CV - HW2 Theory

Amirhossein Izadi - 99243018

Q1.

Applying two 3x3 average filters consecutively is not same as applying a single 9x9 average filter. basicly applying two 3x3 box filter results consecutively is equevalent to applying following 5x5 filter and dimensions are different from 9x9 average filter.

Let's go through the details.The 3x3 mean filter (M) is defined as:

$$M = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

When you convolve M with itself (M * M), you are essentially taking the discrete convolution of the two matrices. The general formula for the discrete convolution of two matrices A and B is given by:

$$(A * B)_{i,j} = \sum_m \sum_n A_{m,n} \cdot B_{i-m,j-n}$$

and In the case of convolving a 3x3 matrix with itself, the resulting matrix will have dimensions

$$(3+3-1) \times (3+3-1) = 5 \times 5$$

```
import numpy as np
from scipy.signal import convolve2d

def mean_filter_3x3():
    return np.ones((3, 3))

def equivalent_filter_3x3_mean_twice():
    mean_filter = mean_filter_3x3()
    equivalent_filter = convolve2d(mean_filter, mean_filter)
    return equivalent_filter

print(equivalent_filter_3x3_mean_twice())
```

and The result as you can see below is some how a weighted average filter and its not box filter.

$$M * M = \frac{1}{81}\begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 9 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

Figure 1.1: original image

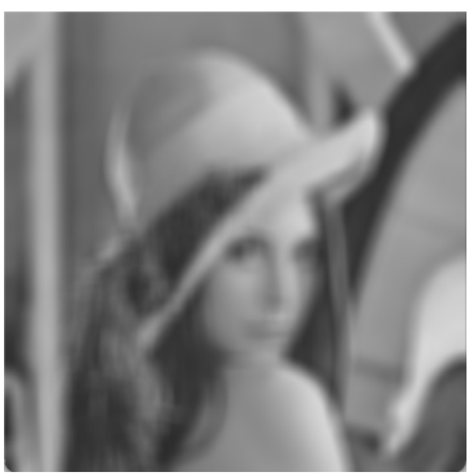


Figure 1.2: applying 9x9 box filter

Figure 1.2: applying two 3x3 box filter consecutively

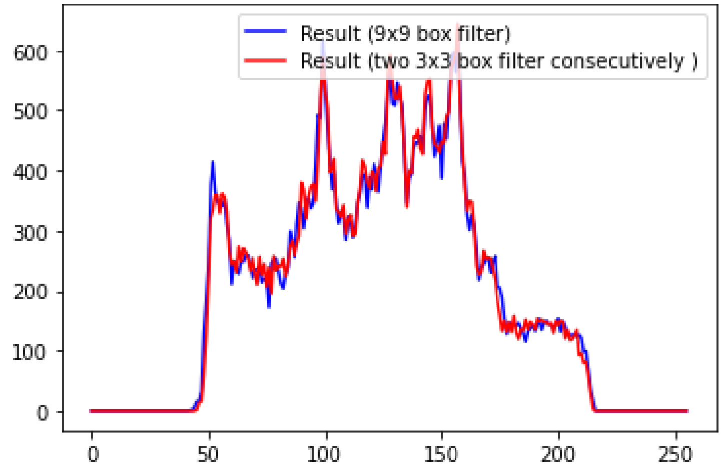


Figure 1.4: histograms of the two resulting images

Q2.

The Laplacian kernel is a convolution kernel commonly used for edge detection and enhancing high-frequency components in an image. The Laplacian operator is defined as the sum of the second derivatives of the image intensity. When this Laplacian kernel is convolved with an image, the resulting image highlights regions where the intensity changes rapidly, such as edges. The negative weight in the central element is crucial for emphasizing the edges, as it causes a strong response when there is a significant difference in intensity between the central pixel and its neighbors.

lets discuss abut each of these two filters $kernel_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -9 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ and

$$kernel_2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

kernel1 is is an modified version of a Laplacian kernel that is designed to highlight regions of rapid intensity changes in an image (edge extraction). it miss some weak edges respect to center weight.

