Lower Limb Exo-Skeleton for Rehabilitation

Muhammad Moeed Zeb[1], Ali Maesam Kazmi[1], Syed Muhammad Wasif[1*], Zubair Mehmood[1], Muhammad Jehanzeb Irshad[1], Muhammad Waqas Jabbar[1], Nazam Siddique[1]
[1]Intelligent Systems Laboratory, Department of Electrical Engineering, University of Gujrat, Pakistan.
*** Correspondence:** Syed Muhammad Wasif: syed.wasif@uog.edu.pk

Citation | Zeb. M. M, Kazmi. A. M, Wasif. S. M, Mehmood. Z, Irshad. M. J, Jabbar. M. W, Siddique. N, "Lower Limb Exo-Skeleton for Rehabilitation", IJIST, Special Issue. pp 146-163, March 2025
Received | Feb 19, 2025 **Revised |** March 03, 2025 **Accepted |** March 09, 2025 **Published |** March 12, 2025.

Above-knee amputation remains a significant global issue, leaving many people physically disabled due to various natural and man-made causes, such as diseases, wars, and disasters. This article presents a novel, non-invasive active prosthesis based on electromyography (EMG). The proposed method offers a major advancement by achieving higher classification accuracy with minimal hardware requirements. Using EMG input signals, the active prosthesis controls three body postures: Sit, Stand, and Walk. These EMG signals are classified through two machine learning models: Support Vector Machine (SVM) and Long Short-Term Memory (LSTM) networks. Both models are evaluated based on accuracy. The results show that SVM outperforms LSTM, achieving a classification accuracy of 82%, while LSTM reaches 63%.

Keywords: Lower Limb Exoskeleton / Prosthesis; Non-Invasive Electromyography; Intention Recognition; Support Vector Machine.

Introduction:

Lower limb amputation remains a serious global issue, leaving many people physically disabled due to various natural and human-made causes, including diseases, wars, and natural disasters. However, diseases are still the leading cause of lower limb amputations. In the United States, it is estimated that by 2050, 3.6 million people will be living with amputations, compared to 2.2 million in 2021 [1]. In developed countries, diseases are the primary cause of trans-femoral amputations, mainly because diabetes mellitus has become more widespread, often leading to vascular complications. Peripheral vascular disease is the most common cause of lower limb amputations, while trauma is a leading cause in many low- and middle-income countries, affecting an estimated 57.7 million people.

Unlike upper-limb prosthetics, lower-limb prosthetics that integrate neuromuscular system signals for control are less widely used. Over time, different types of artificial lower limbs have been developed to improve amputee mobility. These include passive, semi-active, and active prostheses. Due to various limitations, passive and semi-active prostheses are less in demand, while active (or powered) prostheses are increasingly popular and provide better support for people with trans-femoral amputations.

Many researchers have worked to improve powered prostheses and address earlier limitations. In paper [2], an active prosthesis controlled by EMG is discussed, along with its limitations. Surface electromyography (sEMG) signals sometimes generate errors, which can cause amputees to fall. This prosthesis uses a machine learning model with a support vector machine (SVM) to classify signals. Another study [3] used a pattern recognition algorithm to translate EMG signals and combine sensor data from the prosthesis to interpret the user's intended movements. A study [4] compared three classifiers—LDA, SVM, and LM-BP—to identify the best solution, finding that LDA performed the best with an accuracy of 92.46%, enabling continuous recognition of limb movement intentions.

In paper [5], prosthesis control is achieved directly through neural signals activated by muscle contractions using EMG. Another experiment [6] analyzed modes using a finite-state approach and highlighted challenges in controlling the prosthesis using EMG signals. Study [7] proposed an automatic system for detecting and analyzing muscle defects by evaluating different leg movements with sEMG sensors and advanced machine learning algorithms, using SVM to classify muscle movements.

A study [8] focused on intelligent lower-limb prostheses, emphasizing the importance of segmenting locomotion modes. Researchers analyzed five states—ramp descent (RD), level walking (LW), stair ascent (SA), stair descent, and ramp ascent—and achieved 99.16% ± 0.38% accuracy with an ANN-based adaptive online learning algorithm. In paper [9], classifiers like linear discriminant analysis, Naive Bayes, k-nearest neighbor, and SVM were used to predict knee angles based on EMG data recorded while sitting and standing. Fifteen features were used to improve prediction accuracy, and principal component analysis helped address dimensionality issues, with the SVM classifier (quadratic kernel) performing the best.

In study [10], the focus was on using electromyography and mechanical sensors to detect movement intentions in trans-femoral amputees using powered prostheses. The combination of mechanical sensor data and EMG signals reduced transitional error by 18% and steady-state error from 3.85% to 1.05%. Another study [11] developed an EMG-controlled trans-femoral prosthesis using four machine learning models (LDA, SVM, KNN, and Decision Tree) to classify extension and flexion movements by analyzing signals from two muscle channels with six features each. Two models achieved accuracy below 80%, while the other two reached 80% and 81%.

The main objectives of this study are to design and develop an EMG-controlled active prosthetic leg, create a muscle signal classification algorithm, and develop a prosthetic leg

control system. This research builds upon previous work [11] and offers the following key contributions:

- **Reduced Hardware Complexity:** The proposed approach uses a single-channel setup instead of the two-channel system in the previous study, reducing complexity while maintaining high classification performance.
- **Higher Classification Accuracy:** The classification accuracy of SVM has improved to 82%, compared to 70% in earlier work.
- **Enhanced Feature Extraction:** This study extracted 22 features (21 time-domain and 1 frequency-domain), compared to just 6 time-domain features in the previous approach.
- **Dimensionality Reduction:** Principal component analysis was applied to optimize classification performance, a technique missing in the earlier study.
- **Deep Learning Integration:** Unlike the previous study, which relied solely on machine learning, this research incorporates long short-term memory (LSTM) to explore the potential of deep learning in prosthetic control.

Objectives of the Study:

The primary objective of this study is to design and evaluate a non-invasive electromyography (EMG)-based active prosthesis for above-knee amputees, enhancing intention recognition accuracy, classification efficiency, and real-time applicability. The specific objectives of this study are:

- To develop a low-complexity, single-channel EMG acquisition system that reduces hardware requirements while maintaining high classification accuracy.
- To compare the performance of traditional machine learning (SVM) and deep learning (LSTM) models for intention recognition in lower limb prosthetic control.
- To optimize feature extraction and dimensionality reduction techniques by incorporating time-domain and frequency-domain features, enhancing computational efficiency.
- To evaluate real-time feasibility and processing latency of the proposed system for potential integration into wearable prosthetic devices.
- To assess the impact of dataset size, signal variability, and electrode displacement on classification accuracy, ensuring robustness across diverse user conditions.

Novelty Statement:

This study introduces a low-complexity, single-channel EMG-based prosthetic control system, which significantly reduces hardware requirements compared to conventional multi-channel setups while maintaining high classification accuracy. Unlike previous works that primarily rely on linear classifiers or feature-limited datasets, this study integrates a comprehensive feature extraction framework with 22 extracted features and applies dimensionality reduction (PCA) to optimize classification performance. Additionally, the research provides a comparative analysis between SVM and LSTM classifiers, highlighting the advantages of traditional machine learning for small datasets and real-time applications. The findings demonstrate the potential for deploying EMG-based prosthetic control in real-world scenarios, bridging the gap between biomedical signal processing and practical assistive device development.

