



Research article

EMG gesture signal analysis towards diagnosis of upper limb using dual-pathway convolutional neural network

Hafiz Ghulam Murtza Qamar¹, Muhammad Farrukh Qureshi², Zohaib Mushtaq³, Zubariah Zubariah⁴, Muhammad Zia ur Rehman^{5,6}, Nagwan Abdel Samee⁷, Noha F. Mahmoud^{8,*}, Yeong Hyeon Gu^{9,*} and Mohammed A. Al-masni^{9,*}

¹ School of Electrical Engineering, Yanshan University, Qinhuangdao, Hebei 066104, China

² Department of Electrical Engineering, Riphah International University, Islamabad 44000, Pakistan

³ Department of Electrical, Electronics and Computer Systems, College of Engineering and Technology, University of Sargodha, Sargodha 40100, Pakistan

⁴ Department of Physiotherapy, Isfandyar Bukhari Civil Hospital, District Headquarter Hospital, Attock 43600, Pakistan

⁵ Department of Biomedical Engineering, Riphah International University, Islamabad 44000, Pakistan

⁶ Department of Health Science and Technology, Aalborg University, Aalborg 9220, Denmark

⁷ Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

⁸ Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia

⁹ Department of Artificial Intelligence Data Science, College of Software & Convergence Technology, Sejong University, Seoul 05006, Republic of Korea

* **Correspondence:** Email: nfmahmoud@pnu.edu.sa, yhgu@sejong.ac.kr, m.almasani@sejong.ac.kr.

Abstract: This research introduces a novel dual-pathway convolutional neural network (DP-CNN) architecture tailored for robust performance in Log-Mel spectrogram image analysis derived from raw multichannel electromyography signals. The primary objective is to assess the effectiveness of the proposed DP-CNN architecture across three datasets (NinaPro DB1, DB2, and DB3), encompassing both able-bodied and amputee subjects. Performance metrics, including accuracy, precision, recall, and F1-score, are employed for comprehensive evaluation. The DP-CNN demonstrates notable mean accuracies of $94.93 \pm 1.71\%$ and $94.00 \pm 3.65\%$ on NinaPro DB1 and DB2 for healthy subjects, respectively. Additionally, it achieves a robust mean classification accuracy of $85.36 \pm 0.82\%$ on amputee subjects in DB3, affirming its efficacy. Comparative analysis with previous methodologies on the same datasets reveals substantial improvements of 28.33%, 26.92%, and 39.09% over the baseline

for DB1, DB2, and DB3, respectively. The DP-CNN's superior performance extends to comparisons with transfer learning models for image classification, reaffirming its efficacy. Across diverse datasets involving both able-bodied and amputee subjects, the DP-CNN exhibits enhanced capabilities, holding promise for advancing myoelectric control.

Keywords: electromyography; nina pro; convolutional neural network; log mel spectrogram; upper limb; gesture recognition

1. Introduction

Classifying upper limb gestures using multichannel surface electromyography (sEMG) poses a formidable challenge with implications for both diagnostic and therapeutic applications [1, 2]. The inherent non-linear and stochastic nature of sEMG signals introduces a significant hurdle, complicating the precise categorization of upper limb gestures [3, 4]. The complex interplay among various muscles, coupled with their complex signaling patterns, contributes to the complexity of extracting meaningful information from sEMG recordings [5, 6]. This challenge is particularly pronounced in the context of employing sEMG signals for electromechanical hand prostheses, where precision and reliability are paramount [7, 8]. The variability inherent in sEMG signals, influenced by factors such as muscle fatigue, electrode placement, and individual anatomical differences, adds layers of intricacy to the task of achieving accurate and robust gesture classification [9, 10]. Effective classification methods for sEMG in diverse applications require innovative solutions to address the challenges posed by unique characteristics of upper limb gesture signals.

Machine learning and deep learning techniques have emerged as transformative tools in the domains of information processing, providing unprecedented capabilities for various tasks such as pattern recognition [11, 12], feature extraction [13, 14], and classification [15–17]. In the context of sEMG for gesture classification, numerous studies have explored the applicability of machine learning and deep learning techniques [18, 19]. For instance, Saeed et al. [20] applied machine learning techniques to raw signals from the DB1 dataset, achieving accuracies of 85.41% and 91.14% using an artificial neural network (ANN) and linear discriminant analysis (LDA), respectively. Karnam et al. [21] achieved a classification accuracy of 88.8% on DB1 using K-nearest neighbours (KNN). Akmal et al. [22] explored training strategies for artificial neural networks (ANN), which emerged as a pivotal aspect in sEMG signal classification. This study analyzes twelve different training strategies, evaluating their performance on multiday EMG data. The results highlight the resilience of backpropagation and scaled conjugate gradient methods, providing valuable insights into optimal training approaches for efficient prosthetic control. SVM-based classification for prosthetic finger movements is investigated for real-time implementation [23]. Leveraging the stability and efficiency of SVM on a Raspberry Pi, this study achieved 78% classification accuracy. Inam et al. [24] explored gender-specific considerations in sEMG for upper-limb prosthetics. Evaluating EMG differences between males and females, this research employs an ANN for classification. While overall similarities are observed, certain features exhibit gender-specific variations, shedding light on the importance of tailored approaches for diverse user populations. In the domain of deep learning, Hu et al. [25] implemented a recurrent neural network (RNN) and a convolutional neural network

(CNN) on sEMG signals from NinaPro DB1, attaining an accuracy of 87%. Pancholi et al. [26] applied a CNN model to various NinaPro datasets, achieving classification accuracies ranging from 81.67% to 99.11%. Cheng et al. [27] utilized the NinaPro DB1 dataset, applying CNN to sEMG-feature-images extracted from raw signals and achieving a classification accuracy of 82.54%. Additionally, Tong et al. [28] applied a CNN and long-short term memory (LSTM) based hybrid classifier on the NinaPro DB1 dataset, yielding an accuracy of 78.31%. [29] explored a concatenate feature fusion recurrent convolutional neural network (CFF-RCNN) to address this by introducing a concatenate feature fusion (CFF) strategy, achieving notable accuracies. CFF-RCNN surpasses reported results, achieving 88.87% on DB1, 99.51% on DB2, and 99.29% on DB4 with over 50 gestures. Qureshi et al. [30] introduced the efficient concatenated convolutional neural network (E2CNN) as a robust solution for real-time sEMG classification. By converting raw sEMG signals into Log-Mel spectrograms (LMS) and employing concatenation layers, E2CNN achieves high accuracy and response times for both non disabled and amputee subjects, positioning it as a potential candidate for prosthetic control in real-world scenarios. Another study [31] also explored improving myoelectric control in wearable prostheses using CNNs. By comparing multiday sEMG recordings, the proposed CNN exhibits superior accuracy for able-bodied and amputee subjects in within-day and between-day analyses. This research underscores the CNN's efficacy and computational efficiency, presenting a promising avenue for enhancing prosthetic hand control.

In the proposed study, we are introducing a dual-pathway convolutional neural network (DP-CNN) to classify sEMG signals from both healthy and amputee subjects. The novelty of this architecture resides in its ability to process Log-Mel spectrogram (LMS) images derived from raw multichannel EMG signals. Utilizing spectrograms instead of raw EMG signals has demonstrated enhanced performance across various studies in the field [26, 30–32]. This trend is evident in multiple investigations where LMS significantly improved the efficiency of the classification system. LMS effectively captured the time-frequency characteristics of the EMG signals [31, 33], and they are commonly used in various applications, including driver fatigue detection and EMG classification [31]. For instance, in a study focused on driver fatigue detection, a model based on LMS and a convolution recurrent neural network (CRNN) was proposed and demonstrated high accuracy in distinguishing between alert and fatigued states [33]. LMS also brings superior performance to the deep learning models. This is evidenced in a study in which a CNN achieved an impressive classification accuracy of over 90% on healthy and amputee subjects when applied to LMS-based images [30]. Furthermore, LMS images derived from EMG signals have been employed successfully in hybrid deep-learning methods to classify four different EMG-signal patterns, achieving significant classification results [34]. Moreover, LMS also offers a data augmentation method, which has been shown to significantly enhance the accuracy of deep learning models in classifying EMG signals [32].

