ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

ARTIFICIAL INTELLIGENCE

MASTER THESIS

in

Architectures And Platforms For Artificial Intelligence

FOUNDATION MODELS FOR EMG HUMAN-MACHINE INTERFACES

CANDIDATE
Matteo Fasulo

SUPERVISOR
Dr. Angelo Garofalo (DEI)

CO-SUPERVISORS
Giusy Spacone (ETHZ)
Dr. Yawei Li (ETHZ)
Dr. Andrea Cossettini (ETHZ)

Academic year 2024-2025

Session 2nd

A mia mamma e a mio papà, per aver creduto in me fin dall'inizio, per l'impegno costante e per avermi dato l'opportunità di intraprendere questo percorso di studi. La vostra fiducia e il vostro sostegno sono stati la base su cui ho potuto costruire ogni traguardo.

A mia sorella Emma, con la speranza che questo lavoro possa essere per te una piccola fonte di ispirazione e incoraggiamento, affinché tu possa affrontare i tuoi futuri studi con determinazione e serenità.

Ai miei nonni, fari di luce nei momenti più bui: grazie per la vostra presenza silenziosa ma sempre concreta, per l'impegno e per l'affetto che hanno rappresentato un sostegno fondamentale durante questi anni.

Ai miei amici più cari – Matteo, Lorenzo, Francesco, Simone, Alessio – con cui ho condiviso non solo il tempo dello studio ma anche momenti indimenticabili di vita. Un pensiero speciale a Veronica e Katia, per il loro supporto costante, le risate, le conversazioni infinite e la capacità di rendere più leggeri anche i periodi più difficili.

Ai miei compagni di corso – Luca, Abylay, Antonello, Davide, Gabriele, Maksim, Michele, Norberto e Paul – con i quali ho condiviso lezioni, fatiche, sfide e soddisfazioni. Insieme abbiamo reso più sopportabili i momenti difficili e più significativi i successi, trasformando il percorso accademico in un'esperienza di crescita collettiva.

A tutto l'Appartamento 4 e a chi, con fortuna, ne ha fatto parte, trasformando uno spazio quotidiano in un luogo di amicizia, crescita e condivisione.

Infine, ad Alessia, per la sua presenza instancabile, per il sostegno incondizionato e per aver accompagnato ogni passo di questo percorso accademico con pazienza, fiducia e affetto.

"Non quia difficilia sunt non audemus, sed quia non audemus difficilia sunt."

— Seneca

Contents

List of Acronyms **x**

1 Introduction **4**

2 Background **10**

2.1 Transformer Architecture and Attention Mechanisms 11

2.1.1 Encoder and Decoder 11

2.1.2 Scaled Dot-Product Attention 14

2.1.3 Multi-Head Attention 15

2.1.4 Positional Encoding 15

2.1.5 Rotary Positional Embedding 17

3 Related Work **19**

3.1 Classical Machine Learning Approaches 19

3.2 Deep Learning for EMG 20

3.2.1 Convolutional and Recurrent Models 20

3.2.2 Transformer-Based Models 20

4 Implementation **24**

4.1 Methodology 24

4.1.1 Overview 24

4.1.2 Model description 25

4.1.3 Pretraining objective and strategy 30

4.1.4 Pretraining setup and reproducibility 32
4.2 Experiments 33
4.2.1 Datasets and tasks 33
4.2.2 Evaluation protocols 34

5 Results **36**
5.1 Pretraining Reconstruction 36
5.2 A. Gesture classification 37
5.3 B. Discrete Gesture 38
5.4 C. Regression 39
5.5 D. Silent Speech 40
5.5.1 D1. Silent Speech Synthesis 40
5.5.2 D2. Silent Speech Recognition 41

6 Conclusion and Future Work **42**
6.1 Conclusion 42
6.2 Future Work 44

A Datasets **45**
A.1 Pretraining Datasets 45
A.2 Downstream Datasets 47

B More Evaluation Results **49**
B.1 Kinematic Regression 49
B.2 Silent Speech 49
B.3 Discrete Gesture 50
B.4 Visualization of EPN612 Experimental Results 52

C FLOPs and Peak Memory Usage **55**

Bibliography **56**

List of Figures

2.1 Transformer architecture overview as proposed by Vaswani et al [1]. The model consists of an Encoder and a Decoder, each containing multiple layers with self-attention and feed-forward networks. 10

4.1 Overview of the proposed EMG transformer-pretraining framework. 24

4.2 Example raw multi-channel EMG (5 channels). Notation used in the text: $\mathbf{X} \in \mathbb{R}^{T \times C}$, with T timesteps and C channels. . . . 25

4.3 Pre-LayerNorm Transformer encoder block (3 heads, $d = 192$). 26

4.4 General Encoder-Decoder architecture. The encoder processes the masked input EMG data, while the decoder reconstructs the original signal. The encoder uses multi head attention to capture temporal and spatial relationships, while the decoder employs only a linear projection to generate the output. 27

4.5 Overview of the finetuning pipeline with the proposed concat channel fusing approach. 31

5.1 Reconstruction of a windowed EMG recording by the proposed EMG Foundation Model. 36

B.1 Phoneme confusability (darker lines indicate more confusion - maximum darkness is 13% confusion) 50
B.2 Confusion matrix on the EPN612 test set. 52
B.3 ROC curves for each gesture in the EPN612 test set. 53
B.4 t-SNE embeddings for the EPN612 dataset. 53

List of Tables

4.1 Hyperparameters for masked EMG pre-training with EMG Transformer. 28
4.2 Hyperparameters for downstream fine-tuning with EMG Transformer. 33
4.3 sEMG corpora used for pretraining 33
4.4 Public datasets used for downstream evaluation. 34

5.1 EMG gesture recognition. Accuracy and F1 scores. 37
5.2 EMG Discrete Gesture. Classification Error Rate (CLER) for full sequence and windowed inference. 38
5.3 EMG regression on Ninapro DB8. Mean Absolute Error in Degrees across DoAs. 39
5.4 EMG Silent Speech Synthesis. Word Error Rate (WER) reported for the EMG–audio modality. 40
5.5 EMG Silent Speech Recognition. Word Error Rate (WER) reported for the EMG–text modality. 41

B.1 Complete EMG Kinematic Regression Results across-subject. Additional metrics are reported: Pearson correlation, R^2, RMSE, and Explained Variance. 49
B.2 Results for the most confused pairs of phonemes 51
B.3 Ablation study on the window size for the discrete gesture recognition task, measured by CLER. 51

C.1 Per-GPU FLOPs and peak memory usage during pretraining and finetuning. 55

List of Acronyms

ASR . . . . . Automatic Speech Recognition

CLER . . . . Classification Error Rate

CNN . . . . Convolutional Neural Network

CV . . . . . Computer Vision

DNN . . . . Deep Neural Network

ECG . . . . . Electrocardiography

EEG . . . . . Electroencephalography

EMG . . . . Electromyography

FM . . . . . Foundation Model

GPU . . . . . Graphics Processing Unit

GRU . . . . Gated Recurrent Unit

HCI . . . . . Human-Computer Interaction

k-NN . . . . k-Nearest Neighbors

LDA . . . . . Latent Discriminant Analysis

LLM . . . . Large Language Model

LSTM . . . . Long Short-Term Memory

NLP . . . . . Natural Language Processing

PPG . . . . . Photoplethysmography

RNN . . . . Recurrent Neural Network

RoPE . . . . Rotary Position Embedding

sEMG . . . . Surface Electromyography

SoA . . . . . State-of-the-Art

SSL . . . . . Self-Supervised Learning

SSR . . . . . Silent Speech Recognition

SVM . . . . Support Vector Machine

ViT . . . . . Vision Transformers

WER . . . . Word Error Rate

Acknowledgements

I would like to express my sincere gratitude to my supervisor Dr. Angelo Garofalo and co-supervisors Giusy Spacone, Dr. Yawei Li, Dr. Andrea Cossettini, and Professor Luca Benini, for their invaluable guidance, support, and constructive feedback throughout this Master Thesis. Their expertise and insights were instrumental in shaping this research. I also extend my thanks to the Integrated Systems Laboratory at ETH Zurich for providing the resources and environment necessary for this work, and to the Swiss National Supercomputing Centre (CSCS) for granting access to the Alps HPC infrastructure, which was essential for model training and experimentation.

Abstract

The development of generalizable models for Electromyography (EMG) signal analysis is a significant challenge, limited by high variability across subjects, conditions, and acquisition devices and platforms, alongside a reliance on large, task-specific labeled datasets. This thesis introduces a new paradigm to address these limitations: a compact, pre-trained Foundation Model specifically for the EMG domain. We propose an encoder-only Transformer architecture trained using a self-supervised, masked-signal modeling objective on large-scale unlabeled data. By adapting vision-style tokenization for multi-channel EMG and incorporating Rotary Positional Embedding to allow for extrapolation, the model learns robust and transferable representations.

The resulting 3.6 million parameter model demonstrates a remarkable combination of efficiency and high performance. It sets a new state-of-the-art on the EPN-612 (96.60% accuracy) and UCI EMG (97.86% accuracy) gesture recognition benchmarks, significantly outperforming prior models with over ten times the parameters. The model's versatility is further proven by achieving a competitive 8.53° Mean Absolute Error in cross-subject kinematic regression, surpassing LSTM baselines in discrete gesture decoding, and showing remarkable performance in silent speech recognition despite its unimodal, EMG-only pre-training regime.

This work validates that a single, self-supervised encoder can serve as a powerful foundation for diverse EMG tasks. Its high accuracy, coupled with a modest parameter count, paves the way for a new generation of robust, data-efficient human-machine interfaces and opens the door to their deployment on resource-constrained embedded environments.

Chapter 1

Introduction

Electromyography (EMG) is a biosensing technique that captures the electrical activity produced by the skeletal muscles. It is widely used in various applications, including prosthetics, rehabilitation, and Human-Computer Interaction (HCI). EMG methods are broadly classified into invasive approaches (e.g., needle EMG) and non-invasive techniques, with Surface Electromyography (sEMG) being especially suitable for wearable neuromuscular interfaces due to ease of use and comfort [2]. This makes sEMG highly relevant across diverse domains such as clinical diagnostics, rehabilitation, ergonomics, HCI, and sports science.

However, despite its wide applicability, EMG signal analysis poses several challenges. Signal characteristics can vary significantly between individuals due to differences in muscle physiology, electrode placement, and environmental factors. This variability can lead to difficulties in developing robust models that generalize well across different users and conditions. Additionally, EMG signals are non-stationary, evolving over time and affected by various noise sources, including electrical interference, motion artifacts, and

physiological signals [3]. These challenges require advanced signal processing techniques and machine learning models capable of handling the inherent variability and noise in the EMG data.

To tackle these complexities, various signal processing techniques have been combined with machine learning and deep learning strategies [4]. Early methods often relied on handcrafted features, but the field has steadily shifted toward deep learning paradigms with automatic representation learning capabilities. This transition has been driven by the need for more flexible and powerful models that can adapt to the diverse and complex nature of EMG signals.

Despite these advances, there remains a lack of models that can be effectively generalized to different users, electrode placements, and other variations. Traditional machine learning approaches often struggle with this variability, and while deep learning has shown promise, it typically requires large amounts of labeled data for training. This is where Foundation Model (FM) [5] come into play, offering a potential solution by leveraging Self-Supervised Learning (SSL) techniques to pre-train on vast amounts of unlabeled data.

FMs have revolutionized the fields of Natural Language Processing (NLP) and Computer Vision (CV) by enabling models to learn rich representations from large unlabeled datasets just by rethinking the pre-training process. This shift has opened up new possibilities for transfer learning, allowing models to be fine-tuned on specific tasks with relatively small amounts of labeled data.

Beyond vision and language, emerging work explores electrophysiological biosignals (EXG), ranging from Electrocardiography (ECG), Electroencephalography (EEG), EMG, Photoplethysmography (PPG), and related surface bioelectric recordings, using FMs [6, 7].

Common pretraining paradigms (masked reconstruction, contrastive temporal alignment, and cross-modal distillation) have begun to produce generalist encoders for EEG and ECG [8, 9], using large public repositories and relatively standardized channel configurations (e.g., 12-lead ECG, 10–20 EEG montages). These efforts report transferable gains in the detection of arrhythmias [10], the staging of sleep [11], the mental workload or affect classification [12], and the potential decoding related to events, indicating that self-supervised representation training can reduce the needs of labeled data.

