Explainable AI-powered Graph Neural Networks for HD EMG-based Gesture Intention Recognition

Silvia Maria Massa, Daniele Riboni, Kianoush Nazarpour

***Abstract*—The ability to recognize fine-grained gestures enables several applications in different domains, including healthcare, robotics, remote control, and human-computer interaction. Traditional gesture recognition systems rely on data acquired from cameras, depth sensors, or smart gloves. More recently, techniques for recognizing gestures based on signals acquired by high-density (HD) EMG electrodes worn on the forearm have been proposed. An advantage of these techniques is that they do not rely on the use of external devices, and they are feasible also to people who underwent amputation. Unfortunately, the extraction of complex features from raw HD EMG signals may introduce delays that deter the real-time requirements of the system. To address this issue, in a preliminary investigation we proposed to use graph neural networks for gesture recognition from raw HD EMG data. In this paper, we extend our previous work by exploiting Explainable AI algorithms to automatically refine the graph topology based on the data in order to improve recognition rates and reduce the computational cost. We performed extensive experiments with a large dataset collected from 20 volunteers regarding the execution of 65 fine-grained gestures, comparing our technique with state-of-the-art methods based on handcrafted features and different machine learning algorithms. Experimental results show that our technique outperforms the state of the art in terms of recognition performance while incurring significantly lower computational cost at run-time.**

***Index Terms*—Pervasive healthcare; gesture recognition; prosthetic arm control; HD EMG sensor data; GNN; deep learning.**

I. Introduction

The use of EMG sensors worn on the forearm has several advantages with respect to other gesture recognition technologies. As regards cameras and depth sensor methods [1], EMG sensors are not susceptible to external factors like ambient light, and they do not require a direct line of sight to the hand. Differently from smart gloves, EMG sensors do not interfere with hand movements. Moreover, EMG sensors typically have low latency, enabling real-time gesture recognition. This characteristic is crucial in applications that require real-time interactions, such as virtual reality gaming or robotic control. However, data acquired from standard EMG electrodes may be insufficient to recognize gestures at a fine-grained level. Indeed, most surface EMG readers include a limited number of electrodes; hence, they provide limited spatio-temporal information [2]. Li et al. state that gestures that include a large number of degrees of freedom are more easily recognized increasing the number of channels for muscular data acquisition [3]. Smith et al. [4] observed in their research that spatial and temporal information (i.e., the number of electrode channels used, the length of the analysis window, and the degree of window overlap) are directly related. Hence, in order to enhance the recognition of fine-grained gestures, a few researchers investigated the use of high-density EMG (HD EMG) electrodes [5]. The electrodes of these devices are arranged in a dense two-dimensional array and can acquire extensive spatio-temporal muscular data [6].

Most existing works for EMG-based gesture intention recognition rely on the application of supervised machine learning algorithms to statistical features extracted from muscular signals [7]. To improve recognition rates, several features are extracted from raw signals in the time, frequency, and time-frequency domains. Of course, the extraction of several complex features at run-time introduces delays that may deter the real-time response of the recognition algorithms, negatively impacting the usability of the system [8]. This problem is particularly evident in HD EMG-based solutions, given the large number of channels from which statistical features are extracted. Indeed, the time taken by feature extraction and classification must not exceed the length of the sliding window used for processing the HD EMG signals [9]. On the other hand, longer sliding windows may negatively impact the response time of the system for real-time applications.

In order to address this challenging issue, we propose the use of graph neural networks (GNNs) for recognizing gesture intentions from raw signals. GNNs are deep learning architectures composed of several propagation modules. Relational spatio-temporal information is propagated between nodes of the GNN, capturing both feature-based and topological information [10]. This solution has different advantages. Indeed, the GNN may be fed with raw signal data, without the need of extracting features that may introduce delays in the recognition process. Moreover, GNNs are an appealing solution for handling HD EMG data, since they proved to be effective in classification tasks where abundant information with strong spatio-temporal correlations is available [11]. Previous studies investigated different machine learning methods to exploit HD EMG data for gesture recognition [12]. However, to the best of our knowledge, no previous study, except our preliminary investigation reported in [13], used a graph neural network (GNN) in conjunction with HD EMG signals to identify the gesture intention of an amputee.

Our preliminary investigation showed the potential of applying GNNs to HD EMG signals for gesture recognition, obtaining promising results. In particular, we performed an experiment about the recognition of 65 gestures executed by 20

Silvia Maria Massa and Daniele Riboni are with the Dept. of Mathematics and Computer Science, University of Cagliari, Via Ospedale 72, 09124 Cagliari, Italy (e-mail: silviam.massa@unica.it, riboni@unica.it).

Kianoush Nazarpour is with the School of Informatics, University of Edinburgh, EH8 9AB Edinburgh, U.K (e-mail: kianoush.nazarpour@ed.ac.uk).

volunteers wearing two HD EMG electrodes on the forearm. We obtained an average classification error rate of 8.75% with a standard deviation of 4.92. However, 20 gestures out of 65 were detected with an average error rate larger than 10%, where the latter is considered a threshold value for the practical usability of upper limb prostheses [14]. Moreover, the computational cost should be reduced to improve the real-time performance of the system. A standard method for reducing the computational cost and improving recognition rates in machine learning systems is feature selection [15]. However, in our case, GNN nodes are not associated with feature vectors. Instead, each node is associated with a vector containing raw data regarding the electrical activity of the muscle. Therefore, feature selection is not applicable in our case.

