Noise and Restoration of Images

Dr. Praveen Sankaran

Department of ECE
NIT Calicut

February 24, 2013
Outline

1. Noise Models
2. Restoration from Noise Degradation
3. Estimation of Degradation
4. Restoration from Degradation
   - Inverse and Wiener Filtering
   - Constrained Least Squares Filtering
Degradation Model

- All digital images have some amount of noise and/or some form of degradation in them.
- Some have more due to varying sources
  - Acquisition
    - Environmental conditions.
    - Quality of sensing elements - CCD noise.
  - Transmission
  - Formation
    - Quantization noise
  - Blur
Noise Models

Degradation Model

\[ g(x, y) = f(x, y) \ast h(s, t) + \eta(x, y) \]  \hspace{1cm} (1)

- \( \eta \rightarrow \) uncorrelated noise (no relation between noise and pixel value of the image).
- think of an example where both could be correlated?
Noise Models

- Gaussian
- Uniform
- Salt & Pepper

Gaussian noise pdf (sdev 25)

Uniform noise pdf (sdev 74)

Salt and Pepper Noise
Noise Models - Explained

Un correlated Noise

The most likely value is 128 with an average difference of 25 from 128 (std. dev.).

This is sparse noise: Only 12.5% of the pixels contain noise. Of those 12.5% ...

Gaussian

All values occur with equal probability.

Salt & Pepper

... black pixels occur 75% of the time and white pixels occur 25% of the time.
In Color - Gaussian

Gaussian: Luminance pdf

Gaussian: Red Band pdf

Gaussian: Green Band pdf

Gaussian: Blue Band pdf
In Color - Uniform

- Uniform: Luminance pdf
- Uniform: Red Band pdf
- Uniform: Green Band pdf
- Uniform: Blue Band pdf
Periodic Noise
Identifying System Noise

Imaging system available
Capture a set of images of flat environments - image a solid gray board that is illuminated uniformly.

Imaging system NOT available, we have only the images
Estimate parameters from small patches of roughly constant background. We already know how to calculate mean and variance from a histogram.
Model - Simplified

\[ g[x, y] = f[x, y] + \eta[x, y] \]  \hspace{1cm} (2)

\[ G[u, v] = F[u, v] + N[u, v] \]  \hspace{1cm} (3)

- Noise term is unknown \(\rightarrow\) No blind subtraction possible.
- Spatial filtering is used mostly.

Dr. Praveen Sankaran (Department of ECE NIT Calicut)
DIP Winter 2013
February 24, 2013
Typical Filters

We have come across a lot of these already. Filter of size $m \times n$

- Arithmetic mean: Noise reduced as a result of blurring.

- Geometric mean:
  \[
  \hat{f}[x,y] = \left\{ \prod_{s,t} f[s,t] \right\}^{1/mn} \tag{4}
  \]

- Harmonic mean: works for salt, fails for pepper
  \[
  \hat{f}[x,y] = \frac{mn}{\sum_{s,t} \frac{1}{g[s,t]}} \tag{5}
  \]

- Contra-harmonic mean: works well to remove salt&pepper
  \[
  \hat{f}[x,y] = \frac{\sum_{s,t} g[s,t]^{Q+1}}{\sum_{s,t} g[s,t]^Q} \tag{6}
  \]
Typical Filters

We have come across a lot of these already. Filter of size $m \times n$

- Arithmetic mean: Noise reduced as a result of blurring.

- Geometric mean:
  \[
  \hat{f}[x, y] = \left\{ \prod_{s,t} f[s, t] \right\}^{1/mn} \tag{4}
  \]

- Harmonic mean: works for salt, fails for pepper
  \[
  \hat{f}[x, y] = \frac{mn}{\sum_{s,t} \frac{1}{g[s, t]}} \tag{5}
  \]

- Contra-harmonic mean: works well to remove salt & pepper
  \[
  \hat{f}[x, y] = \frac{\sum_{s,t} g[s, t]^{Q+1}}{\sum_{s,t} g[s, t]^{Q}} \tag{6}
  \]
Typical Filters

We have come across a lot of these already. Filter of size $m \times n$

- Arithmetic mean: Noise reduced as a result of blurring.

