Image Processing
INT3404 20
Week 10:
Noise and restoration

Lecturer: Nguyen Thi Ngoc Diep, Ph.D.

Email: ngocdiep@vnu.edu.vn

Slide & code: https://github.com/chupibk/INT3404_20

1

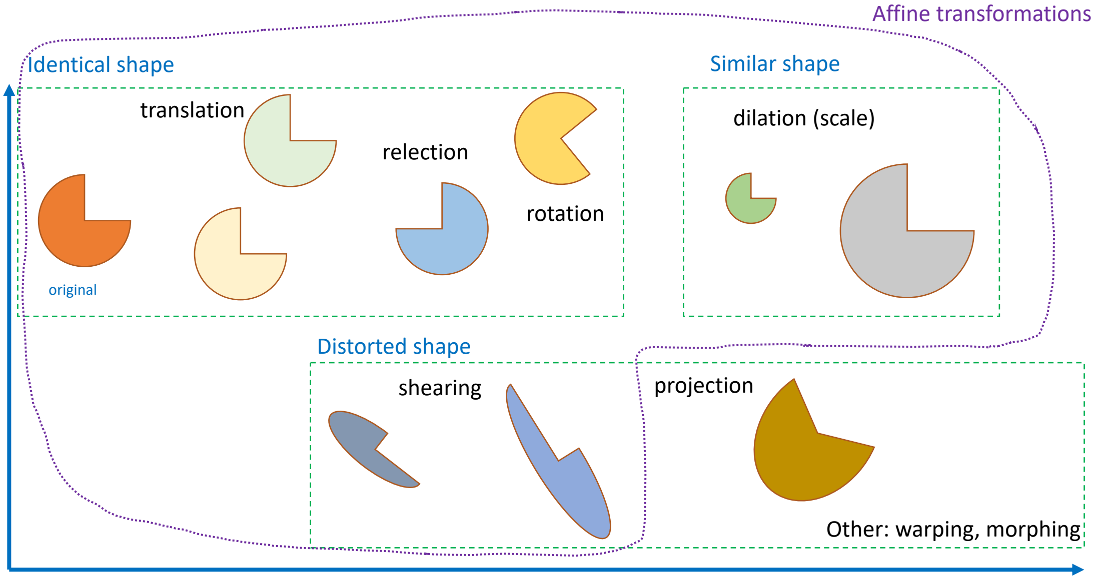
Schedule

Week	Content	Homework
1	Introduction	Set up environments: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Digital image – Point operations Contrast adjust – Combining images	HW1: adjust gamma to find the best contrast
3	Histogram - Histogram equalization – Histogram-based image classification	Self-study
4	Spatial filtering - Template matching	Self-study
5	Feature extraction Edge, Line, and Texture	Self-study
6	Morphological operations	HW2: Barcode detection → Require submission as mid-term test
7	Filtering in the Frequency domain Announcement of Final project topics	Final project registration
8	Color image processing	HW3: Conversion between color spaces, color image segmentation
9	Geometric transformations	Self-study
10	Noise and restoration	Self-study
11	Compression	Self-study
12	Final project presentation	Self-study
13	Final project presentation Class summarization	Self-study

2

2

Recall week9: Geometric transformations



3

Two basic operations of geometric transformation

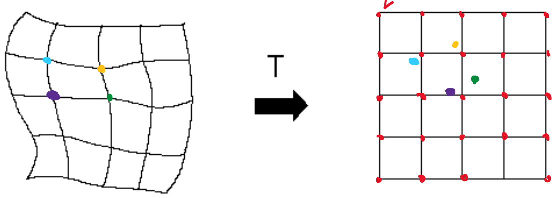
Forward \ Backward mapping

1. Spatial transformation of coordinates

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \end{bmatrix}$$

2. Intensity interpolation that assigns intensity values to the spatially transformed pixels

?



4

General tranformation matrix

Translation vector

Rotation matrix

$$\begin{bmatrix} a1 & a2 & b1 \\ a3 & a4 & b2 \\ c1 & c2 & 1 \end{bmatrix}$$

Projection vector

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \cong \begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} a1 & a2 & b1 \\ a3 & a4 & b2 \\ c1 & c2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Also referred to as “homography matrix”

More on projective transformation: https://mc.ai/part-ii-projective-transformations-in-2d/

5

Noise & restoration

6

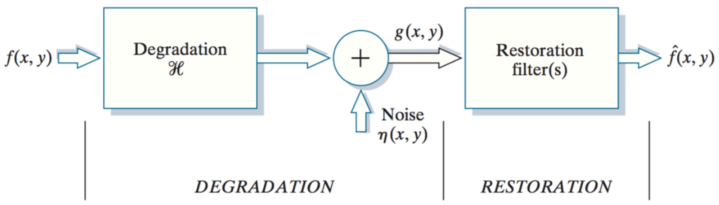
Image restoration

- To recover an image that has been degraded by using a priori knowledge of the degradation phenomenon
- → modeling the degradation and applying the inverse process in order to recover the original image
- Applying in both spatial and frequency domains

7

Image degradation/restoration process

Linear, position-invariant



Spatial domain $g(x,y) = (h \star f)(x,y) + \eta(x,y)$

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

h(x, y) : degradation function
n(x, y) : additive noise term

Restoration seeks to find filters that apply the process in reverse (deconvolution filters)

8

Where noise comes from?

