Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

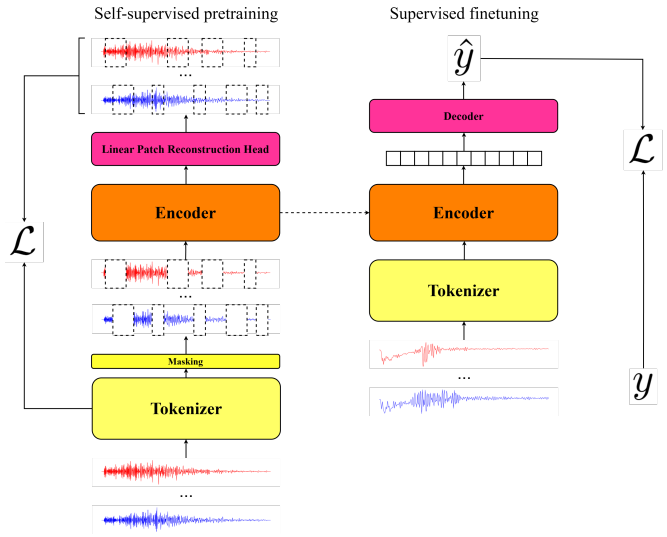
Institut für Integrierte Systeme
Integrated Systems Laboratory

Department of Information Technology and
Electrical Engineering

Spring Semester 2025

Developing Foundation Models for raw Ultrasound Data

Master Thesis



Pau Altur Pastor
paltur@student.ethz.ch

25th April, 2025

Supervisors: Dr. Yawei Li, yawli@ethz.ch
Dr. Christoph Leitner, cleitne@ethz.ch
Giusy Spacone, gspacone@ethz.ch

Professor: Prof. Dr. Luca Benini, lbenini@ethz.ch

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Dr. Yawei Li, Dr. Christoph Leitner, and Giusy Spacone, for their continuous guidance, insightful feedback, and support throughout this project, particularly during our weekly discussions. I am also grateful to Prof. Luca Benini for his valuable input on the weekly reports. This work has greatly benefited from the contributions of other researchers at IIS working on foundation models for biosignals. In particular, I would like to thank Thorir Mar Ingolfsson and Federica Palmisano for their assistance with the use of high-performance computing resources, and Glenn Bucagu and Alexandru Dimofte for their development and introduction of the TimeFM codebase. I also extend my appreciation to the researchers at the Integrated Systems Laboratory, whose efforts in designing the devices and collecting the raw ultrasound data were foundational to this project. I am also thankful to the Cineca and CSCS organizations for providing the computational resources necessary to carry out this work. Finally, I would like to thank my friends and family for their constant support and encouragement, with special thanks to Paula for her unwavering presence and inspiration throughout this journey.

Abstract

A-mode ultrasound (US) has recently emerged as a promising modality for low-cost, continuous wearable monitoring in both clinical and everyday environments. Its applications span a range of domains, including neural interfaces, musculoskeletal monitoring, cardiovascular assessment, and bladder tracking. Despite growing interest in this technology, current research remains fragmented, with highly specialized models tailored to specific devices, acquisition protocols, and tasks. This fragmentation limits model generalization and reusability across domains. Moreover, the development of robust, general-purpose models is hindered by the limited availability of large-scale, A-mode US datasets. This scarcity is primarily due to the modality's minimal role in conventional clinical practice, where imaging-based ultrasound remains dominant. In this work, we present the first application of the foundation model framework to raw A-mode ultrasound data. Drawing inspiration from recent advances in other biosignal domains, we adopt a two-stage approach: pretraining large transformer-based models on a diverse corpus of acoustic data, including natural audio and synthetically generated ultrasound, and subsequently fine-tuning these models on smaller, labeled datasets across various downstream tasks. Our findings indicate that transformer models with increased capacity can achieve or exceed state-of-the-art performance in certain applications, such as hand movement regression. However, the overall benefit of pretraining remains inconclusive, with mixed results across different tasks and evaluation settings. These outcomes highlight both the promise and the challenges of applying foundation models to raw ultrasound data. Ultimately, this project serves as a proof of concept, laying the groundwork for future research on generalizable, large-scale modeling of raw ultrasound signals and their application in wearable health monitoring systems.

Declaration of Originality

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisors. For a detailed version of the declaration of originality, please refer to Appendix B.

Pau Altur Pastor,
Zurich, 25th April, 2025

Contents

List of Acronyms **xii**

1. Introduction **1**
1.1. Motivation . . . 1
1.2. Objective . . . 2
1.3. Ultrasound physics . . . 3
1.3.1. Ultrasound interactions . . . 4
1.3.2. Ultrasound acquisition . . . 6

2. Related work **8**
2.1. A-mode Hardware . . . 8
2.1.1. A-mode US Systems . . . 8
2.1.2. A-mode US Transducers . . . 9
2.2. A-mode US applications . . . 10
2.2.1. Discrete Hand Gesture Recognition . . . 10
2.2.2. Continuous Hand Movement Estimation . . . 11
2.2.3. Lower Limb Movement Estimation . . . 11
2.2.4. Biomedical Monitoring . . . 12
2.3. Biosignal Foundation Models . . . 12
2.3.1. Foundation Models for EEG . . . 13
2.3.2. Foundation Models for ECG and PPG . . . 14
2.3.3. Foundation Models for Wearable Data . . . 16

3. Methods **17**
3.1. Data sets . . . 17
3.1.1. Pre-training . . . 17
3.1.2. Finetuning . . . 20
3.2. US signal representation . . . 21
3.3. Tokenization . . . 22
3.4. Model Architecture . . . 23

3.5. Pre-training strategy . . . 24
3.5.1. Pre-training style . . . 25
3.5.2. Loss function . . . 25
3.6. Finetuning strategy . . . 26

4. The TimeFM repository **28**
4.1. Original State of TimeFM . . . 29
4.1.1. Datasets . . . 29
4.1.2. Data Modules . . . 29
4.1.3. Models . . . 30
4.1.4. Model Heads . . . 30
4.1.5. Tasks . . . 30
4.1.6. Criteria . . . 31
4.1.7. Schedulers . . . 31
4.2. Modifications to TimeFM . . . 31
4.2.1. Transforms . . . 32
4.2.2. Data Modules . . . 32
4.2.3. Metrics . . . 32

5. Results **34**
5.1. Pretraining . . . 34
5.1.1. Normalization strategies . . . 34
5.1.2. Depth . . . 35
5.1.3. Alpha . . . 36
5.1.4. Masking ratio . . . 36
5.1.5. Tokenization parameters . . . 37
5.1.6. Attention type . . . 39
5.1.7. MAE strategies . . . 39
5.2. Finetuning . . . 40
5.2.1. HWT . . . 40
5.2.2. HWC . . . 49
5.2.3. MCC . . . 50

6. Discussion **52**

7. Conclusion **55**

A. Declaration of Originality **56**

B. Declaration of Originality **65**

C. Dataset statistics **67**
C.0.1. Hand and wrist tracking dataset (HWT) . . . 67
C.0.2. Hand and wrist classification dataset (HWC) . . . 68
C.0.3. Muscle contraction classification dataset (MCC) . . . 69

D. Preprocessing details **71**
D.1. Pretraining datasets 71
D.1.1. Freesound 71
D.1.2. Multi-modal conversations in smart glasses (MMCSG) 71
D.1.3. USDataRecycler 72
D.2. Finetuning datasets 72
D.2.1. Hand and wrist tracking (HWT) 72
D.2.2. Hand and wrist classification (HWC) 73
D.2.3. Muscle contraction classification (MCC) 73

E. Training recipes **74**

F. What did not work **76**

List of Figures

1.1. Illustration of US wave propagation through a tissue, causing compression and rarefaction. Extracted from [6] . . . 3
1.2. Diagrams of the three main US interactions: a) total reflection, b) reflection and refraction, and c) scattering. Extracted from [7]. . . . 4
1.3. Schematics of A-mode, B-mode, and M-mode US sensing modalities. Extracted from [9]. . . . 7

3.1. Diagram of the a) scaled-dot product attention mechanism, b) multi-head self attention mechanism, and c) transformer block. Images a) and b) are excerpts of [95]. . . . 24
3.2. Diagrams of the model pipeline for both pretraining and finetuning. . . . 26

5.1. Loss evolution during pretraining according to normalization strategy. . . 35
5.2. Loss evolution during pretraining according to the number of transformer blocks (i.e. depth) of the encoder. . . . 35
5.3. Loss evolution during pretraining according to the α parameter in the loss function. . . . 36
5.4. Effect of the α parameter on each of the pretraining loss terms in the MMCSG dataset. . . . 37
5.5. Loss evolution during pretraining according to the masking ratio. . . . 38
5.6. Validation loss according to tokenization parameters. . . . 38
5.7. Loss evolution in pretraining by attention type. . . . 39
5.8. Loss evolution in pretraining for different MAE strategies. . . . 40
5.9. Sample reconstructions of pretrained models with Freesound dataset for each MAE strategy. . . . 41
5.10. Sample reconstructions of pretrained models with MMCSG dataset for each MAE strategy. . . . 42
5.11. Sample reconstructions of pretrained models with USDataRecycler dataset for each MAE strategy. . . . 43
5.12. Effect of embedding dimension on cross-validation performance in HWT . 47

C.1. Distribution of regression targets a) WR_{FE}, b) WR_{RUD}, and c) FG_{FE} of the HWT dataset. 67
C.2. Distribution of metadata values a) session and b) subject of the HWT dataset. 68
C.3. Sample distribution of gesture labels on the HWC dataset. 68
C.4. Sample distribution of sessions on the HWC dataset. 69
C.5. Sample distribution of contraction labels on the MCC dataset. 69
C.6. Sample distribution of datasets (recording sessions) on the MCC dataset. 70
C.7. Sample distribution of subjects on the MCC dataset. 70

List of Tables

3.1. Overview of pre-training datasets 18
3.2. Overview of finetuning datasets 20

5.1. Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an aggregated split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall. . 44
5.2. Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an inter-session split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall. . 45
5.3. Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an inter-session split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall. . 46
5.4. Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. Models with two different embedding dimensions and number of parameters where evaluated on each of the HWT regimes. We also compare models with MMCSG pretraining versus models with no pretraining at all. Bold values show the best performing model for each pretraining dataset. 48
5.5. Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. Full fine-tuning and linear probing were evaluated on each of the HWT experiments. Bold values show the best performing model on each task. 49
5.6. Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. For each task three models are shown: best pretrained model, no pretraining baseline, and SOTA (or encoder-less baseline). Bold values show the best performing model on each task. 49

5.7. Classification metric values (mean ± std) resulting from cross-validation on the HWC dataset for each task. Results for models pretrained on each dataset as well as a non-pretrained and encoderless baseline are shown. Bold values show the best performing model on each task. 50
5.8. Classification metric values (mean ± std) resulting from cross-validation on the HWC dataset for each task. Results from linear probing and full fine-tuning are shown. Bold values show the best performing model on each task. 50
5.9. Classification metric values (mean ± std) resulting from cross-validation on the MCC dataset for each task. Results for models pretrained on each dataset as well as a non-pretrained and encoderless baseline are shown. Bold values show the best performing model on each task. 51
5.10. Classification metric values (mean ± std) resulting from cross-validation on the MCC dataset for each task. Results for models fine-tuned with linear probing and full fine-tuning are shown. Bold values show the best performing model on each task. 51

E.1. Pretraining hyperparameters. 74
E.2. Finetuning hyperparameters. 75

List of Acronyms

AI . . . . . .Artificial Intelligence
AR . . . . . .Artificial Reality
AUPR . . . . . .Average Precision
AUROC . . . . .Area Under the Receiver Operating Curve

CNN . . . . . .Convolutional Neural Network

DOF . . . . . .Degree of Freedom

ECG . . . . . .Electro-cardio-graphy
EEG . . . . . .Electro-encephalo-graphy

GUI . . . . . .Graphic User Interface

HR . . . . . .Heart Rate

IIS . . . . . .Integrated Systems Laboratory
IoT . . . . . .Internet of Things

LAION . . . . .Large-scale Artificial Intelligence Open Network

MLP . . . . . .Multilayer Perceptron

NLP . . . . . .Natural Language Processing

RF . . . . . .Radio-frequency

sEMG . . . . . .Surface Electro-mio-graphy

SoS . . . . . .Speed of Sound

US . . . . . .Ultrasound

Chapter 1

Introduction

1.1. Motivation

Ultrasound is a cornerstone of modern medical imaging, celebrated for its real-time capabilities, safety profile, and portability. Unlike X-ray based modalities such as CT, ultrasound does not rely on ionizing radiation and, in contrast to most MRI systems, it can be deployed at the bedside, making it indispensable in point-of-care scenarios ranging from emergency diagnostics to obstetrics. Its affordability and accessibility further solidify its role in global health, particularly in resource-limited settings.

Conventional ultrasound imaging, however, represents only the final processed visual output. During acquisition ultrasound devices scan the patient with acoustic waves and record raw radiofrequency (RF) data which will later be processed to reconstruct an image. This raw data contains richer acoustic and temporal information, with great potential for advanced signal analysis and data-driven learning. In contrast to the 2D or 3D grayscale images clinicians usually interpret, RF data preserves the full fidelity of the acoustic interaction with tissue, allowing for potentially superior diagnostic inference if harnessed correctly.

Recent innovations in hardware [1] have led to the emergence of wearable and miniaturized ultrasound systems capable of capturing A-mode signals in real-time. These systems open new frontiers in continuous, unobtrusive health monitoring [2, 3], but also shift the landscape toward raw data-centric applications. Processing of data in wearable and/or embedded scenarios introduces computational and memory constraints that discourage image-based approaches due to the expensive nature of image reconstruction.

In parallel, there has been an explosion in the field of foundation models. These are large-scale models pretrained on extensive unlabeled datasets and later fine-tuned on task-specific data which demonstrate remarkable cross-domain generalization. They leverage

scale to develop flexible, reusable representations and have shown promise in medical imaging domains [4]. Applying the foundation model paradigm to raw ultrasound data could unlock new levels of performance and generalization.

Yet, a significant obstacle persists: the lack of publicly available, large-scale datasets of raw ultrasound signals. Unlike medical images or EEG and ECG, raw ultrasound data is rarely used in clinical environments which makes the collection of standardized, high quality and large enough datasets very difficult. This bottleneck has spurred our interest in using alternative data sources such synthetic raw ultrasound data and audio waveforms. These approaches may serve as scalable proxies, enabling pretraining on abundant unlabeled data while preserving relevance to the acoustic nature of ultrasound.

This thesis explores how foundation models can be adapted and applied to raw ultrasound data, with a focus on wearable and A-mode devices. It seeks to be a first proof of concept adapting the foundation model framework to a new modality, raw ultrasound data.

1.2. Objective

Building on the promise of foundation models and the unique characteristics of raw ultrasound data, this thesis aims to explore how large-scale, self-supervised learning techniques can be adapted to this emerging domain. The overarching goal is to advance the understanding of how foundational learning approaches can enable robust, scalable, and data-efficient analysis of raw ultrasound signals, particularly in the context of wearable and A-mode systems.

The specific objectives of the thesis are as follows:

- **Adaptation of Foundation Models to Raw Ultrasound Data**. The first objective is to investigate how the foundation model framework, typically employed in image, audio, and text modalities, can be extended and tailored to raw ultrasound signals. This involves rethinking model architecture, input representation, and pretraining tasks to suit the temporal and acoustic nature of RF data.
- **Self-Supervised Pretraining Using Proxy Domains**. Given the current scarcity of large-scale raw ultrasound datasets, this thesis explores the feasibility of pretraining on alternative sources such as natural audio or synthetically generated ultrasound signals. The hypothesis is that self-supervised pretraining on these structurally analogous domains can lead to improved generalization and downstream performance when fine-tuned on smaller, labeled ultrasound datasets.
- **Evaluation of Pretraining Strategies**. A core component of this research is the systematic evaluation of different self-supervised learning strategies. In particular, we investigate the impact of incorporating domain-specific signal features, such as

the signal envelope, into the pretraining task. These strategies are designed to enhance the model's ability to capture physiologically relevant information embedded in the raw signal.

- **Downstream Task Validation**. Finally, the pretrained foundation models will be fine-tuned and evaluated on a diverse set of downstream tasks involving raw ultrasound data. These tasks will serve to benchmark the utility, adaptability, and performance of the models across clinically and technically relevant applications.

Through these objectives, the thesis seeks to contribute a framework for foundational learning in raw ultrasound, offering insights into scalable methods that leverage self-supervision and proxy modalities for the next generation of A-mode ultrasound systems across a number of applications.

1.3. Ultrasound physics

The term Ultrasound (US) refers to acoustic waves with a frequency beyond the upper limit of the human audible spectrum which has been measured at 20 kHz [5]. These waves propagate through a physical medium which, in the case of medical or human-centered applications, is usually biological tissue or liquid. Ultrasound waves are longitudinal waves which means they cause the particles in the medium to oscillate along their direction of travel. This causes regions of high pressure (compression) and low pressure (rarefaction) as you can see in Figure 1.1. As in any other type of wave, only energy is propagated while the particles remain in place after the wave has passed.

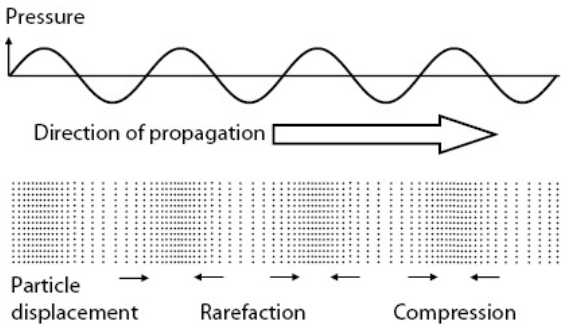


Figure 1.1.: Illustration of US wave propagation through a tissue, causing compression and rarefaction. Extracted from [6]

.

The medium through which a wave travels plays a key role in how it propagates. The transducer excites the medium at a given frequency f, which is usually in the range of 2-15 MHz. This frequency is independent of the medium and is preserved when changing from one medium to another. By contrast, the speed of the wave c or speed of sound (SoS) changes depending on the propagation medium and is fully determined by it.

Specifically, there are two properties of the medium that determine the acoustic speed of sound: the mass density ρ, which relates mass and volume of a substance, and the stiffness k, which is a measure of the material's resistance to deformation under pressure. Their relationship to the SoS is capture by the following equation:

$$c = \sqrt{\frac{k}{\rho}} \tag{1.1}$$

Another key parameter of a US wave is the wavelength, as it has a direct impact on the axial resolution of a US system. As such, it controls the ability of a US system to capture information about fine anatomical structures. Both the frequency f and the SoS c determine the wavelength λ of an acoustic wave through the equation:

$$\lambda = \frac{c}{f} \tag{1.2}$$

1.3.1. Ultrasound interactions

US sensing functions by the pulse-echo principle. This means that a transducer is triggered to send a pulsed acoustic wave through the medium, and, subsequently, this same transducer or others sense the echoes emanating from the interactions of the pulsed wave with the propagation medium. Three interactions of interest are illustrated in Figure 1.2.

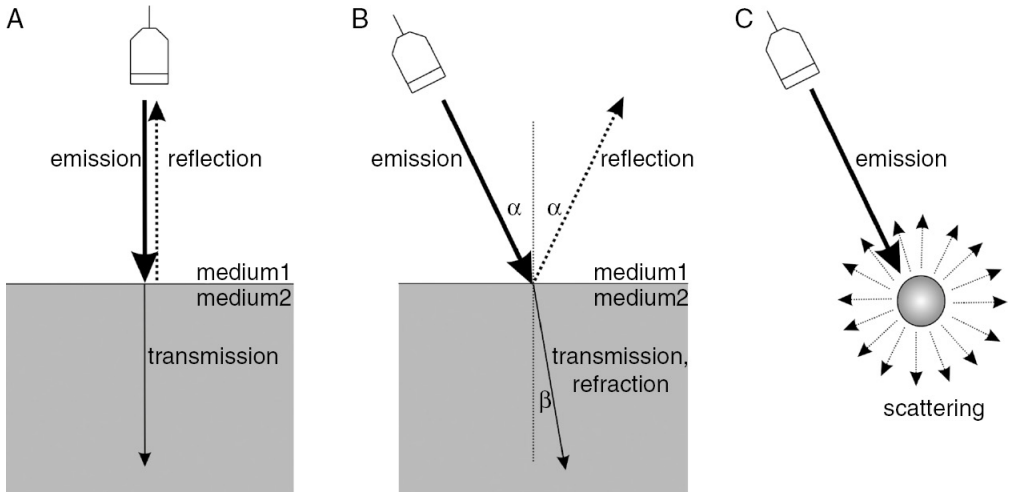


Figure 1.2.: Diagrams of the three main US interactions: a) total reflection, b) reflection and refraction, and c) scattering. Extracted from [7].

Both reflection and refraction are a product of US waves crossing the boundary between different media with different acoustic impedance. The acoustic impedance Z of

a medium is a physical parameter that describes how resistant that medium is to the propagation of acoustic waves. Mathematically, it is defined as:

$$Z = \rho \cdot c \tag{1.3}$$

with ρ once again being the medium density, and c being the speed of sound in that medium. As stated before, reflection and refraction happen when a US wave traveling through a medium encounters an interface with a different medium. The reflected pressure amplitude at an interface between medium 1 and medium 2 can be calculated by,

$$\frac{p_r}{p_i} = \frac{Z_2 - Z_1}{Z_2 + Z_1} \tag{1.4}$$

where p_r and p_i are the reflected and incident pressure amplitudes respectively, while Z_1 and Z_2 are the acoustic impedance of each medium. Assuming conservation of energy in the wave, the refracted pressure amplitude will be the difference between p_r and p_i. Thus, the larger the impedance difference between the two media, the lower the transmission of the wave through the interface. That is why US struggles to sense through dense materials such as bone and it is mostly used to image soft tissue.

It must be noted that Equation 1.4 assumes that the incident wave is perpendicular to the interface between the two media. If the acoustic wave has an incidence angle α as in Figure 1.2 B, then the wave will reflect with an equal angle, in the case of a smooth interface. Similarly, the angle of the transmitted/refracted wave will also change according to Snell's law:

$$\frac{sin(\alpha_i)}{sin(\alpha_t)} = \frac{c_1}{c_2} \tag{1.5}$$

All the above interactions however, rely on the interfaces between media, being large enough. When the interfaces are quite small, caused by inclusions of a different material in a larger medium for example, scattering happens and these inclusions are usually termed scatterers. The boundary separating interfaces that cause reflection/refraction from those that cause scattering is the wavelength. In mathematical terms, for objects whose size d is much smaller than the wavelength λ the scattered power is proportional to the frequency like,

$$W_s \propto \frac{d^6}{\lambda^4} \propto d^6 f^4 \tag{1.6}$$

This specific type of scattering is called Rayleigh scattering, and it makes up most of the echoes that are received by the transducer during US sensing. In many instances, it is a source of contrast that allows us to distinguish between different tissues [8].

As acoustic waves propagate through a medium they suffer attenuation which leads to their amplitude decreasing. This is caused by both scattering and absorption, two

phenomena which are frequency dependent. As a consequence, we can describe attenuation using the attenuation coefficient α which has units of $\frac{dB}{cm \cdot Hz}$. Thus, for a signal of frequency f we can estimate its attenuation at a depth z through,

$$\text{Attenuation } [dB] = \alpha \left[\frac{dB}{cm \cdot MHz} \right] \cdot f[MHz] \cdot z[cm] \tag{1.7}$$

1.3.2. Ultrasound acquisition

Ultrasound sensing relies on a transducer which emits pulsed acoustic waves and then records the echoes that they generate. To do so, they take advantage of the piezoelectric effect, a physical phenomena by which certain materials generate an electrical potential when they are subjected to a mechanical deformation, and alternatively, deform when an electric potential is applied to them. US transducers are generally composed of several tens to hundreds of piezoelectric crystals, each corresponding to a channel in the RF signal. When these crystals are excited with an oscillating voltage they deform periodically and emit acoustic waves at the input frequency. Once the voltage is turned off, they can sense the echoes returning to the transducer by converting their deformation into electric potential variations. If the ultrasound is stimulated with electrical pulses s of the form,

$$s(t) = p(t) \cdot cos(2\pi f_0 t) \tag{1.8}$$

where f_0 is the center frequency of the pulses, t is the time, and $p(t)$ is the envelope of the pulses, then the interfaces of the media can be expressed as,

$$r(t,n) = \sum_{i=1} A_i \cdot p\left(t - \frac{2d_i}{c} - \frac{2\nu T_s n}{c}\right) \cdot cos\left(2\pi f_0 \left(t - \frac{2d_i}{c} - \frac{2\nu T_s n}{c}\right)\right) \tag{1.9}$$

where n is the transmit pulse number, A is the scattering amplitude of the ith interface, d_i is the depth of the ith interface, c is the SoS, T is the time interval between two continuous transmit pulses ($T = \frac{1}{F_s}$, F_s is the pulse repetition frequency), and ν_i is the moving speed of the ith interface. If we neglect the effects of attenuation, this can be regarded as a convolution between the transmitted pulse and the impulse function of the interface. Thus, we can visualize the position and/or velocity of the tissue interfaces through time-of-flight and Doppler methods.

