

Research Article

Hybrid Deep Learning Approaches for sEMG Signal-Based Lower Limb Activity Recognition

Ankit Vijayvargiya ^{1,2}, Bharat Singh ¹, Rajesh Kumar ¹, Usha Desai ³,
and Jude Hemanth ⁴

¹Malaviya National Institute of Technology, Jaipur, India

²Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur, India

³Department of Electronics and Communication Engineering, SR University, Warangal, Telangana, India

⁴Department of Electronics and Communication Engineering, Karunya University, Coimbatore, Tamilnadu, India

Correspondence should be addressed to Jude Hemanth; judehemanth@karunya.edu

Received 6 July 2022; Revised 16 October 2022; Accepted 28 October 2022; Published 26 November 2022

Academic Editor: Paolo Crippa

Copyright © 2022 Ankit Vijayvargiya et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Lower limb activity recognition utilizing body sensor data has attracted researchers due to its practical applications, such as neuromuscular disease detection and kinesiological investigations. The employment of wearable sensors including accelerometers, gyroscopes, and surface electromyography has grown due to their low cost and broad applicability. Electromyography (EMG) sensors are preferable for automated control of a lower limb exoskeleton or prosthesis since they detect the signal beforehand and allow faster movement detection. The study presents hybrid deep learning models for lower limb activity recognition. Noise is suppressed using discrete wavelet transform, and then the signal is segmented using overlapping windowing. Convolutional neural network is used for temporal learning, whereas long short-term memory or gated recurrent unit is used for sequence learning. After that, performance indices of the models such as accuracy, sensitivity, specificity, and F-score are calculated. The findings indicate that the suggested hybrid model outperforms the individual models.

1. Introduction

Lower limb activity recognition (LLAR) has increased in popularity due to its ability to monitor or identify daily lower limb human actions in a range of applications such as elderly health monitoring, surveillance and security systems, human fall detection, and so on [1, 2]. The two methodologies utilized for acquiring human activity data are visual and wearable sensors [3]. Wearable sensors such as inertial measurement units, goniometers, and sEMG electrodes are placed on the subject's body for data collection [4]. The vision-based approach has limited capability in terms of applicability, security and complexity [5]. Wearable sensors have seen significant technical advancements in recent times. It results in a lower overall cost, making it more accessible. Popular sensors used in wearable research for activity recognition include inertial measurement units,

accelerometers, gyroscopes, electromyography, and barometers [6]. EMG sensors are better than others since they can predict movement in advance in a very short amount of time. Out of these sensors, the EMG sensors are superior because they can anticipate movement in advance in a very short amount of time [7, 8]. Neuromuscular activity generates the biological signal known as the EMG signal. It can be detected by the electrical currents in muscles during the muscle contraction. Surface (non-invasive) EMG and intramuscular (invasive) EMG are the two approaches that are employed for the recording of the EMG signal [9]. Intramuscular electromyogram (iEMG) signals are captured by placing the wire electrodes within the muscles, whereas surface electromyogram (sEMG) signals are captured by placing the surface electrodes just above the muscle's surface. The following are the advantages of sEMG over the iEMG [10]:

- (1) There is no requirement of medical supervision for placing the electrodes; also, there is no discomfort.
- (2) Infection risk is mitigated.

Surface EMG signals are used in a wide range of healthcare applications, which include the control of prostheses or exoskeletons, neuromuscular disease assessment, activity monitoring, and many more [11–13]. According to Kiguchi et al., sEMG signal-based neuro-fuzzy approach can be used to control an upper limb robotic exoskeleton [14]. It is found that sEMG signals can be used for multiple applications. Krasin et al. [15] proposed the low-cost elbow joint powered exoskeleton. The major goal of this sEMG signal-based exoskeleton is for strengthening the biceps brachii. Sharmila et al. [16] presented a low-cost sEMG-controlled prosthetic arm for upper limb amputees. Sensors are employed for the recording of the sEMG signals from the muscles during various activities to control prostheses autonomously. Then, the actuators can be controlled using artificial intelligence methodology. Pancholi and others [17] developed hardware for amputees to recognize real-time arm gestures. Vijayvargiya et al. developed low-cost sEMG data acquisition system for collecting the sEMG signals [18]. Cai et al. [19] identified the upper limb motion pattern for controlling a rehabilitation robot using sEMG data and a support vector machine approach.

Compared to lower limb sEMG signals, much emphasis has been dedicated to the classification and pattern recognition of upper limb sEMG signals in the recent decade. Classification of lower limb sEMG signals has shown to be more complicated than the classification of upper limb signals. It is because of the complexity induced due to the inherent coupling of lower limb sEMG signals. Souit et al. [20] presented the control approach using sEMG for a lower limb exoskeleton. The exoskeleton can be operated autonomously utilizing artificial intelligence approaches by analyzing the sEMG signal produced by muscles during various activities. Khimraj et al. [21] investigated classification between six lower limb activities and evaluated the performance of various machine learning classifiers for the same. Silva et al. [22] carried out research investigation on spinal cord damage based on the EMG signal that has been captured while the upper limb movements were being performed. Vijayvargiya et al. [23] investigated the detection of knee abnormalities using unbalanced sEMG data for walking activity. Here, the authors demonstrated the impact of an imbalanced signal on the model performance for the detection of a knee issue and evaluated the performance of several oversampling methods to improve machine learning model performance. Ertugrul et al. [24] proposed an adaptive local binary pattern (ALBP) approach for the retrieval of the characteristics and classification of healthy and abnormal knee participants with an accuracy of 85%. Handcrafted characteristics from signals are required for machine learning models, which may be extracted using statistical methods. Choosing the proper feature set manually is a tedious process. According to the available literature, deep learning models such as CNN, LSTM, GRU, and other techniques have been employed to solve the problem

[25]. The features are retrieved first by the algorithm in deep learning techniques, and then the classification procedure is carried out. According to earlier studies, these deep learning models have been used for various applications and have demonstrated very high performance [26–30].

This study aims to apply hybrid deep learning algorithms to identify lower limb activity. The authors present the models that integrate the advantage of convolutional neural network (CNN), long short-term memory (LSTM), and gated recurrent unit (GRU). CNN architecture has multiple layers such as the input layer, output layer, dense layer, convolutional layer, rectified linear unit layers, and dropout layers. One of the significant issues with CNN is its inability to analyze the characteristics of the time-series data such as the previous or temporal data. Therefore, the LSTM or GRU can be employed for the analysis of data which efficiently collects temporal information present in the data. As a result, a combination of CNN with LSTM (CNN-LSTM) or CNN with GRU (CNN-GRU) is proposed as the better strategy for processing EMG data. In the suggested hybrid models, a convolutional neural network provides temporal learning, whereas the LSTM or GRU is used for capturing sequence-to-sequence learning. The major contributions of this study are as follows:

- (1) Surface electromyography (sEMG) data acquired from leg muscles are used in this study to examine lower limb movements in healthy and knee deformity individuals by using the hybrid deep learning framework.
- (2) The preprocessing technique known as wavelet denoising is applied in order to eliminate noises of sEMG signal.
- (3) Hybrid deep learning models, CNN-LSTM and CNN-GRU, are proposed for recognition of lower limb activities. Here, a convolutional neural network (CNN) is used for temporal learning, while long short-term memory (LSTM) or gated recurrent unit (GRU) is used for sequence learning.
- (4) The proposed hybrid CNN-GRU model has high performance compared to the existing models.