$$g(x,y) = (h \star f)(x,y) + \boxed{\eta(x,y)}$$

- During image acquisition and/or transmission
- Imaging sensors:
  - Environmental factors, e.g.: light levels, sensor temperature
  - Quality of the sensing elements
- Transmission
  - For example, corrupted by lightning or other atmospheric disturbance when using a wireless network

9

Noise models

- Consider noise as random variables, characterized by a probability density function (PDF)
- PDFs that are useful for modeling a broad range of noise corruption situations found in practice:
  - Gaussian: electronic circuit noise and sensor noise caused by poor illumination and/or high temperature
  - Rayleigh: range imaging
  - Exponential and gamma: laser imaging
  - Impulse noise: quick transients, such as faulty switching
  - Uniform density: used in simulations

10

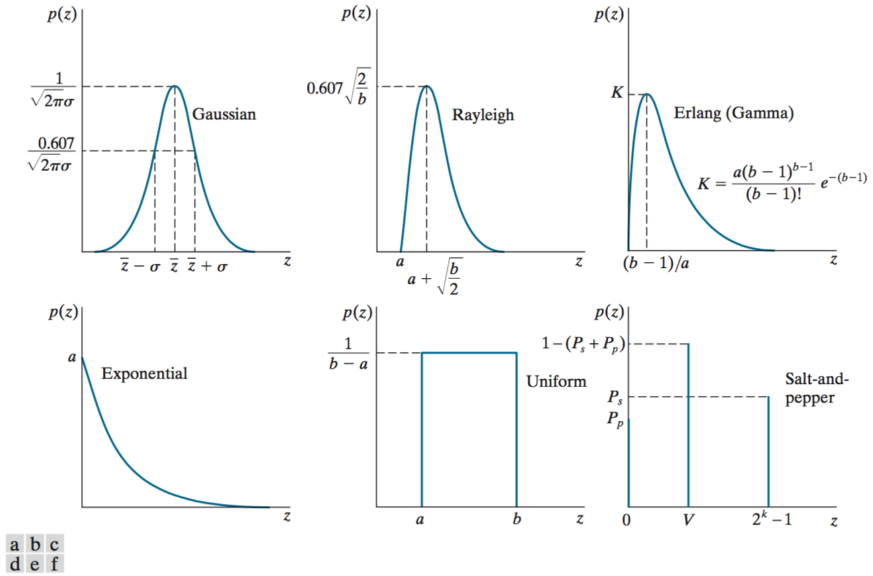


FIGURE 5.2 Some important probability density functions.

11

Gaussian noise

The PDF of a *Gaussian* random variable, z, is defined by the following familiar expression:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(z-\bar{z})^2}{2\sigma^2}} \qquad -\infty < z < \infty \tag{5-3}$$

where z represents intensity, $\bar{z}$ is the mean (average) value of z, and σ is its standard deviation.

12

Rayleigh noise

The PDF of *Rayleigh* noise is given by

$$p(z) = \begin{cases} \frac{2}{b}(z - a)e^{-(z-a)^2/b} & z \geq a \\ 0 & z < a \end{cases} \qquad (5\text{-}4)$$

The mean and variance of z when this random variable is characterized by a Rayleigh PDF are

$$\bar{z} = a + \sqrt{\pi b/4} \qquad (5\text{-}5)$$

and

$$\sigma^2 = \frac{b(4 - \pi)}{4} \qquad (5\text{-}6)$$

13

Erlang (gamma) noise

The PDF of Erlang noise is

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!}e^{-az} & z \geq 0 \\ 0 & z < 0 \end{cases} \qquad (5\text{-}7)$$

where the parameters are such that $a > b$, b is a positive integer, and "!" indicates factorial. The mean and variance of z are

$$\bar{z} = \frac{b}{a} \qquad (5\text{-}8)$$

and

$$\sigma^2 = \frac{b}{a^2} \qquad (5\text{-}9)$$

14

Exponential noise

The PDF of *exponential* noise is given by

$$p(z) = \begin{cases} ae^{-az} & z \geq 0 \\ 0 & z < 0 \end{cases} \tag{5-10}$$

where $a > 0$. The mean and variance of z are

$$\bar{z} = \frac{1}{a} \tag{5-11}$$

and

$$\sigma^2 = \frac{1}{a^2} \tag{5-12}$$

Note that this PDF is a special case of the Erlang PDF with $b = 1$. Figure 5.2(d) shows a plot of the exponential density function.

15

Uniform noise

The PDF of *uniform* noise is

$$p(z) = \begin{cases} \dfrac{1}{b - a} & a \leq z \leq b \end{cases} \tag{5-13}$$

The mean and variance of z are

$$\bar{z} = \frac{a + b}{2} \tag{5-14}$$

and

$$\sigma^2 = \frac{(b - a)^2}{12} \tag{5-15}$$

Figure 5.2(e) shows a plot of the uniform density.

