Project 3: Color Segmentation

ENPM673 Perception for Autonomous Robots – Spring 2017

University of Maryland

Project Report by

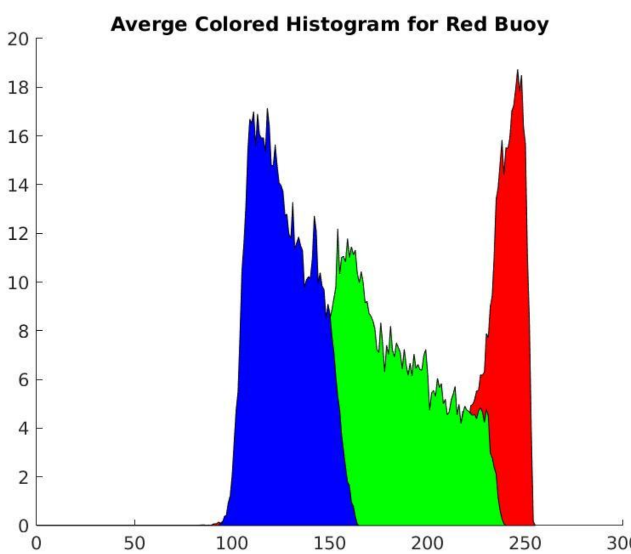
Anirudh Topiwala

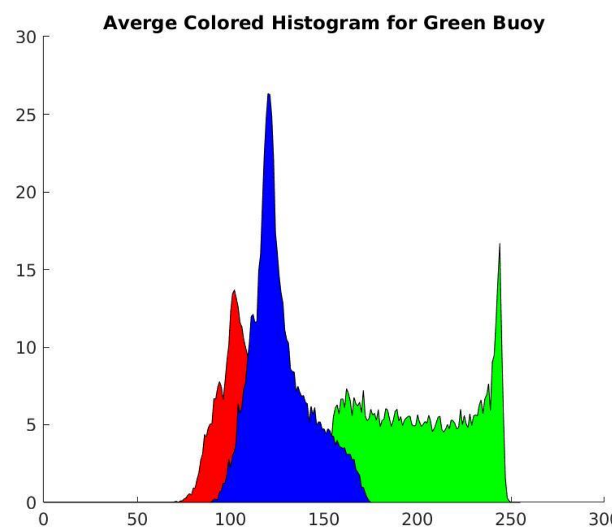
UID: 115192386

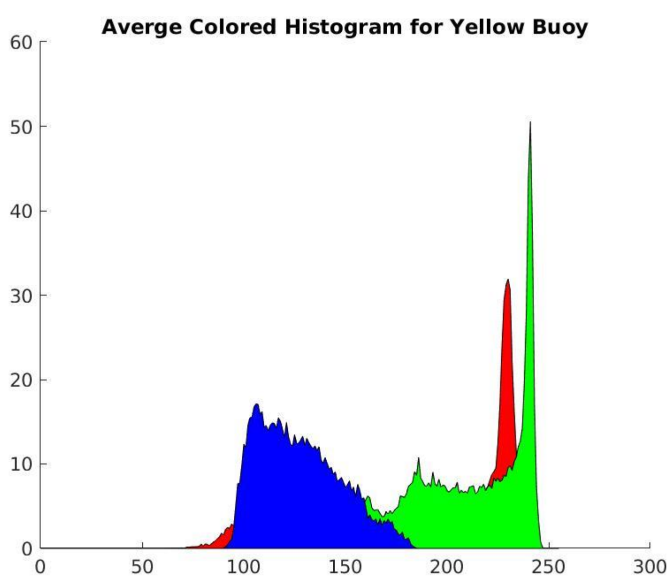
Contents

Part 0: Data Preparation and Vanilla Approach

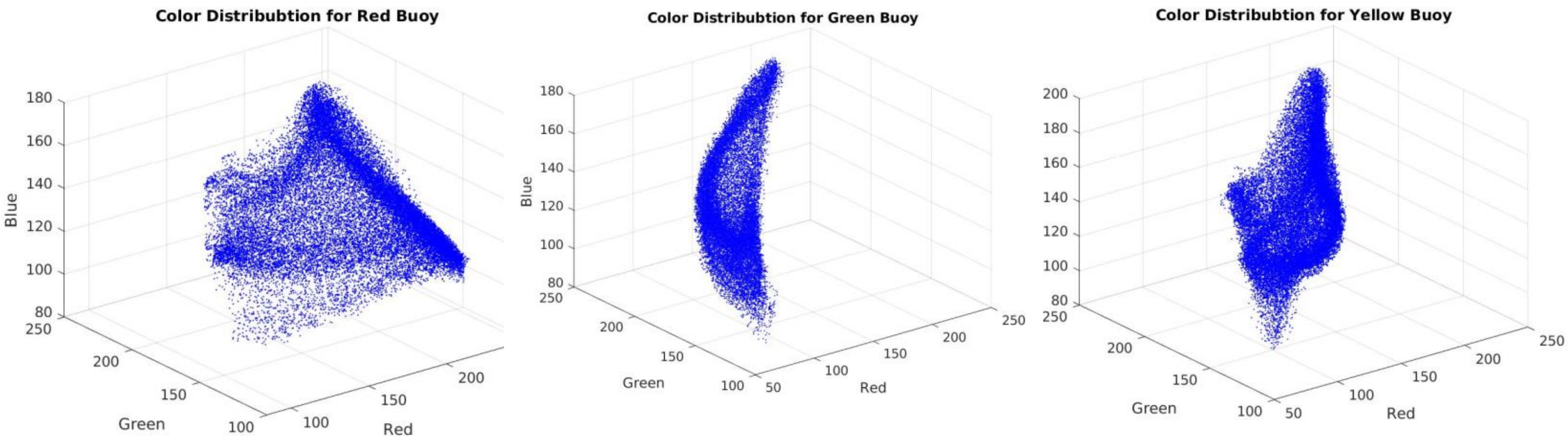
1) **Training Data and Test Data:**
   - The images were extracted from the video with a frame rate of 5 per second hence generating 200 images. The first 42 images consist of all the three buoys. Therefore, the alternate images from the first 42 along with every 10^{th} image after the first 42 are taken as the training data.
   - The rest of the images are my test data.
   - For the training data, each buoy was extracted by using roipoly () function manually.
   - The Training and the Test data can be found in the Images Folder. They are labelled according to their color and frame number.
2) **Average Color Histogram.**
   - Run averageHistogram.m to generate the average colored histogram.
   - Methodology:
     - Using imgaussfilt to smoothen the input image data.
     - Extract the intensity values for the red, green and yellow buoy using the roipoly mask for the training set.
     - Reshape the row vector of intensity into an image and use imhist to get bincount and values data for the histogram. Do this for all buoys. (gethistdata.m).
     - Take average of this data over all the images of training set.
     - Plot the average colored histogram.



