University of Benghazi

Faculty of Engineering

Electrical and Electronic Engineering Department

EMG-Based Prosthetic hand

A project Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science (B.Sc.) in Electrical and Electronic Engineering.

EE500

By

Abdullrhman Mohammed Ezwie (20551)

Ahmed Gumma Elmagbri (19811)

Supervisor

Dr. Lamia Abusedra

Spring 2017

جامعـــة بنغازي / كليـــة الهندســـة

قسم الهندسـة الكهربائيــة والإلكترونية

ذراع صناعية تعمل بواسطة الإشارات الكهروعضلية

قدم المشروع لإستكمال جزء من متطلبات الحصول على درجة البكالوريوس في الهندسة الكهربائية والإلكترونية

قـــدم من قبــل :

1.عبدالرحمن محمد الزوي (20551)

2. أحمد جمعة المجبري (19811)

تحت إشــراف

د. لمياء أبوسدرة

ربيع 2017

CERTIFICATE

The project entitled:

EMG-Based Prosthetic hand

The project entitled:

Abdullrhman Mohammed Ezwie (20551)

Ahmed Gumma Elmagbri (19811)

In the partial fulfillment of requirement for the award of the B.Sc. degree in Electrical and Electronic Engineering. This has been carried out under my supervision and accepted for presentation & examination.

Signature:

Dr. Lamia abusedra

Date: / / 2017

The Supervisor

CERTIFICATE

The project entitled:

EMG-Based Prosthetic hand

The project entitled:

Abdullrhman Mohammed Ezwie (20551)

Ahmed Gumma Elmagbri (19811)

In the partial fulfillment of requirement for the award of the B.Sc. degree in Electrical and Electronic Engineering. This has been carried out under my supervision and accepted for presentation & examination.

1st Examiner (The supervisor)	**2nd Examiner**
Signature:	**Signature:**
Name :	**Name :**
Date: / /2017	**Date: / /2017**

3rd Examiner

Signature:

Name :

Date: / /2017

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

i. Abstract:

Aimed at improving the quality of life for amputees, this project aims to design and build of an EMG-based prosthetic hand, to participate in healing the current tragedy in our society. This project implements the concepts for the development of a prosthetic limb that operates as a part of the human body's nervous system. Our design is implemented by adding electrodes into the nervous system detected its electrical activities. The acquired EMG signals are then required to be converted into mechanical motion, so as to provide the nature sensations experienced by a human arm. Since that, some processing techniques have to be attained to proper the desired system. This model manipulates the electrical pulses by extracting some features -*using Statistical features and Wavelet techniques*- classified *by many types of machine learning* into multi-classes depending on their functionalities. The model is operated by servo motors which are controlled by Arduino Uno.

الملخص :

سعياً إلى تحسين جودة حياة مبتوري الأطراف ، هذا المشروع يهدف إلى تصميم وبناء يد صناعية يتم التحكم بها بواسطة الإشارات الكهروعضلية ، وذلك للمساهمة في تطبيب ومعالجة المأساه الحالية في مجتمعنا.
يُطبِق هذا المشروع المفاهيم اللازمة لتطوير يد صناعية تعمل كجزء من الشبكة العصبية لجسم الإنسان. تصميمنا يقوم بإضافة مجسات توصل بالشبكة العصبية تعمل على تعقب الإشارات الكهرو عضلية حتى يتم استخدامها وتحويلها إلى حركة ميكانيكة مما تمنح الإحساس الطبيعي الذي كانت تختبره اليد الإنسانية ، ولفعل ذلك ، عدد من تقنيات معالجة الإشارة يتم تطبيقها لتجهيز وملائمة النظام المطلوب. هذا النموذج يقوم بمعالجة الإشارات الكهروعضلية باستخلاص بعض الخصائص *– بواسطة تقنية Wavelet-* التي تُصنف إلى عدد من التصنيفات *– بواسطة أنواع مختلفة من التعليم الألي -* اعتماداً على الوظيفة المطلوبة. يتم تشغيل هذا النظام بواسطة مواتير يُتحكم بها من قبل المتحكم أردوينو أونو.

ii. Objective:

The main objective of this project is to design and build an EMG-based prosthetic hand using advanced signal processing techniques.

Sub-objectives:

- To understand, implement and test the ability of advanced signal processing techniques .i.e. Wavelet Analysis and statistical features.
- To learn and implement machine learning concepts and apply pattern recognition techniques.
- To create and implement algorithms on a microcontroller hardware.

iii. Motivation:

Due to the civil war and current tragedy at our society that has caused thousands of people to loss their limbs and missing their ability to participate in daily life and due to the strong duty that the scientific research at academic organizations should hold and bring to their society, we have believed that there exists a massive need for the participation of growth the developing and building of prosthetic limbs design; and a massive need for accelerating the current local researches since this technology aims to give the amputees the ability to compensate the missing limb with the natural sense; therefore affording them an impact improves their life and their participating in society, that also has a great impact on our society since it cannot be imaginable of thinking of society development and investment in youth with this huge number of disable people were prevented to join and participate in the society. Globally, there is a global drastic increment of the demand for prosthetic devices and biomedical technologies in general, being as the highly interested research areas recent years. Since there is over a billion people –about 15% of the world's population- have some form of disability of who nearly 200 million experience considerable difficulties in functioning. In the years ahead, disability will be an even greater concern because its prevalence is on the rise. [1] This is due to ageing populations and the higher risk of disability in older people as well as the global increase in chronic health conditions such as diabetes, cardiovascular disease, cancer and mental health disorders and also the increased amputees due to civil wars and accidents injuries, etc.

Part I

Introduction and Background

CHAPTER (1): OVERVIEW

Human bioelectrical signals are extensively studied and applied in various clinical and psychophysiological researches. However, the intention of using these signals in the field of information technology is newer.

The limbs can move by generating electrical pulses from the brain to the muscles through the nervous system that connects every part of tissue in the human body as shown in figure (1), which is called action potential, every motion has a specific signal which can be recognized by the brain, in some cases people might lose their limbs but the nervous system still carry the produced electrical pulses. These pulses can be acquired and recorded by electrodiagontic medicine technique called Electromyography, these detected signals opened a powerful advantage field to be used through diverse applications in biomedicine and signal processing engineering, one of its usages is as an electrical signal input to operate and control a robotic hand motion. [2, 3]

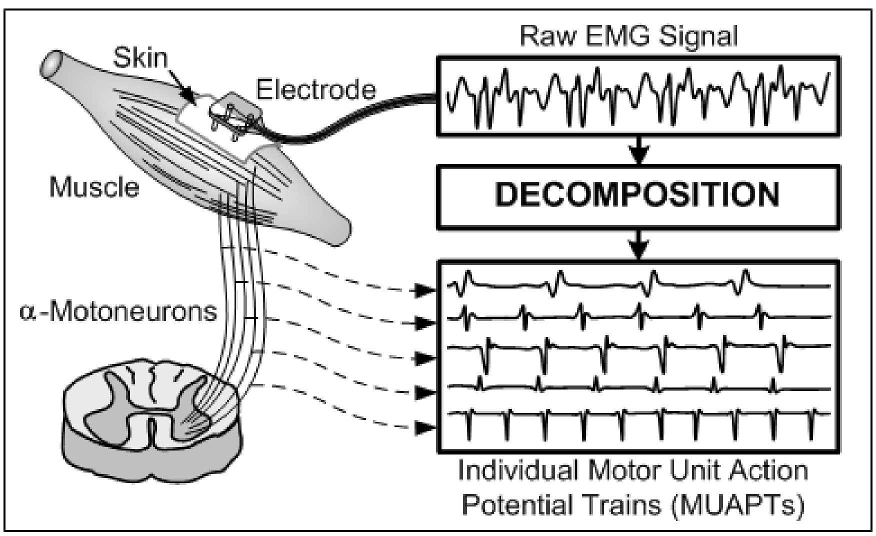


Figure (1): EMG Signal

The Engineering task approach is making these EMG signals proper to be converting into a mechanical motion, similar to the motion that the missing hand was doing.

To make a computer-based system distinguish these commands, some *features* are extracted from the EMG signals after passing through some filtering pre-processing, these extracted features are then *classified* to number of classes, and each class is responsible on one form of motion. This command is sent to the *motor* to apply it using a *microcontroller*. [4-6]

The 5 stages of procedures that EMG signals are applied to, can be shown in figure (2):

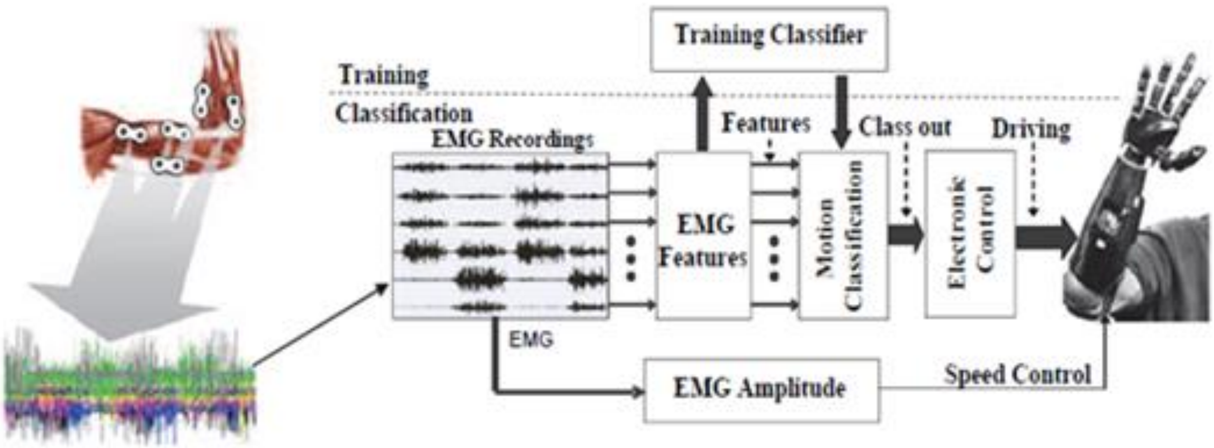


Figure (2): The stages of building prosthetic hand

1. Signal Acquisition:

The EMG signals are acquired using chemical electrodes which fixed around the skin of the amputee. Optimum position, the sampling rate has selected in different rates depending on the task.

2. Pre-Processing:

To proper the EMG signals for the processing, some filtering procedure have to be implemented using specific types of digital filters to clear the signals from the noise that is caused by motion artifacts, power line noise, electrode noise, ambient noise, and inherent noise.

3. Feature Extraction:

The goal is to make computers distinguishing the signals depending on the function that was supposed to proceed to the limb; therefore, some features are extracted from the raw signals to transformation them into relevant data vector. Many techniques of extracted features are implemented either based on time domain, frequency domain or time-frequency domain.

4. Classification:

The extracted features from the different hand functionalities; opening, closing or rotating, etc, are needed to be classified based on that functionality to give a sense of nature attitude. The classification process is a machine learning technique uses a mathematical models and algorithms to train the system to classify the extracted data into different set of classes.

5. The Microcontroller:

To implement the designed system, a set of hardware components are required: Microcontroller to implement the processing; Servo motors to control the prosthetic hand; and the cosmetic hand. [7-10]

CHAPTER (2): DESIGN STAGES

Based on the previous overview and the desired procedures of the engineering task of making the electrical pulses (EMG signal) to be proper for the model of the prosthetic hand, this project has built the design stages that has been planned to be followed and implemented to the desired goal of the project clarifying each stage with the tools and techniques aimed to be implemented.

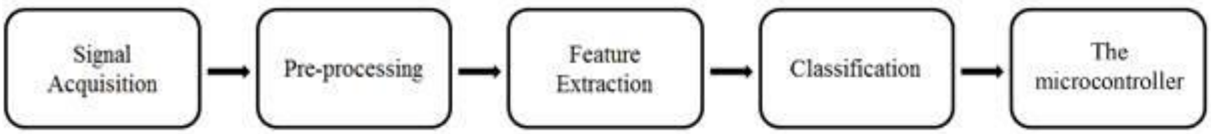


Figure (3) Design stages

1. Signal Acquisition:

A *commercial electrode* (sEMG sensor circuit designed by EPET) is used to collect the EMG data, which is used to train and test our design, beside some *data-bank* that available to researchers and developers (KIC Laboratory research group); to get more accurate and reliable data improves the results.

2. Pre-processing:

To pick up the EMG signals without noises; a *Butterworth band-pass filter* is used (15 Hz to 500 Hz), with a *notch 60-Hz filter*. This filtering design prepares clear proper EMG signals for further processing.

3. Feature extraction:

Wavelet transform and analysis is the mathematical tool used for feature extraction by using discrete wavelet transform which allow a time-scale decomposition of a signal, therefore, the project aim to study the wavelet transform effectiveness. This advanced technique is used due to the clear distinctive representation it provides to the signal, which can be utilized to simplify the categorization process. Beside a plenty of mathematical statistical features are used to get best features of the raw EMG signals.

4. Classification:

To guarantee that the best classifier is used to get the best possible results in the model, a plenty of classifier techniques are used and experimented, *(Quadratic Support Vector Machine, Cubic Support Vector Machine, Cubic K-Nearest Neighbor, Fine K-Nearest Neighbor),* to end up with the accurate classifier can be implemented to our model.

5. The Software Tools:

- The *MATLAB computing program* is used for the analysis, processing and implementing and build the model.
- The *LabVIEW* is used for practical simulation, to get the finalized built model into the real-time application, due to the strong abilities and properties of the LabVIEW.
- The *Arduino C* program is used to operate and send the commands that resulted from the processing of the model to be uploading into the hardware part (LEDs or servomotor).
- The *Audacity* open source program is the software used to acquire and save the EMG signals from the electrodes to the computer.

Chapter (3): FEATURE EXTRACTION

Once the EMG signal that acquired in a form of digitized data is preprocessed (filtered using band-pass 15-500Hz filter and 50Hz notch filter), we need to determine features from the raw signal by the use of digital processing techniques to make the computer system able to distingue the different movements. This process is named ‘feature extraction’.

Feature extraction is a special form of dimensionality reduction. Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately.

The features had selected after testing their effectiveness, since choosing suitable number of features is required to avoid redundancy and make the algorithm more practical.

3.1 The Need for feature extraction:

When the input data to an algorithm is too large to be processed and it is suspected to be redundant which is much data, but not much information then the input data will be transformed into a reduced representation set of features.

Transforming the input data into the set of features is called features extraction. Thus the extraction of discriminatory features in the signal enhances the reduction of the length of the data vector by eliminating redundancy in the signal and compressing the relevant information into a feature vector of significantly lower dimension.

3.2 The Extracted Statistical Features:

Some mathematical equations can be used to extract some statistical features from the raw data of the signal, these statistics are introduces in the following part:

- ***Root Mean Square:***

RMS is square root of the arithmetic mean of the squares of the values, or the square of the function that defines the continuous waveform. In Physics, the RMS current is the "value of the direct current that dissipates power in a resistor.

$$RMS = \sqrt{\left(\frac{1}{N}\right)\sum_{n=1}^{N} D_i^2\ (n)} \tag{1}$$

- ***Willison Amplitude:***

W-Amplitude is the number of times that the difference between sEMG signal amplitude among two adjacent segments that exceeds a predefined threshold to reduce noise effects.

$$WAMP = \sum_{i-1}^{N} f(|x_{i-}\ x_{i+1}|) \tag{2}$$

- ***Waveform length (WL):***

  WL is the cumulative length of the waveform over the time segment. WL is related to the waveform amplitude, frequency and time.

$$WL = \sum_{n=1}^{N-1} |x_{n+1} - x_n| \tag{3}$$

- ***Simple Square Integral (SSI):***

  uses the energy of the EMG signal as a feature. It can be expressed as

$$SSI = \sum_{n=1}^{N} |x_n^2| \tag{4}$$

- ***Integrated EMG (IEMG):***

  IEMG is calculated as the summation of the absolute values of the EMG signal amplitude. Generally, IEMG is used as an onset index to detect the muscle activity that used to oncoming the control command of assistive control device. It is related to the sEMG signal sequence firing point, which can be expressed as

$$IEMG = \sum_{n=1}^{N} x_n \tag{5}$$

Where N denotes the length of the signal and x_n represents the EMG signal in a segment.

