7-9 Monday – 309-GD2

Xử lý ảnh
INT3404 1

Giảng viên: TS. Nguyễn Thị Ngọc Diệp

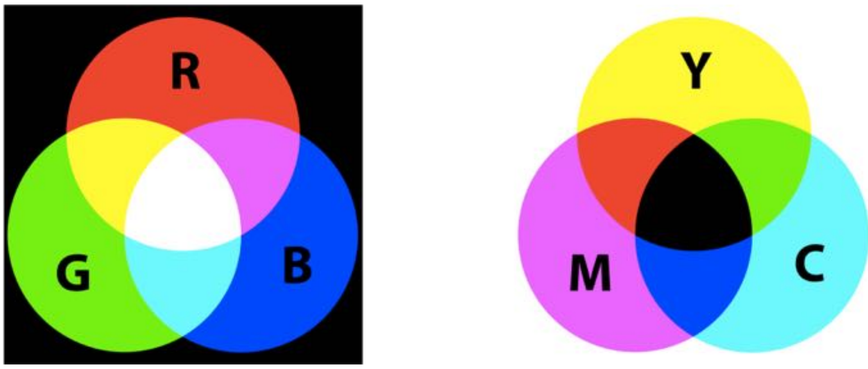
Email: ngocdiep@vnu.edu.vn

Slide & code: https://github.com/chupibk/INT3404_1

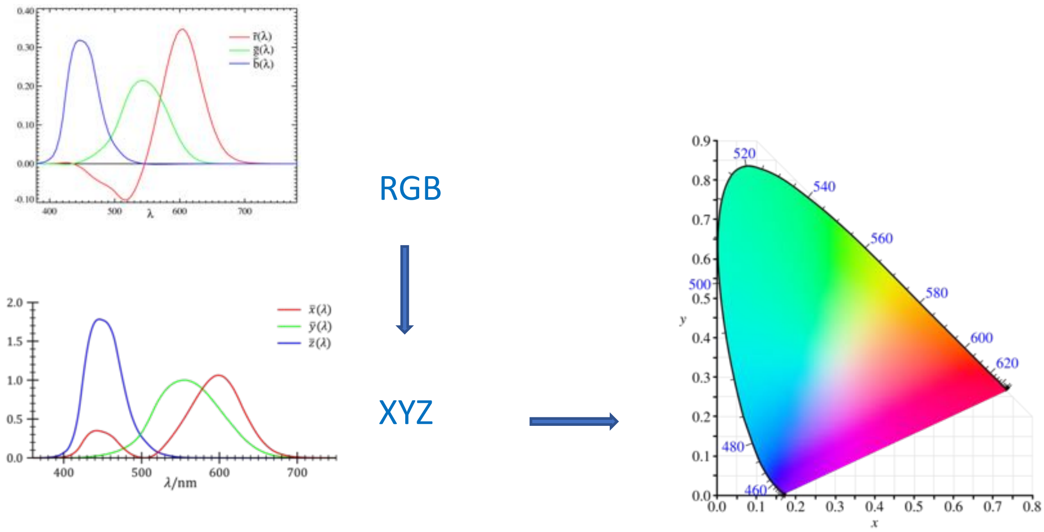
Final project - Registration form

- https://forms.gle/Qpp1hbX9QPG875gT8
- Speadsheet to edit:
  - https://docs.google.com/spreadsheets/d/1c13rgmfNdIpJEPD544qDY2sHxH-ZOxzobeW8fI1CI_w/edit#gid=82493842
  - → Edit with care!
- Final project schedule: week 13, 14, 15

