University of Padua

Department of Information Engineering
Master degree in Bioengineering

EEG and EMG graph CNN for upper limb movement classification

Supervisor
Prof. Manfredo Atzori

Candidate
Alessio Palatella

Co-Supervisor
Dr. Stefano Tortora

November 28, 2022

Academic Year 2021-2022

Abstract

Neural Human-Machine Interfaces (HMI) are technologies that allow a human being to communicate his/her intention to external devices – such as computers, robots and prostheses – through the acquisition of signals from the central nervous system. To overcome the limitations of interfaces based on unimodal neural signals, the research community recently introduced the concept of hybrid neural interfaces based on the combination of brain and muscular signals. However, state-of-the-art approaches still relies on simple additive fusion techniques or heuristic decision rules, limiting the performance in complex decoding tasks.

To go beyond the state-of-the-art, this thesis proposes to model the connectivity in the human neuromuscular system through a weighted undirected graph: the nodes of the graph represent the electroencephalography (EEG) and electromyography (EMG) channels, and the weight of the connections represents signals' correlation. Then, a graph convolutional neural network (GCN) with learning structure was defined and implemented to let the model learn both the graph connections and the feature extraction automatically from the data. The GCN has been evaluated in different deep learning architectures combined with gated recurrent units (GRU), as well as with input signals in both the time and frequency domain, in a multi-class upper limb motion classification application.

The obtained results show that the proposed cortico-muscular graph neural network (CMGNet) is capable of predicting reaching, grasping and wrist twisting with more than 99% accuracy on average. In addition, it shows promising performance in a challenging classification scenario with 11 different upper limb motions, outperforming state-of-the-art machine learning approaches.

The encouraging results suggest that the development of advanced AI approaches which explicitly consider the inner function of the human neuromotor system may be the key to significantly improve the reliability of neuro-driven devices.

Contents

Abstract i
List of Figures vii
List of Tables ix
List of Acronyms xiv

1 Introduction **1**
1.1 Background 1
1.2 Human Machine Interface 3
1.2.1 Brain Computer Interface (BCI) 4
1.2.2 Myoelectric Interface 8
1.2.3 Limitations of unimodal interfaces 10
1.3 Hybrid Human Machine Interface 11
1.3.1 Works in literature for hHMI 14
1.3.2 Deep learning in hHMI 16
1.4 Thesis aims and structures 18

2 Materials and Methods **21**
2.1 Overview 21
2.2 Deep learning fundamentals 22
2.2.1 Supervised learning and backpropagation 23
2.2.2 Convolutional neural network (CNN) 26

2.2.3 Graph convolutional network (GCN) 28
2.3 Cortico-Muscular Graph Network (CMGNet) 32
2.3.1 GCN with learning structure 32
2.3.2 Network architectures 34
2.4 Network validation 37
2.4.1 Dataset 38
2.4.2 Data processing 40
2.4.3 Data organization 42
2.4.4 Machine learning comparisons 43

3 Results **47**

4 Discussion **59**

5 Conclusions **65**

References **67**

Acknowledgements **87**

List of Figures

1.1 The HMI closed loop. Bottom left: subject-acquisition block; Upper right: processing-feature extraction block; Bottom Right: execution-feedback block 4

1.2 The EEG rhythms 6

1.3 Surface EMG and intramuscular (needle) EMG 9

1.4 Main types of fusion approaches depending on where data are fused: data, feature and decision level [1] 12

1.5 Main types of fusion approaches depending on when data are fused: feature combination, parallel processing and cascade prediction [2] 13

1.6 Main types of fusion approaches depending on how data are fused: additive, multiplicative and their combination [3] 14

2.1 Scheme of the proposed approach 22

2.2 Scheme of a feedforward net 24

2.3 Scheme of a recurrent net 25

2.4 The tested network with parallel processing. The input features are divided in three pathways: the upper and the lower paths receive respectively the data from the EEG and EMG channels, while the middle one that contains the graph layer uses both modalities 36

2.5 The tested network with sequential processing. After the graph's dropout layer, the features are divided according to the input data composition: the upper path receives the first 35 output features (corresponding to the EEG channels), while the bottom path uses the last 6 output features (corresponding to the EMG channels) 37

2.6 Electrodes configuration for the 60 EEG, 7 EMG, and 4 EOG acquisitions [4] 39

2.7 Single trial paradigm and the visual cues of the 11 task 40

2.8 Left: EEG channels' locations before the pre-processing. Right: EEG channels' locations after the pre-processing 42

3.1 Raw EEG and the same signal portion after the pre-processing (subject 7) 47

3.2 Raw EMG and the same signal portion after the pre-processing and the Hilbert transformation (subject 15) 48

3.3 Subjects' mean accuracies of the trained networks predicting the 3 macro classes 49

3.4 Subjects' mean final loss of the trained networks predicting the 3 macro classes 50

3.5 Subjects' mean accuracies of the trained networks predicting the 11 classes 53

3.6 Subjects' mean final loss of the trained networks predicting the 11 classes 54
3.7 Varying subjects' mean accuracy by accumulating the prediction rates of the parallel network for the 11 classes 56
3.8 Varying subjects' mean accuracy by accumulating the prediction rates of the sequential network for the 11 classes 56
3.9 Varying subjects' mean accuracy by accumulating the prediction rates of the network with only a GCN layer for the 11 classes 57
3.10 Varying subjects' mean accuracy by accumulating the prediction rates of the EMG features models for the 11 classes . . . 57
3.11 Upper part: subjects' mean confusion matrices of the predicted trial classes at different moments. The first is at 10% between the maximum and the minimum values of the EMG which correspond to the effective movement trigger. The second is after 300 ms from the previous point which is considered the maximum delay for online application [5] and the last is at the end of the considered trial length. The white spaces correspond to values of zero. Bottom part: best accumulating subjects' mean accuracy predictions for the 11 classes in comparison with the trials mean EMG Hilbert transformation of the first EMG channel 58
3.12 Normalized weighted adjacency matrix of the time input sequential net with and without fixed diagonal of one subject (subject 15) for the 11 classes 58

List of Tables

2.1 Summary of the parameters used for the nets' training 37
2.2 Summary of the training parameters used within the Adam optimizer 38

3.1 Summary of the subjects' mean accuracies and losses of the tested networks for the 3 macro classes in the time domain . . 51
3.2 Summary of the subjects' mean accuracies and losses of the tested networks for the 3 macro classes in the frequency domain 52
3.3 Summary of the subjects' accuracies and losses of the tested networks for the 11 classes in the time domain 55
3.4 Summary of the subjects' accuracies and losses of the tested networks for the 11 classes in the frequency domain 55

List of Acronyms

ALS Amyotrophic Lateral Sclerosis.

AMICA Adaptive Mixture ICA.

ASR Artifact Subspace Reconstruction.

BCI Brain Computer Interface.

BMI Brain Machine Interface.

CAR Common Average Reference.

CCA Canonical Correlation Analysis.

CNN Convolutional Neural Network.

DL Deep Learning.

DNN Deep Neural Network.

ECoG Electrocorticography.

EEG Electroencephalography.

EMG Electromyography.

EOG Electrooculography.

EP Evoked Potential.

ERD Event-Related Desynchronization.

ERP Event-Related Potential.

ErrP Error Potential.

ERS Event-Related Synchronization.

fMRI functional Magnetic Resonance Imaging.

fNIRs functional Near-Infrared Spectroscopy.

GCN Graph Convolutional Network.

GD Gradient Descent.

GNN Graph Neural Network.

GRU Gated Recurrent Unit.

hHMI Hybrid Human Machine Interface.

HMI Human Machine Interface.

ICA Independent Component Analysis.

KRLS-T Kernel Recursive Least Square Tracker.

LDA Linear Discriminant Analysis.

LSTM Long Short-Term Memory.

MAV Mean Absolute Value.

MDF Median Frequency.

MEG Magneto Encephalography.

ML Machine Learning.

MNF Mean Frequency.

MRI Magnetic Resonance Imaging.

nEMG Needle Electromyography.

NIRs Near-Infrared Spectroscopy.

NS Negative Slope.

PSD Power Spectral Density.

ReLU Rectified Linear Unit.

RF Random Forest.

RGB Red Green Blue.

RMS Root Mean Square.

RNN Recursive Neural Network.

SD Standard Deviation.

sEMG Surface Electromyography.

SSEP Steady State Evoked Potentials.

SSVEP Steady State Visual Evoked Potentials.

STFT Short-Time Fourier Transform.

SVM Support Vector Machine.

WL Wavelet Length.

WT Wavelet Transform.

ZC Zero Crossing.

1 Introduction

1.1 Background

The rapid technological advancement has seen an increase in the use of computers in fields where normally there would not be. The area most affected by this phenomenon is the medical system, which has acquired new methods and tools for diagnosis and treatments. Some of them has the objective not to cure the people, but to improve their life because the illness that afflict them can be impossible to totally resolve or irreversible, *e.g.* amyotrophic lateral sclerosis, amputations or Parkinson disease.

One of the tools that is receiving attention from the researchers is the human-machine interface (HMI), that refers to a channel that allow communications between a person and a machine, computer program, or system. Other terms exist which are synonyms or similar concepts, such as human-machine interaction [6] or man-machine interface [7]. The HMIs' field is very wide, they can be applied everywhere, in fact every day we use some HMI such as the mouse and the keyboard to utilize the computer, the smartphone touchscreen, the buttons to change the radio station or the volume in the car or also the switches in an industrial machine.

The medical HMIs utilize biological signals to determine what action should be done based on them. Especially the neurophysiological signals

has started to been used thanks also to the fact that they are electrical by nature and so they can be measured easily without too much effort. For this reason the neuroscience has started to develop HMIs to interpret the signals and use them to predict what the subject is doing anticipating the action. This high HMIs usage has led to two new terms, one to refer to this specific HMI application, the brain-computer interface, and the other to the neuroscience branch that implement the command execution into robots, the neurorobotics.

Usually electroencephalography (EEG) and electromyography (EMG) are used in neurorobotics due to the fact that they are non-invasive and their components are portable therefore make these type of acquisitions wearable by the subjects. EMG signals are more informative about various movements, but the activity can be impaired by various neuromuscular pathologies. The EEG signal instead, can be used without the actual movement, so it can potentially predict the intentions before the actual execution. On the other hand it is noisier and more complex to process due to artifacts and non-stationarities.

To overcome these issues, some HMIs that use more than a modality are born and called hybrid or multimodal HMIs (hHMI). The aim of hHMIs is to extract the most discriminative information from every signal and combine them to increase the performance of their unimodal counterparts. A lot of research effort is spent in identifying novel approaches to fuse multimodal signals since different fusion methods can drastically change the results, in fact many research papers are constantly published to understand and improve them but also to increase the chance for a progress breakthrough.

My thesis try to propose a bioinspired hHMI that simulates the connectivity between the brain and the muscles and also between themself [8, 9, 10].

The interface learns to combine the features coming from the multimodal data by itself, making the model more independent of the choices made during the development and more subject dependent, improving its inter-subject adaptability.

1.2 Human Machine Interface

The birth of HMIs is due to the continuous technological advancement, that has led to low cost and more powerful electronics and the necessity of helping people. By definition, there is a large variety of interfaces with very different usage, *e.g.* home, hospitals and industries

As has been said in Section 1.1, the terms human machine interface indicate anything that is between a human and a machine, computer program, or system that is controlled by him. A lot of types of HMI do exist with many applications, such as the button that can start some action from a machine, the remote controller for the television or also the graphic interface that is used for controlling a dam or a nuclear reactor.

Every HMI can be schematized as a closed-loop. It is composed by three blocks that send signals to each other [Figure 1.1] [11, 12].
The starting point is the subject-acquisition block where the human interaction or intention is acquired and processed. Then the feature extraction from the previous information is done to use a classification or regression to transform the command machine readable. The control data are sent to the last part of the loop, the execution-feedback block that return a visible feedback on the interface or with the commands' actuation on devices.

The most common unimodal HMI that uses biological signals are the brain computer interfaces that use the brain signals and the myoelectric interfaces

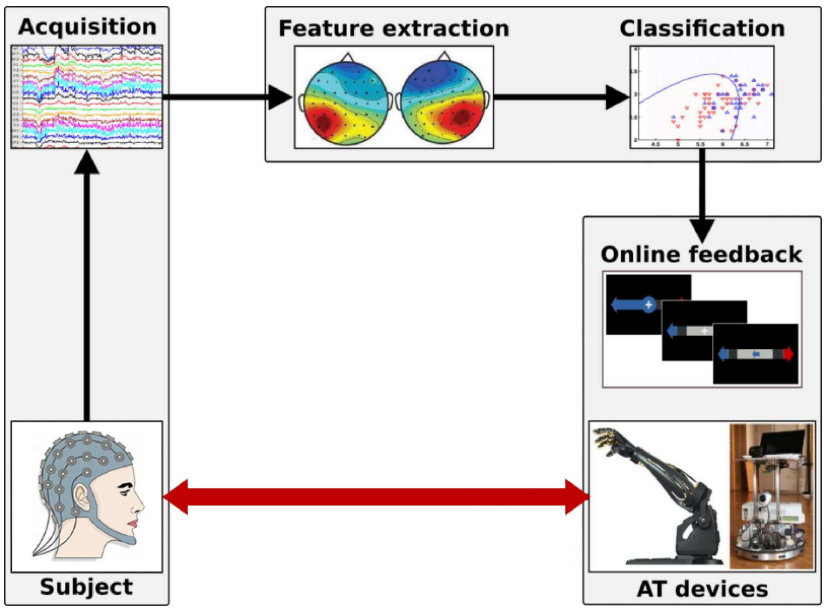

Figure 1.1 The HMI closed loop. Bottom left: subject-acquisition block; Upper right: processing-feature extraction block; Bottom Right: execution-feedback block

that instead use the EMG signal.

1.2.1 Brain Computer Interface (BCI)

Electroencephalography (EEG) is one of the most common signal used in BCI, since it is noninvasive, low-cost, and with an high temporal resolution compared with other methods [13]. The EEG has become popular, unlike other methods such as magneto encephalography (MEG) or functional near-infrared spectroscope (fNIRS), for its inexpensiveness, the higher wearability and its direct measures of the brain activity [14, 15].

The signals are recorded by placing some electrodes on the scalp to detect the pyramidal neurons' electrical activity, which are parallels to each others. The main reasons that make these neurons very important are their firing synchronisation and their proximity to the head, which leads to an increase in the common electrical fields that can be detectable from the sensors [16].

The spatial resolution is poor due to the combination of signals from many neurons close to each others, while the temporal resolution is very high because the recordings are directly collected from the activity sampling. The EEG signal is very noisy as a result of the presence of neurons activity also in a resting state. For this reason, the signal without external stimuli is called spontaneous and, when stimulus are given, a brain response can be seen in the signal in the form of an Event-Related Potential (ERP) [17].

The amplitude of the spontaneous EEG oscillations is in the range of 50-200 μV while for the ERP is between 10-20 μV.

In the frequency domain, the EEG occupies a range from 0.1 Hz to 100 Hz that is commonly divided in the literature into five sub-bands linked with different neurophysiological states. The five bands are illustrated in Figure 1.2 [18, 19]:

- [0.1 - 4 Hz] *Delta waves*: The amplitude decreases with the age and in adults they are associated with a deep sleep. If a large activity is present during the waking state, the waves are considered pathological. They are present posteriorly in child and frontally in adults.

- [4 - 8 Hz] *Theta waves*: They are common in children and in adults during sleep or, in some cases, linked with stress. They are located in the parietal areas.

- [8 - 13 Hz] *Alpha waves*: Present during relaxation in a waking state and their amplitude increases if the eyes are closed and decreases if reopened or some mental activity is done. They are principally located in the occipital areas. Within this same frequency range, there is also the μ *rhythm* which is associated to motor activity and, thus, it is located over the sensory-motor area.

- [13 - 30 Hz] *Beta waves*: Their presence indicates that the brain is involved in some mental effort during the waking state. These frequencies are also linked with movements that can be real or imaginary. They are present in the frontal and parietal regions.

- [30 - 100 Hz] *Gamma waves*: They are linked with deep concentration or during muscle maximum contraction. The activity is present on the somatosensory cortex.

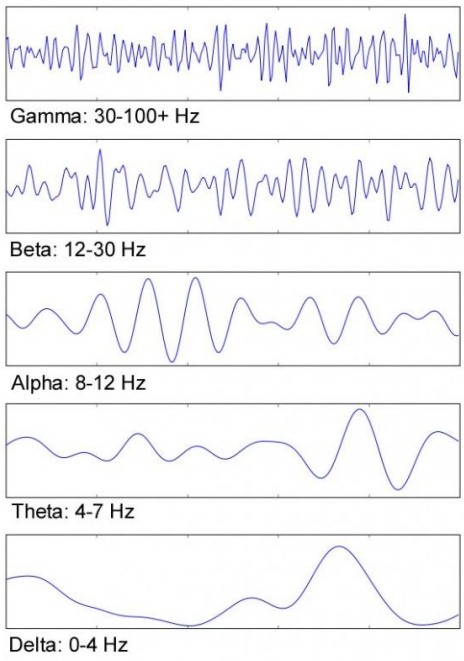

Figure 1.2 The EEG rhythms

Regarding the evoked potentials, which come from the presence of external stimuli, two of the most common ones are the *P300* and the *Steady State Visual Evoked Potentials* (SSVEP)[20]. The P300 occurs in response to an infrequent stimuli, and it's characterized by positive peak about 300 ms after the rare stimulus appearance, commonly in the scalp central position [21].

The SSVEP is a type of steady-state potential which is evoked by periodic or repetitive visual stimuli [22].
Another recently explored ERP is represented by the *error potential (ErrP)*. This potential is generated as a consequence of an incorrect action (*response ErrP*), of a feedback that indicates a mistake (*feedback ErrP*), from observing other people making an error (*observation ErrP*) or when an object does not respect a certain order (*interaction ErrP*) [23]. The ErrP is characterized by a three-phase wave with two negative peaks and a positive peak in the middle, appearing in the central-frontal area approximately 450 ms after the error recognition.

Another category of features that can be extracted from the EEG signal are the ones voluntarily generated by the subject through the imagination and/or execution of a movement, *i.e. sensorimotor rhythms*. In particular, the modulation of the μ and β frequency bands is attenuated in correspondence to a motor task or a sensory stimulation, that can be also imagined, a phenomenon known as Event-Related Desynchronisation (ERD).
When the motor task is terminated, the Event-Related Synchronisation (ERS) of the β band occurs, also known as β-rebound [24]. The ERS symbolize a deactivated or inhibited cortical areas, instead the ERD represent the activated areas related to the motor imagination or the preparation for the execution of a movement.