- ***Variance (VAR):***
  VAR uses the power of the EMG signal as a feature. Generally, the variance is the mean value of the square of the deviation of that variable. However, mean of EMG signal is close to zero. In consequence, variance of EMG can be calculated by

$$VAR = 1/(N-1) \sum_{n=1}^{N} x_n^2 \tag{6}$$

- ***Mean Absolute Value (MAV):***
  MAV is similar to average rectified value (ARV). It can be calculated using the moving average of full-wave rectified EMG. In other words, it is calculated by taking the average of the absolute value of EMG signal. It is an easy way for detection of muscle contraction levels and it is a popular feature used in myoelectric control application. It is defined as

$$MAV = 1/N \sum_{n=1}^{N} |x_n^2| \tag{7}$$

- ***Zero crossing (ZC):***
  ZC is the number of times that the amplitude value of EMG signal crosses the zero y axes. In EMG feature, the threshold condition is used to abstain from the background noise. This feature provides an approximate estimation of frequency domain properties. It can be formulated as

$$ZC = \sum_{n=1}^{N} [\, sgn\,(x_n \times x_{n+1}) \cap |x_n - x_{n+1}| \geq threshold] \tag{8}$$

- ***Slope Sign Change (SSC):***
 is similar to ZC. It is another method to represent the frequency information of EMG signal. The number of changes between positive and negative slope among three consecutive segments are performed with the threshold function for avoiding the interference in EMG signal. The calculation is defined as

$$SSC = \sum_{n=1}^{N} [\, f\, [(x_n - x_{n+1}) \times (x_n - x_{n+1})]] \qquad (9)$$

CHAPTER (4): WAVELET ANAYLSIS

One frequent way of collecting experimental data by scientists and engineers is as sequences of values at regularly spaced intervals in time. These sequences are called time-series. The fundamental problem with the data in the form of time-series is how to process them in order to extract meaningful and correct information, i.e., the possible signals embedded in them. If a time-series is stationary one can think that it can have harmonic components that can be detected by means of Fourier analysis, i.e., Fourier transforms (FT). However, in recent times, it became evident that many time-series are not stationary in the sense that their mean properties change in time. The waves of infinite support that form the harmonic components are not adequate in the latter case in which one needs waves localized not only in frequency but in time as well. They have been called wavelets and allow a time-scale decomposition of a signal. Significant progress in understanding the wavelet processing of non-stationary signals has been achieved over the last two decades. However, to get the dynamics that produces a non-stationary signal it is crucial that in the corresponding time-series a correct separation of the fluctuations from the average behavior, or trend, is performed. Therefore, people had to invent novel statistical methods of selecting the data that should be combined with the wavelet analysis. A bunch of such techniques have been developed lately for the important class of non-stationary time series that display multi-scaling behavior of the multi-fractal type.

To give a better idea of why wavelet transform is importance tool, let us discuss and introduce the development of the analysis techniques:

4.1 Fourier analysis

Signal analysts already have at their disposal an impressive arsenal of tools. Perhaps the most well-known of these is *Fourier analysis*, which breaks down a signal into constituent sinusoids of different frequencies as shown in figure (4). Another way to think of Fourier analysis is as a mathematical technique for *transforming* our view of the signal from a time-based one to a frequency-based one.

Figure (4): Fourier analysis

For many signals, Fourier analysis is extremely useful because the signal's frequency content is of great importance. So why do we need other techniques, like wavelet analysis? Fourier analysis has a serious drawback. In transforming to the frequency domain, time information is lost. When looking at a Fourier transform of a signal, it is impossible to tell *when* a particular event took place.

If a signal doesn't change much over time — that is, if it is what is called a *stationary* signal — this drawback isn't very important. However, most interesting signals contain numerous non-stationary or transitory characteristics: drift, trends, abrupt changes, and beginnings and ends of events. These characteristics are often the most important part of the signal, and Fourier analysis is not suited to detecting them.

4.2 Short-Time Fourier analysis

In an effort to correct this deficiency, Dennis Gabor (1946) adapted the Fourier transform to analyze only a small section of the signal at a time — a technique called *windowing* the signal as shown in figure (4). Gabor's adaptation, called the *Short-Time Fourier Transform* (STFT), maps a signal into a two-dimensional function of time and frequency.

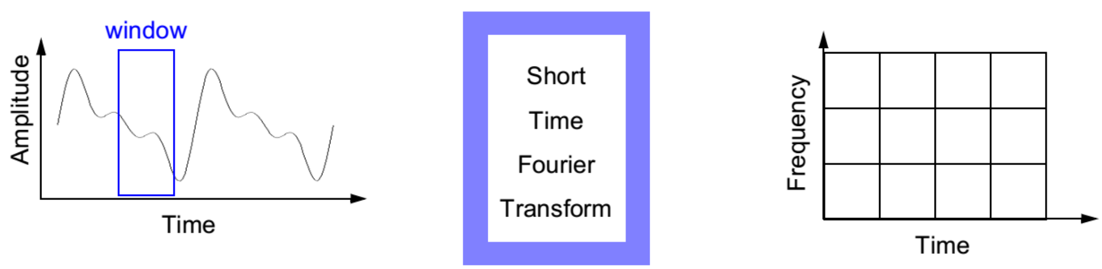


Figure (5) Short-Time Fourier analysis

The STFT represents a sort of compromise between the time- and frequency-based views of a signal. It provides some information about both when and at what frequencies a signal event occurs. However, you can only obtain this information with limited precision, and that precision is determined by the size of the window. While the STFT's compromise between time and frequency information can be useful, the drawback is that once you choose a particular size for the time window, that window is the same for all frequencies. Many signals require a more flexible approach — one where we can vary the window size to determine more accurately either time or frequency.

4.3 Wavelet Analysis

Wavelet analysis represents the next logical step as shown in figure (5), a windowing technique with variable-sized regions. Wavelet analysis allows the use of long time intervals where we want more precise low frequency information, and shorter regions where we want high frequency information. Here's what this looks like in contrast with the time-based, frequency-based, and STFT views of a signal:

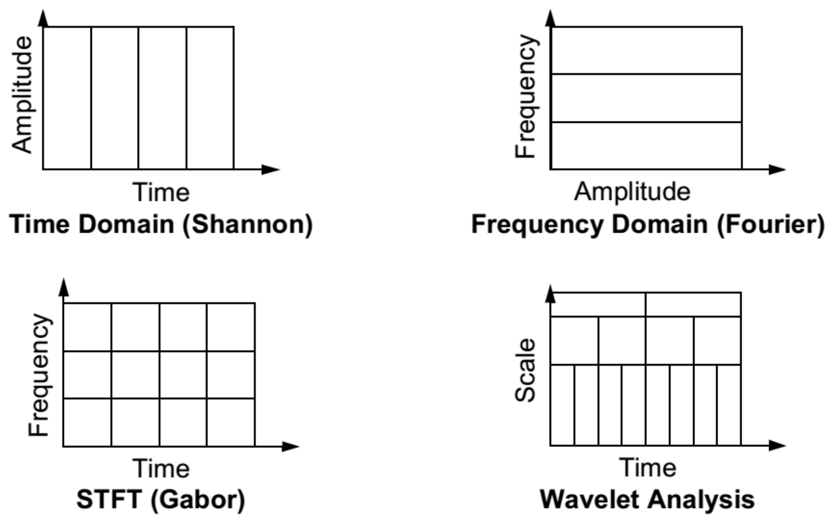


Figure (6): comparison between the discussed methods

4.3.1 What Can Wavelet Analysis Do?

One major advantage afforded by wavelets is the ability to perform *local analysis* — that is, to analyze a localized area of a larger signal. Consider a sinusoidal signal with a small discontinuity as shown in figure (7) — one as tiny as to be barely visible. Such a signal easily could be generated in the real world, perhaps by a power fluctuation or a noisy switch.

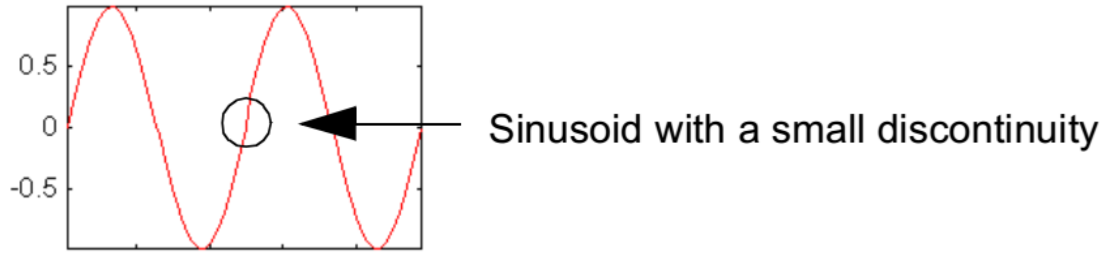


Figure (7): sinusoid with a small discontinuity.

A plot of the Fourier of this signal shows nothing particularly interesting: a flat spectrum with two peaks representing a single frequency. However, a plot of wavelet coefficients clearly shows the exact location in time of the discontinuity as shown in figure (8).

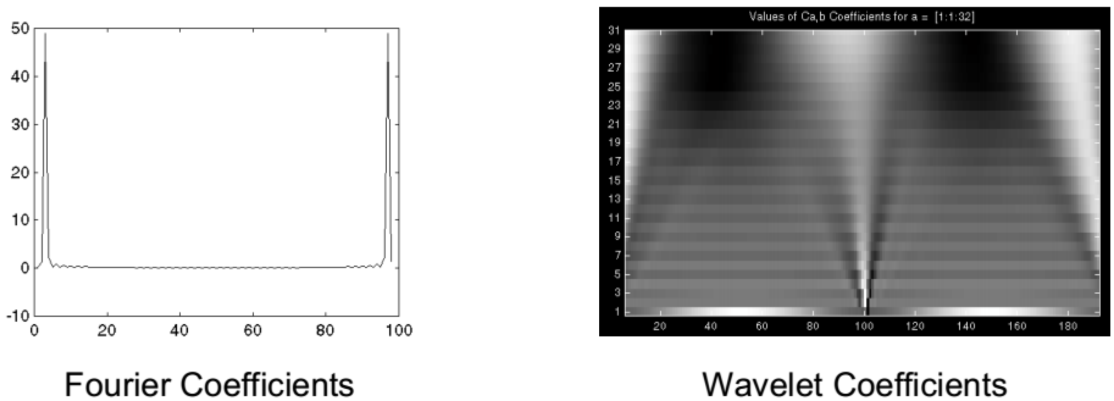


Figure (8): The coefficients of Fourier and Wavelet analysis

Wavelet analysis is capable of revealing aspects of data that other signal analysis techniques miss aspects like trends, breakdown points, discontinuities in higher derivatives, and self-similarity. Further, because it affords a different view of data than those presented by traditional techniques, wavelet analysis can often compress or de-noise a signal without appreciable degradation. Indeed, in their brief history within the signal processing field, wavelets have already proven themselves to be an indispensable addition to the analyst's collection of tools and continue to enjoy a burgeoning popularity today.

4.3.2 The Wavelet transforms:

There is a relation call admissibility condition which implies that the wavelet must have a zero average:

$$\int_{-\infty}^{\infty} \varphi(t)dt = \hat{\varphi}\,(0) = 0 \tag{10}$$

The wavelet function is called dilated–translated wavelets function

$$\varphi_{a,b}(t) = (1/\sqrt{a}\,)\ \varphi((t-b)/a) \tag{11}$$

There are two types of wavelet transform:

1. **Continuous Wavelet Transform:**

Any signal processing performed on a computer using real-world data must be performed on a discrete signal — that is, on a signal that has been measured at discrete time intervals. It is important to remember that the continuous wavelet transform is also operating in discrete time. So what exactly is "continuous" about it?
What's "continuous" about the CWT, and what distinguishes it from the discrete wavelet transform (to be discussed in the following section), are the scales at which it operates. Unlike the discrete wavelet transform, the CWT can operate at every scale, from that of the original signal up to some maximum scale which you determine by trading off your need for detailed analysis with available computational horsepower.
The CWT is also continuous in terms of shifting: during computation, the analyzing wavelet is shifted smoothly over the full domain of the analyzed function.

2. **Discrete Wavelet Transform:**

It turns out, rather remarkably, that if we choose scales and positions based on powers of two — so-called *dyadic* scales and positions — then our analysis will be much more efficient and just as accurate. We obtain just such an analysis from the *discrete wavelet transform* (DWT).

The discrete wavelet transform function is:

$$\varphi_{m,n}(t) = a_0^{-m/2}\ \varphi(\frac{t-nb_0}{a_0^m}) \tag{12}$$

Symbols m and n are integer values.

4.3.3 Mother wavelet:

There are many types of wavelet they can be used in wavelet transform and we refer to them as the mother wavelet, in the project the daubechies2 as shown in figure (9) is used and represented as:

$$\alpha_{k=0}\ if\ k < 0\ or\ k > 2n \tag{13}$$

$$\sum_{k=-\infty}^{\infty} \alpha_k \alpha_{k+2m} = \delta_{0m} \qquad for\ all\ integer\ m \tag{14}$$

$$\sum_{k=-\infty}^{\infty} \alpha_k = \sqrt{2} \tag{15}$$

$$\sum_{k=-\infty}^{\infty} \beta^k k^m = 0 \quad 0 \le m \le N-1 \tag{16}$$

$$where\ \beta_k = (-1)^k \alpha_{k+1}$$

We can find compactly supporting scaling function $\Phi(t)$ from the above progression the finction is one sulotion of the functional equation:

$$\Phi(t) = \sum_{k=-\infty}^{\infty} \alpha_k \sqrt{2}\, \Phi(2t-k) \tag{17}$$

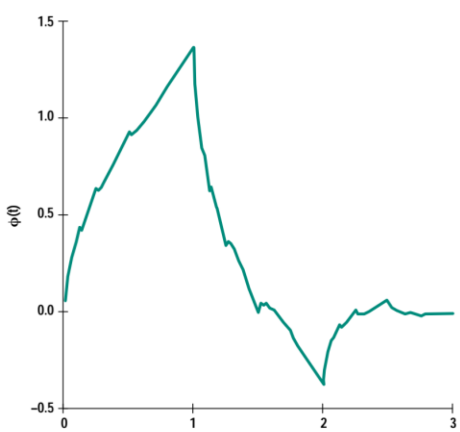


Figure (9): the daubechies2

CHAPTER (5): MACHINE LEARNING

Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed. It is algorithms can figure out how to perform important tasks by generalizing from examples. In other words machine learning is the scientific discipline whose goal is the classification of ***objects*** into a number of categories or ***classes.*** Depending on the application, these objects can be images or signal waveforms **or** any **type** of measurements that need to be classified Machine Learning:

- Grew out of work in AI
- New capability for computers

5.1 Examples of machine learning applications:

- **Database mining**
  large datasets from growth of automation/web.
  E.g., Web clicks data, medical records, biology, engineering
- **Applications can't program by hand.**
  E.g.: Autonomous helicopter, handwriting recognition, most of Natural Language Processing (NLP), Computer Vision.
- **Self-customizing programs**
  E.g., Amazon, Netflix product recommendations
- **Understanding human learning (brain, real AI).**

5.2 Types of machine learning technique

Machine learning technique has two types:

Supervised learning: where the algorithm generates a function that maps inputs to desired outputs. One standard formulation of the supervised learning task is the classification problem: the learner is required to learn (to approximate the behavior of) a function which maps a vector into one of several classes by looking at several input-output examples of the function.

Unsupervised learning: which models a set of inputs, labeled examples are not available. So the model is making and creating structure itself and categorizes them automatically.

Supervised learning has two types

1- **Regression:** which the method of getting information according to given data, predict continuous valued output, E.g.: predicting the price of a house according to its area, by knowing the data of given prices and areas.
2- **Classification**: used to make computers decide to classify a testing data into one from given classes (by trained data) according to some features extracted.