2. Dataset

The authors used the publicly accessible sEMG signal dataset from the UCI machine learning repository by Sanchez et al. in their study [24, 31]. The data comprise sEMG signals from the lower extremities of 22 individuals over the age of 18, 11 of which have been healthy and 11 of which have known knee injuries. There is no history of knee injury or discomfort in the healthy participants. The sciatic nerve was injured in one abnormal knee subject, the anterior cruciate ligament (ACL) was injured in six abnormal knee subjects, and the meniscus was injured in the remaining four abnormal knee subjects. The sEMG signals were acquired using a Biometrics Ltd. DataLog MWX8 and a goniometer while the participants performed one of three tasks: walking, sitting, and standing. The biceps femoris (BF), vastus

medialis (VM), rectus femoris (RF), and semitendinosus (ST) muscles had their sEMG data taken with goniometer affixed towards the outer surface of the knee. The sEMG signal was recorded on the damaged limb of the person with a defective knee and the left leg of the healthy individuals. The data were collected at a sample rate of 1 kHz with a resolution of 14 bits. A band-pass filter with a passband frequency of 20 Hz to 460 Hz has already been applied to the sEMG signals. The sEMG signals collected throughout each activity by the healthy and abnormal knee subjects are shown in Figure 1.

3. Proposed Methodology

This section explains the methods used for lower limb activity recognition (wavelet denoising, segmentation, and deep learning frameworks). Figure 2 depicts the proposed deep learning-based method for the identification of the lower limb activity based on the sEMG signals. First, noise is removed from the raw sEMG signal with the help of discrete wavelet transform, and then the signal is segmented using the overlapping windowing technique. After that, deep learning models CNN, CNN-LSTM, and CNN-GRU are applied to identify the lower limb activities in healthy and abnormal knee individuals. In these hybrid models, a convolutional neural network (CNN) is used for temporal learning, while long short-term memory (LSTM) or gated recurrent unit (GRU) is used for sequence learning.

3.1. Wavelet Denoising. Various types of noises have been interlaced with the sEMG signals during the recording. The most prominent noises are as follows [32]:

- (i) Electronic devices and electromagnetic interference generate inherent and ambient noises, respectively.
- (ii) The subject's walking leads to electrodes' movement, which induces the artifacts.
- (iii) The firing rate of motor units can lead to the inherent instability noise.

These noises can badly impact the performance of the classifier. Therefore, it is required to filter the noise from the sEMG signal. There are various topologies proposed by various researchers in the literature. Traditional approaches such as low pass, high pass, and band pass can be employed for the filtering of noises that do not fall within the frequency band of the sEMG signal, which is 20 to 460 Hz. However, these methods have failed to filter noise in the active spectrum of sEMG signals. In recent times, several researchers have successfully employed empirical mode decomposition, independent component analysis, and wavelet decomposition for the filtering of noises from the sEMG signals [33].

The use of wavelet decomposition [34, 35] has seen a rising trend in sEMG signal denoising for both the upper and lower limbs. This is because it can effectively eliminate the white Gaussian noise from the signal. In the wavelet method, firstly, the mother wavelet function is selected. Then, the frequency and temporal analysis can be performed

by the low and high-frequency versions of the wavelet, respectively. Wavelet decomposition includes the following steps:

- (i) The signal is decomposed using the discrete wavelet transform.
- (ii) A threshold is chosen.
- (iii) The signal is reconstructed using the inverse wavelet transform.

Decomposition generates the approximation and detail coefficients. The level of decomposition decides the number of the coefficients. After this, thresholding is applied. Thresholding forced all coefficients to zero, which is below a certain threshold. Afterward, the signal is reconstructed again. By scaling (s) and translating (τ) a single basic wavelet $\Psi_{s,\tau}$, many mother wavelets are created. The mathematical expression of basic wavelet is presented in the following equation:

$$\Psi_{s,\tau} = \frac{1}{\sqrt{s}} \Psi\left(\frac{t-\tau}{s}\right). \quad (1)$$

As per literature, multiple thresholding approaches are employed, such as hard, soft, and universal. In this work, the authors have employed a db4 mother wavelet from the Daubechies family with four decomposition levels and applied garotte thresholding on second detail coefficients.

3.2. Segmentation. Since the sEMG signal's characteristic is stochastic, it is suggested by Vijayvargiya et al. to segment the sEMG signal into smaller portions. However, different lengths of sEMG signal impact the classifier accuracy. Therefore, commonly used windowing topologies such as adjacent and overlapping are employed on the signal segmentation. Here, the authors have used the overlapping-based windowing technique with a window size of 256 ms and 25% overlapping [36].

3.3. Deep Learning Framework

3.3.1. Convolutional Neural Network (CNN). Generally, the artificial neural network has a fully connected network of neurons, which means every neuron is connected via weights to the next layer completely. This connectivity can lead to overfitting of the network. To address this problem, several researchers in the literature have presented regularization techniques, which involve the magnitude of the weight-to-loss ratio. Similarly, CNN has been adopted for regularization because of its ability to progressively exploit the dataset's structure by gathering the simple structures. It uses the principle of convolution instead of matrix pointwise multiplication. An additional advantage is that these networks have fewer parameters as compared to other fully connected neural networks. Therefore, the training of these networks is fast. A CNN includes a sequence of multiple layers such as input, hidden, and output layers. Hidden layers constitute the three different types of layers which have convolutional, max-pooling, and fully connected

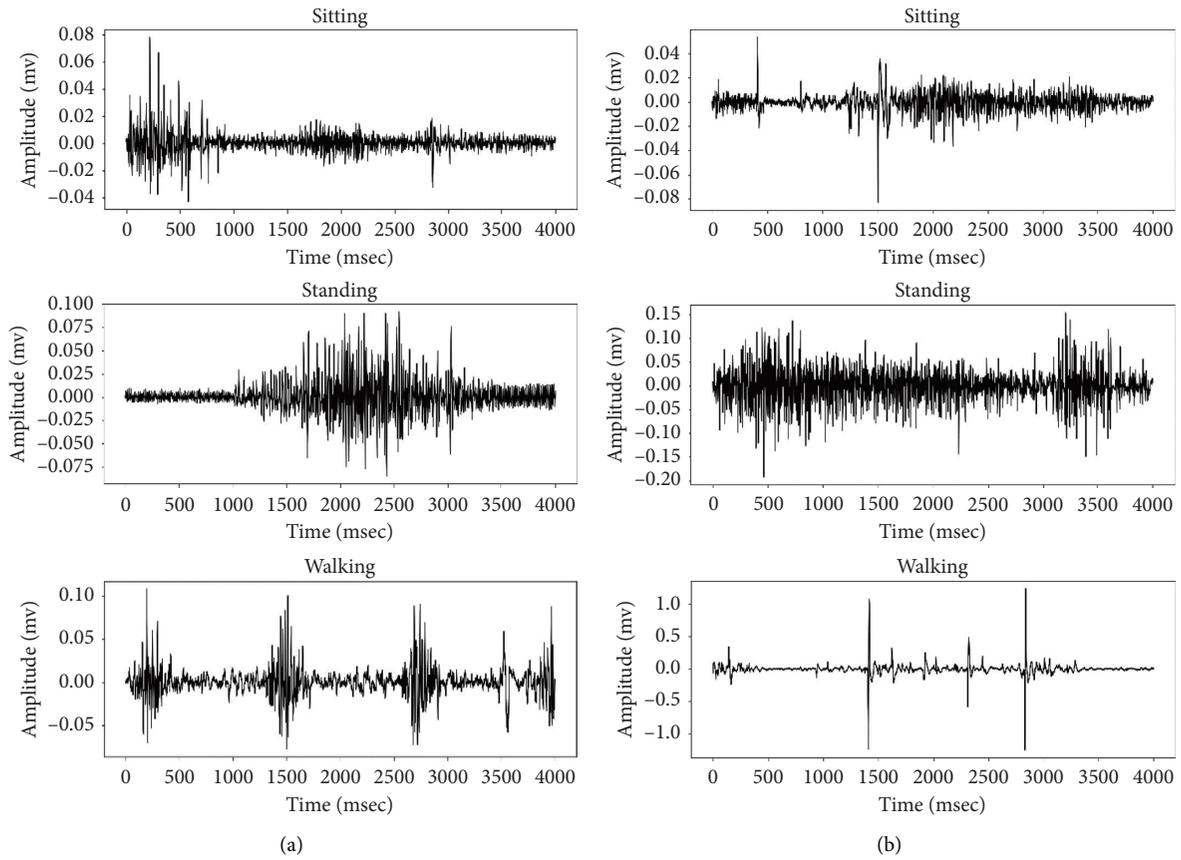


FIGURE 1: sEMG signals acquired during walking, standing, and sitting: (a) normal subject and (b) abnormal subject.