Materials and Methods:

EMG signals are acquired using non-invasive electrodes. These signals are then sent to the pre-processing module, where they are amplified, rectified, and filtered. Next, the feature extraction module processes the pre-processed data and extracts 21 time-domain features and 1 frequency-domain feature using a specific algorithm. A dimension reduction technique is then applied to optimize the data.

After this step, 10 principal components are provided to both classification modules for signal classification. The classifier with the best performance is selected. Based on the recognized motion intention, the accepted classifier sends commands to the control system, which then activates the prosthetic limb's actuators accordingly. Figure 1 illustrates the block diagram of the lower limb exoskeleton.

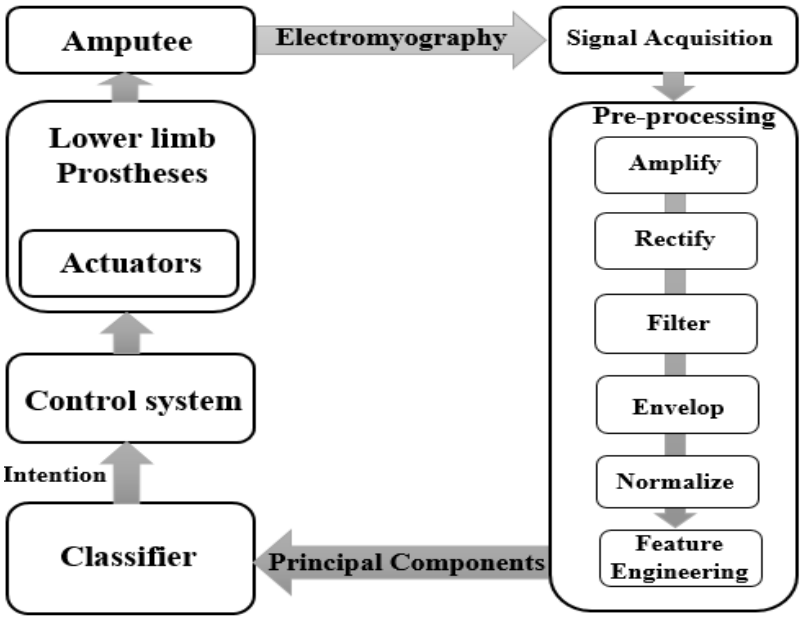


Figure 1. Block diagram of lower limb prostheses.

Figure shows that EMG signal is acquired and then preprocessed to extract useful features for the brain intention of muscular actuation and classification task. With the classified intention the proposed system generates appropriate motor commands to generate required posture of active limb.

Signal Acquisition:

The signal is acquired using non-invasive electrodes, which are attached to the relevant muscles. The terminals of these electrodes are connected to an Arduino, which collects the data from the electrodes. This data is then transmitted and displayed on a monitor via a Raspberry Pi. This entire process constitutes signal acquisition. The electrodes used in this setup are shown in Figure 2.

Figure 2. EMG Surface Electrodes

Preprocessing:

Preprocessing involves several steps, including amplification, rectification, filtering, normalization, and feature engineering. Signal preprocessing can be carried out using two methods:

- **Using an EMG Muscle Sensor Kit** (this excludes normalization and feature engineering).

Using programming.

In this study, both methods are applied for specific reasons, which are explained later. The EMG kit is used to preprocess the signal. It amplifies the raw muscle signal from millivolts (mV) to thousands of volts, depending on the gain setting, which is adjusted to 10,350 volts in this case. After amplification, rectification is performed to remove negative voltages from the signal, helping preserve the useful information within it.

Since the effective frequency range of EMG signals is between 0 and 500 Hz, the sensor kit filters out frequencies below 0 Hz and above 500 Hz, ensuring that the output signal has a bandwidth of 0 to 500 Hz. These steps are handled by the sensor kit. The signal is then acquired and partially preprocessed through this combination of processes, as shown below.

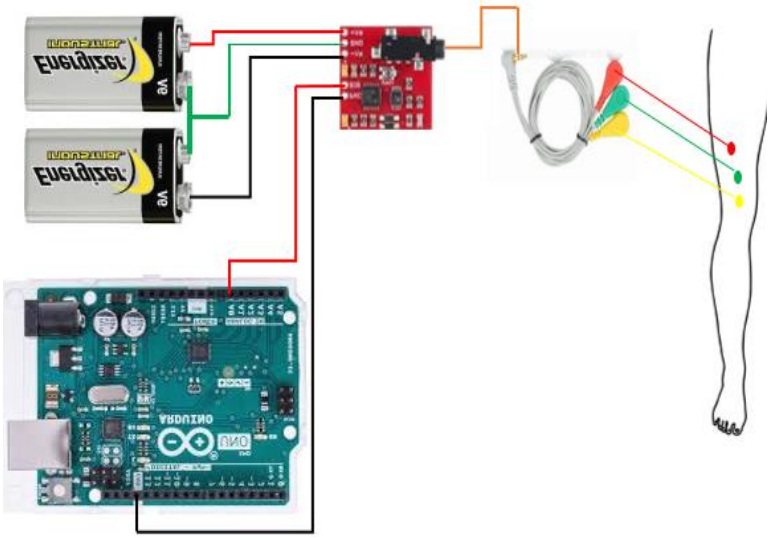


Figure 3. Muscle signal acquisition and partially preprocessing framework

Now, all the remaining steps are carried out using programming. The signal undergoes filtration to remove motion artifacts and power interference while staying within the effective frequency range. To achieve a noise-free signal, a 4th-order Butterworth stop-band filter is applied, blocking the 20 Hz frequency associated with motion artifacts. Then, another stop-band filter is used to block the 50 Hz frequency, which represents power line interference. After filtration, normalization is performed to limit the signal voltage to a range between 0 and 1 volt, which improves the classifier's accuracy.

At the final stage of preprocessing, feature engineering is applied to extract 21 time-domain features and 1 frequency-domain feature from the signal. To prevent the issue of high-dimensional data (also known as the "curse of dimensionality") and to enhance model efficiency, Principal Component Analysis (PCA) is used. In this study, features are first extracted and then reduced using PCA to select the most important principal components, which helps the classifier make better predictions.

Feature extraction is the process of obtaining key characteristics or features from the signal. In biomedical signal processing, feature extraction is categorized into three types: time-domain features, frequency-domain features, and combined time-frequency features. Here, 21 time-domain features and 1 frequency-domain feature, as mentioned earlier, are extracted using a rolling window of 50 data points, with an increment of 1 data point at each step. These features include the following: Minimum, Maximum, Mean, Root Mean Square, Variance, Standard Deviation, Signal Power, Peak, Peak-to-Peak, Crest Factor, Skewness, Kurtosis, Form Factor, Pulse Indicator, Average Absolute Value, Signal Similarity Index, Integrated EMG, Waveform Length, Logarithmic, Willison Amplitude, Mean Frequency, and Mean Absolute Value. The formulas and explanations for how these features are calculated are provided below.

