Sign Language Gesture Recognition

Author

Syed Hammad Nazeer Gilani

Regn Number

2010-NUST-MS-PHD-MTS-13

Supervisor

Dr. Umar Shahbaz Khan

DEPARTMENT OF MECHATRONICS ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

SEPT, 2014

Sign Language Gesture Recognition

Author

Syed Hammad Nazeer Gilani

Regn Number

2010-NUST-MS-PHD-MTS-13

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Mechatronics Engineering

Thesis Supervisor:

Dr. Umar Shahbaz Khan

Thesis Supervisor's Signature:____

DEPARTMENT OF MECHATRONICS ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY, ISLAMABAD

SEPT, 2014

Declaration

I certify that this research work titled "*Sign Language Gesture Recognition"* is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

Syed Hammad Nazeer Gilani

2010-NUST-MsPhD-Mts-13

Language Correctness Certificate

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.

Signature of Student

Syed Hammad Nazeer Gilani

2010-NUST-MsPhD-Mts-13

Signature of Supervisor

Dr. Umar Shahbaz Khan

ACKNOWLEDGEMENT

I would like to thank The Allah Almighty, who blessed me with hope and courage whenever I faced difficulties during my research work.

I am extremely grateful to Dr. Umar Shahbaz who allowed me to work under his supervision. He has been very helpful and supportive. He always encouraged me and boosted my moral whenever I was depressed. Without his guidance and persistent help, this dissertation would not have been possible.

I would like to thanks honorable faculty members who served on my committee: Dr. Kunwar Faraz, Dr. Umar Izhar and Dr. Adnan Masood for their support, guidance and useful suggestions.

I would like to thank my beloved parents and family, who prayed for me, their endless effort and support made it possible for me, to be, what I am today.

In the end I would like to thank the entire honorable faculty of Mechatronics Department of the College of Electrical and Mechanical Engineering to be so supportive.

Dedicated to my beloved & exceptional parents ***Mr. & Mrs. Syed Sajjad Hussain Gilani*** *and adored siblings whose tremendous support and cooperation led me to this wonderful accomplishment*

ABSTRACT

Recognition of Sign Language is one of the rising areas of research now days. Invention of first data glove made this field a topic of research. The basic concept of this project is to use computing technology to enable communication between two people who cannot converse directly. One person using sign language while the other unable to understand it needs something in between which can perform translation between the two different modes of communication they know. This project intends to assist the person using sign language in communicating with those who cannot understand sign language. The concept is to minimize this communication gap using a system which outputs the conversation as text & audio output.

With the advancement of science and technology many techniques have been developed not only to minimize the problem of deaf people but also to implement it in different fields. Many research works related to Sign languages have been done as for example the American Sign Language, the British Sign Language, Japanese Sign Language etc. It is the need of the time to introduce such a system that not only reduce the communication breach between deaf and normal community but also is a mean of bringing deaf community to the normal world.

The developed algorithm converts sign language signals collected by a 5 DT data glove and Nintendo power Glove data into text & audio output. The data has been processed by different classifiers and the results have been compared within the various classifiers and existing techniques to highlight the optimum methodology.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

SLR	Sign Language Recognition
LDA	Linear Discriminant Analysis
QDA	Quadratic Discriminant Analysis
DT	Decision Trees
NB	Naïve Bayes
GUI	Graphical User Interface
ZCR	Zero Crossing Rate

CHAPTER 1: INTRODUCTION

1.1 Introduction

Gesture is a practice of non-verbal communication using numerous body parts, commonly hand and face. Gesture is the firstborn method of communication in human. Primitive men used to communicate the information of food/ prey for hunting, source of water, information about their enemy, request for help etc. within themselves through gestures. Still gestures are used widely for different applications on different domains. This includes human-robot interaction, sign language recognition, interactive games, vision-based augmented reality etc. For communication by the people at a visible, but not audible distance (surveyors) and by the physically challenged people (mainly the deaf and dumb) gesture is the only method.

Posture is another term often confused with gesture. Posture refers to only a single image corresponding to a single command (such as stop), where as a sequence of postures is called gesture (such as move the screen to left or right). Sometimes they are also called static (posture) and dynamic gesture (gesture). Posture is simple and needs less computational power, but gesture (i.e. dynamic) is complex and suitable for real environments. Though sometimes face and other body parts are used along with single hand or double hands, hand gesture is most popular for different applications. A few of them are discussed below.

With the advancement of human civilization, the difficulty of interpersonal communication, not only in terms of language, but also in terms of communication between common people and hearing impaired people is gradually being abolished. If development of sign language is the first step, then development of hand recognition system using computer vision is the second step.

Many new techniques have been developed recently in these fields. Different approaches have been used by different researchers for recognition of various hand gestures which were implemented in different fields. Some of the approaches were vision based approaches, data glove based approaches, soft computing approaches like Artificial Neural Network, Fuzzy logic, Genetic Algorithm and others like PCA, Canonical Analysis, etc. The whole approaches could be divided into three broad categories- Hand segmentation approaches, Feature extraction approaches and Gesture recognition approaches.

1.2 Synopsis/ Thesis statement

Researchers have explored this field in the world but with the development of new classification techniques and methods, the scope of work is remained untouched in Pakistan. Basic idea is to apply new techniques and methods which are not applied yet on sign language glove data to classify the particular sign. After comparing the results of several techniques, suggest the best fit method on the basis of best classification accuracy. Aiding the hearing impaired community by producing text output and voice narration using a graphical user interface.

1.3 Rationale

The aim of this research is to develop a system for the automatic recognition of sign language, based on classification techniques. The research is motivated by two contrasting but complementary goals. The first is that a sign language system would be potentially helpful in assisting communication between associates of the Deaf community and the hearing community. The second is that the process of introducing such a system will explore this area of research in Pakistan which will practically help the Deaf-dumb community to teach, test and practice the sign gestures.

1.4 Objectives

The main objectives of this project is to develop a system that will be trained by supervised learning technique, resulting in recognizing a specific gesture of sign language which will depict a way of communicating between the deaf-dumb society with the normal ones. The system will display text output of the sign gesture and will also synthesize the output as voice narration. The main tasks are:

- Simple Data acquisition system/approach
- Extract and explore features of glove data for classification
- Apply several classifiers for classification and their comparison on extent of accuracy
- Text output of the gesture recognized
- Speech/voice output
- GUI design

1.5 Scope

This sole purpose of this project is to introduce such a system that will aid the deaf community to converse with the normal community. The scope of the project includes study of different classifiers, most effective features and producing text and voice output.

Scope of this project includes:

- Developing such a system that will recognize sign gesture
- Text output of the gesture recognized
- Synthesized speech output
- Graphical User interface to communicate with the system and to represent physical output of the system

1.6 Strategic plan

This project is divided into different sub-tasks including following steps with sequence

- Literature Review
- Sign gesture database finding
- Understanding the technique of data acquisition used to store the sign gesture
- Understanding the sign sensors data and their specification
- Pre-processing of the sensor data/ filtration for noise reduction
- Feature extraction
- Designing a methodology to recognize the signs
- Selection of classifiers to classify the sign gestures
- Training & testing of classifiers
- Improving the classification accuracy
- Generating text and voice as output of the system
- GUI design for the system

1.7 Research type/ Statement of research type

Sign language gesture recognition is a field of exploration since the invention of first data glove. Many researchers have designed successive methodologies to for sign recognition but nothing is perfect, some methodologies defects in accuracy, time consuming, non-efficient, limited signs. The key features to keep in mind during this project will be

- Explore new classification techniques
- More efficient in time
- More efficient in accuracy
- Text and speech output
- GUI design for the system

1.8 Methodology

After detailed study and research review methodology has been proposed to insure that it will results in best classification accuracy. The methodology used is shown below in figure

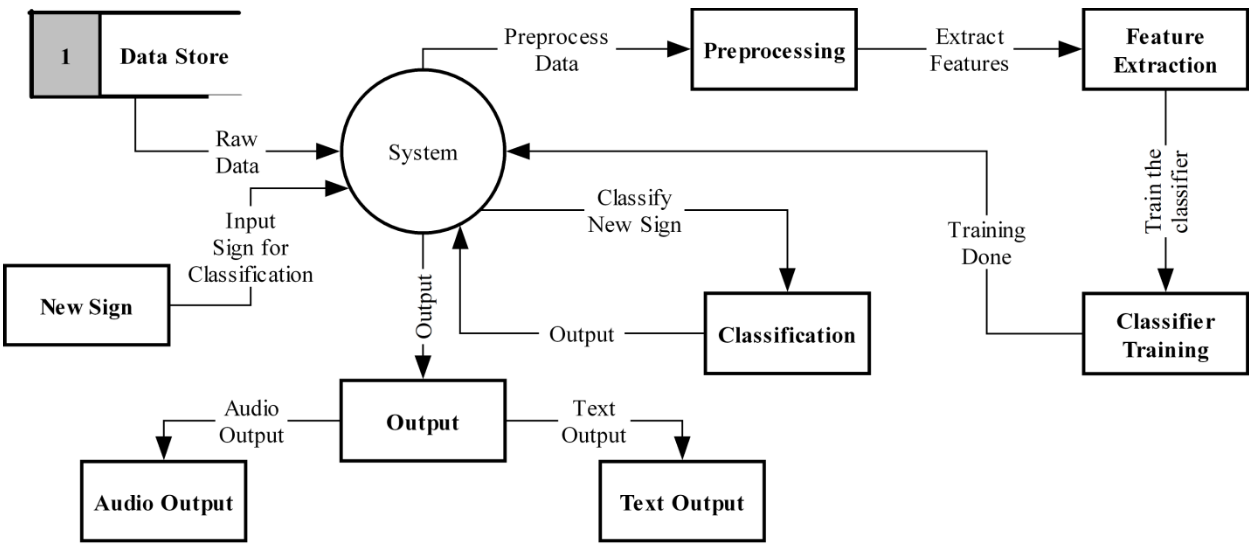


Figure 1 Design Flow Diagram

The system gets raw data from the repository. The system applies pre-processing steps to minimize the noise, spikes and other factors which affect the classification accuracy. After pre-processing it will extract features from the data and save them separately in other files with mapping of the sign with the features extracted. Several classification techniques i.e., classifiers applied on that featured data step by step. As we have applied supervised learning techniques so first classifier is trained and then tested on that data. We have applied 10 fold cross validation which means complete data is divided into 10 folds. 9 folds are used for training and 10^{th} fold is used for testing classifier. In this way testing and training data kept different.

1.8.1 Pre-Processing

Raw data acquired from the glove have some noise of spikes as it is gathered using Nintendo power glove which uses transmitter and receiver.

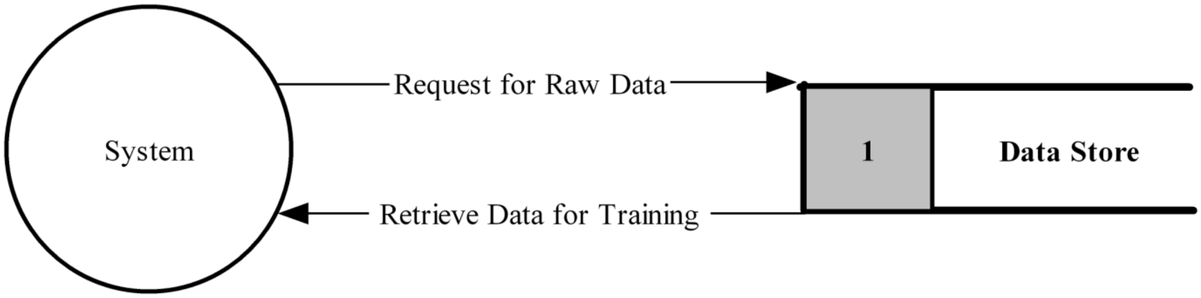


Figure 2 Pre-Processing of Data

1.8.1.1 Median filter

In signal processing, it is much desirable to apply noise reduction techniques on signal or an image. The 'median filter' is used to reduce noise and it is categorized as a nonlinear digital filtering technique. Noise reduction is a pre-processing step to enhance the results of processing applied.

Median Filter is applied on the data to remove the occasional spikes from the sign gesture data. It is applied individually on each file/sign data.

1.8.2 Feature Extraction

1.8.2.1 Mean

It is the arithmetic mean, and is computed by summing all data numbers, then dividing by the total count of all data numbers. The mean of an example x1, x2……., xn is the addition of the sampled numbers divided by the total number of data numbers in the sample

$$x = \frac{x_1 + x_2 + \cdots + x_n}{n} \qquad \text{………………………………… (1)}$$

1.8.2.2 Variance

In statistics, variance measures how far a set of numbers is spread out. A variance of zero indicates that all the values are identical. Variance is always non-negative: a small variance indicates that the data tend to be very close to the mean (expected value) and hence to each other, while a high variance indicates that the data are very spread out around the mean and from each other.

$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i \quad \text{………………………………… (2)}$$

1.8.2.3 Standard Deviation

In statistics and probability theory, the standard deviation (SD) (represented by the Greek letter sigma, σ) measures the amount of variation or dispersion from the average. A low standard deviation indicates that the data points tend to be very close to the mean (also called expected value); a high standard deviation indicates that the data points are spread out over a large range of values.

$$\sigma = \sqrt{(\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2)} \quad \text{.....................................} \; (3)$$

The standard deviation of a random variable, statistical population, data set, or probability distribution is the square root of its variance.

1.8.2.4 Distance

Each file contains sequence of frames, each frame providing the information of x, y, z position

Let x_i y_i z_i the x, y and z position stored in the *ith* frame of the data of sample. Change can be defined as

$$\Delta x_i = x_i - x_{i-1} \quad \text{.....................................} \; (4)$$

$$\Delta y_i = y_i - y_{i-1} \quad \text{.....................................} \; (5)$$

$$\Delta z_i = z_i - z_{i-1} \quad \text{.....................................} \; (6)$$

The vector (Δx_i Δy_i Δz_i) shows the direction sign is converging to. if length of this vector is calculated and called as Δi

$$\Delta_i = \sqrt{(\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2)} \quad \text{.....................................} \; (7)$$

Then it will provide us amount of speed as it is the first derivative with respect to time.

$$\Delta^2 x_i = \Delta x_i - \Delta x_{i-1} \quad \text{.....................................} \; (8)$$

$$\Delta^2 y_i = \Delta y_i - \Delta y_{i-1} \quad \text{.....................................} \; (9)$$

$$\Delta^2 z_i = \Delta z_i - \Delta z_{i-1} \quad \text{.....................................} \; (10)$$

$$\Delta_i^2 = \sqrt{(\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2)} \quad \text{.....................................} \; (11)$$

Again, the vector ($\Delta^2 x_i$ $\Delta^2 y_i$ $\Delta^2 z_i$) characterizes the alteration in direction that has followed Δ^2_i is likewise the length & norm of that vector. The total addition Δi will provide the total distance travelled by the sign

$$distance = \sum_{i=1}^{n} \Delta i \text{ .... (12)}$$

Where 'n' is the amount of frames in the sign.

1.8.2.5 Energy

The exact estimation of 'energy' cannot be measured because of involvement of so many variables and factors. Some assumption are made on the basis of approximation

- The sensor transmits data packets at identical time intervals. Therefore the time is constant in between two consecutive frames
- Positional motion provides the maximum of the energy.
- The effective mass of hand is constant and same amount energy is required to move hand in all directions

Using above mentioned assumptions extent of energy can be derived as.

$$W = \sum_{i=0}^{n} F\Delta si \text{ .... (13)}$$

As F=ma and 'm' is a constant

$$W \propto \sum_{i=0}^{n} a\Delta i \text{ .... (14)}$$

But $a = \frac{\Delta v}{\Delta t}$ and Δt is also constant and Δv is identical with $\Delta^2{}_i$

After applying we get

$$energy = \sum_{i=2}^{n} \Delta^2 \mathrm{i}\, \Delta \mathrm{i} \text{ .... (15)}$$

1.8.2.6 No. Of Frames/Maneuverability of gesture

Every sign has different no of frames depend upon the length of gesture or maneuverability of the gesture. It is wise to think total number of frames/lines/rows in file as a feature.

1.8.2.7 Bounding Box

Several signs are larger than the rest and that they diverge in their location with respect to the body. It is an understandable thing to consider as feature is the bounding box or range of sign.

As this is shown in the figure3, six features are vital to illustrate the bounding box or range of the sign.

- Xmin
- Xmax
- Ymin

- Ymax
- Zmin
- Zmax

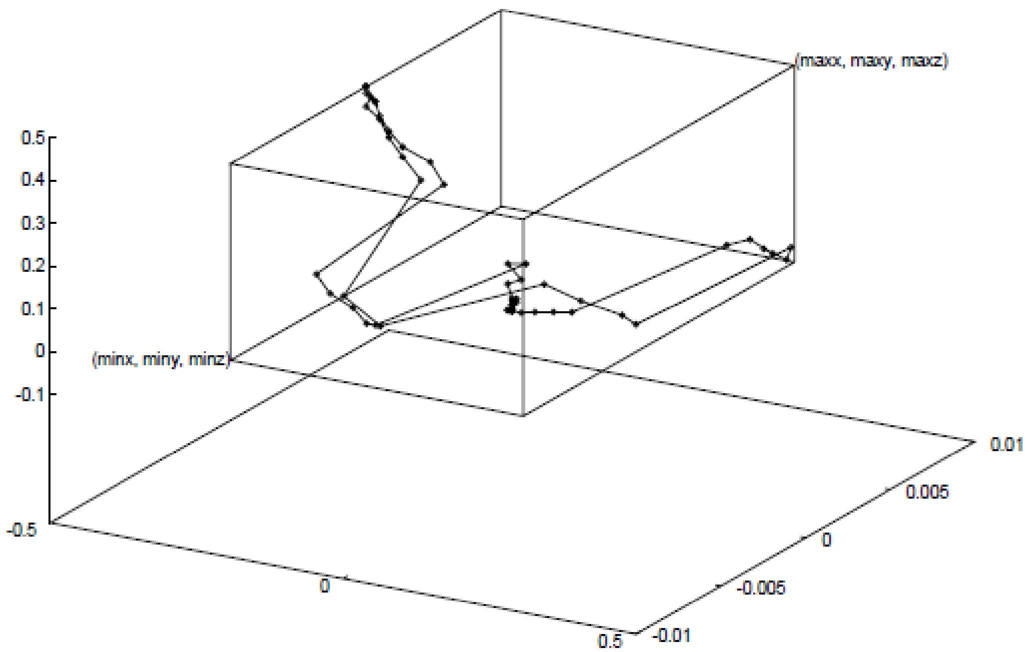


Figure 3 Bounding Box diagram

1.8.2.8 Zero Crossing rate

A zero-crossing is a point where the sign of a mathematical function changes (e.g. from positive to negative), represented by a crossing of the axis (zero value) in the graph of the function. It is a commonly used term in electronics, mathematics and signal processing.

The zero-crossing rate is the rate of sign-changes along a signal, i.e., the rate at which the signal changes from positive to negative or back. This feature has been used heavily for classification and recognition.

ZCR is defined formally as

$$zcr = \frac{1}{T-1}\sum_{t=1}^{T-1} \mathbb{I}\left\{s_t s_{t-1} < 0\right\}$$

Zero crossing tells us that how many times '0' has been crossed by sensor during performing particular sign

1.8.3 Classification

The main purpose of classification is to recognize one's aims based on some informative characteristics in any system. In order to attain high level of accuracy in classification, supervised learning algorithms are employed. Classification algorithms/methods can be employed for online and/or offline learning sessions. Online classification is far difficult than offline classification. Because of having no time limit, data analysis can be revised multiple times in offline sessions. For the classification of sign language, various efforts have been done; some of the important classifiers are described as under.

1.8.3.1 Linear Discriminant Analysis (LDA)

Without involving high calculations, this method brings significant level of precision in classification. It provides extreme level of class separation. One of its principal characteristics is that the position of data remains unchanged. A linear decision area is sketched between the signs by LDA.

A linear discrimination function is expressed by LDA and for discrimination between the classes; a hyper plane is represented in feature space. The side of the hyper plane, where the vector is found, is real measure of the decision of class to which the characteristic vector belongs. More than one hyper planes, all are linear, will be sketched for more than two classes (N>2). The mathematical expression of the plane can be:

$$g(x) = w^T + w_0 \qquad \text{.................. (17)}$$

Where, 'x' represents the characteristic vector (which is to classify), 'w' represents a weight vector & w_0 represents threshold. If the data is overlying between the classes then, its performance will be reduced because quadratic decision plane does not define LDA.

1.8.3.2 Quadratic Discriminant Analysis (QDA)

This is one more good method and is closely associated with LDA. It discriminates the data by sketching quadratic decision line. It is employed for numerous diverse sets of data classification analysis.

1.8.3.3 Naïve Bayes

This method is a very widespread classifier and is used for several classification difficulties. It is based on called Bayesian Theorem and is suitable in a case having high dimensionality. The first significant parameter of this method is the information, which will be based on the previous experience, regarding the Prior Probabilities. The subsequent guesstimate is the Likelihood of the class. The concluding classification is made by the addition of both sources of information, i.e., the prior and the likelihood, to form a posterior probability.

1.8.3.4 Decision Trees (DT)

This ethos is exercised for the prediction of the class by recursive partition of the instance space. It signifies rules. It is the tempting point of DT. It has a tree like structure and has nodes, whether it is leaf or decision node. It is a dominant tool of classifying a sample by opening at the root of the tree and moving through it until a leaf node, which delivers the classification of the instance. Classification Tree algorithm is applied in the work done. The tree begins with complete input data, and inspects all possible binary splits on every predictor. The split with best optimization criterion is nominated. The optimization criterion for Tree is Gini's diversity index (gdi). The Gini index of node,

$$gdi = 1 - \sum_i P^2(i) \quad \text{..................} \ (18)$$

Where sum is over the classes i at the node and P(i) is the observed fraction of classes with class i that reach the node.

For classification tree stops partition of the instance space when it will reach to the pure node; a node is pure if it contains observation of one class.

1.8.4 *K* fold classification

In k-fold cross-validation, the complete data is randomly divided into k equal size subsamples. Of the k subsamples, a single subsample is kept as the validation data for analysis of the model, and the remaining k − 1 sub samples are used as training data. The cross-validation process is then iterated k times (the folds), with each of the k subsamples used exactly once as the validation data. The results from k folds can then be averaged to estimate a single result. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. Commonly 10-fold cross-validation is implemented.

