Noise Reduction

- Noise typically presents itself in imagery as a random, high-frequency, uncorrelated modulation in the signal level.

- Typical methods for noise removal are:
  - Spatial filtering with a blur kernel
  - Frequency domain filtering with a lowpass filter
  - Nonlinear filtering using a statistical filter such as the median filter

- All of these methods reduce the appearance of noise in the image, but at the expense of other high-frequency information, most notably edge detail.
Traditional Approaches

Original Image

Original Image + $N(0, 10)$

Gaussian Filter ($\sigma = 5$)

Median Filter (10x10)
“Bilateral filtering smooths images while preserving edges, by means of a nonlinear combination of nearby image values. The method is noniterative, local, and simple. It combines gray levels or colors based on both their geometric closeness and their photometric similarity, and prefers near values to distant values in both domain and range. In contrast with filters that operate on the three bands of a color image separately, a bilateral filter can enforce the perceptual metric underlying the CIE-Lab color space, and smooth colors and preserve edges in a way that is tuned to human perception. Also, in contrast with standard filtering, bilateral filtering produces no phantom colors along edges in color images, and reduces phantom colors where they appear in the original image.” [Tomasi & Manduchi, 1998]
The Idea - Spatial Domain

A lowpass spatial domain filter applied to a multiband image $f(x)$ produces a multiband response image as follows

$$h(x) = \frac{1}{k_d(x)} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi)c(\xi, x)d\xi$$  \hspace{1cm} (1)$$

where $c(\xi, x)$ is a measure of geometric closeness between the neighborhood center $x$ and a nearby point $\xi$. The normalization factor $k_d(x)$ is given by

$$k_d(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, x)d\xi$$  \hspace{1cm} (2)$$

If the filter is shift-invariant, $c(\xi, x)$ is only a function of the vector difference $\xi - x$ and $k_d(x)$ is a constant.
Range (brightness) domain filtering is carried out similarly as

$$h(x) = \frac{1}{k_r(x)} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi)s(f(\xi), f(x))d\xi$$  \hspace{1cm} (3)$$

where $s(f(\xi), f(x))$ measures the photometric (brightness) similarity between the pixel at the neighborhood center and that of a nearby point $\xi$.

The normalization factor in this case is

$$k_r(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(f(\xi), f(x))d\xi$$  \hspace{1cm} (4)$$

Note that the normalization factor in the range filtering case is dependent on the image $f$ and as such nonlinear.
Bilateral filtering simultaneously combines the spatial and range domain filters

\[ h(x) = \frac{1}{k(x)} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi)c(\xi, x)s(f(\xi), f(x))d\xi \]  

(5)

where the normalization factor is

\[ k(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, x)s(f(\xi), f(x))d\xi \]  

(6)

This process replaces the pixel at \( x \) with an average of photometric/radiometric similar and nearby pixel values.
The most common implementation of bilateral filtering utilizes the shift-invariant Gaussian filter for both the closeness, \( c(\xi, x) \), and the similarity, \( s(f(\xi), f(x)) \) functions. These filters are Gaussian functions of the Euclidean distance between their arguments.

For the closeness function we have

\[
c(\xi, x) = e^{-\frac{1}{2} \left( \frac{d(\xi, x)}{\sigma_d} \right)^2} \tag{7}
\]

where \( d(\xi, x) \) is the Euclidean distance between \( x \) and \( \xi \)

\[
d(\xi, x) = d(\xi - x) = \|\xi - x\| \tag{8}
\]
Analogously we have the similarity function

\[ s(\xi, x) = e^{-\frac{1}{2} \left( \frac{\delta(f(\xi), f(x))}{\sigma_r} \right)^2} \]  

(9)

where \( \delta(\phi, f) \) is the Euclidean distance between two suitable intensity measures \( \phi \) and \( f \), namely

\[ \delta(\phi, f) = \delta(\phi - f) = \|\phi - f\| \]  

(10)

which in the greyscale image case could simply involve image intensity values.
The Gaussian Case (continued)

Current Neighborhood

Closeness Filter

\[ c(\xi, x) = e^{-\frac{1}{2} \left( \frac{d(\xi, x)}{\sigma_d} \right)^2} \]

Similarity Filter

\[ s(\xi, x) = e^{-\frac{1}{2} \left( \frac{\delta(f(\xi), f(x))}{\sigma_r} \right)^2} \]