The rationale behind this preference for spectrograms lies in their ability to capture intricate time-frequency characteristics inherent in EMG signals. Spectrograms, particularly LMS, provide a more comprehensive representation of the signal, highlighting nuanced patterns that might be obscured in raw EMG data. This richer representation aids in extracting deep information from the signals, contributing to increased effectiveness in classification tasks. EMG signals exhibit complex and dynamic frequency patterns that convey information about muscle contractions. The Mel-frequency bands in LMS provide an effective means of representing these intricate patterns, offering a more compact and discriminative representation of the signal. The logarithmic

transformation in LMS compresses the higher frequencies, emphasizing the lower-frequency components. This is beneficial for EMG analysis, as the lower frequencies often contain valuable information related to muscle activities and gestures.

In alignment with these findings, the current study also adopts the use of spectrograms, specifically LMS, as the primary input for the proposed DP-CNN. The choice is grounded in the well-substantiated efficacy of spectrograms, particularly LMS, as demonstrated in previous literature. Spectrograms not only effectively capture time-frequency characteristics, but also offer superior performance to deep learning models. The LMS technique involves transforming raw EMG signals into a visual representation that emphasizes the spectral features relevant for gesture classification. By utilizing Mel-frequency bands, which are perceptually spaced to mimic human hearing, and applying a logarithmic scale, the LMS efficiently captures the frequency patterns within EMG signals.

As such, the adoption of LMS as the primary input in our proposed DP-CNN is grounded in evidence of its efficacy and superior performance in previous studies. By implementing this methodology, we aim to validate the performance of DP-CNN on LMS images extracted from sEMG signals, thereby offering a robust, reliable, and real-time solution for prosthetic control applications. The proposed DP-CNN architecture has been implemented on surface EMG (sEMG) of healthy and amputee subjects taken from NinaPro DB1, DB2, and DB3, respectively. The proposed DP-CNN is based on the CNN, and input is provided as a spectrogram images converted from raw EMG signals. The LMS images are obtained from the raw multichannel EMG signals available in NinaPro Databases 1 (DB1) and 3 (DB3). The contribution of this work is validating the performance of the DP-CNN implemented on LMS images extracted from bio-signals, in this case, EMG signals.

The main contributions of this paper are:

- Preprocessing raw EMG signals into Log-Mel spectrogram images, enhancing feature extraction.
- Development of a dual-pathway convolutional neural network (DP-CNN) combining convolutional and dense pathways for robust EMG signal classification.
- Extensive assessment of the proposed DP-CNN's effectiveness and generalizability across diverse datasets, including both healthy and amputee subjects.
- Thorough comparison of the DP-CNN's performance against prior works on the same dataset, providing insights into advancements.
- Benchmarking against pre-trained transfer learning models (AlexNet, MobileNet, VGG19, DenseNet121, ResNet50), showcasing the uniqueness and efficacy of the proposed approach.

The paper is organized as follows: in Section 2, we provide a detailed description of the methodology used in this study. We first discuss the datasets used and the pre-processing steps undertaken to prepare the data for analysis. We then provide the details of the proposed deep neural network (DNN) used for classification. Section 3 illustrates the results obtained from the proposed methodology and we discuss the implications of our findings and their potential applications. Section 4 concludes the paper.

2. Materials and methods

2.1. Experimental datasets and setup description

In this study, we are using three datasets from a publicly available database: NinaPro Database 1 (DB1) [35], Database 2 (DB2) [36], and Database 3 (DB3) [37] for investigation and validation of the proposed methodology. The experiments were approved by the Ethics Commission of the state of Valais (Switzerland), the main place for data acquisition [35]. The details of each dataset are given below:

- 1) The first NinaPro database contains 10 repetitions of 52 different movements carried out by 27 intact subjects and serves as a standard dataset for the classification of myoelectric motions. The DB1 dataset includes three exercises categorized into (A) basic movements of the fingers, (B) isometric, isotonic hand configurations and basic wrist movements, and (C) grasping and functional movements. Ten Otto Bock MyoBock 13E200 electrodes are used to collect sEMG data; a Cyberglove 2 data glove is used to collect kinematic data. Each subject and exercise has a corresponding MATLAB file in the database that contains information about the subject, the exercise, the electrodes' sEMG signal, the 22 cyberglove sensors' uncalibrated signal, the subject's repeated movement, and the stimulus' repeated occurrence.
- 2) The DB2 dataset is composed of three types of exercises that are categorized into three groups: (A) basic movements of fingers and wrist, (B) grasping and functional movements, and (C) force patterns. To collect kinematic data, a dataglove (Cyberglove 2) and an accelerometer on the wrist were used, while a Delsys Trigno Wireless EMG system were utilized with 12 active double-differential wireless electrodes to record muscular activity. The sampling rate for sEMG signals is 2 kHz. Each exercise and subject have a synchronized MATLAB file that contains various variables, such as subject and exercise number, sEMG signal, kinematic information, inclinometer signal, movement repeated by the subject, and force recorded during the third exercise. Additionally, force sensor calibration values for the least and highest force are included in each file.
- 3) The third NinaPro database is a comprehensive resource for the development and evaluation of naturally-controlled non-invasive robotic hand prostheses. The experiment consists of the same three exercises as DB2: basic finger and wrist motions, grasping and functioning movements, and force patterns. The dataset includes 49 different movements (including rest) performed by 10 amputee participants with each movement repeated 6 times, and the movements were chosen from the hand taxonomy as well as literature on hand robotics. A Delsys Trigno Wireless EMG system with 12 active double-differential wireless electrodes was used to collect the muscular activity. The database provides one MATLAB file with synchronized variables for each exercise and subject, including subject and exercise number, sEMG signal, kinematic information, inclinometer signal, movement repeated by the subject, and force recorded during the third exercise. The collection also contains force sensor calibration information for the minimum and maximum force.

The number of repetitions of each gesture is ten in DB1, and six in DB2 and DB3. For this study, we randomly selected ten subjects from DB1 and DB2, five male and five female each; and ten from DB3 for validation of the proposed technique. We utilized the common movements from the three

databases; exercise C of DB1 and exercise B of DB2 and DB3 are the only common gestures among the datasets. The exercise is composed of 23 hand gestures and is illustrated in Figure 1.



Figure 1. Gestures utilized in this study [35].

2.2. Preprocessing technique

The data in this study was recorded using different numbers of channels in the three databases: 10 channels in DB1, and 12 channels in DB2 and DB3. The sampling rate for DB1 was 100 Hz, while for the DB2 and DB3 datasets, it was 2 kHz. The DB1 data was already shielded from power line noise; however, DB2 and DB3 are not shielded from power line noise [35]. Therefore, a 50 Hz second-order Butterworth notch filter was used for DB2 and DB3. Furthermore, DB1 data was filtered at 1 Hz using a second-order Butterworth filter [38]. In DB1, the signals available were root mean squared (RMS) values of raw signals, while in DB2 and DB3, raw EMG signals were available. To ensure optimal processing for EMG-based prosthetics, it is recommended to use signal segments with duration ranges from 150 ms to 250 ms [39, 40]. Therefore, each raw EMG signal is

divided into smaller segments with a duration of 200 ms with an overlapping increment of 50 ms. Since the number of repetitions are different in each dataset, we obtained different numbers of segmented signals. Specifically, we extracted 5750×10 segmented signals from DB1, while we extracted $14,950 \times 12$ segmented signals from DB2 and DB3 for each subject. Here, the 10 and 12 represent the number of channels in each signal.

After segmentation, each signal was converted into a Log-Mel spectrogram (LMS) image using the librosa library in Python. The LMS is a representation of the spectral content of a signal on a logarithmic frequency scale. By using LMS, we were able to analyze the frequency content of the segmented signals, which can provide more information for classification.

