Vocabulary Tree Assignment

Computer Vision

Aman Bhatia

05, October, 2016

Database Description

The database consists of 10,000 images of Delhi. They are all downloaded from the internet using a script. No manual filtering is done on the database so you can also find images of TajMahal and other famous monuments.

Vocabulary Tree

For building Vocabulary Tree, I used the very popular library built by Snavely(https://github.com/snavely/VocabTree2). I tried several combinations of branching factor(3,5,7) and depth(9,10) of the tree and found that the output for all of them is almost the same. This is because the images in the data are quite clustered. If the query image is of say Qutub Minar, then all the 30-40 Qutub Minar images in the dataset were returned as output. Finally I used the branching factor to be 3 and depth to be 10.

Match Graph and Clustering

Every image in the database is a vertex in the graph and the edges and edge weights between two images represent the similarity between them. More the edge weight, more similar the images are. For building the match graph, I quried every image in the database and extracted 100 closest images to the query image in a ranked order. I connected the query image with those 100 images and set their edge weights to be same as the similarity weight returned by the vocabulary tree.

For clustering the graph, I used Markov Clustering(MCL) (http://micans.org/mcl/). This also includes experimentation with the clustering parameters. We can control the coarsity of the clustering. Default settings made ~350 clusters. Clusters were good but there was redundancy i.e. say, for a set images of qutub minar, which were taken from two different views, clustering produces two different clusters. With different parameters, I increased the coarsity and number of clusters came down to ~150. The clustering was very good i.e. it clustered all kind of images of a place, say qutub minar in one single cluster.

--END OF DOCUMENT--