There are three main types of US sensing modalities as illustrated in Figure 1.3: A-mode, B-mode, and M-mode. A-mode is the simplest one, and it stands for Amplitude mode. In A-mode ultrasound, a single focused beam is emitted and its echoes are received by the transducer with a single channel. The recorded data corresponds to the reflected amplitudes along the signal line over time. For a known medium, there is a direct mapping between the time at which an echo was recorded and the depth within the medium from which it emanated. A-mode is the simplest form of ultrasound sensing, which allows it to be performed with small size, light mass and low-power consumption

transducers, ideal for wearable applications [9]. However, the most commonly used US modality in medical practice is B-mode imaging, where the B stands for brightness. In this variant, an image is formed from the data recorded by the transducer, which uses focused US beams to scan the tissue line by line. By recording data across many channels, a 2D US image can be formed that allows clinicians to visualize soft tissues within a patient's body [10]. Finally, there is M-mode imaging, where M stands for motion. It is a US mode that records a single scan line over time, thus allowing the operator to monitor the movement of internal structures. It is widely used in cardiac imaging to assess heart valve movements or fetal heart activity [11].

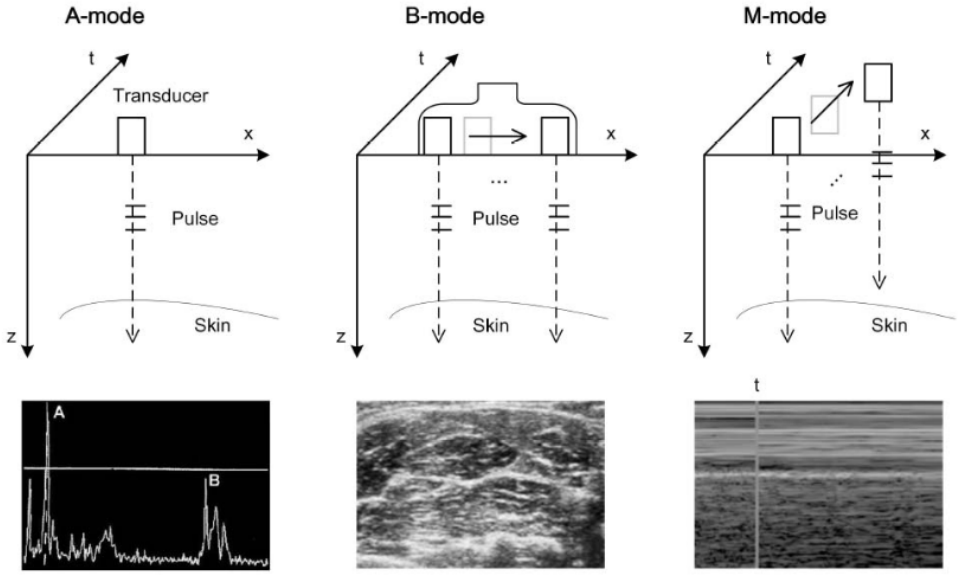


Figure 1.3.: Schematics of A-mode, B-mode, and M-mode US sensing modalities. Extracted from [9].

Chapter 2

Related work

This section provides an overview of prior research relevant to the present study. The discussion is organized into three key areas: (1) recent advancements in A-mode ultrasound (US) research hardware, with an emphasis on wearable technologies; (2) established and emerging applications of A-mode US, particularly in limb movement estimation and biomedical monitoring; and (3) the development and utilization of foundation models for biosignal processing tasks, encompassing modalities such as EEG, ECG, PPG, and other wearable sensor data.

2.1. A-mode Hardware

As interest grows in wearable sensing technologies, researchers have increasingly explored A-mode ultrasound (US), a modality based on single-element transducers and one-dimensional signal acquisition. While largely replaced in clinical practice by B-mode and other imaging techniques, A-mode US remains attractive for its simplicity, compact form factor, and potential for low-power operation, making it well-suited to wearable applications.

2.1.1. A-mode US Systems

A number of wearable A-mode US systems have emerged, each tailored for specific biomedical monitoring use cases. Protopappas et al. [12] introduced one of the earliest wearable A-mode systems for remote monitoring of long bone fracture healing. Similarly, Lanata et al. [13] proposed a compact system for cardiac monitoring. In a different domain, Piech et al. [14] designed a wireless A-mode system for neural recording in rodents. In pursuit of low-power operation, Tarbox et al. [15] introduced a system

based on time-delay spectrometry that operates using low-voltage (5 Vpp) excitation pulses, marking a shift toward energy-efficient ultrasound electronics.

More recently, attention has turned toward muscle activity monitoring. Yang et al. [16] developed a multichannel wearable system optimized for detecting muscle contractions in real time, while Yin et al. [17] further simplified the design, integrating it into a prosthetic socket to enable intuitive control of assistive devices. Brausch et al. [18] and Sgambato et al. [19] have also contributed to this space, proposing systems that emphasize form factor and usability in daily wear scenarios.

A significant milestone in this line of research is the ultra-low-power wearable platform by Frey et al. [1], known as WULPUS. This system weighs only 13 grams, consumes under 25 mW, and supports eight A-mode channels, making it one of the most integrated and versatile solutions to date for applications such as carotid artery monitoring and muscle activity detection.

2.1.2. A-mode US Transducers

Progress in A-mode ultrasound hardware has been complemented by innovations in transducer design. Most commercial A-mode transducers, originally intended for industrial or ophthalmic use, do not fully meet the ergonomic and mechanical requirements of wearable applications. To bridge this gap, several efforts have aimed at developing lightweight and flexible transducers.

Sun et al. [20] investigated both single and dual-frequency transducer designs for muscle activity sensing, combining simulation and experimental validation. Yan et al. [21] proposed a soft, wearable transducer made from polyvinylidene difluoride (PVDF), suitable for integration into human-machine interfaces. AlMohimeed et al. [22, 23] presented additional iterations of flexible transducers for continuous ultrasound sensing, while Steinberg et al. [24] demonstrated transducers capable of long-term skin contact with minimal discomfort.

Most recently, Keller et al. [25] and Giordano et al. [26] introduced fully printed, highly flexible A-mode transducers with a bending radius of 3.5 mm. These designs represent a significant advancement in conformability and are poised to facilitate broader adoption of ultrasound sensing in wearable formats.

Collectively, these developments highlight the growing maturity of A-mode US hardware and transducer technologies, establishing a solid foundation for their integration into next-generation wearable systems.

2.2. A-mode US applications

Although A-mode ultrasound has largely been replaced by B-mode imaging in clinical settings, it has recently attracted significant interest for continuous human monitoring due to its simplicity, low power requirements, and compatibility with wearable systems.

2.2.1. Discrete Hand Gesture Recognition

One particularly active area of research is discrete gesture recognition, where A-mode ultrasound has demonstrated notable potential as an alternative or complement to surface electromyography (sEMG). Several studies have compared A-mode ultrasound with sEMG in gesture classification tasks. Huang et al.[27] analyzed sparse features from a limited number of ultrasound image columns, mimicking A-mode signals, and found that ultrasound provided competitive performance to sEMG for gesture classification with similar conclusions being reached by Akhlaghi et al. [28] and Fernandes et al[29].

Other works have developed real-time classification systems using multi-channel A-mode setups. Yang et al. [30] implemented a four-channel system that achieved high classification accuracy for 11 finger gestures in both offline and online scenarios. Comparative analyses by He et al. [31] and Yang et al. [32] showed that A-mode ultrasound generally outperforms sEMG for gesture classification and is more robust to nonstationary muscle activity. Xia et al. [33] extended these findings by demonstrating improved performance through sensor fusion of A-mode ultrasound and sEMG for 20 gestures.

Beyond basic classification, Yang et al. [34] proposed a method for simultaneous prediction of finger gestures and wrist rotation, achieving high accuracy with a subclass discriminant analysis approach. Guo et al. [35] and Cai et al. [36] introduced methods for recognizing both predefined and novel gestures, with the latter applying dynamic time warping to support dynamic gesture recognition. Zeng et al. [37] further evaluated the fatigue sensitivity of A-mode versus sEMG-based systems, showing promising robustness of ultrasound in prolonged use.

For amputee applications, Yang et al. [38] applied wearable A-mode sensing and a multi-output Gaussian process model to achieve simultaneous classification of discrete finger gestures and continuous wrist motions in transradial amputees.

In terms of algorithmic advancements, deep learning has begun to play a major role. Zeng et al. [39] showed the superiority of Convolutional Neural Networks (CNNs) for gesture recognition. More recently, Vostrikov et al. [2] developed a lightweight, low-power wearable ultrasound armband integrated with an XGBoost classifier, achieving high classification accuracy with minimal session-to-session variability. They later introduced an end-to-end framework for feature extraction from raw ultrasound signals optimized for edge computing applications [40].

Together, these studies underscore the growing role of A-mode ultrasound in wearable gesture recognition systems, offering a compelling alternative to traditional biosignals.

2.2.2. Continuous Hand Movement Estimation

A related area of interest for A-mode US is continuous hand movement estimation. Early work by Guo et al. [41, 42, 43] leveraged 1-D A-mode ultrasound to monitor dynamic changes in skeletal muscle thickness during contraction. Their studies found that A-mode US outperformed sEMG in tasks involving wrist extension, highlighting its sensitivity to subtle muscle deformations.

Yang et al. [34] expanded this line of work by estimating isometric grasp forces across eight distinct hand gestures using wearable A-mode ultrasound sensors. Although they found that ultrasound-based force estimation was less accurate than sEMG [32], their results underscored the modality's viability for wearable applications. Building upon this, Zou et al. [44] introduced a multimodal approach, employing a multikernel convolutional neural network to enhance grasp force estimation from A-mode signals.

Significant progress has also been made toward decoding more complex, multi-degree-of-freedom (DoF) movements. In 2020, Yang et al. [16] demonstrated for the first time the simultaneous decoding of wrist rotation and hand grasp using wearable A-mode ultrasound outperforming sEMG. More recently, they proposed a semi-supervised learning framework to decode proportional and simultaneous wrist and hand movements [45].

Further advancing the field, Spacone et al. [3] demonstrated the regression of three hand-wrist DoFs using a lightweight, fully wearable A-mode ultrasound armband. Their method achieved state-of-the-art results, and, notably, they also addressed the critical issue of robustness to transducer repositioning across sessions, a first in A-mode ultrasound research for hand movement estimation.

Despite these advancements, current work remains largely constrained to a maximum of three DoFs, with few studies exploring robustness across subjects, devices, or varying sensor placements. These challenges highlight the need for further investigation into scalable and generalizable A-mode ultrasound solutions for continuous, multi-DoF hand movement estimation.

2.2.3. Lower Limb Movement Estimation

Recent studies have begun exploring the use of A-mode ultrasound for the control and monitoring of lower-limb prostheses. Murray et al. [46] investigated ambulation mode classification using wearable A-mode ultrasound on seven transfemoral amputees. Their system achieved a classification accuracy of 91.8% using ultrasound alone and 95.8%

when combined with kinematic data, suggesting that ultrasound can provide complementary information for prosthetic control.

Further work by Mendez et al. [47] demonstrated the feasibility of predicting knee kinematics using wearable A-mode ultrasound in transfemoral amputees, reinforcing its potential in voluntary control of lower-limb prostheses.

2.2.4. Biomedical Monitoring

Beyond neural interfaces, A-mode ultrasound has been applied in a variety of biomedical contexts. For musculoskeletal analysis, Hafthorsdottir et al. [48] demonstrated that muscle pennation angles in the gastrocnemius can be extracted from raw ultrasound data, a method later adapted for embedded devices by Vostrikov et al. [49].

In cardiovascular monitoring, Giordano et al. [50] developed a wrist-worn IoT device capable of accurate heart rate extraction using a single A-mode channel. Lin et al. [51] introduced a fully integrated ultrasonic system-on-patch, which enables continuous tracking of deep physiological signals, such as central blood pressure and cardiac output, for up to 12 hours.

Frey et al. have contributed significantly to wearable A-mode ultrasound applications: they developed a low-power, sEMG-triggered ultrasound system for long-term muscular activity monitoring [52], and a compact device for in-vivo A-mode measurement of the common carotid artery [1]. Building on this, Vostrikov et al. [53] extended the use of A-mode data for full cardiorespiratory monitoring, achieving reliable heart and respiratory rate estimates with low error margins.

In contrast, Kenny et al. [54] presented a hands-free, continuous-wave Doppler ultrasound patch that adheres to the neck and tracks blood flow velocity in the common carotid artery. Unlike A-mode systems, their work focuses on Doppler signal processing to capture hemodynamic changes using an automated algorithm.

Additionally, Leuteren et al. [55] evaluated the SENS-U, a small wireless ultrasonic sensor for continuous bladder monitoring. The device notifies users when bladder filling reaches a threshold, and was validated for everyday use in children.

2.3. Biosignal Foundation Models

Foundation models have achieved remarkable success in domains such as natural language processing and computer vision. Motivated by these developments, researchers have increasingly investigated their application to biosignals, with electroencephalography (EEG) emerging as a particularly promising area. To date, we are unaware of any works applying this approach to A-mode US making this the first project of its kind.

2.3.1. Foundation Models for EEG

Early efforts in EEG foundation modeling were led by Mohsenvand et al. [56], who adapted the SimCLR framework [57] for time-series data to enable channel-wise contrastive learning. This demonstrated the viability of self-supervised learning in the EEG domain. Building on this momentum, Kostas et al. [58] introduced BENDR, an adaptation of wav2vec 2.0 [59], which tokenized EEG signals through convolutional layers before processing them with a Transformer encoder. This approach offered sequence-length independence, but lacked modelling of spatial dependencies across EEG channels. To address this, BIOT [60] extended the architecture by incorporating frequency embeddings via FFT and learned positional encodings. With contrastive pretraining on corrupted segment pairs and linear attention modules, BIOT marked a notable improvement in both temporal and spatial understanding.

A shift toward time-frequency representations was seen in BrainBERT [61], which utilized spectrograms generated from STFT or Superlet transforms as model inputs. Despite achieving effective spectrogram reconstruction using masked autoencoding, the model was limited by its single-channel focus and reliance on private iEEG datasets. A follow-up work by Zhang et al. [62] proposed Brant, a dual-Transformer architecture that explicitly modeled inter-channel relationships through separate temporal and spatial encoders. Frequency information was encoded using spectral power features, allowing for improved spatial awareness. However, its considerable model size and use of private data introduced challenges for fair comparison. Brant2 [63] further scaled this design, introducing CNN-based tokenization and data augmentation prior to pretraining with a hybrid masked autoencoding and forecasting objective. Despite its impressive 1B parameter scale, the lack of peer-reviewed publication limits its current impact.

Parallel to these advances, EEGFormer [64] brought vector quantization to the field by discretizing Fourier-transformed EEG patches and reconstructing them via a Transformer decoder. Yet, its omission of explicit channel embeddings and limited evaluation of the quantization step raise concerns. Expanding on EEGFormer, LaBraM [65] introduced a learned VQ-VAE [66] tokenizer and trained a Transformer encoder to predict masked tokens. While this approach was trained on a broad and diverse EEG corpus, its lack of ablation studies and comparisons to existing models like BrainBERT or Brant leaves open questions about its relative effectiveness.

Beyond masked autoencoding, NeuroGPT [67] employed an autoregressive GPT-style decoder with a novel causal masking scheme. Multiple masked copies of the same input were processed per forward pass, progressively revealing more context. However, the paper omits details on how positional or channel information is encoded, potentially limiting reproducibility and interpretability.

Moving away from Transformer-based models, Mentality [68] introduced the use of Mamba [69], a selective state-space model offering linear complexity in sequence length.

It combined convolutional and linear channel mixing with a U-Net [70] and Mamba-based encoder-decoder structure. Although innovative, the lack of rigorous comparative analysis with earlier models makes it difficult to assess its relative strengths.

An alternative architecture was proposed by MultiViT [71], which trained a separate Vision Transformer for each EEG channel and aggregated [CLS] tokens across channels. While this approach captured intra-channel dynamics well, it did not enable cross-channel attention and required significant computational resources.

More recently, EEG2Rep [72] introduced a novel objective in which models predict abstract representations of unmasked data using masked inputs. This context-driven learning strategy aims to filter out irrelevant raw-level features, though its reliance on channel-mixed tokenization may obscure spatial interpretability.

Finally, CEReBrO [73] offered a more compact and interpretable alternative. It introduced a split attention mechanism to disentangle inter and intra-channel modeling, supporting both temporal and spectral inputs. This work is also the blueprint for the current project.

In summary, EEG foundation models have rapidly evolved from early contrastive learners to sophisticated architectures leveraging frequency-domain representations, attention mechanisms, and large-scale pretraining. Despite these advances, the field continues to grapple with challenges in cross-dataset generalization, consistent evaluation benchmarks, and the effective encoding of spatial and temporal EEG characteristics.

2.3.2. Foundation Models for ECG and PPG

In recent years, foundation models for biosignals such as electrocardiography (ECG) and photoplethysmography (PPG) have gained increasing attention. One of the earliest examples of self-supervised-learning (SSL) applied to ECG was presented by Cheng et al. [74], who developed a subject-aware contrastive learning framework incorporating both a subject-specific contrastive loss and an adversarial training objective. This design encourages subject-invariant representations while still capturing discriminative information relevant for downstream tasks. Their model, based on a ResNet encoder, was evaluated on both EEG decoding and ECG anomaly detection, demonstrating its flexibility across modalities. Building on the concept of physiology-informed learning, Gopal et al. [75] introduced *3KG*, a contrastive learning approach tailored to 12-lead ECG signals. Their method applies 3D augmentations to generate multiple views of each sample and processes each lead independently through a 1D convolutional neural network. Pretrained on the PhysioNet 2020 challenge dataset, 3KG showed strong performance on downstream classification tasks.

In a related vein, Kiyasseh et al. [76] proposed *CLOCS*, a suite of contrastive objectives that align ECG representations across space, time, and patient identities. Using four

datasets with varying lead configurations, CLOCS achieved consistently better performance than existing contrastive baselines such as BYOL and SimCLR, particularly when fine-tuned for clinical outcome prediction.

Extending the utility of SSL to emotion recognition, Sarkar et al. [77] developed a multi-task SSL framework wherein six distinct signal transformations serve as pretext tasks. A convolutional encoder followed by a fully connected classifier forms the architecture, and the model, pretrained and evaluated on the same datasets but with distinct tasks, achieved superior results to existing emotion recognition baselines.

Complementing these efforts, Mehari et al. [78] explored a wide range of SSL paradigms in the ECG domain, including instance discrimination and latent forecasting. Their analysis revealed that contrastively pre-trained models not only outperformed supervised baselines by approximately 1% but also exhibited enhanced label efficiency and robustness to noise, especially in clinical ECG classification tasks.

Addressing stress detection from ECG signals, Rabbani et al. [79] adapted the SimCLR framework to propose a contrastive SSL model for stress assessment. Their approach was validated on two publicly available stress-related ECG datasets, highlighting SSL's applicability to affective computing.

For ECG analysis in wearable contexts, Lai et al. [80] designed a momentum contrastive learning framework using a ResNet18 encoder and a Siamese network structure. By employing four carefully designed data augmentations, their system was able to classify 60 diagnostic categories, reflecting the potential of SSL to scale across large diagnostic vocabularies in real-world data.

In a more specific application, Kumar et al. [81] proposed a two-phase SSL method for hypopnea detection from single-channel ECG. In the first phase, an encoder learns generalized signal representations from unlabeled data, while the second phase fine-tunes both the encoder and a classifier for the target task, enabling effective detection with limited annotations.

Finally, Abbaspourazad et al. [82] presented a large-scale SSL framework for both ECG and PPG modalities using data from the Apple Heart and Movement Study (AHMS). Their model, built around an EfficientNet-style 1D CNN encoder, combines participant-aware positive pair selection with a regularized contrastive loss. This approach not only supports training on large-scale consumer-grade sensor data but also generalizes well across both ECG and PPG signals.

Together, these studies underscore the growing maturity of SSL for ECG analysis. Most of the works presented focus on contrastive learning and employ convolutional encoders. However, there is a lack of general foundation models that can generalize across tasks, and most efforts are still focused on a single application domain.

2.3.3. Foundation Models for Wearable Data

With the proliferation of consumer-grade sensors, wearable devices now generate vast amounts of longitudinal physiological data. However, the limited availability of labeled annotations and the variable quality of signals present significant challenges for model training. Recent advances in self-supervised learning (SSL) have shown great promise in leveraging unlabeled wearable data to learn generalizable and personalized representations.

An early step in this direction was taken by Hallgrimsson et al. [83], who employed a supervised attentional convolutional network to extract individualized cardiovascular response signatures from minute-level wearable data, including heart rate (HR) and step count.

Expanding beyond supervised learning, Wu et al. [84] proposed *HeartSpace*, which combines a time series encoding module with a pattern aggregation network. Using a Convolutional Autoencoder backbone and contrastive pretraining, HeartSpace captures both intra- and inter-series correlations, enabling robust representation learning for downstream tasks like personality prediction and user identification.

Spathis et al. [85], developed a self-supervised model that predicts heart rate responses from physical activity signals using a deep neural network and a custom quantile loss function. Trained on over 280,000 hours of combined wrist accelerometer and wearable ECG data, their embeddings generalized effectively across a range of downstream health and demographic prediction tasks.

In parallel, Tang et al. [86] adapted the SimCLR contrastive learning framework to the domain of human activity recognition (HAR). By systematically evaluating 81 combinations of signal transformations, they identified effective augmentation strategies and demonstrated performance gains over traditional supervised and unsupervised methods.

Finally, Yuan et al. [87] developed a large-scale self-supervised model for sleep stage classification using accelerometry and laboratory-based polysomnography. Their framework was trained and validated across diverse cohorts in three countries and demonstrated strong generalization when applied to 100,000 UK Biobank participants, showcasing the scalability of SSL for population-level inference in sleep health.

Together, these studies highlight the evolving landscape of foundation models for wearable data, driven by innovations in self-supervised learning, physiological modeling, and large-scale deployment.

Chapter 3

Methods

3.1. Data sets

Many different datasets were used in the current work. They can be broadly subdivided into two categories: pre-training datasets and finetuning datasets. Pre-training datasets are large sets of unlabeled data that are used to pre-train the foundation models in a self-supervised way while finetuning datasets are much smaller corpuses of high-quality labeled data that are used to train the model to perform relevant tasks.

