PROSTHETIC VOICE

PROJECT REPORT

Submitted by

K PRATHIK	13EE139
SANJEEV N	13EE144
K SHASHANK HEGDE	13EE224

Under the guidance of

Dr. C M C KRISHNAN
Assistant Professor

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL
SRINIVASNAGAR, MANGALORE 575025
KARNATAKA, INDIA
APRIL 2017

DECLARATION

We hereby declare that the Project work entitled Prosthetic Voice, taken up as a Major Project to the Department of Electrical and Electronics Engineering is a record of original work done by our team under the guidance of: Dr C.M.C Krishnan, Assistant Professor Department of Electrical and Electronics Engineering, NITK Surathkal.

All the material and information included in the report and analysis have been duly acknowledged.

The results embodied in this study have not been submitted to any other University or Institute for the award of any degree.

Date – 25th April, 2017

Name of Team Members:
K Prathik 13EE139
Sanjeev N 13EE144
K Shashank Hegde 13EE224,

Certificate of Completion

This is to certify that the project report titled “Prosthetic Voice” is a bonafide work carried out by:-

K Prathik 13EE139
Sanjeev N 13EE144
K Shashank Hegde 13EE224,

the students of VIII semester, B.Tech Electrical and Electronics Engineering, National Institute of Technology Karnataka Surathkal. This project has been completed under the guidance of Dr. C.M.C Krishnan, Assistant Professor, NITK Surathkal, during the academic year 2016-17 as Major Project - 2 (EE499).

Dr. C M C Krishnan
Assistant Professor,
Department of Electrical and Electronics Engineering
NITK

CONTENTS

TOPIC PAGE

1. Abstract 5
2. Introduction 5
3. Forearm Muscles 5
4. Monitoring Muscular Activity 7
5. Methodology 9
6. Signal Conditioning 11
7. Feature Extraction 14
8. Principal Component Analysis 16
9. WEKA Attribute Selector 17
10. Correlation Based Feature Selection 18
11. Classification 18
12. Results and Discussions 20
13. Future Work and Scope for Improvement 23
14. Conclusion 23
15. References 24

Abstract

Deaf-mute individuals have an arduous task in communicating to a normal individual if that person is unaware of sign language. This project aims to develop a product that will convert the signs shown by the deaf-mute individual to an artificially generated voice output that maps onto the respective sign shown through the phonetics in the message(s) conveyed. Here, SEMG signals are extracted to provide the basis for mapping to the phonetics. Electromyographic signals have been broadly employed as a means of diagnosis and as a control signal in rehabilitation in health care. Commercial software such as Matlab and CoolTerm/Arduino were used for data acquisition and data analysis.

Introduction

EMG is a nerve conducting test and is accomplished by determining the bioelectric signals from the muscles of a human body. The muscular movements cause the action of muscles and nerves, which is responsible for electrical currents. These currents are created by the interchange of ions through the muscles, which enables the part of the signalling process for the muscle fibres to contract [2]. It can be calculated by attaching electrodes or conductive elements to the skin surface, or within the muscle, invasively. The measurement of surface EMG relies on the amplitude of the surface EMG signal. The signal usually varies within the range of μV to mV. The signal level is too small to capture on the display, hence it is required to amplify the signal level to a TTL level (-5 volts to +5 volts). Critical factors such as noise and other artefacts should be considered before the signal is displayed properly. An additional DC current could also offer offset to the EMG signal. A proper ground reference is required without which the signal acquired could be misleading [1].

Forearm Muscles

The forearm is a part on the upper limb between the wrist and the elbow. The forearm comprises of two bones, the ulna and radius. It encompasses muscles such as the extensor carpi radialis, extensor digitorum communis, extensor carpi ulnaris and a few more. Below we have listed the muscles of interest that we have targeted for classification of actions.

Extensor carpi radialis

Among the five main muscles that control movements at the wrist, the extensor carpi radialis is one of them. This muscle starts on the lateral side of the humerus, attaching to the base of the second metacarpal bone and is known to be quite long. This muscle is an extensor at the wrist joint and moves along the arm's radial side, so will also abduct (radial abduction) the hand at the wrist, as the name proposes.

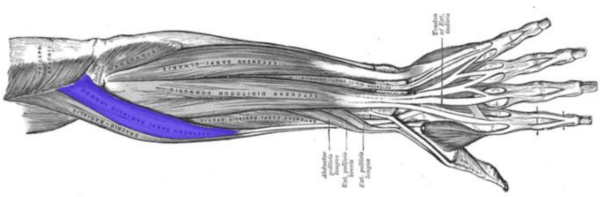


Fig 1: Extensor carpi radialis

Extensor digitorum communis

In humans and other animals, the extensor digitorum muscle (also known as extensor digitorum communis) is a muscle of the posterior forearm. It encompasses the medial four fingers of the hand. This muscle extends the phalanges, then the wrist, and finally the elbow. It tends to differentiate the digits as it extends them.

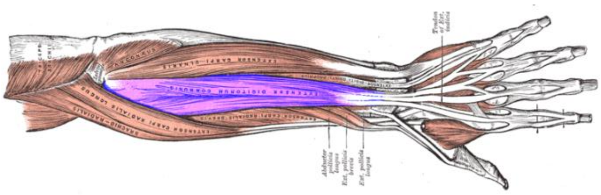


Fig 2: Extensor digitorum communis

Extensor carpi ulnaris

This is a skeletal muscle found on the ulnar side of the forearm. It is aimed to extend and adduct at the carpus/wrist from structural position. The extensor carpi ulnaris helps in extension of wrist, but when acting single-handedly, it inclines the forehand toward the ulnar side; by its continuous action it extends the elbow-joint.

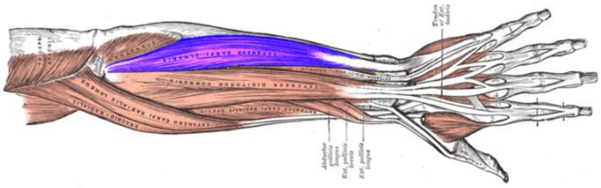


Fig 3: Extensor carpi ulnaris

Flexor carpi radialis

Flexor carpi radialis is a muscle of the forearm that aids to (radial) adduct and flex the hand [1].

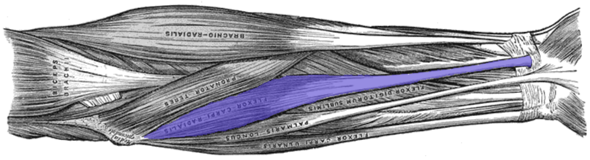


Fig 4: Flexor carpi radialis

Monitoring Muscle Activity

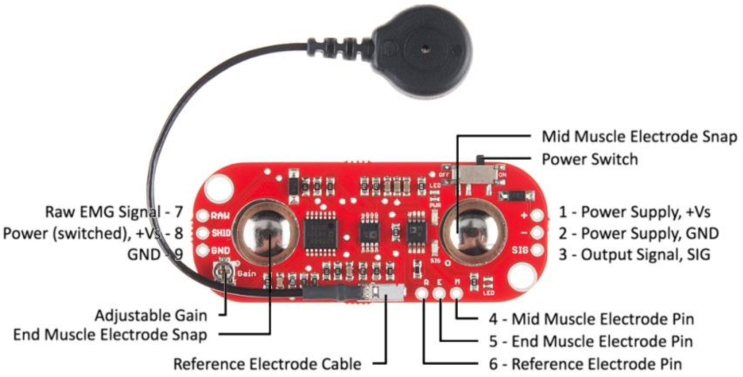

Mid Muscle Electrode Snap
Power Switch
Raw EMG Signal - 7
Power (switched), +Vs - 8
GND - 9
1 - Power Supply, +Vs
2 - Power Supply, GND
3 - Output Signal, SIG
Adjustable Gain
End Muscle Electrode Snap
Reference Electrode Cable
4 - Mid Muscle Electrode Pin
5 - End Muscle Electrode Pin
6 - Reference Electrode Pin

