Moving Towards Combining Deep Feature Learning and Domain Knowledge-Guided Feature Engineering for Surface Electromyography-Based Hand Movement Recognition

Hàorán, Mùyáng
Sichuan University, Chengdu, Sichuan 610207, China

Abstract: Surface electromyography (sEMG)-based hand movement recognition is a machine-learning-driven decision-making challenge that is crucial to the reliable operation of noninvasive neural interfaces like rehabilitation robots and myoelectric prostheses. The noisy, random, and nonstationary nature of sEMG signals continues to limit the performance of today's sEMG-based hand movement recognition systems, despite recent advancements in sEMG-based hand movement recognition using end-to-end deep feature learning technologies based on deep learning models. Researchers have developed a number of techniques that enhance sEMG-based hand movement via feature engineering. This research proposed a progressive fusion network (PFNet) framework that integrates deep feature learning and domain knowledge-guided feature engineering to improve sEMG-based hand movement recognition accuracy while allowing for a trade-off between computational complexity and performance. Specifically, a feature learning network and a domain knowledge network are used to learn high-level feature representations from raw sEMG signals and engineered time-frequency domain features, respectively. A three-stage progressive fusion strategy is then used to gradually fuse the two networks together and derive the final decisions. Our proposed PFNet was evaluated through extensive experiments on five sEMG datasets. The experimental results demonstrated that the proposed PFNet outperformed the state of the arts in hand movement recognition, achieving average accuracies of 87.8%, 85.4%, 68.3%, 71.7%, and 90.3% on the five datasets, respectively.

1. Introduction

All Surface electromyography (sEMG)-based hand movement recognition has been extensively studied in the fields of human-computer interaction [3,4] and rehabilitation engineering [1,2] as an accurate and noninvasive method of deciphering user intent of hand movements. Many attempts have been made to improve sEMG-based hand movement recognition by creating more representative features [5], creating more complex machine-learning models [6], and expanding the number of sensors [7] after it was discovered that one of the main problems with this method is the machine-learning-driven decision-making problem of classifying sequences of sEMG signals.

Existing sEMG-based hand movement identification techniques may be roughly divided into two categories from a machine learning standpoint: (1) feature engineering-based techniques and (2) feature learning-based techniques [8]. While the latter refers to techniques based on end-to-end deep learning models that are capable of learning representative high-level features from raw sEMG signals without the need for any engineered features, the former refers to techniques based on traditional shallow learning models and handcrafted time domain (TD), frequency domain (FD), or time-frequency domain (TFD) features.

In sEMG-based hand movement recognition, feature learning techniques based on end-to-end deep learning models, like convolutional neural networks (CNNs) [9] and recurrent neural networks (RNNs) [10], have been extensively researched throughout the last five years. However, because sEMG is noisy, unpredictable, and nonstationary, researchers have also shown that it is still difficult to achieve reliable sEMG-based hand movement detection accuracy for end-to-end deep learning models. For instance, on the large-scale noninvasive adaptive prosthetics (NinaPro) dataset, one of the earliest studies in this field found that the end-to-end CNN model's average hand movement recognition accuracy was much lower than that of traditional shallow learning models like random forests and support vector machines (SVM) [11]. Subsequent research on this dataset [12,13] showed more encouraging outcomes, with the manually optimized and fine-tuned end-to-end deep learning models outperforming the shallow learning models.

