Journal of Science and Technology
ISSN: 2456-5660 Volume 5, Issue 4, July-August 2020

www.jst.org.in *DOI: https://doi.org/10.46243/jst.2020.v5.i4.pp248-260*

Performance Evaluation of EMG Pattern Recognition Techniques While Increasing The Number of Movement Classes

[1]Ms. Priyanka Sharma, [2]Aakash Kr. Singh, [3]Anupriya Mishra, [4]Biplav Singh Kushwaha

[1](Asst.prof, Galgotias College of Engineering And Technology, GreaterNoida, Uttar Pradesh, India)
[2,3,4](Galgotias College of Engineering And Technology, GreaterNoida, Uttar Pradesh, India)
[4]Corresponding Author: biplavsingh723@gmail.com

To Cite this Article
Ms. Priyanka Sharma, Aakash Kr. Singh, Anupriya Mishra and Biplav Singh Kushwaha, "Performance Evaluationof EMG Pattern Recognition Techniques While Increasing The Number of Movement Classes", Journal of Science and Technology, Vol. 05, Issue 04, July-August 2020, pp248-260

Article Info
Received: 10-04-2020 Revised: 10-07-2020 Accepted: 12-07-2020 Published: 14-07-2020

Abstract:*In the past few years of research done in the field of myoelectric control, many researchers have proposed several models implying a combination of different features and classifiers to increase the movement classes, but all that work fails to explain if there is any correlation between multi-class classification and its accuracy. This paper focuses on finding the factors that decide the limit of movement classes that machine learning algorithms can accurately differentiate and to evaluate the performance of pattern classification techniques using the sEMG signal when the number of movement classes is increased while keeping the simplicity of the system. The results were obtained for eight channels sEMG signal using 7 independent time-domain features and four feature set combinations over 4 classifiers (Support Vector Machine(SVM), K-Nearest Neighbour(K-NN), Decision Tree(DT), and Naïve Bayes(NB)). Then the number of classes was increased in the manner of 5, 7, 10, 12, and 15 classes to determine the highest number of movement classes that the sEMG system with above-described features can classify efficiently. And the effect of increasing the number of movement classes on system accuracy was observed. The highest accuracies for all five class progression were obtained for SVM with the MFL feature, and for DT using MAV, it was successfully observed that the NB classifier had minimum performance depletion for the features used in this work*

Keywords*: sEMG, multi-class movement, performance evaluation*

I. Introduction

The myoelectric control mechanism is divided into several steps namely; data acquisition, data pre-processing, feature extraction, classification, post-processing, and control of actuators. Each step has its parameter consideration and utilization. And the accumulation of these steps with their optimal and efficient use leads to better classification and control. Standard EMG specifications were used for data pre-processing and signal conditioning. (Abbaspour, Linden, GHolamhosseini, Naber, & Catalan, 2020)[1] in their work proposed 44 features from the time-domain, frequency-domain, and time-frequency domain. They suggested that these nine time-domain features- MAV, STD, WL, MPV, DAMV, MFL, IAV, DASDV, and PERC showed the best result for these classifiers *(LDA, KNN, SVM, and Decision Tree)* which formed the base for our study. Among pattern recognition techniques, various machine learning classifiers were used thoroughly and lots of research has been done. (Karlik, 2014)[2], (Toledo-Pereze, Resendiz, Loenzo, & Jauregui-Correa, 2019)[3] and (Parajuli, et al., 2019)[4] compared the work of various researchers that used a combination of different TD feature extractions and pattern recognition algorithms and found their accuracies as 100% for 2 class movement using RMS, MAV, NOR, SUM, MAX, MIN, and RAN, 98% for 6 classes of movement using 4th order autoregression(AR) with fuzzy clustering neural network and an accuracy of 97% for 7 classes of movement using 57 channels, 6th order AR, RMS/PCA, ULDA for feature extraction and KNN, LDA for classification and majority vote method for post-processing. However, it is still unclear if there is any correlation between multi-class classification and its accuracy. And if there is, this should be addressed as it defines the limits of prosthetics when using machine learning algorithms and classifiers. The end goal of this work is to determine the effect of the increased number of movement classes over the model's accuracy. The performance evaluation is done in terms of the classifier's accuracy.

This paper is divided into the following sections.

Section 1 introduction which briefs about the aim of this work and little about the methods and techniques used to fulfill that aim.

Section 2 explains in detail the complete pattern recognition-based myoelectric control method.

Section 3 is the experimental methodology that explains how the information provided in section 2 was used to carry out this experiment.

Section 4 summarizes the experiment's results and observations made from it.

Section5 provides the conclusion of the research done.

II. Pattern Recognition-Based Myoelectric Control

Pattern recognition is a technique of predicting the desired movement from a set of trained movements made by classifying the signal into a particular movement based on the extracted features of their signals respectively. Figure 1 below shows the basic concept of the pattern recognition-based control method.

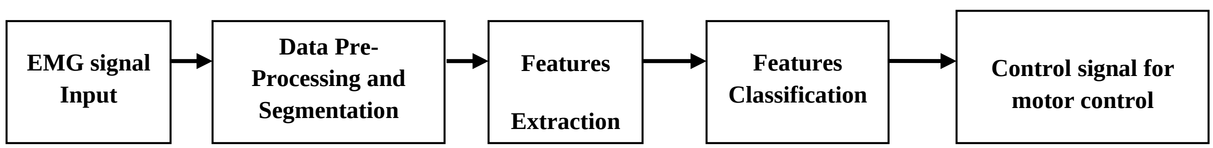


Figure 1: Block Diagram of working of pattern recognition method

Pattern recognition based EMG control (Parajuli, et al., 2019)[4] mainly involves the segmentation of data into a training set and testing data set, extracting features, and classifying them. All of them are explained in detail later sections of this paper. Some of the work done it this field using raw EMG data also require to pre-process the data (Sundarsan & Sekaran, 2012)[5] before segmentation such as filtering, rectification, amplification, base-line drifting, and threshold leveling.

