Cureus Journal of Computer Science

Open Access Original Article

Surface Electromyography-Based Gesture Recognition via Optimizer Selection and Multi-Criteria Performance Analysis

Mohsin Ali Shah [1] , Nisar Uddin [2]

1. Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar, PAK 2. Rehabilitation Medicine, Saidu Group of Teaching Hospitals, Swat, PAK

Corresponding author: Mohsin Ali Shah, mohsinalishah13@gmail.com

Received 08/06/2025
Review began 08/07/2025
Review ended 11/07/2025
Published 11/09/2025

DOI:
https://doi.org/10.7759/s44389-025-09497-9

Abstract

Multichannel surface electromyography-based hand gesture recognition plays an important role in the development of intuitive prosthetic and human-machine interfaces. It requires high accuracy, minimal noise degradation, and electrode drift tolerance for real-world deployment. The goal is to find the best tuned classifier configurations that balance accuracy and real-world robustness, comparing grid search, random search, and Bayesian optimization for the four classical classifiers: K-Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), and Support Vector Machine (SVM). Their performance is evaluated by repeated stratified 5 × 5 cross-validation with 25 runs and leave-one-subject-out testing. Furthermore, we use a multi-objective Pareto front analysis considering three factors: (1) accuracy, (2) adaptation to simulated drift, and (3) relative accuracy degradation under noise. KNN reaches stable 98.4% accuracy on all optimizers, but Bayesian optimization performs better than other optimization techniques in terms of accuracy and other performance metrics when using SVM, DT, and RF classifiers. Additionally, Bayesian optimization gets KNN into the Pareto front with 98.44% accuracy, 98.9% adaptability, and a minimal 0.10% decrease in noise, which likewise clearly improves SVM and RF. These results provide guidance about how to choose effective, real-time electromyography classifiers not only for accuracy but also through trade-offs of accuracy, adaptability, and noise tolerance in prosthetic and rehabilitation systems.

Categories: AI applications, Bioinformatics, Machine Learning (ML)
Keywords: bayesian optimization, electromyography (emg), gesture classification, grid search, random search

Introduction

Signals generated by the skeletal muscles during gestures are known as electromyography (EMG) signals [1]. Surface electrodes are used to capture EMG-based gesture recognition signals, which are commonly from the forearm. It is possible to classify different hand gestures by processing EMG data and applying machine learning techniques. For surface EMG-based gesture recognition, recent reviews [2,3] provide comprehensive overviews of data collection, preprocessing, feature extraction, and machine-learning processes. It is important that these gestures be accurately classified using EMG because EMG gesture recognition has applications in a variety of domains [4], such as prosthetic control [1], sports medicine [5], virtual reality environments [6], sign language interpretation [7], and assistive technologies for disability aid [8]. It provides a natural and easy approach to controlling devices using hand and arm movements, eliminating the requirement for vision-based tracking.

Most of the systems, whether they are EMG-based gesture recognition systems or any other classification system, use different optimizers to tune the classifier hyperparameters. Choosing the right optimization algorithm is critical for efficiently training machine learning models. The optimizer defines how the model parameters are changed during training in order to minimize the loss function. The choice of optimizer is particularly important for prosthetic systems because it affects training time, inference latency, model size, adaptability to changing EMG signals (e.g., electrode shift, muscle fatigue, or inter-session drift), as well as final accuracy. In prosthetic applications, these properties have a direct impact on user comfort and real-time response [9,10]. Different optimization techniques have different pros and cons that cause them to perform better or worse depending on the model design. Techniques such as Bayesian optimization using a Gaussian process (a sequential model-based approach using a Gaussian-process surrogate to guide hyperparameter search) [11], grid search [12], and random search [13] are frequently used to adjust hyperparameters effectively. These techniques pair well or poorly with different learning algorithms due to tradeoffs they make between search breadth, computational cost, and convergence rates.

Recent optimization in EMG classification is utilizing more and more optimization techniques in the EMG classification process, including the use of genetic algorithms (GA), particle swarm optimization (PSO), grey wolf optimization, and hybrid metaheuristic approaches, to automate feature selection and model

How to cite this article
Ali Shah M, Uddin N (November 09, 2025) Surface Electromyography-Based Gesture Recognition via Optimizer Selection and Multi-Criteria Performance Analysis. Cureus J Comput Sci 2 : es44389-025-09497-9. DOI https://doi.org/10.7759/s44389-025-09497-9

optimization. Neural networks and support vector machines (SVMs) have been optimized with PSO [14]. Other hybrid methods, such as grey wolf optimization-neighborhood component analysis, have also been used in EMG classification tasks [15]. Large search spaces are effectively explored by these metaheuristic methods, which is why they are recommended. They also lessen the need for manual hyperparameter tweaking, particularly in situations where calculating analytical gradients is expensive or unavailable.

A single study proposed an artificial neural network that auto-configures itself through a randomized trial-and-error to achieve a network depth and network parameters with minimal manual configurations [16], which reflects a trend of auto-configurable artificial neural networks through AutoML tools on EMG modeling. This new paradigm in AutoML emphasizes the shift toward self-optimizing, adaptive architectures that lessen human bias and speed up the development of EMG models.

In addition, deep learning-based gesture recognition has also been accomplished in the past few years using convolutional neural networks (CNN) and more recently with convolutional recurrent neural networks. Studies by Tacca et al. [17], Zafar et al. [18], and Xie et al. [19] have even seen promising results. However, these methods typically depend on large labeled datasets and substantial computational power, making it difficult to implement on low-power embedded systems with limited processing capabilities. On the other hand, classical machine learning models are still in use today in real-time systems like prosthetic controllers because of their positive tradeoff between accuracy and computational complexity, especially in resource-constrained systems.

Although the current EMG literature is mostly centered around accuracy, and in many cases, high performance is reported on larger datasets such as the Ninapro DB2 (17 gestures) with the CNN models incorporated into SVM [20] or traditional methods [21,22], we pursue a different approach here. Many recent high-accuracy studies [23-25], however, only assess performance on common benchmark splits. Adaptability to electrode displacement, additional sensor noise, and inter-session drift is not specifically tested.

Much more work has been done on the use of many machine learning models and algorithms to deal with gesture classification. One of the most noticeable or important things is the published work related to SVM, which has also achieved very good results in this field. Khushaba et al. [26] performed a comparative study between SVMs, linear discriminant analysis, KNN, and naive Bayes classifiers in EMG data, which led to the conclusion that SVMs present better performance when compared with other techniques. Naik et al. [27] used multi-class SVMs together with wavelet features computed from EMG signals to classify hand gestures. Atzori et al. [28] used linear discriminant analysis classifiers on EMG data for high classification accuracy. Since SVMs have been the subject of voluminous research, many alternatives, such as DT, RF, etc., are equally well-performing on EMG-based gesture sets. Gailey et al. [29] showed the DT model was effective for real-time EMG pattern recognition and was used for the control of powered lower limb prostheses. In a study by Mohapatra et al. [30] using Ninapro-DB5, hand-crafted features - such as waveform length, mean absolute value, and root mean square - combined with an SVM classifier achieved an accuracy of 84%.