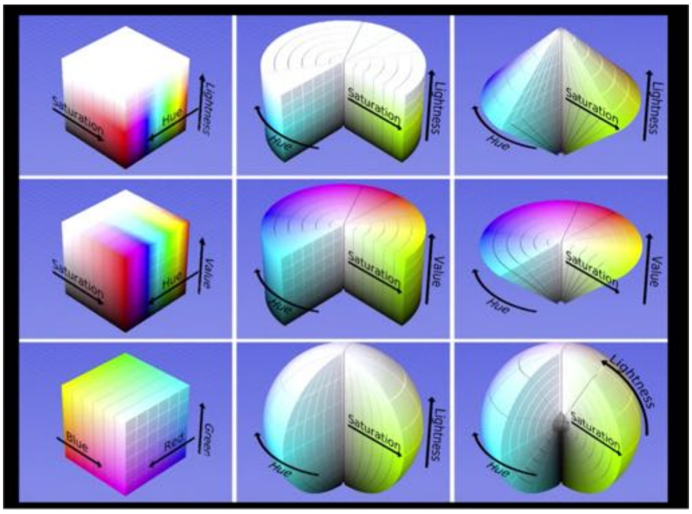
Week 9 recall: Color creation



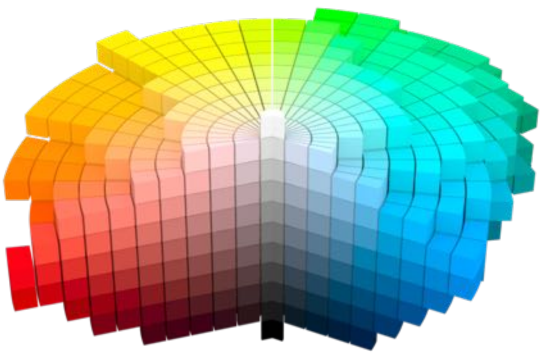
Week 9 recall: CIE chromaticity diagram (1931)



Week 9 recall: Color models



Addition: Munsell color space



Lịch trình

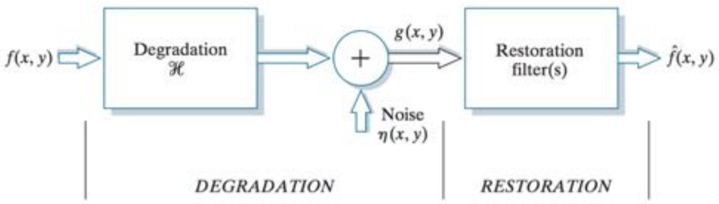
Tuần	Nội dung	Yêu cầu đối với sinh viên
1	Giới thiệu môn học Làm quen với OpenCV + Python	Cài đặt môi trường: Python 3, OpenCV 3, Numpy, Jupyter Notebook
2	Phép toán điểm (Point operations) – Điều chỉnh độ tương phản – Ghép ảnh	Làm bài tập 1: điều chỉnh gamma tìm contrast hợp lý
3	Histogram - Histogram equalization - Phân loại ảnh dùng so sánh histogram	Thực hành ở nhà
4	Phép lọc trong không gian điểm ảnh (linear processing filtering) - làm mịn, làm sắc ảnh	Thực hành ở nhà Tìm hiểu thêm các phép lọc
5	Tìm cạnh (edge detection)	Thực hành ở nhà
6	Các phép toán hình thái (Erosion, Dilation, Opening, Closing) - tìm biển số	Làm bài tập 2: tìm barcode
7	Chuyển đổi không gian - miền tần số (Fourier) - Hough transform	Thực hành ở nhà
8	Phân vùng (segmentation) - depth estimation - threshold-based - watershed/grabcut	Đăng ký thực hiện bài tập lớn
9	Mô hình màu Chuyển đổi giữa các mô hình màu	Làm bài tập 3: Chuyển đổi mô hình màu và thực hiện phân vùng
10	Mô hình nhiễu -Giảm nhiễu -Khôi phục ảnh -Giảm nhiễu chu kỳ - Ước lượng hàm Degration -Hàm lọc ngược, hàm lọc Wiener	Thực hành ở nhà
11	Template matching – Image Matching	Làm bài tập 4: puzzle
12	Nén ảnh	Thực hành ở nhà
13	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
14	Hướng dẫn thực hiện đồ án môn học	Trình bày đồ án môn học
15	Tổng kết cuối kỳ	Ôn tập