Fig 5: MyoWare Muscle Sensor

The MyoWare muscle sensor from Advancer Technologies measures the activity of the muscles by observing the electric potential produced by muscle cells. The device amplifies and processes the complex electrical muscular activity and converts it into a simple analog signal which can easily be read by any microcontroller, such as an A-Star or Arduino – or even a Maestro servo controller, with an analog-to-digital converter. When the target muscle group flexes, there will an increase sensor’s output voltage. The relationship between the muscle activity and the output voltage can be fine-tuned by using the on-board gain potentiometer. The sensor, in order to attach to the skin, requires three electrodes that snap into the sensor’s snap-style connectors, this makes it easy to attach or detach electrodes. Two connectors are placed directly on the PCB, and the third is found at the end of the attached reference electrode cable. The board’s pins contain a 0.1″ pitch and work with 0.1″female headers and 0.1″ male headers [4].The connection pattern to extract the EMG signal is shown below.

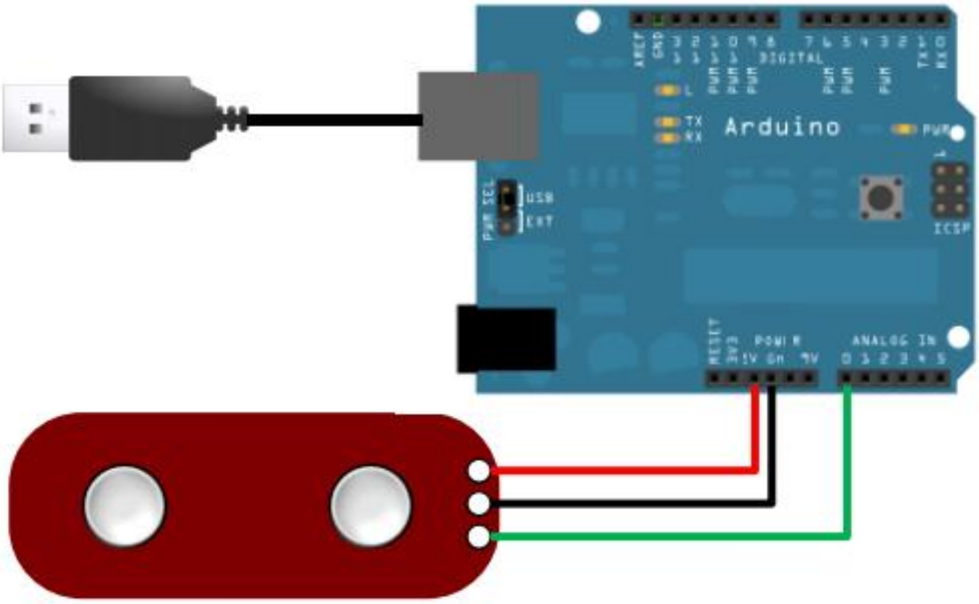


Fig 6: Circuit Diagram

An electrolytic gel is used as a chemical interface between the metallic part of the Gelled electrodes and the skin. Oxidative and reductive chemical reactions take place between the contact regions. The metallic layer lets the current from the muscle to pass more freely across the interface between the electrode and the electrolyte. This presents less electrical noise into the measurement, as paralleled with equivalent metallic electrodes (e.g. Ag) [5]. These gelled electrodes can either be reusable or disposable. Disposable electrodes are very light, hence most commonly used. They come in a wide diversity of sizes and shapes, and the materials comprising the patch and the form of the conductive gel varies between manufacturers. With an appropriate application, disposable electrodes help in minimizing the risk of electrode displacement in spite of rapid movements.

Position and orientation of the muscle sensor electrodes has a vast effect on the strength of the signal. The electrodes should be place in the middle of the muscle body and should be aligned with the orientation of the muscle fibres. Attaching the sensor in other places might reduce the quality and strength of the sensor's signal because of a decrease in the amount of motor units calculated and interference credited to crosstalk [4].

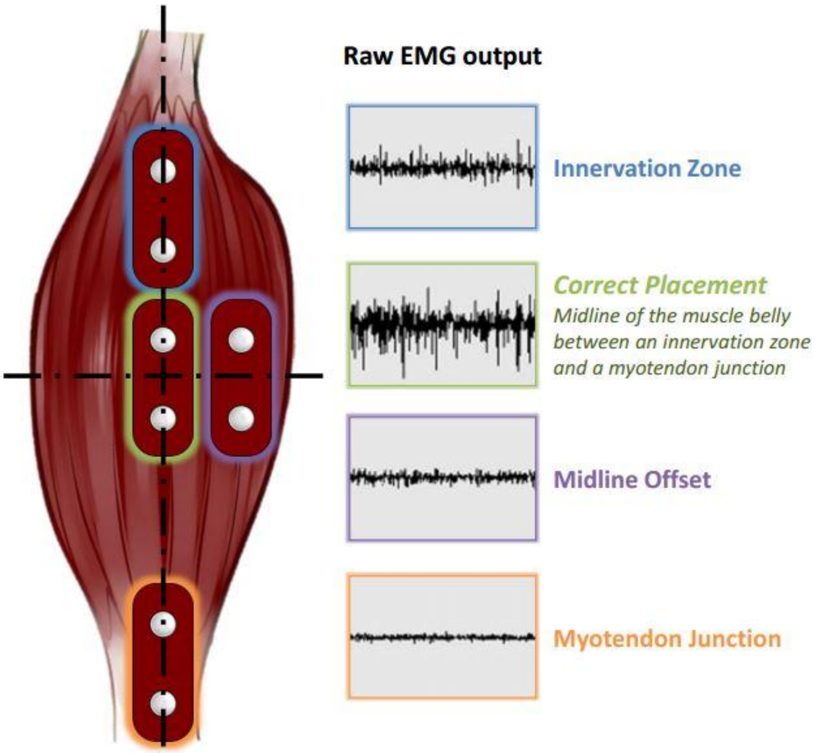
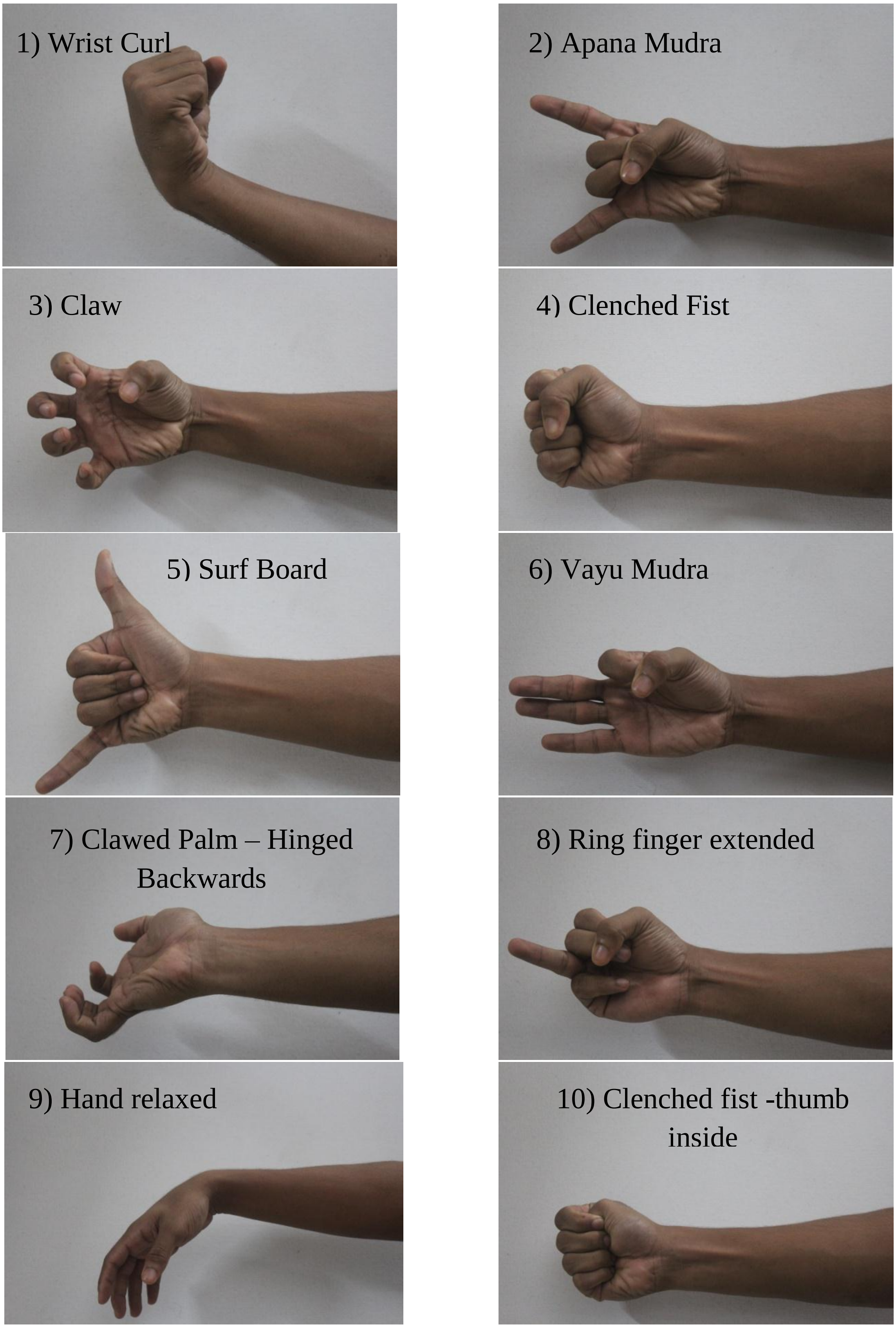