16

Salt-and-pepper noise

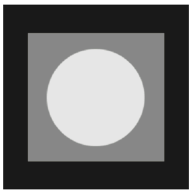
If k represents the number of bits used to represent the intensity values in a digital image, then the range of possible intensity values for that image is $[0, 2^k - 1]$ (e.g., $[0,255]$ for an 8-bit image). The PDF of *salt-and-pepper* noise is given by

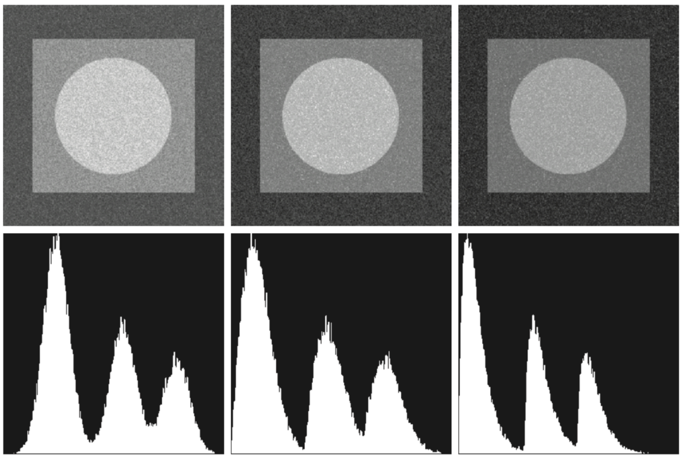
$$p(z) = \begin{cases} P_s & \text{for } z = 2^k - 1 \\ P_p & \text{for } z = 0 \\ 1 - (P_s + P_p) & \text{for } z = V \end{cases} \quad (5\text{-}16)$$

AKA, bipolar impulse noise (unipolar if either Ps or Pp is 0), data-drop-out noise, spike noise

17

Noisy images and their histograms

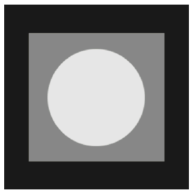


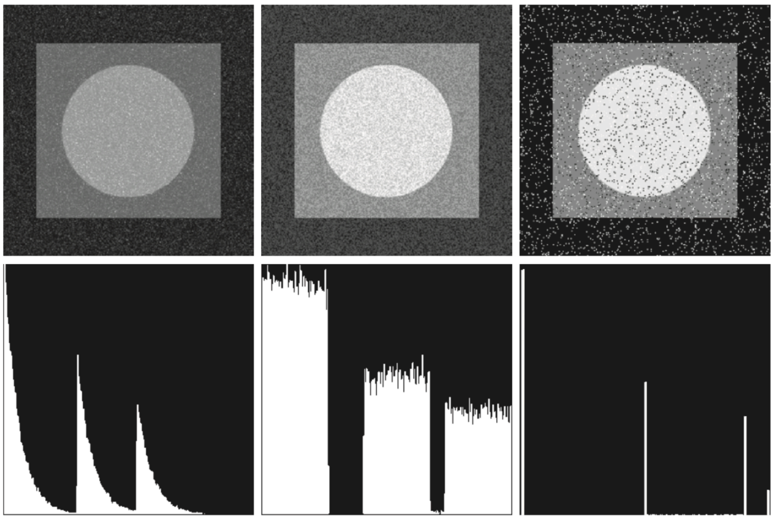


Gaussian Rayleigh Erlanga

18

Noisy images and their histograms



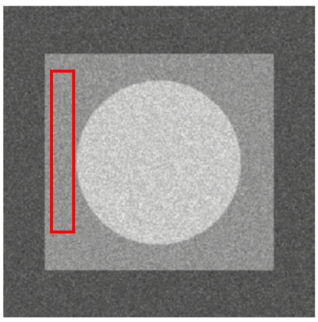


Exponential Uniform Salt-and-pepper

19

Estimation of Noise Parameters

- Estimate parameters of the PDF from small patches of reasonably constant background intensity
  - → called: image strips (or strip, subimage)

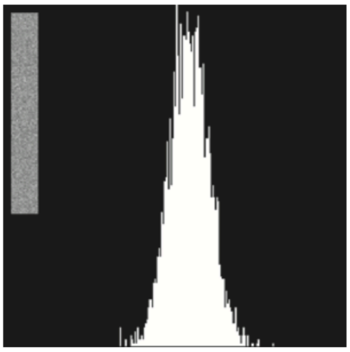


Mean: $\bar{z} = \sum_{i=0}^{L-1} z_i p_S(z_i)$

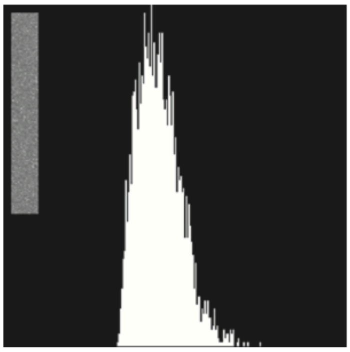
Variance: $\sigma^2 = \sum_{i=0}^{L-1} (z_i - \bar{z})^2 p_S(z_i)$

20

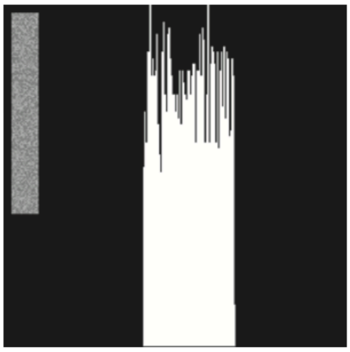
Histogram computed using small strips



Gaussian



Rayleigh



Uniform

21

Restoration in the presence of noise only

Spatial domain $$g(x,y) = (h \star f)(x,y) + \eta(x,y)$$

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

Noise only

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

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

In theory: f = g – noise
For additive random noise: spatial filtering

22

Mean filters

- Arithmetic mean filter
- Geometric mean filter
- Harmonic mean filter
- Contraharmonic mean filter