The hand movement recognition performance of traditional feature engineering approaches is primarily dependent on feature selection and extraction, which is typically carried out manually using the domain knowledge gathered from a large number of experiments and evaluations in the field, as opposed to feature learning approaches. Deep learning-based myoelectric pattern identification is frequently believed to benefit from such heuristically obtained domain knowledge [14]. In order to extract and assess various designed characteristics as the input of their deep learning models, several recent works in this field have attempted to do so. For instance, Millar et al. [15] used a long short-term memory (LSTM) model to extract a collection of 11 TD characteristics from sEMG signals for hand movement detection. They were able to classify a series of functional grasps on two diametric objects with an average recognition accuracy of 99.8%. Using a CNN model, Cheng et al. [16] developed a multi-sEMG feature image for hand movement detection by extracting two TD and one FD feature from sEMG signals. They were able to categorize 52 hand movements across 27 patients with an average recognition accuracy of 82.5%. In their evaluation of several input modalities for a CNN model with transfer learning architecture, Allard et al. [17] discovered that continuous wavelet transform (CWT) features and short-time Fourier transform-based spectrograms performed better than raw sEMG signals in classifying seven hand movements across 17 subjects. In order to perform stacking en- semble CNN-based hand movement identification, Shen et al. [18] retrieved FD and TFD features from sEMG signals, represented them by pictures, and used them to categorize 40 hand movements across 10 participants with an average recognition accuracy of 72.1%. Our previous study [14] classified 50 hand movements across 40 people with an average recognition accuracy of 83.7% by extracting three sets of features from sEMG signals and constructing them into multi-view representations of sEMG signals for hand movement recognition.

In conclusion, current deep learning techniques for sEMG-based hand movement identification can be divided into two categories based on their input: end-to-end and non-end-to-end. Even while the current non-end-to-end deep learning methods used manufactured features rather than raw sEMG signals as their input to increase the performance of sEMG-based hand movement recognition, they mostly disregarded the deep learning models' feature learning capabilities. In other words, the choice of engineering features, which is typically based on domain knowledge or offiline experimental results on a small amount of data, has a significant impact on their hand movement recognition ability. Furthermore, the deployment of these techniques in real-time systems was restricted due to the additional computing time and resources needed for the feature engineering process for methods that used numerous engineered features as the input of deep learning models [14,18].

In order to improve sEMG-based hand movement detection, we propose a progressive fusion network (PFNet) in this study. This network integrates CNN-based deep feature learning and domain knowledge-guided feature engineering in a progressive manner. Specifically, the feature learning network, domain knowledge network, and progressive fusion module are the three components of the suggested PFNet design. The progressive fusion module progressively integrates the two networks through a three-stage procedure. The feature learning network and the domain knowledge network, respectively, learn high-level feature representations from raw sEMG signals and designed features.

The proposed PFNet architecture has two main contributions:

(1) To independently learn discriminative high-level feature representations from raw sEMG signals and the wavelet packet-based TFD features that have been shown to be successful for sEMG-based hand movement recognition in preliminary studies, we constructed two separate neural networks, the feature learning network and the domain knowledge network. This way, both deep feature learning and heuristically accumulated domain knowledge can be used to improve the hand movement recognition performance.
(2) To further integrated domain knowledge-guided feature engineering and deep feature learning in sEMG-based hand movement recognition, we used a three-step procedure. Specifically, two subnetworks were used to fuse the high-level feature representations learned at two different depths of the feature learning network and the do-main knowledge network together. This was done first using feature-level fusion, and then decision-level fusion was used to fuse the output decisions of the two subnetworks together. The performance of hand movement recognition is thought to be enhanced by the ability of such a three-stage integration technique to learn more varied high-level feature representations.

In addition to demonstrating the efficacy of combining deep feature learning and domain knowledge-guided feature engineering in sEMG-based hand movement identification, the experimental findings on five datasets also showed that our method performed better than other cutting-edge techniques.

2. Supplies and Procedures

2.1. Preprocessing and datasets.

Five subdatasets of the NinaPro repository [19], which makes multi-channel sEMG signals from intact patients and trans-radial amputees publically available, were used for the experiments in this investigation. The sEMG datasets utilized in this work are briefly described in Table 1, with the following full descriptions:

NinaPro's initial subdataset, known as NinaP-roDB1, offers 10-channel sEMG signals gathered from 53 hand movements made by 27 healthy participants. Each hand movement was performed ten times (i.e., ten trials per hand movement) in NinaProDB1, which was divided into twelve finger movements (called Exercise A), seventeen wrist movements and hand postures (called Exercise B), twenty-three grasping and functional movements (called Exercise C), and the rest movement [20]. For the convenience of performance comparison, we also eliminated the rest movement from our experiments, much as the majority of previous studies on this NinaProDB1 did [10,12,14,22].

