**Gesture Recognition**

**By: SOUMAYADEEP MANNA**

**Problem Statement :**

Imagine you are working 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:

**1. Thumbs up: Increase the volume**  
**2. Thumbs down: Decrease the volume**  
**3. Left swipe: 'Jump' backwards 10 seconds**  
**4. Right swipe: 'Jump' forward 10 seconds**  
**5. Stop: Pause the movie**

### **Understanding the Dataset :**

The training data consists of a few hundred videos categorized into one of the five classes. Each video (typically 2-3 seconds long) is divided into a sequence of 30 frames(images). These videos have been recorded by various people performing one of the five gestures in front of a webcam - similar to what the smart TV will use.

The data is in a zip file. The zip file contains a 'train' and a 'val' folder with two CSV files for the two folders. These folders are in turn divided into sub folders where each sub folder represents a video of a particular gesture. Each sub folder, i.e., a video, contains 30 frames (or images). Note that all images in a particular video sub folder have the same dimensions but different videos may have different dimensions.

Specifically, videos have two types of dimensions - either 360x360 or 120x160 (depending on the webcam used to record the videos). Hence, you will need to do some pre-processing to standardize the videos.

### **Goals of this Project :**

In this project, you will build a model to recognize 5 hand gestures. The following needs to be accomplished in the project:

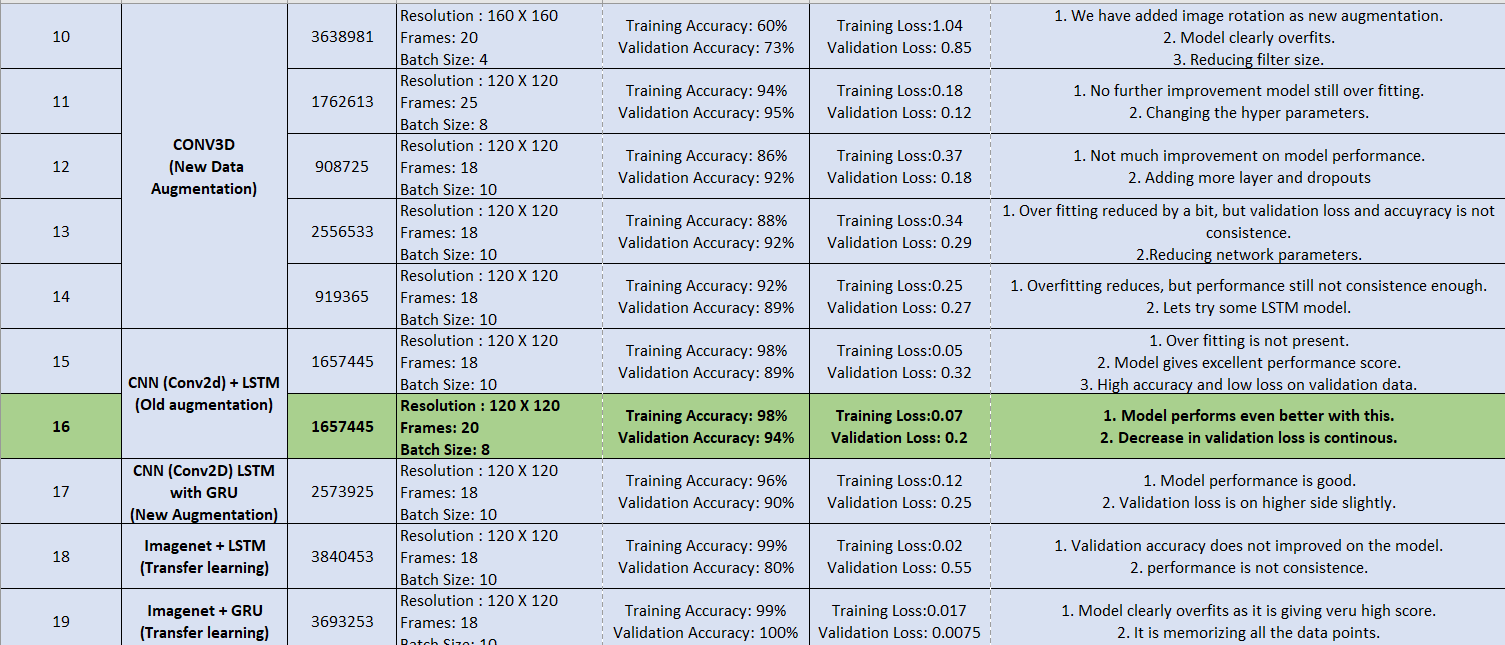
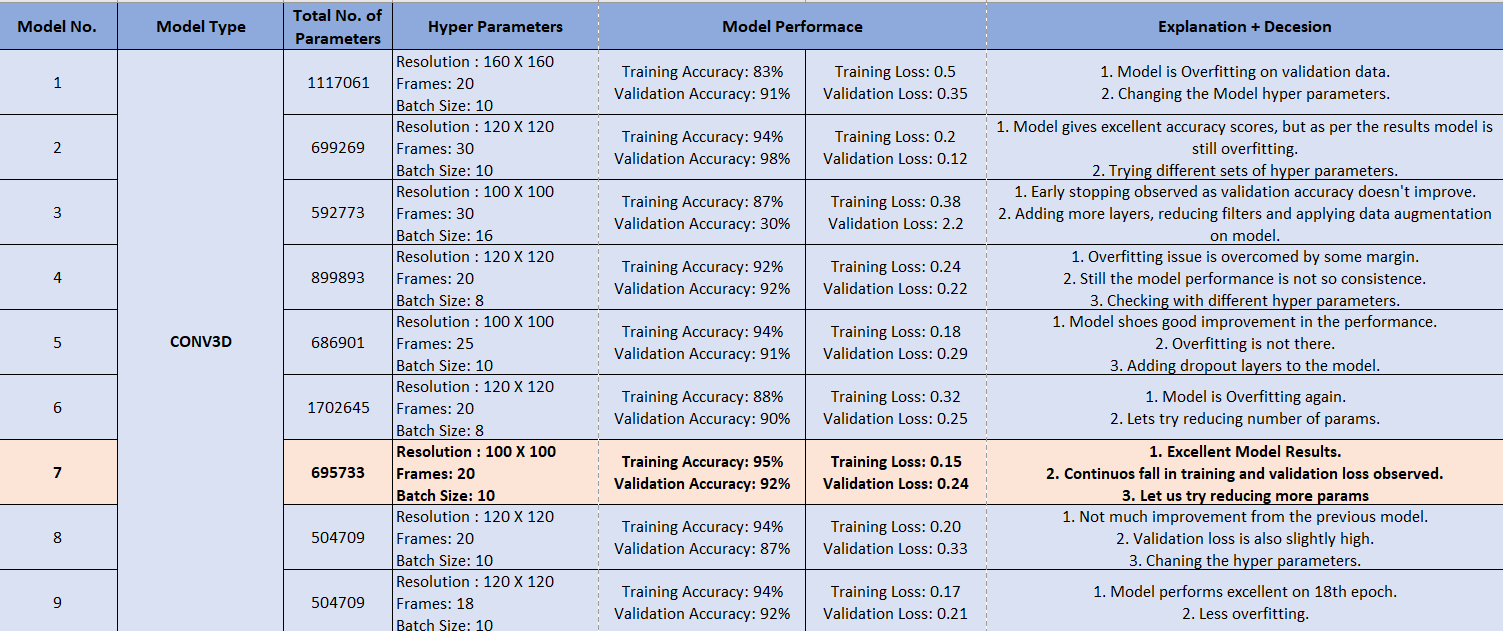
**1. Generator: The generator should be able to take a batch of videos as input without any error. Steps like cropping, resizing and normalization should be performed successfully.**

