Zurich University
of Applied Sciences

School of
Engineering

CAI Centre for
Artificial Intelligence

Master Thesis | MSE Data Science

Cluster-Based Transfer Learning for Motor Imagery BCI: Evaluating Offline Generalization and In-Session Calibration

Author	Ece Asirim
Advisor	Ricardo Chavarriaga
Date	01.09.2025

Zusammenfassung

Nicht-invasive Brain–Computer Interfaces (BCIs) versprechen eine intuitive Steuerung, sind jedoch weiterhin durch erhebliche interindividuelle Variabilität und den Aufwand der benutzerspezifischen Kalibrierung eingeschränkt. Diese Herausforderungen sind besonders ausgeprägt beim Motor Imagery (MI) Decoding mit niedrig aufgelöstem EEG, bei dem Modelle häufig nicht auf unbekannte Nutzer generalisieren. Dieses Projekt verfolgt zwei Ziele: den Aufbau einer robusten Transfer-Learning-Pipeline für das MI-Decoding über mehrere Probanden hinweg sowie die Evaluation, wie sich das vortrainierte Framework auf prospektiv aufgezeichnete Nutzer mit einem EEG-Headset anpassen lässt – mit dem Ziel, die Kalibrierung zu minimieren und gleichzeitig die Genauigkeit beizubehalten.

Ein einheitliches Framework wird eingeführt, das invariantes Repräsentationslernen über Probanden hinweg sowie schnelle Personalisierung durch drei Komponenten ermöglicht: (i) konvolutionale Baselines auf Roh-EEG (Deep4Net), (ii) ein Multi-Task-Modell mit gemeinsamem Feature-Extractor und cluster-spezifischen Köpfen sowie (iii) Transfer-Learning-Protokolle zur Anpassung des vortrainierten Backbones an unbekannte Nutzer unter drei Szenarien: Transfer eines gepoolten Modells, Leave-One-Subject-Out (LOSO) Zero-Shot Transfer und LOSO Few-Shot Kalibrierung. Die Experimente erfolgen auf einem harmonisierten MI-Datensatz mit 85 Probanden sowie auf prospektiv aufgenommenen Sitzungen mit strikter Trennung zwischen Trainings- und Testdaten.

Die Ergebnisse zeigen, dass Common Spatial Pattern (CSP)-Features eine stabile Struktur im Subjekt-Raum offenbaren, wodurch eine effektive Clusterung der Population ermöglicht wird. Cluster-konditioniertes Multi-Task-Learning verbessert die Dateneffizienz in Few-Shot-Szenarien: Die Kalibrierung eines leichtgewichtigen Kopfes mit vier gelabelten Versuchen pro Klasse führt zu konsistenten Verbesserungen gegenüber gepoolten Transfermodellen. Zero-Shot Transfer allein bringt hingegen nur begrenzten Nutzen, und generische Augmentierungen oder eine erhöhte Anzahl an Kalibrierungsversuchen verbessern die Leistung nicht konsistent – was die Bedeutung gezielter Anpassung unterstreicht.

Insgesamt zeigt das Projekt, dass das Lernen gemeinsamer Repräsentationen auf gepoolten Daten, die Stratifizierung von Subjektclustern und eine leichtgewichtige benutzerspezifische Kalibrierung eine genaue, reproduzierbare und daten-effiziente MI-Dekodierung ermöglichen.

Abstract

Non-invasive brain–computer interfaces promise intuitive control but remain limited by substantial inter-subject variability and the cost of per-user calibration. These challenges are particularly pronounced for motor imagery decoding with low-density EEG, where models often fail to generalize to unseen users. This project pursues two objectives: establishing a robust transfer learning pipeline for cross-subject MI decoding and evaluating how the pretrained framework can be adapted to prospective users recorded with an EEG headset, aiming to minimize calibration while maintaining accuracy.

A unified framework is introduced to address subject-invariant representation learning and rapid personalization through three components: (i) convolutional baselines on raw EEG (Deep4Net), (ii) a multi-task model with shared feature extraction and cluster-conditioned heads, and (iii) transfer learning protocols for adapting the pretrained backbone to unseen users under three regimes: a pooled model transfer, leave-one-subject-out (LOSO) zero-shot transfer, and LOSO few-shot calibration. Experiments are conducted on a harmonized 85-subject MI dataset and prospectively recorded sessions, with strict separation between training and testing sets. Findings show that common spatial pattern (CSP) features expose a stable subject-space structure, enabling effective population clustering. Cluster-conditioned MTL improves data efficiency in few-shot settings, as calibrating a lightweight head with four labeled trials per class yields consistent gains over pooled TL models. Zero-shot transfer alone provides limited benefit, and generic augmentations or an increased number of calibration trials do not consistently improve performance, highlighting the importance of targeted adaptation.

Collectively, this project demonstrates that learning a shared representation on pooled data, stratifying subject-level population clusters, and applying lightweight per-user calibration enables accurate, reproducible, and data-efficient MI decoding.

Contents

Chapter 1

Introduction

1.1 Problem Statement

Brain-computer interfaces (BCIs) represent a direct communication pathway between the human brain and external devices, offering transformative potential in fields ranging from neurorehabilitation to assistive technologies. Despite decades of research, practical deployment of BCIs remains severely limited [25, 41]. A central obstacle is the high variability in neural signals across individuals and sessions, which undermines the reliability and usability of current systems. The necessity for extensive user-specific calibration, often requiring long recording sessions, constitutes a significant barrier to adoption outside controlled laboratory conditions [36, 48].

Electroencephalography (EEG) is the most widely used modality for BCIs due to its non-invasive, portable, and relatively inexpensive nature. However, EEG signals are inherently noisy, exhibit significant inter-subject variability, and are highly sensitive to contextual factors such as electrode placement, physiology, and mental state cite Cheng2020. This variability causes models trained on one set of users to generalize poorly to new users. As a result, most BCI systems still rely on subject-specific training, thereby limiting practical, user-ready deployment [30].

Motor imagery (MI) is a particularly well-studied paradigm in EEG-based BCIs, offering intuitive control by leveraging users' mental rehearsal of limb movements. The challenge of inter-subject variability persists in this paradigm [2, 63]. Conventional machine learning models often fail to generalize across subjects or sessions, necessitating lengthy per-user calibration to achieve acceptable performance [36, 79].

Although deep learning techniques have demonstrated stronger within-subject performance, they frequently require retraining for each individual and may still underperform on newly encountered users. Some end-to-end deep transfer models fail to exceed practical accuracy thresholds in subject, independent evaluations [4]. Consequently, motor imagery–based BCIs remain challenging to deploy in real-world settings where fast and reliable personalization is required.

Addressing this challenge requires more sophisticated strategies that leverage information across multiple users while supporting rapid adaptation to new individuals. Transfer learning and multi-task learning approaches provide a framework to bridge this gap, enabling shared representations that capture common structure across subjects while retaining flexibility for individual adaptation [46, 79]. Nonetheless, successfully deploying such approaches under realistic conditions, for instance, with a limited number of EEG channels and minimal calibration data, remains challenging and is a focus of recent work [24, 46, 79]. This project is positioned within this challenge, aiming to contribute methodological and practical insights towards creating MI-based BCIs that are both generalizable and deployable.

To this end, the work undertaken in this project develops and evaluates a comprehensive framework for improving generalization and reducing calibration time in motor imagery BCIs, with two complementary focuses that together form a single pathway from population learning to user-specific adaptation (detailed in Subsection 1.3). These two focuses are connected through a modular pipeline that: (1) extracts stable subject-level features, (2) stratifies the population via k-means clustering to preserve inter-subject structure, (3) trains a Deep4Net backbone with cluster-specific heads using multi-task learning, and (4) performs few-shot adaptation using only the head layer for unseen users. This design isolates generalizable representations while enabling lightweight, calibration-efficient personalization. Together, they establish a complete pathway from population-level learning to subject-specific deployment.

1.2 Related Work

EEG-based BCIs have demonstrated significant potential in enabling direct communication between the brain and external systems, particularly in motor imagery (MI) paradigms. However, robust decoding of EEG signals remains a challenging problem due to the high inter-subject and intra-subject variability of neural patterns, low signal-to-noise ratios, and nonstationary dynamics of brain activity. Consequently, achieving reliable performance across diverse populations and recording sessions requires developing models that generalize effectively while minimizing the need for extensive subject-specific calibration.

Subject-invariant representation learning has emerged as a promising avenue to address these challenges. Studies such as Kostas and Rudzicz (2020) have explored domain adaptation methods to learn representations that are less sensitive to inter-subject differences, thereby improving cross-subject generalization [38]. Similarly, Lawhern et al. introduced EEGNet, a compact convolutional neural network designed to learn features invariant to specific recording conditions, demonstrating improvements in classification performance across multiple EEG-based paradigms [40]. Complementary to deep models, Riemannian geometry-based approaches that operate in covariance space have also shown cross-subject robustness in MI decoding [74]. These findings underscore the need for representation learning techniques that can extract stable, transferable features across heterogeneous populations. However, many evaluations do not explicitly model population structure or apply rigorous leave-one-subject-out (LOSO) validation protocols, which might limit how confidently one can assess subject-independent generalization.

As a result, even the most promising generalization strategies often still rely on some degree of subject-specific calibration to achieve reliable decoding performance. Calibration aims to fine-tune models to an individual's unique neural patterns, thereby compensating for inter-subject variability. He and Wu (2020) demonstrated that small amounts of calibration data can substantially enhance decoding accuracy compared to pooled models trained across multiple subjects [28]. While calibration can improve performance, frequent or extended calibration procedures reduce practical usability. Moreover, many studies emphasize decoding accuracy but provide limited details on calibration effort—such as the number of labeled trials, duration, or consistency over time, making it difficult to assess their viability for real-world deployment.

Transfer learning (TL) has become a central strategy in addressing this trade-off between calibration effort and generalization. Approaches such as those proposed by Jayaram et al. (2016) leverage pre-trained models on large offline datasets and adapt them to new, unseen subjects using limited calibration data. More recent work by Fahimi et al. (2019) has shown that TL can significantly outperform models trained on pooled data alone and even traditional subject-specific training in low-data regimes, highlighting its utility for MI-BCIs. Nevertheless, these benefits are strongly dependent on the quality of the pre-trained representations and the similarity between source and target domains. Recent efforts suggest that introducing structure, such as subgrouping subjects based on shared features, can enhance transferability by guiding models to adapt using more relevant examples. This pairing may help reduce calibration time and improve robustness in low-data conditions.

Multi-task learning (MTL) complements transfer learning by jointly optimizing across multiple related tasks, such as decoding EEG signals from different subjects or paradigms. Zhang et al. (2022) demonstrated that MTL can exploit inter-subject similarities by sharing a common model backbone while allowing subject-specific adaptations via dedicated task heads, resulting in improved performance relative to single-task models [85]. Such architectures are particularly suitable for EEG decoding, where population diversity can be exploited to build robust shared representations while preserving flexibility for individual-specific adjustments. Evaluating MTL frameworks under deployment-relevant constraints, such as LOSO generalization, explicit cluster-based heads, limited channels, and short calibration windows, could offer deeper insight into how shared backbones and specialized heads contribute to accuracy, efficiency, and adaptability in practical settings.

This project integrates these advances by proposing a framework that combines population-based clustering, multi-task learning, and transfer learning to optimize MI decoding in EEG. A pre-trained deep learning model is developed, leveraging population-level feature clustering to define shared representational structures. Multi-task learning is employed to train a shared backbone with cluster-specific heads, capturing both invariant and subgroup-specific patterns. The trained model is subsequently transferred to unseen subjects, where its performance is evaluated in both zero-shot and few-shot calibration settings. By systematically comparing cluster-based and pooled approaches, this work addresses the longstanding challenge of balancing cross-subject generalization with minimal calibration requirements, contributing to the development of BCIs that are both accurate and practical for real-world deployment. The main contribution of this work is to leverage population structure explicitly, enforce subject-independent validation, and demonstrate few-shot personalization under deployment-like conditions.

1.3 Goals

This project aims to develop and evaluate a subject-aware transfer learning framework for motor imagery (MI)– based EEG decoding, with the overarching goal of improving generalization across users while minimizing the need for subject-specific calibration. The work is structured around two interconnected objectives: first, to establish and validate a robust cluster-based transfer learning pipeline using a publicly available MI–EEG dataset; second, to test this pipeline under realistic deployment conditions on newly acquired user data, assessing its practical viability for rapid personalization.

Objective 1 focuses on constructing a reliable population-level transfer learning (TL) framework using a public dataset of 85 subjects recorded under a common MI protocol. The central question is whether explicitly modeling subject heterogeneity through unsupervised clustering can improve cross-subject generalization and few-shot personalization under realistic data constraints. To address this, several subcomponents are investigated. First, candidate feature families, including CSP, ERD/ERS, FBCSP, and Riemannian representations, are compared in terms of their ability to produce stable, interpretable subject clusters. CSP with $k = 3$ clusters is selected as the default basis due to its superior geometric separation and assignment stability. This representation anchors all subsequent comparisons.

Second, the benefit of clustering is evaluated in two settings. In the strict zero-shot case, clustering offers a modest yet consistent improvement over pooled models. In the more practically relevant few-shot setting, where each new subject contributes only a small number of labeled trials, clustered models substantially outperform pooled baselines under identical calibration trials. These improvements are especially pronounced when the support set is restricted to the subject's assigned cluster, revealing that the source of adaptation data matters more than its quantity.

Third, the robustness of the clustered pipeline is tested through targeted ablations. Increasing the number of clusters from 3 to 4 offers no measurable benefit and may fragment the data; doubling the number of calibration trials degrades performance, likely due to nuisance variability. Augmentation strategies, such as time warping, noise, frequency shifts, and mixup, are also evaluated; however, they consistently fail to improve generalization and, in some cases, even harm it. Finally, a feature-level analysis reveals that the benefit of clustering cannot be reliably predicted from scalar EEG features or baseline accuracy, suggesting that its effectiveness is structural rather than feature-driven. Taken together, these findings establish clustered few-shot adaptation, with CSP features and $k=3$ cluster heads, as a stable and data-efficient default for MI decoding on heterogeneous populations.

Objective 2 transitions from offline benchmarking to *in-session evaluation* by applying the pretrained cluster-based pipeline to newly acquired EEG recordings collected with a consumer-grade Unicorn headset. The goal is to evaluate whether the benefits observed offline persist under real-world conditions and minimal calibration. Each new subject is routed to a cluster-specific head using unsupervised embedding assignments, and their performance is tracked as a function of the number of available calibration trials. Results from two held-out users reveal a consistent pattern: cluster initialization yields higher accuracy than pooled initialization when only one labeled trial per class is available, while pooled initialization becomes preferable at higher calibration trials. These outcomes qualitatively mirror the trends observed offline, suggesting that the proposed TL framework transfers reliably to new users and can support subject-specific adaptation with only a handful of labeled examples.

By jointly validating population-level learning and deployment-realistic personalization, the project provides a cohesive demonstration that clustering-based transfer learning can substantially reduce calibration requirements in MI–BCI pipelines, offering both methodological insight and practical steps toward more deployable EEG decoding systems.

Chapter 2

Theoretical Background

2.1 Brain–Computer Interfaces

A brain-computer interface (BCI) allows a user to communicate or control external devices directly through brain signals, bypassing traditional muscle pathways [78]. BCIs have diverse applications, from assistive technologies in healthcare and rehabilitation to novel human-computer interaction in entertainment and education [43]. In particular, BCI systems offer hope for people with severe motor impairments by restoring lost functions, for example, by enabling people with spinal cord injuries to operate prosthetic limbs or helping stroke survivors regain mobility [43].

BCIs can be implemented with implanted invasive electrodes or non-invasive sensors. Among non-invasive methods, electroencephalography (EEG) is the most widely used due to its safety, portability, and relatively low cost, although at the expense of lower spatial resolution compared to invasive techniques [20]. EEG–based BCIs rely on the measurement of voltage fluctuations generated by synchronous neural activity and captured via non-invasive scalp electrodes, typically arranged according to the international 10–20 system. These voltage signals are dominated by rhythmic oscillations spanning standard frequency bands, delta (0.5–4 Hz), theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and gamma ($>$ 30 Hz), each reflecting different cognitive and motor processes.

Different EEG paradigms have been explored in BCI research, including event-related potentials (ERPs) such as the P300 response, steady-state visual evoked potentials (SSVEPs) elicited by flickering stimuli, and motor imagery (MI) tasks that exploit sensorimotor rhythm modulations during imagined movements. These paradigms enable users to issue discrete or continuous commands. While SSVEP offers high information transfer rates, MI provides a more intuitive and natural form of control, but often requires extensive training to achieve competitive accuracy [31, 45].

2.2 Electroencephalography (EEG)

Electroencephalography (EEG) is a method of recording brain electrical activity from the scalp using a collection of electrodes [67]. The measured signal represents voltage fluctuations generated by neuronal activity, predominantly reflecting the summed postsynaptic potentials of large populations of cortical pyramidal neurons firing in synchrony [7, 18, 29, 35].

Despite its advantages, EEG faces several critical limitations that directly affect its effectiveness in BCI systems. Its spatial resolution is relatively low; scalp recordings capture a blurred aggregate of neural signals due to volume conduction, leading to poor localization precision [26] compared to imaging modalities like fMRI [27, 54]. Even with high-density electrode arrays, improvements remain modest and come at the cost of greater setup complexity, computational burden, and increased challenges in maintaining a stable signal-to-noise ratio (SNR) across channels [17].

Furthermore, EEG signals are highly susceptible to noise and artifacts. Because they are typically in the microvolt range, even minor disturbances from eye blinks, muscle contractions, or environmental electrical interference can obscure relevant neural activity [61]. As a result, extensive preprocessing, filtering, and artifact removal are required to extract meaningful information, which remains challenging in real-world, unconstrained environments.

Another significant challenge is inter-individual variability. Studies indicate that approximately 15-30% of users fail to produce reliable EEG control signals even after training, a phenomenon known as *BCI illiteracy* [19]. These differences arise from diverse factors, including neurophysiological variations, cognitive strategies, and attention levels, which can make BCIs effective for some users but less so for others. Finally, usability issues, such as lengthy calibration procedures and performance degradation due to motion artifacts, limit the scalability of EEG-based BCIs outside controlled laboratory settings.

In summary, EEG provides a powerful, non-invasive window into brain activity; however, its limitations in spatial resolution, noise susceptibility, inter-subject variability, and real-world usability pose significant challenges for achieving robust, generalizable BCI performance.

2.3 EEG-based Motor Imagery

Among EEG-based BCI paradigms, motor imagery (MI) stands out for its intuitive, endogenous control mechanism. Unlike paradigms that rely on stimulus-locked brain responses, such as event-related potentials (ERPs) or steady-state visual evoked potentials (SSVEPs), MI enables users to initiate control voluntarily by imagining movements without physical execution [10, 12, 68]. This imagined movement evokes distinct modulations in the sensorimotor cortex, which can be non-invasively measured using EEG.

During MI, the mental rehearsal of limb movements, such as clenching a fist or rotating a wrist, leads to a characteristic decrease in oscillatory power within the μ (8–12 Hz) and β (18–30 Hz) bands over sensorimotor areas, an effect known as event-related desynchronization (ERD) [60, 77]. After the imagery task concludes, these rhythms typically rebound in power, resulting in event-related synchronization (ERS) [58]. The ERD/ERS effects are spatially lateralized; for example, imagining right-hand movement produces stronger ERD over the left (contralateral) hemisphere, typically observed at channels such as C3 and C4 [59]. This lateralization enables reliable classification of imagined left versus right-hand movements using EEG signals.

MI-based BCIs are appealing for their natural and asynchronous control, allowing users to generate commands at will without external stimuli [77]. However, the practical realization of MI-BCIs remains challenging. The EEG signatures of MI vary significantly across individuals, influenced by anatomical, psychological, and attentional factors [2, 34]. Many users require extensive training to learn how to modulate their brain rhythms consistently, while the system must calibrate a model to their individual neural patterns [47, 70]. Even with training, a notable portion of users often struggle to produce sufficiently discriminable EEG patterns for MI classification [1, 33, 64, 69, 81, 84].

To transform raw EEG into actionable commands, MI-BCI systems use multi-stage pipelines that include signal preprocessing, spatial filtering, feature extraction, and classification [47]. The quality of each stage has a critical impact on overall performance. Much of the literature has focused on optimizing these components, including improving spatial filters such as common spatial patterns (CSP), designing robust features, and tuning classifier hyperparameters, to enhance decoding accuracy in both subject-specific and cross-subject scenarios [8, 49, 78].

Despite its challenges, MI remains one of the most extensively studied paradigms for non-invasive BCIs. Its dependence on internally generated signals rather than external stimuli offers a pathway to more autonomous and personalized neural interfaces. Ongoing research continues to address the key limitations of MI-BCIs, particularly the high inter-subject variability and calibration burden, to make them more robust, accessible, and deployable.

2.4 Population Clustering & Multi-Task Learning

2.4.1 Population Clustering

EEG recordings from different individuals often exhibit significant variations in spatial, spectral, and temporal patterns, which impede the generalization of single-model approaches across subjects. Traditional BCI methods typically develop subject-specific classifiers, requiring substantial calibration data for each new user, which is impractical in many real-world applications [11, 48]. To overcome this, strategies including population clustering and multi-task learning (MTL) have been increasingly employed, aiming to leverage shared structure across subjects and enhance model generalization [32, 48, 80].

Population clustering refers to grouping subjects into clusters based on similarities in their EEG data, thus identifying homogeneous subsets of users who share common neural activation patterns or features. By grouping individuals with comparable brain signal characteristics, clustering reduces the complexity of handling inter-subject variability, facilitating the development of more robust and generalizable models tailored to each identified group [37, 39]. Krauledat et al. showed that clustering CSP filters from previous sessions to create prototypical spatial filters, then combining them with a small amount of data from a new session, can produce classifiers that generalize across sessions as well as or better than traditional full-session calibration, significantly reducing the need for lengthy recalibration in experienced BCI users [39]. Using these clusters, a shared model is trained while reducing inter-subject differences, achieving performance comparable to individualized (per-subject) models and significantly better than an unclustered inter-subject model. Ultimately, population clustering establishes meaningful subpopulations within diverse EEG datasets. These subpopulations significantly simplify subsequent model training, whether through transfer learning or multi-task frameworks, by effectively isolating and capitalizing on the common neural structures and minimizing the negative impact of variability.

2.4.2 Multi-Task Learning

Multi-task learning (MTL) is a machine learning paradigm in which a single model is trained on multiple related tasks simultaneously, leveraging shared information to improve both efficiency and generalization. The central hypothesis is that related tasks share latent structure or representations, and learning them jointly yields better generalization than learning them in isolation [14, 15, 62].

In the context of EEG-based BCIs, the natural formulation of MTL is typically subject-specific; each subject's EEG decoding task can be viewed as a separate but related task, leveraging both shared (across-subject) and task-specific (within-subject) components of the neural architecture [40]. Alamgir et al. showed that treating each subject's decoding as a related task within a hierarchical Bayesian MTL framework allows effective information sharing across users. By combining priors from earlier recordings with minimal new data, they achieved high accuracy without full subject-specific calibration and demonstrated robust generalization across different experimental setups [3].

Deep neural networks have further extended MTL applications in EEG-based BCIs by allowing richer representation learning across subjects. For example, Autthasan et al. (2021) proposed a multi-task convolutional neural network (CNN) that simultaneously optimized EEG feature extraction for motor imagery classification and additional auxiliary tasks such as signal reconstruction. This approach explicitly enforced the learning of discriminative, robust, and generalizable EEG representations, resulting in significantly better performance in unseen subject data [6].

Additionally, recent deep learning frameworks such as the MIN2Net have shown substantial improvements by jointly learning spatial, temporal, and spectral features across multiple subjects and classification tasks [6]. These architectures strike a delicate balance between shared convolutional layers, which capture generalizable features, and task-specific layers, which adapt the model to individual subjects or subject groups [15]. Such deep learning frameworks have shown substantial improvements by jointly learning spatial, temporal, and spectral features across multiple subjects and classification tasks.

2.4.3 Clustered Multi-Task Learning Framework

While population clustering and multi-task learning each help mitigate inter-subject variability on their own, combining these strategies can yield even more robust EEG–BCI models. Clustering stratifies the subject pool into more homogeneous groups, reducing gross between-subject differences, and MTL then exploits shared information within or across those groups to learn generalizable features.

One straightforward approach is a two-stage pipeline: first, cluster the subjects based on EEG feature similarity, and then train a multi-task model using those cluster assignments. In practice, this can be implemented as follows:

Cluster Formation: Group users into clusters according to their neural signal characteristics, such as spatial–spectral profiles or activation dynamics. The grouping can be performed using k-means or spectral clustering on EEG-derived feature vectors, or with deep metric learning approaches like PRISM, which encodes EEG similarity [85]. The goal is to identify subpopulations of subjects who share consistent patterns in their brain activity.

Cluster-Specific Multi-Task Learning: For each cluster, train an MTL model in which each subject is treated as a separate task. The model architecture can include shared cluster-level parameters (e.g., a common feature extractor) and subject-specific parameters for individual decoders. This structure ensures that learning is constrained within each subgroup, preventing negative transfer from unrelated subjects. In effect, clustered MTL acts as a hierarchical learning framework, capturing generalizable representations at the group level while preserving subject-level idiosyncrasies.

New Subject Adaptation: For a previously unseen user, a small amount of calibration data can be used to assign them to the closest cluster, using the same features employed during clustering. The corresponding pre-trained cluster MTL model can then be adapted to the new user by fine-tuning only the task-specific decoder, while keeping the shared cluster backbone fixed. This cluster-conditioned strategy dramatically reduces the need for full subject-specific retraining and has been shown to enable efficient transfer with minimal calibration [44].

This combined strategy leverages multi-level knowledge transfer, sharing information globally across all subjects, within each cluster, and individually. Notably, machine learning research has shown that automatically discovering task groupings and incorporating them into MTL can significantly enhance performance. For instance, Liu et al. (2017) introduced a hierarchical clustered MTL approach that alternates between learning a multi-task model and clustering the tasks, thereby jointly finding optimal groupings and shared models [44]. Their results demonstrated that the discovered latent relatedness (task clusters) "aids in inducing the group-wise multi-task learning and boosts the performance," outperforming approaches that treat all tasks as either completely independent or identical. In other words, allowing the model to share parameters only among cluster-related tasks yields better generalization than sharing across all tasks. This idea directly aligns with intuition in BCIs; a model should share features among neurologically similar users, but not necessarily across wildly divergent users.

In summary, population clustering and MTL are complementary; clustering simplifies the inter-subject variability by creating relatively uniform groups, and multi-task learning harnesses the shared structure within those groups (and across groups via higher-level shared layers) to train models that generalize well. Emerging BCI frameworks that integrate both – for example, clustering users by EEG patterns and then applying a multi-task (or transfer learning) model informed by those clusters – have demonstrated superior accuracy and reduced calibration time in comparison to traditional subject-specific training [44, 85]. This combined approach represents a promising pathway toward reduced-calibration or few-shot BCI systems, as it enables the model to learn how to learn from a population, first by recognizing which subgroup a new user belongs to, and then by leveraging the pre-learned representations tuned for that subgroup, thereby drastically minimizing the additional data needed from the new user. The combination of population clustering and MTL effectively capitalizes on shared brain-signal structures at multiple scales, pushing EEG-based BCIs closer to robust, out-of-the-box performance in real-world deployments [85].

2.5 Transfer Learning

Transfer learning (TL) is a machine learning paradigm in which knowledge gained from one domain or task (the source) is leveraged to improve learning performance in a different but related domain or task (the target) [41, 57, 78]. Instead of training a model from scratch on the target domain, which may be constrained by limited data, TL adapts models, features, or representations learned from abundant source data to new conditions, thereby reducing the need for large labeled datasets and shortening training time. TL has found wide application across various domains, including computer vision, natural language processing, and speech recognition, where pre-trained models are routinely adapted to new datasets or tasks. In biomedical signal processing, and particularly in EEG-based systems, TL addresses the challenges of non-stationarity and inter-subject variability by enabling cross-subject, cross-session, or even cross-paradigm adaptation, making it a promising approach for improving generalization and practicality in brain–computer interface applications [78].