NinaPro's second subdataset, known as NinaP-roDB2, offers 12-channel sEMG signals gathered from 50 hand movements made by 40 healthy participants. Nine force patterns (referred to as Exercise D), 23 grasping and functional movements (similar to Exercise C in NinaProDB1), 17 wrist movements and hand postures (similar to Exercise B in NinaProDB1), and the rest movement were the categories into which the hand movements in NinaProDB2 were divided. Each hand movement was performed six times, or six trials per hand movement [20].

Eleven trans-radial amputee participants supplied 12-channel sEMG signals from 50 hand movements in the third subdataset of NinaPro, known as NinaP-roDB3. Each hand movement was performed six times, or six trials per hand movement, and the hand movements in NinaProDB3 are identical to those in NinaProDB2 [20]. According to Atzori et al. [20], two trans-radial amputee subjects utilized only 10 electrodes to collect sEMG signals due to a lack of space, and three trans-radial amputee subjects stopped the experiment early because they were tired or in pain during the NinaProDB3 data recording process. To guarantee that each subject had the same number of hand movement repetitions and sEMG channels, the data from these subjects were excluded from our experiments.
NinaPro's fourth subdataset, known as NinaP-roDB4, offers 12-channel sEMG signals gathered from 53 hand movements made by ten healthy participants. Each hand movement was replicated six times, or six trials per hand movement, and the hand movements in NinaProDB4 are identical to those in NinaProDB1 [21]. Two people (i.e., subjects 4 and 6) did not complete all hand movements, and their data were excluded from our studies after we checked the data.

NinaPro's fifth subdataset, known as NinaP-roDB5, offers 16-channel sEMG signals gathered from 53 hand movements made by ten healthy participants. Each hand movement was replicated six times, or six trials per hand movement, and the hand movements in NinaProDB5 are identical to those in NinaProDB1 [21]. In our experiments, we classified a subset of 41 hand movements; the details of the chosen hand movements are available in [21].

Otto Bock 13E200-50 electrodes recorded the sEMG signals in NinaProDB1 at a sampling rate of 100 Hz, Delsys Trigno Wireless electrodes recorded the sEMG signals in NinaProDB2 and DB3 at a sampling rate of 2k Hz, and the Cometa Wave Plus Wireless sEMG system recorded the sEMG signals in NinaProDB4 at a sampling rate of 2k Hz [20,21]. We downsampled the sEMG signals in NinaProDB2-NinaP-roDB4 from 2k Hz to 100 Hz due to memory constraints. In [14], the identical experimental setup was also used.

Each dataset's raw sEMG signals were divided into segments using sliding windows. We used sliding windows that were no longer than 200 ms for all trials in this investigation in order to segment raw sEMG signals because preliminary research [23,24] has shown that the maximum permitted time delay of real-time myoelectric control systems is 300 ms. The results and discussion part of this study will provide a detailed explanation of the sliding window lengths and stages employed in this investigation.

2.2. Knowledge-Guided Domain Feature

Feature augmentation and engineering. The discrete wavelet transform (DWT) is a time-frequency analysis method that uses a set of half-band filters that are determined by two orthogonal wavelet basis functions to iteratively decompose the original discrete time series into wavelet coefficients in multiresolution sub-bands [25]. A half-band low-pass filter and a half-band high-pass filter, as illustrated in Figure 1(a), break down the original signals X into two sequences of coefficients in the lower resolution space at the first wavelet level. These are the wavelet coefficients CD, which represent the detailed representation of X, and the scaling coefficients CA, which represent the approximate representation of X. The decomposed scaling coefficients at the following wavelet levels undergo an iterative repetition of this procedure, producing a two-channel tree structure that subsamples the signals by two at each node.