Bilateral Filter

\[ c(\xi, x)s(\xi, x) \]

\[ \sigma_d = 5, \sigma_r = 50 \]
Bilateral Filter Visualization

Row 192
Row 128
Row 127
Bilateral Filtering

\[ \sigma_d = 1 \]
\[ \sigma_d = 3 \]
\[ \sigma_d = 5 \]
\[ \sigma_d = 10 \]
\[ \sigma_r = 10 \]
\[ \sigma_r = 30 \]
\[ \sigma_r = 50 \]
\[ \sigma_r = 100 \]
\[ \sigma_r = 300 \]
When filtering true color images, the range/similarity filter is best defined in a CIE-Lab color space where Euclidean distance is proportional to perceptible color changes.
This is a two-step process. First the RGB color triplet needs to be converted to tristimulus values as

\[
\hat{R} = \begin{cases} 
\left( \frac{R+0.055}{R^{1.055}} \right)^{2.4} & \text{if } R > 0.04045 \\
\frac{R}{12.92} & \text{otherwise}
\end{cases}
\]

\[
\hat{G} = \begin{cases} 
\left( \frac{G+0.055}{G^{1.055}} \right)^{2.4} & \text{if } G > 0.04045 \\
\frac{G}{12.92} & \text{otherwise}
\end{cases}
\]

\[
\hat{B} = \begin{cases} 
\left( \frac{B+0.055}{B^{1.055}} \right)^{2.4} & \text{if } B > 0.04045 \\
\frac{B}{12.92} & \text{otherwise}
\end{cases}
\]
and

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
= \begin{bmatrix}
0.4124564 & 0.3575761 & 0.1804375 \\
0.2126729 & 0.7151522 & 0.0721750 \\
0.0193339 & 0.1191920 & 0.9503041
\end{bmatrix}
\begin{bmatrix}
\hat{R} \\
\hat{G} \\
\hat{B}
\end{bmatrix} \tag{12}
\]
Second, the tristimulus values need to be converted to CIE-Lab space (with reference to a particular illuminant) as

\[
L = 116 f \left( \frac{Y}{Y_{\text{reference}}} \right) - 16
\]

\[
a = 500 \left[ f \left( \frac{X}{X_{\text{reference}}} \right) - f \left( \frac{Y}{Y_{\text{reference}}} \right) \right]
\]

\[
b = 200 \left[ f \left( \frac{Y}{Y_{\text{reference}}} \right) - f \left( \frac{Z}{Z_{\text{reference}}} \right) \right]
\]

where

\[
f(x) = \begin{cases} 
\frac{x^{1/3}}{3} & \text{if } x > 0.008856 \\
7.787x + \frac{16}{116} & \text{otherwise}
\end{cases}
\]
and the white reference tristimulus values for several different standard illuminants are given in the following table

<table>
<thead>
<tr>
<th>Illuminant</th>
<th>$X_{reference}$</th>
<th>$Y_{reference}$</th>
<th>$Z_{reference}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>D65 (daylight)*</td>
<td>95.0430</td>
<td>100.0000</td>
<td>108.8801</td>
</tr>
<tr>
<td>A (tungsten)</td>
<td>109.8490</td>
<td>100.0000</td>
<td>35.5825</td>
</tr>
<tr>
<td>F2 (cool white fluorescent)</td>
<td>99.1858</td>
<td>100.0000</td>
<td>67.3938</td>
</tr>
<tr>
<td>F11 (narrow band fluorescent)</td>
<td>100.9610</td>
<td>100.0000</td>
<td>64.3506</td>
</tr>
<tr>
<td>F7 (daylight fluorescent)</td>
<td>95.0416</td>
<td>100.0000</td>
<td>108.7489</td>
</tr>
<tr>
<td>F8 (daylight fluorescent)</td>
<td>96.4274</td>
<td>100.0000</td>
<td>82.4211</td>
</tr>
</tbody>
</table>

* typically used for RGB values on an standard display
Bilateral Filtering

Example

Original

\[ \sigma_d = 10, \sigma_r = 1 \text{ (CIE L*a*b*)} \]
Example - Iterative Application

Original

\[ \sigma_d = [5, 5, 5], \sigma_r = [1, 1, 1] \text{ (CIE L*a*b*)} \]