Transfer learning has proven to be a transformative strategy in motor-imagery (MI) EEG-based brain–computer interfaces (BCIs), addressing the central challenge of domain adaptation where differences in brain physiology, head anatomy, electrode configurations, and cognitive strategies across subjects and sessions cause models trained on one dataset to perform poorly on another [32, 57]. Traditionally, each user must undergo a lengthy calibration session to collect subject-specific data, which hampers usability and scalability. Transfer learning mitigates this by leveraging data or model parameters from previously recorded subjects (source domains) to improve performance on new users (target domains), offering a way to generalize while minimizing calibration time [50]. This trade-off, improving generalization with less target data, lies at the core of transfer learning's appeal in MI-BCIs [32, 73].

A variety of transfer learning strategies have been proposed in recent years to address the distributional differences between source and target EEG data, which are caused by inter-subject variability and signal non-stationarity. These approaches generally fall into three broad categories: (1) preprocessing-based alignment methods that project data into a common feature space, (2) deep learning frameworks that learn domain-invariant representations, and (3) optimization techniques designed to preserve the intrinsic structure of EEG data across domains. Within these categories, techniques such as Riemannian and Euclidean alignment, label alignment, adversarial domain adaptation, and manifold or optimal transport–based mappings have consistently improved cross-subject and cross-session performance in MI-BCI settings [13, 16, 28, 53, 83].

Another widely adopted approach involves fine-tuning pre-trained deep neural networks. In this paradigm, a model, often a convolutional architecture such as EEGNet [69], is first trained on a large, diverse subject pool and subsequently adapted to a new user using only a small number of calibration trials [21, 52]. Shared feature extraction layers capture population-level EEG patterns, such as sensorimotor rhythms, while fine-tuning selectively adjusts higher-level layers to enable rapid personalization.

More recently, meta-learning approaches, such as Model-Agnostic Meta-Learning (MAML), have been explored to accelerate adaptation to unseen users further. By explicitly training models to learn efficiently from limited data, MAML enables rapid calibration with only a few gradient updates [22]. Similarly, emerging few-shot learning strategies integrate small labeled datasets with unlabeled data streams via unsupervised fine-tuning, moving toward reduced-calibration BCIs while preserving classification accuracy [42].

While existing transfer learning methods have improved cross-subject generalization in MI-BCIs, many either assume that subjects can be modeled uniformly using a single shared representation or require substantial calibration data to achieve reliable adaptation. Such limitations leave open the challenge of developing strategies that balance generalizable representation learning with efficient personalization, particularly under constraints such as limited channels and a small number of labeled trials.

In conclusion, transfer learning is a key technique in MI-EEG BCI systems, enabling rapid personalization and enhancing cross-subject generalizability with minimal calibration effort. Through feature alignment, fine-tuning, and adaptive learning, TL mitigates the effects of domain shift (i.e., systematic differences between training and target distributions due to subjects, sessions, or hardware) while harnessing collective knowledge from prior users. Cluster-based modeling, multi-task learning, and lightweight subject-specific adaptation are complementary strategies that can further enhance calibration and facilitate the translation of offline-trained models to operational settings. While challenges such as negative transfer, limited data, and non-stationarity must be carefully managed, evidence from foundational and recent studies underscores the substantial benefits of well-designed TL pipelines for practical, effective MI-BCI systems.

Chapter 3

Methods

This chapter outlines the methodological framework employed in this project and, where relevant, presents intermediate results that inform subsequent design choices. We compare (i) a raw-EEG CNN baseline (Deep4Net), (ii) a multi-task variant with a shared encoder and lightweight heads, and (iii) a transfer-learning scheme that adapts the encoder to new users. Across all experiments, transforms are fit on TRAIN only (no leakage; the held-out subject is excluded from training/validation in LOSO), and performance is reported on TEST per subject, including accuracy, κ, precision, recall, and F1-score.

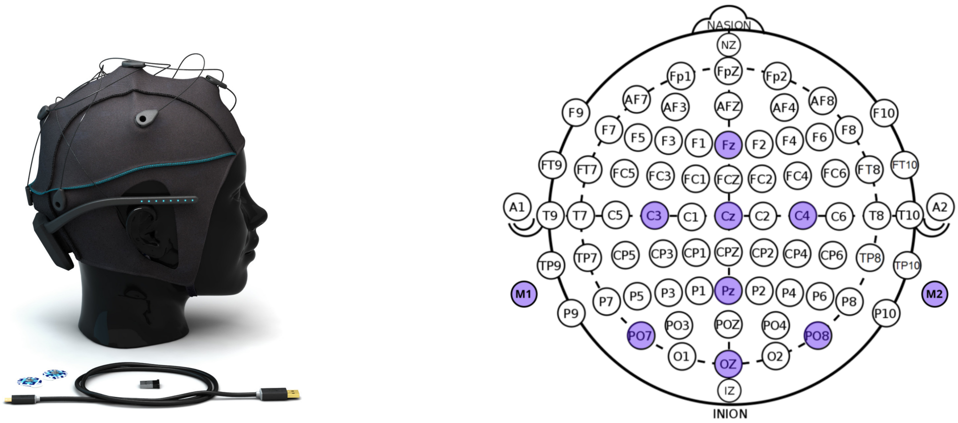
3.1 EEG Acquisition and Paradigm

Because the EEG acquisition stack and motor imagery (MI) paradigm were adopted from existing tools and are not novel contributions of this work, only the essential information necessary to interpret the offline analyses is presented in this section. Full implementation details are provided in Appendix A for reproducibility.

3.1.1 EEG headset

Prospective recordings were obtained using the Unicorn Hybrid Black [75], an eight-channel portable EEG system conforming to the international 10–20 montage. Active electrodes were placed at Fz, C3, Cz, C4, PO7, Oz, PO8, and Pz, matching the layout used throughout this project; two adhesive reference electrodes (M1, M2) were placed on the mastoids. The amplifier sampled at 250 Hz with 24-bit resolution and an input range of approximately ±750 mV, and communicated with the host computer via Bluetooth.

To ensure stable signal quality, gel-based electrodes were used in all sessions. Before each recording, electrode seating and contact quality were verified using the vendor's impedance/quality indicators, and adjustments were made as needed to minimize impedance and slow drifts.



(a) 8-channel wireless EEG headset.

(b) Electrode positions according to the 10-20 system.

Figure 3.1: Unicorn hybrid black system [75].

3.1.2 Acquisition Setup and MI Paradigm

EEG data and event markers were transmitted via Lab Streaming Layer (LSL). The vendor application published the EEG stream, while the paradigm/game emitted markers on a separate LSL stream. LSL's time-correction provided a common clock, enabling alignment of cue onsets and EEG samples at single-sample resolution. All streams were persisted in MNE-compatible FIF format, with annotations derived from the marker stream.

Data collection relied on two existing Python-based applications: The *recorder* (developed by Manuel Weiss), which handled EEG acquisition, real-time channel monitoring, event labeling, and data persistence; and the *paradigm/game* (by Annina Bazzigher and Zoe Widmer), built on Pygame to deliver visual cues for left-hand MI, right-hand MI, and rest conditions. Although the paradigm supports feedback, this functionality was not used in this project. Both tools were adopted without modification. Closed-loop inference was prototyped in the recorder but not employed for any evaluation reported here. Implementation details are provided in

Appendix A.

The MI paradigm followed a cue-based design. Each trial began with a visual cue indicating the required task (left-hand imagery, right-hand imagery, or rest). Participants performed the task for a fixed period while an EEG was recorded. Trial timing, cue onset synchronization, and marker emission were fully controlled by the paradigm/game. The complete acquisition schedule is provided in Appendix A.

3.2 Datasets

This project utilizes two datasets serving complementary roles: a large, harmonized publicly available dataset (the *source dataset*) used for model pretraining, clustering, and evaluation under subject-independent protocols, and a smaller, prospectively acquired dataset (the *target dataset*) used to evaluate cross-dataset transfer and rapid personalization.

3.2.1 Source Dataset

The training dataset used in this project was assembled by Bazzigher and Widmer [9] through the aggregation and harmonization of multiple public motor-imagery (MI) EEG datasets, utilizing a unified schema. The primary sources comprise PhysioNet MI [23] (109 subjects; 5 MI tasks), High-Gamma MI [66] (14 subjects; 4 MI classes), Stieger [72] (62 subjects; 7–11 runs per subject; 4 MI tasks), and Weibo [82] (10 subjects; 7 MI tasks) (Table 3.1). Additional candidate sets that met channel-count and quality criteria but lacked explicit rest labels (e.g., Cho, Lee, Liu, Shin, A, Grosse Wentrup) were excluded to preserve a consistent three-class label space (left hand, right hand, rest).

Because sources differed in hardware, sampling rates, montages, cueing schemes, and recording conditions, a standardized harmonization pipeline was applied before release. Channel subsets were mapped to the eight-channel target montage (with spatial interpolation/derivation where exact placements were unavailable); sampling rates spanning 160–1000 Hz were normalized by downsampling higher-rate recordings; and power-line interference was mitigated using dataset-appropriate notch filters (50 Hz for European/Asian recordings; 60 Hz for U.S. recordings). Dataset-specific adjustments followed the same principles (PhysioNet [23]: 60 Hz notch, reduction to 8 channels, retention of left/right/rest; Weibo [82]: 50 Hz notch, reduction to 8 channels; Stieger [72] and High-Gamma [66]: notch filtering and reduction to 8 channels with subsequent resampling).

After harmonization and curation, the delivered working dataset comprises 85 subjects stored as MNE/FIF files under a unified schema: sampling rate 200 Hz; eight EEG channels (Fz, C3, Cz, C4, PO7, Oz, PO8, Pz); and a three-class label space with standardized event codes $\{0 = \text{left hand}, 1 = \text{right hand}, 2 = \text{rest}\}$. Consistent annotation names map to these integer labels, simplifying downstream processing. No additional content changes were made by the present author before analysis; subsequent transformations (e.g., resampling for model compatibility, epoching, and standardization) are specified in Section 3.3.1. This harmonized multi-cohort dataset is referred to as the *source dataset* throughout this work.

Table 3.1: MI DATASETS USED TO CONSTRUCT THE SOURCE DATASET.

Dataset	#EEG ch.	Events (subset)	f_s	Total subjects
PhysioNet MI [23]	64	rest, LH, RH, feet, both hands	160 Hz	109
High-Gamma MI [66]	128	rest, LH, RH, feet	250 Hz	14
Stieger [72]	60	rest, LH, RH, both hands	1000 Hz	62
Weibo [82]	60	rest, LH, RH, both hands/feet	200 Hz	10

Composed by Bazzigher and Widmer [9]; values compiled from the original publications [23, 66, 72, 82].

3.2.2 Target Dataset

Prospective recordings were collected with the Unicorn Hybrid Black [75]. An acquisition summary is provided in Section 3.1.2 (for full implementation detail see Appendix A. Two subjects were recorded on two separate days, with two sessions per day, yielding a total of eight sessions (2 subjects × 2 days × 2 sessions). Signals were sampled at 250 Hz from eight EEG channels (Fz, C3, Cz, C4, PO7, Oz, PO8, Pz) with mastoid references (M1, M2) and saved per session in MNE-compatible FIF format, along with annotations, for downstream processing.

Each session followed a fixed motor-imagery schedule, consisting of alternating rest periods and cued blocks of left- or right-hand imagery. Per session, this produced 12 left-MI and 12 right-MI trials, interleaved with predefined rest intervals. Event markers were synchronized to cue onset via LSL and fused into the recordings as annotations to ensure sample-level alignment.

The dataset adopts a three-class label space, left hand, right hand, rest, mapped to integer codes {0,1,2} for consistency with downstream analyses; rest trials are not used for supervised decoding but are retained for diagnostics and clustering.

Basic post-acquisition quality control was performed after each session to confirm data integrity before preprocessing: (i) visual inspection of raw traces for continuity, amplitude plausibility, and channel ordering; (ii) channel-wise variance/RMS summaries to identify flat or saturated channels; and (iii) power spectral density estimates to verify suppression of line interference and the presence of sensorimotor μ/β activity ($\approx$ 8–30 Hz). Sessions passing these checks were retained for analysis. This dataset is used exclusively for offline adaptation and evaluation; no closed-loop inference results are reported. This prospectively recorded dataset is referred to as the *target dataset* throughout this work.

3.3 Experimental Setup

This section formalizes the experimental design used to evaluate cross-subject MI decoding. We compare (i) a raw-EEG convolutional baseline (Deep4Net), (ii) a multi-task variant with a shared encoder and lightweight heads, and (iii) a transfer-learning scheme that adapts the shared encoder to new users. Three protocols are considered: pooled training with subject-disjoint validation, leave-one-subject-out (LOSO) zero-shot transfer, and LOSO few-shot adaptation with a small number of calibration trials per class. Unless stated otherwise, all preprocessing statistics, feature transforms, clustering (when used), and model selection are derived strictly from TRAIN data; in LOSO protocols, the held-out subject contributes no samples to training. Performance is reported per subject on TEST with accuracy as the headline metric, complemented by Cohen's κ and macro-precision/recall/F1-score, and results are averaged over independent seeds.

3.3.1 Preprocessing

Effective preprocessing is essential for the success of subsequent experiments and determines the reliability of both decoding and adaptation results. This section presents the default preprocessing procedures applied throughout the study, which are standardized to ensure comparability across sources and experiments while preventing train–test leakage. The source dataset is delivered at 200 Hz with eight EEG channels (Fz, C3, Cz, C4, PO7, Oz, PO8, Pz); the target dataset is recorded at 250 Hz with the same montage and mastoid references (M1, M2).

Raw Signal Preprocessing

All analyses operate at a unified sampling rate of 250 Hz. Source dataset files are resampled from 200 Hz using zero-phase resampling with automatic padding; target dataset files remain at their native 250 Hz. To avoid double filtering, no additional notch or band-pass is applied to the source dataset beyond its documented construction. For the target dataset, a 50Hz notch filter was used to suppress power-line interference, followed by an 8–30Hz IIR band-pass filter to isolate sensorimotor rhythms.

Independent component analysis (FastICA, eight components, fixed seed) is configured for the target dataset to support diagnostic visualization; component removal is *not* performed in reported runs, and neither dataset includes dedicated EOG channels. Before epoching, each session undergoes basic integrity checks (continuity, amplitude plausibility, and channel ordering), channel-wise variance/RMS screening for flat or saturated channels, and PSD inspection to verify line-noise suppression and energy in the 8–30 Hz band.

File handling mirrors acquisition structure. Source dataset recordings are processed separately for each subject and file, without concatenation. Target dataset runs from the same subject are concatenated before epoching so that cue timing and rest intervals are preserved across successive runs.

Epoch Preprocessing

Epochs are time-locked to the cue onset for the three classes. For each event, a window from -1.0 s to +2.0 s is extracted, yielding a 3.0 s analysis segment. Within each epoch, fixed-length crops of 3.0 s are generated with a 0.5 s hop. This sliding procedure standardizes input duration and increases the number of training examples while keeping the temporal context aligned with cue-related activity cite Schirrmeister2017. No baseline correction is applied.

Channel-wise exponential moving standardization (EMS) is applied to mitigate slow drifts and changes in variance. To prevent leakage, EMS parameters are estimated per subject on the TRAIN partition only and then reused unchanged for the corresponding TEST partition.

All preprocessed outputs are persisted as serialized MNE `Epochs` objects, along with the configuration that produced them (dataset, sampling rate, channel list, epoch limits, sliding parameters, standardization settings, and random seed), ensuring the exact regeneration of results. After preprocessing, several complementary feature families are explored to capture both spectral and spatial information from the EEG.

3.3.2 Feature Extraction

Feature extraction maps preprocessed epochs to fixed-length vectors suitable for classification and transfer learning. The implementation supports several families of features whose outputs can be concatenated before dimensionality reduction and selection. The principal methods are event-related desynchronization/synchronization (ERD/ERS) band-power contrasts, common spatial patterns (CSP), filter-bank CSP (FBCSP), and Riemannian geometry-based representations.

ERD/ERS band-power: ERD/ERS quantifies relative power change between a baseline and analysis window within physiologically motivated bands. Baseline and analysis windows are defined on the epoch time axis and clamped to available samples if necessary. By default, baseline and analysis windows are $[0.0, 0.5]$ s and $[0.5, 4.0]$ s, and the μ and β bands are $[8, 12]$ Hz and $[13, 30]$ Hz. For each band, epochs are band-limited using MNE's filtering, and the mean-squared amplitude is computed over channels and time for both windows. A per-epoch, per-band ERD/ERS score is formed as a percent change relative to baseline; the vector length equals the number of bands.

Common Spatial Patterns (CSP): CSP is applied directly to epoch matrices and requires at least two classes. The implementation uses MNE's CSP with a configurable number of spatial filters (four retained in the main configuration), no regularization, and no trace normalization. When preprocessing includes an 8–30 Hz band-pass, CSP operates on the band-limited signal; otherwise, it uses broadband epochs.

Filter-Bank CSP (FBCSP): FBCSP extends CSP by decomposing the signal into contiguous frequency bands and applying CSP within each band. In the reference configuration, bands cover ranges spanning low to high $beta$ ([4–8], [8–12], [12–16], . . . , [32, 38]), retaining two spatial filters per band. Outputs are concatenated to form a single vector. FBCSP is used in ablations to examine the trade-off between spectral resolution and complexity; the default pipeline in reported results employs single-band CSP.

Riemannian features: These features were obtained by estimating a covariance matrix with OAS shrinkage and mapping it to the tangent space at the manifold reference point to get a Euclidean vector suitable for linear classifiers. Upper-triangular vectorization without mapping is available; however, the tangent embedding was preferred in exploratory trials due to its greater stability with small sample sizes.

When multiple methods are enabled, feature outputs are concatenated along the feature dimension to produce a per-session matrix. For training, per-session matrices from all subjects' TRAIN partitions are vertically stacked to form a pooled set. A standard scaler is fitted on this pooled TRAIN set and applied uniformly to all matrices, including held-out TEST portions, thereby fixing a common affine normalization learned strictly from training data. Dimensionality reduction uses PCA fitted on the pooled TRAIN features; unless configured otherwise, the target is set to retain 95% variance (skipped when dimensionality is below a threshold, 300 by default).

Feature selection uses recursive feature elimination with cross-validation (RFECV) built around a linear SVC (step size 30; 3-fold CV; accuracy scoring; minimum 10 retained features). RFECV is fit once on the pooled TRAIN features, after applying the training-fitted scaler and, when enabled, PCA, to produce a boolean mask that identifies the retained dimensions.

For the target dataset, the exact scaler, PCA projection, and RFECV mask learned on the pooled TRAIN set is loaded and applied unchanged before any modeling or analysis. This process projects the features of the target subject into the same space as the source dataset, thereby preventing leakage from the target subjects into the feature selection process. In acquisition-only evaluations, these saved transforms (scaler/PCA/mask) are reused without refitting. Final per-session matrices, therefore, consist of standardized, optionally PCA-reduced features, restricted by the fixed RFECV mask, paired with the original epoch labels. All feature artifacts are persisted to support exact reproduction of results.

3.3.3 Population Clustering

Inter-subject variability is a central challenge in EEG-based motor imagery decoding, as differences in brain physiology, electrode placement, and cognitive strategy can degrade cross-subject generalization. To address this, subject clustering was explored as a strategy to stratify the population into more homogeneous subgroups before model training. By grouping subjects with similar EEG representations, the approach aims to exploit shared structure within clusters while limiting negative transfer between dissimilar users.

Clusters are formed in the fixed feature space defined by the preprocessing and feature-extraction pipeline (Section 3.3.1, 3.3.2) using transforms learned *only* on the pooled TRAIN subjects to avoid leakage. For each subject, all epoch-level feature vectors are: (i) standardized with the global scaler, (ii) projected via PCA fitted on the pooled TRAIN set (when enabled), and (iii) restricted by the RFECV mask learned on the same TRAIN subset. The resulting vectors are averaged across runs and splits to produce a single, fixed-dimensional subject-level representation, which serves as the input to clustering.

The implementation supports centroid-based, hierarchical, and density-based clustering methods. Following results from prior work, k-means is used as the default method with k-means++ initialization, a maximum of 300 iterations, 10 random restarts, and a fixed random seed of 42. Agglomerative clustering with Ward linkage provides a hierarchical alternative when Euclidean structure is appropriate, and DBSCAN offers a density-based option with Euclidean distance ($\varepsilon = 1.0$), a minimum of five samples, and a leaf size of 30. All hyperparameters are exposed via configuration and were varied in exploratory analyses. The special case $k = 1$ serves as a pooled baseline, effectively collapsing clustering into a single group to quantify the marginal value of stratification.

The number of clusters is selected using internal metrics and visual diagnostics. For k-means, inertia and the silhouette coefficient are computed across candidate k values, and scree plots of PCA eigenvalues are inspected to estimate intrinsic dimensionality. Cluster-size distributions are checked to avoid degenerate splits, while low-dimensional embeddings (such as PCA or t-SNE) overlaid with baseline subject performance provide qualitative insights into separability.

Cluster assignments are integrated into the multi-task architecture by conditioning the classifier head on a subject's cluster identity. During training, each mini-batch is associated with its corresponding cluster index, and examples are routed through the shared backbone to their assigned cluster-specific head. After training, the fitted clustering model, preprocessing transforms, and cluster assignments are serialized as part of a cluster wrapper alongside network weights. With k-means, new subjects are assigned to the nearest centroid using the saved model, enabling consistent head selection during transfer. (Hierarchical and DBSCAN variants are retained in implementation for exploratory analyses but do not currently support out-of-sample assignment.) An optional restriction flag enables evaluation confined to within-cluster adaptation.

For quantitative validation and model selection, multiple metrics are reported: the silhouette score (higher is better), the Davies–Bouldin index (lower is better), and the Calinski–Harabasz score (higher is better), alongside an inertia-based elbow-curve inspection. Stability is assessed by repeating k-means with different random seeds and via bootstrap resampling of epochs within subjects; partitions are compared using the adjusted Rand index. Cluster-size balance and baseline subject-accuracy distributions are also monitored to avoid degenerate solutions. Numerical results and the selected k for each feature set are presented in the Results section.

3.3.4 Data Augmentation

Augmentation is applied to potentially improve generalization in the presence of inter-subject differences and limited labeled data. In all experiments, augmentations are applied *only* to the TRAIN split; validation and TEST inputs remain unchanged. Subject clustering is computed from features extracted from unaugmented data to avoid altering group structure. Raw-signal augmentations are evaluated during multi-task learning; a batch-level interpolation (mixup) is assessed during transfer learning. Exactly one augmentation is enabled per run to isolate its contribution, and comparisons are reported under identical model settings and random seeds in the pooled-only regime.

Gaussian noise: Gaussian noise models sensor and impedance fluctuations by adding zero-mean perturbations to each channel. Conceptually, this encourages the encoder to become invariant to small amplitude variations that do not carry task information. Concretely, for an epoch array of shape (channels, time), we compute each channel's within-epoch standard deviation and add i.i.d. Gaussian noise scaled by σ times that standard deviation (with a 10^{-8} offset for numerical stability). The configuration uses $\sigma = 0.02$ ($\approx 2\%$ of per-epoch channel amplitude). Noise is applied on-the-fly at batch construction and never at validation/test time.

Time warping: Time warping introduces local dilations or compressions to emulate latency jitter and tempo variability in motor imagery. The transformation selects a single contiguous segment within the epoch by uniformly sampling a start index and fixing a segment length up to one-half of the epoch duration. That segment is resampled using SciPy's resampler to a length scaled by a factor drawn uniformly from $[1 - 0.15, 1 + 0.15]$ and then reinserted at its original location. If resampling shortens the segment, zero-padding restores the original segment length; if it lengthens the segment, trailing samples are cropped so that the overall epoch length remains unchanged. This procedure produces small, realistic timing perturbations while preserving boundaries and label alignment.

Frequency shift: Frequency shifting simulates modest drifts in dominant rhythm frequencies that can arise from state changes or slight electrode displacements. The approach computes the analytic signal along the time axis via the Hilbert transform for each channel, multiplies it by a complex exponential $e^{j2\pi ft}$ using a common carrier for all channels, and finally takes the real part. With a sampling frequency of 250 Hz and a configured shift magnitude of ±1 Hz, this results in a controlled spectral translation without circular wrap-around. The operation is applied independently to each epoch and preserves phase continuity.

Mixup: Mixup is employed at the transfer stage, as a label-aware interpolation that augments decision boundaries rather than raw waveforms. For a mini-batch X with labels y, a coefficient λ is sampled from $\mathrm{Beta}(\alpha, \alpha)$ with $\alpha = 0.2$, and a random permutation is generated on-device. Training proceeds on $\tilde{X} = \lambda X + (1 - \lambda)X'$ and the loss is formed as a convex combination of cross-entropy terms,

$$L = \lambda \cdot \mathrm{CE}(f(\tilde{X}), y) + (1 - \lambda) \cdot \mathrm{CE}(f(\tilde{X}), y')$$

Mixup is applied during pooled fine-tuning and few-shot calibration of subject-specific heads, when enabled. Evaluation uses unmodified inputs, and no raw-signal augmentations are used at the transfer stage.

Each augmentation is toggled in a dedicated run while keeping preprocessing, architecture, optimizer, subject splits, and seeds fixed. Augmentations are applied only during batch construction on TRAIN; validation and textsc test remain unchanged.

Clustering always uses features extracted from unaugmented data. Although the raw-signal pipeline can compose noise → warp → shift, experiments enable at most one method at a time. Configuration files record the active augmentation and parameters (σ for noise, warp ratio, and maximum segment fraction for time warping, shift magnitude for frequency shifting, and α for mixup) to ensure exact reproduction.

Table 3.2: Augmentation configurations used during training.

Augmentation	Parameter	Value
Gaussian noise	σ (relative to epoch std)	0.02
Time warp	Local warp ratio	±0.15
	Max warped segment	0.5 of 600-sample epoch
Frequency shift	Sampling rate f_s; shift	250 Hz; ±1 Hz
Mixup	Beta parameter α	0.2

The extracted features, with or without augmentation, are then used to train and evaluate three complementary modeling approaches: a baseline Deep4Net model, a multi-task learning framework with cluster-specific heads, and a transfer learning approach that adapts these models to unseen subjects.

3.4 Classification Models

3.4.1 Deep4Net

Deep4Net is a convolutional architecture tailored for raw EEG that stacks four convolution–pooling blocks and concludes with a dense softmax layer [66]. The first block factorizes the entrance transformation into a temporal convolution followed by a spatial convolution across electrodes without an intervening nonlinearity, regularizing the input mapping by decoupling temporal filtering from spatial unmixing. Subsequent blocks are conventional conv–pool stages. Exponential linear units (ELUs), batch normalization, and dropout are integral design choices that have been shown to stabilize optimization in this family of models. The original work also introduced "cropped training", i.e., learning from dense, sliding windows within trials, to increase the number of supervised examples and to trade off decoding delay against accuracy in online scenarios.

