Computer Vision HW1 Report

Yuan Zi 1881448

a. What are the parameters that influence your algorithm? Explain their effect?

K: The number of clusters

Larger K causes a longer time to segmentation but can usually generate more fine and smooth output images. On the contrary, the small K can make code run fast, but generate a rough result.

b. Does your segmentation code always generate the same segments for a given k? Explain?

Not exactly the same, my code will initialize the centroids randomly with variable random seeds, if I set the constant random seeds, sometime it will generate the same centroid and make one cluster empty, then when calculate the inside cluster average distance among the empty cluster will generate divide 0 error. So I choose to randomly generate centroids and monitor if there is an empty cluster the code will generate a new set of centroids, and my stop criterion is based on the loss function difference between two adjacent iterations. If the difference is lower than 1 means the algorithm is convergence which you can tell from the loss descent figures from the output folder. The multiplicity of the solution makes the result slightly different, but are in substantial agreement.

c. What is the objective of this implementation? Which set of features performs better (color vs greyscale)? Explain?

My object function is the

Based on the results pictures, we can easily conclude that the color set performs better because of the more information contained in the color images.

d. Is there any relationship between the choice of k and the time it takes to segment an image? If so, explain.

Yes, based on the human eyes I can tell how many colors contain in the images, for example, the circles, have a black background, red, blue and green circles, totally 4 different colors, so the k = 4 have the best performance. If I choose less than 4 would not have that good result, and if I choose larger k will waste time.

e. Is there a relationship between compute time and the number of image attributes (1d for grey scale vs. 3d for color)? If so, explain.

Yes, the color image need more time to computation, cause in the grey scale the code only need calculate the intensity difference as the feature to assign pixels to clusters. But for color images the code need to calculate difference in the R G B three channels and assign the pixels. Moreover the centroid’s size of grey scale is just the number of clusters, but the size of color images’ centroid’s size is 3 times the number of clusters. We have to renew all these values when iteration.