JOURNAL OF
EMERGING INVESTIGATORS

Article

Feature extraction from peak detection algorithms for enhanced EMG-based hand gesture recognition models

Adikesh Nathan[1], Shanmuganathan Raju[1]
[1]Cambridge Center for International Research, Princeton Junction, New Jersey

SUMMARY

Achieving precise control of hand prosthetics is essential for enhancing the quality of life for individuals with arm amputations. Electromyography (EMG) signals are widely used as interfaces for these prosthetics. A key challenge in developing such interfaces is hand gesture recognition, where EMG signals are analyzed to predict the intended hand shape of the user. This study advances machine learning models for EMG-based hand gesture recognition by exploring the use of peak detection algorithms for feature extraction. We evaluated three distinct algorithms and a baseline without peak features with features derived from the detected peaks as well as other common methods. Our initial hypothesis was that peak features from a savitzky-golay filter would provide the highest accuracy because it uniquely smoothens the signal. Contrary to our hypothesis, however, when trained with a random forest model, the features from the wavelet-based peak detection algorithm achieved a higher classification accuracy than the maxima-based and savitzky-golay-based algorithms as well as the data without peak detection features when classifying two gestures (hand open/close). Further, the extracted peak features were among the most important features for all three algorithms. These results demonstrate that peak features, particularly those extracted using the best-performing, wavelet-based approach, can enhance the performance of hand gesture recognition models. This improvement could significantly benefit patients relying on prosthetic devices by enabling more accurate translation of their intended motions into device actions, ultimately improving their quality of life.

INTRODUCTION

Amputation is a profound and life-altering event affecting millions of people worldwide, often resulting from trauma, disease, or congenital conditions (1). For many of these individuals, prostheses become essential for restoring mobility and independence, and significant research has focused on developing effective interfacing tools for prostheses to improve their quality of life. One common area of study, specifically for arm amputees, is hand gesture recognition, where an interfacing tool detects the hand gesture a patient attempts to create (2). Developing advanced interfaces holds the promise of significantly enhancing the quality of life for millions of arm amputees, enabling them to perform everyday tasks with greater ease and precision.

Electromyography (EMG) is a standard measurement technique used for prosthetic control. EMG measures the electrical activity of specific muscle groups in the body (3). This activity is measured through the use of electrodes placed either on the surface skin of the subject, called surface electromyography (sEMG), or in the muscle itself via needle electrodes, called intramuscular electromyography (iEMG) (3). For prosthetic interfaces, sEMG data is widely used as it is easy to collect and requires little physical effort by the patient (4). However, sEMG signals are noisy and unreliable as skin electrodes may measure the electrical activity of multiple motor units at a time, causing disturbances between signals (4). While iEMG can yield better results direct access to muscle activity, collecting iEMG data can be uncomfortable for the subject and difficult to set up (5). For these reasons, sEMG has been more commonly used in literature studying EMG-based prosthetic control (2).

Over the past few years, many studies have been conducted with the goal of using ML to interpret sEMG data in the realm of prostheses and make predictions, specifically with hand gesture recognition (2). In 2008, a decision-level fusion of k-NN and Bayes linear classifiers and achieved a classification accuracy of 94% with different hand gestures (6). In 2009, mean and median frequencies (MMNF) for robust feature extraction in EMG data with white Gaussian noise were used, achieving 5-10% error with MMNF in a weak EMG signal with high noise (7). In 2020, a convolutional auto-encoder and a convolutional neural network (CAE+CNN) was used to classify a dataset containing 10 different hand gestures, achieving an accuracy of 98.13% with a Gaussian noise level of 1e-5 applied to the data (8). Ultimately, the common trend among these and other studies exploring the topic of EMG hand gesture recognition is that they aim to add or change the model or inputted features in developing the model to improve accuracy.

The specific application of our research is EMG-based hand gesture recognition using machine learning (ML). In this task, a patient generates electrical activity from intact motor units in the arm. EMG data is generated from these intact muscles and passed into a model that predicts what hand shape a subject is trying to make (2). A hand prosthetic would then create these gestures in real time. ML is well suited for EMG analysis as certain feature extraction and pre-processing methods integrated with machine learning models can alleviate the inherent noise in EMG data and better convey the muscle movement of a subject. Pre-processing techniques, such as Butterworth filtering, can minimize irrelevant frequencies using a bandpass filter, and specific feature extraction techniques can enhance the quality of data (9).

We aim to enhance EMG-based prosthesis control by experimenting with the use of features derived from peak de-

https://doi.org/10.59720/24-321

tection algorithms for EMG hand gesture recognition models. Specifically, this model will be a binary classification model that distinguishes between EMG signals that show hand open and hand close. These peaks are significant for EMG signals because they can represent the relevant contractions of a muscle, which may help an ML model to differentiate between hand gestures, as different gestures may involve different contractions at various times and magnitudes. While these detection algorithms have been studied in EMG signal analysis, they have yet to be extensively applied to hand gesture recognition (10). In addition, the model was tested without using any peak detection-derived features, providing a baseline for comparison.

We tested three distinct algorithms, with features extracted from each algorithm and passed into a ML model. A random forest classifier was used to analyze the features extracted from the peaks and other methods; the model works by combining the outputs of multiple simpler models, which allow the random forest to handle complex relationships between features.