Deep4Net fulfils two complementary functions in this work. First, it serves as a standalone baseline classifier, trained either separately for each subject or once on

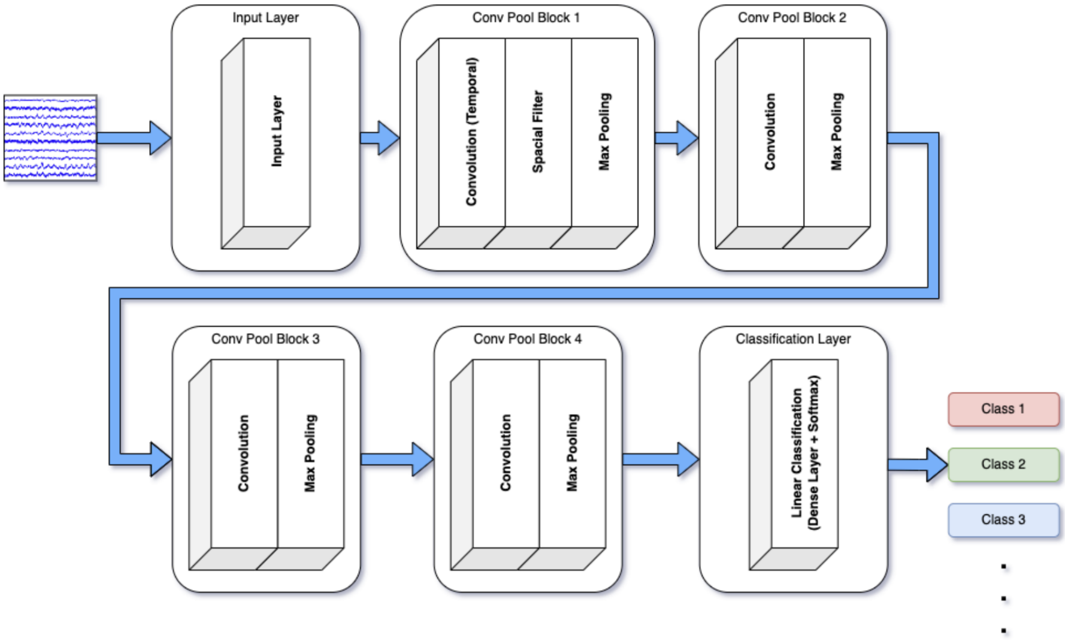


Figure 3.2: Architecture of Deep4Net [65].

the pooled cohort, always using the same raw windows and fixed train-test partitions; trial-level decisions are obtained by averaging window-level logits within a trial. Second, the same convolutional trunk is reused as a shared encoder in the subsequent multi-task and transfer settings, where lightweight subject- or cluster-conditioned heads are attached on top. Localizing adaptation to these heads, while keeping the trunk frozen or updating it at a markedly slower rate, provides a controlled way to compare a purely supervised baseline against models that leverage shared representations for cross-subject generalization and rapid per-user calibration.

The complete training and reporting procedures for the standalone baselines are specified in Section 3.5. Building on the same encoder, the following subsections introduce the multi-task and transfer frameworks that overlay subject/cluster heads and define how adaptation is performed.

3.4.2 Multi-Task Learning Model

In addition to a standalone baseline defined in Sections 3.4.1 and 3.5.2, Deep4Net also serves as the *backbone* of the MTL/TL pipeline designed in this work. In the MTL stage, the Deep4Net trunk is shared across tasks, while cluster-conditioned heads specialize in decision boundaries. In TL, the same trunk is reused (optionally frozen or trained at a reduced learning rate) and augmented with subject or cluster heads, depending on the transfer mode. Anchoring both MTL and TL against the single-subject and pooled Deep4Net baselines isolates the contribution of subject-aware routing and parameter reuse.

The multi-task learning (MTL) phase is a representation-learning stage whose goal is to train a subject-agnostic convolutional trunk (Deep4Net) while allowing lightweight specialization at the decision layer. Each "task" is defined by a subject cluster; consequently, MTL explicitly optimizes a shared backbone on pooled data and a small classifier head for each cluster. This yields two artifacts that the transfer stage depends on: (i) a pretrained backbone that encodes motor-imagery structure shared across users; and (ii) a bank of *cluster heads* that capture cluster-specific decision boundaries.

In addition to the shared trunk, the network maintains one lightweight multilayer perceptron head per cluster. Each head receives the backbone feature vector and comprises a linear layer with 128 hidden units, ReLU activation, dropout with a rate of 0.5, and a final linear layer that produces two logits (left vs. right). These are the *cluster heads* exported by the MTL stage and later reused by the transfer-learning (TL) stage for zero-shot inference or as initialization for subject-specific heads. During a forward pass, each trial carries the integer cluster index of its subject; mixed-cluster mini-batches are handled by routing each sample to its corresponding head while sharing the backbone.

Cluster assignments are computed once, before MTL training, from fixed, unaugmented features to prevent altering the group structure. Concretely, all epoch-level feature vectors available for a subject are standardized by the pooled train-fitted scaler, optionally projected by PCA, and pruned by the RFECV mask (all learned strictly on the pooled TRAIN set, see Section 3.3.2). Averaging across epochs and splits yields a single subject-level representation. K-means is then applied to these representations with configuration-controlled hyperparameters (see Section 3.3.3). The fitted model, scaler/PCA/mask, subject representations, and the subject–cluster maps are persisted as a cluster wrapper alongside MTL weights to enable consistent head selection in TL.

Training uses pooled TRAIN windows across subjects with subject and cluster labels preserved. Mini-batches are shuffled from the pooled set. When augmentation is enabled for MTL (see Section 3.3.4), raw-signal transforms are applied on-the-fly to TRAIN windows only; validation and TEST inputs remain unmodified. Optimization uses Adam with weight decay 10^{-3}; runs use batch size 64, 100 epochs (no early stopping), and three independent runs are launched with seeds 42–44. After each run, evaluation is performed on per-subject TEST splits, and predictions are saved with ground-truth labels for downstream TL analyses. Model checkpoints include the shared backbone and all cluster heads.

The MTL stage learns a subject-agnostic temporal–spatial representation in the shared trunk while allowing cluster heads to specialize decision boundaries for subpopulations. This produces two artifacts consumed by TL: (i) a pretrained backbone encoding motor-imagery structure that can be frozen or gently fine-tuned, and (ii) a set of cluster heads (numeric keys `"0".."k-1"`) that support zero-shot routing and cluster-conditioned few-shot initialization. In the limit $k = 1$, MTL reduces to a standard multi-subject classifier with a single shared head.

The training logs, per-run predictions, MTL weights, and the clustering wrapper (model, preprocessing transforms, and assignments) are persisted. This ensures that TL can deterministically recover head indices, can reassign new subjects in a consistent feature space, and overlay additional subject-specific heads without disturbing the shared representation.

3.4.3 Transfer Learning Model

Transfer learning in this project builds on a multi-task pretrained Deep4Net backbone trained on the *source dataset* and adapts it to unseen subjects drawn from either the source or the *target dataset.* The `TL Model` wraps the trained `MTL Model`, exposing the shared backbone together with a dynamic registry of classification heads. Unless explicitly disabled, transfer begins from the pre-trained MTL weights, allowing adaptation to proceed from a trunk that already encodes subject-invariant motor imagery representations.

Internally, the `TL Model` retains the pretrained MTL backbone and maintains a keyed registry of lightweight MLP heads. Each head maps the backbone feature vector to two logits via a simple architecture (`Linear` → `ReLU` → `Dropout 0.5` → `Linear`). Heads can represent individual subjects, entire clusters, or newly added users. At initialization, the TL model reconstructs the shared backbone and any cluster heads stored in the MTL checkpoint. If a head does not exist for the requested subject or cluster, a new one is automatically allocated. This design localizes adaptation: the backbone can remain fixed or be updated at a reduced rate, while the newly added heads learn more quickly, minimizing the risk of overfitting when calibration data are scarce.

Four evaluation protocols are supported, each probing a distinct adaptation regime:

- *Pooled fine-tuning*: labeled training trials from all subjects (including the target) are pooled to fit a single decision layer on top of the shared backbone. This provides an optimistic upper bound when extensive calibration data are available under matched conditions.
- *Hold-out zero-shot*: the target subject is excluded from training and validation and is evaluated without any calibration. This directly measures subject-independent generalization.
- *Hold-out few-shot*: starting from weights trained on all non-held subjects, a new subject-specific head is calibrated using only a small, fixed number of labeled trials per class from the held subject, evaluating the effectiveness of adaptation under limited calibration.
- *In-session transfer*: for the target dataset, a brief calibration prefix at the start of each session is used to fit a subject-specific head; inference then proceeds in strict temporal order over the remainder of the session. This approximates a deployment scenario where only a short early-session calibration is available.

All protocols enforce strict subject disjointness between training and validation to prevent trial-level leakage. Evaluation always uses the held subject's TEST split, reporting accuracy, Cohen's κ, and macro-averaged precision, recall, and F1-score. Training loop details, optimizer configurations, and early-stopping strategies are described separately in Section 3.5.

Comparing these protocols side by side reveals the performance decomposition: pooled fine-tuning reflects the optimistic upper bound, zero-shot evaluates subject-independent generalization without calibration, few-shot assesses the effectiveness of adaptation under limited calibration, and in-session transfer approximates a realistic deployment setting. Their relative differences reveal the quality of subject-invariant representations, the magnitude of domain shift, and the benefit of lightweight, head-localized adaptation on top of the pretrained backbone.

3.5 Training and Evaluation Protocols

This section formalizes the training and evaluation of models, as well as the realization of subject transfer on top of the multi-task backbone. All procedures are aligned with the preprocessing and model definitions introduced earlier, and every protocol is executed with the same optimizer, batch size, and early-stopping discipline to ensure comparability. Unless otherwise stated, input windows comprise eight EEG channels and 750 samples, the number of classes is two, and the classifier is a Deep4Net backbone with an MLP head. Optimization uses Adam with a weight decay of 0.001, a batch size of 64, and a two-rate schedule in transfer learning (head learning rate 1e-3 and backbone learning rate 1e-5 when the backbone is unfrozen). Early stopping monitors a subject-disjoint validation split. Randomness is controlled through three independent runs, with seeds starting at 42; results are aggregated per subject across runs for reporting purposes. Augmentation is disabled by default at transfer time, except when explicitly enabled.

3.5.1 Cross-Subject Partitioning and Transfer Protocols

This subsection specifies the partitioning of trials for training, validation, and testing across multi-task and transfer learning, as well as the safeguards that enforce cross-subject separation.

MTL split (intra-subject): Each subject's subepochs are randomly divided into TRAIN and TEST partitions in a 70/30 ratio using a fixed seed. EMS parameters are fitted on TRAIN only and applied to the corresponding TEST portion, so normalization statistics never include test samples. When multiple runs exist for the same subject, the runs are concatenated before splitting, allowing the 70/30 partition to operate on the combined recording.

Subject-level representations used for optional clustering are computed from the feature matrices by averaging per-trial feature vectors for each subject. Clustering does not affect routing and is retained solely for compatibility with multi-cluster ablations. Under multi-cluster settings, representations should be derived from TRAIN features only to maintain strict separation.

TL protocols (cross-subject): The TL stage reuses the MTL-pretrained Deep4Net backbone and evaluates three complementary cross-subject protocols that differ only in how the target subject's data is used:

- *(i) Pooled fine-tuning:* Training uses the pooled TRAIN trials from all subjects (including target training data), and evaluation is performed on each subject's TEST trials. Because the target subject contributes labeled TRAIN data, this setting provides an optimistic upper bound when a realistic calibration session is available under matched conditions; performance primarily reflects backbone capacity and the robustness of a single shared decision layer after exposure to population variability.
- *(ii) Hold-out zero-shot:* Training uses TRAIN trials from all subjects except the held subject, and evaluation is performed on that held subject with no calibration. The gap to pooled fine-tuning indicates the amount of subject-specific variability that remains after pretraining and pooled adaptation. If multiple clusters are used, the held subject can be routed to the nearest cluster head; with a single cluster, inference reduces to applying the shared head without adaptation.
- *(iii) Hold-out few-shot:* Starting from the state trained on all non-held subjects, the backbone is frozen and a new subject-specific head is fine-tuned using a small labeled trial from the held subject ($k\_shot = 4$ per class). Improvements over zero-shot can thus be attributed to decision-layer adaptation rather than representational changes. When clustering is active, the new head may optionally be initialized from the assigned cluster head before few-shot calibration.

In addition to these protocols, the pipeline offers an *in-session transfer* workflow that remains fully offline, emulating deployment on the *target* dataset, where the subject has never been seen during offline training. The system begins with a population-trained backbone that was learned without the target subject. A short, initial calibration prefix (the first k labeled trials per class) is used to fit a fresh subject-specific head, with the backbone kept fixed or updated only at a much smaller rate if enabled. After this brief calibration, parameters are frozen and inference proceeds over the remainder of the session strictly in temporal order: no shuffling, no look-ahead, and no further updates. Only post-calibration trials contribute to the reported test metrics. Preprocessing and normalization are performed based on the estimates obtained from the training cohort and are not refit on the evaluation data. When clustering is active, the new head may be initialized from the nearest cluster head; with a single cluster, it is randomly initialized. This setting approximates a realistic online workflow with minimal early-session calibration followed by fixed-parameter operation, without requiring live streaming.

Across all modes, leakage safeguards are enforced. Validation subjects are disjoint from training subjects during pooled fine-tuning and LOSO training; the held subject contributes no data to training or validation in LOSO protocols; EMS statistics never use TEST data; and augmentation, when enabled, is applied only to TRAIN samples. Each experiment is repeated for three runs with seeded randomness (42, 43, 44), and evaluations are computed solely on the fixed TEST partitions produced by the initial 70/30 intra-subject split. This design yields directly comparable operating points: pooled fine-tuning as an optimistic ceiling with pooled calibration; LOSO zero-shot as plug-and-play generalization without calibration; and LOSO few-shot as the speed of per-user adaptation under a fixed, small labeled trial.

3.5.2 Baseline Experiments

All multi-task and transfestandaloneresults reported in this project are evaluated against a *stand-alone* Deep4Net baseline trained outside the MTL/TL framework, using the same raw windows and the same TRAIN/TEST splits to ensure like-for-like comparisons.

Single-subject baseline: One Deep4Net is trained *per subject* using only that subject's TRAIN windows and evaluated on that subject's TEST, with Adam optimization and three runs with seeds 42–44. Predictions are aggregated to the *trial* level by averaging logits over all windows sharing a trial identifier and taking the argmax. Metrics reported are accuracy, Cohen's κ, precision, recall, and F1-score. This baseline approximates an upper bound when ample labeled data from the target user is available and no population information is shared across users.

Pooled baseline: A single Deep4Net is trained on TRAIN windows pooled across all subjects and evaluated on the pooled TEST. Optimization and evaluation mirror the single-subject setup. This baseline quantifies what a single, non-clustered model can learn from the population without subject-specific heads or transfer.

Both single and pooled baseline configurations use identical convolutional hyperparameters (dropout probability 0.25, 25 temporal and 25 spatial filters in the first block, temporal kernel length 10, pooling length 3), differing only in pooling mode (mean vs. max) to match the training regime; the remainder of the architecture follows the published design [66].

Table 3.3: MEAN DECODING PERFORMANCE FOR SINGLE-SUBJECT VS. POOLED MODELS. Single-subject results: mean ± SD across N=85 participants. Pooled results: mean ± SD across three independent runs on all subjects.

Metric	Single-Subject	Pooled
Accuracy	0.577 ± 0.096	0.695 ± 0.082
κ	0.154 ± 0.183	0.390 ± 0.163
Precision	0.528 ± 0.168	0.745 ± 0.019
Recall	0.577 ± 0.091	0.695 ± 0.081
F_1 score	0.483 ± 0.133	0.671 ± 0.116

These baselines play three roles. First, they sanity-check that the core architecture and preprocessing learn the task under conventional training. Second, the gap between single-subject and pooled performance reveals whether cross-subject aggregation regularizes beneficially or harms subject-specific decoding on this dataset. Third, they establish quantitative targets for the subsequent MTL backbone and TL protocols. Gains over the single-subject baseline indicate benefits from shared representation learning, whereas gains over the pooled baseline indicate added value from subject-aware routing or per-user calibration beyond a single global classifier. The first two roles were examined in prior work that provided the initial prototype of this pipeline and are not investigated in depth here; in this project, they are reported primarily as context for interpreting the transfer-learning results.

3.5.3 Evaluation Metrics

To keep results comparable and interpretable across Baseline, MTL, and TL, this project reports a consistent set of metrics and adopts a clear reporting convention. All metrics are computed on the held-out TEST split at the trial level and—unless noted—are averaged per subject and then summarized across three seeds (mean and standard deviation). When systems are compared (such as MTL vs. Baseline or TL vs. Baseline), the same per-subject metric vectors are contrasted to avoid conflating subject and seed effects.

Accuracy serves as the headline number as it is intuitive and stable for binary MI tasks; however since it might also be misleading under class imbalance, the following metrics are also reported: balanced accuracy (mean of per-class recalls), Cohen's κ (agreement beyond chance), and macro-averaged precision, recall, and F1-score (equal class weighting). When posterior probabilities are available, we include ROC curves and class-wise AUC as diagnostics of ranking quality; these are not used as headline scores. Confusion matrices are plotted with true labels on the y-axis and predicted labels on the x-axis and are always constructed over the full label set (2×2), ensuring comparability even when a class is absent in predictions. Pooled matrices (aggregating subjects) appear alongside per-subject matrices where informative, and cluster-wise matrices are shown in MTL summaries when relevant.

For quantitative comparisons, per-subject means and standard deviations are reported and, where appropriate, complemented with paired, non-parametric significance tests across subjects (e.g., Wilcoxon signed-rank on per-subject accuracy or κ). When such tests are performed, an effect size is also reported, and any correction for multiple comparisons is noted in multi-way contrasts. Finally, to ensure reproducibility, all metric computations use the full set of class labels; confusion-matrix (true=y, pred=x); and every aggregate statistic is traceable to per-subject, per-run CSVs emitted by the evaluators.

Chapter 4

Results

A comprehensive series of experiments was conducted following the methodologies detailed in Chapter 3. This chapter presents a structured summary of the experimental findings while also outlining the solution processes that led to these results. By reporting both the outcomes and the reasoning behind them, the chapter aims to provide a clear understanding of the cause-and-effect relationships between the methods investigated and their performance.

Two complementary datasets are used throughout this chapter. The *source dataset* is a public MI-EEG dataset comprising $N = 85$ subjects, used to train and evaluate the offline transfer learning pipeline under cross-subject scenarios. The *target dataset* consists of recordings from $N = 2$ newly acquired subjects collected with a Unicorn EEG headset, used to assess how well the pretrained source model generalizes to unseen users and to evaluate the benefits of in-session calibration. To further support interpretation, supplementary figures, analysis plots, and extended data tables are provided in Appendix B.

4.1 Subject-Level Representations for Population Clustering

Establishing the most suitable feature family is a first-order decision for this work, as it determines the subject-space geometry and stability which influences downstream cluster-conditioned modelling and transfer learning performance. This section compares four candidate feature extraction methods, CSP, ERD/ERS band-power, FBCSP, and Riemannian geometry features, by the quality and robustness of the subject-space they induce. Throughout these experiments, clustering is performed using k-means (Euclidean distance; k-means++ initialisation) on z-scored per-subject embeddings.

4.1.1 Candidate Feature Spaces for Subject Embedding

The four candidate methods were evaluated across a shared sweep of cluster counts ($k \in \{2, 3, 4, 5, 6, 8\}$). For each method, the best-performing k was retained based on an aggregate measure of grouping quality. Evaluation used four complementary criteria: Silhouette and Calinski–Harabasz indices (higher is better), within-cluster sum of squares (Inertia; lower is better), and assignment stability under resampling quantified by the Adjusted Rand Index (ARI) (Table B.1).

Across methods, the composite performance profile clearly favoured CSP. Aggregating scores across k placed CSP first (composite score 3.0 ± 0.52), with ERD/ERS as the only competitive alternative (3.2 ± 2.26), while the Riemannian representation consistently underperformed(7.4 ± 0.77). FBCSP occasionally achieved attractive geometry scores but repeatedly produced singleton clusters, making it unsuitable for robust population stratification.

At each method's optimal k (by Silhouette), the contrasts were sharp (Table 4.1). CSP at $k = 3$ produced well-separated, stable clusters with manageable size imbalance, making it the most reliable choice. ERD/ERS exhibited sharper boundaries but suffered from unstable assignments and severely skewed cluster sizes, while the Riemannian representation failed to separate subjects meaningfully.

These numerical differences are visually reflected in the subject-space embeddings (Figure 4.2); CSP forms three compact, well-separated clusters with few outliers, whereas ERD/ERS yields one diffuse majority group with two tiny satellites, and the Riemannian representation shows almost complete overlap between subjects (Figure B.1).

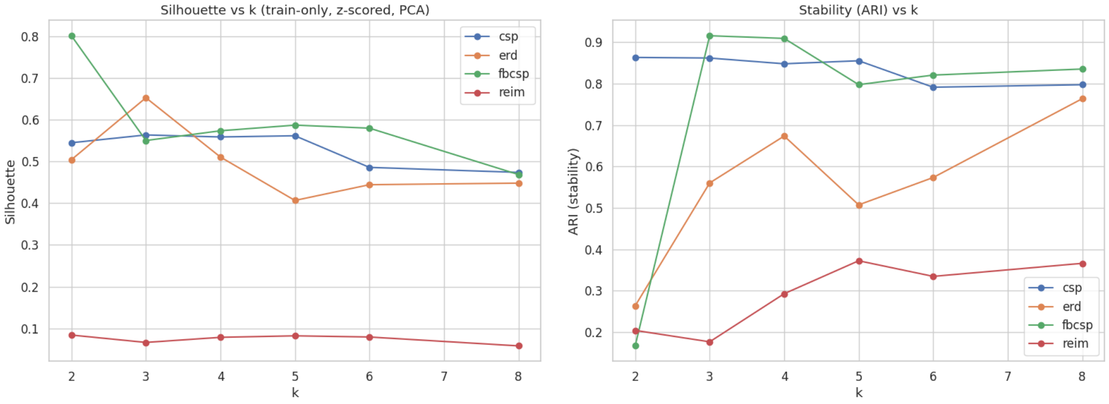
Stability curves tell the same story (Figure 4.1); CSP maintains consistently high assignment stability across all k, indicating robust group structure, while ERD/ERS fluctuates strongly and tends to break into highly imbalanced clusters whenever strong geometric separation is achieved. The trends in Silhouette and ARI across k consistently support choosing $k = 3$ as a balanced trade-off between cluster separation and stability (Figure 4.1, Figure B.2).

Taken together, these results identify CSP at $k = 3$ as the most suitable basis for subject-level population clustering in the subsequent transfer learning pipeline. It provides clear separation without tiny clusters and stable assignments under resampling. ERD/ERS at $k = 3$ is retained as a secondary baseline because of its sharper boundaries, but its weaker stability and severe imbalance make it a less reliable foundation. Excluding FBCSP and the Riemannian representation avoids unreliable groupings and prevents data-starved clusters that would undermine downstream multi-task learning and cluster-based fine-tuning.

This choice matters directly for the remainder of the study. Stable, non-singleton clusters enable robust cluster-conditioned modelling, reduce variance in per-group parameter updates, and provide a principled grouping scheme for subject-specific adaptation. We therefore fix CSP at $k = 3$ for all downstream experiments so that later gains can be attributed unambiguously to transfer learning design choices rather than clustering instability.

Table 4.1: Clustering quality at each method's optimal k (selected by silhouette score). CSP achieves balanced clusters with high agreement under resampling. ERD/ERS provides sharper boundaries but unstable assignments and severe size imbalance. Riemannian features fail to produce meaningful separability. FBCSP yields tiny clusters.

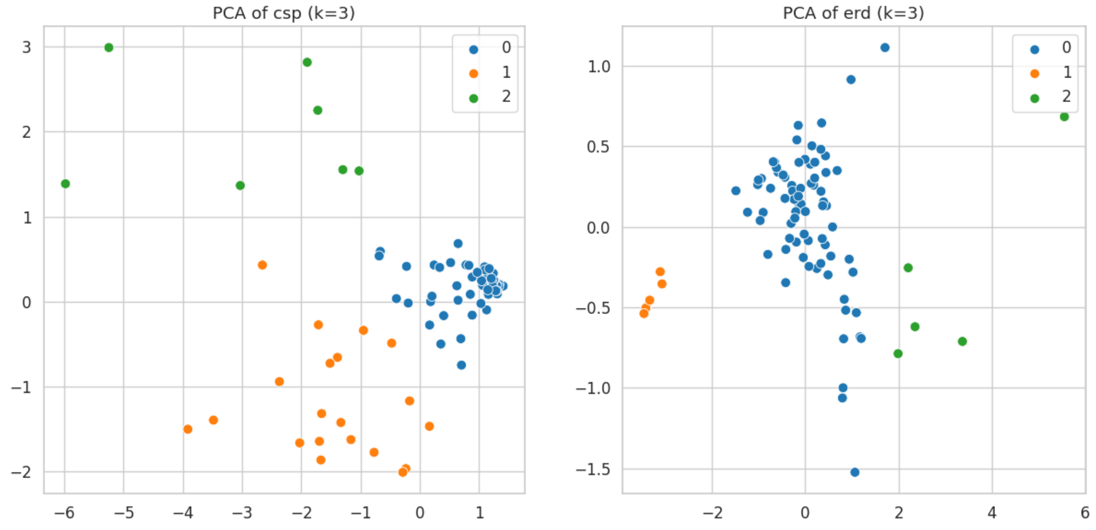
Method	k	Silhouette ↑	ARI ↑	Min size	Imbalance	Tiny cluster
CSP	**3**	**0.563**	**0.862**	**7**	**8.29**	**No**
ERD/ERS	3	0.652	0.560	5	14.80	No
FBCSP	5	0.587	0.797	1	18.67	Yes
Riemann	2	0.084	0.204	31	1.74	No



(a) Silhouette vs. k (geometry).

(b) Stability (ARI) vs. k (resampling).

Figure 4.1: Geometric separation and stability across candidate feature spaces and cluster counts. CSP maintains consistent stability. CSP (blue), ERD/ERS (orange), FBCSP (green), Riemannian (red).



(a) PCA of CSP (k=3).

(b) PCA of ERD/ERS (k=3).

Figure 4.2: SUBJECT-SPACE EMBEDDINGS (PCA, FIRST TWO COMPONENTS) FOR CSP AND ERD/ERS. 3 cluster IDs (0, 1, 2) as shown in each panel.

4.1.2 Transfer Learning Performance across Selected Feature Spaces

Building on Section 4.1.1, where CSP and ERD/ERS were identified as the leading subject-space representations, we evaluated the complete transfer learning pipeline under two deployment regimes: (i) a pooled baseline ($k = 1$), meaning a single model trained on data pooled across all subjects and applied to each subject without personalisation; and (ii) a clustered few-shot model ($k = 3$), in which subjects are first assigned to one of three clusters (subject-level population clustering), cluster-conditioned heads are trained per cluster on pooled data, and then adapted with a small number of subject-specific labelled trials while keeping the shared backbone fixed.

Cross-subject pooled model ($k = 1$)

The two representations are effectively equivalent. Mean accuracy was virtually identical ($\Delta = -0.002$, Wilcoxon $p = 0.38$), and all other metrics show similarly negligible differences (see Table 4.2). The paired-accuracy scatter plot (Figure 4.3, left panel) confirms this: most points lie on the diagonal, indicating that CSP and ERD/ERS perform indistinguishably when a single pooled model is applied across subjects (Figure B.4). Overall, these results support that when a single model is shared by all users, the choice between CSP and ERD/ERS has little consequence for decoding performance.

Clustered few-shot model ($k = 3$)

With the same number of calibration trials, CSP consistently outperforms ERD/ERS across subjects and metrics (Table 4.2). Mean accuracy improves by ~ 2 percentage points ($p < 10^{-10}$), and the effect generalises across Cohen's κ, precision, recall, and macro-F1. As shown in Figure 4.3, the paired accuracy scatter lies predominantly below the diagonal across the full performance range, indicating that CSP's advantage is not restricted to either weak or strong users but is broadly expressed.

These results reveal a clear interaction between representation and adaptation strategy. When a single pooled model is used ($k = 1$), representation choice has minimal impact, as the model averages across inter-subject variability. Once subject-level population clustering is introduced ($k = 3$), however, the quality of the representation becomes critical: CSP's more stable, non-singleton clusters (Section 4.1.1) translate into tangible downstream gains accordingly, while ERD/ERS is proposed only as an alternative in scenarios where data from all subjects are pooled to train a single shared model, CSP at $k = 3$ is standardized in this pipeline for all further analyses.