23

Arithmetic mean filter

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

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

S_{xy}: rectangular subimage window (neighborhood) of size m*n, centered on point (x,y)

24

Geometric mean filter

- Each restored pixel if given by the product of all the pixels in the subimage area, raised to the power of 1/mn
- Achieve smoothing comparable to an arithmetic mean filter
- But tend to lose less image detail

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

25

Harmonic mean filter

- Work well for salt noise but fail for pepper noise
- Also work well with Gaussian noise

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

26

Contraharmonic mean filter

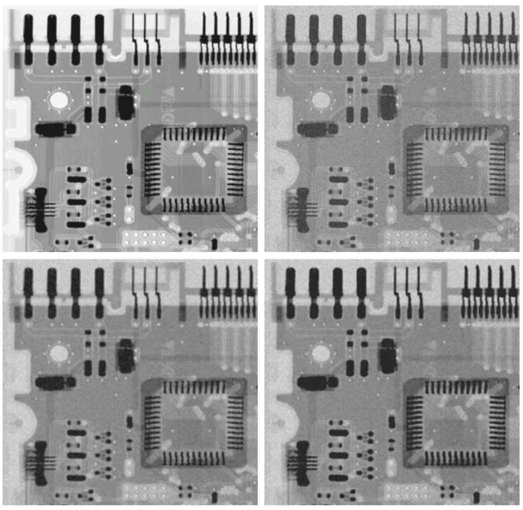
- Is well suited for reducing or virtually eliminating the effects of salt-and-pepper noise.
- Q > 1: eliminates pepper noise
- Q < 0: eliminates salt noise
- Q = 0: arithmetic mean filter
- Q = -1: harmonic mean filter

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

Q is called the order of the filter

27

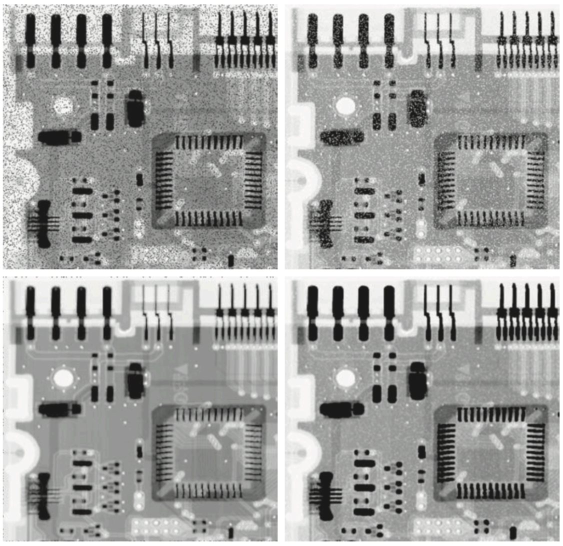
An x-ray image With additive Gaussian noise



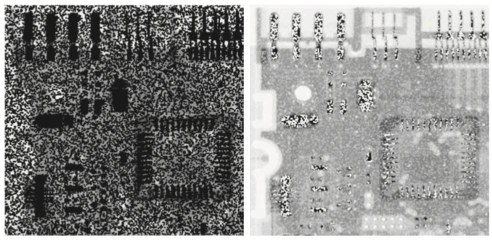
3x3 arithmetic mean filter 3x3 geometric mean filter

28

with additive pepper noise (p=0.1) With additive salt noise (p=0.1)



3x3 contraharmonic filter Q=1.5 3x3 contraharmonic filter Q=-1.5



If wrong sign of contra-harmonic filtering

29

Order-statistic filters

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

30

Median filter

- Excellent noise-reduction capability with considerably less blurring
- Particularly effective in the presence of both bipolar and unipolar impulse noise
- Replace the value of a pixel by the median of the intensity levels in a predefined neighborhood of that pixel

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

31

Max and min filter

- Max filter: useful for finding the brightest points in an image or for eroding dark regions adjacent to bright areas
- Reduce pepper noise

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

- Min filter: useful for finding the darkest points in an image or for eroding light regions adjacent to dark areas
- Reduce salt noise

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

32

Midpoint filter

- Work best for randomly distributed noise, like Gaussian or uniform noise

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

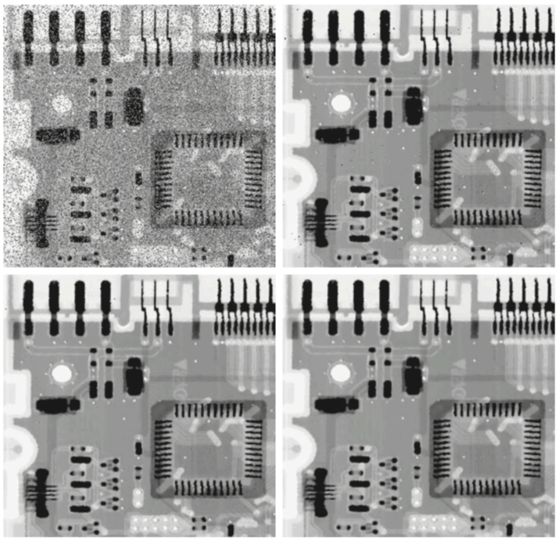
Alpha-trimmed mean filter

- Delete d/2 lowest and d/2 highest intensity values in the neighborhood S_{xy}
- Effective to combination of salt-and-pepper and Gaussian noise

