BovW based classification

**General approach**

We aim to classify images based on features extracted from training data.

Features used?

* we split each image into 16 7x7 sized blocks, the list formed by flattening out each block is then used for k means clustering

How was **k** decided?

* main aim was to achieve higher accuracy
* when run with k ~ 50, accuracy was around 50%
* when run with k > 80, accuracy was 60-70%
* keeping k > 120 was taking a lot of computation time, hence it was discarded
* finally, **k = 100** was taken for running k-means clustering

How was classification performed?

* First, the histograms for all training images were created using the output of k-means algorithm
* Then, for each test image, its histogram was compared with those of training images
* The label of the training image whose histogram was closest to that of test image was assigned to it

**How to run code? (Windows)**

|  |
| --- |
| $ *# create virtual environment* $ python -m venv env  $ *# activate virtual environment* $ env\Scripts\activate  $ *# install requirements* $ pip install -r requirements.txt  $ *# run code* $ python RunAll\_2019CSB1095.py  $ *# run code on smaller dataset add --demo flag* $ python RunAll\_2019CSB1095.py --demo |

**Output**



results.txt file is attached

* It contains output for all test set images
* Columns - **test\_image\_id** | **corresponding\_train\_image\_id** |  **actual\_label** |  **predicted\_label**

kmeans.txt contains output centroids of k means clustering used for testing

Images representing centroids are present in *visual\_words* directory