For this reason, in this paper, we extend our previous work with an eXplainable AI (xAI)-based technique to improve recognition rates and reduce the computational cost of GNN training and classification. Our objective is to refine the GNN topology, keeping only those edges (i.e., logical connections among nodes corresponding to EMG data channels) having high prediction capabilities. The advantage of our approach is twofold. On the one hand, we can increase recognition rates and reduce overfitting by pruning those edges that do not provide useful information for the prediction task. On the other hand, a smaller graph structure can reduce training and classification times. Given the high computational cost of GNN training, a reduction of training time is fundamental for extensively experimenting and refining the algorithm. Moreover, a reduced graph topology corresponds to a smaller model, which determines clear advantages in terms of computational cost and power consumption.

Specifically, we use the GNN explainer algorithm presented in [16]. Indeed, to the best of our knowledge, it is the only deep learning explainer that exploits the relational information provided by the graph. In fact, relational information plays a crucial role in the prediction capabilities of GNNs, in addition to the characteristics of the nodes. That algorithm provides a log probability for each edge, which reflects its contribution to the classification outcome. Hence, we keep only those edges that contribute most to the classification task. We performed several experiments to compare our method with the state of the art and to evaluate the impact of our xAI-based technique considering different thresholds for edge pruning. With the used dataset, our technique clearly outperformed state-of-the-art methods based on handcrafted features and different machine learning algorithms in terms of both recognition performance and run-time computational costs. The best results of the xAI-base technique were achieved by keeping the 50% edges that give more contribution. With this setup, we were able to reduce computational times by 53%. Moreover, the number of gestures that were detected with an average error rate larger than 10% was reduced by 60%.

Summarizing, the main contributions of our work with respect to the preliminary investigation presented in [13] are the following:

- In order to reduce delays that may deter the real-time requirements of the system, we introduce a novel xAI-based technique to improve the topology of the GNN's graph;
- We experimentally evaluate our xAI-based technique with a large real-world dataset, obtaining a reduction of computational times and an improvement of recognition rates;
- We experimentally compare our methods with state-of-the-art techniques based on handcrafted features and different machine learning algorithms, showing that our methods outperform the state of the art in terms of both recognition performance and run-time computational cost.

The rest of the paper is structured as follows. Section II describes the dataset used in our work and our methods. Section III presents our experimental evaluation. Section IV discusses the results and the limitations of our work. Section V concludes the paper.

II. Material and Methods

In this section, we report materials and methods used in our work.

A. HD EMG Dataset

We have evaluated our methods with a large real-world dataset [17] of HD EMG data that was recently released[1]. The dataset is public for research; hence, it can be used to replicate our experiments or for experimentally comparing other techniques. The data collection was performed by 20 able-bodied volunteers (14 men and 6 women) aged between 25 and 57 (mean age 35). The volunteers wore two HD EMG devices on the forearm while executing 65 gestures different fine-grained gestures. Figure 1 shows the device and its placement on the arm. Each participant repeated each gesture five times with a rest period of 5 seconds between each repetition. Table I describes the gestures and their complexity through degrees of freedom (DoF) [18]. The higher the DoF, the more complex the gesture.

Muscular data during the execution of gestures was acquired by two HD EMG surface electrodes. Each device was instrumented with 64 channels arranged in an 8×8 matrix with an inter-electrode spacing of 10 mm. Each channel provides a continuous stream of EMG readings sampled at 2048 Hz. Specific techniques were adopted to reduce noise in acquired EMG signals. Additional details about the device placement and data acquisition methodology can be found in [17].

B. Graph-based modeling of HD EMG data

In order to exploit GNN reasoning, it is necessary to model the relevant information as a graph structure. In our model, we create a node for each HD EMD channel providing muscular data. In GNN frameworks, each node is associated with a feature vector. In our case, we extract the feature vector from the data stream of the node's channel using a sliding window approach. The window's duration is 32 ms, and windows are non-overlapping. Each window contains 65 scaled data samples (i.e., the value distribution is scaled to obtain mean 0 and standard deviation 1).

[1] https://tinyurl.com/ycxtrtym

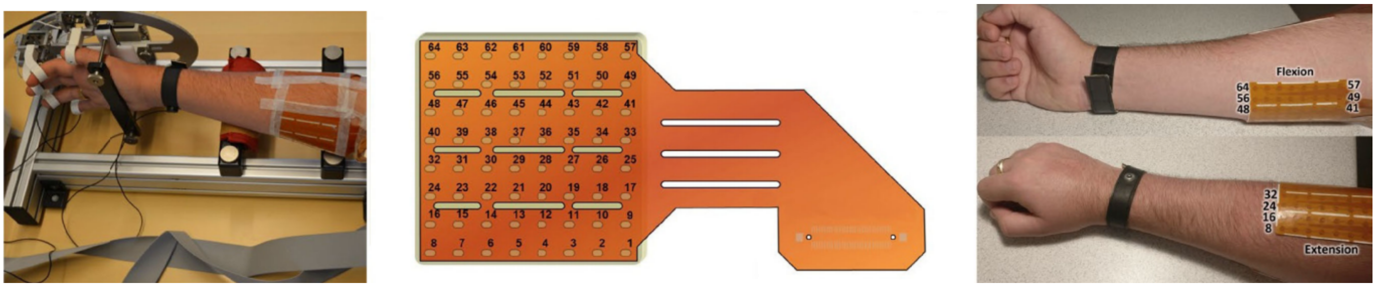


Fig. 1. HD EMG electrodes and their placement for the dataset [17]

TABLE I
AN EXCERPT OF THE 65 CLASSIFIED GESTURES WITH THEIR RESPECTIVE CLASS LABEL, THEIR DESCRIPTION, AND THEIR COMPLEXITY EXPRESSED IN DOF. THE FULL LIST CAN BE FOUND IN [17].

