# BCIs for Everyone, Everyday: Generalized Machine Learning Models for Decoding of Human Brain Data

Zoe Steine-Hanson

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington
2024

*Reading Committee:*
Rajesh Rao, Co-Chair
Bing Brunton, Co-Chair
Matt Golub

Program Authorized to Offer Degree:
Computer Science and Engineering

University of Washington

**Abstract**

BCIs for Everyone, Everyday: Generalized Machine Learning Models for Decoding of Human Brain Data

Zoe Steine-Hanson

Co-chairs of the Supervisory Committee:

Professor Rajesh Rao
Computer Science and Engineering

Professor Bing Brunton
Biology

Brain Computer Interfaces (BCIs) offer immense potential to enhance the quality of life for individuals worldwide, spanning applications in prosthetics, mental health, and virtual reality. Recent advancements in deep learning techniques and the exponential growth of available data have significantly improved BCI research and neural decoding over the last decade. However, there remains a critical need to extend the applicability of these systems to everyone, everyday. While much of the current BCI research has focused on developing neural decoders for individuals performing specific experimental tasks, the true potential of BCIs lies in their real-world applicability. To achieve broad BCI use, we must enhance key aspects of BCI systems, such as their ability to generalize across users, be robust to natural behaviors, and provide inclusive user experiences. This thesis addresses these challenges by exploring various components of BCI systems to improve their real-world applicability. Chapter 3 introduces HTNet, a deep learning neural decoder for ECoG data that is robust to naturalistic movement behaviors in humans, and individual variability. Building on this work, Chapter 4 extends the HTNet model to cognitive behaviors in EEG, highlighting its task-relevant robustness. Chapter 5 explores the structure of naturalistic human neural data by extracting estimates of neural manifolds and describing their geometry. We find that naturalistic neural activity resides within similarly aligned low-dimensional neural manifolds across movement types, days and individuals, po-

tentially indicating a shared manifold for natural behavior. Chapter 6 explores various strategies to enhance the robustness of neural decoders to naturalistic human neural data. These strategies include decoding complex movement information, leveraging unlabeled naturalistic neural data through self-supervised learning, and decoding from neural manifolds. While the majority of this thesis focuses on machine learning improvements for BCI systems, Chapter 7 explores another aspect of BCI systems – the user experience. This chapter investigates gender gaps in BCI performance and uses human-computer interaction techniques to understand how aspects of the user interface influence BCI performance. In the concluding chapter (Chapter 8), we propose suggestions for future research to further develop innovative neural decoders that are not only robust to naturalistic behaviors and individual variability, but also designed with everyone in mind.

# Acknowledgements

Thank you to the many people who have supported me throughout this journey. Thank you to my advisors, Prof. Rajesh Rao and Prof. Bing Brunton, who both gave me great guidance, feedback and advice throughout the many years. I especially appreciate their feedback on my NSF fellowship proposal, which helped me to win the award and gain research freedom. I also appreciate the encouragement they gave me to pursue internships in industry, which ultimately gave me the opportunity to immerse myself in potential future commercial applications of BCIs. Thank you to my committee members, Matt Golub and Amy Orsborn, for thoughtful questions and feedback on the aims and structure of this thesis. Thank you to my many lab-mates between both the Neural Systems lab and the Brunton lab – Preston Jiang, Ares Fisher, Samantha Sun, Courtnie Paschall, Dimitrios Gklezakos, Gabrielle Strandquist, Matthew Bryan, Mozes Jacobs, Prashant Rangarajan, Raphael Bechtold, Vishwas Sathish, Steven Peterson, Satpreet Singh, Elliott Abe, Ali Weber, Harsha Gurnani, Lawrence Hu, Maryam Bahadori, Sophie Balint, Biraj Pandey, Lili Karashchuk, Michelle Hickner, Raveena Chhibber, Zeynep Toprakbasti and many more. Thank you especially to Preston for encouraging me to apply to the PhD program in UW CSE, and for encouraging me to stay and see this through to the end. Without Preston, I would not have received a PhD in computer science. Thank you to Ellie for being my closest colleague and friend throughout the process, and thank you for giving me the motivation and the courage to write the chapter on gender performance gaps in BCIs. Thank you to my pre-PhD mentors, Prof. Margaret Burnett and Prof. Andrea Stocco, for encouraging me to do science, cultivating my excitement, and supporting me throughout the grad school application process. Thank you to Steven Peterson, Satpreet Singh and Nancy Wang for all the work they put into the naturalistic ECoG dataset that I explore throughout this thesis. They walked with this data so that I could run with it and make many interesting discoveries about the naturalistic brain. Thank you also to the staff at Harborview Medical

# DEDICATION

To Juno – The best emotional support animal

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Brain Computer Interface (BCI) research has led to significant progress in a multitude of applications, including letting paralyzed patients control robotic arms [Grigorescu et al.], improving sleep quality [Chen et al.], and mitigating the effects of major depressive disorders [Delaloye and Holtzheimer]. The commercial realm is also beginning to embrace BCIs, with products ranging from electroencephalography (EEG) headphones that monitor focus [Neurable], to headsets tracking mental health [BrainCo], and electromyography (EMG) wristbands for virtual reality interactions [Meta; CTRL-labs at Reality Labs et al.].

BCIs work by capturing neural signals during specific tasks, which are then interpreted by machine learning models. In some systems, known as offline BCIs, these signals are processed later without providing real-time feedback to the user. Neural signals can be collected via non-invasive methods like EEG, EMG, and fMRI, or through invasive techniques such as electrocorticography (ECoG) and microelectrode arrays. Once processed, these signals are decoded by machine learning algorithms tailored to the task of interest. For example, in motor imagery BCIs, models classify imagined movements, which can then be mirrored in a virtual environment in online systems.

Despite recent advancements, several challenges hinder the widespread adoption of BCIs. First, current BCIs rely on models fine-tuned to training data from every individual [Benabid et al.]; this makes it difficult to implement BCIs before substantial training data is collected for each person. Second, most existing BCI research is confined to the laboratory, where behavior is constrained to the research task and thus does not accurately represent the diversity of neural signals. Third, the prevalence of offline and in-lab studies

raises questions about designing user interfaces for BCI systems that enhance user performance during everyday use. Efforts to address these challenges include developing neural decoders that generalize across different individuals, and adapting them to decode more naturalistic neural data, with some work already showing promise in these areas [Safaie et al.; Volker et al.; Wang et al., c; Melbaum et al.]. Explorations into user interfaces for BCI have also begun to improve user engagement [Lotte and Jeunet]. However, more translational work still needs to be done to take advances from the lab to BCIs that work for everyone, everyday.

Therefore, the goal of this thesis is to advance the development of BCIs that work for everyone, everyday. To achieve this, this thesis works on improving real-world applicability of BCI systems in three dimensions. First, enhancing the generalizability of models across participants. Second, decoding behavior in more contexts, including naturalistic behaviors. And third, understanding user interactions with BCIs through various problem-solving styles. We primarily use a unique ECoG dataset collected from individuals during regular activities in a hospital setting, offering a rich source for developing robust decoding algorithms and gaining insights into naturalistic neural activity [Wang et al., f, 2018; Peterson et al., 2022]. I also leverage my background in human-computer interaction to understand user experiences in BCI systems.

The thesis is structured as follows:

- Chapter 2 covers related work, including background on generalizability in machine learning, previous efforts on decoding naturalistic neural data, and the exploration of neural manifolds.

- Chapter 3 introduces HTNet, a neural decoder tailored for naturalistic ECoG data, demonstrating its efficacy in decoding movement and its ability to generalize across individuals.

- Chapter 4 evaluates the adaptability of HTNet to cognitive decoding in EEG, assessing its robustness in different contexts.

- Chapter 5 explores neural manifolds in the naturalistic ECoG dataset to uncover deeper insights into the nature of natural neural data.

- Chapter 6 discusses various strategies we attempted to enhance decoder performance, including self-supervised learning and manifold alignment techniques.

- Chapter 7 explores user interfaces of BCI systems and investigates how differences in problem-solving styles might explain gender disparities in BCI performance.

- Chapter 8 outlines future research directions and summarizes the key findings of this thesis.

Through this research, we aim to advance the field of BCIs by developing more adaptable, effective, and user-friendly systems, ultimately making these technologies accessible for everyone, everyday.

# Chapter 2

# Related Work

## 2.1 Generalizability in Machine Learning

The deep learning (DL) revolution over the past decade has paved the way for numerous novel applications of ML. However, the effectiveness of ML heavily relies on the assumption that the training data and the test data will be Independent and Identically Distributed (*i.i.d.*) [Hoadley]. Regrettably, this *i.i.d.* assumption often results in a lack of generalizability for many ML models when confronted with natural shifts in data. These natural shifts, or out-of-distribution shifts, occur when the model was trained on a specific data distribution, but test data may exhibit a different distribution [Koh et al., 2021]. Real-world datasets frequently violate the *i.i.d.* assumption, especially in the case of neural recordings, where changes in the recorded neurons can occur due to factors like electrode drift, scar tissue growth, or other variables, on a day-to-day basis [Rule et al.]. Achieving robust model generalization is a critical milestone for unlocking a multitude of ML applications, including automating medical screenings, self-driving cars, and robotics.

Despite the constraint provided by the *i.i.d.* assumption in ML, there are many techniques out there that can help with model generalizability, which I highlight a few here. Techniques such as regularization, transfer learning (TL), meta-learning, domain adaptation and data augmentation all can help models avoid over-fitting and improve generalization [Tan et al., b; Liu et al., a]. Regularization techniques are one of the simplest ways to encourage generalization in a model, as they can be easily added to any model training setup. Examples of regularization methods include early stopping [Yao et al.] and dropout [Srivastava et al.,

2014]. TL is another common approach for encouraging generalization, and often works by pre-training a model on a related task, such as image classification in ImageNet, and then using the weights from the pre-trained task to initialize training on a new task, such as classifying medical images [Neyshabur et al., 2020]. TL can help to decrease the amount of data needed from the target domain's data by leveraging information from a source domain to help the model learn the target distribution [Tan et al., b]. Meta-learning, also known as 'learning to learn', is an approach in which the model learns meta-information over multiple learning episodes [Hospedales et al.]. One such approach, Model Soups, was able to further improve ImageNet performance by averaging the weights of multiple pre-trained computer vision models with different initial conditions [Wortsman et al.]. Domain adaptation is another approach for model generalization, which works by mapping domains to each other [Farahani et al.]. This can be done either by mapping the domains directly into each other, or by mapping the domains into some kind of 'universal' domain. Domain adaption includes techniques like CORrelation ALignment (CORAL), which constrains the model to learn similar latent distributions between domains and can be surprisingly easy to implement [Sun et al., a]. The last approach I will mention is data augmentation, though many other approaches exist for improving model generalization. Data augmentation generally works by creating new data from an existing dataset through a series of transformations on the original data [Wen et al.]. These approaches can help with generalization by widening the distribution of the training data. Overall, all of these approaches and techniques can work around the *i.i.d.* assumption and help to improve model generalization in a variety of tasks.

## 2.2   Brain Computer Interfaces

BCIs attempt to decode information from neural signals that can then be used to drive a computer or machine; for example, using electroencephalography (EEG) signals to predict imagined movements [Wolpaw and Wolpaw]. The concept of BCIs has existed for quite some time, with the earliest mention of the term happening in the 1970's [Vidal]. Early BCIs often used linear methods to decode information from neural signals with techniques such as Kalman filters [Omidvarnia et al.; Black et al.; Wu and Hatsopoulos; Gilja et al., b], Wiener filters [Erdoğan, 2009], linear regression [McFarland and Wolpaw], and Support Vector Machines for classification [Rakotomamonjy and Guigue]. These neural decoders have been used on various tasks, such as cursor control for computers [Trejo et al.], robotic arms [Flesher et al.], text input [Farwell and

Donchin], and prosthetics [Murguialday et al.]. Kalman filter techniques have even been used for adaptive BCIs which can adjust the decoder to the user's neural signals [Orsborn et al.].

As DL methods have become more common, useful, and viable, they have also been used in BCIs. Common DL architectures that have been used for BCIs include Convolutional Neural Networks (CNNs) [Peterson et al., b; Li et al., 2017; Lawhern et al.; Frey et al.; Golshan et al.; Wang et al., b; Barger et al.; Supratak et al.; Ahmadi et al.; Schirrmeister et al.] and Long Short-Term Memory (LSTMs) [Wang et al., 2018; Ahmadi et al.; Wang et al., g; Glaser et al.; Xie et al.; Tseng et al.; Naufel et al.; Park and Kim; Du et al.; Pan et al.; Elango et al.]. These methods have shown a variety of improvements on BCI tasks, as these models can handle non-linearities present in the neural data that traditional techniques cannot. The move to DL in BCIs has also opened up the door for new BCI tasks that previously would be impossible, such as reconstructing faces from fMRI data [Dado et al.], decoding handwritten characters from EEG [Pei and Ouyang], or decoding movement in naturalistic electrocorticography (ECoG) data [Peterson et al., b]. While DL methods have ushered in new possibilities in BCIs, they are notoriously data-hungry [Lee], creating a predicament as neural data can be both costly and time-consuming to collect. For example, in [Tang et al.] researchers needed 16 hours of fMRI data per participant to build a model based off large language models that could decode sentences from imagined speech. In order for BCIs to work in a variety of contexts and participants, researchers will need to find ways to train decoders with less domain-specific or participant-specific data.

Some work has already started in the direction of building generalized neural decoders for BCIs that are less data hungry than some DL methods. TL has been leveraged in various neural decoding papers to help reduce the amount of data needed in a target domain [Tan et al., b,a; Zubarev et al.; Wu et al.; Volker et al.; Azab et al.; Makin et al.; Wang et al., e; Thomas et al.; Lee et al., a; Elango et al.; Desai et al.; Fahimi et al.]. For example, in Tan et al. [a], researchers pre-trained a network on the ImageNet dataset, and then transferred the model to classify what music the participant was listening to from their EEG data. Self-Supervised Learning (SSL) has also shown to be a promising approach for BCI generalization and data scarcity, and is discussed further in Section 2.5.

## 2.3 Naturalistic Neural Data

Many of the BCI applications and experiments in the previous section involve decoding activity of experimentally-driven behaviors, such as monkeys performing cued center-out cursor tasks. However, if we want to bring BCIs into the real world for commercial and therapeutic applications, they need to be able to interpret and handle naturalistic neural data. For example, a BCI for prosthetic control that has been trained with cued imagined movements may work well when the user is concentrating intently on the imagined movement, but oftentimes when we are moving in the real-world we are distracted by other stimuli, such as a conversation with a friend or reading a book. Luckily, there has been a growing interest in collecting data during natural-istic behaviors in the neuroscience community as the computational requirements needed have finally caught up [Mathis and Mathis; Kennedy; Anderson and Perona]. Collecting and analyzing naturalistic neural data will not only help to bring BCIs to the real-world, but also improve our understanding of how the brain con-trols complex natural behaviors. For instance, it can shed light on how internal states of an animal impacts behavior [Calhoun et al.; Coen et al.; Johnson et al.; Eyjolfsdottir et al.], and how behavior changes over the course of an animal's lifespan [Zhang et al., b]. Interestingly, another benefit of focusing on naturalistic behaviors is that animals may end up performing better on tasks with naturalistic setups [Rosenberg et al.].

With the increasing focus on naturalistic data, quite a lot of intriguing datasets have been collected in a variety of animal models. Many works have focused on freely moving rats, including Melbaum et al.; DiGiovanna et al.; O'Keefe, where naturalistic behavior is still constrained to a small arena, but rats are able to move about as desired in the space. Researchers have also collected naturalistic behavioral data of fly courtship [Calhoun et al.], zebrafish larvae swimming in a naturalistic environment [Johnson et al.], unconstrained mice running in a maze [Rosenberg et al.], freely flying bats [Ulanovsky and Moss; Yartsev and Ulanovsky], freely moving mice in a small arena [Wiltschko et al.], and freely moving monkeys grabbing treats [Altan et al.]. A good amount of data has even been collected in humans [Wang et al., 2018; Dastjerdi et al.; Singh et al., b; Peterson et al., 2022; Gabriel et al.], including naturalistic speech comprehension and production behaviors [Cai et al.].

This myriad of new naturalistic data has been driven by advances in computer vision algorithms, which can finally allow for the robust detection of human and animal behavioral poses from video data. Pose estimation has long been a computational challenge, as issues like visual occlusions and marker-less joint

tracking have posed significant issues. However, the revolution in DL has improved the computer vision techniques available for pose tracking, both in 2D and in 3D. From this, techniques such as DeeperCut [Insafutdinov et al.], DeepLabCut [Mathis et al.], Anipose [Karashchuk et al.], OpenPose [Cao et al., c], and OmniMotion [Wang et al., d] have emerged for behavioral tracking in both humans and animals. These techniques typically use Deep CNNs to learn how to extract pose information from video frames. For example, a popular method, DeepLabCut, makes use of a ResNet architecture to extract 2D or 3D pose estimations and can transfer the model to new video types with as few as 50 video frames for re-training [Mathis et al.].

As more naturalistic neural data and behavioral tracking emerges, opportunities to build neural decoders for naturalistic data have also emerged. For example, in Melbaum et al., researchers were able to decode whether freely moving rats were turning around, rearing or at rest from their neural data. In another work, researchers were able to decode the EMG arm muscle activations of freely moving monkeys in a cage from their neural activity [Altan et al.]. There have also been several examples of naturalistic neural decoding in humans, indicating the potential for real-world use of BCIs. In Alasfour et al., researchers were able to decode which of four naturalistic behaviors (talking, using electronics, watching TV or resting) human ECoG patients were engaged in. In a series of works from Peterson et al. [b,a], researchers were able to decode moving vs rest states in human epilepsy patients using either a TL approach [Peterson et al., b] (Chapter 3) or an SSL approach [Peterson et al., a]. All together, these results highlight the potential for neural decoding of naturalistic behaviors, but also shows how more work will be needed to expand naturalistic neural decoding to more than a handful of behaviors.

## 2.4   Neural Manifolds

The recent emergence of large-scale neural recordings via improvements in microelectrode arrays has enabled researchers to dig even deeper into the computations in the brain that drive behavior. These large-scale recordings can capture signal from thousands of neurons at once. This allows researchers to develop theories and models about the various neural populations involved in behavior within and across brain regions, rather than just from a small set of neurons. These population level recordings have opened a new line of research that investigates *neural manifolds*, the low-dimensional neural activity within the high-dimensional

neural recordings. Neural activity of neural populations can span as many dimensions as there are neurons, neural manifolds are thus low-dimensional sub-regions of neural space where neural activity is constrained during behavior. Neural manifolds help to both define which sub-populations of neural activity are most related to the behavior of interest, and to uncover the neural latent dynamics that drive behavior within the sub-populations [Chung and Abbott; Gallego et al., b].

Researchers have analyzed neural manifolds in various contexts, including movement, learning, and vision [Chung and Abbott; Gallego et al., b]. Many of the studies involving movement behaviors have focused on experiment-driven movements in micro-array recordings of non-human primates, in order to uncover the low-dimensional neural manifolds that capture a significant fraction of the variability in neural activity and their relation to movement control [Abbaspourazad et al.; Bruno et al.; Churchland et al., b,c,a; Churchland and Shenoy; Elsayed et al.; Gallego et al., c; Kaufman et al., a,b; Michaels et al.; Overduin et al.; Sadtler et al.; Santhanam et al.; Sussillo et al., a; Sun et al., b]. For example, Churchland et al. [b] analyzed the neural manifolds of monkeys doing a center-out movement task, which helped them uncover how neural activity during the preparatory phase could avoid generating movement, and instead would initialize the biological network to eventually generate movement [Churchland et al., a; Kaufman et al., a; Shenoy et al.]. Manifold analysis has also been important for understanding motor learning. Sadtler et al. found that motor learning with a BCI that used activity from the"intrinsic" manifold for cursor control was more easily learned than with a BCI that used neural activity outside the intrinsic manifold; this work was further extended in [Golub et al.]. Researchers have also found evidence of low-dimensional neural manifolds in brain regions other than motor cortex, such as visual cortex [Tsodyks et al.; Cowley et al.]. In their study, Cowley et al. found that the behavioral state of the monkeys influenced which specific regions the neural activity resided on in the neural manifold. While most of the previous work in neural manifolds focused on neural recordings from non-human primates and other animal models, some work has also shown the existence of neural manifolds in human neural recordings [Natraj et al., a; Rizzoglio et al.; Natraj et al., b]. For example, Natraj et al. [b] analyzed neural manifolds of micro-ECoG data from humans performing various hand gestures. They found that (1) the neural activity related to movement did occupy low-dimensional neural manifolds and (2) the neural dynamics within the manifold were constrained based on the movement activity.

Work in neural manifolds has uncovered interesting information about how the brain controls behavior,

but the choice of how to estimate the manifold is not trivial. In order to estimate the neural manifold, the high-dimensional neural data must be reduced to a lower dimension, where latent variables make up the sub-region of the manifold. Broadly, there are two classes of methods for dimensionality reduction and estimating the manifold, linear methods and non-linear methods. Linear methods include techniques like Principal Components Analysis (PCA) and Factor Analysis (FA). Non-linear methods include techniques like Locally Linear Embedding (LLE), Laplacian Eigenmaps (LEMs) and AutoEncoders (AEs). PCA is the most commonly used technique, as it guarantees a solution and results in a highly interpretable low-dimensional embedding [Gallego et al., a; Churchland et al., b; Gallego et al., c; Natraj et al., b; Ahrens et al.; Safaie et al.]. However, the true neural manifold may actually occupy non-linear sub-regions of the neural space, in which case non-linear dimensionality reduction techniques may be more appropriate to estimate the manifold. In Altan et al., the authors first used the intrinsic and embedding dimensionalities of the neural manifold to determine the non-linearity of the neural data they collected from unconstrained behaviors in two monkeys, which helped guide them to the right embedding technique for the data. Even if the neural manifolds are somewhat non-linear, linear approaches may still be preferred over non-linear techniques due to their drawbacks. For example, AEs may not converge to a solution, resulting in a poor estimate of the manifold; LLE and LEMs require dense samples along the manifold to get a reasonable estimate, requiring more data; and non-linear methods also typically result in less interpretable embeddings.

Much of the existing work has established the presence and usefulness of neural manifolds for the control of experiment-driven movements. However, research investigating whether low-dimensional neural manifolds also persist during more complex natural movements is just starting to emerge [Melbaum et al.; Altan et al.; Abbaspourazad et al.]. To understand whether a universal manifold, i.e., one containing and constraining activity for all motor behaviors, is the basis for all motor control, researchers must explore neural manifolds during complex naturalistic movement behaviors [Gallego et al., b]. Abbaspourazad et al. found that one principal mode within the neural manifold of naturalistic monkey reaches was predictive of the movements. Altan et al. also found low-dimensional manifolds in monkeys performing unconstrained movements and that the activity from the manifold was just as useful for decoding as was all neural activity. Melbaum et al. found evidence of low-dimensional manifolds in rats while they were freely moving in their cage, which were ultimately also useful for decoding behavior. While this presents a good start to exploring

manifolds in naturalistic behaviors, more work should be done to uncover the computations within the manifold driving behavior, and to leverage that information for decoding.

From previous work, we know that low-dimensional neural manifolds correspond with various types of behaviors, especially movements. In many of these works, the activity on the low-dimensional neural manifolds seem to be directly related to the brains ability to control movement. Given that many BCI applications involve the generation of movement (e.g. prosthetics and computer cursor movement), it becomes reasonable to ask whether neural activity within the manifold can be useful for movement decoding. It is also worth asking whether information about the manifold may be useful for domain adaption. In fact, much work has already explored neural manifold alignment techniques to decode behavior and adapt decoders to new domains [Gallego et al., a; Farshchian et al.; Natraj et al., a; Bashford et al.; Pandarinath et al.]. These approaches have proved to work well in most cases, as they allow the decoder to ignore noise in individual neurons, and instead learn from latent neural activity, which is much more robust across days and individuals.

One of the main use cases for these neural manifold alignment techniques is for re-calibration of neural decoders across different recording sessions. As electrodes shift and signals from neurons fade and change over the course of long-term recordings, neural decoders often go through lengthy re-calibration steps for each new recording day [Gallego et al., a]. However, neural manifold alignment techniques can help neural decoders learn latent neural dynamics that will persist despite changes to the exact neurons being recorded from. For example, Gallego et al. [a] used Canonical Correlation Analysis (CCA) to align the latent dynamics within the manifold of a monkey on sessions spanning many days; they found that they could realign the decoder even 700 days after the initial recording session of one monkey. In another paper, Natraj et al. [a] used AEs for neural manifold alignment to also perform long-term decoder alignment, this time in a human participant with micro-ECoG recordings. The authors were able to realign the BCI for up to 7 months after the initial recordings, including in online decoding sessions. Farshchian et al. also showed similar results with AEs to align neural manifolds using adversarial techniques to adjust the network. With their approach the researchers were able to align the manifolds up to 16 days apart using as little as one minute of data from the new day.

Alignment of neural manifolds can extend beyond just the same participants on different days, but also

work between participants. For example, previous work in neural manifold alignment techniques has shown that neural manifold alignment can help align decoders across participants [Melbaum et al.; Safaie et al.] and even across species [Rizzoglio et al.]. In Safaie et al., researchers used CCA to align manifolds between different monkeys performing a center-out reaching task, and were able to create cross-participant decoders that performed almost as well as subject-specific decoders. In freely moving rats, Melbaum et al. was able to use neural manifold alignment, via a Procrustes Transformation, for cross-subject and cross-session generalization, indicating the usefulness of manifold alignment even for naturalistic movement decoding. In a different line of work, Rizzoglio et al. also used CCA on the latent dynamics from microelectrode recordings in monkeys to successfully align a decoder to the latent dynamics from microelectrode recordings in a human participant with tetraplegia.

Many of the previous examples utilized linear methods for their neural manifold alignment techniques, though non-linear cases exist as well. The most common technique is CCA, which works by maximizing the correlations between the two datasets to align them. CCA is a linear technique, and requires that the neural manifold data is temporally aligned, otherwise no linear transformation from CCA will find a good alignment. The Procrustes Transformation, another linear alignment technique, loosens this constraint from CCA, but does still require that at least some of the coordinate axes are unchanged between the manifold spaces. While CCA has worked well in many cases, extending neural manifold alignment techniques to naturalistic datasets requires loosening of constraints, and also may require non-linear transformations as the data grows more complex. In this case, non-linear methods may perform better. Methods such as LFADS [Sussillo et al., b] and ADAN (Adversarial Domain Adaptation Network) [Farshchian et al.] provide non-linear mappings between manifolds, and also generate non-linear manifold estimates. Both of these techniques utilize AE-like networks to find the right latent representations of the data for alignment. In the case of ADAN, the network uses an adversarial approach to align the manifolds of different sessions, such that a discriminator network tries to maximize the difference in the latent representations on different days, while an aligner network challenges the discriminator by aligning the data before sending it through the network.

For the most part, the techniques described above utilize a one to one mapping between different sessions or different participants. However, to create a truly generalized model, a universal manifold that spans

multiple sessions or participants would be more helpful. There are some techniques that are more flexible to this kind of approach, such as Generalized Procrustes Alignment (GPA) [Gower] and LFADS [Sussillo et al., b]. GPA is an extension of the Procrustes Transformation described before, but instead of mapping onto an arbitrarily selected manifold from the set of manifolds, the data is transformed onto an optimal universal shape for all manifolds. LFADS also allows for creating an estimate of the universal manifold by jointly training the AE network on multiple sessions at once.

## 2.5 Self-Supervised Learning

The previous background sections highlighted considerable advancements in machine learning models, primarily driven by supervised learning techniques [LeCun et al., 2015]. However, supervised learning faces significant hurdles, including the need for large amounts of labeled data. To address these challenges, self-supervised learning (SSL) has emerged as a promising alternative [Jing and Tian, 2019; Liu et al., 2020; Doersch et al.]. SSL, a variant of unsupervised learning, works by training a model using labels that can be automatically generated from the data, rather than needing someone to explicitly make the labels by hand. For example, in image data researchers can break an image into pieces and then train a model to predict where one piece of the image should be placed relative to an anchor piece, creating a relative position task [Doersch et al.]. This initial step, often referred to as the *pre-task*, not only facilitates the learning of data properties but also prepares the model for subsequent applications, known as the *downstream* task. Other similar SSL approaches in computer vision and other fields have used pre-tasks such as detecting shuffled video frames, and discovering cross-modal features, to enhance model performance on subsequent tasks [Misra et al., 2016; Jaiswal et al., 2020]. Another popular SSL method, Word2Vec, leverages unsupervised techniques to understand word contexts within sentences, significantly influencing various domains and inspiring a range of specialized "2Vec" models that enhance both supervised and unsupervised learning paradigms [Mikolov et al., 2013; Barkan and Koenigstein, 2016; Zou et al., 2019; Ng, 2017; Nalmpantis and Vrakas, 2019; Nalmpantis et al., 2018], including a specialized "brain2vec" [Lesaja et al.]. There are also contrastive SSL approaches which build up model representations by contrasting two pieces of data and guiding the model to learn which pieces have more similar representations. By leveraging features learned from pre-tasks to boost separate downstream tasks, these papers demonstrate the use of SSL for transfer

learning.

Much of SSL literature focuses on data-rich areas like vision or natural language processing. Unfortunately, much less work has applied SSL to biological time series data, including ECG, EEG and ECoG, even though these domains could benefit greatly from techniques that don't require labeling. Previous work with SSL on ECG data used SSL to extract spatiotemporal features for emotion recognition, outperforming purely supervised models [Sarkar and Etemad, 2020]. In EEG data, the Word2Vec-inspired *Wave2Vec* model used an embedding approach to predict features in EEG data that was concatenated with patient demographic data [Yuan et al., 2017]. Additional work in EEG evaluated relative positioning and contrastive methods for pre-training EEG models, and were able to achieve improved performance in sleep stage classification and pathology detection [Banville et al.]. Further contrastive approaches like this have also been tried for microelectrode data [Azabou et al.]. See Weng et al. for a review on further SSL techniques applied to EEG. Surprisingly, very little work has applied SSL methods to ECoG or other intracranial data, with only three papers we could find [Lam et al.; Lesaja et al.], including previous work from colleagues in my own lab [Peterson et al., a]. The sparse research landscape on SSL within biological time-series data, particularly in domains like ECoG, underscores the urgency for further investigation of SSL in biological contexts, where acquiring ample ground truth labels is often prohibitively expensive.

# Chapter 3

# Generalized neural decoders for transfer learning across participants and recording modalities

This chapter includes materials originally published in the Journal of Neural Engineering [Peterson et al., b]. My main contributions to this work are the transfer learning results shown in Figure 3.3. The neural network model, HTNet, presented in this chapter is used in many other chapters in this thesis, and we recommend that you familiarize yourself with the model by reading this chapter.

## 3.1 Introduction

Brain-computer interfaces that interpret neural activity to control robotic or virtual devices have shown tremendous potential for assisting patients with neurological disabilities, including motor impairments, sensory deficits, and mood disorders [Ganzer et al., 2020; Miller et al., 2020; Volkova et al., 2019; Niketeghad and Pouratian, 2019; Martin et al., 2019b; Sani et al., 2018; Wang et al., 2013; Leuthardt et al.]. At the same time, brain-computer interfaces offer new insights about the function of neural circuits, including how sensorimotor information is represented in the brain [Degenhart et al., 2020; Oby et al., 2020; Collinger et al., 2018]. Advances in brain-computer interfaces have been driven in part by improved neural decoding

algorithms [Gu et al., 2020; Rao, 2013]. However, it can be difficult to collect enough data to train decoders, especially given the non-stationary nature of the recorded signals, leading to decoders that generalize poorly to new data and require frequent re-calibrations [Wu et al.; Van Erp et al., 2012; Huang et al., 2010; Cohen, 2014b]. Alternatively, *generalized neural decoders* can be trained by pooling data across multiple participants [Tan et al., 2018; Volker et al.; Zubarev et al.; Azab et al.]. Such generalized decoders must be robust to inter-participant differences and capable of fine-tuning with only a few training examples. By increasing decoder robustness and reducing the burden of repeated calibrations, generalized decoders have the potential to greatly enhance the practical long-term usage of brain-computer interfaces [Brunner et al., 2015].

Frequency-domain techniques that extract spectral power features from time-domain recordings have long been shown to be useful in decoding neural population recordings. These techniques are especially well-suited to neural recordings such as intracranial electrocorticography (ECoG) and scalp electroencephalography (EEG) recordings, which contain oscillatory signals at specific frequency bands that correspond to different behaviors or neural phenomena [Miller, 2019; Parvizi and Kastner, 2018; Takaura et al., 2016; Gunduz et al., 2011; Pistohl et al., 2008]. In addition, relative spectral power patterns between a task and a baseline condition can be surprisingly similar across studies, even when measured with different neural recording modalities [Martin et al., 2019a; Peterson and Ferris, 2018; Hell et al., 2018; Jiang et al., 2018; Zavala et al., 2016; Fujioka et al., 2012]. Such similarities motivate the use of spectral power features for generalized decoding. Because neural recordings are non-stationary, many decoders use instantaneous spectral power features, computed by band-pass filtering the data and then applying the Hilbert transform [Cohen, 2014b].

To make power spectral features directly comparable, inter-participant differences in electrode placement and frequency content must be addressed when developing generalized decoders. While EEG electrode coverage is typically standardized across participants, invasive ECoG electrode placement is clinically motivated and highly variable, making it difficult to align electrodes from one participant to the next [Parvizi and Kastner, 2018; Schalk and Leuthardt, 2011]. A similar cross-participant alignment issue occurs with EEG cortical dipoles following blind source separation [Onton et al., 2006]. To overcome these variable dipole locations, one successful approach has been to project EEG measures onto common brain regions using radial basis function interpolation [Bigdely-Shamlo et al., 2013; Peterson et al., 2018]. ECoG signals can similarly

be projected from electrodes to common brain regions [Peterson et al., 2021], but it has remained unclear how useful this method is for neural decoding. Even with aligned electrode placements, the frequency bands containing behaviorally-relevant spectral power can be highly variable across participants [Peterson et al., 2021; Saha and Baumert, 2019; Krumpe et al., 2017]. This variability is problematic when selecting band-pass filter cutoff frequencies prior to applying the Hilbert transform because researchers often rely on pre-existing knowledge of the neural signal, and traditional frequency bands may not apply to a particular task of interest. To decrease user bias, data-driven approaches have been proposed to analyze neural spectral power [Corcoran et al., 2018; Adam et al., 2014; Chiang et al., 2008]. However, these techniques apply to frequency bands with distinct spectral power peaks and thus ignore other frequencies that might be useful for decoding. While a promising approach has been recently proposed [Cohen, 2020], developing decoders that robustly handle variable frequency content remains an open problem.
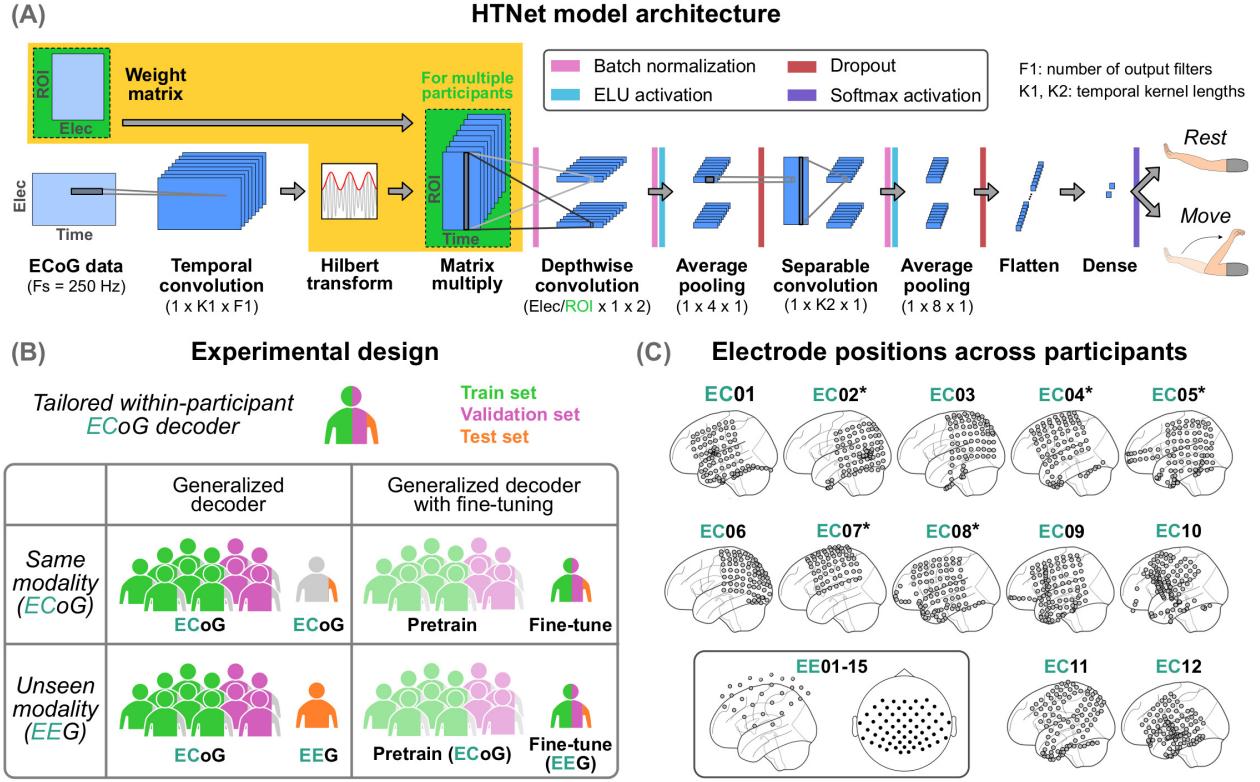
We are motivated to address such cross-participant differences in frequency content by using convolutional neural networks, which generate data-driven features and can also be fine-tuned when presented with new data. Convolutional neural networks combine data-driven feature extraction with pattern recognition and have set the bar for state-of-the-art neural decoding performance for speech, motor imagery, and attention tasks [Wu et al.; Anumanchipalli et al., 2019; Xu et al., 2020; Ming et al., 2019; Roy et al., 2019; Zhang et al., 2019]. Importantly, trained convolutional neural networks can be fine-tuned to new data, a process made more efficient by freezing various network layers during re-training [Makin et al.; Behncke et al., 2018]. Far from being "black-box" models, convolutional layers perform data-driven temporal and spatial filtering, and careful analysis of trained convolutional weights can be used to interpret the spatiotemporal features that are key for decoding [Sakhavi et al., 2018; Schirrmeister et al., 2017; Bashivan et al., 2015]. Our approach builds on EEGNet—a compact convolutional neural network decoder that can be trained on small datasets, provides interpretable model structure, and has outperformed other deep learning decoders [Lawhern et al.].

In this paper, we present *HTNet*, a convolutional neural network architecture that decodes neural data with variable electrode placements using data-driven spectral features projected onto common brain regions (Figure 3.1A). By training on data pooled across multiple participants, HTNet avoids the issue of neural data scarcity that often results in overfit decoders. We developed HTNet by augmenting EEGNet with custom

Hilbert transform and projection onto brain region layers. The main advantage of our approach is its ability to generate data-driven features in the frequency domain that are spatially aligned across participants even when electrode placement is variable, enabling generalized decoding. To characterize its decoding performance, we trained HTNet to classify naturalistic arm movement vs. rest using ECoG recordings from a dataset containing simultaneous video and ECoG recordings from 12 patients being monitored before neurosurgery in a hospital [Peterson et al., 2021]. We then tested these trained decoders' ability to generalize to unseen ECoG participants or to unseen EEG participants performing a similar behavioral task [Ofner et al., 2017]. In both cases, HTNet was able to decode these naturalistic arm movements and consistently outperformed other state-of-the-art decoders. Furthermore, we fine-tuned pretrained HTNet decoders with a few events from individual participants, and the resultant fine-tuned decoders approach the performance of the best tailored decoders with only 50 ECoG or 20 EEG events. Finally, we interpreted HTNet's trained weights to understand how it generalizes and found that it primarily relied on physiologically-relevant features at low-frequencies ($<20$ Hz) near the motor cortex. Our findings demonstrate that HTNet is a generalized decoder that robustly handles inter-participant and inter-modality variability and fine-tunes to new participants using minimal data. We believe our work advances the field of generalized neural decoders with an architecture that is robust, interpretable, and successful at generalizing across unseen participants and recording modalities.

The rest of this paper is structured as follows. We introduce the neural datasets and pre-processing steps used (Section 3.2.1–3.2.3). We then detail HTNet's model structure, alternative decoders used for comparison, and the three generalization scenarios (Section 3.2.4–3.2.5). Next, we describe the methods used to tune decoder hyperparameters, analyze fine-tuning performance, and interpret trained model weights (Section 3.2.6–3.2.11). In Section 3.3, we present our results for cross-participant and cross-modality decoding and fine-tuning. Finally, we discuss the implications of our findings along with the limitations of our approach (Section 3.4).

## 3.2 Methods



**Figure 3.1: Overview of HTNet architecture, experimental design, and electrode locations. (A)** HTNet is a convolutional neural network architecture that extends EEGNet [Lawhern et al.] (differences shown in yellow) by handling cross-participant variations in electrode placement and frequency content. The temporal convolution and Hilbert transform layers generate data-driven spectral features that can then be projected from electrodes (Elec) onto common regions of interest (ROI) using a predefined weight matrix. **(B)** Using electrocorticography (ECoG) data, we trained both tailored within-participant and generalized multi-participant models to decode arm movement vs. rest. Multi-participant decoders were tested separately on held-out data from unseen participants recorded with either the same modality as the train set (ECoG) or an unseen modality (EEG). We then fine-tuned these pretrained decoders using data from the test participant. **(C)** Electrode placement varies widely among the 12 ECoG participants. Electrode coverage is sparser for the 15 EEG participants compared to ECoG, but both modalities overlap in coverage of sensorimotor cortices. Asterisks denote five participants whose electrodes were mirrored from the right hemisphere.

### 3.2.1 Intracranial electrocorticography (ECoG) dataset

We obtained concurrent ECoG and video recordings from 12 human participants (8 males, 4 females) during continuous clinical epilepsy monitoring conducted at Harborview Medical Center in Seattle, WA. These recordings lasted 7±2 days per participant (mean±SD). Participants were aged 29±8 years old (mean±SD)

and had electrodes implanted primarily in one hemisphere (5 right, 7 left). Our study was approved by the University of Washington Institutional Review Board for the protection of human participants, and all participants provided written informed consent.

Our decoding task was to classify upper-limb "move" and "rest" events of the arm contralateral to the implanted electrode hemisphere. We obtained non-concurrent move and rest events from video recordings via markerless pose tracking and automated state segmentation (see Singh et al. [2020] for further details). Move events correspond to wrist movement that occurred after at least 0.5 seconds of no movement, while rest events indicate no movement in either wrist for at least three seconds.

We performed ECoG data processing using custom MNE-Python scripts [Gramfort et al., 2013]. We first removed median DC drift and high-amplitude discontinuities. Each participant's ECoG data was then band-pass filtered (1–200 Hz), notch filtered, and re-referenced to the common median across electrodes. We also removed noisy electrodes based on abnormal standard deviation ($> 5$ IQR) or kurtosis ($> 10$ IQR). Next, we generated 10-second ECoG segments centered around each "move" and "rest" event. ECoG segments with missing data or large artifacts were removed based on abnormal spectral power density. See Peterson et al. [2021] for further ECoG pre-processing details. We then downsampled to 250 Hz and trimmed segments to two seconds centered around each event. For every participant, we balanced the number of move and rest segments within each recording day, resulting in $1155 \pm 568$ events per participant (mean$\pm$SD).

Electrode positions were localized using the Fieldtrip toolbox in Matlab [Stolk et al., 2018; Oostenveld et al., 2011]. This process involved co-registering preoperative MRI and postoperative CT scans, manually selecting electrodes in 3D space, and warping electrode positions into Montreal Neurological Institute (MNI) space [Fonov et al., 2011]. Using this common MNI coordinate system enabled us to directly compare electrode positions between ECoG participants (see Fig. 3.1C).

### 3.2.2 Comparative cross-modal dataset (EEG)

To test decoder generalizability across recording modalities, we used a publicly available EEG dataset of 15 human participants performing cued right elbow flexion movements [Ofner et al., 2017]. Participants were aged $27 \pm 5$ years old (mean$\pm$SD) and performed 60 movement and 60 rest trials each, resulting in 120 total events. Because only cue onset times were available, we determined the onset of each movement event by

thresholding the hand's radial displacement after it was cued to move.

EEG data were recorded at 512 Hz, notch filtered at 50 Hz, referenced to right mastoid, and band-pass filtered between 0.01–200 Hz. We pre-processed the data by average referencing, 1 Hz high-pass filtering, resampling to 250 Hz, and generating 2-second segments centered around each event. Each participant had 61 EEG electrodes, whose MNI positions were estimated using the 10-5 system template from Field-trip [Oostenveld et al., 2011; Oostenveld and Praamstra, 2001].

### 3.2.3   Computing projection matrices

We accounted for variations in electrode placement by mapping electrode positions onto common brain regions based on distance as described below. To increase electrode overlap among ECoG participants, we mirrored all right hemisphere electrode positions onto the left hemisphere. Using EEGLAB and Matlab, we mapped from electrode positions to small, predefined brain regions by computing radial basis function kernel distances between each electrode and brain region (2 cm full-width at half-maximum) [Bigdely-Shamlo et al., 2013; Vert et al., 2004; Delorme and Makeig, 2004]. This projection procedure was performed separately for each participant. We projected to regions within sensorimotor areas, as defined by the AAL atlas [Tzourio-Mazoyer et al., 2002] (precentral, postcentral, and inferior parietal), in order to limit the projected data size. We then normalized these distance values for every region so that each region's values across all electrodes summed to one. These normalized distances created a projection matrix for each participant, which we later used to estimate the activity at each common region of interest by performing a weighted average of electrode-level data.

### 3.2.4   HTNet architecture

HTNet builds upon EEGNet [Lawhern et al.], a compact convolutional neural network developed using Python 3.8 and Tensorflow 2.2. EEGNet has three convolution layers: (1) a one-dimensional convolution analogous to temporal band-pass filtering, (2) a depthwise convolution to perform spatial filtering, and (3) a separable convolution to identify temporal patterns across the previous filters. For HTNet, we added a Hilbert transform layer after this initial temporal convolution to compute relevant spectral power features using a data-driven filter-Hilbert analog (see Figure 3.1A). We then added a matrix multiplication layer to

project electrode-level spectral power onto common brain regions of interest, using the pre-computed weight matrices described in the previous section. Note that the matrix multiplication layer was not necessary when the same participant was used for training and testing, as the electrodes remain consistent. It is worth noting that the time complexity of the Hilbert transform and matrix multiplication layers are $O(n \log n)$ and $O(n^3)$, respectively, where $n$ represents the number of samples per segment in the input neural data [Van Loan, 1992]. All other HTNet layers were the same as EEGNet.

### 3.2.5  Data division and cross-validation

We compared HTNet decoding performance against EEGNet, random forest, and minimum distance decoders. The minimum distance decoder used Riemannian mean and distance values for classification [Congedo et al., 2017; Yger et al., 2016]. We assessed decoder performance during three scenarios (Fig. 3.1B): (1) testing on an untrained recording day for the same ECoG participant (*tailored decoder*), (2) testing on an untrained ECoG participant (*same modality*), and (3) testing on participants from the EEG dataset after training only on the ECoG dataset (*unseen modality*). Note that we used the same trained, multi-participant ECoG decoders for same and unseen modality conditions. Additionally, we projected data onto common regions of interest for all decoders in order to enable reasonable decoding for the same and unseen modality conditions. For all scenarios, we performed 36 pseudo-random selections (folds) of the training and validation datasets, such that each of the 12 ECoG participants was the test participant three times. We used each ECoG participant's last recording day as the test set (orange in Fig. 3.1B) and excluded it from all training and validation sets. All training, validation, and test sets were balanced with equal numbers of move and rest events. We used nonparametric statistics to test for significant effects of decoder type on test accuracy (Friedman test, $p < 0.05$) and significant pairwise differences among decoders (Wilcoxon signed-rank test with false discovery rate correction [Benjamini and Hochberg, 1995]).

### 3.2.6  Hyperparameter tuning

We performed hyperparameter tuning to identify optimal values for each decoder. We tuned HTNet and EEGNet simultaneously using six hyperparameters: temporal kernel length, separable kernel length, temporal filter count, dropout rate, dropout type, and model type (HTNet or EEGNet). For the random forest

42

decoder, we tuned two hyperparameters: maximum depth and number of estimators. The minimum distance decoder had no tunable hyperparameters.

We tuned hyperparameters separately for the tailored decoder and same modality conditions, using the Optuna toolbox [Akiba et al., 2019]. We ran 25 random forest parameter selections (trials) and 100 HTNet/EEGNet trials for each condition. Performance was measured using validation accuracy, averaged over 36 folds for tailored decoding or 12 folds for same modality. We sampled from parameter space using Optuna's tree-structured Parzen estimator [Bergstra et al., 2013, 2011], which selects optimal parameter values based on the performance during previous trials.

Overall, we found that hyperparameter selections minimally affected decoder performance (see Fig. A.1). Still, we selected hyperparameter values from the trial with the highest validation accuracy (Table A.1) for each condition to ensure optimal decoder performance.

### 3.2.7 Fine-tuning decoder performance to the test participant

In addition to testing generalizability, we assessed how much a generalized HTNet decoder improves when re-trained using data from the test participant, a process known as *fine-tuning*. Fine-tuning is a transfer learning technique where some layers of the pretrained model are "frozen" and not adjusted during re-training, reducing the number of parameters to fit [Yosinski et al., 2014]. We fine-tuned our pretrained same and unseen modality decoders using a portion of the test participant's data. Additionally, we fine-tuned each HTNet convolutional layer separately and all layers together, resulting in four re-trained models per fold (see Fig. 3.3A). When separately tuning each convolutional layer, we also re-trained the nearby batch normalization layers, as shown in Fig. 3.3A, which notably boosted performance. We tested for significant differences among these four fine-tuning models using Wilcoxon signed-rank tests with false discovery rate correction.

During fine-tuning, we also varied the amount of training/validation data available in order to assess its impact on fine-tuning performance. For all variations, the test set remained fixed as the last recording day for each ECoG participant and 30 randomly-selected events for every EEG participant. Using the remaining test participant data, we selected four amounts of training/validation data: 17% training/8% validation, 33% training/17% validation, 50% training/25% validation, and 67% training/33% validation. We then

43

linearly modelled the relationship between test accuracy and the logarithm of the number of training events for each fine-tuning model. All training, validation, and test sets contained equal numbers of move and rest events. Note that the number of events used for fine-tuning differs across ECoG participants because the total number of events varies. In addition, we trained randomly-initialized HTNet decoders with the same training/validation data used for fine-tuning in order to compare tailored decoding with our fine-tuned models.

### 3.2.8 Comparing performance of HTNet spectral measures

Although we primarily used HTNet to generate data-driven spectral power features, HTNet can be easily adapted to generate other spectral measures that may boost decoding performance. These spectral measures still use the Hilbert transform, which can be used to find instantaneous power, phase, or frequency [Cohen, 2014b]. For the same three conditions, we tested four spectral measures: (1) log-transform of one plus power, (2) relative power, (3) unwrapped instantaneous phase, and (4) instantaneous frequency, also known as frequency sliding [Cohen, 2014a]. We tested for significant differences in test accuracy when HTNet decoders implemented one of these four measures or spectral power, using Wilcoxon signed-rank tests with false discovery rate correction.

When computing relative power, we first took the log-transform of one plus power and then subtracted out the average power from -1 to -0.5 seconds before each event. This procedure is analogous to baseline-subtraction of spectral power and should be robust to large-scale spectral power variations across days, participants, and recording modalities. Because of its potential robustness to large differences in signal scaling, relative power was also used in HTNet decoders that we used to compare with other decoder types, but only for the unseen modality condition.

### 3.2.9 Interpreting model weights

Like EEGNet, HTNet's first two convolutional layers are interpretable and can indicate the spatiotemporal features used for decoding. To analyze the temporal features, we fed a white noise signal into HTNet's trained temporal convolution and computed the frequency response magnitude using Welch's method, similar to a Bode plot [Yarlagadda, 2010]. We then averaged frequency responses across temporal filters and

folds. To determine important spatial features, we computed the absolute value of HTNet's trained depth-wise convolution weights [Lawhern et al.], averaged across filters. This process generated one weight for each brain region of interest. We then scaled the maximum value per fold to one and averaged across folds.

While the temporal frequency response shows which frequencies were used for decoding, it does not show what the activity looks like at these frequencies. To visualize such activity, we computed the difference in log spectral power between move and rest conditions, projected onto a region near the motor cortex (Fig 3.4B far left region in the second row from the top). We took the difference between the average move and average rest log spectrograms for each participant and then averaged the resulting differences across participants for both the ECoG and EEG datasets.
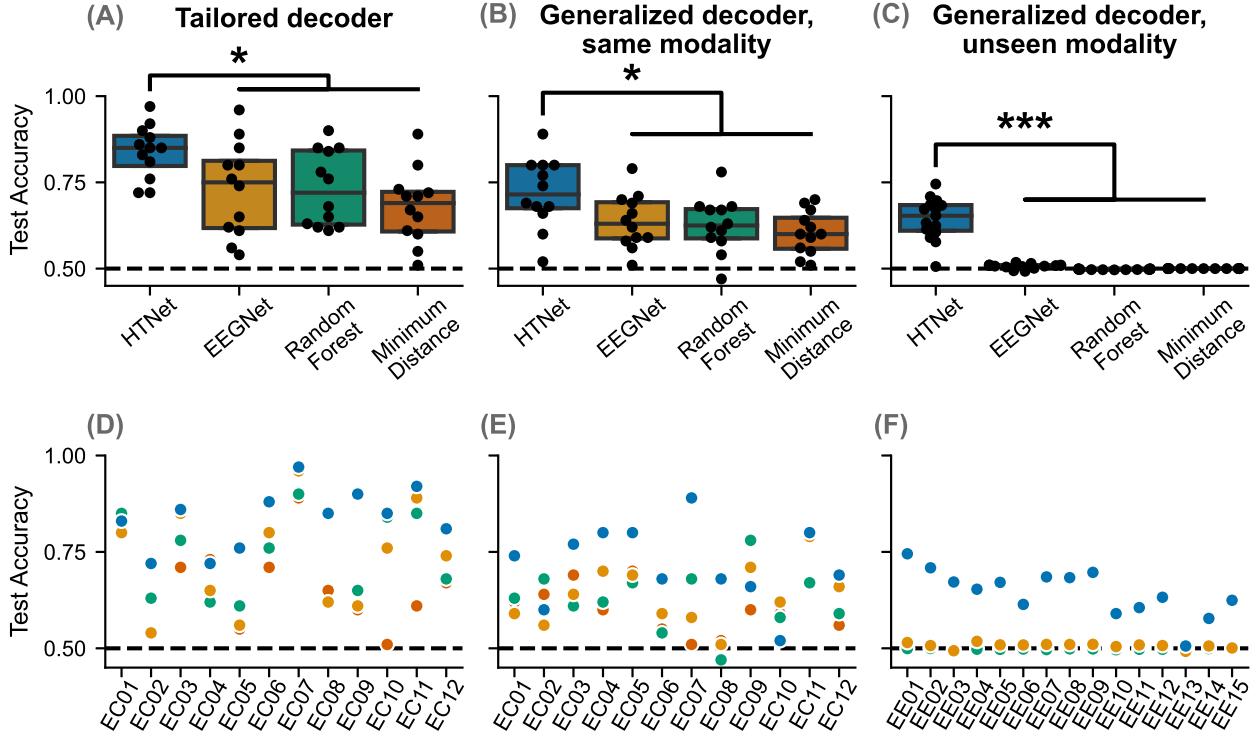
### 3.2.10  Effect of training participants on performance

We also assessed how many training participants are needed for improved decoder performance. Increasing the number of training participants should improve decoder performance on an unseen test participant, but the improvement from adding new participants will likely diminish as more training participants are added. We varied the number of training participants from 1–10, always using one validation participant. Participants were pseudo-randomly selected across 36 folds such that each participant was the test participant three times. We linearly modelled the relationship between test accuracy and the logarithm of the number of training participants, identifying decoders with significant nonzero trends ($p < 0.05$, two-tailed t-test with false discovery rate correction).

### 3.2.11  Effect of electrode overlap on performance

Because electrode locations varied among ECoG participants, we also tested if higher decoding performance corresponded to increased electrode overlap between same modality training and test participants. We estimated electrode overlap between training and test participants using a custom fraction overlap metric that allowed us to combine multiple participants from the training set. For each sensorimotor region, we summed unnormalized projection matrix weights across electrodes to estimate how many electrodes were nearby. We then identified regions of high electrode coverage for each participant by thresholding summations greater than 0.07. Because these regions are common across participants, we could average these summation values

across training participants prior to thresholding. Next, we divided the number of thresholded regions common to both training and test participants by the number of thresholded regions in the training set to obtain fraction overlap. A fraction overlap of 1.0 indicates that the test participant's thresholded regions include all thresholded regions from the training participants. We linearly modelled the relationship between fraction overlap and test accuracy across folds and identified decoders with significant nonzero slope ($p < 0.05$, two-tailed t-test with false discovery rate correction [Benjamini and Hochberg, 1995]).



