Assignment 2 Report

CSE 572: Data Mining
Spring 2018

Submitted to:
Professor Ayan Banerjee
Ira A. Fulton School of Engineering
Arizona State University

Submitted by:
Jagadeesh Basavaraju (1213004713)
Sahan Vishwas (1213094049)
Shailee Desai(1210936321)
Suraj Kattige (1211230381)
Vishal Vaidhyanathan (1211138809)

Table of Contents

1. Introduction……. 2
2. Team Members…… 2
3. Phase 1 - Data Collection…… 2
4. Phase 2 - Feature Extraction and Selection……. 3
   4.1. Task 1 - Segmentation of raw data into classes……. 3
   4.2. Task 2 - Feature Extraction……..... 3
      4.2.1. Techniques Used……. 4
      4.2.2. Intuition…… 5
      4.2.3. Plots……. 8
   4.3. Task 3 - Feature Selection using PCA…… 13

1. Introduction

This project is a part of CSE 572 - Data Mining course for Spring 2018 semester. It is an experiment to develop an intelligent system that can understand human gestures via American Sign Language. This part of the project will cover the following:

- record and identify set of known human gestures
- segment sequence of gestures into classes
- predict and identify unknown gestures based on recorded observations

The various gestures are differentiated based on their respective features and reduced to a suitable format to make predictions in the future. Methods used in this phase are feature extraction, feature selection and dimensionality reduction using Principal Component Analysis (PCA).

2. Team Members

Following are the group members of this project:

- Jagadeesh Basavaraju(jbasavar@asu.edu)
- Sahan Vishwas (shvishwa@asu.edu)
- Shailee Desai (smdesai2@asu.edu)
- Suraj Kattige (suraj.kattige@asu.edu)
- Vishal Vaidhyanathan (vvaidhya@asu.edu)

3. Project Phase 1: Data Collection

The first phase of the project was data collection. One member from the group was designated to gesture for each of the given words. Wristband sensors were used to capture the movements of each of the words, in front of a Kinect, around 20 times each. The data was collected from 37 groups and stored in comma separated value (CSV) files. Words used for gestures were:

- About
- And
- Can
- Cop
- Deaf
- Decide
- Father
- Find
- Go out
- Hearing

4. Project Phase 2: Feature Extraction and Selection

This phase involves feature extraction and application of Principal Component Analysis (PCA) of the Phase I data collected. The number of raw sensor data used are as follows:

- 6 from Accelerometer,
- 6 from Gyroscope,
- 6 from Orientation and,
- 16 from EMG sensors.

The following tasks were performed in this phase of the project:

4.1: Task 1 - Segmentation of raw data into classes

In Task 1 of the project, we receive the data for all the words gestured in phase 1 of the project in CSV files from the following groups: *DM07, DM09, DM11, DM13, DM16, DM27, DM29, DM31 and DM34*. The remaining groups were omitted as there was some data missing in the files of these groups. The input to this phase were CSV files containing columns as different sensors and values of the sensors from time t=1 to t=45 as rows. Each CSV had data about the gesture when actioned once. Depending on the number of times a gesture was performed by each group and number of groups considered, that many files will be present for that particular gesture. Similarly files will be present for remaining 9 gestures as well. So totally, (*number_of_groups*number_of_gestures*number_of_actions)* CSV files were the input for Task 1.

Output of Task 1 will be one file per gesture containing all the actions of all groups for that gesture, for all sensors as rows and values of the sensors from time t=1 to t=45 as columns.

Below is the depiction of one of those output files look like.

Action 1 Acc X 2 3 4 5 1 5 1 6 2 7 8 3 2 1 3 ----------
Action 1 Acc Y 2 3 4 5 1 5 1 6 2 7 8 3 2 1 3 ----------
|
|
Action 2 Acc X 2 3 4 5 1 5 1 6 2 7 8 3 2 1 3 ----------
|
|

These 10 output files were to the Task -2 for feature extraction. These 10 output files are stored in a folder named "*Task-1-Output*".

The code for task 1 is present in the file named "assignment_2_task_1.m". In order to successfully run this file, it has to be made sure that the folders of different group are present in DM folder in the directory where this file is present.

4.2: Task 2 - Feature extraction

The second task of the project is to extract critical features by applying techniques like FFT, DWT, RMS and STD. We'll explain all the techniques used to extract the features that help us differentiate between various gestures. Our input consists of data from 34 sensors but our goal is to select extract features from each of the techniques mentioned below. In order to do this, we wanted to know the subset of sensors on which the mentioned techniques can be applied to extract features out of it. So for that, we figured out an algorithm for which the input is files from task 1 and the different feature extraction techniques. The output from the algorithm will be a subset of sensors for each technique.

4.2.1: Techniques Used

Fast Fourier Transform (FFT)

FFT is used to extract the most common occurrences of values in our data and then select those as features that represent a particular sensor for a gesture. FFT only gives us frequency information but we lose out on all temporal data. In our implementation, we applied the in built Matlab function fft function on the data output from Task 1. Then we selected the top 5 sensors as explained above. From each of the selected sensors, we sort and select the top 4 peak values as features as explained above.

Discrete Wavelet Transform (DWT)

DWT does not lose out on temporal data entirely like FFT. It gives us an intuition of both frequency and time information of our data. It would give us an idea of most frequent occurrences along with info of when that event occured. Here again, we used the in built matlab dwt function on the data from Task 1 and again we used similar methods to extract the top 5 sensors for this technique. From each of the selected sensors, we sort and select the top 4 peak values as features as explained above.

Root Mean Square (RMS)

Root Mean Square on time-series data gives us a single value that could lead to an interesting feature depending on the data. Certain type of data might be suitable for RMS analysis as it helps differentiate between them better. Here, we applied the rms function from matlab on the data output from Task 1. Then we selected the top 5 sensors as explained above. From each of the selected sensors, rms gives you one value which is selected as a feature.

Standard Deviation (STD)

STD again is similar to RMS where a time-series data can be compressed to a single value and used as a feature. For STD, we applied the std function from matlab on the data output from Task 1. Then we selected the top 5 sensors as explained above. From each of the selected sensors, std gives you one value which is selected as a feature.

Average (AVG)

AVG is similar to RMS where a time-series data can be compressed to a single value and used as a feature. For AVG, we applied the mean function from matlab on the data output from Task 1. Then we selected the top 5 sensors as explained above. From each of the selected sensors, mean gives you one value which is selected as a feature.

4.2.2: Intuition behind selecting particular sensors for each of the techniques

The algorithm uses correlation coefficient to see which set of sensors for a particular technique differentiates the gestures better. Lesses the correlation coefficient calculated between two gestures, greater is the dissimilarity between them with respect to a particular sensor. Based on this criteria, 5 sensors which differentiate the gestures better are selected for feature extraction.

Algorithm to select sensors to be considered by each technique:-
FileName: program_to_select_sensors.m
Input: csv file corresponding to each gesture obtained in Task 1, List of techniques
Output: List of sensors for each technique

1. For each technique(FFT, DWT, STD, RMS)
    a. For each sensor(ALX, ALY, ...)
        i. For each gesture(And, About, ...)
            1. Apply the technique on all actions of the gesture on the sensor
            2. Take mean of all action values to obtain a vector (1X45)
        ii. Calculate the similarity between all combination of 2 gestures using the mean valued vector obtained in previous step and take the sum of all to give a similarity value between gestures for the sensor.
    b. Print the 5 sensors which have least similarity value for the technique.

Correlation coefficient (corrcoef function of matlab) was used to find the similarity between two mean valued vectors (1X45). For STD, RMS and AVG, as they don't output a vector, instead a single value is returned, we used minimum(x1,x2)/maximum(x1,x2) as a similarity measure.

Output of the above algorithm
Sensors selected by FFT: "EMG3R","EMG6R","EMG5R","EMG4R","EMG1R"
Sensors selected by DWT: "EMG3R","EMG4R","GLX","GLY","GLZ"
Sensors selected by RMS: "EMG6L","EMG2L","EMG7L","EMG0L","EMG3L"
Sensors selected by STD: "OPL","ALX","EMG2L","EMG7L","ALY"
Sensors selected by AVG: "GRY","GRX","GRZ","ARX","ALZ"

Now that which sensors to concentrate on for each technique has been selected, feature extraction will be done by applying the techniques on only the sensors selected by the algorithm. The output will be a feature matrix for each gesture with rows specifying multiple actions of the same gesture and columns specifying different features extracted using the below mentioned techniques.