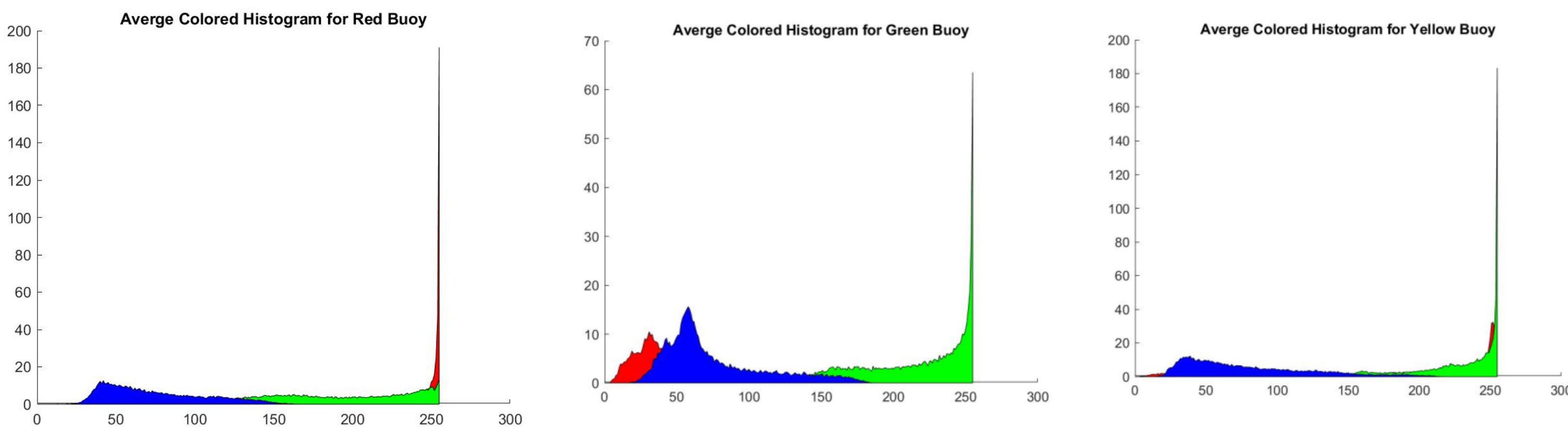




- Inference from the above average colored histograms:
    - **Red Buoy:** The red color intensity is in the range from 180-255, indicating higher presence of red color. The green and blue color are on the lesser intensity range.
    - **Green Buoy:** The green color intensity is in the range from 180-255, indicating higher presence of green color. The red and blue color are on the lesser intensity range.
    - **Yellow Buoy:** As yellow is a mixture of red and green, we can see that both red and green colors are on the higher intensity side.
- The increased number of count for non-dominant colors is because of the noise during cropping and because the color of buoy is not entirely pure.
- I have also plotted the scatter plot for all the three buoys as follows



- To get a better idea of the colored histograms or to identify the buoy better, I am also showing the average colored histograms for buoys with equalized histograms.

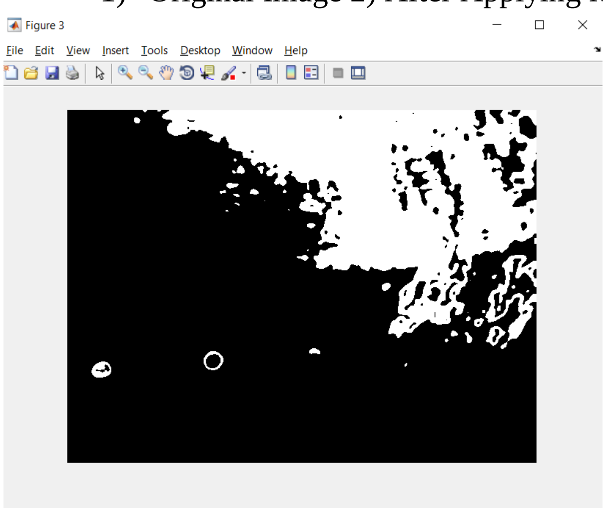


- We can see that after equalizing the image the main color associated with the buoys have the maximum counts with maximum intensities, which is the favorable outcome.

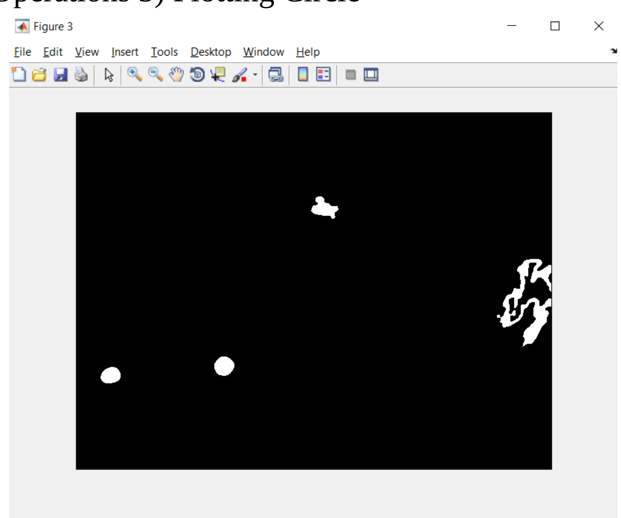
3) Segmentation of Buoys:

- Here I have implemented 1D gaussian to form a pdf and segment the buoys.
- Methodology:
    - Calculate the mean and variance for the main color channel of respective buoys. (E.g.: red color for red buoy and so on).
    - Now calculate the probability density function for each pixel of the image from test set using the mean and variance generated.
    - The probability density function would basically give the probability of let’s say red color on that pixel given the mean and variance as calculated for the red buoy.
    - Now applying thresholding on the pdf to get the pixels with max probability of having the needed red color or identifying the pixels containing the red buoy.
    - Thresholding done using standard deviation and formed a binary image.
    - After fine tuning the images, the threshold limits set are as follows:
        - For Red Color: probR> 2.2*std2(probR);
        - For Green Color: probR> 2.5*std2(probR);
        - For Yellow Color: probR> 2.7*std2(probR);

- **Further Operations on binary image as follows:**
  Outputs for the first frame are shown. Similar Operations for all successive frames are carried out.
  Output for red and yellow buoy:
  1) Original Image 2) After Applying Morphological Operations 3) Plotting Circle

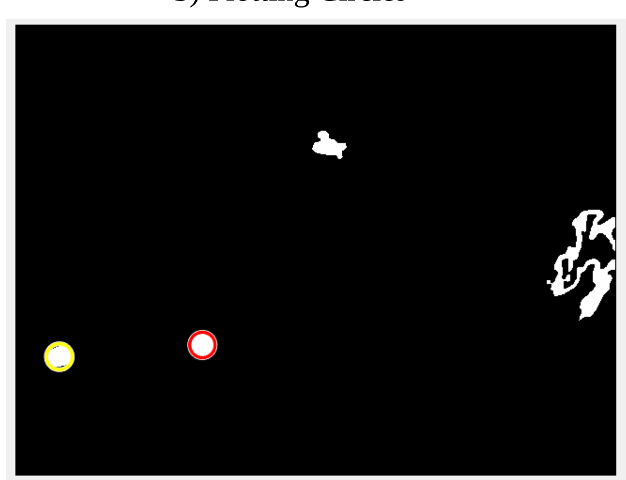


(1) Original Image



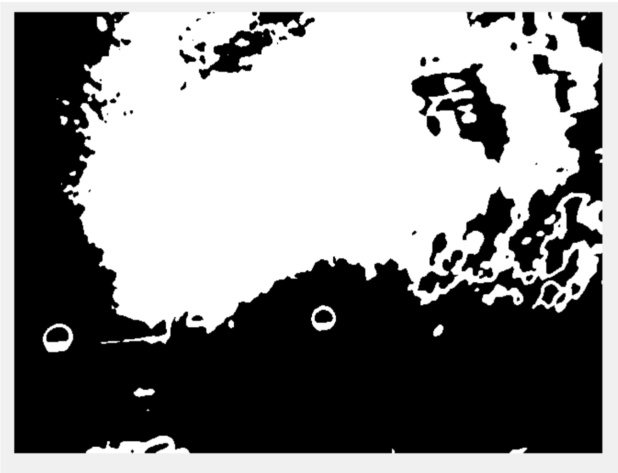
2) After Applying Morphological Operations

