Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

IJ EEE
University of Basrah
College of Engineering

Design of Hand Gesture Classification System Based on High Density-Surface Electromyography Accompanied Force Myography

Alya Ghazi Darweesh*, Mofeed Turky Rashid
Electrical Engineering Department, University of Basrah, Basrah, Iraq.

Correspondance
*Alya Ghazi Darweesh
Electrical Engineering Department,
University of Basrah, Basrah, Iraq.
Email: engineeralyaa2021@gmail.com

Abstract
A robust system that classifies various hand gestures would greatly help those using prosthetic limbs. Recently, emphasis has been placed on extracted features from the High Density - surface Electromyography (HD-sEMG) signals and the size of segmentation windows which augment the recognition accuracy. This paper proposes a hand gestures identification system, in which HD-sEMG signals are employed, and is supported by Force Myography (FMG) signals for this mission. Several feature types have been extracted from FMG and HD-sEMG signals such as MEAN, RMS, MAD, STD, and Variance, these features have been validated under some classifiers such as decision tree (DT), linear discriminant analysis (LDA), support vector machine SVM, and k-nearest neighbor (KNN), in which results showing that MEAN and RMS features are superior to others, while the best classifier is SVM. Several experiments have been achieved by the MATLAB platform to validate the proposed system, in which, a database of HD-sEMG signals comprising 65 isometric hand gestures is employed, where two (8×8) electrodes and 9 force sensors are used to collect the FMG data. This data was derived from 20 intact participants, the first preprocessing step was applied during the recording stage. Ten gestures are chosen to be classified from the 65 hand gestures. Results show the success of the proposed system while the classification accuracy arrived at 99.1%.

Keywords
Hand Gesture, HD-sEMG, FMG, Features Extraction, Classifications.

NOMENCLATURE

To familiarize oneself with the scientific notations used in this paper, Table I provides the convention applied in this paper.

I. INTRODUCTION

An estimated three million people worldwide are using or need a prosthetic hand [1]. Hence, the need for an artificial limb that is easy to handle is one of the issues that preoccupies scientists and engineers. Many prosthetic devices are often restricted to producing only basic functions, such as opening-closing. This limitation was the motivation to improve the gesture recognition system by any means, therefore it could contribute to the design of the prosthetic device controller [2].

The history of hand gesture recognition (HGR) systems started with the invention of glove-based control interfaces [3]. Over the past 35 years, many successful products have developed with high-efficiency performance. Gestures (hand movements) result from the nervous system and muscle interaction. Muscles controlling the hand are located in the nearest parts of the hand and they have long, sometimes multiple tendons crossing several joints and sometimes acting on various fingers [4]. Thus, control of hand movements requires a complex pattern of neural activation and self-control

TABLE I.
THE LIST OF ABBREVIATIONS THAT USED IN THIS PAPER.

Symbol	Description
AI	Artificial Intelligence
CNN	Convolution Neural Network
CNT/PDMS	Carbon Nanotube/Poly(dimethylseloxane)
CT-HGR	Compact Transformer-based Hand Gesture Recognition
CWT	Continuous Wavelet Transform
d-biLSTM	dilated bi-directional Long Short-Term Memory
DAQ	Omega Data Acquisition
DB-a	Database-a
DB-b	Database-b
DB-c	Database-c
DNN	Deep Neural Networks
DT	Decision Tree
DTBO	Driving Training Based Optimization
DTW	Dynamic Time Warping
ECG	Electrocardiography
EMG	Electromyography
FDF	Frequency Domain Features
FFT	Fast Fourier Transform
FMG	Force Myography
GNNs	Graph Neural Networks
HD-sEMG	High-density surface Electromyography
HGR	Hand Gesture Recognition
HOG	Histogram Oriented Gradient
ISVM	Incremental Adaptive SVM Classifier
KNN	K-Nearest Nearest Neighbor
LDA	Linear Discriminant Analysis
LSTM	Long Short-Term Memory
MAD	Mean Absolute Deviation
MDF	Median Frequency
ML	Machine Learning
MMG	Mechanomyography
MNF	Mean Frequency
NN	Neural Network
RCNN	Recurrent Convolution neural network
RMS	Root Mean Square
RNN	Recurrent Neural Network
Skew	Skewness
SSC	Sign Slope Change
STBO	Sewing Training-Based Optimization
STD	Standard Deviation
STW	Spatio-Temporal Warping
SVMs	Support Vector Machine
TDF	Time Domain Features
TFDF	Time-Frequency Domain Features
TTL	Transistor-Transistor Logic
Var	Variance
WL	Waveform Length
ZC	Zero Crossing

of several muscles. The complexity of this system becomes obvious following amputation of a hand and forearm where central command and the proximal nerve signal remain operational but without graspers to complete the desired action. Gestures can be one degree of freedom (require one finger movement or wrist rotation), two degrees of freedom (two fingers are needed to perform the posture), or N degrees of freedom (more than two fingers are required to accomplish the movement).

Hand gesture recognition sensors are devices that can detect and demonstrate human hand movements. These sensors use different technologies such as infrared, ultrasonic, camera-based systems, or wearable devices to capture the electromyography (EMG)signals accompanying each movement. The main steps of the hand gesture recognition system are:

- Data Acquisition: The first step in the process is data acquisition. This involves collecting the signals or photos using the sensors or cameras.

- Preprocessing: The data captured by the sensor or the camera is usually noisy and contains irrelevant information. Preprocessing involves filtering this data to remove noise. This could demand techniques such as image segmentation in the case of camera-based systems, where the hand is separated from the background, or it could demand filtering in the case of raw data to remove the power line noise.

- Feature Extraction: Once the data has been preprocessed, the next step is feature extraction. This involves identifying and extracting the key features that will be used to recognize the hand gesture. These features could require the position or shape in camera-based systems. In wearable systems, they require time-domain or frequency-domain or time-frequency domain features.

- Classification: The extracted features are then fed into a classification algorithm. This algorithm uses machine learning techniques to classify the hand gesture based on the extracted features. The output of the classification algorithm is a label that identifies the hand gesture.

Starting with the data acquisition step mentioned earlier, in addition to the camera-based systems there are wearable sensors like surface electromyography electrodes which include three kinds

1. Spars that contain individual electrodes called seeds.

2. Myo-armband that contains eight electrodes arranged in a bracelet-like. Both 1 and 2 are called surface-EMG.

3. High-density surface electromyography electrodes (HD-sEMG) which is the best among them because they cover a larger area therefore giving more accurate results.

That's why Yu Du et al. [5] used high-density surface electromyography (HD-sEMG) electrodes which yielded the CapgMyo dataset with three sub-datasets DB-a, DB-b, and DB-c that helped in most research [24]. Still, it didn't involve all 23 participants.

Mechanomyography (MMG) can also be used to detect muscular activities. Muscular activity is identified by mechanical vibration, which is generated by the tremble of each muscle fiber. MMG-based methods commonly use an accelerometer and a microphone. However, MMG based on an accelerometer can only be used when the magnitude of acceleration is distinguishable compared to acceleration due to gravity and motion [6, 7].

Recently, the Force Myography (FMG) system that deals with force signals has gained attention as it greatly supports the accuracy of gesture recognition when used in Prosthesis systems [8].

Gesture recognition using a camera faces many challenges. These include diverse contexts, multiple interpretations, low visibility, low contrast, and spatial and temporal differences in gestures. In addition, the intensive computations to separate the hand from a complex background and diverse lighting conditions also contribute to the difficulty of achieving high performance and accuracy in gesture recognition. In [9], Rui Ma et al. preprocessed the image data collected by the depth information sensor and then extracted the corresponding human pose image features, which is the region of interest, and finally used the appropriate classification algorithm to classify the pose. In addition, this method is impossible to use for amputees.

Preprocessing data is the second step in the hand gesture recognition system, which includes, filtering, to remove the noise of the power line signal which can be around 50 or 60 Hz, or filtering data to choose the period when the gesture is more effective, and then segmentation, in which the data will be available to deal with by the machine learning algorithms because they can't deal with huge data, and finally normalization, sometimes is needed when data have large or very small values. Machine learning algorithms may struggle to process huge datasets effectively without appropriate preprocessing and optimization.

The third step is the feature extraction which directly affects the classification accuracy value. As an example, Haiqiang Duan et al. [10] used two-dimensional root-mean-squared (2D RMS) of the monopolar sEMG recordings acquired from $4 \times 8 \times 8$ channels (four electrodes with eight rows and eight columns of electrodes) as a spatial extracted feature. Harun Güneş et al. [11] used Continuous Wavelet Transform (CWT) which is one of the time-frequency domain feature extractors. Hanady Jaber et al. [12] used Amplitude and intensity features compounds with Histogram Oriented Gradient (HOG) features to improve the classification accuracy. Generally, features can be divided into many categories, time domain features (TDF) like root mean square (RMS), standard deviation (SD), variance (V), zero crossing (ZC), skewness (S), and many other time domain features, then there are frequency domain features (FDF). For frequency-domain features, the function's FFT (Fast Fourier Transform) and the corresponding power spectrum must first be obtained. Then, the mean, variance, and peak of the band power spectrum can be evaluated, as well as many other frequency-domain features. Lastly, there are time-frequency domain features (TFDF) which include a transform window technique. With this method, wide time resolution and narrow frequency resolution at high frequencies, narrow time resolution, and wide frequency resolution at low frequencies are obtained because of the changing window size which is very suitable for signals with abrupt changes like EMG or ECG signals.