In the lower resolution space at each wavelet level, the discrete wavelet packet transform (DWPT), an extension of DWT, breaks down both scaling and wavelet coefficients into two sequences of coefficients. The outputs of DWPT, as illustrated in Figure 1(b), consist of 23 8 sequences of DWPT coefficients (DWPTCs) when the wavelet level is 3. This can be thought of as the multiresolution representation of the original signals X in 8 sub-bands.

A popular feature engineering method for extracting TFD features in sEMG-based hand movement identification is the DWPT. Statistical characteristics including energy, average value, standard deviation, skewness, and kurtosis are typically extracted using traditional shallow learning techniques. While the majority of state-of-the-art techniques use images created from DWPTCs in all sub-bands to form the input of deep neural networks [14, 18], conventional shallow learning techniques typically extract statistical features like energy, average value, standard deviation, skewness, and kurtosis from DWPTCs as the input of their classifiers [26, 27]. In our earlier research [14], we assessed 11 engineered features and feature sets as input to a CNN model for sEMG-based hand movement recognition. The findings demonstrated that DWPTCs outperformed all other features and feature sets in terms of hand movement recognition accuracy across various datasets.

In this study, the input pictures of the domain knowledge network were generated by extracting the DWPTCs from raw sEMG signals based on the previously indicated domain information. This study's DWPT hyperparameters are identical to those from our earlier investigation [14]. The wavelet level k was set to log 2N, where N is the length of the input signals (i.e., the length of the sliding window), and we specifically utilized the Daubechies 1 wavelet basis function. A DWPTC image was created by stacking the DWPTC vectors from every sEMG channel after the resulting 2k DWPTC sequences in each sub-band were concatenated to create a DWPTC vector for each sEMG channel.

2.3. The suggested architecture for PFNet.

Our suggested PFNet's architecture, which includes a progressive fusion module, a feature learning network, and a domain knowledge network, is shown in Figure 2. Assuming C-channel sEMG signals are segmented using N-frame sliding windows, the input images of the domain knowledge network are D × M reorganized DWPTC images, as covered in the previous subsection, and the input images of the feature learning network are N × C sEMG images, which are created by stacking C-channel raw sEMG signals together.

2.4. Network for Feature Learning.

Two convolutional layers with 3 × 3 filters, two locally connected layers with 11 filters, and one completely connected layer with 512 hidden units make up the feature learning network, which learns features from raw sEMG signals. In the feature learning network, each neural network layer's output feature map count was set to 64. The first four neural network layers of GengNet [12], which shown encouraging sEMG-based hand movement identification performance in previous studies [12–14], share the same architecture as the feature learning network.

2.5. Domain Knowledge Network

Reorganized DWPTC images are used to teach the domain knowledge network high-level feature representations. One convolutional layer with 1 × 1 filters, one convolutional layer with 2 × 2 filters, two locally connected layers with 1 × 1 filters, and one fully connected layer with 1024 hidden units make up the domain knowledge network's network architecture, which differs slightly from the feature learning network's.

Additionally, 64 output feature maps were set for each neural network layer in the domain knowledge network.

2.6. Module for Progressive Fusion.

Feature-level fusion and decision-level fusion are two types of conventional fusion methods for handling feature vectors from multiple sources. The former concatenates the feature vectors and feeds the resulting feature vector into the classifier, while the latter constructs separate classifiers for each data source's feature vector and then combines the decisions to create the final decisions [29].

In this case, ‖ and ⨁ stand for the concatenation and element-wise summation operations, respectively. θi and yi stand for their parameters and output choices, respectively, while Hi(i 1, 2) are two subnetworks enabling feature-level fusion of high-level features learnt at two distinct layers of feature learning network and domain knowledge network. Equation (3) illustrates that the final choice (classification result) is obtained by adding the output judgments of two subnetworks, H1 and H2, which are represented by softmax scores, at the third stage of fusion.

