ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

ARTIFICIAL INTELLIGENCE

MASTER THESIS

in

Deep Learning for Bio-signals

BIO-SIGNAL (EEG) PROCESSING FOR WITH DEEP LEARNING AT THE EDGE

CANDIDATE
Andrea Fossà

SUPERVISOR
Prof. Roberta Calegari

CO-SUPERVISORS
Michele Romani, PhD.
Dott. Ing. Elisabetta Farella

Academic year 2024-2025
Session V of March 2026

Abstract

Brain-Computer Interface (BCI) suffer from high inter-subject variability and limited labelled data, often requiring lengthy calibration phases. This work is part of a broader collaborative effort developed during an internship at Fondazione Bruno Kessler, aimed at building an end-to-end approach that explicitly models subject dependency using lightweight convolutional neural networks (CNNs) conditioned on the subject's identity. The contributions presented here focus on the design and implementation of the neural architectures, the hyper-parameter optimization pipeline ensuring robustness and reproducibility, and the interpretability analysis of the learned representations. The method integrates two conditioning mechanisms to adapt pretrained models to unseen subjects with minimal calibration data. We benchmark three lightweight architectures on a time-modulated Event-Related Potential (ERP) classification task, providing interpretable evaluation metrics and explainable visualizations. Results demonstrate improved generalization and data-efficient calibration, highlighting the scalability and practicality of subject-adaptive BCIs.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context and Motivation

BCI are a promising area of current neurotechnology. These systems can convert brain activity into digital commands, bypassing the body's natural movement pathways. Electroencephalography (EEG) has become the standard method for non-invasive BCI, mainly because of its safety, portability, low cost, and high temporal resolution [6]. This is especially important in clinical settings, for example, people with locked-in syndrome, spinal cord injuries, or amyotrophic lateral sclerosis can use BCI systems to control robotic prosthetics, computer cursors, or communication devices, which helps them regain a significant level of independence.

Despite decades of research, from the first systematic experiments conducted in the 1970s [7] to modern architectures based on deep learning, a fundamental obstacle continues to limit the practical diffusion of BCIs: **inter-subject variability**. Brain activity recorded via EEG varies substantially between different individuals, due to anatomical (cortical morphology, skull thickness, tissue conductivity), functional (individual neural activation patterns), and instrumental (electrode impedance, environmental conditions) differences. This variability means that a classification model trained on a group of subjects tends to capture the average behaviour of the population, resulting

in suboptimal performance for individuals who deviate significantly from that average.

The most burdensome practical consequence is the need for a specific calibration phase for each new user: a session during which the subject must perform predefined mental tasks so that the system can collect sufficient labelled data to adapt the classifier. This phase can take from tens of minutes to several hours, compromising the usability of the system and representing a significant barrier to both clinical adoption and consumer applications.

This work directly addresses this problem by proposing an end-to-end framework that integrates lightweight neural architectures with subject-specific conditioning mechanisms, with the aim of minimising or eliminating the calibration data needed to adapt the system to a new user, without sacrificing classification performance.

1.2 The calibration challenge

Calibration in BCI systems serves to estimate subject-specific classification model parameters, compensating for inter-individual variability in the EEG signal. There are multiple sources of this variability: neuroanatomical differences influence the propagation and recording of neural signals; neural activation patterns associated with the same cognitive or motor task can vary considerably between different individuals; moreover, an estimated 15% to 30% of the population the so-called “BCI-illiterate” subjects do not show sufficient neural modulations to effectively control a BCI system [6]. In addition to this inter-subject variability, there is significant intra-subject variability due to circadian fluctuations, states of attention and fatigue, and neuroplasticity phenomena. The neurophysiological details about the sources of variability are discussed in depth in the calibration section.

From a practical point of view, the calibration problem manifests itself as a trade-off between three competing factors:

- **Calibration time**: traditional protocols require 15-30 minute sessions, during which the subject must maintain a constant level of attention and motivation. Prolonged sessions induce mental and physical fatigue, degrading the quality of the collected data.
- **Classification performance**: models trained exclusively on data from other subjects (cross-subject approach) suffer a significant drop in performance compared to individually calibrated models, due to the domain shift between subjects.
- **Usability**: for clinical and consumer applications, calibration must be short and simple enough not to discourage the user. A system that requires a long preparatory session before each use is unlikely to be adopted in daily practice.

Recent research has addressed this problem through several strategies: transfer learning techniques, which seek to transfer knowledge from trained subjects to new users [8]; domain adaptation methods, which align data distributions across different domains; and Riemannian geometry-based approaches, which operate in the covariance matrix space by exploiting affine invariance properties [9]. Each approach has specific advantages and limitations, which are discussed in the in the state of the art chapter.

1.3 Proposed Approach

This work proposes an approach that explicitly models subject identity as a covariate of the model, integrating **subject-specific conditioning** mechanisms within lightweight convolutional architectures designed for execution on resource constrained devices (edge devices).

The fundamental insight is as follows: instead of treating subject identity as an implicit confounding variable as in conventional models it is explicitly incorporated into the inference process through embedding vectors learned

end-to-end during training. These embeddings capture the individual neural characteristics of each subject in a shared latent space, allowing the model to adapt its internal representations to the specificities of each user. The framework is designed to operate in two stages:

1. **Multi-subject pre-training**: the model is trained on the set of available subjects, simultaneously learning generalised representations of the EEG signal and embeddings specific to each subject in the training set.

2. **Fine-tuning with minimal calibration**: for a new subject, the embedding is initialised from the distribution of learned embeddings and updated with a minimal amount of calibration data on the order of a single batch of about 60 trials, corresponding to less than 30 seconds of recording.

Three lightweight architectures EEGNet [10], P300MCNN [11], and PhiNet [5] are adapted and evaluated as feature extractors within this framework on a time-modulated evoked potential (ERP) classification task in the BrainForm dataset [4].

1.4 Contributions

This thesis is part of a larger project developed during an internship at the E3DA (Energy-Efficient Embedded Digital Architectures) laboratory of the Bruno Kessler Foundation, aimed at creating an adaptive BCI system that can be deployed on embedded devices. The contributions on which is focused this thesis are:

1. **Design and implementation of neural architectures.** Three lightweight convolutional architectures (EEGNet, P300MCNN, PhiNet) were adapted and implemented for conditional classification of ERP, with particular

attention to compatibility with resource constrained devices. The architectures were integrated into a modular framework that supports the optional insertion of conditioning mechanisms.

2. **Evaluation of subject-specific conditioning mechanisms.** Two conditioning mechanisms H-projection (projection into feature space) and FiLM (Feature-wise Linear Modulation) were integrated and systematically evaluated on the three architectures, analysing the trade-off between parametric parsimony and adaptation flexibility in zero-shot conditions and with incremental fine-tuning.

3. **Robust optimisation and reproducibility.** This was achieved through a hyperparameter optimization pipeline. This pipeline, constructed with Optuna, employed a nested Leave-One-Subject-Out (LOSO) cross-validation protocol and multiple random seeds, thereby guaranteeing stable and reproducible outcomes. Consequently, the optimal configurations for each architectural design and conditioning method were determined via this systematic approach.

4. **Interpretability analysis.** A three-level interpretability framework weight topography, time-frequency analysis of activations, and visualisation of the embedding space via UMAP was developed to verify the neurophysiological plausibility of the learned representations and provide tools for qualitative model validation.

Chapter 2

Context and state of the art

This chapter presents the theoretical background and current state of research that supports this work. It begins with an overview of BCI systems and EEG. Then, it explains traditional EEG signal processing methods, P300-based interfaces, the challenges of calibration, and the differences between individuals. Following this, the chapter discusses techniques based on Riemannian geometry, the use of deep learning in EEG analysis, transfer learning, conditioning mechanisms, and finally, the issue of interpretability.

2.1 Brain-Computer Interfaces: Overview

A BCI is a system that picks and translates neural signals and processes them. into instructions to external parts, without any reference to natural channels. The operation of a BCI is divided into sequential phases: signal acquisition, preprocessing to remove artefacts and noise, extraction of relevant features, classification or decoding, and finally translation into control commands [6].

Historical background. The empirical basis of the field dates back to Hans Berger's recording of brain electrical activity in the 1920s, but the term *Brain-Computer Interface* was coined by Jacques Vidal in the 1970s, when the first

systematic experiments on the use of electroencephalographic signals to control external devices began. The 1990s marked a significant turning point with the work of Birbaumer and Wolpaw, who developed the first clinically viable BCI systems for patients with severe motor disabilities. The beginning of the 21st century saw an exponential acceleration in research, driven by advances in electronics, computer science and understanding of the human brain.

Classification by invasiveness. BCIs are classified according to their level of invasiveness. **Non-invasive** systems use sensors placed on the scalp to record brain electrical activity via EEG: they are completely safe and do not require surgery, but offer limited spatial resolution. Other non-invasive modalities include functional magnetic resonance imaging (fMRI), functional near-infrared spectroscopy (fNIRS) and magnetoencephalography (MEG). Semi-invasive systems employ epidural or subdural electrode arrays to record electrocorticography (ECoG), achieving higher spatial resolution than EEG with relatively low surgical risk. Lastly, invasive systems entail implanting electrodes directly into brain tissue, providing the highest spatial and temporal resolution but with significant surgical risks and long-term biocompatibility issues. This work focuses exclusively on non-invasive EEG-based BCIs applications.

BCI applications range from neurological rehabilitation to the control of robotic prostheses, motorised wheelchairs and augmentative communication devices. They work particularly well in the clinic with people who have spinal cord injuries, strokes, ALS, or locked-in syndrome as they have a direct means of communicating with the world without actually having to move their body muscles.

2.2 Electroencephalography and Event-Related Potentials

The brain signal most commonly used in non-invasive BCI systems is the EEG. The EEG consists of recording brain electrical activity using electrodes placed on the scalp. This signal reflects the sum of postsynaptic potentials generated mainly by cortical pyramidal cells, oriented perpendicular to the cortical surface. Since the electrical current produced by individual neurons is extremely weak, the EEG detects a collective average of the activity of sufficiently large and synchronous neuronal populations, which explains both the high sensitivity of the method and its limited spatial resolution. In other words, what we observe on the electroencephalogram is a "global reflection" of cortical dynamics, which, while sacrificing anatomical precision, retains the ability to rapidly capture functional variations related to cognitive, emotional, and motor states.

2.2.1 Electrode mounting and positioning

The standard reference system for electrode position in EEG is the International 10-20 System, developed in the 1950s and still used today. This standardization takes its name from the distances (10% or 20% of the head measurement) that is used to determine the placement of the contacts along the main anatomical lines of the head (nose, inion, and ear). The electrodes are classified with letters indicating the cortical area:

- **F** = frontal, associated with executive and higher cognitive functions;
- **C** = central, corresponding to the primary motor cortex;
- **P** = parietal, area involved in sensory integration;
- **O** = occipital, primary site of the visual cortex;
- **T** = temporal, connected to auditory and mnemonic functions.

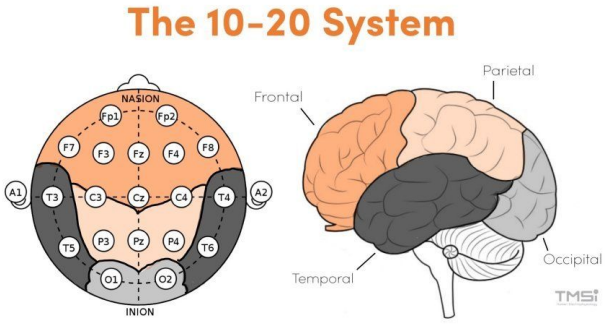


Figure 2.1: Credits: tmsi[1]. The electrode layout of the 10-20 system (left) and corresponding brain regions (right) (pictures adapted from photo and photo).

The numbering distinguishes between the left hemisphere (even) and the right hemisphere (odd), while the letter "z" indicates the median axis. Denser systems, with more electrodes, such as 10–10 or 10–5, allow for more accurate spatial sampling with a greater number of electrodes (up to 256 in high-density configurations), improving the topographical reconstruction of brain activity and enabling more refined analyses, such as source localization.

2.2.2 Frequency bands

The EEG signal is characterized by oscillations that can be broken down into different frequency bands, each of which is associated with specific neurophysiological functions and cognitive states:

- **Delta (0.5–4 Hz)**: typical of deep stages of non-REM sleep; rarely used in BCI, except for clinical applications related to sleep disorders or pathological states.
- **Theta (4–8 Hz)**: associated with mnemonic processes, attention, and meditative states; in some cognitive BCI, it is used as a marker of cognitive load or mental fatigue.
- **Alpha (8–13 Hz)**: emerges mainly in the occipital regions during states

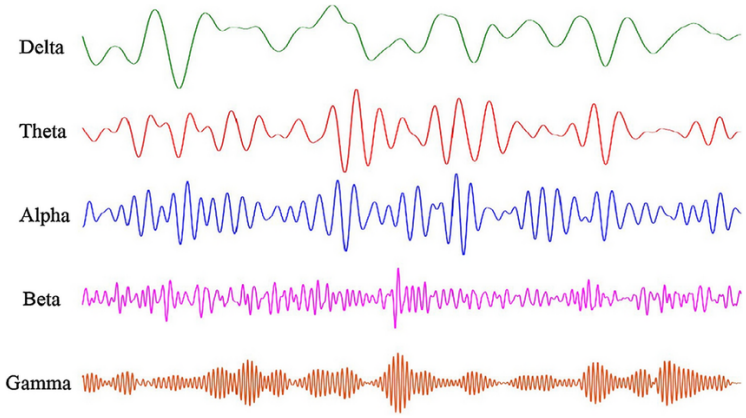


Figure 2.2: Credits:[2]. Frequency bands of EEG signals.

of relaxation with closed eyes; its suppression (event-related desynchronization, ERD) is a key indicator in motor imagery paradigms.

- **Beta (13–30 Hz)**: located in sensorimotor areas, it reflects states of cortical activation and motor processes; beta oscillations are fundamental in motor BCI for the control of prostheses and external devices.

- **Gamma (>30 Hz, up to about 100 Hz)**: related to higher cognitive functions, such as perception, sensory binding, and working memory; more difficult to detect with surface EEG due to low sensitivity to high frequencies and high susceptibility to muscle noise.

The dynamics of these bands, observed through phenomena such as event-related synchronization (ERS) and event-related desynchronization (ERD), constitute one of the main mechanisms on which BCI are based. In particular, the modulation of mu (8–12 Hz, a subclass of alpha) and beta (13–30 Hz) rhythms in the central areas is the basis of motor imagery interfaces, which allow the user to control external devices by imagining simple body movements.

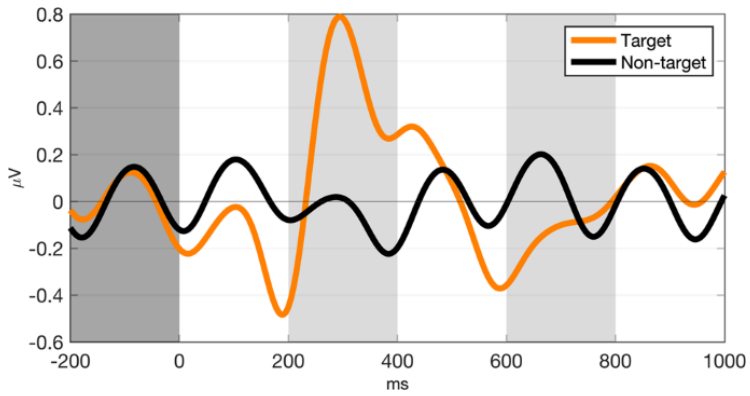


Figure 2.3: Credits:[3]. P300 speller ERP. The grand-averaged ERP response at the midline (Fz, Cz, and Pz) during P300 speller test sessions.

2.2.3 Event-related potentials: the case of P300

In addition to spontaneous oscillations, the EEG allows us to observe evoked potentials, i.e., electrical responses in the brain that occur systematically following a sensory or cognitive stimulus. Among these, the **P300** plays a central role in BCI and is a task that is very close to ours.

The P300 is a positive component of the EEG wave that typically emerges around 300 milliseconds after the presentation of a rare or relevant stimulus in a so-called "oddball" paradigm. For example, if a subject observes a sequence of flashing letters on a screen, the P300 appears at the letter that the subject is voluntarily "looking for" or paying attention to.

From a neurophysiological point of view, the P300 reflects mechanisms of selective attention and working memory updating. In BCI, this component is exploited to build so-called "speller" communication interfaces, in which the user can select symbols or letters simply by focusing their attention on them, without any muscle movement. This approach has enormous clinical relevance for patients with locked-in syndrome, as it allows them to establish a direct channel of communication with the outside world.