Let us define a segmented signal $s_w(t)$ with length L and sampling frequency f_{s_w} in hertz. Its short-term fourier transform (STFT) S_w is then given by

$$S_w(\bar{x}, \bar{y}) = \sum_{t=0}^{N-1} s_w(t + xH) \cdot w(t) \cdot e^{-i2\pi y \frac{t}{\tau}}. \quad (2.1)$$

Here, $H \in \mathbb{N}$ represents the hop length, $w : [0 : \tau - 1] \in \mathbb{R}$ is the Hann window defined as $w = 0.5 - 0.5 \cos(\frac{2\pi t}{\tau-1})$, where $\tau \in \mathbb{N}$ is the length of w , $\bar{x} \in [0 : \frac{L-\tau}{H}]$ denotes the time index, and $\bar{y} \in [0 : \frac{N}{2}]$ denotes the frequency index.

The short-term fourier transform spectrogram of S_w can be obtained as

$$S_{STFT}(\tilde{x}, \tilde{y}) = |S_w(\bar{x}, \bar{y})|^2. \quad (2.2)$$

The Mel spectrum and linear frequency are related by $f_{mel} = 2959 \times \log_{10}(1 + \frac{f}{700})$. We can estimate the LMS using

$$S_{LM}(\hat{x}, \hat{y}) = \sum_{f(y)=f_c(x-1)}^{f_c(x+1)} \log_{10}(M_{FB}(\tilde{x}, \tilde{y}) \cdot S_{STFT}(\tilde{x}, \tilde{y})). \quad (2.3)$$

Here, $M_{FB}(\tilde{x}, \tilde{y})$ is the Mel filter bank and can be estimated from

$$M_{FB}(\tilde{x}, \tilde{y}) = \begin{cases} \frac{f(\tilde{y}) - f_c(\tilde{x}-1)}{f_c(\tilde{x}) - f_c(\tilde{x}-1)} & \text{for } f_c(\tilde{x}-1) \leq f(\tilde{y}) < f_c(\tilde{x}) \\ \frac{f(\tilde{y}) - f_c(\tilde{x}+1)}{f_c(\tilde{x}) - f_c(\tilde{x}+1)} & \text{for } f_c(\tilde{x}) \leq f(\tilde{y}) < f_c(\tilde{x}+1) \\ 0 & \text{others.} \end{cases} \quad (2.4)$$

Here, $f(\tilde{y})$ denotes the linear frequency, and $f_c(\tilde{x}) = \tilde{x} \cdot \delta f_{mel}$ represents the center frequencies on the Mel-scale.

For each windowed signal in DB1, we convert it into an LMS individually. Then, this process is iteratively repeated for all ten channels, providing us with ten LMS images. We then combine these ten LMS vertically to form an input image, as shown in Figure 2(a). This process yields 5750 EMG images as LMS, which serve as the input dataset to the DP-CNN model for each subject in DB1. Similar to DB1, we apply the same technique to each windowed signal in DB2 and DB3, resulting in twelve LMS for each signal. These twelve LMS are combined vertically to form an image, as illustrated in Figure 2(b),(c). This process resulted in 14,950 EMG images as LMS for each subject in DB2 and DB3.

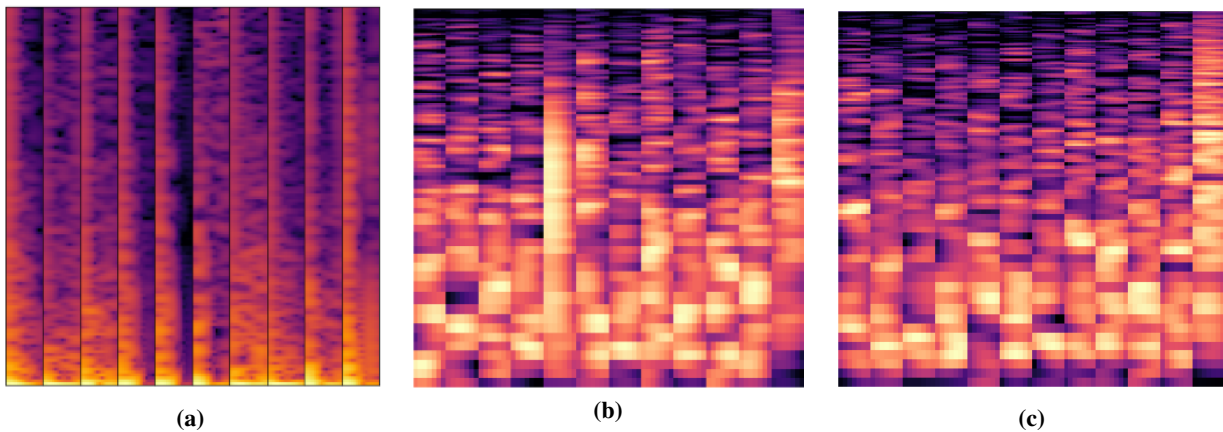


Figure 2. The input Log-Mel spectrogram (LMS) images converted from raw EMG signal: (a) LMS converted from DB1, (b) LMS converted from DB2, (c) LMS converted from DB3.

2.3. Dual pathway convolutional neural network architecture

We propose a dual-pathway convolutional neural network (DP-CNN) with batch normalization and max-pooling functions along with dropout layers for classification of electromyogram (EMG) signals. The proposed model is designed to operate on LMS as input, with a fixed feature size of 224×224 . The DP-CNN is comprised of two pathways, namely the traditional convolutional pathway and a dense pathway, which are combined using a concatenation layer. The input to the DP-CNN is denoted as $S_{LM}(\hat{x}, \hat{y})$, and the convolution operation can be explained from [41] and is given as follows:

$$\begin{aligned} O &= F(S_{LM}(\hat{x}, \hat{y})|a) \\ &= f_n(f_3(f_2(S_{LM}\hat{x}, \hat{y})|\theta_2)|\theta_3)\theta_N). \end{aligned} \quad (2.5)$$

Here, f_n represents the n -th layer of the DP-CNN, and N is the total number of layers used, which in this study is set to 8. The parameter for the n -th layer is denoted as $\theta_1 = [X, b]$. We can express the convolutional layer operations in the following way:

$$\begin{aligned} O_1 &= f_1(S_{LM}(\hat{x}, \hat{y})S_{LM}(\hat{x}, \hat{y})_n|\theta_N) \\ &= h(S_{LM}(\hat{x}, \hat{y})_l + b * X). \end{aligned} \quad (2.6)$$

Here, I_n represents the input of the n th-layer, X is the corresponding filter, $*$ denotes the valid convolution operation, $h(\cdot)$ denotes the pointwise activation function, and b denotes the vector bias term.

In the proposed DP-CNN, the convolutional layers of the first pathway are estimated using

$$C_{i,j} = \sum_{k=1}^n \sum_{l=1}^m W_{k,l} \cdot I_{i+k-1,j+l-1} + b, \quad (2.7)$$

where $C_{i,j}$ is the value of the feature map for the i^{th} row and j^{th} column, $W_{k,l}$ is the weight for the k^{th} row and l^{th} column of the filter, $I_{i+k-1,j+l-1}$ is the value for the corresponding position in the input image, b is the bias, and n and m are the dimensions of the filter.

The activation function, rectified linear unit (ReLU), can be represented as

$$\text{ReLU}(x) = \max(0, x). \quad (2.8)$$

The dense layer equation can be represented by

$$y = D \cdot \sum_{i=1}^n w_i \cdot x_i + b, \quad (2.9)$$

where y is the output of the dense layer, w_i is the weight for the i^{th} unit, x_i is the input for the i^{th} unit, n is the number of units in the dense layer, and the dropout layer D is a binary mask with a probability of 0.2 to set the corresponding value to 0.

The concatenation function can be represented by

$$O = [f_{CL}(f_{RL}(I)), f_{DL}(f_{FL}(f_{CL}(f_{RL}(I))))], \quad (2.10)$$

where O is the final output of the concatenation function, and $[\cdot, \cdot]$ represents the concatenation operation of two arrays.

Overall, the proposed DP-CNN can be represented by

$$O = [C_{1,1}, C_{1,2}, \dots, C_{n,m}, y_1, y_2, \dots, y_k], \quad (2.11)$$

where

$$C_{i,j} = \text{ReLU} \left(\sum_{k=1}^n \sum_{l=1}^m W_{k,l} \cdot I_{i+k-1, j+l-1} + b \right), \quad (2.12)$$

and

$$y_i = \text{ReLU} \left(\sum_{j=1}^m w_j \cdot x_j + b \right) \cdot D, \quad (2.13)$$

and x_j is the output of the previous dense layer.