Our initial hypothesis was that peak features derived from a savitzky-golay filter—a method that smooths data by fitting successive subsets of it to low-degree polynomials and then doing point by point analysis to determine peaks—would be the most effective instead of the wavelet-based or maxima-based algorithms. The filter's smoothing process can remove unwanted peaks caused by noise, potentially increasing class separability between gestures. Contrary to our hypothesis, however, features from the wavelet-based peak detection approach using Continuous Wavelet Transform (CWT), which analyzes signals at multiple scales using mathematical functions called wavelets, provided the highest accuracy. When trained with a random forest classifier, the features derived from this algorithm achieved the highest classification accuracy compared to accuracies of the maxima-based and savitzky-golay-based algorithms, as well as the accuracy without peak detection features when classifying two gestures (hand open/close). Further, the extracted peak features were among the most important for all algorithms evaluated, highlighting the class separability that these features provide. These findings demonstrate the critical role of advanced peak detection techniques in enhancing the performance of EMG-based gesture recognition models.

RESULTS

To determine whether or not incorporating peak features caused relevant improvements in EMG hand gesture recognition, a maxima-based, savitzky-golay-based, and wavelet-based peak detection algorithm were each run on EMG data to extract features, with each algorithm having a separate dataset. These datasets contained the peak features such as number of peaks or mean peak height as well as other common time series features and were used to predict the hand gesture a subject was making. A random forest model, which is effective at handling complex relationships between features, was used as the primary model for training.

To effectively understand how well the random forest predicted what hand gesture a given EMG signal was showing based on the given data, Stratified K-Fold cross-validation (SKF) was used as the training pipeline. SKF is particularly useful because, unlike a single train-test split where the evaluation might be highly dependent on the specific data split, SKF reduces this variance by measuring the performance across multiple splits. This method achieves a more accurate representation of the performance of a model on a dataset. Each dataset was split into five subsets, meaning that 20% of the datasets were used for

Before feature extraction, EMG signals must be preprocessed to remove noise and simplify analysis. For this experiment, a bandpass filter was used as it is commonly applied to isolate dominant frequencies by removing unwanted ranges. After filtering, a periodogram of an EMG signal from the database shows a notable reduction in 400–600 Hz frequencies **(Figure 1)**.

After pre-processing, we extracted peaks using the three algorithms. For comparison, the peaks were graphed on a random EMG signal from the database, illustrating the differences in which peaks are detected and which ones are missed by each algorithm. The maxima-based detection found the most peaks (n=71), with the savitzky-golay-based algorithm finding less (n=56), and the wavelet-based algorithm finding the least number of peaks (n=22) **(Figure 2)**.

After extracting features, the random forest model was trained on the peak feature datasets and the dataset without peak features using SKF. The first clear difference is that all three peak feature datasets, which all had an accuracy above 90%, outperformed the dataset without peak features, which had an accuracy of 88.5% **(Table 1)**. Among the datasets that used peak detection algorithms for feature extraction, peak features derived from the wavelet-based algorithm delivered the best performance, achieving an accuracy of 96.5% and lowest standard deviation of 2.9, surpassing features extracted from the maxima-based and savitzky-golay-based algorithms, which produced accuracies of 94.2% and 91.8%, respectively **(Table 1)**. Additionally, the precision, recall, and F1-scores for the wavelet-based peaks were the highest among the other two algorithms and baseline **(Table 1)**. Precision measures how many of the predicted positives were actually correct, recall reflects the ability to identify all actual positives, and the F1-score balances both metrics to provide a single measure of accuracy.

P-values were also calculated using a paired t-test between each algorithm and the baseline. The paired t-test compares the means of two related groups to determine whether their differences are statistically significant. In this case, it assesses whether the feature sets lead to significantly different model performances. The wavelet-based features were the most statistically significant with a p-value of 0.032 compared to the maxima-based and savitzky-golay-based features, which produced p-values of 0.054 and 0.109, respectively **(Table 1)**.

In addition to testing the accuracy of the peak feature datasets, feature importances were extracted to determine if the peak features had a real effect on accuracy. The feature importances showed the mean peak height (MPH) and number of peak (NP) features were among the top 10 for all 3 algorithms **(Figure 3)**. Additionally, electrodes placed on the wrist demonstrated better predictive capability than those on the forearm as most of the features in the top ten were signal features from electrodes in the wrist **(Figure 3)**.

A logistic regression model was also tested to show that the random forest classifier adds value for analyzing extracted peak features compared to simpler models. The regression model was trained with the highest-performing wavelet-

