# ENPM673 - Perception for Autonomous Robots
# Project 3

Cheng Chen, Haixiang Fang, Timothy Werder

Due April 6, 2020

## Output Video Link

https://youtu.be/S4EUAGH3LiI

## Introduction

Image classification has become a pivotal role in the field of robotics and computer vision. The ability to classify different objects has led to powerful strides in the field of automated driving, medicine and general machine learning. In this project we attempted to implement an unsupervised learning model, by applying a Expectation maximization against a Gaussian mixture model to learn and differentiate three different underwater buoys from each other and their backgrounds.

## Creating Train Set

The first step for any machine learning algorithm is creating or finding the training data. For our project we used mouse clicks to define a center point and an outer point. These would then be used to create an inner bounding zone which was then cut from the original image.

We then iterated through 21 frames (some sequential, some non-sequential) to get a good set of training data across different lighting conditions for all three buoys.

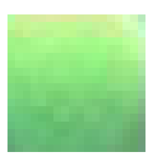


Figure 1: Sample of training imagery: Green

Figure 2: Sample of training imagery: Orange

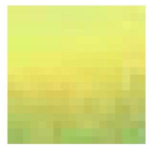


Figure 3: Sample of training imagery:Yellow

These data sets were then saved off to separate folders so they could be independently labeled for training.

## Defining Clusters by Histogram

Once we had the data sets created, each batch was loaded into a histogram that separated red, green and blue values. This was done across each training picture in each batch. We then summed the individual histograms in each batch to create the overall histogram for the desired buoy.

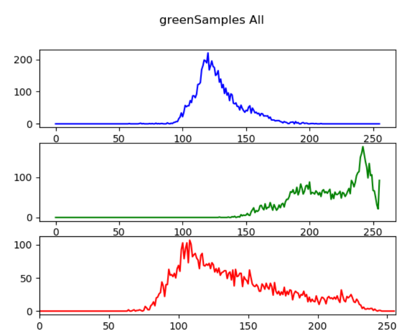



Figure 4: Histogram of training imagery: Green

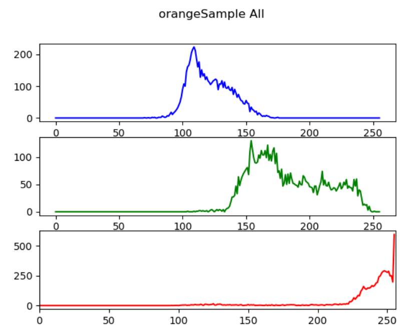



Figure 5: Histogram of training imagery: Orange

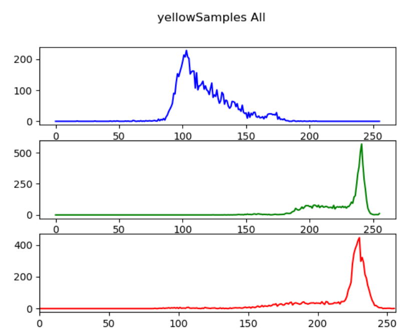
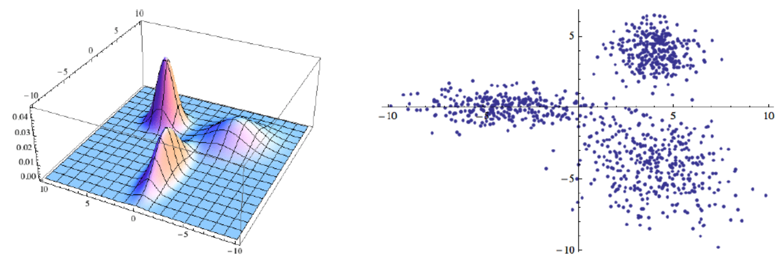



Figure 6: Histogram of training imagery: Yellow

Based on the histograms we were then able to intuitively choose K values for the Gaussian Mixture Model.

# 3D Gaussian Mixture Models

If you are trying to find a color in different lighting conditions, such as underwater, a simple Gaussian model will not suffice because the colors may not be bounded well by an ellipsoid. Therefore, we have to come up with a mixture model evaluation with a weighted sum of Gaussian distributions, as shown in Figure 7 below. Without given the parameters of these distributions, we will need to learn the parameters from known points of the training set and then predict which Gaussian distributions a new unknown point would come from.



(a) A probability distribution on $\mathbb{R}^2$. (b) Data sampled from this distribution.

Figure 7: Soft Clustering with Gaussian Mixture Models

# Expectation Maximization Algorithm

In order to calculate the probability of a pixel x belong to a color $C_l$, which expressed as $P(C_l|x)$, we need to estimate $P(C_l|x)$ using Bayes rules:

$$p(C_l|x) = \frac{p(x|C_l)p(C_l)}{\sum_{i=l}^{L} p(x|C_i)p(C_i)}$$

where $p(C_l|x)$ is the conditional probability of a pixel belong to a color label $C_l$ as this pixel observation x is given, aka the Posterior or the responsibility, and $p(x|C_l)$ is the conditional probability of a $x$ pixel might be observed as the color label is given, aka the Likelihood. and $p(C_l)$ is the probability of color class $C_l$ occurs, aka called the Prior. In General, we assume each color class follow one or more Gaussian distribution in color space, shown in equation below.

$$p(C_l|x) = \sum_{i=l}^{k} \pi_i(x, \mu_j, \sigma_j)$$

In order to define which pixel came from which latent distribution, we employ the Expectation-Maximization technique (EM). The steps are as follows.

