Digital Image Processing

Assignment 2

Mohammad Abdullah Malik

401177

MS-22

12 November 2023

This assignment was done in Python using OpenCV library. The following steps were taken to train a classifier using image processing techniques.

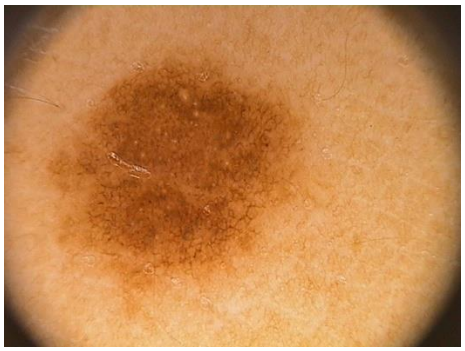
➢ Step 1: Separating Images of Three Classes

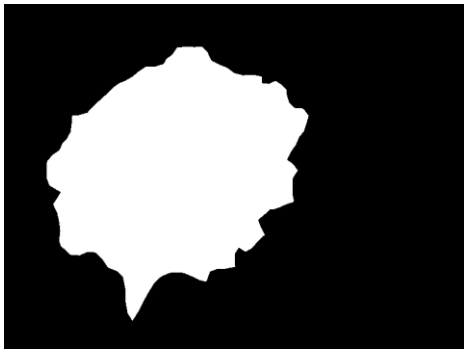
Three classes were given for this assignment. 200 images were given in the same folder which were separated manually with the help of attached excel sheet. The data was also split to an 80-20 ratio. Training and test images were saved in separate folders. 80 images were of class 1, 80 of class 2 and 40 of class 3.

➢ Step 2: Removing Background using Binary Mask

The background was removed from all images so the training could be done using only the values from region of interest. A binary mask was used for this purpose. Binary images were given for all RGB images. Since both RGB and binary images of 1 sample are of the same size, the binary image is placed on top of the RGB image. The area where the white part of coincides with the RGB image is saved and the black part is removed.

An example is shown below.





When these two images are processed, the following result is obtained.

The background is removed from the original image. This will allow us to get more accurate values of parameters.

➢ Step 3: Choosing Parameters

The parameter selection was done based on difference of values of parameters for each class. For all the parameters tested, Class 1 and Class 2 had very similar values whereas Class 3 was visibly separable. After testing on various parameters, two parameters were selected i.e., **'RGB histogram mean' & 'Entropy'.**

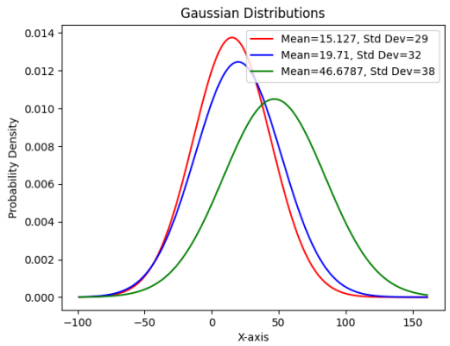
➢ Step 4: Set Threshold Values

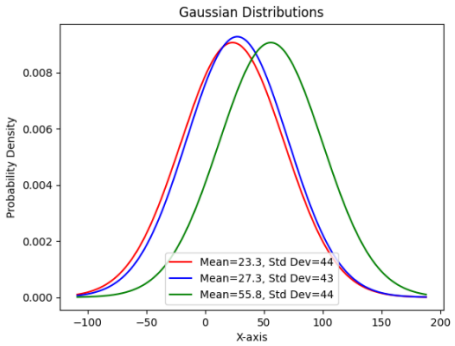
1. Histogram Means

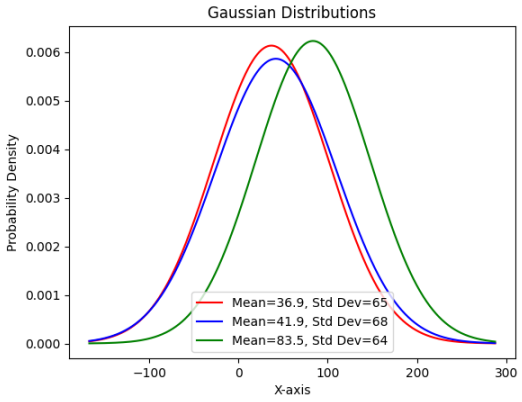
To set the threshold values, we first require values of Histogram means for each class. The python code calculates histogram mean values of all images and stores them in a .csv file. The values given in the table are mean of means of the 3 channels for each class.