2.1: Data segmentation

EMG data have a randomly varying pattern due to which its segmenting is important for extracting desired features precisely. For EMG data segmentation window approach is used wherein two methods are available (Nazmi, Rahman, Yamamoto, Ahmad, Zamzuri, & Mazlan, 2016) [6] adjacent window approach and overlapping window approach as shown in Figure 2. In the former technique, disjoint predefined length segments are used for feature extraction and classification. Due to the high-speed processors, the processing time is usually less than the segment window duration, which makes the processor idle for a certain amount of time (t) which introduces delay. This drawback is overcome by the overlapping window approach wherein a segment slides over the current segment by maintaining the increment time(inc.) less than the delay time(t) so that the idle time is used for acquiring more data to be processed as a result of which a better classification accuracy is achieved and overall delay time is reduced. Practically a classification error less than ≈ 10% yields highly controllable systems and in a worst-case system becomes completely uncontrollable if classification error exceeds 35%. To minimize the classification error, a segment must have an appropriate length. Ideally, a segment of $t \leq 200$ms is of sufficient size to contain enough information to predict the limb movement because that is the minimum interval between distinct contractions.

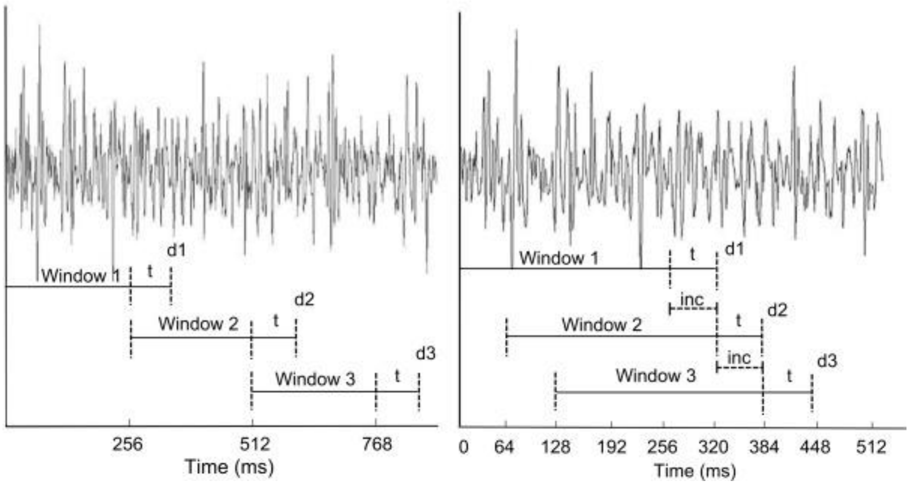


Figure 2: Adjacent window (left) and overlapping window (right)

2.2: Feature extraction

The features of an EMG signal are extracted in three forms (Parajuli, et al., 2019)[4], these are time-domain(TD), frequency domain(FD), and time-frequency domain(TFD). In TD approach, the features are extracted by varying the amplitude of the signal with time while the FD approach uses the power spectrum density of EMG signal and the TFD approach uses the combined features of both TD and FD to extract features. There are many feature extraction methods in each of the three domains (Zhang, et al., 2017)[7], and are used in combination based on the desired features to be extracted. The TD features (Parajuli, et al., 2019)[4] (Meena, 2019)[8] are:

- *The Mean Absolute Value(MAV)* is used for calculating the average of the absolute value of all the time samples and provides information about the muscle contraction levels. It is calculated using the equation (2.1):

$$MAV = \frac{1}{N}\sum_{i=1}^{N}|x_i| \qquad (2.1)$$

- *The Root Mean Square(RMS)* method provide information about the mean power of the EMG signal from each muscle and is given by (2.2):

$$RMS = \sqrt{\frac{1}{N}\sum_{I=1}^{N}x_i^2} \qquad (2.2)$$

- *Variance(VAR)* is the measure of the power density of the signal, given by (2.3):

$$VAR = \frac{1}{N}\sum_{i=1}^{N}(x_I - \overline{x})^2 \qquad (2.3)$$

- *Zero Crossings (ZC)* are the count of the number of times the waveform crosses the zero position i.e. the number of times the waveform goes from positive to the negative side or vice versa. These are used to detect the onset of movement during the procedure of data segmentation, measured by (2.4):

$$ZC = \sum_{i=1}^{N-1}\{f(x_i * x_{i+1})\} \qquad (2.4)$$

$$f(x) = 1\ if x < 0\ and |x_i - x_{i+1}| \geq threshold$$

$$f(x) = 0 \; otherwise$$

- *Slope Sign Change(SSC)* is somewhat similar to zero crossings and provide information about the frequency content of the signal calculated as (2.5):

$$SSC = \sum_{i=2}^{N-1}\left[f\left[(x_i - x_{i-1})(x_i - x_{i+1})\right]\right] \quad (2.5)$$

$$f(x) = 1 \; if \; x \geq threshold$$

$$f(x) = 0 \; otherwise$$

- *Simple Square Integral(SSI)* is simply the measure of the energy of the EMG signal, given by (2.6) :

$$SSI = \sum_{i=1}^{N}(|x_i^2|) \qquad (2.6)$$

- *Waveform Length(WL)* is the increasing length of the waveform over the segment. It is related to the signal amplitude, frequency, and time. It is calculated using (2.7):

$$WL = \sum_{i=1}^{n-1}|x_{i+1} - x_i| \qquad (2.7)$$

- *Auto-Regressive(AR) model* is a method that analyzes the signal by calculating previous samples of the signal. In this model, each sample of the signal represents a linear combination of all its previous samples and is given by the formula (2.8):

$$x(n) = -\sum_{i=1}^{p} a_i x[n-i] + e[n] \qquad (2.8)$$

Where x[n] is the data which is the combination of n data points, p is the AR model, a_i is the AR coefficient and e[n] is white noise independent of the previous sample.