3) Plotting Circles

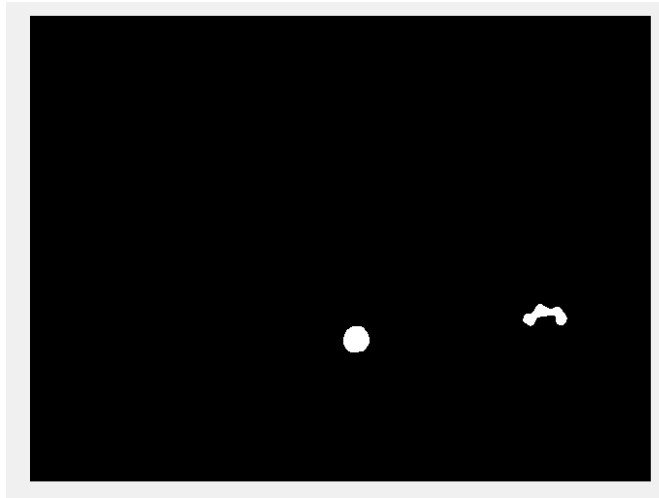


Output for Green buoy:

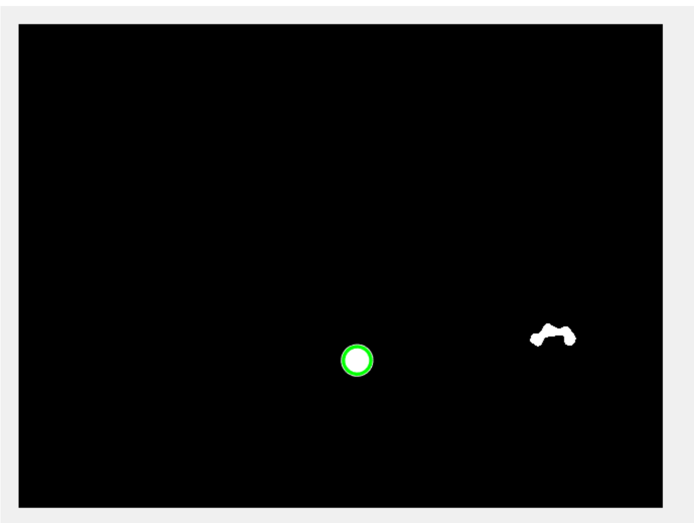
1) Original Image 2) After Applying Morphological Operations 3) Plotting Circle



1) Original Image



2) After Applying Morphological Operations

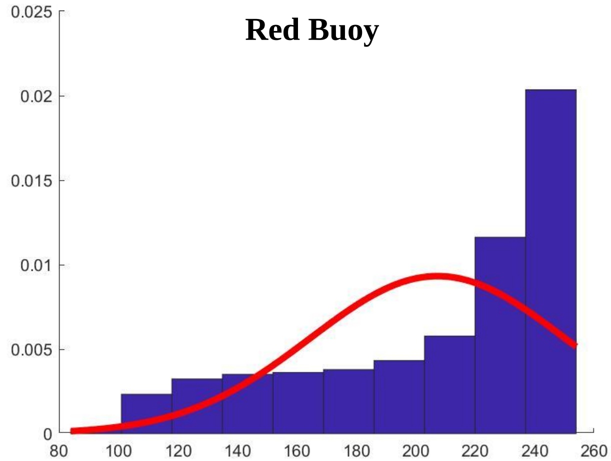


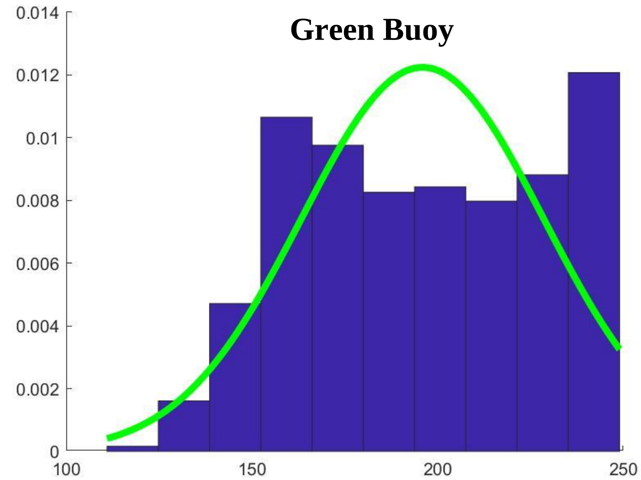
3) Plotting Circle

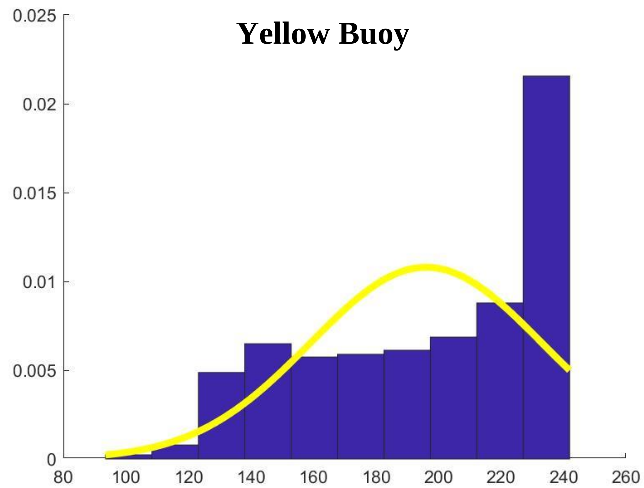
4) Color Segmented Image

- The same sequence of operations is carried out for all frames. We can see in the video that sometimes, the green circle shows somewhere else, this is due to the inability to remove noise completely.
- Also, these results are for a 1D gaussian. As we will see later, when we increase the number of gaussians, we get much better results.

- The 1D gaussian plots for each major color with respect to the buoy is as follows:







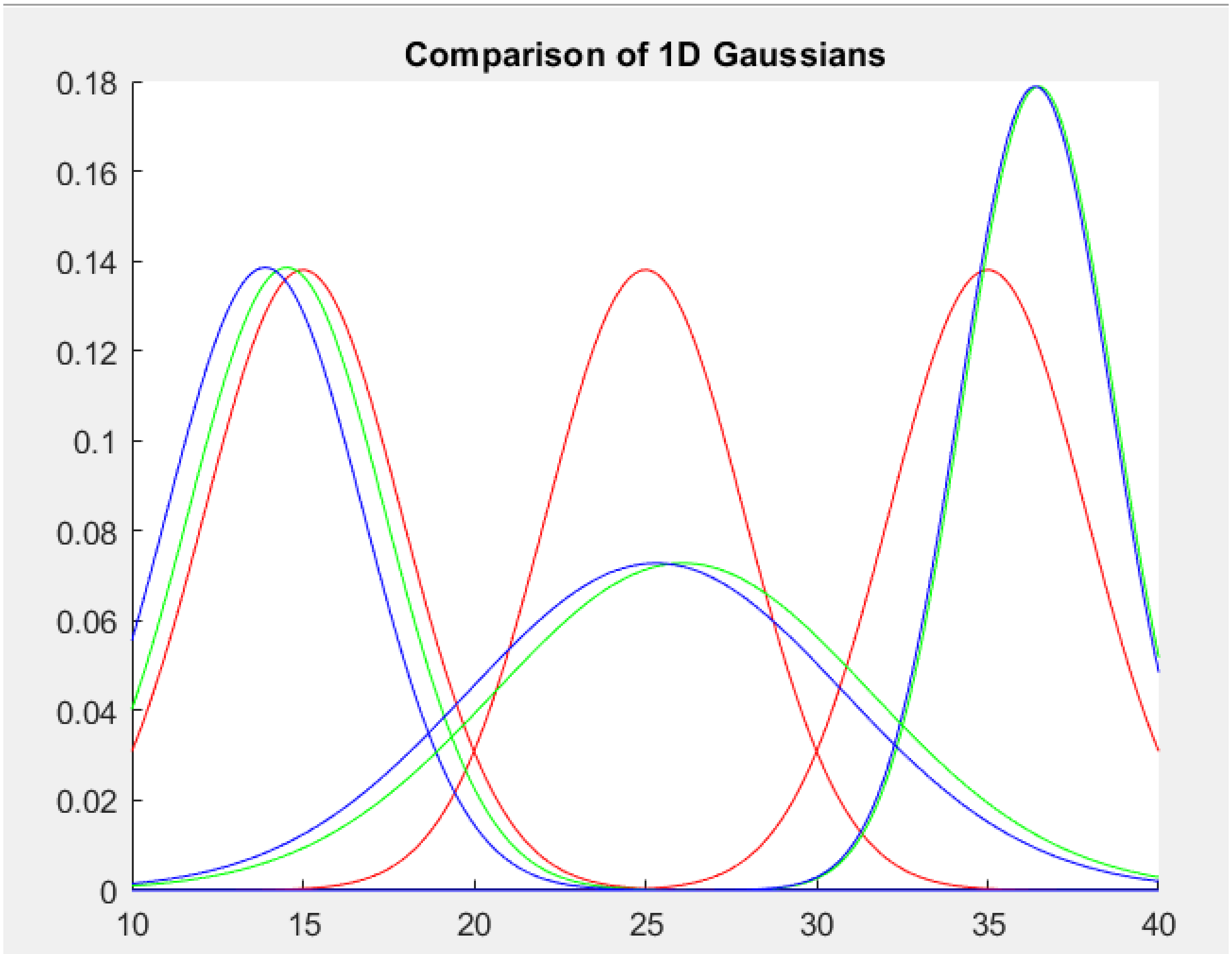
4) Reasoning for the Methedology used:

