Image Restoration and Reconstruction

Image restoration

- Objective process to improve an image, as opposed to the subjective process of image enhancement
  - Enhancement uses heuristics to improve the image for human visual system, for example, by contrast stretching
  - Restoration attempts to reverse engineer the image based on modeling the degradation process, exemplified by removal of image blur
- Recover an image by using *a priori* knowledge of degradation phenomenon
- Operations may be done in spatial (localized) or frequency (global) domain

Model of image degradation/restoration process

- Degradation process modeled as a degradation operator $\mathcal{H}$
- Use additive noise $\eta(x, y)$ and degradation function to operate on an input image $f(x, y)$ to produce a degraded image $g(x, y)$
- Figure 5.1
- Reverse engineering the process of degradation
  - Given $g(x, y)$, degradation function $\mathcal{H}$, and additive noise $\eta(x, y)$
  - Estimate $\hat{f}(x, y)$ of original image
  - Estimate should be as close to original image as possible
  - The more we know about $\mathcal{H}$ and η, the closer $\hat{f}(x, y)$ to $f(x, y)$
- Given $\mathcal{H}$ as a linear, position-invariant process, and $h(x, y)$ as its spatial representation, degraded image in spatial domain is given by
$$g(x, y) = h(x, y) \bigstar f(x, y) + \eta(x, y)$$
- The equivalent frequency domain representation is
$$G(u, v) = H(u, v)F(u, v) + N(u, v)$$

Noise models

- Noise from image acquisition and/or transmission
  - Light level and sensor temperature
  - Atmospheric disturbance during transmission
- Spatial and frequency properties of noise
  - White noise
    * Characterized by constant Fourier spectrum of noise
    * Constant Fourier spectrum implies that all frequencies are present in the function in equal proportion
    * Effectively, it must have its DC component as zero
  - Assume that the noise is independent of spatial coordinates and is uncorrelated with respect to the image
- Important noise probability density functions

- Statistical behavior of intensity values in the noise component
- Random variables characterized by a PDF
- Noise component of the model given as an image $\eta(x, y)$ of the same size an input image
- Gaussian noise or Normal noise
  * PDF of Gaussian noise is given by
$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\overline{z})^2/2\sigma^2}$$
    $-\infty < z < \infty$ is the intensity, $\overline{z}$ is the average of z, and σ is its standard deviation
  * Figure 5.2a
  * Approximately 68% of noise is in the range $[(\overline{z}-\sigma), (\overline{z}+\sigma)]$ and about 95% is in the range $[(\overline{z}-2\sigma), (\overline{z}+2\sigma)]$
  * Typically arises due to electric circuit noise and sensor noise due to poor illumination and/or high temperature
- Rayleigh noise
  * PDF of Rayleigh noise is given by
$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a \end{cases}$$
  * Mean and variance of this density are given by
$$\begin{aligned} \overline{z} &= a + \sqrt{\pi b/4} \\ \sigma^2 &= \frac{b(4-\pi)}{4} \end{aligned}$$
  * Figure 5.2b
  * Useful for approximating skewed histograms
  * Used to characterize noise in range imaging
- Erlang (gamma) noise
  * PDF of Erlang noise is given by
$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases}$$
    · $a > b$ and b is a positive integer
  * Mean and variance of this density are given by
$$\begin{aligned} \overline{z} &= \frac{b}{a} \\ \sigma^2 &= \frac{b}{a^2} \end{aligned}$$
  * Figure 5.2c
  * Observed in laser imaging
- Exponential noise
  * PDF of exponential noise is given by
$$p(z) = \begin{cases} ae^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases}$$
    · $a > 0$
  * Mean and variance of this density are given by
$$\begin{aligned} \overline{z} &= \frac{1}{a} \\ \sigma^2 &= \frac{1}{a^2} \end{aligned}$$