1.9 Resource/ Source of data

The sign gesture repository is used to save raw data. Our aim is to find online repository of sign gestures. The sign gesture should in single file, stored with the sensor data sequentially and frame by frame. We have found two repositories of AuSLAN sign gestures using Nintendo Power Glove and5DT power glove. Single file represents the singe gesture. Same methodology is applied to both the repositories and results have been achieved.

1.9.1 Nintendo Power Glove data

The source of the data is the raw measurements from a Nintendo Power Glove. It was interfaced through a Power Glove Serial Interface to a Silicon Graphics 4D/35G workstation.

1.9.2 5-DT data glove data

Data was captured & recorded using a setup that contained:

- Two Fifth Dimension Technologies (5DT) gloves, one for right hand and one for left hand
- Two Ascension Flock-of-Birds magnetic position trackers, one for each hand
- A four-port serial card to cope with four data sources
- A PC (128MB RAM, Intel Pentium II 266MHz) was used

CHAPTER 2: RESEARCH METHODOLOGY

2.1 Research methodology

Many investigators have premeditated succeeding approaches for sign recognition but limited perfection of accuracy, some methodologies with deficiencies of classification accuracy, time consuming, and used limited sign gestures. Many algorithms like ANNs, HMMs, LVQ, Decision Trees, IBL1, and IBL2 have been used.

In Pakistan this field is un-explored and excessive depth is available to develop some methodology which needs remarkable effort to produce maximum results. A new methodology is devised to attain more accurate recognition results, less computational time & power, include maximum number of sign gestures and produce text & voice output. Methodology introduced has some basic steps which defines and assist the purpose of thesis.

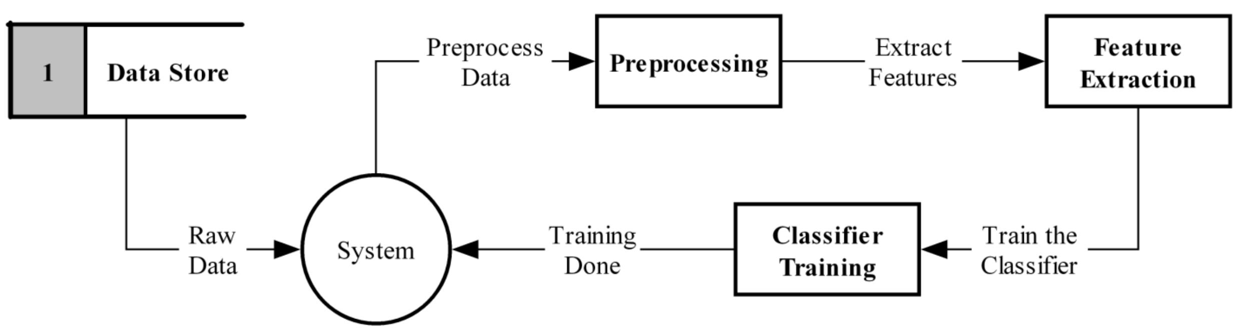


Figure 4 Research Methodology

2.1.1 Pre-Processing Raw data

Pre-processing is required to minimize the noise in data in order to improve the classification results and reduce the factors affecting the end result. The data from Nintendo power glove have some occasional spikes which affects the feature extraction and classification process. Median filter is used and applied to remove this noise and seems to produce better results. This method of applying median filter to raw data improves the working (reduce time of processing) and quality results of the project.

2.1.2 Feature Extraction

Feature extraction is a vital step of this project as classifier applied needs some factors,, based on which classifier will deduce the individual sign to classify for a particular class. Many features have been extracted, saved and tested in this project; mean, variance, standard deviation, bounding box, zero crossing rate, instant count, distance travelled by sign, energy of sign, no of frames of sign are amongst the good features used. Some of the features results in high accuracy and some with low accuracy. This step of the project participated with a vital role to classify with the best accuracy.

2.1.3 Training and Testing Classifiers

Supervised learning technique is used to train, test and validate the classifier for better results. In order to keep training and testing data different *k-fold cross validation* technique is used. In this technique complete data samples are randomly divided into 10-folds, out of which 9 are used for training the classifier and one last fold is used for testing. In this way we get the accuracy results based on true facts and near to real time processing environment.

2.1.4 Text & Voice Output

This is step is the output of this project and it is of key importance for the deaf-dumb person using this technique to communicate with the normal community. It is need of the day to produce text as well as voice output for better communication and ease of use. A GUI interface is developed to produce text output on screen and a speaker is used for voice output. The classifier is mapped with the end result sign gesture with respect to the class it is classified. A new sign gesture is given as input to the system and system will first pre-process that sign data, extract features and these features then pushed to the classifier for identification of the sign gesture. As soon as it gets recognized by the classifier it results the class to the system. System will then produce the text as well as voice output to the end user.

This process of producing text and voice is highly important for this project and the research methodology used benefited the project with great results.

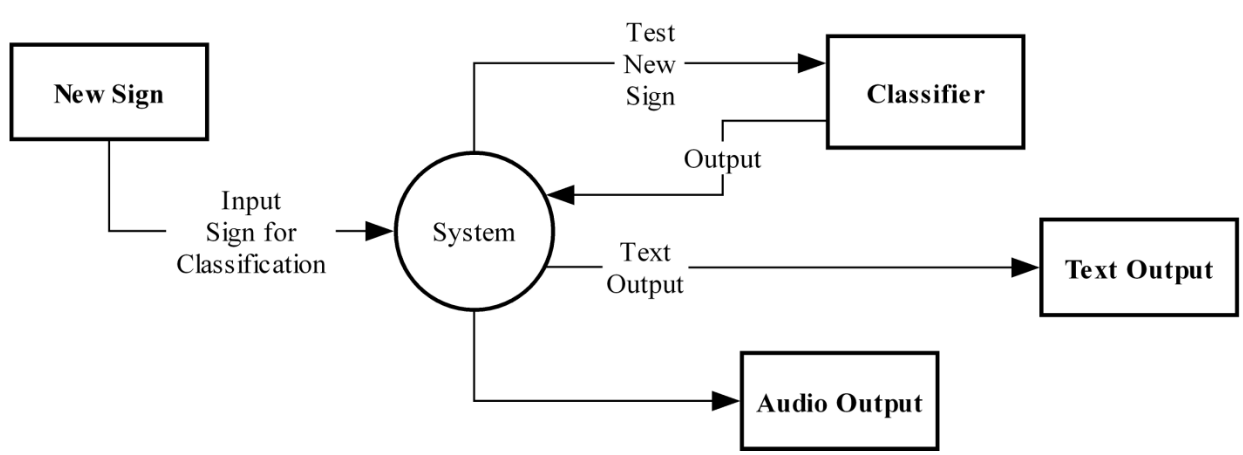


Figure 5 New Sign Recognition System

2.1.5 Graphical User Interface

It is the need of the system to have an interface that is a mean of correspondence between the system and user. This GUI will show the physical output of the system. The text output and the processing status are shown in the GUI. It also has a 'RECOGNIZE' button that will initiate the recognizing system to classify and recognize the gesture. The GUI is shown in the fig below

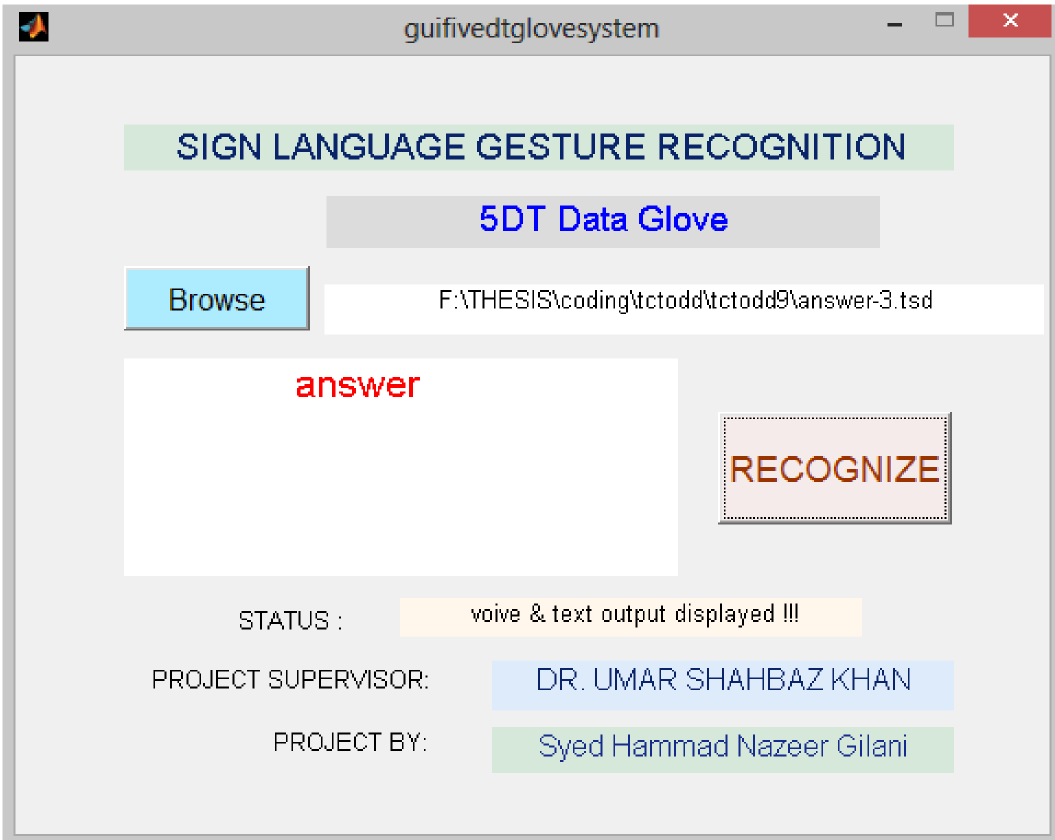


Figure 6 Grpahical user interface of the system

How is it beneficial?

Each and every step of methodology benefited the project with good results. Pre-processing step reduce the effect of noise resulting in pushing for high accuracy. Feature extraction produced the best features for high classification accuracy. Classifier applied analyzed the best classifier for this type of data to be classified with minimum of processing time and high accuracy in results. Text and voice output aided the end user to communicate with the environment and restore the deficiency of speech and hearing. In brief context research methodology used proved to be highly beneficial for the project.

2.2 Source of Data: sample selection methodology

The sign gesture repository is used to save raw data. Our aim is to find online repository of sign gestures. The sign gesture should in single file, stored with the sensor data sequentially and frame by frame. We have found two repositories of AuSLAN sign gestures using Nintendo Power Glove and5DT power glove. Single file represents the singe gesture. Same methodology is applied to both the repositories and results have been achieved.

2.2.1 Nintendo Power Glove data

The source of the data is the raw measurements from a Nintendo Power Glove. It was interfaced through a Power Glove Serial Interface to a Silicon Graphics 4D/35G workstation.

2.2.1.1 Data Acquisition

The Nintendo power glove can be categorized as "cheap and nasty". Information regarding position is measured by the emission of ultrasound from emitters; the glove; to a 3-microphone "L-Bar" receiver. Glove consists of two; and three receivers. This permits the acquisition of 4 attributes of information: left & right that is x, up & down that is y, backward & forward that is z and palm pointing direction that is roll. x, y and z have 8 bit accuracy. To be precise, 1 unit movement in the any direction is not equal to the same distance to 1 unit in the other directions. These x, y, z positions represents the position with respect to the default position of calibration point that is resting position of the palm on the thigh of signer is seated position. Roll has 4 bit accuracy.

Figure 7 Nintendo power glove

The data has some noise of infrequent "spikes" produced by random ultrasound noise. This type of noise can be removed by median filters, which proved to be very helpful.

Finger bend information is produced by bend sensors placed on the fingers of glove. The sensor values vary from 0 means straight to 3 means fully bent. It has accuracy of 2 bits. The gloves apply a hysteresis filter automatically on bend sensors.

Five signers are requested to perform 95 signs, to create a database based on each signer:

Table 1 Nintendo Power Glove Signer Information

Signer	Description	Sessions	No. of Sign Gestures	Total samples/sign
Signer1	Sign interpreter - PhD finished	2	95	8
Signer2	Natural signer	2	95	8
Signer3	Professional Auslan linguist	2	95	8
Signer4	Professional Auslan translator	4	95	8
Signer5	Novice signer	4	95	8

Sessions for each signer was performed at different time intervals.

The "signer1" datasets has some fatigue effects because it is sampled in a fixed order. Datasets of other signers were sampled randomly. The "signer5" & "signer4" datasets were performed with calibration and begins with "cal-". These were considered as a means of calibration, but didn't work out too well.

Average no. of frames contained per sign is 51, but it varies 30 – 102 frames.

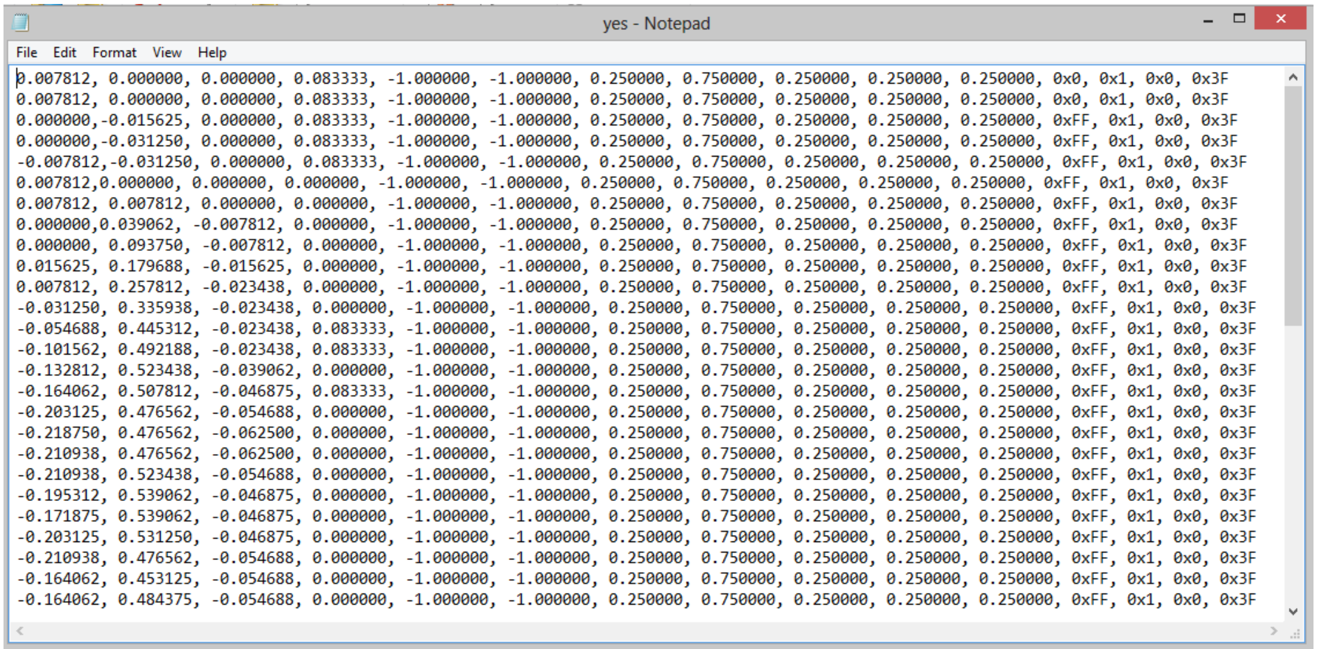
File contains comma separated values of the attributes mentioned below. Each sign is stored separately in single file. The directory order is as follows:

- Each signer signs are in a different directory.
- Each session of signer is stored as a subdirectory. Each session is symbolized by a number.
- Each sample is named on the sign gesture appended by the number of the sample of sign.

Table 2 Attribute Information Nintendo power glove data

X	Continuous	X position valued from -1 to 1. Units are meters
Y	Continuous	Y position valued from -1 to 1. Units are meters
Z	Continuous	Z position valued from -1 to 1. Units are meters
Roll	Continuous	Roll from 0 showing "palm down", rotating clockwise to 1, it is "palm down" as well.
Pitch	-	Should be ignored
Yaw	-	Should be ignored
thumb	Continuous	Thumb bend. valued 0 means straight to 1 means fully bent
Fore	Continuous	Fore finger bend. valued 0 means straight to 1 means fully bent
Index	Continuous	Index finger bend. valued 0 means straight to 1 means fully bent
Ring	Continuous	Ring finger bend. valued 0 means straight to 1 means fully bent
Little	Continuous	Little finger bend. valued 0 means straight to 1 means fully bent
Key code	-	Should be ignored
Gs1	-	Should be ignored
Gs2	-	Should be ignored
Receiver values	-	Should be ignored

The data in the file is shown in the fig below


yes - Notepad
File Edit Format View Help

```
0.007812, 0.000000, 0.000000, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0x0, 0x1, 0x0, 0x3F
0.007812, 0.000000, 0.000000, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0x0, 0x1, 0x0, 0x3F
0.000000,-0.015625, 0.000000, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.000000,-0.031250, 0.000000, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.007812,-0.031250, 0.000000, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.007812,0.000000, 0.000000, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.007812, 0.007812, 0.000000, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.000000,0.039062, -0.007812, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.000000, 0.093750, -0.007812, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.015625, 0.179688, -0.015625, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
0.007812, 0.257812, -0.023438, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.031250, 0.335938, -0.023438, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.054688, 0.445312, -0.023438, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.101562, 0.492188, -0.023438, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.132812, 0.523438, -0.039062, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.164062, 0.507812, -0.046875, 0.083333, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.203125, 0.476562, -0.054688, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.218750, 0.476562, -0.062500, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.210938, 0.476562, -0.062500, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.210938, 0.523438, -0.054688, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.195312, 0.539062, -0.046875, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.171875, 0.539062, -0.046875, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.203125, 0.531250, -0.046875, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.210938, 0.476562, -0.054688, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.164062, 0.453125, -0.054688, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
-0.164062, 0.484375, -0.054688, 0.000000, -1.000000, -1.000000, 0.250000, 0.750000, 0.250000, 0.250000, 0.250000, 0xFF, 0x1, 0x0, 0x3F
```

Figure 8 Sensor output of the Nintendo power glove gesture

2.2.2 5-DT data glove data

Data was taken by means of a setup that contains:

- Two 5DT data gloves of Fifth Dimension Technology, one for right hand & one for left hand
- Two magnetic position trackers of Ascension Flock-of-Birds, one with each hand

Figure 9 5DT Data Glove

2.2.2.1 Data Acquisition

The Flock system is much better than Nintendo system as compared on the basis of quality of the sensor data. As it is two-handed data retrieval system. Moreover, it provides six degrees of freedom (DOF) due to position tracker attached i.e. roll, pitch, yaw and x, y, z. The gloves provide reliable data of all five fingers. The large improvement is in resolution of sensor data both in accuracy & temporal. 14 bit accuracy for position and orientation, providing information with positional error less than one cm, error less than one half of a degree for angle. 8 bits accuracy is measured for finger bend. The refresh rate of whole system is close to 100 frames per second with significantly less noise than the Nintendo system for all signals.

Samples from a single Auslan signer are collected in period of 9 weeks. Total of 27 samples per gesture, total of 2565 sign samples are collected. The average length of all sign gestures is approximately 57 frames per sign.

The data is the raw sign data with no filters applied.

Table 3 5DTGlove Signer Information

Signer	**Description**	**Sessions**	**Total samples/sign**	**No. of sign gestures**
Signer1	Auslan sign interpreter	9	27	95

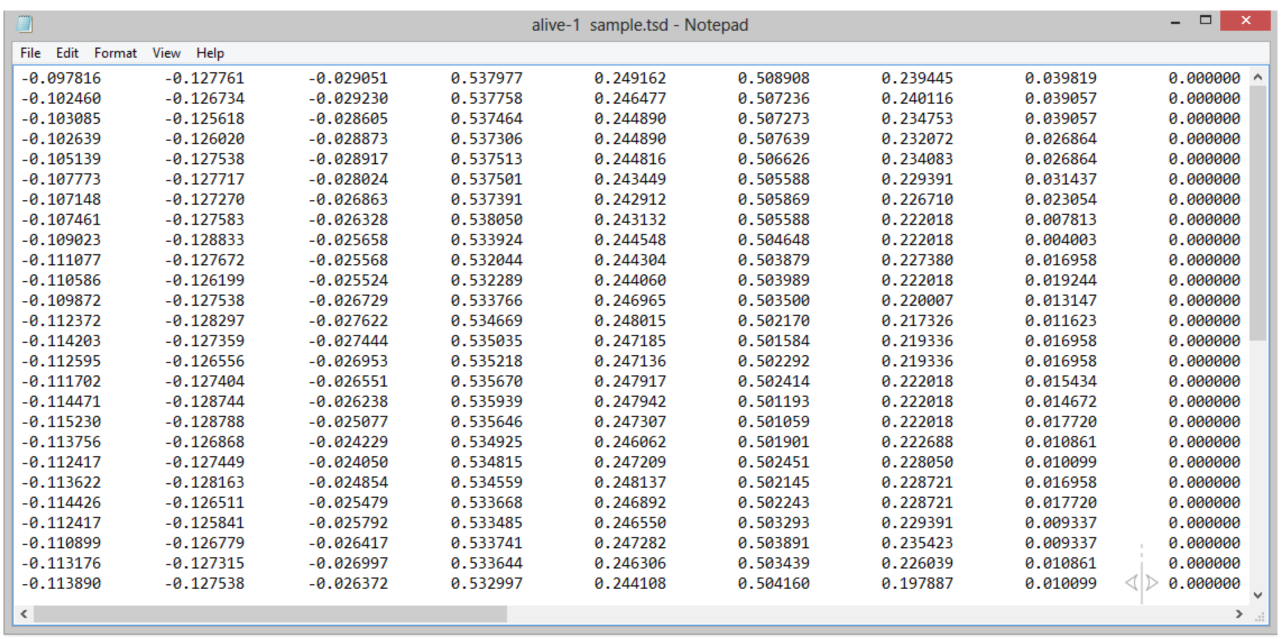
Each file consists of a sequence of lines. Each line consists of 22 whitespace-separated numbers representing the 22 channels of information.