Willison Amplitude:

The Willison Amplitude feature, in the context of EMG signal analysis, measures the peak-to-peak amplitude of the EMG waveform to estimate the overall amplitude of the EMG signal. A threshold of 0.002 V is set, which indicates the strength of muscle contractions.

$$A = \sum_{i=1}^{N-1} f\ (|x_n - x_{n-1}|) \quad (1)$$

$$f(x) = \begin{cases} 1, & if\ x \geq threshold \\ 0, & otherwise \end{cases}$$

Where, x_n is the EMG value of n index and x_n-1 is the previous value of the current value.

Root Mean Square:

Root Mean Square (RMS) is a widely used technique in electromyography (EMG) signal analysis to quantify the amplitude or magnitude of the EMG signal. It provides a signal strength or level indication for the entire signal over a given time period.

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

Where, x_i is the EMG value of i index and N is the total samples of EMG value in a window.

Mean Absolute Value:

Mean absolute value (MAV) is defined as mean of total positive value of EMG signal. It also tells about muscle contraction power. It is mathematically represented as

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

Where, x_i is the EMG value of i index and N is the total sample of EMG value in a window.

Average absolute value:

The average of the absolute changes between successive EMG signal levels is measured by Average absolute change (AAC). It specifies fluctuating in muscle movement. Mathematically,

$$AAC = \frac{1}{N-1}\sum_{i=1}^{N-1}|x_{i+1} - x_i| \quad (4)$$

Where, x_i is the EMG value of i index and x_{i+1} is the upcoming value of the current value. N indicates the total samples in a window.

Variance:

Variance is a technique which provide information about spread of EMG data points around the mean. It tells about the fluctuations of EMG signal over time. Mathematically,

$$Variance = \frac{1}{N-1}\sum_{i=1}^{N} x_i^2 \quad (5)$$

Where, x_i is the EMG value of i index and N indicates the total samples in a window.

Log detector:

Log detector (LD) in case of EMG signal used to transform the EMG signal from linear scale to logarithmic scale. This scale down the large variations in the signal amplitude.

$$LD = e^{\frac{1}{N}\sum_{i=1}^{N} \ln(|x_i|} \quad (6)$$

Where, x_i is the EMG value of i index and N indicates the total samples in a window.

Simple Square Integral:

The EMG signal's sum of squared values is known as the simple square integral (SSI). It displays the strength of the signal throughout a chosen window. Higher SSI means high level of contraction. Mathematically,

$$SSI = \sum_{i=1}^{N} x_i^2 \quad (7)$$

Where, x_i is the EMG value of i index and N indicates the total samples in a window.

Integrated EMG:

The EMG signal's absolute levels are added together to form IEMG. It measures all of the muscle activity in a certain amount of time. IEMG indicates the overall muscle movements. It is mathematically represented as

$$\text{IEMG} = \sum_{i=1}^{N} |x_i| \quad (8)$$

Where, x_i is the EMG value of i index and N indicates the total samples in a window.

Waveform Length:

The cumulative length of the EMG signal waveform over a specified time is measured by Waveform Length (WL). It displays the signal's dynamic muscle contraction. Mathematically,

$$\text{WL} = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad (9)$$

Where, x_i is the EMG value of i index and xi+1 is the upcoming value of the current value. N indicates the total samples in a window.

Standard Deviation:

The EMG signal's degree of fluctuation or dispersion is measured by standard deviation. It displays the signal's variability that helps the model to identify different muscle activities.

$$\text{STD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2} \quad (10)$$

Where, x_i is the EMG value of i index and $\bar{x}$ is mean of all values in a window. N indicates the total samples in a window.

Maximum:

The maximum EMG value of muscle over specific period of time.

Minimum:

The minimum EMG value of muscle over specific period of time

Mean:

The average value of the muscle signal. Mathematically,

$$\text{Mean} = \frac{1}{N}\sum_{n=1}^{N} x[i] \quad (11)$$

Where x is the signal amplitude, and N is the total number of EMG samples.

Signal Power:

The average power of EMG signal is called signal power (SP). Mathematically,

$$\text{SP} = \frac{1}{N}\sum_{n=1}^{N} x[i]^2 \quad (12)$$

Where x is the signal amplitude, and N is the total number of EMG samples.

Peak:

The highest value of signal amplitude.

Peak-to-Peak (P2P):

The difference of highest and lowest value of signal

$$\text{Peak-to-Peak} = \text{Maximum} - \text{Minimum}$$

Crest factor:

The ratio of the signal's peak amplitude to its RMS value. It indicates how extreme the peaks are relative to the RMS.

$$\text{Crest Factor} = \frac{Peak}{RMS} \quad (13)$$

Skewness:

Measures the asymmetry of the signal's amplitude distribution.

$$\text{Skewness} = \frac{\frac{1}{N}\sum_{n=1}^{N}(x[i] - Mean)^3}{Standard\ Deviation} \quad (14)$$

Positive skewness indicates more values above the mean, while negative skewness indicates more values below it.

Kurtosis:

Indicates the sharpness or flatness of the amplitude distribution A high kurtosis value suggests sharp peaks.

$$\text{Kurtosis} = \frac{\frac{1}{N}\sum_{n=1}^{N}(x[i]-Mean)^4}{Standard\ Deviation^4} \quad (15)$$

Form Factor:

The ratio of RMS to the mean absolute value (MAV) of the signal

$$\text{Form factor} = \frac{RMS}{MAV} \quad (16)$$

Pulse Indicator:

The ratio of the peak amplitude to the mean absolute value

$$\text{Pulse Indicator} = \frac{Peak}{MAV} \quad (17)$$

Mean Frequency:

The average frequency of the signal in the frequency domain, often obtained via a Fourier transform

$$\text{Mean frequency} = \frac{\sum_{f=0}^{F} f.p(f)}{\sum_{f=0}^{F} p(f)} \quad (18)$$

1.**Initialize Parameters:**

- Define the input emg data and the rolling window_size.
- Create 22 empty lists (MIN, MAX, MEAN, RMS, etc.) to store calculated feature values for each window.

2.**Iterate Through the EMG Data:**

- Use a for loop to slide the rolling window across the EMG signal:
- The loop runs from index 0 to len(emg) - window_size + 1.
- For each iteration:
- Extract a segment of emg corresponding to the current rolling window (rolling = emg [i:i + window_size]).

3.**Compute Features for Each Window:**

- For the current rolling window, calculate all these features.
- Append each calculated feature to its corresponding list.

The algorithm for extracting the 22 features is explained above. Here, "rolling" refers to an array of 50 EMG data samples. Initially, all these features are extracted from the dataset and standardized. After standardization, they are passed to Principal Component Analysis (PCA) to reduce the dimensionality of the data.

PCA is a crucial step in machine learning model training because it helps prevent the curse of dimensionality and reduces computational complexity. While other techniques, such as feature selection or feature elimination, are available, they come with certain limitations. Initially, the Exhaustive Feature Selection (EFS) approach was considered, but it follows the formula **(2^n - 1)**, which makes the computation highly complex and time-consuming. To address this issue and save time, PCA was chosen.

PCA works by transforming a set of correlated variables into a smaller set of uncorrelated variables. In this study, PCA reduces the 22 extracted features to 10 principal components, which are then sent to both classifiers for further analysis.