- *Standard Deviation(STD)* represents the deviation of each EMG sample with its mean value and is defined by:

$$STD = \left[\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2\right]^{1/2} \qquad (2.9)$$

- *Number of Peaks(NP)* is the total number of values higher than their RMS value(2.2).
- *Mean of Peak Values(MPV)* is the average of the number of peak values.
- *Maximum Fractal length(MFL)* is used to measure the muscle contraction strength for low- level muscle activation and is given by (2.10):

$$MFL = \log_{10}\left(\sqrt{\sum_{i=1}^{N-1}(x_{i+1} - x_i)^2}\right) \qquad (2.10)$$

- *Integrated Absolute Value(IAV)* is simply the summation of absolute values of the EMG signal when taken over a time window of N samples, given by (2.11):

$$IAV = \sum_{i=1}^{N}|x_i| \qquad (2.11)$$

- *Difference Absolute Standard Deviation Value(DASDV)* is found by doing the standard deviation of values of the difference between the adjacent samples, given by (2.12):

$$DASDV = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N-1}(x_{i+1} - x_i)^2} \qquad (2.12)$$

- *Difference Absolute Mean Value(DAMV)* is given by (2.13):

$$DAMV = \frac{1}{N}\sum_{i=1}^{N-1}|x_{i+1}-x_i| \qquad (2.13)$$

- *Integrated EMG(IEMG)* is used in recognition of signals which do not have a fixed pattern. It is the sum of absolute values of each EMG sample, given by equation (2.14):

$$IEMG = \sum_{k=1}^{N}|x_k| \qquad (2.14)$$

For extracting the FD features the EMG signal from time domain is converted into the frequency domain using the Fourier transform and them applying autocorrelation functions of the EMG signal to determine its power spectral density. The FD features are used in diagnosing muscle fatigue and analyzing motor unit recruitment. (Parajuli, et al., 2019)[4] (Meena, 2019)[8] states the methods for FD features extraction these are power spectral density with *Mean Frequency(MNF)* (2.15) *and Median Frequency(MDF)* (2.16)and power spectral density with *Peak Frequency(PKF)* (2.17) *and Mean Power(MNP)* (2.18) to determine the maximum and average power spectrum respectively using the following formulae:

$$MNF = \frac{\sum_{i=1}^{M} f_i P_i}{\sum_{i=1}^{M} P_i} \qquad (2.15)$$

$$\sum_{i=1}^{MDP} P_i = \sum_{i=MDP}^{M} P_i = \frac{1}{2}\sum_{i=1}^{M} P_i \qquad (2.16)$$

$$PKF = \max{(P_i)} \qquad (2.17)$$

$$MNP = \frac{\sum_{i=1}^{M} P_i}{M} \qquad (2.18)$$

The last feature extraction method is TFD features. This can provide signal information in both time as well as frequency domain. The TFD features are a good choice for processing sEMG signals due to their ability to process non-stationary signal like sEMG signals where the presence of frequency components vary with time. (Burhan, Kasno, & Ghzali, 2016) [9] in their work explained the processing of such non-stationary signals in detail using TFD techniques such as *Short-time Fourier Transform(STFT), wavelet transform(WT), and wavelet packet transform(WPT).*

2.3: Features classification

After the extraction of desired features from an EMG signal, it is ready for classification. Extracted features are the input to the classifier model. These features are used to train the model so that it can predict the user's intended movement. (Karlik, 2014)[2] and (Igual, Pardo, Hahne, & Igual, 2019) [10] in their works provided a review of several complex machine learning algorithms developed by many researchers, these algorithms can be used to train the classifier model to predict the movement with a higher level of precision. Some of these algorithms were studied and explained in this section.

*Artificial Neural Networks(*ANN) classifier (Mane, Kambli, Kazi, & Singh, 2015)[11], are computational algorithms that intend to copy the behavior of biological systems composed of neurons.ANN includes a large number of interconnected processing units called neurons that work together to process information and generate meaningful results from it.

The *Fuzzy Logic Classifier* is a classifier system that uses fuzzy logic to categorize variables into different sets based on defined rules. Fuzzy logic is an approach to computing based on degrees of truth rather than traditional Boolean logic. According to studies done by (K., Sivanandan, & Mohandas, 2012)[12], fuzzy logic allows partial membership or degree of membership that involves all intermediate possibilities between digital values of false or true(0 or 1) such that a variable belongs to a class if its value lies within the range of the class.

The *Hybrid Algorithms* are algorithms developed by integration of neural network and fuzzy logic to develop an advanced and more effective neuro-fuzzy system which has the combined benefits of both the classifier stated above. (Subasi, 2012)[13] in his work explained and compared classifiers developed using this approach, these classifiers are adaptive neuro-fuzzy interface system(ANFIS), dynamic fuzzy neural network(DFNN), and multilayer perceptron neural network(MLPNN).

The *Linear Discriminant Analysis(LDA)* classifier is a dimensionality reduction technique in which, the number of dimensions(variables) is reduced while maintaining the maximum information possible. (Phinyomark, HU, Phukpattaranont, & Limsakul, 2012)[14] Suppose if the relationship between two variables is plotted, then LDA uses the information from both features to create a new axis and project the data on it in such a way as to maximize the distance between the mean and minimize the variance of the two classes.

The *Naïve Bayes Classifier* is a probability-based classifier based on Bayes theorem (Joyce, 2019)[15]. (Dev & Singh, 2016)[16] Its working is divided into two steps first is the training step, in which the parameters of a probability distribution are estimated for the given training samples, assuming that the features are conditionally independent. Second, the prediction step in which the posterior probability for any unseen test sample is computed. The test sample is then classified accordingly to the largest posterior probability.