Table 4 Attribute Information 5-DT power glove data

X	Continuous	X position articulated relative to default zero point set considerably below the chin
Y	Continuous	Y position articulated relative to default zero point set considerably below the chin
Z	Continuous	Z position articulated relative to default zero point set considerably below the chin
Roll	Continuous	Roll has as a value varied from -0.5 and 0.5. 0 means palm is down.
Pitch	-	Pitch has as a value varied from -0.5 and

		0.5. 0 means palm is down.
Yaw	-	Yaw has as a value varied from -0.5 and 0.5. 0 means palm is down.
thumb	Continuous	Thumb bends measures from 0 to 1. 0 shows totally flat, 1 shows totally bent
Fore	Continuous	Fore bends measures from 0 to 1. 0 shows totally flat, 1 shows totally bent
Index	Continuous	Index bends measures from 0 to 1. 0 shows totally flat, 1 shows totally bent
Ring	Continuous	Ring bends measures from 0 to 1. 0 shows totally flat, 1 shows totally bent
Little	Continuous	Little bends measures from 0 to 1. 0 shows totally flat, 1 shows totally bent

The sensor data of 5DT data glove gesture is shown below


alive-1 sample.tsd - Notepad

File Edit Format View Help

-0.097816	-0.127761	-0.029051	0.537977	0.249162	0.508908	0.239445	0.039819	0.000000
-0.102460	-0.126734	-0.029230	0.537758	0.246477	0.507236	0.240116	0.039057	0.000000
-0.103085	-0.125618	-0.028605	0.537464	0.244890	0.507273	0.234753	0.039057	0.000000
-0.102639	-0.126020	-0.028873	0.537306	0.244890	0.507639	0.232072	0.026864	0.000000
-0.105139	-0.127538	-0.028917	0.537513	0.244816	0.506626	0.234083	0.026864	0.000000
-0.107773	-0.127717	-0.028024	0.537501	0.243449	0.505588	0.229391	0.031437	0.000000
-0.107148	-0.127270	-0.026863	0.537391	0.242912	0.505869	0.226710	0.023054	0.000000
-0.107461	-0.127583	-0.026328	0.538050	0.243132	0.505588	0.222018	0.007813	0.000000
-0.109023	-0.128833	-0.025658	0.533924	0.244548	0.504648	0.222018	0.004003	0.000000
-0.111077	-0.127672	-0.025568	0.532044	0.244304	0.503879	0.227380	0.016958	0.000000
-0.110586	-0.126199	-0.025524	0.532289	0.244060	0.503989	0.222018	0.019244	0.000000
-0.109872	-0.127538	-0.026729	0.533766	0.246965	0.503500	0.220007	0.013147	0.000000
-0.112372	-0.128297	-0.027622	0.534669	0.248015	0.502170	0.217326	0.011623	0.000000
-0.114203	-0.127359	-0.027444	0.535035	0.247185	0.501584	0.219336	0.016958	0.000000
-0.112595	-0.126556	-0.026953	0.535218	0.247136	0.502292	0.219336	0.016958	0.000000
-0.111702	-0.127404	-0.026551	0.535670	0.247917	0.502414	0.222018	0.015434	0.000000
-0.114471	-0.128744	-0.026238	0.535939	0.247942	0.501193	0.222018	0.014672	0.000000
-0.115230	-0.128788	-0.025077	0.535646	0.247307	0.501059	0.222018	0.017720	0.000000
-0.113756	-0.126868	-0.024229	0.534925	0.246062	0.501901	0.222688	0.010861	0.000000
-0.112417	-0.127449	-0.024050	0.534815	0.247209	0.502451	0.228050	0.010099	0.000000
-0.113622	-0.128163	-0.024854	0.534559	0.248137	0.502145	0.228721	0.016958	0.000000
-0.114426	-0.126511	-0.025479	0.533668	0.246892	0.502243	0.228721	0.017720	0.000000
-0.112417	-0.125841	-0.025792	0.533485	0.246550	0.503293	0.229391	0.009337	0.000000
-0.110899	-0.126779	-0.026417	0.533741	0.247282	0.503891	0.235423	0.009337	0.000000
-0.113176	-0.127315	-0.026997	0.533644	0.246306	0.503439	0.226039	0.010861	0.000000
-0.113890	-0.127538	-0.026372	0.532997	0.244108	0.504160	0.197887	0.010099	0.000000

Figure 10 Sensor output of the 5DT data glove gesture

2.3 Precedent study

Extensive background research study has been done before starting this project. A lot of results have been achieved by various researchers using several techniques. Different sensors have been used to acquire data for processing to recognize sign language. The primary input device used to recognize sign language gesture is; data gloves/virtual reality sensors and camera based systems.

By the introduction of data glove-based hand gesture recognition, researchers have applied on various types of artificial neural networks (ANN). Beale and Edwards (1990) applied ANN to classify five static gestures from the American Sign Language (ASL) [1].

Murakami and Taguchi (1991) applied a back propagation ANN and VPL Data Glove for the classification with 42 signs [2]. Weissman and Salomon (1999) using a Cyber Glove applied feed-forward and radial basis network (RBF) [3] to 20 static gestures. Xu (2006) suggested a gesture recognition system for a VR-based driving training system. He used Cyber Glove with 18 sensors and five users to train a feed forward ANN to recognize 15 different gestures and reported a 98 per cent recognition rate [4]. Jin et al. (2011) used a self-organizing map (SOM) for hand gesture recognition with data glove, 5DT Ultra 14 [5].

Feature extraction had been also practiced to improve gesture recognition rates. Fels and Hinton (1993) practiced ANNs to mine gesture features [6]. Another case is the SLARTI system for classification of Australian sign language by Vamplew and Adams (1996) [7]. Cyber Glove and a hybrid tracking system was used. The system comprised four ANNs which were implemented to form the feature vector. The most recent solution based on feature extraction was proposed by Oz and Leu (2011) [8]. The system classifies ASL using Cyber Glove, tracking system, and a feed-forward ANN.

Eisenstein et al. (2003) developed a four-layer system architecture: data acquisition, processing of gestural predicates, processing of temporal predicates, and also worked with prototype gestures [9]. Ibarguren et al. (2010) suggested a layered sign recognition architecture, processed static gestures and movements using fifth dimension 5DT 14 data glove and a motion tracking equipment [10].

Parvini and Shahabi (2005) suggested hand gestures recognition using biomechanical hand motion model and CyberGlove, where a “range of joint motion” was used to form a specific signature for each gesture [11]. As an extension to this work, Parvini et al. (2009) compared a simple ANN approach, multilayer approach, and the proposed bio-mechanical characteristics approach, using a 22-sensor CyberGlove [12].

Newbee (1994) [13] applied statistical techniques in gesture recognition. Each gesture was assigned an ideal representative – prototype. Gesture recognition was based on similarity of the current input gesture with a particular prototype, which was determined based on the sum of squares.

Eisenstein et al. (2001) implemented K-means and adaptive clustering algorithms to static gestures [14]. Data acquisition was made by using Cyber Glove with Cyber Grasp exoskeleton, gesture recognition was implemented in two stages.

In first stage, system was trained to recognize input gestures into clusters. In the second stage, the input gestures were classified based on their closeness to the previously identified cluster centroids.

Using Cyber Glove, Shahabi et al. (2001) studied three gesture recognition systems based on ANN, C4.5 decision tree, and a Bayesian classifier [15].

Many researchers [16-27] used skin filtering method for segmentation of hand. This technique separated the skin colored pixels from the non-skin colored pixels, thus extracting the hand from the background. Fang [28] practiced Adaptive Boost algorithm could not only notice single hand but also the overlapped hands. In [29-31] color gloves were used for segmentation purpose.

Saengsri [29] used '5DT Data Glove 14 Ultra' data glove attached with 141-10 sensors on fingers & 4 sensors between the fingers, measures flexures and abductions respectively with 94 % accuracy for Thai Sign Language. Kim [30] used 'KHU-1' data glove having 3 accelerometer sensor, Bluetooth and controller which extracted features like joints of hand. Weissmann [31] used Cyberglove which record features like thumb rotation, angle made between the consecutive fingers and wrist pitch.

There have been many approaches for feature extraction PCA, Hit-Miss Transform, Principle Curvature Based Region detector (PCBR), 2-D Wavelet Packet Decomposition (WPD). In [16][32] Principal Component Analysis (PCA) was applied for extracting features for recognition of various hand gestures. Kong [32] segmented the 3-D images into lines and curves and then PCA was used to determine features like direction of motion, shape, position and size. Joyeeta and Karen able to handle different static alphabets of Indian Sign Languages by using Eigen value weighted Euclidean distance between Eigen vectors as a classification technique [33].

It is notable that, beside the application of ANN and various statistical methods, all the discussed contributions use sophisticated data gloves with 10, 14, 18 or 22 sensors.

These data gloves provide abundance of information required for feature extraction and gesture recognition. Moreover, Cyber Glove is a de facto industrial standard. However, the price of these data gloves is prohibitive for low-budget VR systems, i.e. desktop VR configurations which employ the display screen of ordinary computer or low-level workstation while the operator wears stereoscopic glasses to observe the 3-D virtual scenes and interacts with the virtual environment with the help of data gloves and 3-D trackers (Xia et al., 2013) [16]. Ongjan applied probabilistic neural network for 12 data sets of 6 gestures acquired by 5DT ULTRA data glove and achieved 91% of accuracy.

Data glove have been chosen for this project because of its versatility and ease of use.

Past accuracies achieved on the basis of glove based systems

Table 5 Precedent Study Accuracy Comparison Chart

Input / Data	Year	sensors	No of signs	Accuracy	Technique Applied
static gestures using 5DT Data glove	2014	5	12	91	cluster based Probabalistic Neural Network
Indonesian Sign Language Gestures	2013	2	50	91	Adaptive neighborhood based modified backpropagation NN
ASL sign language using cyber glove & flock of bird system	2011	18	50	90	back propagation ANN
SL gestures Cyber glove and VR-system	2006	18	15	98	feed forward NN
Pakistan Sign language Static Alphabets 5DT Dataglove	2005	5	27	78	Template Matching
Vietname Sign language using sensory glove	2005	6	23	90	Fuzzy logic
Gestures Using Nintendo Power Glove Glove	1995	3	95	80	IBL1

CHAPTER 3: RESEARCH ANALYSIS

The mentioned methodology is applied on the two repositories used for research which is explained in detail below:

3.1.1 Nintendo Power Glove

The sign gesture file contains 15 comma separated values in each row giving information of individual factor/feature. As mentioned above in data acquisition that 6 out of 15 were ignored due to irrelevancy of information.

The data file is read separately and after pre-processing, features are calculated and saved in featured file. Each feature has its own individual file. After calculating all the features from the filtered data several classification techniques are applied which includes LDA, QDA, Naïve Bayes, Decision Tree and ANNs.

For Nintendo power glove data we have 5 singers, each signer have 8 samples for each sign and total of 95 signs. Each featured file has 9 columns. So for total of 95 signs and single feature we got 95 rows and 9 columns i.e., 95 x 9 matrixes. For 8 samples each we got 95 x8 =760 i.e., 760 x 9 matrix. 10 fold cross validation is applied on each classifier. Therefore for calculation of accuracy of recognition for each feature, we applied these classifiers.

3.1.2 Signer 1 Results

3.1.2.1 LDA

The accuracy results for LDA for each feature we got following results;

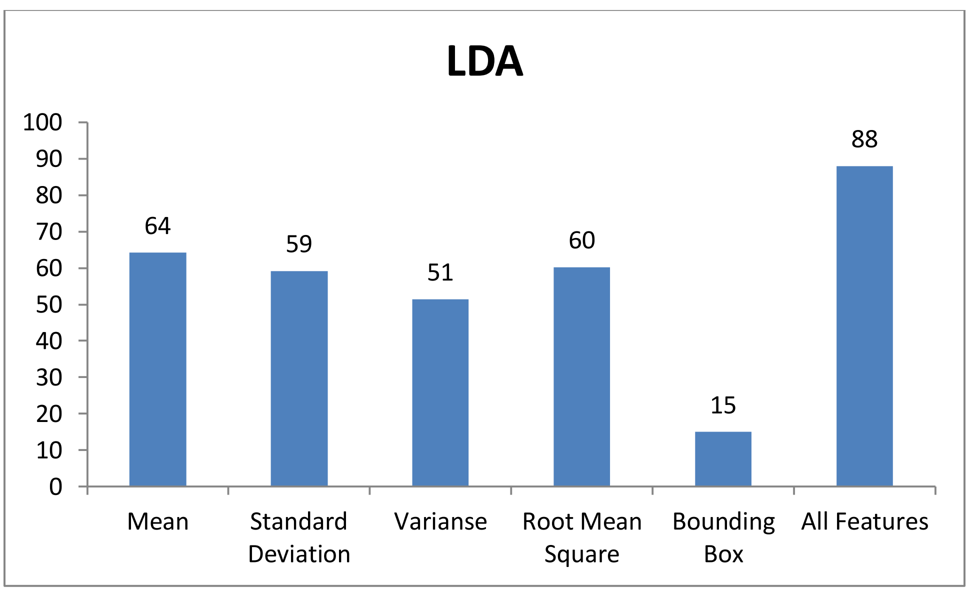


Figure 11 Signer1 LDA Results

3.1.2.2 QDA
The accuracy results for QDA for each feature we got following results;

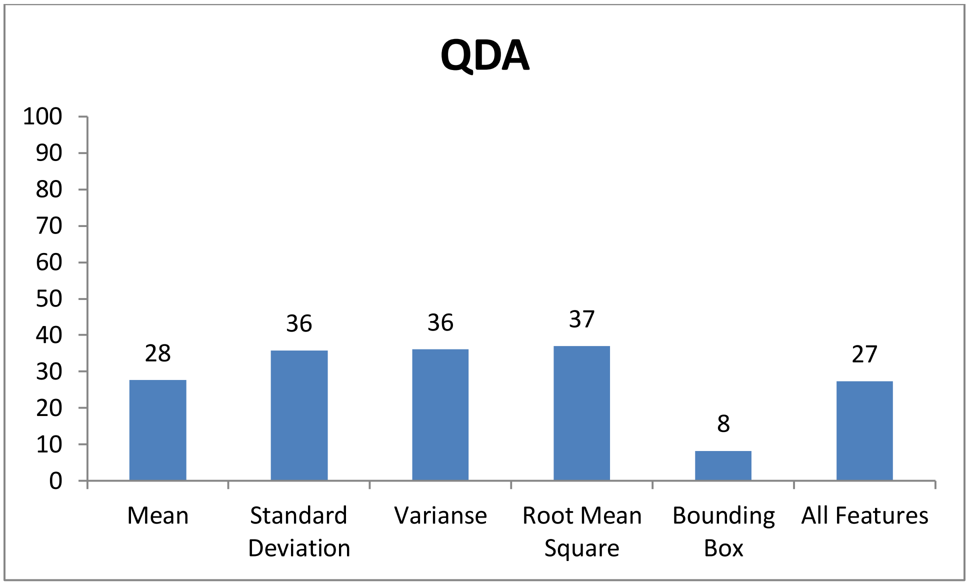


Figure 12 Signer1 QDA Results

3.1.2.3 NB (Gaussian)
The accuracy results for NB (G) for each feature we got following results;

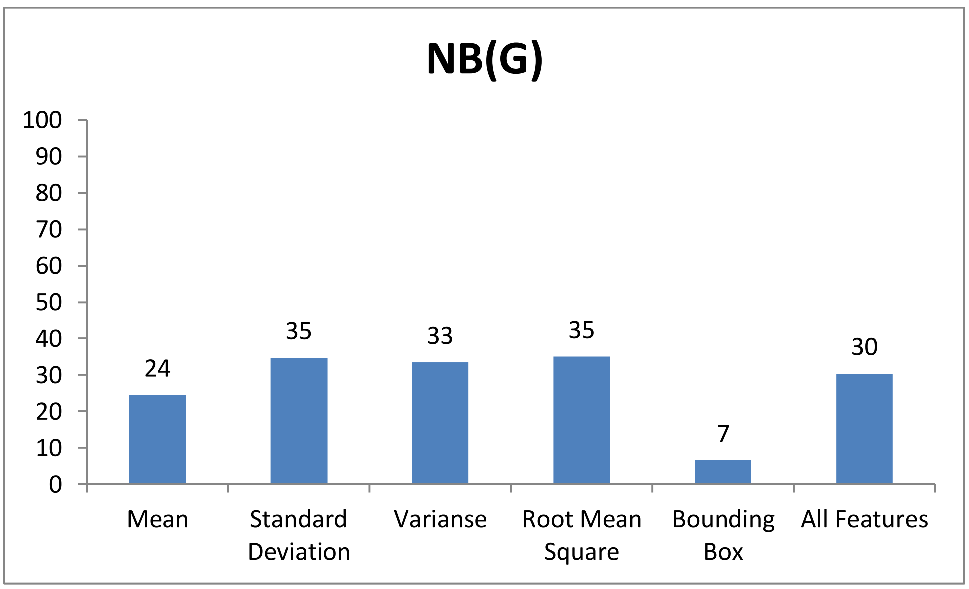


Figure 13 Signer1 NB (G) Results

3.1.2.4 *Naïve Bayes (Kernel)*

The accuracy results for NB (K) for each feature we got following results;

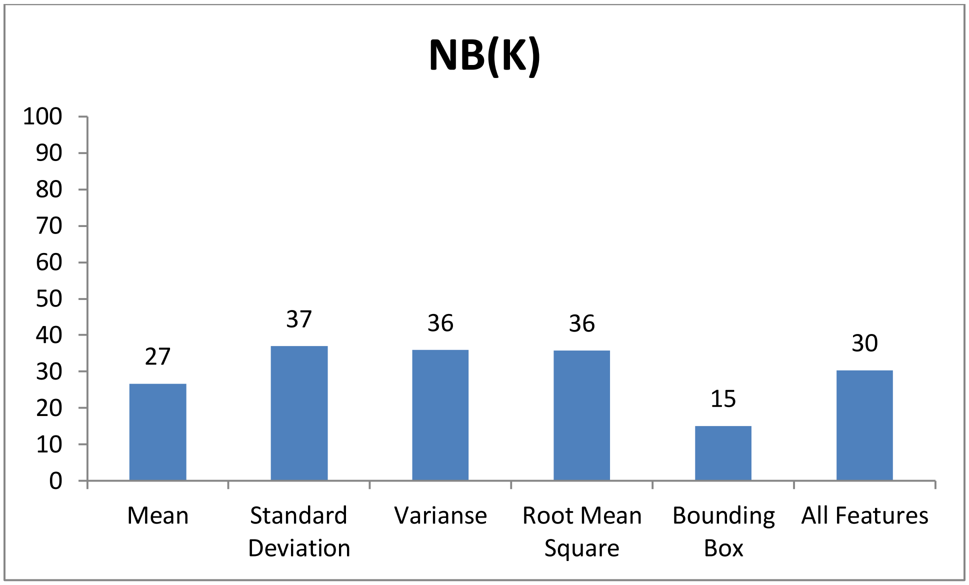


Figure 14 Signer1 NB (K) Results

3.1.2.5 *Decision Tree*

The accuracy results for DT for each feature we got following results;

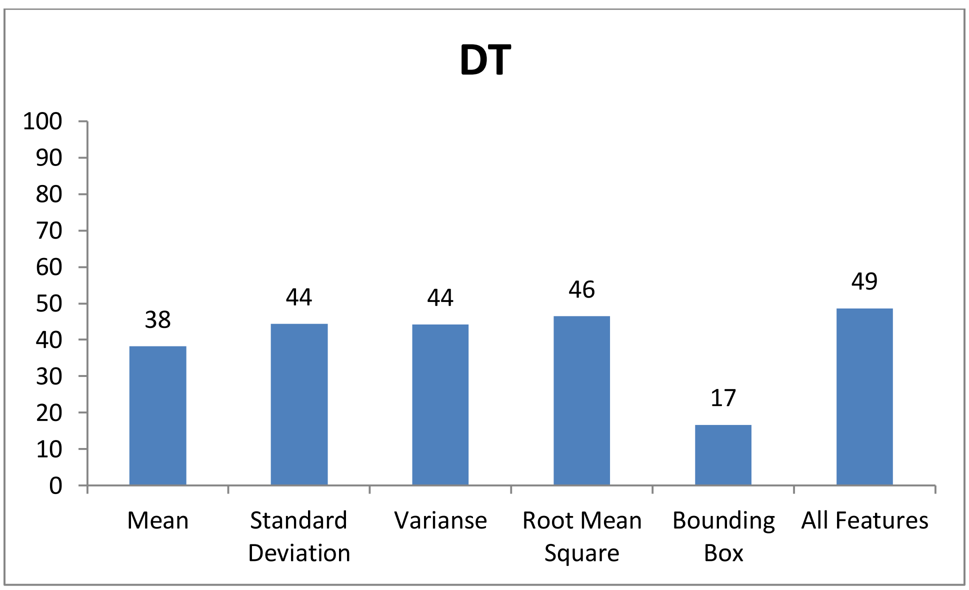


Figure 15 Signer1 DT Results

3.1.3 Signer 2 Results

3.1.3.1 LDA

The accuracy results for LDA for each feature we got following results;

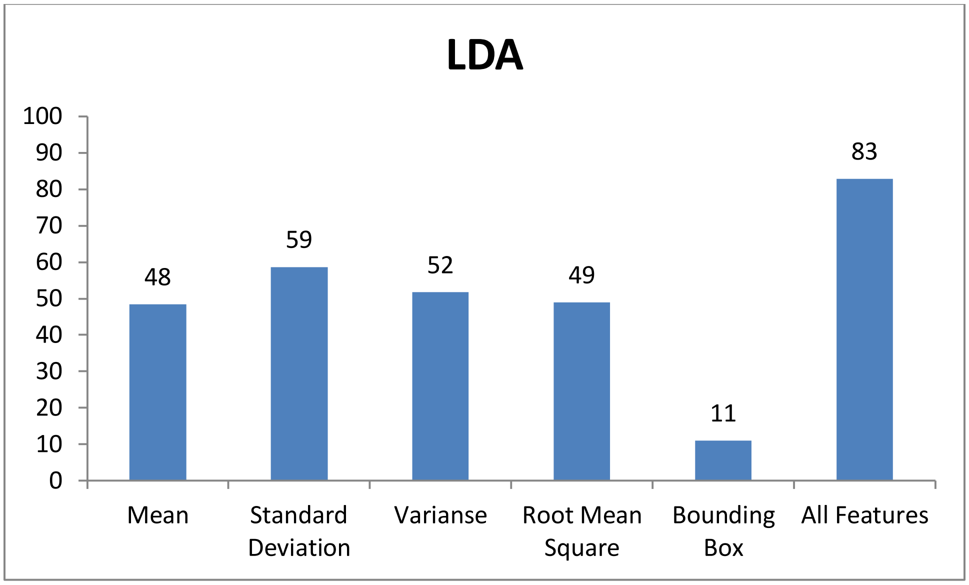


Figure 16 Signer2 LDA Results

3.1.3.2 QDA

The accuracy results for QDA for each feature we got following results;