3.1.1. Pre-training

Foundation model pre-training requires datasets that fulfill a set of criteria. Firstly, the data set must be large enough, given that the models to be trained usually contain millions to billions of parameters. It should also be highly-varied since the goal of pre-training is that the model learns a challenging task on a highly varied data set so that it can later generalize to a wide array of downstream tasks. These assertions are supported by previous work on large language models showing that there exist scaling laws relating model size, data set size, and compute amount [88]. These scaling laws can be used to estimate the optimal size of the pre-training data set given the size of the model, or alternatively, optimize the size of the model for the available datasets. However, there is no evidence that scaling laws derived from the field of natural language processing are directly applicable to the field of foundation models for biosignal processing, much less the specific case of raw ultrasound. There is a need for more research in this field. Ideally, the pre-training data must be of high quality and closely related to the downstream tasks we are targeting. However, in the domain of raw ultrasound, there is a distinct lack of large datasets. This is due to the fact that most medical ultrasound applications rely on US images and most commercial systems do not let the user access the raw ultrasound

data that they acquire. To get around this issue the current project has focused its efforts on exploring two alternative directions:

- The use of other acoustic datasets such as audio data, which stem from similar physical processes as ultrasound though they are distinct from it both in their physical characteristics (bandwidth, frequency composition, medium) and their acquisition (mostly single-channel, smaller sampling frequency).
- The use of synthetic ultrasound data simulated from parametric maps estimated from ultrasound images.

Table 3.1 provides an overview of the pre-training datasets that have been selected for this project.

Dataset	Modality	# channels	# samples	Memory size
Freesound	Audio	1	10k - 100M	49 Mb - 430 Gb
MMCSG	Audio	7	4M	107 Gb
USDataRecycler	Ultrasound	192 (subsampled to 32)	10k - 12M	500 Mb - 511 Gb

Table 3.1.: Overview of pre-training datasets

Freesound

Freesound [89] is a large collaborative database of audio recordings released under Creative Commons licenses. The audio clips are crowd-sourced from users across the globe and the database is curated and maintained by the Music Technology Group at Universitat Pompeu Fabra. Given its crowd-sourced nature, the database contains a huge diversity of recordings both in terms of their content and acquisition. LAION, a non-profit organization dedicated to open-source AI research, compiled the majority of the Freesound database (515581 samples totaling 3033.38 hours) and made them available through the HuggingFace repository. To compose this data set they kept only samples shorter than 3 minutes, discarded samples with a sampling rate lower than 16 kHz, and resampled all samples to 48 kHz.

In order to adapt the Freesound data set for our application, we developed a script that prepared the data for pre-training. We used the streaming feature of the HuggingFace data set to load samples on demand rather than downloading the entire data set to disk. Each sample is then split into segments of length 1000. This length was chosen as it is similar to the sample length of the downstream tasks. The data set was split into training and validation splits with an 80/20 ratio, ensuring that segments from the same sample belonged to the same set, so as to avoid data leakage. Each set is saved to its own hdf5 file in groups to make sample access and loading easier during pre-training. We composed 5 different datasets from the Freesound data with increasing number of samples: 10k, 100k, 1M, 10M and 100M. The smaller datasets are useful for debugging

and testing in modest hardware while the larger ones are to be used to pretrain the models. The most commonly used data set size was 10M as it provided a good tradeoff between its size and the time it took for models to converge when pre-trained on it.

MMCSG

The multi-modal conversations in smart glasses data set [90] comprises two-sided conversations recorded using Aria glasses, an AR research device produced by Meta. It contains multi-channel audio, video, accelerometer, and gyroscope data. The data set was released by Meta to advance research in areas such as automatic speech recognition, activity detection, and speaker diarization. Each recording features a conversation of two participants with optional background noise and is recorded with a 7-channel microphone array. In total, there are 530 recordings totaling 26 hours of audio sampled at 48 kHz.

Similarly to the Freesound data set, we had to adapt the MMCSG recordings for our task. Each clip was divided into segments of 1000 time steps. The data set was divided into training and validation sets with a split of 80/20, ensuring that segments from a specific recording belonged only to one of the sets, so as to avoid data leakage. Each set was saved to its own hdf5 file. Given its comparatively smaller size, the MMCSG data set was used in its entirety for pre-training. Only a single reduced version was composed for debugging and visualization purposes.

USDataRecycler

The Ultrasound Data Recycler data set is a synthetic ultrasound data set simulated from estimated maps of acoustic properties. The parametric maps are estimated from US images through a deep learning algorithm and then used to perform numerical simulations of wave propagation using the k-Wave MATLAB toolbox [91]. It consists of 499,950 samples simulated with 192 channels each and a duration of 1712 time steps. The total disk space of the data set is 625 Gb. For further information on how this data set was generated refer to [92].

As with the audio datasets above, the USDataRecycler data set had to be adapted for foundation model pre-training. Unlike Freesound and MMCSG, its sample length did not need to be adjusted as it was similar to the sample length for downstream tasks. However, the number of channels per sample was very large when compared with what is typically present in a wearable ultrasound scenario. Thus, each sample of 192 channels was split into 24 samples of 8 channels through randomized, non-overlapping, channel selection. This made the samples more similar to those in the downstream datasets and avoided overly large memory consumption when pre-training the models.

3.1.2. Finetuning

The finetuning datasets in this project concern a variety of tasks in the domain of ultrasound. They are small high-quality datasets acquired with a variety of hardware configurations and often times optimized for the specific task.

data set	Task	Origin	# subj.	# sessions per subj.	# ch.	# samples
HWT	Regression	Internal	5	9	4	439867
HWC	Classification	Internal	1	15	4	73607
MCC	Classification	External	8	1-7	1	212872

Table 3.2.: Overview of finetuning datasets

HWT

The hand and wrist tracking data set [3] comprises US data of the forearm acquired while the subjects performed a set of hand/wrist movements. The study measured 5 subjects for 9 sessions each. Within each session, multiple repetitions of the same hand/wrist movements were performed. The movements were categorized according to their complexity into: simple movements (1 DOF change), sequential movements (2 DOF changes sequentially), and complex movements (2 DOF changes simultaneously).

US data was acquired in the form of A-mode scans with a wearable US bracelet based on the WULPUS platform [1]. The bracelet contains 4 ultrasound probes that are excited in a round-robin fashion with a frequency of 50 Hz, resulting in a total of 80 ms per full scan. Each US probe is excited at a center frequency of 2.25 MHz and sampled at 8 MHz, acquiring a total of 400 time steps per excitation. The total number of samples across all sessions and participants is 439867.

The data set also contains three labels per sample: wrist flexion-extension (WR_{FE}), wrist radio-ulnar deviation (WR_{RUD}), and finger flexion-extension (FG_{FE}). These were acquired through an optical motion capture system comprised of six cameras that track reflective optical markers placed on the hand at a frequency of 100 Hz. For further information about the modeling of hand movements and the estimation of the ground truth labels see the original work by Spacone et al.[3].

HWC

The hand and wrist classification data set comprises US data of the forearm acquired while the subject performed a set of hand gestures. The study measured a single subject for 15 sessions. Within each session, multiple repetitions of different hand gestures were performed. In the first 7 sessions, the subject performed a set of 13 different gestures,

while for the last 8 sessions, they performed a reduced set of 4 gestures. The acquisition setup for the US scans is identical to the one described for the HWT data set. Labeling was done through manual recording of the gestures in a GUI during the experiment. In total, the data set comprises 73607 samples, each with 4 channels and 400 time steps per channel.

MCC

The muscle contraction data set is a multi-subject multi-session data set of single-channel A-mode scans acquired during exercise [93]. Eight subjects were measured while they performed squats by placing a commercial ultrasound probe on their calf muscle. The center frequency of the acquisition was 3.5 MHz. The labels registered were binary, distinguishing contracted from uncontracted states, and they were registered through the press of a button while the participants performed the action. Each subject was measured for a different number of sessions ranging from 1 to 7 and each session had a different duration.

3.2. US signal representation

In the current work we have focused on the time representation of raw US signals as waveforms. This is the most common representation as it directly reflects the way in which the US data was acquired by the transducer. Alternative representations such as frequency (Fourier or Wavelet transform) or time-frequency (spectrograms) representations exist but given the pioneering nature of this work, we decided to restrict our analysis to waveforms. The main trade-off between these two signal representations is the time versus frequency resolution. As formalized in the Heisenberg-Gabor uncertainty principle,

$$\sigma_{E,t} * \sigma_{E,f} \geq \frac{1}{4\pi}$$

the standard deviations of the time $\sigma_{E,t}$, and frequency $\sigma_{E,f}$ cannot be both arbitrarily reduced. The time representation of US signals maximizes the temporal resolution at the cost of reduced frequency resolution. Additionally, the use of the waveforms is more computationally efficient as it does not require additional transformations, and it prevents any information loss during those transformations.

For an acquisition system with C channels the waveforms W can be expressed as

$$W \in \mathbb{R}^{C \times T}$$

where T is the number of time steps acquired per channel per pulse and it can be expressed as

$$T = t/f_s$$

where t is the duration of the acquisition and f_s is the sampling frequency of the system.

A derivative of the waveform that is very widely used when analysing US data and other RF signals is the envelope which is a smooth curve outlining the extremes of a signal. In digital signal processing, it is most often estimated with the Hilbert transform,

$$H(u)(t) = \frac{1}{\pi} p.v. \int_{-\infty}^{+\infty} \frac{u(\tau)}{t-\tau} d\tau$$

which can be interpreted as the convolution of $u(t)$ by the function $h(t) = \frac{1}{\pi t}$. The envelope of US RF signals is commonly used as input for wearable ultrasound tasks such as hand gesture regressions [2, 93, 49, 9] or as a target for representational learning [3, 40].

3.3. Tokenization

Transformers are sequence-to-sequence models and thus, require their input signals to be converted into sequences of feature vectors of a given length. These feature vectors are commonly called tokens, and the process of converting the input signal into them is called tokenization. The first step is to divide the input signal $W \in \mathbf{R}^{c \times t}$ into non-overlapping patches of size p. This results in a patched input $W_p \in \mathbf{R}^{c \times N_p \times p}$ with $N_p = t/p$. Using patches rather than individual time steps reduces the memory and runtime of the attention map computation by a factor proportional to the patch size squared. The next step is to project each patch $P_{c,i} \in \mathbf{R}^p$ onto a given embedding dimension d_e through a learnable linear projection $W_{proj} \in \mathbf{R}^{d_e \times N_p}$ such that,

$$E_{c,i} = W_{proj} P_{c,i}^{\top} \tag{3.1}$$

Thus, the tokenized input becomes $W_{tok} \in \mathbf{R}^{c \times N_p \times d_e}$. The embedding dimension of the tokens will be preserved throughout the model, and it has a significant impact on the model size as it increases the width of the MLP layers in each Transformer block. In this work, we have experimented significantly with the embedding dimension as it has a big effect on the model's power and its tendency to underfit/overfit. In practice, both patching and projection are performed in a single step through a learnable convolutional layer with a stride equal to the patch size and the number of kernels equal to the embedding dimension. Patching and projection are performed per channel so that each token only contains information from a single channel. This is unlike what is usually done for RGB images where the three channels are mixed together. The rationale behind this design choice is that we want our models to generalize to datasets with different channel configurations and to preserve the information from each individual transducer as it may refer to different parts of the body. To ensure that the channel and temporal information of each token is preserved once passed to the model, we add learnable

channel and temporal encodings $W_{chan} \in \mathbf{R}^{N_p \times d_e}$ and $W_{temp} \in \mathbf{R}^{c \times d_e}$. Thus, the final input embeddings can be expressed as,

$$E^{in}_{c,i} = E_{c,i} + W_{chan,c} + W_{temp,i} \tag{3.2}$$

where, $W_{chan,c}$ is the positional encoding for channel c and $W_{temp,i}$ is the positional encoder for token i.

3.4. Model Architecture

The architecture of the encoder used in this project is the Transformer, a model architecture that has revolutionized sequence-to-sequence tasks such as NLP and EEG modelling. Transformers are based on a combination of attention layers and multi-layer perceptions that operate on the input tokens at a given embedding dimension d_e. As can be see in Figure 3.1, each transformer block is composed of two sub-layers: an attention layer and a multi-layer perceptron. Additionally, each sub-layer is followed by layer normalization which leverages skip connections to normalize the sublayer output by its input. Thus, their output is

$$LayerNorm(x + Sublayer(x)) \tag{3.3}$$

The classic attention layer introduced in [94] is a function of three vectors: the query (Q) key (K) and value (V) vectors. These are obtained through linear projection of the input tokens X like,

$$Q, K, V = XW^Q, XW^K, XW^V$$

They are then combined to obtain the attention values through,

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_e}}\right)V \tag{3.4}$$

yielding an attention map $A \in \mathbf{R^{l^2}}$ where l is the length of the input sequence. Multi-head attention was introduced in [95] and it splits Q, K, V projection into h attention heads each with a dimensionality of d_e/h. The attention map is then computed by,

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{3.5}$$

where,

$$Head_i = Attention(XW^Q_i, XW^K_i, XW^V_i) \tag{3.6}$$

thus concatenating the individual attention computations and projecting them back to the embedding dimension d_e. The strength of Transformers lies in the attention layer which allows it to attend to relevant long-range dependencies while ignoring irrelevant ones. Simultaneously, given their lack of context compression, Transformer's memory

and time complexity scales quadratically with the length of the input sequence. To get around this constraint several modifications have been proposed [73] but in the current work we will focus on the original attention mechanism.

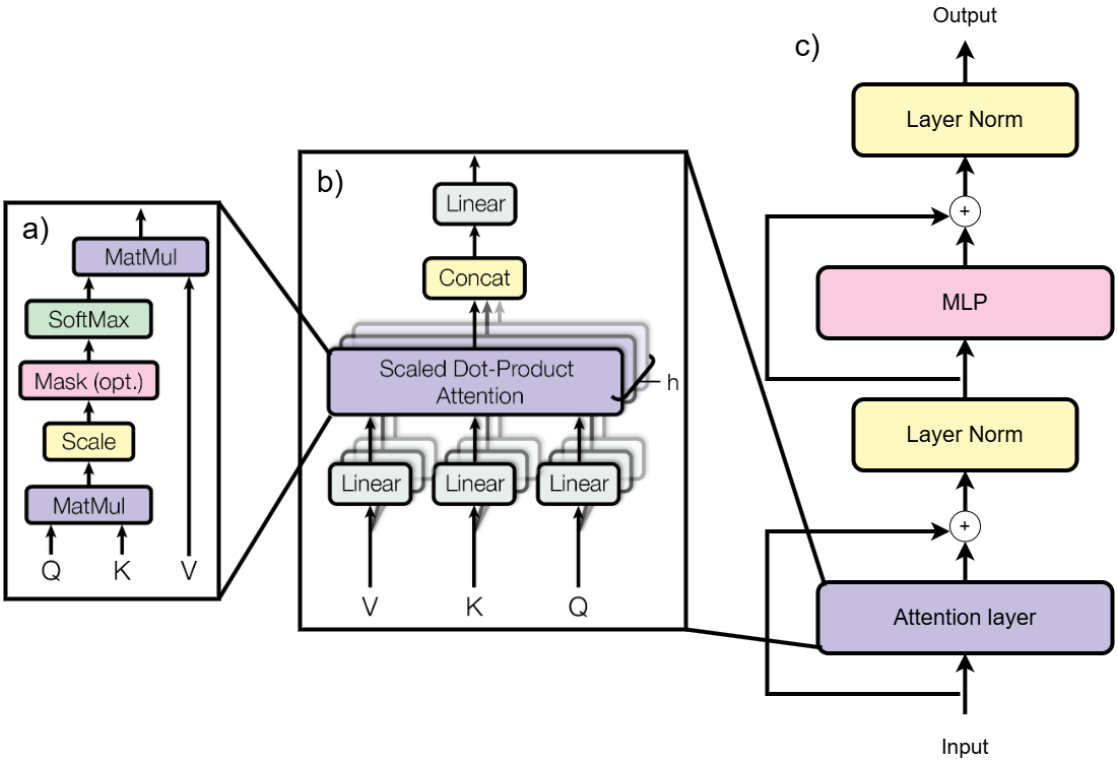


Figure 3.1.: Diagram of the a) scaled-dot product attention mechanism, b) multi-head self attention mechanism, and c) transformer block. Images a) and b) are excerpts of [95].

Our models do not use [CLS] tokens. Instead, during finetuning, we mean pool the encoder output tokens and use this as the input for the model head. We have employed fully-connected model heads in this work with 2 layers: a first layer of 50 neurons and a second layer with as many neurons as needed for the output label. After the first layer we apply a Leaky-ReLU activation. The choice of this model head architecture is motivated to make comparison with encoderless baselines easier. Given the fully-connected nature of our model, the embedding dimension d_e of the encoder (or lack thereof) will influence the number of parameters in the model head.

3.5. Pre-training strategy

We pretrained our encoders using the Masked Autoencoding (MAE) task. This is a very common pretraining task used by several high performing vision [96], timeseries [97] and

biosignals foundation models [73]. Alternatively, we could use contrastive learning, an invariance-based method that works by training the model to generate similar embeddings for different augmentations of the same sample, and it can be supplemented by enforcing different embeddings for different samples (these are known as positive and negative pairs, respectively) [98]. This paradigm is very popular in NLP and computer vision [99, 100] because these modalities benefit greatly from augmentations while still keeping most of their semantic information intact. In US, although there exist augmentation strategies, it is unclear what effect they have on the underlying semantic information present in the signal, given that biosignals such as US are very hard to interpret by humans when compared to images or text.

MAE works by randomly masking a proportion of the input tokens and substituting them by a learnable [MASK] token. The data is then fed to the encoder that generates one embedding of dimension d_e per token. Each of these embeddings is then passed to a common linear layer that projects them back to the original patch size, and the reconstruction is compared with the input. The model learns by computing the L_2 loss and optimizing its parameters according to it. This specific MAE strategy is inspired by SimMiM [96], and has been successfully applied both to images and EEG data [73]. Figure 3.2a presents a visual representation of this approach. In this work, we utilized a masking ratio of 50 % as that is consistent with the current literature.

3.5.1. Pre-training style

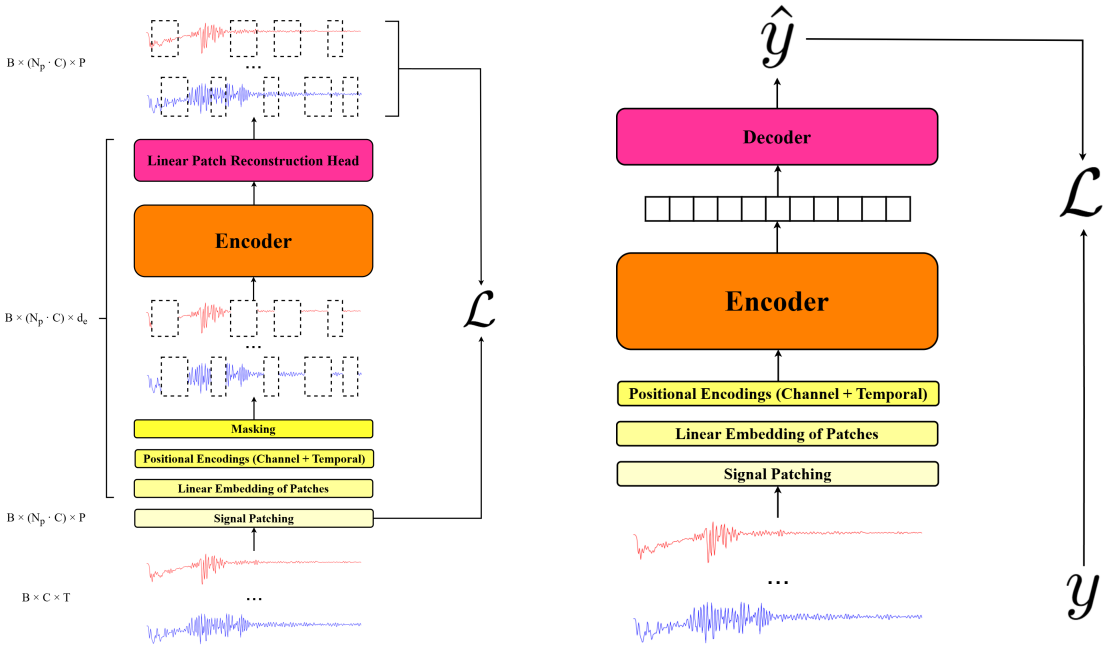
As mentioned in section 3.2, the envelope is a relevant characteristic of the US signal that is often employed to extract features for prediction tasks. That is why we decided to incorporate it into the model pre-training, yielding three different variations of pre-training:

- Wave2Wave: This is the standard MAE task in which the model receives the masked waveform as input and is asked to reconstruct it.
- Wave2Env: In this case, the model receives the masked waveform as input and is asked to predict the envelope of the input as its output.
- Env2Env: In this case, the model receives the masked envelope as input and is asked to reconstruct it.

3.5.2. Loss function

The model is trained to reconstruct the masked and unmasked patches by minimizing the following loss function:

$$\mathcal{L}_{global} = \mathcal{L}_{masked} + \alpha \cdot \mathcal{L}_{visible}$$



(a) Pretraining (b) Finetuning

Figure 3.2.: Diagrams of the model pipeline for both pretraining and finetuning.

The hyperparameter α balances the contributions from the masked and visible terms of the loss. This provides stability during training and helps the model learn at the start of training. We chose the L_2 loss as our loss function, as it has been found to be the best performing in previous works dealing with masked EEG waveform reconstruction [73].

3.6. Finetuning strategy

During pre-training each model is trained for a fixed number of epochs and it is evaluated on a disjoint validation set after each epoch. At the end of training the best performing model on the validation set is selected. There is no perfect linear correlation between validation loss and performance in downstream tasks but this is the more principled approach to take and commonly used in other works. The pretrained encoder is then taken and a decoder is trained on its outputs to accomplish a task of interest. Finetuning can be performed in two different ways: full finetuning where the encoder and model head are optimized jointly or linear probing, where the encoder is frozen and only the model head is fine-tuned. The latter option is useful in assessing how informative the pre-trained embeddings are for downstream tasks, but full finetuning generally yields better results. In addition, to test the effectiveness of pre-training we also train a model from scratch. During fine-tuning we train a model until the loss ceases to improve for

more than 3 epochs or training reaches 100 epochs. A visual diagram of fine-tuning can be seen in Figure 3.2b.

Chapter 4

The TimeFM repository

The work carried out in this project has been performed within the framework of TimeFM. TimeFM is a code repository hosted on Github that focuses on pre-training and finetuning of time-series foundation models in a research setting. It is the product of a collaboration between the Computer Vision and the Integrated Systems laboratories at ETH Zürich and it is actively being developed. Most of the previous work on TimeFM revolves around foundation models for EEG. To our knowledge, ultrasound is the first modality different from EEG that is incorporated into TimeFM.

TimeFM relies on PyTorch Lightning [101] for model training. Pytorch Lightning is an open-source Python library that provides a high-level interface for Pytorch code. Firstly, it organizes and separates the code relating to the model, the data handling and the training. It also automates the training process through its Trainer module which automatically performs forward and backward passes and optimizes the model parameters. This reduces boilerplate code and makes the codebase more compact, readable and maintainable. Finally, it enables models to run at scale by automating the detection and handling of accelerators both locally and in distributed settings.

Training of foundation models involves the careful selection of many hyperparameters controlling everything from data loading and preprocessing to model architecture and optimization. To keep track of them and enable reproducible research it is essential to employ a flexible and effective configuration management framework. Hydra [102] is an open-source Python library that provides a hierarchical configuration framework allowing dynamic composition and overriding of configurations. This allows us to test different experimental conditions and evaluate their effect on the models. Each module in TimeFM has its dedicated yaml file that defines its default condition and allows for experiment-specific modifications where needed.

4.1. Original State of TimeFM

I got access to TimeFM on October 11th 2024. I was instructed to take the work by Dimofte and Bucagu as the foundation for this project. This section details the state of the codebase at that point, which can be consulted in the "cvpr_split_attn" branch of TimeFM: `https://github.com/ofsoundof/TimeFM/tree/cvpr_split_attn_fm`. The following description has been extracted from their master's thesis report [103].

4.1.1. Datasets

Dataset classes in PyTorch handle how the data is loaded from disk and prepared before being input to the model. There are different dataset classes for each downstream task, as well as a specific class for HDF5 files. Each dataset has an attribute defining the path to the data file as well as several others defining which transformations should be applied to the data, whether the data will be used for pre-training or finetuning, and how to cache the dataset during training. The datasets must have a `__len__` method that outputs the number of samples in the dataset and a `__getitem__` method that returns one sample.