The *K-Nearest Neighbour(K-NN)* classifier is a nonparametric pattern classification method used where there is little or no pre-requisite about the distribution of the data. It is an instant-based learning algorithm classifies the variable based on the closest feature space in the training set. These training sets are mapped into the multi-dimensional feature space, which is partitioned into regions based on the categories of the training set. A variable in the feature space is assigned to a particular class if it is the nearest class among the k nearest training data (Karlik, 2014) (Wettschereck, Aha, & Mohri, 1997)[2, 17]. The working of KNN is shown in Figure 3 below.

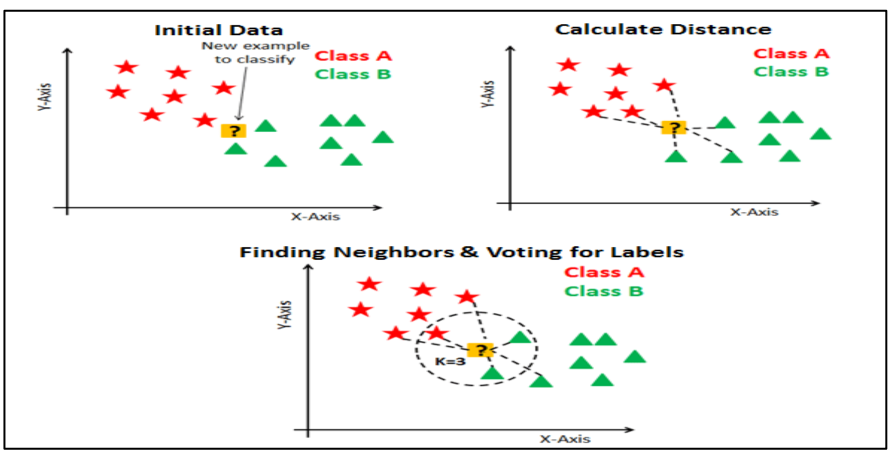


Figure 3: Working of KNN algorithm

The *Support Vector Machine(SVM)* classifier (Ray, 2017)[18] is a supervised machine learning algorithm based classifier which can be used for both classification and regression. In this data-items are plotted as points in n-dimensional space (n is the number of extracted features), then classification is performed by finding the hyper-plane that differentiates the two classes very well with maximum segregation from either nearest data points as shown by Figure 4 below.

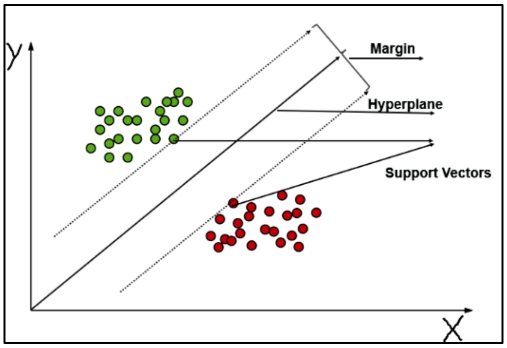


Figure 4: Segregation of data points by hyper-plane

In some cases like in Figure 5 where data points are inseparable and lie in non-linear planes makes hyperplane in-efficient. In those cases, the SVM kernel **(Ray, 2017)**, **(Waseem, 2019)**[18, 19] technique is used. SVM kernel is a function that takes low dimensional input space and transforms it into a higher dimensional space to convert non-separable variables into separable data points. For this, an additional dimension 'z' is added using $z = x^2 + y^2$ **(Ray, 2017)**[18] and then the data points are mapped over that dimension so that they can be easily segregated. Now when the hyper-plane is seen in original input space, it looks like a circle separating data points of different types efficiently.

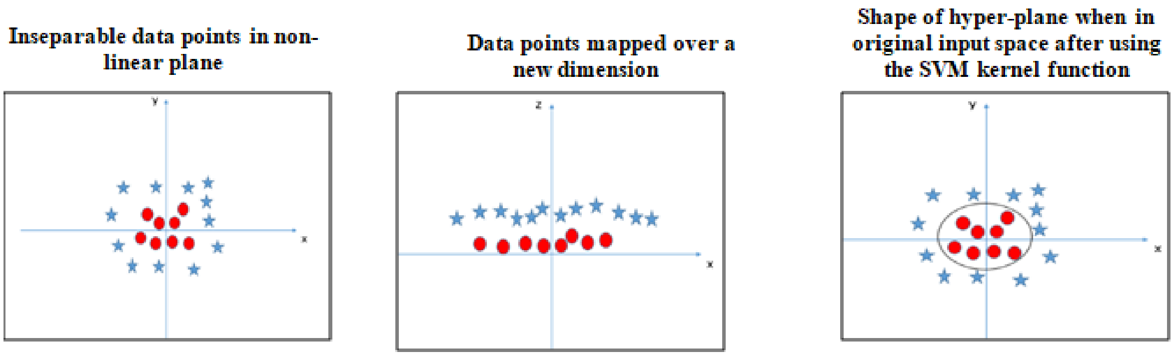


Figure 5: Operation of SVM Kernel for inseparable data in non-linear planes

The *Decision tree* algorithm falls under the category of supervised learning. It is a flowchart structure in which each internal node represents a test on a feature, each leaf node represents a class label and branches represent conjunctions of features that lead to those class labels. The path from the root to leaf represents classification rules constructed via an algorithmic approach that identifies the ways to split a data set based on different conditions. Figure 6 shows the basic flow chart of a decision tree.