In contrast, **no widely adopted FM exists specifically for EMG**: progress is limited by (i) higher inter-subject and session variability [13] (electrode placement shifts, skin impedance, muscle physiology), (ii) heterogeneous sensor layouts (channel count, spacing, high-density vs. sparse arrays), (iii) task diversity [14] (gestures, force/kinematics regression, silent speech, continuous neuromotor decoding), and (iv) fewer large, harmonized, openly licensed corpora compared to ECG/EEG datasets. Current EMG models are predominantly task-specialized, trained with supervised losses from scratch or modest transfer, limiting generalization between users and tasks.

This gap motivates the construction of an EMG-centric FM: a single pretrained encoder producing robust and reusable temporal representations that retain fine-grained motor intent while being resilient to domain shifts; an objective aligned with the broader FM paradigm [5] but still unrealized for EMG.

This work aims to bridge this gap by developing a Foundation Model specifically tailored for EMG signal analysis. Using self-supervised learning techniques, we aim to create a model that can effectively learn from large-scale EMG datasets, enabling it to generalize across different users, acquisition platforms and conditions. The proposed model will be evaluated on several downstream tasks, namely gesture recognition, regression, and silent speech recognition, to demonstrate its effectiveness and robustness.

State of the Art on Target Downstream Tasks

Current State-of-the-Art (SoA) systems on the downstream benchmarks that we report later (Chapter 5) highlight both strong absolute performance and several limitations that our approach addresses.

Gesture recognition (Ninapro DB5, EPN-612, UCI EMG). This task involves classifying time-series EMG data, typically from the forearm, into a set of discrete hand or wrist gestures. Recent Transformer or hybrid sequence encoders (Moment [15], OTiS [16], PhysioWave [17]) span from compact $\sim$ 5M to very large $\sim$ 385M parameters. The strongest published results in our benchmark set are obtained by the 37M parameter PhysioWave Large model: 87.53% Top-1 / 75.42 F1 on Ninapro DB5 [18], 94.50% / 94.56 F1 on EPN-612 [19], and 93.19% / 93.59 F1 on UCI EMG [20].

Discrete gesture sequence decoding (generic neuromotor interface). This advanced task involves decoding continuous sequences of discrete motor actions, such as individual finger movements for typing, from high-fidelity EMG signals. Recurrent LSTM models (6.4M parameters) on high-fidelity wrist/forearm interfaces (Meta / CTRL-labs, [21]) reach a Classification Error Rate (CLER) of 0.1819 on full sequences and 0.1596 on windowed inference.

Kinematic regression (Ninapro DB8).

In contrast to discrete classification, kinematic regression aims to predict continuous, multi-dimensional joint movements (e.g., finger and wrist angles) from forearm EMG signals. Lightweight temporal convolutional networks (TEMPONet TCN [22], <500K params) report per-subject mean absolute error (MAE) 6.89° across 5 DoAs. Event-driven linear regression [23] attains 8.8°$\pm$2.3° (per-subject), while older DeepNet+Kalman pipelines [24] exceed 13.5° (reported as RMSE, only 3 DoAs). Cross-subject performance is rarely

quantified and typically degrades relative to per-subject training, indicating limited representation transfer.

Silent speech (facial EMG to audio/text). This task focuses on decoding intended speech by translating EMG signals captured from facial muscles into either audible speech (synthesis) or written text (recognition). A 54M parameter Transformer model (Gaddy & Klein [25]) achieves 36% Word Error Rate (WER) (EMG-to-audio) and 28.8% WER (EMG-to-text). The Stanford MONA and MONA LISA works [26] reduce EMG-to-text WER to 22.2% and 12.2%, respectively, through greater capacity and contrastive alignment with audio, underscoring gains from richer multimodal pretraining but also additional complexity due to the need of modality-specific encoders.

Across tasks, SoA trends [17, 21, 23, 26] emphasize: (i) increasing reliance on Transformer-style architectures; (ii) substantial parameter counts or task-specific engineering; (iii) predominantly supervised training with limited explicit cross-user robustness analysis; and (iv) performance gaps between per-subject and cross-subject settings. These observations motivate a unified EMG Foundation Model: a single, moderately sized encoder pre-trained self-supervised on heterogeneous EMG to supply reusable, robust representations for classification, regression, and sequence decoding with reduced labeled data dependence.

Contribution summary. We introduce a novel EMG Foundation Model and rigorously benchmark it against state-of-the-art baselines across multiple EMG downstream tasks, including gesture classification, discrete gesture recognition, kinematic regression, and silent speech synthesis/recognition. Compared to existing large-scale generic time-series models (Moment, OTiS, PhysioWave, WaveFormer) and task-specific architectures (e.g., Meta-LSTM, TEMPONet), our model achieves new state-of-the-art results on EPN-612 (96.60% accuracy, 96.69 F1), UCI EMG (97.86% accuracy),

and Ninapro DB5 (84.53 F1) while using only a fraction of the parameters (3.6M vs. 37–385M). In discrete gesture recognition, it slightly outperforms prior LSTM-based methods under comparable windowed inference (CLER 0.1553 vs. 0.1596), and in regression tasks, it demonstrates strong cross-subject generalization (MAE 8.53°). For silent speech, finetuning reduces EMG-to-audio WER to 31.65% and EMG-to-text WER to 32.75%, highlighting the model's ability to handle both generative and discriminative EMG tasks. Overall, this work demonstrates that a compact foundation model can significantly advance EMG modeling, offering superior performance, cross-task versatility, and efficient deployment potential.

Chapter 2

Background

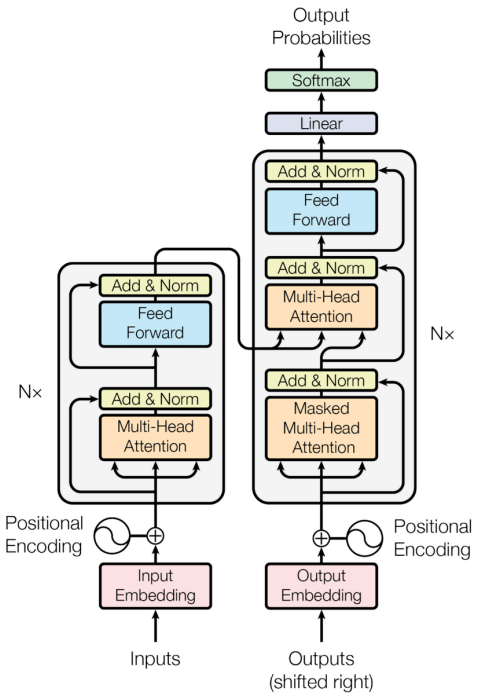

Figure 2.1: Transformer architecture overview as proposed by Vaswani et al [1]. The model consists of an Encoder and a Decoder, each containing multiple layers with self-attention and feed-forward networks.

2.1 Transformer Architecture and Attention Mechanisms

The Transformer architecture, introduced by Vaswani et al. [1], revolutionized sequence modeling by eliminating recurrence and convolution in favor of self-attention mechanisms. This enabled faster, more parallelizable models that outperform traditional RNNs and CNNs in many domains, including NLP and time-series analysis.

2.1.1 Encoder and Decoder

The architecture comprises an Encoder and a Decoder, each consisting of multiple identical layers:

Encoder: Each encoder layer contains:

- Multi-head self-attention sub-layer
- Feed-forward network sub-layer
- Residual connections and Layer Normalization

Skip Connections The input of a transformer block is an embedding for a token, which has dimension d. This initial embedding gets passed up (by residual connections) and is progressively added to by the other components of the transformer: the attention layer and the feedforward layer. Residual or skip connections help the model learn identity mappings, effectively preventing the vanishing gradient problem and rank collapse [27].

Before the attention and feedforward layer, Layer Normalization is applied to the input embedding. In this way, the input embedding is normalized before

being passed to the attention layer, and the result is added back to the input embedding via a residual connection.

Feedforward layer The feedforward layer is a fully-connected 2-layer network with one hidden layer and two weight matrices. The weights are the same for each token position i, but are different from layer to layer. It is common to make the dimensionality d_{ff} of the hidden layer of the feedforward network to be larger than the dimensionality of the model d. (for example, with an expansion factor of 4, i.e. $d_{ff} = 4d$).

$$\mathrm{FFN}(\mathbf{x_i}) = \mathrm{ReLU}(\mathbf{x_i}\mathbf{W_1} + b_1)\mathbf{W_2} + b_2 \tag{2.1}$$

Layer Norm At two stages in the transformer block, we normalize the input vector via a process called Layer Normalization. Layer Normalization is one of the many forms of normalization that can be used to improve training performance in Deep Neural Network (DNN) by keeping the values of a hidden layer in a range that facilitates gradient-based training.

Layer normalization is a particular case of **z-score** normalization but applied to a single vector in a hidden layer. Layer normalization is applied to the embedding vector of a single token, thus the input to the layer norm is a single vector of dimensionality d and the output is that vector normalized, again of dimensionality d. The first step in layer normalization computes the mean μ and the standard deviation σ, over the elements of the vector to be normalized.

Given an embedding vector $\mathbf{x}$ of dimensionality d, these values are calculated as follows:

$$\mu = \frac{1}{d}\sum_{i=1}^{d} x_i \tag{2.2}$$

$$\sigma = \sqrt{\frac{1}{d}\sum_{i=1}^{d}(x_i - \mu)^2} \tag{2.3}$$

The normalized vector is then computed with two additional learnable parameters γ and β representing the gain and offset values, respectively. The normalized vector is computed as follows:

$$\text{LayerNorm}(\mathbf{x}) = \gamma \frac{(\mathbf{x} - \mu)}{\sigma} + \beta \tag{2.4}$$

The original architecture used post layer normalization, where normalization was applied after residual addition, while more recent implementations often adopted pre layer normalization for improved training stability [28].

According to post layer normalization, given an input $\mathbf{X}$, the output is computed as:

$$\mathbf{O} = \text{LayerNorm}(\mathbf{X} + \text{MultiHeadAttention}(\mathbf{X})) \tag{2.5}$$

$$\mathbf{H} = \text{LayerNorm}(\mathbf{O} + \text{FFN}(\mathbf{O})) \tag{2.6}$$

while for pre layer normalization, the output is computed as:

$$\mathbf{O} = \mathbf{X} + \text{MultiHeadAttention}(\text{LayerNorm}(\mathbf{X})) \tag{2.7}$$

$$\mathbf{H} = \mathbf{X} + \text{FFN}(\text{LayerNorm}(\mathbf{O})) \tag{2.8}$$

Decoder: Each decoder layer includes the following:

- Masked self-attention
- Cross-attention to encoder outputs
- Feed-forward network
- Residual connections and LayerNorm

The decoder uses masking self-attention to ensure autoregressive generation, attending only to previously generated tokens instead of future ones. In this way, causality can be maintained, and the model can generate sequences in a step-by-step manner.

Masked self-attention is computed as

$$\mathbf{A} = \mathrm{Softmax}\left(\mathrm{Mask}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\right)\mathbf{V} \tag{2.9}$$

where Mask is a function that sets the elements in the upper-triangular portion of the matrix to $-\infty$, ensuring that the model does not take care of future tokens.

In practice, this is done by adding a mask matrix M in which $M_{i,j} = -\infty \quad \forall j > i$ (i.e. for the upper-triangular portion) and $M_{i,j} = 0$ otherwise.

2.1.2 Scaled Dot-Product Attention

Given input $\mathbf{X} \in \mathbb{R}^{n \times d}$, where n is the sequence length and d is the embedding dimension; queries, keys, and values are obtained by learning linear projections.

$$\mathbf{Q} = \mathbf{X}\mathbf{W}^{\mathbf{Q}} \tag{2.10}$$

$$\mathbf{K} = \mathbf{X}\mathbf{W}^{\mathbf{K}} \tag{2.11}$$

$$\mathbf{V} = \mathbf{X}\mathbf{W}^{\mathbf{V}} \tag{2.12}$$

where $\mathbf{W}^{\mathbf{Q}}, \mathbf{W}^{\mathbf{K}} \in \mathbb{R}^{d \times d_k}$ and $\mathbf{W}^{\mathbf{V}} \in \mathbb{R}^{d \times d_v}$ are weight matrices for queries, keys, and values, respectively. The attention mechanism computes the attention scores as follows:

The core attention mechanism is as follows:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}, \tag{2.13}$$

which enables the model to compute attention weights over all pairs of input tokens.