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

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)

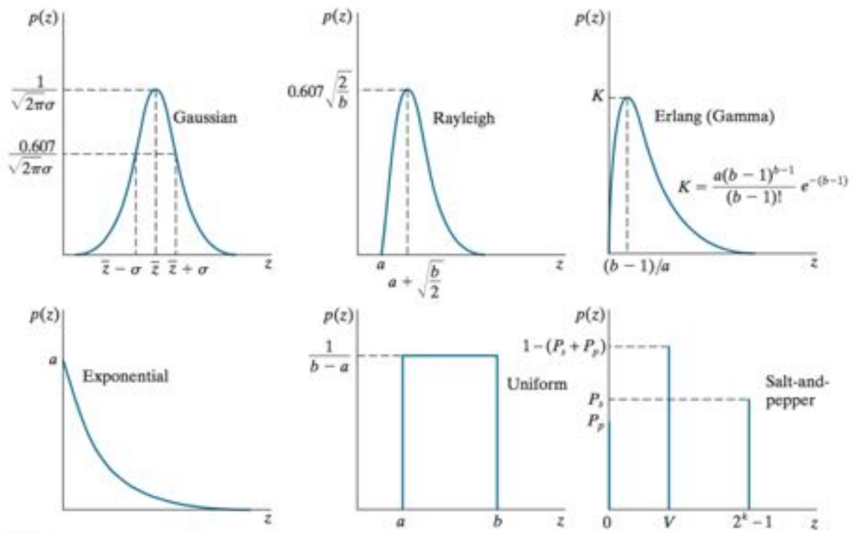
Where noise comes from?

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

- 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

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



a b c
d e f

FIGURE 5.2 Some important probability density functions.

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.

Rayleigh noise

The PDF of *Rayleigh* noise is given by

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

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

$$\bar{z} = a + \sqrt{\pi b/4} \tag{5-5}$$

and

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

Erlang (gamma) noise

The PDF of Erlang noise is

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

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

$$\bar{z} = \frac{b}{a} \tag{5-8}$$

and

$$\sigma^2 = \frac{b}{a^2} \tag{5-9}$$

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.

Uniform noise

The PDF of *uniform* noise is

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

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.