4.1.2. Data Modules

The data modules in this work are all subclasses of `LightningDataModule`. This is an object in PyTorch Lightning that aims to encapsulate all of the code relating to data management. They must have a setup method that performs any steps needed prior to training and then several dataloader methods for each split: `train_dataloader`, `val_dataloader`, `test_dataloader`, and `predict_dataloader`. Each of these methods returns a PyTorch dataloader object, which wraps the dataset objects presented in section 4.1.1 and allows them to be iterated in batches during training or inference. Originally, there were two data modules:

- `EEGDataModule` takes in three datasets for training, validation and testing and wraps them in a standard PyTorch DataLoader. Despite its name it is not at all specific to EEG.
- `ConcatenatedEEGDataModule` takes in a list of datasets and parameters defining the ratio of data to use for validation or testing. The input datasets are concatenated and then split randomly. Once again, despite its name its implementation is not specific to EEG.

4.1.3. Models

In TimeFM, the term "model" refers specifically to an "encoder", i.e. a module that generates embeddings that can later be used by a model head for either reconstruction of the original data or as input to accomplish a relevant downstream task like regression or classification. The only encoder available in TimeFM when I started to work with it was the so-called CEReBrO encoder. It is an encoder based on the Transformer architecture, and it can be instantiated with a varying number of blocks, attention heads, and many other parameters.

4.1.4. Model Heads

The term "model head" is often used interchangeably with the term "decoder" in the foundation model literature. It refers to a prediction/reconstruction head that can be attached to an encoder and used during pre-training or finetuning. There are several model heads available in TimeFM:

- `PatchReconstructionHead` is a linear model used during pretraining that maps the output tokens of an encoder to their original dimension.
- `MAEDecoder` is a Transformer decoder used during pretraining that maps the output tokens of an encoder to their original dimension.
- `MLPClassificationHead` is a fully connected model for classification tasks. Each layer is composed of dropout, a linear mapping, and optionally a user-defined activation. The last layer maps the hidden dimension to the number of output classes in the task.
- `MultistepRegression` is a linear model that takes in the mean of the output tokens of an encoder and maps them to a number of regression targets.

In this case, the corresponding Hydra configuration controls hyperparameters such as the input and output dimensionality of the model head as well as the width and depth of hidden layers.

4.1.5. Tasks

The main element of a PyTorch Lightning pipeline is the `LightningModule` which controls what happens in each training, validation, test or prediction step, as well as the optimization and logging. In TimeFM these modules are called "tasks" and they encapsulate the forward and backward pass for each specific problem. They also implement data augmentations, learning rate decays and encoder freezing/finetuning. They can be divided into:

- Pre-training tasks:

  - `SimmimMAETask` implements a self-supervised masked auto-encoding regression task where the encoder takes in the masked and unmasked tokens, and the decoder is a linear layer.
  - `VITMAETask` implements a self-supervised masked autoencoding regression task where the encoder only takes in the visible tokens while the decoder, which is Transformer-based, takes in the encoder outputs and the [MASK] tokens at relevant positions.

- Finetuning tasks:
  - `RegressionTask` implements a supervised regression task with user-defined encoder and decoder.
  - `ClassificationTask` implements a supervised classification task with a user-defined encoder and decoder.

4.1.6. Criteria

The criteria define which loss function to optimize for a given model and task. During pre-training, in addition to the specific loss function, we must also define the relative contributions of the masked and unmasked terms, as well as the modality of the input and output (waveforms vs spectrogram). For finetuning, we must also select the appropriate measure (L1 or L2 loss vs cross entropy), whether the output is binary or multi-class, the number of targets, the weighing of different classes and how much label smoothing to apply.

4.1.7. Schedulers

Learning rate schedulers control the value of the learning rate over the entire training period. In addition to the type of scheduler used, we must also specify the base learning rate and the duration of the warm-up period.

4.2. Modifications to TimeFM

In order to adapt the previous pipeline of TimeFM we have had to make several modifications to the existing pipeline. Below are the most significant. For an updated view of the TimeFM repository including our updates you can consult the "us_trf_fm" branch at `https://github.com/ofsoundof/TimeFM/tree/us_trf_fm`.

4.2.1. Transforms

Transforms as we have defined them in this project are any sort of modifications to the input data and/or labels that are performed before each sample is input to the model. Additionally, they should only depend on the input itself and user-defined parameters. Originally, each dataset had a predetermined set of transforms that could be applied or not depending on the user. This made the code inflexible and difficult to reuse for new modalities or datasets. On the other hand, there was also a preprocessor module that transformed the data in the `LightningModule`, rather than within the `LightningDataModule`, something that goes against the spirit of PyTorch Lightning.

To improve this, we leveraged the ability of Hydra to recursively instantiate objects, together with the transform framework of the TorchVision library [104]. Now, each dataset has two optional transform parameters in its constructor, `sample_transform` and `label_transform`, which expect a PyTorch module with a `forward` method. Thus, a single transform can be provided, or multiple independent transforms can be stacked together using TorchVision's `Compose` transform. In this way, an arbitrary transform can be configured solely by modifying some lines in a configuration file. Additionally, it removes unnecessary code within the dataset for every transform, as well as the control arguments for each of the predefined transforms.

4.2.2. Data Modules

Since all the downstream tasks in this project are evaluated through Leave-One-Out Cross-Validation, a new data module had to be created for it. This data module, named `CrossValidationDataModule` has two file path (or list of file paths) arguments, `train_files` and `val_files`, which defines which files will be part of each split. It also has a `dataset_target` argument that controls which Dataset class will be used to load and preprocess all the files and a `dataset_params` argument which defines all the relevant arguments for the Dataset class beyond the file path.

4.2.3. Metrics

Originally each task in TimeFM had a set of hardcoded metrics that would be computed and logged during training. This made the code hard to repurpose for new tasks where the user might be interested in a different set of metrics to compare to the state of the art. To enable more flexibility, we took advantage of the TorchMetrics library [105], a collection of over 100 metric implementations in PyTorch that are rigorously tested, standardized and compatible with distributed training. Their `MetricCollection` class allows the user to define a list of metrics that they want to track and reduces the amount of boilerplate code significantly. Once again, Hydra's recursive instantiation of objects allows maximum flexibility without having to modify the codebase, while TorchMetrics

deals with all of the accumulation over batches, synchronization across devices and logging.

Chapter 5

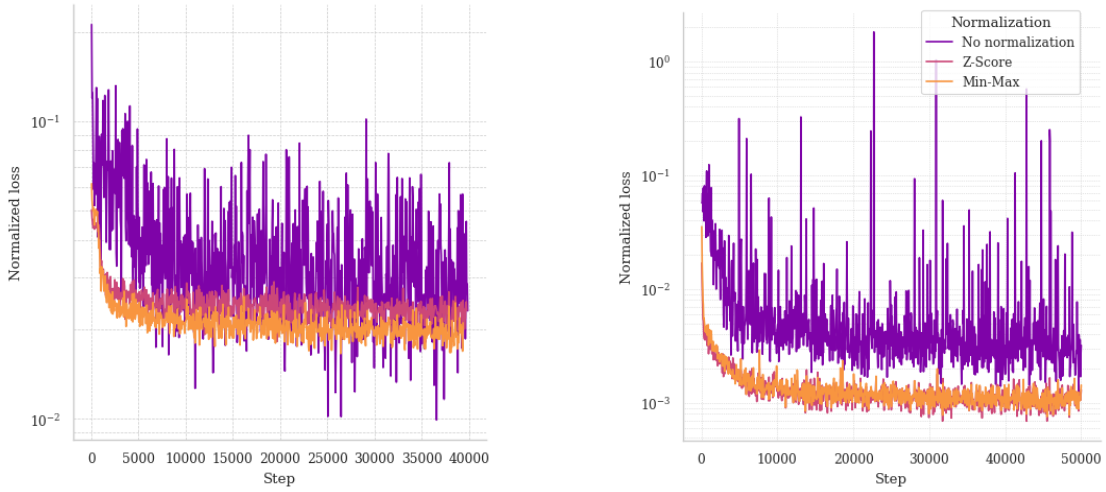
Results

5.1. Pretraining

In this section, the analysis of results will be mostly quantitative, looking at the influence of different model parameters on pretraining performance in different contexts, thus informing the choices for the models that will be applied downstream. Most of the experiments have been performed on pretraining with audio datasets (Freesound and MMCSG), due to their availability from the beginning of the project, and the findings have been used to inform pretraining with synthetic US data (USDataRecycler).

5.1.1. Normalization strategies

Normalization of input data is a common tool used in machine learning to standardize inputs and aid model convergence. In this project we explored two different normalization strategies: min-max normalization and z-score normalization. Figure 5.1 shows the results of this analysis. For direct comparison, the loss values for each normalization approach are divided by the square of the signal's dynamic range, since the loss function used is based on the L2 loss. This is done to allow fair comparison between runs with different scales for the ground truth. It can clearly be seen that not normalizing the data leads to loss instability and slower learning than normalization in both cases. In the Freesound case, min-max normalization is slightly better than Z-score normalization. In the MMCSG case, both min-max and Z-score normalization show very similar performance. Thus, we have chosen min-max normalization as the default normalization strategy during pretraining.

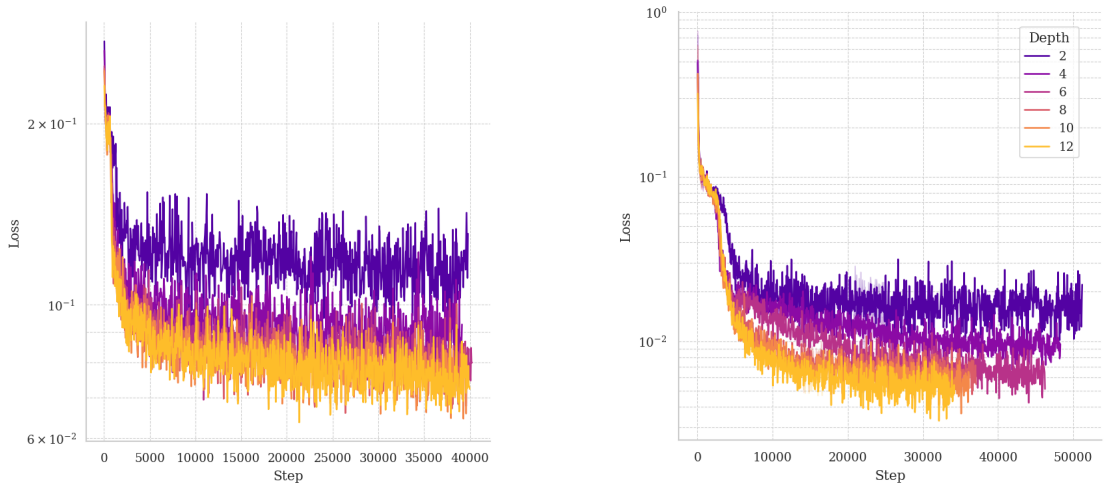


(a) Freesound (b) MMCSG

Figure 5.1.: Loss evolution during pretraining according to normalization strategy.

5.1.2. Depth

Another key parameter in Transformer models is the number of blocks comprising the model, which is usually termed the depth of the model. In Figure 5.2 we present the performance of pretraining models with increasing depth.



(a) Freesound (b) MMCSG

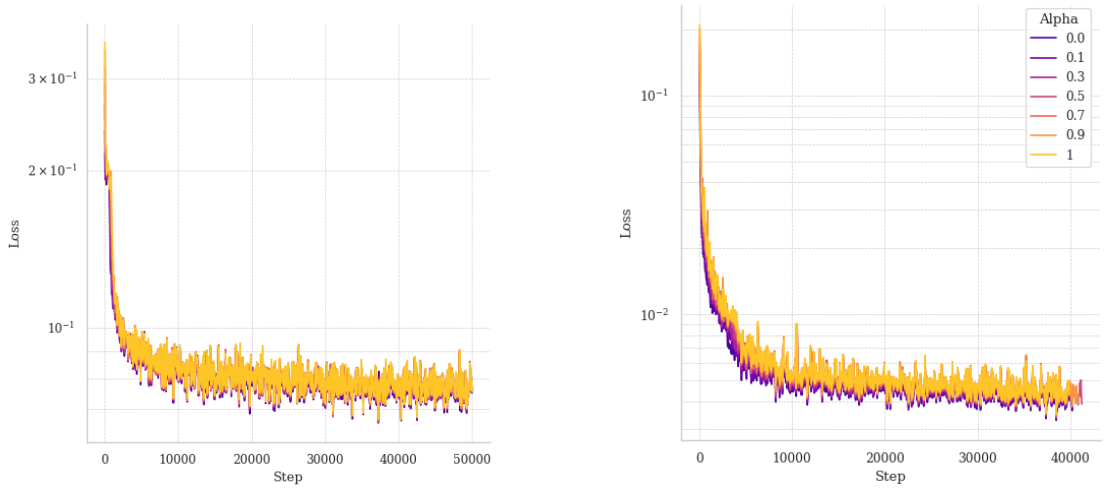
Figure 5.2.: Loss evolution during pretraining according to the number of transformer blocks (i.e. depth) of the encoder.

As expected, the deeper the model, the better performance is, though we see diminishing

returns as we add more blocks. That is why for this work, we have mostly focused on models with 8 blocks, as this seems to be a good tradeoff between model performance and size.

5.1.3. Alpha

As explained in section 3.5.2 the parameter α balances the loss contributions of the masked and unmasked tokens in the reconstruction task. As such it should play a key role in ensuring proper learning during pretraining. Figure 5.3 shows however, that the effect of varying alpha on the loss value during pretraining is minimal.



(a) Freesound (b) MMCSG

Figure 5.3.: Loss evolution during pretraining according to the α parameter in the loss function.

A more thorough investigation of the models response to α is presented in Figure 5.4. We can see that the effect of alpha on the masked term is minimal to non-existent. Its only effect is on the unmasked term whose reconstruction Mean Absolute Error decreases for increasing α something to be expected. Thus, in this project, we have opted to keep α at 1, since it simplifies the loss to a conventional $L2$ loss and gives equal weight to both terms of the loss function.

5.1.4. Masking ratio

Figure 5.5 shows the influence of the masking ratio parameter on model performance during pretraining. This parameter controls what percentage of tokens is masked before

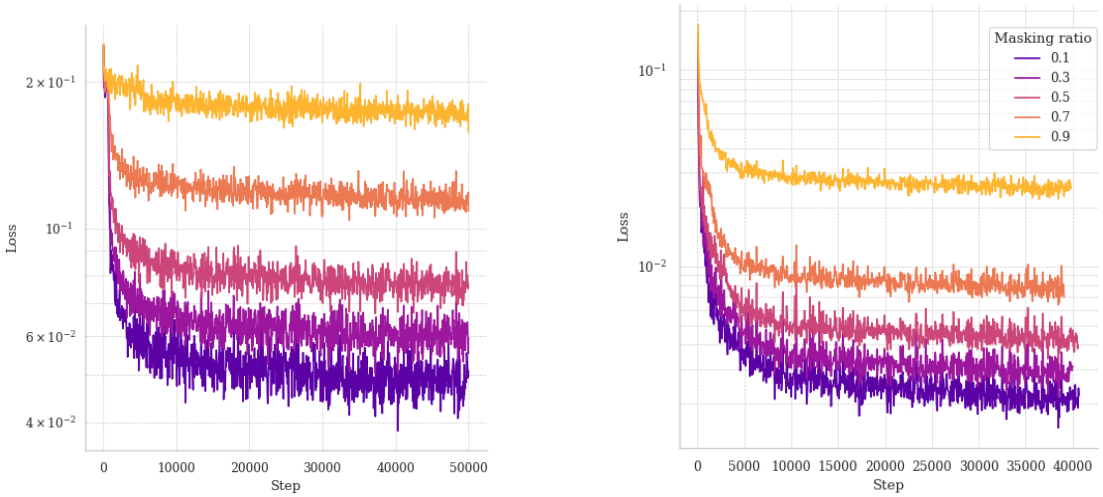
Figure 5.4.: Effect of the α parameter on each of the pretraining loss terms in the MM-CSG dataset.

being input to the encoder. The results follow intuition, as the larger the masked percentage is the worse it performs. Nevertheless, the aim of pretraining is to challenge the model so it learns meaningful representations of the data. Thus, we opted for a masking ratio of 0.5 as it has been shown to be optimal for masked waveform paradigms in other works [73].

5.1.5. Tokenization parameters

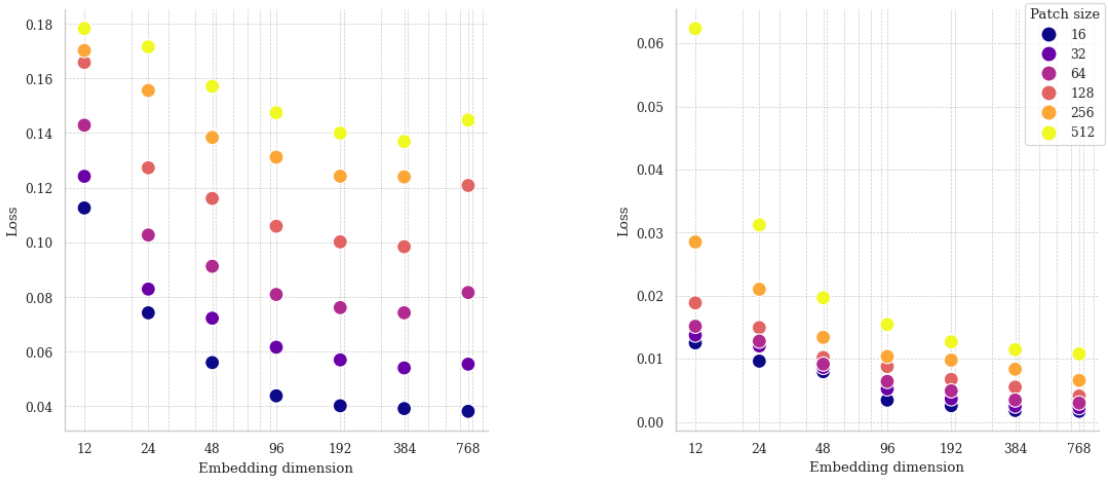
A key duo of parameters that are relevant for Transformer-based models of biosignals is the patch size and the embedding dimension as explained in Section 3.3. Patch size controls how granular the tokens are and has a significant impact on memory consumption. Embedding dimension determines the power of the model by controlling the space it operates in and the number of parameters it has. Figure 5.6 displays the validation loss as a function of these two parameters.

We can see two clear trends: the smaller the patch size the lower the loss is, while the opposite is true of the embedding dimension. In the case of the patch size, though the model does not significantly increase its number of parameters with decreasing patch size, it does incur a penalty in terms of memory consumption which should be kept in mind, specially for compute constrained settings. For the embedding dimension, we see diminishing returns for larger embedding dimensions since the number of parameters grows linearly with embedding dimension (which in this experiment is growing exponentially). Thus, for this work we have opted for a patch size of 16 and an embedding dimension of 192.



(a) Freesound (b) MMCSG

Figure 5.5.: Loss evolution during pretraining according to the masking ratio.

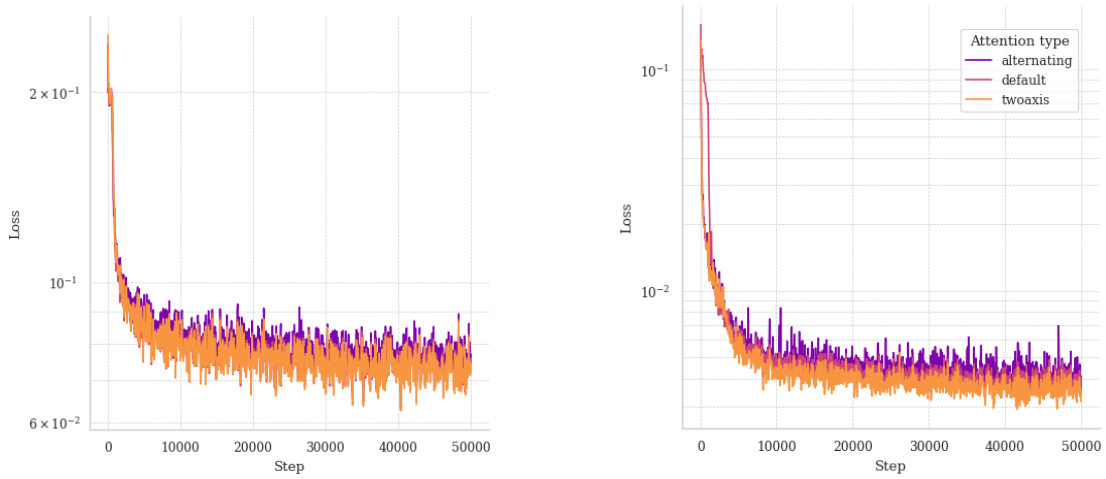


(a) Freesound (b) MMCSG

Figure 5.6.: Validation loss according to tokenization parameters.

5.1.6. Attention type

In Dimofte et al. [73], the authors introduced two new attention computation strategies to minimize the memory overhead which becomes a problem in settings such as EEG or US that benefit from small patch sizes and often have inputs with hundreds of channels. These two strategies are termed alternating and two-axis attention. Their aim is to split the attention computation into intra-channel and inter-channel attention. Figure 5.7 shows the pretraining loss for each of these attention types as well as the conventional attention computation. We can see that performance is very similar, with alternating attention being slightly worse than default and two-axis attention. Thus, in compute constrained settings any of two-axis and alternating attention could be chosen to minimize memory consumption. However, since this project did not face such constraints, we opted for the default attention mechanism as a first proof of concept, so as to reduce complexity.



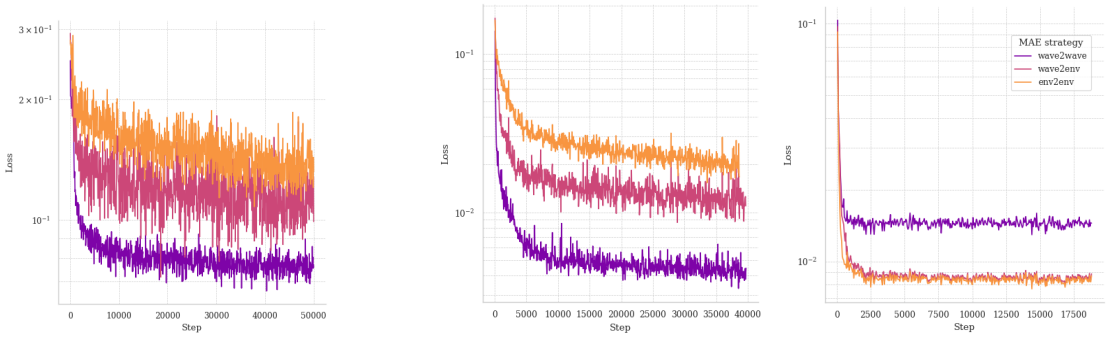
(a) Freesound (b) MMCSG

Figure 5.7.: Loss evolution in pretraining by attention type.

5.1.7. MAE strategies

As presented in section 3.5.1 this project explores three different Masked Auto Encoding approaches: Wave2Wave, Wave2Env, and Env2Env. Figure 5.8 shows pretraining performance for each paradigm and each dataset. In both Freesound and MMCSG, the Wave2Wave strategy has the lowest loss, followed by Wave2Env and Env2Env. By contrast in USDataRecycler, Wave2Env and Env2Env show very similar performance while Wave2Wave is much worse.

Figures 5.9, 5.10, and 5.11 shows some sample reconstructions for each of the pretraining strategies and datasets. It is clear that reconstructions of masked patches is harder than



(a) Freesound (b) MMCSG (c) USDataRecycler

Figure 5.8.: Loss evolution in pretraining for different MAE strategies.

that of unmasked patches for all models. In the case of the models pretrained on audio datasets (Freesound and MMCSG) the reconstruction of masked tokens is still relatively good.

By contrast, in Figure 5.11 we can see that the models pretrained on USDataRecycler struggle significantly with the reconstruction of masked tokens. USDataRecycler Wave2Wave specially displays very poor reconstruction of masked tokens which is usually a flat line. We have tried to improve this through several modifications such as increasing model and dataset size, reducing α or patch size but results have not varied significantly. A possible reason for this is the comparatively higher frequency of US features compared to audio ones, which might be harder for the model to learn and reconstruct appropriately.