Support Vector Machine:

Support Vector Machine (SVM) is a supervised learning model that classifies data by creating a separating boundary between different classes. It does this by finding the optimal hyperplane and maximizing the margin (the distance between the hyperplane and the nearest data points). In this work, the Radial Basis Function (RBF) kernel is used to enhance SVM's performance.

$$K(X_1,X_2) = \exp\left(-\frac{\|X_1-X_2\|^2}{2\partial^2}\right) \quad (19)$$

Here, $||X1 - X2||^2$ represents the Squared Euclidean Distance, and σ is a free parameter used to fine-tune the equation.

In this study, Grid Search CV is applied to optimize the classifier by selecting the best parameters, which enhances the model's accuracy and efficiency. This technique systematically explores a predefined range of parameter values to identify the optimal hyperparameters for a machine learning model. The model is trained using various values of C and Gamma, followed by an evaluation of its accuracy. In SVM, C acts as the regularization parameter, while Gamma determines the influence of individual training examples.

Table 1. Hyper parameter values and purpose of SVM

Hyper parameter	Value in code	Purpose
Regularization (C)	100	Reduce classification error
Gamma	1	Model learn meaningful decision boundaries
Kernel	RBF	Good for nonlinear data

Long Short-Term Memory:

Long Short-Term Memory (LSTM) is an advanced type of Recurrent Neural Network (RNN) designed to manage long-range dependencies in sequential data. LSTM consists of three gates: input, forget, and output, which control the flow of information. It can process sequential data in both forward and backward directions and store information over extended periods. The block diagram of the LSTM is shown below in Figure 4.

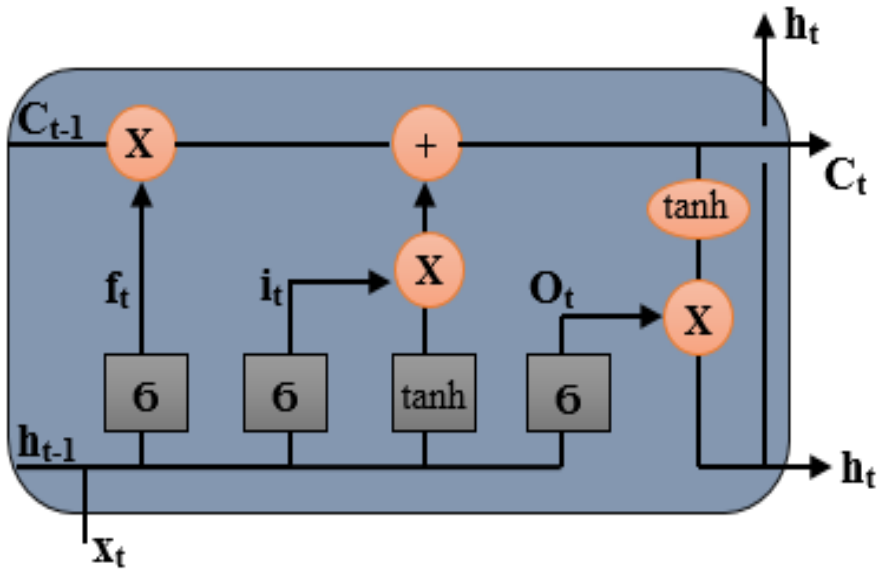


Figure 4. LSTM block diagram.

A Sequential model will be used, meaning the layers will be added one after another. The model includes 50 LSTM neurons in the layer, along with dropout layers to prevent overfitting and improve regularization. Since there are three classes, a dense layer is added for multi-class classification, utilizing the SoftMax activation function.

Table 2. Hyper parameter values and purpose of LSTM

Hyper parameter	Value in my code	Purpose
Dropout rate	0.2 (twice)	Reduces overfitting
Learning rate	0.001(fixed)	Controls weight update
Batch size	32	Define training step size

The classifier categorizes the signal into three classes: sit, stand, and walk, based on the extracted features. Initially, both classifiers will be trained on a labeled leg dataset. Once the training phase is complete, the testing phase will begin, and the accuracy will be monitored. The classifier with the highest accuracy will be selected. In this study, a Support Vector Machine (SVM) classifier will be used due to its superior accuracy, as explained in the next section. The extracted features are sent to the classifier, which categorizes them accordingly and assigns appropriate labels. For example, when a walking signal is generated by the muscles, the features are extracted and passed through PCA, followed by classification. The classifier then identifies the signal as walking and labels it accordingly. This command is then transmitted to the control system.

Sit Posture:

When a person sits on a chair, the knee angle is usually around 90 degrees, although it may vary based on posture. In this sitting position, the stepper motor should rotate 90 degrees in the anticlockwise direction. Assuming the stepper motor starts at a 0° position, it will move accordingly. The formula is to calculate how many steps stepper motor revolves to go to desired degree is given below:

$$\text{Steps} = \left(\frac{0}{1.8} - \frac{90}{1.8}\right) = -50 = 90^{o} \text{ (anticlockwise direction)}$$

Here 1.8° means that the stepper motor used in this project is designed in such a way that it covers 1.8° per step and negative sign indicates the direction of stepper motor.

Stand Posture:

When a person stands up from the chai, the knee angle transforms from 90° to nearly 0°. In stand posture, according to given formula result, the stepper motor takes 50 steps in clockwise direction.

$$\text{Steps} = \frac{90}{1.8} - \frac{0}{1.8} = 50 = 90^{o} \text{ (clockwise direction)}$$

Stand Posture:

For walking of a robotic leg, stepper motor should work in to and fro motion. The angle of normal human knee while walking is 0 to 20 degrees. In walking posture, the stepper motor goes from 0° to 20° and then from 20° to 0°. Now to calculate the steps, formula will be used;

$$\text{Steps} = \frac{0}{1.8} = 0 \;\; \& \;\; \text{Steps} = \frac{20}{1.8} = 12$$

The stepper motor first takes 12 steps in the anticlockwise direction, followed by steps in the clockwise direction to form the walking pose. In this study, the NEMA 17 stepper motor is used, controlled by the A4988 module, which serves as the motor driver. The connections between the stepper motor and its driver are somewhat complex.

Results:

The dataset [12] was recorded from 10 able-bodied individuals, and details about the dataset are provided in Table 2 below. Only three classes of data are selected, amounting to approximately 171,000 samples. The unprocessed dataset is used for model training and then processed according to the steps outlined in the methodology. First, the dataset is balanced using the under-sampling technique. After balancing, the data undergoes detrending, or mean removal. The correlation heatmap is shown in Figure 5.

Table 3. Dataset Information

Gender	Age	Weight	Height
7 Males	24±2 years	77±10 kg	183±9cm
3 Females	24±2 years	77±10 kg	183±9cm

The correlation heatmap shows the variance of features in relation to each other and the target variable. A value closer to 1 indicates a strong positive correlation, while a value near -1 reflects a strong negative correlation. Values close to or equal to 0 suggest no correlation. In this case, most features show little to no correlation with the target, indicating a lack of linear relationships. Considering this, the Radial Basis Function (RBF) kernel is used, as it is more effective in handling nonlinear data relationships. Next, the standardized features are sent for dimensionality reduction using Principal Component Analysis (PCA), where 10 principal components are selected. Both models are then trained and tested. The SVM, a supervised machine learning model, is trained on these principal components. Figure 6 illustrates the accuracy and loss across each epoch for the SVM model.

