Digital Image Processing

Spatial Filters in Image Processing

DR TANIA STATHAKI
READER (ASSOCIATE PROFESSOR) IN SIGNAL PROCESSING
IMPERIAL COLLEGE LONDON
Spatial filters called spatial masks are used for specific operations on the image. Popular filters are the following:

- Low pass filters
- High boost filters
- High pass filters
Image averaging in the case of many realizations

- We have $M$ different noisy images:
  \[ g_i(x, y) = f(x, y) + n_i(x, y), \ i = 0, ..., M - 1 \]
- Noise realizations are zero mean and white with the same variance, i.e.,
  \[ E\{n_i(x, y)\} = 0 \text{ and } R_{n_i}[k, l] = \sigma_{n_i}^2 \delta[k, l] = \sigma_n^2 \delta[k, l] \]
- We define a new image which is the average
  \[ \bar{g}(x, y) = \frac{1}{M} \sum_{i=0}^{M-1} g_i(x, y) = f(x, y) + \frac{1}{M} \sum_{i=0}^{M-1} n_i(x, y) \]
- Notice that average is calculated across realizations.
- **Problem:** Find the mean and variance of the new image $\bar{g}(x, y)$.
  \[ E\{\bar{g}(x, y)\} = E\{\frac{1}{M} \sum_{i=0}^{M-1} g_i(x, y)\} = \frac{1}{M} \sum_{i=0}^{M-1} E\{g_i(x, y)\} = f(x, y) \]
  \[ \sigma_{\bar{g}(x,y)}^2 = \sigma_{f(x,y)}^2 + \frac{1}{M} \sigma_{n(x,y)}^2 = \frac{1}{M} \sigma_{n(x,y)}^2 \]
Example: Image Averaging (number refers to realizations)
Example: Image Averaging

Noise-free Image
Noise Variance = 0.05

$M = 2$
$M = 5$
$M = 10$

$M = 25$
$M = 50$
$M = 100$
Spatial masks

Array A

Array B

\[
\begin{array}{ccc}
\text{w}_1 & \text{w}_2 & \text{w}_3 \\
\text{w}_4 & \text{w}_5 & \text{w}_6 \\
\text{w}_7 & \text{w}_8 & \text{w}_9 \\
\end{array}
\]
Local averaging spatial masks for image de-noising (smoothing)

\[
\frac{1}{9} \times \begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{bmatrix}
\quad \frac{1}{25} \times \begin{bmatrix}
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
\end{bmatrix}
\quad \frac{1}{49} \times \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\end{bmatrix}
\]
Response of 1D and 2D local averaging spatial masks.
Other (weighted) local averaging spatial masks

![Image](image-url)

- Solid line shows weighted mask
- Dashed line shows simple averaging
Example of local image averaging

Original Image  Noisy Image  3 × 3 smoothing  5 × 5 smoothing
Example: Image Averaging

original

n=5

n=15

n=3

n=7

n=25
A noiseless image and its noisy version

Original Lena image

Lena image with 10db noise
And some denoising!

3 × 3 averaging mask  
5 × 5 averaging mask
Median filtering

Neighbourhood values:
115, 119, 120, 123, 124, 125, 126, 127, 150

Median value: 124
Example of median filtering

Original

Original with Noise

After N=3 Median Filter

After N=5 Median Filter
Comparison of local smoothing and median filtering

Original Image

Result of 5x5 neighbourhood averaging

Image corrupted by impulse noise

Result of 5x5 median filtering
Median versus local averaging filter response around local step edge
Median versus local averaging filter response around local ridge edge
High pass filtering

- We use masks with positive coefficients around the centre of the mask and negative coefficients in the periphery.
- In the mask shown below the central coefficient is +8 and its 8 nearest neighbours are -1.
- The reversed signs are equally valid, i.e., -8 in the middle and +1 for the rest of the coefficients.
- The sum of the mask coefficients must be zero. This leads to the response of the mask within a constant (or slowly varying) intensity area being zero (or very small).

\[
\begin{align*}
\frac{1}{9} \times & \begin{pmatrix}
-1 & -1 & -1 \\
-1 & 8 & -1 \\
-1 & -1 & -1 \\
\end{pmatrix}
\end{align*}
\]
Example of high pass filtering
Example of high pass filtering

High pass filtered image

High pass filtered image with 10db noise
The goal of high boost filtering is to enhance the high frequency information without completely eliminating the background of the image.

We know that:

\[(\text{High-pass filtered image}) = (\text{Original image}) - (\text{Low-pass filtered image})\]

We define:

\[(\text{High boost filtered image}) = A \times (\text{Original image}) - (\text{Low-pass filtered image})\]

\[(\text{High boost}) = (A - 1) \times (\text{Original}) + (\text{Original}) - (\text{Low-pass})\]

\[(\text{High boost}) = (A - 1) \times (\text{Original}) + (\text{High-pass})\]

As you can see, when \( A > 1 \), part of the original is added back to the high-pass filtered version of the image in order to partly restore the low frequency components that would have been eliminated with standard high-pass filtering.