**Figure 3.2: HTNet generalizes better than EEGNet and other decoders.** HTNet achieves significantly higher test accuracy than EEGNet, random forest, and minimum distance decoders across all three scenarios: **(A)** tailored ($p < 0.05$), **(B)** same modality ($p < 0.05$), and **(C)** unseen modality ($p \leq 0.001$). Note that the trained models for same and unseen modality conditions are identical; only the test set differs. **(D–F)** Bottom row displays decoder performance grouped by test participant for each fold. For unseen modality, HTNet uses relative power to minimize cross-modal scaling differences, resulting in performance that is above chance (dashed line) despite only training on ECoG data.
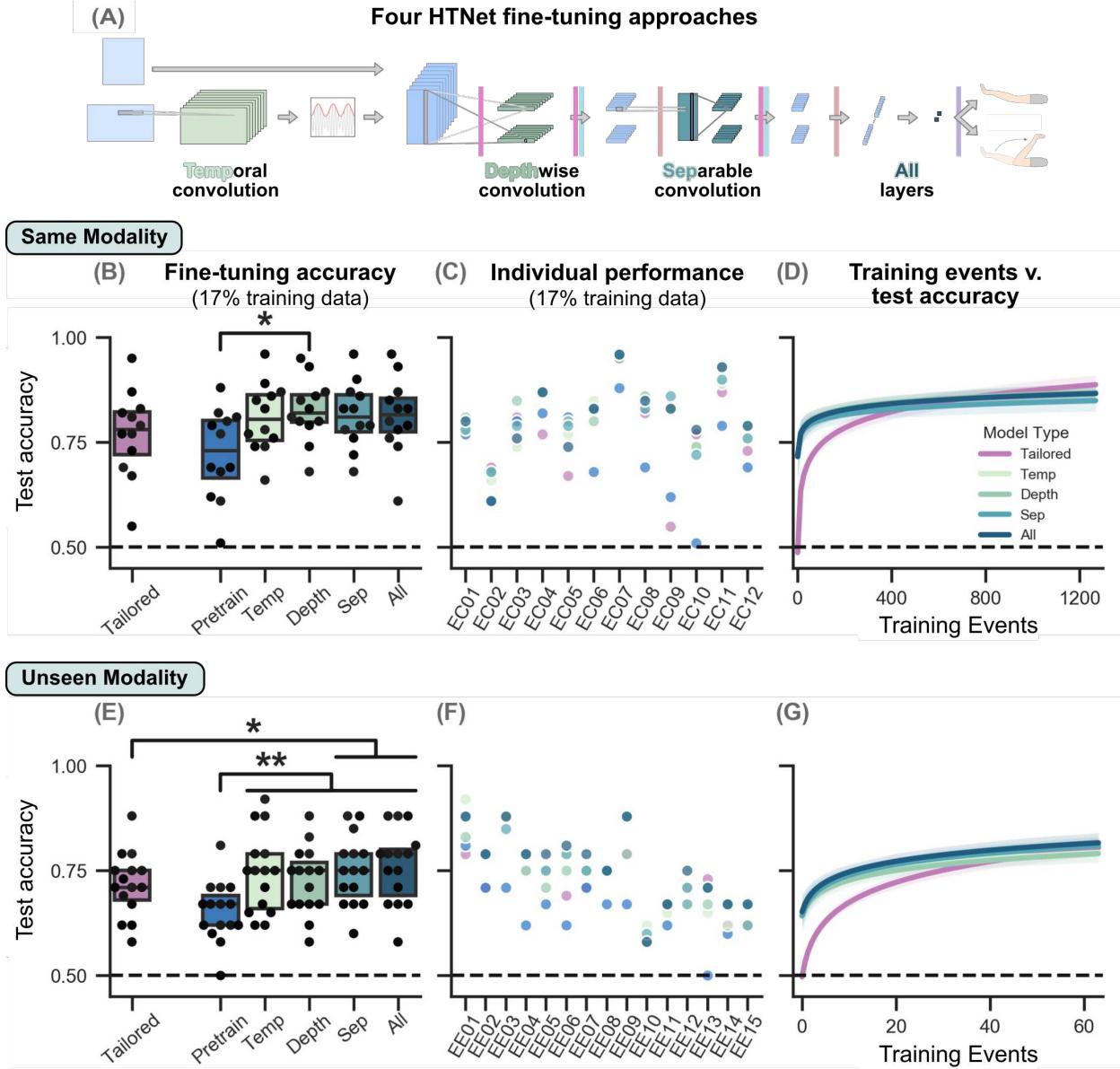
## 3.3 Results

Here, we show that our approach to movement decoding, HTNet, is generalizable and tunable, learning common patterns from the training data that transfer to unseen participants and recording modalities. Through

a series of systematic experiments, we use decoders tailored to each participant—by far the most common approach to training decoders—as a standard to which the performance of our generalized decoders are compared (Figure 3.1B). In particular, our training data are from 12 ECoG participants during uninstructed, naturalistic arm movements (Figure 3.1C, [Singh et al., 2020; Peterson et al., 2021]); our test data are then either one ECoG participant withheld from the training set, or participants from an entirely independent EEG dataset. HTNet consistently outperformed other decoders, and fine-tuning pre-trained HTNet decoders with a small number of the unseen participant's events yielded decoders that approached the performance of tailored decoders trained on many more events. Further, we show that HTNet works by extracting physiologically-relevant spectral features from the data.

### 3.3.1 Decoder generalization

HTNet consistently outperformed other decoders in all conditions, including tailored within-participant decoding, generalized decoding to unseen participants in the same modality, and generalized decoding to unseen modality participants (Figure 3.2). For each condition, we found a significant effect of decoder type on test accuracy (tailored: $p = 3.94e{-}4$, same modality: $p = 0.034$, unseen modality: $p = 1.95e{-}8$ respectively; Friedman test). For tailored decoding, HTNet achieved test accuracy of $84\% \pm 8\%$ (mean $\pm$ SD), which was significantly higher than EEGNet ($73\% \pm 14\%$, $p = 0.003$), random forest ($73\% \pm 11\%$, $p = 0.004$), and minimum distance decoders ($68\% \pm 11\%$, $p = 0.010$; Wilcoxon signed-rank test with false discovery rate correction). For same modality, HTNet test accuracy of $72\% \pm 10\%$ was again significantly increased compared to EEGNet ($64\% \pm 8\%$, $p = 0.042$), random forest ($63\% \pm 8\%$, $p = 0.037$), and minimum distance decoders ($60\% \pm 6\%$, $p = 0.021$). No other significant pairwise differences were found for either condition.
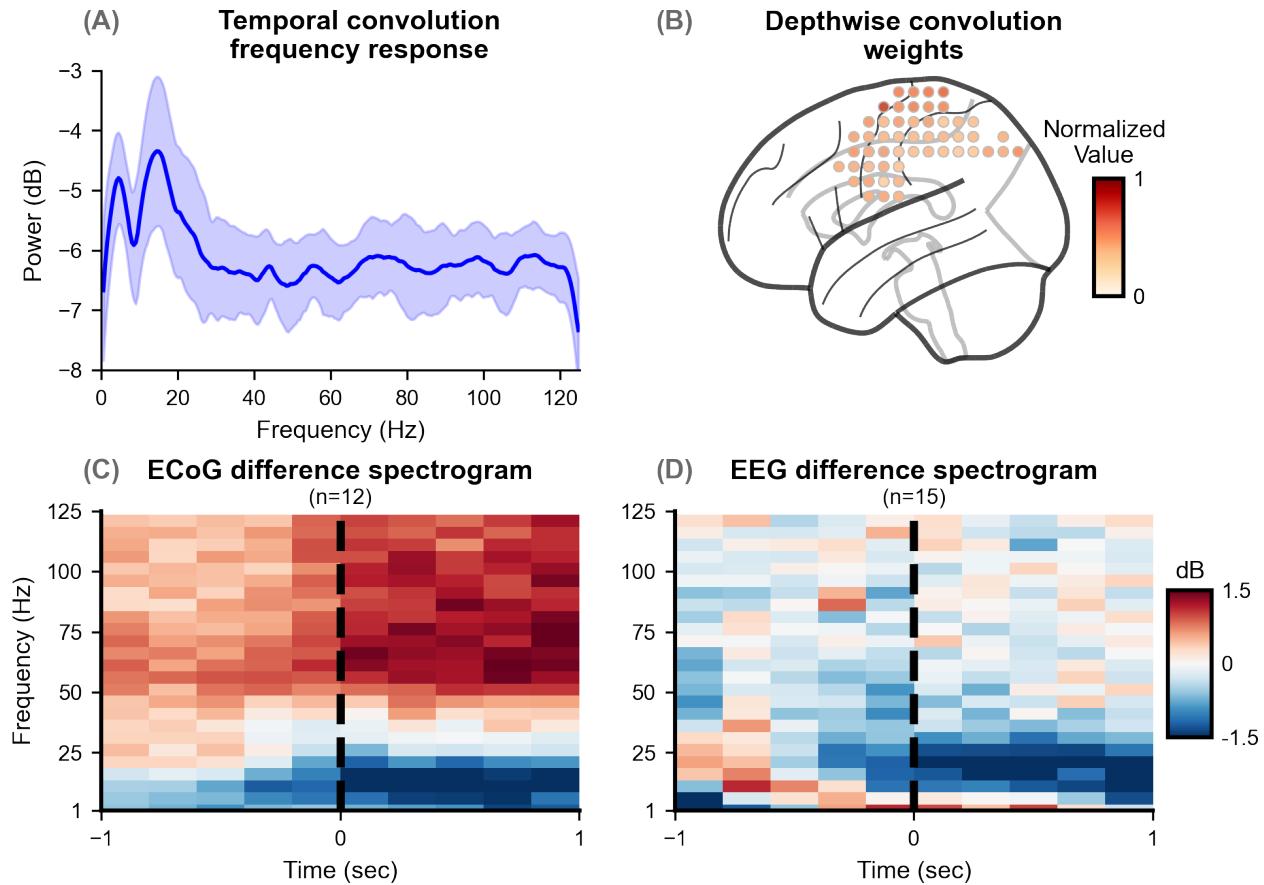
**Figure 3.3: Fine-tuning HTNet improves performance, even when few training events are available.** **(A)** For pretrained, multi-participant HTNet decoders, we separately fine-tuned each convolution layer (with nearby batch normalizations) and also fine-tuned all trainable layers. Layers with trainable parameters are highlighted. **(B, E)** For both same and unseen modality cases, we find no performance differences between fine-tuning approaches when training on 17% of available events, but our fine-tuned decoders often significantly improve test accuracy compared to the pretrained models and, in some cases, the randomly-initialized tailored decoders (* $p < 0.05$, ** $p \leq 0.01$). **(C, F)** Similarly, a breakdown of performance by participant shows that fine-tuning usually improves performance compared to the pretrained model (blue), but no single fine-tuning approach decodes consistently better than the others. **(D, G)** As we decreased the number of training events below 400 (same modality) or 30 (unseen modality), fine-tuned decoders generally achieved higher test accuracy than tailored decoders and remained close to accuracies achieved by tailored decoders trained on all available events. Lines show logarithmic fits for each group, with shading denoting the 95% confidence interval of the slope. Note that pretrained model accuracies are included for all fine-tuned models when there are 0 training events.

In the unseen modality condition, HTNet was the only decoder to generalize above chance. All pairwise comparisons were statistically significant ($p < 0.05$) in the unseen modality condition, due to low variability in cross-participant test accuracy for each non-HTNet decoder. However, HTNet's unseen modality test accuracy was, on average, ~15% higher than all other decoders and the only decoder to perform well-above chance (50%). Therefore, we only show the pairwise statistical results in Figure 3.2C between HTNet and each of the other three decoders. Average computational time during training is shown for each decoder type in Tables A.2 and A.3. In addition to spectral power, we also developed HTNet models that decoded using instantaneous phase and frequency features (Fig. A.2 and Tables A.4, A.5). Briefly, HTNet with phase performed worse than spectral power for tailored and same modality conditions, while using instantaneous frequency resulted in similar performance to HTNet with spectral power.

In addition to outperforming the other decoders on average, we show in Figure 3.2D–F that HTNet was the single best arm movement decoder for almost every individual participant in all conditions. HTNet performance was also consistently well-above random chance for all but two participants (EC10 same modality and EE13), much more than any other decoder. HTNet's consistently high accuracy demonstrates its ability to robustly generalize to a variety of participants with differences in electrode placement and signal quality.

### 3.3.2 Fine-tuning generalized decoders
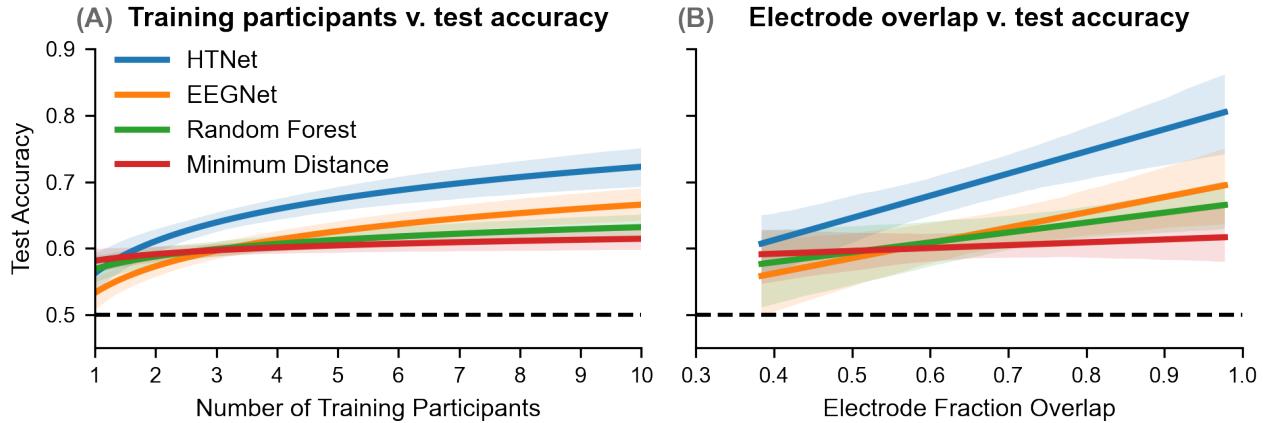


**Figure 3.4: HTNet extracts physiologically-relevant features at low frequencies and near the motor cortex.** By analyzing HTNet's early convolution layers, we determined the types of spatial and temporal features consistently used for multi-participant decoding. **(A)** The temporal convolution layer's frequency response, averaged across filters and folds, shows a consistent focus on low frequency features (<20 Hz). **(B)** Based on depthwise convolution weight magnitudes, cortical regions near the central sulcus and towards the midline were found to be consistently important for decoding, as expected for upper-limb movements. Out of the 144 total regions of interest used for decoding, we show here the 51 regions (shaded circles) located on the cortical surface. **(C–D)** Difference spectrograms between arm movement and rest events reveal a common low-frequency (<25 Hz) spectral power decrease near movement onset (0 sec) of similar magnitude across ECoG and EEG datasets.

By fine-tuning these pretrained HTNet decoders using as few as 50 events from the held-out participant, HTNet can approach performances of the best tailored decoders trained using all available participant data. We tested fine-tuning each convolutional layer separately as well as re-training all layers of the network, as shown in Fig. 3.3A. We found no significant differences in performance between our fine-tuning ap-

proaches, but we did find significant improvements in test accuracy when comparing fine-tuned decoders to pretrained decoders or tailored decoders trained on the same data, even when training on only 17% of the test participant's available events. For same modality (Fig. 3.3B), fine-tuning just the depthwise convolution significantly increased test accuracy compared to the pretrained models ($p = 0.049$; Wilcoxon signed-rank test with false discovery rate correction). In the unseen modality condition (Fig. 3.3E), all fine-tuning approaches significantly increased test accuracy compared to the pretrained models (temporal convolution: $p = 0.0054$, depthwise convolution: $p = 0.0094$, separable convolution: $p = 0.0054$, all layers: $p = 0.0054$). Additionally, fine-tuning the separable convolution and all HTNet layers resulted in significantly higher test accuracy than tailored HTNet decoders ($p = 0.021$ and $p = 0.015$, respectively). No other comparisons were statistically significant.

Similarly, decoder performance for each participant shows that, in general, fine-tuning HTNet increases test accuracy. All tested fine-tuning approaches yielded similar performance (Fig. 3.3C,F). In addition, computation times are consistent across fine-tuning approaches (Tables A.6, A.7). By varying the number of events used (Fig. 3.3D,G), we find that fine-tuning approaches outperform randomly-initialized, tailored decoders when fewer than ~400 ECoG or ~30 EEG events from the test participant are available for training (see Fig. A.3 for separated plots of each fine-tuning approach). In addition, fine-tuned HTNet decoders approach performances of the best tailored decoders with as few as 50 ECoG or 20 EEG events available for training. Finally, we demonstrated that we can fine-tune HTNet decoders to ECoG events after pretraining on EEG events (Fig. A.4).

**Figure 3.5: HTNet performance improves with increases in training participants and electrode overlap.** **(A)** Adding more training participants significantly improved test accuracy for HTNet, EEGNet, and random forest decoders ($p < 0.05$). The line of best fit is shown for each group, with shading denoting the 95% confidence interval of the slope. The logarithmic relationship seen suggests that performance will not substantially improve once a certain number of training participants are used. **(B)** We also compared test accuracy as a function of electrode coverage overlap between the training and test participants and found a significant positive relationship only for HTNet and EEGNet ($p < 0.05$).

### 3.3.3 Interpreting network computations

Trained HTNet models achieve generalized decoding by extracting physiologically-relevant features, specifically at low frequencies and near the motor cortex (Fig. 3.4). HTNet's temporal and depthwise convolutional layers are interpretable, so we can probe these trained layers to determine the spatiotemporal features most often used for decoding, just like with EEGNet [Lawhern et al.]. Here, we analyzed the trained layers of generalized HTNet decoders that were used for the same and unseen modality conditions.

Across decoders, we find a consistent emphasis on low-frequency ($<20\,\text{Hz}$) temporal features (Fig. 3.4A) when analyzing the average frequency response over temporal convolution filters. These decoders also frequently focused on cortical regions near the motor cortex (towards the central sulcus and midline), based on trained depthwise convolution weights (Fig. 3.4B). When we take the difference in spectral power between arm movement and rest events near these motor cortical regions, we find, consistent with previous ECoG and EEG studies [Chung et al., 2017; Miller et al., 2007], that low-frequency spectral power decreases during movement onset, with a similar magnitude for both ECoG and EEG data. This consistent magnitude in low-frequency power between recording modalities likely explains why our trained HTNet models generalized to the EEG participants prior to any fine-tuning. It is important to note that the increase in ECoG

high-frequency ($>$50 Hz) power, a pattern commonly linked to active movement [Nowak et al., 2018; Başar et al., 2001], was not present in the EEG dataset, likely due to temporal filtering by the skull [Srinivasan et al., 1998].

Another crucial factor in HTNet performance is the spatial overlap of electrodes between the training and test participants, as well as how many participants were used for training (Fig. 3.5). We observe significant positive trends between electrode fraction overlap and test accuracy for HTNet (0.33 slope, $p = 0.005$; t-test with false discovery rate correction) and EEGNet (0.23, $p = 0.025$). We also find significant logarithmic relationships between the number of training participants and test accuracy for HTNet (0.07 slope, $p = 2.77\mathrm{e}-7$), EEGNet (0.06, $p = 2.77\mathrm{e}-7$, and random forest (0.03, $p = 0.003$). We initially tried a linear fit, but found that logarithmically scaling the number of training events resulted in a better fit. This logarithmic trend suggests that, at a certain point, adding more participants will not noticeably improve decoder performance. The number of training participants also affects computation time and number of epochs used during training (Fig. A.5). Our findings suggest that HTNet is best able to incorporate information when many training participants are used, especially those with similar electrode placement to the test participant.

## 3.4 Discussion

We demonstrated that HTNet can outperform state-of-the-art neural decoders when generalizing to new participants, even when a different recording modality is used. Developed as an extension of EEGNet [Lawhern et al.], our key contributions to the neural network architecture are the addition of a Hilbert transform layer and a weight matrix to project individual electrode locations onto common brain regions. These trained HTNet decoders can be fine-tuned to a new test participant with fewer than 100 of the test participant's events and still decode almost as well as tailored decoders that have been trained on substantially more events. To achieve this performance, HTNet decoders consistently extract physiologically-relevant features, as revealed by our analysis of trained decoder weights.

To our knowledge, HTNet is the first decoder that can generalize and transfer its learning across both ECoG participants and different recording modalities. Previous studies have implemented decoders that can transfer across different EEG devices [Wu et al.; Xu et al., 2020; Wu et al., 2016; Nakanishi et al., 2020] or leverage data from concurrent recording modalities [Muraskin et al., 2018; Fatima and Kamboh, 2017], but

none of these decoders have demonstrated the ability to generalize to an entirely different recording modality. Yet, cross-modal generalization can be hampered by the limitations of each recording modality. For example, the high-frequency power increase seen with ECoG is not present in the EEG dataset (Fig. 3.4C–D), demonstrating the infeasibility of transferring from EEG to ECoG when high-frequency features are necessary to decode well. As for generalizing across participants, many decoders can do this with EEG data [Wu et al.], including EEGNet [Lawhern et al.], but development of analogous ECoG decoders has been hindered by the high variation in electrode placement across ECoG patients. Similarly, several studies have implemented fine-tuning when decoding EEG or ECoG data [Azab et al.; Makin et al.; Elango et al.; Wang et al., e; Desai et al.; Uran et al., 2019], but these decoders were trained on either one participant or non-brain data, instead of multiple ECoG participants. By training on multiple participants' data, HTNet can generalize effectively to unseen participants and avoid overfitting to any one participant.

When we compared performance across different types of decoders, HTNet consistently outperformed EEGNet, even in the tailored condition. These performance differences between HTNet and EEGNet arise solely from HTNet's Hilbert transform layer because for all decoders, we projected onto common brain regions when training on multiple participants. We used the Hilbert transform to compute spectral power, which unlike the time-domain signal does not include a phase component. When we used just this phase component for decoding, we found that HTNet performance substantially worsened compared to using spectral power (see Fig. A.2). This difference in accuracy suggests that decoding directly from the time-domain signal is suboptimal for our specific decoding task because the phase component is less informative than spectral power. Based on this insight, we would expect EEGNet to perform as well as or better than HTNet when decoding neural data with highly informative phase, such as event-related potentials [Aydarkhanov et al., 2020; Kappenman and Luck, 2016], depending on whether HTNet uses phase or spectral power to decode.

More broadly, HTNet demonstrates the value of integrating computational models, such as deep learning, with insights from neural signal processing. Fusing computational methods with scientific insight has inspired novel solutions that leverage the strengths of both approaches [Anumanchipalli et al., 2019; Schirrmeister et al., 2017; Wang et al., g]. An alternative to our approach would have been to develop an end-to-end neural network model that simply learns to compute spectral power or project electrode-level

signals onto a common grid. However, incorporating explicit transformations to spectral power and common cortical regions minimized the number of trainable parameters (and hence the amount of training data needed) and, like EEGNet, kept HTNet's initial layers interpretable. Furthermore, we were able to apply insights from previous ECoG/EEG research by explicitly baseline-subtracting spectral power within HTNet in order to generalize from ECoG to unseen EEG data [Hu et al., 2014; Parra et al., 2005]. On the other hand, deep learning models can generate data-driven features that may be computationally expensive to obtain using other available methods. While many data-driven spatial filtering methods are available [Cohen, 2014b; Makeig et al., 1996], identifying frequencies with relevant spectral power often requires either a brute force search or applying techniques such as wavelet convolution to compute power at several frequencies, increasing the size of an already high-dimensional dataset [Lee and Choi, 2019; Tayeb et al., 2019]. In contrast, HTNet converges quickly and provides a low-dimensional feature representation in the spectral domain. We believe that further embedding of neural signal processing into data-driven methods such as deep learning will continue to enhance the robustness and generalizability of future neural decoders.

Our study has several important limitations to consider. First, we classified two event types (arm movement versus rest), which is substantially less than the types of complex behaviors present in many neural decoding paradigms. Nonetheless, HTNet's architecture allows for decoding more than two types of events, and EEGNet has been shown to perform well when decoding among four types of behavior [Lawhern et al.]. Still, decoding more complex behaviors than the current task may impact HTNet's generalizability. We chose ECoG and EEG datasets from a task with only two event types because the electrode positions were in a common coordinate system (MNI), which was essential when projecting data onto common brain regions for multi-participant decoding. Our analysis also may have been limited by projecting only onto 144 sensorimotor brain regions, which we chose based on the decoding task. For real-world decoders, the most informative regions may not be clear, requiring data-driven region-selection approaches to avoid high memory usage and slow computation times [Peterson et al., 2018, 2021]. Additionally, we could have constrained HTNet's temporal convolution layer to learn more meaningful narrow-band filters [Ravanelli and Bengio, 2018]. Finally, HTNet itself is limited by the assumption that spatiotemporal features are consistent across participants and the use of traditional spectral measures that often conflate periodic and aperiodic components of neural activity [Donoghue et al., 2020].

We are currently exploring extensions of HTNet for a variety of applications such as cross-frequency coupling [van der Meij et al., 2012; Canolty and Knight, 2010], long-term state decoding [Sani et al., 2018], cross-task decoding [He and Wu, 2020], and data-driven regression [Cui et al., 2019; Wu et al., 2017]. In addition, other decoding measures could be substituted for the Hilbert transform, including non-Fourier methods [Cole and Voytek, 2019, 2017], and more complex interpolation schemes could be used to generate the projection matrix by incorporating participant-specific cortical anatomy [Owen et al., 2020; Vermaas et al., 2020]. Besides ECoG and EEG, HTNet may also be useful for generalizing across participants with stereotactic EEG or local field potential recordings [Herff et al., 2020; Jackson and Hall, 2016]. Overall, HTNet provides a useful decoding framework applicable across a variety of tasks and overcomes important obstacles towards developing robust, generalized neural decoders that can be fine-tuned with minimal data.

## 3.5   Summary

In this chapter, we developed and tested the HTNet model. We showed that this model could decode either ECoG or EEG data, and could generalize to unseen participants in either modality. The model was also able to decode data that came from naturalistic contexts - where ECoG participants spontaneously generated their movements based on personal need. Our generalizable model could be further fine-tuned to new participants to improve model performance. This chapter emphasizes model improvements that can led to more useful "out-of-the-box" models for BCIs. In many of the following chapters we explore additional use cases for this generalized model and make improvements to the architecture.

## Code and data availability

Our HTNet code is publicly available at: `https://github.com/BruntonUWBio/HTNet_generalized_decoding`. The code in this repository can be used in conjunction with publicly available ECoG (`https://figshare.com/projects/Generalized_neural_decoders_for_transfer_learning_across_participants_and_recording_modalities/90287`) and EEG [Ofner et al., 2017] datasets to generate all of the main findings and figures from our study.

# Chapter 4

# Data augmentation and cross-participant generalization of EEG decoders for cognitive tasks

In the previous chapter, we explored generalized deep learning models for decoding movement — an essential aspect of human behavior. However, our daily lives encompass a variety of cognitive tasks that are equally critical to navigate the world. To enhance the versatility and applicability of Brain-Computer Interface (BCI) systems across diverse contexts, we must also focus on developing generalized neural decoders for cognitive states.
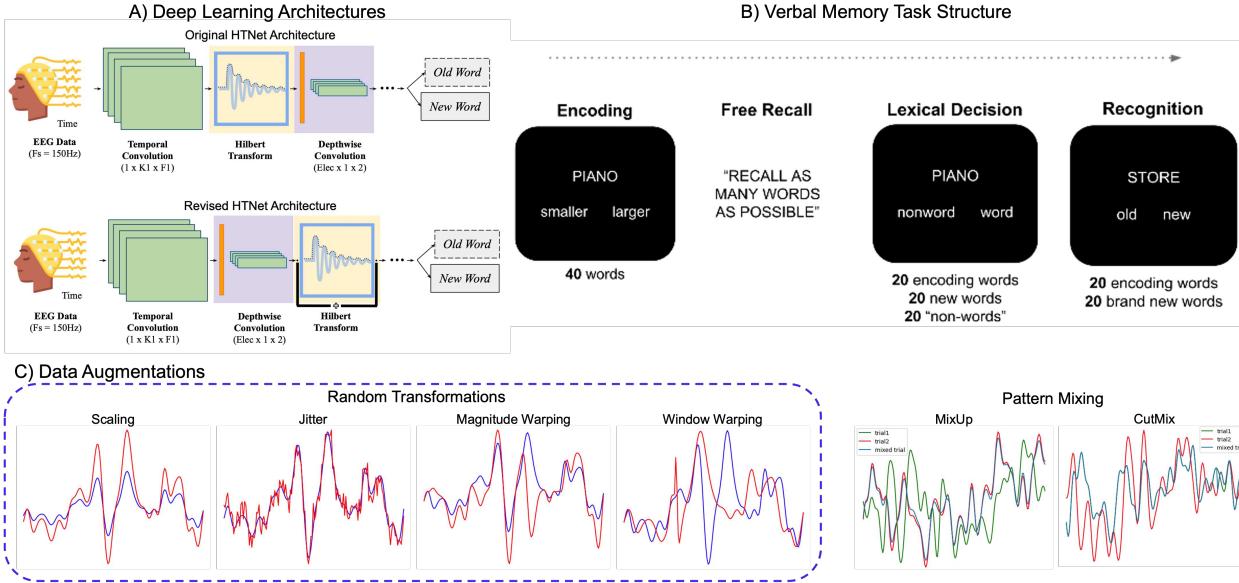
## 4.1   Introduction

In this chapter, I present our efforts to develop neural decoders to detect cognitive information from electroencephalography (EEG) data. We trained and developed neural decoders on data from a verbal recognition task, where participants were exposed to a series of words, and later tasked with recalling whether they had previously encountered each word (detailed methodology can be found in Section 4.2). We trained our models to predict whether participants had recognized the words during the recall phase of the experiment. Our objective was to investigate whether our deep learning models could identify distinct neural signatures

associated with cognitive states of recognition and to engineer models that generalize across different participants. This research aligns with the broader aims of the DARPA-sponsored project, NEAT, which seeks to create neural decoders that can detect early symptoms of depression, psychosis, and suicidality. The hope is that neural decoders like this will enable clinicians to initiate timely interventions for individuals with these conditions, potentially altering the trajectory of their mental health [Morgan].

The development of passive BCIs — those that detect various user states without requiring active input or conscious attention from the user — is well-established [Nijboer et al., b; Angrisani et al.]. For example, passive BCIs can be used to detect cognitive states such as attention, and has shown promise using both EEG and ECoG technologies [Thiery et al.; De Taillez et al.]. The success of these methods has even paved the way for commercial applications, such as Neurable's EEG headphones, which monitor the user's focus [Neurable]. Research has expanded into other cognitive domains as well, including emotion, cognitive load, working memory load, and fatigue [Liang et al.; Yoo et al.; Żygierewicz et al.; Zeng et al.]. Despite these advancements, the decoding of mental health states such as depression remains relatively underexplored [Akbari et al.; García-Ponsoda et al.]. Significant work is still required to refine these models and integrate them into clinical mental health settings.

As discussed throughout this dissertation, a major barrier to the deployment of BCIs in real-world applications is the lack of cross-participant generalizability in neural decoders. This issue is particularly pronounced in passive BCIs and in the development of systems aimed at monitoring mental health. In this chapter, we will further explore the concept of generalizability. While deep learning models are advantageous for their ability to discern complex patterns in data, they also require substantial datasets to train effectively. This poses a significant challenge for BCIs, as acquiring neural data is both difficult and costly. One strategy to mitigate this issue is through data augmentation, a technique that enhances the size and diversity of a dataset by applying various transformations to existing data [Shorten and Khoshgoftaar]. For example, an image of a cat can be transformed into multiple variations by applying random crops, rotations, and color adjustments, thereby artificially expanding the dataset. In the case of image data, applying useful transformations is relatively straightforward and can be verified with visual inspection. However, the adaptation of similar techniques to time-series data, such as EEG or ECoG recordings, is less intuitive [Wen et al.]. Previous studies on data augmentation for EEG data have indicated that the effectiveness of specific

**Figure 4.1:** Model architecture, experimental paradigm, and data augmentation strategies for developing generalized EEG decoders for cognitive tasks. A) Comparison between the original HTNet architecture (top) and the revised architecture proposed in this chapter (bottom). The revised model includes a re-positioned Hilbert transform layer after the depthwise convolution and an additional skip connection around the Hilbert layer. B) Illustration of the verbal memory task used for training our models, comprising four stages: encoding, free recall, lexical decision, and recognition. Figure adapted from Marquardt et al.. C) Examples of the six data augmentation methods shown on an EEG channel, including four "random transformation" methods and two "pattern mixing" methods.

augmentations can vary significantly depending on the task at hand, but overall, augmentation is beneficial for improving model performance [Lashgari et al.]. In light of these findings, this work also investigates the use of data augmentation to enhance both the accuracy and generalizability of our models, aiming to make significant strides toward the practical application of BCIs for mental health settings.

## 4.2 Methods

### 4.2.1 Dataset

We used data from 20 participants performing a verbal memory task to train our machine learning models, leveraging previously published data from Marquardt et al.. For a detailed description of the experiment, readers are referred to the original publication, but we summarize here. The task comprised four distinct phases: an encoding phase, free recall phase, lexical decision phase and recognition phase (see Figure 4.1

B). In the encoding phase, participants were presented with forty different words and prompted to respond to stimuli. In the free recall phase, participants were asked to recall as many words as possible from the encoding phase. In the lexical decision-making phase, participants determined if presented words were real or not (including twenty words from the encoding phase, twenty new words, and twenty non-words). In the last stage, the recognition phase, participants indicated if presented words were new or previously encoded. We focused our analysis on the recognition phase, specifically on trials where participants correctly identified new or old words. This phase yielded approximately 80 trials per participant. EEG recordings were collected throughout the experiment using a 128-channel BioSemi headcap at a sampling rate of 256 Hz.

For data processing, we initially synchronized the trials to the stimulus onset and extracted windows from 200ms before to 1000ms after stimulus onset. We then applied standard EEG pre-processing techniques, which involved notch filtering to remove line noise at 60Hz, down-sampling to 125Hz, re-referencing, and normalizing each channel. Further details regarding our pre-processing pipeline can be found in Bigdely-Shamlo et al.; Mullen et al.. Additionally, we used the automatic artifact removal methods outlined in [Chang et al., a].

### 4.2.2 Machine Learning Models

We conducted a comparative analysis of four machine learning models, including three deep learning approaches. Our baseline model utilized random forest with 100 estimators and no maximum depth. In addition, we evaluated three deep learning models: EEGNet, HTNet (Orig HTNet), and a revised version of HTNet (Rev HTNet) designed specifically for EEG analysis. The EEGNet model and HTNet models were initially developed for decoding EEG and ECoG data, respectively, with detailed descriptions available in [Lawhern et al.] and [Peterson et al., b] (also see Chapter 3).

We made the following changes to HTNet from the original [Peterson et al., b]. Unlike the original HTNet paper, we did not require an electrode projection step because all participants used the same EEG caps, ensuring consistent electrode placement. For the Rev HTNet model, we made architectural and training adjustments to enhance performance with EEG data containing event-related potentials (ERPs). Specifically, we re-positioned the Hilbert transform layer after the depthwise convolution and introduced a skip

connection around the Hilbert layer (see Figure 4.1 A). These modifications were motivated by the need to incorporate spatial filter information extracted by the depthwise convolution into the Hilbert transformation, in addition to the temporal filters provided by the temporal convolution. The skip connection was included to enable the model to leverage spatiotemporal features from both the temporal and depthwise convolutions in both the frequency and time domains. Furthermore, we switched the optimizer for the Rev HTNet model from regular Adam to AdamW, which includes weight decay in the optimization process. We also implemented an exponential decay for the learning rate in this model. Additionally, we adjusted the kernel length of the temporal convolution to 64, set the F1 and F2 parameters to 6 and 2, respectively, and increased the batch size to 64 for the Rev HTNet model. Other parameters of model training were consistent across the three deep learning models, including training for 50 epochs with early stopping patience of 5 epochs, a learning rate of 0.001, and binary cross-entropy as the loss function. Finally, we assessed model performance of all models on the ROC AUC score (Area Under the Receiver Operating Characteristic Curve) on the test set.

**Within-Participant Training**

In a subset of our analyses, we trained models on one participant and tested on data reserved for that same participant, akin to the tailored models discussed in Chapter 3. To prevent bias towards the dominant class in the training data, we balanced the number of classes in the training set. Each participant's data was randomly divided into 80% for training, 10% for validation, and 10% for testing. We repeated this process three times for each participant to approximate the distribution of model performance.

**Cross-Participant Training**

To assess the generalizability of our models, we developed models that were trained and tested across participants. In this training scheme, we trained the model on a subset of participants and tested it on held out participants. We used a k-fold cross-validation scheme to split participants into training, validation and test data. Consequently, the training set comprised $\frac{7}{8}$ of the participants, while the remaining $\frac{1}{8}$ were evenly split between validation and testing. Our models were trained over 8 folds, with each fold executed 3 times to capture the range of model performance.

### 4.2.3   Data Augmentation

Deep learning methods are notoriously data hungry. However, collecting substantial amounts of neural data is often time and cost prohibitive. In our dataset, we are limited to approximately 80 trials per participant, which is typically insufficient for training large deep learning models. To address this limitation, we turned to data augmentation as a means to artificially expand our dataset for model training. We employed six different data augmentation methods, known to be effective with time series data [Lashgari et al.; He et al.; Yang and Desell; Yang et al.] (see examples of the data augmentations in Figure 4.1 C). Among these methods, four fall into the category of random transformations — jitter, magnitude warping, scaling, and window warping [Lashgari et al.]. The remaining two methods we classified as pattern mixing techniques — cutmix and mixup [Yun et al.; Zhang et al., a]. For each augmentation method, we tripled the amount of data available for training. We also exclusively trained cross-participant models to maximize the training data and improve generalizability.

**Augmentation Category: Random Transformation**

Our first augmentation method in the category of random transformations is scaling. This method aims to modify the magnitude of the EEG signals by either reducing or amplifying the signal strength. Within the context of EEG data, an increase in signal power might suggest a more pronounced response to stimuli, while a decrease could indicate a less prominent reaction. In our implementation, the same scaling factor is uniformly applied to all channels within each trial. We determined this scaling factor by randomly sampling from a Gaussian distribution with a mean of 1 and a standard deviation of 0.1.

The next data augmentation technique we examined is jitter, which involves injecting small amounts of additional noise into the signal. This method is beneficial for enhancing the model's robustness to various sources of noise inherent in EEG data [Hassani and Karami; Piastra et al.]. By introducing controlled noise during training, the model becomes more adept at handling real-world noise scenarios. In this approach, we generated a randomly sampled matrix, mirroring the shape of our EEG data (trials x channels x time), from a Gaussian distribution with a mean of 0 and a standard deviation set to 20% of the standard deviation of the trial data. We then added the randomized matrix to the original trial.

Our next data augmentation technique was magnitude warping, a method akin to scaling but with a twist.

Unlike scaling, where the scaling factor is uniformly applied across the entire trial time, magnitude warping introduces variability by applying different scaling factors to different parts of the signal. For example, the first 200ms of the signal may be amplified, while the rest may be decreased. To implement this, we first sampled scaling factors from a Gaussian distribution with a mean of 1 and a standard deviation of 0.1. We then randomly determined the number of inflection points in the warped scaling by sampling from a uniform distribution between 10 and 15. These inflection points dictate the number of times the scaling changes direction. Next, we used the CubicSpline function from the sklearn library to interpolate the warped scaling, which is then multiplied with the original signal. In our implementation, a unique warped scale is applied to each channel of the data.

Our final random transformation technique is window warping, which involves stretching and shrinking the signal in the time domain rather than altering its magnitude, as in magnitude warping. This data augmentation simulates variations in a participant's reaction time to stimuli, ideally providing the model with valuable insights into response variability. Similar to the magnitude warping technique, we generated scaling factors from a Gaussian distribution, randomly sampled the number of inflection points, and used CubicSpline for interpolating the warped scale. The main difference lies in the application of the warped scale in the time domain instead of on channel magnitude. In our approach, each channel undergoes a distinct randomly generated warp scale.

**Augmentation Category: Pattern Mixing**

CutMix is a data augmentation technique for training deep learning models, particularly effective for improving the performance and robustness of image classification models [Yun et al.; Yang and Desell]. The core idea of CutMix is to create new training samples by combining pairs of samples and their corresponding labels in a novel way. Specifically for image data, CutMix selects a random patch from one image and pastes it over a corresponding patch in another image. The ground truth labels for these new images are then mixed proportionally to the area of the patches contributed by each original image. For instance, if 70% of the pixels in the new image come from an image labeled "cat" and 30% from an image labeled "dog", the target label would be a blend of 70% "cat" and 30% "dog". This approach not only diversifies the training data but also encourages the model to focus on less dominant features of the classes, thereby improving
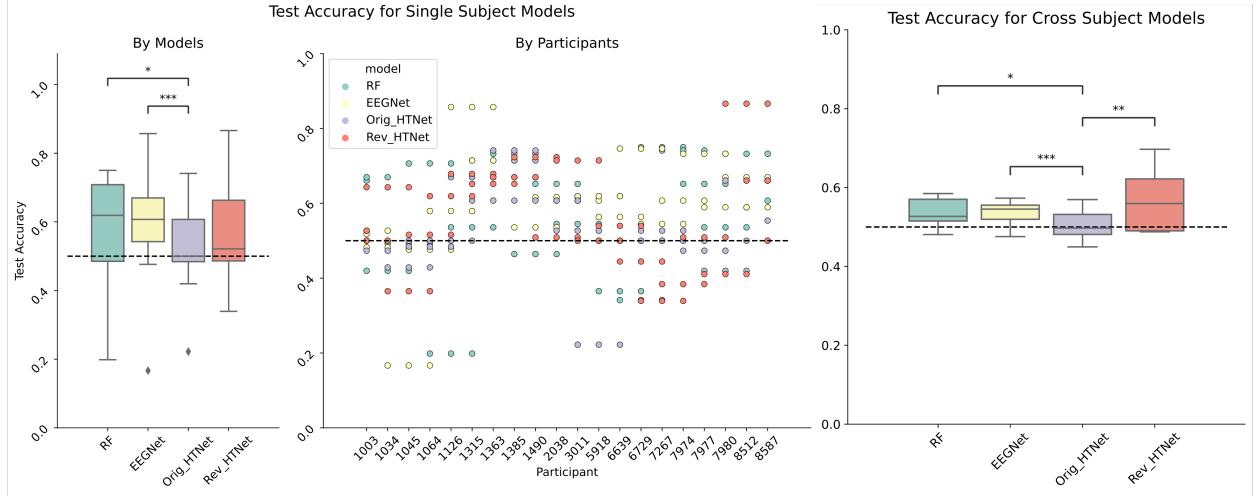
its generalization capabilities. In our approach, instead of combining images, we combine two EEG trials. The trials can be merged either in the channel domain or the time domain, with each new sample randomly selected to be merged in one domain or the other. When the cut occurs in the time domain, a random subset of contiguous time from a randomly selected trial is incorporated into the original trial. If the merge occurs in the channel domain, a random subset of channels from a random trial is swapped into the original trial. In both cases, the timing and channel positions remain consistent. To determine the proportion of data merged from the second trial, we randomly sample from a uniform distribution between 0.05 and 0.95. To ensure the generated data is meaningful, we only merge trials from the same participant with the same labels.

MixUp is another common technique used to improve the generalization of machine learning models, especially in tasks like image classification [Zhang et al., a; Yang et al.]. The method works by blending two images and their corresponding labels to generate new training samples. For image data, two images are taken and combined together using a weighted average, where the weight is randomly sampled from a distribution. In our application to EEG data, the weighted average was randomly sampled from a uniform distribution between 0.05 and 0.95. This weight determines the proportion of the original trial that remains in the mixed signal. We applied the same weighted average consistently across both the time and channel domains for each newly generated trial. Similar to our approach with CutMix, we also only mix trials from the same participant with the same labels.

## 4.3    Results

### 4.3.1    Within-Participant Models

In this section, we report the results from the within-participant models, as detailed in the first two panels of Figure 4.2. All test accuracies are quantified using ROC AUC metrics. Among the evaluated models, EEGNet achieves the highest mean test accuracy at 59.6%. The random forest model follows with a mean test accuracy of 57.5%, while the revised HTNet (Rev HTNet) model shows a mean accuracy of 56.0%. The original HTNet (Orig HTNet) model shows the lowest mean accuracy of 52.9%. Notably, significant differences in performance are observed between the random forest and Orig HTNet models (p-value: 0.02225), as well as between EEGNet and Orig HTNet (p-value: 0.0001431). Of the participants, 1363, 1385, 1490,

**Figure 4.2:** Test Accuracy (ROC AUC) for Within-Participant and Cross-Participant Models. Panel 1: Displays a boxplot of test accuracy for each model trained within participants. Stars above the boxplots denote significant differences in model performance. Panel 2: Displays a scatter plot detailing the test accuracy of all four models, segmented by individual participant. Panel 3: Displays a boxplot of the test accuracy of the different models trained across participants. This panel does not provide a participant-specific breakdown, as the test data encompasses several participants.
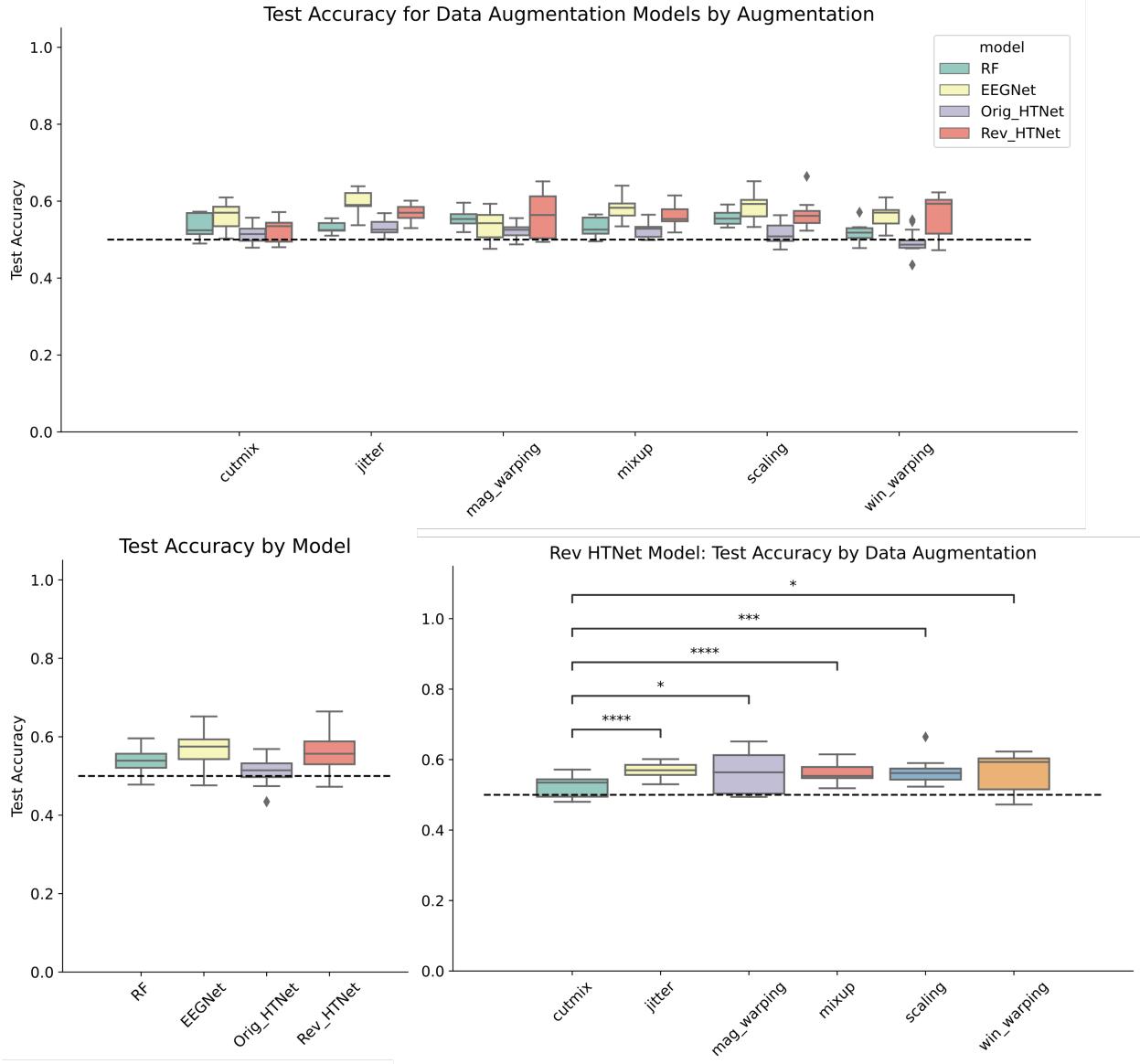
and 8587 consistently demonstrate the best average performance across all models. The robust performance of the random forest models and EEGNet can likely be attributed to their effectiveness in handling the limited data typical of within-participant model training.

### 4.3.2 Cross-Participant Models

In this section, we report the results for the cross-participant models, as depicted in the last panel of Figure 4.2. Among the cross-participant models, the revised HTNet (Rev HTNet) model achieves the highest mean test accuracy at 56.7%. The EEGNet model demonstrates the next best mean accuracy at 53.9%, while the random forest model and the original HTNet (Orig HTNet) model exhibit mean test accuracies of 53.6% and 50.6%, respectively. Significant differences in performance are observed between the random forest and Orig HTNet models (p-value: 0.01211), EEGNet and Orig HTNet (p-value: 0.0007353), as well as Rev HTNet and Orig HTNet (p-value: 0.001145). These results combined with the within-participant results suggest that while the original HTNet model may not be well-suited for low-data scenarios with EEG data, the revised architecture demonstrates strong generalization capabilities in such settings.
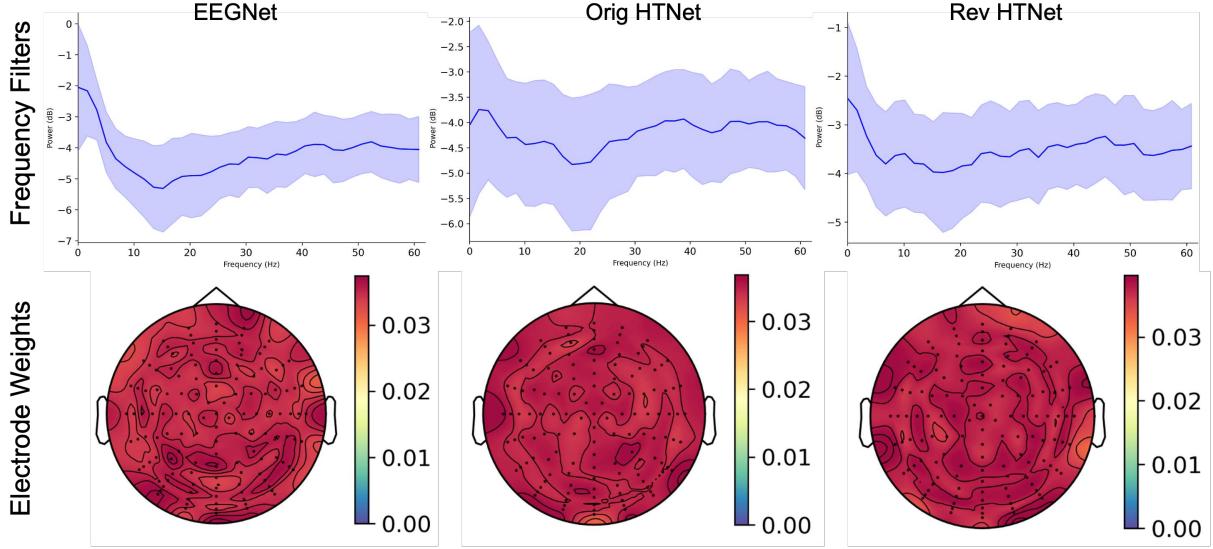
### 4.3.3 Data Augmentation Models

In this section, we report the results of six data augmentation methods applied to cross-participant models, each utilizing a data augmentation factor of three. The results, summarized in Figure 4.3, suggest that data augmentation has a marginal impact on test accuracy. Among the augmentation techniques, jitter stands out with the highest average test accuracy across models at 55.7%. The other augmentation methods yield comparable results on average across the models (CutMix = 53.4%, magnitude warping = 54.2%, MixUp = 54.9%, scaling = 55.5% and window warping = 53.4%). Comparing these results to the baseline cross-participant models without data augmentation, we note minimal differences in performance (random forest average test accuracy = 53.8%, delta from no data augmentation model = +0.02%; EEGNet = 57.0%, delta = +3.1%; Orig HTNet = 51.6%, delta = 1.0%; Rev HTNet = 55.7%, delta = -1.0%). These results indicate that data augmentation helped the EEGNet model generalize the best. EEGNet benefits the most from jitter, achieving a significant boost in accuracy to 59.7%, the highest overall for any combination of model and data augmentation. Other methods also enhance EEGNet's performance (CutMix = 56.2%, magnitude warping = 53.6%, MixUp = 57.7%, scaling = 58.3% and window warping = 56.3%). Conversely, the random forest model shows negligible change across all augmentation techniques, maintaining a performance level similar to baseline cross-participant models (CutMix = 53.4%, jitter = 53.2%, magnitude warping = 55.4%, MixUp = 53.2%, scaling = 55.8% and window warping = 51.9%). The Orig HTNet model continues to underperform, indicating that additional data does not significantly enhance this architecture's effectiveness (CutMix = 51.6%, jitter = 53.0%, magnitude warping = 52.1%, MixUp = 52.6%, scaling = 51.7% and window warping = 49.0%). In contrast, the Rev HTNet model exhibits a varied response to different augmentations. Notably, CutMix shows the weakest result at 52.6%, significantly lagging behind the other methods, whereas jitter leads with a performance of 57.1% (see bottom right panel in Figure 4.3). All other data augmentations show comparable performance in Rev HTNet (magnitude warping = 55.8%, MixUp = 56.4%, scaling = 56.1% and window warping = 56.5%). These findings underscore that while data augmentation can slightly enhance model generalization, particularly with jitter, the overall impact on model performance is modest.
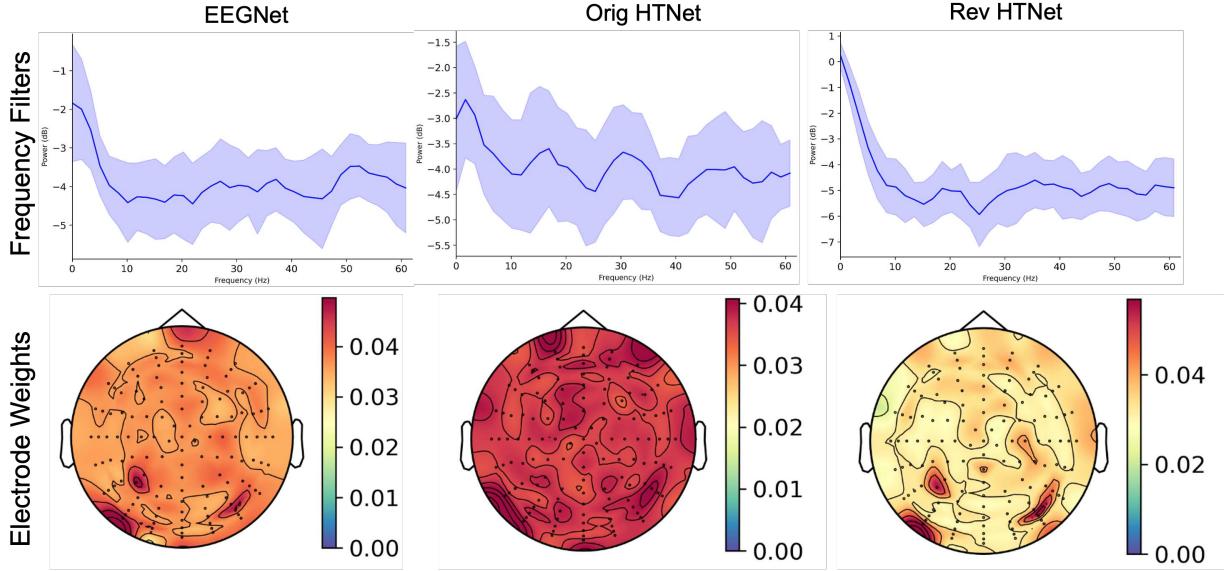
**Figure 4.3:** Test Accuracy (ROC AUC) for Data Augmentation Models. The top panel shows test accuracy for all six data augmentation methods, categorized by model type. The bottom left panel displays the average test accuracy across all data augmentations for each model. The bottom right panel showcases test accuracy specifically for the Rev HTNet model, with stars denoting significant differences in test accuracy between data augmentation methods in this model.

**Figure 4.4:** Model Interpretability in Within-Participant and Cross-Participant Models. We visualize the learned frequency filters and electrode weights from the three deep learning models. Panel A displays results from within-participant models. Panel B shows the results from cross-participant models. Note that we focus on EEGNet, Orig HTNet, and Rev HTNet models, as their convolution layers correspond to the frequency filters and electrodes under analysis.