In recent years, EMG-based gesture recognition has seen new advancements with the rise of deep learning approaches. In order to address privacy issues on the edge devices that use muscle activity signals for controlling robotic systems, a federated learning-based deep learning model was proposed by Zafar et al. [18]. This study was performed by applying short-term frequency transform on an eight-channel EMG armband using federated learning techniques such as federated averaging (FedAvg), federated optimization in heterogeneous networks (FedProx), and federated stochastic gradient descent (FedSGD). In complex gesture recognition, the temporal feature extraction is enhanced by a channel-fused gated temporal convolutional network [18], achieving 92.71% accuracy on 53 gestures from the NinaPro-DB5 dataset. Tacca et al. [17] demonstrated that high-density EMG techniques, utilizing 150-electrode forearm sleeves, significantly improved gesture classification accuracy, achieving 97.3% for 37 hand movements with deep learning, and enabled real-time prediction of joint angles for practical applications. Also, feature selection methods have been investigated by Rumman et al. [31] for enhancing performance. To select discriminative features, a swarm intelligence-based model is proposed, which outperforms other conventional optimizers like PSO, genetic algorithm, and gravitational cohesion. Collectively, these studies point out advancements in EMG-based gesture recognition with deep learning, hybrid architecture, transfer learning, and feature optimization to improve accuracy. With deep learning-based approaches giving great improvements, these can be achieved at the cost of large datasets and high computational resources. In our study, we systematically compare three hyperparameter optimization methods for multiple ML classifiers performing EMG-based gesture recognition and gain insights on how to optimize model performance using the limited data and resources available.

Hyperparameter tuning with traditional machine learning models plays a major role in achieving high classification performance. Improper hyperparameter tuning can cause our model to underfit or overfit, which will hinder the generalization of the model. Hyperparameter optimization has been a topic of much research over the years, and some popular methods include grid search, random search, Bayesian optimization, etc. Grid search is an exhaustive technique in which a predefined set of values is evaluated for each hyperparameter used [12]. This technique can be expensive in terms of computation for high-dimensional hyperparameter spaces. In contrast, random search is a simple stratified method introduced by Bergstra and Bengio [13] that randomly samples from the hyperparameter space using specified distributions to reduce computation while still exploring diverse configurations. This was followed by the Bayesian optimization work of Snoek et al. [11], which takes another approach that could also do well using fewer iterations than a grid or random search. This technique builds a model over the objective function as a Gaussian process and then uses an acquisition function to direct exploration towards parts of hyperparameter space. The optimization process using Bayesian methods is computationally more expensive and has shown state-of-the-art performance in a plethora of machine learning applications.

Some studies have explored the Bayesian optimization techniques, albeit in a limited scope. Gao et al. [32] integrated EMG and inertial measurement unit signals into feature fusion and applied it on a Bayesian XGBoost algorithm for classifying lower limb movement patterns using 94.42% accuracy. A U-Net architecture consisting of a MobileNetV2 encoder and a bidirectional long short-term memory network network was proposed [33]. Based on Bayesian optimization, the bidirectional long short-term memory model was fine-tuned, and the combination achieved higher hand gesture classification accuracy upon several hand gesture databases. In the work done by Miah et al. [34], they used 23 feature extraction techniques on EMG signals and applied an extra tree classifier to select a feature from the EMG signal. However, a comprehensive comparative evaluation of grid search, random search, and Bayesian optimization techniques across multiple classifiers, including K-Nearest Neighbors (KNN), SVM, Decision Tree (DT), and Random Forest (RF), for EMG-based gesture recognition has not been extensively studied. Such an analysis is crucial for understanding the relative strengths, weaknesses, and trade-offs of each optimizer-classifier combination, enabling researchers and developers to make informed choices when designing high-performing, robust EMG-based gesture interfaces.

Instead of many gestures, we focus on six key prosthetic hand gestures to simplify the system complexity, facilitate user control, and improve real-time decoding performance. More to the point, we consider models with respect to three important axes - accuracy, adaptability to electrode drift, and noise-induced accuracy drop - instead of focusing only on accuracy. The analysis of trade-offs among these metrics is done and illustrated with the help of Pareto fronts (a set of non-dominated solutions in a multi-objective optimization problem, where improving one objective would lead to the degradation of at least one other objective) [35].

The objective of this research is to find the best combination of classifier and hyperparameter optimization approach for recognizing hand gestures based on EMG. To achieve this, the study employs a multi-metric assessment paradigm assessing performance using accuracy, adaptability to electrode drift, and noise-induced accuracy degradation. The findings of this study aim to provide insight into the future development of a dependable, real-time prosthetic control device. We compare three hyperparameter optimization techniques (grid search, random search, and Bayesian optimization) of EMG-based gesture recognition using multiple classical classifiers. We use classical machine learning classifiers in this work because of their computational effectiveness and compatibility with low-resource embedded systems.

Materials And Methods

The proposed study is divided into four parts: the dataset, data segmentation, feature extraction, and classification and optimization. Figure *1* depicts the whole workflow.

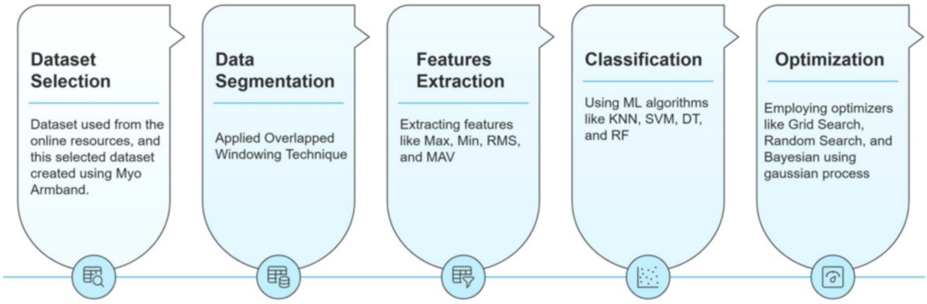

FIGURE 1: Workflow of the proposed study: (1) Dataset, (2) Data segmentation, (3) Feature extraction, and (4) Classification and optimization

The diagram outlines the sequential stages of the proposed pipeline.
Classifiers used are as follows: KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, and RF = Random Forest.
Optimization techniques include Grid Search, Random Search, and Bayesian Optimization based on Gaussian processes. The dataset utilized was collected using the Myo Armband.

The dataset

We utilized a publicly accessible dataset [36]. There are seven gestures included in the dataset: extended palm, hand fist clenched, wrist flexion, wrist extension, radial deviation, ulnar deviation, and a hand at rest. However, for this study, six gestures were used for classification, excluding the extended palm gesture because the data available is not sufficient for this gesture. This raw EMG signals dataset contains the data collected from 36 people using a Myo armband, as shown in Figure *2*, with a fixed sampling rate of 200 Hz.

FIGURE 2: Myo armband

The Myo armband (Model: Thalmic Labs MYB-0001, manufactured by Thalmic Labs Inc., Waterloo, Ontario, Canada) consisted of eight medical-grade stainless steel dry sensors placed uniformly around the forearm to record electromyography signals and send them wirelessly to the PC through Bluetooth with a sampling frequency of 200 Hz.