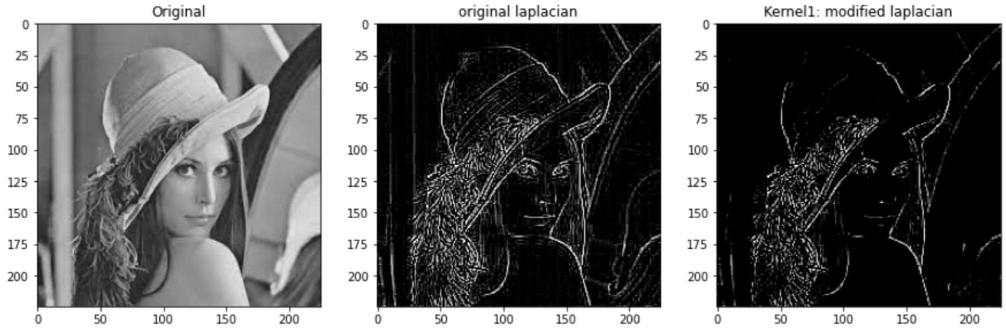


Figure 2.1: original laplacian vs modified laplacian

kernel2 is result Summation/Subtraction of $laplacian = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -9 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ and/to

$$image = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

its sharpern the original image by adding edges to original image.

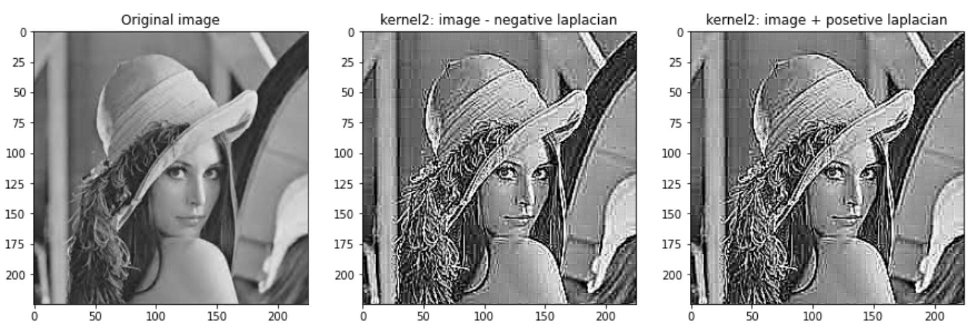


Figure 2.2: kernel_2 result

As you can see above kernel1 extract edges from an image and removing the flat area and background but kernel2 sharpen the image and do not delete the flat area.

Q3.

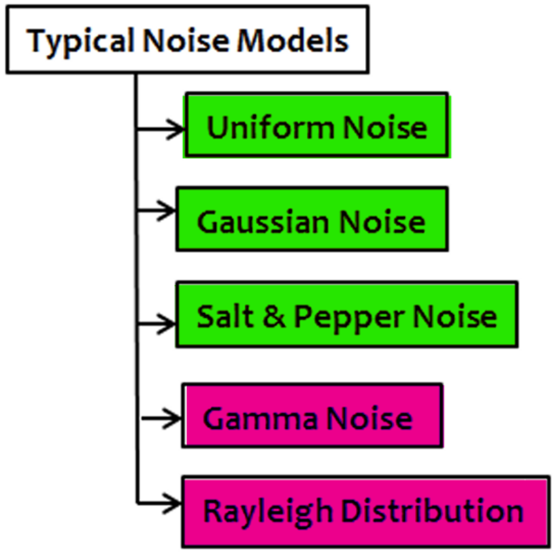
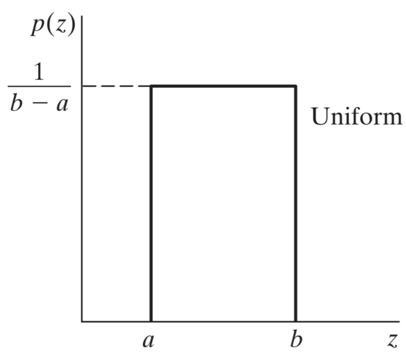


Figure 3.1: Typical noises in DIP (image from [IJETER](#) **)**

Given the materials taught, we just discuss about green Noise models.