### 4.3.4 Model Interpretability

In this section, we apply the same model interpretability techniques outlined in Peterson et al. [b] (Chapter 3). In brief, the temporal convolution captures temporal data features, while the depthwise convolution learns spatial filters across the EEG electrodes. By analyzing the weights and outputs of these layers, we can gain insight into the model's decision-making process. To examine the frequency filters, we first train the deep learning models and then pass white noise through the temporal filter. This process reveals the frequencies that the model prioritizes. For the electrode weights, we extract the weights from the depthwise convolution, average them across the layer's filters, and then map them onto EEG electrodes. Our analyses involve averaging all trained models within each experiment type. For example, in the within-participant results, we average the filters and weights across all participants and training folds.

For within-participant models, we observe minimal discrepancies in the filters and weights learned by the models (refer to Figure 4.4 A). In the frequency filters, EEGNet emphasizes frequencies below 10Hz, reduces weight between 10-20Hz, and maintains a relatively consistent weighting from 20-60Hz. Rev HT-Net exhibits similar frequency filters but with a less pronounced dip between 10-20Hz. Conversely, Orig HTNet showcases somewhat distinct frequency filters, featuring a peak around 5Hz and a dip around 20Hz. Regarding electrode weights, all three models display similar weightings, without any particular regions standing out as significantly salient. This phenomenon could be attributed to the fact that the weight plots are averaged across all participants, potentially diluting participant-specific weightings.

In contrast, cross-participant models reveal more noticeable differences between models. In EEGNet, there's a peak in frequencies below 10Hz and another small peak around 50Hz. We also see that EEGNet learns some salient electrodes for decoding, particularly around the left occipital lobe, left and right parietal, and central frontal lobe. Orig HTNet demonstrates more interesting frequency filters, with peaks around 5Hz, 15Hz, and 30Hz. However, this model does not exhibit clearly salient electrode weights like the other two models. The divergent frequency filters and electrode weights in Orig HTNet may explain its comparatively weaker performance in these experiments. Rev HTNet showcases a frequency filter similar to that of EEGNet, emphasizing frequencies below 10Hz and exhibiting a small peak around 20Hz. Similarly, its electrode mapping mirrors EEGNet, but with even greater weight disparity on salient electrode locations (left and right occipital, and left and right parietal). These findings suggest that establishing a robust electrode
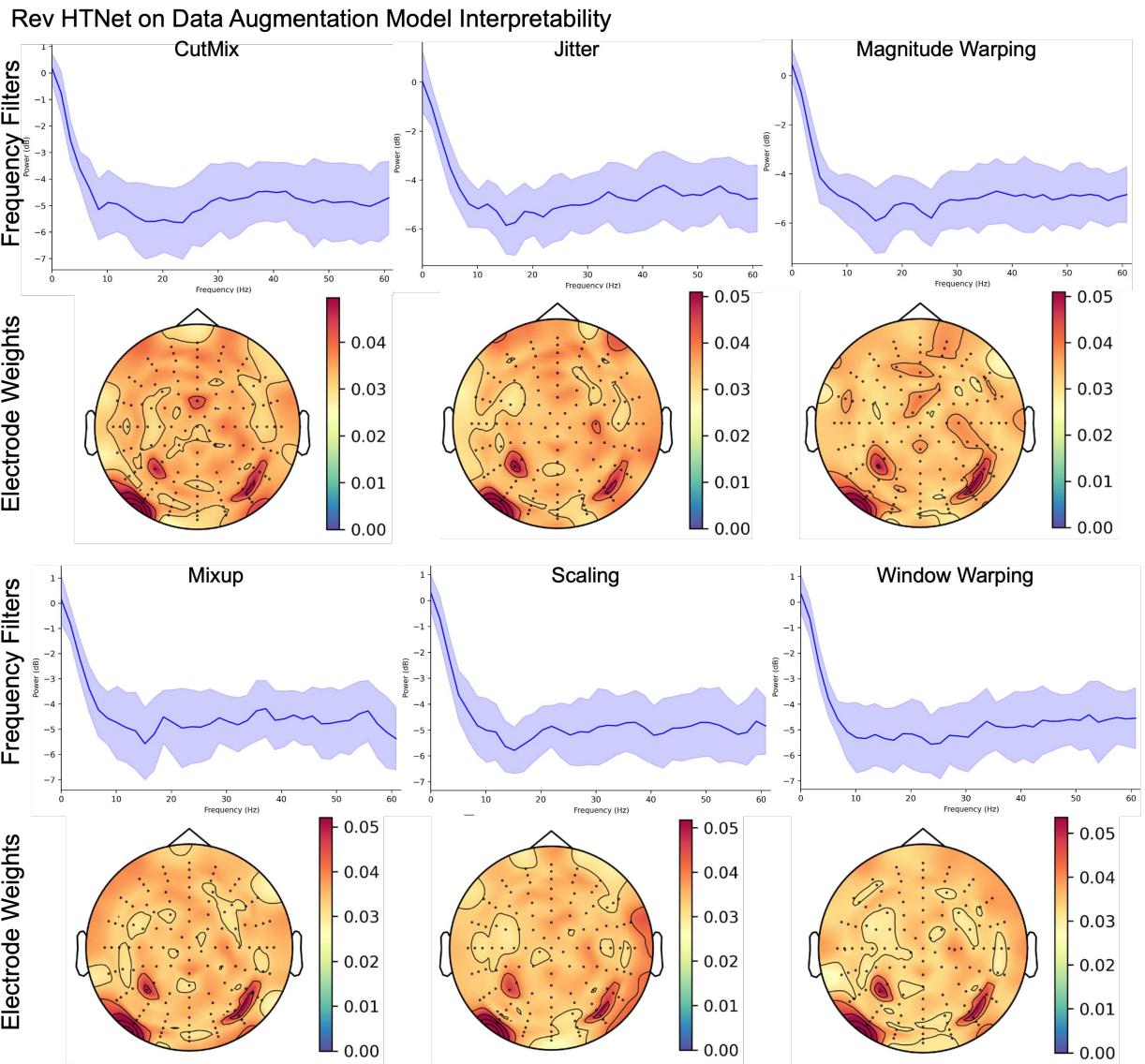
weighting, where certain electrodes are more prominent, might be crucial for decoding cognitive tasks.

Lastly, we explore model interpretability of the Rev HTNet results with data augmentation. Overall, we observe that the models across all data augmentations exhibit similar frequency filters and electrode mappings. Broadly, these models emphasize frequencies below 10Hz, followed by a relatively consistent response across frequencies up to 60Hz. The models also identify the same electrode locations as the cross-participant Rev HTNet model, specifically the left occipital and left and right parietal regions. However, some deviations are noted. The CutMix model displays a small peak around 40Hz in the frequency filter and an additional electrode location near Cz, potentially explaining the significantly poorer performance of the CutMix models. On the other hand, the jitter model reveals additional electrode locations of interest, one near Cz4 and another in the right frontal lobe, which could explain why models trained with jitter showed superior generalizability.

## 4.4 Discussion

Our results underscore the limitations of the original HTNet model proposed in Chapter 3 [Peterson et al., b]. The architecture updates proposed in this chapter demonstrated clear improvements in decoding EEG data for cognitive tasks. However, it remains unclear whether these enhancements would generalize to other modalities beyond EEG or tasks beyond cognitive domains. Since the original HTNet model was designed with ECoG data in mind, future investigations should explore whether the revised HTNet architecture would outperform the original in cognitive tasks recorded with ECoG data or in other ECoG scenarios.

In our exploration of data augmentation, we were surprised to find that the jitter method yielded the best overall performance. We initially expected more sophisticated methods like CutMix and MixUp to generate more realistic synthetic data, thereby enhancing model generalizability by increasing robustness to data variations. However, it appears that the jitter method provided the most beneficial synthetic data for model training and generalization. This could be due to the fact that neural data is inherently noisy [Hassani and Karami; Piastra et al.], and the jitter method naturally introduces substantial noise into the signal. This likely enhanced the model's resilience to noise, a common challenge in EEG data analysis. It is also possible that CutMix and MixUp did not perform as well due to the extent of data mixing. For example, in our MixUp approach we choose the percentage of the original trial from a uniform distribution

**Figure 4.5:** Model interpretability on Rev HTNet with data augmentation. Plots are categorized by the data augmentation methods, displaying frequency filters on the top and electrode weights below.

between 0.05 and 0.95, rather than a Gaussian distribution. The choice to use a uniform distribution could result in trials that are only minimally altered from the original, potentially limiting the effectiveness of the augmentation. Given that we expect other data augmentation methods to also show improvements, we may find that combining different augmentations may result in the best model improvements. Exploring combinations of data augmentation methods and optimizing their hyperparameters could lead to further improvements in model performance.

Our interpretability results revealed that the models showing the best performance tended to emphasize similar electrode locations, particularly around the left occipital and parietal lobes. The occipital lobe is crucial for visual processing, while the parietal lobe is involved in sensory processing [Anderson et al., a]. This suggests that our models may be learning to differentiate classes based on visual and sensory processing times or other factors related to the experimental setup. Additionally, the Rev HTNet model with jitter augmentation, which demonstrated high performance, also learned an additional electrode weighting on C4 and the right frontal electrodes. This may indicate that models trained with the jitter augmentation are capturing additional, task-relevant information. Processing from the frontal lobe is associated with higher-level cognitive functions such as memory [Anderson et al., a; Chayer and Freedman], suggesting that this model variant is effectively leveraging information from this region of the brain.

## 4.5   Summary

Our findings demonstrate the potential to create models that generalize across participants in cognitive tasks. By enhancing our original HTNet model, we devised an architecture better suited for generalization on EEG data. Surprisingly, augmenting the training data did not significantly enhance model performance, although it did improve the generalizability of the EEGNet model. Despite these advancements, there remains room for improvement to make these models viable in real-world applications. These results underscore the ongoing need for the development of more generalizable models in BCI applications, as well as the necessity for data augmentation methods tailored for neural data. Additionally, there is a pressing need to identify a universally adaptable model architecture capable of accommodating variability across users and tasks in BCIs.

## 4.6   Acknowledgements

# Chapter 5

# Neural Manifolds of Human Intracranial Recordings During Naturalistic Arm Movements

In this chapter, we switch our focus from the models used for neural decoding, to the contexts where the neural data was collected. The primary intent of this chapter is to further understand naturalistic neural data and how neural activity during natural movement changes between individuals, and from experimental contexts. In particular, we explore the naturalistic ECoG dataset shown in Chapter 3. This chapter includes materials in preparation for submission to the Journal of Neural Engineering. Please check future publications for updated versions of this chapter.

## 5.1 Introduction

Understanding how neural computations control movement behavior is critical to advancing many fields within neuroscience, including the development of Brain-Computer Interfaces (BCIs). The majority of existing data from neural recordings during movement have focused on data collected during controlled lab experiments on simple directed tasks. However, understanding how neural activity drives movement during naturalistic behavior is essential for expanding these findings outside of the lab. Naturalistic movements,

characterized by their unconstrained, self-generated, and spontaneous nature, are often underexplored because of the additional complexity provided by the various contexts and concurrent behaviors involved. For example, naturalistic movements can cover behaviors ranging from anxiously shaking hands with a new boss, to absentmindedly stretching while working. The variety of behaviors and mental states in natural movements presents significant challenges in understanding and modeling neural computation, especially compared to the cued and constrained behaviors typically studied in experimental settings. Nonetheless, understanding how the brain generates naturalistic behaviors will enhance the generalizability and applicability of research findings and bring BCIs closer to everyday practical use.
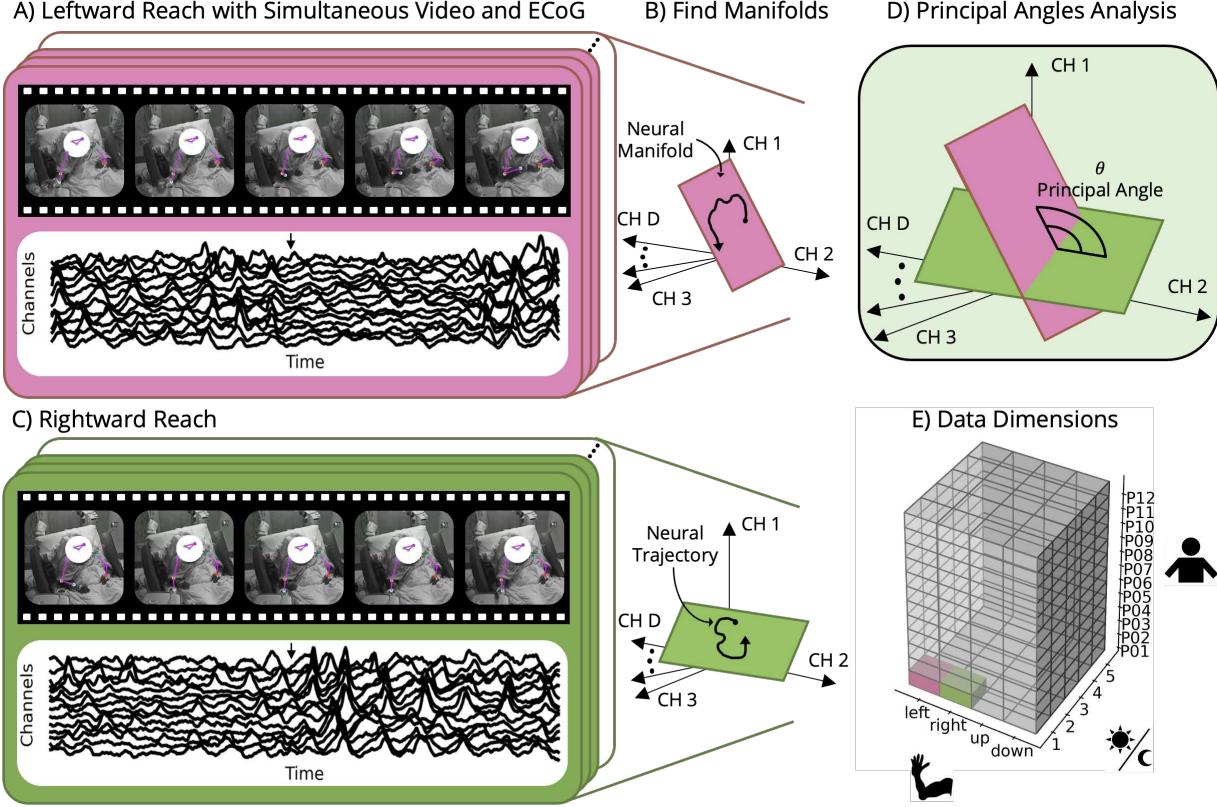
Recent innovations in large-scale neural recordings with electrode arrays is enabling deeper insights into understanding the neural computations underlying movements. Because such large-scale recordings can capture thousands of neurons at once, either through single cell recordings or broader recordings such as EEG, researchers can start to develop theories about the various neural populations involved in behavior. Although these neural recordings encompass thousands of possible dimensions of neural activity, many analyses have found that the neural activity across all these neurons is surprisingly quite low-dimensional [Gallego et al., b; Natraj et al., b; Sadtler et al.; Chung and Abbott; Cunningham and Yu; Yu et al., 2008]. This has opened a new line of research that uses *neural manifolds* to investigate the low-dimensional neural activity within these large-scale neural recordings, to both define which sub-populations of neural activity are most related to the behavior of interest, and to uncover the temporal structures that drive behavior in these sub-populations [Chung and Abbott; Gallego et al., b].

Many studies analyze neural manifolds during simple behaviors in experimental settings, specifically motor behaviors, using micro-array recordings of non-human primates and rodents [Gallego et al., b; Chung and Abbott]. This body of research on low-dimensional neural manifolds has yielded valuable insights into how the brain accomplishes movement control [Abbaspourazad et al.; Ahrens et al.; Bruno et al.; Churchland et al., b,c,a; Churchland and Shenoy; Elsayed et al.; Gallego et al., c; Shenoy et al.; Kaufman et al., a,b; Michaels et al.; Overduin et al.; Sadtler et al.; Santhanam et al.; Sussillo et al., a; Sun et al., b]. Manifold analysis has also been important for understanding motor learning and building neural decoders. Sadtler et al. and Golub et al. found that motor learning with a BCI that used activity from the "intrinsic" manifold for cursor control was more easily learned than with a BCI that used neural activity outside the intrinsic

manifold. As it becomes more established that low-dimensional neural manifolds are related to the control of movement generation of experiment-driven movements, more researchers have started to use neural activity within low-dimensional manifolds for BCI control [Gallego et al., a; Natraj et al., a; Bashford et al.; Farshchian et al.; Pandarinath et al.]. For example, Gallego et al. [a] used Canonical Correlation Analysis (CCA) to align the latent dynamics within the neural manifold for a monkey on sessions spanning many days, and found that they could realign the decoder even 700 days after the initial recording session of one monkey. Other work has also shown that decoding based on manifold alignment can help align decoders across participants [Safaie et al.] and even across species [Rizzoglio et al.].

While most of this work focused on neural recordings from non-human primates, some work has also shown the existence of neural manifolds in human recordings [Natraj et al., b,a; Rizzoglio et al.]. Natraj et al. [b] analyzed neural manifolds of micro-electrocorticography (micro-ECoG) data from humans performing various hand gestures. They found that (1) the neural activity related to movement did occupy low-dimensional neural manifolds and (2) the temporal structures of the neural activity were constrained to sub-components of the manifold based on the movement activity. In a follow-up paper, Natraj et al. [a] used manifold alignment to perform long-term decoder alignment for another micro-ECoG human participant; they were able to realign the BCI for up to 7 months after the initial recordings. In a different line of work, Rizzoglio et al. used the latent dynamics from microelectrode recordings in monkeys to successfully align a decoder to the latent dynamics from microelectrode recordings in a human participant with tetraplegia.

This previous work established the existence and usefulness of neural manifolds for the control of movements in controlled experiments. However, it remains unknown whether low-dimensional neural manifolds persist during more complex natural movements, especially in humans, or if they are a consequence of the simple tasks used in experimental settings Gao and Ganguli [2015]. Some researchers have begun exploring the neural manifolds of naturalistic movements [Abbaspourazad et al.; Altan et al.; Melbaum et al.]. Abbaspourazad et al. found that one principal mode within the neural manifold of naturalistic reaches in monkeys was predictive of their movements. Altan et al. also found low-dimensional manifolds in monkeys performing unconstrained movements and showed that the activity from the manifold was just as useful for decoding as was all neural activity. Melbaum et al. used manifold alignment decoding for cross-subject and cross-session generalization in freely moving rats, indicating the usefulness of manifold-aligned BCIs for

**Figure 5.1: Neural Manifold Analysis of Naturalistic Movements in Humans.** We extract and analyze linear approximations of neural manifolds, i.e. neural sub-spaces, in a dataset containing simultaneous ECoG and video data of naturalistic arm movements. The dataset comes from epilepsy patients who underwent clinical monitoring for five days at Harborview Medical Center in Seattle, WA, USA. **A)** An example of video and ECoG data from a leftward reach using the right arm by participant P01, who, in this case, grabbed an object from their bedside table. Video frames are overlaid with estimates of the joint coordinates of the participant's upper body, with colored dots indicating joint locations and pink lines showing the connecting limbs. Joint coordinates were extracted using DeepLabCut. For additional detail on how joint coordinates were extracted, see Singh et al. [b]. Example ECoG data below video frames shows a sample of 14 ECoG channels from 1 second before and 1 second after movement initiation (indicated with arrow). **B)** Using the estimated joint-coordinate data from video, we grouped all leftward reaches from participant P01 and used PCA to extract an estimate of the neural sub-space (shown in pink as a simplified 2D plane) from the ECoG data on these movements. **C)** An example of simultaneous video and ECoG during a rightward reach in the same participant, who, in this case, placed an object back on the bedside table. The same pipeline is applied to rightward reaches to extract the rightward sub-space, i.e., grouping right reaches and using PCA for the manifold estimate. Each individual reach trial can be visualized on the neural sub-space as a trajectory along the sub-space, as shown by the black arrows on the pink and green 2D planes. **D)** After using PCA to obtain estimates of the neural manifolds for each type of movement, we use Principal Angles Analysis (PAA) to calculate the distance between movement sub-spaces. **E)** We analyze data from 4 movement types spanning 12 different participants over the course of 5 days during their hospital stay. This naturalistic movement dataset is visualized here as a data tensor. We split the data into four different movement categories: left, right, up and down. The green and pink cubes in the data tensor depict the data used for the example leftward reach and rightward reach sub-spaces.

78

naturalistic movements. Whether similar results hold for neural manifolds extracted from human naturalistic movements has remained an open question.

In this paper, we ask how does the *human brain* flexibly support a diverse repertoire of movements during naturalistic everyday tasks? To this end, we analyze the structure of different neural manifolds calculated from human ECoG data recorded during spontaneous, uninstructed natural arm movements (Figure 5.1). We evaluated simultaneous ECoG and video data from twelve epilepsy patients while they underwent clinical monitoring for their epilepsy over the course of five days (see Figure 5.1A, C, E). To simplify the analysis and align it with commonly used center-out reaching tasks, we categorized the spontaneous, uninstructed natural arm movements into four classes (left, right, up, and down). We then use Principal Components Analysis (PCA) to extract linear estimates of the neural manifolds underlying the control of movement (see Figure 5.1B). Since we only calculate linear approximations of the manifolds, we henceforth refer to our manifold estimates more appropriately as "neural sub-spaces". Using Principal Angles Analysis (PAA), we quantify the alignment between neural sub-spaces of the different types of movements (see Figure 5.1D). If the neural activity during different spontaneously generated movements is similarly oriented, this could indicate that a "universal manifold" supports movement in both uninstructed and instructed movements [Gallego et al., b]. If not, the brain may instead employ another strategy to generate a diverse and flexible set of naturalistic movements.

We explore the question of how the brain flexibly supports naturalistic movement in four parts. First, we explore whether low-dimensional neural sub-spaces exist during naturalistic movements and evaluate how similarly oriented those sub-spaces are across different movements within each participant (Section Cross-Movement Sub-Space Comparisons). Second, we evaluate the alignment of neural sub-spaces for similar types of movements over the course of each participant's hospital stay (Section Cross-Days Sub-Space Comparisons). Third, we compare neural sub-spaces of similar movements across different participants in the dataset and evaluate their alignment (Section Cross-Participants Sub-Space Comparisons). Finally, we compare the naturalistic movement ECoG dataset to an ECoG dataset collected during experimentally-controlled movements to uncover how neural sub-spaces might change during different movement contexts (Section Comparing Experimentally-Controlled Sub-Spaces).

| Participant | Left | Right | Up | Down |
|---|---|---|---|---|
| P01 | 137 | 79 | 313 | 130 |
| P02 | 40 | 70 | 68 | 31 |
| P03 | 112 | 102 | 291 | 79 |
| P04 | 42 | 39 | 158 | 29 |
| P05 | 37 | 42 | 85 | 39 |
| P06 | 124 | 217 | 430 | 98 |
| P07 | 149 | 69 | 357 | 100 |
| P08 | 140 | 90 | 439 | 181 |
| P09 | 24 | 25 | 57 | 45 |
| P10 | 99 | 129 | 206 | 186 |
| P11 | 115 | 130 | 518 | 133 |
| P12 | 165 | 148 | 465 | 169 |

**Table 5.1:** Total number of movement instances per participant for each movement category across five recording days.

## 5.2 Methods and Materials
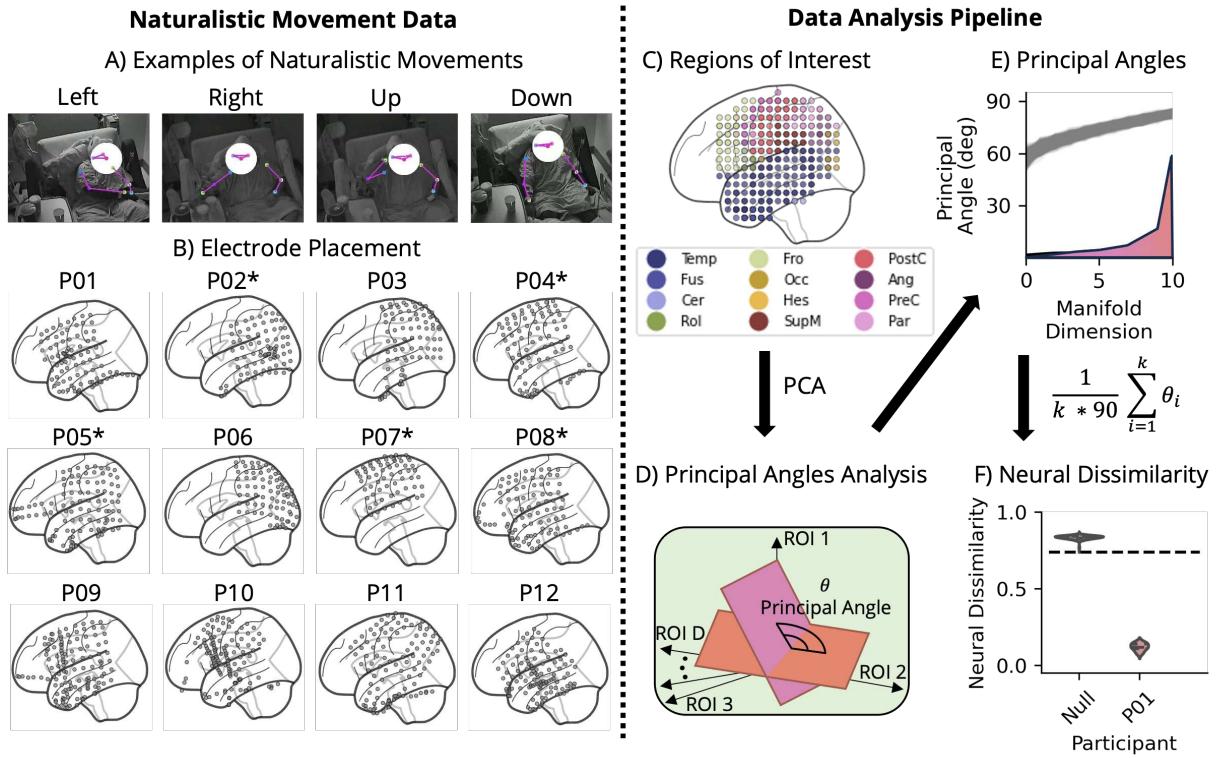
Code for this paper can be found at:

`https://github.com/BruntonUWBio/NaturalisticNeuralManifolds`.

The naturalistic movement dataset can be found at: `https://figshare.com/articles/dataset/`

`ECoG_and_arm_position_during_movement_and_rest/16599782`.

### 5.2.1 Naturalistic Movement Data

For the naturalistic movement dataset, we collected data from 12 human epilepsy patients at Harborview Medical Center in Seattle, WA [Wang et al., 2018; Peterson et al., 2021, 2022]. All participants were part of 24/7 clinical epilepsy monitoring, which included concurrent video and ECoG recordings (see Figure 5.1A and C). Participants included 8 males and 4 females and were aged $29 \pm 8$ years (mean $\pm$ SD). Electrodes were primarily implanted in the participant's left hemisphere (7 left, 5 right); see Figure 5.2B for the electrode placement of each participant. Some participants also received stereoelectroencephalography, i.e., penetrating electrodes into cortical and sub-cortical areas. We extracted data from 5 days of recording for each participant. Our study was approved by the University of Washington Institutional Review Board for the protection of human participants, and all participants provided written informed consent.

**Naturalistic Movement Data**

A) Examples of Naturalistic Movements

Left　　Right　　Up　　Down

B) Electrode Placement

P01　　P02*　　P03　　P04*

P05*　　P06　　P07*　　P08*

P09　　P10　　P11　　P12

**Data Analysis Pipeline**

C) Regions of Interest

Temp　Fro　PostC
Fus　Occ　Ang
Cer　Hes　PreC
Rol　SupM　Par

PCA

E) Principal Angles

Principal Angle (deg)

90
60
30

0　　5　　10

Manifold Dimension

$$\frac{1}{k*90}\sum_{i=1}^{k}\theta_i$$

D) Principal Angles Analysis

ROI 1
$\theta$
Principal Angle
ROI D
ROI 2
ROI 3

F) Neural Dissimilarity

Neural Dissimilarity

1.0
0.5
0.0

Null　P01

Participant

Figure 5.2: **We used PCA and PAA to uncover and analyze the alignment of neural sub-spaces in ECoG data during four types of naturally generated arm movements. A)** Video frame example of each type of naturally generated movement from participant P01 during their hospital stay for epilepsy monitoring. Colored dots indicate the estimated joint coordinate, and pink lines show the connecting limbs. We split the natural movements into four categories: left, up, right and downward reaches of the arm contralateral to the ECoG electrodes. The reach directions were based on the reach angle at the extent of the participant's reach. **B)** Electrode placement of all twelve participants collected in the naturalistic movement dataset. Participants with a * had their electrode placement on the right hemisphere; electrodes on the opposite hemisphere were flipped for clarity of visualization. **C)** Due to the variation in electrode coverage, we weight-averaged electrodes from both datasets onto 311 common ROIs that cover motor regions, the temporal lobe and parts of the frontal lobe [Peterson et al., b]. We show a cross-slice of the 311 ROIs, with colored dots indicating the region for each ROI centroid (Temp - Temporal, Fus - Fusiform, Cer - Cerebellum, Rol - Rolandic, Fro - Frontal, Occ - Occipital, Hes - Heschl, SupM - SupraMarginal, PostC - Postcentral, Ang - Angular, PreC - Precentral, Par - Parietal). **D)** After projecting the electrode space onto the 311 ROIs, we applied PCA to each movement type for each participant. Each PCA analysis produced an estimate of the neural manifold, where the neural activity primarily lies as a hyperplane in the ROI space (simplified pink and orange planes in the figure). For each principal component direction (starting with the direction of highest variance accounted for), the angle between the hyperplanes in that dimension can be calculated, producing the principal angle for that principal component. **E)** Using the top $k$ principal angles, we calculated the principal angles between the two neural manifold estimates. In this plot, we show possible principal angles between the pink and orange hyperplanes (pink/orange colored region). We compared the principal angles from the neural data to 1000 null samples generated with TME (grey region), which defined the upper bound of significance for the principal angles [Elsayed and Cunningham, 2017]. **F)** We then calculated the sum of all principal angles, up to our $k$ cut-off point, and normalized by the maximum possible sum of angles, generating our neural dissimilarity metric. This metric indicates how close the neural sub-spaces are to each other. We also calculated the neural dissimilarity of the null data generated via TME and used the 1% significance level as an upper bound for sub-space alignment.

**Movement Behavior Preprocessing**

Since our primary interest was in brain signals during naturalistic movements, we used the simultaneous video data to extract naturalistic upper body movements. We extracted movements from 24/7 video recorded at 30 frames per second in each patient's room for clinical monitoring purposes. Using markerless pose tracking via DeepLabCut [Mathis et al.], we estimated coordinates for key points like the wrist, elbow, shoulder, and nose. Movement events were identified using automated state segmentation on pose data, focusing on wrist movements of the hand contralateral to the electrodes occurring after at least 0.5 seconds of no movement. We also used video data to exclude private participant times from our analysis.

After processing the video data, we categorized the ECoG data into four movement classes (leftward, rightward, upward, and downward reaches) based on joint pose estimates. These classes broadly indicated the reach's target, such as the bedside table or face, simplified our later comparisons, and made the data more akin to center-out reaching tasks, for which the methods we used were developed. Although it would be ideal to understand the full spectrum of naturalistic movements, we lacked certain information, such as details about the stimuli participants were attending to, which constrained our ability to analyze the naturalistic behaviors more comprehensively. Our movement segmentation was based on the angle of the contralateral wrist at the furthest extent of the reach, categorizing movements into quadrants on a unit circle (left: 135 degrees to 225 degrees, right: 315-45 degrees, up: 45-135 degrees, down: 225-315 degrees). We extracted movement events over 5 days for each participant, but not every class was present every day, leading to some analyses excluding specific participant-day-class combinations. Examples of the four movements with overlaid joint pose estimates are shown in Figure 5.2A, and Table 5.1 shows the number of movement instances for each participant.

**ECoG Preprocessing**

For ECoG data preprocessing, we followed the steps from Peterson et al. [b, 2021], including DC drift removal, notch filtering (60Hz and harmonics), band-pass filtering (1–200 Hz), electrode re-referencing and normalization using MNE-Python [Gramfort, 2013]. We then downsampled the data to 250 Hz and extracted 2-second segments around each movement event. Electrode locations were determined using the Field-trip toolbox in Matlab and mapped into Montreal Neurological Institute (MNI) space [Stolk et al., 2018;

Oostenveld et al., 2011; Fonov et al., 2011]. To account for variations in electrode placement, and to facilitate comparisons across participants, we then mapped all electrodes onto 311 ROIs based on average density for each ROI using the technique in Peterson et al. [b]. The resulting 311 ROIs covered 12 regions of the AAL atlas [Rolls et al., 2020] (Angular, Cerebellum, Frontal, Fusiform, Heschl, Occipital, Parietal, Postcentral, Precentral, Rolandic, SupraMarginal, and Temporal); see Figure 5.2C. From the original electrode locations, we also evaluated the electrode overlap between participants using the technique from Peterson et al. [b]. The percentage of electrode overlap indicates the ratio of regions with shared electrode coverage between two participants, with a value of 100% indicating that the two participants' regions with electrode coverage match exactly.

After converting the ECoG data to a common ROI space, we band-pass filtered the data using a 4th order Butterworth filter into 6 frequency bands: Low Frequency Oscillations (LFO) (0.5-4Hz), Alpha (8-12Hz), Beta (12-20Hz), Low Gamma (30-60Hz), Gamma (60-100Hz), and High Gamma (100-120Hz). We then trimmed the signal to 1 second before and after the movement event and normalized the signal with z-score standardization. We focused on the Beta band for our analyses, as it has shown the most relevance to movement activity in previous literature [Peterson et al., b; Miller et al.]. We found that the results did not change much across bands; see Supplement for results in other frequency bands (Supplemental Figures B.3 and B.4).

### 5.2.2 Experimentally-Controlled Data

To compare results from our naturalistic movement dataset to results from experimentally-controlled data, we used a publicly available dataset of concurrent ECoG and dataglove recordings from nine human participants performing five finger flexion tasks [Miller et al.; Miller, 2019]. In this paper, we refer to this as 'experimentally-controlled data'. Participants were visually cued to do one of five finger flexions in their hand contralateral to their electrodes: thumb, index, middle, ring, and pinky flexions, followed by a 2-second period of rest. Both ECoG and finger joint angles from the dataglove were originally sampled at 1000 Hz. We excluded four participants from the original dataset because their electrode locations were not available in a common coordinate space (Talairach coordinates).

For the remaining five participants, we converted their coordinates to MNI using the Yale Bioimage

Suite [Papademetris and University, 2014; Lacadie et al., 2008]. Each participant had an equal number of trials per finger flexion, as follows: E01 had 28 trials per class, E02 had 27, E03 had 23, E04 had 10, and E05 had 28 (2 males, 3 females; aged $25 \pm 8$ years). All patients participated in a purely voluntary manner, after providing informed written consent, under experimental protocols approved by the Institutional Review Board of the University of Washington (#12193). All patient data was anonymized according to IRB protocol, in accordance with HIPAA mandate. These data originally appeared in the manuscript "Human Motor Cortical Activity Is Selectively Phase- Entrained on Underlying Rhythms" published in PLoS Computational Biology in 2012 [Miller et al.]. This dataset is publicly available at:

https://searchworks.stanford.edu/view/zk881ps0522 [Miller, 2019].

To preprocess the experimentally-controlled data, we followed steps similar to those used for the naturalistic movement dataset pipeline. We notch filtered the ECoG data at 60Hz and its harmonics, then average referenced and downsampled the data to 250Hz; we also downsampled the dataglove sensor to 250Hz. Next, we projected the ECoG data to the same 311 ROIs as the naturalistic movement data and then band-pass filtered the data into the same six frequency bands.

### 5.2.3  Principal Angles Analysis

To obtain a linear estimate of the neural manifolds in the data, we used PCA to extract the directions of maximal variance in the ECoG data [Chung and Abbott; Gallego et al., c; Natraj et al., b; Jolliffe and Cadima, 2016]. After processing the ECoG data as described previously, we applied PCA to trial-concatenated, normalized data individually for each movement class, participant, and day of the naturalistic movement dataset (see Figure 5.1E). This process involved grouping the data by movement class, participant, and day, creating a tensor $X \in R^{(n \times c \times t)}$, where $n$ was the number of instances of the class, $c$ the number of ROIs, and $t$ the number of time samples in the 2-second span of ECoG data. We then trial-concatenated each tensor to produce a new matrix $X' \in R^{(c \times (t \times n))}$, and used PCA to calculate the orthonormal basis of principal components $D \in R^{c \times c}$. These principal components determined the ROIs with covarying neural activity, or neural modes. We visualized the contributions of each ROI to the principal components to understand the span of neural modes across ROI channels. The number of dimensions preserved, denoted $k$, was determined using the Variance Accounted For (VAF) with a threshold of 80%. We then defined the

reach-direction specific low-dimensional neural sub-spaces by projecting the data onto the first $k$ PCs of each datapoint ($D' \in R^{c \times k}$. We performed PCA using the scikit-learn library in Python.

After obtaining the low-dimensional neural sub-spaces, we used PAA to uncover whether the sub-spaces were similarly oriented in high-dimensional ROI space across the various data dimensions (Figure 5.2D) [Gallego et al., c; Natraj et al., b; Bjorck and Golub, 1973; Meyer, 2000]. PAA was chosen as our representational similarity metric because it allows for similarity assessment without the need for a transformation step to align the data, as required in methods like CCA [CCA, 2007; Williams et al., 2022]. Given the high variability in the naturalistic movements, the transformation step required in CCA would not be suitable. For pairs of lines, PAA simplifies to the angle between the two, but at higher dimensions, it becomes more complex due to the exponential growth in possible combinations of lower dimensions. To handle higher-dimensional spaces, we used the results from PCA to constrain our lower-dimensional combinations and determine the smallest achievable angles between the two reach-specific sub-spaces.

We performed singular value decomposition (SVD) of the dot product matrix between the resulting $D'$ sub-spaces from PCA for any two sub-spaces, A and B, by calculating:

$$P_a S P_b = SVD((D'_a)^T \cdot D'_b) \tag{5.1}$$

Completing the SVD calculation provided $S$, a diagonal matrix containing the cosine of the $k$ ordered principal angles (smallest to largest): $S = diag(cos(\theta_1), cos(\theta_2), ..., cos(\theta_k))$. For each $k$-dimensional neural sub-space, we computed the principal angles for pairwise comparisons along the relevant data dimension (see Figure 5.2E). For example, across movement classes we computed principal angles between the neural sub-spaces for leftward and rightward reaches, leftward and upward reaches, and so forth. This process resulted in six pairwise comparisons for each participant and each day in the naturalistic movement data (see Figure 5.1).

After calculating the principal angles for each sub-space comparison, we calculated a single metric of (dis)similarity between sub-spaces, which we call "neural dissimilarity." The neural dissimilarity metric provided an indication of how closely aligned two sub-spaces were, and thus the representational similarity of the sub-spaces. We calculated neural dissimilarity by summing across all $k$ principal angles and normalizing the value to the highest possible dissimilarity. The highest possible dissimilarity would have occurred if all

principal angles between the sub-spaces were orthogonal, i.e., at 90 degrees, yielding a value of 1. A neural dissimilarity of 0 would have indicated the lowest level of dissimilarity, i.e., perfect alignment between sub-spaces. We calculated this metric using the following equation:

$$ND = \frac{1}{k \times 90} \sum_{i=1}^{k} \theta_i \tag{5.2}$$

Note that equation 5.2 assumes that the principal angles $\theta_i$ are in degrees.

In the naturalistic movement data, we calculated the PAA pairwise sub-space comparisons along three data dimensions: movement (see Section Cross-Movement Sub-Space Comparisons), day (see Section Cross-Days Sub-Space Comparisons), and participant (see Section Cross-Participants Sub-Space Comparisons). We also bootstrapped some comparisons for the cross-days analysis, repeating our bootstrapping 100 times. In the experimentally-controlled data, we calculated the pairwise sub-space comparisons across the movement and participant dimensions (see Section Comparing Experimentally-Controlled Sub-Spaces). Since all data in this dataset was collected on a single experiment day, we did not consider a cross-days analysis.

### 5.2.4   Null Data Comparison

Our null hypothesis is that the neural sub-spaces are not more aligned in ROI space than expected by random chance. To test this, we used Tensor Maximum Entropy (TME) to generate a distribution of null data matching key statistical features of the real data [Gallego et al., c; Natraj et al., b; Elsayed and Cunningham, 2017]. We used TME to generate simulated data tensors from $R^{c \times t \times mvmt}$ that preserved the first and second-order statistical moments (mean and covariance) along the trial-concatenated time ($t$) and movement dimensions ($mvmt$) but not the channel dimension ($c$). After generating the simulated data, we computed the neural dissimilarity using the same approach as with the real data, iterating this procedure 1000 times to create a null distribution of principal angles. Principal angles below this null distribution at the 0.01 significance level indicated similarity between two sub-spaces.

86

### 5.2.5 Movement Similarity

To investigate the connection between movement and neural sub-spaces, we compared the kinematic similarity of movement tracking data from both datasets to the neural sub-space alignment. There are two possibilities for this relationship: (1) variations in movement kinematics change which neurons are recruited during the behavior, thus altering the neural channels the sub-space spans, or (2) the channel space of the neural sub-space remains constant, and the time-varying neural dynamics change based on the executed movements. Essentially as movement varies, either (1) the orientation of the sub-space changes, or (2) the time-varying neural dynamics change. Natraj et al. [b] found that changes in movement behavior primarily impacted neural dynamics rather than changing the orientation of the sub-space. Therefore, we hypothesized that the alignment of the neural sub-spaces would not correlate with behavioral similarity.

To test this hypothesis, we compared behavioral similarity to cross-movement neural dissimilarity (see Section Principal Angles Analysis). Behavioral similarity was calculated as the Pearson correlation of pose data between each trial, resulting in a pose correlation metric. We gathered pose coordinates from the naturalistic movement data and dataglove sensor information from the experimentally-controlled data, and trimmed them to the same time as the ECoG data. The behavioral data began as matrix $B \in R^{c_b \times t \times nt}$, where $c_b$ was the number of behavioral data channels (9 in the naturalistic movement data and 5 in the experimentally-controlled data), $t$ denoted the time samples, and $nt$ was the total number of movement behaviors across all classes. After concatenating the data along the channel and time axis to create a matrix $B' \in R^{nt \times (c_b \times t)}$, we calculated the Pearson correlation between each trial of data, resulting in a distance matrix $BD \in R^{nt \times nt}$ for each participant. We grouped the pose correlations by movement class and calculated the average correlations for each movement class comparison, i.e., we calculated the average pose correlation between all leftward reaching trials and all rightward reaching trials, and so on. We then plotted the neural dissimilarity against the average pose correlation for all participants and checked for trends between the two.

Given the limited number of movement class comparisons in the dataset, which could have constrained our ability to find trends between neural and behavioral data, we bootstrapped the pose correlations and neural dissimilarity metrics. For each participant, we sub-sampled 80% of the ECoG and behavioral data (extracting the same trials for both data types) 100 times and recalculated the neural dissimilarity and pose

correlation. All bootstrapped comparisons were added to our behavioral results to gain a clearer understanding of the relationship between neural dissimilarity and pose correlation.
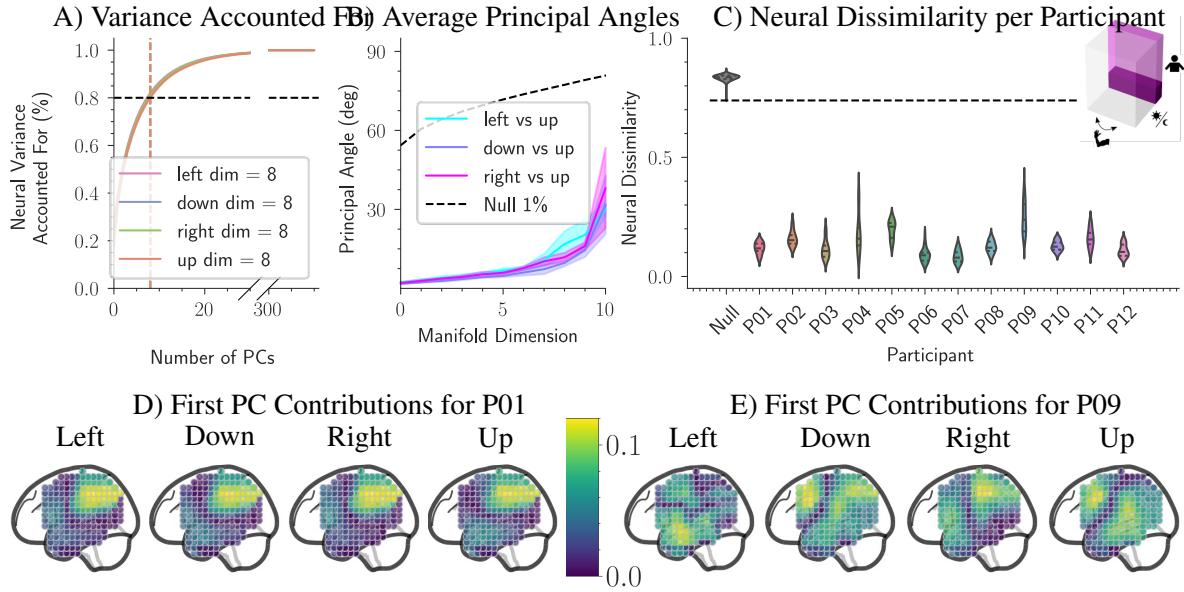
## 5.3 Results

We investigated the representational similarity of neural activity across various naturalistic movements, days, and participants by estimating neural sub-spaces using PCA, determining angles between sub-spaces with PAA, and calculating our metric of neural dissimilarity from these angles. Our findings reveal that neural activity during naturalistic movement occupies a low-dimensional sub-space, consistent with previous literature on experimentally-controlled movements. The neural activity across different naturalistic movements within the same individual is also more similar than expected by chance. Additionally, neural activity remains more similar than chance within participants over the five-day hospital stay, and between different individuals performing similar naturalistic reaching movements. Comparing these results to an experimentally-controlled movement dataset, we find that neural activity from the experimentally-controlled movements occupies a lower-dimensional sub-space than the naturalistic data and exhibits higher representational similarity.

### 5.3.1 Cross-Movement Sub-Space Comparisons

We first analyze the alignment of the estimated neural sub-spaces for different types of movements in the same participant. For this comparison, we quantify the principal angles between the neural sub-spaces of the four movement classes in the same participant on the same day (see embedded sub-figure in Figure 5.3C, where the arrow indicates the comparison dimension). Thus, the PAA results in six pairwise comparisons of neural sub-spaces for each participant on each day. If a participant does not have movement data for a movement class on a given day, we omit that comparison.

Consistent with previous research on neural manifolds, we find that the manifold estimates for human naturalistic movements occupy a low-dimensional sub-space [Gallego et al., b,c; Natraj et al., b]. Figure 5.3A) shows the average VAF (over participants and days) for each movement class over all 311 ROIs. For each movement, 80% VAF was reached by 10 dimensions, indicating that the neural sub-spaces of naturalistic movements do occupy a low-dimensional sub-space. As a result, we set our sub-space dimension

**Figure 5.3: Low-dimensional neural sub-spaces of different movement types for the same participant show more alignment than chance.** We performed movement comparisons for each day of each participant's hospital stay. **A)** The VAF for each movement type is shown over the 311 ROI dimensions. **B)** Some example pairwise comparisons of the principal angles between movements for one participant (P01); in this case, all comparisons are for upward reaches. Solid lines show the mean principal angles for that movement comparison, and colored bands show the standard deviation for movement comparisons over the 5 days of data. The dashed line shows the 1% significance level from the null data (see Null Data Comparison). **C)** Using the neural dissimilarity score (see Principal Angles Analysis and Figure 5.2), we observe that each participant's cross-movement sub-space comparisons are smaller than the 1% significance from the null data (see dotted line). **D)** We plot the principal component contributions of all 311 ROIs for the first principal component in P01 for each movement type. Dark purple regions contribute the most to the first principal component, and yellow regions contribute the least. **E)** ROI contributions for the first PC for participant P09, who has the highest average cross-movement neural dissimilarity.

cut-off to k=10 for further analyses on neural sub-spaces within the Beta frequency band (see Supplemental Figure B.3 for the VAF across the other frequency bands).

We find that the principal angles between movement sub-spaces are consistently smaller than expected by chance (see Section 5.2.4 for how we determined chance level). Figure 5.3B) illustrates some pairwise movement comparisons and their principal angles over the 10 sub-space dimensions. These angles start small (average principal angle = 2.691 degrees, std = 1.676) and gradually increase with each compared dimension (average at $k^{th}$ dimension = 48.844 degrees, std = 20.394). All principal angles between movement comparisons are lower than the chance level, defined by the lowest 1% of angles between data generated with TME (indicated by the dashed line). The average neural dissimilarity across all participants is 0.142
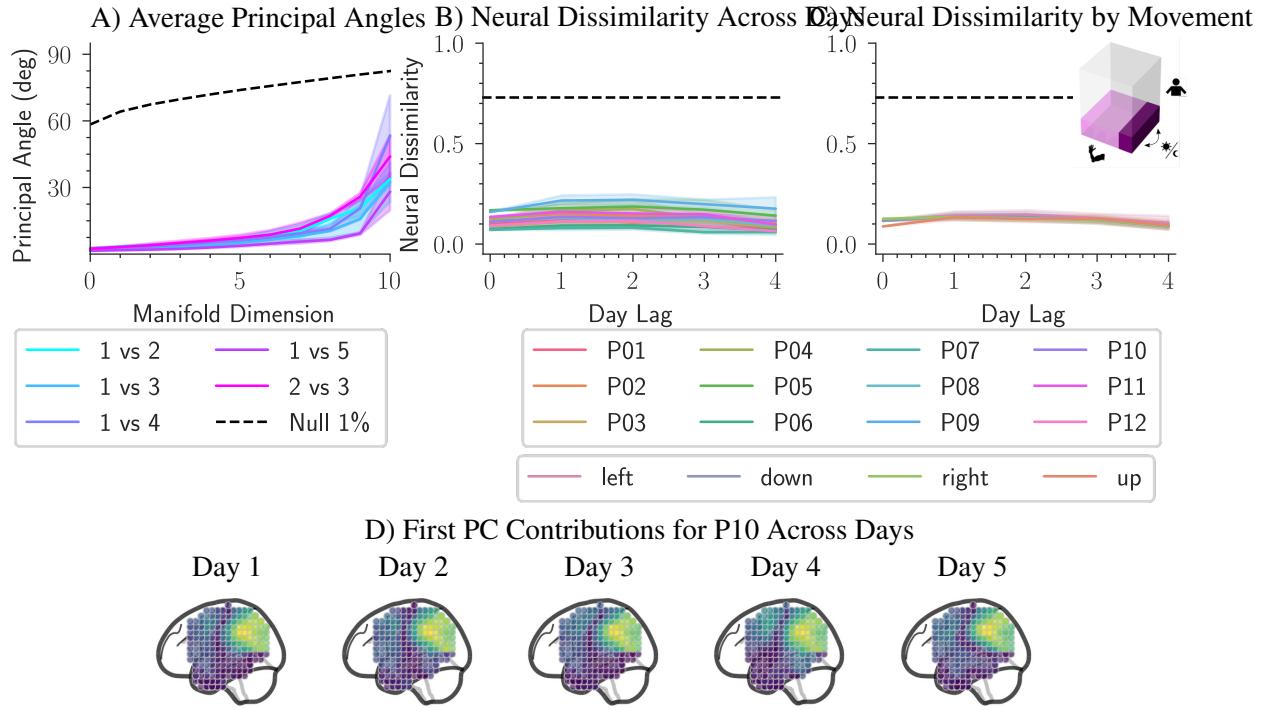
(std = 0.063), compared to a neural dissimilarity of 0.739 for the 1% null significance level (indicated by the dotted line on Figure 5.3C), indicating alignment in the neural activity across movements.

We also examined the ROI contributions to the first principal component of the sub-spaces to understand what regions they span and possible drivers of cross-participant differences. In participant P01, the ROI contributions for each movement type show similar mappings, potentially explaining P01's low neural dissimilarity (Figure 5.3D). For P01, we also find the regions around motor and sensorimotor areas, as well as the parietal lobe, contribute most to the first principal component. In contrast, participant P09 shows the highest neural dissimilarity for cross-movement sub-space comparisons. The ROI contribution maps for P09 (Figure 5.3E) reveal divergent contributions across movements, such as larger contributions in the temporal lobe for leftward reaches, explaining the higher neural dissimilarity.

Previous results only include neural sub-spaces in the Beta frequency band, though we find that neural sub-space alignment across all six frequency bands remains lower than chance (see Supplemental Figure B.3). For all sub-space comparisons (cross-movement, cross-days and cross-participants), neural dissimilarity is lower than chance across all frequency bands. Since there was no large difference in alignment across frequency bands, and because the 12-20Hz Beta band has been shown to correspond with movement [Peterson et al., b; Miller et al.], we only report alignment in the Beta band moving forward.

In comparing cross-movement neural dissimilarity to movement behavior similarity, we find no clear relationship between neural dissimilarity and pose correlation, indicating that the sub-spaces are not re-oriented in ROI space depending on the behavior. We find that the pose correlation across all movement trials is relatively high (average pose Pearson correlation across all participants: 0.698, std = 0.129, see Supplemental Figure B.5 A and B). When bootstrapping the data, we also find that the data is highly clustered by participant, indicating consistency in the cross-movement neural sub-space alignment and movement correlations.

Overall, these results show that neural activity during naturalistic movements exists along low-dimensional neural sub-spaces, which are closely aligned in ROI space. This may suggest that the human brain utilizes these similarly aligned low-dimensional sub-spaces not only to control experimentally-controlled movements, but also to control naturalistic movements. While this reveals that neural activity during the same day is well aligned, it remains unclear whether these sub-spaces are preserved over multiple days.

A) Average Principal Angles  B) Neural Dissimilarity Across Day  C) Neural Dissimilarity by Movement

**Figure 5.4: Low-dimensional neural sub-spaces for the same types of movements across the 5 days of participants' hospital stay did not drift significantly and remained similarly oriented in ROI space.** **A)** Example of principal angles on a subset of cross-days comparisons for P01 are lower than chance. Solid lines show average principal angles for cross-day comparisons, shaded regions shows standard deviation across the movement types, and a dashed line indicates 1% null significance. **B)** The neural dissimilarity of the same types of movements across days remains lower than expected by chance for all participants. Solid lines show average neural dissimilarity for cross-day comparisons, shaded regions shows standard deviation across the movement types, and a dashed line indicates 1% null significance. **C)** The cross-days neural dissimilarity by movement type shows that all movement types remain more aligned than chance, with no major differences between movement types. **D)** We show the ROI contributions of the first PC for participant P10 for leftward reaches across the five days (dark purple indicates higher contribution, and yellow indicates less contribution).

## 5.3.2 Cross-Days Sub-Space Comparisons

We now analyze the neural sub-spaces of the same types of movements across the participants' five-day hospital stay. For this comparison, we quantified principal angles between the neural sub-spaces of the same movement class in the same participant over the course of the five days (see embedded sub-figure in Figure 5.4C, where the arrow indicates the comparison dimension). Thus, the PAA results in 10 pairwise comparisons of neural sub-spaces for each participant and each movement class. If a participant does not have data for a movement class on a given day, we omit that comparison. We also bootstrap comparisons of
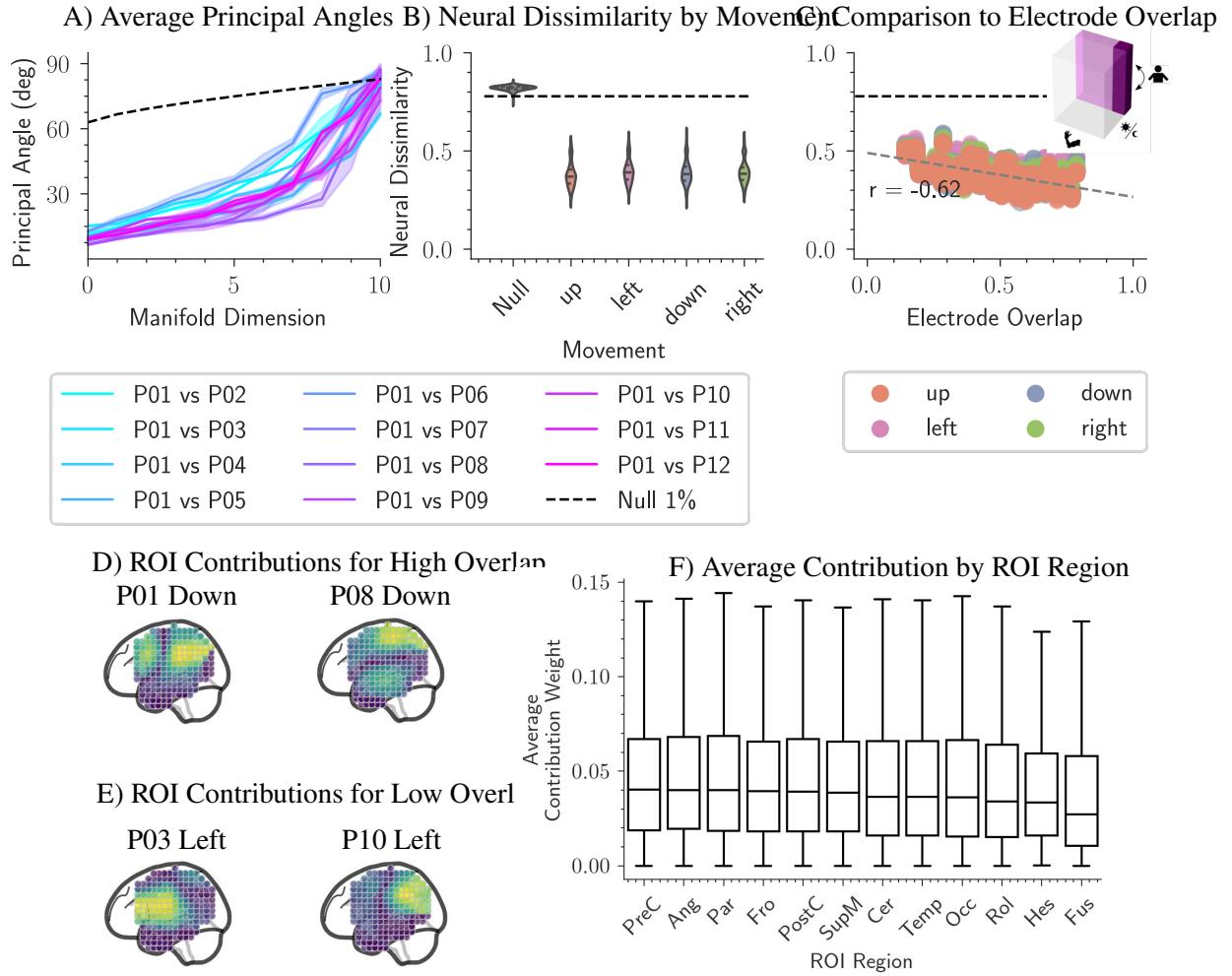
the same day to calculate a baseline of the sub-space alignment on the same day.

