

Chapter 6

Application of Wavelet Decomposition and Ma-Chine Learning for the sEMG Signal Based Ges-Ture Recognition



Hala Rabih Fatayerji, Majed Saeed, Saeed Mian Qaisar, Asmaa Alqurashi, and Rabab Al Talib

Abstract The amputees throughout the world have limited access to the high-quality intelligent prostheses. The correct recognition of gestures is one of the most difficult tasks in the context of surface electromyography (sEMG) based prostheses development. This chapter shows a comparative examination of the several machine learning-based algorithms for the hand gestures identification. The first step in the process is the data extraction from the sEMG device, followed by the features extraction. Then, the two robust machine learning algorithms are applied to the extracted feature set to compare their prediction accuracy. The medium Gaussian Support Vector Machine (SVM) performs better under all conditions as compared to the K-nearest neighbor. Different parameters are used for the performance comparison which include F1 score, accuracy, precision and Kappa index. The proposed method of hand gesture recognition, based on sEMG, is thoroughly investigated and the results have shown a promising performance. In any case, the miscalculation during feature extraction can reduce the recognition precision. The profound learning technique are used to achieve a high precision. Therefore, the proposed design takes into account all aspects while processing the sEMG signal. The system secures a highest classification accuracy of 92.2% for the case of Gaussian SVM algorithm.

H. R. Fatayerji · S. Mian Qaisar (✉) · A. Alqurashi · R. Al Talib
Electrical and Computer Engineering Department, Effat University, Jeddah, Saudi Arabia
e-mail: sqaisar@effatuniversity.edu.sa

M. Saeed
Department of Urology, Sahiwal Teaching Hospital, Sahiwal, Pakistan

6.1 Introduction

The hands are the most versatile and dexterous component of the human body, capable of interacting with the world through a wide range of motion techniques such as touch, grasp, manipulation, and so on. Amputees all throughout the world are currently struggling to achieve anything more than the bare minimum with their prosthetics. In fact, according to the World Health Organization (WHO), only 1 in 10 individuals who need assistive devices, such as prostheses, have access to them currently due to the costly expense of the items as well as a lack of knowledge, availability, trained staff, policy, and funding [1]. Furthermore, in the varying circumstances, the system, and signal acquisition and detection should be effective. Therefore, the proposed system design identifies hand gestures by processing surface electromyography (sEMG) signals in order to contribute to the development of prosthetic hands.

The myoelectric signal (MES), also known as an electromyography (EMG) signal, obtained from the forearm skin surface gives vital information regarding neuromuscular processes. The complexity, non-linearity, and a considerable variance characterize the signals generated by EMG, which makes the signals difficult to interpret. Hence, before the usage of EMG signals to build a classification system for hand motions (pattern recognition and classification), there should be an identification process of the signals' attributes (features). In this situation, the pattern is represented by the temporal signal in an EMG signal. In most circumstances, the acquired signal may be defined in terms of its amplitude, frequency, and phase, all of which are time-variant. Electrical currents are formed in the muscle during a muscle contraction, showing neuromuscular activity, because muscle contraction and relaxation are constantly under the control of the neurological system. The nervous system and anatomical and physiological features of muscles interact to generate a complex EMG signal. While passing through various tissues, the EMG signal picks up noise. Moreover, if the EMG detector is placed near the skin's surface, it might gather signals from many motor organs at the same time, resulting in signal interference.

Clinical diagnosis and biological applications are the primary drivers of interest in EMG signal analysis, and evidently one of the key application areas is the treatment and rehabilitation of people with motor disabilities. EMG signals including MUAPs (Motor Unit Action Potentials) contain valuable information for the identification of diseases of the neuromuscular system. An understanding of EMG signals may be gained if the relevant algorithms and methods for their analysis are easily accessible. As a consequence, hardware implementations for different EMG signal-related applications can advance and be applied to invigorate the field's stagnation, but the high unpredictability of sEMG and the scarcity of available data restrict the deployment of gesture recognition technology [2].

The necessity for another generation of upper appendage prostheses prompted the development of a cost-effective prosthetic hand with an easy-to-use control interface. Advanced signal processing apparatuses and a programmed control

calculation have been incorporated into the myoelectric hand to enable the replacement of costly, precise actuators and sensors with less expensive components and to reduce the difficulty associated with the gadget's activation and navigation. High-level control is used to discern the client's goal to impel the hand model, which was meticulously created and built for this duty. Correspondingly, low-level control naturally plays out the problem of gaining a hold on it. Low-level input provides the control with both power and joint position data, whilst a vibrotactile feedback framework (undeniable level input) provides the client with a circuitous experience of touch [3]. When a prosthetic device is limited by electromyography signals produced by the muscles of the remaining of the amputated limb, six key perspectives must be considered to provide successful control:

- (a) To comprehend the characteristics of sEMG signals and how they are acquired.
- (b) To describe in detail the numerous advances produced.
- (c) To become acquainted with the various machine learning algorithms and their workings by evaluating how well they function.
- (d) Client control ought to be intuitive and need little mental effort.
- (e) There shouldn't be any discernible postponement in the reaction time of the framework.
- (f) To put into practice a paradigm that will benefit amputees from all societal strata.

6.2 Literature Review

6.2.1 Background

After centuries of many American Civil War losses, there was a large increase in demand for prosthetic limbs. Due to the limited functionality of the available limbs, many veterans began creating their own prostheses. One of the first amputees of the war, James Hanger, invented the "Hanger Limb." Samuel Decker was a pioneer in the development of modular limbs and also created his own mechanical arms. Decker's design has a spoon that is hooked to his artificial arms in recognition of the requirement for him to be able to carry out daily tasks with his prosthesis. Designs now needed to restore some of the amputees' prior abilities in addition to replacing the lost limb. A generation of young men was finally be able to live independent lives with artificial limbs. Specialized artificial limbs were a notion that the forerunners of prosthetic design had started to explore and around the 1900s, limb design grew more specialized and focused on purposes other than decoration [4]. Slowly, the fundamental idea behind surface EMG-based human-machine interfaces developed to be the use of machine learning techniques to transform sEMG data into controlled signals. The implementation of prosthetic hand control and high dependability of the human-machine interface have been made possible by the accuracy and adaptability of the information processing and classification algorithms.

The deep learning has lately drawn more interest of academicians for studying the detection of sEMG based patterns. By bypassing the tedious feature extraction and optimization procedures, it can automatically learn features of various abstract levels from a variety of input samples to achieve an end-to-end sEMG based gesture recognition [2]. The wavelet transform is the most commonly acknowledged approach for dissecting the EMG in the time-recurrence space. The organizing stage entails selecting the class to which the element vector retrieved from an EMG belongs. Support vector machines and neural networks are the most commonly used continuous models for characterization. A mix of decision trees, k-means clustering, and hidden Markov models is employed. In this case, a mix of support vector machines (SVMs) and hidden Markov models (HMMs) is applied. Signal acknowledgment frameworks have had to work constantly for a long time. The precision of these frameworks should be comparable to that of disconnected frameworks. A motion acknowledgment framework must be able to detect a move in less than 300 milliseconds in order to function constantly. This is equivalent to at least three motions per second. Furthermore, these frameworks are typically executed with limited computational resources. These requirements impose a requirement on the complexity of an acknowledgment model. As a result, the test is to design a continuous signal acknowledgment framework with a cheap computational expense and excellent performance. Although other machine learning techniques have been discussed in earlier research, the k-Nearest Neighbor (k-NN), SVM, and Artificial Neural Networks (ANNs) are the three classifiers that produce the best classification results [5].

