HW 6-7 Report

Shanmuga Perumal Harikumar

Problem Statement:

The aim of this assignment was to create a neural network for hand gesture recognition using WPI's hand gesture database.

Approach:

The process of classifying input images into different classes has become more efficient with the advent of Deep neural networks that use convolutional layers. There have been some really good network architectures that have demonstrated very high accuracies in the task of classification.

For the purpose of classifying the gesture dataset, I wanted to use VGG 19 network architecture. The initial weights were available online which I used. The pretrained network saves time and has been proven efficient in other applications.

The first 18 layers were left untrained and the available weights were used. Only the final fully connected layer that outputs the class was trained. The final layer was modified from being able to output 1000 classes to one that outputs class classification.Keras has functions that let certain layers to be left untrained. The layer object has an attribute called '**trainable**' which defaults to true. This parameter when set to false will make the layer skip training.

For our purposes all layers were set to be untrainable and once the weights were loaded the final fully connected layer and dropout layer were popped and replaced with layers that output 5 classes. '**Model.layer,pop**' method available in keras lead to errors because the number of outputs from the previous layers did not match the number of inputs in the next layers. This required a function called '**pop_layer(model)**' to be written. This function takes care of changing the number of inbound and outbound nodes.

In order to parse the data '**ImageDataGenerator()**' was used. A separate instance of the object was created for training and test data. The **'flow_from_directory()'** method was used to generate augmented images from a file located on the system. The '**ImageDataGenerator()**' object lets the user perform lots of preprocessing on the images. For our purposes no real preprocessing (whitening,rotating,skewing) of the images was needed. A batch size of 64 and a sample size of 5000 per epoch was chosen. This led to warnings because the number of samples was not divisible by the batch size.So the batch size was changed to 50.

Stochastic Gradient Descent was used initially as optimizer with a learning rate of 0.1.the number of epochs were chosen to be 25,which gave a maximum accuracy of **93.21%** on the validation dataset.

When adam optimizer was used the best accuracy of **94.56%** was achieved. The average training time of a epoch was around 200 seconds.The accuracy could definitely be improved by playing around more with the hyperparameters.

The network and weights performed really well with real time classification of gestures using the web camera on a computer. It was able to correctly classify 14 out of 15 trials.

Result:

The VGG 19 model was modified to classify gestures. The WPI gesture database was used to train and validate the network.An accuracy of 94.56 % was achieved with adam optimizer,batch size 50 ,samples per epoch 5000 and 25 epochs. The network was tested with webcam images in real time.weights can be found in the embedded drive link.