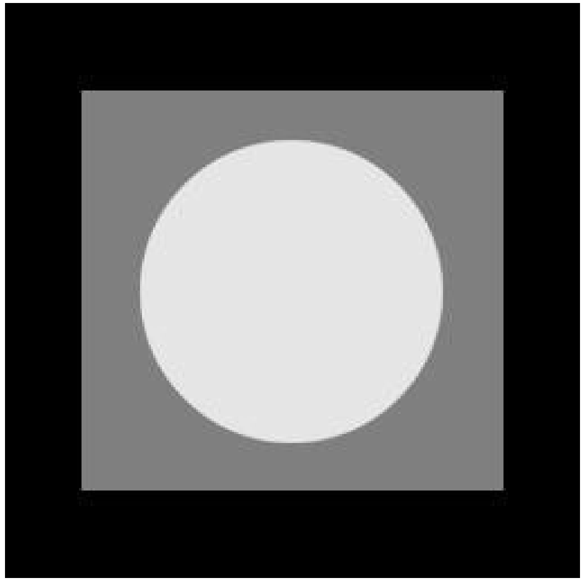


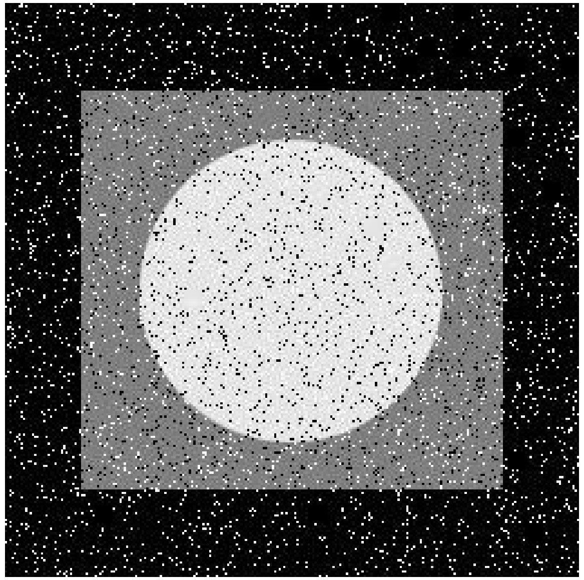
Image degradation

Input signal or image → **Degradation** e.g. camera motion, transmission channel etc. → Degraded image e.g. noisy or blurred image → **Improving image quality** e.g. enhancement, restoration → Improved image

Un-degraded image



Noise affected



Most images in this ppt have been taken from the DIP book by Gonzalez & Woods

Image degradation

Input signal or image → **Degradation** e.g. camera motion, transmission channel etc. → Degraded image e.g. noisy or blurred image → **Improving image quality** e.g. enhancement, restoration → Improved image

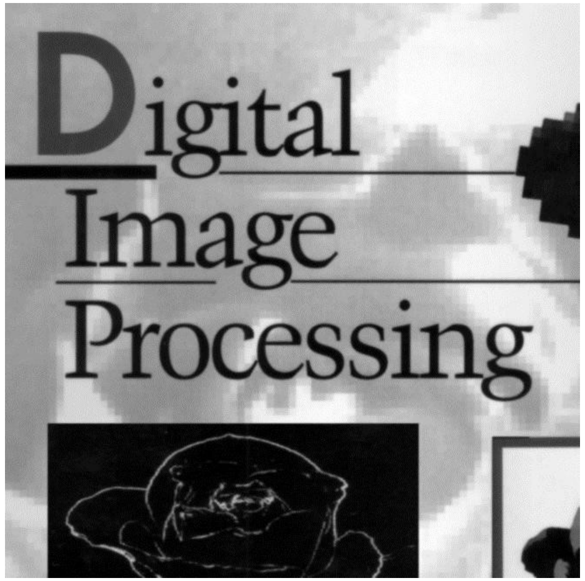
Un-degraded image

Turbulence affected

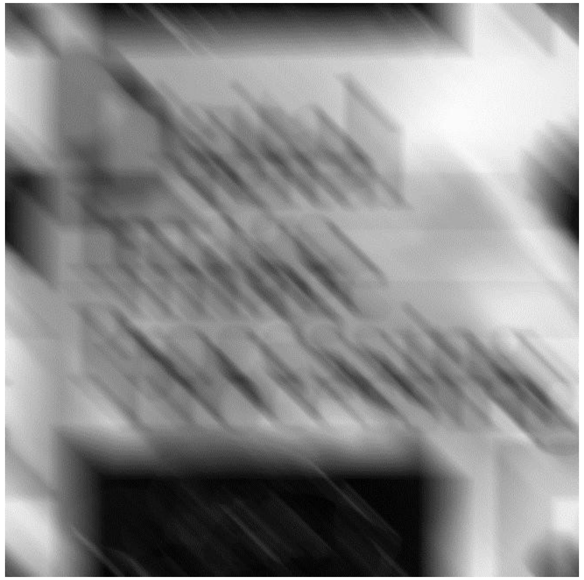
Image degradation

Input signal or image → **Degradation** e.g. camera motion, transmission channel etc. → Degraded image e.g. noisy or blurred image → **Improving image quality** e.g. enhancement, restoration → Improved image

Un-degraded image



Motion blurred



Camera is moving

Image degradation

Input signal or image → **Degradation** e.g. camera motion, transmission channel etc. → Degraded image e.g. noisy or blurred image → **Improving image quality** e.g. enhancement, restoration → Improved image

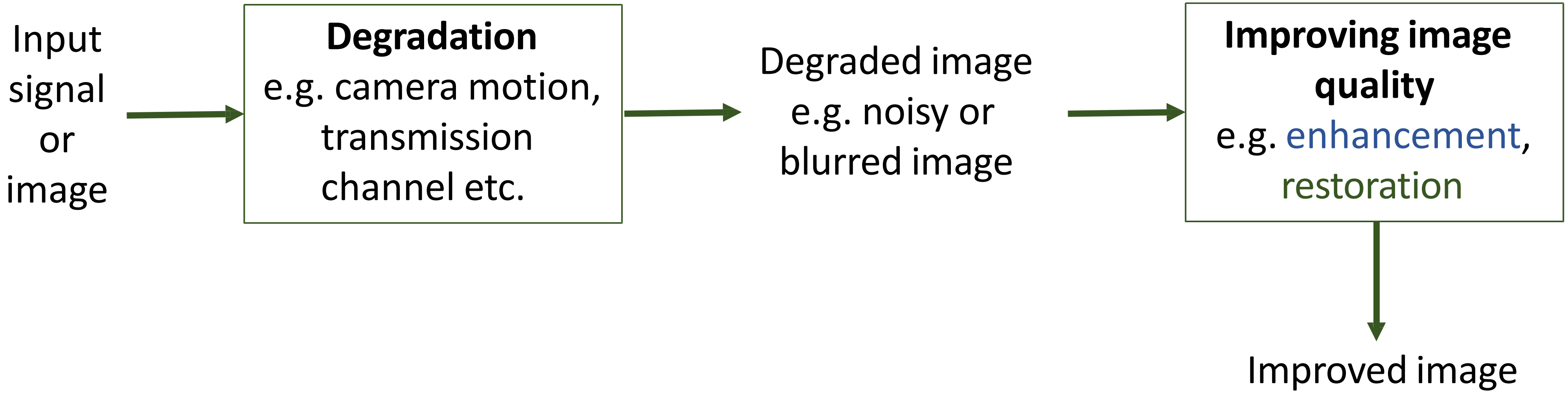
Motion blurred (object)

Object is moving

Motion blurred (background)

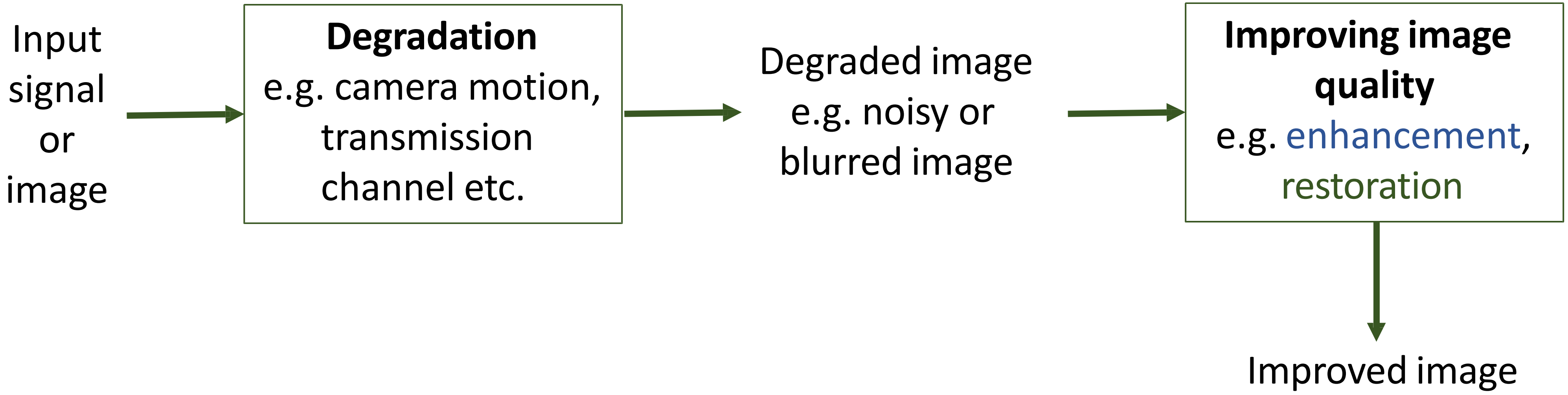
Camera and object are moving at same speed

Image restoration vs enhancement



- Image enhancement: subjective process
  - Try different techniques with different parameters and select the best method for enhancement
  - Example: use different filters and mask size
  - use different gamma value in correction
  - use different transformation functions

Image restoration vs enhancement



- Image restoration: objective process
  - We have knowledge about the degradation model (e.g. velocity of object/camera)
  - Use that to undo the degradation and restore the image
  - Any system has finite response time, it cannot pass all frequencies
  - Example - Signal has many high frequency component; so some kind of degradation e.g. blurring may occur if high frequencies are removed
  - For restoration, we are trying find models that mimic natural systems that do degradation

Degradation is not always bad!

- Object of interest is sharp in focus: good quality
- Other regions e.g. background are not in focus: blurred
  - because camera has finite depth of field
  - Degradation is somewhat deliberately introduced by the photographer by focussing on objects
- Even Google Meet introduced option for blurring background!

good pictures, even though blurred

Bad quality images

- These images need restoring

images from internet

Nutshell

- Image-restoration techniques aim at reversing the degradation undergone by an image to recover the true image.
- Images may be corrupted by degradation such as linear frequency distortion, noise, and blocking artifacts.
- The degradation consists of two distinct processes:—
  - the deterministic blur and
  - the random noise.
- The blur may be due to a number of reasons, such as motion, defocusing, and atmospheric turbulence.
- The noise may originate in the image-formation process, the transmission process, or a combination of them.
- Most restoration techniques model the degradation process and attempt to apply an inverse procedure to obtain an approximation of the original image.
- Iterative image restoration techniques often attempt to restore an image linearly or non-linearly by minimizing some measures of degradation such as maximum likelihood, constrained least square, etc.
- Blind restoration techniques attempt to solve the restoration problem without knowing the blurring function.
- No general theory of image restoration has yet evolved; however, some solutions have been developed for linear and planar invariant systems.

Image Degradation

- The process by which the original image is blurred is usually very complex and often unknown.
- To simplify the calculations, the degradation is often modeled as a linear function which is often referred as Point Spread Functions

- The different Causes of Image degradation are
  - Improper opening and closing of shutter
  - Atmospheric turbulence
  - Misfocus of lens
  - Relative motion between camera and object which causes motion blur

Types of Image Blur

- Blur can be introduced by an improperly focused lens, relative motion between the camera and the scene, or atmospheric turbulence.
- Blurring is a form of bandwidth reduction of an ideal image owing to the imperfect image formation process.
- It can be caused by relative motion between the camera and the scene or by an optical system that is out of focus.

- Image blur can be broadly classified as
  - Gaussian Blur
  - Out-of-focus blur
  - Motion blur

Gaussian Blur

Gauss blur is defined by the following point-spread function:

$$h(x, y) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Here, σ is called the variance of the blur. Gauss blur occurs due to long-time atmosphere exposure.

Out of Focus Blur

This blurring is produced by a defocused optical system. It distributes a single point uniformly over a disk surrounding the point. The point-spread function of the out-of-focus blur is given by

$$h(x, y) = c\begin{cases} 1, & \sqrt{(x-c_x)^2 + (y-c_y)^2} \le r \\ 0, & \text{otherwise} \end{cases}$$

where r is the radius and (c_x, c_y) is the centre of the out-of-focus point-spread function. The scaling factor c has to be chosen such that $\iint h(x, y)dxdy = 1$.