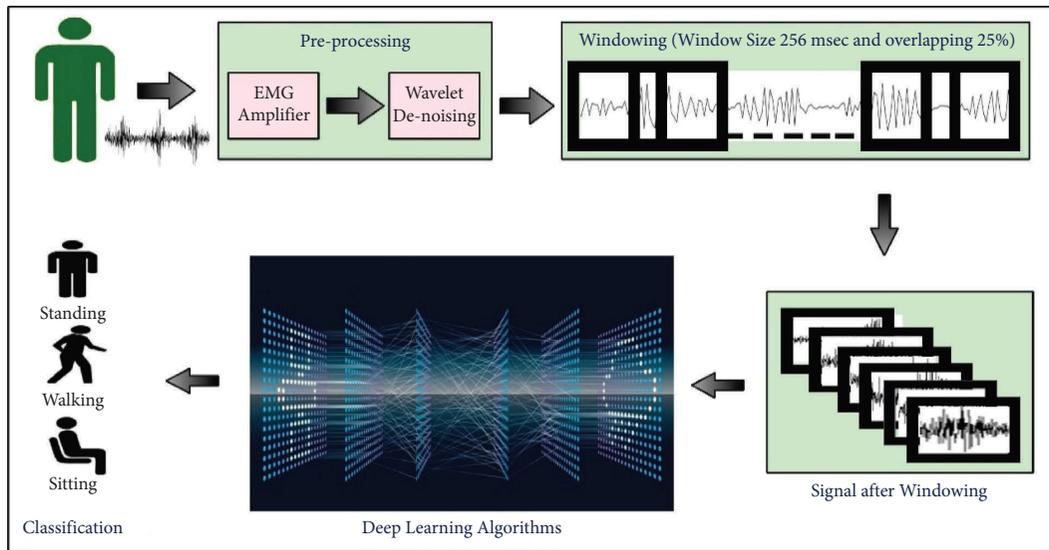


FIGURE 2: Block diagram of the proposed methodology for lower limb activity.

layers. The convolutional layer convolves with multiplication. Generally, ReLU is actively employed as the activation function in the neuron.

Mainly, CNN is employed on 2D datasets such as video or images. Therefore, CNN is named as the 2D CNN. As per

the literature, some authors have modified 2D CNN to 1D CNN [37].

Mathematical expressions of layers of CNN are described in equations (2) to (9).

(1) *Convolution Layer.*

(i) Forward propagation:

$$a_{ij}^{(k)} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} W_{st}^{(k)} x_{(i+s)(j+t)} + b^{(k)}. \quad (2)$$

(ii) Backpropagation to update the weight:

$$\begin{aligned} \frac{\delta E}{\delta W_{st}^{(k)}} &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\delta E}{\delta a_{ij}^{(k)}} \frac{\delta a_{ij}^{(k)}}{\delta W_{st}^{(k)}} \\ &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\delta E}{\delta a_{ij}^{(k)}} x_{(i+s)(j+t)}, \end{aligned} \quad (3)$$

$$\begin{aligned} \frac{\delta E}{\delta b^{(k)}} &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\delta E}{\delta a_{ij}^{(k)}} \frac{\delta a_{ij}^{(k)}}{\delta b^{(k)}} \\ &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\delta E}{\delta a_{ij}^{(k)}}. \end{aligned} \quad (4)$$

(iii) Backpropagation to previous layer:

$$\begin{aligned} \frac{\delta E}{\delta x_{ij}} &= \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} \frac{\delta E}{\delta a_{(i-s)(j-t)}^{(k)}} \frac{\delta a_{(i-s)(j-t)}^{(k)}}{\delta x_{ij}} \\ &= \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} \frac{\delta E}{\delta a_{(i-s)(j-t)}^{(k)}} W_{st}^{(k)}. \end{aligned} \quad (5)$$

(2) *Max-Pooling Layer.*

(i) Forward propagation:

$$a_{ij} = \max(0, x_{(i+s)(j+t)}). \quad (6)$$

(ii) Backpropagation:

$$\begin{aligned} \frac{\delta E}{\delta x_{(i+s)(j+t)}} &= \frac{\delta E}{\delta a_{ij}^{(k)}}, \\ \frac{\delta a_{ij}^{(k)}}{\delta x_{(i+s)(j+t)}} &= \begin{cases} \frac{\delta E}{\delta a_{ij}^{(k)}}, & \text{if } a_{ij}^k = x_{(i+s)(j+t)}, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (7)$$

(3) *Fully Connected Layer.*

(i) Forward propagation of ReLU activation function:

$$a_{ij} = \max(0, x_{ij}). \quad (8)$$

(ii) Backpropagation of ReLU activation function:

$$\frac{\delta E}{\delta x_{ij}} = \frac{\delta E}{\delta a_{ij}^{(k)}} \frac{\delta a_{ij}^{(k)}}{\delta x_{ij}} = \begin{cases} \frac{\delta E}{\delta a_{ij}^{(k)}}, & \text{if } a_{ij}^k \geq 0, \\ 0, & \text{otherwise,} \end{cases} \quad (9)$$

where x is the input, a^k denotes the output after convolution layer k , k denotes the layer index, W denotes the kernel (filter), $m * n$ denotes the filter size, $M * N$ denotes the input size, b denotes the bias, and E denotes the cost function.

3.3.2. Long Short-Term Memory (LSTM). Conventional neural networks failed to achieve the desired accuracy for sequential time-series datasets because they do not have any memory element to store the previous state. This means it only relies on the present state. Therefore, the recurrent neural network allows a feedback mechanism to learn the previous step state. This structure of recurrent provides better handling of the time-series data. However, it can often encounter a problem when the network becomes deeper. Vector-Jacobian gradient vector tends to shrink due to its repetitive nature. This can lead to training being slower and less effective. This phenomenon is known to be a gradient vanishing problem. Thus, these recurrent neural networks perform badly in terms of long-term dependency. Thus, the long short-term memory (LSTM) [38] can effectively deal with the gradient vanishing problem by tackling the long-term dependencies.