$$\hat{f}(x,y) = \frac{1}{mn-d}\sum_{(r,c)\in S_{xy}} g_R(r,c)$$

d = 0 → arithmetic mean filter
d = mn-1 → median filter

Example: repeated median filter

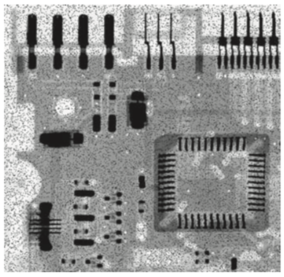


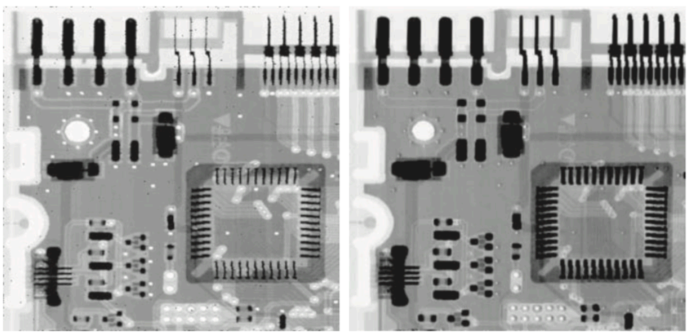
a b
c d

FIGURE 5.10
(a) Image corrupted by salt-and-pepper noise with probabilities $P_s = P_p = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.

35

Example: max & min filters





a b

FIGURE 5.11
(a) Result of filtering Fig. 5.8(a) with a max filter of size 3×3. (b) Result of filtering Fig. 5.8(b) with a min filter of the same size.

36

Adaptive filters

- Filter behaviors change based on statistical characteristics of the image inside the filter region
- Many types of adaptive filters:
  - Adaptive local noise reduction filter
  - Adaptive median filter
  - Minimum mean square error (Wiener) filter
  - Constrained least squares filter
  - ...

37

Adaptive local noise reduction filter

$$\hat{f}(x,y) = g(x,y) - \frac{\sigma_\eta^2}{\sigma_L^2}\left[g(x,y) - m_L\right]$$

σ_η^2 is the variance of the noise

σ_L^2 is the variance of pixels in S_{xy}

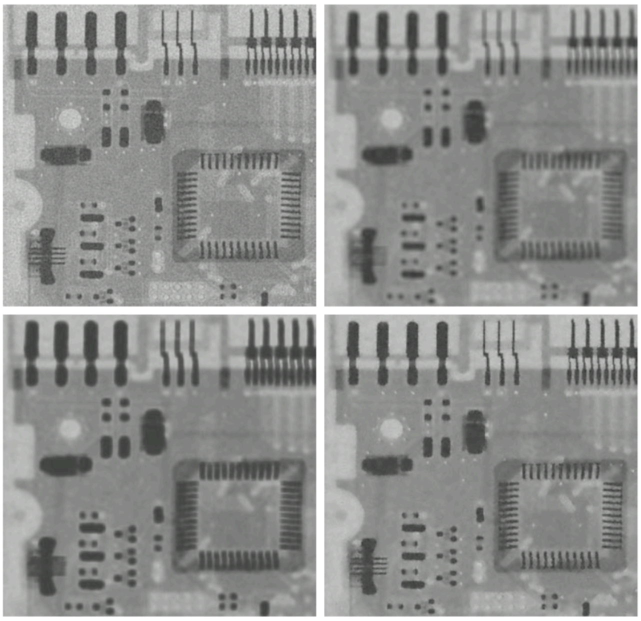
m_L is the local mean of pixels in S_{xy}

- If $\sigma_\eta^2 = 0$: no noise, filter returns g(x,y)
- If $\sigma_L^2 \gg \sigma_\eta^2$: edge regions, filter returns value close to g(x,y)
- if $\sigma_L^2 \approx \sigma_\eta^2$: local area has the same properties as the overall image, filter returns average value

38

a b
c d

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



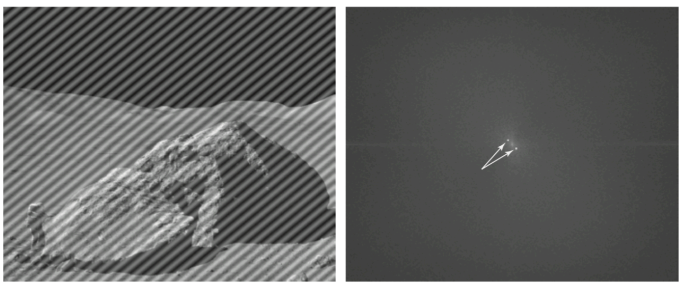
39

Periodic noise

- Arise from electrical or electromechanical interference during image acquisition

a b

FIGURE 5.5 (a) Image corrupted by additive sinusoidal noise. (b) Spectrum showing two conjugate impulses caused by the sine wave. (Original image courtesy of NASA.)