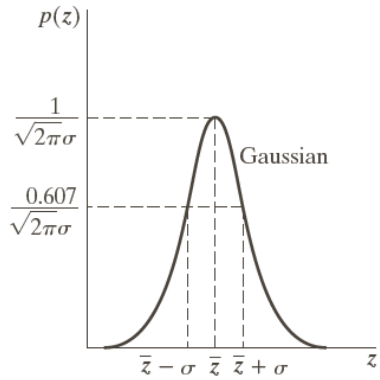
- **Uniform Noise:**

$$p(z) = \begin{cases} \dfrac{1}{b-a} & \text{if } a \le z \le b \\ 0 & \text{otherwise} \end{cases}$$



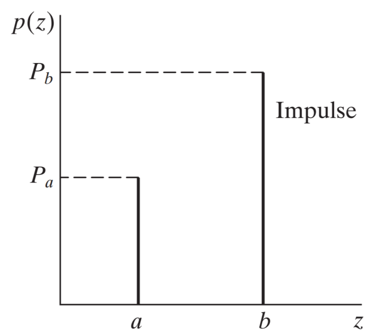
- **Gussain Noise:**

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\bar{z})^2/2\sigma^2}$$



- **Salt & Pepper Noise:**

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$



Q4.

1. Wavelet Filters:
   - Concept: Wavelet analysis involves decomposing a signal or an image into its constituent wavelet components. Wavelets are functions that are localized in both time and frequency domains, allowing them
to capture both high and low-frequency information.

2. Laplacian Filters:
   - Concept: The Laplacian filter is based on the Laplacian operator, which is a second-order derivative. It measures the rate of change of intensity in an image. In the context of image processing, it highlights regions of rapid intensity change, such as edges.

3. Fourier Filters:
   - Concept: Fourier analysis involves decomposing a signal or an image into its sinusoidal components using the Fourier transform. The resulting frequency spectrum represents the distribution

of different frequency components in the signal or image.

Q5.

The Butterworth low-pass filter order influences its performance characteristics. Lower-order filters have gentler roll-offs with minimal ringing, making them suitable for applications where smooth transitions are crucial. Second-order filters provide a balance between roll-off steepness and ringing. As the order increases beyond 20, the roll-off becomes ultra-steep, resembling an ideal filter, but there's a trade-off with pronounced ringing. Careful consideration is necessary when selecting the filter order to meet the specific requirements of an application, balancing the need for a sharp roll-off with the tolerance for potential distortion introduced by ringing.

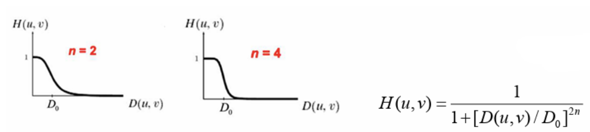


Figure 5.1: BLPF in various orders

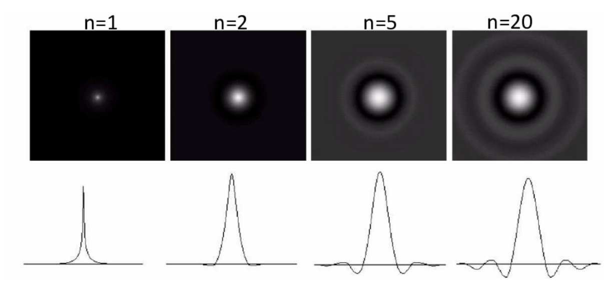


Figure 5.2: BLPF spatial representation in various orders and $D_0 = 5$

Q6.

Low pass filter: Low pass filter is the type of frequency domain filter that is used for smoothing the image. It attenuates the high-frequency components and preserves the low-frequency components.

High pass filter: High pass filter is the type of frequency domain filter that is used for sharpening the image. It attenuates the low-frequency components and preserves the high-frequency components.