The architecture of LSTM consists of an LSTM cell. Each cell contains a well-designed vectorized pointwise multiplication between the new and the previous states. In addition, mathematical functions such as the hyperbolic tan and sigmoid functions are employed to control the flow of information instead of only a single layer as in simple recurrent layers. The LSTM cell consists of the three operational gates with memory element via feedback, and each gate has its bias and weight vectors. Three gates are forget gate, input/update gate, and output gate. Firstly, the current and previous activation state information flows through the sigmoid function, and this layer is known to be the forget layer. This function squashed the information into the range of 0 and 1. This operation quantifies how much information is required for the next step prediction. Secondly, the processed information from the forget gate and input vector flows through the hyperbolic tangent function. This layer is known as the input/update layer because it uses the previous state and present state to generate the new information for the cell. Finally, the linear vector-matrix addition is performed. Then, the information flowed through the hyperbolic tangent. Afterward, the output gate scaled the resultant final values from the previous gate. Overall, the sigmoid function controls the flow such as which information is required or what is needed to forget from the previous cell. It

significantly increases the accuracy. The mathematical expression is given in equations (10) to (15).

Forget gate:

$$F_t = \sigma(W_{fx}X_t + W_{fh}H_{t-1} + B_f). \quad (10)$$

Input gate:

$$I_t = \sigma(W_{ix}X_t + W_{ih}H_{t-1} + B_i). \quad (11)$$

Output gate:

$$O_t = \sigma(W_{ox}X_t + W_{oh}H_{t-1} + B_o). \quad (12)$$

Intermediate state:

$$\check{C}_t = \tanh(W_cX_t + W_{ch}H_{t-1} + B_c). \quad (13)$$

Final state:

$$C_t = F_t \odot C_{t-1} + I_t \odot \check{C}_t. \quad (14)$$

New state:

$$H_t = O_t \odot \tanh(C_t), \quad (15)$$

where W and B are the layer weight and bias vectors, respectively, and the input vector is denoted by X .

3.3.3. Gated Recurrent Unit (GRU). The gated recurrent unit [39] is a variant of the LSTM. Therefore, it has similar properties as the LSTM, and it also solves the vanishing gradient problem. This overall improves the learning long-term capability of the neural network. It also has the sigmoid and hyperbolic tangent functions. It does not have a separate forget and input gate like LSTM, which means not having a separate memory element. Therefore, it has fewer parameters as compared to the LSTM cell. It yields efficiently compared to the former. The update gate consists of both sigmoid and hyperbolic tangent functions. In addition, a reset gate is also designed in the GRU cell. All process flow is the same as the LSTM cell. The mathematical expression is given in equations (16) to (19).

Update gate:

$$U_t = \sigma(W_{ux}X_t + W_{uh}H_{t-1} + B_u). \quad (16)$$

Reset gate:

$$R_t = \sigma(W_{rx}X_t + W_{rh}H_{t-1} + B_r). \quad (17)$$

Internal activation:

$$\check{H}_t = \tanh(W_{hx}X_t + W_{hh}(R_t \odot H_{t-1}) + B_h). \quad (18)$$

Output activation:

$$H_t = U_t \odot \check{H}_t + (1 - U_t) \odot \check{H}_{t-1}, \quad (19)$$

where W and B represent the respective layer weight and bias vector and X represents the input vector.

3.3.4. Proposed Hybrid Models. One of the most serious shortcomings of CNN is its inability to analyze historical or

temporal data contained within time-series signal, as it lacks a memory element for storing the previous state. This means that it is entirely dependent on the current situation. Thus, a feedback mechanism enables the recurrent neural network to learn the previous step state. This recurrent structure enables more efficient handling of time-series data. However, it frequently encounters a gradient vanishing challenge and performs badly in long-term dependency. By addressing long-term dependencies, long short-term memory or gated recurrent unit, the variants of the recurrent neural network, can successfully address the gradient vanishing problem.

In this study, the authors proposed a hybrid of a convolutional neural network with long short-term memory and gated recurrent unit to recognize lower limb activities. CNN is employed for capturing the temporal relationship present in the dataset, whereas the LSTM or GRU is employed for sequence-to-sequence learning. The authors have used the dataset collected from the four channels collected at a sampling frequency of 1 kHz. Then, the signal is segmented as 256 ms. Thus, the total window length is equal to 256 samples. Therefore, the four channels are connected serially. It results in an array size of 1×1024 , which can be applied as the input for the 1D CNN. Firstly, the input features are normalized in range of 0 to 1. Then, the normalized signal is passed through the two convolutional layers which consist of the convolutional and non-linear layers (ReLU). Afterward, the processed data are passed through the max-pooling layer and then followed through two LSTM or GRU layers. After that, the signal is passed through the two fully connected layers. Feature dimension is reduced by integrating the pooling layer after the convolutional layer. The parameters of studied deep learning models are shown in Table 1 which are found out using the trial-and-error approach.

4. Results and Discussion

We show our findings for several situations involving lower limb activity recognition in this section. The suggested model was trained and evaluated using TPU Google Colab, a cloud-based system to detect a human's lower limb actions. The model's code within Colab notebook executes on a Google cloud server. Python's Keras modules have been used to recognize the various 1D CNN models, and the outcomes were computed. The Adam optimizer integrates the classic backpropagation approach with a cross-entropy loss function and employs the stochastic gradient descent strategy. The hyperparameters of the optimizer have the following values: learning rate (0.001), epsilon (0.00000001), beta 1 (0.9), beta 2 (0.999), and locking (false).

In this investigation, total of 22 subjects consisting of normal persons and knee difficulty persons in equal numbers were examined. The performance indicators were evaluated on (1) healthy adults and (2) individuals with a knee anomaly. The authors have considered starting 70% of the signal for every subject as a training dataset and the

TABLE 1: Parameters considered in the proposed study using deep learning models.

Model parameters		CNN	LSTM	GRU	CNN-LSTM	CNN-GRU
Convolution layer 1	Number of filters	16	—	—	16	16
	Kernel size	3	—	—	3	3
Convolution layer 1	Number of filters	8	—	—	8	8
	Kernel size	3	—	—	3	3
Pooling layer	Type of pooling	Max	—	—	Max	Max
	Kernel size	2	—	—	2	2
LSTM	Unit size	—	16	—	16	—
LSTM	Unit size	—	8	—	8	—
GRU	Unit size	—	—	16	—	16
GRU	Unit size	—	—	8	—	8
Fully connected	Unit size	16	16	16	16	16
Fully connected	Unit size	3	3	3	3	3

remaining 30% as a testing dataset. It helps in the reduction of the temporal dependencies and arbitrarily tests set selection. The authors have considered five performance metrics such as accuracy, precision, sensitivity, specificity, and F-score like the previous study of Vijayvargiya et al. [40].

Table 2 shows the summary indices of the studied deep learning models in percentage for the three activities such as walking, sitting, and standing, under research obtained from sEMG data gathered from healthy people, whereas Table 3 shows the performance for knee abnormality subjects. The data in these tables allow the individual models to be compared in terms of participants with and without knee abnormalities, confirming that the suggested hybrid models outperform the individual model. In the case of the healthy subjects, the CNN-GRU model obtained an accuracy of 99.86% while for CNN, LSTM, GRU, and CNN-LSTM, it was 98.88%, 95.38%, 98.32%, and 99.02%, respectively, as indicated in Table 2. Similarly, F-score value of CNN-GRU model was 99.79% while for CNN, LSTM, GRU, and CNN-LSTM, it was 98.31%, 93.82%, 97.51%, and 98.50%, respectively. In the case of the abnormal knee subjects, the CNN-GRU model obtained an accuracy of 98.69% while for CNN, LSTM, GRU, and CNN-LSTM, it was 92.62%, 54.54%, 96.69%, and 97.62%, respectively, as indicated in Table 3. Similarly, F-score value of the CNN-GRU model was 98.61% while for CNN, LSTM, GRU, and CNN-LSTM, it was 92.22%, 47.64%, 96.55%, and 97.42%, respectively.