- Geometric mean:

$$\hat{f}[x,y] = \left( \prod_{s,t} f[s,t] \right)^{1/mn}$$  \hspace{1cm} (4)

- Harmonic mean: works for salt, fails for pepper

$$\hat{f}[x,y] = \frac{mn}{\sum_{s,t} \frac{1}{g[s,t]}}$$  \hspace{1cm} (5)

- Contra-harmonic mean: works well to remove salt&pepper

$$\hat{f}[x,y] = \frac{\sum_{s,t} g[s,t]^{Q+1}}{\sum_{s,t} g[s,t]^Q}$$  \hspace{1cm} (6)
Typical Filters

We have come across a lot of these already. Filter of size $m \times n$

- Arithmetic mean: Noise reduced as a result of blurring.
  \[
  \hat{f}[x,y] = \left( \prod_{s,t} f[s,t] \right)^{1/mn}
  \] (4)

- Geometric mean:
  \[
  \hat{f}[x,y] = \left\{ \prod_{s,t} f[s,t] \right\}^{1/mn}
  \] (4)

- Harmonic mean: works for salt, fails for pepper
  \[
  \hat{f}[x,y] = \frac{mn}{\sum_{s,t} 1 / g[s,t]}
  \] (5)

- Contra-harmonic mean: works well to remove salt & pepper
  \[
  \hat{f}[x,y] = \frac{\sum_{s,t} g[s,t]^{Q+1}}{\sum_{s,t} g[s,t]^Q}
  \] (6)
FIGURE 5.7
(a) X-ray image.
(b) Image corrupted by additive Gaussian noise. (c) Result of filtering with an arithmetic mean filter of size $3 \times 3$. (d) Result of filtering with a geometric mean filter of the same size.
(Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)
Illustration

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 $Q = -1.5$. 
Order Static Filters

1. Median filter:

\[ \hat{f}[x, y] = \text{median}_{s,t \in S_{xy}} \{f[s, t]\} \quad (7) \]

2. MinMax filters:

3. Mid point filter:

\[ \hat{f}[x, y] = \frac{1}{2} \{ \max_{s,t \in S_{xy}} \{f[s, t]\} + \min_{s,t \in S_{xy}} \{f[s, t]\}\} \quad (8) \]
Adaptive Median Filter

Key idea \(\rightarrow\) variable window area \(S_{xy}\) to work with.

- So how do we decide what area to work with? \(\rightarrow\) provide conditions.
- Assumptions
  - \(z_{min}\) = minimum intensity,
  - \(z_{max}\) = maximum intensity,
  - \(z_{med}\) = median
  - \(z_{xy}\) = value at point \([x, y]\\
  - \(S_{max}\) = maximum threshold of window size.
Restoration from Noise Degradation

Adaptive - Conditions

Stage 1
- $A_1 = z_{med} - z_{min}$
- $A_2 = z_{med} - z_{max}$
- If $A_1 > 0$ and $A_2 < 0$, go to Stage 2.
- Else increase the window size.
- If window size $\leq S_{max}$, repeat Stage 1.
- Else output $z_{med}$.

Stage 2
- $B_1 = z_{xy} - z_{min}$
- $B_2 = z_{xy} - z_{max}$
- If $B_1 > 0$ and $B_2 < 0$, output $z_{xy}$.
- Else, output $z_{xy}$. 
Idea → selectively filter out frequencies relating to noise. Can have band-reject, band-pass or notch filters.

- **Example**: band-reject

![Diagram](image.png)

**Figure 5.15** From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.
FIGURE 5.16
(a) Image corrupted by sinusoidal noise.
(b) Spectrum of (a).
(c) Butterworth bandreject filter (white represents 1).
(d) Result of filtering.
(Original image courtesy of NASA.)
Estimation of Degradation

1. Observation
2. Experimentation
3. Mathematical modeling
Degradation Model

\[ g[x, y] = H[f[x, y]] = H[x, y] \ast F[x, y] \]  


Let’s assume for now,

\[ \eta[x, y] = 0 \]
Modeling Atmospheric Turbulence

\[ H[u, v] = e^{-k[u^2 + v^2]^{5/6}} \]  (12)
Outline

1. Noise Models

2. Restoration from Noise Degradation

3. Estimation of Degradation

4. Restoration from Degradation
   • Inverse and Wiener Filtering
   • Constrained Least Squares Filtering
**Inverse Filtering**

\[
\hat{F}[u, v] = \frac{G[u, v]}{H[u, v]} \tag{13}
\]