The BCI gives new channels to control and communicate that do not depend on the normal brain routes like muscles and nerves and so, the possibility to control devices without the necessity to physically control them. This is very useful for people with some disabilities caused by, for example, strokes, amputations, spinal cord injury or ALS, which impairs the communication between the brain and the other body parts.

BCIs can be with or without the aim of rehabilitation, both using a open or closed loop system. The latter objective is to analyze the brain activity to create a direct pathway to control the equipment, the other instead uses the feedback coming from the system to recover the brain activity and plasticity, [25].

The BCIs can be also classified according to the type of signal acquisition used [26]. The non-invasive BCI use the homonymous signal acquisitions, which has only the problem of low amplitude that needs to be amplified. The invasive BCI has a better signal in amplitude, frequency range and spatial resolution but it implies other difficulties, such as the surgical operation with insertion risks and the formation of glial scars, that after some time involves the removing of the electrodes with a new operation.

Synchronous and asynchronous BCIs [27] exist as well. For the synchronous BCI the activity has to be made in precise moments, *e.g.* after a visual cue, and outside that period the computer will not receive any inputs. On the other hand, the asynchronous BCIs can receive commands in every moment, so the subject can do the task at self pace. Clearly these methods gives more flexibility but it needs to be always active and it has to classify a lot more data than its counterpart, with the risk of delivering undesired spurious commands.

1.2.2 Myoelectric Interface

In this particular type of interfaces the EMG signals are used, which measure the electrical activity of the muscles motor units, but what is really recorded is the signal that the muscle fibers use to generate the strength [28].

For the acquisition there are two possibilities, invasive or non invasive. For the invasive acquisition, needles are placed between the muscle fibers

and they are commonly adopted for diagnostic purposes [29]. For the noninvasive acquisition, electrodes are placed on the skin in correspondence of the muscle belly [30]. The two acquisition modalities provide signals with different amplitudes and wave forms, *i.e.* the amplitude (sEMG: 500μV-2mV, nEMG: 50μV-50mV) and the useful frequencies (sEMG: 5Hz-500Hz, nEMG: 1KHz-20KHz) [Figure 1.3]. When the muscular effort increases, to obtain more strength the firing frequency of the fibers increase and more motor units are recruited, resulting in a burst with higher amplitude in the EMG recording.

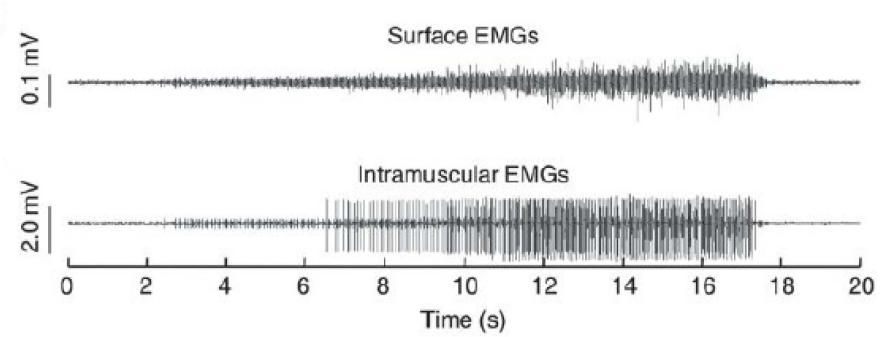

Figure 1.3 Surface EMG and intramuscular (needle) EMG

Also for the EMG signal, features can be extracted in both the time and frequency domain [31, 32] to be used in myoelectric interfaces.

Common time domain features are the *mean absolute value* and the *mean absolute value slope*, the *integrated EMG*, and the *zero crossing*. Other common time features analyse the variation of the signal around its mean, and are the *EMG variance*, the *root mean square* and the *waveform length* or the *standard deviation*. The frequency features, *e.g.* the *frequency median*, *centroid of frequency* and the *modified median frequency*, can be extracted directly from the signal, *e.g.* the *autoregressive coefficients*, or from a prior transformation in the frequency domain using the Short-Time Fourier Trans-

form (STFT) or the Wavelet Transform (WT) [33, 34, 35].

1.2.3 Limitations of unimodal interfaces

Neurophysiological signals have been shown to be very useful in HMI, but taken individually they have some limitations due to their nature.

EEG can be used to predict the subject intentions and executions, but the signal is very noisy and subjected to artifact disturbances that hide the brain-related activity [36]. This increase the difficulty to correctly process the features and to create an online BCI. The source of artifacts can be divided in two categories [37]: the internal artifacts are generated by body activity, *e.g.* eye blinking, neck muscular signal or movement-related artifacts and the external artifacts can be generated from nearby equipment, *e.g.* electromagnetic noises from the power supply or motors. Even if several algorithms have been proposed in the literature to get rid of artifacts, these methodologies have also the risk of removing useful brain information, limiting the reliability of EEG as driving signal to control complex applications [38].

EMG are more informative about the movement, but several neuromuscular pathologies can impair their activity, *e.g.* paraplegia or quadriplegia [39]. For subjects with such problems, the EMG signal is often not usable or the usage of this signal alone to control a robotic device can lead to early muscle fatigue. In addition, the use of the muscular activity alone may prevent the identification of a clear-cut biomarker that precedes biomechanical modifications, which is most likely to be a cerebral signature, *i.e.* anticipatory brain potentials.

For these reasons in the last years more and more hybrid HMI are born to overcome these limitations.

1.3 Hybrid Human Machine Interface

An hybrid HMI, also know as multimodal HMI, records two or more physiological signals and combines their information at a certain level of the processing pipeline [40, 41]. hHMI are used to overcome the single-signal limitations utilizing the strength of multiple modalities to have a better model.

The most discriminant taxonomy in hHMI is where and when to combine the data [42, 43].
Regarding where the data are fused in the model structure, three distinction can be made [Figure 1.4] [1]:

- *Data level*: the acquisitions are fused before every elaboration. This implies that there isn't information loss but it's very susceptible to noise as no preprocessing has been applied yet. Usually the data level fusion is used when the signals are coming from the same or similar source.

- *Feature level*: the features from every data are joined together before the decision-making. Unlike the data level fusion, it has a bit of information loss but handles the noise better. This kind of union is used with modalities that are synchronized or closely paired.

- *Decision level*: after the features classification, the outputs of the classifiers are combined to make a final choice. The noise sensitivity is very low due to the previous elaborations, but it does not provide the possibility to extract multimodal features. Nevertheless, this way of combining the data is the most used in multimodal models because the join is done after the signals are elaborated independently, hence acquisitions that are very different can be used also together.

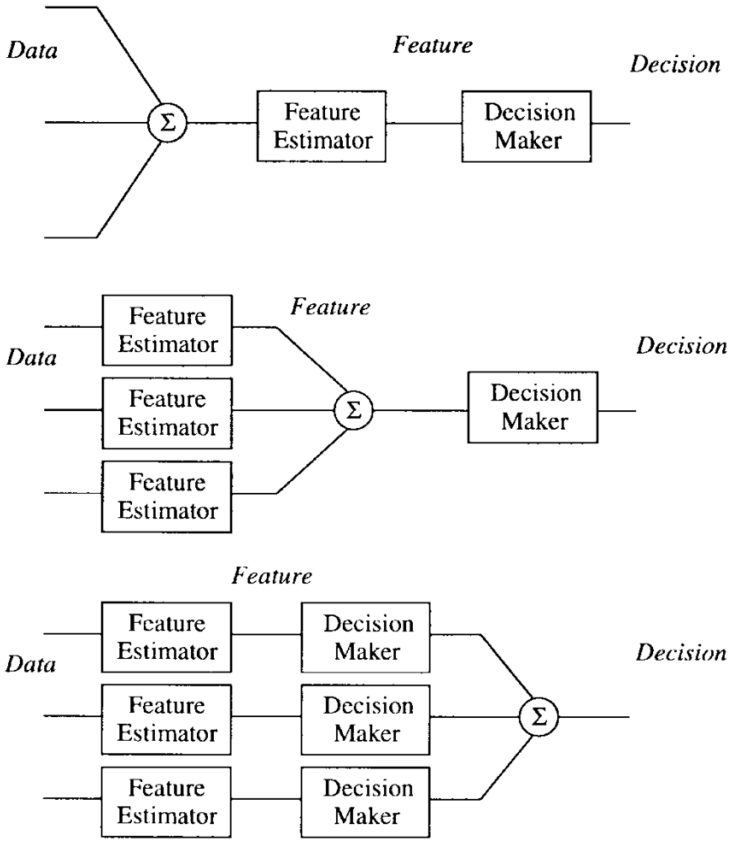

Figure 1.4 Main types of fusion approaches depending on where data are fused: data, feature and decision level [1]

Looking for when the combination can be done other distinctions can be done from a temporal point of view [Figure 1.5] [2]:

- *Feature combination*: is the most common and as input the raw signals or their features are used with same weights to have a complete interaction between them. This union can be seen as data or feature level fusion depending on the model's input. The objective is to improve the model increasing the data informativeness using other signals.

- *Parallel processing*: every modality has its own model and the elaborations are done in parallel. The combination is done at the end with

their outputs, as the decision level fusion, with the purpose to improve the predictions using additional modalities.

- *Cascade prediction*: it processes the modalities sequentially, constraining the outputs relying on the previous. In other words, from the first model decision the second one is chosen and so on, to restrict the possible solutions. This approach is used when a signal is better, or it has more important information, than the others.

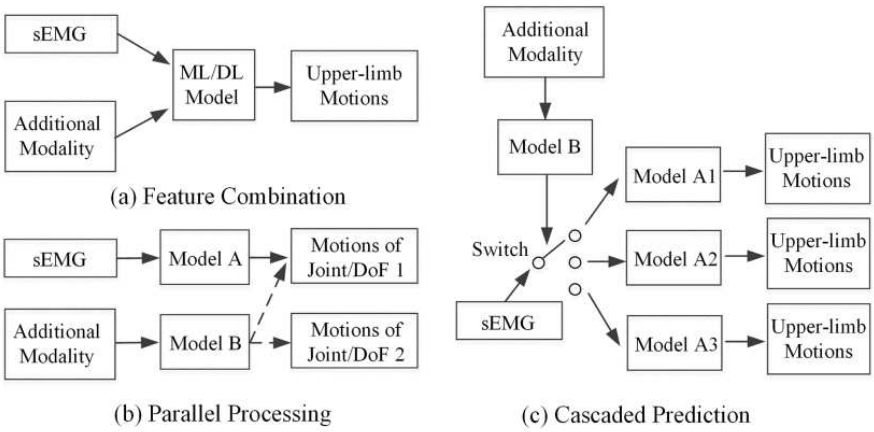

Figure 1.5 Main types of fusion approaches depending on when data are fused: feature combination, parallel processing and cascade prediction [2]

Regarding how the fusion can be done there are two possibilities [Figure 1.6] [3]:

- *Additive fusion*: it's the simplest one, it assumes that every input is potentially useful and so it should be used in the model. The HMI has to determine the quality of each signal using the combined features and this is true if there is a large amount of data.

- *Multiplicative fusion*: it uses some weight for every input to take into account their quality and model meaningfulness. In this way weak

modalities has less importance than the others, restricting their output influence and improving the overall performance. It's also called *Bayesian fusion* because usually the weights are linked with the inverse of the errors' distribution, like the Bayesian inference.

These two modalities can be used together, *i.e.* the additive fusion to create a mixture of the data to increase the number of inputs and the available information, then the multiplicative one to selects useful modalities and their mixtures [3].

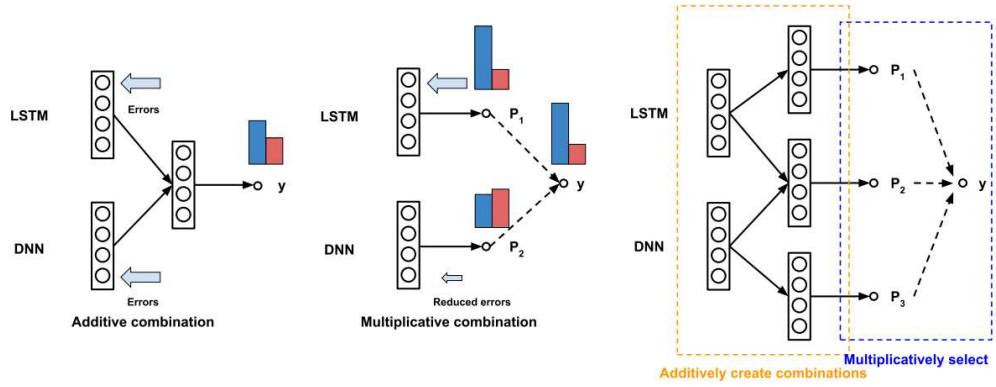

Figure 1.6 Main types of fusion approaches depending on how data are fused: additive, multiplicative and their combination [3]

hHMIs need a deep data analysis to work correctly and to have satisfying performances. For this reason, the models that fit better this need are coming from machine learning and deep learning networks and in fact usually hHMI are developed using these strategies.

1.3.1 Works in literature for hHMI

The classical methods used for the hybrid HMI and the features merging are based on machine learning techniques.

Li et al. [44] acquired sEMG and EEG to improve the control of the hand prosthesis using the features combination. After the data pre-processing, the information from the data were extracted independently and then merged to be feeded into a Linear Discriminant Analysis (LDA) classifier.
Bakshi et al. [45] used the parallel processing of EMG and EEG data to predict the movements of different parts of the arm. When the muscular features were extracted, three Kernel Recursive Least Square Tracker (KRLS-T) were used to tracking the 3D positions of the upper part of the arm, while the EEG information were used in a two-stage multiclass Support Vector Machine (SVM) to estimate the motion of the hand and wrist.
Davarinia et al. [46] with a cascade prediction wanted to obtain the elbow angle trajectory using the EMG and the SSVEP. After the models train, the brain signal was used to distinguish the arm target and so the corrected trained model was chosen and used with the new EMG data to estimate the elbow angle.
Leerskov et al. [47] objective was to check if a subject with spinal cord injury could control an HMI. They extracted the EMG and EEG characteristics independently and then they were merged at features level before being used in a LDA classifier.
Tryon et al. [48] wanted to classify the elbow extension and flexion movements at decision level. They used two SVM classifiers for the two data modalities and then the last SVM for merging the previous decision. This attempt didn't give the hoped-for results, in fact the usage of only the EMG gave better results than the hybrid one.
Riccio et al. [49] used an hHMI to correct the errors during spelling tasks. They used a feature level joint obtaining great results and positive feedback from the subjects.

Leeb et al. [50, 51] did a study about the classification accuracy that decrease in presence of muscular fatigues. In both the reports they merged at classification level, but in the first case they used an additive fusion, like all the previous works, and in the second paper a multiplicative fusion was used. The results were very similar but in the bayesian fusion the standard deviation of the accuracy was lower.
Hooda et al. [52] wanted to classify the unilateral foot movements testing a parallel processing and a cascade prediction. The last structure obtained better results, using the EEG to recognize the resting and non resting state while the EMG was used for the movement classification with a decision tree. In the parallel processing a SVM classifier was used.
Cui et al. [53] classified nine lower limbs movements using a decision level structure. For the EEG a SVM was used while for the EMG a random forest (RF) technique was applied. They used different models for the decision merging such as SVM, RF and nearest neighbour.

1.3.2 Deep learning in hHMI

In recent years, deep neural networks are more and more used in HMI [54]. Differently from standard machine learning (ML) methods that requires hand-crafted features, the advantage of the deep neural networks (DNN) is the possibility to achieve a end-to-end learning in which the most discriminative features in the data are automatically extracted by the model during training [55].

Different hHMI types and DNN models can be found in the literature.
Tortora et al. [56] utilizes EEG and EMG for gait decoding. Every modality is processed in parallel and decoded through stacked LSTM layers, and their decisions are weighted and added using the principle of Bayesian inference.

In particular, the EEG signal is utilized to recognize the movement intention, while the EMG identify which leg is moving. It was seen that the performances of the multimodal network are better than the ones with only the EEG signals or the EMG signals.
Chiarelli et al. [57] uses two brain signals, *i.e.* EEG and fNIRS, to do a motor imagery classification. A parallel processing is used with a 6 layer CNN and the inputs are the moving average of a 1 second window for every channel. The objective is to classify if the motor imagery task is done with left or right hand. Different structures are used but the multimodal DNN has the best performance, with respect to ML models and single modality DNN.

Deep learning in multimodal HMIs is giving interesting and promising results with its restrains. Besides the classical DL limitations [58], such as the dependency on hyper-parameters or the structure and the impossibility to have explicit rules to determine them, the literature models treat different signals independently or at least, if the data come from the same source, they are directly joined [59, 57].

In the case of HMI, and more in BCI, the data coming from the subjects are biological signals that are somewhat linked or with some dependency [60, 61]. The DNN are bio-inspired, in fact they were developed to mimic how brain neurons work, so not consider the coherence between biological signals during the model construction can lead to results that not follow the source functioning and connectivity.

1.4 Thesis aims and structures

Despite the promising performance with respect to single-signal interfaces, the usage of hybrid brain-muscle interfaces for real-time control applications is currently scarce and only a few studies exist that have considered them for clinical applications [62, 47, 63]. The reason for this is twofold: first, as shown in Section 1.3.1, state-of-the-art approaches for combing brain and muscle signals still rely on standard machine learning methods with simple additive fusions (e.g., concatenation, Bayesian fusion), or heuristic decision rules (e.g., the EMG interface is triggered when the BMI output is above a confidence threshold). This aspect poses a strong limit to the performance that can be achieved with hybrid interfaces, particularly in complex decoding tasks such as multi-class classification or limb trajectory prediction. Second, traditional approaches often disregard the temporal relationship among the brain and muscle signals and the deeper semantic coupling to the neurophysiological processes underpinning the movement generation. Corticomuscular coactivation was found to be one of the underlying mechanisms for effective corticospinal interaction which can improve motor functionality [64]. Thus, it would be important for clinical applications of hybrid interfaces to model the cortical and peripheral nerve connectivity, which will be suitable not only for providing neurofeedback related to a therapeutic exercise, but also effective for motor recovery monitoring. Lately, deep learning models have been introduced to open a new perspective for dealing with multimodal inputs, as shown in Section 1.4, but their use with neurophysiological signal is still at its infancy and, definitely not completely explored. In conclusion, in the literature exists a lack of approaches that exploit explicitly the connections that exist in our neuromusculoskeletal system, considering the EEG and EMG

signals as fully independent multimodal inputs.