Table 4 presents the time taken to complete an epoch of studied deep learning models. It indicates that when the CNN layer is used, the time taken for completion of an epoch is less required than the individual LSTM or GRU model. In healthy participants, the proposed hybrid CNN-GRU and CNN-LSTM deep learning models needed 4.26 s and 4.34 s computing time to execute an epoch, respectively. In contrast, the CNN, LSTM, and GRU deep learning models required 2.26 s, 94.06 s, and 117.60 s. In abnormal knee subjects, the proposed hybrid CNN-GRU and CNN-LSTM deep learning models needed 4.58 s and 6.15 s computing time to execute an epoch, respectively, whereas the CNN, LSTM, and GRU deep learning models required 5.19 s, 98.87 s, and 106.67 s. When the proposed hybrid models were examined in terms of computational

TABLE 2: Performance parameters for healthy subjects (in %).

Approach/Parameters	CNN	LSTM	GRU	CNN-LSTM	CNN-GRU
Accuracy	98.88	95.38	98.32	99.02	99.86
Precision	98.49	93.81	97.49	99.20	99.88
Recall/sensitivity	98.13	93.86	97.53	97.88	99.70
Specificity	99.45	97.73	99.20	99.46	99.92
F-score	98.31	93.82	97.51	98.50	99.79

TABLE 3: Performance parameters for abnormal knee subjects (in %).

Approach/Parameters	CNN	LSTM	GRU	CNN-LSTM	CNN-GRU
Accuracy	92.62	54.54	96.69	97.62	98.69
Precision	92.67	79.63	96.50	97.35	98.61
Recall/sensitivity	91.92	51.60	96.62	97.54	98.62
Specificity	96.22	75.31	98.37	98.87	99.36
F-score	92.22	47.64	96.55	97.42	98.61

time, a significant difference was observed for the hybrid models (CNN-LSTM and CNN-GRU) and individual models LSTM and GRU, and a very small variance was observed between the hybrid models and CNN, but the accuracy and F-score values are relatively high in hybrid models than in individual models.

Figure 3 presents the confusion matrices for the CNN, LSTM, GRU, CNN-LSTM, and CNN-GRU models for healthy patients, whereas Figure 4 depicts the confusion matrices obtained for abnormal knee subjects. The confusion matrix provides a tabular representation of the performance of a classification method. It comprises the data on the real and predicted labels from the model. As illustrated in Figure 3(a), 105, 282, and 320 samples of walking, sitting, and standing activities are accurately predicted, but five samples of walking activity are incorrectly predicted as sitting and three samples of sitting activity are incorrectly predicted as walking.

Figure 5 depicts the variation between the loss vs. epoch for the CNN, LSTM, GRU, CNN-LSTM, and CNN-GRU models for healthy patients, whereas Figure 6 depicts the training plot between loss vs. epoch for abnormal knee

TABLE 4: Time taken per epoch (in seconds).

Approach/subjects	CNN	LSTM	GRU	CNN-LSTM	CNN-GRU
Healthy subjects	2.26	94.06	117.60	4.34	4.26
Abnormal knee subjects	5.19	98.87	106.67	6.15	4.58

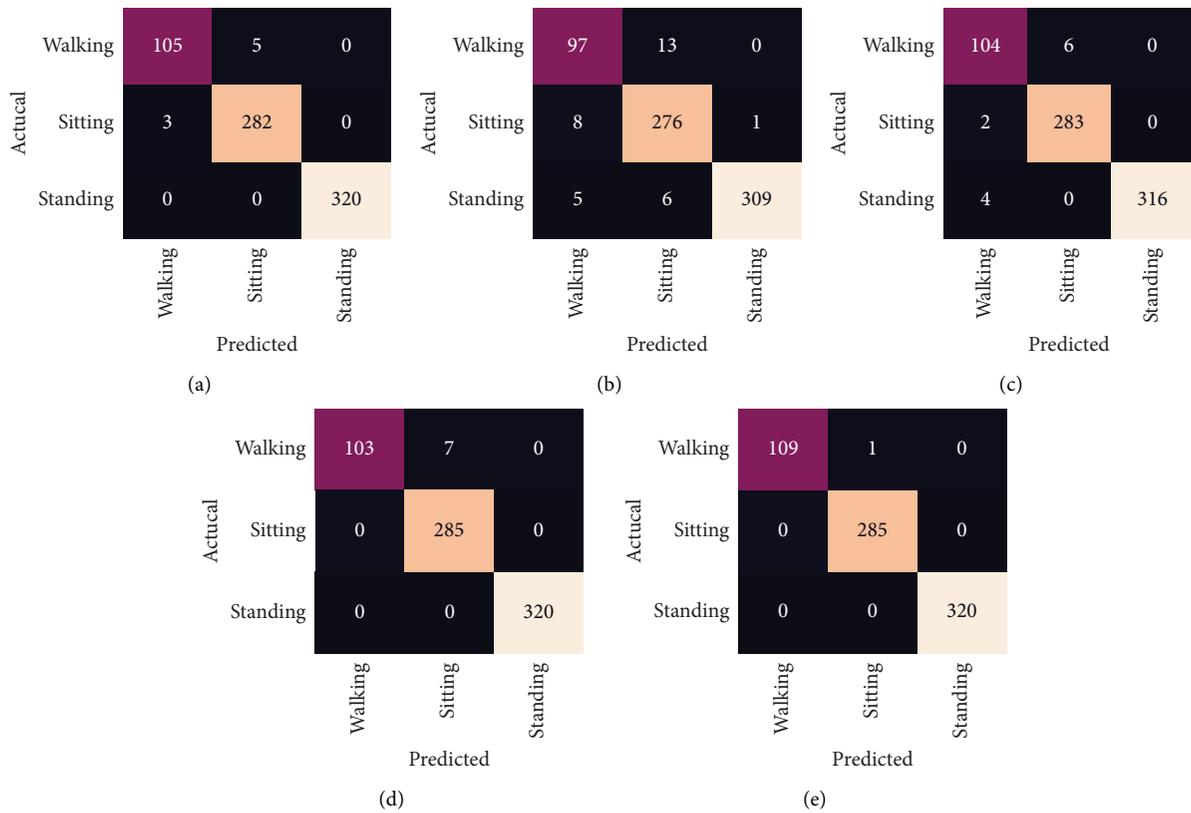


FIGURE 3: Confusion metrics for healthy subjects. (a) CNN. (b) LSTM. (c) GRU. (d) CNN-LSTM. (e) CNN-GRU.

subjects. These plots demonstrate that the value of the loss function decreases as the number of epoch increases, until it approaches a steady state, which indicates that the overfitting problem is resolved. The proposed hybrid models achieve a steady state as they approach the tenth epoch, which is significantly better than the other examined models.

Numerous approaches for recognizing lower limb activity have been presented using similar datasets. Table 5 presents a comparison assessment of the proposed model's performance vs. prior findings, allowing us to conclude that the proposed CNN-GRU model performed well for recognizing lower limb activity in healthy and abnormal knee individuals.