5.3 Classification:

The concept of classification is to insert an input into some algorithm and mathematical equation to map it to desired outputs
To make the concept clear, let us go through the procedure of bi-class, to map inputs into one of two classes.

Two classes –binary classes- then the output will be:

$$Y \in \{0, 1\}$$

The idea is to make a line separate the two classes, but a linear regression is not a good idea to classification

$$h_\theta(x) = \theta^T x \qquad (18)$$

Since we want $0 < h_\theta(x) < 1$

$$h_\theta(x) = g(\theta^T x)$$

So

$$g(z) = \frac{1}{1 + e^{-z}}$$

$$g(z) = \frac{1}{1+e^{-\theta^T x}}$$ “Sigmoid function” (19)

This is logistic regression:

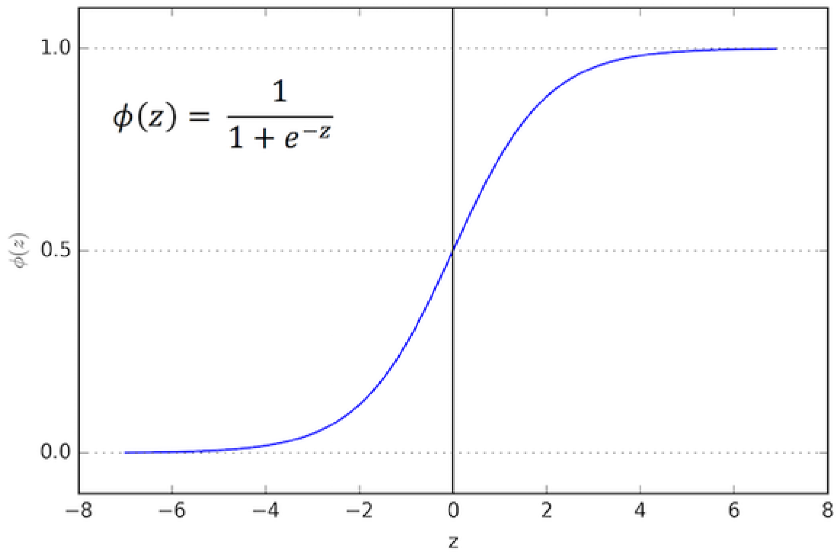


Figure (10): logistic regression equation.

5.3.1 Linear decision boundaries:

Every feature has a set of X's {X1, X2, X3, X4 … Xn}
Suppose we have a feature X1= [feature 1], the expected output would be rather Y1=0 "negative class" or Y2=1 "positive class". Using the conditional probability to check the class that X1 belongs to:

Example: the probability that given a feature X1, the output will be Y1 is P (y1|x1)

$$P(y1|x1) = \frac{1}{1+e^{-\theta^T x}} = 0.7$$

$$P(y1|x1) = \frac{1}{1+e^{-\theta^T x}} = 0.3$$

Then X1 maps to Y1

Now the question is how to make this boundary decision to separate the trained positive data from the trained negative data, The boundary equation is when $h_\theta(x) = g(\theta^T x) = 0.5$, $\theta^T x=0$

Where:

Θ is the fit values, $\theta^T = [a_1, a_2, a_3 \dots a_n]$.

X is the feature set $X = \{X1, X2, X3, X4 \dots Xn\}$

For Example if $\Theta = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ so the $h_\theta(x) = x_1 - x_2$

To find the linear decision boundary line, let $H_\theta(x)=0$, then the equation for decision boundary is x1=x2

Note: the boundary decision does not need to be linear, and could be a function that describes a circle (e.g. $z=\theta_0+\theta_1 x^2_1+\theta_2 x^2_2$) or any shape to fit our data.

Now how to fit the parameter theta?

Let us introduce the cost function, which is the function of having wrong decision:

$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y=1 \\ -\log(1-h_\theta(x)) & \text{if } y=0 \end{cases} \tag{20}$$

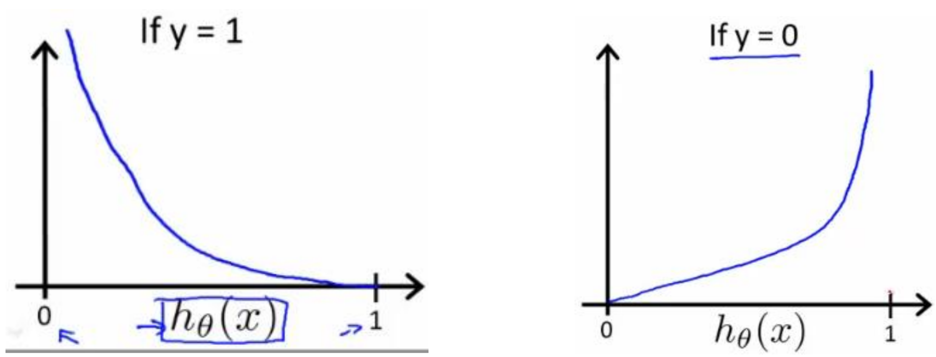


Figure (11): The cost function.

The more our hypothesis is off from y, the larger the cost functions output. If our hypothesis is equal to y, then our cost is 0, then the best theta values those would give minimum cost function.

So the cost function in one equation is:

$$J(\theta) = Cost(h\theta(x),y) = -y\log(h\theta(x)) - (1-y)\log(1-h\theta(x)) \tag{21}$$

To fit parameters θ, find min(J(θ)), by letting $\frac{dJ(\theta)}{d\theta} = 0$ and get θ.

There are different techniques develop out of this concept; let us cover the concepts behind the techniques used in this project:

5.3.2 Nearest neighbor (NN)

NN is one of the most popular classification rules, although it is an old technique. We are given c classes, ωi, i = 1,2,...,c, and N *training* points, xi, i = 1,2,...,N, in the l-dimensional space, with the corresponding class labels. Given a point, x, whose class label is unknown, the task is to classify x in one of the c classes. The rule consists of the following steps . According to the k-nearest neighbor estimation technique is:

1. Choose a value for k.
2. Find the distance between x and all training points xi, i = 1,2,..., N. Any distance measure can be used (e.g., Euclidean, Mahalanobis).

Euclidean distance: $||x - mi|| \equiv \sqrt{(x - mi)^T(x - mi)}$

3. Find the k-nearest points to x.
4. Out of the k-closest neighbors, identify the number ki of the points that belong to class ωi.
5. Assign x to class ωi, for which $ki > kj$, $j \neq i$. In other words, x is assigned to the class in which the majority of the k-closest neighbors belong.

5.3.3 The Bayesian classifier

We are given a pattern whose class label is unknown and we let $x \equiv [x(1), x(2), \ldots, x(l)]$ its corresponding feature vector, which results from some measurements. Also, we let the number of possible classes be equal to c, that is, $\omega_1 \ldots \omega_c$.

According to the Bayes decision theory, x is assigned to the class ω_i if

$$P(\omega_i|x) > P(\omega_j|x)$$

The *Gaussian* pdf is extensively used in pattern recognition because of its mathematical tractability as well as because of the central limit theorem. The latter states that the pdf of the sum of a number of statistically independent random variables tends to the Gaussian one as the number of summands tends to infinity. In practice, this is approximately true for a large enough number of summands, therefore each feature x with its training set are represented by a Gaussian distribution, the probability of any testing data to be that training set determines the class belongs to.

Part II

Project Management

CHAPTER 6: THE FIRST STAGE OF THE PROJECT

The project is managed into a set of tasks covering the procedures of design stages which divided during the two semester period, and it had done in two stages.

Table (1) Gantt chart Of the First Stage Design Work.

Task	Aug 1-7	Aug 8-14	Aug 15-21	Aug 22-28	Aug 29-Sep 6	Sep 7-13	Sep 14-20	Sep 21-27	Sep 28-Oct 3	Oct 4-10	Oct 11-17	Oct 18-24	Oct 25-Nov 1	Nov 2 -8	Nov 8 -14
Cover Medical Background	1 Week														
Study some Classifiers		2 Weeks													
Implemented Bayesian Classifier				1 week											
Study some Statistical Features					1 week										
Study Wavelet Analysis						4 Weeks									
Classified and extracted data										2 weeks					
Study types of digital filters												2 weeks			
Learning how to use Arduino Uno													3 Weeks		

The First Stage Tasks:

a- Covering the medical background:

The basic fundamental medical concepts are covered to well-understood the core of the problem.

b- Study and implement different types of classification:

The general concepts of machine learning and pattern recognition classifiers were covered and deeply focused on Bayesian, and Nearest-Neighbors classifiers.

c- Implemented Bayesian classifier:

Some pre-feature extracted dataset are used to testing our Bayesian classifier algorithm, the task was to distinguish between three classes of EMG signals represent different body activities "walking, running and jumping".

d- Study some statistical feature extraction:

Mean absolute value (MAV), variance (VAR), root mean square (RMS) are learned.

e- Study wavelet transform and analysis:

The fundamental concept of wavelet is learned, beside the understanding and using of wavelet toolbox in MATLAB.

f- Classifying and extracting statistical features of 50 data collection:

An obtained pre-collected data of 50 open and 50 closed hand EMG signals are classified successfully into two classes using three different statistical features; Mean absolute value (MAV), variance (VAR), root mean square (RMS), program is shown in the procedure.

g- Study different types of digital filters:

This part is well-covered in the digital signal processing course.

h- Study and implement Arduino Uno:

Some simple steps and commands to control a servo motor has been learned, Program is shown in Appendix (A).

CHAPTER 7: THE SECOND STAGE OF THE PROJECT

In the first stage, as a team work of two members, a solid knowledge and deep concepts have been covered by both members synchronously in the first stage of the presented design, which creating a strong base and clear insight motivating us to successfully achieve the desired target in perfect way. The second stage is managed and its plan is well-prepared to start with, carrying great enthusiasm and optimism to create a remarkable aimed participation of the growth development of prosthetic researches in our society which it needs the most.

Table (2) Gantt chart Of the Second Stage Design Work.

Task	Dec 1-7	Dec 8-14	Dec 15-21	Dec 22-28	Dec 29-Jan 6	Jan 7-13	Jan 14-20	Jan 21-27	Jan 28-Fab 3	Fab 4-10	Fab 11-17	Fab 18-24	Fab 25-Mar 1	Mar 2-8	Mar 9-15
Review the presented design	2 Weeks														
Advanced Wavelet studies		3 Weeks													
Study Support Vector Machine classifier					2 Weeks										
Implemented the national task							4 weeks								
Learning LabVIEW concepts										3 weeks					
Implementing and processing our collected data													2 Weeks		
Acquiring the practical data and implemented them by using Arduino														2 weeks	
Writing the final project essay				12 weeks											

a- Review the presented design:

Starting with a general review and finalizing the presented design.

b- Advanced Wavelet studies:

Getting a deeper understanding and advanced studies on the concept of wavelet, with its tools and advantages.

c- **Study Support Vector Machine classifier**

After testing and implementing group of classifiers by using MATLAB, the quadratic support vector machine had the best performance, so we have studied it is theatrical background.

d- Implementing the national task

Implementing the task, this is put by university of Patras.

e- Learning LabVIEW concepts

LabVIEW software is efficient in practical tasks.

f- Implementing and processing our collected data

The data was acquired using the Electrodes of the Electronic Prosthesis Engineering Team (EPET).

g- Acquiring the practical data and implemented them by using Arduino

Using the commercial electrodes to acquire the signals and processing them by using LabVIEW and controlling the servo motors by using Arduino.

Part III

Procedure and discussion

CHAPTER (8): OPEN AND CLOSE MOVEMENTS

Due to a complexity and non-stationarity of EMG, a large number of studies have focused on the investigation and evaluation of the optimal features obtained from wavelet coefficients. Many applications of pattern classification by wavelet analysis based on EMG feature extraction have been proposed, such as in sport science, determining muscle force and muscle fatigue, characterizing low-back pain, and identifying hand motions for the control of prostheses. Most studies to date have focused on applications classifying hand motions, and in our project EMG data were acquired from a dataset of two different hand motions for 50 EMG signals represent open hand motion and 50 EMG signals represent close hand motion. In this project, we propose that the features extracted from acquired raw EMG contribute to the two different muscular contraction classifications. The features which we using are based on the Wavelet Transform approach.

8.1 Introduction

Feature Extraction*:* The feature extraction stage involves the transformation of the raw signal to relevant data structure, called feature vector, by deleting noise and highlighting important data. The goal of feature extraction is to make computer-based systems distinguishing these signals. Many techniques of extracted features are implemented either based on time domain, frequency domain or time-frequency domain.

Wavelet Transform*:* Fourier analysis has the serious drawback that transitory information is lost in the frequency domain. This may be alleviated in the use of a short-time Fourier Transform (STFT),but when a time window is chosen, that window is the same for all frequencies of the signal being analyzed, causing possibly essential information to be lost at very low or high frequencies. The essential advantage of the wavelet transform over Fourier transform or STFT is that the time-frequency window is flexible and adapts in such a way that there is always about the same number of periods of the frequency analyzed in the time window.

Wavelet analysis allows the use of long time-intervals for more precise low frequency information, and shorter regions for more high frequency information. The wavelet transformation is achieved by the breaking up of a signal into shifted and scaled versions of the original (or mother) wavelet. Wavelets are able to determine if a quick transitory signal exists, and if so, can localize it. This feature makes wavelets very useful for the study of the EMG waveforms.

To achieve optimal performance in the wavelet analysis, a suitable wavelet function must be employed. Most studies of EMG analysis have concluded that the Daubechies (Db) wavelet family is the most suitable wavelet for EMG signal analysis, and in this project, the Daubechies wavelets, Db7, which are commonly used, were evaluated.

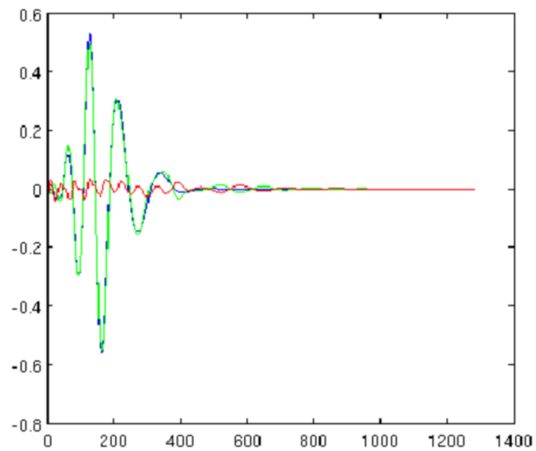


Figure (12): The Daubechies wavelets; Db7

The wavelet transform method can be divided into two types: discrete (DWT) and continuous (CWT). DWT was selected in our project because of its concentration on real-time applications. Briefly, the DWT technique iteratively transforms the signal of interest into multiresolution subsets of coefficients, and then the original EMG (S) is passed through high-pass and low-pass filters, the coefficients of the filters depending on the wavelet function type, to yield both a detailed coefficient subset (cD1) and an approximation coefficient subset (cA1) at the first level. To achieve a multi-resolution analysis, repetitious transformation is performed. This process is duplicated until the desired final level is yielded. Different levels of wavelet decomposition (n) were also evaluated in the experiments with the maximum level limited to 8 and the fixed sample length at 256 samples. As an illustration, if the decomposition level was set at 4, DWT generates respectively the coefficient subsets at the fourth level approximation (cA4) and the first to the fourth level details (cD1, cD2, cD3 and cD4).

To investigate the usefulness of extracting features from individual wavelet components instead of extracting them from all the components, the wavelet coefficient subsets (cD1 – cD4) can be used in combination as the presented signal that extracted the features of the two different motions of our 50 open/close subjects. In this project, the most widely used and most successful features based on the time domain and the frequency domains were evaluated. The mathematical definition of all the features is shown in Table (1):

8.2 Methodology:

For many signals, the low-frequency content is the most important part. It is what gives the signal its identity. The high-frequency content, on the other hand, imparts flavor or nuance. Consider the human voice. If you remove the high-frequency components, the voice sounds different, but you can still tell what's being said. However, if you remove enough of the low-frequency components, you hear gibberish. It is for this reason that, in wavelet analysis, we often speak of approximations and details. The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale, high-frequency components. the coefficients of the filters depending on the wavelet, which in our project is db7. The EMG signal is break into Decomposition (detailed and approximate) level which is we choose 4 levels is the optimum number of decomposition as shown in figure (3).