Low pass filter:

- Used for smoothing the image.
- Attenuates high frequencies.
- Preserves low frequencies.
- Allows frequencies below the cutoff frequency to pass through.
- Consists of a resistor followed by a capacitor.
- Helps in the removal of aliasing effects.
- Mathematical representation: $G(u, v) = H(u, v) . F(u, v)$

High pass filter:

- Used for sharpening the image.
- Attenuates low frequencies.
- Preserves high frequencies.
- Allows frequencies above the cutoff frequency to pass through.
- Consists of a capacitor followed by a resistor.
- Helps in the removal of noise.
- Mathematical representation: $H(u, v) = 1 - H'(u, v)$

[answer source.]

Q7.

An ideal edge is a theoretical concept used in image processing and computer vision to represent a boundary between two distinct regions or objects in an image. The characteristics of an ideal edge include:

1. Infinitesimal Thickness: An ideal edge is considered to have zero thickness. It is conceptualized as a mathematical transition from one intensity level to another, occurring instantaneously at a specific position.

2. Infinite Intensity Contrast: The transition across an ideal edge is assumed to be abrupt and instantaneous, with a perfect and infinite contrast between the two sides. This means that the pixel intensities change sharply from one value to another.

3. Spatial Localization: Ideal edges are precisely located in the image, and their positions are well-defined. The transition between the two regions occurs at a specific pixel or sub-pixel location.

4. No Noise or Disturbances: Ideal edges are free from any noise or disturbances that might affect real-world images. In practical scenarios, images are often corrupted by noise, making the identification of ideal edges challenging.

In contrast to the ideal edge, a real edge refers to the edges observed in actual images captured by cameras or other imaging devices. Real edges exhibit several characteristics that deviate from the idealized concept:

1. Finite Thickness: Real edges have a finite thickness due to the limitations of imaging systems and the physical properties of objects. The transition between regions is not instantaneous but occurs over a certain spatial extent.

2. Limited Intensity Contrast: The intensity contrast across a real edge is not infinite. It may be affected by factors such as lighting conditions, material properties, and the characteristics of the imaging sensor, resulting in a more gradual intensity transition.

3. Localization Uncertainty: The precise localization of a real edge is subject to uncertainty. Factors like noise, blurring, and sampling limitations can introduce variations in the perceived position of an edge.

4. Noise and Disturbances: Real-world images are susceptible to noise, artifacts, and other disturbances. These imperfections can affect the clarity and accuracy of edge detection in actual images

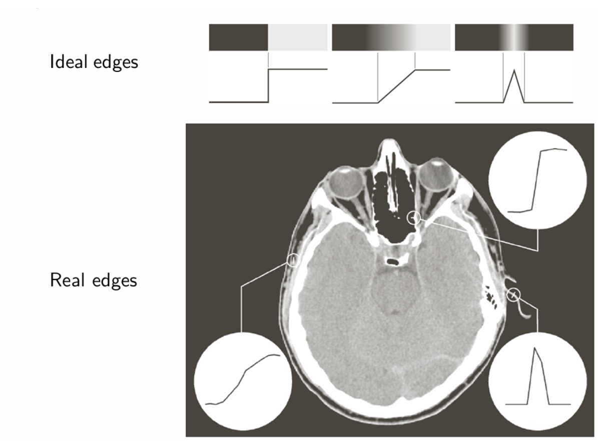


Figure 7.1: real edge versus ideal edge

Q8.

Ideal High-pass Filter:

-Frequency Response: An ideal high-pass filter allows high-frequency components to pass through while attenuating low-frequency components. It has a sharp cutoff at a specified frequency.
-Cutoff Frequency: The cutoff frequency is well-defined, and frequencies above this value are passed, while frequencies below it are attenuated.
-Transition Band: The transition between the passband and stopband is abrupt, resulting in a perfect separation of high and low frequencies.
-Ripple and Roll-off: There is no ripple in the passband, and the roll-off is very steep.

