2023 International Conference on Computer, Electronics & Electrical Engineering & their Applications (IC2E3) | 979-8-3503-3800-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/IC2E357697.2023.10262690

sEMG-Based Classification of Finger Movement with Machine Learning

Naveen Gehlot
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India
gehlotnaveen27@gmail.com

Ashutosh Jena
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India
jenaashutosh498@gmail.com

Ankit Vijayvargiya
Department of Electrical Engineering
Swami Keshvanand Institute of Technology
Management & Gramothan, Jaipur, India
ankitvijayvargiya29@gmail.com

Rajesh Kumar
Department of Electrical Engineering
Malaviya National Institute of Technology
Jaipur, India
rkumar.ee@mnit.ac.in

***Abstract*—Classification of finger movements is a challenging task due to the complications introduced by noise artifacts on low amplitude biopotential signals. Electromyography enables the visualization and analysis of changes in biopotential signal due to different muscular activities, which further allows the classification of muscular signals. In this article, surface electromyography (sEMG) based signal has been collected from two forearm muscles corresponding to the dominant hand, using the BIOPAC acquisition system. The raw signal collected, has been pre-processed using static filtering techniques and converted into seventeen time domain and frequency domain based features. Conversion of filtered signal into features is done using overlapping windowing technique. The thirty four extracted features corresponding to two muscles are used as input in five machine learning (ML) classifiers and a comparative analysis has been presented among those classifiers using performance measures such as Accuracy, Precision, Recall, and F1-score.**

***Index Terms*—Electromyography Signal, Machine Learning, Feature Extraction, Activity Classification.**

I. Introduction

According to a survey conducted by WHO, 15% of the population throughout the world suffers from physical impairments. Such people face difficulties while doing daily life chores. And they struggle to get similar privileges in medicine, employment, and education [1]. Amputation or wrist dis-articulation is a form of physical impairment among people [2].

In modern life, almost all activities ranging from operating a computer gadget to washing dishes, requires dexterous movements. This implies the movement of multiple fingers in a coordinated way to carry out a specific task. So, the final goal is to apply the knowledge of such movements into the building of prosthetics [3]. Prosthetics are intelligent robotic technology designed to assist in rehabilitation or interface human to machine. Such sophisticated equipment should provide significant comfort to amputees. Different biopotential signals, such as EEG, ECG, or EMG, are used to stimulate the prosthetics. However, the most common biopotential signal being used by researchers in forearm assisting devices is EMG signal [4].

EMG is the process of collecting and visualizing signals generated by the muscular part of the body. Signals can be collected using needles (iEMG) or surface electrodes (sEMG). In iEMG invasive thin wire is injected into the muscle. While, sEMG signal is obtained from the surface of the skin which avoids painful experiences by the subject [5]. Conventionally, Ag/AgCl based electrodes are preferred mostly with a conductive gel to enhance conductivity on the skin surface [6].

Jiang et al. [7] employed four channels to acquire and extract a time frequency domain feature to classify six distinct finger motions in their investigation of dexterous finger movement. Tenore et al. [8] classified 12 finger movement activity using 32 channels. The fractal dimension characteristic was employed by Naik et al. [9] to distinguish four sets of finger movements. Kanitz et al. [10] used a genetic algorithmic optimizer and an SVM classifier to classify 12 distinct finger actions based on time domain characteristics.

According to study [11], the number of EMG channels employed for finger movement should be more than or equivalent to the number of fingers whose classification needs to be done for better accuracy in decision. However, more electrodes cover a larger region on the forearm, which is not ideal for amputees. Therefore, the challenge is to achieve higher classification accuracy with optimal number of channels. By a decrease in the channel, the cost of the hardware is reduced, as is the processing time.