The aim of this thesis is to combine deep learning and neuroscience to pave the way for a new generation of hybrid neural interfaces. In particular, the breakthrough idea of my work is based on the modeling of the neuromuscular system as a weighted undirected graph: the nodes are the EEG and EMG channels, while the edges represent the brain-to-brain, muscle-to-muscle and brain-to-muscle connectivity. The strength of these connectives is unknown a priori. Thus, in this thesis I propose and validate a novel DNN model to estimate the neuromuscular connectivity through a graph convolutional network (GCN) with learning structure [65], and to use the learned model for the multi-class classification of upper limb movements.

The rest of the thesis manuscript is organized as follows: Chapter 2 first introduces the main principles of standard convolution and graph convolution in DNN. Then, the GCN implementation and the proposed hHMI are described in details. Finally, the dataset used for the method validation, as well as the pre-processing methods to clean the data from noise, are presented; Chapter 3 shows the results coming from the training and testing of proposed GCN, and the performance are compared with respect to machine learning methods which represent the gold-standard for movement classification. The results are discussed in Chapter 4, and some conclusive remarks and future perspectives are provided in Chapter 5.

2 | Materials and Methods

2.1 Overview

The neural networks creation and validation pipeline is composed by the pipeline shown in Figure 2.1. From different subjects, the data is pre-processed with a downsampling, to decrease the data size, and some filters are applied to remove the noise.

Different signal features have been considered as input for the proposed neural network, either the time or frequency domains. In addition, different network configurations have been implemented and evaluated in order to identify the deep learning architecture maximazing the classification performance. To avoid the risk of overfitting, the dataset of each subject is split into a train set, a validation set and a test set. Networks are trained on the train set, the validation set was considered to check the absence of overfitting during the training progress, while the final performance have been obtained on the unseen data of the test set. Two classification scenarios have been taken into consideration: a simpler one in which the network is used for the classification of three macro groups of movements, *i.e.* reaching, twisting and grasping. Then, a more challenging condition is considered in which the network is asked to recognize between 11 different movements. The metrics used for the network goodness are principally the accuracy, the loss and the

confusion matrix between the single movements and their macro groups.

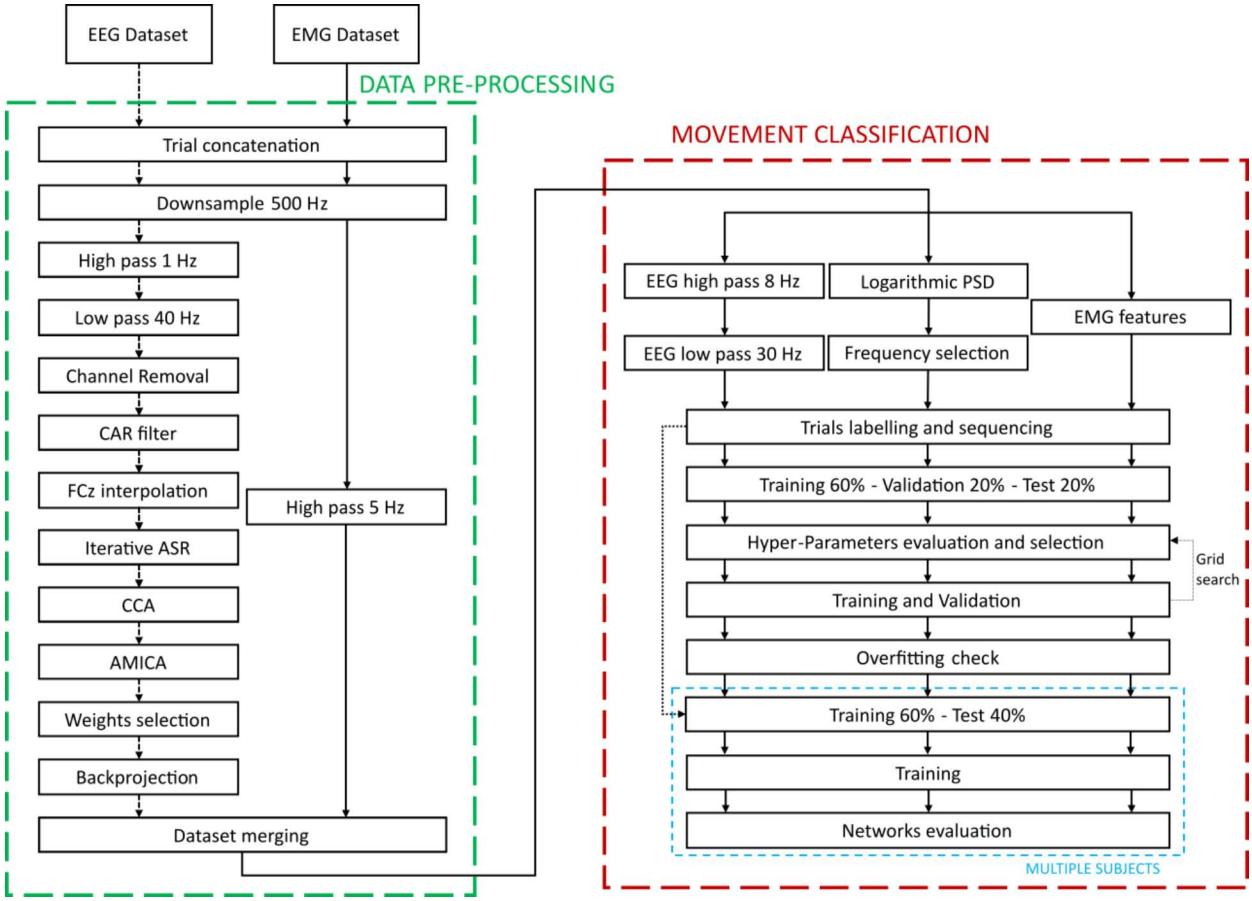

Figure 2.1 Scheme of the proposed approach

2.2 Deep learning fundamentals

The machine learning has evolved during the years creating new learning algorithms and new evaluation techniques. The most important achievements is the improvement of artificial neural networks (ANN), or neural networks (NN), with deeper architectures with better learning abilities called deep learning (DL) [66]. These new methods have been applied in a variety of cases, *e.g.* speech and visual recognition, object detection, genomics and

natural language, surpassing the state-of-the-art thanks to their ability to discover intricate structures in high-dimensional data [67]. The learning process in DL consists in the estimation of the model parameters and weights fitting the input data and the desired output.

Exist different types of NN and learning, *i.e.* supervised, unsupervised, semi-supervised and reinforced [68], but only the first one will be considered due to the fact that it's the learning process used for the proposed graph neural network (GNN). It is a new neural network born just some years ago to try to overcome some situations where the data is not completely suitable for the classical nets. Its core is the usage of the graphs as structure for the features processing.

2.2.1 Supervised learning and backpropagation

The supervised learning need a labelled dataset that will be used in the process of learning, giving the desired outputs with the corresponding inputs [69]. A NN is a series of artificial neurons that create an output using an activation function and the combination of the inputs [70]. The neurons are grouped into multiple layers connected in a network and can be distinguished into *input layer*, that receive the input data, the *output layer*, that is responsible of the NN output, and the *hidden layers* which links the input and output layer. The different neurons connections present inside the layers and between them, are expressed as weight values that are modified during the training process.

The principal supervised nets are of two types

- *Feedforward* [Figure 2.2 [71]: the connections pass the layers output to the next one, and so on until the output layer is reached. In this case connections inside the same layer or with the previous are not present.

- *Recurrent* [Figure 2.3 [72]: it's similar to the feedforward nets but feedback connections are added, usually in the same layer but not always. This links allow the information to persist, giving some kind of memory to the neural net but increasing the complexity of the training and the flow of information. The present of a memory requires to use a sequence of data points which makes this net useful in sequence data analysis such as speech, biomedical signal and natural language processing. The Long Short-Term Memory (LSTM) unit and Gated Recurrent Units (GRU) [73] are two of the most used nets of this category.

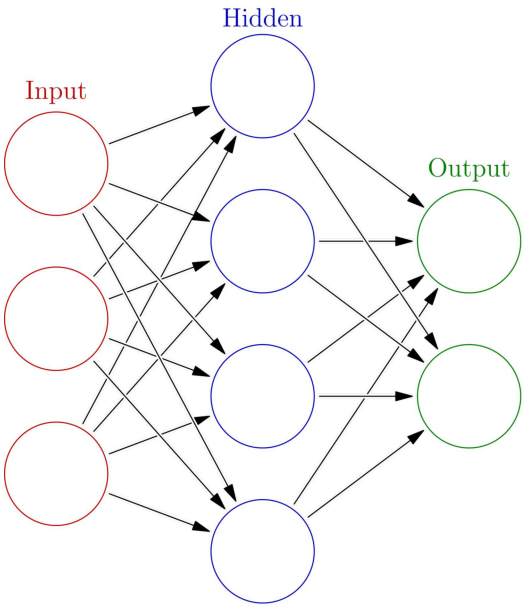

Figure 2.2 Scheme of a feedforward net

The neural networks use the *gradient descent* (GD) and the *backpropagation* to modify their connections.

The GD is used to minimize the loss function L parameterized by the net's weights $W \in \mathbb{R}^m$ by updating them in the opposite direction of the loss gradient $\nabla_W L(W)$. The learning rate α is the size of the step that is done along

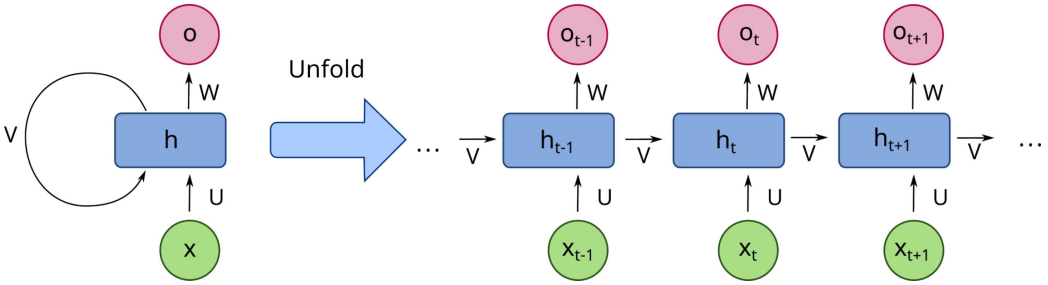

Figure 2.3 Scheme of a recurrent net

the GD [74].

$$\nabla_W L(W) = \left[\frac{\partial L}{\partial W_1}, \quad \frac{\partial L}{\partial W_2}, \quad ..., \quad \frac{\partial L}{\partial W_m} \right] \tag{2.1}$$

$$W^{t+1} = W^t - \alpha \nabla_W L(W) \tag{2.2}$$

The predictions of the neural networks, as seen in this section, are a combination of the input x and the net weights depending on the net structure and its decision function $h(\cdot)$. Calling $\tilde{y} = h(x, W)$ the predictions, the loss is calculated as $L(\tilde{y}, y)$ but since $h(\cdot)$, and so $\tilde{y}$, is a function of functions [75] the computation of the gradient descent is not immediate. For this reason the backpropagation algorithm applied the chain rule to compute the loss gradient to be used in (2.2) [76]

$$\begin{aligned} \nabla_W h(W) &= \left[\frac{\partial h}{\partial W_1}, \quad \frac{\partial h}{\partial W_2}, \quad ..., \quad \frac{\partial h}{\partial W_m} \right] \\ \nabla_W L(W) &= \frac{dL}{dh} \nabla_W h(W) \end{aligned} \tag{2.3}$$

As the number of parameters increase, the calculations for their update increase considerably and this can lead to a weights disappearing or explosion that creates issues for the gradient which can vanish or explode [77].

2.2.2 Convolutional neural network (CNN)

The CNNs are traditional neuronal networks inspired by the mechanism of the animal visual perception [78] and they are composed by different layers such as a pooling, an activation and a fully connected. As the name suggests, the convolution layer uses the convolution to extract the features from the previous layer outputs. The image processing normally uses the spatial filters to acquire the information needed and it is done using an operation of convolution, the same used in the layer. In fact, the CNNs are the best neural networks for images elaboration, for example the classification, the retrieval and the segmentation. Another reason why they are widely used is their ability to benefit from the correlations in the data, both spatially and temporally.

The layer's neurons act as a series of convolutional kernels defined by the width and the stride. The first determines their size, so how many points are taken into account per convolution, and the second instead is the number of points that the kernel skip for every computation. In this way the original input is divided into smaller blocks, called receptive fields, to help the feature extraction [79].

The kernels are composed by the weights that the CNN learns during the training, if the width is 4, the total number of weights for a kernel is 16. These values are used into the convolution as

$$f_j(r,c) = \sum_{x,y} i_n(x,y) \cdot k_j(K_x - x, K_y - y), \quad n = 1, 2, ..., N \tag{2.4}$$

where f is a feature map's pixel of the jth neuron in position (r, c) and i is the receptive field coming from the input that is divided in N smaller parts. The neuron kernel k with dimensions (K_x, K_y) is flipped around the center

and multiplied element wise with the receptive field to be summed together. Usually the kernel's dimensions are equal and odd, to have a squared matrix with a single pixel in the center.

Assuming for the jth neuron that the inputs has dimension (X, Y) and $strd$ is the stride, the output feature map is expressed as

$$F_j = \begin{bmatrix} f_j(1,1) & \cdots & f_j(1,C) \\ \vdots & \ddots & \vdots \\ f_j(R,1) & \cdots & f_j(R,C) \end{bmatrix}$$

with $R = \frac{X-K_x}{strd} + 1, \quad C = \frac{Y-K_y}{strd} + 1$.

Unless it's used $strd = 1$ and $K_x = K_y = 1$, the dimensions of the features maps decrease in the successive layers and their shrinkage is more accentuated with big kernels and with a large stride. This is due to the fact that the input corners can't be a center of a receptive field if it were, the kernel would go out of bounds which is not possible. To overcome this, usually at the input is applied a padding that increases the input dimensions adding pixels around the borders. Different types of padding exist, but the most used is the zero-padding that add zeros to not introduce artificial values and noise. If $K_x = K_y = K$, the padding dimension has to be of $\frac{K-1}{2}$ to maintain the input dimensions in every layer.

Based on the kernel dimensions, three different layers with different convolutional kernels can be constructed.

The *1D convolution* it's the only kernel that is not squared and, as the name suggests, it has only one dimension $(1, K)$. This kernel is preferable to the others for the 1D inputs, *e.g.* biological signals. To reach the state of the art, shallow 1D CNN nets with 1 or 2 layers are sufficient and there are not so many weights to train. Thanks to this, they are easier to process because it's

not necessary to have a big computational power and so its implementation in low-cost and real time devices are suggested.

The *2D convolution* is the most used to process images or videos frame by frame, but also all the data that can be written as a grid, for example the concatenation of signals where the rows and the columns can be channels and observations. They are usually used in deep neural networks for their ability in the features' extraction also from raw data input, and the possibility that increasing the number of layers the features' abstraction also increase. Due to their structure, 2D CNNs can compute big data more efficiently than the fully-connected networks because the neurons are not all connected to each others but they are sparsely-connected. Independently from the input size, the 2D convolutional network can adapt their-self, making themselves immune to small geometric transformation of the data.

The *3D convolution* is the same as the 2D convolution but with 3 dimensions (K_x, K_y, K_z). It's used with data volumes such as RGB images, videos or MRI. Differently from the previous type, the 3D convolution extracts the features from multiple input slices simultaneously.

2.2.3 Graph convolutional network (GCN)

Using classical neural networks, *e.g.* CNN and RNN, it's assumed that the data are coming from an euclidean space. This is not always true, in particular in data mining, biology and molecular chemistry, so a different type of layer is needed. One way to represent non-euclidean data, *e.g.* the connectivity between the brain areas, the structure of molecules or the language processing, are the graphs [80]. The GNNs utilize this structure to predict or classify using euclidean and non-euclidean data making them very adaptable in a lot of situations.

The main applications of the GNNs can be divided in five groups

- *Graph classification*: the neural network learns to divide the graph into subgroups to classify them. It's similar to the image classification or segmentation, but in the graph domain.

- *Node classification*: one or more vertices has to be labeled from their neighbours. The inputs are not all known and the training is made in a semi-supervised way.

- *Link prediction*: differently from the other cases, all or a part of the graph structure is unknown and from the data the network tries to forecast which nodes are connected with each other.

- *Graph clustering*: the vertex or graphs clustering exists. The first organizes the network nodes into highly linked groups based on the edge distance or weights, while int the second the graphs are considered the ones to be grouped based on similarity.

- *Graph visualization*: it's the visual projection of the graphs, which shows the structures and eventually the abnormalities in the data to help the user to understand them.

A graph G is a structure consisting of two components, the nodes or vertices V, and the edges $E \subseteq \{\{m, n\} \in V \mid m \neq n\}$ between them. If the edges link symmetrically the nodes, then it's an *undirected graph*, otherwise it's a *directed graph*.

Calling $ne_{[i]}$ and $co_{[i]}$ the neighbours of node v_i and its edges, l_i and $l_{(i,j)}$ the label of the node v_i and the label of the link between him and the node

v_j, the state x_i can be evaluated as

$$x_i = \sum_{j \subseteq ne_{[i]}} h_w(l_i, l_{(i,j)}, x_j, l_j), \quad i \in V \tag{2.5}$$

$$x_i = h_w(l_i, l_{co_{[i]}}, x_{ne_{[i]}}, l_{ne_{[i]}}) \tag{2.6}$$

where h_w is a parametric function that transform the input vector into a d dimensional space. The (2.5) refers to a general GNN where the relative positions of the nodes to each other is not important, instead for the (2.6) they are utilized in the formula. Given the state x_i the output of the node i is calculated

$$o_i = g_w(x_i, l_i) \tag{2.7}$$

with g_w the parametric local output function [81].

To represent the edges is usually used the adjacency matrix A. Assuming that the total number of nodes is N, $A = \mathbb{R}^{NxN}$ because for every vertex exist N possible links with the others, including itself. If $A(v_1, v_2) = 1$, it means that between node v_1 and v_2 an edge exists, if not $A(v_1, v_2) = 0$. A can be symmetrical and in that case the edges have no direction because every link has its opposite, otherwise at least one edge has not its opposite creating a non symmetrical matrix.

The adjacency matrix A brings only the information about if an edge exists or not, calling the graph *unweighted*. In some cases can be useful to know some kind of numerical proprieties of the links that depends on the problem, *e.g.* if the nodes are cities the edges can be the distance between them but also the travel cost or its time, the traffic density or its probability to occur. To take into account this need, a weight is associated with every edge from A, leading to a weight matrix W and a *weighted* graph. All the

considerations for the adjacency matrix can be applied to the weight matrix with only the variation that inside W the values are not only zeros and ones.

