Project 3 : Computer Vision and Image Processing

KRISHNA SEHGAL UB ID: 50291124

1. Software Used

- Anaconda - Navigator
- Jupyter Notebook
- Python 3.6

2. Libraries Used

- Numpy
- Open CV
- Matplot
- Math

3. Task

3.1 Morphology Image Processing

Erosion

Erosion is one of the two basic operators in the area of mathematical morphology, the other being dilation. It is typically applied to binary images. The basic effect of the operator on a binary image is to erode away the boundaries of regions of foreground pixels. Thus areas of foreground pixels shrink in size, and holes within those areas become larger.

$$\text{Erosion} \qquad A \ominus B = \{z \mid (B)_z \subseteq A\}$$

The erosion operator takes two pieces of data as inputs. The first is the image which is to be eroded. The second is a (usually small) set of coordinate points known as a structuring element (also known as a *kernel*). It is this structuring element that determines the precise effect of the erosion on the input image.

Dilation

Dilation is one of the two basic operators in the area of mathematical morphology, the other being erosion. It is typically applied to binary images, but there are versions that work on grayscale images. The basic effect of the operator on a binary image is to gradually enlarge the boundaries of regions of foreground pixels (*i.e.* white pixels, typically). Thus areas of foreground pixels grow in size while holes within those regions become smaller.

The dilation operator takes two pieces of data as inputs. The first is the image which is to be dilated. The second is a (usually small) set of coordinate points known as a structuring element (also known as a *kernel*). It is this structuring element that determines the precise effect of the dilation on the input image.

$$\text{Dilation} \qquad A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing\}$$

Opening
The Opening operation is a combination of the Erosion operation followed by a Dilation operation.

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

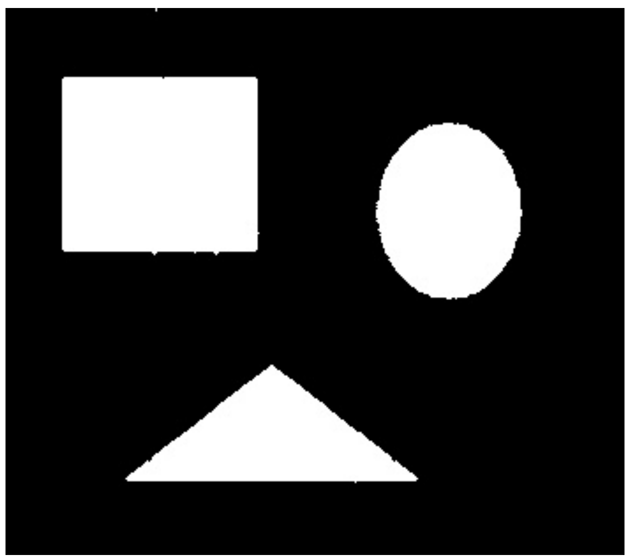
Closing
The Closing operation is a combination of the Dilation operation followed by an Erosion.

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

Noise Operations

We can apply combination of Erosion and Dilation to remove noise from an image. As Opening and Closing are combination of Erosion and Dilation itself so noise can be removed from an image by applying opening and closing operations.