Fig 7: Electrode placement

Methodology

To start building a data set, it is necessary to have the building blocks of the same ready before extracting. In order to do this, 10 different hand signs are considered.



1) Wrist Curl

2) Apana Mudra

3) Claw

4) Clenched Fist

5) Surf Board

6) Vayu Mudra

7) Clawed Palm – Hinged Backwards

8) Ring finger extended

9) Hand relaxed

10) Clenched fist -thumb inside

Fig 8: Hand signs

As the aim of this project is to enable deaf-mute individuals to communicate via sign language, the selected actions may use any muscle required.

The sensors were placed on 3 locations – one each on the dorsal and ventral side of the fore arm and the third, place on the bicep of the same arm. These locations were chosen in accordance with the hand signs selected, as maximum muscle activities are observed at these points. Further, the locations of the sensors were strictly kept towards the elbow due to 2 main reasons. First, the muscle bellies are located towards end of the fore arm. As a result, tapping of signals are much easier than the tapering end towards the wrist, where it is difficult to gauge and pin point the muscle whose activities might overlap with neighbouring ones. Second, each MyoWare sensor has a reference electrode which is ideally placed at a bony area. In order to give a common body ground, the reference electrodes are placed around the elbow joint.

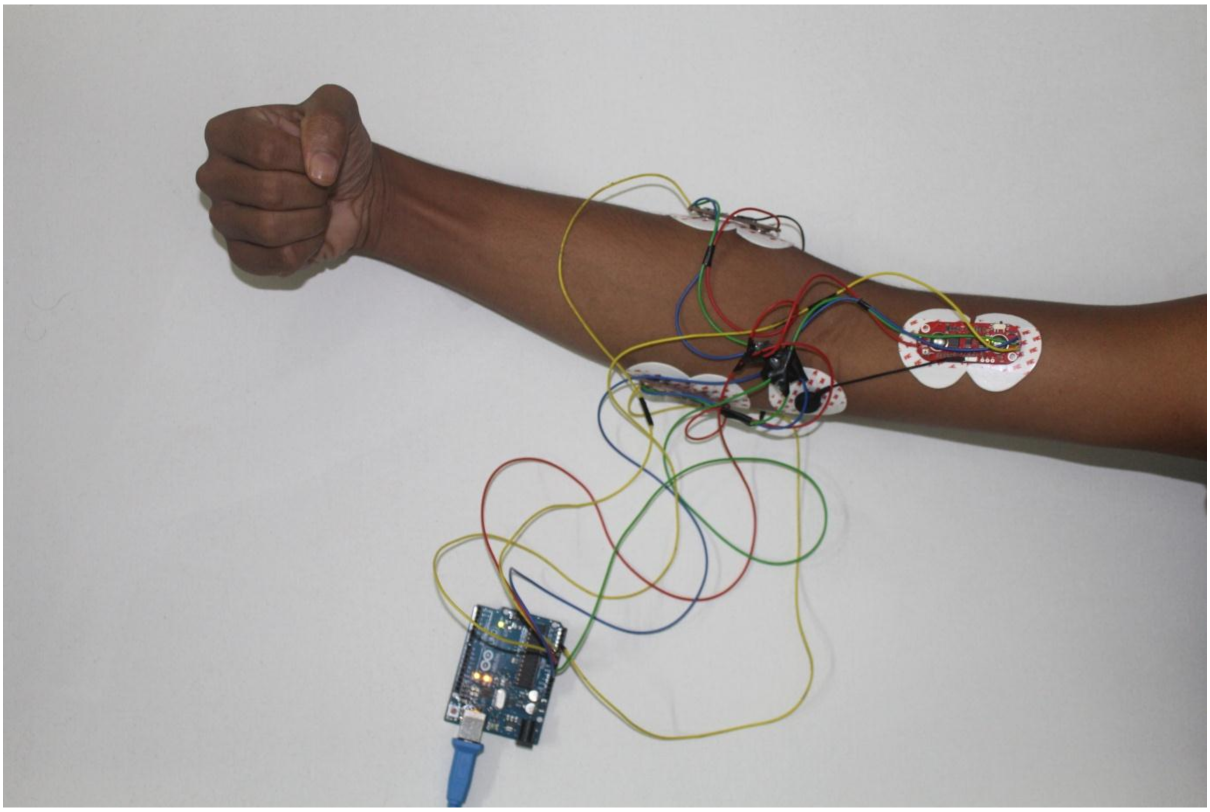


Fig 9: Final set up with all electrodes

Multi stranded wires were soldered to the MyoWare and the output from the same were fed to Arduino Uno's analog pins. Data extraction was done though CoolTerm, an open source software that publishes serial monitor values to a text file (.asc format). Each of the actions was performed in a trial for 65 seconds – alternating between rest and flexion for a period of 5 seconds. Signals were taken in the above mentioned way for 13 individuals – 8 males and 5 females, each of them having given a signed consent to participate in the research activity. The actions were taken in such a way that the transition between rest and flexion was minimal. Also, the first 15 seconds of the signal collection was taken as a rest period in order to avoid transients when the Arduino reads the signals through its Analog Input pins.