6.2.2 Preprocessing for sEMG Based Gesture Recognition

The sEMG-based hand motion recognition technique is divided into several steps. The initial step is to acquire the raw signals. The number and location of electrodes must be determined based on the type of motion done in the experiment. Because each action is dependent on multiple muscles, most observations about motion identification for whole-hand movements have been recorded from four or more channels on the entire arm. However, in hand-posture research, the utilized muscles are primarily in the forearm, and thus the number of channels can be decreased. Although this may affect the accuracy quality, it can be improved by integrating feature data from multiple channels appropriately. The major sEMG processing blocks are shown in Fig. 6.1. A description of different blocks is provided in the following.

Active segment signals and information from inactive segment are both included in the continual process of collecting sEMG signals. Non-active segment information must be removed in order to increase the recognition model's precision and speed [6]. The sEMG signal is considered a noisy one, which indicates that the sEMG's probability distribution varies with time. The non-stationarity signal of sEMG can be reduced by filtering. Filtering is used to remove the noise and obtain

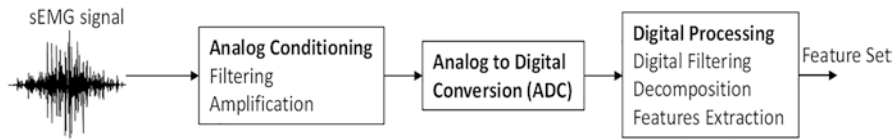
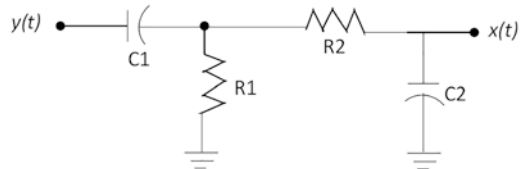


Fig. 6.1 The signal acquisition and preprocessing stages

Fig. 6.2 The analog band-pass filter

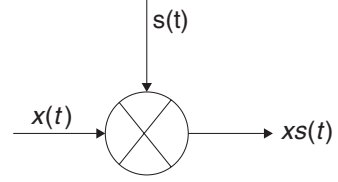


the most important information. Because roughly 95% of the sEMG signal strength is concentrated at 400–500 Hz, the sensor's lowest sample frequency must be more than double the sEMG's maximum frequency, according to the Nyquist–Shannon sampling theorem. In the meanwhile, the filter uses a low-pass filtering approach, or a moving average method, which might be considered a special low-pass filter. Some characteristics (such as MAV, ARV, or RMS) are computed using the moving average technique by windowing the signals and then averaging the features of all channels, or by computing the features of the average of all channels directly (Fig. 6.2).

Signal conditioning is the process of altering an analog signal such that it complies with the demands of the following step of further processing. Anti-aliasing filtering and voltage or current limiting are examples of signal conditioning used in analog-to-digital converter applications. The voltage and current, frequency, and electric charge are all acceptable input formats. Devices for signal conditioning can use a variety of outputs including voltage, current and frequency. Amplification, filtering, range matching, isolation, and other procedures are used in signal conditioning to prepare sensor output for further processing.

Analog signals are converted into digital signals by a device called an analog-to-digital converter (ADC). It converts a signal with continuous time and amplitude to one with discrete time and amplitude. The first step is band-pass filtering between the frequency range of [0.5; 150] Hz, given by Eq. (6.1) and shown in Fig. 6.1. In Eq. (6.1), $x(t)$ is the filtered version of signal $y(t)$ and $h(t)$ is the impulse response of this filter. In second step, the sampling is carried out, given by Eq. (6.2) and shown in Fig. 6.3. In Eq. (6.2), $x_s(t)$ is the sampled version of $x(t)$ and $s(t)$ is the sampling function. Onward, the input is quantized throughout the quantization process. We employed 500 Hz sampling rate and 12-bit resolution quantizer during the A/D conversion. The process of quantization involves condensing an infinite set of continuous values into a more manageable set of discrete values. It involves estimating the real-world values with a digital representation that restricts the precision and range of a value in the context of simulation and embedded computing. Rounding mistakes, underflow or overflow, and computational noise are all introduced by

Fig. 6.3 The sampling process



quantization. The behavior of an ideal system and estimated numerical behavior, thus, differ numerically. When choosing suitable data types for capturing real-world signals, one must take into account the precision, range, and scaling of the data type used to encode the signal as well as the non-linear cumulative effects of quantization on the numerical behavior of your algorithm. When mechanisms such as feedback loops are present, the cumulative influence of quantization is increased.

In embedded systems, the quantization is an essential step in accelerating inference while reducing memory and power usage. As a result, it may be installed on hardware with a reduced memory footprint, which frees up more RAM for control logic and extra algorithms. For a particular hardware architecture, examples of quantization features include integer processing, employing hardware accelerators, and fusing layers [7, 8]. The quantization process is given by Eq. (6.3). In Eq. (6.3), $x[n]$ is the digitized version of $x_s[n]$ and $Qe[n]$ is its corresponding quantization error. The upper bound on the $Qe[n]$ is posed by the quantum q and the relationship is given by Eq. (6.4) [9, 10].

$$x(t) = y(t) * h(t). \quad (6.1)$$

$$x_s(t) = x(t) \times s(t). \quad (6.2)$$

$$x[n] = x_s[n] + Qe[n]. \quad (6.3)$$

$$Q_e[n] \leq \pm \frac{q}{2}. \quad (6.4)$$

Each prolonged sEMG signal was divided into 6-second segments comprising 3000 samples to increase information size and the rectangular window approach was used to complete the division, with the windowing process is given by Eqs. (6.5) and (6.6) [11–13].

$$zw_n = x_n \times w_n. \quad (6.5)$$

$$zw_n = \sum_{\frac{-T}{2}}^{\frac{T}{2}} z_n. \quad (6.6)$$

Where, the digitized version of the sEMG band limited signal acquired from the examined dataset is denoted by x_n . Its segmented form is zw_n . w_n represents the vector of window function coefficients. It has a period of 6-seconds and 3000 magnitude 1 coefficients. The windowing method breaks up the longer sEMG signal into reduced chunks. Each segment is treated as a distinct instance. A total of 180 cases are examined, with 30 from each class.

The basic principle behind wavelets is to interpret data based on scale. The size with which we examine data is important in wavelet analysis. Wavelet algorithms operate on data at various sizes or resolutions. Gross characteristics can be seen if the signal is seen via a big “window.” Similarly, minor characteristics would be seen if the signal was viewed via a narrow “window.” Wavelet analysis produces the ability to view both the forest and the trees. Approximating functions that are cleanly confined in finite domains can be employed with wavelet analysis. Wavelets are ideal for estimating data with severe discontinuities. A wavelet function, known as an analyzing wavelet or mother wavelet is used in the wavelet analysis. If the best wavelets are not selected for the intended data or if the coefficients resolution is decreased below a certain threshold, then the data will not be correctly represented [7, 14]. The Eq. (6.7) can be used to express wavelet mathematically. Where s denotes the scaling operator and u is the translation operator.

$$\Psi(t) = \frac{1}{\sqrt{S}} \Psi((t-u)/s). \quad (6.7)$$

$$W_x(u,s) = \frac{1}{\sqrt{S}} \int_{-\infty}^{+\infty} x(t) \Psi^*((t-u)/s) dt. \quad (6.8)$$

The Eq. (6.8) demonstrates how to use the wavelet transform to break down a signal $x(t)$. When it comes to the decomposition of signals in terms of a base set of functions, the Discrete Wavelet Transform (DWT) is similar to the Fourier transform. A single parameter is used in Fourier transformations, and the basis set is made up of sines and cosines, whereas the expansion has only one parameter. A single “mother” wavelet is used to create the functions (wavelets) in the wavelet transform, with dilation and offsets matching to the two variables of the expansion being used to construct the functions.

