Assignment Report

Question 1: Histogram Computation: Compute the histogram of the image coins.png, by finding the frequency of pixels for each intensity level {0, 1, . . . , 255}. Show the histogram by plotting frequencies w.r.t. intensity levels. Comment on what you observe. Also, find the average intensity of the image using this histogram. Verify the result with the actual average intensity.

Results:

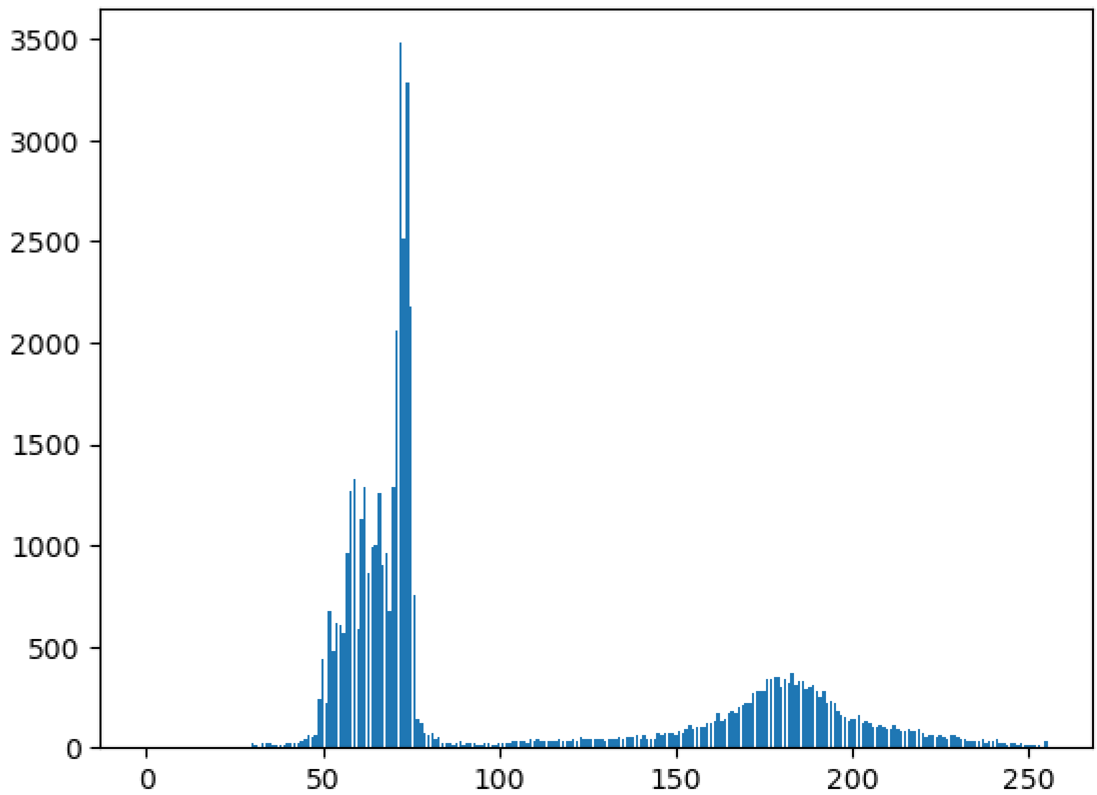

Figure 1: Histogram of coins.png

- Average value of pixels in the image = `103.30500158906722`
- Since pixels can take only discrete intensity values (assuming unit8), we can round off the average intensity level to be 103.

Inferences:

- The image is a reality dark image as the mass of histogram is more towards the lower intensity values.
- There is a clear separation of foreground (coins) and background(dark background) of the image as there are two distinct distributions on the darker intensity values and the brighter intensity values.
- Eventhough though the average pixel intensity value is obtained as 103, there are very few pixels with the actual value of 103. So, average intensity can’t be considered as a good parameter to get an idea on the most likely value of pixels in an image. This is a general inference for all mean values.

Question 2: Otsu's Binarization: In the class, we showed that σ_w^2 (t) + σ_b^2 (t) = σ_T^2 , where t is the threshold for binarization. Binarize the image coins.png by finding the optimal threshold t by:
(a) Minimizing the within class variance σ_w^2 (t) over t.
(b) Maximizing the between class variance σ_b^2 (t) over t.
Verify that both methods are equivalent. Compare the time taken by each of the approaches.