Each subject did two series consisting of six of the fundamental gestures selected from each series, the motion being taken in each for 3 seconds, and after 3 seconds, there was a break. The preprocessing step of bandpass filtering was applied in order to improve data quality. In Figure *3*, all the hand gestures that are present in the dataset are shown, except hand rest.

FIGURE 3: Set of hand gestures used for electromyography classification

Data segmentation

The overlapping window technique is used during the signal segmentation step. The window size for this approach is very carefully set to be 1,000 per sample and set to 500 samples for the overlapped window, giving a balanced and constant sample size for analysis. This overlapping technique is essential because it helps in the capture of any shifts and changes within the data, ensuring that no important information is lost throughout the processing step.

Feature extraction

Feature extraction is an important topic in signal processing and data analysis. It entails converting raw data into a collection of representative features that capture the essential data required for further analysis or classification tasks. Due to their widespread use in wearable EMG systems and their low computational cost, we chose simple time-domain features, which include min, max, mean absolute value, and root mean square [37]. This choice reduces run-time costs and increases the pipeline's suitability for future embedded or low-latency implementations. While more complicated features (entropy, spectral measurements, and time-frequency wavelet coefficients) might improve discriminability, they also raise processing costs and the possibility of overfitting on smaller datasets [37]. Therefore, we leave it to future studies to systematically evaluate these features. After the feature extraction, 70% of the extracted data is used for training and 30% for testing.

Classification and optimization

We used machine learning classifiers such as SVM, RF, KNN, and DT to classify the EMG hand gesture data. Because of the size of the dataset and the work's focus on lightweight models appropriate for low-latency embedded deployment, deep learning techniques were not used. Traditional machine learning models (KNN, SVM, RF, and DT) are easier to deploy on prosthetic devices and require less data and computation. We plan to explore deep models in future work when larger datasets or hardware resources are available. To optimize the classifiers, Bayesian optimizer using Gaussian process, grid search, and random search optimizers along with the classifiers are employed. All approaches seek to identify the optimum set of hyperparameters for a specific machine learning model that produces the highest performance. Grid Search exhaustingly tests all combinations within the predefined parameter space [12], while Random Search samples parameter combinations randomly, which is less computationally intensive but can skip optimal settings [13]. In contrast, Bayesian optimization is quite different from grid search and random search because it does not use brute force or random sampling but constructs a probabilistic model to guide the search for optimal hyperparameters. Specifically, it uses a Gaussian process model to predict the objective function (e.g., classification accuracy) from previous evaluations. Bayesian optimization strategically searches over hyperparameters using an acquisition function, which is unlike grid search, which exhaustively tests all possible values, and random search, which randomly samples hyperparameters. It strikes a balance between exploration (testing new regions of the hyperparameter space) and exploitation (exploring the already discovered promising area in the hyperparameter space) [11].

Model evaluation and performance reporting

The optimizers adjusted the tuning hyperparameters, as shown in Table *1*, to achieve the highest accuracy. We use two complimentary summaries to assess each classifier-optimizer combination. To evaluate central tendency and variability, we first calculate accuracy (mean ± standard deviation [SD]) over 25 stratified cross-validation (CV) runs (5 folds × 5 repetitions). Second, we record the optimal hyperparameters from each optimizer, retrain the model with them, and provide the tuned results for downstream analysis. Both standard and custom evaluation metrics were produced using the optimized/tuned models listed in Table *3*. The metrics included accuracy, precision, recall, F1-score, and one-vs-rest gesture-wise accuracy, adaptability and noise drop. Accuracy measures the model’s ability to make accurate predictions overall. Precision is the fraction of true positives among all predicted positives, and recall is the fraction of true positives among all actual positives. The F1-score is the harmonic mean of precision and recall. The performance specific to each class was assessed using one-vs-rest gesture accuracy and is computed as the per-class recall. To evaluate generalization across subjects, we did a leave-one-subject-out (LOSO) evaluation with the tuned configurations listed in Table 3 (Results). Each held-out subject was tested on the classifier after it had been retrained using the fixed best-hyperparameter values (no re-tuning) on the other subjects. Per-subject accuracy was calculated, and each tuned classifier summary was obtained as the mean and standard deviation across the 36 subjects. To evaluate robustness under non-ideal signal conditions, two metrics were defined. The first is adaptability, which indicates how well the model can perform even with test signals slightly varying to simulate electrode drift. The second parameter is the Noise Drop, which presents the reduction in the model accuracy when the Gaussian noise (sigma = 0.1) is applied to the testing signals, defined as:

$$\text{Noise Drop} = \frac{\text{Accuracy}_{\text{clean}} - \text{Accuracy}_{\text{noisy}}}{\text{Accuracy}_{\text{clean}}}$$

Classifiers	Tuning Hyperparameters
KNN	n_neighbors = range from 1 to 10
	p = {1, 2}
	leaf_size: {10, 15, 20, 25, 30, 35, 40, 45, 50}
	weights: {'uniform', 'distance'}
	algorithm: {'auto', 'ball_tree', 'kd_tree', 'brute'}
SVM	C = range from 10^{-6} to 10^{6}
	gamma = range from 10^{-6} to 10^{6}
DT	max_depth = {10, 20, 30, 40, 50, 60}
	min_samples_leaf = {1, 2, 4, 8, 16, 32}
	min_samples_split = {2, 5, 10, 20, 30, 40}
RF	n_estimators = {200}
	max_depth = {10, 20, 30, 40, 50, 60}
	min_samples_leaf = {1, 2, 4, 8, 16, 32}
	min_samples_split = {2, 5, 10, 20, 30, 40}

TABLE 1: Search space of tuning hyperparameters

Hyperparameters: For KNN, n_neighbors refer to the number of neighbors, p is the distance metric power, leaf_size indicates the tree leaf size, weights define the prediction weighting, and algorithm specifies the neighbor search method. For SVM, C represents the regularization parameter, and gamma is the kernel coefficient. For DT and RF, max_depth denotes the tree depth, min_samples_leaf is the minimum number of samples per leaf, and min_samples_split is the minimum number of samples required to split a node. In RF, n_estimators is the number of trees, fixed at 200.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

Results And Discussion

The following findings summarize the performance evaluations of all classifier-optimizer combinations, followed by statistical and comparative analysis. Table 2 compares the stability of classifier-optimizer combinations by reporting accuracy (mean ± SD) throughout the 25 repeated CV runs. The mean estimates predicted performance across different splits, while the SD represents run-to-run variability, indicating robustness to sample variations. KNN optimized with Bayesian search has the best mean accuracy (98.3 ± 0.1) and minimal variation, suggesting good stability. SVM and RF achieved comparable accuracy (95.2 ± 0.3 and 96.6 ± 0.3, respectively), but with significantly more variation.