	Histogram_Mean_R	Histogram_Mean_G	Histogram_Mean_B
Class 1	**36.96687269**	**23.33847608**	**15.12753509**
Class 2	**41.95332198**	**27.3514385**	**19.7111779**
Class 3	**83.5074423**	**55.82050575**	**46.67872458**

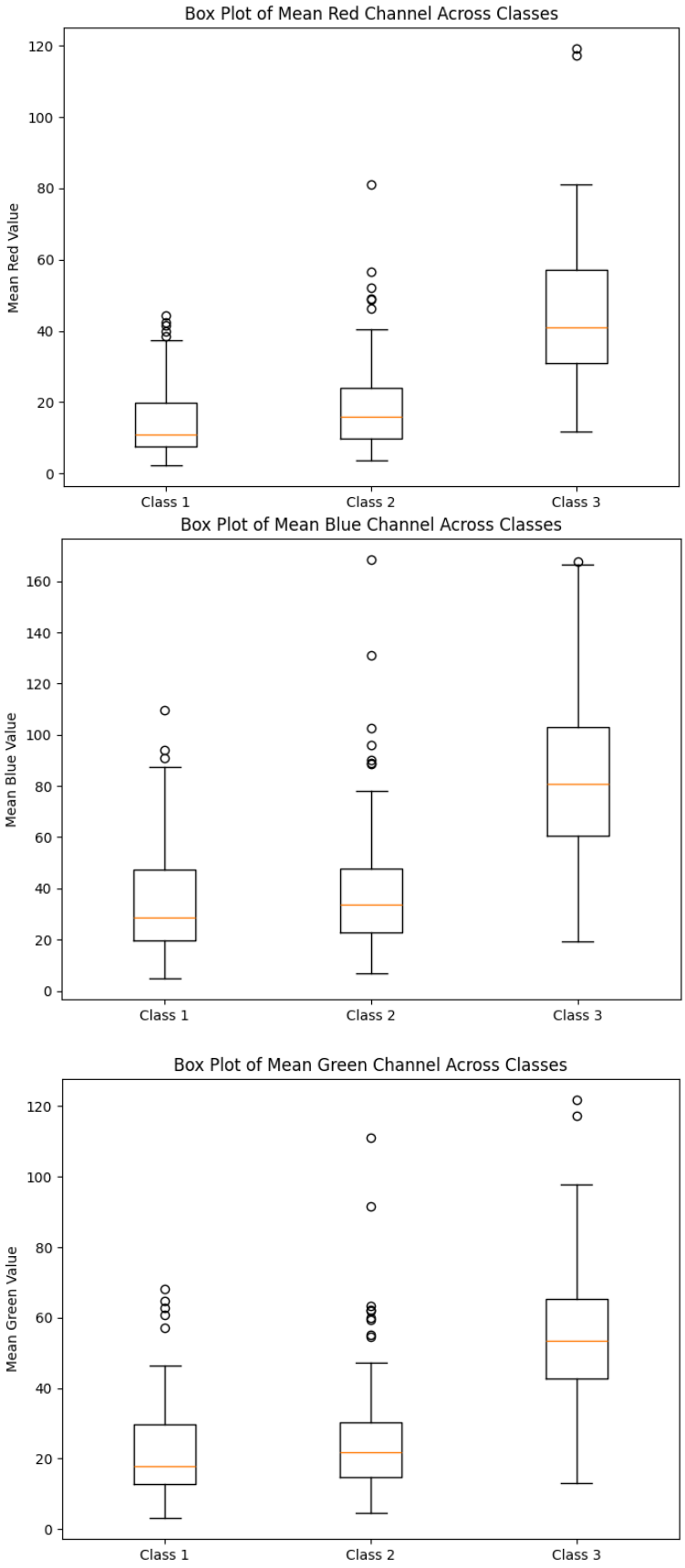
There is little difference between Class 1 and Class 2 but significant difference with Class 3. This can be observed much more clearly using Gaussian distribution for the 3 classes.







Using boxplots for each channel, we can set the threshold values for each class.