- The mean and variances were calculated for the main color for each buoy, as the probabilty that for instance red color for red buoy will be maximum.
- The thresholding on standard deviation was carried out by observing the images and finding the sweet spot between noise and buoy detection.
- All the morphological operations were optimzed by observation.
- The logic on finally plotting the circles is based on the intensity obtained from regionprops() function and the major and minor axis obtained for each detected object in the binary image.
- This is again tuned by obervation over all the 200 frames.

Part 1: Gaussian Mixture models and Maximum Likelihood Algorithm

- The task is to recover the model paramters for the combined data used to generate 3 1D gaussains.
- Random data set of 10 points each is generated using rand function.
- Apllied Expectation Maximization Algorithm as per the equations given in Bishop - Pattern Recognition And Machine Learning. Pg- 438.
- Number of Iterations are set to 50, as Log Liklihood function converges in this time.
- Let's now see the comaprison between the input, my EM algo and Matlabs EM algo using fitgmdist.

Red = Input ; Blue = My EM algorithm Implementation ; Green= Matlabs EM (fitgmdist()).

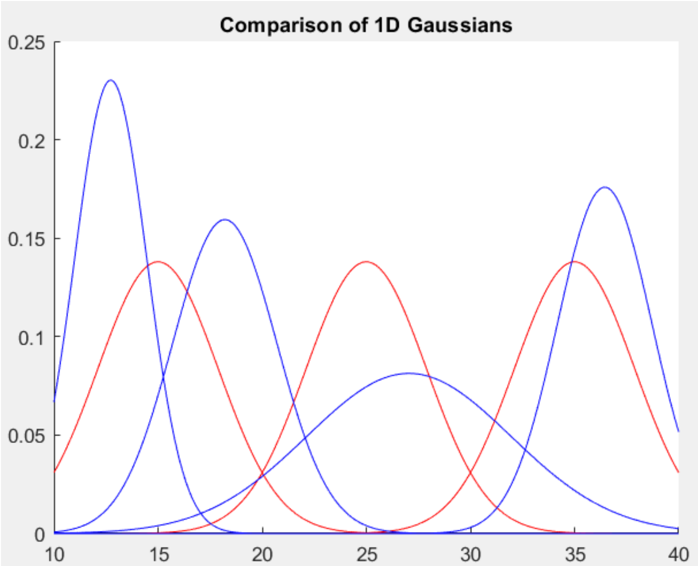


	1st Gaussian	2nd Gaussian	3rd Gaussiam
Input	Mu= 15 ; Sigma=8.3584	Mu= 25 ; Sigma=8.35	Mu= 35 ; Sigma=8.35
My EM algo	Mu= 14.52 , Sigma=8.29	Mu=26.09 ; Sigma=30.02	Mu= 36.48 ; Sigma=4.97
Matlabs EM algo	Mu= 13.89 ; Sigma= 6.12	Mu= 25.3 ; Sigma=34.42	Mu= 36.39 ; Sigma=5.21

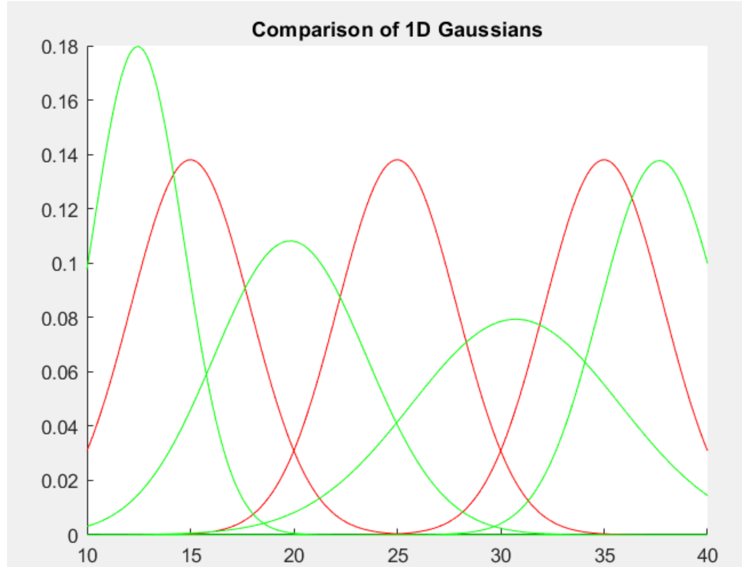
- We can see that two gaussian fit perfectly to the input data whereas the middel gausian sigma is not fitting properly, this can be accounted due to the fact that, I have use linspace data to plot the gaussians.
- Another thing to observe is that the difference between my model paramters generated and matlab's model parameters generated is very less and therefore I conclude that the Em algo implementation is correct.

Effect of generating 4 gaussians for 3 gaussian input data.