2.1.3 Multi-Head Attention

Multi-Head Attention allows learning multiple attention patterns in parallel:

$$\text{head}_i = \text{Attention}(\mathbf{Q^i}, \mathbf{K^i}, \mathbf{V^i}) = \text{Softmax}\left(\frac{\mathbf{Q^i}\mathbf{K^i}^\top}{\sqrt{d_k}}\right)\mathbf{V^i}, \tag{2.14}$$

$$\text{MultiHeadAttention}(\mathbf{X}) = (\text{head}_1 \oplus \text{head}_2 \cdots \oplus \text{head}_h)\mathbf{W^O}, \tag{2.15}$$

Each head captures distinct relationships; their outputs are concatenated and projected back to the embedding space. Typically, $d_v = d/h$ for h attention heads.

2.1.4 Positional Encoding

Transformers are permutation-invariant on the input token (time) dimension unless explicit order information is injected. Let $\mathbf{x}_{pos} \in \mathbb{R}^d$ be the embedding of the token at position $pos \in \{0, \ldots, n-1\}$. A positional encoding $\mathbf{p}_{pos} \in \mathbb{R}^d$ is added (or sometimes concatenated) to form

$$\tilde{\mathbf{x}}_{pos} = \mathbf{x}_{pos} + \mathbf{p}_{pos}. \tag{2.16}$$

Absolute Sinusoidal Encoding (Vaswani et al.). Define inverse frequencies

$$\omega_i = 10000^{-2i/d}, \qquad i = 0, \ldots, d/2 - 1. \tag{2.17}$$

Then for each pair of dimensions $(2i, 2i+1)$

$$p_{pos,2i} = \sin(pos \cdot \omega_i), \tag{2.18}$$

$$p_{pos,2i+1} = \cos(pos \cdot \omega_i). \tag{2.19}$$

In matrix form, let $\mathbf{\Omega} = (\omega_0, \ldots, \omega_{d/2-1})^\top$:

$$\mathbf{p}_{pos} = \Big[\sin(pos\mathbf{\Omega}) \parallel \cos(pos\mathbf{\Omega})\Big] \in \mathbb{R}^d. \tag{2.20}$$

Key properties. 1. *Deterministic / parameter-free*: no learned parameters; enables extrapolation beyond training length. 2. *Multiscale*: frequencies form a geometric progression that covers short and long range. 3. *Linear relative shift signal inside dot products*: Consider (single head) attention logits after projection:

$$\alpha_{t,s} = \frac{(\mathbf{q}_t + \mathbf{W}^Q\mathbf{p}_t) \cdot (\mathbf{k}_s + \mathbf{W}^K\mathbf{p}_s)}{\sqrt{d_k}}. \tag{2.21}$$

Cross terms $(\mathbf{W}^Q\mathbf{p}_t) \cdot (\mathbf{W}^K\mathbf{p}_s)$ include $\sin(t\omega)\sin(s\omega) + \cos(t\omega)\cos(s\omega) = \cos((t-s)\omega)$, providing an implicit encoding of relative position $t-s$. Thus, absolute encodings induce relative phase signals.

Limitations for long sequences. For very long sequences, highest frequencies may become too dense (i.e. phase wrapping), and absolute addition fixes every position to a unique pattern, less flexible when only relative timing is important (e.g. EMG muscle activation pattern shifts). This motivates rotational / relative formulations.

Learned Absolute Positional Embeddings. An alternative is to learn a table $\mathbf{P} \in \mathbb{R}^{n_{\max} \times d}$ with $\mathbf{p}_{pos} = \mathbf{P}_{pos}$. The advantages are that it allows for task adaptation, but it suffers from poor extrapolation beyond $n_{\max}$ and requires interpolation or resizing for longer inputs. For biosignals where the acquisition window length can vary, deterministic or relative encodings are often

preferable.

2.1.5 Rotary Positional Embedding

Rotary Position Embedding (RoPE) introduce position by **rotating** query and key subvectors in 2D planes instead of adding a position vector [29]. This yields attention scores that depend on relative positions through phase differences, enabling clean extrapolation and better inductive bias for continuous signals like EMG [30, 31, 32]. Given the fact that EMG windows can shift in time (latency variation), RoPE makes attention logits depend primarily on relative offsets, aiding pattern alignment (e.g. onset vs. peak) and generalizing across window boundaries. Multi-frequency rotations capture both short bursts (high frequency muscular activity) and longer envelope trends.

Construction. Split a d-dimensional vector into $d/2$ complex (or real 2D) components. For inverse frequencies ω_i as above, define an angle

$$\theta_{pos,i} = pos \cdot \omega_i. \tag{2.22}$$

Define the 2D rotation matrix

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{2.23}$$

Given a (projected) query (or key) vector $\mathbf{q} \in \mathbb{R}^d$, reshape into pairs $\mathbf{q}^{(i)} \in \mathbb{R}^2$. Apply:

$$\mathrm{RoPE}(\mathbf{q}, pos) = \bigoplus_{i=0}^{d/2-1} \mathbf{R}(\theta_{pos,i})\mathbf{q}^{(i)}. \tag{2.24}$$

Do the same for the keys to obtain $\hat{\mathbf{q}}_{pos}$, $\hat{\mathbf{k}}_{pos}$.

Complex notation. Map each pair (q_{2i}, q_{2i+1}) to $z_i = q_{2i} + jq_{2i+1}$. Then

$$\mathrm{RoPE}(z_i, pos) = z_i \cdot e^{j\theta_{pos,i}}. \tag{2.25}$$

Relative position emerges in dot product. Consider a single frequency component with complex numbers for brevity. The contribution to the attention logit between positions t and s:

$$\Re\left\{z_q e^{j\theta_t} \cdot \overline{z_k e^{j\theta_s}}\right\} = \Re\left\{z_q \overline{z_k} e^{j(\theta_t - \theta_s)}\right\}, \tag{2.26}$$

which depends only on the difference $\theta_t - \theta_s = (t-s)\omega$. Summing over frequencies yields multiscale relative encoding without explicit relative position matrices.

Attention formula with RoPE. Let $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{n \times d_k}$ after linear projections. Define rotary versions $\hat{\mathbf{Q}}, \hat{\mathbf{K}}$ by rotating each row according to its position index. Scaled dot-product attention becomes

$$\text{Attention} = \text{Softmax}\left(\frac{\hat{\mathbf{Q}}\hat{\mathbf{K}}^\top}{\sqrt{d_k}}\right)\mathbf{V}. \tag{2.27}$$

Comparison to additive sinusoidal. For additive sinusoidal embeddings, the encoding is absolute + implicit relative (via trigonometric identities). In RoPE, it is directly relative in logit space (phase differences), eliminating absolute bias with no extra parameters (same as sinusoidal). Additionally, extrapolation is robust with angles growing linearly with position, preserving relative differences.

Relation to other methods. Shaw et al. [33] introduce learned relative embeddings $\mathbf{a}_{t-s}$ added directly to the logits. T5 [34] employs bucketed relative position biases, while ALiBi [35] applies linear, distance-dependent biases. In contrast, RoPE captures relative position continuously across multiple frequencies, without relying on embedding tables or additional parameters.

Chapter 3

Related Work

3.1 Classical Machine Learning Approaches

Early EMG research relied heavily on handcrafted features extracted from the time-domain, the frequency-domain or time–frequency representations. These features were typically fed into Support Vector Machine (SVM), Latent Discriminant Analysis (LDA), or k-Nearest Neighbors (k-NN). Although such pipelines achieved respectable accuracy in controlled settings, they suffered from the following.

- **Limited generalizability:** Feature sets tuned for one gesture set or electrode montage often failed when electrodes were repositioned or subjects changed [36].

- **Extensive manual effort:** Designing and validating robust feature extractors requires deep domain expertise and iterative experimentation [37].

- **Sensitivity to noise:** Motion artifacts, cross-talk, and electrode

impedance variations could drastically degrade classification performance without careful preprocessing [38].

3.2 Deep Learning for EMG

3.2.1 Convolutional and Recurrent Models

The advent of deep learning enabled automatic feature extraction directly from raw or minimally preprocessed EMG waveforms. Convolutional Neural Network (CNN) treat the EMG as a 1D time-series or a 2D time-frequency ”image,” capturing local temporal patterns and inter-channel correlations [39, 40]. Recurrent Neural Network (RNN), especially Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), have been applied to model temporal dependencies and the dynamics of muscle activation, e.g. across amputees, force levels, and non-stationary muscle activations [41, 42, 43]. Hybrid CNN–RNN models combined both strengths, but still encountered:

- **Limited receptive field:** Vanilla CNNs capture only local context unless made very deep or widened, increasing the parameter count [40].

- **Sequential bottlenecks:** RNNs process timesteps one at a time, constraining parallelism, and slowing training [42]

3.2.2 Transformer-Based Models

Early adaptations of the vanilla Transformer to generic time-series tokenize each timestep independently (sequence length n equals raw sample count or frame count). Self-attention then offers a global receptive field since any timestep can be accommodated by any other in one layer, capturing long-range muscle co-activation or temporal context (e.g. preparation–execution phases). The inherent parallelism removes the typical sequential dependency of RNNs.

However, naive per-timestep tokenization exhibits several drawbacks: (i) the quadratic cost of memory and FLOPs with respect to the sequence length n, which for high-density EMG can quickly become prohibitive [44, 45]; (ii) the low signal-to-token ratio, where each sample carries little semantic information, leading to noisy attention weights [46]; (iii) the inductive bias of treating each timestep as an independent token, which is not well suited for continuous signals like EMG [44, 45]; and (iv) the over-sensitivity to misalignment, where minor latency jitters can shift many tokens, reducing robustness [46].

Time-Series Transformers. Time Series Transformers [47] mitigate sequence length or emphasize temporal priors via: (i) sparse or probabilistic attention variants with reduced pairwise interactions [48]; (ii) low-rank approximations like linearized attention [49]; (iii) hierarchical pooling or pyramidal structures that progressively shorten the sequence length [50]. These reduce cost but still treat the elementary token as (near) a raw timestep.

Patch (ViT-Like) Embedding for Time-Series. Inspired by Vision Transformers (ViT) [51], a 1D patch embedder slices the signal into non-overlapping (or mildly overlapping) windows of length L; each patch is linearly projected onto a d-dimensional token:

$$\mathbf{z}_i = \mathrm{Proj}\Big(\mathbf{X}_{iL:(i+1)L,:}\Big), \quad i = 0, \ldots, \frac{n}{L} - 1. \tag{3.1}$$

with several benefits, namely the reduction in the length of the sequence from n to $\frac{n}{L}$, the higher semantic density given that each token aggregates local temporal patterns, and implicit local smoothing/denoising, thus reducing high-frequency noise before attention. This approach is particularly effective for EMG, where muscle activation patterns often span several samples and the signal can be noisy due to motion artifacts or variations in the impedance of the electrode. The drawbacks of patch-based tokenization lie in the loss of fine-grained temporal resolution, as the model can only attend to patterns on

the scale of L samples, hence the need to pick a L aligned with the temporal task granularity.

Channel Awareness and ChannelViT Motivation. The surface EMG (and the wider EXG) recordings are multichannel; each electrode captures spatially localized physiology (muscle belly, proximity to the innervation zone, crosstalk). Simple patching that first concatenates channels along time and collapses them into a single vector could inadvertently blur inter-channel spatial structure by underrepresenting weaker but discriminative channels and losing electrode identity [52]. This is especially problematic for EMG, where electrode placement can vary significantly between subjects and tasks, leading to different spatial patterns of muscle activation.

ChannelViT [53] addresses this by constructing patch tokens independently from each input channel. This simple modification to the original ViT architecture enables the model to reason across both locations and channels. However, while ChannelViT can leverage existing efficient implementations of ViT with minimal modifications, increasing the sequence length introduces additional computational requirements, thus the choice of L becomes even more critical than before.

When each patch token is derived from a single channel, the sequence length becomes $\frac{n}{L} \cdot C$, where C is the number of channels. This means that the overall computational cost scales with the number of channels, making it essential to balance the patch size L with the number of channels C to maintain efficiency.

Applications to Biosignals (EXG). Recent work has explored Transformer-based architectures for various biosignals, including but not limited to EEG, ECG, and PPG.

FM for wearable biosignals [54] proposes a foundation model approach to leverage large-scale pre-training for wearable biosignal data (PPG and ECG).

BrainBERT [55] uses an Intracranial Electroencephalograph (iEEG) spectrogram as tokens to a Transformer encoder, pre-trained via Masked Autoencoding (MAE) [56].