For the classification stage, to improve the accuracy of classification, intuitiveness, and control performance of hand prosthetic systems, Artificial Intelligence (AI) algorithms range from conventional Machine Learning (ML) models to highly complicated Deep Neural Networks (DNN) [13, 14]. Conventional ML models, such as SVMs and LDAs, utilized for sEMG-based hand gesture recognition, typically work well even when dealing with small datasets, which depend on manual extraction of handcrafted (engineered) features, which limits their generalizability as human knowledge is needed to find the best set of features. Milad Jabbari et al. [15] used a novel feature extraction method. Dynamic time warping (DTW) was employed to efficiently capture the nonlinear similarity between the EMG signals. For the temporal aspect, they developed a novel feature, named spatio-temporal warping (STW). The limitation of this study is that the DTW algorithm may prove too complex for real-time implementation and may require additional hardware for parallel processing. Abid Saeed Khattak et al. [16] developed an efficacious scheme for hand gesture recognition using SDTO (which is a combination of Sewing Training Based Optimization). In which, signal pre-processing is done through Gaussian filtering. Thereafter, desired and appropriate features are extracted. Following that, effective features are chosen using SDTO. At last, hand gesture identification is accomplished based on DRN, and this network is effectively fine-tuned by SDTO, which is a combination of Sewing Training Based Optimization (STBO) and Driving Training Based Optimization (DTBO). The designed model has gained a maximum accuracy of 94.3%. The dataset used in this study was based on three wearable MYO-armband

sensors that give less accuracy than HD-sEMG electrodes since the last one covers a larger area of the participant's forearm. Silvia Maria Massa et al. [17] introduced the use of Graph Neural Networks (GNNs) to process HD EMG data for hand gesture recognition of people with an amputation who use a robotic prosthetic hand. The researchers obtained an average classification error rate of 8.75% with a standard deviation of 4.92, and 45 out of 65 gestures were detected with an error rate of less than 10%. This method can be applied to robotic arm control, but unlike this work, may be problematic due to the reduced residual muscle capacity of disabled people. Various approaches have been proposed to identify movement intuition and lead to promising achievements. Hanady Jaber et al. [7, 12, 14] used conventional classifiers represented by several types of Support Vector Machine (SVM) classifiers such as the original SVM model and an incremental adaptive SVM classifier (ISVM) these proposed systems achieved good accuracy and robustness in performance, but the shortages in this study have not compared with other conventional classifiers for the same case study. Figure 1 the common system for hand gesture recognition using wearable EMG sensors.

In this work ten gestures were selected from a combination of 65 gestures database [18], The sEMG data were recorded at the level of the forearm from 20 able-bodied and healthy participants with two anterior EMG electrodes and 9 force gauges that record the force accompanied with these EMG signals. This study aims to classify those ten gestures using the force signals accompanied by the HD-sEMG signals. The advantage of FMG over EMG is that FMG signals have shown a higher overall stability over time. In addition, the variance of the signal is lower than EMG, and FMG signals have shown a distinct pattern in transient responses. Force signals also require fewer force-sensing resistors (pressure gauges) to detect a change in pressure due to the contraction of the muscles. The overall implementation of FMG is less expensive in terms of sensors and equipment requirements [8]. In comparison with other existing works, our work is applicable because all signals were recorded in an isometric manner to simulate the lost hand.

The proposed system classified the desired gestures through:

1. The desired data of the ten gestures was separated for the five repetitions.

2. A suitable window size was selected for segmentation (62.5 and 125 ms) consequently.

3. Suitable features were elected for their importance in machine learning-based algorithms, five time-domain features were evaluated, RMS, MAD, STD, Var, and Mean).

4. The data have been split into training (80%) and testing (20%) data.

5. The classification process was implemented with the help of four machine learning algorithms (DT, LDA, SVM, and KNN) to optimize the performance of the suggested system and compare results.

The proposed system has been validated and performance evaluated through the MATLAB platform. Several experiments have been achieved, in which, the results show that force signals can be used to design a robust hand gesture recognition system to improve classification accuracy. The contribution of this work can be considered as:

1. Using the FMG signal, which is supported by the HD-sEMG signal.

2. Selecting a suitable window size for the segmentation.

3. Robust features have been employed which are RMS, MAD, STD, Var, and Mean.

4. For the proposed system, simple and low-cost classification algorithms have been chosen (DT, LDA, SVM, and KNN).

II. METHODOLOGY

Figure 2 illustrates the block diagram of the proposed system; the dataset includes HD-sEMG and FMG signals that have been used for hand gesture identification [18]. The signals have been passed through the preprocessing to remove all unwanted data. Ten gestures data was separated and relabeled, furthermore, the segmentation has been performed. Five time-domain features have been extracted, which will be used in the classification process. Eventually, the classification assignment was accomplished successively through four machine-learning algorithms.

A. Data Acquisition

The biosignal is an electrical signal that is generated with any movement of the biological organs and can be recorded as a voltage signal which is called an Electromyography (EMG) signal. This signal is accompanied by another myography signal that is responsible for muscle contraction, which is denoted by the force myography signal (FMG). These two signals are the dataset source used in this study, and the recording system is shown in Fig. 3. The block diagram of the recording protocol can be observed in Fig. 4. [18]
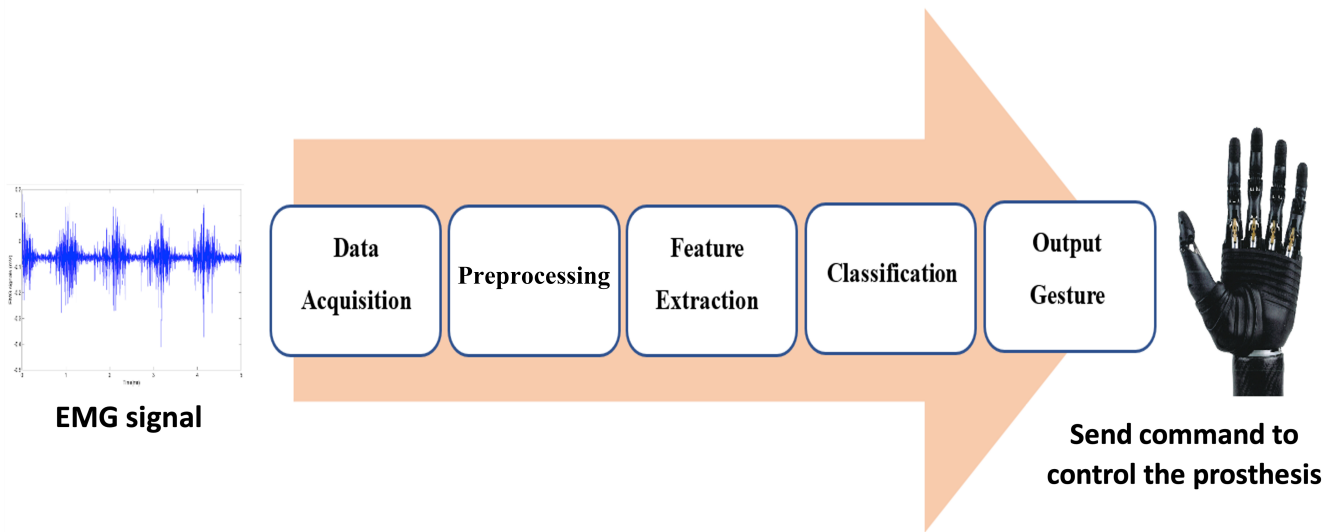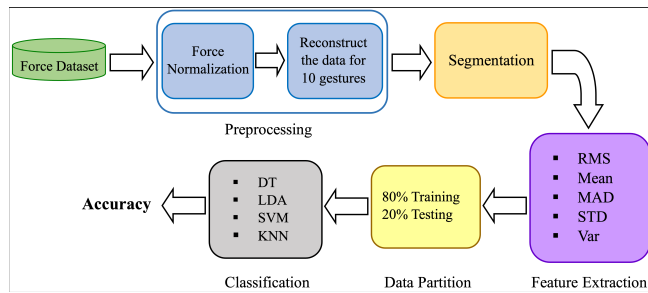
Fig. 1. Common hand gesture recognition system.

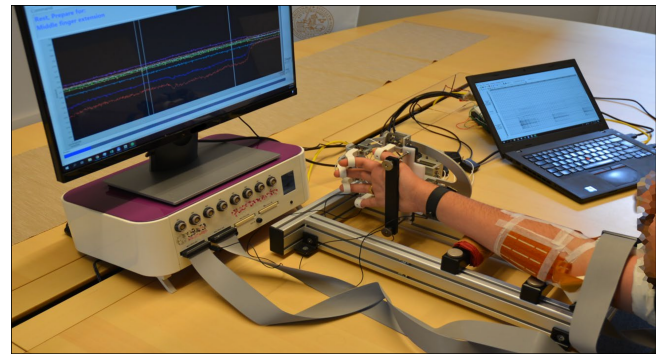Fig. 2. The block diagram of the proposed system.

Fig. 3. Data acquisition [18].