40

Notch filter

- Highpass filter transfer functions whose centers have been translated to the center of the notches

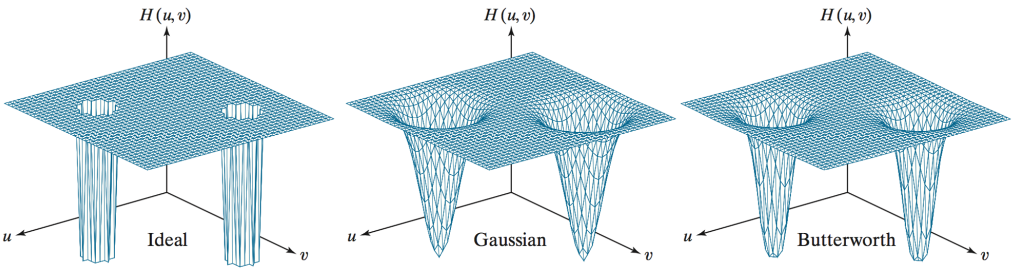
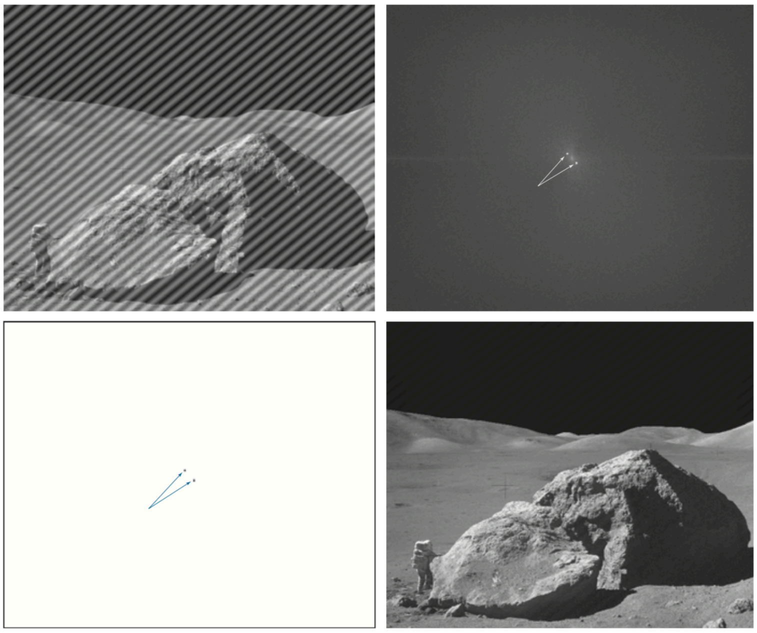


FIGURE 5.15 Perspective plots of (a) ideal, (b) Gaussian, and (c) Butterworth notch reject filter transfer functions.

41

a b
c d

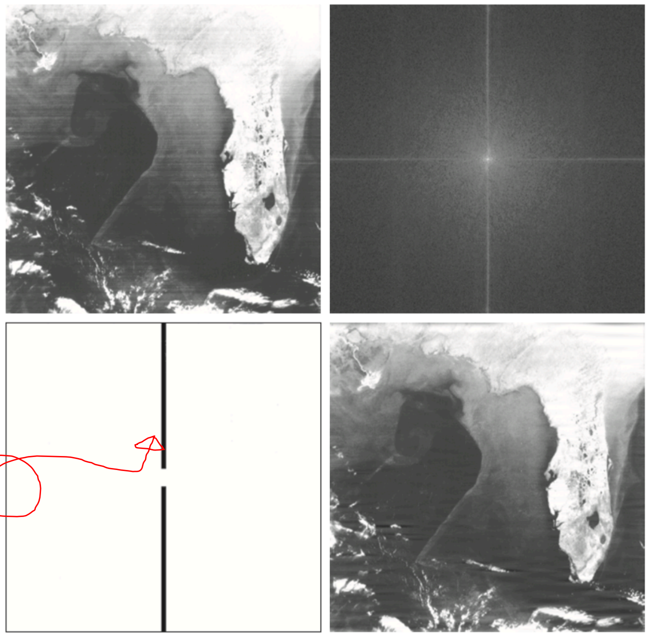
FIGURE 5.16
(a) Image corrupted by sinusoidal interference. (b) Spectrum showing the bursts of energy caused by the interference. (The bursts were enlarged for display purposes.) (c) Notch filter (the radius of the circles is 2 pixels) used to eliminate the energy bursts. (The thin borders are not part of the data.) (d) Result of notch reject filtering. (Original image courtesy of NASA.)



43

a b
c d

FIGURE 5.18
(a) Satellite image of Florida and the Gulf of Mexico. (Note horizontal sensor scan lines.) (b) Spectrum of (a). (c) Notch reject filter transfer function. (The thin black border is not part of the data.) (d) Filtered image. (Original image courtesy of NOAA.)

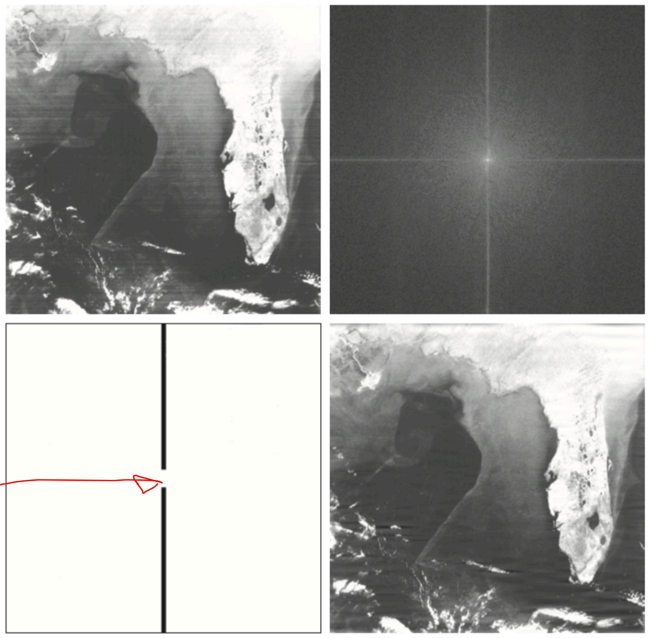


Use a narrow, rectangular notch filter function that extends along the vertical axis
→ Eliminates all components of the interference along that axis

