Theodore Margoles

Hw 5

4/7/2019

1. Methods for computing clusters ( Part A) :
   1. K Means clustering: implemented k means algorithm with feature vector coming from external get feature vector. Start with k random centers, find all point in sphere from each center of radius r, compute the new center of the cluster, and repeat.
   2. HAC clustering. For each cluster we join to that cluster the most similar other cluster in the group. Using a few helper function such as Merge Clusters, Merge2, etc, I have implemented HAC, This algorithm is very slow. Can be modified to return the HAC tree by just linking up each parent with its successive child node in the tree.
   3. Normalized features: As mentioned implemented a method for normalization following the formula in the writeup.
   4. Non-normalized features. Non normalized feature vectors are [X, Y, I]. I for intensity.

2. Visualizations: I had difficulty getting enough RAM free to run my algorithms for HAC or K Means,

3 . Question (part C / E) s:

1. Feature normalization appears to help the quality of segmentation some although not the end all be all in my case. HAC clustering works better than k Means for low values of k. If number of clusters is too small and / or radius is too small the clusters come back as empty / unhelpful clusters. Using features with X, Y, and Intensity had better results than just using Intensity.
2. Feature normalization is a small increase in time to compute. Then HAC is much slower than K Means however they are both rather slow to compute. HAC is really slow due to the initial number of groups. HAC also slower due to need to save whole tree of results.
3. The bigger the image the more data we have to store especially for HAC. if we have an NxN image we have to store N^2 feature vectors one for each point. This can use up too much memory depending on implementation. Another effect of image properties is it effects where the ground truth clusters actually are, and how disparate pixel intensities are and what the ranges of pixel intensities are.