

# Time-domain feature and ensemble model based classification of EMG signals for hand gesture recognition

the date of receipt and acceptance should be inserted later

**Abstract** The electromyography (EMG) signal is the electrical current generated in muscles due to the interchange of ions during their contractions. It has many applications in clinical diagnostics and the biomedical field. This paper has experimented with various ensemble algorithms and time-domain features to classify eight types of hand gestures. To train and test the machine learning models, we have extracted eight types of time-domain features from the raw EMG signals, such as integrated EMG (IEMG), variance, mean absolute value (MAV), modified mean absolute value type 1, waveform length, root mean square, average amplitude change, and difference absolute standard deviation value. The ensemble machine learning models are based on stacking, bagging, and gradient boosting. We have used four different-sized training sets to evaluate the performance of these classifiers. From the performance evaluation, we have identified the XG-Boost (gblinear) classifier with the IEMG feature as the best classifier-feature pair. The proposed classifier-feature pair has given better performance with a classification accuracy of 98.33% and a processing time of  $5.67 \mu\text{s}$  for one vector than the existing extended associative memory-MAV classifier-feature pair.

**Keywords** Electromyography · Prosthesis · Myoelectric Signal Processing · Time-Domain features · Ensemble Learning · Pattern Recognition

## 1 Introduction

A prosthetic arm is an artificial device that replaces a missing arm. The goal of the prosthesis is to reinstate the functionalities of the missing body part as much as

Address(es) of author(s) should be given

possible to help the arm amputees to maintain a physically active lifestyle. There are significant numbers of research that have used surface EMG signals for prosthetic arms.

Electromyography (EMG) signal measures the electrical current generated in a muscle due to its contraction. The muscle contractions stimulate electrical activities through adjacent tissues and bones. It is recorded noninvasively by placing surface electrodes on the skin (surface EMG or sEMG) or planting needle electrodes into the muscle tissues (needle EMG or nEMG) (Wu et al. 2013). The sEMG is preferred to the nEMG because it is painless, covers motor units of a larger area, and allows simultaneous record-keeping of muscular activities from multiple places. EMG signals have vast applications to medical technologies. It is widely used in the clinical diagnosis of various neurological disorders. The recorded motor unit action potentials (MUAPs) for a duration is used for clinical diagnosis of muscle disorders like loss of muscle fibers (Liguori et al. 1997), chronic neuropathic disorders and polymyositis (Izzo and Aravabhum 1990; Uncini et al. 1990), rapidly progressing diseases of motor neurons and chronic myositis (Daube and Rubin 2009). These signals are also used in developing biomechatronic prosthetic limbs (Sheibani and Pourmina 2016) and hybrid assistive limbs (George et al. 2011). EMG signals are used in assessing diabetic peripheral neuropathy (DPN) disorder (Ullah and Iqbal 2020; Rabbi et al. 2019; Perkins and Bril 2014) and the development of foot ulcers in diabetic patients (Rabbit et al. 2019).

EMG signals are at the core of designing a perfectly responsive prosthetic arm. We have reviewed several recent empirical studies about EMG signal acquisition, processing, and classification. Leobardo E. Sánchez-Velasco et al. (2020) used an open-source six-degree-of-

freedom prosthetic arm (Krausz et al. 2015), a master-slave system, and a Myo armband. The master device gathered and classified EMG signals generated from muscle activities and received by a Myo armband. The slave device controlled the movements of the prosthetic arm.

Humberto Sossa et al. (2004) proposed an associative model for the classification of real-valued features. The proposed classifier performed better with the arithmetic average than the median operator. Angkoon Phinyomark et al. (2012) discussed several feature extraction techniques for EMG signals in the frequency domain and time-domain. The feature extraction techniques were classified into four categories based on mathematical property and information such as energy and complexity, time dependence, prediction model, and frequency. Frequency domain features were computed according to the statistical parameters of the EMG power spectral density. It was found that they were not very useful for classifying EMG signals. Lastly, Hudgins' feature set and Du's feature set were analyzed to produce an appropriate feature set free of redundant features. Hyun-Chul Kim et al. (2002) proposed Support Vector Machine (SVM) ensembles with bagging (bootstrap aggregating) for classification and regression tasks. Each SVM was trained independently with a randomly resampled training set using a bootstrap method. The outputs of the models were combined to produce the final decision. The proposed SVM ensemble outdid a single SVM by achieving much higher classification accuracies. The proposed ensemble was extended to the multiclass classification problems. Rodrigo E. Russo et al. (2018) used a MyoWare Muscle sensor to capture myoelectric signals (EMG) for three types of hand movements. Four types of time-domain features like MAV, slope sign change, zero-crossing, and waveform length were considered. Rodrigo E. Russo et al. used classifiers like SVM, K-NN, and Neural Networks. The SVM achieved the highest test accuracy.