The combination of the four detailed signal (cD1-cD4) and the approximated signal are used in feature extraction rather than the raw EMG signal; to get clear and an accurate result taking advantage of the multi-resolution and high-accuracy characteristics of DWT.

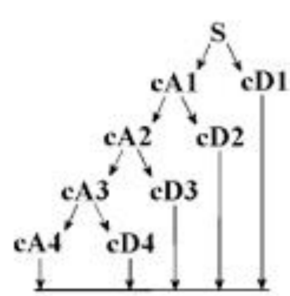


Figure (13): DWT decomposition tree from decomposition level 4.

Feature extraction	Mathematical definition
Integrated EMG (IEMG)	$\text{IEMG} = \sum_{n=1}^{N} \lvert x_n \rvert$
Mean absolute value (MAV)	$\text{MAV} = \frac{1}{N} \sum_{n=1}^{N} \lvert x_n \rvert$
Modified Mean Absolute Value (MMAV)	$\text{MMAV} = \frac{1}{N} \sum_{n=1}^{N} w_n \lvert x_n \rvert;$ $w_n = \begin{cases} 1, & if\ 0.25N \le n \le 0.75N \\ 0.5, & \text{otherwise} \end{cases}$
Simple Square Integral (SSI)	$\text{SSI} = \sum_{n=1}^{N} \lvert x_n \rvert^2$
Variance of EMG (VAR)	$\text{VAR} = \frac{1}{N-1} \sum_{n=1}^{N} x_n^2$
Root Mean Square (RMS)	$\text{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} x_n^2}$
Median Frequency (MDF)	$\sum_{j=1}^{\text{MDF}} P_j = \sum_{j=\text{MDF}}^{M} P_j = \frac{1}{2} \sum_{j=1}^{M} P_j$
Mean Frequency (MNF)	$\text{MNF} = \sum_{j=1}^{M} f_j P_j \Big/ \sum_{j=1}^{M} P_j$
Waveform length (WL)	$\text{WL} = \sum_{n=1}^{N-1} \lvert x_{n+1} - x_n \rvert$
Willison amplitude (WAMP)	$\text{WAMP} = \sum_{n=1}^{N-1} f\left(\lvert x_n - x_{n+1} \rvert\right);$ $f(x) = \begin{cases} 1, & if\ x \ge threshold \\ 0, & \text{otherwise} \end{cases}$

Table (3): Mathematical definition of some statistical features

The discrete wavelet transform decomposition tree from decomposition level 4 processes is implemented to the 100 data (50 Open-hand and 50 Close-hand) for the feature extraction issue. The following figure shows the DWT 4 level procedure to one of our EMG signals.

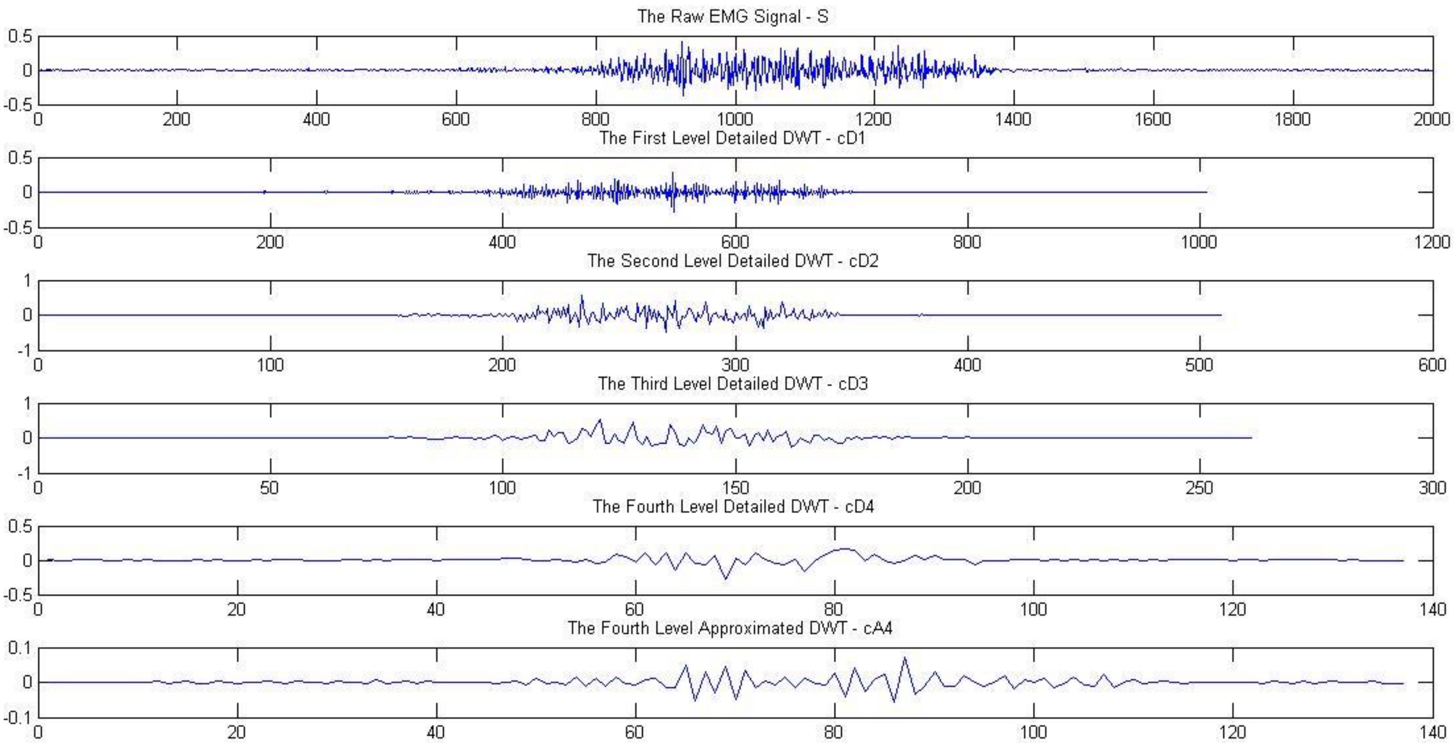


Figure (14): Example of the EMG signal using wavelet multi-resolution analysis with Db7 wavelet and 4-level decomposition.

To investigate the usefulness of extracting features from individual wavelet components instead of extracting them from all the components, the wavelet coefficient subsets matrices (cD1 – cD4) can be used in combination with (cA4) as the extracted features evaluated from some mathematical operations to reduce the matric into single value. The DWT coefficients matric experiment 10 different mathematical operation to extract the best representation for the DWT matric that represent one kind of motion, we have tried the 10 features and compare between them according to the separation among the two motion; this comparison can tell us clearly the best way to take advantage of the discrete wavelet transform coefficients and this powerful mathematical technique to represent non-stationary signals.

This methodology can be summarizing through the following stages at figure (15).

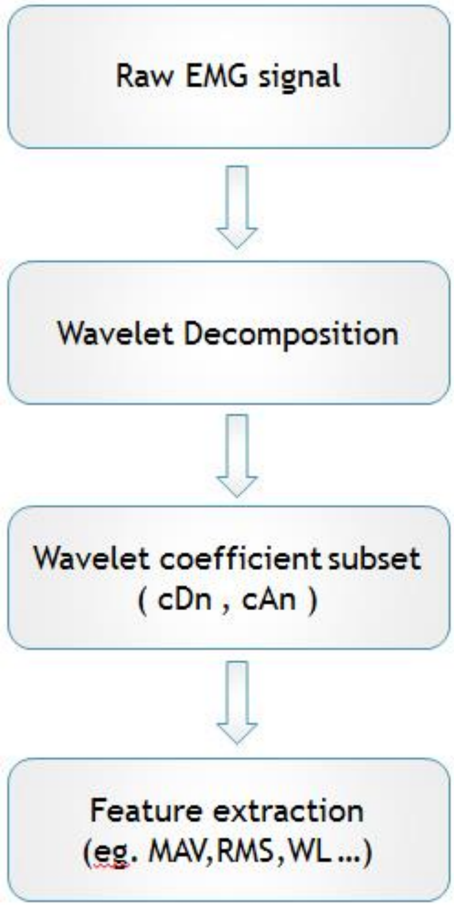


Figure (15): Procedure for the extraction of EMG features from the wavelet coefficients.

8.3 Evaluation:

Each EMG signal is transformed to DWT, the resulted signal coefficient is then using in feature extraction by implementing 10 different mathematical operations to DWT signal coefficient matric of the two different motion.

The Evaluation of the performance of the EMG features is a very critical process to be able to recognize the most reliable features extraction techniques.

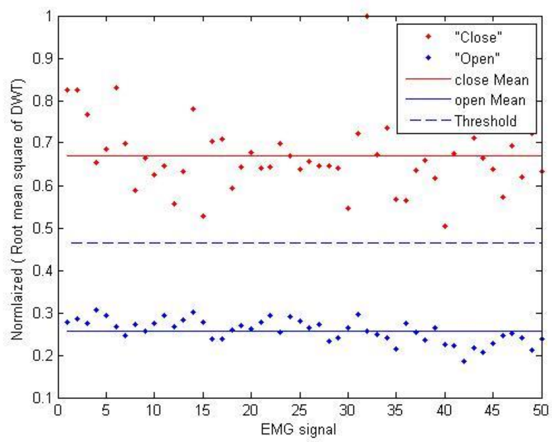
In this section we aim to show the evaluation result of our project; 100 EMG signal extracted (50 open, 50 closes), 7 features were extracted from the DWT coefficient signal, normalized and plotted. (For MATLAB program code; see Appendix A).

The result is as following:

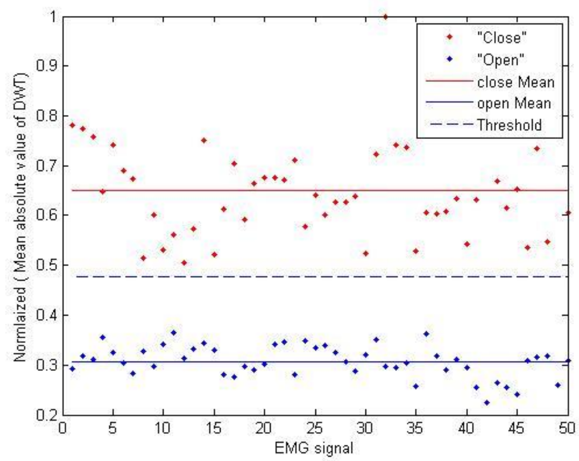
The Blue dots represent the 50 EMG close-hand signals, and the red dots represent the 50 open-hand signals. The threshold dashed line represents the separation difference between the two EMG motions.

The following results are the feature extraction of EMG signals using discrete wavelet transform with 8 different mathematical operations extracts the optimum features from its coefficients.

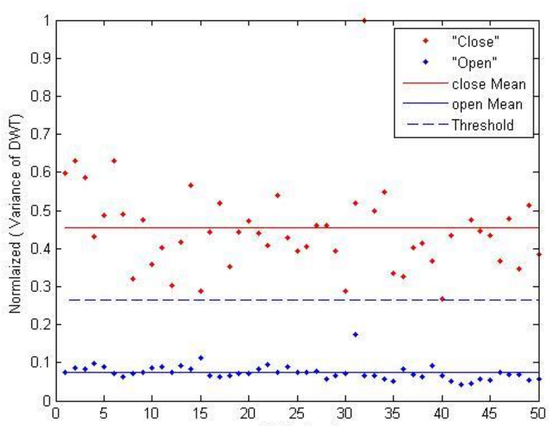
The Root Mean Square



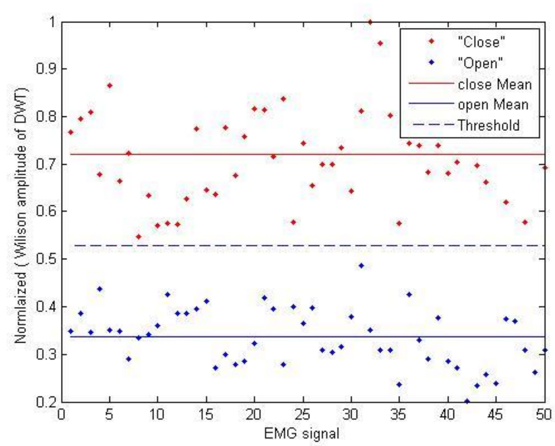
The Mean Absolute Value



The Variance



Willison Amplitude



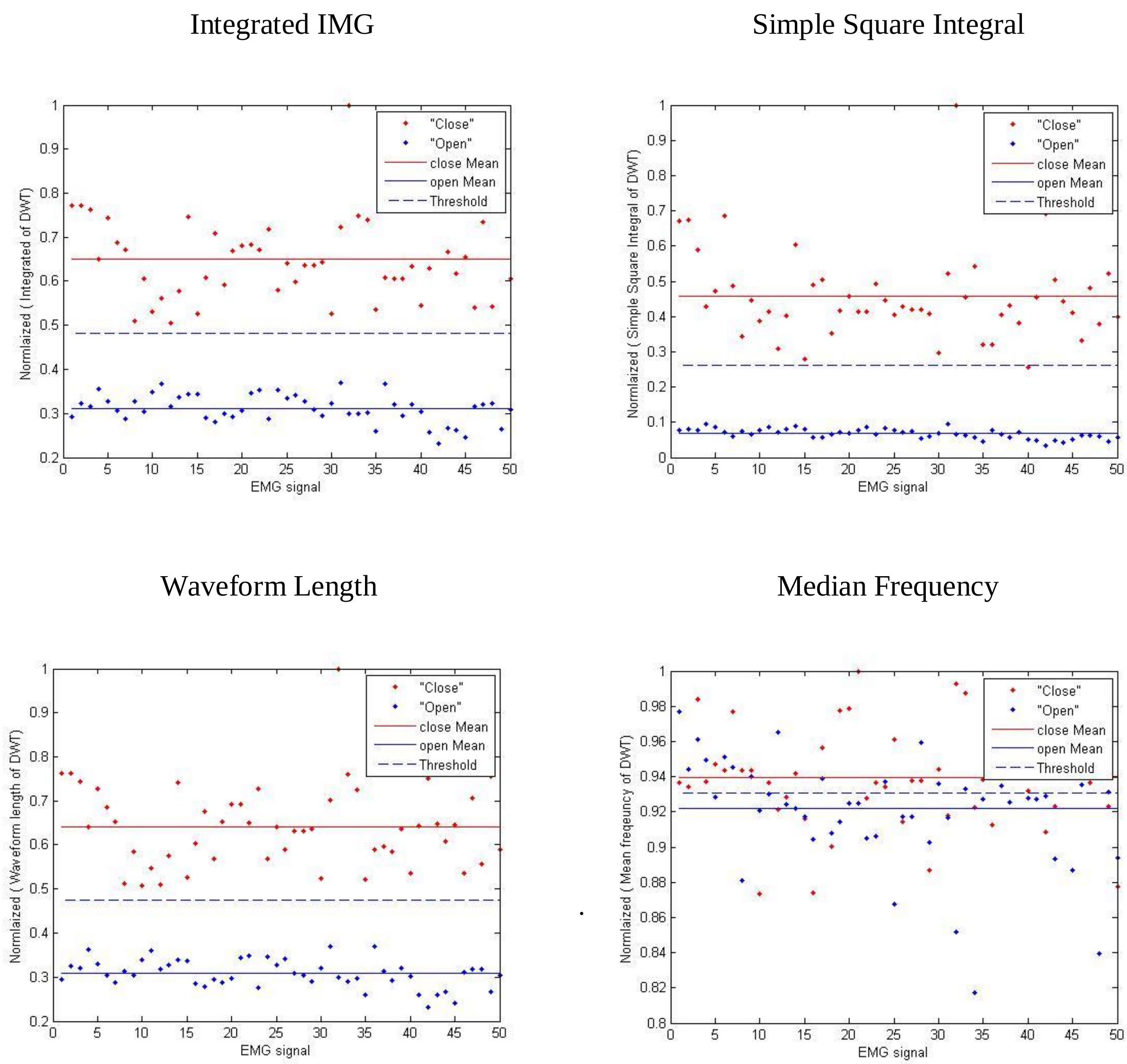


Figure (16): The results of feature extractions of EMG using DWT signals.