Algorithm to extract features from the sensors and techniques:-

FileName: assignment_2_task_2_final.m

Input: csv file corresponding to each gesture obtained in Task 1, List of techniques, List of sensors for each technique

Output: Feature Matrix for each gesture

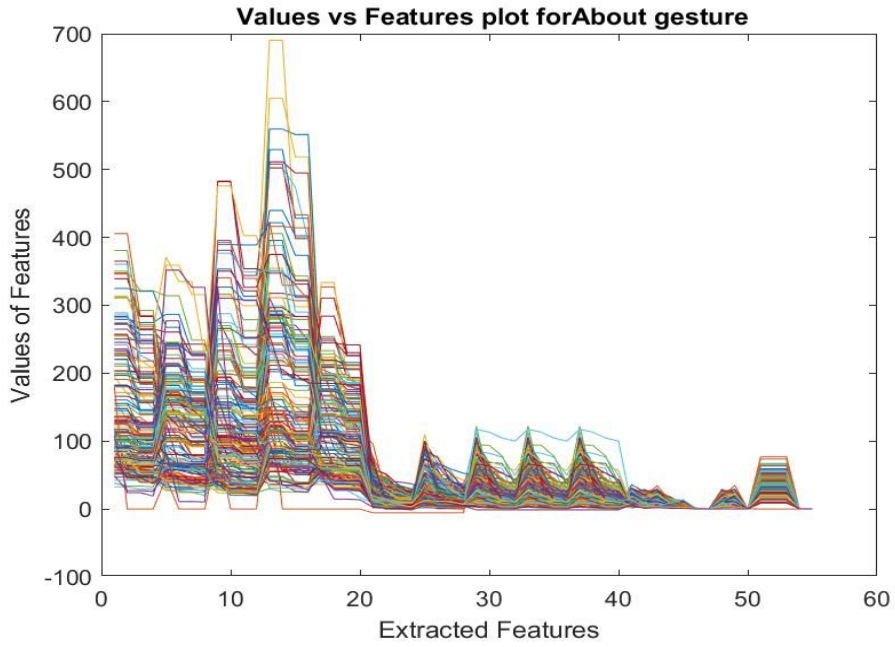
1. For each gesture
    a. FeatureMatrix = empty
    b. For each technique
        i. For each sensor
            1. Apply the technique on all the actions of the gesture for the particular sensor.
            2. Sort the values for each action in descending order for FFT and DWT to consider peak values. This step is not necessary for RMS and STD as the output is not a vector, instead a single value and hence can be considered directly as feature.
            3. Select only highest 4 values (peak) for each action as features.
            4. Stack these action-feature values horizontally to the FeatureMatrix. Rows: Actions, Columns: Features Extracted.
    c. Save the FeatureMatrix in a csv which will be used as input to task 3.
    d. Plot the feature matrix on a graph with features extracted on x-axis and values for the features on y-axis. Multiple actions of the same gesture will be plotted in the same graph.

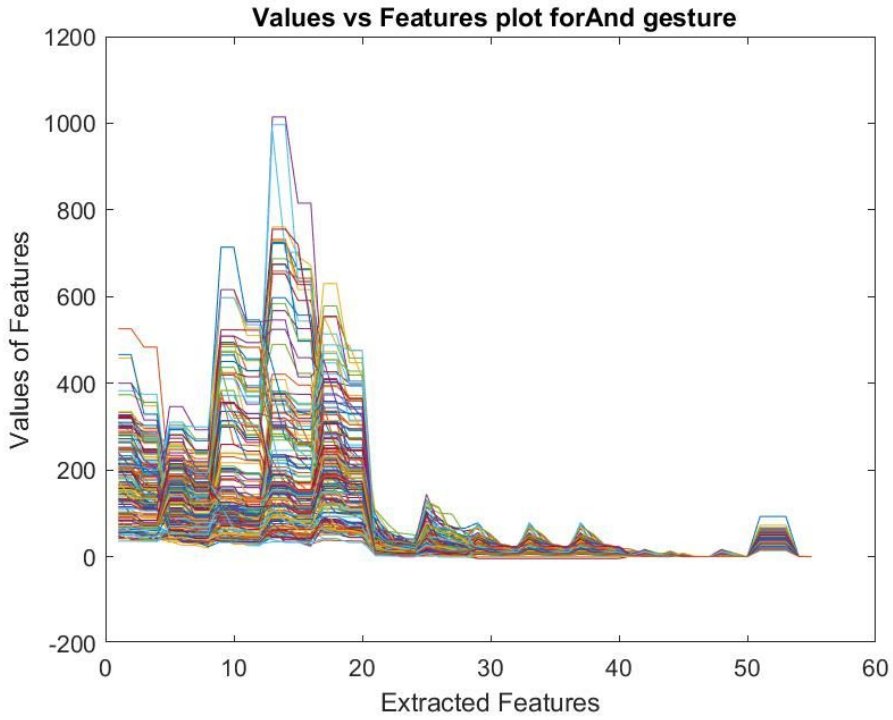
The total number of features will be the sum of below ones and in the same order:-

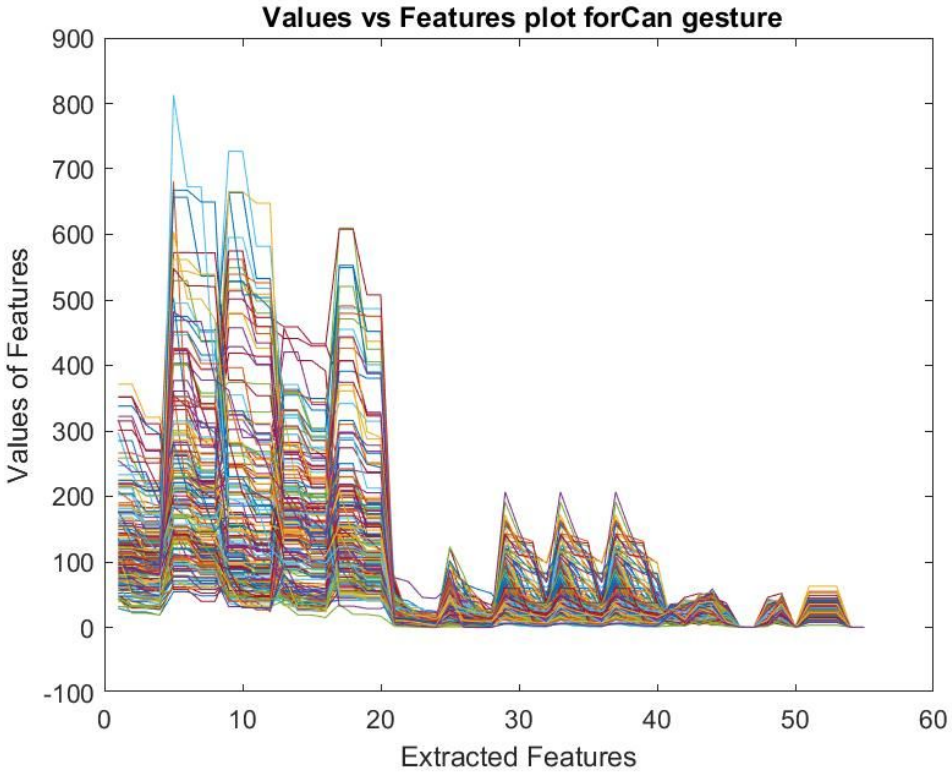
1. 4 peak points from each of the 5 sensors selected for FFT.
2. 4 peak points from each of the 5 sensors selected for DWT.
3. 1 point from each of the 5 sensors selected for RMS.
4. 1 point from each of the 5 sensors selected for STD.
5. 1 point from each of the 5 sensors selected for AVG.