Activity classification can be done using primitive thresholding methods. While it's fast and reliable, it can only classify two categories at best. Ulker et al. [12] used fuzzy control based decision maker to classify activity. While these methods are reliable and easy they can't be used when the number of activities increases or while dealing with signal collected from amputees. In such cases the pattern of signal is helpful for classification. ML provides a very efficient way

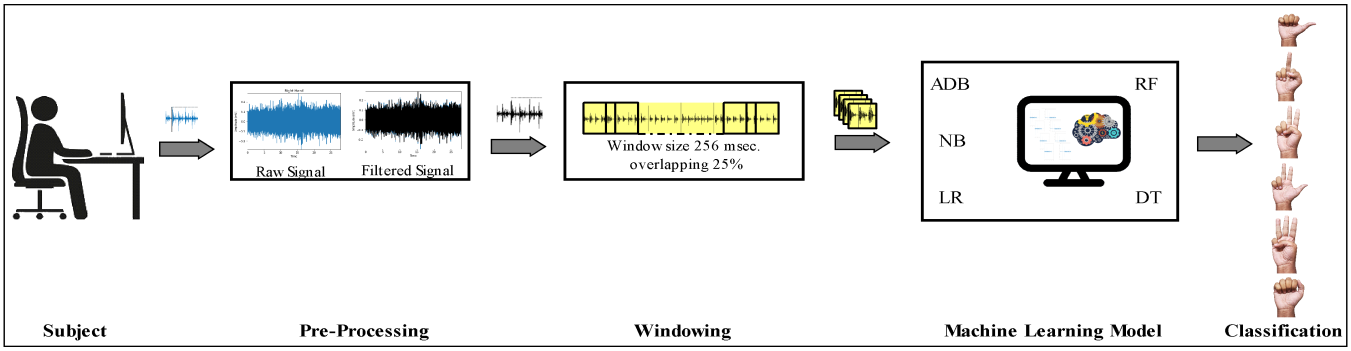


Fig. 1: Flow diagram of signal classification

to identify and classify these patterns [13].

In this study five Machine Learning (ML) techniques are employed for finger activity classification: Decision Tree (DT), AdaBoost (ADB), Random Forest (RF), Logistic Regression (LR), and Naive Bayes (NB). The dataset description is provided in section II. The final results and discussion has been presented in section III, and future work and conclusions are described in section IV.

II. Dataset

In this study, five classifiers are compared with respect to their classification accuracy obtained after training them on a single subject based dataset. In the research mainly six activities of fingers have been performed for the analysis ML classifiers. Classification is done on the extracted feature. There are two forms of feature extraction from the obtained data: time domain and frequency domain. In these domains, feature extraction is only possible when the input is a recurring time series signal. In the entire sEMG signal feature extractions have been done with a window size of 256 msec having a 25% overlapping. Then the recovered feature is utilized as input to various ML classifiers [14].

A. Data Acquisition

Signal is collected from the dominant hand (Right) of a healthy subject. The subject was instructed to recline in his chair and lay his arm on the armrests. The subject complies with the following protocols:

1) Pre-task phase: In this phase subject is briefly instructed about the activities to be performed.
2) Task Phase: In this phase, the subject performed one activity after the other with an interval of 2 minutes to reduce the impact on the muscles.
3) Post-task phase: In this phase subject is advised to perform muscle relaxation activities.

During the task phase, the participant is instructed to meditate and relax.

Signal is acquired using the BIOPAC MP150 system (BIOPAC Systems, Inc.). Hair is removed from the forearm and skin is wiped with alcoholic sanitizer before collecting data. Conductive gel is rubbed onto a clean skin, Ag/AgCl sEMG electrodes are placed. Two channels link the MP 150 to the six sEMG electrodes on the forearm. The extensor digitorum and flexor pollicis longus are the two primary finger activity muscles. The raw sEMG signal data is acquired at a sampling rate of 2000 Hz, amplified by a BIOPAC TEL 100MC with a gain of 1000, and then fed through a 500 Hz low pass filter. Signal is collected in two intervals of 20 seconds each for each finger activity.

During the collection of the sEMG signal, the following six finger movements were carried out: thumb extension (TE), middle extension (ME), fore + middle extension (FME), fore + middle + thumb extension (FMTE), fore + middle + ring extension (FMRE), and hand closure (HC). In Fig. 1 represents the flow from signal collection to signal classify the finger activity of a subject. On the preprocessed data implementation of the ML classifier to classify the activity is described in the figure.

B. Data Preprocessing and Feature Extraction

The collected sEMG signal is filtered using a butterworth notch band stop filter of the fourth order, with a low and high filter cut off frequencies of 49 and 51 Hz, respectively. Then intervals of 256 msec and 25% overlapping length are created and converted to small segments of signal called signal windows. Then feature extraction technique is applied onto these signal windows to necessarily reduce the length of signal to be fed as input to the classifier, in order to reduce the calculation cost and training time of classifiers. Seventeen time and frequency domain based features are calculated, from each signal window, whose mathematical expression are shown in TABLE I.