Input and My EM algo

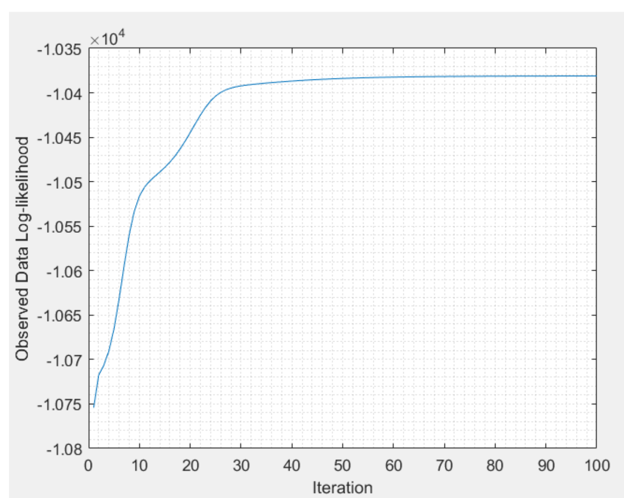


Input and Matlab's Em algo



	1^{st} Gaussian	2^{nd} Gaussian	3^{rd} Gaussiam	4^{th} Gaussian
Input	Mu=15, Sigma=8.3584	Mu=25;Sigma=8.35	Mu=35;Sigma=8.35	
My EM algo	Mu=12.72,Sigma=6.26	Mu=27.04;Sigma=24.06	Mu=36.44;Sigma=5.14	Mean=18.20, Sigma= 3.0033
MatlabsEM algo	Mu=12.45;Sigma= 2.41	Mu=30.72;Sigma=15.10	Mu=37.67,Sigma=2.16	Mean=19.82 Sigma=16.75

- When we generate 4 gaussians for 3 gaussian input, we can see that all the 3 previous gaussians are shifted to fit another gaussian. This disrupts the previous fits and therefore is not recommended to be used to generate probability distribution function. The fourth gaussian is usually generated in between a valley of the previous gaussians.
- Another thing to note is that the mean values though are shifted , but are not entirely different. The maximum deviation of 3.54 is observed.
- Also, the output by matlabs function and my EM function is almost same.
- I have also shown the convergence of log likelihood function as shown in the adjacent figure.
- We can see that the EM converges in almost 50 iterations.
- The implementation of D dimesnional gaussians is shown in the next part.

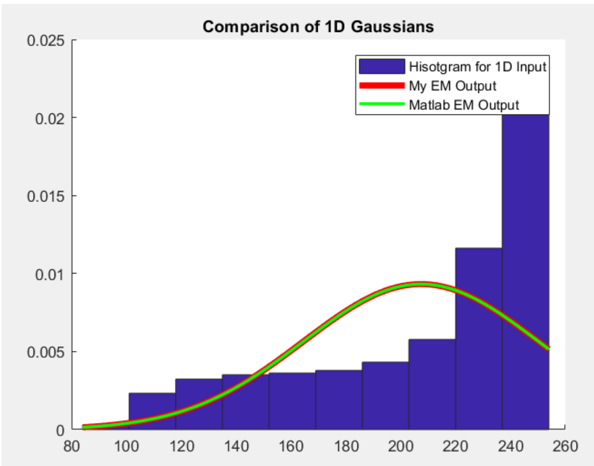


Part 2: Color Model Learning

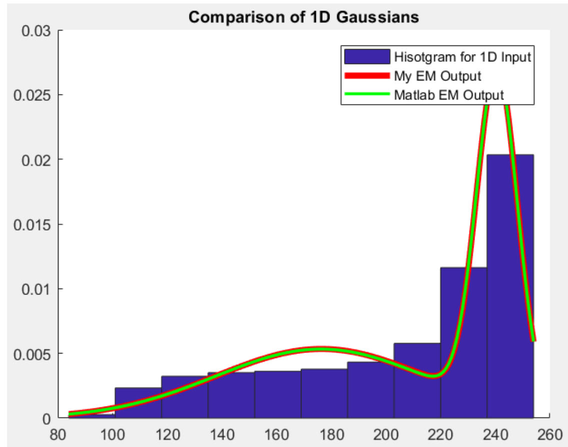
This part is divided into two sections. In the first section we will see how many gaussians are needed to fit 1D data input for each major color of thee buoy. Which was the input in Part0-Color Segmentation. In the second section we will calculate the model parameters for the 3D input of (RGB colors) for each buoy with different gaussians.

Section 1: 1D Input. (Major Color for each buoy respectively).
Red Color for Red Buoy.

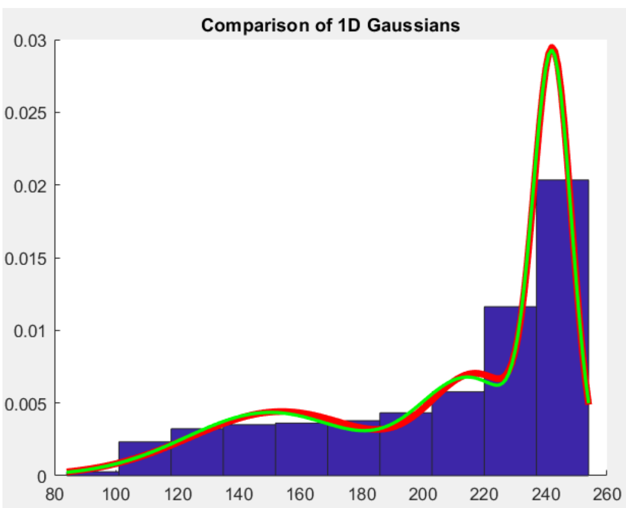
1 Gaussian



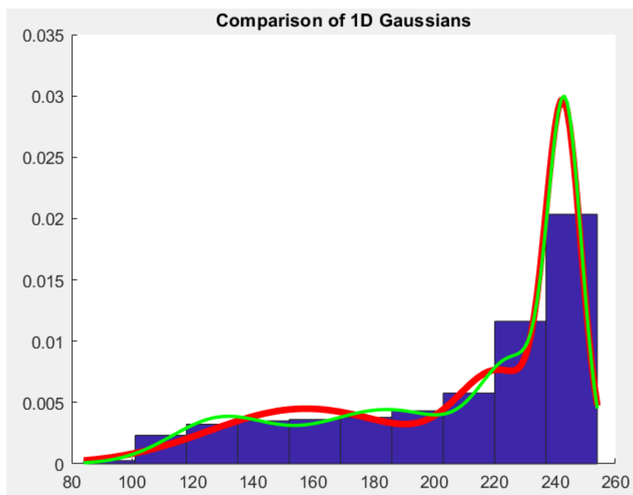
2 Gaussian



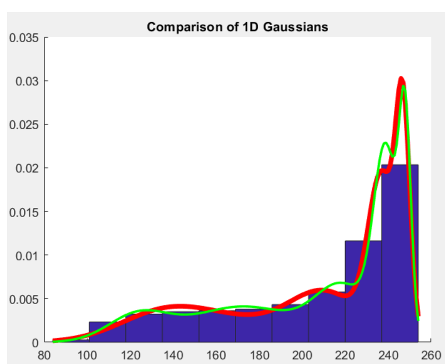
3 Gaussian



4 Gaussian



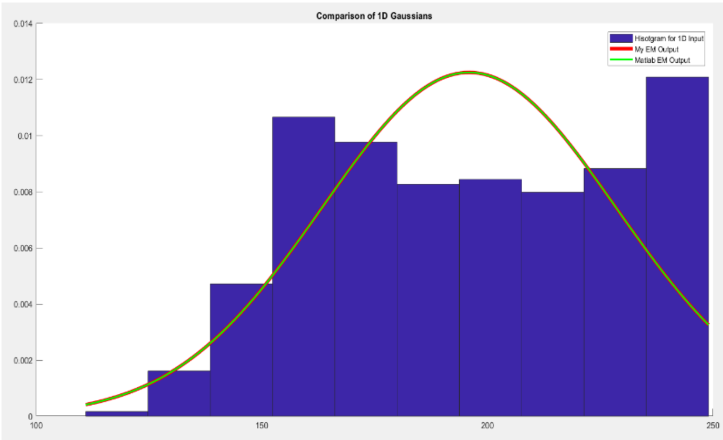
5 Gaussians



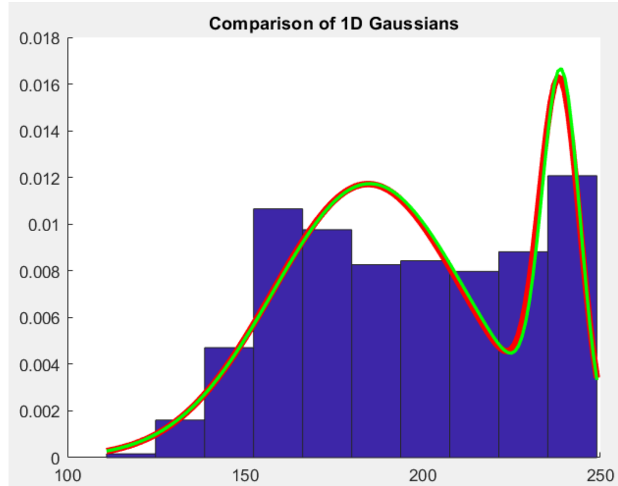
- From the above generated comparisons, we can see that generating 4 gaussians best fit the model. When we generate 5 gasusians, there are unneceesary peaks and therefore, for the **1D input of red channel for red buoy, 4 gaussians best fit the model.**