For the cross-days sub-space comparison, we find that the dimensionality of the sub-spaces remains low and more aligned than expected by chance. The average VAF stays around 10 dimensions to achieve 80% VAF across each day (Supplemental Figure B.1). Figure 5.4A shows an example of the principal angles between some cross-days comparisons, which star small (average principal angle = 2.424 degrees, std = 1.336) and increase up to an average of 45.313 degrees (std = 19.905) by the last dimension, similar to the cross-movement sub-spaces (Section Cross-Movement Sub-Space Comparisons). The principal angles across all dimensions remain well below the null significance level.

We then calculate the neural dissimilarity over all cross-days comparisons, and we find that the neural sub-spaces remain stable over the course of 5 days. To identify how the sub-spaces change over time, we analyze all cross-days comparisons by how many days have passed. For example, we represent neural sub-space comparisons of leftward movements on day 1 vs day 3 as a 2-day lag and day 1 vs day 5 as a 4-day lag. We calculate day lags of 0 days by bootstrapping the sub-space comparisons within that day. On average across the days, neural dissimilarity stays around 0.111 for all participants (std = 0.056), compared to a null significance of 0.730 neural dissimilarity. Interestingly, the average neural dissimilarity between the first and the last day (day lag = 4) is 0.095 (std = 0.04), dropping from the average neural dissimilarity by 0.016 (Figure 5.4B). Neural dissimilarity across the day lag, when categorized by movement type, also demonstrates consistent similarity levels across different types of movements (see Figure 5.4C).

From the ROI maps, we can further observe that the sub-spaces remain stable over the 5 days. In Figure 5.4D, we plot the ROI contributions of the first PC of the leftward reaches for P10 on all days. All five days show high concentrations in postcentral and parietal regions. However, the regions of low contributions change over the course of the hospital stay (see purple regions in plots).

Overall, these results show that the neural activity of related naturalistic movements remains similarly oriented in ROI space over the course of several days. These results may indicate how the brain is able to consistently control naturalistic behavior across multiple days despite changes in the individual's neural activity, environment and behavior [Peterson et al., 2021]. It thus appears that neural sub-space activity remains consistent within participants, but are similar sub-spaces used for naturalistic movement control across different individuals' brains?

**Figure 5.5: Neural sub-spaces for the same movement types across different participants show greater alignment than chance, but less than within participant sub-spaces.** **A)** Example principal angles between participants on leftward reaches. Solid lines show average principal angles for cross-participant comparisons, shaded regions shows standard deviation across days, and a dashed line indicates 1% null significance. **B)** The average neural dissimilarity score for each movement type across participants is lower than chance. **C)** Comparison of participants' electrode overlap (see Section Naturalistic Movement Data) to cross-participant neural dissimilarity and find some correlation. **D)** First PC contributions for the same movement type (down) for two participants, P01 and P08, with high electrode overlap. **E)** Comparison between first PC contributions for two participants, P03 and P10, who have low electrode overlap. **F)** Average PC contribution weight by brain region for each ROI across all participants, days and movement types, sorted by median value.

### 5.3.3 Cross-Participants Sub-Space Comparisons

We now explore the neural sub-spaces of the same types of movements, but across all participants. For

this comparison, we quantify the principal angles between the neural sub-spaces of the same movement

93

class between different participants (see embedded sub-figure in Figure 5.5C, where the arrow indicates the comparison dimension). The PAA results in 66 pairwise comparisons of neural sub-spaces for each day and each movement class. We also make PAA comparisons for all combinations of days between participants, i.e., we compare the same movement types for P01 on day 1 to all days for P02, and so on, creating a total of 1650 comparisons. If a participant does not have data for a movement class on a given day, we omit that comparison.

In comparing across participants, we find that the principal angles, and thus neural dissimilarity, increases relative to cross-movement and cross-day sub-spaces but remains more aligned than expected by chance. Reviewing the example principal angles in Figure 5.5A, we observe that the principal angles still start small (1st dimension avg = 10.034 degrees, std = 2.203), but approach the null significance line in the higher dimensions (last dimension avg = 82.200 degrees, std = 7.646). The neural dissimilarity of cross-participant comparisons is overall higher than it was in the cross-movement and cross-days comparisons (average neural dissimilarity = 0.388, std = 0.061); see Figure 5.5B. However, the cross-participant sub-space comparisons still show more alignment than expected by chance (neural dissimilarity of null significance level = 0.778).

Interestingly, we find a strong relationship between the electrode overlap between participants and their neural dissimilarity (Figure 5.5C). We calculate electrode overlap as a percentage of how many electrodes share region coverage between participants (average electrode coverage between participants = 46.0%, std = 16.4%). We calculate Pearson correlation coefficient between participants' neural dissimilarity and their electrode overlap for all cross-participant sub-space comparisons, and find a strong negative correlation (r=-0.62, p-value < 0.001). Thus, participant pairs with more electrode overlap tend to have lower neural dissimilarity between them. We also evaluate the relationship between electrode overlap and neural dissimilarity by movement type and find no difference in the trend.

We can visualize ROI contribution maps between participants with high overlap (Figure 5.5D) and low overlap (Figure 5.5 E) to understand how the neural sub-spaces differ between participants. P01 and P08 have high electrode overlap (overlap percentage = 76.9%, neural dissimilarity between downward reaches = 0.254), and both of their maps show high contributions in the parietal lobe and near sensorimotor regions. P03 and P10 have much lower electrode overlap (overlap percentage = 25.7%, neural dissimilarity between
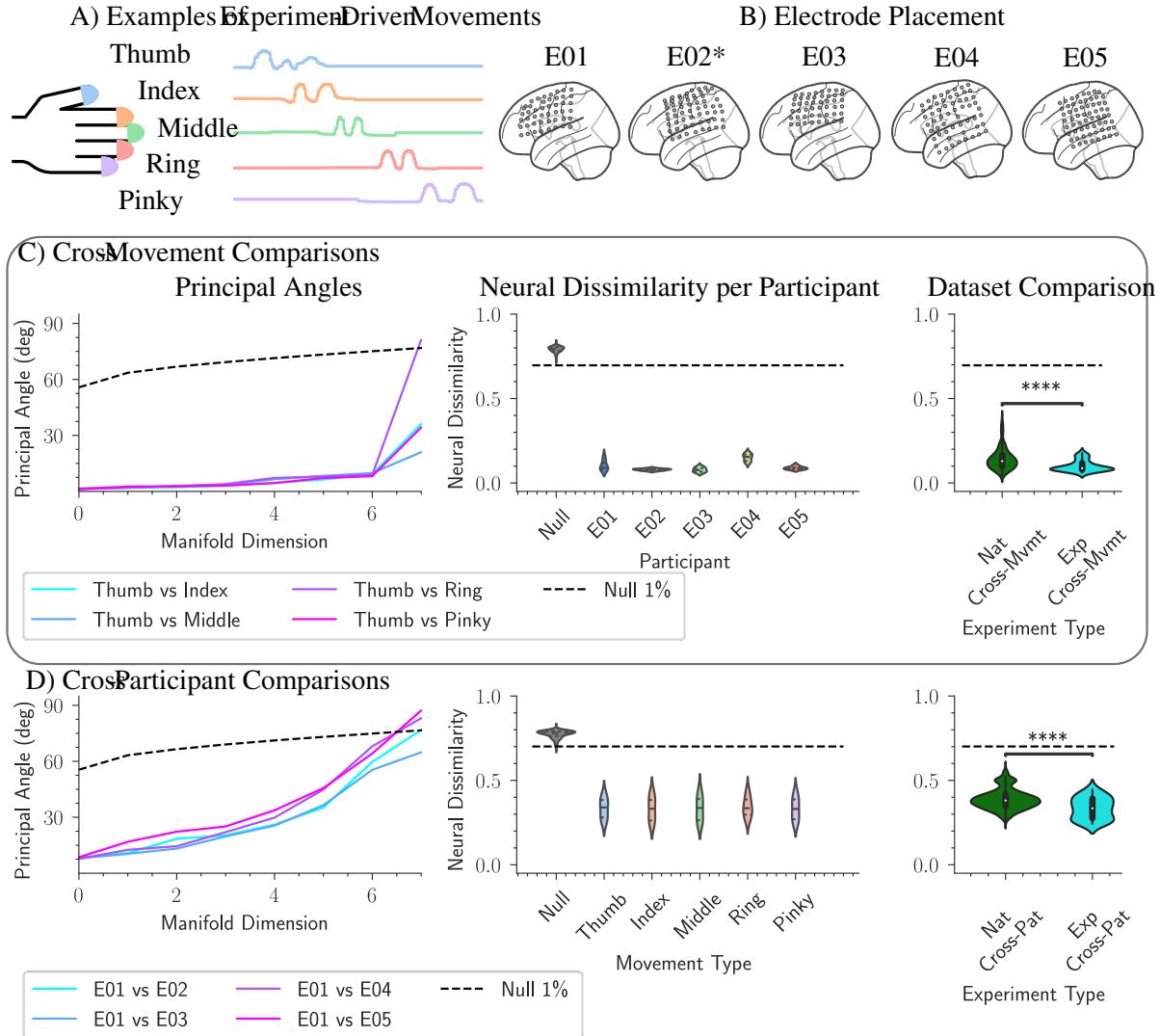
leftward reaches = 0.419), and their maps show markedly different concentrations of first PC contributions; P03 shows high contributions around frontal and precentral regions, while P10 has high contribution around postcentral and parietal regions.

We also examine the average ROI contributions across all PCs and find that regions related to motor activity contribute nearly equally to the extracted principal components. We calculated the average ROI contributions across all participants and movements as well as across all 10 dimensions that define the neural sub-spaces. From the average contribution weights, we find that precentral ROIs contribute the most to the movement sub-spaces (median contribution = 0.040, std = 0.033). The next highest contributions come from the angular gyrus (median contribution = 0.040, std = 0.035) and the parietal lobe (median contribution = 0.040, std = 0.035). The other main motor region, postcentral, also shows relatively high contributions (median contribution = 0.039, std = 0.034).

In summary, we find that all participants share similarly oriented neural sub-spaces to control naturalistic movements. The results also show that these neural sub-spaces generally occupy the precentral region, with some other movement-related regions (parietal and postcentral) also showing high contributions. These areas seem to span core regions in the grasp network [Brochier and Umiltà, 2007; Davare et al., 2011; Fagg and Arbib, 1998; Jeannerod et al., 1995; Schaffelhofer and Scherberger, 2016], consistent with their relation to movement control. Overall, it appears that most of the human brains we evaluated employ a similar neural sub-spaces to control naturalistic movement, which may help us to design and align cross-participant naturalistic neural decoders moving forward.

### 5.3.4 Comparing Experimentally-Controlled Sub-Spaces

Next we ran the same set of neural sub-space analyses with PAA on ECoG data recorded during experimentally-controlled movements [Miller et al.]. We compare our naturalistic movement dataset to the experimentally-controlled movement dataset to understand how the neural sub-spaces change between movement contexts, and explore whether low-dimensional neural sub-spaces are a consequence of experimental constraints. In this dataset, participants were directed to perform five finger flexion movements while wearing a dataglove (Figure 5.6A). Like our naturalistic movement dataset, participant electrode coverage spans a wide variety of brain regions (Figure 5.6B). We run the same process of converting electrodes to ROIs and then run the

**Figure 5.6: Comparing naturalistic movement data to the experimentally-controlled dataset, we find that neural sub-spaces in the latter are more similarly oriented than in the former, both across movements and across participants. A)** Experimentally-controlled data comes from a set of ECoG participants who were instructed to do five different finger flexions with their contralateral hand while wearing a cyberglove (see Miller [2019]). The five movements in this data are: index flexion, thumb, middle, ring and pinky finger flexion. We show examples of data for each movement type from the cyberglove with colored lines. **B)** Electrode coverage of the five participants from the experimentally-controlled dataset, which also show variation in where electrodes were placed. Participants with an * had their electrodes on the right hemisphere. We flipped electrodes on the opposite hemisphere for clarity of visualization. **C)** Cross-movement sub-space comparisons for experimentally-controlled data are also more aligned than expected by chance and show lower neural dissimilarity than the naturalistic movement data. The first panel shows example principal angles for cross-movement comparisons on participant E01, specifically movement comparisons to thumb flexions. The dashed line indicates null significance. The second panel shows neural dissimilarity for cross-movement comparisons, which is lower than chance across the experiments' participants (dashed line indicating null 1% significance). The third panel shows a comparison of cross-movement neural dissimilarity between the naturalistic movement and experimentally-controlled data. **D)** Cross-participant sub-space comparisons for experimentally-controlled data are also more aligned than expected by chance and are more aligned than the naturalistic case. The first panel shows example principal angles for cross-participant comparisons on thumb flexion, specifically showing all comparisons to participant E01, with null significance indicated as a dashed line. The second panel shows the neural dissimilarity for each movement type, which is below the null significance. The third panel shows the comparison in cross-participant neural dissimilarity between the naturalistic movement and experimentally-controlled data.

same PCA to PAA processing pipeline as the naturalistic movement data (Figure 5.2C-F). Since data for this dataset was collected during one session for each participant, we cannot run cross-day sub-space comparisons. We do make cross-movement and cross-participant comparisons, as we did for the naturalistic movement data. We also compare the behavioral data from the dataglove to the neural dissimilarity, as we did for the naturalistic movement data.

We find that the neural sub-spaces in the experimentally-controlled ECoG data also occupy a low-dimensional sub-space of the ROI space. We also find that the number of PCs needed to reach 80% VAF in the Beta frequency neural sub-spaces is at least 7 dimensions across all participants and all movements (see Supplemental Figure B.2). This indicates that the experimentally-controlled movement sub-spaces require slightly fewer dimensions (to reach 80% VAF) than the sub-spaces for the naturalistic movement data. For the following analyses, we reduce the experimentally-controlled sub-space dimensions down to the top 7 principal angles (k=7) to cover all 80% VAF. Across frequency bands, the experimentally-controlled ECoG data consistently exhibits low-dimensional sub-spaces, with the highest dimension being 10 in the Gamma band, and also neural dissimilarity remains lower than expected by chance, indicating similar orientations of the sub-spaces across the analyzed frequency bands (Supplemental Figure B.4A).

We analyze the principal angles between the neural data for the same finger flexions (Figure 5.6C) and find that the neural dissimilarity is lower than chance, and also lower than the naturalistic movement data. The first panel in Figure 5.6C shows an example of the principal angles between different movements in E01. As in the naturalistic movement data, the principal angles start small in the initial dimensions and gradually increase (1st dimension average principal angle = 1.934 degrees, std = 0.552; last dimension average principal angle = 31.043 degrees, std = 15.827). Reviewing the overall neural dissimilarity of cross-movement sub-spaces across participants, we see that all sub-spaces are more aligned than chance, with an average neural dissimilarity of 0.100 (std = 0.032; null significance level = 0.697). We also compared the overall cross-movement neural dissimilarity between the experimentally-controlled data and the naturalistic movement data. We find that the average cross-movement neural dissimilarity for the experimentally-controlled data is significantly lower than cross-movement neural dissimilarity for the naturalistic movement participants (p-value <= 1.00e-04).

Consistent with our analysis of naturalistic movement data, we also analyze neural sub-spaces of the

same movements across participants, and find that neural dissimilarity is lower than chance, but higher than within-participant comparisons. The first panel in Figure 5.6D shows example principal angles between different participants' sub-spaces for thumb flexions. We find that while the principal angles are still lower than chance (as indicated by the dashed line), they are larger than the principal angles shown in the cross-movement comparisons (1st dimension average principal angle = 7.804 degrees, std = 1.81; last dimension average principal angle = 66.073 degrees, std = 12.436). In the examples shown, some principal angles in the higher dimensions even start to cross the null significance threshold by the last dimension. When reviewing overall neural dissimilarity, we find that all cross-participant comparisons for all movement types are more aligned than expected by chance (average neural dissimilarity = 0.334, std = 0.064; null significance level = 0.701). We also find that neural dissimilarity by movement type shows minimal differences. We then compare the overall cross-participant neural dissimilarity between the experimentally-controlled data and naturalistic movement data, finding that experiment-based cross-participant sub-spaces are more aligned than naturalistic movement cross-participant sub-spaces (p-value <= 1.00e-04).

Similar to the way we analyzed ROI regions in the naturalistic movement data, we also examined the average PC contribution weights by ROI region in the experimentally-controlled data, and find nearly equal contributions from the various regions, including motor-related regions (see Supplemental Figure B.6). We take the average across all participants and movements as well as across the 7 sub-space dimensions. We find that the experimentally-controlled data had the highest average contributions from the occipital lobe (median contribution = 0.047, std = 0.032), followed by high contributions from rolandic (median contribution = 0.047, std = 0.031) and angular gyrus (median contribution = 0.047, std = 0.033) (see Supplemental Figure B.6). Interestingly, traditional motor regions, specifically precentral and postcentral sulci, do not show as high a concentration in this dataset, but the contribution of the angular gyrus is consistent with results from the naturalistic movement data discussed above.

Finally, we analyze the relationship between movement behavior and the cross-movement neural dissimilarity in the experimentally-controlled data, and similarly to the naturalistic movement data, we do not find a clear relationship (see Supplemental Figure B.5C and D). We find that the movement correlation across all movement trials is low (average movement correlation across all participants = -0.018, std = 0.078). When bootstrapping the data, we find that the data is highly clustered by participant, much as it is in the naturalistic

movement data.

Thus, we find that neural activity driving both the naturalistic and experimentally-controlled movements evolves on low-dimensional neural sub-space. Further, we find that the experimentally-controlled data exists on a slightly lower dimensional sub-space than the naturalistic movement data, and that the neural sub-spaces of experimentally-controlled movements are significantly more aligned than they are for naturalistic movements. These results indicate that while naturalistic neural data also inhabits low-dimensional neural sub-spaces, these sub-spaces are higher-dimensional and show more variability than experimentally-controlled counterparts.

## 5.4   Discussion

This study used ECoG data during naturalistic movements to investigate how the human brain flexibly supports a range of reaching movements in everyday tasks. Building on prior research in non-human primates and humans [Gallego et al., c; Natraj et al., b], we applied similar analysis techniques to reveal low-dimensional neural sub-spaces in human ECoG data during naturalistic movements. We aimed to determine whether these low-dimensional sub-spaces underlie neural activity during naturalistic movements or if they are specific to experimentally-controlled movements. Our findings indicate the presence of closely aligned low-dimensional sub-spaces during the control and generation of naturalistic movements, even over several days and across different individuals. Together, these results suggest that neural activity occupies stable low-dimensional sub-spaces for both experimentally-controlled and naturalistic movements.

The significant alignment of these neural sub-spaces is intriguing, considering the high variability between movements and participants [Peterson et al., 2021]. Despite naturalistic movements varying in many factors, such as speed, magnitude, and bi-manual involvement, neural activity for all movements tends to lie within similar sub-spaces, suggesting the importance of maintaining neural activity within these sub-spaces to generate various movements. The low-dimensionality of the naturalistic neural data also does not seem to be a consequence of the dimensionality of the movement pose data, which showed a dimensionality of four for 80% VAF, versus ten for neural data. We also found no direct relationship between movement pose correlation and the alignment of neural sub-spaces (see Supplemental Figure B.5), indicating that the orientation of neural sub-spaces is not directly related to the type of movement being generated. This is in-line

with research from Natraj et al. [b], who also found that neural activity of different movements occupied similar sub-spaces, but that the neural dynamics occupied sub-regions of the manifold based on movement behavior. Future work could explore the relationships between neural activity dynamics along the manifold and naturalistic movement behaviors, potentially revealing similar compartmentalization results. We may also find that the degree of change in the neural dynamics is proportional to the changes in movement behavior, potentially providing an indication of the neural code for naturalistic movement generation.

In further understanding the neural activity in our dataset, we also analyzed the brain regions that contributed most to the sub-spaces. Unlike many studies that focus on microelectrode recordings [Gallego et al., 2022], we evaluated macro-scale ECoG recordings, which provides insights into larger-scale networks involved in movement generation. While Natraj et al. [b] also used ECoG, their micro-ECoG grids recorded data only in motor regions (primary motor, premotor, sensorimotor, and parietal cortex), whereas our work includes recordings from several regions, including the frontal and temporal lobes. We found that neural sub-spaces of naturalistic movements span most of the recorded brain areas, with the primary motor cortex showing the highest contributions. Additionally, we observed significant contributions from the angular gyrus, known for its role in language, attention, and 3D spatial orientation, which is relevant for tasks involving object manipulation in 3D space [Seghier, 2013; Chen et al., 2012]. Other major contributors included the parietal lobe and postcentral gyrus, known to be part of the grasp network [Brochier and Umiltà, 2007; Davare et al., 2011; Fagg and Arbib, 1998; Jeannerod et al., 1995; Schaffelhofer and Scherberger, 2016]. Our findings are consistent with Natraj et al. [b], who also found that neural sub-spaces occupied regions of the grasp network, but our more distributed recordings allowed us to identify additional brain regions involved in naturalistic movement generation, such as the angular gyrus and frontal cortex. Overall, we observed that neural sub-spaces are distributed across most of the recorded brain regions, with a near-equal distribution among areas of the grasp network.

Our results indicate that the neural sub-spaces for naturalistic movement data exhibit minimal drift over the course of five days, with no significant increase in neural dissimilarity over time; in fact, we observed a slight decrease in average neural dissimilarity on the longest day lag. This is in contrast to other work in neural manifolds, specifically work that uses manifold-alignment-based-decoding to realign decoders after several days or weeks after the initial recordings [Gallego et al., a; Degenhart et al., 2020; Brianna

M Karpowicz et al., 2022; Jude et al., 2022; Natraj et al., a]. In many of these cases, researchers find that without realigning the decoder performance degrades over time, implying that the neural activity changes either in manifold orientation or neural dynamics. However, it is very possible these neural changes are a result of participants' learning the experimental movement task Orsborn et al.. In the case of naturalistic movements, the participants do not need to learn new behaviors, and instead can rely on existing neural circuitry to generate those movements. These differences in learning contexts may explain why we do not see changes in neural sub-space alignment over the course of time.

Given the stable low-dimensional sub-spaces across days for naturalistic movements and their consistent orientation across participants, using a manifold-based decoding approach could be beneficial for long-term and cross-participant decoding of naturalistic movements. Previous studies have used manifold alignment for long-term decoding in non-human primates [Gallego et al., a; Degenhart et al., 2020; Brianna M Karpowicz et al., 2022; Jude et al., 2022], and recently in humans [Natraj et al., a]. Previous work has also demonstrated cross-participant decoding in non-human primates [Safaie et al.] and even between monkeys and humans [Rizzoglio et al.]. While these techniques have shown impressive results, they often rely on assumptions that may not hold for naturalistic movements, such as the ability to time-align different movement behaviors for CCA Gallego et al. [a]; Safaie et al.. Techniques that do not rely on such assumptions, such as adversarial techniques [Farshchian et al.; Ma et al.] or unsupervised domain adaptation [Jude et al., 2022; Sussillo et al., b], may be more suitable for decoding naturalistic movement data. Additionally, the high degree of similarity in sub-space orientation of naturalistic neural activity suggests that incorporating temporal information into decoding algorithms will be crucial. However, despite the potential of manifold alignment techniques, achieving success with manifold alignment between participants requires some level of initial domain alignment, as shown in our electrode overlap results. Thus, while manifold alignment decoding may not be a universal solution for extreme domain misalignment, they can still be valuable tools for decoding complex naturalistic neural data in humans.

While our results offer initial evidence of neural manifolds in humans during naturalistic movement, there are several opportunities for improvement. Accurately tracking naturalistic behavior poses inherent challenges, especially with single-camera pose tracking, which may lead to occlusions and affect the quality of pose estimations [Singh et al., b]. Additionally, our decision to categorize arm movements into four

classes instead of analyzing the continuous spectrum of reaches may have limited our understanding of the naturalistic neural data. Future research could explore the overarching neural manifold controlling all reaching behaviors and the continuous nature of these reaches within the sub-space. Additional future research should also investigate the neural dynamics on the sub-space, which could inform neural decoding design. Our use of ECoG arrays, while beneficial for identifying a broad range of brain regions involved in movement generation, lacks the fine-grained recording capabilities of microelectrodes or micro-ECoG, future work may look to bridge this gap between micro and macro recordings. Although our linear methods successfully demonstrated the similarity of neural sub-spaces within and across participants, non-linear methods may provide richer information about the neural activity underlying movement control [Fortunato et al., 2023; Sussillo et al., b; Farshchian et al.; Roweis and Saul, 2000]. Our comparison of naturalistic movement data to experimentally-controlled data could also be improved by comparing to ECoG data recorded during directed 3D reaches in virtual reality [Paschall et al., 2023; Śliwowski et al., 2022]. Lastly, while our analysis focused on naturalistic movement behaviors, the data was not fully naturalistic, as all participants were epilepsy patients undergoing post-surgical recovery, potentially limiting the generalizability of our findings to healthy individuals and different contexts.

## 5.5   Summary

In this chapter, we find evidence that neural activity during naturalistic movement exists in low-dimensional neural sub-spaces in the human brain. We find that the neural sub-spaces of different types of naturalistic movements are similarly oriented for the same participant. We also find that neural sub-spaces of similar movements for a participant remain stable over the course of several days, and that neural sub-spaces are even more aligned than expected by chance between different participants. We further find that the sub-spaces span grasp-related cortical regions, but they also include activity from other regions, including the angular gyrus and frontal areas. The results in this chapter expand our understanding of the geometry of neural activity during natural behavior, and support the possibility of using manifold alignment for neural decoding in long-term and cross-participant decoding of naturalistic movements.

# Chapter 6

# Decoding Naturalistic Human Neural Data - Trials and Tribulations

## 6.1   Introduction
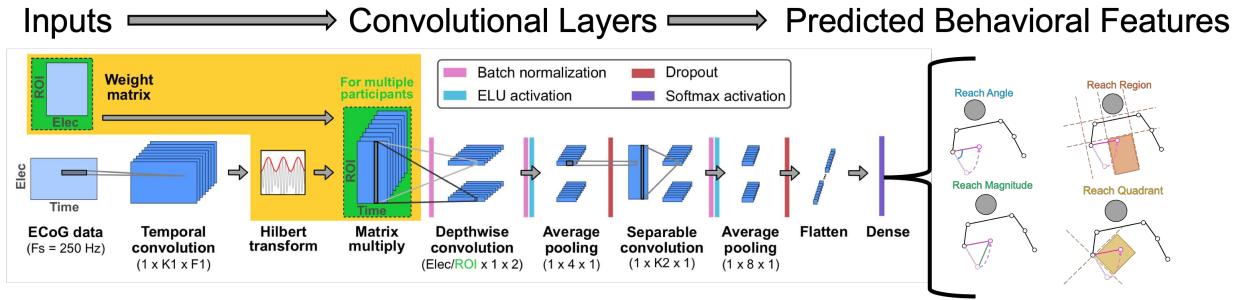
In this chapter, I will discuss several projects that, for various reasons, did not reach the publication stage. Often, the results were either inconclusive or incomplete, resulting in a chapter that could affectionately be called the "failures" chapter. Despite this, there are many useful insights from the work presented in this chapter. All projects in this chapter aimed to advance the neural decoding research outlined in Chapter 3 [Peterson et al., b]. In Section 6.3, I discuss my attempts to enhance ECoG-based neural decoding to predict more intricate movement features, such as reach angle and magnitude. The goal of this project was to develop models that could move beyond predicting movement and rest states, and predict movement features that would be more useful for various applications, such as prosthetics. Following that, Section 6.4 presents a collaborative project with Gabrielle Strandquist and Zeynep Toprakbasti to explore various self-supervised learning methods for decoding our naturalistic ECoG data. The objective in this project was to create neural decoders that could utilize the extensive, but unlabeled, ECoG data at our disposal. Lastly, in Section 6.5, I explore the use of neural manifold approximations in naturalistic ECoG for neural decoding. Building on the findings from Chapter 5, this work aimed to develop decoders that could rapidly generalize across different participants using manifold alignment.

## 6.2   The Data

In all of these projects, we used ECoG data from the same source - AJILE [Wang et al., f, 2018; Singh et al., b]. This data came from 12 epilepsy patients at Harborview Medical, who underwent invasive brain surgery to help locate their seizures. All participants stayed in the hospital for about a week, and we extracted data from two days after the electrode implanting surgery, until they left for explantation, providing around 5 days of data per participant. During their stay at the hospital, video, audio and neural data were collected to monitor participant behaviors, thus allowing us to build up a neural dataset that tracked *naturalistic* behaviors. Researchers then used the video data to generate features of movement behaviors, such as when movements started, or the angle of their wrist during movement. More details about this dataset and how the movement features were extracted can be found in Singh et al. [b]. The audio data was used to determine occurrences of speaking, which is used for downstream decoding in Section 6.4. All behaviors were spontaneously generated by the participants, providing interesting insights into the brain during spontaneous, natural behavior. This dataset was also used in Chapters 3 and 5. See each project section below for details on the features used for decoding.

## 6.3   Predicting Complex Movement Features

As discussed in the introduction, the goal of this project was to expand upon the neural decoding results shown in Chapter 3 by building neural decoders that can predict more complex movement features. By developing these models to predict more useful movement features, we can expand the applicability of these models to additional use cases, and bring BCIs closer to everyday use. For this project, I used the same convolutional neural network (CNN) model as Chapter 3, and train the model on the same naturalistic ECoG movement data [Wang et al., f; Singh et al., b]. For this particular project, I am most interested in decoding wrist angles during arm reaching movements, as it presents the most useful feature for prosthetic applications, but will also explore other features to see which feature is most correlated with the neural data. An overview of the project pipeline is shown in Figure 6.1.

**Figure 6.1:** Overview of the CNN model used (HTNet [Peterson et al., b]) and the movement features I train the model to predict in this project. Inputs include the ECoG data with all ECoG channels during a time window, and a weight matrix when training a model across multiple participants. Convolutional layers in the network include a temporal convolution, depthwise convolution, and separable convolution. The model also includes a special Hilbert transform layer to transform the ECoG signal into the Hilbert signal. For the predicted features, we show examples of these features while a person is moving their arm from their lap (light pink) to their chest (dark pink). The different colors, blue, green, orange and yellow, show how the features are calculated, with cool colors representing regressed features (reach angle and magnitude), and warm colors representing classified features (reach region and reach quadrant).

### 6.3.1 Methods

**The Data**

The features of interest to this project are reach angle, reach magnitude, reach region and reach quadrant (see figure 6.1). Since all of these features are only relevant to movement events, I only include data from labeled movement events, and discard rest events from the AJILE dataset. The features can be split into regressed features, and classified features. For the regressed features, reach angle represents the angle difference between the wrist joint before the movement starts, to when the movement reaches its furthest displacement. We measure the angle based on the furthest displacement, rather than when the movement ends, because the movements often started and ended in the same place (the participants lap, for example). An example of how reach angle is calculated is presented in blue in figure 6.1. Reach magnitude measures the maximum displacement during the movement event, computed as the pixel different between the wrist joint at the beginning of the movement and at the point of maximum displacement. This feature is represented in green in figure 6.1.

For the classified features, reach region is measured by overlaying grid regions on the video data, and measuring which region the wrist joint is in. For this project, I extracted reach region at the start of the

movement to simplify the data. The regions generally map to areas in the hospital room, such as the bedside table or their head area. There are either 7 or 6 reach regions in the data, depending on whether the dataset is balanced or not. The regions for this feature are represented in orange in figure 6.1. Reach quadrant discretizes the reach angle feature into upwards, downwards, left and right wrist movements. Since most reaches occupy upwards or downwards reaches, we rotate the quadrant axis by 45 degrees to ensure that similar reaches are not broken apart. The mapping of these quadrants is shown in yellow in figure 6.1.

**The Model**

I adapt the HTNet model presented in Chapter 3 [Peterson et al., b] to more complex movement features. In brief, the HTNet model contains 3 convolutional layers with one Hilbert transform layer, which together converts the raw neural data to data-driven spatiotemporal features that was used to classify movement vs rest states. Peterson et al. [b] trained HTNet to work for individual participants, and also to generalize to unseen participants. Since this model was developed specifically for ECoG and with generalizability across participants in mind, we hope we can also expand the task generalizability of the model architecture to complex movement behaviors, like reach angle, over a variety of ECoG participants.

Training a model for each of the four features requires some differences in model architecture, primarily with the last layer and the loss function. In both of the regression models, I change the final classification layer in HTNet to a dense layer with a single output node that predicts the value of the feature. For reach magnitude, I use the standard mean squared error loss to calculate the error between ground truth and predictions. For reach angle, a different loss function is required for training. Because of the cyclical nature of the labels (-180 and +180 are the same value), a custom loss function is needed:

$$\arctan(\sin(y_{pred}, y_{true}), \cos(y_{pred}, y_{true}))^2$$

This function outputs the smallest angle between the truth and the prediction, unlike in the case of MSE. An alternative loss function that was also shown to work is cosine similarity [Google].

For the classification problems, both reach region and reach quadrant use the same loss of categorical crossentropy. The output of the model is this case is a dense layer with nodes equal to the number of classes, and a softmax activation to determine which class the model predicts as most likely.
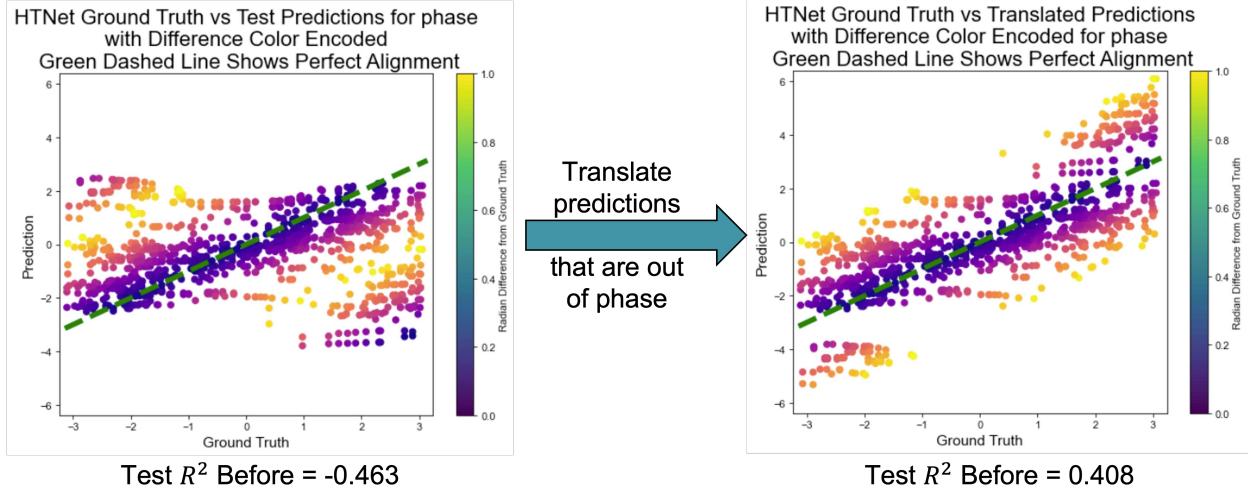
In previous iterations, HTNet needed to predict movement features on a held out day. Due to the inherent variability of brain data, there can be vast differences in how someone's brain activity looks from one day to another. Thus, for the regression problems (reach angle and magnitude), I simplified the problem down to leave-one-out cross validation (LOOCV) on a single day, which has less variability in the brain data, and should make it easier for the model to generalize. In the case of subject 01, I also tested LOOCV across all days of data, to get a sense of the performance trade-off between more data and variability across days. For the classification problems (reach region and quadrant), I compared the performance when balancing the classes by downsampling vs leaving them unbalanced with the initial skew that existed.

To test the validity of this approach, I use standard machine learning metrics. For each of the possible complex movement features, I compare the neural network model's performance on either R-squared or F1 score. For reach angle, I also look at the value of the loss function on the test data after training. Because of the cyclical nature, I have found that the initial R-squared correlation does not provide a comprehensive picture of model performance on reach angle. Thus, I also look at the model loss, and an adjusted R-squared value to evaluate reach angle performance (more details below).

To better understand the performance of the trained HTNet models, I conduct several interpretability analyses. First, I examine how the model's performance varies across different participants and relate it to the amount of training data available for each participant. This analysis should give a sense of the amount of data necessary for effective training, and assess the impact of day-to-day variability in the data on model performance. Additionally, I explore instances where the model fails to make accurate predictions, investigating any patterns in the types of errors it makes. This analysis helps to identify whether the model is relying on shortcuts during training.

**Verifying Custom Loss for Reach Angles**

To ensure that the custom loss function would work for reach angles, I first tested the loss on some synthetic data. The synthetic data contained a series of sine waves with some noise added to each wave. I uniformly generated the starting phase angle across data points, and then trained the HTNet model to predict the starting phase angle. Training the HTNet model to predict phase angle from sine waves resulted in a final train loss of 1.76 and final test loss of 1.88, and an initial R-squared train value of -0.367 and test value

107

Test $R^2$ Before = -0.463          Test $R^2$ Before = 0.408
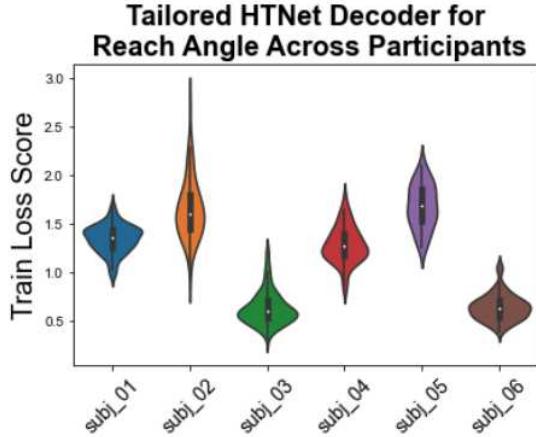
**Figure 6.2:** Synthetic test data ground truth vs HTNet predictions for the initial predictions and the adjusted predictions. Initial predictions show a poor R-squared value (-0.463), but when adjusting the predictions that are out of phase, the model shows a much better R-squared value (0.408). The color encoding shows how close the predictions are to the ground truth in terms of the angle difference between the two. Darker colors mean closer predictions.

of -0.463. Looking at the model's test predictions vs the ground truth, we see that the initial R-squared values don't quite capture how well the model is predicting the phase angle. To align the R-squared values to the performance of the model, I found the predictions that were "out of phase" with the landscape of the loss function, and then adjusted them by $2\pi$ to get them back in phase. In practice, this means adjusting predictions that are greater than the ground truth plus $\pi$ by $-2\pi$, and adjusting predictions that are less than the ground truth minus $\pi$ by $2\pi$. An example of this adjustment is shown in Figure 6.2. The adjusted predictions show an R-squared value of 0.408, indicating that the custom loss function is learnable by the HTNet model.

### 6.3.2   Results

**LOOCV for Reach Angle**

Participants 3 and 6 showed decent loss scores on the training data during leave-one-out cross validation, with participants 1 and 4 not far behind (figure 6.3). However, the model did not perform so well when generalizing to the left out datapoint (figure 6.4). The loss scores really jump up when looking at the test data, with scores even as high as 12 in the trailing tails. Looking at adjusted test $R^2$ scores shows that in
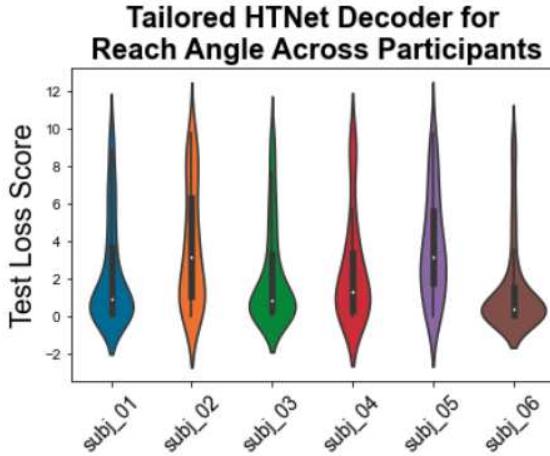
**Figure 6.3:** Train loss score for 6 of the 12 participants on reach angle LOOCV

| Subject Number | Adjusted $R^2$ Test Score |
|---|---|
| Subject 01 | 0.1911 |
| Subject 02 | 0.0536 |
| Subject 03 | 0.1958 |
| Subject 04 | -0.1275 |
| Subject 05 | -0.0980 |
| **Subject 06** | **0.3998** |
| Subject 01 All Days | 0.3462 |

**Table 6.1:** Reach angle adjusted $R^2$ test scores for each participant for one day of LOOCV, and for all days of Subject 01 in last row.

some cases, such as Subject 06, the model was able to perform decently (Table 6.1). However, most subjects still show relatively poor $R^2$ scores. In the case of Subject 01 on all days of data, we see an improvement in $R^2$ from just one day of data, but the score is still not as good as just one day of Subject 06. This indicates that perhaps the variability across days presents too much of a challenge for the model during training. Looking at each participant's predictions vs the ground truth helps to understand where the model tends to fail (figure 6.5). As shown in the figure, the trained models generally predict around the same value, such as in Subject 05, rather than matching to the broad range of ground truth values as shown on the x axis. It seems that the model tends to get drawn towards predicting $-\frac{\pi}{2}$, which would be equivalent to upward movements (the angles are flipped in the data). This makes sense that the model would tend to predict the most common value in the dataset (upward movements), indicating the need for data with an even distribution for training.

109

**Figure 6.4:** Test loss score (Un-adjusted $R^2$) for 6 of the 12 participants on reach angle LOOCV

| Data Type | F1 Test Score |
|---|---|
| Unbalanced | 0.27 |
| Balanced | 0.25 |

**Table 6.2:** Reach region f1 scores for balanced and unbalanced datasets
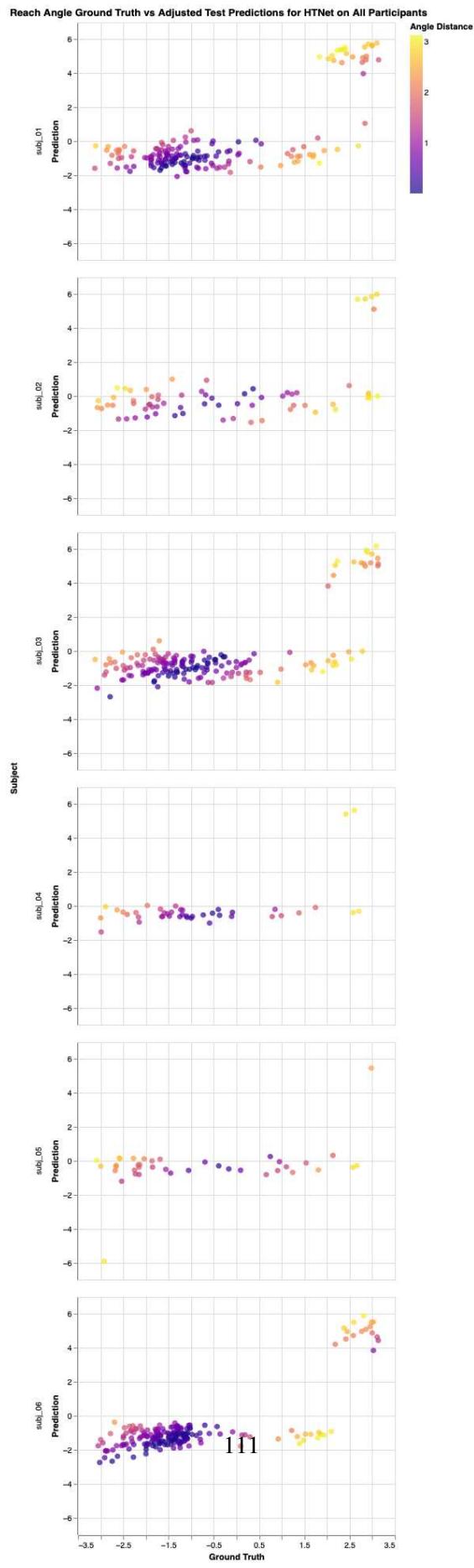
**LOOCV for Reach Magnitude**

I also looked at HTNet predictions for reach magnitude for Subject 1 on leave-one-out cross validation. As shown in figure 6.6, the model also performs quite poorly at predicting reach magnitude, again just predicting around one particular value, near 0.25. Looking at the $R^2$ score (0.02), the poor performance is corroborated. This again shows how the model is generalizing poorly to the held out datapoint.
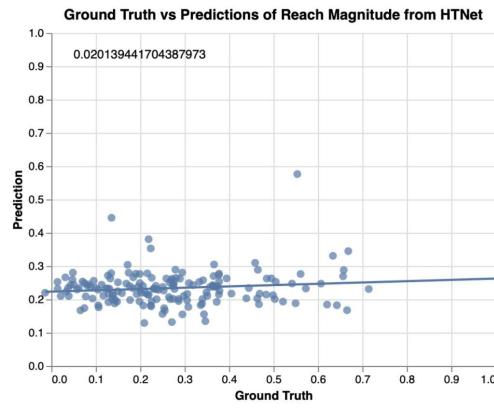
**Reach Region**

When looking at reach region predictions for Subject 1, we see that the unbalanced data appears to have better accuracy (Figure 6.7). However, if we dig deeper and look at the F1 scores, we see that both types of models perform similarly, and the performance of the unbalanced dataset is not as good as the accuracy shown (Table 6.2). Unfortunately, this means both models sit close to chance accuracy for 6 (balanced) or 7 (unbalanced) classes.
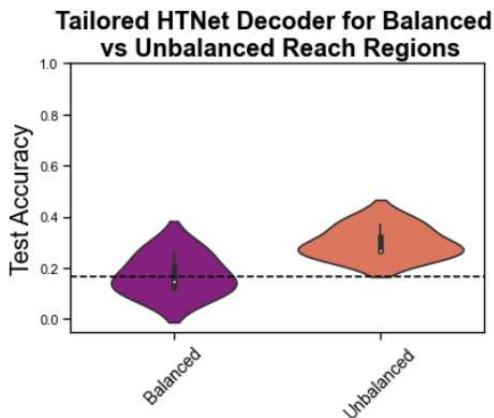
Because the model seemed to learn slowly, with the loss hardly decreasing between steps, I decided to explore the effects of different learning rates when training the model on the unbalanced reach regions. The results of this hyperparameter exploration can be seen in figure 6.8. It seems that increasing the learning
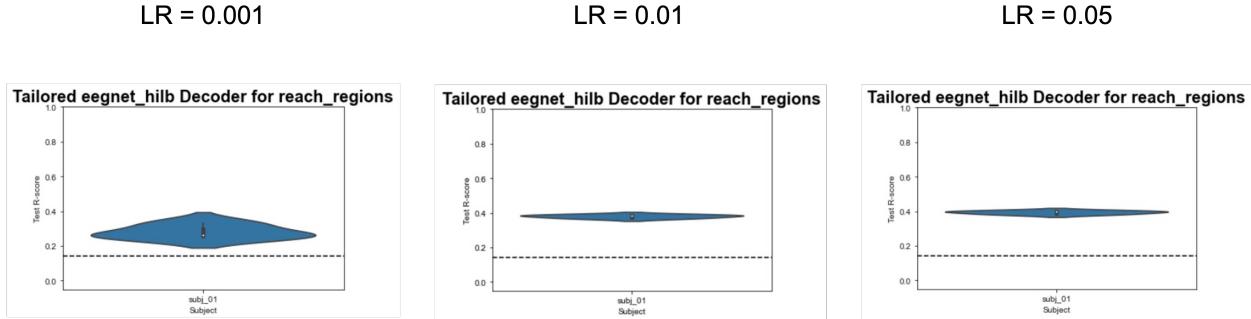
**Figure 6.5:** Reach Angle LOOCV Ground Truth vs Test Predictions for 6 of the 12 participants
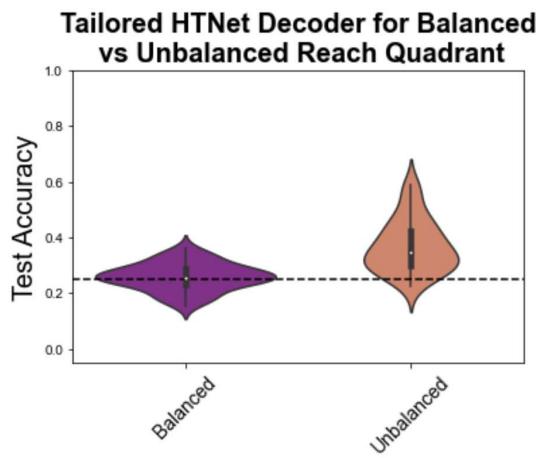
**Figure 6.6:** Reach magnitude ground truth vs test predictions for LOOCV on Subject 01



**Figure 6.7:** Reach region test accuracy on balanced vs unbalanced datasets. Dashed line shows chance accuracy for 6 classes

**Figure 6.8:** Effects of learning rate on unbalanced reach region test performance
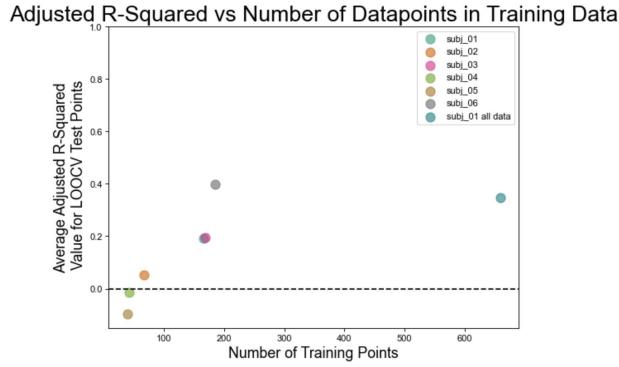


**Figure 6.9:** Comparison of Balanced and Unbalanced versions of reach quadrants data

rate decreased the amount of variance in the test performance. However, the effect was not that significant between 0.01 and 0.05, so future work should continue to use a learning rate of 0.01.

**Reach Quadrant**

When looking at reach quadrant, I also tested both an unbalanced and balanced version of the dataset. On the unbalanced dataset, the average test F1 score across participants was 0.21, and average test accuracy was 0.37, slightly better than chance, but likely a result of the unbalanced data (Figure 6.9 orange violin plot). When balancing the number of classes in the data, performance falls down to chance accuracy (0.26) (Figure 6.9). This corroborates the reasoning that the unbalanced data only performed decently because of the imbalance in the classes.
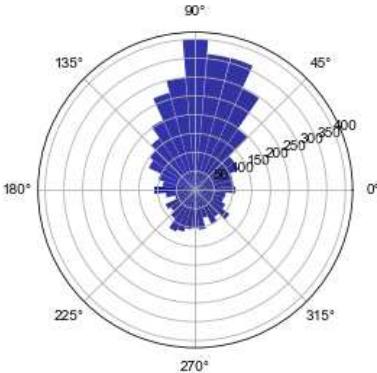
**Figure 6.10:** Reach angle adjusted R-Squared for ground truth and test predictions vs the amount of data in training set.
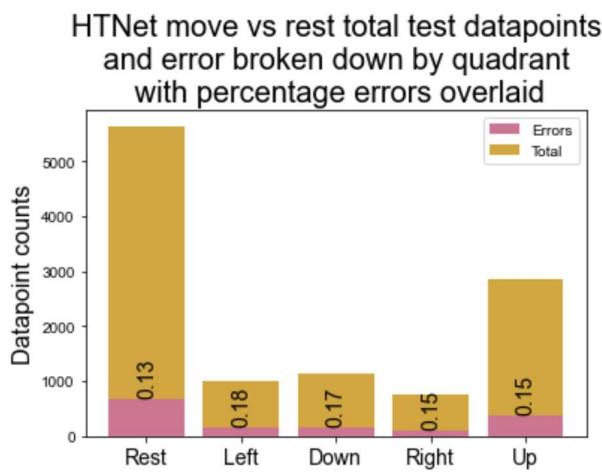
## Model Interpretation

To get a better sense of why the HTNet model might be performing so poorly on these various movement features, I looked into how the performance changed with more data. For example, in the reach angle predictions, the adjusted $R^2$ scores generally increased with more data, as shown in figure 6.10. However, when testing Subject 01 on all days of data, providing many more data points for training, we see that the performance boost does not follow the linear increase seen for one day of LOOCV. This indicates that perhaps using more data could help train the model, but at the cost of keeping the ECoG data from becoming too variable, as it does across days. Currently, the maximum number of data points available for one day is around 180, quite low for a deep learning model. One possible solution is to train the model on data from all but one participant, which would also increase complexity, but would also provide a lot more data for the model to learn from.

While working on these various features, I started to wonder if the previous HTNet model in Peterson et al. [b] had ended up just learning to distinguish between rest events vs upward arm movements, rather than movements in general. There is clearly a skew in the data towards upward trajectories, as can be inferred in figure 6.5, and made more clear in figure 6.11. To explore this possibility, I trained subject specific HTNet models for move vs rest, with the labels aligned to the quadrant the movement went towards. I then looked at the percentage of prediction errors in each bin (rest, left, down, right and up), with the results shown in figure 6.12. Each quadrant shows a percentage error around 0.16, indicating no significant difference between quadrants. Thus, it does not seem to be the case that the original HTNet model for move vs rest

**Figure 6.11:** There is a clear skew towards upwards movements.



**Figure 6.12:** HTNet move vs rest decoder errors broken down by reach quadrant

only learned a shortcut to distinguish between rest and specific kinds of movements.

### 6.3.3  Discussion

Overall, none of the behavioral features were learned particularly well by the HTNet model. Perhaps hyperparameter tuning or training with more data could help the model improve, as evidenced by Figure 6.10. Another consideration is the temporal nature of the behavioral features. Because these the participant does not just instantaneously get to a particular angle or region, it may be prudent to consider models that predict the features continuously. Kalman filters have shown significant success in continuous decoding BCI paradigms [Anderson et al., b; Leuthardt et al.]. Using Kalman filters or Recurrent Neural Networks could

115

be more viable at learning to predict these kind of features, since they can continuously decode the signal, and thus use the previous time points to help guide predictions. Future work should also look to consider balancing the regressed features so that there is not as much of a skew towards particular values. This may help the model from getting stuck in predicting the most common value. However, a more important consideration could be that the ECoG data does not contain a strong enough signal to decode these particular fine-grained features.

Although the HTNet model was not able to learn the movement features tested here, there was still a lot learned from this project. For example, using more training data unsurprisingly allowed the model to learn better, but these gains did not go as far if the data was pooled across multiple days. This highlights the trade-off in accounting for the variability in brain data, vs gathering more data for training. In general, the skew in the datasets towards particular kinds of movements made learning these features difficult for the model. For example, in both reach regions and reach quadrants, the HTNet model seemed to predict better for the unbalanced versions of the dataset, but looking at the f1 score highlighted how the model was actually learning to exploit the imbalance. Overall, these results highlight the need for models that can better handle this highly variable and skewed type of data that exists in naturalistic settings.

## 6.4   Self-Supervised Learning

In this project, we apply four SSL pre-tasks to our naturalistic ECoG dataset. From this dataset, we have many days and hours of unlabeled data from all ECoG participants. However, labeling all of that data is costly and time-consuming. Therefore, in this project, we wanted to leverage the benefits of SSL, which does not require data to be hand-labeled, and make use of this plethora of data. Our first SSL pre-task was inspired by work from Banville et al., and uses a relative-positioning task in the time dimension of the ECoG data. The next pre-task was inspired by Sarkar and Etemad [2020], and uses a set of signal transformations on the data to create pseudo-labels. The third pre-task uses a method called Contrastive Predictive Coding (CPC) [Oord et al.]. Our last pre-task is a novel task which uses features of the power of different frequency bands to generate the pseudo-labels for pre-task training. For the downstream tasks, we classify two different ECoG paradigms, one for movement (hand movement vs rest), and one for speech (subject speaking vs silence). For both tasks, there exists labeled data for at least three participants. We only

test the speech downstream task on the relative positioning and signal transformation pre-tasks. We then compare the test accuracy of each self-supervised method to a supervised decoding algorithm previously used on the AJILE dataset (see Chapter 3) [Peterson et al., b]. This project was completed in collaboration with Gabrielle Strandquist and Zeynep Toprakbasti.