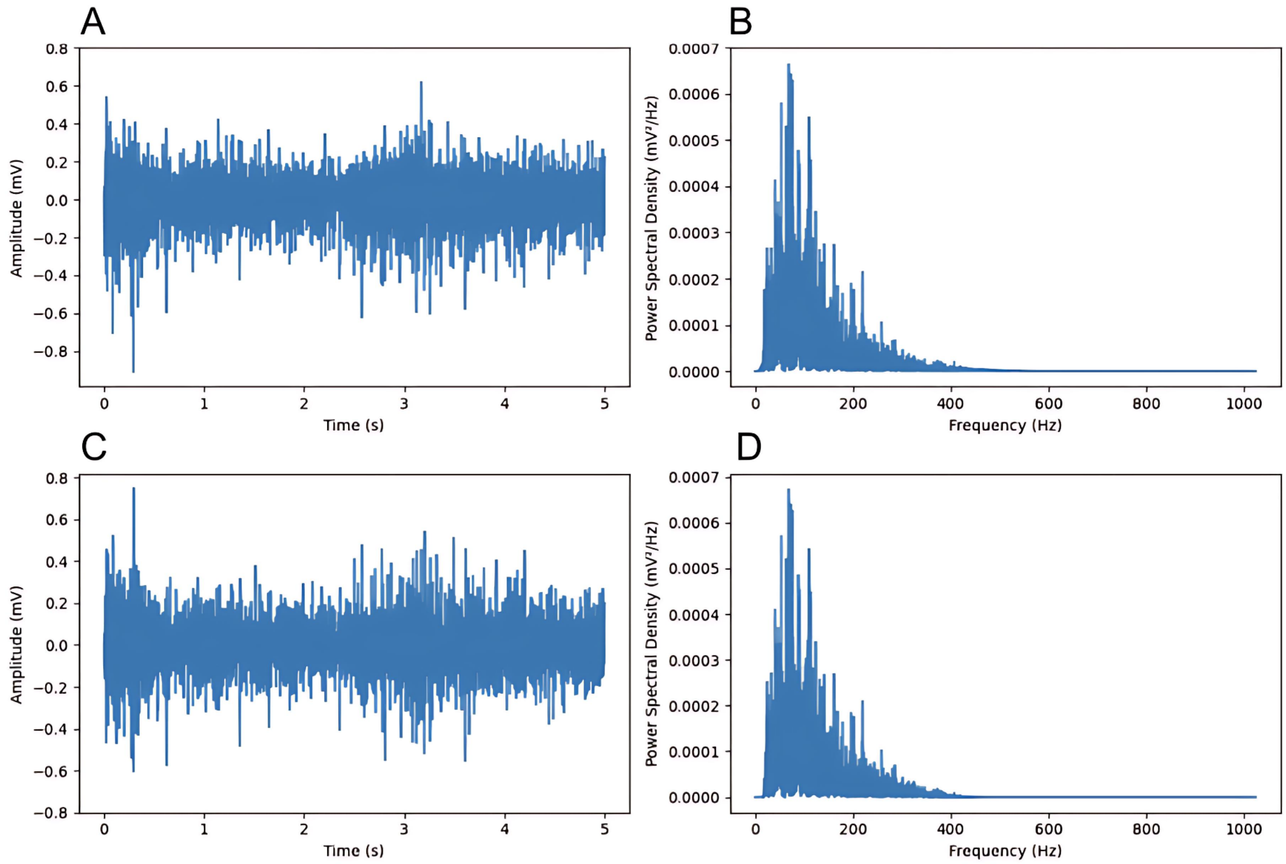

Figure 1. Effect of band-pass filtering on EMG signals using a Butterworth filter. Periodogram showing the power spectral density of an EMG signal before and after applying a 20–400Hz Butterworth band-pass filter (order 6) with recommended parameters (13). a) Graph and b) periodogram of original signal. c) Graph and d) periodogram of signal after band-pass is applied

based peak features using the same SKF framework and achieved an accuracy of 57% when distinguishing between hand open and hand closed **(Table 1)**.

DISCUSSION

To evaluate whether incorporating peak features led to meaningful improvements in EMG hand gesture recognition, three peak detection algorithms—maxima-based, savitzky-golay-based, and wavelet-based—were applied separately to EMG data to extract features, with each algorithm generating its own dataset. The model achieved a baseline accuracy of 88.5% without using peak detection-derived features **(Table 1)**. Incorporating peak features enhanced the model's accuracy, demonstrating their importance in providing valuable information to improve hand gesture classification and among the peak-feature datasets, the wavelet-based approach for peak detection was most effective, achieving an accuracy of 96.5 ± 2.9%, the highest accuracy and the lowest standard deviation for all three algorithms **(Table 1)**. The wavelet-based approach also achieved a *p*-value of 0.032 when compared to the baseline, showing that adding wavelet-based peak features created a statistically significant difference in model performance **(Table 1)**. These results show that measured wavelet-based peaks were the most distinct between each hand gesture, leading to better class separability and accuracy.

The comparison between a simple logistic regression model with the more complex random forest model shows that the more complex model is better suited for EMG signal analysis and categorization. When a logistic regression model was trained with the wavelet-based peak features, which were the best-performing features, the model achieved an accuracy of 57.0 ± 19.8% **(Table 1)**. With the accuracy being close to 50% and the deviation being extremely high coupled with the much higher 96% achieved by a random forest, it is clear that the simpler logistic regression model could not make effective distinctions between the EMG signals.

The savitzky-golay-based algorithm unexpectedly underperformed compared to the maxima-based and wavelet-based methods **(Table 1)**. Further, the savitzky-golay-based algorithm had a high *p*-value of 0.109, illustrating that adding the savitzky-golay features did not provide a statistically significant difference relative to the baseline model **(Table 1)**. This outcome could be attributed to the filter's tendency to eliminate critical information. While the savitzky-golay-based algorithm detected more peaks than the wavelet-based method, it identified fewer than the maxima-based approach **(Figure 2)**. However, the quality of the peaks the savitzky-golay-based algorithm identified was inferior to those detected by the wavelet-based method, as it primarily analyzed amplitude