The Discrete Time Wavelet Packet Decomposition (DWPD) is a wavelet transform that applies additional filters to the signal than the DWT. Wavelet packets are one-of-a-kind linear wavelet combinations. Many of the orthogonality, smoothness, and localization characteristics of their parent wavelets are retained in the bases they produce. The DWPD is a wavelet transform that applies additional filters to the signal than the DWT. Wavelet packets are one-of-a-kind linear wavelet combinations. Many of the orthogonality, smoothness, and localization characteristics of the parent wavelets are retained in the bases they produce. Each freshly created wavelet packet coefficient sequence serves as the root of its own analysis tree as the coefficients in the linear combinations are computed recursively. The WPD decomposes

both the detail and approximation coefficients. The WPD creates 2^n separate sets of coefficients (or nodes) for n degrees of decomposition, whereas the DWT produces $(n + 1)$ sets. However, because of the down sampling process, the total number of coefficients remains constant and there is no redundancy [15]. The process of computing the approximation and detailed coefficients is respectively given by Eqs. (6.9) and (6.10).

$$a_m = \sum_{k=1}^{K_g} x d_n \cdot g_{2n-k} \tag{6.9}$$

$$d_m = \sum_{k=1}^{K_g} x d_n \cdot h_{2n-k} \tag{6.10}$$

The wavelet decomposition, utilized to produce the sub-bands of the sEMG signal is shown in Fig. 6.4. The wavelet decomposition approach has been shown to lower the empirical risk in certain circumstances. Results from sEMG signal time series show that our method makes sense [16].

6.2.3 Feature Selection Techniques for sEMG Based Gesture Recognition

A feature is a functional requirement of a system. In general, the phrase feature includes both functional and non-functional criteria. The feature is an observable behavior of the system that the user might activate in this research. One of the most difficult aspects of programming is comprehending how a certain feature works. One must first locate the feature’s implementation in the code before they can fully understand it. In many cases, systems are composed of many modules, each of which contains tens or hundreds of lines of code. Most of the time, it’s difficult to tell wherein the source code a specific functionality is implemented. Original architects of the system may not be available, or their perspective may be skewed due to alterations made by others since the documentation’s creation (if any). Maintaining

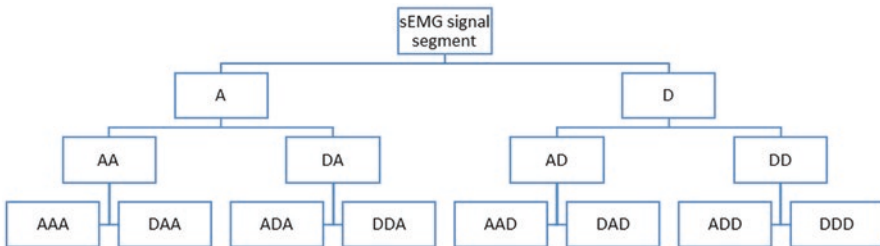


Fig. 6.4 The Wavelet Decomposition

a system introduces illogical modifications that weaken its overall structure. Every time something is altered in the system, it becomes more difficult to understand. One alternative is to reverse engineer the system from the ground up to get out of this rut, identifying all of its components and assigning functions to them. An incremental semiautomatic procedure uses established automatic ways for retrieving component data and validating them by hand before using them. Exhaustive procedures, on the other hand, are not economically viable. Fortunately, understanding the components that implement a certain set of functionalities is often sufficient [17].

For a system to be understood in its implementation of a specific feature, it must first be identified as to which computing units within the system are responsible for that characteristic. There are many instances where mapping features to source code is not well documented. The intended behavior of a system is described in abstract terms by its features. While concentrating solely on the implementation details and static structure of a system, reverse engineering methodologies neglect the dynamic relationships between the many pieces that only show when the system is in operation. By developing a model in which characteristics are tied to structural entities, we want to enhance the static and dynamic analyses that have already been performed. When it comes to dynamic analysis, there is a multitude of information available; nevertheless, this amount of knowledge creates a challenge in the analysis. In order to cope with it, we used Latent Semantic Indexing, an information retrieval technique that works with both documents and keywords. The objectives were to find linked features, as well as associated classes that participate in features in order to complete the task. For the text corpus, they used function calls from the traces; for the document corpus, they employed two mappings to documents: classes as documents as well as traces as documents.

The reduction of duplicate data in a data source is made possible through feature extraction. Data reduction expedites the learning and generalization stages of the machine learning process while also assisting the computer in building the model with less manual labor. Standard deviation, absolute minimum of n^{th} level approximation coefficients, highest absolute value of n^{th} level detail coefficients, mean of average absolute of all sub-bands, ratios of the mean of average absolute of succeeding sub-bands, root-mean-square value of time series, skewness & kurtosis of sub-band coefficients, absolute peak-to-peak difference of sub-band coefficients, and energies are all mined for each considered sub-band [18].

6.2.4 Machine Learning and Deep Learning Techniques for sEMG Based Gesture Recognition

In order to manage complicated activities autonomously or with little to no human participation, artificial intelligence (AI) is generally defined as any approach that allows computers to imitate or surpass human decision-making and mimic human behavior. As a result, it has connections to a wide range of tools and approaches and is concerned with a broad range of important issues, such as knowledge

representation, reasoning, learning, planning, perception, and communication. A computer might then use logical inference techniques to reason about hard-coded positions in formal languages, which were the main focus of early AI research. The knowledge base technique is another name for this. However, the paradigm has significant disadvantages since individuals fail to articulate all of their tacit knowledge necessary to do difficult tasks. Such constraints exist.

Artificial intelligence is divided into several subfields, including deep learning, machine learning, and neural networks. Deep learning, a type of machine learning, is divided into neural networks. Both deep learning and machine learning use distinctive learning techniques. The majority of the feature extraction process is automated using deep learning, eliminating the requirement for some manual human interaction and enabling the usage of larger data sets. Traditional, or “non-deep,” machine learning is taught by humans. Human experts create a collection of traits to recognize differences in data inputs, which frequently require the use of additional organized data to understand. Although they are not necessary, labeled datasets or supervised learning can help “deep” machine learning algorithms. Unstructured data may be ingested in its raw form, and it can automatically recognize the properties that set distinct data types apart. Data analysis, in contrast to machine learning, does not need human contact, allowing us to scale machine learning in more intriguing ways. Speech recognition, natural language processing, and computer vision have all advanced more quickly as a result of deep learning. Deep learning is a term used to describe the number of layers in a neural network. More than three layers of inputs and outputs make up a deep learning algorithm, sometimes referred to as a deep neural network. There are just two or three layers in a simple neural network [19].

According to ML, a computer program’s performance generally increases over time in relation to a range of tasks and performance indicators. It attempts to automate the process of developing analytical models in order to carry out cognitive tasks like object identification and language translation. This is accomplished by employing algorithms that continually learn from training data specific to the task at hand, giving computers the ability to identify intricate patterns and hidden insights without being explicitly trained. For high-dimensional data operations like classification, regression, and grouping, machine learning is incredibly helpful. It can assist in obtaining reliable and repeatable findings by learning from earlier calculations and seeing patterns in huge databases. Machine learning algorithms have succeeded in a number of fields as a result, including fraud detection, credit scoring, analysis of the next-best offer, audio and image identification, and natural language processing (NLP). On the basis of the problem and the data provided, three types of ML may be identified:

- (a) Supervised learning,
- (b) Unsupervised learning
- (c) Reinforcement learning

Using labeled datasets, supervised learning is the process of creating algorithms that can accurately categorize data or forecast outcomes. When new input data is

added, the model's weights are changed until it is well fitted. To make sure the model is not overfitting or underfitting, this is done as part of the cross validation stage. Organizations may solve a variety of complex real-world problems with the help of supervised learning, such as separating spam from email. In supervised learning, classification and regression techniques including Naive Bayes, Neural Networks, Linear Regression, Logistic Regression, Random Forest, Support Vector Machine (SVM), and others are utilized. The user trains the algorithm to provide a response based on a collection of known and labeled data.