Green Color for Green Buoy

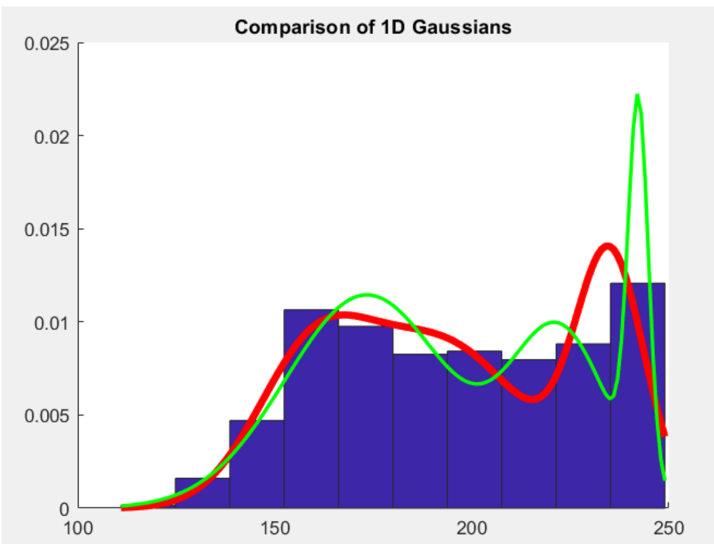
1 Gaussian



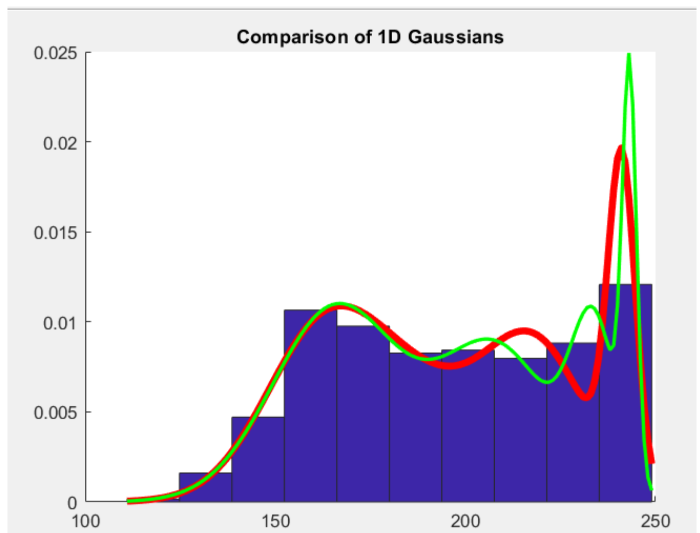
2 Gaussian



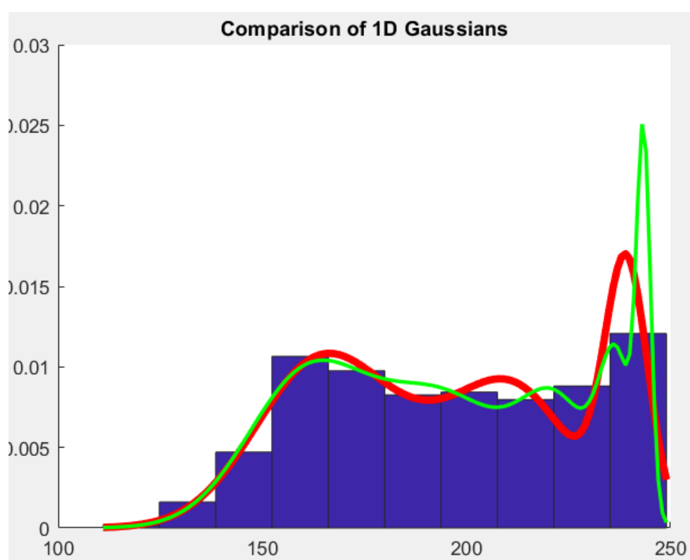
3 Gaussian



4 Gaussian



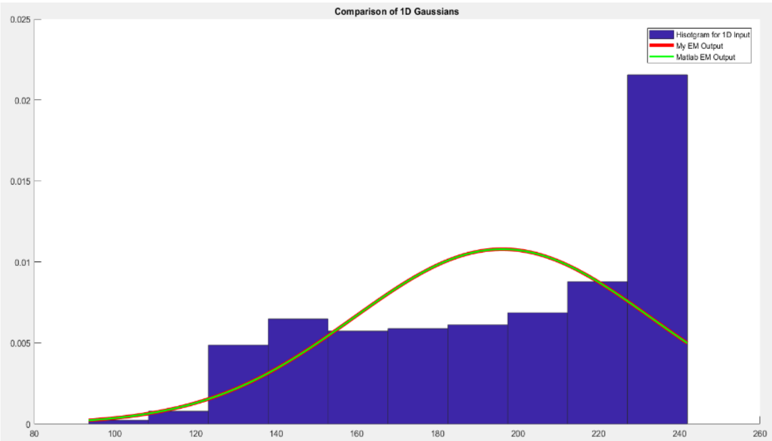
5 Gaussians



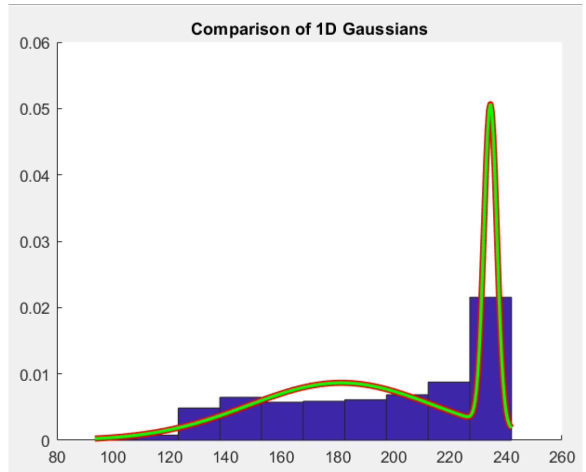
- From the above generated comparisons, we can see that generating 3 gaussians best fit the model. When we generate 4 or 5 gasusians, there are unneceesary peaks and therefore, for the **1D input of green channel for green buoy, 3 gaussians best fit the model.**