Motion Blur

- Motion blur is due to relative motion between the recording device and the scene.
- When an object or the camera is moved during light exposure, a motion blurred image is produced.
- The motion blur can be in the form of a translation, a rotation, a sudden change of scale, or some combination of these.
- When the scene to be recorded translates relative to the camera at a constant velocity $\vartheta_{\text{relative}}$ under an angle of ø radians with the horizontal axis during the exposure interval $[0, t_{exposure}]$, the distortion is one-dimensional.
- Defining the length of motion by

$$L = \vartheta_{\text{relative}} \times t_{\text{exposure}}$$

Motion Blur

The point-spread function is given by

$$h(x, y, L, \phi) = \begin{cases} \dfrac{1}{L} & \text{if } \sqrt{x^2 + y^2} \leq \dfrac{L}{2} \text{ and } \dfrac{x}{y} = -\tan\phi \\ 0 & \text{otherwise} \end{cases}$$

The discrete version of this equation is not easily captured in a closed form expression in general.
For special case, when ø = 0, an appropriate approximation is given by

$$h(n_1, n_2, L) = \begin{cases} \dfrac{1}{L} & \text{if } n_1 = 0, |n_2| \leq \left\lfloor \dfrac{L-1}{2} \right\rfloor \\ \dfrac{1}{2L}\left\{(L-1) - 2\left\lfloor \dfrac{L-1}{2} \right\rfloor\right\} & \text{if } n_1 = 0, |n_2| = \left\lceil \dfrac{L-1}{2} \right\rceil \\ 0 & \text{elsewhere} \end{cases}$$

(a) (b) (c) (d)

(a). Original Image, (b). Motion blur with [10,25], (c). Motion blur with [15,35], (d). Motion blur with [25,30]

Atmospheric Turbulence Blur

Atmospheric turbulence is a severe limitation in remote sensing.
Atmospheric turbulence depends on a variety of factors like temperature, wind speed, exposure time
Point spread function can be described reasonably well by a Gaussian function

$$h(x, y, \sigma_G) = \mathrm{C} \exp\left(-\frac{x^2 + y^2}{2\sigma_G^2}\right)$$

Here, σ_G determines the amount of spread of the blur, and the constant C is to be chosen

Classification of Image Restoration Technique

- Image-restoration techniques are methods which attempt the inversion of some degrading process.
- Image-restoration technique can be broadly classified into two types depending upon the knowledge of degradation.
- If the prior knowledge about degradation is known then the deterministic method of image restoration can be employed.
- If it is not known then the stochastic method of image restoration has to be employed.
- Classical linear techniques restore the true image by filtering the observed image using a properly designed filter. Examples are inverse filtering, Wiener filtering and the Tikhonov—Miller algorithm.
- Restoration often exhibits ringing artifacts near the edges, as linear methods are unable to recover missing frequency components which lead to the Gibbs effect.
- Linear methods do not necessarily maintain image non-negativity or signal dependent noise. This has led to the development of non-linear and iterative restoration algorithms.

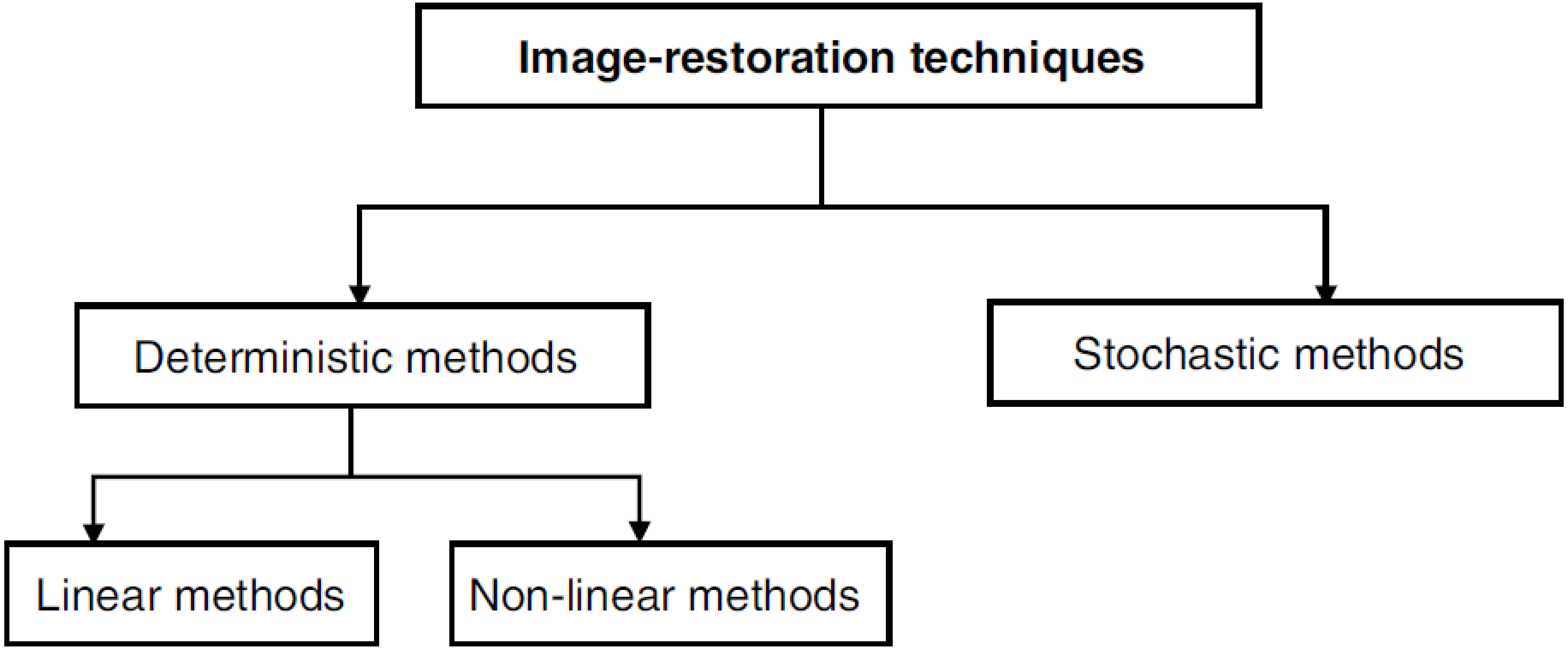
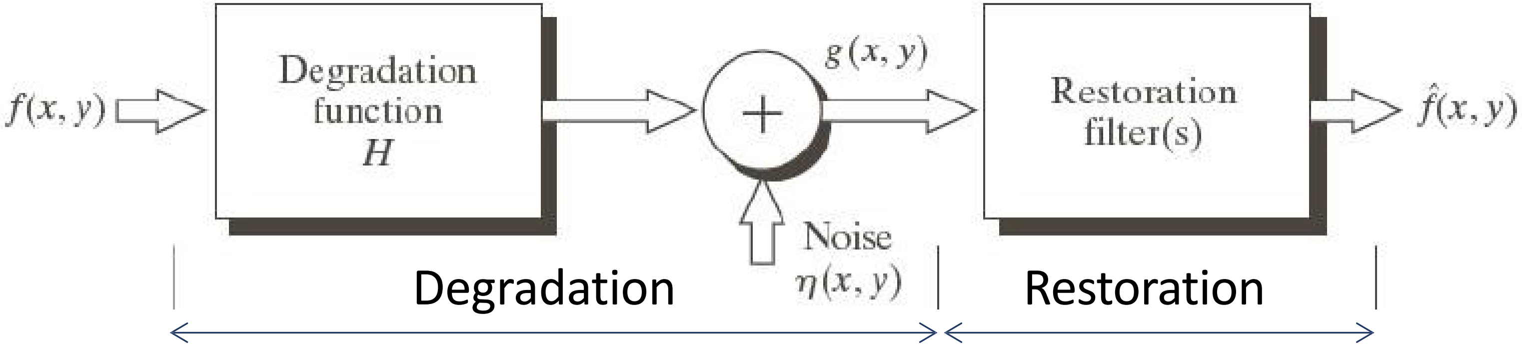


Image-restoration techniques
Deterministic methods
Stochastic methods
Linear methods
Non-linear methods

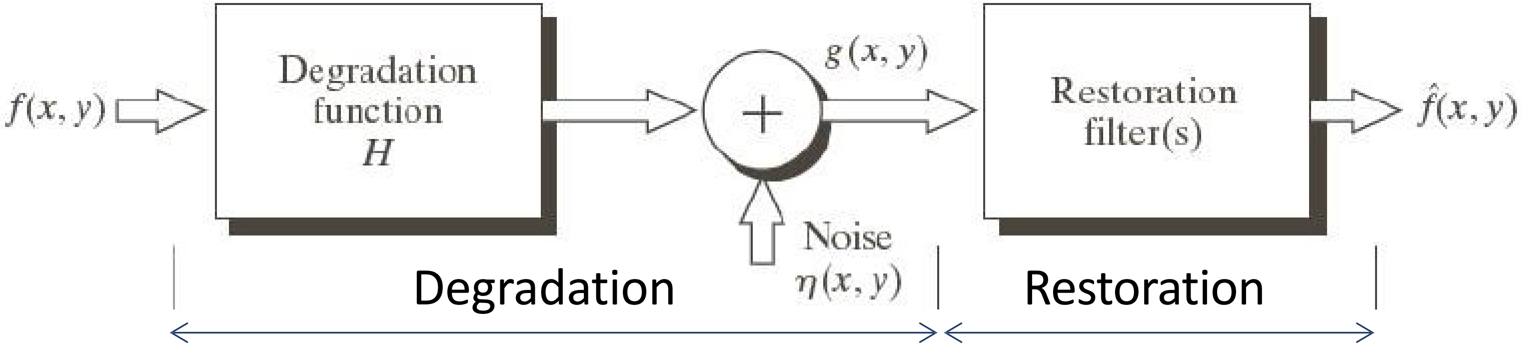
Degradation and restoration model overview



- Undegraded image $f(x, y)$
- Noise $\eta(x, y)$ and the system's impulse response $h(x, y)$
- In spatial domain, degraded image $g(x, y) = \varphi(f(x, y), h(x, y), \eta(x, y))$

Degradation operation

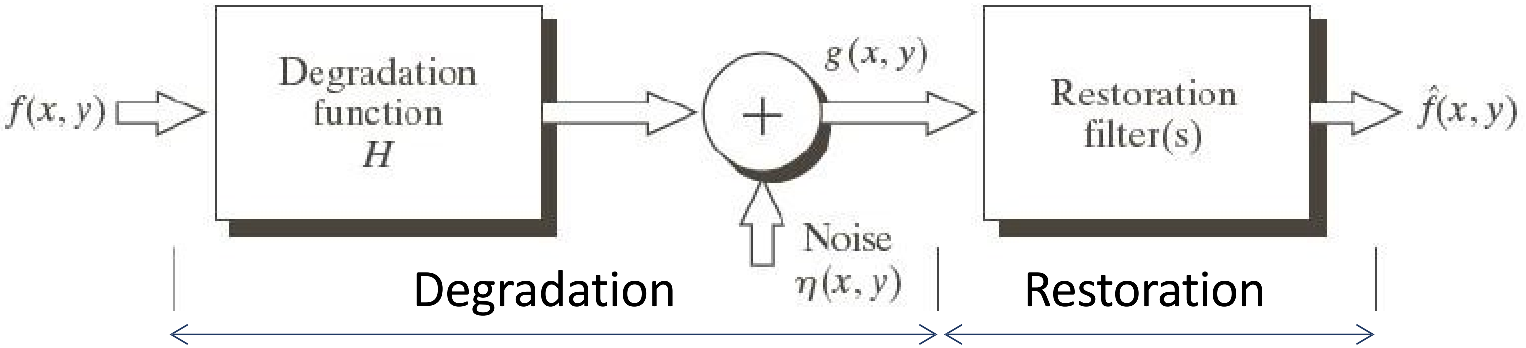
Degradation and restoration model overview



- If noise can be separated,
  - multiplcative noise: $g(x,y) = \psi(f(x,y), h(x,y)) \otimes \eta(x,y)$
  - noise effect is signal dependent
  - additive noise: $g(x,y) = \psi(f(x,y), h(x,y)) + \eta(x,y)$
  - this assumption holds good in many cases

$$\begin{bmatrix} 100 & 30 \\ 7 & 1 \end{bmatrix} * 2 = \begin{bmatrix} 200 & 60 \\ 14 & 2 \end{bmatrix}$$

Degradation and restoration model overview



- If system is linear and position/shift invariant (LSI), degraded image is modeled as
- In spatial domain $g(x, y) = f(x, y) * h(x, y) + \eta(x, y)$
- In frequency domain $G(u, v) = F(u, v)H(u, v) + N(u, v)$

$\psi()$ is convolution

- Goal is to estimate $\hat{f}(x, y)$ as close to $f(x, y)$ as possible given some knowledge of noise $\eta(x, y)$ and the system's impulse response $h(x, y)$ or the system function $H(u,v)$

Linear Image Restoration Technique

Inverse filter
Pseudo inverse filter
Wiener filter
Constrained least square filter

The linear restoration techniques are quick and simple but have limited capabilities

Inverse Filtering

(a) Original image (b) Degraded image (c) Restored image

Advantage:

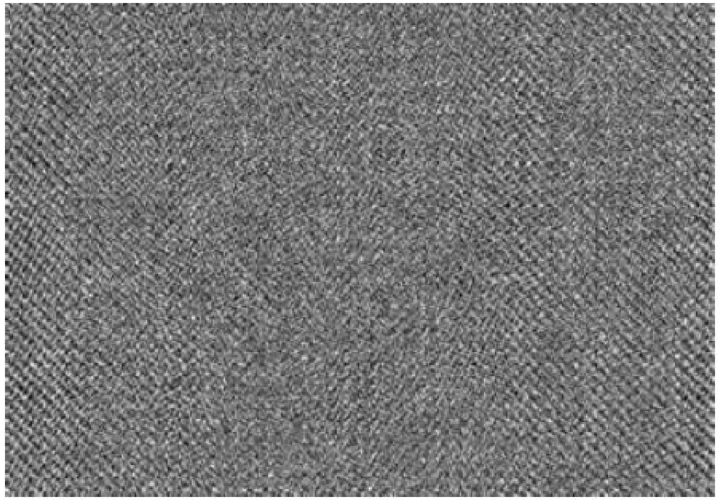
- The advantage of inverse filter is that it requires only the blur point spread function as a priori knowledge.
- The inverse filter produces perfect reconstruction in the absence of noise.

Drawbacks:

- The main drawback of inverse filtering is that it is not always possible to obtain an inverse.
- For an inverse to exist, the matrix should be non-singular. In case, it is difficult to obtain inverse filtering, the choice is to use a pseudo-inverse filter.
- Another main drawback of an inverse filter is that it will not perform well in the presence of noise. If noise is present in the image, the inverse filter will tend to amplify noise which is undesirable. In that case, Wiener filter is good option.