1) The HD-sEMG Dataset

The dataset used in this work [18] is a recently released HD-sEMG dataset that contains two 64-electrode square grids (8 × 8) with an inter-electrode distance of 10 mm (ELSCH064NM3, OT Bioelettronica, Torino, Italy), which were positioned on extensor and flexor muscles of the forearm of 20 participants (14 men and 6 women) aged between 25 and 57 years (mean age 35 years). The electrode used is shown in Fig. 5a and the layout of both electrodes is illustrated in Fig. 5b. The participants performed 65 hand gestures that are combinations of three groups, the first group contains 16 gestures each of one degree of freedom, 42 gestures each of two degrees of freedom, and 7 gestures each of N degree of freedom. The subjects performed each gesture 5 times Each of those repetitions would last for 5 seconds. The signals were recorded through a Quattrocento (OT Bioelettronica, Torino, Italy) bioelectrical amplifier system with a 2048 Hz sampling frequency. Preamplifiers with 5x gain are located at the electrode connectors, amplifiers within the Quattrocento device, and the A/D con-
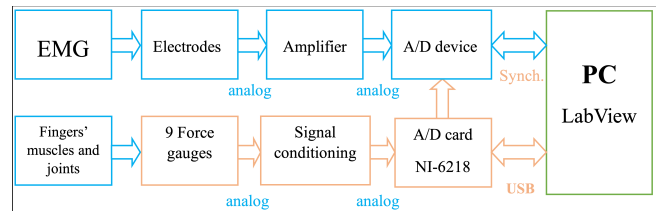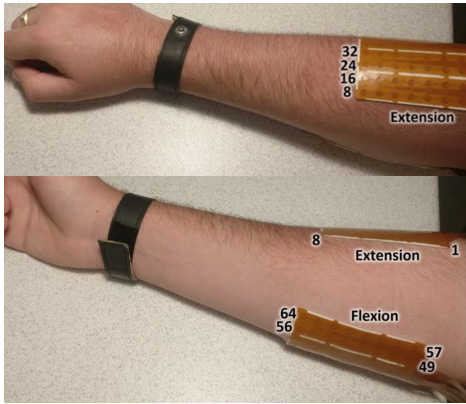

Fig. 4. Recording protocol [19].

verters. Overall, including preamplifiers and amplifiers, the HD-sEMG signals were amplified 150 times. Figure 6 shows the five trials for the first participant with the EMG signals implemented during each trial of the first gesture recorded from the extensor muscles.

2) Force Recording

All forces were recorded instantaneously with EMG signals in an isometric manner by using a custom-made force measure-
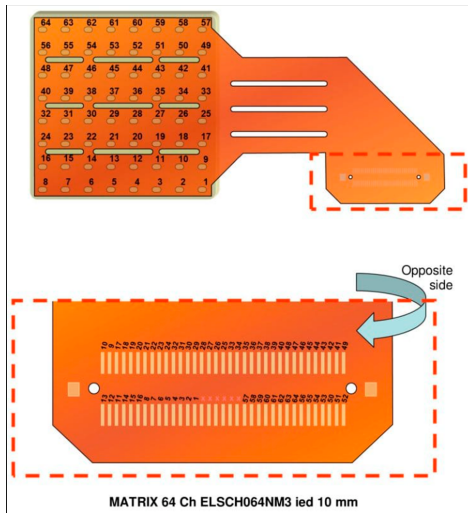
(a) The electrode used to record EMG signals.

(b) The electrode used to record EMG signals.

Fig. 5. HD-sEMG electrode square grids (8×8) [18].

Fig. 6. The EMG signals of the five trials for the first participant [18].
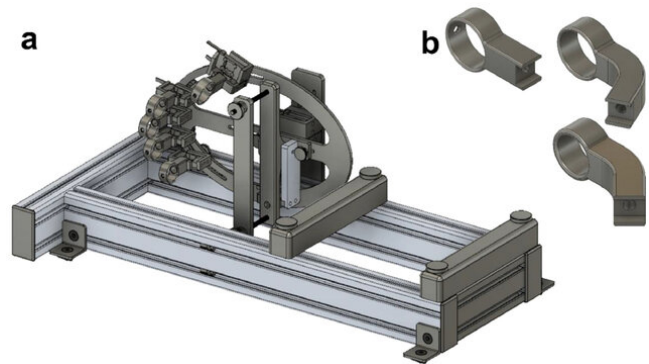
Fig. 7. (a) Force measurement device. (b) 3D printed finger brace [19].

ment device. The device is shown in Fig. 7. The reason for choosing an isometric setup was to simulate muscle behavior in a forearm amputee where the remaining muscles have a relatively small contraction amplitude.

The force measurement device comprises nine contract pressure meters, four measuring D2-D5 (flexion-extension) forces, two measuring thumb (flexion-extension), and (abduction-adduction), and three measuring wrists (flexion or extension), (radial or ulnar), and (pronation-supination) deviation. Those nine devices were interfaced through 3D-printed finger braces which were specially chosen for each subject. Using the braces, the fingers were placed in a neutral position, approximately in the middle of the range of motion. Figure 8 represents the force signal that coincides with the HD-sEMG signal that shown in Fig. 6 [18].
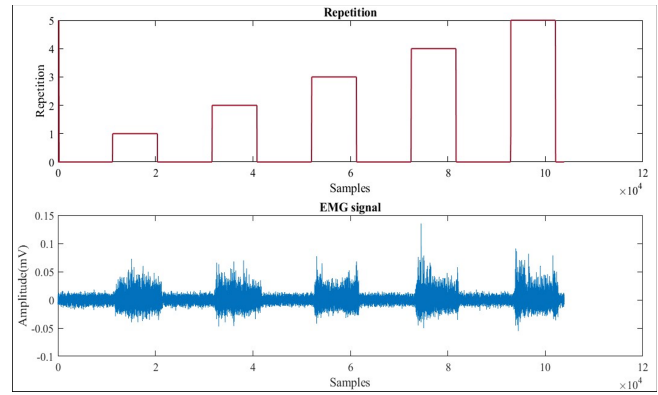
B. Hand Gestures

The hand gestures used in this study are shown in Table II. The first two gestures are one degree of freedom (1DoF), while the next six gestures are two degrees of freedom (2DoF), and the last two ones are N degree of freedom (NDoF).
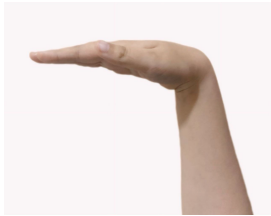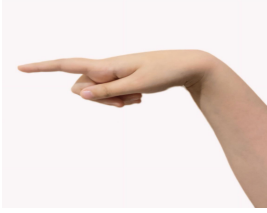
C. Data Preprocessing

Preprocessing is an efficient stage to enhance the robustness and reliability of data concerning potential pollutes. Augmenting the signal-to-noise ratio can improve the discriminating characteristics of EMG or force signals. The preprocessing steps can be seen in Fig. 9.

1) HD-sEMG Filtering

The first preprocessing step for the raw HD-sEMG dataset (sampled at a rate of 2048 Hz) was done during recording with a hardware high-pass filter at 10 Hz and a low-pass filter at 900 Hz. It was filtered offline to remove power line noise with a zero-phase 3rd order band-pass Butterworth filter centered at 50 Hz with 4 Hz.

TABLE II.
THE TEN GESTURES USED IN THIS WORK.

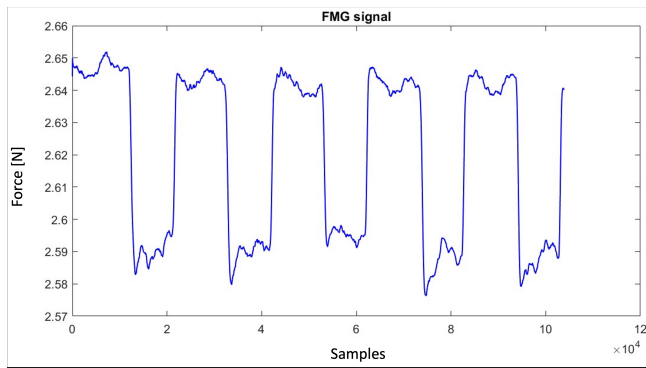| Label	Gesture Name	Gesture	Label	Gesture Name	Gesture
1	Wrist: bend		6	Ring finger: bend +Thumb: down	
2	Wrist: stretch		7	Middle finger: bend +Thumb: down	
3	Little finger: bend +Ring finger: bend		8	Index finger: bend +Thumb: down	
4	Little finger: bend +Thumb down		9	Palmer grasp	
5	Ring finger: bend +Middle finger: bend		10	Pointing	

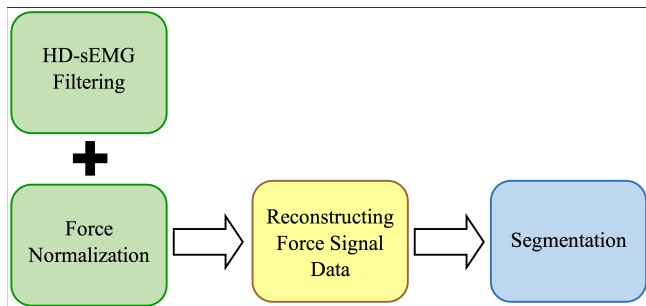Fig. 8. Force signal accompanied with the EMG signal in Fig. 6.

Fig. 9. Preprocessing steps flow chart.

2) Force Normalization

The measured force signals were normalized to analog signals in the range of 0-5 V for the force range ± 100 N, where in neutral position (0 N) the force sensors value was 2.5 V. The signals from the force sensors were digitalized using a NI-USB 6218 (National Instruments, Austin, Texas, USA) A/D with 16-bit amplitude resolution and sampling frequency of 200 Hz. The contemplation and the recording of the signals were programmed by a custom-made LabVIEW (National Instruments, Austin, Texas, USA) program. The same program used to control synchronization between the Quattrocento and the force signals by generating TTL pulses recorded by both devices. The pulses generated by one of the digital outputs of NI-USB 6218 were 0.2 s wide and occurred every 2 s. These forces are given in volts [V], and the relation between sensor analog output and the force is as follow:

$$\text{force} = \text{analog\_voltage} \times 40 - 100 \qquad [V] \qquad (1)$$

The protocol of EMG signal and force data acquisition is shown in Fig. 3.