### 6.4.1 Methods

**Neural Network Model**
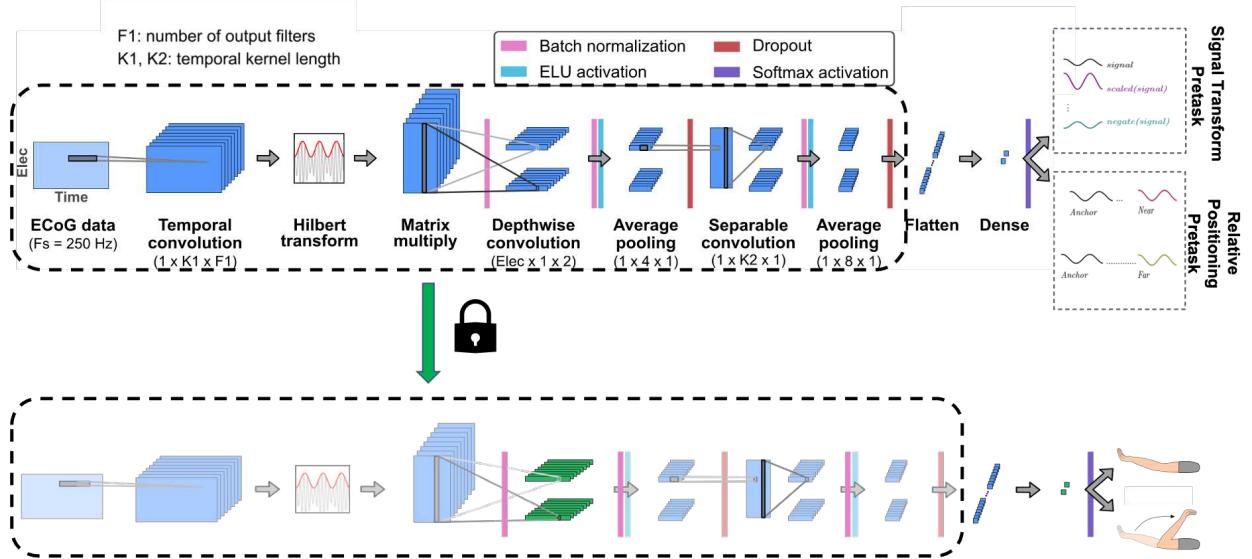
For this project, we used the same HTNet model for decoding ECoG data, see Chapter 3 and [Peterson et al., b]. The model implements a CNN architecture, which includes three convolutional layers, and a special Hilbert transform layer (see Figure 6.13 for details of the architecture). We implemented all model training in Python v.3.8 with Tensorflow. Code for the relative positioning and signal transformation pre-tasks can be found here: `https://github.com/strandquistg/ez-ssl/tree/main`. Please reach out to the author for code on the other pre-tasks.

**Datasets**

Previous work in our lab curated labels for wrist movement events in our lab's ECoG dataset for 12 participants, of which further details can be found in Peterson et al. [2021]; Wang et al. [2018]; Singh et al. [b]. Using the same dataset, labels were also obtained for speech onset events for 3 participants. Each subject has data from 2 separate days, and speech-events were selected with at least 1 preceding second of silence. Contrasting silence-events were selected with at least 2 seconds of silence before and after the event. Movement events were epoched at 1 second before and after the event, while speech events were epoched at 3 seconds before and after the event. The classes were balanced for both movement and speech events.

In the case of the relative-positioning and signal transformation pre-tasks, we used the data from the already labeled movement and speech datasets to train the pre-task and then fine-tune the downstream models. We used an 80/20 split for our pre-task and downstream data. In the pre-task, instead of using the labels that were attached to the ECoG events, we used the pseudo-labels generated by the pre-tasks.

Considering that one of the benefits of SSL is that we can make use of unlabeled data, we decided to follow a different approach for the pre-task data in our CPC and Domain Features pre-tasks. In these

**Figure 6.13: HTNet architecture during the pre-task and downstream.** The top half of the figure shows HTNet during the pre-task. During the pre-task, the model is trained on one of the four pre-tasks (see Section 6.4.1 for details about the pre-tasks). In this stage, all layers in the model are trainable, so that the model can hopefully learn useful representations of the data during pre-task training. The bottom half of the figure shows the model during downstream training. For the downstream, the first layers up to the flatten layer from the model from the pre-task are loaded in (layers in the dashed box) and a new flatten and dense layer are attached to fit the new task. All layers except for the depthwise convolutional layer and the dense layer are frozen and unable to be trained in the downstream task. Then the model is trained on either the movement or speech task, depending on what data was used in the pre-task.

pre-tasks, we extracted continuous unlabeled ECoG data from a random time point during the middle of the day for each participant. The goal of this approach was to simulate a live recording session, as if the model were pre-trained on a contiguous chunk of neural data from the participant before starting their actual neural decoding session. Before selecting a time for continuous ECoG extraction, we filtered out times with atypical activity, such as seizures. We chose to take activity during the middle of the day (around noon), as this is when participants would likely be most active. We left out the last day of each participant's hospital stay when extracting this contiguous data, to ensure that no test data ended up in the pre-task training data. To understand how much contiguous data would be needed for sufficient pre-training, we tested three amounts of contiguous data - 10 minutes, 20 minutes and 80 minutes - resulting in 600, 1200, and 4800 one-second ECoG epochs respectively. Of this data, 80% was used for training the pre-task, while the remaining 20% was used for validation and testing of the pre-task.

**Pre-tasks**

The overall goal of SSL pre-tasks is to automatically derive pseudo-labels from data to train HTNet in classification tasks. The learned weights from these pre-tasks can then be saved in the HTNet model for the downstream tasks. With the exception of the loss function, the HTNet parameters were identical for all pre-tasks (see Chapter 3 for details on model parameters). In the training period, we established a patience level of 15 epochs, and trained for a maximum of 64 epochs.
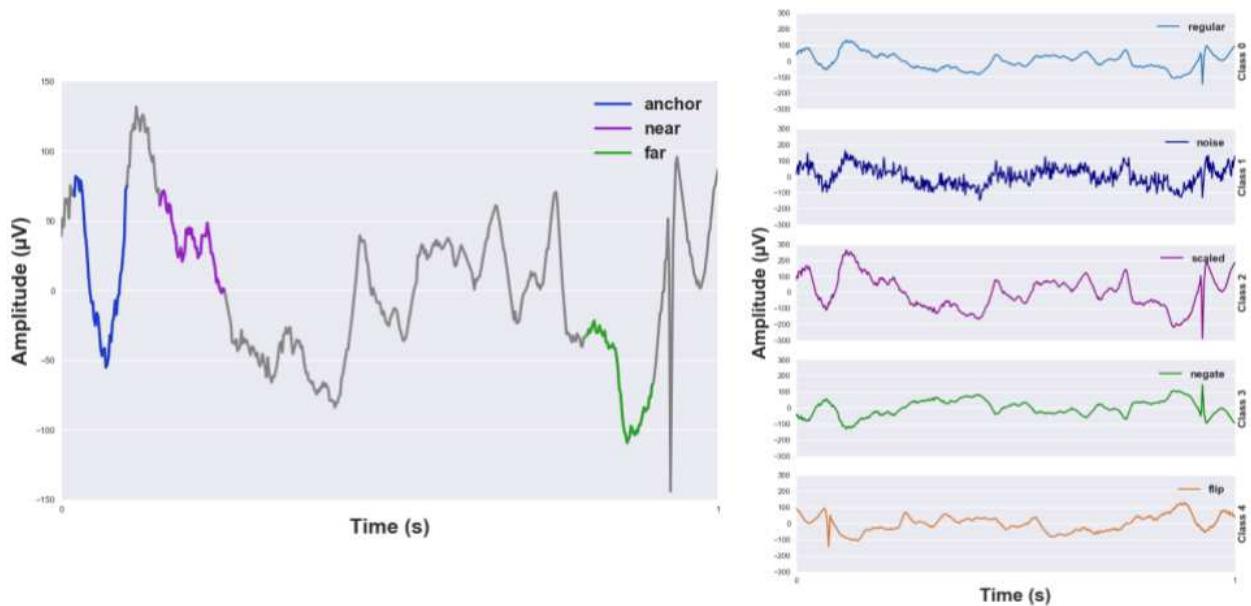
**Relative Positioning**    The relative positioning pre-task is designed to learn meaningful representations of time series ECoG data by discovering differences in brain activity that occur close together versus far apart in time. To design this, we considered each epoch as being an "anchor" point by which to compare the temporal proximity of subsequent epochs. For each anchor, we selected 3 epochs that were "far" and 3 epochs that were "near" to the anchor, as determined by hyperparameters. The hyperparameters for determining near and far were experimentally chosen to be 6 and 36 epochs, respectively, based on model performance. This means that for an epoch to be considered near, it could be no more than 6 chronological epochs away from the anchor. For an epoch to be considered far, it had to be at least 36 chronological epochs away from the anchor. We note one exception with subject f0b, whose smaller than average dataset required the far parameter to be set to 20. Once selected, the anchor was separately concatenated to each far and near epoch,

yielding paired epochs that were twice the original length. Each pair was then labeled as near or far. Since there are only two classes in this pre-task the loss in HTNet was set to binary cross-entropy. See the first panel in Figure 6.14 for visual representations of the relative positioning pre-task.

**Signal Transformation** For the signal transformation pre-task, we considered a different way of learning representations of our data by having the model learn different patterns of ECoG. For this design we applied four separate transformations to the entirety of the dataset which can be seen in the right plot of Figure 6.14. Each transformation was applied to all epochs, generating five times as much data as the original dataset. Given the four separate signal transformations, resulting in five classes for training, the loss we used to train was categorical cross-entropy. The four transformations are listed below:

1. Noise added - adds noise to the original signal by generating values using the numpy random.normal function in a specified range, then adding these values to the original signal. For both event types we experimentally chose 10 as our upper range.

2. Scaled - scales the signal by multiplying the entire signal by a specified constant. For both event types we experimentally chose 15.

3. Negated - negates the signal by a simple multiplication of -1 on the entire signal.

4. Flipped - flips the signal in the time domain by reversing the signal's order using the numpy.flip function.
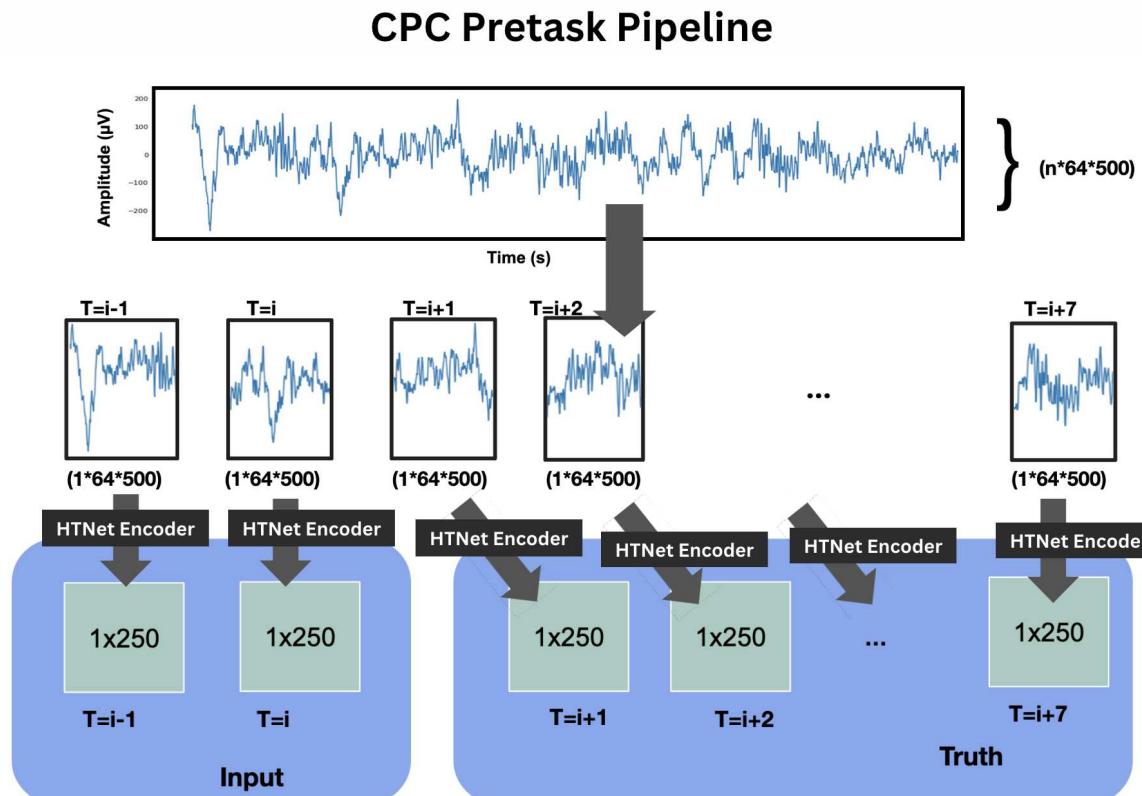
**Contrastive Predictive Coding** Drawing inspiration from Oord et al., we use the CPC approach on our ECoG data. The HTNet model is given a fixed length of consecutive ECoG epochs and is tasked with predicting the latent representations of the subsequent sequence. This task initially involves training the model to distinguish between correctly sequenced and shuffled ECoG sequences. The effectiveness of the predictions is assessed using the Info Noise-Contrastive Estimation (Info-NCE) loss function from Oord et al., which determines how representative the predicted ECoG sequence is of the actual subsequent sequence. We found that the optimal combination of input and prediction sequence lengths for this pre-task is 2 epochs and 7 epochs of ECoG data, respectively. Here, "optimal" refers to the hyperparameters impact on enhancing

**Figure 6.14: Relative Positioning and Signal Transformation Pretasks.** The left plot shows how epochs were selected for the relative positioning pre-task. An anchor region, shown in blue, is closer to the purple region than the green region, chronologically in time. The anchor is concatenated with the purple region and together they are labeled 'near' as a single joint data point. Similarly, the anchor and the green region is concatenated together and labeled 'far'. The right plot shows how epochs were designed for the signal transformation pretask. Four separate transformations were applied to the entire dataset and labeled as a separate class, for a total of 5 classes including the original signal. From top to bottom, the signal transformations are original, noise added, scaled, negated, and flipped. Both plots show a real signal from a single channel from participant cb4.

downstream decoding performance, rather than its success in the pre-task itself. Notably, while predicting shorter sequences within the pre-task tends to be easier and achieves higher accuracies, these parameters are less beneficial for downstream tasks.

To encode the ECoG data into the latent space, we used the first many layers of HTNet. We truncated HTNet at its last pooling layer, stripping the model of the output layers. This produced latent representations of the ECoG data in 250 dimensions. This latent representation from HTNet was then fed into the rest of the CPC architecture to complete the predictive task and evaluated with Info NCE loss. For a visual representation of the CPC mechanism, refer to Figure 6.15.



**Figure 6.15:** Contrastive Predictive Coding adapted for ECoG data. A sequence of $n$ ECoG epochs are separated into a input sequence to be provided for the model as context in order to make the prediction, and a follow-up sequence which the prediction is compared to. In the experiment, the optimal input length given limited resources was found to be 2 epochs. The following 7 epochs are reserved for the true sequence. Figure adapted from Zeynep Toprakbasti's honor's thesis.

**Domain Features**   For our final pre-task, we developed a novel pre-task designed specifically for ECoG data. In the pre-task, we asked the HTNet model to predict the power of different frequency bands in the ECoG signal. We calculated the power of each frequency band using the fast fourier transform [Al-Fahoum and Al-Fraihat]. The frequency bands we calculated the power for are low-frequency oscillations (LFO) from 0.5-4Hz, Alpha from 8-14Hz, Beta from 14-30Hz, Low Gamma from 30-60Hz, Gamma from 60-100Hz, and High Gamma from 100-200Hz. For each frequency, we created a single label by averaging over all channels at all time points. We then normalized the power labels between 0 and 1 for model training, and used mean squared error as the loss function. To train this pre-task, we used 20 minutes of consecutive ECoG data (1200 data points), as described in the data section.

**Downstream**

The downstream task involved either decoding the movement or speech data from the ECoG data of one participant. The downstream speech task was only used in the relative positioning and signal transformation pre-tasks, while all pre-tasks were evaluated on the downstream move vs rest task. For each participant, we trained three different downstream models to get a sense of the average performance for each participant. To do this, we loaded in the pre-task model and kept the first layers up to the flatten layer in the downstream model (dashed box in Figure 6.13), and then added on a new flatten and dense layer to create the downstream model. We then froze all layers except the depthwise convolution and the dense layers. This allowed the representations from the pretask to be maintained, while still adapting the model to the new task. We chose to also train the depthwise convolution during the downstream following the transfer learning results from Peterson et al. [b], which show that also training convolutions during the transfer step helps improve performance. After creating the downstream model, we first tested the model on the human-labeled data, without re-training, to see how the pre-task representations performed, later called the pre-task model when reporting accuracies. Then we trained the downstream model with the human-labeled data. For training, we established a patience level of 15 epochs, and trained for a maximum of 64 epochs, and used binary cross entropy as the loss for both movement and speech. After training, we tested the downstream model again to see how re-training the model improved performance.

For comparison, we also trained a supervised model on the same data as the downstream to compare to

the traditional approach for brain decoding. The main difference between these models is that this model is randomly initialized and all layers are trained, while the downstream model started from the pretask weights and had most layers frozen. We followed the same 3-fold per participant approach as we did for the downstream model. This model is referred to as the supervised model in the Results section.

## 6.4.2 Results

**Pre-tasks**

|  | a0f | cb4 | abd | b45 | b4a | c95 | ec1 | ec7 | f0b | f3b | fcb | ffb | Chance |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RP Pretask** [%] | 53 | 56 | 54 | 55 | 51 | 51 | 54 | 53 | 46 | 57 | 50 | 54 | 50 |
| **ST Pretask** [%] | 57 | 59 | 60 | 60 | 59 | 60 | 60 | 60 | 60 | 60 | 60 | 40 | 20 |

**Table 6.3: Movement Data for Relative Positioning and Signal Transformation Pre-task Test Accuracy.** Accuracy from 12 participants compared to random chance is shown in each column. Rows correspond to the two pretasks, where"RP" designates "relative positioning and "ST" designates "signal transform" pretasks.

|  | a0f | cb4 | b45 | Chance |
|---|---|---|---|---|
| **RP Pretask [%]** | 52 | 49 | 49 | 50 |
| **ST Pretask [%]** | 79 | 80 | 80 | 20 |

**Table 6.4: Speech Data Pretask Test-data Accuracy.** Accuracy from 3 participants compared to random chance is shown in each column. Rows correspond to the two pretasks, where "RP" designates "relative positioning and "ST" designates "signal transform" pretasks.

**Relative Positioning**    The first row of Table 6.3 shows the HTNet accuracy for all 12 participants for the relative positioning pre-task trained on data epoched during movement events. Relative positioning performs poorly, sitting around chance accuracy (50%). The first row of Table 6.4 shows the HTNet accuracy for 3 participants trained on data epoched during speech events. Again, the model performs near chance accuracy in this pre-task. This indicates that the model did not learn this pre-task well, and likely will not transfer well to downstream tasks.

**Signal Transformation**    The second row of Table 6.3 shows the HTNet accuracy for all 12 participants for the signal transformation pre-task trained on data epoched during movement events. Signal transformation performs well, with the average accuracy around 58% (chance accuracy is 20%). The second row of Table
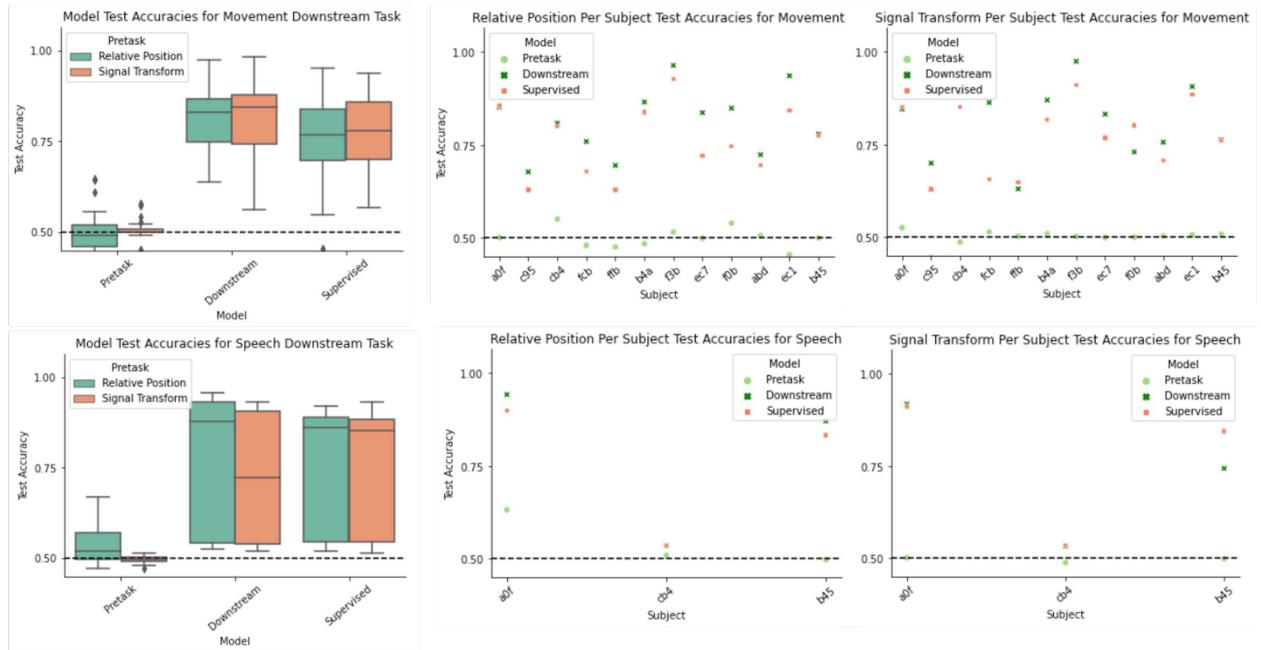
6.4 shows the HTNet accuracy for 3 participants trained on data epoched during speech events. Again, the model performs well in this pre-task, with average accuracy at 80%. Interestingly, pre-task model performance on speech data is superior to the model trained on movement data. This difference in performance could be due to the larger epoch lengths in the speech data.

**CPC**    Unfortunately, we do not have results on pre-task accuracy with models trained on CPC, and thus omit results here.

**Domain Features**    The first six panels of Figure 6.18 shows the model predictions for each frequency band in the pre-task, colored by participant. Three frequency bands show superior performance from the pre-task model - Beta, Gamma and High Gamma, with R-squared scores of 0.642, 0.257 and 0.678 respectively. The other three frequency bands show much lower performance (LFO R-squared = -0.298, Alpha = 0.201, and Low Gamma = -0.337). These differences in model performance across frequency bands could indicate that certain frequencies are more salient in the ECoG signal. However, these frequencies might not be the most related to our downstream tasks. Therefore, one improvement could be to include a custom loss which adds weight to the frequencies that we anticipate will be more relevant to our downstream task.

**Downstream**

**Relative Positioning**    Figure 6.16 shows the results from the downstream tasks for relative positioning and signal transformation. Here, we cover the results from the relative positioning pre-task. In the movement task, the average test accuracy across participants for the pre-task model was 49.25%. Again, this is the model right after pre-task training, with no additional fine-tuning. For the speech task, the average test accuracy across participants for the pre-task model was 54.61%. Thus, it seems the relative positioning model transferred slightly better to the speech task. The downstream movement model had an average test accuracy of 81.25%, and the downstream speech model had an average test accuracy 78.30%. This is somewhat surprising after the pre-task models showed that more useful features were learned for the speech task initially. Compared to the supervised model, which did not include any pre-training, we find that this model achieved around 76.79% in the movement task, and 75.98% in the speech task. Thus, the downstream model for movement performed significantly better than both the pre-task and supervised models (p-value

**Figure 6.16: Comparing results for the ECoG movement and speech tasks.** The top row shows results from the movement task, and the bottom row shows results from the speech task. The first column shows the average test accuracies across participants for the pretask, downstream and supervised model in both the relative positioning and signal transform task. The second two columns show individual participant averages, first in the relative positioning task, and then in the signal transform task.
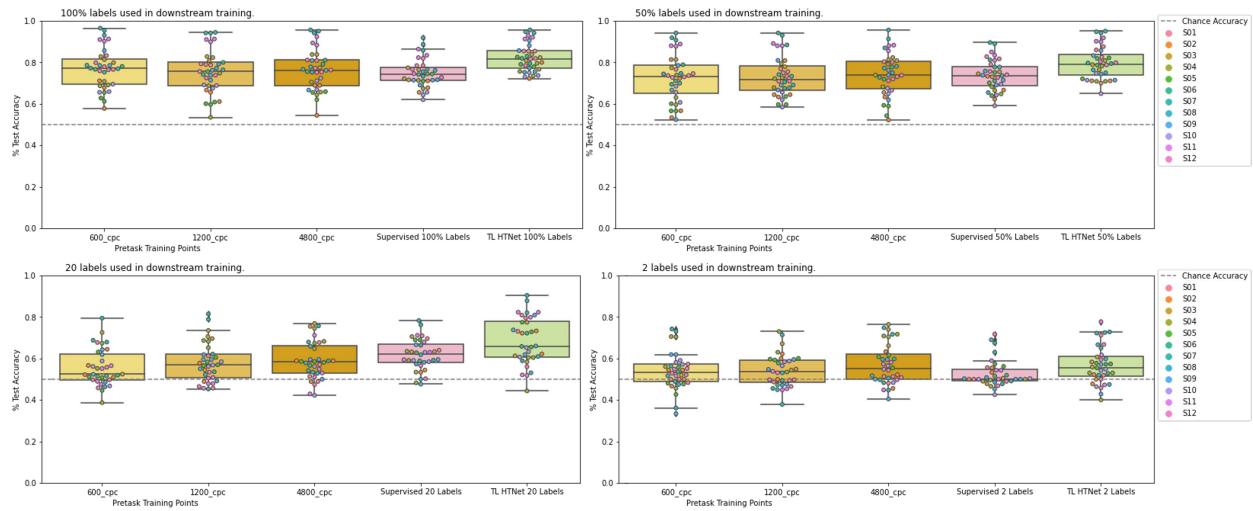
between pre-task and downstream = 2.63e-07, and p-value between downstream and supervised = 2.67e-04). However, the downstream model did not perform significantly better than the supervised model in the speech task.

**Signal Transformation**    Here, we cover the results from the signal transformation pre-task. In the movement task, the average test accuracy across participants for the pre-task model was 50.02%. Again, this is the model right after pre-task training, with no additional fine-tuning. For the speech task, the average test accuracy across participants for the pre-task model was 49.60%. Thus, it seems the signal transformation pre-task did not transfer well to without fine-tuning to either task. The downstream movement model had an average test accuracy of 81.37%, and the downstream speech model had an average test accuracy 73.15%. Like in the relative positioning pre-task, we find that the downstream model performed the best in the movement task (p-value between pre-task and downstream = 2.63e-07, and p-value between downstream and supervised = 8.19e-04), but that there was no significant difference in the speech task. Thus, it seems these two pre-tasks learn useful ECoG features related to movement, and can be fine-tuned to improve performance over regular supervised models.
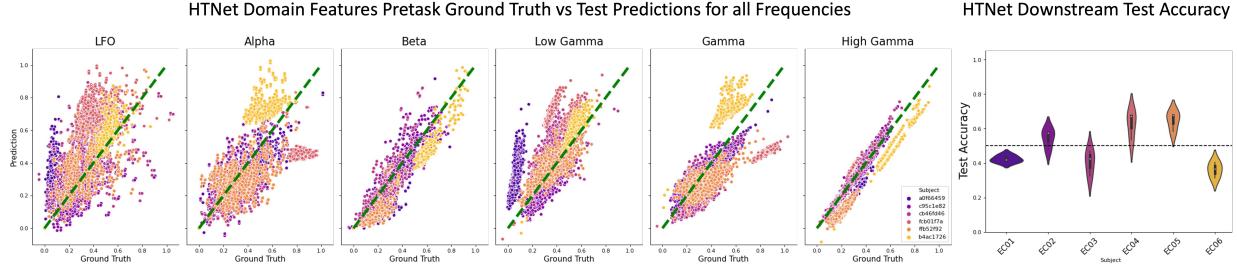
**CPC**    Figure 6.17 shows the results from the CPC method over four different variations of downstream training data (100% of available labels, 50% of available labels, 20 labels and 2 labels). All results are done on the movement downstream task. Across all variations we find that the CPC models trained on the largest amount of pre-task data (4800 data points), show the best model performance. More pre-task training data seems to provide the most boost in the 20 and 2 label downstream training tasks. In the case of 2 labels, the CPC model trained with 80 minutes of consecutive unlabeled data shows superior performance, even over the transfer learning approach developed in Peterson et al. [b]. In the cases of larger downstream training data, the CPC model generally performs slightly worse than the transfer learning approach. However, the CPC model has the benefit of not requiring a lot of labeled data across many participants for pre-training, giving this approach a more realistic use case.

**Domain Features**    In this last pre-task, we evaluate model performance on the downstream movement task. The results from fine-tuning the downstream model to movement decoding is shown in the last panel

**Figure 6.17:** CPC performance on downstream move vs rest decoding, with varying amounts of pre-training data and downstream finetuning data. All CPC models (yellow box plots) are compared to single subject models that are fully supervised (pink box plot), and to transfer learning models following the approach in Chapter 3 [Peterson et al., b] (green box plot). Top left panel shows CPC performance when using 100% of available datapoints for downstream finetuning. Top right panel shows CPC performance when using 50% of available datapoints for downstream finetuning. Bottom left panel shows CPC performance when using only 20 datapoints for downstream finetuning. Top left panel shows CPC performance when using only 2 datapoints for downstream finetuning. Figure adapted from Zeynep Toprakbasti's honor's thesis.

**Figure 6.18:** Pre-task and downstream results from the domain features pre-task. The first six panels show the pre-task results for each frequency band. The green dashed line indicates perfect alignment between model predictions (y-axis) and ground truth (x-axis). Scatterplot is colored by participant, see last panel bottom right for legend. The last panel shows the downstream results after pre-training the model on the domain features pre-task. Violin plots are categorized by participant.

of Figure 6.18. We find that the average test accuracy for the downstream model is 50.1%, indicating that the features learned from this pre-task did not transfer well to the downstream task, even with fine-tuning. This is somewhat surprising, as this pre-task was specifically designed for ECoG data. We may find that other ECoG signal features, such as phase, provide a better learning signal for model pre-training.

### 6.4.3   Discussion

In the relative positioning and signal transformation pre-tasks, we find that HTNet is better able to classify across 5 different signal transformations than it is across only 2 classes with the relative positioning. Despite 5 classes having greater complexity then 2, this is not necessarily surprising in light of the data we had to work with. Since we epoched our data chronologically but not consecutively in these pre-tasks, the relative positioning task likely becomes less meaningful, as there are gaps between epochs. Making use of the data that we later epoched consecutively may have improved performance for this pre-task. However, as we saw from the domain features pre-task, good model performance in the pre-task does not indicate strong performance in the downstream task. Of the pre-tasks, the domain features pre-task showed the lowest performance on the downstream movement task, even though it showed strong performance in the pre-task. On the other hand, the signal transformation and the CPC pre-tasks both performed well, with the CPC pre-task especially showing promise in very low downstream data regimes. In general, a dominant conclusion we drew from our literature review as well as these experiments is that the success of transferring a SSL model is only as good as the design of the pre-task. Thus, careful engineering of the pre-tasks is vital for

good results.

In the pre-tasks where we evaluated movement and speech tasks, the downstream model performed better for the movement task than the speech task. This could be because the movement task had more data points for each class, so the model is likely able to learn more about movement than speech. The ECoG input data is also much shorter in time for movement than for speech, so the model may have an easier time pinpointing features in the ECoG data when there is less data to parse in the downstream. Another consideration is that in the speech task, one participant in particular (cb4) does really poorly, pulling down the average. This participant likely does poorly because their electrodes did not have good coverage of speech areas, which could limit decoding accuracy.

Based on the results from CPC, it seems that further work should explore how much data can be used in the pre-task to fully boost model performance. In the three settings we tested, model performance consistently went up, so it is likely that more data for pre-training will help. Future work should also look to explore other SSL approaches. One promising approach is MYOW - Mine Your Own vieW - which was designed specifically with neural data in mind [Azabou et al.]. Other recent work on SSL in video and images also suggests additional promising ways forward for SSL, which could be applied to neural data [Assran et al.].

## 6.5   Neural Manifold Decoding
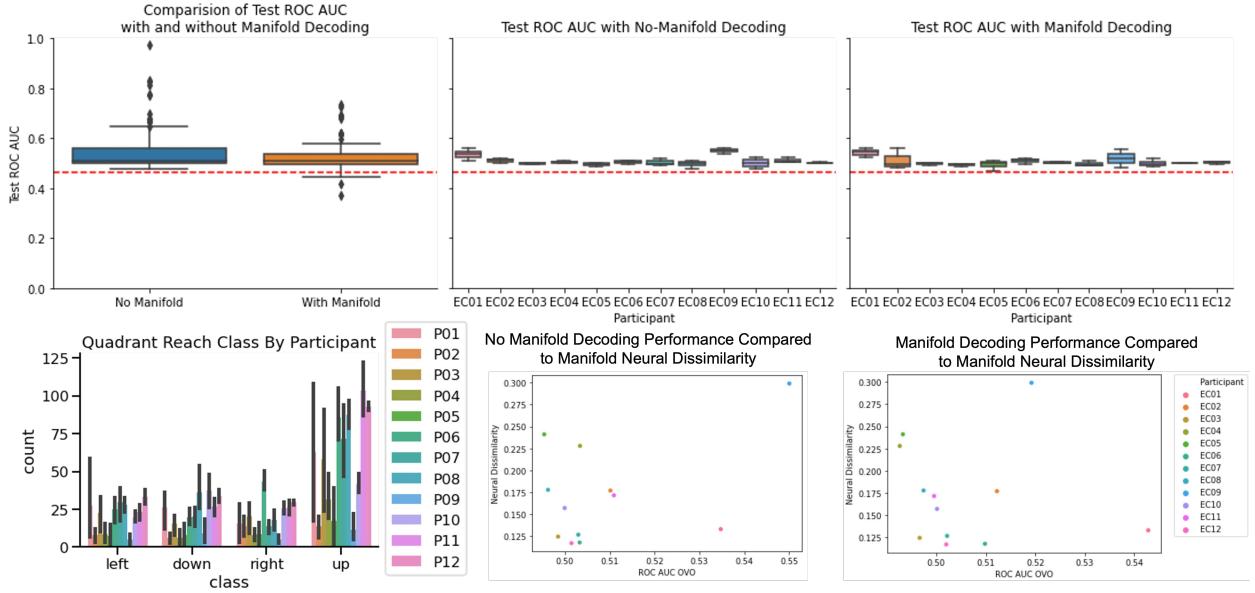
In this section, I report on a project that takes inspiration from the manifold decoding work discussed in Chapter 2. This work expands upon the results in Chapter 5, where we found that naturalistic ECoG data contains low-dimensional neural manifolds. As suggested by previous work, we can use these low-dimensional neural manifolds as input for neural decoders, and also align the manifolds for domain adaptation [Gallego et al., a; Farshchian et al.; Natraj et al., a; Bashford et al.; Pandarinath et al.]. One of the foci of this dissertation is the generalization of neural decoders across participants. Therefore, I was interested in applying techniques from the neural manifold adaption literature to improve cross-participant model generalization of naturalistic neural decoders. Specifically, I compare neural decoders either trained on the original ECoG data, the extracted ECoG manifolds, or neural manifolds that are re-aligned using Canonical Correlation Analysis (CCA). For these results, I move beyond the move vs rest decoding used in Chapter 3, and instead

classify reach quadrants (left, right, up and down; see Section 6.3 and Chapter 5 for details).

### 6.5.1 Methods

For all of the neural decoding results presented in this section, I trained the HTNet model to decode between four movement classes - left, right, up and downward reaches on individual participant data (see Section 6.3 and Chapter 5 for details on how these quadrant labels were extracted). The lower left panel in Figure 6.19 shows the amount of each quadrant label for each participant. Data for each participant was split by holding out the last day of available data for test data. The other remaining days of the participants' hospital stay was used for training and validation. For the no manifold decoding models, I trained the HTNet model on participant data that was projected to motor regions of interest (ROIs). The motor ROIs are described in Chapter 3, but essentially cover motor and sensorimotor areas of the brain. For the manifold decoding models, I trained the HTNet model on the neural data in the manifold approximations calculated using PCA. See Chapter 5 for more details on how the manifolds were calculated. In both these models, results are reported in ROC AUC OVO (one-vs-one), as the data labels were unbalanced. I used the same hyperparameter settings for HTNet as set in Chapter 3.
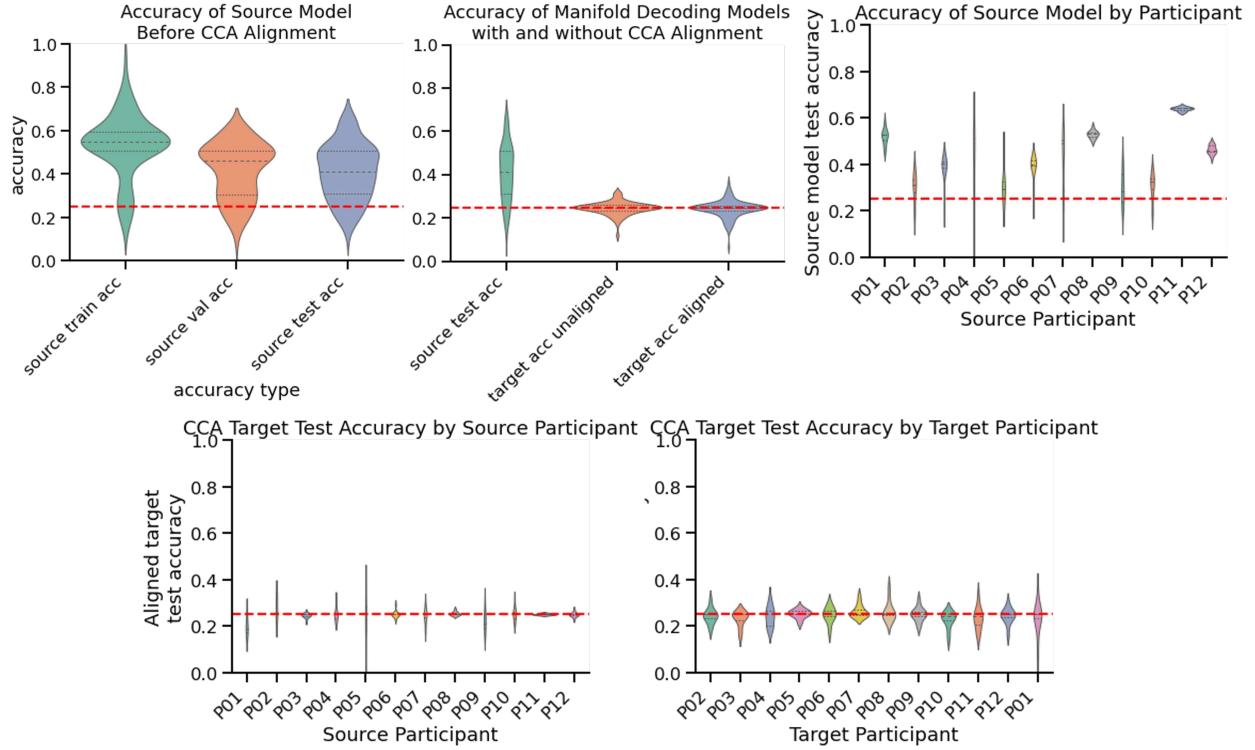
To perform manifold alignment, I used the CCA technique used in many other manifold papers [Gallego et al., a; Natraj et al., a; Pandarinath et al.; Safaie et al.]. CCA is a linear alignment technique which optimizes the correlations between data points to align data. However, CCA for neural manifold alignment requires that the neural manifold data is temporally aligned, otherwise no linear transformation from CCA will find a good alignment. In experimental setups, it is straightforward to ensure this alignment, but not so much in the case of naturalistic behaviors. This assumption for time aligned data may end up resulting in subpar model performance, as we will see. To use CCA, I use the manifold decoding models trained on one participant - the source participant - and then align the manifolds from a target participant using CCA. For the CCA models, I also trained the models on a balanced reach quadrant dataset, which was balanced by down-sampling the movement events to the class with the lowest amount of data. Thus, I report the results from the CCA models simply as percent accuracy. I then test the manifold decoding model on the realigned target participant data, without fine-tuning the model.

**Figure 6.19:** Results for no manifold and manifold decoding of reach quadrants. The top left panel shows the average test ROC AUC of no manifold decoders (blue box) and manifold decoders (orange box). The red dashed line shows the random chance accuracy of reach quadrant decoding given that the classes are imbalanced. The top middle panel shows per participant test ROC AUC for no manifold decoding, and the top right panel shows average test ROC AUC per participant. The bottom left panel shows the event count for each of the quadrant classes, colored by participant. The bottom middle and right panels show the test ROC AUC (x-axis) plotted against the within-participant manifold dissimilarity (y-axis) calculated in Chapter 5 for the no-manifold and manifold decoders respectively.

### 6.5.2 Results

The results for the no manifold decoding and manifold decoding models is shown in Figure 6.19. Between the two decoding models, we find little difference in their performance (no manifold average test ROC AUC = 0.509, manifold decoding = 0.504). When looking at model performance by participant, we see that P01 and P09 show the highest performance in the no manifold decoding setting. For the manifold decoders, P01 and P02 also perform highly, along with P06. To see if the initial geometry of the manifolds had any impact on performance, we also compared decoding performance to the neural dissimilarity metric which was developed in Chapter 5. In brief, neural dissimilarity indicates how closely aligned two manifolds are in electrode space. Therefore, if the neural dissimilarity between two manifolds is low, then the two manifolds include activity in similar brain regions. We find that there is a small trend between neural dissimilarity and the model performance of both models, such that lower neural dissimilarity results in improved performance.

132

**Figure 6.20:** Results from CCA manifold alignment. Top left panel shows train, validation and test accuracy on the source models. The top middle panel shows source test accuracy (green) in comparison to the target participant before calculating CCA alignment (orange) and after CCA alignment (blue). The top right panel shows the breakdown of test accuracy on the source participant. Bottom left panel shows the test accuracy on target data after CCA alignment, categorized by the source participant. Bottom right panel shows the test accuracy on target data after CCA alignment, categorized by the target participant.

However, the results from P09 seem to be an outlier to this trend. This could indicate that the models are able to learn to decode reach quadrants better when the neural activity is more closely aligned.

The results from CCA alignment are shown in Figure 6.20. Overall, we find that the CCA alignment approach does not result in much model improvement, with an average test accuracy improvement of -0.253%. We evaluate whether or not the source models perform well in the first place, and find that the source models perform well overall (train = 53.4%, validation = 41.7%, test = 41.2%, chance accuracy = 25.0%). Before aligning the data, average test accuracy on target manifolds is 24.7%. After CCA alignment, average test accuracy is 24.4%, slightly worse than before alignment. CCA alignment performance is highest when P02 was the source participant (average target test accuracy = 27.8%), though not by much compared to other participants. P01 performs the worst overall as the source participant with an average target test

accuracy of 19.9%. CCA alignment performance does not seem to be impacted significantly by which participant was used as the target. Overall, this indicates that CCA alignment does not improve decoding model performance on naturalistic ECoG data, but that the choice of source participant has some impact on alignment performance.

### 6.5.3 Discussion

Our results show that the HTNet model can decode interesting movement features using neural manifolds alone. These results are somewhat surprising, as the neural manifold approximations that we calculated are of a much lower dimension than the original ECoG data (10 dimensions vs around 64 channels). This ability highlights the potential for more efficient models with reduced parameter counts, which would be particularly beneficial in scenarios with limited data availability, like neural data. These results also open up avenues for further research into using neural manifold decoding, potentially leading to more streamlined and generalizable models.

In this project, we also observed that CCA did not significantly enhance model performance; in fact, it sometimes led to a deterioration in performance. This outcome is likely due to the inherent timing assumption in CCA, which may not align well with the temporal dynamics of our naturalistic movement data. To improve manifold alignment in future work, we suggest exploring alternative approaches that do not rely on such strict timing assumptions. Techniques such as LFADS and ADAN may prove useful for this [Sussillo et al., b; Farshchian et al.]. It may also be worth using techniques such as dynamic time warping to temporally align the data before completing the CCA step. See Chapter 8 for more details on how to potentially expand this work.

## 6.6 Summary

The previously unpublished projects in this chapter underscore the challenges of developing neural decoders for interpreting naturalistic human brain data. The first project involved enhancing the HTNet model to decode movement features beyond simple rest and movement states. However, these results showed the complexity of extracting sufficient signal from the brain for decoding such detailed movement information, as the models mostly performed at chance levels. We also saw that the models struggled to decode addi-

tional movement features due to the inherent class imbalance in the dataset. Future work may find more success in decoding features closely tied to neural representations of movement, such as 3D spatial location, re-sampling the data to minimize class imbalances, and by employing continuous decoding using LSTMs or Transformers, which could enhance performance and enable the development of neural decoders suitable for naturalistic contexts. The second project explored SSL approaches for training naturalistic neural decoders, revealing that certain SSL pre-tasks, particularly CPC, yielded model representations that transferred effectively to our naturalistic decoding tasks. Techniques like SSL are crucial for leveraging unlabeled data to train models, which will be a necessity for bringing BCIs into real-world settings. The final project employed a combination of the HTNet model and CCA for decoding neural manifolds, presenting an alternative approach to cross-participant generalization. Although we found CCA to be ineffective for alignment, neural manifold decoding shows promise, suggesting that further research in this area could lead to a faster and more efficient cross-participant generalization technique for naturalistic data, potentially surpassing the transfer learning approach developed in Chapter 3.

# Chapter 7

# Understanding Gender Gaps in Brain Computer Interface Performance from a User Experience Lens

## 7.1 Introduction

Brain-Computer Interfaces (BCIs) are capturing increasing attention from the public, fueled by high-profile demonstrations and investments by major companies like Neuralink and Meta. For instance, Neuralink recently showcased a live Twitter stream featuring a human participant controlling a computer cursor via an implant [Neuralink]. Similarly, Meta has unveiled a non-invasive neural interface designed to facilitate seamless interaction with augmented and virtual reality environments [Meta; CTRL-labs at Reality Labs et al.]. This expanding interest is further evidenced by the emergence of startups such as OpenBCI, Neurable, and Kernel, which are exploring consumer applications of BCI technology [OpenBCI; Neurable; Kernel; Douibi et al.]. These developments are not only promising for enhancing the lives of individuals with disabilities, such as those suffering from paralysis, but they also suggest a broader integration of BCIs into everyday life for the general population. The potential of BCIs to transform how we interact with technology, and revolutionize computing across various aspects of daily living, holds exciting possibilities for the future.

However, previous research has found that certain demographic and personality factors have an impact

on BCI performance, potentially limiting the effectiveness of these systems across users. For example, Randolph identified age, musical instrument proficiency, and the use of affective drugs as significant contributors to BCI performance. Additionally, personality traits such as motivation, confidence, and perceived relevance to the user have been recognized as important factors for BCI performance [Jeunet et al., a; Herbert; Chavarriaga et al.; Kleih and Kubler; Ahn and Jun]. Understanding these dynamics is crucial for developing BCIs that are accessible and effective across a diverse user base.

Previous research exploring the impact of human factors on BCI performance also suggests a notable gender gap in BCI performance, where men generally exhibit lower performance than women across various BCI experiments [Randolph; Allison et al.; Alimardani and Gherman; Leeuwis et al., a] For instance, a study by Randolph identified participant gender as a significant predictor of offline BCI performance. However, not all studies have consistently demonstrated significant statistical impacts of this gender gap [Jeunet et al., d,c], highlighting the need for additional research to fully understand how gender affects BCI performance. The interplay between the previously identified demographic and personality elements with gender further complicates patterns of BCI performance [Wood and Kober].

These findings echo and give substance to the many ethical concerns that surround the adoption of BCI systems into daily life. Neuroethics, a field dedicated to examining the moral issues associated with neurotechnology, explores critical questions related to agency, privacy, responsibility, authority, and justice [Klein et al.; Society]. The previous findings on performance gaps and user demographics raise the questions: Who will BCI systems work for? Who will be excluded from using BCI systems? These questions underscore the importance of addressing ethical considerations proactively during the development of BCIs, and incorporating neuroethics in the evolution of these technologies. As BCIs move towards becoming more widespread, it is crucial to consider who benefits most from these technologies and whether existing gender disparities will persist, diminish, or reverse. Considering these issues is essential for ensuring equitable access and effectiveness as BCIs evolve into a mainstream technology.

The performance disparities observed in BCI research, including gender gaps, may stem from various system components such as the decoding algorithms used, the nature of the experimental tasks, and the design of user interfaces [Chavarriaga et al.]. While much of the existing research has concentrated on enhancing machine learning models to better generalize across different users [Chavarriaga et al.; Nguyen

et al.; Vidaurre et al.; Fukuma et al.], including notable contributions from our own work [Peterson et al., b], this perspective focuses on elements of the user experience (UX). As noted in Lotte et al. [a], improvements to the machine learning pipeline are useful, but ultimately the user also needs to be considered in the development of successful BCIs. In this perspective, we will explore how elements of the user experience and interface design contribute to these performance gaps, and provide suggestions for UX solutions.

Thus, in this perspective, we examine a crucial ethical question: What gender biases are present in current BCI implementations, and how can we design these systems to minimize performance disparities between genders? To explore this question, we use the GenderMag Method, a systematic approach enabling software stakeholders to identify and address gender-related inclusivity issues [Burnett et al., a,d]. The GenderMag Method not only helps explain the observed gender performance gaps in BCI studies but also provides a framework for designing UX solutions to these inclusivity barriers. We will review existing literature on the impact of human factors on BCI performance and provide an overview of the GenderMag Method. We will then use the GenderMag Method to understand the gender performance gap in BCIs and develop targeted UX solutions. We will then discuss these solutions within the broader context of the literature, and speculate on potential shifts to the gender gap as BCI technology advances towards consumer applications.

## 7.2 Background

### 7.2.1 Impact of Human Factors on BCI Performance

As discussed in the introduction, BCIs have the vast potential to improve peoples lives outside of the laboratory. However, a significant challenge in translating BCI technology for broader use is that BCI tasks and models do not function effectively for all individuals. This issue, often referred to in the literature as "BCI illiteracy" or "BCI inefficiency", describes situations where BCI systems fail to work due to low classification accuracy from the underlying machine learning models [Allison and Neuper; Thompson; Lee et al., c]. Traditionally, this term has been used to suggest deficiencies in the brain signals of specific users rather than shortcomings in the BCI system itself, a notion that has sparked debate within the research community [Thompson]. Despite the label, the reality remains that BCIs do not yield functional outcomes for every

user. Intriguingly, research indicates that while individuals may struggle with certain BCI tasks, there typically exists at least one task that proves effective for each user, suggesting that universal BCI inefficiency is unlikely [Lee et al., c]. Extensive studies have explored the various factors contributing to BCI inefficiency, uncovering influences such as task type [Lee et al., c; Guger et al.; Dhindsa et al., b], model adaptation [Peterson et al., b; Nguyen et al.; Vidaurre et al.], feedback style [Sollfrank et al.; Jeunet et al., a], physiological differences [Leeuwis et al., b], personality traits [Baykara et al.; Hammer et al.; Herbert; Kleih and Kubler; Li et al.; Lotte and Jeunet], and demographics [Zich et al.; Randolph; Alimardani and Gherman; Allison et al.]. In this section, we will focus on how personality and demographic traits, i.e. key human factors, play a role in contributing to BCI inefficiency.

A comprehensive array of human factors influencing BCI performance has been identified through extensive research [Ahn and Jun; Jeunet et al., a; Chavarriaga et al.; Herbert]. For example, both Allison et al.; Zich et al. found age as a significant determinant, with younger participants generally exhibiting better BCI performance. Additionally, personal backgrounds involving past experience with art, music, and sports have been found to positively correlate with BCI performance [Dhindsa et al., b,a]. Skill-related factors such as fine motor skills, the vividness of visual imagery, and mental rotation capabilities also play a crucial role [Hammer et al.; Leeuwis et al., a; Jeunet et al., d]. Moreover, personality dimensions, including learning styles (active versus reflective) [Jeunet et al., d,c], and locus-of-control (personal belief in control of events around you) [Wood and Kober; Jeunet et al., a], further expand upon potential factors, suggesting that BCI performance is influenced by a blend of experiential and intrinsic personal traits. With many more factors identified in the literature [Ahn and Jun; Jeunet et al., a; Chavarriaga et al.; Herbert], the extensive list underscores the necessity for continued research into consistent and repeatable factors that shape BCI performance. This exploration is vital for advancing our understanding of how diverse human characteristics impact performance with BCI systems.

Despite the broad spectrum of potential factors impacting BCI performance, two human factors consistently emerge as significant in the literature: user motivation and user confidence [Herbert; Jeunet et al., a]. Many studies measure the impact of these factors using participants' responses to the Questionnaire for Current Motivation (QCM), which tracks motivational states across four dimensions: mastery confidence, fear of incompetence, challenge, and interest [Hammer et al.; Kleih et al.; Nijboer; Nijboer et al., a; Baykara

140

et al.]. Mastery confidence and fear of incompetence are also closely linked to user confidence. Research indicates that in settings with visual feedback tasks, participant performance significantly correlates with motivation levels, such that higher motivation typically enhances modulation of sensorimotor rhythms [Nijboer et al., a]. This correlation with motivation is supported by additional studies on sensorimotor rhythms and extended to other BCI tasks such as the P300 [Nijboer; Hammer et al.; Kleih et al.; Baykara et al.]. Intriguingly, findings also suggest that a higher fear of incompetence can negatively affect performance, underscoring that confidence, alongside motivation, is a crucial determinant of BCI performance [Nijboer et al., a]. This relationship is further exemplified in studies linking computer anxiety with decreased BCI performance, highlighting the broader impact of user confidence on BCI interactions [Jeunet et al., a].

Another common factor in the literature is gender. The gender [1] gap in BCI performance was first noticed in the early 2010's with work from [Allison et al.; Randolph]. In their study, Allison et al. discovered that female participants exhibited significantly better performance in SSVEP (steady state visually evoked potential) systems, which measure brain responses to visual stimuli to determine the focus of attention, and is often used for language inputs. Randolph similarly reported superior performance by women, in this case, in an offline study of mu rhythm modulation, which is a phenomenon in the alpha signal (8-13Hz) of EEG related to real and imagined movements. Subsequent studies have reinforced these findings. For instance, research by Alimardani and Gherman; Leeuwis et al. [a] consistently found women outperforming men in various BCI tasks. Although all these works surveyed different BCI tasks, they all found an impact of gender on performance. However, Jeunet et al. [d] was unable to replicate these gender-specific results. Other work related to gender in BCIs found that creating gender-specific models shows improved performance, particularly for women participants [Catrambone et al.; Cantillo-Negrete et al., b,a]. Additionally, the gender of experimenters themselves has been shown to influence outcomes [Wood and Kober; Pillette et al.]. Overall, these results indicate a complex relationship between gender and BCI performance, where the effect is not always straightforward or dominant. In this perspective, we hypothesize that "cognitive styles", which tend to cluster by gender, drive these performance gaps. This hypothesis offers a potential explanation for the inconsistent significance of gender effects across studies. We will explain and expand upon these cognitive

---

[1]Note that historically, studies have often conflated sex and gender, though modern standards distinguish between these terms [Committee on Measuring Sex, Gender Identity, and Sexual Orientation et al.]. In this work, we will equate the sex of the participant (female) with the gender of the participant (woman) to aid in interpreting these older papers.

styles in more depth in Section 7.2.2.

## 7.2.2  The GenderMag Method

The GenderMag method is a process for software developers and UX professionals to uncover, understand and fix inclusiveness "bugs" related to gender biases in software and user interfaces [Burnett et al., a,d]. The GenderMag method involves using gender-associated personas - "Abi", the "Pats" and "Tim" - to capture how people problem solve in different ways. The personas share five facets (cognitive styles), but each has their own facet values. Decades of research has shown that these five cognitive styles tend to cluster by gender, and was used to develop the facets of each persona [Beckwith and Burnett; Beckwith et al., a,b,c; Burnett et al., b,e; Cafferata and Tybout; Cao et al., b,a; Cassell; Cazan et al.; Chang et al., b; Charness and Gneezy; Coursaris et al.; Cunningham et al.; Dohmen et al.; Fisher and Margolis; Grigoreanu et al., a,b; Hallström et al.; Hartzel; Hou et al.; Huffman et al.; Lee and Ko; Margolis and Fisher; Meyers-Levy and Durairaj; Meyers-Levy and Loken; Norman; Riedl et al.; Rosner and Bean; Showkat and Grimm; Simon; Singh et al., a; Weber et al.]. The Abi persona covers cognitive styles typically shown in women, the Tim persona covers cognitive styles typically shown in men, and the Pat persona covers a mix of Abi and Tim's styles. Although named GenderMag, the focus of this method extends beyond gender alone. The emphasis on gender stems from the observation that certain cognitive styles cluster by gender. Thus, the core aim of the GenderMag method is to reveal and address *cognitive biases* in software, making it a tool for enhancing usability across a broader spectrum of users.

