**Gesture Recognition Project**

* **Kavya Marigowda**
* **Rakesh Krishnamurthy**

# **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 recognize 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.

**Here’s the data:** <https://drive.google.com/uc?id=1ehyrYBQ5rbQQe6yL4XbLWe3FMvuVUGiL>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODEL** | **EXPERIMENT** | **RESULT** | **DECISION + EXPLANATION** | **PARAMETERS** |
| **Conv3D** | **1** | **OOM Error** | **Reduce the batch size and reduce the number of neurons in Dense layer** | **-** |
| **2** | **Training Accuracy : 0.99**  **Validation Accuracy : 0.81** | **Overfitting**  **Let’s add some Dropout Layers** | **1,117,061** |
| **3** | **Training Accuracy : 0.65**  **Validation Accuracy : 0.52**  ***(Best weight Accuracy,Epoch:6/25)*** | **Val\_loss did not improve from 1.24219 so early stopping stop the training process. Lowering the learning rate to 0.0002.** | **3,638,981** |
| **4** | **Training Accuracy : 0.76**  **Validation Accuracy : 0.72**  ***(Best weight Accuracy,Epoch:12/25)*** | **Overfitting has been reduced but accuracy did not improved. *Trying to add more layers*** | **1,762,613** |
| **5** | **Training Accuracy : 0.83**  **Validation Accuracy : 0.76** | ***Don’t see much performance improvement. Let's try adding dropouts.*** | **2,556,533** |
| **6** | **Training Accuracy : 0.84**  **Validation Accuracy : 0.69** | **Overfitting Increase, adding dropouts has further reduced validation accuracy. Let's try to reduce the parameters** | **2,556,533** |
| **7** | **Training Accuracy : 0.84 Validation Accuracy : 0.74** | **Overfitting reduced, but validation accuracy low. Let's try to reduce the parameters.**  **Val Accuracy: 0.49, Train Accuracy: 0.54** | **696,645** |
| **8** | **Training Accuracy : 0.82 Validation Accuracy : 0.73** | **Accuracy remains below same. Let’s switch to CNN+LSTM.** | **504,709** |
| **CNN+LSTM** | **9**  **(Model-8 on Notebook)** | **Training Accuracy : 0.93 Validation Accuracy : 0.85** | **CNN - LSTM model - we get a best validation accuracy of 85%.** | **1,657,445** |
| **Conv3D** | **Apply some Data Augmentation techniques & check the model performance.** | | | |
| **10** | **Training Accuracy : 0.78 Validation Accuracy : 0.82** | **(3, 3, 3) Filter & 160 x 160 image resolution** | **3,638,981** |
| **11** | **Training Accuracy : 0.72 Validation Accuracy : 0.75** | **(2, 2, 2) Filter & 120 x 120 image resolution. Increase epoch count to 20. Network is generalizing well.** | **1,762,613** |
| **12** | **Training Accuracy : 0.87 Validation Accuracy : 0.78** | **Adding more layers.** | **2,556,533** |
| **13** | **Training Accuracy : 0.65 Validation Accuracy : 0.25** | **Very low performance. Let’s reduce the network parameters.** | **2,556,533** |
| **14** | **Training Accuracy : 0.89 Validation Accuracy : 0.78** | **After reducing network parameters, model’s performance is quite good.** | **696,645** |
| **15** | **Training Accuracy : 0.88 Validation Accuracy : 0.81** | **Reducing network parameters again.** | **504,709** |
| **CNN LSTM with GRU** | **16** | **Training Accuracy : 0.98 Validation Accuracy : 0.77** | **Overfitting is considerably high, not much improvement.** | **2,573,541** |
| **Transfer Learning(Optional)** | **17** | **Training Accuracy : 0.85 Validation Accuracy : 0.58** | ***We are not training the MobileNet weights that can see, validation accuracy is very poor.*** | **3,840,453** |
| **Transfer Learning with GRU &(Optional)** | **18** | **Training Accuracy : 0.98 Validation Accuracy : 0.93** | **Best result** | **3,692,869** |