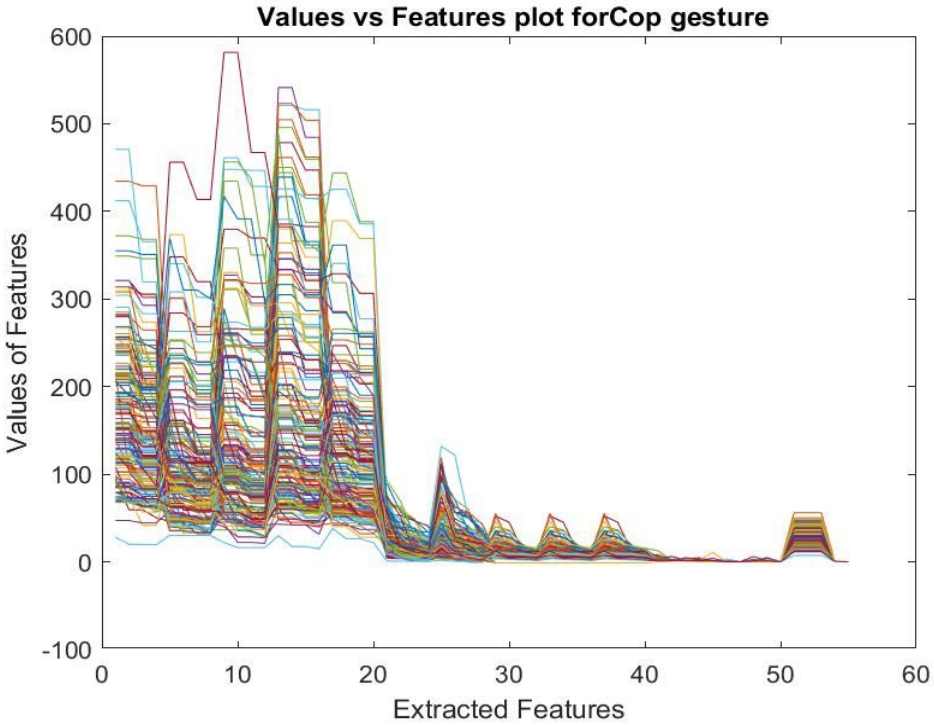
4.2.3: Plots

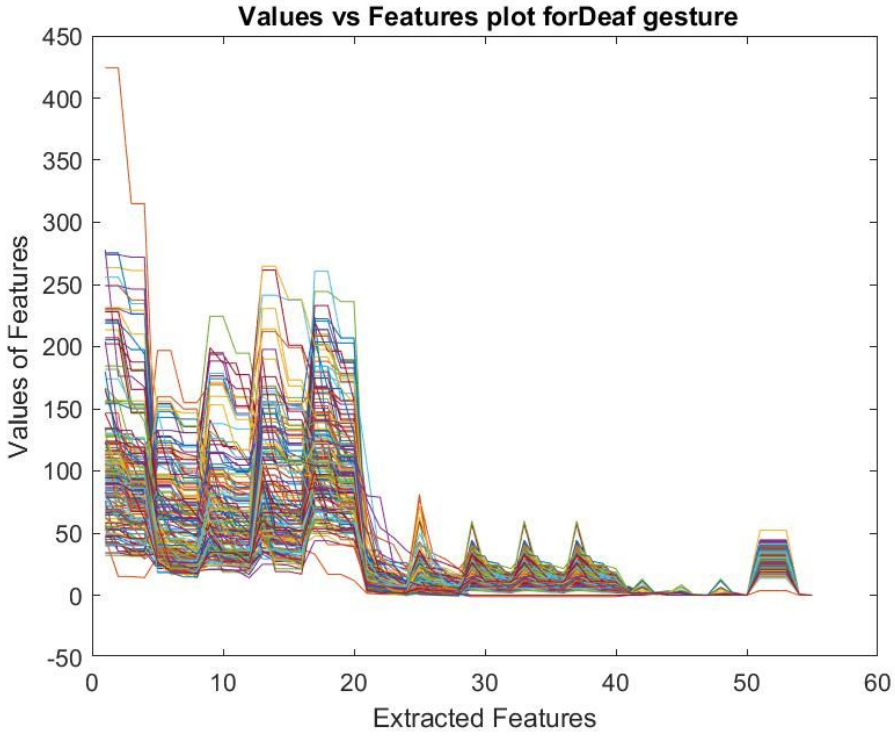
Plots of Features Extracted vs Values for all actions of the individual gestures have been depicted below.

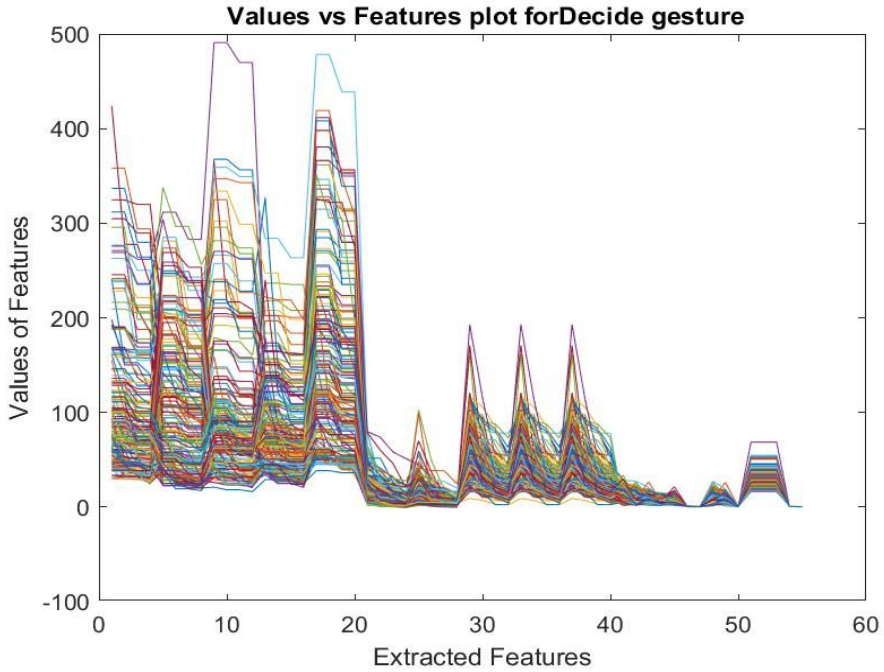


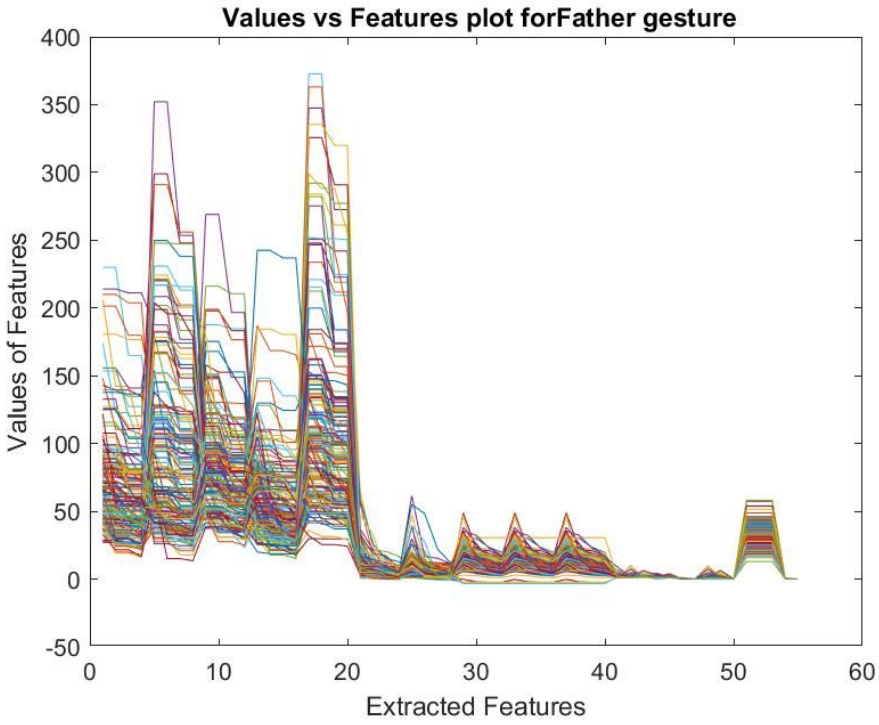


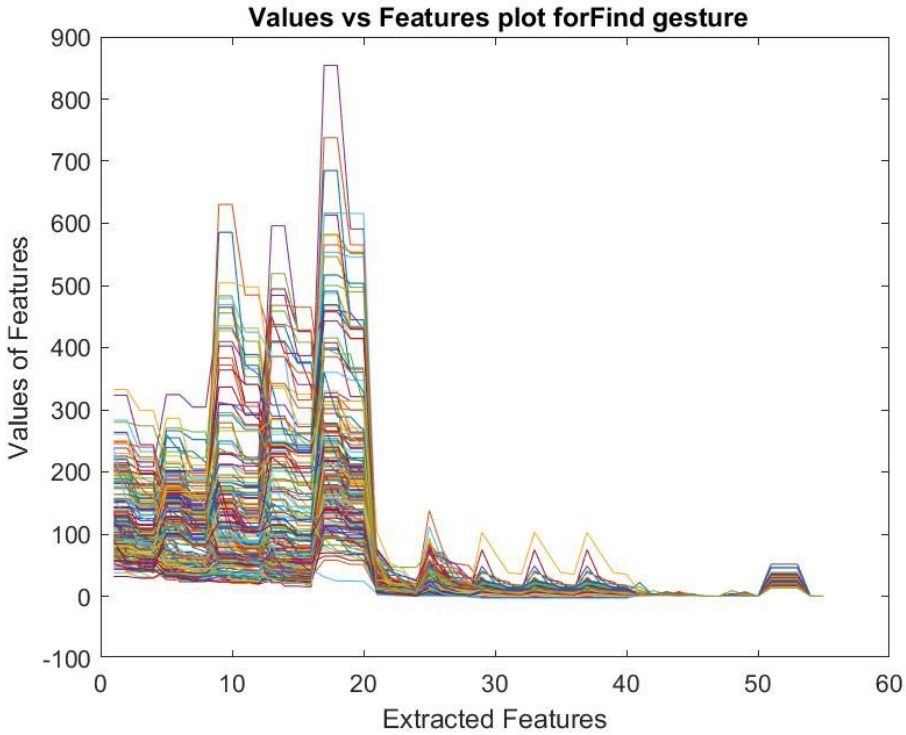

Values vs Features plot forCan gesture
Values of Features
Extracted Features
900
800
700
600
500
400
300
200
100
0
-100
0
10
20
30
40
50
60