Unsupervised machine learning analyzes and sorts unlabeled data sets using machine learning techniques; these algorithms find hidden patterns or data groupings. Because of its capacity to identify similarities and differences in data, it is the ideal choice for exploratory data analysis, cross-selling tactics, consumer segmentation, and picture and pattern recognition. Additionally, it is used in the dimensionality reduction process to reduce the number of features in a model; principal component analysis (PCA) and singular value decomposition are two typical techniques for this (SVD). There are several unsupervised learning techniques and clustering algorithms available, including neural networks, k-means clustering, probabilistic clustering, and others. The algorithms create answers from unlabeled and unknown data. Data scientists frequently employ unsupervised approaches to uncover patterns in fresh data sets, and they may build machine learning algorithms utilizing a variety of technologies and languages, as well as pre-built machine learning frameworks, to speed up the process [19, 20].

Deep learning is a subcategory of machine learning that employs both supervised and unsupervised learning techniques. It is based on the representation learning subfield of machine learning theory (or feature learning). Artificial neural networks (ANN), also known as deep learning neural networks, mimic the workings of the human brain by using data inputs, weights, and bias. Together, these elements describe, categorize, and identify data items. Each layer of deep neural networks, which are made up of several interconnected ones, improves and fine-tunes categorization or prediction. A network's transmission of calculations is referred to as forward propagation. The layers of a deep neural network that are visible are the input and output layers. The layers of a deep neural network that are visible are the input and output layers. Before producing the final prediction or classification in the output layer, the deep learning model processes data in the input layer.

Deep learning models provide results quicker than traditional machine learning approaches because they employ a hierarchical learning process to extract high-level, complicated abstractions as data representations. In other words, rather of requiring the data scientist to select the important attributes manually, a deep learning model will learn them. Backpropagation is a different approach that uses methods like gradient descent to produce prediction errors before altering the weights and biases of the function by repeatedly going back through the layers to train the model. Forward and back-propagation function in tandem to allow a neural network to foresee and correct for errors. Deep learning models provide results quicker than traditional machine learning approaches because they employ a hierarchical learning process to extract high-level, complicated abstractions as data representations.

In other words, rather of requiring the data scientist to select the important attributes manually, a deep learning model will learn them. Backpropagation is a different approach that uses tools like gradient descent to produce prediction errors before changing the function's weights and biases by repeatedly going back through the layers to train the model. Forward and back-propagation function in tandem to allow a neural network to foresee and correct for errors. The algorithm's accuracy improves with time, and the "deep" in deep learning refers to the several layers used in deep learning models:

The completion of tasks like object detection and identification is made possible by convolutional neural networks (CNNs), which are extensively employed in computer vision and image classification applications. CNNs are able to recognize patterns and discriminate between properties in an image. CNN can be made up of numerous layers of models, each accepting input from the previous layer, processing it, and then passing it on to the next layer in a daisy-chain pattern. Recurrent neural networks (RNNs), on the other hand, are often utilized in natural language and speech recognition applications because they utilize sequential or time series data [20, 21].

6.3 Methodology

Figure 6.5 displays the designed system block diagram. The following describes several processing phases with materials and methods.

6.3.1 Dataset

The dataset came from a research that involves frequently and freely grasping of different items [22]. The individuals were given complete control over the speed and force of grasping. The six motions, shown in Fig. 6.6, were asked to be repeated by five healthy volunteers between the ages of 20 and 22. Figure 6.7 displays the surface plots of feature sets that were retrieved from examples of various categories. For each fundamental movement, the experiment was repeated 30 times with the subject performing each one for 6 seconds. 180 sEMG signals were therefore acquired for each subject.

In addition to being non-invasive, repeating patterns, and capable of categorizing signals in real time, sEMG has a wide range of applications, including gesture recognition, prosthesis development, and human-computer interfaces. The sEMG signals in this dataset can also be used to enhance other datasets for more accurate categorization of similar signals. There are 16 recorded EMG signals, each lasting 70 seconds, in the sEMG database of objects gripping activities. The signals were gathered from a healthy person. Six tasks were offered to the participant: spherical, palmar, tiny tools, lateral, cylindrical and hook gestures (cf. Figure 6.6) [22, 23].

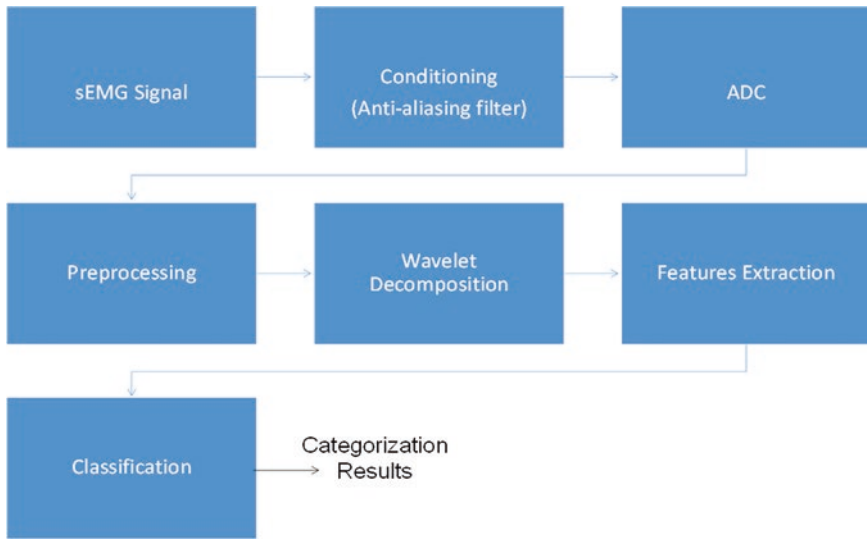


Fig. 6.5 The proposed system block diagram

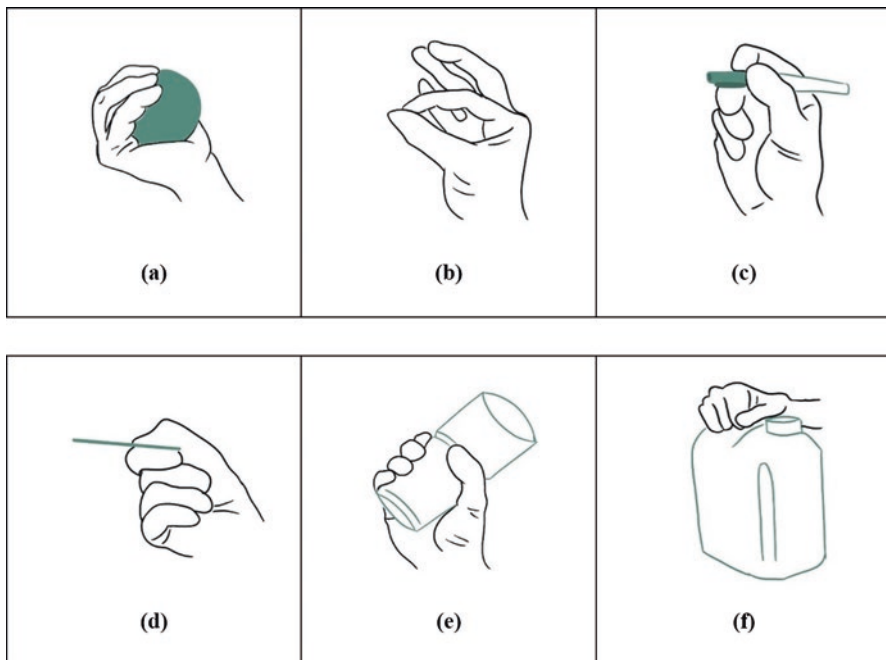


Fig. 6.6 (a) Spherical Gesture, (b) Tiny Tools Gesture, (c) Palmar (Grip) Gesture, (d) Lateral Gesture, (e) Cylindrical Gesture, and (f) Hook Gesture

