BLG 513E

Image Processing Homework-1

Kemal Alperen Çetiner

Edge in image corresponds to an “edge like” region where there exists a significant change in intensity, color or texture. Edge detection is a computer vision task by which we try to locate areas where edges occur. Edges are of great significance since they represent important information about the structure and content of an image. They lay the stepping stone for various image processing applications such as object recognition, object tracking and image segmentation. There are several methods to obtain edges in an image. Edge detection techniques can be divided into two main categories namely gradient based methods and model based methods. Model based methods endeavor to fit a structure or a model to the image to be able to find edges. Hough Transform where one tries to parameterize curves and objects to find correlations between such curve and the image. This includes working in a parameter space and voting mechanism to find if such curve exists. For the gradient based methods, main approach is to process the image and its derivative image to detect edges in an image. This report examines a gradient based method, Canny Edge Detector. Steps and implementation details of Canny Edge Detector is demonstrated.

Figure 1: Edge Detection

Smoothing: Since images are a result of sampling and discretization of a real world signal, it is expected to have noises in images. Source of the noise can vary from sensor related reasons to environmental noises. Therefore, it is important apply a smoothing operation to remove high-frequency noise or rapid intensity variations while preserving the essential structures and features of an image. In this assignment Gaussian Smoothing is employed and implemented. It is an approximation of Gaussian distribution.

1/16

1	2	1
2	4	2
1	2	1

1/273

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

1/1003

0	0	1	2	1	0	0
0	3	13	22	13	3	0
1	13	59	97	59	13	1
2	22	97	159	97	22	2
1	13	59	97	59	13	1
0	3	13	22	13	3	0
0	0	1	2	1	0	0

Figure 2: Different Scales of Gaussian Kernel

A generic function to output a Gaussian kernel of single dimension is implemented. This is due to the fact that Gaussian kernel is a separable filter which means that it can be implemented by two 1D convolutions along x and y axis. The kernels given in the figure 1 are generated from Gaussian distributions with different standard deviations. Since Gaussian distribution has infinite support, it is truncated at a certain level. This results in kernel size being dependent on standard deviation of the underlying distribution.

Figure 3: Smoothed Image

<u>Finding Gradients:</u> As an image is a discrete signal, the gradient is defined by the difference operation. This can be thought of as an approximation of the derivative operation.

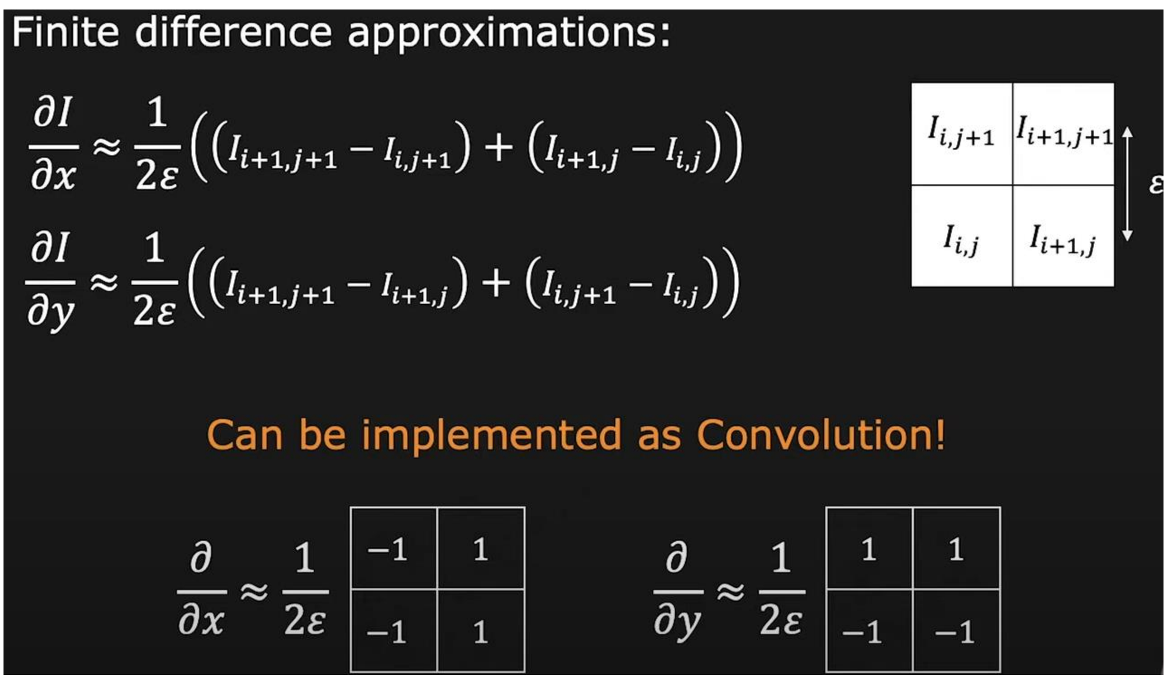


Figure 4: Gradient Formula

However, a smoothed and not-pixel shifting version of this kernel is employed usually. This is because, derivative operation can be thought as a scaling operation at local level. This means that if there is a noisy the resulting image will be noisier. Moreover, even sized kernels shifts the image which requires post-adjustments to return the image to its original coordinates.