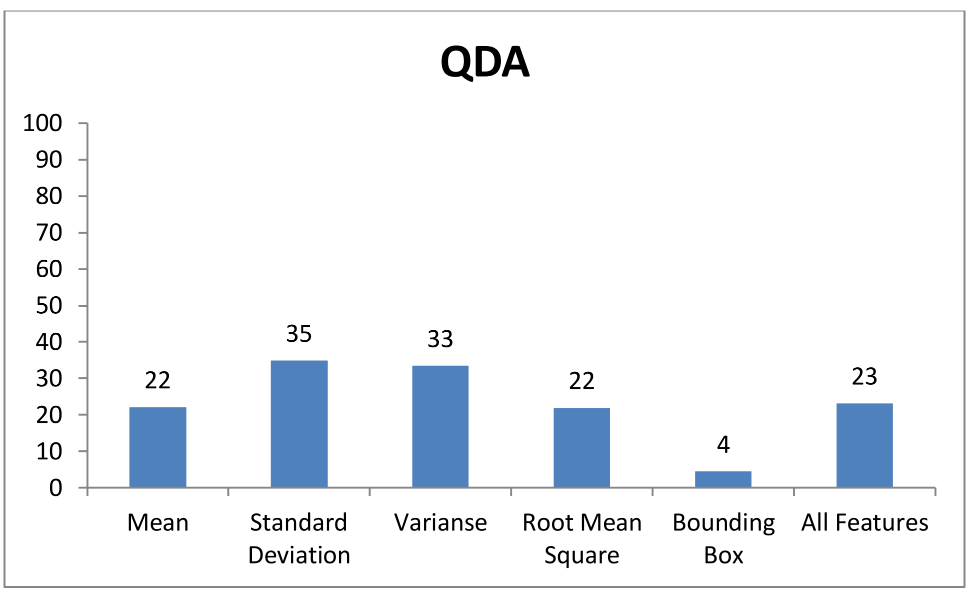


Figure 17 Signer2 QDA Results

3.1.3.3 NB (Gaussian)

The accuracy results for NB (G) for each feature we got following results;

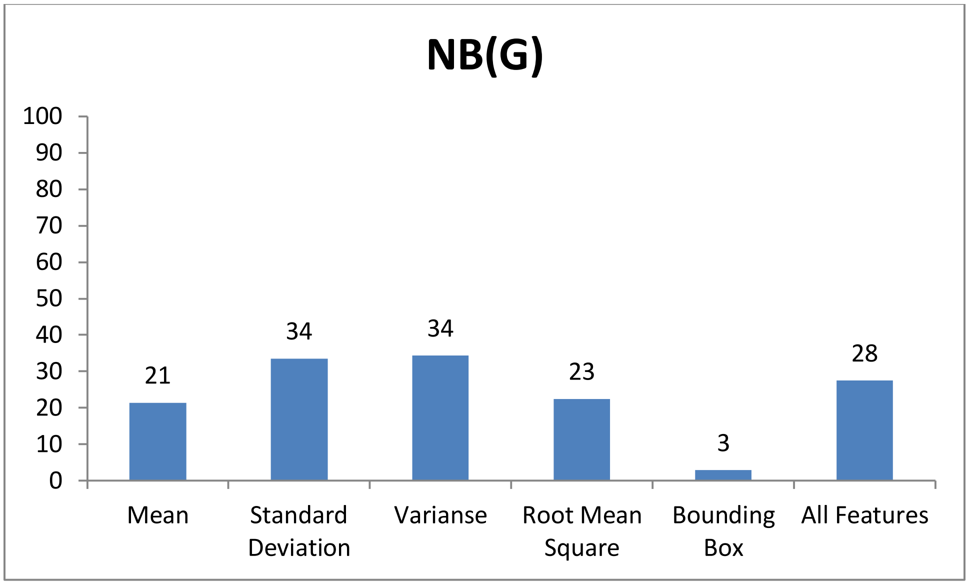


Figure 18 Signer2 NB (G) Results

3.1.3.4 Naïve Bayes (Kernel)

The accuracy results for NB (K) for each feature we got following results;

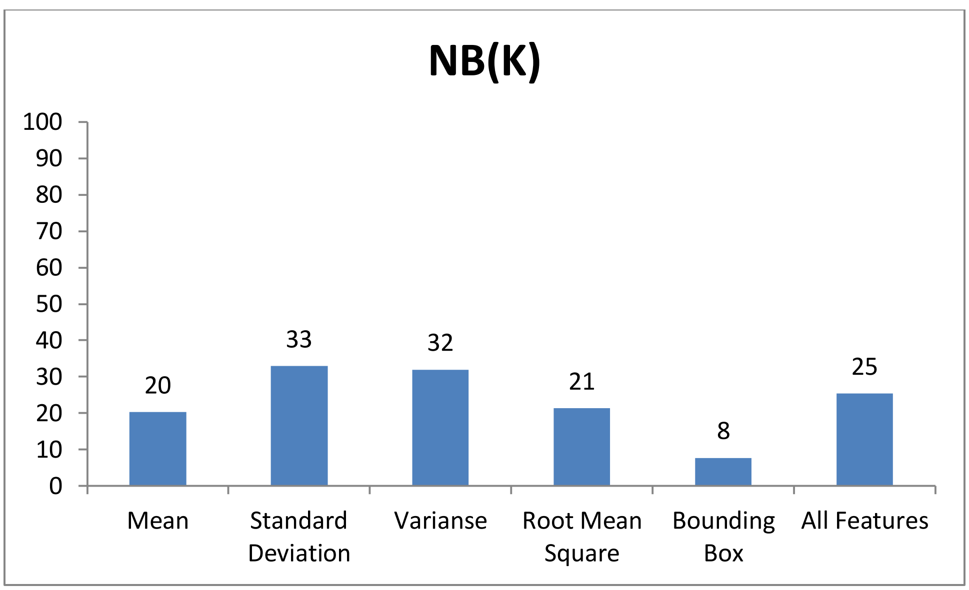


Figure 19 Signer2 NB (K) Results

3.1.3.5 Decision Tree

The accuracy results for DT for each feature we got following results;

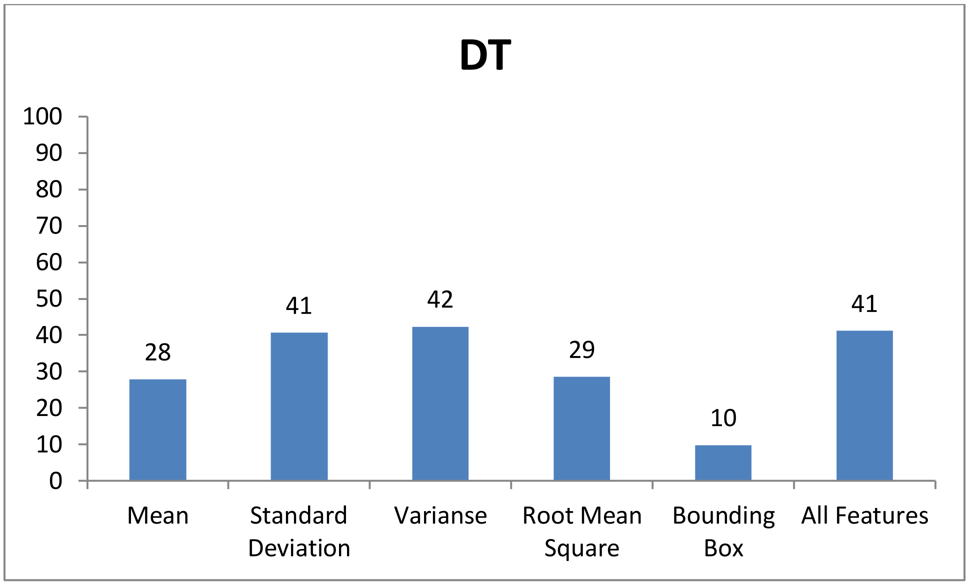


Figure 20 Signer2 DT Results

3.1.4 Signer 3 Results

3.1.4.1 LDA

The accuracy results for LDA for each feature we got following results;

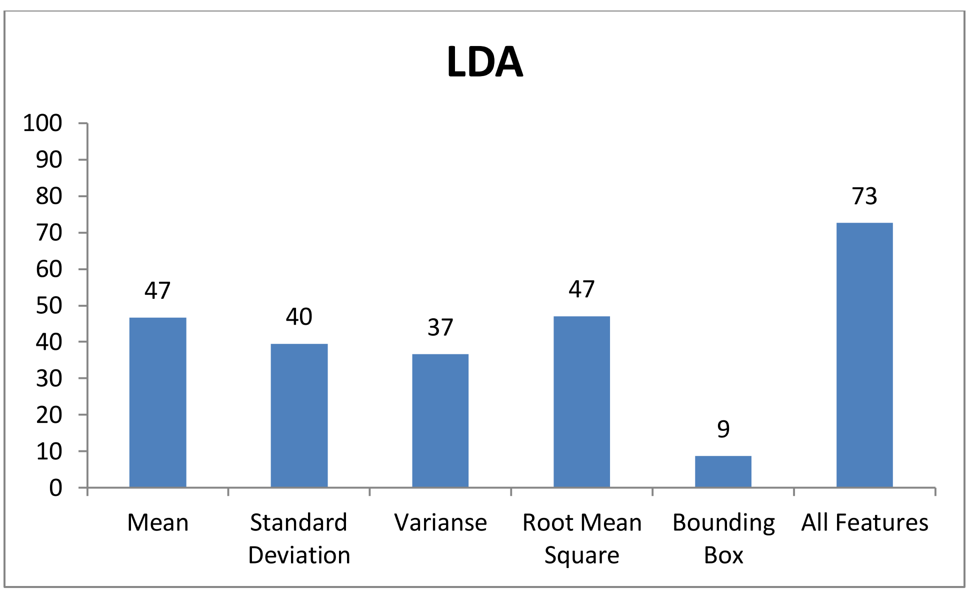


Figure 21 Signer3 LDA Results

3.1.4.2 QDA

The accuracy results for QDA for each feature we got following results;

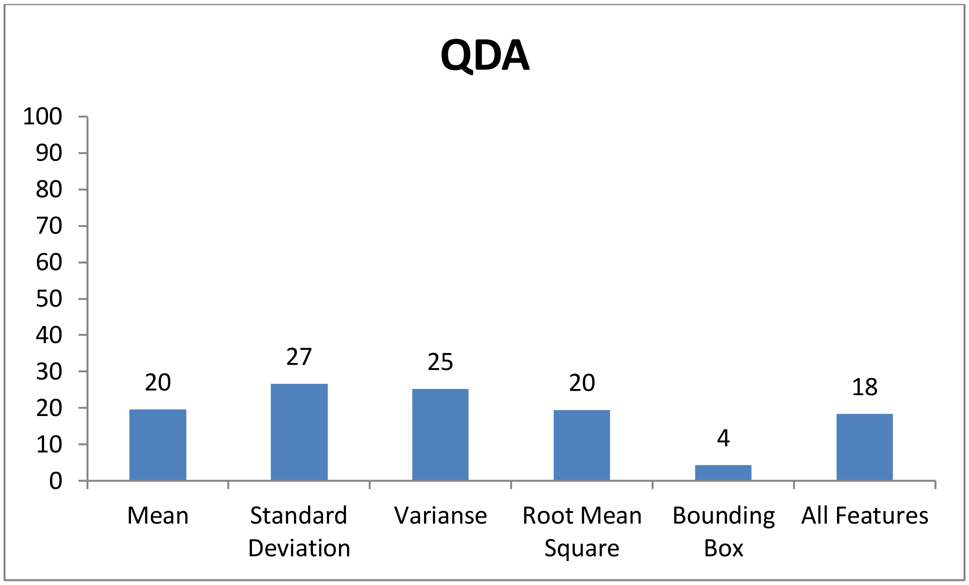


Figure 22 Signer 3 QDA Results

3.1.4.3 NB (Gaussian)

The accuracy results for NB (G) for each feature we got following results;

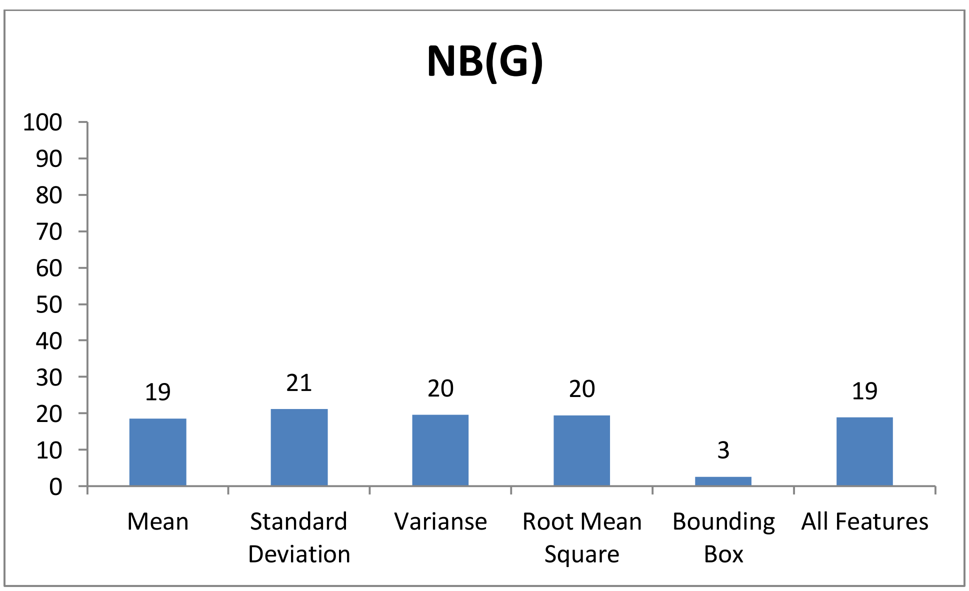


Figure 23 Signer3 NB (G) Results

3.1.4.4 NB (Kernel)

The accuracy results for NB (K) for each feature we got following results;

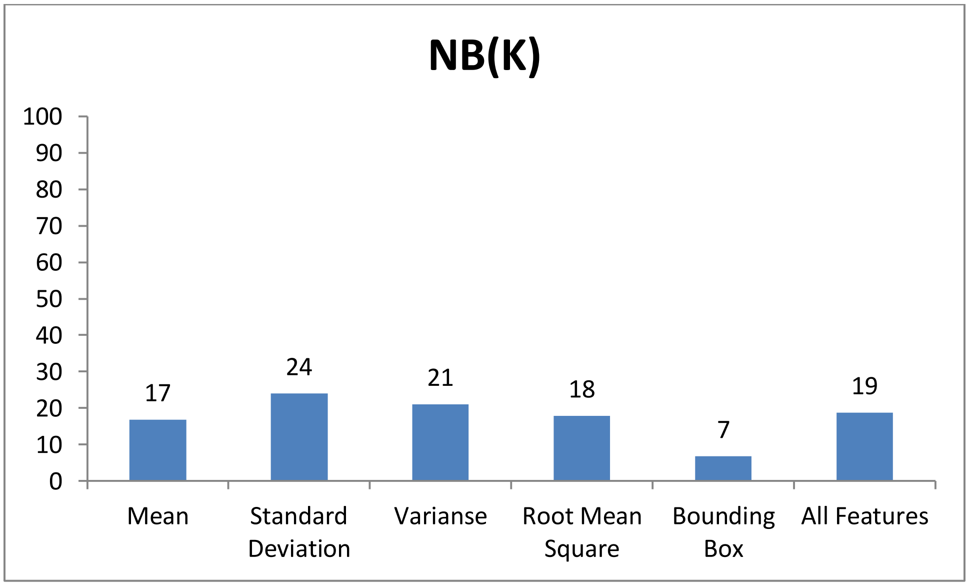


Figure 24 Signer3 NB (K) Results

3.1.4.5 Decision Tree

The accuracy results for DT for each feature we got following results;

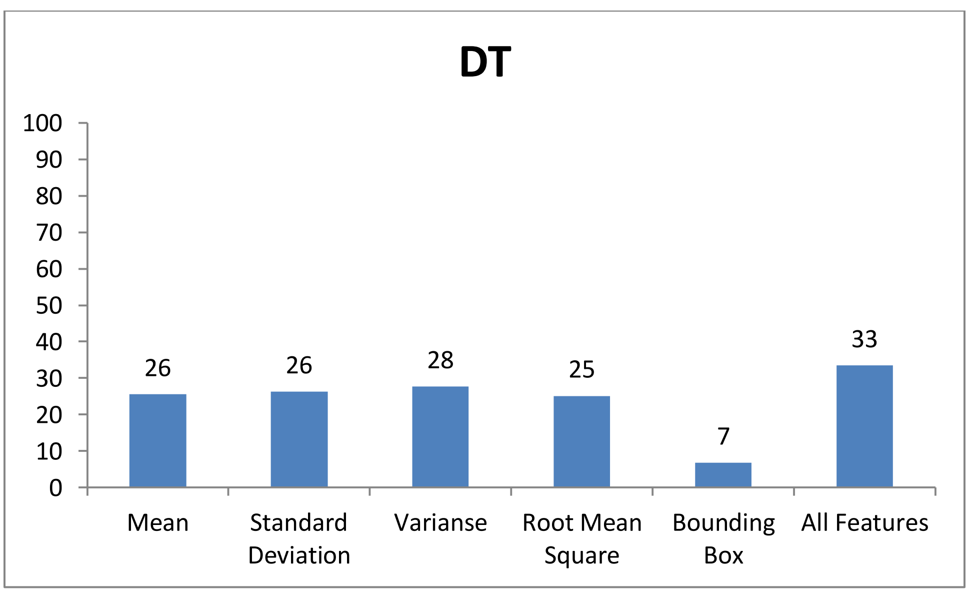


Figure 25 Signe3 DT Results

3.1.5 Signer 4 Results

3.1.5.1 LDA

The accuracy results for LDA for each feature we got following results;

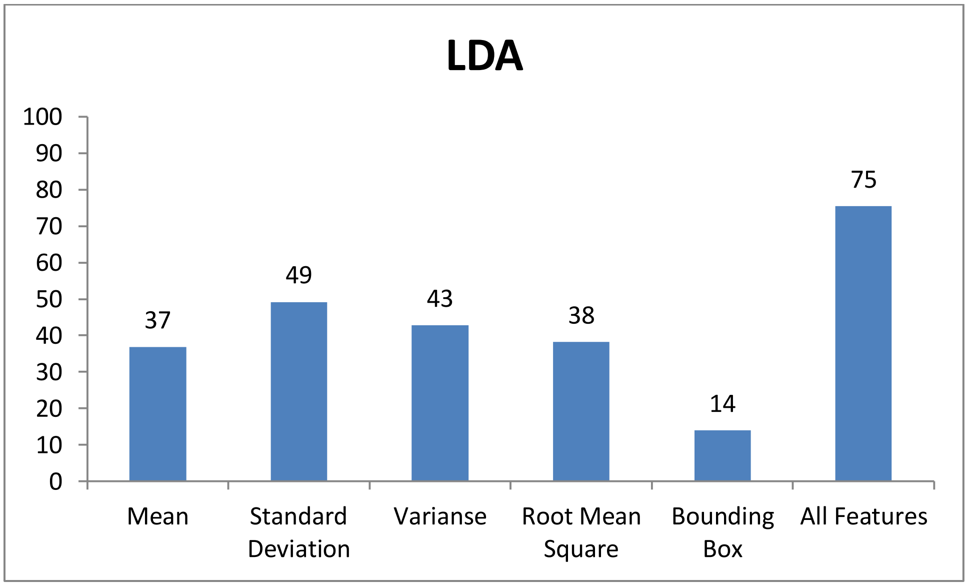


Figure 26 Signer4 LDA Results

3.1.5.2 QDA

The accuracy results for QDA for each feature we got following results;

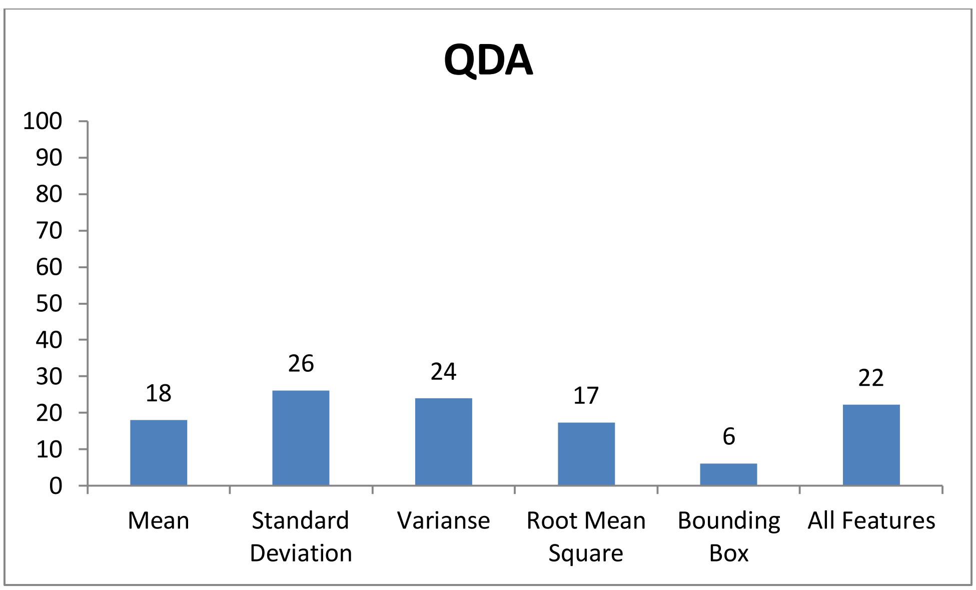


Figure 27 Signer4 QDA Results

3.1.5.3 NB (Gaussian)

The accuracy results for NB (G) for each feature we got following results;