Here x_m is the m^{th} input sample of sEMG signal, N is the total number of samples, P_j is the power at j^{th} frequency and M is the length of power spectrum density [15]–[17].

C. Machine Learning Models

The following sub section provides a brief description of the ML models employed in the activity classification of sEMG signal.

1) Decision Tree: It is a knowledge based classification algorithm which employs an if-then-rule based classification. In a DT, nodes represent attributes and branches are split

TABLE I: Time-Frequency domain features

SN	Feature Name	Mathematical expression	SN	Feature Name	Mathematical expression
Time Domain Features					
1	Mean Absolute Value (MAV)	$\frac{1}{N}\sum_{m=1}^{k}\lvert x_m\rvert$	2	Root Mean Square (RMS)	$\sqrt{\frac{1}{N}\sum_{m=1}^{k}\lvert x_m\rvert^2}$
3	Variance (VAR)	$\frac{1}{N-1}\sum_{i=1}^{m}x_m^2$	4	Average Amplitude Change (AAC)	$\frac{1}{N}\sum_{m=1}^{N-1}\lvert x_{m+1}-x_m\rvert$
5	Difference Absolute Stabdard Deviation Value (DASDV)	$\sqrt{\frac{1}{N-1}\sum_{m=1}^{N-1}(x_{m+1}-x_m)^2}$	6	Zero Crossing (ZC)	$\sum_{m=1}^{N-1}f(x_m)$ $where\ f(x_m)=\begin{cases}1 & \text{if } (x_m>0\ and\ x_{m+1}<0)\\ & or\ (x_m<0\ and\ x_{m+1}>0)\\ 0, & \text{otherwise}\end{cases}$
7	Waveform Length (WL)	$\frac{1}{N-1}\sum_{m=1}^{N}\lvert x_m\rvert^2$	8	Willson Amplitude (WAMP)	$\sum_{m=1}^{N-1}f(\lvert x_{m+1}-x_m\rvert)$ $where\ f(x_m)=\begin{cases}1 & \text{if } x\geq Threshold\\ 0, & \text{otherwise}\end{cases}$
9	Integrated Electromyogram (iEMG)	$\sum_{m=1}^{N}\lvert x_m\rvert$	10	Myopulse (MYOP)	$\frac{1}{N}\sum_{m=1}^{N}f(x_m)$ $where\ f(x_m)=\begin{cases}1 & \text{if } x\geq Threshold\\ 0, & \text{otherwise}\end{cases}$
11	Log Detector (Log)	$e^{\frac{1}{N}\sum_{m=1}^{N}log(\lvert x_m\rvert)}$			
Frequency Domain Features					
12	Total Power (TP)	$\sum_{j=1}^{M}P_j$	13	Peak Frequency (PKF)	$\frac{1}{2}\sum_{i=1}^{M}P_j$
14	Frequency Ratio (FR)	$\frac{\sum_{j-LLC}^{ULC}P_j}{\sum_{LHC}^{UHC}P_j}$	15	Mean Frequency (MNF)	$\frac{\sum_{j=1}^{M}f_jP_j}{\sum_{j=1}^{M}P_j}$
16	Median Frequency (MDF)	$\frac{1}{2}\sum_{i=1}^{M}P_j$	17	Mean Power (MNP)	$\sum_{i=1}^{M}\frac{P_j}{M}$

according to the values of the attribute represented by a particular parent node.

The first stage in this process is finding out the characteristics of the input data. In the second stage parameters like ginni index or information gain are calculated, which guides the splitting criteria of this classifier [18]. Information gain is weighted entropy due to different values of features. Entropy associated with a particular feature is calculated with the help of equation 1.

$$Entropy=-T_plog_2T_p-F_plog_2F_p \tag{1}$$

Ginni index is also used many-a-times, to decide value for an attribute to make the split. And mathematically it is calculated using following equation 2.

$$Ginni\ index=1-(T_p^2+F_p^2) \tag{2}$$

Here, T_p is probability of a true outcome. F_p is the probability of a false outcome.

2) Random Forest: RF combines outputs of many decision tree classifiers to finally become a strong classifier. Here, a subset of features is chosen randomly by a decision tree, to lower the correlation with the other decision trees. Then all the DTs are trained separately. Predictions from various trained tree classifiers are passed through a voting system. The majority vote decides the final output of a set of input feature [19].