The architecture of the DP-CNN model is shown in Figure 3. The primary advantage of the convolutional pathway is its ability to automatically learn and extract relevant features from input images without the need for manual feature engineering. This is accomplished through the use of convolutional and pooling layers, which learn local and global patterns in the input data. The model is able to learn increasingly complex representations of the input images by stacking multiple convolutional layers on top of each other.

On the other hand, the advantage of the dense pathway is its ability to detect global patterns and relationships in the input data. This is accomplished through the use of fully connected layers that learn to combine features from all parts of the input data. The model is able to learn increasingly complex representations of the input data by stacking multiple fully connected layers on top of each other. Another benefit of the dense pathway is its ability to handle input data of any size and shape, as long as it can be converted to a one-dimensional format. This makes the model suitable for a wide range of input data types, such as text or time series data, which may not have the same spatial structure as images. Furthermore, the use of dropout layers in the dense pathway helps to regularize the model and prevent overfitting. Dropout layers randomly set a fraction of the activations in the previous layer

to zero during training, which helps to reduce co-adaptation between neurons and forces the model to learn more robust features.

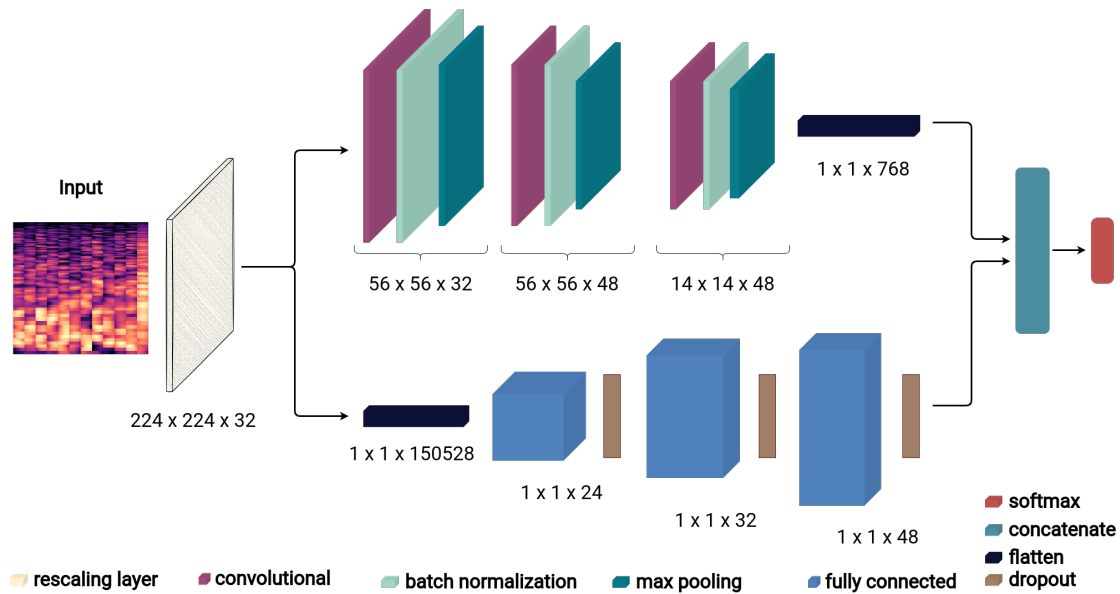


Figure 3. General architecture of the proposed dual-pathway convolutional neural network.

The DP-CNN model can capture different types of information from the input images because it has two separate pathways for processing the input data. The convolutional pathway can extract spatial features from images by using convolutional and pooling layers, whereas the dense pathway can capture global patterns and relationships in input data by using fully connected layers. The model can make more accurate predictions by combining the outputs of both pathways.

The use of multiple pathways allows for greater regularization and reduces the risk of overfitting. By having two distinct pathways that process the input data in different ways, the model is less likely to memorize the training data and more likely to generalize well to new, previously unseen data. Furthermore, the use of dropout layers in the dense pathway provides additional regularization and helps to prevent overfitting.

The DP-CNN model can capture different types of information from the input images by using two separate pathways for processing the input data. The convolutional pathway can extract spatial features from images by using convolutional and pooling layers, whereas the dense pathway can capture global patterns and relationships in the input data by using fully connected layers. By combining the outputs of both pathways, the model is able to make more accurate predictions by leveraging both types of information.

The use of multiple pathways improves regularization and reduces the risk of overfitting. The model is less likely to memorize the training data and more likely to generalize well to new, unseen data by having two separate pathways that process the input data in different ways. Furthermore, the use of dropout layers in the dense pathway provides additional regularization and aids in the prevention of overfitting.

For training the proposed DP-CNN model, a total of 50 epochs are set and a batch size of 32 is used.

The Adam optimizer is employed for optimization. The learning rate was set to 0.001. The input to the DP-CNN model is an image with a size of $224 \times 224 \times 3$. Prior to feeding the images to the DP-CNN model, they are resized using a rescaling layer (RL), which scales the pixel values in the range of 0 to 1. This is done to ensure that the input data is within a suitable range for the neural network model. The rescaled input images are then provided as input to the DP-CNN model for training.

Convolutional pathway: The input images processed by the rescaling layer are fed to the convolutional pathway. Details of each layer are given below:

- **Layer 1:** The first layer is composed of a convolutional layer of 32 filters with a filter size of 9×9 , with ReLU and L2 regularization of 0.001. The convolutional layer is succeeded by a batch normalization layer. After that, a maximum pooling layer is used with a pool size of 4×4 .
- **Layer 2:** The second layer is similar to the first layer and is composed of same batch normalization and max-pooling layer, except the the number of filters are 48 and a filter size of 5×5 is used.
- **Layer 3:** The third layer is made up of a convolutional layer of 48 filters with a filter size of 3×3 with ReLU activations. A batch normalization layer is introduced after the convolutional layer, and then a max-pooling layer is used with a pool size of 2×2 .

Dense pathway: The same input images from the rescaling layer are converted to one-dimensional data using a flatten layer and then are fed to the dense pathway:

- **Layers 1 & 2:** The first layer in the second pathway is a fully connected dense layer composed of 24 units, and ReLU is used as the activation function. The second layer is a dropout layer with a rate of 0.2.
- **Layers 3 & 4:** The third and fourth layers are similar to the first and second layer, except the fully connected dense layers have 32 units.
- **Layers 5 & 6:** The fifth and sixth layers are similar to the above two layers, except the fully connected dense layers have 48 units.

Concatenation and classification layer: The data from the first and second pathways are combined using a concatenation layer. This layer allows the outputs of both pathways within a neural network to be combined into a single output. The last layer is composed of 23 units.

2.4. Performance evaluation metrics

The performance of the proposed DP-CNN is evaluated on metrics based on multi-class classification. For this study, we are using mean accuracy, mean precision, mean recall, and mean F1-score along with their standard deviations [42].

Let C_{TP} be true positives and C_{TN} true negatives, while C_{FP} are false positives and C_{FN} false negatives. Multi-class classification accuracy can be estimated using

$$Accuracy = \frac{C_{TP} + C_{TN}}{C_{TP} + C_{TN} + C_{FP} + C_{FN}}. \quad (2.14)$$

For binary-class classification $n = 1$, precision $P_{n=1}^k$ and recall $R_{n=1}^k$ for a particular class k can be calculated as follows:

$$P_{n=1}^k = \frac{C_{TP}^k}{C_{TP}^k + C_{FP}^k}, \quad (2.15)$$

$$R_{n=1}^k = \frac{C_{TP}^k}{C_{TP}^k + C_{FN}^k}. \quad (2.16)$$

For multi-class classification $n = N$, precision P_N^k and recall R_N^k can be calculated as follows:

$$P_N^k = \frac{\sum_{k=1}^K P_{n=1}^k}{K}, \quad (2.17)$$

$$R_N^k = \frac{\sum_{k=1}^K R_{n=1}^k}{K}. \quad (2.18)$$

F1-score for multi-class classification is given as

$$F1 - score = 2 \times \left(\frac{P_N^k \times R_N^k}{(P_N^k)^{-1} + (R_N^k)^{-1}} \right). \quad (2.19)$$

3. Results and discussion

Table 1 illustrates the developmental setup employed in this study.