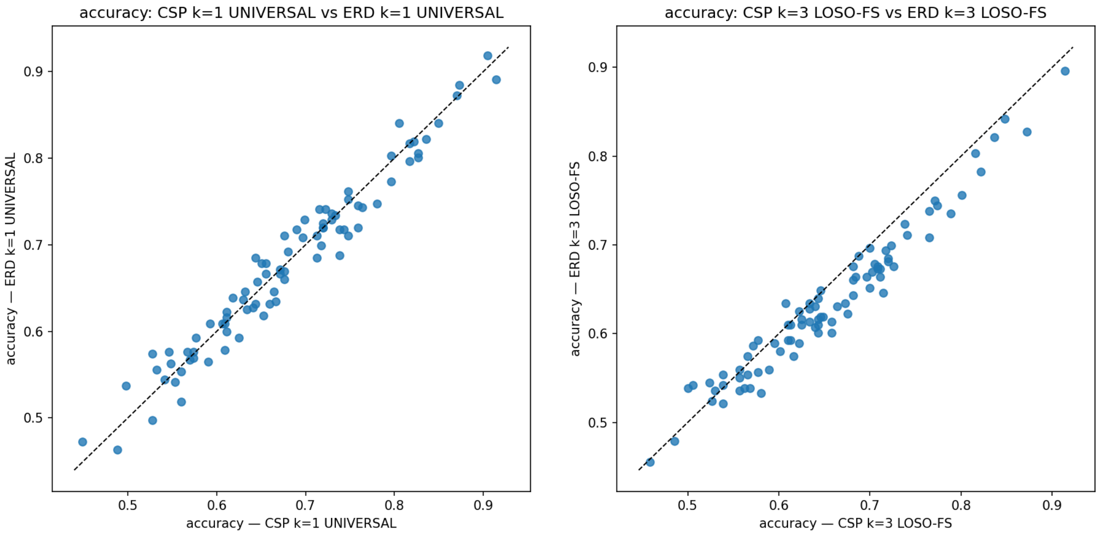
Table 4.2: Paired subject-wise comparison of CSP vs. ERD/ERS across two transfer learning regimes. CSP significantly outperforms erd/ers across accuracy, Cohen's κ, precision, recall, and macro-F1. $N = 85$.

(a) Single pooled model across subjects ($k = 1$).

Metric	CSP	ERD/ERS	Δ	p	r
Accuracy	0.679	0.677	−0.0020	0.380	0.114
κ	0.358	0.353	−0.0044	0.340	0.120
Precision	0.671	0.671	−0.0002	0.967	0.005
Recall	0.680	0.678	−0.0023	0.311	0.128
F1-score	0.642	0.638	−0.0042	0.122	0.193

(b) Few-shot model with cluster-conditioned heads ($k = 3$).

Metric	CSP	ERD/ERS	Δ	p	r
Accuracy	0.655	0.635	−0.0198	1.74×10^{-10}	0.811
κ	0.310	0.271	−0.0394	2.82×10^{-10}	0.792
Precision	0.646	0.638	−0.0079	0.0450	0.250
Recall	0.656	0.636	−0.0201	1.72×10^{-10}	0.802
F1-score	0.607	0.574	−0.0330	6.78×10^{-13}	0.897



(a) Pooled model across all subjects (k = 1).

(b) Few-shot model with cluster-conditioned heads (k = 3).

Figure 4.3: SUBJECT-WISE ACCURACY COMPARISON BETWEEN CSP AND ERD/ERS ACROSS TWO TRANSFER LEARNING REGIMES.. In the pooled setting, CSP and ERD/ERS perform equivalently. In the clustered few-shot setting, most points lie below the diagonal, indicating significantly higher accuracy for CSP.

4.2 Transform-Based Augmentation

This section investigates whether lightweight, label-preserving transformations improve motor imagery decoding under two deployment regimes: (i) a pooled baseline (k=1) and (ii) a clustered few-shot model (k=3). Augmentations are aimed to enrich the training distribution with plausible temporal and spectral variability so that the model learns useful invariances and becomes less sensitive to non-task-related variability across trials and subjects. Three transformations, Gaussian noise (σ=0.02), nonlinear time warping (±15%, ≤ 50% window), and frequency shift (±1 Hz), are applied to the *raw signals* during the *multi-task learning stage* to perturb the shared backbone. A fourth augmentation, mixup (α=0.2), is evaluated separately at the *transfer learning stage* as a label-aware batch interpolation. Each augmentation is applied in isolation. Features for clustering are always extracted from unaugmented data to avoid biasing group assignments.

Cross-subject pooled model (k = 1)

Here a single cross-subject classifier is trained on data pooled from all available subjects and evaluated per subject, without any per-subject calibration. Across all four augmentations, a consistent pattern emerged; none produced a systematic improvement. As shown in Figure 4.4a, per-subject differences between the augmented and baseline models are clustered near zero, with overlapping accuracy distributions and similar medians. While a few subjects benefited from certain transformations, a larger subset showed small decrements, resulting in an overall effect that was neutral to slightly negative (Table B.2). For a pooled classifier trained on heterogeneous subject data, the tested augmentations added no consistent value.

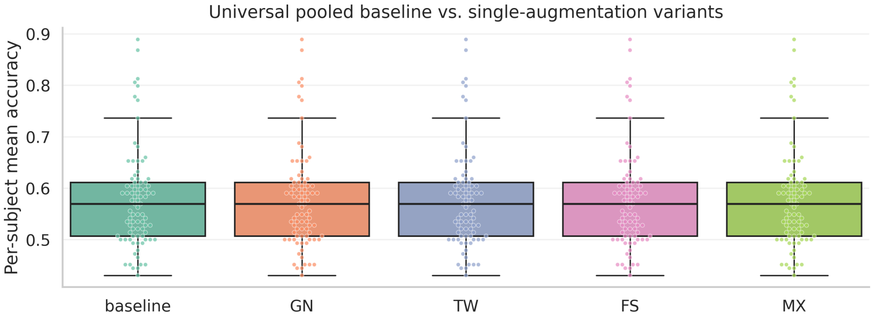
Clustered few-shot model ($k = 3$)

In this evaluation, one subject was held out; training was restricted to the subject's similarity cluster (as defined in Section 4.1.1), and three labeled trials per class were provided for calibration. Since temporal warping appeared most promising under pooled evaluation, it was tested directly against the non-augmented baseline in this regime. However, as illustrated in Figure 4.4b, performance consistently deteriorated: most subjects experienced reduced accuracy, and statistical testing confirmed a significant drop across both accuracy and F1-score scores (Table 4.3, Table B.3).

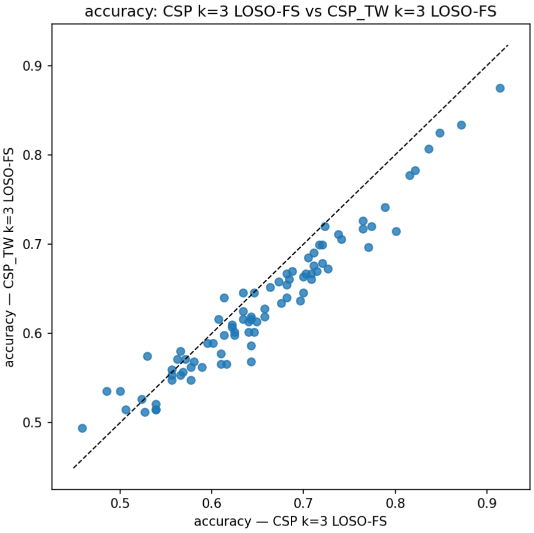
Taken together, these experiments show that none of the tested augmentations improved decoding performance. When training a pooled model on pooled multi-subject data, adding variability neither consistently helped nor hurt. In the clustered few-shot regime, even the augmentation that initially seemed most promising, temporal warping, reliably degraded performance. These findings suggest that in low-density, cue-locked MI paradigms, naive augmentation strategies are ineffective and may harm transferability, likely due to distortions that violate physiological plausibility. Consequently, augmentation is disabled by default for all subsequent experiments.

Table 4.3: Effect of time warping on CSP-based decoding under clustered few-shot adaptation regime. Temporal warping significantly reduces accuracy and macro-F_1 scores, despite appearing promising under pooled evaluation. Negative Δ values indicate performance drops. Holm-adjusted p-values confirm statistical significance; r denotes rank-biserial effect size.

Metric	Baseline	+ Time warp	Δ	p_{Holm}	r
Accuracy	0.6551	0.6321	−0.0230	5.10×10^{-10}	0.799
F1-score	0.6068	0.5697	−0.0372	3.08×10^{-12}	0.899



(a) Effect of four augmentations on a single pooled model ($k = 1$). Per-subject accuracies for baseline versus Gaussian noise (GN), time warping (TW), frequency shift (FS), and mixup (MX). Nearly identical means show that one of the augmentations provides systematic benefits when training a pooled classifier.



(b) Impact of time warping in clustered few-shot adaptation ($k = 3$). Paired accuracies for baseline versus time-warped training. Most points lie below the identity line, indicating reduced accuracy with time warping.

Figure 4.4: Affect of transform-based augmentations on motor imagery decoding.

4.3 Cross-Subject Transfer on the Source Dataset

In this section, *cross-subject transfer* on a publicly available MI EEG dataset is examined, relative to the first objective of this project. Specifically, the analysis evaluates whether explicitly modelling inter-subject heterogeneity through subject-level clustering improves personalization under realistic calibration constraints.

EEG responses in MI BCI vary substantially across users, creating challenges for generalization. This objective therefore compares a pooled model against a clustered approach that groups subjects based on similarity, assessing whether cluster-conditioned heads provide a better starting point for rapid adaptation.

All analyses use a leave-one-subject-out (LOSO) protocol to emulate deployment to unseen users. To isolate clustering effects, all other components are held fixed: clusters formed on CSP features (Section 4.1), identical preprocessing, and no data augmentation. Throughout this section, Δ denotes the paired difference *clustered* − *pooled*), so positive values indicate a benefit of clustering.

4.3.1 Few-Shot Personalization with Clustered and Pooled Models

Because practical onboarding often affords only a few labeled calibration trials per class, this analysis evaluates whether modelling cross-subject structure via clustering improves personalization compared to a pooled model when only limited labeled data are available from the target subject.

Two CSP-based decoding pipelines were compared under a LOSO evaluation protocol using a fixed set of four labeled calibration trials per class from the held-out subject. In the clustered regime, the remaining training subjects were partitioned into $k = 3$ data-driven groups (fit on training data only), and one head was trained per cluster using only its members. In the pooled regime ($k = 1$), a single head was trained on all training subjects combined. At test time, both models were calibrated on the same support set and evaluated on the same disjoint test split, ensuring a perfectly paired comparison.

Across subjects, clustering provides a clear and statistically robust benefit. The clustered few-shot model achieves higher mean accuracy (0.655) than the pooled model (0.616), yielding a mean paired improvement of $\Delta = +0.0389$ ($p \approx 1.1\times10^{-12}$; Table 4.4). Secondary metrics follow the same pattern, with significant gains in Cohen's κ, precision, recall, and F_1 (Table B.4).

As shown in Figure 4.5, most subjects achieve higher accuracy under clustering, with paired points lying predominantly above the identity line. The ordered distribution of subject-wise differences Δ is strongly right-skewed (Figure 4.6), reflecting widespread gains and only small losses. Notably, subjects who perform best under clustering tend to exhibit the largest drops under pooling, suggesting that clustering especially benefits subpopulations whose neural patterns are not well represented by a single decision boundary (Figure B.5).

In practical terms, when only a few labeled trials are available for personalization, cluster-conditioned training substantially improves decoding performance over pooled training for the vast majority of users. This indicates that clustering might particularly benefit subpopulations whose neural response patterns deviate from the global average. In other words, clustering recovers structure that would otherwise be obscured by pooled models, an effect that contributes disproportionately to the overall performance gains.

Table 4.4: FEW-SHOT PERSONALIZATION: CLUSTERED VS. POOLED TRAINING. Subject-level mean accuracies for clustered (k = 3) and pooled (k = 1) training are reported, along with the mean paired difference (Δ = clustered − pooled), 95% bootstrap confidence interval, and Wilcoxon signed-rank statistics. Positive Δ values favor clustering.

	$\bar{A}_{\text{clustered}}$	$\bar{A}_{\text{pooled}}$	Δ	95% CI	W	p
Accuracy	0.6551	0.6162	**+0.0389**	[0.0313, 0.0470]	175.0	1.08×10^{-12}

4.3.2 Effect of Clustering on Zero-Shot Transfer

This subsection examines whether clustering subjects during training improves *zero-shot* cross-subject generalization, where performance is evaluated on a previously unseen subject without using any subject-specific calibration data. Models are trained on the public MI-EEG dataset using a CSP-based decoding pipeline with k=3 cluster-conditioned heads derived from k-means grouping. At test time, the held-out subject is assigned to one of the cluster heads using an unsupervised rule based on similarity to training subjects. The clustered model is compared against a single *pooled* model trained on all subjects combined.

Clustering provides a small but statistically reliable advantage over the pooled baseline. Mean accuracy increases from 0.633 (pooled) to 0.640 (clustered), corresponding to an average gain of $\Delta \approx +0.64$ pp (Wilcoxon p = 0.0091; rank-biserial r = 0.33). As shown in Figure 4.7, the paired accuracies lie slightly above the identity line for

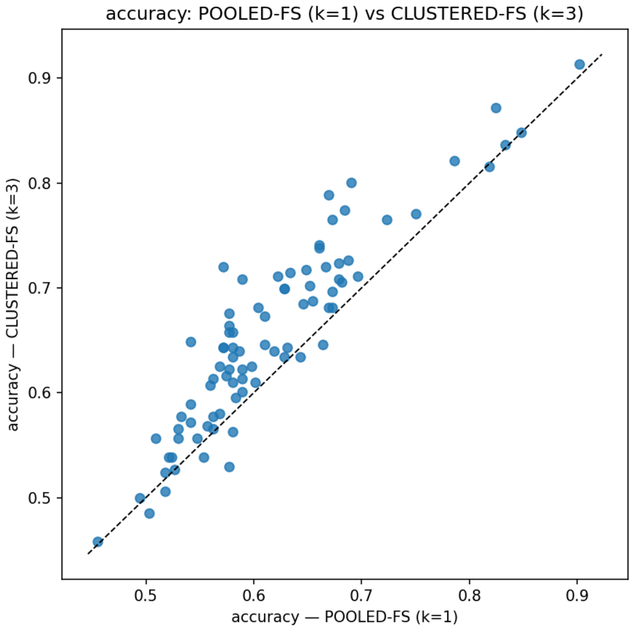


Figure 4.5: Few-shot personalization: clustered vs. pooled training. Each point represents a subject's accuracy ($N = 85$). The dashed line marks parity. Points above the line indicate better performance under clustering.

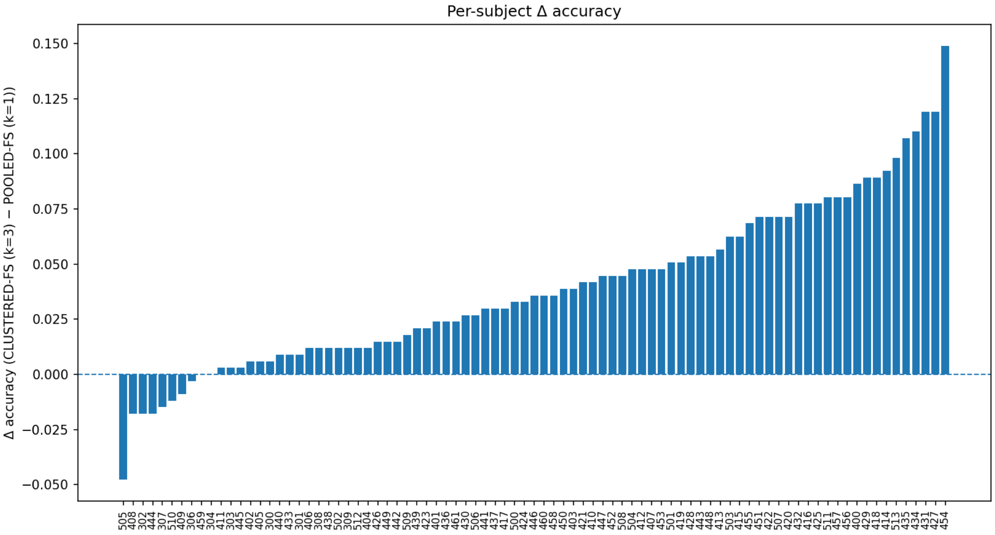


Figure 4.6: Subject-wise benefit from clustering in few-shot personalization. Bars show the ordered per-subject difference Δ = clustered − pooled. The strong right skew highlights broad, consistent gains.

most subjects, indicating modest benefits overall. Subject-level effects are heterogeneous: 55 of 85 subjects improve, 29 decline, and one ties, with Δ ranging from approximately −6.5 pp to +5.2 pp (Figure B.6).

Table 4.5: Zero-shot cross-subject transfer performance. Mean accuracy, paired difference (Δ = clustered − pooled), 95% bootstrap confidence intervals, and Wilcoxon p-value. Clustering provides a small but statistically significant average improvement.

Metric	Pooled	Clustered	Mean Δ	95% CI(Δ)	p	I/D/T
Accuracy	0.6333	0.6397	+**0.00637**	[0.107, 0.01161]	0.0091	55/29/1

Secondary metrics show a consistent, though not uniform, pattern of improvement. Precision increases most strongly (+2.69 pp; $p = 3.8 \times 10^{-7}$), while Cohen's κ, recall, and F1-score-score exhibit smaller but significant gains for many subjects (Table 4.5, Table B.5). Figure 4.7 illustrates these improvements at the subject level: although the effect size is modest, clustering tends to tighten decision boundaries, reducing misclassifications even when overall accuracy changes are small (Figure B.6).

These findings indicate that clustering during training can modestly improve zero-shot transfer to unseen subjects. Still, the benefits are heterogeneous and considerably smaller than those observed in the few-shot setting (Section 4.3.1). In practical terms, clustered training is preferable to pooling when no calibration data are available, but the most substantial personalization benefits emerge when even a small labeled support set is provided.

4.3.3 Cluster-Conditioned Support in Few-Shot Adaptation

Building on earlier results, where clustering yielded modest zero-shot gains and larger improvements once a few labeled trials were available, this analysis isolates whether the few-shot advantage genuinely arises from *cluster conditioning*; that is, drawing the support set from the test subject's assigned cluster, rather than from other procedural factors such as number of heads, representation choice, adaptation schedule, or class balance.

Both settings use the same pipeline with $k_{\text{shot}} = 4$ calibration trials, differing only in how the support examples are selected: the pooled regime draws support from all training subjects, whereas the clustered regime restricts support to the held-out subject's assigned cluster ($k = 3$).

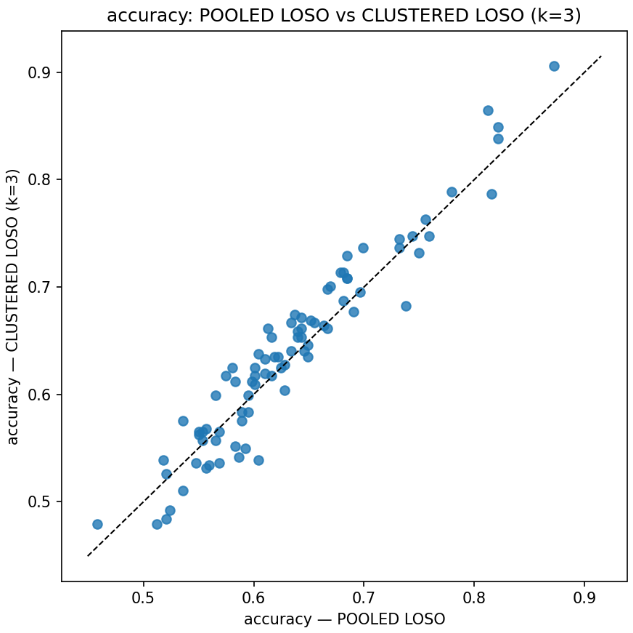


Figure 4.7: Zero-shot transfer performance per subject. Paired accuracies for pooled (x-axis) vs. clustered (y-axis) training. Points above the dashed identity line indicate subjects show a modest but consistent benefit from clustering without calibration.

Crucially, this isolates *why* clustering works in the few-shot regime. Because calibration examples are drawn from subjects with similar signal statistics and class balance, the adapted classifier starts from a more appropriate initialization. This improves both head selection and the calibration of CSP filters and thresholds. In contrast, when calibration examples are mixed from unrelated subjects (pooled support), adaptation often overfits to mismatched patterns, leading to unstable or suboptimal decision boundaries.

Importantly, the benefit of cluster-conditioned support is broad but heterogeneous; 74/85 subjects improve when support is cluster-conditioned, but the size of the gain varies across individuals (Table 4.6, Figure 4.8; Table B.6, Figure B.8). Decile analyses based on baseline accuracy under the pooled model show consistent improvements across most performance groups, tapering only at the very top. A small, non-significant correlation between pooled baseline accuracy and benefit (Spearman $\rho = 0.12$, $p = 0.26$) suggests that subjects with both low and high baseline performance under pooled training benefit on average (Figure B.9).

Together, these results demonstrate that the substantial gains observed in clustered few-shot adaptation arise specifically from using cluster-conditioned support. Restricting the support set to the assigned cluster stabilizes calibration, whereas pooled support can degrade performance.

Table 4.6: Few-shot adaptation: accuracy with cluster-conditioned vs. pooled support. Positive Δ values favor clustering. (improved/declined/tied: 74/8/3)

	Pooled	Clustered	Δ (pp)	p (Holm)
Accuracy	0.624	0.655	+3.07	1.7×10^{-13}

95% CI for Δ: [2.49, 3.67] pp; rank-biserial $r = 0.95$.

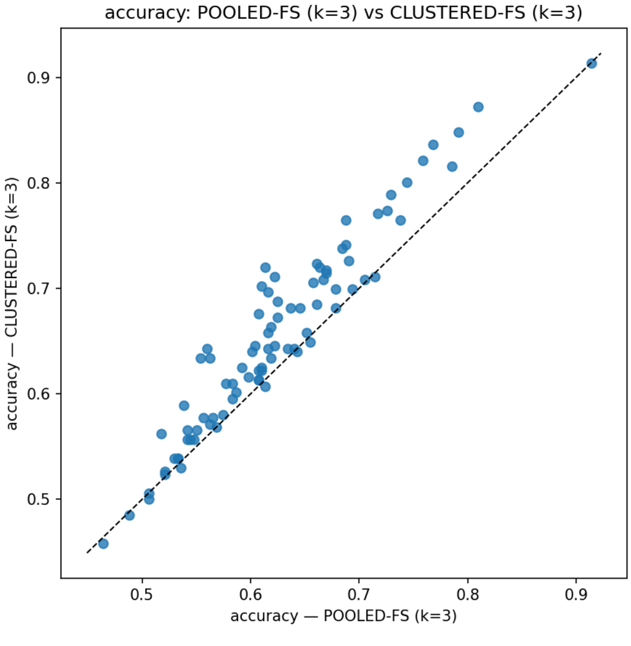


Figure 4.8: Paired subject accuracies for few-shot adaptation: pooled vs. cluster-conditioned support. Each point represents one subject; the dashed line marks $y = x$. Most points lie above the line, indicating improved accuracy when support examples are drawn from the assigned cluster.

4.3.4 Effect of Calibration Trials

To assess whether performance improves with greater supervision, this analysis investigates whether increasing the number of labeled calibration trials further improve personalization. This subsection isolates this aspect by comparing adaptation outcomes when the per-class support set is doubled, from 4 to 8 trials, while all other components of the pipeline remain unchanged.

Surprisingly, the larger support set consistently degrades generalization. Subject-level deltas are predominantly negative, indicating that most subjects perform worse at 8 shots than at 4. The paired scatter (Figure 4.9) confirms this trend, with the vast majority of points falling below the identity line (Figure B.10).

A likely explanation is that larger calibration sets introduce harmful variance. Within-subject fluctuations, such as temporary attentional shifts, label noise, or low-frequency drifts, can push the adapted decision boundary away from the stable test-time distribution. Since the adaptation rule treats all support trials equally, adding more data does not necessarily yield more useful information; rather, it can dilute the signal and compromise calibration. This effect appears strongest for subjects with already high accuracy under 4-shot adaptation, as seen in the Figure B.11; the largest losses are concentrated among the best-performing subjects, suggesting a regression effect where useful initial generalization is disrupted by overfitting to spurious patterns (Figure B.11). These results suggest that, under the current calibration strategy, increasing the number of support trials (beyond 4 per class) introduces enough unwanted variation to degrade generalization. Potentially, more accurate and label-efficient performance is achieved with smaller, well-targeted support sets.

Table 4.7: FEW-SHOT PERFORMANCE WITH 4 VS. 8 SUPPORT TRIALS PER CLASS. Δ is accuracy at k_{shot}=8 minus k_{shot}=4. Performance declines with more support data (improved/declined/tied: 11/73/1).

	k=4	k=8	Δ (pp)	p (Holm)
Accuracy	0.655	0.633	−2.24	$< 10^{-10}$

95% CI for Δ: [−2.72, −1.78] pp; rank-biserial $r = -0.91$.

4.3.5 Feature-Centric Analysis of Clustering Benefit

This section investigates whether specific subject-level features predict who benefits most from clustered few-shot adaptation. To isolate this effect, subjects were strati-

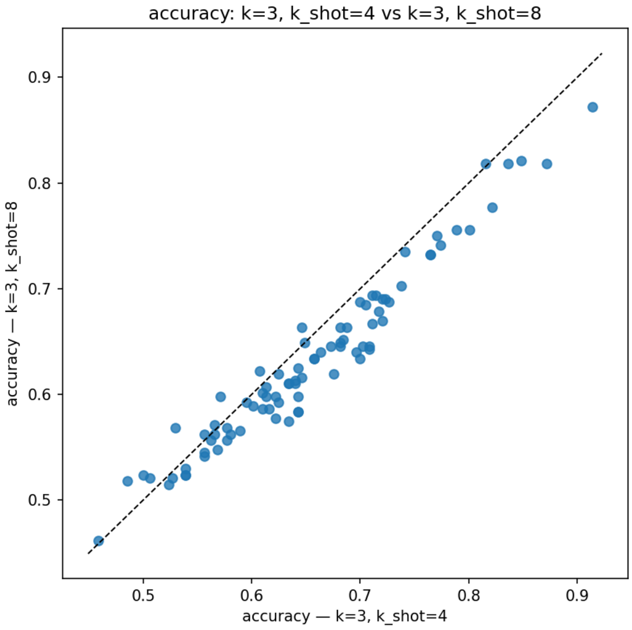


Figure 4.9: Few-shot accuracy per subject comparing support sets of 4 vs. 8 trials per class. Each point is a subject; most lie below the $y=x$ line, indicating lower accuracy when using larger support sets.

fied into deciles based on multiple neurophysiological and feature-derived indicators, and the paired performance gain (Δ) between clustered and pooled models was computed. Positive Δ indicates improved accuracy under clustering. While no single feature achieves a strong monotonic correlation with clustering benefit, decile-based trends reveal informative heuristics for prioritizing cluster-based personalization.

Baseline accuracy (pooled)

Clustering benefit varies substantially across the accuracy spectrum. Subjects in the lowest accuracy decile show near-zero or slightly negative Δ, while those in deciles 2 through 8 exhibit progressively larger benefits, peaking at $\Delta \approx 0.056$ in decile 8. A drop is observed at the top decile, likely due to ceiling effects. These findings suggest that subjects with *mid-to-high accuracy* under pooled models stand to gain most from cluster-conditioned adaptation. Presumably, their MI patterns might already be decodable to some extent, and clustering could further tune by removing cross-subject variance. This trend is illustrated in Figure 4.10, with corresponding statistics reported in Table 4.8.

Beta-band power

Across most deciles, clustering benefit remains mildly positive, with Δ values typically in the 0.03–0.05 range. However, the overall Spearman correlation is small and

Table 4.8: Clustering benefit (Δ) by decile of pooled accuracy in the few-shot setting. Mean subject-wise benefit with 95% bootstrap confidence intervals. Positive values indicate improved accuracy under clustering. The highest benefit occurs in decile 8.