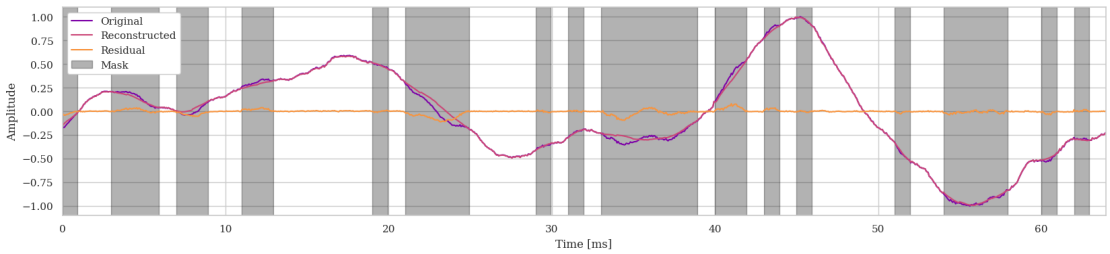
5.2. Finetuning

In this section, we present the results of the best performing pretrained models in comparison to previous state of the art when available. We also show some relevant tests for specific aspects of the models or the finetuning process.

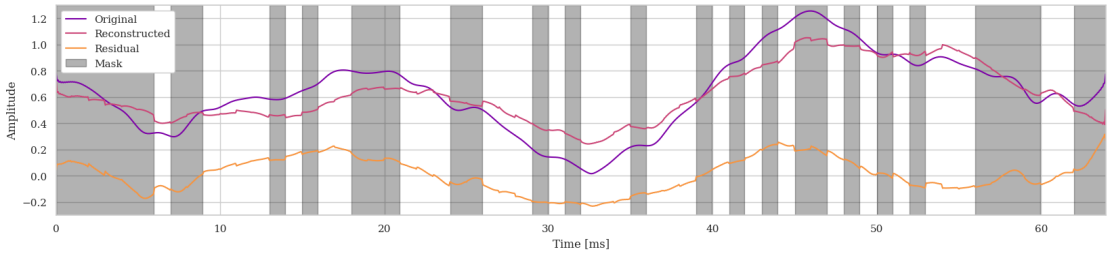
5.2.1. HWT

Effect of pretraining

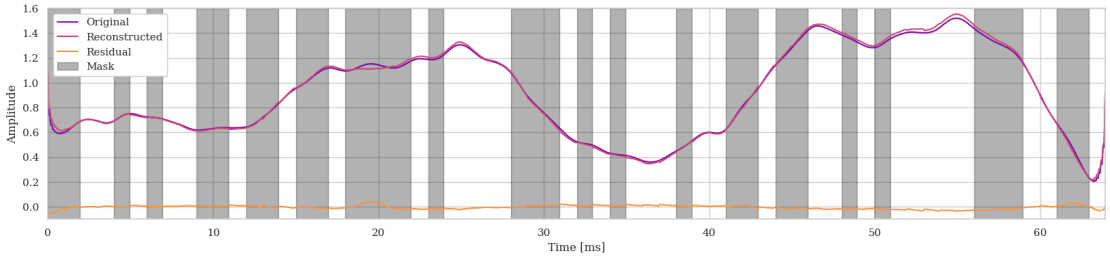
Tables 5.1 5.2 5.3 show the performance of pretrained models on the HWT dataset. Model performance has been evaluated with three distinct splits of increasing complexity:



(a) Wave2Wave

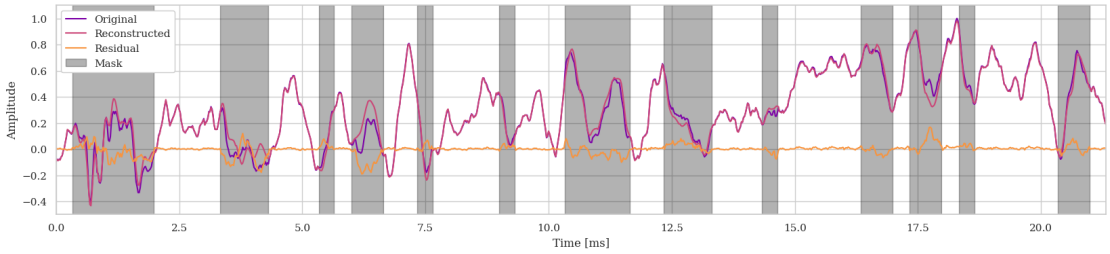


(b) Wave2Env

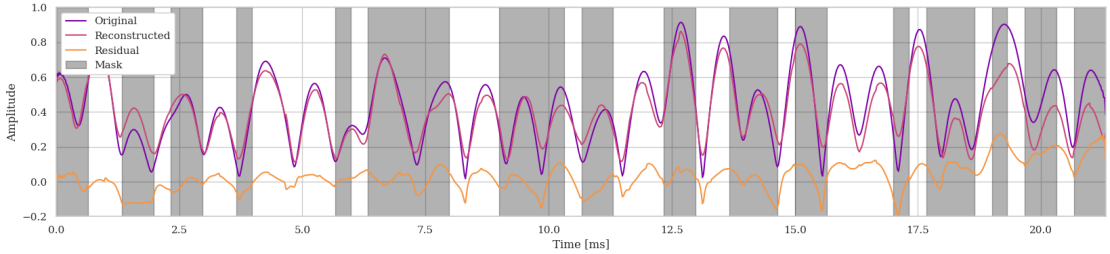


(c) Env2Env

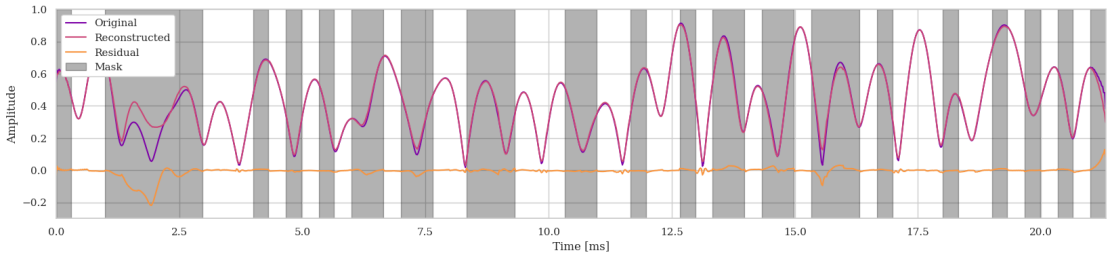
Figure 5.9.: Sample reconstructions of pretrained models with Freesound dataset for each MAE strategy.



(a) Wave2Wave

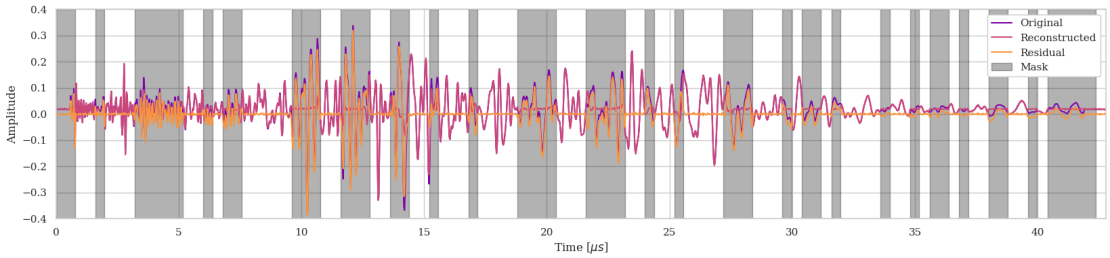


(b) Wave2Env

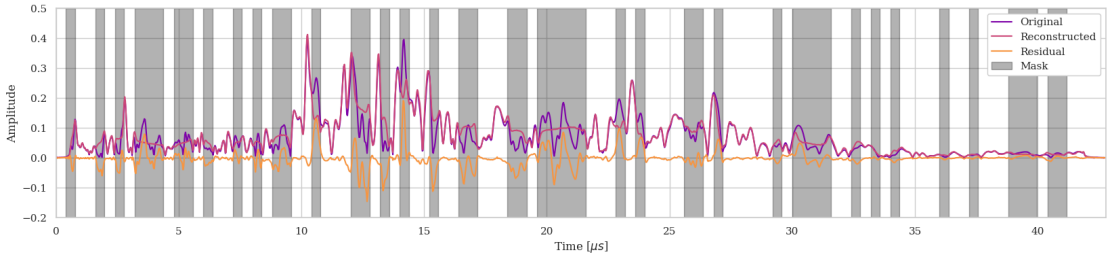


(c) Env2Env

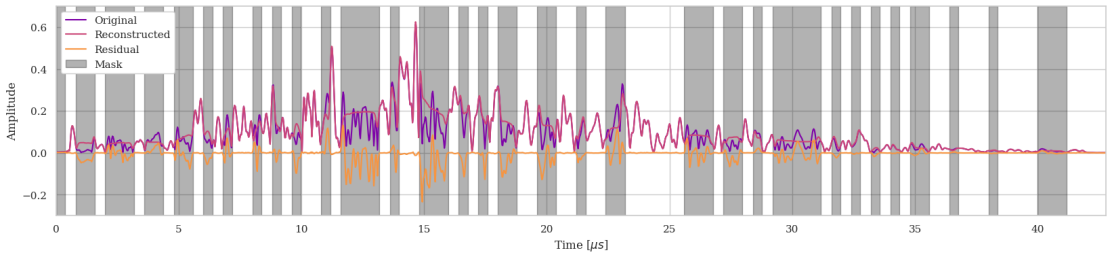
Figure 5.10.: Sample reconstructions of pretrained models with MMCSG dataset for each MAE strategy.



(a) Wave2Wave



(b) Wave2Env



(c) Env2Env

Figure 5.11.: Sample reconstructions of pretrained models with USDataRecycler dataset for each MAE strategy.

- Aggregated split where samples are randomly assigned to 1 of 5 folds without any constraints.
- Inter-session split where each fold contains samples from one session of each subject.
- Inter-subject split where each fold contains all the samples from one subject.

In the aggregated case (Table 5.1), all models achieve very good performance on this task with Mean Absolute Errors around or below 2° for all three regression targets. However, there are some interesting differences across datasets. For the audio datasets (Freesound and MMCSG), Wave2Wave is the optimal pretraining strategy. In contrast, the Wave2Env model pretrained on the USDataRecycler dataset outperforms the other two pretraining strategies. This might indicate that the Wave2Env strategy is only useful when the pretraining data comes from a modality that is similar to the downstream one. Nevertheless, all pretrained models are worse than the model fine-tuned directly on HWT without pretraining. This means that the current pretraining is not providing a benefit for this task and might actually be detrimental.

Table 5.1.: Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an aggregated split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall.

Model Characteristics		**MAE**		
Dataset	**Strategy**	$\mathbf{WR_{FE}}$ [°]	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{FG_{FE}}$ [°]
None	None	**1.09 ± 0.01**	**0.97 ± 0.01**	**1.3 ± 0.02**
Freesound	Wave2Wave	**1.66 ± 0.04**	**1.44 ± 0.04**	**2.15 ± 0.05**
	Wave2Env	1.79 ± 0.07	1.57 ± 0.05	2.3 ± 0.08
	Env2Env	1.77 ± 0.09	1.64 ± 0.06	2.27 ± 0.08
MMCSG	Wave2Wave	**1.26 ± 0.06**	**1.17 ± 0.04**	**1.56 ± 0.1**
	Wave2Env	1.61 ± 0.01	1.43 ± 0.01	2.1 ± 0.02
	Env2Env	1.65 ± 0.03	1.45 ± 0.03	2.14 ± 0.02
USDataRecycler	Wave2Wave	1.68 ± 0.11	1.46 ± 0.11	2.16 ± 0.13
	Wave2Env	**1.38 ± 0.06**	**1.22 ± 0.07**	**1.85 ± 0.08**
	Env2Env	1.62 ± 0.07	1.43 ± 0.06	2.11 ± 0.07

In the inter-session case (Table 5.2), the error values increase across the board as does their standard deviation. However, as in the previous case, there is a difference between the best performing strategy for models pretrained on audio versus the best performing strategy for models pretrained on the USDataRecycler dataset. In this case, the Env2Env strategy yields the best performance when pretraining on Freesound and MMCSG, while Wave2Env is once again the optimal choice for pretraining on the USDataRecycler.

Additionally, the best pretrained model in this experiment, USDataRecycler pretraining with a Wave2Env strategy, outperforms the model trained from scratch on the HWT data, even if by a small margin. This means that in this case, there might be a benefit to pretraining.

Table 5.2.: Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an inter-session split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall.

Model Characteristics		**MAE**		
Dataset	**Strategy**	$\mathbf{WR_{FE}}$ [°]	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{FG_{FE}}$ [°]
None	None	**7.16 ± 1.38**	**6.65 ± 1.04**	**9.15 ± 2.74**
Freesound	Wave2Wave	7.7 ± 1.17	6.59 ± 0.97	10.48 ± 2.7
	Wave2Env	7.72 ± 1.14	6.42 ± 0.78	10.38 ± 2.19
	Env2Env	**7.52 ± 1.46**	**6.35 ± 0.69**	**9.61 ± 1.98**
MMCSG	Wave2Wave	7.26 ± 1.28	6.74 ± 0.82	**9.05 ± 2.05**
	Wave2Env	7.44 ± 1.34	**6.3 ± 0.66**	9.72 ± 2.4
	Env2Env	**7.24 ± 1.19**	6.36 ± 0.7	9.82 ± 2.28
USDataRecycler	Wave2Wave	7.17 ± 1.2	6.29 ± 1.12	9.53 ± 2.7
	Wave2Env	**6.75 ± 1.22**	**5.73 ± 0.96**	**9.07 ± 2.66**
	Env2Env	7.25 ± 1.15	6.28 ± 0.84	9.43 ± 2.01

In the inter-subject experiment (Table 5.3), the performance of all models drops significantly and oftentimes there is no model that performs better across all three regression targets. In the case of pretraining on Freesound, the Env2Env model outperforms the others on WR_{FE} and WR_{RUD} estimation while it is actually 9° worse at estimating FG_{FE}. For MMCSG pretraining, each task is better than the others at one of the regression targets. Finally, for USDataRecycler, the Wave2Wave method performs best for WR_{RUD} and FG_{FE} while Wave2Env is better at estimating WR_{FE}. This stands in contrast to the previous experiments where Wave2Env was the best performer. In comparison to the model fine-tuned directly on HWT, the best models from each pretraining dataset perform better at estimating WR_{FE} and FG_{FE} while they perform worse at estimating WR_{RUD}. Overall, the large errors and the variability in performance across models makes it very hard to extract any conclusions and probably means this task at present is too difficult for these models to tackle successfully.

Table 5.3.: Mean absolute error values (mean ± std) resulting from evaluating the models with cross-validation on the HWT dataset with an inter-session split. Bold values show the best performing model for each pretraining dataset, and underlined values show the best performing model overall.

Model Characteristics		**MAE**		
Dataset	**Strategy**	$\mathbf{WR_{FE}}$ [°]	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{FG_{FE}}$ [°]
None	None	**14.25 ± 2.65**	**15.16 ± 1.36**	**39.1 ± 11.96**
Freesound	Wave2Wave	17.16 ± 3.89	16.21 ± 2.07	34.18 ± 9.78
	Wave2Env	13.74 ± 1.95	15.98 ± 1.65	**31.57 ± 9.42**
	Env2Env	**12.68 ± 2.28**	**15.80 ± 2.55**	40.35 ± 18.17
MMCSG	Wave2Wave	**13.54 ± 2.19**	17.81 ± 5.34	38.54 ± 16.27
	Wave2Env	13.76 ± 2.91	17.96 ± 6.11	**35.86 ± 10.77**
	Env2Env	14.71 ± 1.61	**17.58 ± 6.71**	37.41 ± 14.2
USDataRecycler	Wave2Wave	14.16 ± 2.85	**17.56 ± 3.24**	**33.32 ± 11.96**
	Wave2Env	**13.48 ± 1.98**	17.61 ± 4.81	37.92 ± 14.86
	Env2Env	13.71 ± 1.92	17.77 ± 6.36	36.89 ± 14.6

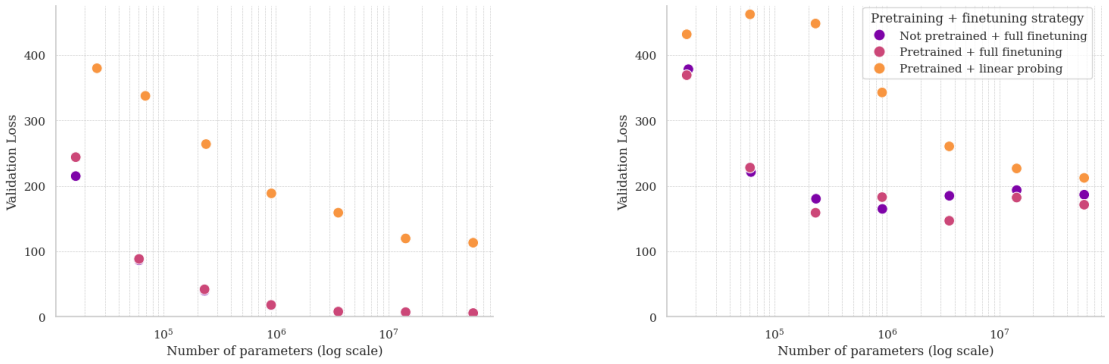
Effect of model power

A key parameter to consider when looking at improvements in performance on the HWT task is the power of the model. Models with more parameters can improve performance in supervised tasks with small datasets by overfitting. This will lead to a lack of generalization when applying the model to unseen data. To explore the interplay between model power and performance, we have tested the performance of models with increasing number of parameters on the aggregated and inter-session tasks in the HWT dataset. The results can be seen in Figure 5.12. We have performed a logarithmic sweep of the embedding dimension parameter which is linearly correlated with the number of parameters in the model.

In the aggregated case (Figure 5.12a), we can see that the model's performance, gets better with increasing number of parameters and there is no limit to their improvement. This could be explained by the model memorizing the training set, which contains samples from all sessions and subjects. Then, during validation, the model could be interpolating by finding the most similar sample in the training set to each validation sample and assigning a similar label. This would allow it to transfer this performance to the validation set, but would fail with the addition of new data from other sessions and/or subjects.

By contrast, in the inter-session setting we introduce a domain shift between training

and validation set. In Figure 5.12b we can observe that the fine-tuned models display an initial improvement in performance as embedding dimension grows, but it soon stops. Models with a number of parameters between 10^5 and 10^7 perform very similarly. This means that the increased model power in this case does not translate to better performance, likely because of the models inability to bridge the domain shift between training and validation sets.



(a) Aggregated split

(b) Inter-session split

Figure 5.12.: Effect of embedding dimension on cross-validation performance in HWT

Table 5.4 shows the Mean Absolute Error for each target of pretrained and non-pretrained models with embedding dimension 48 and 192, respectively. For the aggregated setting there is a significant difference between the performance of the smaller and larger model. However, in the inter-session case, this difference is much smaller and in the inter-subject case it is negligible with both models performing very similarly and oftentimes the smaller model outperforming the larger one. This supports the hypothesis that larger models benefit from overfitting in the simpler setting but are unable to translate their performance gains to the more complex tasks.

Effect of fine-tuning strategy

Following on the analysis of HWT, Table 5.5 compares the performance of pretrained models according to the finetuning strategy used. Linear probing keeps the encoder frozen and only finetunes the model head, while full finetuning trains both together. We can see that full finetuning produces a large performance improvement on both of the tasks. This finding suggests that pretraining with MMCSG does not directly translate into improved downstream performance, as the encodings learned during pretraining produce very poor performance without additional finetuning. In the inter-subject case, the performance of linear probing slightly edges out that of full-finetuning. This supports the idea that more powerful models overfit on the training set and as a result are biased in cross-validation. Linear probing, by freezing the encoder, reduces the model power and its capacity to memorize the training set.

Table 5.4.: Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. Models with two different embedding dimensions and number of parameters where evaluated on each of the HWT regimes. We also compare models with MMCSG pretraining versus models with no pretraining at all. Bold values show the best performing model for each pretraining dataset.

Model Characteristics			**MAE**		
Task	**Pretraining**	**ED**	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{WR_{FE}}$ [°]	$\mathbf{FG_{FE}}$ [°]
Aggregated	None	48	3.85 ± 0.13	3.97 ± 0.03	4.77 ± 0.05
		192	**0.97 ± 0.01**	**1.09 ± 0.01**	**1.30 ± 0.02**
	MMCSG	48	3.71 ± 0.13	3.86 ± 0.15	4.68 ± 0.20
		192	1.17 ± 0.04	1.26 ± 0.06	1.56 ± 0.10
Inter-session	None	48	7.66 ± 1.94	8.14 ± 1.60	11.13 ± 3.90
		192	**6.65 ± 1.04**	**7.16 ± 1.38**	9.15 ± 2.74
	MMCSG	48	7.23 ± 0.83	7.61 ± 0.96	10.43 ± 2.84
		192	6.74 ± 0.82	7.26 ± 1.28	**9.05 ± 2.05**
Inter-subject	None	48	15.77 ± 2.17	13.23 ± 0.52	34.22 ± 10.72
		192	**15.16 ± 1.36**	14.25 ± 2.65	39.10 ± 11.96
	MMCSG	48	17.03 ± 3.16	13.63 ± 1.19	32.93 ± 11.28
		192	17.81 ± 5.34	**13.54 ± 2.19**	**38.54 ± 16.27**

Comparison to State of the Art

Finally, it is useful to compare the best performing models on each task to the previously reported state of the art (SOTA). Table 5.6 shows the best pretrained model in each task alongside the SOTA and the performance of a model fine-tuned from scratch.

In the aggregated task both foundation models perform significantly better than the SOTA, with the model without pretraining being the best. For the inter-session case the foundation models once again beat the SOTA but the pretrained model is better than the model tuned from scratch and it is significantly better than the SOTA. It must be noted that the results reported in Spacone et al. [3] were obtained by training a single model per subject whereas in this project we have trained the models on the combined data of all subjects. This provides more data for training but also increases the variability in the data that is not associated with the target of interest. In the inter-subject case, there is no reported SOTA so we compare to a model trained to predict the regression targets directly from the signal envelope. In this case, the foundation models also perform better than the encoder-less baseline.

Table 5.5.: Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. Full fine-tuning and linear probing were evaluated on each of the HWT experiments. Bold values show the best performing model on each task.

Model Characteristics			**MAE**		
Pretraining	**Task**	**Finetuning**	$\mathbf{WR_{FE}}$ [°]	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{FG_{FE}}$ [°]
MMCSG	Aggregated	Linear probing	7.56 ± 0.13	7.94 ± 0.09	8.78 ± 0.12
		Full finetuning	**1.26 ± 0.06**	**1.17 ± 0.04**	**1.56 ± 0.1**
	Inter-session	Linear probing	10.16 ± 1.08	10.3 ± 0.7	13.35 ± 1.63
		Full finetuning	**7.26 ± 1.28**	**6.74 ± 0.82**	**9.05 ± 2.05**
	Inter-subject	Linear probing	**13.46 ± 2.97**	**13.74 ± 1.81**	**35.96 ± 10.2**
		Full fine-tuning	13.54 ± 2.19	17.81 ± 5.34	38.54 ± 16.27

Table 5.6.: Mean absolute error values (mean ± std) resulting from cross-validation on the HWT dataset. For each task three models are shown: best pretrained model, no pretraining baseline, and SOTA (or encoder-less baseline). Bold values show the best performing model on each task.