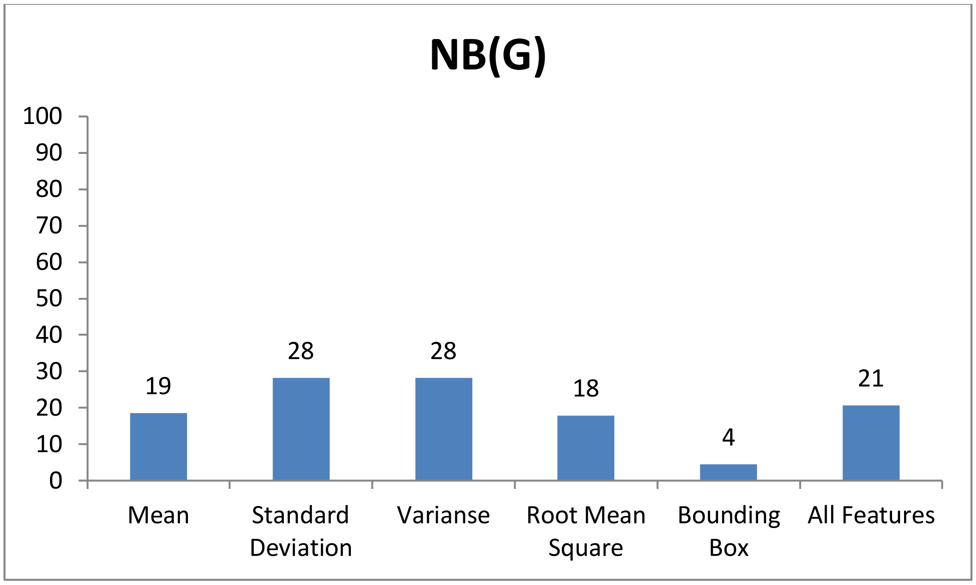


Figure 28 Signer4 NB (G) Results

3.1.5.4 NB (Kernel)

The accuracy results for NB (K) for each feature we got following results;

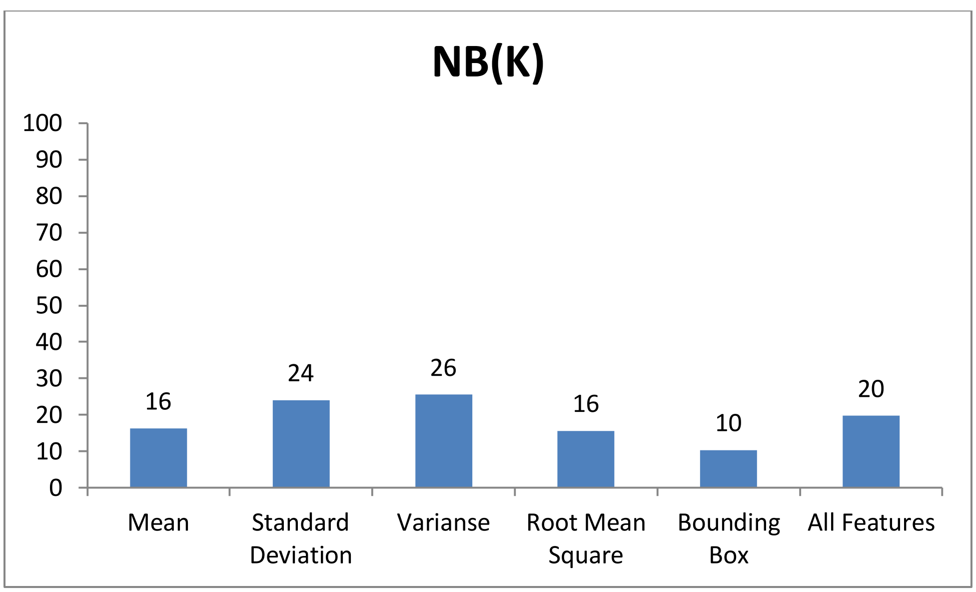


Figure 29 Signer4 NB (K) Results

3.1.5.5 Decision Tree

The accuracy results for DT for each feature we got following results;

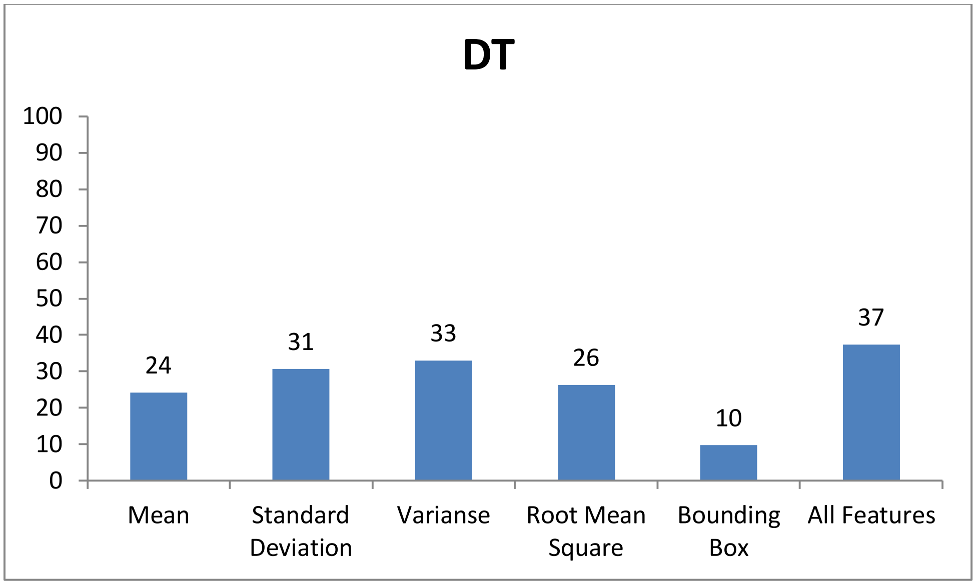


Figure 30 Signer4 DT Results

3.1.6 Signer 5 Results

3.1.6.1 LDA

The accuracy results for LDA for each feature we got following results;

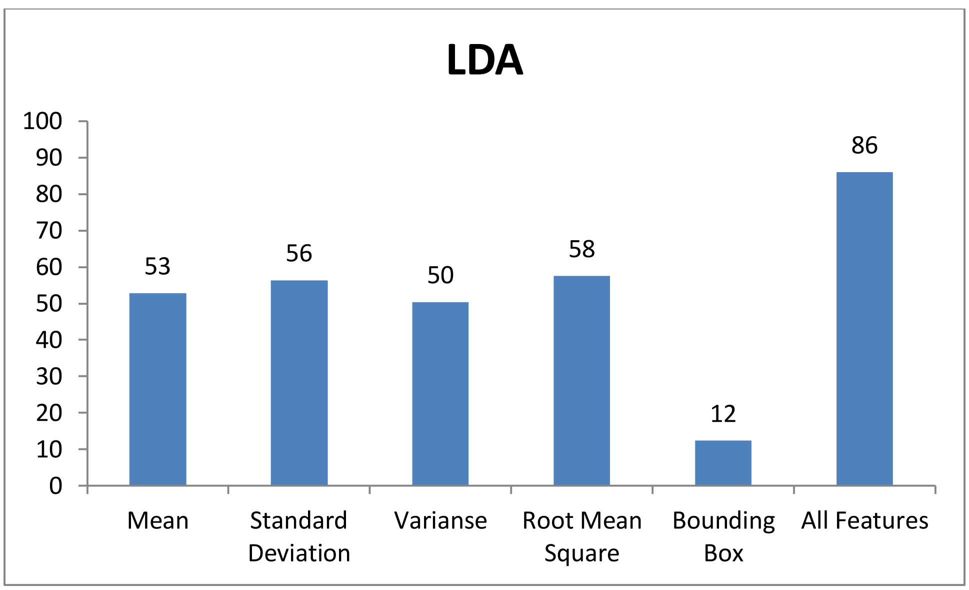


Figure 31 Signer5 LDA Results

3.1.6.2 QDA

The accuracy results for QDA for each feature we got following results;

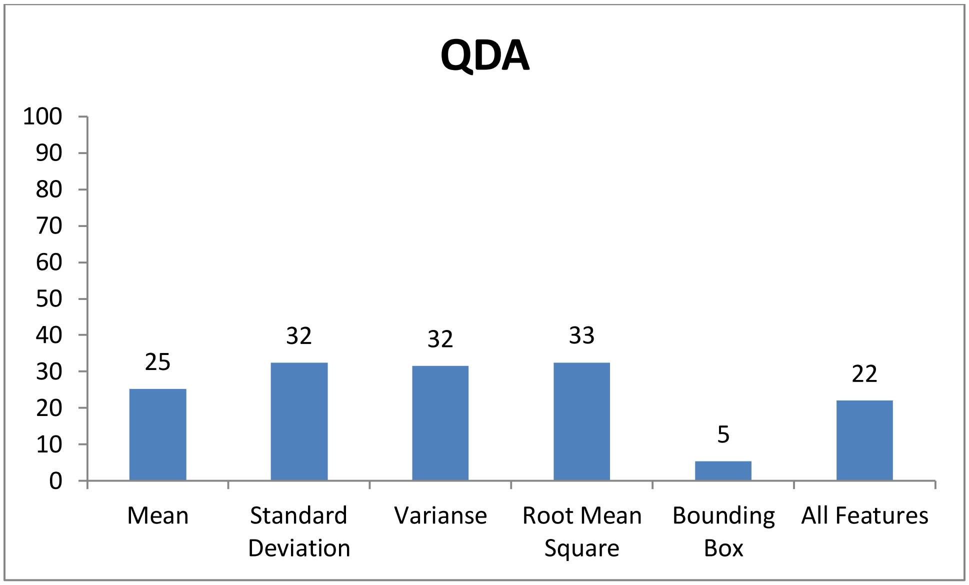


Figure 32 Signer5 QDA Results

3.1.6.3 NB (Gaussian)

The accuracy results for NB (G) for each feature we got following results;

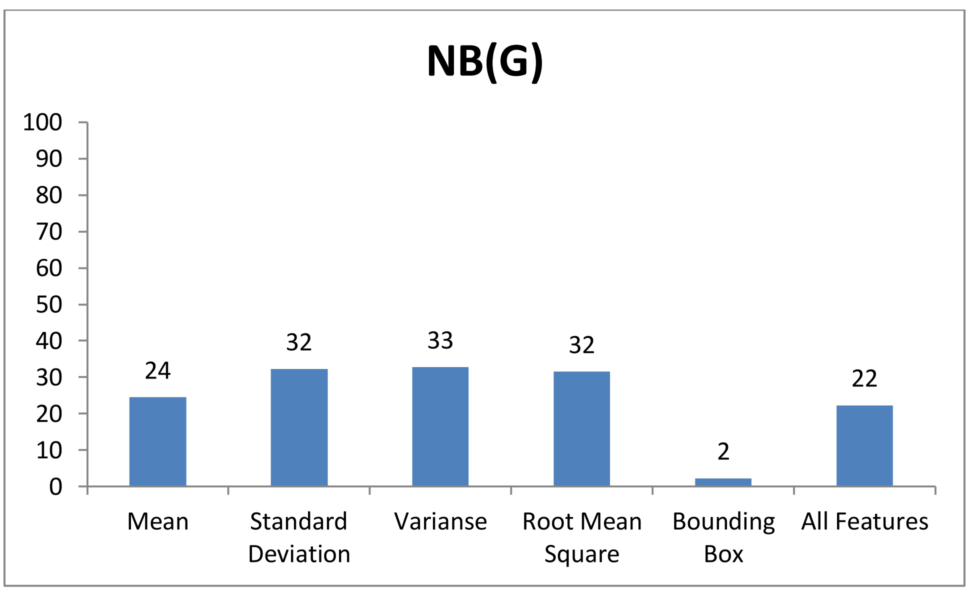


Figure 33 Signer5 NB (G) Results

3.1.6.4 NB (Kernel)

The accuracy results for NB (K) for each feature we got following results;

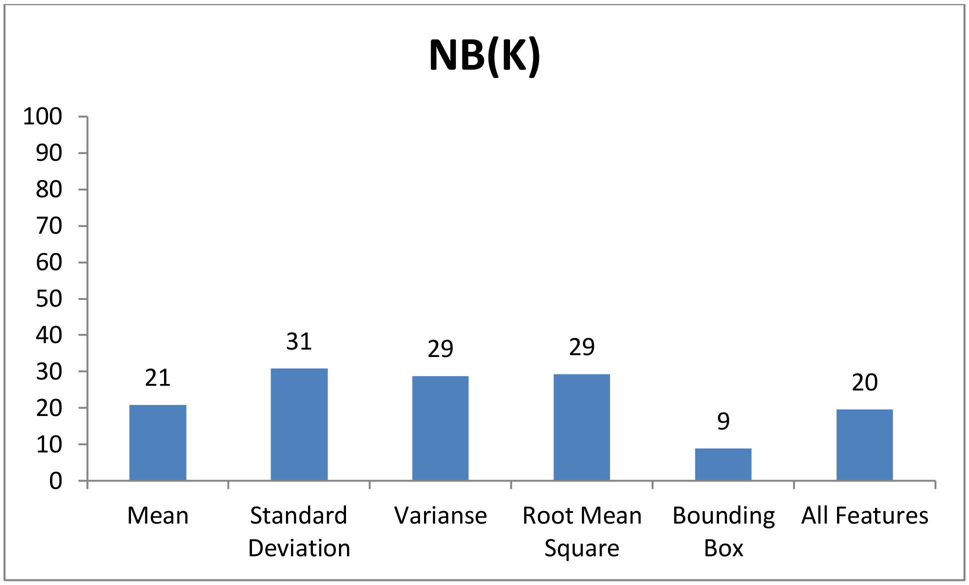


Figure 34 Signer5 NB (K) Results

3.1.6.5 Decision Tree

The accuracy results for DT for each feature we got following results;

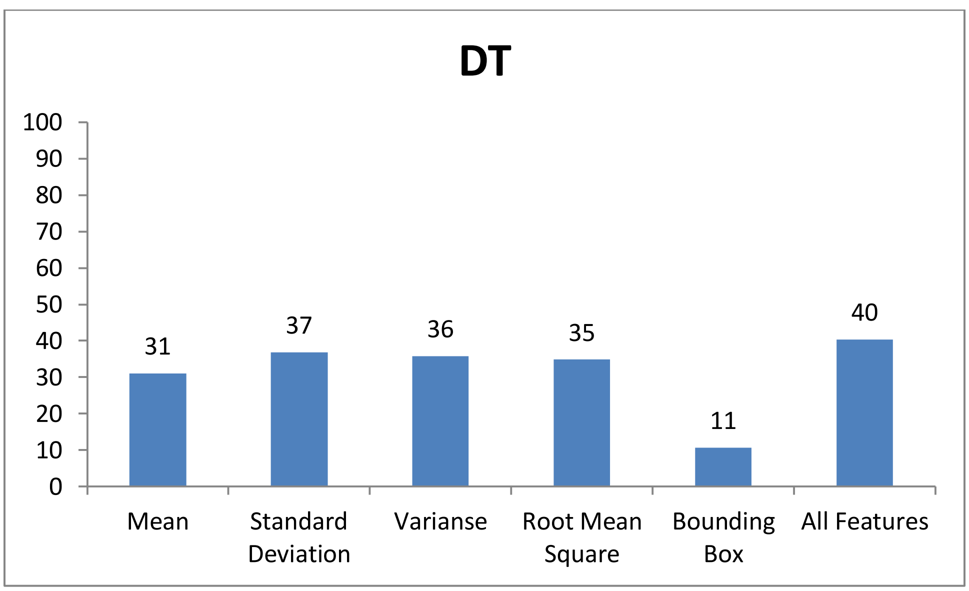


Figure 35 Signer5 DT Results

3.1.7 Conclusion for Nintendo Power Glove

Nintendo power lies in the cheap and nasty gloves. The probability of providing false and noisy information by this glove is high. It provides low accuracy data information. Based on these facts the accuracy results for this data is very low. Moreover LDA classifies with the highest accuracy for this glove data because LDA includes PCA in its methodology and LDA proves to produce better results for linear and continuous data. As information data is linear and continuous LDA finds its space to achieve results in better way than the other classifiers applied. The best feature which provides the good accuracy is mean value and standard deviation value. Mean shows the mid value of the data, standard deviation provides over all spread of the data; that's why they showed the acceptable results.

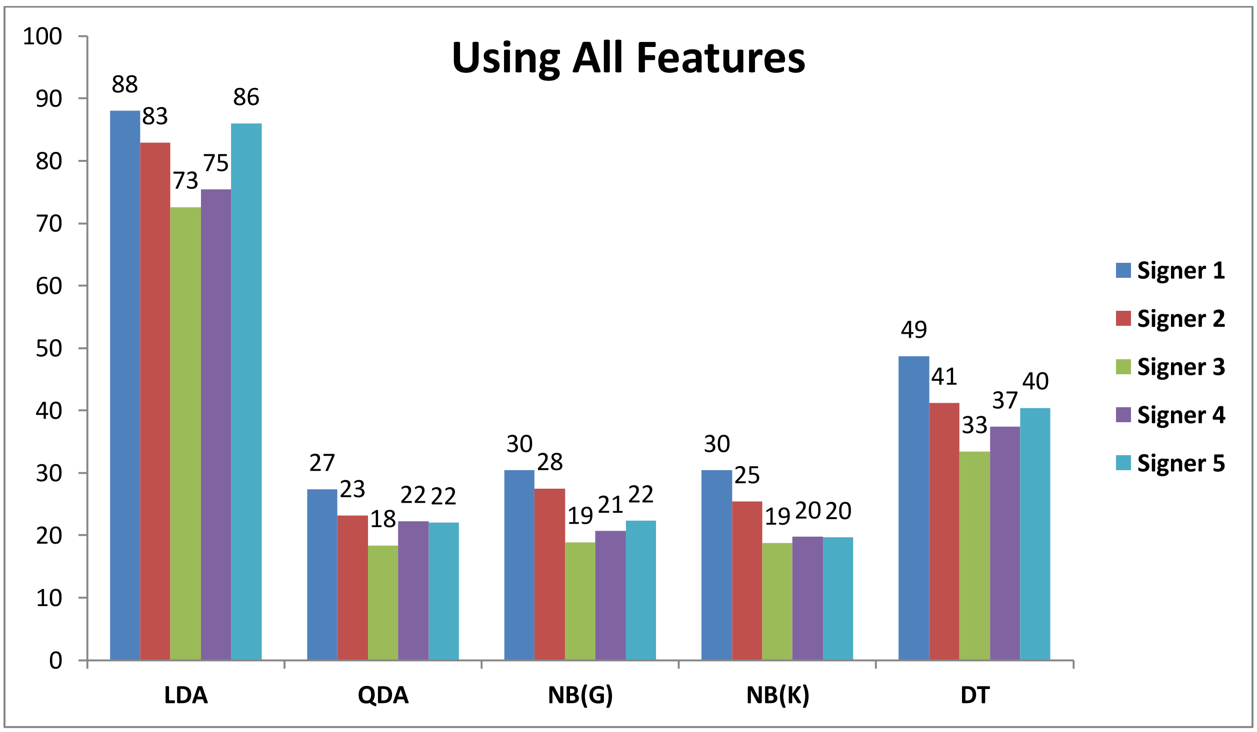


Figure 36 Comparison of All 5 Signer using All Effective Features

The confusion matrix analysis for Signer 1 is shown below;

Table 6 confusion matrix analysis signer 1

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR
LDA	669	76	75	0	0.89	0.89	0.89	0.11

QDA	204	254	299	0	0.45	0.41	0.43	0.59
NBG	339	209	212	0	0.62	0.62	0.62	0.38
NBK	456	179	125	0	0.72	0.78	0.75	0.22

The confusion matrix analysis for Signer 2 is shown below;

Table 7 confusion matrix analysis signer 2

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR
LDA	579	90	91	0	0.86	0.86	0.86	0.13
QDA	177	275	308	0	0.39	0.36	0.37	0.63
NBG	326	240	194	0	0.57	0.62	0.60	0.37
NBK	426	191	143	0	0.69	0.74	0.71	0.25

The confusion matrix analysis for Signer 3 is shown below;

Table 8 confusion matrix analysis signer 3

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR
LDA	513	111	136	0	0.82	0.79	0.80	0.20
QDA	166	281	313	0	0.37	0.34	0.35	0.65
NBG	260	282	218	0	0.47	0.54	0.50	0.45
NBK	361	243	156	0	0.59	0.69	0.64	0.30

The confusion matrix analysis for Signer 4 is shown below;

Table 9 confusion matrix analysis signer 4

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR

LDA	519	113	128	0	0.82	0.80	0.81	0.19
QDA	144	323	293	0	0.30	0.32	0.31	0.67
NBG	225	282	253	0	0.44	0.47	0.45	0.52
NBK	343	225	192	0	0.60	0.64	0.62	0.35

The confusion matrix analysis for Signer 5 is shown below;

Table 10 confusion matrix analysis signer 5

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR
LDA	608	76	76	0	0.88	0.88	0.88	0.11
QDA	161	314	285	0	0.33	0.36	0.34	0.63
NBG	283	243	234	0	0.53	0.54	0.54	0.45
NBK	358	203	199	0	0.63	0.64	0.64	0.35

3.2 5DT Data Glove

The sign gesture file contains 22 space separated values in each row giving information of individual factor/feature. As mentioned above in data acquisition that two gloves are used for both hands separately. First 11 values shows the attributes information for left hand and next 11 i.e., 12-22 values shows attributes information for right hand.

The data file is read separately and after pre-processing, features are calculated and saved in featured file. Each feature has its own individual file. After calculating all the features from the filtered data several classification techniques are applied which includes LDA, QDA, Naïve Bayes, Decision Tree and ANNs.

For 5DT data glove we have single singer, 27 samples for each sign and total of 95 signs. Each featured file has 22 columns. So for total of 95 signs and single feature we got 95 rows and 22 columns i.e., 95 x 22 matrixes. For 27 samples each we got 95 x 27 =760 i.e., 2565 x 22 matrix. 10 fold cross validation is applied on each classifier. Therefore for calculation of accuracy of recognition for each feature, we applied these classifiers.