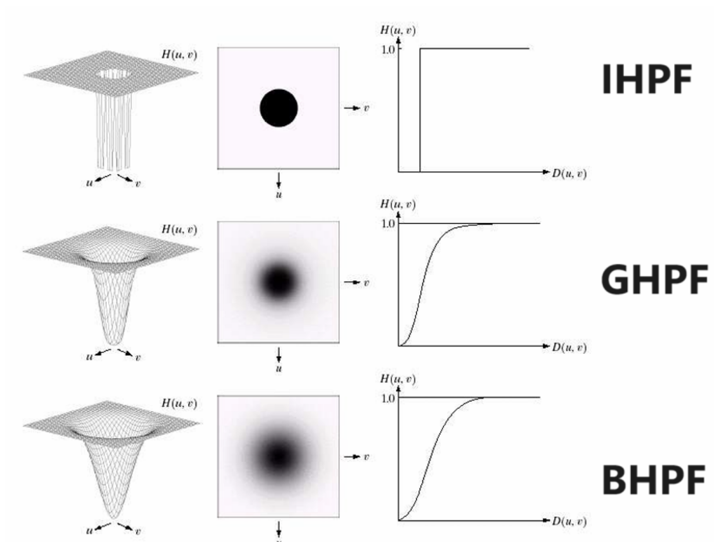
Gaussian Filter:

- Response: Gaussian filters have a smooth and bell-shaped frequency response.

-Cutoff Frequency: Unlike ideal filters, Gaussian filters do not have a well-defined cutoff frequency. Instead, the cutoff is often defined as the point at which the amplitude drops to a certain percentage of its peak value.
-Transition Band: The transition between the passband and stopband is gradual and smooth.
-Ripple and Roll-off: There is no ripple in the passband, and the roll-off is not as steep as that of an ideal filter.

Butterworth Filter:

-Frequency Response: Butterworth filters have a maximally flat frequency response in the passband, resulting in a more natural and gradual transition between the passband and stopband.
-Cutoff Frequency: The cutoff frequency is well-defined, similar to the ideal filter, but the transition is not abrupt.
-Transition Band: The transition band is smoother compared to an ideal filter but steeper than a Gaussian filter.
- Ripple and Roll-off: There is no ripple in the passband, and the roll-off is not as steep as that of an ideal filter but is generally better than a Gaussian filter.



Q9.

Here are some common methods:

1. Feature-Based Matching:

This method involves detecting key features (like edges, corners, or interesting points) in both images and then finding matches between these features. Algorithms like SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features), and ORB (Oriented FAST and Rotated BRIEF) are commonly used for feature detection and description.

Process:
- Detect features in both images.
- Compute descriptors (feature vectors) for each feature.
- Match features between the two images based on the similarity of their descriptors.
- Optionally, use a geometric transformation to align the images based on the matched features.

2. Template Matching:
This technique is used when one image is a small part (template) of another larger image. It involves sliding the template image over the larger image to find the area with the highest correlation.

Process:
- Use correlation methods to compare the template with each region of the larger image.
- The region with the highest correlation score is considered the match.

3. Histogram Comparison:
This method is suitable for comparing the overall color or intensity distribution of images.

Process:
- Compute histograms for each image (color histograms or grayscale histograms).
- Compare these histograms using measures like Chi-Square, Correlation, or Bhattacharyya distance.

4. Deep Learning Methods:
Convolutional Neural Networks (CNNs) can be trained to understand and compare the content of images.

Process:
- Use a pre-trained CNN model to extract features from both images.
- Compare the extracted features using a similarity metric (like cosine similarity).
- For more advanced applications, use Siamese networks specifically designed for image comparisons.

5. Structural Similarity Index (SSIM):
This method is used for measuring the similarity between two images. It considers changes in texture, luminance, and contrast.