One of the most used GNN variants is the GCN, which computes the convolution between connected nodes to obtain the graph output. For this type of neural network, the variation of the nodes' number inside the layer can lead to a vanishing or exploding gradient problem. To overcome this possibility, the normalized adjacency matrix $\tilde{A}$ is calculated using the degree matrix D and its square root $\tilde{D}$

$$\begin{gathered} D_{ii} = \sum_{j=1}^{N} A_{ij}, \quad i = 1, 2, ..., N \\ \tilde{D}_{mn} = \sqrt{D_{mn}}, \quad m = 1, 2, ..., N \quad n = 1, 2, ..., N \\ \tilde{A} = \tilde{D}^{-1} \cdot A \cdot \tilde{D}^{-1} \end{gathered} \tag{2.8}$$

Can be demonstrated [82] that the nodes' convolution evaluation can be done using the graph matrices

$$C = \tilde{A} \cdot X \cdot \Phi \tag{2.9}$$

with $X \in \mathbb{R}^{NxF}$ the input with N nodes and F features for each one of them, $\Phi \in \mathbb{R}^{FxT}$ a matrix of filter parameters and $C \in \mathbb{R}^{NxT}$ the convolved input matrix.

Due to the fact that the GCN can compute the convolution with nodes that can be both nearby and far away, it can be seen as a generalization of the CNN, which does a convolution between pixels but only with those directly next to. In the GCN instead it is not necessarily that close node are in the same calculation and so the CNN is a particular type of GCN where the pixels are the nodes and the central pixel of every kernel has a link with

the other ones inside it.

2.3 Cortico-Muscular Graph Network (CMGNet)

For utilizing the GCN for classification, the graph's structure has to be known but it's not always the case, because it can be impossible or hard to obtain leaving it completely or partially unknown. Recently some algorithms were developed to try to obtain the edges from the data, using both weighted and unweighted graphs [83, 84, 85, 86].

To obtain the graph structure I was inspired from connections between the brain and the muscles that are both physical and functional [87, 88]. One way to measure this connections is the cortico-muscular connectivity [8], therefore the graph kernel of my network is constructed to simulate these links. The network tries to evaluate the bonds from the data and the acquired model structure tries to simulate the cortico-muscular system for the movement predictions.

I have created different neural networks, using only the GCN and also adding other kind of layers such as LSTM and GRU. I did this because in biological signals exist a time correlation and so, using networks with memory, they can help with the classification. The results with the GRUs were better than the ones with the LSTMs, thus only them are presented.

2.3.1 GCN with learning structure

The GCN layer that I have created tries to obtain the graph structure using a symmetrical weighted matrix W_A while doing a classification. To reduce the computational time, only a vector of weights $M \in \mathbb{R}^{Gx1}$ with $G = N(N+1)/2$ are set to be trainable. The length of M is equal to the number of values of

the lower triangular part of W_A and to reconstruct it, M is reshaped inside W_A and then flipped.
During the training and its error back-propagation, the M's weights are updated and since the graph structure matrix W_A is constructed using them, the edges are generated and refined.

The objective of the GCN layer is to repeat the acquisition setup to see if the graph structure and the nodes that it learns can be used to simulate the real channels. The usage of the weights in the edges could be used to understand them and to give an evaluation of the links that can or could not exist between the sources that have created the signals. In my case I had 35 EEG channels and 6 for the EMG, so the numbers of nodes N is set to be 41 and therefore the weighted matrix has dimensions of 41x41. In this way every channels has a corresponding node and the edges connections are the channels connectivity.

To compute the convolution the layer uses the equations in (2.8) and (2.9), with $A = W_A$ and Φ as a second learnable matrix which is used to define the layer output. In particular the dimension T is the number of output features coming from the GCN.

As input X I have used both the frequency and the temporal domain using EEG and EMG together. Before extracting the inputs, the EMG signals are passed to the Hilbert transform and then their absolute values are taken to make the signal proprieties similar to the EEG ones.

For the time domain, the EEG signals are filtered between the 8 and 30 Hz, the most related frequencies to the movements thanks to the presence of the alpha and beta waves. This approach prefers to remove the most amount of noise while deleting also some signal information to have a high signal-to-noise ratio.

In the latter case the data coming from EEG and EMG is given to the neural network using every acquisition point as single input. The frequency input instead, is composed of the PSD's natural logarithm [Table 2.1] that is extracted from the brain and muscles signals. Not all the frequencies are used, in fact only the ones that are between the 4 Hz and the 30 Hz are held. The reasons are the same of the other domain, but they start from the 4 Hz due to the fact that calculating the PSD the frequencies are more spaced. The presence of the GRUs in the networks implies that the inputs have to be constructed as sequences of the signals or of the PSDs because blocks of data is needed for the memory flow.

2.3.2 Network architectures

The constructed networks can have two types of input, one constructed using the data points and the other with the data PSD. Referring to the equation (2.9) only the matrix X changes, in the first case the number of features F are equal to 1 while in the second F is 14 because the PSD's frequencies are spaced by 2 Hz and they are limited between 4 Hz and 30 Hz.

Some considerations were made about the connections of the graph nodes to take into account different cortico-muscular connections.
At the start the adjacency matrix weights were used only to evaluate only the edges between the brain and the muscles without their inter-connections, putting their weights to zero. This was done to consider the system as simple as possible with only two groups that can communicate only to each other to have a first evaluation of the links and to remove spurious bonds that could create noise inside the structure.
To mimic the reality, also the weighted adjacency matrix with all the weights is trained to consider the brain connectivity [89] and the muscles connectivity

[10]. Doing this every link is used, trying to improve the model using the all existing connectivities.
Having every channel corresponding to a node means that the values of every channel with a vertices has to be 1, but giving no restrain in the weights means that is not certain to be respected. For this reason I have also tried to fixed the weights in the weighted adjacency matrix diagonal to 1 to have an effective evaluation of the links between a channel and the others coming from the EEG and EMG.

At the beginning, I created a neural network with only a GCN and a classification block to test the new layer and to create a starting point, using it also as a compass for the networks improvements.

The neural networks parts can be summarized in four blocks

- *Input*: a z-score normalization is done between the sequences
- *GCN*: it's composed by the custom graph layer with a ReLU activation function. Then a batch normalization is done to overcome gradient problems and a drop out layer is placed for the over-fitting.
- *GRU*: only a GRU layer is present with its drop out
- *Classification*: it's the last block and it's used for the classification. It start with a fully-connected layer with a softmax, and then a classification layer to extract the final classes

The first network [Figure 2.4] utilizes a parallel processing to extract the features. A GCN block receives the complete dataset composed by the EMG and EEG data, while other two GRUs blocks are used to process the information that comes from different sources. The starting dataset is splitted into a subgroup that contain the 35 channels EEG signals and another one

that has the 6 channel EMG ones. This division was made with the idea that computing the dataset in different forms independently could have better features and merging them could improve the network performances. Also the GCN block output goes thought a GRU block, trying to learn the temporal correlation, before being concatenated with the other GRUs outputs and to be given to the classification block.

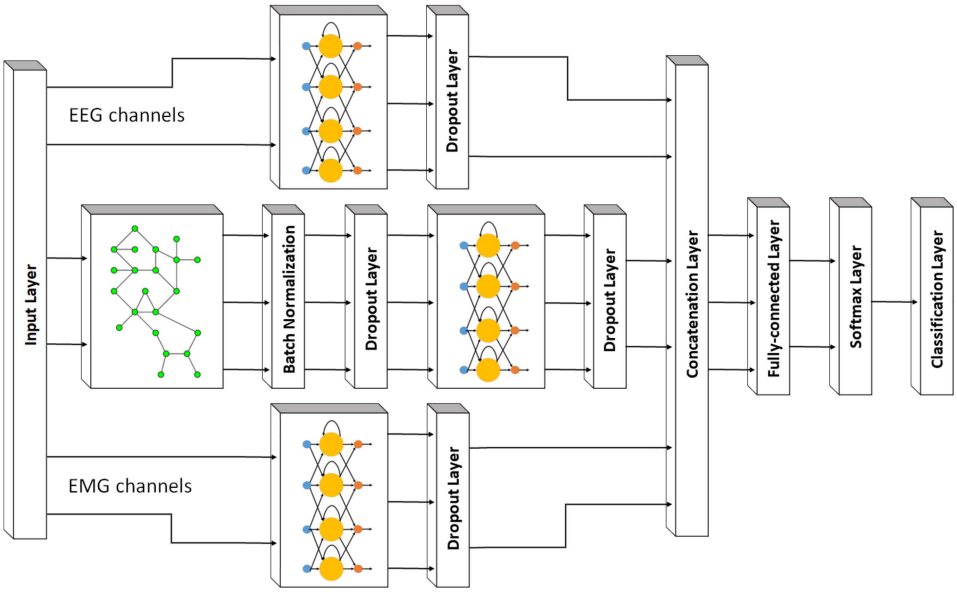

Figure 2.4 The tested network with parallel processing. The input features are divided in three pathways: the upper and the lower paths receive respectively the data from the EEG and EMG channels, while the middle one that contains the graph layer uses both modalities

The second network [Figure 2.5] is similar to the first one with the difference that the blocks are used in series. It start with a GCN block with the complete dataset as input, then its output is divided using the same first network principle, so the upper one use 35 output features and the bottom one the remaining 6. These two new inputs are given to different GRUs to use the time correlations of the data sequences. The GRUs' outputs are concatenated and given to the last block for the classification.

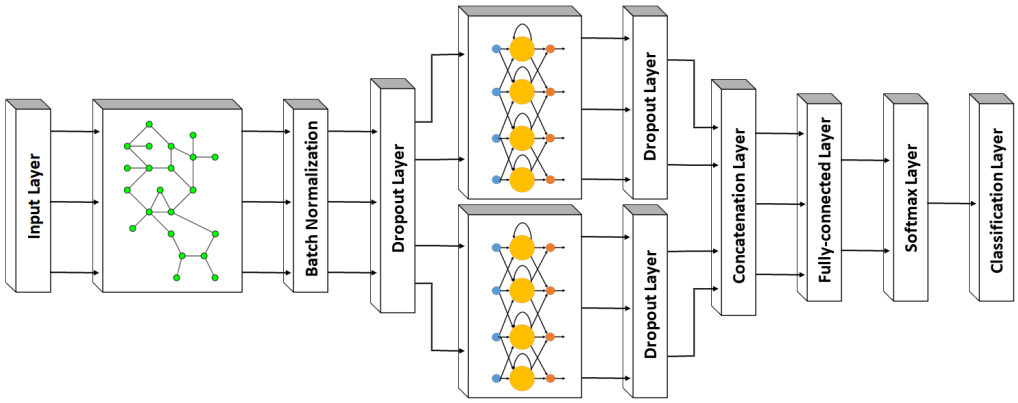

Figure 2.5 The tested network with sequential processing. After the graph's dropout layer, the features are divided according to the input data composition: the upper path receives the first 35 output features (corresponding to the EEG channels), while the bottom path uses the last 6 output features (corresponding to the EMG channels)

With regards to the network hyper-parameters I have done a dense grid search and using the best performances to choose the values.

Train Parameters	Value
GCN output features	42
GRU hidden units	150
GCN drop out	0.3
GRU drop out	0.3
PSD window length	0.5 s
PSD window shift	0.02 s
Input sequence window length	1 s
Input sequence window shift	0.04 s

Table 2.1 Summary of the parameters used for the nets' training

2.4 Network validation

The dataset chosen is coming from an open repository thanks to Jeong et al. [4]. It's composed of EEG, EMG and EOG signals coming from 25 young

Adam Parameters	Value
β_1	0.900
β_2	0.999
ϵ	10^{-8}
Regularization term	L2
Regularization factor	0.0001
Max epochs	7 or 15
Mini-Batch size	256
Initial Learning Rate	10^{-2} or 10^{-3}
Learning Rate Drop	10^{-1}
Drop Period	2 or 12

Table 2.2 Summary of the training parameters used within the Adam optimizer

subjects (15 men and 10 women) that performed 11 different movement upper limb tasks.

The participants are all right-handed and completely healthy without abnormalities that can impair the acquisitions. At them was also asked to get sufficient sleep, avoid alcohol, caffeine, and strenuous physical activity before the experiment.

2.4.1 Dataset

The EEG acquisitions was taken using a 64 channels EEG cap (actiCap, BrainProduct GmbH, Gilching, Bayern, Germany), which 60 of them used for the EEG (10-20 international configuration) and 4 for the EOG [Figure 2.6]. Ground and reference channels were placed on the Fpz and FCz and the impedances of all the electrodes were maintained less than 15 kΩ. The signal was sampled at 2500 Hz with a notch filter at 60 Hz to remove electrical noises. The same acquisition method was used for the EOG, with the objective to use it for removing ocular artifacts in the preprocessing phase. For the EMG, 7 silver/silver chloride electrodes were used [Figure 2.6] and

the EEG acquisition setup was used, such as ground and reference channels, sampling and notch frequency and impedances. The last electrode was placed on the elbow as an alternative reference signal.

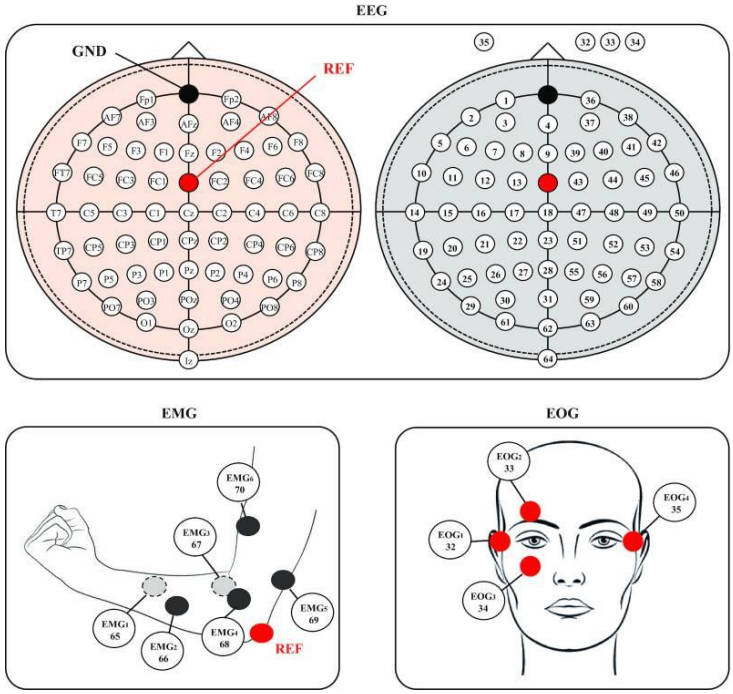

Figure 2.6 Electrodes configuration for the 60 EEG, 7 EMG, and 4 EOG acquisitions [4]

The experiment was divided in 3 sessions to take into account the inter-session, inter-participant variabilities and to let the subjects rest. The data contains the acquisitions of 11 different upper limb movements that were done both via imagery and moving the arm. The tasks can be divided in 3 macro groups, arm-reaching, hand-grasping, and wrist-twisting. The reaching was composed by 6 directions, the grasping by 3 shapes and the twisting by clockwise and anti-clockwise rotation [Figure 2.7].

To pace the movements, visual instructions were provided on a monitor. First, a black cross was displayed for 4 s to rest, then a visual cue with a text sign for 3 s to prepare the participants to the task. When it disappeared,

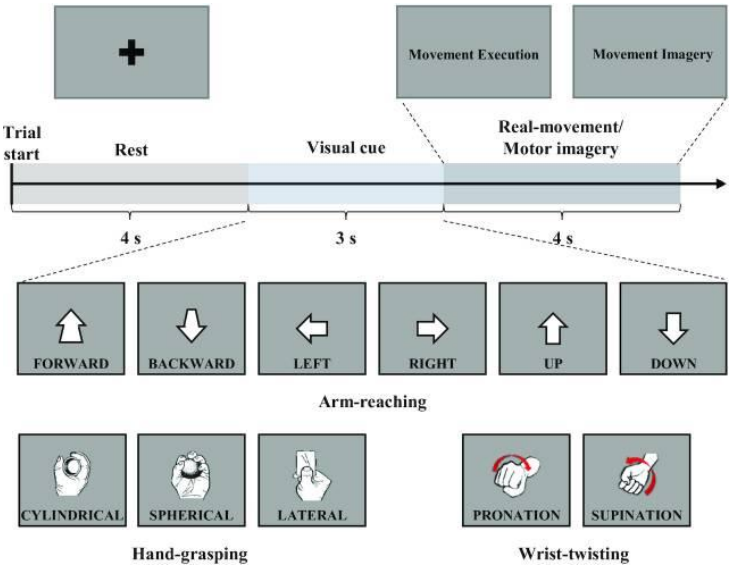

Figure 2.7 Single trial paradigm and the visual cues of the 11 task

a text was presented ("Movement Execution" or "Movement Imagery"), and they were requested to perform the corresponding movement for 4 s [Figure 2.7]. Every movement was done for 50 times per session, for a total of 3300 trials per movement.

The dataset was validated to confirm the accuracy and the truthfulness of the acquisitions and the results was reported in the paper.

2.4.2 Data processing

The public dataset is divided in subsets by session, subject, type of macro movement and if it's a real or imaginary movement is done. Due to the presence of significant EMG signal only in the effective movement the imagery one is discarded. To prepare the dataset for the thesis purpose, a pre-processing pipeline is applied to it which can be seen in the left part of Figure 2.1.

First of all, the files from 5 different subjects were concatenated to have a continuous signal like a single session. To decrease the space occupied and to

speed up the following operations of pre-processing the sampling frequency was decreased from 2500 Hz to 1000 Hz.

To the EEG was applied a zero-phase forward and reverse filtering with a low and a high pass 4^{th} order Butterworth filter [90]. The chosen frequencies were of 1 Hz to clean every slow artifact, *e.g.* respiratory, and of 40 Hz to keep only the important information about the movements.
For the EMG instead only a high pass filter with the same modality and type for the EEG was utilized. The cutting threshold was of 5 Hz because in the signal, due to the movements, various drastic changes in its mean were present. A low pass filter was not applied because in the process of down-sampling, it was already applied with a cut off frequency equal to the new sampling. So a low pass filter of 500 Hz is sufficient to clean the EMG [91].

I have chosen to remove the electrodes and their signals that are not coming from the motor sensory cortex to not introduce other noises or neurons activations that are not due to the movements [92]. From the starting 60 electrodes, after removing the ones that are not useful, they ended to be 34. The FCz electrode is used as reference in the EEG and so its signal and location is not present [Figure 2.8]. This gap in the cap can compromise the correct evaluation of the dataset. This was corrected using a common average reference (CAR) filter to re-reference the EEG signal [93] and then using a spherical interpolation from the other electrodes to obtain the FCz values. Adding this signal, the final number of electrodes increased to 35.