2.2.4 Characteristics and limitations of the EEG signal

The EEG has a number of advantages that make it an ideal tool for non-invasive BCI: it is safe, relatively inexpensive, portable, and characterized by high temporal resolution (in the order of milliseconds). Unfortunatly, it has significant limitations:

- **Low spatial resolution**: the conductivity of cranial tissues and the distance between electrodes and cortical generators attenuate and diffuse the signal, making it difficult to precisely locate neural sources.
- **Sensitivity to artifacts**: EEG signals are easily contaminated by physiological (EMG, EOG, ECG) and environmental noise.
- **Reduced amplitude**: the recorded potentials are in the order of microvolts (10–100 μV), which requires extremely sensitive amplifiers and advanced filtering techniques.

Nevertheless, EEG is the most widely used modality in BCI studies that are not performed in an invasive fashion since it allows collecting detailed and valuable information regarding brain activity in a comparably simple and safe way. The wide range of neural activities that EEG can potentially demonstrate, including both spontaneous and evoked potentials, including the P300 is what makes EEG a highly flexible instrument that has made a great contribution to the creation and sharing of BCI systems.

2.3 P300-Based Brain-Computer Interfaces

The P300 evoked potential, described in the previous section from a neurophysiological point of view, forms the basis of one of the most widespread and established classes of BCI. The basic idea is simple: if the system can

detect which stimulus elicited a P300 in the user's EEG signal, then it can infer the user's intention without the user having to make any voluntary movement. This section describes the main stimulation paradigms used in P300-based BCIs, with particular attention to the temporal modulation adopted in the BrainForm dataset used in this work.

2.3.1 The oddball paradigm and Farwell and Donchin's speller

The anomaly paradigm is the fundamental mechanism underlying P300-based BCIs. In a sequence of stimuli, most are non-target (frequent and irrelevant), while a minority are target (rare and relevant to the user's activity). The P300 component emerges selectively in response to target stimuli, as the user's brain recognises the expected stimulus and updates its representation of the context [10].

The first practical application of this principle to BCIs was proposed by Farwell and Donchin in 1988 with the so-called *P300 speller*. In this system, the letters of the alphabet are arranged in a matrix (typically 6×6) and the rows and columns are illuminated in rapid sequence. The user focuses their attention on the desired letter: when the row or column containing that letter lights up, a P300 is generated in the EEG signal. By identifying the intersection between the row and column that caused the P300, the system determines the letter selected by the user.

However, this row-column paradigm (RCP) has some limitations. In case several stimuli are activated in parallel per row or column, adjacency distraction may take place: the attention of the user is somehow held by stimuli near the target and the P300 response is smaller in magnitude and the error rate increases. Furthermore, the simultaneous illumination of multiple elements introduces ambiguities that require multiple repetitions to achieve reliable classification.

2.3.2 Single-stimulus and time-modulation paradigms

To overcome the limitations of the row-and-column paradigm, alternative approaches have been developed. In *single-stimulus* (*single-character paradigm*) paradigms, each element of the interface is highlighted individually, eliminating the ambiguity of simultaneous illumination. Although this approach reduces perceptual confusion, it involves longer stimulation times when the number of possible targets is high, as each stimulus must be presented separately.

A class particularly relevant to the present work is that of BCIs based on time-modulated evoked potentials (*time-modulated ERPs*). In these systems, visual stimuli are presented in rapid sequential succession: each target flashes individually for a short duration (e.g., 100 ms in the BrainForm protocol), and the user must focus their attention on the desired stimulus. Target/Non-Target classification occurs for each individual stimulation epoch, making the problem a binary classification task applied to each event.

The BrainForm dataset used in the experiments in this thesis adopts this paradigm with a maximum of 10 competing stimuli. The direct consequence is an intrinsic imbalance between classes: for each stimulation cycle, only one stimulus in ten is the target, producing a 1:9 ratio between Target and Non-Target epochs. This highly imbalanced distribution poses a significant challenge for training classifiers, as discussed in the section dedicated to the dataset.

2.3.3 From binary classification to BCI command

It is important to distinguish between the classification problem addressed by neural models and the overall functioning of the BCI system. The models described in this thesis operate at the single epoch level, classifying each EEG response as Target or Non-Target. However, in an operational BCI system, the final decision on which command to execute requires the aggregation of

responses over multiple stimulation cycles.

In a paradigm with K possible stimuli, the system typically repeats the stimulation sequence for R repetitions, accumulating the posterior probabilities for each stimulus. The stimulus with the highest cumulative probability is selected as the command. The number of repetitions R represents a trade-off between communication speed and accuracy: a higher number of repetitions improves classification reliability but reduces the system's interaction speed, typically measured in bits per minute (*bit rate*).

This two-level architecture binary classification at the single epoch level and aggregation at the command level implies that even classifiers with moderate accuracy at the single epoch level can produce reliable BCI systems, provided that the number of repetitions is sufficient. Therefore, the goal of optimisation is not only to maximise binary classification accuracy, but to find the balance between classifier performance and the number of repetitions needed to achieve an acceptable level of reliability for the end user.

2.3.4 Specific challenges of P300-based BCIs

P300-based BCIs present specific challenges that distinguish them from other paradigms, such as motor imagery-based interfaces or steady-state evoked potentials (SSVEPs):

Class imbalance is intrinsic to the paradigm: the Target/Non-Target ratio is determined by the number of stimuli in the interface and cannot be modified without altering the stimulation protocol. Loss function-based strategies, such as Focal Loss or weighted BCE, prove to be more appropriate than explicit oversampling, as demonstrated in the experimental results of this thesis.

The *variability of P300 morphology* between subjects is particularly pronounced: latency, amplitude, and topographic distribution of the component

vary considerably depending on age, experience with BCI systems, and individual neuroanatomical characteristics. This variability motivates the adoption of subject-specific conditioning mechanisms, which constitute the central contribution of this work.

Finally, the low amplitude of the P300 (typically 5-20 μV) compared to the background noise of the EEG makes it necessary to use accurate filtering techniques and relatively long time epochs (300-600 ms post-stimulus) to capture the entire dynamics of the evoked response. The choice of the optimal time window is in fact one of the most influential hyperparameters, as emerged from the optimisation with Optuna described in Chapter 5.

2.4 The Calibration Problem in BCIs

One of the main obstacles to the clinical and consumer diffusion of BCI systems is the need for a subject specific calibration phase before each use. This phase, during which the subject must perform specific mental tasks to allow the system to collect labelled data, can take from tens of minutes to several hours, compromising the practical usability of the system.

Calibration serves to estimate subject specific classification model parameters, compensating for the inter-individual variability discussed above. Minimising the amount of calibration data required, while maintaining acceptable performance, is therefore a central goal in modern BCI research [8].

2.4.1 Neurophysiological Origins of Variability

The variability of the EEG signal between subjects is a phenomenon that has its roots in individual biology. Anatomical differences in cortical morphology, skull thickness, and brain tissue conductivity significantly influence the propagation and recording of neural signals [8]. From a functional perspective, the neural activation patterns associated with specific motor or cognitive

tasks can vary considerably between individuals: the peak frequency of sensorimotor rhythms, for example, typically ranges between 8 and 12 Hz but can differ by several Hz between subjects, while the amplitude and reactivity of these rhythms show inter-individual variations that can exceed an order of magnitude.

The morphology of event-related potentials (ERP), such as the P300, also varies between subjects in terms of latency, amplitude, and topographic distribution. These parameters are influenced by individual factors such as age, level of attention, previous experience with BCI systems, and the cognitive characteristics of the subject [12].

2.4.2 The phenomenon of BCI illiteracy

A particularly relevant aspect of inter-subject variability is the so called phenomenon of "BCI illiteracy": an estimated 15% to 30% of the population is unable to effectively control an EEG-based BCI system, regardless of the amount of training received [6]. These subjects, defined as "BCI illiterate," do not show the expected modulation of sensory-motor rhythms during motor imagery, or produce evoked potentials of insufficient amplitude to be classified reliably.

The causes of this phenomenon are still the subject of active research and include factors such as neuroanatomy, differences in individual motor imagery mechanisms, and characteristics of cognitive personality. The presence of BCI illiterate subjects in evaluation datasets introduces a source of heterogeneity that makes fair comparison between different systems difficult and can significantly lower the reported average performance.

2.4.3 Intra-subject variability

In addition to variability between individuals, the EEG signal of the same subject exhibits significant fluctuations over time. These variations occur on different time scales: circadian oscillations linked to natural biological rhythms, short term variations related to attention and alertness, changes due to the accumulation of mental and physical fatigue during prolonged sessions, and long term modifications associated with neuroplasticity and learning of the BCI paradigm [13].

The emotional and motivational state of the subject can significantly alter the modulation of sensorimotor rhythms, making BCI sessions highly dependent on the psychophysiological conditions of the moment. This makes it extremely difficult to design systems that maintain stable performance over time without continuous adaptation mechanisms.

Neuroplasticity introduces an additional level of complexity: although neural adaptation can improve BCI performance in the long term, it also causes *drift* in signal characteristics that require periodic updates of decoding algorithms. This phenomenon is particularly evident in the first weeks of using a BCI system, when the user's brain adapts to the system's demands and, at the same time, the system must adapt to neural changes.

2.4.4 Impact on model design

The combination of inter-subject and intra-subject variability has profound implications for the design of deep learning-based BCI systems. Models trained on a group of subjects tend to model the average behaviour of the population, losing sensitivity to individual specificities. On the other hand, models trained on a single subject require large amounts of labeled data for that subjecta condition that is difficult to satisfy in clinical and consumer scenariosand do not benefit from the information available from other subjects.

This has motivated the development of intermediate approaches, such as

transfer learning and subject specific conditioning, which seek to combine the advantages of generalized knowledge learned on many subjects with the ability to adapt to individual specificities with minimal amounts of calibration data [8].

2.5 Traditional Methods for EEG Signal Classification

For decades, EEG signal processing in BCI systems has been based on traditional machine learning pipelines, characterised by a clear separation between the feature extraction (feature engineering) and classification phases. The canonical approach involves: (i) preprocessing the raw signal to remove artefacts and noise, (ii) extraction of manually designed features, such as spectral power in specific frequency bands, signal variance, or spatial projections of neural patterns, and (iii) classification using algorithms such as Linear Discriminant Analysis (LDA), Support Vector Machines (SVM), or Bayesian classifiers [6].

These classical methods have important practical advantages: they are interpretable, require modest amounts of labelled data, and achieve competitive performance in many standard scenarios. However, these pipelines suffer from a fundamental limitation: the quality of the extracted features depends on the designer's experience and ability to a priori encode the relevant signal structures. Manual feature engineering is laborious, difficult to generalise across different paradigms, and potentially insufficient to capture the wealth of information contained in the EEG signal.

2.5.1 Common Spatial Patterns (CSP)

The Common Spatial Patterns (CSP) method is a fundamental algorithm in the classification of EEG signals in motor imagery experiments. CSP determines

linear spatial filters that maximise the variance of the signal in one mental condition and minimise it in the opposite condition, deriving discriminative features from the spatial projections of the EEG signal.

Formally, given an EEG signal $\mathbf{X} \in R^{C \times T}$, CSP searches for a projection matrix $\mathbf{W}$ such that $\mathbf{Z} = \mathbf{W}^\top \mathbf{X}$ maximises the ratio between the variance of the projection for the two classes. This reduces to a problem of simultaneous diagonalisation of the covariance matrices of the two classes [6].

Although it does well when used in the context of training and testing on the same individual, CSP is poor at generalizing to unknown individuals due to its high dependence on the cortical structure of each subject. Numerous variants have been proposed to improve cross-subject robustness, including regularised CSP and covariance-based approaches [6].

2.5.2 Methods based on xDAWN and subspace decomposition

For paradigms based on ERP, such as P300, a particularly effective approach is the xDAWN algorithm, designed specifically to improve the signal-to-noise ratio of evoked potentials [6]. xDAWN identifies spatial filters that maximise the signal-to-noise ratio of evoked potentials by projecting the EEG signal into a subspace where stimulus-related components are emphasised and background noise is suppressed.

Combined with Riemannian geometry, typically through the covariantisation of the xDAWN output and subsequent classification in the space of positive definite symmetric matrices (SPD), this approach is still one of the most competitive pipelines for P300 detection, especially in cross-subject scenarios with scarce calibration data [14].

2.6 Deep Learning for EEG

The advent of deep learning has opened up new perspectives in the automatic analysis of biomedical signals. Deep neural networks hierarchically learn representations directly from raw data, eliminating the need to manually design features. In the EEG domain, this translates into the ability to jointly learn spatial filters which combine the contributions of different electrodes in a manner analogous to *Common Spatial Patterns* (CSP) methods and temporal filters that capture relevant dynamics at different temporal scales [10].

Convolutional neural networks (CNNs), in particular, have established themselves as the reference architecture for EEG signal analysis, thanks to their ability to exploit the local structure of the signal, both in the temporal and spatial (channels) dimensions. Unlike fully connected architectures, CNNs share filter parameters between different temporal and spatial locations, significantly reducing the number of parameters and limiting the risk of overfitting, which is particularly feared with EEG datasets that are typically small in size.

More recently, architectures based on self-attention mechanisms and Transformers [15] have also shown promising results in neural signal analysis, thanks to their ability to model long-range dependencies in the time sequence. However, their high computational cost limits their use in real-time Edge AI and BCI contexts, making lightweight CNNs the preferred choice for applications on embedded devices.

The application of deep learning to EEG signals presents unique challenges that distinguish it from other domains such as computer vision or natural language processing . First, EEG datasets are typically small: collecting annotated data is expensive, requiring the collaboration of human subjects and prolonged experimental sessions. This limits the ability of deep networks to generalise, making them prone to *overfitting*.

Second, the EEG signal is highly non-stationary: its statistical characteristics vary over time due to physiological fluctuations, mental fatigue, variations in electrode impedance, and changes in the subject's cognitive state . This non-stationarity makes it difficult to train robust models that maintain their performance throughout a session or between distinct sessions.

Finally, inter-subject variability is perhaps the most critical challenge for the generalisation of deep learning models in the context of BCIs. Unlike image recognition, where the same object generates similar visual patterns for any observer, the neural patterns associated with the same cognitive or motor task vary substantially between different individuals, making the transfer of knowledge between subjects an open problem that is still far from a definitive solution.

These challenges collectively motivate the adoption of compact architectures with few trainable parameters and a fundamental step toward designing efficient neural networks for resource constrained devices, that we used in this thesis, was the introduction of depthwise separable convolutions, introduced with MobileNet architectures [16] for mobile computer vision. These operations factorize a standard convolution into two steps: a depthwise convolution that applies a separate filter for each input channel, followed by a pointwise convolution (1×1) that linearly combines the resulting feature maps.

Formally, given an input feature map $\mathbf{F} \in R^{C_{in} \times H \times W}$:

$$\text{Depthwise:} \quad \mathbf{F'}_c = \mathbf{K}_c * \mathbf{F}_c, \quad c = 1, \ldots, C_{in} \tag{2.1}$$

$$\text{Pointwise:} \quad \mathbf{F'}_c = \sum_{c=1}^{C_{in}} \mathbf{W}_{c',c} \, \mathbf{F'}_c, \quad c' = 1, \ldots, C_{out} \tag{2.2}$$

The computational cost of a standard convolution with filter $D_K \times D_K$, C_{in} input channels, and C_{out} output channels is proportional to $D_K^2 \cdot C_{in} \cdot C_{out}$, while that of a depthwise-separable convolution is proportional to $D_K^2 \cdot C_{in} +$

$C_{in} \cdot C_{out}$, with a computational saving of approximately $1/C_{out} + 1/D_K^2$ compared to the standard version [16]. This cost reduction is particularly advantageous for architectures intended for deployment on microcontrollers and edge devices with severely limited computational and energy resources.

In the context of EEG-based BCIs, depthwise separable convolutions have taken on a key role in the architecture of networks such as EEGNet [10], where they allow for the decoupled learning of temporal filters (which capture the spectro temporal dynamics of the signal) and spatial filters (which combine the contributions of different electrodes), while maintaining an extremely low number of parameters.

2.7 Transfer Learning and Domain Adaptation

The literature proposes numerous domain adaptation techniques aimed at aligning data distributions across different domains (subjects, sessions, acquisition conditions), reducing the *domain shift* that degrades the performance of cross-subject models.

Euclidean alignment (EA) rescales and centres the covariances EEG of each subject towards a common reference matrix, reducing systematic differences between subjects before applying the model [8]. Methods based on *Adversarial Domain Adaptation* train the network to produce subject-invariant representations, making it difficult for a subject discriminator to distinguish the origin of the data.