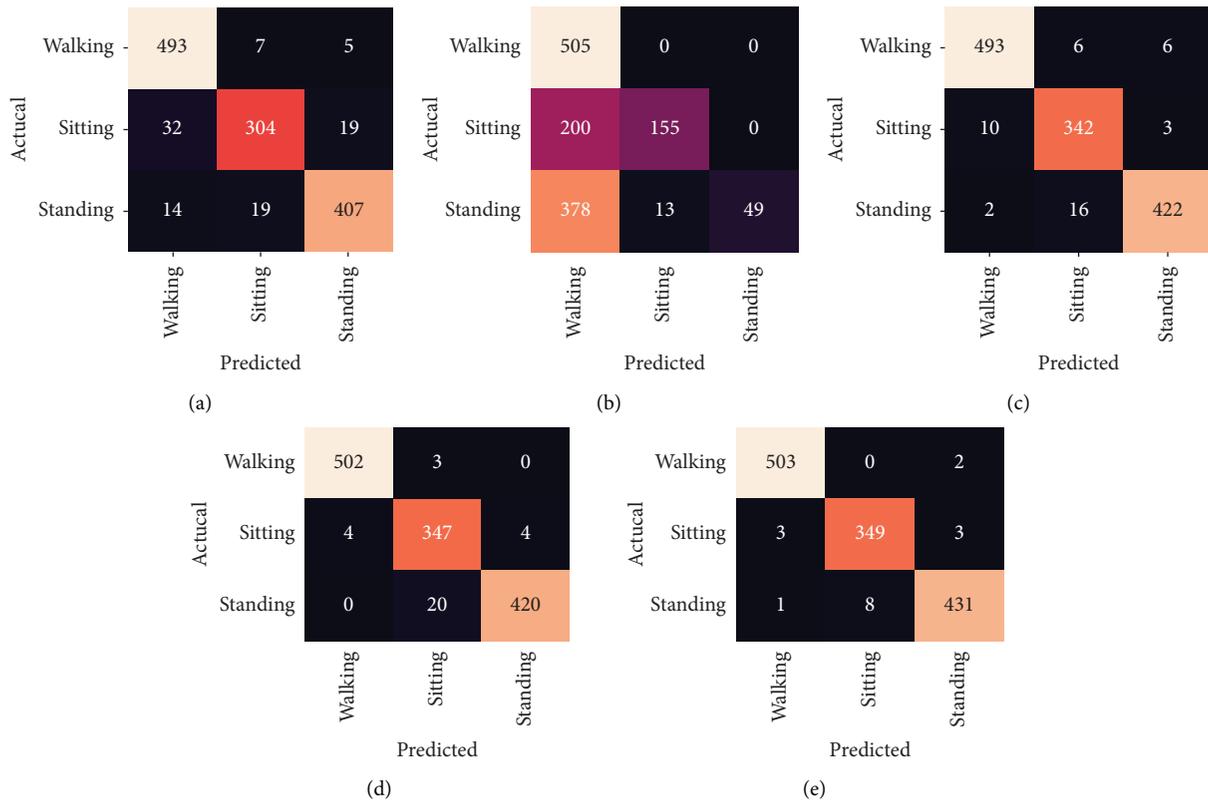


FIGURE 4: Confusion metrics for abnormal knee subjects. (a) CNN. (b) LSTM. (c) GRU. (d) CNN-LSTM. (e) CNN-GRU.

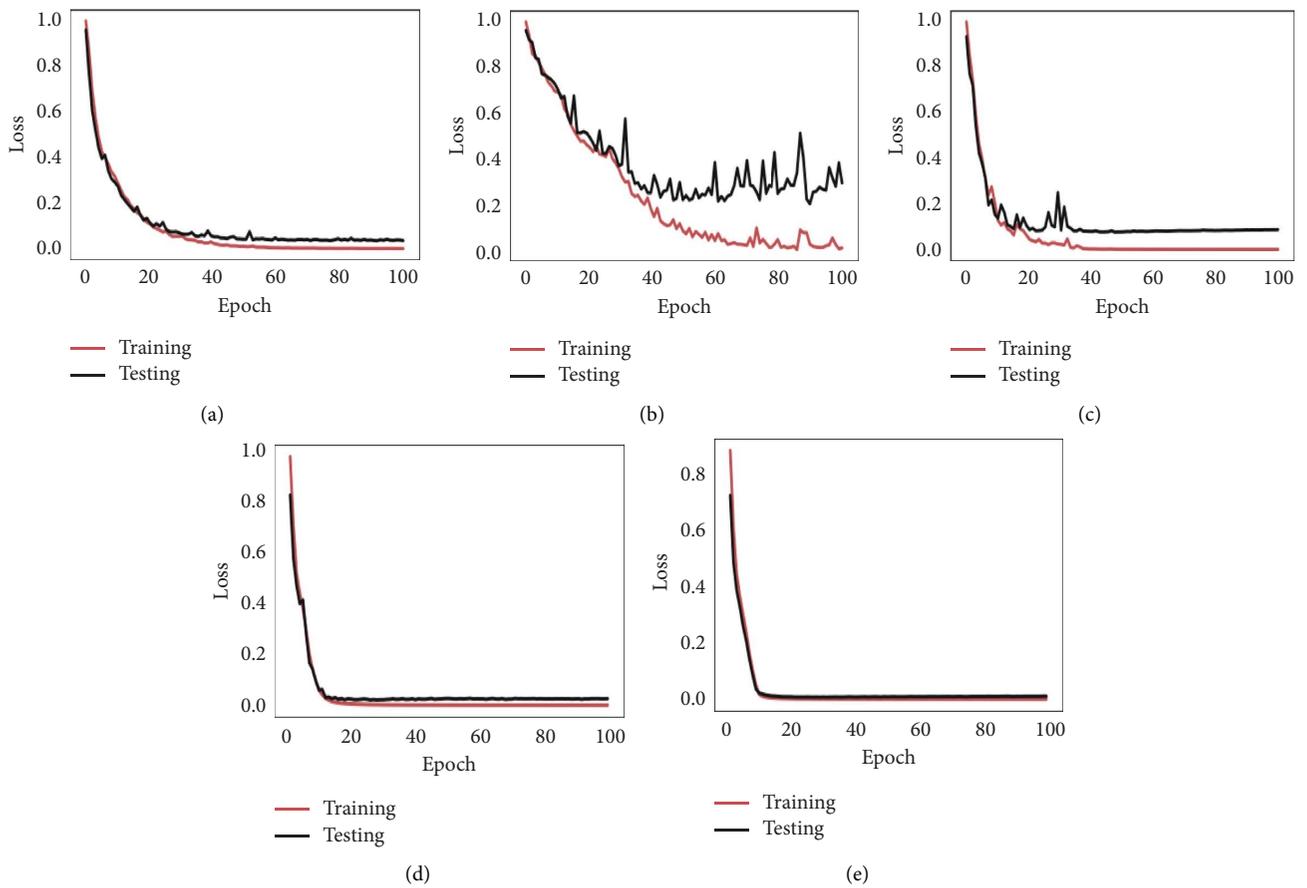


FIGURE 5: Loss vs. epoch curve for healthy subjects. (a) CNN. (b) LSTM. (c) GRU. (d) CNN-LSTM. (e) CNN-GRU.