LaBraM [57] introduces a learnable neural tokenizer that maps EEG waveform patches to embeddings and then processes them via the MAE framework.

Neuro-GPT [58] adopts causal auto-regressive MAE for EEG waveform modeling similar to Large Language Model (LLM) pre-training, but with a focus on EEG data.

CEReBrO [8] draws inspiration from ChannelViT, proposing a compact encoder for EEG using a ViTMAE [56] pre-training approach with an encoder-decoder architecture which processes only visible tokens in the encoder and then reconstructs the full sequence in the decoder.

Chapter 4

Implementation

4.1 Methodology

4.1.1 Overview

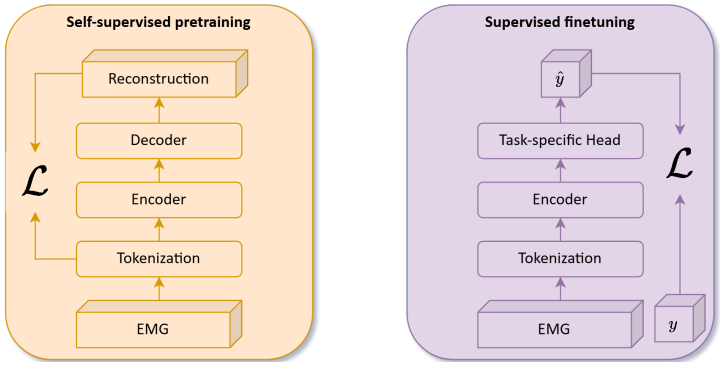

Figure 4.1: Overview of the proposed EMG transformer-pretraining framework.

The model adapts vision-style tokenization and masked image modeling to multi-channel EMG by: (i) converting raw waveforms into a sequence of temporally local, channel-aware patch tokens; (ii) applying an *encoder-only*

Transformer with RoPE for relative temporal inductive bias and length extrapolation; (iii) reconstructing masked patches from a lightweight linear head so that representational burden is almost entirely in the encoder.

4.1.2 Model description

Tokenization

Following current literature [59], EMG waveforms are sliced into equally-sized non-overlapping patches to: (i) reduce sequence length thus computation and memory usage; (ii) extract semantic information and improve locality; (iii) attend to long-range dependencies.

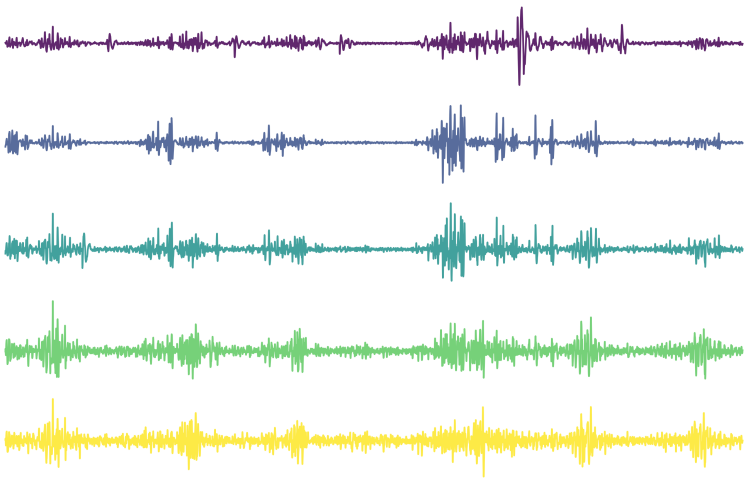

Figure 4.2: Example raw multi-channel EMG (5 channels). Notation used in the text: $\mathbf{X} \in \mathbb{R}^{T \times C}$, with T timesteps and C channels.

Let the EMG waveform be $\mathbf{X} \in \mathbb{R}^{T \times C}$ (timesteps T, channels C). For a patch length of L and a stride S, the set of results of patches is $\mathbf{P} \in \mathbb{R}^{N_p \times C \times L}$ where:

$$N_p = \left\lfloor \frac{T - L}{S} \right\rfloor + 1. \tag{4.1}$$

is the number of patches *per-channel*.

Each patch $\mathbf{P}_{c,i} \in \mathbb{R}^L$ from channel c and patch index i is projected onto an embedding space of dimension d_e using a learnable linear projection $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{d_e \times L}$. The final embedding patches are given by $\mathbf{E}_{c,i} = \mathbf{W}_{\text{proj}} \mathbf{P}_{c,i}^{\top}$. No learnable positional embedding is added as RoPE already incorporates such information.

This per-channel patch granularity allows the model to capture both temporal dynamics within each channel and spatial relationships across channels, which is crucial for EMG data, where different muscles may exhibit distinct activation patterns.

Encoder architecture

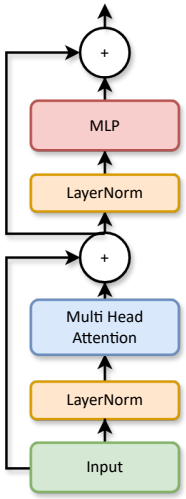

Figure 4.3: Pre-LayerNorm Transformer encoder block (3 heads, $d = 192$).

The model architecture is based on a pre-LayerNorm Transformer encoder with RoPE composed by 8 layers, each with 3 attention heads and embedding

dimension 192 as visible in figure 4.3.

Reconstruction head and forward pass

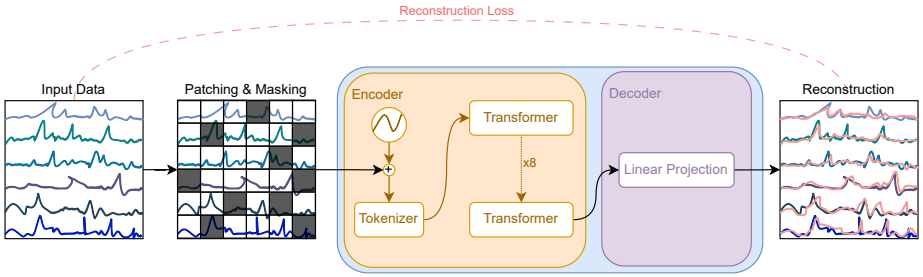

Figure 4.4: General Encoder-Decoder architecture. The encoder processes the masked input EMG data, while the decoder reconstructs the original signal. The encoder uses multi head attention to capture temporal and spatial relationships, while the decoder employs only a linear projection to generate the output.

Table 4.1: Hyperparameters for masked EMG pre-training with EMG Transformer.

Hyperparameter	EMG Transformer
Transformer Encoder	
Timesteps	1000
Patch size $(H \times W)$	$\{1, 20\}$
Number of channels	16
Embed dimension	192
Encoder layers	8
Attention heads	3
QKV bias	Yes
QK norm	No
MLP ratio	4.0
MLP size	768
Attn drop	0.1
Proj drop	0.1
Drop-path	0.1
Decoder	
Decoder embed dim	192
Pre-training Setup	
Batch size	512
Peak / minimal LR	1×10^{-4} / 1×10^{-6}
Optimizer (β_1, β_2)	AdamW (0.9, 0.98)
LR scheduler	Cosine
Weight decay	0.01
Total / warm-up epochs	50 / 10
Accumulated grad batches	8
Gradient clipping	3
Mask ratio	0.5
Max sequence length	1000

The encoder output is then passed to a linear layer that outputs a sequence $\hat{\mathbf{P}}$, which is a reconstruction of the original patch sequence $\mathbf{P}$.

The choice of avoiding a deep decoder is in contrast to the asymmetric MAE design but motivated by the fact that, in this way, all the burden of the reconstruction is onto the encoder, without relying on a complex decoder. This design choice emphasizes the encoder's ability to learn rich representations that can generalize across different tasks and conditions.

The complete pre-training framework is illustrated in Figure 4.4.

Parameter budget and deployability

The full model has approximately $\mathbf{3.6 \times 10^6}$ parameters. To relate this budget to embedded targets, we convert parameter counts to memory footprints under common numeric formats. A single parameter requires 4 B in FP32, 1 B in INT8, and 0.5 B when packed in INT4; therefore the model weights occupy approximately:

$$\begin{aligned} \text{FP32: } & 3.6 \times 10^6 \times 4\,\text{B} = 14{,}400{,}000\,\text{B} \approx 13.73\,\text{MiB}, \\ \text{INT8: } & 3.6 \times 10^6 \times 1\,\text{B} = 3{,}600{,}000\,\text{B} \approx 3.43\,\text{MiB}, \\ \text{INT4 (packed): } & 3.6 \times 10^6 \times 0.5\,\text{B} = 1{,}800{,}000\,\text{B} \approx 1.72\,\text{MiB}. \end{aligned}$$

By comparison, GreenWaves GAP9 devices expose an on-chip L2 SRAM on the order of ~ 1.5 MiB and cluster L1/TCDM slices of ~ 128 KiB, with an additional on-package nonvolatile/eMRAM region (≈ 2 MiB) and the possibility to attach external PSRAM.

4.1.3 Pretraining objective and strategy

Masking strategy

During pre-training a random subset $\mathcal{M}$ of tokens is replaced by a learnable [MASK] token. We sample independently per sample, each iteration without a fixed schedule, promoting reconstruction robustness across diverse occlusion patterns. Masking at the *patch* granularity enforces contextual inference over tens of milliseconds (physiologically meaningful burst segments), rather than trivial gap filling. Extremely low ratios (BERT-style 15%) underutilized reconstruction capacity, while very high ratios (>70%) destabilized early optimization. The adopted mid ratio of 50% balances the removal of information and the gradient signal and is supported by prior works [8, 60].

Loss and targets

The chosen loss function is the Smooth L1 loss, defined as:

$$\mathrm{SmoothL1}(x, y) = \begin{cases} 0.5(x-y)^2/\beta & \text{if } |x-y| < \beta \\ |x-y| - 0.5 \cdot \beta & \text{otherwise} \end{cases} \tag{4.2}$$

where β is a hyperparameter that controls the transition point between the loss L2 when the absolute difference is small, and the loss L1 when it is large. Generally, it is less sensitive to outliers than L2 loss and in some cases prevents exploding gradients. This follows recent work such as PhysioWave [17], which uses Smooth-L1 as the reconstruction objective between masked and original patches in physiologic signal pretraining.

We define the following loss components:

$$\mathcal{L}_{\text{masked}} = \frac{1}{|\mathcal{M}|} \sum_{(c,i)\in\mathcal{M}} \mathrm{SmoothL1}(\mathbf{P}_{c,i}, \hat{\mathbf{P}}_{c,i}) \tag{4.3}$$

$$\mathcal{L}_{\text{visible}} = \frac{1}{|\bar{\mathcal{M}}|} \sum_{(c,i)\in\bar{\mathcal{M}}} \text{SmoothL1}(\mathbf{P}_{c,i}, \hat{\mathbf{P}}_{c,i}) \tag{4.4}$$

where $\mathcal{M}$ and $\bar{\mathcal{M}}$ are the sets of masked and visible patches, respectively.

The total loss function during pre-training is:

$$\mathcal{L} = \mathcal{L}_{\text{masked}} + \alpha \cdot \mathcal{L}_{\text{visible}} \tag{4.5}$$

Reconstruction target / decoder details

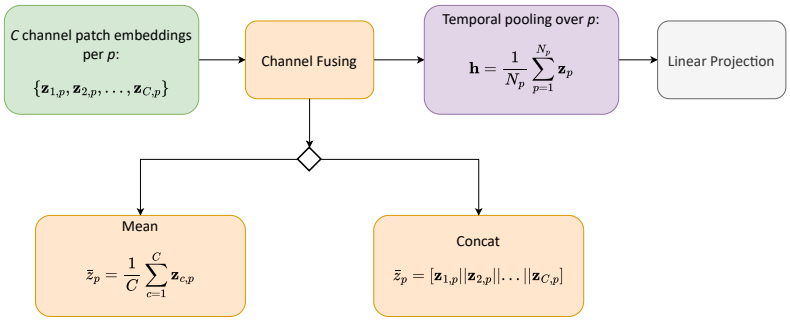

Figure 4.5: Overview of the finetuning pipeline with the proposed concat channel fusing approach.

The pre-trained encoder outputs $C \cdot N_p$ patch tokens. We omit a [CLS] token and instead fuse channel information per patch, then pool over time.