Table 1. System specifications.

Components	Specifications
Processor	Intel (R) Xeon (R) CPU E5-2630
RAM	32 GB
GPU	NVIDIA Tesla T4
	Ubuntu
	CUDA 11.8
	Python 3.11
	Keras 2.90
Software	TensorFlow 2.9.2
	Matplotlib 3.6.1
	Scikit-Learn 1.1.2
	NumPy 1.23.4
	Pandas 1.5.0
	Seaborn 0.12.0

3.1. Performance assessment of individual pathways & dual-pathway CNN

We have evaluated the performance of the individual pathways of the proposed dual-pathway CNN, and compared the results of each pathway with the DP-CNN. For this purpose, we have only tested the performance on a single subject from each dataset. For all three datasets, the dense pathway

and concatenation layer are removed and the performance is validated. Alternatively, in the second phase, the convolutional pathway along with the concatenation layer in DP-CNN are removed and the performance of only the dense path is validated. Finally, the performance of the complete DP-CNN is validated on a single subject from each dataset. Figure 4 illustrates the accuracies achieved by each pathway along with DP-CNN on one subject from all three datasets. It can be seen that neither pathway could achieve the desired accuracy compared to the complete DP-CNN.

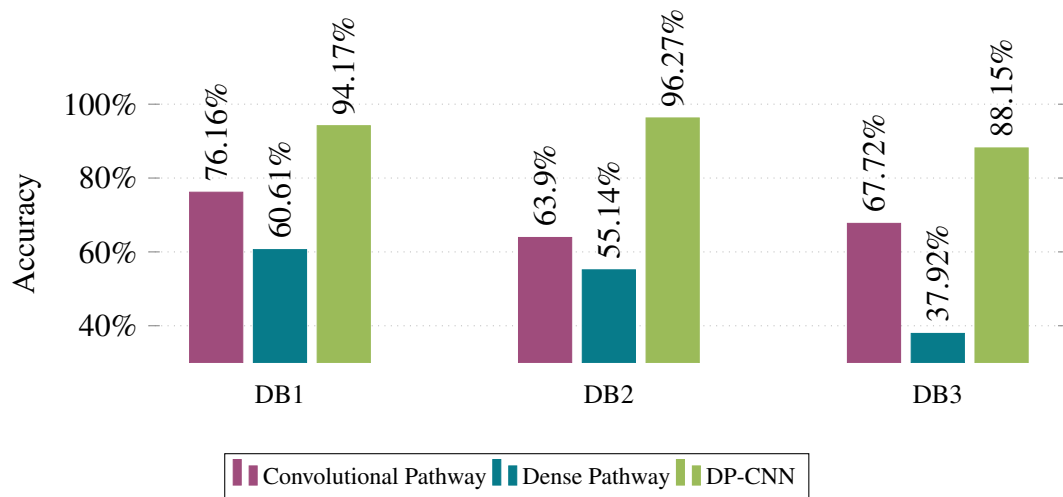


Figure 4. Performance of each individual pathway and the complete DP-CNN on a single subject from each database.

3.2. Performance assessment of the proposed method

We have evaluated the proposed DP-CNN on 30 subjects: 20 able-bodied subjects from DB1 and DB2, and 10 amputee subjects from DB3. To ensure diversity and balance, we randomly selected ten subjects from each of the two databases DB1 and DB2. In each database, we ensured that five males and five females were selected. Regarding DB3, we were able to select ten subjects out of the eleven available, but we could not include subject one due to the unavailability of the desired gestures in the dataset [35]. Additionally, subjects 7 and 8 had fewer electrodes, resulting in ten channels instead of twelve, but we still processed their data for our analysis. This approach allowed us to gather a representative sample that can help us draw accurate conclusions and insights from our research. The testing and training are done for each subject from both dataset. Each dataset is divided into a 70–30 split, where 70% is used for training and 30% for validation. This is done for each subject and the performance metrics are determined for each subject. Using the proposed DP-CNN, DB1, DB2, and DB3 are classified individually, and then the mean accuracy, mean precision, mean recall, and mean F1-score are determined from all subjects. Table 2 shows the accuracy, precision, recall, and F1-score for DB1 subjects. The proposed DP-CNN achieved a mean classification accuracy of $94.93 \pm 1.71\%$, mean precision of $94.93 \pm 1.71\%$, mean recall of $94.93 \pm 1.71\%$, and mean F1-score of $94.93 \pm 1.71\%$ on the subjects of the DB1 dataset. Similarly, the proposed DP-CNN achieved a mean classification accuracy of $94.00 \pm 3.56\%$, mean precision of $94.00 \pm 3.56\%$, mean recall of $94.00 \pm 3.56\%$, and mean F1-score of $94.00 \pm 3.56\%$ when applied to DB2 subjects. Table 3 illustrates the accuracy,

precision, recall, and F1-score for DB2 subjects. Similarly, the proposed DP-CNN has achieved a mean classification accuracy of $85.36 \pm 0.82\%$, mean precision of $85.35 \pm 0.86\%$, mean recall of $85.34 \pm 0.81\%$, and mean F1-score of $85.36 \pm 0.82\%$, when applied to DB3 subjects. Table 4 illustrates the accuracy, precision, recall, and F1-score for DB3 subjects.

Table 2. Performance of the proposed DP-CNN on DB1 subjects.

Subject	Accuracy	Precision	Recall	F1-score
1	94.17%	94.17%	94.16%	94.15%
2	94.24%	94.25%	94.25%	94.25%
3	95.46%	95.46%	95.46%	94.45%
4	95.13%	95.14%	95.13%	95.13%
5	94.57%	94.58%	94.56%	94.57%
6	95.23%	95.24%	95.24%	95.24%
7	96.75%	96.75%	96.75%	96.75%
8	94.76%	94.76%	94.77%	94.76%
9	93.24%	93.21%	93.21%	93.24%
10	95.97%	95.98%	96.00%	95.98%
Mean \pm SD	94.93 \pm 1.71%	94.93 \pm 1.71%	94.93 \pm 1.71%	94.93 \pm 1.71%

Table 3. Performance of the proposed DP-CNN on DB2 subjects.

Subject	Accuracy	Precision	Recall	F1-score
1	96.27%	96.27%	96.19%	96.25%
2	93.95%	93.95%	93.95%	93.95%
3	91.39%	91.39%	91.36%	91.32%
4	97.25%	97.25%	97.26%	97.95%
5	92.63%	92.65%	92.62%	92.63%
6	91.00%	91.01%	91.00%	90.29%
7	96.96%	96.99%	96.98%	96.98%
8	92.67%	92.67%	92.67%	92.67%
9	95.17%	95.17%	95.17%	95.17%
10	92.75%	92.72%	92.72%	92.72%
Mean \pm SD	94.00 \pm 3.56%	94.00 \pm 3.56%	94.00 \pm 3.56%	94.00 \pm 3.56%

Table 4. Performance of the proposed DP-CNN on DB3 subjects.

Subject	Accuracy	Precision	Recall	F1-score
1	88.15%	88.11%	88.12%	88.15%
2	82.69%	82.69%	82.69%	82.69%
3	87.22%	87.22%	87.23%	87.22%
4	84.81%	84.83%	84.81%	84.81%
5	86.91%	86.91%	86.91%	86.91%
6	82.39%	82.41%	82.40%	82.40%
7	87.01%	87.00%	87.02%	87.00%
8	86.15%	86.95%	86.95%	86.95%
9	81.95%	81.95%	81.95%	81.95%
10	86.95%	86.95%	86.95%	86.95%
Mean \pm SD	85.36 \pm 0.82%	85.35 \pm 0.86%	85.34 \pm 0.81%	85.36 \pm 0.82%

3.3. Comparison of results with previous studies

In this study, we evaluated the performance of DP-CNN on LMS extracted from raw EMG signals of the publicly available NinaPro datasets. To assess the effectiveness of our approach, we compared the results with previous studies that utilized deep learning-based techniques on the same datasets. Although, to best of our knowledge, only one experiment has been conducted on the NinaPro database using Mel or Log-Mel spectrograms, we have compared our results with other studies that have employed any deep learning-based techniques on this database.