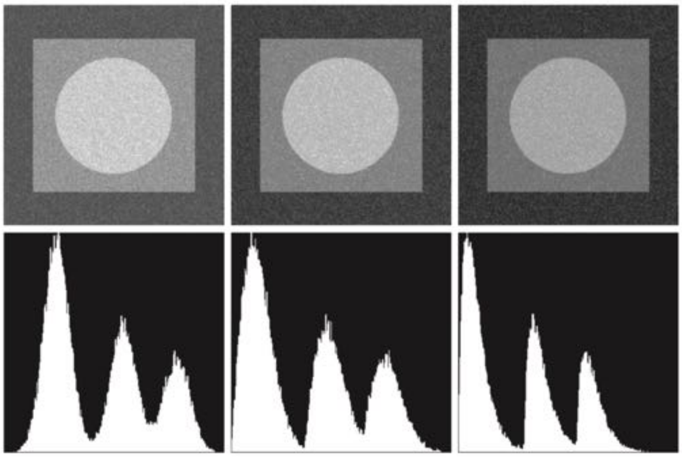
Salt-and-pepper noise

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

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

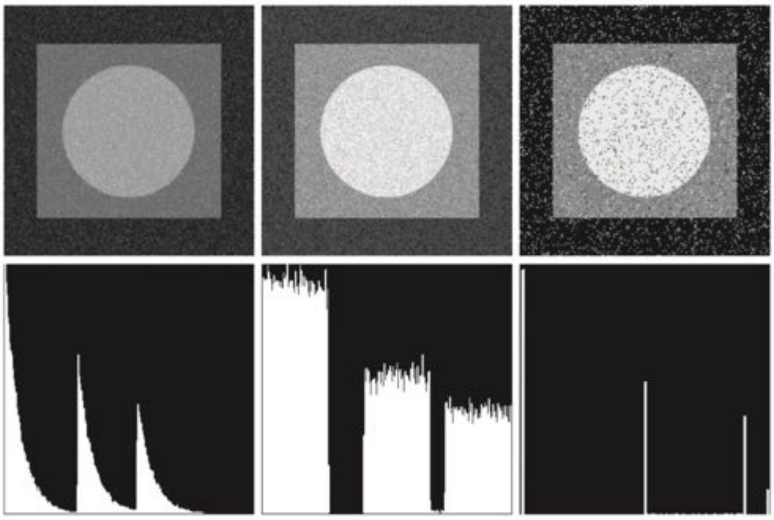
AKA, bipolar impulse noise (unipolar if either Ps or Pp is 0), data-drop-out noise, spike noise

Noisy images and their histograms



Gaussian Rayleigh Erlanga

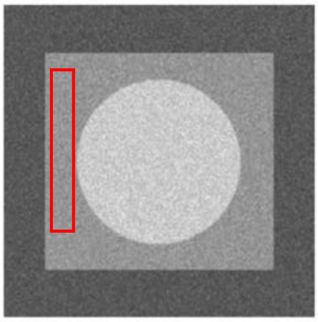
Noisy images and their histograms



Exponential Uniform Salt-and-pepper

Estimation of Noise Parameters

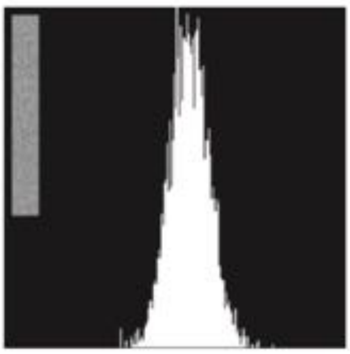
- Estimate parameters of the PDF from small patches of reasonably constant background intensity
  - → called: image strips (or strip, subimage)



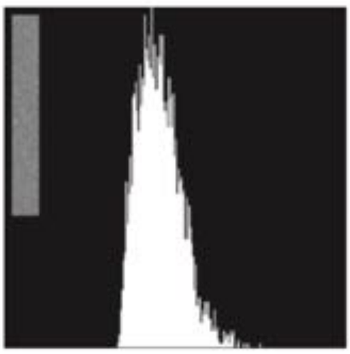
Mean: $$\bar{z} = \sum_{i=0}^{L-1} z_i p_S(z_i)$$

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

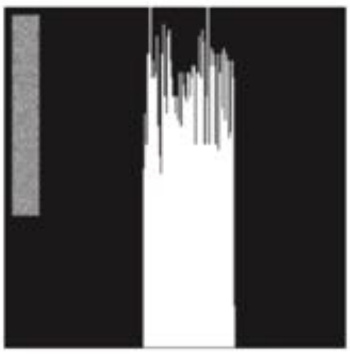
Histogram computed using small strips



Gaussian



Rayleigh



Uniform

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

Mean filters

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

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)