Channel fusion. For each temporal patch index p we have $\{\mathbf{z}_{c,p} \in \mathbb{R}^{d_e}\}_{c=1}^{C}$.

$$\text{Mean: } \bar{\mathbf{z}}_p = \frac{1}{C}\sum_{c=1}^{C}\mathbf{z}_{c,p} \in \mathbb{R}^{d_e} \qquad \text{Concat: } \tilde{\mathbf{z}}_p = [\mathbf{z}_{1,p}\| \cdots \|\mathbf{z}_{C,p}] \in \mathbb{R}^{Cd_e}. \tag{4.6}$$

Mean gives channel invariance (robust, lower dimensional) but discards differential activation patterns given by selective muscle recruitment, which can be important for distinguishing between different gestures.

Concat preserves per-electrode structure, allowing the linear head to weight the channels independently (useful under placement shifts or heterogeneous

SNR), at the cost of higher dimensionality and higher overfitting risk (later mitigated with weight decay and label smoothing). Empirically concat improved significantly over mean in most tasks. Figure 4.5 shows the proposed finetuning pipeline with the two aforementioned channel fusing approaches.

After fusion, we perform temporal average pooling:

$$\mathbf{h} = \frac{1}{N_p} \sum_{p=1}^{N_p} \mathbf{z}_p, \quad \mathbf{z}_p \in \begin{cases} \mathbb{R}^{d_e} & \text{(mean)} \\ \mathbb{R}^{Cd_e} & \text{(concat)} \end{cases} \tag{4.7}$$

and apply a linear layer to obtain logits.

This two-stage, channel then temporal pooling, avoids premature mixing that a single global average over all CN_p tokens would induce while retaining discriminative electrode structure prior to temporal aggregation.

4.1.4 Pretraining setup and reproducibility

Compute, runtime and reproducibility

All experiments were implemented in Python 3.10 using PyTorch Lightning and Hydra for modularity, configurability, and reproducibility. The proposed foundation model was trained on the CSCS Alps HPC infrastructure using NVIDIA GH200 GPUs, employing a single node with 4x NVIDIA GH200 GPUs in Distributed Data Parallel (DDP) mode for both pre-training and fine-tuning.

Pre-training took approximately 8 hours using around 500 GB of EMG data. In order to keep a modest parameter count, a single parameter configuration of 3.6M has been adopted. Fine-tuning each downstream task ranged from 30 minutes to 2 hours, depending on dataset size and DDP configuration.

Silent speech experiments used the original codebase of Gaddy & Klein [25]

without DDP, while the discrete gesture task used the original codebase with PyTorch Lightning framework [21].

4.2 Experiments

Table 4.2: Hyperparameters for downstream fine-tuning with EMG Transformer.

Hyperparameter	Value
Batch size	32
Peak / minimal learning rate	5×10^{-4} / 1×10^{-5}
Learning rate scheduler	Cosine
Optimizer (β_1, β_2)	AdamW $(0.9, 0.98)$
Weight decay	0.01
Total epochs	Early stopping (max 50)
Warm-up epochs	5
Drop-path	0.10
Layer-wise learning rate decay	0.90
Label smoothing (multi-class classification)	0.10

Each recording undergoes a denoising step with a specific band-pass filter followed by a 50 Hz notch. EMG signals with less than 16 channels are zero-padded and then resampled at 2 kHz. MinMax Channel-wise normalization followed by a shifting operation is applied to keep the signals in the range $[-1, 1]$. Each recording is then segmented into fixed-length windows of 1000 samples with 50% overlap, providing sufficient temporal context while controlling computational load. Prior work [61, 62] indicates that windows in the 150-500 ms range are a common sweet spot.

4.2.1 Datasets and tasks

Table 4.3: sEMG corpora used for pretraining

Dataset	Subjects	Records	Dur.(s)	f_s (Hz)	Channels	Size
Ninapro DB6	10	~ 8.4 k	4	2000	14	20.3 GB
Ninapro DB7	22	~ 5.4 k	5	2000	12	30.9 GB
EMG2Pose	192	25253	60	2000	16	431 GB

The previously described architecture is pre-trained for an FM specific for EMG signals using the most extensive open-access corpora currently available (see table 4.3).

4.2.2 Evaluation protocols

Table 4.4: Public datasets used for downstream evaluation.

Dataset	Subjects	f_s (Hz)	Channels	Task
Ninapro DB5	10	200	16	Hand gestures
EPN-612	612	200	8	Hand gestures
UCI EMG	36	200	8	Hand gestures
Discrete Gestures	100	2000	16	Hand gestures
Ninapro DB8	12	2000	16	Kinematic regression
Silent Speech	1	800	8	Silent speech recognition

The pretrained encoder is evaluated on the datasets listed in Table 4.4. All downstream experiments are performed with the same hyperparameters, as shown in Table 4.2. Each benchmark is then split by subject into training, validation, and test sets with a 60/20/20 partition, preventing subject leakage.

Three different settings are used for downstream tasks: (i) **Supervised** (scratch), where the model is trained from scratch on the task-specific dataset without relying on pre-trained weights; (ii) **Linear Probing**, where the encoder is frozen and only the task-specific head is trained; (iii) **Full Finetuning**, where the entire model is fine-tuned using layer-wise learning rate decay to avoid catastrophic forgetting [63].

A. Classification Classification tasks used CrossEntropy Loss, with Top1 Accuracy and F1 score as evaluation metrics to assess model performance and compare with state-of-the-art methods.

B. Discrete Gesture Meta - Discrete Gestures task uses Binary CrossEntropy Loss with Classification Error Rate (CLER) as evaluation metric. CLER is computed as the proportion of events detected by the model that were assigned the incorrect gesture, in a balanced average across all gestures.

C. Regression Regression task on Ninapro DB8 used L1 loss, with Mean Absolute Error reported in degrees across all Degrees of Articulation (DoAs) as evaluation metric. Additional regression metrics are available in Table B.1.

D. Silent Speech Silent speech tasks used Word Error Rate as evaluation metric, with two distinct tasks: voicing silent speech (audio, referred as D1) and silent speech recognition (text, referred as D2). The voicing task uses Dynamic Time Warping (DTW) Loss with weighted phoneme loss for aligning mel spectrograms, HiFi-GAN vocoder, and Wav2Vec2 ASR for transcription into text. The recognition task uses CTC-loss with a Language Model Beam Search decoding. The WER metric is used to evaluate the performance of the model on both tasks.

Chapter 5

Results

5.1 Pretraining Reconstruction

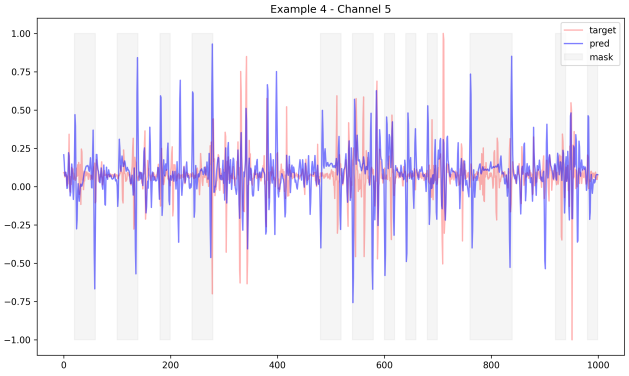

Figure 5.1: Reconstruction of a windowed EMG recording by the proposed EMG Foundation Model.

Figure 5.1 illustrates the reconstruction of a single windowed EMG recording produced by the proposed EMG Foundation Model. The model reliably recovers masked segments of the signal, although reconstruction quality degrades as the size of contiguous occlusions increases. Given the relatively compact parameter budget of 3.6M, the observed reconstruction fidelity is notable and supports the model's capacity to learn useful signal structure despite its modest size.

5.2 A. Gesture classification

Table 5.1: EMG gesture recognition. Accuracy and F1 scores.

Method	Params	Ninapro DB5		EPN-612		UCI EMG	
		Acc.	F1	Acc.	F1	Acc.	F1
Moment	385 M	86.41	74.42	90.87	90.16	90.45	91.75
OTiS	45 M	85.31	72.61	87.55	88.03	90.62	89.28
PhysioWave Small	5 M	84.78	72.54	93.12	93.40	90.35	89.51
PhysioWave Base	15 M	86.02	73.78	93.68	93.91	91.92	92.77
PhysioWave Large	37 M	**87.53**	75.42	94.50	94.56	93.19	93.59
WaveFormer	3.1 M	**87.53**	74.66	95.21	95.22	93.10	93.20
Supervised	3.6 M	78.59	78.06	95.84	95.85	97.50	**97.85**
Linear Probing	3.6 M	76.79	75.13	79.94	79.96	97.14	97.14
Finetuning	3.6 M	85.30	**84.53**	**96.60**	**96.69**	**97.86**	97.50

Table 5.1 summarizes the performance of the proposed model against the publicy available large-scale *generic* time-series models, namely *Moment* (385 M) and *OTiS* (45 M), as well as the more recent *PhysioWave* family of models (5 M, 15 M, and 37 M) and *WaveFormer* (3.1 M).

The proposed Foundation Model for EMG signals even with a modest parameter count of 3.6 M outperforms the larger models on the EPN-612 and UCI EMG datasets while achieving competitive results on the Ninapro DB5 dataset. On EPN-612, the model achieves new state-of-the-art results of 96.60% accuracy and 96.69% F1 score, surpassing both *PhysioWave Large* and *WaveFormer*. On UCI EMG, the model achieves 97.86% accuracy and 97.50% F1 score, outperforming all other models with a +4% improvement over *PhysioWave Large*. On Ninapro DB5, the model achieves 85.30% accuracy, which is slightly lower than *PhysioWave Large*, but a 84.53% F1 score, achieving new state-of-the-art results using 1/10 of the parameters.

5.3 B. Discrete Gesture

Table 5.2: EMG Discrete Gesture. Classification Error Rate (CLER) for full sequence and windowed inference.

Method	Params	CLER	Inference Method
Meta - LSTM	6.4 M	0.1819	Full sequence
Meta - LSTM	6.4 M	0.1596	Windowed
Supervised	3.6 M	0.1594	Windowed
Finetuning	3.6 M	0.1553	Windowed

Table 5.2 summarizes the performance of the proposed model on the discrete gesture recognition task, specifically focusing on Classification Error Rate (CLER) for both full sequence and windowed inference methods.

The original work by Meta adopts a stacked LSTM architecture with 3 layers, trained on non-overlapping windows of 8 seconds. The proposed model is trained under the same conditions, allowing for a fair comparison between the two approaches. However, at inference time, the LSTM was tested on the full sequence of EMG data, while the proposed Transformer model cannot process the entire sequence due to quadratic complexity, thus requiring windowed inference.

The windowed inference approach involves sliding a window of 8 seconds over the EMG data, with a stride of 2 seconds, to ensure that the model can process the data in manageable chunks. This method allows the Transformer model to maintain performance while being coherent to its architectural constraints. In order to compare the original LSTM architecture with the proposed Transformer model, the same windowed inference strategy is applied to the LSTM during evaluation, and both the results are reported.

The results indicate that the Transformer model achieves comparable performance to the original LSTM architecture (tested under windowed inference) with a CLER of 0.1553, which is slightly better than the LSTM's CLER of

0.1596. This demonstrates that the proposed Transformer model, although with modest parameter count, can effectively handle the discrete gesture recognition task while adhering to its architectural constraints and even outperforming the original LSTM while leveraging pre-trained weights.

5.4 C. Regression

Table 5.3: EMG regression on Ninapro DB8. Mean Absolute Error in Degrees across DoAs.

Method	Params	MAE°	Notes
TEMPONet TCN	<500 K	6.89	All 5 DoAs, per-subject
Event-based Linear Regr.	—	8.8 ± 2.3	All 5 DoAs, per-subject
DeepNet+Kalman	—	13.5 (RMSE)	3 DoAs only, per-subject
Supervised	3.6 M	8.87	All 5 DoAs, across-subject
Linear Probing	3.6 M	9.48	All 5 DoAs, across-subject
Finetuning	3.6 M	8.53	All 5 DoAs, across-subject

Table 5.3 summarizes the performance of the proposed model on the Ninapro DB8 dataset for kinematic regression, which involves predicting joint angles across five degrees of freedom (DoAs). The results are reported in terms of Mean Absolute Error (MAE) in degrees.