Model Characteristics					**MAE**		
Task	**Study**	**Dataset**	**Strategy**	**Parameters**	$\mathbf{WR_{FE}}$ [°]	$\mathbf{WR_{RUD}}$ [°]	$\mathbf{FG_{FE}}$ [°]
Aggregated	This work	None	None	3.6 M	**1.09 ± 0.01**	**0.97 ± 0.01**	**1.3 ± 0.02**
		MMCSG	Wave2Wave	3.6 M	1.26 ± 0.06	1.17 ± 0.04	1.56 ± 0.1
	Spacone et al. [3]	GRAWUS	Env Reconstruction	5 K	4.59 ± 0.69	4.8 ± 1.11	6.54 ± 1.47
Inter-session	This work	None	None	3.6 M	7.16 ± 1.38	6.65 ± 1.04	9.15 ± 2.74
		USDataRecycler	Wave2Env	3.6 M	**6.75 ± 1.22**	**5.73 ± 0.96**	**9.07 ± 2.66**
	Spacone et al. [3]	GRAWUS	Env Reconstruction	5 K	7.56 ± 2.0	7.46 ± 2.0	10.36 ± 5.0
Inter-subject	This work	None	None	3.6 M	14.25 ± 2.65	**15.16 ± 1.36**	39.1 ± 11.96
		Freesound	Wave2Env	3.6 M	**13.74 ± 1.95**	15.98 ± 1.65	**31.57 ± 9.42**
		None	Decoder(Envelope)	80K	16.78 ± 3.31	16.96 ± 2.45	35.37 ± 11.01

5.2.2. HWC

Table 5.7 shows the performance of models pretrained on audio and synthetic ultrasound on each of the HWC tasks. As in HWT we evaluate the performance on an aggregated and inter-session split. However, since the dataset contains data coming from a single subject we do not evaluate an inter-subject split. Given the lack of a published SOTA, we have added the results of a model with no encoder for comparison.

In the aggregated task, both the pretrained and non-pretrained foundation models outperform the encoderless model, with the non-pretrained being slightly better. However, in the inter-session case, the performance of all models is significantly worsened and now the pretrained model substantially outperforms the non-pretrained one and slightly edges out the encoderless model. This can be taken as support of the theory that pretraining

Table 5.7.: Classification metric values (mean ± std) resulting from cross-validation on the HWC dataset for each task. Results for models pretrained on each dataset as well as a non-pretrained and encoderless baseline are shown. Bold values show the best performing model on each task.

Task	Model	Pretraining	Accuracy	AUROC	AUPR
Aggregated	Encoder + Decoder	None	**0.94 ± 0.0**	**0.99 ± 0.0**	**0.94 ± 0.0**
		Freesound	0.84 ± 0.0	0.94 ± 0.0	0.89 ± 0.0
		MMCSG	0.92 ± 0.0	**0.99 ± 0.0**	0.93 ± 0.01
		USDataRecycler	0.8 ± 0.01	0.93 ± 0.0	0.87 ± 0.0
	Only decoder	None	0.86 ± 0.0	0.97 ± 0.0	0.87 ± 0.0
Inter-session	Encoder + Decoder	None	0.25 ± 0.15	0.49 ± 0.24	0.43 ± 0.23
		Freesound	0.12 ± 0.09	0.12 ± 0.01	0.73 ± 0.08
		MMCSG	**0.37 ± 0.15**	**0.56 ± 0.25**	**0.6 ± 0.2**
		USDataRecycler	0.06 ± 0.04	0.1 ± 0.02	0.65 ± 0.08
	Only decoder	None	0.36 ± 0.14	0.54 ± 0.25	0.55 ± 0.21

is helpful to overcome session domain shift as seen in the HWT dataset.

Table 5.8 shows the difference between linear probing and full fine-tuning in the HWC case. As in the case of HWT, full fine-tuning shows better performance when compared with linear probing both in the aggregated and inter-session, lending further support to the idea that pretrained embeddings learned on MMCSG do not translate well to downstream tasks.

Table 5.8.: Classification metric values (mean ± std) resulting from cross-validation on the HWC dataset for each task. Results from linear probing and full fine-tuning are shown. Bold values show the best performing model on each task.

Task	Pretraining	Finetuning	Accuracy	AUROC	AUPR
Aggregated	MMCSG	Linear probing	0.75 ± 0.01	0.95 ± 0.0	0.73 ± 0.01
	MMCSG	Full finetuning	**0.92 ± 0.0**	**0.99 ± 0.0**	**0.93 ± 0.01**
Inter-session	MMCSG	Linear probing	0.19 ± 0.15	0.48 ± 0.27	0.36 ± 0.14
	MMCSG	Full finetuning	**0.37 ± 0.15**	**0.56 ± 0.25**	**0.6 ± 0.2**

5.2.3. MCC

Table 5.9 shows the result of fine-tuning our pretrained models on the MCC dataset along with non-pretrained and encoderless baselines. In the aggregated setting, both pretrained and non-pretrained models perform better than the encoderless baseline, with the model pretrained on USDataRecycler slightly outperforming the rest. By contrast, in the inter-subject case, the encoderless baseline performs better than both with regard

to accuracy and precision, while it is worse than the pretrained models in terms of recall and F1-score. Overall, the inter-subject task seems very difficult for the models as they all perform very poorly.

Table 5.9.: Classification metric values (mean ± std) resulting from cross-validation on the MCC dataset for each task. Results for models pretrained on each dataset as well as a non-pretrained and encoderless baseline are shown. Bold values show the best performing model on each task.

Task	Model	Pretraining	Accuracy	Precision	Recall	F1-score
Aggregated	Encoder + Decoder	None	0.96 ± 0.0	0.71 ± 0.01	0.71 ± 0.01	0.71 ± 0.01
		Freesound	0.96 ± 0.0	0.71 ± 0.01	0.71 ± 0.01	0.71 ± 0.01
		MMCSG	0.96 ± 0.0	0.71 ± 0.01	0.71 ± 0.01	0.71 ± 0.01
		USDataRecycler	**0.97 ± 0.0**	**0.72 ± 0.01**	**0.72 ± 0.0**	**0.72 ± 0.0**
	Only decoder	None	0.87 ± 0.0	0.64 ± 0.01	0.6 ± 0.01	0.6 ± 0.01
Inter-subject	Encoder + Decoder	None	0.49 ± 0.05	0.36 ± 0.13	0.36 ± 0.2	0.32 ± 0.16
		Freesound	0.51 ± 0.06	**0.42 ± 0.07**	0.4 ± 0.1	**0.37 ± 0.08**
		MMCSG	0.5 ± 0.05	0.41 ± 0.09	0.4 ± 0.18	0.36 ± 0.15
		USDataRecycler	0.47 ± 0.04	0.41 ± 0.07	**0.41 ± 0.17**	0.36 ± 0.12
	Only decoder	None	**0.6 ± 0.09**	**0.42 ± 0.08**	0.37 ± 0.15	0.34 ± 0.1

Table 5.10 shows the comparison between linear probing and full fine-tuning for MCC tasks.

Table 5.10.: Classification metric values (mean ± std) resulting from cross-validation on the MCC dataset for each task. Results for models fine-tuned with linear probing and full fine-tuning are shown. Bold values show the best performing model on each task.

Task	Pretraining	Finetuning	Accuracy	Precision	Recall	F1-score
Aggregated	MMCSG	Linear probing	0.45 ± 0.0	**0.84 ± 0.0**	0.6 ± 0.01	0.61 ± 0.0
	MMCSG	Full finetuning	**0.96 ± 0.0**	0.71 ± 0.01	**0.71 ± 0.01**	**0.71 ± 0.01**
Inter-subject	MMCSG	Linear probing	**0.51 ± 0.06**	0.4 ± 0.07	0.34 ± 0.15	0.32 ± 0.12
	MMCSG	Full finetuning	0.5 ± 0.05	**0.41 ± 0.09**	**0.4 ± 0.18**	**0.36 ± 0.15**

While in the aggregated case full fine-tuning performs significantly better than linear probing for all metrics, except for precision, in the inter-subject case both approaches perform very similarly, with linear probing slightly edging out on accuracy while being worse for Recall and F1-score. Overall, this supports the findings that more powerful models struggle to overcome domain shifts between training and validation.

Chapter 6

Discussion

This work has investigated the application of foundation models to the domain of raw ultrasound (US) data, with a particular focus on their utility as a general framework for modeling and analysis across a variety of tasks and datasets. Notably, we have demonstrated that transformer-based models can match or even exceed state-of-the-art performance in hand movement regression tasks. Furthermore, we have shown that such models have the potential to unify processing pipelines across different acquisition protocols, devices, and datasets, thereby offering a standardized approach to working with raw US data.

Our exploration of pretraining strategies yielded several key insights. We found that foundation models benefit from small patch sizes and large embedding dimensions during pretraining. Given the inherently multi-channel nature of US data, this results in a substantial number of input tokens per sample. Combined with the high parameter count of transformer architectures, this leads to significant memory and computational demands. These findings underscore the need for more efficient attention mechanisms to reduce memory overhead during training. Approaches such as alternating attention [73] are promising candidates in this regard.

However, the utility of pretraining on downstream tasks remains inconclusive. While some tasks, such as inter-session hand movement tracking (HWT), showed modest benefits from pretraining, others exhibited neutral or even negative effects. Importantly, we observed no consistent advantage between audio-based and synthetic ultrasound-based pretraining modalities. Among the pretraining strategies evaluated, models trained with USDataRecycler and the Wave2Env objective performed best within the synthetic US domain, whereas Wave2Wave yielded superior results in the audio domain.

A critical observation pertains to the interaction between model capacity and the nature of the data split. As shown in Figure 5.12, for aggregated splits, performance generally improves with increasing model size. However, this trend does not hold for more

challenging splits such as inter-session or inter-subject evaluation. In these cases, model performance plateaus, or even declines, as model size increases. For instance, Table 5.5 illustrates that linear probing outperforms full fine-tuning in the inter-subject setting, suggesting that large models may overfit to the training set and fail to generalize when faced with significant domain shifts. This indicates that model expressiveness alone is insufficient to overcome the inherent variability in raw US data across sessions and subjects.

Based on the findings of this study, several promising directions for future research emerge. A foundational limitation of this work, is the lack of large-scale, diverse raw US datasets. Future efforts should prioritize the acquisition of extensive unlabeled data from wearable ultrasound devices, covering a wide range of subjects, activities, and hardware configurations. Such a dataset would provide a more representative basis for pretraining and could enhance the generalizability of learned representations.

The mixed results of current pretraining strategies suggest that alternative approaches warrant exploration. In related domains such as EEG, frequency-domain representations (e.g., spectrograms) have yielded substantial performance gains [61, 73, 60, 62, 63]. Applying similar representations to US data may enhance model performance, especially when paired with self-supervised learning strategies. In particular, contrastive learning methods, successfully used in ECG [74, 76, 77, 78, 84] and wearable sensor data [83, 85, 82], could offer improved robustness and task transferability.

Another challenge identified in this work is the heterogeneous nature of raw US data collection. In contrast to EEG and other clinical modalities, which benefit from well-established acquisition standards, US data, especially from experimental wearable devices, varies widely in quality, structure, and preprocessing. Establishing standardized protocols for acquisition and evaluation would not only enhance reproducibility but also accelerate progress by facilitating fair comparisons across studies.

All downstream evaluations in this work were performed using cross-validation, which is standard in prior literature but suboptimal for training large-scale foundation models. Given the significant computational costs, future studies would benefit from pre-defined train-test splits that enable efficient and consistent evaluation. This, however, depends on the availability of larger datasets and broader community consensus.

Finally, one of the most salient findings of this work is the detrimental effect of domain shifts, particularly those arising from device repositioning or subject variability. Our results show that pretraining alone is insufficient to address this issue. Consequently, future research should investigate strategies specifically designed to mitigate domain shifts, such as domain adaptation, invariant representation learning, or augmentation techniques tailored to the unique characteristics of US data.

In summary, this work presents a compelling case for the potential of foundation models in raw ultrasound applications, while also highlighting critical limitations and challenges. Our findings provide a roadmap for future research, emphasizing the need for better data,

more effective pretraining strategies, standardized evaluation frameworks, and targeted approaches to domain generalization. Addressing these challenges will be essential to advance the use of raw ultrasound as a reliable and scalable technology for human-centered sensing.

Chapter 7

Conclusion

This thesis presents a pioneering exploration into the application of foundation model frameworks to A-mode ultrasound (US) data. Specifically, the study implemented a self-supervised pretraining approach on a diverse corpus comprising large-scale audio data and synthetic ultrasound signals, followed by fine-tuning across several downstream tasks. This framework allowed for an in-depth investigation into the effects of varying datasets, pretraining strategies, model architectures, and fine-tuning techniques.

The empirical results revealed a nuanced landscape. While the overall benefit of self-supervised pretraining was limited, certain pretrained models demonstrated performance gains that surpassed existing state-of-the-art benchmarks. These instances suggest that, under specific conditions, foundation models can indeed enhance performance in A-mode ultrasound applications.

Notably, the capacity of the model emerged as a critical factor influencing downstream performance. The relationship between model scale and generalization across domain shifts, such as those between training and validation data, proved to be complex and non-trivial, warranting further investigation. Despite the progress made, several open questions remain. Chief among them are the identification of the most effective signal representations for A-mode US and the determination of optimal pretraining methodologies.

In summary, this thesis lays the groundwork for a new and underexplored area of research, offering key insights while also highlighting significant opportunities for further advancement. By extending the paradigm of foundation models into the domain of A-mode ultrasound, this work contributes both conceptually and empirically to the ongoing development of A-mode ultrasound for human-centered sensing.

Appendix A

Declaration of Originality

Master Thesis at the Department of Information Technology and Electrical Engineering

Autumn Semester 2024

Pau Altur Pastor

Ultrasound Foundation Model

October 7, 2024

Advisors:	Dr. Yawei Li, ETZ J78, yawli@iis.ee.ethz.ch Dr. Christoph Leitner, ETZ J69.2, christoph.leitner@iis.ee.ethz.ch Giusy Spacone, ETZ J 68.2, gspacone@iis.ee.ethz.ch
Supervisor:	Prof. Dr. Luca Benini, lbenini@iis.ee.ethz.ch

Handout:	September, 2024
Due:	see mystudies

1. Project Outline

Medical ultrasound (US) imaging is a vital diagnostic tool and has many areas of application. The raw data from US imaging, known as radio-frequency (RF) data, contains more information than US images and has valuable use cases. Although large datasets of processed US images are widely available, raw RF data remains scarce.

In a Master thesis [1] we explored the first system designed to predict raw RF data from US images. We have developed a data-driven, physically-informed model that combines deep learning (DL) techniques with numerical simulations of ultrasound waves. Specifically, we employed a TransUNet neural network architecture [2] to predict the acoustic properties of the underlying tissue. The neural network is trained using a custom synthetic dataset generated by us. To create extensive and meaningful synthetic datasets of paired RF data and US images, we proposed four different strategies. These include using geometric features from standard images in the ImageNet dataset and information from medical CT images [3].

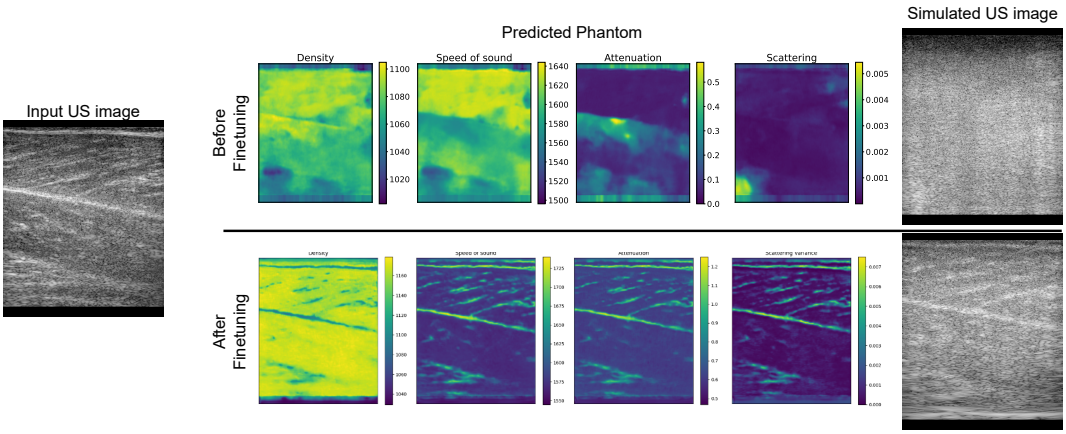


Figure 1: An example visual result, comparing the prediction of a medial gastrocnemius muscle ultrasound image before and after fine-tuning our model.

The performance of our model using different synthetic datasets was compared and evaluated for real RF data and we found that the model which generalizes best is trained on the CT phantom dataset. Additionally, we explored fine-tuning the trained models to predict RF data for medial gastrocnemius muscle images using a dedicated synthetic muscle phantom dataset created by us. We verifies that fine-tuning our model for prior-known tissue types significantly improves the predicted RF data. By comparing our physics-based model to an end-to-end DL model trained for RF data prediction, the benefit of introducing the numerical simulation into our predictions was highlighted.

In this project we build upon the results achieved by Reitsam 2024 [1] and the experience of EEG foundation model by Bucagu and Dimofte [4]. We aim to develop the first foundation model for US RF data. Foundation models have been expanding from language

and vision fields to science fields in particular biomedical signal processing. Yet, most of the AI based solutions for ultrasound signals are still limited to US images due to the scarcity of large-scale RF data. We will investigate the benefit of foundation models for tasks and applications such as gesture recognition with ultrasound arm band, hand gesture regression, regression of pennation angle using US data.

2. Research Questions

The research questions of this thesis are:

- Whether pre-training can help to improve the performance of deep models on downstream ultrasound tasks based raw-frequency data?

2.1. Methodology

- **Workpackage I - Project Setup and Initial Research:** This phase involves conducting a comprehensive literature review on ultrasound RF data and foundation models in AI, establishing a clear understanding of existing techniques, and defining the scope of the project. During this stage, the necessary computational environment is also set up, ensuring that the team has access to all required software tools, and a project plan is developed, including a risk assessment.

  Deliverables:

  - **D1:** Fine Grained Project Plan and Gantt Chart.
  - **D2:** Documentation of the literature review findings.
  - **D3:** Restructured Code Repository.

- **Workpackage II - Data Preparation, Tokenization, and Downstream Task Definition:** In this work package, the focus is on preparing with audio data or the generated RF data and developing a tokenization strategy, ensuring the data is suitable for model training. Additionally, the downstream tasks that will be used to validate the model's effectiveness, such as gesture recognition using an ultrasound armband, hand gesture regression, and pennation angle estimation, will be defined and set up. This phase ensures that both the data and the tasks are ready for the model's training and evaluation phases.

  Deliverables:

  - **D4:** Preprocessed audio data or ultrasound RF dataset.
  - **D5:** Tokenization method for ultrasound RF signals.
  - **D6:** Defined downstream tasks and datasets for gesture recognition, hand gesture regression, and pennation angle estimation.

- **Workpackage III - Model Architecture Design:** Here, the team will experiment with different deep learning architectures, such as Transformers and Mamba, adapting them to the specific needs of ultrasound RF data. This phase is iterative, as different models will be tested and refined to identify the most suitable architecture for processing RF data. The selected architecture will serve as the foundation for the rest of the project.

  Deliverables:

  - **D7:** Proposed model architecture for ultrasound RF data processing.

- **Workpackage IV - Self-Supervised Pretraining:** In the fourth work package, the emphasis is on self-supervised pretraining of the foundation model. The team will implement and apply self-supervised learning techniques to train the model on large-scale RF data. Pretraining is critical for capturing the underlying patterns in the data, and its success will be measured by the model's performance on various ultrasound signal tasks.

  Deliverables:

  - **D8:** Pretrained foundation model.
  - **D9:** Self-supervised learning implementation.

- **Workpackage V - Finetuning for Downstream Tasks:** The focus here is on applying the foundation model to tasks such as gesture recognition, finger and hand gesture regression or vital sign monitoring. Through fine-tuning, the model will be optimized for each of these tasks, and performance evaluations will be conducted to assess its effectiveness.

  - **D10:** Fine-tuned models for each downstream task.
  - **D11:** Evaluation report of model performance on downstream tasks.

- **Workpackage VI - Report and Presentation:** Work on the final report and thesis presentation.

  Deliverables:

  - **D11:** Final report summarizing the entire project and clean and documented code repository.
  - **D12:** Prepared materials and presentation for the thesis run-through and defense.

3. Project Realization

3.1. Project Plan

Within the first week of the project you will be asked to prepare a project plan. This plan should identify the tasks to be performed during the project and sets deadlines for those tasks. The prepared plan will be a topic of discussion of the first week's meeting between you and your advisers. Note that the project plan should be updated constantly depending on the project's status.

3.2. Meetings

Weekly meetings and reports must be held. The exact time and location of these meetings will be determined within the first week of the project in order to fit the student's and the assistant's schedule. These meetings will be used to evaluate the status and progress of the project. Besides these regular meetings, additional meetings can be organized to address urgent issues as well.

3.3. Report

Documentation is an important and often overlooked aspect of engineering. One final report has to be completed within this project. The common language of engineering is de facto English. Therefore, the final report of the work is preferred to be written in English. Any form of word processing software is allowed for writing the reports, nevertheless, the use of LaTeX with Tgif[1] or any other vector drawing software (for block diagrams) is strongly encouraged by the IIS staff.

Code Repository As many of our projects are heavily code-based, the documentation of the code and its repository is also considered important in the grading of your thesis. We suggest to follow coding standards and style guides when writing code, e.g. C [5], Python [6],...

Final Report The final report has to be presented at the end of the project and a digital copy need to be handed in. Note that this task description is part of your report and has to be attached to your final report. Our LaTeX template can be found here for download: https://iis-projects.ee.ethz.ch/index.php?title=Final_Report)

[1] See: http://bourbon.usc.edu:8001/tgif/index.html and http://www.dz.ee.ethz.ch/en/information/how-to/drawing-schematics.html.

3.4. Presentation

There will be a presentation (15 min for the semester thesis, and 20 min for the MS thesis presentation followed by 5 min Q&A) at the end of this project in order to present your results to a wider audience. The exact date will be determined towards the end of the work. Additional tipps for preparing the presentation can be found here: https://iis-projects.ee.ethz.ch/index.php?title=Final_Presentation

Place and Date Zurich 08/10/24 Signature Student _____

References

[1] M. Reitsma, "Ultrasound Image Recycling: Predicting Raw Data with Simulation-Driven Deep Learning," Zurich, May 2024.

[2] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation," Feb. 2021, arXiv:2102.04306 [cs]. [Online]. Available: http://arxiv.org/abs/2102.04306

[3] H. Edgar, S. Daneshvari Berry, E. Moes, N. Adolphi, P. Bridges, and K. Nolte, "New Mexico Decedent Image Database (NMDID)," publisher: [object Object]. [Online]. Available: http://nmdid.unm.edu/

[4] G. A. Bucagu and A. Dimofte, "Foundation models for EEG signal analysis," Zurich, 2024.