https://doi.org/10.59720/24-321

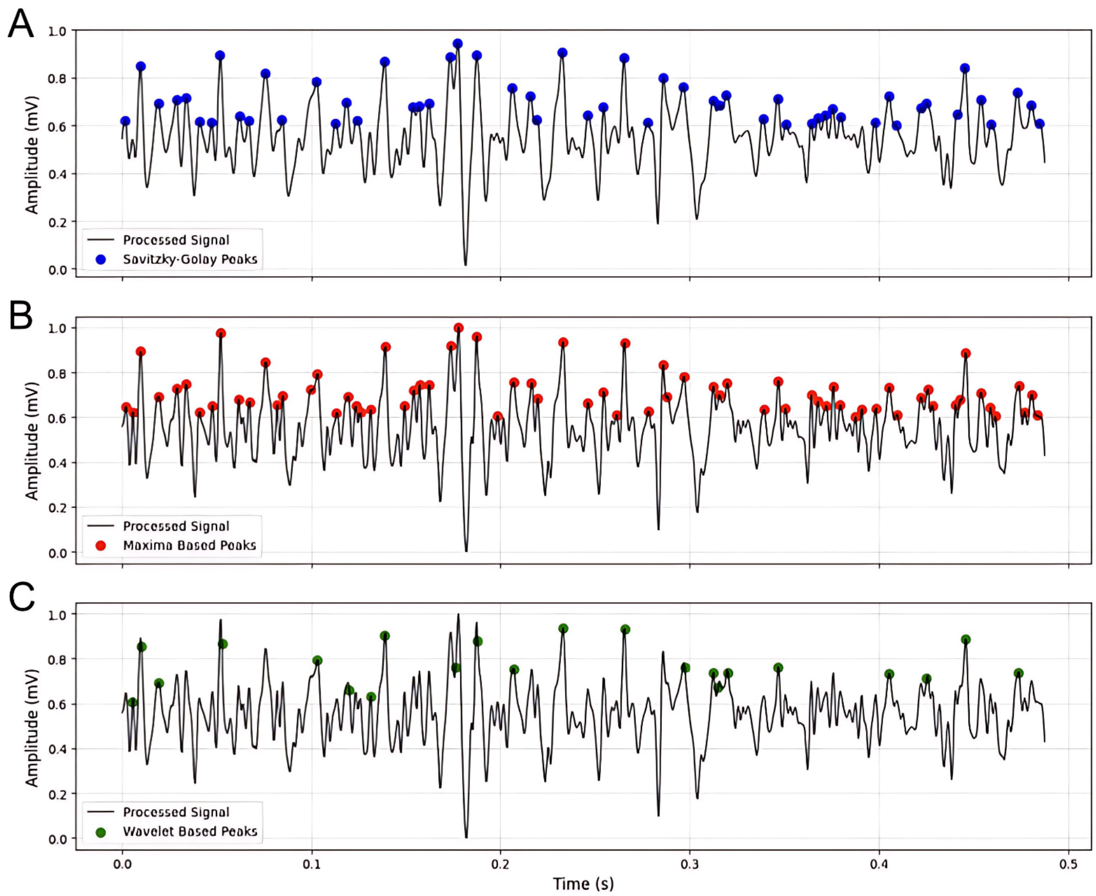

Figure 2. EMG signal with peaks identified by three different detection algorithms. EMG signal with peaks identified by the three algorithms analyzed in this study. As an EMG signal measures the electrical activity of a muscle, points of high activity or peaks represent relevant muscle contractions in the measurement area and are represented in the graphs. Peaks detected by the a) savitzky-golay-based algorithm (blue), b) maxima-based algorithm (red), and c) wavelet-based algorithm (green). Only peaks with an amplitude above 0.6 within an interval of 0.5 seconds are shown. Peaks detected by the savitzky-golay-based algorithm are displayed on the smoothed signal after applying the filter.

rather than shape. Additionally, the filtering process may have inadvertently removed significant peaks identified by the maxima-based method, both of which likely contributed to its diminished performance.

The effectiveness of the wavelet-based method can be attributed to its focus on signal shape rather than solely on amplitude, as seen in the savitzky-golay-based and maxima-based methods. Unlike these methods, which rely on point-by-point comparisons to detect peaks, the wavelet-based approach analyzes the overall shape of a range of values by comparing it to a predefined wavelet (10). This enables it to identify a wider variety of peaks, making it a more robust and versatile technique (10).

Regarding feature importance, the NP and MPH were among the most significant features, consistently ranking in the top 10 across all 3 datasets, showing that these features greatly contributed to the model's ability to distinguish between hand gestures **(Figure 3)**. Additionally, electrodes placed on the wrist demonstrated better predictive capability than those on the forearm, as most of the top-ranked features were from wrist channels **(Figure 3)**. This is likely due to the wrist's proximity to the hand, making it more sensitive to changes in gesture.

These findings underscore the effectiveness of including peak detection features, particularly from wavelet-based algorithms, in EMG-based hand gesture recognition models. The high accuracy of the wavelet-based features compared to the other two algorithms and the baseline, coupled with the high importance of the peak features, underscores the impact that these peak detection algorithms have on model

Our study trained a binary classifier, but hand gesture models are more useful when they can perform multi-classification on many different gestures, and much of the prior work on EMG hand gesture recognition handles more than two gestures (2). While our study has shown the possibility of using peak features, further work must be done on using peak features in a larger dataset with more gestures to truly see the performance of these features on a more real-world scale. Further work could also be done to test more peak detection algorithms. While this study focused on three, many more exist, which could provide better results. Additionally, our study extracted only four peak features from the algorithms: number of peaks, mean peak height, standard deviation of peak height, and time of max peak height. Other peak features may offer more class separability.

Improvements in EMG-based prosthesis control models

https://doi.org/10.59720/24-321

Method	Classification	Precision	Recall	F1-Score	P-values vs baseline
Without Peak Features	Hand Open	87.4 ± 8.0	90.8 ± 13.0	88.5 ± 8.4	-
	Hand Closed	92.3 ± 10.7	85.8 ± 12.4	88.0 ± 8.6	-
	Overall Accuracy	88.5 ± 8.0	-	-	-
Maxima-Based Peak Detection	Hand Open	95.8 ± 5.2	93.1 ± 5.7	94.2 ± 3.5	0.054
	Hand Closed	93.5 ± 5.4	95.0 ± 6.1	94.1 ± 4.0	-
	Overall Accuracy	94.2 ± 3.7	-	-	-
Wavelet-Based Peak Detection	Hand Open	95.8 ± 5.2	97.8 ± 4.4	96.6 ± 2.8	0.032
	Hand Closed	97.8 ± 4.4	95.3 ± 5.8	96.3 ± 3.0	-
	Overall Accuracy	96.5 ± 2.9	-	-	-
Savitzky-Golay-Based Peak Detection	Hand Open	90.1 ± 8.6	95.6 ± 5.4	92.5 ± 5.2	0.109
	Hand Closed	95.1 ± 6.1	87.8 ± 11.2	90.9 ± 7.1	-
	Overall Accuracy	91.8 ± 7.1	-	-	-
Wavelet-Based Peak Detection with Logistic Regression	Hand Open	57.0 ± 16.5	70.6 ± 16.4	62.9 ± 16.3	-
	Hand Closed	56.0 ± 26.2	43.3 ± 25.1	48.5 ± 25.6	-
	Overall Accuracy	57.0 ± 19.8	-	-	-