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

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

Harmonic mean filter

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

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

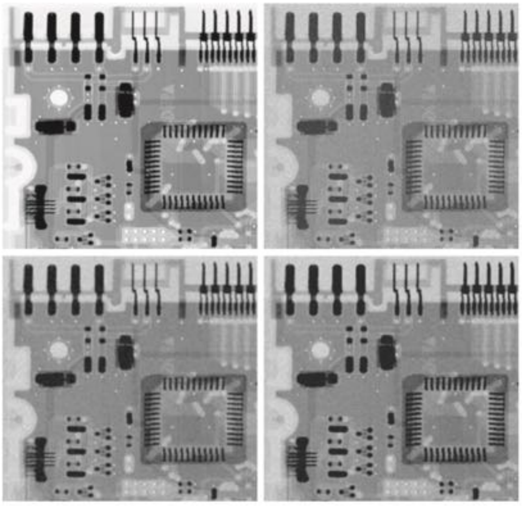
Contraharmonic mean filter

- 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

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

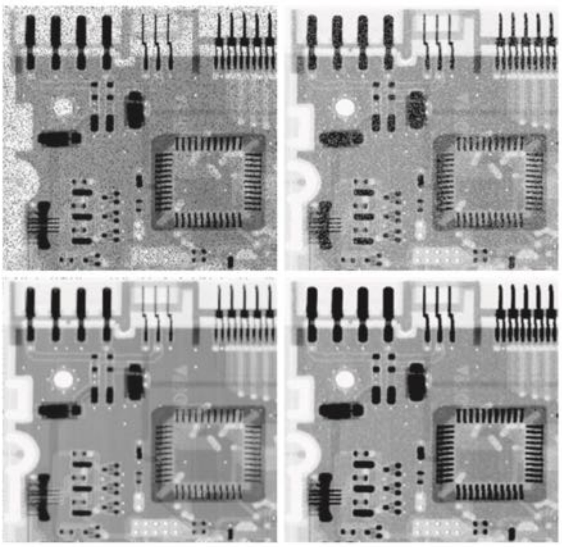
Q is called the order of the filter

An x-ray image With additive Gaussian noise

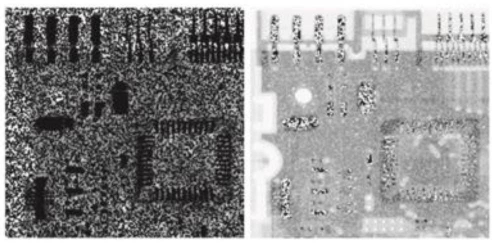


3x3 arithmetic mean filter 3x3 geometric mean filter

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 contraharmonic filtering

Order-statistic filters

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

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

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

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

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

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

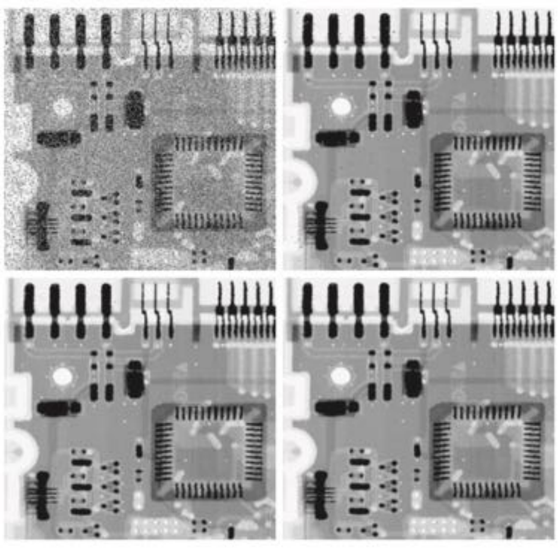
$$\hat{f}(x,y) = \frac{1}{mn - d} \sum_{(r,c)\in S_{xy}} g_R(r,c)$$

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

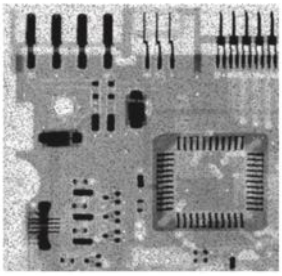
Example: repeated median filter

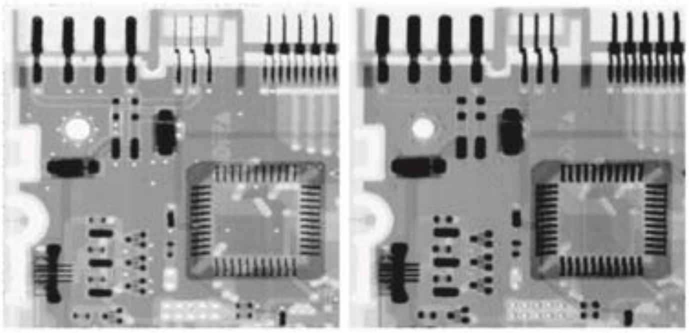
a b
c d

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.