Yellow Color for Yellow Buoy

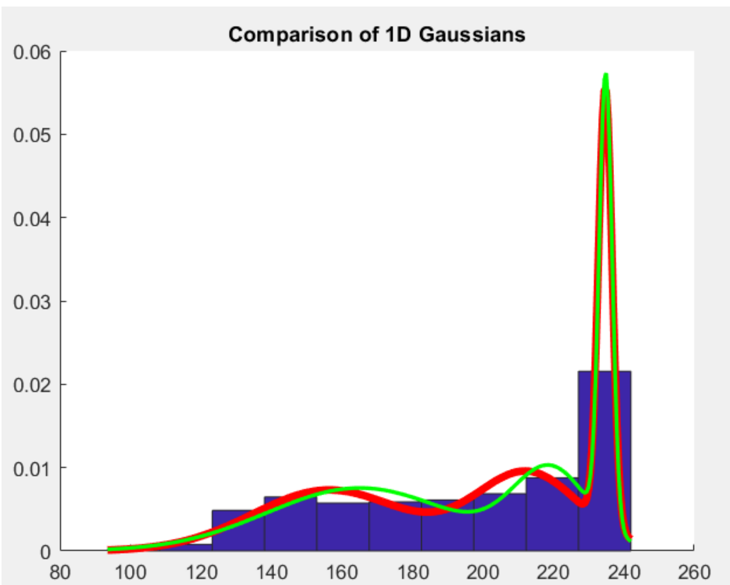
1 Gaussian



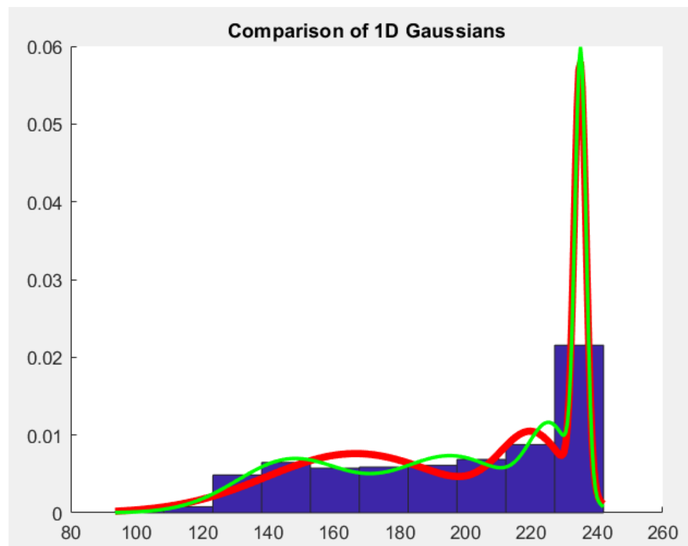
2 Gaussian



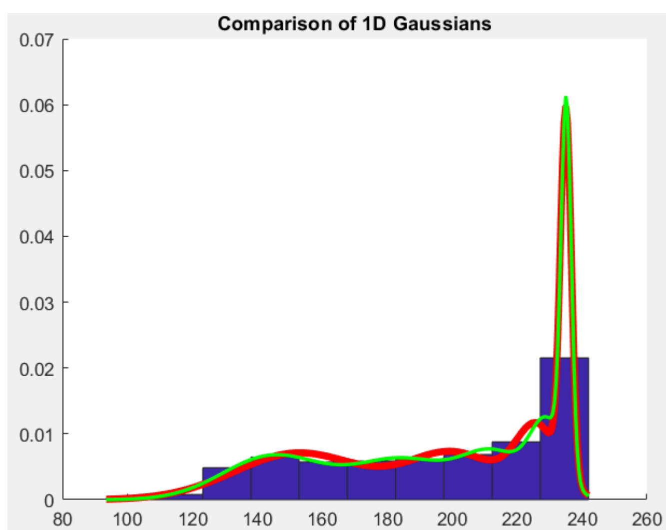
3 Gaussian



4 Gaussian



5 Gaussians



- From the above generated comparisons, we can see that generating 3 ,4 or 5 gaussians equally well fit the data. But, I will select 3 gaussians as it will have reduced processing time.Therefoe, **1D input of red plus green channel for yellow buoy, 3 gaussians best fit the model.**

Section 2: Calculating Model paramters for 3D input.

Model Paramters for Different Number of Gaussians are as follows:
The 3D input is as follows for each buoy : [Red, Green, Blue]
1) Red Buoy

a) 1 D:

```
Gaussian mixture distribution with 1 components in 3 dimensions
Component 1:
Mixing proportion: 1.000000
Mean:   207.3756   179.8233   126.8587
```

b) 2D:

```
Gaussian mixture distribution with 2 components in 3 dimensions
Component 1:
Mixing proportion: 0.535594
Mean:   177.1859   159.2625   115.0316

Component 2:
Mixing proportion: 0.464406
Mean:   242.1931   203.5357   140.4988
```

c) 3D:

```
Gaussian mixture distribution with 3 components in 3 dimensions
Component 1:
Mixing proportion: 0.404578
Mean:   243.6146   206.7766   142.2169

Component 2:
Mixing proportion: 0.375267
Mean:   159.0276   153.7449   111.4303

Component 3:
Mixing proportion: 0.220155
Mean:   223.1915   174.7434   124.9334
```

This data will later be used to see how many number of gasussian will give us the best and the most fitting probability density function.
Note: The gaussian object generated by my code matched matlab's code exactly till k=2 gasussians and it is slightly different for k=3 gaussians.

Similarly such data, is generated for all three buoys.

After generating binary image by calculating pdf as done in part0 for various number of gaussians, I come to the conclusion that 3 Gaussians give the best binary image output. This can be seen by observing the binary image videos produce for each count of the gaussians. The videos can found in Output folder of part 3.

Part 3: Buoy Detection

- As per the discussion above, when given a 3 Dimensional input of [R G B] intensity values, a total of 3 Gaussians give the best binary image output.
- When we compare the binary image output of 3Dimensional input with 1 Dimensional input we see that the binary image of 3D input has a lot of noise near the buoys and therefore it is difficult to segment them. Even after applying thresholding using standard deviation, the noise near the buoys was hindering the segmentation process. Compared to 1D input, even though we identify multiple colored buoy, the noise near the buoys is less and therfore easier to segemnt.
- Therefore, for the final buoy detection methdod, I have used 1D input with multiple gaussians.
- The input is as follows:
    - Red intensity values for red buoy.
    - Green intensity values for green buoy.
    - (Red + Green)/2 intensity values for yellow buoy
- The number of gaussians are seelcted based upon the histogram fitting carried out in part 2 of the report.
    - Therefore, I have chosen 4 Gaussians for Red Color, Red Buoy
    - 4 Gaussians for green color, green buoy
    - 4 gaussiand for red and green color, yellow buoy.
- The Gmobj formed formed for each buoy is as follows:

Red Buoy

```
Gaussian mixture distribution with 4 components in 1 dimensions
Component 1:
Mixing proportion: 0.272113
Mean:  209.2585

Component 2:
Mixing proportion: 0.160856
Mean:  247.6135

Component 3:
Mixing proportion: 0.262160
Mean:  145.0069

Component 4:
Mixing proportion: 0.304872
Mean:  238.0958
```

Green Buoy

```
Gaussian mixture distribution with 4 components in 1 dimensions
Component 1:
Mixing proportion: 0.100370
Mean:  243.1534

Component 2:
Mixing proportion: 0.313223
Mean:  207.9809

Component 3:
Mixing proportion: 0.132676
Mean:  233.8067

Component 4:
Mixing proportion: 0.453731
Mean:  166.1155
```

```
Gaussian mixture distribution with 4 components in 1 dimensions
Component 1:
Mixing proportion: 0.231939
Mean:   235.0348

Component 2:
Mixing proportion: 0.286772
Mean:   147.2787

Component 3:
Mixing proportion: 0.314759
Mean:   196.0623

Component 4:
Mixing proportion: 0.166530
Mean:   226.0018
```