Table 1. Comparison of peak detection algorithms for EMG-based hand gesture recognition. Peak features were extracted from three different detection algorithms, generating separate datasets for each algorithm. A baseline dataset without peak features and a Logistic Regression model using wavelet-based peak features were also included for comparison. Each dataset, excluding the Logistic Regression, was trained using a hyperparameter-tuned random forest model. Accuracy metrics (mean ± standard deviation) were calculated across all datasets using K-Fold cross-validation. Performance metrics include classification accuracy, precision, recall, and F1 score. *P*-values (calculated using paired *t*-test) was also calculated for each algorithm compared to the baseline dataset without peak features to determine if adding peak features provided a statistically significant difference in model performance.

can have a significant positive impact on the lives of arm amputees. Enhanced model accuracy and reliability can lead to more intuitive and responsive prosthetic devices, allowing users to perform daily tasks more easily and confidently (11). This can improve arm amputees' overall quality of life by increasing their independence and ability to engage in various activities. Additionally, advancements in this field could pave the way for further innovations in prosthetic technology, ultimately benefiting a larger population of individuals who rely on these devices.

https://doi.org/10.59720/24-321

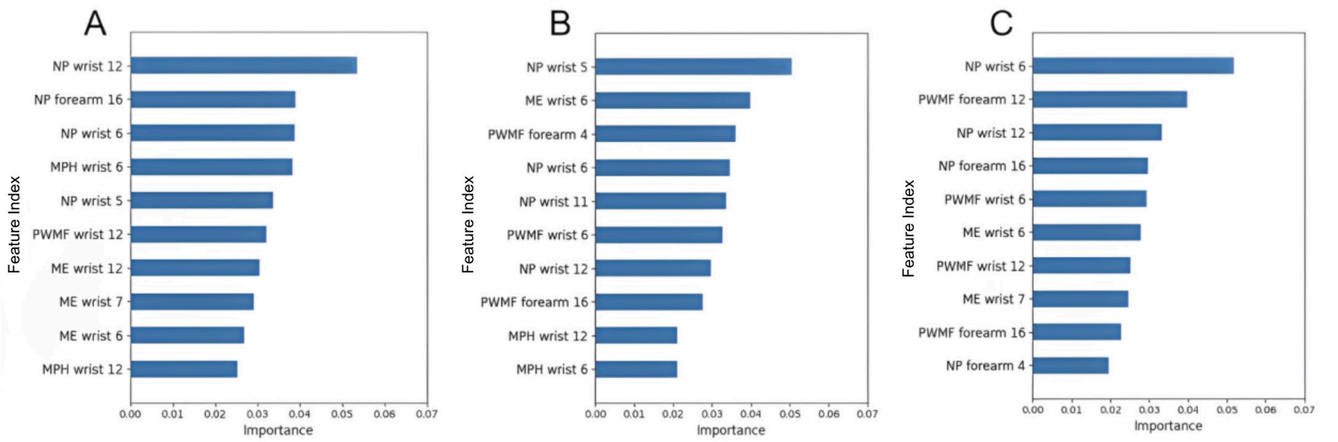

Figure 3. Top 10 most important features for each peak detection algorithm. The 10 most important features identified by a random forest model for each peak detection algorithm (n=3). Top 10 feature importances for a) maxima-based, b) wavelet-based, and c) savitzky-golay-based peaks. The feature naming convention includes the feature acronym, electrode location, and channel number. Corresponding feature acronyms and full feature names are detailed in Table 2.

MATERIALS AND METHODS

Dataset Description

The Gesture Recognition and Biometrics ElectroMyo-Gram (GRABMyo) dataset from the PhysioNet library was used in this study (12–14). This dataset contains EMG recordings sampled at 2048 Hz from 43 participants (23 men and 20 women) with an average age of 26.35 ± 2.89 years, performing 17 distinct hand gestures, including rest. On the forearm and wrist of each individual, 16 and 12 monopolar sEMG electrodes (AM-N00S/E, Ambu, Denmark) were placed, respectively, each in the form of two rings (12) **(Figure 4**).

The collection process took 3 days, during which all 43 participants performed 17 gestures for 7 trials, with 5 seconds of recorded sEMG data for 28 channels per trial and a 10-second rest period between contractions (12).

This data set was selected for its open-access availability, recent publication date, and ease of use. Its files follow a consistent naming scheme that clearly indicates the participant, trial, and gesture, making it straightforward to download and integrate into code. Additionally, the publishers provided transparent details about the equipment used and the participants' demographics, further enhancing its suitability for this study.

In the GRABMyo database, there are many separate 5-second EMG recordings; using them all would lead to extensive training time and complexity. To address this, the data was reduced to the most relevant gestures and channels.