Classifier	Grid Search	Random Search	Bayesian Optimization
KNN	98.2 ± 0.2	98.1 ± 0.3	98.3 ± 0.1
SVM	95.0 ± 0.4	94.7 ± 0.5	95.2 ± 0.3
DT	86.9 ± 1.0	86.1 ± 1.2	87.5 ± 0.8
RF	96.2 ± 0.4	95.5 ± 0.5	96.6 ± 0.3

TABLE 2: Classification accuracy (mean ± SD, %) for each classifier–optimizer pair (25 repeats of stratified 5 × 5 CV)

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest, SD = Standard Deviation, CV = Cross Validation

Table *3* lists the best hyperparameters found by each optimizer and the corresponding model accuracy after retraining. When optimized using a grid, random search, or Bayesian optimizer, KNN consistently obtained 98.4% accuracy, as shown in Table *3*. This is likely due to its inherent simplicity as a distance-based algorithm with few hyperparameters and well-separated classes in the dataset. SVM classifiers with grid search obtained accuracy of 95.4%, random search obtained an accuracy of 95.21%, and Bayesian optimization slightly improved the accuracy to 95.50%. DT models achieved accuracies ranging from 87.29% to 88.27% across various optimization strategies, whereas RF classifiers produced accuracies ranging from 96.57% to 96.87%, with Bayesian optimization yielding the best accuracy.

Classifier	Optimizer	Best Hyperparameters	Accuracy
KNN	Grid	n=1, p=1, leaf=10, weights=uniform, algo=auto	98.4%
	Random	n=2, p=1, leaf=35, weights=distance, algo=ball_tree	98.4%
	Bayesian	n=1, p=1, leaf=50, weights=distance, algo=auto	98.4%
SVM	Grid	C=100, gamma=0.215	95.4%
	Random	C=4.6415, gamma=0.215	95.2%
	Bayesian	C=440.77, gamma=0.224	95.5%
DT	Grid	depth=20, leaf=1, split=2	87.9%
	Random	depth=30, leaf=1, split=5	87.3%
	Bayesian	depth=40, leaf=1, split=2	88.3%
RF	Grid	n=200, depth=20, leaf=1, split=2	96.6%
	Random	n=200, depth=60, leaf=1, split=5	96.0%
	Bayesian	n=200, depth=60, leaf=1, split=2	96.9%

TABLE 3: Results after using Grid Search, Random Search, and Bayesian Search

Optimization methods include Grid Search (Grid), Random Search (Random), and Bayesian Optimization using a Gaussian process (Bayesian). For the definitions of all hyperparameters shown in this table, refer to Table 1.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

Moreover, the performance of each of the models was evaluated on standard classification metrics such as accuracy, F1-score, precision, and recall, as shown in Figure *4*. These metrics help in benchmarking every optimizer-classifier combination regarding its overall classification ability.

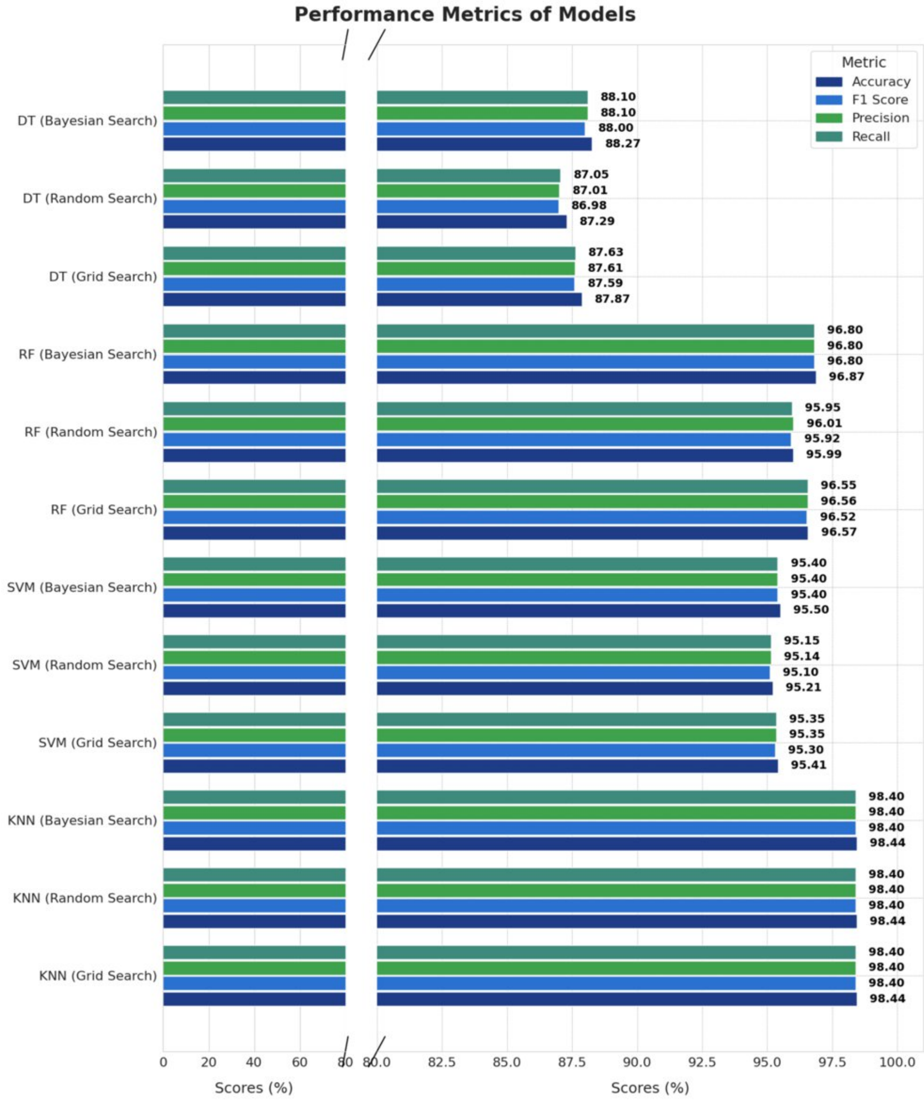

FIGURE 4: The performance matrices for each optimized model

Bar plots showing accuracy, precision, recall, and F1-score for four classifiers, including KNN (K-Nearest Neighbors), SVM (Support Vector Machine), DT (Decision Tree), and RF (Random Forest). Each is optimized with Bayesian optimization using a Gaussian process, Grid search, and Random search.

Further, in order to understand better the behavior of class-wise prediction, we visualized one-vs-rest accuracy of each gesture, as shown in Figure *5*. The per-gesture classification accuracy of all classifier-optimizer combinations is presented using this heatmap. As expected, KNN models are the best performing and are exceedingly accurate, with an overall accuracy of 99% and above in all six gestures among all optimizers. These results underscore the presence of classifier-varying results in classifying individual gestures and validate the gesture-wise robustness study.

