Deep Learning Methods for Hand Gesture Recognition via High-Density Surface Electromyogram (HD-sEMG) Signals

Mansoorehsadat Montazerin

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Electrical and Computer Engineering) at

Concordia University

Montréal, Québec, Canada

September 2023

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mansoorehsadat Montazerin**

Entitled: **Deep Learning Methods for Hand Gesture Recognition via High-Density Surface Electromyogram (HD-sEMG) Signals**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Habib Benali

_____ Examiner
Dr. Wei-Ping Zhu

_____ Supervisor
Dr. Arash Mohammadi

_____ Co-supervisor
Dr. Farnoosh Naderkhani

Approved by _____
Dr. Yousef R. Shayan, Chair
Department of Electrical and Computer Engineering

_____ 2023

_____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Deep Learning Methods for Hand Gesture Recognition via High-Density Surface Electromyogram (HD-sEMG) Signals

Mansoorehsadat Montazerin

Hand Gesture Recognition (HGR) using surface Electromyogram (sEMG) signals can be considered as one of the most important technologies in making efficient Human Machine Interface (HMI) systems. In particular, sEMG-based hand gesture has been a topic of growing interest for development of assistive systems to improve the quality of life in individuals suffering from amputated limbs. Generally speaking, myoelectric prosthetic devices work by classifying existing patterns of the collected sEMG signals and synthesizing intended gestures. While conventional myoelectric control systems, e.g., on/off control or direct-proportional, have potential advantages, challenges such as limited Degree of Freedom (DoF) due to crosstalk have resulted in the emergence of data-driven solutions. More specifically, to improve efficiency, intuitiveness, and the control performance of hand prosthetic systems, several Artificial Intelligence (AI) algorithms ranging from conventional Machine Learning (ML) models to highly complicated Deep Neural Network (DNN) architectures have been designed for sEMG-based hand gesture recognition in myoelectric prosthetic devices. In this thesis, we, first, perform a literature review on hand gesture recognition methods and elaborate on the recently proposed Deep Learning/Machine Learning (DL/ML) models in the literature. Then, our utilized High-Density sEMG (HD-sEMG) dataset is introduced and the rationales behind our main focus on this particular type of sEMG dataset are explained. We, then, develop a Vision Transformer (ViT)-based model [1] for gesture recognition with HD-sEMG signals and evaluate its performance under different conditions such as variable window sizes, number of electrode channels, and model's complexity. We compare its performance with that of two conventional ML and one DL algorithm that are typically adopted in this domain. Furthermore, we introduce another capability of our proposed framework for instantaneous training, which is its ability to classify hand gestures based on a single frame of HD-sEMG dataset. Following

that, we introduce the idea of integrating the macroscopic and microscopic neural drive information obtained from HD-sEMG data into a hybrid ViT-based framework for gesture recognition, which outperforms a standalone ViT architecture in terms of classification accuracy. Here, microscopic neural drive information (also called Motor Unit Spike Trains) refers to the neural commands sent by the brain and spinal cord to individual muscle fibers and are extracted from HD-sEMG signals using Blind Source Separation (BSP) algorithms. Finally, we design an alternative and novel hand gesture recognition model based on the less-explored topic of Spiking Neural Networks (SNN), which performs spatio-temporal gesture recognition in an event-based fashion. As opposed to the classical DNN architectures, SNNs are of the capacity to imitate human brain's cognitive function by using biologically inspired models of neurons and synapses. Therefore, they are more biologically explainable and computationally efficient.

Acknowledgments

First of all, I am literally grateful to my considerate supervisors, Dr. Arash Mohammadi and Dr. Farnoosh Naderkhani, for their exceptional mentorship and guidance throughout the process of crafting this thesis. Their insightful feedbacks, patience, and unwavering belief in my abilities have been pivotal in shaping the course of this research work. I was very fortunate to have had the privilege of working under their supervision, and I extend my heartfelt thanks for their constant encouragement and support. I would also like to express my heartfelt appreciation to my supervisors for their invaluable support which played a significant role in my acceptance into a PhD program I always dreamed for.

In addition, I would like to extend my gratitude to our collaborators, Dr. Seyed Farokh Atashzar, Dr. Svetlana Yanushkevich and Dr. Hamid Alinejad-Rokny, for their valuable contributions and commitment in carefully reading my papers and giving me their professional comments and constructive feedbacks. This has significantly improved the quality of my research works and taught me the how-tos of conducting scientific research.

I would also like to thank my labmates, Parastoo, Sadaf, Elahe and Soheil, whose generous support and insightful discussions provided me with a friendly and professional environment in which I improved my research and coding skills.

Finally, my deepest thanks goes to my family, whose endless love, genuine encouragement and constant belief in me have been the foundation upon which I have built my aspirations and accomplishments. I am always appreciative of the love and support you are giving me!

Contents

List of Abbreviations

Abbreviation	Description
2D	2-Dimensional
3D	3-Dimensional
AI	Artificial Intelligence
AR	Augmented Reality
BSP	Biological Signal Processing
BSS	Blind Source Separation
CCE	Categorical Cross-Entropy
CNN	Convolutional Neural Network
CPH	Cyber-Physical Human
DL	Deep Learning
DNN	Deep Neural Network
DoF	Degree of Freedom
DVS	Dynamic Vision Sensor
FC	Fully Connected
FLOP	Floating Point Operation
gCKC	gradient Convolution Kernel Compensation
HD-sEMG	High Density surface Electromyogram
HGR	Hand Gesture Recognition
HMI	Human Machine Interface
ICA	Independent Component Analysis

IQR	Interquartile Range
kNN	k-Nearest Neighbors
LDA	Linear Discriminant Analysis
LIF	Leaky Integrate and Fire
LSTM	Long Short-Term Memory
MAV	Mean Absolute Value
MCC	Matthew's Correlation Coefficient
ML	Machine Learning
MLP	Multilayer Perceptron
MMSE	Minimum Mean Squared Error
MSA	Multi-Head Self Attention
MU	Motor Unit
MUAP	Motor Unit Action Potential
MUST	Motor Unit Spike Train
NA	Not Applicable
PSD	Power Spectral Density
RMS	Root Mean Square
RNN	Recurrent Neural Network
sEMG	surface Electromyogram
SNN	Spiking Neural Network
SSC	Slope Sign Change
STA	Spike-Triggered Averaging
STD	Standard Deviation
SVM	Support Vector Machine
ViT	Vision Transformer
VR	Virtual Reality
WL	Waveform Length
ZC	Zero Crossings

List of Figures

List of Tables

Chapter 1

Thesis Overview

Hand Gesture Recognition (HGR) is a cutting-edge technology that has proven to have a significant effect on revolutionizing Human Machine Interface (HMI) systems. Thanks to the recent emergence of Artificial Intelligence (AI), sophisticated algorithms and advanced Machine Learning (ML) techniques have been developed to enable computers to automatically interpret and learn the gestures made by individuals, transforming them into meaningful commands or responses. This innovative field has caught considerable attention across various industries, ranging from Augmented/Virtual Reality (AR/VR) to robotics, exoskeletons and smart devices. As technology progresses, the ability to recognize hand gestures automatically opens the door to a more natural and intuitive way of communicating with machines, bridging the gap between human expressions and digital interfaces. This introductory paragraph only scratches the surface of the vast potential and exciting possibilities that hand gesture recognition holds for the future.

Owing to the development of state-of-the-art AI algorithms, more and more learning-based models have been proposed in the literature to perform automatic HGR with the aim of developing real-time, robust and user adaptable HMI devices with low latency. Cutting-edge advancements in Deep Learning (DL), neural networks, and data augmentation techniques are further pushing the boundaries of gesture recognition systems, propelling them toward even higher levels of performance and real-world applicability. However, gesture recognition still faces several challenges that need to be addressed to enhance its practicality and efficiency. Among the most important challenges are data scarcity which happens when the amount of labelled data for each gesture is not

enough for the DL model to learn them effectively. Furthermore, achieving real-time performance in resource-constrained systems, such as smartphones or wearable devices, can be demanding due to the computational complexity of some ML/DL algorithms. Some hand gestures, also, may have ambiguous interpretations and patterns leading to inaccurate gesture classification especially when the utilized dataset has a vast number of gestures with multiple degrees of freedom (DoF).

Accordingly, a large number of strategies are adopted with respect to the technology, data collection and algorithm design of HGR to solve the above-mentioned issues. Collecting comprehensive datasets, adopting data augmentation techniques, suggesting different pre-processing methods on the raw dataset and continuously exploring and fine-tuning existing ML/DL models are the common approaches mostly used in the literature.

1.1 Thesis Objectives

In brief, this thesis focuses mainly on development of automatic DL-based frameworks for hand gesture recognition via HD-sEMG signals using the following two different approaches:

1.1.1 Gesture Recognition based on Macroscopic Neural Drive Information

Generally speaking, hand gesture recognition has been investigated in the literature through the following two main directions: (i) The Vision-based approach in which RGB or depth cameras are used to track and recognize different hand gestures by analyzing the visual appearance of hands, and; (ii) The Sensor-based approach in which the signals related to position, orientation and movement of the hands are recorded through a set of touchless (e.g., infrared or ultrasonic) or touch-based (e.g., Electromyography (EMG) electrodes) sensors [3]. According to [4–6], the vision-based methods, compared to their sensor-based counterparts, often suffer from the following drawbacks: (a) Requiring excessive preprocessing and segmentation steps; (b) Being sensitive to the environment where the signals are being recorded, and; (c) Having higher latency and response time due to indirect estimation of the physical properties of various hand movements. Therefore, in this thesis, our focus is only on the sensor-based approach of gesture recognition and we dive specifically into the touch-based HD-sEMG electrodes that comprise a two-dimensional (2D) grid of densely placed

electrodes recording the electrical activity of the muscle's Motor Unit Action Potentials (MUAPs) in response to the neural signals.

HD-sEMG signals are considered as macroscopic neural drive information that collect the overall neural input sent from the central nervous system (brain and spinal cord) to the muscle's surface [7, 8]. These signals are commonly used to record the electrical signals from multiple muscles simultaneously. By studying sEMG signals, researchers can gain insights into muscle activation patterns and the timing of muscle recruitment during different hand gestures. In the recent literature, there has been a vast research on various ML/DL models for automated or semi-automated gesture recognition from HD-sEMG signals. These models span from traditional ML models that require a handcrafted feature extraction process as in References [9–12] to simpler or more complex end-to-end DL architectures such as Convolutional [13, 14] and Recurrent [15] Neural Networks (CNNs and RNNs), Transformers [16, 17] and hybrid architectures [18]. Despite their demonstrated success in many applications of hand gesture recognition, the aforementioned models suffer from major drawbacks such as not fully exploiting the temporal, spatial and neurophysiological characteristics of sEMG signals or being computationally complex and expensive. Moreover, they often need large amounts of data to generalize well and their overall training time and resource requirements make them unviable to be used in real-time HMI systems. Therefore, in order to tackle many of the above-mentioned issues with the existing hand gesture recognition models, we propose two distinct DNN architectures for gesture recognition using macroscopic neural drive information. First of all, we capitalize on the recent breakthrough role of the transformer architecture by introducing a standalone Vision Transformer (ViT)-based architecture which can accurately classify a large number of hand gestures from scratch without any need for data augmentation and/or transfer learning. Thanks to their parallelized structure and the underlying attention mechanism, ViTs have less training time and consume smaller system's memory with fewer number of trainable parameters. Since HD-sEMG datasets have a 3-Dimensional (3D) structure (one dimension in time and two dimensions in space), Vision Transformers (ViT) [19] can be considered as an appropriate architecture to be applied on them. The proposed ViT-based architecture is evaluated in terms of its classification accuracy, training time, testing time, memory usage and the number of trainable parameters using

various settings of the input signals. Our proposed framework is also compared with two conventional ML models each fed with two different sets of features and a 3D CNN model which is a Neural Network typically applied on 3D data.

In our second experiment, we develop an alternative and novel hand gesture recognition model based on the less-explored topic of Spiking Neural Networks (SNN), which performs spatio-temporal gesture recognition in an event-based fashion [20, 21]. An event-based processing approach, refers to a type of data processing in which the system is susceptible to the occurrence of events rather than the static input [22]. Unlike traditional neural networks, which use continuous-valued activation functions and propagate information through real-valued weights, SNNs operate on discrete events that represent the timing and rate of neuron's firing. SNNs imitate human brain's cognitive function by using biologically inspired models of neurons and synapses [21]. Accordingly, SNNs become more biologically explainable and computationally efficient requiring remarkably less amount of memory and processing units for their even-triggered processing and low-precision computation [22, 23]. SNNs have been the topic of interest for many computer vision-related tasks such as image classification [24], object tracking [25] and gesture recognition [21]. However, there is a limited number of works [26–28] on utilizing SNNs for EMG-based hand gesture recognition. therefore, in this part, we focus on developing a light (compact) two-layer MLP model with Leaky Integrate and Fire (LIF) spiking neurons to classify a set of 1 DoF gestures via HD-sEMG signals. In our work, HD-sEMG signals are normalized, windowed and fed to the spiking MLP model. This is a more straight-forward approach in comparison with using energy-density maps, spike coding and feature extraction as in [26, 28] to provide inputs for SNN architectures. We show that by considering each sample in the HD-sEMG dataset as a single time step, and inputting a batch of normalized values of HD electrode channels to the network at each time, the SNN model can differentiate between different hand gestures with maximum accuracy of around in a number of subjects. In this way, the network can work well on a quite limited amount of data with no need for data augmentation, preprocessing and spike coding.

1.1.2 Gesture Recognition based on Microscopic Neural Drive Information

The focus, in this part, is on HD-sEMG decomposition to extract microscopic neural drive information. HD-sEMG signals have encouraged emergence of sEMG decomposition algorithms in the last decade [29] as they provide a significantly high-resolution 2D image of Motor Unit (MU) activities in each time point. sEMG decomposition refers to a set of Blind Source Separation (BSS) [30] methods that extract discharge timings of motor neuron action potentials from raw HD-sEMG data. Single motor neuron action potentials are summed to form Motor Unit Action Potentials (MUAPs) that convert neural drive information to hand movements [31]. Motor unit discharge timings, also known as Motor Unit Spike Trains (MUSTs), represent sparse estimations of the MU activation times with the same sampling frequency and time interval as the raw HD-sEMG signals [32]. Extracted MUSTs are used in several domains such as identification of motor neuron diseases [33], analysis of neuromuscular conditions [34], and myoelectric pattern recognition [35]. HD-sEMG signals can be modelled as a spatio-temporal convolution of MUSTs, which provide an exact physiological description of how each hand movement is encoded at neurospinal level [36]. Thus, MUSTs are of trustworthy and discernible information on the generation details of different hand gestures.

Throughout this approach, we design a hybrid ViT-based architecture that classifies hand gestures using a combination of macroscopic and microscopic neural drive information. We, first, extract MUSTs of HD-sEMG signals via a BSS technique comprising of the two commonly used approaches in the literature, i.e. gradient Convolution Kernel Compensation (gCKC) and fast Independent Component Analysis (fastICA). Then, MUAPs are derived from spike trains using the Spike-Triggered Averaging method and their peak-to-peak values are computed. Two independent ViT-based architectures similar to those utilized in the previous section are fed with HD-sEMG signals and peak-to-peak MUAPs. The two models combine useful information from both of the ViTs and classify hand gestures based on the learned information. In this way, the whole architectures surpasses standalone ViT-based models that work solely with either macroscopic or microscopic neural drive information. This implies that both HD-sEMG signals that are collected from skin's surface and MUAPs that well represent physiological characteristics of the system of neurons and synapses carry valuable information about intended hand gestures that can be integrated to achieve

higher classification accuracy.

1.2 Contributions

The primary objective of this thesis is to design automated DL-based frameworks for hand gesture recognition via HD-sEMG signals. More specifically, this thesis proposes automated frameworks to classify a large number of hand gestures comprising one, two or multiple DoFs and evaluates the performance of these frameworks based on the classification accuracy, train/test time, memory usage and complexity. Besides model development, this thesis compares the performance of the proposed frameworks with that of two conventional ML models and a 3D CNN model, explaining in what aspects the proposed architecture functions more efficiently. In another part of this thesis, a more complex, hybrid architecture is designed integrating two different types of information obtained from HD-sEMG signals to achieve higher accuracy. This and the above-mentioned models are compared both numerically and statistically to have a clearer view of each model's function compared to the other models. The main contributions of this thesis research work are briefly outlined below:

(1) **The ViT-HGR Framework [1]:** Here, we investigate and design a Transformer-based architecture [37] to perform hand gesture recognition from HD-sEMG signals. Intuitively speaking, we capitalize on the recent breakthrough role of the Vision Transformer architecture together with its great potential for employing more input parallelization with attention mechanism. In this approach, we resort only to the transformer encoder and add a Fully Connected (FC) layer to its end to convert latent features to our labels. As direct application of ViT to HD-sEMG is not possible and straightforward, a particular signal processing step is developed to convert the HD-sEMG signals to a specific format that is compatible with ViTs. In other words, the proposed ViT-based ViT-HGR framework can learn from HD-sEMG signals rather than images. The signal processing approach, here, includes 3 consecutive steps, namely windowing, low-pass filtering and normalization. Each one of these steps is required to convert raw HD-sEMG signals to a waveform that is shorter in the time domain and more understandable for the ViT-based network. We utilize a specific normalization function, which is called

μ-law normalization and is different from the commonly-used Min-Max normalization, to increase the discriminative power of the proposed framework. Furthermore, the proposed framework can accurately classify a large number of hand gestures from scratch without any need for data augmentation and/or transfer learning.

(2) **The CT-HGR Framework [38]:** We show that the proposed CT-HGR architecture which is a similar framework to ViT-HGRachieves near baseline accuracy using instantaneous HD-sEMG data samples that are single frames of HD-sEMG images in a single time point. This is considered as a significant milestone as it paves the way for real-time learning from HD-sEMG signals. The proposed CT-HGR model is also evaluated using different window sizes and number of electrode channels. The capacity of the network is also increased and its performance is compared to the simpler CT-HGR model's. Cosine similarities of the positional embedding vectors for each model and different window sizes are sketched and the model's performance in assigning correct positions to the patches of HD-sEMG data is assessed. In addition to the classification accuracy, other metrics like recall, precision and F1-score for a specific case of the CT-HGR network is reported to show a better estimation of the network's ability to make accurate positive predictions. We also introduce the idea of integrating macroscopic and microscopic neural drive information through a hybrid DNN framework. The proposed variant of the CT-HGR framework, is a hybrid model that simultaneously extracts a set of temporal and spatial features through its two independent ViT-based parallel architectures (the so called Macro and Micro paths). The Macro Path is the baseline CT-HGR model, while the Micro path is fed with the peak-to-peak values of the extracted MUAPs of each source. The two independent Macro and Micro models are connected with two FC layers that perform final classification via the features derived from both models. We demonstrate that the proposed hybrid architecture outperforms standalone CT-HGR models in classifying a large number of hand gestures. Finally, a comprehensive comparison of our own method with other proposed models that utilized the same dataset is carried out in respect of the utilized window size, number of electrode channels, how the train/test splits are created and the classification accuracy.

(3) **The SNN-based Framework [39]:** In this part, We proposed an SNN-based model for hand gesture recognition from HD-sEMG signals by decoding neuromuscular information into spikes. We introduce a more straight-forward approach in comparison with using energy-density maps, spike coding and feature extraction as in [26, 28] to provide inputs for SNN architectures. We show that by considering each sample in the HD-sEMG dataset as a single time step, and inputting a batch of normalized values of HD electrode channels to the network at each time, the SNN model can well differentiate between different hand gestures in a number of subjects. This is considered as a well-suited and interpretable approach for sEMG signal classification considering the way they are generated through a convolutive mixture of a set of impulse trains (also known as spikes) with Motor Unit Action Potentials [17]. Moreover, our method is a compact SNN model that works efficiently for a small number of data samples with no need for huge pre-processing tasks, spike encoding and feature extraction. This can remarkably decrease the required time and memory for processing the data and training the SNN model.

It is worth mentioning that in all parts of the thesis, since the paper [2] on the HD-sEMG dataset did not refer to the train and test sets as a basis for comparison, we performed a 5-fold cross-validation as there are 5 sessions in the dataset. In this way, one (out of 5) repetition is considered as the test set and the remaining are assigned to the train set. Each time, the test set is changed until all the repetitions have been tested. Finally, the accuracy of each fold together with the average accuracy across all the folds are reported.

1.3 Thesis Organization

The rest of the thesis is organized as follows:

- Chapter 2 provides a literature review on Hand Gesture Recognition. In addition, this chapter provides the background material required to follow the developments presented in the remainder of the thesis. Furthermore, the detailed description of the dataset and the pre-processing procedures used in this thesis are presented in this chapter.

- Chapter 3 presents the proposed ViT-HGR framework developed for automatic gesture recognition from HD-sEMG signals.
- Chapter 4 presents the proposed CT-HGR architecture which is a similar model to the ViT-HGR and is evaluated using different settings of the HD-sEMG signal. In addition, a hybrid variant of the CT-HGR is introduced in this chapter which works based on a combination of macroscopic and microscopic neural drive information and outperforms a standalone CT-HGR structure.
- Chapter 5 provides a detailed description of the proposed SNN-based model that performs spatio-temporal gesture recognition in an event-based fashion. As opposed to the classical DNN architectures in previous chapters, SNNs are of the capacity to imitate human brain's cognitive function by using biologically inspired models of neurons and synapses which makes them more biologically explainable.
- Chapter 6 concludes the thesis and explains some directions for future research studies.

Chapter 2

Literature Review and Background

As stated previously, in the last few years, there has been a surge of significant interest on application of Deep Learning (DL) models to autonomously perform hand gesture recognition using surface Electromyogram (sEMG) signals. In this chapter, recent DL/ML-related research works proposed in the literature for HGR are presented. Background materials, which are widely used throughout this thesis and required to follow the subsequent chapters are also provided. Finally, an overview of the dataset used in this thesis is presented.

2.1 Hand Gesture Recognition

Hand gesture recognition using surface Electromyogram (sEMG) signals can be considered as one of the most important technologies in making efficient Human Machine Interface (HMI) systems. Hand gesture recognition-based HMI systems are applicable to a wide range of applications including prosthetics, neurorobotics, exoskeletons, and in Mixed (Augmented/Virtual) Reality settings, some of which targeting able-bodied individuals. In particular, sEMG-based hand gesture has been a topic of growing interest for development of assistive systems to help individuals with amputated limbs. Generally speaking, myoelectric prosthetic devices work by classifying existing patterns of the collected sEMG signals and synthesizing the intended gestures [40]. While conventional myoelectric control systems, e.g., on/off control or direct-proportional, have potential advantages, challenges such as limited Degree of Freedom (DoF) due to crosstalk have resulted in

the emergence of data-driven solutions. More specifically, to improve efficiency, intuitiveness, and the control performance of hand prosthetic systems, several Artificial Intelligence (AI) algorithms ranging from conventional Machine Learning (ML) models to highly complicated Deep Neural Network (DNN) architectures have been designed for sEMG-based hand gesture recognition in myoelectric prosthetic devices [41–44]. The ML-based models encompass traditional approaches such as Support Vector Machines (SVMs), Linear Discriminant Analysis (LDA), and k-Nearest Neighbors (kNNs) [9–12], and DNN-based models consist of frameworks such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformer-based architectures [14, 45–49].

sEMG signals represent the electrical activities of the muscles and are recorded by a set of non-invasive electrodes that are placed on the muscle tissue [7, 8]. Broadly speaking, there are two types of sEMG acquisition systems, called sparse and high-density [50, 51]. Both of these groups are obtained by placing electrodes on the surface of the muscle and recording the electrical activity of the muscle's Motor Unit Action Potentials (MUAPs) in response to the neural signals. Unlike sparse sEMG acquisition that involves a limited number of electrodes to record muscle activities, High-density sEMG (HD-sEMG) signals are obtained through a two-dimensional (2D) grid of electrodes, which cover an area of the muscle tissue and a large number of associated motor units [52, 53]. When comparing HD and sparse sEMG signals, it can be stated that more computational power is required for the signal processing and training stages when using HD-sEMG signals in contrast to the scenario where sparse sEMG signals are used. This point has also been observed in the prior works [40, 54], where it is stated that HD-sEMG-based interfaces result in more complex analog front-end and processing facilities leading to increase of the computation demand. It is, therefore, more difficult to design an ML/Deep Learning (DL)-based algorithm for hand gesture recognition from HD-sEMG signals. However, HD-sEMG signals are considered more potent than their sparse counterparts because of their ability to include both temporal and spatial information of muscle activities, which provides a high-resolution 3-dimensional (3D) signal (two dimensions in space and one in time) [55]. The HD-sEMG signal acquisition can evaluate functionality of the underlying neuromuscular system more precisely in terms of spatial resolution. Accordingly,

developing an efficient DNN-based framework that can effectively learn from a comprehensive HD-sEMG dataset is of great importance in neuro-rehabilitation research and clinical trials [56], which is the focus of this thesis.

Conventional ML models, such as SVMs and LDAs, utilized for sEMG-based hand gesture recognition, typically work well when dealing with small datasets. These methods, however, depend on manual extraction of handcrafted (engineered) features, which limits their generalizability as human knowledge is needed to find the best set of features [57]. Increasing the number of utilized electrodes and the number of gestures entails extracting more features, therefore, the feature extraction process becomes significantly complex and time-consuming. This is because more trials and efforts are required to boost the discriminative power of the model. Dependence on engineered features is partially/fully relaxed by utilization of DNN-based models. Among the most frequently used DNN architectures for the task of hand gesture recognition is the CNN-based frameworks. For example, Reference [14] converts sEMG signals to 3D images and uses transfer learning to feed them to a popular CNN trained on a database of natural images. CNNs, however, are designed to concentrate on learning spatial features of the input signals and fail to extract temporal features of the sEMG data. Accordingly, authors in [58] introduced an RNN-based network to catch the temporal features of HD-sEMG signals. This network contains dilated Long Short-Term Memories (LSTMs) to classify hand gestures from the transient phase of HD-sEMG signals. To overcome the issue of watching solely the spatial or temporal features of HD-sEMG data, researchers turned their attention to hybrid CNN-RNN frameworks that were designed to take both spatial and temporal information of the time-series sEMG datasets into account [18, 59]. For instance, Hu *et al.* [18] have applied attention mechanism on top of a hybrid CNN-LSTM (Long Short-Term Memory) model to perform hand gesture recognition based on sEMG signals with relatively large window sizes (i.e. 150 ms and 200 ms). They achieved classification accuracy of up to 87% using the largest window size. In [59], a dimensionality reduction method is proposed and assumed to enhance the classification accuracy when used with a hybrid CNN-LSTM architecture. In this framework [59], the classification accuracy is 88.9% on the same dataset as that of [18] for the 250 ms window size. Nonetheless, as well as not allowing entire input parallelization, hybrid CNN-RNN frameworks are usually computationally demanding and reveal important limitations with respect to the memory

usage and large training times. To alleviate the problem of lacking input parallelization in the aforementioned networks, References [16, 49] proposed transformer-based models for gesture recognition via sparse sEMG signals. For instance, in [16] a Vision Transformer (ViT) network is stacked to CNNs for gesture classification using the frequency domain information (Fourier Transform) of a set of sparse sEMG signals.

