Noise Cleaning

Image Noise Models

1. **Most common model:** additive Gaussian noise

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

   \[ \text{underlying true image values} \]

   \[ \text{observed noisy image values} \]

   \[ n(x,y) \sim N(\mu, \sigma^2) \]

   independent of \( f \)

   \[ \text{A} \text{ Where does this arise?} \]

   **Thermal noise:** For photoelectronic detection and recording devices, have to contend with random thermal noise sources in the circuits that sense, acquire and process the signal from the detector's photoactive surface. Modeled as zero-mean additive Gaussian

   \[ \text{B} \text{ Image is a map of light intensity values... and intensity is never negative. So the noise cannot be strictly Gaussian.} \]

2. **Salt and Pepper Noise:** wide variety of processes result in the same basic degradation: only a few pixels are noisy, but they are very noisy.

   \[ \begin{align*}
   \Pr(g = f) &= 1 - \alpha \\
   \Pr(g = \text{MAX}) &= \frac{\alpha}{2} \\
   \Pr(g = \text{MIN}) &= \frac{\alpha}{2}
   \end{align*} \]

   \[ \begin{align*}
   \Pr(g = \text{LARGE}) &= \frac{\alpha}{2} \\
   \Pr(g = \text{SMALL}) &= \frac{\alpha}{2}
   \end{align*} \]
1. Where does this arise?
   1. dust/dirt on the optics
   2. electronic cameras: malfunctioning pixels
      (faulty memory locations)
   3. Suppose a grayscale image is thresholded to become binary. If some original pixels were too noisy, then in the binary version, the black regions will contain isolated white points and vice versa.
   4. Channel errors, e.g., BSC affecting the MSB
      \[10110100\] MSB flip: -128
      2nd MSB flip: +64 So error in the MSB contributes 4x as much to the MSE as next MSB

3. Quantization Noise: difference between the original and the quantized versions

\[ f \]
\[ r_{k+1} \]
\[ r_k \]
\[ r_{k-1} \]
\[ \Delta \]

Clearly not independent of signal. Deterministic function of signal. \[ g = f + n \] depends on \( f \)

Something like independence holds for high resolution quantization. The conditional mean and variance of the input given the quantization error are each approximately constant for all values of the quantization error.
So quantization noise is usually modeled as being additive uniform noise distributed between \(-\frac{\Delta}{2}\) and \(\frac{\Delta}{2}\) where \(\Delta\) is the step size.

**Noise from Counting Statistics**: uncertainty in the exact number of incident particles (photons, electrons) that arrive at the image detector to form the image. Statistical fluctuations matter. Photon arrivals modeled as a Poisson process

\[
Pr(a=k) = \frac{e^{-\lambda} \lambda^k}{k!}
\]

Property of Poisson distribution: mean = \(\lambda\) variance = \(\lambda\)

Brighter region of an image has higher \(\lambda\) \(\Rightarrow\) higher variance.

**Where does this arise?**
- Positron Emission Tomography (PET)
- Scanning Electron Microscopy (SEM)
- Radiography, extremely low light levels

**Film Grain Noise**: In the exposure and development of photographic film, silver halide grains that are exposed to a sufficient quantity of light are converted to grains of metallic silver. The silver halide grains are randomly distributed over the surface of the film. Silver halide grains experiencing an equivalent exposure are not necessarily converted to the same size/shape
of metallic silver.

Film grain noise sometimes modeled as a multiplicative noise process

\[ g(x,y) = f(x,y) \cdot n(x,y) \]
\[ = f(x,y) [1 + n(x,y)] \]
\[ = f(x,y) + f(x,y) \cdot n(x,y) \]

Additive noise here is proportional to the signal

Film grain noise sometimes modeled as additive Gaussian.

**Spatial Averaging**

Consider the additive Gaussian noise.

Common approach is spatial averaging filter

\[ H_{1} = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \]

\[ \text{cleanim} = \text{filter2}(H_{1}, \text{noisyim}) \]

1. Why is this a good idea?

Suppose the non-noisy gray levels are \( z_{1}, z_{2}, \ldots, z_{n} \)

Additive noise values \( w_{1}, w_{2}, \ldots, w_{n} \sim \mathcal{N}(0, \sigma^{2}) \)

Observed values \( z_{1} + w_{1}, z_{2} + w_{2}, \ldots \)

\[ \text{Avg} : \frac{\sum_{i=1}^{n} z_{i}}{n} + \frac{\sum_{i=1}^{n} w_{i}}{n} \]

\( \sim \text{mean zero} \)

\( \text{S.D.} \frac{\sigma}{\sqrt{n}} \)
First term = image blur
2nd term = sample of a r.v. with mean zero, \( \sigma / \sqrt{n} \) S.D.
So the noise is weakened.

2. Central Weighting

\[
H_2 = \frac{1}{10} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \\
H_3 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \\
H_4 = (\frac{1}{b+2})^2 \begin{bmatrix} 1 & b \\ b & b^2 \\ 1 & b \end{bmatrix}
\]

Effect of central weighting → less noise smoothing
→ less blurring

3. Larger arrays: 5×5, 7×7
→ more noise smoothing
→ more blurring

4. What to do about edges?

   1. Zero pad, especially if supposed to be zero there
   2. Assume image wraps around left/right, top/bottom
   3. Treat image edges as mirrors, each row duplicated beyond
   4. Leave edges unfiltered (include in output same as input)
   5. Make output image smaller
   6. Define special little asymmetrical masks for edges/corners

5. Sum elements in array, multiply by coefficient out front, get 1
   ⇒ average value unchanged
   ⇒ range \([0, 255]\) can't get larger
   (not true if some of the filter elements are negative)
6. Measuring blur of a filter:

\[
\begin{array}{llllllllllllllll}
\text{input image} & 0 & 0 & 0 & 90 & 90 & 90 & 0 & 0 & 0 & 90 & 90 & 90 \\
0 & 0 & 0 & 90 & 90 & 90 & 0 & 0 & 0 & 90 & 90 & 90 \\
0 & 0 & 0 & 90 & 90 & 90 & 0 & 0 & 0 & 90 & 90 & 90 \\
0 & 30 & 60 & 90 & 60 & 30 & 0 & 30 & 60 & 90 & 60 & 3 \times 3 \\
36 & 36 & 36 & 54 & 54 & 36 & 36 & 36 & 54 & 54 & 54 & 5 \times 5 \\
51 & 51 & 51 & 39 & 39 & 39 & 51 & 51 & 51 & 39 & 39 & 39 & 7 \times 7 \\
\end{array}
\]

Acutance: average steepness of slope

\[
0 \quad 0 \quad 0 \quad 90 \quad 90 \quad 90
\]

\[
3 \times 3 \rightarrow 0 \quad 0 \quad 30 \quad 60 \quad 90 \quad 90
\]

\[
5 \times 5 \rightarrow 0 \quad 18 \quad 36 \quad 54 \quad 72 \quad 90
\]

Resolution: can you still count the bars?

\[3 \times 3 \; \& \; 5 \times 5 \; \text{yes}\]

\[7 \times 7 \; \text{NO} \quad \text{spurious resolution}\]

3 bars get counted as 2

7. Variations to blur less:

A. Directional smoothing:

Spatial avg calculated in several directions

In output image, center pixel replaced by directional avg whose value is closest to the pixel value

\[3 \times 3 \; \text{Filter: 6 good pixels, 3 bad pixels from wrong side of the edge}\]

\[\leftarrow \text{Vertical edge, will choose vertical average}\]