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.

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
  - ...

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

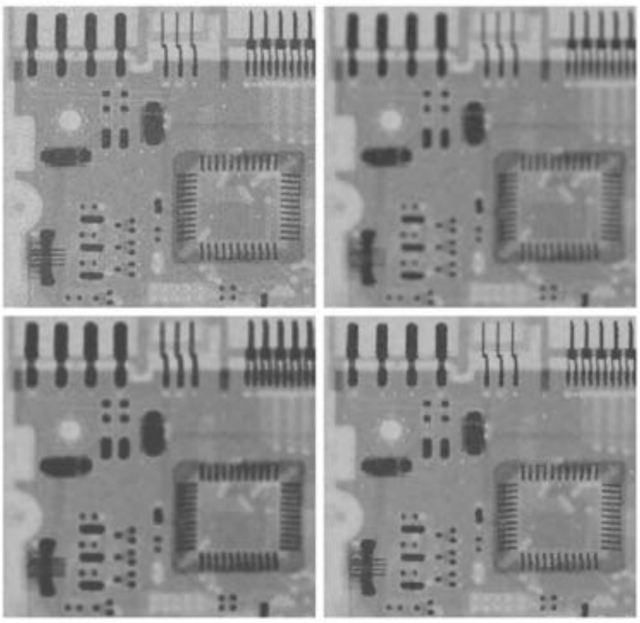
σ_L^2 is the variance of pixels in S_{xy}

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

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.



Periodic noise

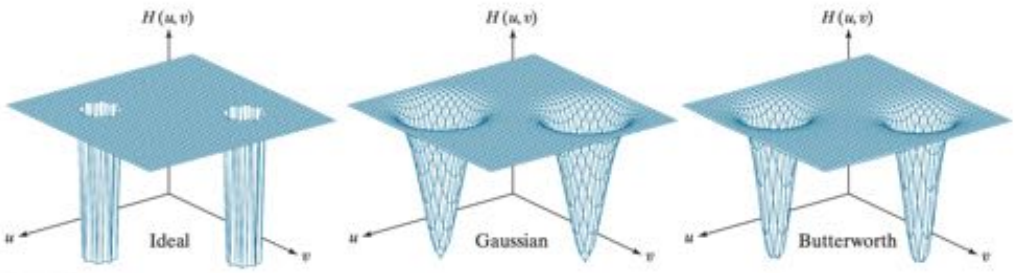
- Arise from electrical or electromechanical interference during image acquisition

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.)