In this thesis, we develop and evaluate functionality of different DL methods for gesture recognition from HD-sEMG signals. On the one hand, DL models are more complicated than conventional ML solutions and the latter requires operator interventions for feature engineering, which is a burdensome procedure. On the other hand, gesture recognition based on sparse sEMG requires precisely locating the electrodes over the muscle to make sure that the same MUs are being recorded. Different from sparse sEMG, for HD-sEMG acquisition, a little change in the position of the electrode grid still records the MU activities with no significant change in the characteristics of the signal. This is why we aimed to focus on a series of DNN architectures explained in Chapters 3, 4, 5 that work based on HD-sEMG signals. In Chapters 3, 4 by eliminating the complexity of simultaneously exploiting CNNs/RNNs or merging them with transformers, we aim to construct a compact and stand-alone framework with reduced computational overhead. When it comes to real-time HMI devices, we hypothesized that by introducing a compact DL-based model developed based on HD-sEMG signals that has the capacity to classify a large number of hand gestures with a small amount of memory and training time, we can put a step forward towards development of more dextrous control interfaces. In Chapter 4 we also introduce the idea of integrating the macroscopic and microscopic neural drive information obtained from HD-sEMG data into a hybrid transformer-based framework for gesture recognition. Differently, in Chapter 5, we aim to develop an alternative and novel hand gesture recognition model based on the less-explored topic of Spiking Neural Networks (SNN), which performs spatio-temporal gesture recognition in an event-based fashion [20, 21]. An event-based processing approach refers to a type of data processing in which the system is susceptible to the occurrence of events rather than the static input [22]. It is worth noting that as opposed to the classical DNN architectures, SNNs are of the capacity to imitate human brain's cognitive function by using biologically inspired models of neurons and synapses [21]. Section 2.2 explains the structure and mathematical representation of our proposed Vision Transformer (ViT) framework.

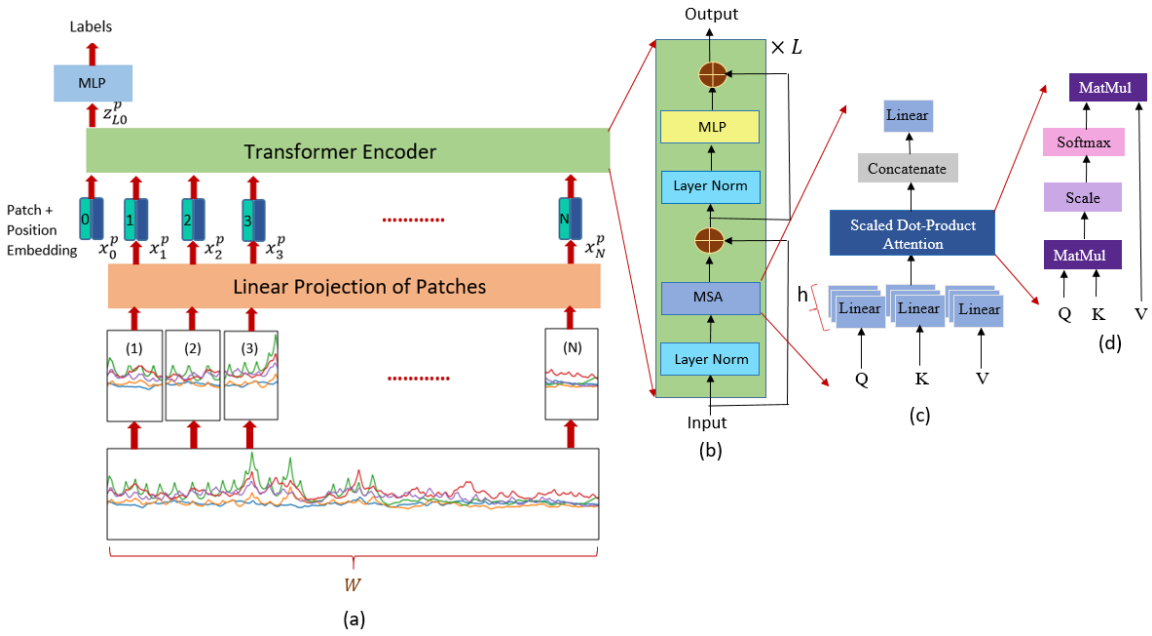

Figure 2.1: Overview of the ViT-HGR network. (a) The windowed HD-sEMG signal is fed to the ViT-HGR and split into smaller patches. The patches go through a linear projection layer which converts them from 3D to 2D data samples. A class token is added to the patches and the $N + 1$ patches are input to a transformer encoder. Ultimately, the first output of the transformer corresponding to the class token is chosen for the multi-class classification part. (b) The transformer encoder which is the fundamental part of the ViT, responsible for processing the input patches with its main part called Multi-head Self Attention (MSA). (c) The Multi-head Self Attention (MSA) Structure. (d) The Scaled Dot-Product module in the MSA block.

Detailed structure and functionality of SNNs is clarified in Chapter 5.

2.2 The Proposed Vision Transformer Framework

In this section, description of our proposed ViT framework(referred to as ViT-HGR), its main building blocks, and its adoption for the task of hand gesture recognition are presented. The proposed ViT-HGR framework is developed based on the ViT network in which the attention mechanism is utilized to understand the temporal and spatial connections among multiple data segments of the input. In this thesis, we demonstrate that attention mechanism can work independently of any other network and achieve high accuracy when trained from scratch with no data augmentation. We also show that the proposed framework can be trained even on small window sizes and more importantly on instantaneous data samples.

An overall illustration of the ViT-HGR is indicated in Fig. 2.1. After completion of the pre-processing steps discussed in the previous section, we have 3D signals of shape $W \times N_{ch} \times N_{cv}$, where W is the window size and N_{ch} and N_{cv} are the number of horizontal and vertical channels respectively. As an intuitive approach for patching the input data with 32, 64 or 128 electrode channels, we considered window sizes that are powers of two (in samples), which allows to smoothly divide input into smaller patches [60]. Therefore, the utilized window sizes in our experiments are of 64, 128, 256, and 512 data points (31.25, 62.5, 125, and 250 ms respectively considering $2,048$ Hz sampling frequency of the dataset). Furthermore, we have assessed the effect of changing the number of electrode channels by using 32, 64 and 128 out of the whole 128 channels. Therefore, we set N_{ch} to 4, 8, and 16 each time while N_{cv} remains constant at 8. In what follows, the major blocks of the proposed ViT-HGR network, namely "Patch Embedding", "Position Embedding", "Transformer Encoder", and the "Multilayer Perceptron (MLP)" blocks.

Patch Embedding

In this block, the 3D signals are divided into N small patches either horizontally, vertically or both. Therefore, we have N patches of size $H \times V \times N_{cv}$ that are then linearly flattened to 2D signals of size $N \times HVN_{cv}$ where, N is equal to WN_{ch}/HV and is the effective sequence length of the transformer's input and terms H and V represent the horizontal and vertical patch sizes, respectively. Consequently, there are N patch vectors $\boldsymbol{x}_i^p$, for $(1 \leq i \leq N)$. Using a trainable linear projection layer, the $\boldsymbol{x}_i^p$ vectors are embedded with the model's dimension d. The linear projection is shown with matrix $\boldsymbol{E}$, which is multiplied to each of the $\boldsymbol{x}_i^p$ and yields N vectors of dimension d. Moreover, a class token named $\boldsymbol{x}_0^p$ similar to what was previously used in the Bert framework [61] is prepended to the aforementioned vectors to gather all the useful information learned during the training stage and is used in the final step when different hand gestures are classified. The final sequence length of the transformer after adding the class token is $N + 1$.

Position Embedding

Unlike RNNs that process their inputs sequentially, transformers apply the attention mechanism to all of the data segments in parallel, which deprives them of the capacity to intrinsically learn about

the relative position of each patch of a single input. Because sEMG signals are time-series sequences of data points in which the location of each point matters for hand gesture classification tasks, we need to train the network to assign a specific position to each sample. Generally speaking, positional embedding is an additional piece of information that is injected into the network, helping it to identify how data points are ordered. There are different types of positional embeddings offered such as relative, 1D, 2D, and sinusoidal positional embeddings that may be learnable or non-learnable. In this context, we use a learnable 1D positional embedding vector that is added to each of the embedded $\boldsymbol{x}_i^p$ vectors to maintain and learn the position of each patch during the training phase. The final output $\boldsymbol{z}_0$ of the "Patch + Position Embedding" blocks is given by

$$\boldsymbol{z}_0 = [\boldsymbol{x}_0^p; \boldsymbol{x}_1^p \boldsymbol{E}; \boldsymbol{x}_2^p \boldsymbol{E}; \ldots; \boldsymbol{x}_N^p \boldsymbol{E}] + \boldsymbol{E}^{pos}, \tag{1}$$

where $\boldsymbol{E}^{pos}$ is an $(N+1) \times d$ matrix, holding the relative position of each patch in a d-dimensional vector.

Transformer Encoder

A typical transformer model consists of two major parts called encoder and decoder. In this paper, we aim to utilize only the former part. The transformer encoder is where the attention mechanism tries to find the similarities among the $N+1$ patches that arrive at its input. As can be seen in Fig. 2.1(b), there are L identical layers of transformer encoder in the ViT-HGR network and each has three separate blocks, named as "Layer Norm", "Multi-head Self Attention (MSA)" and "MLP". The $\boldsymbol{z}_0$ sequence of patches that is explained above is first fed to a normalization layer to improve the generalization performance of the model and accelerate the training process [62]. The "Layer Norm block" is then followed by the MSA module, which incorporates h parallel blocks (heads) of the scaled dot-product attention (also known as self attention). In the context of self attention, three different vectors $Keys(K)$, $Queries(Q)$ and $Values(V)$ of dimension d are employed for each input patch. For computing the self attention metric, the dot product of $Queries$ and all the $Keys$ are calculated and scaled by $1/\sqrt{d}$ in order to prevent the dot products from generating very large numbers. This matrix is then, converted into a probability matrix through a $softmax$ function and

is multiplied to the $Values$ to produce the attention metric as follows

$$Attention = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}})V. \tag{2}$$

In the MSA block (Fig. 2.1(c)), instead of dealing with d-dimensional $Queries$, $Keys$ and $Values$, we split them into h parallel heads and measure the self attention (Fig. 2.1(d)) metric on these heads independently. Finally, after finding the corresponding results for each head, we concatenate them to obtain the d-dimensional vectors of patches. As indicated in Fig. 2.1(b), residual paths from the encoder's input to the output of the MSA block are employed to avoid the gradient vanishing problem. The formulations for the above explanations are as follows

$$\boldsymbol{z}_l^{'} = MSA(LayerNorm(\boldsymbol{z}_{l-1})) + \boldsymbol{z}_{l-1}, \tag{3}$$

$$\boldsymbol{z}_l = MLP(LayerNorm(\boldsymbol{z}_l^{'})) + \boldsymbol{z}_l^{'}, \tag{4}$$

where $\boldsymbol{z}_l$ is the l^{th} transformer layer's output and $l = 1, \ldots, L$. The final output of the transformer encoder is given by

$$\boldsymbol{z}_L = [\boldsymbol{z}_{L0}^p; \boldsymbol{z}_{L1}^p; \ldots; \boldsymbol{z}_{LN}^p], \tag{5}$$

where $\boldsymbol{z}_{Li}^p$ is the final layer's output corresponding to the i^{th} patch and $i = 1, \ldots, N$. As mentioned before, among all the above vector of patches, the $\boldsymbol{z}_{L0}^p$ vector matching the class token is chosen for gesture classification. Authors in [19] claim that the learned features in the sequence of patches will eventually be included in the class token, which has a decisive role in predicting the model's output. Therefore, $\boldsymbol{z}_{L0}^p$ is passed to a linear layer which outputs the predicted gesture's label as

$$y_{\text{predicted}} = Linear(\boldsymbol{z}_{L0}^p). \tag{6}$$

2.3 Dataset

In this section, the HD-sEMG dataset used for model development and evaluation in this thesis is described in detail. Furthermore, additional information on the pre-processing operations performed on raw HD-sEMG data is explained in detail.

The dataset [2] used in this study is a recently released HD-sEMG dataset that contains two 64-electrode square grids (8×8) with an inter-electrode distance of 10 mm, which were placed on extensor and flexor muscles. The HD-sEMG acquisition setup is shown in Fig. 2.2. According to [2], the two HD-sEMG electrode grids covered the dorsal and the volar muscles of the forearm, specifically full or partial parts of flexor digitorum profundus and flexor digitorum superficialis, which is for flexion of fingers D2-D5, extensor digitorum communis for extension of fingers D2-D5, flexor carpi radialis and flexor carpi ulnaris for wrist flexion, extensor carpi radialis longus and extensor carpi ulnaris for wrist extension, pronator teres, supinator, and flexor pollicis longus for thumb flexion, extensor pollicis longus for thumb extension and abductor pollicis longus. Data from 20 participants is provided through the dataset. One of the subjects is not included in the study from the beginning due to its incomplete information. The participants performed 65 hand gestures that are combinations of 16 basic single degree of freedom movements. One of the gestures is carried out twice, therefore, there are 66 movements in total. The subjects performed each gesture 5 times with 5 seconds rest in between. Fig. 2.3 illustrates how the raw dataset is organized. The red plot shows the acquired HD-sEMG signal for one single channel of one specific hand movement. The blue line shows the repetition number of that gesture and the rest intervals. The signals were recorded through a Quattrocento (OT Bioelettronica, Torino, Italy) bioelectrical amplifier system with $2,048$ Hz sampling frequency. Signals of the successive channels were subtracted from each other (i.e., the sEMG data is acquired in a bipolar fashion) to lower the amount of common-mode noise. The rational behind selection of this publicly available dataset is that it comprises of a large number of gestures and electrodes, which allows development of a generalizable framework by investigating different settings of the input data. Additionally, this dataset provides straightforward instructions on how to deploy the dataset for different evaluation purposes. However, since the paper [2] on this

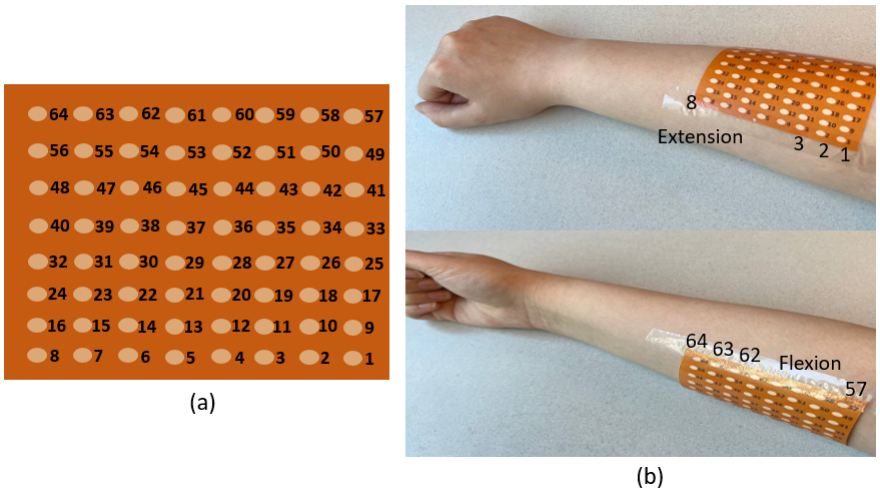

Figure 2.2: Representation of the HD-sEMG acquisition setup [2]: (a) The (8×8) HD-sEMG grid of electrodes. (b) The flexion and extension electrodes positioned on supinated and fully pronated forearm muscles.

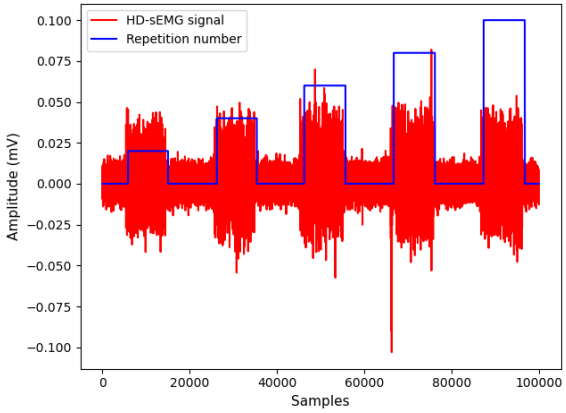

Figure 2.3: Illustrative example of the raw HD-sEMG dataset. The red plot is the sEMG signal for one single channel and one single movement and the blue plot shows the repetition number and the rest intervals for that movement.

dataset did not refer to the train and test sets as a basis for comparison, we performed a 5-fold cross-validation as there are 5 sessions in the dataset. In this way, one (out of 5) repetition is considered as the test set and the remaining are assigned to the train set. Each time, the test set is changed until all the repetitions have been tested. Finally, the accuracy of each fold together with the average accuracy across all the folds are reported.

Data Pre-processing

The raw HD-sEMG dataset is pre-processed following the common practice before being fed to the proposed ViT-HGR framework. More specifically, there is a consensus in the literature that pre-processing of sEMG signals should involve the following steps: (i) Band pass filtering; (ii) Rectification; (iii) Linear envelope computation, and (iv) Normalization. The utilized dataset is band-pass filtered with a hardware high-pass filter at 10 Hz and a low-pass filter at 900 Hz during recordings. All filter types are second order butterworth filters. Prior to the filtering step, full wave rectification is performed, i.e., absolute value of the signal is computed. The rectification step coupled with the low-pass filtering results in getting the shape or "envelope" of the sEMG signal. The envelope obtained by low-pass filtering is used to acquire active segment data [63, 64]. The purpose of the low pass filtering is to attenuate higher frequencies present in the signal while keeping the DC and low frequency values. In this regard, a low-pass first-order butterworth filter at 1 Hz is applied separately to each of the 128 channels of the data. We would like to mention that the utilized low-pass filtering approach is common in the literature, e.g., References [65–68] also applied a low-pass filter with cutoff frequency of 1 or 2 Hz and then windowed the signal. Shallower filters are widely recommended as they produce less signal distortions and spread them less in the time domain due to a shorter impulse response. Using the Fourier transform of the HD-sEMG signals [63,69], we observed that the cut-off frequencies up to 10 Hz are reasonable, as such we have also tested the ViT-HGR model's performance for 5 and 10 Hz low-pass filters in Chapter 4.1. It is worth nothing that low-pass filtering can be seen, more or less, to smoothing the data with a sliding averaging window. In this regard, theory predicts that a moving average filter will have a cutoff frequency equal to $f = \frac{0.443}{T_w}$ (e.g., a moving average filter with 1 Hz cutoff frequency corresponds to a 443 ms window size). Having said that, Butterworth filter in the time domain has an infinite impulse response with positive and negative lobes in contrast to the moving average filter, which is a finite positive window with constant values in time. Intuitively speaking, the positive and negative lobes of the butterworth filter neutralize the effect of averaging over time instants. In final pre-processing phase, the filtered signals are normalized by the μ-law normalization algorithm, which reduces significant changes in the dynamic range of the signals acquired from different electrodes.

The μ-law normalization is performed based on the following formulation

$$F(x_t) = \text{sign}(x_t)\frac{\ln\left(1+\mu|x_t|\right)}{\ln\left(1+\mu\right)}, \tag{7}$$

where x_t is the time-series sEMG signal for each electrode channel, and μ is the extent to which the signals are scaled down and is determined empirically. According to [41, 70], μ-law normalization helps the network to learn gestures more effectively. Fig. 2.4 shows the effects of the μ-law normalization. As can be seen from Fig. 2.4, original signals are closely spaced and their amplitudes change in a very small range (i.e. $\approx$ 0-0.02 V). They are, however, apparently separated after applying the μ-law normalization, which results in the sEMG signals ranging from $\approx$ 15-50 V. Having separated values provide the network with better learning capabilities to discriminate between different gestures. Finally, the sEMG signals are segmented following the common approach in the literature [71–74]. More specifically, after removing the rest intervals from the dataset, the signals are segmented with a specific window size creating the main 3D input of the ViT-HGR with shape $W \times N_{ch} \times N_{cv}$, where W is the window size and N_{ch} and N_{cv} are the number of horizontal and vertical channels respectively. It is worth mentioning that the window length should be less than 300 ms to satisfy the acceptable delay time [75], which is the real-time response required for practical myoelectric prosthetic control. Therefore, the window length for the classification purpose cannot surpass 300 ms [75]. This completes our discussion on the pre-processing stage.

2.4 Summary

In this chapter, an overview of hand gesture recognition, sEMG signals and the existing gesture recognition networks proposed in the literature is provided together with a detailed description of the background materials, the HD-sEMG dataset and the data pre-processing stages used in this thesis. The limitations and advantages of the existing DL/ML-based solutions for the gesture recognition task are also discussed in this chapter. In brief, the existing solutions require time-consuming manual extraction of handcrafted features, usually attend to either temporal or spatial information of the HD-sEMG signals and are computationally demanding in respect of training time and memory usage. Furthermore, this chapter features recent surges of interest in the context of hand gesture

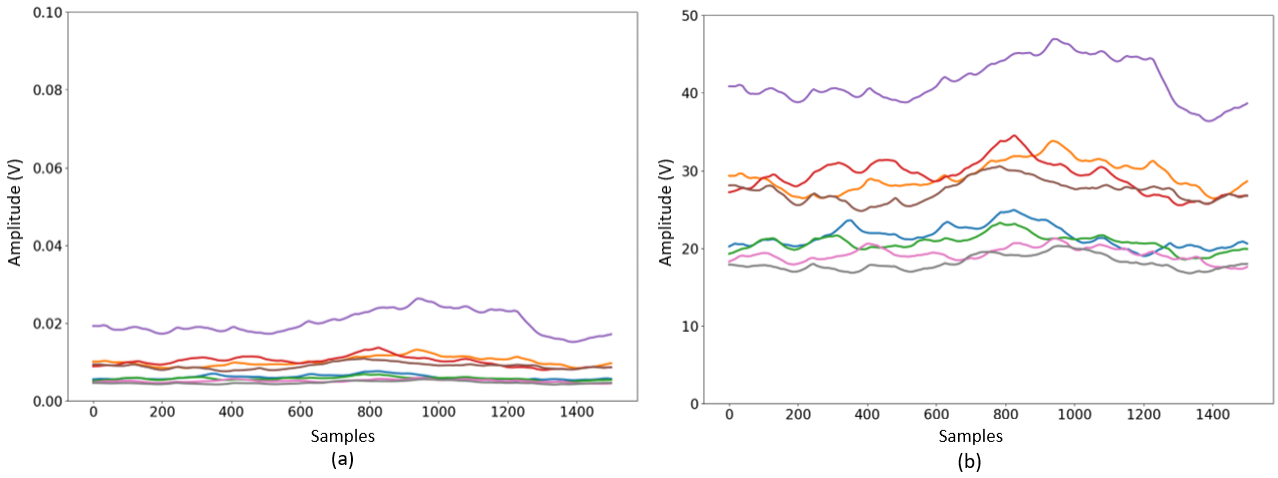

Figure 2.4: The impact of the μ-law normalization on the sEMG signals: (a) Low-pass filtered sEMG signals of 8 different electrode channels of the extensor grid before normalization. (b) Low-pass filtered sEMG signals of 8 different electrode channels of the extensor grid after normalization.

recognition as well as the variety of frameworks that have been suggested in the literature for this specific task. Finally, details and mathematics of the proposed ViT framework (ViT-HGR) are provided and the main building blocks of a ViT called the Multi-head Self Attention (MSA) and the Scaled Dot-Product are illustrated.

Chapter 3

Vision Transformer-based Hand Gesture Recognition from High Density Surface EMG Signals

Thanks to the recent evolution in the field of Artificial Intelligence (AI), specifically Deep Neural Networks (DNNs), significant advancements are expected on development of highly functional hand prostheses for upper limb amputees. Generally speaking, such advanced prosthesis systems are, typically, designed using surface Electromyogram (sEMG) signals [41, 44, 76, 77], representing action potentials of the muscle fibers [78]. The sEMG signals, after passing through a preprocessing stage, could be a valuable input for DNN architectures to perform different tasks including but not limited to motor control, prosthetic device control, and/or hand motion classification. Researchers are, therefore, turning their attention to development of DNN-based Human Machine Interface (HMI) algorithms using sEMG signals to design more accurate and more efficient myoelectric prosthesis control systems.

As mentioned before, sEMG signals are generally classified into two main categories, i.e., sparse and high-density [79–81]. Although using a large number of electrodes makes the computational process challenging, there has been a surge of recent interest in the use of High-Density sEMG

(HD-sEMG) signals which is the focus of this thesis. Recently, there has been a surge of significant interest on application of Deep Learning (DL) models to autonomously perform hand gesture recognition using surface Electromyogram (sEMG) signals. Many of the existing DL models are, however, designed to be applied on sparse sEMG signals. Furthermore, due to the complex structure of these models, typically, we are faced with memory constraint issues, require large training times and a large number of training samples, and; there is the need to resort to data augmentation and/or transfer learning. In this chapter, we investigate and design a Vision Transformer (ViT)-based architecture to perform hand gesture recognition from High Density (HD-sEMG) signals. Intuitively speaking, we capitalize on the recent breakthrough role of the transformer architecture in tackling different complex problems together with its potential for employing more input parallelization via its attention mechanism. The proposed Vision Transformer-based Hand Gesture Recognition (ViT-HGR) framework can overcome the aforementioned training time problems and can accurately classify a large number of hand gestures from scratch without any need for data augmentation and/or transfer learning. Our experiments with 64-sample ($31.25\ ms$) window size yield average test accuracy of $84.62 \pm 3.07\%$, where only $78,210$ learnable parameters are utilized in the model. The compact structure of the proposed ViT-based ViT-HGR framework (i.e., having significantly reduced number of trainable parameters) shows great potentials for its practical application for prosthetic control. Since HD-sEMG data sets have a 3-Dimensional (3D) structure, Vision Transformers (ViT) [19] can be considered as an appropriate architecture to address the challenges identified above. The proposed ViT-HGR framework can overcome training time problems and evaluation accuracy that we mostly face while working with other similar networks such as conventional ML algorithms or more advanced DNNs such as Long Short-Term Memories (LSTMs). However, we cannot directly provide HD-sEMG signals as input to the ViTs, and particular signal processing steps are required to modify the signal into a format that is compatible with the ViT's input. Therefore, the proposed ViT-HGR architecture converts the main signal into smaller portions using a specific window size and then feeds each of these portions to the ViT for further analysis. To develop and evaluate the proposed ViT-HGR framework, we used the HD-sEMG dataset explained in 2.3 consisting of 65 isometric hand gestures and 128 distinct channels for recording the signals [2]. Our results show superior performance of the ViT-HGR framework compared to its

counterparts illustrating reduced training time and increased testing accuracy.

The remainder of the chapter is organized as follows: The proposed ViT-HGR framework together with steps used to prepare HD-sEMG signals to be fed to the ViT architecture are concisely discussed in Section 3.1. Experimental results are presented in Section 3.2. Finally, Section 3.3 concludes the chapter.

3.1 The proposed ViT-HGR Framework

The proposed ViT-HGR framework in this chapter is implemented based on the transformers and attention mechanism [19] and explained in detail in Chapter 2.2. The attention mechanism incorporated with CNNs and LSTMs were formerly used for the hand movement classification tasks because of their proven ability to leverage temporal information of the sEMG signals [18]. However, in this work, we indicate that the attention framework itself is sufficient to surpass the other networks and because of our data preparation approach, there is no need for data augmentation.

The overall structure of the proposed ViT-HGR architecture is shown in Fig. 3.1. In this chapter, we focused on a 64-sample (31.25 ms considering the $2,048$ Hz sampling frequency) sliding window size with a skip_step equal to 32 samples and utilized 64 electrode channels to convert the HD-sEMG signals to the acceptable input for the ViT. Although using a larger window size potentially leads to better performance, use of shorter windows (e.g., 31.25 ms) is preferred allowing extra necessary time for practical implementation in real scenarios.