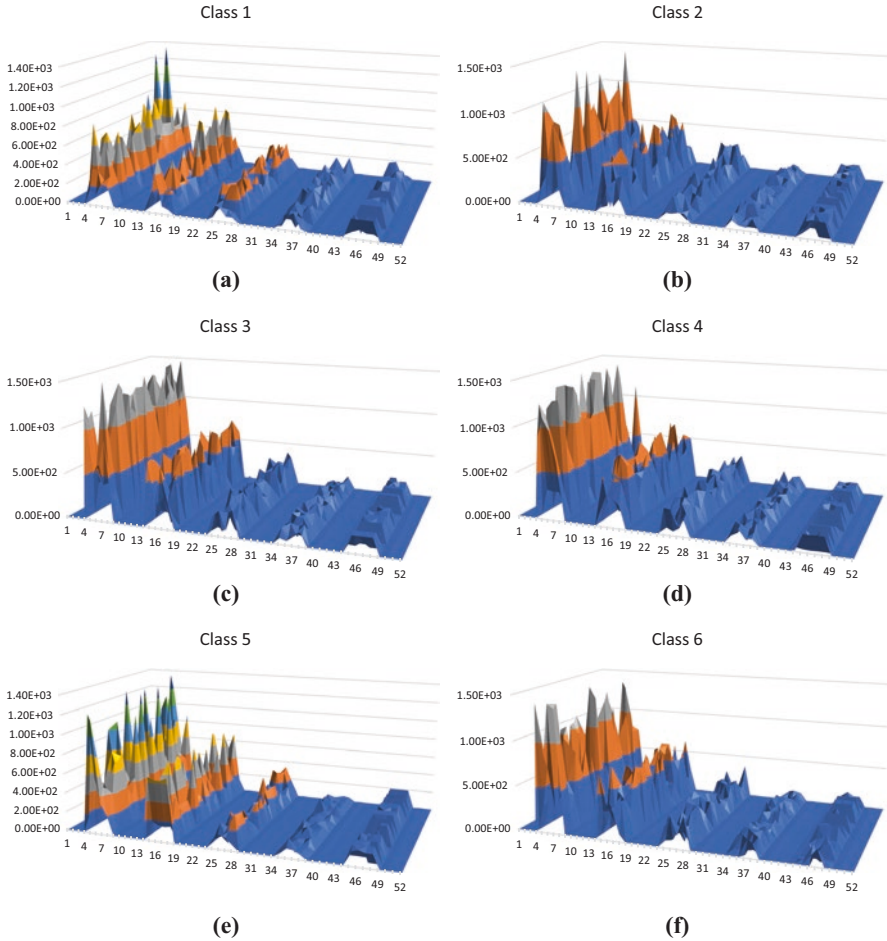


Fig. 6.7 (a) Class 1: Spherical, (b) Class 2: Tip, (c) Class 3: Cylindrical, (d) Class 4: Palmar, (e) Class 5: Lateral, (f) Class 6: Hook. The x-axis is presenting the number of attributes and y-axis is presenting their corresponding magnitudes

6.3.2 Machine Learning Algorithms

The classifier is an algorithm for performing the categorization tasks. It learns from the labeled dataset and onward the trained classifier may then be used to classify unknown documents or nodes based on the samples that were passed through the classifier to learn what makes a specific class where some parameters are set to better understand the status of the signal. Optimizer options are hyper-parameter options deactivated and all features used in the model before PCA are set to the default of “Gaussian Naive Bayes” in Model Type Preset. Gaussian is the distribution name for numerical predictors. Multivariate multinomial distribution is the name of the distribution used for categorical predictors (MVMN). The default

choice for the feature selection and cost matrix is MVMN. Fine KNN, Euclidean distance metric, weight equality, and standardize data are all true. Hyper-parameter options are deactivated in the optimizer settings, and feature selection is enabled for the model type preset. PCA is disabled and the Misclassification Cost Matrix is set as the default for all features in the model prior to PCA. Our method considers K of these data points, which is the predefined number. Therefore, the distance metric and K value of the KNN algorithm are crucial elements to take into consideration. As far as distance measurements go, there's no better option than using Euclidean distance. In addition to this, you have the option of using the Hamming, Manhattan, or even the Minkowski distance. The training dataset's data points are all taken into account when predicting the class or continuous value of a new data point. Use feature space, class labels, or continuous values to find the "K" Nearest Neighbors of new data points. In Discriminant Linear All features used in the model prior to PCA are selected, and PCA is deactivated. The Misclassification Costs Matrix is set to Default. The Model Type Preset is Linear Discriminant, and the Covariance Structure is full. In the SVM Model Type Preset, the Gaussian Kernel Function, the 7.2-scale kernel, the one-level box constraint, and the One-vs-One (OvO) multiclass method are all set to the default values of medium. Settings for hyper-parameters are disabled; All features used in the model before to PCA are referred to as feature selection; The Misclassification Cost Matrix has a True default value since PCA is deactivated. There are 30 learner types and 26 subspace dimensions in Ensemble Classifiers' model type pre-sets, and hyper-parameter choices have been deactivated in the optimizer settings. There are no PCA or misclassification cost matrices since the model uses all characteristics before PCA.

6.3.2.1 Support Vector Machine Classifier (SVM)

A sparse kernel decision machine, the SVM approach builds its learning model without taking posterior probabilities into account. SVM provides a systematic solution to machine learning problems because to its mathematical foundation in statistical learning theory. Frequently used for classification, regression, novelty detection, and feature reduction problems, SVM develops a solution by using a subset of the training input.

When a program is executing, it generates new parameter values. Preventative maintenance can save a lot of money in the long run if the engine begins to show signs of failure early on. In order to solve the problem that the diagnostic model's generalization ability decreases due to the motor's variable operating circumstances, this research proposed a rolling application bearing cross-domain defect detection strategy based on a medium Gaussian SVM. End-to-end diagnostics is made possible using only the original signal as an input. To evaluate a model, this approach requires prior knowledge of the label for the target domain in order to achieve supervised domain adaptation.

The SVM approach creates its learning model without taking posterior probabilities into account. It is a sparse kernel decision machine. Due to its mathematical

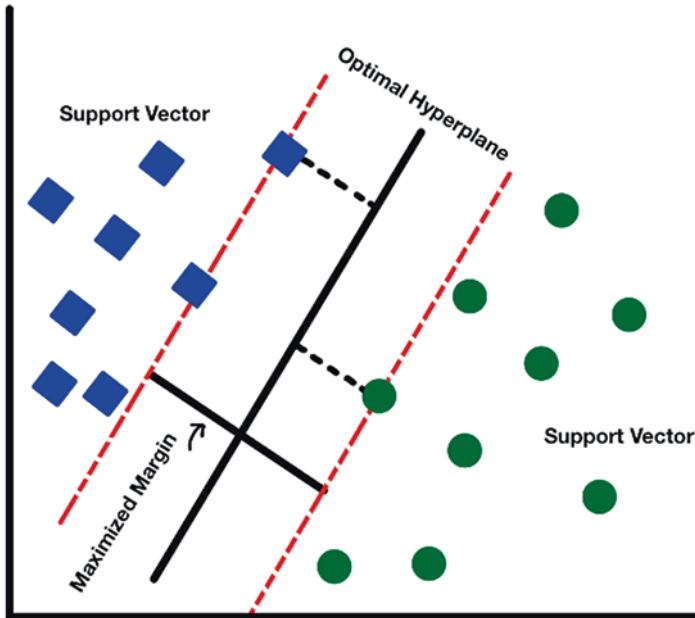


Fig. 6.8 Support vector machine

basis in statistical learning theory, SVM provides a systematic solution to machine learning problems. SVM is often used for classification, regression, novelty detection, and feature reduction problems and provides a solution by using a subset of the training input. These are their two main advantages (in the thousands). This method is perfect for problems involving text classification when a dataset of a few thousand tagged samples is the norm (Fig. 6.8).