Notch filter

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

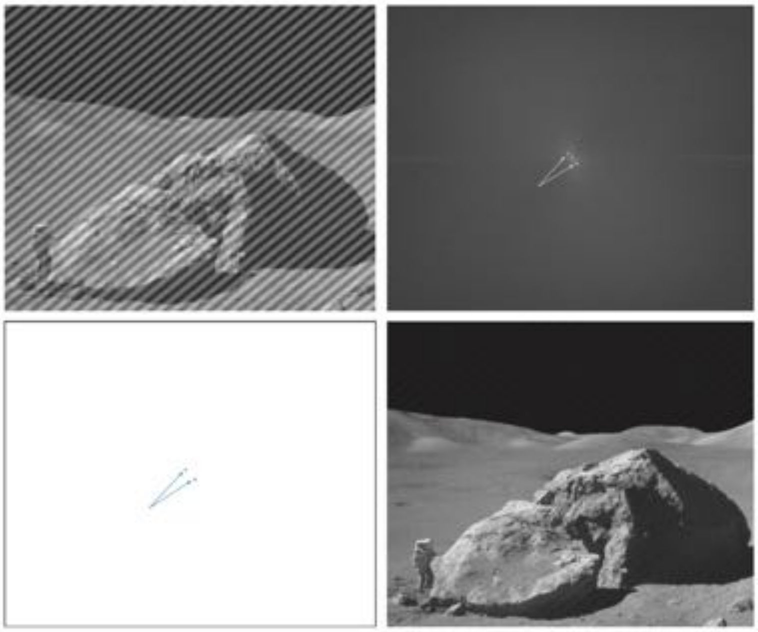


a b c

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

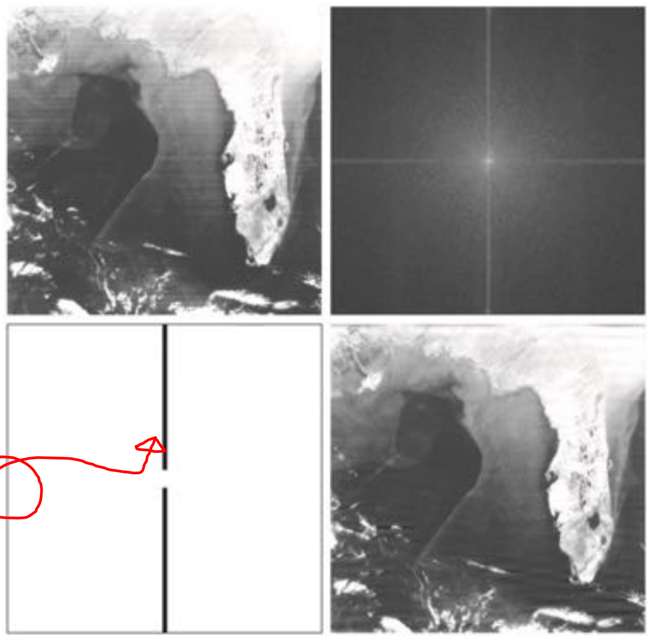
a b
c d

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.)



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

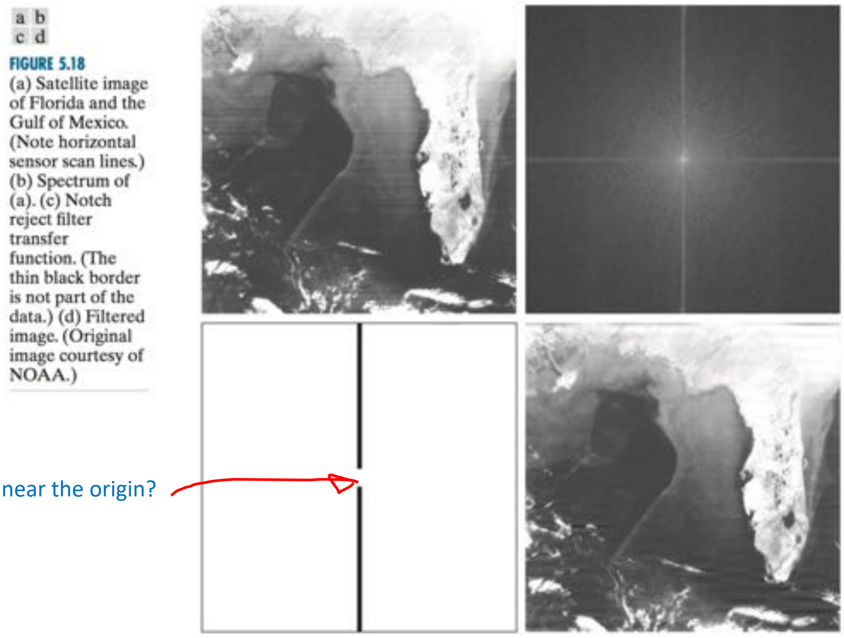


FIGURE 5.18 (a) Satellite image of Florida and the Gulf of Mexico. (Note horizontal sensor scan lines.) (b) Spectrum of (a). (c) Notch reject filter transfer function. (The thin black border is not part of the data.) (d) Filtered image. (Original image courtesy of NOAA.)