HD-sEMG signals were displayed on a laptop screen after being amplified while hand forces and cues were shown on a separate screen on a table in front of the participant.

3) Reconstructing Data

The force myography signals recorded by the nine pressure sensors are used as a dataset to classify the ten gestures shown in Table II. The third Preprocessing step started with separating the force data for each gesture with its repetitions and then labeling the new data for using them later in the classification procedure.

4) Segmentation

Segmentation is a necessary preprocessing step to analyze any huge data because it reduces the dimensions of data which ensures better recognition when feeding it to machine learning classifiers, then the process of training will be faster and less computational, and segmentation is also preferable with non-stationary signals like EMG signals [20].

Windowing is a segmentation technique that can be implemented by two schemes, overlapping windowing and adjacent (non-overlapping) windowing, in adjacent window segmentation, successive windows are extracted from a time series by incrementally increasing an index by the window size. Overlapping windows have common parts between windows. Both schemes are shown in Fig. 10. In [21] Fernando D Farfán et al. suggested that the optimal window length was 200 ms, but in [22] Rami N. Khushaba et al. proved that, even with windows sizes as small as 32 ms, one can achieve considerably high classification accuracy that reaches to 90%.

In this work, adjacent (non-overlapping) windowing is used to segment the force signals into frames each of 128 samples (62.5 ms) for the first experiment and 256 samples (125 ms) for the second experiment.
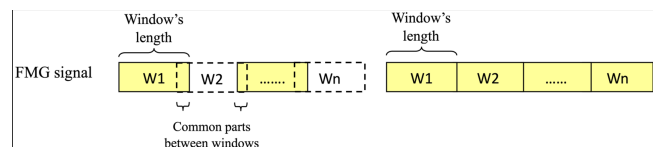
Fig. 10. Overlapping and non-overlapping segmentation.

D. Feature Extraction

Useful information can be extracted by transforming raw data into more effective features that can be processed while maintaining the information in the original data set. It leads to better results than feeding machine learning directly to the raw data.

Feature extraction can be implemented manually (an engineer or a scientist can choose features that are relevant for a given problem) like the mean of a window or automatically (specialized algorithms or deep networks are used to extract features automatically from signals or images without the need for human stepping in) like wavelet scattering.

Basically, there are three categories to extract features:

1. Time domain features (TDF): like Mean Absolute Value (MAV), Variance, Waveform Length (WL), Zero Crossings (ZC), Root Mean Square (RMS), Slope Sign Changes (SSC).

2. Frequency domain features (FDF): the Median Frequency (MDF), the Mean Frequency (MNF), and the Power Spectral Density, which can be extracted after transforming data to Fourier Transform (FT) or fast Fourier Transform (FFT).

3. Time-frequency domain features (TFDF): A wavelet scattering network which can derive, with minimal configuration, low-variance features from real-valued time series and image data for use in machine learning and deep learning applications. A key advantage it has over Fourier transforms is temporal resolution because it captures both frequency and location information (location in time).

Many researchers are still investigating new algorithms for extracting features that gives the best recognition accuracy. In [23] Zhongfu Ye et al. proposed a method employs higher order local autocorrelation (HLAC) features for feature extraction, while in [24] [25] Hanadi A Jaber et al. proposed spatial features (AIH)that evaluated by combining HOG features of HD-sEMG map and intensity features calculated from the average of segmented HD-sEMG map which is denoted as (AIH) features that gave an accuracy reaches 99.08%.

In this work, five-time domain features are chosen to be extracted from the segmented signals; these are:

1. Root Mean Square (RMS)

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{n}|x_i|^2} \qquad (2)$$

2. Mean Absolute Deviation (MAD)

$$MAD = \frac{1}{N}\sum_{j=1}^{N}|x_i - \bar{x}| \qquad (3)$$

Where $\bar{x}$ is the mean

3. Standard Deviation (STD) represents the difference between each sample of the signal and its mean value

$$STD = \left[\frac{1}{N-1}\sum_{1=1}^{N}(x_i - \bar{x})^2\right]^{1/2} \qquad (4)$$

4. Variance (Var) Variance represents the power of the signal

$$V_{ar} = \frac{1}{N-1}\sum_{i=1}^{N}x_i^2 \qquad (5)$$

5. Mean value (Mean)

$$\text{Mean} = \frac{1}{N}\sum_{1=1}^{N}x_i \qquad (6)$$

E. Classification Process

Classification is the task of assigning a class label to an input pattern. The class label indicates one of a given set of classes. The classification is carried out with the aid of a model obtained using a learning procedure. According to the type of learning used, there are two categories of classification algorithms, one using supervised and the other using unsupervised learning. Supervised learning makes use of a set of examples that already have the class labels assigned to them. Unsupervised learning attempts to find inherent structures in the data.

Many classifiers are categorized as supervised learning like the support-vector machine (SVM), k-Nearest Neighbors (kNN), Linear Discriminant Analysis (LDA), and Random Forests (RF). Boser et al. [26] introduced a training algorithm that maximizes the margin between the training patterns and the decision boundary by a linear combination of supporting patterns. This algorithm was trained to assign new examples to one of two categories, which was later called support vector machine (SVM).

Deep learning classifiers are branches of supervised algorithms. These classifiers are required to encode suitable distribution properties from raw signals to represent the bias and weight of neural network (NN) architecture. Deep learning needs a huge amount of data to achieve optimal performance such as convolution neural network CNN, recurrent neural network RNN, and Recurrent Convolution neural network RCNN. In [27] Silvia Maria Massa et al. exploited explainable AI algorithms to automatically refine the graph topology based on the data in order to improve recognition rates and reduce the computational cost.

In this work results were recorded from hold-out validation where data acquired for each gesture from the 20 subjects was divided into 80% training set that is used to train four classifiers then the rest data (20%) is used to test classifiers and extract the accuracy. This was achieved using the classification learner in MATLAB. The classifiers used are:

1) Decision Tree (DT)
DT is a supervised learning algorithm commonly used in machine learning to model and predict outcomes based on input

data. It is a tree-like structure where each internal node tests on a predictor, each branch corresponds to the predictor value and each leaf node represents the final decision or prediction. They can be used to solve both regression and classification problems. Decision trees can handle various data types (discrete or continuous values), they can also handle values with missing values.

2) Linear Discriminant Analysis (LDA)

LDA is also a supervised learning algorithm used for classification tasks in ML. Its technique is to find a linear combination of features that best separates the classes in a dataset, it works by distinguishing the data onto a lower-dimensional space that maximizes the separation between the classes. It does this by finding a set of linear discriminants that maximize the ratio of between-class variance to within-class variance. In other words, it finds the directions in the feature space that best separate the different classes of data. It is sometimes used as a feature extraction methodology. The LDA classifier was used because it can manage high-dimensional data, it handles multicollinearity, and its efficiency of computation.

3) Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm used for both classification and regression. The main objective of the SVM algorithm is to find the optimal hyperplane in an N-dimensional space that can separate the data points in different classes in the feature space. The hyperplane tries to maximize the margin between the closest points of different classes as much as possible. SVMs perform better with high-dimensional data and are less prone to overfitting.

4) k-Nearest Neighbor

The k-NN is one of the supervised machine learning algorithms. It is a classification and regression algorithm where neighbors contribute according to distance. Each neighbor weighs in proportion to the NB (Naïve Bayes) classifier. Naïve Bayes is part of a family of generative learning algorithms, meaning that it seeks to model the distribution of inputs of a given class or category. Unlike discriminative classifiers, like logistic regression, it does not learn which features are most important to differentiate between classes, which is 1/d, where d is the distance to the neighbor. This work selects k as 5 and Euclidean distance is used. KNN is simple to implement because it only requires a k value and a distance metric, and it adapts easily when new training samples are added, the algorithm adjusts to account for any new data since all training data is stored in memory. The four classifiers were chosen for their simplicity of implementation, lack of huge data requirements (unlike neural networks), and efficient computation, ensuring quick results.

F. Performance Metrics

The problem often confronted in classification interests making a suitable choice of a classification metric. The success of a classification metric is often tied to data because different classification metrics perform differently given different datasets. A classification metric can be constructed using different methods, but in most cases, the advantages and drawbacks of each one are often data-dependent. It is important to note that each of these metrics has a connection to the confusion matrix [28].

1) Confusion Matrix

A confusion matrix is an *nxn* table with information about the predictions of a classification model, it contains information about the true positive (TP), true negative (TN), false positive (FP), and false negative (FN).

2) Classification Accuracy

The accuracy rate can be defined as the number of correct classifications over the entire test set or contrarily the fraction of correct prediction of a classifier, over the entire test set. Based on the confusion matrix it is defined as:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

III. Experiments and Results

The selection of window size and features has a great influence on the recognition accuracy. Several experiments were executed to evaluate the performance of the proposed framework under different configurations. These experiments were carried out utilizing the following sequence of algorithmic methods:

1. Collect the force myography signals data for the ten chosen gestures shown in Table III with their five repetitions from the total set of 65 gestures dataset provided in [4].

2. The collected data has been given labels to recognize each gesture, these labels refer to every gesture by its number arranged in the same sequential order in Table III.

