KHULNA UNIVERSITY OF ENGINEERING & TECHNOLOGY

Department of Computer Science and Engineering

CSE 4128

Image Processing and Computer Vision Laboratory

Assignment-02

Date of Submission: 06 March, 2024

Submitted By	Submitted To
Md. Zakaria Hossain Roll: 1907056 Year: 4^{th} Semester: 1^{st} Department of Computer Science and Engineering (CSE) Khulna University of Engineering and Technology (KUET), Khulna-9203	**Dr. Sk. Md. Masudul Ahsan** Professor **Dipannita Biswas** Lecturer Department of Computer Science and Engineering (CSE) Khulna University of Engineering and Technology (KUET), Khulna-9203

Task:

Apply **Canny edge detection** for detecting edges in images.

The steps involved in implementing the Canny edge detection algorithm:

1. **Differential operators** along x and y axis : the value of sigma will be user input.
2. **Non-maximum Suppression** finds peaks in the image gradient
3. **Hysteresis thresholding** locates edge strings

Gradient Calculation:

1. Gaussian kernel is used to apply convolution to the grayscaled image.

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

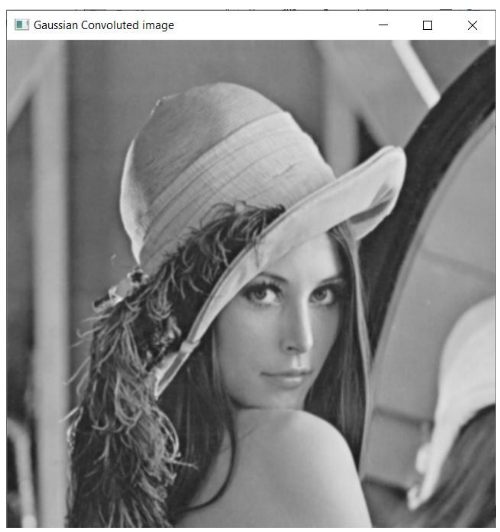

Fig-1: Convolved image using gaussian kernel

2. Partial derivatives w.r.to x and y are calculated. These are the kernel_x and kernel_y. Formulas are:

$$\frac{\partial G_\sigma(x, y)}{\partial x} = -\frac{x}{\sigma^2} G_\sigma(x, y) \qquad \frac{\partial G_\sigma(x, y)}{\partial y} = -\frac{y}{\sigma^2} G_\sigma(x, y)$$

3. The gaussian convolved image is again convolved using the x_derivative and y_derivative kernels:

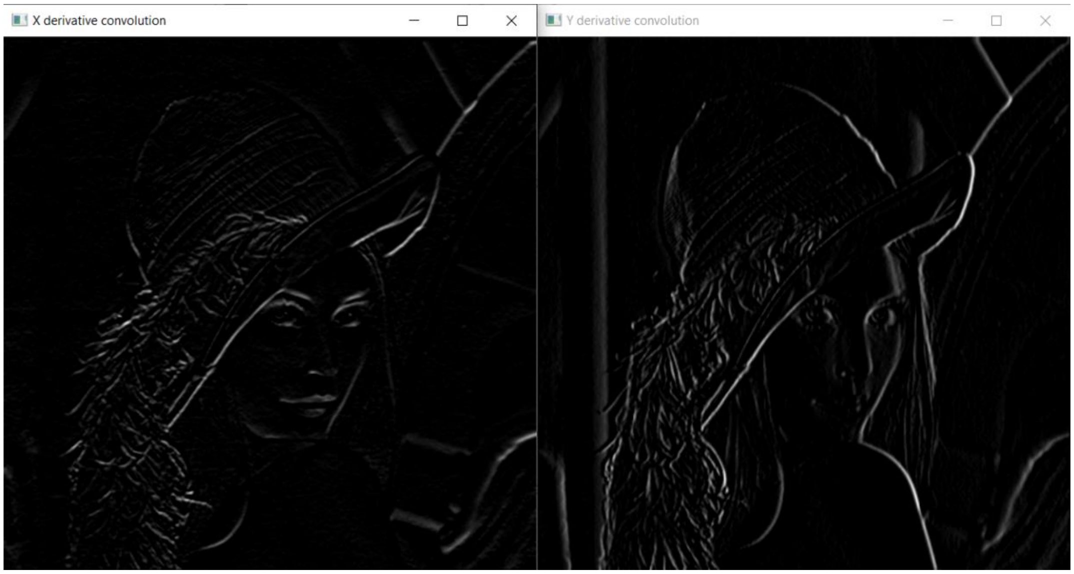

Fig-2: Convolved image using x_derivative(left) and y_derivative(right)

