Gesture Recognition Assignment

1. Group Members (MLC35):

- Asawari Bhosale
- Neethu K

2. Problem Statement

As a data scientist at a home electronics company which manufactures state of the art smart televisions. You want to develop a cool feature in the smart-TV that can recognize five different gestures performed by the user which will help users control the TV without using a remote.

The gestures are continuously monitored by the webcam mounted on the TV. Each gesture corresponds to a specific command:

- Thumbs up: Increase the volume
- Thumbs down: Decrease the volume
- Left swipe: 'Jump' backwards 10 seconds
- Right swipe: 'Jump' forward 10 seconds
- Stop: Pause the movie

Each video is a sequence of 30 frames (or images).

3. Model assessment write-up:

Model No.	Model	No of image	Image size	No of Epoc	Batch size	Train time	Result	Experiment/Descission
0	Conv3D	30	160x160	30	40	NA	ResourceExhausted Error: OOM when allocating tensor with shape[40,16,30,160,160] Error	We need to reduce the number of the layers and images as well size of the images.
1	Conv3D	18	64x64	5	25	137s	train_acuraccy of 41% and val_acurracy of 21%.	Here the model accuracy is very low. it is clearly observed that it underfits. hence we will increase the epoc, image size etc.
2	Conv3D	18	64x64	20	30	502s	train_acuraccy of 33% and val_acurracy of 26%.	Here as we see changing the no of epoc and batch time does not have much change on the accuracy. hence we will change the number of frames and neurons from 64 to

								128. Also we are reducing the cropping to 3 % from 10 % so we dont loose any data.Increase image size to 84 from 64.
3	Conv3D	18	84*84	20	30	483	train_acuraccy of 42% and val_acurracy of 33%.	*Increase the amount of trainable data. Still model is underfitting,so reducing the dense nuerons to 64 from 128.Lets increase number of epoch and no of images as well*
4	Conv3D	30	100x100	30	30	1320	train_acuraccy of 64% and val_acurracy of 48%.	*The model has started overfitting but it has improved accuracy. Also the training time has increased to 1320 sec due to increase in number of input images to read.Try other models as Conv3D not giving desired accuracy for given data-set.*
5	*CNN-LSTM Model*	30	120x120	20	20	1020	train_acuraccy of 87% and val_acurracy of 63%.	*The base hybrid model has achieved the better accuracy than all the Conv models that we have experimented, although the model is overfitting here due to large number of parameters* 1,657,445 *in the model.*
6	*CNN-LSTM Model*	30	160x160	20	20	1380	train_acuraccy of 76% and val_acurracy of 66%.	*The model is overfitting and its very complex model right now with too many parameter. Lets reduce the image size to 120 from 160 and number of neurons in the network for this.Also, the accuracy can be further improved.*
7	*CNN-LSTM Model*	30	120x120	25	20	1043	train_acuraccy of 78% and val_acurracy of	*The model performance has improved marginally. Lets increase the dense neurons to see if*

							75%.	*increasing the network parameters helps.*
8	*CNN-LSTM Model*	30	120x120	25	20	1500	train_acuraccy of 92% and val_acurracy of 79%.	*The models is overfitting. But the network is heavy right now with these many network paramaeters and 1500 sec training time. Lets try experimenting with some other hybrid models. An LSTM has 4 gates, while GRU has 3 gates. Using GRU will significantly reduce the training times as it needs to compute values for 3 gates and its performance is at par with the LSTMs.*
9	*CNN-LSTM with GRU*	30	12x120	20	20	989	train_acuraccy of 98% and val_acurracy of 81%.	*This model has decent val accuracy but it is clearly overfitting here. Lets try other to use pre-trained models for our own problem here using the Transfer Learning.*
10	*MobileNet Transfer Learning Model*	18	120x120	5	20	158	train_acuraccy of 95% and val_acurracy of 73%.	*The training time is less here. The model is again overfitting here with poor validation accuracy performance. We are not training the mobilenet weights and we see validation accuracy is very poor. Let's train them as well and observe if there is performance improvement.*
11	*Transfer Learning with GRU and training weights*	18	120x120	5	20		train_acuraccy of 92% and val_acurracy of 93%.	*This mode is selected as the final model because it has very high accuracy of 93 % for validation data.*

Final Model Selected:

The model is getting saved for all the experiment for each epoch in the form of .h5 file in the disk. The final model chosen is model-00004-0.20399-0.92308-0.22136-0.93000.h5

This model is based on the transfer learning with GRU and re-training the weights. The pre-trained model of MobileNet is used for this purpose with weights of ImageNet.

Now since the ImageNet dataset does not have exact dataset available as per our problem statement, we have not feeded the layers that we are using for our final model and we have re-trained the weights in our transfer learning.