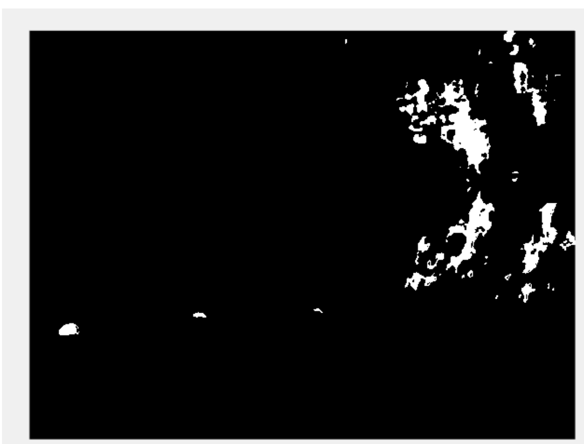
Yellow Buoy

- The thresholding limits are set as:
  - For Red Color: probR> 1.5*std2(probR);
  - For Green Color: probR> 2*std2(probR);
  - For Yellow Color: probR> 1.7*std2(probR);
- **Further Operations on binary image as follows:**

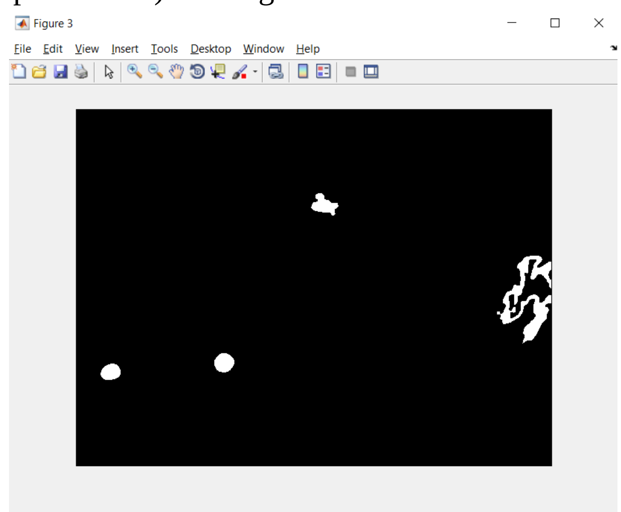
  Outputs for the first frame are shown. Similar Operations for all successive frames are carried out.

  Output for red and yellow buoy:

  2) Original Image 2) After Applying Morphological Operations 3) Plotting Circle

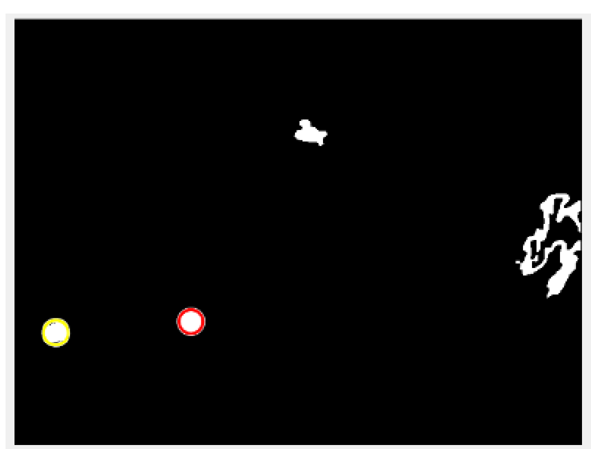


(2) Original Image

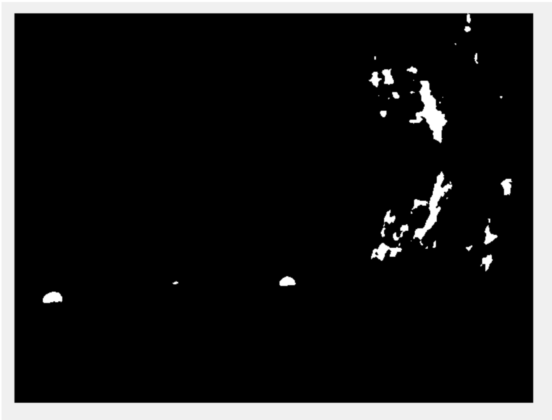


2) After Applying Morphological Operations

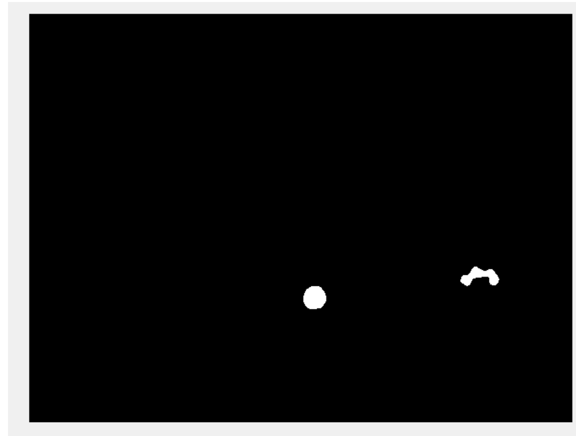
3) Plotting Circles



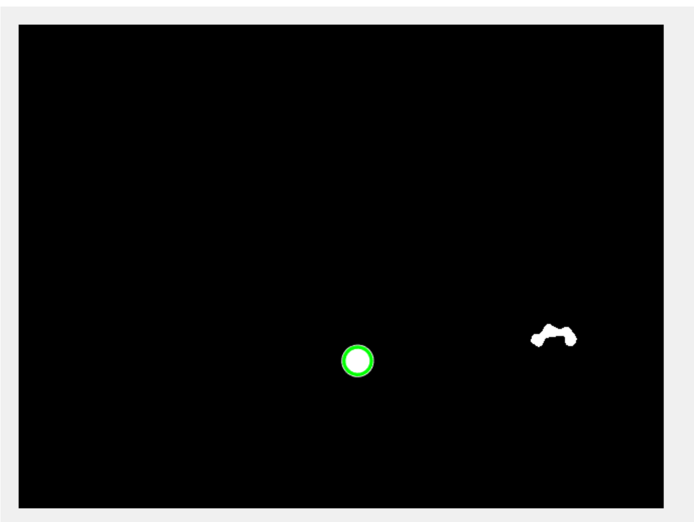
Output for Green buoy:
2) Original Image 2) After Applying Morphological Operations 3) Plotting Circle



1) Original Image



2) After Applying Morphological Operations



3) Plotting Circle

4) Color Segmented Image

4) Extra Credit:

Use of other color Spaces.

Using the RGB workspace comes with the inherent problem that none of the images are of pure color under idea conditions. Noises like background color, lighting, motion makes it difficult to segemnt the buoys. In my opinion the use of HSV or LAB workspace should help the color segemntation as HSV or Lab are more adaptive to the changes in the images such as lighting, shadow or any such noises. The hue would stay relatively constant for the buoys, the differences in lighting would be represented through the value component and the "colorfulness" of the each color through the saturation component. The hue and saturation values remains stable, despite the changes in lighting This was although, a bit contradictory with the **HSV** workspace because of the following observation:

- I faced difficulty in segmenting the green color in this workspace. I think this is because HSV works on the hue saturations value and because of the green color background, segmenting the green buoy became all the more difficult.

Using **Lab workspace** gave excellent results with the input video. These results were almost as good as the results obtained by taking multivariate gaussians with 1D input for RGB image. This may be accounted for the reason that, Lab worspace have specific planes for green /red channels and yellow/blue channels.

Alternative Color Space

An alternative to the conventional, RGB and HSV color space could be RGBY (Red Green Blue Yellow). This color space would add an additional value to the image at includes a yellow layer. This will help prevent the issue of false detection of green buoy. Since yellow is a combination of red and green layers, sometimes the yellow buoy would be recognized as a green buoy if the green tint was too high. Additionally, the addition of the yellow layer would allow for the noise that occurs from sunlight (and other light sources) since it is generally a yellow tint.This could thus negate the effect of changing light intensty on the rgb values..

5) References

1) Lecture notes

2) Bishop - Pattern Recognition And Machine Learning

3) Matlab-Docs