The five cognitive styles defined in the GenderMag personas are: motivations, information processing style, computer self-efficacy, attitude toward risk, and learning by process vs by tinkering. The motivations facet explains why a user wants to use a new technology. For example, a user might be motivated to use the technology solely to accomplish their task, whereas another user may be motivated to explore all functions of the technology, even if they are not related to the task at hand [Burnett et al., b,e; Cassell; Hallström et al.; Hou et al.; Margolis and Fisher; Simon]. Information processing style explains how users like to gather information in software systems, either comprehensively or selectively [Meyers-Levy and Durairaj; Meyers-Levy and Loken]. Individuals with comprehensive information processing prefer to gather a lot of information before acting (e.g., reading a whole page of documentation before making a change

to some code) whereas those with selective information processing styles prefer to gather small bits of information and tend to act on these bits of information more frequently (e.g., reading the relevant parts of the documentation page and acting on it as they come across it) [Cafferata and Tybout; Cao et al., a; Grigoreanu et al., a; Jernigan et al.; Lee and Ko; Lee et al., b; Meyers-Levy and Durairaj; Meyers-Levy and Loken; Riedl et al.; Simon]. Computer self-efficacy represents both a users confidence with technology, and their locus-of-control in relation to technology. For example, some users may have low confidence about using unfamiliar technology, and may blame themselves if they run into problems; whereas other users might have high confidence with unfamiliar technology, and will blame the system if things don't work [Beckwith et al., b,c; Burnett et al., b,e; Hartzel; Margolis and Fisher; Singh et al., a]. Attitude towards risk indicates a users level of tolerance for risk in software systems. Users with low risk tolerance tend to stick with the same features they already know to complete their tasks, whereas users with high risk tolerance are more likely to try new features [Cassell; Dohmen et al.; Charness and Gneezy; Weber et al.]. The last cognitive style, learning style, explains how users approach learning new software, either through tinkering or process oriented styles. Some users prefer to learn about new software in process-oriented ways (e.g., tutorials that show the steps of bringing different features together), while others prefer to tinker and explore, constructing their own understanding of the software (e.g., trying out different options and backtracking if needed) [Beckwith et al., c; Burnett et al., b; Cao et al., a; Chang et al., b; Hou et al.; Rosner and Bean]. Additional information about the three personas and their specific facet values can be found in the GenderMag method and kit [Burnett et al., d,a], and are summarized in Table 7.1.

The GenderMag personas and their facets help software designers detect cognitive biases during a specialized Cognitive Walkthrough (CW) [Wharton et al.]. Based on the cognitive styles identified for inclusiveness issues during their evaluation, evaluators can then tailor their design choices to address and eliminate these specific issues. To set up the walkthrough, evaluators first decide which persona they want to take the perspective of during the walkthrough. They also choose a task that can be achieved in the software of interest, and define the start and end states of that particular task. Once the persona and task are determined, evaluators walk step-by-step through the interface, asking the following set of questions from the perspective of the chosen GenderMag persona:

| Cognitive Style | Abi | Tim |
| --- | --- | --- |
| **Motivations** for using technology | Wants what the technology can accomplish | Technology is a source of fun |
| **Computer Self-Efficacy** about using unfamiliar technology | Low compared to peer group | High compared to peer group |
| **Learning Styles** for learning new technology | Process-oriented | Tinkerer |
| **Information Processing Style** for gathering information to solve problems | Comprehensive | Selective |
| **Attitude towards Risk** when using technology | Risk-Averse | Risk-Tolerant |

**Table 7.1:** Cognitive style facet values for the Abi and Tim GenderMag personas

SubgoalQ: Will <persona> have formed this subgoal as a step to their overall goal? (Yes/no/maybe, why, what facets are involved in your answer).

ActionQ1: Will <persona> know what to do at this step? (Yes/no/maybe, why, what facets...).

ActionQ2: If <persona> does the right thing, will s/he know s/he did the right thing and is making progress toward their goal? (Yes/no/maybe, why, what facets...).

Previous research utilizing the GenderMag method has demonstrated that addressing issues identified during GenderMag walkthroughs leads to more inclusive software [Vorvoreanu et al.]. In addition, several commercial and open-source software teams have successfully implemented GenderMag to identify and fix cognitive biases in their products [Burnett et al., c; Cunningham et al.; Mendez et al.; Shekhar and Marsden]. Further studies have analyzed the software solutions made following GenderMag evaluations, resulting in a set of "best practices" for gender inclusive design, particularly in the areas of information processing and learning styles [Steine-Hanson et al.]. Additionally, research by the GenderMag team has shown that integrating an inclusiveness perspective into computer science education not only enhances student retention but also fosters more inclusive design solutions, suggesting that increased awareness of inclusiveness issues promotes inclusive software and more equitable work environments [Garcia et al.].
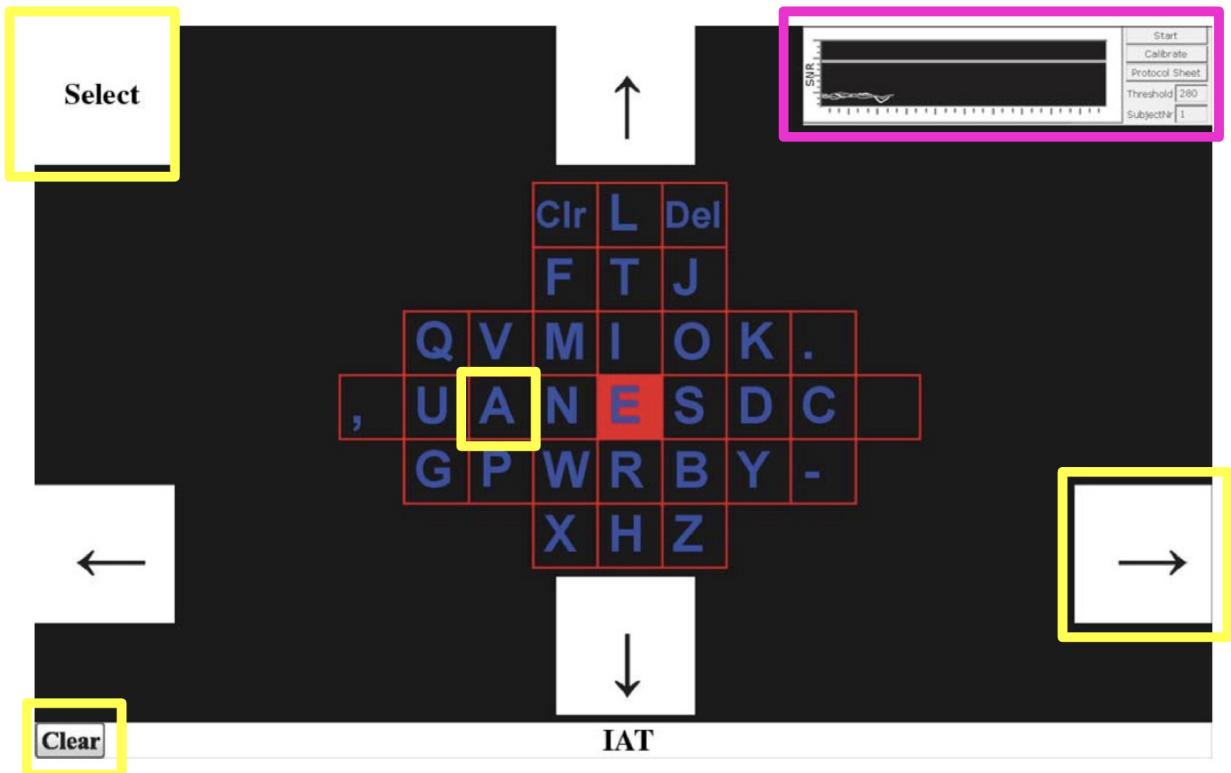
## 7.3 The GenderMag Framework in BCI tasks

In this section, we will leverage insights from the GenderMag personas to analyze the gender gap observed in BCI performance. Our goal is to propose design modifications to BCI systems that could potentially close this gap. Given that GenderMag highlights differences in cognitive styles, we propose that *cognitive* biases, rather than gender biases, underlie these performance gaps. The bulk of research indicates men often exhibit lower performance in BCI tasks, therefore, our analysis will concentrate on cognitive biases associated with the "Tim" persona, which embodies the cognitive styles commonly attributed to men. We will begin by examining the facets motivation, self-efficacy, and learning styles, which have evidence in the BCI literature for impacting performance. Following this, we will explore information processing styles and attitudes towards risk, aiming to identify additional potential cognitive biases in BCI systems. Through these examinations, we aim to suggest effective BCI design changes that enhance inclusiveness and performance across genders.

### 7.3.1 User Motivation

As discussed in the background section, user motivation appears to be a significant factor influencing BCI performance [Herbert; Jeunet et al., a]. Many studies have found a positive correlation between higher motivation levels and improved BCI performance. However, this does not tell us *why* a user might be motivated in a BCI experiment. According to the GenderMag method, individuals with cognitive styles similar to Tim's are motivated to explore and learn all available functionalities of a system, even if these functionalities are not necessary to complete their task [Burnett et al., d,a]. On the other hand, individuals with motivations akin to Abi are more focused on using technology solely to complete the task at hand. These differing motivation styles could significantly impact BCI experiment results and partially explain the gender gap in BCI performance. BCI experiments are highly directed and task-focused [Lotte et al., b], aligning more closely with Abi-like motivations. Therefore, individuals with motivations similar to Abi's may report higher motivation scores in BCI experiments, contributing to their higher performance.

If people like Tim are motivated to fully explore a system's functionality, how can we support these motivations in BCI experiments and systems to improve Tim's BCI performance? One approach is to incorporate dedicated "free-play time," allowing users to explore and familiarize themselves with the BCI

145

**Figure 7.1:** An SSVEP interface with areas of potential interest to Tim highlighted. Figure adapted from Allison et al.. Yellow boxes indicate areas where Tim may ask "what happens if I focus here?" In this case, focusing on the "A" will not produce any output, but may motivate Tim by learning additional features of the system. Pink box indicates a system component that Tim may be motivated to ask further questions about to learn about its functionality.

system. For instance, in an SSVEP setup, users could be given five minutes before experimental tasks to explore what happens when they focus on different parts of the user interface (Figure 7.1). Alternatively, experimenters could provide comprehensive training on all aspects of the system, including potential use cases and extensions. Encouraging users to ask questions about the system's functionality and applications can also be beneficial for these users. However, inclusivity is a balancing act, and Abi's motivations also need to be kept in mind when designing inclusive solutions. For example, Abi may not value free-play time before the experiment, so such time should be optional based on user preference, rather than mandatory.

### 7.3.2 User Computer Self-Efficacy

As highlighted in the background section, the cognitive facet of self-efficacy consists of a user's confidence in using a system and their locus-of-control. Previous research in BCIs has demonstrated that user confidence, measured by mastery confidence and incompetence fear, significantly impacts BCI performance [Jeunet et al., a; Nijboer et al., a]. Studies have shown that higher levels of mastery confidence are associated with improved performance, while increased incompetence fear is linked to lower performance [Nijboer et al., a]. Additional research has indicated that computer anxiety is also negatively correlated with BCI performance [Jeunet et al., a,d,c]. This suggests that a high level of confidence in using technology generally leads to better performance in BCI systems.

Regarding locus-of-control, researchers have found that a high locus-of-control is associated with improved BCI performance [Wood and Kober]. Interestingly, Tim's persona exhibits high confidence in using technology, which the literature suggests that this trait should lead to better BCI performance. However, Tim also tends to blame the technology if things go wrong, indicating a low locus-of-control with technology for Tim. In contrast, Abi demonstrates a high locus-of-control when it comes to technology, as she tends to blame herself for any issues that arise with technology. This difference in locus-of-control with technology may shed light on why men often perform poorly in BCI studies. Tim is likely to blame the BCI system or the machine learning algorithm, rather than adjust his neural strategy to improve cooperation with the system. Thus, Tim's tendency to blame external factors for software issues rather than adjusting his approach could hinder his BCI performance. However, Tim's high confidence in using technology may actually benefit his BCI performance. The conflict in these two aspects of Tim's self-efficacy, higher performance with higher confidence, but lower performance with low locus-of-control, may explain why gender gaps in BCI performance are not always clear.

If the main cognitive bias around computer self-efficacy has to do with locus-of-control, how do we then encourage users to have a higher, more internal, locus-of-control with BCI systems? One possible solution is to offer neuro-feedback that aligns with the machine learning model's expectations. For example, in a motor imagery BCI system that anticipates decreases in mu power to determine motor imagery classes, providing feedback on mu power relative to the system's expected threshold may help improve performance. This feedback would help users like Tim understand how their neural strategy influences the system, fostering a

sense of control rather than assuming that the system is malfunctioning. Additionally, this approach could enhance Abi's performance by boosting her confidence in the effectiveness of her neural strategy.

### 7.3.3 User Learning Style

Previous literature from [Jeunet et al., d] indicates that active vs reflective learning styles significantly influences BCI performance, such that active learners could more efficiently learn BCI control of a motor imagery task. Active learners prefer to "learn by doing something with information... and trying it out," whereas reflective learners prefer to "think things through and understand things before acting" [of Waterloo]. This concept of active learning resonates with Tim's tinkering learning style, where he learns software by experimenting with different functionalities. Based on prior findings, it seems that tinkering and active learning should enhance BCI performance, particularly benefiting individuals like Tim, who are more likely to be men. However, some experimental BCI setups may not facilitate active learning or tinkering due to constraints in the experimental design.

While some BCI experiments may support learning through tinkering [Jeunet et al., d], not all setups may incorporate this crucial component for BCI learning and performance. For example, in many P300 and SSVEP BCI experiments, participants are instructed to input specific phrases into the system [Allison et al.; Baykara et al.; Kleih et al.]. This highly structured experimental setup may restrict Tim's ability to learn the system, as there is limited opportunity for tinkering. Although some experiments allow users to input their own phrases of interest, this is typically permitted only at the end of the experiment [Allison et al.]. Therefore, experimental designs should allocate time for users to tinker and explore the system, ideally before engaging in the experimental tasks. This recommendation also aligns with Tim's motivation facet, where we suggested a similar solution. The yellow boxes in Figure 7.1 illustrate where Tim might want to tinker with the system.

### 7.3.4 User Information Processing Style

While previous research on human factors impacting BCI performance did not include information processing style, we believe it is worth considering in this context. Exploring all cognitive styles typically associated with men and their potential impact on BCI performance can provide valuable insights into po-

tential solutions and identify areas for further investigation. In Tim's case, his information processing style tends to be selective or "depth first," meaning he usually pursues the first promising option to complete a task until he determines its viability. In the case of BCI systems, this suggests that individuals like Tim may adopt the first promising neural strategy they encounter. However, this strategy might not be optimal in the long run. For example, in a motor imagery task, a user may initially focus on imagining movements broadly on the left and right sides of the body for a task involving left versus right hand movement. However, this approach may no longer work if additional motor imagery classes are introduced, such as foot movements. Tim's selective style may therefore hinder his ability to choose an optimal neural strategy, as he tends to continue with a strategy until he determines its ineffectiveness.

One approach to supporting selective information processing styles in BCI systems is to provide users with timely and relevant feedback on their performance and neural strategy. For example, in a motor imagery task, experimenters could offer feedback on how the machine learning model is classifying the users' neural signals. Incorporating neurofeedback as well can assist users like Tim in adjusting their specific neural strategies. More frequent feedback can help individuals like Tim receive appropriate error signals regarding their current strategy, allowing them to refine their approach. Without such feedback, Tim may assume that his initial strategy is effective in achieving the task. Additionally, adaptive neural decoders may prove beneficial in enhancing BCI performance for individuals with selective information processing styles, as these decoders can adapt to the initial promising strategy that Tim employs [Orsborn et al.]. However, it's essential to consider Abi's comprehensive information processing style as well, as Abi prefers to gather as much relevant information as possible before completing a task. To accommodate both learning styles, users could be allowed to determine when they receive feedback, rather than having it prescribed at specific times. For Abi, feedback may be most useful before beginning experimental tasks, while for Tim, it might be more beneficial between trials and tasks. Overall, the feedback should provide users with an indication of whether their neural strategy is effective and encourage them to refine or adapt their approach.

### 7.3.5   User Attitude Towards Risk

This final cognitive style covered by GenderMag, though not previously emphasized in BCI literature, is worth exploring for a comprehensive understanding. Tim exhibits a high risk tolerance, meaning that failures

| Cognitive Style | Evidence | BCI Performance Impact | Solution |
|---|---|---|---|
| Motivation | Herbert; Jeunet et al. [a] | Tim is motivated to explore BCI systems, but may not be able to explore sufficiently in experimental setups | Incorporate free-play time |
| Self-Efficacy | Wood and Kober | Tim's low locus-of-control with technology may cause him to blame the BCI system, rather than adjust his neural strategy | Incorporate neuro-feedback |
| Learning Style | Jeunet et al. [d] | Tim's tinkering by learning may be hindered by rigid experimental constraints | Incorporate free-play time |
| Information Processing Style | | Tim's selective information processing style may cause him to stick with a sub-optimal neural strategy | Incorporate neuro-feedback |
| Attitude Towards Risk | | Tim's high risk tolerance may make him more robust to BCI failures, but Abi's low risk tolerance will not | Include familiar features |

**Table 7.2:** Overview of how each cognitive style may impact BCI performance, and solutions to incorporate into BCI systems.

with technology do not deter his attitude towards technology. We hypothesize that this cognitive style is likely to support Tim's learning of BCI systems, as failures are unlikely to discourage him or create a negative perception of the system. In contrast, Abi has a low risk tolerance and is more inclined to rely on familiar features to complete tasks. Given that BCIs are still an emerging technology, much of the user experience with BCIs may be unfamiliar. While this cognitive style may not cause performance gaps in experimental setups, where participants are incentivized to engage with the system, thus lowering the associated risk, it may become more relevant as BCIs are commercialized. It is therefore essential to ensure that commercial BCI systems cater to users with low risk tolerance who prefer familiar features over new and unfamiliar ones. Transparency regarding the system's benefits and the average time required to become proficient may support the needs of low risk-tolerant users [Steine-Hanson et al.]. Additionally, incorporating familiar UX elements from existing software systems, such as using QWERTY keyboard layouts for text input, may benefit these users.

## 7.4 Discussion

This perspective leverages the GenderMag method and its personas to illuminate the gender gaps in BCI performance and offers insights into why these gaps were not always consistent in previous research. Since the cognitive styles represented by the GenderMag personas tend to cluster by gender, men with cognitive styles akin to Abi's may not necessarily exhibit poorer BCI performance. Therefore, we propose that *cognitive* biases, rather than gender biases, underlie these performance gaps.

While some of Tim's cognitive styles are not supported in current BCI systems, there are others that appear to benefit him in this context (see Table 7.2). Using the GenderMag method and existing literature on human factors in BCI performance, we find that four of Tim's cognitive styles may contribute to BCI performance challenges: motivation for using technology, low locus-of-control with technology, tinkering learning style, and selective information processing style. However, his high confidence and risk tolerance may actually bolster his ability to learn new BCI systems.

Our design recommendations to better support Tim's cognitive styles primarily focus on providing feedback to the user and allowing them to explore the system independently. Providing more feedback supports Tim's low locus-of-control (a component of computer self-efficacy) and his selective information processing style. Existing literature on user experience in BCIs also underscores the importance of feedback in enhancing BCI performance [Jeunet et al., a; Herbert; Lotte et al., a]. Additional research exploring *neuro*-feedback has also shown benefits to BCI performance in motor imagery [Zhou et al.], and SSVEP tasks [Wan et al.]. Based on Tim's cognitive styles, specifically his motivations and tinkering style, we also suggest incorporating exploratory time into BCI experiments. Previous work from Jeunet et al. [b] has also emphasized the importance of allowing users to explore different strategies, which ultimately led to better BCI learning.

In this evaluation, we primarily focused on the cognitive styles associated with the Tim persona, which align with men's typical traits. However, it's important to acknowledge that gender is not binary and includes non-binary, gender-neutral, and transgender identities, to name a few. Unfortunately, much of the existing literature on human factors in BCI performance uses sex as a proxy for gender, limiting the scope of their analyses to a binary framework. This approach hinders the exploration of potential performance gaps for genders beyond men and women. As we've highlighted, the observed performance gaps are more likely linked to cognitive biases rather than gender biases. Therefore, designing for a spectrum of cognitive

styles should encompass genders beyond the binary. This also underscores the need for further evaluations using the other GenderMag personas, such as Abi and the Pats, in the ongoing development of BCI systems. Additionally, other identities, such as socioeconomic status, age, and race, may also influence BCI performance [Hu et al.]. For BCI systems to be truly inclusive, it's imperative we consider all identities and cognitive styles in the design of these systems.

In the current BCI literature, men generally exhibit poorer BCI performance. However, in many software systems, women encounter more barriers and cognitive biases due to their cognitive styles [Burnett et al., c; Cunningham et al.; Mendez et al.; Shekhar and Marsden; Vorvoreanu et al.]. For example, research on gender diversity in Open Source Software (OSS) projects revealed that over 80% of the newcomer barriers identified are biased against cognitive styles typically associated with women [Mendez et al.]. Consequently, there is concern that as consumer BCIs become more prevalent, the existing gender gap in BCI performance may reverse, as current non-inclusive design practices are incorporated into BCI systems. For example, Tim's motivation and tinkering learning style can both be hindered by rigid experimental setups, leading to poorer performance [Volkova* and Gusev; Shirzadi et al.]. Previous studies outside of the context of BCIs have found that rigid experimental settings can induce boredom, which impacts performance and disproportionately affects men [Wang et al., a]. As BCI systems transition from lab settings to everyday use, these constraints are likely to disappear, and BCI systems may start to assume that users prefer to experiment with the system rather than follow tutorials. This shift would leave individuals with Abi's process-oriented learning style with sub-optimal ways to learn the BCI system. Therefore, it is crucial to incorporate inclusive design principles through the development of consumer BCI systems [Microsoft; Lotte et al., a].

The BCI studies we reviewed focused exclusively on "active" BCI systems, which demand users' active engagement and modulation of neural activity to use the system [Nijboer et al., b]. Examples of active systems include motor imagery tasks, P300 and other text input systems. In contrast, passive BCIs decode user states without requiring direct interaction. Passive systems can monitor attention, emotions, or fatigue levels, among other features. Given that passive systems operate without explicit user interaction, performance disparities in these systems are unlikely to stem from the cognitive biases discussed in this perspective. Instead, addressing performance gaps in passive BCI systems may necessitate improvements in machine learning models and signal processing. However, any user interfaces providing feedback for passive BCI

systems should adhere to inclusive design principles [Microsoft].

While the strategies outlined in this perspective can address issues stemming from cognitive biases in BCI software interfaces, challenges related to physiological differences and machine learning biases remain significant factors affecting BCI performance. Future efforts should aim to integrate solutions addressing these challenges, with a particular emphasis on mitigating model bias.

Future research should further investigate the role of cognitive biases in BCI systems. For one, empirical studies are needed to validate whether cognitive styles indeed contribute to BCI performance gaps. Researchers can utilize the GenderMag facets survey to assess participants' cognitive styles and examine if certain styles are correlated with higher BCI performance [Hamid et al.]. Additionally, future studies exploring gender differences in BCI performance should encompass a broader range of genders beyond the traditional categories of men and women to uncover additional potential biases. Lastly, comparative testing of various design solutions is essential to validate the effectiveness of inclusive design solutions.

## 7.5 Conclusion

In summary, this perspective offers a new lens on the gender gap in BCI performance and proposes strategies to mitigate this gap. We find evidence for low BCI performance in three of five cognitive styles typically associated with men which may contribute to their lower BCI performance (see Table 7.2). Our recommendations include allowing time for users to explore the system before engaging in experimental tasks, and integrating neuro-feedback and other kinds of feedback into the system. While these solutions can help address the current gender performance gap, it is important for practitioners not to rely solely on them to improve BCI systems, especially consumer systems. Ensuring developers, designers and other stakeholders adhere to inclusive design principles throughout the development of BCI systems is vital for creating the next generation of human-computer interaction. If we want BCI systems to work for everyone, we need to ensure that all identities are considered in the design of these systems.

# Chapter 8

# Discussion and Future Work

As portrayed in the introduction, the goal of this thesis is to is to advance the development of BCIs that work in people's daily lives. While BCI systems comprise many different components, we focused on improvements in three dimensions of BCIs – model generalizability, applicability to different contexts, and user experience. Improving model generalizability is essential to creating BCIs that can be used by anyone with minimal fine-tuning and minimal additional data collection. Building models that apply to many different contexts will also help to bring BCIs out of the lab and into various real-world use cases, such as prosthetics and mental health. Ensuring an inclusive user experience is also essential to ensure that BCIs will be accessible to everyone, not just people who's cognitive styles match with the developer. Together, improvements in these three dimensions can lead us towards BCIs that work for everyone, everyday.

In total, this thesis makes strides towards building BCI systems that work in real-world settings. We showed a deep learning model that could interpret neural data during naturalistic movement, generalize across participants, and could be adapted to different tasks, specifically movement classification and cognitive states. This model presents a potential way forward to develop BCIs that are robust in the real-world, especially since this model does not require excessive amounts of data for training. We then dived deeper into the naturalistic neural data to understand neural activity using neural manifolds. This led to insights about the alignment of neural activity between different naturalistic reaching movements, across days, and between individuals. We also covered additional approaches for improving BCI models, with self-supervised learning and neural manifold decoding both showing promising ways forward. Lastly, we explored how

improving the user experience using techniques from inclusive design can also improve BCI performance, especially for demographics that have previously shown poor performance. This work highlights the need for not just machine learning development to bring BCIs to the real-world, but also improvements that consider how people will interact with these systems.

Although the work in this thesis brings us closer to realizing practical, real-world BCIs, many aspects still require development before BCIs can achieve mainstream adoption. One significant area is the translation from offline to online systems. The research presented here focused solely on offline decoding, where data is analyzed post-collection. While this approach helps in understanding and refining neural decoders under more constraints, it does not address the challenges of real-time signal processing and decision-making that online systems require. Online BCIs must perform reliably with minimal latency under varying environmental conditions, presenting a complex set of engineering and computational challenges that are crucial for everyday applicability. Additionally, offline neural decoders often fail to translate to online [Gilja et al., a], and neuro-adaptation from the user can further complicate the neural decoding problem [Orsborn et al.].

Another significant area for development is in the dichotomy between invasive and non-invasive BCIs. Invasive methods, such as those using ECoG or microelectrodes, offer high-resolution signals and are less susceptible to noise, making them powerful tools for precise decoding. However, their broad applicability is limited due to the risks associated with surgical implantation. Non-invasive methods, while safer and more practical for consumer use, often suffer from lower signal clarity and higher susceptibility to artifacts. Balancing the benefits of invasive techniques with the safety of non-invasive approaches remains a critical problem in BCI research. To move towards mainstream BCI adoption, future research must focus on developing robust online systems that can operate effectively in naturalistic environments, and achieve high data quality with non-invasive or minimally invasive approaches.

In this thesis, we emphasize the benefits and use cases of generalizable, "out-of-the-box" models. However, there is a trade-off between generalized models and tailored models. While generalized models can quickly get to functional use for many users, tailored models trained with enough data can provide better performance for users. This trade-off is especially exemplified in Figure 3.3, where the tailored HTNet models ultimately show higher performance once they are trained with enough data. Figure 5.5 also highlights that there is only a certain amount of neural overlap between users, indicating that generalized models may
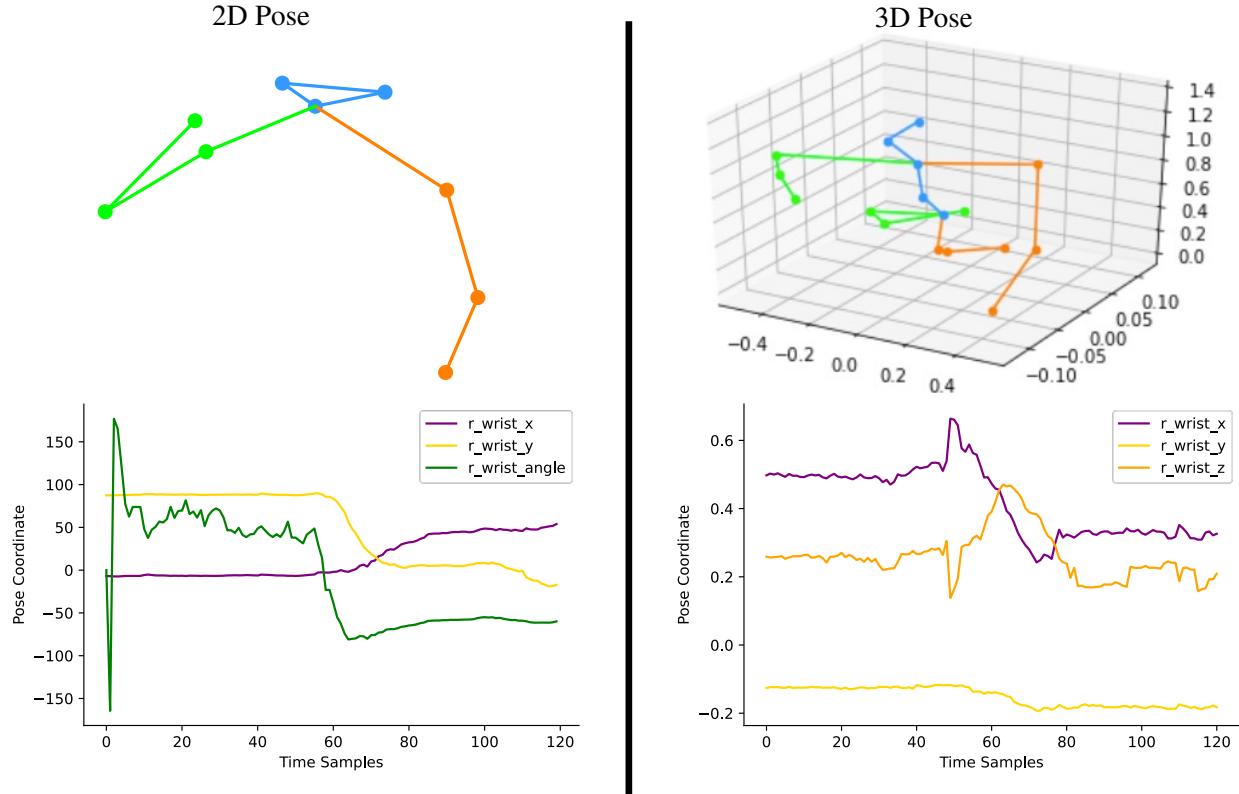
only be able to go so far to support all users. While it is important to ensure that out-of-the-box models are useful for the majority of users, tailored models may still be necessary for the development of everyday BCIs, especially for users who may be further out-of-distribution to the training data. The need for some level of tailored models is further exemplified in Chapter 7, as it is clear that users with different cognitive styles will want different approaches to use their BCIs. In this case, tailored models may be better able to serve users who prefer different neural strategies than what training users often used. We imagine that future broad implementations of BCI systems may need to initially use out-of-the-box models for all users, but will collect data to analyze how out-of-distribution a given user is and train tailored models for users who would benefit from them most.

In the following sections, we propose future research projects that build upon the findings of this thesis and aim to further enhance these advancements.

## 8.1 Continuous Decoding of 2D and 3D Pose

In Chapter 6, our efforts to decode complex movement features from naturalistic ECoG data were met with limited success. This shortfall could stem from several factors, including the choice of model architectures or inherent noise within the data. A likely cause, however, appears to be the absence of continuous temporal predictions in the deep learning models we tested. Unlike static computational models, the brain dynamically updates its internal representations and predictions throughout the execution of a task, adapting to new information as it becomes available [Rao and Ballard; Rao et al.]. Consequently, ECoG signals are likely to contain dynamic information reflecting the evolving state of a task. Our previous decoding strategies primarily aimed at predicting discrete, singular event features – such as the reach angle at the extent of the reach – rather than modeling the trajectory of movement features continuously through time. To enhance the complexity and accuracy of naturalistic ECoG decoders, future research should focus on developing models capable of making continuous predictions across the duration of movements.

To support this work, we developed a dataset containing continuous 2D and 3D pose labels corresponding to the movement events previously identified within the naturalistic ECoG dataset (see Figure 8.1). For the 2D pose labels, we utilized pose data extracted using DeepLabCut, as detailed in Singh et al. [b]. This method estimated poses based on 9 key joint locations: the nose, both ears, shoulders, elbows, and wrists.

157

**Figure 8.1:** Example 2D and 3D poses for future continuous decoding work. Left side shows the estimated 2D poses. Right side shows the estimated 3D poses. Top figures show an example of all estimated joints in the poses for a single frame. 2D pose contains x and y coordinates for 9 joints, while 3D pose contains x, y and z coordinates for 17 joints. Bottom plots show continuous joint coordinates for the right wrist. In the case of 2D pose, we show x and y coordinates, as well as the calculated reach angle available in the dataset. In 3D pose, we show x, y and z coordinates.

```
<xarray.Dataset>
Dimensions:          (ecog_channels: 98, ecog_time_samps: 16, events: 1318, feature_names: 15, pose_channels: 18, pose_time_samps: 1, trial_samples: 120)
Coordinates:
  * events          (events) float64 3.0 3.0 3.0 3.0 3.0 ... 7.0 7.0 7.0 7.0
  * trial_samples   (trial_samples) int64 0 1 2 3 4 5 ... 115 116 117 118 119
  * ecog_channels   (ecog_channels) object 'GRID1' 'GRID2' ... 'ECGL' 'ECGR'
  * pose_channels   (pose_channels) object 'L_Ear_x' ... 'R_Wrist_y'
  * ecog_time_samps (ecog_time_samps) int64 0 1 2 3 4 5 6 ... 10 11 12 13 14 15
  * pose_time_samps (pose_time_samps) float64 0.0
  * feature_names   (feature_names) object 'day' ... 'shoulder_slope_mean'
Data variables:
    ecog_data       (events, trial_samples, ecog_channels, ecog_time_samps) float64 ...
    pose_data       (events, trial_samples, pose_channels, pose_time_samps) float64 ...
    pose_angle      (events, trial_samples, pose_time_samps) float64 ...
    pose_quad       (events, trial_samples, pose_time_samps) float64 ...
    pose_wrist_disp (events, trial_samples, pose_time_samps) float64 ...
    features        (events, feature_names) object ...
```

**Figure 8.2:** Data structure for the 2D pose data in xarray format. Data presented here from EC01. Data contains all ECoG channels, all pose channels, calculated pose features and additional metadata about the movement events.

The 2D labels were sampled at a rate of 30Hz, consistent with the video data frequency. To synchronize these labels with the ECoG data, we aligned them with the pre-processed ECoG signals, which were down-sampled to 500Hz from the original 1000Hz recording. This meant that each pose label correlated with 16 ECoG data samples, essential for continuous decoding. Due to the mismatched sampling rates between the pose and ECoG data, we omitted one ECoG sample between each label to maintain alignment. In addition to the x and y pose coordinates available from Singh et al. [b], our dataset also includes continuous labels for additional features, such as reach angle and wrist displacement. The dataset is structured as shown in Figure 8.2 and is available upon request from the author.

While 2D pose information provides valuable insights, it captures only a limited aspect of movement and may omit details crucial to understanding how the brain encodes movement. Fortunately, recent advancements in 3D pose estimation using a single camera [Liu et al., b], have enabled us to extract 3D poses from our naturalistic video data effectively. Therefore, we have also compiled a dataset of continuous 3D poses (illustrated in Figure 8.1). To generate 3D poses, we utilized publicly available models from MMCV, a widely recognized open-source library for computer vision that offers a straightforward interface for running model inference on video data [MMCV]. Specifically, we employed their default human3D pose model, MotionBERT [Zhu et al.], to perform inference. The output pose data is stored in the Human3.6M format [Ionescu et al.], providing pose information for each individual detected in the video. Alignment of the 3D pose data with the ECoG recordings follows the same procedure as with the 2D data, ensuring consistency in temporal correlation. Additionally, considering the presence of multiple individuals in some video
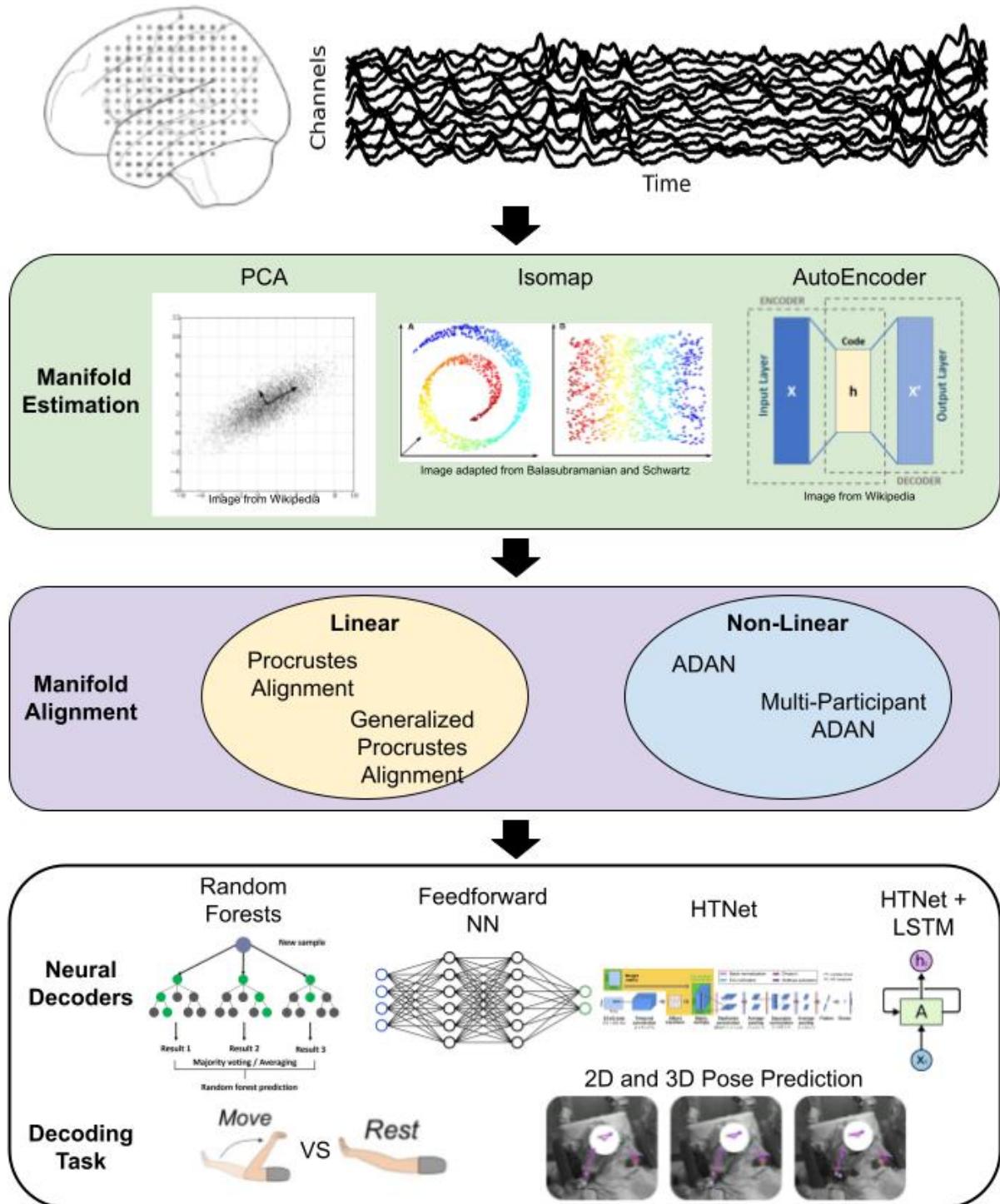
segments, we implemented a processing pipeline to maintain consistent tracking of the same individual throughout each video. To optimize computational resources, we limited our inference runs to two-second video clips centered on movement events previously identified in Singh et al. [b], rather than processing the entire continuous video dataset. This targeted approach allows us to focus on the most relevant data while conserving computational efforts.

We have successfully developed datasets for continuous decoding; however, future work still needs to design suitable neural decoders. We suggest enhancing the HTNet architecture by incorporating LSTM layers to achieve this. The CNN layers of HTNet could serve as feature extractors on the ECoG data, with the extracted features subsequently processed by LSTM layers to capture dynamic temporal information and historical context from the ECoG signals. Additionally, we recommend developing models that only contain LSTM layers to help assess the intrinsic value of the feature extraction process in our neural decoders. Future work may also want to explore transformers for continuous decoding, as transformers have demonstrated substantial gains in predicting sequential data [Islam et al.]. Developing neural decoders that effectively handle continuous pose information in naturalistic movements represents a significant advancement in BCIs, moving us closer to creating devices that are practical and effective in everyday environments.

## 8.2   Decoding Cross-Participant Generalization with Neural Manifold Alignment

As shown in Chapter 5, naturalistic neural data contains low-dimensional manifolds during movement control. This chapter also established that the linear neural manifolds extracted with PCA showed more alignment than chance between participants. Future work can leverage this low-dimensional manifold information to build neural decoders that use neural manifold alignment techniques to achieve cross-participant generalizability. Chapter 6 showed some work on neural manifold alignment with CCA, but this did not result in useful decoding performance. Future work should therefore explore additional manifold alignment methods, including non-linear techniques, and apply these to a variety of naturalistic movement decoding tasks. Future work should also measure how the manifold estimates change with different alignment techniques and test how much data is needed to achieve reasonable performance with manifold alignment. An

overview of the suggested future work is shown in Figure 8.3.



**Figure 8.3:** Overview of suggested work for manifold decoding of naturalistic ECoG.
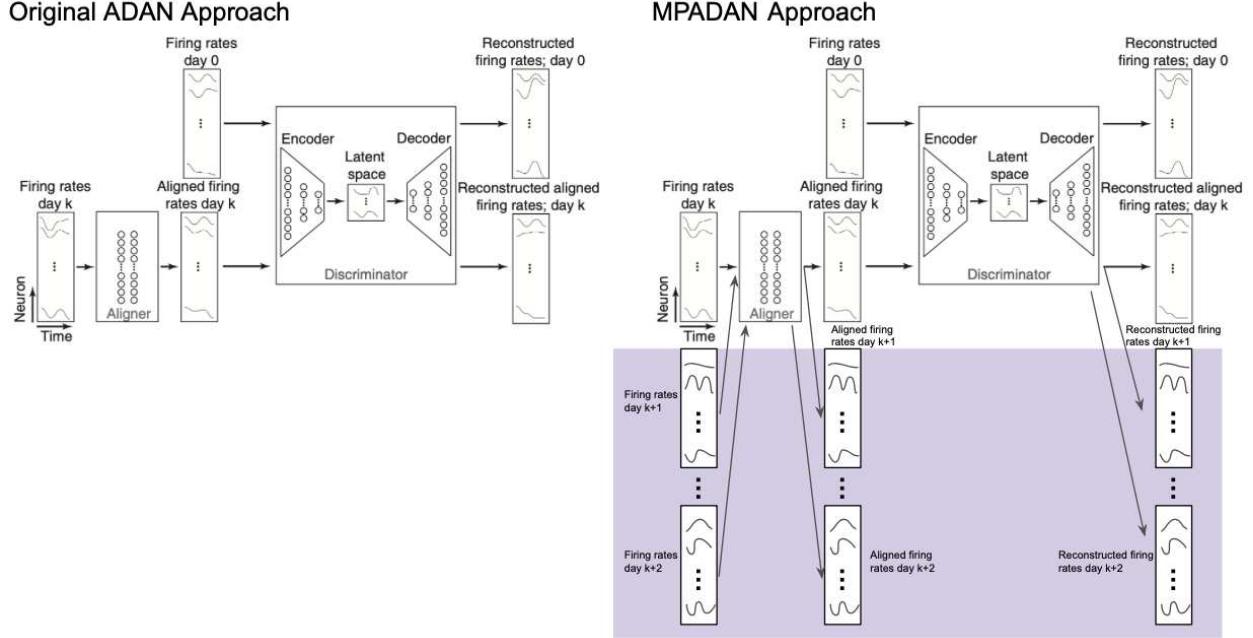
### 8.2.1 Manifold Estimation Methods

In Chapter 5, we calculated estimates of the neural manifold using PCA. However, many other techniques exist to estimate neural manifolds. We suggest that future work explore techniques such as Isomap and autoencoders (AEs) to approximate neural manifolds in the naturalistic ECoG data. Additional manifold estimation techniques may reveal new insights on the naturalistic data, especially for non-linear approaches like AEs. We suggest Isomap because it showed slightly better naturalistic decoding performance in Melbaum et al., and because Isomap preserves global structure better than LEM. AEs fit nicely into the ADAN technique suggested in section 8.2.2, and they can be jointly trained with a decoder to optimize the extracted manifold for decoding. However, AEs often have problems reaching convergence, especially with noisy data. To handle this issue, we suggest developing denoising algorithms on the naturalistic ECoG data before passing through the AE. All combinations of these manifold estimation methods and manifold alignment techniques (section 8.2.2) should be tested to see which approaches provide the most useful decoding performance.

### 8.2.2 Manifold Alignment Techniques

Future work should aim to develop manifold alignment techniques for cross-participant decoding based on the techniques seen in Chapter 2. Two linear methods and two non-linear methods are string candidates. Procrustes Alignment and Generalized Procrustes Alignment (GPA) are suggested for linear methods. We would suggest CCA, though the results from Chapter 6 and the requirement that the data is time locked indicate that CCA will not work well on the naturalistic ECoG dataset. Following the work from Melbaum et al., Procrustes Alignment seems promising, as they successfully used it for cross-participant alignment and it is also a simple approach. However, future work could improve upon their approach by also trying GPA, which instead of mapping onto an arbitrarily selected shape from the data, the data is transformed onto an optimal universal shape for all data points. A generalized approach like this has the benefit if being able to extend to more participants, rather than a paired approach like Procrustes Alignment and CCA. For the non-linear methods, we suggest ADAN and a novel updated variation of ADAN that can align multiple participants at once, which we call Multi-Participant ADAN or MPADAN. ADAN is a good approach because it does not rely on the time locked constraint of CCA, and also can produce non-linear

mappings between participants. We suggest extending this approach by training the ADAN architecture to align and discriminate between multiple participants at once, thus also learning a generalized manifold alignment transformation like in GPA. An overview of the ADAN and MPADAN is shown in Figure 8.4.



**Figure 8.4:** The original ADAN approach from Farshchian et al. and the novel MPADAN approach we suggest for future work. Figure adapted from Farshchian et al..

### 8.2.3 Test Manifold Alignment Decoders on Naturalistic ECoG

After building up implementations of the five manifold alignment techniques, we suggest testing how useful each approach is for neural decoding. These techniques should first be tested in a simple move vs rest classification, as seen in Chapter 3. For this classification task, we suggest using the same HTNet architecture as in Chapter 3 and Peterson et al. [b], which will take in the aligned manifold data as input. We also suggest comparing to simpler models, such as random forests and small feed-forward neural networks, to see how much the CNN model improves performance. After testing the neural decoders on this simpler task, we suggest trying more complex tasks, such predicting 2D and 3D pose information during the course of the naturalistic movements as suggested in section 8.1.

### 8.2.4 Compare Neural Latent Dynamics Between Alignment Techniques

After extracting the neural manifolds and training the neural decoders, we suggest performing interpretability analyses on the learned manifold spaces. First, we suggest exploring how jointly training the manifold estimate along with the neural decoder influences the learned dynamics. This can be explored through the ADAN approach by training an unsupervised version, which only focuses on the alignment step at training time, and a supervised version, which jointly trains the ADAN alignment and the AE estimating the manifold with the classification or regression task. Exploring these differences in the latent dynamics will likely reveal important task-relevant features. Second, we suggest exploring how the manifolds learned in the generalized alignment approaches (GPA, MPADAN) differ from their paired counterparts. Exploring the differences in the latent dynamics of the generalized models will likely give insight to any universal brain dynamics between participants.

### 8.2.5 Test Minimum Amount of Data Required for Alignment

Lastly, we suggest testing how little data these techniques can use by slowly reducing the amount of data used in the manifold alignment step. Since one motivation of this work is the generalization of these models in real-world contexts, seeking out the minimum amount of data needed for alignment is crucial. Across all five manifold alignment approaches, we suggest trying varying amounts of data from the new participants, starting with as much as 1200 events, and working down to as few as 10.

# Bibliography

2007. *Canonical Correlation Analysis*, pages 321–330. Springer Berlin Heidelberg, Berlin, Heidelberg.

Hamidreza Abbaspourazad, Mahdi Choudhury, Mahdi Choudhury, Yan T. Wong, Bijan Pesaran, and Maryam M. Shanechi. Multiscale low-dimensional motor cortical state dynamics predict naturalistic reach-and-grasp behavior. 12(1):607–607. MAG ID: 3123056764.

Asrul Adam, Mohd Ibrahim Shapiai, Mohd Zaidi Mohd Tumari, Mohd Saberi Mohamad, and Marizan Mubin. 2014. Feature selection and classifier parameters estimation for eeg signals peak detection using particle swarm optimization. *The Scientific World Journal*, 2014.

Nur Ahmadi, Timothy G. Constandinou, and Christos-Savvas Bouganis. Decoding hand kinematics from local field potentials using long short-term memory (LSTM) network. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 415–419. IEEE.

Minkyu Ahn and Sung Chan Jun. Performance variation in motor imagery brain–computer interface: A brief review. 243:103–110.

Misha B. Ahrens, Jennifer M. Li, Michael B. Orger, Drew N. Robson, Alexander F. Schier, Florian Engert, and Ruben Portugues. Brain-wide neuronal dynamics during motor adaptation in zebrafish. 485(7399):471–477.

Hesam Akbari, Muhammad Tariq Sadiq, Siuly Siuly, Yan Li, and Paul Wen. Identification of normal and depression EEG signals in variational mode decomposition domain. 10(1):24.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A

next-generation hyperparameter optimization framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Amjed S. Al-Fahoum and Ausilah A. Al-Fraihat. Methods of EEG signal features extraction using linear analysis in frequency and time-frequency domains. 2014:1–7.

Abdulwahab Alasfour, Paolo Gabriel, Xi Jiang, Isaac Shamie, Lucia Melloni, Thomas Thesen, Patricia Dugan, Daniel Friedman, Werner Doyle, Orin Devinsky, David Gonda, Shifteh Sattar, Sonya Wang, Eric Halgren, and Vikash Gilja. Coarse behavioral context decoding. 16(1):016021.

Maryam Alimardani and Diana-Eliza Gherman. Individual differences in motor imagery BCIs: a study of gender, mental states and mu suppression. In *2022 10th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–7. IEEE.

Brendan Allison, Thorsten Luth, Diana Valbuena, Amir Teymourian, Ivan Volosyak, and Axel Graser. BCI demographics: How many (and what kinds of) people can use an SSVEP BCI? 18(2):107–116.

Brendan Z. Allison and Christa Neuper. Could anyone use a BCI? In Desney S. Tan and Anton Nijholt, editors, *Brain-Computer Interfaces*, pages 35–54. Springer London. Series Title: Human-Computer Interaction Series.

Ege Altan, Xuan Ma, Lee E. Miller, Eric J. Perreault, and Sara A. Solla. Low-dimensional neural manifolds for the control of constrained and unconstrained movements.

David J. Anderson and Pietro Perona. Toward a science of computational ethology. 84(1):18–31.

Michael L. Anderson, Josh Kinnison, and Luiz Pessoa. a. Describing functional diversity of brain regions and brain networks. 73:50–58.

Nicholas R. Anderson, Tim Blakely, Gerwin Schalk, Eric C. Leuthardt, and Daniel W. Moran. b. Electro-corticographic (ECoG) correlates of human arm movements. 223(1):1–10.

Leopoldo Angrisani, Pasquale Arpaia, Antonio Esposito, Ludovica Gargiulo, Angela Natalizio, Giovanna Mastrati, Nicola Moccaldi, and Marco Parvis. Passive and active brain-computer interfaces for rehabilitation in health 4.0. 18:100246.

Gopala K. Anumanchipalli, Josh Chartier, and Edward F. Chang. 2019. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753):493–498.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture.

Ruslan Aydarkhanov Aydarkhanov, Marija Uscumlic, Ricardo Chavarriaga, Lucian Gheorghe, and Jose del R Millan. 2020. Spatial covariance improves bci performance for late erps components with high temporal variability. *Journal of Neural Engineering*.

Ahmed M. Azab, Lyudmila Mihaylova, Kai Keng Ang, and Mahnaz Arvaneh. Weighted transfer learning for improving motor imagery-based brain–computer interface. 27(7):1352–1359.

Mehdi Azabou, Mohammad Gheshlaghi Azar, Ran Liu, Chi-Heng Lin, Erik C. Johnson, Kiran Bhaskaran-Nair, Max Dabagia, Keith B. Hengen, William Gray-Roncal, Michal Valko, and Eva L. Dyer. Mine your own vieW: Self-supervised learning through across-sample prediction.

Hubert Banville, Omar Chehab, Aapo Hyvärinen, Denis-Alexander Engemann, and Alexandre Gramfort. Uncovering the structure of clinical EEG signals with self-supervised learning.

Zeke Barger, Charles G. Frye, Danqian Liu, Yang Dan, and Kristofer E. Bouchard. Robust, automated sleep scoring by a compact neural network with distributional shift correction. 14(12):e0224642.

O. Barkan and N. Koenigstein. 2016. ITEM2VEC: Neural item embedding for collaborative filtering. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6.

Erol Başar, Canan Başar-Eroglu, Sirel Karakaş, and Martin Schürmann. 2001. Gamma, alpha, delta, and theta oscillations govern cognitive processes. *International journal of psychophysiology*, 39(2-3):241–248.

L Bashford, I Rosenthal, S Kellis, D Bjånes, K Pejsa, Bw Brunton, and Ra Andersen. Neural subspaces of imagined movements in parietal cortex remain stable over several years in humans.

Pouya Bashivan, Irina Rish, Mohammed Yeasin, and Noel Codella. 2015. Learning representations from eeg with deep recurrent-convolutional neural networks. *arXiv preprint arXiv:1511.06448*.

E. Baykara, C.A. Ruf, C. Fioravanti, I. Käthner, N. Simon, S.C. Kleih, A. Kübler, and S. Halder. Effects of training and motivation on auditory p300 brain–computer interface performance. 127(1):379–387.

L. Beckwith and M. Burnett. Gender: An important factor in end-user programming environments? In *2004 IEEE Symposium on Visual Languages - Human Centric Computing*, pages 107–114. IEEE.

L. Beckwith, S. Sorte, M. Burnett, S. Wiedenbeck, T. Chintakovid, and C. Cook. a. Designing features for both genders in end-user programming environments. In *2005 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05)*, pages 153–160. IEEE.

Laura Beckwith, Margaret Burnett, Susan Wiedenbeck, Curtis Cook, Shraddha Sorte, and Michelle Hastings. b. Effectiveness of end-user debugging software features: are there gender issues? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 869–878. ACM.

Laura Beckwith, Cory Kissinger, Margaret Burnett, Susan Wiedenbeck, Joseph Lawrance, Alan Blackwell, and Curtis Cook. c. Tinkering and gender in end-user programmers' debugging. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 231–240. ACM.

Joos Behncke, Robin Tibor Schirrmeister, Martin Volker, Jiri Hammer, Petr Marusic, Andreas Schulze-Bonhage, Wolfram Burgard, and Tonio Ball. 2018. Cross-paradigm pretraining of convolutional networks improves intracranial eeg decoding. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1046–1053. IEEE.

Alim Louis Benabid, Thomas Costecalde, Andrey Eliseyev, Guillaume Charvet, Alexandre Verney, Serpil Karakas, Michael Foerster, Aurélien Lambert, Boris Morinière, Neil Abroug, Marie-Caroline Schaeffer, Alexandre Moly, Fabien Sauter-Starace, David Ratel, Cecile Moro, Napoleon Torres-Martinez, Lilia Langar, Manuela Oddoux, Mircea Polosan, Stephane Pezzani, Vincent Auboiroux, Tetiana Aksenova, Corinne Mestais, and Stephan Chabardes. An exoskeleton controlled by an epidural wireless brain–machine interface in a tetraplegic patient: a proof-of-concept demonstration. 18(12):1112–1122.

Yoav Benjamini and Yosef Hochberg. 1995. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300.

James Bergstra, Daniel Yamins, and David Cox. 2013. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pages 115–123.

James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554.

Nima Bigdely-Shamlo, Tim Mullen, Christian Kothe, Kyung-Min Su, and Kay A. Robbins. The PREP pipeline: standardized preprocessing for large-scale EEG analysis. 9.

Nima Bigdely-Shamlo, Tim Mullen, Kenneth Kreutz-Delgado, and Scott Makeig. 2013. Measure projection analysis: a probabilistic approach to eeg source comparison and multi-subject inference. *NeuroImage*, 72:287–303.