44

a b
c d

FIGURE 5.18
(a) Satellite image of Florida and the Gulf of Mexico. (Note horizontal sensor scan lines.) (b) Spectrum of (a). (c) Notch reject filter transfer function. (The thin black border is not part of the data.) (d) Filtered image. (Original image courtesy of NOAA.)



Why do not filter near the origin?

45

Estimating the degradation function

$$g(x,y) = (\boxed{h} \star f)(x,y) + \eta(x,y)$$

46

Degradation estimation by image observation

$$g(x,y) = (\boxed{h} \star f)(x,y) + \eta(x,y)$$

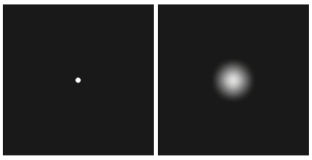
$$G(u,v) = \boxed{H(u,v)} F(u,v) + N(u,v)$$

- Degradation system H is completely characterized by its impulse response
- Select a small section from the degraded image $g_s(x,y)$
- Reconstruct an unblurred image of the same size $\hat{f}_s(x,y)$
- The degradation function can be estimated by $H_s(u,v) = \dfrac{G_s(u,v)}{\hat{F}_s(u,v)}$

47

Degradation estimation by experimentation

- When equipment similar to the equipment used to acquire the degraded image is available
  - → varying the acquisition settings until they are degraded as closely as possible to the image we wish to restore
  - Then, obtain the impulse response of the degradation by imaging an impulse (small dot of light) using the same setting
  - The impulse response is commonly referred as the Point Spread Function (PSF)



$$H(u,v) = \frac{G(u,v)}{A}$$

48

Degradation estimation by modeling

- Modeling the environments
  - Example: turbulence (wind)
- Derive a mathematical model
  - Example: model of motion (between the image and the sensor during image acquisition)

49

Modeling the environments

- Model the environmental conditions that cause degradations
- For example, a degradation model based on the physical characteristics of atmospheric turbulence (Hufnagel & Stanley 1964):

$$H(u,v) = e^{-k(u^2 + v^2)^{5/6}}$$

k: a constant that depends on the nature of the turbulence

- Gaussian lowpass filter - recall

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2}$$

50

Modeling turbulence

$$H(u,v) = e^{-k(u^2 + v^2)^{5/6}}$$

a b
c d

FIGURE 5.25 Modeling turbulence. (a) No visible turbulence. (b) Severe turbulence, $k = 0.0025$. (c) Mild turbulence, $k = 0.001$. (d) Low turbulence, $k = 0.00025$. All images are of size 480×480 pixels. (Original image courtesy of NASA.)

51

Inverse filtering

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

An estimate

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} = F(u,v) + \frac{N(u,v)}{H(u,v)}$$

However,

1. Even if we know the degradation function, we cannot recover the undegraded image exactly because N(u,v) is not known
2. If the degradation function has zero or very small values, then the ratio N(u,v)/H(u,v) could easily dominate the term F(u,v)

   → Solution: to limit the filter frequencies to values near the origin H(0,0)

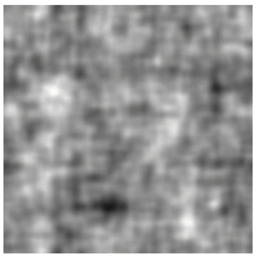
52

Inverse filter example

Input image: severe turbulence, k = 0.0025

Degradation function:

$$H(u,v) = e^{-k\left[(u + M/2)^2 + (v - N/2)^2\right]^{5/6}}$$



Using full filter

Cut off outside a radius of 40

Cut off outside a radius of 70

Cut off outside a radius of 85

53

Wiener filtering

- Incorporate both the degradation function and statistical characteristics of noise into the restoration process
- Method:
  - Images and noise = random variables
  - Objective:

Minimize the error: $$e^2 = E\{(f - \hat{f})^2\}$$

uncorrupted image   estimated

55

Wiener filtering

$(h * x)(t)$

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

$G =$

Given a system:

$$y(t) = (h * x)(t) + n(t)$$

where $*$ denotes convolution and:

- $x(t)$ is some original signal (unknown) at time t.
- $h(t)$ is the known impulse response of a linear time-invariant system
- $n(t)$ is some unknown additive noise, independent of $x(t)$
- $y(t)$ is our observed signal

Our goal is to find some $g(t)$ so that we can estimate $x(t)$ as follows:

$$\hat{x}(t) = (g * y)(t)$$

where $\hat{x}(t)$ is an estimate of $x(t)$ that minimizes the mean square error

$$\epsilon(t) = \mathbb{E}|x(t) - \hat{x}(t)|^2,$$

with $\mathbb{E}$ denoting the expectation. The Wiener deconvolution filter provides such a $g(t)$. The filter is most easily described in the frequency domain:

$$G(f) = \frac{H^*(f)S(f)}{|H(f)|^2 S(f) + N(f)}$$

where:

- $G(f)$ and $H(f)$ are the Fourier transforms of $g(t)$ and $h(t)$,
- $S(f) = \mathbb{E}|X(f)|^2$ is the mean power spectral density of the original signal $x(t)$,
- $N(f) = \mathbb{E}|V(f)|^2$ is the mean power spectral density of the noise $n(t)$,
- $X(f)$, $Y(f)$, and $V(f)$ are the Fourier transforms of $x(t)$, and $y(t)$, and $n(t)$, respectively,
- the superscript * denotes complex conjugation.