Our results show that the mean difference of the 50 signals of one motion is remarkably obvious and far from the other 50 signals at most the evaluated features; proves the high-resolution time-frequency representation that DWT provided to extract special separated features from its coefficients; expect for the mean frequency, since the EMG signal has almost the same frequency spectrum for all motion.

The following table shows the different evaluated features are ordered according to the best mean difference and separation between mean value of the extracted 50 open signals and the close signals.

Feature	Mean difference
Root Mean Square	0.4124
Simple square Integral	0.3886
Willison Amplitude	0.3857
Variance	0.3799
Mean Absolute Value	0.342
Integrated EMG	0.3394
Waveform length	0.3332
Mean frequency	0.0178

Table (4): Normalized Mean difference between open and close EMG signal

The usefulness of successful EMG features extracted from multiple-level decompositions of EMG based on DWT has been investigated in this section. Some useful EMG features are recommended, for instance, a feature vector extracted by using the RMS, SSI and WAMP features of the four level decomposition DWT of EMG signal with the Db7 wavelet, their use ensures that the resulting classification accuracy will be as high as possible and will also be better than signal extraction from the original EMG signal. The
results of the experiment reported in our project can be used in a wide class of clinical and engineering application

CHAPTER (9): SIX BASIC HAND MOVEMENTS

After successfully implemented and design a model to classify two movements (open, close), it was decided to go further to develop and improve the model to classify six different movements using two channel electrodes.

So, data set from (KIC Laboratory) research group was used to be analyzed and processed, the idea is to use a well-acquired data to verify our design and model.

9.1 Data Set Information:

9.1.1 Instrumentation:

The data were collected at a sampling rate of 500 Hz, using as a programming kernel the National Instrument€™s (NI) LabVIEW. The signals were band-pass filtered using a Butterworth Band Pass filter with low and high cutoff at 15Hz and 500Hz respectively and a notch filter at 50Hz to eliminate line interference artifacts.

The hardware that was used was an NI analog/digital conversion card NI USB- 009, mounted on a PC. The signal was taken from two Differential EMG Sensors and the signals were transmitted to a 2-channel EMG system by Delsys Bagnoli,, Handheld EMG Systems.

9.1.2 Protocol:

The experiments consisted of freely and repeatedly grasping of different items, which were essential to conduct the hand movements. The speed and force were intentionally left to the subject€™s will. There were two forearm surface EMG electrodes Flexor Capri Ulnaris and Extensor Capri Radialis, Longus and Brevis) held in place by elastic bands and the reference electrode in the middle, in order to gather information about the muscle activation.

The subjects were asked to perform repeatedly the following six movements, which can be considered as daily hand grasps:

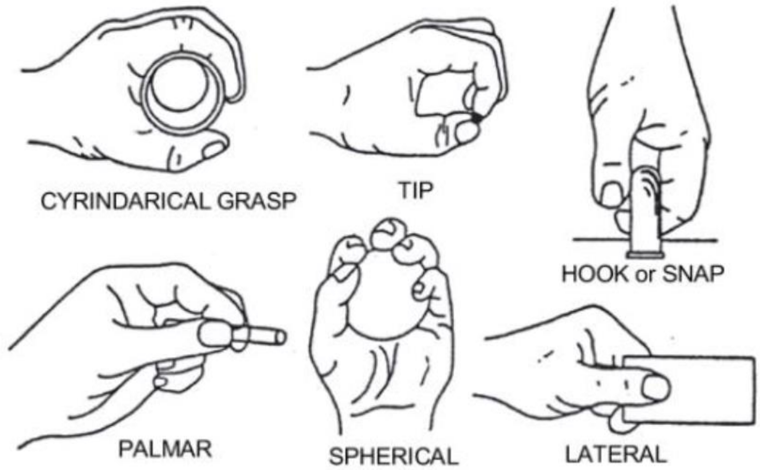


Figure (17): The six hand movements grasps.

a) **Spherical:** for holding spherical tools
b) **Tip**: for holding small tools
c) **Palmar**: for grasping with palm facing the object
d) **Lateral**: for holding thin, flat objects
e) **Cylindrical**: for holding cylindrical tools
f) **Hook**: for supporting a heavy load

5 healthy subjects (two males and three females) of the same age approximately (20 to 22-year-old) conducted the six grasps for 30 times each. The measured time is 6 sec. There is a mat file available for every subject.

9.1.3 Our protocol to the data:

Each data was named as xn_b, which n is the number of the sample (each sample represent a movement, and b is the channel number.

There have been 30 samples, 6 movements for 5 subjects, each sample has 30 trails.

N	subject	1: Spherical
1-6	Male 1	2: Tip
7-12	Male 2	3: Palmar
13-18	Female 1	4: Lateral
19-24	Female 2	5: Cylindrical
25-30	Female 3	6: Hook

Table (5): The protocol of arrangement the data

This protocol is used to get the best possible algorithm with minimum line of code

9.2 Preprocessing stage:

The data were collected at a sampling rate of 500 Hz; the signals were band-pass filtered using a Butterworth Band Pass filter with low and high cutoff at 15Hz and 500Hz respectively and a notch filter at 50Hz to eliminate line interference artifacts

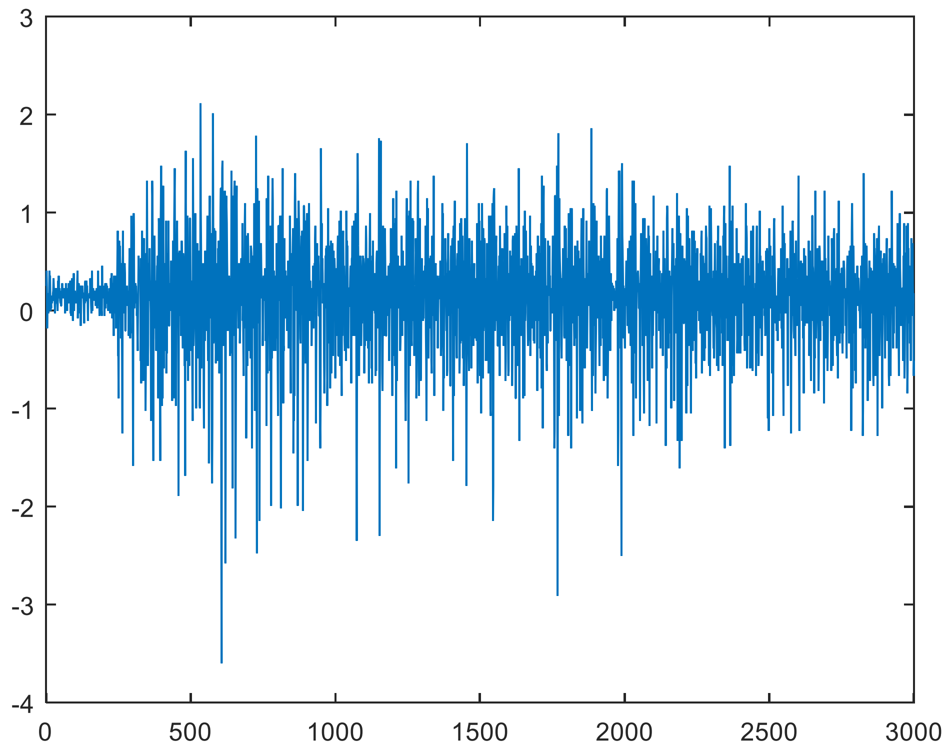


Figure (18): The raw signal of close movement after filtering

9.3 Feature extraction:

9.3.1 Statistical feature

The feature extractions were accomplished by using 10 statistical features:

1- Root Mean Square (RMS)
2- Willison Amplitude (WAMP)
3- Waveform length (WL)
4- Simple Square Integral (SSI)
5- Integrated EMG (IEMG)
6- Variance (VAR)
7- Mean Absolute Value (MAV)
8- Zero crossing (ZC)
9- Slope Sign Change (SSC)
10- Mean Frequency (MNF)

These features are implemented and applied into MATLAB by algorithms at (Appendix C)

Now the main algorithm is implemented and coded as follows:

MATLAB:

```
Tic
clear all;
f=[];
load('database.mat');
for a=1:2
    for i=1:30
for k=1:30
    v=eval(['x' int2str(i) '_' int2str(a)]);
x= v(k,:)';
 ls = length(x);
[cA,cD] = dwt(x,'db2');
 A = upcoef('a',cA,'db2',1,ls);
    x1=RMS(A);
    x2=WAMP(A,0.05);
    x3=WL(A);
    x4=SSI(A);
    x5=IEMG(A);
    x6=VAR(A);
    x7=MAV(A);
    x8=ZeroCross(A);
    x9=MNF(A);
    x10=SSC(A);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
    end
end
ch1=f(1:900,:);
```

```
ch2=f(901:1800,:);
X=table(ch1,ch2);
load('A5.mat');
X.activity=Y;
toc
```

Each feature computed to the 900 samples by the 2 channels, so the result of the procedure of feature extraction is a matrix with a 900 data of the row and 20 columns represent 10 features to one channel and 10 to the other.

Taking the average values of each feature in every movement at one channel , to show the result of the mathematical operation of the statistical feature, this table could show the differences in values that each feature do, to prepare them into the classification stage.

	RMS	WAMP	WL	SSI	IEMG	VAR	MAV	ZC	MNF	SSC
'cyl'	0.437207	2608.293	1059.419	771.0226	1000.972	0.236816	0.333657	878.32	134.2004	1771.827
'hook'	0.683516	2769.267	1631.313	1646.644	1439.008	0.528951	0.479669	1037.587	176.1564	1640.693
'lat'	0.229254	2409.753	477.2507	168.2901	547.1189	0.035769	0.182373	725.8	90.07274	1848.627
'palm'	0.349327	2512.02	761.8695	456.8389	795.4559	0.132162	0.265152	746.72	119.056	1774.747
'spher'	0.764387	2766.1	1839.815	2364.065	1680.662	0.768385	0.560221	993.94	163.7339	1585.627
'tip'	0.247608	2413.727	523.1315	206.1969	586.4852	0.048497	0.195495	685.08	98.39875	1903.213

Table (6): The average features values of each movement.

A performance compression between the 10 features has been studied, using the sum of the standard deviation that the feature makes at every movement

$$STD(i) = \frac{Xi - Xavg}{Xavg}$$

Where Xi is the average value of a particular movement

Xavg is the average value of all movement at a particular feature

	RMS	WAMP	WL	SSI	IEMG	VAR	MAV	ZC	MNF	SSC
'cyl'	0.032477	0.011021	0.010125	0.175826	0.007252	0.188328	0.007252	0.039956	0.030174	0.010093
'hook'	0.512594	0.073417	0.555409	0.760157	0.427185	0.812947	0.427185	0.228532	0.352244	0.064664
'lat'	0.49267	0.065936	0.544955	0.820109	0.457376	0.877405	0.457376	0.140632	0.308567	0.053876
'palm'	0.226953	0.026296	0.27358	0.511668	0.211079	0.547022	0.211079	0.115862	0.08608	0.011758
'spher'	0.691559	0.07219	0.75421	1.527035	0.666855	1.633588	0.666855	0.176853	0.256884	0.096057
'tip'	0.452053	0.064396	0.501209	0.779589	0.418333	0.833779	0.418333	0.188846	0.244654	0.084995
Sum	2.408306	0.313257	2.639489	4.574384	2.188081	4.893069	2.188081	0.890681	1.278602	0.321442

Table (7): The standard deviation of each feature on the movement

$$Xavg = \sum_{i=1}^{6} Xi$$

One of the performance parameters that can be measured to verify the feature efficiency is that each feature should make every movement deviated from other movements, to make it easy to distinguish them from each other, so in this comparison the sum of the standard deviations that each feature make is tested which gives the following result:

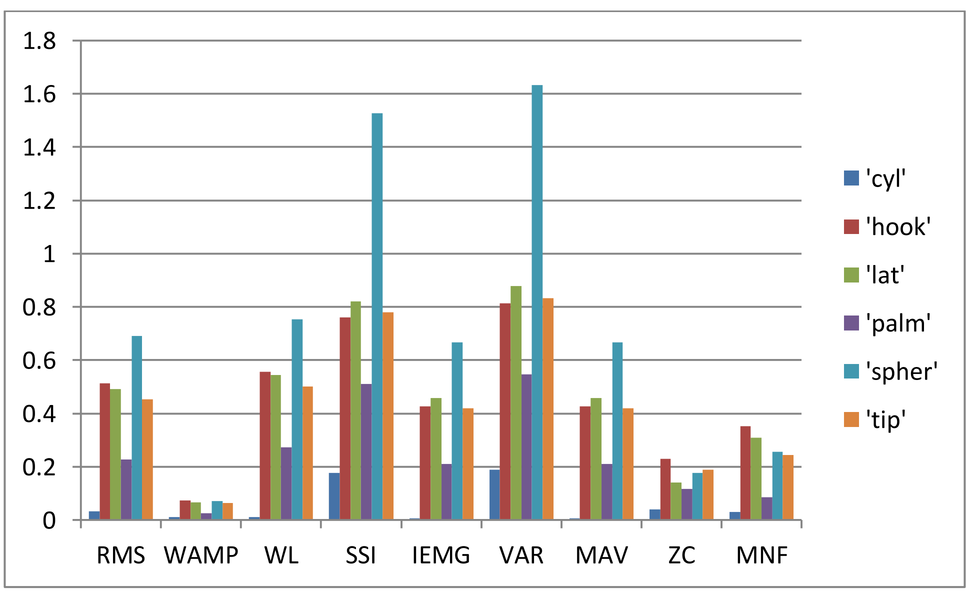


Figure (19): The graph of standard deviation

So we can compare the summation of standard deviation in one plot to get the best statistical features performance:

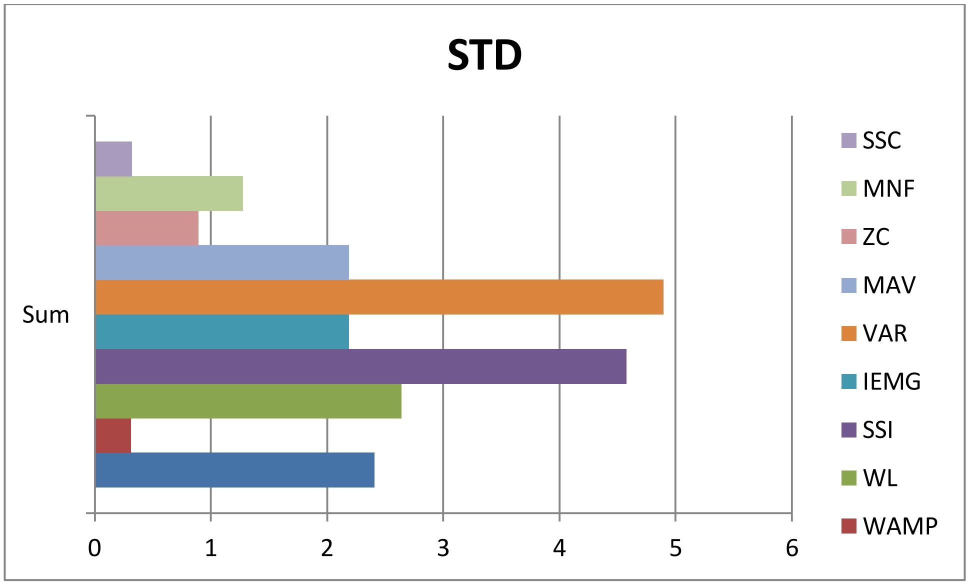


Figure (20): The summation of the standard deviation

The results show that Variance and slope sign change give obviously the best deviations among the experimented feature extractions that have been used in this study.