3.2.1.1 LDA

The accuracy results for LDA for each feature we got following results;

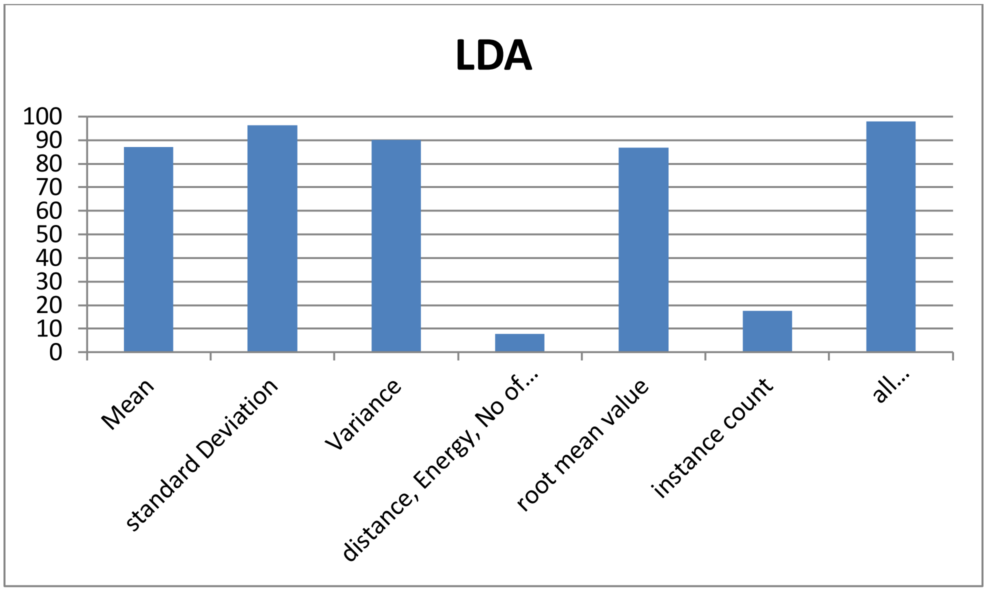


Figure XXXVII LDA Results for 5DT Glove

3.2.1.2 QDA

The accuracy results for QDA for each feature we got following results;

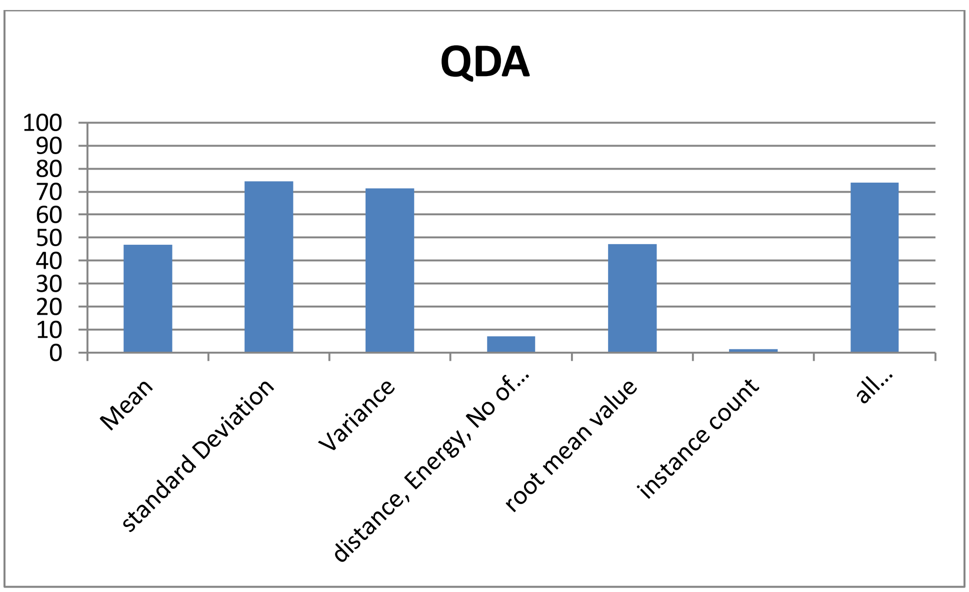


Figure XXXVIII QDA Results for 5DT Glove

3.2.1.3 NB (Gaussian)

The accuracy results for NB (G) for each feature we got following results;

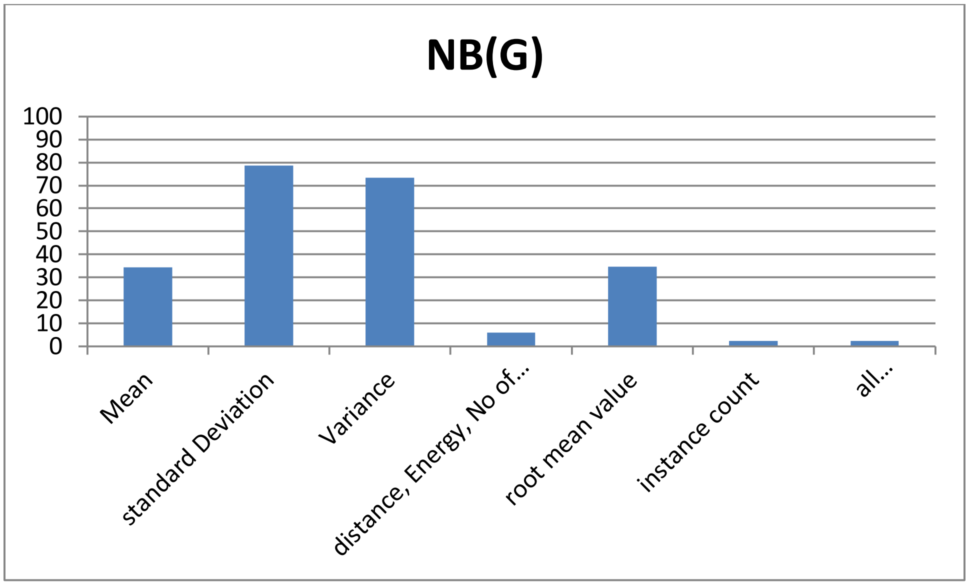


Figure XXXIX NB(G) Results for 5DT Glove

3.2.1.4 Decision Tree

The accuracy results for DT for each feature we got following results;

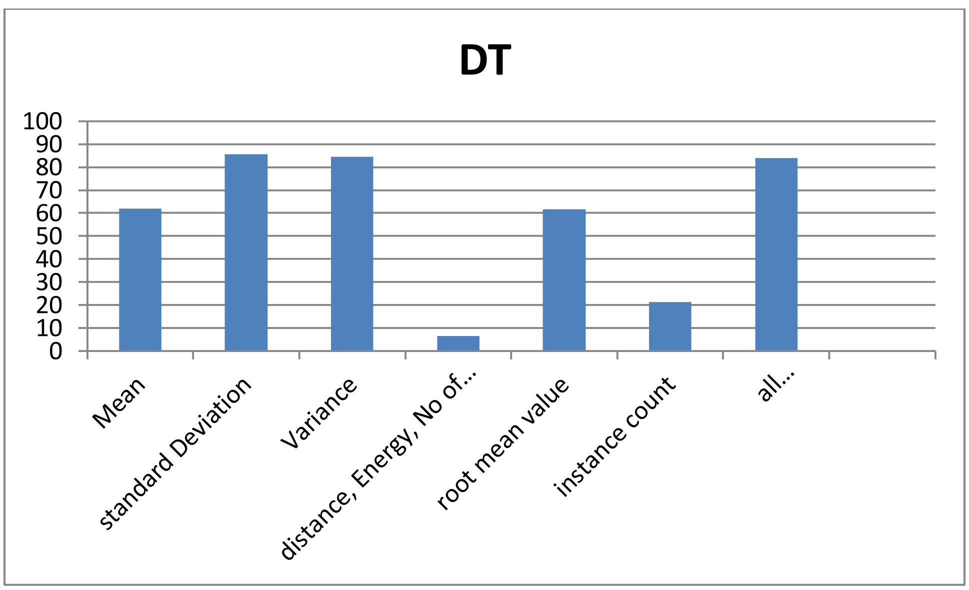


Figure XL DT Results for 5DT Glove

The confusion matrix attributes results are:

Table 11 5DT Data Glove Accuracy Evaluation LDA Results

	True Positive	False Positive	False Negative	True Negative	Precision	Recall	F-measures	FNR	FPR
Class 1	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 2	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 3	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 4	26	2	1	2512	0.93	0.96	0.95	0.04	0.00
Class 5	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 6	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 7	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 8	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 9	24	1	2	2514	0.96	0.89	0.92	0.11	0.00
Class 10	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 11	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 12	23	1	3	2515	0.96	0.88	0.92	0.12	0.00
Class 13	27	2	0	2511	0.93	1.00	0.96	0.00	0.00
Class 14	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 15	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 16	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 17	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 18	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 19	27	1	0	2511	0.96	1.00	0.98	0.00	0.00
Class 20	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 21	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 22	26	0	1	2512	1.00	0.96	0.98	0.04	0.00

Class 23	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 24	26	1	1	2512	0.96	0.96	0.96	0.04	0.00
Class 25	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 26	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 27	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 28	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 29	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 30	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 31	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 32	27	2	0	2511	0.93	1.00	0.96	0.00	0.00
Class 33	25	4	2	2513	0.86	0.93	0.89	0.07	0.00
Class 34	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 35	25	0	0	2513	1.00	1.00	1.00	0.00	0.00
Class 36	26	1	1	2512	0.96	0.96	0.96	0.04	0.00
Class 37	23	0	4	2515	1.00	0.85	0.92	0.15	0.00
Class 38	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 39	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 40	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 41	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 42	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 43	27	1	0	2511	0.96	1.00	0.98	0.00	0.00
Class 44	26	1	1	2512	0.96	0.96	0.96	0.04	0.00
Class 45	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 46	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 47	27	0	0	2511	1.00	1.00	1.00	0.00	0.00

Class 48	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 49	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 50	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 51	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 52	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 53	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 54	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 55	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 56	24	1	3	2514	0.96	0.89	0.92	0.11	0.00
Class 57	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 58	27	1	0	2511	0.96	1.00	0.98	0.00	0.00
Class 59	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 60	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 61	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 62	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 63	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 64	27	1	0	2511	0.96	1.00	0.98	0.00	0.00
Class 65	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 66	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 67	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 68	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 69	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 70	25	1	1	2513	0.96	0.96	0.96	0.04	0.00
Class 71	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 72	27	0	0	2511	1.00	1.00	1.00	0.00	0.00

Class 73	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 74	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 75	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 76	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 77	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 78	26	1	0	2512	0.96	1.00	0.98	0.00	0.00
Class 79	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 80	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 81	26	0	1	2512	1.00	0.96	0.98	0.04	0.00
Class 82	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 83	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 84	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 85	23	0	0	2515	1.00	1.00	1.00	0.00	0.00
Class 86	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 87	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 88	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 89	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 90	25	0	0	2513	1.00	1.00	1.00	0.00	0.00
Class 91	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 92	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 93	26	0	0	2512	1.00	1.00	1.00	0.00	0.00
Class 94	27	0	0	2511	1.00	1.00	1.00	0.00	0.00
Class 95	27	0	0	2538	1.00	1.00	1.00	0.00	0.00
overall	2513	22	30	0	0.99	0.99	0.99	0.01	1.00

The confusion matrix analysis for 5DT Glove is shown below;

Table 12 confusion matrix analysis 5DT Glove

	TP	FP	FN	TN	Precision	Recall	F-measures	FNR
LDA	2516	27	22	0	0.99	0.99	0.99	0.01
QDA	1777	385	403	0	0.82	0.82	0.82	0.18
NBG	2165	198	202	0	0.92	0.91	0.92	0.09
NBK	2388	89	88	0	0.96	0.96	0.96	0.04

3.2.2 Conclusion for 5DT Data Glove

5dt data glove lies in the category of quality and high precision gloves. The probability of providing false and noisy information by this glove is comparatively very low. It provides high accuracy data information of 14 bits. Based on these facts the accuracy results for this data is very good and acceptable. Moreover LDA classifies with the highest accuracy for this glove data because LDA includes PCA in its methodology and LDA proves to produce better results for linear and continuous data. As information data is linear and continuous LDA finds its space to achieve results in better way than the other classifiers applied. The best feature which provides the good accuracy is mean value, variance value and standard deviation value. Mean shows the mid value of the data, standard deviation provides over all spread of the data; that's why they showed the remarkable results.

3.3 Conclusion

Results show the difference between the two data repositories. 5DT data glove provides more accurate and acceptable results as compared to Nintendo power love data. The main reason behind this is 5dt data glove is more accurate in providing right information, provides high bits data, provides low noise data, and mainly we have data for both hands; left and right hand. Therefore theoretically and practically we expected more enhancing and good results from 5dt data glove repository, which are achieved successfully.

Coming to the classifiers applied LDA shows the best accuracy results as compared to the others. DT also compete LDA in some results. As data from both repositories is continuous and linear, LDA gives best results on linear data whereas QDA is for quadratic data, Naïve Bayes classifier proves to be good for Gaussian data and ANNs provides good results for data which is variant rather than linear. Therefore theoretically and practically we expected good results from LDA, which are achieved successfully.

In terms of best feature, mean and standard deviation proves to be the best features in providing the best result for classification. Mean provides the average value of the data, in x, y, z positional data mean provides the average, median value of the position in respective axis; which is the best information for the gesture/sign. Standard deviation gives the information regarding the spread of the data; how far data is dispersed, healthy information of the gesture. Combining these two features we got the overall best result for accuracy of 98%.

CHAPTER 4: SYNTHESIS

4.1 Synthesis and key finding

We have used set of features to classify the gestures. Each feature shows some accuracy but not enough to use it alone. In this situation we have combine some features to make the results more promising and effective.

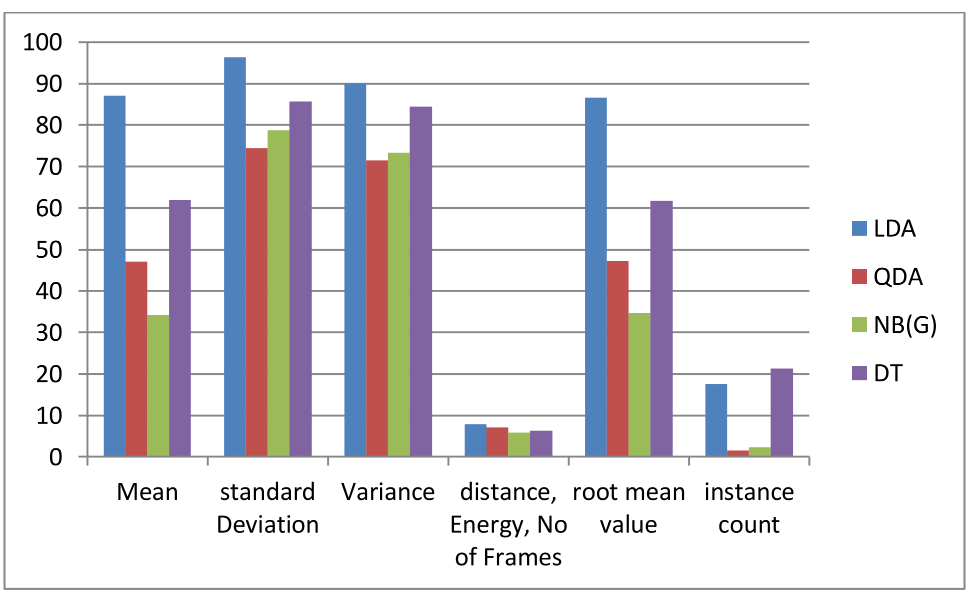
Of course we cannot expect that one feature has 50% accuracy and other also has 50% accuracy and combining both of them, we get 100% accuracy because there is an intersection in the input data of samples that would be correctly classified by either in other words, there easy samples that would be classified correctly by either feature set. In fact, for the above to occur, one set of feature would have to get every sample that the other one get wrong and vice versa. Also it is possible that adding some non-accurate set to accurate set will reduce the accuracy and higher the error rate.

Now questions to be answered now;

- Which is the most effective feature?
- Which is the most effective classifier?

4.1.1 Most effective feature

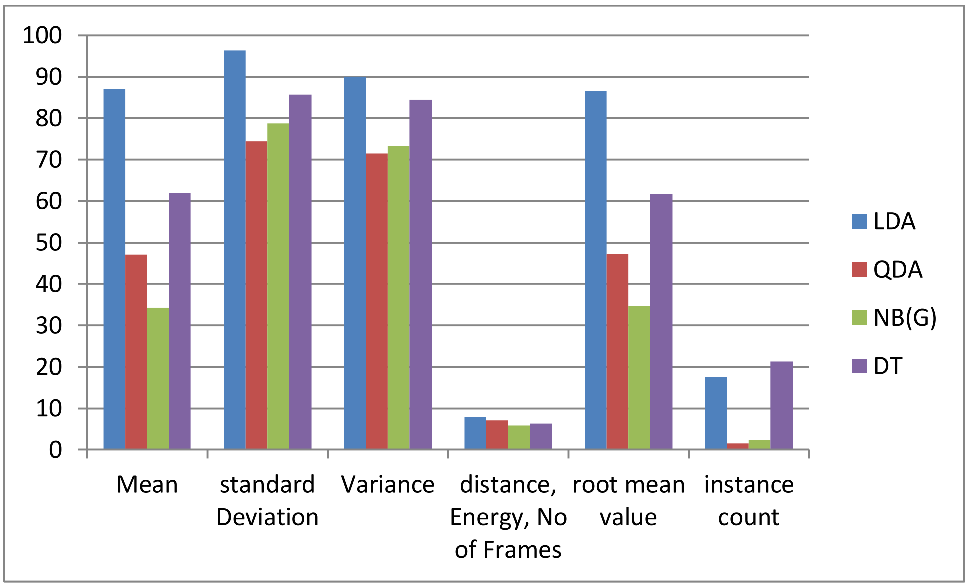
The criteria to judge most effective feature is simply the level of accuracy it has shown. Higher the accuracy and lesser the error rate will decide the feature to be best feature. For different classifiers different attributes has shown the better accuracy rate. The figure below shows the respective accuracy rates of feature used with the respective classifier



The standard deviation and variance shows the better results for all the classifiers used because they shows the spread of data, that how data is spread statistically. Both of them can be declared as the best feature used for the classification.

4.1.2 Most effective classifier

The criterion to judge most effective classifier is simply the level of accuracy it has shown. Higher the accuracy and lesser the error rate will decide the classifier to be best feature. For different classifiers different attributes has shown the better accuracy rate. The figure below shows the respective accuracy rates of feature used with the respective classifier



LDA has the better accuracy rate as compare to the other classifiers used because LDA performs better with the linear and continuous data, as the data used is linear and continuous the results are more accurate from LDA.

4.2 Architect's brief

This project is aimed to provide the way of communication to the deaf community. The idea is that signer will perform the gesture/sign and system will classify the sign and shows text and voice output as a way to communicate with the other normal person/community. The attributes proposed to cover are:

- Input the gesture to system
- Pre-processing of sign
- Feature Extraction
- Classification of gesture
- Text output
- Voice output
- GUI

The idea is to minimize the communication gap between person performing the sign language and the other who cannot understand the sign language. The deaf person will try to communicate with the community using this system, he will just preform the sign and rest is done by the system in form of text and voice narration.

The project is started with the aim to introduce a system that for somehow the deaf will use with ease, and minimize the complexity, make it very simple to work efficiently and for a longer period of time.

The signer will input the sign using the data glove, the glove provides the sensory output; different sensors used in different location of the glove; the sensor output is then fed to the system, system will first perform the pre-processing if the data i.e., reduce/filter the noise and make data more reliable to use. The system will then extract some basic feature set to use for classification. The feature sets then fed to the classifier to classify and recognize the sign. On the basis of their class attributes classifier will declare the class for that sign. Then the class is fed to the text and voice narration system it will produce text output on the screen and voice speech narration as well for the person who cannot understand the sign language.

Moreover the project includes the study of different classification techniques used before, implementation of those techniques on this data and comparison of their results. It also includes the study of features to be used, feature extraction and their impact on the accuracy respectively. Comparison of accuracy results of the features with respect to the classifier. Graphical user interface is designed for the system to make the use comfortable. A common person can handle the system.

4.3 Design process

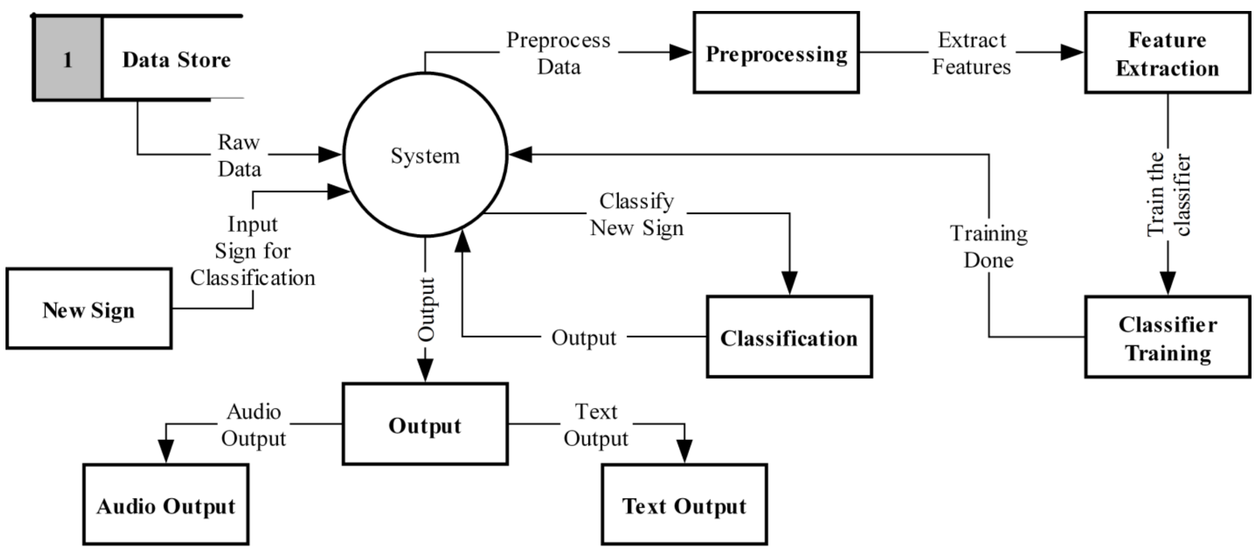
The system is designed with the purpose to acquire data from the signer using some sensor device. The design attributes kept in mind during the project are

- Simplicity of the system/project
- Efficiency of the system
- Accuracy of the classification
- Text and voice output
- GUI

The simpler the design is the more efficient would it be. The project emphasizes on the simplicity with the physical system involved and the functionality of the system as well. Many sensors can be used to input the sign data to the system like still cameras, video cameras, Kinect devices and sensory gloves.

