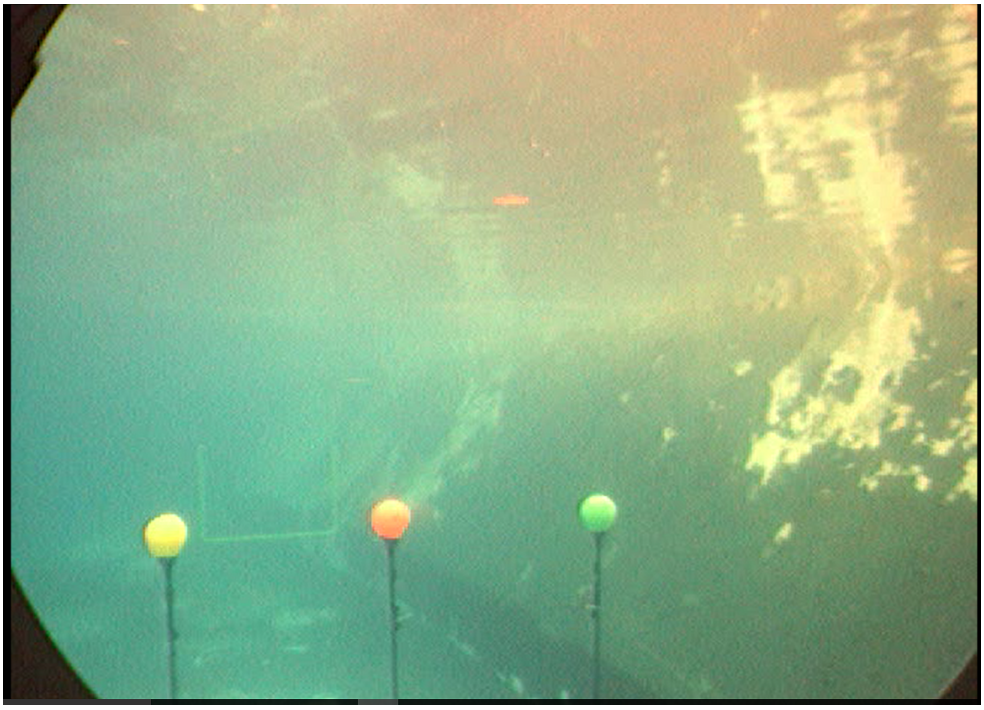
ENPM 673, Robotics Perception
Project 3: Color Segmentation
Due on: Thursday, April 19, 2018

A **buoy** is a distinctively shaped and colored floating device, anchored to the bottom, for designating moorings, navigable channels, or obstructions in a water body.

It is sometimes used as underwater markings for navigation. The dataset for the project is a camera view of an underwater vehicle with three distinctly colored buoys in surrounding. The goal is to segment and detect the buoys in the video sequence (detectbuoy.avi).



Since, the buoys are distinctly colored and shaped, one approach to segment them is to use color. But, because of noise and changing light intensities (due to properties of the environment), it will be difficult to segment the buoys, using vanilla color threshold based techniques. Hence, we will use Gaussian Models to learn the color distribution, and use the model to segment and detect the buoys.

In this project your task is to obtain a tight segmentation of the buoys. Put a contour enclosing each of the buoys with the color of the contour matching the detected buoy. That is, a red buoy, should have a red contour etc.

Part 0. Data preparation and Vanilla Approach **(10)**

1. Extract and Save in memory, the samples of Buoy images from the video sequence provided in the data-set. Extraction of the samples can be done by using the **roipoly** function in MATLAB.

   (It is advised to divide the video sequence into training frames and testing frames, such that the training frames are used to extract model parameters using EM and the testing frames are used to evaluate the performance of the model. You DO NOT want to over-fit your training data!).

Submission Details:

- Create a ColorSeg folder under P3_submission folder which will be uploaded as YourDirectoryID_proj3.zip
- Save the Training Frames in ColorSeg/Images/TrainingSet/Frames/<FrameNo>.jpg and ColorSeg/Images/TestSet/Frames/<FrameNo >.jpg .
- Save the cropped Buoy images in ColorSeg/Images/TrainingSet/CroppedBuoys/<Color_FrameNo>.jpg Ex: (ColorSeg/Images/TrainingSet/CroppedBuoys/R_001.jpg)

MAKE SURE YOU SUBMIT ONLY A **.zip** FILE AND STRICTLY FOLLOW THE DIRECTORY STRUCTURE AS SUGGESTED.

2. For each colored buoy, compute and visualize the average color histogram for each channel of the sampled images (RGB image).

Submission Details:

- Submit the script ColorSeg/Scripts/Part0/averageHistogram.m which reads the images from ColorSeg/Images/TrainingSet/CroppedBuoys/* and saves the plot in in ColorSeg/Output/Part0/<Color>_hist.jpg (ex. Output/R_hist.jpg for red).