After the windowing step, each of the (window_size, N_{ch}, N_{cv}) sequences, where N_{ch} is the number of horizontal channels and N_{cv} is the number of vertical channels, are considered as the 3D input of the ViT and is divided into N small square patches in the Patch Embedding block. N_{cv} and N_{ch} parameters are both set to 8 in this work. The sequence of patches is then fed to the transformer encoder clarified in 2.2 for gesture classification.

3.2 Experiments and Results

The raw dataset consists of plenty of sharp fluctuations that commonly occur in the EMG signals. Not only are these fluctuations required for an accurate gesture recognition task but they also

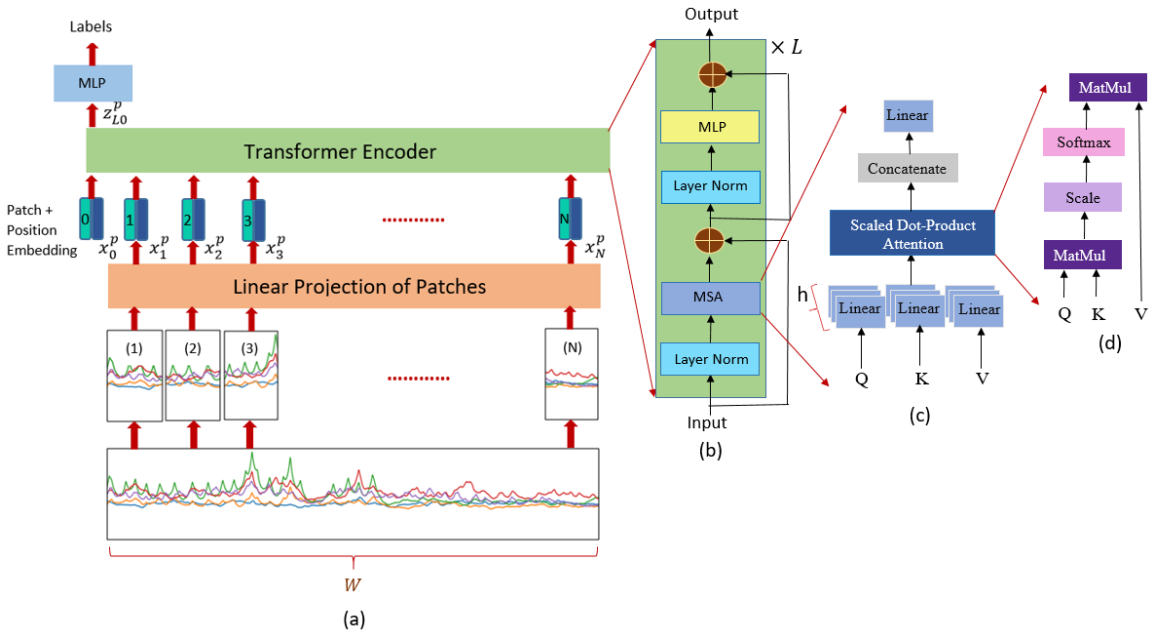

Figure 3.1: Overview of the ViT-HGR network. (a) The windowed HD-sEMG signal is fed to the ViT-HGR and split into smaller patches. The patches go through a linear projection layer which converts them from 3D to 2D data samples. A class token is added to the patches and the $N + 1$ patches are input to a transformer encoder. Ultimately, the first output of the transformer corresponding to the class token is chosen for the multi-class classification part. (b) The transformer encoder which is the fundamental part of the ViT, responsible for processing the input patches with its main part called Multi-head Self Attention (MSA). (c) The Multi-head Self Attention (MSA) Structure. (d) The Scaled Dot-Product module in the MSA block.

prevent the network from training the useful information and increasing its accuracy. As a result, a set of pre-processing tasks explained in Chapter 2.3 are performed on the HD-sEMG data.

We evaluate our proposed framework on all the 65 various hand gestures of the dataset and considered 3 distinct models, in which MLP size and the Embedding dimension are different. For each model a window size of 64 samples (31.25 ms) is tested to assess the impact of increasing the window size on performance of ViTs. The patch size, models' depth and the number of heads in all of the models are set to $(4, 4)$, 1 , 12 respectively. Adam optimization method is deployed with (β_1 , β_2) equal to $(0.9, 0.999)$, with the learning rate of 0.0001 and the weight decay of 0.001. We fix the batch size and the number of epochs to 128 and 30 respectively and use the Cross-entropy loss function for calculating the models' performance. The Model IDs and their corresponding parameters are presented in Table 3.1.

Table 3.2 demonstrates the average accuracy over 19 subjects for each repetition, the average accuracy after performing 5-fold cross validation (i.e., Acc. F1 to Acc. F5) and the corresponding

Table 3.1: Model IDs and their parameters

Model ID	MLP Size	Embed Dimension
I	384	192
II	96	96
III	48	48

Standard Deviation (STD) for each model. It also shows how many parameters are trained for each specific model. As can be seen, the highest accuracy, in general, pertains to the 3^{rd} repetition and the lowest to the 1^{st} one. The average accuracy rises by 0.38% from model I to model II although the number of parameters in the former is roughly 4 times as large as that in the latter. This indicates that to obtain decent accuracy in ViTs, there is no need to increase the number of parameters and hence the complexity and the training time when this accuracy is achieved with almost $78,000$ parameters. The STD also decreases by a minimal amount when increasing the number of parameters, leading to a reasonable trade-off between the number of parameters on the one hand and the acquired accuracy and STD on the other hand. Fig. 3.2 visualizes the results from Table 3.2. According to the Wilcoxon signed-rank test, the difference in accuracy between models I, III and models II , III is statistically significant. The ns, **, *** signs in Fig. 3.2 correspond to the following p-values:

- Not significant (ns): $5.00\text{e}-02 < \text{p} \leq 1.00\text{e}+00$
- ** : $1.00\text{e}-03 < \text{p} \leq 1.00\text{e}-02$
- *** : $1.00\text{e}-04 < \text{p} \leq 1.00\text{e}-03$

For comparison purposes, the proposed ViT-HGR framework is evaluated against the LDA approach, which is among the most popular conventional ML algorithms that has been widely used for sEMG gesture recognition. We should mention that at the time of preparing the [1] paper, as the utilized dataset had been released very recently, there were only couple of other works [71, 82] developed based on this dataset. Reference [71] focused on the same task as our work but used traditional ML methods such as LDA. The test-train split, however, was not mentioned in Reference [71]

Table 3.2: Comparison of the average/overall accuracy for each fold over 19 participants for each ViT model

Model ID	Acc. F1 (%)	Acc. F2	Acc. F3	Acc. F4	Acc. F5	Avg. Acc.	STD (%)	# Parameters
I	75.92	87.79	88.47	87.71	83.22	**84.62**	3.07	340,866
II	75.21	87.34	88	87.88	82.78	**84.24**	3.14	78,210
III	73.92	87.09	87.56	87.09	81.65	**83.46**	3.17	25,314

Table 3.3: Comparison of the average/overall accuracy for each repetition over 19 participants for the LDA model.

Acc. F1 (%)	Acc. F2	Acc. F3	Acc. F4	Acc. F5	Avg. Acc.	STD (%)
82.58	69.65	84.21	82.74	75.27	**78.89**	11.15

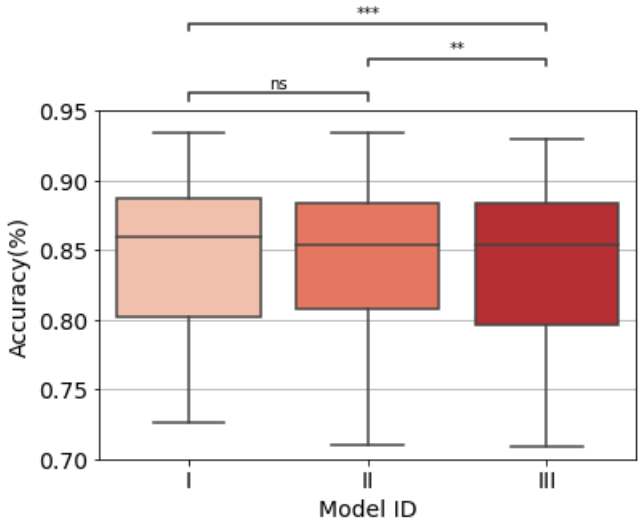

Figure 3.2: Accuracy boxplots and Wilxocon test's results of 3 different models of the ViT-HGR framework. Each boxplot represents the Interquartile Range for 19 subjects. The accuracy for each subject is the average accuracy after performing 5-fold cross validation.

rendering direct comparison inapplicable. Reference [82], on the other hand, only focused on the dynamic and transient phase of gesture movements when the signals are not stabilized or plateaued, which is a different task as this work. A thorough comparison of our proposed framework's accuracy with that of the the similar works on the same dataset is done in Chapter 4.3.6. Consequently, to have a fair comparison in Reference [1], we have implemented an LDA method similar to that

of [71] based on the same setting as our proposed ViT framework. Five key features for classification of sEMG signals including Mean Absolute Value (MAV), the number of Zero Crossings (ZC), Waveform Length (WL), Root Mean Square (RMS) and Slope Sign Change (SSC) along with four auto regressive coefficients of each cropped window are fed to the LDA algorithm [71, 83]. In Table 3.3, the above-mentioned results obtained from the LDA model are presented. Evidently, the average accuracy in the ViT is around 5% bigger and STD is around 8% smaller than that in the LDA, which highlights the great power of ViTs in solving HD-sEMG hand movement classification problems. Furthermore, the signal processing + training time for both ViT-HGR and LDA and for each repetition of one subject is measured. This time is 168 seconds for the ViT-HGR model I and 367 seconds for the LDA, which means for achieving the total average accuracy over all the 19 subjects, we require 4.4 hours while this will be 9.6 hours for the LDA.

3.3 Discussion and Summary

In this chapter, we introduced a Vision Transformer-based framework, referred to as the ViT-HGR, for application on HD-sEMG signals for the task of hand gesture classification. To implement the ViT-HGR framework, we capitalize on the recent breakthrough of Transformers in different ML domains and their potentials for employing more input parallelization, therefore, reducing complexity of the underlying model. As direct application of ViT to HD-sEMG is not straightforward, a particular signal processing step is developed to convert the HD-sEMG signals to a specific format that is compatible with ViTs. The proposed ViT-HGR framework can overcome the training time problems associated with recurrent networks and can accurately classify a large number of hand gestures from scratch without any need for data augmentation and/or transfer learning. By comparing the test accuracy associated with three unique variants of the ViT-HGR network, we showed that it could reach average accuracy of 84.24% (for 65 various hand gestures over 19 participants) with no more than $78,000$ parameters. Also, there is a significant discrepancy between the accuracy obtained for models I & III and models II & III, implying that the smallest ViT-HGR model produces statistically different results from the other two ones and that increasing the capacity of the ViT-HGR framework can have significant effect on the produced results. Moreover, the average

accuracy for the LDA model is 5% lower and its signal processing + training time is more than twice that of the ViT-HGR. This illustrates potentials of the proposed ViT-HGR framework to act as a feasible substitute for LDAs in hand gesture recognition tasks. This is because the ViTs are more straightforward to be implemented on HD-sEMG data sets with no need for any additional feature extraction calculations and it also takes far less training time, which is a significantly critical issue when working with large data sets. Our primary focus in this chapter was on 64-sample portions of the flexor signals because our purpose was to assess the proposed network's performance on small patterns of the HD-sEMG dataset. In the next chapter, we aim to extend the number of channels and the variety of the window size to evaluate their impact on the framework's efficacy in terms of the train time, test time, required memory, average accuracy, STD, and the number of learnable parameters.

Chapter 4

Transformer-based Hand Gesture Recognition from Instantaneous to Fused Neural Decomposition of High-Density sEMG Signals

Designing efficient and labor-saving prosthetic hands requires powerful hand gesture recognition algorithms that can achieve high accuracy with limited complexity and latency. In this context, this chapter proposes a Compact Transformer-based Hand Gesture Recognition framework referred to as CT-HGR, which employs a vision transformer network to conduct hand gesture recognition using HD-sEMG signals. This model is an extension of the ViT-HGR model described in the previous section. Attention mechanism in the proposed model identifies similarities among different data segments with a greater capacity for parallel computations and addresses the memory limitation problems while dealing with inputs of large sequence lengths. One of the differences between the ViT and a typical transformer is that the ViT is generally designed to be applied on 2D RGB images that have an additional dimension (the 3rd dimension) as the color channel rather than 2D time-series signals. Considering the fact that HD-sEMG signals comprise of two dimensions in space and one in time (3 dimensions in total), they can be an appropriate input to a ViT. The CT-HGR architecture

is very similar to the ViT-HGRframework expounded in Chapter 2.2. In this chapter, a comprehensive evaluation of the proposed ViT-based framework for hand gesture classification on HD-sEMG dataset is carried out. Here, we assess performance of our proposed ViT-based architecture in more detail and report accuracy results of utilizing different window sizes and electrode channels with two different versions of the CT-HGR framework. Additionally, as mentioned in [84], instantaneous training with HD-sEMG signals refers to training the network with a 2D image depicting MUAP activities under a grid of electrodes at a single time point. In this chapter, we also show that there are reproducible patterns among instantaneous samples of a specific gesture which could also be a physiological representation of muscle activities in each time point. We demonstrate that the proposed framework can perform instantaneous hand gesture classification using sEMG image spatially composed from HD-sEMG signals. In other words, it can achieve acceptable accuracy when receiving, as an input, a single frame of the HD-sEMG image. A variant of the CT-HGR is also designed to incorporate microscopic neural drive information in the form of Motor Unit Spike Trains (MUSTs) extracted from HD-sEMG signals using Blind Source Separation (BSS). This variant is combined with its baseline version via a hybrid architecture to evaluate potentials of fusing macroscopic and microscopic neural drive information. The utilized HD-sEMG dataset is the HD-sEMG dataset explained in Chapter 2.3. Briefly speaking, the proposed CT-HGR framework is applied to 31.25, 62.5, 125, 250 ms window sizes of the above-mentioned dataset utilizing 32, 64, 128 electrode channels. Our results are obtained via 5-fold cross-validation by first applying the proposed framework on the dataset of each subject separately and then, averaging the accuracies among all the subjects. The average accuracy over all the participants using 32 electrodes and a window size of 31.25 ms is 86.23%, which gradually increases till reaching 91.98% for 128 electrodes and a window size of 250 ms. The CT-HGR achieves accuracy of 89.13% for instantaneous recognition based on a single frame of HD-sEMG image. The proposed model is also statistically compared with a 3D Convolutional Neural Network (CNN) and two different variants of Support Vector Machine (SVM) and Linear Discriminant Analysis (LDA) models. The accuracy results for each of the above-mentioned models are paired with their precision, recall, F1 score, required memory, and train/test times. The results corroborate effectiveness of the proposed CT-HGR framework compared to its counterparts.

The rest of the chapter is structured as follows: Our proposed framework is presented in Section 4.1. Our experiments and evaluations of implementing the proposed framework are discussed in Section 4.3, a detailed discussion of the acquired results is generated in Section 4.4 and finally, Section 4.5 concludes the paper.

4.1 The proposed ViT-HGR Framework

The proposed CT-HGR framework in this chapter is implemented based on the transformers and attention mechanism [19] and explained in detail in Chapter 2.2. The CT-HGR is developed based on the ViT network in which the attention mechanism is utilized to understand the temporal and spatial connections among multiple data segments of the input. As stated previously, several studies have employed the attention mechanism together with hybrid CNN-RNN models to force the network to learn both spatial and temporal information of the signals [18, 42]. However, in this chapter, we demonstrate that attention mechanism can work independently of any other network and achieve high accuracy when trained from scratch with no data augmentation. We also show that the proposed framework can be trained even on small window sizes and more importantly on instantaneous data samples. this chapter's work has been published in [38]. It is worth noting that in the recent literature, there are some works [71, 84] that focused on small windows sizes achieving accuracies in the range of 89.3 - 91.81 %. We, in this chapter, try to compare our proposed CT-HGR model with other suggested models in terms of the classification accuracy in instantaneous gesture recognition. The overall structure of the proposed CT-HGR architecture is shown in Fig. 4.1.

The dataset [2] used in this study is the HD-sEMG dataset that is explained in Chapter 2.3. The raw HD-sEMG dataset is pre-processed following the common practice described in Chapter 2.3 before being fed to the proposed CT-HGR framework. Using the Fourier Transform of the HD-sEMG signals [63, 69], we observed that the cut-off frequencies up to 10 Hz are reasonable for low-pass filtering, as such we have also tested the model's performance for 5 and 10 Hz low-pass filters as shown in Table 4.1.

After completion of the pre-processing steps discussed in Chapter 2.3, we have 3D signals of shape $W \times N_{ch} \times N_{cv}$, where W is the window size and N_{ch} and N_{cv} are the number of horizontal

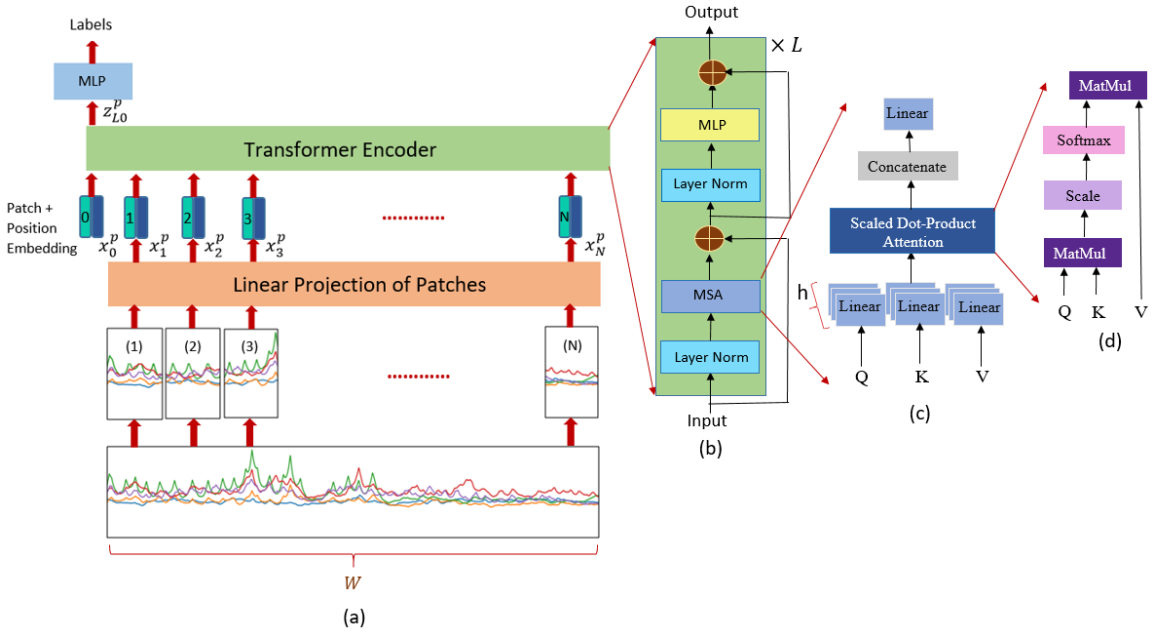

Figure 4.1: Overview of the CT-HGR network. (a) The windowed HD-sEMG signal is fed to the CT-HGR and split into smaller patches. The patches go through a linear projection layer which converts them from 3D to 2D data samples. A class token is added to the patches and the $N+1$ patches are input to a transformer encoder. Ultimately, the first output of the transformer corresponding to the class token is chosen for the multi-class classification part. (b) The transformer encoder which is the fundamental part of the ViT, responsible for processing the input patches with its main part called Multi-head Self Attention (MSA). (c) The Multi-head Self Attention (MSA) Structure. (d) The Scaled Dot-Product module in the MSA block.

Table 4.1: Comparison of classification accuracy and STD for each fold and their average for $W = 64, 128$ electrode channels (CT-HGR-V1), and different cutoff frequencies for the low-pass filter. The accuracy and STD for each fold is averaged over 19 subjects.

# Channels	Window size (samples)	Cutoff freq(Hz)	Fold1(%)	Fold2 (%)	Fold3 (%)	Fold4 (%)	Fold5 (%)	Average (%)
128	64	1	82.14 (±3.26)	93.30 (±2.14)	93.75 (±2.08)	93.39 (±2.11)	90.07 (±2.55)	90.53 (±2.43)
		5	81.94 (±3.74)	92.74 (±2.46)	93.48 (±2.12)	93.33 (±2.10)	89.64 (±2.95)	90.23 (±2.67)
		10	80.40(±3.44)	91.42 (±2.38)	92.27 (±2.28)	91.98 (±2.28)	88.30 (±2.80)	88.87 (±2.64)

and vertical channels respectively. As an intuitive approach for patching the input data with 32, 64 or 128 electrode channels, we considered window sizes that are powers of two (in samples), which allows to smoothly divide input into smaller patches [60]. Therefore, the utilized window sizes in our experiments are of 64, 128, 256, and 512 data points (31.25, 62.5, 125, and 250 ms respectively considering $2,048$ Hz sampling frequency of the dataset). Furthermore, we have assessed the effect of changing the number of electrode channels by using 32, 64 and 128 out of the whole 128 channels. Therefore, we set N_{ch} to 4, 8, and 16 each time while N_{cv} remains constant at 8.

4.2 Power Spectral Density (PSD) Analysis

One of the experiments we did in [38] and in this chapter is comparing performance of our proposed CT-HGR architecture with that of the conventional ML and a 3D CNN models. For the former, we design two sets of traditional ML algorithms based on SVMs and LDAs, which are commonly [10, 85–88] used for hand gesture recognition tasks. In the first experiment and following [10, 85, 86], we trained SVM and LDA models based on the following set of classical features: Root Mean Square (RMS), Zero Crossings (ZC), Slope Sign Change (SSC), and Wavelength (WL). To observe effects of recently proposed feature extraction methods, we did a second experiment based on features introduced in Reference [88]. These features are a rough estimate of the Power Spectral Density (PSD) of the signal by finding an approximate relation between the PSD in the frequency domain and the time-domain signal utilizing characteristics of the Fourier transform and the Parseval's theorem. According to Parseval's theorem, the sum of squares of a function is equal to the sum of squares of its Fourier transform, i.e.,

$$\sum_{j=0}^{N-1} |x\,[j]|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X\,[k]\,X*[k]| = \sum_{k=0}^{N-1} P\,[k] \tag{1}$$

where x is the original sEMG signal, X, its discrete Fourier transform, $X*$, the conjugate of X, P is the power spectrum, and terms j, k are the time and frequency indices, respectively. The utilized set of features are m_0, $m_1 - m_0$, m_2, $m_3 - m_2$, and $m_4 - m_3$, which are defined as follows

$$m_0 = \frac{A^\lambda}{\lambda}, \quad m_1 = \frac{B^\lambda}{\lambda}, \quad \text{and} \quad m_2 = \frac{C^\lambda}{\lambda} \quad m_3 = \frac{D^\lambda}{\lambda} \quad \text{and} \quad m_4 = \frac{E^\lambda}{\lambda},$$

where

$$A = \sqrt{\frac{\sum_{j=0}^{N-1} |x\,[j]^2|}{N}}, \quad B = \sqrt{\frac{\sum_{j=0}^{N-1} |\Delta \bullet x\,[j]|^2}{N}}, \quad D = \sqrt{\frac{\sum_{j=0}^{N-1} |\Delta^2 \bullet x\,[j]\,|^2}{N}},$$

$$C = \sqrt{\frac{\sum_{j=0}^{N-1} \Delta d_1^2}{N}}, \quad \text{and} \quad E = \sqrt{\frac{\sum_{j=0}^{N-1} \Delta d_2^2}{N}},$$

Table 4.2: Comparison of classification accuracy and STD for each fold and their average for different window sizes and number of channels (CT-HGR-V1). The accuracy and STD for each fold is averaged over 19 subjects.

# Channels	Window size (samples)	Fold1 (%)	Fold2 (%)	Fold3 (%)	Fold4 (%)	Fold5 (%)	Average (%)
32	64	76.85 (±3.83)	89.30 (±2.61)	89.91 (±2.54)	89.62 (±2.67)	85.49 (±3.07)	86.23 (±2.94)
	128	77.21 (±3.56)	89.48 (±2.60)	90.05 (±2.63)	90.00 (±2.61)	85.83 (±2.96)	86.51 (±2.87)
	256	77.63 (±3.50)	90.51 (±2.52)	90.79 (±2.45)	90.99 (±2.42)	86.66 (±2.97)	87.32 (±2.77)
64	64	79.64 (±3.38)	91.92 (±2.41)	92.55 (±2.18)	92.37 (±2.32)	88.16 (±2.77)	88.93 (±2.61)
	128	80.26 (±3.44)	92.32 (±2.27)	92.94 (±2.20)	92.48 (±2.22)	88.46 (±2.77)	89.29 (±2.58)
	256	81.43 (±3.31)	92.89 (±2.15)	93.42 (±2.13)	93.05 (±2.18)	89.29 (±2.69)	90.02 (±2.49)
128	64	82.14 (±3.26)	93.30 (±2.14)	93.75 (±2.08)	93.39 (±2.11)	90.07 (±2.55)	90.53 (±2.43)
	128	82.80 (±3.22)	93.47 (±2.13)	93.98 (±2.03)	93.82 (±2.10)	90.30 (±2.48)	90.87 (±2.39)
	256	83.20 (±3.21)	94.19 (±2.00)	94.25 (±1.97)	94.42 (±1.91)	90.70 (±2.46)	91.35 (±2.31)
	512	83.87 (±3.21)	94.62 (±1.88)	95.26 (±1.80)	94.89 (±1.85)	91.26 (±2.37)	91.98 (±2.22)

Table 4.3: Comparison of classification accuracy and STD for each fold and their average for different window sizes and 128 electrode channels (CT-HGR-V2). The accuracy and STD for each fold is averaged over 19 subjects.

# Channels	Window size (samples)	Fold1 (%)	Fold2 (%)	Fold3 (%)	Fold4 (%)	Fold5 (%)	Average (%)
128	64	83.82 (±3.22)	94.03 (±2.02)	94.58 (±1.9)	94.29 (±2.05)	90.84 (±2.58)	91.51 (±2.35)
	128	83.98 (±3.17)	94.09 (±2.00)	94.82 (±1.86)	94.65 (±1.94)	90.89 (±2.45)	91.69 (±2.28)
	256	84.74 (±3.13)	94.60 (±1.92)	95.19 (±1.80)	95.06 (±1.86)	91.59 (±2.44)	92.24 (±2.23)
	512	85.27 (±3.12)	95.55 (±1.70)	95.81 (±1.65)	95.60 (±1.73)	92.16 (±2.32)	92.88 (±2.10)

where $\Delta\bullet$, $\Delta^2\bullet$ are the signs for the first and second derivatives and d_1, d_2 are the first and second derivatives of the original sEMG signal.

In the next section, the results corresponding to running conventional ML models using the above-mentioned sets of features are shown. Moreover, we will describe all other various experiments performed in this study and present the obtained results and their explanations in detail.

4.3 Results

We perform several experiments to evaluate performance of the proposed framework under different configurations. In the following, each of the conducted experiments and their corresponding results are presented separately. The implemented models are evaluated on all the 66 gestures of the HD-sEMG dataset performed by 19 healthy subjects. The implementations were developed in the PyTorch framework and the models are trained using an NVIDIA GeForce GTX 1080 Ti GPU.