More recently, approaches based on *meta-learning* such as Model-Agnostic Meta-Learning (MAML) have shown promising results in cross-subject BCI, training the model to initialise at a point in parameter space from which it can quickly adapt to a new subject with few gradient steps.

2.8 Conditional Neural Networks and Feature Modulation

Subject specific conditioning is a strategy that aims to integrate information about the subject's identity directly into the neural network's inference process, allowing the model to adapt its internal representations to the individual characteristics of each user. This approach differs from simple individual fine tuning which requires separate training for each subject in that conditioning operates within a unified framework, where generalised knowledge is preserved and individual specificities are captured through dedicated parameters.

The need for conditioning arises directly from the limitations of subject-agnostic approaches: models trained on a heterogeneous pool of subjects tend to capture the "average" variance of neural patterns, resulting in suboptimal performance for subjects that deviate from the population average. Conditioning offers a principled mechanism for incorporating the identity of the subject as an explicit covariate of the model, rather than treating it as an implicit confounding factor.

2.9 Interpretability and XAI in BCIs

2.9.1 The need for interpretability

The use of deep learning models in BCI systems, especially in clinical contexts , raises important questions of interpretability. The "black box" nature of deep neural networks makes it difficult to understand the mechanisms through which the model reaches its decisions, limiting the confidence of clinicians and users in the system and hindering the identification of potential systematic errors or biases in the training data.

Explainable AI (XAI) offers tools to make deep learning models more transparent and understandable, providing explanations of predictions in terms

that are interpretable by domain experts. In the context of EEG analysis, this typically translates into the ability to identify which channels, frequency bands, or time intervals contributed most to the model's decision, verifying whether the model's focus coincides with neurophysiological expectations [11].

2.9.2 Topographical Analysis of Weights

A simple but effective approach to analysing interpretability in EEG models is to examine the weights of the first convolutional layer, which is directly exposed to the raw signal and therefore provides a direct indication of the relevance of the channels to the model. The relevance of channel c can be quantified as the energy of its weights:

$$I_c = \sum_{d,k} w_{c,d,k}^2$$

where d represents the number of filters and k the time index. By normalising these values and visualising them on a topographic map of the head, it is possible to identify which cortical regions the model considers most informative for the task under consideration.

2.9.3 Time-Frequency Analysis of Activations

To complement the weight-based analysis, the activation of convolutional filters can be analysed in the time-frequency domain using the Morlet transform 2.4. For each filter of the convolutional layer analysed, the responses to Target and Non-Target epochs are converted into spectrograms, revealing the oscillatory patterns emphasised by the architecture in the early stages of processing.

This analysis allows us to verify whether the learned filters correspond to frequency bands that are physiologically relevant to the task under consideration, and to compare the differences between the responses to Target and

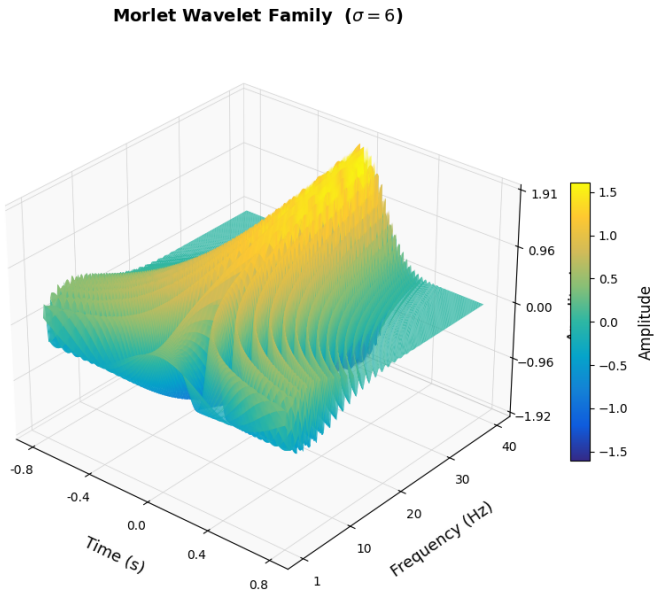


Figure 2.4: Real part of the Morlet wavelet family as a function of time and central frequency $f_0 \in [1, 40]$ Hz, with shape parameter $\sigma = 6$. Increasing f_0 produces wavelets with narrower temporal support and higher oscillation rate, reflecting the time-frequency uncertainty principle inherent to wavelet analysis.

Non-Target stimuli, providing a deeper understanding of how the model differentiates between the two categories.

Chapter 3

Dataset and Experimental Setup

3.1 The Brainform EEG Dataset

The BrainForm dataset used in those experiments was collected through an experimental protocol based on a game designed for training and evaluating reactive BCI based on event-related potentials (ERPs) [4]. The protocol was designed to combine an engaging gaming environment with structured and controlled collection of EEG signals and interaction metadata, allowing for analysis of both the evolution of BCI control skills and system performance under repeated and comparable conditions. The experiment involved a total of 22 subjects, all of whom were experienced computer users but without prior experience with BCI systems. Each participant took part in two main sessions, each of which included a calibration phase and a sequence of game runs. Each calibration phase consisted of 60 trials, during which the subject had to focus their attention on a specific flashing visual stimulus in order to collect sufficient data for training a subject-dependent ERP classifier. Following calibration, the subjects performed a series of game runs, including an initial tutorial and complete tasks designed to evaluate BCI control in scenarios with different levels of complexity. The protocol involved the execution of two distinct tasks, characterized by a maximum number of 10 competing stimuli. The first task, more oriented towards game dynamics, required a timed and

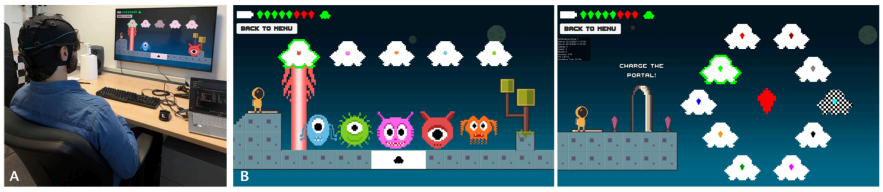


Figure 3.1: Credits: [4]; A) A participant taking part into the BrainForm experiment. B) Left: an example of the Complex Task; right: an example of the Speller Task. The color of the target is represented by the small symbol in the middle of the alien ships. The green outline provides real-time feedback on the spelled symbol. Targets flash sequentially for 100ms each.

coordinated selection of a correct stimulus in relation to the visual context. The second task, a speller task, was more directly related to the evaluation of BCI performance, as it required the accurate selection of symbols from a set of flashing visual stimuli. The two tasks were integrated into the game and designed to be done in real time using BCI control alone. In addition to this two main tasks, 16 subjects chose to complete an additional free-play session, including a new calibration phase and an additional game run. In this study, all optional calibration sessions were included to enrich the training set, while they were excluded from the leave-out fold used in the evaluation in order to ensure a fair and unbiased comparison of the test results. EEG data were recorded using a g.tec Unicorn system with conductive gel, equipped with eight electrodes positioned according to the international 10-20 system (Fz, C3, Cz, C4, Pz, PO7, Oz, PO8), with reference and ground electrodes positioned on the mastoids. Signals were acquired at a sampling frequency of 250 Hz. Throughout the sessions, both raw EEG signals and event triggers synchronized with the presentation of visual stimuli were recorded, allowing for subsequent accurate data segmentation. The data was then pre processed to enhance the signal-to-noise ratio for our task.

3.2 Signal Preprocessing Pipeline

Pre-processing is an essential step in the EEG data. In this work we employed a dedicated pre-processing pipeline that can be described in five steps and was implemented using MNE-Python (the standard Python library for this task):

Step 1, Bandpass filtering: A bandpass filter is applied between 2 Hz (high pass) and 15 Hz (low pass). The high-pass filter removes slow drifts and baseline variations; the low-pass filter removes high-frequency noise and, above all, muscle artefacts. The P300 has a peak energy in the 0.5-10 Hz range, making this filtering appropriate.

Step 2, Downsampling: An optional decimation step reduces the sampling frequency by a factor of resampling. With the default setting of 2.0, the signal is downsampled from 250 Hz to 125 Hz. This halves the temporal axis (from 151 to 76 samples per epoch), reducing memory requirements and computational cost while preserving the frequency content relevant to the P300.

Step 3, Epoch extraction: Time-synchronised epochs are extracted from the continuous signal relative to the onset of the stimulus:

Pre-stimulus baseline: from -100 ms to 0 ms (pre-event = 0.1 s) Post-stimulus window: from 0 to 500 milliseconds (post-event = 0.5 seconds) The resulting epoch length is the sum of the pre-event and post-event multiplied by the sampling frequency. It makes a total of 150 samples (plus the initial sample). After downsampling by 2 the signal reaches 76 samples. Baseline correction is applied by subtracting the average of the pre-stimulus period.

Step 4, Recalibration: Raw EEG values are multiplied by 10^6 to convert volts to microvolts, improving numerical stability during training.

Step 5, Normalization: One of the following normalisation methods was used depending on the optimization strategy:

- ChannelwiseStandardScaler (default): Z-score normalisation calculated

independently for each EEG channel. This removes average channel-level offsets and variance differences that may result from impedance variations.

- SafeMinMaxScaler: min-max normalisation per channel to [0, 1], with protection against channels with zero range.
- RobustScaler: normalisation using the median and interquartile range, more resistant to amplitude outliers.
- ChannelWiseRobustScaler: robust normalisation applied independently for each channel.

All scalers are adapted only to the training set and applied identically to the validation, tuning, and test sets, preventing data leakage.

3.3 Epoch Extraction and Class Imbalance

ERP-based BCI paradigms inherently generate highly unbalanced datasets. In a paradigm with K possible stimuli, only one corresponds to the user's desired target. In the specific case of the BrainForm dataset with 10 competing stimuli, this produces a ratio of 1:9 between target and non-target epochs (60 versus 540 epochs per session). Without adequate measures, models tend to classify all observations as non-targets to artificially maximise overall accuracy, completely failing to detect targets.

To address this imbalance, three strategies are evaluated:

- **Baseline (No resampling):** The imbalance is handled implicitly by choosing metrics (MCC, AUC) that are not biased by class frequency.
- **Random Oversampling:** The minority class (Target) is upsampled by randomly duplicating existing samples via `imbalanced-learn`.
- **Random Undersampling:** The majority class (Non-Target) is downsampled by randomly removing samples.

- **Loss functions:** We choose and implemented loss functions that are designed to address class imbalance.

To prevent the data leakage, resampling is applied only to the training set after the train/validation split.

3.4 Data Splits in LOSO Evaluation

To assess the ability of each model to generalise to unseen subjects, we adopt a Leave-One-Subject-Out (LOSO) cross-validation scheme: in each fold, one subject is held out for fine-tuning and testing, while all remaining subjects are used for training. For each fold of the LOSO evaluation with test subject test id: Training set: All sessions (0, 1, and optionally 2) of all subjects except test id. A 92%/8% random train/validation split is applied. Validation set: 8% of the training set, used for early stopping. Fine-tuning set and Test set: The two sessions of the held-out subject are combined and temporally divided into a fine-tuning set and a test set. To counterbalance potential fatigue effects, the split is performed in an interleaved manner: the first half of each session is merged with the second half of the other session, while preserving the original class-label distribution. Subject embedding indices are assigned as consecutive integers: subject 0 through N-1 for training subjects, and index N for the test subject. This ensures that the test subject's embedding can be initialized from the training embeddings via mean or geometric median interpolation.

Chapter 4

Model Architectures

Consider a dataset composed of N subjects, formally defined as:

$$S := \{s_i\}_{i=1}^{N},$$

where each s_i represents a unique identifier for the i-th subject.

For each subject, we have access to a set of EEG epochs, each represented as:

$$\mathbf{X} \in R^{C \times T},$$

where:

- C denotes the number of EEG channels (electrodes),
- T denotes the number of temporal samples in the epoch.

Each epoch is associated with a binary label $y \in \{0, 1\}$, where:

$$y = 0 \quad \Rightarrow \quad \text{Target (T)}, \qquad y = 1 \quad \Rightarrow \quad \text{Non-Target (NT)}.$$

In the time-modulated ERP paradigm, similar to the classical P300 paradigm, the dataset is highly imbalanced. The calibration phase exhibits a label distribution of one Target every nine Non-Targets. Formally, if N_T denotes the

number of Target epochs and N_{NT} the number of Non-Target epochs, we have:

$$\frac{N_T}{N_{NT}} = \frac{1}{9}.$$

This imbalance reflects the fact that in a typical BCI application with multiple possible stimuli, only one option corresponds to the user's intended target, while all others are non-targets.

Learning problem. The classification task can be formalized as learning a function:

$$f_\theta : R^{C\times T} \times S \to [0, 1],$$

that maps an EEG epoch $\mathbf{X}$ and the subject identity s_i to the probability that the epoch corresponds to a Target stimulus.

The learning objective is to optimize the parameters θ by minimizing a suitable loss function $\mathcal{L}$ on the training distribution:

$$\theta^* = \arg\min_\theta \; E_{(\mathbf{X},y,s_i)\sim\mathcal{D}_{\text{train}}}\Big[\mathcal{L}(f_\theta(\mathbf{X}, s_i), y)\Big],$$

where $\mathcal{D}_{\text{train}}$ denotes the training data distribution.

Main challenge. The model must generalize effectively to unseen subjects ($s_{\text{new}} \notin S_{\text{train}}$) using only a minimal amount of calibration data from s_{new}.

Model decomposition. We decompose the function f_θ into modular components:

$$f_\theta(\mathbf{X}, s_i) = \Theta_\theta\Big(\Psi_\psi(\mathbf{X}, \mathbf{e}_{s_i})\Big),$$

where:

- $\Psi_\psi : R^{C\times T} \times R^d \to \mathcal{H} \subseteq R^d$ is a feature extractor that maps the EEG signal, conditioned on the subject embedding, to a latent representation space $\mathcal{H}$,

- $\Theta_\theta : \mathcal{H} \rightarrow [0, 1]$ is a classifier mapping latent features to the class probability,
- $\mathbf{e}_{s_i} \in R^d$ is a learnable embedding specific to subject s_i,
- ψ and θ denote the learnable parameters of the feature extractor and classifier, respectively.

This decomposition enables:

1. learning subject-conditioned representations that explicitly capture inter-subject variability,
2. separating feature extraction from final classification,
3. facilitating transfer learning and fine-tuning on new subjects.

4.1 EEGNet

EEGNet is a compatct but remarkably efficient convolutional neural network (CNN) architecture [10], developed specifically for the analysis and classification of EEG signals in the field of BCI. Born out of the need to overcome the limitations of traditional models or highly specialised networks for individual activities, EEGNet is nowadays considered a valid baseline for experiments in the EEG analysis and BCI fields. It is a structure capable of functioning effectively in different BCI paradigms, such as sensory motor rhythms (SMR), P300 evoked potentials, other types of event-related potentials ERP and error-related negativity (ERN). Its innovative design is based on the intelligent use of a small number of parameters, a result achieved by drawing inspiration from efficient architectures for mobile devices [16]. Specifically, the network is structured to effectively combine a series of 2D temporal convolutions to capture spectral characteristics (that are acting as true bandpass filters) with spatial *depthwise* convolutions (separable in depth, channels), which

extract and learn optimal spatial filters on different electrodes without excessively multiplying the number of weights. Subsequently, the network uses separable convolutions (combining *depthwise* and *pointwise* 1x1 operations) to mix the extracted feature maps. Together with the use of ReLU activation functions, Batch Normalisation and Dropout layers, this strategy gives it a huge advantage in terms of efficiency. All this makes it less weak to the problem of over-fitting than previous heavier models (such as DeepConvNet, this because big network that works with ”simple” data are very incline to over-fit), a crucial aspect since EEG datasets are often limited in size and characterised by high noise. In essence, EEGNet represented a fundamental step forward in deep learning applied to BCIs, demonstrating how it is possible to achieve high performance and better generalisation from complex signals, while maintaining a lean, multi-task architecture.

In the project it was implemented in PyTorch based on the standard EEGNet architecture but introduces a crucial modification to address our inter-subject variability: the user specific projection mechanism, that is an embedding layer `self.s_i`, where the model creates a unique representation vector for each subject, which is then used in the forward method to “shape” the features extracted from that user’s data. Finally, we also changed ELU activation for ReLU activation, which reduces the computational overhead.