3. Suppose, we model the color distribution of the buoys using a 1-D Gaussian. Design and implement the scheme to segment the buoys. Also detect the buoys based on their color (red green and yellow).

Submission Details:

- Submit the script ColorSeg/Scripts/Part0/segment1D(Frame).m which takes the input frame as parameter and displays the output of images with colored contours around the buoy.
- Save the segmented images in ColorSeg/Output/Part0/seg_<FrameNo>.jpg
- Save the plot of the 1-D Gaussian Model used in ColorSeg/Output/Part0/gauss1D.jpg

4. Describe the reasoning for the decisions you take for the scheme you propose.

Part 1. Gaussian Mixture Models and Maximum Likelihood Algorithm (30)

A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance of the data as well as the centers of the latent Gaussians.

The main difficulty in learning Gaussian mixture models from unlabeled data is that one usually doesn't know which points came from which latent component (if one has access to this information it gets very easy to fit a separate Gaussian distribution to each set of points).

Expectation-maximization is a well-founded statistical algorithm to get around this problem by an iterative process. First one assumes random components (randomly centered on data points, learned from k-means, or even just normally distributed around the origin) and computes for each point a probability of being generated by each component of the model. Then, one tweaks the parameters to maximize the likelihood of the data given those assignments. Repeating this process is guaranteed to always converge to a local optimum.

1. Generate data samples from 3 1-D Gaussians, with different means and variances. (You can generate 10 samples from each distribution)

2. Implement the Expectation Maximization (EM) algorithm according to Equation 5.29 - 5.32 on slide 31 of "Image Segmentation.pptx." CM Bishop's Lecture Slides, provide background information and more details.

3. Recover the model parameters for 3 Gaussians (means and variances of the 3 Gaussians mentioned in step 1 using your implementation of Expectation Maximization.

Submission Details:

- Submit the script ColorSeg/Scripts/Part1/EM(N, data, plot_path).m with the number of Gaussians N, data, and plot path as input, and output the array NxD mean and NxDxD covariance model parameters. The function should plot the computed Gaussians and save it to the relative path mentioned in plot_path (if plot_path = '', the plot and save operation should be avoided).

Save the plot of the model for the data generated in step 1 with N = 3, in ColorSeg/Output/Part1/EM1D3N.jpg

4. Try to recover the model parameters for 4 Gaussians (means and variances of 4 Gaussians that fit the data generated by 3 in step 1 (Save the plot of your output as described in step 3). Discuss what you observe.

Submission Details:

- Save the plot of the model for the data generated in step 1 with N = 3, in ColorSeg/Output/Part1/EM1D4N.jpg

5. Confirm your implementation for generalized D-Dimensional Gaussians.

Part 2. Color Model Learning **(20)**

Now, we extend the clustering concept to use it for our goal of color segmentation.

1. For each colored buoy, compute and visualize the average color histogram for each channel of the sampled images (RGB image: use Part 0 output). This should provide some intuition to decide on the number of Gaussians **[N]** to fit the color histogram. Also, can you determine the dimension of each Gaussian **[D]** for your model?
2. Use the previous implementation of Expectation Maximization to compute the model parameters (Means and Variances of N D-Dimensional Gaussian)

Submission Details:

- Save the plot of the model in ColorSeg/Output/Part2/EM_<Color>.jpg for each colored buoy.

Part 3. Buoy Detection **(40)**

1. Implement a Buoy Detection function which will take a frame id (from the video sequence) and the computed model parameters (from Part 1) as input and generate a color-segmented binary image.
2. For each frame in the video sequence, compute the color-segmented binary image for each buoy and compute the corresponding contours and center.
3. Draw the bounding contours around the detected buoy (color of the contour should be the color of buoy detected).
4. The assignment will be graded on how tight the bounding contours cover the buoys.

Submission Details:

- The Buoy Detection function should be: ColorSeg/Scripts/Part2/detectBuoy(FrameID,means,covariances(R)…means,covariances(Y),plot_path).m.
- The function should
  - Display and Save the binary image ColorSeg/Output/Part3/binary_<FrameID>.jpg
  - Display and Save the image with contours in ColorSeg/Output/Part3/Frames/out_<FrameID>.jpg
- Write a function to generate a video sequence by reading images from ColorSeg/Output/Part3/Frames/and save it in ColorSeg/Output/Part3/Video/

Extra Credit: **(20)**

The Color Segmentation scheme used the RGB representation of the images. Are there any other representations of images that would be more effective to the algorithm (Yes/No, defend your reasoning)?

Can you come up with your own representation that encompasses the color-distribution information which can improve the performance of the algorithm?

Implement the algorithm for any other image representation schemes and compare the results.

A detailed report (pdf) of the steps taken, methods used, important plots – with analysis, difficulties faced and reasoning of the solution should be submitted as P3_submission/Reports/ColorSeg.pdf