    * Exponential noise is a special case of Erlang noise, with $b = 1$
    * Figure 5.2d
  - Uniform noise
    * PDF of uniform noise is given by
$$p(z) = \begin{cases} \frac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$
    * Mean and variance of this density are given by
$$\begin{aligned} \overline{z} &= \frac{a+b}{2} \\ \sigma^2 &= \frac{(b-a)^2}{12} \end{aligned}$$
    * Figure 5.2e
  - Impulse (salt-and-pepper) noise
    * Number of bits/pixel in the image given by k
    * Range of possible intensity values $[0, 2^k - 1]$
    * PDF of impulse (bipolar) noise is given by
$$p(z) = \begin{cases} P_s & \text{for } z = 2^k - 1 \\ P_p & \text{for } z = 0 \\ 1 - P_s - P_p & \text{otherwise} \end{cases}$$
    * If $P_s = 0$ or $P_p = 0$, the impulse noise is called unipolar
    * If neither probability is zero, and $P_s \approx P_p$, impulse noise will resemble randomly distributed salt and pepper granules
    * Figure 5.2f
    * Let $\eta(x, y)$ denote a salt-and-pepper noise image
    * Corrupt an image $f(x, y)$ of the same size as $\eta(x, y)$ by changing all pixels in $f(x, y)$ to 0 or $2^k - 1$ to match similar valued pixels in $\eta(x, y)$; pixels corresponding to other values in $\eta(x, y)$ are left unchanged
    * Probability of a pixel to be corrupted by salt or pepper noise is $P = P_s + P_p$
      · P also known as *noise density*
    * Mean and variance of sal-and-pepper noise are given by
$$\begin{aligned} \overline{z} &= (0)P_p + K(1 - P_s - P_p) + (2^k - 1)P_s \\ \sigma^2 &= (0 - \overline{z})^2 P_p + (K - \overline{z})^2 (1 - P_s - P_p) + (2^k - 1)^2 P_s \end{aligned}$$
    * Found in situations with quick transitions, such as faulty switching during imaging
  - Noisy images and their histograms
    * Figure 5.3
      · Test pattern to illustrate the characteristics of the noise PDFs
      · Simple constant areas spanning the gray scale from black to white in three increments
    * Figure 5.4
      · Addition of six types of noise and the resulting histograms

- Periodic noise
  - Result of electrical or electromechanical interference during image acquisition
  - Spatially dependent
  - Can be reduced significantly by frequency domain filtering
  - Figure 5.5a

        * Corrupted by sinusoidal noise of various frequencies
        * Fourier transform of a pure sinusoid is a pair of conjugate impulses located at the conjugate frequencies of the sine wave

- Estimation of noise parameters
    - Periodic noise parameters estimated by inspecting Fourier spectrum of the image
        * Periodic noise tends to produce frequency spikes
        * You can attempt to infer the periodicity of noise components directly from the image but that is only possible in simplistic cases
    - Parameters of noise PDFs may be known from sensor specifications
        * Often need to estimate them for a particular imaging arrangement
        * Capture a set of images of "flat" environments
            · Uniformly illuminated solid gray board
        * Use of test patterns
    - Estimate PDF from small patches of reasonably constant background
        * Figure 5.6: Vertical strips of 150×20 pixels of gray scales (with noise) cropped from Figure 5.4
        * Calculate the mean and variance of intensity levels
            · Consider a strip denoted by S
            · Let $p_S(z_i), i = 0, 1, 2, \ldots, L-1$ be the probability estimates of pixels in S
            · The standard computation for mean and variance is

$$\begin{aligned} \overline{z} &= \sum_{i=0}^{L-1} z_i p_S(z_i) \\ \sigma^2 &= \sum_{i=0}^{L-1} (z_i - \overline{z})^2 p_S(z_i) \end{aligned}$$

        * Mean and variance are enough to characterize the Gaussian distribution
        * For other noise shapes, we solve for parameters a and b using mean and variance
        * Impulse (salt and pepper) is characterized by the peaks for black and white pixels as P_p and P_s

Restoration in the presence of noise only – spatial filtering

- When the only degradation in images is noise, we have

$$\begin{aligned} g(x, y) &= f(x, y) + \eta(x, y) \\ G(u, v) &= F(u, v) + N(u, v) \end{aligned}$$