4.3.1 Overall Performance Evaluation under Different Configurations

In this experiment, we employ 4 different window sizes together with 3 different combination of electrodes of the HD-sEMG dataset and report the achieved accuracy for each of the 5 test folds and

Table 4.4: The number of learnable parameters for different number of electrodes and window sizes.

# Channels	Window size (samples)	# Parameters (CT-HGR-V1)	# Parameters (CT-HGR-V2)	# Parameters (3D CNN)
32	**64**	46,530	-	-
	128	47,042	-	-
	256	48,066	-	-
64	**64**	62,914	-	294,914
	128	63,426	-	311,298
	256	64,450	-	319,490
128	**64**	95,682	273,346	-
	128	96,194	274,370	-
	256	97,218	276,418	-
	512	99,266	280,514	-

the overall averaged accuracy. In the first model, referred to as the CT-HGR-V1, the simplest and smallest CT-HGR model that gives acceptable results is chosen. The length of windowed signals, in this model, is set to 64, 128, 256 and 512 (31.25, 62.5, 125, 250 ms respectively) with skip step of 32 except for the window size of 512 for which the skip step is set to 64. To measure effects of increasing the number of channels on the performance of the proposed architecture, we consider three different settings using all, half, and $1/4$ of the 128 electrodes. In the half mode, electrodes of multiple of 2 and in the $1/4$ mode, electrodes of multiple of 4 were chosen. In this regard, we chose one electrode out of four adjacent electrodes to make sure that the utilized electrodes still cover the whole recorded area and the only thing that changes is the distance among the chosen electrodes. In such a scenario (which intuitively speaking can be interpreted as an unbiased way of choosing the electrodes), we make sure that we do not miss much of the information that high density grids usually provide and the model do not lose its generalizability when being fed with the data from fewer number of electrode channels. As stated previously, the number of horizontal electrode channels in the CT-HGR's input is 4, 8, and 16 while the number of vertical channels is 8. Regarding the hyperparameters of the model, the model's (embedding) dimension is 64, and the patch size is set to $(8, 4)$, $(8, 8)$, and $(8, 16)$ for 32, 64, and 128 number of channels, respectively. The CT-HGR-V1 model contains only 1 transformer layer and 8 heads. The MLP block's hidden size is set to 64, the same as its input size. The CT-HGR-V1 model is trained with 20 epochs and batch size of 128 for each subject independently. The optimization method used is Adam with $\beta 1 = 0.9$ and $\beta 2 = 0.999$ parameters, learning rate of 0.0001 and weight decay of 0.001. Learning rate annealing is deployed after the first 10 epochs for faster convergence. The cross-entropy loss function is considered as the objective function. Table 4.2 represents the acquired accuracy and

standard deviation (STD) for each individual window size and number of channels. It is worth noting that the 512 window size is only tested with the whole electrode channels of the dataset to indicate the potential best performance of the network.

A second variant of the CT-HGR model, referred to as CT-HGR-V2, is also tested where the model's dimension and the number of hidden layers in the MLP layer are twice those of CT-HGR-V1. We apply the CT-HGR-V2 model on the data samples derived from the whole 128 electrodes to compare it with the last 4 rows of Table 4.2. The results are shown in Table 4.3. Table 4.4 illustrates the number of learnable parameters for each window size and number of channels in both models. Fig. 4.2 demonstrates the box plots for the accuracy of CT-HGR-V1 obtained for each individual fold and different window sizes from $W = 64$ to $W = 512$ (Fig. 4.2(a-d)). The box plots are drawn based on the Interquartile Range (IQR) of accuracy for 19 subjects when all the 128 electrodes are included in the experiment. The black horizontal line represents the median accuracy for each fold. In Fig. 4.3, the Wilcoxon signed rank test is applied for CT-HGR-V1 and CT-HGR-V2 separately when the number of channels is fixed at 128. The box plots show the IQR for each window size that decreases minimally from CT-HGR-V1 to CT-HGR-V2. The Wilcoxon test's p-value annotations in Fig. 4.3 are as follows:

- ns: $5.00e-02 < p <= 1.00e+00$
- *: $1.00e-02 < p <= 5.00e-02$
- **: $1.00e-03 < p <= 1.00e-02$
- ***: $1.00e-04 < p <= 1.00e-03$
- ****: $p <= 1.00e-04$

Although the average accuracy does not change significantly, the STD in CT-HGR-V2 with $W = 512$ declines significantly compared to CT-HGR-V1.

The gestures in the HD-sEMG dataset are ordered according to their DoF and similarity in performance. The simple 1 DoF gestures are labeled from 1 to 16, 2 DoF gestures are from 17 to 57 and the most complex ones are from 58 to 66. To be more specific, the confusion matrices for

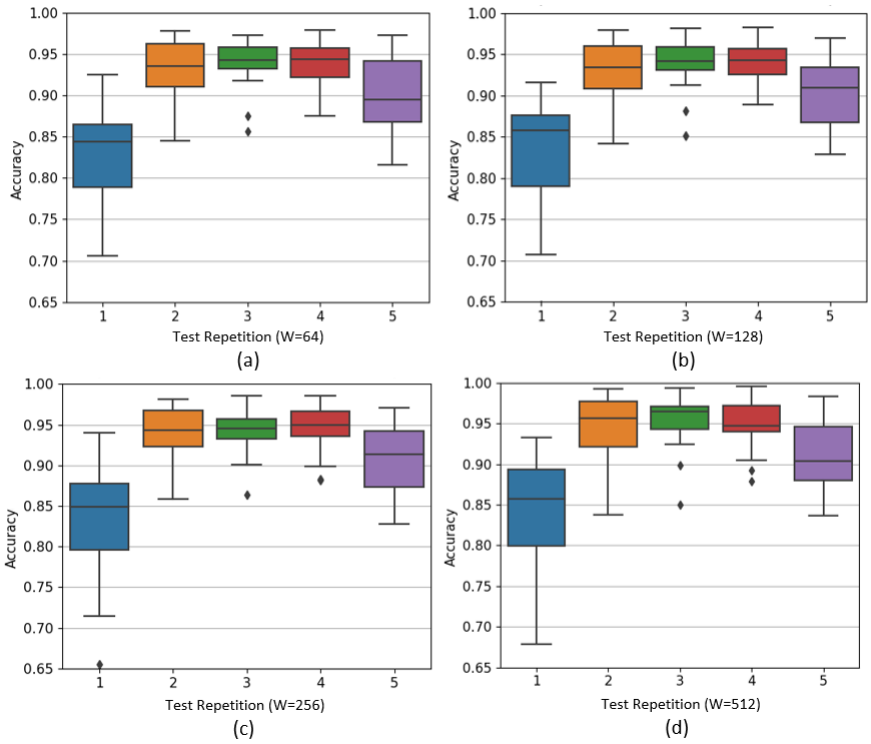

Figure 4.2: Comparison of the accuracy CT-HGR-V1 obtains for each fold and window sizes of (a) $W = 64$ (b) $W = 128$ (c) $W = 256$ and (d) $W = 512$. The number of utilized electrode channels in these plots is 128.

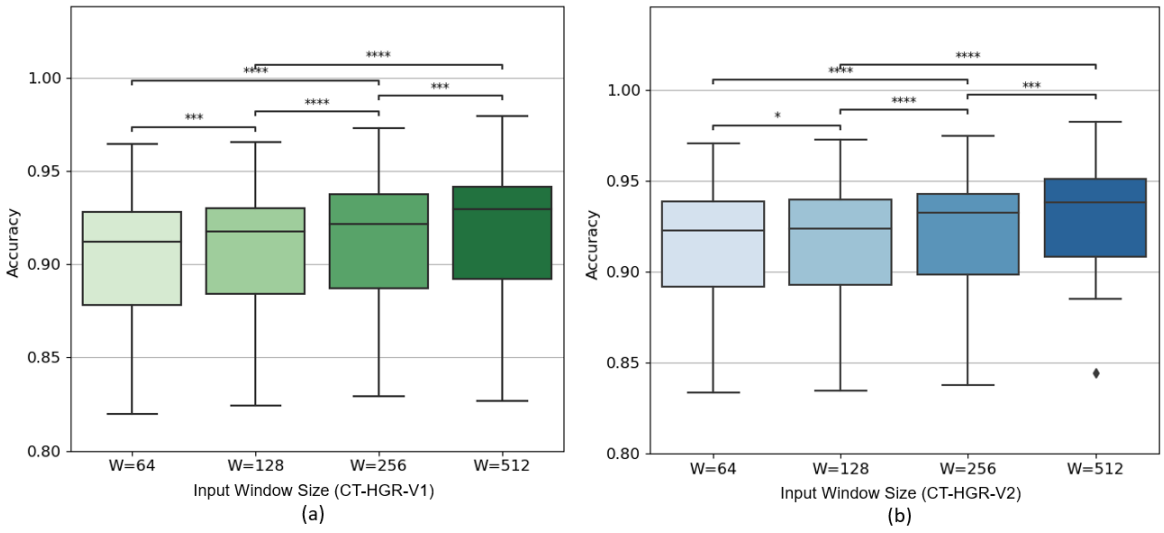

Figure 4.3: Statistical analysis of training over different window sizes, i.e., $W = 64$, $W = 128$, $W = 256$, and $W = 512$ for (a) CT-HGR-V1, and (b) CT-HGR-V2. The box plots are drawn based on the Interquartile Range (IQR) of the accuracy for all the subjects and all the electrodes.

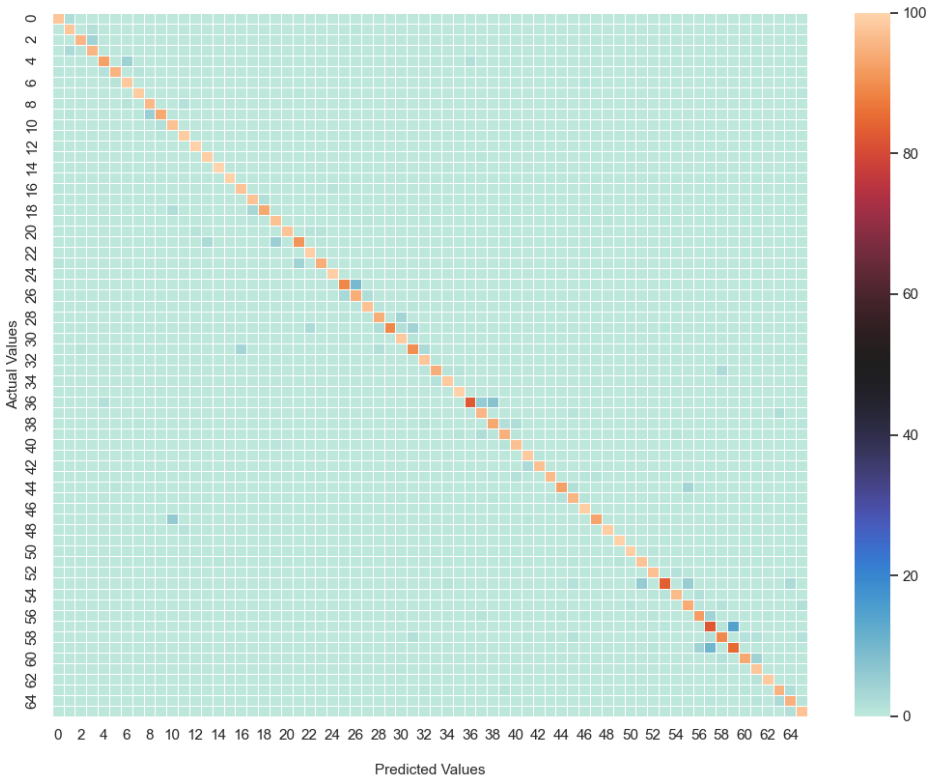

Figure 4.4: Average confusion matrix of Model CT-HGR-V1 with $W = 512$ and 128 number of electrodes over repetition 3 of 19 subjects.

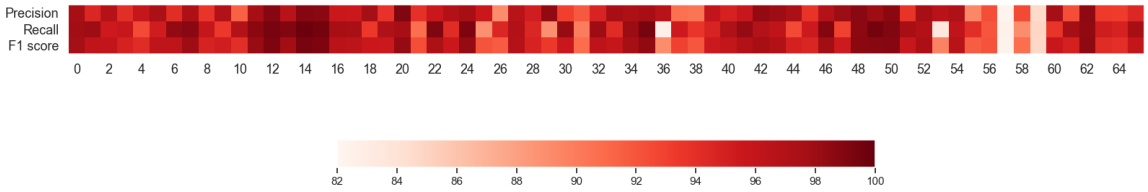

Figure 4.5: Representation of Precision, Recall and F1 Score of Model CT-HGR-V1 with $W = 512$ and 128 number of electrodes over repetition 3 of 19 subjects. These measures are obtained from the confusion matrix of Fig. 4.4 and shown for each class separately.

Model CT-HGR-V1 with $W = 512$ and 128 number of channels are obtained for repetition 3 of all the subjects. The matrices are summed and normalized row-wise. The final confusion matrix is shown in Fig. 4.4. The diagonal values show the average accuracy acquired for each hand gesture among 19 subjects. The average accuracy for most of the gestures is above 94%. The density of the non-zero elements in Fig. 4.4 is utmost near the diagonal, which implies that the possibility of the network making mistakes in gesture classification is higher in gestures that have the same DoF and are performed similarly. Fig. 4.5 represents precision, recall, and F1 score associated with

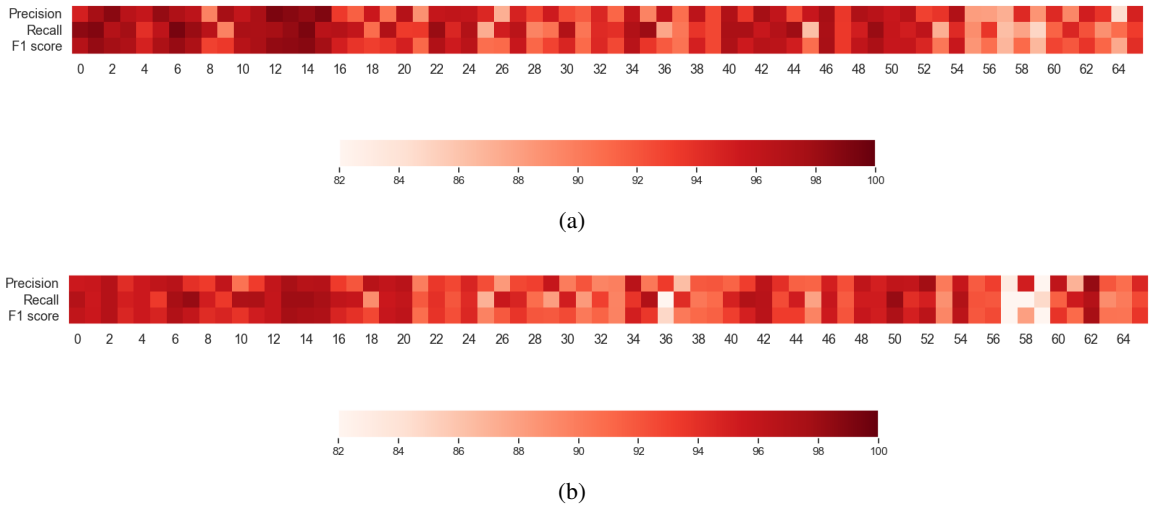

Figure 4.6: Representation of Precision, Recall and F1 Score with $W = 256$ and 64 number of electrodes over repetition 3 of all 19 subjects: (a) Model SVM-V1. (b) CT-HGR-V1.

Model CT-HGR-V1 for each gesture based on the confusion matrix shown in Fig. 4.4. This figure is included to provide the readers with a better sense of the gestures for which the above metrics were significantly high or low. Corresponding results for each gesture are illustrated in Table 4.5 and the average Matthews Correlation Coefficient (MCC) measure among all the subjects is calculated as 95.2%.

4.3.2 Comparisons with a Conventional ML and a 3D Convolutional Model

In the first part of this sub-section, we design two sets of traditional ML algorithms based on SVMs and LDAs, which are commonly [10, 85–88] used for hand gesture recognition tasks. In the first experiment and following [10, 85, 86], we trained SVM and LDA models based on the following set of classical features: Root Mean Square (RMS), Zero Crossings (ZC), Slope Sign Change (SSC), and Wave-length (WL). This experiment resulted in two models called SVM-V2 and LDA-V2. There are, however, some promising new feature extraction methods proposed in the recent literature [87–91]. To observe effects of recently proposed feature extraction methods, we did a second experiment based on features introduced in Reference [88]. These features are a rough estimate of the Power Spectral Density (PSD) of the signal by finding an approximate relation between the PSD in the frequency domain and the time-domain signal utilizing characteristics of

Table 4.5: Average Precision, Recall and F1 Score of Model CT-HGR-V1 with $W = 512$ and 128 number of electrodes over repetition 3 of all 19 subjects.

Class #	Precision(%)	Recall(%)	F1 Score(%)	Class #	Precision(%)	Recall(%)	F1 Score(%)
1	97.6 (±3.8)	97.8 (±5.3)	97.7 (±3.7)	**34**	97.5 (±4.8)	94.2 (±12.3)	95.8 (±8.3)
2	94.5 (±7.2)	97.9 (±9.7)	96.1 (±7.0)	**35**	97.2 (±4.4)	98.1 (±5.4)	97.7 (±3.9)
3	96.8 (±9.2)	95.4 (±15.3)	96.1 (±13.0)	**36**	97.5 (±4.7)	99.4 (±1.4)	98.4 (±2.5)
4	94.1 (±12.0)	95.6 (±9.4)	94.8 (±9.6)	**37**	96.7 (±8.7)	82.6 (±27.8)	89.1 (±23.2)
5	95.9 (±23.6)	92.5 (±23.7)	94.2 (±23.5)	**38**	90.6 (±12.7)	95.4 (±10.0)	92.9 (±9.9)
6	97.4 (±3.5)	95.2 (±11.2)	96.3 (±6.9)	**39**	90.4 (±13.7)	93.5 (±13.1)	92.0 (±12.7)
7	94.1 (±12.1)	98.4 (±3.5)	96.2 (±8.3)	**40**	96.0 (±7.3)	94.7 (±12.8)	95.4 (±10.4)
8	97.2 (±6.6)	98.7 (±2.1)	97.9 (±4.3)	**41**	94.8 (±6.7)	97.2 (±5.9)	96.0 (±5.3)
9	93.6 (±8.7)	95.9 (±9.1)	94.8 (±8.3)	**42**	95.6 (±7.8)	98.3 (±2.7)	96.9 (±4.8)
10	96.9 (±7.5)	93.4 (±11.4)	95.1 (±8.3)	**43**	98.3 (±2.6)	96.9 (±10.3)	97.6 (±6.9)
11	91.4 (±12.0)	96.9 (±12.7)	94.1 (±11.1)	**44**	98.0 (±3.3)	96.1 (±12.6)	97.0 (±8.8)
12	97.6 (±5.7)	98.7 (±2.9)	98.1 (±3.3)	**45**	97.4 (±8.3)	92.5 (±16.9)	94.9 (±13.3)
13	98.8 (±2.7)	99.4 (±1.2)	99.1 (±1.4)	**46**	93.7 (±9.1)	95.4 (±8.7)	94.5 (±8.2)
14	96.6 (±6.5)	99.0 (±2.0)	97.8 (±3.8)	**47**	96.9 (±4.5)	98.9 (±1.8)	97.9 (±2.6)
15	98.9 (±2.2)	99.7 (±1.0)	99.3 (±1.2)	**48**	98.0 (±22.4)	92.8 (±22.2)	95.3 (±22.1)
16	98.7 (±2.0)	99.5 (±1.5)	99.1 (±1.2)	**49**	98.7 (±2.0)	98.5 (±3.7)	98.6 (±2.3)
17	95.5 (±9.0)	97.5 (±4.8)	96.5 (±6.0)	**50**	98.0 (±3.5)	99.5 (±1.1)	98.7 (±1.9)
18	95.4 (±6.6)	97.4 (±9.5)	96.4 (±7.1)	**51**	98.6 (±2.8)	99.0 (±1.7)	98.8 (±1.7)
19	97.7 (±6.5)	93.5 (±9.7)	95.5 (±6.9)	**52**	93.7 (±11.4)	97.5 (±4.6)	95.6 (±8.1)
20	93.8 (±12.5)	97.2 (±6.2)	95.4 (±10.0)	**53**	97.6 (±3.9)	97.0 (±5.8)	97.3 (±4.4)
21	99.0 (±1.4)	97.5 (±5.0)	98.2 (±3.1)	**54**	96.6 (±19.1)	83.5 (±27.3)	89.6 (±26.3)
22	93.9 (±22.4)	90.7 (±23.1)	92.3 (±22.2)	**55**	97.2 (±7.6)	96.2 (±11.0)	96.7 (±9.2)
23	95.5 (±7.6)	99.1 (±3.0)	97.3 (±4.7)	**56**	89.3 (±15.3)	94.0 (±11.5)	91.6 (±12.7)
24	96.9 (±3.3)	94.2 (±12.4)	95.5 (±8.1)	**57**	92.3 (±14.2)	91.9 (±11.1)	92.1 (±12.6)
25	97.5 (±4.3)	99.1 (±1.2)	98.3 (±2.3)	**58**	82.2 (±15.2)	82.4 (±27.9)	82.3 (±25.6)
26	95.5 (±14.0)	88.8 (±25.4)	92.0 (±23.6)	**59**	92.5 (±11.5)	89.1 (±19.5)	90.7 (±15.7)
27	89.0 (±15.8)	94.5 (±10.0)	91.6 (±12.3)	**60**	84.6 (±15.3)	84.8 (±24.9)	84.7 (±20.5)
28	96.6 (±5.7)	97.0 (±5.1)	96.8 (±4.4)	**61**	97.6 (±4.0)	93.7 (±17.1)	95.6 (±13.2)
29	95.1 (±5.6)	94.5 (±14.4)	94.8 (±10.5)	**62**	92.3 (±11.5)	97.7 (±6.6)	94.9 (±8.1)
30	98.4 (±3.1)	88.8 (±19.6)	93.4 (±15.0)	**63**	98.5 (±2.7)	98.4 (±4.9)	98.5 (±2.9)
31	93.0 (±9.6)	98.2 (±2.4)	95.5 (±5.6)	**64**	93.3 (±8.8)	95.2 (±8.4)	94.2 (±7.0)
32	91.8 (±23.4)	89.9 (±25.4)	90.8 (±23.9)	**65**	93.2 (±8.2)	94.8 (±8.2)	94.0 (±6.7)
33	94.7 (±10.6)	98.0 (±3.5)	96.3 (±7.2)	**66**	94.4 (±9.2)	97.5 (±6.7)	96.0 (±6.6)

Table 4.6: Comparison of classification accuracy and STD for different window sizes and 64 electrode channels using CT-HGR-V1, 3D CNN, SVM-V1, SVM-V2, LDA-V1, and LDA-V2 models. The accuracy and STD are averaged over all the 5 folds and 19 subjects.

# Channels	Window size (samples)	CT-HGR-V1 (%)	3D CNN (%)	SVM-V1 (%)	SVM-V2 (%)	LDA-V1 (%)	LDA-V2 (%)
64	**64**	88.93 (±2.61)	86.15 (±2.95)	86.01 (±7.05)	74.49 (±11.56)	83.05 (±7.35)	71.40 (±12.45)
	128	89.29 (±2.58)	86.68 (±2.85)	89.95 (±5.19)	83.4 (±8.66)	87.97 (±5.38)	81.10 (±9.59)
	256	90.02 (±2.49)	87.45 (±2.77)	90.71 (±4.88)	87.77 (±5.84)	90.85 (±4.46)	86.72 (±7.37)

the Fourier transform and the Parseval's theorem. The procedures on how to extract these features from raw HD-sEMG data is explained in Section 4.2.

In the second part, we implement a 3D CNN model that is originally utilized for video-based

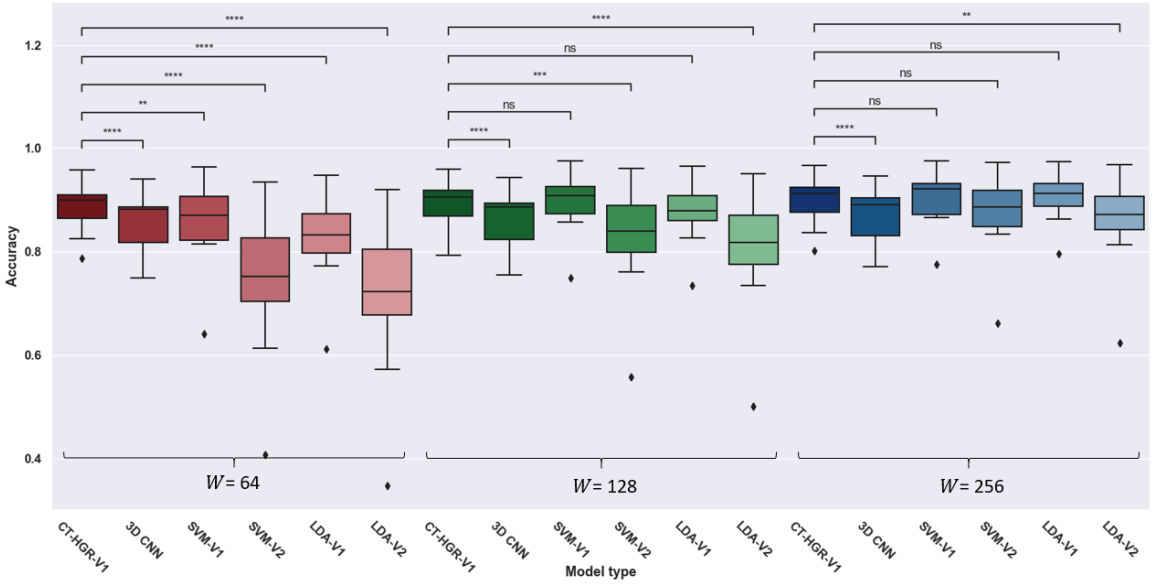

Figure 4.7: Box plots and IQR of CT-HGR-V1, 3D CNN, SVM-V1, SVM-V2, LDA-V1, and LDA-V2 for different window sizes ($W = 64$ $W = 128$ and $W = 256$) and 64 number of channels.

hand gesture recognition tasks [92] and is found effective by authors in [54] to be applied on HD-sEMG datasets as they resemble video data in having one dimension in time and two dimensions in space. Therefore, in spite of a typical 2D CNN model, a 3D CNN architecture is able to extract both the temporal and spatial features in HD-sEMG datasets. The 3D signals of shape $W \times N_{ch} \times N_{cv}$ go through the 3D CNN architecture that has two consecutive 3D CNN layers with 16 and 32 respective filters of size $(5, 3, 3)$, each followed by a GELU activation function, a dropout and a max pooling layer. Then, two fully connected (FC) layers of size 256 and 128 are deployed before the output layers which consists of an MLP head similar to the one used in our CT-HGR models followed by a $softmax$ function for classification. The other hyperparameters of the network are set similar to those of the CT-HGR model. The stride values in both 3D CNN layers are 1. Table 4.6 shows the acquired results for the ML and 3D CNN models in which the number of channels in the dataset is set to 64. For the case of ML models, Fig. 4.6 compares precision, recall, and F1 score metrics obtained from the best performing ML model (SVM-V1) with that of our proposed CT-HGR-V1 with the same settings ($W = 256$ and 64 number of electrode channels). The average MCC measure for SVM-V1 is calculated as 94.2% and for CT-HGR-V1 as 93.1%. Fig. 4.7 shows the box plots and the results of Wilcoxon signed rank statistical test that is conducted for comparing

Table 4.7: Comparison of train time, test time, and the maximum allocated memory for $W = 256$ and 64 electrode channels using CT-HGR-V1, 3D CNN, SVM-V1, SVM-V2, LDA-V1, and LDA-V2 models.