$$(A \bullet B) \circ B$$



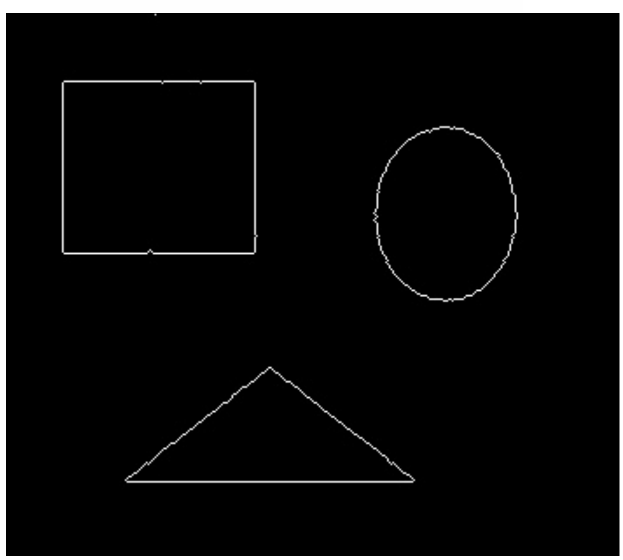
res_noise1.jpg

$$\left(A \circ B\right) \bullet B$$

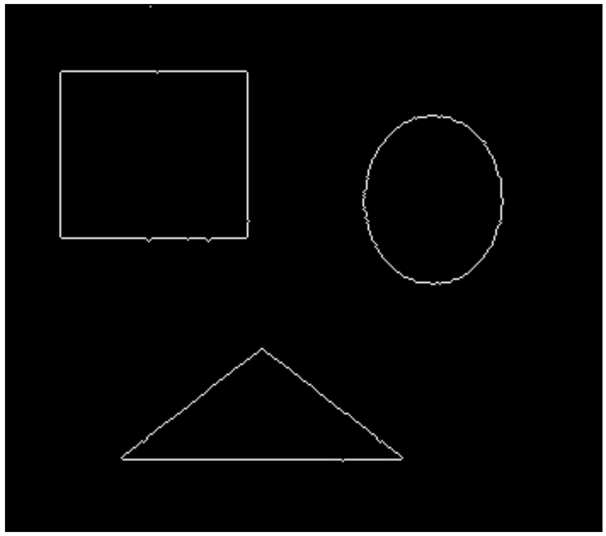
res_noise2.jpg

Boundary Extraction

$$\beta(A) = A - (A \ominus B)$$



res_bound1.jpg



res_bound2.jpg

I have used two operations to remove noise from image, In the first method Closing operation followed by Opening is used and in the second noise reduction method I have first applied Opening operation followed by a Closing operation. In the two outputs of noise reduction we can observe there is a very little difference in the results. The first image has some points coming out of the geometrical shape whereas in the second result some points of the geometric box are replaced by a 0 value. But, overall both the noise operations have removed noise.

3.2.1 Point Detection

In Point Detection we have applied convolution across the entire image.

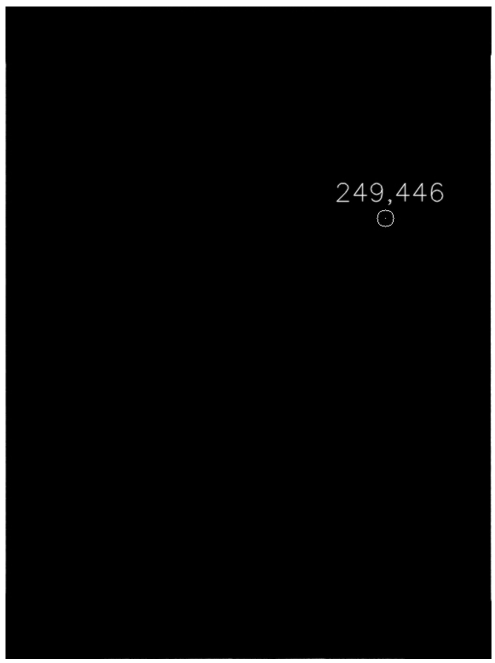
$$R = \sum_{i=1}^{9} w_i z_i$$

The formulation measures the weighted difference between the center point and its neighbors. A point is detected at the location on which the mask is centered if

$$|\mathbf{R}| >= \mathbf{T}$$

I have used the kernel :

−1	−1	−1
−1	8	−1
−1	−1	−1



Point_detection.jpg

3.2.2 Image Segmentation

Segmentation is to subdivide an image into its constituent regions or objects. Segmentation algorithms generally are based on one of 2 basic properties of intensity values:

Similarity: to partition an image into regions that are similar according to a set of predefined criteria. (Thresholding)

Discontinuity: to partition an image based on abrupt changes in intensity (Point, Line and Edge Detection)

A thresholded image g(x, y) is defined as:

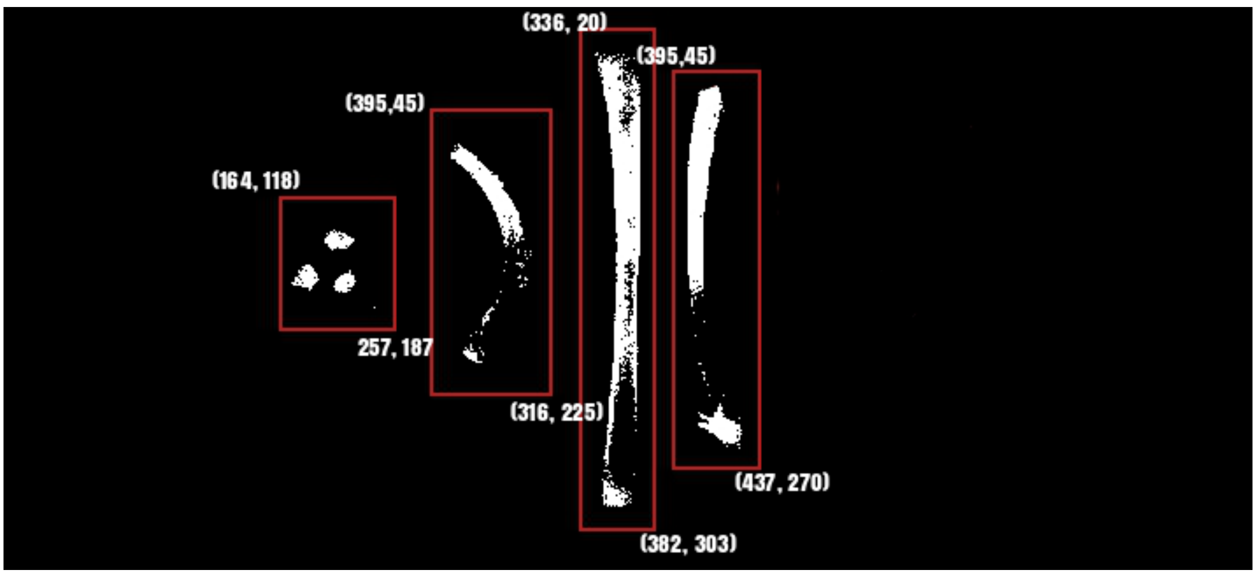
$g(x, y)=1$ if $f(x, y) > T$

$g(x, y)=0$ if $f(x, y) <= T$

where T is the threshold given by :

$$T = T[x, y, p(x, y), f(x, y)]$$

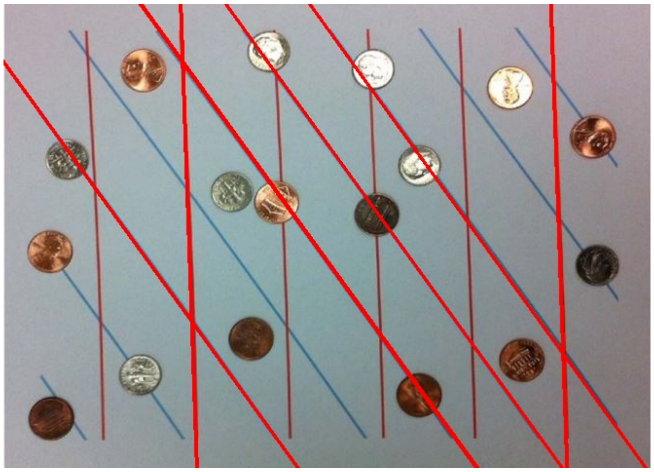
Image_Segmentation.jpg



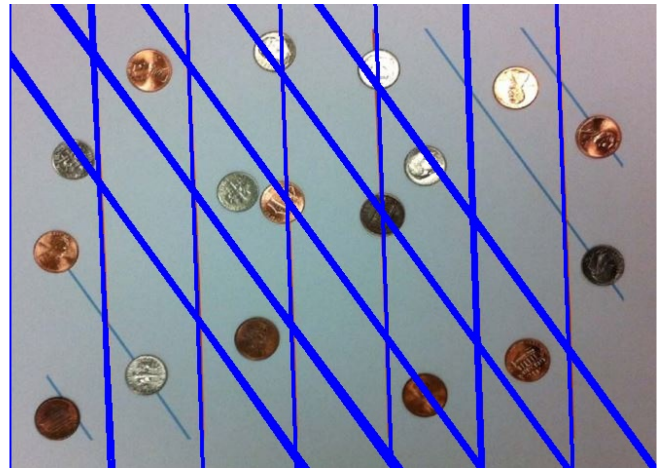
3.3 Hough Transform

Hough transform is a technique that can be used to detect (link) regular curves such as lines, circles, and ellipses in an image. A line in x-y-plane is a set of points (x, y) that satisfy equation y=a'x+b' which is mapped into a point in ab-plane(a',b'). A point in x y-plane may have many lines go through it, which is mapped into a line in ab-plane. (x_j , y_j) implies $b = y_j - a x_j$. All points (x_i, y_i) on a same line in the image must fall into a same point (a_i, b_i) in the parametric space.

Reference: https://github.com/alyssaq/hough_transform/blob/master/hough_transform.py



Red_lines.jpg



Blue_lines.jpg

I have detected 4 red lines and 6 blue lines after applying Hough Transform to the image.