The proposed model demonstrates competitive performance compared to existing methods, achieving a MAE of 8.87° across all five DoAs in a cross-subject setting. Notably, the finetuning approach yields the best performance with a MAE of 8.53°, showcasing the effectiveness of the proposed model in capturing the underlying patterns in the EMG signals for regression tasks. Compared to TEMPONet TCN, which is a *per-subject* setting with a MAE of 6.89°, the proposed model achieves an higher MAE of 8.53° but in a *cross-subject* setting, indicating the model's ability to generalize across different

subjects with remarkable performance.

5.5 D. Silent Speech

5.5.1 D1. Silent Speech Synthesis

Silent speech synthesis is the process of converting EMG signals into audible speech. This involves a model that interprets the muscle activity and generates corresponding audio. Table 5.4 presents the Word Error Rate for various methods in the EMG-to-audio domain, indicating how accurately the synthesized speech can be transcribed back into text.

Table 5.4: EMG Silent Speech Synthesis. Word Error Rate (WER) reported for the EMG–audio modality.

Method	Params	EMG–audio (WER)
Gaddy & Klein	54 M	36.00%
Ren et al., 2024	—	32.00%
Scheck & Schultz, 2023	—	40.00%
EMGVox-GAN	12 M	36.00%
Supervised	4.5 M	34.14%
Finetuning	4.5 M	31.65%

As the table 5.4 illustrates, different models and techniques yield varying levels of accuracy. For instance, the Finetuning method with 4.5 million parameters achieves a relatively low WER of 31.65%, suggesting a higher accuracy in synthesizing intelligible speech from EMG signals.

5.5.2 D2. Silent Speech Recognition

Silent speech recognition, on the other hand, focuses on directly converting EMG signals into written text, bypassing the audio generation step. This modality is particularly useful for silent dictation or for individuals who have lost their voice but retain control over their facial muscles. Table 5.5 shows the WER for different EMG-to-text methods.

Table 5.5: EMG Silent Speech Recognition. Word Error Rate (WER) reported for the EMG–text modality.

Method	**Params**	**EMG–text (WER)**
Gaddy & Klein	54 M	28.8%
Stanford MONA	—	22.2%
Stanford MONA LISA	—	12.2%
Supervised	4.5 M	33.90%
Finetuning	4.5 M	32.75%

The Stanford MONA LISA model achieves a remarkable Word Error Rate (WER) of just 12.2% in this domain. This high accuracy in directly transcribing silent speech from muscle signals to text is achieved through a multimodal approach that learns from joint EMG-audio pairs. This result underscores the potential of EMG-based interfaces as a robust communication alternative and highlights the significant advantages of multimodal learning.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this work, we addressed the critical challenge of generalization in Electromyography-based Human-Machine Interfaces. Traditional models often require extensive, task-specific labeled data and struggle to adapt across different users, sessions, and tasks. We successfully demonstrated that the Foundation Model paradigm, powered by SSL, offers a potent solution. We introduced and validated a compact, EMG-centric Foundation Model, proving that a single, pre-trained encoder can learn rich, transferable representations of neuromuscular activity from large-scale unlabeled data.

The effectiveness of our approach is reflected in the robust empirical performance of the 3.6 M parameter model across multiple downstream tasks (see Chapter 5): it achieves superior gesture-classification results relative to much larger baselines (e.g., PhysioWave Large, 37M params; Moment, 385M params), posts a competitive Mean Absolute Error of 8.53° on Ninapro DB8

for cross-subject kinematic regression, outperforms traditional LSTM architectures on discrete gesture decoding, and attains strong silent-speech recognition from purely unimodal EMG pretraining. For additional context, recent work on WaveFormer (a lightweight sEMG transformer of ≈ 3.1M parameters) reports $\approx 95\%$ classification accuracy on EPN612; our model delivers broader downstream transfer performance while maintaining a similar compact footprint.

Crucially, the significance of these results is twofold. First, they validate that a single, unified model can effectively replace task-specific engineering for a wide array of EMG applications. Second, the **modest parameter count** is not merely an academic footnote but a central feature of our contribution: it directly enables deployment on resource-constrained embedded platforms. Concretely, our transformer has 3.6 million parameters, which corresponds to ≈ 13.73 MiB in FP32, ≈ 3.43 MiB in INT8, and ≈ 1.72 MiB when packed as INT4.

By comparison, GreenWaves GAP9 provides on the order of ≈ 1.5 MiB of L2 SRAM and 128 KiB of L1/TCDM per cluster (plus ≈ 2 MiB of on-package nonvolatile memory/eMRAM and the option to attach external PSRAM). These numbers imply that, without further compression or system-level strategies, the 3.6M-parameter model cannot fit entirely in GAP9's L2 in FP32 or INT8 form; even INT4 quantization is only marginally above the L2 budget. Therefore, practical deployment on GAP9 will require as first a quantization step, pruning or other compression techniques and possibly tiling or streaming so weights are loaded into L1/L2 on demand; each of these steps trades implementation complexity, latency, and energy for reduced on-chip memory pressure.

6.2 Future Work

This thesis establishes a strong foundation for future research in EMG-based interfaces. Several promising avenues can be explored to build upon this work:

- **Architectural Enhancements**: While our Transformer-based encoder is highly effective, future work could explore even more computationally efficient architectures. Investigating hybrid models that combine the global receptive field of Transformers with the efficiency of Mamba-based State Space Models or incorporating Mixture-of-Experts (MoE) layers could further scale up performance.

- **Expanding the Pre-training Corpus**: The strength of a Foundation Model is directly tied to the scale and diversity of its pre-training data. Future iterations should aim to incorporate even larger and more heterogeneous EMG datasets, covering a wider range of subjects, pathologies, and acquisition hardware to further bolster the model's robustness and zero-shot capabilities.

- **Quantization and On-Device Optimization**: To fully realize the potential for embedded deployment, a systematic study on model quantization is necessary. Exploring the trade-offs of 8-bit and 4-bit quantization on performance would be a critical next step for deploying this model on low-power microcontrollers and edge AI accelerators, such as GreenWaves GAP processors.

- **Multi-Modal Foundation Models**: While our unimodal approach was highly successful, future research could explore pre-training multi-modal Foundation Models that learn to fuse EMG signals with other biosignals, such as EEG or inertial measurement unit (IMU) data, to decode user intent with even greater accuracy and reliability.

Appendix A

Datasets

A.1 Pretraining Datasets

For the pre-training of the FM, the raw EMG data are processed with several critical preprocessing steps, including band-pass filtering, notch filtering, normalization, and segmentation into overlapping windows.

First, a bandpass filter is applied to remove unwanted noise and artifacts, typically in the range of 20-450 Hz, which is suitable for capturing the relevant frequency components of EMG signals. A notch filter at 50 Hz is also applied to eliminate power line interference. Normalization is then applied via min-max scaling to ensure that the range of the EMG signals falls within [-1 , 1].

Segmentation into overlapping windows is performed to create a dataset suitable for training. Each window is typically 1000 samples long with a step size of 500. If the sampling rate of the raw EMG signal is below 2000 Hz, the data is up-sampled to 2000 Hz to ensure consistency across the dataset. Furthermore, if the number of channels is less than 16, zero-padding is applied to

ensure that all samples have the same number of channels. Such pre-processed data is then stored in HDF5 format, for efficient storage and retrieval during training.

Ninapro DB6 Ninapro DB6 contains sEMG from 10 subjects performing 7 hand grasp types, each repeated 12 times over 5 separate recording days. Signals were acquired with 14 wireless Trigno electrodes with a sampling rate of 2 kHz and time□aligned with inertial measurements capturing forearm motion. The dataset targets robust grasp recognition under temporal and session variability, supporting evaluation of prosthetic control algorithms [64].

Ninapro DB7 Ninapro DB7 provides simultaneous myoelectric and inertial data from 20 intact subjects and 2 transradial amputees. Recordings employ 12 wireless Trigno EMG sensors plus co□located 9□axis IMUs (2 kHz) alongside an 18□DOF CyberGlove on the contralateral hand for kinematic ground truth. Subjects executed 40 movements spanning isolated finger/wrist actions and grasp patterns, enabling multimodal fusion studies for prosthetic intent decoding [65].

EMG2Pose EMG2Pose pairs 16 sEMG channels acquired at 2 kHz with synchronized joint angle trajectories for hand motion across 29 staged activities. It comprises 25,253 HDF5 sessions from 193 participants (some held out for generalization splits), up to a total of 370 hours. Rich metadata (subject ID, session, laterality, split flags) facilitates standardized benchmarking of EMG□driven pose estimation, gesture recognition, and cross□subject transfer [66].

A.2 Downstream Datasets

Ninapro DB5 Ninapro DB5 comprises 10 intact subjects with 16 forearm electrodes at 200 Hz of repeated executions of a set of 52 wrist/hand gestures. Each movement is performed in multiple repetitions under controlled timing, providing a medium-scale benchmark for gesture classification and cross-subject generalization with moderate channel count and relatively low sampling rate [18].

EPN-612 EPN-612 contains 8-channel, 200 Hz Myo armband recordings from 612 subjects performing five active gestures (wave-in, wave-out, pinch, open, fist) plus rest, typically 50 trials per class. Its large subject pool emphasizes inter-person variability and supports evaluation of robustness, calibration reduction, and domain adaptation methods [19].

UCI EMG The UCI EMG Gesture dataset (36 subjects, 8 channels at 200 Hz) captures multiple hand/wrist gesture classes (commonly 8–10 plus rest in derived splits) with labeled repetitions. Its compact size and consistent sensor layout make it a lightweight benchmark for rapid prototyping and ablation of preprocessing or model components [20].

Discrete Gestures (Meta's Generic Neuromotor Interface) Meta (Reality Labs) presents a high-fidelity, non-invasive neuromotor interface in their 2025 Nature article, "A generic non-invasive neuromotor interface for human-computer interaction", introducing the *discrete gestures* dataset. This dataset comprises segmented forearm and wrist gestures collected from 100 participants, recorded via surface electromyography (sEMG) using 16 high-rate channels sampled at 2 kHz, designed for low-latency interaction studies [21].

Ninapro DB8 Ninapro DB8 (12 subjects, 16 channels at 2 kHz) pairs sEMG with synchronized multi-DoF finger and wrist kinematics (glove / motion

capture) over continuous movement protocols. It serves kinematic (angle) regression and proportional control tasks, stressing fine temporal alignment, amplitude scaling, and inter□joint coordination modeling [67].

Silent Speech In their EMNLP 2020 paper "Digital Voicing of Silent Speech," Gaddy and Klein introduce a silent speech dataset comprising nearly 20 hours of facial sEMG signals from a single speaker, recorded via eight channels at a 800 Hz sampling rate. The dataset includes parallel silent and vocalized utterances with time-aligned transcriptions, enabling the transfer of audio targets to silent EMG via dynamic time warping and feature alignment [25]. This facilitates the evaluation of EMG-to-text and EMG-to-speech models, supporting tasks such as mapping muscle activity to phoneme sequences and acoustics under limited subject diversity but rich sentence-level variability. Follow-up improvements using convolutional plus Transformer models further integrated phoneme prediction as an auxiliary task to enhance intelligibility in open-vocabulary settings [68].

Appendix B

More Evaluation Results

B.1 Kinematic Regression

Table B.1: Complete EMG Kinematic Regression Results across-subject. Additional metrics are reported: Pearson correlation, R^2, RMSE, and Explained Variance.

Method	MAE°	Pearson	R^2	RMSE	Explained Variance
Supervised	8.87	0.7731	0.5958	13.81	0.596
Linear Probing	9.48	0.7463	0.5589	14.49	0.5598
Finetuning	**8.53**	**0.7918**	**0.627**	**13.31**	**0.6281**

B.2 Silent Speech

The Silent Speech task is also evaluated comparing the frequency of errors between two phonemes to the frequency of correct predictions on those phonemes.

Confusion is defined as follows:

$$(e_{p1,p2} + e_{p2,p1})/(f_{p1} + f_{p2}) \tag{B.1}$$

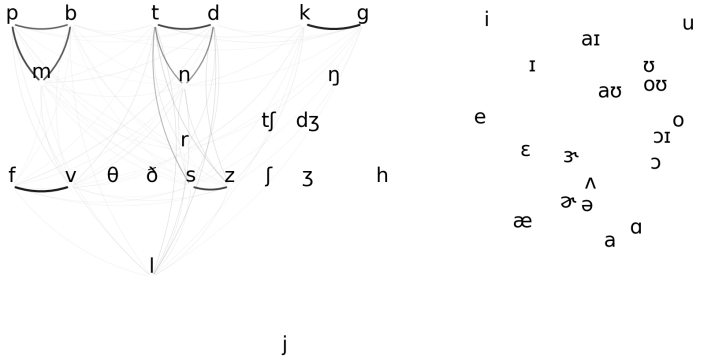

Figure B.1: Phoneme confusability (darker lines indicate more confusion - maximum darkness is 13% confusion)

while Accuracy is defined as:

$$(e_{p1,p1} + e_{p2,p2})/(f_{p1} + f_{p2}) \tag{B.2}$$

where $e_{p1,p2}$ is the number of times $p2$ was predicted when the label was $p1$, and f_{p1} is the number of times phoneme $p1$ appears as a target label.

