CMSC426 Lab 1 Report

Authors: Jerry Qian, Kyle Chang, Taru Rustagi

- **Your choice of color space, initialization method and number of gaussians in the GMM**

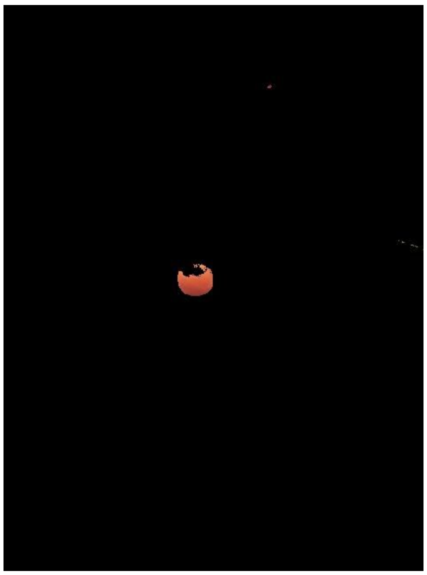


Figure 1: 2 gaussian

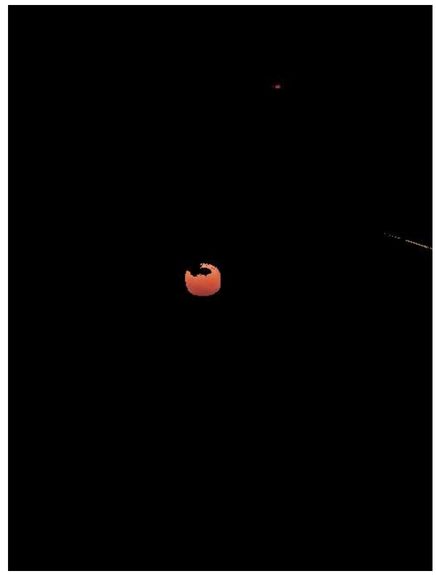


Figure 2: 6 gaussians

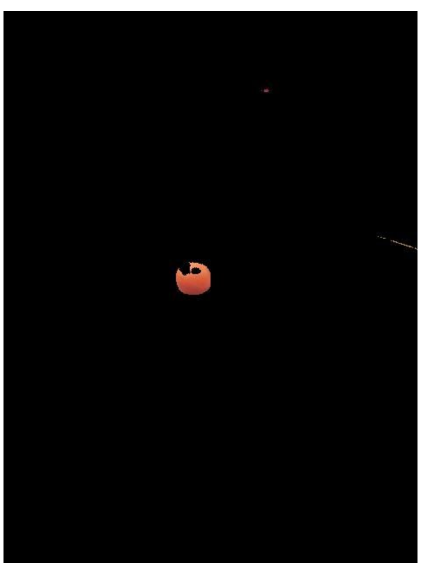


Figure 3: 8 gaussians

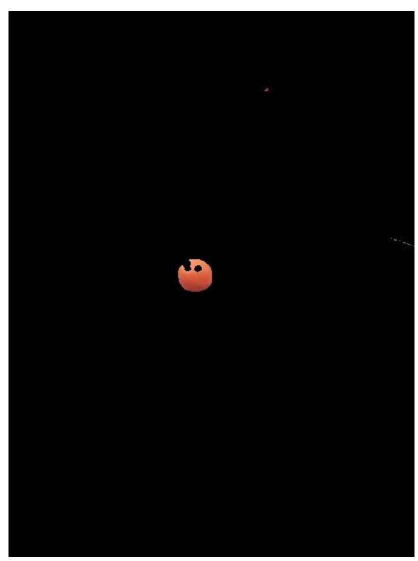


Figure 4: 12 gaussians

In this project, we made various choices in regards to the initialization, colorspace, and number of gaussians. In terms of colorspace, we chose to use rgb colorspace due to a mixture of time constraints and convenience. Given more time we would have experimented with different color spaces to see with space would yield the best result.

However when it came to initialization, we decided to use k means as opposed to random initialization. This is motivated by the intuition that k means will locate starting location that will converge to the optimal solution, as opposed to random initialization which does not have that guarantee.

In terms of number of gaussian we tested various numbers and observed how they influence the result. Here are the results from two gaussian, six gaussian, eight gaussian, and twelve gaussian. As you can see the results for two gaussian is clearly the worst result: In Figure 1, many regions of the orange ball are not captured such as the extreme light oranges. For six gaussian, it does a slightly better job: In Figure 2, the tip of the ball is captured more fully. For eight gaussian, a further improvement: In Figure 3, the tip region grows as our model can more fully capture the rgb values that represent the ball. As expected, twelve gaussian, gave us the best result: In Figure 4, the top region grows and joins through the middle. Even with a large number of gaussians, the brightest white spots of the ball were difficult to capture. Raising the number of gaussians further would make the model prone to noise as the white shine is similar to whites found in other portions of the image.