**2. Model: Develop a model that is able to train without any errors which will be judged on the total number of parameters (as the inference(prediction) time should be less) and the accuracy achieved.**

**3. Write up: This should contain the detailed procedure followed in choosing the final model. The write up should start with the reason for choosing the base model, then highlight the reasons and metrics taken into consideration to modify and experiment to arrive at the final model.**

## **Comparing all Models performance:**

**Below we have listed down all the 19 model and its performance details:**

****

### **Observations:**

1. **Out of all the 19 models, Model 7 and Model 16 gives us very good results.**
2. **Each model gives accuracy over 90% on training and validation data.**
3. **Validation loss on both the models are gradually decreasing after 10-12th epochs.**

### **Selecting the Final Model:**

## **Model 16 - CNN (Conv2D) + LSTM Model is selected as out final model.**

**Reasons:**

1. **Model 16 gives training accuracy of 98% and validation accuracy of 94%, from accuracy point of view we also have similar kind of good performance in Model 7 as well.**
2. **Model 16 gives very low training loss of 0.07 and validation loss of 0.2, which is better than Model 7.**
3. **Though we observed some fluctuation in early epochs, but after 10th epochs learning rate decreases and validation loss goes to minimum on a continuous drop.**

### **Best trained weights of Model 16 is selected from 19th epoch is:**

### **model-00019-0.0776-0.9811-0.1996-0.9400.h5**

**We will be using the following model trained weights to predict the data in future purpose.**

**-------------------------------------------------------------------------------------------**