Nagaswathi Amamcherla et al. (2018) proposed a model to classify ten types of hand gestures using EMG signals. The model used four statistical feature modalities like time-domain statistics and Hjorth features, spectral statistics, and autoregression coefficients. The sequential forward selection algorithm was used to discard the irrelevant features. A multiclass SVM was trained with the selected features and achieved a classification accuracy of 92.3%.

Piotr Kaczmarek et al. (2019) built a putEMG signal dataset for the classification of hand gestures. The dataset was recorded with 44 able-bodied subjects. The dataset contained eight gestures - four pinches, idle, and three full hand gestures. 90% accuracy was achieved us-

ing SVM classifier and LDA classifier with RMS feature, and Hudgin's or Du's feature set, respectively. Further scrutiny revealed that LDA/Du combination performed better for full hand gestures, whereas SVM/RMS was more suitable for pinch gestures.

The contributions of this paper are as follows:

1. Six types of ensemble algorithms and eight types of time-domain features are used to classify eight types of hand gestures with a public EMG signal dataset.
2. Training sets of four different sizes are used to train the ensemble classifiers. The smallest size of the training set on which the classifiers achieve a satisfactory classification accuracy is finally considered.
3. A combination of an ensemble-based algorithm and a time-domain feature is picked as the best classifier-feature pair in terms of adequately high classification accuracy and the lowest processing time.
4. Confusion matrix and other evaluation metrics are computed for the proposed classifier-feature pair to evaluate the model performance.
5. The performance of the proposed classifier-feature pair is compared to that of an existing one in terms of classification accuracy, true positive rate, true negative rate, false positive rate, false negative rate, and processing time.

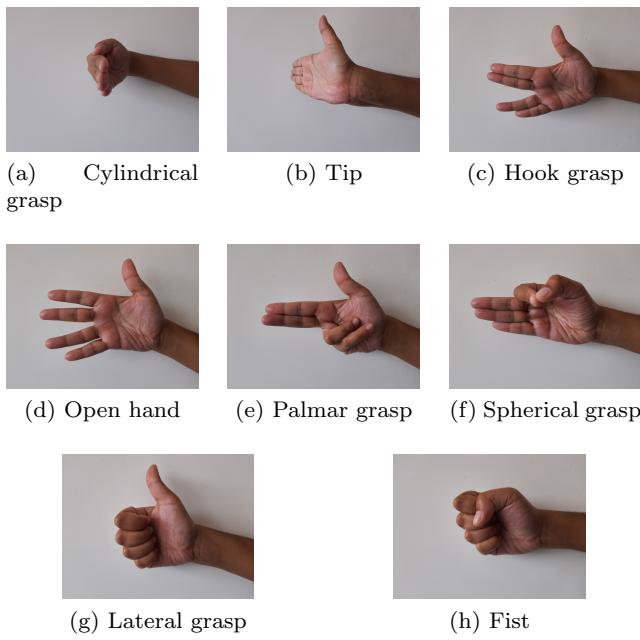
The rest of the paper is presented as follows: Section 2 gives insight into the publicly available dataset used in this paper. Section 3 discusses the various phases of the proposed algorithm. Section 4 discusses the results of the proposed classifier-feature pair and compares it to an existing one. Lastly, Section 5 concludes the paper.

## 2 The dataset

We have used EMG data of the Myo Armband (Sánchez-Velasco et al. 2019), a public dataset, to train and test our classification models. The dataset contains EMG signals caused by eight types of hand gesture recorded with a Myo Armband, as shown in Fig. 1: (a) cylindrical grasp, (b) tip grasp, (c) hook grasp, (d) open hand, (e) palmar grasp, (f) spherical grasp, (g) lateral grasp, and, (h) fist. There are eight files for eight gestures in the dataset. Every file contains 50 readings recorded by a Myo armband which incorporates eight sensors. Each sensor produces 100 samples for a reading. The sampling frequency of the EMG signal is 50 samples per second.

## 3 Proposed algorithm

The proposed algorithm is divided into three phases as shown in Fig. 3:



