## **Deep Learning Course Project- Gesture Recognition**

* **Mayur Pardeshi**
* **Ambar Ashish**

# Problem Statement

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

* 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.

Following is the summary of experiments and observations for the project.

* We did many more experiments but listing down the main ones along with the explanations.
* In the model column itself, we have tried to give the summary of model.
* Note that dropouts (usually it was set to 0.25 in dense connections) and batch normalization play an important role for regularization but not explicitly mentioned them in the model column.
* We zeroed down to batch size of 10 for all the models as it was not giving any memory issues and models were training well with it.
* We zeroed down to experimenting with either 20 images OR 30 images per video.
* We zeroed down to experimenting with either 120 by 120 image size OR 160 by 160 image size.
* Final models are highlighted in yellow along with their accuracy and loss numbers.
* Data transformations/augmentation didn’t seem to be helping much in improving accuracy or loss so we started avoiding them in the final constructive modeling.

You can draw inspiration from the concepts taught in the Industry demo in CNNs to experiment with the data and different architectures.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D** | **Training Accuracy : 0.8989**  **Validation Accuracy : 0.54** | **Increase the number of epoch**  **Increase the filter size** |
| **2** | **Conv3D** | **Training Accuracy : 0.9050**  **Validation Accuracy : 0.8100** | **Increase the filter size.**  **Increased height and width of image** |
| **3** | **Conv3D** | **Training Accuracy : 0.9442**  **Validation Accuracy : 0.8100** | **reduce the filter size** |
| **4** | **Conv3D** | **Training Accuracy : 0.9502**  **Validation Accuracy : 0.7800** | **Reduce number of epoch, increased the filter size , reduced height and width and used CONV2d+GRU with 128 GRU cells and and 128 dense cells** |
|  |  |  |  |
| **5** | **Transfer learning Conv2d+GRU** | **Training Accuracy : 0.9849**  **Validation Accuracy : 0.9500** | **Random data transformations on training data set (augmanetation)** |
| **6** | **Transfer learning Conv2d+GRU** | **Training Accuracy : 0.9894**  **Validation Accuracy : 0.9700** | Fine Tuned RNN and filter decreases |
| **7** | **Transfer learning Conv2d+GRU** | **Training Accuracy : 0.9910**  **Validation Accuracy : 0.8900** | False fine tuned with 128 LSTM cell and 128 dense cell |
| **8** | **Transfer learning Conv2d+LSTM** | **Training Accuracy : 0.9412**  **Validation Accuracy : 0.9600** | **Reduced number of filter and added 64 dense nodes** |
| **9** | **Transfer learning Conv2d+GRU** | **Training Accuracy : 0.9894**  **Validation**  **Accuracy : 0.9800** |  |
| **Final Model** | **Transfer learning Conv2d+GRU** | **Training Accuracy : 0.9894**  **Validation**  **Accuracy 0.9800.** | **It has took less amount of time and training and validation accuracy are much better** |