The proposed DP-CNN model was compared to earlier studies on the three databases: DB1, DB2, and DB3. Table 5 reveals that the DP-CNN model attained an accuracy of 94.93% on DB1, exceeding earlier research that ranged from 66.60% to 91.27%. On DB2, the DP-CNN model achieved the similar accuracy as on DB1, 94.00%, as shown in Table 6. Previous investigations achieved accuracies ranging from 60.27% to 89.45%. Finally, on DB3 the DP-CNN model attained an accuracy of 85.36%, as shown in Table 7, which is higher than earlier research, which ranged from 46.27% to 81.67%. These findings show that the suggested DP-CNN model is successful for gesture classification across all three datasets, with higher accuracy than earlier studies.

Table 5. Comparison of the proposed DP-CNN with previous methods applied on the NinaPro DB1.

Reference	Year	Dataset	Classifier	Accuracy
[38]	2016	DB1	CNN	66.60%
[43]	2016	DB1	CNN	77.80%
[44]	2019	DB1	CNN	85.00%
[28]	2019	DB1	CNN-LSTM	71.20%
[45]	2019	DB1	Multi-view CNN	88.20%
[46]	2022	DB1	Deformable CNN	81.80%
[47]	2022	DB1	Dual-View CNN	86.29%
[30]	2023	DB1	E2CNN	91.27%
This Work	2023	DB1	DP-CNN	94.93%

Table 6. Comparison of the proposed DP-CNN with previous methods applied on the NinaPro DB2.

Reference	Year	Dataset	Classifier	Accuracy
[35]	2014	DB2	SVM	60.27%
[48]	2017	DB2	CNN	60.31%
[45]	2019	DB2	CNN	83.70%
[26]	2021	DB2	DLPR	89.45%
[49]	2022	DB2	CNN	87.56%
This Work	2023	DB2	DP-CNN	94.00%

Table 7. Comparison of the proposed DP-CNN with previous methods applied on the NinaPro DB3.

Reference	Year	Dataset	Classifier	Accuracy
[35]	2014	DB3	SVM	46.27%
[48]	2017	DB3	CNN	73.31%
[45]	2019	DB3	CNN	64.30%
[26]	2021	DB3	DLPR	81.67%
[49]	2022	DB3	CNN	74.24%
This Work	2023	DB3	DP-CNN	85.36%

Table 8 shows the computational cost of our proposed model compared to other studies. When comparing our DP-CNN model to previous methods, we found that, although it achieves slightly lower accuracy in DB2 compared to CFF-RCNN [29], it has notable advantages in terms of computational efficiency. In DB2, where CFF-RCNN achieves an accuracy of 99.51%, our proposed DP-CNN maintains a competitive accuracy of 94.00%. However, DP-CNN outperforms CFF-RCNN in terms of training time, taking only 462.82 seconds compared to CFF-RCNN's 542.245 seconds. This represents a 14.56% reduction in training time, highlighting the efficiency of DP-CNN. Our proposed model also has a lower prediction time, further emphasizing its computational advantages.

Table 8. Comparison of the proposed DP-CNN with previous methods in terms of computational cost.

Reference	Year	Dataset	Classifier	Accuracy	Training Time	Prediction Time
[29]	2022	DB1	CFF-RCNN	88.87%	1727.415 s	-
[29]	2022	DB2	CFF-RCNN	99.51%	542.245 s	-
[30]	2023	DB1	E2CNN	91.27%	38.062 s	0.093 s
		DB1		94.93%	482.37 s	0.863 s
This Work	2023	DB2	DP-CNN	94.00%	462.82 s	0.649 s
		DB3		85.36%	501.23 s	0.895 s

3.4. Comparison and analysis of results with transfer learning models

Since the sEMG signals are converted to images, we have compared the performance of our proposed DP-CNN with pre-trained transfer learning models that has been trained on million of

images. The transfer learning models utilized for comparison in this study include AlexNet, MobileNet, VGG19, DenseNet121, and ResNet50. The performance of the DP-CNN on each database (DB1, DB2, and DB3) was compared to these transfer learning models. As shown in Figure 5, the DP-CNN outperformed all other models on DB1 and DB2, achieving accuracies of 94.00% and 94.93%, respectively. However, on DB3, the DP-CNN achieved an accuracy of 85.36%, which was lower than the accuracy achieved by AlexNet (87.29%) and VGG19 (87.96%). The other models achieved lower accuracies on all three databases compared to the DP-CNN.

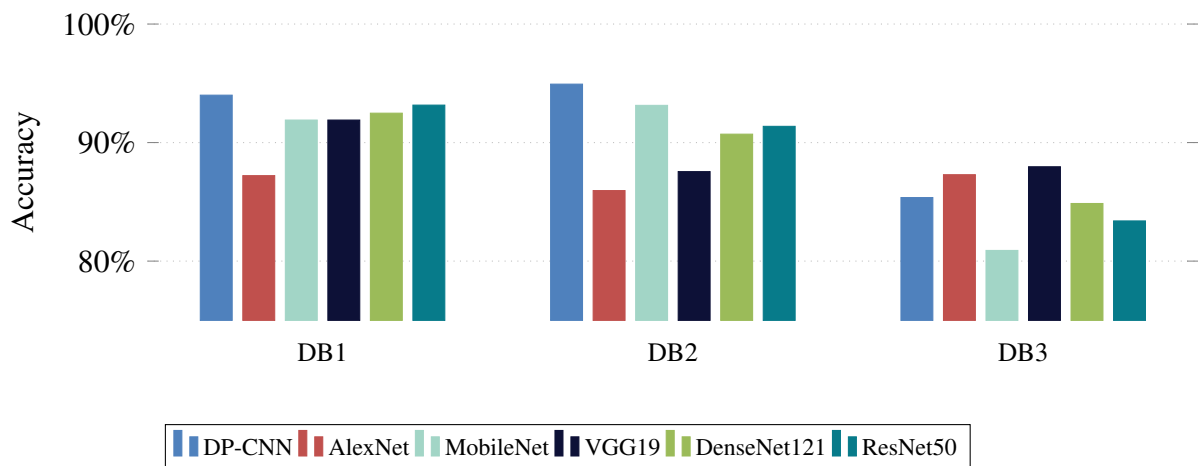


Figure 5. Performance of DP-CNN on each database.

The proposed DP-CNN model's performance on the NinaPro datasets was examined in this work. The method involved converting sEMG signals to LMS and then classifying them with the DP-CNN model. To assess the proposed model's performance, the findings were compared to earlier studies that used deep learning-based techniques on the same datasets.

The DP-CNN architecture comprises two pathways, namely the convolutional and dense pathways. These pathways are capable of capturing local patterns in the EMG signals and global patterns and relationships in the signals, respectively. Through the integration of the outputs from both pathways, the model can enhance its predictive accuracy and improve its capacity to classify EMG signals. The utilization of batch normalization and dropout layers in both pathways serves to regularize the model and mitigate overfitting.

The suggested DP-CNN model's findings were compared to earlier studies on the NinaPro DB1, DB2, and DB3. The accuracy gained by each model was used in the comparison. The suggested DP-CNN model surpassed earlier studies in terms of accuracy across all three datasets, according to the results. On DB1, the DP-CNN model achieved an accuracy of 94.93%, the highest accuracy achieved on this dataset thus far. On DB2, the DP-CNN model achieved an accuracy of 94.00%, the highest accuracy ever achieved on this dataset. Finally, on DB3, the DP-CNN model attained an accuracy of 85.36%, the highest accuracy achieved on this dataset thus far. These findings show that the DP-CNN model performs well for gesture classification across all three datasets.

