**Classification of American Sign Language using online RESTful application.**

***Darshil Shah, Sachi Shah, Yash Vakil, Rajvi Desai.***

*Arizona State University, School of Computing, Informatics ,*

*and Decision Systems Engineering.*

1. INTRODUCTION

This document report provides desired layout to develop online application service that accepts Human Skeletal key points of a sign video and return the label of the sign in a JSON response. The document contains the information about extraction of key points from the videos using Tensor Flow’s Pose Net library and four different deep learning models that can classify American Sign Languages into six different signs. i.e {buy, fun, hope, really, communicate, mother }. Moreover, it also contains the information about hosting service using flask API on ‘PythonAnywhere’ and steps involving handling Http requests coming from different users.

1. TECHNICAL APPROACH

Firstly, we have accumulated all the raw video data sets which have been recorded as a part of Assignment-1, and extracted frames of the particular timeline. Then, we used Tensor Flow’s Pose Net library in order to extract key points from the images , which is considerably used as training data for models. We have tried three different approaches to preprocess data and picked the one which gives best accuracy for the trained models.

***Approach-1:*** Scaling down raw data using Universal Normalization technique and extracted few features like- Standard Deviation , Moving Mean of Window size 5, Zero Crossing Rate, Dynamic Time Warping distance and built feature matrix. Then we applied PCA on feature matrix and using K-fold Cross validation we trained four deep learning models named as **Convolutional Neural Network , K nearest neighbor , Support Vector Machine and Random Forest.** The average accuracy of the given models lied between 60-65%

***Approach- 2:*** As a part of second approach , we expelled some features by observing the movement of each body part in videos for different signs and made feature matrix of only important features. Then we apply Standard Scaler and Min Max Scaler in order to normalize data and trained our models using the first approach.

Somehow, we were able to increase the average accuracy of the models by 10%.

***Approach -3 :*** We have observed in second approach that , our model is only considering the static coordinates of each body part, so we subtracted each coordinates of different body parts from static body part and processed the data in same manner. So , by doing this approach we got our highest average accuracy which lies between 85% to 90%.

1. INITIAL FEATURE EXTRACTION
2. *Zero Crossing Rate*
3. *Moving Average Window*
4. *Standard Deviation*
5. *Dynamic Time Warping Distance.*

***Zero Crossing Rate* :** The zero crossing rate is the rate of sign- changes along with a signal, i.e the rate at which the signal changes from positive to zero to negative or vice versa. Zero Crossing Rate can be used as primitive pitch detection algorithm for signal processing.

***Moving Average Window :*** Moving Average is optimal for reducing random noise while retaining a sharp step response. This makes it the premier filter for the time domain encoded signals

***Standard Deviation:*** The standard deviation is measure of how far the signal fluctuates from the mean. It also depicts how does data disperse near mean of particular data series.

***Dynamic Time Warping Distance :*** DTW measures the similarity between two temporal series data. Any linear sequence data can be analyzed with DTW, it aims at aligning two sequences of feature vectors by warping the time axis iteratively until an optimal match between the two sequences is found.

***Feature Engineering:***

We have expelled few features by observing the movement of each body part for different signs and made feature matrix with only important features. Below is the list of features that we considered for training models.

["nose\_x", "nose\_y", "leftShoulder\_x", "leftShoulder\_y", "rightShoulder\_x", "rightShoulder\_y","leftElbow\_x", "leftElbow\_y", "rightElbow\_x", "rightElbow\_y", "leftWrist\_x", "leftWrist\_y”, "rightWrist\_x", "rightWrist\_y"]

Here, we observed that the coordinates value of each body part show static position for given time, so we have subtracted each body part coordinates value from corresponding static body part’s coordinates. Here , we have considered “nose” as static body part and subtracted each body part with corresponding X and Y coordinates.

The above mentioned approach would become more simpler for the models to understand the movement of each body part , as we have relative position for each body part, model can easily predict certain gestures by examining positive or negative sides of coordinates.

1. MODELS USED:
2. *K nearest neighbor*
3. *Convolution Neural Network*
4. *Support Vector Machine*
5. *Random Forest*

***K Nearest Neighbor:*** The K Nearest Neighbor classifier is one of the most simple machine learning algorithms that simply relies on the distance feature vectors. It classifies unknown data by finding the most common classes among k nearest examples. The majority vote of the class label is assigned to unknown data. As KNN is lazy learning algorithm, it works more efficiently when our dataset has been distributed in multi classes.

***Support Vector Machine :*** The core idea of Support Vector Machine is to find hyperplane that separates between two sets of objects having different classes. It uses a technique called as kernel trick to transform data and based on these transformation it finds an optimal boundary. It is considered as one of the most robust and accurate algorithm among other classifiers.

***Random Forest :*** Random Forest is an ensemble classifier , it takes multiple individual models and combined them into more powerful aggregate model. So, let say if we have different individual model, then there might be case that they work efficiently because some part of data set would be over fitted to model. So by combining them, we can reduce the chances of error. Random forest built upon by aggregating n possible decision trees which might be generating by randomly picking data set row as root. So, as the dataset would be increasing, possibilities of generating random decision trees would also increase and aggregating different decision tree models lead to increase efficiency of aggregated model, too.

1. REFERENCES:
2. <https://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch15.pdf>
3. <https://en.wikipedia.org/wiki/Zero-crossing_rate>
4. <https://en.wikipedia.org/wiki/Dynamic_time_warping>
5. https://community.alteryx.com/t5/Data-Science-Blog/Why-use-SVM/ba-p/13844