[5] Recommended C Style and Coding Standards. [Online]. Available: https://www.doc.ic.ac.uk/lab/cplus/cstyle.html

[6] PEP 8 – Style Guide for Python Code | peps.python.org. [Online]. Available: https://peps.python.org/pep-0008/

IIS - Digital Circuits and Systems Group
Semester/Master theses grading help sheet

Project: **Assistant** _____
Students: **Date** _____

Grade Aspects		Grading Modifiers							Grade Suggestion		
		Worse than Expected			GOOD	Better than expected			min	suggested	max
		-50	-25	-10		10	25	50			
Difficulty of project	Not all projects have the same difficulty. Sometimes during the course of the project, additional unexpected problems come up making the project much more challenging than originally intended. In other cases, projects may have to be simplified significantly during the course of the project to enable students to have at least some results	tasks were repeatedly simplified during project, extremely simple	Tasks had to be simplified during project, very simple	Simpler than usual	Well balanced project	More involved than usual	Very difficult		-50		25
Effort	A master thesis is a full time effort (40 hours per week), and a semester thesis is supposed to be a half-time work (20 hours/week). Some students spend significantly more (or less) time on the projects. This part is for their effort regardless of their efficiency and quality of results (which is separately handled)	Student was mostly absent	Did not spend enough time on the project	Spent less time than necessary	Student spent the required time for the project	Spend much more time on the project than expected	Student was constantly working on the project		-50		25
In/Dependence	A certain amount of guidance is expected in every project, and the students are expected to contribute to the project. This part rates deviations from the expectations, students that needed too much supervision, would not be able to continue without significant help.		had to be constantly told what to do	needed more supervison than expected	Some guidance was given, but student worked independently	Student worked completely independently			-25		10
Interest/Initiative	It is expected that students are engaged with the project, and contribute on their own accord to further the project. In some projects, the students are hard to motivate to work for the project and do very little if at all if they are not explicitly told to do by the supervisors.		showed very little interest in the project	was difficult to motivate for the project	Student was engaged with project and showed initiative	Made major additions to the project from own initiative			-25		10
Methodology	General approach to solving problems. Students are expected to use a systematic approach	Student did not follow any methdology at all	Larger flaws in the approach to the problem	not all aspects of the work was done systematically	The work was done systematically	The appraoch of the student was excellent, much better than what was expected			-50		10
Team Work	How did the student perform while working together with the assistants and other students. Were there major frictions and problems?		major problems working together as a team	problems in working together as a team	Student showed good team working skills	Was exceptionally good in team work			-25		10
Learning Effect	Some students posses less than average skills at the beginning of the project, however theya re able to learn and improve themselves during the project. This is meant to reward these students that improved more than expected during the course of the project.				Student showed development throughout the project	Remarkable development during the project	Extraordinary improvement during the course of the project		0		25
Results	A major part of the evaluation is what results were achieved. Note that, good results do not necessarily mean that the goals of the project were reached. For example it is a very good result if the work conclusively demonstrates that approach A is not suitable for realizing X. The quality of the results should be judged on how good they can be used further. The results are not good, if they are not substantiated, they can not help us in further projects and are unreliable. So the pratical value of the results are also a consideration, a PCB that can not be used due to design flaws is a bad result.	Project did not produce any results that can be used, results are demonstrably inferior to what has been achieved before	Weak results, the results can not be reliably used in other projects	Problems with some of the results, worse than expectations	Project achieved all its goals, and yielded good results	Project achieved much better results than was expected	Very good results	Breakthrough results, the project resulted in major new findings that can used further	-50		50
Documentation	Good documentation is an essential part of the project. It should contain, problem description, state of the artdescribing what has been done, and explaining why this is not good, list alternative approaches, justify the selected approach, explain implementation, describe the measurements, present results, compare to state of the art and provide a conclusion.	Documentation is unusable, missing major parts	Documentation is missing proper explanations for larger parts, it is not easy to follow what has been done	Incomplete/sloppy documentation, missing parts or insufficient explanations	Good Documentation, concepts are explained clearly and can be re-used for future works	Very good documentation, much better than what is expected	Excellent documentation, could be used as an example of how to write a report		-50		25
Presentation	Presentation exposes the project to the others. Good presentations should be rewarded		Very bad presentation, student unable to show what has been done	Problematic presentation, unable to conway the work/theory/results properly	Good Presentation	Excellent presentation			-25		10
Publications	If the results of the projects are very good and the student project is contributing to the publications this should be rewarded. This should not be dependent on the acceptance of the paper, if the assistants see it good enough to be submitted it should be good enough for a reward				There are no direct plans to use the results in a scientific paper	Student is contributing to the results of a paper	Student is actively participating in the preparation of a scientific paper		0		25
					5.00			**GRADE**	1.50	**5.00**	7.25

Appendix B

Declaration of Originality

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. **In consultation with the supervisor**, one of the following two options must be selected:

☐ I hereby declare that I authored the work in question independently, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies[1].

x I hereby declare that I authored the work in question independently. In doing so I only used the authorised aids, which included suggestions from the supervisor regarding language and content and generative artificial intelligence technologies. The use of the latter and the respective source declarations proceeded in consultation with the supervisor.

Title of paper or thesis:

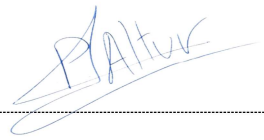
Foundation Models for Raw Ultrasound Data

Authored by:
If the work was compiled in a group, the names of all authors are required.

Last name(s):	**First name(s):**
Altur Pastor	Pau

With my signature I confirm the following:
- I have adhered to the rules set out in the Citation Guidelines.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date	**Signature(s)**
Zurich, 25/04/2023	

If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

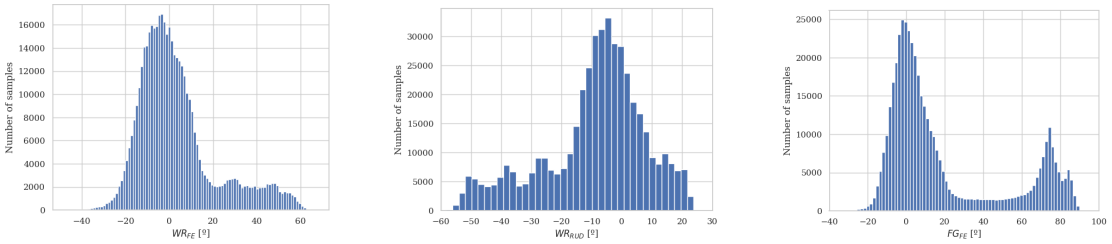
[1] For further information please consult the ETH Zurich websites, e.g. https://ethz.ch/en/the-eth-zurich/education/ai-in-education.html and https://library.ethz.ch/en/researching-and-publishing/scientific-writing-at-eth-zurich.html (subject to change).

Appendix C

Dataset statistics

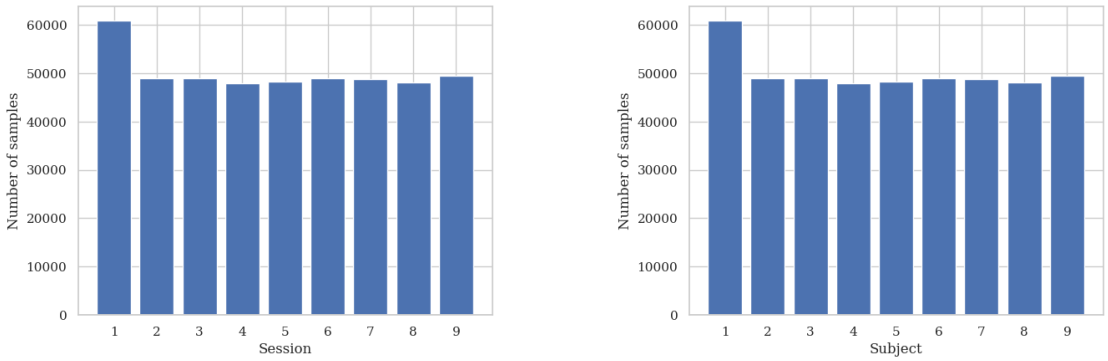
C.0.1. Hand and wrist tracking dataset (HWT)

Figures C.1 and C.2 illustrate the distribution of samples on the HWT dataset.



(a) (b) (c)

Figure C.1.: Distribution of regression targets a) WR_{FE}, b) WR_{RUD}, and c) FG_{FE} of the HWT dataset.



(a) (b)

Figure C.2.: Distribution of metadata values a) session and b) subject of the HWT dataset.

C.0.2. Hand and wrist classification dataset (HWC)

Figures C.3 and C.4 illustrate the distribution of samples on the HWC dataset.

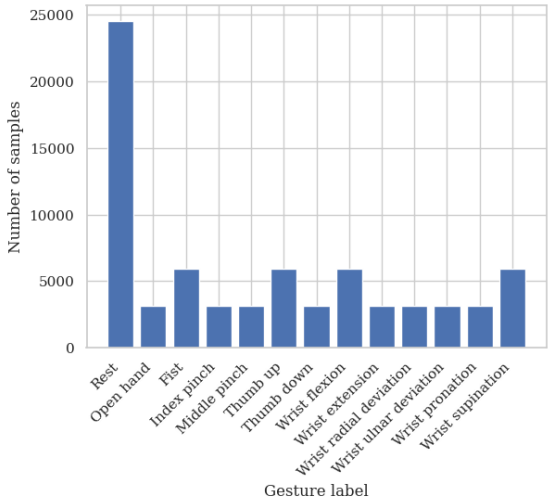


Figure C.3.: Sample distribution of gesture labels on the HWC dataset.

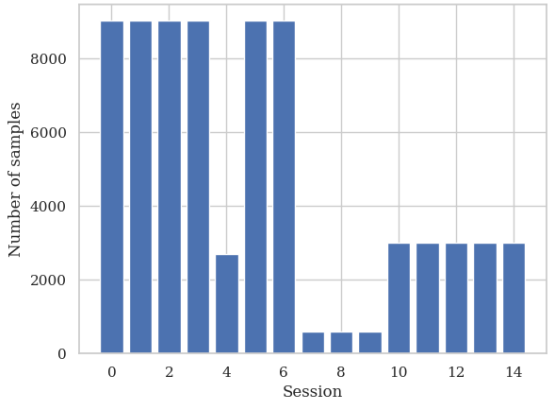


Figure C.4.: Sample distribution of sessions on the HWC dataset.

C.0.3. Muscle contraction classification dataset (MCC)

Figures C.5, C.6, and C.7 illustrate the distribution of samples on the MCC dataset.

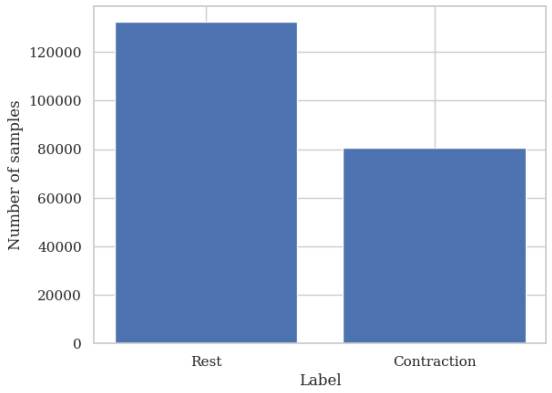


Figure C.5.: Sample distribution of contraction labels on the MCC dataset.

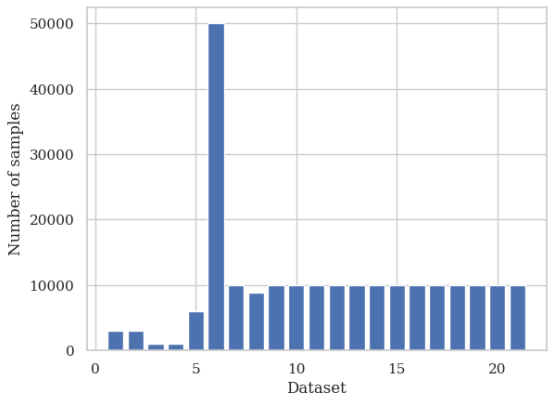


Figure C.6.: Sample distribution of datasets (recording sessions) on the MCC dataset.

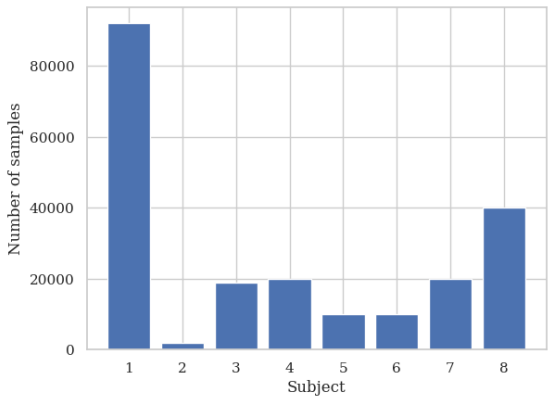


Figure C.7.: Sample distribution of subjects on the MCC dataset.

Appendix D

Preprocessing details

D.1. Pretraining datasets

D.1.1. Freesound

The Freesound dataset is first prepared for pretraining by splitting each audio clip into samples of fixed length 1000.

During data loading, each sample is Min-Max normalized by,

$$X_{norm} = \left(\frac{X - min(X)}{max(X) - min(X)} - 0.5\right) * 2$$

such that all values are between -1 and 1.

When the signal envelope is required for pretraining each sample is,

- Min-max normalized as before.
- Filtered with a 4th order Butterworth filter with $f_c = [0.1, 2000]$ Hz.
- Hilbert transformed with the scipy implementation.

D.1.2. Multi-modal conversations in smart glasses (MMCSG)

The MMCSG dataset is first prepared for pretraining by splitting each audio clip into samples of fixed length 1024.

During data loading, each sample is Min-Max normalized by,

$$X_{norm} = \left(\frac{X - min(X)}{max(X) - min(X)} - 0.5 \right) * 2$$

such that all values are between -1 and 1. The normalization is global across channels.

When the signal envelope is required for pretraining each sample is,

- Min-max normalized as before.
- Filtered with a 4th order Butterworth filter with $f_c = [0.1, 2000]$ Hz.
- Hilbert transformed with the scipy implementation.

D.1.3. USDataRecycler

The USDataRecycler dataset is prepared for pretraining by subsampling the number of channels in each sample from 192 to 8. This is done in a randomized non-overlapping manner.

During data loading, each sample is Min-Max normalized by,

$$X_{norm} = \left(\frac{X - min(X)}{max(X) - min(X)} - 0.5 \right) * 2$$

such that all values are between -1 and 1. The normalization is global across channels.

When the signal envelope is required for pretraining each sample is,

- Min-max normalized as before.
- Filtered with a 4th order Butterworth filter with $f_c = [1, 15]$ MHz.
- Hilbert transformed with the Scipy implementation.

D.2. Finetuning datasets

D.2.1. Hand and wrist tracking (HWT)

During fine-tuning the HWT samples are,

- Min-max normalized as explained in the previous sections.
- If needed their envelope is computed as explained in previous sections by filtering the data with a 4th order Butterworth filter with $f_c = [1, 3.5]$ MHz and then computing the envelope with Scipy's `Hilbert` function.

D.2.2. Hand and wrist classification (HWC)

During fine-tuning the HWC samples are,

- Min-max normalized as explained in the previous sections.
- If needed their envelope is computed as explained in previous sections by filtering the data with a 4th order Butterworth filter with $f_c = [1, 3.5]$ MHz and then computing the envelope with Scipy's `Hilbert` function.

D.2.3. Muscle contraction classification (MCC)

During fine-tuning the HWC samples are,

- Min-max normalized as explained in the previous sections.
- If needed their envelope is computed as explained in previous sections by filtering the data with a 4th order Butterworth filter with $f_c = [0.5, 6.5]$ MHz and then computing the envelope with Scipy's `Hilbert` function.

Appendix E

Training recipes

Table E.1.: Pretraining hyperparameters.

Parameter Name	Parameter Value
Batch size	512
Scheduler	Cosine + Linear Warmup
Optimizer	AdamW
Betas	[0.9, 0.98]
Weight decay	0.05
Peak learning rate	$1.25e-3$
Minimum learning rate	$2.5e-7$
Total epochs	100
Warmup epochs	5
Layer-wise learning rate decay factor	0.75
Drop path	0.2
Loss	$\mathcal{L}_2$

Table E.2.: Finetuning hyperparameters.

Parameter Name	Parameter Value
Batch size	1024
Scheduler	Cosine + Linear Warmup
Optimizer	AdamW
Betas	[0.9, 0.999]
Weight decay	0.05
Peak learning rate	$5e-4$
Minimum learning rate	$5e-7$
Total epochs	100
Warmup epochs	3
Masking ratio	0.5
Encoder layers	8
Number of attention heads	12
Embedding dimension	192
Patch size	16
Loss	$\mathcal{L}_2$

Appendix F

What did not work

This appendix outlines several experimental directions and implementation efforts that were ultimately unsuccessful or not pursued further due to limited relevance to the core objectives of the project.

- An attempt was made to pretrain the encoder proposed by Vostrikov et al. [40] on the HWT dataset, with the goal of establishing a baseline for comparison with our models. However, the model exhibited extremely poor convergence, and further optimization was deemed out of scope for the primary focus of this work.
- A transformer-based architecture was initially implemented from scratch for self-supervised pretraining. However, empirical performance was inferior to that of the existing TimeFM implementation, which was subsequently adopted and adapted for our experiments.
- Difficulties arose with checkpoint loading due to modifications in the load_from_checkpoint method within the PyTorch Lightning tasks. Specifically, the method had been altered to load only the model weights, excluding the model head. This was resolved by updating the method to support optional loading of the head via a configurable flag.
- During hyperparameter sweeps involving the embedding dimension and patch size, a large number of SLURM jobs were inadvertently launched due to misuse of batch arrays. This was later mitigated through GPU oversubscription strategies to more efficiently utilize the computational resources on each HPC node.
- A pretraining strategy using filtered audio data was explored, with the intent of matching its relative frequency content more closely to that of ultrasound signals. However, aggressive filtering eliminated much of the informative signal content, resulting in poor pretraining performance and ineffective representations.

- Fine-tuning was attempted on the GRAWUS dataset, but the performance was found to be inferior to a baseline fully connected network trained directly on the signal envelope. As a result, this direction was not pursued further.
- Fine-tuning was also attempted on the MPAE dataset. However, the regression outputs collapsed to a single constant value across all samples. This behavior is suspected to have resulted from issues such as ReLU activation death or other training instabilities. Due to time constraints, a detailed investigation could not be conducted.

Bibliography

[1] S. Frey, S. Vostrikov, L. Benini, and A. Cossettini, "WULPUS: a Wearable Ultra Low-Power Ultrasound probe for multi-day monitoring of carotid artery and muscle activity," in *2022 IEEE International Ultrasonics Symposium (IUS)*, Oct. 2022, pp. 1–4, iSSN: 1948-5727. [Online]. Available: https://ieeexplore.ieee.org/document/9958156

[2] S. Vostrikov, M. Anderegg, C. Leitner, L. Benini, and A. Cossettini, "Hand Gesture Recognition via Wearable Ultra-Low Power Ultrasound and Gradient-Boosted Tree Classifiers," in *2023 IEEE International Ultrasonics Symposium (IUS)*. Montreal, QC, Canada: IEEE, Sep. 2023, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/10307059/

[3] G. Spacone, S. Vostrikov, V. Kartsch, S. Benatti, L. Benini, and A. Cossettini, "Tracking of Wrist and Hand Kinematics with Ultra Low Power Wearable A-mode Ultrasound," *IEEE Transactions on Biomedical Circuits and Systems*, pp. 1–13, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10685090/

[4] A. Thieme, A. Nori, M. Ghassemi, R. Bommasani, T. O. Andersen, and E. Luger, "Foundation Models in Healthcare: Opportunities, Risks & Strategies Forward," in *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*. Hamburg Germany: ACM, Apr. 2023, pp. 1–4. [Online]. Available: https://dl.acm.org/doi/10.1145/3544549.3583177

[5] D. Purves, G. J. Augustine, D. Fitzpatrick, L. C. Katz, A.-S. LaMantia, J. O. McNamara, and S. M. Williams, *Neuroscience*, 2nd ed. Sinauer Associates, 2001. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK10924/

[6] P. Hoskins, K. Martin, and A. Thrush, Eds., *Diagnostic Ultrasound, Third Edition: Physics and Equipment*, 3rd ed. Boca Raton: CRC Press, Apr. 2019.

[7] L. Oláh, "Ultrasound principles," in *Manual of Neurosonology*, C. Baracchini and L. Csiba, Eds. Cambridge: Cambridge University Press, 2016, pp. 1–14. [Online].

Available: https://www.cambridge.org/core/books/manual-of-neurosonology/ultrasound-principles/ACA78734C35917857038E06573E66969

[8] P. L. P. Allan and M. J. Weston, *Clinical Ultrasound*. Churchill Livingstone, 2011, google-Books-ID: 6xTlzAEACAAJ.

[9] X. Yang, C. Castellini, D. Farina, and H. Liu, "Ultrasound as a Neurorobotic Interface: A Review," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 6, pp. 3534–3546, Jun. 2024, conference Name: IEEE Transactions on Systems, Man, and Cybernetics: Systems. [Online]. Available: https://ieeexplore.ieee.org/document/10436655

[10] D. Neumann and E. Kollorz, "Ultrasound," in *Medical Imaging Systems: An Introductory Guide*, A. Maier, S. Steidl, V. Christlein, and J. Hornegger, Eds. Cham: Springer International Publishing, 2018, pp. 237–249. [Online]. Available: https://doi.org/10.1007/978-3-319-96520-8_11

[11] H. Feigenbaum, "Role of M-mode Technique in Today's Echocardiography," *Journal of the American Society of Echocardiography*, vol. 23, no. 3, pp. 240–257, Mar. 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0894731710000581

[12] V. Protopappas, D. Baga, D. Fotiadis, A. Likas, A. Papachristos, and K. Malizos, "An ultrasound wearable system for the monitoring and acceleration of fracture healing in long bones," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 9, pp. 1597–1608, Sep. 2005. [Online]. Available: https://ieeexplore.ieee.org/document/1495704

[13] A. Lanata, E. P. Scilingo, R. Francesconi, G. Varone, and D. De Rossi, "New Ultrasound-Based Wearable System for Cardiac Monitoring," in *2006 IEEE SENSORS*, Oct. 2006, pp. 489–492, iSSN: 1930-0395. [Online]. Available: https://ieeexplore.ieee.org/document/4178665

[14] D. K. Piech, J. E. Kay, B. E. Boser, and M. M. Maharbiz, "Rodent wearable ultrasound system for wireless neural recording," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2017, pp. 221–225, iSSN: 1558-4615. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8036802

[15] E. Tarbox, N. Akhlaghi, A. S. Dhawan, B. Mukherjee, P. Gammell, P. Chitnis, and S. Sikdar, "Low-power ultrasound imaging systems using time delay spectrometry," in *2017 IEEE International Ultrasonics Symposium (IUS)*, Sep. 2017, pp. 1–4, iSSN: 1948-5727. [Online]. Available: https://ieeexplore.ieee.org/document/8092322

[16] X. Yang, Z. Chen, N. Hettiarachchi, J. Yan, and H. Liu, "A Wearable Ultrasound System for Sensing Muscular Morphological Deformations," *IEEE Transactions*

on Systems, Man, and Cybernetics: Systems, vol. 51, no. 6, pp. 3370–3379, Jun. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/8760415

[17] Z. Yin, H. Chen, X. Yang, Y. Liu, N. Zhang, J. Meng, and H. Liu, "A Wearable Ultrasound Interface for Prosthetic Hand Control," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 11, pp. 5384–5393, Nov. 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9872106

[18] L. Brausch, H. Hewener, and P. Lukowicz, "Towards a wearable low-cost ultrasound device for classification of muscle activity and muscle fatigue," in *Proceedings of the 2019 ACM International Symposium on Wearable Computers*, ser. ISWC '19. New York, NY, USA: Association for Computing Machinery, Sep. 2019, pp. 20–22. [Online]. Available: https://dl.acm.org/doi/10.1145/3341163.3347749

[19] B. G. Sgambato, M. H. Hasbani, D. Y. Barsakcioglu, J. Ibáñez, A. Jakob, M. Fournelle, M.-X. Tang, and D. Farina, "High Performance Wearable Ultrasound as a Human-Machine Interface for Wrist and Hand Kinematic Tracking," *IEEE Transactions on Biomedical Engineering*, vol. 71, no. 2, pp. 484–493, Feb. 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10227550

[20] X. Sun, X. Yang, X. Zhu, and H. Liu, "Dual-Frequency Ultrasound Transducers for the Detection of Morphological Changes of Deep-Layered Muscles," *IEEE Sensors Journal*, vol. 18, no. 4, pp. 1373–1383, Feb. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8123846

[21] J. Yan, X. Yang, X. Sun, Z. Chen, and H. Liu, "A Lightweight Ultrasound Probe for Wearable Human–Machine Interfaces," *IEEE Sensors Journal*, vol. 19, no. 14, pp. 5895–5903, Jul. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8667816

[22] I. AlMohimeed, H. Turkistani, and Y. Ono, "Development of wearable and flexible ultrasonic sensor for skeletal muscle monitoring," in *2013 IEEE International Ultrasonics Symposium (IUS)*, Jul. 2013, pp. 1137–1140, iSSN: 1051-0117. [Online]. Available: https://ieeexplore.ieee.org/document/6725109/

[23] I. AlMohimeed and Y. Ono, "Ultrasound Measurement of Skeletal Muscle Contractile Parameters Using Flexible and Wearable Single-Element Ultrasonic Sensor," *Sensors*, vol. 20, no. 13, p. 3616, Jun. 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/13/3616