4.2 P300MCNN

The P300MCNN is a convolutional neural network proposed by Liu et al. in [11] for P300 detection. It was designed with the specific goal of being extremely lightweight and efficient, reducing parameters from millions to a few thousands or even hundreds, while maintaining cutting-edge performance in detecting the event-related P300 potential. The architecture was developed with the help of XAI to ”demystify” the black box of CNNs and optimise each component. Structure and Key Components The P300MCNN essentially

consists of these main layers:

- Input: A matrix of filtered signals (channels × time).
- Precise Separable Convolution: This is the heart of the architecture. Instead of using standard convolutions, it uses a separable convolution that simultaneously captures temporal and spatial features.
- Activation Function: A fixed activation applied after the separable convolution.
- Flatten and Dense Layer: For final classification.

The efficiency of P300MCNN is based on four innovative concepts derived from XAI analysis: The first is ”Precise” Separable Convolution that unlike other models that keep filter parameters fixed, P300MCNN adapts the kernel size (k) and stride (s) based on the number of EEG channels and the type of task (intra-subject or cross-subject). The general rule is that k and s decrease slightly as the number of channels or subject variability increases to balance temporal and spatial features. The second is the use of Adaptive Activation (Adaptive Linear $\tanh(x)$) function. The model selects the activation function based on the non-linearity of the data:

- Cross-Subject: Always use Tanh, as data from multiple subjects introduces greater non-linearity.
- Within-Subject: Chooses between Linear (y=x) and Tanh. It uses Linear when the data is less noisy or less numerous (allowing for better gradients and faster training), and Tanh when the complexity is high.

In our implementation, this adaptive mechanism is replaced by a fixed activation, as detailed at the end of this section. The third concept concerns *high learning rate and few epochs*: Through visual analysis of weight updates, the authors discovered that optimisation can be dramatically accelerated. They use *cosine or restart learning rate schedulers*. This reduces training time from

hundreds of epochs (typical of EEGNet or SepConv1D) to a few dozen. Finally, the last concept is Selective Batch Normalisation. In fact, Batch Normalisation is used exclusively for cross-subject detection, where it serves to reduce the high variability between subjects and to ”smooth” the loss function landscape. It is omitted in intra-subject tasks to maintain the minimalist architecture.

In our study, P300MCNN was chosen as the feature extractor due the fact that it’s a light architecture and that it use of depthwise-separable convolutions, which are effective for learning spatial filters and temporal summaries from EEG signals in resource-constrained settings. P300MCNN was modified to include mechanisms that explicitly model subject dependence through conditioning processes; in this process, several components were adapted with respect to the original paper. Specifically: The Adaptive Linear Tanh activation is replaced by a fixed ELU activation, which proved more stable during training and the default kernel-to-stride ratio is set to $k/s = 5$ (with $k = 25$, $s = 5$), deviating from the $k/s \approx 2$ rule of the original paper to better capture the temporal dynamics of the signals used. The output layer consists of a linear projection without sigmoid, compatible with ArcFace-based metric learning objectives. It achieved its best performance after the fine-tuning phase, exceeding its own initial benchmark and benefiting greatly from the FiLM approach during adaptation. We also used P300MCNN to generate topographic maps of weights and spectrograms of activations, confirming that the architecture manages to focus on temporal and frequency patterns relevant to P300 targets. In summary, we confirmed its robustness which, although requiring fine-tuning to perform at its best on unseen data, responds well to advanced optimisation and subject conditioning techniques.

4.3 PhiNet for EEG

PhiNet [5] is a family of convolutional neural networks (CNN) 'backbones' designed specifically for Edge AI and for running on devices with extremely limited resources. The main goal is to maximise performance while minimising energy consumption, memory and computational operations. The structure consists of several consecutive blocks, the fundamental one is the inverted residual block, optimised to decouple computational and memory costs. The block structure follows this sequence:

- Expansion Convolution (Pointwise): Increases the number of channels.
- Depthwise Convolution: Operates spatially on each channel separately.
- Squeeze-and-Excitation (SE): A block that recalibrates the weights of the feature maps to improve the network's attention.
- Projection Convolution (Pointwise): Reduces the number of channels to the next block.
- Activation: The swish activation function is used.
- Skip Connections: Present between blocks with the same resolution to facilitate gradient flow, similar to MobileNetV2 [17].

The network performs down-sampling via stride convolutions, reducing the resolution by a specific factor (e.g. 32x) from input to output. It also includes a final up-sampling neck to recover spatial resolution and maximise the receptive field. Unlike other architectures (such as EfficientNet [18]) that scale all dimensions uniformly, PhiNet introduces a decoupled scaling mechanism to adapt to three specific hardware constraints:

1. Operations (MACC): Controlled by the standard parameters of resolution (w $\times$ h), width (α) and depth (number of blocks B).

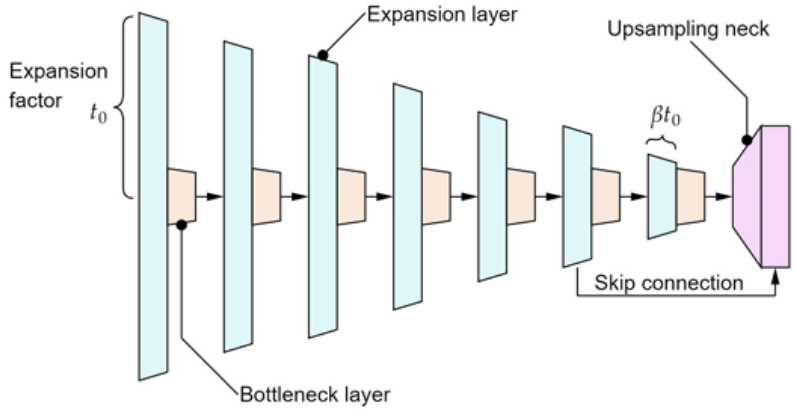


Figure 4.1: Credits: [5]; An overview of the PhiNets family network architecture. The proposed architecture scales with respect to the expansion factor *t0*, number of convolutional blocks, shape factor β and width multiplier

2. Dynamic Memory (RAM): Controlled by the base expansion factor (*t0*). This parameter determines the size of the tensors in the expansion blocks. Reducing *t0* linearly reduces RAM usage.

3. Static Memory (Flash/Parameters): Controlled by the shape factor (β). This hyper-parameter modifies the expansion factor in subsequent blocks of the network.

The formula defining the expansion factor t for a block N (out of a total of B blocks) is

$$t = t_0 \frac{(\beta - 1)N + B}{B} \tag{4.1}$$

Thanks to this formula, if β<1, the network will have fewer parameters in the deep layers (saving Flash memory); if β>1, it will have more.

A 1D adapted version of this PhiNet architecture was used as an extractor due to its extreme efficiency, scalability and low number of parameters. Implemented with an optimised hyperparameter configuration using Optuna, it is also modified for conditioning and, despite the reduced time window (0.35s) and the small number of parameters (approximately 3500), PhiNet maintained comparable performance with the other architectures. Again, analysis of the

topographic maps of the learned weights showed that PhiNet correctly focuses on the parietal and occipital areas, consistent with the detection of P300 signals.

4.4 Subject-Specific Conditioning Mechanisms

Inter-subject variability represents one of the most critical challenges in the design of deep learning-based BCI systems. Models trained on a heterogeneous pool of subjects tend to capture the "average" behaviour of the population, resulting in suboptimal performance for individuals who deviate significantly from that average. On the other hand, models trained on a single subject require large amounts of labelled data specific to that individual, a condition that is rarely met in clinical and consumer scenarios and do not benefit from the information available from other subjects.

Subject-specific conditioning offers an intermediate solution to this dilemma: instead of treating the subject's identity as an implicit confounding variable, it is explicitly integrated into the neural network's inference process. In this way, the model maintains a unified framework in which the generalised knowledge learned about the entire population is preserved, while individual specificities are captured through dedicated parameters.

All three architectures presented in the previous sections (EEGNet, P300MCNN, PhiNet) support the optional integration of a conditioning mechanism, applied to the feature representation in the penultimate layer of the network, i.e., immediately before the classification layer. In terms of the decomposition introduced in Section 4, conditioning acts on the composite function:

$$f_\theta(\mathbf{X}, s_i) = \Theta_\theta\Big(\mathcal{C}(\Psi_\psi(\mathbf{X}), \mathbf{e}_{s_i})\Big),$$

where $\mathcal{C}$ represents the conditioning operation that modulates the extracted features $\Psi_\psi(\mathbf{X})$ as a function of the subject's embedding $\mathbf{e}_{s_i}$. Conditioning is

controlled by three configurable parameters:

- `apply conditioning` (Boolean): enables or disables the conditioning mechanism;
- `conditioning method` (string): selects the method from 'film', 'h_projection' and 's_projection';

4.4.1 Fundamentals: Conditional Modulation in Neural Networks

Before describing the proposed mechanisms in detail, it is useful to frame the problem of conditioning within the broader landscape of *conditional computation* in neural networks. The idea of modulating the behaviour of a network as a function of an external signal has a long history in deep learning and has been implemented in numerous variants.

In conditional generative models, such as *Conditional Generative Adversarial Networks* (cGAN) [19], conditional information (e.g., class label) is concatenated to the input or intermediate representations to guide the generative process. Similarly, *Conditional Batch Normalisation* [20] modulates the normalisation parameters (γ and β of Batch Normalisation) as a function of a conditional vector, allowing the generation of stylistically different outputs from the same input.

In the context of *visual question answering* and conditional visual reasoning, Perez et al. [21] proposed *Feature-wise Linear Modulation* (FiLM), a general mechanism for modulating the intermediate features of a network through affine transformations parameterised by a conditional signal. FiLM has proven to be an extremely versatile mechanism, applicable to different architectures and domains.

In the specific context of BCIs, subject-specific conditioning is an active area of research. Nemes and Eigner [22] proposed an *attentive subject-fusion* framework that encodes subject information via descriptors based on power

spectral density (PSD) for *motor imagery* paradigms. Their approach uses an attention mechanism to integrate subject-specific features into the latent representations of the network. Although effective, this method relies on manually constructed descriptors, requiring a separate PSD computation step from the end-to-end training of the network. Sun et al. [23, 24] introduced an approach that integrates conditional identification information to exploit interactions between EEG signals and individual traits, combining identity information with EEG features through a conditional generator.

Compared to these approaches, the methodology proposed in this work stands out for two fundamental characteristics: (i) end-to-end learning of subject embeddings without the need for manually constructed descriptors and (ii) the architecture agnostic nature of the mechanism, which allows its application to different feature extractors without modifying their fundamental structure.

4.4.2 Subject Embedding Table

The fundamental concept behind the proposed conditioning is the use of a **subject embedding table**: a set of dense and continuous vectors learned jointly with the network parameters during training. Formally, for a dataset consisting of N subjects $S := \{s_i\}_{i=1}^{N}$, an embedding table is defined:

$$\mathbf{e} \in R^{N \times d}$$

where d is the dimensionality of the feature extractor's latent space. Each subject s_i is associated with an embedding vector $\mathbf{e}_{s_i} \in R^d$ (or R^{2d} in the case of FiLM), learned end-to-end during training. The embedding vectors are initialised randomly and updated with the backpropagation at the same time as the network weights, without explicit supervision on the structure of the embedding space.

This strategy is conceptually similar to the use of *word embeddings* in linguistic models [25]: just as a word embedding captures the semantic properties of a word in a continuous space, the subject-specific embedding captures individual neural characteristics ERP morphology, topographical distribution of activity, amplitude and latency of evoked components in a shared latent space.

4.4.3 Mathematical Foundations: Projections in Spaces with Inner Product

Before presenting the projection-based conditioning mechanisms, it is appropriate to briefly review the notions of linear algebra on which they are based.

Given a vector space R^d equipped with the standard inner product $\langle \cdot, \cdot \rangle$, the **orthogonal projection** of a vector $\mathbf{a}$ onto the direction defined by a non-zero vector $\mathbf{b}$ is defined as:

$$\text{proj}_{\mathbf{b}}\, \mathbf{a} = \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{\|\mathbf{b}\|^2}\, \mathbf{b} \tag{4.2}$$

The result is a vector in the direction of $\mathbf{b}$ whose magnitude is proportional to the component of $\mathbf{a}$ along $\mathbf{b}$. If $\mathbf{b}$ is normalised to unit norm ($\|\mathbf{b}\|_2 = 1$), the formula simplifies to:

$$\text{proj}_{\mathbf{b}}\, \mathbf{a} = \langle \mathbf{a}, \mathbf{b} \rangle\, \mathbf{b} = (\mathbf{a}^\top \mathbf{b})\, \mathbf{b}$$

The **scalar product** $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^\top \mathbf{b}$ is closely related to the **cosine similarity** between the two vectors:

$$\cos\theta = \frac{\mathbf{a}^\top \mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|} \tag{4.3}$$

where θ is the angle between $\mathbf{a}$ and $\mathbf{b}$. When $\|\mathbf{b}\|_2 = 1$, the scalar product $\mathbf{a}^\top \mathbf{b} = \|\mathbf{a}\| \cos\theta$ directly expresses the component of $\mathbf{a}$ along the direction of $\mathbf{b}$, weighted by the magnitude of $\mathbf{a}$.

This relationship is fundamental to understanding how projection-based

conditioning mechanisms work: the scalar product between the feature vector $\mathbf{h}$ and the normalised embedding $\mathbf{e}_{s_i}$ provides a measure of how much the features extracted for a given input are "aligned" with the neural profile learned for subject s_i. The projection therefore allows the representations to be modulated in a geometrically interpretable way, selecting the feature components that are consistent with the individual characteristics of the subject.

4.4.4 Approach I: Projection into Feature Space (*H-Projection*)

The first conditioning mechanism proposed operates through a projection into feature space and is called *H-projection* because the resulting vector lies in the direction of the feature vector $\mathbf{h}$. This approach represents one of the main contributions of this work and stands out for its geometric elegance, its parametric parsimony, and its ability to introduce effective subject-specific adaptation with negligible computational overhead.

Mathematical formulation. Given the vector of features extracted by the feature extractor $\mathbf{h} = \Psi_\psi(\mathbf{X}) \in R^d$ and the subject embedding $\mathbf{e}_{s_i} \in R^d$, normalised to unit norm ($\|\mathbf{e}_{s_i}\|_2 = 1$), the conditional features are calculated as the scalar projection of the subject embedding onto the feature direction:

$$\tilde{\mathbf{h}} = \text{proj}_{\mathbf{h}}\, \mathbf{e}_{s_i} = \left(\mathbf{h}^\top \mathbf{e}_{s_i}\right) \mathbf{h} \tag{4.4}$$

This operation preserves the **direction** of $\mathbf{h}$ but modulates its **magnitude** through the scalar factor $\alpha s_i := \mathbf{h}^\top \mathbf{e}_{s_i}$, which corresponds to the cosine similarity between the features and the subject's embedding, weighted by the feature norm:

$$\alpha_{s_i} = \mathbf{h}^\top \mathbf{e}_{s_i} = \|\mathbf{h}\| \cdot \cos\theta$$

where θ is the angle between $\mathbf{h}$ and $\mathbf{e}_{s_i}$ in the space R^d. The norm of the resulting vector is therefore:

$$\|\tilde{\mathbf{h}}\| = |\alpha_{s_i}| \cdot \|\mathbf{h}\| = \|\mathbf{h}\|^2 \cdot |\cos\theta| \tag{4.5}$$

Geometric interpretation. H-projection implements a subject-specific gating mechanism in the feature space with particularly significant geometric properties. To understand how it works, it is useful to analyse the behaviour of the scale factor α_{s_i} as a function of the geometric relationship between $\mathbf{h}$ and $\mathbf{e}_{s_i}$:

- **Selective amplification** ($\cos\theta \approx 1$): when the extracted features are well aligned with the direction learned from the subject's embedding, the factor α_{s_i} is positive and of high magnitude, resulting in an amplification of the feature vector. Intuitively, the model "recognises" that the features extracted from this input are consistent with the subject's typical neural profile and emphasises them accordingly;

- **Suppression of orthogonal components** ($\cos\theta \approx 0$): when the features are orthogonal to the embedding direction, the factor α_{s_i} tends to zero, effectively suppressing the signal. This acts as a bandpass filter in the feature space, removing components that are not informative for the specific subject;

- **Signal inversion** ($\cos\theta \approx -1$) : in the case where the features are anti-aligned with respect to the subject's embedding, the factor α_{s_i} is negative, inverting the direction of the feature vector. This property allows the model to encode not only affinities but also dissimilarities between the observed features and the expected profile of the subject;

- **Subject-specific receptive field**: the overall effect is equivalent to learning an individual *receptive field* in feature space. Each subject implicitly defines, through its embedding, a preferred direction $\mathbf{e}_{s_i}$ in the dimensional space. Features lying along this direction (or close to it) are preserved and amplified, while orthogonal components are suppressed. In this sense, projection conditioning acts as a *learned spatial filter* in the latent representation space, whose orientation is determined by the subject's identity.