Results:

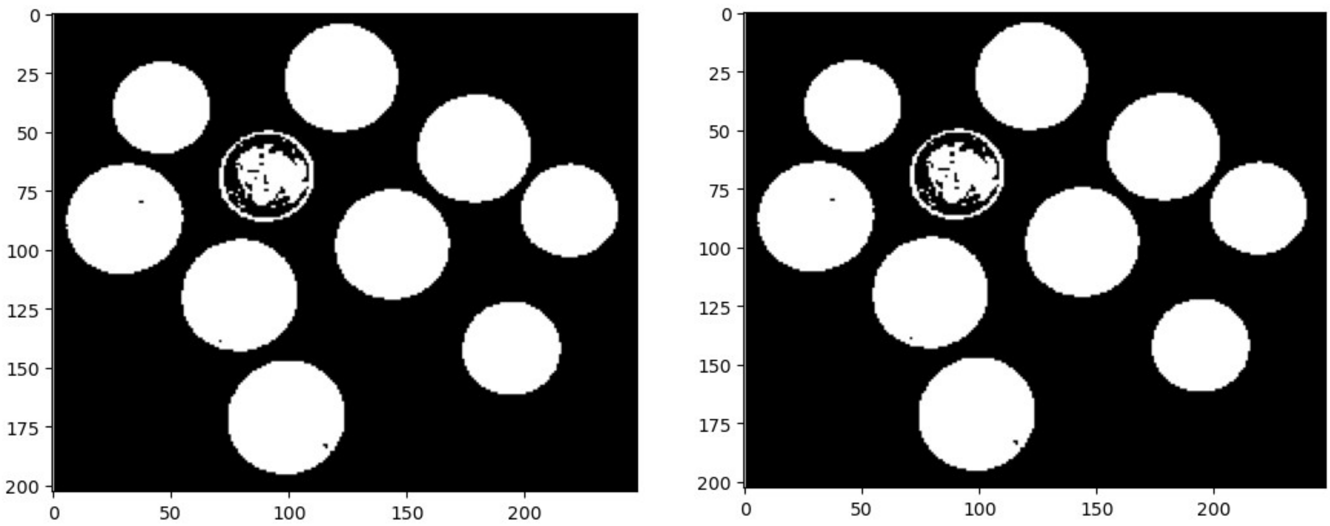

Figure 2: Images obtained after binarization using Otsu's threshold (a) by minimizing within class variance and (b) maximizing between class variance.

Otsu threshold
Otsu threshold for minimizing within class variance= 125
Otsu threshold for minimizing between class variance = 125
Both are verified to be the same.

Time taken (library used -
Time taken to calculate Otsu threshold for minimizing within class variance = 25592.063 µs
Time taken to calculate Otsu threshold for minimizing between class variance =8255.769 µs

Inferences:

- The overall variance of the image σ_T^2 is found to be a constant and it is a sum of within class variance σ_w^2 (t) and between class variance σ_b^2 (t). So maximizing one of them will obviously result in minimizing the other. Thus, minimizing within class variance is equivalent to maximizing between class variance. This is further verified by the result as both approaches gives the same Otsu threshold of 125.
- The time take to for Otzu's threshold using within class variance is more than between class variance approach. This reason for is the calculation of second order variance of the class in each iteration which is computationally more complex than the first order term in between class variance.
- The resultant binary image obtained by Otsu threshold is giving fairly good results, but the binarization is not ideal for the given image. Few foreground portions (pixels) of the image are incorrectly labelled as background. This leads to a requirement of other thresholding techniques for better binarization.

Question 3: The image IIScTextDepth.png is an inverse depth map of IIScText.png.
A depth map indicates the depth of an object from the camera for each pixel. Particularly, an inverse depth map has a higher value when the object is nearer to the camera and a lower value when it is farther apart. Binarize the inverse depth map IIScTextDepth.png and use that information to extract the text in IIScText.png and display it over the background image IIScMainBuilding.png.

Results:

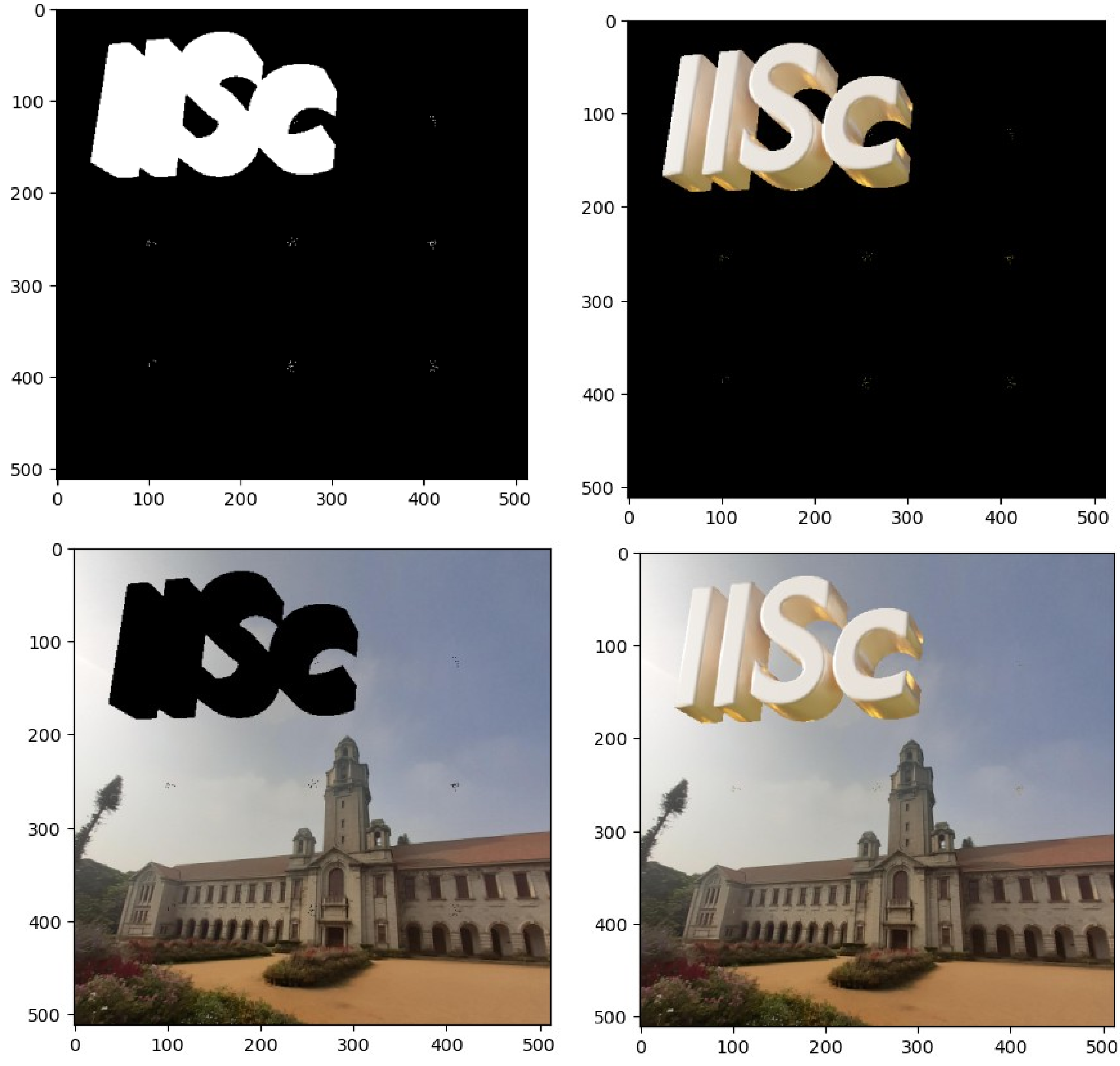

Figure 3: Different image stages of extracting IISc foreground and displaying it over building image background.

Question 4: Binarize the image quote.png and count the total number of characters excluding punctuations using connected component analysis

Results:

Connected component analysis is done in raster scan order using N-4 neighbors.

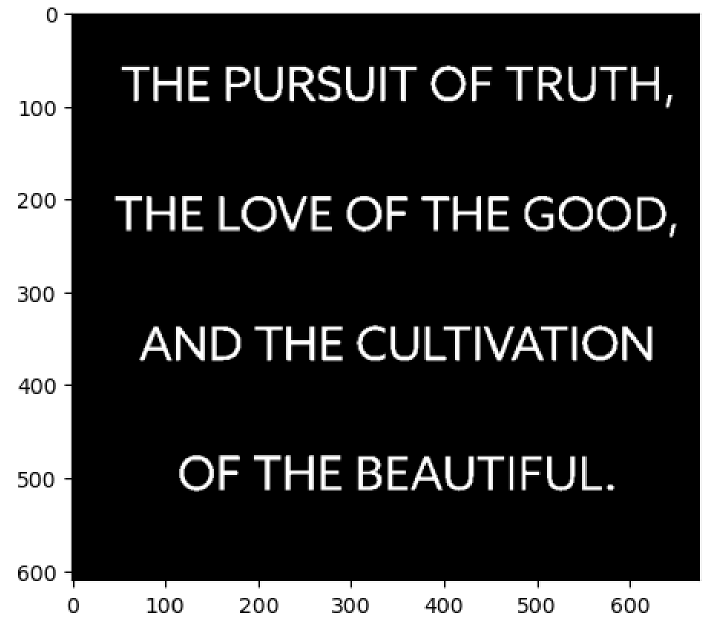

Figure : Image obtained after binarization using Otsu's threshold of quote.png

Figure : Image showing different regions after connected component analysis displayed with different gray levels with the initial labels as lower gray level (smaller region label) and final labels as brighter gray levels. (Inital and final obtained in raster scan order.)

- Total number of connected regions before thresholding = 67

- Total number of connected regions after thresholding (removing punctuation) = 64
- Thresholding size of connected region = 60 (hyperparameter chosen according to the size of the punctuation marks)

Inference/Analysis

Connected component analysis is performed on a sample alphabet ‘V’ cropped from quote.png and the label values of each pixel of the connected component/region is analysed to appreciate how the algorithm works.

Figure: Region labels after first pass without updation

Figure: Region labels after updating the labels, all pixels of the connected region of character ‘V’ marked as 1.