These majority votes of output give high accuracy to handle outliers and noise in the data. It uses the same parameters as that in DT to choose the most selective feature for branching. Employing large number of DTs also reduces the problem of overfitting [20].

3) Logistic Regression: Unlike any regression problem, LR maps input to outputs by estimating the values of coefficients of the hypothesis function. A hypothesis function is the final output of a function estimator after tuning all the coefficients. A simple regression problem is given by the following equation 3-4,

$$y=w^Tx+\varepsilon \tag{3}$$

$$h(x)=w^Tx \tag{4}$$

Here, $h(x)$ is the hypothesis function which needs to be estimated. In comparison to other regression methods, LR works more like a classifier. The output of the hypothesis function ranges between 0 and 1. Which can be interpreted as two classes of a binary classification problem. The modification in LR lies in the choice of hypothesis function. A LR employs the sigmoid function, otherwise known as logistic function.

$$g(k)=\frac{1}{1+e^{-k}} \tag{5}$$

Here in equation 5 $g(k)$ denotes a sigmoid function parameterized in k. A logistic function is based on rule of logitboost learning with linear regression function as shown in equation 6.

$$g(w^T x) = \frac{1}{1+e^{-w^T x}} \tag{6}$$

While this model is more suitable for binary classification. It can, with little modification, be used as multiclass classifier. The modification lies in the choice of thresholds (in the range of 0 to 1). Coefficients can be updated on the basis of loss minimization. Which is achieved by using a very primitive optimization tool called gradient descent. The weight update equation at j^{th} iteration, using gradient descent is given by equation 7.

$$w_j \leftarrow w_j + \alpha \sum_{i=1}^{m} (y^{(i)} - h_w(x^{(i)}))x_j^{(i)} \tag{7}$$

Here, m is the number of training samples. w_j is the updated coefficients. By making use of the coefficient, it translates the input features into a value that lies between 0 and 1 [21], [22].

4) AdaBoost: The ADB technique is an extension of boosting classifier which follows the policy of giving more weightage to incorrectly classified data. All the input features are given equal weights in the base learner, which is a basic one level DT as described in equation 8.

$$w_i = \frac{1}{n} \tag{8}$$

While training each tree, the algorithm updates the weight of each instance in a manner that it focuses learning on a particular set of features (incorrectly classified) that already have a significant weight applied to them. At each stage of the classification process, the learning instance undergoes re-weighting which is determined by the output. If the classifier's output is correct, the weights are decreased; however, if the classification is incorrect, the weights are raised. Mathematically, it is implemented using the following equations 9-10.

$$\alpha_t = \frac{1}{2}\log_e\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right) \tag{9}$$

$$w_{t+1} = \begin{cases} w_t \times e^{\alpha} & \text{if incorrectly classified} \\ w_t \times e^{-\alpha} & \text{if correctly classified} \end{cases} \tag{10}$$

Here α_t is the performance parameter after t iterations and ε is the total error in that iteration due to the choice of a particular weight value. w_t is the value of weightage given to a particular instance. At each and every stage, the level of difficulty is assessed. Level of difficulty is higher for an incorrect classification. Hence, accurately classifying occurrences are much frequent, while incorrectly classifying them is more difficult and less frequent. By following this strategy, the algorithm is able to produce specialists who will compete with one another to provide better results. Moreover, its sequential nature allows for improvement in model. And at last, ADB gives the averaged model of all base learners like other boosting models. [23], [24].

5) Naive Bayes: The probabilistic model formulation is one way to classify given input instance *x*. For this purpose, the Maximum aposterior (MAP) estimation is applied on the data in order to estimate the parameters corresponding to this classifier [25]. The equation of posterior probability according to bayes theorem is given by equation 11.