To remove the noise from the signals, a sequence of algorithms was used. Starting with artifact subspace reconstruction (ASR) [94] to normalize the peaks that surpass a threshold, and canonical correlation analysis (CCA) [95]. At the end, Adaptive Mixture ICA (AMICA) [96] was applied to decompose the signals into components and then the ICLabel algorithm [97] was used to

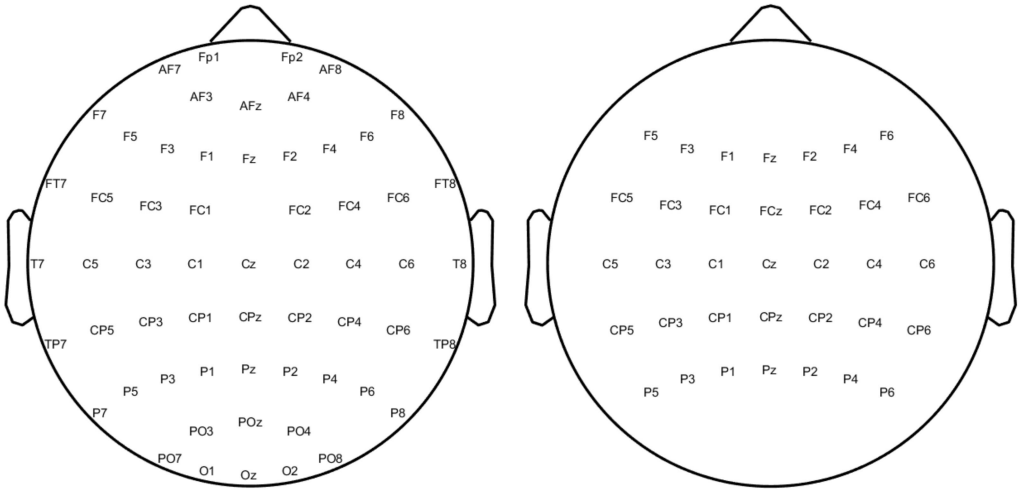

Figure 2.8 Left: EEG channels' locations before the pre-processing. Right: EEG channels' locations after the pre-processing

have a first recognition of their nature, *e.g.* brain, muscles, eyes and noise,. To better recognize which are not coming from neuronal activations, I have divided the dataset into epochs using the movement onset as a reference to see if some peaks appears in the same moment every time. Only the components that appear to have brain signals with a good signal to noise ratio were held.

2.4.3 Data organization

The data that were used to train and evaluate the neural networks were divided into a training, a validation and a test set.

The fact that the signals were divided for computing the PSD, implies that the casual division into the subsets can put adjacent windows in different groups, *e.g.* the validation and the test set. This can impair the training and the performances evaluation, because signal parts that are nearby are correlated due to the fact that are biological and belonging to the same

movement. If this happens, it would bring to an over evaluation of the accuracies for the reason that conceptually very similar signals are used in subgroups that must be independent to each others.

For this reason, the division was made between whole trials instead of the individual windows. To maintain the correct distribution of the movements, every class was divided independently to not have strong imbalances and to train the neural network correctly. I have placed 60% of the trials in the training set, 20% into the validation set and the last 20% in the test set for the initials evaluations. The final trainings are performed removing the validation set and distributing its data into the other subset, which then were of 60% of the total for the training set and the remaining 40% for the test set.

2.4.4 Machine learning comparisons

To make a comparison between the state-of-the-art EMG movement classification and my networks, I have used two machine learning techniques, linear discriminant analysis (LDA) and random forest (RF) trained with a set of hand-crafted features [98].

LDA try to find the best discriminant features considering the distance between-class and the distance within-class, maximizing their ratio [99], while the RF is a set of tree-structured classifier where the inputs are chosen randomly with the same probability among the input features and every tree vote for an output [100].

The chosen EMG features were selected from the literature [101]. Calling x the EMG signal inside a window with N total acquisition point and P_j the power spectrum at frequency j with M total frequencies, I have extracted these features for the movement classification, which formulations can be

found in [31]

- *Mean absolute value* (MAV): the average of the EMG absolute values

$$MAV = \frac{1}{N}\sum_{n=1}^{N}|x_n| \tag{2.10}$$

- *Standard deviation* (SD): the standard deviation of the EMG values

$$SD = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(x_n - \mu)^2}, \quad where \;\; \mu = \frac{1}{N}\sum_{n=1}^{N}x_n \tag{2.11}$$

- *Zero crossing* (ZC): how many times the EMG cross the value of zero

$$\begin{gathered} ZC = \sum_{n=1}^{N-1}\left[sgn(x_n \times x_{n+1}) \cap |x_n - x_{n+1}| \geq 0\right], \\ where \;\; sgn(x) = \begin{cases} 1 & if \;\; x \geq 0 \\ 0 & otherwise \end{cases} \end{gathered} \tag{2.12}$$

- *Root mean square* (RMS): the average of the EMG squared values

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N}x_n^2} \tag{2.13}$$

- *Waveform length* (WL): the cumulative length of the EMG waveform

$$WL = \frac{1}{N}\sum_{n=1}^{N-1}|x_{n+1} - x_n| \tag{2.14}$$

- *Mean frequency* (MNF): the weighted mean of the EMG frequencies

and their power spectrum

$$MNF = \frac{\sum_{j=1}^{M} f_j \ P_j}{\sum_{j=1}^{M} P_j} \tag{2.15}$$

- *Median frequency* (MDF): the EMG frequency with the median power spectrum value

$$\sum_{j=1}^{MDF} P_j = \sum_{j=MDF}^{M} P_j = \frac{1}{2} \sum_{j=1}^{M} P_j \tag{2.16}$$

For the EMG windows the suggested length are about 200-400 ms, but the classification accuracy increases with the windows' length [102]. Based on this and to keep consistency with the PSD windows used in the MCGnet, the EMG features have been computed in moving windows with a length of 500 ms and a shift of 20 ms.

3 | Results

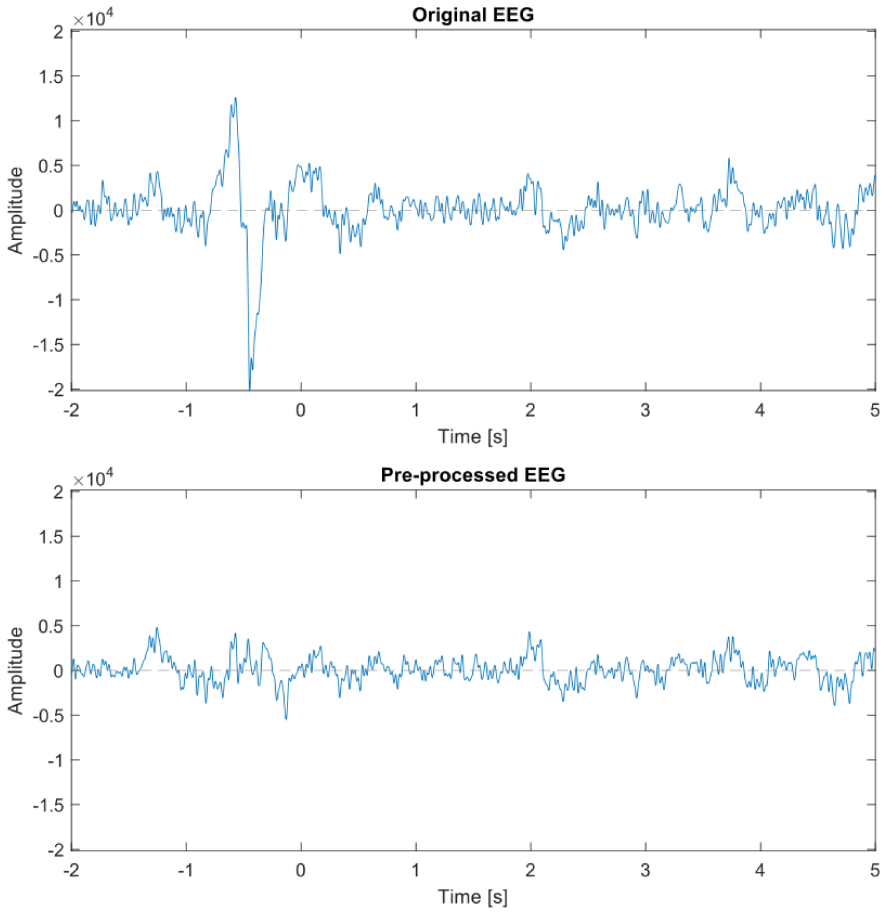

Figure 3.1 Raw EEG and the same signal portion after the pre-processing (subject 7)

The pre-processing in the EEG and EMG signals has successfully re-

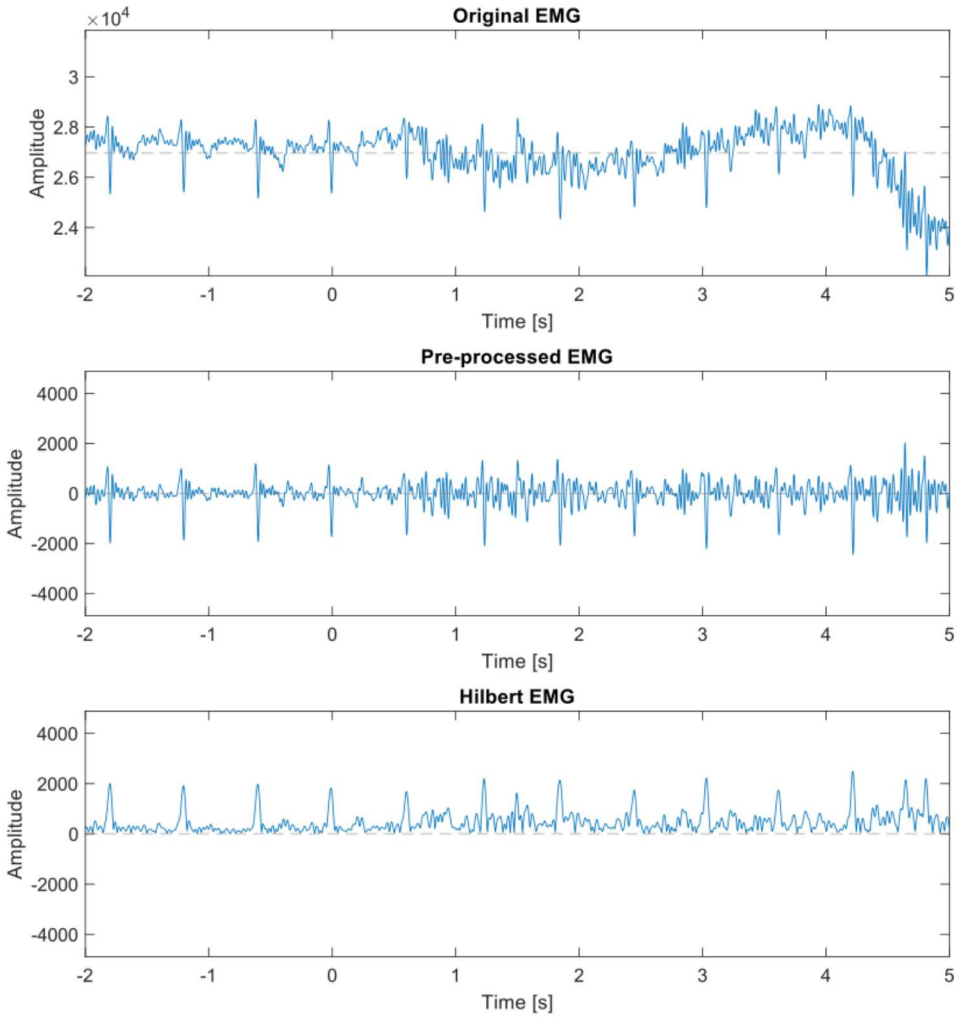

Figure 3.2 Raw EMG and the same signal portion after the pre-processing and the Hilbert transformation (subject 15)

moved the principal artifacts and noises from the data. In Figure 3.1 a high frequency noise is present and also a very high amplitude artifact around the zero with smaller ones during the movement can be found in the signal, while in the processed one they are not present. In the raw EMG is present an high low frequency component due to the electrodes shifting during the movement that have been removed by the pre-processing [Figure

3.2], together with the high-frequency noise. Even after the pre-processing, the subject's heart-beat (i.e., electrocardiogram (ECG)) is still overlapped with the EMG signal. Eventually, it can be removed with an adaptive filter [103], but since this artifact represents a common signal component to all the trials, it does not compromise the classification performance as it is not a discriminant feature.

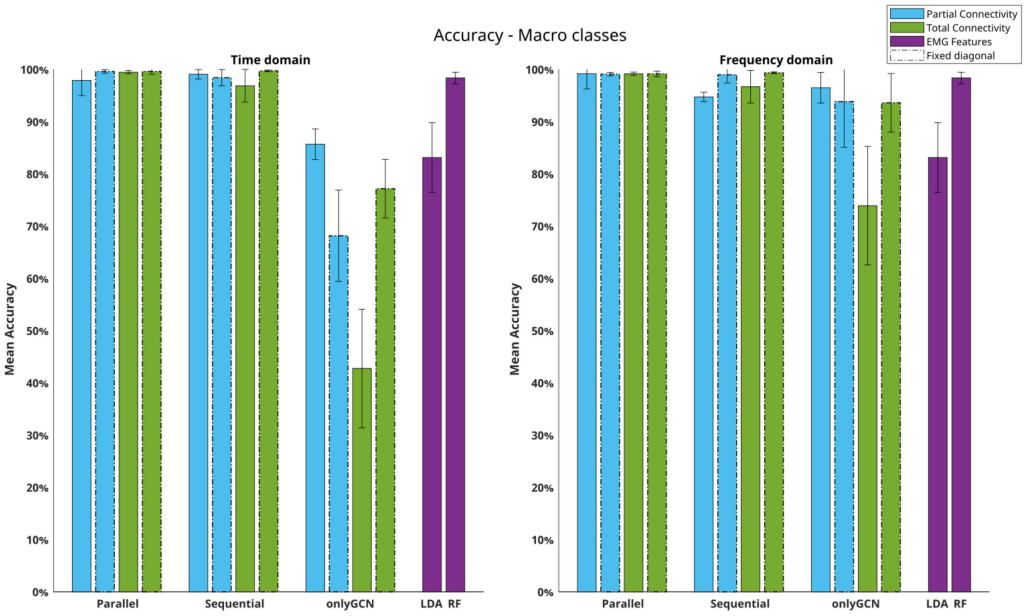

Figure 3.3 Subjects' mean accuracies of the trained networks predicting the 3 macro classes

The subjects' mean accuracies [Figure 3.3] evaluated using the macro classes for the parallel and the sequential networks matches the random forest ones in both input domains, reaching the 100% recognition rate. The net that uses only a GCN layer has different results depending on the input. The higher accuracies are obtained using the GCN trained with inputs from the frequency domain. Using features from the time domain, the results are lower but comparable with the LDA results. The worst result is obtained

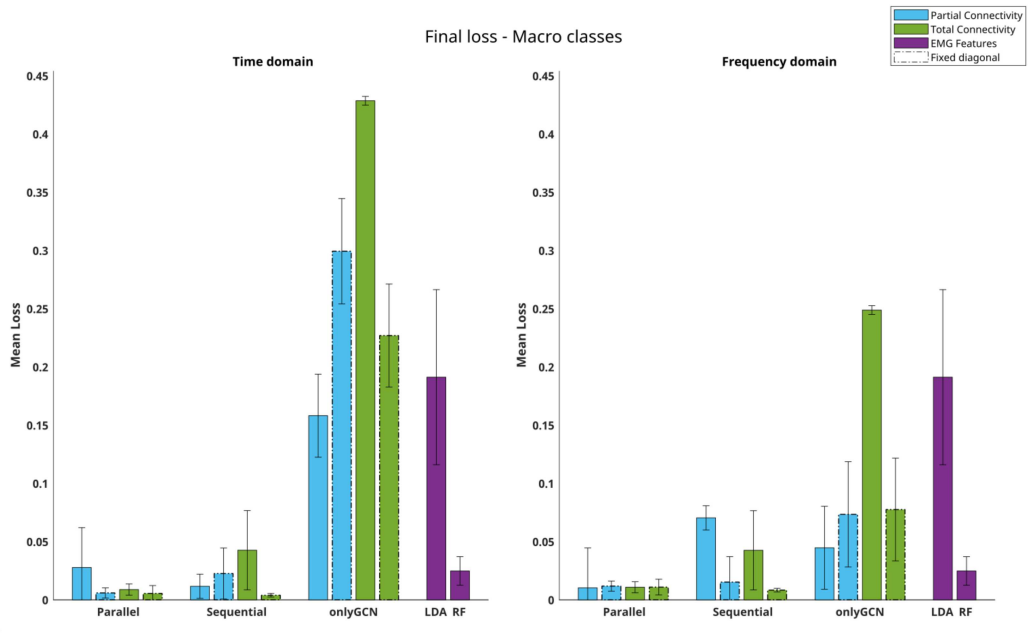

Figure 3.4 Subjects' mean final loss of the trained networks predicting the 3 macro classes

by the only GCN in the time domain with total connectivity and no fixed diagonal, achieving an accuracy of 43±11% which is the lowest. With regard to the subjects' mean cross-entropy loss [Figure 3.4], it reflects the situation in the recognition rate, in fact the networks that has a higher accuracy has also a lower loss.

Considering the case with the 11 classes [Figure 3.5] and the input in the time domain among the subjects, as for the macro classes the net with only the GCN layer has the worst results, but still comparable with the results obtained by the LDA classifier. The other networks, *i.e.* the parallel and the sequential, reach the random forest's results, with two exceptions. The second net with total complete adjacency matrix and fixed diagonal surpasses the state-of-the-art, while the same structure but without the fixed diagonal reaches only the LDA accuracy. Regarding the frequency domain results

Type of net		Accuracy [%]	Loss
Parallel	Partial Connectivity	98 ± 3	0.03 ± 0.03
	with fixed diagonal	100 ± 0	0.01 ± 0.00
	Total Connectivity	99 ± 0	0.01 ± 0.00
	with fixed diagonal	100 ± 1	0.01 ± 0.01
Sequential	Partial Connectivity	99 ± 1	0.01 ± 0.01
	with fixed diagonal	98 ± 2	0.02 ± 0.02
	Total Connectivity	97 ± 3	0.04 ± 0.03
	with fixed diagonal	**100 ± 0**	**0.00 ± 0.00**
Only GCN	Partial Connectivity	86 ± 3	0.16 ± 0.04
	with fixed diagonal	68 ± 9	0.30 ± 0.05
	Total Connectivity	43 ± 11	0.43 ± 0.00
	with fixed diagonal	77 ± 6	0.23 ± 0.04
LDA		83 ± 7	0.19 ± 0.08
Random Forest		98 ± 1	0.02 ± 0.01

Table 3.1 Summary of the subjects' mean accuracies and losses of the tested networks for the 3 macro classes in the time domain

instead, almost all networks' result are similar without much differences with only one exception, the simplest net with the full weighted adjacency matrix without the fixed diagonal has the worst results that are very far from the other networks.