Why do not filter near the origin?

Estimating the degradation function

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

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 $H_s(u,v) = \dfrac{G_s(u,v)}{\hat{F}_s(u,v)}$

Degradation estimation by experimentation

- Point spread function (PSF)
- Used in optics
- The impulse becomes a point of light
- The impulse response is commonly referred as the PSF

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

Degradation estimation by modeling

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.)

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

Inverse filtering

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

An estimate

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

However,

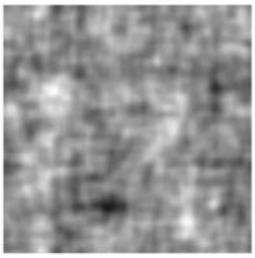
1. Even if we know the degradation function, we cannot recover the undegraded image exactly because N(u,v) is not known
2. If the degradation function has zero or very small values, then the ratio N(u,v)/H(u,v) could easily dominate the term F(u,v)
   → Solution: to limit the filter frequencies to values near the origin H(0,0)

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

Wiener filtering

$$\begin{aligned}\hat{F}(u,v) &= \left[\frac{H^*(u,v)S_f(u,v)}{S_f(u,v)|H(u,v)|^2 + S_\eta(u,v)}\right]G(u,v) \\ &= \left[\frac{H^*(u,v)}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}\right]G(u,v) \\ &= \left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}\right]G(u,v) \boxed{= \left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2 + K}\right]G(u,v)}\end{aligned}$$

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.

Inverse filter vs Wiener filter

a b c

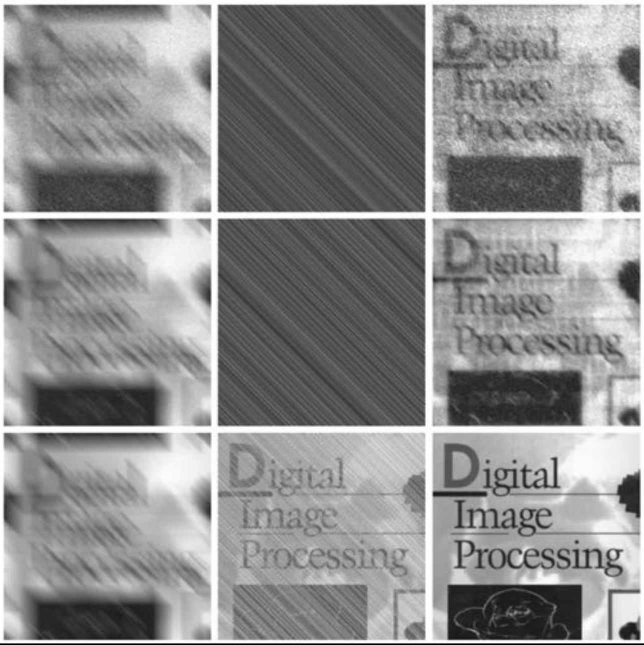
FIGURE 5.28 Comparison of inverse and Wiener filtering. (a) Result of full inverse filtering of Fig. 5.25(b). (b) Radially limited inverse filter result. (c) Wiener filter result.

Inverse filter Wiener filter

Additive Gaussian noise, zero mean
variance = 650

Variance = 65

Variance = 650/10^5



Reference

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

Final exam exemption – special opportunity

- Write a report that clarifies three following questions:

1. Explain the intuition and the derivation of the degradation function modeling

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

2. Explain the derivation of the Wiener filter and why it is equivalent to minimize mean square error

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

3. Explain why the inverse filter result (right image) caused the black regions in the original image (left) less black.

Exemption rules

- Answer all three questions correctly
- Only one student who made the best report
  - Fastest and most correct
- Submit via email:
  - Mail to: ngocdiep@vnu.edu.vn
  - Title: [final exam exemption] Student ID
  - Deadline: Nov 10, 2019 23:59 (Hanoi time)