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)} \tag{11}$$

Here $P(y|x)$ is the posterior probability found using $P(x|y)$, the likelihood $P(y)$, the class prior, and $P(x)$, the likelihood prior probability. The goal of MAP is to estimate parameters to maximize the posterior probability. However, the estimation of $P(x|y)$ is much more complex then it seems, due to the occurrence of joint probabilities in the calculation. To ease the complexities, some assumptions are incorporated in the NB classifier. That assumption being, the independency of features within a class. So, the likelihood can be re-written as in equation 12.

$$P(x|y) = \prod_{i=1}^{n} P(x_i|y) \tag{12}$$

During the training period, the NB classifier estimates $P(y)$ for all the output classes and $P(x_i|y)$ for all the input features in the training dataset. In this activity categorization, we will be basing our decisions on the classes that have the highest probability [26], which is obtained following the given equation 13

$$.P(y|x) \propto P(y)\prod_{i=1}^{n} P(x_i|y) \tag{13}$$

III. Result and Discussion

This section provides an overview of the experimental outcomes of classification of sEMG signal obtained from TE, ME, FME, FMTE, FMRE, and HC finger activity. The 17 features, as discussed in TABLE I, are calculated for two channels of sEMG signals and provided as input to the classifiers.

To perform classification, the dataset of extracted features is divided as follows: 70% of the data is used for training the classifier, while the remaining 30% is utilized to evaluate the accuracy of the classifier. Accuracy, Precision, Recall, and F1-score are the performance metrics that are being measured to evaluate the classifier. These metrics are defined as follows:

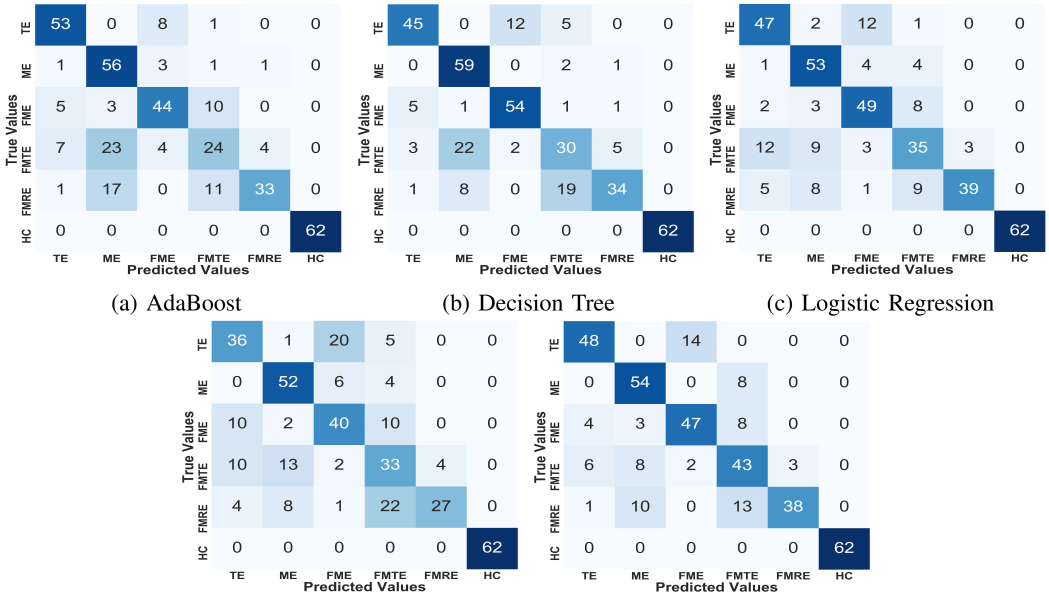
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

$$F1-score = 2 * \frac{PrecisionRecall}{Precision+Recall} \tag{17}$$

Here,



(a) AdaBoost (b) Decision Tree (c) Logistic Regression

(d) Naive Bayes (e) Random Forest

Fig. 2: Confusion Matrix of Classifier

- TP (True Positive): The data are classified as true while they are true.
- TN (True Negative): The data are classified as false while they are false.
- FP (False Positive): The data are classified as true while they are false.
- FN (False Negative): The data are classified as false while they are true.

TABLE II: Performance analysis of classifier in (%)

Classifier	Accuracy	Precision	Recall	F1-score
Random Forest	78.49	80.06	78.49	78.55
Logistic Regression	76.61	77.68	76.61	76.48
Decision Tree	76.34	77.31	76.34	75.79
AdaBoost	73.38	74.97	73.66	73.09
Naïve bayes	67.20	69.68	67.20	67.01

This article makes use of five different classifier model, RF, LR, DT, ADB, and NB. TABLE II gives a comparison between the employed ML classifiers on the basis of the four performance parameters. From the TABLE II it is clear that, the RF gives the highest accuracy, of about 78.49%. Next to RF, LR, and DT attain high levels of accuracy. It can be inferred that RF performed better for our dataset. Apart from accuracy, it also has the highest F1-score, precision and recall (78.55%, 80.06%, and 78.49%, respectively) among all classifiers that are considered during the study. Next to RF, LR and DT produce higher F1-score precision and recall than other classifiers. In this case, NB has the lowest performance metrics of all. As a result, it is observed that on this data NB does not permit multiclass interaction.