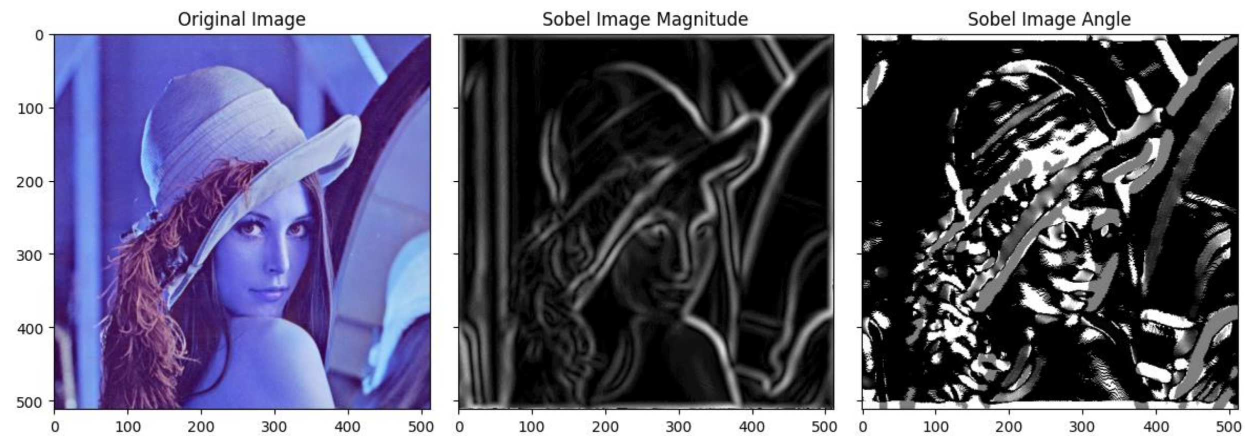
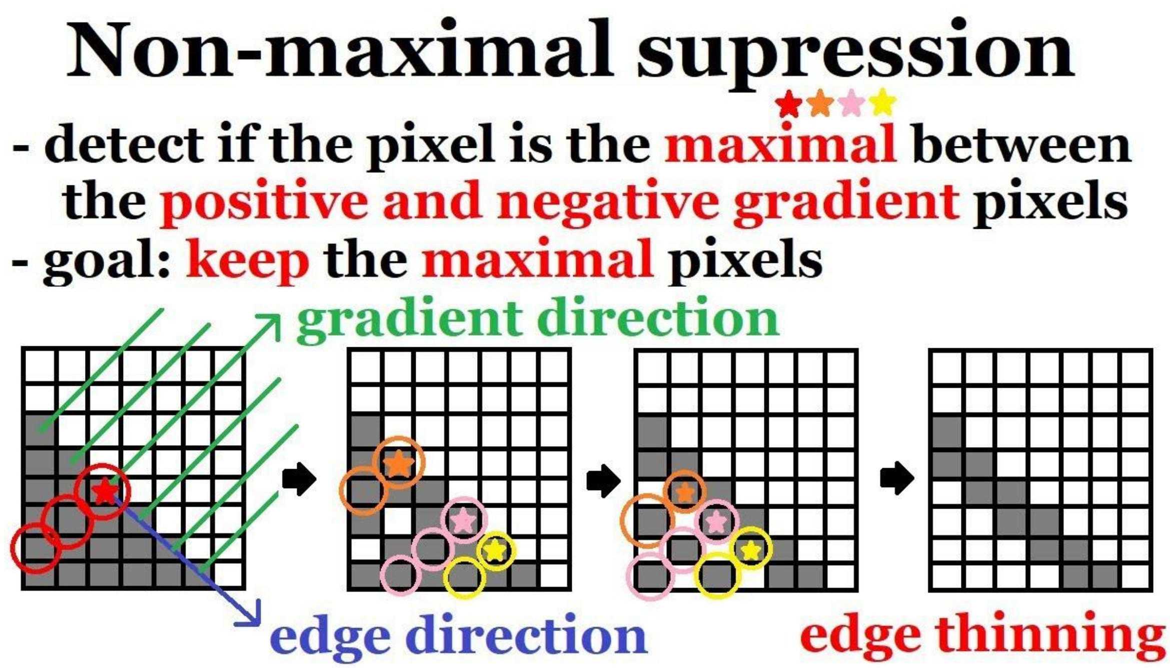


Figure 5: Gradient Magnitude and Angle Images

Non-Maximal Suppression: Edges are thin parts of an image but after obtaining the gradient map, it is observed that edge like regions are not thin enough. Therefore, they should be reduced for only the greatest value in a particular direction of gradient for certain neighborhood. NMS is utilized for this task.



