Project Phase 2 Report

## Recognition of Stretching Body Through Video

Muhammad Hussain Anwaar

Graduate Student

Industrial and Systems Engineering

Texas A&M University

**Improvement Strategies:**

One of the major flaws I detected in the performance of the previously trained model was that about 20% of the times, model was not able to differentiate between a person talking through hand gestures and stretching. It was labelling them as Stretching while in reality there was No Stretching happening in the video.

Secondly, the different type of stretches that were performed were mostly done by 4 or 5 individuals. So, I felt that if the same person is performing different stretches, maybe this factor would affect the model’s performance.

**Data Generation:**

To address above issues hindering the performance of model, I thought of expanding the training dataset. Previously, I had 230 videos out of which 130 were “Stretching” videos and 100 were “No-Stretching” videos. I gathered about 170 more videos from “Youtube.com” out of which 76 were of category “No-Stretching” and about 94 were of category “Stretching”.

Now I had 400 videos in total and here’s the class proportion:

1. Stretching: 224 Videos
2. No-Stretching: 176 Videos

Test Dataset was also created out of these 400 videos. About 61 videos were randomly selected from these 400 videos to be used as test dataset. The class proportion in test dataset is as follows:

1. Stretching: 34 Videos
2. No-Stretching: 27 Videos

**Advanced Model Architectures:**

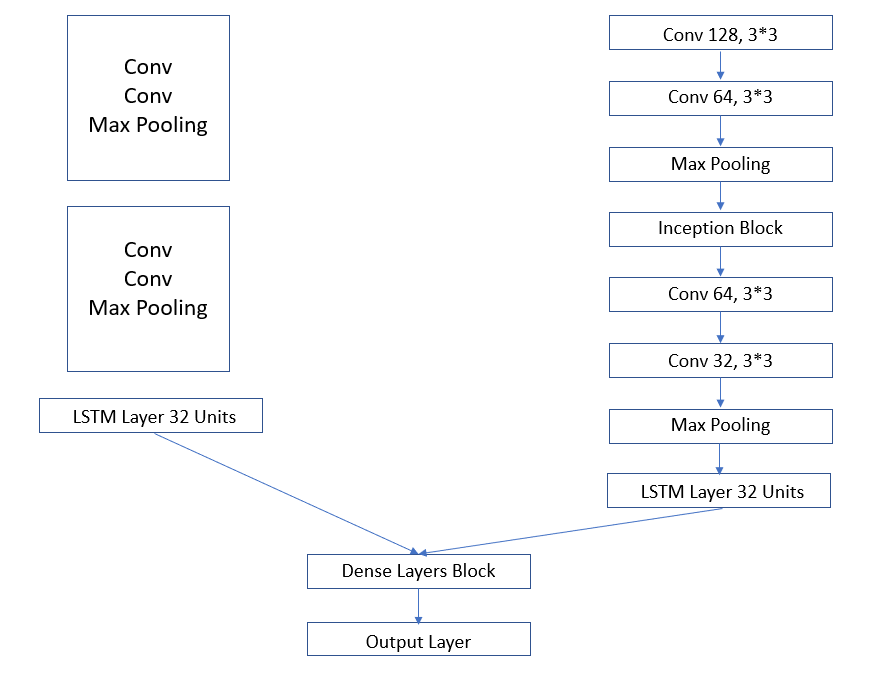
To improve the model performance further, I decided on using some of the advanced model architectures and techniques to see how they would perform on this task of “Stretching” recognition from a video.

1. **Inception Modules in Time Distributed CNN and LSTM Architecture:** Time Distributed CNN and LSTM Architecture gave the best performance of about 73% in the first phase of the project. So, this time I thought of using the same architecture as a baseline and improve on it. So, now, idea is to use 2 to 3 inception blocks in the CNN part of the architecture to extract features from the video frames and then use an LSTM.
2. **Using Model Ensemble Technique:** Model Ensembling is a powerful technique which can be utilized to achieve great performance. The idea here is to try to use Different Architectures and try Ensembling them to see if their collaboration can outer perform one single state of the art model.

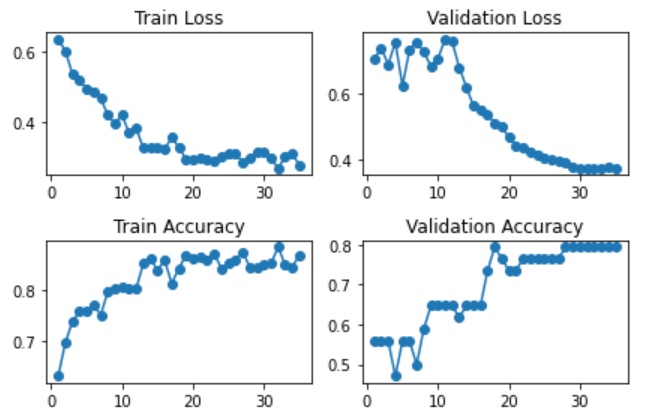
**Model Architectures:**

**Model Ensemble Architecture:**

Model Ensembling is a powerful tool, where we can combine the two models together and where they both would try to learn through the data in parallel and then combined at the output level. Motivation to join two models is that they are of different architectures and the hope is that they would learn different features that might help with the predictions. Thus, I combined the two architectures, one is a deep CNN and LSTM architecture, while the other comprises of Time Distributed CNN with Inception Blocks embedded. Below is the figure defining the architecture.



The above-mentioned model was trained on 25, 30 and 35 epochs, with batch size of 5, 10 and 15. I used “Adam” as the optimizer and the finest results were obtained from epochs=35 and batch size=15. The test accuracy from this ensemble architecture is 82% which is an improvement of 3% from the previously defined architecture, and a highly significant improvement from the architecture of phase 1. The figure below gives the model’s performance on training data w.r.t loss and accuracy.



**Next Steps:**

For next steps, I would try ensembling 3-D CNN model with the existing architecture and enhance the dataset a bit by adding a few more videos in the training set and see if these techniques can help me with the task.