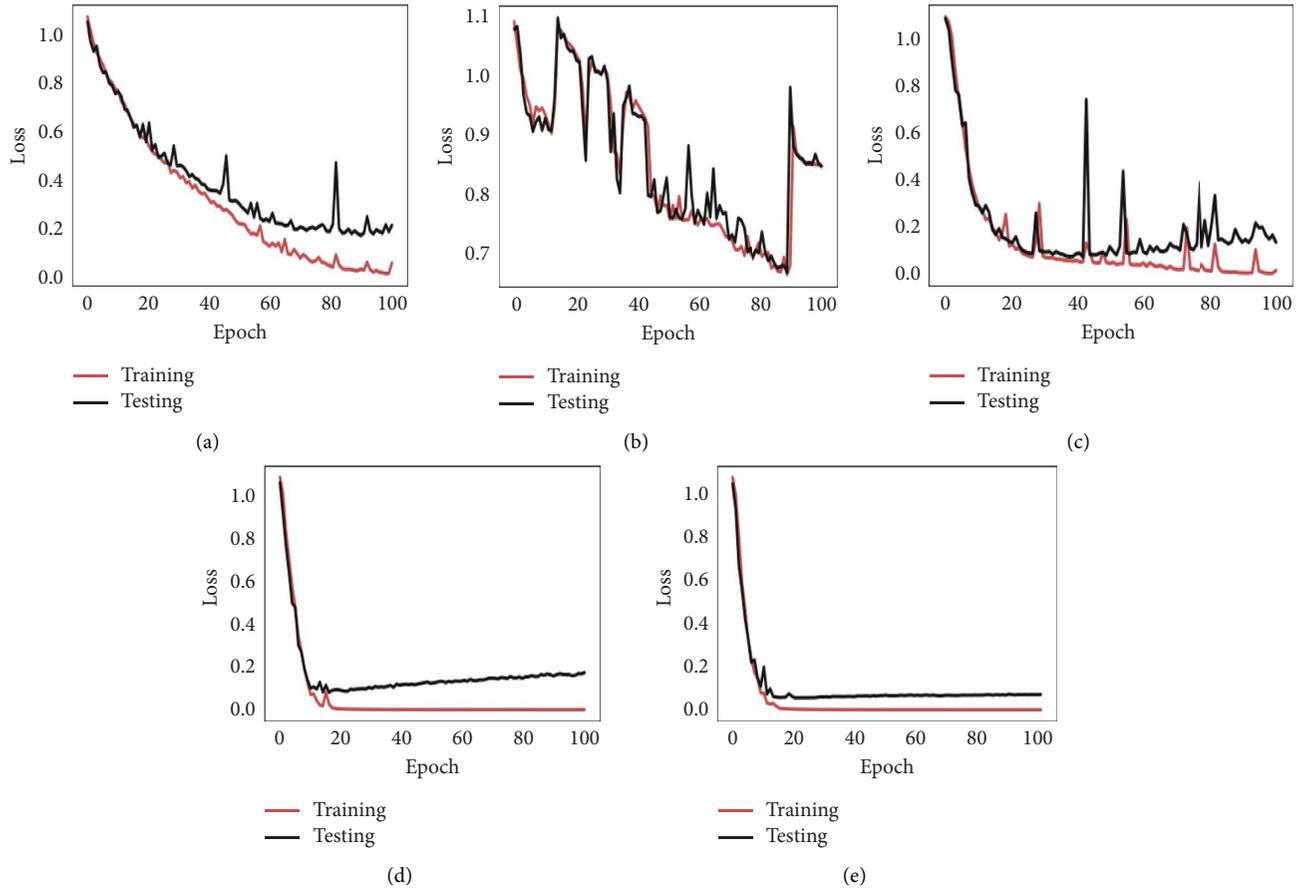


FIGURE 6: Loss vs. epoch curve for abnormal knee subjects.

TABLE 5: Comparison of the proposed method's performance with studies using same dataset (in %).

Approach	Subject	Accuracy		
		Walking	Sitting	Standing
EMD [41]	Healthy	64	67	69
MEMD [41]	Healthy	73	79	82
NA-EMD [41]	Healthy	79	83	83
MP-ANN [42]	Knee deformity	88	94	92
LRCN [43]	Healthy	98.2	97.7	98.4
	Knee deformity	92.8	92.3	92.2
ICA-EBM [36]	Healthy	96.0	96.2	96.2
	Knee deformity	86.6	86.4	85.5
WD-EEMD [34]	Healthy	85.11	88.70	93.50
	Knee deformity	98.86	96.38	96.77
V-1D-CNN [37]	Healthy	96.86	99.57	99.98
	Knee deformity	99.06	96.04	97.24
CNN-GRU (proposed)	Healthy	99.09	100	100
	Knee deformity	99.60	98.30	97.95

5. Conclusion and Future Scope

The research proposes the use of hybrid deep learning models CNN-LSTM and CNN-GRU to analyze sEMG data to detect lower limb activity in individuals with and without knee abnormalities in which CNN is used for temporal learning, and LSTM or GRU is used for sequence learning. To begin, the authors have used discrete wavelet denoising to denoise the original sEMG signal and then introduced overlapping windowing approaches for data segmentation to mitigate the issue of a small dataset. After that, deep learning models CNN, LSTM, GRU, CNN-LSTM, and CNN-GRU are implemented. In the proposed hybrid models, CNN is used for temporal learning, and LSTM or GRU is used for sequence learning. The proposed hybrid CNN-GRU model achieves classification accuracy of 99.86 and 98.69% and computational time of an epoch of 4.26 and 5.58 s for healthy and abnormal knee subjects, respectively. The results were compared with those obtained by individual models with the hybrid approach proving superior to them.

The sEMG dataset used in this study is limited to 22 individuals to evaluate the suggested approach. Thus, the strategy could be validated in the future by utilizing a large real-time dataset. The suggested methodology was evaluated