The gradient direction is quantized into four or eight discrete directions (usually using angles like 0°, 45°, 90°, 135°, etc.). For each pixel in the gradient magnitude image, Non-Maximum Suppression involves comparing the gradient magnitude of the pixel with its neighbors in the direction of the gradient. If the gradient magnitude of the current pixel is the maximum among its neighbors in the direction of the gradient, it is retained; otherwise, it is suppressed (set to zero). This ensures that only local maxima in the gradient direction are preserved. Thus, a thinned image is obtained.

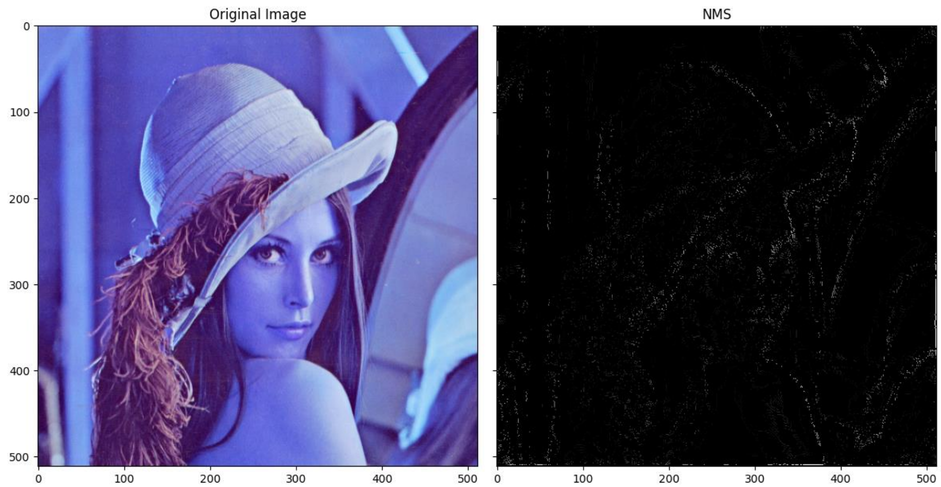


Figure 6: Non-Maximal Suppression

Double Thresholding and Hysteresis: After obtaining the NMS output, there are regions regarded as edge even though some regions are not edges but small artifacts or noisy structures. In order to eliminate such false edges, we threshold the image and extend the real edges. Gradient magnitude image is thresholded to classify edge pixels into three categories strong, weak, and non-edges.

- Strong Edges: Pixels with gradient magnitudes higher than a high threshold are considered strong edge pixels.
- Weak Edges: Pixels with gradient magnitudes between a low threshold and the high threshold are considered weak edge pixels.
- Non-Edges: Pixels with gradient magnitudes below the low threshold are considered non-edges and are discarded.

Hysteresis is used to connect weak edges to strong edges, forming continuous edges. The process involves tracing along the edges in the image and determining whether weak edges should be included in the final edge map. Weak pixels are eliminated and for non-edge pixels, it checked that whether they have at least one strong edge within their neighborhood.

