Reminder: Proposal of Final Project

Due: 11:59 pm, October 4\textsuperscript{th}.

Late submission penalty applies.

Include

\begin{itemize}
  \item Title and names of the team member
  \item Topic: a research project or a survey
  \item Brief introduction on the background
  \item Timeline and project management for a teamwork
\end{itemize}

At most one page

Each team only needs one abstract
Today’s Agenda

• Image Degradation and Restoration
Mean Filter

Contraharmonic Mean Filter

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

Q is the order of the filter
• positive Q removes pepper noise
• negative Q removes salt noise
• Special cases: Q=0, Q=-1
**An Example**

**Pepper noise**

**Salt noise**

**FIGURE 5.8**
(a) Image corrupted by pepper noise with a probability of 0.1. (b) Image corrupted by salt noise with the same probability. (c) Result of filtering (a) with a $3 \times 3$ contra-harmonic filter of order 1.5. (d) Result of filtering (b) with $Q = -1.5$.

$Q=1.5$

$Q=-1.5$
A Failed Case with Wrong Sign of Contraharmonic Filter

**FIGURE 5.9**
Results of selecting the wrong sign in contraharmonic filtering.
(a) Result of filtering Fig. 5.8(a) with a contraharmonic filter of size $3 \times 3$ and $Q = -1.5$.
(b) Result of filtering 5.8(b) with $Q = 1.5$. 
Summary: Mean Filters for Continuous Noise Models

Arithmetic Mean Filter: a linear filter

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

• Work well for continuous noise
Summary: Non-linear Mean Filters

Geometric Mean Filter

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

Work well for:
- Continuous noise
- Salt noise

Harmonic Mean Filter

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

Fail for the pepper noise

Contraharmonic Mean Filter

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

Q is the order of the filter
- positive Q removes pepper noise
- negative Q removes salt noise
- Special cases: Q=0, Q=-1

Work well for:
- Continuous noise
- Salt noise

Fail for the pepper noise
Order-Statistic Filters -- Median Filter

Repeating median filter will remove most of the noise while increase image blurring.
Order-Statistic Filters  -- Max/Min Filters

- Find the extreme points
- Remove the targeting impulse noise
Order-Statistic Filters

Midpoint filter

- Combine order statistics and averaging
- Works best for randomly distributed noise, like Gaussian or uniform noise
- Not suitable for impulse noise and blur the boundary

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

Random noise
Salt noise
Pepper noise

Order-Statistic Filters

Alpha-trimmed mean filter

• Delete d/2 lowest and d/2 highest intensity values
• A balance between arithmetic mean filter and median filter
• Suitable for combined salt-and-pepper and Gaussian noise

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

FIGURE 5.12
(a) Image corrupted by additive uniform noise.
(b) Image additionally corrupted by additive salt-and-pepper noise. Image (b) filtered with a $5 \times 5$;
(c) arithmetic mean filter;
(d) geometric mean filter;
(e) median filter; and (f) alpha-trimmed mean filter with $d = 5$. 
Adaptive Filters

Adaptive local noise reduction filter

Key elements:
- the intensity value $g(x, y)$
- the variance of the noise $\sigma_n^2$
- the local mean of the neighborhood $m_L$
- the local variance of the neighborhood $\sigma_L^2$

\[
\hat{f}(x, y) = g(x, y) - \frac{\sigma_n^2}{\sigma_L^2} [g(x, y) - m_L]
\]

Properties:
- If $\sigma_n^2 = 0$, $\hat{f}(x, y) = g(x, y) \rightarrow$ ideal case
- If $\sigma_n^2 / \sigma_L^2$ is low, preserve the edge information $\hat{f}(x, y) \approx g(x, y)$
Examples

**FIGURE 5.13**
(a) Image corrupted by additive Gaussian noise of zero mean and variance 1000.
(b) Result of arithmetic mean filtering.
(c) Result of geometric mean filtering.
(d) Result of adaptive noise reduction filtering. All filters were of size $7 \times 7$. 
Adaptive Median Filter

Stage A: check if the median value is an extreme value

\[ A1 = z_{med} - z_{min} \]
\[ A2 = z_{med} - z_{max} \]

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 \( z_{med} \)