[24] S. Steinberg, A. Huang, Y. Ono, and S. Rajan, "Continuous Artery Monitoring Using a Flexible and Wearable Single-Element Ultrasonic Sensor," *IEEE Instrumentation & Measurement Magazine*, vol. 25, no. 1, pp. 6–11, Feb. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9693453/

[25] K. Keller, C. Leitner, C. Baumgartner, L. Benini, and F. Greco, "Fully Printed Flexible Ultrasound Transducer for Medical Applications," *Advanced Materials Technologies*, vol. 8, no. 18, p. 2300577, 2023, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/admt.202300577. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/admt.202300577

[26] M. Giordano, K. Keller, F. Greco, L. Benini, M. Magno, and C. Leitner, "Towards a Novel Ultrasound System Based on Low-Frequency Feature Extraction From a Fully-Printed Flexible Transducer," in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Oct. 2023, pp. 1–5, iSSN: 2766-4465. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10388792

[27] Y. Huang, X. Yang, Y. Li, D. Zhou, K. He, and H. Liu, "Ultrasound-Based Sensing Models for Finger Motion Classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1395–1405, Sep. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8082493/

[28] N. Akhlaghi, C. A. Baker, M. Lahlou, H. Zafar, K. G. Murthy, H. S. Rangwala, J. Kosecka, W. M. Joiner, J. J. Pancrazio, and S. Sikdar, "Real-Time Classification of Hand Motions Using Ultrasound Imaging of Forearm Muscles," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 8, pp. 1687–1698, Aug. 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7320970/

[29] A. J. Fernandes, Y. Ono, and E. Ukwatta, "Evaluation of Finger Flexion Classification at Reduced Lateral Spatial Resolutions of Ultrasound," *IEEE Access*, vol. 9, pp. 24 105–24 118, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9344619

[30] X. Yang, X. Sun, D. Zhou, Y. Li, and H. Liu, "Towards Wearable A-Mode Ultrasound Sensing for Real-Time Finger Motion Recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 6, pp. 1199–1208, Jun. 2018, conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering. [Online]. Available: https://ieeexplore.ieee.org/document/8347147

[31] J. He, H. Luo, J. Jia, J. T. W. Yeow, and N. Jiang, "Wrist and Finger Gesture Recognition With Single-Element Ultrasound Signals: A Comparison With Single-Channel Surface Electromyogram," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 5, pp. 1277–1284, May 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8477106/

[32] X. Yang, J. Yan, and H. Liu, "Comparative Analysis of Wearable A-Mode Ultrasound and sEMG for Muscle-Computer Interface," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 9, pp. 2434–2442, Sep. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8943971/

[33] W. Xia, Y. Zhou, X. Yang, K. He, and H. Liu, "Toward Portable Hybrid Surface Electromyography/A-Mode Ultrasound Sensing for Human–Machine Interface," *IEEE Sensors Journal*, vol. 19, no. 13, pp. 5219–5228, Jul. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8662605/

[34] X. Yang, J. Yan, Z. Chen, H. Ding, and H. Liu, "A Proportional Pattern Recognition Control Scheme for Wearable A-mode Ultrasound Sensing," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 1, pp. 800–808, Jan. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8654210/

[35] L. Guo, Z. Lu, L. Yao, and S. Cai, "A Gesture Recognition Strategy Based on A-Mode Ultrasound for Identifying Known and Unknown Gestures," *IEEE Sensors Journal*, vol. 22, no. 11, pp. 10 730–10 739, Jun. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9760285/

[36] S. Cai, Z. Lu, B. Chen, L. Guo, Z. Qing, and L. Yao, "Dynamic Gesture Recognition of A-Mode Ultrasonic Based on the DTW Algorithm," *IEEE Sensors Journal*, vol. 22, no. 18, pp. 17 924–17 931, Sep. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9847605/

[37] J. Zeng, Y. Zhou, Y. Yang, J. Yan, and H. Liu, "Fatigue-Sensitivity Comparison of sEMG and A-Mode Ultrasound based Hand Gesture Recognition," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 4, pp. 1718–1725, Apr. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9585400/

[38] X. Yang, Y. Liu, Z. Yin, P. Wang, P. Deng, Z. Zhao, and H. Liu, "Simultaneous Prediction of Wrist and Hand Motions via Wearable Ultrasound Sensing for Natural Control of Hand Prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 2517–2527, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853615

[39] J. Zeng, Y. Zhou, Y. Yang, J. Wang, and H. Liu, "Feature Fusion of sEMG and Ultrasound Signals in Hand Gesture Recognition," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2020, pp. 3911–3916, iSSN: 2577-1655. [Online]. Available: https://ieeexplore.ieee.org/document/9282818/

[40] S. Vostrikov, M. Anderegg, L. Benini, and A. Cossettini, "Unsupervised Feature Extraction From Raw Data for Gesture Recognition With Wearable Ultralow-Power Ultrasound," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 71, no. 7, pp. 831–841, Jul. 2024, conference Name: IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10538295

[41] J.-Y. Guo, Y.-P. Zheng, Q.-H. Huang, and X. Chen, "Dynamic monitoring of forearm muscles using one-dimensional sonomyography system," *Journal of Rehabilitation Research and Development*, vol. 45, no. 1, pp. 187–195, 2008.

[42] J.-Y. Guo, Y.-P. Zheng, Q.-H. Huang, X. Chen, J.-F. He, and H. Lai-Wa Chan, "Performances of One-Dimensional Sonomyography and Surface Electromyography in Tracking Guided Patterns of Wrist Extension," *Ultrasound in Medicine & Biology*, vol. 35, no. 6, pp. 894–902, Jun. 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301562908005826

[43] J.-Y. Guo, Y.-P. Zheng, L. P. Kenney, A. Bowen, D. Howard, and J. J. Canderle, "A Comparative Evaluation of Sonomyography, Electromyography, Force, and Wrist Angle in a Discrete Tracking Task," *Ultrasound in Medicine & Biology*, vol. 37, no. 6, pp. 884–891, Jun. 2011. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0301562911001566

[44] Y. Zou, L. Cheng, and Z. Li, "A Multimodal Fusion Model for Estimating Human Hand Force: Comparing surface electromyography and ultrasound signals," *IEEE Robotics & Automation Magazine*, vol. 29, no. 4, pp. 10–24, Dec. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9794325/

[45] X. Yang, J. Yan, Z. Yin, and H. Liu, "Sonomyographic Prosthetic Interaction: Online Simultaneous and Proportional Control of Wrist and Hand Motions Using Semisupervised Learning," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 2, pp. 804–813, Apr. 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9903068

[46] R. Murray, J. Mendez, L. Gabert, N. P. Fey, H. Liu, and T. Lenzi, "Ambulation Mode Classification of Individuals with Transfemoral Amputation through A-Mode Sonomyography and Convolutional Neural Networks," *Sensors (Basel, Switzerland)*, vol. 22, no. 23, p. 9350, Dec. 2022. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9736589/

[47] J. Mendez, R. Murray, L. Gabert, N. P. Fey, H. Liu, and T. Lenzi, "A-Mode Ultrasound-Based Prediction of Transfemoral Amputee Prosthesis Walking Kinematics via an Artificial Neural Network," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 1511–1520, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10052730/

[48] S. Hafthorsdottir, S. Vostrikov, A. Cossettini, M. Rieder, C. Leitner, M. Magno, and L. Benini, "Automatic Extraction of Muscle Fascicle Pennation Angle from Raw Ultrasound Data," in *2022 IEEE Sensors Applications Symposium (SAS)*, Aug. 2022, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/9881341/?arnumber=9881341

[49] S. Vostrikov, T. M. Ingolfsson, S. Hafthorsdottir, C. Leitner, M. Magno, L. Benini, and A. Cosettini, "A Muscle Pennation Angle Estimation Framework From Raw Ultrasound Data for Wearable Biomedical Instrumentation," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10329945

[50] M. Giordano, C. Leitner, C. Vogt, L. Benini, and M. Magno, "PuLsE: Accurate and Robust Ultrasound-based Continuous Heart-Rate Monitoring on a Wrist-Worn IoT Device," Oct. 2024, arXiv:2410.16219 [eess]. [Online]. Available: http://arxiv.org/abs/2410.16219

[51] M. Lin, Z. Zhang, X. Gao, Y. Bian, R. S. Wu, G. Park, Z. Lou, Z. Zhang, X. Xu, X. Chen, A. Kang, X. Yang, W. Yue, L. Yin, C. Wang, B. Qi, S. Zhou, H. Hu, H. Huang, M. Li, Y. Gu, J. Mu, A. Yang, A. Yaghi, Y. Chen, Y. Lei, C. Lu, R. Wang, J. Wang, S. Xiang, E. B. Kistler, N. Vasconcelos, and S. Xu, "A fully integrated wearable ultrasound system to monitor deep tissues in moving subjects," *Nature Biotechnology*, vol. 42, no. 3, pp. 448–457, Mar. 2024, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41587-023-01800-0

[52] S. Frey, V. Kartsch, C. Leitner, A. Cossettini, S. Vostrikov, S. Benatti, and L. Benini, "A Wearable Ultra-Low-Power sEMG-Triggered Ultrasound System for Long-Term Muscle Activity Monitoring," Sep. 2023, arXiv:2309.06851 [eess]. [Online]. Available: http://arxiv.org/abs/2309.06851

[53] S. Vostrikov, L. Benini, and A. Cossettini, "Complete Cardiorespiratory Monitoring via Wearable Ultra Low Power Ultrasound," in *2023 IEEE International Ultrasonics Symposium (IUS)*, Sep. 2023, pp. 1–4, iSSN: 1948-5727. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10307398

[54] J.- S. Kenny, C. E. Munding, J. K. Eibl, A. M. Eibl, B. F. Long, A. Boyes, J. Yin, P. Verrecchia, M. Parrotta, R. Gatzke, P. A. Magnin, P. N. Burns, F. S. Foster, and C. E. M. Demore, "A novel, hands-free ultrasound patch for continuous monitoring of quantitative Doppler in the carotid artery," *Scientific Reports*, vol. 11, no. 1, p. 7780, Apr. 2021, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41598-021-87116-y

[55] P. G. v. Leuteren, A. J. Nieuwhof-Leppink, and P. Dik, "SENS-U: clinical evaluation of a full-bladder notification – a pilot study," *Journal of Pediatric Urology*, vol. 15, no. 4, pp. 381.e1–381.e5, Aug. 2019, publisher: Elsevier. [Online]. Available: https://www.jpurol.com/article/S1477-5131(19)30081-6/fulltext

[56] M. N. Mohsenvand, M. R. Izadi, and P. Maes, "Contrastive Representation Learning for Electroencephalogram Classification," in *Proceedings of the Machine Learning for Health NeurIPS Workshop*, ser. Proceedings of Machine Learning Research, E. Alsentzer, M. B. A. McDermott, F. Falck, S. K. Sarkar, S. Roy, and S. L. Hyland, Eds., vol. 136. PMLR, Dec. 2020, pp. 238–253. [Online]. Available: https://proceedings.mlr.press/v136/mohsenvand20a.html

[57] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," Jul. 2020, arXiv:2002.05709 [cs]. [Online]. Available: http://arxiv.org/abs/2002.05709

[58] D. Kostas, S. Aroca-Ouellette, and F. Rudzicz, "BENDR: Using Transformers and a Contrastive Self-Supervised Learning Task to Learn From Massive Amounts of EEG Data," *Frontiers in Human Neuroscience*, vol. 15, Jun. 2021, publisher: Frontiers. [Online]. Available: https://www.frontiersin.orghttps://www.frontiersin.org/journals/human-neuroscience/articles/10.3389/fnhum.2021.653659/full

[59] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," Oct. 2020, arXiv:2006.11477 [cs]. [Online]. Available: http://arxiv.org/abs/2006.11477

[60] C. Yang, M. B. Westover, and J. Sun, "BIOT: Cross-data Biosignal Learning in the Wild," May 2023, arXiv:2305.10351 [eess]. [Online]. Available: http://arxiv.org/abs/2305.10351

[61] C. Wang, V. Subramaniam, A. U. Yaari, G. Kreiman, B. Katz, I. Cases, and A. Barbu, "BrainBERT: Self-supervised representation learning for intracranial recordings," Feb. 2023, arXiv:2302.14367 [cs, eess, q-bio]. [Online]. Available: http://arxiv.org/abs/2302.14367

[62] D. Zhang, Z. Yuan, Y. YANG, J. Chen, J. Wang, and Y. Li, "Brant: Foundation Model for Intracranial Neural Signal," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 26 304–26 321. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/535915d26859036410b0533804cee788-Paper-Conference.pdf

[63] Z. Yuan, D. Zhang, J. Chen, G. Gu, and Y. Yang, "Brant-2: Foundation Model for Brain Signals," Mar. 2024, arXiv:2402.10251 [q-bio] version: 3. [Online]. Available: http://arxiv.org/abs/2402.10251

[64] Y. Chen, K. Ren, K. Song, Y. Wang, Y. Wang, D. Li, and L. Qiu, "EEGFormer: Towards Transferable and Interpretable Large-Scale EEG Foundation Model," Jan. 2024, arXiv:2401.10278 [eess]. [Online]. Available: http://arxiv.org/abs/2401.10278

[65] W.-B. Jiang, L.-M. Zhao, and B.-L. Lu, "Large Brain Model for Learning Generic Representations with Tremendous EEG Data in BCI," May 2024, arXiv:2405.18765 [cs]. [Online]. Available: http://arxiv.org/abs/2405.18765

[66] A. v. d. Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," in *arXiv.org*. Long Beach, CA, USA: arXiv.org, Nov. 2017. [Online]. Available: https://arxiv.org/abs/1711.00937v2

[67] 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," in *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, May 2024, pp. 1–5, iSSN: 1945-8452. [Online]. Available: https://ieeexplore.ieee.org/document/10635453

[68] S. Panchavati, C. Arnold, and W. Speier, "Mentality: A Mamba-Based Approach Towards Foundation Models for EEG," in *International Conference for Learning Representations*, 2024.

[69] A. Gu and T. Dao, "Mamba: Linear-Time Sequence Modeling with Selective State Spaces," May 2024, arXiv:2312.00752 [cs]. [Online]. Available: http://arxiv.org/abs/2312.00752

[70] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," May 2015, arXiv:1505.04597 [cs]. [Online]. Available: http://arxiv.org/abs/1505.04597

[71] R. Hussein, S. Lee, and R. K. Ward, "Multi-Channel Vision Transformer for Epileptic Seizure Prediction," *Biomedicines*, Jun. 2022, publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://open.library.ubc.ca/soa/cIRcle/collections/facultyresearchandpublications/52383/items/1.0422018

[72] N. M. Foumani, G. Mackellar, S. Ghane, S. Irtza, N. Nguyen, and M. Salehi, "EEG2Rep: Enhancing Self-supervised EEG Representation Through Informative Masked Inputs," Jun. 2024, arXiv:2402.17772 [eess]. [Online]. Available: http://arxiv.org/abs/2402.17772

[73] 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," Jan. 2025, arXiv:2501.10885 [cs]. [Online]. Available: http://arxiv.org/abs/2501.10885

[74] J. Y. Cheng, H. Goh, K. Dogrusoz, O. Tuzel, and E. Azemi, "Subject-Aware Contrastive Learning for Biosignals," Jun. 2020, arXiv:2007.04871 [cs]. [Online]. Available: http://arxiv.org/abs/2007.04871

[75] B. Gopal, R. Han, G. Raghupathi, A. Ng, G. Tison, and P. Rajpurkar, "3KG: Contrastive Learning of 12-Lead Electrocardiograms using Physiologically-Inspired Augmentations," in *Proceedings of Machine Learning for Health*, ser. Proceedings of Machine Learning Research, S. Roy, S. Pfohl, E. Rocheteau, G. A. Tadesse, L. Oala, F. Falck, Y. Zhou, L. Shen, G. Zamzmi, P. Mugambi, A. Zirikly, M. B. A. McDermott, and E. Alsentzer, Eds., vol. 158. PMLR, Dec. 2021, pp. 156–167. [Online]. Available: https://proceedings.mlr.press/v158/gopal21a.html

[76] D. Kiyasseh, T. Zhu, and D. A. Clifton, "CLOCS: Contrastive Learning of Cardiac Signals Across Space, Time, and Patients," May 2021, arXiv:2005.13249 [cs]. [Online]. Available: http://arxiv.org/abs/2005.13249

[77] P. Sarkar and A. Etemad, "Self-supervised ECG Representation Learning for Emotion Recognition," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1541–1554, Jul. 2022, arXiv:2002.03898 [eess]. [Online]. Available: http://arxiv.org/abs/2002.03898

[78] T. Mehari and N. Strodthoff, "Self-supervised representation learning from 12-lead ECG data," *Computers in Biology and Medicine*, vol. 141, p. 105114, Feb. 2022, arXiv:2103.12676 [eess]. [Online]. Available: http://arxiv.org/abs/2103.12676

[79] S. Rabbani and N. Khan, "Contrastive Self-Supervised Learning for Stress Detection from ECG Data," *Bioengineering*, vol. 9, no. 8, p. 374, Aug. 2022, number: 8 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2306-5354/9/8/374

[80] J. Lai, H. Tan, J. Wang, L. Ji, J. Guo, B. Han, Y. Shi, Q. Feng, and W. Yang, "Practical intelligent diagnostic algorithm for wearable 12-lead ECG via self-supervised learning on large-scale dataset," *Nature Communications*, vol. 14, no. 1, p. 3741, Jun. 2023, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41467-023-39472-8

[81] C. B. Kumar, A. K. Mondal, M. Bhatia, B. K. Panigrahi, and T. K. Gandhi, "Self-Supervised Representation Learning-Based OSA Detection Method Using Single-Channel ECG Signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–15, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10082890

[82] S. Abbaspourazad, O. Elachqar, A. C. Miller, S. Emrani, U. Nallasamy, and I. Shapiro, "Large-scale Training of Foundation Models for Wearable Biosignals," Mar. 2024, arXiv:2312.05409 [cs]. [Online]. Available: http://arxiv.org/abs/2312.05409

[83] H. T. Hallgrímsson, F. Jankovic, T. Althoff, and L. Foschini, "Learning Individualized Cardiovascular Responses from Large-scale Wearable Sensors Data," Dec. 2018, arXiv:1812.01696 [cs]. [Online]. Available: http://arxiv.org/abs/1812.01696

[84] X. Wu, C. Huang, P. Roblesgranda, and N. Chawla, "Representation Learning on Variable Length and Incomplete Wearable-Sensory Time Series," May 2020, arXiv:2002.03595 [eess]. [Online]. Available: http://arxiv.org/abs/2002.03595

[85] D. Spathis, I. Perez-Pozuelo, S. Brage, N. J. Wareham, and C. Mascolo, "Self-supervised transfer learning of physiological representations from free-living wearable data," in *Proceedings of the Conference on Health, Inference, and Learning*, Apr. 2021, pp. 69–78, arXiv:2011.12121 [eess]. [Online]. Available: http://arxiv.org/abs/2011.12121

[86] C. I. Tang, I. Perez-Pozuelo, D. Spathis, and C. Mascolo, "Exploring Contrastive Learning in Human Activity Recognition for Healthcare," Feb. 2021, arXiv:2011.11542 [cs]. [Online]. Available: http://arxiv.org/abs/2011.11542

[87] H. Yuan, T. Plekhanova, R. Walmsley, A. C. Reynolds, K. J. Maddison, M. Bucan, P. Gehrman, A. Rowlands, D. W. Ray, D. Bennett, J. McVeigh, L. Straker, P. Eastwood, S. D. Kyle, and A. Doherty, "Self-supervised learning

of accelerometer data provides new insights for sleep and its association with mortality," Jul. 2023, pages: 2023.07.07.23292251. [Online]. Available: https://www.medrxiv.org/content/10.1101/2023.07.07.23292251v1

[88] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling Laws for Neural Language Models," Jan. 2020, arXiv:2001.08361 [cs, stat]. [Online]. Available: http://arxiv.org/abs/2001.08361

[89] F. Font, G. Roma, and X. Serra, "Freesound technical demo," in *Proceedings of the 21st ACM international conference on Multimedia*, ser. MM '13. New York, NY, USA: Association for Computing Machinery, Oct. 2013, pp. 411–412. [Online]. Available: https://dl.acm.org/doi/10.1145/2502081.2502245

[90] Meta, "Multi-Modal Conversations in Smart Glasses dataset," 2024. [Online]. Available: https://www.chimechallenge.org/challenges/chime8/task3/data

[91] B. E. Treeby and B. T. Cox, "k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields," *Journal of Biomedical Optics*, vol. 15, no. 2, p. 021314, Mar. 2010, publisher: SPIE. [Online]. Available: https://www.spiedigitallibrary.org/journals/journal-of-biomedical-optics/volume-15/issue-2/021314/k-Wave--MATLAB-toolbox-for-the-simulation-and-reconstruction/10.1117/1.3360308.full

[92] M. Reitsma, "Ultrasound Image Recycling: Predicting Raw Data with Simulation-Driven Deep Learning," Apr. 2024, master Thesis at IIS ETH Zürich.

[93] L. Brausch, H. Hewener, and P. Lukowicz, "Classifying Muscle States with One-Dimensional Radio-Frequency Signals from Single Element Ultrasound Transducers," *Sensors*, vol. 22, no. 7, p. 2789, Apr. 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/7/2789

[94] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," May 2016, arXiv:1409.0473 [cs]. [Online]. Available: http://arxiv.org/abs/1409.0473

[95] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Long Beach, CA, USA: Curran Associates Inc.57 Morehouse LaneRed HookNYUnited States, Apr. 2017, arXiv:1706.03762 [cs]. [Online]. Available: http://arxiv.org/abs/1706.03762

[96] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu, "SimMIM: A Simple Framework for Masked Image Modeling," Apr. 2022, arXiv:2111.09886 [cs]. [Online]. Available: http://arxiv.org/abs/2111.09886

[97] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A Time Series is Worth 64 Words: Long-term Forecasting with Transformers," Mar. 2023, arXiv:2211.14730 [cs]. [Online]. Available: http://arxiv.org/abs/2211.14730

[98] P. H. Le-Khac, G. Healy, and A. F. Smeaton, "Contrastive Representation Learning: A Framework and Review," *IEEE Access*, vol. 8, pp. 193 907–193 934, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9226466

[99] R. Zhang, Y. Ji, Y. Zhang, and R. J. Passonneau, "Contrastive Data and Learning for Natural Language Processing," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Tutorial Abstracts*, M. Ballesteros, Y. Tsvetkov, and C. O. Alm, Eds. Seattle, United States: Association for Computational Linguistics, Jul. 2022, pp. 39–47. [Online]. Available: https://aclanthology.org/2022.naacl-tutorials.6/

[100] Y. Bastanlar, S. Orhan, Y. Bastanlar, and S. Orhan, "Self-Supervised Contrastive Representation Learning in Computer Vision," in *Artificial Intelligence Annual Volume 2022*. IntechOpen, May 2022. [Online]. Available: https://www.intechopen.com/chapters/81791

[101] W. Falcon and T. P. L. Team, "PyTorch Lightning," Mar. 2019. [Online]. Available: https://github.com/Lightning-AI/pytorch-lightning

[102] O. Yadan, "Hydra - A framework for elegantly configuring complex applications," 2019. [Online]. Available: https://github.com/facebookresearch/hydra

[103] G. A. Bucagu and A. Dimofte, "Foundation Models for EEG Signal Analysis," Ph.D. dissertation, ETH Zurich, Zurich, Switzerland, Nov. 2024.

[104] T. maintainers and contributors, "TorchVision: PyTorch's Computer Vision library," 2016, publication Title: GitHub repository. [Online]. Available: https://github.com/pytorch/vision

[105] Nicki Skafte Detlefsen, Jiri Borovec, Justus Schock, Ananya Harsh, Teddy Koker, Luca Di Liello, Daniel Stancl, Changsheng Quan, Maxim Grechkin, and William Falcon, "TorchMetrics - Measuring Reproducibility in PyTorch," Feb. 2022. [Online]. Available: https://github.com/Lightning-AI/torchmetrics