CS 6643 - Computer Vision, Spring 2021 **James Fishbaugh**

Homework 2

Out: 4 / 09 / 2021
Due: 4 / 21 / 2021 (deadline: 11:55PM)

Late submissions: Late submissions result in 10% deduction for each day. The assignment will no longer be accepted 3 days after the deadline.

Instructions: Submit a single pdf with all code included as an appendix.

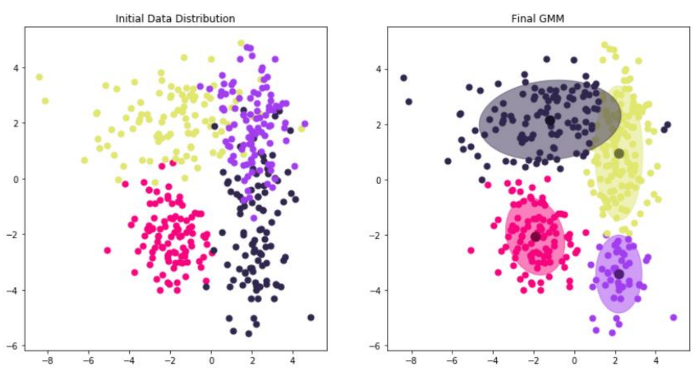
Office hours:

		Mon	Tue	Wed	Thur	Fri
James Fishbaugh	jf146@nyu.edu	4-5 PM		4-5 PM		
Michael Lally	mfl340@nyu.edu		12-1 PM		10-11 AM	
Akshat Khare	ak7674@nyu.edu	2-3 PM				10-11 AM

---

1) Gaussian Mixture Modeling

The company you work for has a client who insists that all delivered code is extensively documented. Your boss has an implementation of Gaussian mixture modeling that she wants to integrate into the final product, but it does not contain a single comment describing the functionality of the code. It is well known around the office that you are knowledgeable about the theory of Gaussian mixture modelling, so you have been given the job to thoroughly document the uncommented code.



- Starting from the attached code for Gaussian mixture modeling, comment and document the code to bring it up the high standards of the client. Be thorough, and reference relevant equations and the EM algorithm. Include the commented code in the appendix.
- The client also wants the option to have a hard clustering/labeling. In your report, describe a method (of your choice) to use the output of the Gaussian mixture model to assign hard class labels to every data point. Implement your method and **show and discuss** the results on random data generated by the code (by showing a plot of data points colored by class/cluster like the figure above).

---

2) Deformable Contour Segmentation

Implement deformable contour segmentation based on the following algorithm. A couple of hints:

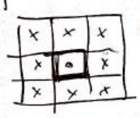
- While working on your implementation, you can test with the included '*ellipse.png*' which represents an ideal image with strong edges and no noise.
- You can immediately update the snake point Pi you are working on, there is no need to keep a copy and update all at once.
- Don't forget to Gaussian blur before computing the gradients for generating the gradient magnitude Eimage. The amount of blurring may have an impact on your results.
- Think about how you want to handle the relationship between P_1 and P_N. In other words, what do P_{i-1} and P_{i+1} mean in these cases?

Implementation:

Model contour $V = \{P_1, P_2, \ldots, P_N\}$.

Greedy approach: every iteration P_i

↳ Compute energy for every point in a neighborhood around p_i



For all 9 points

Move p_i to location which minimizes energy.

↳ Stop when no points move, or stop after a certain # of iterations

How to compute energy terms?

$$E_{cont} = (d - \|p_i - p_{i-1}\|)^2$$

d is the average distace between contour points

$$E_{curve} = \|p_{i-1} - 2p_i + p_{i+1}\|^2$$

Approx. of 2nd derivative

$$E_{image \; external} = -\|\nabla I\|$$

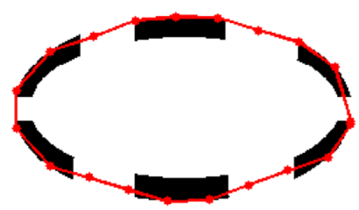
You already know how to compute this

You can initialize the snake in any way of your choosing, with mouse clicks for example if you prefer. You can also initialize with a sampled circle:

```
def create_snake(center_x, center_y, radius, num_pts):

    samples = np.linspace(0, 2*math.pi, num_pts)
    snake = np.zeros((num_pts,2))

    snake[:,0] = np.round(radius * np.cos(samples) + center_x)
    snake[:,1] = np.round(radius * np.sin(samples) + center_y)

    return snake
```

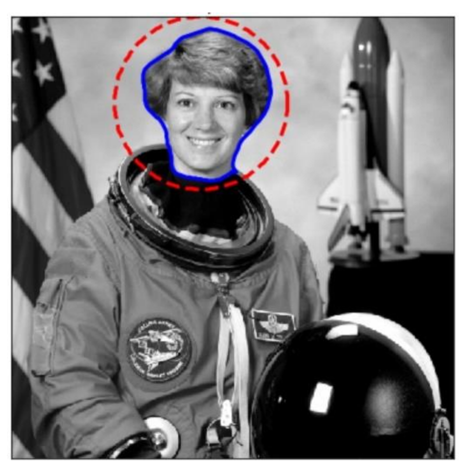
In your report, address the following:

- Using the attached image '*implied_contour.png*', find parameter settings which succeed to segment the implied contour, and also find parameter settings which fail.
    - Include an analysis of the parameter settings. Why did they lead to failure or success?
    - Include images of the final segmentations shown with the original image, like the figure on the right.



- Using the attached image '*rectangle.png*', present your best results using:
    - An initialization with 10 snake points
    - An initialization with 30 snake points
    - An initialization with 500 snake points
    - Show all results and discuss what you observe.

- Using the attached image '*astronaut.png*', show your best result in segmenting the head and hair (as in the figure on the right). Discuss your process for finding the parameters which gave your best result.



3) Parallel Stereo Cameras

Consider a parallel camera setup with identical cameras separated by 300 mm. Each camera has a focal length measured in pixels of 4300 pixels. This roughly corresponds to a camera with a width of 4032 pixels and a field of view of ~50 degrees, corresponding to a focal length of ~50 mm. **You will use focal length in pixels for this exercise.**

- Plot disparity (x-axis) vs depth Z (y-axis) with disparity in the range [0, 200] pixels and Z in the range [0.001, 500000] mm. Use a plotting program rather than a hand drawn sketch.
- Generate two additional plots, with disparity in the range [0, 50] and disparity in the range [0, 20], but keeping Z fixed in the range [0.001, 500000] mm (also limit the axes to these values).
- In all 3 plots, label the x and y axis and include units.
- Describe the shape of the plot. With the shape of the plot and the disparity equation in mind, describe the relationship between disparity and depth Z.