Analysis of parametric complexity. A significant advantage of H-projection is its extreme parametric parsimony. The number of additional parameters introduced by conditioning is equal to $N \times d$, where N is the number of subjects and d is the feature dimensionality. For the architectures tested in this work, with d between 16 and 64 and $N = 22$ subjects, this translates to a few hundred additional parameters a negligible overhead compared to the 3500-9500 total parameters of the base architectures. This parsimony is particularly advantageous in the BCI context, where calibration data is scarce: a reduced number of degrees of freedom in the conditioning layer limits the risk of overfitting during fine-tuning and allows for effective adaptation even with a single batch of data from the new subject.

It is important to note that the modulation performed by H-projection is **scalar and uniform**: all dimensions of the vector $\mathbf{h}$ are scaled by the same factor α_{s_i}. This feature introduces a stronger *inductive bias* than methods that allow heterogeneous modulations by dimension (such as FiLM), which can be advantageous when the available data is limited and the risk of overfitting is high.

Implementation details. The implementation maintains an embedding table $\mathbf{e} \in R^{N \times d}$ initialised with *Xavier normal initialisation* [26] . L2 normalisation of the embedding is applied to each forward pass to ensure the constraint

$\|\mathbf{e}_{s_i}\|_2 = 1$, ensuring that the projection operation depends solely on the direction of the embedding and not on its magnitude. The module supports both flat features of size (B, D) and convolutional features of size (B, D, T) through automatic dimensional broadcasting, allowing conditioning to be applied either after flattening or directly on the temporal feature maps (the latter option is available with the parameter `early_conditioning` for P300MCNN).

Variant: S-Projection. In addition to H-projection, a variant called *S-projection* has been implemented and evaluated, in which the projection is performed along the direction of the subject's embedding rather than the features:

$$\tilde{\mathbf{h}} = \text{proj}_{\mathbf{e}_{s_i}} \mathbf{h} = \left(\mathbf{h}^\top \mathbf{e}_{s_i}\right) \mathbf{e}_{s_i} \tag{4.6}$$

In this case, the resulting vector always lies in the direction of the subject embedding $\mathbf{e}_{s_i}$, scaled by the relevance of the current input (measured by the scalar product $\mathbf{h}^\top \mathbf{e}_{s_i}$). The fundamental difference between the two variants lies in the nature of the output:

- In H-projection, the output $\tilde{\mathbf{h}}$ preserves the structure of the original features (same direction as $\mathbf{h}$), modulating only their intensity. The specific information of the input EEG signal is retained and filtered through the lens of the subject;
- In S-projection, the output $\tilde{\mathbf{h}}$ lies entirely in the subject's embedding space (direction of $\mathbf{e}_{s_i}$), effectively "replacing" the extracted features with a subject-centric representation. The specific information of the input is reduced to a scalar that modulates the magnitude of the embedding.

This difference has profound implications for the model's ability to discriminate between Target and Non-Target epochs: H-projection preserves the variability of the input features essential for distinguishing between different

types of stimuli while S-projection tends to collapse representations onto the embedding direction, reducing discriminative power. Experimental results confirmed a clear superiority of H-projection in most of the configurations tested, motivating the choice of H-projection as the main projection approach.

4.4.5 Approach II: Feature-wise Linear Modulation (FiLM)

The second conditioning mechanism is based on *Feature-wise Linear Modulation* (FiLM) [21], originally proposed in the domain of computer vision for *conditional visual reasoning*. FiLM performs affine transformations on extracted features, giving us independent heterogeneous modulations for each dimension of the feature vector.

Theoretical background: conditional affine transformations. The intuition behind FiLM is that a conditional affine transformation i.e., a *scaling* and *shifting* operation parameterised by an external signal constitutes a sufficiently expressive modulation mechanism for a wide range of conditional tasks. In formal terms, given a feature h_j (the j-th component of the feature vector) and the conditional parameters γ_j and β_j, the affine transformation:

$$\tilde{h}_j = \gamma_j \cdot h_j + \beta_j$$

can implement different operations depending on the values of the parameters: scaling ($\gamma_j \neq 1, \beta_j = 0$), shifting ($\gamma_j = 1, \beta_j \neq 0$), suppression ($\gamma_j \approx 0$), inversion ($\gamma_j < 0$), or any combination of these. Using this transformation independently to each dimension ($j = 1, \ldots, d$) allows fine control over the representation, selectively modulating the different feature components based on the conditional signal.

Mathematical formulation. Given the embedding of the subject $\mathbf{e}_{s_i} \in R^{2d}$, this is partitioned into two halves to obtain the modulation parameters $\boldsymbol{\gamma}_{s_i}, \boldsymbol{\beta}_{s_i} \in$

R^d, which are then normalised to unit norm using L2 normalisation. The conditioned features are calculated as:

$$\tilde{\mathbf{h}} = \boldsymbol{\gamma}_{s_i} \odot \mathbf{h} + \boldsymbol{\beta}_{s_i} \tag{4.7}$$

where $\odot$ denotes element-wise multiplication (Hadamard product). The parameter $\boldsymbol{\gamma}_{s_i}$ controls a multiplicative scaling for each feature dimension, determining which components to amplify, attenuate, or suppress. The parameter $\boldsymbol{\beta}_{s_i}$ introduces an additive shift, allowing activations to be translated in a subject-specific manner. This combination of scaling and shifting allows FiLM to perform independent affine transformations on each of the d dimensions of the feature vector.

Initialisation strategy. The embedding is initialised to approximate the identity transformation in the early stages of training, a crucial design choice for training stability:

- $\boldsymbol{\gamma} \sim \mathcal{N}(1, 0.02)$: initialisation close to 1 for multiplicative scaling;
- $\boldsymbol{\beta} \sim \mathcal{N}(0, 0.02)$: initialisation close to 0 for additive shift.

With this initialisation, the initial transformation is approximately $\tilde{\mathbf{h}} \approx 1 \odot \mathbf{h} + 0 = \mathbf{h}$, i.e. the conditioning layer does not significantly alter the extracted features. This allows the network to initially converge towards a subject-agnostic solution benefiting from the knowledge shared among all subjects in the training set and then progressively specialise the modulations for each subject as training proceeds. This strategy is analogous to the principle of *residual connections* [27]: starting from the identity function and incrementally learning deviations from the baseline.

Regularisation. The L2 normalisation of $\boldsymbol{\gamma}_{s_i}$ and $\boldsymbol{\beta}_{s_i}$ acts as an implicit regulariser, constraining the magnitude of the modulation parameters on a unit

hypersphere and preventing excessive scale transformations that could destabilise training. In addition, *dropout* is applied to the parameters γ and β during training: with probability p, individual components of the modulation parameters are zeroed, forcing the network not to depend excessively on specific conditioning dimensions. This approach improves the robustness of the model and reduces the risk of overfitting, which is particularly relevant in the BCI context where datasets are limited in size.

Structural comparison with H-Projection. The comparison between FiLM and H-projection highlights a fundamental trade-off between flexibility and parsimony:

- **Degrees of freedom**: FiLM introduces $2d$ parameters per subject (the vectors γ and β), compared to the d parameters of H-projection. This double number of parameters allows for richer modulations but increases the risk of overfitting;
- **Type of modulation**: H-projection applies a *scalar and uniform* modulation a single factor α_{s_i} scales all dimensions in the same way, preserving the relative proportions between feature components. FiLM, on the other hand, applies *heterogeneous modulations by dimension*: each component h_j can be scaled and translated independently, allowing the geometry of the feature space to be completely restructured for each subject;
- **Interpretability**: H-projection has an immediate geometric interpretation (projection along a preferred direction in feature space), while FiLM operates as a more general affine transformation, less directly interpretable in geometric terms;
- **Inductive bias**: the uniform modulation of H-projection introduces a stronger inductive bias, which can be advantageous when data is scarce.

FiLM, with its greater flexibility, may require more data to learn effective modulations without overfitting.

The experimental results presented in later confirm that the optimal choice between the two approaches depends on the feature extractor architecture: H-projection is superior for EEGNet and PhiNet, while FiLM performs better with P300MCNN, particularly in the fine-tuning phase. This observation suggests that the interaction between the type of conditioning and the feature extractor structure plays a crucial role in determining the overall performance of the system. It should be noted, however, that the discrepancies observed fall within the standard deviation of the MCC values.

4.4.6 Initialisation of Embeddings for New Subjects

A critical aspect of the proposed framework concerns the handling of new subjects encountered during the fine-tuning phase, for which there is no pre-learned embedding in the table. The embedding initialisation strategy for a subject not seen during training has a direct impact on the quality of initial (*zero-shot*) performance and on the convergence speed of subsequent fine-tuning.

The problem can be formulated as follows: given a new subject $s_{\text{new}} \notin S_{\text{train}}$, we want to find an initialisation $\mathbf{e}_{s_{\text{new}}}^{(0)}$ that maximises the model's performance before any calibration data for the new subject is available. Ideally, this initialisation should place the embedding in a region of the latent space that is representative of the population of subjects, providing a good starting point for subsequent optimisation.

Initialisation is performed based on the distribution of the embeddings of the already learned training subjects, according to one of the following strategies:

Arithmetic mean. The embedding of the new subject is initialised as the arithmetic mean of the embedding vectors of the training subjects:

$$\mathbf{e}_{s_{\text{new}}}^{(0)} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{e}_{s_i} \tag{4.8}$$

This strategy places the new embedding at the centre of gravity of the distribution of training embeddings, representing an "average subject". It is computationally simple but sensitive to the presence of outliers.

Geometric median. The embedding is initialised as the geometric median of the training embeddings, calculated using Weiszfeld's iterative algorithm [28]. The geometric median is defined as the point that minimises the sum of Euclidean distances from all embeddings:

$$\mathbf{e}_{s_{\text{new}}}^{(0)} = \arg\min_{\mathbf{x} \in R^d} \sum_{i=1}^{N} \|\mathbf{x} - \mathbf{e}_{s_i}\|_2 \tag{4.9}$$

Weiszfeld's algorithm solves this optimisation problem through an iterative fixed-point process:

$$\mathbf{x}^{(t+1)} = \frac{\sum_{i=1}^{N} \frac{\mathbf{e}_{s_i}}{\|\mathbf{x}^{(t)} - \mathbf{e}_{s_i}\|_2}}{\sum_{i=1}^{N} \frac{1}{\|\mathbf{x}^{(t)} - \mathbf{e}_{s_i}\|_2}} \tag{4.10}$$

where convergence is typically fast and guaranteed under general conditions.

The geometric median offers a more robust initialisation robust initialisation than the arithmetic mean, particularly when the embeddings of the training subjects are distributed unevenly in the latent space. In the presence of distinct clusters the arithmetic mean could place the new embedding in a region of space devoid of subjects (e.g., at the midpoint between two separate clusters), resulting in an unrepresentative initialisation. The geometric median, thanks to its robustness to outliers, tends to locate the starting point near the region with the highest embedding density, providing a more informative starting point for fine-tuning.

Gradient Masking for Subject-Specific Fine-Tuning. When fine-tuning on a new subject, only the embedding associated with the target subject should be updated, preserving the learned embeddings for all other subjects. In the proposed framework, this behaviour occurs naturally because the tuning batches contain samples from the target subject only. As a result, the embedding lookup operation only accesses the corresponding row of the embedding table, and PyTorch propagates gradients only to that row.

For completeness, the framework also includes an explicit gradient masking mechanism `freeze_other_subjects()`, which registers a backward hook to zero the gradients of all embedding rows except the target one:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{e}_{s_j}} = \begin{cases} \frac{\partial \mathcal{L}}{\partial \mathbf{e}_{s_j}} & \text{if } j = \text{target} \\ \mathbf{0} & \text{otherwise} \end{cases}$$

This ensures that subject-specific fine-tuning does not modify the representations learned for other subjects.

Chapter 5

Training Methodology

5.1 Loss Functions

To address class imbalance, a second strategy, adopted in this work, consists of using loss functions that intrinsically take into account the imbalance, rather than intervening on the data distribution. Weighted Binary Cross-Entropy (BCE) assigns weights inversely proportional to the frequency of each class, increasing the contribution of the target epochs to the loss function:

$$\mathcal{L}_{wBCE} = -\frac{1}{N}\sum_{i=1}^{N}\left[w_T y_i \log \hat{y}_i + w_{NT}(1-y_i)\log(1-\hat{y}_i)\right]$$

with $w_T/w_{NT} = N_{NT}/N_T = 9$ in the specific case [29].

Focal Loss [30] introduces an adaptive mechanism to reduce the weight of easily classifiable examples, focusing learning on difficult examples typically those belonging to the minority class:

$$\mathcal{L}_{FL} = -\frac{1}{N}\sum_{i=1}^{N}\left[(1-\hat{y}_i)^{\gamma} y_i \log \hat{y}_i + \hat{y}_i^{\gamma}(1-y_i)\log(1-\hat{y}_i)\right]$$

where $\gamma > 0$ is the focusing parameter. In this work, Focal Loss proved to be the preferred strategy for most of the architectures tested, in accordance with its theoretical effectiveness in addressing imbalance adaptively.

5.2 Optimizers and Learning Rate Scheduling

AdamW is the default optimiser for all models:

```
optimizer = torch.optim.AdamW(model.parameters(),
lr=learning_rate, weight_decay=0)
```

AdamW decouples weight decay from gradient updating, unlike Adam, which absorbs weight decay into gradient normalisation. With weight-decay=0, AdamW and Adam are equivalent, but the AdamW formulation is retained for potential future use.

Learning rate parameters:

- **Initial learning rate:** 5×10^{-4} (training) / 5×10^{-4} (fine-tuning)
- **LR reduction factor:** 0.1
- **LR step size (`lr_step_size`):** 20 epochs

Step learning rate scheduling reduces the learning rate by a factor of `lr_factor` every `lr_step_size` epochs when `schedule_lr=True`. This simple scheduling helps the model converge towards a flat minimum by reducing the step size as training progresses.

5.3 Data Augmentation

- **Augmentation with Gaussian noise:**

  Implemented in `augmentation/noise.py`. During training, Gaussian noise with zero mean is added to the EEG epochs:

  The standard deviation of the noise is controlled by a parameter: `base std=0.1` (in normalised units). The flag `incremental=False` uses a constant noise intensity throughout training; if set to `True`, the noise

would increase as training progresses to maintain exploration. Augmentation with Gaussian noise acts as a form of Tikhonov regularisation in continuous space, reducing sensitivity to small-amplitude artefacts.

- **Scaling augmentation** (`augmentation/scaling.py`): a standard implementation of random scaling of EEG epoch amplitudes, simulating variability in electrode impedance and signal amplitude.

5.4 Early Stopping and Regularization

- **Early stopping:** It is implemented in `dnn/early_stop.py` with a tolerance (`patience`) of 10 epochs (5 during fine-tuning) and a minimum improvement threshold `delta=0.001`. The validation loss is monitored and training is stopped if no improvement is observed for a number of consecutive epochs equal to `patience`. This approach prevents overfitting on the training set during long experiments and reduces the total training time.

- **Dropout:** It is applied after both pooling stages in EEGNet and after depthwise convolution in P300MCNN. In EEGPhiNet, `Dropout1d` is applied within each block. Dropout rates are treated as hyperparameters and optimised using Optuna.

- **Batch normalisation:** Applied after each convolutional layer in all architectures. During fine-tuning on a small target subject dataset, batch statistics may be unreliable; an adaptive batch normalisation strategy (which resets the current BN stats using unlabelled data from the target domain) is available via `adaptive_batch_norm()` in `dnn/train_test.py`.

5.5 Reproducibility

An important step taken during the experiment was to ensure that the results obtained were not the product of fortuitous configurations, favorable random initializations, or uncontrolled experimental choices. In fact, it has been demonstrated how, for example, the choice of random seed alone can introduce significant variations in the performance of deep learning models [31]. For this reason, the entire training, validation, and optimization process was designed with particular attention to the principles of robustness and reproducibility. From a reproducibility standpoint, all stochastic components were controlled and explicitly dictated (Python, NumPy, and PyTorch). Furthermore, each model was trained with multiple independent seeds in order to quantify the variability due to weight initialization, shuffling, and non-deterministic operations. To verify that the models' performance had indeed stabilized across seeds, a convergence criterion was applied. Specifically, the standard deviation of MCC values across seeds was monitored, and all methods of conditioning reached the target threshold of *standard deviation=0.05* after five seeds, confirming stable and reproducible results. Parameter optimization was made robust through the use of Optuna software. Specifically, each Optuna study was organized according to a hierarchical and nested procedure to capture inter-subject variability and that induced by randomization. For each type of conditioning we proposed, a study was used in which, within each study, Optuna iteratively generates combinations of hyperparameters evaluated on a multi-level structure.