TABLE III highlights the classwise accuracy for the five ML classifiers employed in this work for identifying six finger activity. ADB is the highly accurate classifier for TE activity with 85.48%. DT dominated the ME and FME activity with 95.16% and 87.10% accuracy. The leading classifier for FMTE activity is RF with 69.35% accurate. LR classify FMRE activity with 62.90%. Within HC activity, all the classifiers gives equal accuracy of hundred percent correctness rate. This action is known as the rest posture.

TABLE III: Classwise comparision of classifier in (%)

Classifier	TE	ME	FME	FMTE	FMRE	HC
Random Forest	77.41	87.09	75.80	69.35	61.29	100.00
Logistic Regression	75.81	85.48	79.03	56.45	62.90	100.00
Decision Tree	72.58	95.16	87.10	48.39	54.84	100.00
AdaBoost	85.48	90.32	70.96	38.70	53.22	100.00
Naïve bayes	58.06	83.87	64.52	53.23	43.55	100.00

Another performance evaluation of classifiers, has been done with the help of a confusion matrix. It contains information about the actual and expected labels that a model predicts. A confusion matrix for all the classifiers has been shown in the Fig. 2. In the confusion matrix, the diagonal element represents the number of test instances that has been correctly classified whereas off diagonal represents the incorrect classified. Accuracy is the sum of the diagonal components of the confusion matrix divided by sum of all the components in the matrix. Therefore, by intuition, if the diagonal components in the confusion matrix have a large value, compared to the remaining values in the matrix, then model is a good classifier. And it can be seen in Fig. 2(e) that, RF matrix is most remarkable and has a greater value than any other classifier model on the diagonal position of the confusion matrix as shown in Fig. 2.

According to Fig.2(e), 48 samples of TE class, 54 samples of ME class, 47 samples of FME class, 43 samples of FMTE class, 38 samples of FMRE class, and 62 samples of HC class are correctly classified, whereas 14 samples of TE class, 8 samples of ME class, 15 samples of FME class, 19 samples of FMTE class, 24 samples of FMRE class, and zero samples of HC class are incorrectly classified. Similarly, the confusion matrix of the remaining ML classifiers can be inferred from Fig. 2(a)-(d).

IV. Conclusion and Future Work

sEMG based signal of a single subject performing six finger based activities has been acquired using two channels. The activities include: TE, ME, FME, FMTE, FMRE, HC. On the acquired signal overlapping windowing approach has been used to segment the signal and extract seventeen time and frequency domain features from each segment. The extracted feature has been used as input to ML classifiers. Five different ML classifiers are used. Among all classifiers, RF is the best of all the classifiers based on performance metrics, on the basis of which classifiers are evaluated. The reason for such robustness of the RF classifier is the employment of several DTs and usage of voting mechanism to finalize on a particular output of classifier which can tackle the problem of outliers and noise relatively much better than others. Next to RF, LR, and DT have comparable accuracy and competitive performance. NB has the worst classification outcomes on this dataset.

It's possible, in future, to implement more ML classifiers to obtain even better degree of accuracy. Perhaps, using optimization techniques features can be optimally chosen to obtain even better classification performance metrics considering lower number of features, which will reduce time and calculation complexity of trained model. Deep Learning is another budding domain in the field of classification of signals, which can be implemented instead of weak classifiers, and which can enable, maybe, even better accuracy. This work can be extended to benefit real life hand amputees by stimulating robotic arms and fingers.

References

[1] K. D. Seelman, "The world health organization/world bank's first world report on disability," *International journal of telerehabilitation*, vol. 3, no. 2, p. 11, 2011.

[2] C. Siegel and T. E. Dorner, "Information technologies for active and assisted living—influences to the quality of life of an ageing society," *International journal of medical informatics*, vol. 100, pp. 32–45, 2017.

