Computer Vision Coursework 3
Scene Recognition

Lucas Noyau - ln3g14 - 26921936
Jean-Luc Mourey - jm32g15 - 28231589
December 14, 2017

1 Introduction

2 Classifiers

All classifiers used in this project extend from the abstract class `MyClassifier`. It was created in order to limit the amount of repeated code in the other classifiers, as well as give a framework for us to work from. The content of the class is as follows:

- Class variables are `trainingData` and `testingData`. These hold the datasets that are used for training and testing respectively.
- Constructors, one with no parameters, the other uses a String of the path to get the datasets from.
- `go()` method is used to train and test the classifier on the class variable datasets, with the result being saved by `printResults(ArrayList)`.
- `printResults(ArrayList)` takes a list of the predicted classes in String form, and both prints it to `System.out` and to a file `output.txt`.
- `classify(groupedDataset)` takes a dataset and returns the ArrayList of String that is used by `printResults`. It does this by iterating over all the images in the dataset and calling `classify(FImage)` on each of them.
- `train(GroupedDataset)` takes a dataset but doesn't return anything. Each classifier has a different method for training, therefore this method is abstract.
- `classifyFImage` is another abstract method, as image classification depends on the classifier. This method return a String of the predicted class name.

Therefore the only variations in the classifiers listed below are in the `train(GroupedDataset)` and `classify(FImage)` methods, although other methods are used in order to remove duplicate code and make the code easier to read.

2.1 Run 1: A Simple k-nearest-neighbour Classifier

2.2 Run 2: A Set of Linear Classifiers

2.3 Run 3: Developing The Best Possible Classifier

3 Individual Contributions

A Code