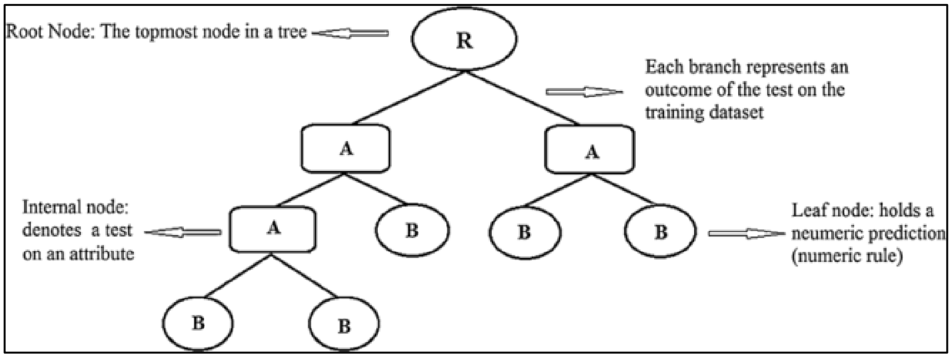


Figure 6: Decision tree flow chart

III. Experimental Methodology

3.1 Data Acquisition

For the experiment, a dataset from (Khushaba, Kodagoda, Liu, & Dissanayake, 2012)[20] was used as the main source. This was a 15 class movement dataset for which eight subjects, six males and two females between the age of 20-35 were recruited. These 15 classes involve the individual movement of thumb(T), index(I), middle(M), ring(R), little(L) and the combined thumb-index(TI), thumb-middle(TM), thumb-ring(TR), thumb-little(TL),index-middle(IM), middle-ring(MR), ring-little(RL), index-middle-ring(IMR), middle-ring-little(MRL), and finally hand close(HC). For the collection of data, the subjects were supposed to flex and hold a movement for 20 seconds for three trials each movement.

For our study, we found that the 20 seconds data was exceeding our requirement so a data of 5 seconds was cropped out of this data set.

3.2 Data Pre-processing

After the acquisition of desired data, it was then pre-processed using MATLAB R2016a. The data was first pre-amplified by designing an amplifier with a fixed gain of 1000 to amplify micro-volts(μV) EMG signal into milli-volts(mV) amplitude signal. After pre-amplification, the signal was filtered using a 4^{th} order lowpass filter with 450Hz cutoff frequency, having no passband ripple, and a stopband ripple of 80dB. The acquired signal was then filtered by a high pass filter of 20Hz cutoff frequency to eliminate the external noise and motion artifacts. In the place of low pass and highpass, a bandpass filter operating between 20-450 Hz frequency can also be used. The filtration process was completed by at last passing the signal through a notch filter to eliminate the 50 Hz frequency for removing the noise that may arise due to 50 HZ power line interference.

After filtering the signal, full-wave rectification was done to remove the negative components from the signal. For this purpose, the signal was simply squared to convert all the negative amplitudes of the signal into positive amplitudes. The signal was then amplified one more time to boost the amplitudes between the range of 0 to 4 volts.

The final step of data pre-processing involves smoothening of data, for which *Savitzky-Golay filtering* tool of MATLAB was used which is an FIR smoothening filter that can detect envelopes of the signal during contraction phase from a noisy signal whose frequency span is high by efficiently rejecting the noise without losing the signal's high-frequency content along with noise.

3.3 Data Segmentation

For the segmentation of data, the adjacent window approach was used with a sampling window of length 200ms.

3.4 Feature Extraction

Out of the above-stated features, 7 TD features namely MAV, WL, MPV, DAMV, MFL, IEMG, and DASDV were used independently, and four feature set combinations i.e. MAV+WL, MAV+MFL, MAV+DASDV, MAV+MFL+WL were used.

3.5 Classification

For the classification of extracted features, the following classifier algorithms were used:- SVM, KNN, DT, and NB. For experimenting, out of 15 classes, initially, the data of the first five movements classes was used, features were extracted from each of them and then all four classifier models were tested over it to compare the accuracy of each of the classifiers. Then the next two classes were also added and the whole process was repeated for these seven classes and the change in performance of classifiers was observed. Gradually the movement classes were increased from 7 to 10, 12, and finally, 15, repeating the whole procedure every time, and the change in classifier performance was observed with an increment of the number of movement classes.

IV. Result and Observation

After the experiment was performed, result evaluation was done by computing the accuracies of all the classifiers individually using the formula given by equation (4.1):

$$Acc_i = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \qquad (4.1)$$

Where TP, TN, FP, FN are true positive, true negative, false positive, and false negative predictions respectively and i is the class index.

The performance comparison of all four classifiers for the aforementioned features and feature combinations is shown by the plots in Figure 7, Figure 8, Figure 9, Figure 10, and Table 1below.

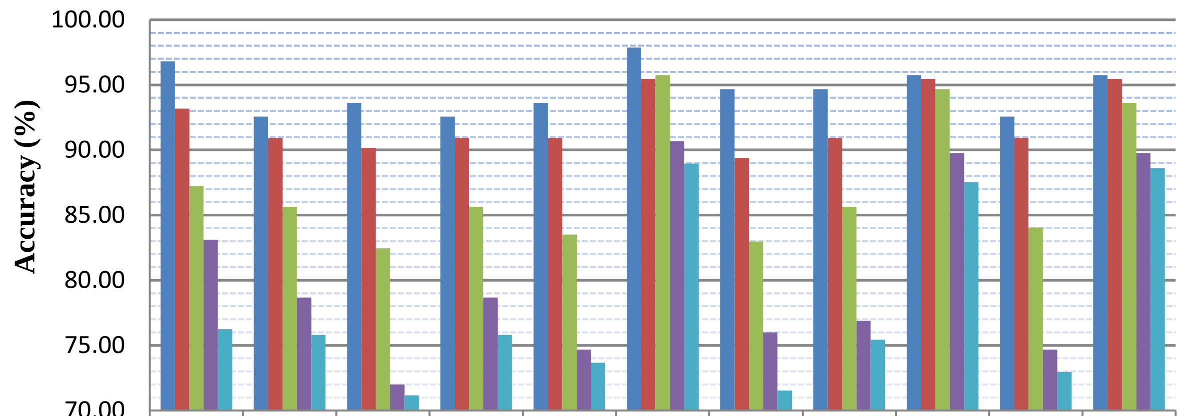
From the table, it was observed that for 5 movement classes best accuracy of 98.94% was achieved using DT classifier with MAV, for 7 classes 97.73% was the highest accuracy achieved by using DT with MAV feature, for 10 classes SVM gave the best result of 95.74% with MFL feature, for 12 class again SVM with MFL feature gave the best result of 90.67% accuracy, and for 15 movement classes, the highest accuracy of 88.97% was achieved again by SVM with MFL feature.