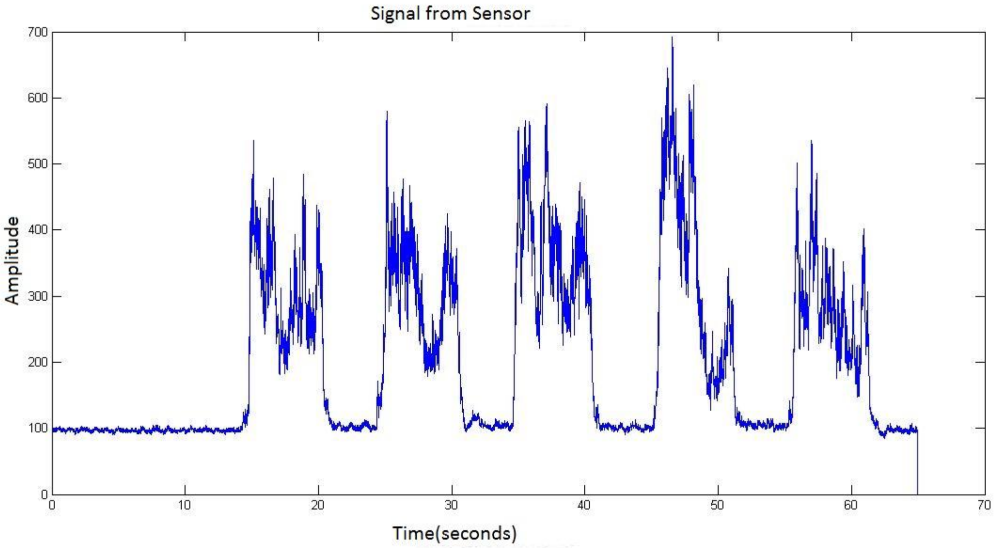


Fig 10: Rectified EMG Output for 5 action repetitions over 65 seconds**.**

9 sounds were recorded and programmed to be triggered when the corresponding action was predicted.

Signal Conditioning

Having collected the signals from the test subjects, the data was concatenated into a matrix (subject vs. data set). In order to process this data suitable for classification, noise was removed as a first step. The entire data set was passed through a fourth order FIR filter of coefficients: 0.25, 0.25, 0.25, 0.25, and then a 30th order Kaiser filter with cut-off frequency as 60Hz and 150Hz.

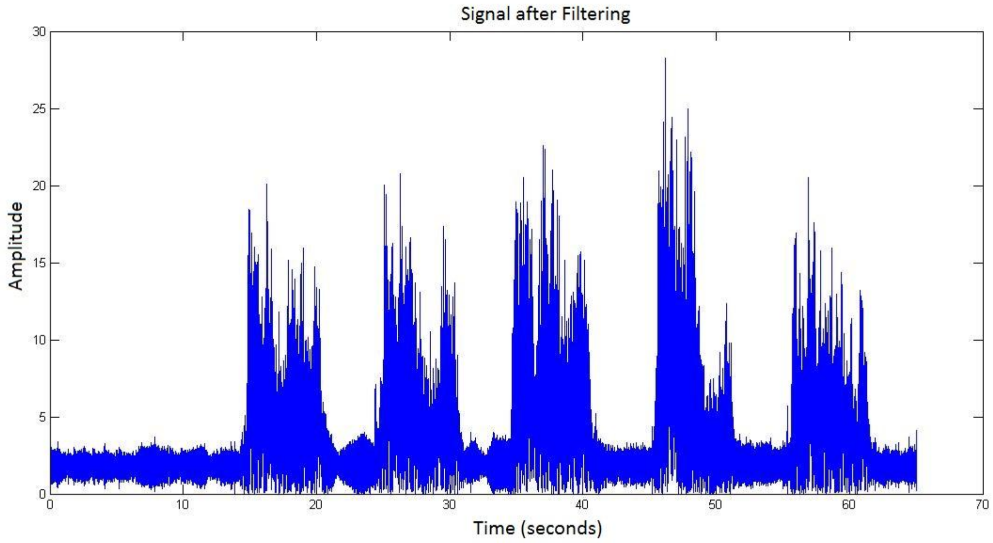


Fig 12: Signal post filtering

The next part of the algorithm was signal detection. The conditioned signal, having a total of 15 transients in all, need to be split into individual transients, minimising the noise to zero. This gives a binary like input for the classification algorithm, high when signal is detected, low when it encounters a zero. In order to detect this signal, a custom algorithm was employed. On observing these signals, it was seen that the signal during muscle action had more energy than that when the hand was at rest. Therefore a dynamic threshold algorithm would help separate the signal. After a few trails it was found that using 15 times mean gave a favourable result. Hence, a window of desired size is considered which runs through the entire signal, split for each sensor (de-multiplexing the data set).

- Consider a window of desired length
- Window runs from start through end of signal
- Compute mean of each window,

  $$\overline{\mathrm{X}} = \frac{1}{\mathrm{n}}\Sigma\big(\mathrm{x}(\mathrm{n})\big) \tag{1}$$

  and choose the sensor that has the maximum mean
- Compute energy of each window,

  $$E = (x(n))^2 \tag{2}$$

- If $\mathrm{E}_{\mathrm{win}}$> 15 $\bar{X}_{\mathrm{win}}$, cut the signal at the start of the window, through the next 5 seconds

Run this process for all such transients. Store this new set in another matrix. The data set was ready for feature extraction.