**Fig. 1** Eight types of hand gestures

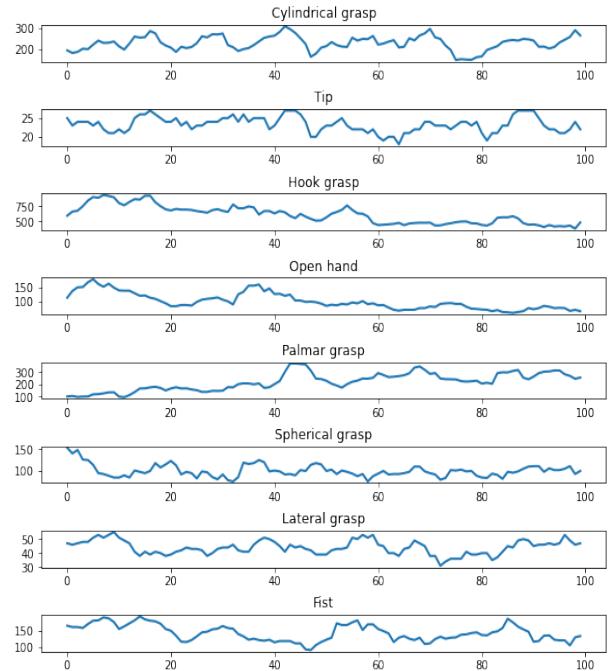
1. **Signal segmentation:** We have segmented the signals in this phase.
2. **Feature extraction:** In this phase, we have extracted eight types of time-domain features from every signal segment.
3. **Classification:** In this phase, we have split the extracted features into training and test features. We have used training features to train six types of ensemble models and test features to evaluate the trained models.

### 3.1 Signal segmentation

In the signal segmentation phase, we have divided the EMG signals into fixed-size segments. We have selected the segment size as 100 samples, as shown in Fig. 1 and Fig. 2, for each gesture. To understand the underlying properties of the signals, we have computed features for every segment. In the next section, we have discussed the feature extraction phase.

### 3.2 Feature extraction

The time-domain features are computationally simple. We have extracted eight types of time-domain features from the signals captured by the eight sensors. For eight categories of hand gestures, we have selected eight types of features. Fig. 4 shows the scatter plots for different features. The extracted features are as follows:



**Fig. 2** Signal segments recorded from the first sensor for different gestures

#### 3.2.1 Integrated EMG

Integrated EMG (IEMG) is used to detect muscle activity. It is the sum of absolute values of the EMG signal amplitudes in a segment. It is expressed as,

$$IEMG = \sum_{i=1}^N |x_i|, \quad (1)$$

where  $N$  denotes the number of samples in a segment and  $x_i$  represents the  $i$ -th sample in the segment.

#### 3.2.2 Mean absolute value

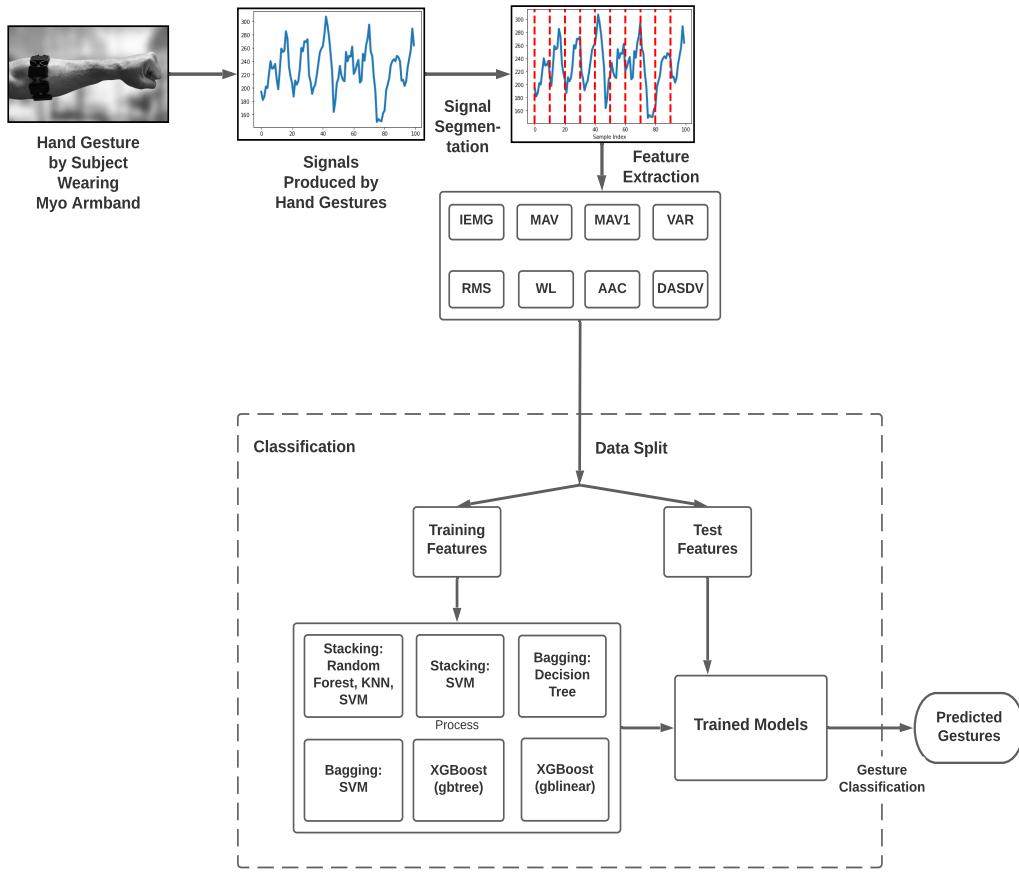
Mean absolute value (MAV) like IEMG is used to detect muscle activities. It is the average of absolute values of the EMG signal amplitudes in a segment. It is defined as,

$$MAV = \frac{1}{N} \sum_{i=1}^N |x_i|, \quad (2)$$

where  $N$  and  $x_i$  are same as before.