Using all the classes higher subjects' mean accuracies doesn't imply lower subjects' mean losses [Figure 3.6]. In fact, in the time domain the parallel net with partial connectivity has the highest loss, almost the double with respect to the other networks with parallel architecture even if the mean accuracy is comparable. Similarly, in the frequency domain, the networks with parallel architecture show the highest losses with respect to the other networks. For what concern the time domain, the lowest loss is obtained mostly by the sequential networks. In addition, on average the networks that include the GCN with full connectivity provide better performance than the ones with

Type of net		Accuracy [%]	Loss
Parallel	Partial Connectivity	**99 ± 0**	**0.01 ± 0.01**
	with fixed diagonal	99 ± 1	0.01 ± 0.01
	Total Connectivity	**99 ± 0**	**0.01 ± 0.01**
	with fixed diagonal	**99 ± 0**	**0.01 ± 0.01**
Sequential	Partial Connectivity	95 ± 10	0.07 ± 0.14
	with fixed diagonal	99 ± 1	0.02 ± 0.01
	Total Connectivity	97 ± 2	0.04 ± 0.02
	with fixed diagonal	**99 ± 0**	**0.01 ± 0.01**
Only GCN	Partial Connectivity	96 ± 2	0.04 ± 0.02
	with fixed diagonal	94 ± 4	0.07 ± 0.05
	Total Connectivity	74 ± 9	0.25 ± 0.07
	with fixed diagonal	94 ± 6	0.08 ± 0.07
LDA		83 ± 7	0.19 ± 0.08
Random Forest		98 ± 1	0.02 ± 0.01

Table 3.2 Summary of the subjects' mean accuracies and losses of the tested networks for the 3 macro classes in the frequency domain

only the partial brain-muscle connectivity.

To have a better view for an online application, the predictions in output from the classifiers are accumulated over time after the appearance of the cue using a linear integrator. In this way, the classification at the current time step depends also on the previous classifier outputs.

As shown in Figure 3.7 (right) and Figure 3.8 (right), the networks trained with features in the frequency domain present no or limited improvements over time, with an average improvement at the end of the trial of 3.1±0.8% and of 2.1±0.3% for the parallel and sequential networks on average.
On the other hand, the networks trained with features from the frequency domain benefit more from the accumulation over time, as shown in Figure 3.7 (left) and Figure 3.8 (left). For the parallel networks, a flex in the classification accuracy occurs after about 800 ms from the trial onset, with the partial

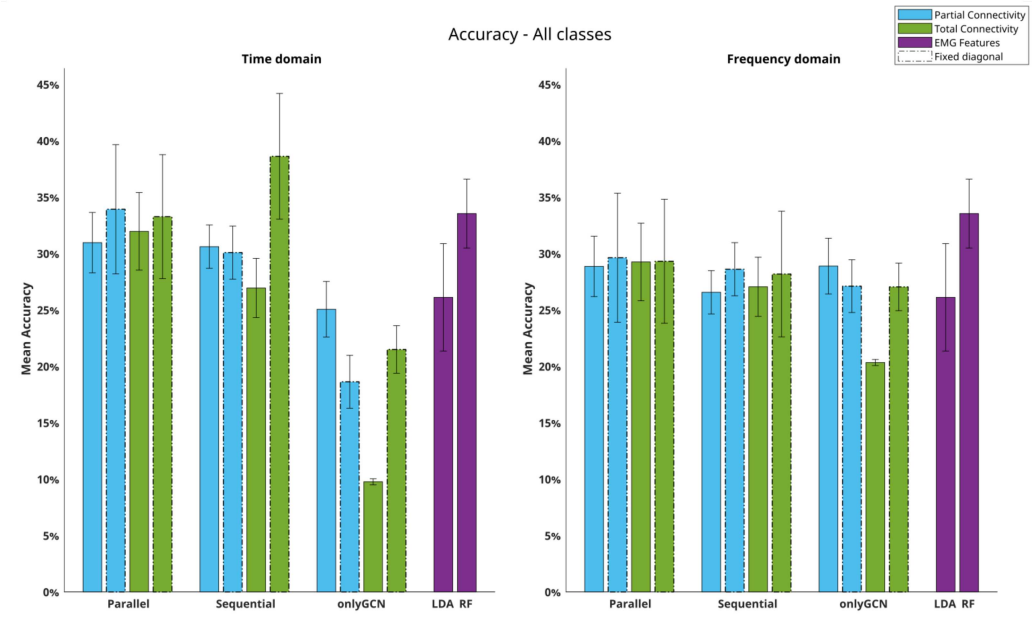

Figure 3.5 Subjects' mean accuracies of the trained networks predicting the 11 classes

connectivity and fixed diagonal that improved the recognition rate of about 10%. Thanks to the accumulation, the best performance are obtained by the sequential network with total connectivity and fixed diagonal trained with time domain features. Figure 3.11 illustrates the classification performance of the best network with respect to the average signal time course of the first EMG channel. The network shows a rapid improvement of the performance at 500 ms from 26% to 46% on average.

Looking at the upper part of Figure 3.11 the overall improvement in time is in the wrist rotation and the object grasping micro movements, which are the first and third groups, while for the arm reaching there is no much difference. In the evaluation at the end of the trial instead, the recognition rate for every micro movement improve, also managing to distinguish clearly between two movements in a case.

Depending on the weighted matrix diagonal, the network extract different

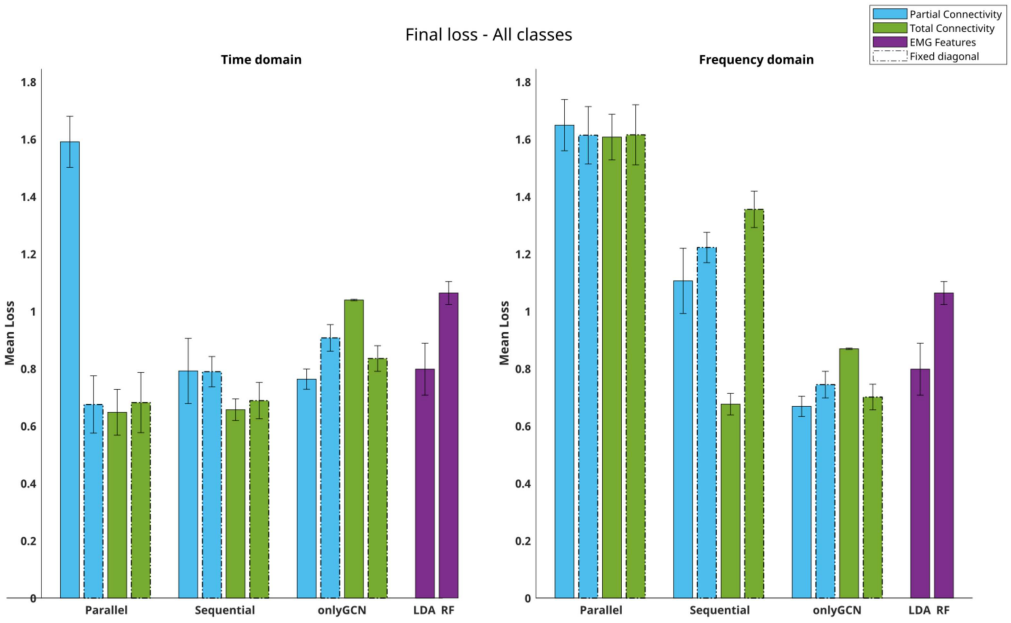

Figure 3.6 Subjects' mean final loss of the trained networks predicting the 11 classes

graph structures [Figure 3.12] due to their different weights values. Both the matrices are symmetrical as expected, and the input channels are present vertically, while the horizontal axis in composed by the first layer of graph nodes. The graph neurons utilize both the data that comes from the brain and from the muscles, managing to merge the multimodal data by itself.

Type of net		Accuracy [%]	Loss
Parallel	Partial Connectivity	31 ± 3	1.59 ± 0.09
	with fixed diagonal	34 ± 6	0.67 ± 0.10
	Total Connectivity	32 ± 3	0.65 ± 0.08
	with fixed diagonal	33 ± 5	0.68 ± 0.10
Sequential	Partial Connectivity	31 ± 2	0.79 ± 0.11
	with fixed diagonal	30 ± 2	0.79 ± 0.05
	Total Connectivity	27 ± 3	0.66 ± 0.04
	with fixed diagonal	**39 ± 6**	**0.69 ± 0.06**
Only GCN	Partial Connectivity	25 ± 2	0.76 ± 0.04
	with fixed diagonal	19 ± 2	0.91 ± 0.05
	Total Connectivity	10 ± 0	1.04 ± 0.00
	with fixed diagonal	21 ± 2	0.84 ± 0.04
LDA		26 ± 5	0.80 ± 0.09
Random Forest		34 ± 3	1.06 ± 0.04

Table 3.3 Summary of the subjects' accuracies and losses of the tested networks for the 11 classes in the time domain

Type of net		Accuracy [%]	Loss
Parallel	Partial Connectivity	29 ± 2	1.65 ± 0.11
	with fixed diagonal	30 ± 2	1.61 ± 0.09
	Total Connectivity	29 ± 2	1.61 ± 0.09
	with fixed diagonal	29 ± 2	1.62 ± 0.09
Sequential	Partial Connectivity	27 ± 5	1.11 ± 0.08
	with fixed diagonal	29 ± 1	1.22 ± 0.11
	Total Connectivity	27 ± 2	0.68 ± 0.03
	with fixed diagonal	28 ± 1	1.36 ± 0.09
Only GCN	Partial Connectivity	29 ± 3	0.67 ± 0.03
	with fixed diagonal	27 ± 2	0.74 ± 0.04
	Total Connectivity	20 ± 2	0.87 ± 0.07
	with fixed diagonal	27 ± 4	0.70 ± 0.06
LDA		26 ± 5	0.80 ± 0.09
Random Forest		**34 ± 3**	**1.06 ± 0.04**

Table 3.4 Summary of the subjects' accuracies and losses of the tested networks for the 11 classes in the frequency domain

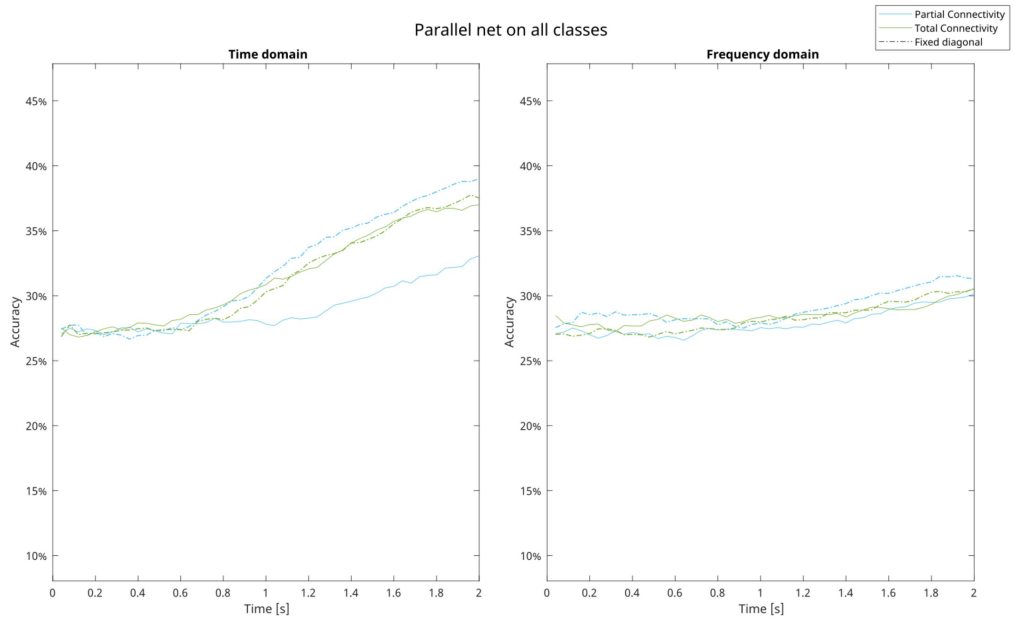

Figure 3.7 Varying subjects' mean accuracy by accumulating the prediction rates of the parallel network for the 11 classes

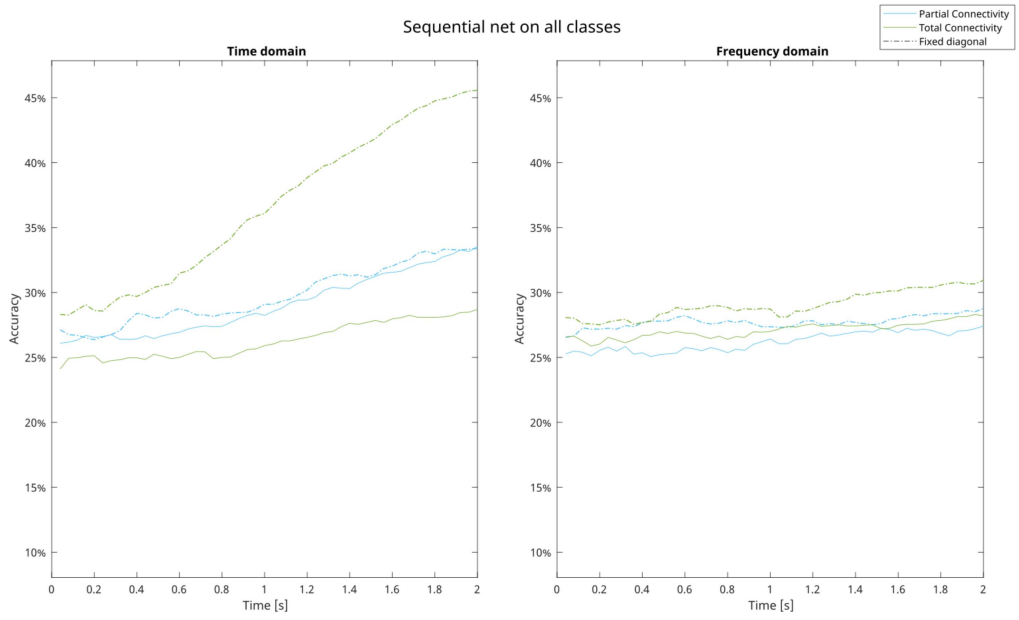

Figure 3.8 Varying subjects' mean accuracy by accumulating the prediction rates of the sequential network for the 11 classes

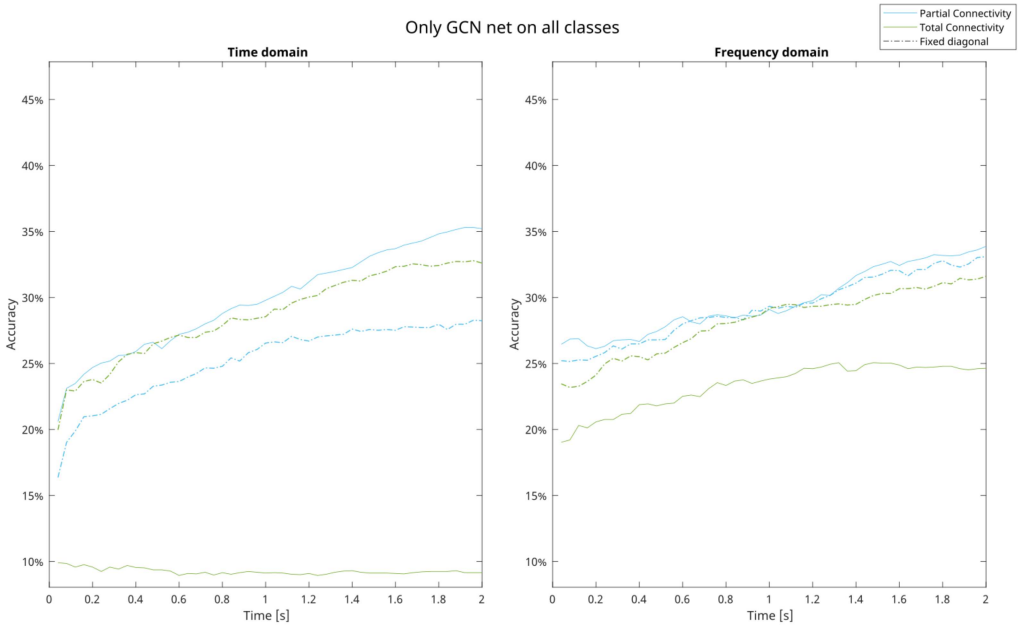

Figure 3.9 Varying subjects' mean accuracy by accumulating the prediction rates of the network with only a GCN layer for the 11 classes

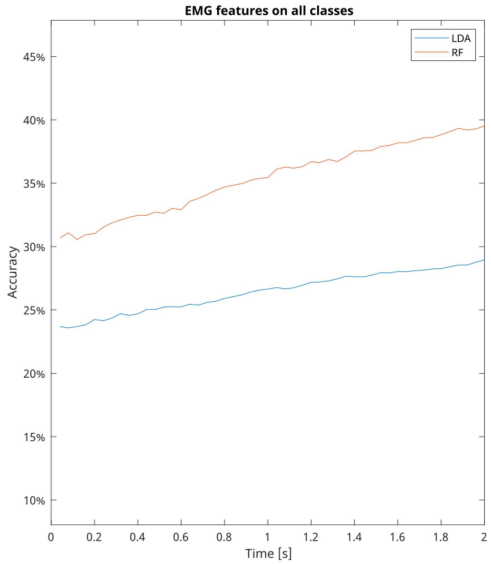

Figure 3.10 Varying subjects' mean accuracy by accumulating the prediction rates of the EMG features models for the 11 classes

