****Types of Architecture tried are,****

* ****CNN + RNN architecture -** passing the images of a video through a CNN which extracts a feature vector for each image, and then pass the sequence of these feature vectors through an RNN.**
* ****3D convolutional network -** The convolutional kernel moves in 3-direction (x, y, z) to calculate the convolutional output.**
* ****Transfer learning –** MobileNet is a computer vision model open-sourced by Google and designed for training classifiers. It uses depth wise convolutions to significantly reduce the number of parameters compared to other networks, resulting in a lightweight deep neural network.**
* ****GRU and LSTM designs are explored.****

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D – Base Model**  **Sequential Model with 3 Conv3D, each Conv layer is followed by MaxPooling3D, Batch normalization is added in all the layers, two dropouts of 25 %** | **- Total params: 4,050,085**  **- Training Accuracy: 0.83**  **- Validation Accuracy: 0.125** | **Overfitting Model let’s try to change the Frames and Epochs. Too may layers with may parameters is causing the model overfit.** |
| **2** | **Conv3D - 16 Frames with 20 Epochs**  **Same as above, the no of frames is 16 and trained for more EPOCHS** | **- Total params: 2,067,621**  **- Training Accuracy - 0.92**  **- Validation Accuracy - 0.625** | **Reduced the parameters and used 14 frames with 20 Epochs. Score is improved but the model is.**  **Overfitting, let’s try to change the Frames and Epochs** |
| **3** | **Conv3D - Frames-30, Epocs-30** | **- Total params: 9,006,245**  **- Training Accuracy - 0.97**  **- Validation Accuracy - 100** | **Increased the parameters, 100% validation accuracy is a symptom of model memorized the pattern instead of generalizing, lets reduce the parameters further to see we can achieve the optimal accuracy** |
| **4** | **CONV3D Model (30 Frames with 20 Epochs)**  **Reduced the parameters** | **- Total params: 5,618,245**  **- Training Accuracy - 0.95**  **- Validation Accuracy - 0.5** | **Reduced the parameters compared to previous model, Model is performing well on Training but not validation** |
| **5** | **CONV3D Model (20 Frames with 20 Epochs)**  **Increased the parameters by adding one additional layer** | **- Total params: 1,907,909**  **- Training Accuracy - 0.935**  **- Validation Accuracy - 0.5** | **Added one more Conv3D and Maxpooling3D layer. Model is performing well on Training but not validation.** |
| **6** | **CONV3D Model (Frames:20, Epocs-20, BS-64, reduced kernel to (2,2,2), switching BatchNormalization before MaxPooling)** | **- Total params: 1,301,045**  **- Training Accuracy - 0.99**  **- Validation Accuracy - 0.5** | **reduced kernel to (2,2,2), switching BatchNormalization before MaxPooling. Model is performing well on Training but not validation** |
| **7** | **Conv2D+LSTM** | **- Total params: 3,084,133**  **- Training Accuracy - 0.97**  **- Validation Accuracy - 0.81** | **Model is comparatively performing well on both training and validation** |
| **8** | **CNN LSTM with GRU** | **- Total params: 2,429,285**  **- Training Accuracy - 0.99**  **- Validation Accuracy - 0.93** | **Model is performing very well on both training and validation** |
| **9** | **Transfer Learning (MobileNet) with LSTM** | **- Total params: 4,611,781**  **- Training Accuracy - 1.0**  **- Validation Accuracy - 1.0** | **percentage of error is zero in the accuracy score which is sign of overfitting, 100% in training and validation is not a optimal model** |
| **10** | **Transfer Learning (MobileNet) with GRU** | **- Total params: 4,284,613**  **- Training Accuracy - 1.0**  **- Validation Accuracy - 1.0** | **percentage of error is zero in the accuracy score which is sign of overfitting, 100% in training and validation is not a optimal model** |
|  |  |  |  |
| **Final Model** | **CNN LSTM with GRU** | **- Total params: 2,429,285**  **- Training Accuracy - 0.99**  **- Validation Accuracy - 0.93** | **Model is performing very well on both training and validation. The difference between the validation and training score explains the model is not overfitting, hence this model is considered as optimal** |