Class	**Gesture**	**DoF**
1	Little finger: bend	1
2	Little finger: stretch	
3	Ring finger: bend	
4	Ring finger: stretch	
5	Middle finger: bend	
…	…	
15	Wrist: rotate anti-clockwise	
16	Wrist: rotate clockwise	
17	Little finger: bend + Ring finger: bend	2
18	Little finger: bend + Thumb: down	
19	Little finger: bend + Thumb: left	
20	Little finger: bend + Thumb: right	
21	Little finger: bend + Wrist: bend	
…	…	
56	Wrist: stretch + Wrist: rotate anti-clockwise	
57	Wrist: stretch + Wrist: rotate clockwise	
58	Extend all fingers (without thumb)	≥3
59	All fingers: bend (without thumb)	
60	Palmar grasp	
61	Wrist: rotate anti-clockwise with the Palmar grasp	
62	Pointing (index: stretch, all other: bend)	
63	3-digit pinch	
64	3-digit pinch with Wrist: anti-clockwise rotation	
65	Key grasp with Wrist: anti-clockwise rotation	

While the choice of representing each channel as a node is straightforward, deciding which couples of nodes should be connected by an edge is challenging and strongly impacts the performance of the GNN [19]. On the one hand, the presence of edges that do not convey information useful for the classification may confuse the classifier and determine overfitting. On the other hand, the lack of edges with high discriminative power may negatively impact the recognition performance. For this reason, we adopt a two-step procedure.

- At first, we connect each node to all its surrounding nodes in the electrode channels matrix. The resulting network topology is illustrated in Figure 2. Each internal node is connected to other 8 nodes; each border node is connected to 5 other nodes; each corner node is connected to 3 other nodes. This setup is intended to replicate the spatial characteristics of the electrodes' placement. In order to simultaneously consider the data acquired from the two electrodes, we added edges from the nodes of the first electrode's last row to the nearest nodes of the second electrode's first row.
- Then, we refine the graph topology based on an xAI algorithm in order to prune edges with low discriminative power for the classification task. Our xAI-based solution is explained in Section II-D.

C. EMG-GNN structure

Our GNN structure is shown in Fig. 3, and it consists of:

- graph convolutional layers and ReLU non-linearity applied to the signals mapped onto the graph structure to embed each node by performing multiple rounds of message passing;
- a READOUT function to learn the representation vector of the entire graph through the aggregation of the node representations from the final graph convolutional layer;
- a multi-layer perceptron (MLP) to classify the graph representation vector.

SAGEConv implements the GraphSAGE operator proposed in [20]. GraphSAGE is a general inductive framework that leverages node feature information to efficiently generate node embeddings for previously unseen data. This framework is designed for large graphs with a high number of nodes. GraphSAGE learns a function that generates embeddings by sampling and aggregating the local neighborhood features of a node, unlike most existing approaches that require all nodes in the graph to be considered during embedding training.

We used the Adam Optimizer with a starting Learning Rate (LR) of 0.001. We also used ReduceLROnPlateau, which reduces the LR when a metric has stopped improving for a

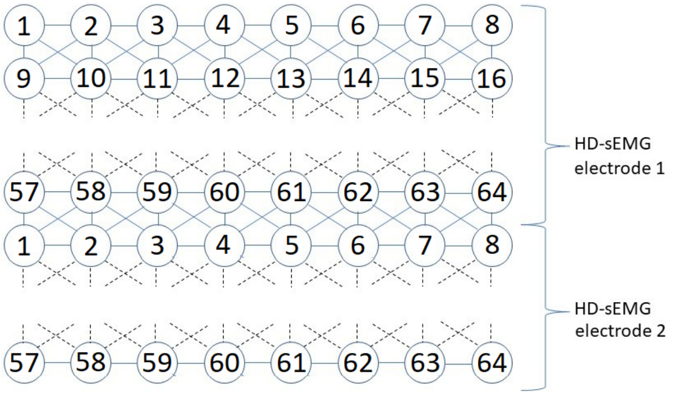


Fig. 2. The graph consists of 128 nodes and 884 edges. The structure of its channels is analogous to the organization of pixels in an image.

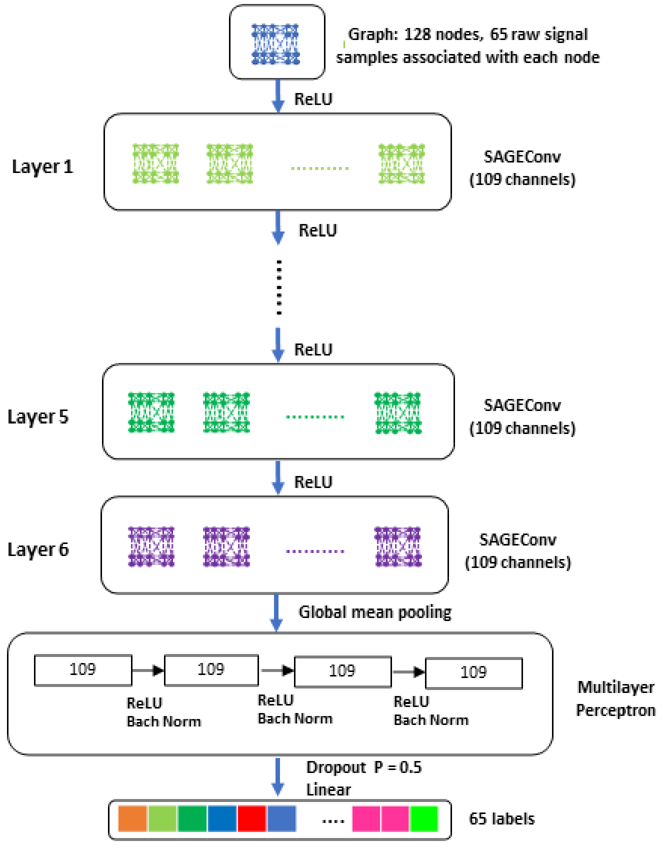


Fig. 3. Schema of the EMG-GNN structure

"patience" number of epochs. In our case, we monitor the Validation Loss, and if its value does not decrease for 10 epochs, the learning rate is reduced by 0.1.