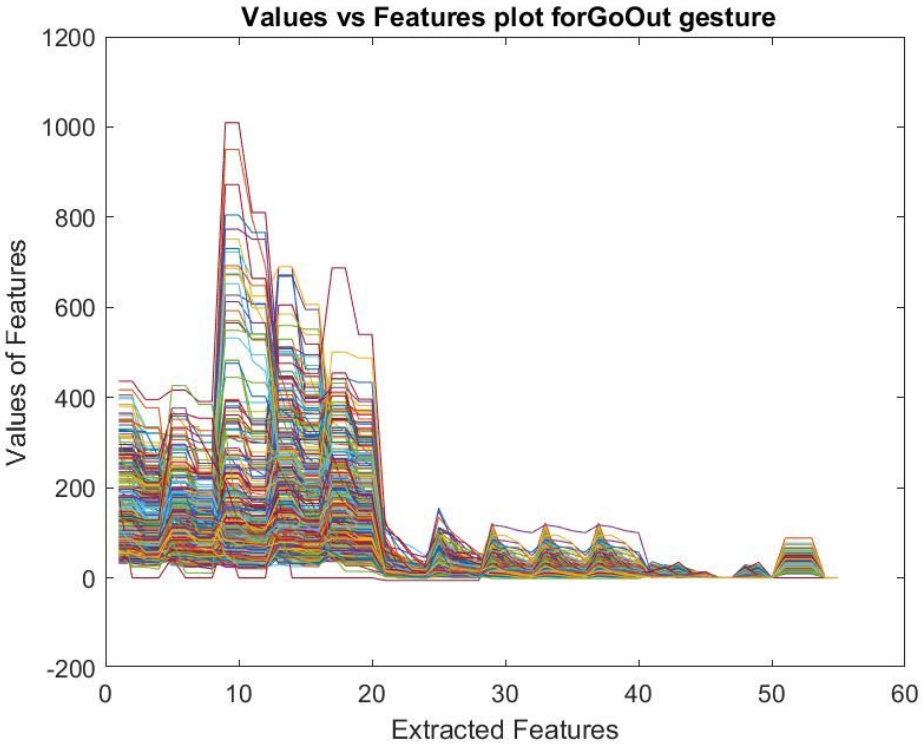

Values vs Features plot forCop gesture
Values of Features
Extracted Features
600
500
400
300
200
100
0
-100
0
10
20
30
40
50
60

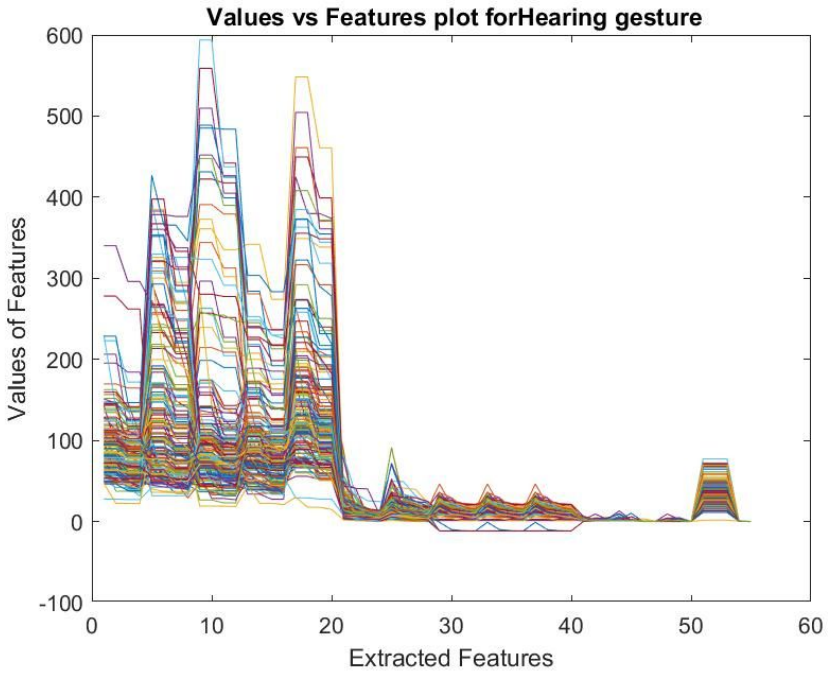

Values vs Features plot forDeaf gesture
Values of Features
Extracted Features
450
400
350
300
250
200
150
100
50
0
-50
0
10
20
30
40
50
60


Values vs Features plot forDecide gesture
Values of Features
Extracted Features
500
400
300
200
100
0
-100
0
10
20
30
40
50
60


Values vs Features plot forFather gesture
Values of Features
Extracted Features
400
350
300
250
200
150
100
50
0
-50
0
10
20
30
40
50
60


Values vs Features plot forFind gesture
Values of Features
Extracted Features
900
800
700
600
500
400
300
200
100
0
-100
0
10
20
30
40
50
60


Values vs Features plot forGoOut gesture
Values of Features
Extracted Features
1200
1000
800
600
400
200
0
-200
0
10
20
30
40
50
60


Values vs Features plot forHearing gesture
Values of Features
Extracted Features
600
500
400
300
200
100
0
-100
0
10
20
30
40
50
60

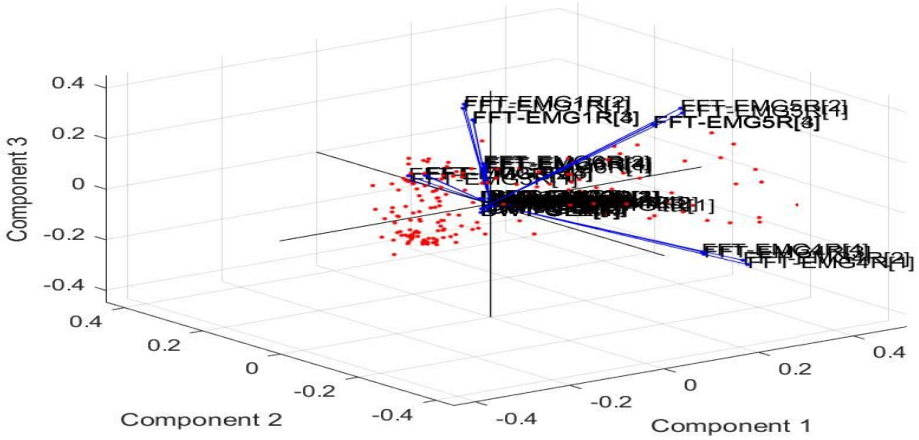
4.3: Task 3 - Feature Selection using PCA

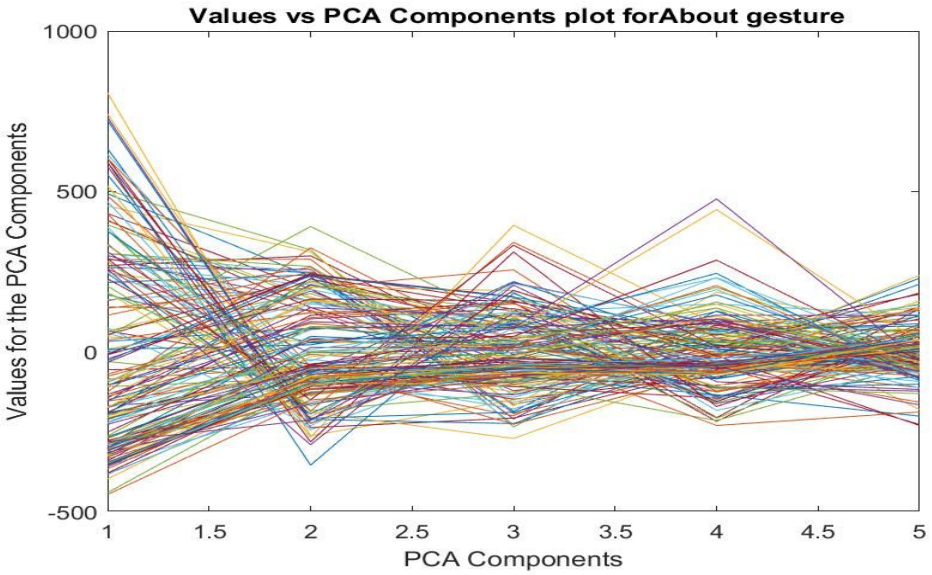
Upon extraction features in Task II, we find the features with the most variance on the data set. This can be accomplished by performing *Principal Component Analysis (PCA)*.