(a)

(b)



(c)

(a) Original image (b) Image degrade with noise (c) Restored image

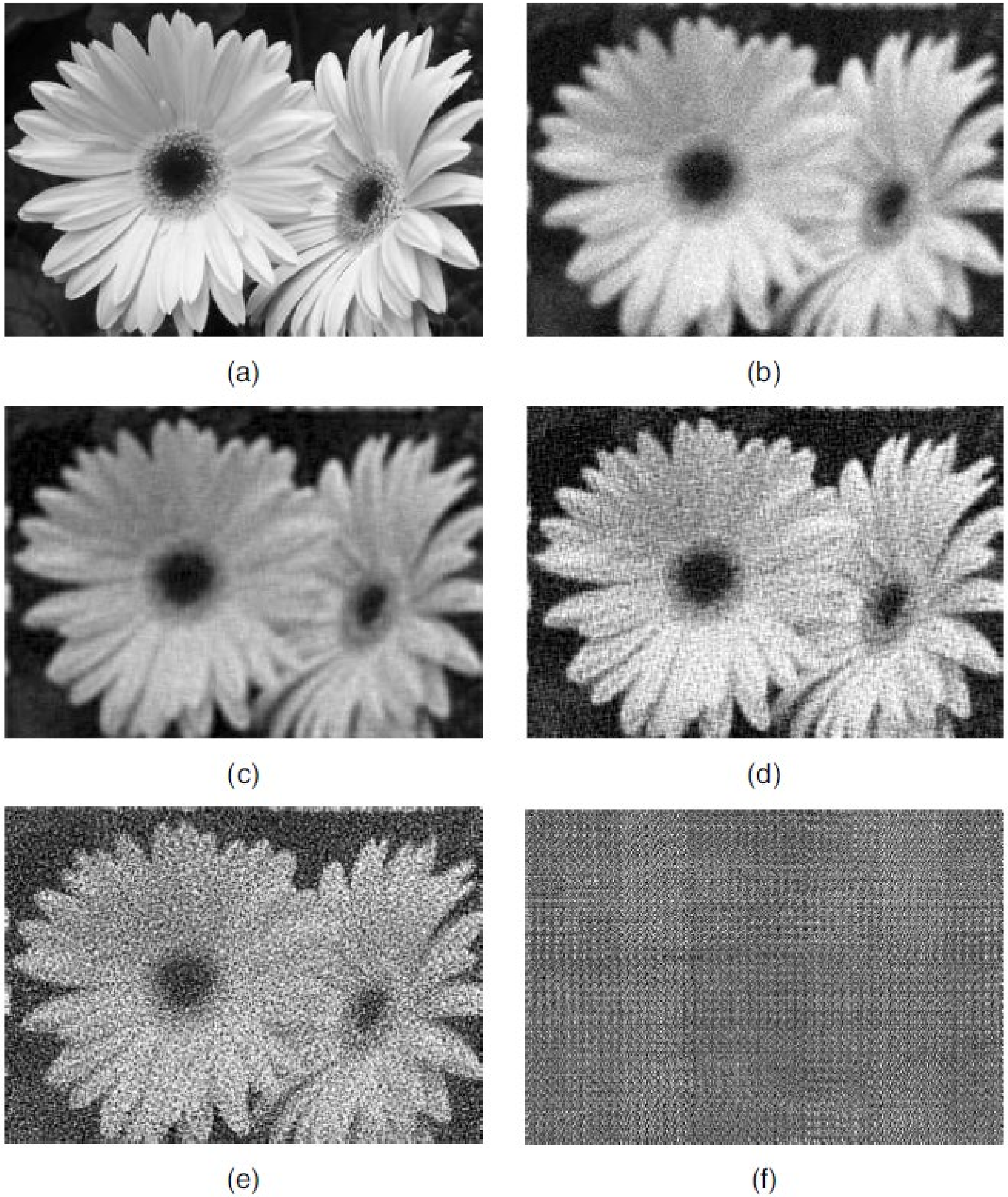
Pseudo Inverse Filter

The equation of an inverse filter in the frequency domain is given by $\hat{F}(k,l) = \dfrac{G(k,l)}{H(k,l)}$. Here, $H(k, l)$ represents the spectrum of point-spread function. Mostly, the point-spread function is a low-pass filter, which implies that $H(K, l) \approx 0$ at high frequencies. The division of $H(k, l)$ leads to large amplification at high frequencies, where the noise dominates over the image. This frequency-dependent amplification leads to significant errors in the restored image, and amplification of noise. To avoid these problems, a pseudo-inverse filter is defined as

$$\frac{1}{H} = \begin{cases} 1/H & \text{if } \; H > \varepsilon \\ \varepsilon & \text{if } \; H \leq \varepsilon \end{cases}$$

The value of ε affects the restored image. With no clear objective selection of ε, restored images are generally noisy and not suitable for further analysis.

(a). Original Image
(b). Image degraded with noide
(c). Restored Image with Ɛ = 0.2
(d). Restored Image with Ɛ = 0.02
(e). Restored Image with Ɛ = 0.002
(f). Restored Image with Ɛ = 0



SVD Approach to Pseudo Inverse Filter

- SVD stands for singular value decomposition.
- Using SVD technique, any matrix can be decomposed into series of eigen matrices.
- The basic strategy in SVD based image restoration is to decompose the blur matric into eigen matrices.
- From the image restoration mode, we have

$$G=Hf + \eta$$

Wiener Filter

- The Wiener filter tries to build an optimal estimate of the original image by enforcing a minimum mean square error constraint between estimate and original image.
- The wiener filter is an optimum filter.
- The objective of a wiener filter is to minimize the mean square error.
- A wiener filter has the capability of handling both the degradation function as well as noise

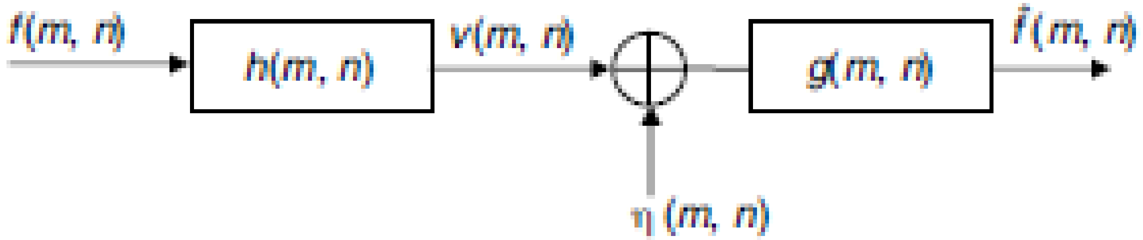


Fig. 6.11 *Image-degradation model*


(a)


(b)


(c)

Constrained Least Square Filter

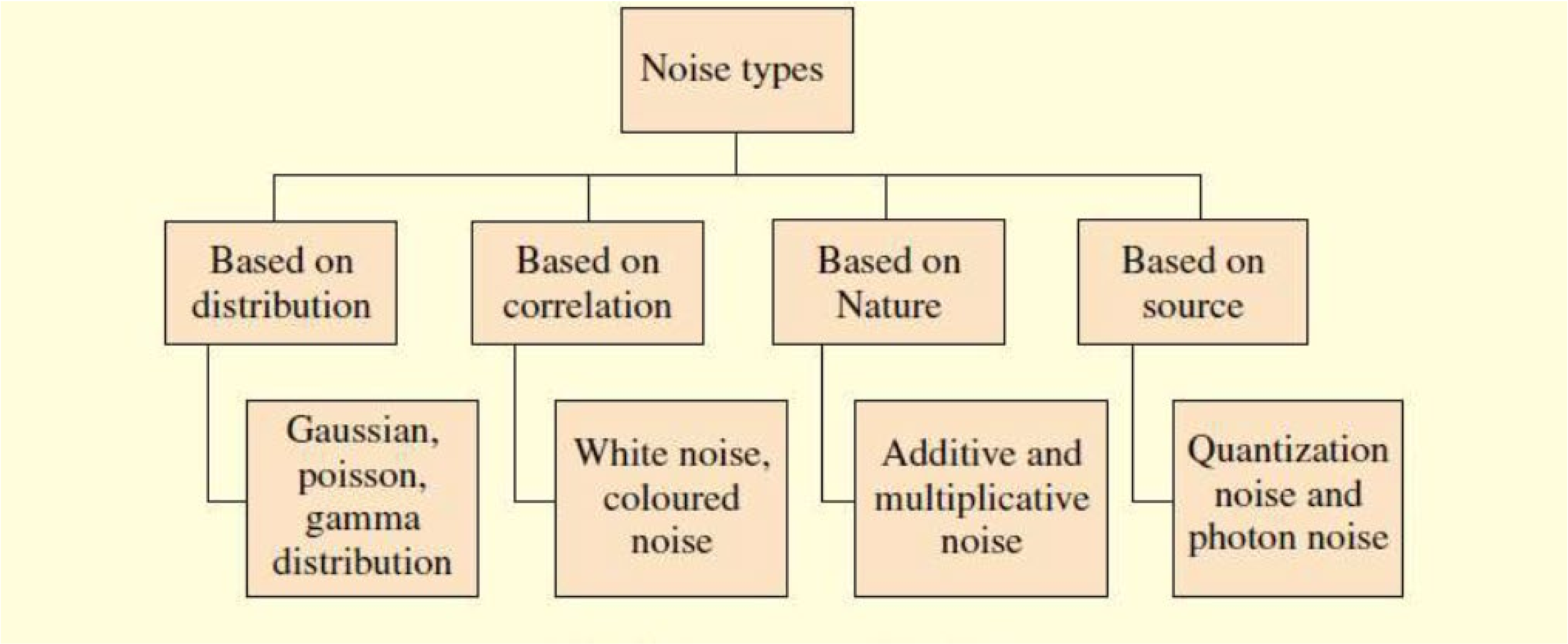
- The effect of information loss in the degraded image can often be mitigated by constraining the restoration.
- Constraints have the effect of adding information to the restoration process.
- If these constraints represent additional knowledge of the original scene to be recovered then this knowledge should contribute to a more faithful restoration of the image.
- Constrained restoration refers to the process of obtaining a meaningful restoration by biasing the solution toward the minimiser of some specified constraint functional.
- Constrained least-square filter is a regularisation technique which adds the Lagrange multiplier, λ, to control the balance between noise artifacts and consistency with the observed data.

The constrained least square reconstruction is given by

$$\hat{F}(k,l) = \left[\frac{H^*(k,l)}{|H(k,l)|^2 + \lambda |P(k,l)|^2} \right] G(k,l)$$

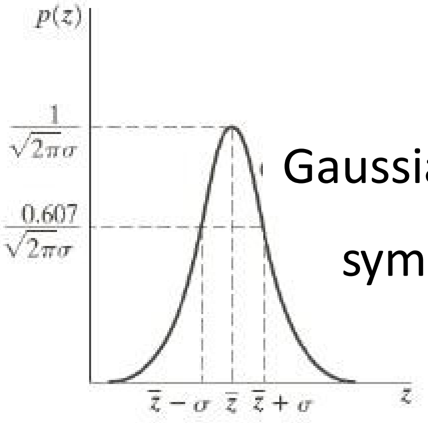
Here, P(k,l) is the Fourier transform of the Laplacian filter.
The filter P(k,l) has a large amplitude at high frequencies, where the noise tends to be dominant.
It modifies the denominator to reduce the noise effects at high frequencies.
Proper choice of P(k,l) and λ can minimize higher order derivatives.

Type of Noise



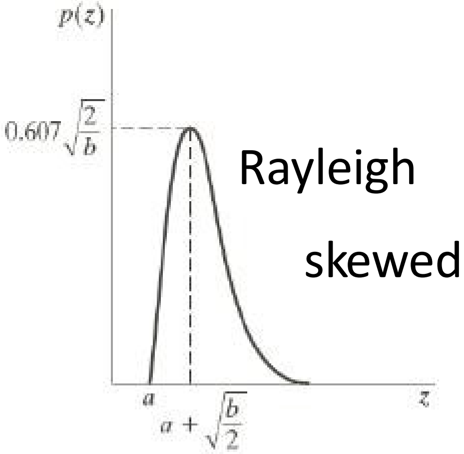
Degradation considering only noise

- Different noise distributions (probability density functions)



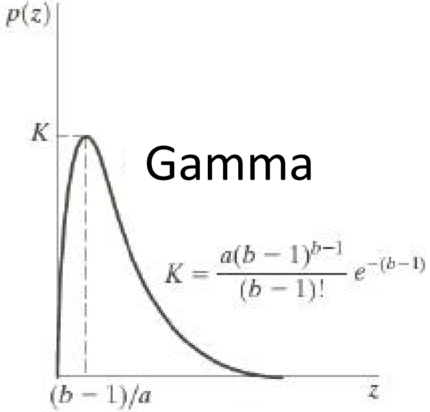
Gaussian

symmetric



Rayleigh

skewed



Gamma

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

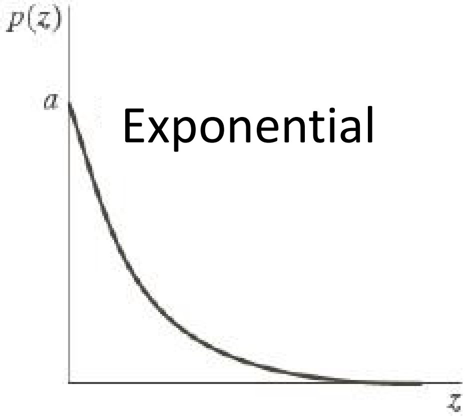
$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a \end{cases}$$

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

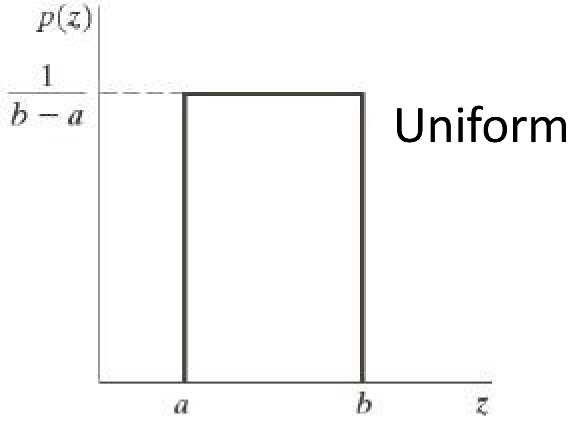
	Gaussian	Rayleigh	Gamma
Mean	$\bar{z}$	$\bar{z} = a + \sqrt{\pi b/4}$	$\bar{z} = \frac{b}{a}$
Variance	σ^2	$\sigma^2 = \frac{b(4-\pi)}{4}$	$\sigma^2 = \frac{b}{a^2}$

Degradation considering only noise

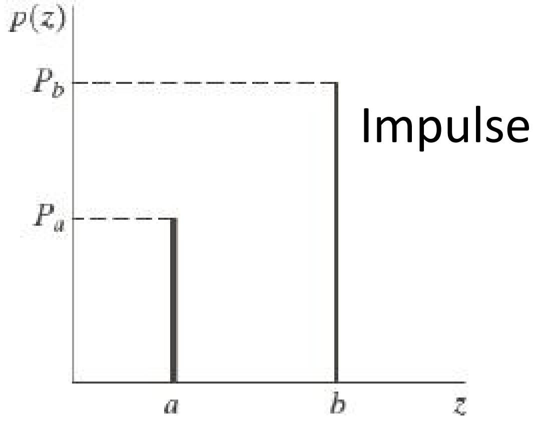
- Different noise distributions (probability density functions)



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



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



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

	Exponential	Uniform	Impulse
Mean	$\bar{z} = \frac{1}{a}$	$\bar{z} = \frac{a+b}{2}$	/
Variance	$\sigma^2 = \frac{1}{a^2}$	$\sigma^2 = \frac{(b-a)^2}{12}$	/

Gaussian Noise

$$F = f(x, y) \pm N_a$$

where N_a is the Gaussian PDF and $f(x, y)$ is the noiseless image.