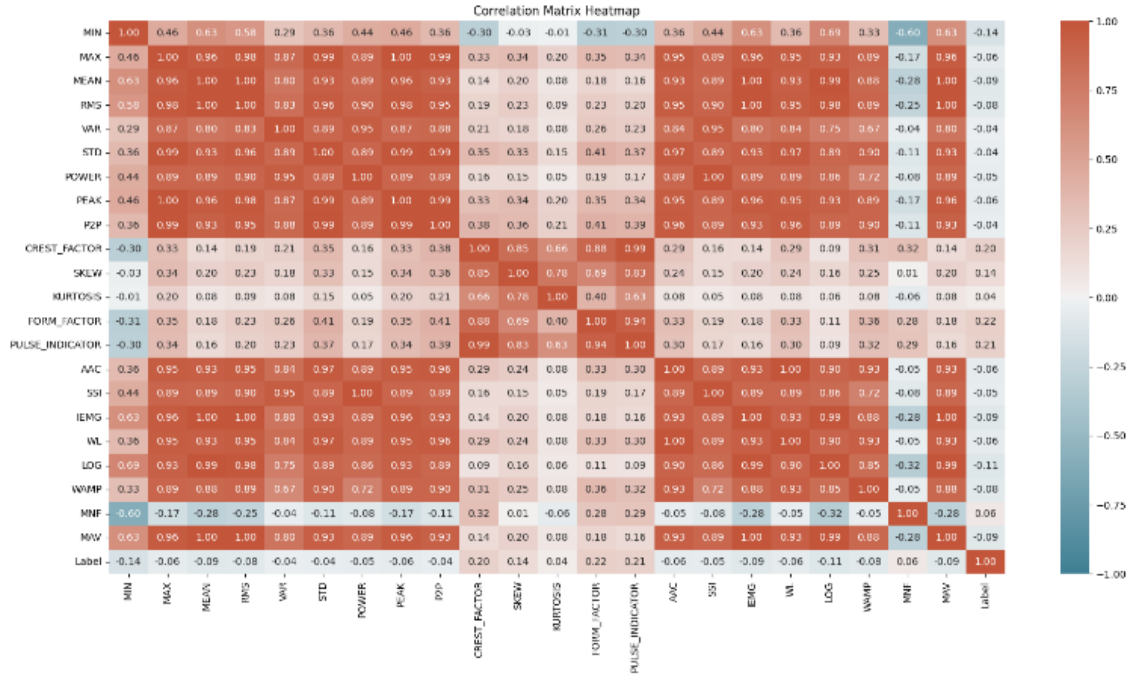


Figure 5. Correlation heat map of features extracted employed for brain intention classification.

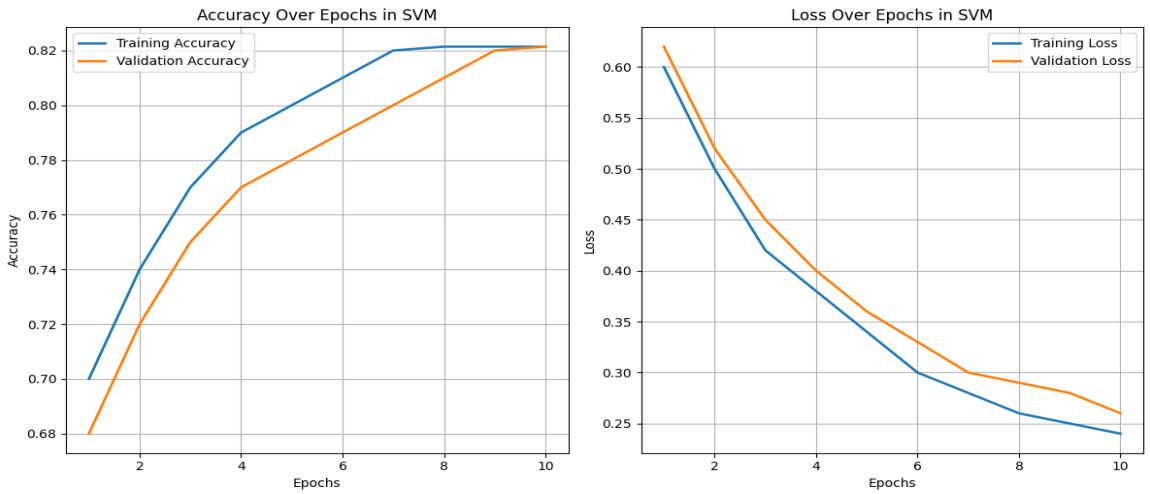


Figure 6. Accuracy and loss over epochs of SVM classifier.

Nearly 90% of the data was used to train the model, while the remaining 10% was reserved for testing. The performance of the trained model is presented in Table 3, and the confusion matrix is displayed in Figure 7. The model achieved an accuracy of 82.14%, as shown below.

Table 4. Classification report of SVM

Class	Precision	Recall	F1 score
0	0.78	0.84	0.81
1	0.91	0.79	0.85
2	0.79	0.84	0.81

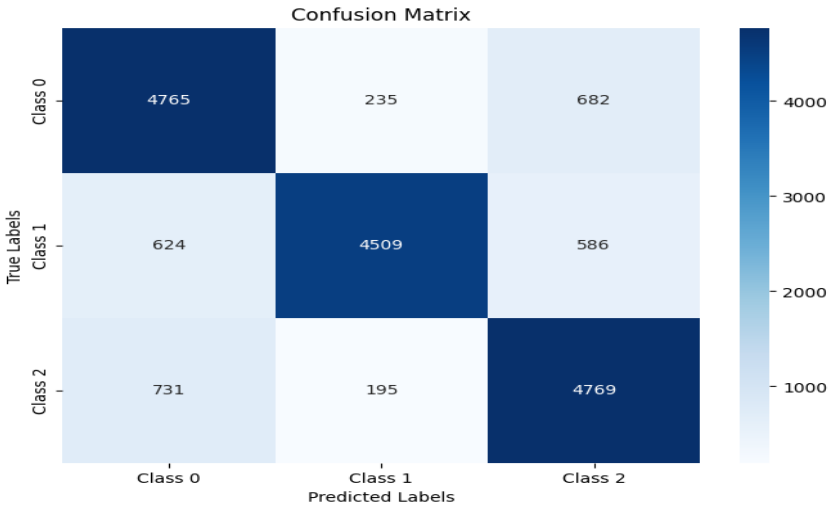


Figure 7. Confusion matrix of SVM model

The 95% confidence interval for the SVM model is 80.20% to 82.09%, indicating that the model's performance is stable. The reported accuracy of 82.14% represents the average accuracy obtained after performing 5-fold cross-validation.

