I. Bag-of-Words Classification Pipeline Overview

Q1) When computing the nearest neighbor for a single test image without using any approximation techniques, such as a K-D tree, the space and time complexity can be determined as follows:

Space Complexity:

The space complexity is primarily determined by the storage of the training set, which consists of pairs of BoW histograms and their corresponding class labels. Each BoW histogram is typically a vector of fixed length.

The space required for the training set is proportional to the number of training images and the size of each BoW histogram.

Space Complexity ≈ $O(N * M)$

Where:

N is the number of training images in the dataset.

M is the size of the BoW histogram (the dimensionality of the feature vectors).

Time Complexity:

To predict the label of a single test image, you need to compare the BoW histogram of the test image with all BoW histograms in the training set using a distance metric.

For each test image, you compute the distance between the test image's BoW histogram and all BoW histograms in the training set.

The time complexity of computing the distance between two BoW histograms is typically $O(M)$, where M is the dimensionality of the histograms.

You need to do this comparison for each training image.

Time Complexity ≈ $O(N * M)$

Where:

N is the number of training images in the dataset.

M is the size of the BoW histogram (the dimensionality of the feature vectors).

In summary, without using any approximation techniques like K-D trees, the time complexity of predicting the label of a single test image with respect to the size of the training set is linear with respect to the number of training images and the dimensionality of the BoW histograms. The space complexity is also linear with respect to these parameters. This means that as the training set grows in size, both time and space requirements will increase linearly, making it computationally expensive for large datasets.

[Constructing a histogram based on the BoW model. | Download Scientific Diagram (researchgate.net)]

II. Feature Description with Histograms of Oriented Gradients (HOG)

The need to run the model evaluation over several iterations (10 in this case) stems from several reasons:

1. Variability in test data: Model performance can vary from run to run due to variability in test data. By running the model on multiple test sets, you get a better estimate of its average performance.

2. Sampling bias: If the training and test data are not representative of the true distribution of the data, you may get a biased estimate of performance. Multiple runs help to mitigate this bias.

3. Robustness assessment: Repeatedly running the model allows you to assess its robustness to different partitions of the training and test data. This can reveal how the model performs against different data and whether it is consistent in its performance.

4. Reliability of results: By averaging performance over several runs, you get a more reliable estimate of the model's performance. This reduces the effects of chance and one-off variations.

The accuracy for ten classes is expected to be lower than the accuracy for 2 classes because it is harder for the model to classify images in 10 classes than 2. As we can see on the results, just below

For 10 classes :

```
Mean Accuracy: 0.19
Random Classifier Mean Accuracy: 0.09
```

For 2 classes : one iteration :

```
Mean Accuracy: 0.52
Random Classifier Mean Accuracy: 0.23
```

With the L1, results are slightly worse than with the L2.