CSE 573 Computer Vision & Image Processing

Project 3 Report

UBitName : abhave

personNumber : 50289049

Name: Abhishek Bhave

Date : December 3, 2018

Task 1- Morphology image processing

1a

In the given task we are asked to remove noise using two morphology image processing algorithms. We take these 2 morphological operations as Dilation and Erosion.
For denoising we use the Opening and Closing morphological operations.

Figure 1: Image for showing Opening Operation - res_noise1.jpg

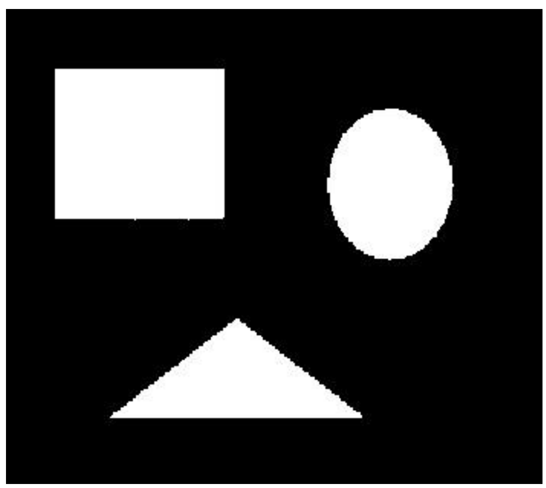

Figure 2: Image for showing Closing Operation - res_noise2.jpg

Opening Operation

$$A \circ B = (A \ominus B) \oplus B.$$

Closing Operation

$$A \bullet B = (A \oplus B) \ominus B.$$

Where A = image and B=structuring element

1b

The above images are results obtained from Opening and Closing operations performed on the give image.

An opening is an erosion followed by a dilation whereas closing is a dilation followed by an erosion. In terms of the shape of the image, erosion slightly shrinks the area of the image while dilation slightly enlarges it. [2]

By doing an erosion followed by a dilation (opening), we are shrinking the object and then growing it again. This will bring the image back to where it was before since we have a 3x3 square structuring element. Similarly, for closing operation in which we do dilation followed by an erosion, we are growing the object and then shrinking it again, also bringing the image back to where it was before.

We notice that opening and closing morphological operations gives slightly results different or varying results even though both operations are able to remove noise. Both morphological operations keep the size and shape of the image and objects intact, but I have noticed slight changes in the boundary of the objects in both opening and closing.

1c

Now to extract the boundary we will use erosion i.e. erode the output images of the above task. After eroding these images, we will subtract the resulting images from the noise free images that we obtained from the above task.
The result is as follows:

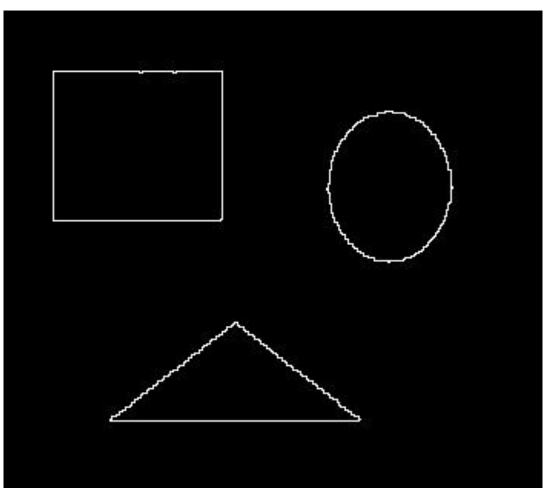

Figure 3: Image for showing boundary - res_bound1.jpg

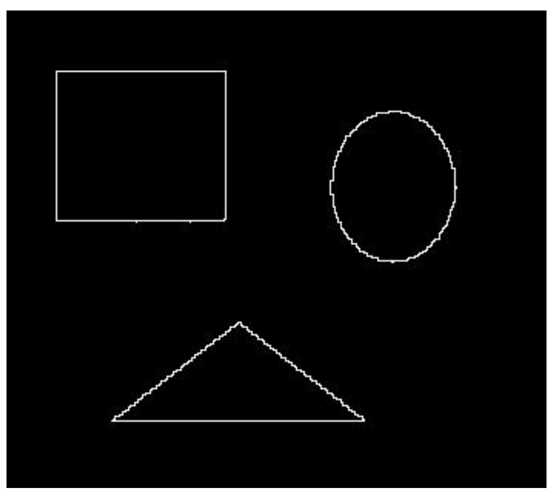

Figure 4: Image for showing boundary - res_bound2.jpg

Task 2 - Image segmentation and point detection

2a

For this task I have used the lossless turbine-blade.jpg image that was made available by the Teaching Assistant through Piazza. The purpose of using this image is to get optimum result.