We applied Cross-Entropy Loss and monitored the Validation Loss to decide when to stop training. Early Stopping allows us to speed up learning and avoid overfitting. If the Validation Loss value does not decrease for 30 epochs, model training is stopped; otherwise, the model is trained until the 100th epoch is executed. We batched the graphs, setting the size to 32, before putting them into the GNN to ensure full GPU utilization.

D. xAI-based graph topology refinement

In order to refine the graph topology, we need to identify which edges provide more utility for the gesture classification task. To solve this problem, we rely on a recently proposed algorithm (named GNN Explainer) for the explainability of GNN, presented in [16]. The GNN Explainer algorithm provides interpretable explanations of the predictions of any GNN-based model in any graph-based deep learning task, such as node classification, link prediction, and graph classification. Its aim is to maximize the mutual information between the prediction of the model trained with full graphs and the prediction of the explanatory model trained with simplified graphs. Given an instance, it identifies a compact subgraph structure and a small subset of node features having a crucial role in the GNN prediction [16].

More precisely, for each edge, GNN Explainer computes a log probability reflecting its contribution to the classification result. Considering the log probability edges, we simplify the graph topology keeping only those edges that contribute most to the classification. Our technique is flexible since we can decide the number $\mathcal{E}$ of edges to be kept. In our experimental evaluation, reported in Section III, we evaluate the impact of $\mathcal{E}$ on the performance of our system.

In our experiments, we used the Pytorch Geometric (PyG) library's class `GNNExplainer` that implements the GNNExplainer algorithm. In particular, we employed the method `explainer.explain_graph`, since the task of our model is graph classification. In `GNNExplainer`, we set the number of epochs to 100 and the LR to 0.001.

III. Experimental evaluation

In this section, we illustrate the experimental evaluation of our technique. After explaining the experimental setup (Section III-A), we illustrate the results achieved with a state-of-the-art method based on handcrafted features and supervised machine learning (Section III-B). Then (Section III-C), we report the results obtained with the full (i.e., non-refined) graph topology and raw signal data. Finally, in Section III-D we report the results obtained with our xAI-based graph refinement method.

A. Experimental setup

We conducted all the experiments on a workstation with AMD Ryzen 5 5600X 6-Core 3.70 GHz Processor, 32 GB RAM, and NVIDIA GeForce RTX 2060 GPU. All the algorithms have been developed in Python. For each subject, we randomly divided the dataset into 80% instances (i.e., sliding windows data) for the training set and the remaining 20% instances for the test set, and we evaluated each subject's dataset separately. Since we needed to fine-tune the hyperparameters of the GNN, in the GNN experiments we further divided the training set into 75% instances for training and 25% instances for validation. We used the same test instances in all the experiments.

B. Machine learning & handcrafted features

In order to compare our method with the state of the art, we implemented a baseline technique for recognizing gesture intentions based on muscular signals. According to the recent literature, most state-of-the-art solutions in this field rely on the extraction of handcrafted features from raw signals, and on classical machine learning algorithms for classification. In order to choose which features to extract, we relied on the review reported in [21]. We selected the most widely adopted features in the temporal and frequency domains. Given $\langle x_1, \ldots, x_N \rangle$, which is the temporal sequence of raw signals acquired from a channel during an N-length sliding window, we extract the following features:

- Waveform length $WL = \sum_{n=1}^{N-1} |x_{n+1} - x_n|$, which is the cumulative length of the waveform over the temporal window. This feature proved to be the most effective in different studies in this field [22].

- Root mean square $RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x_n^2}$, which is frequently used in conjunction with WL.
- Power spectral density (PSD) coefficients, which are standard frequency-domain features used in several signal processing applications [23]. We compute those features using the Welchs method based on the fast Fourier transform [24]. Since we consider sliding windows having 65 signal samples each, the Welchs method extracts 33 PSD coefficients.

Since we acquire muscular signals from 128 channels and we extract 35 features per channel, in total we extract 4,480 features from each sliding window's data. In our experiments, we evaluated five of the machine learning algorithms that are most frequently adopted for this task [21], i.e.: Linear Discriminant Analysis (LDA), Support Vector Machines (SVM), k-Nearest Neighbour (kNN), Linear Regression (LR), and Random Forests (RF).

Experimental results of the different machine learning algorithms with handcrafted features are reported in the first five rows of Table II. In our case, the feature extraction process is independent by the adopted machine learning algorithm. On average, the feature extraction process takes 62.878 ms for extracting a sliding window's features. Unfortunately, the feature extraction time largely exceeds the sliding windows duration (i.e., 32 ms in our setup), making the handcrafted features approach unfeasible for real-time applications. As anticipated before, this result is determined by the large number of channels available in HD EMG devices.

Among the evaluated algorithms, the one achieving the best results in terms of accuracy, precision, recall, and F_1 score is LR, which is also the algorithm with shortest classification time. Indeed, with LR, the feature vector extracted from a sliding window is classified in 0.027 ms on average. LDA, RF, and SVM achieve similar classification results. However, while LDA and RF provide very fast classification times, SVM classification is significantly slower. The algorithm achieving the worst classification results is kNN, which provides relatively fast classification times with this setup. LR is also the algorithm providing the best results in terms of the number of gestures with low recognition performance (i.e., 34 gestures out of 65 with an error rate greater than 10%). Indeed, a prosthetic system to be usable should have an error rate smaller than 10% [14].