4. Magnitude at each pixel(x,y) is calculated using

$$\sqrt{((value\_at\_x)^2 + (value\_at\_y)^2)}$$

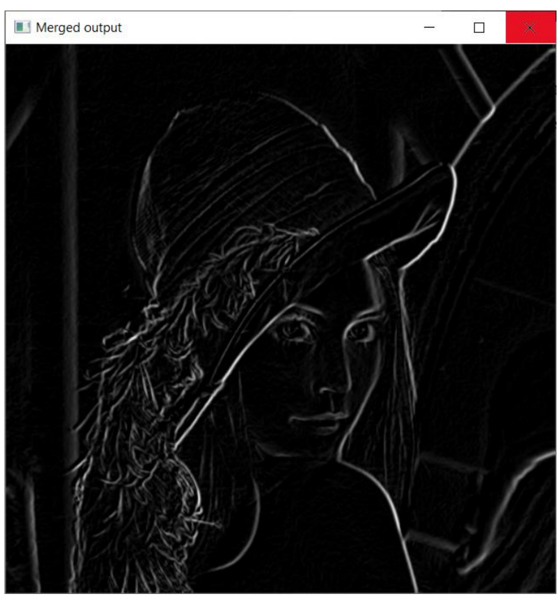

Fig-3: Merged output of previous two images

Non-maximum Suppression:

1. Converts the detected thick edges into thin edges.
2. Uses the gradient to find the pixels to compare with.

i-1,j+1	i,j+1	i+1,j+1
i-1,j	i,j	i+1,j
i-1,j-1	i,j-1	i+1,j-1

3. If the value at comparing pixel is greater than the values of other two pixels, then the value remains same, otherwise 0.

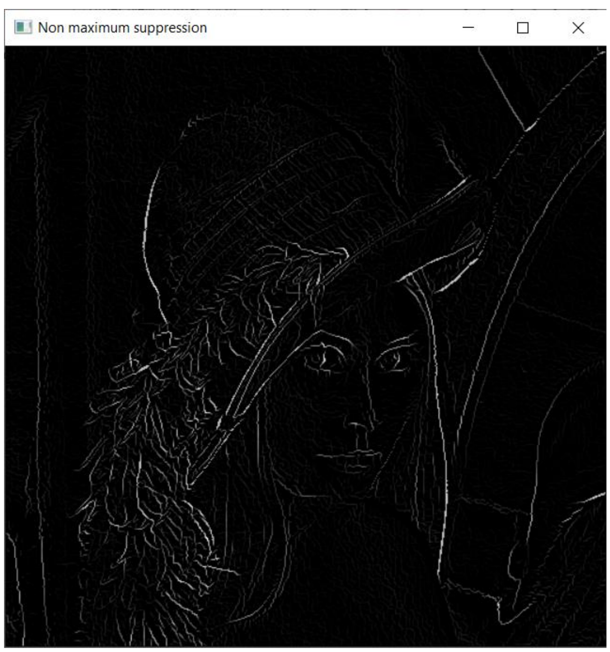

Fig-4: Result after non maximum suppression

Hysteresis thresholding:

1. Acquire lower and higher threshold value using some low and high threshold ratio.
2. If the pixel value is
   a. larger than higher threshold, then assign 255
   b. smaller than lower threshold, then assign 0.

c. Otherwise, assign a weak value (25).

Fig-5: Result after double thresholding

3. Acquire final output by performing hysteresis by the following technique: For each weak pixel of the image, if it is connected to
   a. any strong pixel, it is made 255,
   b. otherwise, it is made 0.

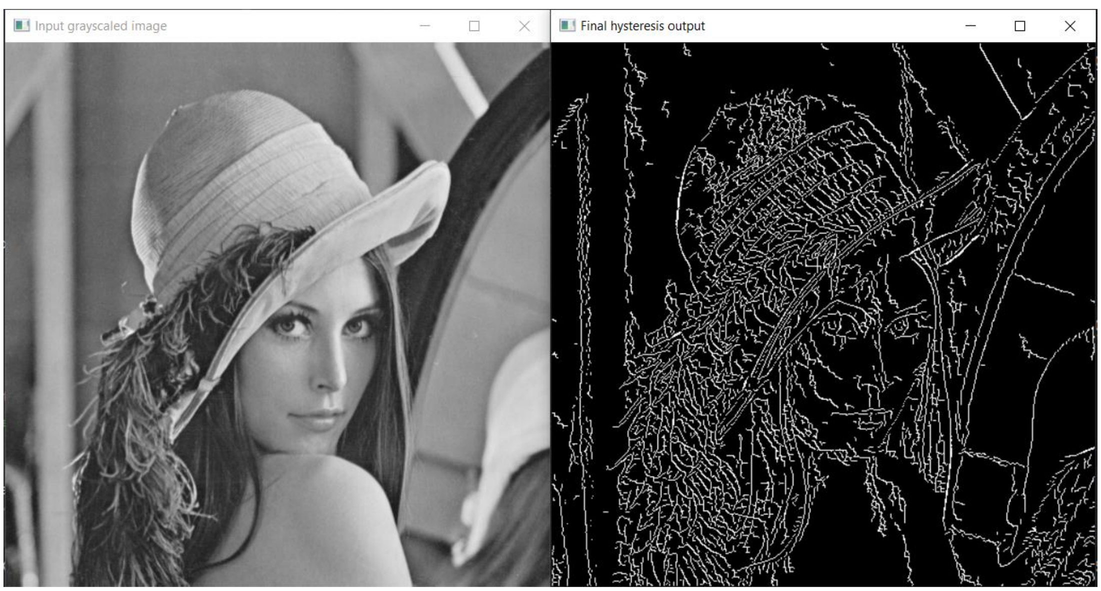

Fig-6: Final Input and Output Image