# Channels	Window Size (samples)	Parameter	CT-HGR-V1	3D CNN	SVM-V1	SVM-V2	LDA-V1	LDA-V2
64	256	Train Time (s)	382.9	1228.9	203.2	187.4	149.3	160
		Test Time (s)	69	8	237.3	374.7	31.6	36.2
		Memory (GB)	14.80	14.81	40.60	21.47	40.60	21.47

CT-HGR-V1, 3D CNN, SVM-V1, SVM-V2, LDA-V1, LDA-V2 model's performance accuracy on 19 subjects. In this experiment, the window sizes for all the models are changed ($W = 64$, $W = 128$ and $W = 256$), but the number of channels is fixed at 64. Therefore, only the models accepting the same window size as the input are compared to assess the discrepancy between two different models with the same input data.

When it comes to evaluation of the computational cost for DL models, the ultimate objective is to measure the needed amount of resources in training and inference. Computational cost can be measured in a variety of ways, among which time, memory and number of Floating Point Operations (FLOPs) are the common metrics. To evaluate computational cost of the proposed framework, in addition to the number of trainable parameters shown in Fig. 4.4, we have calculated the train time, test time and maximum allocated memory for each of the CT-HGR-V1, 3D CNN, SVM-V1, SVM-V2, LDA-V1, and LDA-V2 models, which are shown in Table 4.7. Please note that the train/test times reported in Table 7 correspond to the whole train/test data containing all segments of 256-sample windows. Considering 4 repetitions in the train set and 1 repetition in the test set for each subject, we have approximately $73,000$ and $18,000$ samples in the train and test set, respectively. This means that, CT-HGR-V1 for which the test time is reported as 69 seconds, needs 3.8 ms to predict each 256-sample window's corresponding gesture. We should point out that different factors, such as the GPU memory, how the code is organized, and the utilized batch size, can affect test time specifically in the small scale of each window size. It is also worth noting that memory bandwidth is considered instead of FLOPs because on existing hardware architectures, a single memory access is much slower than a single computation.

Table 4.8: Accuracy and STD for the shuffled dataset of all the 5 repetitions and different window sizes (CT-HGR-V1).

# Channels	Window size (samples)	# Avg accuracy (%)
64	**64**	98.05 (±1.19)
	128	98.43 (±1.05)
	256	98.79 (±0.96)

Table 4.9: Accuracy and STD of each fold and their average for instantaneous training.

# Channels	Window size (samples)	Fold1 (%)	Fold2 (%)	Fold3 (%)	Fold4 (%)	Fold5 (%)	Average (%)
64	**1**	80.02 (±3.45)	92.33 (±2.27)	92.47 (±2.26)	92.16 (±2.31)	88.69 (±2.74)	89.13 (±2.61)

4.3.3 Performance Evaluation based on Shuffled Data

In the previous sub-sections, a 5-fold cross-validation technique was applied on the HD-sEMG dataset in which the test set (repetition) is entirely unseen and is not included in the train set (repetitions). However, another approach followed in the literature [12, 93] to split the train/test sets is to shuffle the whole dataset with n repetitions and assign an arbitrary portion to the train set and the remaining to the test set. Along the same line, in some of the previous works [71, 72, 74] either the train/test splits were not specified or it was mentioned that data for each subject was shuffled and then randomly divided into train/test sets. Intuitively speaking, by shuffling the dataset across different repetitions, the model can better catch variations of the underlying signals and provide improved performance. In practice, the overall objective would be to have a generalizable model that works under different conditions as such one can acquire different repetitions and train the model over all to boost the performance. To observe effects of such a training approach on the overall achievable accuracy, we have decided to include such an experiment by shuffling the dataset. The results and observations are on a par with those reported in the aforementioned reference [71, 72, 74].The obtained average accuracy over 19 participants using 64, 128, 256 window sizes using the hyperparameters of CT-HGR-V1 are summarized in Table 4.8.

4.3.4 Instantaneous Performance Evaluation

In this sub-section, our objective is to assess the functionality of the proposed framework on instant HD-sEMG data points. In other words, we consider window size of only 1 sample as the input to our model, which requires no patching. We set the number of electrodes to 64. The

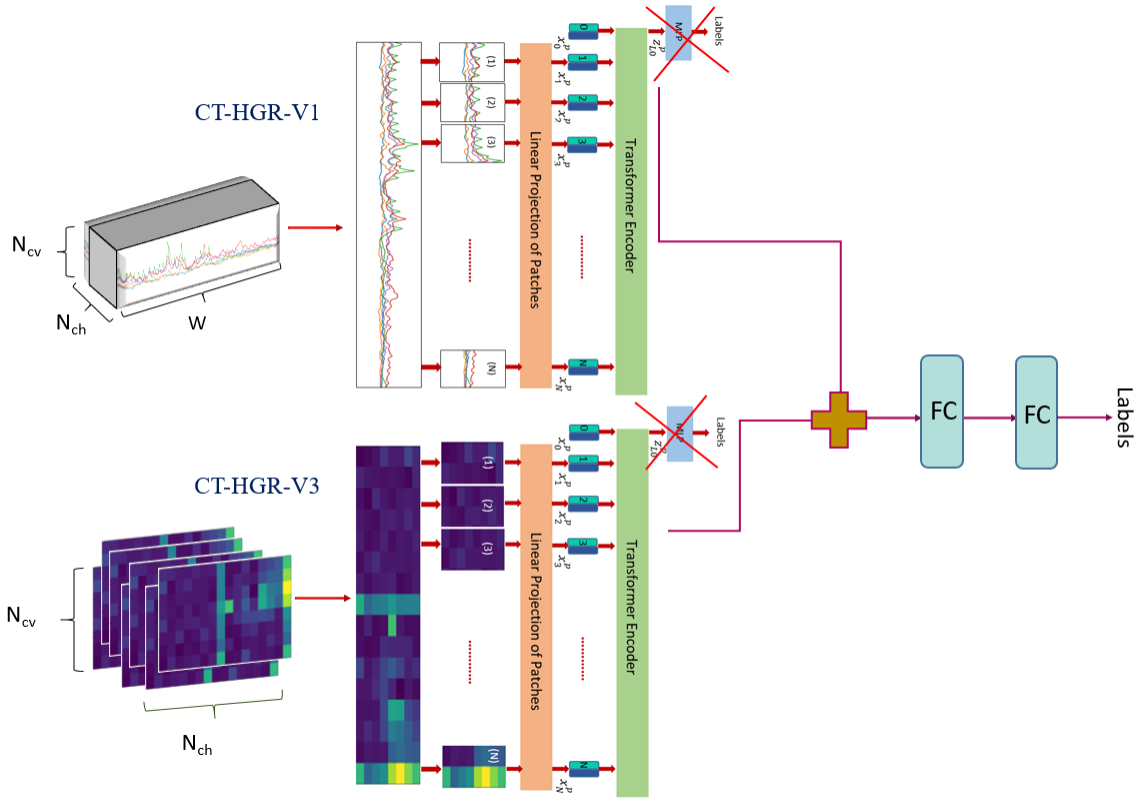

Figure 4.8: The fused CT-HGR framework. In the first stage, the ViT-based models in the Macro and Micro paths are trained based on 3D, HD-sEMG and 2D, p-to-p MUAP images, respectively. In the second stage, the Macro and Micro weights are frozen (not being updated with gradient descent during training). The final Micro and Macro class tokens are concatenated and converted to a $1,024$-dimensional feature vector, which is fed to a series of FC layers for gesture classification.

hyperparameters used in this experiment are the same as those used for CT-HGR-V1. The accuracy results are presented in Table 4.9.

4.3.5 Evaluation of a Hybrid Model based on Raw HD-sEMG and Extracted MUAPs

As mentioned previously, sEMG signals measure the electrical activities of the underlying motor units in limb muscles and are collected non-invasively from the electrodes placed on skin surface [94]. In particular, High Density sEMG (HD-sEMG) signals are acquired through a two-dimensional (2D) grid with a large number of closely-located electrodes [95], capturing both temporal and spatial information of muscle activities. HD-sEMG acquisition, therefore, provides superior spatial resolution of the neuromuscular system in comparison to its sparse acquisition counterpart. This has inspired targeted focus on development of DNN-based HGR methods based on HD-sEMG

signals [1, 18, 58, 84]. Broadly speaking, HD-sEMG-based Biological Signal Processing (BSP) approaches can be classified into the following two main categories:

(i) *Raw HD-sEMG Processing for HGR:* Algorithms belonging to this category directly use raw HD-sEMG signals for the task of HGR. In this context, e.g., Reference [84] performed instantaneous training of a Convolutional Neural Network (CNN) using a 2D image of a single time measurement. In [18], Recurrent Neural Networks (RNNs) are combined with CNNs to create a hybrid attention-based [37] CNN-RNN architecture, which has improved HGR performance due to joint incorporation of spatial and temporal features of HD-sEMG signals. Sun *et al.* [58] introduced a network of dilated Long Short-Term Memories (LSTMs) to classify hand gestures from the transient phase of HD-sEMG signals.

(ii) *HD-sEMG Decomposition:* The focus, here, is on HD-sEMG decomposition to extract microscopic neural drive information. HD-sEMG signals have encouraged emergence of sEMG decomposition algorithms in the last decade [29] as they provide a significantly high-resolution 2D image of Motor Unit (MU) activities in each time point. sEMG decomposition refers to a set of Blind Source Separation (BSS) [30] methods that extract discharge timings of motor neuron action potentials from raw HD-sEMG data. Single motor neuron action potentials are summed to form Motor Unit Action Potentials (MUAPs) that convert neural drive information to hand movements [31]. Motor unit discharge timings, also known as Motor Unit Spike Trains (MUSTs), represent sparse estimations of the MU activation times with the same sampling frequency and time interval as the raw HD-sEMG signals [32]. Extracted MUSTs are used in several domains such as identification of motor neuron diseases [33], analysis of neuromuscular conditions [34], and myoelectric pattern recognition [35].

A third category can be identified when the extracted MUSTs in Category (ii) are used for HGR at microscopic level. HD-sEMG signals are modelled as a spatio-temporal convolution of MUSTs, which provide an exact physiological description of how each hand movement is encoded at neurospinal level [36]. Thus, MUSTs are of trustworthy and discernible information on the generation details of different hand gestures, which leads to adoption of another group of HGR algorithms that accept MUSTs [96] as their input. Nevertheless, due to complexities of the decomposition stage and

added computational overhead, microscopic level HGR using MUST is less explored than models of Category (i), which use HD-sEMG signals at a macroscopic level. There are, however, some promising works [35,97,98] in which MUSTs carrying microscopic neural drive information are exploited for HGR instead of directly using raw sEMG signals. To discover a direct connection between different hand gestures and extracted MUSTs, these methods have suggested estimating MUAPs of the identified sources and extracting a set of useful features from MUAPs that are unique for each hand gesture. We should point out that the temporal profile of MUAPs obtained from MUSTs encode information about MU recruitments and the temporal profile of the EMG recordings. Therefore, using sliding windows for extraction of MUSTs informs us about the most current profile of the active MUs, their recruitments, and how much they are involved in each stage of performing the hand gestures. For instance, in [35], the peak-to-peak (p-to-p) values of MUAPs are calculated for each MU and each electrode channel separately and a 2D image of MUAP p-to-p are constructed for all the channels of a single MU. Afterwards, this 2D image is fed to a CNN architecture and its performance is compared to that of traditional ML methods. In short, using HD-sEMG decomposition results for HGR is still in its infancy, and in this section, we aim to further advance this domain.

Here, we present the results of fusing CT-HGR-V1 with a third variant of the CT-HGR called CT-HGR-V3 that works based on the extracted MUAPs from raw HD-sEMG signals. More specifically, CT-HGR-V3 uses HD-sEMG decomposition to extract microscopic neural drive information from HD-sEMG signals for hand gesture recognition. Multi-channel sEMG signals are generated as a convolutive mixture of a set of impulse trains representing the discharge timings of multiple MUs, i.e.,

$$\boldsymbol{x}_i(t) = \Sigma_{l=0}^{L-1}\Sigma_{j=1}^{N} h_{ij}(l)s_j(t-l) + \boldsymbol{\nu}_i(t), \tag{8}$$

where $\boldsymbol{x}_i(t)$ is the i^{th} channel's EMG data (from the entire M channels); $h_{ij}(l)$ is the action potential of the j^{th} MU (from the entire N extracted MUs) measured at the i^{th} channel; $s_j(t)$ is the MUST at the j^{th} MU, and; $\boldsymbol{\nu}_i$ is the additive white noise at channel i. Additionally, t is the time index; D is the duration of sEMG recordings; and L is the duration of MUAPs. Eq. (8) is represented

as

$$\boldsymbol{X}(t) = \Sigma_{l=0}^{L-1} \boldsymbol{H}(l) S(t-l) + \underline{\boldsymbol{\nu}}(t), \tag{9}$$

where $\boldsymbol{X}(t) = [\boldsymbol{x}_1(t), \boldsymbol{x}_2(t), \ldots, \boldsymbol{x}_M(t)]^T$ and $S(t) = [s_1(t), s_2(t), \ldots, s_N(t)]^T$ are the recordings of all the M electrode channels and the MUSTs of all the N extracted sources at time t, respectively. Term $\boldsymbol{H}(l)$ is the $(M \times N)$ matrix of action potentials, which is considered to be constant in duration D. The convolutive equality of Eq. (9) is the basic BSS assumption.

The objective is to find the maximum number of independent matrices $S(t)$ from Eq. (9) if $\boldsymbol{X}(t)$ is the only known parameter. Eq. (9) can be written in an instantaneous form, where the source vectors are extended with their $L-1$ delayed contributions. Additionally, to adapt the model for BSS conversions, the observation vectors are extended with their T delayed versions, resulting in the following final convolutive model

$$\widetilde{\boldsymbol{X}}(t) = \widetilde{\boldsymbol{H}}\, \widetilde{S}(t) + \underline{\widetilde{\boldsymbol{\nu}}}(t), \tag{10}$$

where each of the $\widetilde{\boldsymbol{X}}(t)$, $\widetilde{\boldsymbol{H}}$, $\widetilde{S}(t)$, and $\underline{\widetilde{\boldsymbol{\nu}}}(t)$ are the extended versions of the observation, MUAPS, sources, and noise matrices, respectively. Among the existing BSS approaches [29] suggested for HD-sEMG decomposition, gradient Convolution Kernel Compensation (gCKC) [99, 100] and fast Independent Component Analysis (fastICA) [101] are of great prominence and frequently used in the literature. To achieve better accuracy, the utilized BSS algorithm [29] is a combination of gCKC [99, 100] and fastICA [101] algorithms. In the gCKC method, the MUSTs are estimated using a linear Minimum Mean Squared Error (MMSE) estimator as follows

$$\hat{s}_j(t) = \hat{\boldsymbol{c}}_{s_j x}^T C_{xx}^{-1} \boldsymbol{x}(t), \tag{11}$$

in which $\hat{s}_j(t)$ is the estimate of the j^{th} MUST at time t, $\hat{\boldsymbol{c}}_{s_j x} \approx \boldsymbol{E}\{\boldsymbol{x}(t) {s_j}^T(t)\}$ is approximation of the unknown cross-correlation vector between the MUSTs and the observations, and $C_{xx} = \boldsymbol{E}\{\boldsymbol{x}(t)\boldsymbol{x}^T(t)\}$ is the correlation matrix of observations. Term $\boldsymbol{E}\{\cdot\}$ indicates the mathematical expectation. According to Eq. (11), as $\hat{\boldsymbol{c}}_{s_j x}$ is unknown, a blind estimation of MUSTs is iteratively

found with gradient descent [99]. On the other hand, in the fastICA, the goal is to estimate separation vectors w such that

$$\hat{s}_j(t) = w_j^T(k)\boldsymbol{Z}(t), \tag{12}$$

where $\hat{s}_j$ is the j^{th} MUST; w_j is the j^{th} separation vector; and $\boldsymbol{Z}$ is the whitened matrix of observations. The separation vectors are identified through the fixed-point optimization algorithm [29, 101]. Note that term k in Eq. (12) denotes the separation vector identifying fixed-point iterations. In the method utilized here, the number of extracted sources is dependent on the following two different parameters that are determined before initiating the algorithm: (i) The number of iterations of gCKC and fastICA algorithms in which a new MU is found, and; (ii) The silhouette threshold, which determines whether the extracted MU is of high quality to be accepted or ignored. As stated in [35, 97], the activation level/area of MUs in limb muscles is highly variable across different hand gestures. Accordingly, if the p-to-p values of MUAPs for each MU and all the channels are calculated, a set of 2D images can be acquired, which have a predictable pattern among different hand gestures. Therefore, after extracting the MUSTs of HD-sEMG signals, the corresponding MUAPs are found using Spike-Triggered Averaging (STA) method [97] with an averaging window of 20 samples. As stated in [29], extension factor T in Eq. (10) multiplied by the number of sEMG channels should be greater than the number of extracted sources multiplied by the length of MUAPs. Furthermore, it is empirically shown that extension factors greater than 16 have almost the same impact on the number and quality of extracted MUSTs. Therefore, we set extension factor to 20 to be greater than $\frac{N \times L}{M}$. Then, the p-to-p values of the MUAPs are calculated and a 2D image of shape $N_{ch} \times N_{cv}$ is constructed for each MU.

Below, shows the list of operations done to implement the BSS algorithm in [29] and extract MUSTs from raw HD-sEMG signals:

1. The mean is subtracted from the observations $\widetilde{\boldsymbol{X}}(t)$.
2. $\widetilde{\boldsymbol{X}}(t)$ is whitened and converted to $\boldsymbol{Z}(t)$.
3. Separation matrix B is defined as an empty matrix.
4. For $i = 1, 2, \ldots, M$ repeat:

- Separation vectors $w_i(0)$ and $w_i(1)$ are randomly initialized.
- While $|w_i(n)^T w_i(n-1) - 1| < Tol$ and $n < max\_iterations$ do:

  a. Separation vector $w_i(n)$ is defined as: $w_i(n)$= $\boldsymbol{E}\{\boldsymbol{Z}g[w_i(n-1)^T\boldsymbol{Z}]\} - \widetilde{A}w_i(n-1)$ when $\widetilde{A} = \boldsymbol{E}\{g^{'}[w_i(n-1)^T\boldsymbol{Z}]\}$.

  b. Separation vector $w_i(n)$ is orthogonalized as: $w_i(n) = w_i(n) - BB^T w_i(n)$.

  c. Separation vector $w_i(n)$ is normalized as: $w_i(n) = \frac{w_i(n)}{||w_i(n)||}$.

  d. $n = n + 1$
- End while
- Cov_n and Cov_{n-1} are randomly initialized.
- While $Cov_n < Cov_{n-1}$ and $n < max\_iterations$ do:

  a. The i-th source is estimated as: $\hat{s}_i(t) = w_j^T(\boldsymbol{n})\boldsymbol{Z}(t)$.

  b. The ST_n is estimated using peak detection and K-means method.

  c. Cov_n is updated by calculating covariance of ST_n.

  d. Separation vector is updated as: $w_i(n+1) = \frac{1}{J}\Sigma_{j=1}^{J}\boldsymbol{Z}(t_j)$ where t_j is when ST_n is equal to 1.

  e. $n = n + 1$
- End while
- If $Silhoeutte > 0.92$ do:

  a. The source estimate in the previous step is accepted.

  b. Separation vector w_i is added to matrix B.

5. End for

A summary of the adopted procedures from taking the raw HD-sEMG signals to calculating the MUAP p-to-p images is shown in Fig. 4.9(a). Fig. 4.9(b-d) illustrate the extracted MUAPs for a single MU of the first 512-sample window of gesture 1 (bending the little finger), 2D image of

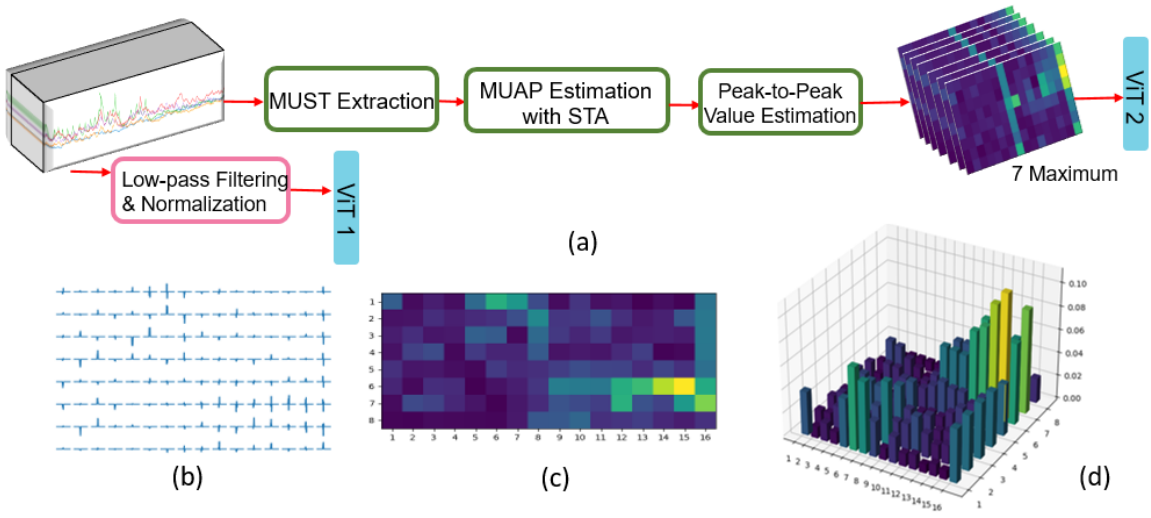

Figure 4.9: (a) Diagram of the adopted procedures for obtaining MUAP p-to-p images. (b) MUAPs for a single MU of the first windowed signal corresponding to the first repetition of gesture 1 (bending the little finger). The MUAPs are estimated/shown for each channel separately. (c) p-to-p values of MUAPs represented as a 2D image. (d) 3D representation of MUAP p-to-p values.

their p-to-p values, and a 3D representation of the p-to-p values, respectively. As can be seen, the muscles under the electrodes of the extensor grid were more active in the course of bending the little finger.

The fused variant of the CT-HGR is designed to simultaneously extract a set of temporal and spatial features from HD-sEMG signals through its two independent ViT-based parallel paths. The former is the CT-HGR-V1 that accepts raw HD-sEMG signals as input, while the latter is the CT-HGR-V3 fed with the p-to-p values of the extracted MUAPs of each source. A fusion path, structured in series to the parallel ones and consisting of FC layers, then combines extracted temporal and spatial features for final classification. Fig. 4.8 illustrates the overall hybrid architecture of the fused model. In particular, CT-HGR-V1 extracts both temporal and spatial features of HD-sEMG signals as it is fed with time-series raw HD-sEMG signal that are variable both in terms of time and space. However, the CT-HGR-V2 can extract another set of spatial features from p-to-p values of MUAPs that are variable in space.

In our experiments, the number of iterations (Item (i)) is set to 7 and the silhouette measure (Item (ii)) is set to 0.92, therefore, depending on the quality of the extracted MUSTs, a maximum of 7 sources are estimated for each windowed signal. Therefore, each windowed signal of shape

Table 4.10: Comparison of classification accuracy and STD for each fold and their average for each of the 3 models. The accuracy and STD for each fold is averaged over 19 participants.

Model Name	Fold1 (%)	Fold2 (%)	Fold3 (%)	Fold4 (%)	Fold5 (%)	Average (%)
CT-HGR-V1	79.92 (±3.39)	91.43 (±2.48)	93.84 (±2.05)	92.57 (±2.28)	88.96 (±2.83)	89.34 (±2.61)
CT-HGR-V3	81.53 (±3.45)	88.03 (±2.66)	89.63 (±2.39)	89.11 (±4.02)	84.92 (±2.97)	86.64 (±3.10)
Fused	**89.38 (±2.88)**	**96.86 (±1.82)**	**96.82 (±1.75)**	**96.65 (±2.75)**	**94.61 (±1.90)**	**94.86 (±2.22)**

$W \times N_{ch} \times N_{cv}$ is of maximum 7 MUs that retain various activation levels for each electrode channel. These 2D images are considered as new input data to the CT-HGR-V3. Thus, according to Fig. 4.8, for each windowed signal that is fed to CT-HGR-V1, a maximum of 7 p-to-p MUAPs are created and fed to CT-HGR-V3. After training CT-HGR-V1 and V3 independently, the models' weights are frozen, i.e., are kept constant (not being updated with gradient descent during training) and the final classification linear layer is removed for both models. Then, the final class tokens of CT-HGR-V1 and CT-HGR-V3 are joined together and fed to a FC layer for final classification. In this way, the hybrid model decides based on raw HD-sEMG signals as well as p-to-p images of MUAPs obtained for each MU independently. The CT-HGR-V3's hyperparameters are set as follows: For both CT-HGR-V1 and V3, HD-sEMG data is divided into windows of shape (512,8,16) with skip step of 256. Therefore, the image size and the number of input channels for 2D images are set to (8×16), and 1, respectively. For each p-to-p image, we considered 2 patches by setting patch size to (8×8). The model's embedding dimension (d) and number of heads is the same as the two previous models. The optimization algorithm is Adam with learning rate of 0.0003 and weight decay of 0.001. Each batch has 64 data samples and the model is trained through 50 epochs. Table 4.10 compares accuracy and STD for CT-HGR-V1, CT-HGR-V3 and their fused model for each fold. The box plots showing accuracy and Interquartile Range (IQR) measured for 19 subjects is represented in Fig. 4.10 for each model. It is worth mentioning that authors in [102] have adopted a quite similar approach to ours by combining activations of individual DoFs (obtained from decomposed MUSTs) with residual HD-sEMG signals for predicting wrist DoF angles using a linear regression method. The main distinctions between the two methods are as follows: (i) The method of [102] focuses on predicting DoF angles in wrist kinematics and not gesture recognition, and (ii) Considered combining residual HD-sEMG signals with DoF activations, which is a different concept from combining p-to-p MUAPs with original HD-sEMG signals.

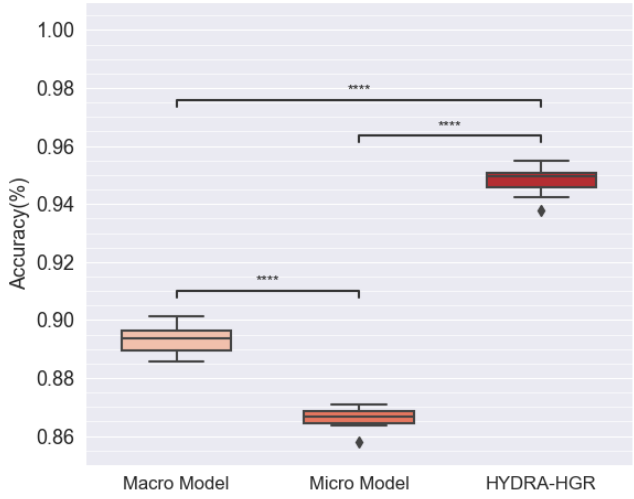

Figure 4.10: Boxplots and IQR of the 3 models over all the 19 subjects.