using an offline dataset; hence, future research could focus on clinical validation using a real-time dataset.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] S. Ranasinghe, F. Al Machot, and H. C. Mayr, "A review on applications of activity recognition systems with regard to performance and evaluation," *International Journal of Distributed Sensor Networks*, vol. 12, no. 8, Article ID 155014771666552, 2016.
- [2] V. B. Semwal, N. Gaud, P. Lalwani, V. Bijalwan, and A. K. Alok, "Pattern identification of different human joints for different human walking styles using inertial measurement unit (IMU) sensor," *Artificial Intelligence Review*, vol. 55, no. 2, pp. 1149–1169, 2022.
- [3] X. Liu, L. Liu, S. J. Simske, and J. Liu, "Human daily activity recognition for healthcare using wearable and visual sensing data," in *Proceedings of the IEEE International Conference on Healthcare Informatics (ICHI)*, pp. 24–31, IEEE, Chicago, IL, USA, 2016, October.
- [4] V. B. Semwal, A. Gupta, and P. Lalwani, "An optimized hybrid deep learning model using ensemble learning approach for human walking activities recognition," *The Journal of Supercomputing*, vol. 77, no. 11, Article ID 12256, 2021.
- [5] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE communications surveys & tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.
- [6] A. Vijayvargiya, B. Singh, R. Kumar, and J. M. R. S. Tavares, "Human lower limb activity recognition techniques, databases, challenges and its applications using sEMG signal: an overview," *Biomedical Engineering Letters*, vol. 12, no. 4, pp. 343–358, 2022.
- [7] B. S. Yang and S. T. Liao, "Fall detecting using inertial and electromyographic sensors," in *Proceedings of the 36th Annual Meeting of the American Society of Biomechanics*, pp. 15–18, Gainesville, FL, USA, 2012, August.
- [8] J. Cheng, X. Chen, and M. Shen, "A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals," *IEEE journal of biomedical and health informatics*, vol. 17, no. 1, pp. 38–45, 2012.
- [9] D. Farina and F. Negro, "Accessing the neural drive to muscle and translation to neurorehabilitation technologies," *IEEE Reviews in biomedical engineering*, vol. 5, pp. 3–14, 2012.
- [10] N. Nazmi, M. A. Abdul Rahman, S. I. Yamamoto, S. A. Ahmad, H. Zamzuri, and S. A. Mazlan, "A review of classification techniques of EMG signals during isotonic and isometric contractions," *Sensors*, vol. 16, no. 8, p. 1304, 2016.
- [11] S. K. Au, P. Bonato, and H. Herr, "An EMG-position controlled system for an active ankle-foot prosthesis: an initial experimental study," in *Proceedings of the 9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005*, pp. 375–379, IEEE, Chicago, IL, USA, 2005, June.
- [12] A. Vijayvargiya, P. L. Singh, S. M. Verma, R. Kumar, and S. Bansal, "Performance comparison analysis of different classifier for early detection of knee osteoarthritis," in *Sensors for Health Monitoring* Academic Press, Massachusetts, MA, USA, 2019.
- [13] L. Burkow-Heikkinen, "Non-invasive physiological monitoring of exercise and fitness," *Neurological Research*, vol. 33, no. 1, pp. 3–17, 2011.
- [14] K. Kiguchi, T. Tanaka, and T. Fukuda, "Neuro-fuzzy control of a robotic exoskeleton with EMG signals," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 481–490, 2004.
- [15] V. Krasin, V. Gandhi, Z. Yang, and M. Karamanoglu, "EMG based elbow joint powered exoskeleton for biceps brachii strength augmentation," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6, IEEE, Killarney, Ireland, 2015, July.
- [16] K. Sharmila, T. V. Sarath, and K. I. Ramachandran, "EMG controlled low cost prosthetic arm," in *Proceedings of the IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, pp. 169–172, IEEE, Mangalore, India, 2016, August.
- [17] S. Pancholi and A. M. Joshi, "Electromyography-based hand gesture recognition system for upper limb amputees," *IEEE Sensors Letters*, vol. 3, no. 3, pp. 1–4, 2019.
- [18] A. Vijayvargiya, P. Singh, R. Kumar, and N. Dey, "Hardware implementation for lower limb surface EMG measurement and analysis using explainable AI for activity recognition," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–9, 2022.
- [19] S. Cai, Y. Chen, S. Huang et al., "SVM-based classification of sEMG signals for upper-limb self-rehabilitation training," *Frontiers in Neurobotics*, vol. 13, p. 31, 2019.
- [20] C. Souit, D. S. Coelho, M. Szylio, F. Camargo-Junior, M. P. C. Junior, and A. Forner-Cordero, "Design of a lower limb exoskeleton for experimental research on gait control," in *Proceedings of the 6th IEEE International Conference on Biomedical Robotics and Biomechanics (BioRob)*, pp. 1098–1103, IEEE, Singapore, 2016, June.
- [21] P. K. Shukla, A. Vijayvargiya, and R. Kumar, "Human activity recognition using accelerometer and gyroscope data from smartphones," in *Proceedings of the International Conference on Emerging Trends in Communication, Control and Computing (ICONC3)*, pp. 1–6, IEEE, Lakshmanagarh, India, 2020, February.
- [22] U. S. L. G. Da Silva, H. A. Villagra, L. L. Oliva, and N. F. Marconi, "EMG activity of upper limb on spinal cord injury individuals during whole-body vibration," *Physiology International (Acta Physiologica Hungarica)*, vol. 103, no. 3, pp. 361–367, 2016.
- [23] A. Vijayvargiya, C. Prakash, R. Kumar, S. Bansal, and J. M. Rs Tavares, "Human knee abnormality detection from imbalanced sEMG data," *Biomedical Signal Processing and Control*, vol. 66, Article ID 102406, 2021.
- [24] Ö. F. Ertuğrul, Y. Kaya, and R. Tekin, "A novel approach for SEMG signal classification with adaptive local binary patterns," *Medical, & Biological Engineering & Computing*, vol. 54, no. 7, pp. 1137–1146, 2016.
- [25] V. P. Gurupur, S. A. Kulkarni, X. Liu, U. Desai, and A. Nasir, "Analysing the power of deep learning techniques over the traditional methods using medicare utilisation and provider data," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 31, no. 1, pp. 99–115, 2019.
- [26] N. Dua, S. N. Singh, V. B. Semwal, and S. K. Challa, "Inception inspired CNN-GRU hybrid network for human activity recognition," *Multimedia Tools and Applications*, pp. 1–35, 2022.

- [27] R. Jain, V. B. Semwal, and P. Kaushik, "Deep ensemble learning approach for lower extremity activities recognition using wearable sensors," *Expert Systems*, vol. 39, no. 6, Article ID e12743, 2022.
- [28] N. Dua, S. N. Singh, and V. B. Semwal, "Multi-input CNN-GRU based human activity recognition using wearable sensors," *Computing*, vol. 103, no. 7, pp. 1461–1478, 2021.
- [29] S. K. Challa, A. Kumar, and V. B. Semwal, "A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data," *The Visual Computer*, pp. 1–15, 2021.
- [30] A. Vijayvargiya, B. Singh, N. Kumari, and R. Kumar, "sEMG-based deep learning framework for the automatic detection of knee abnormality," *Signal, Image and Video Processing*, pp. 1–9, 2022.
- [31] O. F. A. Sanchez, J. L. R. Sotelo, M. H. Gonzales, and G. A. M. Hernandez, "EMG dataset in lower limb data set," *UCI machine learning repository*, vol. 2, 2014.
- [32] R. H. Chowdhury, M. B. Reaz, M. A. B. M. Ali, A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors*, vol. 13, no. 9, pp. 12431–12466, 2013.
- [33] U. Desai, R. J. Martis, C. G. Nayak, K. Sarika, and G. Seshikala, "Machine intelligent diagnosis of ECG for arrhythmia classification using DWT, ICA and SVM techniques," in *Proceedings of the Annual IEEE India Conference (INDICON)*, pp. 1–4, IEEE, New Delhi, India, 2015, December.
- [34] A. Vijayvargiya, V. Gupta, R. Kumar, N. Dey, and J. M. R. S. Tavares, "A hybrid WD-EEMD sEMG feature extraction technique for lower limb activity recognition," *IEEE Sensors Journal*, vol. 21, no. 18, Article ID 20431, 2021.
- [35] U. Desai, C. G. Nayak, and R. J. Seshikala, "Automated diagnosis of Coronary Artery Disease using pattern recognition approach," in *Proceedings of the 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 434–437, IEEE, Jeju, Korea (South), 2017 July.
- [36] G. R. Naik, S. E. Selvan, S. P. Arjunan et al., "An ICA-EBM-based sEMG classifier for recognizing lower limb movements in individuals with and without knee pathology," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 3, pp. 675–686, 2018.
- [37] A. Vijayvargiya, R. Kumar, R. Kumar, and N. Dey, "Voting-based 1D CNN model for human lower limb activity recognition using sEMG signal," *Physical and Engineering Sciences in Medicine*, vol. 44, no. 4, pp. 1297–1309, 2021.
- [38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [39] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.
- [40] A. Vijayvargiya, R. Kumar, N. Dey, and J. M. R. Tavares, "Comparative analysis of machine learning techniques for the classification of knee abnormality," in *Proceedings of the IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, pp. 1–6, IEEE, Greater Noida, India, 2020 October.
- [41] Y. Zhang, P. Xu, P. Li et al., "Noise-assisted multivariate empirical mode decomposition for multichannel EMG signals," *BioMedical Engineering Online*, vol. 16, no. 1, p. 107, 2017.
- [42] M. Herrera-González, G. A. Martínez-Hernández, J. L. Rodríguez-Sotelo, and O. Avilés-Sánchez, "Knee functional state classification using surface electromyographic and goniometric signals by means of artificial neural networks," *Ingeniería y Universidad*, vol. 19, no. 1, pp. 51–66, 2015.
- [43] A. Gautam, M. Panwar, D. Biswas, and A. Acharyya, "Myonet: a transfer-learning-based IRCN for lower limb movement recognition and knee joint angle prediction for remote monitoring of rehabilitation progress from sEMG," *IEEE J Transl Eng Health Med*, vol. 8, pp. 1–10, 2020.