6.3.2.2 K-Nearest Neighbor (KNN)

Using no previous knowledge of the original dataset, the KNN is a nonparametric classification technique. It is renowned for both its efficiency and ease of usage. The class of the unlabeled data can be predicted because the data points in a labelled training dataset are divided into multiple classes. Although this classifier is straightforward, the 'K' value is crucial for identifying unlabeled data. The term "k nearest neighbor" refers to the ability to repeatedly run the classifier with various values to determine which one produces the best results.

Automated model parameter estimation and manual setting of model hyper parameters are used to estimate model parameters. As the components of machine learning that need to be manually set and tweaked, model hyper parameters are sometimes referred to as "parameters." The K Nearest Neighbors operates in this manner. The nearest neighbors of our new data point are the data points that are

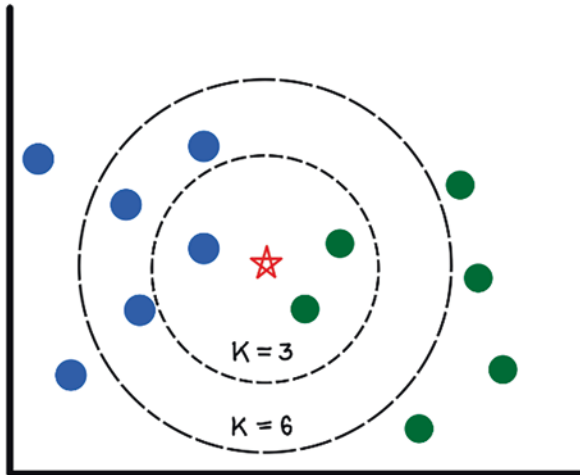


Fig. 6.9 K-nearest neighbor

separated from it by the least feature space. Our approach considers K of these data points, which is a fixed quantity. The distance metric and K value of the KNN algorithm are therefore important parameters. The Euclidean distance is the distance unit that is used the most frequently. There are also the Minkowski and Hamming distances, as well as the Manhattan and Manhattan distances. When determining the class or continuous value of a new data point, the training dataset's whole collection of data points is considered. Finds the K nearest neighbors of new data points by searching feature space, class labels, or continuous values (Fig. 6.9).

6.3.3 Evaluation Measures

6.3.3.1 Accuracy

According to [24], the disarray framework concept is used to evaluate the classifier's demonstration. The total number of predictions made to determine classification accuracy divides the total number of accurate predictions given a dataset. Accuracy is insufficient as a performance metric for imbalanced classification problems. This is mostly due to the fact that the dominant class(es) will exceed the minority class(es), which implies that even untrained models can get accuracy scores of 90% or 999%, depending on how severe the class imbalance is. There are four categories for each administered class. Focusing on the classes of lateral and hook gestures, we are defining:

Lateral and hook gestures

- True Positive (TP): How frequently does the characterization computation predict “lateral” even when “lateral” is the true class?
- False Positive (FP): How frequently does the categorization computation predict “lateral” even when the real class is “hook”? Also known as a “Type I Error”.
- False Negative (FN): How often does the order computation predict, “hook” when the real class is “lateral”? Also known as a “Type II Error”
- How often does the arrangement computation predict “hook” when the true class is “palmar”? [24].

The accuracy is the range of real orders that can range from 0 to 1, with 1 representing the best accuracy result as given in Eq. (6.11).

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \times 100\%. \quad (6.11)$$

6.3.3.2 Precision

Another statistical measure is called precision. It counts the number of accurate positive forecasts. Precision calculates the accuracy for the minority class as a result. It is determined by dividing the total expected number of positive occurrences by the number of accurately predicted positive cases. When the TNs and TPs classifications are appropriate, Eq. (6.12) may be used to quantitatively describe this measure. Findings from categorization that are FPs or FNs are wrong.

$$Precision = \frac{TP}{(TP + FP)}. \quad (6.12)$$

6.3.3.3 Specificity

As indicated in Eq. (6.13), the percentage of accurately detected adverse events is known as specificity.

$$Specificity = \frac{TN}{(TN + FP)}. \quad (6.13)$$

6.3.3.4 Recall

The recall, as it is described in Eq. (6.14), is a measure that counts the actual positive predictions that were made as opposed to all possible positive predictions. Recall takes into account all positive predictions, as opposed to accuracy, which

only takes into account the right positive predictions among all positive predictions. Recall in this method indicates the coverage of the positive class [24].

$$Recall = \frac{TP}{(TP + FN)}. \quad (6.14)$$

6.3.3.5 F-Score

The F-score evaluates the precision of a model on a certain dataset. It is used to assess algorithms that categorize occurrences as either “positive” or “negative,” or in between. A statistic for assessing information retrieval systems is the F-score. It is possible to adjust the F-score to emphasize accuracy over recall or the opposite. Equation represents the harmonic mean of accuracy and recall, which is the classic F1 score Eq. (6.15) [25].

$$F = \frac{2 * precision * recall}{precision + recall}. \quad (6.15)$$

6.3.3.6 Kappa Statistics

Cohen’s K-coefficient, which measures inter-rater agreement, measures the degree of agreement between two variables; hence, kappa most frequently deals with data that is the result of a judgment rather than a measurement. The likelihood of agreement is compared by Kappa to what may be expected if the ratings were independent. Kappa is another means of conveying the classifier’s accuracy [26]. Conditions, Eqs. (6.16) to (6.18), can be used to calculate Kappa.

$$kappa = 1 - \frac{1 - p_o}{1 - p_e}. \quad (6.16)$$

$$p_o = \frac{(TP + TN)}{(TP + TN + FP + FN)}. \quad (6.17)$$

$$p_e = \frac{(TP + TN)(TP + FN) + (FP + TN)(FP + FN)}{(TP + TN + FP + FN)^2}. \quad (6.18)$$

6.4 Results and Discussion

The performance of the created method is assessed using the six hand-gesture characteristics. Each gesture was made by the participants 30 times for a total of six seconds each gesture. The recordings were made with a conversion resolution of 12 bits and a sample rate of 500 Hz. The proposed windowing method is used to segment the ADC output. The maximum segment length is three seconds. To provide the qualities of each instance, features from each segment are extracted and combined. The extracted feature set is then processed using ML-based classifiers. The 10-CV approach is used to evaluate performance. Tables - describe the findings obtained for SVM and KNN, respectively.

The confusion matrices, obtained for the case of each hand gesture are outlined in the following Tables 6.1, 6.2, 6.3, 6.4, 6.5, and 6.6.

The evaluation measures for the case of SVM classifier are outlined in Table 6.7.

In Table 6.7, the spherical gesture (C1) shows the highest values of evaluation indices and the best AUC graph in the prediction parameters of all classes of gestures. When the SVM classification technique is used, it is the simplest to discriminate between maneuvers and the rest of the movements. All six classifiers evaluated had an average recall of 92.2% and an average AUC value of 99.0%, with the Medium Gaussian SVM coming out on top (Fig. 6.10)

The evaluation measures for the case of KNN classifier are outlined in Table 6.8.

In Table 6.8, the measures for the Fine KNN algorithm show the lowest outcome in overall accuracy (70.71%). Based on a confusion matrix and a prediction graph, it has the lowest prediction outcomes. Compared to other algorithms, the Weighted KNN method achieved 94.1 percent accuracy and took just 0.97885 Sec. to run. On the other hand, the classification accuracy of Cosine KNN was the lowest at 81.3%. Cubic KNN, on the other hand, took the longest to train at 30.441 seconds. According to the selection of two attributes, the prediction model presents the predicted with accurate and wrong predictions in the X and Y axis. The Euclidian distance measure, equal distance weight, and a default number of neighbors of 10 are all part of the Medium KNN. With the default parameters, this method has a 91.6% accuracy.