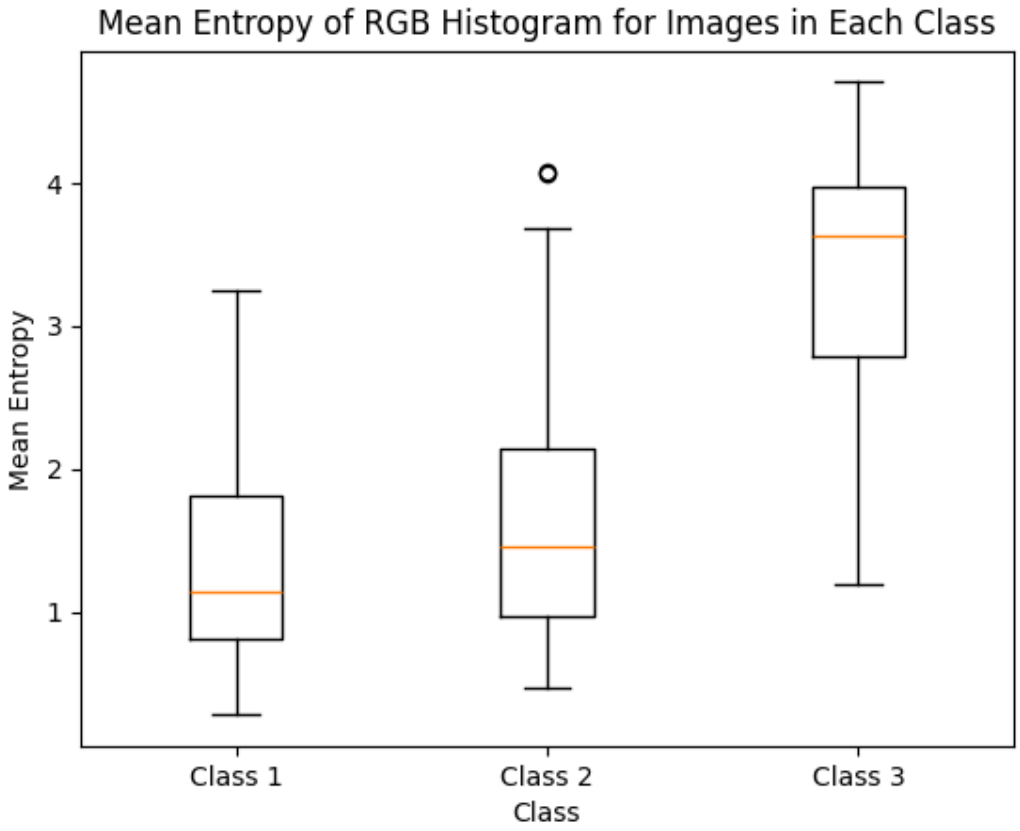
We can get a idea for the lower and upper bounds of threshold values.

	Lower Mean Bounds	Upper Mean Bounds
Class 1	**[2.5, 3.3, 6]**	**[39, 45, 80]**
Class 2	**[3.3, 5, 10]**	**[45, 47, 80]**
Class 3	**[12, 15, 20]**	**[100, 99, 150]**

2. Entropy

The other feature used is entropy which is a measure of the degree of randomness in the image.

The boxplots of entropy for each class is given below.



Once again, there is a small difference between Class 1 and Class 2 but a visible difference between Class 3. Entropy for each image was calculated and average was taken for each class.