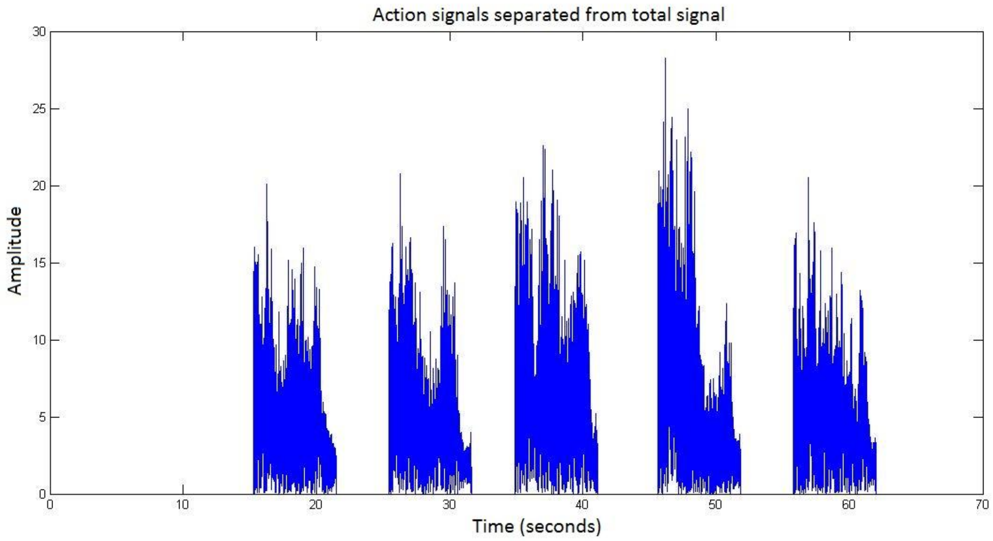


Fig 13: Signal after removing no-action noise

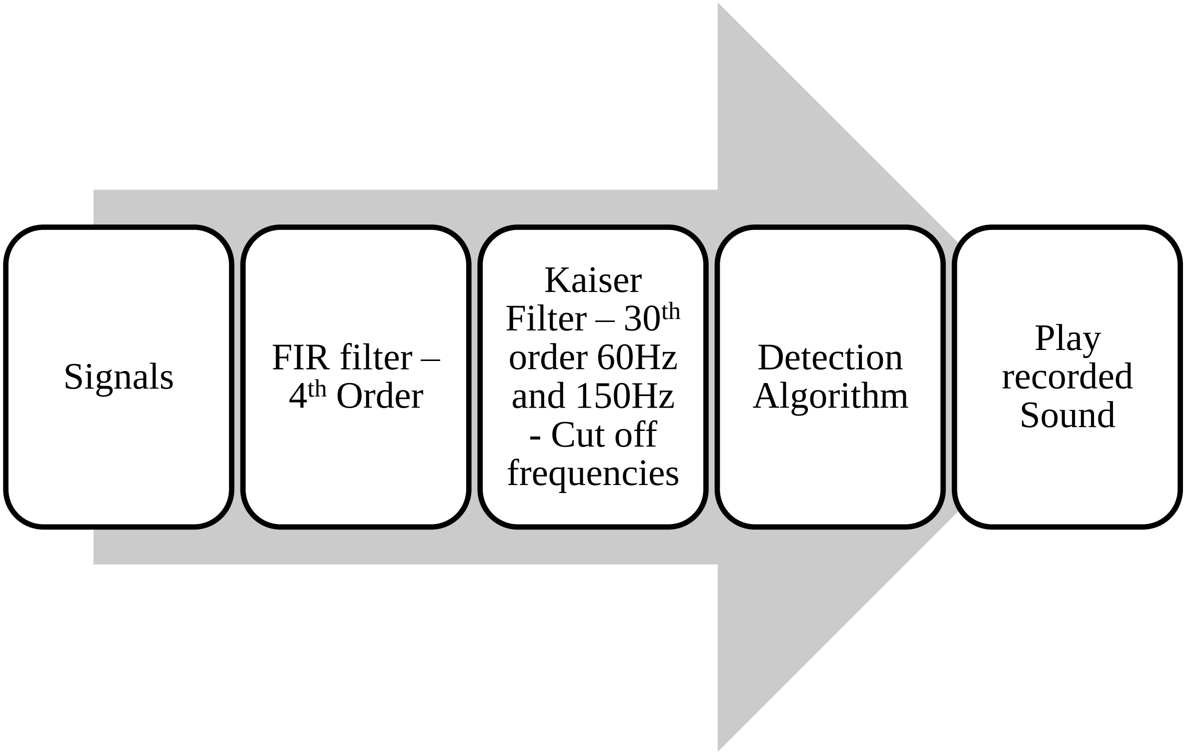


Fig 14: Process Flowchart

Feature Extraction

In machine learning and pattern recognition, extraction of features begins with an initial set of measured data and builds derived values (features) projected to be informative and non-redundant, easing the subsequent learning and simplification steps, and in some cases prominent to human interpretations. Extraction of features includes reducing the amount of attributes required to pronounce a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved.

In this project, time domain, frequency domain and several other parameters were chosen to extract information from the refined data set collected. Most of the features so chosen have two other parallel versions for each sensor.

Time domain features have been extensively used in engineering and medical practices and researches. They are used in classification of signals due to its quick and easy implementation. Additionally, it does not require any transformation of the features that are calculated based on raw EMG time series. The EMG signal's non-stationary property, changing in statistical properties over time arises to be a disadvantage for the features in time domain, but they assume the data as a stationary signal. Nonetheless, compared to frequency domain and time-frequency domain, features in time domain have been commonly used due to their performances of signal classification in low noise environments and their lower complexity in computation.

They are as follows:

1. **Mean**: The means of each sensor data were computed.
2. **RMS**: The Root Mean Squares of each sensor were computed. RMS is determined as the square root of the mean over time of the square of the vertical distance of the graph from the rest state, related to the constant force and non-fatiguing contraction of the muscle.
3. **Variance**: Variance represents the extent of fluctuation of a signal from its mean. Variance is a property widely used especially when contractions of muscles are extremely strong and visible changes in signal patterns are observed[7].. Variance uses power of a signal as a feature. Since variance is the mean value of the square of that variable, it can be computed as

$$\mathrm{Var} = \frac{1}{N-1}\sum_{n=1}^{N} \mathrm{x}_{\mathrm{n}}^{2} \tag{3}$$

4. **Power**: The power of a signal is the sum of the absolute squares of its time-domain samples divided by the signal length, or, equivalently, the square of its RMS level.
5. **Amplitude**: Maximum amplitude (MAX) is defined as the peak amplitude of a signal. It is often used in areas where the measured signal is not sinusoidal, where the signal swings above and below a zero value [11].

6. **Spectral roll-off**: The roll-off is a measure of spectral shape useful for distinguishing actual signal from noise that is a part of the signal. The frequency below which 85% of the magnitude distribution of the spectrum is concentrated is known as Roll-Off [15]. That is, if K is the largest bin that fulfils,

$$\sum_{k=1}^{N/2} |\mathrm{X_r(k)}| \leq 0.85 \sum_{k=1}^{N/2} |\mathrm{X_r(k)}| \tag{4}$$

7. **Spectral centroid**: Centroid is the gravity of the spectrum, where the sign function is defined by

$$\mathrm{C_r} = \frac{\sum_{k=1}^{N/2} \mathrm{f(k)|X_r(k)|}}{\sum_{k=1}^{N/2} \mathrm{|X_r(k)|}}, \tag{5}$$

   where N is a number of FFT points, X_r [k] is the STFT of frame x_r, and f [k] is a frequency at bin k. Centroid models the sound sharpness. Sharpness is related to the high frequency content of the spectrum [15].

8. **Mean frequency**: It is an average frequency which is calculated as the sum of product of the EMG power spectrum and the frequency divided by the total sum of the power spectrum.

$$\mathrm{MNF}=\frac{\sum_{j=1}^{\mathrm{M}} \mathrm{f_j P_j}}{\sum_{j=1}^{M} \mathrm{P_j}}, \tag{6}$$

   where Pj is the EMG power spectrum at the frequency bin j, fj is the frequency value of EMG power spectrum at the frequency bin j and M is the length of frequency bin. In the analysis of EMG signal, M is usually defined as the next power of 2 from the length of EMG data in time-domain [10].

9. **Spectral Deformation**: The **Ω** ratio is sensitive to changes in spectral symmetry and provides an indication of spectral deformation. It is computed as

$$\Omega = \frac{\sqrt{\mathrm{M_{2/M_0}}}}{\mathrm{M_1/M_0}}, \tag{7}$$

   where M_n is the n^{th} spectral moment defined as

$$M_n = \sum_{i=0}^{\mathrm{i_{max}}} \mathrm{P_i\, f_i^n}, \tag{8}$$

   where P_i is the power spectral density value at frequency f_i [9].