Table 6.1 Class 1 Dataset Records


Gesture	Input (Dataset)
	Spherical gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 3 FPs: 7 FNs: 0 TNs: 110	Medium Gaussian SVM TPs: 29 FPs: 0 FNs: 1 TNs: 137

Table 6.2 Class 2 Dataset Records



Gesture	Input (Dataset)
	Tiny tools gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 28 FPs: 0 FNs: 2 TNs: 112	Medium Gaussian SVM TPs: 30 FPs: 1 FNs: 0 TNs: 136

Table 6.3 Class 3 Dataset Records

Gesture	Input (Dataset)
	Palmar gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 19 FPs: 7 FNs: 11 TNs: 121	Medium Gaussian SVM TPs: 22 FPs: 3 FNs: 8 TNs: 144

It employs the Euclidian distance metric, equal weight of the distance, and a default of 100 neighbors as its default settings for coarse KNN. Based on the default parameters, this algorithm's Accuracy is 92.3 percent. In its computation, the cosine KNN uses a cosine distance metric, equal distance weight, and a de-fault number of neighbors of 10. The accuracy of this method is 81.3% with the default parameters. The Cubic KNN method makes use of an initial set of 10 neighbors and an equal distance weight. This algorithm's accuracy with default parameters is 93.5%. WKNN employs Euclidean distance metrics, the square of squared inverse distance weighted by 10 neighbors, and a default number of neighbors. This algorithm is accurate only 81.1% of the time (Fig. 6.11).

The accuracy is 77.8%, total misclassification costs are 40, prediction speed is ~3800 obs/sec, and training time is 0.8864 seconds in the results of the KNN simulation. Model type is acceptable, KNN number of neighbors is one, distance metric is Euclidean, distance weight is equal, standardize data is true, hyper-parameter options are disabled in the optimizer options, all features used in the model are selected, PCA is disabled before misclassification cost analysis, and the default

Table 6.4 Class 4 Dataset Records



Gesture	Input (Dataset)
	Lateral gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 23 FPs: 9 FNs: 7 TNs: 117	Medium Gaussian SVM TPs: 29 FPs: 0 FNs: 1 TNs: 137

Table 6.5 Class 5 Dataset Records

Gesture	Input (Dataset)
	Cylindrical gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 24 FPs: 3 FNs: 6 TNs: 116	Medium Gaussian SVM TPs: 29 FPs: 2 FNs: 1 TNs: 137

misclassification cost matrix is used. On the other hand, the SVM's accuracy is 92.2%, the cost of misclassification as a whole is 14, the prediction speed is ~3400 obs/sec, the training time is 0.77719 sec, the model type is Medium Gaussian SVM, the Kernel Function is Gaussian, the Kernel Scale is 7.2, and the Box Constraint Level is 1. One-to-one standardized data is utilized in the multiclass method, hyper parameter choices in the optimizer are deactivated, all features used in the model are selected, PCA is turned off before PCA, and the default misclassification cost matrix is used.

The event-driven tools are beneficial in terms of the computational effectiveness, processing activity and power consumption reduction and real-time compression [27–29]. The feasibility of incorporating these tools in the suggested method can be investigated in future.

Table 6.6 Class 6 Dataset Records


Gesture	Input (Dataset)
	Hook gesture 30 times for the duration of 6 seconds.
Result	
Fine KNN TPs: 16 FPs: 14 FNs: 14 TNs: 124	Medium Gaussian SVM TPs: 27 FPs: 8 FNs: 3 TNs: 139

Table 6.7 Prediction evaluations of SVM

RF	Accuracy	Precision	Recall	Specificity	F1	Kappa	AUC
C1	0.994	1.000	0.967	1.000	0.983	0.993	1.00
C2	0.994	0.967	1.000	0.992	0.983	0.993	1.00
C3	0.937	0.880	0.733	0.979	0.80	0.921	0.97
C4	0.994	1.000	0.967	1.000	0.983	0.993	1.00
C5	0.982	0.935	0.967	0.985	0.950	0.978	1.00
C6	0.937	0.771	0.900	0.945	0.830	0.921	0.97
Avg	0.973	0.925	0.922	0.983	0.921	0.966	0.99

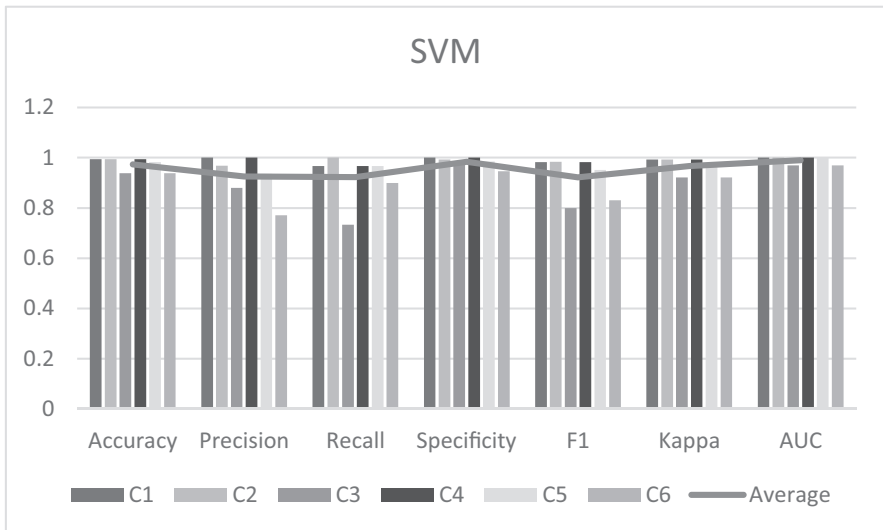


Fig. 6.10 Comparison of Gaussian SVM Prediction Parameters

Table 6.8 Prediction evaluations of KNN

RF	Accuracy	Precision	Recall	Specificity	F1	Kappa	AUC
C1	0.952	0.811	1.000	0.940	0.895	0.938	0.98
C2	0.986	1.000	0.933	1.000	0.965	0.982	0.97
C3	0.886	0.731	0.633	0.945	0.678	0.846	0.79
C4	0.897	0.719	0.767	0.928	0.742	0.862	0.85
C5	0.939	0.889	0.800	0.975	0.842	0.921	0.89
C6	0.833	0.533	0.533	0.899	0.533	0.767	0.72
Avg	0.916	0.780	0.778	0.948	0.776	0.886	0.87

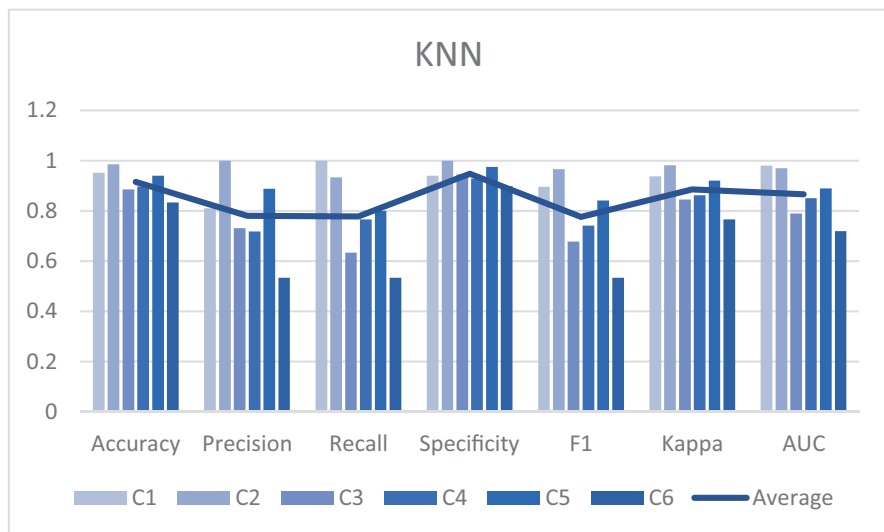


Fig. 6.11 Comparison of KNN Prediction Parameters

6.5 Conclusion

This chapter describes a contemporary automated system that uses sEMG signals to identify hand gestures. The sEMG signals are one of the most often utilized biological signals for predicting the upper limb movement intentions. Turning the sEMG signals to useful control signals frequently necessitates a large amount of computational power and sophisticated techniques. This chapter compares the performance of k-Nearest Neighbor and Support Vector Machine techniques for hand gesture detection based on the processing of sEMG signals. The first stage in this method is to capture the signal from the skin’s surface, followed by conditioning, segmentation, and feature extraction. The feature extraction highlights the needed characteristics from the data to recognize the gesture. Following that, the k-Nearest Neighbor and Support Vector Machine techniques were applied on the mined feature set. The training and testing is carried out while following the cross-validation strategy. The prediction of accuracy, AUC, F1 score, precision and Kappa are among the measures utilized in the comparison. The comparison confirms that SVM produces

superior results and is best suited, among the studied methods, for the needed application of gesture recognition.