	Entropy
Class 1	**2.08**
Class 2	**2.53**
Class 3	**4.9**

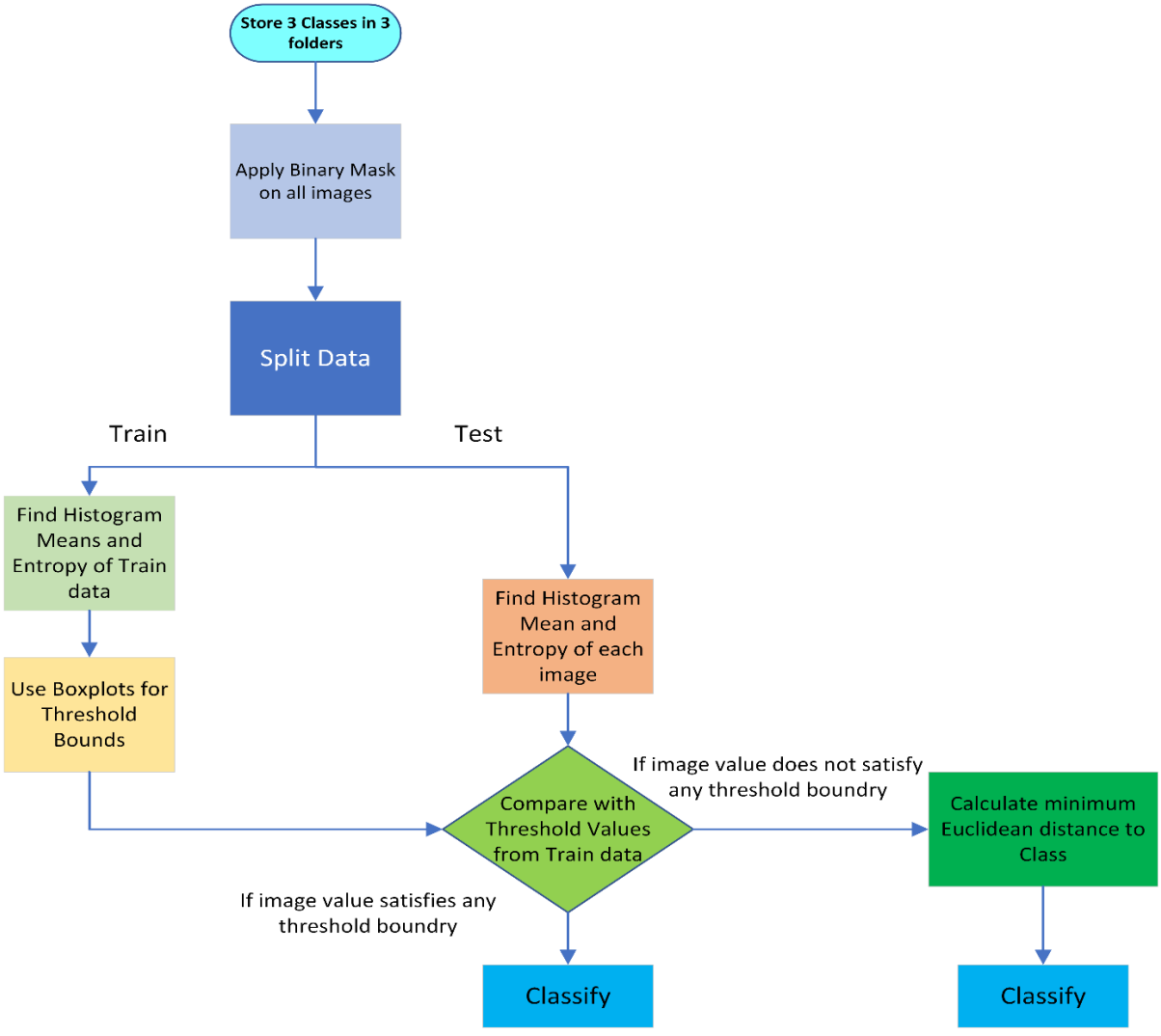
The upper and lower bounds are estimated from the scatter plots.

	Lower Entropy Bounds	Upper Entropy Bounds
Class 1	**0.6**	**2.9**
Class 2	**0.9**	**3.3**
Class 3	**1.6**	**5**

➢ Step 5: Classification Algorithm

Once all the thresholds are set, we start the testing of our model. The test images of three classes are stored in three folders for the sake of finding accuracy. The code iterates through all images one by one and finds the RGB histogram and then the means and then the entropy. The means and entropy of an image are compared with the set threshold boundaries and classified. If an image does not fall between any boundary, it is classified by minimum Euclidean distance to the value of a class. Hence all images are classified.

A flowchart of the whole process is given below.



The following results are obtained from the classifier.

```
Total Images: 40, Correct Assignments: 29, Overall Accuracy: 72.50%
Unclassified Images: 0
Class: Class1, Correct Assignments: 10, Total Images: 16, Accuracy: 62.50%
Class: Class2, Correct Assignments: 11, Total Images: 16, Accuracy: 68.75%
Class: Class3, Correct Assignments: 8, Total Images: 8, Accuracy: 100.00%
```

Conclusion

A classifier for dermoscopic images of melanocytic lesion was made in this assignment. The classifier uses Image Processing techniques instead of Machine Learning algorithms. Binary mask was applied to remove the background of all images. Then the data was split 80-20. The features selected are Histogram means of RGB channels and entropy. The train data was used to calculate the RGB means of histogram and then entropy. Threshold bounds were selected from boxplots of the features. The means and entropy of the test data was then compared with the thresholds and classified. If the test image did not fall in boundary of any class, minimum Euclidean distance was calculated and then the image was classified. The results are satisfactory considering two of the classes i.e., common nevi and atypical nevi are almost similar.