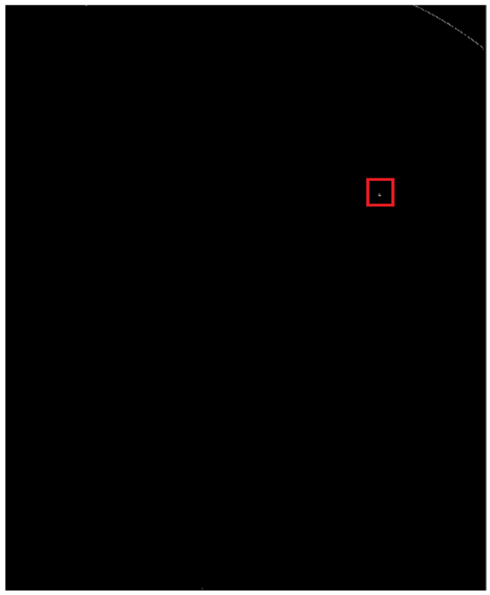

Figure 5: Image for showing detected porosity

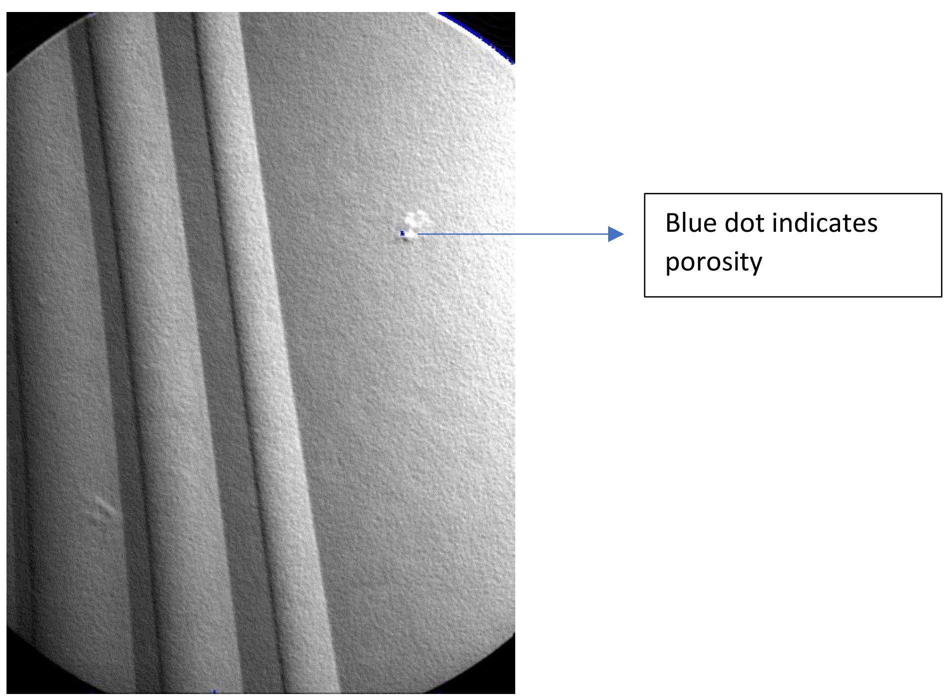

Figure 6: Image for showing detected porosity on input image

I have used a 3x3 point detection kernel which is as follows [4]:

−1	−1	−1
−1	8	−1
−1	−1	−1

I tried using the 5x5 point detection kernel but using the 3x3 kernel with the lossless image was giving me a better result.

I varied the value of the threshold manually and found 5 points of porosity. They are listed below.

The coordinates of the points of the porosity are as follows:

[247, 443], [247, 444], [249, 443], [249, 444], [249, 445]

2b

The task is to segment the object from background by thresholding on 'segment.jpg' by choosing the optimum threshold. I have used a histogram to calculate the optimum threshold and am using that optimum threshold to segment the object from background.