4.3.6 Comparison with Other Works on The Utilized Dataset

In this section, we compare our proposed CT-HGR model with 4 other works [71–74] that proposed ML/DL methods for hand gesture recognition based on the same dataset utilized in this study. Sun, *et al.* [73] proposed three different CNN-based models for hand gesture recognition with 1D, 2D and 3D convolutional layers that are applied on both transient and steady phases of HD-sEMG data. In our study and differently from [73], we jointly considered the transient and steady phases of the sEMG signals when providing the input to the model, therefore, data distribution should be different. We, however, compared our results with their steady phase as there is more similarity between these two types in comparison to the transient phase. Using a window size of 200 ms, all the 128 electrode channels, and the same 5-fold cross validation technique as we implemented, the maximum median accuracy obtained by the model of [73] is 84.6% whereas the proposed framework obtained 91.98% accuracy for 250 ms window and 128 electrode channels. In [71], a similar study to ours is conducted by changing the window size and the number of channels to evaluate their effect on the performance of the model. In this paper, 5 time-domain features of the signal along with sixth-order autoregressive coefficients are extracted and given to an LDA model. Average accuracy of 81.39% is obtained for the window size of 32 ms when 32 channels were used. The accuracy increases to 91.5% for the same window size with 128 channels. It finally reaches 96.14%

Table 4.11: Comparison of classification accuracy and STD obtained by the other works on our utilized dataset with CT-HGR-V1 and CT-HGR-V2.

Reference	Window size (ms)	# Channels	Accuracy (%)	Train/Test Split
Ref [73]	200	128	84.6 (NA)	5-fold Cross Validation
CT-HGR-V1	250	128	91.98 ($\pm$2.22)	5-fold Cross Validation
CT-HGR-V2	250	128	92.88 ($\pm$2.10)	5-fold Cross Validation
Ref [71]	32	32	81.39 ($\pm$10.77)	NA
CT-HGR-V1	31.25	32	86.23 ($\pm$2.94)	5-fold Cross Validation
Ref [71]	256	128	96.14 ($\pm$4.67)	NA
CT-HGR-V1	250	128	91.98 ($\pm$2.22)	5-fold Cross Validation
CT-HGR-V2	250	128	92.88 ($\pm$2.10)	5-fold Cross Validation
Ref [72]	31.7	128	91.25 ($\pm$4.92)	NA
CT-HGR-V1	31.25	128	90.53 ($\pm$2.43)	5-fold Cross Validation
CT-HGR-V2	31.25	128	91.51 ($\pm$2.35)	5-fold Cross Validation
Ref [74]	32	128	94 (NA)	NA
CT-HGR-V1	31.25	128	90.53 ($\pm$2.43)	5-fold Cross Validation
CT-HGR-V2	31.25	128	91.51 ($\pm$2.35)	5-fold Cross Validation
Ref [74]	256	128	97.2 (NA)	NA
CT-HGR-V1	250	128	91.98 ($\pm$2.22)	5-fold Cross Validation
CT-HGR-V2	250	128	92.88 ($\pm$2.10)	5-fold Cross Validation

for the 256 ms window and 128 channels with minimum STD of 3.82%. We should note that autoregressive coefficient extraction could be a time-consuming process for HD-sEMG data potentially slowing the learning process. Along a similar path, Reference [74] introduced a new feature extraction approach using Wavelet Scattering Transform, applied an SVM model on the extracted features and compared their results with that of [71]. The results show an increase in the accuracy for different window sizes and 128 electrode channels which is $\approx$ 94% and 97.2% for 32 ms and 256 ms window sizes, respectively. We should note that in these works, the utilized method for splitting the train/test data is not explicitly specified. A Graph Neural Network approach is adopted in [72] with window sizes of 65 samples using 128 channels resulting in the average accuracy of 91.25% with STD of 4.92%. Using the same setting, we acquired accuracy of 90.53% and STD of 2.43% with CT-HGR-V1 and 91.51% and STD of 2.35% with CT-HGR-V2. When it comes to train/test datasets, it is mentioned in [72] that data for each subject was shuffled and then randomly divided into train/test sets. Table 4.11 represents the average accuracies obtained by the above-mentioned papers and the settings they utilized to assess their performance. If the STD and train/test split is not mentioned in the paper, “NA" (Not Applicable) is shown.

4.4 Discussion

Based on the results shown in Table 4.2 and Table 4.3, the accuracy for each fold and the average accuracy increases by increasing both the window size and the number of channels. Doubling the number of electrode channels from 32 to 64 results in $2-3\%$, and from 64 to 128 in $1-2\%$ increase in all the reported accuracies. Intuitively speaking, on the one hand, increasing the window size feeds more data to the model at each epoch, which can enhance its performance as the difference among various gestures is more detectable through larger window sizes. On the other hand, instead of increasing the skip step while increasing the window size, we kept the skip step constant at 32 to feed more data to the model. In this scenario, the model has access to much more different samples of the training data as such possibly better learns the underlying representations of the data compared to the scenario where the skip step is larger but the model is fed with fewer data samples. Therefore, the model could be more generalizable while avoiding overfitting over to the train samples. Generalization refers to the ability of the model to make correct predictions for previously unseen data samples. More specifically, although the model is tested with completely unseen data samples, it has seen more samples during the training phase as such should be able to more effectively detect the underlying patterns among different gestures as such perform better on the unseen test data. The small skip step (32) chosen here means that the predictions are made every 15.3ms, causing a very small latency for real-time implementation of the proposed network in prosthetic devices. As it is evident from Table 4.2, starting from 86.23%, the average accuracy increases by $0.3-0.8\%$ each time the window size is increased reaching 91.98% when the window size and the number of channels are at the maximum. Therefore, the number of utilized channels, in general, has a greater impact on the accuracy in comparison to the window size. Moreover, the smallest accuracy is for $Fold1$ while the highest is for $Fold3/Fold4$, which could be due to the fact that in the first repetition, the subject was not completely aware of the procedure and how to exactly perform the required gesture. Intuitively speaking, the subject was being trained to perform the requested task. We hypothesize that, in the 3^{rd} and 4^{th} repetitions, the subject might have completely learned about the gesture and performed it more consistently, however, in the 5^{th} repetition, fatigue might be a factor resulting in lower performance and relatively large drop in the accuracy.

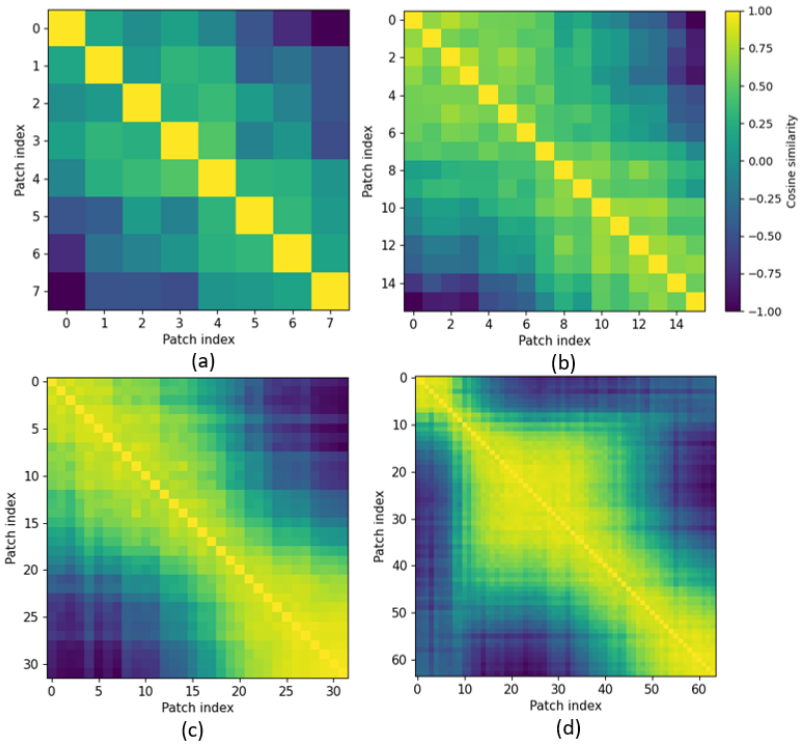

Figure 4.11: Cosine similarities of repetition 3, subject 20 of CT-HGR-V1 for (a) $W = 64$ (b) $W = 128$ (c) $W = 256$ and (d) $W = 512$

As can be seen from Fig. 4.2, choosing the first repetition as the test set considerably differs from choosing the third or fourth repetition as the former yields much lower accuracy on average. STD for each fold and their average follows the same pattern as that of the accuracy, however, in an opposite direction, meaning that the best accuracy is usually associated with the least STD. This issue justifies the difference between the acquired accuracy in our proposed CT-HGR-V1 model with that of References [71, 72, 74] using the same HD-sEMG dataset [2]. As mentioned before, two ML/DL models could be fairly comparable only if their train/test datasets are similar.

As can be seen in Table 4.3, Model CT-HGR-V2 is generally a better model compared to its CT-HGR-V1 variant as the accuracy for each fold and the overall average are higher. This is because CT-HGR-V2 is a bigger model with larger embedding dimension than CT-HGR-V1 in which the variations among different patches are more effectively embedded helping it to better discriminate between different hand gestures. Nevertheless, while the best improvement in accuracy occurs for $Fold1$ with $\approx 1.5\%$ increase compared to CT-HGR-V1, not much improvement (less than 1% in

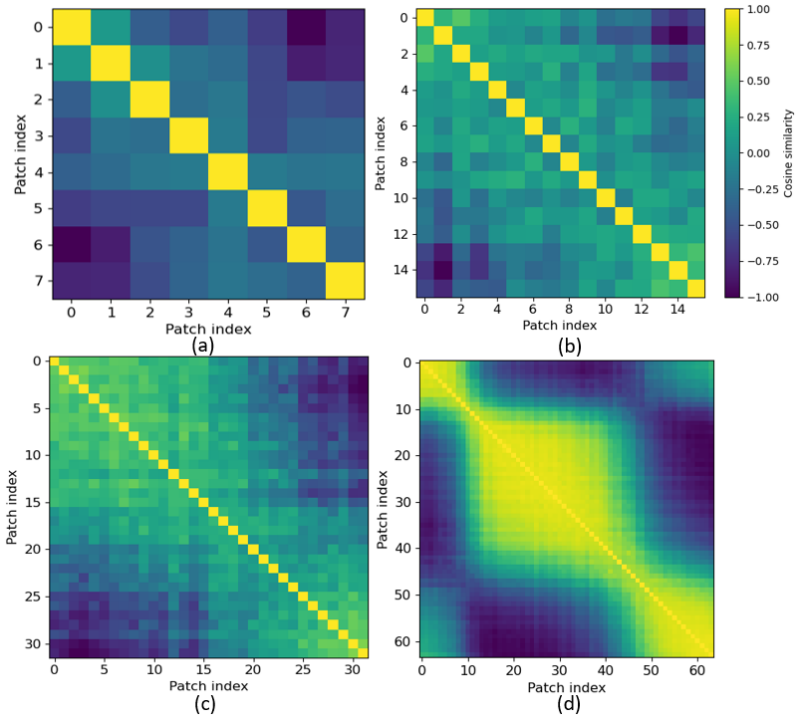

Figure 4.12: Cosine similarities of repetition 3, subject 20 of CT-HGR-V2 for (a) $W = 64$ (b) $W = 128$ (c) $W = 256$ and (d) $W = 512$

most cases) is observed in the other folds and the final average. As indicated in Table 4.4, CT-HGR-V2's number of learnable parameters is roughly 3 times the number of learnable parameters of CT-HGR-V1, however, there is a marginal progress in its performance in comparison to the former model. This shows that the hyperparameters used in CT-HGR-V1, producing no more than $100,000$ learnable parameters for the model, are sufficient for learning the 66 hand movements with high accuracy and there is no need to use more complex models for hand gesture classification using the proposed CT-HGR framework on this specific HD-sEMG dataset. Clearly, deploying more complex models takes more memory and training time, which in turn reduces the overall efficiency of the model. According to the box plots shown in Fig. 4.3, all the comparisons between different window sizes are statistically significant. According to our results and those of [71], in the case of HD-sEMG data, changing the window size has a great impact on the model's accuracy in contrast to sparse sEMG signals. In HD-sEMG signals, thanks to using large number of electrode channels, there exists valuable information about differentiable patterns among hand gestures even in small window sizes. We should also mention that there exists a direct link between the window size

and responsiveness in prosthetics [103]. For CT-HGR-V2, we have $p \leq 0.001$ for the $W = 64$ / $W = 128$ and $W = 256$ / $W = 512$ pairs, which is less statistically significant than the other pairs with $p \leq 0.0001$. For CT-HGR-V2, the results for the $W = 64$ / $W = 128$ pair are with $p \leq 0.05$ which is less statistically significant than that for the other pairs. In our experiments, we aimed to verify that our proposed model can extract the underlying patterns in a single sample or very small portion of HD-sEMG data while these patterns are not easily discernible in sparse sEMG data. Although this may not be widely used in today's real-time HMI devices, it can be a potential field of research and development of the current devices for window sizes of 2 ms and below to evaluate user's experience.

As mentioned previously, the positional embedding used in the CT-HGR framework is a 1D trainable embedding vector that is added to each of the embedded patches. By increasing the window size in our experiments, the patch size remains constant and the number of patches increases. This causes the positional embedding, which is the principal factor in determination of the input samples' succession, to learn the positions more precisely. Fig. 4.11 illustrates the cosine similarity matrices of the positional embedding in Model CT-HGR-V1. Cosine similarities are sketched for different window sizes, 128 electrode channels and the trained model on subject 20 when repetition 3 is considered as the test set. In this case, models with window sizes of 64, 128, 256, and 512 have $(8, 16)$ patch sizes. Therefore, each contain 8, 16, 32 and 64 patches in total. The x and y coordinates show the patch indices for each case and each row shows the similarities between each patch and the other patches. The diagonal values in each matrix are the largest values because their positional embedding vector is the same and its cosine is maximum. Similarity in the learned positional embedding vector of patches declines as the patches become farther. For $W = 512$, the model learns the positions better and cosine similarities change more smoothly. Fig. 4.12 demonstrates the cosine similarity matrices of the positional embedding in Model CT-HGR-V2. Evidently, Model CT-HGR-V2 has learned the position embeddings more effectively as there is less similarity between the distant patches for all the window sizes. The more the window size increases, the more the model discriminates between the distant patches and the more the adjacent patches are considered similar to each other. As illustrated in Fig. 4.11 and Fig. 4.12, for $W = 512$, Model CT-HGR-V2 behaves in a more orderly fashion than Model CT-HGR-V1 and consequently, extracts

the positional information better.

Regarding instantaneous training, authors in [84] implemented a CNN to conduct instantaneous classification of 8 gestures in the CapgMyo DB-a dataset. They applied various pre-processing and hyperparameter tuning steps and achieved the best performance of 89.3 for 18 subjects and 8 different gestures when all the 128 channels of the electrode grid were utilized. However, we achieved average accuracy of 89.13% for 19 subjects and 66 hand gestures with 64 channels. It is worth mentioning that 89.13 for 19 subjects and 66 gestures is achieved with the lightest version of our framework. Based on the results shown in Table 4.9, no significant discrepancy between the results for instantaneous training and larger window sizes is found. The results, in this case, are very similar to that of CT-HGR-V1, when W=128 and number of channels is equal to 64. This suggests that instantaneous training can sometimes work even better than training on very large window sizes with our proposed framework. More specifically, the model is able to achieve high accuracy in learning 66 hand movements with a single-point input which can be considered as an important breakthrough in the field of hand gesture recognition. This proves that HD-sEMG datasets provide highly valuable information of the muscles' activity in each time point which are sufficient for the model to learn various hand gestures with no need for larger window sizes. Furthermore, training with single-point windows of data provides a great number of input samples to the CT-HGR which helps the model generalize better and avoid overfitting. Based on the results shown in Table 4.8, the average accuracy and STD with shuffling is $\approx 9\%$ higher and $\approx 1.4\%$ lower than the results of the 5-fold cross-validation, respectively. This, however, can cause major issues in practice when dealing with hand prosthetic devices since the test data is entirely unseen and the pre-trained model could not perform reliably while testing with new datasets. In other words, the results reported without shuffling should be used as the bases for practical utilization.

Based on the results shown in Table 4.6 and Fig. 4.7, contrary to CT-HGR, increasing the window size leads to significant improvements in the average accuracy of the conventional ML models. In general, the achieved accuracy for the best performing ML models, i.e., SVM-V1 and LDA-V1 (trained with a newly proposed set of features), is $3-6\%$ lower and $0.5-0.8\%$ higher than CT-HGR-V1 with $W=64$ and $W=256$, respectively. Furthermore, as indicated in Table 4.6 and Table 4.4, our proposed CT-HGR-V1 framework surpasses the 3D CNN model by $\approx 3\%$ average

accuracy while employing less than $1/4$ of the learnable parameters used in the 3D CNN model. According to Table 4.6 and Fig. 4.7, the accuracy of both the deep networks (CT-HGR-V1 and 3D CNN) increases by less than 1% with doubling the window size. As shown in Fig. 4.7, there is statistically significant difference among the six models with window size of 64 ($p \leq 0.0001$), implying that the proposed CT-HGR-V1 gives its best performance at smaller window sizes. For $W = 128$, the difference between CT-HGR-V1 and SVM-V1 and LDA-V2 is not significant although these models achieve twice the STD of CT-HGR-V1. The proposed CT-HGR-V1 model seems to perform similarly to SVM-V1, LDA-V1 and SVM-V2 models when the window size is set to 256 as the Fig. 4.7 shown no significant discrepancy in the average accuracy of these models. In this case, there is still significant difference between CT-HGR-V1 and 3D CNN architectures with $p \leq 0.0001$.

According to Table 4.7, the train and test times for the two LDA models are less than that of CT-HGR-V1 while the maximum allocated memory for ML models with the second set of features that resulted in better accuracy is much higher than the maximum memory requirement of the CT-HGR-V1. This can be attributed to fact that the process of extracting five features from each channel of the HD-sEMG signals requires a great amount of system memory. On the contrary, DL-based models do not need a separate feature extraction step and the input windowed signals are the only item that needs system's memory allocation. It is worth nothing that when it comes to the train time, CT-HGR-V1 needs 20 epochs to secure the minimum loss and the best convergence of the model. However, if the CT-HGR-V1 model is run with even 10 epochs, the accuracy drops around 0.8%, but the train time halves, i.e., 189 seconds. As stated previously, the train and test times are calculated in seconds for training the whole signal of one complete repetition for one subject. The batch size used for the testing stage of the CT-HGR-V1 is set equal to that of the training phase, i.e., 128. This impacts the test time of the CT-HGR-V1 (with larger batch sizes, the test time should reduce) compared to the ML models where the whole test data is provided at once. As can be seen in Table 4.7, the test time for the 3D CNN model is the least, but it has much larger training time, larger number of trainable parameters and less accuracy in comparison to CT-HGR-V1.

Based on Fig. 4.6, CT-HGR-V1 architecture performs poorly for gestures 57 and 59 as it achieves low precision, recall and F1 score for these two gestures. Gesture 36, also, in this model

has a low recall measure implying that of all the samples that were labelled as class 36, not a great number of them were labelled correctly by CT-HGR-V1. SVM-V1 model was also incapable of effectively classifying gestures 57 and 59, but acted more precisely than CT-HGR-V1 for these gestures. This model, however, performs worse than CT-HGR-V1 on gesture 64 in terms of precision and F1 score. According to Table 4.10 in which the studies are reported for the 250 ms window size, CT-HGR-V1's accuracy is higher than that of the CT-HGR-V3 by $\approx 3-4$ %, except $Fold1$ for which the p-to-p values of MUAPs provide more accurate information of the performed hand gesture than the HD-sEMG signals. However, a great improvement in average performance of the fused model in comparison to both stand-alone models is witnessed which is 8.22 and 5.52 % increase compared to CT-HGR-V1 and V3, respectively. Additionally, according to Fig. 4.10, Micro Model has the least IQR and the CT-HGR stands significantly higher than the stand-alone models in terms of its accuracy among 19 subjects. As a side note on current challenges in EMG-based control of prosthetic hands, according to Reference [104], one of the future perspectives to achieve the real-time usability of prosthetic, is to improve the feature extraction component of the EMG-based solutions. Deep learning is envisioned as one fruitful approach to address the feature extraction problem, which is the focus of this study. When it comes to real-time continuous classification, beside achieving high accuracies, one requires rapid response. The proposed framework provides high accuracies over small window sizes, therefore, can generate fast and dense decision flows. In summary, we hypothesized that by introducing a compact DL-based model that has the capacity to classify a large number of hand gestures with a small amount of memory and training time, we can put a step forward towards development of more dextrous control interfaces.

As a final remark, here we focus on clarifying specific questions related to the overall design of the proposed framework. The first question that comes to the mind is how to extract the MUAP in real-time. The decomposition method utilizing STA (from extracting MUSTs to obtaining MUAPs) is performed offline, which is considered as a limitation of the method as stated in the Section 4.5. Real-time extraction of MUAPs is a fruitful direction for future research and our suggested intuition is to design a DL-based model for extraction of MUSTs in real-time. Another question is on the rational of the statement that the MUAP in the sliding window contains information on MU recruitment. MUSTs show temporal activities of each MU in the course of performing different hand

gestures. Duration of signals for each hand gesture in our dataset is about 4.5 seconds, therefore, during the entire process of performing a hand movement, different MUs with different levels of activities (forces) are involved. Consequently, extracting MUAPs based on small segments of the whole signal can provide us with more accurate information on MU recruitment at each stage of performing a specific hand gesture. Authors in References [105, 106] have also adopted a similar measure to perform STA by using sliding windows of various sizes based on their application. In [106], it is explained that since the force level changes during performing a hand gesture, sliding STA is used to obtain detailed information of the MU recruitments within small time intervals. Another key question is the rational behind integration of MUAP with raw EMG signals. Intuitively speaking, each of these signals provide different information about how a specific hand gesture was performed. HD-sEMG signals reflect the macroscopic view of the neural drive information when performing a hand gesture. These signals provide useful information about amplitude variation, signal envelope, and onset/offset times of muscle contraction which are all extracted from the signals on the skin surface. However, MUAPs represent a microscopic view of the neural drive which is very similar to the behavior of human's brain and individual motor neurons when a hand movement is being performed. This includes information about MU recruitments, MU firing rates, MU size/shape and MUAP amplitudes which are not readily provided by raw HD-sEMG signals. As the two signals are relevant to different parts of body and provide distinct views of macroscopic and microscopic neural drive information, we combined them to achieve more accurate classification accuracy for the gesture recognition task.

4.5 Conclusion

In this study, we proposed a ViT-based architecture, referred to as the CT-HGR framework, for hand gesture recognition from HD-sEMG signals. Efficacy of the proposed CT-HGR framework is validated through extensive set of experiments with various numbers of electrode channels and window sizes. Moreover, the proposed model is evaluated on instantaneous data samples of the input data, achieving, more or less, a similar accuracy to scenarios with larger window sizes. This

provides the context for real-time learning from HD-sEMG signals. Although increasing the number of learnable parameters of the CT-HGR network leads to higher accuracy, the network works reasonably well on 66 hand gestures with less than 65k number of learnable parameters. This is exceptional as its conventional DL-based counterparts have, at times, millions of parameters. Besides, a hybrid model that is trained on raw HD-sEMG signals and their decomposed MUAPs is introduced, which substantially enhances the accuracy of the single CT-HGR model trained solely on raw HD-sEMG data.

Although the utilized HD-sEMG dataset in this study is a comprehensive dataset acquired for a large number of hand gestures and from various subjects, it is obtained only from able-bodied individuals. This can be considered as a limitation of our developments. One direction for future works is to incorporate neurophysiological characteristics of hand amputees by acquiring a more generalized dataset that includes signals from this population. Moreover, the HD-sEMG decomposition phase in this study is conducted offline, preventing the proposed hybrid model to be employed in real-time HMI devices. This can be considered another limitation of our developments and a second fruitful direction for the future work to design a DL-based architecture for extracting MUSTs in real-time for development of online HMI systems. Another fruitful and important direction for future research is to focus on explainable AI to represent the extracted feature space through the proposed network and compare it with that of the conventional ML models. Finally, it would be interesting and intuitively pleasing to research potentials of Spiking Neural Networks (SNN) in this domain.

Chapter 5

Spiking Neural Networks for sEMG-based Hand Gesture Recognition

Hand gesture recognition is, nowadays, considered as a vital part of myoelectric control in Human-Machine Interaction (HMI) systems. In particular, to improve the effectiveness of HMI systems for upper-limb amputees, learning-based hand gesture recognition is deployed as a replacement for interactive devices such as keyboards or joysticks [107]. Generally speaking, hand gesture recognition has been investigated in the literature through the following two main directions: (i) The Vision-based approach in which RGB or depth cameras are used to track and recognize different hand gestures by analyzing the visual appearance of hands, and; (ii) The Sensor-based approach in which the signals related to position, orientation and movement of the hands are recorded through a set of touchless (e.g., infrared or ultrasonic) or touch-based (e.g., Electromyography (EMG) electrodes) sensors [3]. According to [4–6], the vision-based methods, compared to their sensor-based counterparts, often suffer from the following drawbacks: (a) Requiring excessive preprocessing and segmentation steps; (b) Being sensitive to the environment where the signals are being recorded, and; (c) Having higher latency and response time due to indirect estimation of the physical properties of various hand movements. In this chapter, therefore, we focus on sensor-based hand gesture recognition, in particular using surface EMG (sEMG) signals.

Owing to the recent advancements in the field of Artificial Intelligence (AI), assorted Machine

Learning/Deep Learning (ML/DL) models are proposed for the task of automatic sEMG-based hand gesture recognition [10]. These learning-based models span from conventional ML algorithms such as Support Vector Machines (SVMs) and Linear Discriminant Analysis (LDA) [11] to a wide range of simple or advanced Deep Neural Networks (DNNs) such as Convolutional [13, 14] and Recurrent [15] Neural Networks (CNNs and RNNs), Transformers [16, 17] and hybrid architectures [18]. The aforementioned models have been the focus of interest in the last few years as they have proven to work efficiently in detecting the underlying hand gesture patterns in sEMG signals. However, despite their general demonstrated effectiveness, such models suffer from major drawbacks such as not fully exploiting the temporal, spatial and neurophysiological characteristics of sEMG signals or being computationally complex and expensive. In particular, stand-alone CNN and RNN structures, typically, fail to jointly capture the time-series and spatial features of sEMG data, transformers require huge amount of training data and powerful computational resources, and hybrid architectures are difficult to optimize due to their large number of trainable parameters [21, 42, 108, 109].

In this chapter, we aim to develop an alternative and novel hand gesture recognition model based on the less-explored topic of Spiking Neural Networks (SNN), which performs spatio-temporal gesture recognition in an event-based fashion [20, 21]. An event-based processing approach, as described previously, refers to a type of data processing in which the system is susceptible to the occurrence of events rather than the static input [22]. As opposed to the classical DNN architectures, SNNs are of the capacity to imitate human brain's cognitive function by using biologically inspired models of neurons and synapses [21]. In a classical DNN, a non-linear activation function (e.g., Sigmoid or ReLU) is applied to the weighted sum of each neuron's input producing a continuous value in output. Contrarily, in SNNs, an specific activation function similar to the biological neurons is implemented, which outputs discrete-valued spikes (0 or 1) in reaction to the input [110]. Elaborating on the function of spiking neurons in SNNs, these neurons are activated at a time step if their membrane potential reaches a threshold. In this case, the neuron transmits a spike (1) to its downstream neurons and returns to its resting state potential for the next time step [22, 110]. Accordingly, SNNs provide sparse tensors in output in which a majority of entries are zero in most of the time steps. As a result, SNNs become more biologically explainable and computationally efficient requiring remarkably less amount of memory and processing units for their even-triggered

processing and low-precision computation [22, 23].