Another observation was made that shows that the performance of classifiers highly depends on the features that are used with them, for example, the SVM overall gave better results than other classifiers when MFL feature was employed even with the combinations that involved MFL in them like MAV+MFL, and MAV+MAF+WL while accuracy was highly reduced when these features were applied to the DT classifier. Similarly, for the MAV feature, the DT classifier outperformed SVM, K-NN, and NB in all the movement classes. This case remained the same even with feature set combination wherein for some combinations like MAV+WL and MAV+DASDV the DT classifier performed better than others. Overall it was observed that SVM gives the highest accuracy for all the classes when the MFL feature was used with it but fails to perform well with other features. The K-NN and the NB classifiers have almost similar performance with NB performing slightly better than K-NN for a higher number of movement classes when used with DAMV, IEMG, and MAV+DASDV features. The DT classifier was found to be the most versatile classifier as it not only gave the highest accuracies when used with the MAV feature but also performed fairly good enough with all other features, MFL, MAV+MFL, and MAV+MFL+WL being the exceptional cases.

Table 1: Performance of Different Classifiers over various features while increasing the number of movement classes

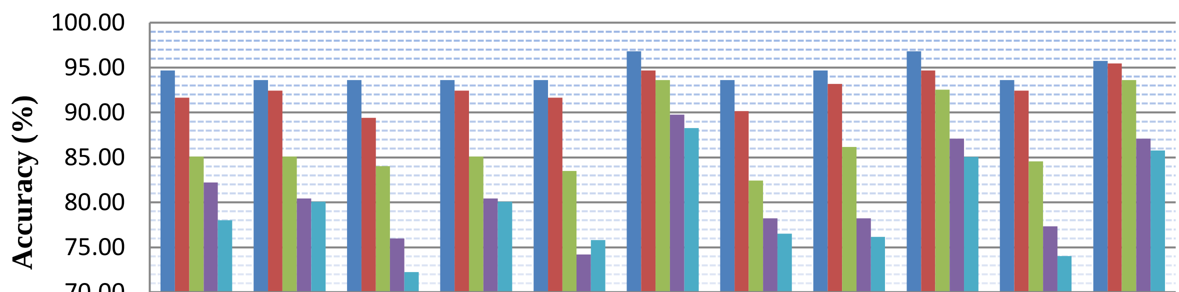
Features	**Classifiers Used**																			
	Support Vector Machine					**K - Nearest Neighbour**					**Decision Tree**					**Naïve Bayes**				
	Number of Movement Classes																			
	5	**7**	**10**	**12**	**15**	**5**	**7**	**10**	**12**	**15**	**5**	**7**	**10**	**12**	**15**	**5**	**7**	**10**	**12**	**15**
MAV	96.81	93.18	87.23	83.11	76.24	94.68	91.67	85.11	82.22	78.01	98.94	97.73	86.17	88.89	81.91	94.68	94.70	84.57	85.78	81.21
WL	92.55	90.91	85.64	78.67	75.80	93.62	92.42	85.11	80.44	80.07	93.62	95.45	89.36	84.89	82.56	94.68	93.18	87.23	86.22	82.21
MPV	93.62	90.15	82.45	72.00	71.17	93.62	89.39	84.04	76.00	72.24	94.68	92.42	87.23	85.33	80.07	93.62	93.18	86.70	79.11	76.51
DAMV	92.55	90.91	85.64	78.67	75.80	93.62	92.42	85.11	80.44	80.07	93.62	95.45	89.36	84.89	82.56	94.68	93.18	87.23	86.22	82.21
IEMG	93.62	90.91	83.51	74.67	73.67	93.62	91.67	83.51	74.22	75.80	95.74	93.94	84.57	81.78	75.80	93.62	92.42	85.64	82.22	79.00
MFL	97.87	95.45	95.74	90.67	88.97	96.81	94.70	93.62	89.78	88.26	92.55	92.42	87.23	85.78	77.22	95.74	94.70	92.02	89.33	86.83
DASDV	94.68	89.39	82.98	76.00	71.53	93.62	90.15	82.45	78.22	76.51	92.55	93.18	86.70	85.33	77.58	94.68	93.94	85.64	82.22	80.43
MAV+WL	94.68	90.91	85.64	76.89	75.44	94.68	93.18	86.17	78.22	76.16	93.62	94.70	85.64	83.11	79.72	93.62	93.18	87.23	84.00	81.14
MAV+MFL	95.74	95.45	94.68	89.78	87.54	96.81	94.70	92.55	87.11	85.05	94.68	93.18	87.23	82.67	80.07	95.74	94.70	93.62	87.11	86.12
MAV+DASDV	92.55	90.91	84.04	74.67	72.95	93.62	92.42	84.57	77.33	74.02	94.68	93.18	87.23	82.67	80.07	94.68	93.18	86.70	82.67	80.43
MAV+MFL+WL	95.74	95.45	93.62	89.78	88.61	95.74	95.45	93.62	87.11	85.77	94.68	93.18	82.45	83.11	80.78	95.74	95.45	92.55	86.22	86.12

For observing the effect of increasing movement classes on the performance of classifiers, degradation of accuracy was calculated by finding the difference between the accuracy for 5 movement classes from that of the 15 movement classes for all the aforementioned features individually and then the average accuracy degradation rate was calculated from it for all the classifiers. From the calculations, it was found that although all the classifiers had similar accuracies for a lower number of movement classes, but on increasing the number of movement classes, SVM classifier suffered the maximum performance degradation of around 16.58% difference in the initial and the final performance, K-NN classifier suffered 15.46% performance degradation in its accuracies over increased classes, DT classifier with 14.70% degradation, and the NB classifier which suffered minimum accuracy depletion of 12.66% with the increase in movement classes