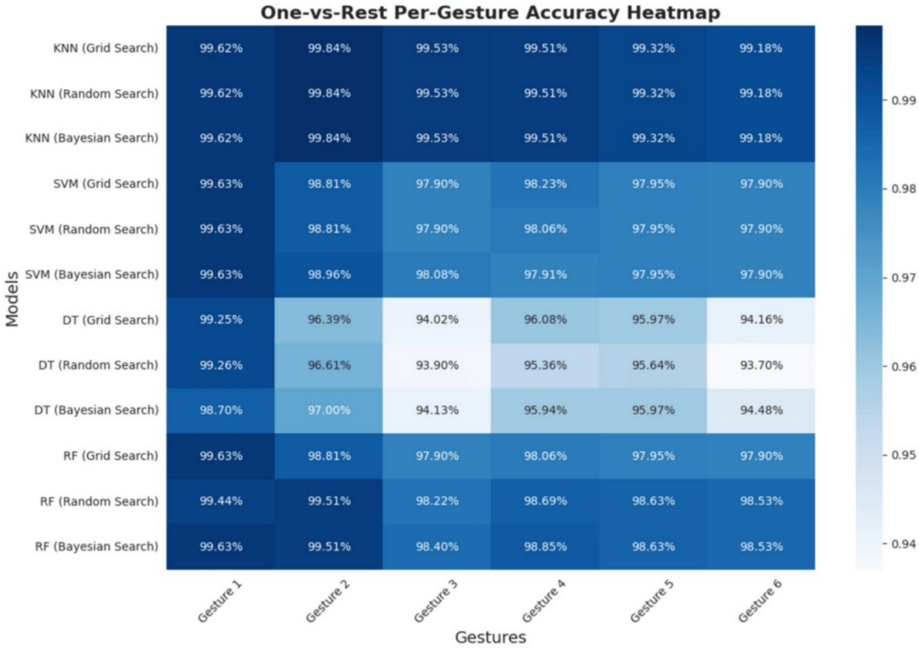

FIGURE 5: One-vs-rest per gesture accuracy heatmap

A heatmap showing gesture-wise classification accuracy using the one-vs-rest method. Each column represents a gesture, and each row represents a classifier–optimizer combination. Cells show accuracy for classifying each gesture against all others. High values (dark blue) indicate high accuracy.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

In order to further assess subject-level generalization, a LOSO assessment was performed with the fixed tuned hyperparameters of Table *3*. The classification accuracy mean ± SD for each classifier-optimizer combination is shown in Table *4* for all 36 subjects. KNN maintained the best overall performance with a mean LOSO accuracy of 98.38 ± 0.45% using Bayesian optimization, ahead of RF (96.71 ± 0.66%) and SVM (95.15 ± 0.72%). The DT models have the most inter-subject variability (SD > 1.3%), making them the least reliable. The low SD values in KNN and RF indicate good cross-subject stability, showing that the tuned configurations performed well when applied to subjects who did not participate in training.

Classifier	Grid Search	Random Search	Bayesian Optimization
KNN	98.21 ± 0.55	98.14 ± 0.48	98.38 ± 0.45
SVM	94.96 ± 0.86	94.78 ± 0.94	95.15 ± 0.72
DT	86.71 ± 1.65	86.14 ± 1.82	87.92 ± 1.38
RF	96.12 ± 0.78	95.62 ± 0.92	96.71 ± 0.66

TABLE 4: LOSO accuracy (mean ± SD, %) across 36 subjects for tuned models

Mean and standard deviation (SD) calculated with 36 subjects with leave-one-subject-out (LOSO) analysis. The average subject-level test accuracy is displayed in each cell, keeping tuned hyperparameters (Table 3) constant and retraining the classifier on the remaining 35 subjects.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

To properly examine real-world robustness, we used a multi-objective approach with three axes: (1) accuracy, (2) adaptation to simulated electrode drift, and (3) noise robustness. Table *5* shows the Pareto-optimal models that correspond to the best-performing optimizer for each classifier, identified by a non-dominated sorting method that evaluates trade-offs across all three objectives. In this approach, a model can be called Pareto-optimal if no other model achieves higher accuracy, adaptability, and noise robustness. Together, these metrics describe the model's ability for generalization and robustness to signal noise.

Classifier	Optimizer	Accuracy (%)	Adaptability (%)	Noise Drop (%)
KNN	Bayesian	98.4	98.9	0.1
RF	Bayesian	96.9	95.5	1.1
SVM	Bayesian	95.5	95.7	0.32
DT	Bayesian	88.3	89.4	5.7

TABLE 5: Best-performing optimizer for each classifier with corresponding tuned accuracy, adaptability, and noise drop values

Accuracy represents standard classification accuracy. Adaptability measures performance under simulated electrode drift. Noise Drop quantifies performance degradation when noise is added. Higher Adaptability (closer to 100) and lower Noise Drop are better. “Bayesian” refers to Bayesian optimization using a Gaussian process.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

In Figure 6, a 3D Pareto front view of evaluated models is shown. Among these, KNN with Bayesian optimization was the highest-performing model, with an accuracy of 98.4%, 98.99% adaptability, and only 0.10% performance deterioration by noise. These findings show that, whereas many models perform well in terms of accuracy alone, only a few retain performance under situations similar to electrode replacement and ambient EMG noise, highlighting the importance of Pareto analysis in robust classifier selection.68y6y7uik

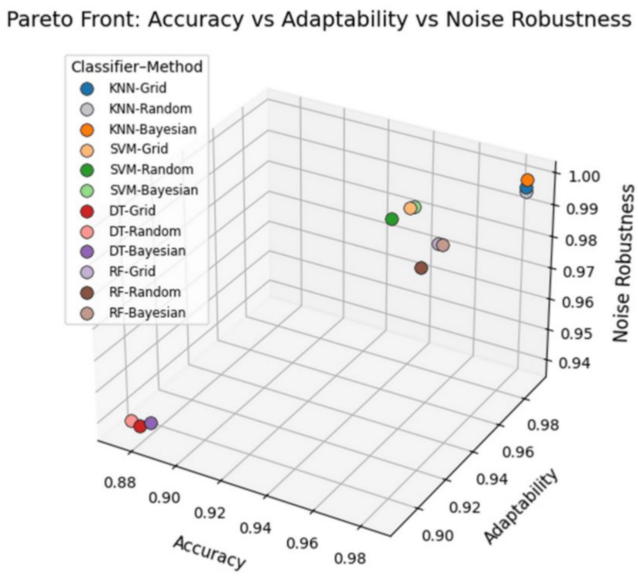

FIGURE 6: Pareto front visualization of all classifier–optimizer models across three objectives: accuracy, adaptability, and noise robustness

Three optimization strategies were evaluated: Grid Search (Grid), Random Search (Random), and Bayesian optimization using a Gaussian process (Bayesian). Each point represents a classifier-method combination plotted in a 3D space of accuracy, adaptability, and noise robustness.

KNN = K-Nearest Neighbors, SVM = Support Vector Machine, DT = Decision Tree, RF = Random Forest

According to Pareto-front analysis, the best model was defined as KNN with Bayesian optimization because of its compromise between accuracy, adoptability, and tolerance to noise. Figure 7 shows that the confusion matrices of the top KNN, RF, SVM, and DT models to demonstrate their classification behavior. KNN has the highest diagonal dominance, all gestures have recalls greater than 98%, and RF also has high per-class performance. On the other hand, SVM and DT exhibit more off-diagonal value, meaning more confusion between some gestures. This comparison also reveals why KNN came out as the overall best model.