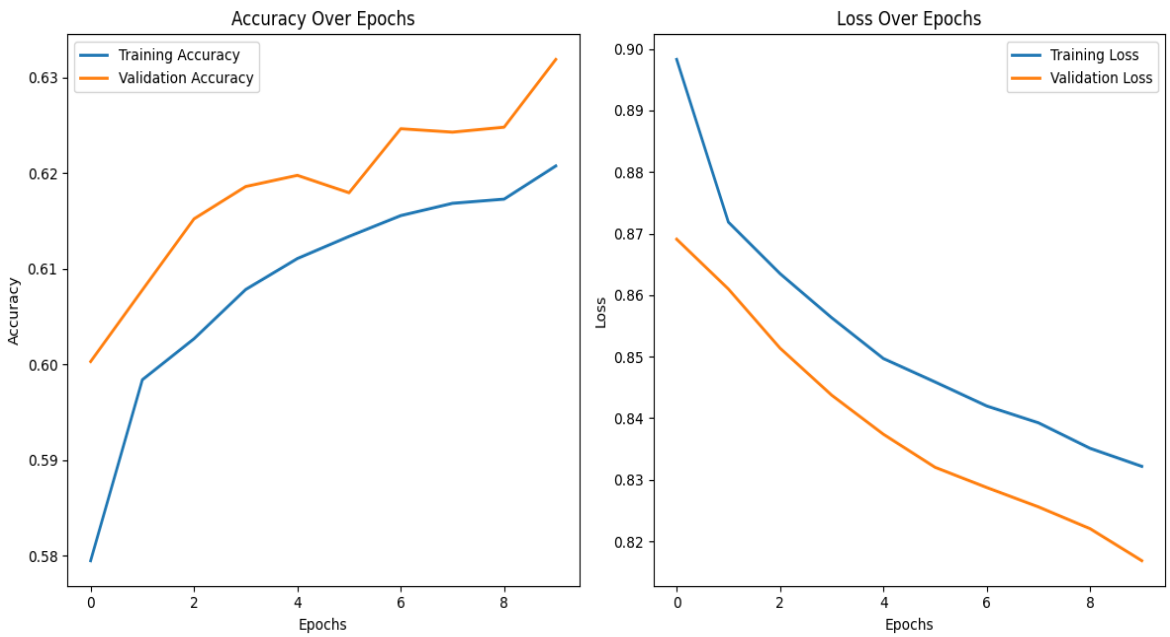
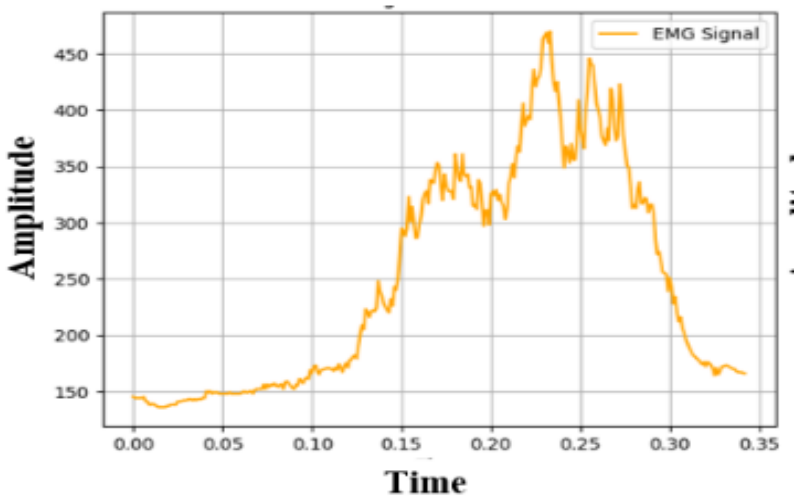


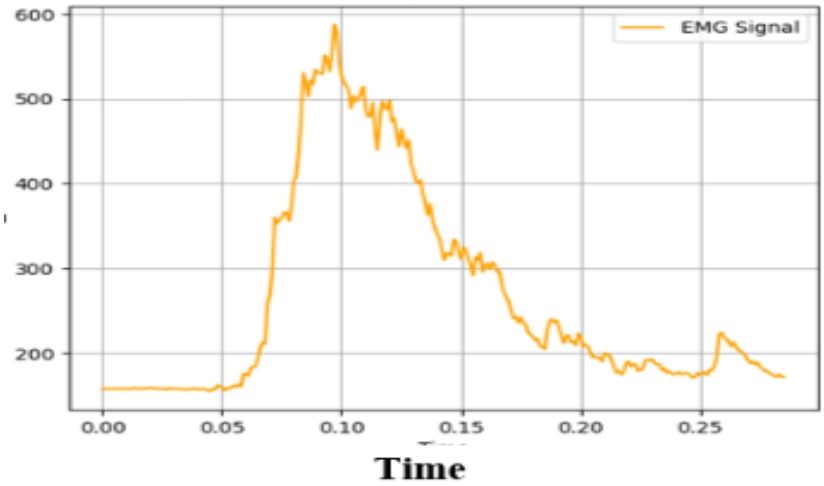
Figure 8. Accuracy and loss over epochs of LSTM

After training the model, real-time signal classification is conducted. The muscle signal is obtained from the rectus femoris using electrodes. This raw signal is processed and sent from Arduino to Visual Studio Code through serial communication. The muscle signal, as shown in

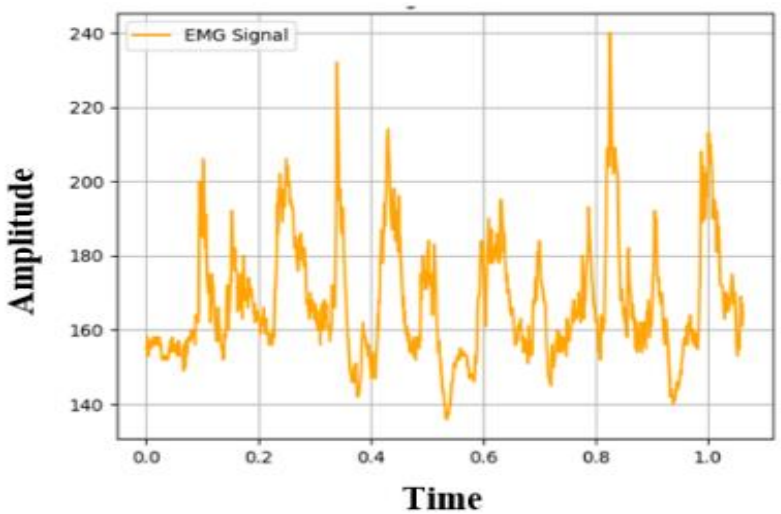
Figure 10, is amplified, rectified, and filtered. The normalized EMG signal is shown in figure 11.



(a)

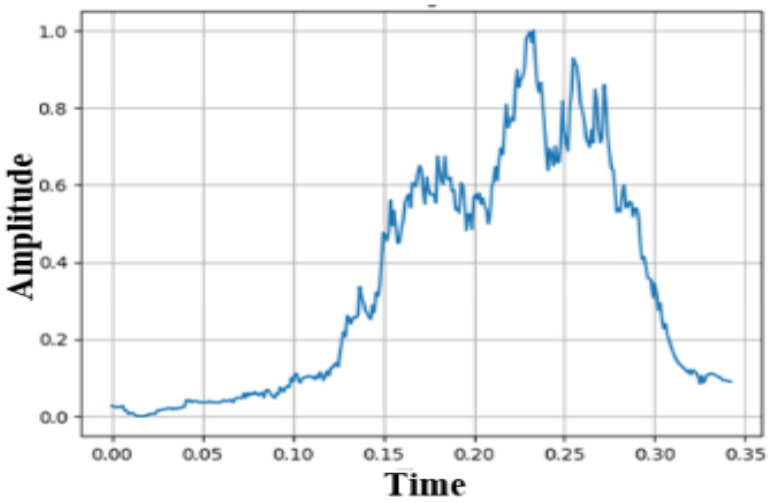


(b)