Figures 4a and 4b show the confusion matrices of the two machine learning algorithms achieving the best performance. As can be observed, the classifiers tend to confuse neighboring gestures. The reason is due to the ordering of gestures in the dataset (see Table I). According to the ordering, consecutive gestures are often very similar to each other. For instance, gestures identified with numbers from 37 to 40 involve the movement of the middle finger and wrist.

C. Full graph GNN & raw signals

Results achieved by the Full graph GNN & raw signals are reported in the last row of Table II. This method outperforms the Machine learning & handcrafted features approach in most regards. In terms of computational time, the GNN takes in input the raw HD EMG signals; hence, it does not consume time to extract features from raw data. The average classification time for a sliding window's instance is very fast (i.e., 2.224 ms). Consequently, the execution of the GNN approach is feasible for real-time applications even considering short-length time windows as in this work.

In terms of classification results, the GNN technique clearly outperforms the Machine learning & handcrafted features approach. Indeed, the GNN technique achieves accuracy, precision, recall, and F_1 score values larger than 0.91, improving by 0.09 the best results achieved by the other methods. The GNN technique also provides the best results in terms of the number of gestures with low recognition performance (i.e., 20 gestures vs $\geq$ 34 gestures obtained by the other methods). Among the 20 poorly recognized gestures, only one gesture has DoF=1, while sixteen gestures have DoF=2, and three gestures have DoF=3. As expected, the larger the DoF of a gesture, the more difficult its recognition.

Figure 4c shows the full graph GNN confusion matrix. Also in this case, the classifier tends to confuse neighboring classes. For instance, class 59 in which all fingers are bent (without the thumb) is often confused with class 60 in which a palmar grip is performed; and vice-versa.

D. Refined GNN graph topology & raw signals

We have performed experiments to evaluate our xAI-based pruning method presented in Section II-D, varying the number $\mathcal{E}$ of edges to be kept. For deciding which edges to prune, we used the validation set of the first subject's dataset, consisting of 9745 graphs, in which the classes were equally distributed. For each of these graphs, the explainer returns the log probability for each edge and node feature. The main purpose of this study is to simplify the structure of the graph. For this reason, we analyzed the results regarding edges. Furthermore, since node features are equivalent to raw signal sections, the results concerning node features are not usable.

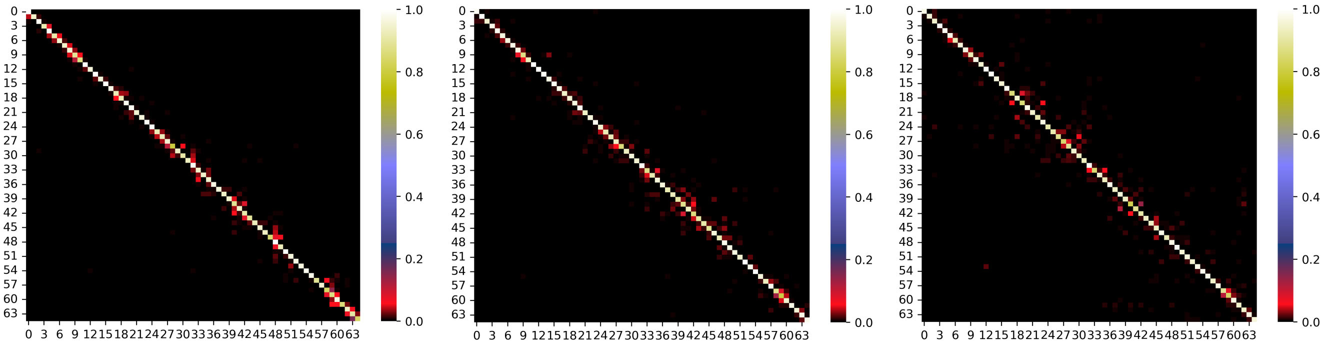
During the analysis of the results returned by the explainer, for each edge we computed a single representative value of its importance in the predictions of the different classes. That value is computed as the average of the log probabilities obtained for each edge by testing the different graphs. Then, the edges are sorted according to the values obtained in ascending order. Once we sorted the edges by importance, we trained, validated, and tested the model as illustrated in Section III-C, using graphs containing different number $\mathcal{E}$ of edges ranging from 884 to 55.

Results are shown in Table III. In terms of accuracy, precision, recall, and F_1 score, the best results are achieved with $\mathcal{E} = 663$ and $\mathcal{E} = 442$, where the value of those metrics is around 0.95. Inspecting the trend of recognition performance metrics, we can observe that the performance tends to improve when pruning nodes until a 50% rate. By pruning more nodes,

TABLE II

MACHINE LEARNING & HANDCRAFTED FEATURES VS FULL GRAPH GNN & RAW SIGNALS. BEST RESULTS AMONG MACHINE LEARNING & HANDCRAFTED FEATURES TECHNIQUES ONLY ARE REPORTED IN *italic*. OVERALL BEST RESULTS ARE REPORTED IN **BOLD**.