The term image segmentation refers to the partition of an image into a set of regions that cover it. We do the following steps:

1. Read the given image that is to be segmented

2. Create histogram (procedure described below)

3. Get optimal threshold value

4. Apply optimal threshold on image to get segmented image

5. Draw bounding boxes on detected images

To create the histogram, I have made a dictionary which has image value i.e. (0-255) as key and its number of occurrences as the values.
Next, I have plotted the key and values of the dictionary. Here I have only used the values from 1 to 255 as most of the pixels in the image are 0 i.e. black and hence we get a skewed histogram.
The histogram returns a threshold value of 200.

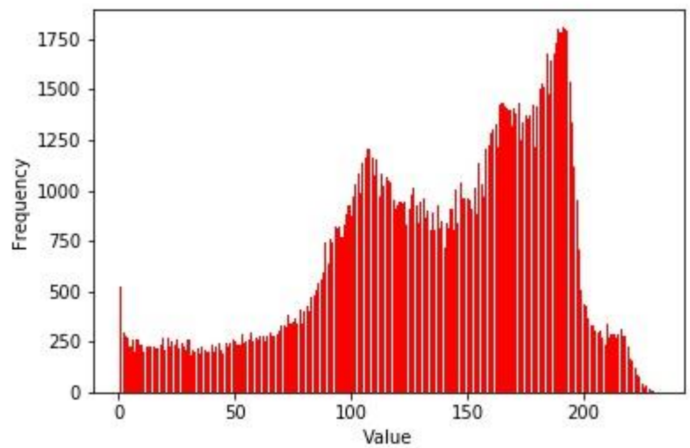

Figure 7: Image for showing histogram

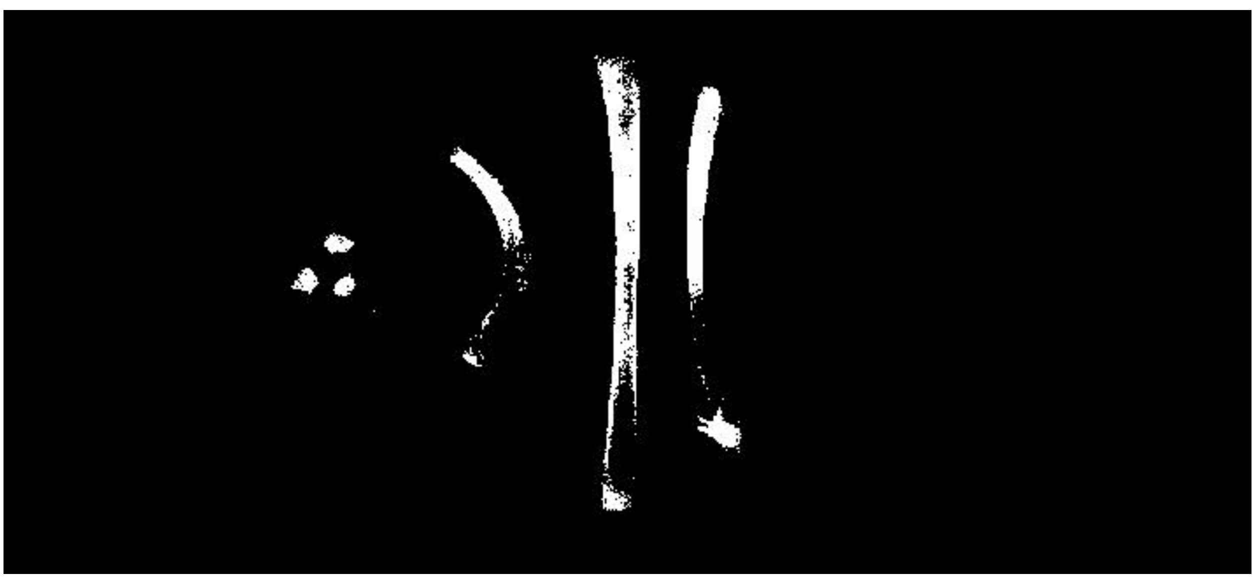

Figure 8: Image for showing segmented image

The above figure i.e figure 8 depicts the segmentation of the object from background by thresholding.