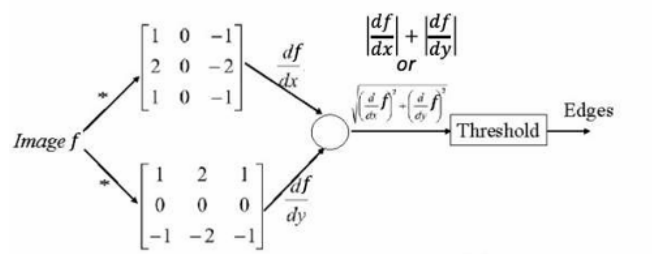
Process:
- Compute the SSIM index between the two images.
- The SSIM index ranges from -1 to 1, where 1 indicates perfect similarity.

Q10.

(I) Edge detections:

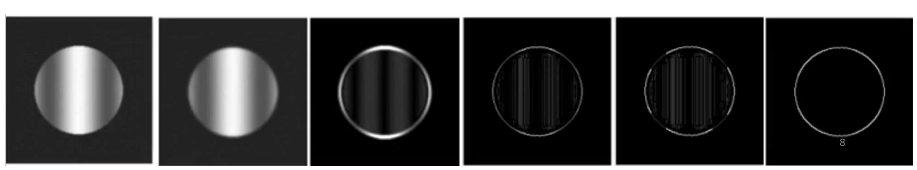
Sobel Edge Detection:

Sobel edge detection is a popular method for gradient-based edge detection. It employs convolution with Sobel operators, typically a 3x3 kernel, to approximate the gradient of the image intensity in both the horizontal and vertical directions. The gradients are then combined to obtain the magnitude and direction of the edges. Sobel operators are effective in highlighting edges and are computationally efficient. However, they may be sensitive to noise and might miss some subtle edges due to their simplicity.



Canny Edge Detection:

The Canny edge detector is a multi-stage algorithm widely used for its robust performance. It includes Gaussian smoothing to reduce noise, gradient calculation using Sobel operators, non-maximum suppression to thin edges, and hysteresis thresholding to eliminate weak edges. Canny edge detection excels in accurately detecting edges, and the two threshold values allow for flexibility in edge selection. However, its multi-stage nature makes it computationally more expensive compared to simpler methods.



Prewitt Edge Detection:

Prewitt edge detection is similar to Sobel but uses a different convolution kernel. It computes the gradient by convolving the image with Prewitt operators, which are 3x3 matrices. Like Sobel, Prewitt is sensitive to edge direction and is effective in detecting edges with a clear orientation. However, it may also suffer from noise-related issues, and the choice between Sobel and Prewitt often depends on the specific characteristics of the edges in the image.

Written in separable form

Kernel in x direction: $\begin{bmatrix} -1 & 0 & 1 \\ -1 & \mathbf{0} & 1 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ \mathbf{1} \\ 1 \end{bmatrix} * \begin{bmatrix} -1 & \mathbf{0} & 1 \end{bmatrix}$

Kernel in y direction: $\begin{bmatrix} -1 & -1 & -1 \\ 0 & \mathbf{0} & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{1} & 1 \end{bmatrix} * \begin{bmatrix} -1 \\ \mathbf{0} \\ 1 \end{bmatrix}$

Laplacian of Gaussian (LoG) Edge Detection:

The Laplacian of Gaussian is an edge detection method that combines Gaussian smoothing with the Laplacian operator. First, the image is convolved with a Gaussian filter to reduce noise. Then, the Laplacian operator is applied to enhance regions of rapid intensity change, highlighting edges. The LoG method is effective in detecting edges and provides a more isotropic response compared to gradient-based methods. However, it can be computationally expensive due to the two-stage process.

$$LoG\ kernel\ : \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

(I) corner detections:

Harris Corner Detection:

The Harris corner detection algorithm is a widely used method for identifying corners in an image. It operates by analyzing the variations in intensity for small shifts of a window across the image. The algorithm considers a local autocorrelation matrix for each pixel and evaluates its eigenvalues. Corners are characterized by having two large eigenvalues in the presence of a corner structure. The Harris response function, often denoted as R, is defined based on these eigenvalues. High values of R indicate the presence of corners. The Harris corner detector is robust to changes in lighting conditions and provides accurate corner localization. However, it may be sensitive to noise and is computationally more expensive due to the computation of eigenvalues.