#### 3.2.3 Modified mean absolute value type 1

Modified mean absolute value type 1 (MAV1) is a weighted MAV feature. The weighted window function



**Fig. 3** Proposed algorithm

$w_i$  increases the robustness of the MAV feature. The MAV1 feature and the weights are calculated as follows,

$$MAV1 = \frac{1}{N} \sum_{i=1}^N w_i |x_i|, \quad (3)$$

where  $N$ ,  $x_i$  are same as before and

$$w_i = \begin{cases} 1; & \text{if } 0.25N \leq i \leq 0.75N \\ 0.5; & \text{otherwise} \end{cases}$$

### 3.2.4 Variance of EMG

Variance (VAR) records the power of an EMG signal. As the mean of an EMG signal is negligible ( $\sim 10^{-10}$ ), the variance is expressed as,

$$VAR = \frac{1}{N-1} \sum_{i=1}^N x_i^2, \quad (4)$$

where  $N$  and  $x_i$  are same as before.

### 3.2.5 Root mean square

Root mean square (RMS), like VAR, records the power of an EMG signal. It is defined as,

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}, \quad (5)$$

where  $N$  and  $x_i$  are same as before.

### 3.2.6 Waveform length

Waveform length (WL) is the cumulative length of the waveform over a segment. WL is defined as,

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad (6)$$

where  $N$  and  $x_i$  are same as before.

### 3.2.7 Average amplitude change

Average amplitude change (AAC) is WL averaged over the segment size. It is defined as,

$$AAC = \frac{1}{N} \sum_{i=1}^{N-1} |x_{i+1} - x_i|, \quad (7)$$

where  $N$  and  $x_i$  are same as before.

### 3.2.8 Difference absolute standard deviation value

Difference absolute standard deviation value (DASDV) is expressed as,

$$DASDV = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (x_{i+1} - x_i)^2} \quad (8)$$

where  $N$  and  $x_i$  are same as before.

We have split each reading into eight segments. For every segment in a reading, we have computed a time-domain feature. We have computed eight features for a reading. They together have formed a feature vector. For instance, considering the IEMG feature, we have defined the feature vector as follows,

$$x = [IEMG^1, IEMG^2, \dots, IEMG^7, IEMG^8]$$

For 50 readings in a file, we have computed 50 feature vectors. We have repeated the entire procedure for all of the eight files for the respective gestures. Total  $50 \times 8 = 400$  feature vectors for each feature have been computed. After that, we have randomly divided the entire data into a training set and a test set.

### 3.2.9 Classification models

We have experimented with different kinds of ensemble models. The ensemble method combines several Machine Learning algorithms to decrease bias, variance or improve the model's prediction ability.

**a) Stacking:** Stacking (Wolpert 1992) is a learning technique in which a meta model combines different machine learning algorithms. In the first level, the different machine learning algorithms are used as base models. Each of the algorithm are trained individually on the original training set. In the second level, the machine learning algorithm is used as a meta model. A new training set is formed for the meta model which consists of outputs of all of the base models considered as features and the original labels. For each sample  $\{x_i, y_i\}$  in original training dataset  $D$ , a new dataset  $\{x'_i, y'_i\}$  is formed for the meta model where  $x'_i = \{h_1(x_i), h_2(x_i), \dots, h_T(x_i)\}$ , here  $\{h_1, h_2, \dots, h_T\}$  are the first level models. The second level model is trained with the newly formed dataset. For an unseen sample  $x$ , the prediction of the stacking model is  $h' \{h_1(x), h_2(x), \dots, h_T(x)\}$

where  $\{h_1, h_2, \dots, h_T\}$  are described as before and  $h'$  is the second level classifier.

We have used two types of stacking algorithms. In the first type, we have used Random forest and K nearest neighbours (KNN) as the first level classifiers and a Support vector machine (SVM) as the second level classifier. For the second type, we have used three different SVMs, tuned for different hyperparameters. We have used two of them as the first level classifiers and the remaining one as the second level classifier. Probabilities predicted by the first-level classifiers are used as features for the second-level classifier.

**b) Bagging:** Bagging or bootstrap aggregation (Breiman 1996) is a learning technique that reduces the variance of a model. So it works well on models with low bias and high variance. In bootstrapping phase, a training dataset is resampled iteratively with replacement to train the base estimators. Features are randomly selected with or without replacement. The individual predictions of all base estimators are aggregated with a meta-estimator by majority voting (for classification) or averaging (for regression).