Stage B: check if the center pixel is an extreme value

\[ B1 = z_{xy} - z_{min} \]
\[ B2 = z_{xy} - z_{max} \]

If \( B1 > 0 \) AND \( B2 < 0 \), output \( z_{xy} \)
Else output \( z_{med} \)

Goal 1: remove salt-and-pepper noise with higher probability
Goal 2: smoothing the noise other than impulses
Goal 3: reduce distortion
Adaptive Median Filter -- Example

**FIGURE 5.14** (a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.25$. (b) Result of filtering with a $7 \times 7$ median filter. (c) Result of adaptive median filtering with $S_{\text{max}} = 7$. 
Periodical Noise

Image is corrupted by a set of sinusoidal noise of different frequencies.

FIGURE 2.40
(a) Image corrupted by sinusoidal interference. (b) Magnitude of the Fourier transform showing the bursts of energy responsible for the interference. (c) Mask used to eliminate the energy bursts. (d) Result of computing the inverse of the modified Fourier transform. (Original image courtesy of NASA.)
Linear, Position-Invariant Degradations – Noise Free Case

\[ g(x, y) = H[f(x, y)] \]

Linearity \( H[af_1(x, y) + bf_2(x, y)] = aH[f_1(x, y)] + bH[f_2(x, y)] \)

Position/space invariant \( H[f(x - \alpha, y - \beta)] = g(x - \alpha, y - \beta) \)

\( H \) does not depend on the location \((x,y)\), only represented by the input and output.
Impulse Response for Linear $H$

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)\delta(x - \alpha, y - \beta)\,d\alpha\,d\beta$$  \hspace{1cm} \text{Sifting property}$$

$$g(x, y) = H[f(x, y)]$$

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)H[\delta(x - \alpha, y - \beta)]\,d\alpha\,d\beta$$
Impulse Response for Linear $H$

\[
g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)H[\delta(x - \alpha, y - \beta)]d\alpha d\beta
\]

Impulse response \( h(x, \alpha, y, \beta) = H[\delta(x - \alpha, y - \beta)] \)

\[
g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)h(x, \alpha, y, \beta)d\alpha d\beta
\]

If $H$ is position invariant

\[
g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)h(x - \alpha, y - \beta)d\alpha d\beta
\]

convolution
Image Degradations

\[ g(x, y) = H[f(x, y)] + \eta(x, y) \]

\[ g(x, y) = h(x, y) \otimes f(x, y) + \eta(x, y) \]
Degradation VS Restoration

\[ g(x, y) = H[f(x, y)] + \eta(x, y) \]

Note: a linear, position invariant degradation system with additive noise can be modeled as the convolution of the degradation function with the image plus the additive noise.

\[ g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) h(x-\alpha, y-\beta) \, d\alpha \, d\beta + \eta(x, y) \]
Estimate the Degradation Function

- Observation
- Experimentation
- Mathematical modeling
Estimation by Modeling – Motion Blur

Constant velocity along x and y direction:

\[ x_0(t) = \frac{at}{T} \quad y_0(t) = \frac{bt}{T} \]

**FIGURE 5.26**
(a) Original image. (b) Result of blurring using the function in Eq. (5.6-11) with \( a = b = 0.1 \) and \( T = 1 \).
An example of motion blur

\[ g(x, y) = \int_{0}^{T} f[x - x_0(t), y - y_0(t)] dt \]

Motion in both x and y direction during acquisition
An example of motion blur

\[
G(u, v) = F(u, v) \int_{0}^{T} e^{-j2\pi[ux_0(t)+vy_0(t)]} dt
\]

\[
H(u, v) = \int_{0}^{T} e^{-j2\pi[ux_0(t)+vy_0(t)]} dt
\]
Estimation by Modeling – Example

Constant velocity along x and y direction:

\[ x_0(t) = \frac{at}{T} \quad y_0(t) = \frac{bt}{T} \]

What is \( H(u, v) \)?

\[
H(u, v) = T \frac{\sin[\pi(ua + vb)]}{\pi(ua + vb)} e^{-j\pi(ua+vb)}
\]
Image Restoration

Given the degradation system $H$ and the input image $G$, recover the original image $F$

- Inverse filtering
- Wiener filtering
Inverse Filtering

Ideally:

\[ G(u, v) = H(u, v)F(u, v) \]

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

In practice:

\[ G(u, v) = H(u, v)F(u, v) + N(u, v) \]

\[ \hat{F}(u, v) = F(u, v) + \frac{N(u, v)}{H(u, v)} \]

Low pass filtering

Limiting the analysis to frequencies near the origin (0,0)
An Example of Inverse Filtering

Original image

Degraded image

\[ \frac{G(u,v)}{H(u,v)} \]
An Example of Inverse Filtering (Cont.)