1. Initialization

Each of the Gaussian distrubation from K clusters will have characteristic parameters $\theta_j = (\mu_j, \sigma_j)$, where are the mean and co-variance respectively of the $j$-th Gaussian.
The co-variance matrices are initialized to be the identity matrix.
The means can be initialized

2. Maximum Likelihood Estimation (MLE)

View this color classification problem from the prospective of model optimization, the problem is to find $\theta_j = (\mu_j, \sigma_j)$ for each cluster, which maximize the correctness of the above model, which measured as $\sum_{i=1}^{N} logP(x|C_l)$. A brief expression of this algorithm is

$$\underset{\{\mu_1,\mu_2,\cdots,\mu_k,\Sigma_1,\Sigma_2,\cdots,\Sigma_k,\pi_1,\pi_2,\cdots,\pi_k\}}{\operatorname{argmax}} \sum_{i=1}^{N} \log p(x_i)$$

Figure 8: Maximum Likelihood Estimation.

where $N$ is the number of training samples.

3. E step

$$p(j \mid x_i, \Theta) = \frac{\alpha_j f_j(x_i \mid \theta_j)}{\sum_{k=1}^{K} \alpha_k f_k(x_i \mid \theta_k)}$$

$$f_j(x \mid \theta_j) = \frac{1}{(2\pi)^{d/2} \mid \Sigma_j \mid^{1/2}} e^{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)}$$

Figure 9: E step of EM.

4. M step

$$\mu_j^{new} = \frac{\sum_{i=1}^{N} x_i p(j \mid x_i, \Theta^{old})}{\sum_{i=1}^{N} p(j \mid x_i, \Theta^{old})}$$

$$\Sigma_j^{new} = \frac{\sum_{i=1}^{N} p(j \mid x_i, \Theta^{old})(x_i - \mu_j^{new})(x_i - \mu_j^{new})^T}{\sum_{i=1}^{N} p(j \mid x_i, \Theta^{old})}$$

$$\alpha_j^{new} = \frac{1}{N} \sum_{i=1}^{N} p(j \mid x_i, \Theta^{old})$$

Figure 10: M step of EM.

5. Alternate until converged We assume convergence occurs if cluster means doesn't change much. To be more specific, if $\sum_i ||\mu_i^{t+1} - \mu_i^t|| < threshold$

## Detecting Buoys Contours

After all parameters of GMM (Gaussian mature model) is learned, we can estimate the pro ability of a which color a pixel may belongs to, defined as conditional probability of a color label Cl given the color observation with the pixel value x.

$$p(C_l|x) = \sum_{i=l}^{k} \pi_i(x, \mu_j, \sigma_j)$$

Once we have isolated the theorized color, we use a Gaussian blur to smooth the image then use canny to approximate the edges and then apply the contours on the edges. We then sort the contours and approximate the outer shell of object in the image. Finally we locate the center of the hull and create a circle with a center x and y and an approximated radius.

Once we have the centers of the circles, we then check the distance between the the most likely false positive object (yellow) and all other centers.

We then threshold against the radius to avoid small superfluous objects and ensure a minimum distance between false positives. If all parameters are matched a circle is drawn.The final results can be seen below.

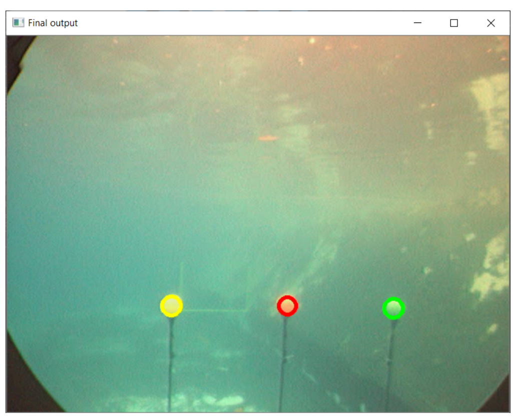



Figure 11: Best frame of all all three buoys tagged with correct color

## Lessons Learned

This project's main difficulty came in two aspects: the project and working on the project remotely.

Project: The main issues came from the concept wasn't explained crystal clear on lecture, but the additional resource (CMSC426 project 1 instruction) provided by TA is helpful. I hope the professor go slower and take the advantage of visual supplemental material, such as video or elaborate reading materials.

However, this project was significantly more difficult by extenuating circumstances, I.E. Sars-Cov-19, which forced our team to isolate and work.

## Reference


[1] @miscelie$_2$019$, title = SoftClusteringwithGaussianMixtureModels(GMM), url = https://fallfordata.com/soft-clustering-with-gaussian-mixture-models-gmm/, journal = Fallfordata, author = Elie, year = 2019, month = Jul$