3. Every force column signal was segmented into a proper window size according to every experiment procedure.

4. The new segmented data was relabeled based on the new data size.

5. Adequate features were extracted for every window of the column data signal. The features are mentioned with the experiment name.

TABLE III.

THE AVERAGE CLASSIFICATION ACCURACY OF SINGLE TD FEATURE/CLASSIFIER COMBINATIONS ACROSS 20 SUBJECTS.

Feature Type	Average Classification Accuracy			
	DT	LDA	SVM	KNN
RMS	99	96.7	99	98.7
Mean	99	96.9	99	98.7
MAD	35.9	33.4	43.7	36.7
STD	37.9	32.5	44.4	37.3
Var	36.9	16.2	41	31.8

6. The "Classification Learner" in MATLAB was used to classify the gestures with four classification algorithms mentioned in 2.6 after splitting data into 80% training and 20% test data.

7. The analysis of experiments and the comparison of the results were implemented individually for every experiment.

A. Extracting Features in the Case 62.5 ms Window Size

In this experiment, the proposed features are validated in the case of 62.5 ms windows width, in which the classification accuracy of the classifier has been evaluated for each subject, furthermore, the average classification accuracy of the 20 participants has been calculated, as shown in Table III.

From Table III it can be seen that the RMS and the Mean features gave superior results upon the remaining three features, also it is clear that with RMS, Mean, STD, and Var features the SVM (Support Vector Machine) and the DT (Decision Tree) classifiers outperformed the other two classifiers, while for MAD features the SVM and KNN classifiers proved their performance efficiency.

Generally, the SVM classifier proved comparatively superior when compared to the other classifiers, and contrarily the LDA classifier ranks last in terms of accuracy of classification. All the five time-domain features established in Table III can be observed in Fig. 11.

Addressing the sixth participant data to investigate features and their effect on the system performance. Figure 12a represents the distribution of data after evaluating the RMS feature for the segmented windows. Each one of the ten gestures is illustrated with a different color, so it obviously can be separated easily, and the classification performance was so high that the accuracy reached 99% as can be seen from the scattering plot of the predicted model as shown in Fig. 12b and its confusion matrix as shown in Fig. 12c. Also extracting the mean value recorded a promising result reached 99.8% as shown in Fig. 13.
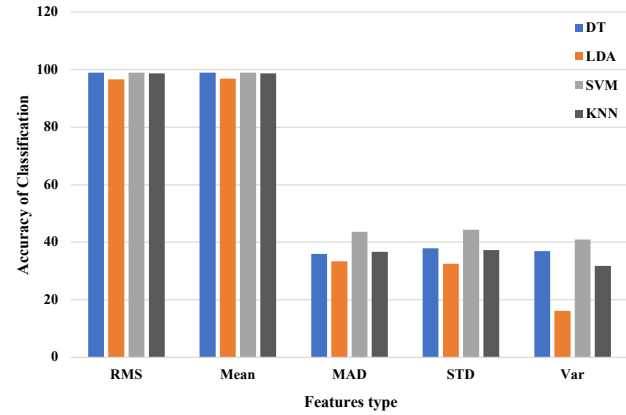

Fig. 11. Average Accuracy of classifiers vs several features type in the case of 62.5 ms window size.

Moving to the mean absolute deviation (MAD), extracting this feature led to haziness results as shown in Fig. 14, because most of the samples have the same MAD value according to normalization which made values between (0-5), therefore, the accuracy recorded was much less reached 64.1% in this case.

Fig. 15 implements the distribution of the ten classes after evaluating the standard deviation (STD) and the accuracy value is close to that achieved when the (MAD) feature was applied. The last feature chosen in this study was the variance whose results are shown in Fig. 16 and these results come in the last of the selected features list for their worse performance. From observing data and during the experiment performance recording for participants individually the variance recorded extremely bad results reached 16.2% accuracy with the LDA classifier. All of the mentioned outlines can be seen and recognized in Fig. 11.

It is crystal clear from Fig. 17 that with RMS and Mean features (a and e) how the data was distributed around a value that differs from one class to another, made the recognition process easier with accuracy records higher than those acquired from the MAD, STD, and Var features (b, c, and d) consequently.

B. Merge Features in the Case of 62.5 ms Window Size

From the previous experiment, we observed that RMS and MEAN are the best, and MAD, STD, and VAR are given less classification accuracy, therefore in this experiment, trying to increase the classification accuracy by proposing a feature that is the merging of the best together (RMS and MEAN), while the other experiment is to test the ability to improve the performance of other features by merging it (MAD, STD, and VAR), the results of experiments are shown in Table IV, furthermore, the average accuracy for 20 participants is
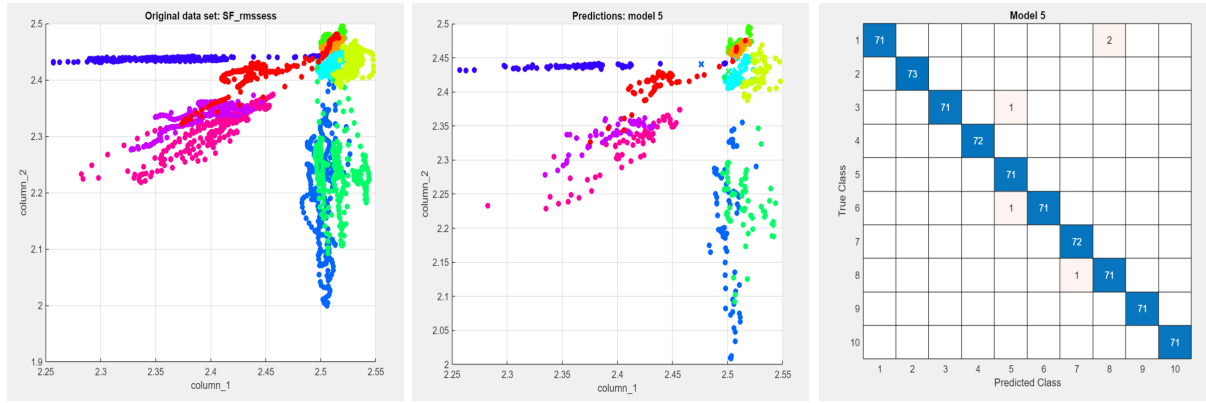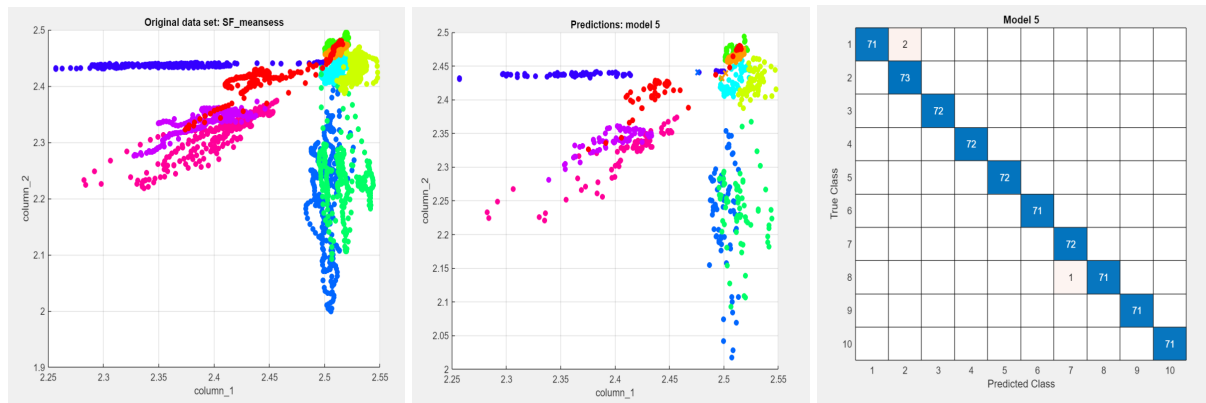
(a) Scattering plot before training.   (b) Scattering plot for predicted model.   (c) Confusion matrix for predicted model.

Fig. 12. RMS of the sixth subject data.

(a) Scattering plot before training.   (b) Scattering plot for predicted model.   (c) Confusion matrix for predicted model.

Fig. 13. Mean of the sixth subject data.

TABLE IV.

THE AVERAGE CLASSIFICATION ACCURACY OF THE TWO COMBINATIONS OF FEATURES ACROSS 20 SUBJECTS.

| Features Type	Average Classification Accuracy (%)			
	DT	LDA	SVM	KNN
RMS+Mean	99	96.7	99	99.5
MAD+STD+Var	38.6	37.3	45.9	37.9

evaluated.

To have a good view so that the results can be judged and compared, Figure 18 shows the results for both experiments, where it can be seen that the combination of RMS and MEAN led to a modest improvement in the classification accuracy value, while the other feature led to a better improvement but the classification accuracy is still much lower than the ambition.

### C. Merge All Proposed Features in the Case of 62.5 ms Window Size

Combining the five time-domain features can give a comprehensive evaluation to improve classification accuracy and control prostheses more effectively. The recognition accuracy recorded is shown in Table V. From Fig. 19 the SVM classifier (the red line) accomplished the best results, and the LDA classifier (the green line) scored the least accuracy, while both DT and KNN (yellow and blue) are close in outcomes. Also, some subjects (participants) recorded performance gained mastery over others. This can be due to skin impedance or the position of FMG sensors. As an example (from observing data) subject 16 took a time longer than other participants to execute each gesture which influenced his results to be the lowest in all experiments.