Original image

Degraded image

\[ \frac{G(u,v)}{H(u,v)} \]

Full filter

Cutoff radius = 40

Cutoff radius = 70

Cutoff radius = 85
Minimum Mean Square Error (Wiener) Filtering

Assumptions:
- Noise and image are uncorrelated
- Noise has zero mean

Original
Inverse filtering with cutoff = 70
Wiener filtering
The Formulation

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

\[
\hat{F}(u,v) = \left[ \frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + |N(u,v)|^2 / |F(u,v)|^2} \right] G(u,v)
\]

Least square error filter

\(|N(u,v)|^2\) is the power spectrum of noise

\(|F(u,v)|^2\) is the power spectrum of undegraded image
The Formulation (Cont.)

Signal to noise ratio: the metric to evaluate the restoration performance

Frequency domain: \[ SNR = \frac{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |F(u, v)|^2}{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} |N(u, v)|^2} \]

Spatial domain: \[ SNR = \frac{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \hat{f}(x, y)^2}{\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} [\hat{f}(x, y) - f(x, y)]^2} \]
The Formulation

\[ \hat{F}(u, v) = \left[ \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + |N(u, v)|^2 / |F(u, v)|^2} \right] G(u, v) \]

An approximation

\[ \hat{F}(u, v) = \left[ \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \right] G(u, v) \]
Example 2 – Motion Blur + Additive Noise

Figure 5.29 (a) 8-bit image corrupted by motion blur and additive noise. (b) Result of inverse filtering. (c) Result of Wiener filtering. (d)–(f) Same sequence, but with noise variance one order of magnitude less. (g)–(i) Same sequence, but noise variance reduced by five orders of magnitude from (a). Note in (h) how the deblurred image is quite visible through a “curtain” of noise.
Constrained Least Square Filtering

Assumption: the mean and variance of the noise is known

Matrix-vector representation: \( g = Hf + \eta \)

Alleviate the noise sensitivity by minimizing

\[
C = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} [\nabla^2 f(x, y)]^2
\]

subject to

\[
\|g - H\hat{f}\|^2 = \|\eta\|^2
\]

The image is smooth
Constrained Least Square Filtering

Frequency domain solution:

\[
\hat{F}(u, v) = \left[ \frac{H^*(u, v)}{|H(u, v)|^2 + \gamma |P(u, v)|^2} \right] G(u, v)
\]

Parameter to satisfy the constraint

The Fourier transform of Laplacian 

\[
p(x, y) = \begin{bmatrix}
0 & -1 & 0 \\
-1 & 4 & -1 \\
0 & -1 & 0
\end{bmatrix}
\]
Example

CLSF

Wiener
Optimize $\gamma$

Residual: \[ r = g - Hf \]

$\phi(\gamma) = \|r\|^2 \rightarrow$ A function monotonically increasing of $\gamma$

Adjust $\gamma$ to satisfy the constraint:

\[ \|r\|^2 = \|\eta\|^2 \pm a \rightarrow $ Accuracy factor
Optimize \( \gamma \)

1. Specify an initial \( \gamma \)
2. Compute \( \phi(\gamma) = \|r\|^2 \)
3. If \( \|r\|^2 = \|\eta\|^2 \pm a \), stop or adjust \( \gamma \) accordingly
How to Calculate $\|\eta\|^2$

$$\|\eta\|^2 = MN(\sigma_\eta^2 + m_\eta^2)$$
Example

Iteratively search for optimal parameter \( \gamma \)

**FIGURE 5.31**

(a) Iteratively determined constrained least squares restoration of Fig. 5.16(b), using correct noise parameters.
(b) Result obtained with wrong noise parameters.
Reading Assignments

Chapter 5.10 – 5.11