$$R = det(M) - k \times trace(M)$$

M is the autocorrelation matrix for each pixel, and k is an empirically determined constant

Moravec Corner Detection:

Moravec corner detection is an early approach to corner detection that relies on comparing the intensity variations in different directions. It computes a simple measure based on the sum of squared differences between the intensities of a pixel and its neighbors for various shifts. The minimum value across these shifts is used to identify corners. Moravec's method is computationally less intensive compared to Harris, but it tends to be less robust in the presence of noise and may produce false positives. It is a straightforward and conceptually simple method for detecting corners.

$$E(u,v) = \sum_{\delta x,\delta y} [\ I(x+\delta x, y+\delta y) - I(x,y)\ ]^2$$

$I(x, y)$ is the intensity at pixel (x, y) and the sum is over shifts $(\delta x, \delta y)$ in different directions..

Q11.

In image processing, erosion and dilation are fundamental morphological operations used to modify and enhance the structures within an image. Erosion is a process that involves the shrinking or wearing away of boundaries in an image. It is achieved by convolving the image with a structuring element, typically a small binary matrix, and updating each pixel in the image to the minimum value within the corresponding neighborhood defined by the structuring element. Erosion is particularly useful for tasks such as noise reduction, separating connected objects, and refining the boundaries of distinct regions. It essentially removes pixels from the edges of objects, making them thinner and more well-defined.

On the other hand, dilation is the counterpart to erosion and involves expanding or thickening the boundaries of objects in an image. Similar to erosion, dilation is performed by convolving the image with a structuring element, but in this case, each pixel in the image is updated to the maximum value within the neighborhood defined by the structuring element. Dilation is often employed to close gaps between objects, connect broken structures, and increase the overall size of objects within an image. Together, erosion and dilation are powerful tools in morphological image processing, providing a means to manipulate the shapes and sizes of objects, enhance or suppress certain features, and prepare images for subsequent analysis or recognition tasks.

$$A \oplus B = \left\{ x \,\middle|\, \left(\hat{B}\right)_x \cap A \neq \emptyset \right\}$$

Figure 11.1: Dilation

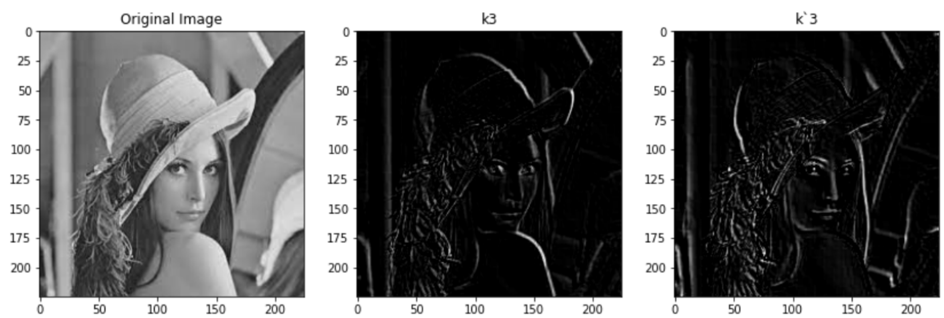
$$A \ominus B = \{x \mid (B)_x \subseteq A\}$$

Figure 11.2: Erosion

Q12.

in first case lets clarify the difference between each filter and it's dual by an example of kernel3

$$k_3 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad k_3' = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$



as you can see:

k_3 extract *light to dark* $\nearrow$ *direction* edges.
in contrast k_3' extract *dark to light* $\nearrow$ *direction* edges.

and for the rest kernels we have:

Kernel1 : $\begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ it detects edges in $\searrow$ *light to dark* as you can see

Kernel2: $\begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ it detects edges in $\downarrow$ *light to dark* as you can see

Kernel4: $\begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ it detects edges in $\longrightarrow$ *light to dark* as you can see :

all results thogheter :