In Figure 3, we designated the first and second subnetworks (i.e., H1 and H2) with blue and red lines, respectively, to provide a clearer image of the two subnetworks for feature-level fusion in the three-stage progressive fusion process.

2.7. Hyperparameter settings and neural network configurations.

We accelerated the training process by using rectified linear unit (ReLU) activation function [31] and batch normalization [30] to each neural network layer of the PFNet to decrease the internal covariate shift. In order to prevent overfitting, we also implemented dropout regularization [32] after five neural network layers, as illustrated in Figure 3. These layers include the first fully connected layer of the first subnetwork H1, the second locally connected layers, the first fully connected layers of the feature learning network, and the domain knowledge network.

We adopted a pre-training technique, which has been widely used in sEMG-based hand movement recognition systems [10,12–14,33], for all trials in this investigation to avoid overfitting. Specifically, we pre-trained a model using all of the training data that was available for each experiment, and then we utilized the pre-trained model as the initial model in each validation fold. The stochastic gradient descent (SGD) technique was used for both pre-training and training, with a batch size of 1000 and a training epoch count of 28. We also used a learning rate decay technique [34] to increase convergence. At the 16th and 24th epochs, we divided the learning rate by 10 after initializing it at 0.1. The dropout rate for layers with dropout regularization was adjusted to 0.5 before training and to.65 during training.

Metrics for evaluation. The assessment metrics employed in this work were identical to those used in previous research on the NinaPro dataset for the sake of performance comparison [10,12,14,20,22,33,35]. We specifically adhered to the intra-subject classification schemes suggested by the NinaPro dataset's author [20,21], which used sEMG signals from roughly two-thirds of each subject's hand movement repetitions as the training set and the remaining hand movement repetitions from the same subject as the test set. The average of the attained accuracies across all individuals is used to determine the final hand movement recognition accuracy on each dataset.

NinaProDB1: the test set consists of the sEMG signals from the second, fifth, and tenth repetitions of all hand movements, while the training set consists of the sEMG signals from the first, third, fourth, sixth, seventh, eighth, and ninth repetitions of all hand movements.

NinaProDB2, NinaProDB3, NinaProDB4, and NinaProDB5: the training set consists of the sEMG signals from the first, third, fourth, and sixth repetitions of every hand movement, whereas the test set consists of the sEMG signals from the second and fifth repetitions of every hand movement.

3. Findings and Conversation

3.1. Efficiency and Computational Time.

Every experiment in this study was carried out on an NVIDIA GeForce GTX 1080 Ti GPU using MXNet [36]. Since all of the offiine experimental data (i.e., sEMG signals) are stored on a network-attached storage (NAS) device, it is challenging to estimate the computational time of our proposed PFNet for sEMG-based hand movement recognition in real-world scenarios. In our experiments, the hardware factors that affected the computational time and training speed included not only the GPU utilization percentage but also the network throughput. Nevertheless, the approximate computational time and training efficiency that we determined are as follows. It took roughly 23–30 minutes on NinaProDB1, 11–17 minutes on NinaProDB2, 18–20 minutes on NinaProDB3, 37–39 minutes on NinaProDB4, and 3–4 minutes on NinaProDB5 to train each fold (i.e., each subject) of the intra-subject evaluation. The training speed on NinaProDB1, NinaProDB2, NinaProDB3, NinaProDB4, and NinaProDB5 was roughly 3500 samples per second, 3300 samples per second, 6400 samples per second, 3300 samples per second, and 3500 samples per second, respectively.