Ake Bjorck and Gene H. Golub. 1973. Numerical methods for computing angles between linear subspaces. 27(123):579.

M.J. Black, E. Bienenstock, J.P. Donoghue, M. Serruya, Wei Wu, and Yun Gao. Connecting brains with machines: the neural control of 2d cursor movement. In *First International IEEE EMBS Conference on Neural Engineering, 2003. Conference Proceedings.*, pages 580–583. IEEE.

BrainCo. BrainCo - FocusCalm.

Brianna M Karpowicz, Yahia H Ali, Lahiru N. Wimalasena, Andrew R. Sedler, Mohammad Reza Keshtkaran, Kevin Bodkin, Xuan Ma, Lee E. Miller, and Chethan Pandarinath. 2022. Stabilizing brain-computer interfaces through alignment of latent dynamics. MAG ID: 4224214225 S2ID: bb01681116b47c164d301c02728225b326c97520.

Thomas Brochier and Maria Alessandra Umiltà. 2007. Cortical control of grasp in non-human primates. 17(6):637–643.

Clemens Brunner, Niels Birbaumer, Benjamin Blankertz, Christoph Guger, Andrea Kübler, Donatella Mattia, José del R Millán, Felip Miralles, Anton Nijholt, Eloy Opisso, et al. 2015. Bnci horizon 2020: towards a roadmap for the bci community. *Brain-computer interfaces*, 2(1):1–10.

Angela M. Bruno, William N. Frost, and Mark D. Humphries. Modular deconstruction reveals the dynamical and physical building blocks of a locomotion motor program. 86(1):304–318.

M Burnett, S Stumpf, L Beckwith, and A Peters. a. The GenderMag kit: How to use the GenderMag method to find inclusiveness issues through a gender lens.

Margaret Burnett, Scott D. Fleming, Shamsi Iqbal, Gina Venolia, Vidya Rajaram, Umer Farooq, Valentina Grigoreanu, and Mary Czerwinski. b. Gender differences and programming environments: across programming populations. In *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, pages 1–10. ACM.

Margaret Burnett, Anicia Peters, Charles Hill, and Noha Elarief. c. Finding gender-inclusiveness software issues with GenderMag: A field investigation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 2586–2598. ACM.

Margaret Burnett, Simone Stumpf, Jamie Macbeth, Stephann Makri, Laura Beckwith, Irwin Kwan, Anicia Peters, and William Jernigan. d. GenderMag: A method for evaluating software's gender inclusiveness. 28(6):760–787.

Margaret M. Burnett, Laura Beckwith, Susan Wiedenbeck, Scott D. Fleming, Jill Cao, Thomas H. Park, Valentina Grigoreanu, and Kyle Rector. e. Gender pluralism in problem-solving software. 23(5):450–460.

P Cafferata and A Tybout. *Gender Differences in Information Processing: A Selectivity Interpretation, Cognitive and Affective Responses to Advertising*. Lexington Books.

Jing Cai, Alex E. Hadjinicolaou, Angelique C. Paulk, Ziv M. Williams, and Sydney S. Cash. Natural language processing models reveal neural dynamics of human conversation.

Adam J. Calhoun, Jonathan W. Pillow, and Mala Murthy. Unsupervised identification of the internal states that shape natural behavior. 22(12):2040–2049.

Ryan T Canolty and Robert T Knight. 2010. The functional role of cross-frequency coupling. *Trends in cognitive sciences*, 14(11):506–515.

Jessica Cantillo-Negrete, Josefina Gutierrez-Martinez, Ruben I Carino-Escobar, Paul Carrillo-Mora, and David Elias-Vinas. a. An approach to improve the performance of subject-independent BCIs-based on motor imagery allocating subjects by gender. 13(1):158.

Jessica Cantillo-Negrete, Josefina Gutierrez-Martinez, Ruben I. Carino-Escobar, Teodoro B. Flores-Rodriguez, and David Elias-Vinas. b. Time-frequency analysis of EEG signals from healthy subjects allocated by gender for a subject-independent BCI-based on motor imagery. In *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 37–40. IEEE.

Jill Cao, Irwin Kwan, Faezeh Bahmani, Margaret Burnett, Scott D. Fleming, Josh Jordahl, Amber Horvath, and Sherry Yang. a. End-user programmers in trouble: Can the idea garden help them to help themselves? In *2013 IEEE Symposium on Visual Languages and Human Centric Computing*, pages 151–158. IEEE.

Jill Cao, Kyle Rector, Thomas H. Park, Scott D. Fleming, Margaret Burnett, and Susan Wiedenbeck. b. A debugging perspective on end-user mashup programming. In *2010 IEEE Symposium on Visual Languages and Human-Centric Computing*, pages 149–156. IEEE.

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. c. OpenPose: Realtime multi-person 2d pose estimation using part affinity fields.

J Cassell. Genderizing human-computer interaction. In *The Human- computer Interaction Handbook*, pages 401–412.

Vincenzo Catrambone, Alberto Greco, Giuseppe Averta, Matteo Bianchi, Gaetano Valenza, and Enzo Pasquale Scilingo. Predicting object-mediated gestures from brain activity: An EEG study on gender differences. 27(3):411–418.

Ana-Maria Cazan, Elena Cocoradă, and Cătălin Ioan Maican. Computer anxiety and attitudes towards the computer and the internet with romanian high-school and university students. 55:258–267.

Chi-Yuan Chang, Sheng-Hsiou Hsu, Luca Pion-Tonachini, and Tzyy-Ping Jung. a. Evaluation of artifact subspace reconstruction for automatic artifact components removal in multi-channel EEG recordings. 67(4):1114–1121.

Shuo Chang, Vikas Kumar, Eric Gilbert, and Loren G. Terveen. b. Specialization, homophily, and gender in a social curation site: findings from pinterest. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 674–686. ACM.

Gary Charness and Uri Gneezy. Strong evidence for gender differences in risk taking. 83(1):50–58.

Ricardo Chavarriaga, Melanie Fried-Oken, Sonja Kleih, Fabien Lotte, and Reinhold Scherer. Heading for new shores! overcoming pitfalls in BCI design. 4(1):60–73.

Cèline Chayer and Morris Freedman. Frontal lobe functions. 1(6):547–552.

Qi Chen, Ralph Weidner, Simone Vossel, Peter H. Weiss, and Gereon R. Fink. 2012. Neural mechanisms of attentional reorienting in three-dimensional space. 32(39):13352–13362.

Tao Chen, Haiyun Huang, Jiahui Pan, and Yuanqing Li. An EEG-based brain-computer interface for automatic sleep stage classification. In *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pages 1988–1991. IEEE.

AKI Chiang, CJ Rennie, PA Robinson, JA Roberts, MK Rigozzi, RW Whitehouse, RJ Hamilton, and E Gordon. 2008. Automated characterization of multiple alpha peaks in multi-site electroencephalograms. *Journal of Neuroscience Methods*, 168(2):396–411.

Jae W Chung, Edward Ofori, Gaurav Misra, Christopher W Hess, and David E Vaillancourt. 2017. Beta-band activity and connectivity in sensorimotor and parietal cortex are important for accurate motor performance. *NeuroImage*, 144:164–173.

SueYeon Chung and L.F. Abbott. Neural population geometry: An approach for understanding biological and artificial neural networks. 70:137–144.

Mark M. Churchland, John P. Cunningham, Matthew T. Kaufman, Justin D. Foster, Paul Nuyujukian, Stephen I. Ryu, and Krishna V. Shenoy. a. Neural population dynamics during reaching. 487(7405):51–56.

Mark M. Churchland, John P. Cunningham, Matthew T. Kaufman, Stephen I. Ryu, and Krishna V. Shenoy. b. Cortical preparatory activity: Representation of movement or first cog in a dynamical machine? 68(3):387–400.

Mark M. Churchland and Krishna V. Shenoy. Temporal complexity and heterogeneity of single-neuron activity in premotor and motor cortex. 97(6):4235–4257.

Mark M Churchland, Byron M Yu, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, David C Bradley, Matthew A Smith, Adam Kohn, J Anthony Movshon, Katherine M Armstrong, Tirin Moore, Steve W Chang, Lawrence H Snyder, Stephen G Lisberger, Nicholas J Priebe, Ian M Finn, David Ferster, Stephen I Ryu, Gopal Santhanam, Maneesh Sahani, and Krishna V Shenoy. c. Stimulus onset quenches neural variability: a widespread cortical phenomenon. 13(3):369–378.

Philip Coen, Jan Clemens, Andrew J. Weinstein, Diego A. Pacheco, Yi Deng, and Mala Murthy. Dynamic sensory cues shape song structure in drosophila. 507(7491):233–237.

Michael X Cohen. 2014a. Fluctuations in oscillation frequency control spike timing and coordinate neural networks. *The Journal of Neuroscience*, 34(27):8988.

Michael X Cohen. 2020. A data-driven method to identify frequency boundaries in multichannel electrophysiology data. *bioRxiv*.

Mike X Cohen. 2014b. Analyzing Neural Time Series Data: Theory and Practice.

Scott Cole and Bradley Voytek. 2019. Cycle-by-cycle analysis of neural oscillations. *Journal of neurophysiology*, 122(2):849–861.

Scott R Cole and Bradley Voytek. 2017. Brain oscillations and the importance of waveform shape. *Trends in cognitive sciences*, 21(2):137–149.

Jennifer L Collinger, Robert A Gaunt, and Andrew B Schwartz. 2018. Progress towards restoring upper limb movement and sensation through intracortical brain-computer interfaces. *Current Opinion in Biomedical Engineering*, 8:84–92.

Committee on Measuring Sex, Gender Identity, and Sexual Orientation, Committee on National Statistics, Division of Behavioral and Social Sciences and Education, and National Academies of Sciences, Engineering, and Medicine. *Measuring Sex, Gender Identity, and Sexual Orientation*. National Academies Press. Pages: 26424.

Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. 2017. Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, 4(3):155–174.

Andrew W Corcoran, Phillip M Alday, Matthias Schlesewsky, and Ina Bornkessel-Schlesewsky. 2018. Toward a reliable, automated method of individual alpha frequency (iaf) quantification. *Psychophysiology*, 55(7):e13064.

Constantinos Coursaris, Sarah Swierenga, and Ethan Watrall. An empirical investigation of color temperature and gender effects on web aesthetics. 3(3):103–117.

Benjamin R. Cowley, Matthew A. Smith, Adam Kohn, and Byron M. Yu. Stimulus-driven population activity patterns in macaque primary visual cortex. 12(12):e1005185.

CTRL-labs at Reality Labs, David Sussillo, Patrick Kaifosh, and Thomas Reardon. A generic noninvasive neuromotor interface for human-computer interaction.

Yuqi Cui, Yifan Xu, and Dongrui Wu. 2019. Eeg-based driver drowsiness estimation using feature weighted episodic training. *IEEE transactions on neural systems and rehabilitation engineering*, 27(11):2263–2273.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. 17(11):1500–1509.

Sally Jo Cunningham, Annika Hinze, and David M. Nichols. Supporting gender-neutral digital library creation: A case study using the GenderMag toolkit. In Atsuyuki Morishima, Andreas Rauber, and

Chern Li Liew, editors, *Digital Libraries: Knowledge, Information, and Data in an Open Access Society*, volume 10075, pages 45–50. Springer International Publishing. Series Title: Lecture Notes in Computer Science.

Thirza Dado, Yağmur Güçlütürk, Luca Ambrogioni, Gabriëlle Ras, Sander Bosch, Marcel Van Gerven, and Umut Güçlü. Hyperrealistic neural decoding for reconstructing faces from fMRI activations via the GAN latent space. 12(1):141.

Mohammad Dastjerdi, Muge Ozker, Brett L. Foster, Vinitha Rangarajan, and Josef Parvizi. Numerical processing in the human parietal cortex during experimental and natural conditions. 4(1):2528.

Marco Davare, Alexander Kraskov, John C Rothwell, and Roger N Lemon. 2011. Interactions between areas of the cortical grasping network. 21(4):565–570.

Tobias De Taillez, Birger Kollmeier, and Bernd T. Meyer. Machine learning for decoding listeners' attention from electroencephalography evoked by continuous speech. 51(5):1234–1241.

Alan D. Degenhart, William E. Bishop, Emily R. Oby, Elizabeth C. Tyler-Kabara, Steven M. Chase, Aaron P. Batista, and Byron M. Yu. 2020. Stabilization of a brain–computer interface via the alignment of low-dimensional spaces of neural activity. 4(7):672–685.

Sibylle Delaloye and Paul E. Holtzheimer. Deep brain stimulation in the treatment of depression. 16(1):83–91.

Arnaud Delorme and Scott Makeig. 2004. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of Neuroscience Methods*, 134(1):9 – 21.

Sharanya Arcot Desai, Thomas Tcheng, and Martha Morrell. Transfer-learning for differentiating epileptic patients who respond to treatment based on chronic ambulatory ECoG data. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 1–4. IEEE.

Kiret Dhindsa, Dean Carcone, and Suzanna Becker. a. A brain-computer interface based on abstract visual and auditory imagery: Evidence for an effect of artistic training. In Dylan D. Schmorrow and Cali M.

Fidopiastis, editors, *Augmented Cognition. Enhancing Cognition and Behavior in Complex Human Environments*, volume 10285, pages 313–332. Springer International Publishing. Series Title: Lecture Notes in Computer Science.

Kiret Dhindsa, Dean Carcone, and Suzanna Becker. b. Toward an open-ended BCI: A user-centered coadaptive design. 29(10):2742–2768.

Jack DiGiovanna, Nadia Dominici, Lucia Friedli, Jacopo Rigosa, Simone Duis, Julie Kreider, Janine Beauparlant, Rubia Van Den Brand, Marco Schieppati, Silvestro Micera, and Grégoire Courtine. Engagement of the rat hindlimb motor cortex across natural locomotor behaviors. 36(40):10440–10455.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction.

Thomas Dohmen, Armin Falk, David Huffman, Uwe Sunde, Jürgen Schupp, and Gert G. Wagner. INDIVIDUAL RISK ATTITUDES: MEASUREMENT, DETERMINANTS, AND BEHAVIORAL CONSEQUENCES. 9(3):522–550.

Thomas Donoghue, Matar Haller, Erik J Peterson, Paroma Varma, Priyadarshini Sebastian, Richard Gao, Torben Noto, Antonio H Lara, Joni D Wallis, Robert T Knight, et al. 2020. Parameterizing neural power spectra into periodic and aperiodic components. *Nature neuroscience*, 23(12):1655–1665.

Khalida Douibi, Solène Le Bars, Alice Lemontey, Lipsa Nag, Rodrigo Balp, and Gabrièle Breda. Toward EEG-based BCI applications for industry 4.0: Challenges and possible applications. 15:705064.

Anming Du, Shuqin Yang, Weijia Liu, and Haiping Huang. Decoding ECoG signal with deep learning model based on LSTM. In *TENCON 2018 - 2018 IEEE Region 10 Conference*, pages 0430–0435. IEEE.

Venkatesh Elango, Aashish N Patel, Kai J Miller, and Vikash Gilja. Sequence transfer learning for neural decoding.

Gamaleldin F. Elsayed and John P. Cunningham. 2017. Structure in neural population recordings: an expected byproduct of simpler phenomena? 20(9):1310–1318. MAG ID: 2742829545 S2ID: 759e1dc783227e53d8febd06d685fd42fa1ec754.

Gamaleldin F. Elsayed, Antonio H. Lara, Matthew T. Kaufman, Mark M. Churchland, and John P. Cunningham. Reorganization between preparatory and movement population responses in motor cortex. 7(1):13239.

Hasan Balkar Erdoğan. 2009. A design and implementation of p300 based brain-computer interface. Master's thesis, Middle East Technical University.

Eyrun Eyjolfsdottir, Kristin Branson, Yisong Yue, and Pietro Perona. Learning recurrent representations for hierarchical behavior modeling.

Andrew H Fagg and Michael A Arbib. 1998. Modeling parietal–premotor interactions in primate control of grasping. 11(7):1277–1303.

Fatemeh Fahimi, Zhuo Zhang, Wooi Boon Goh, Tih-Shi Lee, Kai Keng Ang, and Cuntai Guan. Intersubject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. 16(2):026007.

Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. A brief review of domain adaptation. In Robert Stahlbock, Gary M. Weiss, Mahmoud Abou-Nasr, Cheng-Ying Yang, Hamid R. Arabnia, and Leonidas Deligiannidis, editors, *Advances in Data Science and Information Engineering*, pages 877–894. Springer International Publishing. Series Title: Transactions on Computational Science and Computational Intelligence.

Ali Farshchian, Juan A. Gallego, Joseph P. Cohen, Yoshua Bengio, Lee E. Miller, and Sara A. Solla. Adversarial domain adaptation for stable brain-machine interfaces.

L.A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. 70(6):510–523.

Sarwat Fatima and Awais M Kamboh. 2017. Decoding brain cognitive activity across subjects using multi-modal m/eeg neuroimaging. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3224–3227. IEEE.

Allan Fisher and Jane Margolis. Unlocking the clubhouse: the carnegie mellon experience. 34(2):79–83.

Sharlene N. Flesher, John E. Downey, Jeffrey M. Weiss, Christopher L. Hughes, Angelica J. Herrera, Elizabeth C. Tyler-Kabara, Michael L. Boninger, Jennifer L. Collinger, and Robert A. Gaunt. A brain-computer interface that evokes tactile sensations improves robotic arm control. 372(6544):831–836.

Vladimir Fonov, Alan C. Evans, Kelly Botteron, C. Robert Almli, Robert C. McKinstry, and D. Louis Collins. 2011. Unbiased average age-appropriate atlases for pediatric studies. 54(1):313–327.

Cátia Fortunato, Jorge Bennasar-Vázquez, Junchol Park, Joanna C. Chang, Lee E. Miller, Joshua T. Dudman, Matthew G. Perich, and Juan A. Gallego. 2023. Nonlinear manifolds underlie neural population activity during behaviour.

Markus Frey, Sander Tanni, Catherine Perrodin, Alice O'Leary, Matthias Nau, Jack Kelly, Andrea Banino, Daniel Bendor, Julie Lefort, Christian F Doeller, and Caswell Barry. Interpreting wide-band neural activity using convolutional neural networks. 10:e66551.

Takako Fujioka, Laurel J Trainor, Edward W Large, and Bernhard Ross. 2012. Internalized timing of isochronous sounds is represented in neuromagnetic beta oscillations. *Journal of Neuroscience*, 32(5):1791–1802.

Ryohei Fukuma, Takufumi Yanagisawa, Hiroshi Yokoi, Masayuki Hirata, Toshiki Yoshimine, Youichi Saitoh, Yukiyasu Kamitani, and Haruhiko Kishima. Training in use of brain–machine interface-controlled robotic hand improves accuracy decoding two types of hand movements. 12:478.

Paolo G Gabriel, K J Chen, A Alasfour, T Pailla, W K Doyle, O Devinsky, D Friedman, P Dugan, L Melloni, T Thesen, D Gonda, S Sattar, S G Wang, and V Gilja. Neural correlates of unstructured motor behaviors. 16(6):066026.

Juan A. Gallego, Tamar R. Makin, and Samuel D. McDougle. 2022. Going beyond primary motor cortex to improve brain–computer interfaces. 45(3):176–183.

Juan A. Gallego, Matthew G. Perich, Raeed H. Chowdhury, Sara A. Solla, and Lee E. Miller. a. Long-term stability of cortical population dynamics underlying consistent behavior. 23(2):260–270.

Juan A. Gallego, Matthew G. Perich, Lee E. Miller, and Sara A. Solla. b. Neural manifolds for the control of movement. 94(5):978–984.

Juan A. Gallego, Matthew G. Perich, Stephanie N. Naufel, Christian Ethier, Sara A. Solla, and Lee E. Miller. c. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. 9(1):4233.

Patrick D. Ganzer, Samuel C. Colachis, Michael A. Schwemmer, David A. Friedenberg, Collin F. Dunlap, Carly E. Swiftney, Adam F. Jacobowitz, Doug J. Weber, Marcia A. Bockbrader, and Gaurav Sharma. 2020. Restoring the sense of touch using a sensorimotor demultiplexing neural interface. *Cell*, 181(4):763–773.e12.

Peiran Gao and Surya Ganguli. 2015. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current Opinion in Neurobiology*, 32:148–155. Large-Scale Recording Technology (32).

Rosalinda Garcia, Patricia Morreale, Lara Letaw, Amreeta Chatterjee, Pankati Patel, Sarah Yang, Isaac Tijerina Escobar, Geraldine Jimena Noa, and Margaret Burnett. "regular" CS × inclusive design = smarter students and greater diversity. 23(3):1–35.

Sandra García-Ponsoda, Jorge García-Carrasco, Miguel A. Teruel, Alejandro Maté, and Juan Trujillo. Feature engineering of EEG applied to mental disorders: a systematic mapping study. 53(20):23203–23243.

V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, S. I. Ryu, and K. V. Shenoy. a. A brain machine interface control algorithm designed from a feedback control perspective. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1318–1322. IEEE.

Vikash Gilja, Paul Nuyujukian, Cindy A Chestek, John P Cunningham, Byron M Yu, Joline M Fan, Mark M Churchland, Matthew T Kaufman, Jonathan C Kao, Stephen I Ryu, and Krishna V Shenoy. b. A high-performance neural prosthesis enabled by control algorithm design. 15(12):1752–1757.

Joshua I. Glaser, Ari S. Benjamin, Raeed H. Chowdhury, Matthew G. Perich, Lee E. Miller, and Konrad P. Kording. Machine learning for neural decoding. 7(4):ENEURO.0506–19.2020.

Hosein M. Golshan, Adam O. Hebb, and Mohammad H. Mahoor. LFP-net: A deep learning framework to recognize human behavioral activities using brain STN-LFP signals. 335:108621.

Matthew D. Golub, Patrick T. Sadtler, Emily R. Oby, Kristin M. Quick, Stephen I. Ryu, Elizabeth C. Tyler-Kabara, Aaron P. Batista, Steven M. Chase, and Byron M. Yu. Learning by neural reassociation. 21(4):607–616.

Tensorflow Google. tf.keras.losses.CosineSimilarity.

J. C. Gower. Generalized procrustes analysis. 40(1):33–51.

Alexandre Gramfort. 2013. MEG and EEG data analysis with MNE-python. 7.

Alexandre Gramfort, Martin Luessi, Eric Larson, Denis Engemann, Daniel Strohmeier, Christian Brodbeck, Roman Goj, Mainak Jas, Teon Brooks, Lauri Parkkonen, and Matti Hämäläinen. 2013. Meg and eeg data analysis with mne-python. *Frontiers in Neuroscience*, 7:267.

Valentina Grigoreanu, Margaret Burnett, Susan Wiedenbeck, Jill Cao, Kyle Rector, and Irwin Kwan. a. End-user debugging strategies: A sensemaking perspective. 19(1):1–28.

Valentina I. Grigoreanu, Margaret M. Burnett, and George G. Robertson. b. A strategy-centric approach to the design of end-user debugging tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 713–722. ACM.

Sorin M. Grigorescu, Thorsten Lüth, Christos Fragkopoulos, Marco Cyriacks, and Axel Gräser. A BCI-controlled robotic assistant for quadriplegic people in domestic and professional life. 30(3):419–431.

Xiaotong Gu, Zehong Cao, Alireza Jolfaei, Peng Xu, Dongrui Wu, Tzyy-Ping Jung, and Chin-Teng Lin. 2020. Eeg-based brain-computer interfaces (bcis): A survey of recent studies on signal sensing technologies and computational intelligence approaches and their applications. *arXiv preprint arXiv:2001.11337*.

Christoph Guger, Shahab Daban, Eric Sellers, Clemens Holzner, Gunther Krausz, Roberta Carabalona, Furio Gramatica, and Guenter Edlinger. How many people are able to control a p300-based brain–computer interface (BCI)? 462(1):94–98.

Aysegul Gunduz, Peter Brunner, Amy Daitch, Eric C Leuthardt, Anthony L Ritaccio, Bijan Pesaran, and Gerwin Schalk. 2011. Neural correlates of visual-spatial attention in electrocorticographic signals in humans. *Frontiers in human neuroscience*, 5:89–89.

Jonas Hallström, Helene Elvstrand, and Kristina Hellberg. Gender and technology in free play in swedish early childhood education. 25(2):137–149.

Montaser Hamid, Amreeta Chatterjee, Mariam Guizani, Andrew Anderson, Fatima Moussaoui, Sarah Yang, Isaac Tijerina Escobar, Anita Sarma, and Margaret Burnett. How to measure diversity actionably in technology.

Eva Maria Hammer, Sebastian Halder, Benjamin Blankertz, Claudia Sannelli, Thorsten Dickhaus, Sonja Kleih, Klaus-Robert Müller, and Andrea Kübler. Psychological predictors of SMR-BCI performance. 89(1):80–86.

Kathleen Hartzel. How self-efficacy and gender issues affect software adoption and use. 46(9):167–171.

Malihe Hassani and MohammadReza Karami. Noise estimation in electroencephalogram signal by using volterra series coefficients. 5(3):192.

Chao He, Jialu Liu, Yuesheng Zhu, and Wencai Du. Data augmentation for deep neural networks model in EEG classification task: A review. 15:765525.

He He and Dongrui Wu. 2020. Different set domain adaptation for brain-computer interfaces: A label alignment approach. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(5):1091–1108.

Franz Hell, Paul CJ Taylor, Jan H Mehrkens, and Kai Bötzel. 2018. Subthalamic stimulation, oscillatory activity and connectivity reveal functional role of stn and network mechanisms during decision making under conflict. *Neuroimage*, 171:222–233.

Cornelia Herbert. Analyzing and computing humans by means of the brain using brain-computer interfaces - understanding the user – previous evidence, self-relevance and the user's self-concept as potential superordinate human factors of relevance. 17:1286895.

Christian Herff, Dean J Krusienski, and Pieter Kubben. 2020. The potential of stereotactic-eeg for brain-computer interfaces: Current progress and future directions. *Frontiers in Neuroscience*, 14:123.

Bruce Hoadley. Asymptotic properties of maximum likelihood estimators for the independent not identically distributed case. 42(6):1977–91.

Timothy M Hospedales, Antreas Antoniou, Paul Micaelli, and Amos J. Storkey. Meta-learning in neural networks: A survey. pages 1–1.

Weimin Hou, Manpreet Kaur, Anita Komlodi, Wayne G. Lutters, Lee Boot, Shelia R. Cotten, Claudia Morrell, A. Ant Ozok, and Zeynep Tufekci. "girls don't waste time": pre-adolescent attitudes toward ICT. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pages 875–880. ACM.

Catherine Hu, Christopher Perdriau, Christopher Mendez, Caroline Gao, Abrar Fallatah, and Margaret Burnett. Toward a socioeconomic-aware HCI: Five facets. Publisher: [object Object] Version Number: 1.

Li Hu, P Xiao, ZG Zhang, André Mouraux, and Gian Domenico Iannetti. 2014. Single-trial time–frequency analysis of electrocortical signals: Baseline correction and beyond. *Neuroimage*, 84:876–887.

Gan Huang, Guangquan Liu, Jianjun Meng, Dingguo Zhang, and Xiangyang Zhu. 2010. Model based generalization analysis of common spatial pattern in brain computer interfaces. *Cognitive neurodynamics*, 4(3):217–223.

Ann Hergatt Huffman, Jason Whetten, and William H. Huffman. Using technology in higher education: The influence of gender roles on technology self-efficacy. 29(4):1779–1786.

Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. DeeperCut: A deeper, stronger, and faster multi-person pose estimation model.

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. 36(7):1325–1339.

Saidul Islam, Hanae Elmekki, Ahmed Elsebai, Jamal Bentahar, Najat Drawel, Gaith Rjoub, and Witold Pedrycz. A comprehensive survey on applications of transformers for deep learning tasks.

Andrew Jackson and Thomas M Hall. 2016. Decoding local field potentials for neural interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1705–1714.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. 2020. A Survey on Contrastive Self-supervised Learning. *arXiv:2011.00362 [cs]*. ArXiv: 2011.00362.

M Jeannerod, MA Arbib, G Rizzolatti, and H Sakata. 1995. Grasping objects: the cortical mechanisms of visuomotor transformation. 18(7):314–20.

Will Jernigan, Amber Horvath, Michael Lee, Margaret Burnett, Taylor Cuilty, Sandeep Kuttal, Anicia Peters, Irwin Kwan, Faezeh Bahmani, and Amy Ko. A principled evaluation for a principled idea garden. In *2015 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 235–243. IEEE.

C. Jeunet, B. N'Kaoua, and F. Lotte. a. Advances in user-training for mental-imagery-based BCI control. In *Progress in Brain Research*, volume 228, pages 3–35. Elsevier.

Camille Jeunet, Emilie Jahanpour, and Fabien Lotte. b. Why standard brain-computer interface (BCI) training protocols should be changed: an experimental study. 13(3):036024.

Camille Jeunet, Bernard N'Kaoua, Sriram Subramanian, Martin Hachet, and Fabien Lotte. c. Correction: Predicting mental imagery-based BCI performance from personality, cognitive profile and neurophysiological patterns. 18(2):e0282281.

Camille Jeunet, Bernard N'Kaoua, Sriram Subramanian, Martin Hachet, and Fabien Lotte. d. Predicting mental imagery-based BCI performance from personality, cognitive profile and neurophysiological patterns. 10(12):e0143962.

Jun Jiang, Kira Bailey, and Xiao Xiao. 2018. Midfrontal theta and posterior parietal alpha band oscillations support conflict resolution in a masked affective priming task. *Frontiers in human neuroscience*, 12:175.

Longlong Jing and Yingli Tian. 2019. Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey. *arXiv:1902.06162 [cs]*. ArXiv: 1902.06162.

Robert Evan Johnson, Scott Linderman, Thomas Panier, Caroline Lei Wee, Erin Song, Kristian Joseph

Herrera, Andrew Miller, and Florian Engert. Probabilistic models of larval zebrafish behavior reveal structure on many scales. 30(1):70–82.e4.

Ian T. Jolliffe and Jorge Cadima. 2016. Principal component analysis: a review and recent developments. 374(2065):20150202.

Justin Jude, Matthew G. Perich, Lee E. Miller, and Matthias H. Hennig. 2022. Robust alignment of cross-session recordings of neural population activity by behaviour via unsupervised domain adaptation.

Emily S Kappenman and Steven J Luck. 2016. Best practices for event-related potential research in clinical populations. *Biological psychiatry: cognitive neuroscience and neuroimaging*, 1(2):110–115.

Pierre Karashchuk, Katie L. Rupp, Evyn S. Dickinson, Sarah Walling-Bell, Elischa Sanders, Eiman Azim, Bingni W. Brunton, and John C. Tuthill. Anipose: A toolkit for robust markerless 3d pose estimation. 36(13):109730.

Matthew T Kaufman, Mark M Churchland, Stephen I Ryu, and Krishna V Shenoy. a. Cortical activity in the null space: permitting preparation without movement. 17(3):440–448.

Matthew T. Kaufman, Jeffrey S. Seely, David Sussillo, Stephen I. Ryu, Krishna V. Shenoy, and Mark M. Churchland. b. The largest response component in the motor cortex reflects movement timing but not movement type. 3(4):ENEURO.0085–16.2016.

Ann Kennedy. The what, how, and why of naturalistic behavior. 74:102549.

Kernel. Kernel | home.

S.C. Kleih, F. Nijboer, S. Halder, and A. Kübler. Motivation modulates the p300 amplitude during brain–computer interface use. 121(7):1023–1031.

Sonja C Kleih and Andrea Kubler. Psychological factors influencing brain-computer interface (BCI) performance. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3192–3196. IEEE.

Eran Klein, Tim Brown, Matthew Sample, Anjali R. Truitt, and Sara Goering. Engineering the brain: Ethical issues and the introduction of neural devices. 45(6):26–35.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. Wilds: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR.

Tanja Krumpe, Katrin Baumgaertner, Wolfgang Rosenstiel, and Martin Spüler. 2017. Non-stationarity and inter-subject variability of eeg characteristics in the context of bci development. In *GBCIC*.

Cheryl M. Lacadie, Robert K. Fulbright, Nallakkandi Rajeevan, R. Todd Constable, and Xenophon Papademetris. 2008. More accurate talairach coordinates for neuroimaging using non-linear registration. 42(2):717–725.

Van K. Lam, Chima Oliugbo, Abhijeet Parida, Marius G. Linguraru, and Syed M. Anwar. Self-supervised learning for seizure classification using ECoG spectrograms. In *Medical Imaging 2024: Computer-Aided Diagnosis*, page 93. SPIE.

Elnaz Lashgari, Dehua Liang, and Uri Maoz. Data augmentation for deep-learning-based electroencephalography. 346:108885.

Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. 15(5):056013.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*, 521(7553):436–444. Number: 7553 Publisher: Nature Publishing Group.

D.-Y. Lee, J.-H. Jeong, K.-H. Shim, and S.-W. Lee. a. Decoding movement imagination and execution from EEG signals using BCI-transfer learning method based on relation network.

Douglass B. Lee. Requiem for large-scale models. 39(3):163–178.

Hyeon Kyu Lee and Young-Seok Choi. 2019. Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface. *Entropy*, 21(12):1199.

Michael J. Lee, Faezeh Bahmani, Irwin Kwan, Jilian LaFerte, Polina Charters, Amber Horvath, Fanny Luor, Jill Cao, Catherine Law, Michael Beswetherick, Sheridan Long, Margaret Burnett, and Amy J. Ko. b. Principles of a debugging-first puzzle game for computing education. In *2014 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 57–64. IEEE.

Michael J. Lee and Amy J. Ko. Personifying programming tool feedback improves novice programmers' learning. In *Proceedings of the seventh international workshop on Computing education research*, pages 109–116. ACM.

Min-Ho Lee, O-Yeon Kwon, Yong-Jeong Kim, Hong-Kyung Kim, Young-Eun Lee, John Williamson, Siamac Fazli, and Seong-Whan Lee. c. EEG dataset and OpenBMI toolbox for three BCI paradigms: an investigation into BCI illiteracy. 8(5):giz002.

Nikki Leeuwis, Alissa Paas, and Maryam Alimardani. a. Vividness of visual imagery and personality impact motor-imagery brain computer interfaces. 15:634748.

Nikki Leeuwis, Sue Yoon, and Maryam Alimardani. b. Functional connectivity analysis in motor-imagery brain computer interfaces. 15:732946.

Srdjan Lesaja, Morgan Stuart, Jerry J. Shih, Pedram Z. Soroush, Tanja Schultz, Milos Manic, and Dean J. Krusienski. Self-supervised learning of neural speech representations from unlabeled intracranial signals. 10:133526–133538.

Eric C Leuthardt, Gerwin Schalk, Jonathan R Wolpaw, Jeffrey G Ojemann, and Daniel W Moran. A brain–computer interface using electrocorticographic signals in humans. 1(2):63–71.

Lili Li, Zhongliang Yu, Hui Ma, Zhibin He, Zixiang Zhang, Zhiqing Wu, Yuze Zhang, Zhizhong Wang, Liyu Lin, and Shaolong Kuang. The individual difference of motor imagery ability evoked by visual stimulus and its personality manifestation. 10(5):e26922.

Yitong Li, michael Murias, samantha Major, geraldine Dawson, Kafui Dzirasa, Lawrence Carin, and David E Carlson. 2017. Targeting eeg/lfp synchrony with neural nets. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Zhen Liang, Shigeyuki Oba, and Shin Ishii. An unsupervised EEG decoding system for human emotion recognition. 116:257–268.

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. a. Towards out-of-distribution generalization: A survey.

Xiao Liu, Fanjin Zhang, Zhenyu Hou, Zhaoyu Wang, Li Mian, Jing Zhang, and Jie Tang. 2020. Self-supervised Learning: Generative or Contrastive. *arXiv:2006.08218 [cs, stat]*. ArXiv: 2006.08218.

Yang Liu, Changzhen Qiu, and Zhiyong Zhang. b. Deep learning for 3d human pose estimation and mesh recovery: A survey.

F Lotte, L Bougrain, A Cichocki, M Clerc, M Congedo, A Rakotomamonjy, and F Yger. a. A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. 15(3):031005.

Fabien Lotte and Camille Jeunet. Towards improved BCI based on human learning principles. In *The 3rd International Winter Conference on Brain-Computer Interface*, pages 1–4. IEEE.

Fabien Lotte, Florian Larrue, and Christian Mühl. b. Flaws in current human training protocols for spontaneous brain-computer interfaces: lessons learned from instructional design. 7.

Xuan Ma, Fabio Rizzoglio, Kevin L Bodkin, Eric Perreault, Lee E Miller, and Ann Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. 12:e84296.

Scott Makeig, Anthony J Bell, Tzyy-Ping Jung, and Terrence J Sejnowski. 1996. Independent component analysis of electroencephalographic data. In *Advances in neural information processing systems*, pages 145–151.

Joseph G. Makin, David A. Moses, and Edward F. Chang. Machine translation of cortical activity to text with an encoder–decoder framework.

Jane Margolis and Allan Fisher. *Unlocking the clubhouse: women in computing*. The MIT Press.

Craig A. Marquardt, Victor J. Pokorny, Seth G. Disner, Nathaniel W. Nelson, Kathryn A. McGuire, and Scott R. Sponheim. Inefficient attentional control explains verbal-memory deficits among military veterans with posttraumatic reexperiencing symptoms. 10(3):499–513.

Anne B Martin, Xiaofang Yang, Yuri B Saalmann, Liang Wang, Avgusta Shestyuk, Jack J Lin, Josef Parvizi, Robert T Knight, and Sabine Kastner. 2019a. Temporal dynamics and response modulation across the human visual system in a spatial attention task: An ecog study. *Journal of Neuroscience*, 39(2):333–352.

Stephanie Martin, Josédel R. Millán, Robert T. Knight, and Brian N. Pasley. 2019b. The use of intracranial recordings to decode human language: Challenges and opportunities. *Brain and Language*, 193:73–83.

Alexander Mathis, Pranav Mamidanna, Kevin M. Cury, Taiga Abe, Venkatesh N. Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. 21(9):1281–1289.

Mackenzie Weygandt Mathis and Alexander Mathis. Deep learning tools for the measurement of animal behavior in neuroscience. 60:1–11.

D.J. McFarland and J.R. Wolpaw. Sensorimotor rhythm-based brain-computer interface (BCI): feature selection by regression improves performance. 13(3):372–379.

Roemer van der Meij, Michael Kahana, and Eric Maris. 2012. Phase–amplitude coupling in human electrocorticography is spatially distributed and phase diverse. *Journal of Neuroscience*, 32(1):111–123.

Svenja Melbaum, Eleonora Russo, David Eriksson, Artur Schneider, Daniel Durstewitz, Thomas Brox, and Ilka Diester. Conserved structures of neural activity in sensorimotor cortex of freely moving rats allow cross-subject decoding. 13(1):7420.

Christopher Mendez, Hema Susmita Padala, Zoe Steine-Hanson, Claudia Hilderbrand, Amber Horvath, Charles Hill, Logan Simpson, Nupoor Patil, Anita Sarma, and Margaret Burnett. Open source barriers to entry, revisited: a sociotechnical perspective. In *Proceedings of the 40th International Conference on Software Engineering*, pages 1004–1015. ACM.

Meta. Inside facebook reality labs: Wrist-based interaction for the next computing platform.

Carl Meyer. 2000. *Matrix Analysis and Applied Linear Algebra*. SIAM.

Joan Meyers-Levy and Maheswaran Durairaj. Exploring differences in males' and females' processing strategies. 18(1):63–70.

Joan Meyers-Levy and Barbara Loken. Revisiting gender differences: What we know and what lies ahead. 25(1):129–149.

Jonathan A. Michaels, Benjamin Dann, and Hansjörg Scherberger. Neural population dynamics during reaching are better explained by a dynamical system than representational tuning. 12(11):e1005175.

Microsoft. Microsoft inclusive design.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. ArXiv: 1301.3781.

Kai J. Miller. 2019. A library of human electrocorticographic data and analyses. *Nature Human Behaviour*, 3(11):1225–1235.

Kai J. Miller, Dora Hermes, Christopher J. Honey, Adam O. Hebb, Nick F. Ramsey, Robert T. Knight, Jeffrey G. Ojemann, and Eberhard E. Fetz. Human motor cortical activity is selectively phase-entrained on underlying rhythms. 8(9):e1002655.

Kai J Miller, Dora Hermes, and Nathan P Staff. 2020. The current state of electrocorticography-based brain–computer interfaces. *Neurosurgical Focus*, 49(1):E2.

Kai J. Miller, Eric C. Leuthardt, Gerwin Schalk, Rajesh P.N. Rao, Nicholas R. Anderson, Daniel W. Moran, John W. Miller, and Jeffrey G. Ojemann. 2007. Spectral changes in cortical surface potentials during motor movement. *Journal of Neuroscience*, 27(9):2424–2432.

Yurui Ming, Weiping Ding, Danilo Pelusi, Dongrui Wu, Yu-Kai Wang, Mukesh Prasad, and Chin-Teng Lin. 2019. Subject adaptation network for eeg data analysis. *Applied Soft Computing*, 84:105689.

Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. 2016. Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 527–544, Cham. Springer International Publishing.

MMCV. MMCV.

Christine Morgan. Can we do a better job of detecting suicidality? u of m team using a DARPA award to test the possibility.

Tim R. Mullen, Christian A. E. Kothe, Yu Mike Chi, Alejandro Ojeda, Trevor Kerth, Scott Makeig, Tzyy-Ping Jung, and Gert Cauwenberghs. Real-time neuroimaging and cognitive monitoring using wearable dry EEG. 62(11):2553–2567.

Jordan Muraskin, Truman R Brown, Jennifer M Walz, Tao Tu, Bryan Conroy, Robin I Goldman, and Paul Sajda. 2018. A multimodal encoding model applied to imaging decision-related neural cascades in the human brain. *NeuroImage*, 180:211–222.

Ander Ramos Murguialday, Vikram Aggarwal, Aniruddha Chatterjee, Yoonju Cho, Robert Rasmussen, Brandon O'Rourke, Soumyadipta Acharya, and Nitish V. Thakor. Brain-computer interface for a prosthetic hand using local machine control and haptic feedback. In *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pages 609–613. IEEE.

Masaki Nakanishi, Yu-Te Wang, Chun-Shu Wei, Kuan-Jung Chiang, and Tzyy-Ping Jung. 2020. Facilitating calibration in high-speed BCI spellers via leveraging cross-device shared latent responses. *IEEE Transactions on Biomedical Engineering*, 67(4):1105–1113.

Christoforos Nalmpantis, Odysseas Krystalakos, and Dimitris Vrakas. 2018. Energy profile representation in vector space. In *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, SETN '18, pages 1–5, New York, NY, USA. Association for Computing Machinery.

Christoforos Nalmpantis and Dimitris Vrakas. 2019. Signal2Vec: Time Series Embedding Representation. pages 80–90.

Nikhilesh Natraj, Sarah Seko, Reza Abiri, Hongyi Yan, Yasmin Graham, Adelyn Tu-Chan, Edward F. Chang, and Karunesh Ganguly. a. Flexible regulation of representations on a drifting manifold enables long-term stable complex neuroprosthetic control.

Nikhilesh Natraj, Daniel B. Silversmith, Edward F. Chang, and Karunesh Ganguly. b. Compartmentalized dynamics within a common multi-area mesoscale manifold represent a repertoire of human hand movements. 110(1):154–174.e12.

Stephanie Naufel, Joshua I. Glaser, Konrad P. Kording, Eric J. Perreault, and Lee E. Miller. A muscle-activity-dependent gain between motor cortex and EMG. 121(1):61–73.

Neurable. Neurable.

Neuralink. Neuralink was live.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. 2020. What is being transferred in transfer learning? In *Advances in Neural Information Processing Systems*, volume 33, pages 512–523. Curran Associates, Inc.

Patrick Ng. 2017. dna2vec: Consistent vector representations of variable-length k-mers. *arXiv:1701.06279 [cs, q-bio, stat]*. ArXiv: 1701.06279.

Chuong H. Nguyen, George K. Karavas, and Panagiotis Artemiadis. Adaptive multi-degree of freedom brain computer interface using online feedback: Towards novel methods and metrics of mutual adaptation between humans and machines for BCI. 14(3):e0212620.

Nijboer. The influence of psychological state and motivation on brain-computer interface performance in patients with amyotrophic lateral sclerosis - a longitudinal study.

Femke Nijboer, Adrian Furdea, Ingo Gunst, Jürgen Mellinger, Dennis J. McFarland, Niels Birbaumer, and Andrea Kübler. a. An auditory brain–computer interface (BCI). 167(1):43–50.

Femke Nijboer, Fabrice O. Morin, Stefan P. Carmien, Randal A. Koene, Enrique Leon, and Ulrich Hoffmann. b. Affective brain-computer interfaces: Psychophysiological markers of emotion in healthy per-

sons and in persons with amyotrophic lateral sclerosis. In *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–11. IEEE.

Soroush Niketeghad and Nader Pouratian. 2019. Brain machine interfaces for vision restoration: The current state of cortical visual prosthetics. *Neurotherapeutics*, 16(1):134–143.

Donald A. Norman. *The design of everyday things*, revised and expanded edition edition. Basic Books.

Magdalena Nowak, Catharina Zich, and Charlotte J Stagg. 2018. Motor cortical gamma oscillations: What have we learnt and where are we headed? *Current behavioral neuroscience reports*, 5(2):136–142.

Emily R Oby, Jay A Hennig, Aaron P Batista, M Yu Byron, and Steven M Chase. 2020. Intracortical brain–machine interfaces. In *Neural Engineering*, pages 185–221. Springer.

Patrick Ofner, Andreas Schwarz, Joana Pereira, and Gernot R. Müller-Putz. 2017. Upper limb movements can be decoded from the time-domain of low-frequency eeg. *PLOS ONE*, 12(8):1–24.

John O'Keefe. Place units in the hippocampus of the freely moving rat. 51(1):78–109.

A. H. Omidvarnia, F. Atry, S. K. Setarehdan, and B. N. Arabi. Kalman filter parameters as a new EEG feature vector for BCI applications. pages pp. 1–4.

Julie Onton, Marissa Westerfield, Jeanne Townsend, and Scott Makeig. 2006. Imaging human eeg dynamics using independent component analysis. *Neuroscience & biobehavioral reviews*, 30(6):808–822.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding.

Robert Oostenveld, Pascal Fries, Eric Maris, and Jan-Mathijs Schoffelen. 2011. FieldTrip: Open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. 2011:1–9.

Robert Oostenveld and Peter Praamstra. 2001. The five percent electrode system for high-resolution eeg and erp measurements. *Clinical Neurophysiology*, 112(4):713–719.

OpenBCI. OpenBCI | home.

Amy L. Orsborn, Helene G. Moorman, Simon A. Overduin, Maryam M. Shanechi, Dragan F. Dimitrov, and Jose M. Carmena. Closed-loop decoder adaptation shapes neural plasticity for skillful neuroprosthetic control. 82(6):1380–1393.

Simon A. Overduin, Andrea d'Avella, Jinsook Roh, Jose M. Carmena, and Emilio Bizzi. Representation of muscle synergies in the primate brain. 35(37):12615–12624.

LLW Owen, TA Muntianu, AC Heusser, PM Daly, KW Scangos, and JR Manning. 2020. A gaussian process model of human electrocorticographic data. *Cerebral Cortex (New York, NY: 1991)*.

Gang Pan, Jia-Jun Li, Yu Qi, Hang Yu, Jun-Ming Zhu, Xiao-Xiang Zheng, Yue-Ming Wang, and Shao-Min Zhang. Rapid decoding of hand gestures in electrocorticography using recurrent neural networks. 12:555.

Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, L. F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. 15(10):805–815.

Xenios Papademetris and Yale University. 2014. Yale BioImage suite - MNI to talairach converter.

Jisung Park and Sung-Phil Kim. Estimation of speed and direction of arm movements from m1 activity using a nonlinear neural decoder. In *2019 7th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–4. IEEE.

Lucas C Parra, Clay D Spence, Adam D Gerson, and Paul Sajda. 2005. Recipes for the linear analysis of eeg. *Neuroimage*, 28(2):326–341.

Josef Parvizi and Sabine Kastner. 2018. Promises and limitations of human intracranial electroencephalography. *Nature Neuroscience*, 21(4):474–483.

Courtnie Paschall, Rajesh P.N. Rao, Jason Hauptman, Jeffrey G. Ojemann, and Jeffrey Herron. 2023. An immersive virtual reality platform integrating human ECOG & sEEG: Implementation & noise analysis.

Leisi Pei and Guang Ouyang. Online recognition of handwritten characters from scalp-recorded brain activities during handwriting. 18(4):046070.

Steven M Peterson and Daniel P Ferris. 2018. Differentiation in theta and beta electrocortical activity between visual and physical perturbations to walking and standing balance. *eneuro*, 5(4).

Steven M Peterson, Rajesh P N Rao, and Bingni W Brunton. a. Learning neural decoders without labels using multiple data streams. 19(4):046032.

Steven M Peterson, Estefania Rios, and Daniel P. Ferris. 2018. Transient visual perturbations boost short-term balance learning in virtual reality by modulating electrocortical activity. *Journal of neurophysiology*, 120 4:1998–2010.

Steven M. Peterson, Satpreet H. Singh, Benjamin Dichter, Michael Scheid, Rajesh P. N. Rao, and Bingni W. Brunton. 2022. AJILE12: Long-term naturalistic human intracranial neural recordings and pose. 9(1):184.

Steven M. Peterson, Satpreet H. Singh, Nancy X. R. Wang, Rajesh P. N. Rao, and Bingni W. Brunton. 2021. Behavioral and neural variability of naturalistic arm movements. 8(3):ENEURO.0007–21.2021.

Steven M Peterson, Zoe Steine-Hanson, Nathan Davis, Rajesh P N Rao, and Bingni W Brunton. b. Generalized neural decoders for transfer learning across participants and recording modalities. 18(2):026014.

Maria Carla Piastra, Andreas Nüßing, Johannes Vorwerk, Maureen Clerc, Christian Engwer, and Carsten H. Wolters. A comprehensive study on electroencephalography and magnetoencephalography sensitivity to cortical and subcortical sources. 42(4):978–992.

Léa Pillette, Aline Roc, Bernard N'Kaoua, and Fabien Lotte. Experimenters' influence on mental-imagery based brain-computer interface user training. 149:102603.

Tobias Pistohl, Tonio Ball, Andreas Schulze-Bonhage, Ad Aertsen, and Carsten Mehring. 2008. Prediction of arm movement trajectories from ecog-recordings in humans. *Journal of Neuroscience Methods*, 167(1):105–114.

Alain Rakotomamonjy and Vincent Guigue. BCI competition III: Dataset II- ensemble of SVMs for BCI p300 speller. 55(3):1147–1154.

Adriane B. Randolph. Not all created equal: Individual-technology fit of brain-computer interfaces. In *2012 45th Hawaii International Conference on System Sciences*, pages 572–578. IEEE.

Rajesh P. N. Rao. 2013. *Brain-Computer Interfacing: An Introduction*. Cambridge University Press, Cambridge.

Rajesh P. N. Rao and Dana H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. 2(1):79–87.

Rajesh P. N. Rao, Dimitrios C. Gklezakos, and Vishwas Sathish. Active predictive coding: A unified neural framework for learning hierarchical world models for perception and planning.

Mirco Ravanelli and Yoshua Bengio. 2018. Interpretable convolutional filters with sincnet. *ArXiv*, abs/1811.09725.

Riedl, Hubert, and Kenning. Are there neural gender differences in online trust? an fMRI study on the perceived trustworthiness of eBay offers. 34(2):397.

Fabio Rizzoglio, Ege Altan, Xuan Ma, Kevin L. Bodkin, Brian M. Dekleva, Sara A. Solla, Ann Kennedy, and Lee E. Miller. Monkey-to-human transfer of brain-computer interface decoders.

Edmund T. Rolls, Chu-Chung Huang, Ching-Po Lin, Jianfeng Feng, and Marc Joliot. 2020. Automated anatomical labelling atlas 3. 206:116189.

Matthew Rosenberg, Tony Zhang, Pietro Perona, and Markus Meister. Mice in a labyrinth show rapid learning, sudden insight, and efficient exploration. 10:e66175.

Daniela Rosner and Jonathan Bean. Learning from IKEA hacking: i'm not one to decoupage a tabletop and call it a day. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 419–422. ACM.

Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. 290(5500):2323–2326.

Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. 2019. Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*, 16(5):051001.

Michael E Rule, Timothy O'Leary, and Christopher D Harvey. Causes and consequences of representational drift. 58:141–147.

Patrick T. Sadtler, Kristin M. Quick, Matthew D. Golub, Steven M. Chase, Stephen I. Ryu, Elizabeth C. Tyler-Kabara, Byron M. Yu, and Aaron P. Batista. Neural constraints on learning. 512(7515):423–426.

Mostafa Safaie, Joanna C. Chang, Junchol Park, Lee E. Miller, Joshua T. Dudman, Matthew G. Perich, and Juan A. Gallego. Preserved neural population dynamics across animals performing similar behaviour.

Simanto Saha and Mathias Baumert. 2019. Intra-and inter-subject variability in eeg-based sensorimotor brain computer interface: a review. *Frontiers in Computational Neuroscience*, 13:87.

Siavash Sakhavi, Cuntai Guan, and Shuicheng Yan. 2018. Learning temporal information for brain-computer interface using convolutional neural networks. *IEEE transactions on neural networks and learning systems*, 29(11):5619–5629.

Omid G Sani, Yuxiao Yang, Morgan B Lee, Heather E Dawes, Edward F Chang, and Maryam M Shanechi. 2018. Mood variations decoded from multi-site intracranial human brain activity. *Nature Biotechnology*, 36(10):954–961.

Gopal Santhanam, Byron M. Yu, Vikash Gilja, Stephen I. Ryu, Afsheen Afshar, Maneesh Sahani, and Krishna V. Shenoy. Factor-analysis methods for higher-performance neural prostheses. 102(2):1315–1330.

Pritam Sarkar and Ali Etemad. 2020. Self-supervised ECG Representation Learning for Emotion Recognition. *IEEE Transactions on Affective Computing*, pages 1–1. ArXiv: 2002.03898.

Stefan Schaffelhofer and Hansjörg Scherberger. 2016. Object vision to hand action in macaque parietal, premotor, and motor cortices. 5:e15278.

G. Schalk and E. C. Leuthardt. 2011. Brain-computer interfaces using electrocorticographic signals. *IEEE Reviews in Biomedical Engineering*, 4:140–154.

Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. 38(11):5391–5420.

Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. 2017. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420.

Mohamed L. Seghier. 2013. The angular gyrus: Multiple functions and multiple subdivisions. 19(1):43–61.

Arun Shekhar and Nicola Marsden. Cognitive walkthrough of a learning management system with gendered personas. In *Proceedings of the 4th Conference on Gender & IT - GenderIT '18*, pages 191–198. ACM Press.

Krishna V. Shenoy, Maneesh Sahani, and Mark M. Churchland. Cortical control of arm movements: A dynamical systems perspective. 36(1):337–359.

Sima Shirzadi, Mehrdad Dadgostar, Zahra Einalou, Sinem Burcu Erdoğan, and Ata Akin. Sex based differences in functional connectivity during a working memory task: an fNIRS study. 15:1207202.

Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. 6(1):60.

Dilruba Showkat and Cindy Grimm. Identifying gender differences in information processing style, self-efficacy, and tinkering for robot tele-operation. In *2018 15th International Conference on Ubiquitous Robots (UR)*, pages 443–448. IEEE.

Steven John Simon. The impact of culture and gender on web sites: an empirical study. 32(1):18–37.

Anil Singh, Vikram Bhadauria, Anurag Jain, and Anil Gurung. a. Role of gender, self-efficacy, anxiety and testing formats in learning spreadsheets. 29(3):739–746.

Satpreet H. Singh, Steven M. Peterson, Rajesh P. N. Rao, and Bingni W. Brunton. 2020. Towards naturalistic human neuroscience and neuroengineering: behavior mining in long-term video and neural recordings.

Satpreet H. Singh, Steven M. Peterson, Rajesh P.N. Rao, and Bingni W. Brunton. b. Mining naturalistic human behaviors in long-term video and neural recordings. 358:109199.

International Neuroethics Society. Minding the gap: Equity and justice in AI and neurotech.

T. Sollfrank, A. Ramsay, S. Perdikis, J. Williamson, R. Murray-Smith, R. Leeb, J.d.R. Millán, and A. Kübler. The effect of multimodal and enriched feedback on SMR-BCI performance. 127(1):490–498.

Ramesh Srinivasan, Don M Tucker, and Michael Murias. 1998. Estimating the spatial nyquist of the human eeg. *Behavior Research Methods, Instruments, & Computers*, 30(1):8–19.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Zoe Steine-Hanson, Claudia Hilderbrand, Lara Letaw, Jillian Emard, Christopher Perdriau, Christopher Mendez, Margaret Burnett, and Anita Sarma. Fixing inclusivity bugs for information processing styles and learning styles.

Arjen Stolk, Sandon Griffin, Roemer Van Der Meij, Callum Dewar, Ignacio Saez, Jack J. Lin, Giovanni Piantoni, Jan-Mathijs Schoffelen, Robert T. Knight, and Robert Oostenveld. 2018. Integrated analysis of anatomical and electrophysiological human intracranial data. 13(7):1699–1723.

