# Gesture Recognition Dataset

Problem Statement

A home electronics company that manufactures state of the art smart televisions 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. Let's have professor Raghavan introduce you to the problem statement: 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 Some important fact about data:

Each video is a sequence of 30 frames (or images). There are 666 videos provided as training data and 100 videos provided as validation data all images in a particular video subfolder have the same dimensions different videos may have different dimensions. Specifically, videos have two types of dimensions - either 360x360 or 120x160 There are two csv(one for train, one for validation) files having path of videos Our task is to train a model on the 'train' folder which performs well on the 'val' folder as well Thus, there are two types of architecture commonly used for analysing videos, both explained below. Convolutions + RNN 3D Convolutional Network, or Conv3D

Objective :

Generator Write code for generator function so, that it can provide data in batch while training model. Also, write code for preprocessing image files in generator function Model Create models with two architectures Convolution 3D Convolution 2D + RNN a. RNN with LSTM b. RNN with GRU Output Tune model to achieve good accuracy on train as well as validation data Provide all metric with changes done in model to tune it in the write up

Dataset was provided by institute. Hence, i cannot share it here. But you can try this notebooks with any dataset that having criteria listed below:

Each video is a sequence of 30 frames (or images). all images in a particular video subfolder have the same dimensions different videos may have different dimensions. Specifically, videos have two types of dimensions - either 360x360 or 120x160 There are two csv(one for train, one for validation) files having path of videos

Link is here : <https://drive.google.com/file/d/1ehyrYBQ5rbQQe6yL4XbLWe3FMvuVUGiL/view>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Experience No | Model Result | Max. Training Accuracy | Max Validation Accuracy | Decision + Explanation |
| Model# 1 | Conv3D  No of frames = 16  No of Epochs = 20  Batch size = 64 | 0.91 | 0.875 | Observation:  we can see that batch size is not impacting training time but number of frames. We will use batch size of 64 as it seems optimal. It is commented as it takes a lot of time to run that is restricting for the notebook to completely run  Next Step:  Crop the images correctly, Reduce Parameter |
| Model#2 | Conv3D | 0.92 | 0.87 | Observation: There are no significant improvement in the result.  Next Step: we will increase the number of frames. |
| Model#3 | Conv3D  No of frames = 30  No of Epochs = 20  Batch size = 64 | 0.96 | 0.37 | Observation:  We tried to crop the images and parameters were reduced  The model return the training accuracy of ~0.96, however validation accuracy has hampered.  Next Step:  Same pooling in Conv3D layer |
| Model#4 | Conv3D  No of frames = 20  No of Epochs = 20  Batch size = 64 | 0.96 | 0.5 | The model returned the accuracy of ~0.95 which is not good fit compare to the last model  Next Steps:  Batch Normalization before MaxPooling |
| Model# 5 | Conv3D  No of frames = 20  No of Epochs = 20  Batch size = 64 | 0.97 | 0.37 | Observations:  The accuracy has been increased a bit compare to the model#4 but its still not a good fit compare model #3  Next Step:  Switching Model architecture to Conv2D+LSTM |
| Model# 6 | CNN + LSTM  No of frames = 20  No of Epochs = 30  Batch size = 32 | 0.98 | 0.87 | Observations:  The model accuracy is ~0.98 which is the best till now after increasing the epochs  Next Step:  Conv2D + LSTM with  Reduced Parameters |
| Model#7  Final Model | CNN + LSTM  No of frames = 20  No of Epochs = 30  Batch size = 32 | 1.0 | 1.0 | Observation: The model is at accuracy of 1 which should be fit model |