Decile	**Δ [95% CI]**	*n*
0	0.007 [–0.003, 0.020]	9
1	0.036 [0.015, 0.061]	8
2	0.049 [0.026, 0.077]	10
3	0.046 [0.019, 0.070]	11
4	0.042 [0.017, 0.076]	6
5	0.046 [0.026, 0.067]	7
6	0.042 [0.019, 0.065]	8
7	0.048 [0.026, 0.071]	11
8	**0.056 [0.032, 0.083]**	**6**
9	0.019 [0.008, 0.031]	9

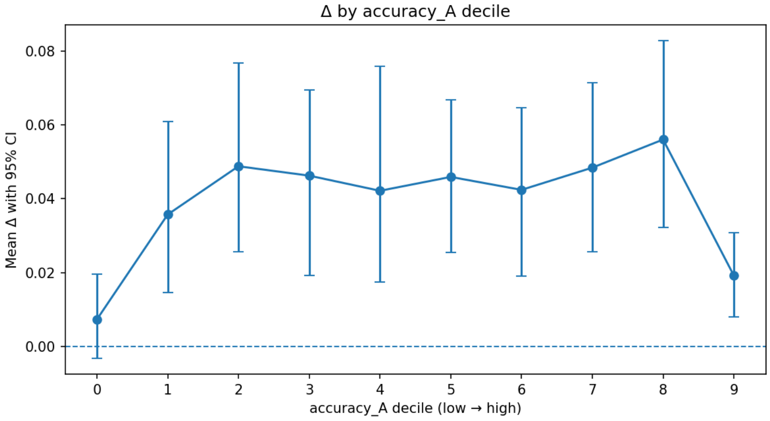


Figure 4.10: Clustering benefit by decile of pooled accuracy. Subjects are stratified into deciles based on their accuracy under the pooled model. Bars show the mean improvement Δ in each decile, with error bars indicating 95% confidence intervals. Clustering provides the greatest benefit for subjects with mid-to-high baseline accuracy (deciles 2–8), while those in the lowest or highest deciles benefit less.

non-significant ($\rho \approx -0.06$, $p \approx 0.61$), and variability across subjects is substantial. While beta power may relate to motor cortex activation, these results suggest that it provides only a weak and inconsistent signal of clustering responsiveness. At best, beta power may serve as a soft heuristic when combined with other indicators (Appendix B.8).

Mu-band power

Mu power shows a similar pattern: modest positive benefit in most deciles, with mild peaks in the mid-range ($\Delta \approx 0.05$) (Figure 4.11). Yet, the correlation between mu power and benefit is weak and statistically insignificant ($\rho \approx -0.10$, $p \approx 0.34$). These results imply that although mu power is neurophysiologically relevant to motor imagery, it does not reliably predict who will benefit from clustering. It may still offer some prioritization value when used in conjunction with other features.

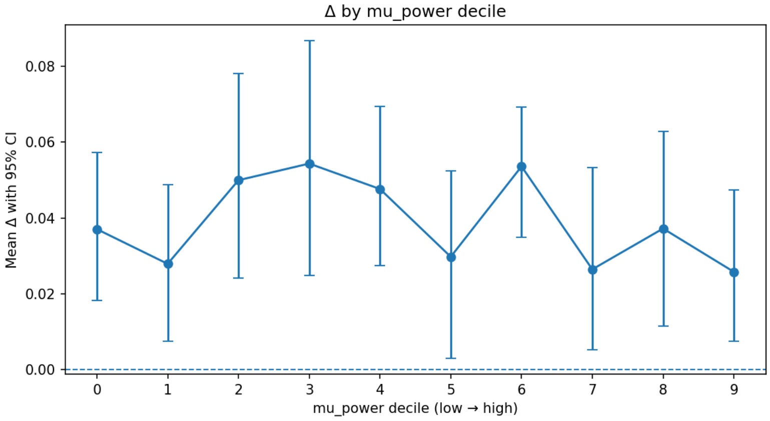


Figure 4.11: CLUSTERING BENEFIT BY MU-BAND POWER DECILES. Subjects were stratified into deciles based on their absolute mu-band power, and the mean paired improvement Δ in few-shot accuracy was computed for each decile.

CSP component mean (component 1)

CSP components 1 and 2 refer to the first two spatial filters extracted via CSP; component 1 corresponds to the projection maximizing variance for one class, and component 2 for the opposite class. Each captures a symmetric but class-specific motor pattern across trials. The mean of the first CSP component exhibits a non-monotonic but structured trend. Benefit peaks at decile 2 ($\Delta \approx 0.06$) and again around decile 7, with a dip in the middle. This suggests that very low or very high CSP mean projections may be less useful for clustering, whereas moderate values reflect more discriminable spatial filters. Although the correlation is weak ($\rho \approx -0.09$, $p \approx 0.41$), the decile-based trends are sufficiently stable to suggest heuristic value (Figure B.12).

CSP component variability.
In the projections of the first two CSP components, trial-to-trial variability is quantified, which represents symmetric discriminative filters for the two MI classes. Variability in the CSP component 2 shows a clear positive association with clustering benefit, increasing steadily across deciles and peaking at $\Delta \approx 0.06$. In contrast, variability in the CSP component 1 shows no consistent trend and has a negative regression weight, indicating limited predictive value. This suggests that subjects with more dynamic or expressive CSP components may benefit more from clustering (Figure 4.12, Figure B.12). In contrast, *CSP component 1* shows mixed results: while some mid-to-high deciles yield positive Δ, its regression coefficient is negative and correlation weak ($\rho \approx -0.08$, $p \approx 0.46$), indicating limited standalone predictive value (Appendix Table B.12).

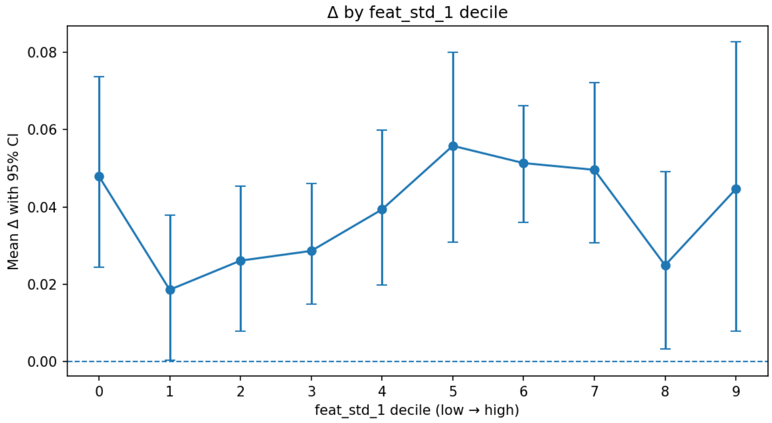


Figure 4.12: CLUSTERING BENEFIT BY DECILE OF CSP FEATURE VARIABILITY. Subjects are grouped by decile of the standard deviation of their second CSP component across trials. Clustering benefit Δ increases steadily with higher variability, peaking above 0.06 in the top deciles. This suggests that subjects with more expressive or dynamic CSP patterns benefit most from cluster-conditioned few-shot adaptation.

Taken together, these findings support the use of subject-level features as indicators (not deterministic rules) for prioritizing the use of clustered few-shot adaptation. While no feature is strongly predictive in isolation, combinations such as moderate pooled accuracy, high *CSP component 2)*, and elevated mu or beta power form useful heuristics for identifying responsive users, this is consistent with the policy curve analysis (Figure 4.14), which shows that even weak predictors can improve model selection when used for subject ranking.

Table 4.9 summarizes the key subject-level predictors of clustering benefit. While individual correlations are weak and mostly non-significant, certain decile trends and regression coefficients reveal useful heuristics, Table B.8. In particular, `feat_std_1` emerges as the strongest candidate predictor, whereas measures such as ERD or spectral entropy contribute little.

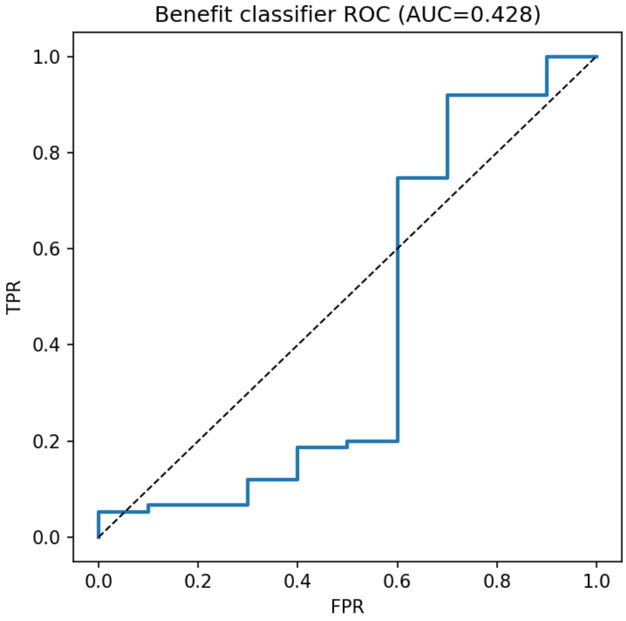


Figure 4.13: ROC CURVE FOR CLASSIFYING CLUSTERING BENEFIT FROM SUBJECT-LEVEL FEATURES. The area under the curve (AUC) falls below 0.5, indicating that no linear combination of the analyzed features is sufficient to separate "benefit" vs. "no benefit" users with high accuracy. This supports the conclusion that benefit prediction should rely on ranking (e.g., policy-based targeting) rather than classification.

4.3.6 Concluding Remarks on Cross-Subject Transfer

The cross-subject transfer experiments explored three questions: (i) whether clustered models consistently outperform pooled ones under zero- and few-shot calibration, (ii) whether using support trials from a subject's assigned cluster provides a better inductive bias than pooled calibration across all subjects, and (iii) whether subject-level features can explain or predict those gains.

Clustering resulted in broad and reliable advantages in the few-shot setting. At fixed calibration trials, accuracy improvements were observed across most baseline-performance groups and feature deciles, with positive trends visible in both scalar and distributional comparisons. Zero-shot gains were smaller and more variable; present for some subjects, particularly those with stronger baselines, but not reliably predictable. Crucially, no single scalar feature (including baseline accuracy) showed a monotonic or generalizable association with benefit. These findings support three conclusions: (1) clustering improves few-shot personalization across diverse users,

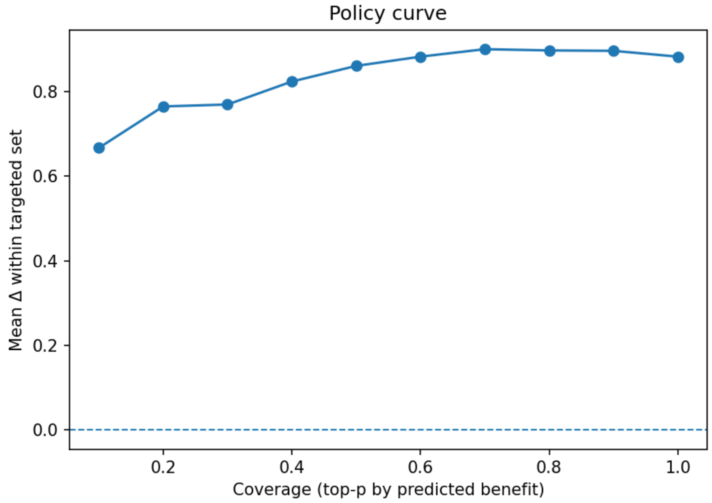


Figure 4.14: Policy curve: mean clustering benefit Δ as a function of subject coverage. Subjects are ranked by predicted benefit from a weak feature-based model. The y-axis shows the average actual Δ achieved within the top x% of subjects (coverage). Even with weak predictors, targeted selection yields substantial gains: the top 20% of users achieve nearly double the population-average benefit. This supports the practical use of soft prioritization policies.

Table 4.9: Summary of predictive indicators for clustering benefit (Δ). Correlation and regression statistics reflect subject-level relationships. The "Decile Trend" column summarizes observed patterns in clustering benefit across deciles.

Feature	ρ	p**-val**	**Coef.**	**Decile Trend**
Baseline accuracy	0.123	0.261	–0.182	U-shaped: mid-high benefit
CSP variability (2nd comp.)	0.112	0.306	+0.396	Rising monotonic trend
CSP mean (1st comp.)	–0.091	0.409	+0.057	Non-monotonic (peaks at 2, 7)
Mu-band power	–0.105	0.339	–0.238	Mild mid-decile peaks
Beta-band power	–0.056	0.612	–0.259	Weakly positive
Mu-band ERD	+0.068	0.537	–0.406	Flat
Beta-band ERD	–0.008	0.939	–0.087	Flat
Spectral entropy	–0.042	0.701	–0.020	None
CSP variability (1st comp.)	–0.082	0.457	–0.596	Inconsistent
CSP mean (2nd comp.)	+0.026	0.811	–0.416	No trend

offering a reliable advantage even with minimal labeled data, (2) clustered few-shot decoding is a robust and label-efficient default, and (3) zero-shot gains are modest, uneven, and unlikely to be well-targeted using simple heuristics.

4.4 Personalized Decoding on the Target Dataset

This section evaluates *in-session transfer* (i.e., adapting a pretrained model to a new subject using data from the same recording session) on *target* subjects recorded with the Unicorn headset using the pretrained TL model fit on the public cohort. While previous sections focused on offline generalization across subjects and group-level effects, this section shifts to deployment in realistic, session-based conditions, corresponding to the second objective of this project.

In-session transfer is evaluated on two genuinely unseen subjects (Subject 1 and Subject 2) to assess (i) whether subject-specific calibration improves performance beyond zero-shot transfer, and (ii) whether the *initialization source*, a pooled head versus a cluster-matched head, modulates the outcome of few-shot adaptation.

The pretrained model comprises a shared feature extractor and a set of cluster-specific heads obtained via k-means clustering (k=3) on CSP-based subject embeddings. For each new subject, the stored pipeline is reused to compute the CSP representation (via the pretrained projection and scaler), which is then assigned to the nearest cluster centroid—without using any labels.

Two zero-calibration baselines are evaluated: a *pooled baseline* given by the head trained on all public subjects (head-0), and a *cluster baseline* that routes the subject to the head associated with their assigned cluster. Few-shot adaptation is then performed with $k_{\text{shot}} \in \{1, 2, 3, 4\}$ labeled trials, starting either from the pooled head (pooled initialization) or the matched cluster head (clustered initialization). Evaluation is performed on held-out trials from the same session, with accuracy as the primary outcome and secondary metrics (kappa, precision, recall, F1-score) reported for completeness.

4.4.1 Subject 1 Results

A pretrained backbone paired with a set of cluster-specific heads was deployed on an in-session MI recording obtained from Subject 1. The subject was assigned to Cluster 1 using the pretrained clustering model, based on their projected position in CSP transform space; the Euclidean distance to the nearest centroid was 0.552.

To establish a baseline for zero-calibration performance, two routing strategies were compared: the *pooled head*, trained on all public subjects, and the *zero-shot cluster head*, obtained by assigning the subject to a cluster and directly applying the corresponding head without adaptation. Both routes resulted in identical accuracy (0.654), suggesting that cluster-based routing, without further tuning, had no immediate effect for this subject.

Subject-specific adaptation was then performed using $k_{\text{shot}} \in \{1, 2, 3, 4\}$ labeled trials per class, with two initialization strategies: (i) *pooled initialization* (copying the pooled head) and (ii) *cluster initialization* (copying the assigned cluster head, Cluster 1).

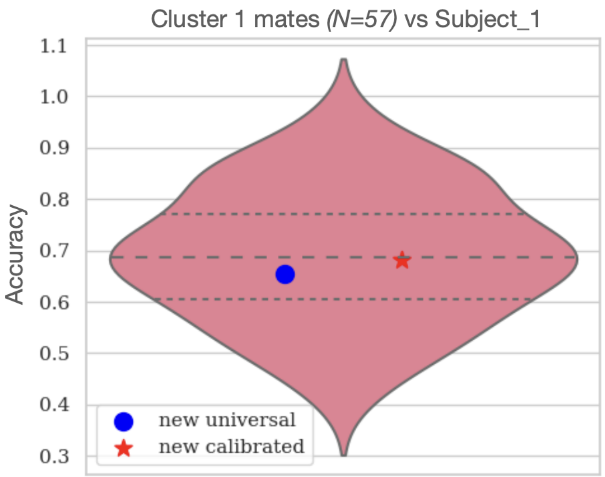


Figure 4.15: SUBJECT 1: POSITION WITHIN CLUSTER-LEVEL PERFORMANCE. Violin plot of offline accuracies for Cluster 1 subjects. Blue dot marks Subject 1's pooled zero-shot accuracy; red star shows their final calibrated performance (k_{shot}=4). Dashed lines indicate quartiles of the offline distribution.

With a single labeled trial (k_{shot}=1), cluster initialization achieved the highest observed accuracy (0.800), outperforming pooled initialization (0.720) and exceeding both zero-shot baselines. As the calibration trials increased, the two trajectories diverged: pooled initialization improved monotonically, while the cluster-initialized path dipped between one and three trials, only partially recovering with four shots. At $k_{\text{shot}} = 4$, pooled initialization reached 0.773 whereas cluster initialization plateaued at 0.682. Relative to their respective baselines, cluster initialization yielded a modest gain (+0.028), while pooled initialization produced a substantial improvement (+0.119). Secondary metrics (κ, precision, recall, F1-score) mirrored this pattern.

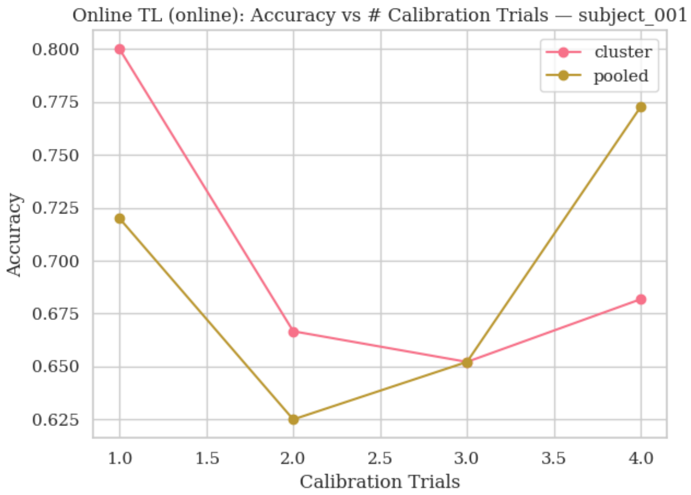
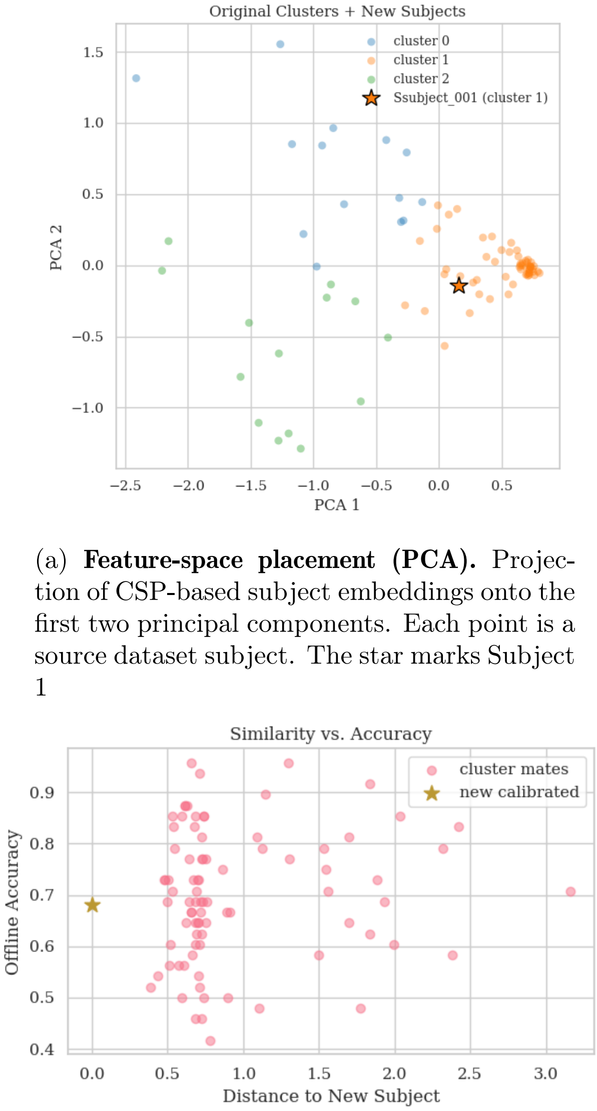


Figure 4.16: SUBJECT 1: CALIBRATION ACCURACY AS A FUNCTION OF k_{SHOT}. Accuracy vs. number of labeled trials per class (k_{shot}), for cluster-based and pooled initialization. Lines show mean accuracy over repeated stratified draws on disjoint test data.

The calibration curve (Figure 4.16) shows an early advantage for cluster initialization that diminishes and reverses by k=4. A violin plot of offline accuracies for Cluster 1 peers places Subject 1's pooled baseline slightly below the cluster median, with the final calibrated point moving toward the center of the distribution (Figure 4.15). A scatter of cluster-mate similarity versus offline accuracy (Figure 4.17b) shows no monotonic trend, indicating that geometric proximity alone does not predict performance for this subject.

Subject 1 illustrates a clear tradeoff between early gains and long-term stability. Clustered initialization offers a strong inductive prior, yielding the highest accuracy in the one-shot setting. However, this early advantage erodes with additional calibration data: while pooled initialization starts lower, it improves steadily and ultimately surpasses the cluster-based path, achieving the best performance at k_{shot}=4. Cluster-based head assignment without any subject-specific labels provides no benefit in the zero-shot case. Effective personalization emerges only through progressive adaptation for this subject.



(a) **Feature-space placement (PCA).** Projection of CSP-based subject embeddings onto the first two principal components. Each point is a source dataset subject. The star marks Subject 1

(b) **Within-cluster similarity vs. performance.** For each Cluster 1 peer, the x-axis shows its Euclidean distance to Subject 1 in the same PCA+scaler space; the y-axis shows that peer's *offline* accuracy. The star marks Subject 1's calibrated point (shown for reference at x=0).

Figure 4.17: Subject 1: feature-space placement and cluster diagnostics.

Table 4.10: SUBJECT 1: ACCURACY AND CALIBRATION GAIN UNDER TWO INITIALIZATIONS. Calibrated accuracies and baseline-relative improvements for cluster- and pooled-initialized models across increasing k_{shot} values. Δ_{cluster} and Δ_{pooled} quantify gains relative to the assigned zero-shot head. Accuracies are means over repeated stratified draws on disjoint test sets. Bold values indicate the better initialization at each k.

	Accuracy		Δ vs. baseline	
Setting	Cluster init	Pooled init	Δ_{cluster}	Δ_{pooled}

Assignment: cluster id = 1;
distance = 0.552				

Zero-shot baselines	**0.654**	**0.654**	–	–
k_{shot}=1	**0.800**	0.720	+0.146	+0.066
k_{shot}=2	0.665	0.625	+0.011	−0.029
k_{shot}=3	0.651	0.651	−0.003	−0.003
k_{shot}=4	0.682	**0.773**	+0.028	+0.119

4.4.2 Subject 2 Results

For the second target subject, the two zero-calibration baselines were identical (0.480). The subject was assigned to Cluster 2 with a transform-space distance of 0.956 to the nearest centroid, substantially larger than for Subject 1, indicating that the subject's representation lies farther from the known manifold. The PCA overlay places Subject 2 within the Cluster 2 region but not near its densest core, while the violin plot shows that both the pooled baseline and final calibrated results fall below the cluster median, consistent with a generally more difficult recording.

Few-shot calibration yielded modest but interpretable gains. At k_{shot}=1, cluster initialization reached 0.500 accuracy (+0.020 vs. the cluster baseline), while pooled initialization underperformed at 0.417 (–0.063 vs. the pooled baseline). At k_{shot}=2, both paths converged at 0.478, showing no net improvement. At k_{shot}=3, pooled initialization surpassed the cluster path with 0.545 (+0.065), while the cluster path remained flat at 0.500 (+0.020). At k_{shot}=4, cluster initialization improved slightly to 0.524 (+0.044), whereas pooled initialization regressed to 0.476 (–0.004). Secondary metrics (κ, F1-score) followed the same trajectory, confirming that the changes are not attributable to class imbalance. The calibration trajectories are visualized in Figure 4.19, showing the early advantage of cluster initialization at k_{shot}=1, followed by a crossover and peak performance under pooled initialization at k=3.

The diagnostics support these trends. The violin plot places Subject 2 in the lower tail of Cluster 2's offline distribution (Figure 4.18), while the PCA map shows the subject lying on the cluster periphery (Figure 4.20a). A similarity–accuracy scatter among Cluster 2 members reveals no monotonic association between transform-space proximity and offline performance (Figure 4.20b), reaffirming that geometric closeness does not predict adaptation quality.

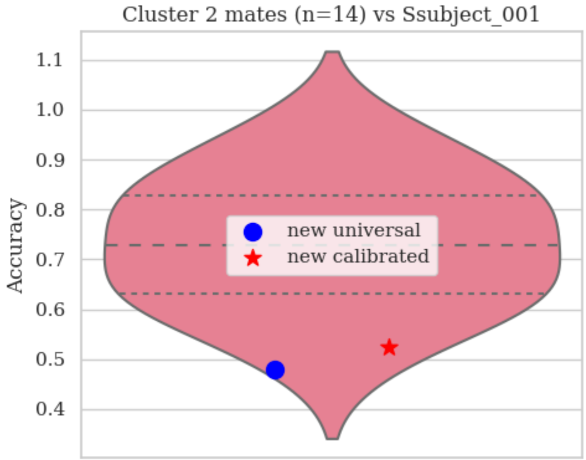


Figure 4.18: Subject 2: position within cluster-level performance. Violin plot of offline accuracies for Cluster 2 subjects. Blue dot marks Subject 2's pooled zero-shot accuracy; red star shows their best calibrated result (0.545 at k_{shot}=3, pooled initialization). Dashed lines indicate quartiles of the offline distribution.

Overall, Subject 2 reflects a challenging in-session case. Cluster-based routing provides no benefit over the pooled baseline. Cluster initialization offers a small gain only in the one-shot setting. Pooled initialization becomes superior at k_{shot}=3, but shows no stable upward trend. This pattern echoes the findings from Objective 1: clustering rarely improves performance in zero-shot settings, cluster initialization may offer value at a small number of calibration trials, and the largest gains typically stem from supervised calibration—not from routing alone. For difficult subjects like this one, non-monotonic responses to k are to be expected. A detailed breakdown of calibrated accuracies and baseline-relative improvements is provided in Table 4.11

4.4.3 Concluding Remarks on Personalized Decoding

With two genuinely unseen subjects evaluated under the same online protocol, this section synthesizes the effects of in-session calibration. Zero-shot routing to the nearest cluster head offered, at best, parity with the pooled head and did not yield measurable benefit for either subject—reinforcing that subject-specific gains arise from calibration rather than routing alone. Calibration was evaluated across $k_{\text{shot}} \in \{1, 2, 3, 4\}$ under two initialization strategies: a pooled head and the subject's assigned cluster head. Across both cases, the pattern echoes the results from Objective 1. In the one-shot setting, clustered initialization yielded better performance than pooled initialization, suggesting a stronger inductive prior encoded in

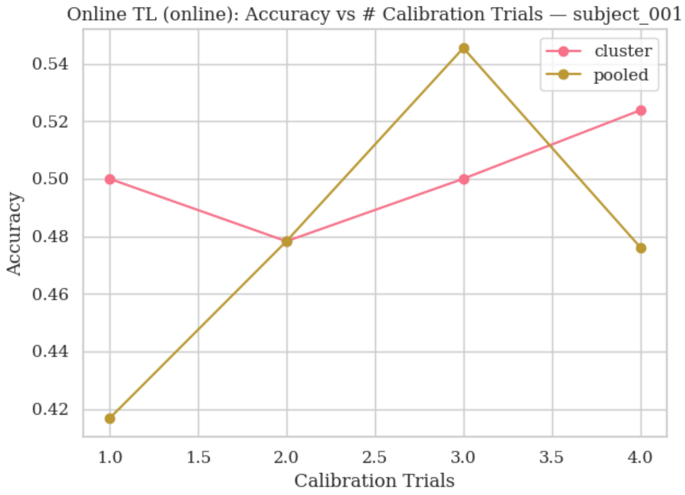


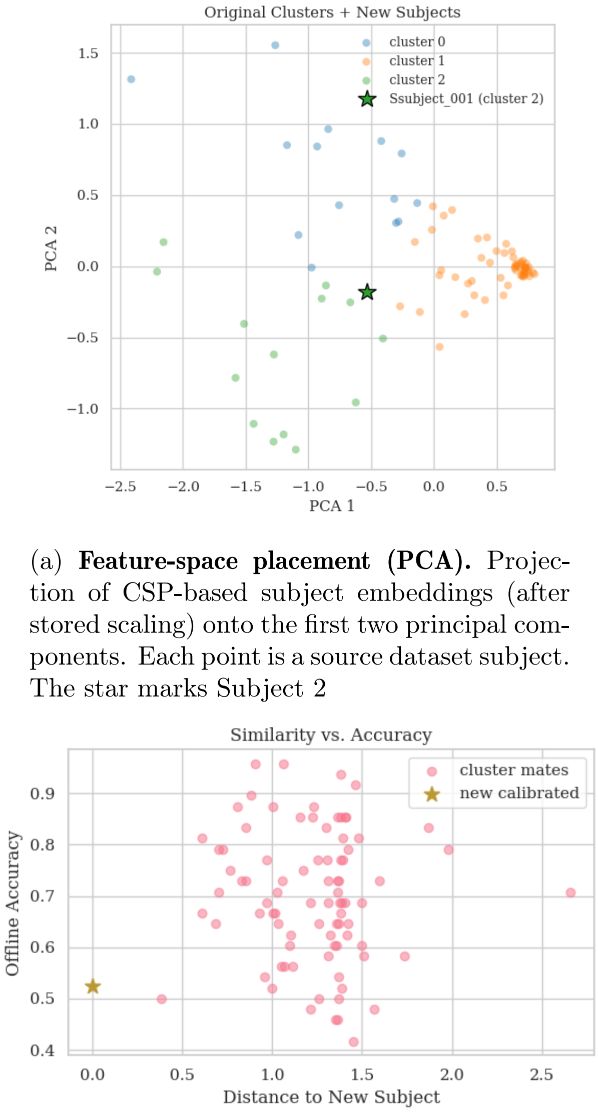
Figure 4.19: SUBJECT 2: CALIBRATION ACCURACY AS A FUNCTION OF k_{SHOT}. Accuracy vs. number of labeled trials per class (k_{shot}), for cluster-based and pooled initialization. Lines show mean accuracy over repeated stratified draws on disjoint test data.