Although SNNs have been the topic of interest for many computer vision-related tasks such as image classification [24], object tracking [25] and gesture recognition [21], there is a limited number of works [26–28] on utilizing SNN for EMG-based hand gesture recognition. In Reference [24], a new technique for converting DNNs to SNNs was proposed and tested using VGG-16 [111] and ResNet [112] architectures for image classification. Authors in [25] introduced a spiking transformer-based model for event-based object tracking, which fused spatial and temporal features of the data by dynamically altering the spiking threshold of the Leaky Integrate and Fire (LIF) neurons. Vision-based hand gesture recognition was done in [21] where a spiking version of CNNs and RNNs wee combined to generate a robust framework for hand gesture recognition via Dynamic Vision Sensor (DVS) dataset. Regarding EMG-based hand gesture recognition, Reference [26] developed a Convolutional SNN (CSNN) for classification of 8 different gestures via two different sets of HD-sEMG data. In the aforementioned paper [26], common energy-density maps were obtained and fed to the CSNN model. In [27], spiking MLP and spiking CNN models were tested for a combination of both DVS and EMG sensors. Finally, Reference [28] implemented a three-layer Fully-Connected (FC) SNN paired with temporal coding and feature extraction on sEMG data for 8 different gestures.

In this chapter, a light (compact) two-layer MLP model with LIF spiking neurons is utilized to classify a set of 1 Degree of Freedom (DoF) gestures via High Density sEMG (HD-sEMG) signals. In our work, after applying Min-Max normalization on raw HD-sEMG signals, they are segmented into windowed signals of 128 samples with no overlap and fed to the spiking MLP model. This is a more straight-forward approach in comparison with using energy-density maps, spike coding and feature extraction as in [26, 28] to provide inputs for SNN architectures. We show that by considering each sample in the HD-sEMG dataset as a single time step, and inputting a batch of normalized values of HD electrode channels to the network at each time, the SNN model can differentiate between different hand gestures with maximum accuracy of around in a number of subjects. In this way, the network can work well on a quite limited amount of data with no need for data augmentation, preprocessing and spike coding. More specifically, we show that our proposed model can efficiently differentiate 14 hand movements by considering each sample of

the HD-sEMG data as a single time step for the SNN architecture. We evaluate our SNN model using a 5-fold cross-validation scheme and categorize different participants based on the range of classification accuracy we obtained for them. The following results are acquired by segmenting HD-sEMG signals into windows of size 62.5ms with no overlap. The proposed method led to 6 out of 19 subjects achieving average classification accuracy of $\geq 80\%$ with maximum accuracy of 98% associated with 3rd session of the sEMG dataset as the test set.

The remainder of the chapter is organized as follows: Section 5.1 provides an introduction to the overall structure of SNNs. In Section 5.2, our proposed SNN model for hand gesture recognition and the utilized learning method are presented. Section 5.3 describes our experiments and the classification accuracies we obtained and finally, the chapter is brought to an end with Section 5.4.

5.1 Spiking Neural Networks

As mentioned previously, SNNs are a particular type of DNNs that encompass biologically inspired spiking neurons that replicate human brain's sensory system by being sensitive to changes in events instead of updating their internal states continuously. The main building blocks of such networks are the spiking neurons that have been biologically modeled in the literature through a wide variety of methods like the LIF, Izhikevich and Hodgkin–Huxley (HH) [113]. These models try to provide a biological explanation of how spiking neurons generate spikes in each time step. In this chapter, we focus on the LIF neuron model which is the most computationally efficient and commonly used model in this context [114].

The LIF neuron takes the weighted sum of its inputs in each time step and integrates it with the input from other time steps in a leaky manner [22]. This behavior is similar to what happens in a low-pass filter in a circuit with one Resistor (R) and a Capacitor (C). In case the membrane potential of the LIF neuron which is the leaky integration of the potential over the current and previous time steps surpasses a threshold θ, a single spike (discrete event) is emitted in that time step and the potential returns to zero or any formerly-defined resting state potential. As a result, the output spikes represent the timing and frequency characteristics of the input spikes to the LIF neuron. Fig. 5.1 presents a single LIF neuron that, at each time step, integrates weighted sum of 3

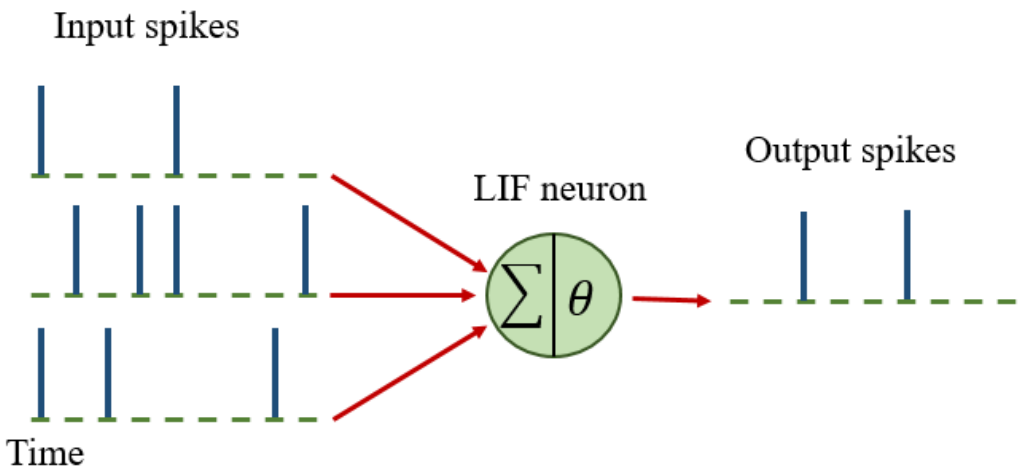

Figure 5.1: Simulation of LIF neuron's response to input spikes.

input features, compares it to the threshold and emits a single spike if enough stimulated. Fig. 5.2 shows LIF neuron's input and membrane potential in different time steps. As can be seen, depending on the length and frequency of the input, membrane potential changes with leakage and resets to the resting potential when the threshold is reached. In this work, a first-order LIF neuron is utilized in each layer for which the membrane potential at time step t is calculated as follows

$$V(t) = \beta V(t-1) + I_{input}(t) - \theta S(t-1), \tag{13}$$

where V is the membrane potential, I_{input} is the neuron's input value (current in the context of RC circuits), β is the decay rate of LIF neuron and θ is the membrane's threshold. Here, S represents the activation status of the neuron in the previous time step and takes binary values of 0 if the neuron is not activated, or 1 if the neuron was activated. A detailed explanation of the differential equations for the membrane potential derived from the low-pass RC circuit can be found in [22].

5.2 The Proposed SNN Architecture

In this section, we develop the proposed SNN architecture for sEMG-based gesture recognition. In particular, we highlight the way sEMG signals are prepared to be fed to the proposed SNN architecture, and how it classifies distinct hand gestures differently from a classical DNN architecture.

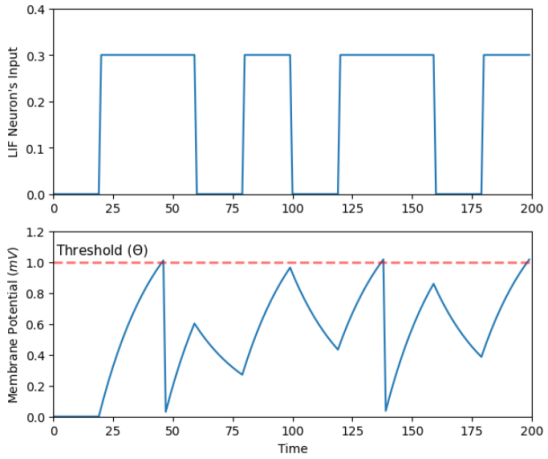

Figure 5.2: LIF neuron's input signal (Up) and membrane potential (Bottom) in different time steps. In this particular case, we have 2 output spikes as the membrane potential surpasses the threshold twice.

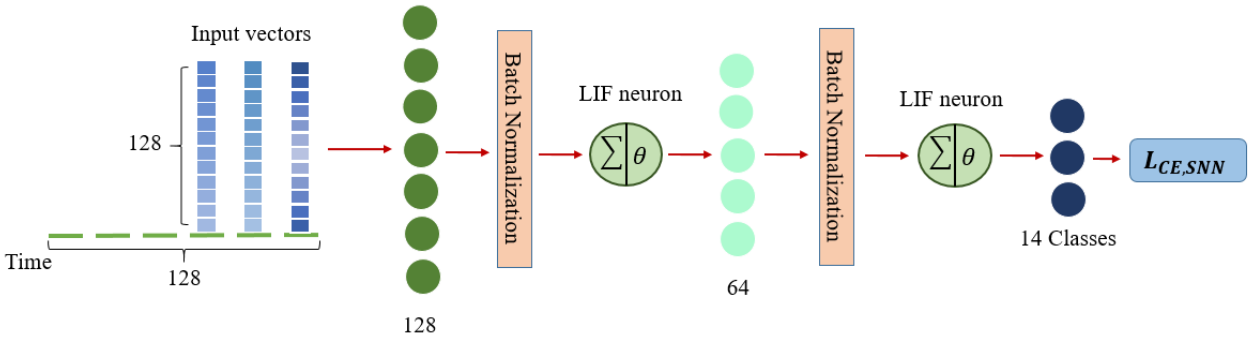

Figure 5.3: Representation of the proposed SNN architecture with two FC layers. There is a total of 128 time steps for each of which a vector of 128 features is fed to the network and the $\mathcal{L}_{CE,SNN}(t)$ is calculated. Total loss is the summation of loss across all time steps.

The proposed SNN architecture is a light/compact spiking MLP consisting of 2 linear layers followed by spiking activation functions. Batch normalization is used after the linear layers to prevent any changes in distribution of inputs to spiking activation functions and to improve the generalization capacity of the model. To construct the proposed SNN model using only a small amount of input data, HD-sEMG signals (described in Chapter 2.3) were divided into windows of 128 samples (62.5 ms) with no overlap. In this way, the model is trained and tested with around $4,100$ and $1,000$ data samples, respectively, which significantly reduces the required time and computational power for both the training and testing stages.

In the context of vision-based object/gesture recognition approaches using SNNs, event-based

video streams of shape $(time\_resolution, height, width)$ are collected, sampled at an arbitrary sampling frequency and windowed to form the model's input data with shape $(window\_size, height, width)$ [21]. Each of such windowed frames, therefore, are assumed to happen in one time step and are given to the network in order of occurrence. For instance, if the time resolution of the video stream is 2 sec. and the sampling time and the window size are set to 1 and 20 ms, 100 input frames of size $(20, height, width)$ are generated and fed to the SNN that has 100 time steps. However, in order to increase the time resolution and decrease the memory usage of our proposed network, we considered the sampling frequency as the HD-sEMG dataset's original frequency, $2,048$ Hz, which yields a time resolution of $\frac{1}{2048}$ sec. Then, the input electrode channels were flattened to be of 1 dimension and windowed signals of shape $(window\_size, No\_of\_channels)$ ($(128, 128)$ in our study) were generated. Different from other computer vision methods using event-based data [20, 21, 115], in this study, each sample of the input window was assumed to be a single time step in the proposed SNN, meaning that in each time step, a 1-dimensional vector of 128 features pertinent to the signals of electrode channels is fed to the network and converted to a 1-dimensional vector of 14 classes in output.

In multi-class classification tasks using typical DNNs, the Categorical Cross-Entropy (CCE) loss function, in which the output neuron with the maximum activation value is accepted as the predicted class, is utilized. For pure SNNs, however, several loss functions have been suggested within the relevant literature that adopt a different measure for deciphering the behavior of output spikes that leads to giving more importance to the correct class among others [21, 22, 116]. In this work, we used one of the most common SNN loss functions called the Cross-Entropy (CE) rate loss, which is a combination of CE with spike count rate. Through this method, the output spikes at each time step are passed to a CE function and a single loss is calculated for the time step. Then, the losses for all the time steps are accumulated and introduced as the final loss, which favors the neuron with highest number of spikes as the predicted class. Thus, instead of calculating cross-entropy for neuron continuous-value activations, it is applied to discrete spikes by first using the Softmax function as

$$p_i(t) = \frac{e^{s_i(t)}}{\sum_{k=1}^{C} e^{s_k(t)}}, \tag{14}$$

for ($1 \leq i \leq C$), where $\mathrm{p}_i(t)$ is the probability of neuron i representing the correct class at time step t, s_i is the activation status of neuron i, which can take values of 0 or 1, and C is the total number of classes. Afterwards, the final CE loss function is calculated as

$$\mathcal{L}_{CE,SNN} = \sum_{t=1}^{T} \mathcal{L}_{CE,snn}(t), \tag{15}$$

in which $\mathcal{L}_{CE,snn}(t)$ is defined by

$$\mathcal{L}_{CE,SNN}(t) = -\sum_{i=1}^{C} y_i \log\left(p_i(t)\right), \tag{16}$$

where y_i is the one-hot target vector and T is the total number of time steps.

Fig. 5.3 shows the overall structure of the proposed SNN network from accepting the input in distinct time steps to computing the CE rate loss function. As presented in the figure, input vectors of 128 features go through a network of two consecutive FC layers, between which Batch Normalization and LIF neurons are positioned. At the end, CE rate loss is calculated for 14 output neurons and target vectors.

5.3 Experiments and Results

In this section, we evaluate the performance of our proposed SNN architecture by comparing the classification accuracy over different subjects and different sessions via a 5-fold cross-validation technique. A common way to measure accuracy in SNNs is to employ a metric referred to as spike count accuracy in which the neuron with the highest number of spikes in all time steps is chosen as the predicted class and compared to the target class.

Our proposed SNN model is developed using the snnTorch framework, which is specifically designed for implementing gradient-based DL architectures with SNNs [22]. Each LIF neuron in our SNN model requires a differentiable gradient function for backpropagation. As spikes (often represented by the Heaviside function) are not intrinsically differentiable, we used the surrogate fast sigmoid gradient function, which acts as a Heaviside in the forward pass and replaces the gradient of fast sigmoid in the backward pass to make everything differentiable [117]. The hyperparameters of

Table 5.1: Hyperparameters of the proposed SNN framework

Hyperparameter	**Value**
Batch size	128
Optimizer	Adam
Learning rate	0.0005-0.00025
No of epochs	20
LIF threshold (θ)	1 (trainable)
LIF decay rate (β)	0.9 (trainable)

the SNN model can be found in Table 5.1. To prevent the model from overfitting, $L2$ regularization and learning rate annealing is used.

Fig. 5.4 demonstrates the boxplots and Interquartile Range (IQR) for accuracy and standard deviation (STD) of each fold for all the 19 subjects. According to Fig. 5.4 and the Wilcoxon signed-rank test's annotations, the discrepancy of the accuracy results for folds $2-5$ is not statistically significant. Nevertheless, since the distribution of the first session's HD-sEMG signals was markedly different from the average distribution of folds $2-5$, average accuracy when session 1 was held as the test set is substantially lower than the other 4 conditions. This can be observed from the Wilcoxon test's results (p-value ≤ 0.05) on the accuracy obtained for fold 1 with that of the other 4 folds. Fig. 5.5 shows a specific case of Fig. 5.4 when sessions 1 and 3 were considered as the test set. As can be seen, the accuracy varies significantly among subjects, ranging from 29% to 93% in Session 1 and from 47% to 98% in session 3.

The LIF neuron's threshold (θ) and decay rate (β) were set to 1 and 0.9, respectively, for the whole experiments. We hypothesized that although these parameters were expected to update continually in each epoch, they did not change significantly during the training stage with the small 0.0005 learning rate used. Therefore, fixing these parameters for all the participants results in the network not accurately simulating some participant's brain activity when performing different hand movements. More specifically, inappropriate threshold and decay rate in SNNs can cause inaccurate generation of spikes, which has a direct effect on the classification accuracy. Having taken this fact into account and according to Table 5.2, Subjects 1,10, 11, 13, 16, and 19 had the best performance among others with more than 80% average accuracy over 5 folds, implying that the θ and β parameters were chosen properly for their brain function. Fig. 5.6 shows raster plots of two different samples of the test set for Subject 16 and fold 3. Each of these plots indicate output spikes of 14

Table 5.2: Categorization of 19 subjects based on 5 accuracy ranges.

Accuracy range (%)	Subjects
80-100	1,10,11,13,16,19
70-80	2,8,12,18
60-70	4,5,6,7,9,17
50-60	14
40-50	3,15

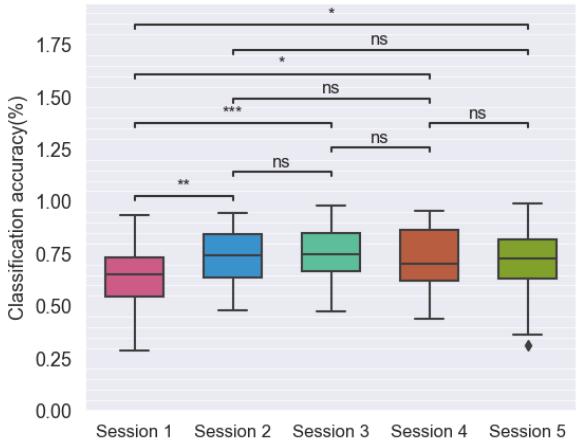

Figure 5.4: Boxplots and IQR of classification accuracy for 5 folds (sessions) of the dataset over 19 participants.

neurons in all time steps. The actual hand gesture in these samples were the 11^{th} and 3^{rd} (shown in green), respectively, which had the largest spike counts through all time steps and were predicted rightly by the proposed SNN model. It is worth noting that comparison with other sEMG-based SNN models for hand gesture recognition [26–28] was not feasible as they used different dataset and pre-processing methods for feeding data into the SNN model.

5.4 Summary and Conclusion

In this chapter, we presented a compact (light) SNN model for hand gesture recognition from HD-sEMG data. Compared to classical DNN architectures, SNNs can more accurately mimic neurophysiological characteristics of the human brain by utilizing specific neurons that interact with each other via emitting discrete-value (0 or 1) signals instead of producing continuous values all the

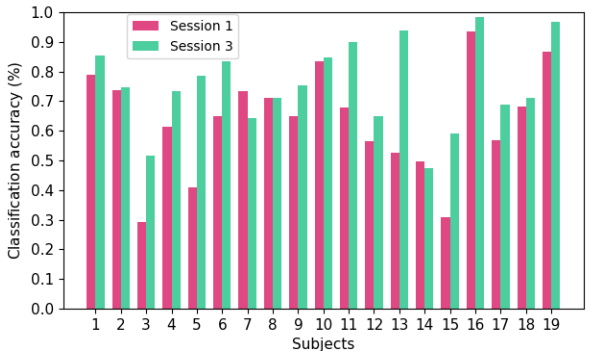

Figure 5.5: Comparison of classification accuracy of folds (sessions) 1 and 3, representing the worst and best folds, for all the 19 subjects.

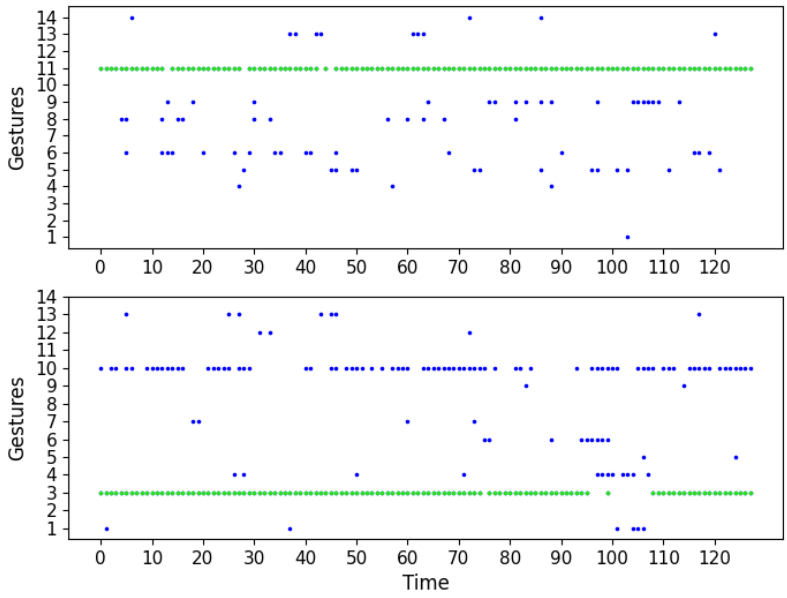

Figure 5.6: Raster plots of output spiking neurons for subject 16 when session 3 is considered as the test set. The classes with the highest spike counts are shown in green which were predicted correctly by the SNN model.

time. Using 5-fold cross-validation, we evaluated the performance of our proposed SNN model on each subject separately. We showed that depending on data collection session used as the test set, classification accuracy could vary significantly among different participants and the model could yield a huge range of classification accuracies, from 29% to 98% for 14 hand gestures. Wilcoxon signed-rank statistical analysis demonstrated that the discrepancy between the accuracies acquired

from folds $2-5$ was not significant. But, fold 1's performance was significantly different from other folds' on account of its distinctive distribution of sEMG data. A prospect for future work could be to improve performance and generalizability of the SNN framework on all subjects by employing transfer learning approaches.

Chapter 6

Summary and Future Research Directions

This chapter concludes the thesis with a list of main contributions made in this dissertation and some proposed directions for future works.

6.1 Summary of Thesis Contributions

The research works presented in this thesis are motivated by recent advances in the design and implementation of AI and DL-based models for signal processing, aiming to develop efficient Human Machine Interface (HMI) systems with a wide variety of applications in prosthetics, neurorobotics and mixed AR/VR settings. Considering recent progress in development of innovative DL-based architectures, particularly Transformers, Vision Transformers (ViTs), and Spiking Neural Networks (SNNs), this thesis aimed to tackle the limitations and drawbacks of the existing ML/DL-based models, focusing on two different approaches, i.e., gesture recognition from macroscopic and microscopic neural drive information. In this regard, the thesis made a number of contributions, as briefly outlined below:

(1) **The ViT-HGR Framework:** A ViT-based framework is proposed for hand gesture recognition from HD-sEMG signals. Thanks to the input parallelization and attention mechanism it incorporates, many of the problems associated with the other DL/ML-based models [9–12, 14, 45–49] proposed in the literature have been solved. These problems generally include large training times, huge memory usage and limited generalizability due to depending on handcrafted features [57]. The proposed ViT-HGR framework is able to address the failure of CNN [14] and RNN-based [58] frameworks in only attending to either the spatial or the temporal information in HD-sEMG signals. It can also be a suitable alternative to hybrid CNN-RNN [18, 59] structures due to their complexity and inevitably sequential nature necessitating the network to process data in order. By eliminating the complexity of simultaneously exploiting CNNs/RNNs or merging them with transformers, we aim to construct a compact and stand-alone framework with reduced computational overhead. Also, owing to a specific signal processing approach we utilize before feeding raw HD-sEMg signals to the model, we observe that the ViT-HGR framework achieves high accuracy when trained from scratch with no data augmentation. The efficiency of the proposed ViT-HGR framework is evaluated using a recently-released HD-sEMG dataset consisting of 65 isometric hand gestures. Our experiments with 64-sample (31.25 ms) window size yield average test accuracy of 84.62 ± 3.07%, where only 78,210 learnable parameters are utilized in the model. The compact structure of the proposed ViT-based ViT-HGR framework (i.e., having significantly reduced number of trainable parameters) shows great potentials for its practical application for prosthetic control.

(2) **The CT-HGR Framework:** In this section, on the one hand, an extension of the ViT-HGR framework is tailored and evaluated using different settings of the input HD-sEMG data. A comprehensive evaluation of the proposed CT-HGR architecture is carried out with variable window sizes, number of electrode channels, and complexity of the network. Additionally, the train/test times, memory consumption, and classification accuracy was reported and compared for different settings. We also indicate that the CT-HGR framework is able to work with instantaneous data samples which are single frames of HD-sEMG signals in one

time point. This suggests that there are reproducible patterns among instantaneous samples of a specific hand gesture which could also be a physiological representation of muscle activities. Therefore, the network can achieve acceptable accuracy when receiving, as an input, a single frame of the HD-sEMG image. Furthermore, a detailed comparison of the proposedCT-HGR model with two ML algorithms (i.e. LDA and SVM) and a 3D CNN model is drawn. Two various sets of handcrafted features are computed for each ML method [10, 85, 86, 88] and all of the 6 models are compared in terms of their accuracy, precision, recall, F1-score, train/test times, and memory usage. A Wilcoxon's signed-rank test is also applied on the accuracies of models over different subjects to observe discrepancies between the proposed framework and the other models and to show the effect of changing feature sets in ML algorithms on their performance. The proposed CT-HGR framework is applied to 31.25, 62.5, 125, 250 ms window sizes of the HD-sEMG dataset utilizing 32, 64, and 128 electrode channels. Our results are obtained via 5-fold cross-validation by first applying the proposed framework on the dataset of each subject separately and then, averaging the accuracies among all the subjects. The average accuracy over all the participants using 32 electrodes and a window size of 31.25 ms is 86.23%, which gradually increases till reaching 91.98% for 128 electrodes and a window size of 250 ms. The CT-HGR achieves accuracy of 89.13% for instantaneous recognition. On the other hand, a hybrid ViT-based model is introduced that combines HD-sEMG signals (macroscopic neural drive information) with MUAPs (microscopic neural drive information) to perform more accurate prediction of the entire 66 gestures in our utilized dataset. HD-sEMG signals are modelled as a spatio-temporal convolution of MUSTs, which provide an exact physiological description of how each hand movement is encoded at neurospinal level [36]. Thus, this method is proposed to advance other promising works in the literature [35, 97, 98] only exploiting MUSTs for gesture recognition and to show that the combination of HD-sEMG signals with MUSTs achieves higher accuracy than using either of these signals distinctly. The fused CT-HGR model includes two stand-alone CT-HGR models (called CT-HGR-V1 and CT-HGR-V3) that accept either raw HD-sEMG signals or the peak-to-peak values of MUAPs which are then concatenated using two FC layers. According to

our experiments, a great improvement in average performance of the fused model in comparison to both stand-alone models is witnessed which is 8.22 and 5.52 % increase compared to CT-HGR-V1 and V3, respectively.

(3) **The SNN-based Framework:** Here, our goal is to create a unique and innovative hand gesture recognition model by utilizing the less-explored concept of Spiking Neural Networks (SNN). This model focuses on recognizing gestures in a spatio-temporal manner using event-based processing. There are a limited number of works on SNN-based gesture recognition models using sEMG signals. Our proposed network is suggested to reduce the complexity of other similar works [26–28] coupling SNNs with CNNs, energy-density maps and temporal coding to perform EMG-based gesture recognition. We design a compact MLP model with LIF spiking neurons to classify a set of 1 Degree of Freedom (DoF) gestures via HD-sEMG signals. In our study, following the application of Min-Max normalization to HD-sEMG signals, we divide them into windowed signals containing 128 samples each, without any overlap. These segmented signals are then inputted to the spiking MLP model. We indicate that considering each sample in the HD-sEMG dataset as a single time step results in the network performing well on a quite limited amount of data with no need for data augmentation, preprocessing and spike coding. Particularly, we show that our proposed model can efficiently differentiate 14 hand movements by considering each sample of the HD-sEMG data as a single time step for the SNN architecture. We evaluate our SNN model using a 5-fold cross-validation scheme and categorize different participants based on the range of classification accuracy we obtained for them. The following results are acquired by segmenting HD-sEMG signals into windows of size 62.5ms with no overlap. The proposed method led to 6 out of 19 subjects achieving average classification accuracy of $\geq 80\%$ with maximum accuracy of 98% associated with 3^{rd} session of the sEMG dataset as the test set.

