**Gesture Recognition Project Write Up**

**Experiment -1:**

Model – 1:



|  |  |
| --- | --- |
| Frame Sequence | 0,2,4,6,8,10,12,14,16,18,20,22,24,26,28 |
| Optimizer | adam |
| Number of Epochs | 20 |

Observations

* I have got resource exhausted error for batch size 600, 300, 150, 100 and 50. So I have used batch size of 30 for the training
* Training accuracy – 0.8597
* Validation accuracy - .5900
* While training I got multiple callback error and there were no consistency in the error pattern, sometime I got the error after 3 epochs and some time I got the error after 5 epochs. In order to save GPU credits I have commented the callbacks argument of the fit method, this has resolved the error.
* I used 20 epochs, the model was clearly over fitting
* I have used every alternate frame i.e. img\_idx = [0,2,4,6,8,10,12,14,16,18,20,22,24,26,28]
* Highest validation accuracy was observed for epoch number 15.

Next Step:

* I instead of every alternate frame, I will use skip every fourth frame and rest of the parameter will remain same.

**Experiment -2 :**

* Model-2:
  + It has the exactly same architecture as that of Model-1, the difference is in the training data, in order to train this model we will use 3 consecutive frames and skip the fourth one i.e. . img\_idx = [0,1,2,4,5,6,8,9,10,12,13,14,16,17,18,20,21,22,24,25,26,28,29].
  + Also the batch size used is 50. I have tried 100 again however received resource exhausted error.
* Observations:
  + Training accuracy after epoch 20 – 0.8145
  + Validation accuracy after epoch 20 – 0.6000
  + Highest validation accuracy was observed for epoch 16 and the validation accuracy was 0.7100
  + Model is still over fitting
* Next Step:
  + Reduce the complexity in the model by increasing the dropout in the convolution layer to 0.25 and to 0.4 in the dense layers as well

**Experiment -3:**

* Model-3:
  + The model architecture is the same as above, however here we have increased the dropout rate from 0.10 to .25 in the convolution layer. Also the drop out in the dense layer has been increased to 0.4
  + 

|  |  |
| --- | --- |
| Frame Sequence | 0,1,2,4,5,6,8,9,10,12,13,14,16,17,18,20,21,22,24,25,26,28,29 |
| Optimizer | adam |
| Number of Epochs | 20 |

* Observations:
  + Training accuracy is .3846 and validation accuracy is .4000
  + We were able to overcome the issue of over fitting however we have now under fitting.
* Next Step:
  + To overcome the under fitting we will add one more layer of Conv3D and one Dense layer and try to make the model more complex.

**Experiment - 4:**

* Model-4
  + 

|  |  |
| --- | --- |
| Frame Sequence | 0,1,2,4,5,6,8,9,10,12,13,14,16,17,18,20,21,22,24,25,26,28,29 |
| Optimizer | adam |
| Number of Epochs | 20 |

* Observations:
  + Training accuracy is 0.6893 and validation accuracy is .0.6600
  + Despite adding more complexity the model is not over fitting that is a good thing however the overall accuracy is still low which is a clear sign of under fitting.
* Next Step:
  + To overcome the under fitting we will try the following :
    - Increase the neurons in Dense layer
    - Change the optimizer to rmsprop
    - Increase the epochs to 30

**Experiment – 5:**

* Model – 5:
  + 

|  |  |
| --- | --- |
| Frame Sequence | 0,1,2,4,5,6,8,9,10,12,13,14,16,17,18,20,21,22,24,25,26,28,29 |
| Optimizer | rmsprop |
| Number of Epochs | 30 |

* Observations:
  + Training accuracy is 0.5581 and validation accuracy is .0.5500
  + Under fitting has worsen in comparison to pervious model.
* Next Step:
  + To overcome the under fitting we will try the following :
    - Change the optimizer back to adam

**Experiment -6:**

* Model-6:
  + Architecture of model 6 is same as that of model 5, the only difference in model 5 and model 6 is that of the optimizer. We have used “RMSPROP” in model 5 and in model 6 we have used “ADAM”
* Observations:
  + Training accuracy is 0.8100 and validation accuracy is .0.7500
  + So far this is the best model that we have obtained
* Next Step:
  + In order to achieve even higher train and validation accuracy we will try to make it more complex and add more epochs
    - We will increase the epochs to 40
    - We will add one more dense layer of 128 neurons

**Experiment -7:**

* Model-7:
  + 

|  |  |
| --- | --- |
| Frame Sequence | 0,1,2,4,5,6,8,9,10,12,13,14,16,17,18,20,21,22,24,25,26,28,29 |
| Optimizer | adam |
| Number of Epochs | 40 |