    - Noise term is unknown, and so, cannot be simply subtracted from $g(x, y)$ or $G(u, v)$ to restore the original image
    - Periodic noise may be estimated from the spectrum of $G(u, v)$
        * In this case, it is simple to subtract $N(u, v)$ from $G(u, v)$ to obtain the original image
    - Use spatial filtering when only additive random noise is present

- Mean filters
    - Arithmetic mean filter
        * Simplest mean filter
        * Let S_{xy} be the set of coordinates in a rectangular neighborhood of size $m \times n$, centered at (x, y)

    * Compute the average value of the corrupted image $g(x,y)$ in the area defined by S_{xy}

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

    * Use a spatial filter of size $m \times n$ in which all coefficients have the value $1/mn$
    * Smooths local variations in an image by blurring it and reducing the noise
  - Geometric mean filter
    * Given by the expression

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

    * Achieves smoothing comparable to arithmetic mean filter while losing less image detail
  - Figure 5.7: Arithmetic and geometric mean filters
  - Harmonic mean filter
    * Given by the expression

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

    * Works well for salt noise but fails for pepper noise
    * Performs well for Gaussian noise as well
  - Contraharmonic mean filter
    * Given by the expression

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

      where Q is the order of the filter
    * Well suited for reducing salt and pepper noise
    * Reduces pepper noise for positive values of Q and salt noise for negative values of Q but cannot do both simultaneously
    * Reduces to arithmetic mean filter for $Q = 0$ and to harmonic filter for $Q = -1$
  - Figure 5.8: Contraharmonic filter; Q=1.5 and -1.5
  - Figure 5.9: Selecting wrong sign in contraharmonic filtering

- Order-statistic filters
  - Response based on ordering or ranking the pixel intensities in a neighborhood
  - Median filter
    * Replace the value of the pixel by the median of the intensity levels in the neighborhood of the pixel

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

    * Provide noise reduction with considerably less blurring
    * Effective in the presence of bipolar and unipolar impulse noise
    * Figure 5.10
  - Max and min filters
    * Given by

$$\begin{aligned} \hat{f}_{\max}(x,y) &= \max_{(r,c)\in S_{xy}} \{g(r,c)\} \\ \hat{f}_{\min}(x,y) &= \min_{(r,c)\in S_{xy}} \{g(r,c)\} \end{aligned}$$

    - Max filter finds the brightest points in the image; reduces pepper noise
    - Min filter finds the darkest points in the image; reduces salt noise
    - Figure 5.11
  - Midpoint filter
    - Computes the midpoint between the maximum and minimum values in the neighborhood

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

    - Combines order statistics and averaging
    - Good for randomly distributed noise, like Gaussian noise and uniform noise
  - Alpha-trimmed mean filter
    - Delete $d/2$ lowest and $d/2$ highest values in the neighborhood
    - Average the remaining $mn - d$ pixels, denoted by $g_g(r,c)$
    - Given by

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

    - d can range from 0 to $mn - 1$
    - When $d = 0$, the filter is arithmetic mean filter
    - When $d = mn - 1$, the filter is the median filter
  - Figure 5.12

- Adaptive filters
  - Change behavior based on statistical characteristics of neighborhood under the filter
  - Better performance but increase in filter complexity
  - Adaptive, local noise reduction filter
    - Mean gives a measure of average intensity in the region while variance quantifies contrast
    - Response of filter on local region S_{xy} based on four quantities
      1. $g(x,y)$ – value of noisy image at (x,y)
      2. σ_η^2 – variance of corrupting noise
      3. m_L – local mean in the neighborhood
      4. σ_L^2 – local variance in the neighborhood
    - Behavior of the filter should be
      1. No noise case: If σ_η^2 is zero, return the value of $g(x,y)$
      2. Edges: If $\sigma_L^2 \gg \sigma_\eta^2$, return a value close to $g(x,y)$
      3. Neighborhood has the same properties as overall image: if $\sigma_L^2 \approx \sigma_\eta^2$, reduce local noise by averaging
    - An adaptive expression capturing the above is:

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

    - Figure 5.13
    - Need to know the variance of overall noise σ_η^2
    - We assume that $\sigma_\eta^2 \leq \sigma_L^2$
  - Adaptive median filter
    - Can handle impulse noise with larger spatial density than very little ($P_p, P_s > 0.2$)
    - Preserves detail while smoothing nonimpulse noise; median filter unable to achieve that
    - Works with an adaptive neighborhood, by changing the size of S_{xy}