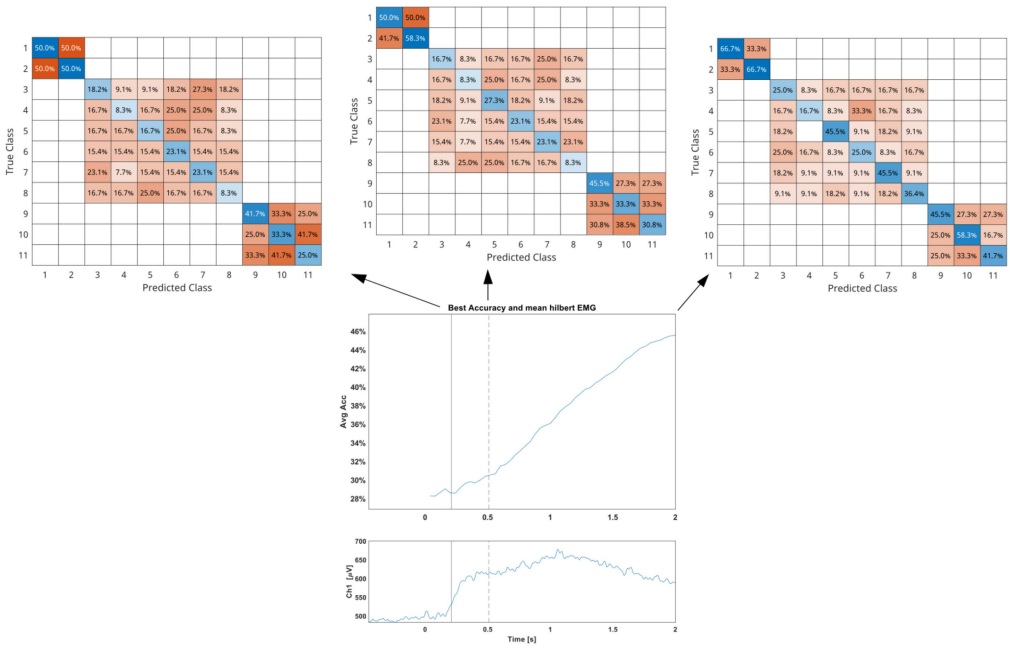

Figure 3.11 Upper part: subjects' mean confusion matrices of the predicted trial classes at different moments. The first is at 10% between the maximum and the minimum values of the EMG which correspond to the effective movement trigger. The second is after 300 ms from the previous point which is considered the maximum delay for online application [5] and the last is at the end of the considered trial length. The white spaces correspond to values of zero. Bottom part: best accumulating subjects' mean accuracy predictions for the 11 classes in comparison with the trials mean EMG Hilbert transformation of the first EMG channel

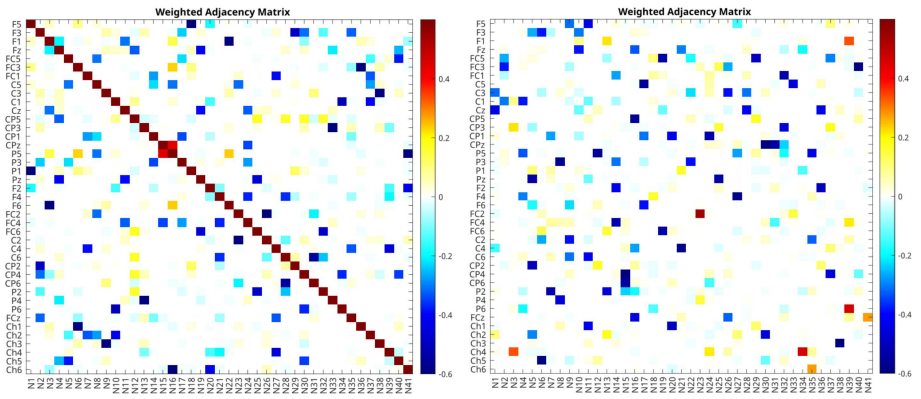

Figure 3.12 Normalized weighted adjacency matrix of the time input sequential net with and without fixed diagonal of one subject (subject 15) for the 11 classes

4 | Discussion

My thesis motivation was to create a new hybrid neural interface that combine neuroscience and deep learning using a weighted undirected graph to model the connectivity of the neuromuscular system.

Evaluating the accuracy for the networks in the three macro-movements classification scenario among the subjects [Figure 3.3], *i.e.* reaching, wrist twisting and grasping, it can be seen that the two most complex networks can recognize them clearly, but have some difficulties to classify the movements that belong to the same group as the random forest. My networks instead has less variance between the subjects and a lower loss than the RF, implying an higher confidence in the classes prediction. Overall, no significant differences in the classification performance have been found between the parallel and the sequential architecture, as well as if the network is trained with time or frequency domain features, even if with the time domain features a lower loss is obtained on average.

On the other hand, differences in the network configuration become clearer when the more complex classification scenario with all the 11 classes is considered. In this case, the network shows higher performance in the time domain rather than in the frequency domain. In addition, allowing the network to learn the total connectivity between the input channels increases the discriminant power of the proposed approach. My proposed approach

reaches or surpasses the state-of-the-art performances, in particular in the sample-to-sample accuracy and in the loss.

The common difficulty of distinguishing the micro-movement can be due to the electrodes position [Figure 2.6]. To have a better signal difference and a better recognition between the different reaching positions, but also between them and the grasping ones, other electrodes should be placed on the deltoid anterior, deltoid medial and deltoid posterior [104].

In general, the addition of memory-based layers, such as the GRU, significantly improves the accuracy of the GCN. Nevertheless, it is worth highlighting that for the networks trained with features from the frequency domain, the networks with only the GCN layer achieved comparable accuracy, and even lower loss than the use of deeper networks. This results are thanks to the presence of the cortico-muscular coherence in the dataset, that normally it can be evaluated doing an analysis in the frequency domain [105].

From the definition of the standard binary cross-entropy loss [106], it can be seen as a valuation of the predictions' confidence when the networks classifications are correct or, in the case they aren't, the loss evaluates how much the predictions are distant from the correct ones. This interpretation becomes more evident when the output of the network is accumulated over time. Networks and the machine learning methods with higher sample-to-sample classification loss have lower benefit from the accumulation. In fact, putting in correspondence the average net results, for example the parallel net [Figures 3.5, 3.6] with the accumulating ones [Figure 3.7], the accuracies start from the same level from their average recognition rate. In the input frequency domain and in one case in the time domain, the loss values are the highest and the evolution during the trial doesn't change so much, remaining almost constant or increasing by up to 5%. On the other hand, the proposed

sequential network with time domain features and total connectivity, that achieved the lowest loss, achieves an improvement in the accuracy of more than 20%, outperforming the state-of-the-art approaches.
There is a general increase of the loss with the frequency input in the complex networks than the simplest one. The time domain data contain multiple samples within the same window and so the GRUs are likely to be better able to identify discriminative temporal features. In contrast, in the frequency domain, to obtain the PSD we have decrease the temporal resolution in the windows, making it more difficult for GRUs to discriminate.

As the EMG signal suggests [Figure 3.2] the effective movements occur after 500 ms from the starting cue but in the EEG some movement-related cortical potentials are present for the motor preparation, in particular the negativity slope (NS) which is present about 400 ms before the motor execution [107, 108]. The improvement of the network performance before the motion onset can be explained by the possibility for the hybrid networks to exploit the pre-movement EEG correlates until the appearance of the EMG activity.
The parallel net [Figure 3.7] at 800 ms has an increase in the accuracy thanks to the support of the EMG that helps to increase the prediction rate for a specific movement during the duration of the trial. Instead, the sequential net [Figure 3.8] with the full weighted matrix and fixed diagonal succeeds to utilize the muscular features without any delay from the movement onset, improving the classification with a steeper curve. Interestingly, the networks that has only the GCN layer are the only ones characterized by a steeper increase of the classification accuracy about 100 ms before the motion onset [Figure 3.9]. Further exploration of the data and the network parameters should be performed in the future work to understand if this phenomenon is

effectively due to the recognition of a NS feature in the EEG data.
To conclude, the usage of multimodal data has effectively improved the classes predictions than the state-of-the-art methods that use unimodal data, with also the possibility to predict the user's intention before the motion onset.

To better understand what the GCN is learning, I looked at the weighted adjacency matrix obtained after the network training. Indeed, the adjacency can be seen as a proxy of the task-specific neuromuscular connectivity. The networks have successfully learnt by itself how to combine the brain and the muscle modalities. In the networks in which the diagonal of the adjacency matrix is fixed during learning, the network is forcing the nodes' self connections and only learns the relationship between each channel and the other channels, either EEG or EMG in the case of the total connectivity. In the case of the fixed diagonal, it can be noticed that in general the connections between EEG and EMG channels have generally negative weights, which can be interpreted as a negative correlation between the two signals. This aspect can be explained by the fact that the EEG signal was filtered in the frequency range of the sensory-motor rhythms. Indeed, the alpha and beta EEG rhythms decrease in power during the execution of a movement (*i.e.* ERD) [109, 110]. At the same time, a burst in the muscle activity signal generates an increase in the EMG signal power, allowing the graph to learn the negative correlation with the activity in the EEG channels. On the other hand, the adjacency matrix obtained when the diagonal is not fixed are harder to interpret. To give it a biological meaning for the neuromuscular system, its characteristics such as the corticomuscular coherence [111] or the brain-to-muscle connectivity [60] have to be evaluated on the dataset and compared with the adjacency matrix values. For this reason it's not explic-

itly comprehensible the weights meaning and their correlation in a biological context, and a deep exploration of the patterns obtained should be performed in future work, which goes beyond the scope of this thesis.

5 | Conclusions

In this thesis I proposed and developed a novel hHMI based on GCN layers that uses EEG and EMG data to create a graph structure that attempts to simulate the neuromuscular system and to classify upper limb movements.

The hHMIs have some design choices that can create very different results and designer-dependent structures. In particular where and when the data has to be fused and how it has to be done. For this reason the development of hybrid interfaces is very hard and time consuming. To resolve these problems, I propose a new hHMI structure that use a GCN layer to make the data merging automatic and try to extract some discriminant data characteristics in an almost end-to-end fashion.

The proposed approach overcomes the results [Figure 3.3, 3.5] of the random forest which is used as comparison method for the movement classification with the EMG signals.
The hHMI has automatically learnt the graph structure during the training [Figure 3.12] and it has successfully merged the EEG and the EMG signals due to the presence of brain and muscle weights for every GCN node. The graph edges can be correlated with the brain behaviour of neurons synchronization [110] during the movements.

Future developments will focus on the study of the weighted adjacency matrix explanation linked with the corticomuscular coherence [111], the brain-

to-brain connectivity [112] and the muscle synergies [113]. Then, an online application for devices control could be designed to evaluate the performance in a time control application.

Bibliography

[1] R. Sharma, V. I. Pavlović, and T. S. Huang, "Toward multimodal human–computer interface," in *Advances in image processing and understanding: A Festschrift for Thomas S Huang*, pp. 349–365, World Scientific, 2002.

[2] T. Bao, S. Q. Xie, P. Yang, P. Zhou, and Z. Zhang, "Towards robust, adaptive and reliable upper-limb motion estimation using machine learning and deep learning–a survey in myoelectric control," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2022.

[3] K. Liu, Y. Li, N. Xu, and P. Natarajan, "Learn to combine modalities in multimodal deep learning," *arXiv preprint arXiv:1805.11730*, 2018.

[4] J.-H. Jeong, J.-H. Cho, K.-H. Shim, B.-H. Kwon, B.-H. Lee, D.-Y. Lee, D.-H. Lee, and S.-W. Lee, "Multimodal signal dataset for 11 intuitive movement tasks from single upper extremity during multiple recording sessions," *GigaScience*, vol. 9, no. 10, p. giaa098, 2020.

[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] G. Johannsen, "Human-machine interaction," *Control Systems, Robotics and Automation*, vol. 21, pp. 132–62, 2009.

[7] S. Singh, "The role of speech technology in biometrics, forensics and man-machine interface.," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 9, no. 1, 2019.

[8] I. Echeverria-Altuna, A. J. Quinn, N. Zokaei, M. W. Woolrich, A. C. Nobre, and F. van Ede, "Transient beta activity and cortico-muscular connectivity during sustained motor behaviour," *Progress in Neurobiology*, vol. 214, p. 102281, 2022.

[9] O. Sporns, "Brain connectivity," *Scholarpedia*, vol. 2, no. 10, p. 4695, 2007.

[10] T. W. Boonstra, A. Danna-Dos-Santos, H.-B. Xie, M. Roerdink, J. F. Stins, and M. Breakspear, "Muscle networks: Connectivity analysis of emg activity during postural control," *Scientific reports*, vol. 5, no. 1, pp. 1–14, 2015.

[11] P. Rani, C. Liu, and N. Sarkar, "Affective feedback in closed loop human-robot interaction," in *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-robot Interaction*, pp. 335–336, 2006.

[12] M. Van Gerven, J. Farquhar, R. Schaefer, R. Vlek, J. Geuze, A. Nijholt, N. Ramsey, P. Haselager, L. Vuurpijl, S. Gielen, *et al.*, "The brain–computer interface cycle," *Journal of neural engineering*, vol. 6, no. 4, p. 041001, 2009.

[13] P. Wierzgała, D. Zapała, G. M. Wojcik, and J. Masiak, "Most popular signal processing methods in motor-imagery bci: a review and meta-analysis," *Frontiers in neuroinformatics*, vol. 12, p. 78, 2018.

[14] R. Abiri, S. Borhani, E. W. Sellers, Y. Jiang, and X. Zhao, "A comprehensive review of eeg-based brain–computer interface paradigms," *Journal of neural engineering*, vol. 16, no. 1, p. 011001, 2019.

[15] C. Guger, B. Z. Allison, and M. A. Lebedev, "Recent advances in brain-computer interface research—a summary of the bci award 2016 and bci research trends," *Brain-computer interface research*, pp. 127–134, 2017.

[16] T. Kirschstein and R. Köhling, "What is the source of the eeg?," *Clinical EEG and neuroscience*, vol. 40, no. 3, pp. 146–149, 2009.

[17] R. F. Helfrich and R. T. Knight, "Cognitive neurophysiology: event-related potentials," *Handbook of clinical neurology*, vol. 160, pp. 543–558, 2019.

[18] D. P. Subha, P. K. Joseph, R. Acharya U, C. M. Lim, *et al.*, "Eeg signal analysis: a survey," *Journal of medical systems*, vol. 34, no. 2, pp. 195–212, 2010.

[19] J. S. Kumar and P. Bhuvaneswari, "Analysis of electroencephalography (eeg) signals and its categorization–a study," *Procedia engineering*, vol. 38, pp. 2525–2536, 2012.

[20] R. Fazel-Rezai, B. Z. Allison, C. Guger, E. W. Sellers, S. C. Kleih, and A. Kübler, "P300 brain computer interface: current challenges and emerging trends," *Frontiers in neuroengineering*, p. 14, 2012.

[21] R. J. Barry, G. Z. Steiner, F. M. De Blasio, J. S. Fogarty, D. Karamacoska, and B. MacDonald, "Components in the p300: Don't forget the novelty p3!," *Psychophysiology*, vol. 57, no. 7, p. e13371, 2020.

[22] R. Kuś, T. Spustek, M. Zieleniewska, A. Duszyk, P. Rogowski, and P. Suffczyński, "Integrated trimodal ssep experimental setup for visual, auditory and tactile stimulation," *Journal of neural engineering*, vol. 14, no. 6, p. 066002, 2017.

[23] P. W. Ferrez and J. d. R. Millán, "Error-related eeg potentials generated during simulated brain–computer interaction," *IEEE transactions on biomedical engineering*, vol. 55, no. 3, pp. 923–929, 2008.

[24] C. Neuper, M. Wörtz, and G. Pfurtscheller, "Erd/ers patterns reflecting sensorimotor activation and deactivation," *Progress in brain research*, vol. 159, pp. 211–222, 2006.

[25] M. A. Lebedev and M. A. Nicolelis, "Brain-machine interfaces: From basic science to neuroprostheses and neurorehabilitation," *Physiological reviews*, vol. 97, no. 2, pp. 767–837, 2017.

[26] D. Steyrl, R. J. Kobler, G. R. Müller-Putz, *et al.*, "On similarities and differences of invasive and non-invasive electrical brain signals in brain-computer interfacing," *Journal of biomedical science and engineering*, vol. 9, no. 08, p. 393, 2016.

[27] H. Wang, F. Yan, T. Xu, H. Yin, P. Chen, H. Yue, C. Chen, H. Zhang, L. Xu, Y. He, *et al.*, "Brain-controlled wheelchair review: From wet electrode to dry electrode, from single modal to hybrid modal, from synchronous to asynchronous," *IEEE Access*, vol. 9, pp. 55920–55938, 2021.

[28] R. H. Chowdhury, M. B. Reaz, M. A. B. M. Ali, A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors*, vol. 13, no. 9, pp. 12431–12466, 2013.

[29] J. R. Daube and D. I. Rubin, "Needle electromyography," *Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine*, vol. 39, no. 2, pp. 244–270, 2009.

[30] C. J. De Luca, A. Adam, R. Wotiz, L. D. Gilmore, and S. H. Nawab, "Decomposition of surface emg signals," *Journal of neurophysiology*, vol. 96, no. 3, pp. 1646–1657, 2006.

[31] A. Phinyomark, A. Nuidod, P. Phukpattaranont, and C. Limsakul, "Feature extraction and reduction of wavelet transform coefficients for emg pattern classification," *Elektronika ir Elektrotechnika*, vol. 122, no. 6, pp. 27–32, 2012.

[32] K. Mahaphonchaikul, D. Sueaseenak, C. Pintavirooj, M. Sangworasil, and S. Tungjitkusolmun, "Emg signal feature extraction based on wavelet transform," in *ECTI-CON2010: The 2010 ECTI International Confernce on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, pp. 327–331, IEEE, 2010.

[33] H. F. Hassan, S. J. Abou-Loukh, and I. K. Ibraheem, "Teleoperated robotic arm movement using electromyography signal with wearable myo armband," *Journal of King Saud University-Engineering Sciences*, vol. 32, no. 6, pp. 378–387, 2020.

[34] A. H. Ali, "An investigation of electromyographic (emg) control of dextrous hand prostheses for transradial amputees," 2013.

[35] C. Altın, "Comparison of different time and frequency domain feature extraction methods on elbow gesture's emg," *European journal of interdisciplinary studies*, vol. 2, no. 3, pp. 35–44, 2016.

[36] C.-Y. Chang, S.-H. Hsu, L. Pion-Tonachini, and T.-P. Jung, "Evaluation of artifact subspace reconstruction for automatic eeg artifact removal," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1242–1245, IEEE, 2018.

[37] I. Kaya, "A brief summary of eeg artifact handling," *Brain-Computer Interface*, 2019.

[38] J. A. Urigüen and B. Garcia-Zapirain, "Eeg artifact removal—state-of-the-art and guidelines," *Journal of neural engineering*, vol. 12, no. 3, p. 031001, 2015.

[39] I. Elamvazuthi, N. Duy, Z. Ali, S. Su, M. A. Khan, and S. Parasuraman, "Electromyography (emg) based classification of neuromuscular disorders using multi-layer perceptron," *Procedia Computer Science*, vol. 76, pp. 223–228, 2015.