Table 4.11: SUBJECT 2: ACCURACY AND CALIBRATION GAIN UNDER TWO INITIALIZATIONS. Calibrated accuracies and baseline-relative improvements for cluster- and pooled-initialized models across increasing k_{shot} values. Baseline accuracies (no calibration) are equal here (0.480). Δ_{cluster} and Δ_{pooled} quantify gains relative to the assigned zero-shot head. Accuracies are means over repeated stratified draws on disjoint test sets. Bold values indicate the better initialization at each k.

	Accuracy		Δ vs. baseline	
Setting	Cluster init	Pooled init	Δ_{cluster}	Δ_{pooled}

Assignment: cluster id = 2;
distance = 0.956
Zero-shot baselines
	0.480	**0.480**	–	–

k_{shot}=1	**0.500**	0.417	+0.020	−0.063
k_{shot}=2	**0.478**	**0.478**	−0.002	−0.002
k_{shot}=3	0.500	**0.545**	+0.020	+0.065
k_{shot}=4	**0.524**	0.476	+0.044	−0.004



(a) **Feature-space placement (PCA).** Projection of CSP-based subject embeddings (after stored scaling) onto the first two principal components. Each point is a source dataset subject. The star marks Subject 2

(b) **Within-cluster similarity vs. performance.** For each Cluster 2 peer, the x-axis shows its Euclidean distance to Subject 2 in the PCA+scaler space; the y-axis shows that peer's offline accuracy. The star marks Subject 1's calibrated point (shown for reference at $x = 0$.

Figure 4.20: Subject 2: feature-space placement and cluster diagnostics.

the cluster head. However, this advantage attenuated as more calibration data became available: by k=2, the pooled path matched or exceeded the cluster-initialized path, and at a higher number of calibration trials, pooled initialization consistently delivered better or more stable performance. These findings are descriptive (N=2), meant to validate the online deployment pipeline and its qualitative consistency with prior results, rather than to support generalizable inferences.

Taken together, the two case studies suggest a practical operating rule that aligns with Objective 1: clustered initialization is most valuable when only one labeled example per class can be collected at onboarding, while a pooled head initialization becomes competitive and often preferable once two or more labeled trials are available. This rule-of-thumb is supported here by consistent directional effects across both subjects and will be refined as additional users are acquired.

Table 4.12: In-session calibration: cluster vs. pooled initialization. Accuracy and the paired difference Δ = cluster – pooled for each subject at each number of calibration trials. Subjects are abbreviated as S1 and S2.

	S1 (Cluster 1, d=0.552)			S2 (Cluster 2, d=0.956)			Mean Δ
k_{shot}	Cluster	Pooled	Δ	Cluster	Pooled	Δ	
1	0.800	0.720	+0.080	0.500	0.417	+0.083	+**0.082**
2	0.667	0.625	+0.042	0.478	0.478	+0.000	+**0.021**
3	0.652	0.652	+0.000	0.500	0.546	−0.046	**-0.023**
4	0.682	0.773	−0.091	0.525	0.476	+0.049	**-0.021**

Chapter 5

Discussion and Future Work

This chapter interprets the findings of this project in the context of the stated objectives and situates them relative to prior work. Section 1 connects the empirical results from both the source dataset (Objective 1) and the target dataset (Objective 2) back to the objectives of this work and earlier literature, while Section 2 outlines promising directions for future work.

5.1 Discussion

This project pursued two main objectives: Objective 1: Establish a robust transfer learning pipeline for cross-subject MI decoding on a harmonized multi-subject dataset; Objective 2: Evaluate whether the pretrained framework generalizes effectively to prospectively recorded users under minimal calibration using an 8-channel EEG headset. The results are discussed below in the context of these objectives and contrasted with related studies.

Representation choice and stable population structure.
A core finding under Objective 1 is that CSP features revealed the most reliable and interpretable subject-space structure. Clustering based on CSP achieved stable partitions (high silhouette, strong ARI) with three clusters providing optimal separability while avoiding singleton or imbalanced groups. These findings partially align with prior work; Zhang et al. (2022) demonstrated that CSP-derived embeddings facilitate stratification for clustered MI-BCI modeling, but their experiments used 32-channel setups [85]. The findings of this work extend these observations to a low-density, 8-channel montage, demonstrating that stable population

structure can still be uncovered under more constrained hardware conditions. Conversely, Riemannian geometry-based embeddings, widely used in cross-subject BCIs [28, 83], failed to yield meaningful separability in the source dataset, likely due to the limited spatial resolution available in this context. Similarly, FBCSP representations frequently collapsed into tiny groups, contradicting their reported benefits in denser, high-SNR contexts [5]. For low-density MI-BCIs, simple, task-informed representations, such as CSP, seem to outperform more generic or geometry-driven embeddings. This insight sets the stage for downstream benefits.

Clustering with few-shot calibration enables effective personalization.

A central contribution of this work is showing that cluster-conditioned few-shot adaptation consistently outperforms pooled baselines when calibration data are scarce. Under LOSO, few-shot experiments on the source dataset, conditioning a lightweight subject head on three CSP-based clusters improved test accuracy by $\approx$ +4% on average relative to a pooled TL baseline trained with the same support size. These gains extended to κ, macro-F1, precision, and recall, underscoring robustness across evaluation metrics. This result extends the findings of Liu et al. (2017), who reported the benefits of hierarchical clustered multi-task learning [44], by showing that three clusters suffice to balance specialization against data sufficiency per head. Moreover, the framework presented in this project goes further by testing generalization on new users, providing a deployment-oriented perspective absent in most earlier works. In contrast, meta-learning approaches such as Model-Agnostic Meta-Learning [22] and prototypical networks [42, 71] attempt to accelerate personalization by optimizing explicitly for rapid adaptation. While such methods report strong few-shot performance, they require meta-trained initializations and often dense calibration episodes. Here, positive few-shot gains were achieved without meta-training, simplifying deployment for low-density MI-BCIs. For this 8-channel MI setting, *structure first, light calibration second* approach emerges as an effective and reproducible recipe. Stable CSP-based clustering provides meaningful priors.

Limited Benefits from Zero-Shot Transfer

Contrary to optimistic claims in parts of the EEG-TL literature [13, 28], assigning unseen users directly to pretrained cluster heads without calibration produced no consistent improvement over pooled zero-shot models. Although minor uplifts were observed for some subjects, these effects were heterogeneous and not statistically robust. This finding aligns with predictive coding intuitions: inter-subject variability in sensorimotor rhythms (topography, frequency tuning, activation latency) introduces latent factors that a purely static mapping cannot resolve. By contrast, even tiny calibration trials (1–4 trials per class) consistently unlocked substantial performance gains. While clustering enhances initialization, minimal subject-specific adaptation remains critical. A purely "plug-and-play" MI-BCI remains elusive due to current modeling capacity limitations.

Augmentation and Calibration: When More Data Doesn't Help

One surprising result is that common augmentations reported as beneficial elsewhere, such as Gaussian noise, time-warping, small frequency shifts, and mixup, did not improve performance in this setting, and sometimes degraded it. For example, time-warping, which often helps in continuous BCI paradigms [84], consistently reduced accuracy under clustered few-shot adaptation here. This contradicts prior findings [28, 40] and highlights task-specific dependencies. Short cue-locked epochs limit tolerance for temporal distortion, small calibration heads are prone to overfitting variability injected by augmentation, and the constrained 8-channel montage restricts the representational bandwidth, where augmentation typically helps. Augmentation strategies could be physiologically informed to avoid harming performance.

Similarly, increasing the number of calibration trials per class unexpectedly reduced accuracy on average. This effect is likely due to strategy drift, user fatigue, or intra-session variability outweighing benefits from extra labels, a nuance rarely discussed in prior TL studies. For low-density, cue-locked MI-BCIs, less might be more; short, high-quality calibrations outperform larger, noisier ones.

In-Session Personalization on Target Dataset

The in-session evaluations on the two prospectively recorded subjects provide a crucial reality check on the practical feasibility of clustered transfer learning under Objective 2. These experiments approximate a realistic deployment scenario: starting from a pretrained backbone learned on the source dataset, each new user is assigned to the closest population cluster based on their feature representation. A lightweight subject-specific head is then calibrated using a small number of labeled trials, enabling a direct comparison between cluster-initialized and pooled-initialized transfer paths under zero-shot and few-shot trials.

For Subject 1, results illustrate both the promise and limitations of clustering in practice. Zero-shot accuracy was identical under both pooled and cluster-initialized heads (0.654), indicating that routing to a pretrained cluster alone provides no benefit without supervision. However, with only one labeled trial per class, cluster initialization achieved an accuracy of 0.800, substantially outperforming pooled initialization at 0.720. This shows that cluster-specific heads offer more effective starting points than pooled models, as they are pre-adapted to subjects with similar neural patterns. Interestingly, this advantage was dampened as more data became available: with four trials per class, pooled initialization achieved an accuracy of 0.773, while clustered initialization regressed to 0.682.

Subject 2, by contrast, exhibited uniformly lower accuracies and negligible gains from calibration under either initialization strategy. While early cluster-based calibration offered a slight edge at $k = 1$, pooled initialization became superior at $k = 3$, though neither path showed stable improvement. Embedding diagnostics confirmed that Subject 2's representation fell farther from the cluster centroid (distance = 0.956), located near the periphery of Cluster 2 and close to bordering regions in the feature space. This spatial position likely weakened the relevance of the assigned cluster prior, explaining the limited benefits of clustering. Combined with lower signal quality or session-specific factors, this suggests that cluster-conditioned decoding may be less effective for edge-case users unless the clusters themselves are refined or augmented during deployment.

Together, these cases highlight the heterogeneity of adaptation success in MI-BCIs and clarify when clustered personalization is most beneficial. For subjects whose neural representations align closely with one of the discovered population clusters, as is the case with Subject 1, cluster-conditioned initialization provides measurable advantages when only one or two labeled trials per class are available. For others, such as Subject 2, pooled initialization becomes more robust once slightly more calibration data can be collected, while clustering provides little added value. While Objective 1 showed consistent benefits of clustered few-shot transfer across calibration sizes, the in-session evaluations reveal that individual factors, such as representational alignment or signal quality, can limit the practical advantage of clustering in deployment.

These patterns suggest a broader principle: structure and inductive bias are most valuable when supervision is scarce, while flexible adaptation becomes essential as more data become available. Rather than viewing clustered and pooled strategies as mutually exclusive, future work could explore hybrid strategies that interpolate between inductive priors and learned capacity depending on calibration availability. This perspective helps reconcile the tension between efficiency and flexibility in real-world BCI deployment.

Together, these results reveal a clear deployment guideline; cluster-conditioned personalization is most valuable when calibration resources are minimal, but pooled initialization becomes the safer and more consistent option once additional labeled trials are feasible. Compared to earlier transfer-learning studies in EEG-based BCIs [28, 32, 83], which evaluated adaptation only offline, these findings extend prior conclusions by demonstrating that clustered few-shot personalization can generalize prospectively to unseen users while also exposing its current limitations. Zero-shot transfer remains unreliable; subject-specific calibration, even if minimal, is almost always necessary, and individual differences can still significantly impact performance. These insights suggest promising extensions, such as adaptive routing, meta-learned initialization strategies, and uncertainty-aware calibration mechanisms, to handle challenging subjects better and further close the gap between offline benchmarks and real-world BCI deployment. This contrast is summarized in Table 4.12, where the one-shot advantage of clustered heads and later crossover to pooled superiority is shown across both subjects.

Concluding Remarks and Limitations

This project establishes a reproducible framework for label-efficient MI-BCIs by integrating representation learning, population clustering, and few-shot transfer learning. The findings demonstrate that learning a shared representation on pooled data, stratifying subjects into three CSP-based clusters, and calibrating a lightweight subject-specific head with only four labeled trials per class provides a practical pathway to accurate and data-efficient decoding.

Several limitations point toward opportunities for future work. These results build on an eight-channel montage and a binary MI paradigm; extending the framework to multi-class decoding, richer control schemes, and hybrid BCIs would test its generality. Zero-shot performance remains limited by simplistic k-means routing; probabilistic assignment or embedding-aware gating could better exploit population structure without supervision. Likewise, future research should revisit augmentation with physiologically constrained perturbations, integrate stronger priors into subject models, and explore meta-learning approaches or prototypical adapters to reduce calibration costs further.

Additional limitations include the small number of newly recorded subjects (N=2), which restricts the statistical generalizability of Objective 2 findings. While these case studies support the feasibility of the proposed pipeline, broader deployment studies are required to validate its reliability across diverse users. Moreover, the adaptation module used here is deliberately lightweight, and more expressive or meta-learned heads may offer improved personalization under non-stationary or noisy conditions. Future systems may benefit from embedding-aware cluster prediction and routing at onboarding, enabling more personalized adaptation paths and maximizing calibration efficiency under real-world constraints. Finally, while closed-loop prototypes were developed, they were not evaluated in this project; real-time studies are essential to quantify latency, stability, and user experience in operational BCIs.

In summary, this work demonstrates the value of CSP-based embeddings for revealing stable population structure, establishes that clustered few-shot calibration consistently outperforms pooled transfer under tight calibration trials, and challenges common assumptions that data augmentation and larger calibration sets universally improve performance. By validating these findings both offline and prospectively, this study bridges a critical gap between controlled cross-subject benchmarks and real-world BCI deployment, paving the way toward accurate, efficient, and reproducible MI-BCI systems.

5.2 Future Work

This section outlines concrete project proposals derived from the findings and limitations of this project. Each project proposal builds directly on the insights gained here and addresses open questions that remain critical for advancing subject-invariant MI decoding in EEG-based BCIs. The projects are organized by priority and designed to strike a balance between experimental feasibility and scientific impact.

Project 1: Adaptive Calibration and Transfer Learning for Personalized MI-BCIs A key outcome of this project is that cluster-conditioned few-shot calibration substantially improves decoding performance compared to pooled transfer when calibration data are scarce. However, the results also revealed marked heterogeneity across users in the target dataset. This suggests that a fixed calibration trial and uniform transfer strategy are insufficient for practical deployment.

Future work should focus on designing adaptive transfer learning frameworks where the degree of feature reuse, model fine-tuning, and calibration trial are dynamically optimized per subject. One promising direction is to integrate subject similarity measures derived from deep feature embeddings into the transfer process. For example, subjects located close to a cluster centroid in the CSP-based latent space can reuse more pre-trained features with minimal calibration. In contrast, subjects farther away may require deeper fine-tuning of early convolutional layers. Such similarity measures may include cosine distance in CSP- or Deep4Net-derived latent spaces, enabling fine-grained routing decisions based on representational proximity. Bayesian optimization or reinforcement learning agents could automatically select the optimal transfer configuration for each subject, eliminating the need for manual hyperparameter tuning.

In addition, incorporating uncertainty-aware calibration strategies would allow the system to stop collecting labels once confidence thresholds are met adaptively. Such systems could be evaluated in closed-loop setups to determine how quickly confidence thresholds can be reached without degrading user experience. By combining similarity-based routing, adaptive model reuse, and confidence-driven calibration, this project targets a central bottleneck for real-world MI-BCIs: delivering personalized decoding performance with minimal user burden. The outcome would be a highly flexible transfer pipeline that generalizes across diverse users while providing plug-and-play usability in practice.

Project 2: Hierarchical Multi-Task Learning with Adaptive Routing

This project demonstrated that population clustering into three CSP-based groups enabled effective few-shot personalization while avoiding over-fragmentation and instability. However, the results also highlighted a limitation: hard k-means assignments may fail for subjects that lie near cluster boundaries, forcing them into suboptimal priors. Moreover, the current framework uses a flat architecture, with a single shared encoder feeding either pooled or cluster-specific heads, limiting its flexibility.

A natural extension is to develop a hierarchical MTL framework that explicitly models variability at three levels: the population, clusters, and individuals. At the top level, a shared population encoder learns broadly generalizable features that apply to all subjects. At the intermediate level, cluster-adaptive layers specialize representations for homogeneous subgroups, refining the population encoding while preserving statistical power. Finally, subject-specific heads are fine-tuned with few-shot calibration to capture individual differences. This tiered structure mirrors the nested variability observed in this work, characterized by strong between-cluster separation and subtle within-cluster diversity.

To address the limitations of hard clustering, soft probabilistic routing can be introduced. Instead of assigning each subject to a single cluster, the model would compute posterior probabilities over multiple cluster heads and weight predictions accordingly. Soft routing could also be implemented via attention mechanisms that dynamically weight cluster heads during inference, enabling differentiable, end-to-end learning of the optimal adaptation path. For subjects near cluster boundaries, this approach leverages information from multiple priors rather than forcing a binary choice, mitigating one of the significant weaknesses revealed by the in-session experiments.

Such a hierarchical framework would not only improve decoding accuracy for challenging users but also provide neuroscientific insight by disentangling population-, cluster-, and subject-level sources of variability. Variants could further experiment with mixture-of-experts gating, allowing context-dependent selection of either shared, cluster-adapted, or subject-specific routes. It represents a principled approach to integrating global invariance with local specialization, thereby extending the strengths of this project into a more flexible and interpretable architecture.

Project 3: Physiologically Constrained Data Augmentation for Robust Personalization

One of the more interesting findings of this project is that standard data augmentation techniques failed to improve performance and often degraded clustered few-shot calibration. These results contradict earlier studies, where such augmentations have been shown to enhance generalization in EEG decoding. The discrepancy highlights a critical gap: in cue-locked MI tasks with short 3-second windows and only 8 EEG channels, naive augmentation strategies seem to violate label invariances and introduce physiologically implausible variability.

Future work could therefore explore physiologically constrained augmentation techniques tailored to MI-BCIs. Rather than indiscriminately perturbing the signal, augmentations should preserve the neurophysiological meaning of motor imagery patterns. Examples include: Phase-preserving narrow-band amplitude modulation, which perturbs oscillatory power without disrupting ERD/ERS dynamics. Latency-aware temporal jittering, introducing small timing shifts while respecting cue-locked epochs. Generative modeling approaches, such as variational autoencoders (VAEs) or diffusion models trained to synthesize realistic, label-consistent EEG. Additionally, augmentation strategies should be adaptable to the calibration regime. For example, when only one or two trials per class are available, augmentation can help expand limited datasets; conversely, under larger calibration trials, stronger regularization techniques (e.g., mixup on latent embeddings) may prevent overfitting without corrupting the label structure.

By grounding augmentation techniques in neurophysiological constraints, this project addresses a limitation exposed by this work and opens a pathway toward improving robustness without sacrificing label fidelity. Such methods could substantially enhance personalization, particularly for subjects whose signals deviate subtly from the training distribution.

Project 4: Cross-Hardware Robustness and Domain Generalization

All experiments in this project were conducted on a harmonized 8-channel dataset and validated on prospectively recorded users using a single Unicorn EEG headset. While this provided validity within a controlled setting, the results leave open a significant limitation: how well does the proposed pipeline generalize across different hardware, montages, and acquisition protocols? To quantify inter-device domain shifts, statistical measures such as Wasserstein distance or t-SNE cluster overlap can be used to assess representational drift and guide the need for adaptation.

Future work could systematically evaluate cross-hardware robustness and develop domain generalization techniques to enhance transferability across various acquisition setups. A first step involves curating a multi-center MI-EEG dataset combining recordings from diverse headsets, electrode configurations, and sampling rates. The proposed pipeline could then be benchmarked against naive pooling baselines to quantify performance degradation under domain shifts.

To mitigate these shifts, domain-invariant representation learning techniques could be incorporated into the training process. Examples include adversarial alignment methods, where a domain discriminator is trained jointly with the encoder to encourage indistinguishable latent representations across devices, as well as maximum mean discrepancy (MMD) losses that enforce statistical similarity between distributions. Alternatively, unsupervised domain adaptation approaches could be explored, leveraging unlabeled data from new devices to refine model parameters without additional calibration labels.

Such work is essential for scaling MI-BCI systems beyond tightly controlled laboratory environments. By combining cluster-conditioned few-shot adaptation with domain-robust representations, this project aims to produce pipelines capable of seamless transfer across hardware platforms and acquisition conditions—a key requirement for real-world deployment.

Project 5: Interpretable Clustering and Neurophysiological Feature Attribution
This project established that CSP-based embeddings reveal a stable, interpretable subject-space structure, enabling population clustering that drives effective few-shot personalization. However, the underlying neurophysiological basis of these clusters remains unexplored. Why do specific subjects group together, and which brain rhythms or spatial patterns define these distinctions? Answering these questions would both deepen our understanding of neuroscience and improve cluster-based modeling strategies.

Future work could integrate explainable AI (XAI) techniques to uncover the features driving cluster formation. For example, layer-wise relevance propagation or integrated gradients could be applied to CSP-derived embeddings or intermediate model activations to identify the most influential time-frequency components and electrode locations. To improve interpretability, cluster-wise spectral fingerprints or averaged topographical maps could be computed, providing direct visualizations of the neurophysiological signatures associated with each cluster. These analyses would clarify whether clusters reflect distinct ERD/ERS patterns, spectral peaks, or spatial lateralization effects, providing biologically meaningful interpretations of inter-subject variability.

Insights from these analyses could also inform better cluster-based calibration strategies. For instance, subjects identified as sharing strong contralateral mu desynchronization patterns could be initialized with priors optimized explicitly for that feature set. Beyond technical improvements, interpretable clustering bridges the gap between machine learning and neuroscience, linking latent model structure to underlying brain dynamics.

This approach would transform clustering from a purely algorithmic tool into a neuroscientific instrument, offering both improved model personalization and richer insights into the variability of motor imagery across individuals.

Declarations

Throughout this work, ChatGPT by OpenAI [55, 56] and Grammarly were used to improve written content through paraphrasing, grammar correction, and language refinement. Additionally, ChatGPT supported the implementation of the transfer learning pipeline.

References

[1] Acqualagna, L., Botrel, L., Vidaurre, C., Kübler, A. and Blankertz, B. [2016], 'Large-scale assessment of a fully automatic co-adaptive motor imagery-based brain–computer interface', *PLoS ONE* **11**(2), e0148886.

[2] Ahn, M. and Jun, S. C. [2015], 'Performance variation in motor imagery brain–computer interface: A brief review', *Journal of Neuroscience Methods* **243**, 103–110.

[3] Alamgir, M., Grosse-Wentrup, M. and Altun, Y. [2010], Multitask learning for brain-computer interfaces, *in* Y. W. Teh and M. Titterington, eds, 'Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics', Vol. 9 of *Proceedings of Machine Learning Research*, PMLR, Chia Laguna Resort, Sardinia, Italy, pp. 17–24.

[4] Alimardani, M., Kocken, S. and Leeuwis, N. [2023], 'End-to-end deep transfer learning for calibration-free motor imagery brain–computer interfaces', *arXiv preprint arXiv:2307.12827* .

[5] Ang, K. K., Chin, Z. Y., Zhang, H. and Guan, C. [2008], Filter bank common spatial pattern (FBCSP) in brain–computer interface, *in* '2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)', IEEE, Hong Kong, China, pp. 2390–2397.

[6] Autthasan, P. et al. [2022], 'Min2net: End-to-end multi-task learning for subject-independent motor imagery eeg classification', *IEEE Transactions on Biomedical Engineering* **69**(6), 2105–2118.

[7] Avitan, L., Teicher, M. and Abeles, M. [2009], 'Eeg generator—a model of potentials in a volume conductor', *Journal of Neurophysiology* **102**(5), 3046–3059.

[8] Bashashati, A., Fatourechi, M., Ward, R. K. and Birch, G. E. [2007], 'A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals', *Journal of Neural Engineering* **4**(2), R32.

[9] Bazzigher, A. and Widmer, Z. [2025], 'I know what you're thinking! – cluster-based transfer learning for motor imagery brain–computer interfaces'.

[10] Beisteiner, R., Höllinger, P., Lindinger, G., Lang, W. and Berthoz, A. [1995], 'Mental representations of movements: Brain potentials associated with imagination of hand movements', *Electroencephalography and Clinical Neurophysiology* **96**(2), 183–193.

[11] Blankertz, B., Sannelli, C., Halder, S., Hammer, E. M., Kübler, A., Müller, K.-R., Curio, G. and Dickhaus, T. [2010], 'Neurophysiological predictor of smr-based bci performance', *NeuroImage* **51**(4), 1303–1309.

[12] Byczuk, M., Poryzała, P. and Materka, A. [2012], SSVEP-based brain–computer interface: On the effect of stimulus parameters on VEPs spectral characteristics, *in* Z. S. Hippe, J. L. Kulikowski and T. Mroczek, eds, 'Human–Computer Systems Interaction: Backgrounds and Applications 2. Advances in Intelligent and Soft Computing', Springer, Berlin, Heidelberg, pp. 3–14.

[13] Cai, Y., She, Q., Ji, J., Ma, Y., Zhang, J. and Zhang, Y. [2022], 'Motor imagery eeg decoding using manifold embedded transfer learning', *Journal of Neuroscience Methods* **370**, 109489.

[14] Caruana, R. [1997], 'Multitask learning', *Machine Learning* **28**, 41–75.

[15] Caruana, R. A. [1993], 'Multitask learning: A knowledge-based source of inductive bias', *Machine Learning* pp. 41–48.

[16] Chen, P., Wang, H., Sun, X., Li, H., Grebogi, C. and Gao, Z. [2022], 'Transfer learning with optimal transportation and frequency mixup for eeg-based motor imagery recognition', *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **30**, 2866–2875.

[17] Chen, X., Wang, Y., Nakanishi, M., Gao, X., Jung, T. and Gao, S. [2015], 'High-speed spelling with a noninvasive brain–computer interface', *Proceedings of the National Academy of Sciences of the United States of America* **112**(44), E6058–E6067.

[18] da Silva, F. L., Mulert, C. and Lemieux, L. [2010], Eeg-fmri: Physiological basis, technique, and applications, *in* 'EEG: Origin and Measurement', Springer, New York, pp. 19–39.

[19] Dickhaus, T., Sannelli, C., Müller, K. R., Curio, G. and Blankertz, B. [2009], 'Predicting BCI performance to study BCI illiteracy', *BMC Neuroscience* **10**(Suppl 1), P84.