Method	**Acc.**	**Prec.**	**Rec.**	**F_1 score**	**Feat. extract. time (ms)**	**Classific. time (ms)**	**Total time**	**Gestures with error rate > 10%**
LDA & handcrafted features	0.810	0.813	0.809	0.809	*62.878*	0.030	62.908	36
SVM & handcrafted features	0.802	0.807	0.801	0.803	*62.878*	90.702	153.580	43
kNN & handcrafted features	0.796	0.797	0.796	0.795	*62.878*	3.280	66.158	44
LR & handcrafted features	*0.820*	*0.820*	*0.820*	*0.820*	*62.878*	***0.027***	*62.905*	*34*
RF & handcrafted features	0.805	0.806	0.804	0.804	*62.878*	0.090	62.968	38
Full graph GNN & raw signals	**0.912**	**0.913**	**0.912**	**0.912**	**0**	2.224	**2.224**	**20**



(a) LDA confusion matrix

(b) LR confusion matrix

(c) Full graph GNN confusion matrix

Fig. 4. Confusion matrices of machine learning & handcrafted features

TABLE III

REFINED GNN GRAPH TOPOLOGY & RAW SIGNALS. THE BEST RESULTS ARE REPORTED IN **BOLD**. $\mathcal{E}$ IS THE NUMBER OF EDGES KEPT IN THE GNN GRAPH FOR THE CORRESPONDING EXPERIMENT.

Method	**Acc.**	**Prec.**	**Rec.**	**F_1 score**	**Feat. extract. time (ms)**	**Classific. time (ms)**	**Total time**	**Gestures with error rate > 10%**
Refined GNN ($\mathcal{E}$ = 884)	0.941	0.942	0.941	0.941	**0**	2.560	2.560	10
Refined GNN ($\mathcal{E}$ = 663)	**0.951**	**0.951**	**0.950**	**0.950**	**0**	1.819	1.819	8
Refined GNN ($\mathcal{E}$ = 552)	0.946	0.946	0.946	0.946	**0**	1.505	1.505	6
Refined GNN ($\mathcal{E}$ = 442)	**0.951**	0.950	**0.950**	**0.950**	**0**	1.213	1.213	**4**
Refined GNN ($\mathcal{E}$ = 332)	0.946	0.946	0.945	0.945	**0**	1.112	1.112	6
Refined GNN ($\mathcal{E}$ = 221)	0.936	0.936	0.935	0.935	**0**	0.910	0.910	15
Refined GNN ($\mathcal{E}$ = 110)	0.915	0.915	0.914	0.914	**0**	0.809	0.809	22
Refined GNN ($\mathcal{E}$ = 55)	0.906	0.906	0.905	0.905	**0**	**0.707**	**0.707**	24

the recognition performance starts decreasing. With $\mathcal{E} = 55$, the values of those metrics is below 0.91. The confusion matrix obtained using $\mathcal{E}$ = 442 is shown in Figure 5. It can be observed that most errors are concentrated on neighboring gestures, which are rather similar, while errors among more different gestures (i.e., red points far from the diagonal of the matrix) are uncommon.

Considering the number of gestures with low recognition performance (i.e., gestures with error rates > 10%), the best result is achieved with $\mathcal{E} = 442$. With this value, only 4 gestures are poorly recognized: two of them have DoF=2 (i.e., 'Ring finger: bend + Thumb: down' and 'Index finger: bend + Thumb: left'), while the remaining ones have DoF=3 (i.e., 'Extend all fingers (without thumb)', and 'All fingers: bend (without thumb)'). Also in this case, there is a prevalence of difficulties in recognizing gestures with large DoF values. With smaller or larger values of $\mathcal{E}$, the number of poorly recognized gestures increases; i.e., 6 gestures with $\mathcal{E} = 552$ or $\mathcal{E} = 332$, and 8 gestures with $\mathcal{E} = 663$. The largest number of poorly recognized gesture is achieved with $\mathcal{E} = 55$. With that

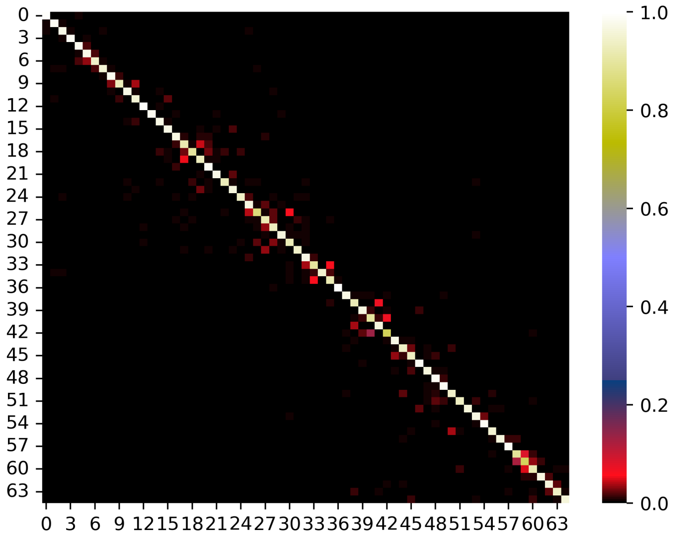


Fig. 5. Confusion matrix of Refined GNN ($\mathcal{E}$ = 442)

TABLE IV
ALGORITHMS, SLIDING-WINDOW SIZE, STEP SIZE, AND ACCURACY ACHIEVED BY DEEP LEARNING APPROACHES USING THE HD EMG DATASET [17]

Model	Sliding-window size (ms)	Step size (ms)	Acc. (%)	Reference paper
MLP	200	**10**	49.87	[25]
CNN	200	**10**	59.56	[25]
LSTM, Heterogeneous Dilation, Middle Focus	100	**10**	80.40	[25]
LSTM, Heterogeneous Dilation, Middle Focus	200	**10**	83.30	[25]
LSTM, Heterogeneous Dilation, Middle Focus	300	**10**	82.40	[25]
ViT-HGR, MLP size 384, Embedding Dimension 192	**31.25**	15.62	84.62	[26]
ViT-HGR, MLP size 96, Embedding Dimension 96	**31.25**	15.62	84.24	[26]
ViT-HGR, MLP size 48, Embedding Dimension 48	**31.25**	15.62	83.46	[26]
Stand-alone Macro Model	250	125	89.34	[27]
Stand-alone Micro Model	250	125	86.64	[27]
HYDRA-HGR	250	125	**94.86**	[27]

value, 24 gestures are poorly recognized: one gesture with DoF=1, twenty-one gestures with DoF=2, and two gestures with DoF=3. Based on these results, we can conclude that the best recognition results are achieved with $\mathcal{E} = 442$.