For each Optuna trial, performance evaluation is not performed on a single split, but through a Leave-One-Subject-Out (LOSO) protocol. In this context, each subject in the dataset is taken in turn as a validation subject, while all others are used for training. Within each LOSO fold, model training and validation are further repeated on multiple predefined seeds. For each combination of subject and seed, the model is:

- initialized deterministically;
- trained using the current hyperparameters of the trial;
- evaluated using the MCC metric on the validation set.

This produces a collection of MCC values associated with all combinations (subject, seed). The final performance of the trial is therefore not based on a single value, but on the average of the MCCs obtained across all subjects and all seeds. This average value is returned to Optuna as the objective function to be maximized.

Table 5.1: Hyperparameter search space common to all architectures. Conditional parameters (marked with †) are sampled only when their parent parameter is active.

Parameter	Search Space	Notes
`loss_method`	{BCE, Focal}	
`focal_alpha`†	{0.05, 0.1, 0.15, 0.2, 0.25}	if Focal
`focal_gamma`†	{1.0, 1.5, 2.0, 2.5, 3.0}	if Focal
`resampling_method`	{None, Undersample}	Disabled if Focal
`scaler_method`	{None, StdScaler, RobustCW}	
`gaussian_noise`	{True, False}	
`learning_rate`	{0.005, 0.001, 0.0005}	
`post_event`	{0.35, 0.4, 0.5, 0.6} s	Epoch time window duration
`arcface`	{True, False}	
`arcface_s`†	{16.0, 30.0, 64.0}	if arcface
`arcface_m`†	{0.3, 0.5, 0.7}	if arcface
`seed`	{7, 42, 4321, 5321, 7678}	

Table 5.2: Architecture-specific hyperparameter search spaces used in the Optuna optimisation.

EEGNet		
Parameter	**Search Space**	**Notes**
F_1 (temporal filters)	{2, 4, 6, 8}	
D (spatial multiplier)	{1, 2}	
kernLength	{8, 16, 24, 32, 64}	
dropout_rate	{0.1, 0.2, 0.25, 0.5, 0.75}	
P300MCNN		
Parameter	**Search Space**	**Notes**
F (filters)	{6, 8, 12}	
s (stride)	{2, 3, 4, 5}	
p (padding)	{0, 1, 2, 3}	
kernLength	{8, 16, 24, 32, 64}	
dropout_rate	{0.1, 0.2, 0.3, 0.5}	
PhiNet		
Parameter	**Search Space**	**Notes**
num_layers	{2, 3, 4, 5}	
α (width mult.)	{0.1, 0.2, 0.3}	
β (shape factor)	{0.5, 0.75, 1.0, 1.25}	
t_0 (expansion base)	{2.0, 3.0, 4.0}	
kernLength	{8, 12, 16, 32, 64}	
compatibility	{True, False}	
h_swish	{True, False}	
squeeze_excite	{True, False}	
dropout_rate	{0.1, 0.3, 0.5}	

Chapter 6

Evaluation Protocol

6.1 Leave-One-Subject-Out Cross-Validation

As already mentioned the *Leave-One-Subject-Out* (LOSO) protocol has been implemented. It is the de facto the standard one for cross-subject evaluation in BCI systems. In this scheme, the entire dataset is partitioned into N folds, where N is the number of subjects: in each fold, one subject is used as the test set while the other $N - 1$ subjects make up the training set. The process is repeated for all subjects, and the final performance is the average of the results obtained on each fold.

The LOSO protocol simulates the real-world scenario in which the system

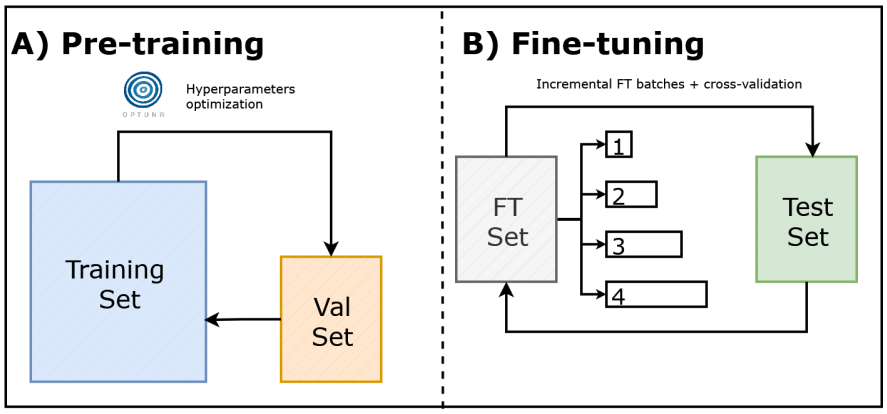


Figure 6.1: Credits: [4]. Training and adaptation procedure.

must adapt to a new user who has never interacted with the system before. Its importance lies in providing an unbiased estimate of cross-subject generalisation performance, avoiding contamination between training and test data that could artificially inflate the reported performance [14].

6.2 Fine-Tuning Strategies

Fine-tuning is the most widely used transfer learning technique in the BCI context: starting from a model pre-trained on a large number of subjects, the parameters are updated on a small dataset of the new subject to adapt the model to its specificities. This approach leverages the general knowledge acquired during pre-training as a starting point, significantly reducing the amount of subject specific data required compared to training from scratch [8].

In the presented work, an incremental fine-tuning approach was adopted: starting from a single batch of unseen subject data, the amount of calibration data is progressively increased up to a maximum of four batches. For models without conditioning mechanisms, only the final classification layer is updated during fine-tuning, freezing the feature extractor parameters. For conditioned models, only the parameters of the conditioning layer subject embedding and FiLM or projection parameters are updated, while all other network parameters are kept fixed.

This approach drastically reduces the risk of overfitting on a small amount of calibration data, ensures fast and stable convergence, and allows full exploitation of the representations learned during pre-training.

6.3 Incremental Cross-Validation

The incremental cross-validation procedure has been designed to provide a statistically significant approximation of fine-tuning performance as the amount of calibration data at hand varies. This protocol simulates the real-life case

where a BCI system should be able to adapt to a new subject by slowly adding the data volume, starting with the lowest possible amount of one batch and leading to the maximum possible of four batches.

As described in the previous section, the two sessions of the held-out subject are combined and divided temporally into a *fine-tuning set* and a *test set*, each consisting of exactly 10 batches of 60 epochs. The test set remains fixed throughout the entire evaluation, ensuring maximum comparability between the different configurations.

Combinatorial protocol. For each fine-tuning dimension $k \in \{1, 2, 3, 4\}$, the protocol evaluates **all possible combinations** of k batches selected from the N_{batches} batches available in the fine-tuning set. The total number of combinations for each value of k is given by the binomial coefficient:

$$\binom{N_{\text{batches}}}{k} = \frac{N_{\text{batches}}!}{k! \cdot (N_{\text{batches}} - k)!} \tag{6.1}$$

For each combination, the k selected batches constitute the training set for fine-tuning, while the remaining $(N_{\text{batches}} - k)$ batches serve as the validation set. For each combination, the procedure involves the following steps:

1. loading the pre-trained weights from the checkpoint with the best validation MCC;
2. initialising the embedding for the new subject (in the case of models with conditioning);
3. fine-tuning the model on the selected k batches, with validation on the remaining batches;
4. evaluation on the fixed test set.

Summary statistics mean and standard deviation of the MCC, as well as test metrics corresponding to the combination with the best validation MCC

are calculated across all combinations. This exhaustive approach allows us to quantify the variability in performance due to the specific selection of calibration batches, providing a more reliable estimate than a single random split.

Rationale. The importance of this protocol lies in the fact that, in a real BCI scenario, the quality of calibration data can vary significantly between batches for example, due to fluctuations in the subject's attention, variations in electrode impedance, or fatigue accumulation. Examining every combination, as opposed to any one incremental sequence, allows distinguishing between the effect of dataset size and the variation attributed to the particular quality of the chosen batches.

6.4 K-Fold Cross-Validation for Fine-Tuning

To complement incremental cross-validation, an additional *k-fold cross-validation* protocol was applied to the entire fine-tuning set for $k \in \{2, 3, 4, 5\}$.

Protocol. The fine-tuning set is divided into k folds of approximately equal size. For each fold $i \in \{1, \ldots, k\}$:

1. fold i is used as the fine-tuning validation set;
2. the remaining $(k-1)$ folds constitute the training set for fine-tuning;
3. the model is fine-tuned on the training set and evaluated on both the validation fold and the fixed test set.

Lastly, performance is measured as the average of the values achieved in the k folds thus giving an approximation of the generalization ability of the fine-tuned model less affected by the peculiarities of a specific data partition.

Relationship with the incremental protocol. While incremental cross validation fixes the size of the fine-tuning set and varies the batch selection, k-fold

cross-validation varies the *proportion* between training and validation data within the fine-tuning set. As k increases, the fraction of data used for training increases ($\frac{k-1}{k}$), allowing us to measure the **saturation point of the learning curve**: beyond which amount of calibration data, performance stops improving significantly. This information is directly relevant to the design of efficient calibration protocols, as it identifies the minimum amount of data needed to achieve stable performance, minimising the time required by the user.

6.5 Metrics

The choice of evaluation metric is critical in imbalanced contexts such as ERP-based BCIs. Traditional metrics such as accuracy are misleading in the presence of strong imbalances: a classifier that always predicts Non-Target achieves 90% accuracy on a dataset with a 1:9 ratio, yet is completely useless for the BCI application.

Even the F1-score, although more robust than accuracy, does not fully capture the model's ability to discriminate between classes in all possible scenarios. The *Matthews Correlation Coefficient* (MCC) [32] is instead a robust metric for evaluating binary classifiers in the presence of imbalance, defined as:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{6.2}$$

where TP, TN, FP, and FN are true positives, true negatives, false positives, and false negatives, respectively. The MCC varies between -1 and $+1$: a value of $+1$ indicates perfect classification, 0 indicates a random classifier, and -1 indicates systematically incorrect classification.

An important comparative study [33] has shown that the MCC is more reliable than balanced accuracy, informedness and markedness in the evaluation of binary classifiers from confusion matrices, especially in the presence of

significant imbalance between classes. For these reasons, MCC was selected as the target metric in all optimisation phases of this work. Other computed metrics:

- Accuracy: Overall fraction of correct predictions. Reported for completeness but interpreted cautiously given the 9:1 class imbalance.
- Binary Cross-Entropy Loss: The training loss on the test set, indicating model calibration.
- F1 Score (Binary): Harmonic mean of precision and recall for the Target class.
- ROC-AUC (Area Under the Receiver Operating Characteristic Curve): A threshold-independent measure of discrimination ability. A classifier with AUC=0.5 performs at chance; AUC=1.0 indicates perfect separation of class score distributions.
- Balanced Accuracy: Average of sensitivity (true positive rate) and specificity (true negative rate). Appropriate for imbalanced classes.

Chapter 7

Experimental Results and Analysis

This chapter presents the results obtained from experiments conducted on the BrainForm dataset, analysing the performance of the three architectures (EEG-Net, P300MCNN, PhiNet).

All results reported are expressed as the mean ± standard deviation of the Matthews Correlation Coefficient (MCC) calculated on the folds of the Leave-One-Subject-Out (LOSO) protocol, unless otherwise indicated.

7.1 Zero-Shot Performance and Incremental Fine-Tuning

The table reports the complete results of the nine configurations tested (three architectures × three conditioning modes: none, H-projection, FiLM) in the five experimental conditions: zero-shot (pre-trained model without any test subject data) and incremental fine-tuning with 1, 2, 3, and 4 batches of calibration data.

7.1.1 Zero-Shot Analysis

Zero-shot performance characterises the direct cross-subject generalisation ability of each model, i.e., the performance achievable on a completely new

Model	Zero-Shot MCC	FT I1 MCC	FT I2 MCC	FT I3 MCC	FT I4 MCC
EEGNet	0.6460 ± 0.2151	0.6518 ± 0.2102	0.6532 ± 0.2112	0.6535 ± 0.2120	0.6539 ± 0.2128
EEGNet (H)	$\mathbf{0.6535 \pm 0.2227}$	0.6568 ± 0.2076	0.6574 ± 0.2093	0.6581 ± 0.2091	0.6577 ± 0.2099
EEGNet (F)	0.6289 ± 0.2252	$\mathbf{0.6574 \pm 0.2102}$	$\mathbf{0.6609 \pm 0.2111}$	$\mathbf{0.6620 \pm 0.2120}$	$\mathbf{0.6621 \pm 0.2133}$
P300MCNN	$\mathbf{0.6088 \pm 0.2075}$	$\mathbf{0.6404 \pm 0.1844}$	$\mathbf{0.6621 \pm 0.1737}$	$\mathbf{0.6772 \pm 0.1698}$	$\mathbf{0.6890 \pm 0.1661}$
P300MCNN (H)	0.5799 ± 0.2135	0.5790 ± 0.2040	0.5838 ± 0.2104	0.5871 ± 0.2159	0.5888 ± 0.2204
P300MCNN (F)	0.4687 ± 0.2777	0.6064 ± 0.2085	0.6308 ± 0.2048	0.6324 ± 0.2072	0.6349 ± 0.2089
PhiNet	0.6227 ± 0.1906	0.6361 ± 0.2019	0.6364 ± 0.2006	0.6377 ± 0.1996	0.6394 ± 0.1980
PhiNet (H)	$\mathbf{0.6269 \pm 0.2229}$	$\mathbf{0.6389 \pm 0.2154}$	$\mathbf{0.6413 \pm 0.2121}$	$\mathbf{0.6439 \pm 0.2101}$	$\mathbf{0.6469 \pm 0.2085}$
PhiNet (F)	0.6231 ± 0.2388	0.6359 ± 0.2275	0.6400 ± 0.2257	0.6431 ± 0.2221	0.6460 ± 0.2206

Table 7.1: Zero-shot and fine-tuning performance. Bold characters denote the best score of each column. Models denoted with H use the projection-based conditioning, while F use FiLM-based conditioning.

subject without any calibration data.

In the zero-shot condition, models equipped with a conditioning layer generally achieve better or comparable performance than unconditioned models, with the notable exception of the P300MCNN architecture. In particular, the proposed H-projection method achieves the highest zero-shot score for EEGNet (0.6535) and PhiNet (0.6269), surpassing both FiLM and the configuration without conditioning. For EEGNet, H-projection clearly outperforms FiLM (0.6535 vs. 0.6289), confirming that the uniform scalar modulation of H-projection is sufficient and in this case preferable to capture inter-subject variability in the features extracted by EEGNet.

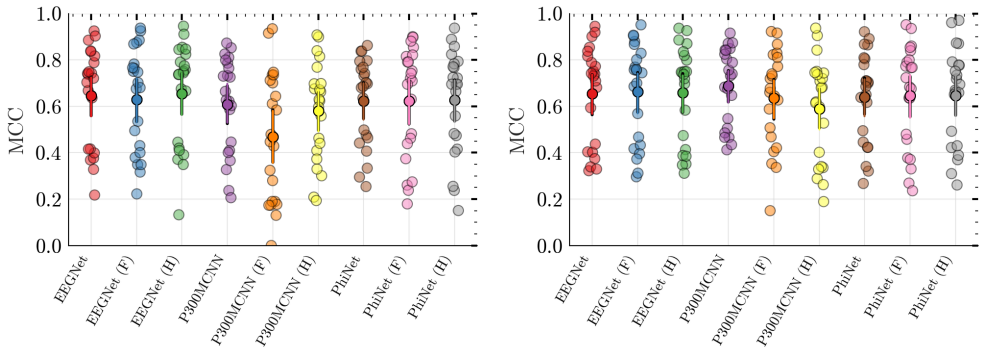
The case of P300MCNN represents a significant exception: the model without conditioning achieves an MCC of 0.6088, while H-projection (0.5799) and FiLM (0.4687) show lower performance. This result suggests that conditioning during the pre-training phase may interfere with the learning process of P300MCNN, probably due to the interaction between conditioning mechanisms and the selective Batch Normalisation characteristic of this architecture. FiLM's low zero-shot score for P300MCNN (0.4687) is particularly indicative: the greater flexibility of FiLM conditioning, with twice as many parameters as H-projection, seems to lead to overfitting of the modulation parameters during pre-training.

For PhiNet, the three configurations show very similar performance in zero-shot (0.6227, 0.6269, 0.6231), indicating that PhiNet is inherently robust to inter-subject variability, probably thanks to the Squeeze-and-Excitation blocks

that provide an adaptive attention mechanism on the channels. The very small time window (0.35s) just enough to capture the ERP components does not prevent PhiNet from achieving performance comparable to other architectures with larger windows.