[20] Elashmawi, W. H., Ayman, A., Antoun, M., Mohamed, H., Mohamed, S. E., Amr, H., Talaat, Y. and Ali, A. [2024], 'A comprehensive review on brain–computer interface (bci)-based machine and deep learning algorithms for stroke rehabilitation', *Applied Sciences* **14**(14), 6347.

[21] Fazli, S., Popescu, F., Danóczy, M., Blankertz, B., Müller, K.-R. and Grozea, C. [2009], 'Subject-independent mental state classification in single trials', *Neural Networks* **22**(9), 1305–1312.

[22] Finn, C., Abbeel, P. and Levine, S. [2017], 'Model-Agnostic Meta-Learning for fast adaptation of deep networks', *arXiv preprint arXiv:1703.03400* .

[23] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R. et al. [2000], 'Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals', *Circulation* **101**(23), e215–e220.

[24] Han, D.-K., Jang, G.-D. and Kim, D.-Y. [2023], Subject-independent brain–computer interfaces with open-set subject recognition, *in* 'Proceedings of the IEEE International Conference on Brain-Computer Interfaces (BCI)', pp. 1–4.

[25] He, B., Baxter, B., Edelman, B. J., Cline, C. C. and Ye, W. W. [2015], 'Noninvasive brain–computer interfaces based on sensorimotor rhythms', *Proceedings of the IEEE* **103**(6), 907–925.

[26] He, B., ed. [2005], *Neural Engineering*, Vol. 2, Kluwer Academic/Plenum.

[27] He, B., Yang, L., Wilke, C. and Yuan, H. [2011], 'Electrophysiological imaging of brain activity and connectivity—challenges and opportunities', *IEEE Transactions on Biomedical Engineering* **58**(7), 1918–1931.

[28] He, H. and Wu, D. [2020], 'Transfer learning for brain–computer interfaces: A euclidean space data alignment approach', *IEEE Transactions on Biomedical Engineering* **67**(2), 399–410.

[29] Holmes, G. L., Khazipov, R., Blum, A. S. and Rutkove, S. B. [2007], The clinical neurophysiology primer, pp. 19–33.

[30] Huang, G., Zhao, Z., Zhang, S., Hu, Z., Fan, J., Fu, M., Chen, J., Xiao, Y., Wang, J. and Dan, G. [2023], 'Discrepancy between inter- and intra-subject variability in EEG-based motor imagery brain–computer interface: Evidence from multiple perspectives', *Frontiers in Neuroscience* **17**, 1122661.

[31] Islam, M. K. and Rastegarnia, A. [2023], 'Editorial: Recent advances in eeg (non-invasive) based bci applications', *Frontiers in Computational Neuroscience* **17**, 1151852.

[32] Jayaram, V., Alamgir, M., Altun, Y., Schölkopf, B. and Grosse-Wentrup, M. [2016], 'Transfer learning in brain–computer interfaces', *IEEE Computational Intelligence Magazine* **11**(1), 20–31.

[33] Jeunet, C., Jahanpour, E. and Lotte, F. [2016], 'Why standard brain–computer interface (bci) training protocols should be changed: An experimental study', *Journal of Neural Engineering* **13**(3), 036024.

[34] Jeunet, C., N'Kaoua, B., Subramanian, S., Hachet, M. and Lotte, F. [2015], 'Predicting mental imagery-based bci performance from personality, cognitive profile and neurophysiological patterns', *PLoS ONE* **10**(12), e0143962.

[35] Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S., Hudspeth, A. J. and Mack, S. [2000], *Principles of Neural Science*, 4 edn, McGraw-Hill, New York.

[36] Khademi, Z., Ebrahimi, F. and Montazery Kordy, H. [2023], 'A review of critical challenges in MI-BCI: From conventional to deep learning methods', *Journal of Neuroscience Methods* **383**, 109736.

[37] Kim, Y.-T., Lee, S., Kim, H., Lee, S.-B., Lee, S.-W. and Kim, D.-J. [2019], Reduced burden of individual calibration process in brain–computer interface by clustering the subjects based on brain activation, *in* '2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)', pp. 2139–2143.

[38] Kostas, D. and Rudzicz, F. [2020], 'Thinker invariance: Enabling deep neural networks for BCI across more people', *Journal of Neural Engineering* **17**(5), 056008.

[39] Krauledat, M., Schröder, M., Blankertz, B. and Müller, K.-R. [2006], Reducing calibration time for brain–computer interfaces: A clustering approach, *in* 'Advances in Neural Information Processing Systems 19', MIT Press, pp. 753–760.

[40] Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P. and Lance, B. J. [2018], 'Eegnet: A compact convolutional neural network for eeg-based brain–computer interfaces', *Journal of Neural Engineering* **15**(5), 056013.

[41] Lebedev, M. A. and Nicolelis, M. A. L. [2006], 'Brain–machine interfaces: Past, present and future', *Trends in Neurosciences* **29**(9), 536–546.

[42] Li, C., Denison, T. and Zhu, T. [2024], 'A survey of few-shot learning for biomedical time series', *arXiv preprint arXiv:2405.02485* .

[43] Liao, W., Liu, H. and Wang, W. [2025], 'Advancing bci with a transformer-based model for motor imagery classification', *Scientific Reports* **15**, 23380.

[44] Liu, A. A., Su, Y. T., Nie, W. Z. and Kankanhalli, M. [2017], 'Hierarchical clustering multi-task learning for joint human action grouping and recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**(1), 102–114.

[45] Liu, X. Y., Wang, W. L., Liu, M. et al. [2025], 'Recent applications of eeg-based brain–computer interface in the medical field', *Military Medical Research* **12**, 14.

[46] Long, T., Wan, M., Jian, W., Dai, H., Nie, W. and Xu, J. [2023], 'Application of multi-task transfer learning: The combination of EA and optimized subband regularized CSP to classification of 8-channel EEG signals with small dataset', *Frontiers in Human Neuroscience* **17**, 1143027.

[47] Lotte, F. [2015], 'Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces', *Proceedings of the IEEE* **103**(6), 871–890.

[48] Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A. and Yger, F. [2018], 'A review of classification algorithms for eeg-based brain–computer interfaces: A 10-year update', *Journal of Neural Engineering* **15**(3), 031005.

[49] Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F. and Arnaldi, B. [2007], 'A review of classification algorithms for eeg-based brain–computer interfaces', *Journal of Neural Engineering* **4**(2), R1.

[50] Lotte, F. and Guan, C. [2010], Learning from other subjects helps reducing brain–computer interface calibration time, *in* '2010 IEEE International Conference on Acoustics, Speech and Signal Processing', Dallas, TX, USA, pp. 614–617.

[51] LSL [2024], 'Lab streaming layer'.
URL: *https://github.com/sccn/labstreaminglayer*

[52] Lu, H., Eng, H.-L., Guan, C., Plataniotis, K. N. and Venetsanopoulos, A. N. [2010], 'Regularized common spatial pattern with aggregation for eeg classification in small-sample setting', *IEEE Transactions on Biomedical Engineering* **57**(12), 2936–2946.

[53] Ma, B.-Q., Li, H., Zheng, W.-L. and Lu, B.-L. [2019], Reducing the subject variability of eeg signals with adversarial domain generalization, *in* 'Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part I', Springer, Berlin, Heidelberg, pp. 30–42.

[54] Nicolas-Alonso, L. F. and Gomez-Gil, J. [2012], 'Brain computer interfaces, a review', *Sensors* **12**(2), 1211–1279.

[55] OpenAI [2024], 'Chatgpt: GPT-4.0 model', Large language model.
URL: *https://chat.openai.com/*

[56] OpenAI [2025], 'Chatgpt: GPT-5.0 model', Large language model.
URL: *https://chat.openai.com/*

[57] Pan, S. J. and Yang, Q. [2010], 'A survey on transfer learning', *IEEE Transactions on Knowledge and Data Engineering* **22**(10), 1345–1359.

[58] Pfurtscheller, G. [1992], 'Event-related synchronization (ERS): An electrophysiological correlate of cortical areas at rest', *Electroencephalography and Clinical Neurophysiology* **83**(1), 62–69.

[59] Pfurtscheller, G. and Lopes da Silva, F. H. [1999], 'Event-related EEG/MEG synchronization and desynchronization: Basic principles', *Clinical Neurophysiology* **110**(11), 1842–1857.

[60] Pineda, J. A. [2005], 'The functional significance of mu rhythms: Translating "seeing" and "hearing" into "doing"', *Brain Research Reviews* **50**(1), 57–68.

[61] Rashid, M., Sulaiman, N., Abdul Majeed, A. P. P., Musa, R. M., Ab. Nasir, A. F., Bari, B. S. and Khatun, S. [2020], 'Current status, challenges, and possible solutions of eeg-based brain–computer interface: A comprehensive review', *Frontiers in Neurorobotics* **14**, 25.

[62] Ruder, S. [2017], 'An overview of multi-task learning in deep neural networks'.

[63] Saha, S., Ahmed, K. I. U., Mostafa, R., Hadjileontiadis, L. and Khandoker, A. [2018], 'Evidence of variabilities in eeg dynamics during motor imagery-based multiclass brain–computer interface', *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **26**(2), 371–382.

[64] Sannelli, C., Vidaurre, C., Müller, K. R. and Blankertz, B. [2019], 'A large scale screening study with a smr-based bci: Categorization of bci users and differences in their smr activity', *PLoS ONE* **14**(1), e0207351.

[65] Sauceda, J., Marquez, B. and Esqueda Elizondo, J. [2024], 'Emotion classification from electroencephalographic signals using machine learning', *Brain Sciences* **14**(12), 1211.

[66] Schirrmeister, R. T., Springenberg, J. T., Fiederer, L. D. J. et al. [2017], 'Deep learning with convolutional neural networks for EEG decoding and visualization', *Human Brain Mapping* **38**(11), 5391–5420.

[67] Schomer, D. L. and Lopes da Silva, F. H., eds [2011], *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, 6 edn, Lippincott Williams & Wilkins.

[68] Sellers, E. W., Arbel, Y. and Donchin, E. [2012], BCIs that use P300 event-related potentials, *in* J. Wolpaw and E. W. Wolpaw, eds, 'Brain–Computer Interfaces: Principles and Practice', Oxford University Press, New York, NY, pp. 215–226.

[69] Shu, X., Chen, S., Yao, L., Sheng, X., Zhang, D., Jiang, N. and Zhu, X. [2018], 'Fast recognition of bci-inefficient users using physiological features from eeg signals: A screening study of stroke patients', *Frontiers in Neuroscience* **12**, 93.

[70] Singh, A., Hussain, A. A., Lal, S. and Guesgen, H. W. [2021], 'A comprehensive review on critical issues and possible solutions of motor imagery based electroencephalography brain–computer interface', *Sensors* **21**(6), 2173.

[71] Snell, J., Swersky, K. and Zemel, R. S. [2017], 'Prototypical networks for few-shot learning', *arXiv preprint arXiv:1703.05175* .

[72] Stieger, J. R., Engel, S. A. and He, B. [2021], 'Continuous sensorimotor rhythm based brain–computer interface learning in a large population', *Scientific Data* **8**(1), 98.

[73] Thomas, A. W., Lindenberger, U., Samek, W. and Müller, K.-R. [2021], 'Evaluating deep transfer learning for whole-brain cognitive decoding'.

[74] Tibermacine, I. E., Russo, S., Tibermacine, A., Rabehi, A., Nail, B., Kadri, K. and Napoli, C. [2024], 'Riemannian geometry-based EEG approaches: A literature review', *arXiv preprint arXiv:2407.20250* .

[75] *Unicorn Hybrid Black* [2025], [Online]. g.tec medical engineering GmbH.
URL: *https://www.gtec.at/product/unicorn-hybrid-black/*

[76] Weiss, M. [2024], Generative data augmentation for EEG-based brain–machine interfaces: Offline analysis and closed-loop prototyping tool, Master's thesis.

[77] Wierzgala, P., Zapala, D., Wojcik, G. M. and Masiak, J. [2018], 'Most popular signal processing methods in motor-imagery bci: A review and meta-analysis', *Frontiers in Neuroinformatics* **12**, 78.

[78] Wu, D., Jiang, X. and Peng, R. [2022], 'Transfer learning for motor imagery based brain–computer interfaces: A tutorial', *Neural Networks* **153**, 235–253.

[79] Wu, D., Xu, Y. and Lu, B.-L. [2020], 'Transfer learning for EEG-based brain–computer interfaces: A review of progress made since 2016', *arXiv preprint arXiv:2004.06286* .

[80] Xue, Y., Liao, X., Carin, L. and Krishnapuram, B. [2007], 'Multi-task learning for classification with dirichlet process priors', *Journal of Machine Learning Research* **8**, 35–63.

[81] Yao, L., Sheng, X., Zhang, D., Jiang, N., Mrachacz-Kersting, N., Zhu, X. and Farina, D. [2017], 'A stimulus-independent hybrid bci based on motor imagery and somatosensory attentional orientation', *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **25**(9), 1674–1682.

[82] Yi, W., Qiu, S., Wang, K., Qi, H., Zhang, L., Zhou, P. et al. [2014], 'Evaluation of EEG oscillatory patterns and cognitive process during simple and compound limb motor imagery', *PLOS ONE* **9**(12), e114853.

[83] Zanini, P., Congedo, M., Jutten, C., Said, S. and Berthoumieu, Y. [2018], 'Transfer learning: A riemannian geometry framework with applications to brain–computer interfaces', *IEEE Transactions on Biomedical Engineering* **65**(5), 1107–1116.

[84] Zhang, R., Li, X., Wang, Y., Liu, B., Shi, L., Chen, M. and Hu, Y. [2019], 'Using brain network features to increase the classification accuracy of mi–bci inefficiency subjects', *IEEE Access* **7**, 74490–74499.

[85] Zhang, Y., Zhou, T., Wu, W., Xie, H., Zhu, H., Zhou, G. and Cichocki, A. [2022], 'Improving eeg decoding via clustering-based multitask feature learning', *IEEE Transactions on Neural Networks and Learning Systems* **33**(8), 3587–3597.

List of Figures

List of Tables

Appendix A

Supplementary Methods

A.1 Acquisition and External Tooling

This subsection describes the tools adopted for the acquired subject recordings (target dataset) via the Unicorn Headset.

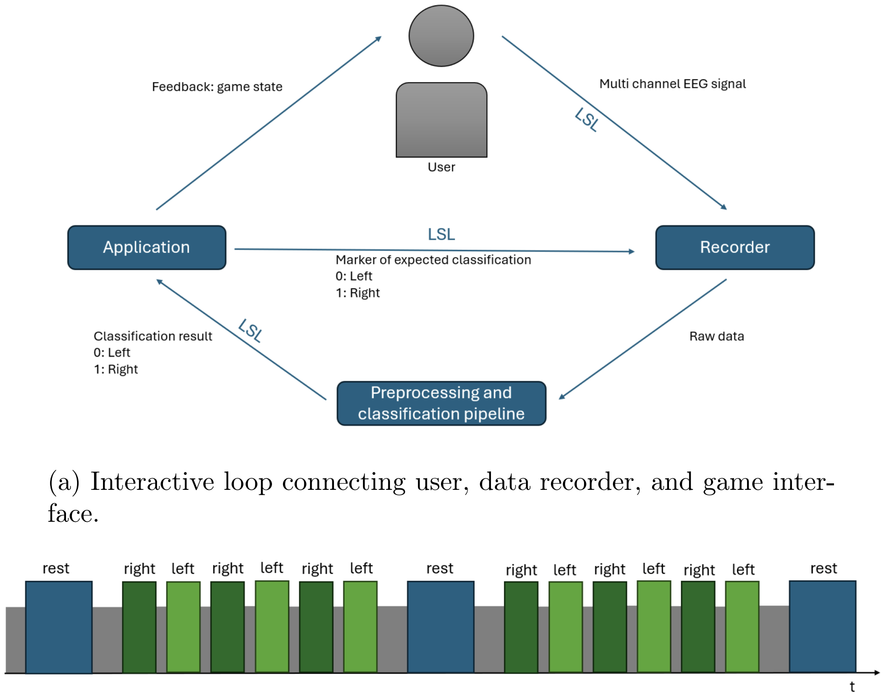
Communication between the independent services is handled by the Lab Streaming Layer (LSL) [51], an open-source middleware framework for networked acquisition that supports streaming, reception, time-synchronization, and recording of neural, physiological, and behavioral signals. The vendor application published the EEG stream; the recorder subscribed via `mne-lsl`, while the paradigm/game emitted event markers over a dedicated marker stream using `pylsl`. Tooling was implemented in Python with widely supported libraries to maximize portability and maintainability: MNE for I/O, filtering, epoching, and visualization; NumPy/SciPy for numerical routines; scikit-learn for baseline machine-learning utilities; and Pygame for the paradigm interface. Configuration was managed via structured configuration files to enable repeatable runs and explicit parameter recording. Recordings were persisted in an MNE-compatible format with annotations derived from the marker stream so that EEG and events remained aligned at sample resolution.

A.2 Recorder Application

The recorder used in this study was originally developed by Manuel Weiss [76]. The architecture separates concerns into (i) paradigm/data generation, (ii) signal acquisition, (iii) preprocessing, (iv) optional training/inference, and (v) device control. Earlier iterations incorporated a motor-imagery paradigm and provided standard preprocessing options (such as CSP, PCA) with conventional classifiers (such as SVM). Subsequent extensions introduced runtime configuration via a simple UI, template configurations for common tasks (training, inference), and a continuous-recording mode that appends labeled data segments at fixed intervals, designed to support both interactive session logging and structured data collection blocks. Subject-clustering utilities (e.g. k-means, Ward, GMM) and hooks for transfer-learning workflows were also integrated to facilitate base-model selection per cluster. For the purposes of this thesis, the recorder functioned as the acquisition and labeling hub. A prototype integration of the thesis decoder was explored; closed-loop processing was not adopted for evaluation due to unresolved hardware issues. All collected signals and markers were persisted for subsequent offline preprocessing and analysis. Recordings were saved per session with accompanying metadata (subject identifier, session index, montage, sampling rate, start time), and event markers were fused into the same file as annotations to preserve synchronization.

A.3 Game Application

The paradigm/game application was developed by Annina Bazzigher and Zoe Widmer. It provides structured cueing for motor imagery and, when enabled, interactive feedback within a simple 2-D interface where a cyclist moves laterally to avoid obstacles. EEG is streamed from the headset to the recorder via LSL, while event markers for left-hand motor imagery (MI), right-hand MI, and rest are emitted from the game over a dedicated LSL marker stream. When inference is active, directional control can be derived from a configurable majority vote over recent classifier outputs; in this thesis, closed-loop control was prototyped but not used for evaluation. Two operating modes are available. Gameplay mode enables closed-loop interaction and logs intended directions as markers. Recording mode delivers a fixed, labelable schedule suitable for supervised data collection. Each recording session followed a standardized sequence designed to produce time-aligned labels: an on-screen countdown preceded each block; a 10-s rest interval established a baseline; a motor-imagery block then presented cued trials in which the road segment indicating the impending obstacle was highlighted in blue and an arrow specified the required imagery direction. Participants sustained the corresponding hand-movement imagery while the arrow remained visible. Within each MI block, six trials per direction (left and right) were presented for a fixed duration configured in the application (held constant across trials); the MI block was repeated once within the session. A final 10-second rest interval concluded the sequence. Thus, per session, the schedule yielded 12 left-MI trials and 12 right-MI trials, interleaved with three 10-s rest periods. Marker emission was synchronized to cue onset so that labels and EEG remained aligned at sample resolution via LSL's time-correction mechanism. In this thesis, the application was employed primarily in recording mode to generate precisely time-stamped labels for the acquisition-only dataset used in offline analyses. The application also exposes exploratory metrics (e.g., accuracy, F1-score, and information-transfer rate) to support rapid prototyping; these diagnostics were not used for the primary evaluations reported later.



(a) Interactive loop connecting user, data recorder, and game interface.

(b) Interactive loop connecting user, data recorder, and game interface.

Figure A.1: Overview of user interaction and data flow in the recording setup [9].

A.4 Hyperparameter Configurations

Table A.1: DEEP4NET ARCHITECTURE USED ACROSS ALL EXPERIMENTS. The only variation is the pooling mode: *mean* for single-subject and *max* for pooled training.

Parameter	Value
Dropout	0.25
Temporal filters	25
Spatial filters	25
Temporal kernel length	10
Temporal pooling length	3
Epochs	100
Batch size	64
Learning rate	0.0005
Optimizer	Adam
Loss	Cross-entropy
Weight decay	0.001
Pooling mode	*mean (single)* *max (pooled)*

Table A.2: HYPERPARAMETERS FOR MULTI-TASK LEARNING MODEL. Pooled model uses a single head ($k = 1$), while clustered uses one head per cluster ($k = 3$).

Parameter	**Value**
Dropout (backbone)	0.5
Temporal filters	25
Spatial filters	25
Temporal kernel length	10
Temporal pooling length	3
Hidden dimension (head)	128
Dropout (head)	0.5
Optimizer	Adam
Learning rate	0.0001
Weight decay	0.001
Epochs	100
Batch size	64
Number of clusters (k)	1 (pooled), 3 (clustered)

Table A.3: HYPERPARAMETERS FOR TRANSFER LEARNING MODEL. Across zero-shot, few-shot, and pooled modes; Each TL model initializes from the pretrained clustered MTL backbone and adapts a lightweight subject-specific head.

Parameter	**Value**
Pretrained model	MTL (clustered, k=3)
Freeze backbone	False
Backbone learning rate	1×10^{-5}
Head learning rate	1×10^{-3}
Hidden dimension (head)	128
Dropout (head)	0.5
Optimizer	Adam
Weight decay	0.001
Epochs	100
Batch size	64
Transfer modes	Zero-Shot, Universal, Few-Shot ($k_{\text{shot}} = 4$)

Table A.4: HYPERPARAMETERS FOR IN-SESSION TRANSFER LEARNING. The backbone remains frozen; only the head is updated using a small learning rate for 10 calibration epochs.

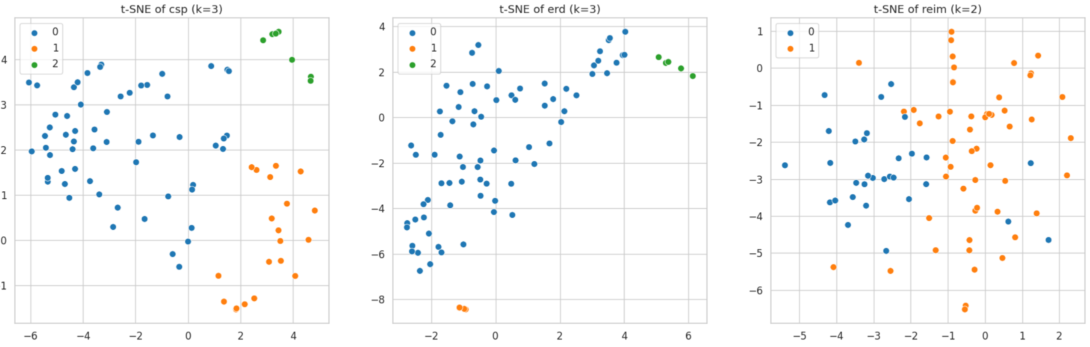
Parameter	**Value**
Calibration trials per class (k_{calib})	4
Calibration epochs	10
Calibration learning rate	0.0005
Fine-tuned component	Head only
Dropout (head)	0.5

Appendix B

Supplementary Results

B.1 Extended Results for Subject-Level Representations for Population Clustering

This subsection provides additional tables and figures supporting the representation and clustering analyses from subsection 4.1. It includes comparative metrics across feature families (CSP, ERD/ERS, Riemannian), quantitative clustering indicators (Silhouette, ARI, Calinski–Harabasz, Inertia), and subject-space visualizations via t-SNE projections. These results reinforce the selection of CSP-based embeddings as the most stable and compact feature representation for cross-subject stratification.



(a) CSP (k=3)

(b) ERD/ERS (k=3)

(c) Riemannian (k=2)

Figure B.1: t-SNE visualisations of the same subject-space embeddings used for clustering.

Table B.1: Full clustering quality metrics across all feature extraction methods and cluster counts. Silhouette↑, Calinski–Harabasz (CH)↑, Inertia↓, ARI mean (μ)↑, ARI std (σ). *Cluster sizes* lists the number of subjects per cluster. *Tiny cluster (T/F)* indicates whether any cluster has < 5 subjects.

Method	k	Silh.	CH	Inertia	ARI μ	ARI σ	Imbal.	Cluster sizes	Tiny
csp	2	0.544	61.574	195.194	0.863	0.160	2.864	[63,22]	F
csp	3	0.563	62.979	134.066	0.862	0.095	8.286	[58,20,7]	F
csp	4	0.558	57.096	109.162	0.848	0.096	8.286	[58,11,7,9]	F
csp	5	0.561	55.424	90.157	0.855	0.089	18.667	[56,11,6,9,3]	T
csp	6	0.486	55.110	75.758	0.791	0.076	16.667	[50,9,4,9,3,10]	T
csp	8	0.473	56.123	55.719	0.797	0.057	23.000	[46,6,5,6,10,2,7,3]	T
ers/ers	2	0.504	55.556	101.836	0.264	0.313	5.071	[71,14]	F
ers/ers	3	0.652	87.348	54.306	0.560	0.214	14.800	[74,6,5]	F
ers/ers	4	0.510	116.050	32.087	0.673	0.230	28.000	[21,6,2,56]	T
ers/ers	5	0.406	120.003	24.285	0.507	0.206	37.000	[34,6,1,37,7]	T
ers/ers	6	0.444	154.733	15.751	0.573	0.151	33.000	[30,6,1,33,4,11]	T
ers/ers	8	0.448	203.476	8.719	0.764	0.144	26.000	[22,6,4,13,1,11,2,26]	T
fbcsp	2	0.801	38.208	931.288	0.167	0.288	84.000	[84,1]	T
fbcsp	3	0.550	58.167	562.286	0.915	0.118	63.000	[63,1,21]	T
fbcsp	4	0.573	58.485	429.548	0.909	0.105	63.000	[63,1,9,12]	T
fbcsp	5	0.587	52.321	376.101	0.797	0.156	63.000	[63,1,4,9,8]	T
fbcsp	6	0.579	49.874	327.191	0.821	0.120	61.000	[61,1,2,7,7,7]	T
fbcsp	8	0.468	56.009	223.252	0.835	0.085	55.000	[55,3,1,7,6,1,1,11]	T
reim	2	0.084	7.668	2408.853	0.204	0.295	1.742	[31,54]	F
reim	3	0.066	6.245	2283.600	0.177	0.121	2.786	[14,39,32]	F
reim	4	0.079	6.354	2130.152	0.293	0.149	4.714	[7,33,23,22]	F
reim	5	0.082	6.057	2019.767	0.372	0.149	4.500	[6,27,21,23,8]	F
reim	6	0.079	6.362	1876.042	0.335	0.118	27.000	[7,27,19,23,8,1]	T
reim	8	0.058	7.392	1573.819	0.366	0.092	22.000	[7,12,21,20,1,1,1,22]	T

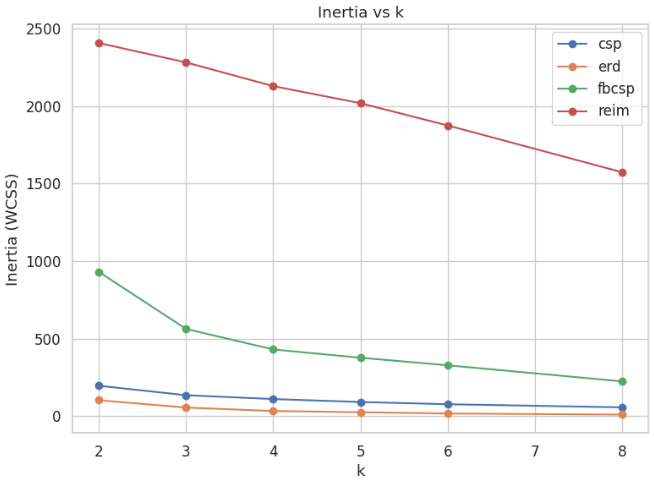


Figure B.2: Inertia (within-cluster sum of squares) versus cluster count k for all feature types. As expected, Inertia decreases monotonically. No sharp knee is visible, supporting the decision to rely on Silhouette and ARI for choosing k.