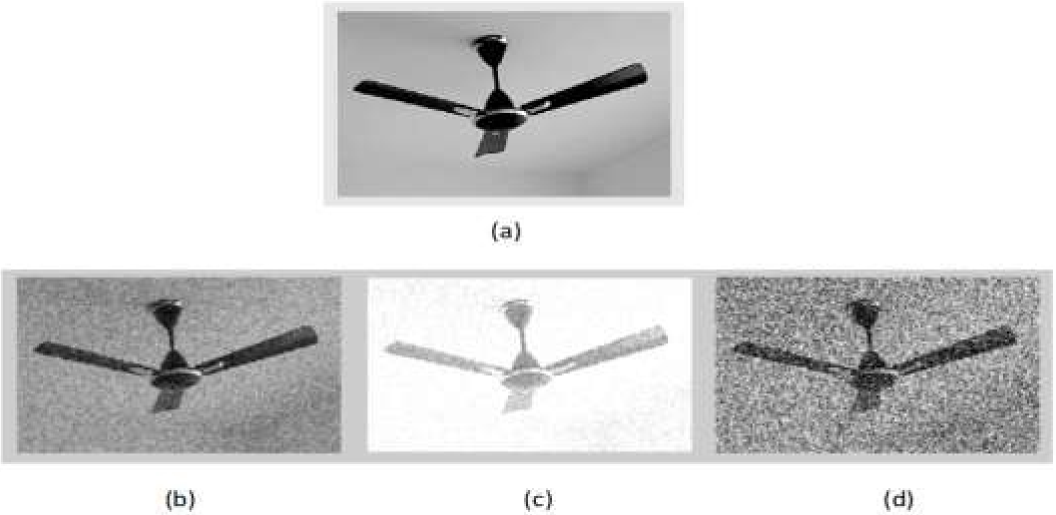


Fig. 6.10 Gaussian noise (a) Original image (b) Image with Gaussian noise (default variance = 0.01) (c) Image with Gaussian noise (mean = 0.5, variance = 0.01) (d) Image with Gaussian noise (mean = 0, variance = 0.07)

Exponential Noise

$$P(z) = \begin{cases} a \times e^{-az} & \text{for } z \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

The mean and variance are given as $\frac{1}{a}$ and $\frac{1}{a^2}$, respectively.

(a) (b)

Fig. 6.12 Illustration of exponential noise (a) Original image (b) Image with exponential noise

Gamma Noise

$$P(z) = \begin{cases} \dfrac{a^b \times z^{b-1}}{(b-1)!} e^{-a^2} & \text{for } z \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Fig. 6.13 Image with gamma noise

Rayleigh Noise

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

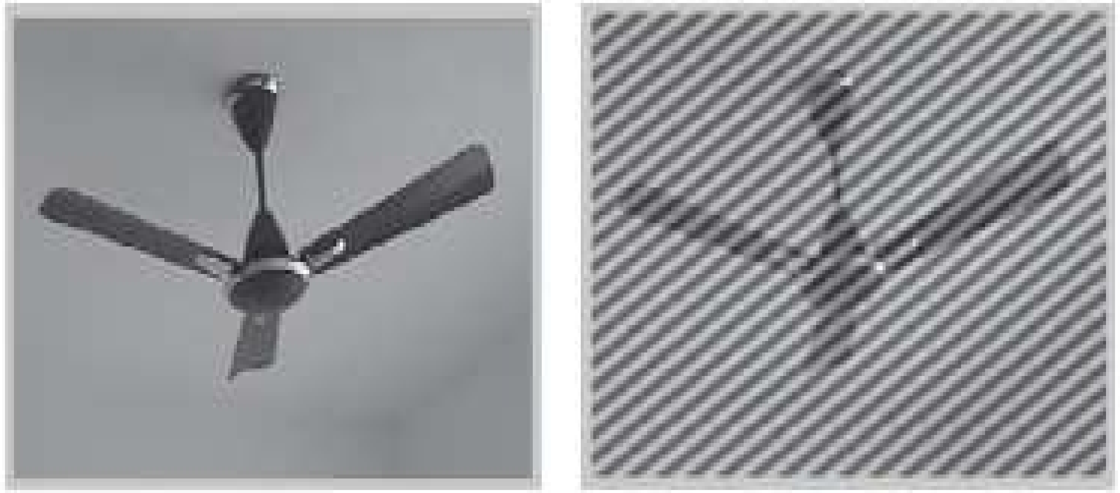
Fig. 6.14 Image with Rayleigh noise

Uniform Noise

$$P(z) = \begin{cases} \dfrac{1}{b-a} & \text{for } a \le f(x,y) \le b \\ 0 & \text{otherwise} \end{cases}$$

Fig. 6.15 Image with uniform noise

Periodic Noise

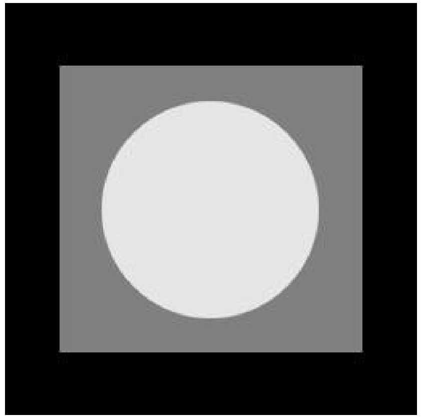


(a) (b)

Fig. 5.37 Periodic noise (a) Original image (b) Image with periodic noise

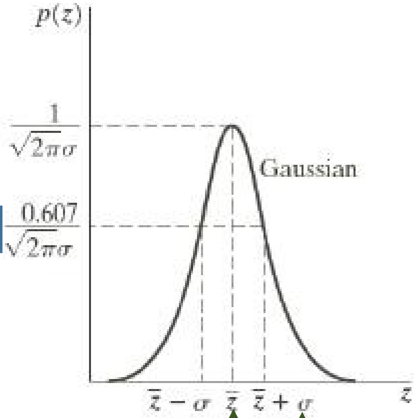
What does it mean to add noise of some distribution?

- Noise image of same size is produced. Its pixel values are obtained from the pdf



$$\begin{bmatrix} 3 & 2 & 3 & 3 & 3 \\ 3 & 140 & 141 & 141 & 2 \\ 2 & 139 & 220 & 140 & 3 \\ 2 & 139 & 139 & 141 & 2 \\ 2 & 2 & 2 & 2 & 3 \end{bmatrix}$$

$$\begin{bmatrix} 5 & 6 & 4 & 5 & 7 \\ 5 & 4 & 5 & 7 & 4 \\ 6 & 3 & 6 & 5 & 5 \\ 6 & 5 & 4 & 5 & 5 \\ 5 & 5 & 4 & 5 & 6 \end{bmatrix}$$



mean 5
variance 1

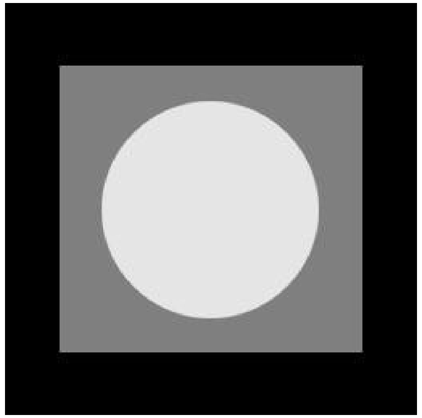
- Noise pixels are position invariant
- Values from the distribution can be at any random location (except periodic noise)
- Noise is uncorrelated with the image, i.e. noise pixels values do not depend on the image pixel values (except X-Ray, multiplicative noise etc.)

Very few pixels of this gray value

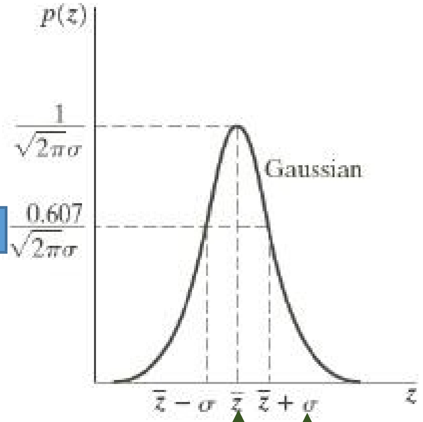
Many pixels of this gray value

What does it mean to add noise of some distribution?

- Noise image of same size is produced. Its pixel values are obtained from the pdf

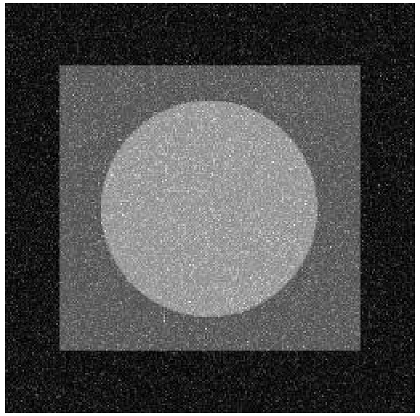


$$+\begin{bmatrix} 5 & 6 & 4 & 5 & 7 \\ 5 & 4 & 5 & 7 & 4 \\ 6 & 3 & 6 & 5 & 5 \\ 6 & 5 & 4 & 5 & 5 \\ 5 & 5 & 4 & 5 & 6 \end{bmatrix}$$



mean 5
variance 1

$$\begin{bmatrix} 3 & 2 & 3 & 3 & 3 \\ 3 & 140 & 141 & 141 & 2 \\ 2 & 139 & 220 & 140 & 3 \\ 2 & 139 & 139 & 141 & 2 \\ 2 & 2 & 2 & 2 & 3 \end{bmatrix}$$



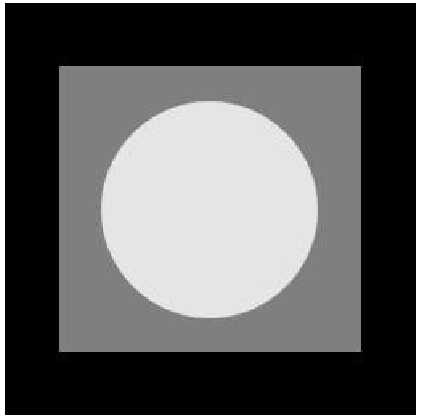
$$\begin{bmatrix} 8 & 8 & 7 & 8 & 10 \\ 8 & 144 & 146 & 148 & 6 \\ 8 & 142 & 226 & 145 & 8 \\ 8 & 144 & 143 & 146 & 7 \\ 7 & 7 & 6 & 7 & 9 \end{bmatrix}$$