Feature Matrix generated for all words will be fed individually to the PCA (pca function in matlab). The graph for observations-components matrix is plotted for each of the gesture. The percentage of variance explained by each of the top 5 principal components for all the words are also printed. The file to be run for this task is "*assignment_2_task_3.m*".

Below plots show the contribution of each of the 55 features extracted to the first 3 Principal Components of all gestures.

Plot showing contribution of extracted features to first 3 Principal Components for About gesture:-



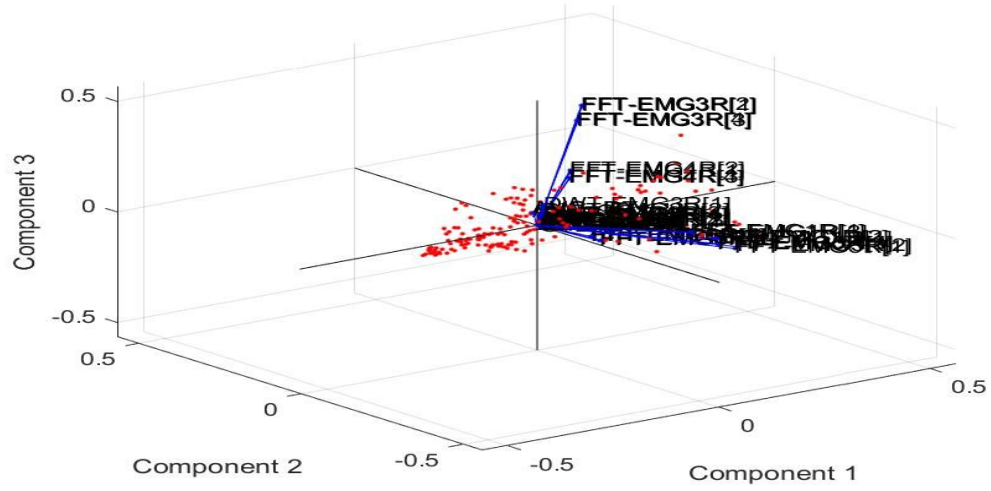


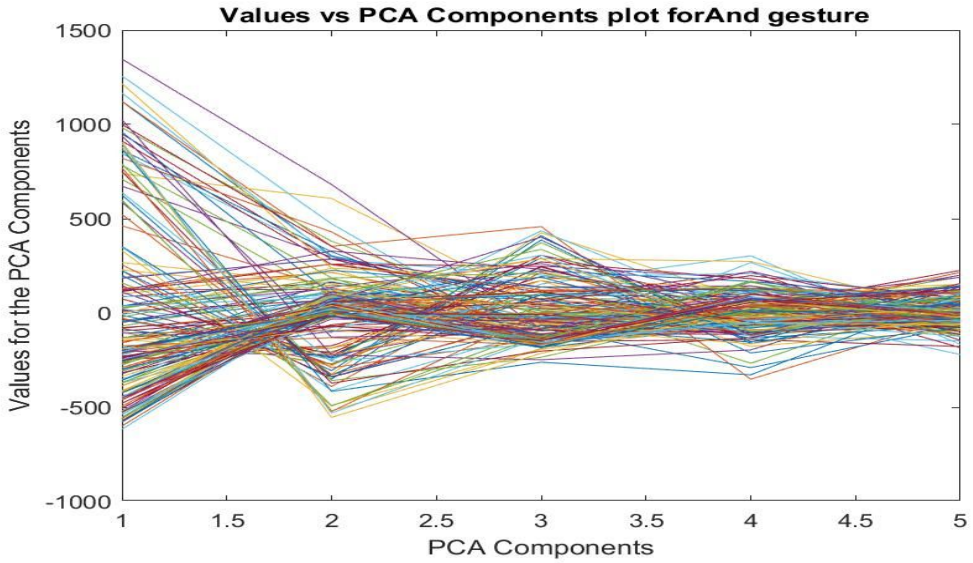
Percentage of variance covered by the top-principal components for **About**:

51.3009
18.8196
12.8947
9.3075
4.5988

Variance captured by top 5 Principal Components: 96.9215
Variance captured by top 3 Principal Components: 83.0152

Plot showing contribution of extracted features to first 3 Principal Components for And gesture:-





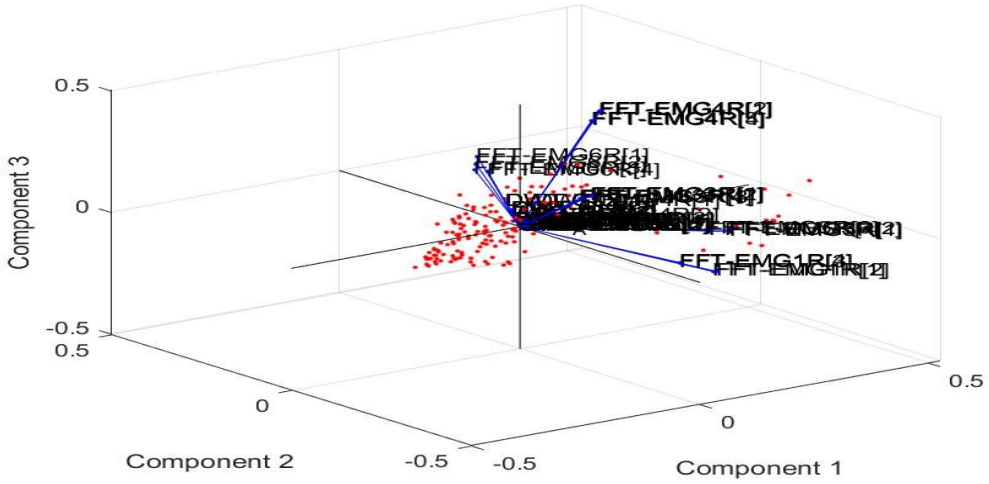
Percentage of variance covered by the top 5 principal components for **And**:

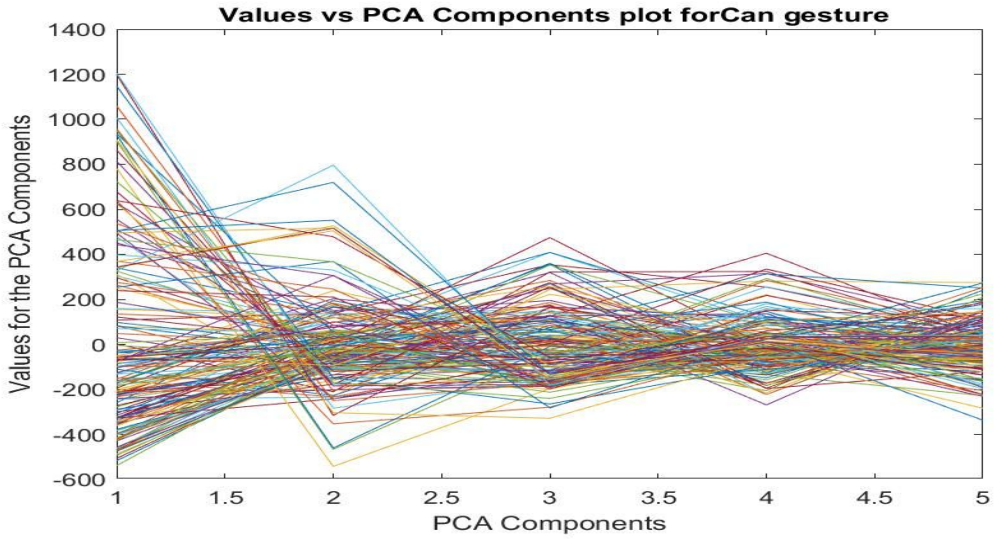
75.4338
9.6264
5.3238
3.3853
2.5869

Variance captured by top 5 Principal Components: 96.3562

Variance captured by top 3 Principal Components: 90.384

Plot showing contribution of extracted features to first 3 Principal Components for Can gesture:-