Typical values for \( A \) are values slightly higher than 1, as for example 1.15, 1.2, etc.
The resulting image looks similar to the original image with some edge enhancement.

The spatial mask that implements the high boost filtering algorithm is shown below.

The resulting image depends on the choice of $A$.

High boost filtering is used in printing and publishing industry.
Example of high boost filtering
Example of high boost filtering

\[ A = 1.15 \]

\[ A = 1.2 \]
Edges are abrupt local changes in the image value (for example grey level).

Edges are of crucial important because they are related to the boundaries of the various objects present in the image.

Object boundaries are key feature for object detection and identification.

Edge information in an image is found by looking at the relationship a pixel has with its neighborhoods.

If a pixel’s grey-level value is similar to those around it, there is probably **not** an edge at that point.

If a pixel’s has neighbors with widely varying grey levels, it may present an edge point.
Abrupt versus gradual change in image intensity
Type of edges

- **Step edges**
  - [Diagram]
  - [Diagram]
  - [Diagram]

- **Roof edge**
  - [Diagram]

- **Line (ridge) edges**
  - [Diagram]
Edges are caused by a variety of factors:

- Surface normal discontinuity
- Depth discontinuity
- Surface color discontinuity
- Illumination discontinuity
From a grey level image to an edge image (or edge map)
How do we detect the presence of a local edge?

- The gradient (first derivative) of the image around a pixel might give information about how detailed is the area around a pixel and whether there are abrupt intensity changes.

- The image is a 2-D signal and therefore, the gradient at a location \((x, y)\) is a 2-D vector which contains the two partial derivatives of the image with respect to the coordinates \(x, y\).

\[
\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}
\]

- Edge strength is given by \(\|\nabla f\| = \left(\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right)^{\frac{1}{2}}\) or \(\|\nabla f\| \approx \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|\).

- Edge orientation is given by \(\theta = \tan^{-1}\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right)\).
• Edge strength is given by $\|\nabla f\| = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 \frac{1}{2}$ or $\|\nabla f\| \approx \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|$. 

• Edge orientation is given by $\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$.
Spatial masks for edge detection

- Consider an image region of size $3 \times 3$ pixels. The coordinates $x, y$ are shown.
- The quantity $r$ denotes grey level values.
- The magnitude $\| \nabla f \| = \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right|$ of the gradient at pixel with intensity $r_5$ can be approximated by:
  - Differences along $x$ and $y$
    $$\| \nabla f \| = |r_5 - r_6| + |r_5 - r_8|$$
  - Cross-differences (along the two main diagonals)
    $$\| \nabla f \| = |r_6 - r_8| + |r_5 - r_9|$$

<table>
<thead>
<tr>
<th></th>
<th>$r_1$</th>
<th>$r_2$</th>
<th>$r_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_4$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$r_7$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$r_5$</td>
<td>$r_6$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$r_8$</td>
<td>$r_9$</td>
<td></td>
</tr>
</tbody>
</table>

$x$}

$y$
The magnitude \( \| \nabla f \| = |\frac{\partial f}{\partial x}| + |\frac{\partial f}{\partial y}| \) of the gradient at pixel with intensity \( r_5 \) can be approximated by:

- differences along \( x \) and \( y \)
  \[ \| \nabla f \| = |r_5 - r_6| + |r_5 - r_8| \]
- cross-differences (along the two main diagonals)
  \[ \| \nabla f \| = |r_6 - r_8| + |r_5 - r_9| \]

Each of the above approximation is the sum of the magnitudes of the responses of two masks. These sets of masks are the so called **Roberts operator**.
Spatial masks for edge detection – Prewitt and Sobel operators

- The magnitude $\|\nabla f\| = \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right|$ of the gradient at pixel with intensity $r_5$ can be also approximated by involving more pixels:
  - Roberts operator
    $$\|\nabla f\| = |(r_3 + r_6 + r_9) - (r_1 + r_4 + r_7)| + |(r_7 + r_8 + r_9) - (r_1 + r_2 + r_3)|$$
  - Sobel operator
    $$\|\nabla f\| = |(r_3 + 2r_6 + r_9) - (r_1 + 2r_4 + r_7)| + |(r_7 + 2r_8 + r_9) - (r_1 + 2r_2 + r_3)|$$
- Each of the above approximation is the sum of the magnitudes of the responses of two masks.

\[
\begin{array}{ccc}
-1 & 0 & 1 \\
-1 & 0 & 1 \\
-1 & 0 & 1 \\
\end{array} + \begin{array}{ccc}
-1 & -1 & -1 \\
0 & 0 & 0 \\
1 & 1 & 1 \\
\end{array} \quad \begin{array}{ccc}
-1 & -2 & -1 \\
0 & 0 & 0 \\
1 & 2 & 1 \\
\end{array} + \begin{array}{ccc}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1 \\
\end{array}
\]
Example of edge detection using Prewitt
Example of edge detection using Prewitt
Example of edge detection using Sobel
The second order gradient gives also information about how detailed is the area around a pixel.

It is defined as:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

A mask that can approximate the above function is the Laplacian:

$$\nabla^2 f \approx 4r_5 - (r_2 + r_4 + r_6 + r_8)$$