Baochen Sun, Jiashi Feng, and Kate Saenko. a. Correlation alignment for unsupervised domain adaptation. In Gabriela Csurka, editor, *Domain Adaptation in Computer Vision Applications*, pages 153–171. Springer International Publishing. Series Title: Advances in Computer Vision and Pattern Recognition.

Xulu Sun, Daniel J. O'Shea, Matthew D. Golub, Eric M. Trautmann, Saurabh Vyas, Stephen I. Ryu, and Krishna V. Shenoy. b. Cortical preparatory activity indexes learned motor memories. 602(7896):274–279.

Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. DeepSleepNet: a model for automatic sleep stage scoring based on raw single-channel EEG. 25(11):1998–2008.

David Sussillo, Mark M Churchland, Matthew T Kaufman, and Krishna V Shenoy. a. A neural network that finds a naturalistic solution for the production of muscle activity. 18(7):1025–1033.

David Sussillo, Rafal Jozefowicz, L. F. Abbott, and Chethan Pandarinath. b. LFADS - latent factor analysis via dynamical systems. ARXIV_ID: 1608.06315 MAG ID: 2513369614 S2ID: e9bfa6320013c56126f8ce8a7da08bff3a5ca88b.

Kana Takaura, Naotsugu Tsuchiya, and Naotaka Fujii. 2016. Frequency-dependent spatiotemporal profiles of visual responses recorded with subdural ecog electrodes in awake monkeys: Differences between high- and low-frequency activity. *NeuroImage*, 124:557 – 572.

Chuanqi Tan, Fuchun Sun, Tao Kong, Bin Fang, and Wenchang Zhang. a. Attention-based transfer learning for brain-computer interface.

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *ICANN*.

Chuanqi Tan, Fuchun Sun, and Wenchang Zhang. b. Deep transfer learning for EEG-based brain computer interface.

Jerry Tang, Amanda LeBel, Shailee Jain, and Alexander G. Huth. Semantic reconstruction of continuous language from non-invasive brain recordings. 26(5):858–866.

Zied Tayeb, Juri Fedjaev, Nejla Ghaboosi, Christoph Richter, Lukas Everding, Xingwei Qu, Yingyu Wu, Gordon Cheng, and Jörg Conradt. 2019. Validating deep neural networks for online decoding of motor imagery movements from eeg signals. *Sensors*, 19(1):210.

Thomas Thiery, Tarek Lajnef, Karim Jerbi, Martin Arguin, Mercedes Aubin, and Pierre Jolicoeur. Decoding the locus of covert visuospatial attention from EEG signals. 11(8):e0160304.

Armin W. Thomas, Klaus-Robert Müller, and Wojciech Samek. Deep transfer learning for whole-brain fMRI analyses.

Margaret C. Thompson. Critiquing the concept of BCI illiteracy. 25(4):1217–1233.

L.J. Trejo, R. Rosipal, and B. Matthews. Brain-computer interfaces for 1-d and 2-d cursor control: designs using volitional control of the EEG spectrum or steady-state visual evoked potentials. 14(2):225–229.

Po-He Tseng, Núria Armengol Urpi, Mikhail Lebedev, and Miguel Nicolelis. Decoding movements from cortical ensemble activity using a long short-term memory recurrent network. 31(6):1085–1113.

M. Tsodyks, T. Kenet, A. Grinvald, and A. Arieli. Linking spontaneous activity of single cortical neurons and the underlying functional architecture. 286(5446):1943–1946.

N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou, F. Crivello, O. Etard, N. Delcroix, B. Mazoyer, and M. Joliot. 2002. Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *NeuroImage*, 15(1):273 – 289.

Nachum Ulanovsky and Cynthia F Moss. Hippocampal cellular and network activity in freely moving echolocating bats. 10(2):224–233.

Axel Uran, Coert Van Gemeren, Rosanne van Diepen, Ricardo Chavarriaga, and José del R Millán. 2019. Applying transfer learning to deep learned models for eeg analysis. *arXiv preprint arXiv:1907.01332*.

Jan Van Erp, Fabien Lotte, and Michael Tangermann. 2012. Brain-computer interfaces: beyond medical applications. *Computer*, 45(4):26–34.

Charles Van Loan. 1992. *Computational frameworks for the fast Fourier transform*. SIAM.

M Vermaas, MC Piastra, TF Oostendorp, NF Ramsey, and PHE Tiesinga. 2020. Femfuns: A volume conduction modeling pipeline that includes resistive, capacitive or dispersive tissue and electrodes. *Neuroinformatics*.

Jean-Philippe Vert, Koji Tsuda, and Bernhard Schölkopf. 2004. A primer on kernel methods. *Kernel methods in computational biology*, 47:35–70.

J J Vidal. Toward direct brain-computer communication. 2(1):157–180.

Carmen Vidaurre, Claudia Sannelli, Klaus-Robert Müller, and Benjamin Blankertz. Machine-learning-based coadaptive calibration for brain-computer interfaces. 23(3):791–816.

Martin Volker, Robin T. Schirrmeister, Lukas D. J. Fiederer, Wolfram Burgard, and Tonio Ball. Deep transfer learning for error decoding from non-invasive EEG. In *2018 6th International Conference on Brain-Computer Interface (BCI)*, pages 1–6. IEEE.

Ksenia Volkova, Mikhail A Lebedev, Alexander Kaplan, and Alexei Ossadtchi. 2019. Decoding movement from electrocorticographic activity: A review. *Frontiers in neuroinformatics*, 13.

Nataliia N. Volkova* and Alexey N. Gusev. Do cognitive styles affect psychophysical tasks performance? pages 445–452.

Mihaela Vorvoreanu, Lingyi Zhang, Yun-Han Huang, Claudia Hilderbrand, Zoe Steine-Hanson, and Margaret Burnett. From gender biases to gender-inclusive design: An empirical investigation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–14. ACM.

Feng Wan, Janir Nuno Da Cruz, Wenya Nan, Chi Man Wong, Mang I Vai, and Agostinho Rosa. Alpha neurofeedback training improves SSVEP-based BCI performance. 13(3):036019.

Hua Wang, Yong Xu, Hongwen Song, Tianxin Mao, Yan Huang, Sihua Xu, Xiaochu Zhang, and Hengyi Rao. a. State boredom partially accounts for gender differences in novel lexicon learning. 13:807558.

Jialin Wang, Yanchun Zhang, Qinying Ma, Huihui Huang, and Xiaoyuan Hong. b. Deep learning for single-channel EEG signals sleep stage scoring based on frequency domain representation. In Hua Wang, Siuly Siuly, Rui Zhou, Fernando Martin-Sanchez, Yanchun Zhang, and Zhisheng Huang, editors, *Health Information Science*, volume 11837, pages 121–133. Springer International Publishing. Series Title: Lecture Notes in Computer Science.

Nancy Wang, Ali Farhadi, Rajesh Rao, and Bingni Brunton. 2018. AJILE movement prediction: Multimodal deep learning for natural human neural recordings and video. 32(1).

Nancy X. R. Wang, Jared D. Olson, Jeffrey G. Ojemann, Rajesh P. N. Rao, and Bingni W. Brunton. c. Unsupervised decoding of long-term, naturalistic human neural recordings with automated video and audio annotations. 10.

Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. d. Tracking everything everywhere all at once.

Ran Wang, Xupeng Chen, Amirhossein Khalilian-Gourtani, Zhaoxi Chen, Leyao Yu, Adeen Flinker, and Yao Wang. e. Stimulus speech decoding from human cortex with generative adversarial network transfer learning. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 390–394. IEEE.

Wei Wang, Jennifer L. Collinger, Alan D. Degenhart, Elizabeth C. Tyler-Kabara, Andrew B. Schwartz, Daniel W. Moran, Douglas J. Weber, Brian Wodlinger, Ramana Vinjamuri, Robin C. Ashmore, John W. Kelly, and Michael L. Boninger. 2013. An electrocorticographic brain interface in an individual with tetraplegia. *PLoS ONE*, 8.

Xin Wang, Ali Farhadi, Rajesh Rao, and Bingni Brunton. f. AJILE movement prediction: Multimodal deep learning for natural human neural recordings and video.

Y. Wang, W. Truccolo, and D. A. Borton. g. Decoding hindlimb kinematics from primate motor cortex using long short-term memory recurrent neural networks. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1944–1947. IEEE.

Universty of Waterloo. Understanding learning styles.

Elke U. Weber, Ann-Renée Blais, and Nancy E. Betz. A domain-specific risk-attitude scale: measuring risk perceptions and risk behaviors. 15(4):263–290.

Qingsong Wen, Liang Sun, Fan Yang, Xiaomin Song, Jingkun Gao, Xue Wang, and Huan Xu. Time series data augmentation for deep learning: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 4653–4660.

Weining Weng, Yang Gu, Shuai Guo, Yuan Ma, Zhaohua Yang, Yuchen Liu, and Yiqiang Chen. Self-supervised learning for electroencephalogram: A systematic survey.

Cathleen Wharton, John Rieman, Clayton Lewis, and Peter Polson. The cognitive walkthrough method: a practitioner's guide. pages 105–140.

Alex H. Williams, Erin Kunz, Simon Kornblith, and Scott W. Linderman. 2022. Generalized shape metrics on neural representations.

Alexander B. Wiltschko, Matthew J. Johnson, Giuliano Iurilli, Ralph E. Peterson, Jesse M. Katon, Stan L. Pashkovski, Victoria E. Abraira, Ryan P. Adams, and Sandeep Robert Datta. Mapping sub-second structure in mouse behavior. 88(6):1121–1135.

Jonathan Wolpaw and Elizabeth Winter Wolpaw. *Brain–Computer InterfacesPrinciples and Practice*. Oxford University Press.

Guilherme Wood and Silvia Erika Kober. EEG neurofeedback is under strong control of psychosocial factors. 43(4):293–300.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time.

Dongrui Wu, Brent J Lance, Vernon J Lawhern, Stephen Gordon, Tzyy-Ping Jung, and Chin-Teng Lin. 2017. Eeg-based user reaction time estimation using riemannian geometry features. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11):2157–2168.

Dongrui Wu, Vernon J. Lawhern, W. David Hairston, and Brent J. Lance. 2016. Switching EEG headsets made easy: Reducing offline calibration effort using active weighted adaptation regularization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(11):1125–1137.

Dongrui Wu, Yifan Xu, and Bao-Liang Lu. Transfer learning for EEG-based brain-computer interfaces: A review of progress made since 2016.

Wei Wu and Nicholas G. Hatsopoulos. Real-time decoding of nonstationary neural activity in motor cortex. 16(3):213–222.

Ziqian Xie, Odelia Schwartz, and Abhishek Prasad. Decoding of finger trajectory from ECoG using deep learning. 15(3):036009.

Lichao Xu, Minpeng Xu, Yufeng Ke, Xingwei An, Shuang Liu, and Dong Ming. 2020. Cross-dataset variability problem in eeg decoding with deep learning. *Frontiers in Human Neuroscience*, 14.

Hong Yang and Travis Desell. Robust augmentation for multivariate time series classification.

Xinyu Yang, Zhenguo Zhang, Xu Cui, and Rongyi Cui. A time series data augmentation method based on dynamic time warping. In *2021 International Conference on Computer Communication and Artificial Intelligence (CCAI)*, pages 116–120. IEEE.

Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. On early stopping in gradient descent learning. 26(2):289–315.

RK Rao Yarlagadda. 2010. *Analog and digital signals and systems*, volume 1. Springer.

Michael M. Yartsev and Nachum Ulanovsky. Representation of three-dimensional space in the hippocampus of flying bats. 340(6130):367–372.

Florian Yger, Maxime Berar, and Fabien Lotte. 2016. Riemannian approaches in brain-computer interfaces: a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1753–1762.

Gilsang Yoo, Hyeoncheol Kim, and Sungdae Hong. Prediction of cognitive load from electroencephalography signals using long short-term memory network. 10(3):361.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen Ryu, Krishna V Shenoy, and Maneesh Sahani. 2008. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

Ye Yuan, Guangxu Xun, Qiuling Suo, Kebin Jia, and Aidong Zhang. 2017. Wave2Vec: Learning Deep Representations for Biosignals. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 1159–1164. ISSN: 2374-8486.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. CutMix: Regularization strategy to train strong classifiers with localizable features.

Baltazar Zavala, Huiling Tan, Keyoumars Ashkan, Thomas Foltynie, Patricia Limousin, Ludvic Zrinzo, Kareem Zaghloul, and Peter Brown. 2016. Human subthalamic nucleus–medial frontal cortex theta phase coherence is involved in conflict and error related cortical monitoring. *Neuroimage*, 137:178–187.

Changqing Zeng, Zhendong Mu, and Qingjun Wang. Classifying driving fatigue by using EEG signals. 2022:1–13.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. a. mixup: Beyond empirical risk minimization.

Libby Zhang, Amber Hu, Claire Bedbrook, Ravi Nath, Karl Deisseroth, Anne Brunet, and Scott Linderman. b. From behavioral syllables to the book of life: A dynamic behavioral topic model. COSYNE 2023.

Xiang Zhang, Lina Yao, Xianzhi Wang, Jessica Monaghan, David Mcalpine, and Yu Zhang. 2019. A survey on deep learning based brain computer interface: Recent advances and new frontiers. *arXiv preprint arXiv:1905.04149*.

Qing Zhou, Ruidong Cheng, Lin Yao, Xiangming Ye, and Kedi Xu. Neurofeedback training of alpha relative power improves the performance of motor imagery brain-computer interface. 16:831995.

Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. MotionBERT: A unified perspective on learning human motion representations.

Catharina Zich, Stefan Debener, Ann-Kathrin Thoene, Ling-Chia Chen, and Cornelia Kranczioch. Simultaneous EEG-fNIRS reveals how age and feedback affect motor imagery signatures. 49:183–197.

Quan Zou, Pengwei Xing, Leyi Wei, and Bin Liu. 2019. Gene2vec: gene subsequence embedding for prediction of mammalian N6-methyladenosine sites from mRNA. *RNA*, 25(2):205–218. Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab.

Ivan Zubarev, Rasmus Zetter, Hanna-Leena Halme, and Lauri Parkkonen. Adaptive neural network classifier for decoding MEG signals.

Maciej Śliwowski, Matthieu Martin, Antoine Souloumiac, Pierre Blanchart, and Tetiana Aksenova. 2022. Decoding ECoG signal into 3d hand translation using deep learning. 19(2):026023.

Jarosław Żygierewicz, Romuald A Janik, Igor T Podolak, Alan Drozd, Urszula Malinowska, Martyna Poziomska, Jakub Wojciechowski, Paweł Ogniewski, Paweł Niedbalski, Iwona Terczynska, and Jacek Rogala. Decoding working memory-related information from repeated psychophysiological EEG experiments using convolutional and contrastive neural networks. 19(4):046053.

# Chapter A

# Appendix One
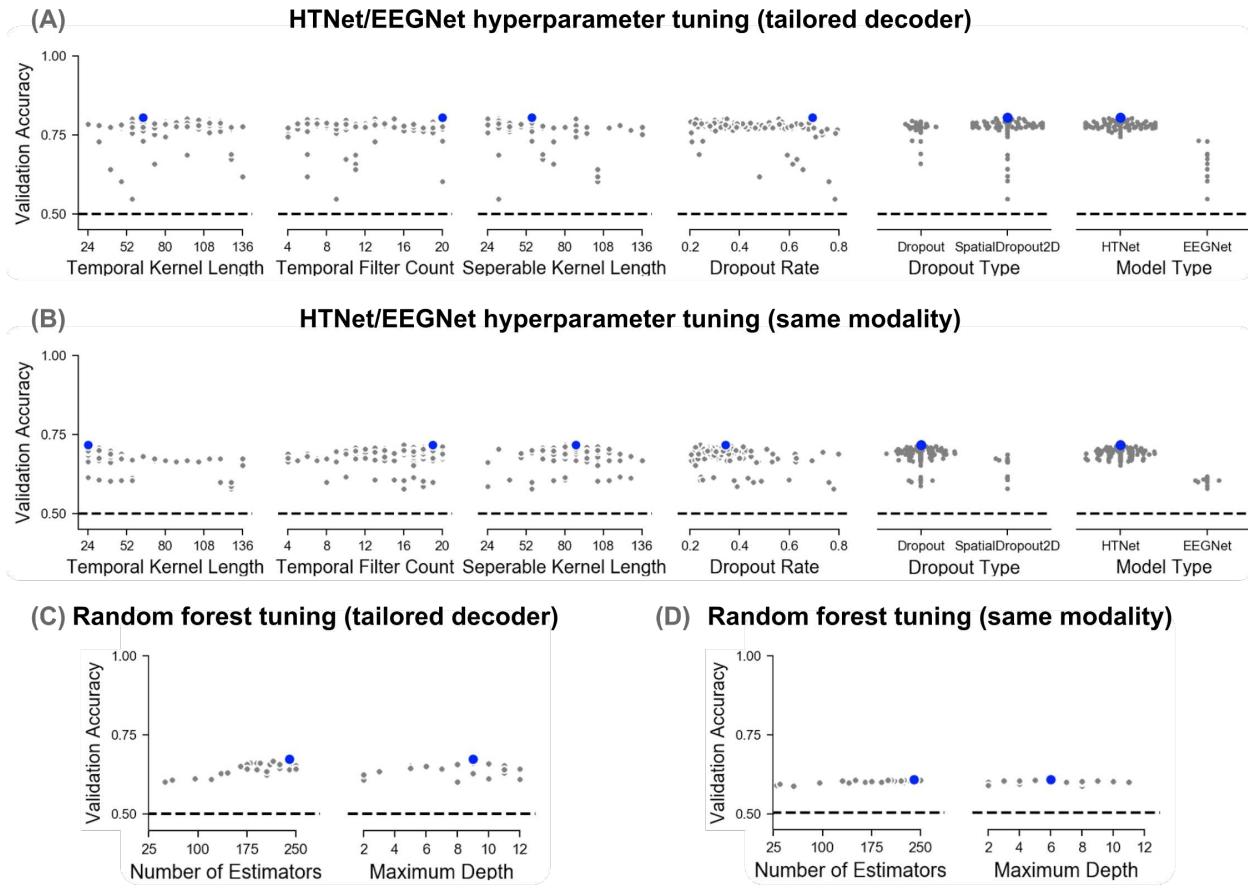
Appendix for Chapter 4 - Generalized neural decoders for transfer learning across participants and recording modalities

| Hyperparameter | Tailored Decoder | Same Modality |
|---|---|---|
| Dropout Rate | 0.69 | 0.34 |
| Temporal Kernel Length (K1) | 64 | 24 |
| Temporal Filter Count (F1) | 20 | 19 |
| Separable Kernel Length (K2) | 56 | 88 |
| Dropout Type | Spatial Dropout 2D | Dropout |
| Number of Estimators | 240 | 240 |
| Maximum Depth | 9 | 6 |

**Table A.1: Optimal parameter values from hyperparameter tuning.** Parameter values are shown for the hyperparameter tuning run with the highest accuracy on the validation data. We tuned hyperparameters separately for tailored decoder and same modality conditions. The first five parameters are for HTNet and EEGNet decoders, while the last two are for random forest decoders; the minimum distance decoder we used had no trainable parameters. Note that we used the same exact trained decoders for the same and unseen modality conditions; only the test set differed.

| | HTNet | EEGNet | Random Forest | Minimum Distance |
|---|---|---|---|---|
| Tailored Decoder | 117.7 (89.4) | 66.4 (49.8) | 12.2 (8.0) | 6.9 (5.7) |
| Same Modality | 430.3 (173.1) | 185.0 (63.6) | 99.6 (12.8) | 247.7 (33.2) |
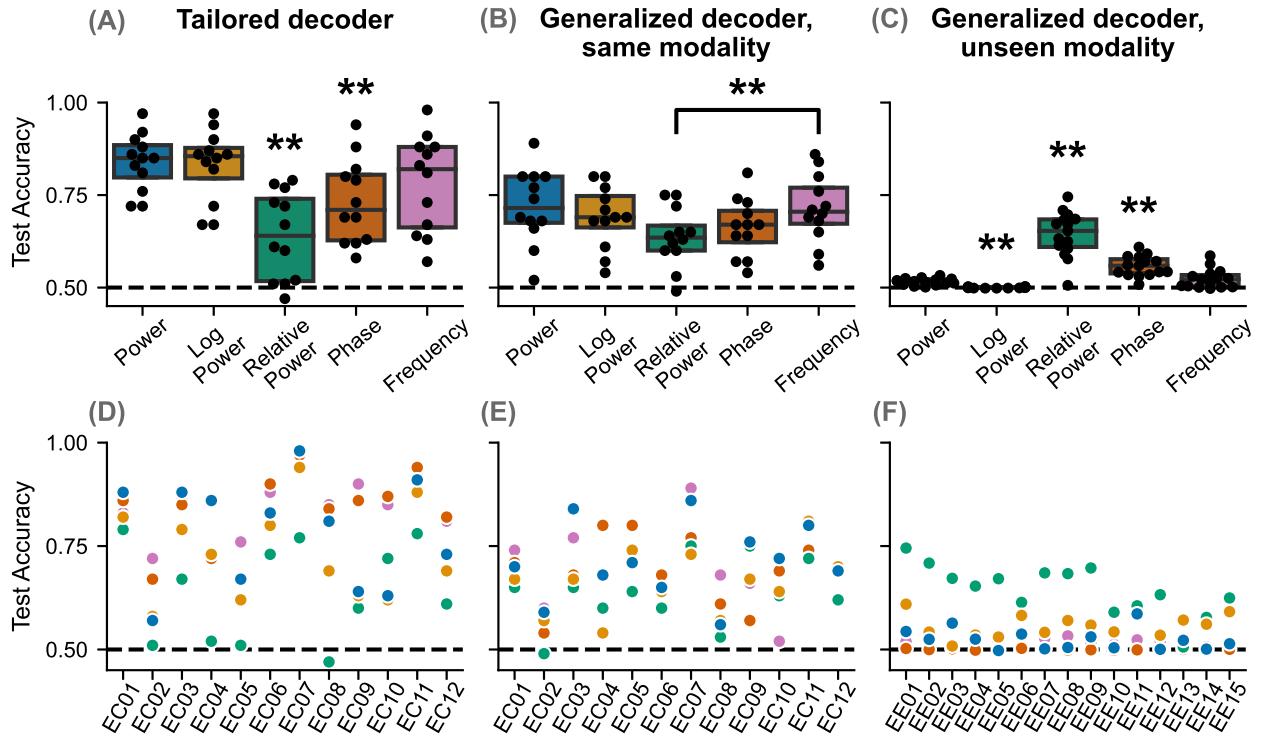
**Table A.2: Training times across decoder types.** The average training time, in seconds, to train each decoder is shown with standard deviation in parentheses. While HTNet takes that longest time to finish training in both conditions, it still takes only a few minutes to train.

**Figure A.1: Most hyperparameter selections do not greatly affect decoder performance.** For each condition, we performed 100 selections of six hyperparameters for HTNet/EEGNet decoders **(A–B)** and 25 selections of 2 hyperparameters for random forest **(C–D)**. We estimated performance using accuracy on a validation dataset, averaged over the 36 data folds. Blue dots indicate the parameter values for the trial with the highest validation accuracy in each condition. In general, our hyperparameter selections do not substantially alter decoder performance, except when selecting model type (HTNet vs. EEGNet).

|                   | HTNet        | EEGNet       |
| ----------------- | ------------ | ------------ |
| Tailored Decoder  | 55.0 (32.5)  | 72.1 (46.9)  |
| Same Modality     | 9.6 (8.8)    | 6.6 (8.3)    |
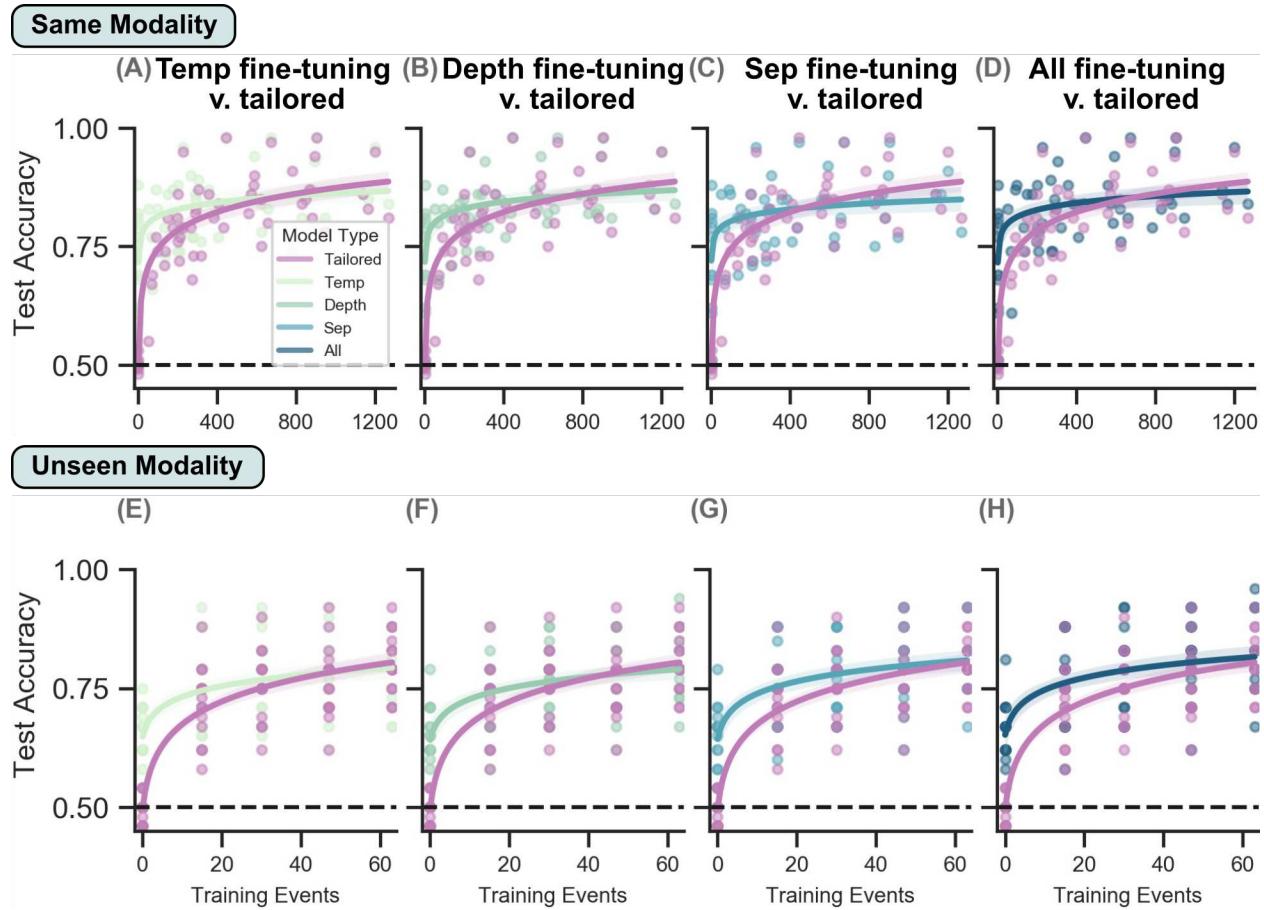
**Table A.3: Number of epochs during training for neural network decoders.** The average number of training epochs are shown for HTNet and EEGNet decoders with standard deviation in parentheses. Because both HTNet and EEGNet had early stopping criteria during training, the number of epochs used during training varied across runs. In general, both decoders trained over a similar number of epochs. Interestingly, same modality decoders needed fewer than 10 training epochs on average, far fewer than the number of training epochs during tailored decoding.

**Figure A.2: HTNet can compute a variety of spectral measures.** HTNet's Hilbert transform layer allows us to compute instantaneous power, phase, and frequency measures that can be used for decoding. During tailored decoding, relative power and phase performed significantly worse than all other measures used ($p < 0.01$ for all; Wilcoxon signed-rank test with false discovery rate correction). For same modality, we only found a significant difference between instantaneous frequency and relative power. In the unseen modality condition, relative power, phase, and log power each performed significantly different from the other measures ($p < 0.01$), with relative phase having the highest test accuracy. Our findings indicate that relative power and phase are sub-optimal measures for decoding within our ECoG dataset, but they generalize well across recording modalities.

| | Power | Log Power | Relative Power | Phase | Frequency |
|---|---|---|---|---|---|
| Tailored Decoder | 117.7 (89.4) | 112.8 (77.3) | 69.9 (42.1) | 485.0 (458.7) | 549.6 (508.3) |
| Same Modality | 430.3 (173.1) | 431.0 (199.9) | 367.3 (54.6) | 1393.1 (601.9) | 1311.7 (450.9) |
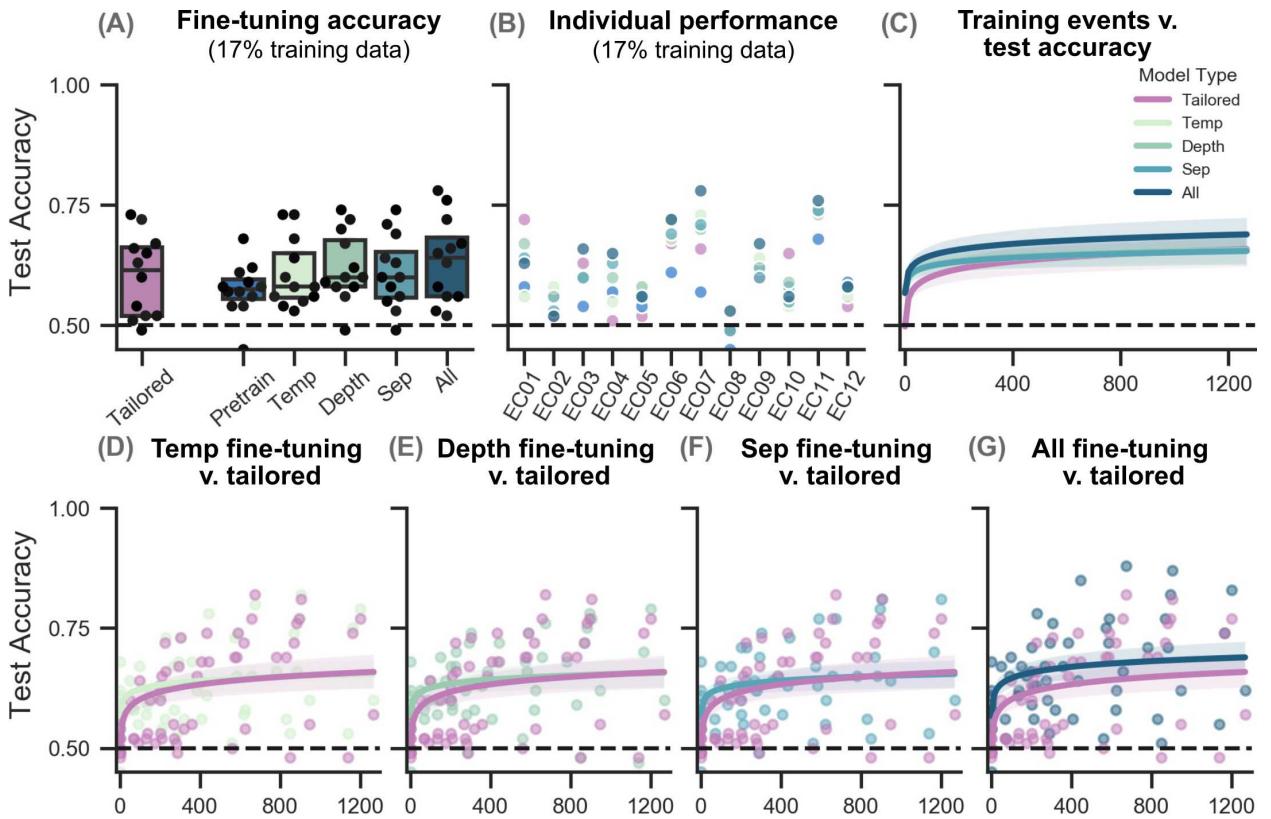
**Table A.4: HTNet training times when different spectral measures are used.** The average training time, in seconds, to train HTNet is shown when different power, phase, and frequency measures are used (standard deviation in parentheses). While all spectral power measures result in similar training times, instantaneous phase and frequency take approximately four times as long to train.

**Figure A.3: HTNet decoder performance as the number of training events varies, separated by fine-tuning approach.** The relationship between test accuracy and the number of events used for fine-tuning is shown separately for each of the four fine-tuning approaches: **(A, E)** temporal convolution (Temp), **(B, F)** depthwise convolution (Depth), **(C, G)** separable convolution (Sep), and **(D, H)** all trainable layers (All). Each fine-tuning approach is compared to the performance of a randomly-initialized, tailored decoder trained on the same test participant data. Logarithmic lines of best fit are shown for each decoder type, with shading indicating the 95% confidence interval of the slope. Dots denote the single fold test accuracy when trained on either 17%, 33%, 50%, or 67% of available data.

|                  | Power       | Log Power   | Relative Power | Phase         | Frequency   |
|------------------|-------------|-------------|----------------|---------------|-------------|
| Tailored Decoder | 55.0 (32.5) | 46.8 (24.8) | 17.4 (10.2)    | 55.0 (43.2)   | 57.5 (37.7) |
| Same Modality    | 9.6 (8.8)   | 8.0 (6.1)   | 8.0 (4.2)      | 16.1 (15.7)   | 12.8 (11.3) |

**Table A.5: Number of epochs when training HTNet with different spectral measures.** The average number of epochs needed to train HTNet decoders are shown when various spectral measures are used (standard deviation in parentheses). For all measures except relative power, tailored decoding needs about 5–6 times as many training epochs as same modality decoding. The tailored relative power decoders used one third as many training epochs as the decoders with other spectral measures and might have substantially benefited from having more time to train.
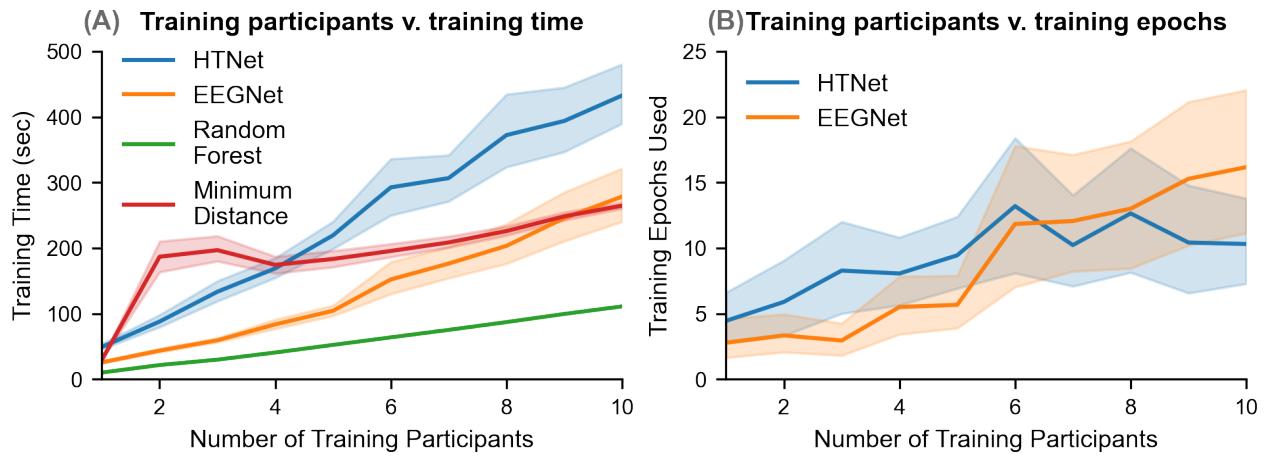


**Figure A.4: HTNet decoders transfer from EEG to ECoG participants.** Here, we tested HTNet's ability to transfer from EEG to ECoG data, instead of from ECoG to EEG data as was done in the unseen modality condition. We used relative power with HTNet to transfer between recording modalities, so decoder performance here differs from the tailored and same modality conditions when just power was used. **(A–B)** Fine-tuning all layers improves upon pretrained test accuracy the most compared to the other fine-tuning approaches. **(C)** Fine-tuning all layers on ∼50 events results in decoders with accuracies approaching the performance of the best tailored decoders that were trained on hundreds of events. **(D–G)** We also show fine-tuning curves separated by fine-tuning approach, with each dot indicating the test accuracy of a single fold when trained on either 17%, 33%, 50%, or 67% of available data.

|  | Temp | Depth | Sep | All |
|---|---|---|---|---|
| Number of parameters | 532 | 5700 | 4940 | 12238 |
| Same Modality | 62.9 (52.5) | 74.8 (72.8) | 51.7 (49.2) | 66.6 (54.3) |
| Unseen Modality (ECoG to EEG) | 15.9 (12.0) | 10.1 (7.5) | 10.4 (6.8) | 10.0 (5.9) |
| Unseen Modality (EEG to ECoG) | 107.3 (153.3) | 43.9 (39.5) | 37.3 (35.3) | 53.4 (48.2) |

**Table A.6: Number of parameters and time to fine-tune pretrained HTNet decoders.** The average training time for our four fine-tuning approaches is shown when training on 33% of available events (standard deviation in parentheses). We performed fine-tuning separately on HTNet's temporal (Temp), depthwise (Depth), and separable (Sep) convolutions, along with fine-tuning all trainable layers (All). Despite having 2–23 times as many parameters as the other approaches, fine-tuning all layers doesn't substantially increase training time on average.

|  | Temp | Depth | Sep | All |
|---|---|---|---|---|
| Number of parameters | 532 | 5700 | 4940 | 12238 |
| Same Modality | 29.8 (17.5) | 66.2 (41.8) | 54.8 (32.9) | 30.8 (18.4) |
| Unseen Modality (ECoG to EEG) | 83.4 (78.0) | 51.0 (55.6) | 70.8 (71.2) | 25.8 (38.1) |
| Unseen Modality (EEG to ECoG) | 53.0 (66.4) | 28.8 (23.8) | 29.4 (22.0) | 12.0 (11.0) |

**Table A.7: Number of training epochs when fine-tuning pretrained HTNet decoders.** The average number of training epochs during HTNet fine-tuning is shown when training on 33% of available events (standard deviation in parentheses). We separately fine-tuned HTNet's temporal (Temp), depthwise (Depth), and separable (Sep) convolutions, along with fine-tuning all trainable layers (All). While fine-tuning individual convolutional layers took a similar number of training epochs, fine-tuning all trainable layers only required about half as many training epochs on average.
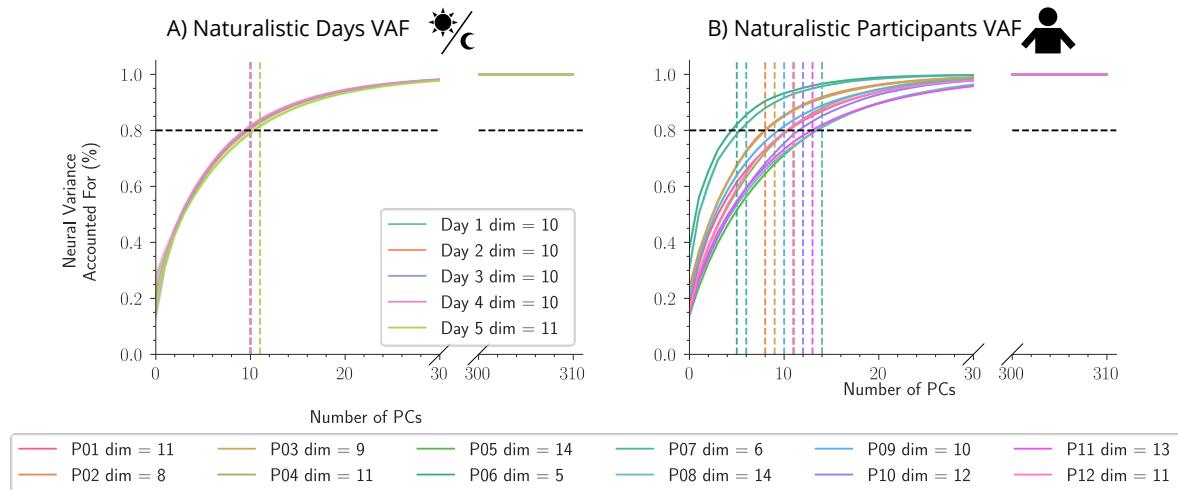
**Figure A.5: Decoder training times and epoch numbers for various number of training participants.**
**(A)** As expected, average training time across folds increases with the number of training participants (shading shows 95% confidence interval). While all decoders take under five minutes to train, HTNet's training time is substantially higher than the other decoder types when many training participants are used. **(B)** The number epochs needed for training increases slightly with more training participants, but always remains below 20 epochs on average (shading shows 95% confidence interval).
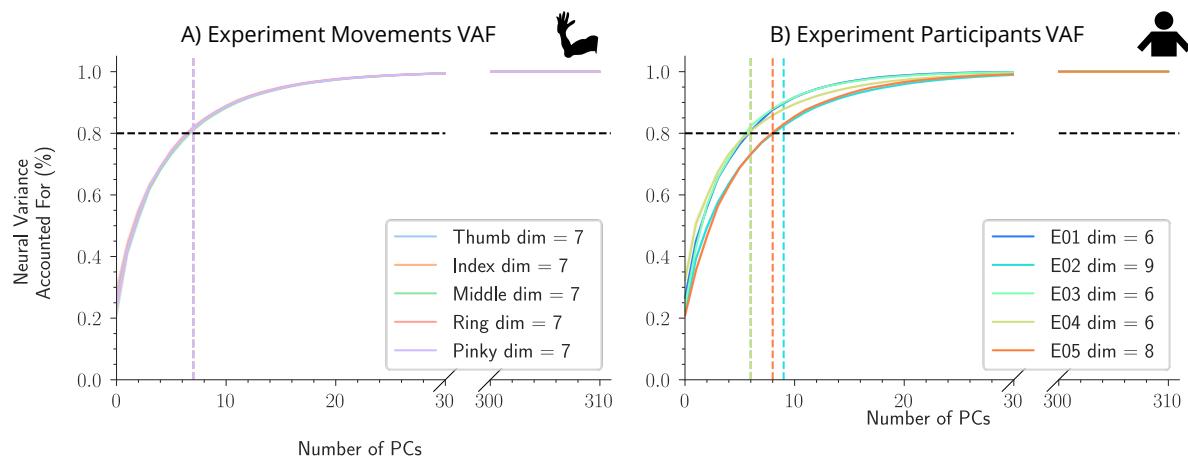
# Chapter B

# Appendix Two
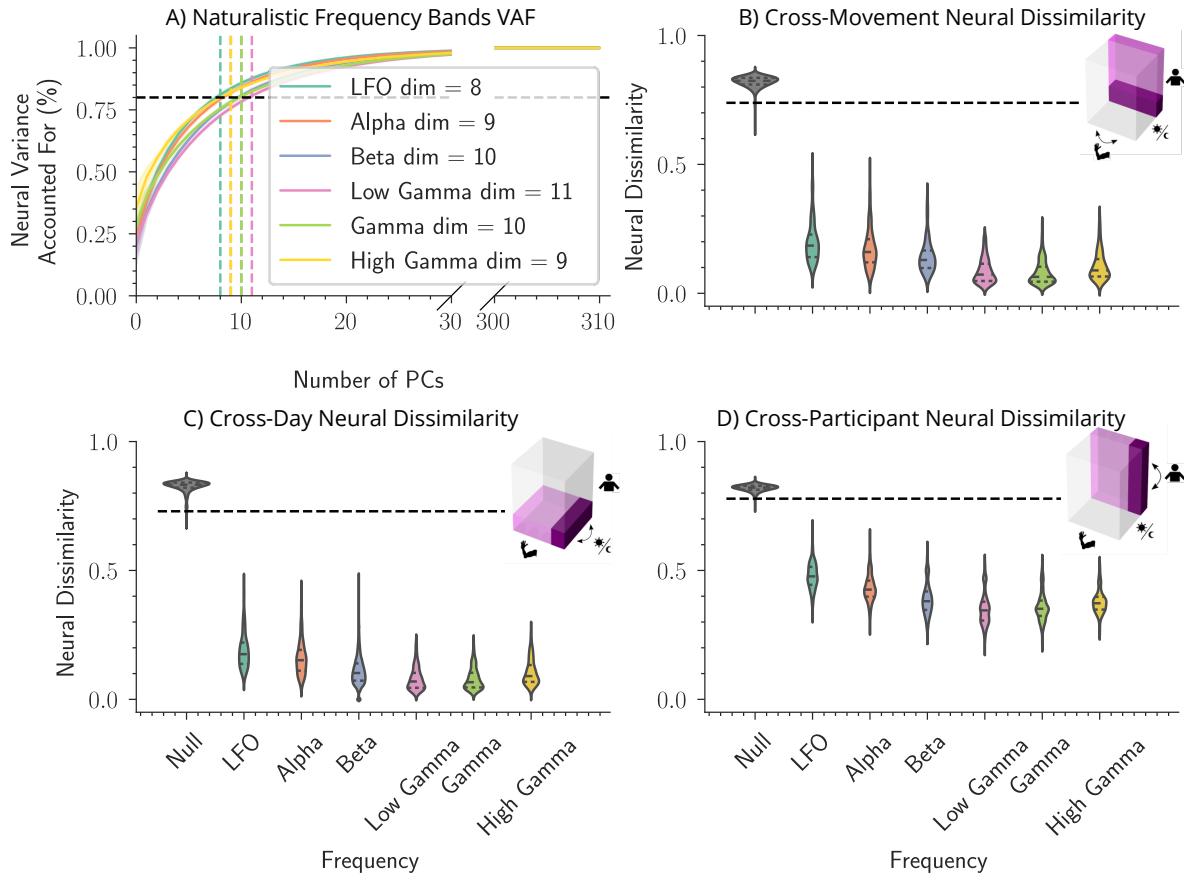
Appendix for Chatper 5 - Neural Manifolds of Human Intracranial Recordings During Naturalistic Arm Movements
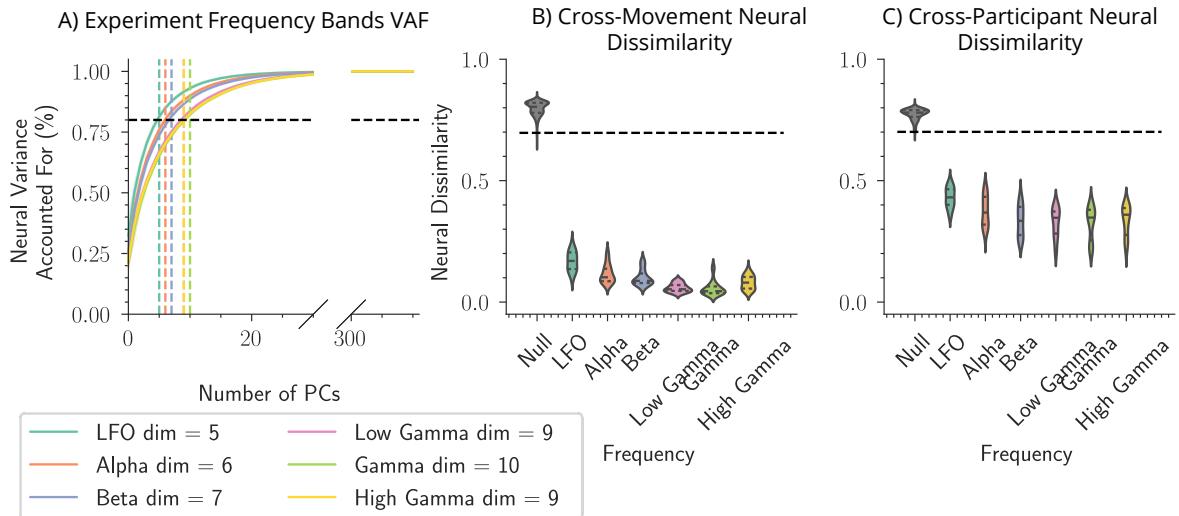


**Figure B.1: Average VAF across the naturalistic movement data dimensions show that the neural manifolds are consistently low-dimensional.** VAF was calculated by getting the mean variance across the two remaining data dimensions. **A)** Manifolds across the different days remain consistently low-dimensional, with 80% VAF at k = 10 dimensions for all days but day 5, which has a dimensionality of 11. The VAF for each day is averaged over both participants and movement types. **B)** Manifolds for each participant are also low-dimensional, with the lowest dimensionality at 5 for participant P06 and the highest at 14 for participants P05 and P08. The VAF for each participant is averaged over movement types and days.
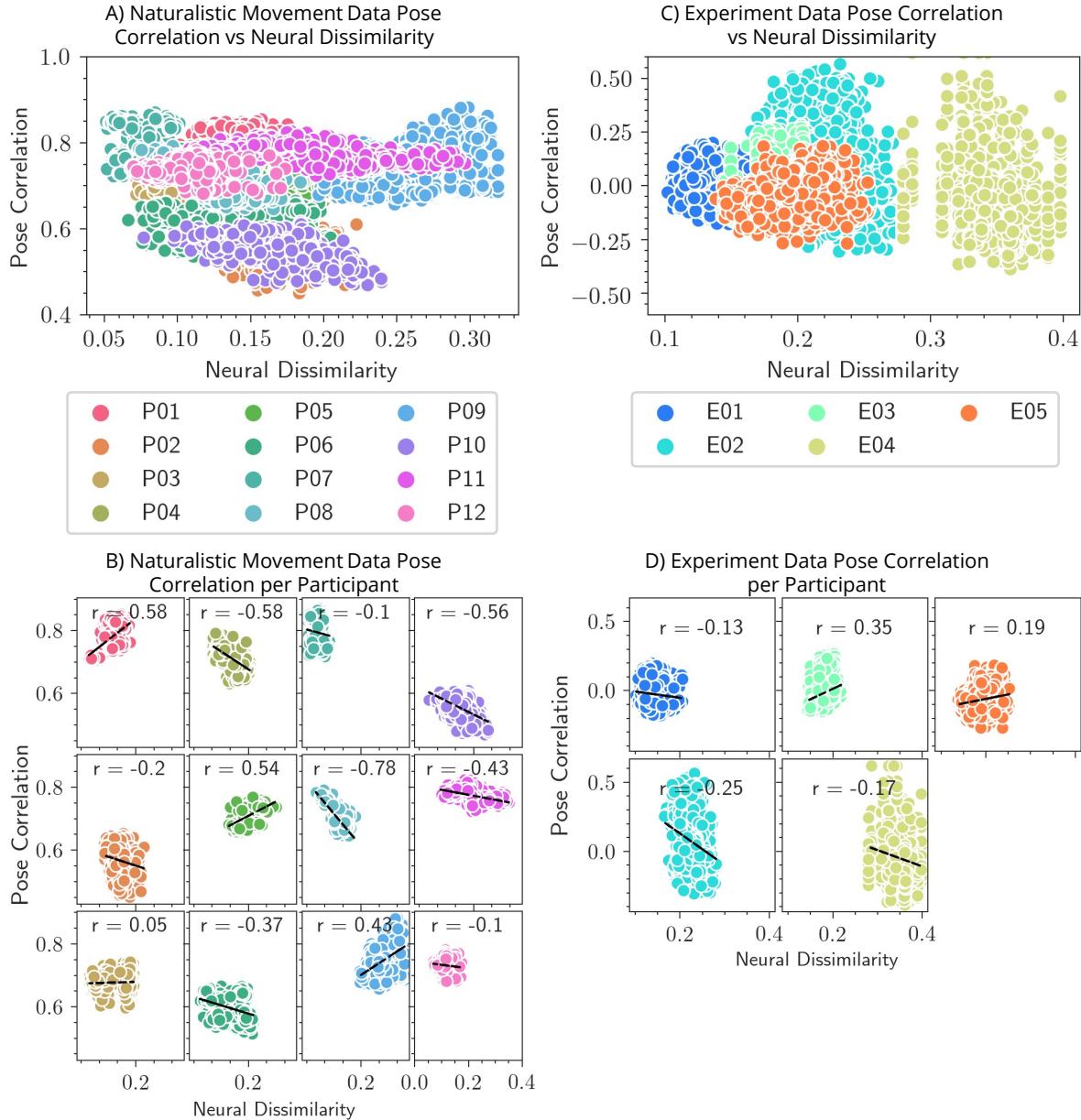
**Figure B.2: Average VAF in the experimentally-controlled data shows that the experimentally-controlled neural manifolds are even lower-dimensional than the naturalistic movement data. A)** The VAF by movement type in the experimentally-controlled data shows that the manifold dimensions are quite small. All of the five finger flexions in the dataset have a dimensionality of 7. **B)** The VAF by experimentally-controlled participant is also quite small, with most participants showing 80% VAF by dimension 6.

**Figure B.3: Neural manifolds in the naturalistic movement data remain low-dimensional and more aligned than chance across frequency bands.** **A)** VAF across frequencies remains low dimensional in the naturalistic movement data. The lowest dimension across frequencies is 8 for LFO, and the highest dimension is 11 for Low Gamma. **B)** Cross-movement neural dissimilarity is still lower than chance across all frequency bands. The higher frequency bands tend to show show slightly lower neural dissimilarity (LFO mean = 0.199, std = 0.078; Alpha mean = 0.174, std = 0.078; Beta mean = 0.142, std = 0.063; Low Gamma mean = 0.086, std = 0.048; Gamma mean = 0.080, std = 0.046; High Gamma mean = 0.106, std = 0.055; Overall mean = 0.131, std = 0.077). **C)** Cross-day neural dissimilarity follows a similar trend, with all frequency bands well below chance and higher frequencies showing less dissimilarity (LFO mean = 0.189, std = 0.070; Alpha mean = 0.160, std = 0.066; Beta mean = 0.111, std = 0.056; Low Gamma mean = 0.080, std = 0.043; Gamma mean = 0.079, std = 0.041; High Gamma mean = 0.105, std = 0.050; Overall mean = 0.111, std = 0.056). **D)** Cross-participant neural dissimilarity is also lower than chance across frequency bands, and still slightly higher than cross-movement or cross-days comparisons (LFO mean = 0.478, std = 0.051; Alpha mean = 0.431, std = 0.051; Beta mean = 0.388, std = 0.061; Low Gamma mean = 0.351, std = 0.062; Gamma mean = 0.358, std = 0.054; High Gamma mean = 0.378, std = 0.046; Overall mean = 0.397, std = 0.070).
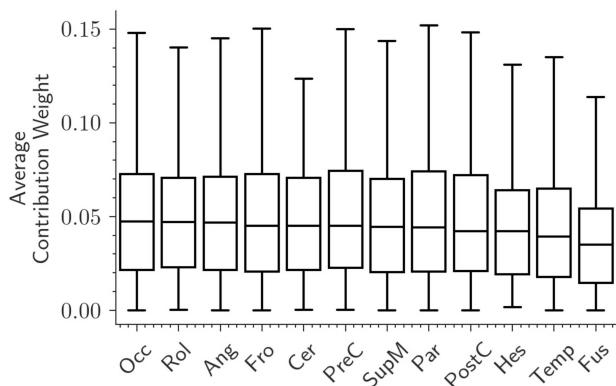
**Figure B.4: Experimentally-controlled data also shows low-dimensional well aligned neural manifolds across various frequency bands. A)** The VAF from PCA on the six frequency bands has low dimensionality (highest K = 10, lowest K = 5, where 80% of variance is accounted for). **B)** Cross-movement neural dissimilarity on the experimentally-controlled data is below chance on all frequency bands, with higher frequency bands showing lower neural dissimilarity (LFO mean = 0.168, std = 0.041; Alpha mean = 0.116, std = 0.039; Beta mean = 0.100, std = 0.032; Low Gamma mean = 0.058, std = 0.018; Gamma mean = 0.056, std = 0.030; High Gamma mean = 0.080, std = 0.029; Overall mean = 0.096, std = 0.050). **C)** Cross-participant neural dissimilarity on the experimentally-controlled data is also below chance across frequency bands, and also shows a similar trend for higher frequency bands (LFO mean = 0.431, std = 0.042; Alpha mean = 0.374, std = 0.062; Beta mean = 0.334, std = 0.064; Low Gamma mean = 0.327, std = 0.061; Gamma mean = 0.325, std = 0.068; High Gamma mean = 0.336, std = 0.062; Overall mean = 0.354, std = 0.071).

**Figure B.5: Pose correlation does not show a significant relationship to neural dissimilarity.** See Section Movement Similarity for details on how we calculated the pose correlations. **A)** Overall pose data correlations vs neural dissimilarity for all participants in the naturalistic movement dataset. Comparisons are highly clustered by participant. Each dot represents a bootstrapped sample of 80% of the participants data. Pose correlation on the y-axis is the average correlation between poses in the pairwise movement comparisons. For example, one dot may represent the average correlation between all leftward reaching pose data and all rightward reaching pose data. Neural dissimilarity is calculated the same as the rest of the paper. **B)** Broken up plots of the pose correlation vs neural dissimilarity for each participant. The pearson r-correlation for each participant is overlaid in each plot. There is no consistent trend in r-score across participants, indicating that there is no clear relationship between neural dissimilarity and the movement behavior. **C)** Overall pose data correlations vs neural dissimilarity for all participants in the experimentally-controlled dataset. Participants in the experimentally-controlled dataset also show strong clustering. **D)** Broken up plots of the pose correlation vs neural dissimilarity for each participant in the experimentally-controlled dataset. Correlations are also inlaid on each participants subplot.

**Figure B.6: The experimentally-controlled dataset shows the highest average contributions from occipital, rolandic and angular regions.** ROI contribution weights ordered by median weights. Weights are averaged over participants, movements and the PC dimensions.