We have used majority voting to determine the final prediction of the meta-model for our classification task. We have experimented with bagging models built on decision trees and SVMs.

**c) Gradient Boosting:** Gradient boosting (Friedman 2001) algorithm produces a prediction model as a combination of several weak prediction models.

Initially the model is trained with a training set  $\{x_i, y_i\}_{i=1}^n$  by minimizing a differentiable loss function  $L(y, F(x))$ , and, is initialized with a constant value,

$$F_0(x) = \arg \min_{\gamma} \sum_{i=1}^n L(y_i, \gamma)$$

From iteration 1 to  $M$ , a pseudo-residual is computed in every iteration as follows,

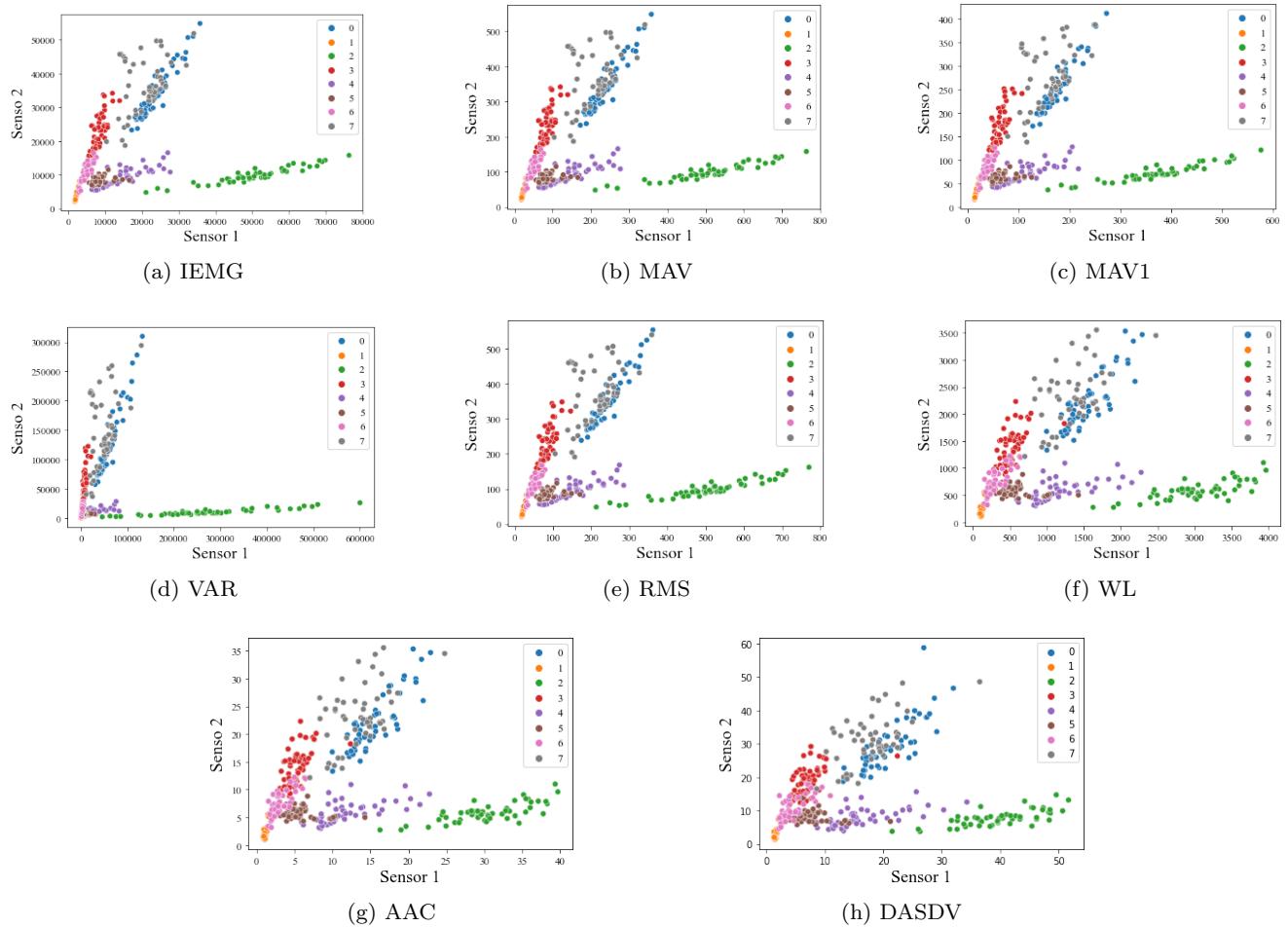
$$r_{i,m} = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}$$

for  $i = 1, 2, \dots, n$ ;  $m$  is the iteration number and  $n$  is the number of samples in  $y$ . Next, a base learner  $h_m$  is trained with the training set  $\{x_i, r_{i,m}\}_{i=1}^n$ . Multiplier  $\gamma_m$  is computed such that,

$$\gamma_m = \arg \min_{\gamma} \sum_{i=1}^n L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i))$$

The formula for model updation is given below,

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$$



**Fig. 4** Scatter plots for different features for data from first two sensors for eight categories of hand gestures

This process continues for  $M$  number of iterations. The final model  $F_M$  is the output of the  $M$ th iteration.