- **Explain why GMM is better than single gaussian**

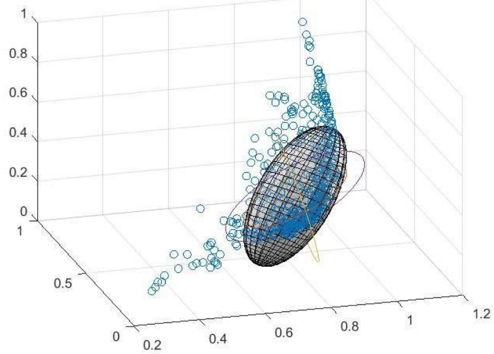


Figure 5: Single-Gaussian

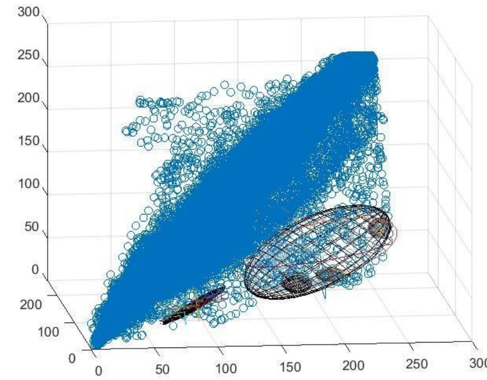


Figure 6: GMM Clusters

If we look at the color space of the images below, we can see that there is a curved cluster of pixels below the diagonal (non - linear). By running Single-Gaussian, it is difficult to obtain accurate clustering, as it captures non-orange pixels (As seen by the overlap of the ellipsoid with the main diagonal of pixels in Figure 5). This happens as lighting and noise leads to high variance. However, in GMM (Figure 6), we are able to accurately fit the model to the shades of orange in the color space. Each ellipsoid model represents a region of a shade of orange with certain lighting conditions and noise. By fitting multiple models, we are able to cluster the pixels more accurately and precisely, as we obtain multiple clusters with low variance.

- **Present your distance estimate and cluster segmentation results for each test image**

Figure 7

Figure 8

Figure 9

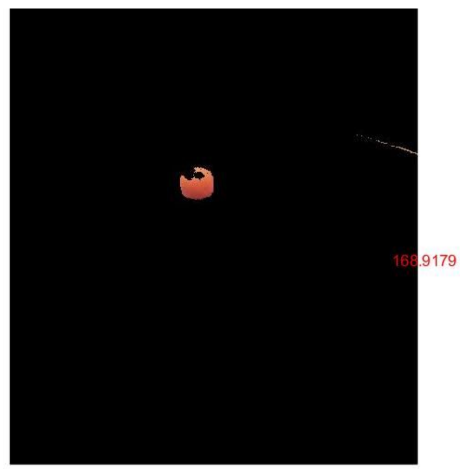
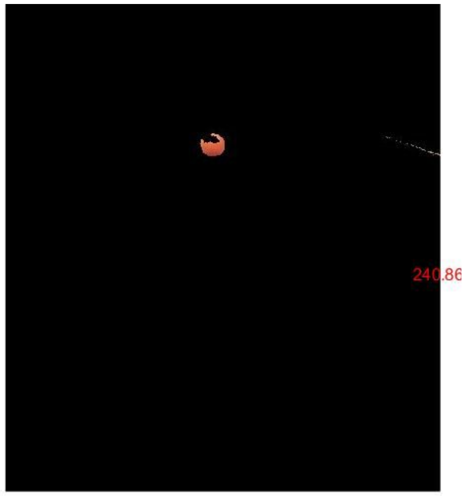


Figure 10



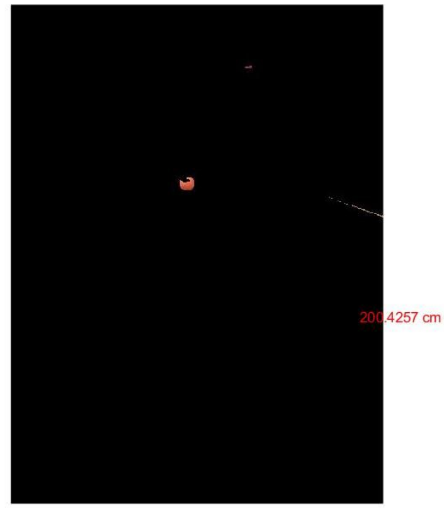
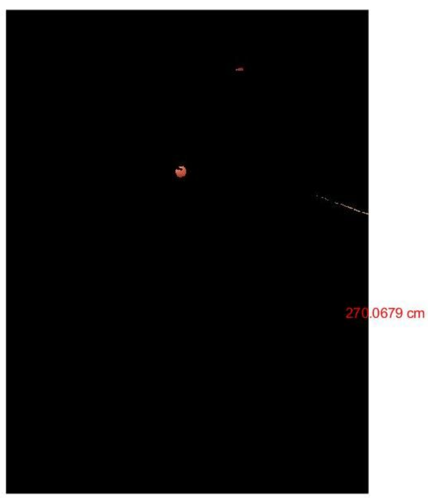


Figure 11

Figure 12



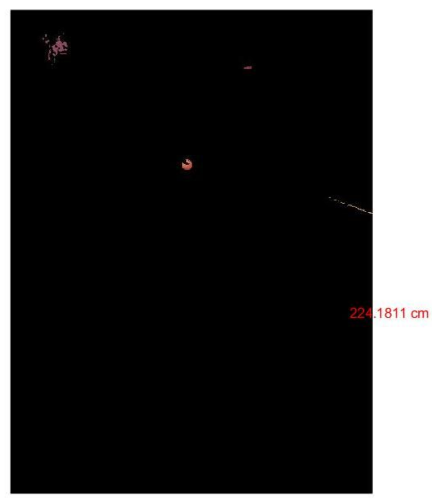


Figure 13

Figure 14

- **Explain strengths and limitations of your algorithm. Also, explain why the algorithm failed on some test images**

Our algorithm does a great job at detecting neutral to dark orange. It doesn’t train the model for orange perfectly, as we see that there is still a decent amount of noise in the background, and certain “non-orange-ball” pixels are also captured (such as the dummy object in some of the test images). The images that our algorithm performed well on are the images without the dummy object. It captures the majority of the orange ball, and having a high number of clusters reduces the noise drastically. One limitation of our algorithm is that upon continuously training our model to improve orange clustering, our cluster weights reach zero as the values slowly get more accurate, rendering some clusters useless. Another limitation would be inaccurate distance estimations due to non optimal test results of orange ball detection.