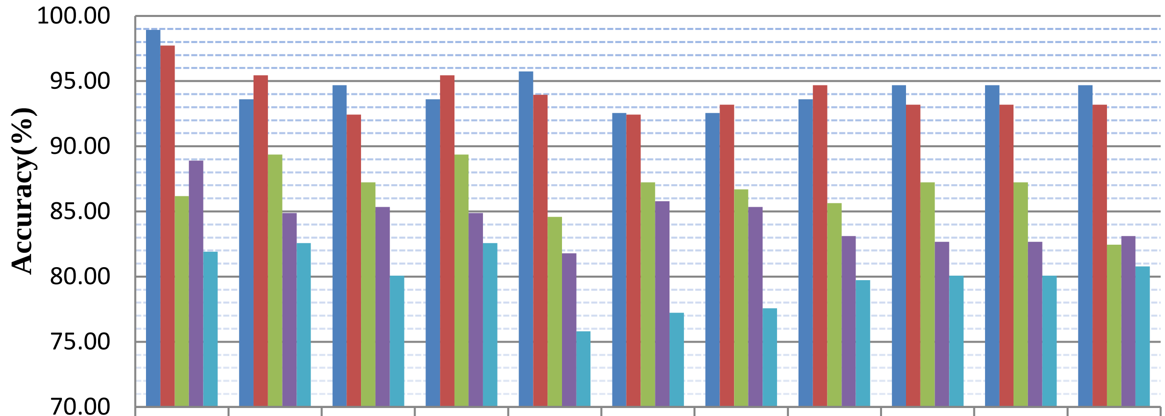
	MAV	WL	MPV	DAMV	IEMG	MFL	DASDV	MAV+WL	MAV+MFL	MAV+DASDV	MAV+MFL+WL
5 Classes	96.81	92.55	93.62	92.55	93.62	97.87	94.68	94.68	95.74	92.55	95.74
7 Classes	93.18	90.91	90.15	90.91	90.91	95.45	89.39	90.91	95.45	90.91	95.45
10 Classes	87.23	85.64	82.45	85.64	83.51	95.74	82.98	85.64	94.68	84.04	93.62
12 Classes	83.11	78.67	72.00	78.67	74.67	90.67	76.00	76.89	89.78	74.67	89.78
15 Classes	76.24	75.80	71.17	75.80	73.67	88.97	71.53	75.44	87.54	72.95	88.61

Figure 7: Feature's performance comparison while increasing the number of movement classes for SVM using 8 channel sEMG data



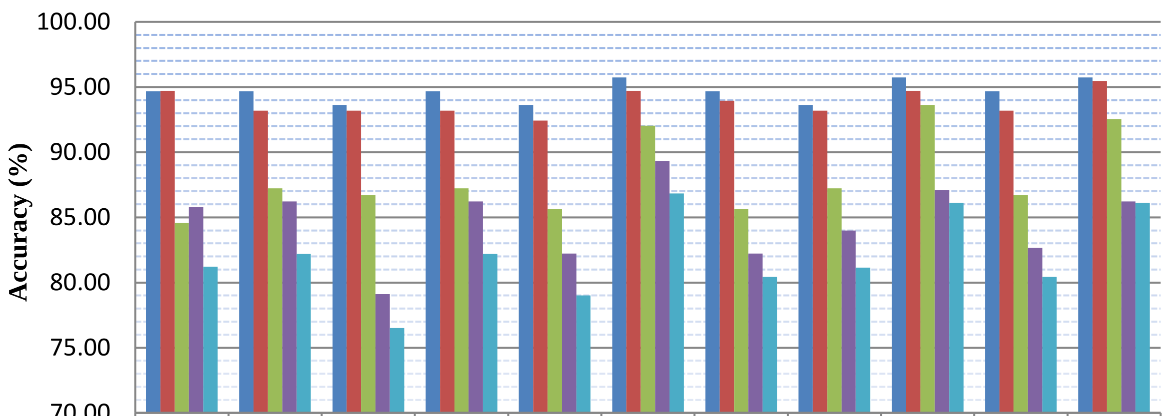
	MAV	WL	MPV	DAMV	IEMG	MFL	DASDV	MAV+WL	MAV+MFL	MAV+DASDV	MAV+MFL+WL
5 Classes	94.68	93.62	93.62	93.62	93.62	96.81	93.62	94.68	96.81	93.62	95.74
7 Classes	91.67	92.42	89.39	92.42	91.67	94.70	90.15	93.18	94.70	92.42	95.45
10 Classes	85.11	85.11	84.04	85.11	83.51	93.62	82.45	86.17	92.55	84.57	93.62
12 Classes	82.22	80.44	76.00	80.44	74.22	89.78	78.22	78.22	87.11	77.33	87.11
15 Classes	78.01	80.07	72.24	80.07	75.80	88.26	76.51	76.16	85.05	74.02	85.77

Figure 8: Feature's performance comparison while increasing the number of movement classes for K-NN using 8 channel sEMG data



	MAV	WL	MPV	DAMV	IEMG	MFL	DASDV	MAV+WL	MAV+MFL	MAV+DASDV	MAV+MFL+WL
5 Classes	98.94	93.62	94.68	93.62	95.74	92.55	92.55	93.62	94.68	94.68	94.68
7 Classes	97.73	95.45	92.42	95.45	93.94	92.42	93.18	94.70	93.18	93.18	93.18
10 Classes	86.17	89.36	87.23	89.36	84.57	87.23	86.70	85.64	87.23	87.23	82.45
12 Classes	88.89	84.89	85.33	84.89	81.78	85.78	85.33	83.11	82.67	82.67	83.11
15 Classes	81.91	82.56	80.07	82.56	75.80	77.22	77.58	79.72	80.07	80.07	80.78

Figure 9: Feature's performance comparison while increasing the number of movement classes for DT using 8 channel sEMG data