Very few pixels of this gray value

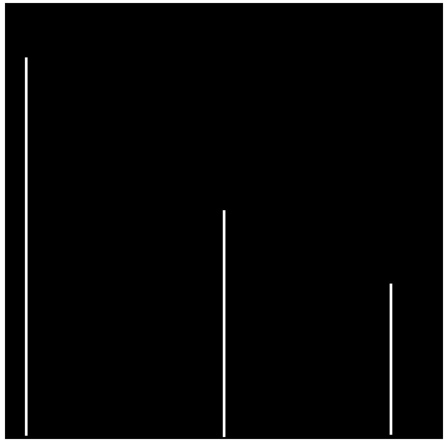
Many pixels of this gray value

Histogram of different noises and degraded images

Undegraded image



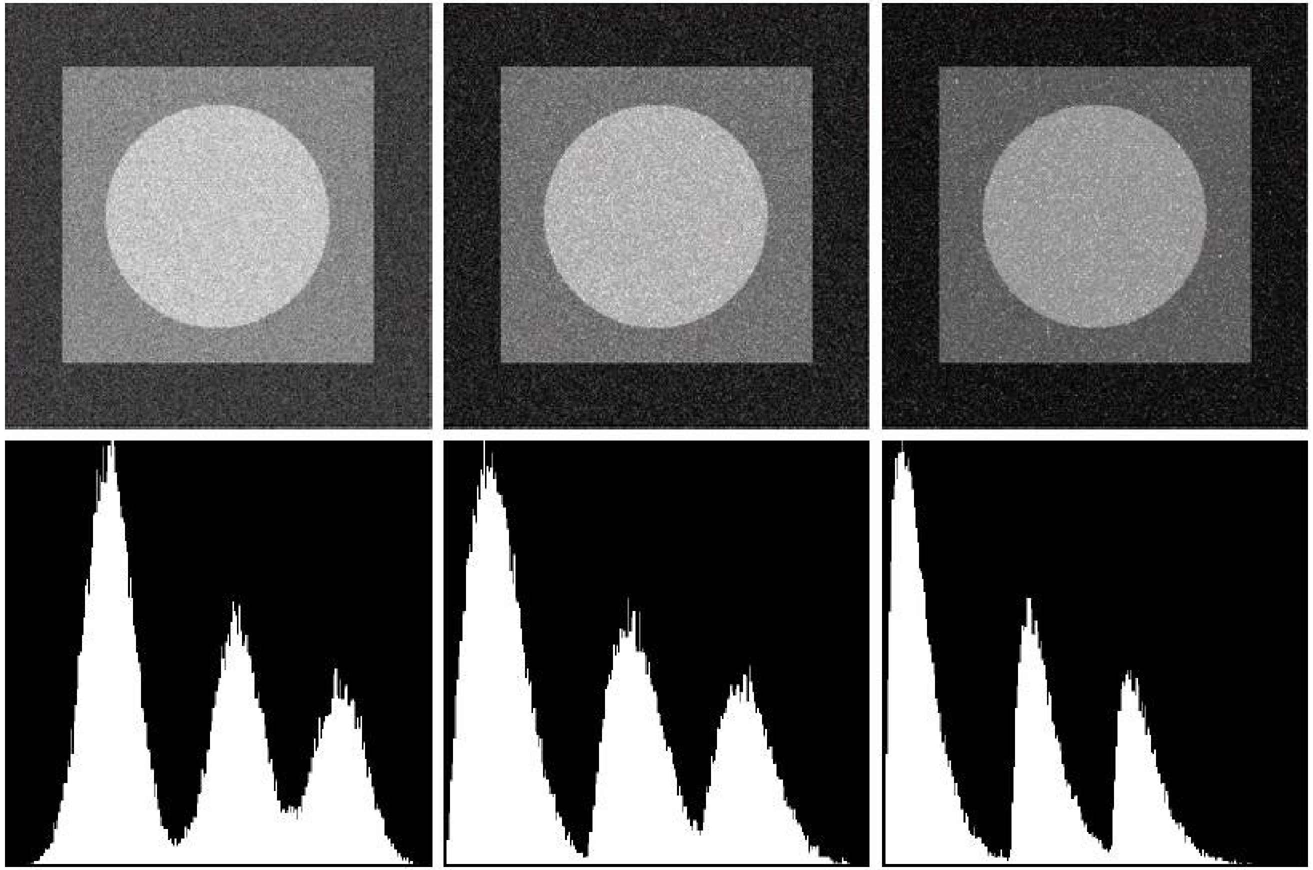
Histogram



Gaussian Noise

Rayleigh Noise

Gamma Noise



Histograms

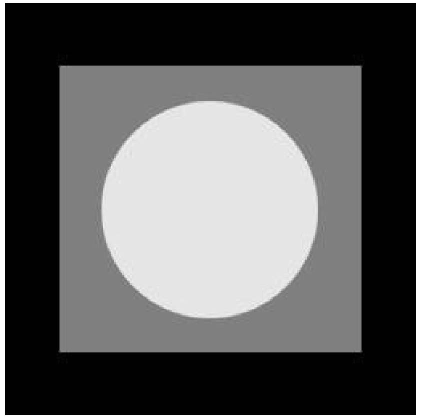
Histogram of different noises and degraded images

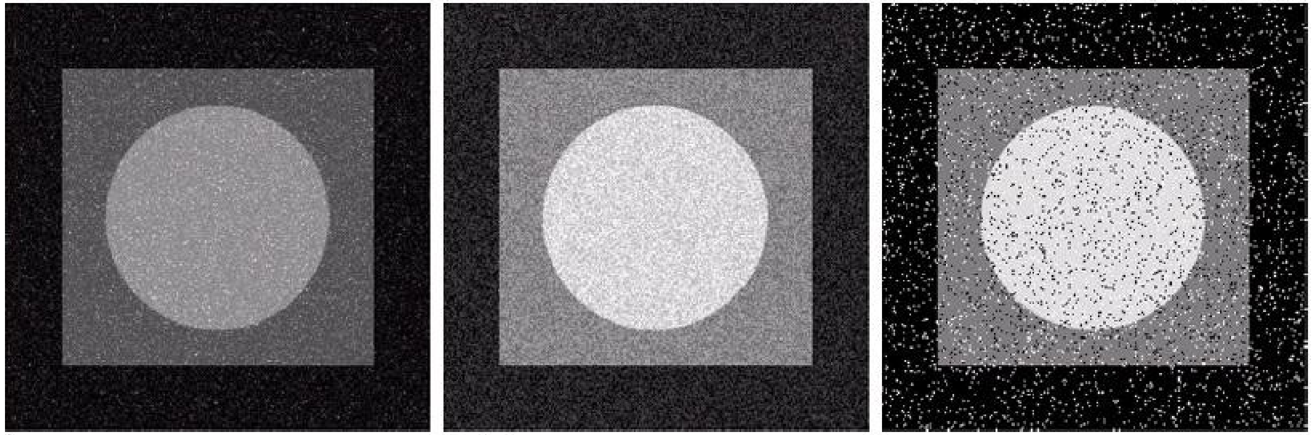
Undegraded image

Exponential Noise

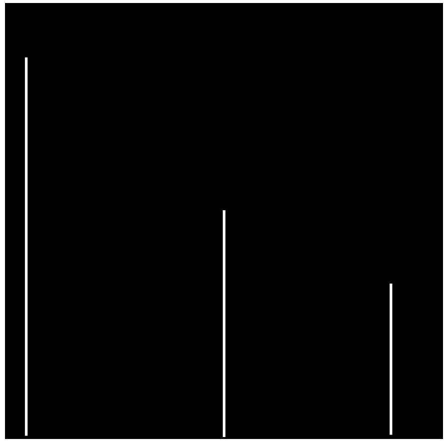
Uniform Noise

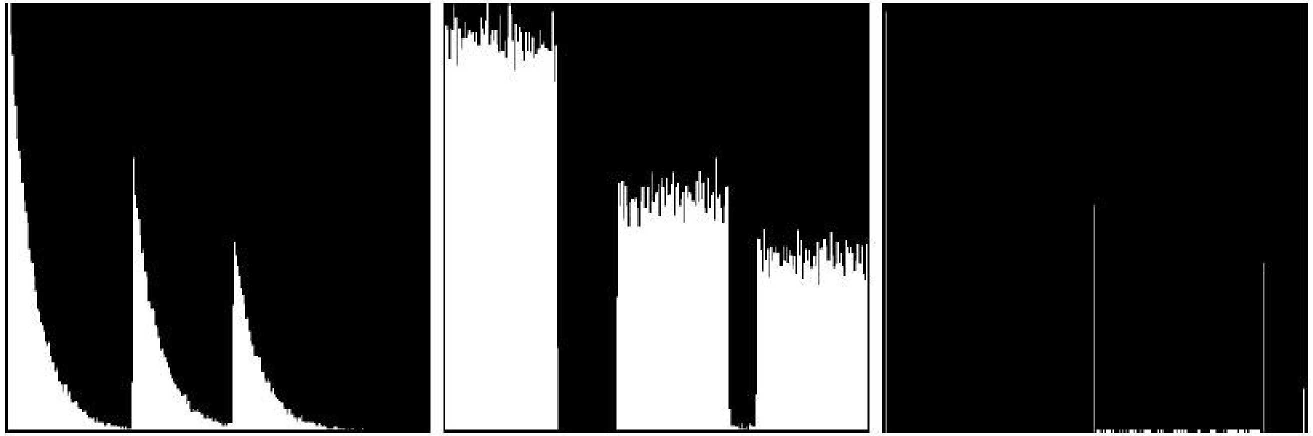
Salt & pepper Noise (impulse)





Histogram





Histograms

Discussions on these noises

- Gaussian noise: sensor noise due to poor illumination or high temperature
  - Commonly used

- Rayleigh distribution: characterizing noise in range imaging (depth image)

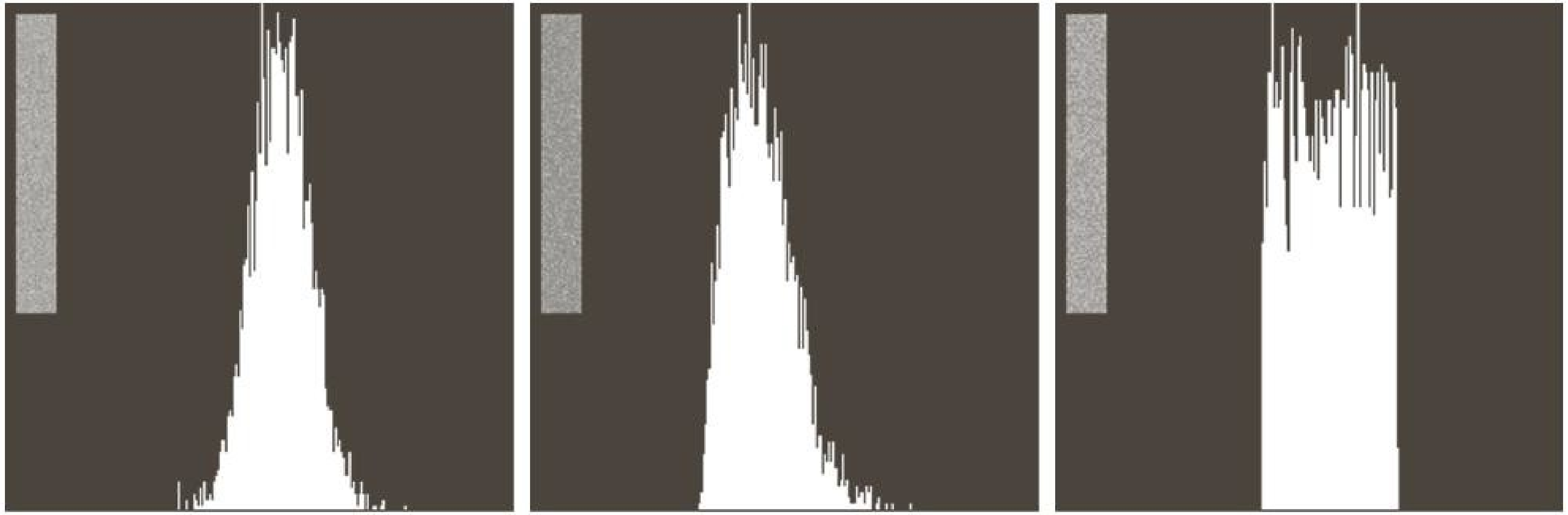
- Gamma and exponential distribution: laser imaging

- Impulse distribution: faulty switching resulting in quick transients
  - this noise amplitude is large compared to image signal value
  - noisy pixel values saturate at 255 (salt) or 0 (pepper)

Restoration of noisy images

- Without using the imaging system:
- Find constant valued patches in the image e.g. in background region
  - compute mean, variance etc. for estimating noise parameters

- Noise distribution in these patches is very similar to the overall noise distribution
  - position invariant noise



Restoration of noisy images

$$
\begin{tikzcd}
f(x,y) \arrow[r] & \bigoplus \arrow[r] & g(x,y) \arrow[r] & \boxed{\text{NR}} \arrow[r] & \hat{f}(x,y) \\
& \eta(x,y) \arrow[u] & & &
\end{tikzcd}
$$

Without estimating noise

$\boxed{m \text{ x } n \text{ window}}$

Arithmatic mean filter

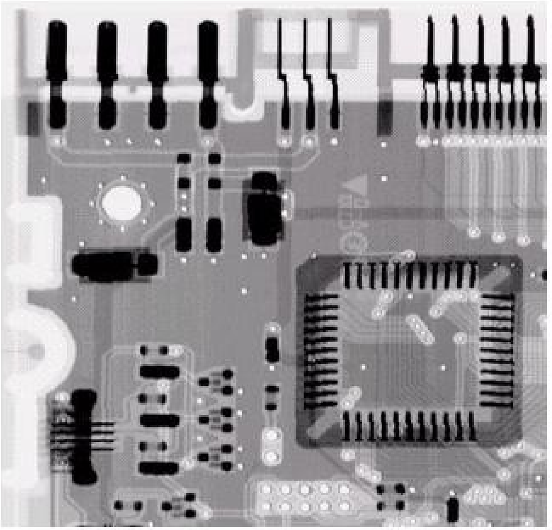
$$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s,t)$$

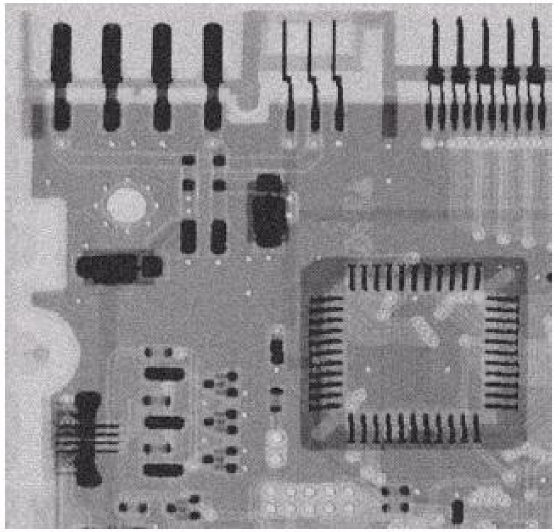
Geometric mean filter

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

Arithmatic and geometric mean filtering

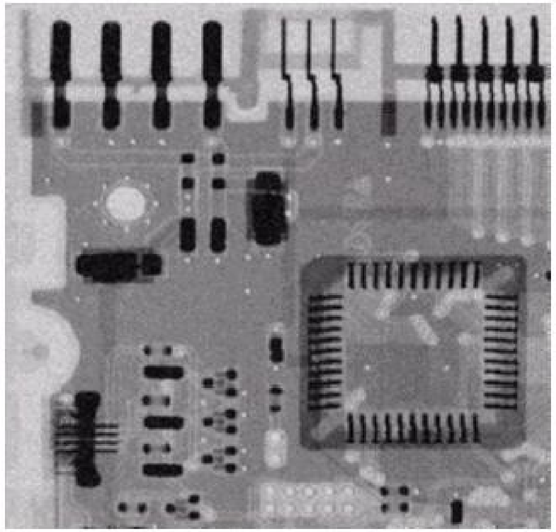
Undegraded image

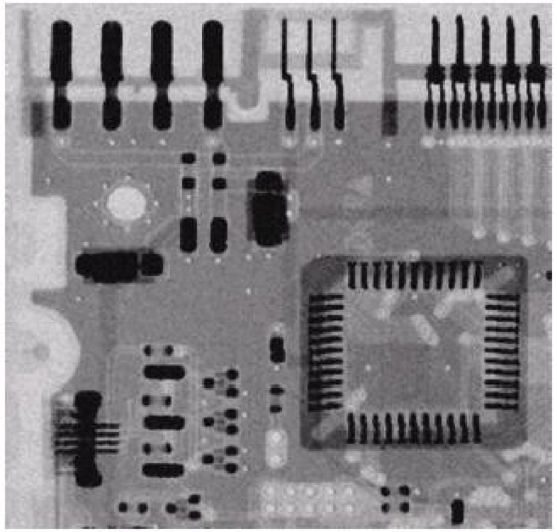




Gaussian noise added

3x3 Arithmatic mean filtered image





3x3 Geometric mean filtered image (Slightly better result, less blurring, less detail lost, sharper image)