* Observations:
  + Training accuracy is 0.8200 and validation accuracy is .0.5300
  + Over fitting has increased a lot due to added layer
* Next Step:
  + We have received the best result so far with model-6, hence next will use the same architecture as model 6 however while training we will use 50 epochs instead of 30

**Experiment -8:**

* Model-8:
  + Architecture of model 8 is same as that of model 6, the only difference in model 6 and model 8 is that of the number of Epochs. We have used 30 epochs in model 6 and in model 6 we have used 50 Epochs.
* Observations:
  + Training accuracy is 0.8900 and validation accuracy is .0.5100, hence the model is over fitting
* Next Step:
  + So far model-6 is the best we have got.
  + Next we will experiment with CNN+RNN networks.

**Experiment -9:**

* Model-9:
  + We have used CNN+GRU based architecture
  + 
* Observation:
  + Training accuracy – 0.8899
  + Validation accuracy – 0.6700
  + The model is over fitting
* Next Step:
  + Increasing the dropout rate
  + Adding L2 regularization

**Experiment-10:**

* Model-10:
  + CNN + GRU based architecture
  + 
* Observation:
  + Training accuracy – 0.8400
  + Validation accuracy – 0.56
  + The model is still over fitting
* Next Step:
  + Reduce the number of neurons in GRU layer and remove one dense layer

**Experiment- 11:**

* Model-11:
  + 
* Observation:
  + Training accuracy – 0.8400
  + Validation accuracy – 0.42
  + No Improvement in the validation accuracy
* Next Step:
  + Reduce the number of neurons in GRU layer to 100
  + Increase the dropout rate in GRU layer to 0.5

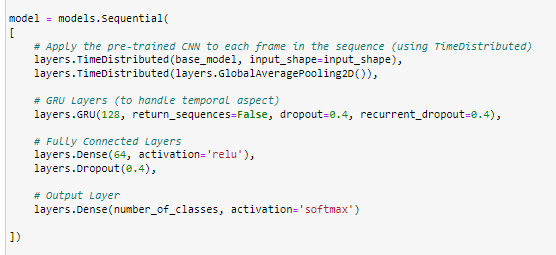
**Experiment- 12:**

* Model-12:
  + 
* Observation:
  + Training accuracy – 0.91
  + Validation accuracy – 0.54
  + No Improvement in the validation accuracy
* Next Step:
  + Reduce the number of neurons in GRU layer to 64

**Experiment- 13:**

* Model-13:
  + 
* Observation:
  + Training accuracy – 0.89
  + Validation accuracy – 0.46
  + No Improvement in the validation accuracy
* Next Step:
  + Use a model with transfer learning

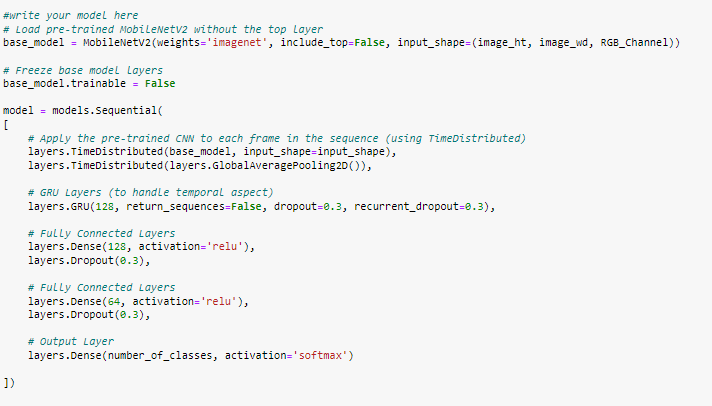
**Experiment- 14:**

* Model-14:
  + Transfer learning based model
  + 
* Observation:
  + Training accuracy – 0.7526
  + Validation accuracy – 0.5500
  + The model is still over fitting
* Next Step:
  + The model is simple, we will add some more complexity to this model

**Experiment- 15:**

* Model-15:
  + 
* Observation:
  + Training accuracy – 0.46
  + Validation accuracy – 0.48
  + Model is not over fitting anymore however now it is under fitting
* Next Step:
  + We will reduce the dropout rate to add more complexity

**Experiment- 16:**

* Model-16:
  + 
* Observation:
  + Training accuracy – 0.74
  + Validation accuracy – 0.54
  + Although the accuracy rate on training data has gone up now however the accuracy on validation is still low

**Conclusion:**

For model 6 we have obtained the best accuracy on training and validation set and hence we will use that for this project.