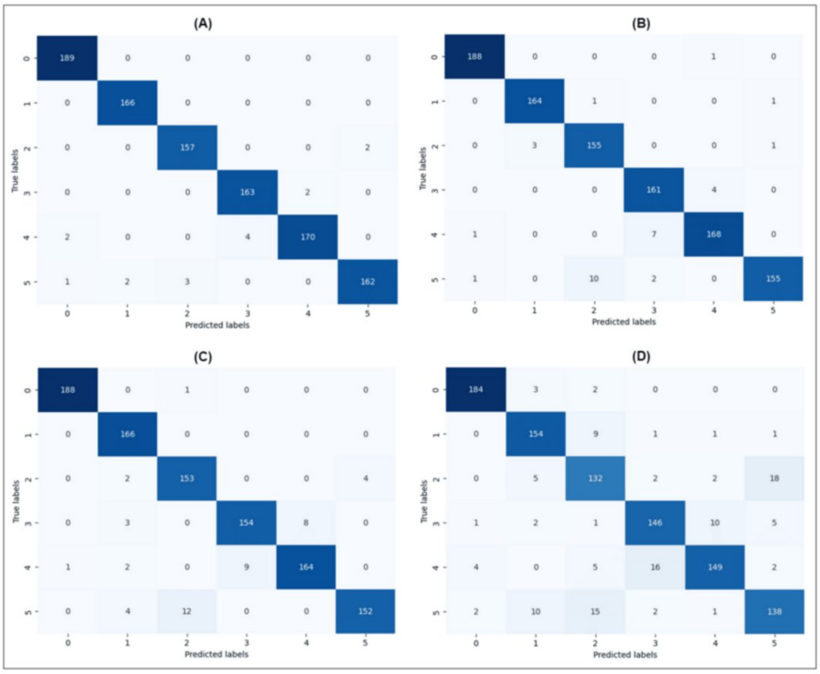

FIGURE 7: Confusion matrices of the models reported in Table 5, each representing the best-performing configuration obtained through Bayesian optimization: (A) K-Nearest Neighbor, (B) Random Forest, (C) Support Vector Machine, and (D) Decision Tree

Each matrix shows the number of samples classified for each class, where the x-axis represents the predicted labels and the y-axis represents the true labels.

A comprehensive discussion of the experimental results is provided below:

The results of the classifications from various classifiers and optimization strategies provide subtle insights into the effectiveness of various approaches. Accuracy, F1 score, precision, and recall are four key performance matrices and also confusion matrices that we used to assess the models in order to identify which classifier and optimizer would be best for gesture recognition. As shown in Figure *4*, grid search, random search, and Bayesian optimization for KNN scores for all performance metrics were consistently around 98.40% to 98.44%. KNN is a top performer here, but their consistent accuracy across all the optimization methods suggests that algorithm is inherently less sensitive to hyperparameter tuning. In contrast to SVM and DT, whose performance enhances considerably by hyperparameter optimization, KNN depends on distance metrics, and hence its performance remains stable with little adjustments. This observation shows that it was KNN's simplicity that made it efficient enough, and even simple optimizations were enough. Plus, class separability for the dataset should be highly clear, thereby reducing the need for a lot of interactions when tuning KNN compared to other models. SVM is also a good performer here because it managed to score high results for all performance metrics, which were in the range of 95.10% to 95.5%. Comparing all the optimization techniques used, there was a minor difference between them. Nevertheless, the Bayesian search exhibited slightly better results. This also implies that SVM can benefit from better hyperparameter tuning techniques, such as Bayesian search, which effectively searches for the best hyperparameter values. RF classification models seem to have provided reasonably good results, with performance matrices ranging from 95.9% to 96.87%, especially when grid search and Bayesian search were used. Another search algorithm that was used was random search, which also provided performances that were somewhat lower but still better. The fact that all of the RF models show high performance, but with the Bayesian optimizer, it obtained good results. When it comes to DT, their performance metrics range from 86.9% to 88.27%, and their results with Bayesian search surpass both random and grid searches, providing the accuracy of 88.27%, with grid search closely following at 87.87% and random search trailing slightly behind at 87.29%. These findings indicate that DTs are sensitive to hyperparameter adjustment, with Bayesian and grid searches outperforming. The adaptive nature of Bayesian optimization provides slight performance gains for SVM, DT, and RF classifiers, as it is able to better manage complex interactions between hyperparameters as opposed to the brute force approaches of grid and random search.

Furthermore, Figure *5* depicts the gesture-wise heatmap, which provides further insights into model performance. As seen in the heatmap, KNN is the most stable and performant model with consistently good gesture-wise accuracy across all optimization strategies. RF and SVM perform somewhat better when using Bayesian search, especially for complex gestures. Particularly for complex models like RF that mainly depend on finely tuned parameters, these enhancements support the finding that Bayesian optimization is the most efficient approach overall. Additionally, the LOSO results in Table *4* show the tuned models are strongly generalized to subjects that they have not seen, suggesting their use in a practical prosthetic or rehabilitation application where retraining per user is limited. With its simplicity and built-in distance-based adaptation, KNN's strong LOSO performance allows it to efficiently handle subject-wise EMG variations. SVM and RF models also generalize quite well with a little higher variability, which indicates that personalized calibration may further increase their stability. Conversely, the lower mean and larger variance of DT compared to all classifiers underline that the heterogeneous EMG signals can be poorly modeled by a single-tree model. Moreover, raw accuracy is significant, but models used in the real world must also have a tradeoff between accuracy and resistance to noise and sensitivity to signal drift. On our Pareto front analysis, KNN with Bayesian optimization produced the best trade-off. Other preferable alternatives were SVM and RF classifiers, which also emerged on the Pareto front. The outcomes have significance relating to the design of EMG-based prosthetic systems in which environmental noise and physiological changes are frequent. The confusion matrix of the Pareto-optimal model (Bayesian-optimized KNN) shows that the model is highly cross-subject-reliable, with nearly all the gestures being identified correctly. Compared to the other best-effective models in Figure *7*, KNN and RF have more defined diagonals and fewer misclassifications, whereas SVM has more misclassifications but less than DT. Notably, the confusion matrix of KNN validates that the classifier can retain good per-class performance, which is crucial to reliable control in prosthetics or wearable systems.

Recent EMG studies from 2024 and 2025 show 97.3% accuracy utilizing HD-EMG CNNs [17], 96.92% with federated edge learning [18], 92.7% with gated temporal convolutional networks [19], and 84% with time-domain feature fusion [30]. Also, based on the same dataset that we used, other studies have shown accuracies of 81.69% with RF [38], 96.43% with Naive Bayes [39], 82.81% with KNN, SVM, and RF [40], 90% with CNN [41], 97.43% with KNN [34], 98.4% with an ensemble bagged tree matches our best accuracy [42], and 98.6% with RF [43], which is very marginally higher than our best accuracy of 98.4%. However, none of these studies assessed robustness to electrode drift or signal noise. On the other hand, our Bayesian-optimized KNN maintains 98.9% accuracy under simulated drift, drops by only 0.10% under noise, and attains an overall accuracy of 98.4%. Our method avoids the high computational cost of deep models and is suitable for real-time embedded prosthetic systems by utilizing lightweight classifiers and basic features. This comparison shows that multi-objective assessment of adaptability and noise tolerance, which is mostly lacking in previous work, is crucial for practical EMG applications beyond simple accuracy.