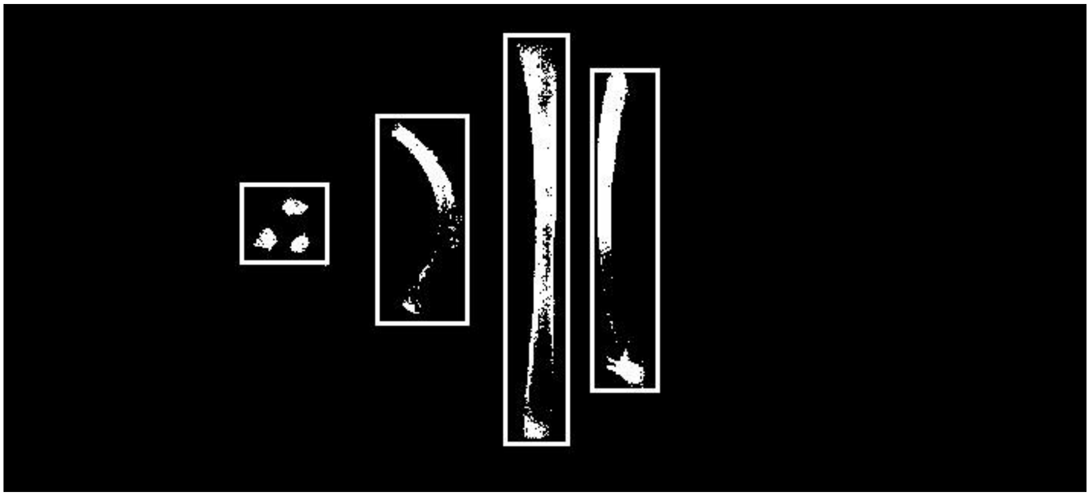

Figure 9: Image for showing bounding boxes

The coordinates of the 4 boxes are as follows:

(left top corner) (right top corner) (left bottom corner) (right bottom corner)

Box 1 - (117,155) (117,213) (171,155) (171,213)

Box 2 - (72,244) (72,305) (210,244) (210,305)

Box 3 - (19,328) (19,371) (289,328) (289,371)

Box 4 - (42,385) (42,430) (254,385) (254,430)

Task 3 - Hough transform

Hough transform [1] is a technique that can be used to detect(link)regular curves such as lines, circles, and ellipses in an image.

In this task we will use Hough Transform to detect vertical and diagonal lines in the given image as well as to detect the coins in the image.

We do the following steps to obtain the vertical and diagonal lines:

1. Apply diagonal Sobel operator to get edges of diagonal lines.

2. Apply vertical Sobel operator to get edges of vertical lines.

3. Apply Hough transform

4.Create accumulator matrix H using equation [4]

$$x\, cos\, \theta + y\, sin\, \theta = P$$

5.Calculate local maximum values from accumulator matrix and get corresponding x,y pairs and rho theta pairs.