[3] C. Pylatiuk, S. Schulz, and L. Döderlein, "Results of an internet survey of myoelectric prosthetic hand users," *Prosthetics and orthotics international*, vol. 31, no. 4, pp. 362–370, 2007.

[4] Y. Fu, J. Zhao, Y. Dong, and X. Wang, "Dry electrodes for human bioelectrical signal monitoring," *Sensors*, vol. 20, no. 13, p. 3651, 2020.

[5] A. Vijayvargiya, B. Singh, R. Kumar, and J. M. R. Tavares, "Human lower limb activity recognition techniques, databases, challenges and its applications using semg signal: an overview," *Biomedical Engineering Letters*, vol. 12, no. 4, pp. 343–358, 2022.

[6] M. Hakonen, H. Piitulainen, and A. Visala, "Current state of digital signal processing in myoelectric interfaces and related applications," *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, 2015.

[7] M. Jiang, R. Wang, J. Wang, and D. Jin, "A method of recognizing finger motion using wavelet transform of surface emg signal," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pp. 2672–2674, IEEE, 2006.

[8] F. V. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. V. Thakor, "Decoding of individuated finger movements using surface electromyography," *IEEE transactions on biomedical engineering*, vol. 56, no. 5, pp. 1427–1434, 2008.

[9] G. R. Naik, D. K. Kumar, and S. Arjunan, "Use of semg in identification of low level muscle activities: features based on ica and fractal dimension," in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 364–367, IEEE, 2009.

[10] G. R. Kanitz, C. Antfolk, C. Cipriani, F. Sebelius, and M. C. Carrozza, "Decoding of individuated finger movements using surface emg and input optimization applying a genetic algorithm," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 1608–1611, IEEE, 2011.

[11] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE transactions on biomedical engineering*, vol. 54, no. 5, pp. 847–853, 2007.

[12] O. ULKIR, G. Gokmen, and E. KAPLANOGLU, "Emg signal classification using fuzzy logic," *Balkan Journal of Electrical and Computer Engineering*, vol. 5, no. 2, pp. 97–101, 2017.

[13] A. Vijayvargiya, R. Kumar, N. Dey, and J. M. R. Tavares, "Comparative analysis of machine learning techniques for the classification of knee abnormality," in *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, pp. 1–6, IEEE, 2020.

[14] A. Vijayvargiya, B. Singh, R. Kumar, U. Desai, J. Hemanth, *et al.*, "Hybrid deep learning approaches for semg signal-based lower limb activity recognition," *Mathematical Problems in Engineering*, vol. 2022, 2022.

[15] A. Vijayvargiya, V. Gupta, R. Kumar, N. Dey, and J. M. R. Tavares, "A hybrid wd-eemd semg feature extraction technique for lower limb activity recognition," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20431–20439, 2021.

[16] U. Côté-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 27, no. 4, pp. 760–771, 2019.

[17] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for emg signal classification," *Expert systems with applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[18] P. L. Singh, S. M. Verma, A. Vijayvargiya, and R. Kumar, "Wd-eemd based voting classifier for hand gestures classification using semg signals," in *2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA)*, pp. 225–230, IEEE, 2021.

[19] A. Vijayvargiya, B. Dubey, N. Kumari, K. Kumar, H. Suthar, and R. Kumar, "semg sensor-based human lower limb activity recognition using machine learning algorithms," in *2022 IEEE International Conference on Data Science and Information System (ICDSIS)*, pp. 1–6, IEEE, 2022.

[20] N. Y. Hassan, W. H. Gomaa, G. A. Khoriba, and M. H. Haggag, "Supervised learning approach for twitter credibility detection," in *2018 13th International conference on computer engineering and systems (ICCES)*, pp. 196–201, IEEE, 2018.

[21] M. P. LaValley, "Logistic regression," *Circulation*, vol. 117, no. 18, pp. 2395–2399, 2008.

[22] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," *Machine learning*, vol. 59, pp. 161–205, 2005.

[23] D. P. Solomatine and D. L. Shrestha, "Adaboost. rt: a boosting algorithm for regression problems," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, vol. 2, pp. 1163–1168, IEEE, 2004.

[24] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[25] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. CRC press, 2012.

[26] M. K. Jain, D. Gopalani, Y. K. Meena, and R. Kumar, "Machine learning based fake news detection using linguistic features and word vector features," in *2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pp. 1–6, IEEE, 2020.