10. **4^{th} order polynomial regression coefficients for normalised data set**: The polynomial $ax^3 + bx^2 + cx +$ d was used to fit the data set in the above curve. These four coefficients, a, b, c, d were used as features of each sensor.

11. **2^{nd} order exponential curve fit coefficients**: The polynomial $ae^{bx} + ce^{dx}$ was used to fit the data set in the above curve. These four coefficients, a, b, c, d were used as features of each sensor.

12. **1^{st} order Linear Predictive Coefficients**: This model is a powerful tool in representing the spectral envelop of a digital signal in a compressed form. The principal objective of LP analysis is to compute the LP coefficients which minimizes the prediction error e (n).The estimate of current sample as a linear combination of past *p* samples establishes the basis of linear prediction analysis where *p* is the order

of prediction. The standard method for calculating the LP coefficients is by least squares auto correlation method. This is realised by minimizing the total prediction error.

13. **Maximum to Minimum drop in Power Density Ratio**: An estimation of the power spectral density (PSD) of noise is a crucial part to retrieve signal in a noisy environment. The DPR is defined as the ratio of the highest mean power density to the lowest mean power density between 50 and 250 Hz. The mean power density is calculated by averaging 13 consecutive points in the EMG power spectrum. The DPR is helpful in indicating whether the power spectrum was adequately peaked within the anticipated frequency range and can detect the absence of EMG activity. For higher frequencies it guarantees that the EMG spectrum drops off, which would enable the detection of high frequency noise and aliasing under sampling [6].
14. **Height of test subjects**: Height of each test subject was measured in centimetre.
15. **Weight of test subjects**: Weight of each test subject was measured in kilogram.
16. **Circumference of relaxed forearm of test subjects**: Measured in centimetre.
17. **Circumference of flexed bicep of test subjects**: Measured in centimetre.

Principal Component Analysis (PCA)

PCA is used to find a linear transformation,

$$\bar{y} = W^T \bar{x}, \tag{9}$$

where $\bar{x} \in \Re^m$ and $\bar{y} \in \Re^n$and $m > n$, in such a way that in the projected space, the variance of the data is maximized. It is an unsupervised transformation which does not require labelled training data for finding the transformation. Mathematically, PCA is a transformation which diagonalizes the covariance matrix of the global data set. While m is the original dimensionality of the feature space, n is the dimension of the desired projected space and is generally defined as the number of significant Eigenvalues in the spectral decomposition of the global covariance matrix.

A general approach to the PCA is to first solve the characteristic polynomial equation for all Eigenvalues and then find their corresponding eigenvectors to produce principal components (PCs) in accordance with descending order of Eigenvalues.

For the given data set, the various *n* values selected were 30,40,45,50 and 61(feature matrix without principal component analysis). It was seen that for *n*=40, there was a higher coefficient of correlation.

Weka attribute selector

WEKA(Waikato Environment for knowledge analysis) is a prevalent suite of machine learning algorithms written in Java, used for solving real-world data mining problems.

Advantages of Weka

- Free availability under the GNU General Public License.
- It is portable, as it runs on any platform and is fully implemented in Java.
- A comprehensive collection of data pre-processing and modelling techniques.
- Ease of use due to its graphical user interfaces.
- WEKA supports several standard data tasks, specifically, data pre-processing, classification, clustering, regression and feature selection.
- Run individual experiments.
- Builds KDD phases.

Feature subset selection is the method of recognising and eliminating as much extraneous and redundant information as possible. This decreases the dimensionality of the data and may allow the algorithms to operate faster and more efficiently. Although in some cases, accuracy on predictive classification can be improved; in others, the result is a more compact and straightforward representation of the target concept. This reduces the dimensionality of the data and may allow learning algorithms to operate faster and more effectively.

Weka offers an attribute selection option. The process is divided into two parts:

Attribute Evaluator: A technique in which attribute subsets are evaluated.

Search Method: A technique in which the space of possible subsets is searched.

In an attribute evaluator for example, they may be assessed by building a model and estimating the accuracy of the model.

The following are some of the examples of attribute evaluation methods:

1. **CfsSubsetEval**: Prioritizes subsets that highly correlate with the class value and correlate minimally with each other.
2. **ClassifierSubsetEval**: Evaluates subsets using a predictive algorithm and another dataset that is specified.
3. **WrapperSubsetEval**: Evaluates subsets using a classifier that is specified and n-fold cross validation.

The Search Method is the organised way in which the search space of probable attribute subsets is directed based on the subset evaluation. Baseline approaches include Random Search and Exhaustive Search, other popular graph search algorithms such as Best First Search.

The following are some of the examples of attribute evaluation methods:

1. **Exhaustive**: Checks all combinations of attributes.
2. **BestFirst**: A best-first search strategy is used to navigate attribute subsets.
3. **GreedyStepWise**: A forward (additive) or backward (subtractive) step-wise strategy is used to navigate attribute subsets.
   In this project, the algorithm performs a greedy forward search in the space of attribute subsets. It begins with no attributes in the space and stops when the addition/deletion of any left-over attributes results in a decrease in evaluation.

Correlation based Feature Selection (CFS)

The problem of feature selection for machine learning was addressed through a correlation based approach. The dominant hypothesis is that good data sets comprises features that are highly correlated with the class, and have a low correlation with each other. A feature evaluation procedure, centred on ideas from test theory, offers a functioning definition of this hypothesis. CFS is an algorithm that couples this evaluation formula with an appropriate correlation measure and a heuristic search strategy. It searches subsets according to the degree of redundancy among the features. The subset evaluators use a numeric measure, such as conditional entropy, to guide the search iteratively and add features that have the highest correlation with the class. Evaluator aims to find the subsets of features that are exclusively correlated highly with the class but have low inter-correlation.

$$M_s = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}} \tag{10}$$

where M_S is the heuristic "merit" of a feature subset S containing k features, r_{cf} is the mean feature-class correlation (f ∈ S), and r_f f is the average feature-feature inter-correlation.

CFS assesses the value of a subset of attributes bearing in mind the individual predictive ability of each feature along with the unit of redundancy between them. Correlation coefficients are used to evaluate the inter-correlations between the features as well as the correlations between subset of attributes and class [18, 19].

Classification

The selected features were then used for classifying the actions studied. A series of classification algorithms were used to train the data from first nine subjects. The algorithms that had coefficient of correlation, between actual class and predicted class, above 0.9 were considered. The following classifiers were studied.

Decision Tree:

A decision tree acquired its name as it is shaped like a tree and used to make decisions. Theoretically, a tree is a set of branches and nodes and each branch descends from a node to another node. To derive a decision using the tree for a given case, the attribute values of the case are considered and the tree is traversed from the root node down to the leaf node that holds the decision [8].The nodes characterise the attributes considered in the decision process and the branches characterise the diverse attribute values. The basic apprehensions in a decision tree classifier are the division of clusters at each non-terminal node and the feature choices that are most effective in separating the group of classes. In a decision tree classifier design, it is desirable to build an optimum tree so as to achieve the maximum possible classification accuracy with the least number of calculations.

