Computer Vision

Homework 5

PART A

Color histogram and k-nearest neighbor (kNN) classifier

Describe your quantization/binning method and parameters.

Each image in the dataset is of data type *256x256x3uint8,* with the 3rd dimension containing colour information.

Since each element of of type uint8, each pixel has R, G and B values going from 0 to 255.

I built histograms of individual R, G and B channels with a bin size of 256, to represent all the values. The bin size of 256 helps represent all the possible 256 values each of the R, G and B components can take, to get the highest level of detail in the histogram.

I then placed all the histograms side by side to obtain a *1x768* vector for each image to be used for kNN algorithm.

Report number of K for the kNN classifier.

The ‘k’ value in the kNN classifier should practically be an odd number, to avoid ties in the result.

I tried four different values of ‘k’ for this part. Below is the table of the used ‘k’ values and their categorization accuracy:

‘k’ Value	Accuracy(%)
1	45
3	45
9	45
15	45

Display the confusion matrix and categorization accuracy.

```
confuseMat =

    26     2     3     3     8     6     2     9
    15    83     1    23    11    27    11    14
     5     1    64     1     3     4     1     3
    15     5     8    31    15    13    19     7
    12     3     6     8    37     7     5    21
    12     3     8     5     8    32     3     5
     2     0     5    15     5     5    52     6
    13     3     5    14    13     6     7    35
```

```
categorization_accuracy =
45%
```

PART B

Bag of visual words model and nearest neighbor classifier

Describe the number of visual words you use, K-means stopping criterion, and the categorization accuracy.

Number of Visual Words Used:

- The features of each image are encoded into a *128x1* feature descriptor vector(SIFT algorithm). All of these descriptors can be thought of as visual words.
- Each image in the training set has about 200 to 300 features.
- However, since many feature descriptors are similar, we perform K-means clustering on all of them to obtain fewer descriptors to speed up computation.
- The means of the clusters formed is the new group of visual words which can be used to classify scenes.
- The number of clusters obtained after K-means clustering is the number of visual words used to represent each image, which is then used to form histograms.

Describe the number of visual words you use, K-means stopping criterion, and the categorization accuracy.

K-means stopping criterion:

- In K-means clustering, labels are assigned to each sample based on the closest mean to the sample.
- The algorithm stops when the labels assigned to each sample do not change between two consecutive iterations, which means that all the samples have been assigned labels to the mean they are closest to. When the labels of samples do not change between iterations, the algorithm converges and it is ended.

Describe the number of visual words you use, K-means stopping criterion, and the categorization accuracy.

Categorization Accuracy:

- In the given dataset, there are 8 categories of scenes.
- Categorization accuracy is the number of correct predictions divided by the total number of predictions.
- To obtain the categorization accuracy, the predicted labels are compared to the test labels(truth). If the two labels match, it is deemed as a correct prediction.
- It is a measure of how many correct predictions an algorithm can make from the total number of predictions.

Display the confusion matrix and categorization accuracy.

NOTE: For K Value = 200

```
confuseMat =

    56     1    19     5     6    14     2     7
     0    77     1     9    10     7     5     5
    17     1    49     1     3    11     3     5
     3     2     3    29     1     2     8    16
    10    10     9     8    52    17    21    13
     8     4     8    10    14    34    10    11
     5     2     5    20    10    12    38    12
     1     3     6    18     4     3    13    31
```

categorization_accuracy =
45.75%

PART C

Bag of visual words model and a discriminative classifier

Report the training time and testing time for SVM

Average Training Time = 0.2036 seconds(average of 8 models).

Model 1 to Model 8:

0.31, 0.16, 0.13, 0.16, 0.22, 0.24, 0.16, 0.22 (seconds).

Average Testing Time = 0.0391 seconds(average of 8 models).

Model 1 to Model 8:

0.06, 0.02, 0.04, 0.03, 0.04, 0.03, 0.03, 0.03 (seconds).

Display the confusion matrix and categorization accuracy.

NOTE: For K Value = 600

```
confuseMat =

    67     0    23     1     2    24     1     5
     0    90     1     5     4     7     7     0
     7     0    48     1     3     4     4     4
     0     0     4    75     0     0    16    17
     9     6     7     2    74    18    10     7
    12     2    12     3     9    43     3     2
     4     1     4     1     4     1    46     1
     1     1     1    12     4     3    13    64
```

```
categorization_accuracy =
63.3750%
```

PART D

CNN as features and a discriminative classifier

Describe the model you used.

The CNN model being used is the Fast CNN-F architecture model designed by Chatfield et al in the paper: *`Return of the Devil in the Details: Delving Deep into Convolutional Networks'* . The architecture is similar to the one used by Krizhevsky et al. It comprises of 8 learnable layers, 5 of which are convolutional, and the last 3 are fully-connected. The input image size is 224×224. Fast processing is ensured by the 4 pixel stride in the first convolutional layer. The main differences between this architecture and that of Krizhevsky et al, are the reduced number of convolutional layers and the dense connectivity between convolutional layers. *(Source: `Return of the Devil in the Details: Delving Deep into Convolutional Networks', Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, BMVC 2014)*

Display the confusion matrix and categorization accuracy.

```
confuseMat =

    94     0     1     0     1     5     0     0
     1    91     1     0     0     3     0     1
     1     0    92     1     1     1     1     0
     0     0     0    92     0     0     1     3
     0     4     1     0    94     2     1     1
     4     5     1     0     4    89     0     0
     0     0     3     4     0     0    97     0
     0     0     1     3     0     0     0    95
```

```
categorization_accuracy =
93%
```

Graduate Credit

1. For bag of visual word models, experiment with different number of visual word, e.g. K = 25, 50, 100, 200, 400, 800, 1600. Report the categorization accuracy for each K.

For bag of visual word models, I measured the categorization accuracy for different values of K and used SVM models to classify scenes. The following table shows the accuracy obtained for all the values of K:

K Value	Accuracy (%)
25	25.2
50	35.5
100	45.8
400	54.2
600(Original)	63.3

2. Try using two different pre-trained CNN models. Report the accuracy of each of the models.

Two different models used for feature extraction were:

1. AlexNet
2. VGG-VD

    Their accuracy is given in the table below:

CNN Model	Accuracy (%)
AlexNet	74.3
VGG-VD	87.6

3. For one specific CNN model (e.g., AlexNet or VGGNet), report the classification accuracy when you use different levels of feature activations, e.g., Pool4, Pool5, Fc6, Fc7.

For the model VGGNet, the classification accuracy using different levels of feature activations is given below:

K Value	Accuracy (%)
Pool 2	92
Pool 5	93
FC 6	95
FC 7	95