[40] E. K. Kalunga, S. Chevallier, O. Rabreau, and E. Monacelli, "Hybrid interface: Integrating bci in multimodal human-machine interfaces," in *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 530–535, IEEE, 2014.

[41] J. Shin, J. Kwon, and C.-H. Im, "A ternary hybrid eeg-nirs brain-computer interface for the classification of brain activation patterns during mental arithmetic, motor imagery, and idle state," *Frontiers in neuroinformatics*, vol. 12, p. 5, 2018.

[42] A. Jaimes and N. Sebe, "Multimodal human–computer interaction: A survey," *Computer vision and image understanding*, vol. 108, no. 1-2, pp. 116–134, 2007.

[43] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: an overview of methods, challenges, and prospects," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, 2015.

[44] X. Li, O. W. Samuel, X. Zhang, H. Wang, P. Fang, and G. Li, "A motion-classification strategy based on semg-eeg signal combination for upper-limb amputees," *Journal of neuroengineering and rehabilitation*, vol. 14, no. 1, pp. 1–13, 2017.

[45] K. Bakshi, R. Pramanik, M. Manjunatha, and C. Kumar, "Upper limb prosthesis control: A hybrid eeg-emg scheme for motion estimation in transhumeral subjects," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2024–2027, IEEE, 2018.

[46] F. Davarinia and A. Maleki, "Ssvep-gated emg-based decoding of elbow angle during goal-directed reaching movement," *Biomedical Signal Processing and Control*, vol. 71, p. 103222, 2022.

[47] K. Leerskov, M. Rehman, I. Niazi, S. Cremoux, and 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)*, pp. 403–410, IEEE, 2020.

[48] J. Tryon, E. Friedman, and A. L. Trejos, "Performance evaluation of eeg/emg fusion methods for motion classification," in *2019 IEEE*

16th International Conference on Rehabilitation Robotics (ICORR), pp. 971–976, IEEE, 2019.

[49] A. Riccio, E. M. Holz, P. Aricò, F. Leotta, F. Aloise, L. Desideri, M. Rimondini, A. Kübler, D. Mattia, and F. Cincotti, "Hybrid p300-based brain-computer interface to improve usability for people with severe motor disability: electromyographic signals for error correction during a spelling task," *Archives of physical medicine and rehabilitation*, vol. 96, no. 3, pp. S54–S61, 2015.

[50] R. Leeb, H. Sagha, R. Chavarriaga, and J. del R Millán, "A hybrid brain–computer interface based on the fusion of electroencephalographic and electromyographic activities," *Journal of neural engineering*, vol. 8, no. 2, p. 025011, 2011.

[51] R. Leeb, H. Sagha, R. Chavarriaga, *et al.*, "Multimodal fusion of muscle and brain signals for a hybrid-bci," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 4343–4346, IEEE, 2010.

[52] N. Hooda, R. Das, and N. Kumar, "Fusion of eeg and emg signals for classification of unilateral foot movements," *Biomedical Signal Processing and Control*, vol. 60, p. 101990, 2020.

[53] C. Cui, G.-B. Bian, Z.-G. Hou, J. Zhao, and H. Zhou, "A multimodal framework based on integration of cortical and muscular activities for decoding human intentions about lower limb motions," *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 4, pp. 889–899, 2017.

[54] S. Liaqat, K. Dashtipour, K. Arshad, K. Assaleh, and N. Ramzan, "A hybrid posture detection framework: Integrating machine learning and deep neural networks," *IEEE Sensors Journal*, vol. 21, no. 7, pp. 9515–9522, 2021.

[55] R. Rajapriya, K. Rajeswari, and S. Thiruvengadam, "Deep learning and machine learning techniques to improve hand movement classification in myoelectric control system," *Biocybernetics and Biomedical Engineering*, vol. 41, no. 2, pp. 554–571, 2021.

[56] S. Tortora, L. Tonin, C. Chisari, S. Micera, E. Menegatti, and F. Artoni, "Hybrid human-machine interface for gait decoding through bayesian fusion of eeg and emg classifiers," *Frontiers in Neurorobotics*, vol. 14, p. 582728, 2020.

[57] A. M. Chiarelli, P. Croce, A. Merla, and F. Zappasodi, "Deep learning for hybrid eeg-fnirs brain–computer interface: application to motor imagery classification," *Journal of neural engineering*, vol. 15, no. 3, p. 036028, 2018.

[58] F. Chollet, "The limitations of deep learning," *Deep learning with Python*, 2017.

[59] A. B. Said, A. Mohamed, T. Elfouly, K. Harras, and Z. J. Wang, "Multimodal deep learning approach for joint eeg-emg data compression and classification," in *2017 IEEE wireless communications and networking conference (WCNC)*, pp. 1–6, IEEE, 2017.

[60] F. Artoni, C. Fanciullacci, F. Bertolucci, A. Panarese, S. Makeig, S. Micera, and C. Chisari, "Unidirectional brain to muscle connectivity

reveals motor cortex control of leg muscles during stereotyped walking," *Neuroimage*, vol. 159, pp. 403–416, 2017.

[61] M. Rana, M. S. Yani, S. Asavasopon, B. E. Fisher, and J. J. Kutch, "Brain connectivity associated with muscle synergies in humans," *Journal of Neuroscience*, vol. 35, no. 44, pp. 14708–14716, 2015.

[62] J. Zhang, B. Wang, C. Zhang, Y. Xiao, and M. Y. Wang, "An eeg/emg/eog-based multimodal human-machine interface to real-time control of a soft robot hand," *Frontiers in neurorobotics*, vol. 13, p. 7, 2019.

[63] O. Ali, M. Saif-ur Rehman, T. Glasmachers, I. Iossifidis, and C. Klaes, "Contranet: A single end-to-end hybrid network for eeg-based and emg-based human machine interfaces," *arXiv preprint arXiv:2206.10677*, 2022.

[64] A. Chowdhury, A. Dutta, and G. Prasad, "Corticomuscular co-activation based hybrid brain-computer interface for motor recovery monitoring," *IEEE Access*, vol. 8, pp. 174542–174557, 2020.

[65] W. Jin, Y. Ma, X. Liu, X. Tang, S. Wang, and J. Tang, "Graph structure learning for robust graph neural networks," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 66–74, 2020.

[66] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021.

[67] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[68] V. Nasteski, "An overview of the supervised machine learning methods," *Horizons. b*, vol. 4, pp. 51–62, 2017.

[69] X. Wang, X. Lin, and X. Dang, "Supervised learning in spiking neural networks: A review of algorithms and evaluations," *Neural Networks*, vol. 125, pp. 258–280, 2020.

[70] A. I. Georgevici and M. Terblanche, "Neural networks and deep learning: a brief introduction," 2019.

[71] M. H. Sazli, "A brief review of feed-forward neural networks," *Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering*, vol. 50, no. 01, 2006.

[72] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, "Recent advances in recurrent neural networks," *arXiv preprint arXiv:1801.01078*, 2017.

[73] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[74] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[75] Q. V. Le *et al.*, "A tutorial on deep learning part 1: Nonlinear classifiers and the backpropagation algorithm," *Mountain View, CA*, 2015.

[76] J. H. Lee, T. Delbruck, and M. Pfeiffer, "Training deep spiking neural networks using backpropagation," *Frontiers in neuroscience*, vol. 10, p. 508, 2016.

[77] E. Haber and L. Ruthotto, "Stable architectures for deep neural networks," *Inverse problems*, vol. 34, no. 1, p. 014004, 2017.

[78] K. Fukushima, "A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol, Cybern*, vol. 36, pp. 193–202, 1980.

[79] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[80] N. A. Asif, Y. Sarker, R. K. Chakrabortty, M. J. Ryan, M. H. Ahamed, D. K. Saha, F. R. Badal, S. K. Das, M. F. Ali, S. I. Moyeen, M. R. Islam, and Z. Tasneem, "Graph neural network: A comprehensive review on non-euclidean space," *IEEE Access*, vol. 9, pp. 60588–60606, 2021.

[81] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.

[82] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[83] L. Franceschi, M. Niepert, M. Pontil, and X. He, "Learning discrete structures for graph neural networks," in *International conference on machine learning*, pp. 1972–1982, PMLR, 2019.

[84] B. Fatemi, L. El Asri, and S. M. Kazemi, "Slaps: Self-supervision improves structure learning for graph neural networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 22667–22681, 2021.

[85] G. Wen, P. Cao, H. Bao, W. Yang, T. Zheng, and O. Zaiane, "Mvs-gcn: A prior brain structure learning-guided multi-view graph convolution network for autism spectrum disorder diagnosis," *Computers in Biology and Medicine*, vol. 142, p. 105239, 2022.

[86] M. Jiang, G. Liu, Y. Su, and X. Wu, "Gcn-sl: Graph convolutional networks with structure learning for graphs under heterophily," *arXiv preprint arXiv:2105.13795*, 2021.

[87] X. Xi, X. Wu, Y.-B. Zhao, J. Wang, W. Kong, and Z. Luo, "Cortico-muscular functional network: an exploration of cortico-muscular coupling in hand movements," *Journal of Neural Engineering*, vol. 18, no. 4, p. 046084, 2021.

[88] F. Tecchio, C. Porcaro, F. Zappasodi, A. Pesenti, M. Ercolani, and P. M. Rossini, "Cortical short-term fatigue effects assessed via rhythmic brain–muscle coherence," *Experimental brain research*, vol. 174, no. 1, pp. 144–151, 2006.

[89] M. Hassan, O. Dufor, I. Merlet, C. Berrou, and F. Wendling, "Eeg source connectivity analysis: from dense array recordings to brain networks," *PloS one*, vol. 9, no. 8, p. e105041, 2014.

[90] S. Daud and R. Sudirman, "Butterworth bandpass and stationary wavelet transform filter comparison for electroencephalography signal," in *2015 6th international conference on intelligent systems, modelling and simulation*, pp. 123–126, IEEE, 2015.

[91] M. Atzori, A. Gijsberts, I. Kuzborskij, S. Elsig, A.-G. M. Hager, O. Deriaz, C. Castellini, H. Müller, and B. Caputo, "Characterization of a benchmark database for myoelectric movement classification," *IEEE*

transactions on neural systems and rehabilitation engineering, vol. 23, no. 1, pp. 73–83, 2014.

[92] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek, "Optimizing the channel selection and classification accuracy in eeg-based bci," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 6, pp. 1865–1873, 2011.

[93] S. Tsuchimoto, S. Shibusawa, S. Iwama, M. Hayashi, K. Okuyama, N. Mizuguchi, K. Kato, and J. Ushiba, "Use of common average reference and large-laplacian spatial-filters enhances eeg signal-to-noise ratios in intrinsic sensorimotor activity," *Journal of Neuroscience Methods*, vol. 353, p. 109089, 2021.

[94] S. Tortora, S. Ghidoni, C. Chisari, S. Micera, and F. Artoni, "Deep learning-based bci for gait decoding from eeg with lstm recurrent neural network," *Journal of neural engineering*, vol. 17, no. 4, p. 046011, 2020.

[95] G. Gómez-Herrero, "Automatic artifact removal (aar) toolbox v1. 3 (release 09.12. 2007) for matlab," *Tampere University of Technology*, 2007.

[96] J. A. Palmer, K. Kreutz-Delgado, B. D. Rao, and S. Makeig, "Modeling and estimation of dependent subspaces with non-radially symmetric and skewed densities," in *International Conference on Independent Component Analysis and Signal Separation*, pp. 97–104, Springer, 2007.

[97] L. Pion-Tonachini, K. Kreutz-Delgado, and S. Makeig, "Iclabel: An automated electroencephalographic independent component classifier, dataset, and website," *NeuroImage*, vol. 198, pp. 181–197, 2019.

[98] M. Atzori, A. Gijsberts, B. Caputo, and H. Müller, "Natural control capabilities of robotic hands by hand amputated subjects," in *2014 36th*

Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 4362–4365, IEEE, 2014.

[99] Z. Wang and X. Qian, "Text categorization based on lda and svm," in *2008 International Conference on Computer Science and Software Engineering*, vol. 1, pp. 674–677, IEEE, 2008.

[100] Y. Liu, Y. Wang, and J. Zhang, "New machine learning algorithm: Random forest," in *International Conference on Information Computing and Applications*, pp. 246–252, Springer, 2012.

[101] S. Pancholi and A. M. Joshi, "Portable emg data acquisition module for upper limb prosthesis application," *IEEE Sensors Journal*, vol. 18, no. 8, pp. 3436–3443, 2018.

[102] A. Gijsberts, M. Atzori, C. Castellini, H. Müller, and B. Caputo, "Movement error rate for evaluation of machine learning methods for semg-based hand movement classification," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 22, no. 4, pp. 735–744, 2014.

[103] G. Lu, J.-S. Brittain, P. Holland, J. Yianni, A. L. Green, J. F. Stein, T. Z. Aziz, and S. Wang, "Removing ecg noise from surface emg signals using adaptive filtering," *Neuroscience letters*, vol. 462, no. 1, pp. 14–19, 2009.

[104] I. Batzianoulis, S. El-Khoury, E. Pirondini, M. Coscia, S. Micera, and A. Billard, "Emg-based decoding of grasp gestures in reaching-to-grasping motions," *Robotics and Autonomous Systems*, vol. 91, pp. 59–70, 2017.

[105] R. Krauth, J. Schwertner, S. Vogt, S. Lindquist, M. Sailer, A. Sickert, J. Lamprecht, S. Perdikis, T. Corbet, J. d. R. Millán, *et al.*, "Corticomuscular coherence is reduced acutely post-stroke and increases bilaterally during motor recovery: a pilot study," *Frontiers in Neurology*, vol. 10, p. 126, 2019.

[106] Y. Ho and S. Wookey, "The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling," *IEEE Access*, vol. 8, pp. 4806–4813, 2019.

[107] K. Wang, M. Xu, Y. Wang, S. Zhang, L. Chen, and D. Ming, "Enhance decoding of pre-movement eeg patterns for brain–computer interfaces," *Journal of neural engineering*, vol. 17, no. 1, p. 016033, 2020.

[108] M. Hallett, "Movement-related cortical potentials.," *Electromyography and clinical neurophysiology*, vol. 34, no. 1, pp. 5–13, 1994.

[109] N. Erbil and P. Ungan, "Changes in the alpha and beta amplitudes of the central eeg during the onset, continuation, and offset of long-duration repetitive hand movements," *Brain research*, vol. 1169, pp. 44–56, 2007.

[110] Y. Jeon, C. S. Nam, Y.-J. Kim, and M. C. Whang, "Event-related (de) synchronization (erd/ers) during motor imagery tasks: Implications for brain–computer interfaces," *International Journal of Industrial Ergonomics*, vol. 41, no. 5, pp. 428–436, 2011.

[111] J. Liu, Y. Sheng, and H. Liu, "Corticomuscular coherence and its applications: A review," *Frontiers in human neuroscience*, vol. 13, p. 100, 2019.

[112] L. Brusini, F. Stival, F. Setti, E. Menegatti, G. Menegaz, and S. F. Storti, "A systematic review on motor-imagery brain-connectivity-based computer interfaces," *IEEE Transactions on Human-Machine Systems*, 2021.

[113] C. W. Antuvan, F. Bisio, F. Marini, S.-C. Yen, E. Cambria, and L. Masia, "Role of muscle synergies in real-time classification of upper limb motions using extreme learning machines," *Journal of neuroengineering and rehabilitation*, vol. 13, no. 1, pp. 1–15, 2016.

[114] S. Vaid, P. Singh, and C. Kaur, "Eeg signal analysis for bci interface: A review," in *2015 fifth international conference on advanced computing & communication technologies*, pp. 143–147, IEEE, 2015.

[115] G. Placidi, L. Cinque, and M. Polsinelli, "A fast and scalable framework for automated artifact recognition from eeg signals represented in scalp topographies of independent components," *Computers in Biology and Medicine*, vol. 132, p. 104347, 2021.

[116] M. Hamedi, S.-H. Salleh, and A. M. Noor, "Electroencephalographic motor imagery brain connectivity analysis for bci: A review," *Neural Computation*, vol. 28, no. 6, pp. 999–1041, 2016.

[117] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: a comprehensive review," *Computational Social Networks*, vol. 6, no. 1, pp. 1–23, 2019.

[118] E. T. Bullmore and D. S. Bassett, "Brain graphs: graphical models of the human brain connectome," *Annual review of clinical psychology*, vol. 7, pp. 113–140, 2011.

[119] M. T. Jurkiewicz, W. C. Gaetz, A. C. Bostan, and D. Cheyne, "Post-movement beta rebound is generated in motor cortex: evidence from neuromagnetic recordings," *Neuroimage*, vol. 32, no. 3, pp. 1281–1289, 2006.

[120] J. R. Wolpaw, N. Birbaumer, W. J. Heetderks, D. J. McFarland, P. H. Peckham, G. Schalk, E. Donchin, L. A. Quatrano, C. J. Robinson, T. M. Vaughan, *et al.*, "Brain-computer interface technology: a review of the first international meeting," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 2, pp. 164–173, 2000.

[121] W. Lai and H. Huosheng, "Towards multimodal human-machine interface for hands-free control: A survey," *Techn. Ber., School of Computer Science & Electronic Engineering, University of Essex*, 2011.

[122] P. Zhang and J. Zhang, "Deep learning analysis based on multi-sensor fusion data for hemiplegia rehabilitation training system for stoke patients," *Robotica*, vol. 40, no. 3, pp. 780–797, 2022.

[123] E. Rocon, J. Gallego, L. Barrios, A. Victoria, J. Ibánez, D. Farina, F. Negro, J. L. Dideriksen, S. Conforto, T. D'Alessio, *et al.*, "Multimodal bci-mediated fes suppression of pathological tremor," in *2010 annual international conference of the IEEE engineering in medicine and biology*, pp. 3337–3340, IEEE, 2010.

[124] W. Feng, N. Guan, Y. Li, X. Zhang, and Z. Luo, "Audio visual speech recognition with multimodal recurrent neural networks," in *2017 International Joint Conference on neural networks (IJCNN)*, pp. 681–688, IEEE, 2017.

[125] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

Acknowledgements

This work crowns the end of my master's degree in bioengineering which could never have happened without the people who have been close to me. I wish to express my sincere thanks to my supervisor Professor Manfredo Atzori for giving me the opportunity to work on and explore a new topic in a field that fascinated me. I would also like to show my gratitude to my co-supervisor Stefano Tortora for his advice and suggestions that have guided me in every step of the way through this thesis. Thanks should also go to my family for supporting me during these years, and to my girlfriend Alice for her encouragement and for always being there for me. Finally thanks to all the friends and the good times spent together over the years.