Out of the 17 different gestures recorded in the database, this study focused on classifying two: Hand open and Hand closed **(Figure 5)**. These steps reduced the over 100,000

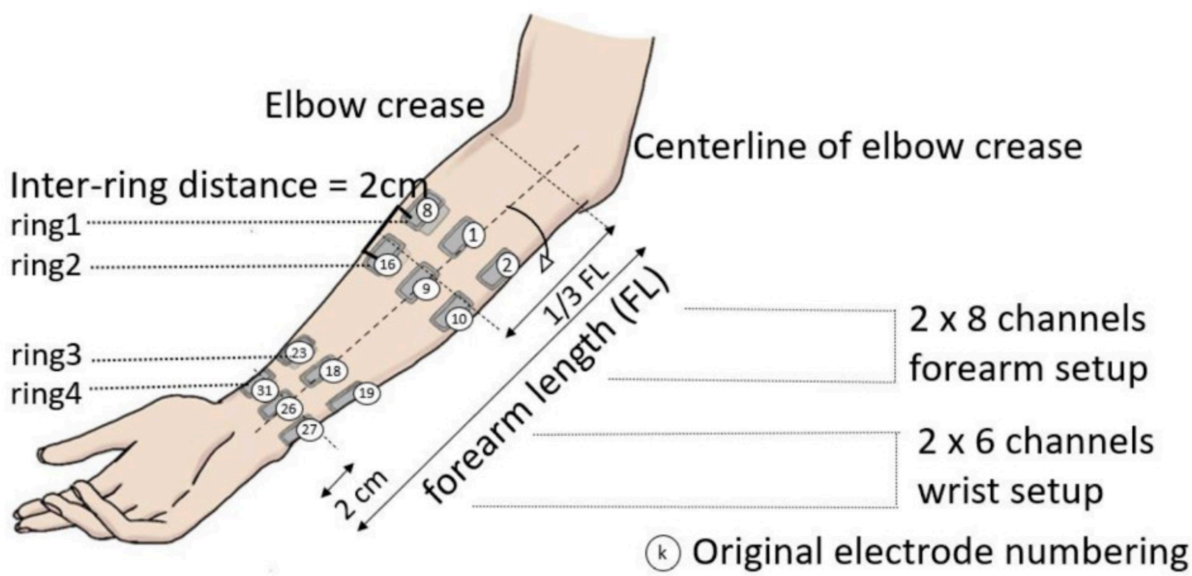

Figure 4. Forearm and wrist electrode locations. From the GRABMyo database (12-14). Location of the 28 electrodes on the forearm and wrist. Each of these electrodes generates a separate EMG signal for each gesture. No changes were made to the original image, and the database was published under an Open Data Commons Attribution License v1.0 (https://opendatacommons.org/licenses/by/1-0/).

JOURNAL OF
EMERGING INVESTIGATORS

https://doi.org/10.59720/24-321

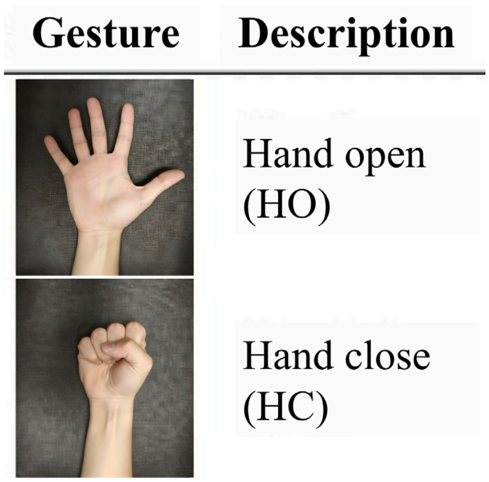

Figure 5: Gesture List. From the GRABMyo database, image of two gestures that were created by participants in the study (12-14). This study distinguishes between the Hand Open and Hand Close gestures. Original image was altered to show 2 of the 16 gestures, and the database was published under an Open Data Commons Attribution License v1.0 (https://opendatacommons.org/licenses/by/1-0/).

separate recordings to 86, with 43 recordings per gesture. This amount was much more manageable for training and testing.

Pre-processing

Pre-processing is an essential step in EMG signal analysis to remove unwanted noise and make data suitable for analysis. Multiple such techniques were employed in this study for each signal used in training.

Firstly, a Butterworth band-pass filter of 20-400Hz with an order of 6 is applied to the signal to remove unwanted frequencies **(Figure 1)**. The high-pass and low-pass values were proposed to effectively reduce noise and artifact contamination while preserving the essential information in the signal (13).

After applying the band-pass filter, the signals are normalized using the min-max scaling method, which transforms the original data to a range between 0 and 1. This normalization is particularly useful as it ensures features are on a similar scale, preventing very large features from skewing the learning process. The min-max normalization is defined by **equation 1:**

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \qquad \text{(Equation 1)}$$

Where x is the original data value, x_{min} and x_{max} are the minimum and maximum values of the signal, respectively, and x' is the normalized value.

Peak Detection Algorithms

Three peak detection algorithms were tested to evaluate whether these peaks impact model performance. A maxima-based peak detection algorithm was the first to be tested. In this algorithm, a point x_i in a signal $x(t)$ is identified as a peak if it satisfies:

$$x_i > x_{i-1} and x_i > x_{i+1} \qquad \text{(Equation 2)}$$

Feature Name	Type
Mean (ME)	Time-Domain
Standard Deviation (SD)	Time-Domain
Maximum Power Frequency (MPF)	Frequency-domain
Maximum Power Spectral Density (MSD)	Frequency-domain
Power Weighted Mean Frequency (PWMF)	Frequency-domain
Mean Absolute Value 4th Level Decomposition (CD4)	Time-Frequency
Number of Peaks (NP)	Time-Domain
Time of Greatest Peak (MPT)	Time-Domain
Mean Height of Peaks (MPH)	Time-Domain
Standard Deviation of Peak Height (SPDH)	Time-Domain

Table 2. List of features utilized for EMG signal analysis. The features extracted from EMG signals for machine learning applications, categorized into time-domain, frequency-domain, and frequency-time-domain features. Additionally, it includes peak features generated by the peak detection algorithms employed in the study. Each feature is accompanied by its corresponding acronym.