7.1.2 Analysis of the Incremental Fine-Tuning Phase

During the fine-tuning phase, the models tend to reach a performance plateau already after the first fine-tuning step (a single batch of 60 epochs, of which about 6 are Target), with subsequent batches producing marginal improvements. This behaviour is clearly visible in the plots in Figure 7.1 and represents a relevant result from a practical point of view: a single batch of calibration data corresponding to approximately 24 seconds of P300 recording with 10 stimuli is sufficient to achieve most of the improvement achievable through fine-tuning.



(a) Pre-training phase (b) Fine-Tuning Stage (4 batches)

Figure 7.1: Comparison of MCC Scores. We analyze how various conditioning strategies affect the performance of the test neural architectures across different training stages.

The main exception to this behaviour is P300MCNN without conditioning, which shows a progressive and substantial improvement during fine-tuning, going from 0.6088 in zero-shot to 0.6890 after 4 batches an increase of $+0.0802$ in MCC, the largest among all tested configurations. This result indicates that P300MCNN significantly benefits from updating the final classifier with subject-specific data, probably because its compact architecture and *Precise*

Separable Convolution extract general features that require a more substantial adaptation of the decision boundary for each subject.

Regarding the comparison of conditioning methods in the fine-tuning phase, architecture-specific patterns emerge:

- **EEGNet**: FiLM achieves the best absolute score (0.6621 after 4 batches), recovering from its initial disadvantage compared to H-projection. This suggests that FiLM's greater flexibility, with its heterogeneous modulations by dimension, becomes advantageous when subject-specific data is available to calibrate the parameters γ and β.

- **P300MCNN**: the model without conditioning clearly dominates (0.6890), while FiLM (0.6349) significantly outperforms H-projection (0.5888) . FiLM's result is particularly noteworthy: starting from the worst zero-shot score among all configurations (0.4687), FiLM recovers $+0.1662$ MCC points in 4 batches, demonstrating remarkable subject-specific adaptability when paired with calibration data.

- **PhiNet**: H-projection achieves the best performance (0.6469 after 4 batches), followed by FiLM (0.6460) and the unconditioned model (0.6394). The margins are small, confirming the intrinsic robustness of PhiNet.

7.1.3 Inter-Subject Variability

A crucial aspect highlighted by the results is the **substantial variability among held-out subjects** in each LOSO fold. As illustrated in the plots (Figure 7.1), the standard deviations of the MCC range from 0.17 to 0.28, indicating that some subjects are classified with MCC values above 0.8 while others obtain values close to 0.0 or negative.

This variability is partially reduced after the first fine-tuning step, indicating that conditioning improves performance for most subjects, but then remains relatively stable despite the addition of further batches. This behaviour

suggests that for subjects with atypical P300 morphology (delayed latency, diffuse scalp distribution, reduced amplitude), fine-tuning with a few batches is not sufficient to bridge the performance gap, highlighting a possible limitation in the quantity and quality of available data.

A relevant observation concerns the comparison between the standard deviations of the different configurations. Unconditioned models tend to have lower standard deviations (e.g., PhiNet: ± 0.1906 in zero-shot), while conditioned models show greater variance (PhiNet (F): ± 0.2388). This suggests that conditioning, while improving average performance for "typical" subjects, may introduce greater variability for atypical subjects whose embedding has not been well learned during pre-training.

7.2 Ablation on Conditioning Methods

To gain a deeper understanding of the effect of conditioning mechanisms, this section analyses the behaviour of FiLM, H-projection, and S-projection in comparison to unconditioned models under the LOSO protocol.

7.2.1 FiLM

FiLM conditioning consistently improves zero-shot performance for EEGNet and PhiNet, albeit to varying degrees. The channel-wise scaling and shifting parameters (γ, β) effectively capture subject-specific differences in the amplitude and phase of ERP components. The improvement is more pronounced for subjects who are outliers in the training distribution: for these subjects, affine modulation allows the features to be "rescaled" to compensate for deviations from the mean distribution.

In the fine-tuning phase, FiLM shows its main potential with P300MCNN, where it achieves an MCC of 0.6349 compared to 0.5888 for H-projection. FiLM also achieves the best absolute score among EEGNet architectures with 0.6621. Updating only the FiLM parameters (frozen backbone embedding)

produces a greater improvement per unit of calibration data than updating only the final classifier, suggesting that FiLM parameters capture subject-specific variability more efficiently than linear classifier weights.

7.2.2 H-Projection

H-projection shows moderate but consistent improvements for EEGNet and PhiNet in zero-shot conditions. The main advantage of H-projection lies in its parsimony: with only d parameters per subject (versus $2d$ for FiLM), it achieves competitive or superior performance in zero-shot, where the risk of embedding overfitting is more relevant due to the absence of subject-specific data during pre-training.

For P300MCNN, H-projection can occasionally degrade performance compared to the unconditioned model. Projection along the feature direction can suppress useful cross-subject components when the subject embedding is not well aligned, a phenomenon that occurs more frequently with P300MCNN due to the interaction between projection and the conditional Batch Normalisation of this architecture.

7.2.3 S-Projection

S-projection shows significantly greater variance in performance than the other methods. As discussed in Section 4.4.4, S-projection forces the output to lie in the direction of the subject embedding, a strong constraint that benefits some subjects those whose embedding has been well learned and faithfully captures the individual neural profile but penalises those for whom the embedding space has not converged adequately. This greater sensitivity to the quality of the learned embedding makes S-projection less reliable as a general conditioning strategy.

7.3 Fine-Tuning Efficiency Analysis

The incremental cross-validation protocol (Section 6.3) provides learning curves that relate the amount of calibration data to test performance. The key results are as follows.

A single batch is sufficient for substantial improvement. A batch of calibration data (60 trials, including approximately 6 targets) produces a significant improvement over zero-shot performance for all architectures and conditioning configurations. In the most striking case, FiLM for P300MCNN goes from 0.4687 to 0.6064 with a single batch an increase of $+0.1377$ in MCC. This amount of data corresponds to approximately 24 seconds of P300 recording at a stimulation rate of 10 targets, an extremely short calibration time compared to traditional protocols.

Decreasing returns beyond 3 batches. The MCC curve shows clear saturation: the average improvement between the third and fourth batches is in the order of 0.001-0.003 MCC points for most configurations. This indicates that the model has sufficiently adapted to the target subject with approximately $3 \times 60 = 180$ samples. The only partial exception is P300MCNN without conditioning, which shows a still appreciable improvement in the fourth batch ($+0.0118$ from FT I3 to FT I4).

The variance decreases rapidly with the number of batches. With a single batch, the standard deviation of the MCC calculated over all possible batch selections is high, reflecting sensitivity to which specific trials are included in the calibration set. Already with 3 batches, the variance is substantially reduced, indicating that the specific composition of the calibration set becomes less relevant as the amount of data increases.

Comparison: fine-tuning of conditioning vs. fine-tuning of the classifier. Fine-tuning only the conditioning parameters (exclusive update of the FiLM embedding or projection embedding, with frozen backbone) achieves performance comparable to fine-tuning the classifier with 1-2 batches. This result confirms that conditioning parameters efficiently capture subject-specific variability, offering a more parsimonious and potentially more robust adaptation mechanism.

7.4 Comparison of Loss Functions

The choice of loss function is a critical aspect in P300 classification, given the strong imbalance between classes (Target:Non-Target ratio of 1:9). Three strategies were evaluated:

BCE with default weights. Standard Binary Cross-Entropy, without any correction for imbalance, achieves high overall accuracy but very low recall for the Target class. The model tends to classify most epochs as Non-Target, a strategy that maximises overall accuracy ($\sim 90\%$) but is completely useless for the BCI application.

BCE with class weights. Weighted BCE (with `pos_weight` ≈ 9, inversely proportional to the frequency of classes) substantially improves the recall of the Target class, at the cost of a reduction in specificity for the Non-Target class. This is the optimal configuration identified by Optuna for P300MCNN, where selective Batch Normalisation and the adaptive activation function (ALT) benefit from explicit weight distribution.

Focal Loss. Focal Loss ($\alpha = 0.1, \gamma = 2.0$) emerges as the preferred method for handling imbalance in most configurations. The adaptive downweighting mechanism of easy examples (γ) consistently improves MCC compared to standard BCE on the imbalanced training set. The effect is more pronounced

for architectures with lower capacity (P300MCNN), which would otherwise tend to always predict Non-Target. Focal Loss was selected as the optimal configuration for both EEGNet and PhiNet, confirming its effectiveness in scenarios with strong imbalance where the focus on difficult examples is particularly advantageous.

7.5 Comparison of Normalisation Strategies

The EEG data normalisation strategy affects training stability and robustness to artefacts. Three options were evaluated:

The **RobustScaler**, which uses the median and interquartile range instead of the mean and standard deviation, was selected as the optimal configuration for all three architectures by Optuna research. Its robustness to outliers is particularly advantageous in the EEG context, where occasional artefacts (muscular, ocular, movement) can introduce samples with amplitudes well above the norm that would excessively influence the statistics of a StandardScaler.

The **StandardScaler** (z-score normalisation per channel) provides stable training in most configurations and is the most consistent among the architectures, while being more sensitive to amplitude artefacts.

The **MinMaxScaler** can lead to instability when the signal amplitude in the test set exceeds the range observed in the training set, a frequent phenomenon with EEG data from new subjects with signal characteristics different from the training population. For this reason, it is not recommended as primary normalisation.

7.6 Interpretability Analysis

To verify that the representations learned by the models are neurophysiologically plausible and to understand the differences between architectures, an

interpretability analysis was conducted on three levels: topographical analysis of weights, time-frequency analysis of activations, and visualisation of subject embeddings.

7.6.1 Topographical Analysis of Channel Relevance

The analysis of channel relevance was performed by examining the weights of the first convolutional layer of each architecture, the layer closest to the raw signal and therefore the one that provides the most direct indication of the relevance of channels for model decisions. The contribution of each channel c was quantified as the energy of its weights:

$$I_c = \sum_{d,k} w_{c,d,k}^2 \tag{7.1}$$

where d represents the number of filters in the convolution and k the time index. These values were normalised and visualised on topographic maps of the scalp, producing an intuitive representation of the cortical regions considered most informative by each model.

As shown in Figure 7.2, and consistent with the results of the neurophysiological literature on P300, the analysis highlights a concentration of activity on the parietal and occipital channels (Pz, PO7, Oz, PO8) for all three architectures. These channels correspond to the cortical regions where the P300 typically reaches its maximum amplitude, confirming that the models have learned neurophysiologically consistent representations.

A particularly relevant aspect is **cross-model consistency**: despite substantial architectural differences between EEGNet, P300MCNN and PhiNet, the topographic maps show similar patterns of relevance, providing an objective measure of cross-architecture consistency and interpretability. This result reinforces confidence in the validity of the learned representations, as models with different structures and inductive biases converge towards the same selection of informative cortical regions.

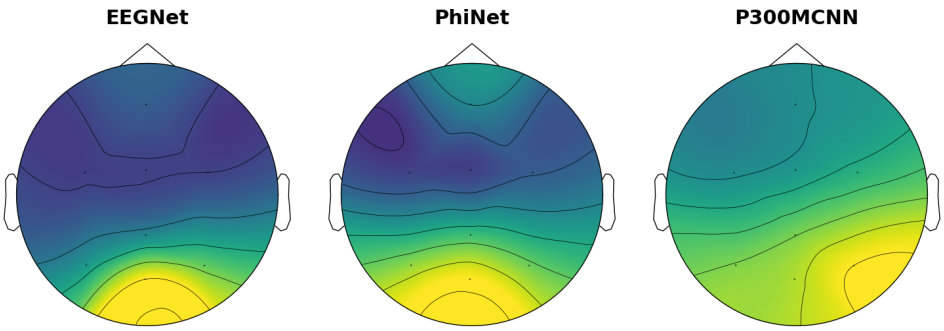
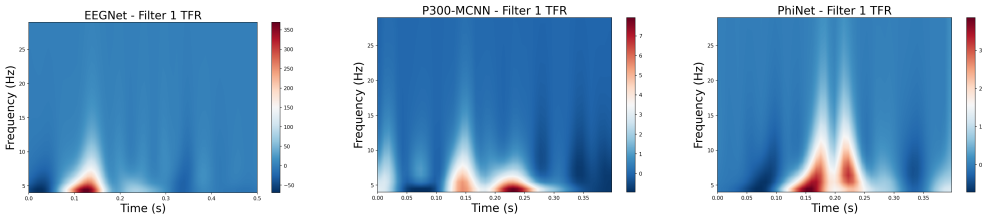


Figure 7.2: Weight energy distribution for the three architectures. The importance of the channels increases from cool colours to warm colours. The three models converge in attributing greater relevance to the parietal and occipital regions, consistent with the expected scalp distribution of the P300.

7.6.2 Time-Frequency Analysis of Activations

Complementing the weight-based analysis, the activations of the same convolutional layers were examined to characterise the time-frequency patterns emphasised by each architecture in the early stages of processing. Specifically, the filter responses to EEG trials in both conditions (Target and Non-Target) were calculated . These activations were transformed into the time-frequency domain using Morlet wavelets, producing spectrograms for each filter that capture the oscillatory dynamics associated with the input signals.



(a) EEGNet (b) P300-MCNN (c) PhiNet

Figure 7.3: Difference in filter responses between Target and Non-Target stimuli, displayed as a function of frequency and time. Warm colours indicate stronger responses for Targets (T), cool colours indicate stronger responses for Non-Targets (NT).

The differential spectrograms (Figure 7.3) show the differences in filter responses between Target and Non-Target conditions. Warm colours indicate stronger responses for Targets, while cool colours indicate stronger responses

for Non-Targets. The analysis confirms that the learned filters emphasise components in the frequency band relevant to ERPs (typically below 15 Hz), with a clear temporal differentiation between Target and Non-Target responses in the 200-500 ms post-stimulus window, consistent with the expected latency of the P300 component.

7.6.3 Visualisation of the Embedding Space

After pre-training, the weights of the embedding table were extracted from the conditioning layers and projected into a low-dimensional space using UMAP [34] for visualisation. As shown in Figure 7.4, subjects form distinct clusters in the embedding space, with groupings identified via k-means clustering.

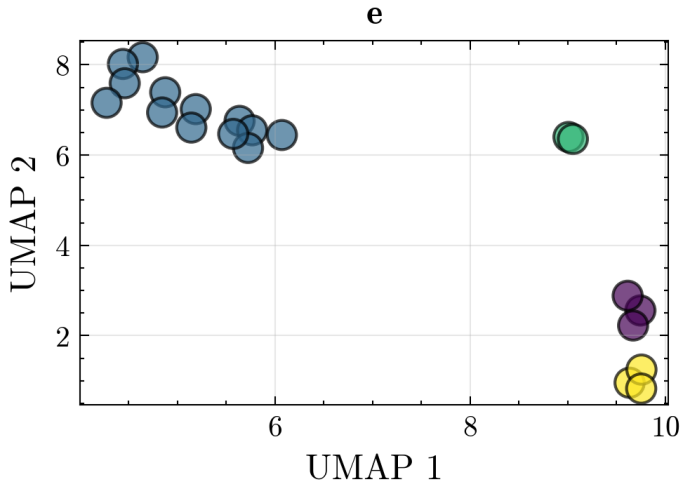


Figure 7.4: UMAP projection of the embedding table $\mathbf{e}$ extracted from PhiNet (F) after pre-training on all subjects except EXP_P12. Each colour represents a cluster extracted using k-means clustering.

The formation of clusters reflects individual differences in EEG signals and ERP components, confirming that the network implicitly learns a similarity structure between subjects through the training process. These observable groupings suggest that similarities between subjects could be exploited to improve both the initialisation of the embedding table and the training process, for example by initialising the embedding of a new subject as the average of

the embeddings of the closest cluster rather than the overall population.

This three-dimensional analysis weights, activations, and embeddings provides a comprehensive picture of the plausibility and interpretability of the learned representations, a particularly important requirement in the context of BCIs where user and clinician trust in the system's decisions is critical for the adoption of the technology.

7.6.4 Preliminary Comparison with the State of the Art

A preliminary comparison with the standard Riemannian pipelines of MOABB [14] in terms of ROC AUC is available on the paper's companion website [35]. In the zero-shot setting, the Riemannian methods

(`Xdw+LDA`, `dwCov+TS+SVM`, `RPCov+TS+LDA`) achieve averages comparable to the proposed deep models, but with higher inter-subject variability, with some subjects dropping as low as 0.65. The conditioned CNN models exhibit more compact distributions, although they do not systematically outperform the baselines on average. It should be noted that this comparison is to be considered preliminary, as it does not include the results following fine-tuning, which lead to further improvements, and the comparison was conducted in an initial experimental phase.