We have used the XGBoost library to implement gradient boosting using gbtree and gblinear as the boosters.

We have generated candidates from a parameter grid for an N-fold cross-validated grid search for each classifier to find the optimal hyperparameters.

## 4 Results

For experimental validation, we have used machine configurations as 64-bit Ubuntu 16.04 LTS operating system, Intel Core i3-5005U CPU @ 2.00GHz × 4, 8 GB of RAM, 611 GB of disk space. The experimental evaluation metrics consist of the following.

- Accuracy is the fraction of total predictions the classifier predicts correctly. It is defined as follows,

$$\text{accuracy} = \frac{\text{number of correct predictions}}{\text{total number of predictions}}$$

- A true positive ( $TP$ ) outcome is a correctly predicted positive class.
- A true negative ( $TN$ ) outcome is a correctly predicted negative class.
- A false positive ( $FP$ ) outcome is a mispredicted positive class.
- A false negative ( $FN$ ) outcome is a mispredicted negative class.
- True positive rate or sensitivity or recall for the  $j$  – th gesture ( $TPR_j$ ), is defined as follows,

$$TPR_j = \frac{TP_j}{TP_j + FN_j}$$

where  $TP_j$ ,  $FP_j$  are  $TP$  and  $FN$  for the  $j$  – th gesture respectively.

- True negative rate or specificity or selectivity for the  $j$  – th gesture ( $TNR_j$ ), is defined as follows,

$$TNR_j = \frac{TN_j}{TN_j + FP_j}$$

where  $TN_j$ ,  $FP_j$  are  $TN$  and  $FP$  for the  $j - th$  gesture respectively.

- False positive rate for the  $j - th$  gesture ( $FPR_j$ ), is defined as follows,

$$FPR_j = \frac{FP_j}{FP_j + TN_j}$$

where  $TN_j$ ,  $FP_j$  are  $TN$  and  $FP$  for the  $j - th$  gesture respectively.

- False negative rate for the  $j - th$  gesture ( $FNR_j$ ), is defined as follows,

$$FNR_j = \frac{FN_j}{FN_j + TP_j}$$

where  $FN_j$ ,  $TP_j$  are  $FN$  and  $TP$  for the  $j - th$  gesture respectively.

- We have computed the confusion matrix to evaluate the performance of the proposed classification model.

#### 4.1 Accuracies and processing times for different classifiers on different features for different sizes of training sets and test sets

We have trained the classifiers with training sets of the following sizes:

1. 70% of total samples (35 samples for each gesture): The test set consists of 15 samples for each gesture. XGBoost with gblinear booster (XGBoost (gblinear)) has achieved 100% test accuracy on IEMG, MAV, MAV1 and RMS features with processing times of 15.53  $\mu s$ , 26.06  $\mu s$ , 20.30  $\mu s$ , and 16.53  $\mu s$  respectively for one vector. Bagging SVM has achieved 100% test accuracies on MAV, MAV1, and RMS features with processing times of 83.46  $\mu s$ , 86.42  $\mu s$ , and 69.28  $\mu s$  respectively for one vector.
2. 60% of total samples (30 samples for each gesture): The test set consists of 20 samples for each gesture. XGBoost (gblinear) has achieved the highest 98.75% test accuracy on IEMG and MAV1 features with processing times of 12.54  $\mu s$  and 13.66  $\mu s$  respectively. Stacking SVM has also achieved 98.75% accuracy on MAV1 with a processing time of 682.44  $\mu s$ .
3. 50% of total samples (25 samples for each gesture): The test set consists of 25 samples for each gesture. XGBoost (gblinear) has produced the highest accuracy of 98% on the IEMG feature and taken 22.93  $\mu s$  to process one vector.

**Table 1** Test accuracies and processing times of the classifiers trained on training sets of different sizes and features

Training Set Size	Test Set Size	Classifier	Feature	Test Accuracy (%)	Processing time for one vector ( $\mu s$ )
35	15	XGBoost (gblinear)	IEMG	100	15.53
35	15	XGBoost (gblinear)	MAV	100	26.06
35	15	XGBoost (gblinear)	MAV1	100	20.30
35	15	XGBoost (gblinear)	RMS	100	16.53
35	15	Bagging SVM	MAV	100	83.46
35	15	Bagging SVM	MAV1	100	86.42
35	15	Bagging SVM	RMS	100	69.28
30	20	XGBoost (gblinear)	IEMG	98.75	12.54
30	20	XGBoost (gblinear)	MAV1	98.75	13.66
30	20	Stacking SVM	MAV1	98.75	682.44
25	25	XGBoost (gblinear)	IEMG	98	22.93
20	30	Bagging SVM	RMS	98.33	59.65
20	30	XGBoost (gblinear)	IEMG	98.33	5.67

4. 40% of total samples (20 samples for each gesture): The test set consists of 30 samples for each gesture. XGBoost (gblinear) has produced the highest accuracy of 98.33% on the IEMG feature with the lowest processing time of 5.67  $\mu s$  for one vector. Bagging SVM has also achieved the same accuracy on the RMS feature with a processing time of 59.65  $\mu s$  for one vector.