A given point in the signal will be set as a peak if the amplitude of points directly before *(i - 1)* and after *(i + 1)* are less than the amplitude of the current point.

Next is a wavelet-based peak detection algorithm, which detects peaks using a Continuous Wavelet Transform (CWT) on a specified wavelet. If $x(t)$ is the original signal and $W(a, b)$ is the wavelet coefficient, CWT is defined as:

$$W(a,b) = \int_{-\infty}^{\infty} x(t)\psi\left(\frac{t-b}{a}\right) dt \qquad \text{(Equation 3)}$$

where $\psi(t)$ is the wavelet function, scaled by a and translated by b. a controls the dilation and compression of the wavelet while b translates the wavelet across the time-series. The wavelet is compared to the signal at different scales by stretching or compressing to match peaks of various sizes. Peaks that consistently stand out in these scales are set as true peaks, as this consistency indicates that the shape is not due to noise or random fluctuations. Simply, this algorithm will take the given wavelet and compare it to different shapes in the signal, and if these shapes match close enough to the given wavelet, a peak will be detected. The algorithm takes two parameters that define the minimum and maximum scales of the wavelet, which control how much the wavelet is stretched or compressed when scanning the signal. Since EMG signals typically exhibit sharp, narrow peaks rather than broad ones, smaller scale values are more appropriate.

The Ricker wavelet (also known as the Mexican Hat wavelet) was used as its shape resembles a peak, with a central maximum and symmetric tails on each side (10). The Ricker wavelet is also easily scalable by changing the value of a; thus, this wavelet can detect both narrow and wide peaks

https://doi.org/10.59720/24-321

Model	Number of Trees	Max Tree Depth	Max Features	Bootstrapping
Maxima-Based	845	18	log2	False
Savitzky-Golay-Based	229	6	sqrt	False
Wavelet-Based	403	14	log2	False

Table 3. Hyperparameter values for each random forest model. The hyperparameter values for each of the peak detection datasets. Four key hyperparameters for random forest Models are optimized for each model using Bayesian Optimization and the Optuna package. Number of Trees specifies how many decision trees are in the forest; more trees generally improve performance but increase computational cost. Max Tree Depth limits how deep each tree can grow, controlling model complexity to prevent overfitting. Max Features determines the maximum number of features considered when splitting nodes, introducing randomness and helping reduce overfitting. Bootstrapping enables random sampling of data points with replacement for training each tree, ensuring diversity among trees and improving generalization.

without complex non-linear fitting (10).

Lastly, a savitzky-golay-based peak detection algorithm was tested. In this algorithm, the savitzky-golay filter, which smooths the signal while preserving its general shape, is applied to the signal as it reduces noise. Maxima are then detected in the signal as peaks. Smoothing the signal beforehand could remove unwanted peaks caused by noise. The savitzky-golay filter smooths a signal by fitting a polynomial to a sliding window of data points. For a signal x_i smoothed value $\dot{x}_i$ is computed as:

$$\dot{x}_i = \sum_{j=i-\frac{w}{2}}^{i+\frac{w}{2}} c_{j-i} \cdot x_j \qquad \text{(Equation 4)}$$

where c_{j-i} are the filter coefficients determined by polynomial fitting within the window of length *w*. The size of the sliding window, which is directly proportional to how smooth the signal becomes, and the degree of polynomial fitting, is passed to the algorithm as a parameter. As EMG signals are only moderately complex, a polynomial degree of fitting of 2 was used with a window length of 10 to smooth the signal without removing significant information.

Feature Extraction

Extracting relevant features from biomedical signals is a crucial step in the machine learning pipeline, as the quality of features can directly impact the model's accuracy. This study utilizes multiple time-domain, frequency-domain, and frequency-time-domain features (Table 2). These features are commonly used in signal processing or suggested for EMG analysis. The mentioned peak detection algorithms were also used to generate four peak features: number of peaks, mean peak height, standard deviation of peak height, and time of max peak height.

For the Time-Frequency domain feature, the Discrete Wavelet Transform (DWT) is utilized, which is particularly effective for analyzing non-stationary signals at multiple

resolutions (14). In this paper, we employ second-order Daubechies wavelets based on previously published recommendations as it was found that these wavelets achieved the highest class separability for EMG signals (14). The mean absolute value (MAV) of the fourth level detail component was used as other MAV features were redundant due to high correlation.

Because each recording had 28 different channels in the wrist and forearm, feature reduction was necessary to limit over-fitting and redundancy when training. To this end, each feature was compared to every other feature, and if a feature pair had a correlation equal to or above 0.75, one of the two was removed.

Models

This study uses a random forest model to analyze the features extracted from EMG signals. Random forest models leverage ensemble learning, which combines multiple decision trees to improve prediction accuracy compared to using a single model (15). Each decision tree in the random forest comprises decision nodes, where conditions are evaluated to create branches leading to further nodes, ultimately reaching an outcome at the leaf nodes (15).

While a single decision tree may struggle with complex data, random forests offer more robust and accurate results by aggregating the predictions of many trees (15). This collaborative approach reduces overfitting and enhances performance on diverse datasets.

Random forest is especially effective for EMG-based gesture recognition because it can manage high-dimensional feature spaces and capture intricate interactions among the features derived from EMG signals (15). Having Time-Domain, Frequency-Domain, and Time-Frequency Domain features requires a model like random forest, which can efficiently manage and integrate these diverse feature types. To further improve the model's predictive capability, hyperparameter tuning was conducted for the number of trees in the forest, the maximum depth of each tree, the number of features to consider when splitting a node, and whether to use bootstrapping **(Table 3)**.