3.2. Ablation Research on the Suggested Approach.

"Ablation studies" in machine learning typically refer to a process that evaluates specific deep neural network components while excluding the other components from the evaluation. In order to confirm the efficacy of the suggested PFNet, we carried out two ablation tests on it, which are as follows:

(1) Ablation Study 1: To confirm the efficacy of integrating domain knowledge-guided feature engineering and deep feature learning in sEMG-based hand movement recognition, a performance comparison was conducted between the proposed PFNet, PFNet without the domain knowledge network and its input (referred to as FLonly), and PFNet without the feature learning network and its input (referred to as DKonly). Figures 4(a) and 4(b) show the neural network architectures of FLonly and DKonly, respectively.
(2) Ablation Study 2: a comparison of the performance of various methods for fusing feature learning networks and domain knowledge networks, including two feature-level fusion techniques, the decision-level (i.e., score)

fusion approach, and the suggested progressive fusion module.
With the exception of experiments on NinaProDB5, where we used the experimental setup employed by Pizzolato et al. [21] and our earlier study [14], which set the window step to 100 ms, the sliding window length and window step were set to 200 ms and 10 ms, respectively, for all of these ablation studies.

3.3. Evaluation against the Industry Standard.

Additionally, we contrasted the suggested PFNet's average hand movement recognition accuracy with the state-of-the-art. We assessed the hand movement identification accuracy attained with sliding windows of 50 ms, 100 ms, 150 ms, and 200 ms, and we only looked at the state of the art that employed the same intra-subject categorization schemes as explained in Section 2.4 in order to provide a fair performance comparison. Except for trials on NinaProDB5 with sliding windows that slid for 50 ms, 100 ms, and 150 ms, where we set the window step to 10 ms, the window step parameters were the same as those used in the ablation research.
The hand movement recognition accuracy of our suggested PFNet and the state-of-the-art on NinaProDB1, NinaProDB2, NinaProDB3, NinaProDB4, and NinaProDB5 is shown in Table 2. Our suggested PFNet outperformed all of the cutting-edge deep learning techniques [10–14, 16, 18, 22, 37, 38] and shallow learning techniques [20, 21] listed in Table 2 on NinaProDB2, NinaProDB3, NinaProDB4, and NinaProDB5, according to the experimental results. In our previous study, we presented MV-CNN, which performed better on NinaProDB1 than our proposed PFNet [14]. However, it should be noted that the performance difference between PFNet and MV-CNN on NinaProDB1 was negligible. MV-CNN is a multi-view deep learning technique that uses three high-dimensional feature sets as its input. These findings show that both feature learning and domain knowledge-guided feature engineering can be used to enhance sEMG-based hand movement recognition in our suggested PFNet architecture.

4. Conclusion

The progressive fusion network (PFNet) framework was proposed in this study with the goal of improving sEMG-based hand movement recognition. It uses a progressive fusion module to fuse the two networks together in a three-stage process and obtain the final decisions after learning high-level feature representations from raw sEMG signals and discrete wavelet packet transform coefficients (DWPTCs) via a feature learning network and a domain knowledge network, respectively.
Five open-source sEMG datasets (NinaProDB1–NinaProDB5) were used for ablation studies. The experimental findings demonstrated the efficacy of the suggested progressive fusion module and the integration of deep feature learning and domain knowledge-guided feature engineering in sEMG-based hand movement recognition.

Additionally, we performed a performance comparison on NinaProDB1-NinaProDB5 against the state of the art. According to the experimental results, the suggested PFNet outperformed the state-of-the-art methods on the majority of the evaluated datasets, achieving average hand movement recognition accuracies of 87.8 ± 4.2%, 85.4 ± 5..1%, 68.3 ± 9.2%, 71.7 ± 7.4%, and 90.3 ± 3.2% on NinaProDB1, NinaProDB2, NinaProDB3, NinaProDB4, and NinaProDB5, respectively. Our suggested PFNet may achieve higher or nearly the same hand movement recognition accuracies using only one type of engineered feature, in contrast to our recently described technique that employed numerous engineered feature sets as its input [14].