Considering the computational cost, we notice that the classification time tends to linearly grow with the number $\mathcal{E}$ of kept edges. As expected, the lowest classification time (i.e., 0.707 ms) is achieved with the smallest value of $\mathcal{E}$. Using the value that achieved the best recognition results (i.e., $\mathcal{E} = 442$), the computational time is 1.213 ms, while the computational cost using the full graph topology is 2.560 ms. Hence, by pruning 50% of edges, our technique can halve the computational cost while significantly improving the recognition performance.

IV. Discussion and limitations

In order to enhance the comprehensiveness of our analysis, we performed a thorough bibliographic research to find all the research works which apply neural network-based methods to the same dataset used in our work. We queried different scientific search engines: Web of Science, ScienceDirect, PubMed, Google Scholar, and SCOPUS, to retrieve all the papers that cited the reference paper of the dataset [17]. Among the 19 retrieved papers, eight papers reported an experimental evaluation with that dataset, and three of them used a neural network-based method; i.e.: [25]–[27]. Table IV reports the algorithms, sliding-window size, step size, and accuracy achieved by the neural network-based approaches. From the comparison, we can observe that our method outperforms most of the other deep learning approaches in terms of accuracy. The only previous work achieving an accuracy value close to the one of our method is the HYDRA-HGR technique presented in [27]. However, in the reported experiments, HYDRA-HGR used a step size of 125 ms, meaning that the current gesture is recognized at 8 Hz. Instead, in our experiments, we used a step size of 32 ms, which means that the current gesture is recognized at about 32 Hz. Of course, a higher frequency of gesture recognition enables the system to detect and respond to gestures more quickly. This results in a more immediate and seamless interaction with the user, providing a smoother and more natural user experience, especially when fine-grained gestures are considered.

While our method can be applied to robotic arm control, we point out that the application of our system to amputees may be problematic due to the reduced residual muscle capacity from disabled people. Hence, in order to evaluate the feasibility of our method for robotic prosthesis control systems, our results should be confirmed through experiments with additional datasets acquired from amputees. Unfortunately, at the time of writing, to the best of our knowledge there is no public dataset of HD EMG signals acquired from amputees during the execution of hand gestures. Indeed, the scarcity of well-maintained public benchmarks is a serious problem in this research field [28].

Our technique strongly relies on the number $\mathcal{E}$ of edges to be kept in the GNN graph. Hence, the choice of $\mathcal{E}$ is critical to the performance of our method. The best choice of $\mathcal{E}$ depends on the data acquisition device. For instance, using an HD EMG device with a larger number of channels, we expect to obtain the best results by pruning a larger number of edges. The number and kind of edges to be pruned may also be influenced by the kind of gestures to be recognized. Hence, the most appropriate value of $\mathcal{E}$ should be experimentally chosen.

While we believe that this work provides an important contribution to HD EMG-based gesture recognition, several challenges remain open for future research. In our experimental evaluation, we noticed that the majority of gestures achieving low recognition rates were associated to a DoF larger than one. However, some simple gestures with DoF equal to one, such as ‘thumb up’, had an unexpectedly low recognition rate. We conjecture that this result may be due to the lack of information about some muscular activity strongly related to those gestures. This problem may be addressed by using a more comprehensive HD EMG device capable of acquiring additional muscular signals. Future work also includes the investigation of multi-modal approaches integrating other kinds of sensors (e.g., inertial sensors) to improve the recognition rates.

Finally, an inherent limitation of EMG-based solutions is that they cannot give neural feedback to the individual, negatively impacting the user experience of amputees. To address this issue, hybrid solutions using also electroneurogram (ENG) devices may be investigated. Indeed, ENG instruments may provide tactile feedback to the amputee, allowing to mitigate the symptoms of phantom limb syndrome and restoring the sense of touch [29].

V. Conclusion

In this paper, we tackled the challenging issue of recognizing gesture intentions based on HD EMG data. We presented a technique based on GNN for recognizing gestures at a fine-grained level, and a xAI-based algorithm for refining the GNN graph topology. Extensive experiments with a large real-world dataset showed that our xAI-based method improves the system accuracy while reducing the computational cost. Comparison with different machine learning methods based on handcrafted features and deep learning approaches show that our methods outperform the state of the art.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions for improving the technical content and presentation of this paper. We acknowledge financial support under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.5 - Call for tender No.3277 published on December 30, 2021 by the Italian Ministry of University and Research (MUR) funded by the European Union NextGenerationEU. Project Code ECS0000038 Project Title eINS Ecosystem of Innovation for Next Generation Sardinia CUP F53C22000430001-Grant Assignment Decree No. 1056 adopted on June 23, 2022 by the Italian Ministry of University and Research (MUR).

References

[1] Z. Yu, B. Zhou, J. Wan, P. Wang, H. Chen, X. Liu, S. Z. Li, and G. Zhao, "Searching multi-rate and multi-modal temporal enhanced networks for gesture recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 5626–5640, 2021.

[2] R. Menon, G. Di Caterina, H. Lakany, L. Petropoulakis, B. A. Conway, and J. J. Soraghan, "Study on interaction between temporal and spatial information in classification of emg signals for myoelectric prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1832–1842, 2017.