Sensory gloves are selected because they are simple and easy to use. No complexity is involved. Moreover they take very less time to produce output.

The system then performs filtration on the sensor data for noise reduction. Feature extraction is performed to mine the attributes for classification of the signs. Classification of gestures is done by applying different classifiers to recognize the sign. Classifiers accuracy is measured in order to maintain the results for the best accuracy achieved. And then simply the text and voice output is generated for the end-user.



The complete system is iterated several times for the different classifiers and for the different data gloves used.

The efficiency of the system is kept high and it will perform more efficiently in time, processing, and accuracy. No complexity is involved. The system is very simple and efficient.

Measurement of classification accuracy is iterated several times i.e., with one data glove, one feature set, one classifier. In total of 2 gloves, 4 classifiers and 6 features set is used so finally the system is iterated almost 48 times.

The system outputs in form of text and voice. A GUI is designed for this purpose in which output inform of text is displayed and speaker narrate the voice.

CHAPTER 5: DESIGN CONCEPT

5.1 Design concept translation

The concept of project is to introduce a system that will input a sign from any input device and produce the text and voice output of that sign for the deaf community to communicate with the hearing community. The purpose of the concept is to minimize the communication depletion region of understanding that sign language.

In order to translate this concept into design we explored several techniques and processes. All the processes followed by the result comparison of different classifiers and features. The accuracy achieved is then compared with the previously achieved accuracy results.

5.2 Architect's brief translation

Following the design concept and architecture of the project, it is a significant achievement with efficiency and accuracy to introduce such a system in Pakistan which is a unique solution to the problem of communication between deaf and hearing community. The design of this system completely satisfies the architect's idea, objectives of the project and requirements of the end-user. This research opens the gateway to invent a device that will help the deaf community for the purpose of conversation with other normal world.

We have achieved a level of research to grasp an extent of accuracy to be accepted as suitable for this project. The design of this system completely satisfies the architect's idea, objectives of the project and requirements of the end-user. The simplicity, efficiency, accuracy and output of the project completely match with the initial objectives of the project.

Sign language is recognized with maximum accuracy attained with the text and voice output. An automated system is introduced to recognize the sign language of deaf and dumb people. Text and voice output will allow them to convey their message to the person who lacks in understanding the sign language.

5.3 Iteration to improve the system

The project is followed using step by step process. It starts with the data acquisition and ends at text and voice output. Several processes have been performed to generate output which includes

Data acquisition

Pre-processing of data/ Filtration/ Noise reduction

Feature extraction

Classification techniques

Training/testing of classifier

One single sign input

Text and voice output

These processes have been performed several times on the basis of features used and classifier applied. Many iterations of this system conclude the best feature and best classifier. Total of 7 features and 5 classifiers have been used for 6 data sets so approximately 210 iterations of the system have been performed.

These iteration effects the design phase by producing best features and best classifier. On the basis of these iterations mean, variance and standard deviation are declared as best feature set and LDA declared as best classifier which satisfies the main concept of the project and end-user requirements.

5.4 Impact/ longevity

Acceptable results are achieved in accuracy and efficiency. The system designed is very useful for the hearing impaired community. This system will not only erase the communication problem for deaf community but also provides them a platform to teach sign language to the children. This system can be used to practice sign language to make the children more comfortable to communicate with the normal world. It will make them realize that they are not aliens in this world but they are just like the other persons alive and common.

There are many uses of this system, like

- Practice sign language
- Teach sign language to the children
- To communicate with the hearing community
- Test ability of the signer to perform particular signs
- It can be used in the deaf schools to perform exam of the students

The worked done so far will have a great impact on the hearing impaired community. This impact will reduce their grief and fear to communicate with the world. It will bring deaf community to the status that they will perform gestures with the thought that their message is conveyed. Therefore this system will have a fruitful impact on the deaf community especially in Pakistan as very rare work is done in this area.

CHAPTER 6: CONCLUSION

6.1 Conclusion

The main concept and objectives of the project has been achieved. It is a very big milestone to take this concept to the further stage of perfection and maturity. The further research in this area will make this system more promising and fruitful in picture that a portable device will introduced which will carried by the hearing impaired person to communicate with the normal world and feel like a common person instead of deficient one.

A lot of milestones have been achieved in classifying the sign language gestures by applying different techniques and achieving remarkably acceptable results. Using Nintendo power glove 5 signers have been used with 760 samples of each signer. On these sign gestures LDA, QDA, NB and DT have been applied.

Following is the best result for each signer for the best feature used to produce maximum accuracy.

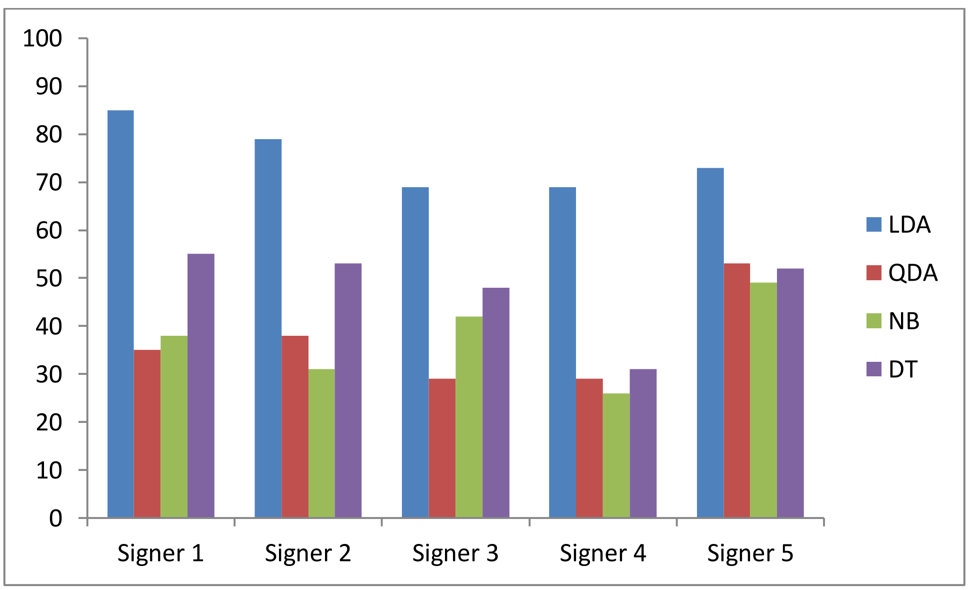


Figure 41 Results comparison of Accuracy for all Signers using Nintendo Power Glove

The result shows that LDA performs best in all the signers. We have achieved the best accuracy of 85% using Nintendo power with LDA.

Similarly for 5DT data glove we have the following results

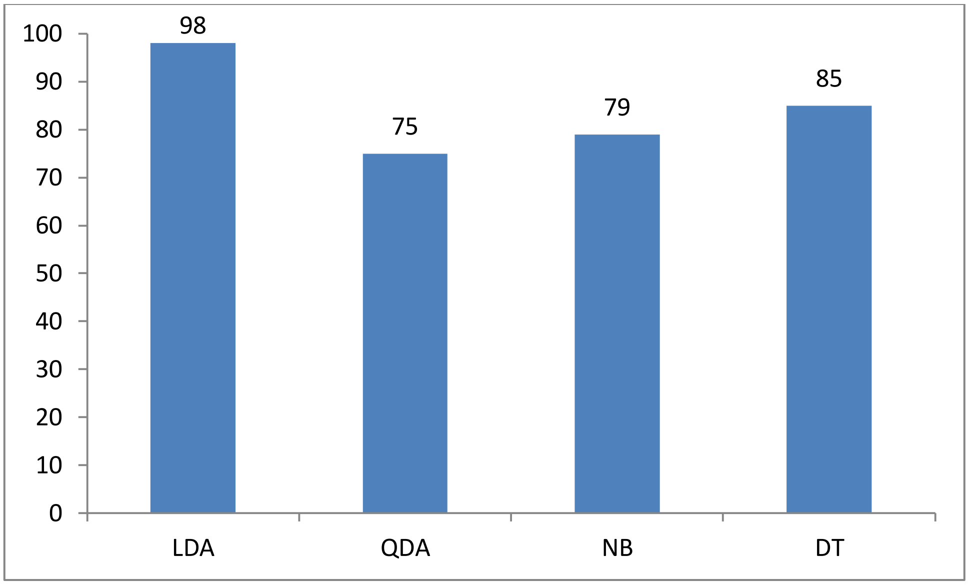


Figure 42 Best Results for 5DT data glove

Here again LDA shows the best result of 98% accuracy.

6.2 Results Comparison with the Previous Work Done

Comparison on the basis of maximum accuracy achieved, maximum number of signs used.

Table 13 Comparison of results Table

Input / Data	Year	sensors	No of signs	Accuracy	Technique Applied
AusSLAN Gestures Using 5DT Data Glove for 2 hands & Flock of Birds	2014	7	95	98	Linear Discriminant Anaysis
AusLAN using Nintendo Power Glove	2014	3	95	89	Linear Discriminant Anaysis
Gestures Using Nintendo Power Glove Glove	1995	3	95	80	IBL1

6.3 Future Work

A milestone has been achieved of maximum accuracy of 98%. More classification algorithms can be implemented to this data. A hardware based potable system can be designed to benefit the deaf community in order to facilitate them with the purpose of communication with the other entire world. The worked done in this project will be beneficial to the hearing impaired community. This impact will overcome their shyness and deficiency to communicate with the world.

REFERENCES

[1] Interpreting Gestural Input using Neural Networks. IEE Colloquium on Neural Nets in Human-Computer Interaction. IEE Digest 1990. (Russell Beale and Alistair D. N. Edwards)

[2] K. Murakami and H. Taguchi. Gesture recognition using recurrent neural networks. In Proceedings of CHI91 Human Factors in Computing Systems, 1991

[3] Weissman, J. and Salomon, R. (1999), "Gesture recognition for virtual reality applications using data gloves and neural networks", Proceedings of IEEE International Joint Conference on Neural Networks, pp. 2043-2046.

[4] Xu, D. (2006), "A neural network approach for hand gesture recognition in virtual reality driving training system of SPG", 18th International Conference on Pattern Recognition (ICPR'06), Vol. 3, pp. 519-522.

[5] Jin, S., Li, Y., Lu, G., Luo, J., Chen, W. and Zheng, X. (2011), "Interaction and control with the auxiliary of hand gesture", Proceedings of the International Conference on Information Science and Technology (ICIST 2011), Nanjing, pp. 1362-1366.

[6] Fels, S.S. and Hinton, G.E. (1993), "Glove-talk: a neural network interface between a data-glove and a speech synthesizer", IEEE Transactions on Neural Networks, Vol. 4 No. 1, pp. 2-8.

[7] Vamplew, P. and Adams, A. (1996), "Recognition of sign language gestures using neural networks", Australian Journal of Intelligent Information Processing Systems, Vol. 5 No. 2, pp. 94-102.

[8] Oz, C. and Leu, C. (2011), "American Sign Language word recognition with a sensory glove using artificial neural networks", Engineering Applications of Artificial Intelligence, Vol. 24, pp. 1204-1213.

[9] Eisenstein, J., Ghandeharizadeh, S., Golubchik, L., Shahabi, C., Jan, D. and Zimmermann, R. (2003), "Device independence and extensibility in gesture recognition", IEEE Virtual Reality Conference, pp. 207-214.

[10] Ibarguren, A., Maurtua, I. and Sierra, B. (2010), "Layered architecture for real time sign recognition: hand gesture and movement", Engineering Applications of Artificial Intelligence, Vol. 23, pp. 1216-1228.

[11] Parvini, F. and Shahabi, C. (2005), "Utilizing bio-mechanical characteristics for user-independent gesture recognition", roceedings of the 21st International Conference on Data Engineering (ICDE'05), pp. 1170-1179.

[12] Parvini, F., McLeod, D., Shahabi, C., Navai, B., Zali, B. and Ghandeharizadeh, S. (2009), "An approach to glove-based gesture recognition", in Jacko, J.A. (Ed.), Human-Computer Interaction – Part II, Springer, Berlin, pp. 236-245.

[13] Newbee, G.B. (1994), "Gesture recognition based upon statistical similarity", Presence, Vol. 3 No. 3, pp. 236-243.

[14] Eisenstein, J., Ghandeharizadeh, S., Huang, L., Shahabi, C., Shanbhag, G. and Zimmermann, R. (2001), "Utilizing biomechanical characteristics for user independent gesture recognition", Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 259-262.

[15] Shahabi, C., Kaghazian, L., Mehta, S., Ghoting, A., Shanbhag, G. and McLaughlin, M.L. (2001), "Understanding of user behavior in immersive environments", in McLaughlin, M.L., Hespanha, J. and Sukhatme, G. (Eds), Touch in Virtual Environments: Haptics and the Design of Interactive Systems, Prentice-Hall, Upper Saddle River, NJ, pp. 238-259.

[16] Xia, P.J., Lopes, A.M. and Restivo, M.T. (2013), "A review of virtual reality and haptics for product assembly (part 1): rigid parts", Assembly Automation, Vol. 33 No. 1, pp. 68-77.

[17] R. Gopalan and B. Dariush, "Towards a Vision Based Hand Gesture Interface for Robotic Grasping", The IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009, St. Louis, USA, pp. 1452-1459.

[18] T. Kapuscinski and M. Wysocki, "Hand Gesture Recognition for Man-Machine interaction", Second Workshop on Robot Motion and Control, October 18-20, 2001, pp. 91-96.

[19] D. Y. Huang, W. C. Hu, and S. H. Chang, "Vision-based Hand Gesture Recognition Using PCA+Gabor Filters and SVM", IEEE Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009, pp. 1-4.

[20] C. Yu, X. Wang, H. Huang, J. Shen, and K. Wu, "Vision-Based Hand Gesture Recognition Using Combinational Features", IEEE Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2010, pp. 543-546.

[21] J. L. Raheja, K. Das, and A. Chaudhury, "An Efficient Real Time Method of Fingertip Detection", International Conference on Trends in Industrial Measurements and automation (TIMA), 2011, pp. 447-450.

[22] Manigandan M. and I. M Jackin, "Wireless Vision based Mobile Robot control using Hand Gesture Recognition through Perceptual Color Space", IEEE International Conference on Advances in Computer Engineering, 2010, pp. 95-99.

[23] A. S. Ghotkar, R. Khatal, S. Khupase, S. Asati, and M. Hadap, "Hand Gesture Recognition for Indian Sign Language", IEEE International Conference on Computer Communication and Informatics (ICCCI), Jan. 10-12, 2012, Coimbatore, India.

[24] I. G. Incertis, J. G. G. Bermejo, and E.Z. Casanova, "Hand Gesture Recognition for Deaf People Interfacing", The 18th International Conference on Pattern Recognition (ICPR), 2006.
[25] J. Rekha, J. Bhattacharya, and S. Majumder, "Shape, Texture and Local Movement Hand Gesture Features for Indian Sign Language Recognition", IEEE, 2011, pp. 30-35.
[26] L. K. Lee, S. Y. An, and S. Y. Oh, "Robust Fingertip Extraction with Improved Skin Color Segmentation for Finger Gesture Recognition in Human-Robot Interaction", WCCI 2012 IEEE World Congress on Computational Intelligence, June, 10-15, 2012, Brisbane, Australia.
[27] S. K. Yewale and P. K. Bharne, "Hand Gesture Recognition Using Different Algorithms Based on Artificial Neural Network", IEEE, 2011, pp. 287-292.
[28] Y. Fang, K. Wang, J. Cheng, and H. Lu, "A Real-Time Hand Gesture Recognition Method", IEEE ICME, 2007, pp. 995-998.
[29] S. Saengsri, V. Niennattrakul, and C.A. Ratanamahatana, "TFRS: Thai Finger-Spelling Sign Language Recognition System", IEEE, 2012, pp. 457-462.
[30] J. H. Kim, N. D. Thang, and T. S. Kim, "3-D Hand Motion Tracking and Gesture Recognition Using a Data Glove", IEEE International Symposium on Industrial Electronics (ISIE), July 5-8, 2009, Seoul Olympic Parktel, Seoul , Korea, pp. 1013-1018.
[31] J. Weissmann and R. Salomon, "Gesture Recognition for Virtual Reality Applications Using Data Gloves and Neural Networks", IEEE, 1999, pp. 2043-2046.
[32] W. W. Kong and S. Ranganath, "Sign Language Phoneme Transcription with PCA-based Representation", The 9th International Conference on Information and Communications Security(ICICS), 2007, China.
[33] Joyeeta Singha, Karen Das, "Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique", (IJACSA) International Journal of Advanced Computer Science and Applications,Vol. 4, No. 2, 2013