Since real-time sEMG-based hand movement recognition systems typically require a more lightweight machine-learning model with fewer parameters and less computational complexity, future improvements to the suggested PFNet framework will concentrate on simplifying the deep neural network architecture without sacrificing performance.

References

[1] J. Fajardo, V. Ferman, D. Cardona, G. Maldonado, A. Lemus, and E. Rohmer, “Galileo hand: an anthropomorphic and affordable upper-limb prosthesis,” *IEEE Access*, vol. 8, pp. 81365–81377, 2020.
[2] R. Ma, L. Zhang, G. Li, D. Jiang, S. Xu, and D. Chen, “Grasping force prediction based on sEMG signals,” *Alex- andria Engineering Journal*, vol. 59, no. 3, pp. 1135–1147, 2020.
[3] U. Coˆte´-Allard, G. Gagnon-Turcotte, F. Laviolette, and B. Gosselin, “A low-cost, wireless, 3-D-printed custom armband for sEMG hand gesture recognition,” *Sensors*, vol. 19, no. 12, p. 2811, 2019.
[4] Y. Sun, C. Xu, G. Li et al., “Intelligent human computer interaction based on non redundant EMG signal,” *Alexandria Engineering Journal*, vol. 59, no. 3, pp. 1149–1157, 2020.

[5] R. N. Khushaba, A. H. Al-Timemy, A. Al-Ani, and A. Al- Jumaily, “A framework of temporal-spatial descriptors-based feature extraction for improved myoelectric pattern recog- nition,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1821–1831, 2017.
[6] V. H. Cene and A. Balbinot, “Resilient emg classification to enable reliable upper-limb movement intent detection,” *IEEE Transactions on Neural Systems and Rehabilitation Engi- neering*, vol. 28, no. 11, pp. 2507–2514, 2020.
[7] C. Amma, T. Krings, J. Bo¨er, and T. Schultz, “Advancing muscle-computer interfaces with high-density electromyography,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 929–938, Seoul, Korea, July 2015.
[8] A. Phinyomark and E. Scheme, “EMG pattern recognition in the era of big data and deep learning,” *Big Data and Cognitive Computing*, vol. 2, no. 3, 2018.
[9] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, “Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method,” *IEEE Journal of Biomedical and Health Informatics*, vol. 1, 2020.
[10] Y. Hu, Y. Wong, and W. Wei, “A novel attention-based hybrid CNN-RNN architecture for semg-based gesture recognition,” *PLoS One*, vol. 13, 2018.
[11] M. Atzori, M. Cognolato, and H. Mu¨ller, “Deep learning with convolutional neural networks applied to electromyography data: a resource for the classification of movements for prosthetic hands,” *Frontiers in Neurorobotics*, vol. 10, no. 9, 2016.
[12] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, “Gesture recognition by instantaneous surface EMG images,” *Scientific Reports*, vol. 6, no. 1, Article ID 36571, 2016.
[13] W. Wei, Y. Wong, Y. Du, Y. Hu, M. Kankanhalli, and W. Geng, “A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer inter- face,” *Pattern Recognition Letters*, vol. 119, pp. 131–138, 2019.
[14] W. Wei, Q. Dai, Y. Wong, Y. Hu, M. Kankanhalli, and W. Geng, “Surface-electromyography-based gesture recognition by multi-view deep learning,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 10, pp. 2964–2973, 2019.
[15] C. Millar, N. Siddique, and E. Kerr, “Lstm classification of functional grasps using semg data from low-cost wearable sensor,” in *Proceedings of the International Conference on Control, Automation and Robotics (ICCAR)*, pp. 213–222, Singapore, 2021.
[16] Y. Cheng, G. Li, and M. Yu, “Gesture recognition based on surface electromyography-feature image,” *Concurrency and Computation: Practice and Experience*, vol. 33, no. 6, Article ID e6051, 2021.
[17] U. Coˆte´-Allard, C. L. Fall, A. Drouin et al., “Deep learning for electromyographic hand gesture signal classification using transfer learning,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 760–771, 2019.