6.Draw lines by converting rho theta pairs to x,y coordinates

3a

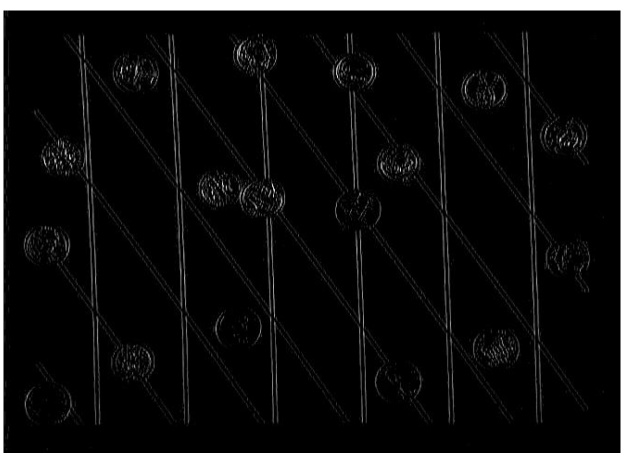

Figure 10: Image for showing edges using Sobel operator

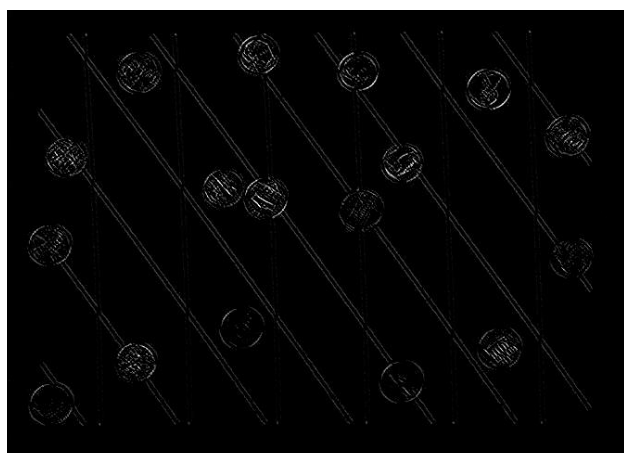

Figure 11: Image for showing edges using Sobel operator

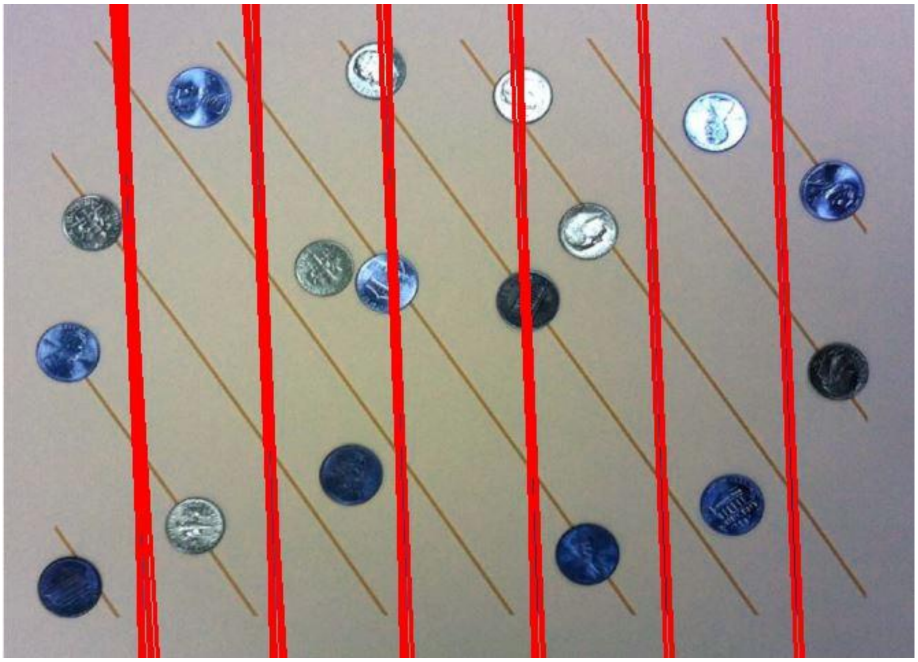

Figure 12: Image for showing detected vertical lines – red_line.jpg

Number of lines detected - 6

3b

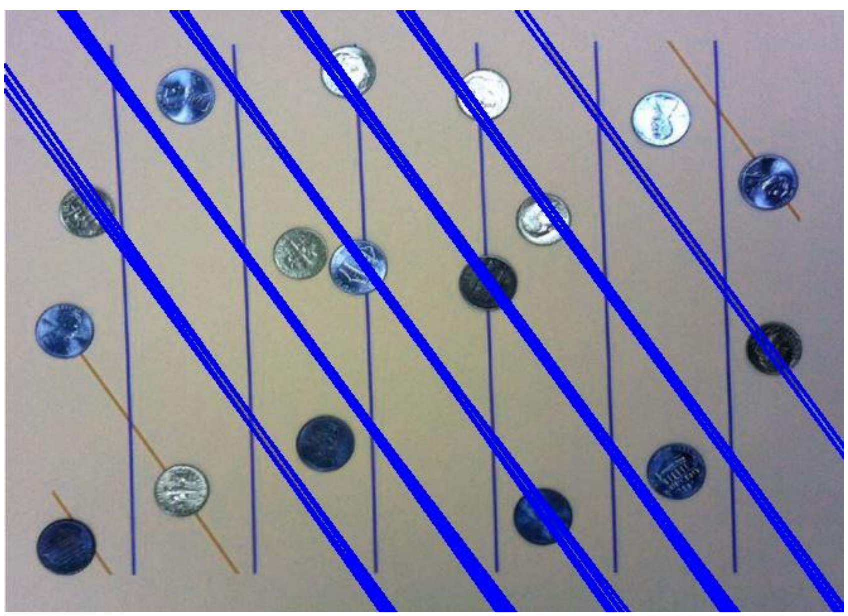

Figure 13: Image for showing detected diagonal lines – blue_lines.jpg

Number of lines detected – 6

References:

[1] – https://gist.github.com/ilyakava/c2ef8aed4ad510ee3987

[2] –https://stackoverflow.com/questions/27085707/cases-where-morphological-opening-and-closing-yields-the-same-results

[3] – Lecture Notes 10

[4] – Lecture Notes 12

[5] – https://courses.cs.washington.edu/courses/cse576/book/ch10.pdf