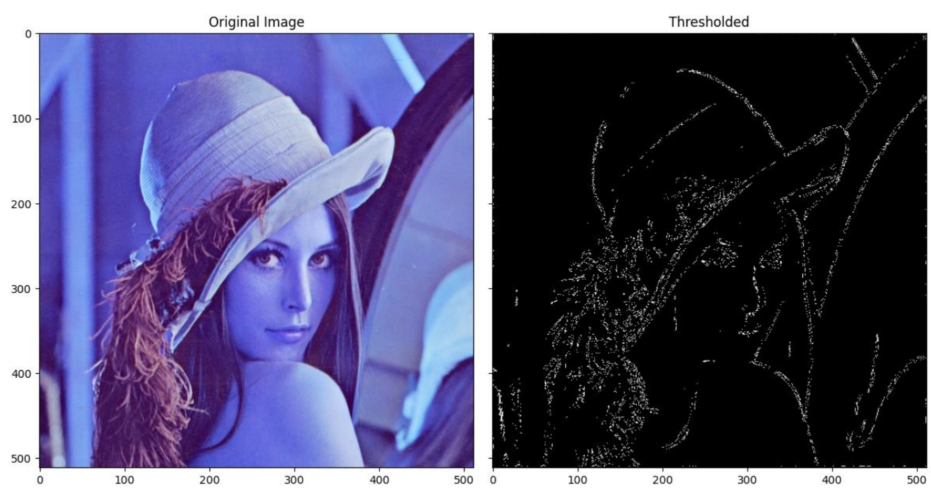


Figure 7: Thresholding and Hysteresis

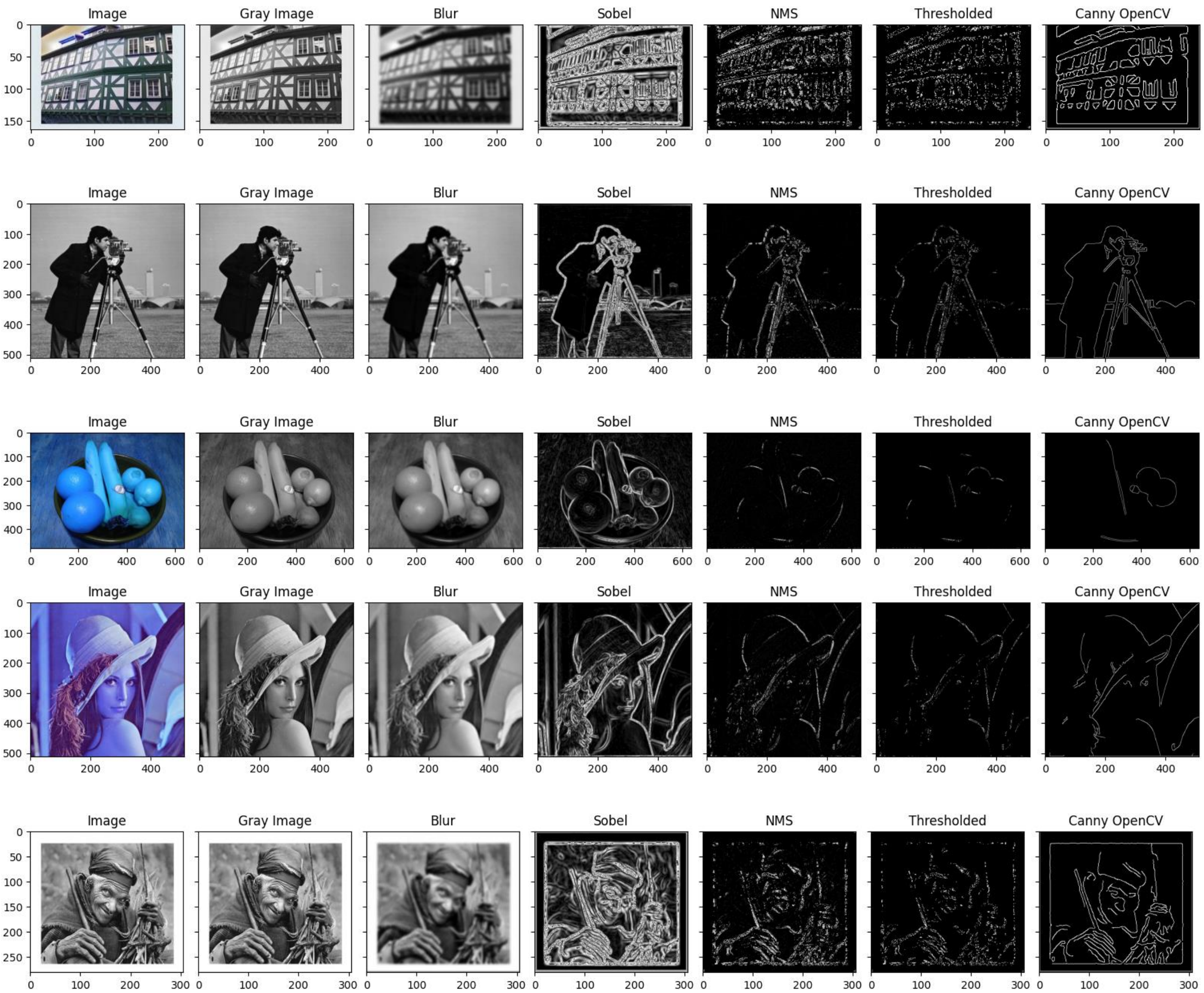


Image
Gray Image
Blur
Sobel
NMS
Thresholded
Canny OpenCV