9.4 Classification:

Now after evaluating the feature extraction stage, the result of this procedure is a matrix with a 900 data of the row and 20 columns represent 10 features to one channel and 10 to the other. Each of the 6 movements has 180 datasets will be used to classifying the six labels.

The dataset used in the training process is 720, and 180 samples are used as testing to the classifier performance, with 20% cross validation.

A plenty of classifiers are used to examine the best classifier can be used in this data, the classifiers used are:

- Quadratic support vector machine.
- Cubic support vector machine.
- Cubic K-Nearest Neighbor.
- Fine K-Nearest Neighbor

The study on the best classifier among them was done, to use the best possible classifier to our model. The result shows the classifier accuracy of each classifier as follows:

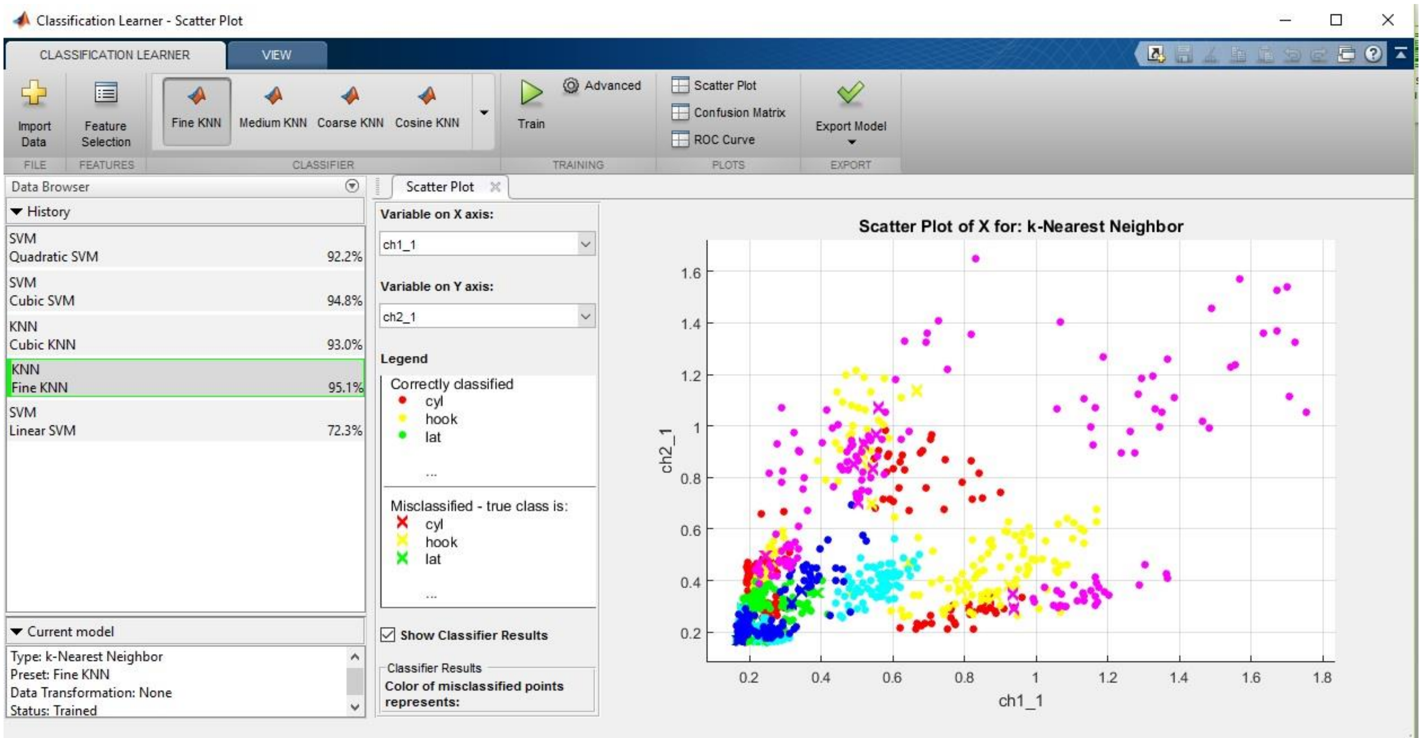


Figure (21): The Classification learner App results

Classifier	Accuracy
Fine KNN	95.1%
Cubic SVM	94.8%
Cubic KNN	93.0%
Quadratic SVM	92.2%

Table (8): The Accuracy percentage of each classifier

The results show that the best accuracy percentage is given by using Fine KNN classifier and cubic SVM.

9.5 Wavelet transform

The project object is compare between using wavelet or not, so the same technique as the above has repeated. The results are shown as:

In MATLAB:

```
x= v(k,:)';
 ls = length(x);
[cA,cD] = dwt(x,'db2');
 A = upcoef('a',cA,'db2',1,ls);
```

The classifiers results:

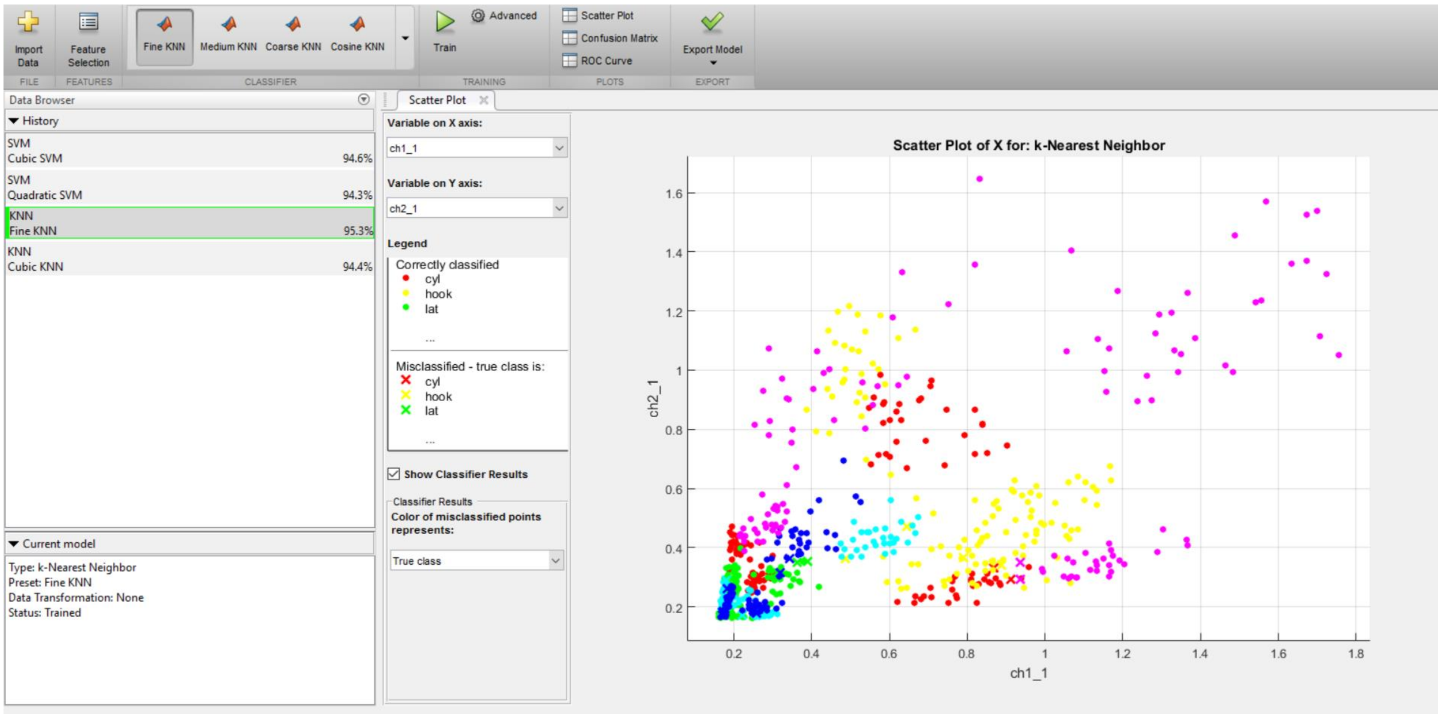


Figure (22): The Classification learner App results with wavelets.

Classifier	Accuracy
Fine KNN	95.1%
Cubic SVM	94.8%
Cubic KNN	93.0%
Quadratic SVM	92.2%

Table (9): The Accuracy percentage of each classifier

CHAPTER (10): IMPLEMENTATION OF OUR SIX MOVEMENTS ACQUIRED DATA

After designing a model and a complete algorithm to a system of six hand movements classification, using a data set from (KIC Laboratory) research group, now is the time to use and acquire our own data, to model our desired design.

10.1 Data Set Information:

10.1.1 Instrumentation:

The data was acquired using the Electrodes of the Electronic Prosthesis Engineering Team (EPET)

Figure (23): The double sides of (EMG) sensor circuit designed by EPET

The EPET has designed and built a surface Electromyography (sEMG) sensor, shown in Fig. n, to collect the EMG signal from the residual limb muscle. The collected signals are amplified 1000 times and then entered to the computer and saved into MATLAB as MAT files.

10.1.2 Protocol:

The experiments consisted of freely and repeatedly 6 different hand movements of one subject 30 times, the speed and force were all as normal movements.

The subject performed repeatedly the following six movements:

a) **Close**: closing the five fingers of the hand.
b) **Open**: opening the five fingers of the hand.
c) **Rotate inside**: Rotate the wrist toward the inside.
d) **Rotate outside**: Rotate the wrist toward the outside.
e) **Left**: moving the wrist toward the left.
f) **Right**: moving the wrist toward the right.

The database: 1 healthy subject (male, 23-year-old) conducted the six movements for 30 times. The measured time is 1.8 sec.

10.2 Data acquisition:

The electrode was connected to computer using the sound plug, the data acquired as a sound by *Audacity* open source software at 44100 Hz, every movement has been repeated 30 times, so every record has a nearly 1 minute and 30 seconds with fs = 44Khz has a result of a vector of 4224640 samples.

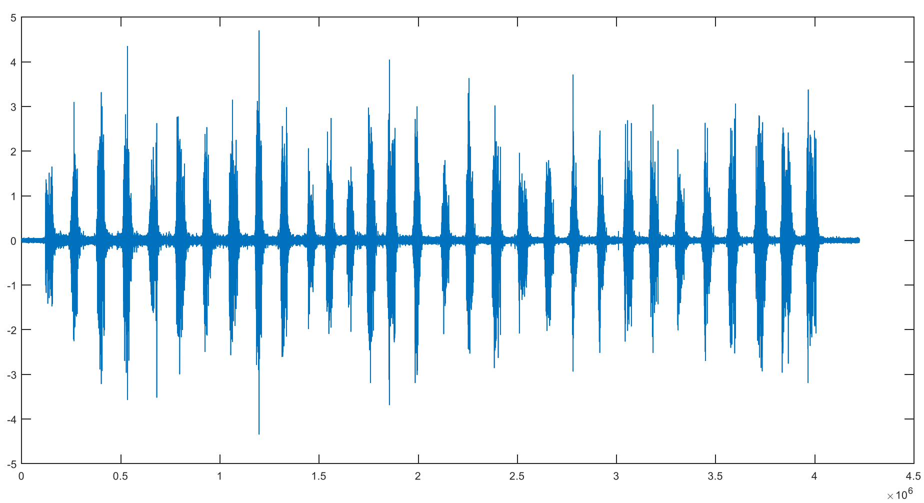


Figure (24): The raw signal of acquired experiment close 30 trails

The 30 samples of the movement have been cut by MATLAB into 30 matrices (y1, y2, and y3… y30) saved as mat file by the name of the movement (ex: open. Mat), each mat file has the 30 samples at its workspace when it is loaded.

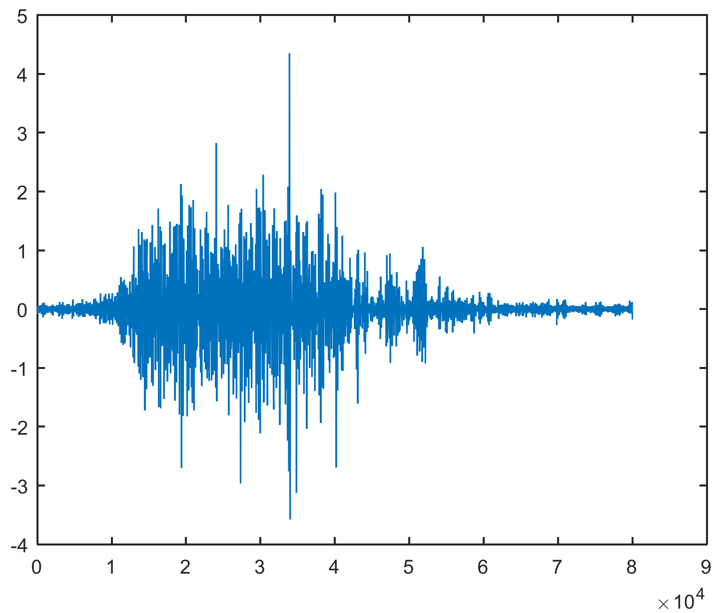


Figure (25): The one sample raw signal of close hand

10.3 Data Preprocessing:

The signals were band-pass filtered using a Butterworth Low Pass filter cutoff frequency at 500Hz and a notch filter at 50Hz to eliminate line interference artifacts.