Random Forest and Bagging:

Random forest is a combination of tree predictors in which each tree relies on the values of a random vector sampled individually and with the same distribution for all trees in the forest. It is defined as follows:

“A random forest is a classifier consisting of a collection of tree structured classifiers

$$h(x, \Theta_{k)}, k=1,... \qquad (11)$$

where the $\{\Theta_k\}$ *are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x”*.[14]

Bagging (Bootstrap Aggregating) is a method of ensemble classification that creates distinct samples of the training dataset and a classifier for each sample. The results of these classifiers are then averaged or combined (such as majority voting). The procedure is that every sample of the training dataset is unrelated, providing each classifier that is trained, a subtly disparate focus and perception on the problem [14].Bagging is similar to random forests. The fundamental difference is that in Random Forests, only a subset of features are selected at random out of the total and the best split feature from the subset is used to split every node in a tree, contrasting from bagging where all features are utilized for splitting a node.

Neural Network:

A neural network comprises of three layers:

- Input Layer – This is the part of ANN where input is fed to the network.
- Output Layer – This is the place where we obtain the output that is calculated by the network.
- Hidden Layers – These are the transition layers between the input and output layers where various calculation occurs which leads to the generation of output [20].

Transfer functions are used to transform the given input to values alternating from zero to one. Each layer comprises a specific number of values and each value is called a node. Distant from these one of the chief components of a neural network are weights. They are numbers or coefficients which are multiplied with each node of an individual layer to form the next layer. The nodes and the layers both are characterised in a form matrix. The multiplication of a layer matrix to its assigned weight matrix leads to generation of next layer. Cost function is an equation relating the calculated output to the actual output and is directly proportional to the error. Hence in order to minimize the error, the cost function has to be minimized.

Results and Discussions

Once, the features were extracted, the entire feature set was fed to a k Nearest Neighbour (kNN) classifier. The correlation coefficient (0.63) was found to be unsatisfactory. To overcome this problem, the feature matrix was subject to Principal Component Analysis. The following were observed for kNN:-

- For dimension length of 30, correlation coefficient was 0.52
- For dimension length of 40, correlation coefficient was 0.71
- For dimension length of 45, correlation coefficient was 0. 61
- For dimension length of 50, correlation coefficient was 0.58

It is evident from above that for a dimension length of 40, the correlation coefficient was highest. In order to further improve, the efficiency of classification, feature selection was implemented. A correlation based feature selection algorithm was used. To find the set of features which gave the best results for the above mentioned evaluator, a greedy stepwise search was realized. This search resulted in ranking the first 10 valid features that would result in high classification efficiency. The selected features were:

- Sensor 3 – Polynomial Regression coefficients 1,2,3,4
- Sensor 1 – Exponential fit coefficients – 4
- Sensor 1 – 1st order LPC
- Sensor 3 – 1st order LPC
- Height of Subjects
- Circumference of forearm
- Circumference of bicep

This reduced feature matrix was classified using classification algorithms such as Random Forest, Bagging and Neural Network. k-Fold cross validation iteration with k =10 was employed for each of the classifier.

For a measure of classification accuracy, we used F-measure as the metric.

$$Precision = \frac{tp}{tp + fp} \tag{12}$$

$$Recall = \frac{tp}{tp + fn} \tag{13}$$

$$F - measure = 2 * \frac{Precision * \ Recall}{Precision + Recall} \tag{14}$$

The confusion matrices with the respective f-measures are listed below.

Random Forest - 10 features - cross validation											
	Predicted										
		Action1	Action2	Action3	Action4	Action5	Action6	Action7	Action8	Action9	Action10
Actual	Action1	74.47	25.53	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action2	8.33	69.44	22.22	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action3	0.00	8.33	88.89	2.78	0.00	0.00	0.00	0.00	0.00	0.00
	Action4	0.00	0.00	2.38	88.10	9.52	0.00	0.00	0.00	0.00	0.00
	Action5	0.00	0.00	0.00	0.00	92.59	7.41	0.00	0.00	0.00	0.00
	Action6	0.00	0.00	0.00	0.00	7.50	90.00	2.50	0.00	0.00	0.00
	Action7	0.00	0.00	0.00	0.00	0.00	6.12	87.76	6.12	0.00	0.00
	Action8	0.00	0.00	0.00	0.00	0.00	0.00	4.55	95.45	0.00	0.00
	Action9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	33.33	66.67	0.00
	Action10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	16.28	83.72
	F-Measures	**0.82**	**0.66**	**0.83**	**0.93**	**0.85**	**0.89**	**0.91**	**0.93**	**0.33**	**0.91**

Table 1: Confusion Matrix of Random Forest - 10 features - cross validation.

Bagging - 10 features - cross validation											
	Predicted										
		Action1	Action2	Action3	Action4	Action5	Action6	Action7	Action8	Action9	Action10
Actual	Action1	82.98	17.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action2	2.78	80.56	16.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action3	0.00	11.11	88.89	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action4	0.00	0.00	7.14	83.33	9.52	0.00	0.00	0.00	0.00	0.00
	Action5	0.00	0.00	0.00	14.81	74.07	11.11	0.00	0.00	0.00	0.00
	Action6	0.00	0.00	0.00	0.00	10.00	90.00	0.00	0.00	0.00	0.00
	Action7	0.00	0.00	0.00	0.00	0.00	12.24	77.55	10.20	0.00	0.00
	Action8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00
	Action9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	33.33	0.00	66.67
	Action10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
	F-Measures	**0.90**	**0.75**	**0.83**	**0.86**	**0.73**	**0.85**	**0.87**	**0.94**	**0.00**	**0.98**

Table 2: Confusion Matrix of Bagging - 10 features - cross validation.

NN - 40 features - cross validation											
	Predicted										
		Action1	Action2	Action3	Action4	Action5	Action6	Action7	Action8	Action9	Action10
Actual	Action1	69.44	30.55	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action2	8.51	61.70	14.89	10.64	0.00	4.26	0.00	0.00	0.00	0.00
	Action3	2.78	19.44	47.22	25.00	2.78	0.00	0.00	0.00	2.78	0.00
	Action4	0.00	0.00	14.29	65.71	14.29	2.86	2.86	0.00	0.00	0.00
	Action5	0.00	0.00	0.00	4.76	83.33	11.90	0.00	0.00	0.00	0.00
	Action6	0.00	0.00	0.00	0.00	18.52	66.67	14.81	0.00	0.00	0.00
	Action7	0.00	0.00	0.00	0.00	0.00	5.00	82.50	12.50	0.00	0.00
	Action8	0.00	0.00	0.00	0.00	2.04	0.00	16.33	65.31	16.33	0.00
	Action9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	16.67	71.43	11.90
	Action10	0.00	0.00	0.00	0.00	0.00	33.33	66.67	0.00	0.00	0.00
	F-Measures	**0.00**	**0.70**	**0.52**	**0.62**	**0.79**	**0.64**	**0.75**	**0.69**	**0.74**	**0.00**

Table 3: Confusion Matrix of NN - 40
features - cross validation.

From the above confusion matrices, it is observed that classification efficiency was highest for the bagging classification at 86.1%. Random forest method was a close second at 85.28%. Neural network, as expected, gave a mediocre classification efficiency of 67.6%. However, on close observation, we find that in the random forest – cross validation method, all actions are classified to a good extent, while in bagging – cross validation method, Action 9 was not classified. In order to make the classification more valid, a random forest – split method was employed. Here, data from the first 9 subjects were used to train the classifier and the data from the remaining 4 subjects were used to test the classifier. The confusion matrix of this classifier is shown below.