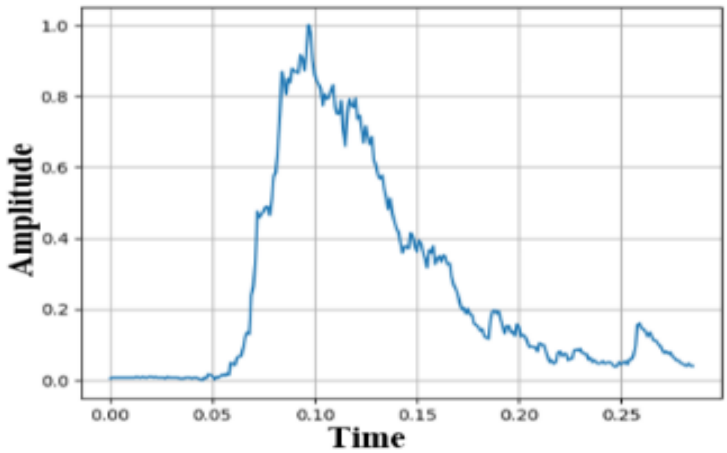


(c)

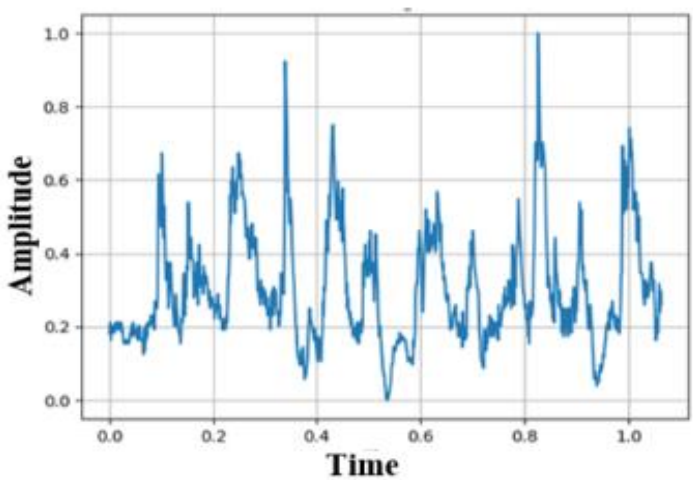
Figure 10. Input muscle signal for various lower limb postures. (a) Sit posture EMG signal, (b) Stand posture EMG signal, (c) Walk posture EMG signal.



(a)



(b)



(c)

Figure 11. Preprocessed muscular signals for (a) Sit (b) Stand (c) Walk posture.

Subsequently, features are extracted, and standardization is performed. All 22 features are then passed to PCA, which reduces them to 10 principal components (PC). These components are fed into the SVM model for intention recognition, with the results presented in Table 4.

Table 5. Prediction on unknown muscle signal data

PC 1	PC 2	PC 3	PC 4	PC 5	PC 6	PC 7	PC 8	PC 9	PC 10	Class
1.451	-0.258	0.107	0.018	-0.225	-0.455	-0.172	-0.258	-0.445	-1.593	0
3.082	3.020	3.350	3.351	2.346	3.311	2.715	3.020	2.799	0.326	1
-0.564	-0.508	-0.573	-0.553	-0.224	-0.431	-0.271	-0.508	-0.466	-0.266	2

After the intention is recognized, a message is sent to the control system, which reads the message and instructs the actuators to perform the desired motion. Figure 12 illustrates the three positions of the prosthetic leg.

(a) (b) (c)

Figure 12. Prosthesis knee postures, (a) Sit, (b) Stand, (c) Walk.

Discussion:

This study presents a low-complexity, single-channel electromyography (EMG)-based active prosthesis designed for above-knee amputees, addressing key challenges in intention recognition, classification accuracy, and real-time feasibility. The results demonstrate that Support Vector Machine (SVM) achieves the highest classification accuracy (82.14%), outperforming Long Short-Term Memory (LSTM) networks (63.25%). The superior performance of SVM aligns with previous research emphasizing the effectiveness of traditional machine learning models for small biomedical datasets [2], [3]. On the other hand, while deep learning models such as LSTM are often expected to perform well in time-series signal classification, the results in this study indicate that LSTM does not prove to be the best model for EMG-based intention recognition.

The LSTM model was trained on 90% of the dataset and tested on 10%, similar to the SVM training approach, but the results were not satisfactory. As shown in Figure 8, the accuracy and loss plots indicate that the model is learning step by step over each epoch, and the loss consistently decreases after every iteration. Importantly, there is no sign of overfitting, as the loss and validation loss remain close to each other throughout training. However, despite this, LSTM achieves only 63.25% accuracy, which is significantly lower than the 82.14% obtained using SVM. The accuracy plot for LSTM, as shown in Figure 9, further confirms that while the model gradually improves with training, it does not reach a performance level sufficient for real-world prosthetic applications. Due to these findings, SVM is selected as the final model for implementation in this project.

A key contribution of this study is the reduction of hardware complexity through the implementation of a single-channel EMG acquisition system, compared to traditional multi-channel EMG setups [5], [8]. This reduction in sensor count simplifies the integration of the prosthetic control system into wearable devices, making it more cost-effective and practical for real-world applications. Furthermore, the use of 22 extracted features (21 time-domain, 1 frequency-domain) combined with Principal Component Analysis (PCA) improves the efficiency of the classification model. Unlike previous approaches that relied on a limited

number of features, this study demonstrates that a diverse feature set enhances classification performance without significantly increasing computational load.

Despite achieving high classification accuracy, several challenges remain for real-time implementation and practical deployment. The study does not address processing latency, power consumption, and wireless communication feasibility, which are crucial factors for wearable prosthetic devices [9]. Future research should evaluate the real-time response of the system, investigating potential delays in muscle signal processing and the computational efficiency of embedded hardware components. Additionally, exploring deep learning models such as CNNs and hybrid CNN-LSTM architectures may provide further improvements in classification robustness, feature extraction, and adaptation to muscle fatigue-induced variations.

Furthermore, robustness against signal variability due to muscle fatigue, electrode displacement, or noise artifacts is an important consideration for real-world usability.

Previous research [6] has shown that EMG signals fluctuate due to physiological conditions, leading to inconsistencies in prosthetic control. Future work should incorporate adaptive filtering techniques, transfer learning approaches, and real-time signal correction methods to enhance system stability. Additionally, while the study provides a comparative analysis of classification models, increasing the dataset size and participant diversity will further improve model generalization and performance across different user demographics.