We have summed up all the findings in Table 1. For building a training set of a larger size, the subjects need to make more gestures; hence there are higher chances of being fatigued, losing concentration, and getting distracted, which may lead to incorrect results. In this experiment, we have considered an optimal-sized training set to achieve a reasonably accurate classification. The training set consists of 20 samples for each gesture. From the experimental evaluation with the training set, it is observed that the XGBoost (gblinear)-IEMG has emerged as the best classifier-feature pair with a 98.33% classification accuracy and processing time of 5.67  $\mu s$  for one vector. Table 2 lists the optimal values of various hyperparameters for the XGBoost (gblinear) classi-

fier for the IEMG feature produced by an N-fold cross-validated grid search.

**Table 2** Optimized hyperparameters for XGBoost (gblinear) on IEMG

Hyperparameter	Optimal Value
Booster	gblinear
Learning rate	0.5
No. of base estimators in the ensemble	100
L1 regularization term (alpha)	0.1
L2 regularization term (lambda)	0.1

## 4.2 Confusion matrix, evaluation metrics for XGBoost (gblinear) on IEMG feature

Table 3 and Table 4 present the confusion matrix and the evaluation metrics for the XGBoost (gblinear) classifier on the IEMG feature.

**Table 3** Confusion matrix for the XGBoost (gblinear) on IEMG

Original Class	Class Predicted							
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
(a)	30	0	0	0	0	0	0	0
(b)	0	30	0	0	0	0	0	0
(c)	0	0	30	0	0	0	0	0
(d)	0	0	0	30	0	0	0	0
(e)	0	0	0	0	30	0	0	0
(f)	0	0	0	0	4	26	0	0
(g)	0	0	0	0	0	0	30	0
(h)	0	0	0	0	0	0	0	30

**Table 4** Evaluation metrics for different gestures for XGBoost (gblinear) on IEMG

	Class							
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
TPR	1	1	1	1	1	0.866	1	1
TNR	1	1	1	1	0.981	1	1	1
FPR	0	0	0	0	0.019	0	0	0
FNR	0	0	0	0	0	0.133	0	0

The misclassification rate of XGBoost (gblinear) on IEMG = 1 - accuracy = 100 - 98.33% = 1.67%.

The true positive rate ( $TPR$ ) or sensitivity, the true negative rate ( $TNR$ ) or specificity, the false positive

rate ( $FPR$ ), and the false negative rate ( $FNR$ ) are calculated from Table 4 as follows,

$$TPR = \frac{TPR_{(a)} + TPR_{(b)} + \dots + TPR_{(h)}}{8} = 0.9833$$

$$TNR = \frac{TNR_{(a)} + TNR_{(b)} + \dots + TNR_{(h)}}{8} = 0.9976$$

$$FPR = \frac{FPR_{(a)} + FPR_{(b)} + \dots + FPR_{(h)}}{8} = 0.0024$$

$$FNR = \frac{FNR_{(a)} + FNR_{(b)} + \dots + FNR_{(h)}}{8} = 0.0166$$

We have compared the performance of the proposed XGBoost (gblinear)-IEMG classifier-feature pair with EAM-MAV (Sánchez-Velasco et al. 2020). The proposed classifier-feature pair has given better performance than the existing one in terms of classification accuracy,  $TPR$ ,  $TNR$ ,  $FPR$ ,  $FNR$ , and processing time, as shown in Table 5.

**Table 5** Comparison of the proposed XGBoost (gblinear)-IEMG classifier-feature pair with an existing one

Classifier + Feature	Test accuracy (%)	TPR	TNR	FPR	FNR	Processing time for all vectors (ms)	Processing time for one vector ( $\mu$ s)
XGBoost (gblinear) + IEMG	98.33	0.9833	0.9976	0.0024	0.0166	1.36	5.67
EAM + MAV (Sánchez-Velasco et al. 2020)	95.83	0.9583	0.994	0.0417	0.0417	1.445	6.022

## 5 Conclusion

In this paper, we have implemented the time-domain feature and ensemble model-based classification of EMG signals for hand gesture recognition. The time-domain features were extracted from raw EMG signals to create feature vectors for training and testing the six types of ensemble classifiers. We have trained and tested these classifiers on training sets and test sets of four different sizes. To reduce human error due to fatigue, we have considered the smallest size of the training set (40% of total samples or 20 samples for each gesture), which has led to a high classification accuracy with the lowest processing time for a quick responsive prosthetic arm. The XGBoost (gblinear) classifier with the IEMG feature has turned to be the best classifier-feature pair with a classification accuracy of 98.33% and a processing time of 5.67  $\mu$ s for one vector. The proposed classifier-feature pair has achieved better accuracy within a lower processing time than the existing EAM-MAV classifier-feature pair.