Harmonic mean filtering

Without estimating noise

$$x = \frac{1}{1/x}$$

harmonic mean filter

m x n window

$$x = \frac{x^{Q+1}}{x^Q}$$

Contraharmonic mean filter

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

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

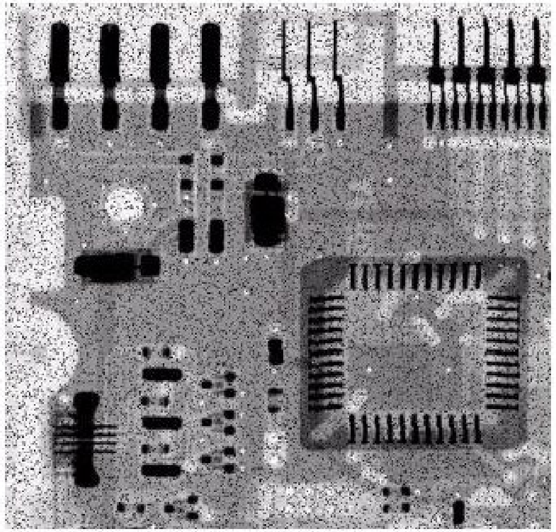
Removes salt noise,
Gaussian nose

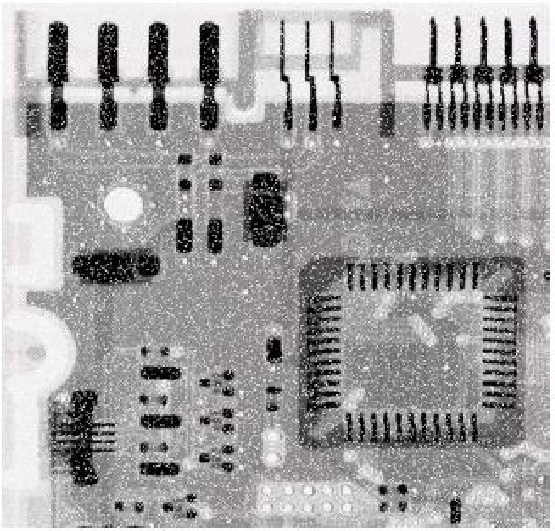
Q = 0: mean filter
Q = -1: harmonic filter

Removes salt noise when Q<0
Removes pepper noise when Q>0

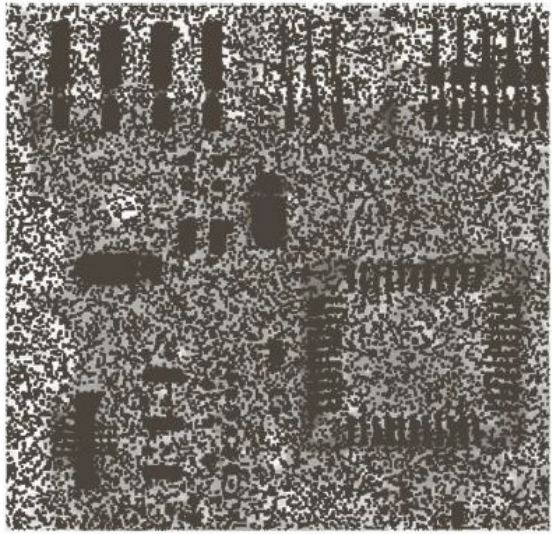
Contraharmonic mean filtering

Pepper noise added





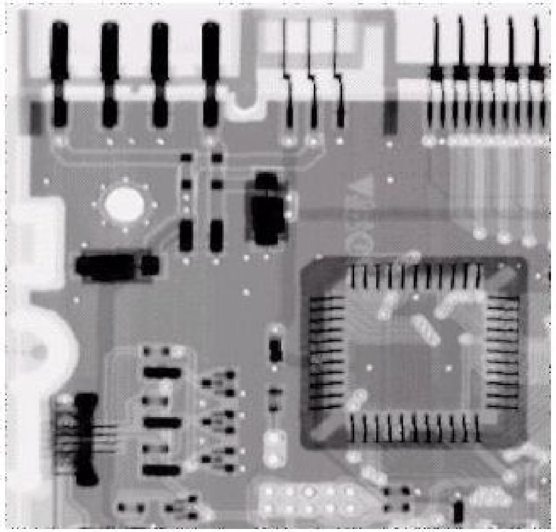
Salt noise added

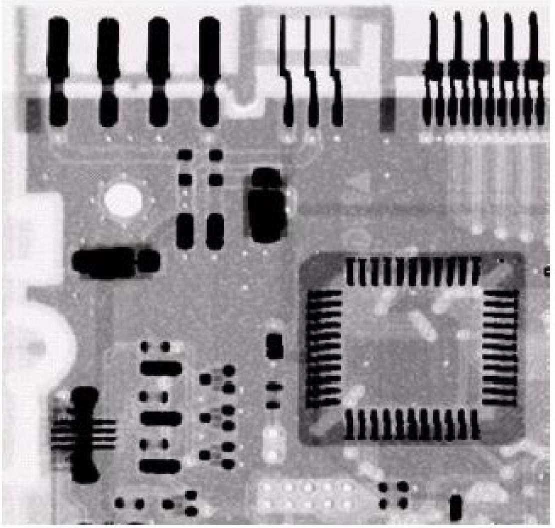


Used Q = -1.5

Pepper noise removed Q = 1.5

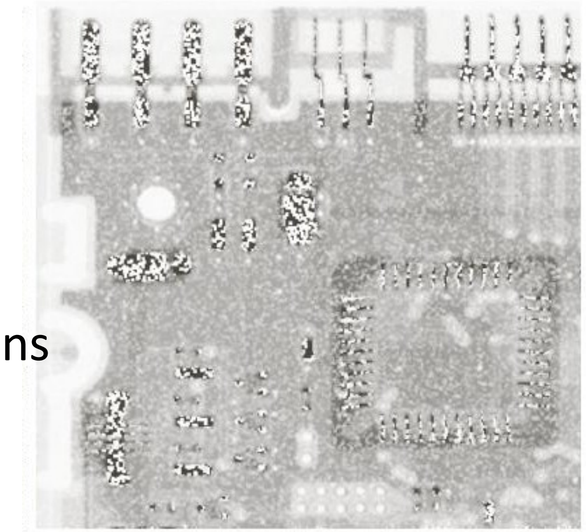
Dark regions thinned





Salt noise removed Q = -1.5

Dark regions intact



Used Q = 1.5

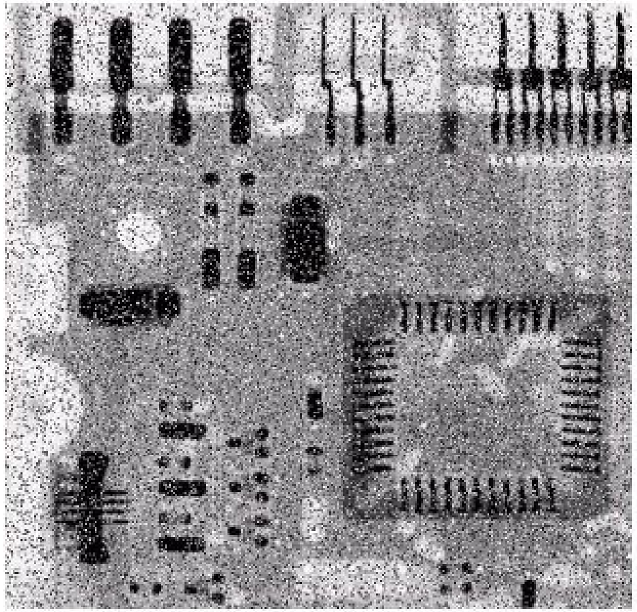
Order statistics filters

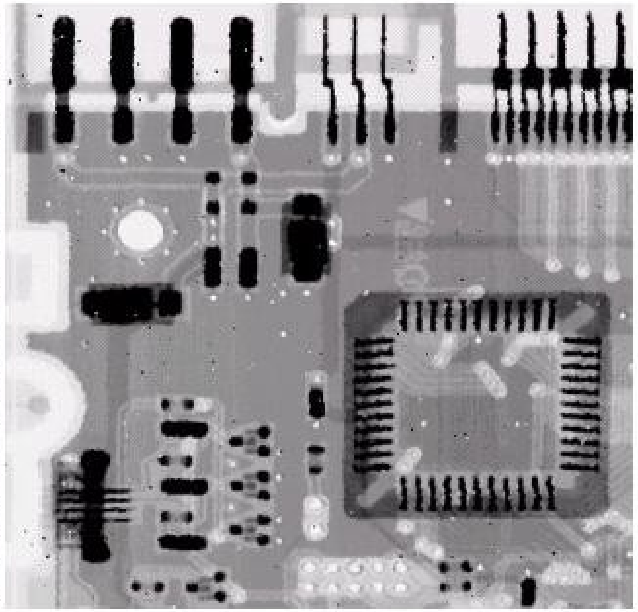
Median filter:

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

$$\begin{bmatrix} 100 & 115 & 120 \\ 99 & 11 & 122 \\ 105 & 117 & 112 \end{bmatrix} -> \begin{bmatrix} 100 & 115 & 120 \\ 99 & 112 & 122 \\ 105 & 117 & 112 \end{bmatrix}$$

Salt-pepper noise added





3x3 Median filtered image

Order statistics filters

$$\begin{bmatrix} 100 & 115 & 120 \\ 99 & 11 & 122 \\ 105 & 117 & 112 \end{bmatrix} -> \begin{bmatrix} 100 & 115 & 120 \\ 99 & 122 & 122 \\ 105 & 117 & 112 \end{bmatrix}$$

Max filter:

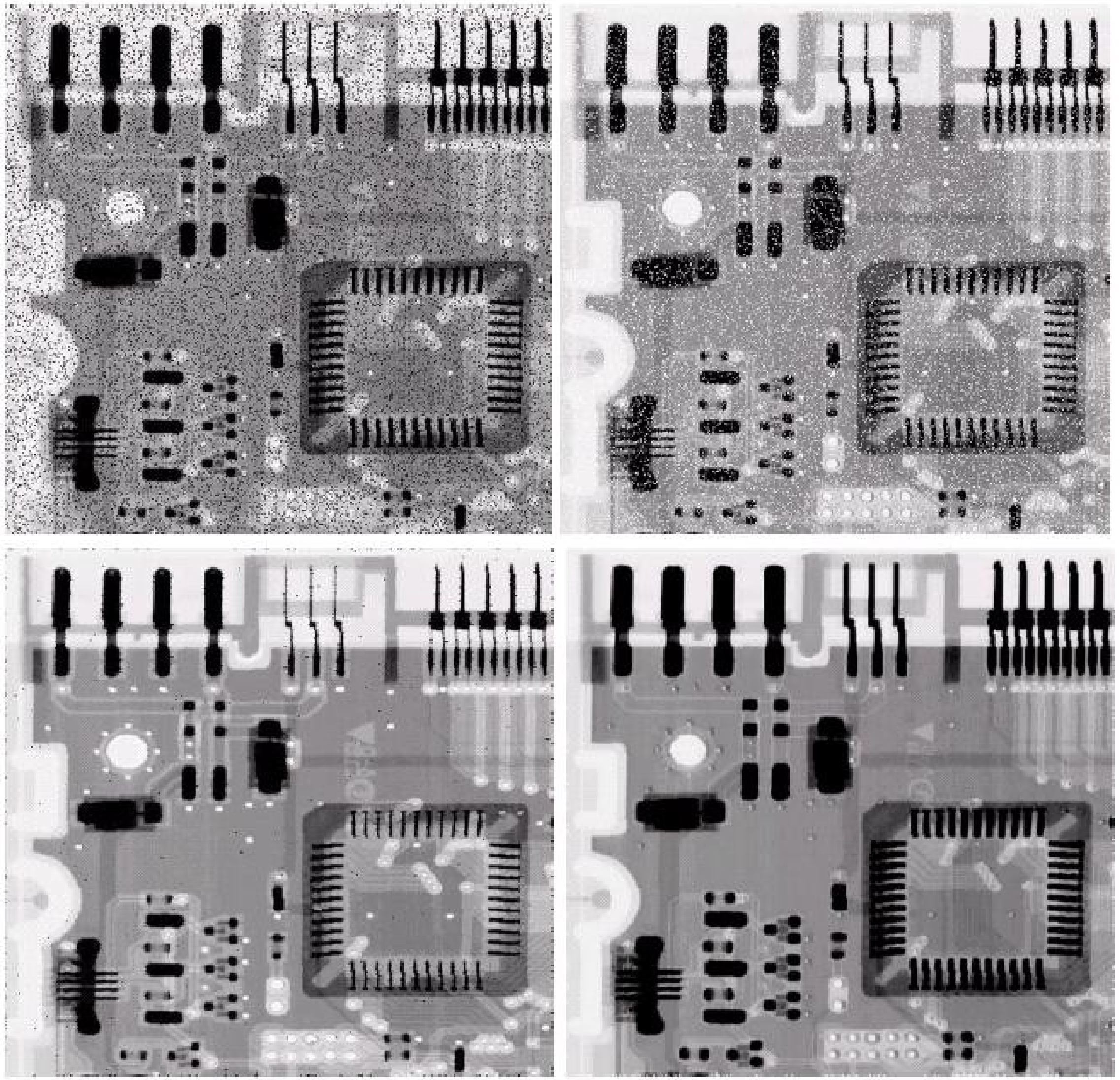
$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}} \{g(s,t)\}$$

Min filter:

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

$$\begin{bmatrix} 100 & 115 & 120 \\ 99 & 254 & 122 \\ 105 & 117 & 112 \end{bmatrix} -> \begin{bmatrix} 100 & 115 & 120 \\ 99 & 99 & 122 \\ 105 & 117 & 112 \end{bmatrix}$$

Pepper noise added | Salt noise added



Max filtered image | Min filtered image

Order statistics filters

- Midpoint filter:

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

computes the midpoint between the minimum and maximum values

works well for Gaussian or uniform noise

- Alpha-trimmed mean filter:

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

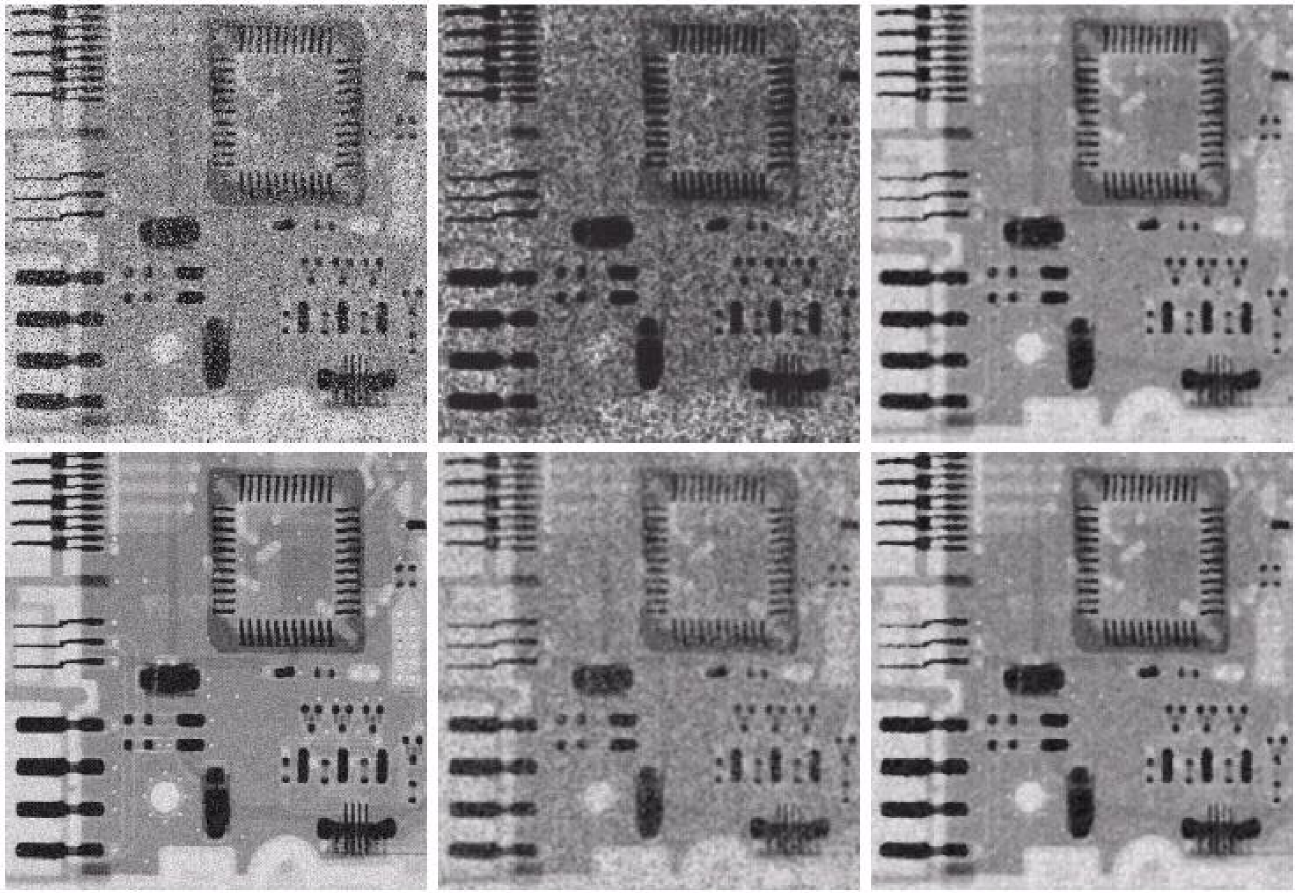
deletes the lowest $d/2$ and highest $d/2$ values and computes the mean of the remaining values

d=0: mean filter
d=mn-1: median filter.
So, works well for combination of Gaussian and salt-pepper noise

Comparing mean filtering with median filtering

5x5 Geometric
mean filtered image

Uniform + salt-
pepper noise added



5x5 Alpha-trimmed
mean filtered image
(d=5)

Uniform noise added

5x5 Median
filtered image

5x5 Arithmatic
mean filtered image

Adaptive median filters

- Goal: remove salt-pepper noise, provide smoothing of other noise and reduce distortion such as thinning or thickening

- Standard median filtering:
    - update each pixel by the median value of the window
    - do not consider local characteristics

$$\begin{bmatrix} 100 & 115 & 120 \\ 99 & 112 & 122 \\ 105 & 117 & 112 \end{bmatrix}$$

no need
to update

Adaptive median filters

- Goal: remove salt-pepper noise, provide smoothing of other noise and reduce distortion such as thinning or thickening

$$\begin{bmatrix} 100 & 115 & 120 \\ 99 & 112 & 122 \\ 105 & 117 & 112 \end{bmatrix}$$ no need to update

- Standard median filtering:
  - update each pixel by the median value of the window
  - do not consider local characteristics

$$\begin{bmatrix} 100 & 0 & 0 \\ 0 & 0 & 0 \\ 105 & 0 & 112 \end{bmatrix}$$ median=0

- Adaptive filtering:
  - update only if it is a salt (very high value) or pepper (very small value) noise pixel
  - if it is a noise pixel and the median pixel value is also the same as the noise pixel value
    - increase window size so that the median pixel value is not very high or very small

Adaptive median filters

Stage A:

$A1 = z_{med} - z_{min}$

$A2 = z_{med} - z_{max}$

If $A1 > 0$ and $A2 < 0$, Go to stage B

Else increase the window size

If window size $\leq S_{max}$ repeat stage A

Else output z_{med}

z_{med} not noise

z_{med} is noise; so increase window size

$$\begin{bmatrix} 100 & 0 & 0 \\ 0 & 0 & 0 \\ 105 & 0 & 112 \end{bmatrix}$$

median=0

$$\begin{bmatrix} 1 & 107 & 103 & 102 & 104 \\ 102 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 106 & 105 & 0 & 112 & 110 \\ 107 & 107 & 108 & 111 & 109 \end{bmatrix}$$

median =102

Stage B:

- Stage A tries to determine whether z_{med} is an impulse or not

Adaptive median filters

Stage A:

$A1 = z_{med} - z_{min}$

$A2 = z_{med} - z_{max}$

If $A1 > 0$ and $A2 < 0$, Go to stage B

Else increase the window size

If window size $\leq S_{max}$ repeat stage A

Else output z_{med}

z_{med} not noise

z_{med} is noise; so increase window size

$$\begin{bmatrix} 100 & 0 & 0 \\ 0 & 0 & 0 \\ 105 & 0 & 112 \end{bmatrix}$$

median=0

$$\begin{bmatrix} 1 & 107 & 103 & 102 & 104 \\ 102 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 106 & 105 & 0 & 112 & 110 \\ 107 & 107 & 108 & 111 & 109 \end{bmatrix}$$

median =102

Stage B:

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

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

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

Else output z_{med}

$$\begin{bmatrix} 100 & 0 & 0 \\ 0 & 5 & 0 \\ 105 & 0 & 112 \end{bmatrix}$$

center not 0

Update if z_{xy} is noise

Do not update if z_{xy} not noise

- Stage A tries to determine whether z_{med} is an impulse or not
- If it is no impulse stage B tries to estimate whether the center of the window z_{xy} is an impulse

Adaptive mean filters

- Reconstruction using arithmatic mean filtering $\hat{f}(x,y) = m_L$

$$\hat{f}(x,y) = g(x,y) - \underbrace{(g(x,y) - m_L)}_{\text{Deviation}}$$

- Adaptive mean filtering: $\hat{f}(x,y) = g(x,y) - \underbrace{\frac{\sigma_\eta^2}{\sigma_L^2}(g(x,y) - m_L)}_{\text{Weighted deviation}}$

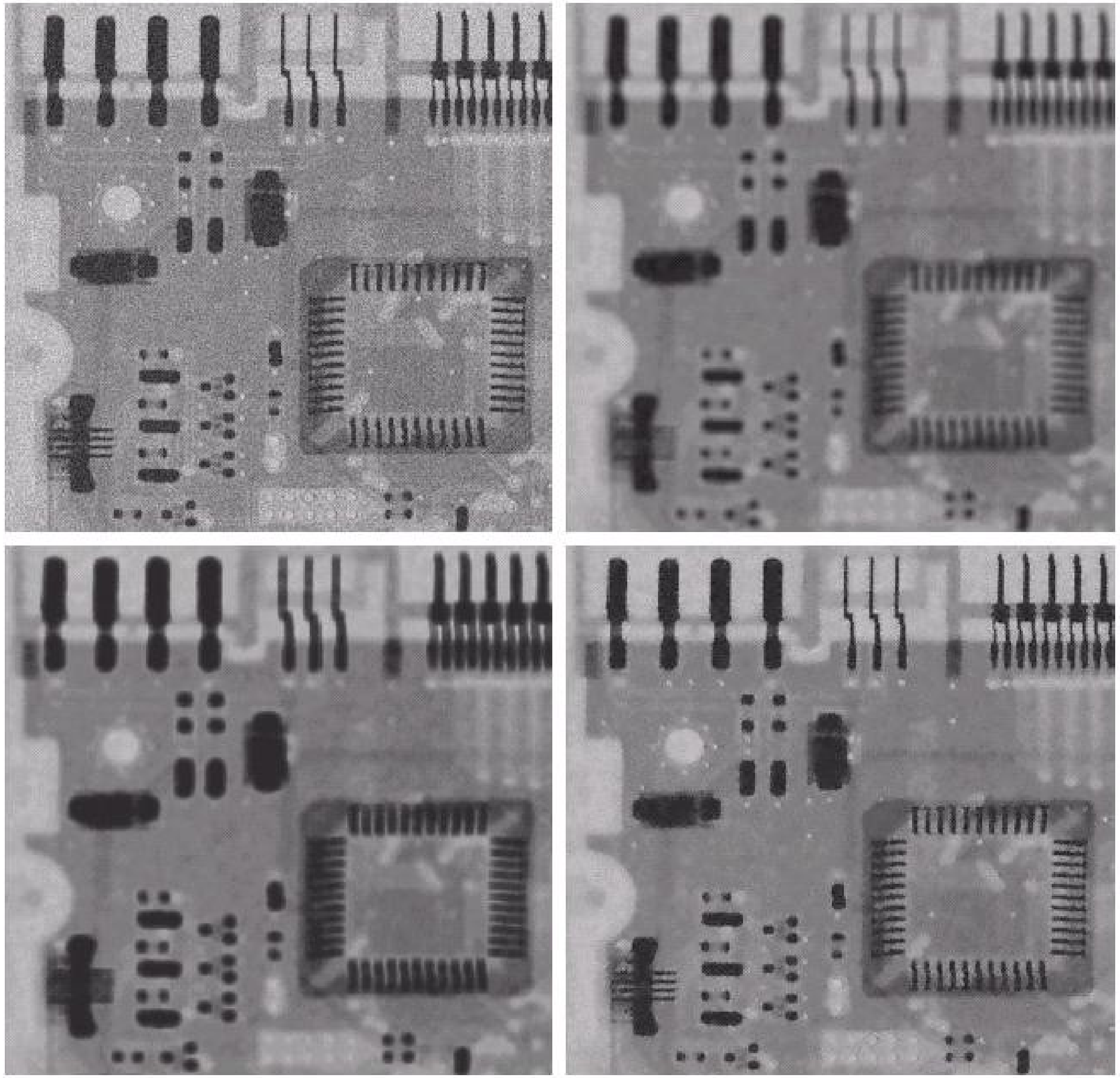
- If noise variance $\sigma_\eta^2 = 0$, no degradation i.e. $\hat{f}(x,y) = g(x,y)$

- If $\sigma_\eta^2 < \sigma_L^2$, high local variance due to presence of edges, so small deviation from $g(x,y)$
- Avoids unnecessary smoothing of edges
- If $\sigma_\eta^2 \geq \sigma_L^2$, truncate $\frac{\sigma_\eta^2}{\sigma_L^2} = 1;$ reduces to arithmatic mean filtering $\hat{f}(x,y) = m_L$
- Noise variance needs to be known

Adaptive mean filtering

Gaussian noise added

7x7 Arithmatic mean filtered image

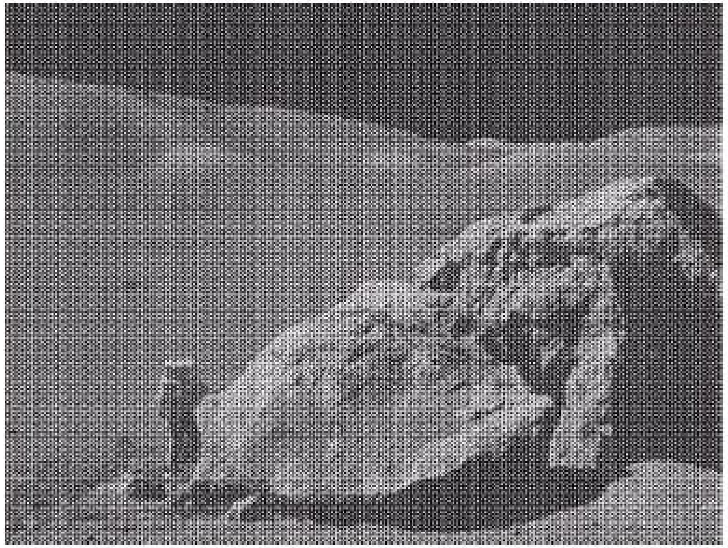


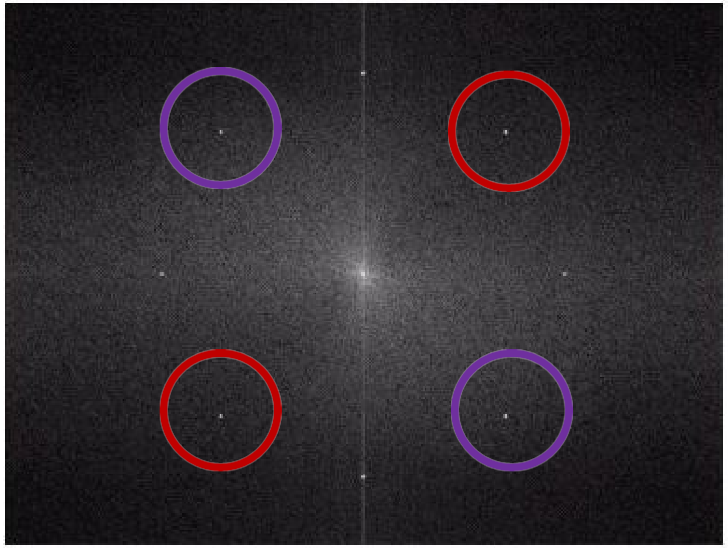
7x7 Geometric mean filtered image

7x7 Adaptive mean filtered image

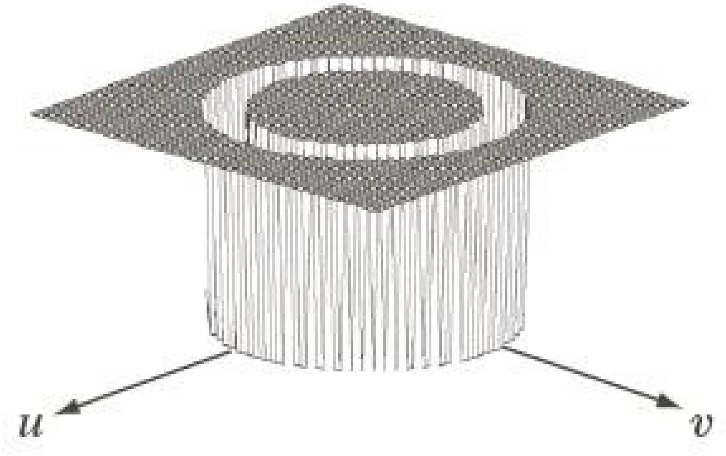
Periodic noise

- Fourier transform sinusoid is a pair of impulses
- If frequency is known, noise can be removed using frequency domain filtering
  - performs better than spatial filtering
- Bandreject filter
  - removes the band of noise frequencies
- Bandpass filter
  - produces the band of noise frequencies