B.3 Discrete Gesture

Table B.3 shows an ablation study of window size on the discrete gesture recognition task, using CLER as the performance metric. This analysis explores the impact of window size on gesture recognition accuracy, relevant for real-time applications. The results indicate that a window size of 16,000 samples yields the best performance for both the Meta-LSTM and finetuned models. Performance degrades as the window size decreases, particularly with a significant drop at 4,000 samples (2 seconds at 2kHz sampling rate), suggesting that smaller windows may not capture sufficient temporal information for accurate gesture recognition.

Table B.2: Results for the most confused pairs of phonemes

IPA Phonemes		Confusion (%)	Accuracy (%)
v	f	13.5	74.1
k	g	13.0	73.3
z	s	10.9	77.1
t	d	10.6	62.8
p	m	10.5	76.5
m	b	9.5	74.2
p	b	8.9	69.7
ʃ	dʒ	8.4	63.4
r	ɜˈ	7.5	77.0
dʒ	tʃ	7.0	56.7
ɛ	ae	6.6	70.3
n	d	6.4	64.9
ɪ	ʊ	6.1	67.9
t	n	6.1	65.6
ɪ	ɛ	5.7	63.0
ʃ	tʃ	5.5	60.1
uː	oʊ	5.4	77.5
j	g	4.9	50.1
θ	ð	4.5	80.1
iː	eɪ	4.4	82.5

Table B.3: Ablation study on the window size for the discrete gesture recognition task, measured by CLER.

Method	Window size	CLER	Inference Method
Meta - LSTM	—	0.1819	Full sequence
Meta - LSTM	16 000	0.1596	Windowed
Finetuning	16 000	0.1553	Windowed
Finetuning	8 000	0.1634	Windowed
Finetuning	6 000	0.1651	Windowed
Finetuning	4 000	0.2770	Windowed

B.4 Visualization of EPN612 Experimental Results

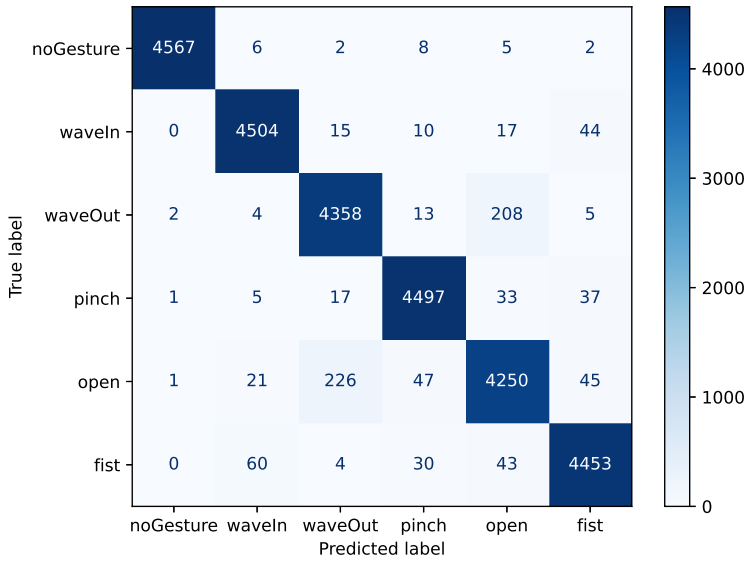

Figure B.2: Confusion matrix on the EPN612 test set.

Figure B.2 shows the confusion matrix for the EPN612 dataset. Performance is generally high, with most gestures being correctly classified. Most misclassifications occur between *waveOut* and *open*. Given the sparse distribution of off-diagonal values, the model demonstrates strong generalization capabilities across the different gestures.

Figure B.3 shows the ROC curves for each gesture in the EPN612 test set. The *noGesture* class achieves the highest AUC of 1.0, indicating perfect classification for this class, followed by the *waveIn, pinch, and fist*, all with AUCs of 0.999. All the curves are located well above the diagonal of the random classifier, indicating strong separation between positive and negative classes across all gestures.

Figure B.4 shows the t-SNE embeddings of the EPN612 dataset. The embeddings are well-separated for most gestures, with a slight overlap between *waveOut* and *open*, indicating some confusion between these two classes

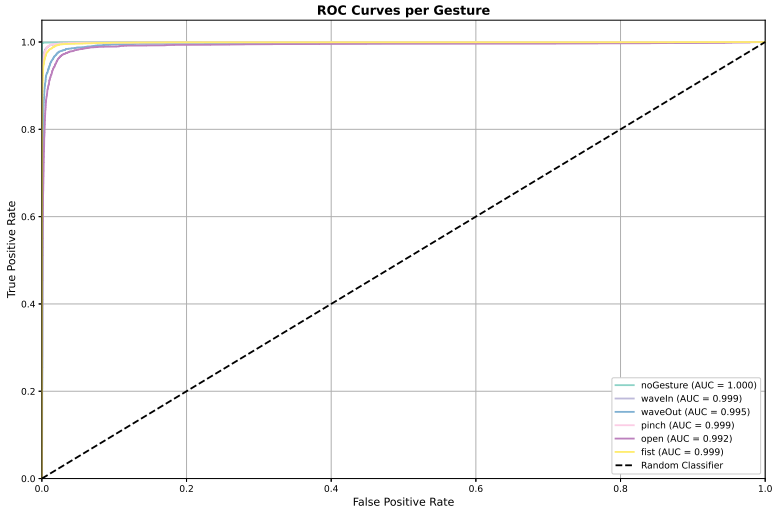

Figure B.3: ROC curves for each gesture in the EPN612 test set.

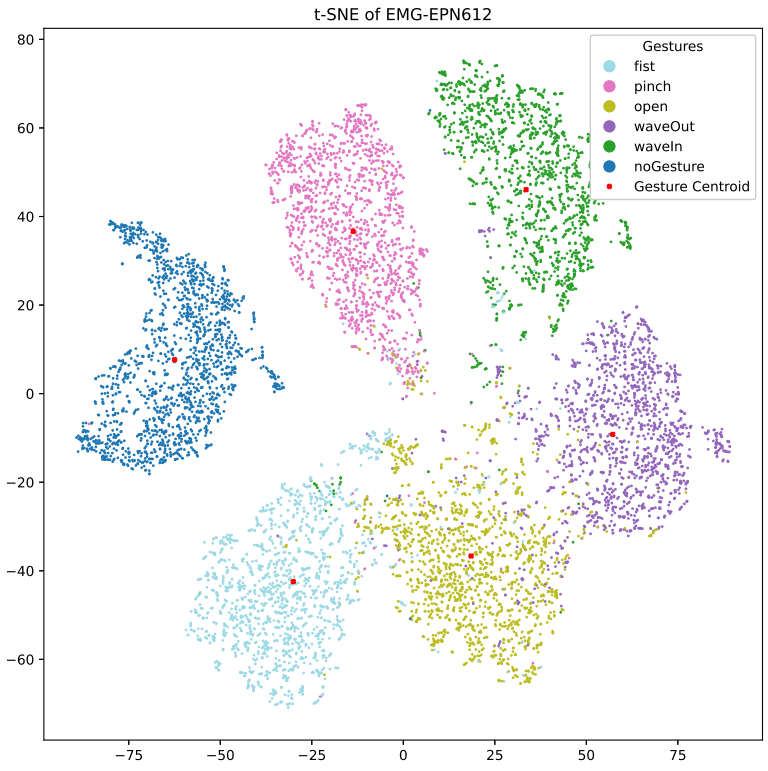

Figure B.4: t-SNE embeddings for the EPN612 dataset.

which is also reflected in the confusion matrix. The *noGesture* class is clearly separated from the others, demonstrating that the model effectively learns to distinguish between active gestures and rest, and the overall visualization reveals non-linear clustering structures.

Appendix C

FLOPs and Peak Memory Usage

Table C.1: Per-GPU FLOPs and peak memory usage during pretraining and finetuning.

Model	FLOPs	Peak memory usage
Pretraining	9.6 G	7,778 MB
Finetuning (DB5)	9.6 G	1,968 MB
Finetuning (EPN-612)	3.8 G	1,060 MB

Table C.1 shows the FLOPs and peak memory usage during pretraining and finetuning on Ninapro DB5 and EPN-612. FLOPs were estimated per GPU using lightning `measure_flops` on the per-GPU training batch (only forward pass); reported FLOPs are per training step per GPU. Peak GPU memory is the maximum per-GPU allocation measured with `torch.cuda.max_memory_allocated()` on rank 0 GPU.

For distributed data-parallel (DDP) training each GPU holds a full model replica and optimizer/activation memory; the aggregate cluster FLOPs can be computed by multiplying the per-GPU GFLOPS by the number of GPUs but communication may reduce observed throughput.

Bibliography

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023. [Online]. Available: https://arxiv.org/abs/1706.03762

[2] R. Merletti and G. Cerone, "Tutorial. surface emg detection, conditioning and pre-processing: Best practices," *Journal of Electromyography and Kinesiology*, vol. 54, p. 102440, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1050641120300821

[3] M. Boyer, L. Bouyer, J.-S. Roy, and A. Campeau-Lecours, "Reducing noise, artifacts and interference in single-channel emg signals: A review," *Sensors*, vol. 23, no. 6, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/6/2927

[4] U. Côté-Allard, E. Campbell, A. Phinyomark, F. Laviolette, B. Gosselin, and E. Scheme, "Interpreting deep learning features for myoelectric control: A comparison with handcrafted features," *Frontiers in Bioengineering and Biotechnology*, vol. 8, Mar. 2020. [Online]. Available: http://dx.doi.org/10.3389/fbioe.2020.00158

[5] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson,

S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. Ré, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang, "On the opportunities and risks of foundation models," 2022. [Online]. Available: https://arxiv.org/abs/2108.07258

[6] S. Abbaspourazad, O. Elachqar, A. C. Miller, S. Emrani, U. Nallasamy, and I. Shapiro, "Large-scale training of foundation models for wearable biosignals," in *ICLR*, 2024. [Online]. Available: https://arxiv.org/abs/2312.05409

[7] G. Narayanswamy, X. Liu, K. Ayush, Y. Yang, X. Xu, S. Liao, J. Garrison, S. Tailor, J. Sunshine, Y. Liu, T. Althoff, S. Narayanan, P. Kohli, J. Zhan, M. Malhotra, S. Patel, S. Abdel-Ghaffar, and D. McDuff, "Scaling wearable foundation models," 2024. [Online]. Available: https://arxiv.org/abs/2410.13638

[8] A. Dimofte, G. A. Bucagu, T. M. Ingolfsson, X. Wang, A. Cossettini, L. Benini, and Y. Li, "Cerebro: Compact encoder for representations of brain oscillations using efficient alternating attention," 2025. [Online]. Available: https://arxiv.org/abs/2501.10885

[9] K. McKeen, S. Masood, A. Toma, B. Rubin, and B. Wang, "Ecg-fm: An open electrocardiogram foundation model," 2025. [Online]. Available: https://arxiv.org/abs/2408.05178

[10] Y. Na, M. Park, Y. Tae, and S. Joo, "Guiding masked representation learning to capture spatio-temporal relationship of electrocardiogram," 2024. [Online]. Available: https://arxiv.org/abs/2402.09450

[11] H.-Y. S. Chien, H. Goh, C. M. Sandino, and J. Y. Cheng, "Maeeg: Masked auto-encoder for eeg representation learning," 2022. [Online]. Available: https://arxiv.org/abs/2211.02625

[12] R. Li, Y. Wang, W.-L. Zheng, and B.-L. Lu, "A multi-view spectral-spatial-temporal masked autoencoder for decoding emotions with self-supervised learning," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 6–14. [Online]. Available: https://doi.org/10.1145/3503161.3548243

[13] F. Douglas, M. Pei, and C. Kuo, "Characterizing the effect of electrode shift & sensor reapplication on common semg features in lower limb muscles," 2024. [Online]. Available: https://arxiv.org/abs/2410.16262

[14] K. Scheck, Z. Ren, T. Dombeck, J. Sonnert, S. van Gogh, Q. Hou, M. Wand, and T. Schultz, "Cross-speaker training and adaptation for

electromyography-to-speech conversion," in *2024 46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2024, pp. 1–4.