4. Conclusions

In this study, we addressed the challenges associated with machine and deep learning algorithms, especially their performance decline in the face of increased number of classes, diverse data collected over multiple days, and population differences. Recognizing the need for a robust learning system, we proposed and applied a dual-pathway convolutional neural network (DP-CNN) to diverse datasets featuring both able-bodied and amputee subjects. The DP-CNN operated on Log-Mel spectrogram-based images derived from surface electromyography signals obtained from NinaPro DB1 and DB3. The results were benchmarked against other CNN models implemented on the same datasets, revealing the superior performance of the proposed DP-CNN. In DB1, the DP-CNN achieved a remarkable mean classification accuracy of 94.93%, a substantial 28.33% increase from the baseline and a noteworthy improvement of 6.73% over the previous highest accuracy. Similar advancements were observed in DB2 and DB3, showcasing the model's consistent and robust performance across datasets. The architecture of the DP-CNN, featuring convolutional and dense pathways, played a pivotal role in capturing both local and global patterns within EMG signals. Integrating outputs from these pathways enhanced predictive accuracy and classification capabilities. The incorporation of batch normalization and dropout layers in both pathways further contributed to model regularization and mitigated overfitting. Comparisons with prior studies on NinaPro DB1, DB2, and DB3 demonstrated that the DP-CNN consistently outperformed earlier models in terms of accuracy. Achieving the highest accuracy on each dataset—94.93% on DB1, 94.00% on DB2, and 85.36% on DB3—the DP-CNN showcased its effectiveness in gesture classification. Additionally, a comparative analysis against pre-trained transfer learning models, including AlexNet, MobileNet, VGG19, DenseNet121, and ResNet50, highlighted the DP-CNN's supremacy in terms of accuracy on DB1 and DB2. Although, on DB3, it slightly lagged behind specific models, but the overall performance improvement in sEMG-based gesture detection was significant. The DP-CNN model, equipped with dual pathways, proved to be an effective solution for improving the accuracy and robustness of sEMG-based gesture classification. This study contributes valuable insights into advancing machine learning techniques for prosthetic control applications, emphasizing the practical significance of employing sophisticated architectures like the DP-CNN in real-world scenarios.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) funded by the Korean government (MSIT) (No. RS-2023-00243034). This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (2021-0-00755, Dark data analysis technology for data scale and accuracy improvement). The authors also express their gratitude to Princess Nourah bint Abdulrahman University Researchers Supporting Project Number PNURSP2024R104, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. Q. Liu, A. Liu, X. Zhang, X. Chen, R. Qian, X. Chen, Removal of EMG artifacts from multichannel EEG signals using combined singular spectrum analysis and canonical correlation analysis, *J. Healthcare Eng.*, **2019** (2019), 4159676. <https://doi.org/10.1155/2019/4159676>
2. Y. Kim, S. Stapornchaisit, M. Miyakoshi, N. Yoshimura, Y. Koike, The effect of ICA and non-negative matrix factorization analysis for EMG signals recorded from multi-channel EMG sensors, *Front. Neurosci.*, **14** (2020), 600804. <https://doi.org/10.3389/fnins.2020.600804>
3. X. Xi, C. Yang, J. Shi, Z. Luo, Y. B. Zhao, Surface electromyography-based daily activity recognition using wavelet coherence coefficient and support vector machine, *Neural Process. Lett.*, **50** (2019), 2265–2280. <https://doi.org/10.1007/s11063-019-10008-w>
4. Z. Qin, Z. Jiang, J. Chen, C. Hu, Y. Ma, sEMG-based tremor severity evaluation for Parkinson's disease using a light-weight CNN, *IEEE Signal Process. Lett.*, **26** (2019), 637–641. <https://doi.org/10.1109/LSP.2019.2903334>
5. K. Leerskov, M. Rehman, I. Niazi, S. Cremoux, M. Jochumsen, Investigating the feasibility of combining EEG and EMG for controlling a hybrid human computer interface in patients with spinal cord injury, in *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, IEEE, (2020), 403–410. <https://doi.org/10.1109/BIBE50027.2020.00072>
6. F. Amin, A. Waris, J. Iqbal, S. O. Gilani, M. Z. ur Rehman, S. Mushtaq, et al., Maximizing stroke recovery with advanced technologies: A comprehensive assessment of robot-assisted, EMG-Controlled robotics, virtual reality, and mirror therapy interventions, *Results Eng.*, **21** (2024), 101725. <https://doi.org/10.1016/j.rineng.2023.101725>
7. T. W. Boonstra, L. Faes, J. N. Kerkman, D. Marinazzo, Information decomposition of multichannel EMG to map functional interactions in the distributed motor system, *NeuroImage*, **202** (2019), 116093. <https://doi.org/10.1016/j.neuroimage.2019.116093>
8. L. C. Chen, P. H. Chen, R. T. H. Tsai, Y. Tsao, Epg2s: Speech generation and speech enhancement based on electropalatography and audio signals using multimodal learning, *IEEE Signal Process. Lett.*, **29** (2022), 2582–2586. <https://doi.org/10.1109/LSP.2022.3184636>
9. S. Inam, S. Al-Harman, S. Shafique, M. Afzal, A. Rabail, F. Amin, et al., A brief review of strategies used for EMG signal classification, in *2021 International Conference on Artificial Intelligence (ICAI)*, IEEE, (2021), 140–145. <https://doi.org/10.1109/ICAI52203.2021.9445257>
10. L. Cai, S. Yan, C. Ouyang, T. Zhang, J. Zhu, L. Chen, et al., Muscle synergies in joystick manipulation, *Front. Physiol.*, **14** (2023), 1282295. <https://doi.org/10.3389/fphys.2023.1282295>

11. C. Shen, K. Zhang, J. Tang, A COVID-19 detection algorithm using deep features and discrete social learning particle swarm optimization for edge computing devices, *ACM Trans. Internet Technol.*, **22** (2022), 1–17. <https://doi.org/10.1145/3453170>
12. S. Khalil, U. Nawaz, Zubariah, Z. Mushtaq, S. Arif, M. Z. ur Rehman, et al., Enhancing ductal carcinoma classification using transfer learning with 3D U-Net models in breast cancer imaging, *Appl. Sci.*, **13** (2023), 4255. <https://doi.org/10.3390/app13074255>
13. Z. Mushtaq, M. F. Qureshi, M. J. Abbass, S. M. Q. Al-Fakih, Effective kernel-principal component analysis based approach for wisconsin breast cancer diagnosis, *Electron. Lett.*, **59** (2023), e212706. <https://doi.org/10.1049/ell2.12706>
14. Z. Hu, J. Tang, P. Zhang, J. Jiang, Deep learning for the identification of bruised apples by fusing 3D deep features for apple grading systems, *Mech. Syst. Signal Process.*, **145** (2020), 106922. <https://doi.org/10.1016/j.ymsp.2020.106922>
15. A. Shahzad, A. Mushtaq, A. Q. Sabeeh, Y. Y. Ghadi, Z. Mushtaq, S. Arif, et al., Automated uterine fibroids detection in ultrasound images using deep convolutional neural networks, *Healthcare*, **11** (2023), 1493. <https://doi.org/10.3390/healthcare11101493>
16. N. Afshan, Z. Mushtaq, F. S. Alamri, M. F. Qureshi, N. A. Khan, I. Siddique, Efficient thyroid disorder identification with weighted voting ensemble of super learners by using adaptive synthetic sampling technique, *AIMS Math.*, **8** (2023), 24274–24309. <https://doi.org/10.3934/math.20231238>
17. A. A. Khan, S. Raza, M. F. Qureshi, Z. Mushtaq, M. Taha, F. Amin, Deep learning-based classification of wheat leaf diseases for edge devices, in *2023 2nd International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETEECTE)*, IEEE, (2023), 1–6. <https://doi.org/10.1109/ETEECTE59617.2023.10396676>
18. D. Huang, B. Chen, Surface EMG decoding for hand gestures based on spectrogram and CNN-LSTM, in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*, IEEE, (2019), 123–126. <https://doi.org/10.1109/CCHI.2019.8901936>
19. J. O. Pinzón-Arenas, R. Jiménez-Moreno, A. Rubiano, Percentage estimation of muscular activity of the forearm by means of EMG signals based on the gesture recognized using CNN, *Sens. Bio-Sens. Res.*, **29** (2020), 100353. <https://doi.org/10.1016/j.sbsr.2020.100353>
20. B. Saeed, S. O. Gilani, Z. ur Rehman, M. Jamil, A. Waris, M. N. Khan, Comparative analysis of classifiers for EMG signals, in *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, IEEE, (2019), 1–5. <https://doi.org/10.1109/CCECE.2019.8861835>
21. N. K. Karnam, A. C. Turlapaty, S. R. Dubey, B. Gokaraju, Classification of sEMG signals of hand gestures based on energy features, *Biomed. Signal Process. Control*, **70** (2021), 102948. <https://doi.org/10.1016/j.bspc.2021.102948>
22. M. Akmal, S. Khalid, M. Moiz, M. J. Abbass, M. F. Qureshi, Z. Mushtaq, Leveraging training strategies of artificial neural network for classification of multiday electromyography signals, in *2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETEECTE)*, IEEE, (2022), 1–5. <https://doi.org/10.1109/ETEECTE55893.2022.10007103>