The filters are built and designed using *Filter Design & Analysis Tool*

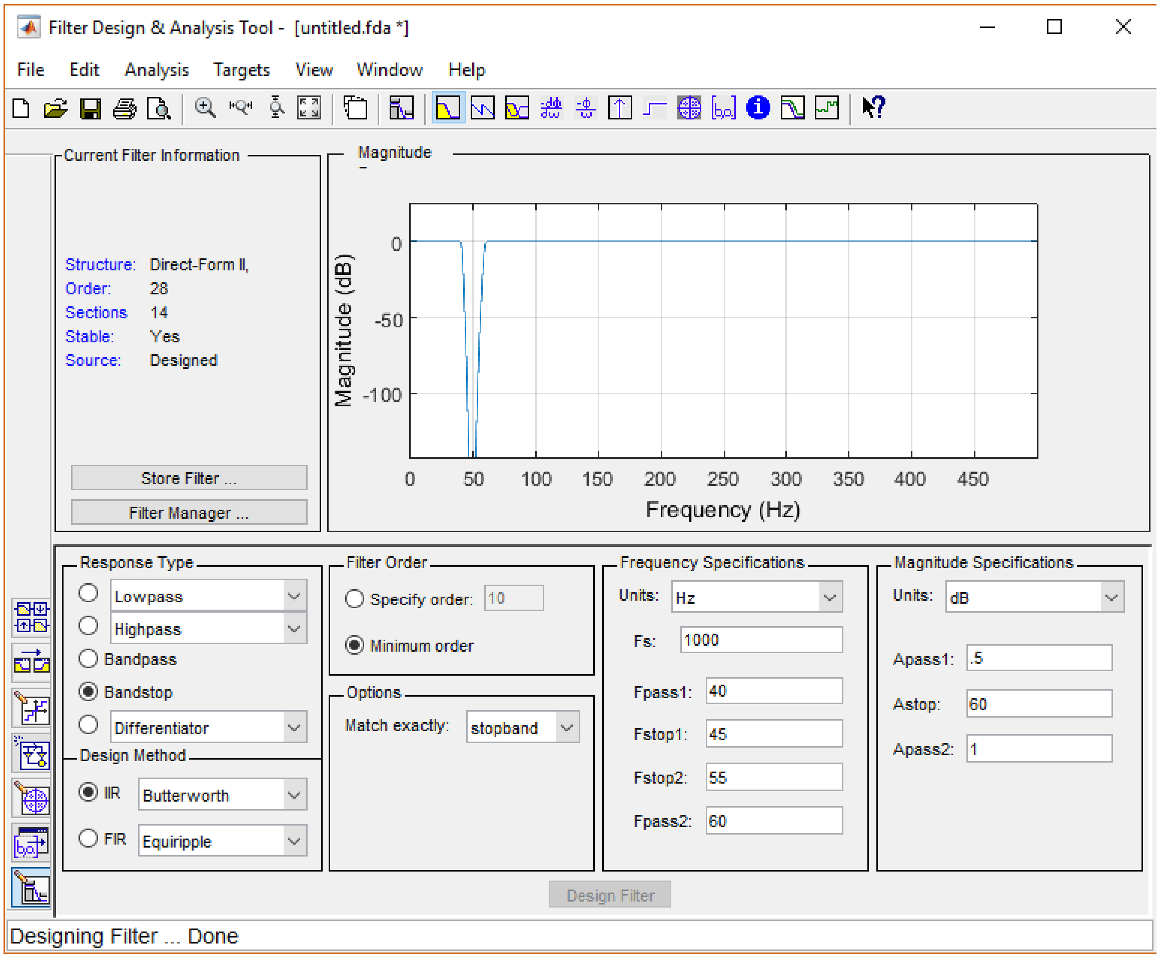


Figure (26): The filter design and analysis Toolbox in MATLAB

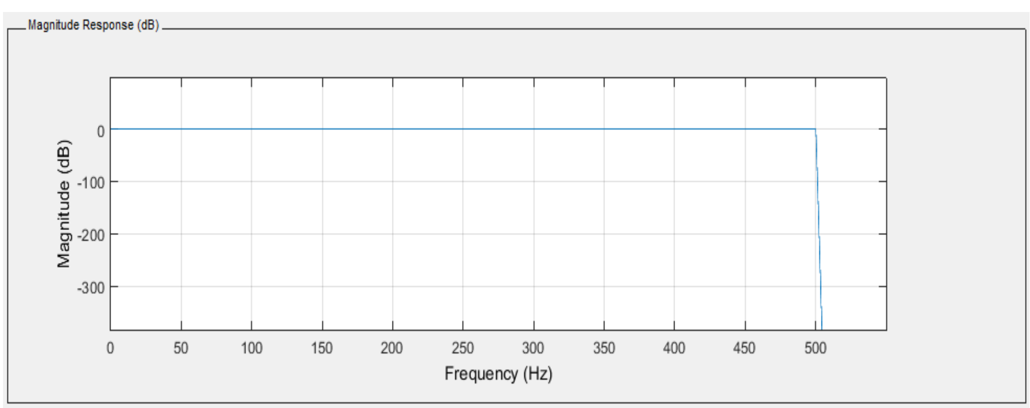


Figure (27): low pass filter with fc=500Hz

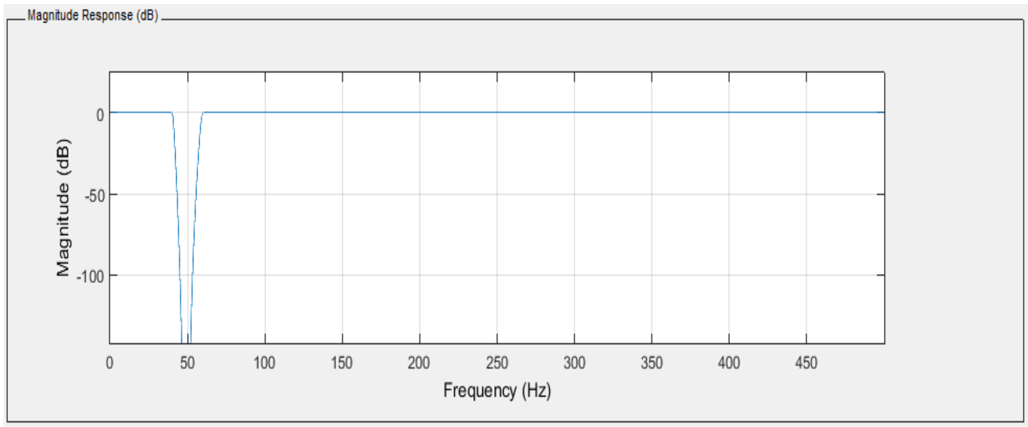


Figure (28): Notch filter with at 50Hz

10.4 Feature extraction:

The feature extractions were accomplished by using the same 10 statistical features that has been used in our previous model, each movement has 30 samples with 10 different features, so the result of the final matrix used into the classification would be with 10 columns “features” and 180 row “30 samples × 6 movements”.

The algorithm:

In MATLAB:

```
Tic
clear all;
load('close.mat');
f=[];
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
```

```
%-------------------------------------------------
load('open.mat');
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
%------------------------------------------------------

load('Rotateinside.mat');
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
%------------------------------------------------------

load('Rotateoutside.mat');
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
%-------------------------------------
load('left.mat');
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
```

```
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end
%-------------------------------------
load('Right.mat');
for i=1:30
    x=eval(['y' int2str(i)]);
  x1=RMS(x);
    x2=WAMP(x,0.05);
    x3=WL(x);
    x4=SSI(x);
    x5=IEMG(x);
    x6=VAR(x);
    x7=MAV(x);
    x8=ZeroCross(x);
    x9=MNF(x);
    x10=SSC(x);
    f=[f;x1,x2,x3,x4,x5,x6,x7,x8,x9,x10];
end

Y1={'close'};
Y1(1:30,:)={'close'};
Y1(31:60,:)={'open'};
Y1(61:90,:)={'Rotateinside'};
Y1(91:120,:)={'Rotateoutside'};
Y1(121:150,:)={'left'};
Y1(151:180,:)={'right'};

X=table(f);
%load('Y1.mat');
%Y2=[Y1;Y1;Y1;Y1;Y1];
X.activity=Y1;
toc
```

10.5 Classification:

Now we want to design the classifier –based on the feature extraction matrix- to classify the desired movements

The plan is to first start with two movements and classifies them, then develop and improve new movement to the model until we eventually end up with a classification model and accuracy percentage of the all stages.

Using Classification Learner Tool, as shown in the previous task, using Quadratic support vector machine, with 10% cross validation, every stage is evaluated and studied its accuracy percentage

NO. of movements	The movements	The classifier accuracy
2	Close -Open	100%
3	Close - Open- Rotate inside	98.9%
4	Close- Open- Rotate inside - Rotate outside	96.7%
5	Close- Open- Rotate inside – Rotate outside - left	95.3%
6	Close- Open- Rotate inside – Rotate outside – left - Right	91.1%

Table (10): The model result based on the number of movements

CHAPTER (11): PRACTICAL REAL-TIME HAND MOVEMENT MODEL

In practical acquiring, MATLAB is not reliable because it not compatible with the sound card .so the LabVIEW software provides reliability and is compatible with the sound card.

LabVIEW: Software is used for a wide variety of applications and industries.

The practical part of the project has accomplished four hand movements:

1- Open.
2- Close.
3- Pointing.
4- Penciling.

11.1 Protocol:

We have used the Electrodes of the Electronic Prosthesis Engineering Team (EPET). The commercial electrode has used to acquire the EMG signal, while the LabVIEW "Acquire Sound Block" has sampled the signal at rate 44100 Hz. then the signal modified and pass through two types of filter which are they:

1- band-pass filter at (15 Hz to 500 Hz).

2- Notch filter at 50 Hz.

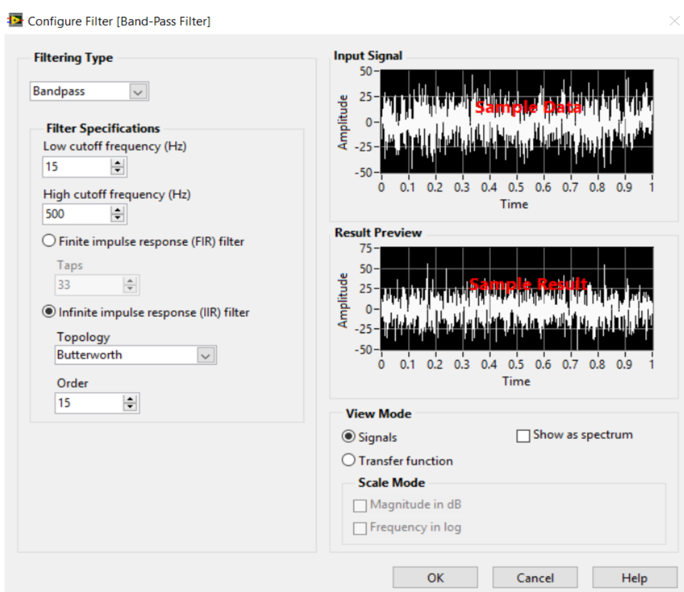


Figure (29): filter box in LabVIEW.

After that the filtered signal will go through statistical feature block, which processed three features (RMS, VAR and maximum value) for every EMG signal. By using the shown block:

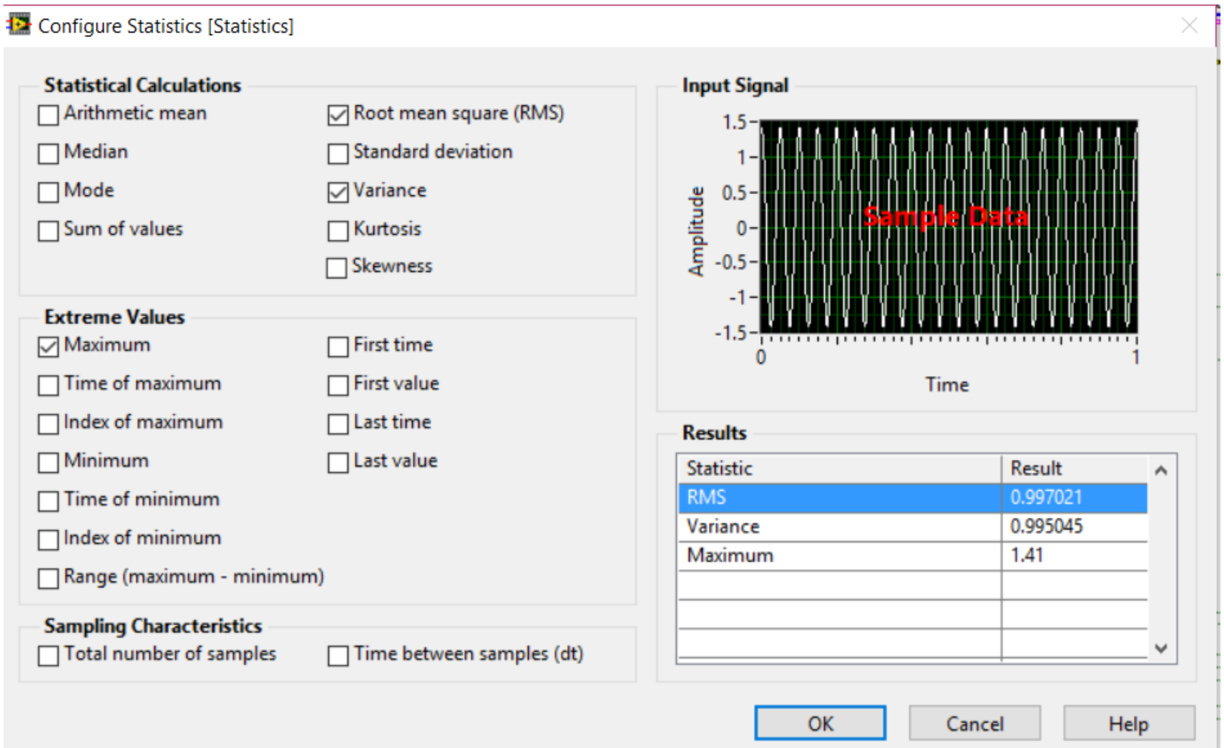


Figure (30): Statistic Toolbox in LabVIEW

11.2 The classifier mechanism:

The conditional thresholds are the principle that we have followed:

1- Open: if RMS< 0.025, VAR<0.05 and MAX<0.1, results (1, 2, 3) are off.

2- Close: if RMS>0.11, VAR>0.13 and MAX>0.5, results (1, 2, 3) are on.

3- Pointing: if RMS<0.11, VAR<0.13 and MAX<0.5, results (1, 2) are on and result (3) is off.

4- Penciling: if RMS>0.025, VAR>0.05 and MAX>0.1, results (1, 2) are off and result (3) is on.

Note: result refers to the led that has a name is "result"

We have showed the practical results on Arduino Uno to be more understandable, we have done that y downloading LabVIEW toolbox to communicate with Arduino, different types of blocks in this toolbox was used like (Digital write, Digital read, etc.).

The results (1, 2, 3) will be shown in the outputs (8, 10, 12).

Arduino code for interfacing with LabVIEW:

File Edit Sketch Tools Help

EMG | AFMotor.cpp | AFMotor.h | AccelStepper.cpp | AccelStepper.h | IRremote.cpp | IRremote.h | IRremoteInt.h | LabVIEWInterface.h | LabVIEWInterface

```
/*********************************************************************************
 **
 **  LVFA_Firmware - Provides Basic Arduino Sketch For Interfacing With LabVIEW.
 **
 **  Written By:    Sam Kristoff - National Instruments
 **  Written On:    November 2010
 **  Last Updated:  Dec 2011 - Kevin Fort - National Instruments
 **
 **  This File May Be Modified And Re-Distributed Freely. Original File Content
 **  Written By Sam Kristoff And Available At www.ni.com/arduino.
 **
 *********************************************************************************/


/*********************************************************************************
 **
 ** Includes.
 **
 ********************************************************************************/
// Standard includes.  These should always be included.
#include <Wire.h>
#include <SPI.h>
#include <Servo.h>
#include "LabVIEWInterface.h"

/*********************************************************************************
 **  setup()
 **
 **  Initialize the Arduino and setup serial communication.
 **
 **  Input:  None
 **  Output: None
 *********************************************************************************/
void setup()
{
  // Initialize Serial Port With The Default Baud Rate
  syncLV();

  // Place your custom setup code here

}
```

LabVIEW block diagram:

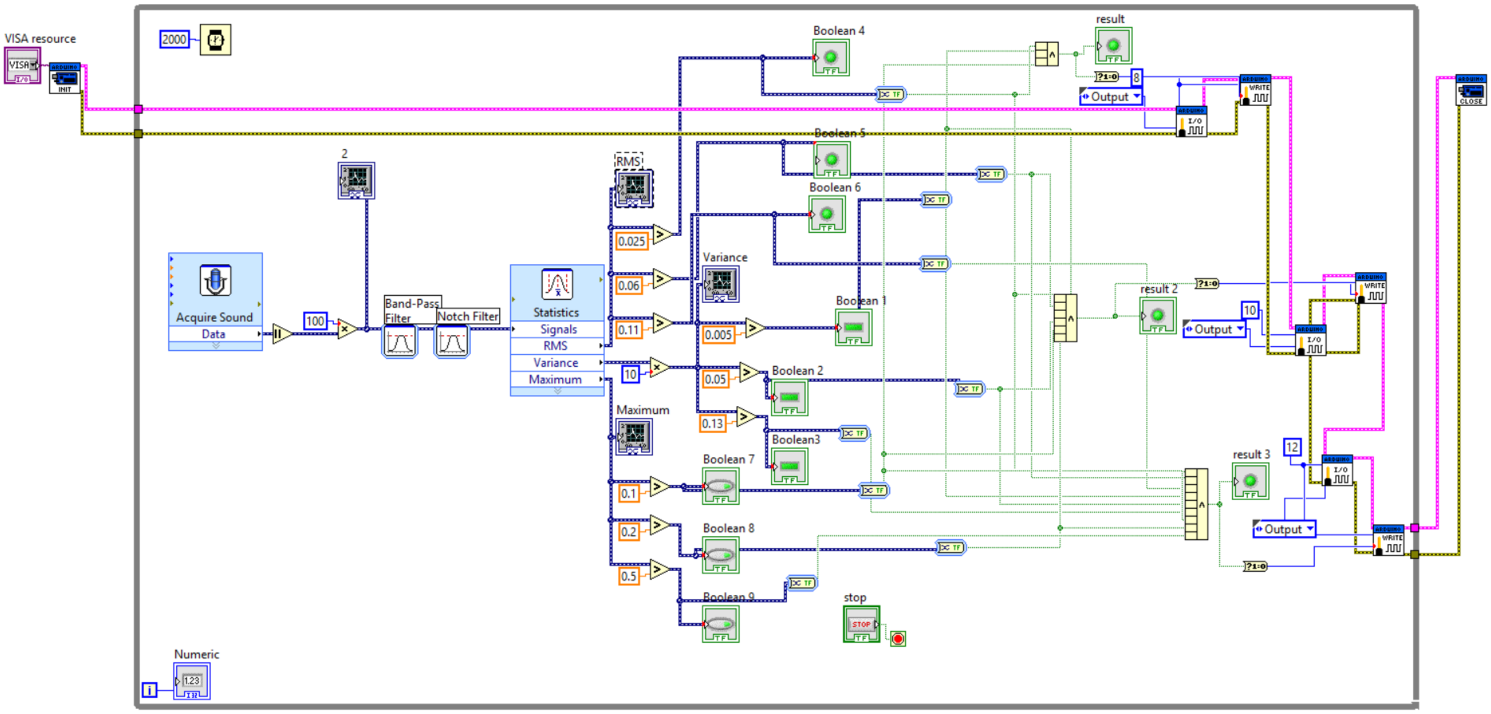


Figure (31): Block diagram of the designed model

LabVIEW front panel:

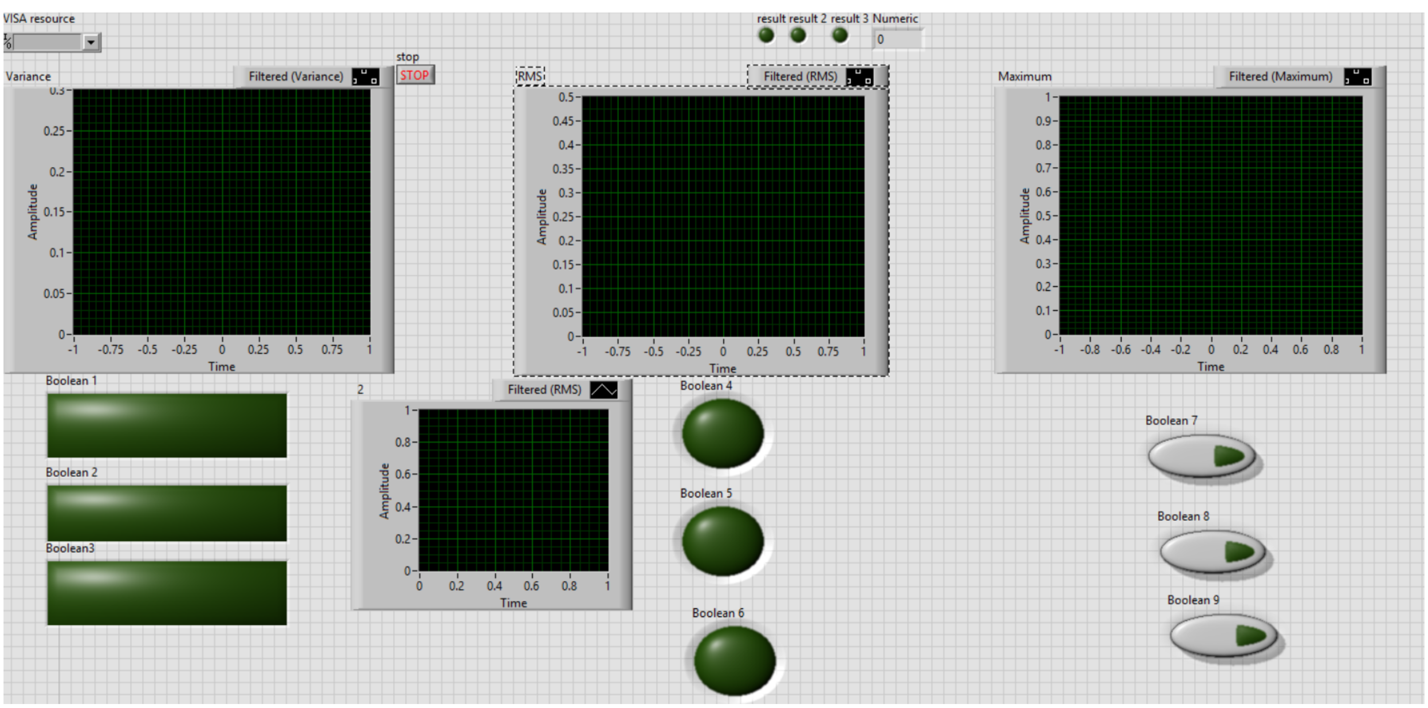


Figure (32): Front panel of the mod

Part IV

Conclusion and Future Work

CHAPTER (12): CONCLUSION

- For the open- close movements part the study shows the effectiveness of the features:

Feature	Mean difference
Root Mean Square	0.4124
Simple square Integral	0.3886
Willison Amplitude	0.3857
Variance	0.3799
Mean Absolute Value	0.342
Integrated EMG	0.3394
Waveform length	0.3332
Mean frequency	0.0178

Table (2): Normalized Mean difference between open and close EMG signal

The usefulness of successful EMG features extracted from multiple-level decompositions of EMG based on DWT has been investigated in this paper. Some useful EMG features are recommended, for instance, a feature vector extracted by using the RMS, SSI and WAMP features of the four-level decomposition DWT of EMG signal with the Db7 wavelet, their use ensures that the resulting classification accuracy will be as high as possible and will also be better than signal extraction from the original EMG signal. The results of the experiment reported in our project can be used in a wide class of clinical and engineering application.

- For the basic hand movements, the processing was accurate because we have used many types of statistical features, after that we examined many types of classifiers.
  Fine KNN had the best performance, so it is the highly recommended classifier. The main goal of this task is to compare between using Wavelet Transform or not.
  The results are shown that, Wavelet Transform has slightly high accuracy, because it allows the use of long time intervals where we want more precise low frequency information, and shorter regions where we want high frequency information, however, Wavelet Transform is required more time because of it is complex math, so we conclude before design and implement any project we have to find the best processing for both accuracy and time, in other word it is a compromise.

- For the data that we have collected by using Electrodes of the Electronic Prosthesis Engineering Team (EPET) which was a magnificent approach, also we have to put in our mind we have used only one channel, therefore, in our research we have got accuracy 91% with different six hand movements and also, we used local design electrodes and one channel is a great achievement.

  Apparently, this field has a great and promise future beside there is a massive need for the participation of growth the developing and building of prosthetic limbs design so, this field has a high priority and it is significant task.

- Basically, the researches in the field of designing prosthetic hands are highly recommended, but there is another approach which is the manufacturing, which is the main goal of our project.

  We have done that by using LabVIEW, which has a better performance in real-time application than other types of software like MATLAB.

CHAPTER (13): FUTURE WORK

- The data set from (KIC Laboratory) research has done with high and reliable accuracy; therefore, the upcoming object in this task is to achieve perfect accuracy by improving the processing techniques, which are going to be done by evolving the feature extraction selection and machine learning combination.

- Improving the signal acquisition techniques that were collected by commercial electrodes, also increasing the number of channels and finding better position to place them.

- Implementing advanced signal processing in LabVIEW, which is made the practical performance more accurate, in spite of it will cause more delay than simple processing which is significant drawback.

- Designing Robotic hand compatible with the practical hand movements, it has high priority in the future by using 3D printer, microprocessor and servo motors, in other word designing prosthetic hands for manufacturing.

- Testing the data set that was collected in random subjects and examine the processing with and without Wavelet Transform to find out the Wavelet Transform performance with completely non-stationary signals.

IV - References:

[1] – World Health Organization, “World Report on Disability” , Publishing of theWHO, Malta, 2011.

[2] - Ed. Joseph D. Bronzino ,"The Biomedical Engineering Handbook: Second Edition", University of Montreal, Canada , 2000.

[3] - Edited by Warren E. Finn, Peter G. LoPresti, "Handbook of neuroprosthetic methods" , Boca Raton, Florida , 2003.

[4] - Richard F. ff. Weir, Ph.D , Northwestern ,"Design of Artificial Armand Hands for Prosthetic Applications" , University Prosthetics Research Laboratory, and Rehabilitation Engineering Research Center, Chicago, Illinois , 2004.

[5] - P. Parker , K. Englehart, B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses" , Institute of Biomedical Engineering, Department of Electrical and Computer Engineering, University of New Brunswick, 2006.

[6] - Edited by Alexander Poularikas ,"THE Electrical Engineering and Applied Signal Processing Series" , Boca Raton, Florida, 2002.

[7] - Miroslav Kubat ,"An Introduction To Machine Learning" , University of Miami Coral Gables, FL, USA, International Publishing Switzerland 2015.

[8] - Jonghwa Kim, Stephan Mastnik, Elisabeth André , "EMG-based Hand Gesture Recognition for Realtime Biosignal Interfacing" , Lehrstuhl für Multimedia Konzepte und ihre Anwendungen Eichleitnerstr. 30, D-86159 Augsburg, Germany, 2008.

[9] - Eldin Henry Shroffe and P.Manimegalai , "Hand Gesture Recognition Based on EMG Signals Using ANN" , Dept. of Electronics and Instrumentation , Karunya University , April 2013.

[10] – Gang Wang, Yanyan Zhang and jueWang , The Analysis of Surface EMG Signals with thw Wavelet-Based Correlation Dimension Methode , Xian jiaotong University , Hindawi Publishing Corporation , 2014.

Appendix A:

In MATLAB:

```
%Mean absolute value MAV
fp=[];fc =[];
threshold =.05;
for i= 1:50
    eval(['load p' int2str(i) ';'] );
    eval(['p' int2str(i) '=y;'] );
    f1=MAV(y);
    fp=[fp;f1];
    a=max(fp);
end

for i= 1:50
    eval(['load c' int2str(i) ';'] );
    eval(['c' int2str(i) '=y;'] );
    f2=MAV(y);
    fc=[fc;f2];
    b=max(fc);
end
c=max(a,b);
figure(1)
plot(fc(:,1)/c,'r.','MarkerSize',11)
hold on
plot(fp(:,1)/c,'b.','MarkerSize',11)
MEANc(1:50)=mean(fc(:,1)/c);
MEANp(1:50)=mean(fp(:,1)/c);
Threshold=0.5*abs(MEANp+MEANc);
plot(MEANc,'-r','MarkerSize',11)
plot(MEANp,'-b','MarkerSize',11)
plot(Threshold,'--','MarkerSize',11)
xlabel('EMG signal')
ylabel('Normlaized ( Mean Absolute Value ) ')
 legend('"Close"','"Open"','close Mean','open Mean','Threshold')
clear all
%-----------------------------------------------------------------------
%Variance VAR
fp=[];fc =[];
threshold=.05;
for i= 1:50
    eval(['load p' int2str(i) ';'] );
    eval(['p' int2str(i) '=y;'] );
    f1=var(y);
    fp=[fp;f1];
    a=max(fp);
end
```

```
for i= 1:50
    eval(['load c' int2str(i) ';'] );
    eval(['c' int2str(i) '=y;'] );
    f2=VAR(y);
    fc=[fc;f2];
    b=max(fc);
end
c=max(a,b);
figure(2)
plot(fc(:,1)/c,'r.','MarkerSize',11)
hold on
plot(fp(:,1)/c,'b.','MarkerSize',11)
MEANc(1:50)=mean(fc(:,1)/c);
MEANp(1:50)=mean(fp(:,1)/c);
Threshold=0.5*abs(MEANp+MEANc);
plot(MEANc,'-r','MarkerSize',11)
plot(MEANp,'-b','MarkerSize',11)
plot(Threshold,'--','MarkerSize',11)
legend('VAR "Close"','VAR "Open"','close Mean','open Mean','Threshold')
ylim([0 1])
xlabel('EMG signal')
ylabel('Normlaized (  Variance of EMG  ) ')
clear all
%-----------------------------------------------------------------------
%RMS
fp=[];fc =[];
threshold=.05;
for i= 1:50
    eval(['load p' int2str(i) ';'] );
    eval(['p' int2str(i) '=y;'] );
    f1=RMS(y);
    fp=[fp;f1];
    a=max(fp);
end

for i= 1:50
    eval(['load c' int2str(i) ';'] );
    eval(['c' int2str(i) '=y;'] );
    f2=RMS(y);
    fc=[fc;f2];
    b=max(fc);
end
c=max(a,b);
figure(3)
plot(fc(:,1)/c,'r.','MarkerSize',11)
hold on
plot(fp(:,1)/c,'b.','MarkerSize',11)
MEANc(1:50)=mean(fc(:,1)/c);
MEANp(1:50)=mean(fp(:,1)/c);
Threshold=0.5*abs(MEANp+MEANc);
```

```
plot(MEANc,'-r','MarkerSize',11)
plot(MEANp,'-b','MarkerSize',11)
plot(Threshold,'--','MarkerSize',11)
legend('RMS "Close"','RMS "Open"','close Mean','open Mean','Threshold')
xlabel('EMG signal')
ylabel('Normlaized ( Root Mean Square  ) ')
clear all
```

Appendix B:

In Arduino Uno :

```
// Sweeps a RC servo motor back and forth across 180 degrees.

#include <Servo.h>

Servo myservo;

int pos = 0;    // variable to store the servo position

void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
  for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
    // in steps of 1 degree
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(15);                       // waits 15ms for the servo to reach the position
  }
  for (pos = 180; pos >= 0; pos -= 1) {
    myservo.write(pos);
    delay(15);
  }
}
```

Appendix C:

In MATLAB:

```
%Root Mean Square (RMS)
function[RMS]=RMS(y)
    N=length(y);
    RMS= sqrt(sum(y.^2)/N);
End
```

In MATLAB:

```
%Mean Absolute Value
function[MAV]=MAV(y)
    N=length(y);
   MAV=1/N*sum(abs(y));
End
```

In MATLAB:

```
%Zero Crossing
function [c]=ZeroCross(g)
c=0;
for i=1:length(g)-1
if sign(g(i+1))~=sign(g(i))
    c=c+1;
end
end
```

In MATLAB:

```
% Mean Frequency (MNF) is the average frequency.
function[MNF]=MNF(y)
L=length(y);
NFFT = 2^nextpow2(L);
Y = fft(y,NFFT)/L;
f = 1000/2*linspace(0,1,NFFT/2);
mx = abs(Y(1:NFFT/2));
mx=2*(2*mx).^2;
MNF=sum(f.*mx')/sum(mx);
end
```

In MATLAB:

```
%Waveform length (WL) is the cumulative length of the
function[WL]=WL(y)
    WL=[y';0 y(1:length(y)-1)'];
    WL=sum(abs([WL(1,:)-WL(2,:)]))-abs(WL(1));

end
```

In MATLAB:

```
%Willison amplitude (WAMP)
function[WAMP]=WAMP(y,threshold)
h=[y';0 y(1:length(y)-1)'];
   WAMP=(abs([h(1,:)-h(2,:)]));
   WAMP=sum((abs(WAMP)>threshold))-(abs(WAMP(1))>threshold);

end
```

In MATLAB:

```
%Willison amplitude (WAMP)
function[WAMP]=WAMP(y,threshold)
h=[y';0 y(1:length(y)-1)'];
   WAMP=(abs([h(1,:)-h(2,:)]));
   WAMP=sum((abs(WAMP)>threshold))-(abs(WAMP(1))>threshold);

end
```


In MATLAB:

```
%Variance
function[VAR]=VAR(y)
    VAR=var(y);

end
```

In MATLAB:

```
%Simple Square Integral (SSI)
 function[SSI]=SSI(y)
   SSI=sum((abs(y)).^2);
end
```

In MATLAB:

```
%slope sign change
function [c]=SSC(g)

x=diff(sign(diff(g)))~=0;
c=sum(x(:) == 1);
end
```

In MATLAB:

```
%Integrated EMG (IEMG)

function[IEMG]=IEMG(y)
   IEMG=sum(abs(y)); %IEMG

end
```