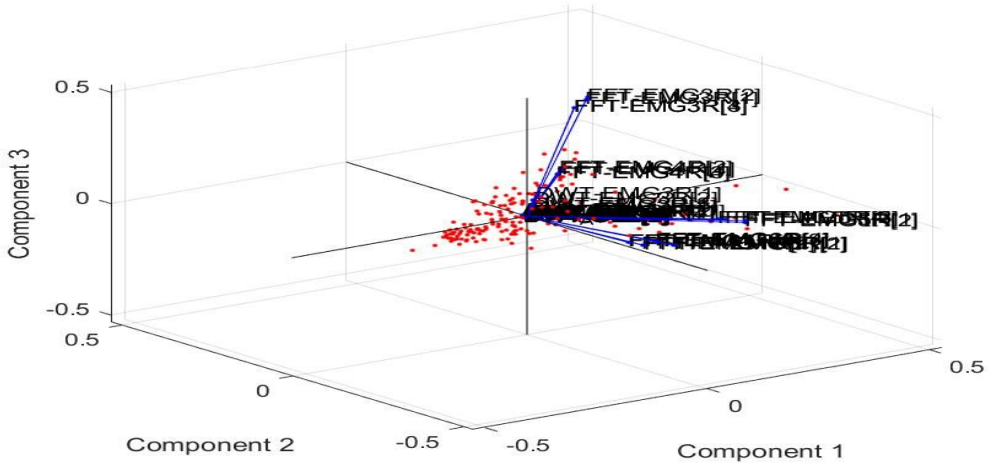
Percentage of variance covered by the top 5 principal components for **Can**:

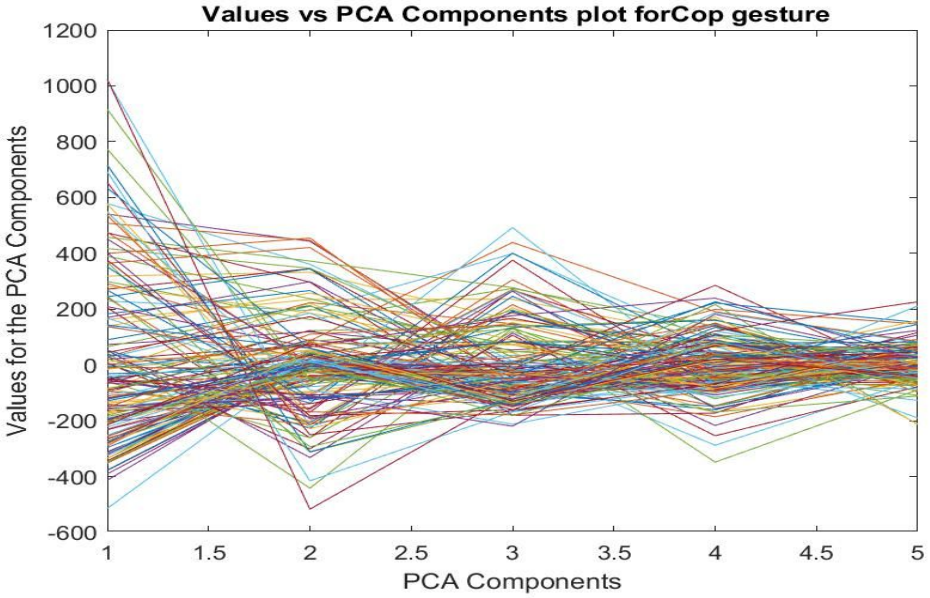
76.2210
7.7504
6.2137
4.8000
3.1083

Variance captured by top 5 Principal Components: 98.0934

Variance captured by top 3 Principal Components: 90.1851

Plot showing contribution of extracted features to first 3 Principal Components for Cop gesture:-





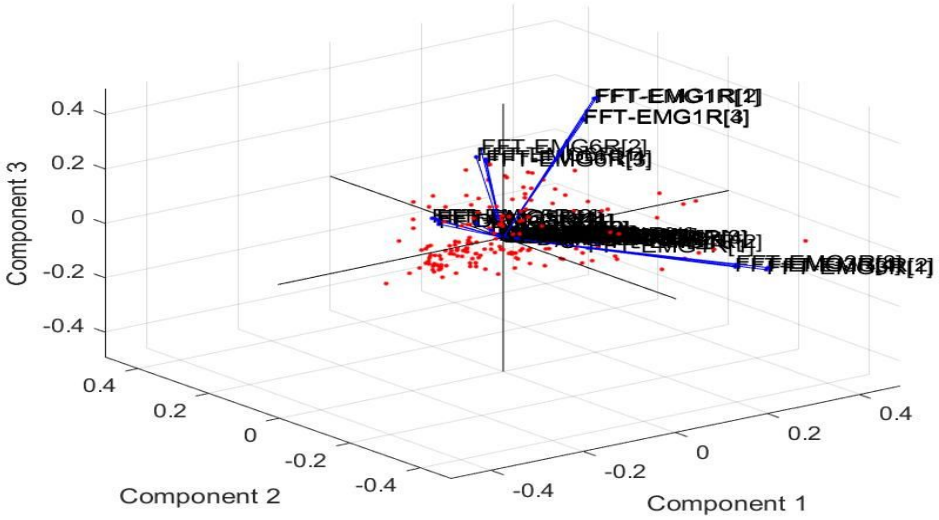
Percentage of variance covered by the top 5 principal components for **Cop:**

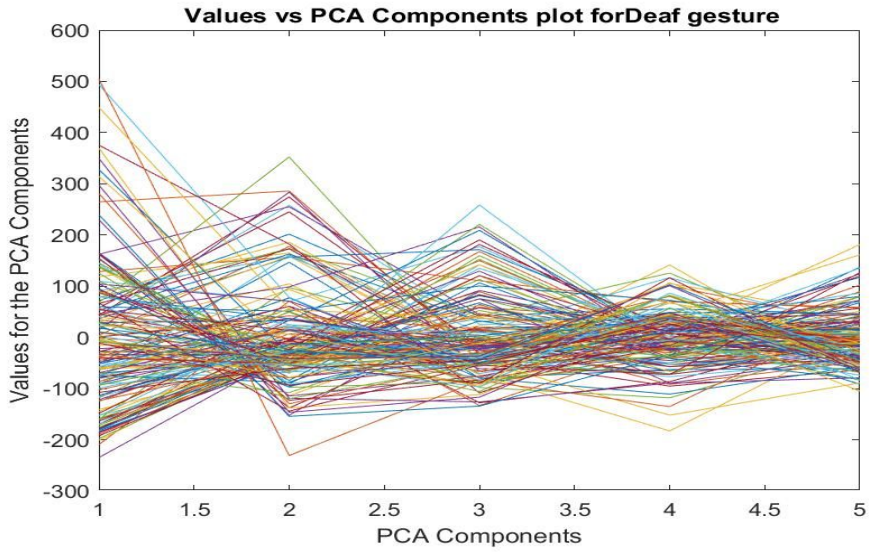
63.3344
16.9464
9.2220
5.6184
2.5548

Variance captured by top 5 Principal Components: 97.676

Variance captured by top 3 Principal Components: 89.5028

Plot showing contribution of extracted features to first 3 Principal Components for Deaf gesture:-