[3] G. Li, A. E. Schultz, and T. A. Kuiken, "Quantifying pattern recognitionbased myoelectric control of multifunctional transradial prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 2, pp. 185–192, 2010.

[4] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the optimal window length for pattern recognition-based myoelectric control: balancing the competing effects of classification error and controller delay," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 2, pp. 186–192, 2010.

[5] A. Stango, F. Negro, and D. Farina, "Spatial correlation of high density emg signals provides features robust to electrode number and shift in pattern recognition for myocontrol," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 189–198, 2014.

[6] G. Drost, D. F. Stegeman, B. G. van Engelen, and M. J. Zwarts, "Clinical applications of high-density surface emg: a systematic review," *Journal of Electromyography and Kinesiology*, vol. 16, no. 6, pp. 586–602, 2006.

[7] A. Asghar, S. Jawaid Khan, F. Azim, C. S. Shakeel, A. Hussain, and I. K. Niazi, "Review on electromyography based intention for upper limb control using pattern recognition for human-machine interaction," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 236, no. 5, pp. 628–645, 2022.

[8] L. Bi, C. Guan *et al.*, "A review on emg-based motor intention prediction of continuous human upper limb motion for human-robot collaboration," *Biomedical Signal Processing and Control*, vol. 51, pp. 113–127, 2019.

[9] N. Parajuli, N. Sreenivasan, P. Bifulco, M. Cesarelli, S. Savino, V. Niola, D. Esposito, T. J. Hamilton, G. R. Naik, U. Gunawardana *et al.*, "Real-time emg based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation," *Sensors*, vol. 19, no. 20, p. 4596, 2019.

[10] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[11] S. Wein, A. Schüller, A. M. Tomé, W. M. Malloni, M. W. Greenlee, and E. W. Lang, "Forecasting brain activity based on models of spatio-temporal brain dynamics: A comparison of graph neural network architectures," *Network Neuroscience*, pp. 1–61, 2021.

[12] R. N. Khushaba and K. Nazarpour, "Decoding hd-emg signals for myoelectric control-how small can the analysis window size be?" *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8569–8574, 2021.

[13] S. M. Massa, D. Riboni, and K. Nazarpour, "Graph neural networks for hd emg-based movement intention recognition: An initial investigation," in *2022 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*. IEEE, 2022, pp. 1–4.

[14] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: state of the art and challenges for clinical use." *Journal of Rehabilitation Research & Development*, vol. 48, no. 6, 2011.

[15] M. M. Manca, B. Pes, and D. Riboni, "Exploiting feature selection in human activity recognition: Methodological insights and empirical results using mobile sensor data," *IEEE Access*, vol. 10, pp. 64 043–64 058, 2022.

[16] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "Gnnexplainer: Generating explanations for graph neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[17] N. Malešević, A. Olsson, P. Sager, E. Andersson, C. Cipriani, M. Controzzi, A. Björkman, and C. Antfolk, "A database of high-density surface electromyogram signals comprising 65 isometric hand gestures," *Scientific Data*, vol. 8, no. 1, pp. 1–10, 2021.

[18] A. Fougner, Ø. Stavdahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of upper limb prostheses: Terminology and proportional myoelectric controla review," *IEEE Transactions on neural systems and rehabilitation engineering*, vol. 20, no. 5, pp. 663–677, 2012.

[19] A. Demir, T. Koike-Akino, Y. Wang, M. Haruna, and D. Erdogmus, "Eeg-gnn: Graph neural networks for classification of electroencephalogram (eeg) signals," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2021, pp. 1061–1067.

[20] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[21] A. Sultana, F. Ahmed, and M. S. Alam, "A systematic review on surface electromyography-based classification system for identifying hand and finger movements," *Healthcare Analytics*, p. 100126, 2022.

[22] A. Phinyomark, S. Hirunviriya, C. Limsakul, and P. Phukpattaranont, "Evaluation of emg feature extraction for hand movement recognition based on euclidean distance and standard deviation," in *ECTI-CON2010*. IEEE, 2010, pp. 856–860.

[23] R. M. Howard, *Principles of random signal analysis and low noise design: The power spectral density and its applications*. John Wiley & Sons, 2004.

[24] P. Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on audio and electroacoustics*, vol. 15, no. 2, pp. 70–73, 1967.

[25] T. Sun, Q. Hu, J. Libby, and S. F. Atashzar, "Deep heterogeneous dilation of lstm for transient-phase gesture prediction through high-density electromyography: Towards application in neurorobotics," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2851–2858, 2022.

[26] M. Montazerin, S. Zabihi, E. Rahimian, A. Mohammadi, and F. Naderkhani, "Vit-hgr: Vision transformer-based hand gesture recognition from high density surface emg signals," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2022, pp. 5115–5119.

[27] M. Montazerin, E. Rahimian, F. Naderkhani, S. F. Atashzar, H. Alinejad-Rokny, and A. Mohammadi, "Hydra-hgr: A hybrid transformer-based architecture for fusion of macroscopic and microscopic neural drive information," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2023, pp. 1–5.

[28] P. Gopal, A. Gesta, and A. Mohebbi, "A systematic study on electromyography-based hand gesture recognition for assistive robots using deep learning and machine learning models," *Sensors*, vol. 22, no. 10, p. 3650, 2022.

[29] S. Micera, L. Citi, and et al., "Decoding information from neural signals recorded using intraneural electrodes: Toward the development of a neurocontrolled hand prosthesis," *Proceedings of the IEEE*, vol. 98, no. 3, pp. 407–417, 2010.