In the conducted study, while processing the sEMG signals with the proposed hybridization of segmentation, discrete wavelet transform, sub-bands feature extraction, and KNN classifiers the tip gesture had the highest accuracy of 98.5%. The accuracy score for the tip gesture is even higher for the case of SVM classifier and it is 99.4%. The average accuracy score of 91.6% and 97.3% is respectively secured by the KNN and SVM classifiers for the 6-intended hand gestures of a mono-subject.

These results are encouraging and the effectiveness of the developed solution will be evaluated in the future for multiple individuals datasets. The Naive bias and other classifiers such as the Artificial Neural Networks, Decision Trees and Random Forests will also be used for categorization. The deep learning and ensemble learning methods will also be investigated.

6.6 Assignments for Readers

- Describe your thoughts and key findings about the use of sEMG signals in prosthetics.
- Mention the important processes that are involved in the pre-processing and sEMG data collection stages.
- Describe how the performance of post feature extraction and classification stages is affected by the sEMG signal conditioning process.
- Identify your thoughts and key points about the sEMG classification techniques used in this chapter.
- Identify your thoughts and key points on the feature s technique used in this chapter.

References

1. WHO standards for prosthetics and orthotics. <https://www.who.int/publications-detail-redirect/9789241512480> (Accessed Sep. 01, 2022)
2. A. Chaiyaroj, P. Sri-Iesaranusorn, C. Buekban, S. Dumnin, C. Thanawatta-no, D. Surangsrirat, Deep neural network approach for Hand, wrist, grasping and functional movements classification using low-cost sEMG sensors, in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, (2019), pp. 1443–1448. <https://doi.org/10.1109/BIBM47256.2019.8983049>
3. A.U. Alahakone, S.M.N. Senanayake, *Vibrotactile feedback systems: Current trends in rehabilitation, sports and information display* (2009), pp. 1148–1153. <https://doi.org/10.1109/AIM.2009.5229741>
4. W. Park, “The geniuses who invented prosthetic limbs.” <https://www.bbc.com/future/article/20151030-the-geniuses-who-invented-prosthetic-limbs> (accessed Sep. 03, 2022)
5. R. Zhang, X. Zhang, D. He, R. Wang, Y. Guo, sEMG signals characterization and identification of hand movements by machine learning considering sex differences. *Appl. Sci.* **12**(6), 2962 (2022). <https://doi.org/10.3390/app12062962>

6. Z. Yang et al., Dynamic gesture recognition using surface EMG signals based on multi-stream residual network. *Front. Bioeng. Biotechnol.* **9** (2021) Accessed: Sep. 03, 2022. [Online]. Avail-able: <https://www.frontiersin.org/articles/10.3389/fbioe.2021.779353>
7. What Is Quantization? How It Works & Applications. <https://www.mathworks.com/discovery/quantization.html> (Accessed Sep. 03, 2022)
8. S.M. Qaisar, A custom 70-channel mixed signal ASIC for the brain-PET detectors signal read-out and selection. *Biomed. Phys. Eng. Express* **5**(4), 045018 (2019)
9. A.A. Abualsaud, S. Qaisar, S.H. Ba-Abdullah, Z.M. Al-Sheikh, M. Akbar, *Design and Implementation of a 5-Bit Flash ADC for Education* (2016), pp. 1–4
10. S.M. Qaisar, A. Mihoub, M. Krichen, H. Nisar, Multirate processing with selective subbands and machine learning for efficient arrhythmia classification. *Sensors* **21**(4), 1511 (2021)
11. S. Kalmegh, Analysis of WEKA Data Mining Algorithm REPTree. *Simple Cart and RandomTree for Classification of Indian News* **2**(2), 9
12. N. Salankar, S.M. Qaisar, EEG based stress classification by using difference plots of variational modes and machine learning. *J. Ambient Intellig. Humanized Comput.*, 1–14 (2022)
13. S. Mian Qaisar, Baseline wander and power-line interference elimination of ECG signals using efficient signal-piloted filtering. *Healthcare Technol. Letters* **7**(4), 114–118 (2020)
14. S. Mian Qaisar, N. Hammad, R. Khan, A combination of DWT CLAHE and wiener filter for effective scene to text conversion and pronunciation. *J. Electrical Eng. Technol.* **15**(4), 1829–1836 (2020)
15. V. Krishnan, B. Anto, Features of wavelet packet decomposition and discrete wavelet transform for malayalam speech recognition. **1** (2009)
16. Wavelet Decomposition - an overview | ScienceDirect Topics. <https://www.sciencedirect.com/topics/engineering/wavelet-decomposition> (Accessed May 09, 2022)
17. What is Feature Extraction? Feature Extraction in Image Processing. <https://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/> (Accessed Dec. 13, 2021)
18. 1.3.5.11. Measures of Skewness and Kurtosis. <https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm> (Accessed Sep. 03, 2022)
19. What is Machine Learning?, Jul. 02, 2021. <https://www.ibm.com/sa-en/cloud/learn/machine-learning> (Accessed Sep. 03, 2022)
20. H.D. Wehle, Machine Learning, Deep Learning, and AI: What's the Difference?. (Jul. 2017)
21. What is Deep Learning?. <https://www.ibm.com/cloud/learn/deep-learning> (accessed Sep. 03, 2022)
22. S.M. Qaisar, A. López, D. Dallet, F.J. Ferrero, *sEMG Signal based Hand Gesture Recognition by using Selective Subbands Coefficients and Machine Learning* (2022), pp. 1–6
23. H. Fatayerji, R. Al Talib, A. Alqurashi, S.M. Qaisar, *sEMG Signal Features Extraction and Machine Learning Based Gesture Recognition for Prosthesis Hand* (2022), pp. 166–171
24. J. Brownlee, How to calculate precision, recall, and F-measure for Im-balanced classification. *Machine Learning Mastery*, Jan. **02** (2020) <https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/> (Accessed Sep. 04, 2022)
25. F-Score. DeepAI. (2019). <https://deeplai.org/machine-learning-glossary-and-terms/f-score> (Accessed Sep. 04, 2022)
26. Kappa Statistics - an overview | ScienceDirect Topics. <https://www.sciencedirect.com/topics/medicine-and-dentistry/kappa-statistics> (Accessed Sep. 04, 2022)
27. S. Mian Qaisar, F. Alsharif, Signal piloted processing of the smart meter data for effective appliances recognition. *J. Electrical Eng. Technol.* **15**(5), 2279–2285 (2020)
28. S.M. Qaisar, S.I. Khan, D. Dallet, R. Tadeusiewicz, P. Pławiak, Signal-piloted processing metaheuristic optimization and wavelet decomposition based elucidation of arrhythmia for mobile healthcare. *Biocybernet. Biomed. Eng.* **42**(2), 681–694 (2022)
29. S.M. Qaisar, Efficient mobile systems based on adaptive rate signal processing. *Comput. Elect. Eng.* **79**, 106462 (2019)