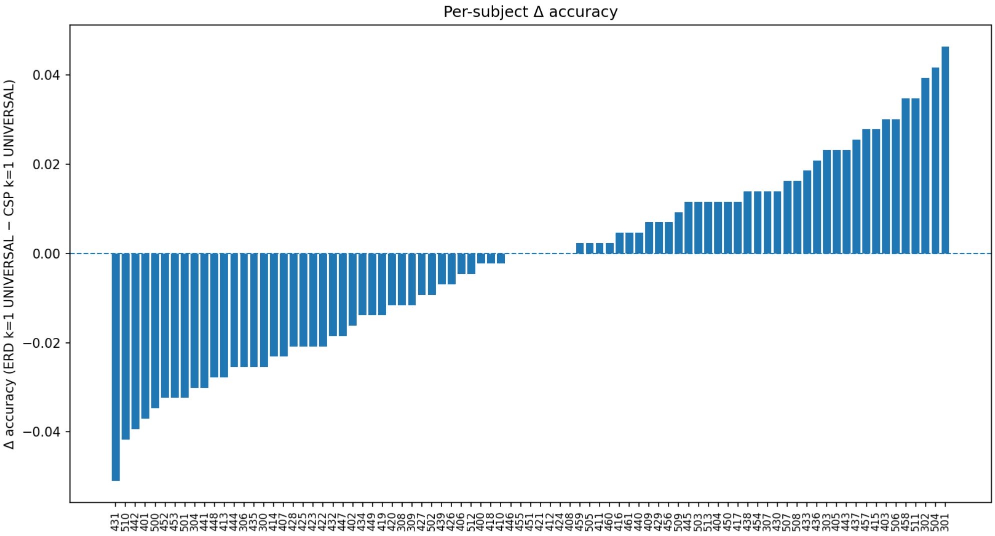


Figure B.3: PER-SUBJECT Δ ACCURACY IN THE POOLED SETTING $(k = 1)$. Accuracy differences between ERD/ERS and CSP representations are small and balanced. Confirms that under pooled modeling, representation choice has minimal effect on decoding performance.

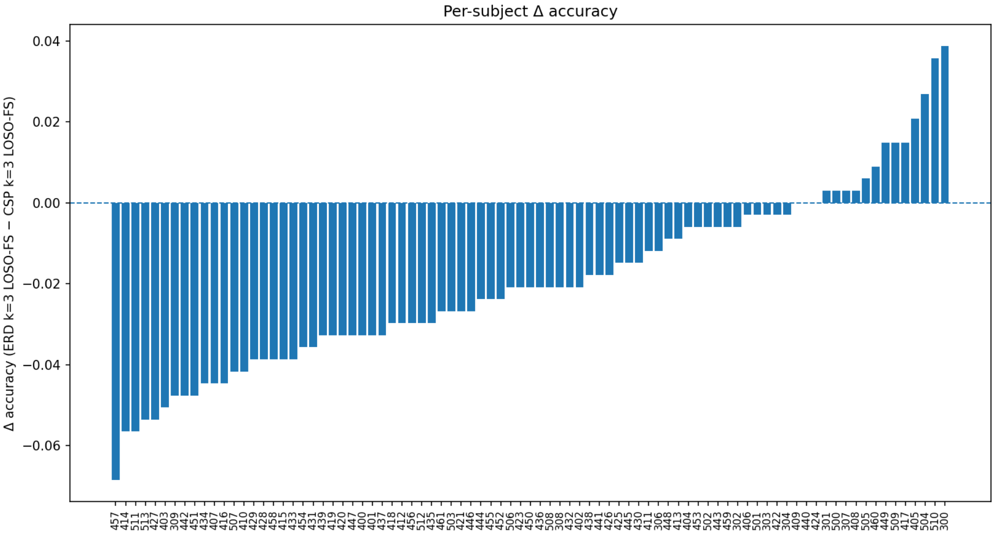


Figure B.4: PER-SUBJECT Δ ACCURACY IN THE CLUSTERED FEW-SHOT REGIME $(k = 3)$. Difference in decoding accuracy between ERD/ERS and CSP representations (Δ = ERD − CSP). Negative values favor CSP. CSP outperforms ERD/ERS for 70 of 85 participants (3 ties), with Δ ranging from −6.85 to +3.87 percentage points.

B.2 Extended Results for Subject-Level Representations for Population Clustering

This subsection provides supplementary results for subsection 4.2, evaluating the effects of four signal-space augmentations (Gaussian noise, time warping, frequency shift, mixup) on decoding accuracy. Tables report subject-wise performance changes under both pooled and clustered few-shot regimes, with significance tests and effect sizes. These analyses support the main finding that naive augmentations fail to improve and can even impair MI decoding performance.

Table B.2: Pooled cross-subject model (k=1): effect of each augmentation on decoding accuracy. Each augmentation was applied individually during multi-task training on pooled data and compared to a non-augmented baseline across N=85 subjects. Δ denotes the accuracy difference (augmented − baseline), reported as mean and median. p-values are Holm-adjusted from Wilcoxon signed-rank tests. Effect size is Cliff's δ. Only frequency shift shows a statistically significant degradation.

Augmentation	$\bar{\Delta}$	$\tilde{\Delta}$	**W**	p_{Holm}	**Effect Size**	**Sig.**
Gaussian noise	−0.0067	−0.0069	1379	0.185	$\delta = -0.129$	×
Time warp	−0.0017	−0.0069	1597	0.629	$\delta = -0.047$	×
Frequency shift	−0.0132	−0.0139	1210.5	0.0156	$\delta = -0.224$	✓
Mixup	−0.0022	−0.0069	1641	0.643	$\delta = -0.082$	×

Table B.3: Clustered few-shot adaptation (k=3): impact of time warping on CSP-based decoding. Subjects were clustered into k=3 groups; each test subject was given three labeled trials per class. Time warp (TW) was applied to the cluster-specific training data. All performance metrics show statistically significant degradation under augmentation. p-values are Holm-adjusted from Wilcoxon signed-rank tests; r is the rank-biserial effect size.

Metric	**Baseline**	**with TW**	Δ	$p_{\textbf{Holm}}$	**Effect size** (r)
Accuracy	0.6551	0.6321	−0.0230	5.10×10^{-10}	0.799
F1-score	0.6068	0.5697	−0.0372	3.08×10^{-12}	0.899
κ	0.3100	0.2640	−0.0470	3.41×10^{-10}	0.796
Precision	0.6460	0.6190	−0.0270	1.74×10^{-7}	0.719
Recall	0.6560	0.6330	−0.0240	4.04×10^{-10}	0.802

B.3 Extended Results for Cross-Subject Transfer on the Source Dataset

B.3.1 Extended Results for Few-Shot Personalization with Clustered and Pooled Models

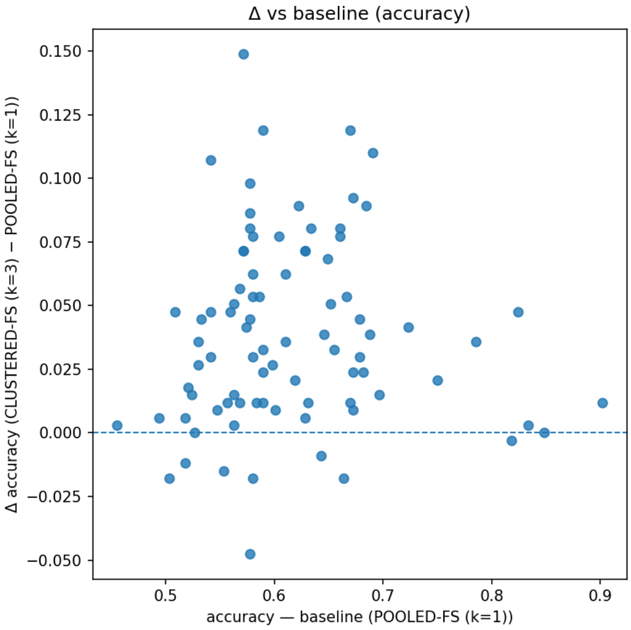


Figure B.5: SUBJECT-WISE CLUSTERING BENEFIT AS A FUNCTION OF POOLED BASELINE ACCURACY. Under the LOSO few-shot regime ($k_{\text{shot}} = 4$), each point represents a participant. The x-axis shows the subject's baseline (pooled) accuracy; the y-axis shows the performance gain from clustering (Δ = clustered − pooled). The majority of subjects show positive Δ across the baseline range, indicating that clustering benefits both low- and high-performing participants.

Table B.4: Aggregated comparison of clustered vs. pooled few-shot models across multiple metrics. Under the LOSO few-shot regime ($k_{\text{shot}} = 4$, $N = 85$), this table reports mean performance for pooled and clustered models across five metrics. Δ is the mean paired difference (clustered − pooled). Confidence intervals are 95% bootstrap CIs. Wilcoxon signed-rank tests assess statistical significance; r is the rank-biserial effect size. *Abbreviations:* Pooled = pooled mean, Clust. = clustered mean, I/D/T = improved / decreased / tied subjects.

Metric	Pooled	Clust.	$\overline{\Delta}$	95% CI	W	p	r	I/D/T
Accuracy	0.616	0.655	+0.039	[0.031, 0.047]	175.0	1.08×10^{-12}	0.900	75/8/2
Cohen's κ	0.233	0.310	+0.077	[0.062, 0.092]	175.5	7.07×10^{-13}	0.902	74/10/1
Precision	0.599	0.646	+0.047	[0.036, 0.057]	306.0	2.61×10^{-11}	0.833	70/15/0
Recall	0.617	0.656	+0.039	[0.031, 0.047]	175.0	6.96×10^{-13}	0.902	73/11/1
F1-score	0.549	0.607	+0.058	[0.048, 0.067]	98.0	3.50×10^{-14}	0.946	77/8/0

B.3.2 Extended Results for Effect of Clustering on Zero-Shot Transfer

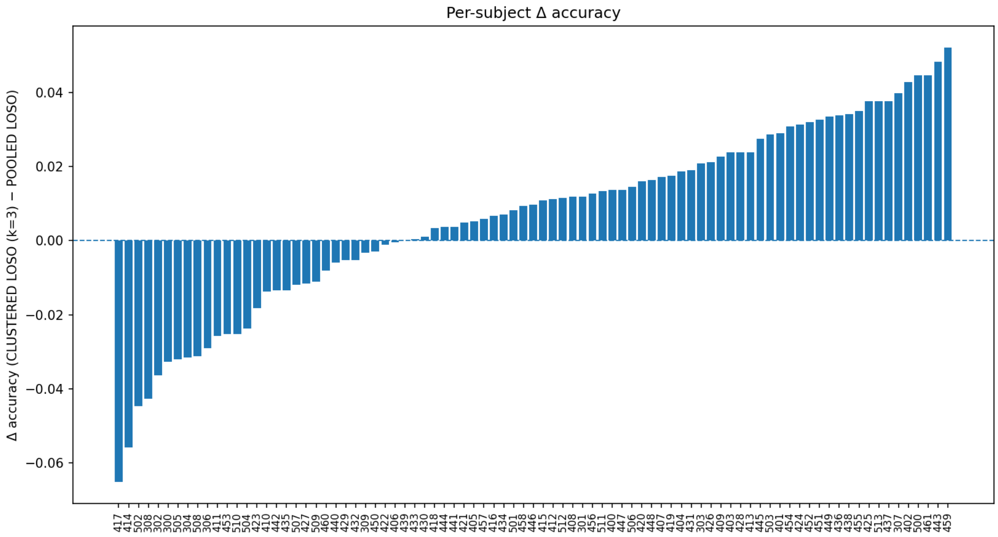


Figure B.6: Ordered per-subject clustering benefit in zero-shot transfer. Bars show the subject-wise accuracy difference Δ = clustered − pooled under LOSO zero-shot training ($k = 3$). While most subjects benefit from clustering, the effect is modest and heterogeneous, ranging from −6.5 to +5.2 percentage points.

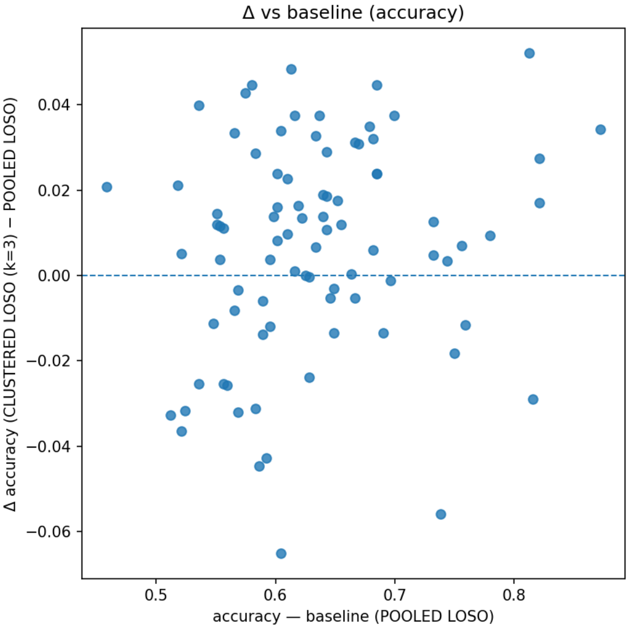


Figure B.7: Clustering benefit as a function of baseline performance in zero-shot transfer. Each point represents a subject under LOSO zero-shot evaluation. The x-axis is pooled model accuracy; the y-axis is the clustering uplift (Δ = clustered − pooled). Gains from clustering are observed across a wide baseline range.

Table B.5: Aggregated comparison of pooled vs. clustered zero-shot models across multiple metrics. Under LOSO zero-shot evaluation (k=3 clusters, N=85), this table reports mean performance, paired differences (Δ), bootstrap 95% confidence intervals, Wilcoxon test statistics, effect sizes (r), and subject counts. Clustering improves most metrics, though effect sizes are moderate. *Abbreviations:* Pooled = pooled mean, Clust. = clustered mean, I/D/T = improved / declined / tied.

Metric	Pooled	Clust.	$\overline{\Delta}$	95% CI	W	p	r	I/D/T
Accuracy	0.633	0.640	+**0.0064**	[0.0011, 0.0116]	1200	0.0091	0.328	55/29/1
Cohen's κ	0.267	0.281	+**0.0140**	[0.0045, 0.0232]	1093	0.00129	0.402	60/25/0
Precision	0.622	0.649	+**0.0269**	[0.0172, 0.0362]	668	3.76×10^{-7}	0.634	65/20/0
Recall	0.634	0.641	+**0.0069**	[0.0020, 0.0116]	1123	0.00202	0.385	59/26/0
F1-score score	0.570	0.582	+**0.0125**	[0.0062, 0.0186]	954	0.000129	0.478	63/22/0

B.3.3 Extended Results for Cluster-Conditioned Support in Few-Shot Adaptation

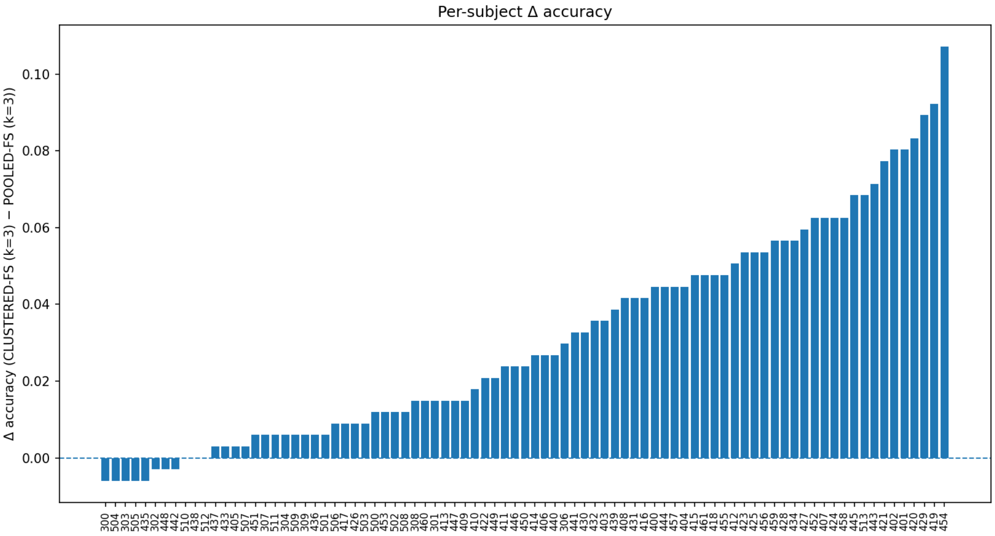


Figure B.8: Subject-wise improvement from cluster-conditioned support. Paired accuracy differences (Δ) under LOSO few-shot adaptation ($k = 3$), sorted by subject. Most participants exhibit positive gains when support examples are drawn from their assigned cluster.

Table B.6: Aggregate results for few-shot adaptation using cluster-conditioned vs. pooled support ($N = 85$). All models were trained using $k = 3$ clusters and $k_{\text{shot}} = 4$. Δ indicates the mean subject-wise improvement (clustered − pooled, in percentage points). Large effect sizes and highly significant p-values confirm the robustness of cluster conditioning across multiple evaluation metrics. *Abbreviations:* Pooled = pooled mean, Clust. = clustered mean, I/D/T = improved / declined / tied.

Metric	Pooled	Clust.	$\overline{\Delta}$ (pp)	95% CI (pp)	W	p	r	I/D/T
Accuracy	0.624	0.655	+3.07	[2.49, 3.67]	88.0	8.6×10^{-14}	0.948	74/8/3
Cohen's κ	0.248	0.310	+6.16	[5.09, 7.25]	91.0	4.2×10^{-14}	0.949	76/8/1
Precision	0.595	0.646	+5.09	[4.27, 5.91]	124.0	1.3×10^{-13}	0.931	75/9/1
Recall	0.625	0.656	+3.12	[2.57, 3.67]	84.5	3.4×10^{-14}	0.953	76/8/1
F1 score	0.563	0.607	+4.43	[3.76, 5.09]	40.0	7.1×10^{-15}	0.978	79/5/1

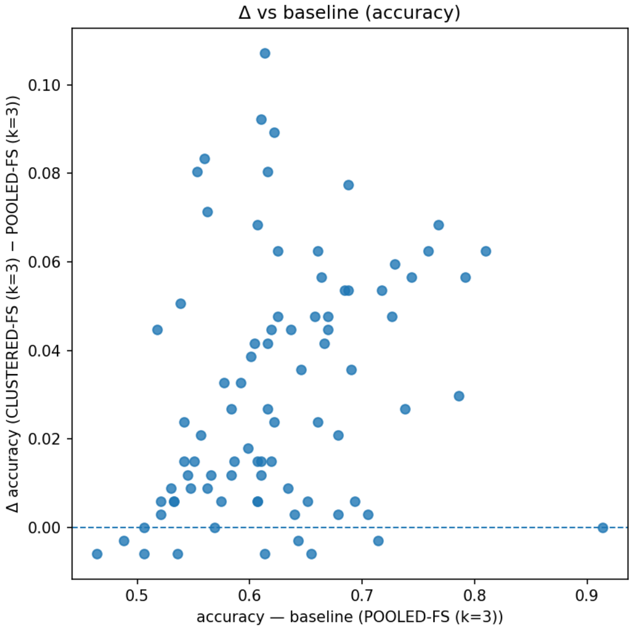


Figure B.9: Clustering benefit as a function of baseline performance (few-shot, $k = 3$). Each point denotes one subject. X-axis: accuracy under pooled support; Y-axis: gain from switching to cluster-conditioned support. The lack of strong correlation indicates that benefits are not restricted to low-performing subjects.

B.3.4 Extended Results for Effect of Calibration Trials

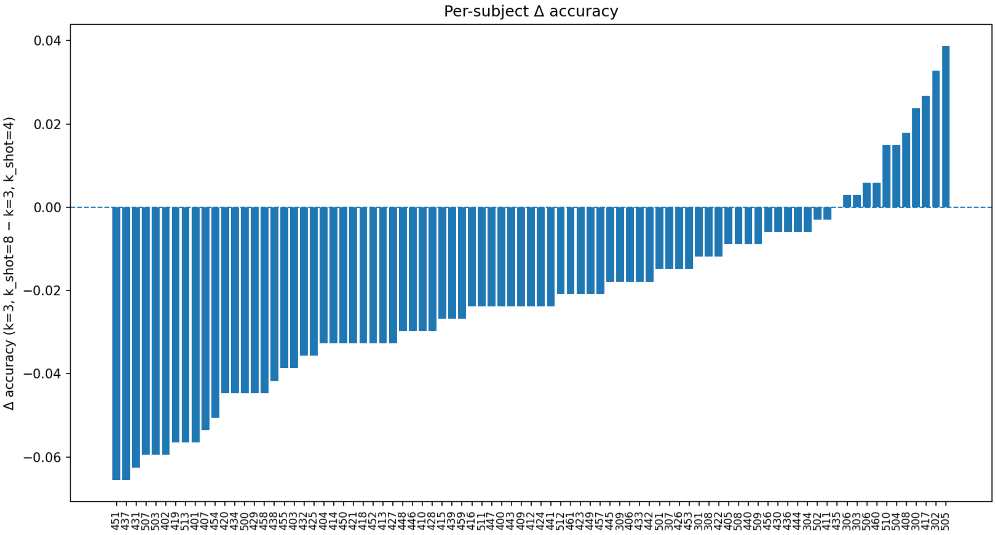


Figure B.10: Subject-wise impact of increasing support set size from 4 to 8 trials per class. Bars show the accuracy difference ($\Delta = \text{Acc}_{k=8} - \text{Acc}_{k=4}$) under LOSO few-shot adaptation. Values are sorted in descending order. The left-skewed distribution confirms that most participants experience performance degradation when given more calibration data.

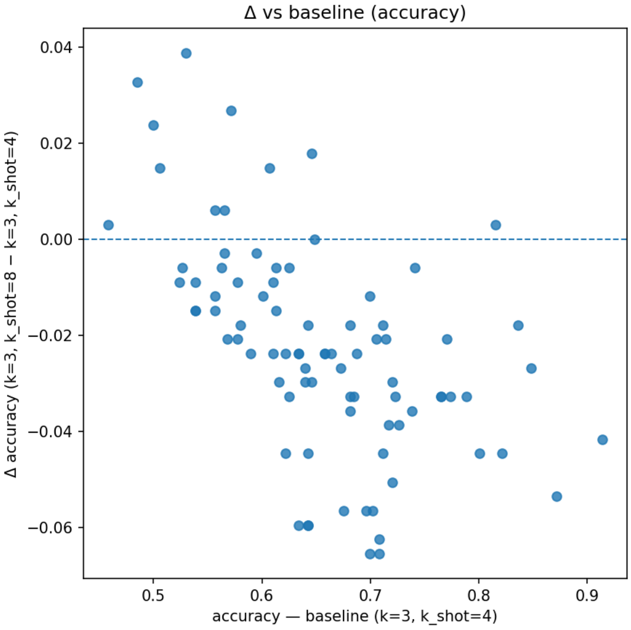
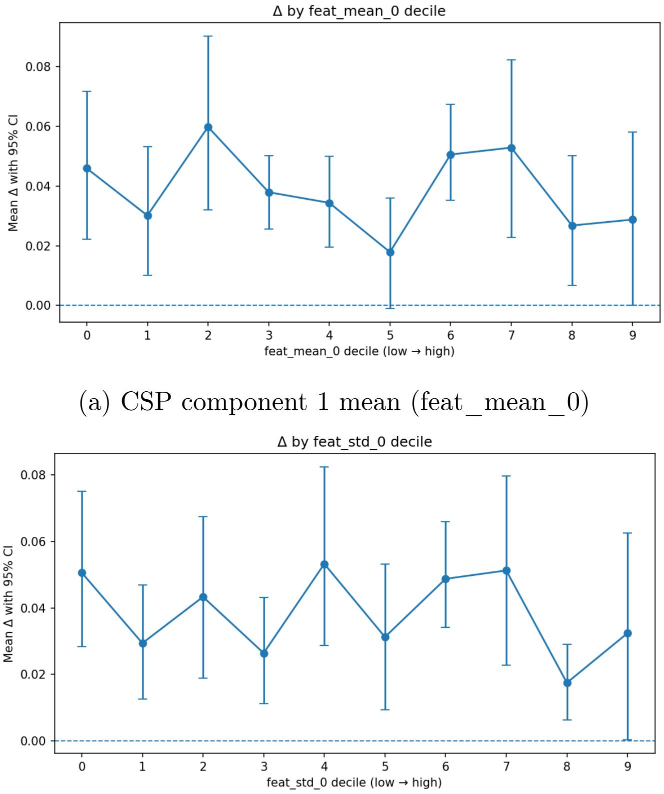


Figure B.11: Performance change due to increased support size vs. baseline performance. Each point represents a subject. X-axis: accuracy at $k = 4$; Y-axis: change in accuracy when increasing to $k = 8$. The negative slope indicates that subjects who initially performed well are disproportionately harmed by additional calibration data.

B.3.5 Extended Results for Feature-Centric Analysis of Clustering Benefit

Table B.7: Clustering benefit (Δ) by deciles of CSP feature statistics in the few-shot setting. Left: standard deviation of CSP component 2 (`feat_std_1`), which captures dominant class-discriminative variance. Right: mean of CSP component 1 (`feat_mean_0`), reflecting average projection magnitude across trials. Values represent mean $\Delta = \text{accuracy}_{\text{clustered}} - \text{accuracy}_{\text{pooled}}$ with 95% bootstrap confidence intervals.$\Delta = \text{accuracy}_{\text{clustered}} - \text{accuracy}_{\text{pooled}}$ with 95% bootstrap CI.

	`feat_std_1` (variability)		`feat_mean_0` (mean)	
Decile	Δ [95% CI]	*n*	Δ [95% CI]	*n*
0	0.011 [−0.007, 0.030]	9	0.031 [0.010, 0.056]	9
1	0.025 [0.004, 0.049]	8	0.053 [0.029, 0.074]	8
2	0.035 [0.012, 0.060]	9	0.062 [0.039, 0.089]	9
3	0.042 [0.017, 0.067]	9	0.035 [0.014, 0.059]	9
4	0.047 [0.021, 0.070]	9	0.030 [0.008, 0.055]	8
5	0.044 [0.018, 0.069]	8	0.037 [0.014, 0.064]	8
6	0.053 [0.029, 0.076]	9	0.041 [0.018, 0.068]	8
7	0.057 [0.035, 0.080]	8	0.059 [0.033, 0.082]	7
8	0.061 [0.038, 0.085]	8	0.034 [0.012, 0.059]	9
9	0.067 [0.044, 0.092]	7	0.021 [0.003, 0.043]	9



(a) CSP component 1 mean (feat_mean_0)

(b) CSP component 1 variability (feat_std_0)

Figure B.12: Supplementary decile overlays for CSP features. Each bar shows the mean clustering benefit Δ across subjects grouped into deciles of a CSP-derived feature, with 95% bootstrap confidence intervals. (a) Mean of CSP component 1 shows two benefit peaks in mid ranges. (b) Variability of CSP component 1 shows no consistent trend and weak standalone predictive value.

Table B.8: Full spearman correlation analysis of clustering benefit Δ with subject-level predictors. Zero-shot and few-shot settings are reported side by side, with rank correlations (r_s), exact p-values, and Benjamini–Hochberg FDR-adjusted q-values ($q < .10$ = significant). No predictor achieves significance after FDR correction. Baseline accuracy (`accuracy_A`) is included as a pseudo-predictor.

	Zero-shot			Few-shot		
Predictor	r_s	p	q	r_s	p	q
accuracy	0.190	0.081	0.405	0.123	0.261	0.652
beta_power	0.121	0.271	0.677	-0.056	0.612	0.764
mu_power	0.082	0.454	0.757	-0.105	0.339	0.678
feat_std_1	0.046	0.675	0.844	0.112	0.306	0.663
mu_erd	0.061	0.581	0.814	0.068	0.537	0.764
feat_std_0	-0.077	0.486	0.757	-0.082	0.457	0.739
feat_mean_0	-0.061	0.578	0.814	-0.091	0.409	0.722
feat_mean_1	-0.029	0.790	0.877	0.026	0.811	0.811
beta_erd	0.034	0.761	0.877	-0.008	0.939	0.939
spec_entropy	0.004	0.972	0.972	-0.042	0.701	0.811