[15] M. Goswami, K. Szafer, A. Choudhry, Y. Cai, S. Li, and A. Dubrawski, "Moment: A family of open time-series foundation models," 2024. [Online]. Available: https://arxiv.org/abs/2402.03885

[16] Özgün Turgut, P. Müller, M. J. Menten, and D. Rueckert, "Towards generalisable time series understanding across domains," 2025. [Online]. Available: https://arxiv.org/abs/2410.07299

[17] Y. Chen, M. Orlandi, P. M. Rapa, S. Benatti, L. Benini, and Y. Li, "PhysioWave: A Multi-Scale Wavelet-Transformer for Physiological Signal Representation," Jun. 2025, arXiv:2506.10351 [cs]. [Online]. Available: http://arxiv.org/abs/2506.10351

[18] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori, "Comparison of six electromyography acquisition setups on hand movement classification tasks," *PLOS ONE*, vol. 12, no. 10, pp. 1–17, 10 2017. [Online]. Available: https://doi.org/10.1371/journal.pone.0186132

[19] M. E. Benalcazar, L. Barona, L. Valdivieso, X. Aguas, and J. Zea, "EMG-EPN-612 Dataset," Nov. 2020. [Online]. Available: https://zenodo.org/records/4421500

[20] N. Krilova, I. Kastalskiy, V. Kazantsev, V. Makarov, and S. Lobov, "EMG Data for Gestures," UCI Machine Learning Repository, 2018, DOI: https://doi.org/10.24432/C5ZP5C.

[21] P. Kaifosh, T. R. Reardon, and C. labs at Reality Labs, "A generic non-invasive neuromotor interface for human-computer interaction,"

Nature, 2025. [Online]. Available: https://www.nature.com/articles/s41586-025-09255-w

[22] M. Zanghieri, S. Benatti, A. Burrello, V. Kartsch, F. Conti, and L. Benini, "Robust real-time embedded emg recognition framework using temporal convolutional networks on a multicore iot processor," *IEEE transactions on biomedical circuits and systems*, vol. PP, 12 2019.

[23] M. Zanghieri, S. Benatti, L. Benini, and E. Donati, "Event-based low-power and low-latency regression method for hand kinematics from surface emg," in *2023 9th International Workshop on Advances in Sensors and Interfaces (IWASI)*, 2023, pp. 293–298.

[24] T. Bao, S. Zaidi, S. Xie, P. Yang, Y. Zhao, and Z. Zhang, "A deep kalman filter network for hand kinematics estimation using semg," *Pattern Recognition Letters*, vol. 143, 01 2021.

[25] D. Gaddy and D. Klein, "Digital voicing of silent speech," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 5521–5530. [Online]. Available: https://aclanthology.org/2020.emnlp-main.445/

[26] T. Benster, G. Wilson, R. Elisha, F. R. Willett, and S. Druckmann, "A Cross-Modal Approach to Silent Speech with LLM-Enhanced Recognition," Mar. 2024, arXiv:2403.05583 [cs]. [Online]. Available: http://arxiv.org/abs/2403.05583

[27] Y. Dong, J.-B. Cordonnier, and A. Loukas, "Attention is not all you need: Pure attention loses rank doubly exponentially with depth," 2023. [Online]. Available: https://arxiv.org/abs/2103.03404

[28] R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T.-Y. Liu, "On layer normalization in the transformer architecture," 2020. [Online]. Available: https://arxiv.org/abs/2002.04745

[29] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," 2023. [Online]. Available: https://arxiv.org/abs/2104.09864

[30] G. Wang, W. Liu, Y. He, C. Xu, L. Ma, and H. Li, "EEGPT: Pretrained transformer for universal and reliable representation of EEG signals," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=lvS2b8CjG5

[31] Z. Song, Q. Lu, H. Xu, H. Zhu, D. Buckeridge, and Y. Li, "Timelygpt: Extrapolatable transformer pre-training for long-term time-series forecasting in healthcare," in *Proceedings of the 15th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, ser. BCB '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3698587.3701364

[32] Z. Zheng, Y. Liang, R. Lyu, J. Bao, Y. Huang, A. Zhou, H. Ma, J. Wang, X. Meng, C. Shao, Y. Tang, and Q. Zhang, "Bp3: Improving cuff-less blood pressure monitoring performance by fusing mmwave pulse wave sensing and physiological factors," in *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 730–743. [Online]. Available: https://doi.org/10.1145/3666025.3699370

[33] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," 2018. [Online]. Available: https://arxiv.org/abs/1803.02155

[34] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," 2023. [Online]. Available: https://arxiv.org/abs/1910.10683

[35] O. Press, N. A. Smith, and M. Lewis, "Train short, test long: Attention with linear biases enables input length extrapolation," 2022. [Online]. Available: https://arxiv.org/abs/2108.12409

[36] M. J. Islam, U. Rumman, A. Ferdousi, M. S. Pervez, I. Ara, S. Ahmad, F. Haque, S. Hamid, M. Ali, K. S. Zaman, M. B. I. Reaz, M. H. Chowdhury, and M. R. Islam, "Impact of electrode position on forearm orientation invariant hand gesture recognition," 2024. [Online]. Available: https://arxiv.org/abs/2410.00029

[37] D. C. Toledo-Pérez, J. Rodríguez-Reséndiz, R. A. Gómez-Loenzo, and J. C. Jauregui-Correa, "Support vector machine-based emg signal classification techniques: A review," *Applied Sciences*, vol. 9, no. 20, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/20/4402

[38] P. Gopal, A. Gesta, and A. Mohebbi, "A systematic study on electromyography-based hand gesture recognition for assistive robots using deep learning and machine learning models," *Sensors*, vol. 22, no. 10, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/10/3650

[39] U. Côté-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. 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.

[40] T. Bao, S. A. R. Zaidi, S. Xie, P. Yang, and Z.-Q. Zhang, "A cnn-lstm hybrid model for wrist kinematics estimation using surface electromyography," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, p. 1–9, 2021. [Online]. Available: http://dx.doi.org/10.1109/TIM.2020.3036654

[41] M. Jabbari, R. N. Khushaba, and K. Nazarpour, "Emg-based hand gesture classification with long short-term memory deep recurrent neural networks," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020, pp. 3302–3305.

[42] M. Simão, P. Neto, and O. Gibaru, "Emg-based online classification of gestures with recurrent neural networks," *Pattern Recognition Letters*, vol. 128, pp. 45–51, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167865519302089

[43] A. Samadani, "Gated recurrent neural networks for emg-based hand gesture classification. a comparative study," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2018, pp. 1–4.

[44] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *ACM Comput. Surv.*, vol. 55, no. 6, Dec. 2022. [Online]. Available: https://doi.org/10.1145/3530811

[45] B. Zhao, H. Xing, X. Wang, F. Song, and Z. Xiao, "Rethinking attention mechanism in time series classification," *Information Sciences*, vol. 627, pp. 97–114, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025523000968

[46] F. Moreno-Pino, Álvaro Arroyo, H. Waldon, X. Dong, and Álvaro Cartea, "Rough transformers: Lightweight and continuous time series modelling through signature patching," 2025. [Online]. Available: https://arxiv.org/abs/2405.20799

[47] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2202.07125

[48] E. Sason, D. Frolova, B. Nazarov, and F. Goldberd, "Attention condensation via sparsity induced regularized training," 2025. [Online]. Available: https://arxiv.org/abs/2503.01564

[49] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020. [Online]. Available: https://arxiv.org/abs/2006.04768

[50] Y.-H. Wu, Y. Liu, X. Zhan, and M.-M. Cheng, "P2t: Pyramid pooling transformer for scene understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, p. 12760–12771, Nov. 2023. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2022.3202765

[51] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words:

Transformers for image recognition at scale," 2021. [Online]. Available: https://arxiv.org/abs/2010.11929

[52] C. Yang, M. B. Westover, and J. Sun, "Biot: Cross-data biosignal learning in the wild," 2023. [Online]. Available: https://arxiv.org/abs/2305.10351

[53] Y. Bao, S. Sivanandan, and T. Karaletsos, "Channel vision transformers: An image is worth 1 x 16 x 16 words," 2024. [Online]. Available: https://arxiv.org/abs/2309.16108

[54] S. Abbaspourazad, O. Elachqar, A. C. Miller, S. Emrani, U. Nallasamy, and I. Shapiro, "Large-scale training of foundation models for wearable biosignals," 2024. [Online]. Available: https://arxiv.org/abs/2312.05409

[55] C. Wang, V. Subramaniam, A. U. Yaari, G. Kreiman, B. Katz, I. Cases, and A. Barbu, "Brainbert: Self-supervised representation learning for intracranial recordings," 2023. [Online]. Available: https://arxiv.org/abs/2302.14367

[56] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked Autoencoders Are Scalable Vision Learners," Dec. 2021, arXiv:2111.06377 [cs]. [Online]. Available: http://arxiv.org/abs/2111.06377

[57] W.-B. Jiang, L.-M. Zhao, and B.-L. Lu, "Large brain model for learning generic representations with tremendous eeg data in bci," 2024. [Online]. Available: https://arxiv.org/abs/2405.18765

[58] W. Cui, W. Jeong, P. Thölke, T. Medani, K. Jerbi, A. A. Joshi, and R. M. Leahy, "Neuro-gpt: Towards a foundation model for eeg," 2024. [Online]. Available: https://arxiv.org/abs/2311.03764

[59] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," 2023. [Online]. Available: https://arxiv.org/abs/2211.14730

[60] A. Tegon, T. M. Ingolfsson, X. Wang, L. Benini, and Y. Li, "Femba: Efficient and scalable eeg analysis with a bidirectional mamba foundation model," 2025. [Online]. Available: https://arxiv.org/abs/2502.06438

[61] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the optimal window length for pattern recognition-based myoelectric control: Balancing the competing effects of classification error and controller delay," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 2, pp. 186–192, 2011.

[62] C. De la Fuente, E. Martinez-Valdes, J. I. Priego-Quesada, A. Weinstein, O. Valencia, M. R. Kunzler, J. Alvarez-Ruf, and F. P. Carpes, "Understanding the effect of window length and overlap for assessing semg in dynamic fatiguing contractions: A non-linear dimensionality reduction and clustering," *Journal of Biomechanics*, vol. 125, p. 110598, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021929021003754

[63] P. Kenneweg, A. Schulz, S. Schröder, and B. Hammer, "Intelligent learning rate distribution to reduce catastrophic forgetting in transformers," in *Intelligent Data Engineering and Automated Learning – IDEAL 2022: 23rd International Conference, IDEAL 2022, Manchester, UK, November 24–26, 2022, Proceedings*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 252–261. [Online]. Available: https://doi.org/10.1007/978-3-031-21753-1_25

[64] F. Palermo, M. Cognolato, A. Gijsberts, H. Müller, B. Caputo, and M. Atzori, "Repeatability of grasp recognition for robotic hand prosthesis control based on semg data," in *2017 International Conference on Rehabilitation Robotics (ICORR)*, 2017, pp. 1154–1159.

[65] A. Krasoulis, I. Kyranou, M. Erden *et al.*, "Improved prosthetic hand control with concurrent use of myoelectric and inertial measurements," *Journal of NeuroEngineering and Rehabilitation*, vol. 14, p. 71, 2017.

[66] S. Salter, R. Warren, C. Schlager, A. Spurr, S. Han, R. Bhasin, Y. Cai, P. Walkington, A. Bolarinwa, R. Wang, N. Danielson, J. Merel, E. Pnevmatikakis, and J. Marshall, "emg2pose: A Large and Diverse Benchmark for Surface Electromyographic Hand Pose Estimation," Dec. 2024, arXiv:2412.02725 [cs]. [Online]. Available: http://arxiv.org/abs/2412.02725

[67] A. Krasoulis, S. Vijayakumar, and K. Nazarpour, "Effect of user practice on prosthetic finger control with an intuitive myoelectric decoder," p. 891, 2019.

[68] D. Gaddy and D. Klein, "An improved model for voicing silent speech," 2021. [Online]. Available: https://arxiv.org/abs/2106.01933