Nicholas Kamper and Eric Henderson

CSSE463 – Image Recognition

Lab 6

# Procedure

For this lab, we implemented the K-Means clustering algorithm using RGB color distance in MATLAB. Once we had a working K-Means implementation, we then ran it over our larger test image for values k=2, k=3, k=4, and k=5.

# K-Means Algorithm

The K-Means algorithm is a method for partitioning data into clusters. The number of clusters must be known prior to executing the algorithm. In addition, there must be some way to represent the data being clustered in such a way that we can use numerical values to determine the cluster location and the “distance” of a point away from the center of the cluster.

For our implementation, we used the 3D Euclidian distance between the values in the R, G, and B color bands to find clusters that were similar in color.

# Results

To test our K-Means implementation, we ran it on this image that we found randomly from Google:



We ran K-Means for k=2,3,4 and 5, and those results are included in order:









# Conclusion

For the K-Means images we obtained, we found that most of the clusters had an average color of gray. This is probably due to the large number of different colors in each cluster resulting in a gray average. One way that our results could’ve been improved would have been to use images with fewer colors.

# Future Ideas

One way that we could improve our algorithm would be to use a different measure of clustering than RGB color bands – perhaps LST or HSV would work better. Finally, we could use physical proximity and edges in addition to color to cluster areas together.