Random Forest 10 features 9-4 split											
	Predicted										
		Action1	Action2	Action3	Action4	Action5	Action6	Action7	Action8	Action9	Action10
Actual	Action1	14.29	85.71	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action2	0.00	60.00	40.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Action3	0.00	0.00	87.50	12.50	0.00	0.00	0.00	0.00	0.00	0.00
	Action4	0.00	0.00	0.00	58.33	41.67	0.00	0.00	0.00	0.00	0.00
	Action5	0.00	0.00	0.00	0.00	90.91	9.09	0.00	0.00	0.00	0.00
	Action6	0.00	0.00	0.00	0.00	0.00	92.31	7.69	0.00	0.00	0.00
	Action7	0.00	0.00	0.00	0.00	0.00	0.00	70.59	29.41	0.00	0.00
	Action8	0.00	0.00	0.00	0.00	9.09	90.91	0.00	0.00	0.00	0.00
	Action9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00
	Action10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
	F-Measures	**0.25**	**0.43**	**0.74**	**0.70**	**0.74**	**0.67**	**0.80**	**0.00**	**1.00**	**1.00**

Table 4: Confusion Matrix of Random
Forest 10 features 9-4 split.

This method, though does not have a high classification efficiency (63.63%), it is a proof that the features extracted from the signal can be used for effective classification of actions. It was also found that Sensor 2 (bicep) had no effect on the classification of the above studied actions.

Future Work and Scope for Improvement

There is a lot of scope for improvement in this project. A few notable ones have been listed below.

- As the actions selected for study here involve only open elbow, data from Sensor 2 was found to be inconsequential during classification. Using closed elbow in combination with the actions mentioned, more actions could be classified using Sensor 2.
- Most of the parameters used in the above classifiers such as maximum depth of trees, weight of vote from each tree in random forest, etc were not modified and experimented on Weka. Tweaking of these values may improve classification efficiency.
- Training data can be expanded to a larger size by involving more subjects. Algorithms like Neural Network and random forest show improvement on increasing the training set.
- The ADC used in this project was nearly 1 kHz, while the band of interest is between 60-150 Hz. A faster ADC could be used in order to sample the signal more effectively.
- As the sampling rate of ADC used was low, an anti-aliasing filter with a cut-off frequency of 400Hz should be used.
- Hardware issues can be handled better by building a casing for the set-up.

Conclusion

It is found that Random Forests are an effective tool in prediction. They do not over fit because of the Law of Large Numbers. Introducing the precise randomness makes them accurate in regression and classification. Moreover, the framework in terms of power of the specific predictors and its correlations gives a perception into the ability of the random forest to predict. The out-of-bag estimation makes concrete the otherwise theoretical values of strength and correlation. In this project, the aim was to create an action to speech convertor. Among the actions studied in this project, all actions are identified using the features mentioned with an acceptable average accuracy of 86.1%. The sensors used in this experiment cost no more than □3,000 and the ADC and Multiplexor of an Arduino UNO would cost about □1,500. The total cost of the marketable product would be at the most, □10,000. This minimalistic approach is simpler, lower cost and less bulky. It consumes less energy, and can be more intuitive for the user. The final outcome of the experiments can be seen as an illustrative step towards gaining useful knowledge that enables to decide which algorithm to use in certain situations.

References

1. TanzimKawnine, “A Radial-Ulnar Deviation and Wrist-Finger Flexion Analysis Based on Electromyography”, Mälardalen University, Sweden, 2008.

2. Jingpeng Wang, Liqiong Tang, John E Bronlund, “Surface EMG Signal Amplification and Filtering”, International Journal of Computer Applications (0975-8887) Volume 82- No1, November 2013.

3. Rami N Khushaba, SarathKodagoda, MaenTakruri, GaminiDissanayake, “Towards Improved Control of Prosthetic Fingers using Surface Electromyogram (EMG) Signals, Elsevier, Expert Systems with Applications 39 (2012) 10731-10738.

4. https://cdn.sparkfun.com/datasheets/Sensors/Biometric/MyowareUserManualAT-04-001.pdf

5. Dr. Scott Ray, “Important Factors in Surface EMG Measurement”, Bortec Biomedical Ltd.
6. Xin Dang, Takayoshi Nakai and Md. Iqbal Aziz Kha , “Noise Power Spectral Density Estimation based on Maximum a Posteriori and Generalized Gamma Distribution”, International Journal of Advanced Science and Technology Vol. 54, May, 2013.
7. AngkoonPhinyomark, ChusakLimsakul, and PornchaiPhukpattaranont, “A Novel Feature Extraction for Robust EMG Pattern Recognition, JOURNAL OF COMPUTING, VOLUME 1, ISSUE 1, DECEMBER 2009, ISSN: 2151-9617
8. Bhanu Prakash Battula, KVSS Rama Krishna and Tai-hoon Kim,” An Efficient Approach for Knowledge Discovery in Decision Trees using Inter Quartile Range Transform”, International Journal of Control and Automation Vol. 8, No. 7 (2015), pp. 325-334.
9. Wan Mohd Bukhari Wan Daud, Abu Bakar Yahya, Chong Shin Horng, Mohamad FaniSulaima, and RubitaSudirman ,“Features Extraction of Electromyography Signals in TimeDomain on Biceps Brachii Muscle”, International Journal of Modeling and Optimization, Vol. 3, No. 6, December 2013.
10. Amjed S. Al-Fahoum and Ausilah A. Al-Fraihat,” Methods of EEG Signal Features Extraction Using Linear Analysis in Frequency and Time-Frequency Domains”, ISRN NeuroscienceVolume 2014, Article ID 730218, 7 pages.
11. L.Ladha, T.Deepa,”FEATURE SELECTION METHODS AND ALGORITHMS”, International Journal on Computer Science and Engineering (IJCSE), ISSN : 0975-3397 Vol. 3 No. 5 May 2011.
12. Mohini D Patil, Dr.Shirish S. Sane,” Effective Classification afterDimension Reduction: A Comparative Study”, International Journal of Scientific and Research Publications, Volume 4, Issue 7, July 2014 1 ISSN 2250-3153.
13. AngkoonPhinyomark, ChusakLimsakul, PornchaiPhukpattaranont,” A Novel Feature Extraction for Robust EMG Pattern Recognition”,Journal of Computing, Volume 1, Issue 1, pp 71-80, December 2009.
14. Leo Breiman, “RANDOM FORESTS”, pp. 2-3, 6-11.

15. Shruti Vaidya, Dr. Kamal Shah," Audio Denoising, Recognitionand Retrieval", IOSR IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. II (Mar-Apr. 2014), pp 107-112.
16. Sachin Sharma, Gaurav Kumar, Sandeep Kumar, DebasisMohapatra, " Techniques for Feature Extraction from EMG Signal", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 1, January 2012 ISSN: 2277 128X.
17. Mahmoud Tavakoli, Carlo Benussi, Joao Luis Lourenco, "Single channel surface EMG control of advanced prosthetic hands: A simple, low cost and efficient approach", Expert Systems With Applications 79 (2017) 322–332.
18. M. A. Hall (1998). Correlation-based Feature Subset Selection for Machine Learning. Hamilton, New Zealand.
19. D. Aha, D. Kibler (1991). Instance-based learning algorithms. Machine Learning. 6:37-66.
20. Deepak P, KshitijShrivastava, Prathik K, Gautham Ganesh, Puneet S, Vijay Mishra, "Artificial Neural Network For Automated Gas Sensor Calibration", Proceedings of 29th IRF International Conference, 24th July, 2016, Bengaluru, India, ISBN: 978-93-86083-69-2