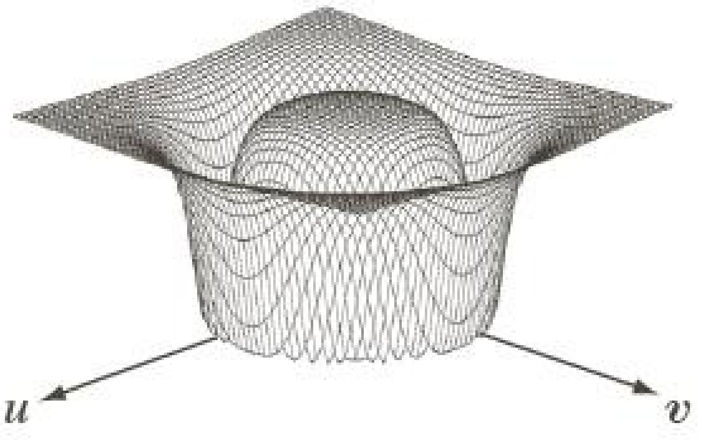




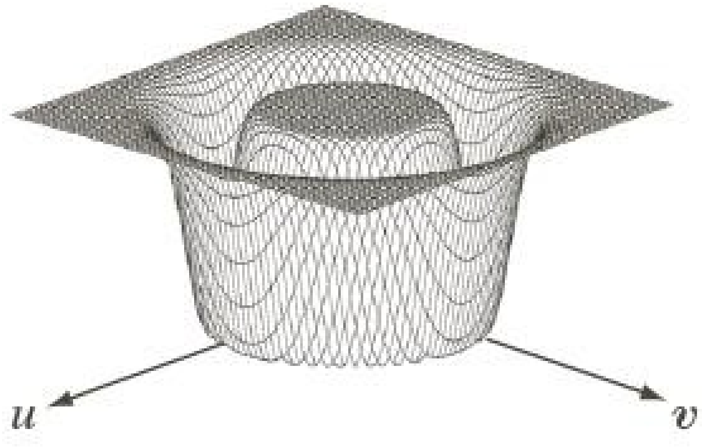
Bandreject filters



Ideal



Butterworth



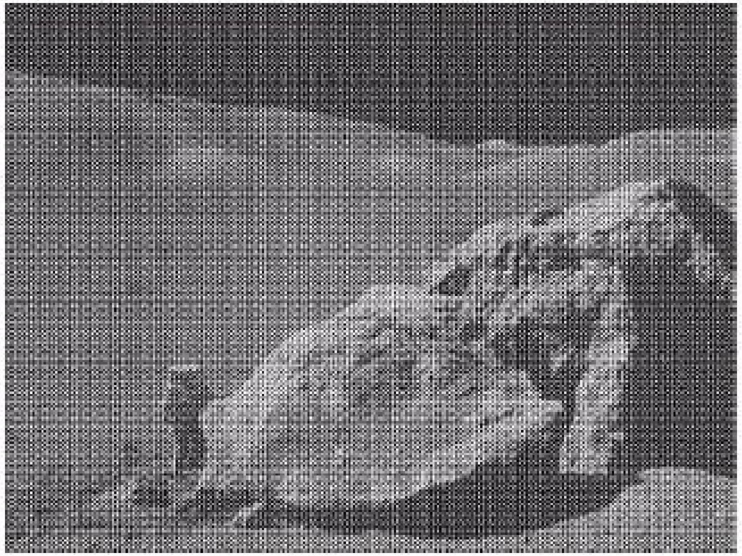
Gaussian

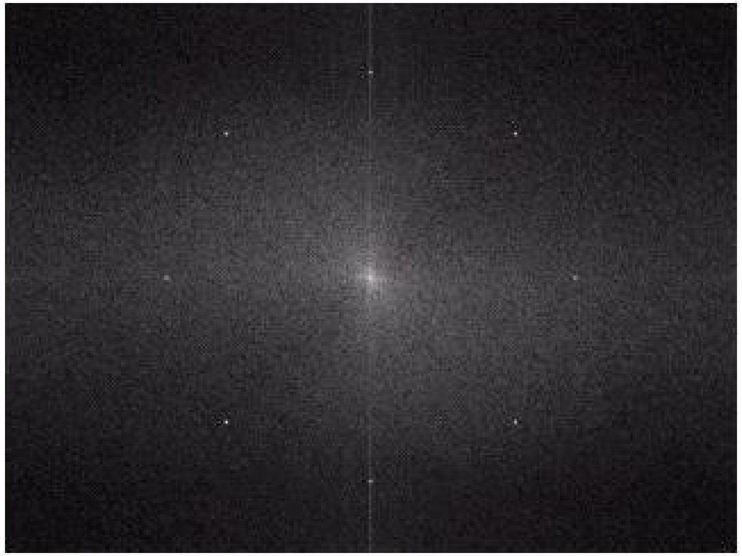
$$H_{BR}^{\text{ideal}}(u,v) = \begin{cases} 1 & \text{if } D(u,v) < D_0 - \dfrac{W}{2} \\ 0 & \text{if } D_0 - \dfrac{W}{2} \le D(u,v) \le D_0 + \dfrac{W}{2} \\ 1 & \text{if } D(u,v) > D_0 + \dfrac{W}{2} \end{cases}$$

Bandreject and bandpass filtering

Spectrum

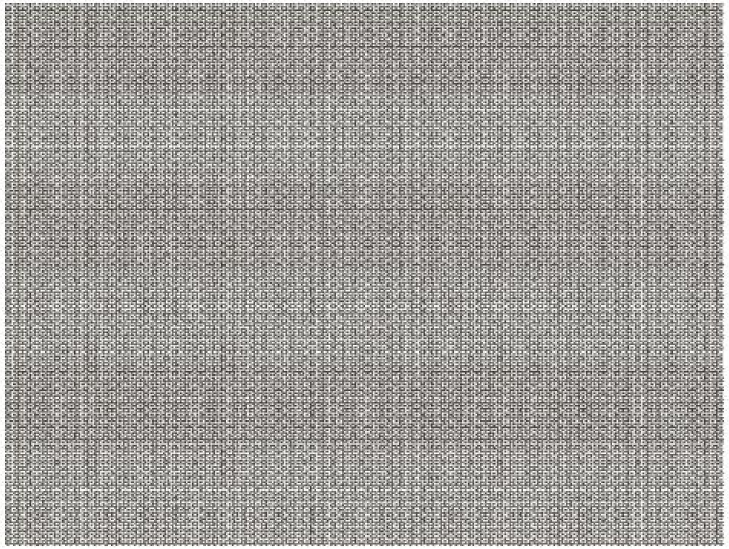
Sinusoidal noise added



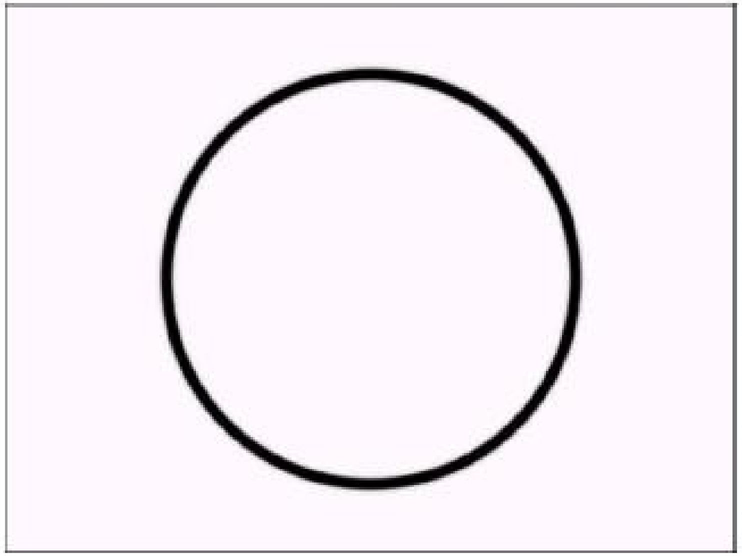


$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

Bandpass filtered image



Butterworth Bandreject filter



Bandreject filtered image

Restoration using spatial domain filtering

- 1D convolution (consider a row from the image):

$$g(m) = f(m) * h(m)$$
$$g(m) = \sum_n h(m-n) f(n)$$

$$g(1) = h(1) f(0) + h(0) f(1) + h(-1) f(2) + \ldots$$

$$g(0) = \sum_n h(-n) f(n)$$

$$g(2) = h(2) f(0) + h(1) f(1) + h(0) f(2) + \ldots$$

$$g(0) = h(0) f(0) + h(-1) f(1) + h(-2) f(2) + \ldots$$

$$\begin{bmatrix} g(0) \\ g(1) \\ g(2) \\ . \\ g(n-1) \end{bmatrix} = \begin{bmatrix} h(0) & h(-1) & h(-2) & . & h(-n+1) \\ h(1) & h(0) & h(-1) & . & h(-n+2) \\ h(2) & h(1) & h(0) & . & h(-n+3) \\ . & . & . & . & . \\ h(n-1) & h(n-2) & h(n-3) & . & h(0) \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ . \\ f(n-1) \end{bmatrix}$$

Restoration using spatial domain filtering

- 1D convolution (consider a row from the image):

$$g(m) = f(m) * h(m)$$

$$g(m) = \sum_n h(m-n) f(n)$$

$$\begin{bmatrix} g(0) \\ g(1) \\ g(2) \\ . \\ g(n-1) \end{bmatrix} = \begin{bmatrix} h(0) & h(-1) & h(-2) & . & h(-n+1) \\ h(1) & h(0) & h(-1) & . & h(-n+2) \\ h(2) & h(1) & h(0) & . & h(-n+3) \\ . & . & . & . & . \\ h(n-1) & h(n-2) & h(n-3) & . & h(0) \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ . \\ f(n-1) \end{bmatrix}$$

H is Toeplitz matrix

Linear convolution

- Matrix multiplication representation $\mathbf{g} = H\,\mathbf{f}$, where $\mathbf{g}, \mathbf{f} \in \Re^N, H \in \Re^{N \times N}$

- Not to be confused with $G(u,v) = H(u,v)\, f(\mathrm{u},\mathrm{v})$

Restoration using spatial domain filtering

- 1D convolution (consider a row from the image): $g(m) = f(m) * h(m)$

$$g(m) = \sum_n h(m-n) f(n)$$

$\boxed{h(0), h(1), \ldots \text{ scalars}}$

$$\begin{bmatrix} g(0) \\ g(1) \\ g(2) \\ . \\ g(n-1) \end{bmatrix} = \begin{bmatrix} h(0) & h(n-1) & h(n-2) & . & h(1) \\ h(1) & h(0) & h(n-1) & . & h(2) \\ h(2) & h(1) & h(0) & . & h(3) \\ . & . & . & . & . \\ h(n-1) & h(n-2) & h(n-3) & . & h(0) \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ . \\ f(n-1) \end{bmatrix}$$

If h is periodic,
H is circulant matrix

Circular convolution

- Matrix multiplication representation $\mathbf{g} = H\,\mathbf{f}$, where $\mathbf{g}, \mathbf{f} \in \Re^N, H \in \Re^{N \times N}$

- Not to be confused with $G(u,v) = H(u,v)\, f(\mathrm{u},\mathrm{v})$

Restoration using spatial domain filtering

- 2D convolution (consider the entire image): $g(m1,m2) = f(m1,m2) * h(m1,m2)$

$$g(m1,m2) = \sum_{n1}\sum_{n2} h(m1-n1, m2-n2) f(n1,n2)$$

$\boxed{H(0), H(1), \text{... matrices}}$

$$\begin{bmatrix} g(0) \\ g(1) \\ g(2) \\ . \\ g(n-1) \end{bmatrix} = \begin{bmatrix} H(0) & H(n-1) & H(n-2) & . & H(1) \\ H(1) & H(0) & H(n-1) & . & H(2) \\ H(2) & H(1) & H(0) & . & H(3) \\ . & . & . & . & . \\ H(n-1) & H(n-2) & H(n-3) & . & H(0) \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ . \\ f(n-1) \end{bmatrix}$$

If h is periodic,
$\mathbf{H}$ is block circulant matrix

Circular convolution

- Matrix multiplication representation $\mathbf{g} = \mathbf{H}\,\mathbf{f}$, where $\mathbf{g}, \mathbf{f} \in \Re^{N^2}, \mathbf{H} \in \Re^{N^2 \times N^2}$

- Not to be confused with $G(u,v) = H(u,v)\, f(\mathrm{u},\mathrm{v})$

For Image Restoration Topics, material used from

Digital Image Processing by S. Jayaraman

Digital Image Processing by S. Sridhar