Succeeding with the same protocol to evaluate every gesture's accuracy and extracting the same five time-domain features proposed in this work, in this experiment, the first
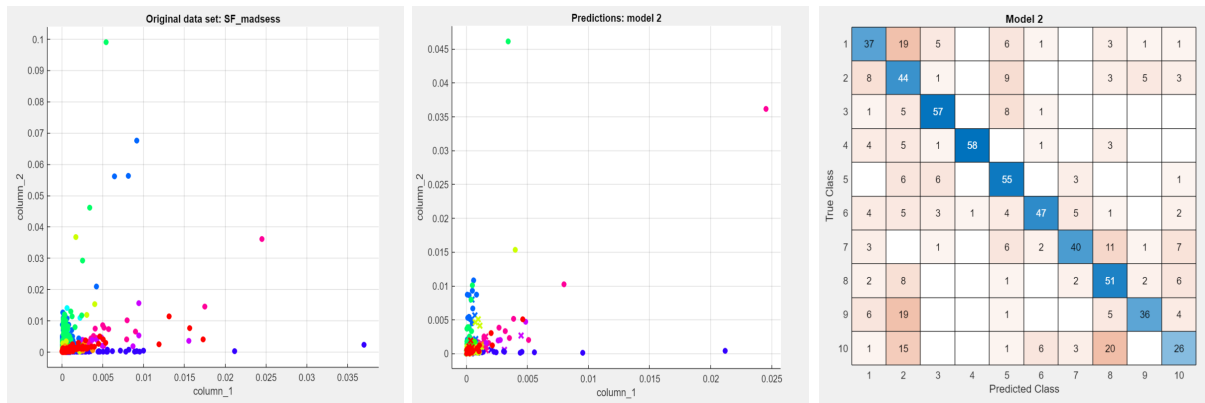
(a) Scattering plot before training. (b) Scattering plot for predicted model. (c) Confusion matrix for predicted model.

Fig. 14. Mean absolute deviation of the sixth subject data.
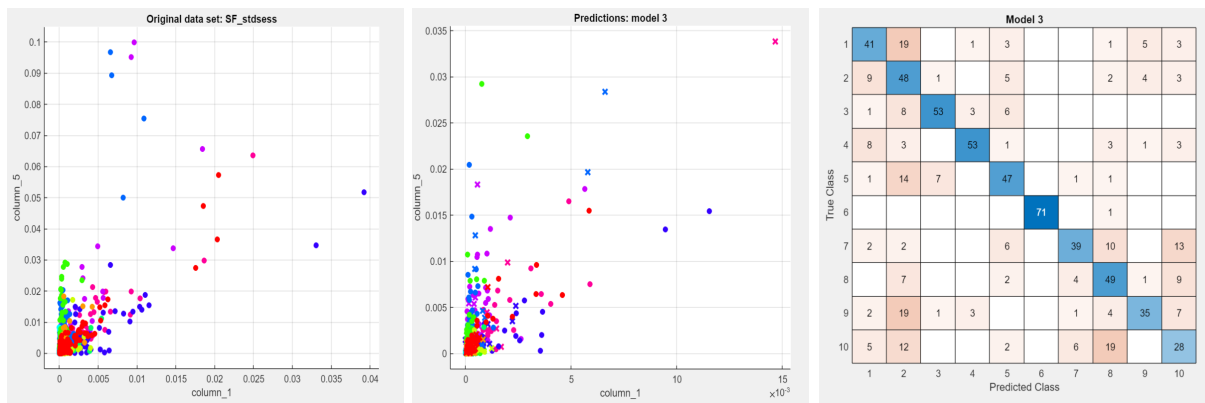
(a) Scattering plot before training. (b) Scattering plot for predicted model. (c) Confusion matrix for predicted model.

Fig. 15. Standard deviation of the sixth subject data.

gesture was separated and gathered for the twenty subjects, but this time they were given labels from 1 to 20 to examine the system ability to recognize the gesture from others. The same procedure was applied for the rest nine gestures. The results are shown in Table VI.

Figure 20 reveals the robustness of the suggested protocol in recognizing the ten gestures used in this study. Only the LDA classifier can be excluded from the results.

D. Merge All Proposed Features in the Case of 125 ms Window Size

To evaluate the system validation in the case of changing window size, this experiment was implemented, where the data was segmented into non-overlapping windows for each of 256 samples (125 ms), then the five features were extracted, and finally, data was split into 80% training and 20% test data. Table VII illustrates the performance accuracy achieved.

Figure 21 shows that all classifiers are relatively close in

their results unlike the previous experiment, where the LDA classifier was far off the other three classifiers.

Comparing the results of the previous experiment and those of this experiment, it is clear that the system accuracy is relatively less when the segmented window is 125 ms, but it still gives remarkable results. Figure 22 represents the confusion matrix for the sixth subject for both the last experiments (62.5 and 125 ms window size) with the SVM classifier which recorded the highest accuracy. The accuracy was 99.4% for 62.5 ms windowing and 99.1% for 125 ms windowing.

IV. COMPARATIVE ANALYSIS

In comparison with existing results, first, we'll compare results achieved from the same dataset we used in this work by other researchers when the suggested systems were designed based on the EMG signals, in [13] M. Montazerin et al used the whole 65 isometric hand gestures of the 20 subjects. The proposed CT-HGR framework was applied to 31.25, 62.5, 125,
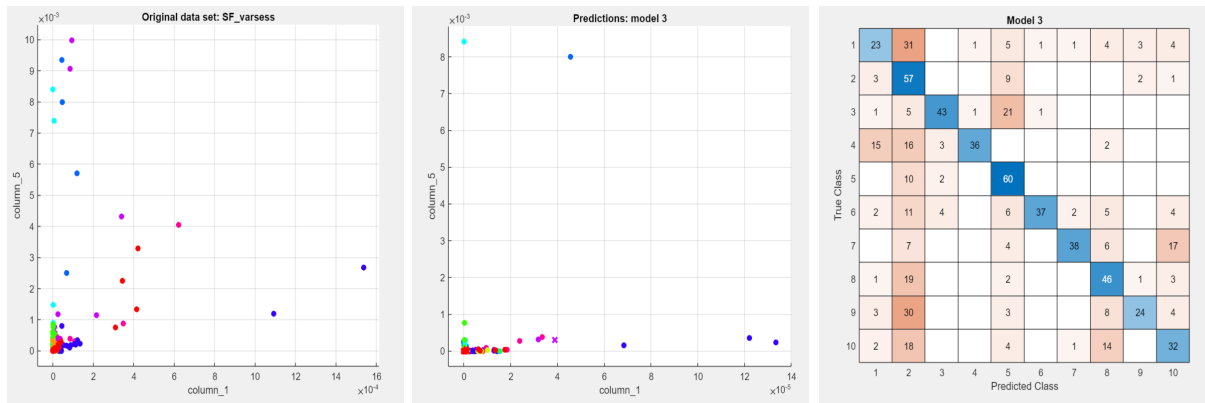
(a) Scattering plot before training. (b) Scattering plot for predicted model. (c) Confusion matrix for predicted model.

Fig. 16. Variance of the sixth subject data.

TABLE V.
THE ACCURACY OF TWENTY SUBJECTS FOR THE THIRD EXPERIMENT.

Subjects	Classification Accuracy (%)			
	DT	LDA	SVM	KNN
Sub 1	99.3	97.2	97.9	98.7
Sub 2	99.2	98.7	99.6	99.3
Sub 3	98.9	94.4	99.7	98.7
Sub 4	99.9	98.9	99.7	99.2
Sub 5	96.2	93.6	98.9	98.6
Sub 6	99.3	97.2	98.3	98.9
Sub 7	99	98.9	99.6	99.6
Sub 8	99.8	98.5	100	99.9
Sub 9	99.7	96.8	99.7	99.6
Sub 10	99.7	99.9	99.9	99.6
Sub 11	99.3	99.6	100	99.7
Sub 12	99.3	96.7	98.9	98.5
Sub 13	98.6	92.5	98.3	98.1
Sub 14	99.3	99.2	99.7	99.2
Sub 15	99.3	98.2	99.6	99
Sub 16	97.9	90.7	94.7	97.7
Sub 17	99.7	97.8	99.9	99.7
Sub 18	99.6	99.9	99.7	99.4
Sub 19	98.9	97.6	99.2	98.7
Sub 20	98.9	97.1	98.6	99.6
Average	99.1	97.2	99.1	99.1

TABLE VI.
THE ACCURACY OF THE TEN GESTURES.