Improving hyperparameter optimization not just helps improve the classification performance but also has direct implications for prosthesis design and related applications. This will result in a more responsive and adaptable control system for prosthetic devices. Machine learning algorithms have been studied to be highly accurate for EMG signal classification, correlating with improved prosthetic functionality [44]. Tables *2*-*5* and Figures *4*-*7* present the quantitative results, and we emphasize five simple, practical implications for implementing these models in real prosthetic systems:

• Fast inference: Use lightweight classifiers (such as KNN) on the prosthesis to make sure it responds rapidly. These models use fewer computing resources and provide reduced control latency.

• Tolerance to electrode shift: Prefer models with high adaptability scores (Table *5*). Because they can better withstand slight changes in electrode placement or muscle condition, users need fewer recalibrations.

• Robustness to noise: Select models with low noise-drop values (Table *5*). These models preserve performance even in the presence of distorted signals.

• Low calibration burden: Use tailored models that have minimal LOSO variability (Table *4*). These require less setup per user and make the system easier to use in clinical settings.

• Resource-aware selection: Choose Pareto-optimal models (Figure *6*) that balance accuracy, adaptability, and noise tolerance while fulfilling the device's memory and energy restrictions. If necessary, prune or quantize the model to minimize its size.

Considering our work limitations, this paper utilizes the data of 36 participants and only six various hand gestures. It is possible that performances may differ with more gestures, various other subjects, or other variations in the signals of the real world. The only basic features and standard machine learning classifiers were used. More benefits might come from sophisticated features or lightweight deep networks. These limitations suggest that future studies should validate our findings in real-time trials and on larger, more diverse datasets. In order to evaluate the clinical effectiveness of these improved models, future research will also involve incorporating them into actual prosthetic systems. Additionally, evaluating these improved classifiers on edge or embedded devices would yield important information about their power efficiency, latency, and real-time responsiveness. Such evaluations will help in ensuring that the suggested plan will be feasible to adopt on a device and use in wearable prosthetics.

Conclusions

The key findings of this study emphasize the choice of optimizer significantly affects classifier performance in EMG-based hand gesture recognition. For most classifiers, grid search is still a reasonable alternative, yes, but Bayesian optimization is a strong opponent, which consistently improved accuracy and robustness. Interestingly, KNN performed very well and stably across all optimizers. When it was paired with Bayesian tuning, it became the Pareto-optimal model, with 98.4% accuracy, 98.9% adaptability, and negligible noise degradation (0.10%). This work developed a multi-objective assessment approach that goes beyond accuracy alone to evaluate classifier dependability in practical settings. This approach includes noise robustness and adaptation to electrode drift. Since physiological and environmental variability are unavoidable in prosthetic control systems, these metrics are essential for their practical implementation.

The presented findings are clinically important because more adaptable and stable EMG systems will lead to enhanced control and usability of prosthetic devices in everyday practice. For clinical-grade EMG applications, future research will concentrate on applying these findings to lightweight deep learning architectures and real-time embedded systems.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Mohsin Ali Shah

Acquisition, analysis, or interpretation of data: Mohsin Ali Shah, Nisar Uddin

Drafting of the manuscript: Mohsin Ali Shah, Nisar Uddin

Critical review of the manuscript for important intellectual content: Mohsin Ali Shah

Supervision: Mohsin Ali Shah

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Haris M, Chakraborty P, Rao BV: EMG signal based finger movement recognition for prosthetic hand control. 2015 Communication, Control and Intelligent Systems (CCIS), Mathura, India. 2015, 194-198. 10.1109/CCIntelS.2015.7437907
2. Sultana A, Ahmed F, Alam MS: A systematic review on surface electromyography-based classification system for identifying hand and finger movements. Healthcare Analytics. 2023, 3:100126. 10.1016/j.health.2022.100126
3. Ni S, Al-qaness MAA, Hawbani A, Al-Alimi D, Abd Elaziz M, Ewees AA: A survey on hand gesture recognition based on surface electromyography: Fundamentals, methods, applications, challenges and future trends. Applied Soft Computing. 2024, 166:112235. 10.1016/j.asoc.2024.112235
4. Yasen M, Jusoh S: A systematic review on hand gesture recognition techniques, challenges and applications. PeerJ Computer Science. 2019, 5:e218. 10.7717/peerj-cs.218