- Notation

  $z_{\min}$ Minimum intensity value in S_{xy}

  $z_{\max}$ Maximum intensity value in S_{xy}

  $z_{\mathbf{med}}$ Median of intensity values in S_{xy}

  z_{xy} Intensity value at coordinates (x, y)

  $S_{\max}$ Maximum allowed size of S_{xy}
- Works in two stages:

```
Stage A    A1 = z_med − z_min
           A2 = z_med − z_max
           if A1 > 0 && A2 < 0
               go to Stage B
           else
               increase the window size
           if window size ≤ S_max
               repeat stage A
           else
               output z_med
Stage B    B1 = z_xy − z_min
           B2 = z_xy − z_max
           if B1 > 0 && B2 < 0
               output z_xy
           else
               output z_med
```

- Three goals of algorithm
  1. Remove salt-and-pepper (impulse) noise
  2. Provide smoothing of non-impulsive noise
  3. Reduce distortion, such as excessive thickening or thinning of object boundaries
- $z_{\min}$ and $z_{\max}$ are considered to be impulse-like components
- Stage A checks whether the median filter output z_{med} is an impulse
- If $z_{\min} < z_{\text{med}} < z_{\max}$, z_{med} cannot be an impulse
  - Stage B checks if the point at the center of window z_{xy} itself is an impulse
- Adaptive median filter does not necessarily replace each point by the median, preserving detail in the process
- Effect of small value of P_p and P_s
  - As density of impulses increases, we need a larger neighborhood to clean up the noise spikes
- Figure 5.14

Periodic noise reduction by frequency domain filtering

- Periodic noise
  - Appears as concentrated bursts of energy in Fourier transform
  - At locations corresponding to frequencies of periodic interference
  - Use a selective filter to isolate the noise
- Bandreject filters
  - Ideal, Butterworth, and Gaussian bandreject filters
  - Figure 4.64
  - Remove noise in applications where the general location of noise components in frequency domain is approximately known

    * Images corrupted by additive periodic noise than can be approximated as 2D sinusoidal functions

- Bandpass filters
  - Opposite of bandreject filter
    $$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$
  - May remove too much image detail
  - Useful in isolating the effects on an image caused by selected frequency bands
- Notch filters
  - Rejects (or passes) frequencies in predefined neighborhoods about a center frequency
  - General form of notch transfer function given by
    $$H_{\text{NR}}(u,v) = \prod_{k=1}^{Q} H_k(u,v)H_{-k}(u,v)$$
  - Appear in symmetric pairs about the origin due to symmetry of Fourier transform, unless located at the origin itself
    * $H_k(u,v)$ and $H_{-k}(u,v)$ are highpass filter transfer functions with centers at (u,v) and $(-u,-v)$, respectively
    * Centers are specified with respect to the center of frequency rectangle $(\lfloor M/2 \rfloor, \lfloor N/2 \rfloor)$
  - Distance computations for the filter transfer functions given by
    $$\begin{aligned} D_k(u,v) &= \sqrt{(u - M/2 - u_k)^2 + (v - N/2 - v_k)^2} \\ D_{-k}(u,v) &= \sqrt{(u - M/2 + u_k)^2 + (v - N/2 + v_k)^2} \end{aligned}$$
  - Butterworth notch reject filter transfer function of order n with three notch pairs
    $$H_{\text{NR}} = \prod_{k=1}^{3} \left[\frac{1}{1 + [D_{0k}/D_k(u,v)]^n} \right] \left[\frac{1}{1 + [D_{0k}/D_{-k}(u,v)]^n} \right]$$
    * Since notches are symmetric pairs, the constant D_{0k} is the same for each pair, but may be different for different pairs
  - The pass filters are the opposite of reject filters
    $$H_{\text{NP}}(u,v) = 1 - H_{\text{NR}}(u,v)$$
  - Figure 5.15
    * Transfer functions for the ideal, Gaussian, and Butterworth notch reject filters with one notch pair
  - Example: Image denoising using notch filtering
    * Figure 5.16
    * Figure 5.17: Sinusoidal pattern of noise
    * Figure 5.18: Narrow rectangular notch filter
    * Figure 5.19
- Optimum notch filtering
  - Figure 5.20
    * Starlike components in Fourier spectrum due to interference
    * Several pairs of components implying multiple sinusoidal components
      · Methods like notch filter and other filters may remove too much image information
      · Also, the interference components may not be single frequency bursts