Gestures	Classification Accuracy (%)			
	DT	LDA	SVM	KNN
G 1	98.3	94.7	99.2	98.8
G 2	98.6	96.1	99.5	98.9
G 3	98.5	95	99.4	98.5
G 4	98.8	98.2	99.4	99.2
G 5	99.4	97.2	99.5	98.5
G 6	98.3	95.8	99.8	99.2
G 7	98.3	96.5	99.5	99.1
G 8	98.5	96.7	99.7	98.8
G 9	96.7	94.7	97.9	98.6
G 10	97.8	98	99.4	98.6
Average	98.3	96.3	99.3	98.8

and 250 ms window sizes of the EMG dataset utilizing 32, 64, and 128 electrode channels. The average accuracy over all the participants using 32 electrodes and a window size of 31.25 ms was 86.23%, which gradually increased till reaching 91.98% for 128 electrodes and a window size of 250 ms. The CT-HGR achieves an accuracy of 89.13% for instantaneous recognition based on a single frame of HD-sEMG image. Also, [22] Khushaba and Nazarpour used different window sizes (from 32 ms to 256 ms) with 128 (8 and 65 hand movements) to 256 electrodes (34 hand movements) to explore the impact of varying the number of electrodes and segmentation windows size on EMG decoding accuracy. The analysis considered varying windows and electrodes from 8 to 128/256. Simple time-domain and auto-regressive model parameters were extracted to train an LDA classifier to identify the intended hand motions. The recorded accuracy reached 90%. Another existing study used the same EMG dataset [29], where Azar
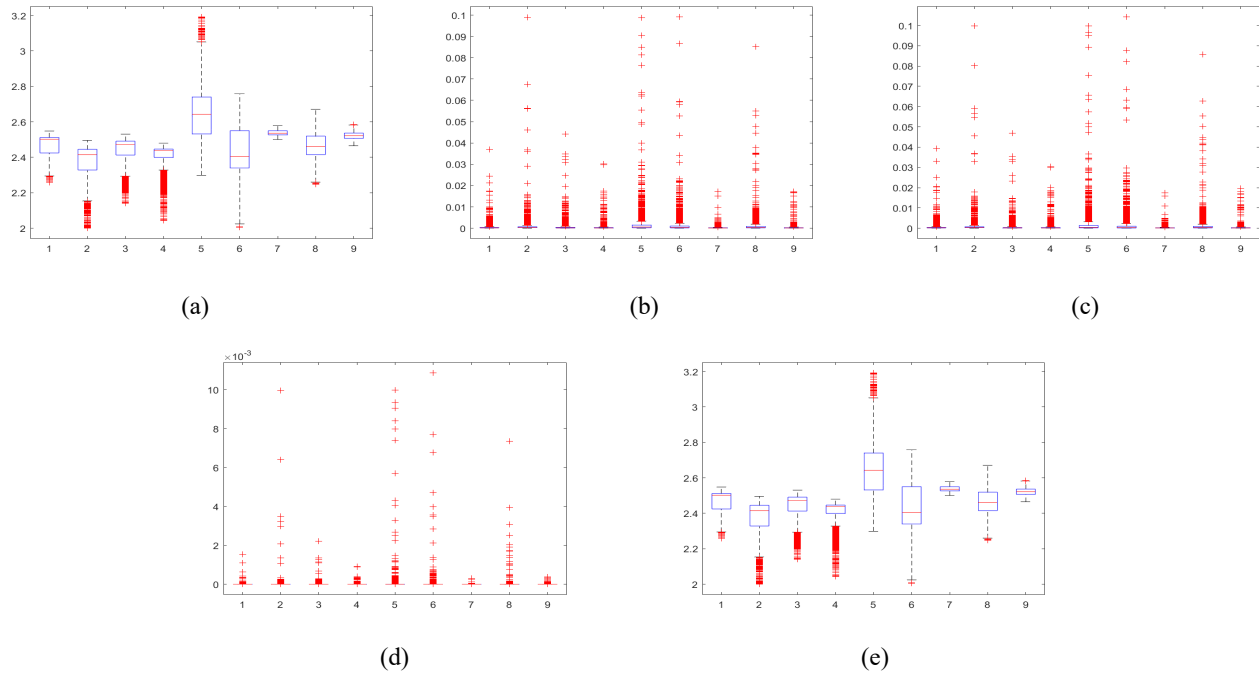
Fig. 17. The boxplot of the five features: (a) RMS (b) MAD (c) STD (d) VAR (e)MEAN.
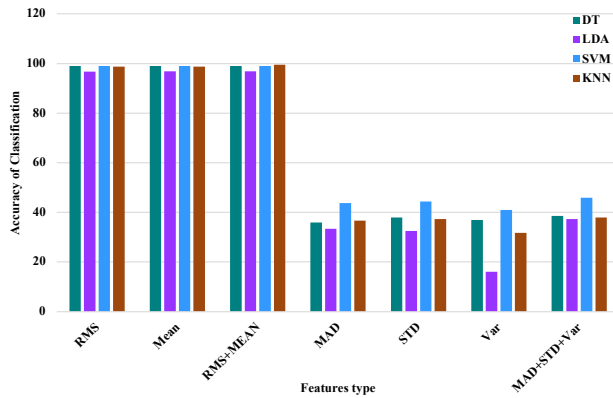

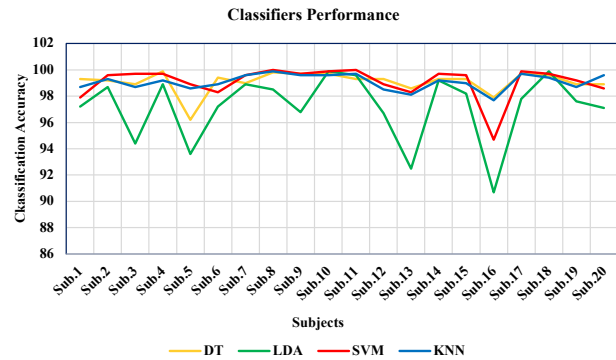Fig. 18. Comparison of single and combined features.

Fig. 19. The chart of experiment no. (3) performance.

et al. proposed a sequential decoder of transient HD-sEMG that achieved 73% average accuracy of all the 65 gestures for partially-observed subjects through subject-embedded transfer learning, leveraging pre-knowledge of HGR acquired during pre-training. The mentioned notes can be seen clearly in Table VIII.

Next, the effectiveness of the suggested machine learning-based hand movement recognition system will be evaluated against two previously published studies. In [30] the authors used the force signals with the KNN classifier and tried dif-ferent values for k (from1 to 9). The highest accuracy was achieved when k=2 and it was 95%. The FMG was collected from 8 subjects with ten trials per gesture. In [8] the classification of hand gestures based on FMG signals was performed. The best classification with 95% accuracy, was achieved through Decision Tree Learning and SVM classifiers. The best classification with 95% accuracy, was achieved through Decision Tree Learning and SVM classifiers. These comparison results can be seen in Table IX.
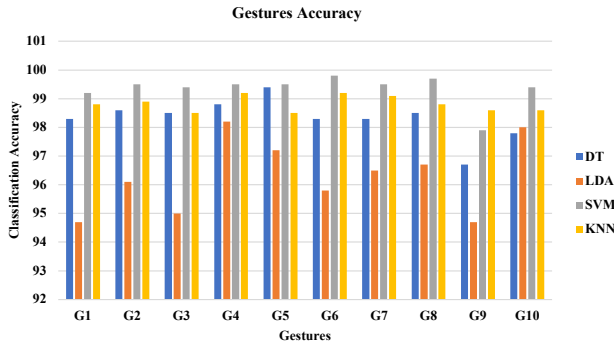
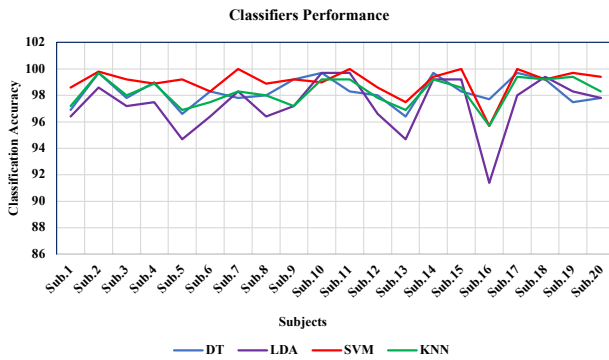Fig. 20. The average accuracy for the ten gestures.

Fig. 21. The chart of experiment no. (5) performance.

V. Conclusion

In this paper, a hand gestures classifier system has been proposed to improve classification accuracy. Two myograpy signal types have been used (FMG and HD-sEMG). This system depends on proposed new features that result from merging RMS, MEAN, MAD, STD, and VAR, in which all these features have been validated separately and performance evaluated for each one. The results show RMS and MEAN are superior to other features. In addition, the effect of segmented window size has been tested on the classification accuracy, in which the experiments led to the best size is 62.5 ms despite the difference being low versus the size of 125 ms. Several classifiers have been proposed for this mission which are TD, LDA, SVM, and KNN. Several experiments have been performed to validate the proposed system, in which the SVM classifier is superior to other classifiers. The results of these experiments show the ability to use the proposed system in the prosthesis, furthermore, the proposed features and classifiers appear good classification accuracy reaching 99.1%. The only limitation of our work is that it wasn't implemented in real-time applications with real prosthetic hands to examine the validity of the proposed system. This can be the goal of

TABLE VII.
THE ACCURACY OF TWENTY SUBJECTS FOR THE FIFTH EXPERIMENT.

Subjects	Classification Accuracy (%)			
	DT	LDA	SVM	KNN
Sub 1	96.9	96.4	98.6	97.2
Sub 2	99.7	98.9	99.8	99.3
Sub 3	97.8	97.2	99.2	98.0
Sub 4	99.0	97.5	98.9	98.9
Sub 5	96.6	94.7	99.2	96.9
Sub 6	98.3	96.4	98.3	97.5
Sub 7	97.8	98.3	99.8	98.3
Sub 8	98.0	96.4	98.9	98.0
Sub 9	99.2	97.2	99.2	97.2
Sub 10	99.7	99.7	99.0	99.2
Sub 11	98.3	99.7	100.0	99.2
Sub 12	98.0	96.6	98.6	97.8
Sub 13	96.4	94.7	97.5	96.9
Sub 14	99.7	99.2	99.4	99.2
Sub 15	98.3	99.2	100.0	98.6
Sub 16	97.7	91.4	95.7	95.7
Sub 17	99.7	98.0	100.0	99.4
Sub 18	99.2	99.4	99.2	99.2
Sub 19	97.5	98.3	99.7	99.4
Sub 20	97.8	97.8	99.4	98.3
Average	98.3	97.4	98.9	98.2

the future works.