In future work, we will build our own dataset using EMG sensors for various types of clinical diagnoses.

## Declarations

**Conflict of interest** The authors declare that there is no conflict of interest regarding the publication of this paper.

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

**Informed consent** Informed consent was obtained from all individual participants included in the study.

## References

- Amamcherla N, Turlapaty A, Gokaraju B (2018) A machine learning system for classification of emg signals to assist exoskeleton performance. In: 2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), IEEE, pp 1–4
- Breiman L (1996) Bagging predictors. *Machine learning* 24(2):123–140
- Daube JR, Rubin DI (2009) Needle electromyography. *Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine* 39(2):244–270
- Friedman JH (2001) Greedy function approximation: a gradient boosting machine. *Annals of statistics* pp 1189–1232
- George T, Shalu G, Sivanandan K (2011) Sensing, processing and application of emg signals for hal (hybrid assistive limb). *IET*
- Izzo K, Aravabhum S (1990) Clinical electromyography. principles and practice. *Clinics in podiatric medicine and surgery* 7(1):179–194
- Kaczmarek P, Mańkowski T, Tomczyński J (2019) putemga surface electromyography hand gesture recognition dataset. *Sensors* 19(16):3548
- Kim HC, Pang S, Je HM, Kim D, Bang SY (2002) Support vector machine ensemble with bagging. In: International Workshop on Support Vector Machines, Springer, pp 397–408
- Krausz NE, Rorrer RA, et al. (2015) Design and fabrication of a six degree-of-freedom open source hand. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 24(5):562–572
- Liguori R, Fuglsang-Frederiksen A, Nix W, Fawcett P, Andersen K (1997) Electromyography in myopathy. *Neurophysiologie Clinique/Clinical Neurophysiology* 27(3):200–203
- Perkins B, Bril V (2014) Electrophysiologic testing in diabetic neuropathy. *Handbook of clinical neurology* 126:235–248
- Phinyomark A, Phukpattaranont P, Limsakul C (2012) Feature reduction and selection for EMG signal classification. *Expert systems with applications* 39(8):7420–7431

- Rabbi MF, Ghazali KH, Altwijri O, Alqahtani M, Rahaman SM, Ali MA, Sundaraj K, Taha Z, Ahamed NU (2019) Significance of electromyography in the assessment of diabetic neuropathy. *Journal of Mechanics in Medicine and Biology* 19(03):1930001
- Russo RE, Fernández J, Rivera R, Kuzman MG, López J, Gemin W, Revuelta MÁ (2018) Algorithm of myoelectric signals processing for the control of prosthetic robotic hands. *Journal of Computer Science & Technology* 18
- Sánchez-Velasco L, Arias-Montiel M, Guzmán-Ramírez E (2019) EMG data of the Myo Armband. Mendeley Data, V1, doi: 10.17632/rwbs7645hg.1
- Sánchez-Velasco LE, Arias-Montiel M, Guzmán-Ramírez E, Lugo-González E (2020) A low-cost EMG-controlled anthropomorphic robotic hand for power and precision grasp. *Biocybernetics and Biomedical Engineering* 40(1):221–237
- Sheibani A, Pourmina M (2016) Study and analysis of emg signal and its application in controlling the movement of a prosthetic limb. *Health and Technology* 6(4):277–284
- Sossa H, Barron R, Vázquez RA (2004) Real-valued pattern classification based on extended associative memory. In: Proceedings of the Fifth Mexican International Conference in Computer Science, 2004. ENC 2004., IEEE, pp 213–219
- Ullah S, Iqbal K (2020) A preliminary review on emg signals for assessment of diabetic peripheral neuropathy disorder. In: 2020 7th International Conference on Electrical and Electronics Engineering (ICEEE), IEEE, pp 42–46
- Uncini A, Lange DJ, Lovelace RE, Solomon M, Hays AP (1990) Long-duration polyphasic motor unit potentials in myopathies: a quantitative study with pathological correlation. *Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine* 13(3):263–267
- Wolpert DH (1992) Stacked generalization. *Neural networks* 5(2):241–259
- Wu Y, Martínez MÁM, Balaguer PO (2013) Overview of the application of emg recording in the diagnosis and approach of neurological disorders. *Electrodiagnosis in New Frontiers of Clinical Research* pp 1–24