        * Interference components may have broad skirts carrying information about the interference pattern
            · Not easily detectable from the normal uniform background
    - Optimality achieved by minimizing local variances of restored estimate $\hat{f}(x,y)$
    - Isolate the principle contributions of interference pattern and then, subtract a variable, weighted portion of the pattern from the corrupted image
        * Extract principal frequency components of interference pattern
            · Use a notch pass filter $H_{\text{NP}}(u,v)$ at the location of each spike
            · Fourier transform of interference pattern given by

$$N(u,v) = H_{\text{NP}}(u,v)G(u,v)$$

        * Notch pass filter built interactively by observing the spectrum of $G(u,v)$ on a display
            · Corresponding pattern in the spatial domain obtained from the expression

$$\eta(x,y) = \mathcal{F}^{-1}\{H_{\text{NP}}(u,v)G(u,v)\}$$

        * The original image can be restored if we completely know the interference $\eta(x,y)$
        * The effect of unknown portions in the estimate of $\eta(x,y)$ can be minimized by subtracting a weighted portion of $\eta(x,y)$ from the corrupted image $g(x,y)$

$$\hat{f}(x,y) = g(x,y) - w(x,y)\eta(x,y)$$

            · $w(x,y)$ is called a weighting or modulation function
            · Select $w(x,y)$ so that the variance of $\hat{f}(x,y)$ is minimized over a specified neighborhood of every point (x,y)
        * Consider a neighborhood of size $(2a+1) \times (2b+1)$ about a point (x,y)
        * Local variance of $\hat{f}(x,y)$ at (x,y) can be estimated by

$$\sigma^2(x,y) = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \left[\hat{f}(x+s,y+t) - \overline{\hat{f}}(x,y)\right]^2$$

        * The average value of $\hat{f}$ in the neighborhood is given by

$$\overline{\hat{f}}(x,y) = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \hat{f}(x+s,y+t)$$

        * Substituting the estimate of restored image into variance gives

$$\sigma^2(x,y) = \frac{1}{(2a+1)(2b+1)} \sum_{s=-a}^{a} \sum_{t=-b}^{b} \left\{[g(x+s,y+t) - w(x+s,y+t)\eta(x+s,y+t)] - \left[\overline{g}(x,y) - \overline{w(x,y)\eta(x,y)}\right]\right\}^2$$

        * Assuming that $w(x,y)$ is essentially constant over the neighborhood gives the approximation

$$w(x+s,y+t) = w(x,y)$$

        for $-a \leq s \leq a$ and $-b \leq t \leq b$
        * This assumption also results in the expression

$$\overline{w(x,y)\eta(x,y)} = w(x,y)\overline{\eta}(x,y)$$

- The variance expression becomes

$$\sigma^2(x,y) = \frac{1}{(2a+1)(2b+1)}\sum_{s=-a}^{a}\sum_{t=-b}^{b}\left\{[g(x+s,y+t) - w(x,y)\eta(x+s,y+t)] - [\overline{g}(x,y) - w(x,y)\overline{\eta}(x,y)]\right\}^2$$

- Minimize $\sigma^2(x,y)$ by solving

$$\frac{\partial \sigma^2(x,y)}{\partial w(x,y)} = 0$$

for $w(x,y)$ yielding

$$w(x,y) = \frac{\overline{g(x,y)\eta(x,y)} - \overline{g}(x,y)\overline{\eta}(x,y)}{\overline{\eta^2}(x,y) - \overline{\eta}^2(x,y)}$$

- Since we assumed $w(x,y)$ to be constant in a neighborhood, we can compute it for just one point in each nonoverlapping neighborhood
- Figures 5.21–5.23