ANNEXURE A: CODE

Reading File & Extracting Features

```
%********** reading and sorting file ***********
clear all
clc
count=1;
%********** making file path to read ************
folder='5DT data/tctodd';       % folder
fnum=char('1/','2/','3/','4/','5/','6/','7/','8/','9/');            % folder
number
sname=char('alive','all','answer','boy','building','buy','change_mind_','cold
','come','computer_PC_','cost','crazy','danger','deaf','different','draw','dr
ink','eat','exit','flash-
light','forget','girl','give','glove','go','God','happy','head','hear','hello
','his_hers','hot','how','hurry','hurt','I','innocent','is_true_','joke','jui
ce','know','later','lose','love','make','man','maybe','mine','money','more','
name','no','Norway','not-my-
problem','paper','pen','please','polite','question','read','ready','research'
,'responsible','right','sad','same','science','share','shop','soon','sorry','
spend','stubborn','surprise','take','temper','thank','think','tray','us','vol
untary','wait_notyet_','what','when','where','which','who','why','wild','will
','write','wrong','yes','you','zero');          % gesture
num=char('-1','-2','-3');                % sample number
ext='.TSD';
for fnm=1:9
    for sign=1:95
        for sample=1:3
            name=strcat(folder,fnum(fnm,:),sname(sign,:),num(sample,:),ext);
            %       opening file
            fid = fopen(name);
            c=1;
            while ~feof(fid)
                tline=fgets(fid);
                A=textscan(tline, '%f %f %f %f %f %f %f %f %f %f %f %f %f %f
%f %f %f %f %f %f %f %f', 'delimiter', ',','EmptyValue', -Inf);
                data(c,:)=A;
                c=c+1;
            end
            fclose(fid);
            B=cell2mat(data);    % converting cellvalues to integers
            clear data;
            [mm,nn]=size(B);   % getting size of matrix

            % ******************** no of frames ********************
            nofframes=mm;

            %*******************************************************
            % ******************** NO OF INSTANCES ********************
            for j=1:nn;
                incount =0;
                for i=2:mm;
                    if  (B(i,j)~=B(i-1,j))
                        incount=incount+1;
                    end
```

```
    end
    instcount(fnm,count,j)=incount;
end

%*******************************************************
%********* Calculating Distance ***********************
for i=1:mm-1;
    dx=B(i+1,1)-B(i,1);     % dx change in x
    dy=B(i+1,2)-B(i,2);      % dy change in y
    dz=B(i+1,3)-B(i,3);      % dz change in z
    D(i,1)=sqrt(dx^2+dy^2+dz^2);  % length or norm of vector
    e(i,1)=D(i,1)^3;       % energy of sign
end
dist=sum(D);            % total distance travelled
enrgy=sum(e);
tdist_e_noframes(fnm,count,1)=dist;
tdist_e_noframes(fnm,count,2)=enrgy;
tdist_e_noframes(fnm,count,3)=nofframes;
%**********************************************************
% ********* calculating and saving mean ********
m=mean(B);
for x=1:23
    if x==23
        means(fnm,count,x)=sign;

    elseif x~=23
        means(fnm,count,x)=m(1,x);
    end
end
%***************************************
% ********* calculating and saving variance ********
v=var(B);
for x=1:23
    if x==23
        varianse(fnm,count,x)=sign;

    elseif x~=23
        varianse(fnm,count,x)=v(1,x);
    end
end
%***************************************
% ********* calculating and saving standard variation ********
s=std(B);
for x=1:23
    if x==23
        standarddeviation(fnm,count,x)=sign;

    elseif x~=23
        standarddeviation(fnm,count,x)=s(1,x);
    end
end
%***************************************
% * calculating and saving xmin,xmax,ymin,ymax,zmin,zmax ********
m=min(B);
mx=max(B);
for x=1:4
```

```
                if x==4
                    minnmaxx(fnm,count,x+3)=sign;

                elseif x<=3
                    minnmaxx(fnm,count,x)=m(1,x);
                    t=x+11;
                    minnmaxx(fnm,count,x+3)=mx(1,x);
                end
            end
            %***************************************
            % ********* calculating and saving RMS ********
            r=rms(B);
            for x=1:23
                if x==23
                    rootms(fnm,count,x)=sign;

                elseif x~=23
                    rootms(fnm,count,x)=s(1,x);
                end
            end
            %***************************************
            %*************** Zero Crossing *************************
            for i=1:nn
                zcc=0;
                for j=2:mm
                    if( B(j,i)==0 && B(j-1,i)~=0)
                        zcc=zcc+1;
                    end
                end
                zc(fnm,count,i)=zcc;
            end
            %********************************************
        count=count+1;
        end
    end
    count=1;
end
for fnm=1:9
    if fnm==1
        c1=1;
        c2=1;
        c3=1;
        c=1;
        for i=1:285
            if c==1
                instcount1(c1,1:22)=instcount(fnm,i,1:22);
                rootms1(c1,1:22)=means(fnm,i,1:22);
                zc1(c1,:)=zc(fnm,i,:);
                minnmaxx1(c1,1:6)=minnmaxx(fnm,i,1:6);
                means1(c1,1:22)=means(fnm,i,1:22);
                tdist_e_noframes1(c1,:)=tdist_e_noframes(fnm,i,:);
                varianse1(c1,1:22)= varianse(fnm,i,1:22);
                standarddeviation1(c1,1:22)=standarddeviation(fnm,i,1:22);
                y1(c1,1)=means(fnm,i,23);
                c1=c1+1;
                c=c+1;
```

```
        elseif c==2
            instcount2(c2,1:22)=instcount(fnm,i,1:22);
            rootms2(c2,1:22)=means(fnm,i,1:22);
            zc2(c2,:)=zc(fnm,i,:);
            minnmaxx2(c2,1:6)=minnmaxx(fnm,i,1:6);
            means2(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes2(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse2(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation2(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount3(c3,1:22)=instcount(fnm,i,1:22);
            rootms3(c3,1:22)=means(fnm,i,1:22);
            zc3(c3,:)=zc(fnm,i,:);
            minnmaxx3(c3,1:6)=minnmaxx(fnm,i,1:6);
            means3(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes3(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse3(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation3(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
if fnm==2
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount4(c1,1:22)=instcount(fnm,i,1:22);
            rootms4(c1,1:22)=means(fnm,i,1:22);
            zc4(c1,:)=zc(fnm,i,:);
            minnmaxx4(c1,1:6)=minnmaxx(fnm,i,1:6);
            means4(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes4(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse4(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation4(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount5(c2,1:22)=instcount(fnm,i,1:22);
            rootms5(c2,1:22)=means(fnm,i,1:22);
            zc5(c2,:)=zc(fnm,i,:);
            minnmaxx5(c2,1:6)=minnmaxx(fnm,i,1:6);
            means5(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes5(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse5(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation5(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount6(c3,1:22)=instcount(fnm,i,1:22);
            rootms6(c3,1:22)=means(fnm,i,1:22);
```

```
            zc6(c3,:)=zc(fnm,i,:);
            minnmaxx6(c3,1:6)=minnmaxx(fnm,i,1:6);
            means6(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes6(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse6(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation6(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
 if fnm==3
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount7(c1,1:22)=instcount(fnm,i,1:22);
            rootms7(c1,1:22)=means(fnm,i,1:22);
            zc7(c1,:)=zc(fnm,i,:);
            minnmaxx7(c1,1:6)=minnmaxx(fnm,i,1:6);
            means7(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes7(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse7(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation7(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount8(c2,1:22)=instcount(fnm,i,1:22);
            rootms8(c2,1:22)=means(fnm,i,1:22);
            zc8(c2,:)=zc(fnm,i,:);
            minnmaxx8(c2,1:6)=minnmaxx(fnm,i,1:6);
            means8(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes8(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse8(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation8(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount9(c3,1:22)=instcount(fnm,i,1:22);
            rootms9(c3,1:22)=means(fnm,i,1:22);
            zc9(c3,:)=zc(fnm,i,:);
            minnmaxx9(c3,1:6)=minnmaxx(fnm,i,1:6);
            means9(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes9(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse9(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation9(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
if fnm==4
    c1=1;
    c2=1;
```

```
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount10(c1,1:22)=instcount(fnm,i,1:22);
            rootms10(c1,1:22)=means(fnm,i,1:22);
            zc10(c1,:)=zc(fnm,i,:);
            minnmaxx10(c1,1:6)=minnmaxx(fnm,i,1:6);
            means10(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes10(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse10(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation10(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount11(c2,1:22)=instcount(fnm,i,1:22);
            rootms11(c2,1:22)=means(fnm,i,1:22);
            zc11(c2,:)=zc(fnm,i,:);
            minnmaxx11(c2,1:6)=minnmaxx(fnm,i,1:6);
            means11(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes11(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse11(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation11(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount12(c3,1:22)=instcount(fnm,i,1:22);
            rootms12(c3,1:22)=means(fnm,i,1:22);
            zc12(c3,:)=zc(fnm,i,:);
            minnmaxx12(c3,1:6)=minnmaxx(fnm,i,1:6);
            means12(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes12(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse12(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation12(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
if fnm==5
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount13(c1,1:22)=instcount(fnm,i,1:22);
            rootms13(c1,1:22)=means(fnm,i,1:22);
            zc13(c1,:)=zc(fnm,i,:);
            minnmaxx13(c1,1:6)=minnmaxx(fnm,i,1:6);
            means13(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes13(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse13(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation13(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
```

```
            c=c+1;
        elseif c==2
            instcount14(c2,1:22)=instcount(fnm,i,1:22);
            rootms14(c2,1:22)=means(fnm,i,1:22);
            zc14(c2,:)=zc(fnm,i,:);
            minnmaxx14(c2,1:6)=minnmaxx(fnm,i,1:6);
            means14(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes14(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse14(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation14(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount15(c3,1:22)=instcount(fnm,i,1:22);
            rootms15(c3,1:22)=means(fnm,i,1:22);
            zc15(c3,:)=zc(fnm,i,:);
            minnmaxx15(c3,1:6)=minnmaxx(fnm,i,1:6);
            means15(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes15(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse15(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation15(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
if fnm==6
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount16(c1,1:22)=instcount(fnm,i,1:22);
            rootms16(c1,1:22)=means(fnm,i,1:22);
            zc16(c1,:)=zc(fnm,i,:);
            minnmaxx16(c1,1:6)=minnmaxx(fnm,i,1:6);
            means16(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes16(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse16(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation16(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount17(c2,1:22)=instcount(fnm,i,1:22);
            rootms17(c2,1:22)=means(fnm,i,1:22);
            zc17(c2,:)=zc(fnm,i,:);
            minnmaxx17(c2,1:6)=minnmaxx(fnm,i,1:6);
            means17(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes17(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse17(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation17(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount18(c3,1:22)=instcount(fnm,i,1:22);
```

```
            rootms18(c3,1:22)=means(fnm,i,1:22);
            zc18(c3,:)=zc(fnm,i,:);
            minnmaxx18(c3,1:6)=minnmaxx(fnm,i,1:6);
            means18(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes18(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse18(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation18(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
end
  if fnm==7
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount19(c1,1:22)=instcount(fnm,i,1:22);
            rootms19(c1,1:22)=means(fnm,i,1:22);
            zc19(c1,:)=zc(fnm,i,:);
            minnmaxx19(c1,1:6)=minnmaxx(fnm,i,1:6);
            means19(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes19(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse19(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation19(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount20(c2,1:22)=instcount(fnm,i,1:22);
            rootms20(c2,1:22)=means(fnm,i,1:22);
            zc20(c2,:)=zc(fnm,i,:);
            minnmaxx20(c2,1:6)=minnmaxx(fnm,i,1:6);
            means20(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes20(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse20(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation20(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount21(c3,1:22)=instcount(fnm,i,1:22);
            rootms21(c3,1:22)=means(fnm,i,1:22);
            zc21(c3,:)=zc(fnm,i,:);
            minnmaxx21(c3,1:6)=minnmaxx(fnm,i,1:6);
            means21(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes21(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse21(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation21(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
  end
  if fnm==8
    c1=1;
```

```
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount22(c1,1:22)=instcount(fnm,i,1:22);
            rootms22(c1,1:22)=means(fnm,i,1:22);
            zc22(c1,:)=zc(fnm,i,:);
            minnmaxx22(c1,1:6)=minnmaxx(fnm,i,1:6);
            means22(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes22(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse22(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation22(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
            c1=c1+1;
            c=c+1;
        elseif c==2
            instcount23(c2,1:22)=instcount(fnm,i,1:22);
            rootms23(c2,1:22)=means(fnm,i,1:22);
            zc23(c2,:)=zc(fnm,i,:);
            minnmaxx23(c2,1:6)=minnmaxx(fnm,i,1:6);
            means23(c2,1:22)=means(fnm,i,1:22);
            tdist_e_noframes23(c2,:)=tdist_e_noframes(fnm,i,:);
            varianse23(c2,1:22)= varianse(fnm,i,1:22);
            standarddeviation23(c2,1:22)=standarddeviation(fnm,i,1:22);
            c2=c2+1;
            c=c+1;
        elseif c==3
            instcount24(c3,1:22)=instcount(fnm,i,1:22);
            rootms24(c3,1:22)=means(fnm,i,1:22);
            zc24(c3,:)=zc(fnm,i,:);
            minnmaxx24(c3,1:6)=minnmaxx(fnm,i,1:6);
            means24(c3,1:22)=means(fnm,i,1:22);
            tdist_e_noframes24(c3,:)=tdist_e_noframes(fnm,i,:);
            varianse24(c3,1:22)= varianse(fnm,i,1:22);
            standarddeviation24(c3,1:22)=standarddeviation(fnm,i,1:22);
            c3=c3+1;
            c=1;
        end
    end
  end
if fnm==9
    c1=1;
    c2=1;
    c3=1;
    c=1;
    for i=1:285
        if c==1
            instcount25(c1,1:22)=instcount(fnm,i,1:22);
            rootms25(c1,1:22)=means(fnm,i,1:22);
            zc25(c1,:)=zc(fnm,i,:);
            minnmaxx25(c1,1:6)=minnmaxx(fnm,i,1:6);
            means25(c1,1:22)=means(fnm,i,1:22);
            tdist_e_noframes25(c1,:)=tdist_e_noframes(fnm,i,:);
            varianse25(c1,1:22)= varianse(fnm,i,1:22);
            standarddeviation25(c1,1:22)=standarddeviation(fnm,i,1:22);
            y1(c1,1)=means(fnm,i,23);
```

```
                c1=c1+1;
                c=c+1;
            elseif c==2
                instcount26(c2,1:22)=instcount(fnm,i,1:22);
                rootms26(c2,1:22)=means(fnm,i,1:22);
                zc26(c2,:)=zc(fnm,i,:);
                minnmaxx26(c2,1:6)=minnmaxx(fnm,i,1:6);
                means26(c2,1:22)=means(fnm,i,1:22);
                tdist_e_noframes26(c2,:)=tdist_e_noframes(fnm,i,:);
                varianse26(c2,1:22)= varianse(fnm,i,1:22);
                standarddeviation26(c2,1:22)=standarddeviation(fnm,i,1:22);
                c2=c2+1;
                c=c+1;
            elseif c==3
                instcount27(c3,1:22)=instcount(fnm,i,1:22);
                rootms27(c3,1:22)=means(fnm,i,1:22);
                zc27(c3,:)=zc(fnm,i,:);
                minnmaxx27(c3,1:6)=minnmaxx(fnm,i,1:6);
                means27(c3,1:22)=means(fnm,i,1:22);
                tdist_e_noframes27(c3,:)=tdist_e_noframes(fnm,i,:);
                varianse27(c3,1:22)= varianse(fnm,i,1:22);
                standarddeviation27(c3,1:22)=standarddeviation(fnm,i,1:22);
                c3=c3+1;
                c=1;
            end
        end
    end
end
tdist_e_noframesall=[ tdist_e_noframes1; tdist_e_noframes2;
tdist_e_noframes3; tdist_e_noframes4; tdist_e_noframes5; tdist_e_noframes6;
tdist_e_noframes7; tdist_e_noframes8; tdist_e_noframes9; tdist_e_noframes10;
tdist_e_noframes11; tdist_e_noframes12; tdist_e_noframes13;
tdist_e_noframes14; tdist_e_noframes15; tdist_e_noframes16;
tdist_e_noframes17; tdist_e_noframes18; tdist_e_noframes19;
tdist_e_noframes20; tdist_e_noframes21; tdist_e_noframes22;
tdist_e_noframes23; tdist_e_noframes24; tdist_e_noframes25;
tdist_e_noframes26; tdist_e_noframes27];
instcountall=[ instcount1; instcount2; instcount3; instcount4; instcount5;
instcount6; instcount7; instcount8; instcount9; instcount10; instcount11;
instcount12; instcount13; instcount14; instcount15; instcount16; instcount17;
instcount18; instcount19; instcount20; instcount21; instcount22; instcount23;
instcount24; instcount25; instcount26; instcount27];
rootmsall=[ rootms1; rootms2; rootms3; rootms4; rootms5; rootms6; rootms7;
rootms8; rootms9; rootms10; rootms11; rootms12; rootms13; rootms14; rootms15;
rootms16; rootms17; rootms18; rootms19; rootms20; rootms21; rootms22;
rootms23; rootms24; rootms25; rootms26; rootms27];
zcall=[ zc1; zc2; zc3; zc4; zc5; zc6; zc7; zc8; zc9; zc10; zc11; zc12; zc13;
zc14; zc15; zc16; zc17; zc18; zc19; zc20; zc21; zc22; zc23; zc24; zc25; zc26;
zc27];
minnmaxxall=[ minnmaxx1; minnmaxx2; minnmaxx3; minnmaxx4; minnmaxx5;
minnmaxx6; minnmaxx7; minnmaxx8; minnmaxx9; minnmaxx10; minnmaxx11;
minnmaxx12; minnmaxx13; minnmaxx14; minnmaxx15; minnmaxx16; minnmaxx17;
minnmaxx18; minnmaxx19; minnmaxx20; minnmaxx21; minnmaxx22; minnmaxx23;
minnmaxx24; minnmaxx25; minnmaxx26; minnmaxx27];
meansall=[ means1; means2; means3; means4; means5; means6; means7; means8;
means9; means10; means11; means12; means13; means14; means15; means16;
```

```
means17; means18; means19; means20; means21; means22; means23; means24;
means25; means26; means27];
varianseall=[ varianse1; varianse2; varianse3; varianse4; varianse5;
varianse6; varianse7; varianse8; varianse9; varianse10; varianse11;
varianse12; varianse13; varianse14; varianse15; varianse16; varianse17;
varianse18; varianse19; varianse20; varianse21; varianse22; varianse23;
varianse24; varianse25; varianse26; varianse27];
standarddeviationall=[ standarddeviation1; standarddeviation2;
standarddeviation3; standarddeviation4; standarddeviation5;
standarddeviation6; standarddeviation7; standarddeviation8;
standarddeviation9; standarddeviation10; standarddeviation11;
standarddeviation12; standarddeviation13; standarddeviation14;
standarddeviation15; standarddeviation16; standarddeviation17;
standarddeviation18; standarddeviation19; standarddeviation20;
standarddeviation21; standarddeviation22; standarddeviation23;
standarddeviation24; standarddeviation25; standarddeviation26;
standarddeviation27];
yall=[ y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1; y1;
y1; y1; y1; y1; y1; y1; y1; y1; y1; y1];
```

Applying Classification and Saving results

```
% ***************************************************************************
%****************************************************************************
% *****************    TRAIING AND TESTING    ****************************
% ****************        FEATURE MEAN        ****************************
N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=meansall;
xte=meansall;
ytr=yall;
yte=yall;

% ***************** LDA *******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(1,1)=ldaCVErr;

% ***************** QDA *******************************************

N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=horzcat(meansall(:,1:6),meansall(:,12:17));
xte=horzcat(meansall(:,1:6),meansall(:,12:17));
ytr=yall;
yte=yall;

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(1,2)=ldaCVErr;

%*********************************************************************
```

```matlab
% ********************** NAIVE BAYES (Guassian) *********************

xtr=horzcat(meansall(:,1:6),meansall(:,12:17));
xte=horzcat(meansall(:,1:6),meansall(:,12:17));
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(1,3)=nbGauCVErr;

% ******************* NAIVE BAYES (KERNAL) **************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(1,4)=nbKDCVErr;

%************************ Decision Trees ***************************

xtr=meansall;
xte=meansall;
ytr=yall;
yte=yall;
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(1,5)=dtCVErr;

%*******************************************************************
%**********   FEATURE : STANDARD DEVIATION ***************************

N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=standarddeviationall;
xte=standarddeviationall;
ytr=yall;
yte=yall;

% ***************** LDA ******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(2,1)=ldaCVErr;

% ***************** QDA *******************************************

xtr=horzcat(standarddeviationall(:,1:6),standarddeviationall(:,12:17));
xte=horzcat(standarddeviationall(:,1:6),standarddeviationall(:,12:17));
```

```
ytr=yall;
yte=yall;
F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(2,2)=ldaCVErr;

%**********************************************************************
% ********************** NAIVE BAYES (Guassian) *********************

xtr=horzcat(standarddeviationall(:,1:6),standarddeviationall(:,12:17));
xte=horzcat(standarddeviationall(:,1:6),standarddeviationall(:,12:17));
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(2,3)=nbGauCVErr;

% ******************** NAIVE BAYES (KERNAL) **************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(2,4)=nbKDCVErr;

%************************ Decision Trees ***************************
xtr=standarddeviationall;
xte=standarddeviationall;
ytr=yall;
yte=yall;
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(2,5)=dtCVErr;

%**********************************************************************
%**********   FEATURE : VARIANCE ***************************

N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=varianseall;
xte=varianseall;
ytr=yall;
yte=yall;

% ***************** LDA ******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
```

```
error(3,1)=ldaCVErr;

% ***************** QDA *******************************************
xtr=horzcat(varianseall(:,1:6),varianseall(:,12:17));
xte=horzcat(varianseall(:,1:6),varianseall(:,12:17));
ytr=yall;
yte=yall;

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(3,2)=ldaCVErr;

%**********************************************************************
% ********************** NAIVE BAYES (Guassian) *********************

xtr=horzcat(varianseall(:,1:6),varianseall(:,12:17));
xte=horzcat(varianseall(:,1:6),varianseall(:,12:17));
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(3,3)=nbGauCVErr;

% ******************* NAIVE BAYES (KERNAL) **************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(3,4)=nbKDCVErr;

%************************ Decision Trees ***************************
xtr=varianseall;
xte=varianseall;
ytr=yall;
yte=yall;
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(3,5)=dtCVErr;

%**********************************************************************
%**********   FEATURE : DISTANCE, ENERY, NO OF FRAMES *****************

N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=tdist_e_noframesall;
xte=tdist_e_noframesall;
ytr=yall;
yte=yall;
```

```
% ****************** LDA ******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(4,1)=ldaCVErr;

% ****************** QDA *******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(4,2)=ldaCVErr;

%**********************************************************************
% *********************** NAIVE BAYES (Guassian) *********************

xtr=tdist_e_noframesall;
xte=tdist_e_noframesall;
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(4,3)=nbGauCVErr;

% ******************** NAIVE BAYES (KERNAL) ***************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(4,4)=nbKDCVErr;

%************************* Decision Trees ****************************

dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(4,5)=dtCVErr;

%**********************************************************************
% ************* rooot mean square value **************
mst=rootmsall;
% concatenating means and standard deviation columns to new mst array
N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=mst;
xte=mst;
ytr=yall;
```

```
yte=yall;

% ***************** LDA ******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(5,1)=ldaCVErr;


%*****************************************************************
%********************* DT ****************************************
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(5,5)=dtCVErr;

% ***************** QDA ******************************************
mst=horzcat(rootmsall(:,1:6),rootmsall(:,12:17));

xtr=mst;
xte=mst;
F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(5,2)=ldaCVErr;

%*****************************************************************
% ********************** NAIVE BAYES (Guassian) ********************

xtr=mst;
xte=mst;
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(5,3)=nbGauCVErr;

% ******************* NAIVE BAYES (KERNAL) *************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(5,4)=nbKDCVErr;
%*****************************************************************
% ************* Bounding Box *************
mst=minnmaxxall;
N = size(yall,1);
cp = cvpartition(yall,'k',10);
```

```
xtr=mst;
xte=mst;
ytr=yall;
yte=yall;

% ****************** LDA *******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(6,1)=ldaCVErr;


%*********************************************************************
%********************** DT ******************************************
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(6,5)=dtCVErr;

% ****************** QDA ********************************************

xtr=mst;
xte=mst;
F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(6,2)=ldaCVErr;

%*********************************************************************
% *********************** NAIVE BAYES (Guassian) *********************

xtr=mst;
xte=mst;
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(6,3)=nbGauCVErr;
%*********************************************************************
% ******************* NAIVE BAYES (KERNAL) ***************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(6,4)=nbKDCVErr;
%*********************************************************************
% ************* instant count **************
mst=instcountall;
% concatenating means and standard deviation columns to new mst array
```

```
N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=mst;
xte=mst;
ytr=yall;
yte=yall;

% ****************** LDA ********************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(7,1)=ldaCVErr;


%**********************************************************************
%********************** DT ****************************************
dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(7,5)=dtCVErr;

% ****************** QDA ********************************************
mst=horzcat(instcountall(:,1:6),instcountall(:,12:17));

xtr=mst;
xte=mst;
F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(7,2)=ldaCVErr;

%**********************************************************************
% ********************** NAIVE BAYES (Guassian) *********************

xtr=mst;
xte=mst;
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)

error(7,3)=nbGauCVErr;
%**********************************************************************
% ******************** NAIVE BAYES (KERNAL) *************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(7,4)=nbKDCVErr;
%**********************************************************************
```

```
% *** MEAN , STD DVT , VArIANSE ********************************
clear mstvtdst
mstvtdst=horzcat(standarddeviationall,varianseall,meansall);
% concatenating means and standard deviation columns to new mst array
N = size(yall,1);
cp = cvpartition(yall,'k',10);
xtr=mstvtdst;
xte=mstvtdst;
ytr=yall;
yte=yall;

% f = @(xtr,ytr,xte,yte)confusionmat(yte,...
% classify(xte,xtr,ytr),'order',order);
% cfMat = crossval(f,xte,yte,'partition',cp);
% fMat = reshape(sum(cfMat),95,95)

% ****************** LDA *******************************************

F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'linear'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(8,1)=ldaCVErr;


%**********************************************************************
%*********************** DT *******************************************

dtClassFun = @(xtr,ytr,xte)(eval(classregtree(xtr,ytr),xte));
dtCVErr  = crossval('mcr',xte,yte, ...
    'predfun', dtClassFun,'partition',cp)

error(8,5)=dtCVErr;

%**********************************************************************
% ****************** QDA ********************************************
mstvtdst1=horzcat(meansall(:,12:17),standarddeviationall(:,12:17),varianseall
(:,12:17));
xtr=mstvtdst1;
xte=mstvtdst1;
F= @(xtr,ytr,xte)(classify(xte,xtr,ytr,'quadratic'));
ldaCVErr  = crossval('mcr',xtr,ytr,'predfun', ...
    F,'partition',cp)
error(8,2)=ldaCVErr;

%**********************************************************************
% *********************** NAIVE BAYES (Guassian) *********************
xtr=mstvtdst1;
xte=mstvtdst1;
ytr=yall;
yte=yall;
nbGauClassFun = @(xtr,ytr,xte)...
    (predict(NaiveBayes.fit(xtr,ytr), xte));
nbGauCVErr  = crossval('mcr',xtr,ytr,...
    'predfun', nbGauClassFun,'partition',cp)
```

```
error(8,3)=nbGauCVErr;

% ******************* NAIVE BAYES (KERNAL) *************************
nbKDClassFun = @(xtr,ytr,xte)...
            (predict(NaiveBayes.fit(xtr,ytr,'dist','kernel'),xte));
nbKDCVErr = crossval('mcr',xte,yte,...
            'predfun', nbKDClassFun,'partition',cp)

error(8,4)=nbKDCVErr;
%***********************************************************************
% *********************  ACCURACY  *******************************
error=100-(error*100);

%**************************************** END ****************************
```