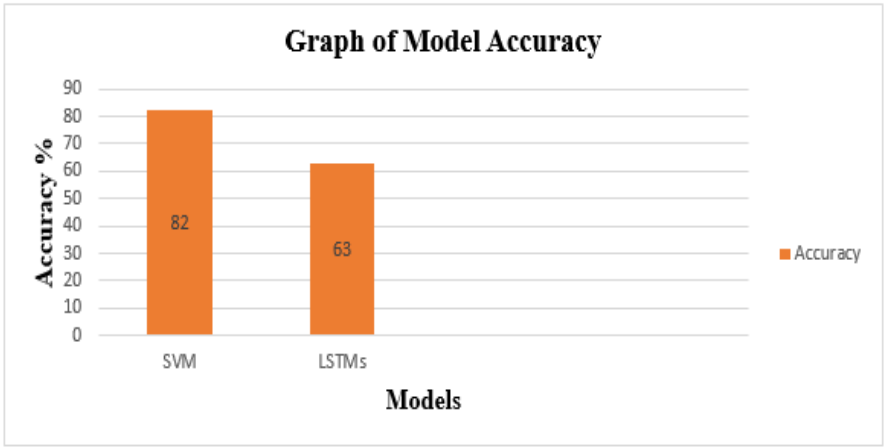


Figure 9. Accuracy graph of SVM and LSTM

The primary objective of this study is to develop an improved and reliable above-knee prosthesis that can efficiently assist amputees. The proposed study incorporates several advancements compared to previous research [11].

Table 6. Comparing results with previous study

Factor	Previous Approach [11]	Our Approach
Muscle classification	2 muscles (Femoris & Vastus)	1 muscle (Rectus Femoris0
Features used	6 features for each muscle	22 features
Dimensionality reduction	Not used	PCA
Model type	4 ML	1 ML and 1 DL
Common model type used	SVM, linear kernel (70% accuracy)	SVM, RBF kernel (82% accuracy)
Total movement recognition	2 (Extension & Flexion)	3 movement recognition

Conclusion:

We discussed that the main goal of our project is to design a lower limb exoskeleton/prosthetic device for rehabilitation. To achieve this, we have developed and designed a mechanical model of a robotic leg, which can be attached to an amputee to enhance efficiency. The core objective of this project is to create lower limb prostheses that improve human health, provide comfort, and remain financially affordable. We also examined different methods to enhance the efficiency of conventional prosthetic devices, which are often uncomfortable and less effective.

Our project is implemented in two stages. In the first stage, we focused on signal acquisition and processing to accurately capture muscle signals. Signal processing includes several operations, such as amplifying the acquired signal, filtering out unwanted noise, rectifying the signal, and normalizing it to scale the output between 0 and 1. This improves signal quality, enabling more accurate feature extraction and enhancing the prediction efficiency of the machine learning model.

In the second stage, we worked on the control system of the robotic leg to execute the intended actions. The machine learning model predicts the required signal, which is then sent to the control system to trigger the actuator and perform the corresponding action. By integrating this device into real-world prosthetic applications, it can assist individuals with above-knee amputations. Due to effective signal processing and modeling, the device has demonstrated promising results when tested on new participants. The test outcomes, shown in the above section, highlight its performance. Additionally, the model's classification accuracy could be further improved by employing more advanced deep learning models, such as CNN-based architectures.

Acknowledgement: The authors would like to thank the administration of Intelligent Systems Laboratory, Department of Electrical Engineering, University of Gujrat for providing an opportunity to conduct research and perform experiments with the available laboratory facilities.

Author's Contribution: Muhammad Moeed Zeb, Ali Maesam Kazmi wrote the initial draft of the article with equal participation. Syed Muhammad Wasif supervised and drafted the final version of the article. Zubair Mehmood improved the formatting of the final draft and proof read and corrected the manuscript with the help of Muhammad Jehanzeb Irshad, Muhammad Waqas Jabbar and Nazam Siddique.

Conflict of interest: The authors have no conflict of interest for publishing this manuscript in IJIST.

References:

[1] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050," *Arch. Phys. Med. Rehabil.*, vol. 89, no. 3, pp. 422–429, Mar. 2008, doi: 10.1016/J.APMR.2007.11.005/ASSET/0D660FF8-C4A8-4595-9B40-B78D51E9E385/MAIN.ASSETS/GR2.SML.

[2] L. H. and M. O.-C. B. Ahkami, K. Ahmed, A. Thesleff, "Electromyography-Based Control of Lower Limb Prostheses: A Systematic Review," *IEEE Trans. Med. Robot. Bionics*, vol. 5, no. 3, pp. 547–562, 2023, doi: 10.1109/TMRB.2023.3282325.

[3] T. A. K. Levi J. Hargrove, Ann M. Simon, Aaron J. Young, Robert D. Lipschutz, Suzanne B. Finucane, Douglas G. Smith, "Robotic Leg Control with EMG Decoding in an Amputee with Nerve Transfers," *New Engl. J. Med. homepage*, vol. 369, no. 13, pp. 1237–1242, 2013, doi: 10.1056/NEJMoa1300126.

[4] B. Liu, Z. Chen, and Y. Hu, "Lower Limb Motion Recognition by Integrating Multi-modal Features Based on Machine Learning Method," *ACM Int. Conf. Proceeding Ser.*, Oct. 2020, doi: 10.1145/3424978.3425120.

[5] S.-K. Wu, G. Waycaster, and X. Shen, "Electromyography-based control of active above-knee prostheses," *Control Eng. Pract.*, vol. 19, no. 8, pp. 875–882, 2011, doi: https://doi.org/10.1016/j.conengprac.2011.04.017.

[6] L. Peeraer, B. Aeyels, and G. Van der Perre, "Development of EMG-based mode and intent recognition algorithms for a computer-controlled above-knee prosthesis," *J. Biomed. Eng.*, vol. 12, no. 3, pp. 178–182, 1990, doi: https://doi.org/10.1016/0141-5425(90)90037-N.

[7] E. D. A. A. Nadeem Shah, Abbas Khan, Hira Zahid, "Design and Development of Human Knee Joint Muscle(s) Classification System using Machine Learning Technique," ResearchGate. Accessed: Mar. 13, 2025. [Online]. Available: https://www.researchgate.net/publication/348550210_Design_and_Development_of_Human_Knee_Joint_Muscles_Classification_System_using_Machine_Learning_Technique

[8] Q. W. Honglei An, Hongxu Ma, "Novel Feature Extraction and Locomotion Mode Classification Using Intelligent Lower-Limb Prosthesis," *Machines*, vol. 11, no. 2, p. 235, 2023, doi: https://doi.org/10.3390/machines11020235.

[9] I. S. Dhindsa, R. Agarwal, and H. S. Ryait, "Performance evaluation of various classifiers for predicting knee angle from electromyography signals," *Expert Syst.*, vol. 36, no. 3, p. e12381, Jun. 2019, doi: 10.1111/EXSY.12381.

[10] A. A. D.-S. Tahir Hussain, Nadeem Iqbal, Hafiz Farhan Maqbool, Mukhtaj Khan, Mohammed Ibrahim Awad, "Intent based recognition of walking and ramp activities for amputee using sEMG based lower limb prostheses," *Biocybern. Biomed. Eng.*, vol. 40, no. 3, pp. 1110–1123, 2020, doi: https://doi.org/10.1016/j.bbe.2020.05.010.

[11] R. D. Babu, S. S. Adithya, and M. Dhanalakshmi, "Design and development of an EMG controlled transfemoral prosthesis," *Meas. Sensors*, vol. 36, p. 101399, 2024, doi: https://doi.org/10.1016/j.measen.2024.101399.

[12] R. Schulte, "Roessingh Research & Development-MyLeg database for activity prediction (MyPredict) (dataset)," ResearchData. dataset. Accessed: Mar. 13, 2025. [Online]. Available: https://data.4tu.nl/datasets/01d30db7-95a8-4c39-afb9-4eb1a2f27539