56

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}$$

57

Wiener filtering

degraded

$$\boxed{G(u,v) = H(u,v)F(u,v) + N(u,v)}$$

estimate

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)S_f(u,v)}{S_f(u,v)|H(u,v)|^2 + S_\eta(u,v)}\right]G(u,v)$$

$$= \left[\frac{H^*(u,v)}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}\right]G(u,v)$$

$$= \left[\frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + \underbrace{S_\eta(u,v)/S_f(u,v)}_{\text{signal-to-noise ratio SNR}}}\right]G(u,v)$$

1. $\hat{F}(u,v)$ = Fourier transform of the estimate of the undegraded image.
2. $G(u,v)$ = Fourier transform of the degraded image.
3. $H(u,v)$ = degradation transfer function (Fourier transform of the spatial degradation).
4. $H^*(u,v)$ = complex conjugate of $H(u,v)$.
5. $|H(u,v)|^2 = H^*(u,v)H(u,v)$.
6. $S_\eta(u,v) = |N(u,v)|^2$ = power spectrum of the noise [see Eq. (4-89)]†
7. $S_f(u,v) = |F(u,v)|^2$ = power spectrum of the undegraded image.

58

Wiener filter derivation

As mentioned above, we want to produce an estimate of the original signal that minimizes the mean square error, which may be expressed:

$$\epsilon(f) = \mathbb{E}\left|X(f) - \hat{X}(f)\right|^2 .$$

The equivalence to the previous definition of ϵ, can be derived using Plancherel theorem or Parseval's theorem for the Fourier transform.

If we substitute in the expression for $\hat{X}(f)$, the above can be rearranged to

$$\begin{aligned}\epsilon(f) &= \mathbb{E}|X(f) - G(f)Y(f)|^2 \\ &= \mathbb{E}|X(f) - G(f)\left[H(f)X(f) + V(f)\right]|^2 \\ &= \mathbb{E}\left|\left[1 - G(f)H(f)\right]X(f) - G(f)V(f)\right|^2\end{aligned}$$

If we expand the quadratic, we get the following:

$$\begin{aligned}\epsilon(f) = &\left[1 - G(f)H(f)\right]\left[1 - G(f)H(f)\right]^* \mathbb{E}|X(f)|^2 \\ &- \left[1 - G(f)H(f)\right]G^*(f)\,\mathbb{E}\left\{X(f)V^*(f)\right\} \\ &- G(f)\left[1 - G(f)H(f)\right]^* \mathbb{E}\left\{V(f)X^*(f)\right\} \\ &+ G(f)G^*(f)\,\mathbb{E}|V(f)|^2\end{aligned}$$

However, we are assuming that the noise is independent of the signal, therefore:

$$\mathbb{E}\left\{X(f)V^*(f)\right\} = \mathbb{E}\left\{V(f)X^*(f)\right\} = 0$$

Substituting the power spectral densities $S(f)$ and $N(f)$, we have:

$$\epsilon(f) = \left[1 - G(f)H(f)\right]\left[1 - G(f)H(f)\right]^* S(f) + G(f)G^*(f)N(f)$$

To find the minimum error value, we calculate the Wirtinger derivative with respect to $G(f)$ and set it equal to zero.

$$\frac{d\epsilon(f)}{dG(f)} = G^*(f)N(f) - H(f)\left[1 - G(f)H(f)\right]^* S(f) = 0$$

This final equality can be rearranged to give the Wiener filter.

https://en.wikipedia.org/wiki/Wiener_deconvolution

59

Minimum mean square error

Minimize the error: $e^2 = E\left\{(f - \hat{f})^2\right\}$

original restored

$$\mathrm{MSE} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[f(x,y) - \hat{f}(x,y)\right]^2$$

If consider the restored image to be "signal"
The difference between this image and the original to be "noise"

Signal to noise ratio:

$$\mathrm{SNR} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\hat{f}(x,y)^2 \Bigg/ \sum_{x=0}^{M-1}\sum_{y=0}^{M-1}\left[f(x,y) - \hat{f}(x,y)\right]^2$$

Low noise → high SNR
High noise → low SNR

60

Inverse filter vs Wiener filter

a b c

FIGURE 5.28 Comparison of inverse and Wiener filtering. (a) Result of full inverse filtering of Fig. 5.25(b). (b) Radially limited inverse filter result. (c) Wiener filter result.

61

Inverse filter Wiener filter

Additive Gaussian noise, zero mean
variance = 650

Variance = 65

Variance = 650/10^5

62

Reference

- R. C. Gonzalez, R. E. Woods, “Digital Image Processing,” 4th edition, Pearson, 2018.