Chapter 8

Discussion

In this chapter, the results are contextualised toBCI research, with the aim of extracting general design principles, positioning the work relatively to the state of the art, and acknowledging its limitations.

8.1 Design Principles for Subject-Adaptive BCIs

The set of experiments conducted with an ablation studies on conditioning methods, comparison of architectures and analysis of fine-tuning curves suggests some design principles: The first principle concerns *parsimony as a form of regularisation*. The results systematically show that methods with fewer degrees of freedom (H-projection, with d parameters per subject) tend to be more stable in zero-shot conditions, where subject-specific data are not available to calibrate complex parameters. FiLM, with $2d$ parameters, outperforms H-projection only after fine-tuning, when subject data becomes available to guide the optimisation of heterogeneous modulations. This parsimony/flexibility trade-off is consistent with the classic bias-variance principle: in data-scarce regimes, a model with stronger inductive bias (H-projection, uniform modulation) generalises better than a more flexible model (FiLM, modulation by dimension). The second principle concerns the interaction between conditioning and architecture. There is no universally superior conditioning

method: H-projection is optimal for EEGNet and PhiNet, FiLM recovers better during fine-tuning for P300MCNN, and P300MCNN without conditioning achieves the best absolute score. The hypothesis that emerges from the analysis of the results is that conditioning is more useful when the architecture has fewer internal adaptation mechanisms. PhiNet, thanks to its Squeeze-and-Excitation blocks, implements an implicit form of channel recalibration that is functionally analogous to conditioning; this explains why the addition of explicit conditioning produces marginal improvements. P300MCNN, with its selective Batch Normalisation activated specifically for cross-subject tasks, may suffer from functional redundancy with conditioning, explaining its inferior performance. EEGNet, the most "neutral" architecture from this point of view, is the one that most clearly benefits from conditioning. The third principle, is that minimal calibration is sufficient. A single batch of 60 trials approximately 24 seconds of ERP recording with 10 stimuli produces most of the improvement achievable through fine-tuning. Diminishing returns beyond 3 batches suggest that prolonged calibration protocols are not justified by the marginal gain in performance. This result indicates that the proposed conditioning stategy can reduce the amount of subject-specific data needed for adaptation.

8.2 The Geometry of Embedding Space

A particularly interesting emerging property of learned embeddings is the spontaneous formation of **subject-dependent clusters** (Figure 7.4). UMAP (*Uniform Manifold Approximation and Projection*) analysis [34] of the weights of the embedding table, extracted after the pre-training phase, revealed distinct groupings corresponding to subjects with similar neural characteristics (another behavioural comparison could be with the metric learning paradigm). The application of *k-means clustering* on the UMAP projections confirmed the presence of non-trivial structures in the embedding space, indicating that the

network implicitly learns a taxonomy of subjects based on their neural characteristics. These clusters suggest that similarities between subjects could be exploited both to improve the initialisation of embeddings for new subjects and for knowledge transfer between individuals with similar neural profiles.

This observation has three implications.

The first is *diagnostic*: a subject's position in embedding space could serve as a predictive indicator of classification quality. And that the distance from the nearest centroid could therefore serve as a system confidence metric, signalling when predictions are less reliable.

The second implication is operational: the cluster structure suggests a more sophisticated embedding initialisation strategy for new subjects than the arithmetic mean or geometric median currently implemented. Initialising the embedding of a new subject at the nearest cluster centroid identified, for example, on the basis of a few initial trials could improve both the quality of zero-shot predictions and the convergence speed of fine-tuning.

The third implication is theoretical: the spontaneous formation of clusters indicates that the embedding space captures genuine structures of inter-subject variability and not optimisation artefacts. This is consistent with the literature on the morphology of evoked potentials, which identifies distinguishable ERP phenotypes related to factors such as age, experience with BCI systems, and individual neuroanatomical characteristics [36].

8.3 Class Imbalance: Lessons Learned

Managing class imbalance has proven to be a determining factor for performance, to the point that the choice of loss function has a comparable or greater impact than the choice of architecture or conditioning method.

Focal Loss emerged as the preferred strategy for EEGNet and PhiNet, thanks to its adaptive mechanism that reduces the weight of easily classifiable examples and focuses learning on hard examples typically Target epochs,

which are both rare and more variable across subjects. Weighted BCE proved optimal for P300MCNN, showing the importance of the weights.

A significant result is that rebalancing strategies based on the loss function proved to be more effective than explicit oversampling of the minority class, which can boost performance at the cost of increased computational time. This is consistent with the literature on deep learning with imbalanced classes: oversampling a few Target examples risks causing over-fitting on specific samples, while Focal Loss operates at the gradient level, modulating learning without altering the data distribution.

8.4 Limitations

Despite the promising results, the work has revealed some limitations.

The most significant one is the use of a single dataset. All experiments were conducted on the BrainForm dataset, which uses 8 EEG channels, an ERP paradigm with 10 concurrent stimuli, and a population of subjects skilled in computer use. Generalisability to different configurations more channels, different paradigms, clinical populations remains to be verified.

Second, the absence of online validation limits the scope of the conclusions. In a real-world BCI setting, factors such as computational latency, real-time feedback, fatigue accumulation, and signal drift over time play a crucial role. The PhiNet architecture, with its $\sim$3500 parameters, is designed for deployment on edge devices and is the most natural candidate for online validation, but this step has not been taken in the present work.

Finally, the proposed framework does not explicitly address the problem of *BCI-illiterate subjects*. For subjects with atypical P300 morphology, standard conditioning and fine-tuning may not produce sufficient improvements. Dedicated strategies meta-learning, specific augmentation, or adversarial domain adaptation approaches may be necessary for this specific population.

Chapter 9

Conclusions and Future Work

9.1 Summary

This study addressed the problem of inter-subject variability in the classification of EEG evoked potentials, proposing an end-to-end framework that integrates lightweight models with subject-specific conditioning mechanisms.

The work faces an open challenge in BCI research: reducing or eliminating the calibration phase required to adapt the system to new users without sacrificing classification performance. The proposed approach addresses this challenge by explicitly modelling the subject's identity as a model covariate through end-to-end learned embeddings and conditioning of latent representations. The main contributions can be summarised in four points.

Architecture-agnostic conditioning. Two conditioning mechanisms H-projection and FiLM were formalised, implemented, and evaluated on three lightweight architectures (EEGNet, P300MCNN, PhiNet). H-projection, based on a scalar projection in feature space, offers the best parsimony-performance trade-off in zero-shot conditions. FiLM, with its dimension-heterogeneous affine modulation, achieves the best absolute performance after fine-tuning for EEGNet (MCC = 0.6621). The optimal choice of conditioning method depends on the architecture, a result that underscores the importance of joint optimisation.

Efficient calibration. A single calibration batch (60 trials, $\sim$12 seconds of recording) is sufficient to achieve most of the improvement achievable through fine-tuning. Diminishing returns beyond 3 batches (average improvement $<$0.003 MCC) confirm that short calibration protocols are not only sufficient but also preferable, avoiding the fatigue and loss of motivation associated with prolonged sessions.

Robust optimisation. Hyperparameter optimisation using Optuna, with nested LOSO and multiple seeds, identified stable and reproducible configurations. Focal Loss and RobustScaler emerged as optimal choices for most architectures, giving us practical ideas for EEG pipeline design.

Interpretability. Analysis on three levels weight topography, activation spectrograms, and UMAP visualisation of embeddings confirms that the learned representations are neurophysiologically plausible. The three architectures converge in focusing on the parietal and occipital regions, consistent with the topographical distribution of P300. Spontaneous cluster formation in the embedding space reveals an implicit taxonomy of subjects that opens up both diagnostic and operational perspectives.

9.2 Future Developments

The results and limitations discussed in this thesis outline several directions for future development, which can be organised along three main axes: deployment on embedded devices, experimental extension of the framework, and methodological improvement.

9.2.1 Deployment on embedded hardware and online validation

The most immediate and concrete direction concerns the deployment of the proposed models on low-resource hardware platforms, with the aim of creating a real-time BCI system. This development fits naturally into the activities of the E3DA (Energy-Efficient Embedded Digital Architectures) laboratory at Fondazione Bruno Kessler, which focuses on the development of energy-efficient artificial intelligence solutions for edge devices. The architectures evaluated in this thesis were selected precisely because of their suitability for deployment on devices with limited computational and energy resources: PhiNet, with approximately 3,500 parameters, is the most natural candidate for this transition. The development of a complete framework that integrates signal acquisition, preprocessing, inference, and real-time feedback is the next phase of the project. Online validation is essential to verify the performance of the system under real operating conditions, where factors such as computational latency, real-time feedback, fatigue accumulation and signal drift during the session play a decisive role. The absence of this validation is one of the main limitations of this work and is therefore a priority for future developments.

9.2.2 Extension to different datasets and paradigms

A significant limitation of this work is the use of a single dataset. All experiments were conducted on the BrainForm dataset, which uses 8 EEG channels, an ERP paradigm with 10 concurrent stimuli, and a population of subjects with computer experience. The generalisability to different configurations a larger number of channels, different paradigms (e.g., motor imagery or SSVEP), clinical populations remains to be verified. Extending the conditioning framework to public datasets and different BCI tasks would allow the robustness and transferability of the proposed approach to be evaluated. In

particular, application to motor imagery paradigms, where inter-subject variability manifests itself differently than in ERPs, would provide a meaningful test of the general applicability of the method.

9.2.3 Methodological improvements

The experimental results suggest several opportunities for methodological improvement. UMAP analysis of the embedding space revealed a cluster structure reflecting neural similarities between subjects. This structure suggests a more sophisticated embedding initialisation strategy for new subjects than the arithmetic mean or geometric median currently implemented. Initialising the embedding of a new subject at the centroid of the nearest cluster identified, for example, on the basis of a few initial trials could improve both the quality of zero-shot predictions and the convergence speed of fine-tuning. An hybrid approach with the state-of-the-art could combine the advantages of both paradigms. For example, the use of Riemannian projections as input features for conditional networks, or the systematic integration of Riemannian conditioning, represent promising directions. Finally, the problem of so-called “BCI-illiterate” subjects, for whom standard conditioning and fine-tuning do not produce sufficient improvements, requires dedicated strategies. Approaches based on meta-learning, augmentations specific to this subpopulation, or adversarial domain adaptation techniques could offer more effective solutions for these subjects.

Bibliography

[1] TMSi, "The 10-20 system for EEG," https://www.tmsi.artinis.com/blog/the-10-20-system-for-eeg, accessed: 2025-03-09.

[2] E. Houssein, A. Hammad, and A. Ali, "Human emotion recognition from eeg-based brain–computer interface using machine learning: a comprehensive review," *Neural Computing and Applications*, vol. 34, 05 2022.

[3] K. Won, M. Kwon, M. Ahn, and S. C. Jun, "Eeg dataset for rsvp and p300 speller brain-computer interfaces," *Scientific Data*, vol. 9, no. 388, 2022. [Online]. Available: https://doi.org/10.1038/s41597-022-01456-1

[4] M. Romani, D. Zanoni, E. Farella, and L. Turchet, "Brainform: a serious game for bci training and data collection," *arXiv preprint arXiv:2510.10169*, 2025, version 2.

[5] F. Paissan, A. Ancilotto, and E. Farella, "PhiNets: a scalable backbone for low-power AI at the edge," Oct. 2021, arXiv:2110.00337 [cs]. [Online]. Available: http://arxiv.org/abs/2110.00337

[6] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A Review of Classification Algorithms for EEG-based Brain-Computer Interfaces: A 10-year Update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 55, Apr. 2018. [Online]. Available: https://inria.hal.science/hal-01846433

[7] J. J. Vidal, "Toward direct brain-computer communication," *Annual Review of Biophysics and Bioengineering*, vol. 2, pp. 157–180, 1973.

[8] X. Huang, Y. Xu, J. Hua, W. Yi, H. Yin, R. Hu, and S. Wang, "A review on signal processing approaches to reduce calibration time in eeg-based brain–computer interface," *Frontiers in Neuroscience*, vol. Volume 15 - 2021, 2021. [Online]. Available: https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2021.733546

[9] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review," *Brain-Computer Interfaces*, vol. 4, no. 3, pp. 155–174, 2017. [Online]. Available: https://doi.org/10.1080/2326263X.2017.1297192

[10] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces," *J Neural Eng*, vol. 15, no. 5, p. 056013, Oct. 2018.

[11] M. Liu, W. Shi, L. Zhao, and F. R. Beyette, "Best performance with fewest resources: Unveiling the most resource-efficient Convolutional Neural Network for P300 detection with the aid of Explainable AI," *Machine Learning with Applications*, vol. 16, p. 100542, Jun. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827024000185

[12] J. Ma, B. Yang, W. Qiu, Y. Li, S. Gao, and X. Xia, "A large EEG dataset for studying cross-session variability in motor imagery brain-computer interface," *Sci Data*, vol. 9, no. 1, p. 531, Sep. 2022, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41597-022-01647-1

[13] G. Huang, Z. Zhao, S. Zhang, Z. Hu, J. Fan, M. Fu, J. Chen, Y. Xiao, J. Wang, and G. Dan, "Discrepancy between inter- and intra-subject

variability in EEG-based motor imagery brain-computer interface: Evidence from multiple perspectives," *Front. Neurosci.*, vol. 17, Feb. 2023, publisher: Frontiers. [Online]. Available: https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2023.1122661/full

[14] S. Chevallier, I. Carrara, B. Aristimunha, P. Guetschel, S. Sedlar, B. Lopes, S. Velut, S. Khazem, and T. Moreau, "The largest eeg-based bci reproducibility study for open science: the moabb benchmark," 2024. [Online]. Available: https://arxiv.org/abs/2404.15319

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[18] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.

[19] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014.

[20] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," 2017.

[21] E. Perez, F. Strub, H. d. Vries, V. Dumoulin, and A. Courville, "FiLM: Visual Reasoning with a General Conditioning Layer," Dec. 2017, arXiv:1709.07871 [cs]. [Online]. Available: http://arxiv.org/abs/1709.07871

[22] □. Gyula Nemes and G. Eigner, "Subject Conditioning for Motor Imagery Using Attention Mechanism," *IEEE Access*, vol. 12, pp. 170 243–170 249, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10716385

[23] P. Sun, J. D. Winne, P. Devos, and D. Botteldooren, "EEG decoding with conditional identification information," Mar. 2024, arXiv:2403.15489 [eess]. [Online]. Available: http://arxiv.org/abs/2403.15489

[24] P. Sun, J. D. Winne, M. Zhang, P. Devos, and D. Botteldooren, "Electroencephalography Decoding with Conditional Identification Generator," *Int. J. Neur. Syst.*, vol. 35, no. 07, p. 2550024, Jul. 2025, publisher: World Scientific Publishing Co. [Online]. Available: https://www.worldscientific.com/doi/10.1142/S0129065725500248

[25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[26] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[28] E. Weiszfeld, “Sur le point pour lequel la somme des distances de n points donnés est minimum,” *Tohoku Mathematical Journal, First Series*, vol. 43, pp. 355–386, 1937.

[29] F. Topsøe, “Bounds for entropy and divergence for distributions over a two-element set.” *JIPAM. Journal of Inequalities in Pure & Applied Mathematics [electronic only]*, vol. 2, no. 2, pp. Paper No. 25, 13 p.–Paper No. 25, 13 p., 2001. [Online]. Available: http://eudml.org/doc/122035

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” Feb. 2018, arXiv:1708.02002 [cs]. [Online]. Available: http://arxiv.org/abs/1708.02002

[31] D. Picard, “Torch.manual_seed(3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision,” *CoRR*, vol. abs/2109.08203, 2021. [Online]. Available: https://arxiv.org/abs/2109.08203

[32] P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen, and H. Nielsen, “Assessing the accuracy of prediction algorithms for classification: an overview,” *Bioinformatics*, vol. 16, no. 5, pp. 412–424, 05 2000. [Online]. Available: https://doi.org/10.1093/bioinformatics/16.5.412

[33] D. Chicco, N. Tötsch, and G. Jurman, “The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation,” *BioData Min*, vol. 14, p. 13, Feb. 2021. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7863449/

[34] L. McInnes, J. Healy, and J. Melville, “Umap: Uniform manifold approximation and projection for dimension reduction,” 2020. [Online]. Available: https://arxiv.org/abs/1802.03426

[35] M. Romani, A. Fossà, E. Farella, and R. Calegari, "Explicit subject conditioning for lightweight cross-subject erp classification," *arXiv preprint*, 2025, forthcoming.

[36] B. Z. Allison and C. Neuper, "Could anyone use a bci?" *Brain-Computer Interfaces*, pp. 35–54, 2010.