- **Issue**
  - Presence of noise in the image.
    \[
    \hat{F}[u, v] = F[u, v] + \frac{N[u, v]}{H[u, v]} \tag{14}
    \]
  - The degradation function may have zero or small values - a real possibility as we move away from the center point of the DFT image.
    \[
    \frac{N[u, v]}{H[u, v]} \Rightarrow \text{large!}
    \]
  - Have to cut-off small values, so have to find an optimal cut-off frequency in the DFT image.
Inverse Filtering Problem Illustration
Inverse Filtering Problem Illustration

**Figure 5.27**
Restoring Fig. 5.25(b) with Eq. (5.7-1).
(a) Result of using the full filter. (b) Result with $H$ cut off outside a radius of 40; (c) outside a radius of 70; and (d) outside a radius of 85.
Wiener Filtering - Variables

Idea → minimize mean square error between the uncorrupted image and the corrupted image.

\[ e^2 = E \left\{ (f - \hat{f})^2 \right\} \]  (15)

- \( H[u, v] = \) degradation function,
- \( H^*[u, v] = \) complex conjugate of \( H[u, v] \),
- \( S_\eta[u, v] = |N[u, v]|^2 = \) power spectrum of the noise, (auto correlation of noise)
- \( S_f[u, v] = |F[u, v]|^2 = \) power spectrum of the undegraded image. (auto correlation of the image)
Wiener Filter Equation

\[
\hat{F}[u,v] = \left[ \frac{H^*[u,v] S_f[u,v]}{S_f[u,v]|H[u,v]|^2 + S_n[u,v]} \right] G[u,v] 
\]

\[
= \left[ \frac{1}{H[u,v]} \frac{|H[u,v]|^2}{|H[u,v]|^2 + S_n[u,v]/S_f[u,v]} \right] G[u,v] 
\]

\[
\approx \left[ \frac{1}{H[u,v]} \frac{|H[u,v]|^2}{|H[u,v]|^2 + K} \right] G[u,v] 
\]

The last equation approximates the wiener filter equation since in most cases we do not know accurately the power spectrum of both noise and the un-degraded image.
Comparative Illustration between Inverse Filtering and Wiener Filtering

**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.
Outline

1. Noise Models
2. Restoration from Noise Degradation
3. Estimation of Degradation
4. Restoration from Degradation
   - Inverse and Wiener Filtering
   - Constrained Least Squares Filtering
Variables - Introduction

\[ g = Hf + \eta \]  \hspace{1cm} (19)

- \( g, f, \eta \) are vectorized form of the the 2D structure, of size \( MN \times 1 \).
- \( H \rightarrow MN \times MN \).
- Objective is to minimize,
  - Criterion function,
    \[ C = M - 1 \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} [\nabla^2 f(x,y)]^2 \]  \hspace{1cm} (20)
    - Laplacian again. So sort of reduce sharpness.
    - With constraint:
      \[ \| g - H\hat{f} \|^2 = \| \eta \|^2 \]  \hspace{1cm} (21)
Final Frequency Domain Form

\[ \hat{F}[u, v] = \left( \frac{H^*[u, v]}{|H[u, v]|^2 + \gamma |P[u, v]|^2} \right) G[u, v] \]  \hspace{1cm} (22)

\[ P[u, v] = \mathbb{S}\left\{\begin{array}{c} p[x, y] = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \end{array}\right\} \]  \hspace{1cm} (23)

- Okay! So we end up finding an approximation to \( \gamma \) here instead of \( K \) in Wiener filtering.
- \( \gamma \) is a scalar value,
- \( K \) was the ratio of two unknown frequency functions.
- Adjusting \( \gamma \) is easier and better.
- Note that we did not really make use of the constraint function till now. That can be made use of, if in need of optimal results.
Questions to solve

- 1-9 (you may write simple matlab codes and observe the effect for each).
- 10, 11
- 18 (We haven’t gone through this, you may solve this out of interest).
- 22, 23
Reference

- Lecture Notes: Reduction of Uncorrelated Noise: Richard Alan Peters