Conflict of Interest

The authors have no conflict of relevant interest to this article.

References

[1] B. Yuan, D. Hu, S. Gu, S. Xiao, and F. Song, "The global burden of traumatic amputation in 204 countries and territories," *Frontiers in public health*, vol. 11, p. 1258853, 2023.

[2] R. Z. Khan and N. A. Ibraheem, "Hand gesture recognition: a literature review," *International journal of artificial Intelligence & Applications*, vol. 3, no. 4, p. 161, 2012.

[3] P. Premaratne and P. Premaratne, "Historical development of hand gesture recognition," *human computer interaction using hand gestures*, pp. 5–29, 2014.

[4] J. D. Nguyen and H. Duong, "Anatomy, shoulder and upper limb, hand long flexor tendons and sheaths," *National library of medicine, National center for biotechnology information, StatPearls*, 2023.

TABLE VIII.
A COMPARISON WITH STUDIES USED THE EMG SIGNALS OF OUR DATASET.

Ref.	Method	Channels	Window size (ms)	Accuracy (%)
M. Montazerin et al. 2023 [13]	CT-HGR framework	32	31.25	86.23
		64	62.5	89.29
		128	125	91.35
		128	250	91.98
		32	32	81.39
R.Kushaba and K.Nazarpour 2021 [22]	5 TDF of the signal along with sixth-order autoregressive coefficients are extracted and given to an LDA model	128	32	91.5
		128	256	96.14
G.A. Azar et al. 2023 [29]	DL using TL with three-layer d-biLSTM, a classifier with fully connected layers and dropout, and an embedding layer that captures subject dependencies in the generalized model.	128	200 with step 10	73
Experiment C	5 TDF with SVM classifier	9	62.5	99.1
Experiment D	5 TDF with SVM classifier	9	125	98.9

TABLE IX.
A COMPARISON WITH STUDIES USED THE FMG SIGNALS

Ref.	Dataset	Method	Accuracy (%)
N. Ha et al. 2018 [8]	The authors used three Piezoelectric sensors along with (DAQ) system to capture the FMG signals	DT and SVM classifiers	95
M. Fora et al. 2021 [30]	The authors used eight nanocomposite CNT/PDMS pressure sensors simultaneously.	KNN classifier, K=2	95
		K=3	91.9
This work	This dataset [19]	62.5 window size and DT classifier	99.1
		62.5 window size and LDA classifier	97.2
		62.5 window size and SVM classifier	99.1
		62.5 window size and KNN classifier	99.1

(a) 62.5 ms window size (b) 125 ms window size
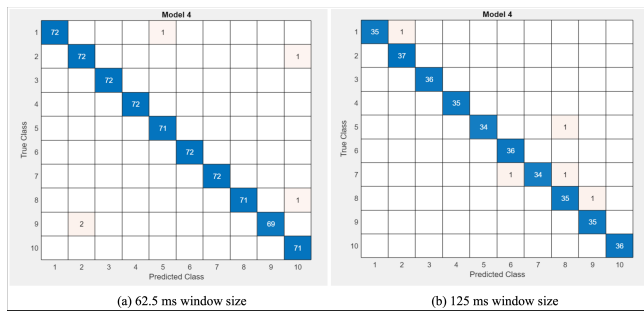
Fig. 22. Confusion matrix for the sixth subject.

[5] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng, "Surface emg-based inter-session gesture recognition enhanced by deep domain adaptation," *Sensors*, vol. 17, no. 3, p. 458, 2017.

[6] S.-W. Byun and S.-P. Lee, "Implementation of hand gesture recognition device applicable to smart watch based on flexible epidermal tactile sensor array," *Micromachines*, vol. 10, no. 10, p. 692, 2019.

[7] K. A. Abbas and M. T. Rashid, "Descriptive statistical features-based improvement of hand gesture identification," *Biomedical Signal Processing and Control*, vol. 92, p. 106103, 2024.

[8] N. Ha, G. P. Withanachchi, and Y. Yihun, "Force myography signal-based hand gesture classification for the implementation of real-time control system to a prosthetic hand," in *Frontiers in Biomedical Devices*, vol. 40789, p. V001T10A013, American Society of Mechanical Engineers, 2018.

[9] R. Ma, Z. Zhang, and E. Chen, "Human motion gesture recognition based on computer vision," *Complexity*, vol. 2021, no. 1, p. 6679746, 2021.

[10] H. Duan, C. Dai, and W. Chen, "The evaluation of classifier performance during fitting wrist and finger movement task based on forearm hd-semg," *Mathematical Problems in Engineering*, vol. 2022, no. 1, p. 9594521, 2022.

[11] H. Güneş and A. E. Akkaya, "Using wavelet analysis and deep learning for emg-based hand movement signal classification," *Sakarya University Journal of Science*, vol. 27, no. 1, pp. 214–225, 2023.

[12] H. A. Jaber, M. T. Rashid, H. Mahmood, and L. Fortuna, "Incremental adaptive gesture classifier for upper limb prostheses," *IEEE Sensors Journal*, vol. 22, no. 14, pp. 14273–14283, 2022.

[13] M. Montazerin, E. Rahimian, F. Naderkhani, S. F. Atashzar, S. Yanushkevich, and A. Mohammadi, "Transformer-based hand gesture recognition from instantaneous to fused neural decomposition of high-density emg signals," *Scientific reports*, vol. 13, no. 1, p. 11000, 2023.

[14] H. A. Jaber, M. T. Rashid, and L. Fortuna, "Adaptive myoelectric pattern recognition based on hybrid spatial features of hd-semg signals," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 45, no. 1, pp. 183–194, 2021.

[15] M. Jabbari, R. Khushaba, and K. Nazarpour, "Spatio-temporal warping for myoelectric control: an offline, feasibility study," *Journal of Neural Engineering*, vol. 18, no. 6, p. 066028, 2021.

[16] A. S. Khattak, A. b. M. Zain, R. B. Hassan, F. Nazar, M. Haris, and B. A. Ahmed, "Hand gesture recognition with deep residual network using semg signal," *Biomedical Engineering/Biomedizinische Technik*, vol. 69, no. 3, pp. 275–291, 2024.

[17] S. M. Massa, D. Riboni, and K. Nazarpour, "Graph neural networks for hd emg-based movement intention recognition: An initial investigation," in *2022 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*, pp. 1–4, IEEE, 2022.

[18] N. Malešević, A. Olsson, P. Sager, E. Andersson, C. Cipriani, M. Controzzi, A. Björkman, and C. Antfolk, "A database of high-density surface electromyogram signals comprising 65 isometric hand gestures," *Scientific Data*, vol. 8, no. 1, p. 63, 2021.

[19] N. Malešević, G. Andersson, A. Björkman, M. Controzzi, C. Cipriani, and C. Antfolk, "Instrumented platform for assessment of isometric hand muscles contractions," *Measurement Science and Technology*, vol. 30, no. 6, p. 065701, 2019.

[20] O. W. Samuel, M. G. Asogbon, Y. Geng, A. H. Al-Timemy, S. Pirbhulal, N. Ji, S. Chen, P. Fang, and G. Li, "Intelligent emg pattern recognition control method for upper-limb multifunctional prostheses: advances, current challenges, and future prospects," *Ieee Access*, vol. 7, pp. 10150–10165, 2019.

[21] F. D. Farfán, J. C. Politti, and C. J. Felice, "Evaluation of emg processing techniques using information theory," *Biomedical engineering online*, vol. 9, pp. 1–18, 2010.

[22] R. N. Khushaba and K. Nazarpour, "Decoding hd-emg signals for myoelectric control-how small can the analysis window size be?," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8569–8574, 2021.

[23] I. Bulugu, Z. Ye, and J. Banzi, "Higher-order local autocorrelation feature extraction methodology for hand gestures recognition," in *2017 2nd International Conference on Multimedia and Image Processing (ICMIP)*, pp. 83–87, IEEE, 2017.

[24] H. A. Jaber, M. T. Rashid, and L. Fortuna, "Using the robust high density-surface electromyography features for real-time hand gestures classification," in *IOP Conference Series: Materials Science and Engineering*, vol. 745, p. 012020, IOP Publishing, 2020.

[25] H. A. Jaber, M. T. Rashid, and L. Fortuna, "Interactive real-time control system for the artificial hand.," *Iraqi Journal for Electrical & Electronic Engineering*, vol. 16, no. 1, 2020.

[26] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.

[27] S. M. Massa, D. Riboni, and K. Nazarpour, "Explainable ai-powered graph neural networks for hd emg-based gesture intention recognition," *IEEE Transactions on Consumer Electronics*, 2023.

[28] J. C. Obi, "A comparative study of several classification metrics and their performances on data," *World Journal of Advanced Engineering Technology and Sciences*, vol. 8, no. 1, pp. 308–314, 2023.

[29] G. A. Azar, Q. Hu, M. Emami, A. Fletcher, S. Rangan, and S. F. Atashzar, "A deep learning sequential decoder for transient high-density electromyography in hand gesture recognition using subject-embedded transfer learning," *IEEE Sensors Journal*, 2024.

[30] M. Fora, B. B. Atitallah, K. Lweesy, and O. Kanoun, "Hand gesture recognition based on force myography measurements using knn classifier," in *2021 18th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 960–964, IEEE, 2021.