V. Lab Five - 31298516

In this lab, we will explore the performances of the convolutional neural network(CNN) by increasing the complexity of our CNN model gradually. The convolutional layer is an essential component in the deep neural network(DNN), which is used for extracting features from highly abstracted data, and it can also achieve multichannel fusion.

In general, we extract features by CNN layers first, and then fit our model to the goal task by adding fully connected layers behind them. Besides, we can also introduce pooling layers into our model for further optimisation on our model. Our goal task is to recognise two parameters of a linear function (y = ax+b) from a series of generated graphs.

A. *A simple CNN Baseline*

In this work, we only use a straightforward model with a conv layer and an fc layer after it. And the numerical results are: ['test_loss': 2.1975858211517334, 'test_acc': 0.8980000019073486]. The loss curve of it is shown below, altogther with loss curves from other models.

B. *A simple CNN with Global Pooling*

In this work, we add an extra conv layer after the first conv layer, and also a global max-pooling layer is added. Consequently, the number of nodes in the fc layer decreases. We know, the overfitting is apt to happen in the fc layer and it can be regarded as a black box in our model. Though we can use dropout, it will be better to solve this problem from its root. The global pooling layer aims to partially replace the fc layer in our model by decomposing our data directly by calculating the max or average pixels. And we can tell it also works in improving the generalisation ability of our model and then accelerating the convergence speed. The numerical results are: ['test_loss': 2.063516139984131, 'test_acc': 0.9419999718666077].

C. *Let's Regress*

In this work, we achieved further improvement based on the previous model by superposing a pair of interleaved layers above each of our original datum which can be regarded as a kind of data augmentation in order to avoid model overfitting and increase its generalisation ability. The numercal results are: ['test_loss': 0.3618202805519104, 'test_acc': 0.9900000095367432]. By using convolution and pooling technologies, the average accuracy and generalisation ability of our 'shallow' neural network model has been brought into a new high level. The loss curves of all three models above and the visualisation of our data augmentation approach are shown below.

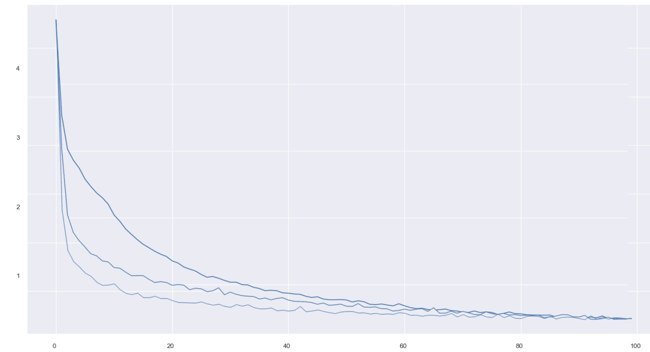

Figure 11: Loss curves, from top to bottom are namely the first model, second model and third model.

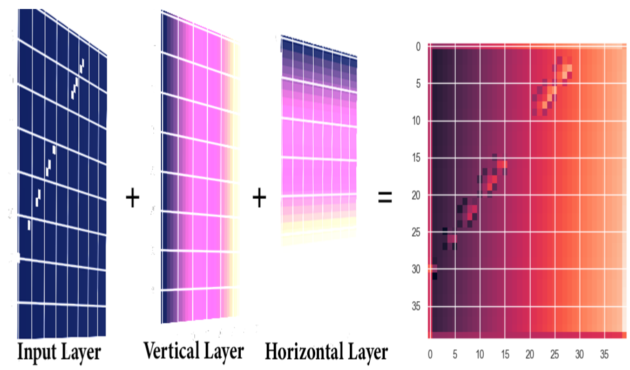

Figure 12: Visualisation the multi-channel fusion process.