	MAV	WL	MPV	DAMV	IEMG	MFL	DASDV	MAV+WL	MAV+MFL	MAV+DASDV	MAV+MFL+WL
5 Classes	94.68	94.68	93.62	94.68	93.62	95.74	94.68	93.62	95.74	94.68	95.74
7 Classes	94.70	93.18	93.18	93.18	92.42	94.70	93.94	93.18	94.70	93.18	95.45
10 Classes	84.57	87.23	86.70	87.23	85.64	92.02	85.64	87.23	93.62	86.70	92.55
12 Classes	85.78	86.22	79.11	86.22	82.22	89.33	82.22	84.00	87.11	82.67	86.22
15 Classes	81.21	82.21	76.51	82.21	79.00	86.83	80.43	81.14	86.12	80.43	86.12

Figure 10: Feature's performance comparison while increasing the number of movement classes for NB using 8 channel sEMG data

V. Conclusion

After observing the experiment it was concluded that the performance of various classifiers along with the features used differs for the number of movement classes. A classifier showing good results for less number of classes over a given feature set may not work the same for an increased number of classes with the same features. Thus from this work, it can be inferred that increment of classes is a major factor to decide which features to use and for which classifiers. The experiment was performed on some commonly used classifier and observed results were shown, in future the researchers may use the proposed model. However, it is recommended to the researchers that they experiment with the classifiers and features to determine the best combination that meets their requirements.

References:

[1]. Abbaspour, S., Linden, M., GHolamhosseini, H., Naber, A., & Catalan, M. O. (2020). Evaluation of surface EMG-based recognition algorithms for decoding hand movements. *Medical & Biological Engineering & Computing* , 83-100.
[2]. Karlik, B. (2014). Machine learning Algorithms for Characterization of EMG Signal. *International Journal of Information and Electronics Engineering* , 189-191.

[3]. Toledo-Pereze, D. C., Resendiz, R. J., Loenzo, R. A., & Jauregui-Correa, J. C. (2019). Support Vector Machine-Based EMG Signal Classification Techniques: A Review. *MDPI* , 1-28.

[4]. Parajuli, N., Sreenivasan, N., Bifulco, P., Cesarelli, M., Savino, S., Niola, V., et al. (2019). Real-Time EMG Based Pattern Recognition Control for Hand Prosthesis: A Review on Exiting Challenges and Future Implementation. *MDPI* , 8-11.

[5]. Sundarsan, S., & Sekaran, D. E. (2012). Design and Development of EMG Controlled Prosthetics Limb. *Elevier Ltd.* , 3547-3549.
[6]. Nazmi, N., Rahman, M. A., Yamamoto, S. I., Ahmad, S. A., Zamzuri, H., & Mazlan, S. A. (2016). A Review of Classification Techniques of EMG Signal during Isotonic and Isometric Contractions. *MDPI* , 8.

[7]. Zhang, Y., Li, P., Zhu, X., SU, S. W., Guo, Q., XU, P., et al. (2017). *Extracting time-frequency feature of single-channel vastut medialis EMG signals for knee exercise pattern recognition.* Chengdu: PLOS ONE.
[8]. Meena, C. (2019). *EMG Feature Extraction and Classification Implementation on FPGA.* Jaipur: MNIT.
[9]. Burhan, N., Kasno, M., & Ghzali, R. (2016). Feature extraction of surface electromyography (sEMG) and signal processing technique in wavelet transform: a review. *IEEE International Conference on Automatic Control and Intelligent Systems* , 142-144.
[10]. Igual, C., Pardo, L. A., Hahne, J. M., & Igual, J. (2019). Myoelectric Control of Upper Limb Prosthesis. *MDPI* , 6-8.

[11]. Mane, S. M., Kambli, R. A., Kazi, P. F., & Singh, P. N. (2015). Hand Motion Recognition From Single Channel Surface EMG Using Wavelet & Artificial Neural Network. *4th International Conference on Advances in Computing, Communication and Control* , 61.

[12]. K., S. G., Sivanandan, K. S., & Mohandas, K. P. (2012). Fuzzy Logic and Probabilistic Neural Network for EMG Classification - A Comparative Study. *International Journal of Engineering & Technology* , 3.
[13]. Subasi, A. (2012). Classification of EMG signal using combined features and soft computing techniques. *ELSEVIER* , 2188-2198.
[14]. Phinyomark, A., HU, H., Phukpattaranont, P., & Limsakul, C. (2012). Application of Linear Discriminant Analysis in Dimensionality Reduction for Hand Motion Classification. *MEASUREMENT SCIENCE REVIEW* , 83-84.
[15]. Joyce, J. (2019, March 21). *The Standford Encyclopedia of Philosophy.* Retrieved May 24, 2020, from plato.standford.edu: <https://plato.stanford.edu/archives/spr2019/entries/bayes-theorem/>

[16]. Dev, R., & Singh, A. K. (2016). Performance analysis of classifiers for EMG signal of different hand movements. *International journal of biomedical engineering and technology* , 240-241.
[17]. Wettschereck, D., Aha, D. W., & Mohri, T. (1997). A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazzy Algorithms. *Artificial Intelligence Review* , 275-277.
[18]. Ray, S. (2017, September 13). *Understanding Support Vector Machine(SVM) algorithm from examples (along with code).* Retrieved May 25, 2020, from analyticsvidhya: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/
[19]. Waseem, M. (2019, November 27). *A Quick Guide To Learn Support Vector Machine.* Retrieved May 25, 2020, from edureka: https://www.edureka.co/blog/support-vector-machine-in-python/
[20]. Khushaba, R. N., Kodagoda, S., Liu, D., & Dissanayake, G. (2012). Electromyogram(EMG) Feature Reduction Using Mutual Components Analysis for Multifunctional Prosthetic Finger Control . *Int. Conf. on Control, Automation, Robotics & Vision(ICARCV)* , 1534-1539.