6.2 Future Research

(1) Although the utilized HD-sEMG dataset in this study is a comprehensive dataset acquired for a large number of hand gestures and from various subjects, it is obtained only from able-bodied individuals. This can be considered as a limitation of our developments. One direction for future works is to incorporate neurophysiological characteristics of hand amputees by acquiring a more generalized dataset that includes signals from this population.

(2) The HD-sEMG decomposition phase in this study (Chapter 4.3.5) is conducted offline, preventing the proposed hybrid model to be employed in real-time HMI devices. This can be considered another limitation of our developments and a second fruitful direction for the future work to design a DL-based architecture for extracting MUSTs in real-time for development of online HMI systems.

(3) Another fruitful and important direction for future research is to focus on explainable AI to represent the extracted feature space through the proposed network in chapter 4 and compare it with that of the conventional ML models.

(4) A prospect for future work on SNNs could be to improve performance and generalizability of the SNN framework on all subjects by employing transfer learning approaches.

(5) Also, our compact SNN model can be integrated with CNNs and RNNs to better leverage the spatial and temporal information found in HD-sEMG signals. This can be simultaneously done with utilizing different kinds of spiking neurons rather than LIF to assess their ability to imitate human brain's physiological nature.

Bibliography

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

[2] Nebojša Malešević, Alexander Olsson, Paulina Sager, Elin Andersson, Christian Cipriani, Marco Controzzi, Anders Björkman, and Christian Antfolk, "A database of high-density surface electromyogram signals comprising 65 isometric hand gestures," *Scientific Data*, vol. 8, no. 1, pp. 63, 2021.

[3] Chung-Ju Liao, Shun-Feng Su, and Ming-Chang Chen, "Vision-based hand gesture recognition system for a dynamic and complicated environment," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, 2015, pp. 2891–2895.

[4] Davinder Kumar and Aman Ganesh, "A critical review on hand gesture recognition using semg: Challenges, application, process and techniques," in *Journal of Physics: Conference Series*. IOP Publishing, 2022, vol. 2327, p. 012075.

[5] Weiya Chen, Chenchen Yu, Chenyu Tu, Zehua Lyu, Jing Tang, Shiqi Ou, Yan Fu, and Zhidong Xue, "A survey on hand pose estimation with wearable sensors and computer-vision-based methods," *Sensors*, vol. 20, no. 4, pp. 1074, 2020.

[6] Debajit Sarma and Manas Kamal Bhuyan, "Methods, databases and recent advancement of vision-based hand gesture recognition for hci systems: A review," *SN Computer Science*, vol. 2, no. 6, pp. 436, 2021.

[7] Cinthya Lourdes Toledo-Peral, Josefina Gutiérrez-Martínez, Jorge Airy Mercado-Gutiérrez, Ana Isabel Martín-Vignon-Whaley, Arturo Vera-Hernández, and Lorenzo Leija-Salas, "semg signal acquisition strategy towards hand fes control," *Journal of Healthcare Engineering*, vol. 2018, 2018.

[8] Ning Jiang, Strahinja Dosen, Klaus-Robert Muller, and Dario Farina, "Myoelectric control of artificial limbs—is there a need to change focus?[in the spotlight]," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 152–150, 2012.

[9] Wenjun Chen and Zhen Zhang, "Hand gesture recognition using semg signals based on support vector machine," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. IEEE, 2019, pp. 230–234.

[10] Kyung Hyun Lee, Ji Young Min, and Sangwon Byun, "Electromyogram-based classification of hand and finger gestures using artificial neural networks," *Sensors*, vol. 22, no. 1, pp. 225, 2021.

[11] Francesca Leone, Cosimo Gentile, Anna Lisa Ciancio, Emanuele Gruppioni, Angelo Davalli, Rinaldo Sacchetti, Eugenio Guglielmelli, and Loredana Zollo, "Simultaneous semg classification of hand/wrist gestures and forces," *Frontiers in Neurorobotics*, vol. 13, pp. 42, 2019.

[12] Ruixuan Zhang, Xushu Zhang, Dongdong He, Ruixue Wang, and Yuan Guo, "semg signals characterization and identification of hand movements by machine learning considering sex differences," *Applied Sciences*, vol. 12, no. 6, pp. 2962, 2022.

[13] Shouan Song, Lei Yang, Man Wu, Yanhong Liu, and Hongnian Yu, "Dynamic hand gesture recognition via electromyographic signal based on convolutional neural network," in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2021, pp. 876–881.

[14] Xiang Chen, Yu Li, Ruochen Hu, Xu Zhang, and Xun Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp. 1292–1304, 2020.

[15] Alejandro Toro-Ossaba, Juan Jaramillo-Tigreros, Juan C Tejada, Alejandro Peña, Alexandro López-González, and Rui Alexandre Castanho, "Lstm recurrent neural network for hand gesture recognition using emg signals," *Applied Sciences*, vol. 12, no. 19, pp. 9700, 2022.

[16] Shu Shen, Xuebin Wang, Fan Mao, Lijuan Sun, and Minghui Gu, "Movements classification through semg with convolutional vision transformer and stacking ensemble learning," *IEEE Sensors Journal*, vol. 22, no. 13, pp. 13318–13325, 2022.

[17] Mansooreh Montazerin, Elahe Rahimian, Farnoosh Naderkhani, S Farokh Atashzar, Hamid Alinejad-Rokny, and Arash Mohammadi, "Hydra-hgr: A hybrid transformer-based architecture for fusion of macroscopic and microscopic neural drive information," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[18] Yu Hu, Yongkang Wong, Wentao Wei, Yu Du, Mohan Kankanhalli, and Weidong Geng, "A novel attention-based hybrid cnn-rnn architecture for semg-based gesture recognition," *PloS one*, vol. 13, no. 10, pp. e0206049, 2018.

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[20] Zihao Zhao, Yanhong Wang, Qiaosha Zou, Tie Xu, Fangbo Tao, Jiansong Zhang, Xiaoan Wang, C-J Richard Shi, Junwen Luo, and Yuan Xie, "The spike gating flow: A hierarchical structure based spiking neural network for online gesture recognition," *arXiv preprint arXiv:2206.01910*, 2022.

[21] Yannan Xing, Gaetano Di Caterina, and John Soraghan, "A new spiking convolutional recurrent neural network (scrnn) with applications to event-based hand gesture recognition," *Frontiers in neuroscience*, vol. 14, pp. 590164, 2020.

[22] Jason K Eshraghian, Max Ward, Emre Neftci, Xinxin Wang, Gregor Lenz, Girish Dwivedi, Mohammed Bennamoun, Doo Seok Jeong, and Wei D Lu, "Training spiking neural networks using lessons from deep learning," *arXiv preprint arXiv:2109.12894*, 2021.

[23] Yongbao Xie, Peifu Wang, Wenyuan Chen, Wenxue Wang, and Lianqing Liu, "A neural-based approach to hand gesture recognition with hd-semg," in *2022 41st Chinese Control Conference (CCC)*. IEEE, 2022, pp. 3144–3149.

[24] Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy, "Going deeper in spiking neural networks: Vgg and residual architectures," *Frontiers in neuroscience*, vol. 13, pp. 95, 2019.

[25] Jiqing Zhang, Bo Dong, Haiwei Zhang, Jianchuan Ding, Felix Heide, Baocai Yin, and Xin Yang, "Spiking transformers for event-based single object tracking," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2022, pp. 8801–8810.

[26] Weijie Ke, Yannan Xing, Gaetano Di Caterina, Lykourgos Petropoulakis, and John Soraghan, "Deep convolutional spiking neural network based hand gesture recognition," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–7.

[27] Enea Ceolini, Charlotte Frenkel, Sumit Bam Shrestha, Gemma Taverni, Lyes Khacef, Melika Payvand, and Elisa Donati, "Hand-gesture recognition based on emg and event-based camera sensor fusion: A benchmark in neuromorphic computing," *Frontiers in Neuroscience*, p. 637, 2020.

[28] Yang Liuy and Long Chengy, "Spiking-neural-network based fugl-meyer hand gesture recognition for wearable hand rehabilitation robot," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–6.

[29] Francesco Negro, Silvia Muceli, Anna Margherita Castronovo, Ales Holobar, and Dario Farina, "Multi-channel intramuscular and surface emg decomposition by convolutive blind source separation," *Journal of neural engineering*, vol. 13, no. 2, pp. 026027, 2016.

[30] Gilles Chabriel, Martin Kleinsteuber, Eric Moreau, Hao Shen, Petr Tichavsky, and Arie Yeredor, "Joint matrices decompositions and blind source separation: A survey of methods, identification, and applications," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 34–43, 2014.

[31] Daniela Souza de Oliveira, Andrea Casolo, Thomas G Balshaw, Sumiaki Maeo, Marcel Bahia Lanza, Neil RW Martin, Nicola Maffulli, Thomas Mehari Kinfe, Bjoern M Eskofier, Jonathan P Folland, et al., "Neural decoding from surface high-density emg signals: influence of anatomy and synchronization on the number of identified motor units," *Journal of Neural Engineering*, vol. 19, no. 4, pp. 046029, 2022.

[32] Alexander Kenneth Clarke, Seyed Farokh Atashzar, Alessandro Del Vecchio, Deren Barsakcioglu, Silvia Muceli, Paul Bentley, Filip Urh, Ales Holobar, and Dario Farina, "Deep learning for robust decomposition of high-density surface emg signals," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 2, pp. 526–534, 2020.

[33] Carlo J De Luca, Alexander Adam, Robert Wotiz, L Donald Gilmore, and S Hamid Nawab, "Decomposition of surface emg signals," *Journal of neurophysiology*, vol. 96, no. 3, pp. 1646–1657, 2006.

[34] Dan Stashuk, "Emg signal decomposition: how can it be accomplished and used?," *Journal of Electromyography and Kinesiology*, vol. 11, no. 3, pp. 151–173, 2001.

[35] Yongle Zhao, Xu Zhang, Xinhui Li, Haowen Zhao, Xiang Chen, Xun Chen, and Xiaoping Gao, "Decoding finger movement patterns from microscopic neural drive information based on deep learning," *Medical Engineering & Physics*, vol. 104, pp. 103797, 2022.

[36] Dario Farina, Ivan Vujaklija, Massimo Sartori, Tamás Kapelner, Francesco Negro, Ning Jiang, Konstantin Bergmeister, Arash Andalib, Jose Principe, and Oskar C Aszmann, "Man/machine interface based on the discharge timings of spinal motor neurons after targeted muscle reinnervation," *Nature biomedical engineering*, vol. 1, no. 2, pp. 1–12, 2017.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[38] Mansooreh Montazerin, Elahe Rahimian, Farnoosh Naderkhani, S Farokh Atashzar, Svetlana Yanushkevich, and Arash Mohammadi, "Transformer-based hand gesture recognition from

instantaneous to fused neural decomposition of high-density emg signals," *Scientific Reports*, vol. 13, no. 1, pp. 11000, 2023.

[39] Mansooreh Montazerin, Farnoosh Naderkhani, and Arash Mohammadi, "Spiking neural networks for semg-based hand gesture recognition," in *IEEE Conference on Systems, Man, and Cybernetics*, 2023.

[40] Wei Li, Ping Shi, and Hongliu Yu, "Gesture recognition using surface electromyography and deep learning for prostheses hand: State-of-the-art, challenges, and future," *Frontiers in neuroscience*, p. 259, 2021.

[41] Elahe Rahimian, Soheil Zabihi, Amir Asif, Dario Farina, Seyed Farokh Atashzar, and Arash Mohammadi, "Fs-hgr: Few-shot learning for hand gesture recognition via electromyography," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 29, pp. 1004–1015, 2021.

[42] Elahe Rahimian, Soheil Zabihi, Amir Asif, Dario Farina, S Farokh Atashzar, and Arash Mohammadi, "Hand gesture recognition using temporal convolutions and attention mechanism," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 1196–1200.

[43] Dario Farina, Arash Mohammadi, Tulay Adali, Nitish V Thakor, and Konstantinos N Plataniotis, "Signal processing for neurorehabilitation and assistive technologies," *IEEE Signal Processing Magazine*, vol. 38, no. 4, pp. 5–7, 2021.

[44] Simon Tam, Mounir Boukadoum, Alexandre Campeau-Lecours, and Benoit Gosselin, "Intuitive real-time control strategy for high-density myoelectric hand prosthesis using deep and transfer learning," *Scientific Reports*, vol. 11, no. 1, pp. 11275, 2021.

[45] G Emayavaramban, S Divyapriya, VM Mansoor, A Amudha, M Siva Ramkumar, P Nagaveni, and M SivaramKrishnan, "Semg based classification of hand gestures using artificial neural network," *Materials Today: Proceedings*, vol. 37, pp. 2591–2598, 2021.

[46] Elahe Rahimian, Soheil Zabihi, S Farokh Atashzar, Amir Asif, and Arash Mohammadi, "Semg-based hand gesture recognition via dilated convolutional neural networks," in *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2019, pp. 1–5.

[47] Reza Bagherian Azhiri, Mohammad Esmaeili, and Mehrdad Nourani, "Real-time emg signal classification via recurrent neural networks," in *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2021, pp. 2628–2635.

[48] Miguel Simão, Pedro Neto, and Olivier Gibaru, "Emg-based online classification of gestures with recurrent neural networks," *Pattern Recognition Letters*, vol. 128, pp. 45–51, 2019.

[49] Elahe Rahimian, Soheil Zabihi, Amir Asif, Dario Farina, S Farokh Atashzar, and Arash Mohammadi, "Temgnet: Deep transformer-based decoding of upperlimb semg for hand gestures recognition," *arXiv preprint arXiv:2109.12379*, 2021.

[50] Usha Kuruganti, Ashirbad Pradhan, and Jacqueline Toner, "High-density electromyography provides improved understanding of muscle function for those with amputation," *Frontiers in Medical Technology*, p. 41, 2021.

[51] István Ketykó, Ferenc Kovács, and Krisztián Zsolt Varga, "Domain adaptation for semg-based gesture recognition with recurrent neural networks," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–7.

[52] Monica Rojas-Martínez, Miguel A Mañanas, and Joan F Alonso, "High-density surface emg maps from upper-arm and forearm muscles," *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–17, 2012.

[53] Dianchun Bai, Shutian Chen, and Junyou Yang, "Upper arm motion high-density semg recognition optimization based on spatial and time-frequency domain features," *Journal of Healthcare Engineering*, vol. 2019, 2019.

[54] Jiangcheng Chen, Sheng Bi, George Zhang, and Guangzhong Cao, "High-density surface emg-based gesture recognition using a 3d convolutional neural network," *Sensors*, vol. 20, no. 4, pp. 1201, 2020.

[55] Mónica Rojas-Martínez, Leidy Yanet Serna, Mislav Jordanic, Hamid Reza Marateb, Roberto Merletti, and Miguel Ángel Mañanas, "High-density surface electromyography signals during isometric contractions of elbow muscles of healthy humans," *Scientific data*, vol. 7, no. 1, pp. 1–12, 2020.

[56] Isabella Campanini, Catherine Disselhorst-Klug, William Z Rymer, and Roberto Merletti, "Surface emg in clinical assessment and neurorehabilitation: barriers limiting its use," *Frontiers in Neurology*, p. 934, 2020.

[57] Kun Yang, Manjin Xu, Xiaotong Yang, Runhuai Yang, and Yueming Chen, "A novel emg-based hand gesture recognition framework based on multivariate variational mode decomposition," *Sensors*, vol. 21, no. 21, pp. 7002, 2021.

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

[59] Pufan Xu, Fei Li, and Haipeng Wang, "A novel concatenate feature fusion rcnn architecture for semg-based hand gesture recognition," *PloS one*, vol. 17, no. 1, pp. e0262810, 2022.

[60] Deren Y Barsakcioglu and Dario Farina, "A real-time surface emg decomposition system for non-invasive human-machine interfaces," in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2018, pp. 1–4.

[61] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[62] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[63] João Lopes, Miguel Simão, Nuno Mendes, Mohammad Safeea, José Afonso, and Pedro Neto, "Hand/arm gesture segmentation by motion using imu and emg sensing," *Procedia Manufacturing*, vol. 11, pp. 107–113, 2017.

[64] Yan Zhang, Fan Yang, Qi Fan, Anjie Yang, and Xuan Li, "Research on semg-based gesture recognition by dual-view deep learning," *IEEE Access*, vol. 10, pp. 32928–32937, 2022.

[65] Manfredo Atzori, Matteo Cognolato, and Henning Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers in neurorobotics*, vol. 10, pp. 9, 2016.

[66] Wentao Wei, Yongkang Wong, Yu Du, Yu Hu, Mohan Kankanhalli, and Weidong Geng, "A multi-stream convolutional neural network for semg-based gesture recognition in muscle-computer interface," *Pattern Recognition Letters*, vol. 119, pp. 131–138, 2019.

[67] Manfredo Atzori, Arjan Gijsberts, Claudio Castellini, Barbara Caputo, Anne-Gabrielle Mittaz Hager, Simone Elsig, Giorgio Giatsidis, Franco Bassetto, and Henning Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Scientific data*, vol. 1, no. 1, pp. 1–13, 2014.

[68] Risto Koiva, Barbara Hilsenbeck, and Claudio Castellini, "Evaluating subsampling strategies for semg-based prediction of voluntary muscle contractions," in *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2013, pp. 1–7.

[69] Zhen Zhang, Kuo Yang, Jinwu Qian, and Lunwei Zhang, "Real-time surface emg pattern recognition for hand gestures based on an artificial neural network," *Sensors*, vol. 19, no. 14, pp. 3170, 2019.

[70] Elahe Rahimian, Soheil Zabihi, Seyed Farokh Atashzar, Amir Asif, and Arash Mohammadi, "Xceptiontime: independent time-window xceptiontime architecture for hand gesture classification," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 1304–1308.

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

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

[73] Tianyun Sun, Jacqueline Libby, JohnRoss Rizzo, and S Farokh Atashzar, "Deep augmentation for electrode shift compensation in transient high-density semg: Towards application in neurorobotics," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 6148–6153.

[74] Ahmed A Al Taee, Rami N Khushaba, Tanveer Zia, and Adel Al-Jumaily, "The effectiveness of narrowing the window size for ld & hd emg channels based on novel deep learning wavelet scattering transform feature extraction approach," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2022, pp. 3698–3701.

[75] Bernard Hudgins, Philip Parker, and Robert N Scott, "A new strategy for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 40, no. 1, pp. 82–94, 1993.

[76] Jörn Vogel and Annette Hagengruber, "An semg-based interface to give people with severe muscular atrophy control over assistive devices," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 2136–2141.

[77] Dario Farina, Arash Mohammadi, Tulay Adali, Nitish V Thakor, and Konstantinos N Plataniotis, "Signal processing for neurorehabilitation and assistive technologies," *IEEE Signal Processing Magazine*, vol. 38, no. 4, pp. 5–7, 2021.

[78] Philipp Koch, Mark Dreier, Marco Maass, Martina Böhme, Huy Phan, and Alfred Mertins, "A recurrent neural network for hand gesture recognition based on accelerometer data," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 5088–5091.

[79] Dianchun Bai, Shutian Chen, Junyou Yang, et al., "Upper arm motion high-density semg recognition optimization based on spatial and time-frequency domain features," *Journal of Healthcare Engineering*, vol. 2019, 2019.

[80] István Ketykó, Ferenc Kovács, and Krisztián Zsolt Varga, "Domain adaptation for semg-based gesture recognition with recurrent neural networks," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–7.

[81] Usha Kuruganti, Ashirbad Pradhan, and Jacqueline Toner, "High-density electromyography provides improved understanding of muscle function for those with amputation," *Frontiers in Medical Technology*, vol. 3, pp. 690285, 2021.

[82] Tianyun Sun, Qin Hu, Paras Gulati, and S Farokh Atashzar, "Temporal dilation of deep lstm for agile decoding of semg: Application in prediction of upper-limb motor intention in neurorobotics," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6212–6219, 2021.

[83] Manfredo Atzori and Henning Müller, "Pawfe: Fast signal feature extraction using parallel time windows," *Frontiers in neurorobotics*, vol. 13, pp. 74, 2019.

[84] Weidong Geng, Yu Du, Wenguang Jin, Wentao Wei, Yu Hu, and Jiajun Li, "Gesture recognition by instantaneous surface emg images," *Scientific reports*, vol. 6, no. 1, pp. 1–8, 2016.

[85] Ulysse Cote-Allard, Cheikh Latyr Fall, Alexandre Drouin, Alexandre Campeau-Lecours, Clement Gosselin, Kyrre Glette, Franccois Laviolette, and Benoit Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE*

transactions on neural systems and rehabilitation engineering, vol. 27, no. 4, pp. 760–771, 2019.

[86] Hongfeng Chen, Runze Tong, Minjie Chen, Yinfeng Fang, and Honghai Liu, "A hybrid cnn-svm classifier for hand gesture recognition with surface emg signals," in *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*. IEEE, 2018, vol. 2, pp. 619–624.

[87] Md Johirul Islam, Shamim Ahmad, Fahmida Haque, Mamun Bin Ibne Reaz, Mohammad AS Bhuiyan, and Md Rezaul Islam, "A novel signal normalization approach to improve the force invariant myoelectric pattern recognition of transradial amputees," *IEEE Access*, vol. 9, pp. 79853–79868, 2021.

[88] Cheng Shen, Zhongcai Pei, Weihai Chen, Jianhua Wang, Jianbin Zhang, and Zuobing Chen, "Toward generalization of semg-based pattern recognition: a novel feature extraction for gesture recognition," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[89] Rami N Khushaba, Ali H Al-Timemy, Ahmed Al-Ani, and Adel Al-Jumaily, "A framework of temporal-spatial descriptors-based feature extraction for improved myoelectric pattern recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1821–1831, 2017.

[90] Md Johirul Islam, Shamim Ahmad, Fahmida Haque, Mamun Bin Ibne Reaz, Mohammad Arif Sobhan Bhuiyan, and Md Rezaul Islam, "Application of min-max normalization on subject-invariant emg pattern recognition," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[91] Mojisola Grace Asogbon, Oluwarotimi Williams Samuel, Yanjuan Geng, Olugbenga Oluwagbemi, Ji Ning, Shixiong Chen, Naik Ganesh, Pang Feng, and Guanglin Li, "Towards resolving the co-existing impacts of multiple dynamic factors on the performance of emg-pattern recognition based prostheses," *Computer methods and programs in biomedicine*, vol. 184, pp. 105278, 2020.

[92] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz, "Hand gesture recognition with 3d convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 1–7.

[93] Abeer Alnuaim, Mohammed Zakariah, Wesam Atef Hatamleh, Hussam Tarazi, Vikas Tripathi, and Enoch Tetteh Amoatey, "Human-computer interaction with hand gesture recognition using resnet and mobilenet," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[94] Panagiotis Tsinganos, Bruno Cornelis, Jan Cornelis, Bart Jansen, and Athanassios Skodras, "Improved gesture recognition based on semg signals and tcn," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1169–1173.

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

[96] Tamás Kapelner, Francesco Negro, Oskar C Aszmann, and Dario Farina, "Decoding motor unit activity from forearm muscles: perspectives for myoelectric control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 1, pp. 244–251, 2017.

[97] Chen Chen, Yang Yu, Shihan Ma, Xinjun Sheng, Chuang Lin, Dario Farina, and Xiangyang Zhu, "Hand gesture recognition based on motor unit spike trains decoded from high-density electromyography," *Biomedical signal processing and control*, vol. 55, pp. 101637, 2020.

[98] Chenyun Dai and Xiaogang Hu, "Extracting and classifying spatial muscle activation patterns in forearm flexor muscles using high-density electromyogram recordings," *International Journal of Neural Systems*, vol. 29, no. 01, pp. 1850025, 2019.

[99] Aleš Holobar and Damjan Zazula, "Gradient convolution kernel compensation applied to surface electromyograms," in *International Conference on Independent Component Analysis and Signal Separation*. Springer, 2007, pp. 617–624.

[100] Ales Holobar and Damjan Zazula, "Multichannel blind source separation using convolution kernel compensation," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4487–4496, 2007.

[101] Maoqi Chen and Ping Zhou, "A novel framework based on fastica for high density surface emg decomposition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 1, pp. 117–127, 2015.

[102] Tamás Kapelner, Ivan Vujaklija, Ning Jiang, Francesco Negro, Oskar C Aszmann, Jose Principe, and Dario Farina, "Predicting wrist kinematics from motor unit discharge timings for the control of active prostheses," *Journal of neuroengineering and rehabilitation*, vol. 16, no. 1, pp. 1–11, 2019.

[103] Todd R Farrell and Richard F Weir, "The optimal controller delay for myoelectric prostheses," *IEEE Transactions on neural systems and rehabilitation engineering*, vol. 15, no. 1, pp. 111–118, 2007.

[104] Nawadita Parajuli, Neethu Sreenivasan, Paolo Bifulco, Mario Cesarelli, Sergio Savino, Vincenzo Niola, Daniele Esposito, Tara J Hamilton, Ganesh R Naik, Upul Gunawardana, et al., "Real-time emg based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation," *Sensors*, vol. 19, no. 20, pp. 4596, 2019.

[105] Xiaogang Hu, Rymer William Z., and Suresh Nina L., "Motor unit pool organization examined via spike-triggered averaging of the surface electromyogram," *Journal of Neurophysiology*, vol. 110, no. 5, pp. 1205–1220, 2013.

[106] Alessandro Del Vecchio, Negro Francesco, Felici Francesco, and Farina Dario, "Associations between motor unit action potential parameters and surface emg features," *Journal of Applied Physiology*, vol. 123, no. 4, pp. 835–843, 2017.

[107] Pei Xu, "A real-time hand gesture recognition and human-computer interaction system," *arXiv preprint arXiv:1704.07296*, 2017.

[108] Eion Tyacke, Shreyas PJ Reddy, Natalie Feng, Rama Edlabadkar, Shucong Zhou, Jay Patel, Qin Hu, and S Farokh Atashzar, "Hand gesture recognition via transient semg using transfer learning of dilated efficient capsnet: towards generalization for neurorobotics," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9216–9223, 2022.

[109] Zohreh HajiAkhondi-Meybodi, Arash Mohammadi, Ming Hou, Jamshid Abouei, and Konstantinos N Plataniotis, "Vit-cat: Parallel vision transformers with cross attention fusion for popularity prediction in mec networks," *arXiv preprint arXiv:2210.15125*, 2022.

[110] Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda, "Towards spike-based machine intelligence with neuromorphic computing," *Nature*, vol. 575, no. 7784, pp. 607–617, 2019.

[111] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[112] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[113] Kashu Yamazaki, Viet-Khoa Vo-Ho, Darshan Bulsara, and Ngan Le, "Spiking neural networks and their applications: A review," *Brain Sciences*, vol. 12, no. 7, pp. 863, 2022.

[114] Muhammad Arsalan, Avik Santra, Mateusz Chmurski, Moamen El-Masry, Gianfranco Mauro, and Vadim Issakov, "Radar-based gesture recognition system using spiking neural network," in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2021, pp. 1–5.

[115] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3857–3866.

[116] Jibin Wu, Yansong Chua, Malu Zhang, Qu Yang, Guoqi Li, and Haizhou Li, "Deep spiking neural network with spike count based learning rule," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–6.

[117] Friedemann Zenke and Surya Ganguli, "Superspike: Supervised learning in multilayer spiking neural networks," *Neural computation*, vol. 30, no. 6, pp. 1514–1541, 2018.