5. Steele CM: Applications of EMG in Clinical and Sports Medicine. InTech, London; 2012. 10.5772/2349
6. Pessoa de Melo MS, Gomes da Silva Neto J, Natario Teixeira JMX, Fontes Da Gama AE, Teichrieb V: An EMG-based virtual reality application for motor rehabilitation. 2019 21st Symposium on Virtual and Augmented Reality (SVR), Rio de Janeiro, Brazil. 2019, 170-177. 10.1109/SVR.2019.00041
7. Ben Haj Amor A, El Ghoul O, Jemni M: Sign language recognition using the electromyographic signal: A systematic literature review. Sensors. 2023, 23:8343. 10.3390/s23198343
8. Kaur A: Wheelchair control for disabled patients using EMG/EOG based human machine interface: a review. Journal of Medical Engineering & Technology. 2020, 45:61-74. 10.1080/03091902.2020.1853838
9. Liao L, Li H, Shang W, Ma L: An empirical study of the impact of hyperparameter tuning and model optimization on the performance properties of deep neural networks. ACM Transactions on Software Engineering and Methodology. 2022, 31:1-40. 10.1145/3506695
10. Wang X, Ao D, Li L: Robust myoelectric pattern recognition methods for reducing users' calibration burden: challenges and future. Frontiers in Bioengineering and Biotechnology. 2024, 12:1329209. 10.3389/fbioe.2024.1329209
11. Snoek J, Larochelle H, Adams RP: Practical Bayesian optimization of machine learning algorithms. arXiv. 2012, 10.48550/arXiv.1206.2944
12. Liashchynskyi P, Liashchynskyi P: Grid search, random search, genetic algorithm: A big comparison for NAS. arXiv. 2019, 10.48550/arXiv.1912.06059
13. Bergstra J, Bengio Y: Random search for hyper-parameter optimization. The Journal of Machine Learning Research. 2012, 13:281-305.
14. Subasi A: Classification of EMG signals using PSO optimized SVM for diagnosis of neuromuscular disorders. Computers in Biology and Medicine. 2013, 43:576-586. 10.1016/j.compbiomed.2013.01.020
15. Little Flower X, Poonguzhali S: Boosting EMG classification: a hybrid NCA-driven evolutionary optimization approach for high accuracy and efficiency. Journal of Electrical Systems and Information Technology. 2025, 12:14. 10.1186/s43067-025-00205-0
16. Briouza S, Gritli H, Khraief N, Belghith S, Singh D: An auto-ANN for EMG classification for the motion recognition of the human hand. 2022 IEEE Information Technologies & Smart Industrial Systems (ITSIS), Paris, France. 2022, 1-6. 10.1109/ITSIS56166.2022.10118372
17. Tacca N, Dunlap C, Donegan SP, et al.: Wearable high-density EMG sleeve for complex hand gesture classification and continuous joint angle estimation. Scientific Reports. 2024, 14:18564. 10.1038/s41598-024-64458-x
18. Zafar MH, Moosavi SKR, Sanfilippo F: Federated learning-enhanced edge deep learning model for EMG-based gesture recognition in real-time human-robot interaction. IEEE Sensors Journal. 2025, 25:9139-9151. 10.1109/jsen.2025.3529841
19. Xie P, Xu M, Shen T, Chen J, Jiang G, Xiao J, Chen X: A channel-fused gated temporal convolutional network for EMG-based gesture recognition. Biomedical Signal Processing and Control. 2024, 95:106408. 10.1016/j.bspc.2024.106408
20. Briouza S, Gritli H, Khraief N, Belghith S, Singh D: Classification of sEMG biomedical signals for upper-limb and hand rehabilitation using a hybrid CNN-SVM architecture. 2022 International Conference on Data Analytics for Business and Industry (ICDABI), Sakhir, Bahrain. 2022, 467-471. 10.1109/ICDABI56818.2022.10041524
21. Briouza S, Gritli H, Khraief N, Belghith S, Singh D: EMG signal classification for human hand rehabilitation via two machine learning techniques: kNN and SVM. 2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia. 2022, 412-417. 10.1109/IC_ASET53395.2022.9765856
22. Briouza S, Gritli H, Khraief N, Belghith S, Singh D: Classification of sEMG biomedical signals for upper-limb rehabilitation using the random forest method. 2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia. 2022, 161-166. 10.1109/IC_ASET53395.2022.9765871
23. Wazir H, Ahmad J, Khan MA, Ullah Jan S, Khan FA, Khan MS: A hybrid neural network with smart skip connections for high-precision, low-latency EMG-based hand gesture recognition. 2024 26th International Multi-Topic Conference (INMIC), Karachi, Pakistan. 2024, 1-6. 10.1109/INMIC64792.2024.11004304
24. Xia Y, Qiu D, Zhang C, Liu J: sEMG-based gesture recognition using multi-stream adaptive CNNs with integrated residual modules. Frontiers in Bioengineering and Biotechnology. 2025, 13:1487020. 10.3389/fbioe.2025.1487020
25. Jiang B, Wu H, Xia Q, Xiao H, Peng B, Wang L, Zhao Y: An efficient surface electromyography-based gesture recognition algorithm based on multiscale fusion convolution and channel attention. Scientific Reports. 2024, 14:30867. 10.1038/s41598-024-81369-z
26. Khushaba RN, Al-Ani A, Al-Jumaily A: Orthogonal fuzzy neighborhood discriminant analysis for multifunction myoelectric hand control. IEEE Transactions on Biomedical Engineering. 2010, 57:1410-1419. 10.1109/tbme.2009.2039480
27. Naik GR, Al-Timemy AH, Nguyen HT: Transradial amputee gesture classification using an optimal number of sEMG sensors: an approach using ICA clustering. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2016, 24:837-846. 10.1109/tnsre.2015.2478138
28. Atzori M, Gijsberts A, Castellini C, et al.: Electromyography data for non-invasive naturally-controlled robotic hand prostheses. Scientific Data. 2014, 1:140053. 10.1038/sdata.2014.53
29. Gailey A, Artemiadis P, Santello M: Proof of concept of an online EMG-based decoding of hand postures and individual digit forces for prosthetic hand control. Frontiers in Neurology. 2017, 8:7. 10.3389/fneur.2017.00007
30. Mohapatra NR, Sahoo JP, Sahoo GK: Hand gesture classification using surface electromyography signals based on fusion of time domain features. 2024 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence (RAEEUCCI), Chennai, India. 2024, 1-6. 10.1109/RAEEUCCI61380.2024.10547968
31. Rumman U, Ferdousi A, Saha B: FORS-EMG: a novel sEMG dataset for hand gesture recognition across multiple forearm orientations. arXiv. 2024, 10.48550/arXiv.2409.07484
32. Gao J, Ma C, Wu D, Xu X, Wang S, Yao J: Recognition of human motion intentions based on Bayesian-optimized XGBOOST algorithm. Journal of Sensors. 2022, 2022:1-15. 10.1155/2022/3015645
33. Rezaee K, Khavari SF, Ansari M, Zare F, Alizadeh Roknabadi MH: Hand gestures classification of sEMG signals

based on BiLSTM-metaheuristic optimization and hybrid U-Net-MobileNetV2 encoder architecture. Scientific Reports. 2024, 14:31257. 10.1038/s41598-024-82676-1
34. Miah ASM, Hassan N, Maniruzzaman M, Asai N, Shin J: EMG-based hand gesture recognition through diverse domain feature enhancement and machine learning-based approach. arXiv. 2024, 10.48550/arXiv.2408.13723
35. Pareto front (Wikipedia). (2025). https://en.wikipedia.org/wiki/Pareto_front.
36. Krilova N, Kastalskiy I, Kazantsev V, Makarov V, Lobov S: EMG Data for Gestures [Dataset]. UCI Machine Learning Repository. (2018). https://doi.org/10.24432/C5ZP5C.
37. Phinyomark A, Khushaba RN, Scheme E: Feature extraction and selection for myoelectric control based on wearable EMG sensors. Sensors. 2018, 18:1615. 10.3390/s18051615
38. Shilaskar S, Bhatlawande S, Chavare R, Ingale A, Joshi R, Vaishale A: Human hand movement classification based on EMG signal using different feature extractor. Biomedical and Pharmacology Journal. 2024, 17:71-82. 10.13005/bpj/2835
39. Karapinar Senturk K, Bakay MS: Machine learning based hand gesture recognition via EMG data. ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal. 2021, 10:123-136.
40. Winursito A, Arifin F, Muslikhin M, Artanto H, Caryn FH: Performance analysis of EMG signal classification methods for hand gesture recognition in stroke rehabilitation. Elinvo (Electronics, Informatics, and Vocational Education). 2024, 8:264-274. 10.21831/elinvo.v8i2.76811
41. Gómez-Verde D, Esteban-Romero S, Gil-Martín M, San-Segundo R: Gesture recognition using electromyography and deep learning. Engineering Proceedings. 2024, 82:38. 10.3390/ecsa-11-20510
42. Sundaram, Sahana BC: Multivariate EMG signal based automated hand gestures recognition framework for elder care. International Journal of Precision Engineering and Manufacturing. 2025, 26:451-466. 10.1007/s12541-024-01116-2
43. Tallapragada D, Sagare V: Multichannel EMG-based gesture recognition utilizing advanced machine learning techniques: A random forest classifier for high-precision signal classification. World Journal of Advanced Research and Reviews. 2024, 24:323-332. 10.30574/wjarr.2024.24.2.3332
44. Yu G, Deng Z, Bao Z, Zhang Y, He B: Gesture classification in electromyography signals for real-time prosthetic hand control using a convolutional neural network-enhanced channel attention model. Bioengineering. 2023, 10:1324. 10.3390/bioengineering10111324