23. M. Akmal, M. F. Qureshi, F. Amin, M. Z. ur Rehman, I. K. Niazi, SVM-based real-time classification of prosthetic fingers using myo armband-acquired electromyography data, in *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, IEEE, (2021), 1–5. <https://doi.org/10.1109/BIBE52308.2021.9635461>
24. S. Inam, F. Amin, M. Z. ur Rehman, Comparative study of flexor and extensor muscles emg for upper limb prosthesis, in *2021 15th International Conference on Open Source Systems and Technologies (ICOSST)*, IEEE, (2021), 1–5. <https://doi.org/10.1109/ICOSST53930.2021.9683956>
25. Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, W. Geng, A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition, *PLoS One*, **13** (2018), e0206049. <https://doi.org/10.1371/journal.pone.0206049>
26. S. Pancholi, A. M. Joshi, D. Joshi, A robust and accurate deep learning based pattern recognition framework for upper limb prosthesis using semg, preprint, arXiv:2106.02463.
27. Y. Cheng, G. Li, M. Yu, D. Jiang, J. Yun, Y. Liu, et al., Gesture recognition based on surface electromyography-feature image, *Concurrency Comput. Pract. Exper.*, **33** (2021), e6051. <https://doi.org/10.1002/cpe.6051>
28. R. Tong, Y. Zhang, H. Chen, H. Liu, Learn the temporal-spatial feature of sEMG via dual-flow network, *Int. J. Humanoid Rob.*, **16** (2019), 1941004. <https://doi.org/10.1142/S0219843619410044>
29. P. Xu, F. Li, H. Wang, A novel concatenate feature fusion RCNN architecture for sEMG-based hand gesture recognition, *PLoS One*, **17** (2022), e0262810. <https://doi.org/10.1371/journal.pone.0262810>
30. M. F. Qureshi, Z. Mushtaq, M. Z. ur Rehman, E. N. Kamavuako, E2cnn: An efficient concatenated cnn for classification of surface emg extracted from upper limb, *IEEE Sens. J.*, **23** (2023), 8989–8996. <https://doi.org/10.1109/JSEN.2023.3255408>
31. M. F. Qureshi, Z. Mushtaq, M. Z. ur Rehman, E. N. Kamavuako, Spectral image-based multiday surface electromyography classification of hand motions using CNN for human–computer interaction, *IEEE Sens. J.*, **22** (2022), 20676–20683. <https://doi.org/10.1109/JSEN.2022.3204121>
32. H. Nodera, Y. Osaki, H. Yamazaki, A. Mori, Y. Izumi, R. Kaji, Deep learning for waveform identification of resting needle electromyography signals, *Clin. Neurophysiol.*, **130** (2019), 617–623. <https://doi.org/10.1016/j.clinph.2019.01.024>
33. D. Gao, X. Tang, M. Wan, G. Huang, Y. Zhang, EEG driving fatigue detection based on log-Mel spectrogram and convolutional recurrent neural networks, *Front. Neurosci.*, **17** (2023), 1136609. <https://doi.org/10.3389/fnins.2023.1136609>
34. T. Tuncer, S. Dogan, M. Baygin, U. R. Acharya, Tetromino pattern based accurate eeg emotion classification model, *Artif. Intell. Med.*, **123** (2022), 102210. <https://doi.org/10.1016/j.artmed.2021.102210>
35. M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A. G. M. Hager, S. Elsig, et al., Electromyography data for non-invasive naturally-controlled robotic hand prostheses, *Sci. Data*, **1** (2014), 140053. <https://doi.org/10.1038/sdata.2014.53>

36. A. Gijsberts, M. Atzori, C. Castellini, H. Müller, B. Caputo, Measuring movement classification performance with the movement error rate, *IEEE Trans. Neural Syst. Rehabil. Eng.*, **89621** (2014), 735–744.
37. M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A. G. M. Hager, E. Simone, et al., Clinical parameter effect on the capability to control myoelectric robotic prosthetic hands, *J. Rehabil. Res. Dev.*, **53** (2016), 345–358. <http://doi.org/10.1682/JRRD.2014.09.0218>
38. M. Atzori, M. Cognolato, H. Müller, Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands, *Front. Neurobot.*, **10** (2016), 9. <https://doi.org/10.3389/fnbot.2016.00009>
39. X. Zhang, X. Li, O. W. Samuel, Z. Huang, P. Fang, G. Li, Improving the robustness of electromyogram-pattern recognition for prosthetic control by a postprocessing strategy, *Front. Neurobot.*, **11** (2017), 51. <https://doi.org/10.3389/fnbot.2017.00051>
40. F. Riillo, L. Quitadamo, F. Cavrini, E. Gruppioni, C. Pinto, N. C. Pastò, et al., Optimization of EMG-based hand gesture recognition: Supervised vs. unsupervised data preprocessing on healthy subjects and transradial amputees, *Biomed. Signal Process. Control*, **14** (2014), 117–125. <https://doi.org/10.1016/j.bspc.2014.07.007>
41. S. Lu, J. Yang, B. Yang, X. Li, Z. Yin, L. Yin, et al., Surgical instrument posture estimation and tracking based on lstm, *ICT Express*, in press. <https://doi.org/10.1016/j.ict.2024.01.002>
42. S. Zhao, W. Liang, K. Wang, L. Ren, Z. Qian, G. Chen, et al., A multiaxial bionic ankle based on series elastic actuation with a parallel spring, *IEEE Trans. Ind. Electron.*, **2023** (2023), 1–13. <https://doi.org/10.1109/TIE.2023.3310041>
43. W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, J. Li, Gesture recognition by instantaneous surface EMG images, *Sci. Rep.*, **6** (2016), 36571. <https://doi.org/10.1038/srep36571>
44. W. Wei, Y. Wong, Y. Du, Y. Hu, M. Kankanhalli, W. Geng, A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface, *Pattern Recognit. Lett.*, **119** (2019), 131–138. <https://doi.org/10.1016/j.patrec.2017.12.005>
45. W. Wei, Q. Dai, Y. Wong, Y. Hu, M. Kankanhalli, W. Geng, Surface-electromyography-based gesture recognition by multi-view deep learning, *IEEE Trans. Biomed. Eng.*, **66** (2019), 2964–2973. <https://doi.org/10.1109/TBME.2019.2899222>
46. H. Wang, Y. Zhang, C. Liu, H. Liu, sEMG based hand gesture recognition with deformable convolutional network, *Int. J. Mach. Learn. Cybern.*, **13** (2022), 1729–1738. <https://doi.org/10.1007/s13042-021-01482-7>
47. Y. Zhang, F. Yang, Q. Fan, A. Yang, X. Li, Research on sEMG-based gesture recognition by dual-view deep learning, *IEEE Access*, **10** (2022), 32928–32937. <https://doi.org/10.1109/ACCESS.2022.3158667>
48. X. Zhai, B. Jelfs, R. H. M. Chan, C. Tin, Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network, *Front. Neurobot.*, **11** (2017), 379. <https://doi.org/10.3389/fnins.2017.00379>

-
49. J. A. Sandoval-Espino, A. Zamudio-Lara, J. A. Marbán-Salgado, J. J. Escobedo-Alatorre, O. Palillero-Sandoval, J. G. Velásquez-Aguilar, Selection of the best set of features for sEMG-based hand gesture recognition applying a CNN architecture, *Sensors*, **22** (2022), 4972. <https://doi.org/10.3390/s22134972>



AIMS Press

©2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)