[18] S. Shen, K. Gu, X.-R. Chen, M. Yang, and R.-C. Wang, “Movements classification of multi-channel sEMG based on CNN and stacking ensemble learning,” *IEEE Access*, vol. 7, pp. 137489–137500, 2019.
[19] M. Atzori, A. Gijsberts, and S. Heynen, “Building the Ninapro database: a resource for the biorobotics community,” in *Proceedings of the IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp. 1258–1265, Rome, Italy, July 2012.
[1] M. Atzori, A. Gijsberts, C. Castellini et al., “Electromyography data for non-invasive naturally-controlled robotic hand prostheses,” *Scientific Data*, vol. 1, Article ID 140053, 2014.
[2] S. Pizzolato, L. Tagliapietra, and M. Cognolato, “Comparison of six electromyography acquisition setups on hand movement classification tasks,” *PLoS One*, vol. 12, no. 10, 2017.
[3] A. E. Olsson, A. Bjo¨rkman, and C. Antfolk, “Automatic discovery of resource-restricted convolutional neural network topologies for myoelectric pattern recognition,” *Computers in Biology and Medicine*, vol. 120, Article ID 103723, 2020.
[4] B. Hudgins, P. Parker, and R. N. Scott, “A new strategy for multifunction myoelectric control,” *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 1, pp. 82–94, 1993.
[5] K. Englehart and B. Hudgins, “A robust, real-time control scheme for multifunction myoelectric control,” *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 7, pp. 848–854, 2003.
[6] X. Zhou, C. Zhou, and B. G. Stewart, “Comparisons of discrete wavelet transform, wavelet packet transform and stationary wavelet transform in denoising PD measurement data,” in *Peroceedings of the Conference Record of the IEEE International Symposium on Electrical Insulation*, pp. 237–240, Toronto, Canada, June 2006.
[7] K. Kiatpanichagij and N. Afzulpurkar, “Use of supervised discretization with pca in wavelet packet transformation- based surface electromyogram classification,” *Biomedical Signal Processing and Control*, vol. 4, no. 2, pp. 127–138, 2009.
[8] Y. Hu and Q. Wang, “A comprehensive evaluation of hidden Markov model for hand movement recognition with surface electromyography,” *Proceedings of the International Confer- ence on Robotics, Intelligent Control and Artificial Intelligence*, vol. 85, 2020.
[9] W. Jiang and Z. Yin, “Human activity recognition using wearable sensors by deep convolutional neural networks,” in *Proceedings of the ACM International Conference on Multi- media*, pp. 1307–1310, New York, NY, USA, 2015.
[10] S. Poria, E. Cambria, and A. Gelbukh, “Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis,” in *Pro- ceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2539–2544, Lisbon, Portugal, March 2015.
[11] S. Ioffe and C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift,” in *Proceedings of the International Conference on Machine Learning*, pp. 448–456, Lille, France, 2015.
[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
[14] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng, “Surface EMG- based inter-session gesture recognition enhanced by deep domain adaptation,” *Sensors*, vol. 17, no. 3, 2017.
[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, Las Vegas, NV, USA, June 2016.

[16] S. Padhy, "A tensor-based approach using multilinear SVD for hand gesture recognition from semg signals," *IEEE Sensors Journal*, vol. 1, 2020.

[17] T. Chen, "MXNet: a flexible and efficient machine learning library for heterogeneous distributed systems," *Neural Information Processing Systems, Workshop on Machine Learning Systems*, vol. 42, 2015.

[18] X. Zhai, B. Jelfs, R. H. M. Chan, and C. Tin, "Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network," *Frontiers in Neu- roscience*, vol. 11, p. 379, 2017.

[19] J. L. Betthauser, J. T. Krall, S. G. Bannowsky et al., "Stable responsive EMG sequence prediction and adaptive reinforcement with temporal convolutional networks," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 6, pp. 1707–1717, 2020.