To effectively understand how well the random forest predicted what hand gesture a given EMG signal was showing based on the given data, Stratified K-Fold cross-validation (SKF) was used as the training pipeline. SKF works by splitting the dataset into a specific number of subsets, where each iteration of the model validates on one subset and trains on the rest of the data. SKF further ensures that the validation and training sets have the same proportion of each classification. After training and testing all the subsets, the final metrics are calculated by averaging the accuracy metrics from each subset. The standard deviation for each metric is also included.

After testing the random forest and finding that the wavelet features were the most effective, a logistic regression algorithm was tested on the wavelet features to determine how well a simpler model would perform on the data. Logistic regression works by combining the extracted EMG features with weights, then passing the result through a sigmoid function to produce a value between 0 and 1. The model adjusts the weights during training to make better predictions.

Software and Tools

The code for this project was written in python (v3.10.12) in Google Colab. In addition, many libraries assisted in the pre-processing, model creation, and hyperparameter tuning steps. Pandas (v2.2.2) was used for data manipulation and pre-processing, Matplotlib (v3.8.0) was used to graph figures, scikit-learn (v1.6.0) was used to create and train the models used in the study, and Optuna (v4.1.0) was used to conduct hyperparameter tuning on the random forest.

The complete code for this study can be found on GitHub which includes clear descriptions and the code for each step in the process (16).

Received: September 23, 2024
Accepted: February 27, 2025
Published: January 10, 2026

REFERENCES

1. Dillingham, Timothy R, et al. "Limb amputation and limb deficiency: epidemiology and recent trends in the United States." *Southern Medical Journal*, vol. 95, no. 8, 2002, pp. 875–84. https://doi.org/10.1097/00007611-200208000-00018
2. Jaramillo-Yáñez, Andrés, et al. "Real-time hand gesture recognition using surface electromyography and machine learning: A systematic literature review." *Sensors*, vol. 20, no. 9, 2020, p. 2467. https://doi.org/10.3390/s20092467
3. Xiong, Dezhen, et al. "Deep learning for EMG-based human-machine interaction: A review." *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 3, 2021, pp. 512–33. https://doi.org/10.1109/JAS.2021.1003865
4. Corbett, Elaine A, et al. "Comparison of electromyography and force as interfaces for prosthetic control." *Journal of Rehabilitation Research and Development*, vol. 48, no. 6, 2011, p. 629. https://doi.org/10.1682/jrrd.2010.03.0028
5. Kamavuako, Ernest N, et al. "Combined surface and intramuscular EMG for improved real-time myoelectric control performance." *Biomedical Signal Processing and Control*, vol. 10, 2014, pp. 102–07. https://doi.org/10.1016/j.bspc.2014.01.007
6. Kim, Jonghwa, et al. "EMG-based hand gesture recognition for real-time biosignal interfacing." *Proceedings of the 13th International Conference on Intelligent User Interfaces*, 2008, pp. 30–39. https://doi.org/10.1145/1378773.1378778
7. Phinyomark, Angkoon, et al. "A novel feature extraction for robust EMG pattern recognition." *arXiv*, 2009. https://doi.org/10.48550/arXiv.0912.3973
8. Jia, Guangyu, et al. "Classification of electromyographic hand gesture signals using machine learning techniques." *Neurocomputing*, vol. 401, 2020, pp. 236–48. https://doi.org/10.1016/j.neucom.2020.03.009
9. Du, Pan, Warren A. Kibbe, and Simon M. Lin. "Improved Peak Detection in Mass Spectrum by Incorporating Continuous Wavelet Transform-Based Pattern Matching." *Bioinformatics*, vol. 22, no. 17, 2006, pp. 2059–2065. https://doi.org/10.1093/bioinformatics/btl355
10. Parajuli, Nawadita, et al. "Real-Time EMG-Based Pattern Recognition Control for Hand Prostheses: A Review on Existing Methods, Challenges and Future Implementation." *Sensors*, vol. 19, no. 20, 2019, p. 4596. https://doi.org/10.3390/s19204596
11. Goldberger, A L, et al. "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals." *Circulation*, vol. 101, no. 23, June 2000, E215–20. https://doi.org/10.1161/01.CIR.101.23.e215
12. De Luca, Carlo J, et al. "Filtering the surface EMG signal: Movement artifact and baseline noise contamination." *Journal of Biomechanics*, vol. 43, no. 8, 2010, pp. 1573–79. https://doi.org/10.1016/j.jbiomech.2010.01.027
13. Mahdavi, Farzaneh Akhavan, et al. "Surface Electromyography Feature Extraction Based on Wavelet Transform." *International Journal of Integrated Engineering*, vol. 4, no. 3, Dec. 2012. https://penerbit.uthm.edu.my/ojs/index.php/ijie/article/view/615
14. "What is random forest?" *IBM*. https://www.ibm.com/topics/random-forest. Accessed 24 Oct. 2024.
15. Li, Zhong, et al. "Estimation of Knee Movement from Surface EMG Using Random Forest with Principal Component Analysis." *Electronics*, vol. 9, no. 1, 28 Dec. 2019, p. 43, https://doi.org/10.3390/electronics9010043
16. Nathan, Adikesh. *EMG Analysis Research*. GitHub, https://github.com/adikeshn/EMG-Analysis-Research.

Copyright: © 2026 Nathan and Raju. All JEI articles are distributed under the attribution non-commercial, no derivative license (http://creativecommons.org/licenses/by-nc-nd/4.0/). This means that anyone is free to share, copy and distribute an unaltered article for non-commercial purposes provided the original author and source is credited.