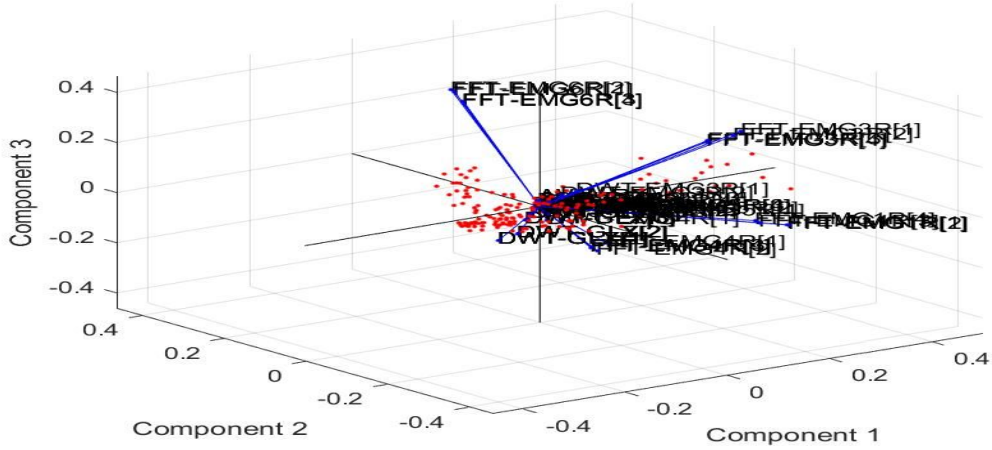


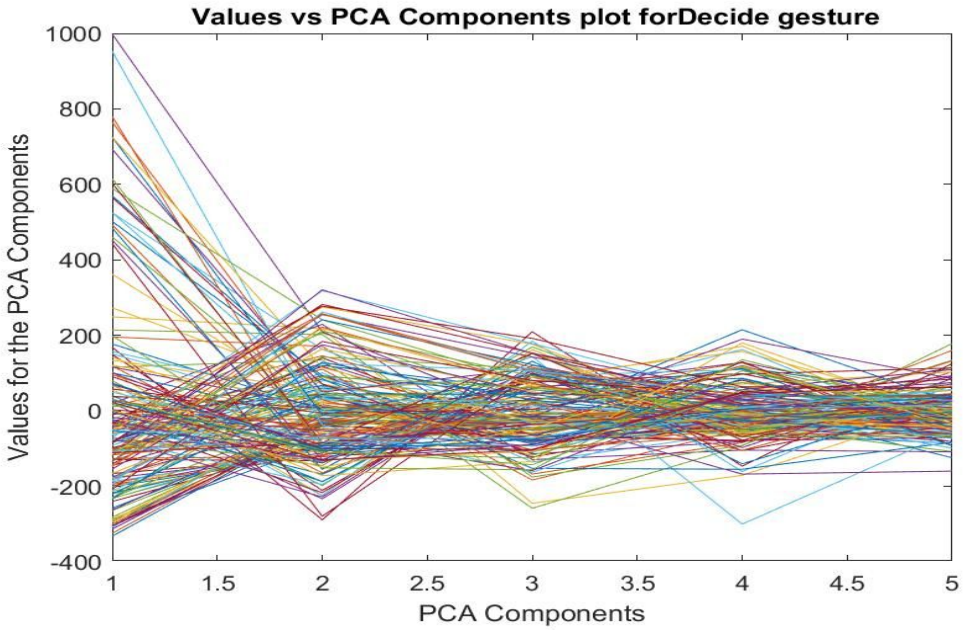
Percentage of variance covered by the top 5 principal components for **Deaf**:

61.9426
12.1330
7.9795
6.2260
5.0328

Variance captured by top 5 Principal Components: 93.3139
Variance captured by top 3 Principal Components: 82.0551

Plot showing contribution of extracted features to first 3 Principal Components for Decide gesture:-



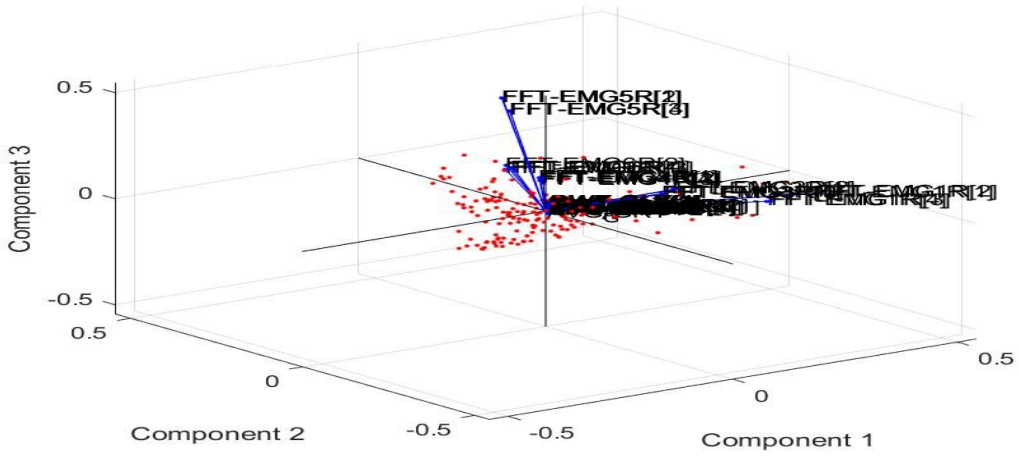


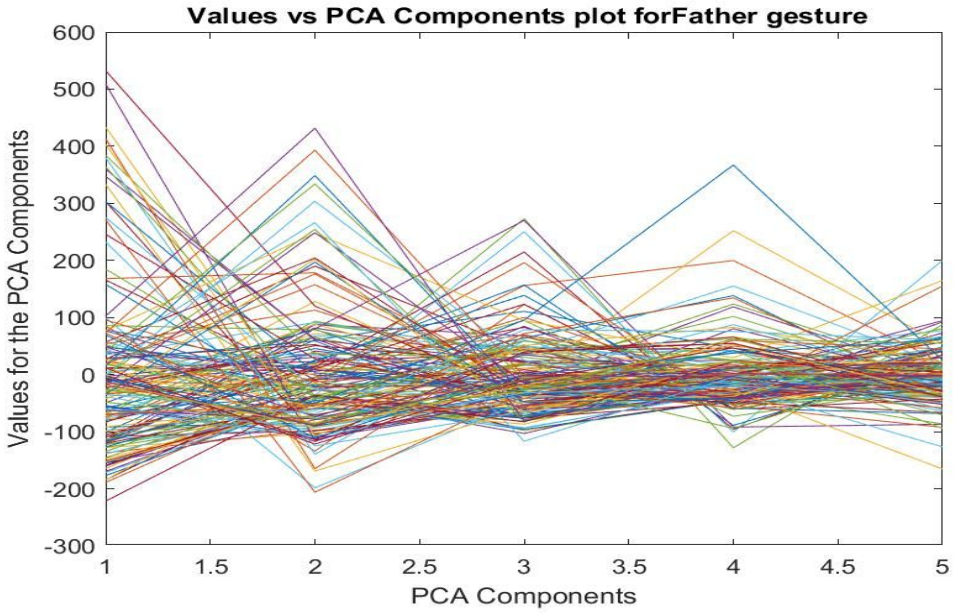
Percentage of variance covered by the top 5 principal components for **Decide**:

51.9537
25.1674
9.8074
5.8532
4.6322

Variance captured by top 5 Principal Components: 97.4139
Variance captured by top 3 Principal Components: 86.9285

Plot showing contribution of extracted features to first 3 Principal Components for Father gesture:-



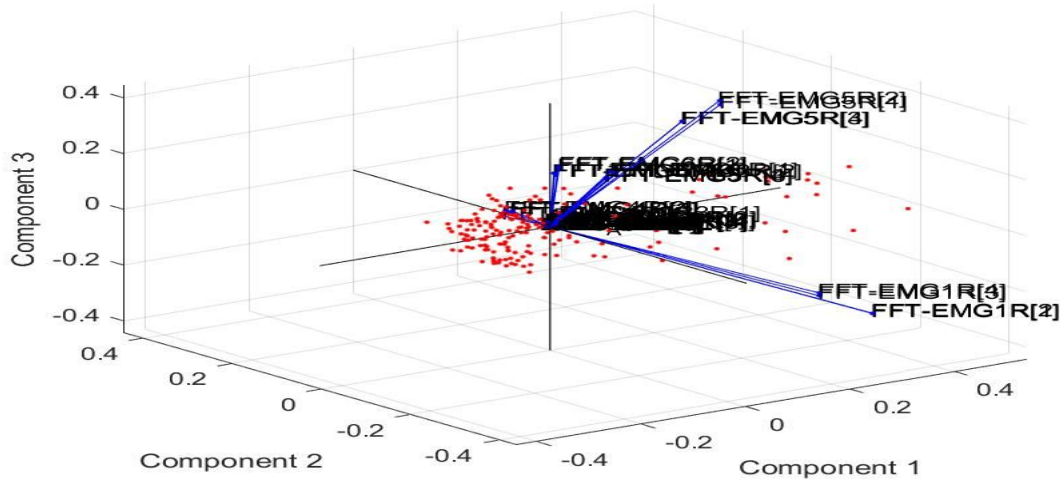


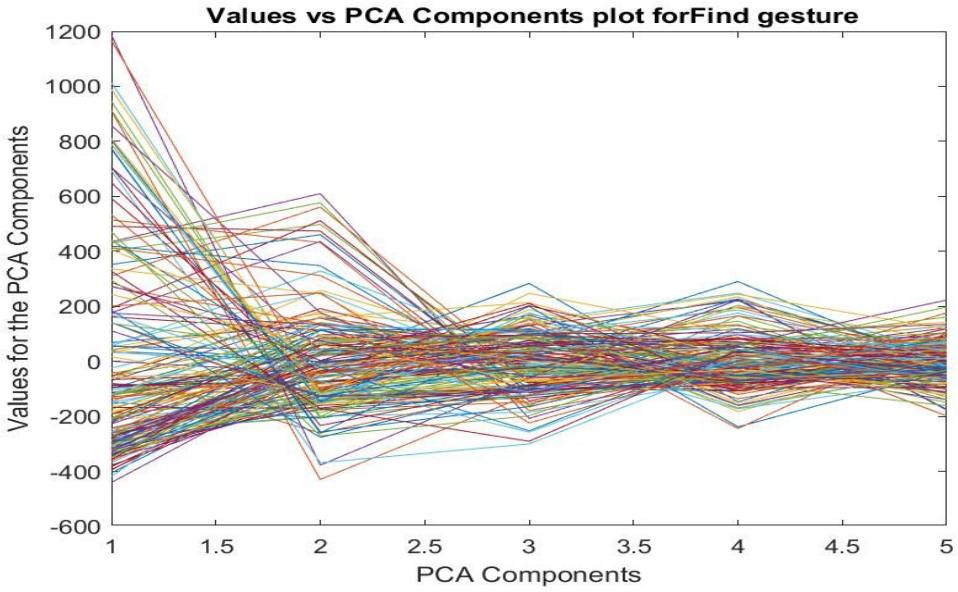
Percentage of variance covered by the top 5 principal components for **Father**:

70.7944
12.6136
6.7013
4.8683
2.6527

Variance captured by top 5 Principal Components: 97.6303
Variance captured by top 3 Principal Components: 90.1093

Plot showing contribution of extracted features to first 3 Principal Components for Find gesture:-



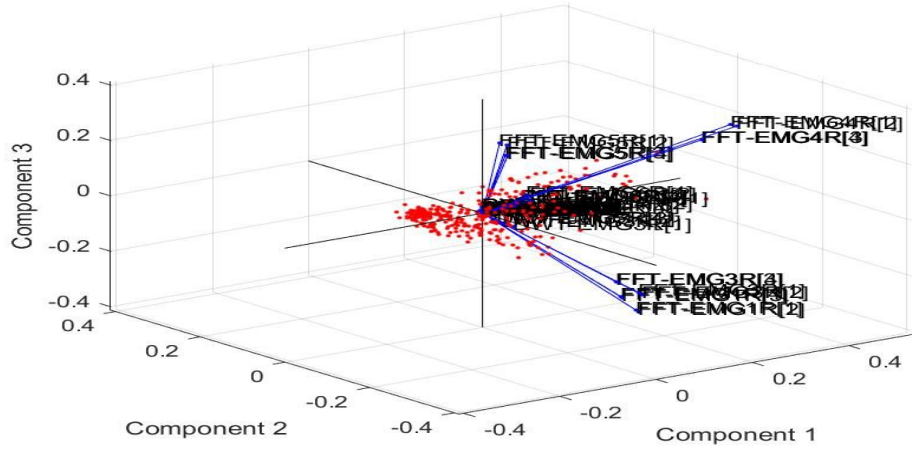


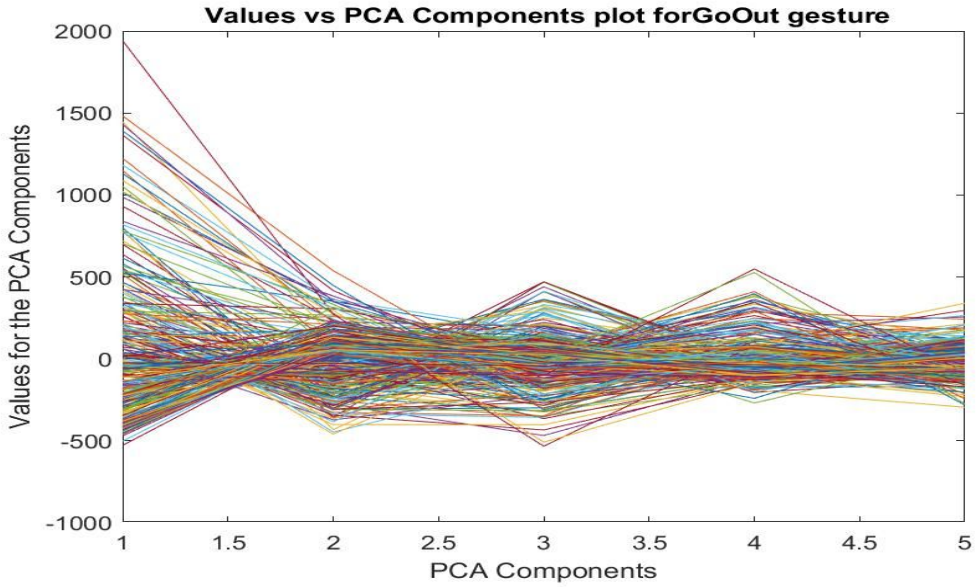
Percentage of variance covered by the top 5 principal components for **Find**:

75.3224
10.6001
5.9070
3.9990
2.4920

Variance captured by top 5 Principal Components: 98.3205
Variance captured by top 3 Principal Components: 91.8295

Plot showing contribution of extracted features to first 3 Principal Components for GoOut gesture:-



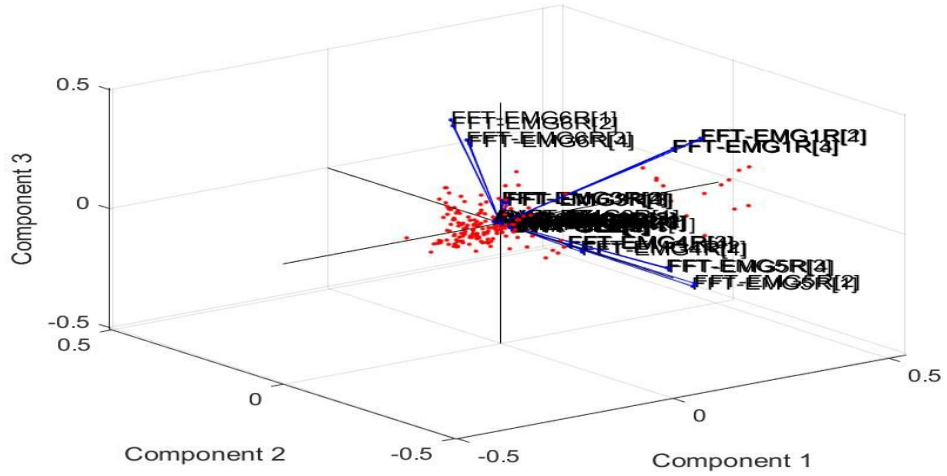


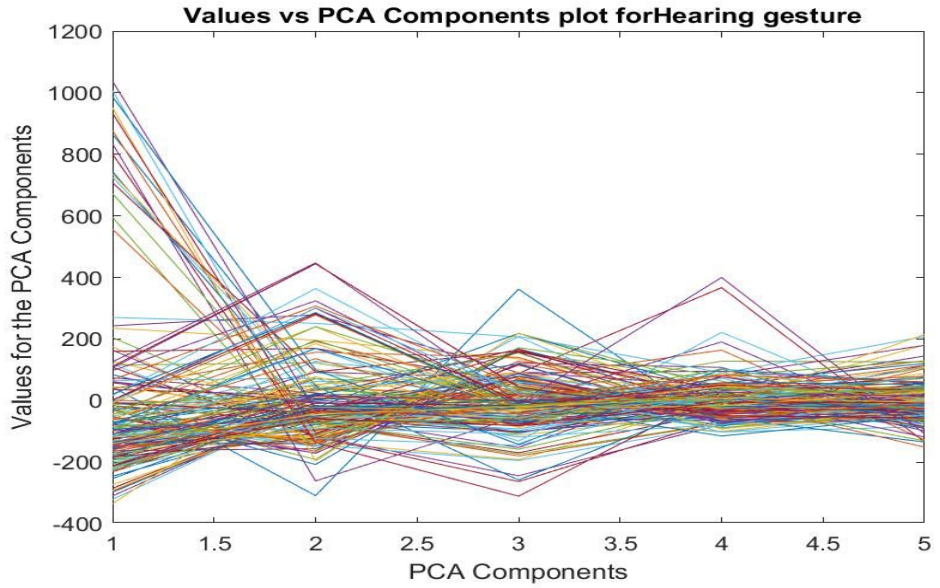
Percentage of variance covered by the top 5 principal components for **Go Out**:

60.1041
15.5923
10.6420
6.9168
4.0609

Variance captured by top 5 Principal Components: 97.3161
Variance captured by top 3 Principal Components: 86.3384

Plot showing contribution of extracted features to first 3 Principal Components for Hearing gesture:-





Percentage of variance covered by the top 5 principal components for **Hearing**:

75.3838
8.5997
4.7979
4.5693
3.9197

Variance captured by top 5 Principal Components: 97.2704

Variance captured by top 3 Principal Components: 88.7184

Average variance captured by top 5 Principal Components over all gestures: 96.6312
Average variance captured by top 3 Principal Components over all gestures: 87.9129

Conclusion of Principle Component Analysis (PCA)

Principal Component Analysis (PCA) was helpful in reducing the feature matrix from a *Nx55* to a *Nx5* matrix, where *N* is the number of actions per gesture, and still captured *96.63%* of the variance on average per gesture. This is important because features with high variance were required to distinguish between gestures. Since we can capture almost the entire variance with around 40% of the features chosen, PCA was helpful in reducing the number of features.